ACKNOWLEDGEMENT

By utilizing this website and/or documentation, I hereby acknowledge as follows:

Effective October 1, 2012, QUALCOMM Incorporated completed a corporate reorganization in which the assets of certain of its businesses and groups, as well as the stock of certain of its direct and indirect subsidiaries, were contributed to Qualcomm Technologies, Inc. (QTI), a wholly-owned subsidiary of QUALCOMM Incorporated that was created for purposes of the reorganization.

Qualcomm Technology Licensing (QTL), the Company's patent licensing business, continues to be operated by QUALCOMM Incorporated, which continues to own the vast majority of the Company's patent portfolio. Substantially all of the Company's products and services businesses, including QCT, as well as substantially all of the Company's engineering, research and development functions, are now operated by QTI and its direct and indirect subsidiaries[1]. Neither QTI nor any of its subsidiaries has any right, power or authority to grant any licenses or other rights under or to any patents owned by QUALCOMM Incorporated.

No use of this website and/or documentation, including but not limited to the downloading of any software, programs, manuals or other materials of any kind or nature whatsoever, and no purchase or use of any products or services, grants any licenses or other rights, of any kind or nature whatsoever, under or to any patents owned by QUALCOMM Incorporated or any of its subsidiaries. A separate patent license or other similar patent-related agreement from QUALCOMM Incorporated is needed to make, have made, use, sell, import and dispose of any products or services that would infringe any patent owned by QUALCOMM Incorporated in the absence of the grant by QUALCOMM Incorporated of a patent license or other applicable rights under such patent.

Any copyright notice referencing QUALCOMM Incorporated, Qualcomm Incorporated, QUALCOMM Inc., Qualcomm Inc., Qualcomm or similar designation, and which is associated with any of the products or services businesses or the engineering, research or development groups which are now operated by QTI and its direct and indirect subsidiaries, should properly reference, and shall be read to reference, QTI.

[1] The products and services businesses, and the engineering, research and development groups, which are now operated by QTI and its subsidiaries include, but are not limited to, QCT, Qualcomm Mobile & Computing (QMC), Qualcomm Atheros (QCA), Qualcomm Internet Services (QIS), Qualcomm Government Technologies (QGOV), Corporate Research & Development, Qualcomm Corporate Engineering Services (QCES), Office of the Chief Technology Officer (OCTO), Office of the Chief Scientist (OCS), Corporate Technical Advisory Group, Global Market Development (GMD), Global Business Operations (GBO), Qualcomm Ventures, Qualcomm Life (QLife), Quest, Qualcomm Labs (QLabs), Snaptracs/QCS, Firethorn, Qualcomm MEMS Technologies (QMT), Pixtronix, Qualcomm Innovation Center (QuIC), Qualcomm iSkoot, Qualcomm Poole and Xiam.

QUALCOMM® CDMA Technologies

QMI Quality of Service Test Application

80-VT625-1 A

October 8, 2009

Submit technical questions at:
https://support.cdmatech.com

QUALCOMM Incorporated
5775 Morehouse Drive
San Diego, CA 92121-1714
U.S.A.

Contents

Figures

Tables

1 Introduction

1.1 Purpose

This document describes the Qualcomm MSM™ Interface (QMI) Quality of Service (QoS) Test Application and is intended for use by infrastructure vendors, handset vendors, and carriers to test QoS functionality on 1xEV-DO Rev A.

1.2 Scope

This document contains a description of the functionality of the QMI QoS Test Application (QMI QTA). The main functions supported by QMI QTA are described and a sample call flow is provided to understand the test environment and call flow.

1.3 Conventions

Function declarations, function names, type declarations, and code samples appear in a different font, e.g., `#include`.

Button and key names appear in a different font, e.g., click EDIT or press ENTER.

1.4 Revision history

The revision history for this document is shown in Table 1-1.

Table 1-1 Revision history

Version	Date	Description
A	Oct 2009	Initial release

1.5 References

Reference documents, which may include QUALCOMM®, standards, and resource documents, are listed in Table 1-2. Reference documents that are no longer applicable are deleted from this table; therefore, reference numbers may not be sequential.

Table 1-2 Reference documents and standards

Ref.	Document	
Qualcomm		
Q1	*Quality of Service (QoS) Feature for 1xEV-DO Revision A*	80-VB296-1
Q2	*1xEV-DO Release A Data Over Signaling (DoS) Feature Definition Document*	80-V2510-1
Q3	*Application Note: QTP-6800 CDMA Test Mobile Software Download Guidelines*	80-B5655-1
Q4	*Application Note: Software Glossary for Customers*	CL93-V3077-1
Standards		
S1	*cdma2000 High Rate Packet Data Air Interface Specification*	3GPP2 C.S0024-A Version 2.0 (July 2005)
S2	*cdma2000 Wireless IP Network Standard*	IS-835-D
S3	*Administration of Parameter Value Assignments for cdma2000 Spread Spectrum Standards*	3GPP2 C.R1001-E Version 1.0 (September 2005)
S4	*RTP: A Transport Protocol for Real-Time Applications*	IETF RFC 1889
Software package released by QCT		
T1	*QOS Data Server (ADP Server/ONC/RPC)*	HK11-VC445-1 EXE

1.6 Technical assistance

For assistance or clarification on information in this guide, submit a case to Qualcomm CDMA Technologies at https://support.cdmatech.com/.

If you do not have access to the CDMATech Support Service website, register for access or send email to support.cdmatech@qualcomm.com.

1.7 Acronyms

For definitions of terms and abbreviations, refer to [Q4].

2 QMI QTA Overview

2.1 QMI QTA capability

QMI QTA allows you to perform a variety of tests to verify QoS functionality and performance using different data transfer options over different QoS flows in the following areas:

- QoS as defined in [Q1]
- Application data performance with different air interface parameters

2.1.1 QoS test capability

QMI QTA allows you to:

- Request/configure QoS flows and filters (TFTs) using the QoS specification configuration file
- Suspend, resume, and release QoS flows individually or bundled
- View the current status of QoS flows

Details of these capabilities are described in Chapter 4.

2.1.2 Data transfer capability

QMI QTA allows you to:

- Start multiple unidirectional/bidirectional data transfers over different QoS flows or Best Effort (BE) flows
- Change data transfer parameters to facilitate different protocol, data rate, packet size, and direction of data transfer
- Start multiple data transfer simultaneously
- Send/receive data to multiple-destination IP addresses simultaneously
- Configure data transfer options using a data configuration file

Details of these capabilities are described in Chapter 5.

Limitations and future enhancements of QMI QTA are described in Chapter 7.

2.1.3 Performance-related statistics

QMI QTA provides the ability to log performance-related statistics for:

- Throughput achieved during the data transfer; both calculated by QMI QTA and the Data Server application
- Total number of bytes transferred
- Jitter in milliseconds for UDP data transfer, as defined in [S4]
- Packets dropped on each link

2.2 Test setup

Figure 2-1 illustrates a network diagram for setting up testing of QoS with QMI QTA.

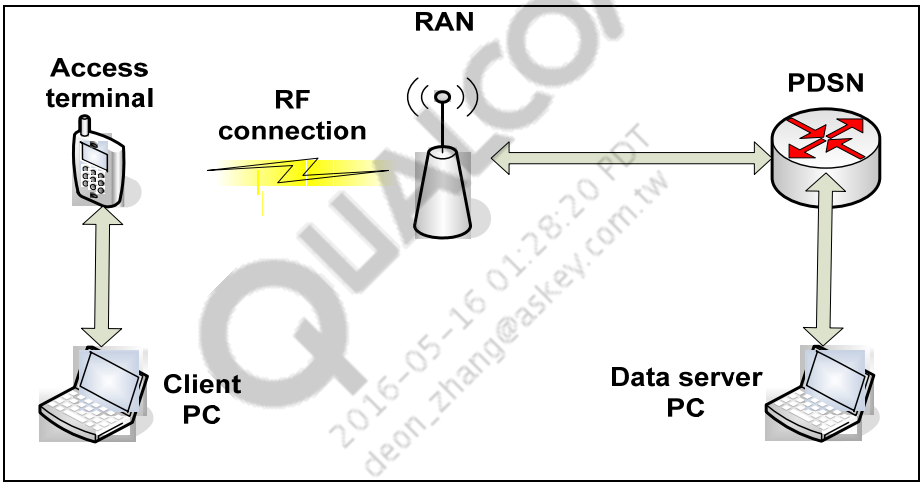

Figure 2-1 Network diagram for semi-dynamic QoS test setup

2.2.1 Access terminal

The Access Terminal (AT) is Qualcomm's MSM6800™ ASIC with QMI software.

2.2.2 Client PC

The client PC is any Windows 2000 or Windows XP PC that has the following:

- USB drivers compatible with the FFAs used for testing
- QPST™
- Diagnostic tools (QXDM or CAIT®) provided by Qualcomm
- QMI QoS Test application installed

2.2.3 RAN

The RAN should be compatible with [S1], [S2], [Q1], and [Q3].

2.2.4 PDSN

The PDSN should be compliant with [S2] and support the call flows described in [Q2] for testing interoperability between the AT and the RAN for RSVP signaling, as defined in [S2].

2.2.5 Data Server

The Data Server [T1] is an application that runs on any Windows 2000 or XP environment. The Data Server is provided by Qualcomm and has support for the following functionality:

- Communicate with QMI QTA to set up data transfers between the FFA and data server PC
- Serve as data endpoint for data transfer to and from QMI QTA
- Ability to set up and tear down single or multiple bidirectional and unidirectional data transfers
- Collect statistics and report them to QMI QTA

2.2.5.1 Data Server PC requirements

The requirements for the Data Server PC are as follows:

- Processor – PC with 1 GHz or higher processor recommended for Windows 2000 or XP installation
- Operating system – Windows 2000 or XP
- Memory – 512 MB RAM

2.3 Call flow with QMI QTA

Figure 2-2 illustrates a call flow describing the steps involved in testing QoS with QMI QTA. This is an example of a simple call flow. Variations of this call flow can be generated by changing the sequence of steps at the application layer, as well as by modifying the responses sent by the AN and the PDSN.

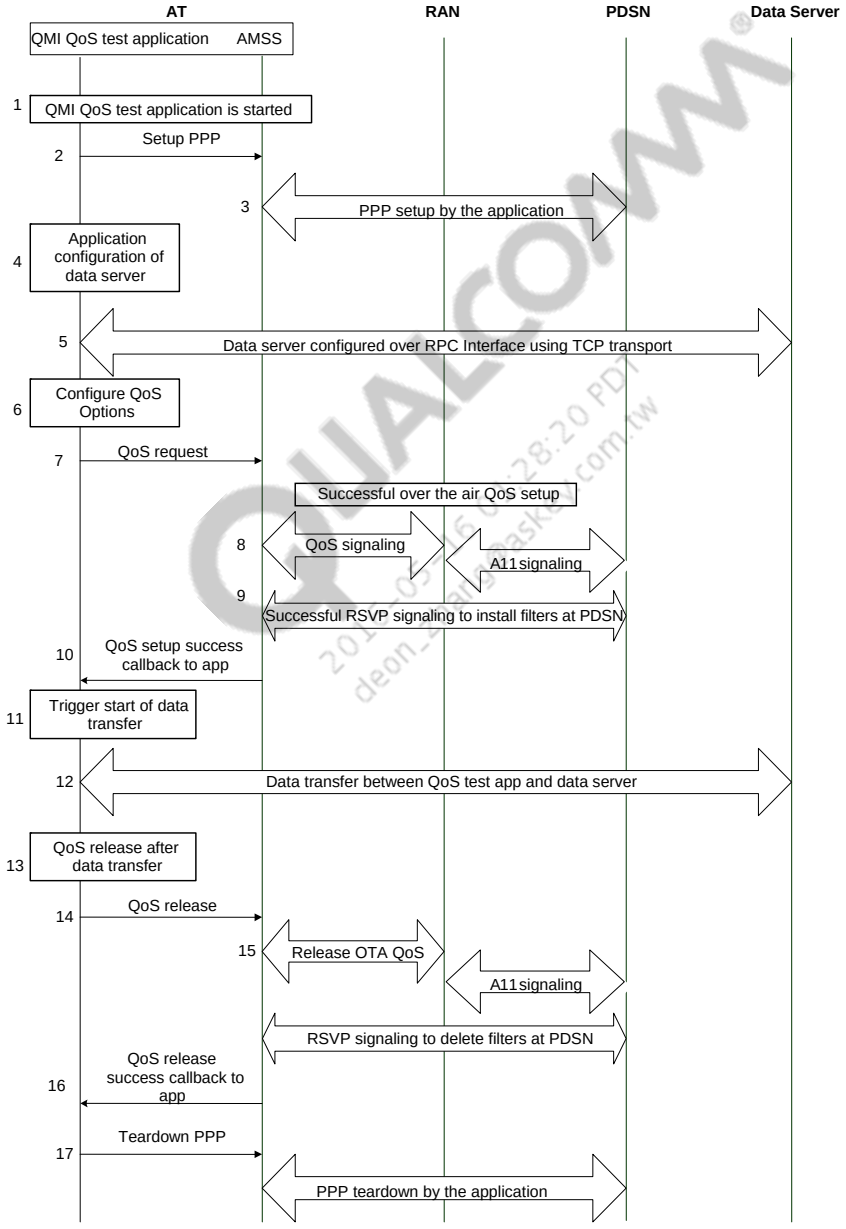

Figure 2-2 Call flow describing QoS

The following numbered paragraphs correspond to Figure 2-2:

1 – 3 QMI QTA starts and sets up a PPP connection.

4 – 5 Data transfer settings are configured using configuration files or manual entry of parameters using the keypad. When the data transfer parameters are set up, a connection is made to the data server. QMI QTA sends commands to the data servers over an RPC interface and prepares the data server for data transfers using the configuration file.

6 – 7 QoS configuration (flow specification and filter specification; refer to Chapter 4 for details) is chosen using preconfigured configuration files, and the application triggers a QoS request.

NOTE QoS setup can be triggered before the data transfers are set up.

8 – 10 The QoS request by the application, in turn, triggers QoS setup between the RAN and the AT (step 8) and RSVP signaling between the AT and the PDSN to install filters for forward link data transfer at the PDSN (step 9). If both steps 8 and 9 are completed successfully, the application will get an indication that the requested QoS is set up. The user receives the information via the user interface.

NOTE Signaling involved in steps 8 and 9 can occur in parallel.

11 – 12 After QoS is set up, the data transfer can be started between the AT and the data server. You can monitor the progress of the data transfer from the user interface.

13 – 16 After the completion of data transfer, QoS release can be invoked from the application. This will, in turn, trigger a release of OTA QoS resources and uninstallation of the filters previously installed at the PDSN.

17 The PPP is torn down by the application.

NOTE Instead of step 17, you can choose to repeat steps 1 through 16 with a different set of parameters.

3 Connection Setup

Connection setup consists of the following steps:

1. Start QMI QTA.

   The first tab that appears is the Connection Manager, as shown in Figure 3-1.

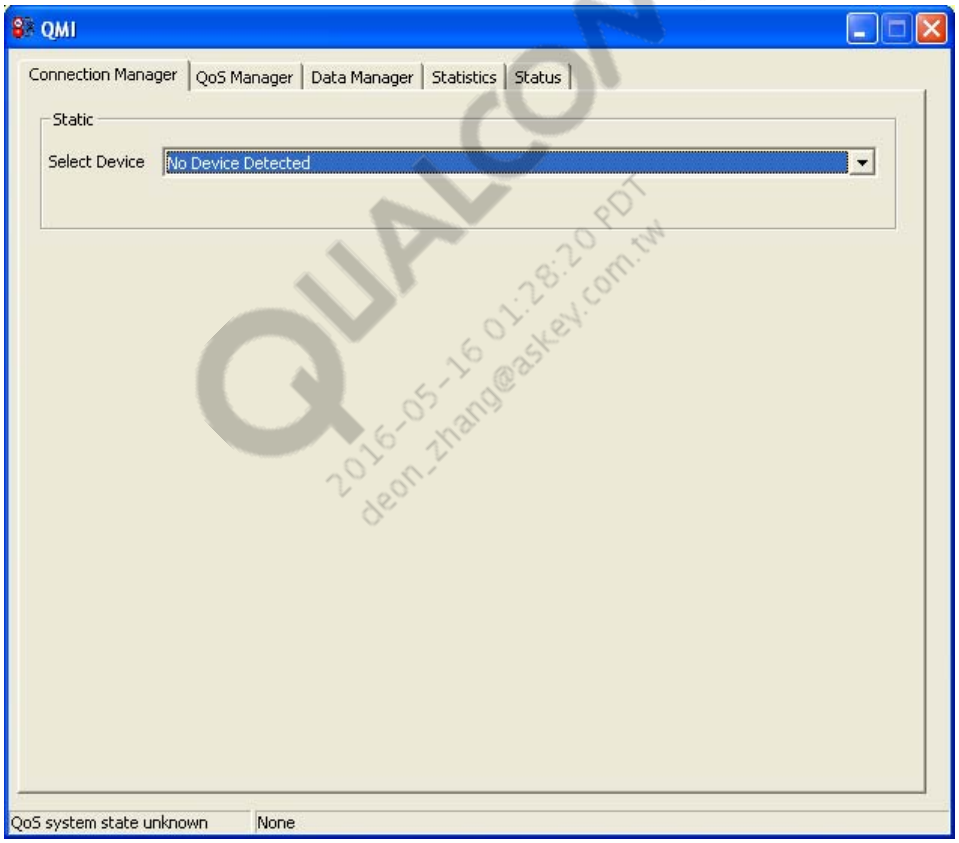

Figure 3-1 Connection Manager

2. Connect a phone via USB to the PC that is running QMI QTA.

   When the phone is connected, QMI QTA automatically populates the Select Device dropdown list, as shown in Figure 3-2.

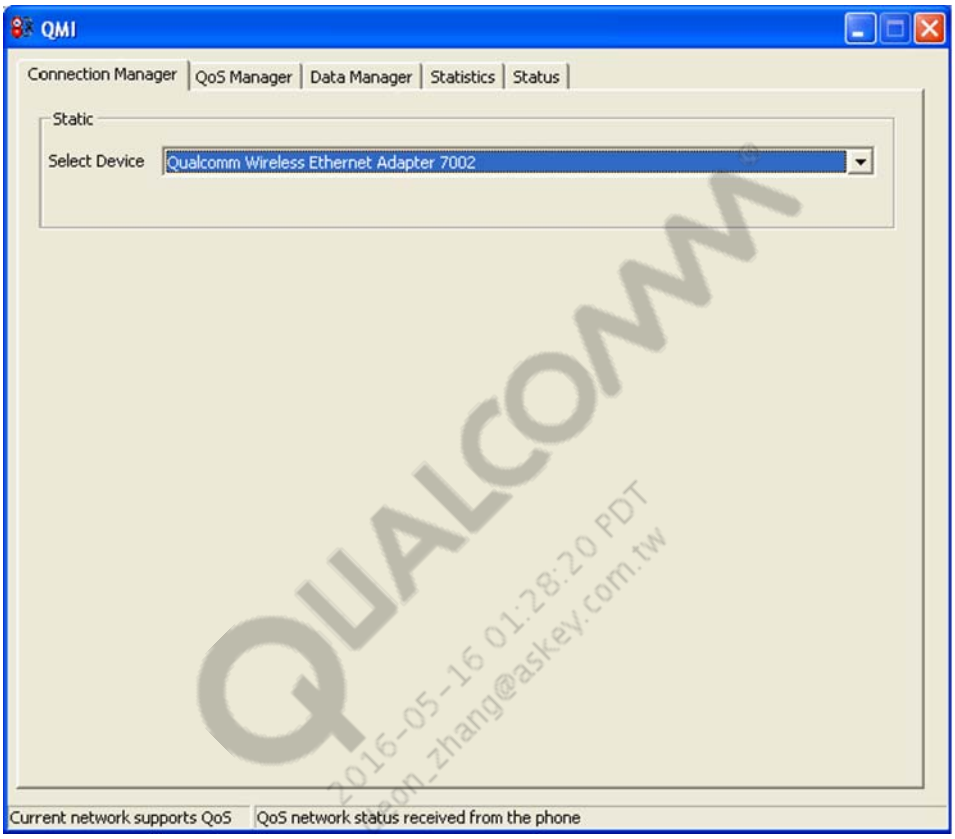

Figure 3-2 Select Device

3. Select a phone from the list.

   QMI QTA sends a request to the phone to determine if the phone is in a QoS aware state and the status is displayed.

4 QoS Setup and QoS Functions

QMI QTA provides the flexibility to configure options that are specific to the QoS feature. QoS options consist mainly of a description of the following areas:

- QoS flow specification – Information in the QoS flow specification is used to set up the QoS between the RAN and the AT by means of the signaling messages defined in [S1].
- QoS filter specification – Information in the QoS filter specification is used to set up Traffic Flow Templates (TFTs) at the PDSN by means of signaling between the AT and the PDSN, as defined in [S2].

For more details on the flow specification and filter specifications, refer to [Q1].

Both the QoS flow specification and the filter specification are defined and set up via a configuration file. The configuration file can be edited using an external text editor and copied to the qos_configuration directory. The flow specification and filter specification are defined in the same file. Multiple configuration files can be created, and you can formulate a QoS request by choosing the desired configuration file.

QMI QTA can be used to trigger QoS requests in the following variations:

- Unidirectional QoS request
- Bidirectional QoS request
- Bundled unidirectional QoS requests, i.e., multiple QoS requests, that will be sent to the RAN in a single air interface message
- Bundled bidirectional QoS requests

For details on how to create a configuration file and its structure, see Appendix A.

The following sections describe the QoS features and procedures to perform QoS operation using QMI QTA.

4.1 QoS setup

To set up QoS, you must go to the QoS Manager tab, as shown in Figure 4-1. Note that the QoS Sessions Table is located in the Other QoS Operations panel. The Status Bar is at the bottom of the screen.

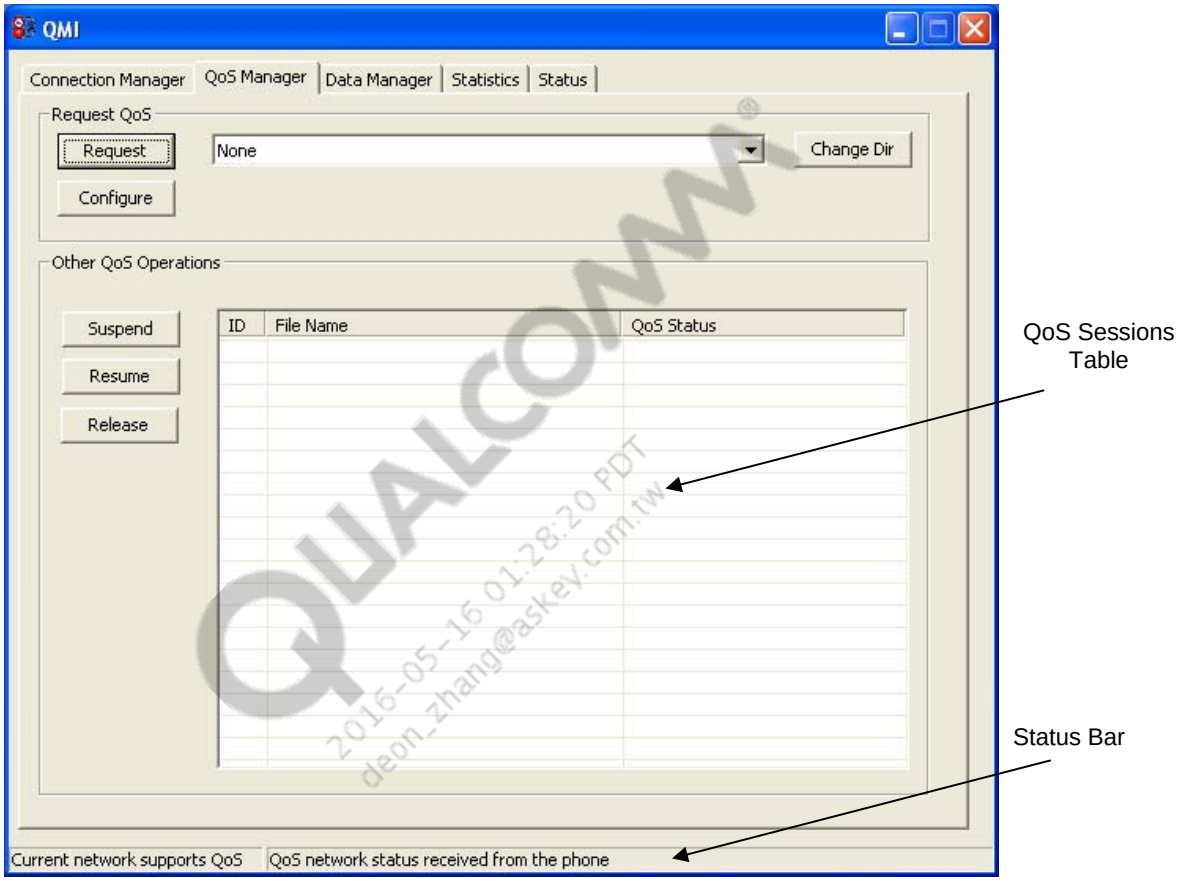

Figure 4-1 QoS Manager

4.1.1 QoS specifications (flow and filter)

You must select a QoS configuration file to start or configure a QoS session. Details of the structure of the QoS configuration file, and how to create and use the file, are provided in Appendix A. Select a QoS configuration file from the dropdown list, as shown in Figure 4-2.

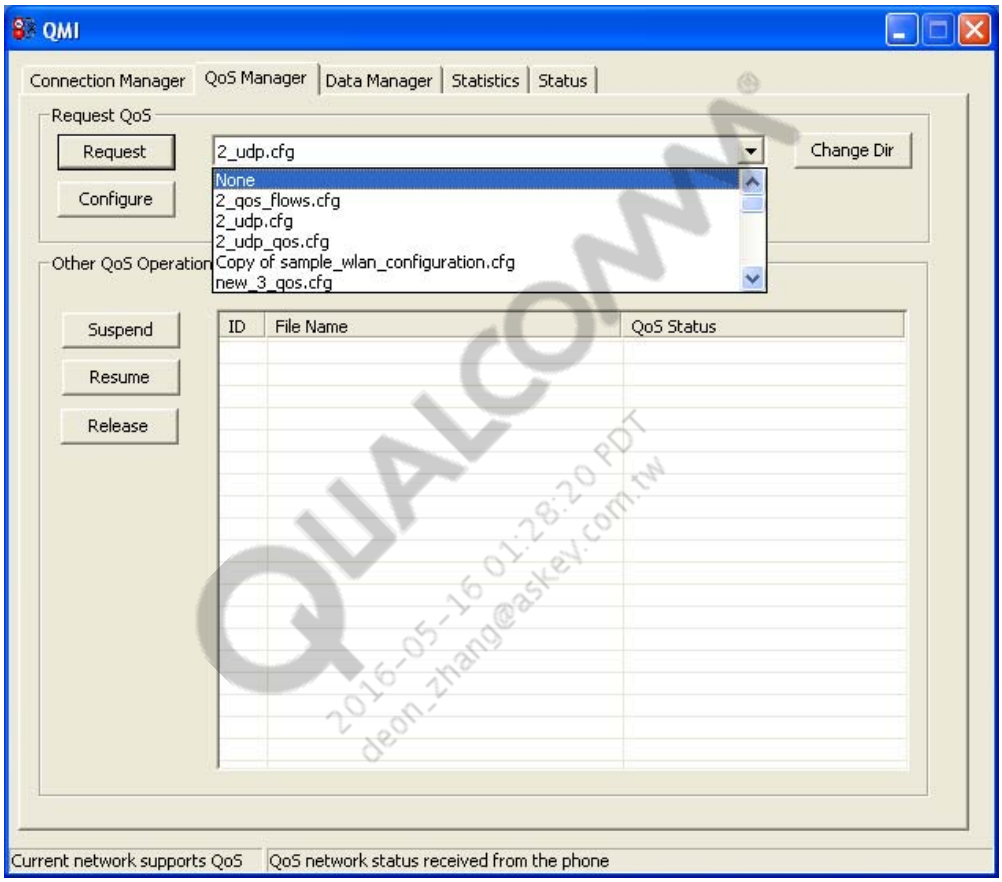

Figure 4-2 QoS configuration files

By default, the application looks for the QoS configuration file in the qos_configuration directory under QMI QTA's home directory. You can instruct QMI QTA to look for a QoS configuration file in another directory by clicking CHANGE DIR (see Figure 4-2). Clicking CHANGE DIR opens the directory selection window, as shown in Figure 4-3.

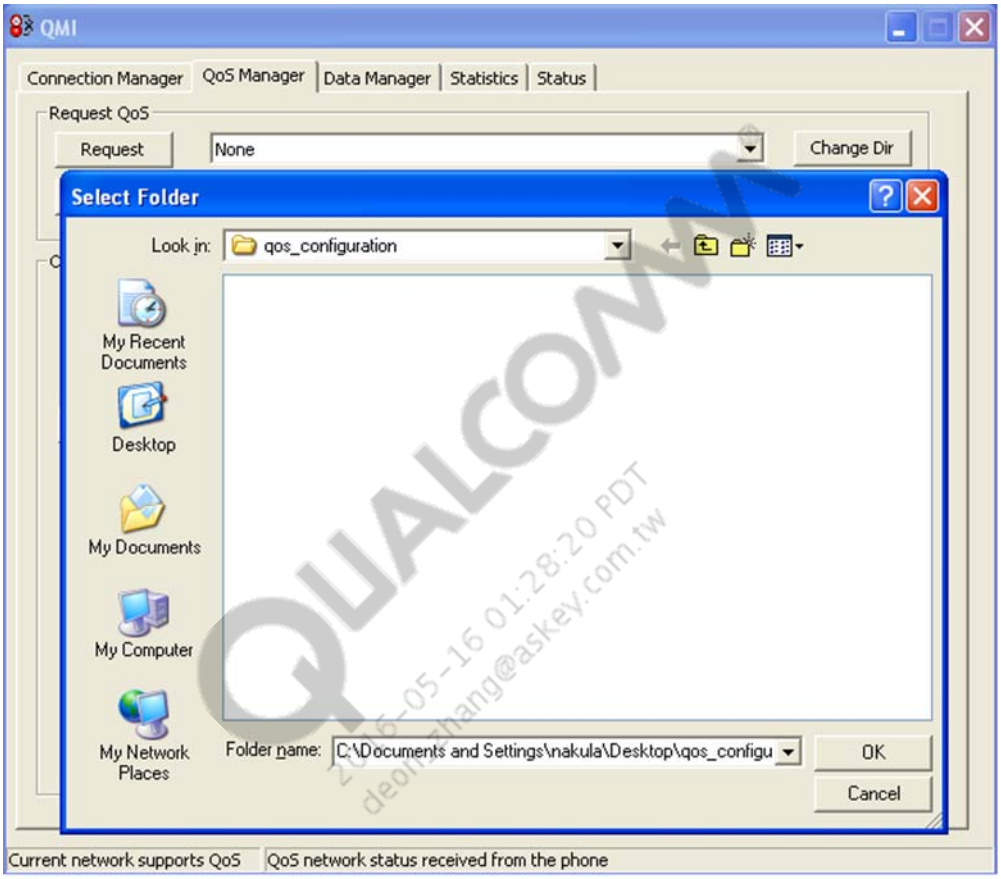

Figure 4-3 QoS configuration file directory selection

4.2 QoS functions

The QoS functions are as follows:

- Start QoS – Generates a ReservationKKQoSRequest from the specific configuration file
- Suspend – Generates a ReservationOffRequest for the specified QoS
- Resume – Generates a ReservationOnRequest for the specified QoS
- Release – Generates a ReservationKKQoSRequest with NULL for the specified QoS

The following sections describe the usage of these QoS functions.

4.2.1 Start QoS – ReservationKKQoSRequest

To start a QoS session, select a configuration file and click REQUEST. Selecting this option immediately triggers a ReservationKKQoSRequest. An error dialog box will appear if there is an error in the configuration file, flow parameters, or filter parameters. Otherwise, you will see the QoS sessions listed in the QoS Sessions Table (see Figure 4-1). In the status bar, a Sending QoS Request message appears and is followed by the message Request was successfully issued. Figure 4-4 illustrates the UI when the QoS is being established.

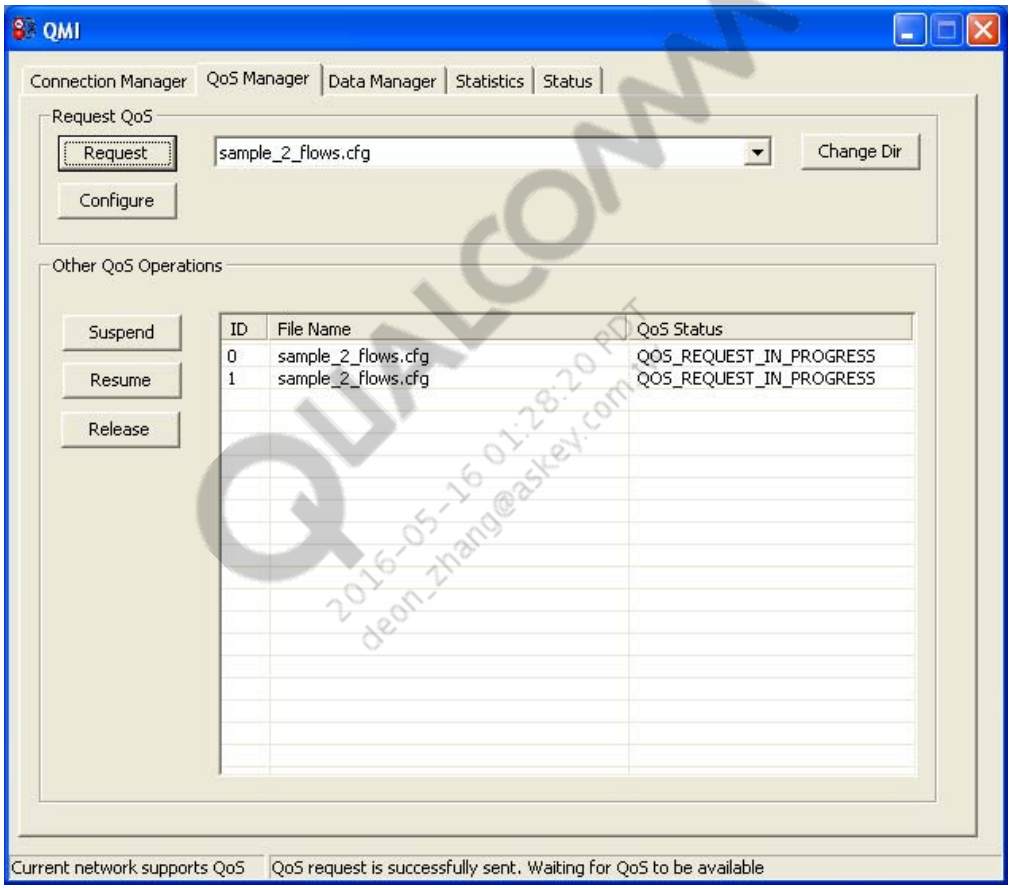

Figure 4-4 Request for QoS

Figure 4-5 illustrates the UI for a successful QoS request, i.e., QoS is granted by the RAN and the TFT setup at the PDSN is successful.

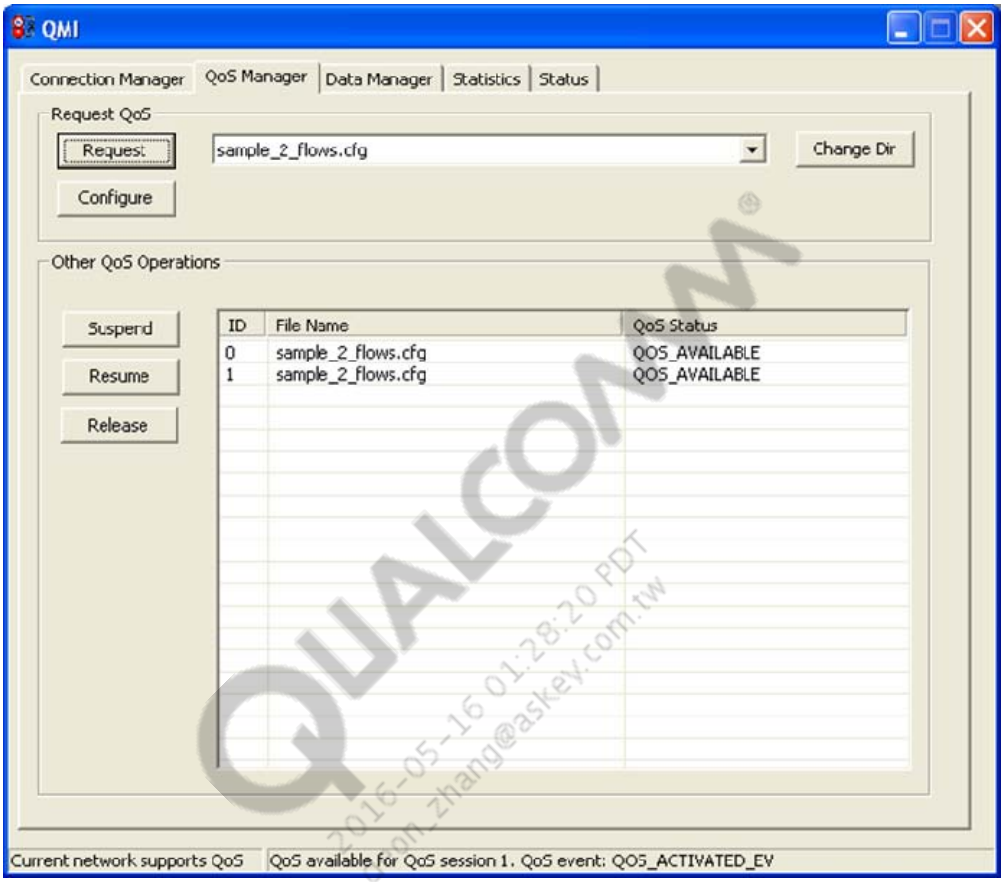

Figure 4-5 QoS available

When the QoS request is successful, a new QoS flow is added to the list of available QoS flows. The QoS Sessions Table lists all the QoS flows that have been started with their respective IDs, the name of the configuration file that has the QoS specification, and the QoS status.

4.2.1.1 Configure QoS

To configure a QoS session, select a configuration file and click CONFIGURE. Selecting this option immediately triggers a ReservationKKQoSRequest. Refer to [Q1] for details on QoS configuration.

4.2.2 Suspend – ReservationOffRequest

To suspend a QoS session or a group of QoS sessions, select the desired QoS session from the QoS Sessions Table, as shown in Figure 4-6. To select more than one QoS session, press CTRL while selecting the sessions. To suspend the QoS session(s), click SUSPEND.

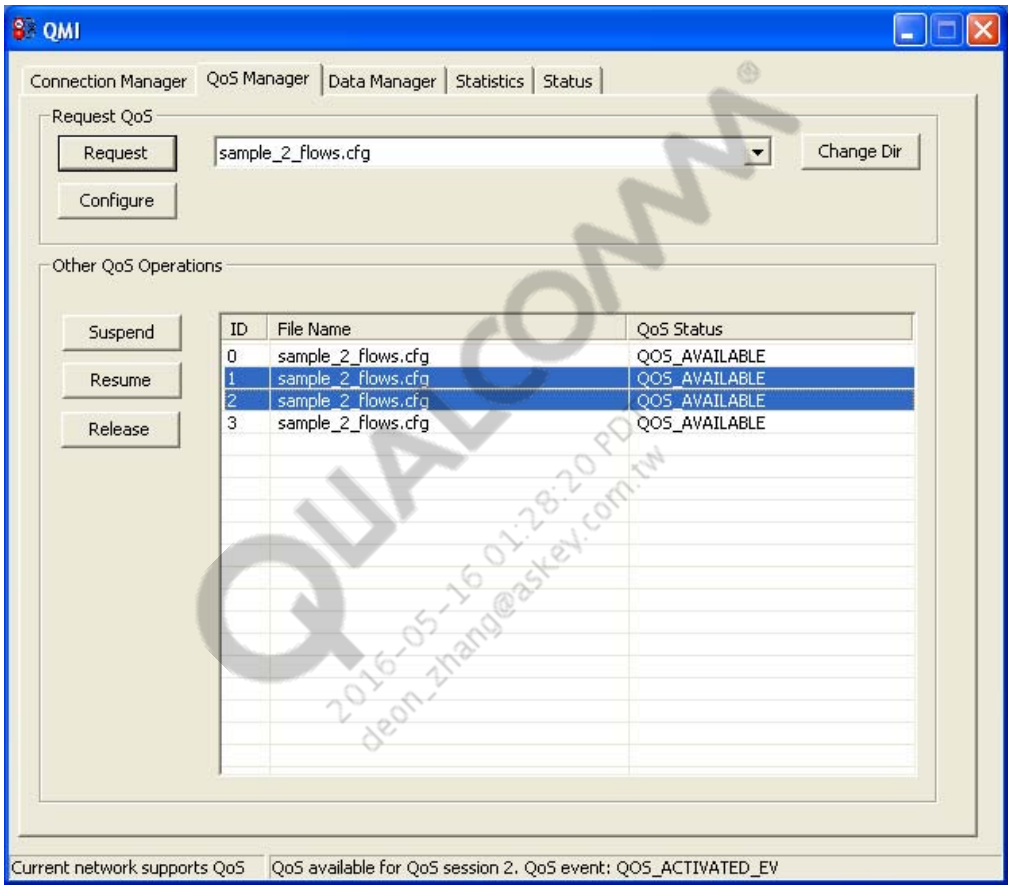

Figure 4-6 Suspend QoS session(s)

When the QoS suspend request is sent and is successful, the QoS Sessions Table will list the status of the suspended QoS as QOS_DEACTIVATED, as shown in Figure 4-7.

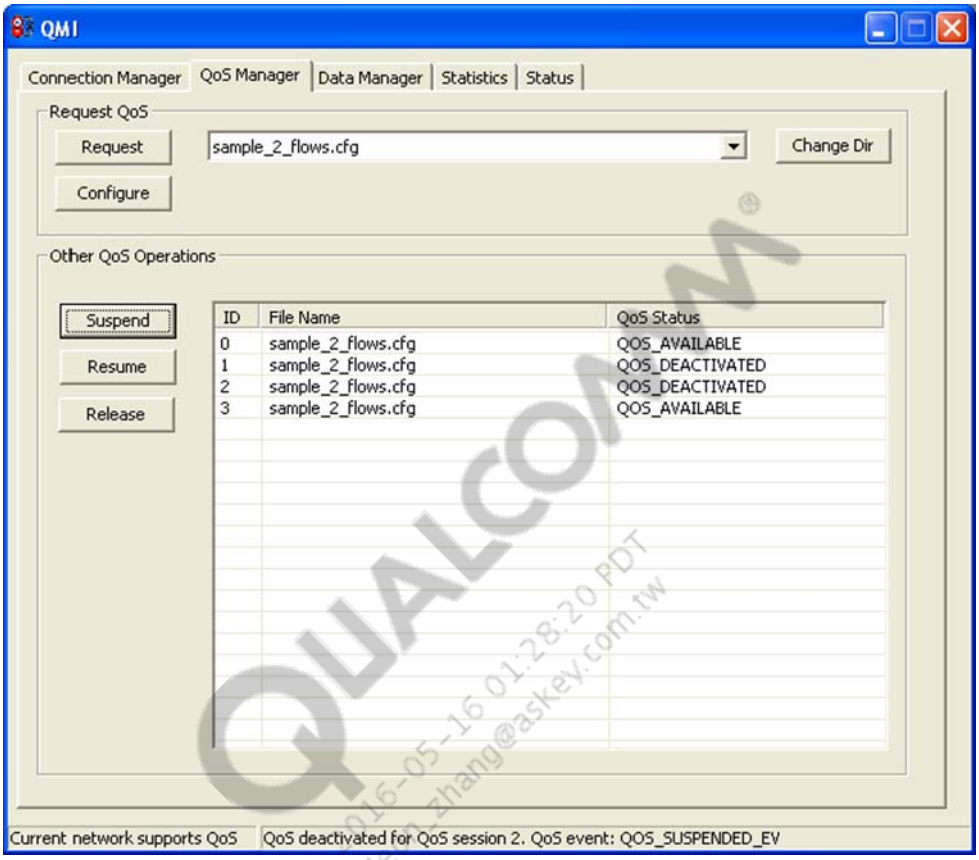

Figure 4-7 QoS suspended (deactivated)

4.2.3 Resume – ReservationOnRequest

You can trigger a ReservationOnRequest for a specific QoS session by selecting the QoS session from the QoS Sessions Table and then clicking RESUME. When the resume request is successful, the status of QoS sessions becomes QOS_AVAILABLE again.

4.2.4 Release – ReservationKKQoSRequest with NULL

You can release a specific QoS session by selecting a QoS session from the QoS Sessions Table and then clicking RELEASE. This triggers the AT to send a ReservationKKQoSRequest with NULL, which will initiate a QoS release sequence of events. You can release QoS as a bundle, i.e., trigger a release of all QoS sessions, by selecting all the QoS sessions. When the release request is successful, the released QoS session no longer appears on the QoS Sessions Table.

5 Data Transfer

QMI QTA communicates with the Data Server application on the data server PC to set up data streams. The following steps are performed to configure data streams:

1. Configure the data transfer options via the Data Manager tab.
2. Select a data configuration file from the dropdown list on the Data Manager tab.
3. Initiate the data transfer on the selected configuration file. This will trigger the application to connect to the data server. The connection will set up the data server and QMI QTA for all the data streams using the options in step 1 and 2, followed by data transfer between QMI QTA and the data server.

The following sections describe the data transfer options and operation using QMI QTA.

5.1 Set up data transfer options

The first step for starting a data transfer is to set up the data transfer options. QMI QTA has two sets of options:

- Options set via the Data Manager tab – These are the general options that can apply to all the data transfers performed through QMI QTA. These options are divided into three sections:
  - Basic options
  - Transfer options
  - Other options
- Data configuration file options – These options are specific to a data transfer.

Figure 5-1 illustrates the options available on the Data Manager tab.

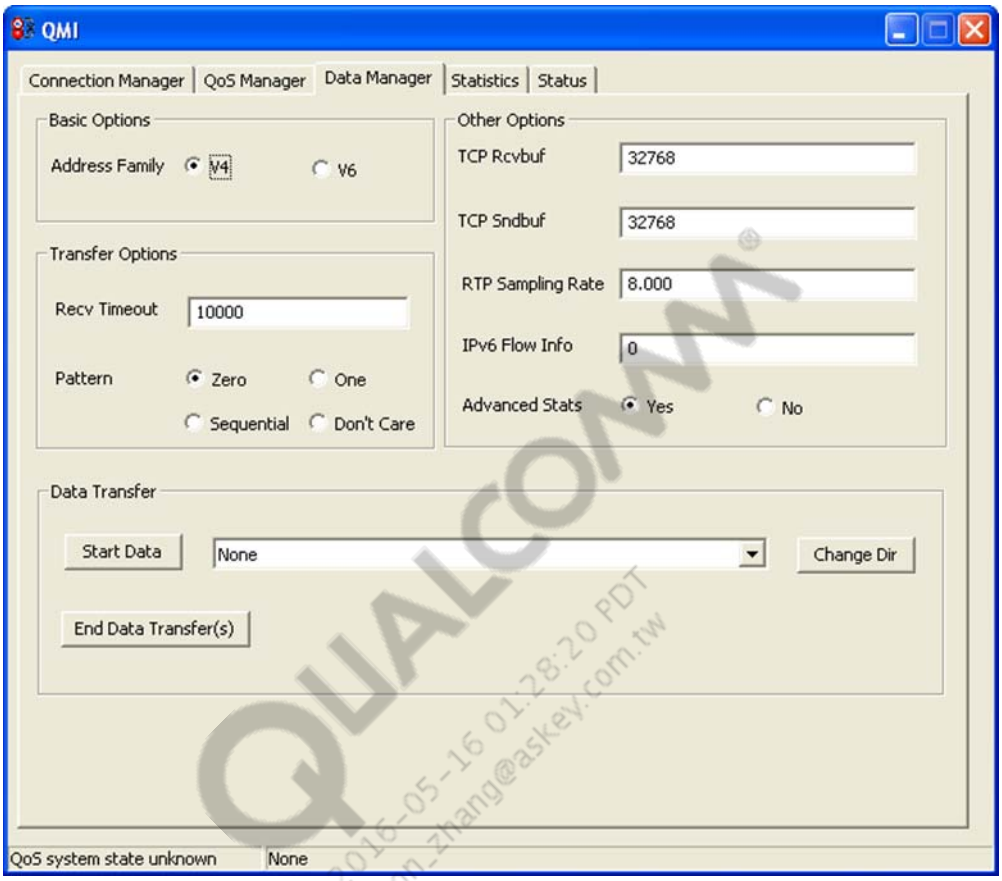

Figure 5-1 Data Manager

5.1.1 Basic option

The following basic option is available:

- Address Family – This option allows either an IPv4- or an IPv6-based data transfer.

5.1.2 Transfer options

The following transfer options are available:

- Recv Timeout – This allows you to specify how long the application should wait for data to arrive before aborting the transfer in the forward direction.
- Pattern – This allows you to specify the pattern of the data to be sent. The pattern can be any one of the following:
  - All 0s
  - All 1s
  - 1 followed by 0
  - Ignore pattern verification; send numbers from 0 to 9

5.1.3 Other options

The following socket options are:

- TCP Rcvbuf – This allows you to specify the size of the receive buffer.
- TCP Sndbuf – This allows you to specify the size of the send buffer.
- RTP Sampling Rate – This allows you to set the sampling rate if the data transfer is type RTP.
- IPv6 Flow Info – This allows you to set the flow information field in the IPv6 header if the data transfer is type IPv6.
- Advanced Stats – This allows you to enable QMI QTA to generate advanced statistics, such as packets dropped on each link, jitter, etc.

5.1.4 Data configuration file options

Details of the structure of the data configuration file, and how to create and use the file, are described in Appendix A. The following options are available:

- DATA_SERVER_IPADDRESS – QMI QTA allows you to enter the IPv4 hostname (DNS name) or the IPv4 address of the server. The application must first establish the control connection (TCP) with the data server to initialize it before the application starts the data transfer.
- DATA_SERVER_IPV6ADDRESS – QMI QTA allows you to enter the IPv6 hostname (DNS name) or the IPv6 address of the server. In the case of IPv6 data transfer, the control connection to the data server still requires the IPv4 address. Hence, both the IPv4 and IPv6 host name need to be specified for IPv6-based data transfers.
- DATA_SERVER_PORT – You do not set the data server port. The port is retrieved from the data server when QTA initializes the data server. This option is here for future enhancements.
- TEST_APP_PORT – This is the port number associated with the socket that is opened by QMI QTA on the phone side. If you set the value to zero, then the application chooses a random value. If a nonzero value is specified, the application uses this value. The test application port should be in the range of 1024 to 9999.
- PROTOCOL – This allows you to specify the protocol to be used during the transfer. Possible values are TCP, UDP, or RTP.
- DIRECTION – This allows you to specify the direction of the data transfer. The direction can be any one of the following:
  - RX – Forward direction data transfer
  - TX – Reverse direction data transfer
  - BOTH – Bidirectional data transfer
- DSCP – This allows you to specify a ToS (DSCP) value. If it is left blank, the default value is chosen. When it is set, this field is used to mark the DSCP bit in the forward and reverse direction.
- RX_DURATION – This allows you to specify the total duration of the transfer for forward direction. The duration interval is in milliseconds. So, for a data transfer of 10 sec, you should set the value to 10000.

- RX_INTERVAL – This allows you to specify the time interval at which the data has to be received.
- RX_PACKET_SIZE – This allows you to specify the size of the data to be received at the given interval.
- TX_DURATION – This allows you to specify the total duration of the transfer for the reverse direction. The duration interval is in milliseconds. So, for a data transfer of 10 sec, you should set the value to 10000.
- TX_INTERVAL – This allows you to specify the time interval at which the data has to be sent for the given duration.
- TX_PACKET_SIZE – This allows you to specify the size of the data to be sent at the given interval for the duration of the data transfer.

5.2 Start data transfer

When you have set all the options on the Data Manager tab and have created a data configuration file, you must select a data configuration file to start a data transfer. The configuration file can contain parameters for multiple data transfers. Details of the structure of the data configuration file, and how to create and use the file, are described in Appendix A. Select a data configuration file from the dropdown list in the Data Transfer panel, as shown in Figure 5-2.

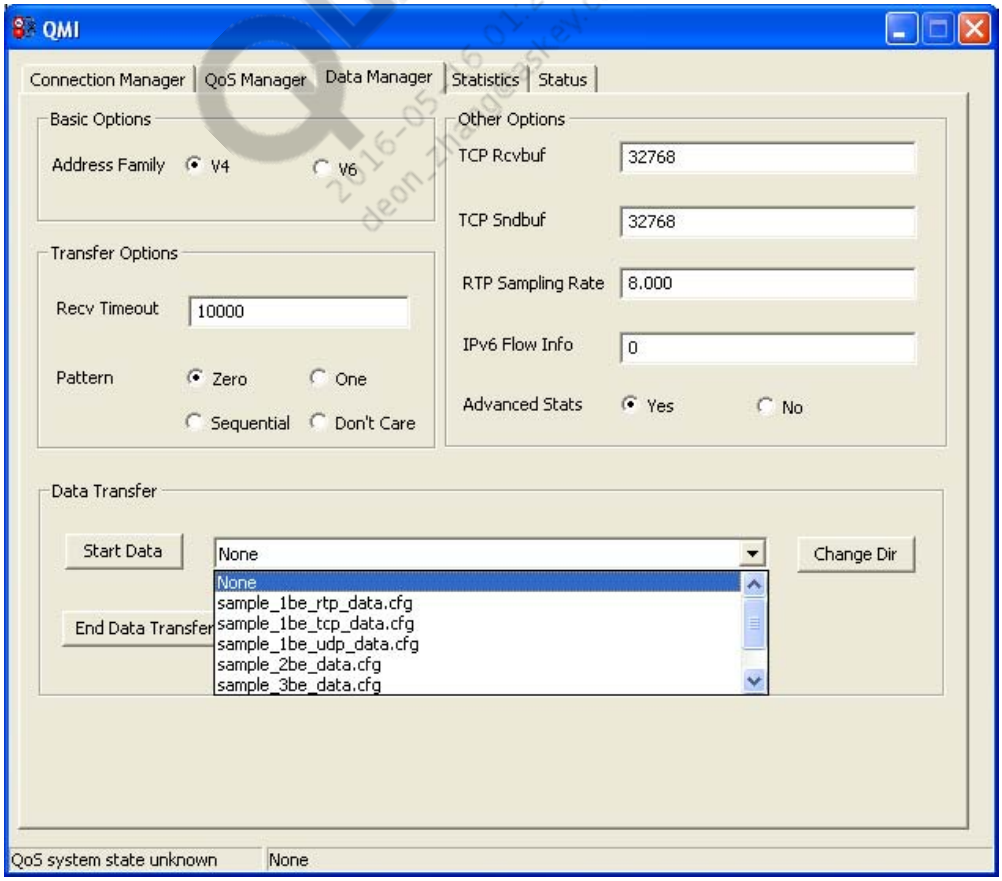

Figure 5-2 Select data configuration files

By default, the application looks for the data configuration file in the data_configuration directory under QMI QTA's home directory. This is where you would usually create your new configuration files. You can instruct QMI QTA to look for a data configuration file in another directory by clicking CHANGE DIR (see Figure 5-2). Clicking CHANGE DIR opens the directory selection window, as shown in Figure 5-3.

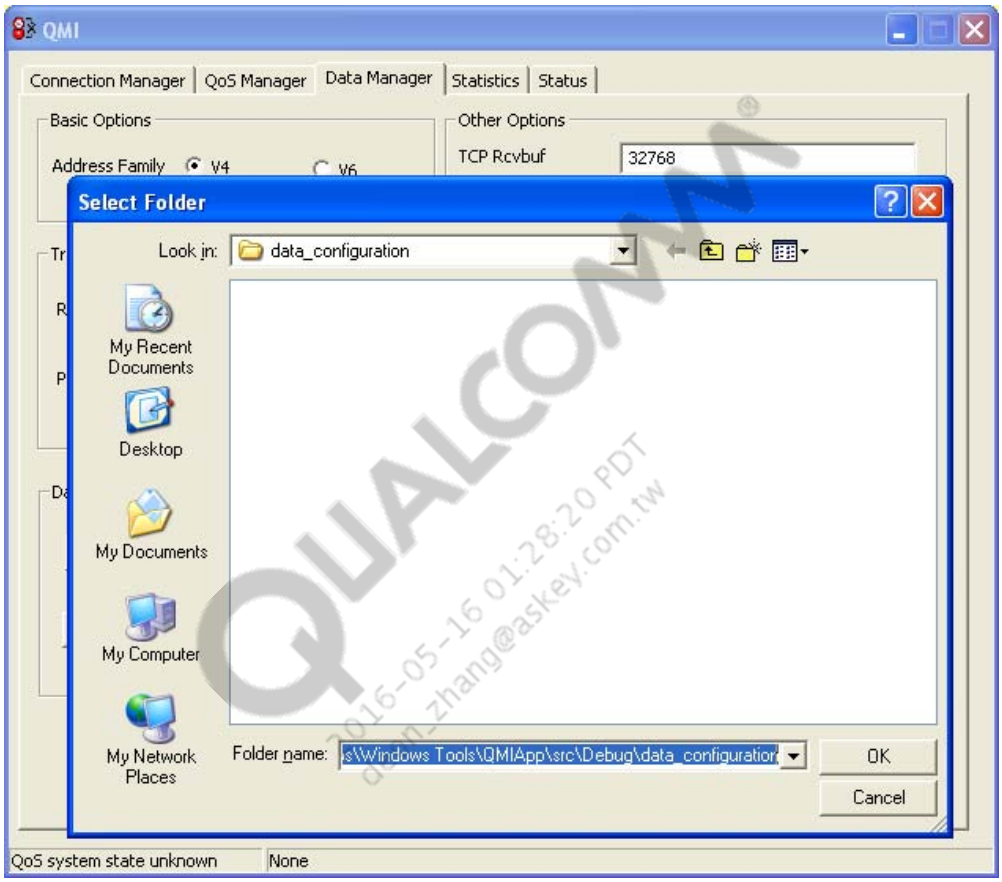

Figure 5-3 Data configuration file directory selection

After you select a data configuration file, click START DATA (see Figure 5-2) to start the data transfer. Clicking this button begins the data server initialization for all the data specifications that are specified in the data configuration file. The AT then establishes a PPP connection to the PDSN and the data transfer options are sent to the data server. When the options are received, the data server creates a new socket and sends the port number for the newly created socket back to QMI QTA. At this time, the application displays a message, such as Data server listening on port number #XXXXX, in the status bar. When the port numbers for all the data streams are received, QMI QTA closes the control connection with the server, establishes the data connection on the returned port numbers, and starts the data transfer to/from the data server.

Figure 5-4, Figure 5-5, and Figure 5-6 illustrate the process of a successful data server initialization.

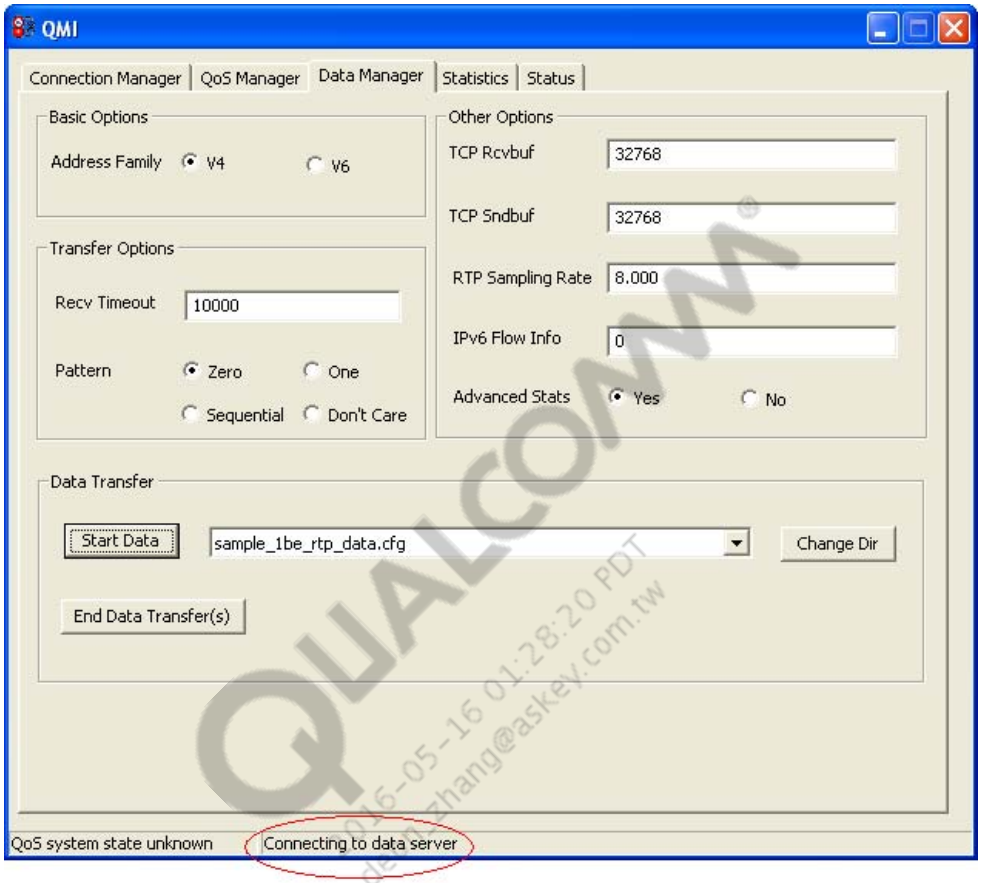

Figure 5-4 Connecting to data server

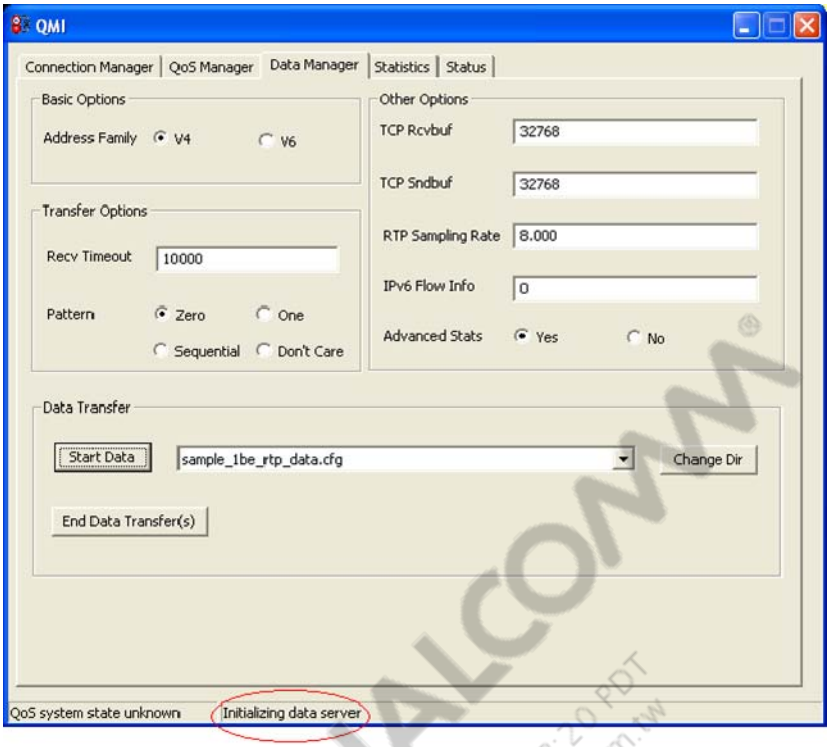

Figure 5-5 Initializing data server

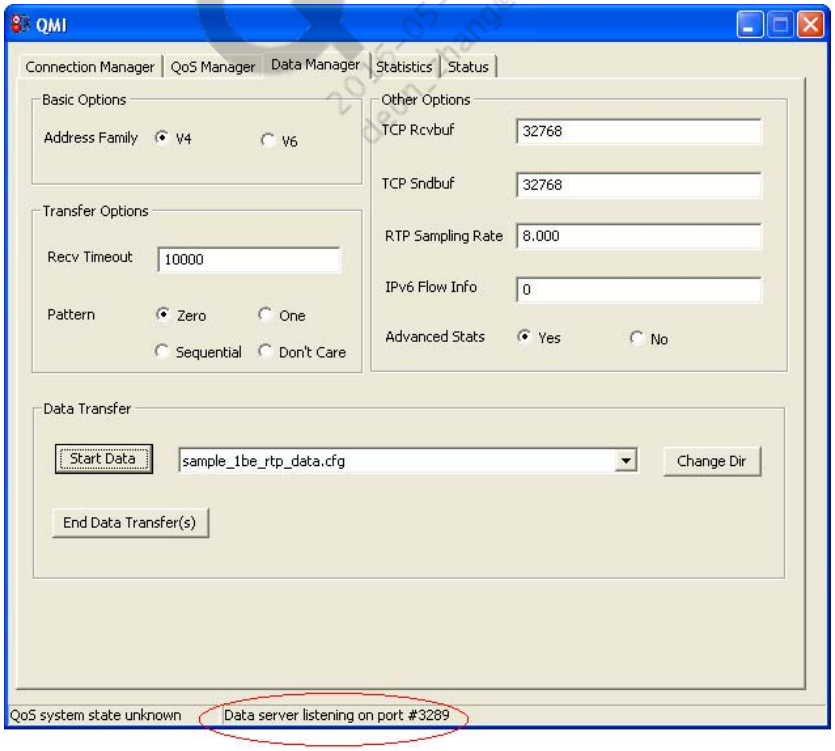

Figure 5-6 Data server listening on port

When the data server is initialized, QMI QTA starts the data transfer and initial information appears on the Statistics tab, as shown in Figure 5-7.

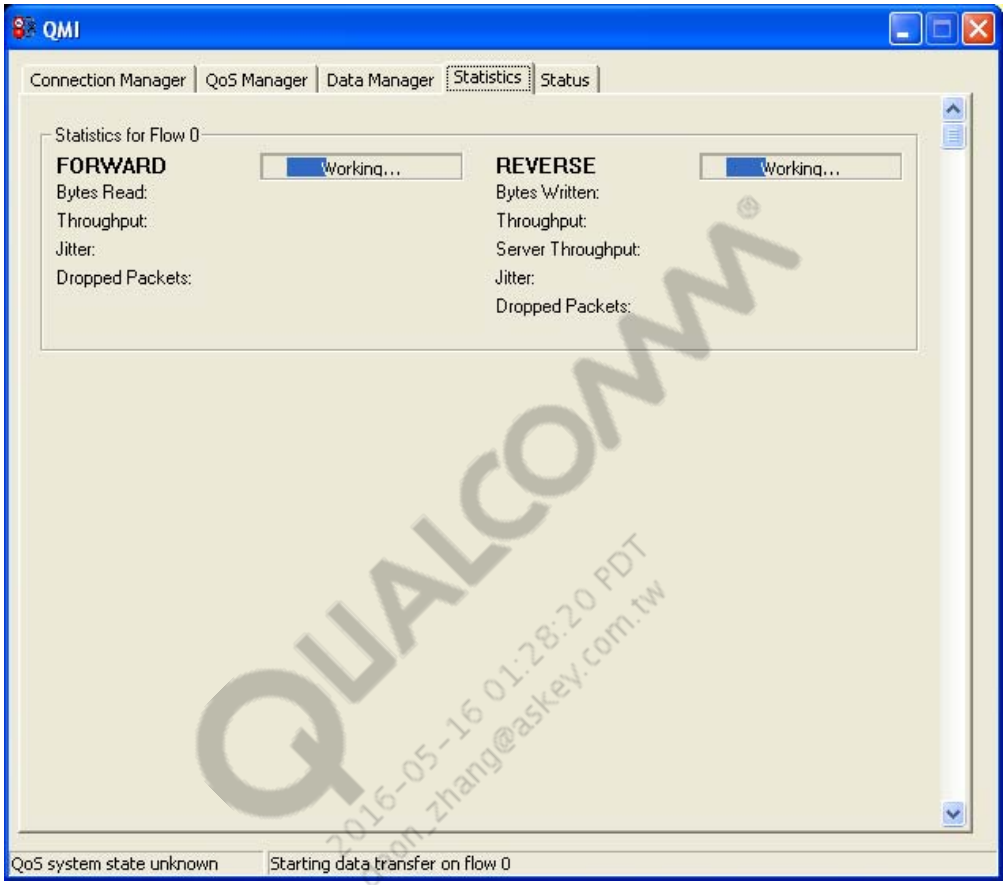

Figure 5-7 Statistics manager

The Statistics tab displays two progress bars, one for each direction of data transfer. When the data transfer is complete, the number of bytes transferred and the net throughput is displayed, as shown in Figure 5-8.

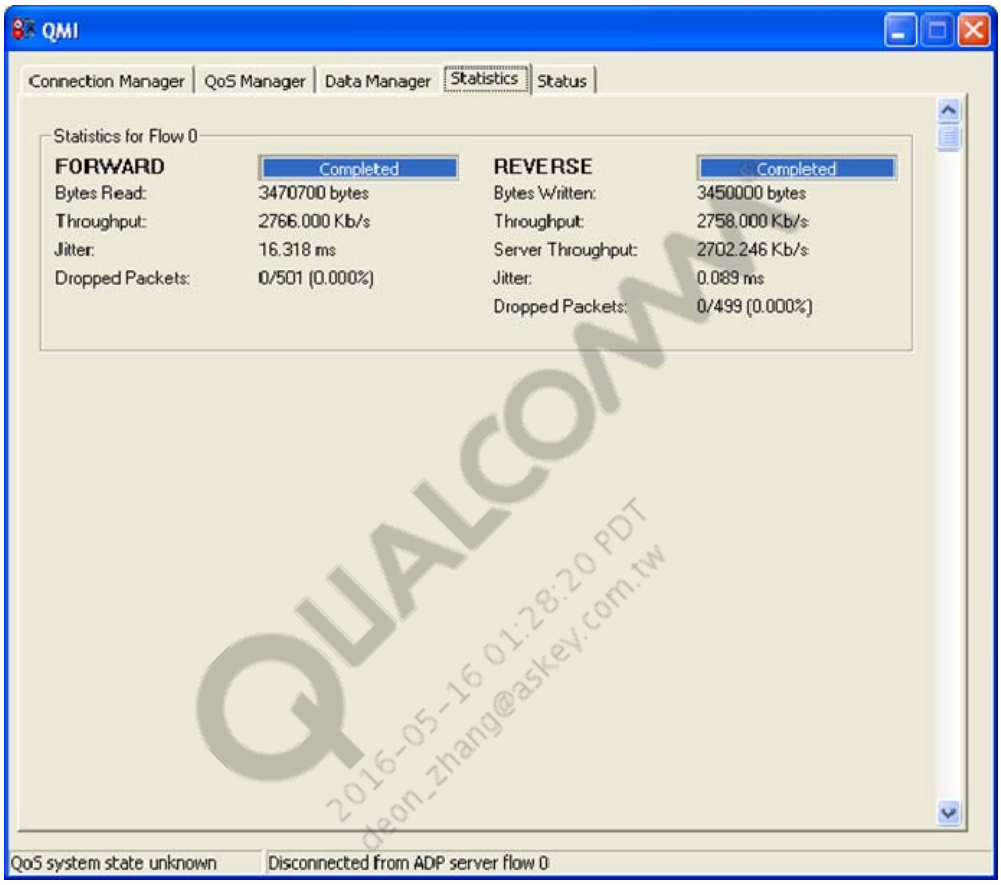

Figure 5-8 Statistics for the data transfer

5.3 Abort data transfer

You can abort all ongoing data transfers at any time during the data transfer. To abort the data transfer, click END DATA TRANSFER(S) on the Data Manager tab (see Figure 5-4). A confirmation dialog box appears, as shown in Figure 5-9. Click YES to abort the data transfer.

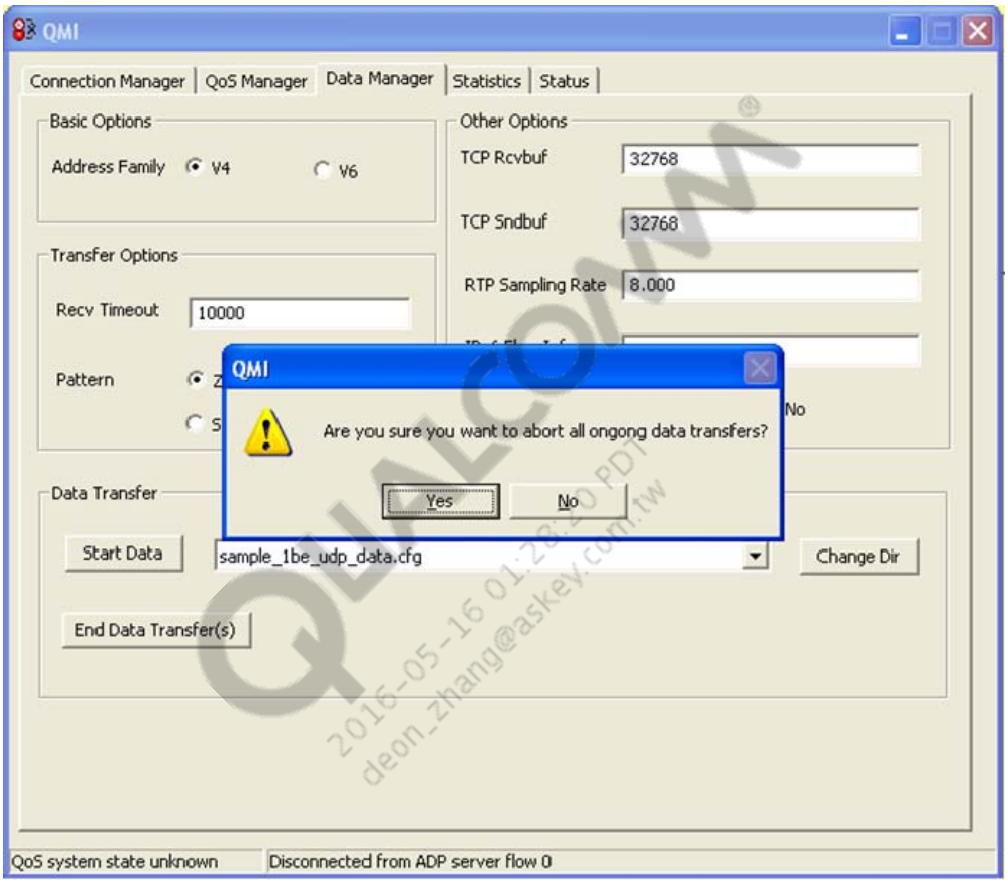

Figure 5-9 Abort data transfer

If a data transfer is still occurring, ensure that you abort the data transfer before you close QMI QTA. QMI QTA must tell the data server that the data transfer ended and the data server can clear the session on its side. Otherwise, the data server keeps the session open until the data server is closed.

6 Status Messages

QMI QTA lists status messages in the Events window of the Status tab, as shown in Figure 6-1. QMI QTA lists the status messages for all messages displayed in the status bar, all error messages, and other messages to help in understanding the call flow of QMI QTA.

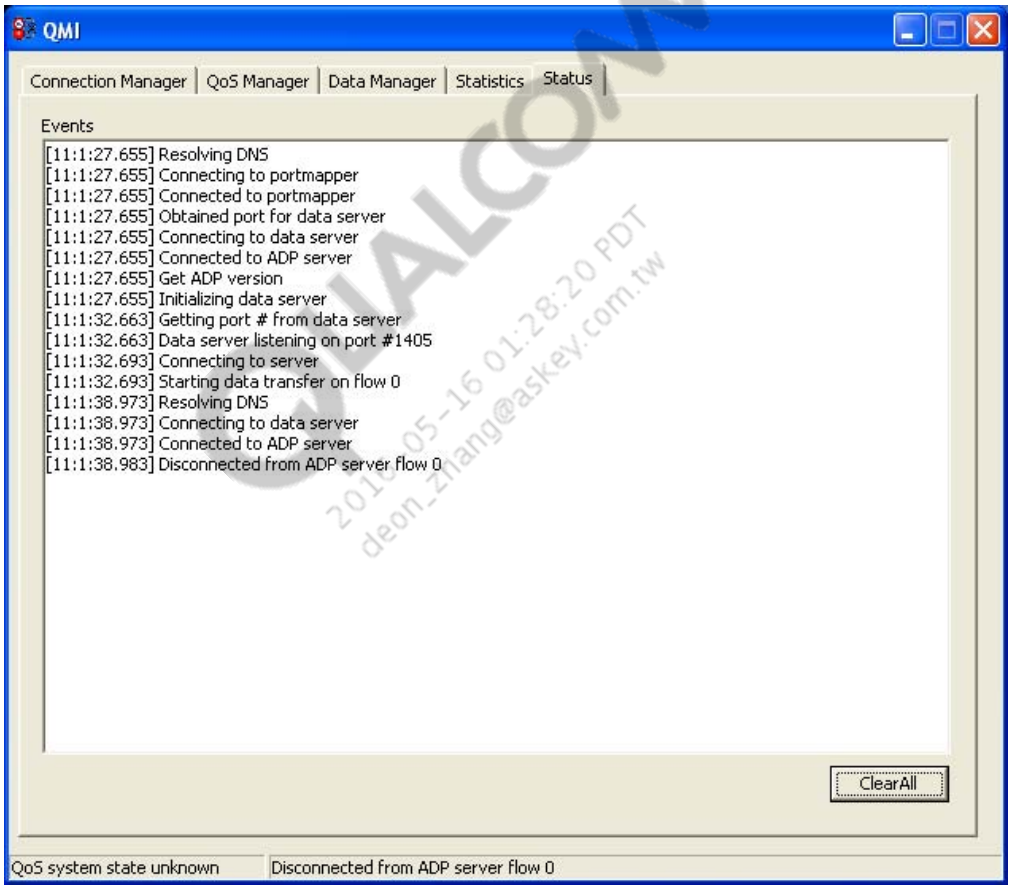

Figure 6-1 Status manager

Every time QMI QTA starts, it generates a new log file in the log directory under the home directory of QMI QTA. The log file contains all the messages that were listed. You can view the messages, even after the application is closed.

7 Application Limitations and Enhancements

7.1 Limitations

The following are known limitations of QMI QTA:

- Data configuration files can have a maximum of 20 data configuration specifications.
- In the QoS configuration file, the maximum number of QoS specifications supported is ten, i.e., maximum of seven Rx and seven Tx filters, and up to seven Rx flows/profile IDs and seven Tx flows/profile IDs.

7.2 Future enhancements

The following features will be added to future releases of QMI QTA:

- Support for latency tests
- Support for data transfer based on pattern files

A Configuration Files

A.1 QoS configuration file example

A QoS configuration file specifies the QoS specification block. A QoS specification consists of Rx and Tx filters and Rx and Tx flows. You can specify multiple numbers of filter and flow structures in a file. For details of a QoS specification, refer to document [Q1]. The QoS configuration file should be present in the qos_configuration directory under the home directory of QMI QTA.

A.1.1 Example 1

The requirements are as follows:

- Bidirectional QoS request
  - One forward link flow with two profiles (100, 101), where Profile 100 is the most-preferred profile and Profile 101 is the less-preferred profile
  - One reverse link flow with two profiles (100, 101), where Profile 100 is the most-preferred profile and Profile 101 is the less-preferred profile
- Filter specification for Rx flow is for TCP protocol (IPv4)
- Filter specification for Tx flow is for UDP protocol (IPv4)

For these QoS requirements, the configuration file should appear as follows:

```
QoSSpec{
RxFilter{
AEEQOSFILTEROPT_IP_VERSION:v4
AEEQOSFILTEROPT_IP_NEXT_PROTOCOL:TCP
AEEQOSFILTEROPT_IPV4_SRC_ADDR:
AEEQOSFILTEROPT_IPV4_DST_ADDR:
AEEQOSFILTEROPT_IPV4_TOS:
AEEQOSFILTEROPT_SRC_PORT:
AEEQOSFILTEROPT_SRC_PORT_RANGE:
AEEQOSFILTEROPT_DST_PORT:
AEEQOSFILTEROPT_DST_PORT_RANGE:
}
TxFilter{
AEEQOSFILTEROPT_IP_VERSION:v4
AEEQOSFILTEROPT_IP_NEXT_PROTOCOL:UDP
AEEQOSFILTEROPT_IPV4_SRC_ADDR:
AEEQOSFILTEROPT_IPV4_DST_ADDR:
AEEQOSFILTEROPT_IPV4_TOS:
```

```
AEEQOSFILTEROPT_SRC_PORT:
AEEQOSFILTEROPT_SRC_PORT_RANGE:
AEEQOSFILTEROPT_DST_PORT:
AEEQOSFILTEROPT_DST_PORT_RANGE:
}
Rx{
AEEQOSFLOWOPT_CDMA_PROFILE_ID:100
}
Rx{
AEEQOSFLOWOPT_CDMA_PROFILE_ID:101
}
Tx{
AEEQOSFLOWOPT_CDMA_PROFILE_ID:200
}
Tx{
AEEQOSFLOWOPT_CDMA_PROFILE_ID:201
}
}
```

NOTE

- Flow specifications should always follow a filter.
- All Tx filters should follow all Rx filters.
- All Tx flows should follow all Rx flows.
- The QoS configuration file must have at least one Rx or at least one Tx flow specification in the configuration file.
- If you do not specify a filter in the configuration file, a default filter is created for Rx and Tx based on the IP version, protocol, test application, and Data Server's port numbers. You can also change these default filters by setting different values in the Rx and Tx filter forms from the UI.

If there is an error in a flow or filter value, an error screen will appear with the details of what is missing in the configuration file.

You can specify multiple QoS specification blocks in the configuration file if QoS flows in each direction are to be bundled into one single QoS request.

A.1.2 Example 2

The following is an example of a configuration file required for a bundled configuration file with the following specifications:

- QoS Rx Flow 1 with Profiles 100 and 101
- QoS Tx Flow 1 with Profiles 200 and 201
- QoS Rx Flow 2 with Profiles 102 and 103
- QoS Tx Flow 2 with Profiles 202 and 203

Rx/Tx Flows 1 and 2 form two different bidirectional QoS requests to be sent to the AN in a single message.

NOTE In this example, filter specifications are required; for simplicity, they are not shown.

```
QoSSpec{
Rx{
AEEQOSFLOWOPT_CDMA_PROFILE_ID:100
}
Rx{
AEEQOSFLOWOPT_CDMA_PROFILE_ID:101
}
Tx{
AEEQOSFLOWOPT_CDMA_PROFILE_ID:200
}
Tx{
AEEQOSFLOWOPT_CDMA_PROFILE_ID:201
}
}
QoSSpec{
Rx{
AEEQOSFLOWOPT_CDMA_PROFILE_ID:102
}
Rx{
AEEQOSFLOWOPT_CDMA_PROFILE_ID:103
}
Tx{
AEEQOSFLOWOPT_CDMA_PROFILE_ID:202
}
Tx{
AEEQOSFLOWOPT_CDMA_PROFILE_ID:203
}
}
```

A.2 Data configuration file example

Details of the options available through the data configuration files are described in Section 5.1.4. The data configuration files should be present in the data_configuration directory, under the home directory of QMI QTA. This section provides examples of a data configuration file with the corresponding specifications.

A.2.1 Example 1

Data specification #1

- Direction – Forward and reverse
- Port number – 8000
- Protocol – UDP
- DSCP – (may be left blank)
- Tx data transfer specifications
  - Duration – 60 sec
  - Packet size – 100 bytes
  - Interval – 20 ms
- Rx data transfer specifications
  - Duration – 60 sec
  - Packet size – 240 bytes
  - Interval – 20 ms

Data specification #2

- Direction – Reverse
- Port number – 8000
- Protocol – TCP
- DSCP – (may be left blank)
- Tx data transfer specifications
  - Duration – 120 sec
  - Packet size – 240 bytes
  - Interval – 20 ms

```
DataSpec{
 DATA_SERVER_IPADDRESS: 10.46.88.196
 DATA_SERVER_PORT:
 TEST_APP_PORT:8000
 PROTOCOL:UDP
```

```
 DIRECTION:BOTH
 DSCP:
 RX_DURATION:60000
 RX_INTERVAL:20
 RX_PACKET_SIZE:100
 TX_DURATION:60000
 TX_INTERVAL:20
 TX_PACKET_SIZE:240
}
DataSpec{
 DATA_SERVER_IPADDRESS: 10.46.88.196
 DATA_SERVER_PORT:
 TEST_APP_PORT:8000
 PROTOCOL:TCP
 DIRECTION:TX
 DSCP:
 RX_DURATION:
 RX_INTERVAL:
 RX_PACKET_SIZE:
 TX_DURATION:120000
 TX_INTERVAL:20
 TX_PACKET_SIZE:240
}
```


A.2.2 Example 2

The following example shows the data specifications of an IPv6-based data transfer:

- Data Server IP Address – 10.43.180.93
- Data Server IPv6 Address – fe80::211:85ff:fe7a:57e4
- Test App Port number – 8000
- Protocol – UDP
- Direction – Reverse
- Tx data transfer specifications from Client
  - Duration – 120 sec
  - Packet size – 240 bytes
  - Interval – 20 ms

```
DataSpec{
 DATA_SERVER_IPADDRESS:10.43.180.93
 DATA_SERVER_IPV6ADDRESS: fe80::211:85ff:fe7a:57e4
 TEST_APP_PORT:8000
```

```
 PROTOCOL:UDP
 DIRECTION:TX
 TX_DURATION:10000
 TX_INTERVAL:20
 TX_PACKET_SIZE:100
}
```