CDEV Gateway

A Message Router/Concentrator for CDEV Clients

Walt Akers, Chip Watson, Jie Chen

Version 1.0 - February 13, 1997

TJNAF - Thomas Jefferson National Accelerator Facility

DOCUMENT DATE: Table of contents generated: February 13, 1997 2:55 pm

TRADEMARKS: UNIX is a registered trademark of AT&T in the USA and other countries.

VxWorks is a register trademark of Wind River Systems.

The X Window System is a trademark of Massachusetts Institute of Technology.

OSF/Motif and Motif are trademarks of Open Software Foundation, Inc.

Ultrix and DEC are registered trademarks of Digital Equipment Corporation.

HPUX is a registered trademark of Hewlett Packard.

SURA/CEBAF: The Southeastern Universities Research Association (SURA) operates the Continuous Electron Beam Accelerator Facility (CEBAF) for the United States Department of Energy under contract DE-AC05-84ER40150.

Table of Contents

List of Figures

1. CDEV Gateway Fundamentals

Overview of the CDEV Gateway

The CDEV Gateway is a server that acts as an intermediary between a CDEV application and remote services. A single network connection is established between the client and the gateway through which all CDEV requests will be routed. When the gateway receives a request from the client, it establishes the appropriate connection to the actual service, makes the request, and returns the result to the client application.

Purpose of the CDEV Gateway

The CDEV Gateway was designed to support the JAVA/CDEV API. Because JAVA code must be small in order to load and execute efficiently, it was critical to move as much code as possible out of JAVA and into a CDEV "proxy" which could perform connection management and dynamic library loading. This allows the JAVA application to access new services provided in C++ without adding any new code.

The JAVA/CDEV library locates the CDEV Gateway by using a hostname and port that is provided in an environment variable. After determining the location of the server, JAVA will establish a connection to the gateway and then begin transmitting its CDEV requests in standard CLIP (CDEV Linear Internet Protocol).

The gateway can also be used by standard C or C++ CDEV applications to perform message routing. This approach can lessen the number of connections to actual control system devices and reduce network traffic significantly. The gateway is located by the gateway service by looking up its name (as specified in the CDEV DDL) in the global name server (The current name server is distributed as part of the CDEV Generic Server in the CDEV distribution).

Features of the CDEV Gateway Service

- The CDEV Gateway is constructed using the CDEV Generic Server Engine and the CLIP Protocol.
- Each gateway server registers its name, host and port with the CDEV Name Server when it is started.
- The name of the specific gateway to be used is associated with a CDEV device/message combination by using the "server" field of the service data in the CDEV DDL file. Syntax is described later in this document.
- The gateway service will establish one connection to a named gateway to perform all transactions.
- The client application uses a special DDL file that redirects all signals to the gateway service. This allows the gateway server to maintain the actual control system connections from a centralized location.
- The gateway server may have a standard DDL that redirects requests to the actual service OR it may have a special DDL that routes requests to another gateway. The system has currently been tested by routing messages through as many as ten gateways before reaching the actual control system. Note that performance begins to degrade substantially after routing through five gateways - but is still reasonable.

2. Building the CDEV Gateway Service

Location of Files

The source code for the Gateway Service is provided with the CDEV distribution starting with version 1.5. The source code is located in the directory $CDEV/applications/cdevGateway. The following steps should be taken to build the Gateway Server and Service.

Steps to Building the Server and Service

1. Obtain and install the CDEV distribution - Version 1.5 or higher.

2. Follow the installation steps to build CDEV on your system. This includes setting all of the environment variables required by CDEV.

3. Follow the installation steps to build the CDEV Generic Server and deploy the CDEV Name Server.

4. There is a collection of platform specific makefiles in the directory *$CDEV/extensions/cdevGenericServer/include/makeinclude*. Link the makefile that is appropriate to your architecture to the file Makefile.OS in THAT directory.

   For example, to link to the makefile for HP-UX, you would type the following command.

   cd $CDEV/extensions/cdevGenericServer/makeinclude
   ln -s Makefile.hpux Makefile.OS

5. Change directories to *$CDEV/applications/cdevGateway*.

6. Type make to compile and link the CDEV Gateway Service shared object and the CDEV Gateway Server. Note that you must be using GNU make in order to build this distribution.

Upon completion the Makefile will generate the shared object for the gateway service and will install it into your CDEVSHOBJ directory. The gateway server will be compiled, linked and installed in the directory $CDEV/applications/cdevGateway/bin/ARCH/cdevGateway. At this point you are ready to start the gateway server.

3. Starting the CDEV Gateway

Starting the Gateway Server

The following steps should be taken to execute the cdevGateway.

1. Ensure that the required CDEV and CDEV Generic Server environment variables have been set. These include:

CDEVDDL	This is the name of the DDL file that indicates where messages should be routed. Typically, this DDL will route the messages to the actual control system devices.
CDEVSHOBJ	This is the location of the CDEV shared object files.
CDEV_NAME_SERVER	This is the host where the CDEV Name Server is running.

2. Move to the directory where the CDEV Gateway Server binary is located: by default $CDEV/applications/cdevGateway/bin/ARCH, where ARCH is the name of the host architecture.

3. Type the command ./cdevGateway at the command line. This will start a gateway server with the name ***Gateway1*** on the current host using TCP/IP port 9573. This information can be confirmed by using the ***dumpServer*** command that is provided with the CDEV Generic Server Engine to display the contents of the CDEV Name Server.

At this point the cdevGateway should be ready for use by client applications.

Start-up Arguments for the Gateway Server

The user can specify the name of port used by the CDEV Gateway from the command line when the server is started. The gateway server will accept the following parameters from start-up.

s=*server name*	This allows the caller to specify the server name that will be registered with the name server. By default the server name is ***Gateway1***.
p=*port number*	This allows the caller to specify the port that the server will use to receive incoming requests. By default the port used by the server is ***9573***.

4. Troubleshooting Server Start-up

When the server is started it may display one or more error conditions that will prevent it from operating properly.

Error: *syntax error at line:#*

Meaning: This message indicates that the CDEVDDL file that is being used by the gateway is formatted improperly.

Solution: Edit the CDEVDDL file and correct any formatting errors that may exist.

Error: *ERROR: Gateway1 server already exists in GATEWAY domain*

Meaning: This message indicates that a gateway server with the name Gateway1 has already been started and is currently registered with the CDEV Name Server.

Solution: The gateway will continue to attempt to register itself with the name server every 30 seconds. If the message continues to appear, then restart the gateway using the s parameter to specify a different server name OR shutdown the other gateway that is using the same name.

Error: *Could not communicate with name server*
ERROR: Failed to execute "REGISTER SERVER" operation

Meaning: This message indicates that the CDEV Name Server could not be accessed either because it is not running or because the CDEV_NAME_SERVER environment variable does not specify the correct host.

Solution: Ensure that the CDEV_NAME_SERVER environment variable is set and indicates the host where the name server is running. If this is correct, check to ensure that the name server is actually running on the specified host.

Error: *Error: Server Gateway1 - Unable to open listening socket*

Meaning: This message indicates that the port that the name server is attempting to use to listen for new connections is already in use.

Solution: Use the command line arguments to specify a different listening port.

5. DDL Entries for the CDEV Gateway Client

Setting Up the DDL

The following example illustrates how to add a service entry to identify the CDEV Gateway Service and how to use the service data to specify the server that the gateway service will use for routing messages.

Figure 1: Simple DDL file for the Gateway Service

```
/*
 * This is the service definition, it tells CDEV that there is
 * a service named Gateway and that it will use the cdevData
 * tag 'server' to locate the gateway using the CDEV Name Server.
 */
service Gateway
  {
  tags {server}
  }

/*
 * This class definition identifies the class gatewayClass which
 * supports the "get","set", "monitorOn" and "monitorOff"
 * commands on attributes "attrib0" and "attrib1". Note that
 * requests made to attrib0 will be routed through "Gateway1"
 * and requests made to attrib1 will be routed through
 * "Gateway2".
 */
class gatewayClass
  {
    verbs {get, set, monitorOn, monitorOff}
  attributes
      {
      attrib0 Gateway {server=Gateway1};
      attrib1 Gateway {server=Gateway2};
      }
}

/*
 * Finally, the CDEV DDL identifies two devices that are of type
 * gatewayClass. Any time one of the supported messages is passed
 * to one of these devices it will be routed through a CDEV
 * Gateway.
 */
gatewayClass : device0, device1;
```