G2 Event Manager

User's Guide

Version 2.3 Rev. 0

G2 Event Manager User's Guide, Version 2.3 Rev. 0
May 2007

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.

Copyright (c) 2007 Gensym Corporation

All rights reserved. No part of this document may be reproduced, stored in a retrieval system, translated, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of Gensym Corporation.

Gensym®, G2®, Optegrity®, and ReThink® are registered trademarks of Gensym Corporation.

NeurOn-Line™, Dynamic Scheduling™ G2 Real-Time Expert System™, G2 ActiveXLink™, G2 BeanBuilder™, G2 CORBALink™, G2 Diagnostic Assistant™, G2 Gateway™, G2 GUIDE™, G2GL™, G2 JavaLink™, G2 ProTools™, GDA™, GFI™, GSI™, ICP™, Integrity™, and SymCure™ are trademarks of Gensym Corporation.

Telewindows is a trademark or registered trademark of Microsoft Corporation in the United States and/or other countries. Telewindows is used by Gensym Corporation under license from owner.

This software is based in part on the work of the Independent JPEG Group.

Copyright (c) 1998-2002 Daniel Veillard. All Rights Reserved.

SCOR® is a registered trademark of PRTM.

License for Scintilla and SciTE, Copyright 1998-2003 by Neil Hodgson, All Rights Reserved.

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (`http://www.openssl.org/`).

All other products or services mentioned in this document are identified by the trademarks or service marks of their respective companies or organizations, and Gensym Corporation disclaims any responsibility for specifying which marks are owned by which companies or organizations.

Gensym Corporation
52 Second Avenue
Burlington, MA 01803 USA
Telephone: (781) 265-7100
Fax: (781) 265-7101

Part Number: DOC005-230

Contents

Preface **xiii**

About this Guide **xiii**

Audience **xiii**

Conventions **xiv**

Related Documentation **xv**

Customer Support Services **xviii**

Chapter 1 **Introduction to the G2 Event Manager** **1**

Introduction **2**

Architecture **4**

Loading GEVM **6**

Event and Message Classes **6**

- Non-Abstract Classes of Event Blackboard **7**
- Non-Abstract Classes of Operator Messages **7**
- Message Attributes **9**

Event Notification **10**

Event and Message Queues **11**

Message Views **12**

Logging Messages **13**

Message Escalation **13**

Event Reports and Statistical Reports **14**

APIs **16**

Programmer's Interface and Settings **16**

Chapter 2 Module Settings 21

Introduction **21**

gevm-module-settings **22**

Configuration File **30**

G2 Error Handling as Operator Messages **33**

Custom Message Colors **34**

Custom Message Priority Escalation **38**

Custom Message Correlation **40**

Chapter 3 User Preferences 41

Introduction **41**

gevm-user-preferences **43**

gevm-user-filter **55**

Methods **62**
- gevm-user-preferences::gevm-email-notification **63**

Chapter 4 Events 65

Introduction **65**

Event Configuration **67**

Event Notification **68**

Event Class Definition **70**
- gevm-event-class **71**

Event Creation Operations **73**
- gevm-event-class::gevm-create-event **74**
- gevm-event-class::gevm-create-event **76**
- gevm-event-class::gevm-create-event **78**
- gevm-event-class::gevm-create-event **80**
- gevm-event-class::gevm-create-event-and-insert-into-queue **82**
- gevm-event-class::gevm-create-event-and-insert-into-queue **84**
- gevm-event-class::gevm-create-event-and-insert-into-queue **86**
- gevm-event-class::gevm-create-event-and-insert-into-queue **88**

Event Classes **90**
- gevm-action **91**
- gevm-action-done **94**
- gevm-action-to-do **97**
- gevm-advisory **100**
- gevm-advisory-message **103**

Chapter 4 Events *(continued)*

Event Classes ***(continued)***
- gevm-alarm **106**
- gevm-calculated-alarm **109**
- gevm-change-in-process-state-alarm **112**
- gevm-command **115**
- gevm-conclusion **118**
- gevm-deviation-alarm **121**
- gevm-discrete-signal-alarm **124**
- gevm-event **127**
- gevm-event-state **130**
- gevm-external-fault **132**
- gevm-fault **135**
- gevm-g2-error-message **138**
- gevm-inferred-fault **141**
- gevm-instrumentation-alarm **144**
- gevm-internal-fault **147**
- gevm-limit-alarm **150**
- gevm-message **153**
- gevm-notification-message **156**
- gevm-rate-of-change-alarm **159**
- gevm-root-cause **162**
- gevm-state-change **165**

Event Management Operations **168**
- gevm-event::gevm-acknowledge **170**
- gevm-event::gevm-change-priority **171**
- gevm-event::gevm-create-sub-event **172**
- gevm-event::gevm-create-sub-event **174**
- gevm-event::gevm-create-sub-event **176**
- gevm-event::gevm-create-sub-event **178**
- gevm-event::gevm-delete-event **180**
- gevm-event::gevm-email-notification **181**
- gevm-event::gevm-execute-action **182**
- gevm-event::gevm-get-event-color **183**
- gevm-event::gevm-get-event-states **184**
- gevm-event::gevm-get-initiating-item **185**
- gevm-event::gevm-get-sub-event-count **186**
- gevm-event::gevm-get-sub-event-count **187**
- gevm-event::gevm-get-sub-events **188**
- gevm-event::gevm-get-sub-events **189**
- gevm-event::gevm-get-superior-event-count **190**
- gevm-event::gevm-get-superior-event-count **191**
- gevm-event::gevm-get-superior-events **192**
- gevm-event::gevm-get-superior-events **193**
- gevm-event::gevm-get-target **194**

Chapter 4 Events *(continued)*

Event Management Operations *(continued)*
- gevm-event::gevm-remove-event **195**
- gevm-event::gevm-show-initiator **196**
- gevm-event::gevm-show-target **197**
- gevm-event::gevm-subordinate-event **198**
- gevm-event::gevm-subordinate-event **199**
- gevm-event::grtl-can-delete **200**
- gevm-acknowledged **201**
- gevm-event-has-sub-event **202**
- gevm-event-is-sub-event **203**
- gevm-get-event-class-names **204**
- gevm-has-initiating-item **205**
- gevm-has-target **206**

Target Event Operations **207**
- item::gevm-acknowledge-all-events-for-target **208**
- item::gevm-acknowledge-events-for-target **209**
- item::gevm-acknowledge-events-for-target **210**
- item::gevm-delete-all-events-for-target **211**
- item::gevm-delete-collected-events-created-outside-time-period-for-target **212**
- item::gevm-delete-collected-events-created-within-time-period-for-target **213**
- item::gevm-delete-events-for-target **214**
- item::gevm-delete-events-for-target **215**
- item::gevm-get-collected-events-created-outside-time-period-for-target **216**
- item::gevm-get-collected-events-created-within-time-period-for-target **218**
- item::gevm-get-collected-events-updated-outside-time-period-for-target **220**
- item::gevm-get-collected-events-updated-within-time-period-for-target **222**
- item::gevm-get-event-count-for-target **224**
- item::gevm-get-event-count-for-target **225**
- item::gevm-get-events-for-target **226**
- item::gevm-get-events-for-target **227**
- item::gevm-get-events-for-target **228**
- item::gevm-remove-all-events-for-target **229**
- item::gevm-remove-events-for-target **230**

Chapter 4 Events *(continued)*

Initiating Event Operations **231**
- item::gevm-acknowledge-all-events-for-initiator **232**
- item::gevm-acknowledge-events-for-initiator **233**
- item::gevm-acknowledge-events-for-initiator **234**
- item::gevm-delete-all-events-for-initiator **235**
- item::gevm-delete-collected-events-created-outside-time-period-for-initiator **236**
- item::gevm-delete-collected-events-created-within-time-period-for-initiator **237**
- item::gevm-delete-events-for-initiator **238**
- item::gevm-delete-events-for-initiator **239**
- item::gevm-get-collected-events-created-outside-time-period-for-initiator **240**
- item::gevm-get-collected-events-created-within-time-period-for-initiator **242**
- item::gevm-get-collected-events-updated-outside-time-period-for-initiator **244**
- item::gevm-get-collected-events-updated-within-time-period-for-initiator **246**
- item::gevm-get-event-count-for-initiator **248**
- item::gevm-get-event-count-for-initiator **249**
- item::gevm-get-events-for-initiator **250**
- item::gevm-get-events-for-initiator **251**
- item::gevm-get-events-for-initiator **252**
- item::gevm-remove-all-events-for-initiator **253**
- item::gevm-remove-events-for-initiator **254**

Escalation Operations **255**
- gevm-event::gevm-clear-reevaluation-scheme **256**
- gevm-event::gevm-get-reevaluation-scheme **257**
- gevm-event::gevm-set-reevaluation-scheme **258**
- gevm-disable-event-escalation **259**
- gevm-disable-reevaluate-period **260**
- gevm-enable-event-escalation **261**
- gevm-set-reevaluate-period **262**

Customization **263**
- Creating Custom Message Classes **263**
- Customizing Message Correlation **264**

Chapter 5 Queues 267

Introduction **267**

Queue Configuration **269**

Event Notification **271**

GEVM Queue Class and Operations **272**

gevm-gqs-queue **274**
gevm-gqs-queue::gevm-add-event-to-queue **279**
gevm-gqs-queue::grtl-initialize **280**
gevm-gqs-queue::gevm-acknowledge-all-events **281**
gevm-gqs-queue::gevm-acknowledge-events **282**
gevm-gqs-queue::gevm-delete-all-events **283**
gevm-gqs-queue::gevm-delete-collected-events-created-outside-time-period **284**
gevm-gqs-queue::gevm-delete-collected-events-created-outside-time-period-for-initiator-and-target **285**
gevm-gqs-queue::gevm-delete-collected-events-created-within-time-period **287**
gevm-gqs-queue::gevm-delete-collected-events-created-within-time-period-for-initiator-and-target **288**
gevm-gqs-queue::gevm-delete-events **290**
gevm-gqs-queue::gevm-get-collected-events **291**
gevm-gqs-queue::gevm-get-collected-events-created-outside-time-period **292**
gevm-gqs-queue::gevm-get-collected-events-created-outside-time-period-for-initiator-and-target **294**
gevm-gqs-queue::gevm-get-collected-events-created-within-time-period **296**
gevm-gqs-queue::gevm-get-collected-events-created-within-time-period-for-initiator-and-target **298**
gevm-gqs-queue::gevm-get-collected-events-for-initiator-and-target **300**
gevm-gqs-queue::gevm-get-collected-events-updated-outside-time-period **301**
gevm-gqs-queue::gevm-get-collected-events-updated-outside-time-period-for-initiator-and-target **302**
gevm-gqs-queue::gevm-get-collected-events-updated-within-time-period **304**
gevm-gqs-queue::gevm-get-collected-events-updated-within-time-period-for-initiator-and-target **305**
gevm-gqs-queue::gevm-get-event-count-for-queue **307**
gevm-gqs-queue::gevm-remove-all-events **308**
gevm-gqs-queue::gevm-remove-events **309**
gevm-find-queue-by-key **310**
gevm-get-all-queues **311**

Chapter 5 Queues *(continued)*

GEVM Queue Class and Operations ***(continued)***
- gevm-get-message-properties **312**
- gevm-get-most-recent-unacknowledged-event **313**
- gevm-get-property-values-of-all-events **314**
- gevm-get-property-values-of-events **316**
- gevm-get-queues-containing-item **318**
- gevm-get-selected-property-values-of-all-events **319**
- gevm-get-selected-property-values-of-events **321**

GQS Queue Class and Operations **323**
- gqs-queue **324**
- gqs-queue::gqs-clear-queue **328**
- gqs-queue::gqs-receive-items **329**
- gqs-activate-attribute-monitoring **330**
- gqs-add-monitored-attributes **331**
- gqs-clear-queue **332**
- gqs-deactivate-attribute-monitoring **333**
- gqs-force-input-buffer-into-queue **334**
- gqs-get-collected-items **335**
- gqs-get-monitored-attributes **336**
- gqs-get-queues-containing-item **337**
- gqs-launch-view **338**
- gqs-receive-items **339**
- gqs-receive-single-item **340**
- gqs-remove-all-monitored-attributes **341**
- gqs-remove-items **342**
- gqs-remove-monitored-attributes **343**
- gqs-remove-single-item **344**
- gqs-send-items **345**
- gqs-send-single-item **346**
- gqs-number-of-collected-items **347**

Customizations **348**
- Customizing Message Queue Classes **348**
- Customizing the Message Log Handler **348**
- Customizing Database Table Creation **351**

Chapter 6 Access Tables 353

Introduction **353**

Access Table Configuration **354**
- How the Access Manager Works **354**
- Creating and Configuring the Queue Access Table **355**

Access Table Classes and Operations **358**
- gqs-queue-access-table **359**
- gqs-get-view-template **361**

Chapter 7 View Templates and View Managers 363

Introduction **364**

Template View Toolbar Buttons and Popup Menus **365**

GEVM View **368**
- View Template Configuration **368**
- View Template Class and Operations **371**
  - gevm-native-view-manager-template 372
  - gevm-native-view-manager-template::grtl-show-properties 379
- View Template Button Class and Operations **380**
  - gevm-native-action-button 382
  - gevm-check-button-selection-state 384
  - gevm-perform-button-function 385
- View Manager Class and Operation **386**
  - gevm-native-view-manager 387
  - gevm-refresh-view 390

GQSV View **391**
- View Template Configuration **391**
  - Creating a View Template **392**
  - Modifying View Colors and Border **392**
  - Modifying Template View Action Buttons **393**
  - Modifying Template View Columns **393**
  - Modifying Column Headers **396**
- View Template Classes and Operations **397**
  - gqsv-tabular-view-template 398
  - gqsv-column 400
  - gqsv-column-header 405
  - gqsv-column-or-header 409
  - gqsv-root-specification 410
  - gqsv-view-configuration 412
  - gqsv-workspace-location 415
  - gqsv-close-tabular-view 418
  - gqsv-delete-view 419
  - gqsv-number-of-viewed-items 420
  - gqsv-view-is-locked 421

Chapter 7 View Templates and View Managers *(continued)*

GQSV View ***(continued)***
- View Template Button Classes **422**
  - gqsv-close-view-button 423
  - gqsv-multiple-row-button 424
  - gqsv-single-row-button 425
  - gqsv-toolbar-button 426
- View Manager Class and Operations **428**
  - gqsv-tabular-view-manager 429
  - gqsv-activate-view-filter 430
  - gqsv-deactivate-view-filter 431
  - gqsv-get-view-filter 432
  - gqsv-set-view-filter 433
  - gevm-refresh-view 434

GQS View Manager Class and Operations **435**
- gqs-view-manager **436**
- gqs-view-manager::gqs-update-view-per-addition **438**
- gqs-view-manager::gqs-update-view-per-attribute **439**
- gqs-view-manager::gqs-update-view-per-delete **440**
- gqs-view-manager::gqs-update-view-per-removal **441**
- gqs-create-view **442**
- gqs-deregister-view **443**
- gqs-register-view **444**
- gevm-refresh-view **445**

Chapter 8 Filters and Subscriptions 447

Introduction **448**

Filter Configuration **449**

Filter and Subscription Classes **450**
- gqs-and-filter **451**
- gqs-attribute-filter **452**
- gqs-compound-filter **453**
- gqs-filter **454**
- gqs-or-filter **455**
- gqs-subscription **456**

Filter and Subscription Operations **457**
- gqs-filter::gqs-apply-filter **458**
- gqs-apply-filter **459**
- gqs-attach-filter-to-subscription **460**
- gqs-detach-filter-from-subscription **461**
- gqs-get-subscription-details **462**
- gqs-get-subscriptions-from-queue **463**
- gqs-get-subscriptions-from-queue-to-queue **464**

Chapter 8 Filters and Subscriptions *(continued)*

Filter and Subscription Operations ***(continued)***
- gqs-get-subscriptions-to-queue **465**
- gqs-populate-compound-filter **466**
- gqs-subscribe **467**
- gqs-unsubscribe **468**

Chapter 9 Logging 469

Introduction **469**

Logging Class **471**
- glf-logging-manager **472**

Logging Operations **475**
- glf-default-file-name-generator **476**
- glf-default-log-file-header-writer **477**
- glf-default-log-file-scheduler **478**
- glf-disable-logging **479**
- glf-enable-logging **480**
- glf-set-fixed-log-closing-times **481**
- glf-write-to-log-file **482**

Chapter 10 GEVM Examples 483

Introduction **484**

Message Queues **485**

Message Browsers **487**

Message Browser Configuration **492**

Native Message Browser **496**

View Templates and Managers **500**

Message Subordination **501**

XML Serialization **501**

Index 503

Preface

Describes this guide and the conventions that it uses.

About this Guide **xiii**

Audience **xiii**

Conventions **xiv**

Related Documentation **xv**

Customer Support Services **xviii**

About this Guide

This guide describes the G2 Event Manager (GEVM) module. This module provides tools for creating, managing, and displaying operator messages, using message queues and message browsers. This module also provides tools for managing a blackboard of internal event states.

Audience

This guide is for G2 developers who want to customize applications, using a set of standard application programmers' interface (API) procedures and methods, and built-in classes. It assumes familiarity with the G2 procedure language.

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
File > Save Properties	GMS and native menu choices
workspace	Glossary terms

Convention Examples	Description
`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 *underlined*. 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)
–> *transferred-items*: 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`

1

Introduction to the G2 Event Manager

Describes the G2 Event Manager (GEVM) module.

Introduction **2**

Architecture **4**

Loading GEVM **6**

Event and Message Classes **6**

Event Notification **10**

Event and Message Queues **11**

Message Views **12**

Logging Messages **13**

Message Escalation **13**

Event Reports and Statistical Reports **14**

APIs **16**

Programmer's Interface and Settings **16**

Introduction

The G2 Event Manager (GEVM) module 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.

GEVM defines events and notification messages, which include warnings, errors, and simple notifications. Message queues manage those events and messages, while views, also called message browsers, provide a user interface for viewing messages contained in queues. The message browsers are available as an end user interface both in G2 and in the Telewindows client.

In addition to messages, message queues, and message browsers, GEVM provides:

- Localization of messages.
- History tracking of state changes of message, such as priority changes, repetitions, and acknowledgment, which includes the state name and the timestamp.
- Archiving of messages to CSV files and databases.
- Metrics collection and reporting about alarm frequency and duration.
- Escalation of alarms.
- Subsuming and correlation of alarms.
- APIs covering a large range of functionality.
- The ability to reroute the contents of the G2 Message Board and Logbook to messages in queues.
- The ability to generate operator messages from G2 signalled errors.

Applications can use the GEVM module to log raw events and messages, and to inform other applications and operators about state changes. Events and notification messages can contain sub-events.

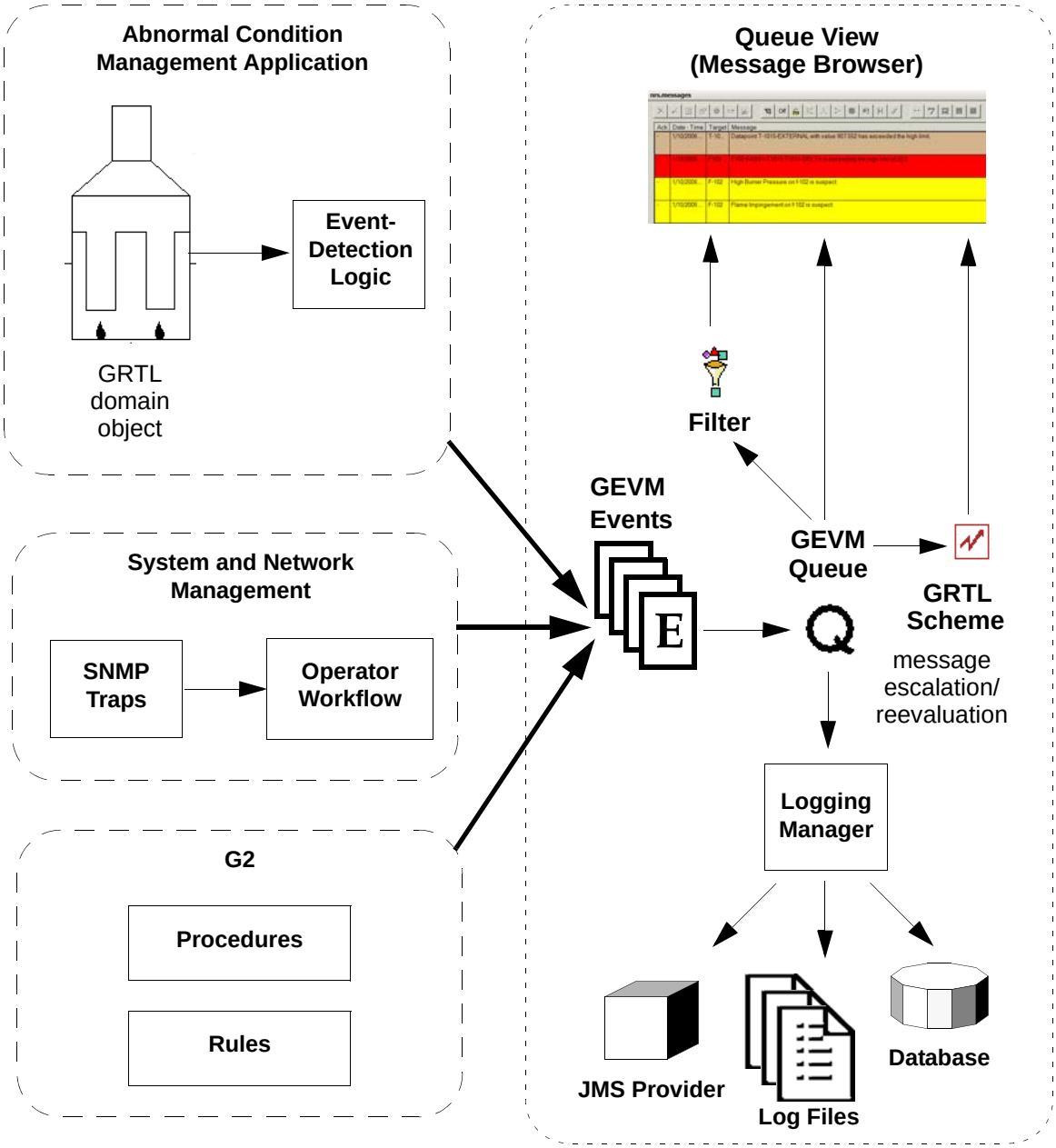
Applications can subscribe to events and reorganize and regroup messages, based on the event logic. Applications can also subscribe to particular queues and messages for notification of changes, such as new messages being inserted into a queue, messages being removed from a queues, message attributes changing. Thus, message browsers can query a message queue to retrieve current messages and to register to the queue or messages to receive notification of changes.

If enabled in the user preferences, the last unacknowledged message is displayed in the status bar. You can click to acknowledge the message and double click to display the message properties.

GEVM encapsulates and supersedes GQS and GLF, which were used in some G2 applications. This document describes the GEVM, GQS, and GLF objects.

Architecture

This figure shows a typical application architecture that uses GEVM to generate and manage raw events, to display operator messages in message browsers, and to log those events. For information on the GRTL objects, see the *G2 Run-Time Library User's Guide*.



This table describes each of the objects in the diagram.

Object	Description
GRTL external datapoint	A G2 variable that is an event source for GEVM events. GRTL external datapoints get their data through a network interface from an external DCS system.
GRTL domain object	A representation of a domain object that is an event source for GEVM events. GRTL domain objects define GRTL internal datapoints, which get their data from GRTL external datapoints.
Event-detection logic	Any type of event-detection logic that converts states of internal and external datapoint values into raw events and/or messages, for example, GEDP event-detection models or G2 system procedures.
GEVM events	An event object with associated properties such as message text, priority, and details, which goes directly to a GEVM queue. A GRTL domain object can generate different types of events, which can go to different GEVM queues. GRTL events can be generated by GRTL internal or external datapoints, as well as by GEDP event-detection models, SymCure fault models, and user-defined procedures.
GEVM queue	An internal blackboard that stores an ordered list of GEVM events. A GEVM queue can be related to one or more GQS queue views. A GEVM queue can also register with other queues with optional message filtering.
GQS filter	An object that filters GEVM events, based on their type and their parameters, so only certain events appear in a GQS queue view.
GRTL scheme	An object that implements an advanced control scheme, for example, to escalate the priority of messages, either as a G2 procedure or using a graphical language such as GEDP.

Object	Description
GLF logging manager	An object that manages logging of GEVM events to log files, databases, or a JMS provider.
GQS queue view	A graphical view associated with a GEVM queue, which displays GEVM events to operators.

Loading GEVM

To use the GEVM module, you must load or merge in `gevm.kb`, which is located in the `g2i\kbs` directory.

The `gevm-demo.kb` is located in the `g2i\examples` directory. It contains numerous demos.

Event and Message Classes

GEVM provides a class hierarchy of messages with each class relevant for a type for message, for example a root cause message, an alarm message or an action message.

There are two major categories of message classes:

- gevm-event-state: The base class for raw event states. For example, intelligent objects define many subclasses of raw events to record the state of the event detection logic. GEVM only defines the root class.
- gevm-message: The base class for operator messages. GEVM defines many subclasses of operator messages for specific purposes, for example, faults, advisory, conclusion, action or root cause messages.

When defining your own raw event state or operator message class, you must use an instance of the gevm-event-class class definition, rather than the basic G2 class definition. This is necessary for messaging to operate correctly. Instances of the gevm-event-class class definition contain additional attributes used to store information related to the class, such as the truth-value to indicate whether metrics are enabled and additional attributes of event and message instances. For details, see the *G2 Run-Time Library User's Guide*.

GEVM uses the message class, message category, message initiator, and message target to detect duplicate messages. If a duplicate message is detected, GEVM keeps only one message and updates the values of the original message, as needed, for example, to raise the repetition count or update the message text. The one exception to this rule is for messages that are instances of gevm-notification-

message. For notification messages and messages representing G2 errors, GEVM always creates duplicate messages.

All raw events and operator messages are associated with an event source and event target. The event source typically is the logic that generates the message or is the cause of the message, while the event target is a domain object such as a step in a business process, a work device, process equipment, or a sensor.

Non-Abstract Classes of Event Blackboard

The following classes of event states are non-abstract classes that can be instantiated and that appear in the event blackboard:

Class or Type of Event	Description
gevm-event-state	Raw event.

Non-Abstract Classes of Operator Messages

The following classes of operator messages are non-abstract classes that can be instantiated and appear in a message browser:

Class or Type of Event Message	Description
gevm-notification-message	Notifications to the operator.
gevm-g2-error-message	G2 error messages that should be displayed in the GEVM Message Browser, rather than in the G2 Message Board or Operator Logbook.
gevm-conclusion	Conclusions identified by the software usually in support of an advisory. For example, observation blocks in a graphical language may generate conclusion messages.
gevm-alarm	Operator advisory alarms.
gevm-limit-alarm	Alarms triggered by continuously comparing a signal with upper and/or lower limits and noting tripping by crossing the trip setting.
gevm-deviation-alarm	Alarms triggered by continuously comparing the difference between two continuous signals, for example, two probes or a probe and a set point, and noting a drift greater than a trip setting.

gevm-rate-of-change-alarm	Alarms triggered by a rate of change of a measurement or a derived variable that exceeds a trip setting.
gevm-calculated-alarm	Alarms triggered by comparing a value with a calculated or derived trip setting. This alarm type is used for advanced control such as inference technology, for example, artificial neural networks, rules, or expert systems.
gevm-discrete-signal-alarm	Alarms triggered by comparing an expected component state with an actual state, for example, a valve on or off.
gevm-change-in-process-state-alarm	Alarms triggered by noting a discrete change of state, for example, putting a process on hold or by-pass.
gevm-instrumentation-alarm	Alarms triggered by faults within the computer system or instrumentation diagnostics, such as overflow, low battery, or database not responding.
gevm-advisory	Intelligent operator advisories.
gevm-action-to-do	Actions that need to be executed either manually or automatically.
gevm-action-done	Actions that have been done either manually or automatically.
gevm-external-fault	Faults external to the application.
gevm-internal-fault	Faults internal to the application, such as a coding error.
gevm-inferred-fault	Faults that are inferred from other conditions.
gevm-root-cause	Root causes of a fault.
gevm-command	Commands that are sent to a system.

Message Attributes

This table lists all the attributes of an event and indicate whether they exist for raw events and operator message:

Attribute	Raw Events	Operator Message	Description
gevm-key	✓	✓	A unique key that is automatically generated, based on the class, type, initiator, target, and category.
gevm-category	✓	✓	A user-defined text defining the category of the message.
gevm-priority	✓	✓	A positive integer defining the priority of the message with 1 being the highest priority. Typically priorities are between 1 and 9.
gevm-repetitions	✓	✓	A positive integer that is the number of times the same message has occurred on the same class, type, initiator, target, and category.
gevm-message	✓	✓	The text of the message.
gevm-detail	✓	✓	The text of the message detail.
gevm-lifetime	✓	✓	The lifetime of the message, in seconds. If this is a positive number, the message is automatically deleted after it expires. **Note:** To balance performance with response time, messages are only reevaluated based on the reevaluation period of messages defined in the `config.txt` file. Therefore, messages may not be deleted at the exact expiration time but rather the next time messages are reevaluated. For example, if the reevaluation time is 1 hour (default) and the expiration time is 10 minutes, the message will be deleted after approximately one hour.
gevm-creation-timestamp	✓	✓	The G2 timestamp in units of seconds when the message was created, as a quantity.
gevm-last-update-timestamp	✓	✓	The G2 timestamp in units of seconds when the message was last updated, as a quantity.
gevm-advice		✓	The text of the message advice.

gevm-comments	✓	Any user-defined comment associated with the message, as a text.
gevm-assigned-to-user	✓	The user name assigned to the message.
gevm-assigned-to-group	✓	The group assigned to the message.
gevm-acknowledgment-required	✓	True if the message needs to be acknowledged.
gevm-acknowledged	✓	True if the message has been acknowledged, false otherwise.
gevm-acknowledged-by-user	✓	The name of the user that acknowledged the message.
gevm-acknowledged-timestamp	✓	The G2 timestamp when the message was acknowledged, as a quantity.

Event Notification

GEVM messages support GRTL event notification. Therefore, any message can register itself as a listener to any event or message class, or be notified of any change on any instance of a class. Listeners can also register to queues and be notified of changes to entries in the queue. For details, see the \.

GEVM extends GRTL notification by providing additional notification messages that support three other ways to notify listeners of changes on raw events or operator messages, each appropriate for different situations:

- The listener registers with a GEVM queue or GEVM queue class. In this case, the event-type attribute of the event structure is equal to the symbol gevm-queue-entry-notification, and the attribute gevm-event-type of the event specifies the subtype of the change on the raw event or message.
- The listener registers with an event source instance or class. In this case, the listener is only notified for raw events or messages when the listener is the event source of the message. The event-type attribute of the event structure is equal to the symbol gevm-event-notification, the attribute gevm-event-type of the event specifies the subtype of the change on the raw event or message, and the gevm-event attribute of the event contains the content of the original event.

- The listener registers with an event target instance or class. In this case, the listener is only notified for raw events or messages when the listener is the event target of the message. The event-type attribute of the event structure is equal to the symbol gevm-event-notification, the attribute gevm-event-type of the event specifies the subtype of the change on the raw event or message, and the gevm-event attribute of the event contains the content of the original event.

Event and Message Queues

GEVM queues are instances of the gevm-gqs-queue class or any subclass. GEVM includes two predefined queues covering two broad application areas:

- Events — Stores event states, that is, instances of gevm-event-state or any subclass, and forms a backboard of events. The contents of this queue is typically not displayed, but is rather used by correlation engines.
- Messages — Stores operator and system messages, that is, instances of gevm-message or any subclass. Messages in this queue are typically viewed by operators.

When defining your own message queue class, you must use an instance of the grtl-event-source-class-definition class definition, rather than the basic G2 class definition. This is necessary for GRTL event dispatching to work correctly. For details, see the *G2 Run-Time Library User's Guide*.

GEVM message queues inherit from both gqs-queue and glf-logging-manager, as well as other classes, which means they support both queue management of messages and message logging.

When messages are created, the following steps occur before the message is inserted into the queue:

1. Messages instances are created by using one of the gevm-create-event or gevm-create-event-and-insert-into-queue APIs.
2. If the queue is specified in the API, the message is inserted into that queue. If the queue is not specified, subclasses of gevm-event-state are inserted into the default queue named gevm-all-events-queue, while operator messages that are subclasses of gevm-message are inserted into the default queue named gevm-all-messages-queue.
3. If the event or message is a duplicate, the attributes of the original are updated, if desired, and the gevm-event-repetition event is sent to listeners. No other processing is done for duplicate messages; thus, step 4 and beyond apply to new messages only.
4. For subclasses of gevm-message, a default escalation procedure is assigned to the message, which is specified in the message-default-escalation-scheme of the active gevm-module-settings. The message escalation scheme must be an

instance of a subclass of a grtl-scheme, which means it can be implemented by using the G2 procedural language or as a GEDP diagram, which allows you to represent the escalation logic by using a graphical language. For details, see the *G2 Run-Time Library User's Guide*.

5. The message is added to the queue by calling gevm-add-event-to-queue. The message is not yet inserted into the message queue, but is instead queued up in the items awaiting insertion list (_gqs-items-awaiting-insertion) on the queue. The events and messages are queued up into this temporary queue and are only inserted into the message list (_gqs-collected-items attribute) after a latency time (gqs-update-latency).
6. The gevm-publish-event event is sent to all listeners, which includes listeners registered on the message, the message class and superior classes, the event source instance, the event source class and superior classes, and the event target and event target classes and superior classes.
7. The queue receives the above notification, calls the message correlation routine specified in the message-subordination-procedure of the active gevm-module-settings, then propagates the event to listeners of the queue and listeners to the queue class.
8. After the latency time (or less), the content of the items awaiting insertion list is moved into the collected items list of the queue, effectively queuing the raw events or operator messages. GQS propagates the notification via callbacks to the callback procedure specified on the queue and to all message browsers.
9. The default implementation of the item addition callback on gevm-gqs-queues logs the activity to the CSV file and/or database, dispatches the gevm-queue-entry-addition event to queue instance and class listeners, and deletes old messages if the length of the message queue exceeds the limit, which is given by the gevm-max-events-to-queue attribute on the queue.

Message Views

The appearance, contents, and behavior of the message browser depend on:

- The access table associated with the queue.
- View templates associated with the queue.
- Filters applied to the queue.

The access table selects the view template, based on the user name or G2-window class and the user mode.

The filters are typically configured by users to select a subset of messages by applying specified criteria to each entry. Only entries that satisfy the criteria are displayed.

A view template defines a queue view, controlling the physical appearance and behavior of the queue. Typically, a browser contains a tabular view displaying the messages and action buttons to perform operations on selected messages, such as deleting messages, acknowledging messages, or showing details of messages.

To use a view template to display the content of a queue, you have two choices depending on the desired behavior:

- To use the template view for all users and all user modes, display the properties of the queue and enter the name of your new view template in the Browser Template attribute
- To select a template view, based on the user and/or the user mode, specify an access table in the Browser Template attribute of the queue and configure the access table to select the template, based on the user name and user mode.

Logging Messages

GEVM queues can be configured to log messages to a CSV file or a database table. GEVM queues merges both the GQS queue functionality and the GLF functionality.

Message Escalation

Operator messages can be reevaluated periodically. The default implementation decrements the repetition counter. Raw events, that is, instances of gevm-event-state and subclasses, cannot be reevaluated.

The value of several GEVM parameters in the configuration file determines the frequency of the reevaluation, as well as other default behavior. For more information, see Chapter 3, “User Preferences” on page 41.

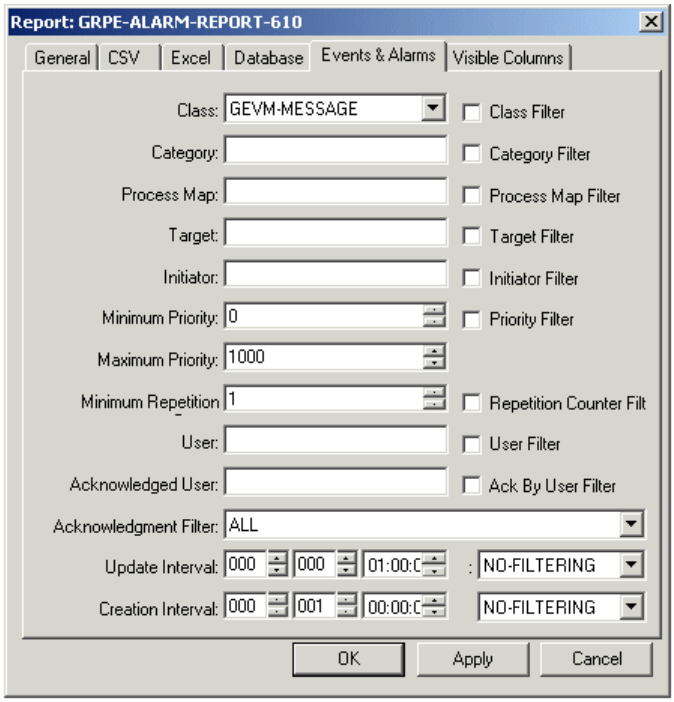
The default reevaluation procedure is specified in the gevm-module-settings. The default implementation is an instance of gevm-priority-escalation-scheme. Custom schemes need to be subclasses of grtl-scheme and must implement at least the evaluator or be an instance of a GEDP diagram. Specific logic can be assigned to each message, and the evaluator can either be implemented as a G2 procedure or as a GEDP diagram. For details, see the *G2 Run-Time Library User’s Guide* and the *G2 Event and Data Processing User’s Guide*.

Event Reports and Statistical Reports

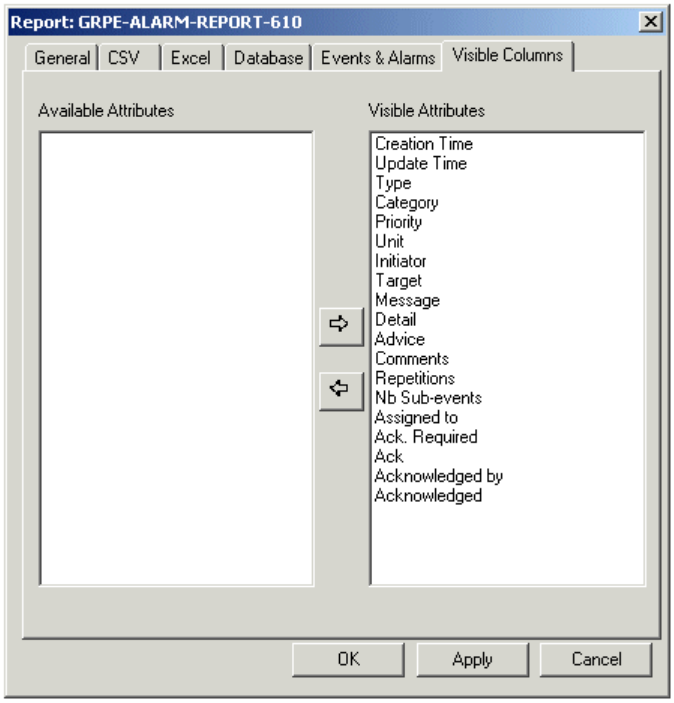
GEVM includes the following pre-defined reports:

- Alarm Report — Builds a report of gevm-message objects. You can configure filters and attributes to create custom reports. This functionality enables you to build reports such as the alarms associated with a piece of equipment or a process unit, a report of the high-priority alarms, or a report of all unacknowledged alarms.
- Alarm Statistic Report — Builds a report of statistics associated with gevm-message objects. You can configure filters and attributes to create custom reports. This functionality enables you to build reports such as the frequency of occurrence of root cause alarms, a summary of alarm frequency on a monthly basis, or the frequency and duration of alarms for a specific piece of equipment.

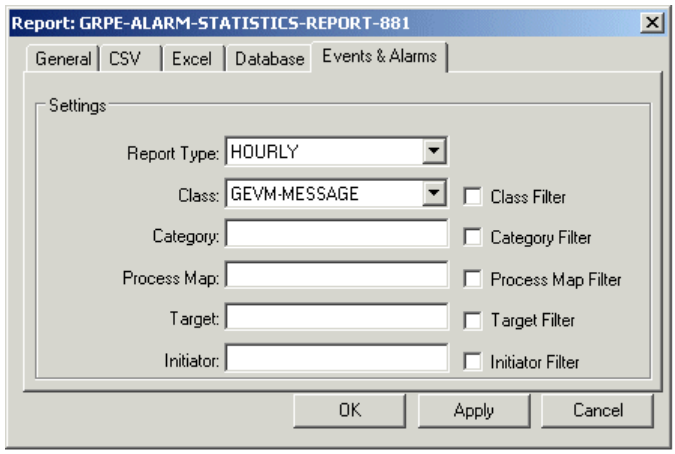
Here is the Events & Alarms tab of the properties dialog for the Alarm report:



Here is the Visible Columns tab of the properties dialog for the Alarm report:



Here is the Events & Alarms tab of the properties dialog for the Alarm Statistics report:



Note that for the Alarm Statistics report, you must enable the Enable Performance option in the System Performance Settings dialog choosing Project > System Settings > System Performance; otherwise, no statistics are collected.

For general information on creating and configuring reports, see the *G2 Reporting Engine User's Guide*.

APIs

GEVM provides many APIs to create, update, or query raw events, operator messages, and queues. The APIs are organized as follows:

- Raw Event and Operator Message APIs
  - Chapter 4, "Events" defines the top-level event class definition, event classes, and APIs for creating and managing events.
- Event and Message Queue APIs
  - Chapter 5, "Queues" defines the queue class and APIs for querying messages given a queue.
  - Chapter 6, "Access Tables" defines classes and APIs for queue views.
  - Chapter 7, "View Templates and View Managers" defines the API for managing queue views.
  - Chapter 8, "Filters and Subscriptions" defines APIs to filter and subscribe to messages in a queue.
  - Chapter 9, "Logging" defines APIs to log messages in a queue.

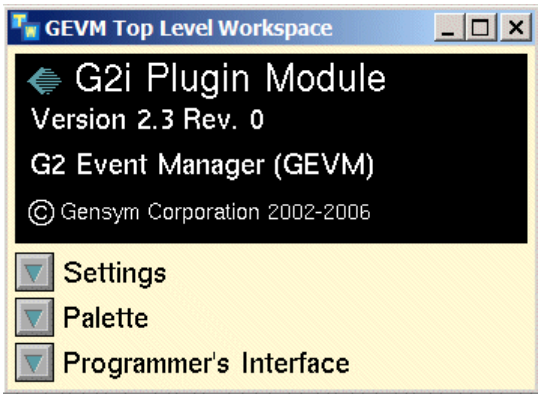
Programmer's Interface and Settings

The programmer's interface provides default methods and procedures to use as templates when creating custom methods and procedures. The module settings provides default lookup tables for message color mappings, as well as the default module settings object for the GEVM module.

Caution If you customize the GEVM module settings object, be sure to save it to a higher-level module in your module hierarchy; otherwise, when you upgrade your application to a new release, the module settings will be overwritten.

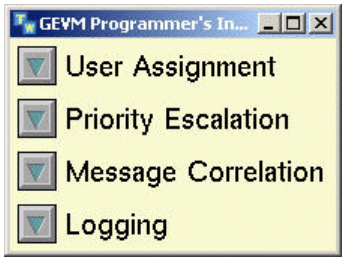
To access the GEVM programmer's interface and settings:

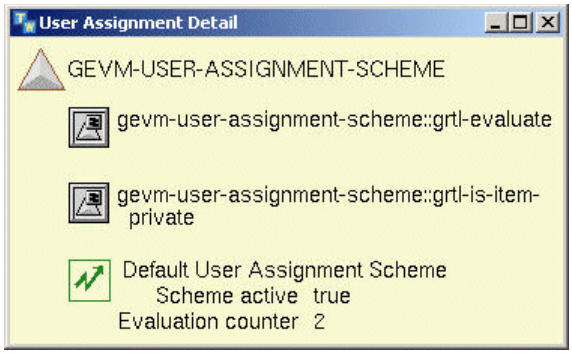
1 Get the gevm-top-level workspace.

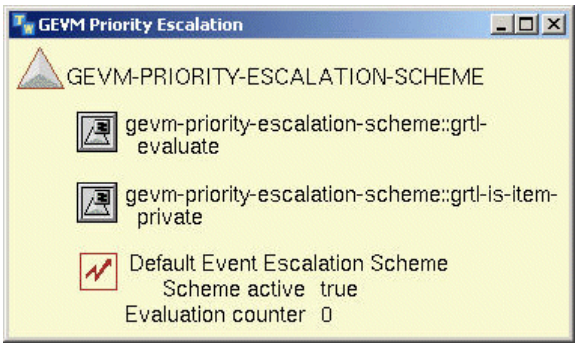


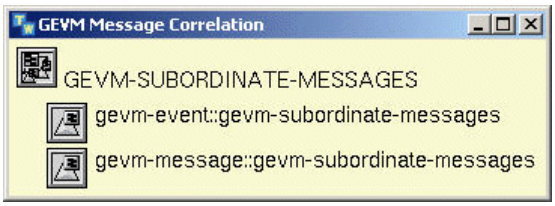
2 Click the Programmer's Interface button.

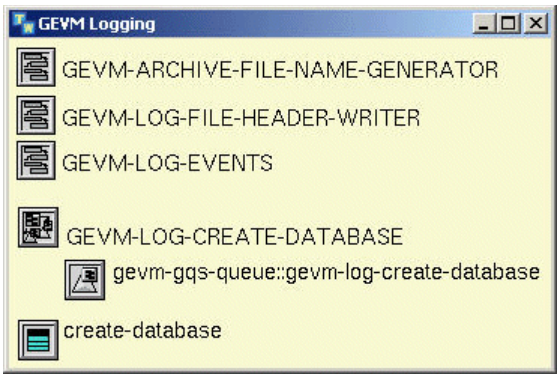
The programmer's interface contains various procedures for customizing messaging, as described in the sections that follow:





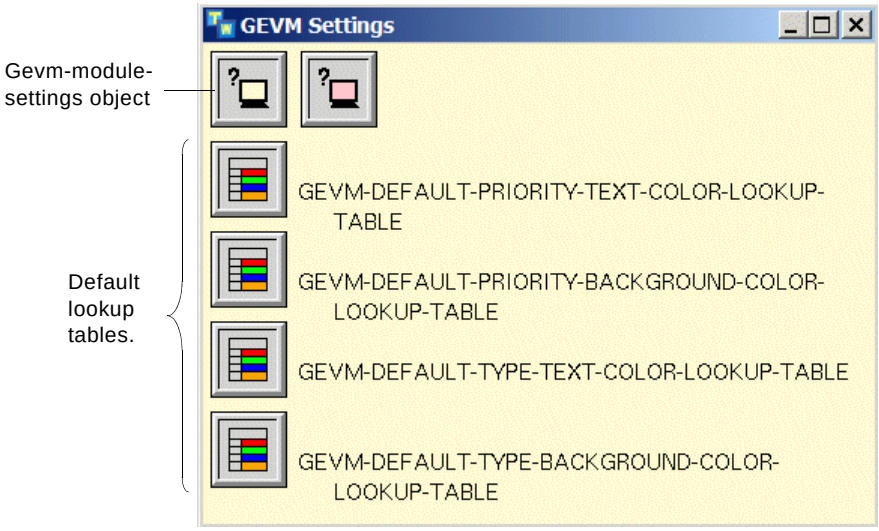

GEVM Priority Escalation
GEVM-PRIORITY-ESCALATION-SCHEME
gevm-priority-escalation-scheme::grtl-evaluate
gevm-priority-escalation-scheme::grtl-is-item-private
Default Event Escalation Scheme
Scheme active true
Evaluation counter 0


GEVM Message Correlation
GEVM-SUBORDINATE-MESSAGES
gevm-event::gevm-subordinate-messages
gevm-message::gevm-subordinate-messages


GEVM Logging
GEVM-ARCHIVE-FILE-NAME-GENERATOR
GEVM-LOG-FILE-HEADER-WRITER
GEVM-LOG-EVENTS
GEVM-LOG-CREATE-DATABASE
gevm-gqs-queue::gevm-log-create-database
create-database

3 On the gevm-top-level workspace, click the Settings button.

Here is the gevm-module-settings object with its default lookup tables, used for customizing message colors and message correlation, as described in the sections that follow:



2

Module Settings

Describes the G2 Event Manager (GEVM) module settings.

Introduction **21**

gevm-module-settings **22**

Configuration File **30**

G2 Error Handling as Operator Messages **33**

Custom Message Colors **34**

Custom Message Priority Escalation **38**

Custom Message Correlation **40**

Introduction

The gevm-module-settings object inherits GFR module settings. Upon startup, GFR locates one module settings object as the active setting, which is typically the instance in the highest level module. The active module is determined when G2 is started. Several APIs take the active module settings object into account during execution.

The gevm-module-settings object enables you to configure system-wide configurations, such as the foreground and background colors of messages, and the algorithm to correlate messages or to create operator messages for G2 signalled errors.

gevm-module-settings

Manages system configurations for the GEVM module.

Class Inheritance Path

gfr-module-settings, object, item

Attributes

Attribute	Description
maximum-event-history-states	The maximum event change history to keep for each gevm-event instance. The event change history tracks changes such as when a message was acknowledged, when its priority was changed, or when its repetition information changed.
Allowable values:	Any positive integer
Default value:	10
message-metrics-enabled	This attribute is not currently used.
Allowable values:	Any truth-value
Default value:	true
message-default-reevaluation-interval	The interval with which messages are reevaluated, in seconds.
Allowable values:	Any positive integer
Default value:	3600
Notes:	See "Configuration File" on page 30.

Attribute	Description
message-default-reevaluation-scheme	The grtl-scheme to call for each message that needs to be reevaluated. This attribute should match the scheme-key of the grtl-scheme. The default value refers to an instance of a gevm-priority-escalation-scheme implemented in GEVM.
Allowable values:	Any text
Default value:	"Default Event Escalation Scheme"
message-default- user-assignment-scheme	The grtl-scheme to call when a new message is created to assign a user or group. This attribute should match the scheme-key of the grtl-scheme. The default value refers to an instance of a gevm-user-assignment-scheme implemented in GEVM.
Allowable values:	Any text
Default value:	"Default User Assignment Scheme"
message-queue-to-queue update-latency	The latency, in seconds, to propagate new entries from one queue to another registered queue. This avoids overhead when the state and number of events change often.
Allowable values:	Any positive integer
Default value:	1

Attribute	Description
message-color-based-on	The logic to use to determine the color of messages displayed in browsers. The options are one of the following symbols: • priority — Color is based on the priority of the message with a look up in the foreground and background lookup tables specified below. • type — Color is based on the class of the message with a look up in the foreground and background lookup tables specified below. • process-map — The foreground and background colors specified on the process map object (class grtl-domain-map-repository) should be used.
Allowable values:	priority, process-map, or type
Default value:	priority
Notes:	See "Configuration File" on page 30.
reverse-message-color-if-acknowledged	When true and the message has been acknowledged, swaps the foreground and background colors.
Allowable values:	truth-value
Default value:	true
Notes:	See "Configuration File" on page 30.
cdg-messages-use-standard-color-mapping	When true, SymCure-generated messages use the color mapping defined above; otherwise uses a SymCure-specific color mapping.
Allowable values:	truth-value
Default value:	false
Notes:	See "Configuration File" on page 30.

Attribute	Description
background-color-when-acknowledged	If the message has been acknowledged and the setting reverse-message-color-if-acknowledged is true, uses this color as the background color, and the original background color as the text color.
Allowable values:	symbol
Default value:	smoke
message-priorty-text-color-lookup-table	The name of a gevm-color-lookup-table instance used to look up text colors when message-color-based-on is priority.
Allowable values:	symbol
Default value:	gevm-default-priority-text-color-lookup-table
message-priority-background-color-lookup-table	The name of a gevm-color-lookup-table instance used to lookup text background colors when message-color-based-on is priority.
Allowable values:	symbol
Default value:	gevm-default-priority-background-color-lookup-table
message-type-text-color-lookup-table	The name of a gevm-color-lookup-table instance used to lookup text colors when message-color-based-on is type.
Allowable values:	symbol
Default value:	gevm-default-type-text-color-lookup-table
message-type-background-color-lookup-table	The name of a gevm-color-lookup-table instance used to lookup text background colors when message-color-based-on is type.

Attribute	Description
Allowable values:	symbol
Default value:	gevm-default-type-background-color-lookup-table
message-subordination-procedure	The procedure or method to call to perform message correlation. This method is called as messages are created and updated. The signature of this procedure must be: my-proc (*event*: class gevm-event)
Allowable values:	symbol
Default value:	gevm-subordinate-messages
message-browser-use-basic-browser	Used at initialization time to select either a basic browser or a more advanced browser when in operator mode.
Allowable values:	truth-value
Default value:	false
message-browser-default-template-for-modeler	The name of an access table (gqs-queue-access-table) or message browser template (gevm-native-view-manager-template) to use when displaying messages in modeler mode.
Allowable values:	symbol
Default value:	gevm-modeler-msg-view-template
message-browser-default-template-for-operator	The name of an access table (gqs-queue-access-table) or message browser template (gevm-native-view-manager-template) to use when displaying messages in operator mode.
Allowable values:	symbol

Attribute	Description
Default value:	gevm-operator-msg-view-template
enable-event-metrics	When true, enables the collection of Event & Alarm metrics for gevm-message instances, including subclasses. This is a global switch; you also need to enable the collection of statistical metrics on individual gevm-message classes in the Event & Alarm Metrics configuration dialog.
Allowable values:	truth-value
Default value:	false
event-metrics-log-to-file	When true, enables logging of statistical metrics to files.
Allowable values:	truth-value
Default value:	false
event-metrics-log-to-database	When true, enables logging of statistical metrics to a database.
Allowable values:	truth-value
Default value:	false
event-metrics-log-directory	The directory on the G2 server where statistical metrics should be loaded from.
Allowable values:	text
Default value:	""
event-metrics-log-database-interface-or-pool	The key of a gdsm-database-interface or gdsm-database-connection-pool to use for database logging.
Allowable values:	text

Attribute	Description
Default value:	""
event-metrics-log-database-table-hourly-metrics	The database table name where the hourly message statistical metrics should be stored.
Allowable values:	text
Default value:	HourlyEventMetrics
event-metrics-log-database-table-daily-metrics	The database table name where the daily message statistical metrics should be stored.
Allowable values:	text
Default value:	DailyEventMetrics
event-metrics-log-database-table-monthly-metrics	The database table name where the monthly message statistical metrics should be stored.
Allowable values:	text
Default value:	DailyEventMetrics
register-message-board-handler	When true, messages posted to the message board are rerouted to become operator messages (gevm-notification-message) and inserted into the primary message queue labelled Messages.
Allowable values:	truth-value
Default value:	false
Notes:	See "Configuration File" on page 30.
message-board-messages-priority	The initial priority of message board messages rerouted as GEVM messages.
Allowable values:	Any positive integer

Attribute	Description
Default value:	9
Notes:	See "Configuration File" on page 30.
register-logbook-message-handler	When true, messages posted to the logbook are rerouted to become GEVM messages (gevm-notification-message) and inserted into the primary message queue labelled Messages.
Allowable values:	truth-value
Default value:	false
Notes:	See "Configuration File" on page 30.
logbook-messages-priority	The initial priority of logbook messages rerouted as GEVM messages.
Allowable values:	Any positive integer
Default value:	8
Notes:	See "Configuration File" on page 30.
enable-g2-error-handler	When true, G2 errors or errors signalled by a KB are converted to GEVM messages (gevm-g2-error-message) and inserted into the primary message queue labelled Messages. G2 errors are never detected as duplicates. This functionality is in addition to any logging configured and displayed in GERR.
Allowable values:	truth-value
Default value:	false
g2-error-message-priority	The initial priority of G2 errors messages converted into GEVM messages.
Allowable values:	Any positive integer

Attribute	Description
Default value:	9
jmail-interface-name	This attribute is not currently used.
Allowable values:	Any symbol
Default value:	none
Notes:	See "Configuration File" on page 30.

Configuration File

This table describes the settings in the configuration file (`config.txt`, by default), the associated group, and the attributes in the gevm-module-settings object that they configure at startup, where relevant.

Group	Configuration File Settings	GDSM Module Settings Attributes
GEVM	MESSAGE-REEVALUATION-ENABLED=false	N/A
GEVM	MESSAGE-REEVALUATION-PERIOD=3600	message-default-reevaluation-interval
GEVM	MAXIMUM-EVENT-HISTORY-STATES=10	maximum-event-history-states
GEVM	REVERSE-MESSAGE-COLOR-IF-ACKNOWLEDGED=true	reverse-message-color-if-acknowledged
GEVM	CDG-MESSAGES-USE-STANDARD-COLOR-MAPPING=true	cdg-messages-use-standard-color-mapping
GEVM	MESSAGE-COLOR-BASED-ON=priority	message-color-based-on
GEVM	REGISTER-LOGBOOK-MESSAGE-HANDLER=false	register-logbook-message-handler
GEVM	LOGBOOK-MESSAGES-PRIORITY=8	logbook-messages-priority
GEVM	REGISTER-MESSAGE-BOARD-HANDLER=false	register-message-board-handler
GEVM	MESSAGE-BOARD-MESSAGES-PRIORITY=9	message-board-messages-priority

Group	Configuration File Settings	GDSM Module Settings Attributes
GEVM	JMAIL-INTERFACE-NAME=none	jmail-interface-name
EVENTS	MAX-ENTRIES-IN-MEMORY=10000	Configures Maximum Entries in Queue of the Events queue.
EVENTS	UPDATE-LATENCY=1.0	Configures Update Latency of the Events queue.
EVENTS	ARCHIVING-ENABLED=false	Configures Logging Enabled of the Events queue.
EVENTS	ARCHIVING-LOG-TO-FILE-ENABLED=false	Configures Log to File of the Events queue.
EVENTS	ARCHIVING-LOG-TO-DATABASE-ENABLED=false	Configures Log to Database of the Events queue.
EVENTS	ARCHIVING-LOG-TO-JMS-ENABLED=false	Configures Log to JMS Provider of the Events queue.
EVENTS	ARCHIVING-LOG-CHANGES-ENABLED=true	Configures Log Changes of the Events queue.
EVENTS	ARCHIVING-LOG-ADDITIONS-ENABLED=true	Configures Log Additions of the Events queue.
EVENTS	ARCHIVING-LOG-REMOVAL-ENABLED=true	Configures Log Removals of the Events queue.
EVENTS	ARCHIVING-DIRECTORY=$APPLICATION-ROOT-DIRECTORY/logs	Configures Log Directory of the Events queue.
EVENTS	ARCHIVING-FILENAME-TEMPLATE=log_events_*.csv	Configures Log Filename Template of the Events queue.
EVENTS	ARCHIVING-INTERVAL-TO-OPEN-NEW-LOG-FILE=86400	Configures Interval to Open New Log File of the Events queue.
EVENTS	ARCHIVING-MAXIMUM-FILE-SIZE=500000	Configures Maximum Log Size of the Events queue.
EVENTS	ARCHIVING-DATABASE-INTERFACE=	Configures Database Interface of the Events queue.
EVENTS	ARCHIVING-DATABASE-TABLE=gevm_events	Configures Database Table of the Events queue.

Group	Configuration File Settings	GDSM Module Settings Attributes
EVENTS	ARCHIVING-LOG-JMS-INTERFACE=	Configures JMS Interface of the Events queue.
EVENTS	ARCHIVING-LOG-JMS-AS-XML=false	Configures Log JMS as XML of the Events queue.
ALARMS	MAX-ENTRIES-IN-MEMORY=10000	Configures Maximum Entries in Queue of the Alarms queue.
ALARMS	ARCHIVING-DIRECTORY=$APPLICATION-ROOT-DIRECTORY/logs	Configures Log Directory of the Alarms queue.
ALARMS	ARCHIVING-FILENAME-TEMPLATE=log-alarms-*.csv	Configures Log Filename Template of the Alarms queue.
ALARMS	ARCHIVING-INTERVAL-TO-OPEN-NEW-LOG-FILE=86400	Configures Interval to Open New Log File of the Alarms queue.
ALARMS	ARCHIVING-MAXIMUM-FILE-SIZE=100000	Configures Maximum Log Size of the Alarms queue.
ALARMS	ARCHIVING-ENABLED=false	Configures Logging Enabled of the Alarms queue.
ROOT CAUSES	MAX-ENTRIES-IN-MEMORY=10000	Configures Log Directory of the Root Causes queue.
ROOT CAUSES	ARCHIVING-DIRECTORY=$APPLICATION-ROOT-DIRECTORY/logs	Configures Log Filename Template of the Root Causes queue.
ROOT CAUSES	ARCHIVING-FILENAME-TEMPLATE=log-alarms-*.csv	Configures Interval to Open New Log File of the Root Causes queue.
ROOT CAUSES	ARCHIVING-INTERVAL-TO-OPEN-NEW-LOG-FILE=86400	Configures Maximum Log Size of the Root Causes queue.
ROOT CAUSES	ARCHIVING-MAXIMUM-FILE-SIZE=100000	Configures Logging Enabled of the Root Causes queue.
ROOT CAUSES	ARCHIVING-ENABLED=false	Configures Log Directory of the Root Causes queue.
TEST ACTIONS	MAX-ENTRIES-IN-MEMORY=10000	Configures Log Directory of the Test Actions queue.

Group	Configuration File Settings	GDSM Module Settings Attributes
TEST ACTIONS	ARCHIVING-DIRECTORY=$APPLICATION-ROOT-DIRECTORY/logs	Configures Log Filename Template of the Test Actions queue.
TEST ACTIONS	ARCHIVING-FILENAME-TEMPLATE=log-alarms-*.csv	Configures Interval to Open New Log File of the Test Actions queue.
TEST ACTIONS	ARCHIVING-INTERVAL-TO-OPEN-NEW-LOG-FILE=86400	Configures Maximum Log Size of the Test Actions queue.
TEST ACTIONS	ARCHIVING-MAXIMUM-FILE-SIZE=100000	Configures Logging Enabled of the Test Actions queue.
TEST ACTIONS	ARCHIVING-ENABLED=false	Configures Log Directory of the Test Actions queue.
REPAIR ACTIONS	MAX-ENTRIES-IN-MEMORY=10000	Configures Log Filename Template of the Repair Actions queue.
REPAIR ACTIONS	ARCHIVING-DIRECTORY=$APPLICATION-ROOT-DIRECTORY/logs	Configures Interval to Open New Log File of the Repair Actions queue.
REPAIR ACTIONS	ARCHIVING-FILENAME-TEMPLATE=log-alarms-*.csv	Configures Maximum Log Size of the Repair Actions queue.
REPAIR ACTIONS	ARCHIVING-INTERVAL-TO-OPEN-NEW-LOG-FILE=86400	Configures Logging Enabled of the Repair Actions queue.
REPAIR ACTIONS	ARCHIVING-MAXIMUM-FILE-SIZE=100000	Configures Log Directory of the Repair Actions queue.
REPAIR ACTIONS	ARCHIVING-ENABLED=false	Configures Log Filename Template of the Repair Actions queue.

G2 Error Handling as Operator Messages

You can choose to display signalled G2 errors in the GEVM Message Browser, rather than in the G2 Logbook or Message Board. Note that the actions such as error logging or posting to the message board enabled in GERR are not disabled or superseded by this functionality. You need to explicitly disable those actions in GERR. This feature is disabled by default in GEVM. If enabled, errors that are signalled by G2 or KB code are converted into gevm-g2-error-message instances, a subclass of gevm-notification message, and inserted into the main message queue labelled Messages. To support the processing of g2-error instances, you configure the following two attributes on gevm-module-settings:

- enable-g2-error-handler — If true, creates an operator message for each g2-error and adds it to the messages queue. The default value is false.
- g2-error-message-priority — The priority of G2 error messages.

Custom Message Colors

You can customize the lookup tables to use for determining message background and text color, based on message priority and message type. You can also configure the background color, based on acknowledgement status.

To customize message color, you must first configure the appropriate parameters in the configuration file. You then configure attributes of the gevm-module-settings object to refer to custom lookup tables and colors, based on the parameter settings.

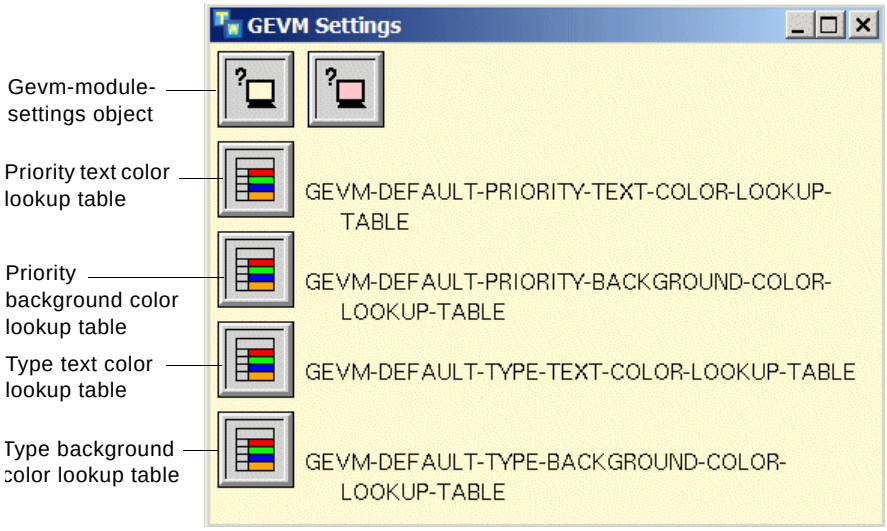
Caution Be sure to save the GEVM module settings object to a higher-level module.

To customize message color, based on priority or type:

1 Configure the message-color-based-on parameter in the configuration file to be priority or type.

2 Go to the GEVM Settings workspace to display the default GEVM settings object and the default color lookup tables for determining message color in priority and type mode.

Here is the GEVM Settings workspace:



3 Clone the appropriate type of lookup table and choose table of values on the lookup table to customize the values in the table.

4 Clone the gevm-module-settings object and place it in a higher-level module in your module hierarchy.

5 Configure the appropriate attributes of your custom gevm-module-settings object to refer to the custom lookup tables, as follows:

In this mode...	Configure this attribute of the GEVM module settings object...	To customize the lookup table for setting...
priority	message-priority-text-color-lookup-table	Message text color.
	message-priority-background-color-lookup-table	Message background color.
type	message-type-text-color-lookup-table	Message text color.
	message-type-background-color-lookup-table	Message background color.

6 Save the higher-level module, as well as the GEVM module.

The lookup tables for configuring message text color uses black for all priorities and types, by default, which you can configure.

7 Configure the default-color to be the default color to use when values cannot be found in the lookup table.

Here are the default lookup table for configuring the background color for messages, based on priority.

GEVM-DEFAULT-PRIORITY-BACKGROUND-COLOR-LOOKUP-TABLE

Table of values

add or delete rows

x	gevm-default-priority-background-color-lookup-table (x)
1	red
2	orange
3	yellow
4	thistle
5	salmon
6	green
7	wheat
8	sienna
9	tan
10	sky-blue

Here are the default lookup table for configuring the background color for messages, based on type:

GEVM-DEFAULT-TYPE-BACKGROUND-COLOR-LOOKUP-TABLE

Table of values

add or delete rows

x	gevm-default-type-background-color-lookup-table (x)
gevm-action-done	sienna
gevm-action-to-do	sienna
gevm-advisory	tan
gevm-alarm	tan
gevm-cdg-action-event	tan
gevm-cdg-alarm-event	tan
gevm-cdg-event	sienna
gevm-cdg-root-cause-event	yellow
gevm-command	tan
gevm-conclusion	tan
gevm-external-fault	salmon
gevm-inferred-fault	salmon
gevm-internal-fault	salmon
gevm-message	sky-blue
gevm-notification-message	sky-blue
gevm-root-cause	yellow

To avoid contrast problems, the default background color of acknowledged messages is smoke; however, you can customize the background color of acknowledged messages, as needed. Be sure to choose a background color that allows all possible text colors to be legible.

To customize message text color, based on acknowledgement status:

1. Configure the reverse-message-color-if-acknowledged parameter to be true.

   This causes the message background color to become the message text color when the message has been acknowledged.

2. In your custom GEVM module settings object stored in a higher-module, configure the background-color-when-acknowledged attribute to be a color other than smoke.

Custom Message Priority Escalation

You can customize the way in which your application escalates message priority. To do this, you must implement your own custom priority escalation scheme, then override the message-default-escalation-scheme attribute of the gevm-module-settings object to refer to your custom scheme. You must also implement your own grtl-evaluate method for your custom scheme. For details, see the *G2 Run-Time Library User's Guide*.

Caution Be sure to save the GEVM module settings object to a higher-level module.

A priority escalation scheme is a type of grtl-scheme. It defines a scheme-key that the module settings object uses to locate the scheme. The grtl-evaluate method determines how messages escalate. A typical implementation raises the priority of the message, though your custom implementation might implement more complex logic and actions.

The default priority escalation scheme is Default Event Escalation Scheme, which is an instance of a gevm-priority-escalation-scheme. The default grtl-evaluate method for this scheme increments the priority, based on the gevm-change-priority API procedure.

You access the default escalation scheme on the Programmer's Interface workspace of the gevm-top-level workspace.

To customize priority escalation:

1. On the GEVM Programmer's Interface workspace, clone the gevm-priority-escalation-scheme class to create your own priority escalation scheme class definition.

   This class must inherit from grtl-object and grtl-scheme. For details, see the *G2 Run-Time Library User's Guide*.

2 Create an instance of your custom priority escalation class and configure the scheme-key.

Note You must be in Developer or Administrator mode to edit the table of the instance.

3 Clone the gevm-priority-escalation-scheme::grtl-evaluate method and provide your own implementation of this method for your custom priority escalation scheme class.

4 On the GEVM Settings workspace, clone the gevm-module-settings object and place it in a higher-level module in your module hierarchy.

5 Edit the message-default-escalation-scheme of your custom GEVM settings object to refer to the scheme key of your custom priority escalation scheme.

6 Enable the reevaluation by doing one of the following:

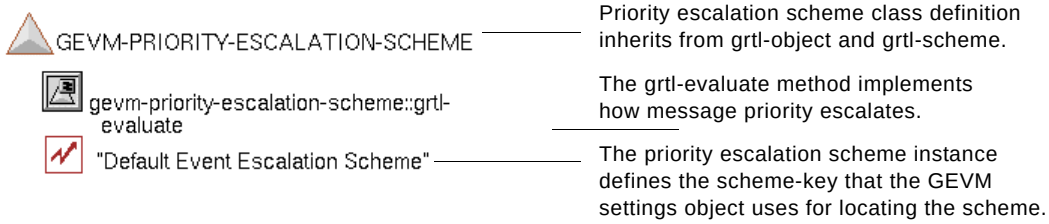
➔ Configure the MESSAGE-REEVALUATION-ENABLED and MESSAGE-REEVALUATION-PERIOD options in the application configuration file (`config.txt` file by default).

or

➔ Call gevm-enable-event-escalation to enable it.

If you change the settings in the application configuration file, they will only be taken into account after the next G2 restart. Note that enabling priority escalation also enables checking the lifetime of messages (gevm-lifetime) and deleting messages after their lifetime has expired.

Here is the default priority escalation scheme class definition, and its associated instance and grtl-evaluate method:



Custom Message Correlation

You can customize the procedure or method to call to perform correlation between messages. For example, the logic might correlate alarm messages and conclusion messages, and conclusion messages might be subsumed by a root cause operator message.

For examples and a detailed description of message correlation, see “Customizing Message Correlation” on page 264.

3

User Preferences

Describes the GEVM user preferences.

Introduction **41**

gevm-user-preferences **43**

gevm-user-filter **55**

Methods **62**

gevm-user-preferences::gevm-email-notification **63**

Introduction

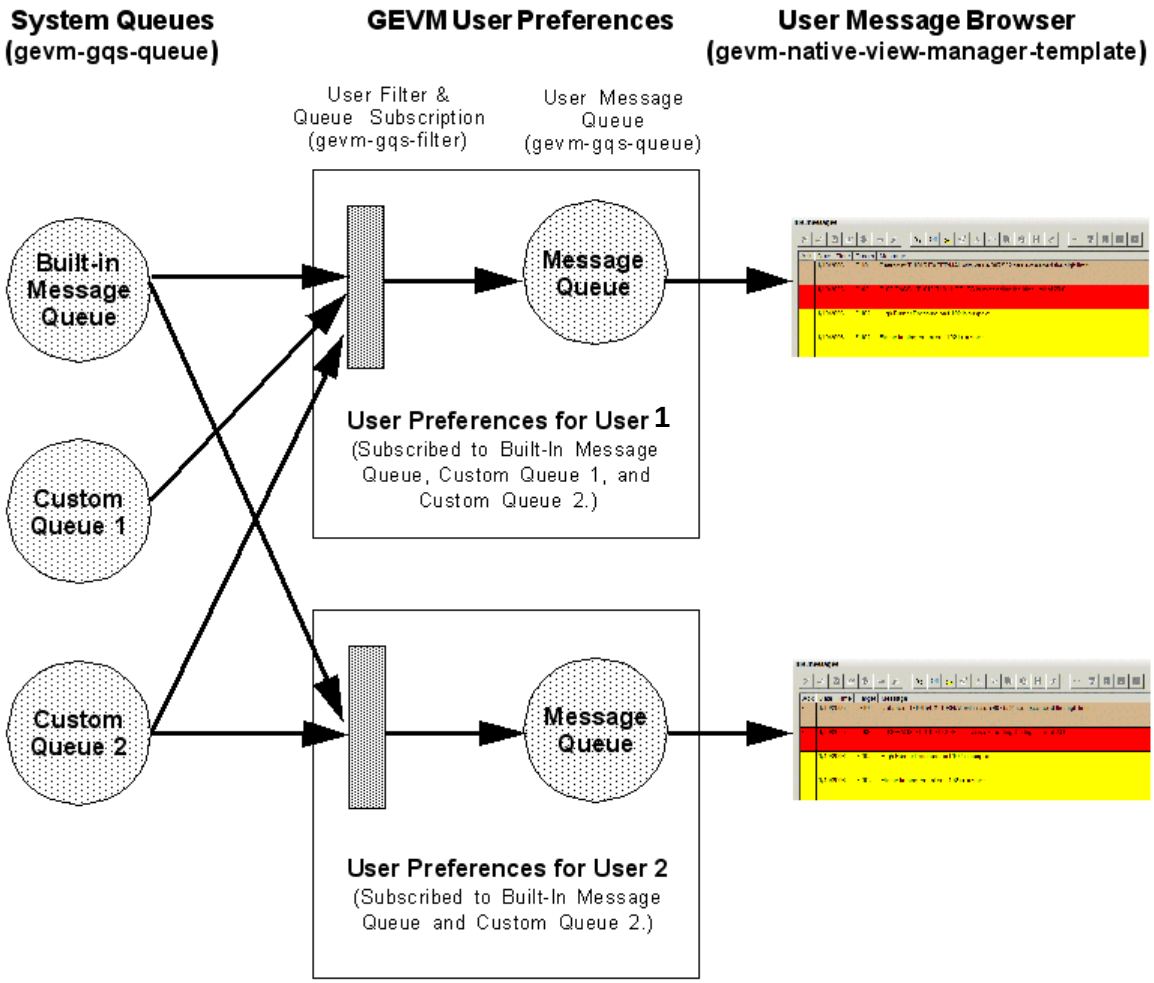
User preferences specify user-specific preferences for the application. The base class, grtl-user-preferences, is described in the *G2 Run-Time Library User's Guide*. Modules might subclass this class and specify it in the active grtl-module-settings instance. This enables you to specify user preference extensions for different modules.

GEVM extends grtl-user-preferences to specify the message queue to use for a particular user, as well as actions to send notifications such as email messages when messages in the user's queue are added or changed. The user's message queue typically registers with one or more system gevm-gqs-queue instances to collect the messages that are relevant for that users.

To provide an example, you might have one system queue for equipment and sensor alarms, and another system queue for system and network-related alarms. A manager's queue might contain high-priority messages from either of these two system queues, while one group of operators might monitor the equipment and

sensor alarms, and another group of operators might monitor alarms related to system and network availability.

The following figure shows how these and other GEVM objects interact:



gevm-user-preferences

Extends the user preferences defined in GRTL by including a queue of messages specific to the user. This queue subscribes to queues defined by the System-Administrator to provide a dashboard of relevant messages for each user.

Class Inheritance Path

gevm-user-preferences, grtl-user-preferences, grtl-object-with-key, object, grtl-item-with-key, grtl-item, item

Attributes

Attribute	Description
user-name	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	Any symbol
Default value:	ANONYMOUS
email-address	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	Any text
Default value:	""
mobile-email-address	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	Any text
Default value:	""
default-user-mode	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	Any symbol
Default value:	OPERATOR

Attribute	Description
home-location	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	Any text
Default value:	"default"
location-history	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	Any sequence
Default value:	sequence ()
location-history-index	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	Any integer
Default value:	-1
location-history-size	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	Any integer
Default value:	20
indicate-items-upon-menu-selection	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	Any truth-value
Default value:	false
disconnect-permission	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.

Attribute	Description
Allowable values:	Any truth-value
Default value:	false
shutdown-permission	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	Any truth-value
Default value:	true
configuration-permission	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	Any truth-value
Default value:	true
acknowledge-messages-permission	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	Any truth-value
Default value:	true
delete-messages-permission	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	Any truth-value
Default value:	true
use-g2-logbook	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	Any truth-value
Default value:	false

Attribute	Description
beep-enabled	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	Any truth-value
Default value:	true
show-extended-menus	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	Any truth-value
Default value:	true
show-native-user-interface-preference	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	Any truth-value
Default value:	false
show-message-browser-by-default	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	Any truth-value
Default value:	true
acknowledge-messages-upon-selection	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	Any truth-value
Default value:	false
ui-theme	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.

Attribute	Description
Allowable values:	windows-theme-2000, windows-theme-2001, windows-theme-2002, or windows-theme-2003
Default value:	windows-theme-2003
release-user-interface-procedure	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	Any symbol
Default value:	GRTL-RELEASE-USER
show-custom-user-interface-procedure	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	Any symbol
Default value:	UNSPECIFIED
show-operator-user-interface-procedure	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	Any symbol
Default value:	UNSPECIFIED
show-modeler-user-interface-procedure	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	Any symbol
Default value:	UNSPECIFIED
show-developer-user-interface-procedure	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	Any symbol
Default value:	UNSPECIFIED

Attribute	Description
show-system-administrator-user-interface-procedure	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	Any symbol
Default value:	UNSPECIFIED
resize-custom-user-interface-procedure	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	Any symbol
Default value:	UNSPECIFIED
resize-operator-user-interface-procedure	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	Any symbol
Default value:	UNSPECIFIED
resize-modeler-user-interface-procedure	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	Any symbol
Default value:	UNSPECIFIED
resize-developer-user-interface-procedure	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	Any symbol
Default value:	UNSPECIFIED

Attribute	Description
resize-system-administrator-user-interface-procedure	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	Any symbol
Default value:	UNSPECIFIED
refresh-user-interface-procedure	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	Any symbol
Default value:	UNSPECIFIED
view-item-procedure	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	Any symbol
Default value:	UNSPECIFIED
view-metrics-procedure	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	Any symbol
Default value:	UNSPECIFIED
view-process-map-procedure	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	Any symbol
Default value:	UNSPECIFIED
view-equipment-procedure	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.

Attribute	Description
Allowable values:	Any symbol
Default value:	UNSPECIFIED
view-event-detection-procedure	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	Any symbol
Default value:	UNSPECIFIED
view-fault-models-procedure	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	Any symbol
Default value:	UNSPECIFIED
user-attribute-1	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	Any item or value
Default value:	NONE
user-attribute-2	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	Any item or value
Default value:	NONE
key	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	inherited
Default value:	""

Attribute	Description
menubar-name	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	symbol
Default value:	default-menubar
menu-command-execute-procedure	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	symbol
Default value:	none
relation-display-mode	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	symbol
Default value:	none
zooming-factor	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	quantity
Default value:	0
tabbed-mdi-mode	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.
Allowable values:	truth-value
Default value:	false
restore-last-child-pane-settings	See grtl-user-preferences in the *G2 Run-Time Library User's Guide*.

Attribute	Description
Allowable values:	truth-value
Default value:	true
messages	The message queue associated with this user. This queue subscribes to other queues, and the content of those queues is propagated to the user queue based on those subscriptions and the message filter described below.
Allowable values:	gevm-gqs-queue instance
Default value:	N/A
message-filter	The filter applied to messages transferred from subscribed queues to the user queue. See messages above.
Allowable values:	gevm-user-filter instance
Default value:	N/A
message-detail-visible-attributes	The list of fields to display in the properties dialog of a message when displayed by this user.
Allowable values:	sequence of text values
Default value:	sequence ("acknowledged", "acknowledged-by-user", "acknowledged-timestamp", "lifetime", "assigned-to-user", "type", "message", "category", "priority", "repetitions", "detail", "creation-timestamp", "last-update-timestamp", "advice", "comments", "acknowledgement-required");

Attribute	Description
email-notification	The email notification action to perform when messages are added to the user's message queue or properties of messages change. Emails are sent to the address specified in the email-address attribute.
Allowable values:	One of the following symbols: never, send-as-short-text, send-as-text, send-as-html, if-not-connected-send-short-text, if-not-connected-send-as-text, if-not-connected-send-as-html, send-high-priority-as-short-text, send-high-priority-as-text, send-high-priority-as-html
Default value:	never
mobile-email-notification	The email notification action to perform when messages are added to the user's message queue or properties of messages change. Emails are sent to the address specified in the mobile-email-address attribute.
Allowable values:	One of the following symbols: never, send-as-short-text, send-as-text, send-as-html, if-not-connected-send-short-text, if-not-connected-send-as-text, if-not-connected-send-as-html, send-high-priority-as-short-text, send-high-priority-as-text, send-high-priority-as-html
Default value:	never
modeler-message-browser-template	The name of an access table (gqs-queue-access-table) or message browser template (gevm-native-view-manager-template) to use when displaying messages in administrator, system administrator, developer or modeler mode for the user associated with this user preference.

Attribute	Description
Allowable values:	symbol
Default value:	gevm-modeler-msg-view-template
operator-message-browser-template	The name of an access table (gqs-queue-access-table) or message browser template (gevm-native-view-manager-template) to use when displaying messages in operator mode for the user associated with this user preference.
Allowable values:	symbol
Default value:	gevm-operator-msg-view-template
destination-queue	The destination queue when using the action button in the message browser to send a message to another gevm-gqs-queue. A dialog is displayed enabling the user to change the destination queue and the selection is stored in this attribute.
Allowable values:	text
Default value:	Messages
enable-status-bar-message-browser	Whether to enable the status bar in the Message Browser.
Allowable values:	truth-value
Default value:	true
telnet-shell-command	A Telnet shell command.
Allowable values:	text
Default value:	"@"C:\Program Files\PuTTY\putty.exe@""

gevm-user-filter

Extends the filter defined in GQS to provide filtering capabilities for messages sent from system queues to the user queue specified in the user preference.

Class Inheritance Path

gevm-user-filter, gqs-filter, gfr-object-with-uuid, object, gfr-item-with-uuid, item

Attributes

Attribute	Description
select-priority-1	When true, propagates messages with a priority of 1 to the user's queue.
Allowable values:	truth-value
Default value:	true
select-priority-2	When true, propagates messages with a priority of 2 to the user's queue.
Allowable values:	truth-value
Default value:	true
select-priority-3	When true, propagates messages with a priority of 3 to the user's queue.
Allowable values:	truth-value
Default value:	true
select-priority-4	When true, propagates messages with a priority of 4 to the user's queue.
Allowable values:	truth-value
Default value:	true

Attribute	Description
select-priority-5	When true, propagates messages with a priority of 5 to the user's queue.
Allowable values:	truth-value
Default value:	true
select-priority-6	When true, propagates messages with a priority of 6 to the user's queue.
Allowable values:	truth-value
Default value:	true
select-priority-7	When true, propagates messages with a priority of 7 to the user's queue.
Allowable values:	truth-value
Default value:	true
select-priority-8	When true, propagates messages with a priority of 8 to the user's queue.
Allowable values:	truth-value
Default value:	true
select-priority-9	When true, propagates messages with a priority of 9 to the user's queue.
Allowable values:	truth-value
Default value:	true

Attribute	Description
select-all-priority-messages	When true, propagates all messages with any priority to the user's queue.
Allowable values:	truth-value
Default value:	true
enable-filtering-for-type	When true, enables filtering based on the message class.
Allowable values:	truth-value
Default value:	false
selected-type	The name of a gevm-message class. Messages of this class and subclasses will be propagated to the user's queue.
Allowable values:	text
Default value:	""
enable-filtering-for-target	When true, enables filtering of messages based on a specific target.
Allowable values:	truth-value
Default value:	false
selected-target	The key of the selected target. Messages associated with this target will be propagated to the user's queue.
Allowable values:	A symbol specifying a subclass of gevm-message
Default value:	gevm-message

Attribute	Description
enable-filtering-for-process-map	When true, enables filtering based on a specific domain map.
Allowable values:	truth-value
Default value:	false
selected-process-map	The key of the selected process map. Messages with a target contained in this process map will be propagated to the user's queue.
Allowable values:	text
Default value:	""
enable-filtering-for-category	When true, enables filtering based on the message Category field.
Allowable values:	truth-value
Default value:	false
selected-category	The category of messages to selected. Messages with matching category will be propagated to the user's queue.
Allowable values:	text
Default value:	
enable-filtering-assigned-to-user	When true, enables filtering based on the Assigned to User field.
Allowable values:	truth-value
Default value:	false

Attribute	Description
selected-user	A user name. Messages with a matching user name in its gevm-assigned-to-user attribute will be propagated to the user's queue.
Allowable values:	text
Default value:	""
enable-filtering-by-update-time	When true, enables filtering based on the Update Time field.
Allowable values:	truth-value
Default value:	false
selected-message-aging	An update period. Any message that has not been updated with this update period will be propagated to the user's queue at the time of evaluation.
Allowable values:	Any positive integer
Default value:	0
select-unacknowledged-messages-only	When true, only propagates unacknowledged messages to the user's queue.
Allowable values:	truth-value
Default value:	false
hide-subsumed-messages	When true, only propagates messages that are not subsumed to the user's queue.
Allowable values:	truth-value
Default value:	false

Attribute	Description
message-queue-subscription	A sequence of text, where each text is the key label of a message queue to which subscribe.
Allowable values:	sequence of text values
Default value:	sequence ("Messages")
custom-message-filter-procedure	The name of a G2 procedure to implement custom filters. The procedure might remove items contained in the second argument based on its logic. The signature of a custom filter procedure is: my-filter (*filter*: class gevm-filter, *result-list*: class item-list, *client*: class object)
Allowable values:	Symbol naming a valid G2 procedure
Default value:	unspecified
enable-filtering-assigned-to-group	When true, enables filtering based on the Assigned to Group field.
Allowable values:	truth-value
Default value:	false
selected-group	The group to which the selected message belongs.
Allowable values:	text
Default value:	""
enable-filtering-for-target-class	When true, enables filtering based on the target class of the selected message.
Allowable values:	truth-value
Default value:	false

Attribute	Description
selected-target-class	The target class of the selected message.
Allowable values:	symbol
Default value:	grtl-domain-object
exclude-messages-for-inactive-targets	Whether to exclude messages when the target is inactive.
Allowable values:	truth-value
Default value:	false

Methods

gevm-user-preferences::gevm-email-notification on page 63

gevm-user-preferences::gevm-email-notification

Synopsis

gevm-email-notification
(*user*: class gevm-user-preferences, *action*: symbol, *mode*: symbol, *email-to*: text, *event*: class gevm-message)

Argument	Description
user	The gevm-user-preferences object.
action	A symbol to insert in the message, for example, "The following message has been *action*."
mode	How to send the email. Options are: never, send-as-short-text, send-as-text, send-as-html, if-not-connected-send-short-text, if-not-connected-send-as-text, if-not-connected-send-as-html, only-high-priority-as-short-text, only-high-priority-as-text, only-high-priority-as-html
email-to	The email address of the person to which to send the email message.
event	The gevm-message to send as email.

Description

Formats a gevm-message as email and sends it using the specified user preference settings.

4

Events

Describes GEVM event configuration and notification; class definition; operations for event creation; the basic event types; operations for event management, target events, initiating events, and event escalation; and customization.

Introduction **65**

Event Configuration **67**

Event Notification **68**

Event Class Definition **70**

Event Creation Operations **73**

Event Classes **90**

Event Management Operations **168**

Target Event Operations **207**

Initiating Event Operations **231**

Escalation Operations **255**

Customization **263**

Introduction

GEVM provides a rich environment to record events that occur in a monitored system, to query for those events, to correlate events, and to display those events. GEVM uses an object-oriented approach to define event types and defines two broad types of events and associated views:

- A blackboard of raw events of type gevm-event-state or any subclass, which are typically used internally to record the presence or absence of the occurrence of an event. Correlation engines or rules reason on the presence or absence of those events for advanced analysis. Event-detection logic, such as one transforming sensor analog information into state information, for example, temperature high, sensor drifting, would create an event when it determines the condition is true and would delete the event when it determines the condition is false. Correlation engines or rules can then analyze patterns, based on the presence, absence, or frequency of those states to determine root causes, for example.
- Queues of events of type gevm-message or any subclass, which are typically called messages and are typically displayed to operators. Messages are collected into queues, dispatched to each user's message queue, and displayed in message views, typically called message browsers.

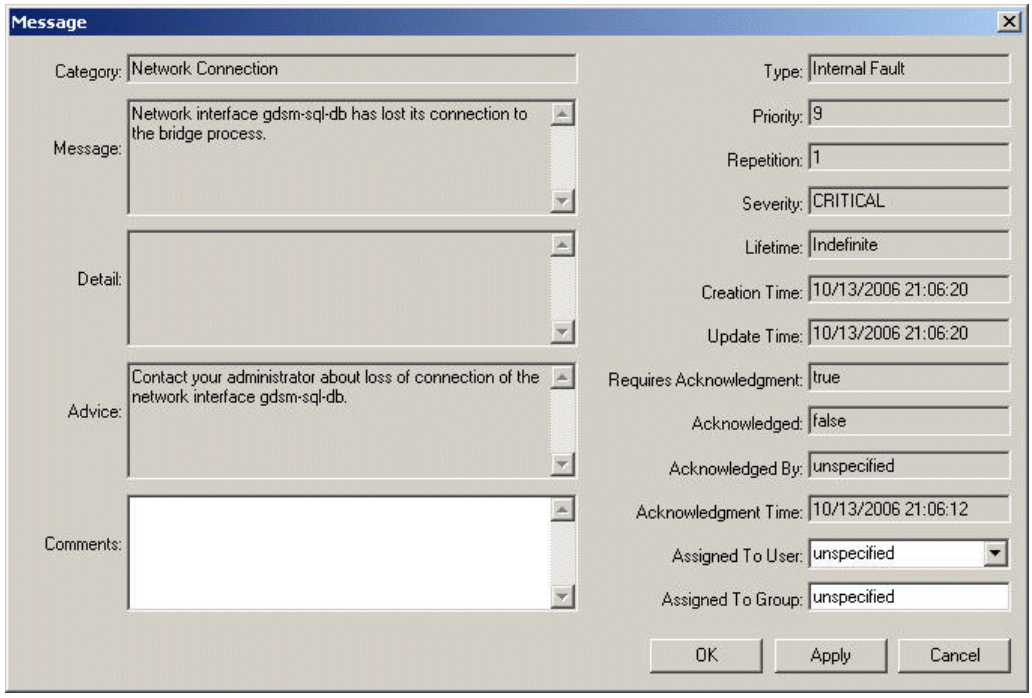
This chapter describes the event and message classes, and methods for creating them, inserting them into queues, and performing various operations. This chapter also describes common event and message customizations.

Event Configuration

Both state events and operator message events are typically configured upon creation, and are potentially updated when the event occurs again. If the event occurs multiple times, its repetition count is incremented and the content of the message might be updated to reflect the latest information.

The only available user level configuration is for instances of gevm-message or any subclass to add comments to a message and to assign the message to a specific user. The other fields are read-only.

Here is the properties dialog for configuring a message:



Event Notification

The table below summarizes the event notification types related to GEVM events. For more information on event notification, see the *G2 Run-Time Library User's Guide*.

Event Type	Event Arguments (Examples)	Applicable to Instances of Class	Description
GEVM			
gevm-acknowledged-event	structure (event-type: the symbol gevm-acknowledged-event, user-name: "user-name")	gevm-event	Sent when an event has been acknowledged.
gevm-deleting-event	structure (event-type: the symbol gevm-deleting-event)	gevm-event	Sent when an event has been deleted.
gevm-subordinating-event	structure (event-type: the symbol gevm-subordinating-event, parent-event: parent-event, parent-event-key: "key-of-parent-event")	gevm-event	Sent when an event has been subordinated to another event.
gevm-publish-event	structure (event-type: the symbol gevm-publish-event)	gevm-event	Sent when an event is published to a message queue.
gevm-event-priority-increase	structure (event-type: the symbol gevm-event-priority-increase, priority: 5)	gevm-event	Sent when the priority of an event increases.

gevm-event-priority-decrease	structure (event-type: the symbol gevm-event-priority-decrease, priority: 5)	gevm-event	Sent when the priority of an event decreases.
gevm-event-repetition	structure (event-type: the symbol gevm-event-repetition, repetition: 5)	gevm-event	Sent when the repetition count of an event changes.
gevm-event-notification	structure (event-type: the symbol gevm-event-notification, gevm-event-type: the event-type-of-the-event, gevm-event: event and other attributes of original event notification)	grtl-domain-object	Sent when a domain object receives an event notification. gevm-event-type is one of: • gevm-acknowledged-event • gevm-deleting-event • gevm-subordinating-event • gevm-publish-event • gevm-event-priority-increase • gevm-event-priority-decrease • gevm-event-repetition

Event Class Definition

gevm-event-class on page 71

gevm-event-class

A class definition that all GEVM event classes use for defining event classes. To create your own event classes, you must use this class definition instead of a normal class definition.

Class Inheritance Path

gevm-event-class, grtl-event-source-class-definition, grtl-class-definition, class-definition, definition, item

Attributes

Attribute	Description
event-metrics-enabled	Determines whether event metrics are calculated for the event. You can set this value by using the Enable Event Metrics and Disable Event Metrics menu choices.
Allowable values:	Any truth-value
Default value:	false
class-description	See grtl-class-definition in the *G2 Run-Time Library User's Guide*.
Allowable values:	inherited
Default value:	""
instance-key-attribute-name	See grtl-class-definition in the *G2 Run-Time Library User's Guide*.
Allowable values:	inherited
Default value:	gevm-key
class-attribute-properties	See grtl-class-definition in the *G2 Run-Time Library User's Guide*.
Allowable values:	inherited
Default value:	sequence ()

Methods

gevm-event-class::gevm-create-event on page 74
gevm-event-class::gevm-create-event on page 76
gevm-event-class::gevm-create-event on page 78
gevm-event-class::gevm-create-event on page 80
gevm-event-class::gevm-create-event-and-insert-into-queue on page 82
gevm-event-class::gevm-create-event-and-insert-into-queue on page 84
gevm-event-class::gevm-create-event-and-insert-into-queue on page 86
gevm-event-class::gevm-create-event-and-insert-into-queue on page 88

Event Creation Operations

gevm-event-class::gevm-create-event on page 74
gevm-event-class::gevm-create-event on page 76
gevm-event-class::gevm-create-event on page 78
gevm-event-class::gevm-create-event on page 80
gevm-event-class::gevm-create-event-and-insert-into-queue on page 82
gevm-event-class::gevm-create-event-and-insert-into-queue on page 84
gevm-event-class::gevm-create-event-and-insert-into-queue on page 86
gevm-event-class::gevm-create-event-and-insert-into-queue on page 88

gevm-event-class::gevm-create-event

Synopsis

gevm-event-class::gevm-create-event
(*event-class*: gevm-event-class, *category*: text, *priority*: integer,
message: text, *detail*: text, *advice*: text, *acknowledgment-required*: truth-value,
comment: text, *initiating-item*: item, *target-item*: item, *lifetime*: quantity,
update-duplicate-event: truth-value, *client*: ui-client-item)
–> *event*: gevm-event

Argument	Description
event-class	The class of the event.
category	The event category.
priority	A priority where low numbers are the most urgent.
message	The event message.
detail	The event detail.
advice	The event advice.
acknowledgment-required	True if the message needs to be acknowledged.
comment	The default comment.
initiating-item	The item sending the event, which must be a type of gevm-event-initiating-item.
target-item	The target item for the event.
lifetime	The maximum time for the event to exist.
update-duplicate-event	If true, updates the message, detail, advice, life time, and acknowledgment required fields of duplicate events.
client	The client of this event invocation, such as a G2 window.

Return Value	Description
event	The new event.

Description

This method creates an event with message text, details, advice, acknowledgement requirement, and comment. It has no provision for text translation or text substitution.

gevm-event-class::gevm-create-event

Synopsis

gevm-event-class::gevm-create-event
(*event-class*: gevm-event-class, *category*: text, *priority*: integer,
message-key: symbol, *message-args*: sequence, *detail-key*: symbol,
detail-args: sequence, *initiating-item*: item-or-value,
target-item: item-or-value,*lifetime*: quantity,
update-duplicate-event: truth-value, *client*: ui-client-item)
–> *event*: gevm-event

Argument	Description
event-class	The class of the event.
category	The event category.
priority	A priority where low numbers are the most urgent.
message-key	A lookup key for the event message.
message-args	A sequence of simple values that will be inserted into the message text.
detail-key	A lookup key for the event detail.
detail-args	A sequence of simple values that will be inserted into the message text.
initiating-item	The item sending the event, which must be a type of gevm-event-initiating-item.
target-item	The target item for the event.
lifetime	The maximum time for the event to exist.
update-duplicate-event	If true updates the message, detail, advice, life time, and acknowledgment required fields of duplicate events.
client	The client of this event invocation, such as a G2 window.

Return Value	Description
event	The new event.

Description

This method creates an event that supports message and detail text translation and text substitution. It looks up the text of the message in the GFR resource file specified in GRTL and uses the GRTL APIs to perform the translation. It has no provision for advice, acknowledgement requirement, and comment.

gevm-event-class::gevm-create-event

Synopsis

gevm-event-class::gevm-create-event
(*event-class*: gevm-event-class, *category*: text, *priority*: integer, *message*: text, *detail*: text, *initiating-item*: item, *target-item*: item, *lifetime*: quantity, *update-duplicate-event*: truth-value, *client*: ui-client-item)
–> *event*: gevm-event

Argument	Description
event-class	The class of the event.
category	The event category.
priority	A priority where low numbers are the most urgent.
message	The event message.
detail	The event detail.
initiating-item	The item sending the event, which must be a type of gevm-event-initiating-item.
target-item	The target item for the event.
lifetime	The maximum time for the event to exist.
update-duplicate-event	If true updates the message, detail, advice, life time, and acknowledgment required fields of duplicate events.
client	The client of this event invocation, such as a G2 window.

Return Value	Description
event	The new event.

Description

This method creates an event with no provision for advice, acknowledgement requirement, or comment, and no provision for text translation and text substitution.

gevm-event-class::gevm-create-event

Synopsis

gevm-event-class::gevm-create-event
(*event-class*: gevm-event-class, *category*: text, *priority*: integer,
message-key: symbol, *message-args*: sequence, *detail-key*: symbol,
detail-args: sequence, *advice-key*: symbol, *advice-args*: sequence,
acknowledgment-required: truth-value, *comment*: text,
initiating-item: item-or-value, *target-item*: item-or-value, *lifetime*: quantity,
update-duplicate-event: truth-value, *client*: ui-client-item)
–> *event*: gevm-event

Argument	Description
event-class	The class of the event.
category	The event category.
priority	A priority where low numbers are the most urgent.
message-key	A lookup key for the event message.
message-args	A sequence of simple values that will be inserted into the message text.
detail-key	A lookup key for the event detail.
detail-args	A sequence of simple values that will be inserted into the message text.
advice-key	A lookup key for the event advice.
advice-args	A sequence of simple values that will be inserted into the advice text.
acknowledgment-required	True if message needs to be acknowledged.
comment	Default comment.
initiating-item	The item sending the event, which must be a type of gevm-event-initiating-item.
target-item	The target item for the event.
lifetime	The maximum time for the event to exist.

update-duplicate-event	If true updates the message, detail, advice, life time, and acknowledgment required fields of duplicate events.
client	The client of this event invocation, such as a G2 window.

Return Value	Description
event	The new event.

Description

This method creates an event that supports message, detail, and advice text translation and text substitution, as well as acknowledgement requirement and comment.

gevm-event-class::gevm-create-event-and-insert-into-queue

Synopsis

gevm-event-class::gevm-create-event-and-insert-into-queue
(*event-class*: gevm-event-class, *category*: text, *priority*: integer, *message*: text, *detail*: text, *advice*: text, *acknowledgment-required*: truth-value, *comment*: text, *initiating-item*: item, *target-item*: item, *lifetime*: quantity, *update-duplicate-event*: truth-value, *queue*: gevm-gqs-queue, *client*: ui-client-item)
–> *event*: gevm-event

Argument	Description
event-class	The class of the event.
category	The event category.
priority	A priority where low numbers are the most urgent.
message	The event message.
detail	The event detail.
advice	The event advice.
acknowledgment-required	True if message needs to be acknowledged.
comment	Default comment.
initiating-item	The item sending the event, which must be a type of gevm-event-initiating-item.
target-item	The target item for the event.
lifetime	The maximum time for the event to exist.
update-duplicate-event	If true updates the message, detail, advice, life time, and acknowledgment required fields of duplicate events.
queue	The destination queue of the event.
client	The client of this event invocation, such as a G2 window.

Return Value	Description
event	The new event.

Description

This method creates an event and inserts it into a queue. The event supports message text, details, advice, acknowledgement requirement, and comment. It has no provision for text translation or text substitution.

This method updates the priority even if the event already exists. If you do not want the priority of an existing message to be updated by this API, provide a negative value.

gevm-event-class::gevm-create-event-and-insert-into-queue

Synopsis

gevm-event-class::gevm-create-event-and-insert-into-queue
(*event-class*: gevm-event-class, *category*: text, *priority*: integer, *message*: text, *detail*: text, *initiating-item*: item, *target-item*: item, *lifetime*: quantity, *update-duplicate-event*: truth-value, *queue*: gevm-gqs-queue, *client*: ui-client-item)
–> *event*: gevm-event

Argument	Description
event-class	The class of the event.
category	The event category.
priority	A priority where low numbers are the most urgent.
message	The event message.
detail	The event detail.
initiating-item	The item sending the event, which must be a type of gevm-event-initiating-item.
target-item	The target item for the event.
lifetime	The maximum time for the event to exist.
update-duplicate-event	If true updates the message, detail, advice, life time, and acknowledgment required fields of duplicate events.
queue	The destination queue of the event.
client	The client of this event invocation, such as a G2 window.

Return Value	Description
event	The new event.

Description

This method creates an event and inserts it into a queue. It has no provision for advice, acknowledgement requirement, or comment, and no provision for text translation and text substitution.

This method updates the priority even if the event already exists. If you do not want the priority of an existing message to be updated by this API, provide a negative value.

gevm-event-class::gevm-create-event-and-insert-into-queue

Synopsis

gevm-event-class::gevm-create-event-and-insert-into-queue
(*event-class*: gevm-event-class, *category*: text, *priority*: integer,
message-key: symbol, *message-args*: sequence, *detail-key*: symbol,
detail-args: sequence, *initiating-item*: item-or-value, *target-item*: item-or-value,
lifetime: quantity, *update-duplicate-event*: truth-value, *queue*: gevm-gqs-queue,
client: ui-client-item)
–> *event*: gevm-event

Argument	Description
event-class	The class of the event.
category	The event category.
priority	A priority where low numbers are the most urgent.
message-key	A lookup key for the event message.
message-args	A sequence of simple values that will be inserted into the message text.
detail-key	A lookup key for the event detail.
detail-args	A sequence of simple values that will be inserted into the message text.
initiating-item	The item sending the event, which must be a type of gevm-event-initiating-item.
target-item	The target item for the event.
lifetime	The maximum time for the event to exist.
update-duplicate-event	If true updates the message, detail, advice, life time, and acknowledgment required fields of duplicate events.

queue	The destination queue of the event.
client	The client of this event invocation, such as a G2 window.

Return Value	Description
gevm-event	The new event.

Description

This method creates an event and inserts it into a queue. The event supports message and detail text translation and text substitution. It has no provision for advice, acknowledgement requirement, and comment.

This method updates the priority even if the event already exists. If you do not want the priority of an existing message to be updated by this API, provide a negative value.

gevm-event-class::gevm-create-event-and-insert-into-queue

Synopsis

gevm-event-class::gevm-create-event-and-insert-into-queue
(*event-class*: gevm-event-class, *category*: text, *priority*: integer,
message-key: symbol, *message-args*: sequence, *detail-key*: symbol,
detail-args: sequence, *advice-key*: symbol, *advice-args*: sequence,
acknowledgment-required: truth-value, *comment*: text,
initiating-item: item-or-value, *target-item*: item-or-value, *lifetime*: quantity,
update-duplicate-event: truth-value, *queue*: gevm-gqs-queue,
client: ui-client-item)
–> *event*: gevm-event

Argument	Description
event-class	The class of the event.
category	The event category.
priority	A priority where low numbers are the most urgent.
message-key	A lookup key for the event message.
message-args	A sequence of simple values that will be inserted into the message text.
detail-key	A lookup key for the event detail.
detail-args	A sequence of simple values that will be inserted into the message text.
advice-key	A lookup key for the event advice.
advice-args	A sequence of simple values that will be inserted into the advice text.
acknowledgment-required	True if message needs to be acknowledged.
comment	Default comment.
initiating-item	The item sending the event, which must be a type of gevm-event-initiating-item.
target-item	The target item for the event.

lifetime	The maximum time for the event to exist.
update-duplicate-event	If true updates the message, detail, advice, life time, and acknowledgment required fields of duplicate events.
queue	The destination queue of the event.
client	The client of this event invocation, such as a G2 window.

Return Value	Description
event	The new event.

Description

This method creates an event and inserts it into a queue. The event supports message, detail, and advice text translation and text substitution, as well as acknowledgement requirement and comment.

This method updates the priority even if the event already exists. If you do not want the priority of an existing message to be updated by this API, provide a negative value.

Event Classes

This chapter describes the superior class of all event types, gevm-event, and its subclasses. The two basic types of events are:

- gevm-message subclasses, which are considered operator messages and can appear in a message browser.
- gevm-event-state subclasses, which are considered raw events and which you can manage through event queues.

The event classes are:

gevm-action on page 91
gevm-action-done on page 94
gevm-action-to-do on page 97
gevm-advisory on page 100
gevm-advisory-message on page 103
gevm-alarm on page 106
gevm-calculated-alarm on page 109
gevm-change-in-process-state-alarm on page 112
gevm-command on page 115
gevm-conclusion on page 118
gevm-deviation-alarm on page 121
gevm-discrete-signal-alarm on page 124
gevm-event on page 127
gevm-event-state on page 130
gevm-external-fault on page 132
gevm-fault on page 135
gevm-g2-error-message on page 138
gevm-inferred-fault on page 141
gevm-instrumentation-alarm on page 144
gevm-internal-fault on page 147
gevm-limit-alarm on page 150
gevm-message on page 153
gevm-notification-message on page 156
gevm-rate-of-change-alarm on page 159
gevm-root-cause on page 162
gevm-state-change on page 165

gevm-action

Use this subclass of gevm-advisory-message to represent an advisory to the operator that a closed loop action needs to be performed.

Class Inheritance Path

gevm-action, gevm-advisory-message, gevm-message, gevm-event, grtl-object, object, grtl-event-source, grtl-item, item

Attributes

Attribute	Description
gevm-advice	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-comments	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-user	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-group	See gevm-message on page 153.
Allowable values:	inherited
Default value:	"unspecified"
gevm-acknowledgement-required	See gevm-message on page 153.
Allowable values:	inherited
Default value:	true
gevm-acknowledged	See gevm-message on page 153.
Allowable values:	inherited

Default value: false

gevm-acknowledged-by-user See gevm-message on page 153.

Allowable values: inherited

Default value: ""

gevm-acknowledged-timestamp See gevm-message on page 153.

Allowable values: inherited

Default value: 0.0

gevm-key See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-category See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-priority See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-repetitions See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-severity See gevm-message on page 153.

Allowable values: inherited

Default value: information

gevm-message See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-detail See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-lifetime See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-creation-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-last-update-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

Methods

gevm-event::gevm-execute-action on page 182

gevm-action-done

Use this subclass of gevm-action to represent an advisory to the operator that a closed loop action was performed by the software.

Class Inheritance Path

gevm-action-done, gevm-action, gevm-advisory-message, gevm-message, gevm-event, grtl-object, object, grtl-event-source, grtl-item, item

Attributes

Attribute	Description
gevm-advice	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-comments	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-user	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-group	See gevm-message on page 153.
Allowable values:	inherited
Default value:	"unspecified"
gevm-acknowledgement-required	See gevm-message on page 153.
Allowable values:	inherited
Default value:	true
gevm-acknowledged	See gevm-message on page 153.
Allowable values:	inherited

Default value: false

gevm-acknowledged-by-user See gevm-message on page 153.

Allowable values: inherited

Default value: ""

gevm-acknowledged-timestamp See gevm-message on page 153.

Allowable values: inherited

Default value: 0.0

gevm-key See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-category See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-priority See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-repetitions See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-severity See gevm-message on page 153.

Allowable values: inherited

Default value: information

gevm-message See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-detail See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-lifetime See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-creation-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-last-update-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-action-to-do

Use this subclass of gevm-action to represent an action that the operator is advised to do.

Class Inheritance Path

gevm-action-to-do, gevm-action, gevm-advisory-message, gevm-message, gevm-event, grtl-object, object, grtl-event-source, grtl-item, item

Attributes

Attribute	Description
gevm-advice	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-comments	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-user	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-group	See gevm-message on page 153.
Allowable values:	inherited
Default value:	"unspecified"
gevm-acknowledgement-required	See gevm-message on page 153.
Allowable values:	inherited
Default value:	true
gevm-acknowledged	See gevm-message on page 153.
Allowable values:	inherited

Default value: false

gevm-acknowledged-by-user See gevm-message on page 153.

Allowable values: inherited

Default value: ""

gevm-acknowledged-timestamp See gevm-message on page 153.

Allowable values: inherited

Default value: 0.0

gevm-key See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-category See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-priority See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-repetitions See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-severity See gevm-message on page 153.

Allowable values: inherited

Default value: information

gevm-message See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-detail See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-lifetime See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-creation-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-last-update-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-advisory

Use this subclass of gevm-advisory to represent an intelligent operator advisory.

Class Inheritance Path

gevm-advisory, gevm-advisory-message, gevm-message, gevm-event, grtl-object, object, grtl-event-source, grtl-item, item

Attributes

Attribute	Description
gevm-advice	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-comments	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-user	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-group	See gevm-message on page 153.
Allowable values:	inherited
Default value:	"unspecified"
gevm-acknowledgement-required	See gevm-message on page 153.
Allowable values:	inherited
Default value:	true
gevm-acknowledged	See gevm-message on page 153.
Allowable values:	inherited
Default value:	false

gevm-acknowledged-by-user See gevm-message on page 153.

Allowable values: inherited

Default value: ""

gevm-acknowledged-timestamp See gevm-message on page 153.

Allowable values: inherited

Default value: 0.0

gevm-key See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-category See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-priority See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-repetitions See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-severity See gevm-message on page 153.

Allowable values: inherited

Default value: information

gevm-message See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-detail See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-lifetime See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-creation-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-last-update-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-advisory-message

An abstract class that is the superior class of any type of advisory message.

Class Inheritance Path

gevm-advisory-message, gevm-message, gevm-event, grtl-object, object, grtl-event-source, grtl-item, item

Attributes

Attribute	Description
gevm-advice	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-comments	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-user	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-group	See gevm-message on page 153.
Allowable values:	inherited
Default value:	"unspecified"
gevm-acknowledgement-required	See gevm-message on page 153.
Allowable values:	inherited
Default value:	true
gevm-acknowledged	See gevm-message on page 153.
Allowable values:	inherited
Default value:	false

gevm-acknowledged-by-user See gevm-message on page 153.

Allowable values: inherited

Default value: ""

gevm-acknowledged-timestamp See gevm-message on page 153.

Allowable values: inherited

Default value: 0.0

gevm-key See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-category See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-priority See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-repetitions See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-severity See gevm-message on page 153.

Allowable values: inherited

Default value: information

gevm-message See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-detail See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-lifetime See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-creation-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-last-update-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-alarm

Use this subclass of gevm-advisory-message to represent an operator advisory that is an alarm.

Class Inheritance Path

gevm-alarm, gevm-advisory-message, gevm-message, gevm-event, grtl-object, object, grtl-event-source, grtl-item, item

Attributes

Attribute	Description
gevm-advice	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-comments	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-user	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-group	See gevm-message on page 153.
Allowable values:	inherited
Default value:	"unspecified"
gevm-acknowledgement-required	See gevm-message on page 153.
Allowable values:	inherited
Default value:	true
gevm-acknowledged	See gevm-message on page 153.
Allowable values:	inherited

Default value: false

gevm-acknowledged-by-user See gevm-message on page 153.

Allowable values: inherited

Default value: ""

gevm-acknowledged-timestamp See gevm-message on page 153.

Allowable values: inherited

Default value: 0.0

gevm-key See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-category See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-priority See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-repetitions See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-severity See gevm-message on page 153.

Allowable values: inherited

Default value: information

gevm-message See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-detail See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-lifetime See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-creation-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-last-update-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-calculated-alarm

Use this subclass of gevm-alarm to represents alarms triggered by comparing a value with a calculated or derived trip setting. This alarm type is used for advanced control such as inference technology, for example, artificial neural networks, rules, or expert systems.

Class Inheritance Path

gevm-calculated-alarm, gevm-alarm, gevm-advisory-message, gevm-message, gevm-event, grtl-object, object, grtl-event-source, grtl-item, item

Attributes

Attribute	Description
gevm-advice	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-comments	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-user	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-group	See gevm-message on page 153.
Allowable values:	inherited
Default value:	"unspecified"
gevm-acknowledgement-required	See gevm-message on page 153.
Allowable values:	inherited
Default value:	true
gevm-acknowledged	See gevm-message on page 153.

Allowable values: inherited

Default value: false

gevm-acknowledged-by-user See gevm-message on page 153.

Allowable values: inherited

Default value: ""

gevm-acknowledged-timestamp See gevm-message on page 153.

Allowable values: inherited

Default value: 0.0

gevm-key See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-category See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-priority See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-repetitions See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-severity See gevm-message on page 153.

Allowable values: inherited

Default value: information

gevm-message See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-detail See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-lifetime See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-creation-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-last-update-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-change-in-process-state-alarm

Use this subclass of gevm-alarm to represents alarms triggered by noting a discrete change of state, for example, putting a process on hold or by-pass.

Class Inheritance Path

gevm-change-in-process-state-alarm, gevm-alarm, gevm-advisory-message, gevm-message, gevm-event, grtl-object, object, grtl-event-source, grtl-item, item

Attributes

Attribute	Description
gevm-advice	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-comments	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-user	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-group	See gevm-message on page 153.
Allowable values:	inherited
Default value:	"unspecified"
gevm-acknowledgement-required	See gevm-message on page 153.
Allowable values:	inherited
Default value:	true
gevm-acknowledged	See gevm-message on page 153.
Allowable values:	inherited

Default value: false

gevm-acknowledged-by-user See gevm-message on page 153.

Allowable values: inherited

Default value: ""

gevm-acknowledged-timestamp See gevm-message on page 153.

Allowable values: inherited

Default value: 0.0

gevm-key See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-category See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-priority See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-repetitions See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-severity See gevm-message on page 153.

Allowable values: inherited

Default value: information

gevm-message See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-detail See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-lifetime See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-creation-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-last-update-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-command

Use this subclass of gevm-message to represent an open-ended command, which is unconfirmed and asynchronous, that is this software system gives to another software system.

Class Inheritance Path

gevm-command, gevm-message, gevm-event, grtl-object, object, grtl-event-source, grtl-item, item

Attributes

Attribute	Description
gevm-advice	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-comments	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-user	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-group	See gevm-message on page 153.
Allowable values:	inherited
Default value:	"unspecified"
gevm-acknowledgement-required	See gevm-message on page 153.
Allowable values:	inherited
Default value:	true
gevm-acknowledged	See gevm-message on page 153.

Allowable values: inherited

Default value: false

gevm-acknowledged-by-user See gevm-message on page 153.

Allowable values: inherited

Default value: ""

gevm-acknowledged-timestamp See gevm-message on page 153.

Allowable values: inherited

Default value: 0.0

gevm-key See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-category See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-priority See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-repetitions See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-severity See gevm-message on page 153.

Allowable values: inherited

Default value: information

gevm-message See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-detail See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-lifetime See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-creation-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-last-update-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-conclusion

Use this subclass of gevm-message to represent conclusions that this software identifies, usually in support of an advisory. For example, observation blocks in a graphical language might generate conclusion messages.

Class Inheritance Path

gevm-conclusion, gevm-message, gevm-event, grtl-object, object, grtl-event-source, grtl-item, item

Attributes

Attribute	Description
gevm-advice	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-comments	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-user	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-group	See gevm-message on page 153.
Allowable values:	inherited
Default value:	"unspecified"
gevm-acknowledgement-required	See gevm-message on page 153.
Allowable values:	inherited
Default value:	true
gevm-acknowledged	See gevm-message on page 153.

Allowable values: inherited

Default value: false

gevm-acknowledged-by-user See gevm-message on page 153.

Allowable values: inherited

Default value: ""

gevm-acknowledged-timestamp See gevm-message on page 153.

Allowable values: inherited

Default value: 0.0

gevm-key See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-category See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-priority See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-repetitions See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-severity See gevm-message on page 153.

Allowable values: inherited

Default value: information

gevm-message See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-detail See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-lifetime See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-creation-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-last-update-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-deviation-alarm

Use this subclass of gevm-alarm to represents alarms triggered by continuously comparing the difference between two continuous signals, for example, two probes or a probe and a set point, and noting a drift greater than a trip setting.

Class Inheritance Path

gevm-deviation-alarm, gevm-alarm, gevm-advisory-message, gevm-message, gevm-event, grtl-object, object, grtl-event-source, grtl-item, item

Attributes

Attribute	Description
gevm-advice	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-comments	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-user	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-group	See gevm-message on page 153.
Allowable values:	inherited
Default value:	"unspecified"
gevm-acknowledgement-required	See gevm-message on page 153.
Allowable values:	inherited
Default value:	true
gevm-acknowledged	See gevm-message on page 153.

Allowable values: inherited

Default value: false

gevm-acknowledged-by-user See gevm-message on page 153.

Allowable values: inherited

Default value: ""

gevm-acknowledged-timestamp See gevm-message on page 153.

Allowable values: inherited

Default value: 0.0

gevm-key See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-category See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-priority See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-repetitions See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-severity See gevm-message on page 153.

Allowable values: inherited

Default value: information

gevm-message See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-detail See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-lifetime See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-creation-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-last-update-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-discrete-signal-alarm

Use this subclass of gevm-alarm to represents alarms triggered by comparing an expected component state with an actual state, for example, a valve on or off.

Class Inheritance Path

gevm-discrete-signal-alarm, gevm-alarm, gevm-advisory-message, gevm-message, gevm-event, grtl-object, object, grtl-event-source, grtl-item, item

Attributes

Attribute	Description
gevm-advice	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-comments	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-user	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-group	See gevm-message on page 153.
Allowable values:	inherited
Default value:	"unspecified"
gevm-acknowledgement-required	See gevm-message on page 153.
Allowable values:	inherited
Default value:	true
gevm-acknowledged	See gevm-message on page 153.
Allowable values:	inherited

Default value: false

gevm-acknowledged-by-user See gevm-message on page 153.

Allowable values: inherited

Default value: ""

gevm-acknowledged-timestamp See gevm-message on page 153.

Allowable values: inherited

Default value: 0.0

gevm-key See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-category See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-priority See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-repetitions See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-severity See gevm-message on page 153.

Allowable values: inherited

Default value: information

gevm-message See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-detail See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-lifetime See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-creation-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-last-update-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-event

The superior class for all GEVM event types, which is a grtl-event-source.

Class Inheritance Path

gevm-event, grtl-object, object, grtl-event-source, grtl-item, item

Attributes

Attribute	Description
gevm-key	A unique key that is automatically generated, based on the class, type, initiator and target.
Allowable values:	Any text
Default value:	""
gevm-category	A user-defined text defining the category of the message.
Allowable values:	Any text
Default value:	""
gevm-priority	A positive integer defining the priority of the message with 1 being the highest priority. Typically priorities are between 1 and 9.
Allowable values:	Any integer
Default value:	0
gevm-repetitions	A positive integer that is the number of times the same message has occurred on the same source and target.
Allowable values:	Any integer
Default value:	0
gevm-message	The text of the message.
Allowable values:	Any text
Default value:	""

gevm-detail The text of the message detail.

Allowable values: Any text

Default value: ""

gevm-lifetime The lifetime of the message, in seconds. If this is a positive number, the message is automatically deleted after it expires.

Allowable values: Any quantity

Default value: 0

gevm-creation-timestamp The G2 timestamp in units of seconds when the message was created, as a quantity.

Allowable values: Any quantity

Default value: 0

gevm-last-update-timestamp The G2 timestamp in units of seconds when the message was last updated, as a quantity.

Allowable values: Any quantity

Default value: 0

Methods

gevm-event::gevm-acknowledge on page 170
gevm-event::gevm-change-priority on page 171
gevm-event::gevm-create-sub-event on page 172
gevm-event::gevm-create-sub-event on page 174
gevm-event::gevm-create-sub-event on page 176
gevm-event::gevm-create-sub-event on page 178
gevm-event::gevm-delete-event on page 180
gevm-event::gevm-execute-action on page 182
gevm-event::gevm-get-event-color on page 183
gevm-event::gevm-get-event-states on page 184
gevm-event::gevm-get-initiating-item on page 185
gevm-event::gevm-get-sub-event-count on page 186
gevm-event::gevm-get-sub-event-count on page 187
gevm-event::gevm-get-sub-events on page 188
gevm-event::gevm-get-sub-events on page 189
gevm-event::gevm-get-superior-event-count on page 190
gevm-event::gevm-get-superior-event-count on page 191
gevm-event::gevm-get-superior-events on page 192

gevm-event::gevm-get-superior-events on page 193
gevm-event::gevm-get-target on page 194
gevm-event::gevm-remove-event on page 195
gevm-event::gevm-show-initiator on page 196
gevm-event::gevm-show-target on page 197
gevm-event::gevm-subordinate-event on page 198
gevm-event::gevm-subordinate-event on page 199
gevm-event::grtl-can-delete on page 200

gevm-event-state

Use this subclass of gevm-event to represent an external fault that is detected and generated outside of this software system.

Class Inheritance Path

gevm-event-state, gevm-event, grtl-object, object, grtl-event-source, grtl-item, item

Attributes

Attribute	Description
gevm-key	See gevm-event on page 127.
Allowable values:	inherited
Default value:	""
gevm-category	See gevm-event on page 127.
Allowable values:	inherited
Default value:	""
gevm-priority	See gevm-event on page 127.
Allowable values:	inherited
Default value:	0
gevm-repetitions	See gevm-event on page 127.
Allowable values:	inherited
Default value:	0
gevm-message	See gevm-event on page 127.
Allowable values:	inherited
Default value:	""
gevm-detail	See gevm-event on page 127.
Allowable values:	inherited
Default value:	""

gevm-lifetime See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-creation-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-last-update-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-external-fault

Use this subclass of gevm-fault to represent an external fault detected and generated outside of this software system.

Class Inheritance Path

gevm-external-fault, gevm-fault, gevm-message, gevm-event, grtl-object, object, grtl-event-source, grtl-item, item

Attributes

Attribute	Description
gevm-advice	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-comments	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-user	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-group	See gevm-message on page 153.
Allowable values:	inherited
Default value:	"unspecified"
gevm-acknowledgement-required	See gevm-message on page 153.
Allowable values:	inherited
Default value:	true
gevm-acknowledged	See gevm-message on page 153.
Allowable values:	inherited

Default value: false

gevm-acknowledged-by-user See gevm-message on page 153.

Allowable values: inherited

Default value: ""

gevm-acknowledged-timestamp See gevm-message on page 153.

Allowable values: inherited

Default value: 0.0

gevm-key See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-category See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-priority See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-repetitions See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-severity See gevm-message on page 153.

Allowable values: inherited

Default value: information

gevm-message See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-detail See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-lifetime See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-creation-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-last-update-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-fault

Abstract class that is the root class for any type of fault message.

Class Inheritance Path

gevm-fault, gevm-message, gevm-event, grtl-object, object, grtl-event-source, grtl-item, item

Attributes

Attribute	Description
gevm-advice	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-comments	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-user	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-group	See gevm-message on page 153.
Allowable values:	inherited
Default value:	"unspecified"
gevm-acknowledgement-required	See gevm-message on page 153.
Allowable values:	inherited
Default value:	true
gevm-acknowledged	See gevm-message on page 153.
Allowable values:	inherited
Default value:	false

gevm-acknowledged-by-user See gevm-message on page 153.

Allowable values: inherited

Default value: ""

gevm-acknowledged-timestamp See gevm-message on page 153.

Allowable values: inherited

Default value: 0.0

gevm-key See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-category See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-priority See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-repetitions See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-severity See gevm-message on page 153.

Allowable values: inherited

Default value: information

gevm-message See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-detail See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-lifetime See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-creation-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-last-update-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-g2-error-message

Use this subclass of gevm-notification-message to represent a G2 error message that is logged to the default GEVM message queue and appears in the GEVM Message Browser. Note that each signalled error generates an individual message and the repetition detection does not apply. To disable displaying messages in the G2 Logbook or Message Board, please refer to the GERR manual on how to disable it.

Class Inheritance Path

gevm-g2-error-message, gevm-notification-message, gevm-message, gevm-event, grtl-object, object, grtl-event-source, grtl-item, item

Attributes

Attribute	Description
gevm-advice	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-comments	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-user	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-group	See gevm-message on page 153.
Allowable values:	inherited
Default value:	"unspecified"
gevm-acknowledgement-required	See gevm-message on page 153.
Allowable values:	inherited
Default value:	true

gevm-acknowledged See gevm-message on page 153.

Allowable values: inherited

Default value: false

gevm-acknowledged-by-user See gevm-message on page 153.

Allowable values: inherited

Default value: ""

gevm-acknowledged-timestamp See gevm-message on page 153.

Allowable values: inherited

Default value: 0.0

gevm-key See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-category See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-priority See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-repetitions See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-severity See gevm-message on page 153.

Allowable values: inherited

Default value: information

gevm-message See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-detail See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-lifetime See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-creation-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-last-update-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-inferred-fault

Use this subclass of gevm-fault to represent faults detected and generated by advanced control plug-ins that use inference technology, such as artificial neural networks or expert systems.

Class Inheritance Path

gevm-inferred-fault, gevm-fault, gevm-message, gevm-event, grtl-object, object, grtl-event-source, grtl-item, item

Attributes

Attribute	Description
gevm-advice	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-comments	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-user	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-group	See gevm-message on page 153.
Allowable values:	inherited
Default value:	"unspecified"
gevm-acknowledgement-required	See gevm-message on page 153.
Allowable values:	inherited
Default value:	true
gevm-acknowledged	See gevm-message on page 153.

Allowable values: inherited

Default value: false

gevm-acknowledged-by-user See gevm-message on page 153.

Allowable values: inherited

Default value: ""

gevm-acknowledged-timestamp See gevm-message on page 153.

Allowable values: inherited

Default value: 0.0

gevm-key See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-category See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-priority See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-repetitions See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-severity See gevm-message on page 153.

Allowable values: inherited

Default value: information

gevm-message See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-detail See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-lifetime See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-creation-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-last-update-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-instrumentation-alarm

Use this subclass of gevm-alarm to represents alarms triggered by faults within the computer system or instrumentation diagnostics, such as overflow, low battery, or database not responding.

Class Inheritance Path

gevm-instrumentation-alarm, gevm-alarm, gevm-advisory-message, gevm-message, gevm-event, grtl-object, object, grtl-event-source, grtl-item, item

Attributes

Attribute	Description
gevm-advice	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-comments	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-user	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-group	See gevm-message on page 153.
Allowable values:	inherited
Default value:	"unspecified"
gevm-acknowledgement-required	See gevm-message on page 153.
Allowable values:	inherited
Default value:	true
gevm-acknowledged	See gevm-message on page 153.

Allowable values: inherited

Default value: false

gevm-acknowledged-by-user See gevm-message on page 153.

Allowable values: inherited

Default value: ""

gevm-acknowledged-timestamp See gevm-message on page 153.

Allowable values: inherited

Default value: 0.0

gevm-key See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-category See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-priority See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-repetitions See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-severity See gevm-message on page 153.

Allowable values: inherited

Default value: information

gevm-message See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-detail See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-lifetime See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-creation-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-last-update-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-internal-fault

Use this subclass of gevm-fault to represent an internal fault detected and generated by this software, using built-in techniques such as limit failures, rate of change failures, or connection failures.

Class Inheritance Path

gevm-internal-fault, gevm-fault, gevm-message, gevm-event, grtl-object, object, grtl-event-source, grtl-item, item

Attributes

Attribute	Description
gevm-advice	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-comments	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-user	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-group	See gevm-message on page 153.
Allowable values:	inherited
Default value:	"unspecified"
gevm-acknowledgement-required	See gevm-message on page 153.
Allowable values:	inherited
Default value:	true
gevm-acknowledged	See gevm-message on page 153.

Allowable values: inherited

Default value: false

gevm-acknowledged-by-user See gevm-message on page 153.

Allowable values: inherited

Default value: ""

gevm-acknowledged-timestamp See gevm-message on page 153.

Allowable values: inherited

Default value: 0.0

gevm-key See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-category See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-priority See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-repetitions See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-severity See gevm-message on page 153.

Allowable values: inherited

Default value: information

gevm-message See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-detail See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-lifetime See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-creation-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-last-update-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-limit-alarm

Use this subclass of gevm-alarm to represent alarms triggered by continuously comparing a signal with upper and/or lower limits and noting tripping by crossing the trip setting.

Class Inheritance Path

gevm-limit-alarm, gevm-alarm, gevm-advisory-message, gevm-message, gevm-event, grtl-object, object, grtl-event-source, grtl-item, item

Attributes

Attribute	Description
gevm-advice	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-comments	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-user	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-group	See gevm-message on page 153.
Allowable values:	inherited
Default value:	"unspecified"
gevm-acknowledgement-required	See gevm-message on page 153.
Allowable values:	inherited
Default value:	true
gevm-acknowledged	See gevm-message on page 153.

Allowable values: inherited

Default value: false

gevm-acknowledged-by-user See gevm-message on page 153.

Allowable values: inherited

Default value: ""

gevm-acknowledged-timestamp See gevm-message on page 153.

Allowable values: inherited

Default value: 0.0

gevm-key See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-category See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-priority See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-repetitions See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-severity See gevm-message on page 153.

Allowable values: inherited

Default value: information

gevm-message See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-detail See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-lifetime See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-creation-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-last-update-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-message

Use this subclass of gevm-event to represent any type of operator message.

Class Inheritance Path

gevm-message, gevm-event, grtl-object, object, grtl-event-source, grtl-item, item

Attributes

Attribute	Description
gevm-advice	The text of the message advice.
Allowable values:	Any text
Default value:	""
gevm-comments	Any user-defined comment associated with the message, as a text.
Allowable values:	Any text
Default value:	""
gevm-assigned-to-user	The user name assigned to the message.
Allowable values:	Any text
Default value:	""
gevm-assigned-to-group	The group assigned to the message.
Allowable values:	Any text
Default value:	"unspecified"
gevm-acknowledgement-required	True if the message needs to be acknowledged.
Allowable values:	Any truth-value
Default value:	true
gevm-acknowledged	True if the message has been acknowledged, false otherwise.
Allowable values:	Any truth-value

Default value: false

gevm-acknowledged-by-user The name of the user that acknowledged the message.

Allowable values: Any text

Default value: ""

gevm-acknowledged-timestamp The G2 timestamp when the message was acknowledged, as a quantity.

Allowable values: Any quantity

Default value: 0.0

gevm-key See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-category See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-priority See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-repetitions See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-severity The message severity.

Allowable values: One of the following symbols: critical, warning, root-cause, action, or information

Default value: information

gevm-message See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-detail See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-lifetime See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-creation-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-last-update-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-notification-message

Use this subclass of gevm-message to represent a message that is a notification to the operator. Note that repetition detection does not apply to notification messages.

Class Inheritance Path

gevm-notification-message, gevm-message, gevm-event, grtl-object, object, grtl-event-source, grtl-item, item

Attributes

Attribute	Description
gevm-advice	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-comments	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-user	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-group	See gevm-message on page 153.
Allowable values:	Any text
Default value:	"unspecified"
gevm-acknowledgement-required	See gevm-message on page 153.
Allowable values:	inherited
Default value:	true
gevm-acknowledged	See gevm-message on page 153.

Allowable values: inherited

Default value: false

gevm-acknowledged-by-user See gevm-message on page 153.

Allowable values: inherited

Default value: ""

gevm-acknowledged-timestamp See gevm-message on page 153.

Allowable values: inherited

Default value: 0.0

gevm-key See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-category See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-priority See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-repetitions See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-severity See gevm-message on page 153.

Allowable values: inherited

Default value: information

gevm-message See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-detail See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-lifetime See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-creation-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-last-update-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-rate-of-change-alarm

Use this subclass of gevm-alarm to represents alarms triggered by a rate of change of a measurement or a derived variable that exceeds a trip setting.

Class Inheritance Path

gevm-rate-of-change-alarm, gevm-alarm, gevm-advisory-message, gevm-message, gevm-event, grtl-object, object, grtl-event-source, grtl-item, item

Attributes

Attribute	Description
gevm-advice	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-comments	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-user	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-group	See gevm-message on page 153.
Allowable values:	Any text
Default value:	"unspecified"
gevm-acknowledgement-required	See gevm-message on page 153.
Allowable values:	inherited
Default value:	true
gevm-acknowledged	See gevm-message on page 153.
Allowable values:	inherited

Default value: false

gevm-acknowledged-by-user See gevm-message on page 153.

Allowable values: inherited

Default value: ""

gevm-acknowledged-timestamp See gevm-message on page 153.

Allowable values: inherited

Default value: 0.0

gevm-key See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-category See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-priority See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-repetitions See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-severity See gevm-message on page 153.

Allowable values: inherited

Default value: information

gevm-message See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-detail See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-lifetime See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-creation-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-last-update-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-root-cause

Use this subclass of gevm-message to represent a root cause message to the operator that internal logic detects.

Class Inheritance Path

gevm-root-cause, gevm-message, gevm-event, grtl-object, object, grtl-event-source, grtl-item, item

Attributes

Attribute	Description
gevm-advice	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-comments	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-user	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-group	See gevm-message on page 153.
Allowable values:	Any text
Default value:	"unspecified"
gevm-acknowledgement-required	See gevm-message on page 153.
Allowable values:	inherited
Default value:	true
gevm-acknowledged	See gevm-message on page 153.
Allowable values:	inherited

Default value: false

gevm-acknowledged-by-user See gevm-message on page 153.

Allowable values: inherited

Default value: ""

gevm-acknowledged-timestamp See gevm-message on page 153.

Allowable values: inherited

Default value: 0.0

gevm-key See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-category See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-priority See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-repetitions See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-severity See gevm-message on page 153.

Allowable values: inherited

Default value: information

gevm-message See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-detail See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-lifetime See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-creation-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-last-update-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-state-change

Use this subclass of gevm-message to represent state changes that require special status as events. Not all state changes need to be treated as a gevm-event.

Class Inheritance Path

gevm-state-change, gevm-message, gevm-event, grtl-object, object, grtl-event-source, grtl-item, item

Attributes

Attribute	Description
gevm-advice	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-comments	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-user	See gevm-message on page 153.
Allowable values:	inherited
Default value:	""
gevm-assigned-to-group	See gevm-message on page 153.
Allowable values:	Any text
Default value:	"unspecified"
gevm-acknowledgement-required	See gevm-message on page 153.
Allowable values:	inherited
Default value:	true
gevm-acknowledged	See gevm-message on page 153.
Allowable values:	inherited

Default value: false

gevm-acknowledged-by-user See gevm-message on page 153.

Allowable values: inherited

Default value: ""

gevm-acknowledged-timestamp See gevm-message on page 153.

Allowable values: inherited

Default value: 0.0

gevm-key See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-category See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-priority See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-repetitions See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-severity See gevm-message on page 153.

Allowable values: inherited

Default value: information

gevm-message See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-detail See gevm-event on page 127.

Allowable values: inherited

Default value: ""

gevm-lifetime See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-creation-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

gevm-last-update-timestamp See gevm-event on page 127.

Allowable values: inherited

Default value: 0

Event Management Operations

The basic operations for managing events include:

- Acknowledging events.
- Acknowledging event states.
- Changing event priority.
- Creating subevents.
- Deleting events.
- Sending e-mail when a message occurs.
- Removing events from a queue.
- Executing actions.
- Getting the event state, color, initiating item, target item, subevents, and superior event.
- Showing the initiator and target of an event.
- Subordinating an event to a parent event.
- Determining if an event can be deleted.

Methods

gevm-event::gevm-acknowledge on page 170
gevm-event::gevm-change-priority on page 171
gevm-event::gevm-create-sub-event on page 172
gevm-event::gevm-create-sub-event on page 174
gevm-event::gevm-create-sub-event on page 176
gevm-event::gevm-create-sub-event on page 178
gevm-event::gevm-delete-event on page 180
gevm-event::gevm-email-notification on page 181
gevm-event::gevm-execute-action on page 182
gevm-event::gevm-get-event-color on page 183
gevm-event::gevm-get-event-states on page 184
gevm-event::gevm-get-initiating-item on page 185
gevm-event::gevm-get-sub-event-count on page 186
gevm-event::gevm-get-sub-event-count on page 187
gevm-event::gevm-get-sub-events on page 188
gevm-event::gevm-get-sub-events on page 189
gevm-event::gevm-get-superior-event-count on page 190
gevm-event::gevm-get-superior-event-count on page 191
gevm-event::gevm-get-superior-events on page 192
gevm-event::gevm-get-superior-events on page 193
gevm-event::gevm-get-target on page 194

gevm-event::gevm-remove-event on page 195
gevm-event::gevm-show-initiator on page 196
gevm-event::gevm-show-target on page 197
gevm-event::gevm-subordinate-event on page 198
gevm-event::gevm-subordinate-event on page 199
gevm-event::grtl-can-delete on page 200

Functions

gevm-acknowledged on page 201
gevm-event-has-sub-event on page 202
gevm-event-is-sub-event on page 203
gevm-get-event-class-names on page 204
gevm-has-initiating-item on page 205
gevm-has-target on page 206

gevm-event::gevm-acknowledge

Synopsis

gevm-event::gevm-acknowledge
(*event*: gevm-event, *client*: ui-client-item)

Argument	Description
event	The event to acknowledge.
client	The client, such as a G2 window.

Description

Acknowledges an event.

gevm-event::gevm-change-priority

Synopsis

gevm-event::gevm-change-priority
(*event*: gevm-event, *new-priority*: integer)

Argument	Description
event	The event whose priority to change.
new-priority	The new priority of the event, as an integer.

Description

Changes the priority of an event.

gevm-event::gevm-create-sub-event

Synopsis

gevm-event::gevm-create-sub-event
(*parent-event*: gevm-event, *event-class*: gevm-event-class, *category*: text, *priority*: integer, *message*: text, *detail*: text, *advice*: text, *acknowledgment-required*: truth-value, *comment*: text, *initiating-item*: item, *target-item*: item, *lifetime*: quantity, *publish-event*: truth-value, *relation-type*: symbol, *client*: ui-client-item)
–> *event*: gevm-event

Argument	Description
parent-event	The parent event of this new sub-event.
event-class	The class of the event.
category	The event category.
priority	A priority from 0 (most urgent) to 9 (least urgent).
message	The event message.
detail	The event detail.
advice	The event advice.
acknowledgment-required	True if message needs to be acknowledged.
comment	Default comment.
initiating-item	The item sending the event.
target-item	The subject of the event against which the event is targeted.
lifetime	The maximum time for the event to exist.
publish-event	Publishes event if true
relation-type	A symbol that names one of these relation types, used for message correlation: an-information-provider-for, a-possible-cause-for, a-possible-action-to-do-for, an-action-for-message, or an-action-done-for.
client	The client, such as a G2 window.

Return Value	Description
event	The new event.

Description

Creates an event that is part of a compound event with message text, details, advice, acknowledgement requirement, and comment. It has no provision for text translation or text substitution.

A subevent is automatically subordinated to his parent. See gevm-event::gevm-subordinate-event on page 198.

gevm-event::gevm-create-sub-event

Synopsis

gevm-event::gevm-create-sub-event
(*parent-event*: gevm-event, *event-class*: gevm-event-class, *category*: text, *priority*: integer, *message-key*: symbol, *message-args*: sequence, *detail-key*: symbol, *detail-args*: sequence, *advice-key*: symbol, *advice-args*: sequence, *acknowledgment-required*: truth-value, *comment*: text, *initiating-item*: item-or-value, *target-item*: item-or-value, *lifetime*: quantity, *publish-event*: truth-value, *relation-type*: symbol, *client*: ui-client-item)
–> *event*: gevm-event

.

Argument	Description
parent-event	The parent event of this new subevent.
event-class	The class of the event.
category	The event category.
priority	A priority from 0 (most urgent) to 9 (least urgent).
message-key	A lookup key for the event message.
message-args	A sequence of simple values that will be inserted into the message text.
detail-key	A lookup key for the event detail.
detail-args	A sequence of simple values that will be inserted into the message text.
advice-key	A lookup key for the event advice.
advice-args	A sequence of simple values that will be inserted into the advice text.
acknowledgment-required	True if the message needs to be acknowledged.
comment	Default comment.
initiating-item	The item sending the event, which must be an instance of the gevm-event-initiating-item class, or the symbol none
target-item	The subject of the event against which the event is targeted, or the symbol none

lifetime	The maximum time for the event to exist.
publish-event	Publishes event if true
relation-type	A symbol that names one of these relation types, used for message correlation: an-information-provider-for, a-possible-cause-for, a-possible-action-to-do-for, an-action-for-message, or an-action-done-for.
client	The client, such as a G2 window.

Return Value	Description
event	The new event.

Description

Creates an event that is part of a compound event and supports message, detail, and advice text translation and text substitution, as well as acknowledgement requirement and comment.

A subevent is automatically subordinated to his parent. See gevm-event::gevm-subordinate-event on page 198.

gevm-event::gevm-create-sub-event

Synopsis

gevm-event::gevm-create-sub-event
(*parentevent*: gevm-event, *event-class*: gevm-event-class, *category*: text, *priority*: integer, *message*: text, *detail*: text, *initiating-item*: item, *target-item*: item, *lifetime*: quantity, *publish-event*: truth-value, *relation-type*: symbol, *client*: ui-client-item)
–> *event*: gevm-event

Argument	Description
parent-event	The parent event of this new sub-event.
event-class	The class of the event.
category	The event category.
priority	A priority from 0 (most urgent) to 9 (least urgent).
message	The event message.
detail	The event detail.
initiating-item	The item sending the event.
target-item	The subject of the event against which the event is targeted.
lifetime	The maximum time for the event to exist.
publish-event	Publishes event if true
relation-type	A symbol that names one of these relation types, used for message correlation: an-information-provider-for, a-possible-cause-for, a-possible-action-to-do-for, an-action-for-message, or an-action-done-for.
client	The client, such as a G2 window.

Return Value	Description
event	The new event.

Description

Creates an event that is part of a compound event with no provision for advice, acknowledgement requirement, or comment, and no provision for text translation and text substitution.

A subevent is automatically subordinated to his parent. See gevm-event::gevm-subordinate-event on page 198.

gevm-event::gevm-create-sub-event

Synopsis

gevm-event::gevm-create-sub-event
(*parent-event*: gevm-event, *event-class*: gevm-event-class, *category*: text, *priority*: integer, *message-key*: symbol, *message-args*: sequence, *detail-key*: symbol, *detail-args*: sequence, *initiating-item*: item-or-value, *target-item*: item-or-value, *lifetime*: quantity, *publish-event*: truth-value, *relation-type*: symbol, *client*: ui-client-item)
–> *event*: gevm-event

Argument	Description
parent-event	The parent event of this new sub-event.
event-class	The class of the event.
category	The event category.
priority	A priority from 0 (most urgent) to 9 (least urgent).
message-key	A lookup key for the event message.
message-args	A sequence of simple values that will be inserted into the message text.
detail-key	A lookup key for the event detail.
detail-args	A sequence of simple values that will be inserted into the message text.
initiating-item	The item sending the event.
target-item	The subject of the event against which the event is targeted.
lifetime	The maximum time for the event to exist.
publish-event	Publishes event if true
relation-type	A symbol that names one of these relation types, used for message correlation: an-information-provider-for, a-possible-cause-for, a-possible-action-to-do-for, an-action-for-message, or an-action-done-for.
client	The client, such as a G2 window.

Return Value	Description
event	The new event.

Description

Creates an event that is part of a compound events and supports message and detail text translation and text substitution. It has no provision for advice, acknowledgement requirement, and comment.

A subevent is automatically subordinated to his parent. See gevm-event::gevm-subordinate-event on page 198.

gevm-event::gevm-delete-event

Synopsis

gevm-event::gevm-delete-event
(*event*: gevm-event)

Argument	Description
event	The event to delete.

Description

Removes the event from all queues and deletes the event.

gevm-event::gevm-email-notification

Synopsis

gevm-event::gevm-email-notification
(*message*: class gevm-message, *action*: symbol, *to-addresses*: sequence, *html-email*: truth-value, *jmail-interface*: class jmail-interface)
–> *status*: text

Argument	Description
message	The message to send as email.
action	A symbol to insert in the message, for example, "The following message has been *action*."
to-addresses	A sequence of texts that are the email addresses to which to send the message.
html-email	True to send the message as an HTML document; false to send it as plain text.
jmail-interface	The jmail-interface object used for sending the email message.

Return Value	Description
status	The status returned by jmail-send-mail or an empty string if the interface was not connected.

Description

Formats the message as email and sends it via the specified jmail-interface. See the *G2 JMail Bridge User's Guide* for details.

gevm-event::gevm-execute-action

Synopsis

gevm-event::gevm-execute-action
(*event*: gevm-event, *client*: ui-client-item)

Argument	Description
event	The source object for which to execute the action.
client	The client, such as a G2 window.

Description

Executes an action associated with an event. You typically use this method for gevm-action classes.

gevm-event::gevm-get-event-color

Synopsis

gevm-event::gevm-get-event-color
(*event*: gevm-event, *key-or-attribute*: symbol)
–> *foreground*: symbol, *background*: symbol, *frame*: symbol

Attribute	Description
event	The event whose color is to be looked up.
key-or-attribute	The attribute for which the color is to be looked up.

Return Value	Description
foreground	The background color of the message.
background	The foreground color of the message.
frame	The frame color of the message.

Description

Returns the color associated with an attribute of a message, based on a lookup table.

gevm-event::gevm-get-event-states

Synopsis

gevm-event::gevm-get-event-states
(*event*: gevm-event)
–> *states*: sequence

Argument	Description
event	The event whose event history to return.

Return Value	Description
states	A sequence of structures that represent the history of actions for the event.

Description

Returns the history of event states, as a sequence of structures. The syntax for each structure is:

structure
(timestamp: *timestamp*, event-action: *symbol*)

- timestamp: The UNIX timestamp of when the even occurred.
- event-action: The action name as a symbol.

gevm-event::gevm-get-initiating-item

Synopsis

gevm-event::gevm-get-initiating-item
(*event*: gevm-event)
–> *initiating-item*: item-or-value

Argument	Description
event	The event whose initiating item to return.

Return Value	Description
item-or-value	The initiating item of the event or the symbol none.

Description

Returns the initiating item of the event or the symbol none if the event does not specify an initiating item.

gevm-event::gevm-get-sub-event-count

Synopsis

gevm-get-sub-event-count
(*event*: class gevm-event)
–> *count*: integer

Argument	Description
event	The event whose subevents to count.

Return Value	Description
count	The number of subevents for the event.

Description

Returns the count of all subevents independent of the subordination relation type.

gevm-event::gevm-get-sub-event-count

Synopsis

gevm-get-sub-event-count
(*event*: class gevm-event, *relation-type*: symbol)
–> *count*: integer

Argument	Description
event	The event whose subevents to count.
relation-type	A symbol that names one of these relation types, used for message correlation: an-information-provider-for, a-possible-cause-for, a-possible-action-to-do-for, an-action-for-message, or an-action-done-for.

Return Value	Description
count	The number of subevents for the event.

Description

Returns the count of all subevents for a specified subordinate relation type.

gevm-event::gevm-get-sub-events

Synopsis

gevm-event::gevm-get-sub-events
(*event*: gevm-event)
–> *subevents*: sequence

Argument	Description
event	The event whose subevents to return.

Return Value	Description
subevents	A sequence of subevents or an empty sequence.

Description

Returns a sequence of subevents merging all subordinate relation types together into one sequence.

gevm-event::gevm-get-sub-events

Synopsis

gevm-get-sub-events
(*event*: class gevm-event, *relation-type*: symbol)
–> *subevents*: sequence

Argument	Description
event	The event whose subevents to return.
relation-type	A symbol that names one of these relation types, used for message correlation: an-information-provider-for, a-possible-cause-for, a-possible-action-to-do-for, an-action-for-message, or an-action-done-for.

Return Value	Description
subevents	A sequence of subevents or an empty sequence.

Description

Returns a sequence of subevents for a specific relation type.

gevm-event::gevm-get-superior-event-count

Synopsis

gevm-get-superior-event-count
(*event*: class gevm-event)
–> *count*: integer

Argument	Description
event	The event whose superior events to count.

Return Value	Description
count	The number of superior events for the event.

Description

Returns the count of all superior events independent of the subordination relation type.

gevm-event::gevm-get-superior-event-count

Synopsis

gevm-get-superior-event-count
(*event*: class gevm-event, *relation-type*: symbol)
–> *count*: integer

Argument	Description
event	The event whose superior events to count.
relation-type	A symbol that names one of these relation types, used for message correlation: an-information-provider-for, a-possible-cause-for, a-possible-action-to-do-for, an-action-for-message, or an-action-done-for.

Return Value	Description
count	The number of superior events for the event.

Description

Returns the count of all superior events for a specified subordinate relation type.

gevm-event::gevm-get-superior-events

Synopsis

gevm-event::gevm-get-superior-events
(*event*: gevm-event)
–> *superior-events*: sequence

Argument	Description
event	The event whose superior event to return.

Return Value	Description
superior-events	A sequence of events or an empty sequence.

Description

Returns a sequence of superior events of an event, independent of the subordinate relation type, or an empty sequence if the event does not have a parent events.

gevm-event::gevm-get-superior-events

Synopsis

gevm-event::gevm-get-superior-events
(*event*: gevm-event, *relation-type*: symbol)
–> *superior-events*: sequence

Argument	Description
event	The event whose superior event to return.
relation-type	A symbol that names one of these relation types, used for message correlation: an-information-provider-for, a-possible-cause-for, a-possible-action-to-do-for, an-action-for-message, or an-action-done-for.

Return Value	Description
superior-events	A sequence of events or an empty sequence.

Description

Returns a sequence of superior events of an event for a specific relation type, or an empty sequence if the event does not have a parent events of the specified relation type.

gevm-event::gevm-get-target

Synopsis

gevm-event::gevm-get-target
(*event*: gevm-event)
–> *target-event*: item-or-value

Argument	Description
event	The event whose target to return.

Return Value	Description
target-event	The target of the event or the symbol none.

Description

Returns the target of the event or the symbol none if the event does not specify a target.

gevm-event::gevm-remove-event

Synopsis

gevm-event::gevm-remove-event
(*event*: gevm-event)

Argument	Description
event	The event to remove.

Description

Removes an event from all queues in which it exists. The event still exists.

gevm-event::gevm-show-initiator

Synopsis

gevm-event::gevm-show-initiator
(*event*: gevm-event, *client*: ui-client-item)

Argument	Description
event	The event whose initiating item to show.
client	The client, such as a G2 window.

Description

Places an indicator arrow next to the initiating item of an event.

gevm-event::gevm-show-target

Synopsis

gevm-event::gevm-show-target
(*event*: gevm-event, *client*: ui-client-item)

Argument	Description
event	The event whose target item to show.
client	The client, such as a G2 window.

Description

Places an indicator arrow next to the target item of an event.

gevm-event::gevm-subordinate-event

Synopsis

gevm-event::gevm-subordinate-event
(*subevent*: gevm-event, *parent-event*: gevm-event)

Argument	Description
subevent	The event to subordinate.
parent-event	The parent event.

Description

Subordinates an event to another event, making one the subevent of the other. The subordination relation type between the parent and sub event is an-information-provider-for relation type.

gevm-event::gevm-subordinate-event

Synopsis

gevm-event::gevm-subordinate-event
(*subevent*: gevm-event, *parent-event*: gevm-event, *relation-type*: symbol)

Argument	Description
subevent	The event to subordinate.
parent-event	The parent event.
relation-type	A symbol that names one of these relation types, used for message correlation: an-information-provider-for, a-possible-cause-for, a-possible-action-to-do-for, an-action-for-message, or an-action-done-for.

Description

Subordinates an event to another event, making one the subevent of the other, using the specified relation type.

gevm-event::grtl-can-delete

Synopsis

gevm-event::grtl-can-delete
(*event*: gevm-event, *win*: ui-client-item)
–> *return-value*: truth-value

Argument	Description
event	The event to test.
win	The client, such as a G2 window.

Return Value	Description
return-value	Always returns true indicating the event can be deleted.

Description

Determines if an event can be deleted. This method always returns true for gevm-event instances.

gevm-acknowledged

Synopsis

gevm-acknowledged
(*event*: gevm-event)
–> *return-value*: truth-value

Argument	Description
event	The event to test.

Return Value	Description
return-value	True if the event has been acknowledged.

Description

Returns true if the event has been acknowledged.

gevm-event-has-sub-event

Synopsis

gevm-event-has-sub-event
(*event*: class gevm-event)
–> *return-value*: truth-value

Argument	Description
event	The event to test.

Return Value	Description
return-value	True if the event has one or more subevents.

Description

Returns true if the event has one or more subevents.

gevm-event-is-sub-event

Synopsis

gevm-event-is-sub-event
(*event*: class gevm-event)
–> *return-value*: truth-value

Argument	Description
event	The event to test.

Return Value	Description
return-value	True if the event has is the subevent of another event.

Description

Returns true if the event is a subevent of another event.

gevm-get-event-class-names

Synopsis

gevm-get-event-class-names
(*item*: item-or-value)
–> *event-classes*: sequence

Argument	Description
item	The item whose event class names to return.

Return Value	Description
event-classes	A sequence of event class names.

Description

Returns a list of all possible event classes in alphabetical order for a given item. You can use this function in user-defined methods for subclasses to return only those classes relevant to a specific context. By default, this function returns all possible names.

gevm-has-initiating-item

Synopsis

gevm-has-initiating-item
(*event*: gevm-event)
–> *return-value*: truth-value

Argument	Description
event	The event to test.

Argument	Description
return-value	True if the event has an initiating item.

Description

Return true if the event has an initiating item.

gevm-has-target

Synopsis

gevm-has-target
(*event*: gevm-event)
–> *return-value*: truth-value

Argument	Description
event	The event to test.

Argument	Description
return-value	True if the event has an target item.

Description

Return true if the event has a target item.

Target Event Operations

These API procedures allow you to perform a variety of tasks, given the target item of an event, including:

- Acknowledge events related to a target.
- Delete events related to a target.
- Delete events related to a target that occur within or outside of a given time period.
- Return a list of events related to a target that are created or are updated within or outside of a given time period.
- Return the number of events related to a target.
- Return a list of events related to a target.
- Remove events related to a target.

The APIs provide different versions of that allow you to perform the action on:

- All events related to a target.
- A subset of events that are filtered, based on the event type and category.

The target event operations are:

item::gevm-acknowledge-all-events-for-target
item::gevm-acknowledge-events-for-target
item::gevm-acknowledge-events-for-target
item::gevm-delete-all-events-for-target
item::gevm-delete-collected-events-created-outside-time-period-for-target
item::gevm-delete-collected-events-created-within-time-period-for-target
item::gevm-delete-events-for-target
item::gevm-get-collected-events-created-outside-time-period-for-target
item::gevm-get-collected-events-created-within-time-period-for-target
item::gevm-get-collected-events-updated-outside-time-period-for-target
item::gevm-get-collected-events-updated-within-time-period-for-target
item::gevm-get-event-count-for-target
item::gevm-get-event-count-for-target
item::gevm-get-events-for-target
item::gevm-get-events-for-target
item::gevm-get-events-for-target
item::gevm-remove-all-events-for-target
item::gevm-remove-events-for-target

item::gevm-acknowledge-all-events-for-target

Synopsis

item::gevm-acknowledge-all-events-for-target
(*target*: item, *client*: ui-client-item)

Argument	Description
target	The target item for an event.
client	The client, such as a G2 window.

Description

Acknowledges all events related to a target.

item::gevm-acknowledge-events-for-target

Synopsis

item::gevm-acknowledge-events-for-target
(*target*: item, *event-type*: symbol, *category-filter*: sequence,
client: ui-client-item)

Argument	Description
target	The target item for an event.
event-type	The event class for filtering events.
category-filter	A sequence of categories, as symbols, for filtering events, or an empty sequence.
client	The client, such as a G2 window.

Description

Acknowledges events of a certain type and category related to a target.

Specify *event-type* to acknowledge only events of the specified class, or any subclass. To acknowledge all types of events, specify gevm-event as the event type.

Specify *category-filter* to acknowledge only events in the specified categories or an empty sequence to acknowledge events in all categories.

item::gevm-acknowledge-events-for-target

Synopsis

item::gevm-acknowledge-events-for-target
(*target*: item, *event-types*: sequence, *client*: ui-client-item)

Argument	Description
target	The target item for an event.
event-types	A sequence of event classes for filtering events.
client	The client, such as a G2 window.

Description

Acknowledges events of a set of types related to a target, or any subclass.

item::gevm-delete-all-events-for-target

Synopsis

item::gevm-delete-all-events-for-target
(*target*: item)

Argument	Description
target	The target item for an event.

Description

Deletes all events related to a target.

item::gevm-delete-collected-events-created-outside-time-period-for-target

Synopsis

item::gevm-delete-collected-events-created-outside-time-period-for-target
(*target*: item, *start-timestamp*: quantity, *end-timestamp*: quantity,
event-type: symbol, *category-filter*: sequence,
inclusive-event-states: sequence, *exclusive-event-states*: sequence)

Argument	Description
target	The target item for an event.
start-timestamp	The start time for considering events for deletion.
end-timestamp	The end time for considering events for deletion.
event-type	The event class for filtering events.
category-filter	A sequence of categories, as symbols, for filtering events, or an empty sequence.
inclusive-event-states	A sequence of event states, as symbols, for including events to delete, or an empty sequence.
exclusive-event-states	A sequence of event states, as symbols, for excluding events to delete, or an empty sequence.

Description

Deletes events created outside of a specified time period and related to a target. You express timestamps in G2 time.

Specify *event-type* to delete only events of the specified class, or any subclass. To delete all types of events, specify gevm-event as the event type.

Specify *category-filter* to delete only events in the specified categories or an empty sequence to delete events in all categories.

Specify *inclusive-event-states* to delete only events whose history includes any of the specified event states. Specify *exclusive-event-states* to delete only events whose history does not include any of the specified event states.

item::gevm-delete-collected-events-created-within-time-period-for-target

Synopsis

item::gevm-delete-collected-events-created-within-time-period-for-target
(*target*: item, *start-timestamp*: quantity, *end-timestamp*: quantity,
event-type: symbol, *category-filter*: sequence,
inclusive-event-states: sequence, *exclusive-event-states*: sequence)

Argument	Description
target	The target item for an event.
start-timestamp	The start time for considering events for deletion.
end-timestamp	The end time for considering events for deletion.
event-type	The event class for filtering events.
category-filter	A sequence of categories, as symbols, for filtering events, or an empty sequence.
inclusive-event-states	A sequence of event states, as symbols, for including events to delete, or an empty sequence.
exclusive-event-states	A sequence of event states, as symbols, for excluding events to delete, or an empty sequence.

Description

Deletes events created within of a specified time period and related to a target. You express timestamps in G2 time.

Specify *event-type* to delete only events of the specified class, or any subclass. To delete all types of events, specify gevm-event as the event type.

Specify *category-filter* to delete only events in the specified categories or an empty sequence to delete events in all categories.

Specify *inclusive-event-states* to delete only events whose history includes any of the specified event states. Specify *exclusive-event-states* to delete only events whose history does not include any of the specified event states.

item::gevm-delete-events-for-target

Synopsis

item::gevm-delete-events-for-target
(*target*: item, *event-type*: symbol, *category-filter*: sequence)

Argument	Description
target	The target item for an event.
event-type	The event class for filtering events.
category-filter	A sequence of categories, as symbols, for filtering events, or an empty sequence.

Description

Deletes events of a certain type and category related to a target.

Specify *event-type* to delete only events of the specified class, or any subclass. To delete all types of events, specify gevm-event as the event type.

Specify *category-filter* to delete only events in the specified categories or an empty sequence to delete events in all categories.

item::gevm-delete-events-for-target

Synopsis

item::gevm-delete-events-for-target
    (*target*: item, *event-types*: sequence)

Argument	Description
target	The target item for an event.
event-types	A sequence of event classes for filtering events.

Description

Deletes events of a set of types related to a target, or any subclass.

item::gevm-get-collected-events-created-outside-time-period-for-target

Synopsis

item::gevm-get-collected-events-created-outside-time-period-for-target
(*target*: item, *start-timestamp*: quantity, *end-timestamp*: quantity,
event-type: symbol, *category-filter*: sequence,
inclusive-event-states: sequence, *exclusive-event-states*: sequence)
–> *events*: sequence

Argument	Description
target	The target item for an event.
start-timestamp	The start time for considering events for selection.
end-timestamp	The end time for considering events for selection.
event-type	The event class for filtering events.
category-filter	A sequence of categories, as symbols, for filtering events, or an empty sequence.
inclusive-event-states	A sequence of event states, as symbols, for including events to delete, or an empty sequence.
exclusive-event-states	A sequence of event states, as symbols, for excluding events to delete, or an empty sequence.

Return Value	Description
events	A sequence of selected events.

Description

Returns a sequence of events created outside of a specified time period and related to a target. You express timestamps in G2 time.

Specify *event-type* to get only events of the specified class, or any subclass. To get all types of events, specify gevm-event as the event type.

Specify *category-filter* to get only events in the specified categories or an empty sequence to get events in all categories.

Specify *inclusive-event-states* to get only events whose history includes any of the specified event states. Specify *exclusive-event-states* to get only events whose history does not include any of the specified event states.

item::gevm-get-collected-events-created-within-time-period-for-target

Synopsis

item::gevm-get-collected-events-created-outside-time-period-for-target
(*target*: item, *start-timestamp*: quantity, *end-timestamp*: quantity,
event-type: symbol, *category-filter*: sequence,
inclusive-event-states: sequence, *exclusive-event-states*: sequence)
–> *events*: sequence

Argument	Description
target	The target item for an event.
start-timestamp	The start time for considering events for selection.
end-timestamp	The end time for considering events for selection.
event-type	The event class for filtering events.
category-filter	A sequence of categories, as symbols, for filtering events, or an empty sequence.
inclusive-event-states	A sequence of event states, as symbols, for including events to delete, or an empty sequence.
exclusive-event-states	A sequence of event states, as symbols, for excluding events to delete, or an empty sequence.

Return Value	Description
events	A sequence of selected events.

Description

Returns a sequence of events created within a specified time period and related to a target. You express timestamps in G2 time.

Specify *event-type* to get only events of the specified class, or any subclass. To get all types of events, specify gevm-event as the event type.

Specify *category-filter* to get only events in the specified categories or an empty sequence to get events in all categories.

Specify *inclusive-event-states* to get only events whose history includes any of the specified event states. Specify *exclusive-event-states* to get only events whose history does not include any of the specified event states.

item::gevm-get-collected-events-updated-outside-time-period-for-target

Synopsis

item::gevm-get-collected-events-updated-outside-time-period-for-target
(*target*: item, *start-timestamp*: quantity, *end-timestamp*: quantity,
event-type: symbol, *category-filter*: sequence,
inclusive-event-states: sequence, *exclusive-event-states*: sequence)
–> *events*: sequence

Argument	Description
target	The target item for an event.
start-timestamp	The start time for considering events for selection.
end-timestamp	The end time for considering events for selection.
event-type	The event class for filtering events.
category-filter	A sequence of categories, as symbols, for filtering events, or an empty sequence.
inclusive-event-states	A sequence of event states, as symbols, for including events to delete, or an empty sequence.
exclusive-event-states	A sequence of event states, as symbols, for excluding events to delete, or an empty sequence.

Return Value	Description
events	A sequence of selected events.

Description

Returns a sequence of events updated outside of a specified time period and related to a target. Event updates include attribute value changes, such as message text, priority, repetition, and acknowledgement status. When the event is updated, the gevm-last-update-timestamp attribute is updated and compared against the start and end timestamps. You express timestamps in G2 time.

Specify *event-type* to get only events of the specified class, or any subclass. To get all types of events, specify gevm-event as the event type.

Specify *category-filter* to get only events in the specified categories or an empty sequence to get events in all categories.

Specify *inclusive-event-states* to get only events whose history includes any of the specified event states. Specify *exclusive-event-states* to get only events whose history does not include any of the specified event states.

item::gevm-get-collected-events-updated-within-time-period-for-target

Synopsis

item::gevm-get-collected-events-updated-outside-time-period-for-target
(*target*: item, *start-timestamp*: quantity, *end-timestamp*: quantity,
event-type: symbol, *category-filter*: sequence,
inclusive-event-states: sequence, *exclusive-event-states*: sequence)
–> *events*: sequence

Argument	Description
target	The target item for an event.
start-timestamp	The start time for considering events for selection.
end-timestamp	The end time for considering events for selection.
event-type	The event class for filtering events.
category-filter	A sequence of categories, as symbols, for filtering events, or an empty sequence.
inclusive-event-states	A sequence of event states, as symbols, for including events to delete, or an empty sequence.
exclusive-event-states	A sequence of event states, as symbols, for excluding events to delete, or an empty sequence.

Return Value	Description
events	A sequence of selected events.

Description

Returns a sequence of events updated within of a specified time period and related to a target. Event updates include attribute value changes, such as message text, priority, repetition, and acknowledgement status. When the event is updated, the gevm-last-update-timestamp attribute is updated and compared against the start and end timestamps. You express timestamps in G2 time.

Specify *event-type* to get only events of the specified class, or any subclass. To get all types of events, specify gevm-event as the event type.

Specify *category-filter* to get only events in the specified categories or an empty sequence to get events in all categories.

Specify *inclusive-event-states* to get only events whose history includes any of the specified event states. Specify *exclusive-event-states* to get only events whose history does not include any of the specified event states.

item::gevm-get-event-count-for-target

Synopsis

item::gevm-get-event-count-for-target
(*target*: item, *event-type*: symbol, *category-filter*: sequence)
–> *count*: integer

Argument	Description
target	The target item for an event.
event-type	The event class for filtering events.
category-filter	A sequence of categories, as symbols, for filtering events, or an empty sequence.

Return Value	Description
count	The number of events that meet the criteria.

Description

Returns the count of events of a certain type and category related to a target.

Specify *event-type* to include only events of the specified class, or any subclass. To include all types of events, specify gevm-event as the event type.

Specify *category-filter* to include only events in the specified categories or an empty sequence to include events in all categories.

item::gevm-get-event-count-for-target

Synopsis

item::gevm-get-event-count-for-target
(*target*: item)
–> *count*: integer

Argument	Description
target	The target item for an event.

Return Value	Description
count	The number of events that meet the criteria.

Description

Returns the count of all events related to a target.

item::gevm-get-events-for-target

Synopsis

item::gevm-get-events-for-target
(*target*: item)
–> *events*: sequence

Argument	Description
target	The target item for an event.

Return Value	Description
events	A sequence of selected events.

Description

Returns all events related to a target.

item::gevm-get-events-for-target

Synopsis

item::gevm-get-events-for-target
(*target*: item, *event-type*: symbol, *category-filter*: sequence)
–> *events*: sequence

Argument	Description
target	The target item for an event.
event-type	The event class for filtering events.
category-filter	A sequence of categories, as symbols, for filtering events, or an empty sequence.

Return Value	Description
events	A sequence of selected events.

Description

Returns a sequence of events of a certain type and category related to a target.

Specify *event-type* to acknowledge only events of the specified class, or any subclass. To acknowledge all types of events, specify gevm-event as the event type.

Specify *category-filter* to acknowledge only events in the specified categories or an empty sequence to acknowledge events in all categories.

item::gevm-get-events-for-target

Synopsis

item::gevm-get-events-for-target
(*target*: item, *event-type*: symbol)
–> *events*: sequence

Argument	Description
target	The target item for an event.
event-type	The event class for filtering events.

Return Value	Description
events	A sequence of selected events.

Description

Returns a sequence of events of a certain type related to a target.

Specify *event-type* to acknowledge only events of the specified class, or any subclass. To acknowledge all types of events, specify gevm-event as the event type.

item::gevm-remove-all-events-for-target

Synopsis

item::gevm-remove-all-events-for-target
(*target*: item, *client*: ui-client-item)

Argument	Description
target	The target item for an event.

Description

Removes all events related to a target.

item::gevm-remove-events-for-target

Synopsis

item::gevm-remove-events-for-target
(*target*: item, *event-types*: sequence)

Argument	Description
target	The target item for an event.
event-types	A sequence of events to remove.

Description

Removes events from a queue.

Initiating Event Operations

These API procedures allow you to perform a variety of tasks, given the initiating item of an event, which must be a gevm-event-initiating-item. You can:

- Acknowledge events related to an initiator.
- Delete events related to an initiator.
- Delete events related to an initiator that occur within or outside of a given time period.
- Return a list of events related to an initiator that are created or are updated within or outside of a given time period.
- Return the number of events related to an initiator.
- Return a list of events related to an initiator.
- Remove events related to an initiator.

The APIs provide different versions that allow you to perform the action on:

- All events related to an initiator.
- A subset of events that are filtered, based on the event type, category, and/or event states.

The initiating event operations are:

item::gevm-acknowledge-all-events-for-initiator
item::gevm-acknowledge-events-for-initiator
item::gevm-acknowledge-events-for-initiator
item::gevm-delete-all-events-for-initiator
item::gevm-delete-collected-events-created-outside-time-period-for-initiator
item::gevm-delete-collected-events-created-within-time-period-for-initiator
item::gevm-delete-events-for-initiator
item::gevm-delete-events-for-initiator
item::gevm-get-collected-events-created-outside-time-period-for-initiator
item::gevm-get-collected-events-created-within-time-period-for-initiator
item::gevm-get-collected-events-updated-outside-time-period-for-initiator
item::gevm-get-collected-events-updated-within-time-period-for-initiator
item::gevm-get-event-count-for-initiator
item::gevm-get-event-count-for-initiator
item::gevm-get-events-for-initiator
item::gevm-get-events-for-initiator
item::gevm-get-events-for-initiator
item::gevm-remove-all-events-for-initiator
item::gevm-remove-events-for-initiator

item::gevm-acknowledge-all-events-for-initiator

Synopsis

item::gevm-acknowledge-all-events-for-initiator
(*initiator*: item, *client*: ui-client-item)

Argument	Description
initiator	The initiating item for an event.
client	The client, such as a G2 window.

Description

Acknowledges all events related to an initiator.

item::gevm-acknowledge-events-for-initiator

Synopsis

item::gevm-acknowledge-events-for-initiator
(*initiator*: item, *event-type*: symbol, *category-filter*: sequence,
client: ui-client-item)

Argument	Description
initiator	The initiating item for an event.
event-type	The event class for filtering events.
category-filter	A sequence of categories, as symbols, for filtering events, or an empty sequence.
client	The client, such as a G2 window.

Description

Acknowledges events of a certain type and category related to an initiator.

Specify *event-type* to acknowledge only events of the specified class, or any subclass. To acknowledge all types of events, specify gevm-event as the event type.

Specify *category-filter* to acknowledge only events in the specified categories or an empty sequence to acknowledge events in all categories.

item::gevm-acknowledge-events-for-initiator

Synopsis

item::gevm-acknowledge-events-for-initiator
(*initiator*: item, *event-types*: sequence, *client*: ui-client-item)

Argument	Description
initiator	The initiating item for an event.
event-types	A sequence of event classes for filtering events.
client	The client, such as a G2 window.

Description

Acknowledges events of a set of types related to an initiator, or any subclass.

item::gevm-delete-all-events-for-initiator

Synopsis

item::gevm-delete-all-events-for-initiator
(*initiator*: item)

Argument	Description
initiator	The initiating item for an event.

Description

Deletes all events related to an initiator.

item::gevm-delete-collected-events-created-outside-time-period-for-initiator

Synopsis

item::gevm-delete-collected-events-created-outside-time-period-for-initiator
(*initiator*: item, *start-timestamp*: quantity, *end-timestamp*: quantity,
event-type: symbol, *category-filter*: sequence,
inclusive-event-states: sequence, *exclusive-event-states*: sequence)

Argument	Description
initiator	The initiating item for an event.
start-timestamp	The start time for considering events for deletion.
end-timestamp	The end time for considering events for deletion.
event-type	The event class for filtering events.
category-filter	A sequence of categories, as symbols, for filtering events, or an empty sequence.
inclusive-event-states	A sequence of event states, as symbols, for including events to delete, or an empty sequence.
exclusive-event-states	A sequence of event states, as symbols, for excluding events to delete, or an empty sequence.

Description

Deletes events created outside of a specified time period and related to an initiator. You express timestamps in G2 time.

Specify *event-type* to delete only events of the specified class, or any subclass. To delete all types of events, specify gevm-event as the event type.

Specify *category-filter* to delete only events in the specified categories or an empty sequence to delete events in all categories.

Specify *inclusive-event-states* to delete only events whose history includes any of the specified event states. Specify *exclusive-event-states* to delete only events whose history does not include any of the specified event states.

item::gevm-delete-collected-events-created-within-time-period-for-initiator

Synopsis

item::gevm-delete-collected-events-created-within-time-period-for-initiator
(*initiator*: item, *start-timestamp*: quantity, *end-timestamp*: quantity,
event-type: symbol, *category-filter*: sequence,
inclusive-event-states: sequence, *exclusive-event-states*: sequence)

Argument	Description
initiator	The initiating item for an event.
start-timestamp	The start time for considering events for deletion.
end-timestamp	The end time for considering events for deletion.
event-type	The event class for filtering events.
category-filter	A sequence of categories, as symbols, for filtering events, or an empty sequence.
inclusive-event-states	A sequence of event states, as symbols, for including events to delete, or an empty sequence.
exclusive-event-states	A sequence of event states, as symbols, for excluding events to delete, or an empty sequence.

Description

Deletes events created within of a specified time period and related to an initiator. You express timestamps in G2 time.

Specify *event-type* to delete only events of the specified class, or any subclass. To delete all types of events, specify gevm-event as the event type.

Specify *category-filter* to delete only events in the specified categories or an empty sequence to delete events in all categories.

Specify *inclusive-event-states* to delete only events whose history includes any of the specified event states. Specify *exclusive-event-states* to delete only events whose history does not include any of the specified event states.

item::gevm-delete-events-for-initiator

Synopsis

item::gevm-delete-events-for-initiator
(*initiator*: item, *event-type*: symbol, *category-filter*: sequence)

Argument	Description
initiator	The target item for an event.
event-type	The event class for filtering events.
category-filter	A sequence of categories, as symbols, for filtering events, or an empty sequence.

Description

Deletes events of a certain type and category related to an initiator.

Specify *event-type* to delete only events of the specified class, or any subclass. To delete all types of events, specify gevm-event as the event type.

Specify *category-filter* to delete only events in the specified categories or an empty sequence to delete events in all categories.

item::gevm-delete-events-for-initiator

Synopsis

item::gevm-delete-events-for-initiator
(*initiator*: item, *event-types*: sequence)

Argument	Description
initiator	The target item for an event.
event-types	A sequence of event classes for filtering events.

Description

Deletes events of a set of types related to an initiator, or any subclass.

item::gevm-get-collected-events-created-outside-time-period-for-initiator

Synopsis

item::gevm-get-collected-events-created-outside-time-period-for-initiator
(*initiator*: item, *start-timestamp*: quantity, *end-timestamp*: quantity,
event-type: symbol, *category-filter*: sequence,
inclusive-event-states: sequence, *exclusive-event-states*: sequence)
–> *events*: sequence

Argument	Description
initiator	The target item for an event.
start-timestamp	The start time for considering events for selection.
end-timestamp	The end time for considering events for selection.
event-type	The event class for filtering events.
category-filter	A sequence of categories, as symbols, for filtering events, or an empty sequence.
inclusive-event-states	A sequence of event states, as symbols, for including events to delete, or an empty sequence.
exclusive-event-states	A sequence of event states, as symbols, for excluding events to delete, or an empty sequence.

Return Value	Description
events	A sequence of selected events.

Description

Returns a sequence of events created outside of a specified time period and related to an initiator. You express timestamps in G2 time.

Specify *event-type* to get only events of the specified class, or any subclass. To get all types of events, specify gevm-event as the event type.

Specify *category-filter* to get only events in the specified categories or an empty sequence to get events in all categories.

Specify *inclusive-event-states* to get only events whose history includes any of the specified event states. Specify *exclusive-event-states* to get only events whose history does not include any of the specified event states.

item::gevm-get-collected-events-created-within-time-period-for-initiator

Synopsis

item::gevm-get-collected-events-created-outside-time-period-for-initiator
(*initiator*: item, *start-timestamp*: quantity, *end-timestamp*: quantity,
event-type: symbol, *category-filter*: sequence,
inclusive-event-states: sequence, *exclusive-event-states*: sequence)
–> *events*: sequence

Argument	Description
initiator	The target item for an event.
start-timestamp	The start time for considering events for selection.
end-timestamp	The end time for considering events for selection.
event-type	The event class for filtering events.
category-filter	A sequence of categories, as symbols, for filtering events, or an empty sequence.
inclusive-event-states	A sequence of event states, as symbols, for including events to delete, or an empty sequence.
exclusive-event-states	A sequence of event states, as symbols, for excluding events to delete, or an empty sequence.

Return Value	Description
events	A sequence of selected events.

Description

Returns a sequence of events created within a specified time period and related to an initiator. You express timestamps in G2 time.

Specify *event-type* to get only events of the specified class, or any subclass. To get all types of events, specify gevm-event as the event type.

Specify *category-filter* to get only events in the specified categories or an empty sequence to get events in all categories.

Specify *inclusive-event-states* to get only events whose history includes any of the specified event states. Specify *exclusive-event-states* to get only events whose history does not include any of the specified event states.

item::gevm-get-collected-events-updated-outside-time-period-for-initiator

Synopsis

item::gevm-get-collected-events-updated-outside-time-period-for-initiator
(*initiator*: item, *start-timestamp*: quantity, *end-timestamp*: quantity,
event-type: symbol, *category-filter*: sequence,
inclusive-event-states: sequence, *exclusive-event-states*: sequence)
–> *events*: sequence

Argument	Description
initiator	The target item for an event.
start-timestamp	The start time for considering events for selection.
end-timestamp	The end time for considering events for selection.
event-type	The event class for filtering events.
category-filter	A sequence of categories, as symbols, for filtering events, or an empty sequence.
inclusive-event-states	A sequence of event states, as symbols, for including events to delete, or an empty sequence.
exclusive-event-states	A sequence of event states, as symbols, for excluding events to delete, or an empty sequence.

Return Value	Description
events	A sequence of selected events.

Description

Returns a sequence of events updated outside of a specified time period and related to an initiator. You express timestamps in G2 time.

Specify *event-type* to get only events of the specified class, or any subclass. To get all types of events, specify gevm-event as the event type.

Specify *category-filter* to get only events in the specified categories or an empty sequence to get events in all categories.

Specify *inclusive-event-states* to get only events whose history includes any of the specified event states. Specify *exclusive-event-states* to get only events whose history does not include any of the specified event states.

item::gevm-get-collected-events-updated-within-time-period-for-initiator

Synopsis

item::gevm-get-collected-events-updated-outside-time-period-for-initiator
(*initiator*: item, *start-timestamp*: quantity, *end-timestamp*: quantity,
event-type: symbol, *category-filter*: sequence,
inclusive-event-states: sequence, *exclusive-event-states*: sequence)
–> *events*: sequence

Argument	Description
initiator	The target item for an event.
start-timestamp	The start time for considering events for selection.
end-timestamp	The end time for considering events for selection.
event-type	The event class for filtering events.
category-filter	A sequence of categories, as symbols, for filtering events, or an empty sequence.
inclusive-event-states	A sequence of event states, as symbols, for including events to delete, or an empty sequence.
exclusive-event-states	A sequence of event states, as symbols, for excluding events to delete, or an empty sequence.

Return Value	Description
events	A sequence of selected events.

Description

Returns a sequence of events updated within a specified time period and related to an initiator. You express timestamps in G2 time.

Specify *event-type* to get only events of the specified class, or any subclass. To get all types of events, specify gevm-event as the event type.

Specify *category-filter* to get only events in the specified categories or an empty sequence to get events in all categories.

Specify *inclusive-event-states* to get only events whose history includes any of the specified event states. Specify *exclusive-event-states* to get only events whose history does not include any of the specified event states.

item::gevm-get-event-count-for-initiator

Synopsis

item::gevm-get-event-count-for-initiator
(*initiator*: item, *event-type*: symbol, *category-filter*: sequence)
–> *count*: integer

Argument	Description
initiator	The target item for an event.
event-type	The event class for filtering events.
category-filter	A sequence of categories, as symbols, for filtering events, or an empty sequence.

Return Value	Description
count	The number of events that meet the criteria.

Description

Returns the count of events of a certain type and category related to an initiator.

Specify *event-type* to include only events of the specified class, or any subclass. To include all types of events, specify gevm-event as the event type.

Specify *category-filter* to include only events in the specified categories or an empty sequence to include events in all categories.

item::gevm-get-event-count-for-initiator

Synopsis

item::gevm-get-event-count-for-initiator
(*initiator*: item)
–> *count*: integer

Argument	Description
initiator	The target item for an event.

Return Value	Description
count	The number of events that meet the criteria.

Description

Returns the count of events related to an initiator.

item::gevm-get-events-for-initiator

Synopsis

item::gevm-get-events-for-initiator
(*initiator*: item)
–> *count*: integer

Argument	Description
initiator	The target item for an event.

Return Value	Description
count	The number of events that meet the criteria.

Description

Returns all events related to an initiator.

item::gevm-get-events-for-initiator

Synopsis

item::gevm-get-events-for-initiator
(*initiator*: item, *event-type*: symbol, *category-filter*: sequence)
–> *events*: sequence

Argument	Description
initiator	The target item for an event.
event-type	The event class for filtering events.
category-filter	A sequence of categories, as symbols, for filtering events, or an empty sequence.

Return Value	Description
events	A sequence of selected events.

Description

Returns a sequence of events of a certain type and category related to an initiator.

Specify *event-type* to acknowledge only events of the specified class, or any subclass. To acknowledge all types of events, specify gevm-event as the event type.

Specify *category-filter* to acknowledge only events in the specified categories or an empty sequence to acknowledge events in all categories.

item::gevm-get-events-for-initiator

Synopsis

item::gevm-get-events-for-initiator
(*initiator*: item, *event-type*: symbol)
–> *events*: sequence

Argument	Description
initiator	The target item for an event.
event-type	The event class for filtering events.

Return Value	Description
events	A sequence of selected events.

Description

Returns a sequence of events of a certain type and category related to an initiator.

Specify *event-type* to acknowledge only events of the specified class, or any subclass. To acknowledge all types of events, specify gevm-event as the event type.

item::gevm-remove-all-events-for-initiator

Synopsis

item::gevm-remove-all-events-for-initiator
(*initiator*: item, *client*: ui-client-item)

Argument	Description
initiator	The target item for an event.

Description

Removes all events related to an initiator.

item::gevm-remove-events-for-initiator

Synopsis

item::gevm-remove-events-for-initiator
(*initiator*: item, *event-types*: sequence)

Argument	Description
initiator	The target item for an event.
event-types	A sequence of event classes for filtering events.

Description

Removes events of a set of types related to an initiator, or any subclass.

Escalation Operations

This chapter describes the API methods and procedures for auto escalation of gevm-event objects in a message browser. The API lets you:

- Enable and disable event escalation.
- Set and disable the reevaluation period of events.
- Get, set, and clear reevaluation schemes associated with an event.

The escalation operations are:

gevm-event::gevm-clear-reevaluation-scheme
gevm-event::gevm-get-reevaluation-scheme
gevm-event::gevm-set-reevaluation-scheme
gevm-disable-event-escalation
gevm-disable-reevaluate-period
gevm-enable-event-escalation
gevm-set-reevaluate-period

gevm-event::gevm-clear-reevaluation-scheme

Synopsis

gevm-event::gevm-clear-reevaluation-scheme
(*event*: gevm-event)

Argument	Description
event	The event whose escalation scheme to clear.

Description

Clears any evaluation scheme associated with an event.

gevm-event::gevm-get-reevaluation-scheme

Synopsis

gevm-event::gevm-get-reevaluation-scheme
(*event*: gevm-event)
–> *scheme*: item-or-value

Argument	Description
event	The event whose escalation scheme to get.

Return Value	Description
scheme	The scheme associated with an event, which is an instance of a grtl-scheme.

Description

Returns the escalation scheme associated with an event, or the symbol none if no scheme exists.

For more information on schemes, see grtl-scheme in the *G2 Run-Time Library User's Guide*.

gevm-event::gevm-set-reevaluation-scheme

Synopsis

gevm-event::gevm-set-reevaluation-scheme
(*event*: gevm-event, *scheme*: grtl-scheme)

Argument	Description
event	The event whose escalation scheme to set.
scheme	The scheme to associate with an event, which is an instance of a grtl-scheme.

Description

Associates a scheme with an event, which evaluates the event on a periodic basis.

For more information on schemes, see grtl-scheme in the *G2 Run-Time Library User's Guide*.

gevm-disable-event-escalation

Synopsis

gevm-disable-event-escalation
()

Description

Disables event escalation for the KB.

gevm-disable-reevaluate-period

Synopsis

gevm-disable-reevaluate-period
    ()

Description

Disables the currently specified reevaluation period for event escalation so the KB uses the default reevaluation period.

gevm-enable-event-escalation

Synopsis

gevm-enable-event-escalation
    ()

Description

Enables event escalation for the KB.

gevm-set-reevaluate-period

Synopsis

gevm-set-reevaluate-period
(*interval*: quantity)

Argument	Description
interval	The time period for reevaluating events for event escalation.

Description

Sets the time interval for reevaluating messages for message escalation.

Customization

You can customize these aspects of events:

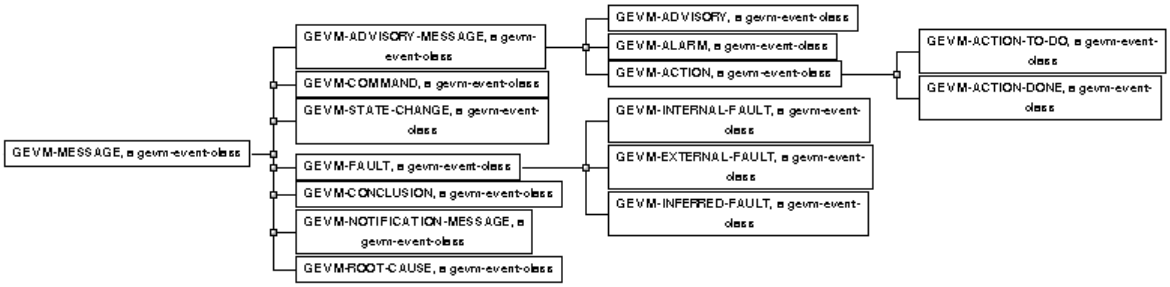
- Event type.
- Message correlation.

Creating Custom Message Classes

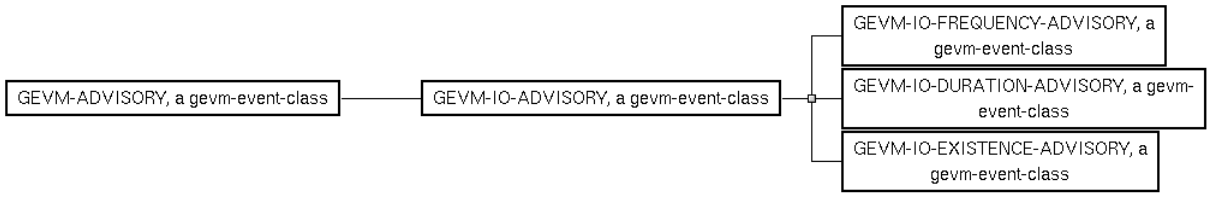
You can create custom message classes by inheriting from the gevm-message class or any of its subclasses, depending on the type of message you want to create. Remember that the class definition for each message class needs to be an instance of a gevm-event-class instead of a G2 class definition.

Note Your application filters and subsumes messages, based on the class hierarchy, so be sure to choose the superior class at the correct level in the class hierarchy.

Here is the class hierarchy for the gevm-message class for basic messaging:



Here is the class hierarchy for gevm-io-advisory, the superior class of all Intelligent Object message types:



Here is the class hierarchy for gevm-cdg-event, the superior class of all SymCure message types:

Customizing Message Correlation

You can customize how your application correlates messages, based on message type. To do this, you must implement your own custom method, then override the message-subordination-procedure attribute of the gevm-module-settings object to refer to your custom method.

Caution Be sure to save the GEVM module settings object to a higher-level module.

The default message subordination method is gevm-subordinate-messages, which you access on the Programmer's Interface workspace of the gevm-top-level workspace.

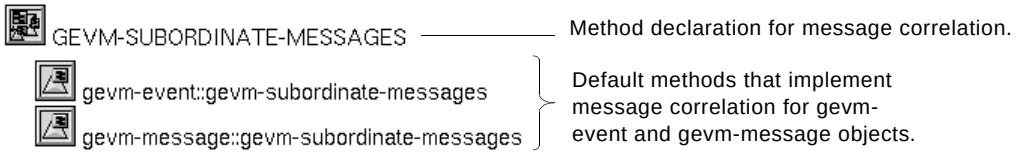
The default method implementation is for the gevm-message class. You can also provide an implementation of this method for the gevm-event class, which has no implementation, by default.

GEVM defines four message relations used for correlation purposes: an-information-provider-for, a-possible-cause-for, a-possible-action-to-do-for, an-action-for, or an-action-done-for.

To customize message correlation:

1. Create a method declaration and method to implement your custom message correlation logic for the gevm-message class.
2. On the GEVM Programmer's Interface workspace, display the text of the gevm-message::gevm-subordinate-messages method and copy the method text to your custom method.
3. Edit the text of your custom method, as needed.
4. On the GEVM Settings workspace, clone the gevm-module-settings object and place it in a higher-level module in your module hierarchy.
5. Edit the message-subordination-procedure of your custom GEVM settings object to refer to your custom method.

Here is the default gevm-subordinate-messages method declaration and the default methods for the gevm-message and gevm-event classes:



5

Queues

Describes event queue configuration and notification, and the classes and operations for creating and managing event queues.

Introduction **267**

Queue Configuration **269**

Event Notification **271**

GEVM Queue Class and Operations **272**

GQS Queue Class and Operations **323**

Customizations **348**

Introduction

This chapter describes the gevm-gqs-queue class and its superior class, gqs-queue, which provides queues for managing events.

The API provides methods that:

- Add events to a queue.
- Acknowledge events in a queue.
- Remove events from a queue.
- Delete all events from a queue.
- Delete a collection of events that are created or updated within or outside of a given time period, and optionally have a given target and initiator.

- Return a collection of events that are created or updated within or outside of a given time period, and optionally have a given target and initiator.
- Return the number of events in a queue.
- Initialize or clear a queue.
- Add a given list of events to a queue.

The methods provide different versions that allow you to perform the action on:

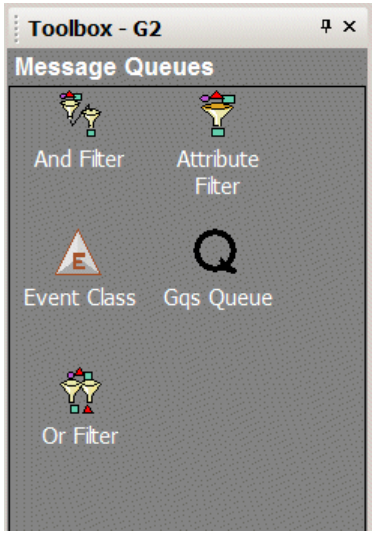
- All events related to a queue.
- A subset of events that are filtered, based on the event type, category, and/or event states.

In addition, the API provides a number of procedure and functions that perform a variety of tasks, such as getting message properties and property values of events.

Queue Configuration

To configure a queue:

1 In Developer mode, choose View > Toolbox - G2 and display the Message Queues palette:

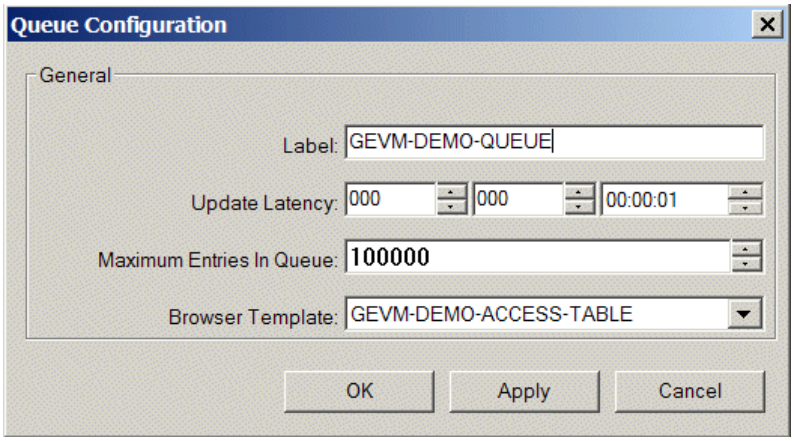


2 Create a Gqs Queue from the Message Queues palette.

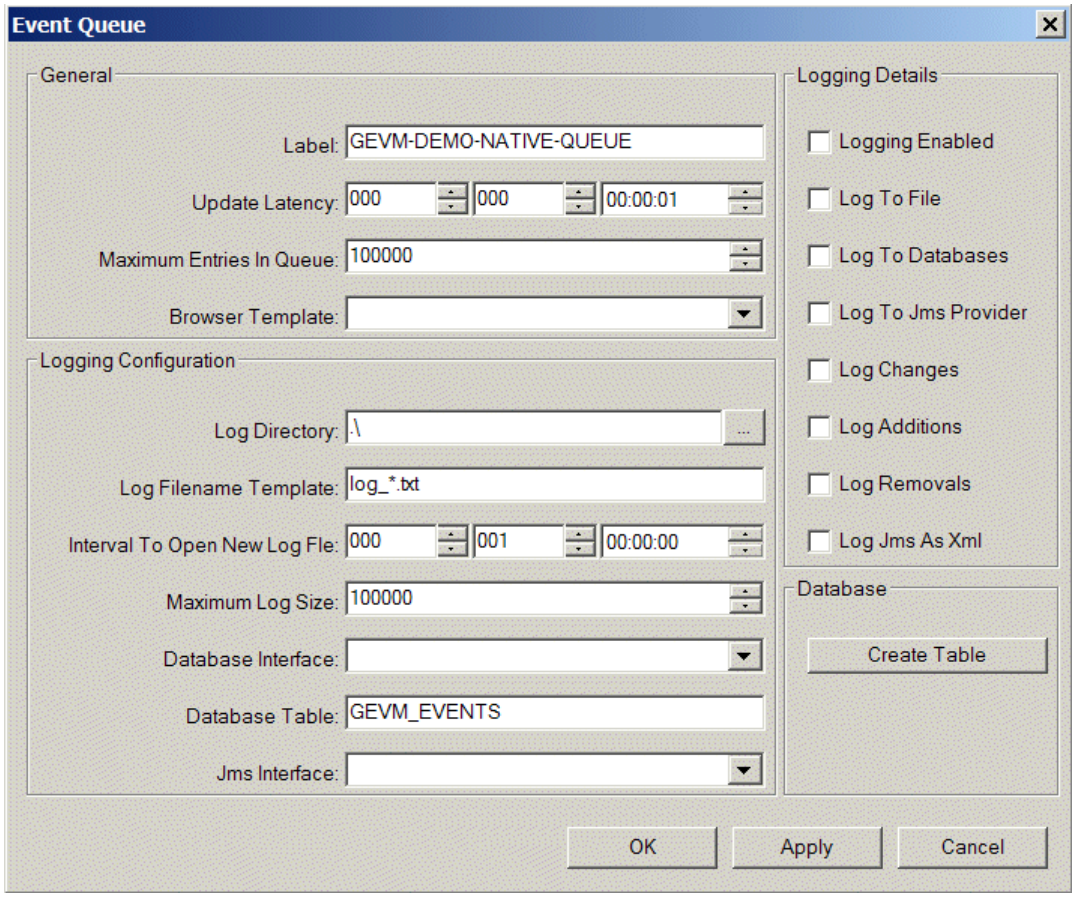
3 Display the properties dialog for the Message Queue and configure its properties.

You configure the Browser Template to specify the access table to use for choosing the tabular view template. If you create only one view manager template for a single user mode, you do not need to create a Queue Access Table or configure the Browser Template of the message queue.

Here is the properties dialog for a message queue in Modeler mode, which uses a queue access table:



Here is the properties dialog in System-Administrator mode:



For a description of these properties, see gevm-gqs-queue on page 274.

Event Notification

The table below summarizes the event types related to a gevm-gqs-queue. For a description of event notification, see the *G2 Run-Time Library User's Guide*.

Event Type	Event Arguments (Examples)	Description
gevm-queue-entry-notification	structure (event-type: the symbol gevm-queue-entry-notification, gevm-event-key: "*key*", gevm-event-type: *symbol*) where gevm-event-type is one of these symbols: • gevm-acknowledged-event • gevm-deleting-event • gevm-subordinating-event • gevm-publish-event • gevm-event-priority-increase • gevm-event-priority-decrease • gevm-event-repetition	Sent by an event queue to redispatch events received from an event.
gevm-queue-entry-additions	structure (event-type: the symbol gevm-queue-entry-additions, gevm-event-keys: *key-sequence*)	Sent when an entry has been added to a queue.
gevm-queue-entry-deletion	structure (event-type: the symbol gevm-queue-entry-deletion, gevm-event-key: "*key*")	Sent when an entry has been deleted from a queue.
gevm-queue-entry-removals	structure (event-type: the symbol gevm-queue-entry-removals, gevm-event-keys: *key-sequence*)	Sent when an entry has been removed a queue.
gevm-queue-entry-attribute-change	structure (event-type: the symbol gevm-queue-entry-attribute-change, gevm-event-key: "*key*", gevm-event-attribute: *symbol*)	Sent when the attribute of an entry has changed.

GEVM Queue Class and Operations

Class

gevm-gqs-queue on page 274

Methods

gevm-gqs-queue::gevm-add-event-to-queue on page 279
gevm-gqs-queue::grtl-initialize on page 280
gevm-get-most-recent-unacknowledged-event on page 313
gevm-gqs-queue::gevm-acknowledge-all-events on page 281
gevm-gqs-queue::gevm-acknowledge-events on page 282
gevm-gqs-queue::gevm-delete-all-events on page 283
gevm-gqs-queue::gevm-delete-collected-events-created-outside-time-period on page 284
gevm-gqs-queue::gevm-delete-collected-events-created-outside-time-period-for-initiator-and-target on page 285
gevm-gqs-queue::gevm-delete-collected-events-created-within-time-period on page 287
gevm-gqs-queue::gevm-delete-collected-events-created-within-time-period-for-initiator-and-target on page 288
gevm-gqs-queue::gevm-delete-events on page 290
gevm-gqs-queue::gevm-get-collected-events on page 291
gevm-gqs-queue::gevm-get-collected-events-created-outside-time-period on page 292
gevm-gqs-queue::gevm-get-collected-events-created-outside-time-period-for-initiator-and-target on page 294
gevm-gqs-queue::gevm-get-collected-events-created-within-time-period on page 296
gevm-gqs-queue::gevm-get-collected-events-created-within-time-period-for-initiator-and-target on page 298
gevm-gqs-queue::gevm-get-collected-events-for-initiator-and-target on page 300
gevm-gqs-queue::gevm-get-collected-events-updated-outside-time-period on page 301
gevm-gqs-queue::gevm-get-collected-events-updated-outside-time-period-for-initiator-and-target on page 302
gevm-gqs-queue::gevm-get-collected-events-updated-within-time-period on page 304
gevm-gqs-queue::gevm-get-collected-events-updated-within-time-period-for-initiator-and-target on page 305
gevm-find-queue-by-key on page 310
gevm-get-all-queues on page 311
gevm-get-message-properties on page 312
gevm-get-property-values-of-all-events on page 314
gevm-get-property-values-of-events on page 316
gevm-get-queues-containing-item on page 318

gevm-get-selected-property-values-of-all-events on page 319
gevm-get-selected-property-values-of-events on page 321

Procedures

gevm-find-queue-by-key on page 310
gevm-get-all-queues on page 311
gevm-get-message-properties on page 312
gevm-get-most-recent-unacknowledged-event on page 313
gevm-get-property-values-of-all-events on page 314
gevm-get-property-values-of-events on page 316
gevm-get-queues-containing-item on page 318
gevm-get-selected-property-values-of-all-events on page 319
gevm-get-selected-property-values-of-events on page 321

gevm-gqs-queue

A GQS queue that manages the number of events and logs events to CSV files and databases.

Class Inheritance Path

gevm-gqs-queue, gqs-queue, gfr-object-with-uuid, glf-logging-manager, object, gfr-item-with-uuid, grtl-event-source, grtl-item-with-key, grtl-item, item

Attributes

Attribute	Description
gevm-max-events-to-queue	The maximum number of events that can be in the queue. When this limit is reached, events are deleted, starting with the oldest events.
Allowable values:	Any integer
Default value:	100000
gevm-log-to-file	True when logging to a CSV file.
Allowable values:	Any truth-value
Default value:	true
gevm-log-to-database	True when logging to a database.
Allowable values:	Any truth-value
Default value:	false
gevm-log-database-interface-or-pool	Either the key of a G2 database interface or the key of a GDSM database pool. If a pool is selected, it uses the least-used database interface in the pool each time it inserts rows in the database. This helps increase throughput of applications that require many interactions with the database.
Allowable values:	Any text
Default value:	""
gevm-log-database-table	The name of the database table in which to add entries.

Allowable values: Any symbol

Default value: gevm_events

gevm-log-additions If true, adds an entry to the log file and/or database when a raw event or message is added to the queue.

Allowable values: Any truth-value

Default value: true

gevm-log-removals If true, adds an entry to the log file and/or database when a raw event or message is removed to the queue.

Allowable values: Any truth-value

Default value: true

gevm-log-state-changes If true, adds an entry to the log file and/or database when the state of a raw event or message changes, for example, when its priority changes, the repetition count changes, or the message is subordinated.

Allowable values: Any truth-value

Default value: false

gevm-log-procedure The procedure called to log a new entry to the CSV file and/or database.

Allowable values: Any symbol

Default value: gevm-log-events

gqs-initially-monitor-for-deletion-events See gqs-queue on page 324.

Allowable values: inherited

Default value: false

gqs-initially-monitor-for-attribute-change-events See gqs-queue on page 324.

Allowable values: inherited

Default value: false

gqs-item-deletion-callback See gqs-queue on page 324.

Allowable values: inherited

Default value: gevm-queue-item-deletion

gqs-attribute-update-callback See gqs-queue on page 324.

Allowable values: inherited

Default value: gevm-queue-attribute-update

gqs-item-addition-callback See gqs-queue on page 324.

Allowable values: inherited

Default value: gevm-queue-item-addition

gqs-item-removal-callback See gqs-queue on page 324.

Allowable values: inherited

Default value: gevm-queue-item-removal

gqs-view-template-or-access-table See gqs-queue on page 324.

Allowable values: inherited

Default value: guif-messages-view-access-table

gqs-logging-manager See gqs-queue on page 324.

Allowable values: inherited

Default value: unspecified

gqs-update-latency See gqs-queue on page 324.

Allowable values: inherited

Default value: 1.0

glf-logging-enabled See glf-logging-manager on page 472.

Allowable values: inherited

Default value: false

glf-log-directory See glf-logging-manager on page 472.

Allowable values: inherited

Default value: ""

glf-log-file-name-template See glf-logging-manager on page 472.

Allowable values: inherited

Default value: "log_*.txt"

glf-log-file-name-generator See glf-logging-manager on page 472.

Allowable values: inherited

Default value: gevm-archive-file-name-generator

glf-current-log-file See glf-logging-manager on page 472.

Allowable values: inherited

Default value: ""

glf-file-header-writer See glf-logging-manager on page 472.

Allowable values: inherited

Default value: gevm-log-file-header-writer

glf-time-interval-to-open-new-log-file See glf-logging-manager on page 472.

Allowable values: inherited

Default value: 86400

glf-maximum-file-size-in-bytes See glf-logging-manager on page 472.

Allowable values: inherited

Default value: 100000

glf-log-file-scheduler See glf-logging-manager on page 472.

Allowable values: inherited

Default value: glf-default-log-file-scheduler

glf-automatically-delete-empty-log-files See glf-logging-manager on page 472.

Allowable values: inherited

Default value:	true
gfr-uuid	See gfr-item-with-uuid in the *G2 Foundation Resources User's Guide*.
Allowable values:	inherited
Default value:	""
key	See grtl-item-with-key in the *G2 Run-Time Library User's Guide*.
Allowable values:	inherited
Default value:	""
gevm-log-jms-interface-or-pool	The name of JMS interface to which to log messages.
gevm-log-jms-as-xml	Whether to log messages as XML to a JMS provider.

gevm-gqs-queue::gevm-add-event-to-queue

Synopsis

gevm-gqs-queue::gevm-add-event-to-queue
(*queue*: gevm-gqs-queue, *event*: gevm-event)

Argument	Description
queue	The event queue.
event	The event to add to the queue.

Description

Sends an event to a queue.

gevm-gqs-queue::grtl-initialize

Synopsis

gevm-gqs-queue::grtl-initialize
(*queue*: gevm-gqs-queue)

Argument	Description
queue	The queue.

Description

Deletes all events contained in a queue.

gevm-gqs-queue::gevm-acknowledge-all-events

Synopsis

gevm-gqs-queue::gevm-acknowledge-all-events
(*queue*: gevm-gqs-queue, *client*: ui-client-item)

Argument	Description
queue	The queue.
client	The client, such as a G2 window.

Description

Acknowledge all events in a queue.

gevm-gqs-queue::gevm-acknowledge-events

Synopsis

gevm-gqs-queue::gevm-acknowledge-events
(*queue*: gevm-gqs-queue, *events*: sequence, *client*: ui-client-item)

Argument	Description
queue	The queue.
events	A sequence of events to acknowledge.
client	The client, such as a G2 window.

Description

Acknowledges a collection of events in a queue.

gevm-gqs-queue::gevm-delete-all-events

Synopsis

gevm-gqs-queue::gevm-delete-all-events
(*queue*: gevm-gqs-queue)

Argument	Description
queue	The queue.

Description

Deletes all events in a queue.

gevm-gqs-queue::gevm-delete-collected-events-created-outside-time-period

Synopsis

gevm-gqs-queue::gevm-delete-collected-events-created-outside-time-period
(*queue*: gevm-gqs-queue, *start-timestamp*: quantity, *end-timestamp*: quantity, *event-type*: symbol, *category-filter*: sequence, *inclusive-event-states*: sequence, *exclusive-event-states*: sequence)

Argument	Description
queue	The queue.
start-timestamp	The start time for considering events for deletion.
end-timestamp	The end time for considering events for deletion.
event-type	The event class for filtering events.
category-filter	A sequence of categories, as symbols, for filtering events, or an empty sequence.
inclusive-event-states	A sequence of event states, as symbols, for including events to delete, or an empty sequence.
exclusive-event-states	A sequence of event states, as symbols, for excluding events to delete, or an empty sequence.

Description

Deletes a collection of events created outside of a specified time period in a queue. You express timestamps in G2 time.

Specify *event-type* to delete only events of the specified class, or any subclass. To delete all types of events, specify gevm-event as the event type.

Specify *category-filter* to delete only events in the specified categories or an empty sequence to delete events in all categories.

Specify *inclusive-event-states* to delete only events whose history includes any of the specified event states. Specify *exclusive-event-states* to delete only events whose history does not include any of the specified event states.

gevm-gqs-queue::gevm-delete-collected-events-created-outside-time-period-for-initiator-and-target

Synopsis

gevm-gqs-queue::gevm-delete-collected-events-created-outside-time-period-for-initiator-and-target
(*queue*: gevm-gqs-queue, *start-timestamp*: quantity, *end-timestamp*: quantity, *initiator*: item, *target*: item, *event-type*: symbol, *category-filter*: sequence, *inclusive-event-states*: sequence, *exclusive-event-states*: sequence)

Argument	Description
queue	The queue.
start-timestamp	The start time for considering events for deletion.
end-timestamp	The end time for considering events for deletion.
initiator	The initiating item for an event.
target	The target item for an event.
event-type	The event class for filtering events.
category-filter	A sequence of categories, as symbols, for filtering events, or an empty sequence.
inclusive-event-states	A sequence of event states, as symbols, for including events to delete, or an empty sequence.
exclusive-event-states	A sequence of event states, as symbols, for excluding events to delete, or an empty sequence.

Description

Deletes a collection of events created outside of a specified time period in a queue with a given target and initiator. You express timestamps in G2 time.

Specify *event-type* to delete only events of the specified class, or any subclass. To delete all types of events, specify gevm-event as the event type.

Specify *category-filter* to delete only events in the specified categories or an empty sequence to delete events in all categories.

Specify *inclusive-event-states* to delete only events whose history includes any of the specified event states. Specify *exclusive-event-states* to delete only events whose history does not include any of the specified event states.

gevm-gqs-queue::gevm-delete-collected-events-created-within-time-period

Synopsis

gevm-gqs-queue::gevm-delete-collected-events-created-within-time-period
(*queue*: gevm-gqs-queue, *start-timestamp*: quantity, *end-timestamp*: quantity, *event-type*: symbol, *category-filter*: sequence, *inclusive-event-states*: sequence, *exclusive-event-states*: sequence)

Argument	Description
queue	The queue.
start-timestamp	The start time for considering events for deletion.
end-timestamp	The end time for considering events for deletion.
event-type	The event class for filtering events.
category-filter	A sequence of categories, as symbols, for filtering events, or an empty sequence.
inclusive-event-states	A sequence of event states, as symbols, for including events to delete, or an empty sequence.
exclusive-event-states	A sequence of event states, as symbols, for excluding events to delete, or an empty sequence.

Description

Deletes a collection of events created within a specified time period in a queue. You express timestamps in G2 time.

Specify *event-type* to delete only events of the specified class, or any subclass. To delete all types of events, specify gevm-event as the event type.

Specify *category-filter* to delete only events in the specified categories or an empty sequence to delete events in all categories.

Specify *inclusive-event-states* to delete only events whose history includes any of the specified event states. Specify *exclusive-event-states* to delete only events whose history does not include any of the specified event states.

gevm-gqs-queue::gevm-delete-collected-events-created-within-time-period-for-initiator-and-target

Synopsis

gevm-gqs-queue::gevm-delete-collected-events-created-within-time-period-for-initiator-and-target
(*queue*: gevm-gqs-queue, *start-timestamp*: quantity, *end-timestamp*: quantity,
initiator: item, *target*: item, *event-type*: symbol,
category-filter: sequence, *inclusive-event-states*: sequence,
exclusive-event-states: sequence)

Argument	Description
queue	The queue.
start-timestamp	The start time for considering events for deletion.
end-timestamp	The end time for considering events for deletion.
initiator	The initiating item for an event.
target	The target item for an event.
event-type	The event class for filtering events.
category-filter	A sequence of categories, as symbols, for filtering events, or an empty sequence.
inclusive-event-states	A sequence of event states, as symbols, for including events to delete, or an empty sequence.
exclusive-event-states	A sequence of event states, as symbols, for excluding events to delete, or an empty sequence.

Description

Deletes a collection of events created within a specified time period in a particular queue with a given target and initiator. You express timestamps in G2 time.

Specify *event-type* to delete only events of the specified class, or any subclass. To delete all types of events, specify gevm-event as the event type.

Specify *category-filter* to delete only events in the specified categories or an empty sequence to delete events in all categories.

Specify *inclusive-event-states* to delete only events whose history includes any of the specified event states. Specify *exclusive-event-states* to delete only events whose history does not include any of the specified event states.

gevm-gqs-queue::gevm-delete-events

Synopsis

gevm-gqs-queue::gevm-delete-events
(*queue*: gevm-gqs-queue, *events*: sequence)

Argument	Description
queue	The queue.
events	A sequence of events to delete.

Description

Deletes a collection of events from a queue.

gevm-gqs-queue::gevm-get-collected-events

Synopsis

gevm-gqs-queue::gevm-get-collected-events
(*queue*: gevm-gqs-queue)
–> *events*: sequence

Argument	Description
queue	The queue.

Return Value	Description
events	

Description

Returns all the events that have been collected in a queue.

gevm-gqs-queue::gevm-get-collected-events-created-outside-time-period

Synopsis

gevm-gqs-queue::gevm-get-collected-events-created-outside-time-period
(*queue*: gevm-gqs-queue, *start-timestamp*: quantity, *end-timestamp*: quantity,
event-type: symbol, *category-filter*: sequence,
inclusive-event-states: sequence, *exclusive-event-states*: sequence)
–> *events*: sequence

Argument	Description
queue	The queue.
start-timestamp	The start time for considering events for selection.
end-timestamp	The end time for considering events for selection.
event-type	The event class for filtering events.
category-filter	A sequence of categories, as symbols, for filtering events, or an empty sequence.
inclusive-event-states	A sequence of event states, as symbols, for including events to delete, or an empty sequence.
exclusive-event-states	A sequence of event states, as symbols, for excluding events to delete, or an empty sequence.

Return Value	Description
events	A sequence of selected events.

Description

Returns a collection of events created outside of a specified time period in a queue. You express timestamps in G2 time.

Specify *event-type* to get only events of the specified class, or any subclass. To get all types of events, specify gevm-event as the event type.

Specify *category-filter* to get only events in the specified categories or an empty sequence to get events in all categories.

Specify *inclusive-event-states* to get only events whose history includes any of the specified event states. Specify *exclusive-event-states* to get only events whose history does not include any of the specified event states.

gevm-gqs-queue::gevm-get-collected-events-created-outside-time-period-for-initiator-and-target

Synopsis

gevm-gqs-queue::gevm-get-collected-events-created-outside-time-period-for-initiator-and-target
(*queue*: gevm-gqs-queue, *start-timestamp*: quantity, *end-timestamp*: quantity,
initiator: item, *target*: item, *event-type*: symbol,
category-filter: sequence, *inclusive-event-states*: sequence,
exclusive-event-states: sequence)
–> *events*: sequence

Argument	Description
queue	The queue.
start-timestamp	The start time for considering events for selection.
end-timestamp	The end time for considering events for selection.
initiator	The initiating item for an event.
target	The target item for an event.
event-type	The event class for filtering events.
category-filter	A sequence of categories, as symbols, for filtering events, or an empty sequence.
inclusive-event-states	A sequence of event states, as symbols, for including events to delete, or an empty sequence.
exclusive-event-states	A sequence of event states, as symbols, for excluding events to delete, or an empty sequence.

Description

Returns a collection of events created outside a specified time period in a queue with a given target and initiator. You express timestamps in G2 time.

Specify *event-type* to delete only events of the specified class, or any subclass. To delete all types of events, specify gevm-event as the event type.

Specify *category-filter* to delete only events in the specified categories or an empty sequence to delete events in all categories.

Specify *inclusive-event-states* to delete only events whose history includes any of the specified event states. Specify *exclusive-event-states* to delete only events whose history does not include any of the specified event states.

gevm-gqs-queue::gevm-get-collected-events-created-within-time-period

Synopsis

gevm-gqs-queue::gevm-get-collected-events-created-within-time-period
(*queue*: gevm-gqs-queue, *start-timestamp*: quantity, *end-timestamp*: quantity,
event-type: symbol, *category-filter*: sequence,
inclusive-event-states: sequence, *exclusive-event-states*: sequence)
–> *events*: sequence

Argument	Description
queue	The queue.
start-timestamp	The start time for considering events for selection.
end-timestamp	The end time for considering events for selection.
event-type	The event class for filtering events.
category-filter	A sequence of categories, as symbols, for filtering events, or an empty sequence.
inclusive-event-states	A sequence of event states, as symbols, for including events to delete, or an empty sequence.
exclusive-event-states	A sequence of event states, as symbols, for excluding events to delete, or an empty sequence.

Return Value	Description
events	A sequence of selected events.

Description

Returns a collection of events created within a specified time period in a queue. You express timestamps in G2 time.

Specify *event-type* to get only events of the specified class, or any subclass. To get all types of events, specify gevm-event as the event type.

Specify *category-filter* to get only events in the specified categories or an empty sequence to get events in all categories.

Specify *inclusive-event-states* to get only events whose history includes any of the specified event states. Specify *exclusive-event-states* to get only events whose history does not include any of the specified event states.

gevm-gqs-queue::gevm-get-collected-events-created-within-time-period-for-initiator-and-target

Synopsis

gevm-gqs-queue::gevm-get-collected-events-created-within-time-period-for-initiator-and-target
(*queue*: gevm-gqs-queue, *start-timestamp*: quantity, *end-timestamp*: quantity,
initiator: item, *target*: item, *event-type*: symbol,
category-filter: sequence, *inclusive-event-states*: sequence,
exclusive-event-states: sequence)
–> *events*: sequence

Argument	Description
queue	The queue.
start-timestamp	The start time for considering events for selection.
end-timestamp	The end time for considering events for selection.
initiator	The initiating item for an event.
target	The target item for an event.
event-type	The event class for filtering events.
category-filter	A sequence of categories, as symbols, for filtering events, or an empty sequence.
inclusive-event-states	A sequence of event states, as symbols, for including events to delete, or an empty sequence.
exclusive-event-states	A sequence of event states, as symbols, for excluding events to delete, or an empty sequence.

Description

Returns a collection of events created within a specified time period in a queue with a given target and initiator. You express timestamps in G2 time.

Specify *event-type* to delete only events of the specified class, or any subclass. To delete all types of events, specify gevm-event as the event type.

Specify *category-filter* to delete only events in the specified categories or an empty sequence to delete events in all categories.

Specify *inclusive-event-states* to delete only events whose history includes any of the specified event states. Specify *exclusive-event-states* to delete only events whose history does not include any of the specified event states.

gevm-gqs-queue::gevm-get-collected-events-for-initiator-and-target

Synopsis

gevm-gqs-queue::gevm-get-collected-events-for-initiator-and-target
(*queue*: gevm-gqs-queue, *initiator*: item, *target*: item, *event-type*: symbol, *category-filter*: sequence, *inclusive-event-states*: sequence, *exclusive-event-states*: sequence)
–> *events*: sequence

Argument	Description
queue	The queue.
initiator	The initiating item for an event.
target	The target item for an event.
event-type	The event class for filtering events.
category-filter	A sequence of categories, as symbols, for filtering events, or an empty sequence.
inclusive-event-states	A sequence of event states, as symbols, for including events to delete, or an empty sequence.
exclusive-event-states	A sequence of event states, as symbols, for excluding events to delete, or an empty sequence.

Description

Returns a collection of events created within a specified time period in a queue with a given target and initiator. You express timestamps in G2 time.

Specify *event-type* to delete only events of the specified class, or any subclass. To delete all types of events, specify gevm-event as the event type.

Specify *category-filter* to delete only events in the specified categories or an empty sequence to delete events in all categories.

Specify *inclusive-event-states* to delete only events whose history includes any of the specified event states. Specify *exclusive-event-states* to delete only events whose history does not include any of the specified event states.

gevm-gqs-queue::gevm-get-collected-events-updated-outside-time-period

Synopsis

gevm-gqs-queue::gevm-get-collected-events-updated-outside-time-period
(*queue*: gevm-gqs-queue, *start-timestamp*: quantity, *end-timestamp*: quantity,
event-type: symbol, *category-filter*: sequence,
inclusive-event-states: sequence, *exclusive-event-states*: sequence)
–> *events*: sequence

Argument	Description
queue	The queue.
start-timestamp	The start time for considering events for selection.
end-timestamp	The end time for considering events for selection.
event-type	The event class for filtering events.
category-filter	A sequence of categories, as symbols, for filtering events, or an empty sequence.
inclusive-event-states	A sequence of event states, as symbols, for including events to delete, or an empty sequence.
exclusive-event-states	A sequence of event states, as symbols, for excluding events to delete, or an empty sequence.

Description

Returns a collection of events updated outside a specified time period in a queue. You express timestamps in G2 time.

Specify *event-type* to delete only events of the specified class, or any subclass. To delete all types of events, specify gevm-event as the event type.

Specify *category-filter* to delete only events in the specified categories or an empty sequence to delete events in all categories.

Specify *inclusive-event-states* to delete only events whose history includes any of the specified event states. Specify *exclusive-event-states* to delete only events whose history does not include any of the specified event states.

gevm-gqs-queue::gevm-get-collected-events-updated-outside-time-period-for-initiator-and-target

Synopsis

gevm-gqs-queue::gevm-get-collected-events-updated-outside-time-period-for-initiator-and-target
(*queue*: gevm-gqs-queue, *start-timestamp*: quantity, *end-timestamp*: quantity,
initiator: item, *target*: item, *event-type*: symbol,
category-filter: sequence, *inclusive-event-states*: sequence,
exclusive-event-states: sequence)
–> *events*: sequence

Argument	Description
queue	The queue.
start-timestamp	The start time for considering events for selection.
end-timestamp	The end time for considering events for selection.
initiator	The initiating item for an event.
target	The target item for an event.
event-type	The event class for filtering events.
category-filter	A sequence of categories, as symbols, for filtering events, or an empty sequence.
inclusive-event-states	A sequence of event states, as symbols, for including events to delete, or an empty sequence.
exclusive-event-states	A sequence of event states, as symbols, for excluding events to delete, or an empty sequence.

Description

Returns a collection of events updated outside a specified time period in a queue with a given target and initiator. You express timestamps in G2 time.

Specify *event-type* to delete only events of the specified class, or any subclass. To delete all types of events, specify gevm-event as the event type.

Specify *category-filter* to delete only events in the specified categories or an empty sequence to delete events in all categories.

Specify *inclusive-event-states* to delete only events whose history includes any of the specified event states. Specify *exclusive-event-states* to delete only events whose history does not include any of the specified event states.

gevm-gqs-queue::gevm-get-collected-events-updated-within-time-period

Synopsis

gevm-gqs-queue::gevm-get-collected-events-updated-within-time-period
(*queue*: gevm-gqs-queue, *start-timestamp*: quantity, *end-timestamp*: quantity,
event-type: symbol, *category-filter*: sequence,
inclusive-event-states: sequence, *exclusive-event-states*: sequence)
–> *events*: sequence

Argument	Description
queue	The queue.
start-timestamp	The start time for considering events for selection.
end-timestamp	The end time for considering events for selection.
event-type	The event class for filtering events.
category-filter	A sequence of categories, as symbols, for filtering events, or an empty sequence.
inclusive-event-states	A sequence of event states, as symbols, for including events to delete, or an empty sequence.
exclusive-event-states	A sequence of event states, as symbols, for excluding events to delete, or an empty sequence.

Description

Returns a collection of events updated within a specified time period in a queue. You express timestamps in G2 time.

Specify *event-type* to delete only events of the specified class, or any subclass. To delete all types of events, specify gevm-event as the event type.

Specify *category-filter* to delete only events in the specified categories or an empty sequence to delete events in all categories.

Specify *inclusive-event-states* to delete only events whose history includes any of the specified event states. Specify *exclusive-event-states* to delete only events whose history does not include any of the specified event states.

gevm-gqs-queue::gevm-get-collected-events-updated-within-time-period-for-initiator-and-target

Synopsis

gevm-gqs-queue::gevm-get-collected-events-updated-within-time-period-for-initiator-and-target
(*queue*: gevm-gqs-queue, *start-timestamp*: quantity, *end-timestamp*: quantity,
initiator: item, *target*: item, *event-type*: symbol,
category-filter: sequence, *inclusive-event-states*: sequence,
exclusive-event-states: sequence)
–> *events*: sequence

Argument	Description
queue	The queue.
start-timestamp	The start time for considering events for selection.
end-timestamp	The end time for considering events for selection.
initiator	The initiating item for an event.
target	The target item for an event.
event-type	The event class for filtering events.
category-filter	A sequence of categories, as symbols, for filtering events, or an empty sequence.
inclusive-event-states	A sequence of event states, as symbols, for including events to delete, or an empty sequence.
exclusive-event-states	A sequence of event states, as symbols, for excluding events to delete, or an empty sequence.

Description

Returns a collection of events updated within a specified time period in a queue with a given target and initiator. You express timestamps in G2 time.

Specify *event-type* to delete only events of the specified class, or any subclass. To delete all types of events, specify gevm-event as the event type.

Specify *category-filter* to delete only events in the specified categories or an empty sequence to delete events in all categories.

Specify *inclusive-event-states* to delete only events whose history includes any of the specified event states. Specify *exclusive-event-states* to delete only events whose history does not include any of the specified event states.

gevm-gqs-queue::gevm-get-event-count-for-queue

Synopsis

gevm-gqs-queue::gevm-get-event-count-for-queue
(*queue*: gevm-gqs-queue)
–> *count*: integer

Argument	Description
queue	The queue.

Return Value	Description
count	The number of events in the queue.

Description

Returns the count of events in a queue.

gevm-gqs-queue::gevm-remove-all-events

Synopsis

gevm-gqs-queue::gevm-remove-all-events
(*queue*: gevm-gqs-queue)

Argument	Description
queue	The queue.

Description

Removes all events from a queue, without deleting the events.

gevm-gqs-queue::gevm-remove-events

Synopsis

gevm-gqs-queue::gevm-remove-events
(*queue*: gevm-gqs-queue, *events*: sequence)

Argument	Description
queue	The queue.
events	A sequence of events to remove from the queue.

Description

Removes a collection of events from a queue, without deleting the events.

gevm-find-queue-by-key

Synopsis

gevm-find-queue-by-key
(*queue-key*: text)
–> *return-value*: truth-value, *queue*: gevm-gqs-queue

Argument	Description
queue-key	The value of the key attribute of a queue.

Return Value	Description
return-value	A truth value indicating whether the queue key was found.
queue	The queue associated with the key.

Description

Returns a queue given its key. If the queue is found, this procedure returns a value of true, as well as a reference to the queue. If the queue is not found, it returns a value of false and a reference to the default Messages queue in which operator messages appear.

gevm-get-all-queues

Synopsis

gevm-get-all-queues
    ()
    –> *queue-keys*: sequence

Return Value	Description
queue-keys	A sequence of queue keys that exist in the KB.

Description

Returns a sequence of all queues in the KB. The sequence that is returned contains an alphabetically sorted list of all queue keys.

gevm-get-message-properties

Synopsis

gevm-get-message-properties
()
–> *priorities*: structure

Return Value	Description
priorities	A structure that contains information about the properties of each message.

Description

Returns information about the properties of all messages. You can use this information to determine the columns to display in a message browser. Not all properties are valid for all message classes, which include gevm-message and any subclass.

This procedure returns a structure with the following syntax:

structure
(g2-start-timestamp: float, key-name: symbol, properties: sequence)

- g2-start-timestamp is the G2 start time, which is the time when G2 was most recently started, as a float. The time is expressed as the number of seconds that had elapsed since January 1, 1970, Greenwich Mean Time, at the moment when G2 was started. This timestamp is required in remote systems to convert timestamps associated with messages, such as the creation time, to a text format.
- key-name is a symbol uniquely naming each message, which is automatically generated based on the class, type, initiator and target. The key that is returned is the value of the gevm-key attribute of the message.
- properties is a sequence of structures listing all properties of a message.

See item::grtl-get-properties-type-info in the *G2 Run-Time Library User's Guide* for information on the syntax of the structure of property attributes.

gevm-get-most-recent-unacknowledged-event

Synopsis

gevm-get-most-recent-unacknowledged-event
(*queue*: gqs-queue)
–> *event*: item-or-value

Argument	Description
queue	The queue.

Return Value	Description
event	The most recently created event that has not been acknowledged.

Description

Returns the most recently created event that has not been acknowledged from a particular queue.

gevm-get-property-values-of-all-events

Synopsis

gevm-get-property-values-of-all-events
(*queue-key*: text, *user-mode*: text, *format-timestamps-as-text*: truth-value)
–> *property-values*: structure

Argument	Description
queue-key	The value of the key attribute of a queue.
user-mode	The user mode in which the property values to get are visible.
format-timestamps-as-text	If true, formats the timestamps as text values.

Return Value	Description
property-values	A structure that contains information about the property values of all events in a queue.

Description

Returns the property values of all events visible in a given user mode in particular queue, as well as other information. The syntax of the structure that is returned is:

structure
(g2-start-timestamp: float, formatted-timestamps-as-text: truth-value, queue-key: text, key-name: symbol, event-properties: sequence)

- g2-start-timestamp is the G2 start time, which is the time when G2 was most recently started, as a float. The time is expressed as the number of seconds that had elapsed since January 1, 1970, Greenwich Mean Time, at the moment when G2 was started. This timestamp is required in remote systems to convert timestamps associated with messages, such as the creation time, to a text format.
- formatted-timestamps-as-text is the value of the *format-timestamps-as-text* argument.
- queue-key is a text naming the key attribute of the queue.

- key-name is a symbol uniquely naming each message, which is automatically generated based on the class, type, initiator and target. The key that is returned is the value of the gevm-key attribute of the message.
- properties is a sequence of structures listing all properties of an event.

If the queue cannot be located, this procedure returns an empty structure.

See item::grtl-get-properties-type-info in the *G2 Run-Time Library User's Guide* for information on the syntax of the structure of property attributes.

gevm-get-property-values-of-events

Synopsis

gevm-get-property-values-of-all-events
(*queue-key*: text, *items-or-item-keys*: sequence, *user-mode*: text,
format-timestamps-as-text: truth-value)
–> *property-values*: structure

Argument	Description
queue-key	The value of the key attribute of a queue.
items-or-item-keys	A sequence of gevm-event objects whose property values to get, or a sequence of text values, where each text is the gevm-key of a gevm-event or subclass. You specify keys rather than event objects for distributed applications.
user-mode	The user mode in which the property values to get are visible.
format-timestamps-as-text	If true, formats the timestamps as text values.

Return Value	Description
property-values	A structure that contains information about the property values of events in a queue.

Description

Returns the property values of a collection of event that are visible in a given user mode in particular queue, as well as other information. The syntax of the structure that is returned is:

structure
(g2-start-timestamp: float, formatted-timestamps-as-text: truth-value,
queue-key: text, key-name: symbol, event-properties: sequence)

- g2-start-timestamp is the G2 start time, which is the time when G2 was most recently started, as a float. The time is expressed as the number of seconds that had elapsed since January 1, 1970, Greenwich Mean Time, at the moment when G2 was started. This timestamp is required in remote systems to convert timestamps associated with messages, such as the creation time, to a text format.

- formatted-timestamps-as-text is the value of the *format-timestamps-as-text* argument.
- queue-key is a text naming the key attribute of the queue.
- key-name is a symbol uniquely naming each message, which is automatically generated based on the class, type, initiator and target. The key that is returned is the value of the gevm-key attribute of the message.
- properties is a sequence of structures listing all properties of an event.

If the queue cannot be located, this procedure returns an empty structure.

See item::grtl-get-properties-type-info in the *G2 Run-Time Library User's Guide* for information on the syntax of the structure of property attributes.

gevm-get-queues-containing-item

Synopsis

gevm-get-queues-containing-item
(item: item)
–> *queues*: sequence

Argument	Description
item	The item contained in one or more queues.

Return Value	Description
queues	A sequence of queues that contain the item.

Description

Returns a sequence of all queues that currently contain a given item.

gevm-get-selected-property-values-of-all-events

Synopsis

gevm-get-selected-property-values-of-all-events
(*queue-key*: text, *property-names*: sequence,
format-timestamps-as-text: truth-value)
–> *property-values*: structure

Argument	Description
queue-key	The value of the key attribute of a queue.
property-names	A sequence of properties names whose values to get.
format-timestamps-as-text	If true, formats the timestamps as text values.

Return Value	Description
property-values	A structure that contains information about the specified property values of all events in a queue.

Description

Returns the values of specified properties of all events in particular queue, as well as other information. The syntax of the structure that is returned is:

structure
(g2-start-timestamp: float, formatted-timestamps-as-text: truth-value,
queue-key: text, key-name: symbol, event-properties: sequence)

- g2-start-timestamp is the G2 start time, which is the time when G2 was most recently started, as a float. The time is expressed as the number of seconds that had elapsed since January 1, 1970, Greenwich Mean Time, at the moment when G2 was started. This timestamp is required in remote systems to convert timestamps associated with messages, such as the creation time, to a text format.
- formatted-timestamps-as-text is the value of the *format-timestamps-as-text* argument.
- queue-key is a text naming the key attribute of the queue.

- key-name is a symbol uniquely naming each message, which is automatically generated based on the class, type, initiator and target. The key that is returned is the value of the gevm-key attribute of the message.
- properties is a sequence of structures listing all properties of an event.

If the queue cannot be located, this procedure returns an empty structure.

See item::grtl-get-properties-type-info in the *G2 Run-Time Library User's Guide* for information on the syntax of the structure of property attributes.

gevm-get-selected-property-values-of-events

Synopsis

gevm-get-selected-property-values-of-all-events
(*queue-key*: text, *keys*: sequence, *property-names*: sequence,
format-timestamps-as-text: truth-value)
–> *property-values*: structure

Argument	Description
queue-key	The value of the key attribute of a queue.
keys	A sequence of message or event keys whose property values to get.
property-names	A sequence of properties names whose values to get.
format-timestamps-as-text	If true, formats the timestamps as text values.

Return Value	Description
property-values	A structure that contains information about the property values of the specified events in a queue.

Description

Returns the values of specified properties of events for a given set of keys in a particular queue, as well as other information. The syntax of the structure that is returned is:

structure
(g2-start-timestamp: float, formatted-timestamps-as-text: truth-value,
queue-key: text, key-name: symbol, event-properties: sequence)

- g2-start-timestamp is the G2 start time, which is the time when G2 was most recently started, as a float. The time is expressed as the number of seconds that had elapsed since January 1, 1970, Greenwich Mean Time, at the moment when G2 was started. This timestamp is required in remote systems to convert timestamps associated with messages, such as the creation time, to a text format.
- formatted-timestamps-as-text is the value of the *format-timestamps-as-text* argument.

- queue-key is a text naming the key attribute of the queue.
- key-name is a symbol uniquely naming each message, which is automatically generated based on the class, type, initiator and target. The key that is returned is the value of the gevm-key attribute of the message.
- properties is a sequence of structures listing all properties of an event.

If the queue cannot be located, this procedure returns an empty structure.

See item::grtl-get-properties-type-info in the *G2 Run-Time Library User's Guide* for information on the syntax of the structure of property attributes.

GQS Queue Class and Operations

Class

gqs-queue on page 324

Methods

gqs-queue::gqs-clear-queue on page 328
gqs-queue::gqs-receive-items on page 329

Procedures

gqs-activate-attribute-monitoring on page 330
gqs-add-monitored-attributes on page 331
gqs-clear-queue on page 332
gqs-deactivate-attribute-monitoring on page 333
gqs-force-input-buffer-into-queue on page 334
gqs-get-collected-items on page 335
gqs-get-monitored-attributes on page 336
gqs-get-queues-containing-item on page 337
gqs-launch-view on page 338
gqs-number-of-collected-items on page 347
gqs-receive-items on page 339
gqs-receive-single-item on page 340
gqs-remove-all-monitored-attributes on page 341
gqs-remove-items on page 342
gqs-remove-monitored-attributes on page 343
gqs-remove-single-item on page 344
gqs-send-single-item on page 346

Functions

gqs-number-of-collected-items on page 347

gqs-queue

This is the basic foundation class for gevm-gqs-queue. The object stores an ordered list of items. Each queue can support a number of views, but does not itself have a visual representation, aside from its icon.

Class Inheritance Path

gqs-queue, gfr-object-with-uuid, object, gfr-item-with-uuid, item

Attributes

Attribute	Description
gqs-initially-monitor-for-deletion-events	Controls whether the queue monitors for the deletion of the items it contains when G2 starts. This is an attribute that is applicable to a GQS queue, which can contain items of any class.
Allowable values:	Any truth value
Default value:	false
gqs-initially-monitor-for-attribute-change-events	Controls whether the queue monitors the attributes of the items it contains, when G2 starts. If your entry class contains additional attributes that need to be monitored, this attribute may come into play. To turn monitoring on or off after G2 has started, use gqs-activate-attribute-monitoring on page 330 or gqs-deactivate-attribute-monitoring on page 333.
Allowable values:	Any truth value
Default value:	false
gqs-item-deletion-callback	The name of the procedure to be called when an item contained by the queue is deleted.
Allowable values:	The name of the procedure to be called when an item contained by the queue is deleted, or the symbol UNSPECIFIED.

Attribute	Description
Default value:	UNSPECIFIED
gqs-attribute-update-callback	The name of the procedure to be called when an attribute of an item contained by the queue receives a value. See also gqs-view-manager on page 436 for an alternate way to customize the behavior of the queues. Your callback must take three arguments: my-update-callback (*Queue*: class gqs-queue, *MonitoredItem*: class item, *AttributeName*: symbol) Also, the attribute must either be monitored (see gqs-add-monitored-attributes on page 331) or be one of the built-in attributes described under gevm-event on page 127.
Allowable values:	The name of the procedure to be called when an item contained by the queue is updated, or the symbol UNSPECIFIED.
Default value:	UNSPECIFIED
gqs-item-addition-callback	The name of a procedure to be called when items are added to the queue. Use this procedure if you want to take an action, on a per-queue basis, as new entries are added to the queue. To simply be notified of entries arriving on the queue, see the description of gqs-view-manager on page 436. Note that callbacks cannot be further extended by your users. If you want to modify what happens upon addition, but want the system to remain open for further extensions, you may want to subclass the queue and extend the method gqs-view-manager::gqs-update-view-per-addition on page 438. If you do write a callback for the queue, your callback must accept three arguments: my-addition-callback (*Queue*: class gqs-queue, *ItemsAdded*: class item-list, *Client*: class object) The last argument is the same client object passed to the procedure that added the items to the queue.

Attribute	Description
Allowable values:	The name of the procedure to be called when one or more items are added to the queue, or the symbol UNSPECIFIED.
Default value:	UNSPECIFIED
gqs-item-removal-callback	The name of a procedure to be called when items are removed from the queue. Use this procedure if you want to take some action, on a per-queue basis, before an entry gets removed. To simply be notified of entry removals from the queue, see the description of gqs-view-manager on page 436. Note that callbacks cannot be further extended by your users. If you want to modify what happens upon removal, but want the system to remain open for further extensions, you may want to subclass the queue and extend the method gqs-view-manager::gqs-update-view-per-removal on page 441. If you do write a callback for the queue, your callback must accept three arguments: my-removal-callback (*Queue*: class gqs-queue, *ItemsRemoved*: class item-list, *Client*: class object) The last argument is the same client object passed to the procedure that removed the items from the queue.
Allowable values:	The name of the procedure to be called when one or more items contained by the queue are removed, or the symbol UNSPECIFIED.
Default value:	UNSPECIFIED
gqs-view-template-or-access-table	The name of the gqs-queue-access-table that controls what template should be used to create a view of the queue, on the current window, or the name of an item that implements the method gqs-create-view. If this attribute is unspecified, the menu choice launch-view does not appear on the queue.

Attribute	Description
Allowable values:	Any symbol naming either a gqs-queue-access-table, a gqmv-tabular-view-template, or other item as described below.
Default value:	UNSPECIFIED
gqs-logging-manager	The name of an optional logging manager for the queue, which must be an instance of a glf-logging-manager.
Allowable values:	Any symbol
Default value:	UNSPECIFIED
gqs-update-latency	The latency, in seconds, between updates when items are added or removed from the queue. If this interval is too short, the efficiency of the system may suffer, because the queue is not able to treat multiple insertions or removals as a group.
Allowable values:	Any positive float
Default value:	1.0

gqs-queue::gqs-clear-queue

Synopsis

gqs-queue::gqs-clear-queue
(*queue*: gqs-queue, *client*: object)

Argument	Description
queue	The queue to be cleared.
client	The client for this call.

Description

Use this procedure to remove all items from a queue. Views, if any, are updated automatically.

If you want to add side effects when all items are removed from a queue, you should subclass gqs-queue and override this method. Be sure to call this method through a call next method statement in your method.

gqs-queue::gqs-receive-items

Synopsis

gqs-queue::gqs-receive-items
(*queue*: gqs-queue, *sender*: item-or-value, *incomingitems*: item-list,
client: object)

Argument	Description
queue	The queue receiving items.
sender	An item indicating the source of the items, or false.
incomingitems	The items to be added to the queue.
client	The client for this call.

Description

Call this method to add a list of items to a queue.

If you want certain side effects to occur when items are added to a queue, you can subclass gqs-queue and override this method. If so, be sure to call this method using a call next method statement.

Note that calling this method does not immediately add the items to the queue; instead, it adds them to the queue's input buffer for later insertion.

For more information, see gqs-force-input-buffer-into-queue on page 334. This method does use the *Sender* argument, which is provided for user overrides of this method.

gqs-activate-attribute-monitoring

Synopsis

gqs-activate-attribute-monitoring
(*queue*: gqs-queue, *client*: object)

Argument	Description
queue	The target queue.
client	The client for this call.

Description

This procedure activates attribute monitoring for the current set of monitored attributes. When attribute monitoring is active, the queue sends attribute-change events to its views, so they reflect the current state of the items in the queue. To specify the monitored attributes, see gqs-add-monitored-attributes on page 331.

Use of attribute monitoring may have a noticeable effect on performance if the number of items in the queue is large, and it may be possible to achieve the same goal by monitoring attribute changes at the view level, which is much more efficient. Also see gqs-deactivate-attribute-monitoring on page 333.

gqs-add-monitored-attributes

Synopsis

gqs-add-monitored-attributes
(*queue*: gqs-queue, *attributes*: symbol-list, *client*: object)

Argument	Description
queue	The queue that contains items to be monitored.
attributes	A list of attributes to add to the set of monitored attributes.
client	The client for this call.

Description

Use this procedure to set or extend the set of attributes monitored in the items contained in *Queue*. See also gqs-remove-monitored-attributes on page 343 and gqs-remove-all-monitored-attributes on page 341. Adding attributes using this procedure does not start monitoring if it is not already on. To turn on attribute monitoring, use gqs-activate-attribute-monitoring on page 330.

gqs-clear-queue

Synopsis

gqs-clear-queue
(*queue*: gqs-queue, *client*: object)

Argument	Description
queue	The queue to be cleared
client	The client for this call.

Description

See gqs-queue::gqs-clear-queue on page 328.

gqs-deactivate-attribute-monitoring

Synopsis

gqs-deactivate-attribute-monitoring
(*queue*: gqs-queue, *client*: object)

Argument	Description
queue	The queue that is the target of this call.
client	The client for this call.

Description

This procedure turns off attribute monitoring on the *Queue*. See gqs-activate-attribute-monitoring on page 330 for more details.

gqs-force-input-buffer-into-queue

Synopsis

gqs-force-input-buffer-into-queue
(*queue*: gqs-queue, *client*: object)

Argument	Description
queue	The queue that may contain items in its input buffer.
client	The client for this call.

Description

For efficiency, each queue maintains a list of items waiting to be added to the queue. Periodically, with a latency determined by the gqs-update-latency of the queue, the items in this list are added as a group to the queue. Occasionally, you may need to force the items from the input buffer into the queue using this procedure, before performing some operation.

For example, if you programmatically send items to the queue, and them programmatically sort the items, you need to call this procedure to assure that all items sent to the queue are actually incorporated into the queue when you perform the sort.

gqs-get-collected-items

Synopsis

gqs-get-collected-items
(*queue*: gqs-queue, *item-list*: item-list, *client*: object)

Argument	Description
queue	The queue whose items are to be accessed.
item-list	The item list that contains the results.
client	The client for this call.

Description

Adds a queue's collected items to the end of *item-list*. The list *item-list* is not cleared before this operation. This procedure enables you to obtain a list of all items in a queue.

gqs-get-monitored-attributes

Synopsis

gqs-get-monitored-attributes
(*queue*: gqs-queue, *attributes*: symbol-list, *client*: object)

Argument	Description
queue	The queue that is the target of this call.
attributes	The list to contain the results.
client	The client for this call.

Description

This procedure returns the set of monitored attributes for the given queue. The symbols are appended to *Attributes* without clearing the list first.

gqs-get-queues-containing-item

Synopsis

gqs-get-queues-containing-item
(*item*: item, *queue-list*: item-list, *client*: object)

Argument	Description
item	The item that may be contained in one or more queues.
queue-list	The list that is to contain the results.
client	The client for this call.

Description

This procedure returns in *queue-list* all the queues that currently contain the given *item*. *Queue-list* is not cleared before the results are inserted.

gqs-launch-view

Synopsis

gqs-launch-view
(*queue*: gqs-queue, *client*: object)
–> *view-manager*: class gqs-view-manager

Argument	Description
queue	The queue that is being asked to launch a view.
client	The window or other client object where the view is to be launched.

Return Value	Description
view-manager	A newly created view manager.

Description

This procedure is used to create a new view manager associated with the given queue.

If the gqs-view-template-or-access-table of *Queue* contains the name of a queue access table, gqs-get-view-template is used to determine the proper view template for the given client.

Otherwise, the gqs-view-template-or-access-table of the *Queue* must name an item that can be used as a view template, that is, any item implementing gqs-create-view. Once the template has been identified, the method gqs-create-view is called to create the view manager, which is returned by this procedure.

gqs-receive-items

Synopsis

gqs-receive-items
(*queue*: gqs-queue, *sender*: item-or-value, *entries*: item-list,
client: object)

Argument	Description
queue	The queue receiving items.
sender	Typically false. Not used by this subclass.
entries	A list of the new entries to be added to the queue.
client	The client for this call.

Description

Call this procedure to add a list of items to *Queue*. Note that calling this procedure does not immediately add the items to the queue; instead, it adds them to the queue's input buffer for later insertion. For more information, see gqs-force-input-buffer-into-queue on page 334. This procedure does use the *Sender*, which is provided for user overrides of this method, but is not used by default.

gqs-receive-single-item

Synopsis

gqs-receive-single-item
(*receiver*: gqs-queue, *item*: item, *client*: object)

Argument	Description
receiver	The queue to receive the item.
item	The item to be introduced into the queue.
client	The client for this call.

Description

Use this procedure to introduce a new item to a queue, without an explicit sender. Note that there is a latency between the receipt of an item by a queue, and the actual incorporation of the item into the queue. See gqs-force-input-buffer-into-queue on page 334 for details. If you need to introduce multiple items into a queue at one time, it is more efficient to use gqs-receive-items on page 339. See also gqs-send-single-item on page 346.

See also gqs-queue::gqs-receive-items on page 329.

gqs-remove-all-monitored-attributes

Synopsis

gqs-remove-all-monitored-attributes
(*queue*: gqs-queue, *client*: object

Argument	Description
queue	The queue that is the target of this call.
client	The client for this call.

Description

This procedure removes all attributes from the set of monitored attributes. This procedure does not turn off monitoring, so if attributes are added via gqs-add-monitored-attributes, monitoring already will be on. To deactivate monitoring, use gqs-deactivate-attribute-monitoring on page 333.

gqs-remove-items

Synopsis

gqs-remove-items
(*queue*: gqs-queue, *items-to-remove*: item-list, *client*: object)

Argument	Description
queue	The queue containing items to be removed.
items-to-remove	A list of items to be removed from *Queue*.
client	The client for this call.

Description

Use this method to remove a list of items from a queue. See also gqs-remove-single-item on page 344.

gqs-remove-monitored-attributes

Synopsis

gqs-remove-monitored-attributes
(*queue*: gqs-queue, *attributes*: symbol-list, *client*: object)

Argument	Description
queue	The queue that is the target of this call.
attributes	A list of attributes that are to be monitored for changes.
client	The client for this call.

Description

Use this procedure to remove attributes that are currently in the set of monitored attributes of the queue. See also gqs-add-monitored-attributes on page 331 and gqs-remove-all-monitored-attributes on page 341.

gqs-remove-single-item

Synopsis

gqs-remove-single-item
(*queue*: gqs-queue, *item*: item, *client*: object)

Argument	Description
queue	The queue that contains the item to be removed.
item	The item to be removed from *Queue*.
client	The client for this call.

Description

Use this procedure to remove an item from a queue, without an explicit sender. If you need to remove multiple items from a queue at one time, it is more efficient to use gqs-remove-items on page 342.

gqs-send-items

Synopsis

gqs-send-items
(*sender*: gqs-queue, *receiver*: gqs-queue, *send-list*: item-list,
client: object)

Argument	Description
sender	The queue responsible for sending items to the receiving queue.
receiver	The queue receiving the items in *send-list*.
sendlist	The list of items to be added to *receiver*.
client	The client for this call.

Description

Use this procedure to send a list of items to a queue. This procedure provides an efficient way to send multiple items from one queue to another. In addition, this procedure insures compatible queue and entry types when entries are sent from one queue to another.

gqs-send-single-item

Synopsis

gqs-send-single-item
(*sender*: item, *receiver*: gqs-queue, *item*: item, *client*: object)

Argument	Description
sender	The source of the item.
receiver	The queue receiving the item.
item	The item being sent to *Receiver*.
client	The client for this call.

Description

Use this procedure to send a single item to a queue. If you need to send multiple items to a queue at one time, it is more efficient to use gqs-send-items on page 345. If the sender is a gqs-queue, this procedure has the same effect as calling gqs-send-items with a list of one item. If the sender is not a gqs-queue, this method is the same as using gqs-receive-single-item on the *Receiver*.

gqs-number-of-collected-items

Synopsis

gqs-number-of-collected-items
(*queue*: class gqs-queue)

Argument	Description
queue	An instance of a gqs-queue.

Description

This function returns the number of items contained in a queue.

Customizations

You can customize these aspects of message queues:

- Message queue classes.
- Message log handling.
- Message database table creation.

Customizing Message Queue Classes

To create a custom queue definition, you must use a grtl-event-source-class-definition class definition rather than a G2 class definition. The superior class of custom queues should be gevm-gqs-queue.

Customizing the Message Log Handler

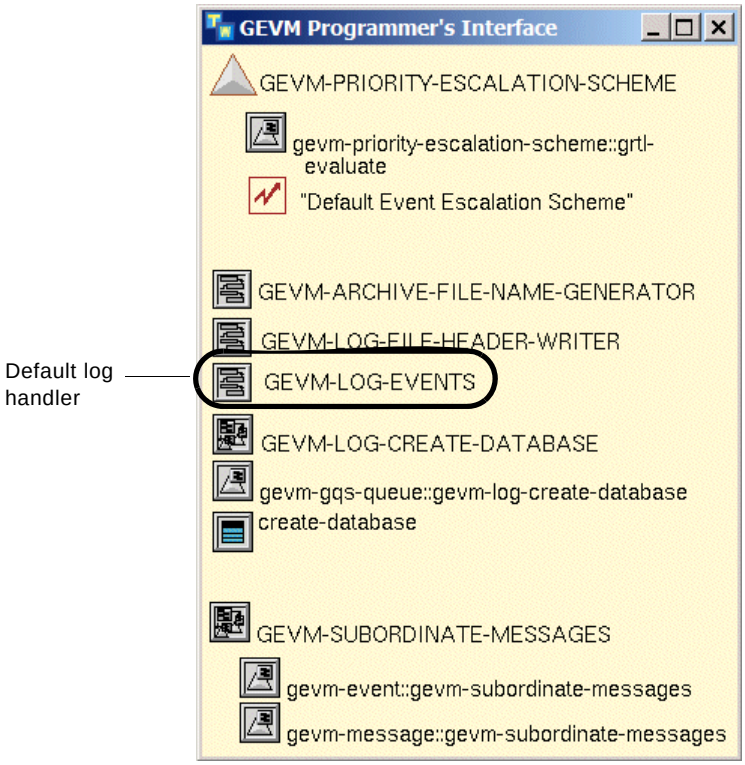
To create a custom message log handler, you must first create a custom procedure, based on the default log handler. You then edit the attribute that defines the log procedure in the gevm-message-queue object, whose log handler you want to customize. You edit the log procedure through the G2 table, which you must do in Administrator mode.

You customize the message log handler for a particular message queue.

To customize the message log handler:

1. Create a procedure to implement your custom log handler.
2. On the Programmer's Interface workspace, display the text of the gevm-log-events procedure and copy the procedure text to your custom procedure.

Here is the default log handler:



3 Edit the procedure text of your custom procedure, as needed.

4 Choose Modules > Queues > Manage, select the queue whose log handler you want to customize, and click the Show button ( ).

You will see the gevm-message-queue object associated with the selected queue.

5 Switch to Administrator mode and choose table on the message queue.

6 Edit the gevm-log-procedure attribute to refer to your custom log handler procedure, whose default is gevm-log-events.

For example, here is gevm-all-messages-queue and its table for customizing the log handler for the Message Queue:

GEVM-ALL-MESSAGES-QUEUE, a gevm-gqs-qu...

UUID	"be4989e0fcef11d6b90d000802639ab8"
Notes	OK
Item configuration	none
Names	GEVM-ALL-MESSAGES-QUEUE
Key	"Messages"
_event notification enabled	true
_listeners	structure ()
Gfr UUID	"aef6f790fcef11d6b90d000802639ab8"
Glf logging enabled	false
Glf log directory	"C:\Program Files\Gensym\g2-8.2r0\g2\logs\"
Glf log file name template	"log-messages-*.csv"
Glf log file name generator	gevm-archive-file-name-generator
Glf current log file	""
Glf file header writer	gevm-log-file-header-writer
Glf time interval to open new log file	86400
Glf maximum file size in bytes	100000
Glf log file scheduler	glf-default-log-file-scheduler
Glf automatically delete empty log files	true
Gqs initially monitor for deletion events	false
Gqs initially monitor for attribute change events	false
Gqs item deletion callback	gevm-queue-item-deletion
Gqs attribute update callback	gevm-queue-attribute-update
Gqs item addition callback	gevm-queue-item-addition
Gqs item removal callback	gevm-queue-item-removal
Gqs view template or access table	gevm-messages-view-access-table
Gqs logging manager	unspecified
Gqs update latency	1.0
_gqs items are awaiting insertion	false
_gqs items awaiting insertion	an item-list
Gevm max events to queue	10000
Gevm log to file	true
Gevm log to database	false
Gevm log database interface or pool	"gdsm-sql-db"
Gevm log database table	gevm_messages
Gevm log to jms	false
Gevm log jms interface or pool	"gdsm-jms-interface-65"
Gevm log jms as xml	false
Gevm log additions	true
Gevm log removals	true
Gevm log state changes	true
Gevm log procedure	custom-log-handler
Message class	gevm-message

CUSTOM-LOG-HANDLER

Configure gevm-log-procedure to be your custom log handler.

Customizing Database Table Creation

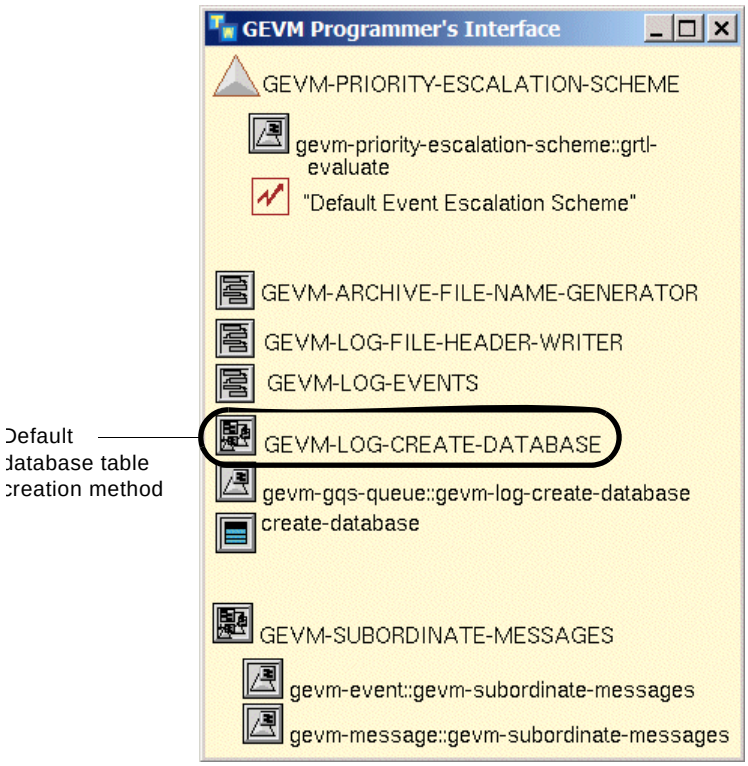
If you create a custom queue, you might want to implement a custom method for creating the database tables for that queue, when logging messages to a database. To do this, you create a custom implementation of the gevm-log-create-database method for your custom queue.

To customize database table creation for a custom queue:

1 Create a method to implement your custom database table creation method for your custom queue class.

2 On the Programmer's Interface workspace, display the text of the gevm-log-create-database method and copy the text to your custom method.

Here is the default database table creation method for the gevm-gqs-queue class:



3 Edit the text of your custom method, as needed.

You application uses the custom method implementation for your custom queue when creating database tables.

6

Access Tables

Describes how to configure access tables, and describes the GQS class and operations for access tables.

Introduction **353**

Access Table Configuration **354**

Access Table Classes and Operations **358**

Introduction

This chapter describes how to configure access tables and the API for access tables.

Access Table Configuration

An access manager specifies the template that defines the queue view for a named user or window class accessing the application in a particular user mode. If the application displays the contents of a queue using only one template, it is not necessary to use an access manager. If, however, the application provides more than one view for a particular queue, it is necessary to create and configure an access table to indicate which users view the queue using which view templates.

By specifying different combinations of user names or window classes, user modes, and queue view templates, you can control how every application user accesses the queues. The access table and the ability to define multiple queue views for a single queue enable you to manage access to the contents of all queues, presenting to each user or category of user only the needed information.

How the Access Manager Works

The access manager contains a sequenced list that associates a queue view template with either a specified user name or window class, accessing the application in a specified access mode.

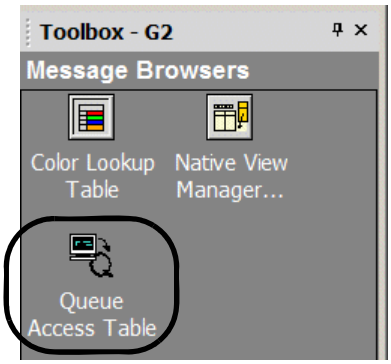
The access manager determines which queue view template to use by following this sequence of steps:

1. The access manager compares the name of the currently logged in user with the user names in its list, starting with the user name specified in the top row in the list.
2. If the currently logged in user appears in the list, the access manager then compares the user mode for that user with the user mode for that user's entry. If the user is logged in with the user mode specified in the row for that user, then the access manager uses the view template specified for the user.
3. If the currently logged in user does not appear in the list or if the user is logged in in a different user mode (and no entry for that user mode is found in the table), then the access manager compares the current window with the list of window classes in the table.
4. If the currently logged in user's window appears in the list and the user is logged in with the specified user mode, then the access manager uses the template associated with this window.
5. If neither the user name nor the window class is found in the list, or if the user mode is not satisfied for the current window, then the access manager signals an error. (In the sample access table, the last row associates a view template with all users not specifically addressed in the previous rows.)

Creating and Configuring the Queue Access Table

To create a queue access table:

1 In Developer mode, choose View > Toolbox - G2 and display the Message Browsers palette:

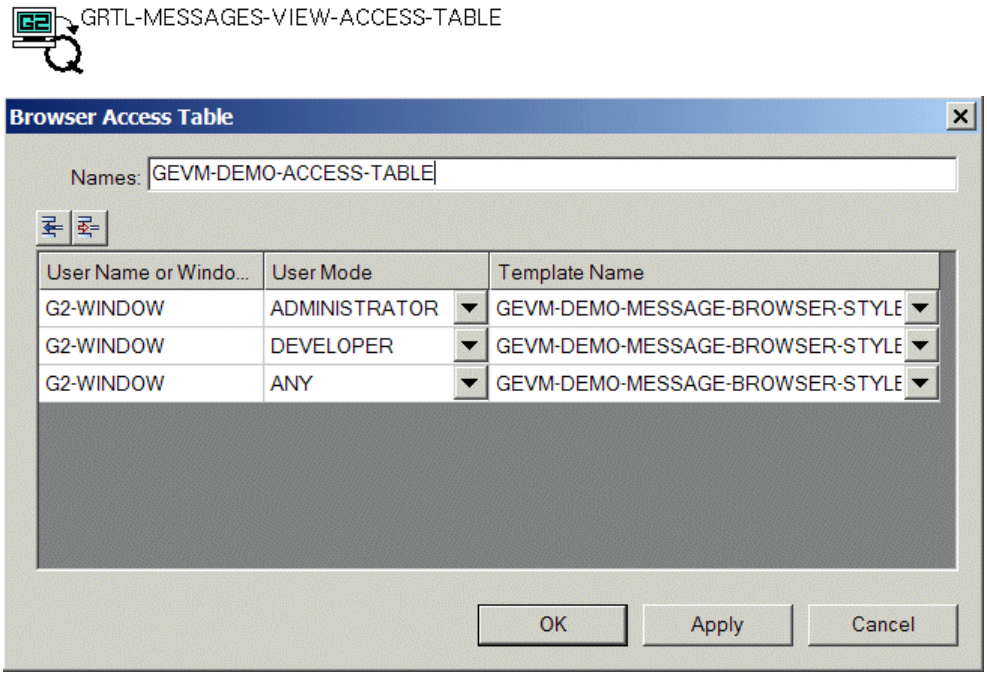


2 Create a Queue Access Table from the palette.

The access table determines which tabular view template to use in each user mode.

3 Display the properties dialog for the queue access table to determine which view template to use with a particular user mode.

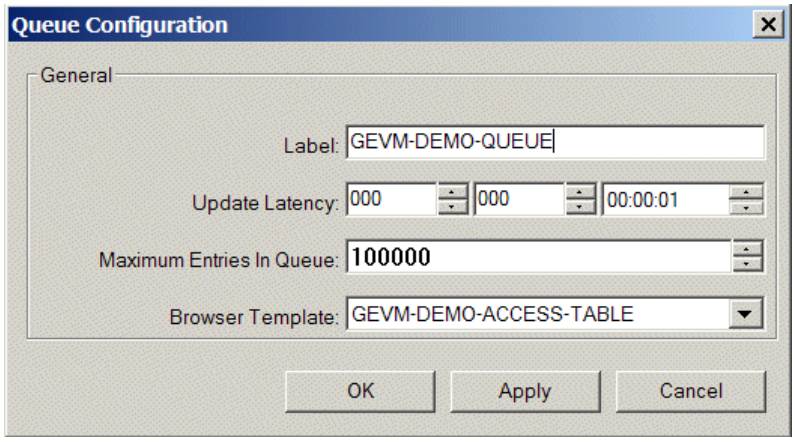
Here is the properties dialog for a Queue Access Table that is configured for administrator and developer modes. The dialog lists the initial operator modes and the corresponding view templates to display in each user mode.



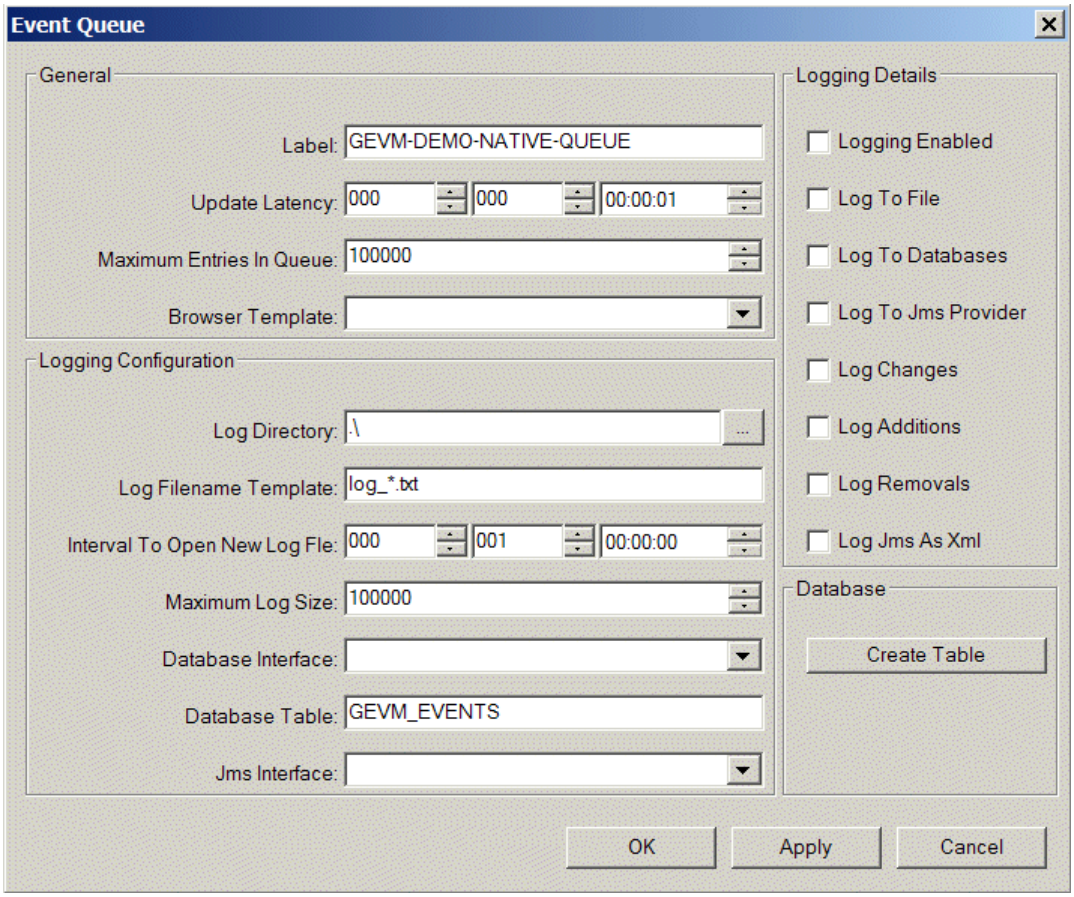
4 Display the properties dialog for the Message Queue and configure the Browser Template to specify the access table to use for choosing the tabular view template.

If you create only one view manager template for a single user mode, you do not need to create a Queue Access Table or configure the Browser Template of the message queue.

Here is the properties dialog for a message queue that uses a queue access table in Modeler mode:



Here is the properties dialog for a message queue that uses a queue access table in System-Administrator mode:



Access Table Classes and Operations

Class

gqs-queue-access-table on page 359

Procedure

gqs-get-view-template on page 361

gqs-queue-access-table

This class is used to specify which template is to be used to create a view for a given user or window. The access table consists of three parallel arrays, giving a user name or window class, a user mode, and a template name, respectively. Given a window, the access table is searched from the first element of the array to the last, stopping when user name or window class and user mode match the properties of the given g2-window. The matching procedure is implemented by gqs-get-view-template.

The easiest way to specify a queue access table is to clone one from the GQS palette, and then select the configure access table menu choice. Fill in the table specifying a user name or window class in the first column, a G2 user mode in the second column, and the desired template name in the third column. "Any" is acceptable in the g2-user-mode column. Add as many rows as required in your table, using the GXL add row control button.

For example, suppose you want to associate Template-1 with User-1 regardless of user mode, Template-2 with any user in administrator mode, and Template-3 for any user in any other mode. Then, the three arrays or columns would be:

gqs-user-name-or-window-class = (User-1, g2-window, g2-window)

gqs-user-mode = (ANY, ADMINISTRATOR, ANY)

gqs-view-template = (TEMPLATE-1, TEMPLATE-2, TEMPLATE-3)

Once the access table is created, give it a name, then set the gqs-view-template-or-access-table attribute of a gqs-queue to the name of your access table.

Class Inheritance Path

gqs-queue-access-table, object, item

Attributes

Attribute	Description
gqs-user-name-or-window-class	A list of user names or G2 window classes.
Allowable values:	Any symbol-array
Default value:	SYMBOL-ARRAY
gqs-user-mode	A list of user modes corresponding to the list of user names/window classes. The symbol ANY can be used if the mode is not part of the selection criteria.
Allowable values:	Any symbol-array
Default value:	SYMBOL-ARRAY
gqs-view-template	A list of template names corresponding to the user name/window class and user mode.
Allowable values:	Any symbol-array
Default value:	SYMBOL-ARRAY

gqs-get-view-template

Synopsis

gqs-get-view-template
(*table*: gqs-queue-access-table, *win*: g2-window)
–> *view-template*: symbol

Argument	Description
table	A queue access table.
win	The window where the queue view is to be launched.

Return Value	Description
view-template	The name of the template for the view to be used on this window.

Description

This procedure interrogates a queue access table and determines the proper view template for the given window. The search in the access table proceeds in the following order:

1. First, there is an attempt to match by user name.
2. If the user name does not exist or no match is found, then the search is conducted by window class.

In each case, a match also requires matching the user mode of the window. See gqs-queue-access-table on page 359 for more details.

7

View Templates and View Managers

Describes the view template and view manager classes and operations for creating custom message browsers in both Telewindows Next Generation and classic G2.

Introduction **364**

Template View Toolbar Buttons **365**

GEVM View **368**
- View Template Configuration **368**
- View Template Class and Operations **371**
- View Template Button Class and Operations **380**
- View Manager Class and Operation **386**

GQSV View **391**
- View Template Configuration **391**
- View Template Classes and Operations **397**
- View Template Button Classes **422**
- View Manager Class and Operations **428**

GQS View Manager Class and Operations **435**
- gqs-view-manager **436**
- gqs-view-manager::gqs-update-view-per-addition **438**
- gqs-view-manager::gqs-update-view-per-attribute **439**
- gqs-view-manager::gqs-update-view-per-delete **440**
- gqs-view-manager::gqs-update-view-per-removal **441**
- gqs-create-view **442**
- gqs-deregister-view **443**
- gqs-register-view **444**
- gevm-refresh-view **445**

Introduction

The previous chapters described the core functionality for creating events and grouping them into queues. This chapter describes how to display the contents of a queue, typically in a tabular view, though other representations could be developed.

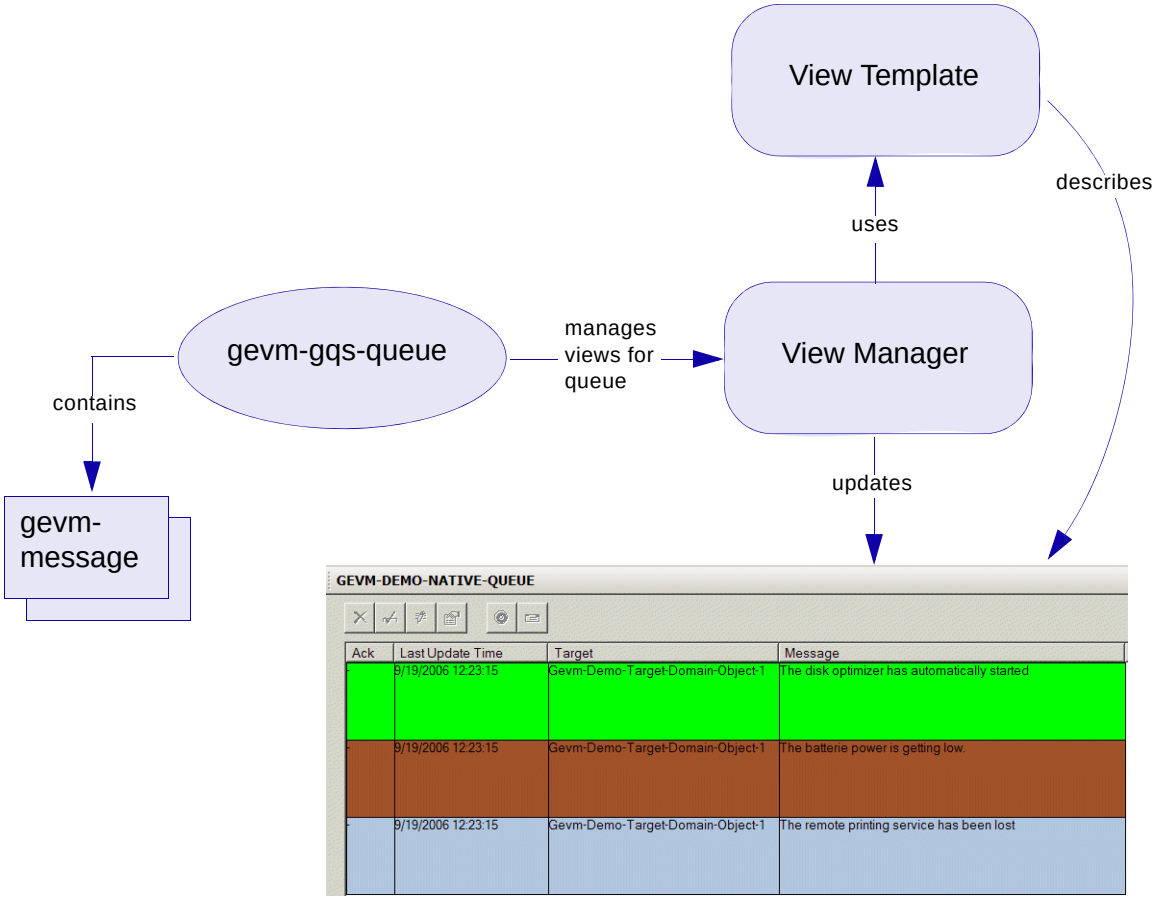
To display the contents of a queue, you must configure the following three components, one of which you must configure and two of which are created dynamically at runtime:

- A view manager template, which describes the layout of the view, either visually or descriptively. You typically configure the view manager template at design time.
- A view manager, which is created dynamically on the server for each view when calling gevm-launch-view to display the browser. The view manager is initialized based on the view manager template and receives update notifications from the queue when changes occur. The view manager updates the visual representation of the view, based on the events, for example, updating the messages in the tabular view.
- The view itself, typically called a message browser, which is typically a dialog or dockable pane with action buttons and a tabular view that displays the messages. The view can contain predefined buttons, or you can create custom buttons.

This chapter describes:

- A view manager template (gevm-native-view-manager-template) and view manager (gevm-native-view-manager) that supports displaying browser views in both Telewindows Next Generation using native controls, and in G2 or Telewindows using a G2 classic user interface based on GXL and UIL. The native view manager template supports custom native action buttons (gevm-native-action-button).
- A view manager template (gqsv-tabular-view-template) and view manager (gqsv-tabular-view-manager) that only supports displaying browser views in a G2 classic user interface based on GXL and UIL. The view manager template supports custom action buttons (gqsv-toolbar-button).
- The root class for all view managers (gqs-view-manager) and its associated operations.

The following diagram summarizes the relationship between events/messages, queues, view templates, view managers, and browser views:



Template View Toolbar Buttons and Popup Menus

You can delete, move, and add toolbar buttons to view templates. The specification for a native view and a G2 classic view are different, though the specification of a G2 classic view may be derived from a native view specification. You can also create customized buttons by subclassing them and defining their public methods.

This table describes the available action buttons for message browsers:

Button Key	Description
Acknowledge	Acknowledges selected message(s). Enabled if message has not been acknowledged, requires acknowledgment, and for users who have the permission to acknowledge messages in grtl-user-preferences configuration.
Change Priority	Changes the priority of the message.
Close	Closes the browser window.
Configure Filters	Configures the user message view filter. Opens a dialog to configure the filter. Always enabled.
Delete	Deletes selected message(s) in server and therefore in browser(s) as well. Enabled for some classes of messages and for users who have the permission to delete messages in the grtl-user-preferences configuration.
Edit Comment	Edits the comment of message. Opens a dialog to edit the comment of the message and then sets it in the server. Enabled for all gevm-message instances and its subclasses.
e-mail Message	Provides a framework for opening ticket for a message in a ticketing system.
Enable Filters	Enables/disables the message view filter, using a toggle button. Always enabled.
Line Break	Inserts a line break.
Lock View	Locks/unlocks the grid view, using a toggle button. When the view is locked, new messages are not added to the view. Locks the view and does not update the browser until unlocked.
Ping	Pings the computer associated with a message. Currently, pingmgr.kb must be merged in for this button to function properly.
Run Action	Executes an action on target object of the selected message.
Run Detection Logic	Runs detection logic on the target object of the selected message.
Run Response Logic	Runs response logic on the target object of the selected message.

Run Test Logic	Runs test logic on the target object of the selected message.
Send to Queue	Sends the selected message to a specified queue.
Separator	Inserts a separator between the buttons.
Show Action History	Shows the history of state changes for a message, for example, the repetition count, acknowledged status, or priority. The history includes a list of timestamps and symbols that specify when the message was created, when it was acknowledged, when the priority changed, and so on.
Show Details	Displays the property dialog of selected message(s). All message classes share a common set of properties, while some add additional properties.
Show Initiator	Displays the initiator of the message. Enabled if an initiator exists for the message.
Show Target	Displays the target of the message. Enabled if a target item exist.
Target Availability	Changes the availability of the target item associated with the selected messages. By putting a device off line or in repair state, messages for that device can be filtered out, for example, based on its availability.
Target Information	Displays the value of the system-information-or-best-practice-url attribute of the grtl-domain-object associated with a message.
Telnet	Opens a Telnet session by executing the command in the telnet-shell-command attribute of the active gevm-user-preferences object.
Trace Route	Traces the network route from the G2 server to the target associated with the selected message.
Custom	Custom button with API on server to query if message should be enabled or disabled for selected message(s) and API that implements behavior. You must provide iconic image for the button.

GEVM View

The GEVM tabular view template supports describing both native views in Telewindows Next Generation and views to be displayed in classic G2 and Telewindows. This template enables you to select one template when configuring a queues or access table, and leave the details of the presentation to the template for each type of user interface.

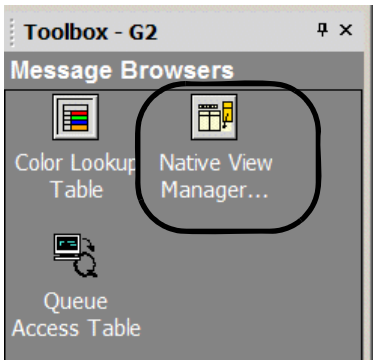
The G2 classic browser view can be derived and auto-generated from the description of the native browser view. This enables you to quickly generate a template that you can then adjust manually. You access both functionalities from the properties dialog of the GEVM native view manager template.

For information on customizing the G2 classic views, see "View Template Configuration" on page 391 for gqsv-tabular-view-template.

View Template Configuration

To create and configure a native view template:

1 In Developer mode, choose View > Toolbox - G2 and display the Message Browsers palette:



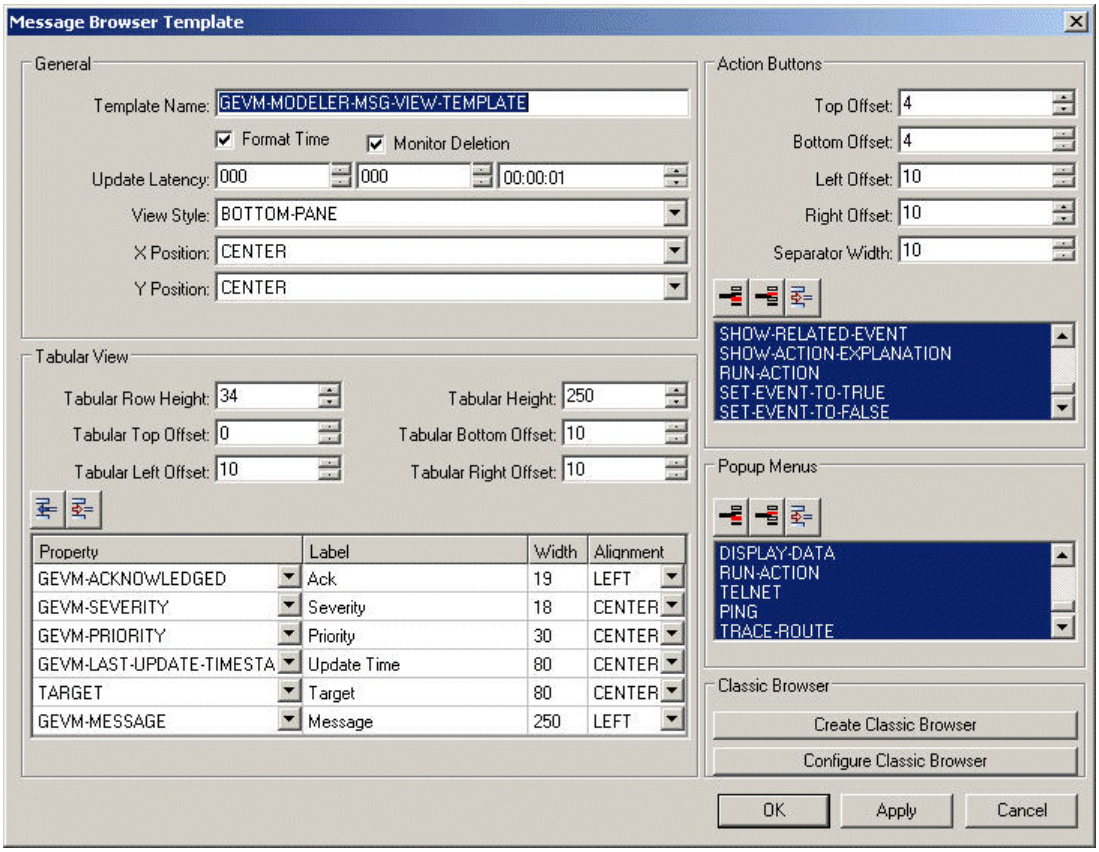
2 Create one or more Native View Manager Templates, one for each user mode.

Each view template determines the layout and behavior in a particular user mode.

3 Display the properties dialog for the Native View Manager Template and configure its properties.

You configure the layout of the message browser that appears in the Telewindows client in the view manager template. This dialog specifies the layout of native views in Telewindows. Based on this specification you can also generate the specification for a G2 classic user interface and access its template.

Here is the properties dialog of the browser view template:



- General properties

  By default, timestamps are formatted as intervals, such as 11/27/06 12:00:00. You can disable the Format Time option to make the timestamp be the number of seconds from the last time G2 was started.

  By default, the message browser automatically deletes messages associated with items that have been deleted. You can disable the Monitor Deletion option to cause messages to remain in the browser even when the associated item is deleted.

  By default, the message browser updates every 1.0 seconds. For applications that receive many messages, you can increase the Update Latency to batch updates and improve performance.

  By default, when viewed through Telewindows, the message browser is displayed as a docked pane at the bottom of the window, which you can auto-hide. You can also configure the View Style to be an MDI document, a modal dialog, a floating pane, or a docked pane in another location.

When displayed as a dialog, document, or floating pane, by default, the view is positioned in the center of the overall window. You can configure the X Position and Y Position to position the browser in a different location when it is initially launched. For information on the various positioning options, see the description of the tabular-view dialog control in the *G2 Reference Manual*.

- Action Button properties

  To configure the action buttons to appear in the view, click an existing button in the list of action buttons, then delete the button or insert a new button before or after the selected row. A list of available buttons appears for inserting in the list. You can also configure the top, bottom, left, and right offsets for the buttons, and the separator width.

- Popup Button properties

  To configure the action buttons to appear in the popup menu of the view, click an existing button in the list of action buttons, then delete the button or insert a new button before or after the selected row. A list of available buttons appears for inserting in the list.

- Tabular View properties

  To configure the tabular view containing the messages, click an existing property, then delete the row or insert a new row before above the selected row. Click the dropdown list to display a list of available message attributes for inserting. You can also configure the column label, initial width, and alignment of each column, as well as the row height, overall height of the tabular view, and top, bottom, left, and right offsets for the tabular view.

4. To configure the classic browser, in the properties dialog for the Native View Manager Template, click the Create Classic Browser button to create the browser template, then click the Configure Classic Browser button.

   For details, see "View Template Configuration" on page 391.

View Template Class and Operations

Class

gevm-native-view-manager-template on page 372

Method

gevm-native-view-manager-template::grtl-show-properties on page 379

gevm-native-view-manager-template

The view template class that describes the layout of both native browsers displayed in Telewindows Next Generation and the G2 classic browsers displayed in G2 or Telewindows.

Class Inheritance Path

gevm-native-view-manager-template, grtl-item, gqsv-tabular-view-template, object, item

Attributes

Attribute	Description
user-selectable-browser	When false, used internally to exclude templates from the list of template choices in dialogs, such as user preferences.
Allowable values:	truth-value
Default value:	true
action-button-names	The list of buttons to display in the native browser. The symbolic names in this sequence must match the button-key attribute of a gevm-native-action-button.
Allowable values:	sequence of symbols
Default value:	sequence (the symbol delete, the symbol acknowledge, the symbol show-details, the symbol show-target, the symbol show-initiator)
popup-menu-action-button-names	The list of buttons to display in the native browser popup menus. The symbolic names in this sequence must match the button-key attribute of a gevm-native-action-button.
Allowable values:	sequence of symbols

Attribute	Description
Default value:	sequence (the symbol delete, the symbol acknowledge, the symbol show-details, the symbol show-target, the symbol show-initiator)
server-converts-timestamps-to-text	When true, formats timestamps into text. The logic will use the timestamp formatting specified in the active grtl-module-settings.
Allowable values:	truth-value
Default value:	true
update-latency	The time, in seconds, from when new messages are added to the queue or their contents is updated. You configure the update latency to delay updating the message browser and batch changes in a single update.
Allowable values:	Any positive float
Default value:	1.0
tabular-view-column-specifications	The specifications of the columns of the tabular view that shows messages. Each entry in the sequence should be a structure with the following syntax: structure (property-name: *symbol*, column-header-label: *text*, {optional} initial-width: *integer)*
Allowable values:	A sequence of structures

Attribute	Description
Default value:	sequence (structure (property-name: the symbol gevm-acknowledged, initial-width: 50), structure (property-name: the symbol gevm-last-update-timestamp, initial-width: 100), structure (property-name: the symbol target, initial-width: 100), structure (property-name: the symbol gevm-message, initial-width: 300))
tabular-view-row-height	The height, in dialog units, of each row in the tabular view displaying messages. Specifying zero as the row height means the tabular view should use the default value, used by the Windows control.
Allowable values:	Any positive integer
Default value:	28
tabular-view-height	The total height, in dialog units, of the tabular view displaying messages.
Allowable values:	Any positive integer
Default value:	252
tabular-view-top-offst	The top offset of the tabular view, in dialog units.
Allowable values:	Any positive integer
Default value:	10

Attribute	Description
tabular-view-bottom-offset	The bottom offset of the tabular view, in dialog units.
Allowable values:	Any positive integer
Default value:	10
tabular-view-left-offset	The left offset of the tabular view, in dialog units.
Allowable values:	Any positive integer
Default value:	10
tabular-view-right-offset	The right offset of the tabular view, in dialog units.
Allowable values:	Any positive integer
Default value:	10
button-top-offset	The top offset between the top of the browser and the row of action buttons, in dialog units.
Allowable values:	Any positive integer
Default value:	4
button-bottom-offset	The bottom offset in dialog units between the row of action buttons and the tabular view, in dialog units.
Allowable values:	Any positive integer
Default value:	4
button-left-offset	The left offset in dialog units of the row of action buttons, in dialog units.

Attribute	Description
Allowable values:	Any positive integer
Default value:	10
button-right-offset	The right offset in dialog units of the row of action buttons, in dialog units.
Allowable values:	Any positive integer
Default value:	10
separator-button-width	The width, in dialog units, for a separator button between buttons in the row of action buttons.
Allowable values:	Any positive integer
Default value:	15
sorting-enabled	When true, enables the user to sort messages in the tabular view by clicking column headers.
Allowable values:	truth-value
Default value:	true
dialog-view-x-position	The x position of the browser dialog. Some of the settings are only relevant if displayed as a floating window and not in a pane.
Allowable values:	One of the following symbols: desktop-center, desktop-left, desktop-right, working-area-center, working-area-left, working-area-right, near-desktop-left, near-desktop-right, near-left, near-right, near-working-area-left, near-working-area-right, center, left , right

Attribute	Description
Default value:	center
dialog-view-y-position	The y position of the browser dialog. The settings are only relevant if displayed as a dialog.
Allowable values:	One of the following symbols: desktop-center, desktop-left, desktop-right, working-area-center, working-area-left, working-area-right, near-desktop-left, near-desktop-right, near-left, near-right, near-working-area-left, near-working-area-right, center, left , right
Default value:	center
view-style	Indicates how to display the browser dialog: as a top level dialog, as a MDI child document view, or as in a pane.
Allowable values:	One of the following symbols: dialog, document, left-pane, right-pane, top-pane, bottom-pane, or floating-pane
Default value:	bottom-pane
icon	The icon of the browser displayed in the caption bar of the dialog view or pane view.
Allowable values:	symbol
Default value:	none
neighbor-window-name	The name of a neighbor docking window.
Allowable values:	symbol

Attribute	Description
Default value:	none

gevm-native-view-manager-template::grtl-show-properties

Synopsis

grtl-show-properties
(*itm*: class gevm-native-view-manager-template, *win*: class ui-client-item)
–> *success*: truth-value

Argument	Description
itm	The view manager template.
win	The client.

Return Value	Description
success	Whether the properties dialog displayed successfully.

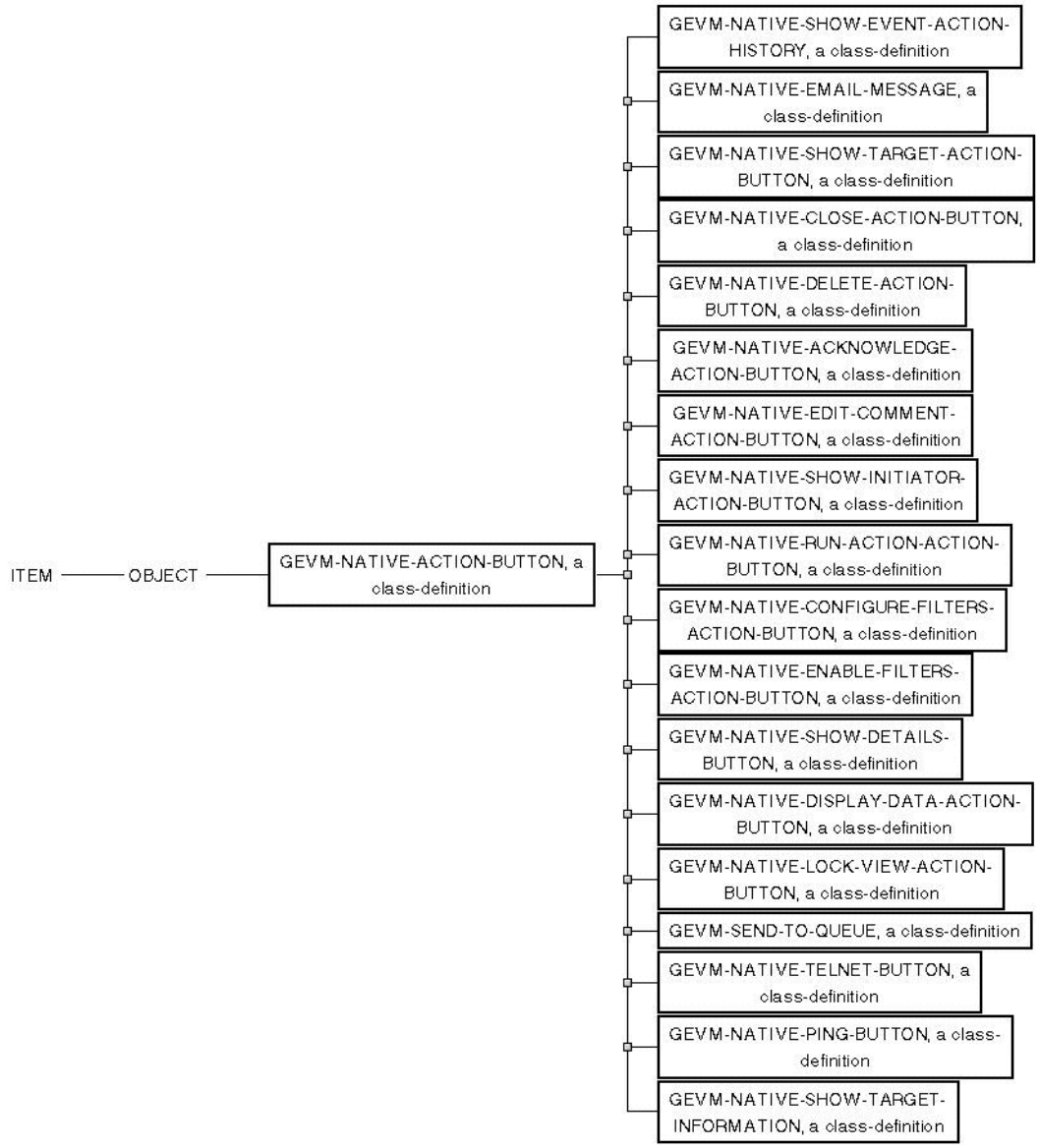
Description

Opens the properties dialog of a view manager template.

View Template Button Class and Operations

This section describes the button classes used on native views. These buttons apply to both the buttons used on the tabular view templates and those used on the detail views. It also describes two API procedures for managing these action buttons.

The root class of all native action buttons is gevm-native-action-button. The following class hierarchy describes the available subclasses in GEVM:



GEVM uses the following methods to query the state of buttons as the selection of messages changes and to perform actions:

- gevm-check-button-selection-state on page 384
- gevm-perform-button-function on page 385

This section does not include descriptions of the actions performed when the buttons are used. Instead, it describes the information you need when subclassing and customizing buttons. For a description of each button, see "Template View Toolbar Buttons and Popup Menus" on page 365.

Note that for the native buttons to work correctly in browsers, you need to create one instance of your custom subclasses. This single instance is used to perform the action and query the status. It is the glue between the G2 server logic and the corresponding buttons in the different browsers. For an example of a custom button, see `gevm-demo.kb`.

gevm-native-action-button

Class Inheritance Path

gevm-native-action-button, object, item

Attributes

Attribute	Description
button-key	A unique key for the button, which is also used to query for the labels in the GRTL text localization.
Allowable values:	symbol
Default value:	g2
show-label	When true, displays the label in the button.
Allowable values:	truth-value
Default value:	false
show-icon	When true, displays the icon in the button. The icon is the icon defined for the gevm-native-action-button class.
Allowable values:	truth-value
Default value:	true
is-toggle-button	When true, configures the button to be a toggle button.
Allowable values:	truth-value
Default value:	false

Attribute	Description
dismiss-button	When true, configures a button to dismiss the browser upon activation.
Allowable values:	truth-value
Default value:	false

gevm-check-button-selection-state

Synopsis

gevm-check-button-selection-state
(*action-button*: class gevm-native-action-button,
view-manager: class gevm-native-view-manager, *count-of-messages*: integer,
selected-messages: sequence, *win*: class ui-client-item)
–> *enabled*: truth-value, *pressed:* truth-value)

Argument	Description
action-button	The button to check.
view-manager	The view manager of the view that contains the button.
count-of-messages	
selected-messages	
win	The client.

Return Values	Description
enabled	True if enabled, false if disabled.
pressed	True if pressed, false if not pressed.

Description

Called by GEVM to query the selection state of the button. This method returns two truth-values that indicate whether the button is available and whether it is pressed.

gevm-perform-button-function

Synopsis

gevm-perform-button-function
(*action-button*: class gevm-native-action-button,
view-manager: class gevm-native-view-manager,
selected-messages: sequence, *win*: class ui-client-item)

Argument	Description
action-button	The action button to execute.
view-manager	The view manager of the view that contains the button.
selected-messages	
win	The client.

Description

Called by GEVM to perform the action of the button.

View Manager Class and Operation

Class

gevm-native-view-manager on page 387

Procedure

gevm-refresh-view on page 390

gevm-native-view-manager

Class Inheritance Path

gevm-native-view-manager, gqs-view-manager, object, item

Attributes

Attribute	Description
manager-template-name	The name of the gevm-native-view-manager-template that was used to create the native view manager instance. This attribute is read-only and should not be changed.
Allowable values:	symbol
Default value:	g2
browser-key	A unique key for each manager. This attribute is read-only and should not be changed.
Allowable values:	text
Default value:	""
action-button-names	Copied from the corresponding attribute in the gevm-native-view-manager-template. This list is used at runtime, is read-only, and should be changed.
Allowable values:	sequence

Attribute	Description
Default value:	sequence (the symbol delete, the symbol acknowledge, the symbol show-details, the symbol show-target, the symbol show-initiator)
popup-menu-action-button-names	Copied from the corresponding attribute in the gevm-native-view-manager-template. This list is used at runtime, is read-only, and should be changed.
Allowable values:	sequence of symbols
Default value:	sequence (the symbol delete, the symbol acknowledge, the symbol show-details, the symbol show-target, the symbol show-initiator)
view-locked	When true, the view will not be updated as new events are available or events are updated.
Allowable values:	truth-value
Default value:	false
server-converts-timestamps-to-text	Copied from the corresponding attribute in the gevm-native-view-manager-template. This attribute is read-only and should not be changed.
Allowable values:	truth-value
Default value:	true
update-latency	Copied from the corresponding attribute in the gevm-native-view-manager-template. Specifies the update latency to batch browser updates.
Allowable values:	Any positive float

Attribute	Description
Default value:	1.0
filter-enabled	When true, enables the filtering within the browser. Toggled by the action button in the browser to enable and disable filtering.
Allowable values:	truth-value
Default value:	false
filter	The browser filter specification.
Allowable values:	Instance of gevm-filter
Default value:	

gevm-refresh-view

Synopsis

gevm-refresh-view
(*manager*: class gevm-native-view-manager, *win*: class ui-client-item)

Argument	Description
manager	The view manager whose view to refresh.
win	The client.

Description

Refreshes the view. It reevaluates all items in the queue, applying the filter, if appropriate, and inserts them into the tabular display.

GQSV View

The GQSV tabular view supports displaying browser views based on the G2 classic user interface, using GXL and UIL. The following sections describe how to configure classic view templates, and the various classes and available operations.

View Template Configuration

You configure view templates to be displayed in the G2 classic user interface by first creating and configuring a native view template, then creating and customizing the resulting classic view template.

Note You can configure various elements of a classic view template, as described below. However, the customizations only affect the classic view; they are not reflected in the view displayed in Telewindows Next Generation.

You can configure these elements of the view template:

- The columns that contain values for the attributes of the entries and the associated column headers.
- The number of lines for each entry.
- The font size for entry text.
- Sorting options, including the column on which to sort, the sort order, whether new entries are to be sorted with existing entries, and whether clicking on a column header causes entries to be sorted.
- The view label, including the text and colors.
- The view colors.
- The toolbar buttons, including the attributes and location.

For example, you might want the default template to display more than the default number of rows, or you might want to include additional columns to display other queue entry attributes.

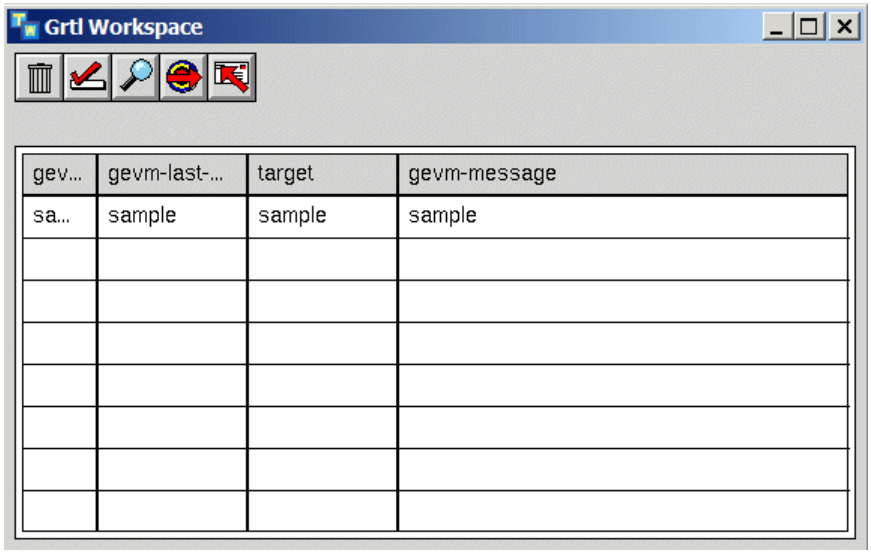
You can cancel changes you make to the table (columns) in the layout. You cannot cancel other changes you make to the layout, such as buttons, although you can change them back to their original state. The changes you make to the default queue view layout take effect the next time you display a queue; currently visible queue views are not affected.

Creating a View Template

To create a view template:

1. Create and configure a native view template, as described in "View Template Configuration" on page 368.
2. Click the Create Classic Browser button.
3. Click the Configure Classic Browser button, then close the dialog to reveal the classic view.

   For example, here is the classic view template created from the default native view template:



   The following sections describe how you can configure the classic view. Note that some of the actions require that you be in Administrator mode.
4. When you are finished configuring the view as described in the following sections, choose Finished Configuration from the popup menu on the view or choose Cancel Configuration to cancel the changes.

Modifying View Colors and Border

You can modify the background and foreground colors of the default queue view by configuring the workspace colors:

- The background is the gray area.
- The foreground controls the color of the column header text, the border around the column header and between each header, the border around the table, and the label text, if it was not explicitly changed.

To configure overall view colors:

1. Display the properties dialog for the view.
2. Configure the Background Color and Foreground Color.
3. Configure the Workspace Margin.

Modifying Template View Action Buttons

To delete an existing button, right click the button and choose delete. To move an existing button to a new location, select it and drag it with the mouse.

To add predefined buttons, configure the buttons in the native view template.

You can also create customized buttons by subclassing them and defining their public methods. See "View Template Button Classes" on page 422.

Modifying Template View Columns

You can modify these characteristics of columns:

- The columns that appear in the view.
- The order of the columns.
- The font used to display the text in the columns.
- The background, text, and border colors of the column cells.
- The number of rows visible in the view.
- The height of each cell in a column.
- The type of data that can be displayed in the column.
- Whether cells can be selected or edited.
- Whether clicking on a column header sorts entries.
- The procedure that validates a user-entered value.
- Callback procedures that execute when a value is entered in a cell or when the cell is selected.
- How columns format floating point values, dates and times, and ordinal numbers.

To modify the column header label and contents, as well as other column header attributes, see "Modifying Column Headers".

To configure the columns in the view:

1. To insert a new column before or after an existing column, select the column header of the column before or after which you want to add the new column header to display its menu.
2. Right click the desired column to display the popup menu and choose either add new column after or add new column before.
3. To delete a column from the view, right click the desired column to display the popup menu, choose the column header of the column you want to delete, and choose delete column.
4. To move a column in the view, while the pointer is on the column header of the column you want to move, click and hold the mouse button, drag the column header until it appears where you want the new location to be, and release the mouse button.

To configure the column properties:

1. Right click the desired row cell except the header to display the popup menu and choose table.
2. To change the number of visible rows, edit the number-of-visible-rows attribute.
3. To change the font size of text displayed in a column, edit the font-size attribute to be small, large, and extra-large.
4. To change the colors of a column, edit the default-background-color, default-text-color, or default-border-color attribute.
5. To modify the height of all columns, edit the cell-height attribute.
6. To modify the width of a column, edit the width attribute.
7. To specify whether clicking on a column header sorts entries, edit the allow-click-to-sort attribute.

   The default is yes, which sorts the entries when the user clicks on the column header.
8. To specify whether a user can edit values in the column cells, edit the cells-are-editable attribute.

   The default is false, which means that the end user cannot edit the values in the cells. Editing the value in a cell modifies the current queue entry.
9. To configure the float formatter for a column, click the float-format attribute and choose subtable, and edit the attributes of the subtable according to the table above.

   By default, all cells that show floating point numbers remove the trailing zeros and show 4 decimal places.

All column cells format floating point numbers by providing an instance of a gxl-float-formatter as the value of the float-format attribute of the column. You can customize these aspects of floating point numbers:

- Whether default formatting is used.
- The minimum number of characters in the formatted numbers.
- The number of significant digits, or the number of decimal places.
- How the value is represented.
- Whether to display zeros to the right of the last nonzero digit.

For more information about this object and its attributes, see the *G2 XL Spreadsheet User's Guide*.

10 To specify whether a column cell can be selected, edit the cells-are-selectable attribute.

11 To customize the procedure that formats the contents of the selected column, define a procedure with the desired behavior with this signature, and specify this procedure as the key-value-conversion-procedure attribute:

```
my-conversion-procedure
(item: class item, attribute: symbol)
–> (value: value)
```

GEVM provides a convenience procedure named gevm-get-event-attribute-for-gqs. This procedure formats GEVM message attributes, such as timestamps, based on the desired format specified in GRTL.

12 To customize the color formatter procedure for a column header, define a procedure with the desired behavior with this signature, and specify this procedure as the dynamic-color-formatter attribute:

```
my-color-formatter-procedure
(entry: class gevm-event, attribute: symbol)
–> (background-color: symbol, border-color: symbol, text-color: symbol)
```

where:

- *entry* is the queue entry whose cell color is to be formatted.
- *attribute* is the queue entry attribute whose cell color is to be formatted.

This procedure returns these values:

- *background-color* is the background color of the cell.
- *border-color* is the border color of the cell.
- *text-color* is the text color of the cell.

GEVM provides a default convenience method named gevm-get-event-color. GEVM provides a default implementation for gevm-event, but it can be

customized for subclasses to implement specific behavior. This procedure relies on the color lookup tables and the message color mode specified in the active gevm-module-settings object to determine the text and background color.

Modifying Column Headers

You can configure various characteristics of column headers, including the header label text, font size, or colors, to modify characteristics of the column headers and of the data contained in the columns. You can also specify whether a column value is to be monitored.

To modify column headers:

1. Right click the header you want to modify and choose table.
2. To configure the attribute to appear in the column, configure the attribute-or-key attribute.

   For a list of available queue entry attributes, see "Event Classes" on page 90.
3. To modify the text of the header label, configure the visible-label attribute.
4. To modify the font size of the label, configure the font-size attribute.
5. To modify the colors of the label, choose default-background-color, default-text-color, or default-border-color and edit the color by entering the name of the color.

   To obtain a list of available colors, click the background of the view and choose color > background-color, then specify one of these colors as a symbol, for example, forest-green.
6. To indicate that a column value is to be monitored, configure the monitor-this-attribute attribute to be yes.

You can write a procedure that monitors the value in a column. Each column header has an attribute that specifies whether the value of the column is to be monitored. For new columns added to the view layout, the value of this attribute is yes. In general, the value should be yes only for columns whose values can change.

View Template Classes and Operations

Classes

gqsv-tabular-view-template on page 398
gqsv-column on page 400
gqsv-column-header on page 405
gqsv-column-or-header on page 409
gqsv-root-specification on page 410
gqsv-view-configuration on page 412
gqsv-workspace-location on page 415

Procedures

gqsv-close-tabular-view on page 418
gqsv-delete-view on page 419

Functions

gqsv-number-of-viewed-items on page 420
gqsv-view-is-locked on page 421

gqsv-tabular-view-template

This object defines the layout and behavior of a queue view.

Class Inheritance Path

gqsv-tabular-view-template, object, item

Attributes

Attribute	Description
gqsv-view-manager-class	The class of view manager associated with this view.
Allowable values:	Any symbol
Default value:	GQSV-TABULAR-VIEW-MANAGER
gqsv-master-dialog	If the view is to be created on a UIL dialog or any other special workspace, specify the dialog ID here, or specify the name of the superior item of the target workspace. When you launch the view, GQS either reserves the appropriate dialog and builds the view on the dialog, or clones the subworkspace of the named superior item, and builds the view on the cloned workspace.
Allowable values:	Any value
Default value:	UNSPECIFIED
gqsv-initial-view-location	The location in the host window, where the workspace of the view is to be shown. See gqsv-workspace-location on page 415 for details.
Allowable values:	Any gqsv-workspace-location
Default value:	GQSV-WORKSPACE-LOCATION

Attribute	Description
gqsv-monitor-deletion-events-on-visible-items	Controls if the view sets up monitors on the deletion of items visible in the view. • When this attribute is true, the view automatically updates if one of the displayed items is deleted. • If the underlying queue is already monitoring deletion events, this attribute should be false. Monitoring deletion events for visible items only is more efficient than monitoring all items in the queue, since only a small subset of items in the queue may be visible at any one time.
Allowable values:	Any truth-value
Default value:	true

gqsv-column

The purpose of this class is to enable users to configure the tabular view. It does not have a persistent life of its own once the view template is closed and should not be referenced as an API. None of the attributes of the class should be accessed programmatically.

This class lets you configure the properties of the cells in a column on a queue view. An instance of this class appears below each column header in the view when you interactively configure a gqsv-tabular-view-template. You set the desired properties of the cells by editing the attributes of the gqsv-column, through its table. See gqsv-column-header on page 405 and gqsv-view-configuration on page 412.

Class Inheritance Path

gqsv-column, gqsv-column-or-header, gqsv-configuration-message, message, item

Attributes

Attribute	Description
number-of-visible-rows	The number of rows to display in the view. This property is common to all columns. That is, editing this property in one of the columns on a table also changes it for all of the other columns.
Allowable values:	Any positive integer
Default value:	5
cell-height	The cell height, in number of pixels.
Allowable values:	Any positive integer
Default value:	28

Attribute	Description
cell-type	The type of cell used in the column. Specify the most specific cell type consistent with the values to be displayed. For any attribute displayed on the built-in queue views, it is probably not appropriate to change this value from the default.
Allowable values:	VALUE-CELL, INTEGER-CELL, FLOAT-CELL, QUANTITY-CELL, TEXT-CELL, SYMBOL-CELL, TRUTH-VALUE-CELL
Default value:	VALUE-CELL
initialization-data	A value passed to the initialization procedure for the cell. Normally left as UNSPECIFIED.
Allowable values:	Any value
Default value:	UNSPECIFIED
cells-are-selectable	A truth-value indicating if clicking on the cell should select (highlight) the cell.
Allowable values:	Any truth-value
Default value:	true
cells-are-editable	A truth-value indicating if the user should be able to interactively edit the contents of the cell.
Allowable values:	Any truth-value
Default value:	false

Attribute	Description
validation-procedure	The name of an optional user-written procedure that validates user input, if the cells in the column are editable. The signature of your validation method must be: my-validation-proc (*Sheet*: class gxl-spreadsheet, *Row*: integer, *Col*: integer, *NewValue*: value, *Win*: class g2-window) –> (*ValueOrErrorText*: value, *Accepted*: truth-value) Note that basic type checking is done automatically based on the cell type and does not require a validation procedure. See the *G2 XL Spreadsheet User's Guide* for more information. *Allowable values:* Any symbol *Default value:* UNSPECIFIED
callback-procedure	The name of an optional user-written procedure that is called when the cell receives a new value. This callback might be used if you want view-specific behavior to be executed when a view is updated. Note that the gqsv-column-header contains an attribute for updating cell colors. The signature of your validation method must be: my-callback (*Sheet*: class gxl-spreadsheet, *Row*: integer, *Col*: integer, *NewValue*: value, *Win*: class g2-window) See the *G2 XL Spreadsheet User's Guide* for more information. *Allowable values:* Any symbol *Default value:* GQSV-SET-ATTRIBUTE-CALLBACK

Attribute	Description
selection-callback-procedure	The name of an optional user-written procedure that is called when any cell in the column is selected. The signature of the selection callback you provide is: my-selection-callback (*Sheet*: class gxl-spreadsheet, *View*: class gxl-spreadsheet-view, *FirstRow*: integer, FirstCol: integer, *NumberOfRows*: integer, *NumberOfCols*: integer, *Win*: class g2-window). See the *G2 XL Spreadsheet User's Guide* for more information.
Allowable values:	Any symbol
Default value:	UNSPECIFIED
float-format	Edit the subtable of this attribute to change the formatting of floating point numbers shown in this column. See the *G2 XL Spreadsheet User's Guide* for more information.
Allowable values:	Any gxl-float-formatter
Default value:	GXL-FLOAT-FORMATTER
font-size	See gqsv-column-or-header on page 409.
Allowable values:	SMALL, LARGE, EXTRA-LARGE
Default value:	SMALL
default-background-color	See gqsv-column-or-header on page 409.
Allowable values:	Inherited
Default value:	WHITE

Attribute	Description
default-text-color	See gqsv-column-or-header on page 409.
Allowable values:	Inherited
Default value:	BLACK
default-border-color	See gqsv-column-or-header on page 409.
Allowable values:	Inherited
Default value:	BLACK

gqsv-column-header

The purpose of this class is to enable users to configure the tabular view. It does not have a persistent life of its own once the view template is closed and should not be referenced as an API. None of the attributes of the class should be accessed programmatically.

This class lets you configure the properties of a column on a queue view. An instance of this class appears at the top of each column of the view when you interactively configure a gqsv-tabular-view-template. You set the desired properties of each column by editing the attributes of the gqsv-column-header, through its table. Also see gqsv-view-configuration on page 412 and gqsv-column on page 400.

Class Inheritance Path

gqsv-column-header, gqsv-column-or-header, gqsv-configuration-message, message, item

Attributes

Attribute	Description
visible-label	The text that appears at the top of the column, which usually represents the property shown in the column.
Allowable values:	Any text
Default value:	"Any label"
attribute-or-key	The name of the attribute that is displayed in this column, or a symbolic key for the column if it is not directly displaying an attribute. In the latter case, the key given here is passed to the key-value-conversion-procedure.
Allowable values:	Any symbol
Default value:	UNSPECIFIED

Attribute	Description
key-value-conversion-procedure	The name of the procedure that provides values for the entries in the column. If the attribute-or-key names an attribute, the default for this attribute (gqsv-get-attribute-value) simply retrieves the value of the named attribute. If you want to specially format the values appearing in this column (for example, representing a float as a timestamp), or if you want to display a value that is not an attribute of the underlying item, you must provide your own key-value-conversion-procedure. The signature of your procedure must be: my-conversion-proc (*Item*: class item, *Key*: symbol) –> *return-value*: value *Key* is the attribute-or-key for this column, the *Item* an item in the queue, and the return value is the value to be displayed in the view.
Allowable values:	Any symbol
Default value:	GQSV-GET-ATTRIBUTE-VALUE
dynamic-color-formatter	The name of an optional procedure that provides color information for cells in this column. The signature of this procedure is: my-color-formatter-proc (*Item*: class item, *Key*: symbol) –> *background*: symbol, *text*: symbol, *border*: symbol) *Item* is the item in the queue, *Key* is the attribute-or-key for this column, and the return values are the desired background, text, and border color for the cell.
Allowable values:	Any symbol
Default value:	UNSPECIFIED
width	The width, in pixels, of this column.

Attribute	Description
Allowable values:	Any integer
Default value:	120
allow-click-to-sort	Controls whether clicking the column header causes the view to sort on the values in the column. If sorting of this column is not allowed, set this attribute to false.
Allowable values:	Any truth-value
Default value:	true
monitor-this-attribute	Setting this value to true indicates to the view that it should continuously monitor the attribute displayed in this column for changes. For the built-in views and the default attributes, this value is always set to false for two reasons. **1** First, most of the attributes are set only on entry creation, and can never change for a given entry. For example, a time stamp is the creation time of the entry and can never be expected to change. **2** Second, the attributes that are expected to change are designed to update the views without the overhead of attribute monitoring. If you created new columns, you should leave this attribute as true (the default) if you expect that value to be changing dynamically, and you want the dynamic changes to be updated in the view.
Allowable values:	Any truth-value
Default value:	true
font-size	See gqsv-column-or-header on page 409.

Attribute	Description
Allowable values:	SMALL, LARGE, EXTRA-LARGE
Default value:	LARGE
default-background-color	See gqsv-column-or-header on page 409.
Allowable values:	Inherited
Default value:	WHITE
default-text-color	See gqsv-column-or-header on page 409.
Allowable values:	Inherited
Default value:	BLACK
default-border-color	See gqsv-column-or-header on page 409.
Allowable values:	Inherited
Default value:	BLACK

gqsv-column-or-header

This is a non-instantiable class that is a parent for gqsv-column-header and gqsv-column, and contains attributes common to both classes, both used in view template configuration.

Class Inheritance Path

gqsv-column-or-header, gqsv-configuration-message, message, item

Attributes

Attribute	Description
font-size	Controls the font size in the view.
Allowable values:	SMALL, LARGE, EXTRA-LARGE
Default value:	SMALL
default-background-color	Controls the background color of the cells in the view, if the background color has not otherwise been set.
Allowable values:	Any symbol
Default value:	WHITE
default-text-color	Controls the text color of the cells in the view, if the text color has not otherwise been set.
Allowable values:	Any symbol
Default value:	BLACK
default-border-color	Controls the border color of the cells in the view, if the border color has not otherwise been set.
Allowable values:	Any symbol
Default value:	BLACK

gqsv-root-specification

Instances of this object reside on the subworkspace of any configured gqmv-tabular-view-template, along with other gxl-specification-objects. Normally, you would not access these objects directly, instead using the graphical configuration tools when you chose Configure Tabular View on the gqmv-tabular-view object.

See gqsv-column on page 400, gqsv-column-header on page 405, and gqsv-view-configuration on page 412.

You can, however, access these objects programmatically. For more information, see the *G2 XL Spreadsheet User's Guide*.

Class Inheritance Path

gqsv-root-specification, gxl-root-specification, gxl-specification-object, gfr-object-with-uuid, object, gfr-item-with-uuid, item

Attributes

Attribute	Description
gqsv-add-new-items-first-or-last	See gqsv-view-configuration on page 412.
Allowable values:	Any symbol
Default value:	LAST
gqsv-scroll-to-new-items	See gqsv-view-configuration on page 412.
Allowable values:	Any truth-value
Default value:	false
gqsv-automatically-resort-new-items	See gqsv-view-configuration on page 412.
Allowable values:	Any truth-value
Default value:	false

Attribute	Description
gqsv-automatically-resort-attribute-changes	Indicates whether or not an attribute change should cause the view to be re-sorted. For example, if a view is sorting on the acknowledged attribute, you may want entries that are acknowledged to be sent to the bottom of the view immediately upon acknowledgement. See also gqsv-view-configuration on page 412 for a description of this attribute.
Allowable values:	Any truth-value
Default value:	false
gqsv-key-for-column-to-sort-initially	Configured through the gqmv-tabular-view-template configuration dialog. See also "gqsv-view-configuration" on page 412 for a description of this attribute.
Allowable values:	Any symbol
Default value:	UNSPECIFIED
gqsv-initial-sorting-order	Configured through the gqmv-tabular-view-template configuration dialog. See also gqsv-view-configuration on page 412 for a description of this attribute.
Allowable values:	ASCENDING, DESCENDING
Default value:	ASCENDING

gqsv-view-configuration

The purpose of this class is to enable users to configure the tabular view. It does not have a persistent life of its own once the view template is closed and should not be referenced as an API. None of the attributes of the class should be accessed programmatically.

This class lets you configure the properties of a queue view. This class is the frame around the cells in the view facsimile shown when you interactively configure a gqsv-tabular-view-template. You set the desired properties of the view by editing the attributes of the gqsv-view-configuration. Show its table by clicking on the edge of the view facsimile. Also, see gqsv-column-header on page 405 and gqsv-column on page 400.

Class Inheritance Path

gqsv-view-configuration, gqsv-configuration-message, message, item

Attributes

Attribute	Description
add-new-items-first-or-last	Determines whether new items are added to the top or bottom of the view. Normally, adding new items last (at the bottom of the view) is more efficient than adding items first (to the top of the view), because screen redraws are needed less frequently.
Allowable values:	Any symbol
Default value:	LAST
scroll-to-display-new-items	Determines whether the queue view should automatically scroll to show any new items added to the view. For example, if this attribute is set to true, when items are added to the bottom of the view, the view scrolls to expose the last item in the view.
Allowable values:	Any truth-value
Default value:	false

Attribute	Description
automatically-resort-new-items	Determines if the view should be sorted according to the current sort column, whenever new items are added to the view. Note that there may be a substantial performance penalty for using this feature.
Allowable values:	Any truth-value
Default value:	false
automatically-resort-attribute-changes	Determines if the view should be sorted according to the current sort column whenever the associated attribute of any item in the queue receives a new value. This feature is only relevant if the queue is monitoring the attribute associated with the current sort column. Note that there may be a substantial performance penalty for using this feature.
Allowable values:	Any truth-value
Default value:	false
key-for-column-to-sort-initially	If the view is configured to sort or resort automatically, this attribute determines which column are sorted when the view is initially displayed. The column is specified by its attribute-or-key, given in the column-header. Note that the user can change the sort column interactively once the view is displayed by clicking on the column header.
Allowable values:	Any symbol
Default value:	UNSPECIFIED
initial-sorting-order	The order of the initial sorting of the view. Note that the user may be able to interactively change the sort order.

Attribute	Description
Allowable values:	ASCENDING, DESCENDING
Default value:	ASCENDING

gqsv-workspace-location

This item provides a generic way to specify the location and scale of a workspace within a g2-window. The workspace is shown by placing a focal point on the workspace (given by workspace-x-location and workspace-y-location) at the given point in the window (given by window-x-location and window-y-location). For an application of this class, see gqsv-tabular-view-template on page 398.

Class Inheritance Path

gqsv-workspace-location, object, item

Attributes

Attribute	Description
x-scale	The horizontal scale of the workspace, where 0.5 is half-size, 1.0 is full size, 2.0 is twice normal scale, etc.
Allowable values:	Any positive float
Default value:	1.0
y-scale	The vertical scale of the workspace, where 0.5 is half-size, 1.0 is full size, 2.0 is twice normal scale, etc.
Allowable values:	Any positive float
Default value:	1.0
window-x-location	This value can be either the symbol TOP, CENTER, or BOTTOM, or an integer indicating a horizontal coordinate in the window coordinates.
Allowable values:	Any value
Default value:	CENTER

Attribute	Description
window-y-location	This value can be either the symbol TOP, CENTER, or BOTTOM, or an integer indicating a vertical coordinate in the window coordinates.
Allowable values:	Any value
Default value:	CENTER
workspace-x-location	This value can be either the symbol TOP, CENTER, or BOTTOM, or an integer indicating a horizontal coordinate in workspace coordinates.
Allowable values:	Any value
Default value:	CENTER
workspace-y-location	This value can be either the symbol TOP, CENTER, or BOTTOM, or an integer indicating a vertical coordinate in workspace coordinates.
Allowable values:	Any value
Default value:	CENTER
auto-scale	When true, automatically scales the browser to fit into the screen. It uses the auto-scale-maximum-screen-usage-percentage to limit the screen coverage to a certain percentage. By default, it keeps the width/height ratio constant.
Allowable values:	A truth-value
Default value:	false
auto-scale-maximum-screen-usage-percentage	The maximum percentage of the screen for scaling the workspace.
Allowable values:	Any float
Default value:	0.9

Attribute	Description
auto-scale-to-full-screen	When true, and auto-scale is also true, then the the width/height ratio of the browser is ignored and the width and height are scaled independently to fit the screen.
Allowable values:	A truth-value
Default value:	false

gqsv-close-tabular-view

Synopsis

gqsv-close-tabular-view
(*manager*: gqsv-tabular-view-manager, *Win*: g2-window)

Argument	Description
manager	The manager for the view.
win	The client for this call.

Description

This procedure hides the workspace of the view associated with *manager* and releases the host dialog, if any, then deletes the view and its manager, and does related cleanup. The underlying queue is not affected by this procedure. See also gqsv-delete-view on page 419.

gqsv-delete-view

Synopsis

gqsv-delete-view
(*manager*: gqsv-tabular-view-manager, *client*: object)

Argument	Description
manager	The manager for the view.
client	The client for this call.

Description

This procedure deletes the view and its manager, and does related cleanup. The underlying queue is not affected by this procedure. See also gqsv-close-tabular-view on page 418.

gqsv-number-of-viewed-items

Synopsis

gqsv-number-of-viewed-items
(*manager*: class gqs-view-manager)

Argument	Description
manager	The view manager associated with a queue view.

Description

This function enables you to determine the number of items currently contained in a queue view. Because of view filtering and latencies, this is not always the same number of items contained by the underlying queue.

gqsv-view-is-locked

Synopsis

gqsv-view-is-locked
    (*view*: class item)

Argument	Description
view	A gxl-spreadsheet, gxl-spreadsheet-view, or gqs-view-manager.

Description

This function enables you to determine if a view is locked, given either the spreadsheet, spreadsheet view, or view manager associated with the view.

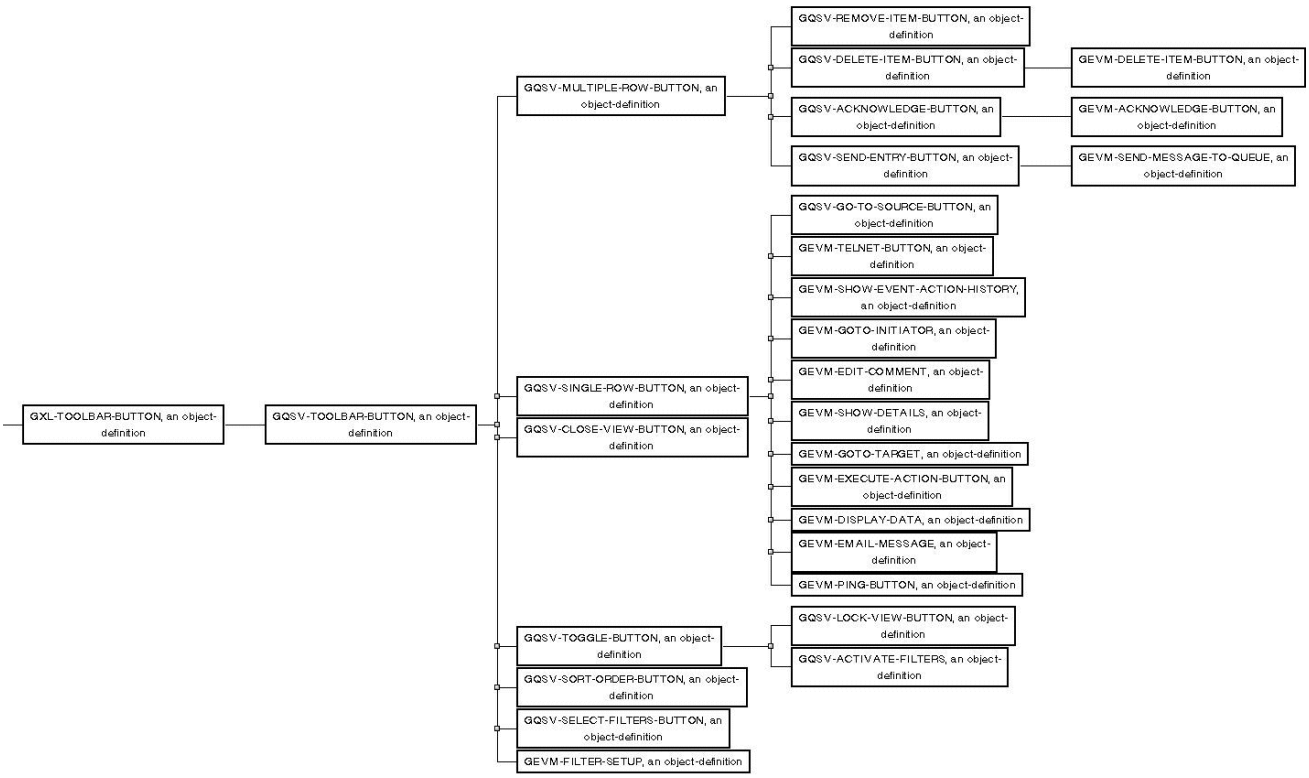
View Template Button Classes

This section describes the button classes used on the views and applies to both the buttons used on the tabular view templates and those used on the detail views. It also describes two API procedures for managing those action buttons.

This section does not include descriptions of the actions performed when the buttons are used. Instead, this section attempts to provide the information you need if you intend to subclass and customize buttons.

For a description of the buttons, see "Template View Toolbar Buttons and Popup Menus" on page 365.

The following class hierarchy describes the available buttons in GEVM:



The view template button classes are:

gqsv-close-view-button
gqsv-multiple-row-button
gqsv-single-row-button
gqsv-toolbar-button

gqsv-close-view-button

This button, which looks similar to the hide-workspace button, closes a queue view, deleting the view. It has the same effect as calling gqsv-close-tabular-view on page 418.

Class Inheritance Path

gqsv-close-view-button, gqsv-toolbar-button, gxl-toolbar-button, gxl-control-button, object, item

Attributes

Attribute	Description
gxl-help-text	See gqsv-toolbar-button on page 426.
Allowable values:	Inherited
Default value:	UNSPECIFIED
gxl-help-resource	See gqsv-toolbar-button on page 426.
Allowable values:	Inherited
Default value:	GQSV-TEXT-RESOURCES
gxl-button-state	See gqsv-toolbar-button on page 426.
Allowable values:	ENABLED, DISABLED, RELEASED
Default value:	RELEASED

Methods

gxl-control-button::gxl-reflect-selection-state-in-button
gxl-control-button::gxl-set-button-state
gxl-control-button::gxl-perform-button-function

gqsv-multiple-row-button

This class can be used as a parent class for custom buttons on queue views. It defines the methods gxl-set-button-state and gxl-reflect-selection-state-in-button so that the button is active when any row or rows are selected in the view. If you subclass from this class, your icon must include the regions icon-highlight and icon-symbol. See gqsv-single-row-button on page 425 and gqsv-toolbar-button on page 426 for alternative parent classes.

Class Inheritance Path

gqsv-multiple-row-button, gqsv-toolbar-button, gxl-toolbar-button, gxl-control-button, object, item

Attributes

Attribute	Description
gxl-help-text	See gqsv-toolbar-button on page 426.
Allowable values:	Inherited
Default value:	UNSPECIFIED
gxl-help-resource	See gqsv-toolbar-button on page 426.
Allowable values:	Inherited
Default value:	GQSV-TEXT-RESOURCES
gxl-button-state	See gqsv-toolbar-button on page 426.
Allowable values:	ENABLED, DISABLED, RELEASED
Default value:	RELEASED

Methods

gxl-control-button::gxl-reflect-selection-state-in-button
gxl-control-button::gxl-set-button-state
gxl-control-button::gxl-perform-button-function

gqsv-single-row-button

This class can be used as a parent class for custom buttons on queue views. This class defines the methods gxl-set-button-state and gxl-reflect-selection-state-in-button so that the button is active when exactly one row is selected in the view. If you subclass from this class, your icon must include the regions icon-highlight and icon-symbol. See gqsv-multiple-row-button on page 424 and gqsv-toolbar-button on page 426. for alternative parent classes.

Class Inheritance Path

gqsv-single-row-button, gqsv-toolbar-button, gxl-toolbar-button, gxl-control-button, object, item

Attributes

Attribute	Description
gxl-help-text	See gqsv-toolbar-button on page 426.
Allowable values:	Inherited
Default value:	UNSPECIFIED
gxl-help-resource	See gqsv-toolbar-button on page 426.
Allowable values:	Inherited
Default value:	GQSV-TEXT-RESOURCES
gxl-button-state	See gqsv-toolbar-button on page 426.
Allowable values:	ENABLED, DISABLED, RELEASED
Default value:	RELEASED

Extensible Methods

gxl-control-button::gxl-reflect-selection-state-in-button
gxl-control-button::gxl-set-button-state
gxl-control-button::gxl-perform-button-function

gqsv-toolbar-button

Use this class as the parent for custom buttons you want to include on queue views. You add functionality to your custom button by defining the methods gxl-perform-function, gxl-reflect-selection-state-in-button, and gxl-set-button-state. If you subclass from this class, your icon must include the regions icon-highlight and icon-symbol. For more details, see the *GXL User's Guide*. Alternatively, you can subclass gqsv-single-row-button on page 425 or gqsv-multiple-row-button on page 424, which define the button activation behavior based on the number of rows selected.

Class Inheritance Path

gqsv-toolbar-button, gxl-toolbar-button, gxl-control-button, object, item

Attributes

Attribute	Description
gxl-help-text	See gxl-control-button in the *G2 XL Spreadsheet User's Guide*.
Allowable values:	Inherited
Default value:	UNSPECIFIED
gxl-help-resource	See gxl-control-button in the *G2 XL Spreadsheet User's Guide*.
Allowable values:	Inherited
Default value:	GQSV-TEXT-RESOURCES
gxl-button-state	See gxl-control-button in the *G2 XL Spreadsheet User's Guide*.
Allowable values:	ENABLED, DISABLED, RELEASED
Default value:	RELEASED

Methods

gxl-control-button::gxl-reflect-selection-state-in-button
gxl-control-button::gxl-set-button-state
gxl-control-button::gxl-perform-button-function

View Manager Class and Operations

Class

gqsv-tabular-view-manager on page 429

Operations

gqsv-activate-view-filter on page 430
gqsv-deactivate-view-filter on page 431
gqsv-get-view-filter on page 432
gqsv-set-view-filter on page 433

gqsv-tabular-view-manager

Class Inheritance Path

gqs-view-manager, object, item

gqsv-activate-view-filter

Synopsis

gqsv-activate-view-filter
(*manager*: gqsv-tabular-view-manager, *client*: object)

Argument	Description
manager	The manager of the view.
client	The client for this call.

Description

This procedure activates the current view filter, if any, and refreshes the view. The filtered items are temporarily removed from view, but are still present in the underlying queue. To specify a view filter, use gqsv-set-view-filter on page 433.

gqsv-deactivate-view-filter

Synopsis

gqsv-deactivate-view-filter
(*manager*: gqsv-tabular-view-manager, *client*: object)

Argument	Description
manager	The manager for the view.
client	The client for this call.

Description

This procedure deactivates the current view filter, and makes all items in the queue visible. The filter remains assigned to the view, and can be reactivated by calling gqsv-activate-view-filter on page 430.

gqsv-get-view-filter

Synopsis

gqsv-get-view-filter
(*manager*: gqsv-tabular-view-manager, *client*: object)
–> *filter*: class item-or-value

Argument	Description
manager	The manager for the view.
client	The client for this call.

Return Value	Description
filter	The gqs-filter associated with the view, if any, or false.

Description

This procedure returns the current view filter, if any. If there is no view filter, this procedure returns false.

gqsv-set-view-filter

Synopsis

gqsv-set-view-filter
(*manager*: gqsv-tabular-view-manager, *filter*: item-or-value,
client: object)

Argument	Description
manager	The manager of the view.
filter	An instance of a gqs-filter, or false.
filename	The file to which to save.
client	The client for this call.

Description

This procedure sets the current view filter. If the *filter* argument is passed as false, any existing view filter is removed.

- If filtering is active, the filter is immediately activated.
- If view filtering is not currently active, the filter is set, but not activated.

A subsequent call to gqsv-activate-view-filter on page 430 is required to activate the filter.

gevm-refresh-view

Synopsis

gevm-refresh-view
(*manager*: class gqsv-tabular-view-manager, *win*: class ui-client-item)

Argument	Description
manager	The view manager.
win	The client.

Description

Refreshes the view. It reevaluates all items in the queue, applying the filter, if appropriate, and inserts them into the tabular display.

GQS View Manager Class and Operations

This chapter describes the root abstract class of all tabular view managers. Note that there is no root abstract class for templates.

Class

gqs-view-manager on page 436

Methods

gqs-view-manager::gqs-update-view-per-addition on page 438
gqs-view-manager::gqs-update-view-per-attribute on page 439
gqs-view-manager::gqs-update-view-per-delete on page 440
gqs-view-manager::gqs-update-view-per-removal on page 441

Procedures

gqs-create-view on page 442
gqs-deregister-view on page 443
gqs-register-view on page 444

gqs-view-manager

This is an abstract (non-instantiable) class that provides a conduit for interaction between a queue and its views. If you want to create a new style of view, you must define a subclass of this class, and implement the four methods that define the behavior of your view in response to removal events, item addition events, and attribute change events. The signatures of these methods are:

gqs-update-view-per-delete
(*manager*: class gqs-view-manager, *queue*: class gqs-queue,
client: class object).

This method is called when one or more items have been deleted that where contained in the queue.

gqs-update-view-per-removal
(*manager*: class gqs-view-manager, *queue*: class gqs-queue,
items: class item-list, *client*: class object).

This method is called when one or more items have been removed from the queue.

gqs-update-view-per-addition
(*manager*: class gqs-view-manager, *queue*: class gqs-queue,
new-items: class item-list, *win*: class g2-window).

This method is called when one or more items have been added to the queue.

gqs-update-view-per-attribute
(*manager*: class gqs-view-manager, *queue*: class gqs-queue,
item: class item, *attribute*: symbol).

This method is called when an item contained in the queue receives a new attribute value. Normally, this requires that the view manager be monitoring for attribute changes.

For example, suppose you were interested in providing a visual indication of alarms generated from another system, such as the native UI of a DCS. Although it would be possible to construct such a system by subclassing entries or writing callbacks on the queue, using a custom view manager provides the cleanest, most modular, and most object-oriented approach to solving this problem. You simply create a subclass of gqs-view-manager, and then write the three methods to contain the code to communicate with your remote UI, for example, using G2 ActiveXLink or G2 JavaLink. That code can access the entries programmatically (*Item* argument or the contents of the *ItemList* argument). To activate the system, create an instance of your class and then subscribe it as a queue listener using the API gqs-register-view on page 444.

A registered view manager could also serve as a programmatic user. That is, a subclass of view manager could listen to a Queue, and take automated actions in lieu of an operator or end user.

Class Inheritance Path

gqs-view-manager, object, item

gqs-view-manager::gqs-update-view-per-addition

Synopsis

gqs-view-manager::gqs-update-view-per-addition
(*manager*: gqs-view-manager, *queue*: gqs-queue, *items*: item-list,
client: object)

Argument	Description
manager	The manager to be updated.
queue	The underlying queue.
items	The list of items added to the queue.
client	The client for this call.

Description

This is a prototype for a method that must be implemented by a subclassed view manager to update the view in response to one or more items being added to the queue. Since gqs-view-manager is an abstract class, this method should never be called.

gqs-view-manager::gqs-update-view-per-attribute

Synopsis

gqs-view-manager::gqs-update-view-per-attribute
(*manager*: gqs-view-manager, *queue*: gqs-queue, *monitored-item*: item,
attribute: symbol)

Argument	Description
manager	The manager to be updated.
queue	The underlying queue.
monitored-item	The item whose attribute value changed.
attribute	The name of the attribute.

Description

This is a prototype for a method that must be implemented by a subclassed view manager to update the view in response to an item contained by the queue receiving a new attribute value. Since gqs-view-manager is an abstract class, this method should never be called.

gqs-view-manager::gqs-update-view-per-delete

Synopsis

gqs-view-manager::gqs-update-view-per-delete
(*manager*: gqs-view-manager, *queue*: gqs-queue, *client*: object)

Argument	Description
manager	The manager to be updated.
queue	The underlying queue.
client	The client for this call.

Description

This is a prototype for a method that must be implemented by a subclassed view manager, to update the view in response to one or more items contained by the queue being deleted. Since gqs-view-manager is an abstract class, this method should never be called.

gqs-view-manager::gqs-update-view-per-removal

Synopsis

gqs-view-manager::gqs-update-view-per-removal
(*manager*: gqs-view-manager, *queue*: gqs-queue, *items*: item-list,
client: object)

Argument	Description
manager	The manager to be updated.
queue	The underlying queue.
items	The list of items removed from the queue.
client	The client for this call.

Description

This is a prototype for a method that must be implemented by a subclassed view manager, to update the view in response to one or more items being removed from the queue. Since gqs-view-manager is an abstract class, this method should never be called.

gqs-create-view

Synopsis

gqs-create-view
(*template*: item, *queue*: gqs-queue, *client*: object)
–> *view-manager*: class gqs-view-manager

Argument	Description
template	Any item that provides the specification of a view by implementing this method.
queue	The queue to which the view is to be linked.
client	The client where the view is to be shown. Note that for gqmv-tabular-view-template, this argument must be a g2-window. For arbitrary user extensions of this method, however, this may be a ui-client-item.

Return Value	Description
view-manager	The manager for the view. Typically this is a gqmv-tabular-view-manager, but it could be any subclass of gqs-view-manager if that has been implemented.

Description

This procedure enables you to create a view programmatically, given a template and a queue. It has the same effect as launching a view from the create-view menu choice, except that the menu choice can use a view access table to select the proper template.

Note that we recommend calling gevm-create-view instead, which has the advantage of displaying a native browser view, if applicable, and checks if any browser view for the specified template and queue is already displayed for a given client, which avoids displaying multiple browser views.

gqs-deregister-view

Synopsis

gqs-deregister-view
    (*queue*: gqs-queue, *view-manager*: gqs-view-manager, *client*: object)

Argument	Description
queue	The queue currently associated with the view manager.
view-manager	The manager for the view.
client	The client for this call.

Description

Use this procedure to disassociate a view and its underlying queue. See also gqs-register-view on page 444.

gqs-register-view

Synopsis

gqs-register-view
(*queue*: gqs-queue, *view-manager*: gqs-view-manager, *client*: object)

Argument	Description
queue	The queue to be associated with the given view manager.
view-manager	Any view manager.
client	The client for this call.

Description

Use this procedure to associate a view manager and a queue. A queue provides updates to each registered view manager. See gqs-view-manager on page 436 for more details on the relationship between queues and view managers. To reverse the effect of this call, use gqs-deregister-view on page 443.

gevm-refresh-view

Synopsis

gevm-refresh-view
(*manager*: class gqs-view-manager, *win*: class ui-client-item)

Argument	Description
manager	The view manager.
win	The client.

Description

Refreshes the view.

8

Filters and Subscriptions

Describes the classes and operations for filtering and subscriptions.

Introduction **448**

Filter Configuration **449**

Filter and Subscription Classes **450**
- gqs-and-filter **451**
- gqs-attribute-filter **452**
- gqs-compound-filter **453**
- gqs-filter **454**
- gqs-or-filter **455**
- gqs-subscription **456**

Filter and Subscription Operations **457**
- gqs-filter::gqs-apply-filter **458**
- gqs-apply-filter **459**
- gqs-attach-filter-to-subscription **460**
- gqs-detach-filter-from-subscription **461**
- gqs-get-subscription-details **462**
- gqs-get-subscriptions-from-queue **463**
- gqs-get-subscriptions-from-queue-to-queue **464**
- gqs-get-subscriptions-to-queue **465**
- gqs-populate-compound-filter **466**
- gqs-subscribe **467**
- gqs-unsubscribe **468**

Introduction

This section describes the API for filtering and subscription. These features are all exposed from the underlying GQS module.

The operations described here can often be accomplished in several different ways. For example, a problem that could be solved using filters and the view filtering buttons might also be solved using subscriptions and filters.

Filters

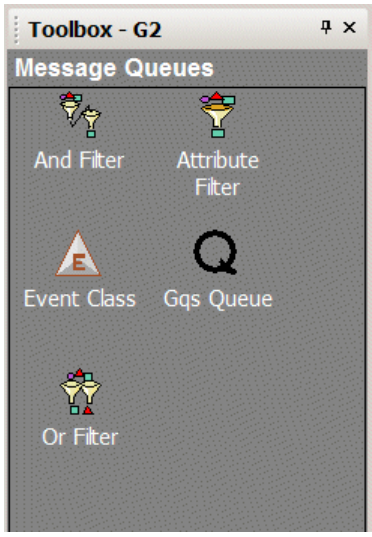
Three instantiable filter classes are provided: gqs-attribute-filter, gqs-and-filter, and gqs-or-filter. The latter two enable two or more instances of the first to be combined into a single algorithm. In addition, the parent class, gqs-filter, can be subclassed to enable you to define any arbitrary filtering algorithm that can then be used with the queues.

Subscriptions

With subscriptions, filters can be applied between two queues. Subscribing queues could also be used, with or without filters, to route entries from multiple queues to a central queue for a single display. For example, multiple alarm queues might be defined to generate different entry classes or have different expiration times, but then a single, subscribing queue be used to collect all the different entries in a single place for display.

Filter Configuration

1 In Developer mode, choose View > Toolbox - G2 and display the Message Queues palette:



2 Create an And Filter or an Or Filter, or create a generic Attribute Filter.

3 Configure the attributes as described in gqs-and-filter on page 451, gqs-or-filter on page 455, and gqs-attribute-filter on page 452.

Filter and Subscription Classes

gqs-and-filter on page 451
gqs-attribute-filter on page 452
gqs-compound-filter on page 453
gqs-filter on page 454
gqs-or-filter on page 455
gqs-subscription on page 456

gqs-and-filter

This is a compound filter that applies AND logic. See gqs-compound-filter on page 453 for details.

Class Inheritance Path

gqs-and-filter, gqs-compound-filter, gqs-filter, gfr-object-with-uuid, object, gfr-item-with-uuid, item

Attributes

Attribute	Description
gqs-filters	See "gqs-compound-filter" on page 453.
Allowable values:	Inherited
Default value:	GQS-PERMANENT-ITEM-LIST
gqs-combination-logic	See "gqs-compound-filter" on page 453.
Allowable values:	AND, OR
Default value:	AND

Methods

gqs-filter::gqs-apply-filter on page 458

gqs-attribute-filter

This filter tests the value of an attribute of an item against a target value, using equals, not-equals, contains, and other comparison functions. You would use an attribute filter, for example, if you want to select alarms with priority less than 3 by setting the attribute-name to priority, test to less-than, and target-value to 3.

Class Inheritance Path

gqs-attribute-filter, gqs-filter, gfr-object-with-uuid, object, gfr-item-with-uuid, item

Attributes

Attribute	Description
attribute-name	The name of the attribute to be tested.
Allowable values:	Any symbol
Default value:	G2
test	The test to be applied.
Allowable values:	EQUALS, DOES-NOT-EQUAL, CONTAINS, DOES-NOT-CONTAIN, GREATER-THAN, GREATER-THAN-OR-EQUAL-TO, LESS-THAN, LESS-THAN-OR-EQUAL-TO, EXISTS, DOES-NOT-EXIST
Default value:	EQUALS
target-value	The target value of the comparison.
Allowable values:	Any value
Default value:	0.0

Methods

gqs-filter::gqs-apply-filter on page 458

gqs-compound-filter

This class defines a filter that is a combination of other filters.

- If the combination logic is AND, an item passes the compound filter only if it passes all subfilters.
- If the combination logic is OR, an item passes the compound filter if it passes any subfilter.

In other words, AND logic acts like subfilters are in series, with fewer and fewer items passing through each filter, while OR logic acts as if the subfilters are in parallel, and the output is the total of all passed items. To add a subfilter to a compound filter, use gqs-populate-compound-filter on page 466. See also gqs-and-filter on page 451 and gqs-or-filter on page 455.

Class Inheritance Path

gqs-compound-filter, gqs-filter, gfr-object-with-uuid, object, gfr-item-with-uuid, item

Attributes

Attribute	Description
gqs-filters	A list of subfilters in this compound filter.
Allowable values:	Any gqs-permanent-item-list
Default value:	GQS-PERMANENT-ITEM-LIST
gqs-combination-logic	The type of combination logic to be applied
Allowable values:	AND, OR
Default value:	AND

Methods

gqs-filter::gqs-apply-filter on page 458

gqs-filter

This is the parent class for view filters and subscription filters. It is an abstract class and should not be instantiated.

To create a new type of filter, subclass gqs-filter and define the method gqs-apply-filter.

See gqs-filter::gqs-apply-filter on page 458 for details.

See also gqs-attribute-filter on page 452, gqs-compound-filter on page 453, gqs-and-filter on page 451, and gqs-or-filter on page 455.

Class Inheritance Path

gqs-filter, gfr-object-with-uuid, object, gfr-item-with-uuid, item

Attributes

none

Methods

gqs-filter::gqs-apply-filter on page 458

gqs-or-filter

This is a compound filter that applies OR logic. See gqs-compound-filter on page 453 for details.

Class Inheritance Path

gqs-or-filter, gqs-compound-filter, gqs-filter, gfr-object-with-uuid, object, gfr-item-with-uuid, item

Attributes

Attribute	Description
gqs-filters	See gqs-compound-filter on page 453.
Allowable values:	Inherited
Default value:	GQS-PERMANENT-ITEM-LIST
gqs-combination-logic	See gqs-compound-filter on page 453.
Allowable values:	AND, OR
Default value:	OR

Methods

gqs-filter::gqs-apply-filter on page 458

gqs-subscription

Use an instance of a gqs-subscription if you want one queue to send items that match certain filter criteria to another queue. A subscription always involves exactly two queues, a sender and receiver. When the receiver subscribes to the sender, it checks any items it subsequently receives against the filter criteria, and sends the appropriate items to the receiving queue. Generally, subscriptions are created programmatically using the create action. See gqs-subscribe on page 467, gqs-unsubscribe on page 468, and gqs-attach-filter-to-subscription on page 460 for more information.

Class Inheritance Path

gqs-subscription, gfr-object-with-uuid, object, gfr-item-with-uuid, item

Attributes

Attribute	Description
gqs-send-already-collected-items	If this attribute is true, when the receiving queue subscribes to a sending queue, the sending queue immediately sends any items contained in the sending queue that match the filter criteria.
Allowable values:	Any truth-value
Default value:	false

Filter and Subscription Operations

gqs-filter::gqs-apply-filter on page 458
gqs-apply-filter on page 459
gqs-attach-filter-to-subscription on page 460
gqs-detach-filter-from-subscription on page 461
gqs-get-subscription-details on page 462
gqs-get-subscriptions-from-queue on page 463
gqs-get-subscriptions-from-queue-to-queue on page 464
gqs-get-subscriptions-to-queue on page 465
gqs-populate-compound-filter on page 466
gqs-subscribe on page 467
gqs-unsubscribe on page 468

gqs-filter::gqs-apply-filter

Synopsis

gqs-filter::gqs-apply-filter
(*Filter*: gqs-filter, *SourceList*: item-list, *ResultList*: item-list, *Client*: object)

Argument	Description
Filter	The filter that is to be applied to *SourceList*.
SourceList	The input list of items to be filtered.
ResultList	An output list of items that pass the filter criterion.
Client	The client for this call.

Description

Extend this method for subclasses of gqs-filter to create your own filtering algorithm.

gqs-apply-filter

Synopsis

gqs-apply-filter
(*Filter*: gqs-filter, *SourceList*: item-list, *ResultList*: item-list, *Client*: object)

Argument	Description
Filter	The filter to be applied to *SourceList*.
SourceList	The items to be filtered.
ResultList	A list to contain the results, which are the items passed by the filter.
Client	The client for this call.

Description

This procedure tests the value of an attribute of an item against a target value, using equals, not-equals, contains, and other comparison functions. You would use an attribute filter, for example, if you want to select alarms with priority less than 3 by setting the attribute-name to PRIORITY, the test to LESS-THAN, and the target-value to 3. The filter is applied to each item in the *SourceList* and the items passing the filter criteria are appended to the *ResultList*, without clearing the *ResultList* first.

If *Filter* is a gqs-compound-filter, then each of the subfilters of *Filter* is applied and the result combined using AND or OR logic. See gqs-compound-filter on page 453 for more details.

See also gqs-filter::gqs-apply-filter on page 458.

gqs-attach-filter-to-subscription

Synopsis

gqs-attach-filter-to-subscription
(*Subscription*: gqs-subscription, *Filter*: gqs-filter, *Client*: object)

Argument	Description
Subscription	The subscription to be associated with *Filter*.
Filter	The filter to be associated with *Subscription*.
Client	The client for this call.

Description

Use this procedure to associate a filter with a subscription. Each subscription can have at most one filter. The effect of multiple filters can be achieved with a compound filter. To remove the filter from a subscription, use gqs-detach-filter-from-subscription on page 461.

gqs-detach-filter-from-subscription

Synopsis

gqs-detach-filter-from-subscription
(*Subscription*: gqs-subscription, *Client*: object)

Argument	Description
Subscription	The target subscription.
Client	The client for this call.

Description

Use this procedure to remove a filter from a subscription.

gqs-get-subscription-details

Synopsis

gqs-get-subscription-details
(*Subscription*: gqs-subscription, *Client*: object)
–> *Queue*: class item-or-value, *Queue*: class item-or-value,
Filter: class item-or-value

Argument	Description
Subscription	Any subscription
Client	The client for this call.

Return Value	Description
Queue	The subscribing queue, or false.
Queue	The providing queue, or false.
Filter	The filter associated with the subscription, or false.

Description

This procedure returns information about a given subscription, in particular, the subscribing and providing queues, and the filter, if any, associated with the subscription. If any of these attributes are not defined for *Subscription*, false is returned.

gqs-get-subscriptions-from-queue

Synopsis

gqs-get-subscriptions-from-queue
(*Queue*: gqs-queue, *ItemList*: item-list, *Client*: object)

Argument	Description
Queue	The queue that is the target of this call.
ItemList	An item list to hold the results of this call.
Client	The client for this call.

Description

This procedure returns a list of subscriptions that have been issued from *Queue*. The subscriptions are appended to *ItemList* without clearing the list first. See also gqs-get-subscriptions-from-queue-to-queue on page 464 and gqs-get-subscriptions-to-queue on page 465.

gqs-get-subscriptions-from-queue-to-queue

Synopsis

gqs-get-subscriptions-from-queue-to-queue
(*Subscriber*: gqs-queue, *Provider*: gqs-queue, *ItemList*: item-list,
Client: object)

Argument	Description
Subscriber	The subscribing queue.
Provider	The queue that is passing items to the subscriber queue.
ItemList	A list to contain the results of this call.
Client	The client for this call.

Description

This procedure returns the subscriptions associated with two queues, the recipient queue (*Subscriber*) and sending queue (*Provider*). The subscriptions are appended to *ItemList*.

gqs-get-subscriptions-to-queue

Synopsis

gqs-get-subscriptions-to-queue
(*Queue*: gqs-queue, *ItemList*: item-list, *Client*: object)

Argument	Description
Queue	The queue that is the target of this call.
ItemList	An item list to contain the results.
Client	The client for this call.

Description

This procedure returns a list of subscriptions that are currently held by a queue. The subscriptions are appended to *ItemList* without clearing the list first. See also gqs-get-subscriptions-from-queue on page 463 and gqs-get-subscriptions-from-queue-to-queue on page 464.

gqs-populate-compound-filter

Synopsis

gqs-populate-compound-filter
(*CompoundFilter*: gqs-compound-filter, *Filters*: item-list,
ClearFirst: truth-value, *Client*: object)

Argument	Description
CompoundFilter	A compound filter containing none or any number of subfilters.
Filters	A list of subfilters to add to the compound filter.
ClearFirst	A flag indicating whether existing subfilters should be removed from the compound filter before adding the filters contained in *Filters*.
Client	The client for this call.

Description

Use this procedure to add subfilters to a compound filter. Using the *ClearFirst* flag, you can specify if the existing subfilters (if any) are removed from *CompoundFilter* before adding the filters given in the *Filters* item list. If the item list is empty and the *ClearFirst* flag is true, this has the effect of clearing the compound filter.

gqs-subscribe

Synopsis

gqs-subscribe
(*Subscription*: gqs-subscription, *Provider*: gqs-queue,
Requestor: gqs-queue, *Client*: object)

Argument	Description
Subscription	The subscription to be added to the queues.
Provider	The queue that holds the subscription and provides items to the *Requestor*.
Requestor	The queue that receives items from the *Provider*.
Client	The client for this call.

Description

Use this procedure to add a subscription between two queues.

- If no filter is attached to the subscription, all items added to the provider queue are sent to the requestor queue. With a filter, only those items fulfilling the filter criteria are sent.
- If the attribute gqs-send-already-collected-items of the subscription is true, then calling gqs-subscribe results in immediate forwarding of all items in the provider queue, or those that meet the filter criteria, if any.

gqs-unsubscribe

Synopsis

gqs-unsubscribe
(*Provider*: gqs-queue, *Subscription*: gqs-subscription,
Subscriber: gqs-queue, *Client*: object)

Argument	Description
Provider	The queue that holds the subscription.
Subscription	The subscription that is to be removed.
Subscriber	The subscriber to the subscription.
Client	The client for this call.

Description

Use this procedure to remove a subscription between two queues.

9

Logging

Describes the classes and operations for logging.

Introduction **469**

Logging Class **471**
- glf-logging-manager **472**

Logging Operations **475**
- glf-default-file-name-generator **476**
- glf-default-log-file-header-writer **477**
- glf-default-log-file-scheduler **478**
- glf-disable-logging **479**
- glf-enable-logging **480**
- glf-set-fixed-log-closing-times **481**
- glf-write-to-log-file **482**

Introduction

This section describes the API for logging. These features are all exposed from the underlying GLF module.

The operations described here can often be accomplished in several different ways. For example, a problem that might be considered one for logging could also be solved more efficiently by using the gqs-view-manager. Suppose you want to log data about alarm occurrences and acknowledgements to a data base. Initially, this might seem like something the logging managers should be customized to do. However, by using the gqs-view-manager, it would be much easier to get notifications of changes to the queues and to send that event elsewhere in your system, packaging the data as needed.

Logging managers provide an easy way to manage the logging of entry data to files. Logging managers use the G2 system procedures for opening, closing, and writing to files, but in a way that manages it out of sight of the user.

The attributes on the logging manager can be set to your own procedures, enabling necessary customization. At any time, you can call glf-write-to-log-file within your code to add to a log.

Logging is initiated from the queues. The command whether to log or not, or the configurations of logging are all queue-based.

Note that view managers provide a nice way to subscribe to events. At that point, you can log them, perhaps in a way unrelated to how the logging manager is set up. For example, if you are logging events to a database, it might be a better solution to access the database from a subscribed view manager rather than through a queue or a logging manager.

Logging Class

glf-logging-manager on page 472

glf-logging-manager

This is the basic foundation class for logging managers. This class should not be extended. It is included in the documentation for reference only.

Class Inheritance Path

glf-logging-manager, object, item

Attributes

Attribute	Description
glf-logging-enabled	This read-only attribute indicates whether logging is enabled or disabled for this logging manager. The value is toggled by the user through menu choices. See also glf-enable-logging on page 480 and glf-disable-logging on page 479.
Allowable values:	Any truth-value
Default value:	false
glf-log-directory	This attribute is configurable by the user. It indicates the directory where log files are to be stored. If no text is given (the default), the default G2 directory is used.
Allowable values:	Any text
Default value:	""
glf-log-file-name-template	
Allowable values:	Any text
Default value:	"log_*.txt"
glf-log-file-name-generator	Names a procedure that generates filenames. The procedure must be of the format as described in glf-default-file-name-generator on page 476.

Attribute	Description
Allowable values:	Any symbol
Default value:	GLF-DEFAULT-FILE-NAME-GENERATOR
glf-current-log-file	This attribute is read-only. The name of the current log file is written here.
Allowable values:	Any text
Default value:	""
glf-file-header-writer	Names a procedure that generates the header for the log files. The procedure must be of the format as described in glf-default-log-file-header-writer on page 477.
Allowable values:	Any symbol
Default value:	GLF-DEFAULT-LOG-FILE-HEADER-WRITER
glf-time-interval-to-open-new-log-file	A configurable time interval, after which to open a new log file.
Allowable values:	Any integer
Default value:	86400
glf-maximum-file-size-in-bytes	The log file size which, when exceeded, causes the logging manager to open a new file. Units are in bytes.
Allowable values:	Any integer
Default value:	100000

Attribute	Description
glf-log-file-scheduler	The symbol names a procedure that determines when a newly started log file should be closed. Modifying this attribute overrides the default behavior. See glf-default-log-file-scheduler on page 478 for the proper format for this procedure.
Allowable values:	Any symbol
Default value:	GLF-DEFAULT-LOG-FILE-SCHEDULER
glf-automatically-delete-empty-log-files	A user-configurable attribute that determines whether, when creating a new file, old and empty files should be removed.
Allowable values:	Any truth-value
Default value:	true

Logging Operations

glf-default-file-name-generator on page 476
glf-default-log-file-header-writer on page 477
glf-default-log-file-scheduler on page 478
glf-disable-logging on page 479
glf-enable-logging on page 480
glf-set-fixed-log-closing-times on page 481
glf-write-to-log-file on page 482

glf-default-file-name-generator

Synopsis

glf-default-file-name-generator
(*Log*: glf-logging-manager, *Client*: object)
–> *Filename*: text

Argument	Description
Log	The logging manager that needs to generate a new filename.
Client	The client for this call.

Return Value	Description
Filename	The new filename.

Description

This procedure is named in the glf-file-name-generator attribute of *Log*. If this procedure is used, the log file is stored in your working directory and is named `log_YYYYMMDDHHMMSS.txt`, where:

- `YYYYMMDD` is the year, month, and day when the file is created.
- `HHMMSS` is the hour, minutes, and seconds when the file is created.

To change the way filenames are generated, write a new procedure with the same arguments as glf-default-file-name-generator. It is suggested that you use the template attribute of *Log* to create the new name.

glf-default-log-file-header-writer

Synopsis

glf-default-log-file-header-writer
(*Log*: glf-logging-manager, *Client*: object)

Argument	Description
Log	The logging manager controlling this logging session.
Client	The client for this call.

Description

When called, this procedure writes a header to the file being logged by *Log*. This procedure should not normally be called by the user, however, as it is called during the normal operation of the glf-logging-manager. If you are writing your own log file header writer, use glf-write-log-to-file(Log, "", Client) to write the lines of your header.

glf-default-log-file-scheduler

Synopsis

glf-default-log-file-scheduler
(*Log*: glf-logging-manager, *Client*: object)
–> *time*: integer

Argument	Description
Log	The logging manager whose closing is to be scheduled.
Client	The client for this call.

Return Value	Description
time	For a file opened at the time of the call to this procedure, returns the time to close the file.

Description

This procedure should not be called directly by the user. The default procedure uses the glf-time-interval-to-open-new-log-file and any time entered via the API glf-set-fixed-log-closing-times and returns the nearest file closing time.

If you override the default procedure, you may be disabling these provided ways of configuring the log file scheduling.

glf-disable-logging

Synopsis

glf-disable-logging
(*Log*: glf-logging-manager, *Client*: object)

Argument	Description
Log	A logging manager that is to cease logging.
Client	The client for this call.

Description

Turns logging off for *Log*.

glf-enable-logging

Synopsis

glf-enable-logging
(*Log*: glf-logging-manager, *Client*: object)

Argument	Description
Log	A logging manager that is to start logging.
Client	The client for this call.

Description

Turns logging on for *Log*.

glf-set-fixed-log-closing-times

Synopsis

glf-set-fixed-log-closing-times
(*Log*: glf-logging-manager, *Times*: integer-list, *Client*: object)

Argument	Description
Log	The logging manager whose closing is to be scheduled.
Times	An integer list. Each element is a closing time.
Client	The client for this call.

Description

Schedules the closing of the named logging manager. Inputs times in minutes since midnight. This procedure checks, sorts and converts to seconds.

glf-write-to-log-file

Synopsis

glf-write-to-log-file
(*Log*: glf-logging-manager, *Text*: text, *CheckSize*: truth-value, *Client*: object)

Argument	Description
Log	The logging manager controlling the target log file.
Text	A line of text to be written by *Log*.
CheckSize	When true, checks the size of the log file after writing the file to see if a new file needs to be generated.
Client	The client for this call.

Description

This procedure writes a line of text to the log specified by *Log*. When using this procedure, GLF is managing the opening and closing of G2 streams and well as managing the maximum size and age of the log files.

The procedure checks the file size after it writes the file to see if a new file needs to be created. GLF allows the user to specify a maximum file size for the log. If a new log file needs to be created, the procedure closes the existing log file, then opens a new one. After the new file is opened, the logging manager calls the log file header writer. User-defined header writers should pass false as the *CheckSize* argument to avoid checking the file size when a new file is already in the process of being created. All other callers should pass true.

10

GEVM Examples

Describes the examples in the GEVM demo.

Introduction **484**

Message Queues **485**

Message Browsers **487**

Message Browser Configuration **492**

Native Message Browser **496**

View Templates and Managers **500**

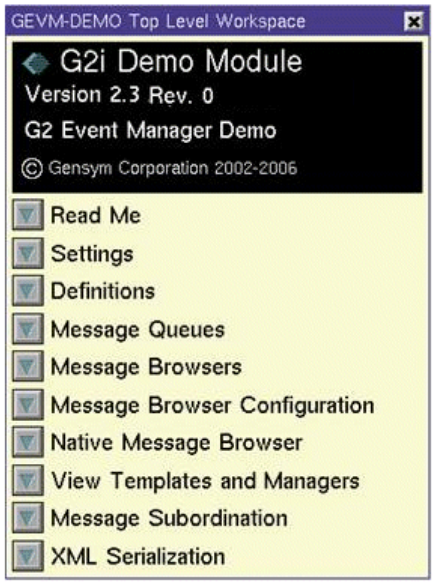
Message Subordination **501**

XML Serialization **501**

Introduction

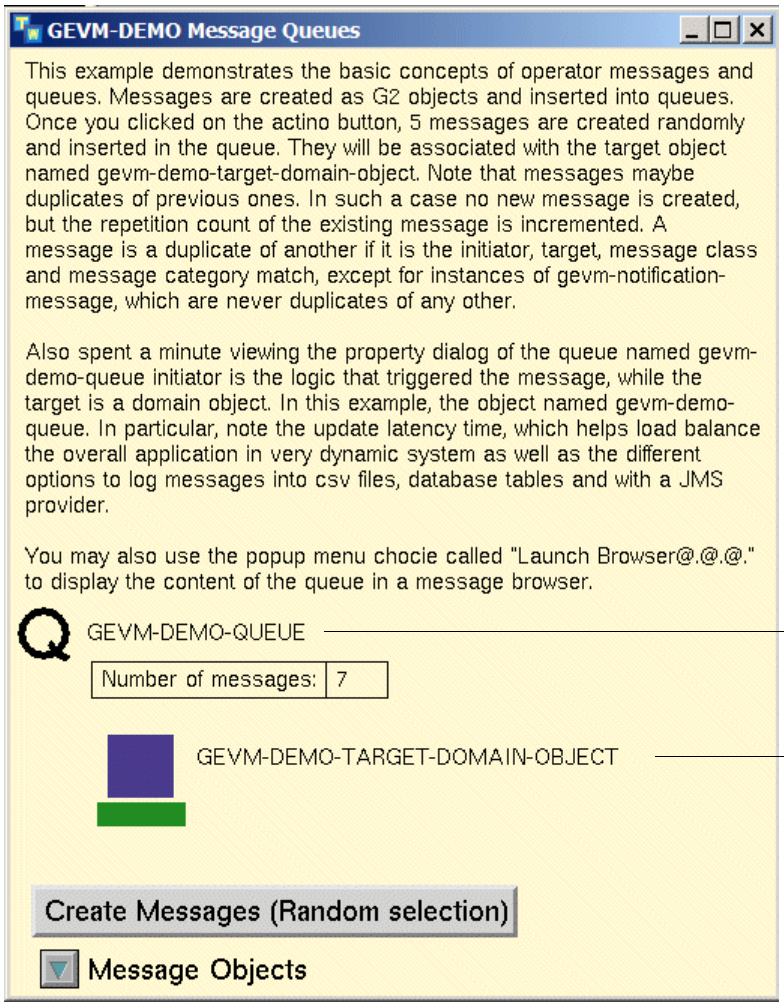
GEVM provides a demo called `gevm-demo.kb` located in the `g2i examples` directory. The demo provides various examples, including how to create custom GEVM message queues and message browsers, both classic and native, and how to configure the various objects required for creating custom message browsers and custom messaging.

Here is the gevm-demo-top-level workspace:

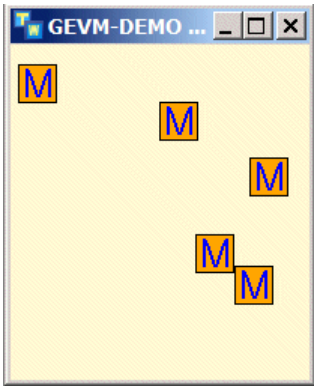


Message Queues

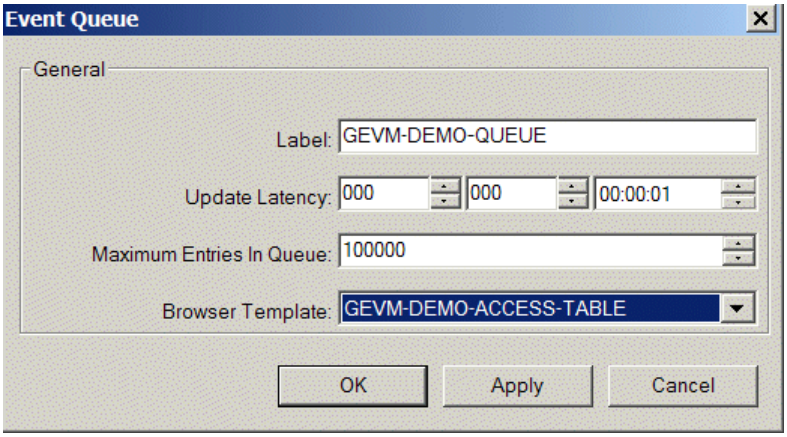
Here is the Message Queues workspace, which shows a gevm-gqs-queue and a target domain object:



Clicking the Create Messages button calls a user-defined procedure defined on the Definitions workspace, which creates random GEVM message objects. These message appear on the subworkspace of the Message Objects button:

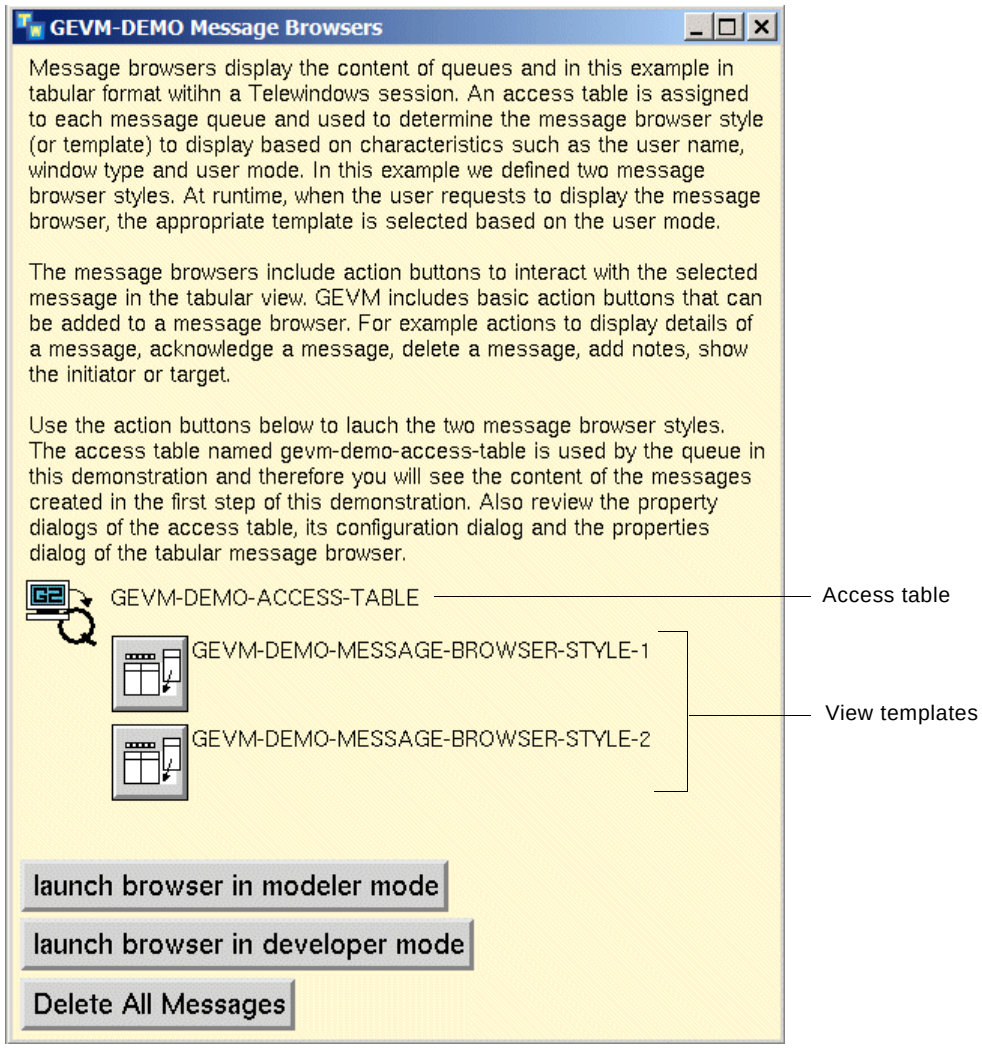


Here is the properties dialog for the gevm-demo-queue, which specifies gevm-demo-access-table as the browser template to use for displaying the messages in a message browser. Browser templates and access tables are described in the following section.

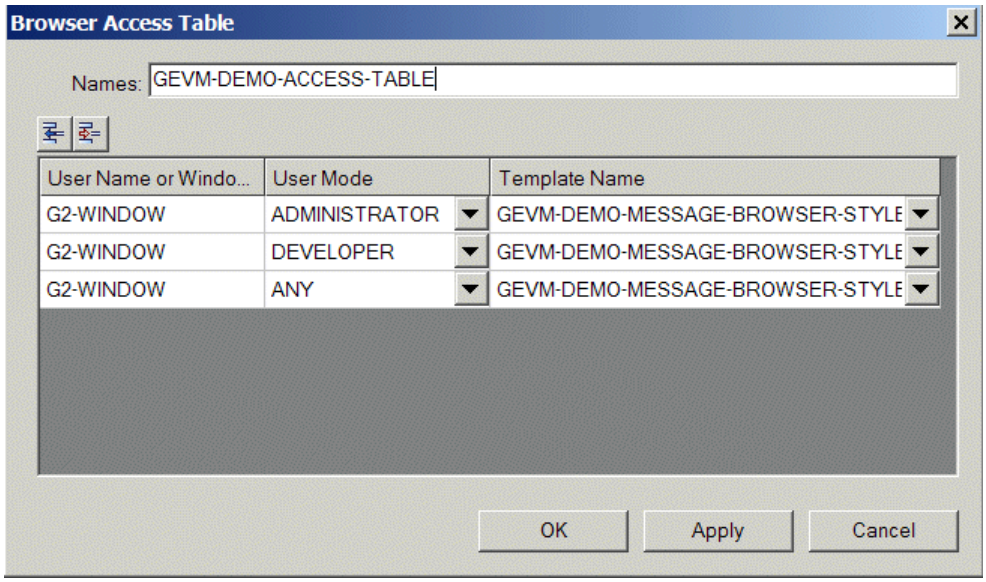


Message Browsers

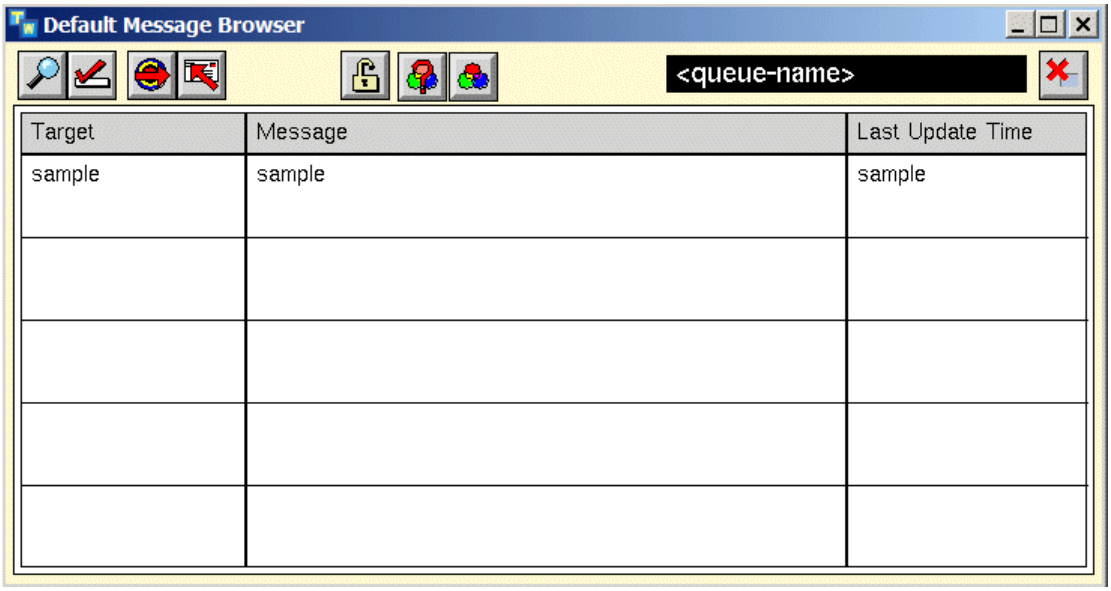
Here is the Message Browsers workspace, which shows a gqs-queue-access-table for configuring the tabular view template to use for different user modes, and two gqsv-tabular-view-template objects:



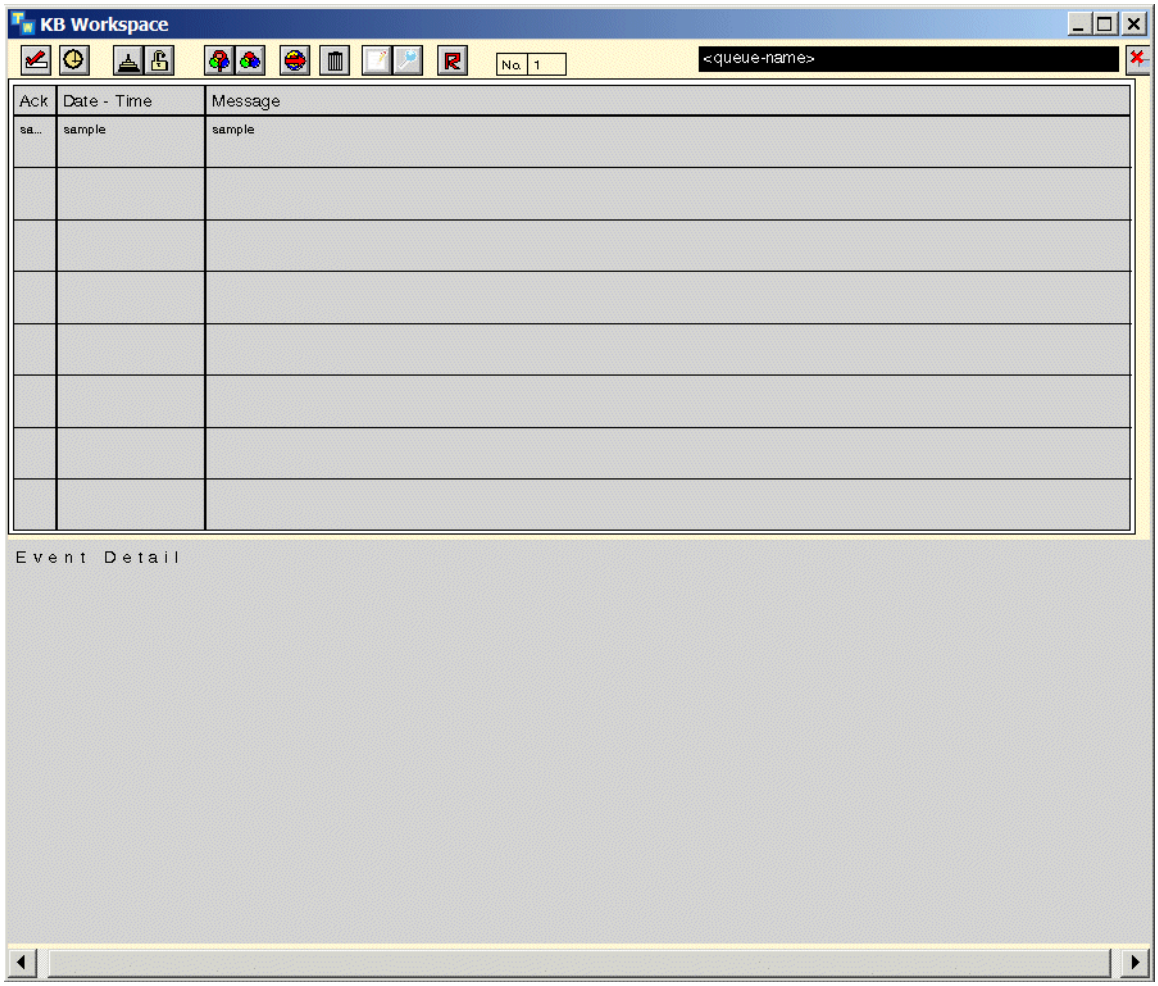
Here is the properties dialog for the access table. (In the classic dialog, you must click the Configure Access Table button in the dialog.) The access table specifies which browser template to use in each user mode. For example, for any user mode other than administrator or developer, the queue should use the gevm-demo-message-browser-style-1 browser template.



To configure the browser template, choose configure tabular view on the tabular view object, which you must do in administrator mode. Here is the tabular view for gevm-demo-message-browser-style-1, which is used in modeler mode:



Here is the tabular view for gevm-demo-message-browser-style-2, which is used in developer and administrator modes:



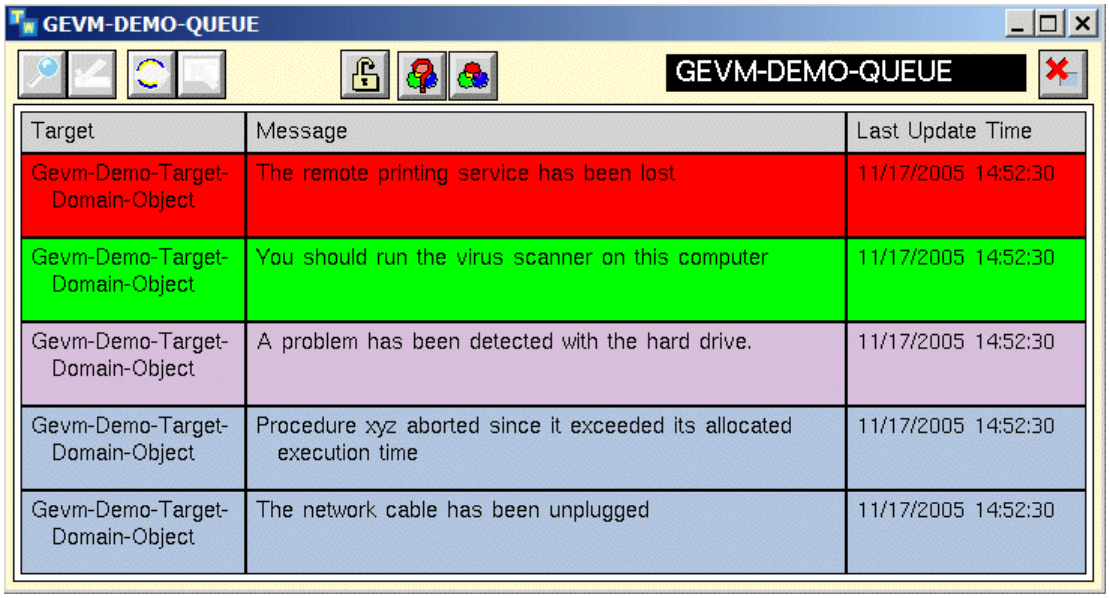
The launch browser in modeler mode button changes the user mode to modeler and launches the tabular view specified in the access table for that user mode, namely gevm-demo-message-browser-style-1:

```
in order conclude that the g2-user-mode of this window = the symbol modeler
    and start gevm-launch-view (gevm-demo-queue, this window)
```

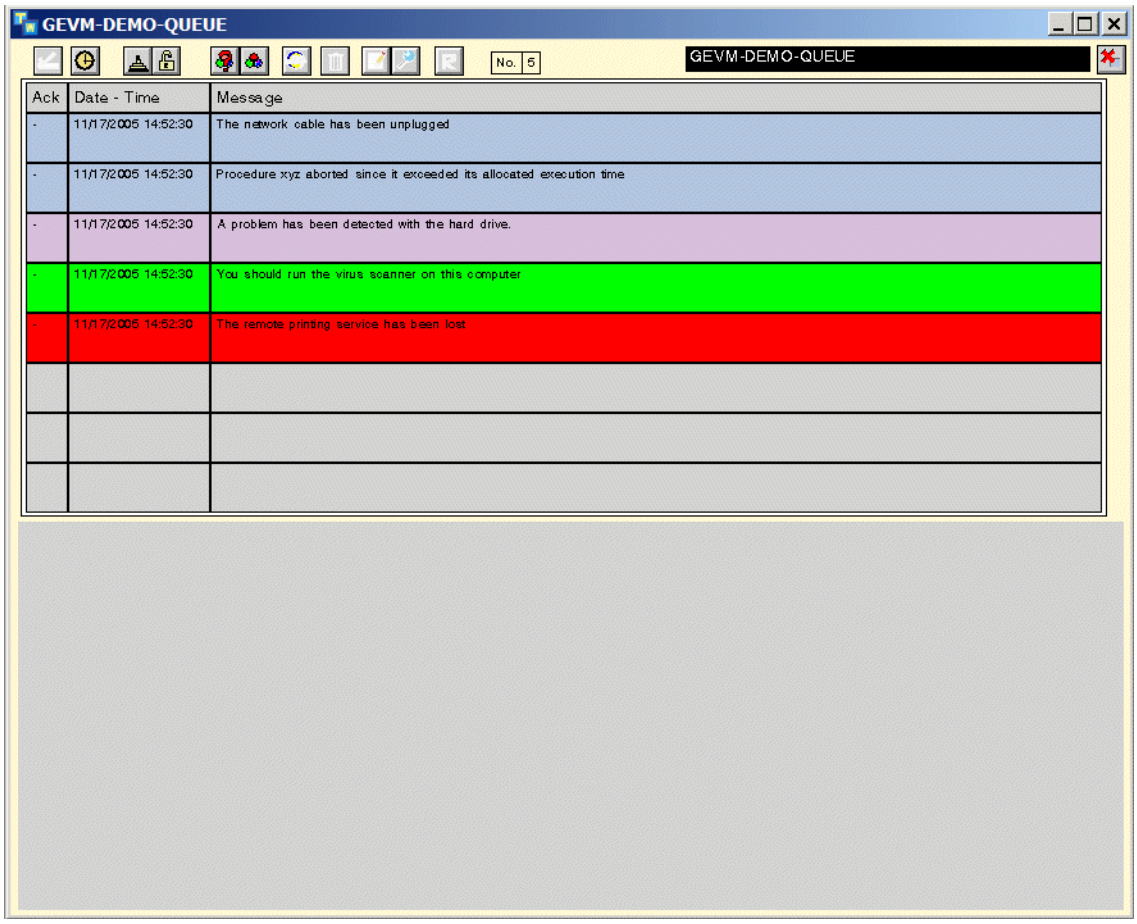
Similarly, the launch browser in developer mode button changes the user mode to developer and launches gevm-demo-message-browser-style-2:

```
in order conclude that the g2-user-mode of this window = the symbol
    developer and start gevm-launch-view (gevm-demo-queue, this window)
```

Here is the result of launching the browser in modeler mode:



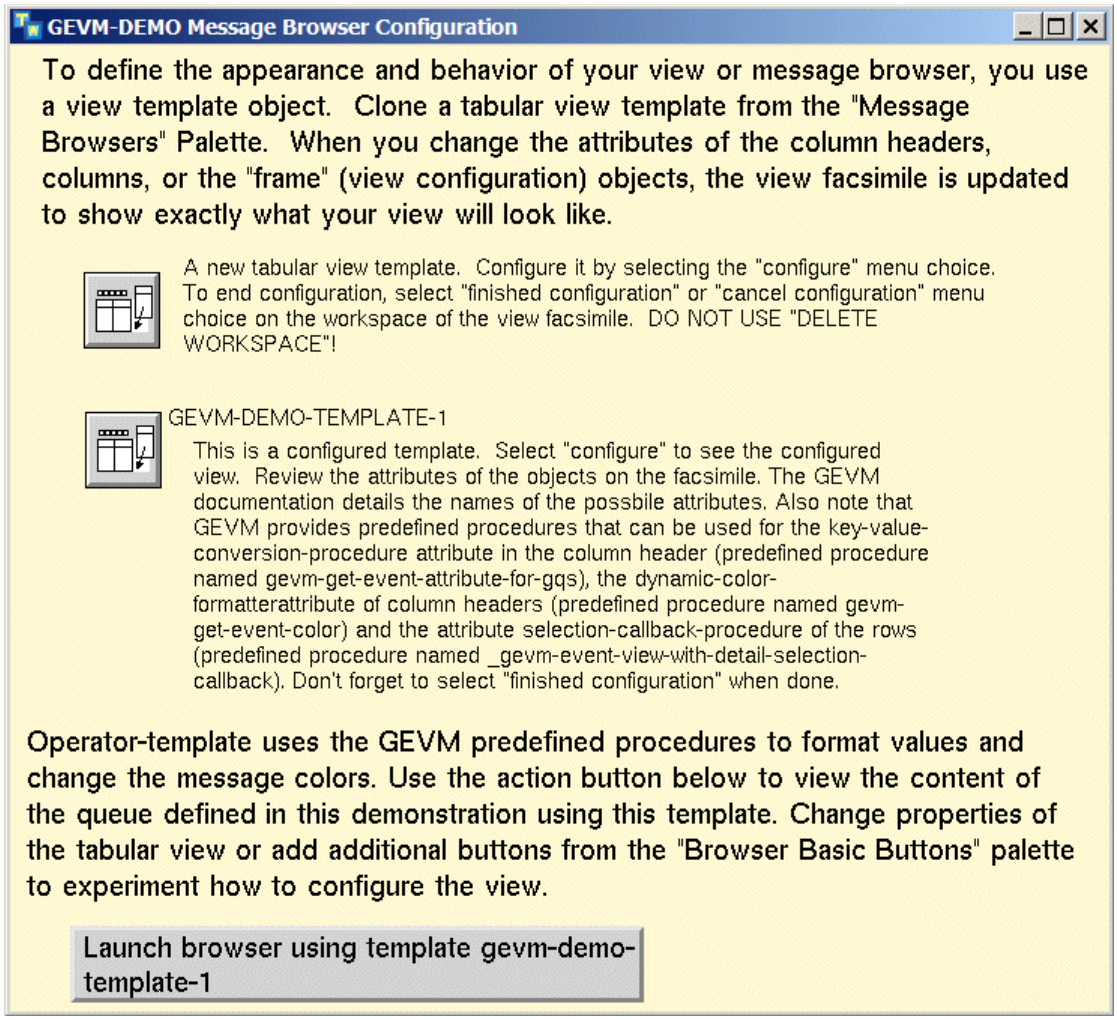
Here is the result of launching the browser in developer mode:



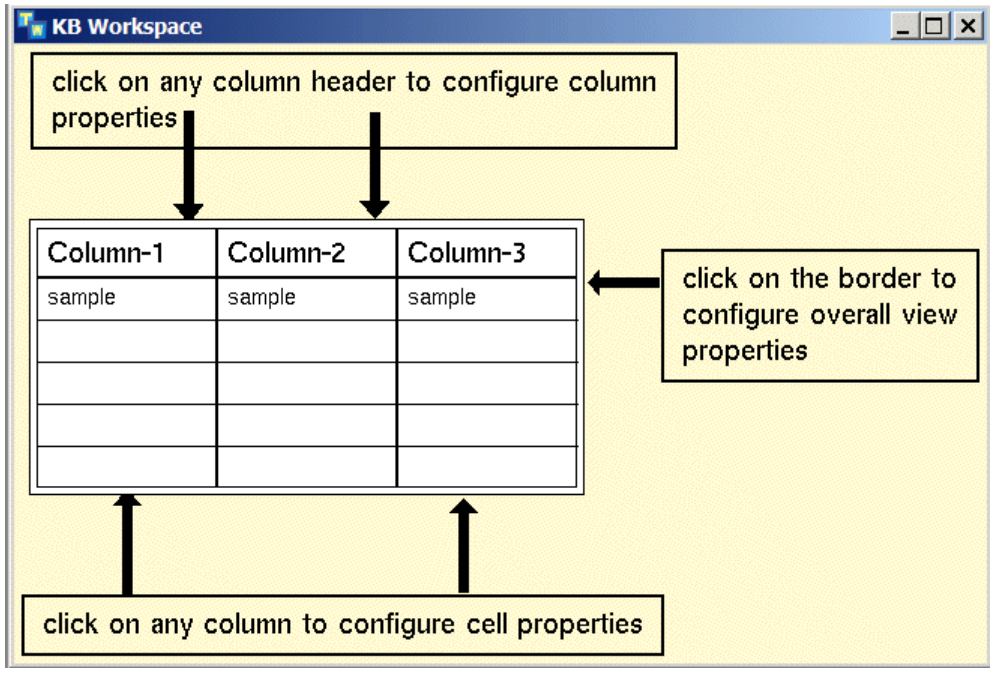
Message Browser Configuration

Here is the Message Browser Configuration workspace, which shows how to configure a tabular view and how to launch a specific browser template for a specific message queue.

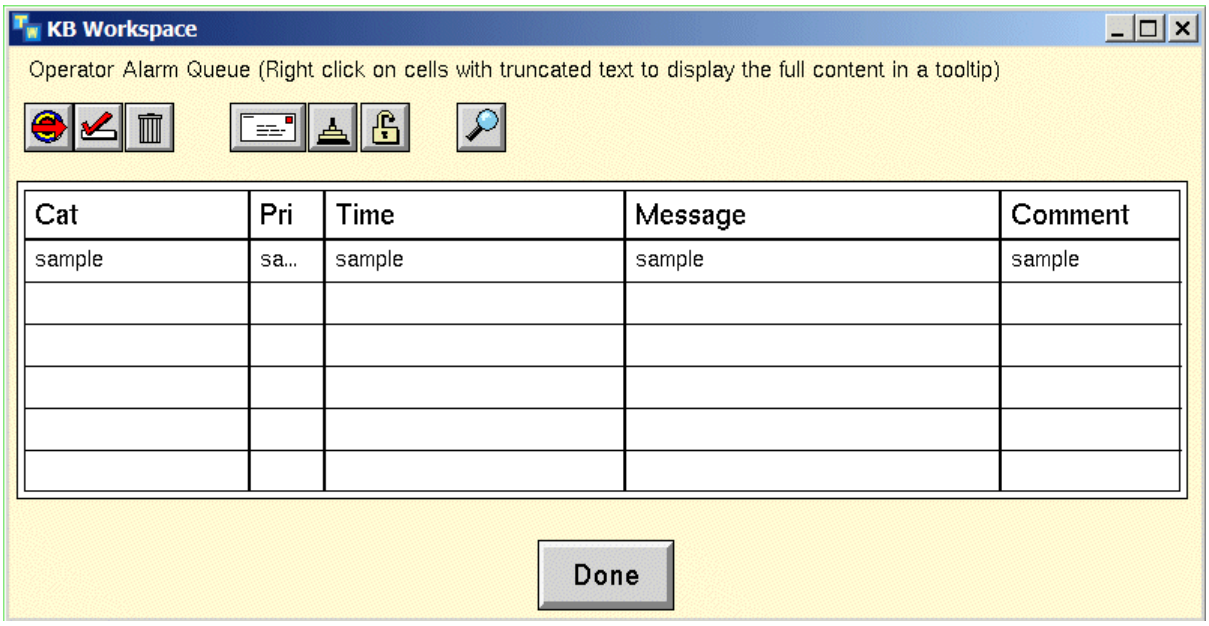
The first tabular view describes how to configure the view, in general, and the second tabular view shows a view that has been configured:



To configure a tabular view, choose configure tabular view on the tabular view object, which you must do in administrator mode. This workspace shows a new tabular view just cloned from the Message Browser palette and describes how to configure the view. Mouse right on the column header to display a popup menu for adding and deleting columns.



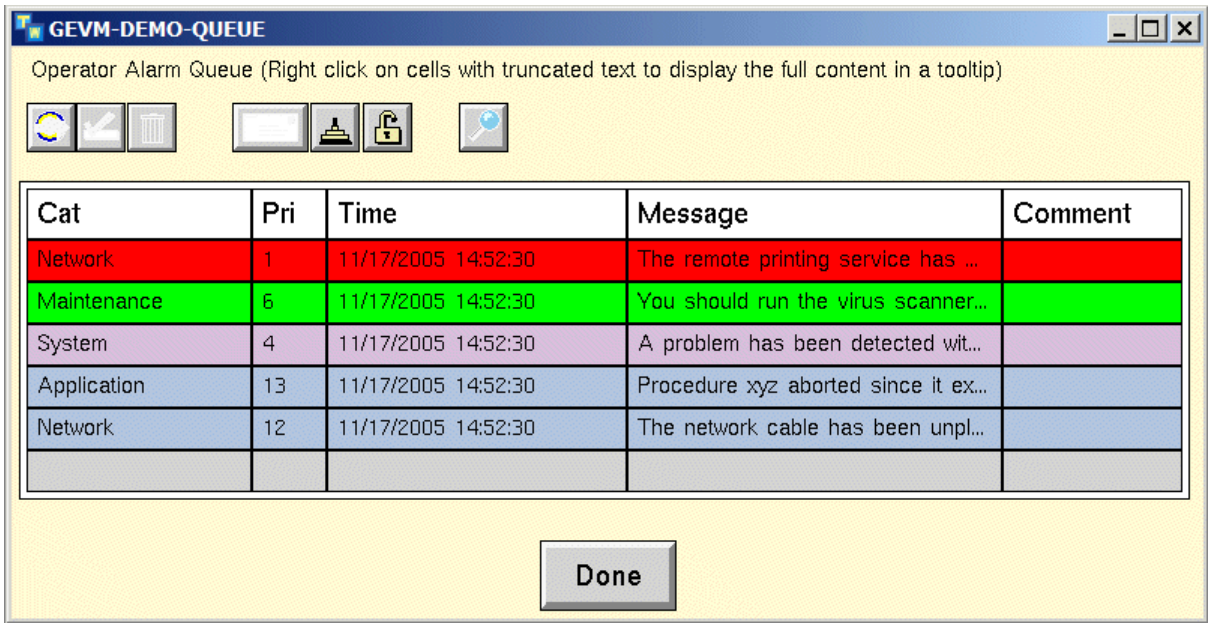
Here is the tabular view configuration for the gevm-demo-template-1 browser template:



The Launch browser using template gevm-demo-template-1 button calls gqs-create-view to launch a specific browser template for a specific queue:

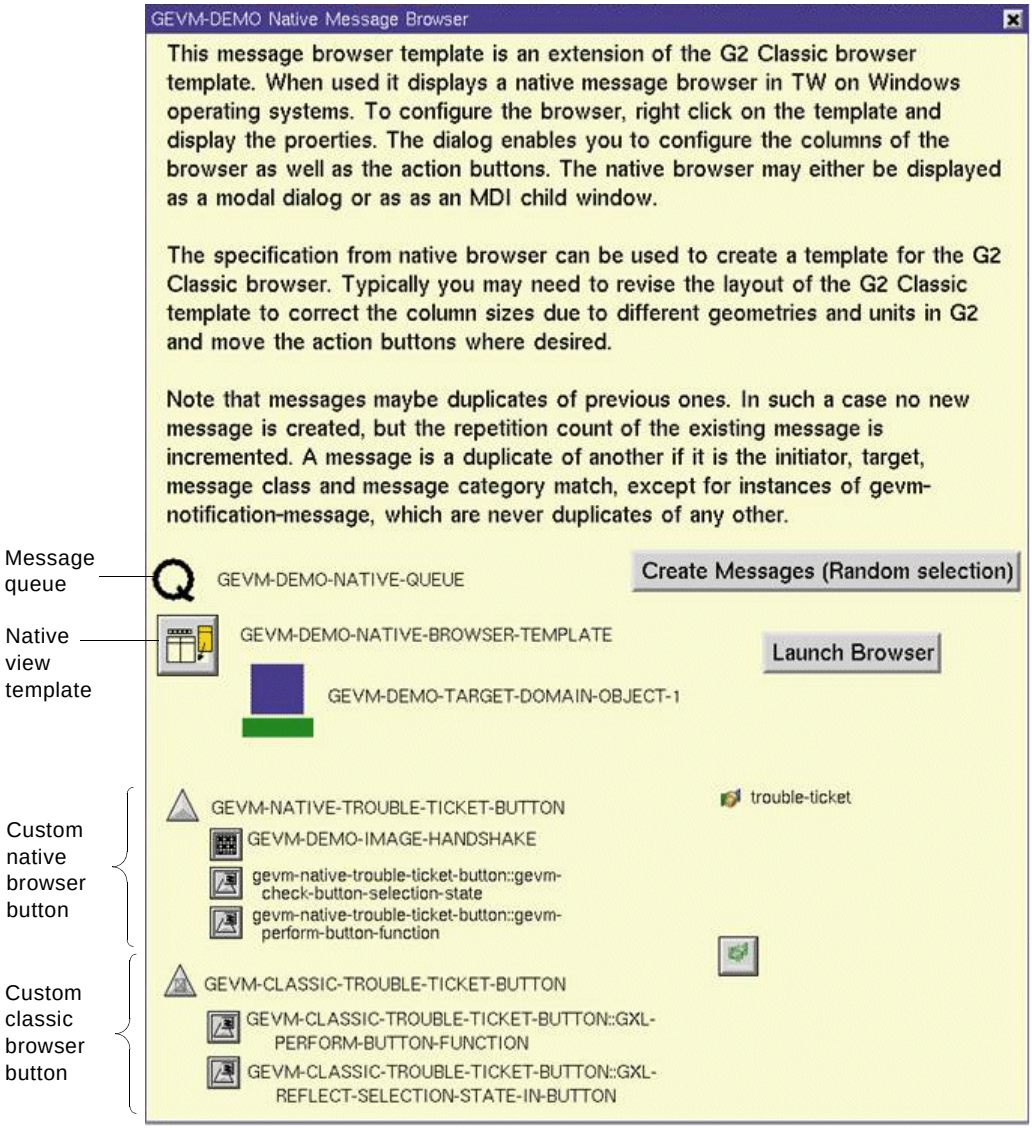
```
start gqs-create-view
    (gevm-demo-template-1, gevm-demo-queue, this window)
```

Here is the result of clicking the Launch browser using template gevm-demo-template-1 button:

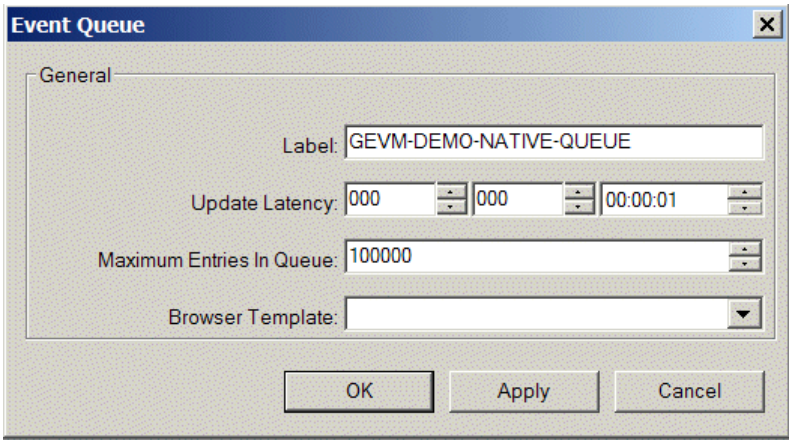


Native Message Browser

Here is the Native Message Browser workspace, which shows how to create a native message browser for a message queue, using a native view template, and how to create custom browser buttons, both native and classic:

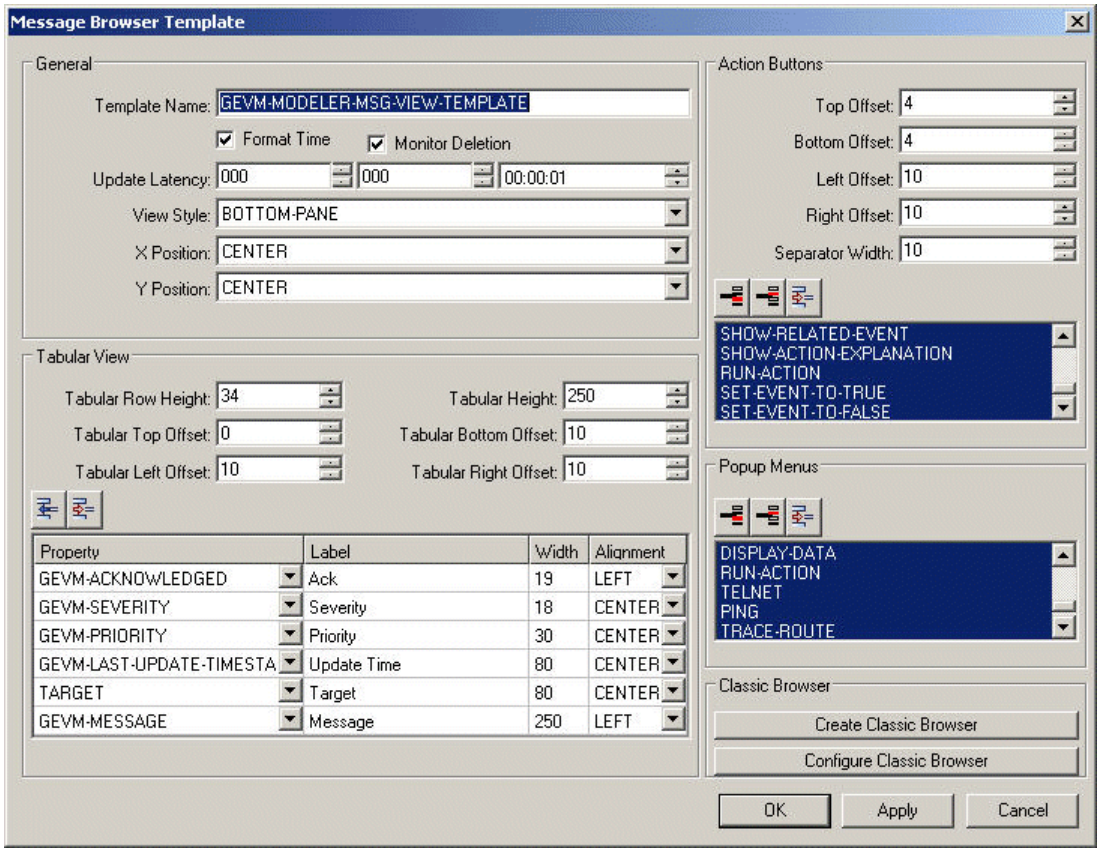


Here is the properties dialog for the gevm-demo-native-queue message queue:



Note When creating native message browsers, you can also specify an access table as the browser template for a queue, as described in "Message Browsers" on page 487.

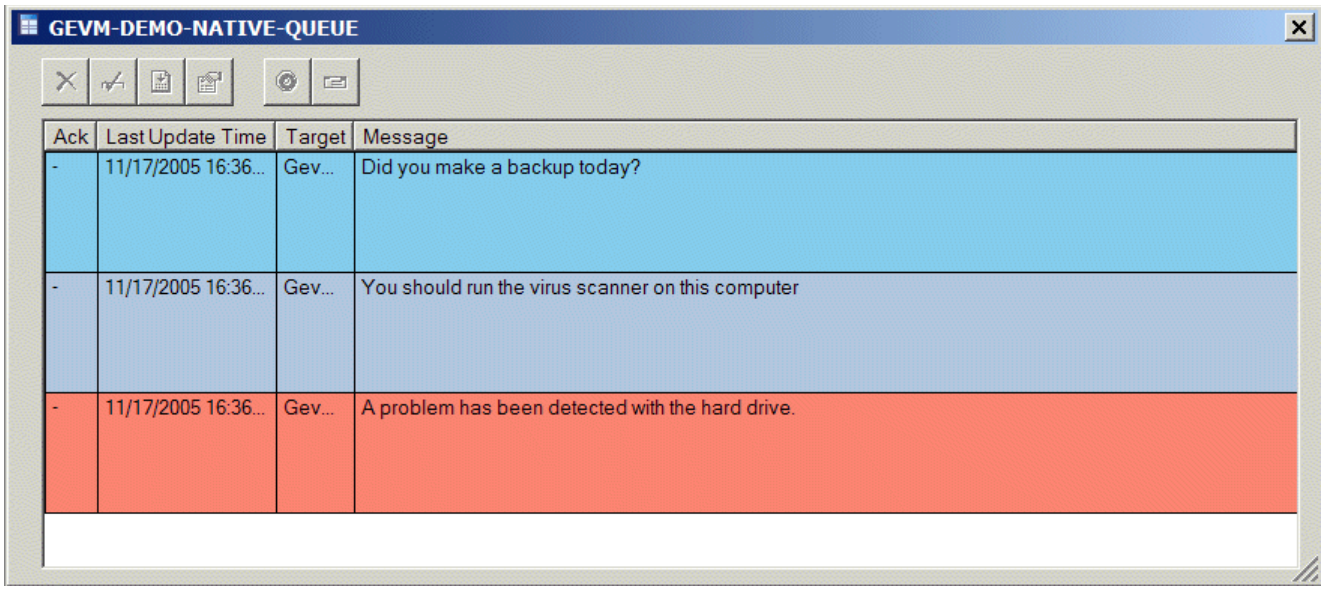
Here is the properties dialog for the gevm-demo-native-browser-template for configuring the native and classic browser view template:



The Launch Browser button calls gevm-launch-view to launch the specified native browser view template on the queue:

```
start gevm-launch-view(gevm-demo-native-queue, this window)
```

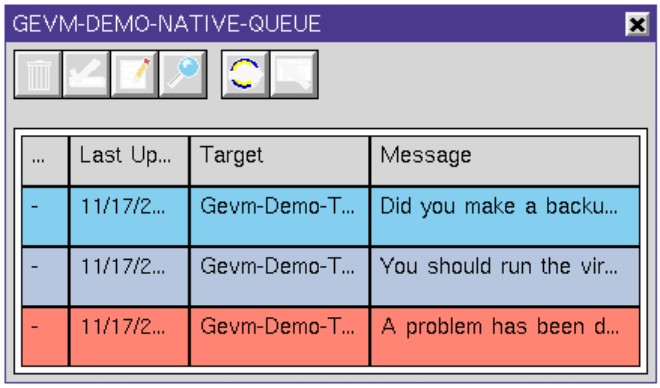
When viewed through Telewindows, the native message browser looks like this:



You can also create and configure a GXL message browser for viewing in G2 classic. When you launch the browser, using gevm-launch-view and view it in G2 classic, the GXL message browser appears.

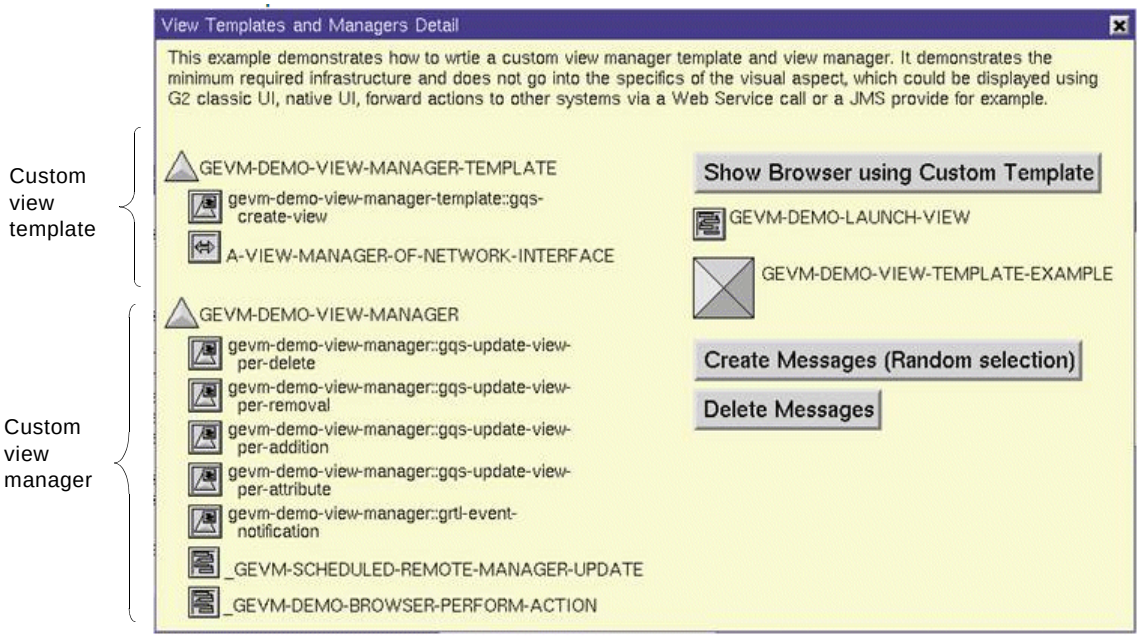
Click the Create Classic Browser button to create a classic browser view template that is based on the current specification of the native view template. To configure the classic view template, click the Configure Classic Browser and move the dialog to view the browser template. You can drag the buttons to different locations, choose delete on a button to delete it or configure button to configure its attributes, if any. You can also add, delete, and configure the columns in the tabular view.

Clicking the Launch Browser button displays this message browser when viewed in G2 classic:



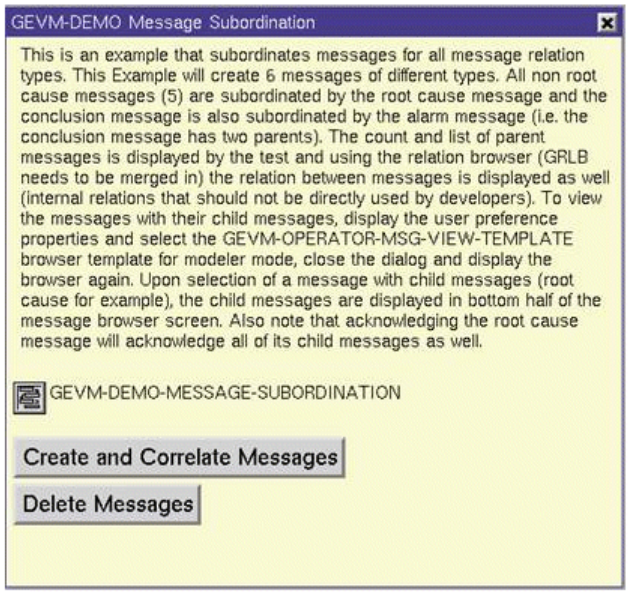
View Templates and Managers

Here is the View Templates and Managers workspace, which shows how to create custom view templates and view managers:



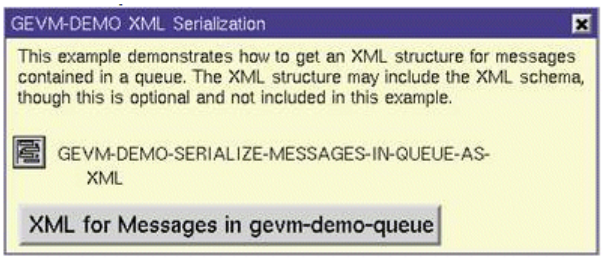
Message Subordination

Here is the Message Subordination workspace, which shows how to create custom message correlation by subordinating related messages:



XML Serialization

Here is the XML Serialization workspace, which shows how to get an XML structure for messages contained in a queue:



Index

A

access tables
  class and operations 353
  configuring 354
  creating and configuring 355
  how access manager works 354
  introduction to 12

B

browsers
  See message browsers and views
buttons
  classes and operations
    classic 422
    native 380
  introduction to 365

C

color, message, customizing 34
configuration file 30
customer support services xviii
customization
  database tables 351
  message classes 263
  message color 34
  message correlation 40, 264
  message log handler 348
  message priority escalation 38
  queue classes 348

D

databases, creating custom database tables 351
demos
  `gevm-demo.kb` 484

E

errors, handling as messages 33
escalation, message
  introduction to 13
event metrics 14
events
  See Also messages
  class definition 70
  classes 90
  classes and operations 65
  classes of event blackboard 7
  configuring 67
  creation operations 73
  customization 263
  escalation operations 255
  event notification
    gevm-event 68
    gevm-gqs-queue 271
    introduction to 10
  initiating event operations 231
  introduction to 6
  management operations 168
  queues 267
  target event operations 207

F

files
  `gevm-demo.kb` 484
filters
  classes and operations 447
  configuring 449

G

G2 errors, handling as operator messages 33
G2 Event Manager (GEVM)
  APIs 16
  architecture 4
  event and message classes 6
  event classes and operations 65

introduction to 2
loading 6
module settings 21
programmers' interface and settings 16
user preferences 41
gevm.kb 6
gevm-acknowledged 201
gevm-action 91
gevm-action-done 94
gevm-action-to-do 97
gevm-advisory 100
gevm-advisory-message 103
gevm-alarm 106
gevm-calculated-alarm 109, 124
gevm-change-in-process-state-alarm 112
gevm-check-button-selection-state 384
gevm-command 115
gevm-conclusion 118
gevm-demo.kb 6
gevm-demo.kb file 484
gevm-deviation-alarm 121
gevm-disable-event-escalation 259
gevm-disable-reevaluate-period 260
gevm-enable-event-escalation 261
gevm-event 127
gevm-event::gevm-acknowledge 170
gevm-event::gevm-change-priority 171
gevm-event::gevm-clear-reevaluation-scheme 256
gevm-event::gevm-create-sub-event 172, 174, 176, 178
gevm-event::gevm-delete-event 180
gevm-event::gevm-email-notification 181
gevm-event::gevm-execute-action 182
gevm-event::gevm-get-event-color 183
gevm-event::gevm-get-event-states 184
gevm-event::gevm-get-initiating-item 185
gevm-event::gevm-get-reevaluation-scheme 257
gevm-event::gevm-get-sub-event-count 186, 187
gevm-event::gevm-get-sub-events 188, 189
gevm-event::gevm-get-superior-event-count 190, 191
gevm-event::gevm-get-superior-events 192, 193
gevm-event::gevm-get-target 194
gevm-event::gevm-remove-event 195
gevm-event::gevm-set-reevaluation-scheme 258
gevm-event::gevm-show-initiator 196
gevm-event::gevm-show-target 197
gevm-event::gevm-subordinate-event 198, 199
gevm-event::grtl-can-delete 200
gevm-event-class 71
gevm-event-class::gevm-create-event 74, 76, 78, 80
gevm-event-class::gevm-create-event-and-insert-into-queue 82, 84, 86, 88
gevm-event-has-sub-event 202
gevm-event-is-sub-event 203
gevm-event-state 130
gevm-external-fault 132
gevm-fault 135
gevm-find-queue-by-key 310
gevm-g2-error-message 138
gevm-get-all-queues 311
gevm-get-event-class-names 204
gevm-get-message-properties 312
gevm-get-most-recent-unacknowledged-event 313
gevm-get-property-values-of-all-events 314
gevm-get-property-values-of-events 316
gevm-get-queues-containing-item 318
gevm-get-selected-property-values-of-all-events 319
gevm-get-selected-property-values-of-events 321
gevm-gqs-queue 274
gevm-gqs-queue::gevm-acknowledge-all-events 281
gevm-gqs-queue::gevm-acknowledge-events 282
gevm-gqs-queue::gevm-add-event-to-queue 279
gevm-gqs-queue::gevm-delete-all-events 283
gevm-gqs-queue::gevm-delete-collected-events-created-outside-time-period 284
gevm-gqs-queue::gevm-delete-collected-events-created-outside-time-period-for-initiator-and-target 285
gevm-gqs-queue::gevm-delete-collected-events-created-within-time-period 287
gevm-gqs-queue::gevm-delete-collected-events-created-within-time-period-for-initiator-and-target 288

gevm-gqs-queue::gevm-delete-events 290
gevm-gqs-queue::gevm-get-collected-events 291
gevm-gqs-queue::gevm-get-collected-events-created-outside-time-period 292
gevm-gqs-queue::gevm-get-collected-events-created-outside-time-period-for-initiator-and-target 294
gevm-gqs-queue::gevm-get-collected-events-created-within-time-period 296
gevm-gqs-queue::gevm-get-collected-events-created-within-time-period-for-initiator-and-target 298
gevm-gqs-queue::gevm-get-collected-events-for-initiator-and-target 300
gevm-gqs-queue::gevm-get-collected-events-updated-outside-time-period 301
gevm-gqs-queue::gevm-get-collected-events-updated-outside-time-period-for-initiator-and-target 302
gevm-gqs-queue::gevm-get-collected-events-updated-within-time-period 304
gevm-gqs-queue::gevm-get-collected-events-updated-within-time-period-for-initiator-and-target 305
gevm-gqs-queue::gevm-get-event-count-for-queue 307
gevm-gqs-queue::gevm-remove-all-events 308
gevm-gqs-queue::gevm-remove-events 309
gevm-gqs-queue::grtl-initialize 280
gevm-has-initiating-item 205
gevm-has-target 206
gevm-inferred-fault 141
gevm-instrumentation-alarm 144
gevm-internal-fault 147
gevm-limit-alarm 150
gevm-message 153
gevm-module-settings 22
gevm-native-action-button 382
gevm-native-view-manager 387
gevm-native-view-manager-template 372
gevm-native-view-manager-template::grtl-show-properties 379
gevm-notification-message 156
gevm-perform-button-function 385
gevm-rate-of-change-alarm 159
gevm-refresh-view 390, 434, 445
gevm-root-cause 162
gevm-set-reevaluate-period 262
gevm-state-change 165
gevm-user-filter 55
gevm-user-preferences 43
gevm-user-preferences::gevm-email-notification 63
glf-default-file-name-generator 476
glf-default-log-file-header-writer 477
glf-default-log-file-scheduler 478
glf-disable-logging 479
glf-enable-logging 480
glf-logging-manager 472
glf-set-fixed-log-closing-times 481
glf-write-to-log-file 482
gqs-activate-attribute-monitoring 330
gqs-add-monitored-attributes 331
gqs-and-filter 451
gqs-apply-filter 459
gqs-attach-filter-to-subscription 460
gqs-attribute-filter 452
gqs-clear-queue 332
gqs-compound-filter 453
gqs-create-view 442
gqs-deactivate-attribute-monitoring 333
gqs-deregister-view 443
gqs-detach-filter-from-subscription 461
gqs-filter 454
gqs-filter::gqs-apply-filter 458
gqs-force-input-buffer-into-queue 334
gqs-get-collected-items 335
gqs-get-monitored-attributes 336
gqs-get-queues-containing-item 337
gqs-get-subscription-details 462
gqs-get-subscriptions-from-queue 463
gqs-get-subscriptions-from-queue-to-queue 464
gqs-get-subscriptions-to-queue 465
gqs-get-view-template 361
gqs-launch-view 338
gqs-number-of-collected-items 347
gqs-or-filter 455
gqs-populate-compound-filter 466
gqs-queue 324
gqs-queue::gqs-clear-queue 328
gqs-queue::gqs-receive-items 329
gqs-queue-access-table 359
gqs-receive-items 339
gqs-receive-single-item 340
gqs-register-view 444

gqs-remove-items 342
gqs-remove-monitored-attributes 343
gqs-remove-single-item 344
gqs-send-items 345
gqs-send-single-item 346
gqs-subscribe 467
gqs-subscription 456
gqs-unsubscribe 468
gqsv-activate-view-filter 430
gqsv-close-tabular-view 418
gqsv-close-view-button 423
gqsv-column 400
gqsv-column-header 405
gqsv-column-or-header 409
gqsv-deactivate-view-filter 431
gqsv-delete-view 419
gqsv-get-view-filter 432
gqs-view-manager 436
gqs-view-manager::gqs-update-view-per-addition 438
gqs-view-manager::gqs-update-view-per-attribute 439
gqs-view-manager::gqs-update-view-per-delete 440
gqs-view-manager::gqs-update-view-per-removal 441
gqsv-multiple-row-button 424
gqsv-number-of-viewed-items 420
gqsv-root-specification 410
gqsv-set-view-filter 433
gqsv-single-row-button 425
gqsv-tabular-view-manager 429
gqsv-tabular-view-template 398
gqsv-toolbar-button 426
gqsv-view-configuration 412
gqsv-view-is-locked 421
gqsv-workspace-location 415

I

item::gevm-acknowledge-all-events-for-initiator 232
item::gevm-acknowledge-all-events-for-target 208
item::gevm-acknowledge-events-for-initiator 233, 234
item::gevm-acknowledge-events-for-target 209, 210
item::gevm-delete-all-events-for-initiator 235
item::gevm-delete-all-events-for-target 211
item::gevm-delete-collected-events-created-outside-time-period-for-initiator 236
item::gevm-delete-collected-events-created-outside-time-period-for-target 212
item::gevm-delete-collected-events-created-within-time-period-for-initiator 237
item::gevm-delete-collected-events-created-within-time-period-for-target 213
item::gevm-delete-events-for-initiator 238, 239
item::gevm-delete-events-for-target 214, 215
item::gevm-get-collected-events-created-outside-time-period-for-initiator 240
item::gevm-get-collected-events-created-outside-time-period-for-target 216
item::gevm-get-collected-events-created-within-time-period-for-initiator 242
item::gevm-get-collected-events-created-within-time-period-for-target 218
item::gevm-get-collected-events-updated-outside-time-period-for-initiator 244
item::gevm-get-collected-events-updated-outside-time-period-for-target 220
item::gevm-get-collected-events-updated-within-time-period-for-initiator 246
item::gevm-get-collected-events-updated-within-time-period-for-target 222
item::gevm-get-event-count-for-initiator 248, 249
item::gevm-get-event-count-for-target 224, 225
item::gevm-get-events-for-initiator 250, 251, 252
item::gevm-get-events-for-target 226, 227, 228
item::gevm-remove-all-events-for-initiator 253
item::gevm-remove-all-events-for-target 229
item::gevm-remove-events-for-initiator 254
item::gevm-remove-events-for-target 230

L

logging
    classes and operations 469

M

message browsers
    See Also views
    configuring 492

demo 487
native 496
message queues
See queues
messages
See Also events
attributes of 9
classes and operations 65
classes of operator messages 7
customizing
classes 263
message color 34
message correlation 264
message escalation 38
message subordination 501
filtering 447
handling G2 errors as 33
introduction to 6
logging
class and operations 469
introduction to 13
message escalation 13
subscriptions 447
XML serialization of 501
metrics, event 14

P

priority, custom message escalation 38

Q

queues
access tables 353
classes and operations 267
configuring 269
customizing
classes 348
database table creation 351
message log handler 348
demo 485
event notification 271
GEVM queue classes and operations 272
GQS queue class and operations 323
introduction to 11

R

raw event 7

S

subscriptions
class and operations 447
gqs-subscription 456

V

view managers
class and operations
classic 428
native 386
customizing 500
GQS view manager class and operations 435
introduction to 12
view templates
classic
button classes 422
classes and operations 397
configuring 391
configuring buttons 393
modifying column headers 396
modifying columns 393
customizing 500
introduction to 12
native
button classes and operations 380
classes and operations 371
configuring 368
toolbar buttons 365
views
See Also view templates and view managers
GEVM view (native) 368
GQSV view 391
introduction to 12
view templates and view managers 363

X

XML serialization 501