Quick Start

This tutorial is a quick introduction to AppTransaction Xpert, which provides powerful visualization and diagnosis capabilities that aid in application analysis.

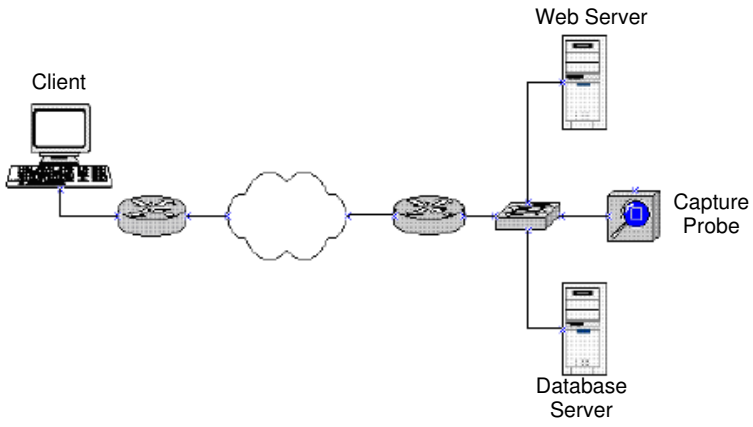
AppTransaction Xpert can be used to perform the following tasks:

- *Diagnose a problem application.* Use AppTransaction Xpert is to find the source of delays for an application. AppTransaction Xpert provides specific information about the root cause of application problems.
- *Predict application behavior.* Use AppTransaction Xpert to answer questions such as, “How will an application behave when deployed to remote users?” or “How much bandwidth is needed to support 500 users at each remote facility?”

This example shows AppTransaction Xpert as a diagnosis tool. The specific application being studied is a three-tier, Web-based application that has poor performance for remote users. These users, who claim that even simple transactions take 4 to 10 seconds, are requesting bandwidth upgrades.

The following figure shows the network topology. A client connects through the Internet to a Web server; the Web server communicates with a back-end database server.

Figure 1-1 Example Topology



Examine the Application

We captured a packet trace of a user performing a typical transaction and opened it in AppTransaction Xpert.

You will now open the transaction analyzer model and examine the application.

Procedure 1-1 Opening the Transaction Analyzer Model

1. Choose **File** > **Open Model** > **Transaction Analyzer…**

2. Select **Transaction Analyzer** from the Files of Type menu (Windows) or the Filters menu (Linux). Then navigate to the following directory, where <*reldir*> is the AppTransaction Xpert release directory:

   ```
   <reldir>\sys\examples\AppTransaction Xpert\examples\
   ```

   Note—If you don't see the Files of Type menu, click the "Switch to general file chooser" button in the lower-left corner of the dialog box.

3. Choose **Quickstart_before_fix** and click **Open**.

   If you do not see the file, add the **<*reldir*>\sys\examples\AppTransaction Xpert\examples** directory to your model directories, as follows:

   3.1 Choose **File > Manage Model Files > Add Model Directory**.

   3.2 In the **Browse for Folder** dialog box, navigate to **<*reldir*>\sys\examples\AppTransaction Xpert\examples**, then click **OK**.

   3.3 In the **Confirm Model Directory** dialog box, enable the **Include all subdirectories** checkbox and click **OK**.

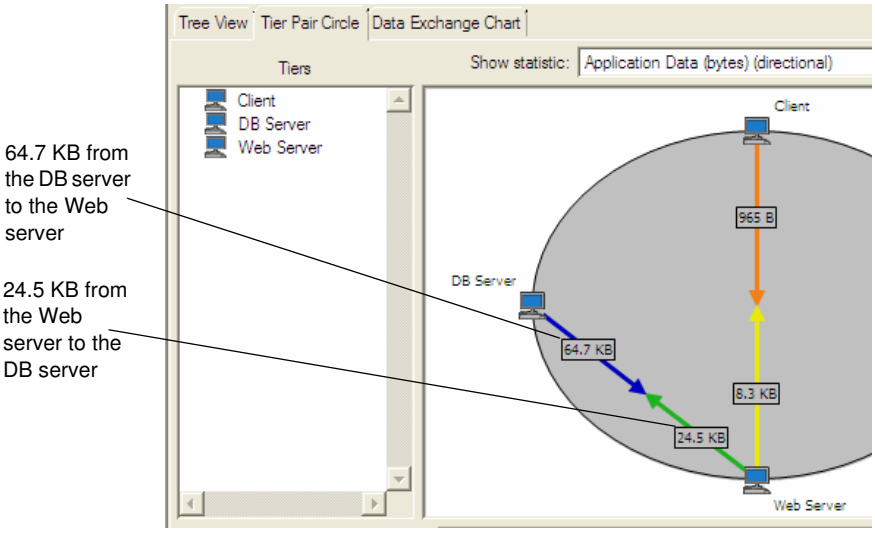
   After adding the directory to your model directories, navigate to the **examples** directory, choose **Quickstart_before_fix** and click **Open**.

4. If the **Getting Started Analyzing an Application Transaction** window appears, click **Close**.

   ➥ The Transaction Analyzer model appears in the Data Exchange Chart.

5. Click the **Tier Pair Circle** tab.

Figure 1-2 Tier Pair Circle



The Tier Pair Circle shows the flow of data between the different tiers for this transaction. Each arrow (in a one-way conversation) or arrow pair (in a two-way conversation) represents a conversation between tiers.

End of Procedure 1-1

For this transaction, the client communicates with the Web server, which communicates with the database (DB) server. The color of the arrows indicates the amount of traffic being sent. The Tier Pair Circle can be used to visualize other statistics, such as application turns, number of packets, network latency, and retransmissions.

Now that you know how the tiers are communicating, you want to see the communication over the course of the transaction, using the Data Exchange Chart.

To display the Data Exchange Chart, perform the following procedure.

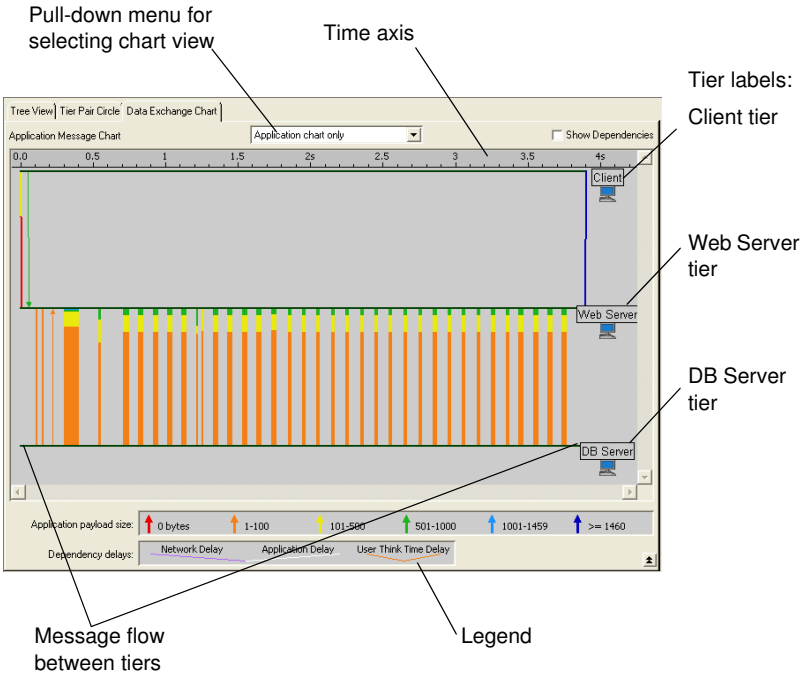
Procedure 1-2 Displaying the Data Exchange Chart

1 Click the **Data Exchange Chart** tab.

➥ The Data Exchange Chart appears.

2 Select **Application chart only** from the pull-down menu in the middle of the dialog box (if it is not already selected).

Figure 1-3 Data Exchange Chart



End of Procedure 1-2

The Data Exchange Chart shows the data transferred between tiers on a time line. As with the Tier Pair Circle, the colors of the application messages indicate the size of the messages. Each group's color represents a histogram of message sizes.

This transaction takes about 4 seconds to complete and consists primarily of orange and yellow messages between the Web server and DB server. Orange messages have between 1 and 100 Bytes. Yellow messages have between 101 and 500 Bytes.

The Data Exchange Chart shows the following:

- The majority of activity is between the two servers
- Most of the traffic consists of small messages
- Very little of the application response time is caused by the Wide Area Network (WAN)

This analysis seems to contradict the users' beliefs that additional WAN bandwidth would improve application performance.

Diagnose the Application

AppDoctor automates the process of application troubleshooting. AppDoctor finds the major components of delay and determines the root cause of application performance problems.

AppDoctor's Summary of Delays divides the total application response time into the following categories:

- *Tier Processing* delay is the total time it takes to process the application at each tier, including user think time.
- *Latency* delay is the component of delay due to latency in the network. (Latency is the time required for 1 bit to be transmitted across the network. Using a ping is one way to measure latency.)
- *Bandwidth* delay is the component of delay caused by the limited bandwidth of the network.
- *Protocol* delay is a metric of network restriction to packet flow. This restriction may be caused by flow control mechanisms imposed by network protocols. TCP, for example, has several built-in flow control mechanisms.
- *Congestion* delay is a measure of packet queuing in the network.

To display the AppDoctor Summary of Delays, perform the following procedure.

Procedure 1-3 Display the AppDoctor Summary of Delays

1. Choose **AppDoctor > Summary of Delays (AppDoctor Analysis).**

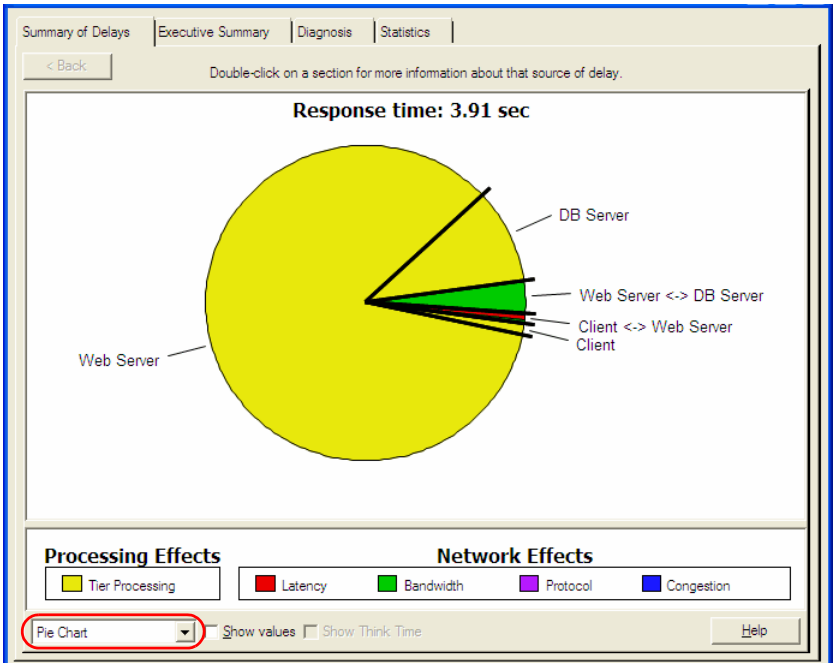
   Alternately, click the **AppDoctor** toolbar button.

2. Select **Pie Chart** from the pull-down menu at the bottom of the window, if it is not already selected.

   2.1 The Summary of Delays pie chart appears, as shown in the following figure.

Figure 1-4 Summary of Delays from AppDoctor



Notice that most of the delay is due to application processing by the Web server and DB server. Very little is related to the network, indicating, once again, that network upgrades will provide little improvement.

3 Place the mouse pointer over the Web server section to display a tooltip.

Notice the Web server is responsible for about 85 percent of the 4-second response time.

End of Procedure 1-3

Diagnosis provides a detailed analysis of probable bottlenecks for the application. Bottleneck categories exist for both network and application issues and are identified by machine or by network segment.

To display the AppDoctor diagnosis, perform the following procedure.

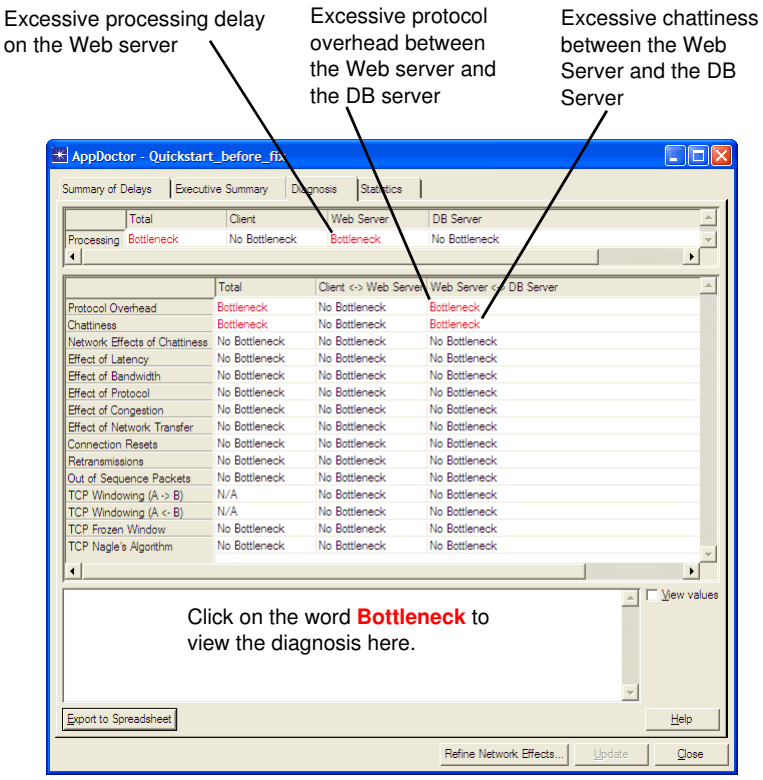
Procedure 1-4 Display the AppDoctor Diagnosis

1 In the AppDoctor window, click the **Diagnosis** tab.

2 Examine the three bottlenecks reported:

- Excessive processing delay on the Web server
- Excessive protocol overhead between the Web server and the DB server
- Excessive chattiness between the Web server and the DB server

Figure 1-5 Diagnosis from AppDoctor



The chattiness and protocol overhead bottlenecks indicate that the application is exchanging data inefficiently.

3 Click on each occurrence of the word Bottleneck (in red) to display suggestions for correcting the issue, in the lower pane of the **Diagnosis** dialog box.

4 Click **Close** to close the AppDoctor window.

End of Procedure 1-4

Fix the Application

Because the analysis shows that the application may be to blame for the poor performance, we consulted with the application developers. We presented the zoomed Data Exchange Chart, showing that the communication between the Web server and the DB server is composed of many small messages.

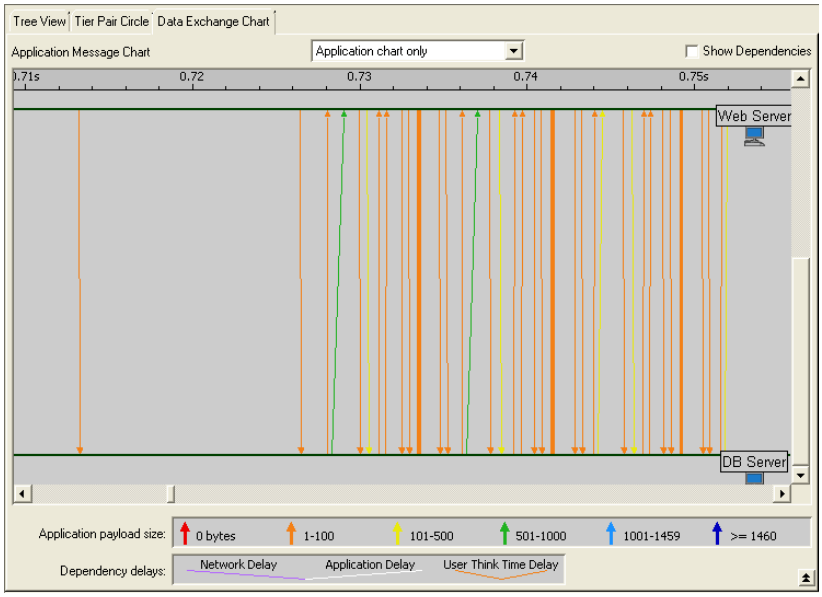
To display the zoomed Data Exchange Chart, perform the following procedure.

Procedure 1-5 Displaying the Zoomed Data Exchange Chart

1 Click the **Zoom** button in the toolbar, then drag the cursor to create a box around the area you want to view.

If you are not happy with your initial zoom, click the **Unzoom** button and try again. After adjusting the zoom level, use the arrow keys to scroll in all directions.

Figure 1-6 Zoomed Data Exchange Chart



End of Procedure 1-5

The developers' investigation revealed that the Web server and DB server were exchanging database fields one at a time. Further investigation showed that a small change to the application code would allow the data to be exchanged in larger segments.

The change had a dramatic impact on application performance: response time was reduced from 4 seconds to 1.1 seconds.

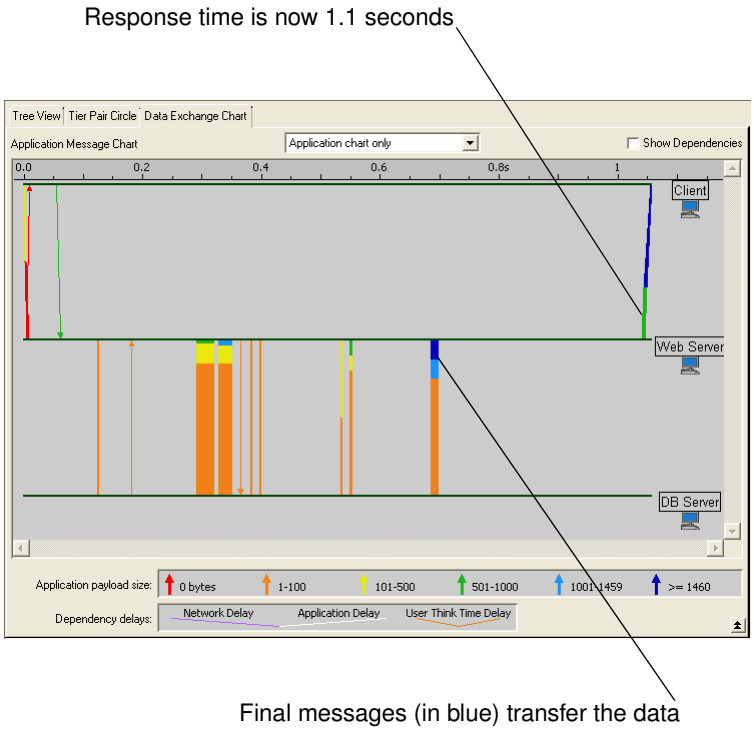
To view the Data Exchange Chart for the fixed application, perform the following procedure.

Procedure 1-6 Viewing the Fixed Application in the Data Exchange Chart

1. Choose **File > Open Model** > **Transaction Analyzer…**
2. Select **Transaction Analyzer** from the Files of Type menu (Windows) or Filters menu (Linux), if it is not already selected.
3. Select **Quickstart_after_fix**, then click **Open**.

   If the **Getting Started Analyzing an Application Transaction** window appears, click **Close**.

Figure 1-7 Data Exchange Chart for the Fixed Application



End of Procedure 1-6

Conclusion

AppTransaction Xpert helps you to visualize and diagnose application performance problems. In this example, users blamed the WAN for performance problems, but the true problem was a chatty application that caused significant delay at the Web server. Because the problem was in the application, a network upgrade would not have fixed the problem.

Choose **File > Close** and close both projects without saving the changes.