

# Gcom Traffic Director

## Introduction

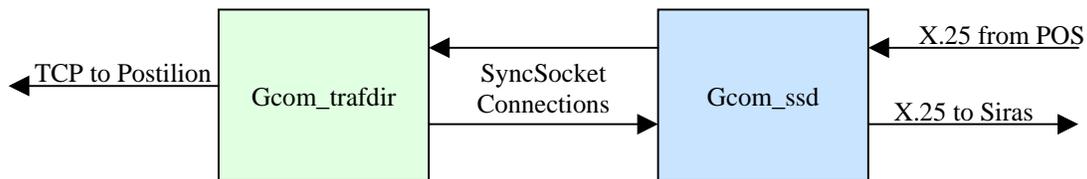
The Gcom Traffic Director is an application level program that directs transaction traffic to one of two destinations depending upon the form of the transaction messages. It receives X.25 calls from a Point of Sale (POS) system and forwards the data packets either to the Siras host via X.25 or the Postilion host via TCP/IP.

The program, Gcom\_trafdir, bases its decision either on the setting of the D-bit in the X.25 data packet from POS or on certain fields in the message data.

This document describes this program, how to run it, and how it handles its connections.

## Traffic Director in Context

The following figure illustrates the place occupied by Gcom\_trafdir in the Gcom software system.



In the figure, the arrows represent the direction of connection initiation. An X.25 connection comes into Gcom\_trafdir from the POS via Gcom\_ssd. Gcom\_trafdir uses SyncSocket connections to Gcom\_ssd as the means of communicating via X.25.

Gcom\_trafdir makes its routing decision and then initiates a connection to either Siras or Postilion. The Siras connection is an X.25 connection via Gcom\_ssd. The Postilion connection is a TCP/IP connection.

Not shown in the figure are the actual X.25 and TCP/IP protocol stacks. Both of these stacks are in the operating system kernel. Both Gcom\_trafdir and Gcom\_ssd are application level programs which use APIs to communicate via TCP (in the case of Gcom\_trafdir) and X.25 (in the case of Gcom\_ssd).

If you would like to see more details in the relationship between Gcom\_trafdir and the rest of the Gcom system visit the Gcom Appliance Solution Architecture web page at <http://www.gcom.com/products/gcom-protocol-appliance-2g/appliance-solution-architecture.html>. Gcom\_trafdir occupies the same position as the application Gcom\_pcd in that illustration. This web page contains numerous links to more detailed information about individual components of the architecture.

## Traffic Director Operation

The internal operation of Gcom\_trafdir proceeds as follows.

It listens for SyncSocket connections from Gcom\_ssd. These connections are presumed to be X.25 connections from the POS.

Whenever it receives a connect request from Gcom\_ssd it simultaneously sends a connect confirm back to Gcom\_ssd and initiates a TCP connection to Postilion. Since the overwhelming majority of connections from POS get routed to Postilion it is highly likely that this Postilion connection will be utilized. By initiating the connection early Gcom\_trafdir is able to get that connection setup in parallel with operations with POS.

Gcom\_trafdir receives data packets from POS. It first combines all packets with the M-bit set into one large message. Once that message is built Gcom\_trafdir examines the message to determine whether to route it to Postilion or to Siras. There are two alternatives for making this decision. The first is to base the decision on the value of the D-bit in the final packet of the M-bit sequence. The second is to base the decision on the contents of the message itself.

Gcom\_trafdir makes the decision as to where to route the message. It queues the message for either Postilion or Siras and initiates a connection, if necessary. In the case of Postilion the connection setup is underway already and may already be complete. In the case of Siras Gcom\_trafdir opens a SyncSocket connection to Gcom\_ssd and issues a connect request to Siras.

When the outgoing connection completes, either to Postilion or to Siras, Gcom\_trafdir forwards the message on the appropriate connection.

If a message arrives from Postilion it is forwarded to the POS as a sequence of X.25 packets with the M-bit set (except, of course, for the last packet). If packets arrive from Siras they are accumulated into a message using the X.25 M-bit. Once the message is assembled it is transmitted to the POS as a sequence of X.25 packets in the same manner as for messages from Postilion.

Each message received from POS is routed independently. So the POS may interleave messages to Postilion and to Siras and Gcom\_trafdir will hold both outgoing connections open and utilize them for this purpose.

When the POS clears the X.25 connection Gcom\_trafdir receives a SyncSocket disconnect request message from Gcom\_ssd. Gcom\_trafdir then closes both the TCP socket for Postilion and the SyncSocket connection for Siras. The kernel TCP stack takes care of closing the TCP connection and Gcom\_ssd takes care of clearing the X.25 connection to Siras.

### **Exchanging Data with Postilion**

The connections to Postilion are TCP connections. TCP connections are what is referred to as “byte stream” connections. This means that data sent over a TCP connection have no inherent message boundaries. TCP regards all data as a stream of bytes.

In order to preserve the message boundaries of transactions received from the POS it is necessary to place a two byte header in front of each message sent to Postilion. The two bytes specify the number of bytes contained in the following message. This allows Postilion to recreate the original message boundaries for the transactions that it is receiving. The first byte gives the high order 8 bits of the message length and the second byte gives the low order 8 bits of the message length.

A command line option for Gcom\_trafdir (-i y) will cause this message length header to be incremented by two so that it includes the bytes of the header as well as the bytes of the message itself. The default is to set the length header to indicate only the length of the message (-i n).

Gcom\_trafdir uses Gcom's Data Tunneling API (DT API) to manage the TCP connections to Postilion. The DT API takes care of the length header "encapsulation" of the messages as they are sent and received. For a list of all the different types of encapsulation headers that DT API implements see the following web page. <http://www.gcom.com/products/gcom-protocol-appliance-2g/architecture/gcom-header-format-summaries.html>.

Messages received from Postilion contain the same length header as messages sent to Postilion. This allows Gcom\_trafdir to establish the boundaries for these messages prior to forwarding them to the POS.

### **Connection Failures**

Gcom\_trafdir passes connection failures with Postilion and Siras back to the POS in the form of an X.25 clearing sequence. The handling of connection failure is slightly different for Postilion and for Siras.

In the case of Postilion the connection is initiated as soon as Gcom\_trafdir receives the SyncSocket connect request from POS via Gcom\_ssd. This connection setup is considered provisional until Gcom\_trafdir makes an actual routing decision in favor of Postilion. If the connection to Postilion fails while still in the provisional mode the failure is not passed back to POS. However, if the connection fails once the routing decision has been made then the connection to the POS is cleared.

In the case of Siras the connection is not initiated until the routing decision has been made. Therefore, any failure of the Siras connection results in clearing the X.25 connection to POS.

### **Routing Decision Making**

A command line parameter determines whether Gcom\_trafdir uses the X.25 D-bit or the message contents as the basis for its routing decision.

If it is using the X.25 D-bit then a message with the D-bit on will be routed to Postilion. A message with the D-bit off will be routed to Siras.

If it is using the message contents to make the decision it proceeds as follows. Gcom\_trafdir extracts 5 bytes from the message beginning with byte offset 30. It assumes that these 5 bytes are coded in ASCII. It then compares these bytes, considered as a single string, with the contents of a command line parameter (-T, see below). If the string from the message can be found within the parameter string then the message is routed to Postilion, otherwise it is routed to Siras.

### **Invocation and Parameters**

Gcom\_trafdir reside in the directory /usr/bin. The Gcom Management Console (GMC) is used to set the parameters for the program and to start it along with the other Gcom related programs that the GMC starts.

Gcom\_trafdir accepts the following parameters on the command line. The GMC has special constructs for the setting of these parameters.

Option	Description
-a <i>DTE-address</i>	The address to use when connecting to Siras.
-B	Run in background mode. Default is foreground.
-c <i>conn-name</i>	Specifies the connection name that Gcom_trafdir will use when making an outgoing connection to Siras via Gcom_ssd. The name must match a connecting (non-listening) type connection definition for Gcom_ssd. Default is "siras".
-C <i>config-file</i>	Specifies the name of a configuration file from which configuration information will be used. NOTE - parameters specified in a configuration file will override command line arguments.
-d <i>number</i>	Bit mask setting debug options for Gcom_trafdir.
-D <i>number</i>	Bit mask setting debug options for the DT API used by Gcom_trafdir for TCP connections to Postilion.
-e	Treat incoming connection with the DBIT UNSET as error and disconnect the connection to POS. Note that this pertains to the connect request and not to the received data packets.
-f <i>filename</i>	Name of the log file used by Gcom_trafdir. Default is /usr/spool/gcom/trafdir.log.
-F <i>filename</i>	Name of the log file used by the DT API within Gcom_trafdir. Default is /usr/spool/gcom/trafdir_dt.log.
-h or -H	Print help text.
-i <i>y</i> or <i>n</i>	Include ( <i>y</i> ) or do not include ( <i>n</i> ) the size of the header in the size field of the encapsulation header used for messages sent to Postilion over a TCP connection.
-l <i>conn-name</i>	Specifies the connection name that Gcom_trafdir will listen on for incoming X.25 calls from the POS. This name must match a listening connection definition for Gcom_ssd. Default is "pos".
-p <i>ssid-port^host:port</i>	Specifies the port number used to connect to the SSD and and the IP address and port used to connect to Postilion respectively. The ssid-port should always be set to 8000.
-P <i>filename</i>	Specifies the name of a file in which the daemon name and daemon process id (PID) will be written. The default is trafdir.pid.
-q <i>logsize</i>	Set the maximum size of Gcom_trafdir's log in kilobytes. The same value is used for the size of the DT API log.
-t <i>number</i>	Specifies the maximum amount of time for which a connect confirmation from Siras will be waited for before the connection attempt is aborted (default is 10 seconds).
-T <i>tid[,tid...]</i>	The argument is a comma separated list of transaction type mnemonics that will be used to determine whether a message is to be routed to Postilion or to Siras. As described above, a transaction with an id that matches one of these will be routed to Postilion with

	all others being routed to Siras. The appearance of this parameter enables transaction id based routing and inhibits routing based upon the X.25 D-bit setting.
-v	Print version information.

### Configuration File

Parameters may be specified to Gcom\_trafdir in a configuration file instead of on the command line. The -C command line argument causes Gcom\_trafdir to read parameters from the named file.

The format of the file is as follows:

```
global:
    Param1 = value
    Param2 = value
    Etc.
```

The word “global” must appear in the file and be position in column one. The word must be followed by a colon.

Parameters are indented after the line containing the global label. The parameters correspond with command line arguments. The following table shows the correspondence between parameter names and command line argument mnemonics.

Parameter	Value Type	Command Line Mnemonic
run-in-background	Number	-B
max-log-size	Number	-q
ssd-port	Number	-p (ssd-port portion)
tcp-host-port	String	-p (host:port portion)
x25-listen-conn-name	String	-l
x25-connect-conn-name	String	-c
x25-dte-addr	String	-a
include-hlen	Number	-i
require-conn-req-dbit	Number	-e
connect-timeout	Number	-t
logopts	Number	-d
transaction-list	String	-T