

CGI Web Server e*Way Intelligent Adapter User's Guide

*Release 5.0.5 for Schema Run-time
Environment (SRE)*

Monk Version



Copyright © 2005, 2010, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related software documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications which may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure the safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. UNIX is a registered trademark licensed through X/Open Company, Ltd.

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Version 20100712162955.

Contents

Chapter 1

Introduction	5
Overview	5
Intended Reader	5
Components	6
System Requirements	6

Chapter 2

Installation	8
Supporting Documents	8
Installing the CGI Web Server e*Way	8
Installation Procedure	8
Windows Installation	8
Pre-installation	9
UNIX Installation	9
Pre-installation	9
Configuring the Participating Host Components in the Schema Manager	11
Configuring the Web Server Components	12
Files/Directories Created by the Installation	13

Chapter 3

Configuration	14
e*Way Configuration Parameters	14
General Settings	14
Request Reply IP Port	15
Push IP Port	15
Rollback if no Clients on Push Port	15
Wait For IQ Ack	15
Send Empty MSG When External Disconnect	16
MUX Instance ID	16
MUX Recovery ID	16
External Configuration Requirements	17

Replacing CGI Payload XML Encoding with Simple Binary Delimiting Logic	17
--	----

Chapter 4

Implementation	19
-----------------------	-----------

The Request/Reply Concept	19
----------------------------------	-----------

Request/Reply and the CGI Web Server e*Way Participating Host Components	19
--	----

The Request/Reply Schema	21
--------------------------	----

ETDs and Form Data	22
--------------------	----

Implementing the CGI Web Server e*Way via a web server	24
---	-----------

Sample Implementation	24
------------------------------	-----------

Index	26
--------------	-----------

Introduction

This chapter provides an overview of the CGI Web Server e*Way Intelligent Adapter.

1.1 Overview

The CGI Web Server e*Way is a gateway to the e*Gate system for a web server. Normally, a web server is limited to sharing local data sources only. By using the CGI Web Server e*Way, the web server can access remote data sources which otherwise are not available. The CGI Web Server e*Way allows the web server to access a variety of data sources through the e*Way system.

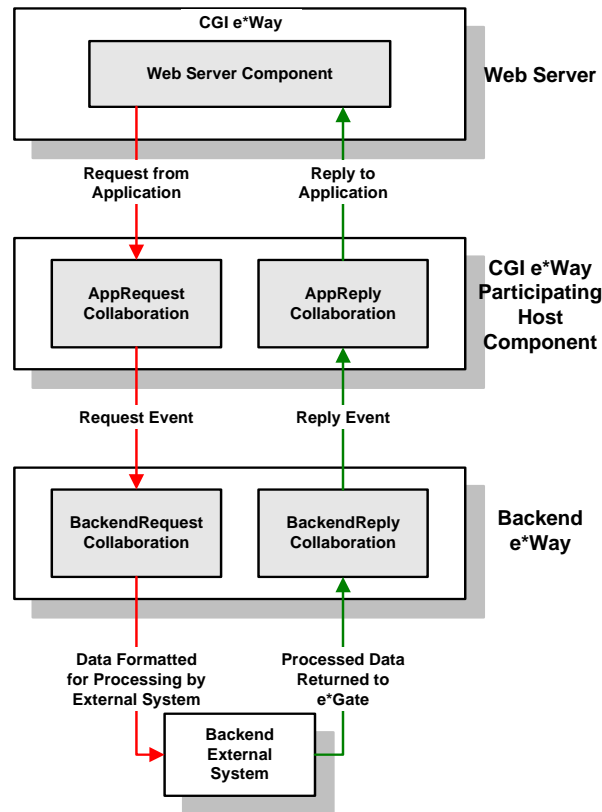
The CGI Web Server e*Way comprises two components, the web server component and the Participating Host component. The web server component resides in a web server's CGI-bin directory. It parses CGI input supplied by a web server using either the GET or POST method, packages the input along with the web server's environment variables in XML format, and sends the packaged message to the Participating Host component which resides on a machine where an e*Gate Participating Host is running. The Participating Host component forwards the message to the e*Gate backend system for processing. After sending the message, the web server component waits for a response from the participating host component. Upon receipt of the response, the response message is sent to the web server which can deliver the message to the requesting web browser.

For additional details, see [Chapter 4](#).

1.1.1. Intended Reader

The reader of this guide is presumed to be a developer or system administrator with responsibility for maintaining the e*Gate system; to have expert-level knowledge of operations and administration for the operating system(s) under which the web server and e*Gate systems run; to be thoroughly familiar with CGI; and to be thoroughly familiar with Windows-style GUI operations.

Figure 1 Overview of the CGI Web Server e*Way implementation



1.1.2. Components

The CGI Web Server e*Way is comprised of the following:

- Web server components also known as the CGI Web Server e*Way client: web server component executable and supporting library files
- Participating Host components also known as the CGI Web Server e*Way server: a multiplexing e*Way and supporting library files

A complete list of installed files appears in [Table 1 on page 13](#).

1.2 System Requirements

To use the CGI Web Server e*Way, you need the following:

- An e*Gate Participating Host
- A TCP/IP network connection

For the web server that will communicate with the CGI Web Server e*Way, you need a client system with the following:

- A Web Server

- Sufficient memory and disk space to support web-server functions. See your CGI web server user's guides for more information about server requirements.

Note: *The e*Gate Participating Host may optionally host the web server, but is not required to do so.*

Installation

This chapter covers the requirements for installing the CGI Web Server e*Way and how to configure the web server components needed. A list of the files and directories created by the installation is also provided.

2.1 Supporting Documents

The following documents are designed to work in conjunction with the *CGI Web Server e*Way Intelligent Adapter User's Guide* and to provide additional information that may prove useful to you.

- *e*Gate Integrator Installation Guide*
- *e*Gate Integrator System Administration and Operations Guide*.
- *README.txt* file on the e*Gate installation CD ROM.

2.2 Installing the CGI Web Server e*Way

This section describes the procedure for installing the CGI Web Server e*Way.

2.2.1. Installation Procedure

If you are installing this e*Way as part of a complete e*Gate installation, please follow the instructions in the *e*Gate Integrator Suite Installation Guide*. The CGI Web Server e*Way is installed as an “Add-on” component in the fourth phase of the installation.

If you are adding the CGI Web Server e*Way to an existing e*Gate installation, follow the instructions below.

2.3 Windows Installation

Before installing e*Gate on your Windows system, read the following sections to ensure a smooth and error-free installation.

You must have Administrator privileges to successfully install e*Gate on a Windows system.

2.3.1. Pre-installation

- Exit all programs before running the setup program, including any anti-virus applications.
- You must have Administrator privileges to install this e*Way.
- Review the **readme.txt** file provided on the installation media for important installation information.

To install the CGI Web Server e*Way on a Windows system:

- 1 Log in as an Administrator on the workstation on which you want to install the e*Way.
- 2 Close any open applications.
- 3 Launch the setup application on the e*Gate installation CD-ROM.
- 4 Follow the online prompts in the InstallShield® Wizard. When the **Select Components** dialog box appears, clear all the check boxes except **Add-ons**. Click **Next** as necessary to proceed through the setup application.
- 5 When the **User Information** dialog box appears, type your name and company name.

Note: e*Gate usernames and passwords are case-sensitive.

- 6 When the **Choose Destination Location** window appears, **do not** change the **Default Destination** folder unless you are directed to do so by Oracle support personnel; simply click **Next** to continue.
- 7 When the **Select Components** dialog box appears, select eWays, click the **Change** button, and select **CGI Web Server e*Way**.
- 8 Follow the on-screen prompts to complete the installation.

After the installation is complete, reboot the computer and launch the Schema Manager.

2.4 UNIX Installation

Before installing the e*Way on your UNIX system, read the following sections to ensure a smooth and error-free installation.

2.4.1. Pre-installation

- Exit all programs before running the setup program, including any anti-virus applications.
- You will need regular (non-root) user access to begin the e*Gate installation.

- Review the **readme.txt** file provided on the installation media for important installation information.

To install the CGI Web Server e*Way on a UNIX system:

- 1 Log on to the workstation on which you want to install the e*Way. If you do not wish to log in as root, you must log in as a user with sufficient privilege to install files in the “egate” directory tree.
- 2 Insert the CD-ROM into the drive.
- 3 If necessary, mount the CD-ROM drive.
On HP-UX systems, you must mount the drive with this command:

```
/etc/mount -F cdfs -o cdcase /dev/cdrom /cdrom
```

where /cdrom is the mount point.
- 4 At the shell prompt, type
cd /cdrom/setup
- 5 Start the installation script by typing:
setup.sh
- 6 If you are not running as root, you will see a message notifying you that services will not start automatically for non-root users. Press **Enter** to continue.
- 7 You will see a message confirming that you are running the e*Gate installation script, and reminding you that you can type - (hyphen) to back up a step or **QUIT** (all capitals) to exit the install program. Press **Enter** to continue.
- 8 You will be prompted to accept the license agreement. Type **y** and press **Enter**.
The platform type and a menu of options will display:

```
Installation type (choose one):  
0. Finished with installation.  Quit.  
1. e*Gate Addon Applications  
2. e*Gate Participating Host (Client)  
3. e*Gate Registry Server
```
- 9 Type 1 to select the **e*Gate Add-on Applications** and press **Enter**.
- 10 You will be prompted for the installation path. Press **Enter** to accept the default path, or enter a new path and press **Enter**.
 - If you are logged in as root, the suggested path will be **/opt/egate/client**.
 - If you are logged in under any other user name, the suggested path will be **/home/username/egate/client**.
Whether you install e*Gate to a **/home** directory to an application directory such as **/opt**, we strongly recommend that you use the recommended relative path “egate/client” as the destination directory for the add-on-application installation.
- 11 When prompted, type **U** to update (overwrite) and press **Enter**.

Note: **U** updates the installation, overriding files as necessary. **M** creates a directory and moves everything in the current directory to **directoryname.old**.

If you selected “U,” you will see a warning regarding shared EXE and DLL files. Read this warning and press **Enter** to continue.

- 12 Enter the name of the Registry Server supporting these add-on applications. If the installation utility detects a Registry Host running on the current server, it will suggest that host’s name.
- 13 You will be prompted for the “administration login” (an e*Gate user with sufficient privilege to create components within a schema). The default is **Administrator**; unless you have created a different “administrative” user name, press **Enter** to accept the default. The default password is listed in the README.TXT file in the root directory of the installation CD-ROM.
- 14 Enter and confirm the password for the user specified in the step above.

Note: e*Gate user names and passwords are case-sensitive.

- 15 A menu of add-on options will appear. Type the number corresponding to the add-on package(s) you wish to install (**CGI Web Server e*Way**) and press **Enter**.
- 16 Follow the on-screen instructions to complete the installation.
- 17 After the add-on application has been installed, the “Choose add-on packages” menu will appear. Repeat step 15 to install additional packages, or **0** and press **Enter** to continue.
- 18 When the “installation type” menu appears, the Add-on Applications installation is complete. Do one of the following:
 - To exit the setup utility, type **0** and press **Enter**.
 - Select another option to continue the installation.

2.4.2. Configuring the Participating Host Components in the Schema Manager

Important

From the perspective of the e*Gate GUIs, the CGI Web Server e*Way is not a system of components distributed between the web server and a Participating Host, but a single component that runs an executable file (the multiplexer **stcewipmp.exe**). When this manual discusses procedures within the context of any e*Gate GUI, the term “e*Way” refers only to the Participating Host component of the CGI Web Server e*Way system.

To configure the Participating Host components:

- 1 If you have not already done so, launch the Schema Manager.
- 2 Using the Component editor, create a new e*Way.
- 3 Display the new e*Way’s properties.
- 4 On the General tab, under **Executable File**, click **Find**.
- 5 Select the file **stcewipmp.exe**.

- 6 Click **OK** to close the properties sheet, or continue to configure the e*Way. Configuration parameters are discussed in [Chapter 3](#). The setup and requirements of schemas required to use this e*Way are discussed in [Chapter 4](#).

Note: *Once you have installed and configured this e*Way, you must incorporate it into a schema by defining and associating the appropriate Collaborations, Collaboration Rules, IQs, and Event Types before this e*Way can perform its intended functions. For more information about any of these procedures, please see the online Help system.*

*For more information about configuring e*Ways or how to use the e*Way Editor, see the e*Gate Integrator User's Guide guide.*

2.5 Configuring the Web Server Components

To configure the web server to use the CGI Web Server e*Way web server components:

The web server should execute the client executable, **stcewcgi.exe**, when a request arrives. It also needs to set the dynamic-load library path in order for **stc_common.dll** and **stc_ewipmplt.dll** to be loaded by **stcewcgi.exe**.

A few parameters may also need to be set. These parameters can be set as environment variables by the web server or passed in as CGI input.

Note: *The environment variables overwrite any CGI input values.*

These parameters include:

STC_EW_SERVER_NAME: The name of the machine on which the CGI Web Server e*Way Server is running. If the parameter is not supplied, the default is **localhost**.

STC_EW_SERVER_PORT: The port number on which the CGI Web Server e*Way Server is listening. If the parameter is not supplied, or a value of zero (0) is supplied, the default port number will be used.

STC_EW_SECONDS_TO_EXPIRE: The number of seconds the message will be active in the e*Gate system. If the parameter is not supplied, a value of zero (0) will default, which means the message will remain active indefinitely.

STC_EW_MILLISECONDS_TO_WAIT: The number of milliseconds the CGI Web Server e*Way Client will wait for the response from the CGI Web Server e*Way Server. The CGI Web Server e*Way Client displays an error message should the CGI Web Server e*Way Server fail to respond in the given time period. If the parameter is not supplied, a value of ten thousand (10,000) will be set as the default.

2.6 Files/Directories Created by the Installation

The CGI Web Server e*Way installation process will install the following files within the e*Gate directory tree. Files will be installed within the “egate\client” tree on the Participating Host and committed to the “default” schema on the Registry Host.

Table 1 Server-side files installed

e*Gate Directory	File(s)
\bin\	stcewipmp.exe stc_common.dll stc_ewipmpclnt.dll
\configs\stcewipmp\	stcewipmp.def

Table 2 lists the files that must be installed on the system running the web server:

Table 2 Client-side files installed

Client Directory	File(s)
\bin\	stc_common.dll stc_ewipmpclnt.dll stcewcgi.exe

Note: After installation, change the file permission to allow the web server to read and execute these files.

Configuration

Important: From the perspective of the e*Gate GUIs, the CGI Web Server e*Way is not a system of components distributed between the web server and a Participating Host, but a single component that runs an executable file (the multiplexer `stcewipmp.exe`). When this manual discusses procedures within the context of any e*Gate GUI (such as those in this chapter, which deals in part with the e*Way Editor), the term “e*Way” refers only to the Participating Host component of the CGI Web Server e*Way system.

3.1 e*Way Configuration Parameters

e*Way configuration parameters are set using the e*Way Editor.

To change e*Way configuration parameters:

- 1 In the Schema Manager’s Component editor, select the e*Way you want to configure and display its properties.
- 2 Under **Configuration File**, click **New** to create a new file, **Find** to select an existing configuration file, or **Edit** to edit the currently selected file.
- 3 In the **Additional Command Line Arguments** box, type any additional command line arguments that the e*Way may require, taking care to insert them *at the end* of the existing command-line string. Be careful not to change any of the default arguments unless you have a specific need to do so.

For more information about how to use the e*Way Editor, see the e*Way Editor’s online Help or the *e*Gate Integrator User’s Guide*.

The e*Way’s configuration parameters are organized into a single section: General Settings.

3.1.1. General Settings

The parameters in this section specify the IP port through which e*Gate and the client system communicates.

Request Reply IP Port

Description

Specifies the IP port that the e*Way will listen (bind) for client connections. This parameter is used for Request/Reply behavior.

Required Values

A valid TCP/IP port number between 1 and 65536. The default is **26051**. Normally, you only need to change the default number if the specified TCP/IP port is in use, or you have other requirements for a specific port number.

Push IP Port

Description

Specifies the IP port through which this e*Way allows an external system to connect and receive unsolicited (without submitting a request) Events.

Required Values

A valid TCP/IP port number between 0 and 65536. The default is **0**.

Additional Information

Any Event that this e*Way receives that has zero values for all fields in the 24 byte MUX header is sent to all callers of the **WaitForUnsolicited**. This parameter is optional. If set to zero, the e*Way will follow the Request/Reply scenario and not accept unsolicited Events.

Rollback if no Clients on Push Port

Description

Specifies whether the Event will continually roll back if there are no push clients connected.

Required Values

Yes or **No**. If set to **Yes**, the Event will continually roll back if there are no push clients connected.

Wait For IQ Ack

Description

Specifies whether the send client function does NOT return until the Event is committed to the IQ.

Required Values

Yes or **No**. If set to **Yes**, the send client function does NOT return until the Event is committed to the IQ.

Caution: *This parameter should be set if the data must be committed to the IQ on every transaction before the API returns to the client. Setting this parameter to Yes will*

significantly impact performance. If normal request/reply type transactions are being sent/received, and the data can be recreated at the client, this parameter should not be set.

Send Empty MSG When External Disconnect

Description

Specifies whether the e*Way sends an empty incoming message (containing only the multi-plexer header) when an external client disconnects.

Required Values

Yes or **No**. If set to **Yes**, the e*Way sends an empty incoming message when an external client disconnects.

MUX Instance ID

Description

Specifies whether the specified 8 (eight) bytes is prepended to the 24 (twenty-four) byte session ID of the request received from the external connection before sending to e*Gate.

Required Values

A string. If this value is other than "0", the 8 bytes are prepended to the 24 byte session ID. The default is 0.

Note: *This is a string where "00" and "00000000" are valid MUX Instance IDs, while "0" is to turn this option off. Only the first 8 bytes are used.*

MUX Recovery ID

Description

Specifies whether the 8 bytes are prepended to the reply and republish back to e*Gate provided the value is other than "0" and the multi-plexer finds that the session related to the MUX ID in the return message has been dropped.

Required Values

A string. If this value is other than "0", the 8 bytes are prepended to the 24 byte session ID. The default is 0.

Note: *This is a string where "00" and "00000000" are valid MUX Recovery IDs, while "0" is to turn this option off. Only the first 8 bytes are used.*

3.2 External Configuration Requirements

To enable the client system to communicate with the e*Gate API Kit, you must do the following:

- 1 Install the required client files on the external system (see [“Files/Directories Created by the Installation” on page 13](#)).
- 2 Configure the client components as necessary to use the port specified above in [“Request Reply IP Port” on page 15](#).

3.3 Replacing CGI Payload XML Encoding with Simple Binary Delimiting Logic

The `ewcgi.cfg` file is provided as an option to map the URL to a set of MUX parameters. This eliminates dependence on user input, allows more than one MUX to be accessed from the same server, and eliminates most of the environment variables.

The configuration file, `ewcgi.cfg`, must be in the same directory as `stcewsgi.exe` (typically, the user’s webserver `cgi-bin` directory). The CGI e*Way looks for the configuration file (`ewcgi.cfg`) in the same directory as the executable. If the configuration file exists, then the following format is expected:

```
url|server_host|server_port|expire|wait|header|NO_XML
```

These values perform the following:

- 1 **url** - reads in and stores the url.
- 2 **server host** - overrides any `STC_EW_SERVER_NAME` settings.
- 3 **server port** - overrides any `STC_EW_SERVER_PORT` settings.
- 4 **expire** - overrides any `STC_EW_SECONDS_TO_EXPIRE` settings.
- 5 **wait** - overrides any `STC_EW_MILLISECONDS_TO_WAIT` settings.
- 6 **header** - set to either **G** or **R** (generate or receive).
 - ♦ **G**: generates headers in code before sending data to the Webserver.
 - ♦ **R** (or none): expects the headers to be provided with the data. Be careful with this option because lack of a proper header -- 2 blank lines -- may cause the browser/HTTP client to see a 500 server side internal error, which could be misleading. Also, the backend that generates the header must observe the 2 blank line format.
- 7 **NO_XML**: XML CGI payload suppression, that is, it does NOT forward the CGI payload in XML format, but in the modified format described as follows:

```
Tag [02] Value [03] Tag [02] Value [03] . . . [04] data [05]
```

where [02],[03],[04],and[05] refer to the binary values used as delimiters. The absence of the NO_XML config flag causes the e*Way to revert to the existing XML output logic.

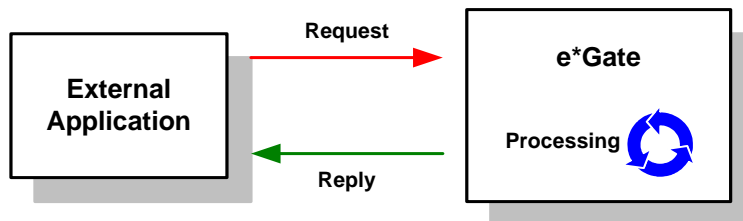
Implementation

4.1 The Request/Reply Concept

All the applications of the CGI Web Server e*Way are based upon the “Request/Reply” concept. At a high-level, this works as follows:

- 1 Request/Reply, where data is sent to the e*Gate system and a response is returned
- 2 Send-only, where data is sent to the e*Gate system but no data is returned
- 3 Receive, where an external system connects to the e*Gate system and allows for the delivery of unsolicited Events

Figure 2 The Request/Reply concept

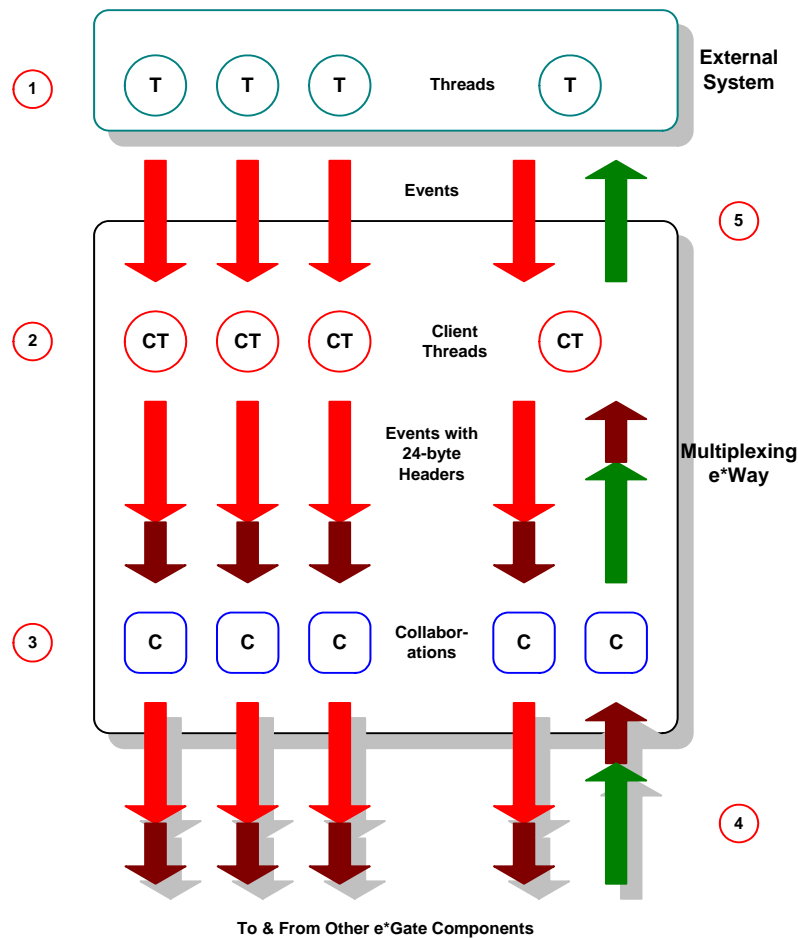


4.1.1. Request/Reply and the CGI Web Server e*Way Participating Host Components

The CGI Web Server e*Way Participating Host component is a multiplexing e*Way that uses a proprietary IP-based protocol to multi-thread Event exchange between the e*Way and external systems or other e*Gate components.

Figure 3 illustrates how the multiplexing e*Way receives data from an external application and returns processed data to the same application.

Figure 3 Data flow through the multiplexing e*Way



- 1 Client threads within the e*Way package the data received from the web server as e*Gate Events, adding a 24-byte header. Among other functions, this header provides "return address" information that can optionally be used to return data to the client thread that originated it.

Each e*Way can handle up to 1,000 client threads at once. If your requirements demand more processing power, you can define more CGI Web Server e*Ways.

- 2 Collaborations within the e*Way perform any appropriate processing that may be required, and route the processed Events to other destinations (such as an external system for additional data retrieval or processing).

Note: The 24-byte header *must* be preserved as the Events are processed through the e*Gate system.

- 3 Processed data, still containing the original 24-byte header, is returned to the CGI Web Server e*Way.
- 4 The e*Way uses the 24-byte "return address" to identify the destination of the data to be returned to the external system.

- 5 The e*Way returns the data, minus the 24-byte header, to the client thread within the web server.

4.1.2. The Request/Reply Schema

The sample schema can be found in the e*Gate installation CD-ROM in the `/samples/ewcgi` directory.

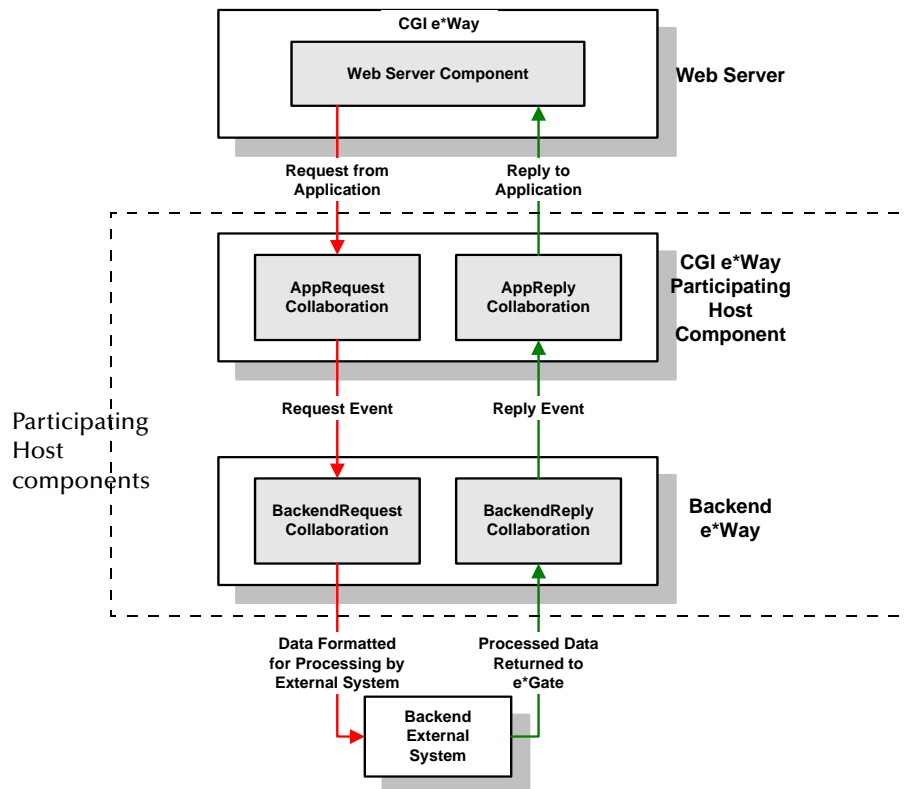
Request/Reply schemas have two classes of components:

- 1 “Front end” components that handle communications with the web server. These components receive requests and route replies to the correct destination.
- 2 “Back end” components that process the requests and compose the replies. These components also provide the bridge between the e*Gate system and your existing systems.

The CGI Web Server e*Way and its related Collaborations comprise the front-end components. A second e*Way and its related Collaborations comprise the back-end components (more e*Ways may be added to communicate with more external systems as required). The backend e*Way(s) can be of any type required to communicate with the external system(s).

Figure 4 below illustrates a typical Request/Reply schema.

Figure 4 The Request/Reply schema



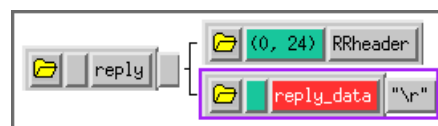
- 1 The user submits data to the web server via a browser form.
- 2 The CGI Web Server e*Way on the web server packages the data, and forwards the data to the Participating Host.
- 3 Data enters the Participating Host through the multiplexing e*Way's WebRequest Collaboration.
- 4 The WebRequest Collaboration publishes the Request Event.
- 5 The BackendRequest Collaboration within the back-end e*Way subscribes to the Request Event, and processes the data as appropriate, and routes it to the external backend system.
- 6 The external backend system processes the data and returns it to e*Gate via the Backend e*Way.
- 7 The Backend e*Way's BackendReply Collaboration publishes the data as the Reply Event.
- 8 The WebReply Collaboration within the multiplexing e*Way subscribes to the Reply Event.
- 9 The multiplexing e*Way returns the processed data to the CGI Web Server e*Way on the web server.
- 10 The web server completes the HTML page and returns the data to the client browser.

4.1.3. ETDs and Form Data

As discussed in [Request/Reply and the CGI Web Server e*Way Participating Host Components](#) on page 19, the CGI Web Server e*Way maintains "return address" information in a 24-byte header that must be preserved as the data flows through the e*Gate system.

The simplest Event Type Definition (ETD) that can be used within a Request/Reply schema has two nodes: one for the header, the second for the remainder of the Event data.

Figure 5 The simplest Request/Reply ETD



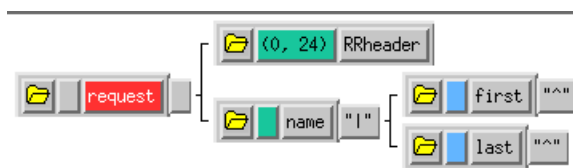
This ETD is sufficient if you wish to send data through the e*Gate system simply as a blob. For example, you can compose the reply to the web server as a block of completely formatted HTML code, then return that reply using the above ETD.

Although the simple ETD in Figure 5 can be sufficient for reply data, request (input) data will probably have a more complex structure. To accommodate this, you must add the additional structure to the basic Request/Reply ETD:

- 1 Be sure to maintain the 24-byte “header” node unchanged.
- 2 Add one subnode beneath the “data” node for each element of input data.
- 3 Modify those subnodes as necessary (for example, to use appropriate delimiters or record lengths).

Figure 6 below illustrates an ETD that describes delimited data (for example, as in the data “First name^Last name”).

Figure 6 A Request/Reply ETD for delimited data

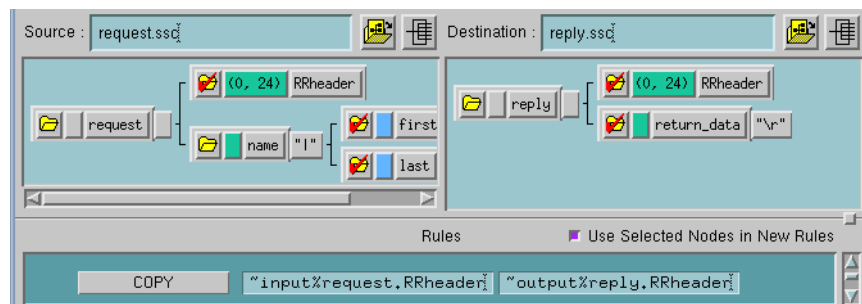


Of course, the more complex the form that collects user data, the more complex the ETD that describes the form data must become. Be sure that the Request ETD meets the requirements for input to the backend system.

Collaboration Rules and the CGI Web Server Header

Collaboration Rules that manipulate data between ETDs must preserve the Request/Reply header (in the figures above, “RRheader”). Be sure that each Collaboration Rule that manipulates Request/Reply data copies the contents of the Request/Reply header from the source ETD to the target ETD (as shown in Figure 7 below).

Figure 7 Copying the Request/Reply header



4.2 Implementing the CGI Web Server e*Way via a web server

The minimal implementation of the CGI Web Server e*Way via a web server requires two components:

- 1 A page that contains a form for accepting user data.
- 2 The **stcewcgi.exe** executable that both sends the user data to the e*Way and posts the results to the user's browser.

The form requires only basic HTML functions to post the data to the server. The ACTION attribute of the <FORM> tag must take the name of the CGI Web Server e*Way client, **stcewcgi.exe**, using the method POST or GET.

You can set the four parameters, `STC_EW_SERVER_NAME`, `STC_EW_SERVER_PORT`, `STC_EW_SECONDS_TO_EXPIRE`, `STC_EW_MILLISECONDS_TO_WAIT`, in the HTML form or set them in the web server environment. (See [“Configuring the Web Server Components” on page 12](#) for more information.)

4.3 Sample Implementation

A sample schema for setting up the CGI Web Server e*Way Server is provided on the e*Gate installation CD-ROM in the directory:

```
/sample/ewcgi
```

- 1 Copy all the files from the e*Gate installation CD-ROM samples directory **/samples/ewcgi** to a temporary working directory on the system on which the CGI Web Server e*Way Server components are installed.
- 2 Copy **sample.html** from the `samples\ewcgi` directory and **stcewegi.exe** from the `\bin` directory to the “CGI bin” directory on the system on which the CGI Web Server e*Way Client components are installed.
- 3 Set the dynamic-load library path which the web server passes to the CGI Web Server e*Way client.
- 4 Change to the temporary working directory on the system where the CGI Web Server e*Way Participating Host components are installed.

Note: *A user cannot run **stcregutl.exe** on machines that do not have the e*Gate Participating Host components installed.*

Client Setup

- 1 Modify the Web server configuration file so that the dynamically loaded library path (`LD_LIBRARY_PATH`, `SHLIB_PATH`, `LIBPATH`, or `PATH`) contains the path of the CGI Web Server e*Way Client dll files, `stc_common.dll` and


```
stc_ewipmpclnt.dll.
```

For example, for the Apache web server on Solaris,

```
setenv LD_LIBRARY_PATH "/usr/egate/client/bin"
```

- 2 Copy the stcewsgi.exe and sample.html to the CGI bin directory.
- 3 Change the permission on stcewsgi.exe, stc_common.dll, and

```
stc_ewipmpclnt.dll
```

so that the web server can read and execute them.

For example, the Apache web server accesses the files as nobody, you can change the permission to allow everyone to read and execute these files.

- 4 Access the sample.html file from a Web browser and send a file to the CGI Web Server e*Way server. If everything works you would see the file you send to the server displayed.

eGate Schema Set up (Use Multiplexer eWay request reply schema)

Request Reply Sample

- 1 Install the CGI Web Server e*Way Server add on.
- 2 Import the sample schema.

Index

C

- components 6
- Configuration parameters
 - Push IP Port 15
 - Request Reply IP Port 15
- configuring the participating host 11
- configuring the web server 12

D

- delimited data, handling in ETDs 23

E

- ETDs, sample 23
- Event Type Definitions, sample 23

F

- files created by installation procedure 13
- files/directories created by install 13

H

- header, in Collaboration Rules 23

I

- implementing CGI 24
- installation
 - files/directories created 13
- installing the CGI Web Server e*Way 8
- intended reader 5

M

- maximum client threads per e*Way 20
- MUX Instance ID 16
- MUX Recovery ID 16

P

- product overview 5
- Push IP Port 15

R

- Request Reply IP Port 15
- request/reply
 - header, in Collaboration Rules 23
 - overview 19
 - schema 21
- Rollback if no Clients on Push Port 15

S

- sample implementation 24
- schema for request/reply configuration 21
- Send Empty MSG When External Disconnect 16
- Supporting Documents 8
- system requirements 6

U

- Unix install 10

W

- Wait For IQ Ack 15