Predicting Application Response Times

This tutorial shows how to use QuickPredict and QuickRecode to do predictive studies that justify changes to network infrastructure or application behavior.

QuickPredict is an analytic simulation mechanism that enables you to test the performance of an application quickly under different network conditions. With QuickPredict, you can test proposed network upgrades to evaluate the possible impact on application performance.

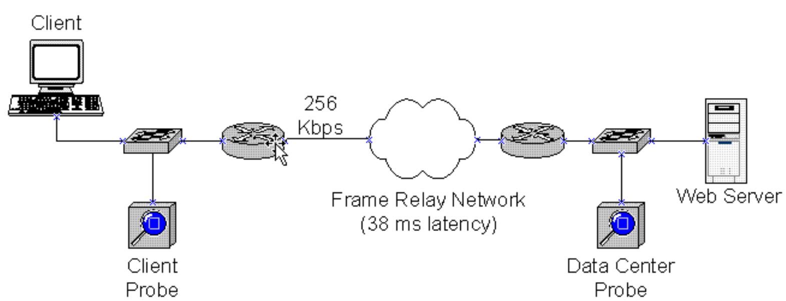
You can use QuickRecode to study the effect that changes to the flow of an application have on performance.

The first part of this tutorial studies the effects of network upgrades on an HTTP transaction. The goal is to predict the application response time under different network conditions.

The original packet trace—a typical web page transaction—was captured in a production environment. Most remote offices are connected through a 256 Kbps circuit.

Packet traces were collected from the client and web servers in the WAN shown in the following figure. When opened in AppTransaction Xpert, the packet traces were merged and synchronized automatically for the best possible analysis of delays at each tier and the network.

Figure 3-1   LAN Network Diagram



The subject web page takes 24 seconds to download. AppTransaction Xpert diagnosis of this application shows that the major bottleneck is limited bandwidth and congestion caused by the 256 Kbps Frame-Relay link.

Analyze the HTTP Application

This section shows the current performance characteristics of the HTTP application; the next section shows how to correct the problems using QuickPredict.

Procedure 3-1 Analyzing the HTTP Application

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

➥ The **Open** dialog box appears.

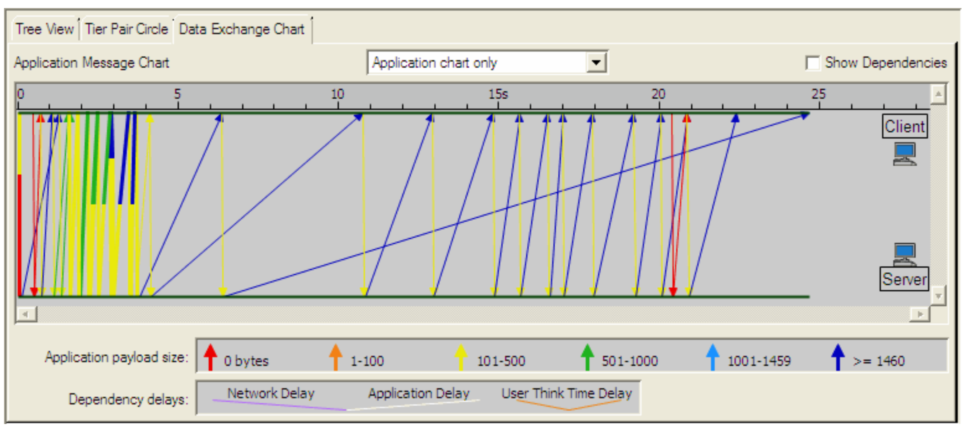
2 Navigate to the **examples** folder using the following path:

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

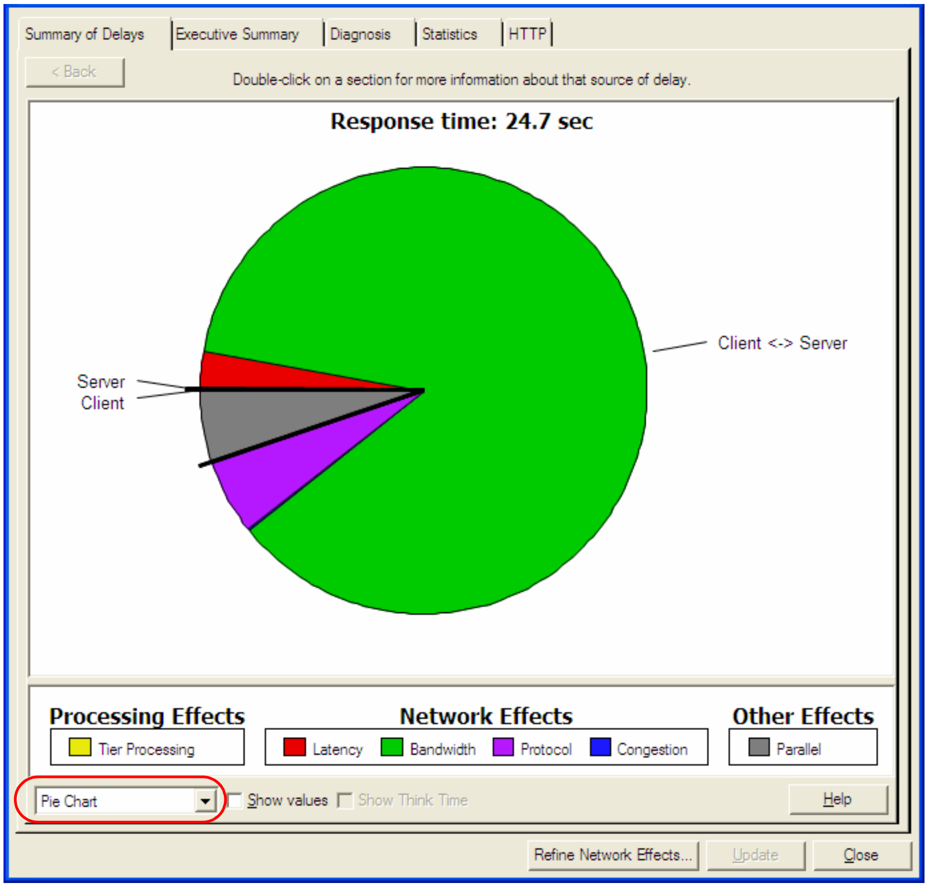
3 Select **low_bandwidth_HTTP** from the list of files, then click **Open**.

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

4 If the Data Exchange Chart is not shown, click the **Data Exchange Chart** tab.



5 Choose **AppDoctor > Summary of Delays (AppDoctor Analysis)**, then verify that the pull-down menu at the bottom left is set to **Pie Chart**.



The Summary of Delays Chart shows that bandwidth is the major source of delay.

6 Close the AppDoctor dialog box.

End of Procedure 3-1

Use QuickPredict

Because the bandwidth of the WAN seems to be a major factor in the slow response time, you will use QuickPredict to evaluate potential WAN upgrades. QuickPredict uses predictive modeling technology to determine how varying bandwidths, latencies, and other network parameters affect application performance. You can study “what-if” scenarios and display the results in a standard analysis panel (x-y graph).

QuickPredict enables you to study the sensitivity of an application to the following network conditions for each tier pair in an AppTransaction Xpert task:

- bandwidth
- latency
- link utilization
- packet loss
- TCP window size

Upgrading Bandwidth

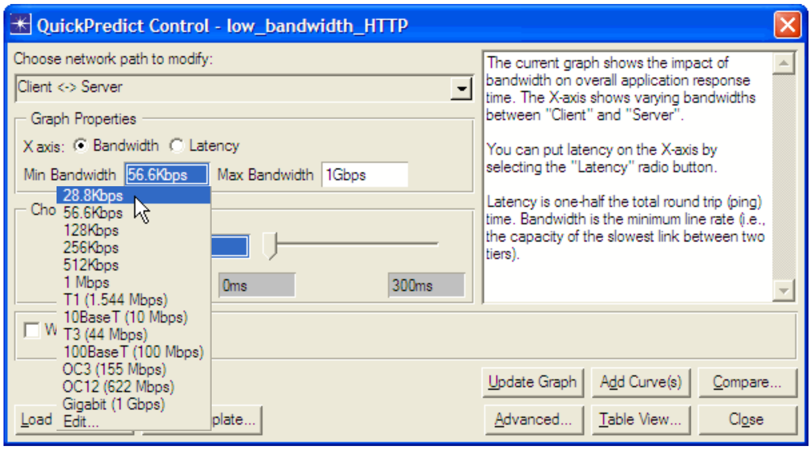
The first study predicts the effect of upgrading the bandwidth of the WAN link. You will simulate the response time of this web page when it is downloaded over a range of WAN bandwidths, from 28.8 Kbps (modem speed) to 1.544 Mbps (T1 speed).

Procedure 3-2 Upgrading Bandwidth

1 Select **Simulation > QuickPredict**.

➥ The QuickPredict Control dialog box and a graph display.

You can change this dialog box to vary latency instead of bandwidth. You can also modify the values of the endpoints by clicking in the box of the value you want to change.

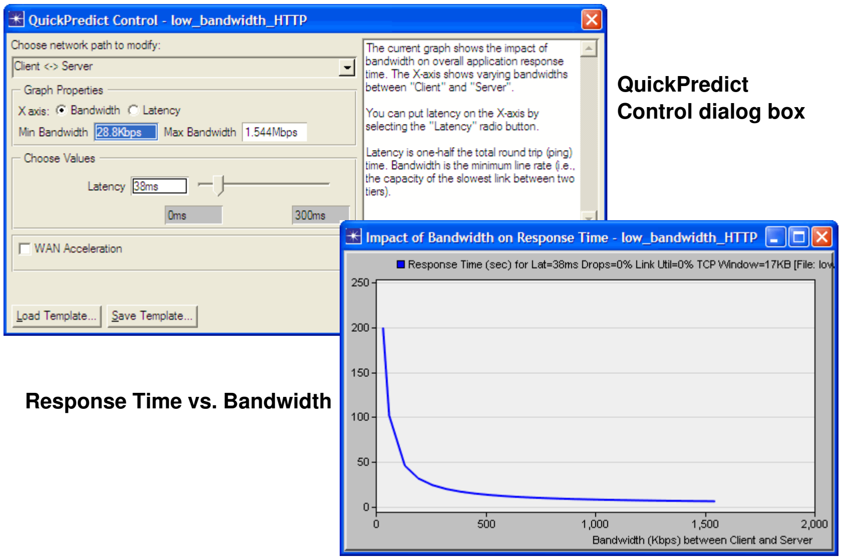


2 Set the following fields in the QuickPredict Control dialog box:

- Min Bandwidth = **28.8Kbps**
- Max Bandwidth = **T1 (1.544Mpbs)**
- Latency = **38ms** (This reflects the latency in the production network, as shown in the LAN Network Diagram.)

3 Click **Update Graph**.

➥ The "Impact of Bandwidth on Response Time" graph shows the effect of the new parameters.



QuickPredict Control dialog box

Response Time vs. Bandwidth

End of Procedure 3-2

Based on the latency in the actual production environment (38 ms), the graph shows the response time of the application over the range specified—28.8 Kbps and 1.54 Mbps. You can change the range of bandwidths used in this study.

Response time for this transaction on a 28.8 Kbps (modem) connection approaches 200 seconds. Response time on a 1.54 Mbps (T1) connection is well under 10 seconds.

Increasing Latency

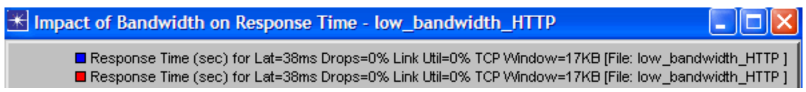
The next network condition you will evaluate is increased latency. This study shows what users might experience when a web application is deployed over a satellite link. To do this study, you will change the latency value in the dialog box and view the impact on application performance in the graph.

Procedure 3-3   Increasing Latency

1 Adjust the windows so you can see both the graph and the QuickPredict Control dialog box.

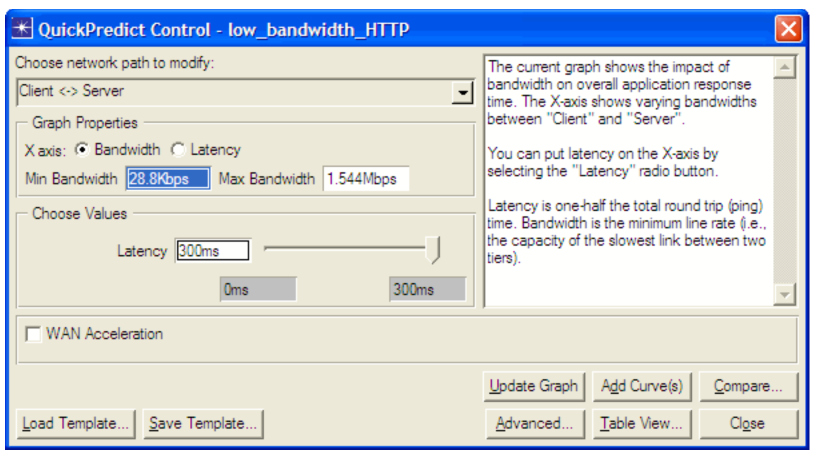
2 In the QuickPredict Control dialog box, click **Add Curve(s)**.

➥ A second (red) curve is added to the graph. You cannot see the curve because it is identical to the existing curve, but you can see a second entry in the Legend.



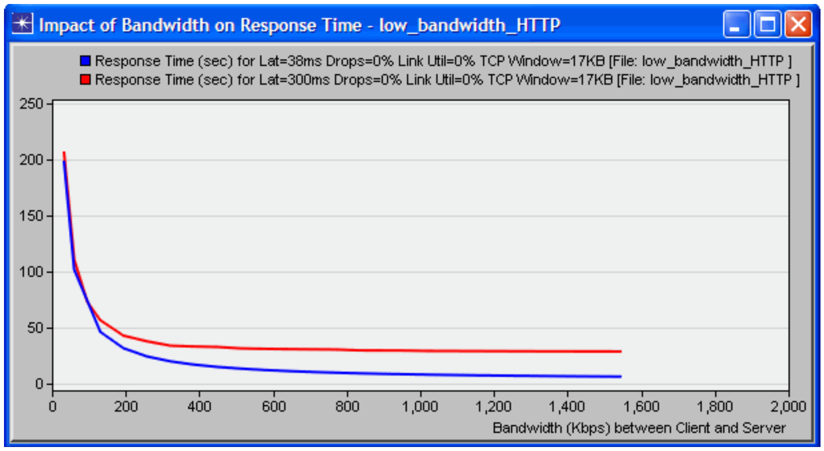
3 Click in the Latency field and select **300ms**.

➥ Notice that the slider indicator moved all the way to the right.



4 Click **Update Graph**.

➥ A new curve reflects the increase in response time due to higher latency.



5 Notice that response time for the new curve is flat after 500 Kbps. It does not decrease with additional bandwidth. You conclude that bandwidth is not the factor that limits the data flow beyond about 500 Kbps.

End of Procedure 3-3

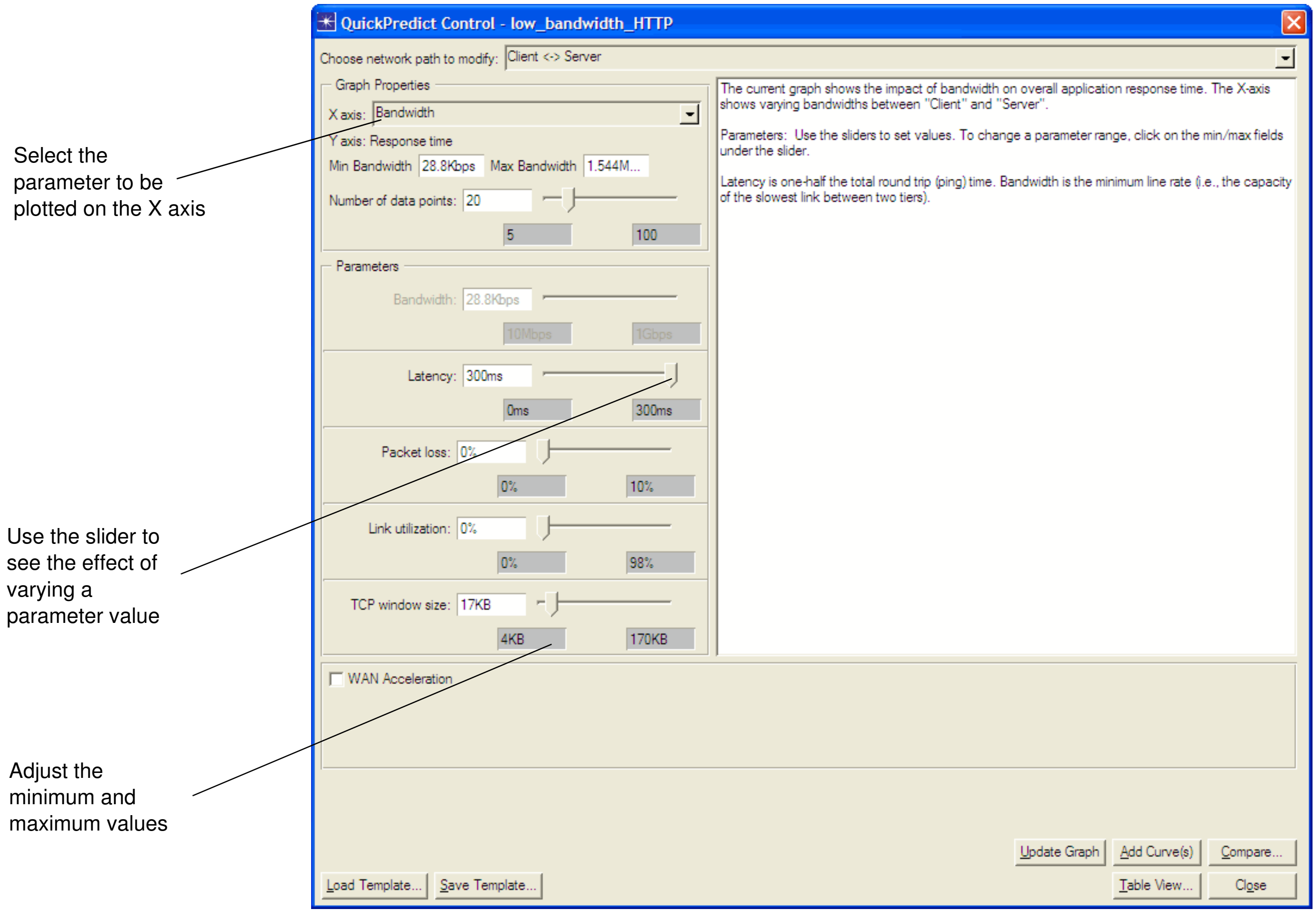
The throughput of an application can be influenced by TCP. TCP has flow control mechanisms that limit the amount of data that is sent. Flow control mechanisms like TCP Windowing can be significant in high latency networks. You can use QuickPredict to validate this theory.

Changing TCP Window Sizes

The first two QuickPredict studies looked at changes to bandwidth and latency. These are only two of the five network parameters that you can assess with QuickPredict. Click the **Advanced...** button to access three additional network parameters: Packet Loss, Link Utilization, and TCP Window Size.

Procedure 3-4 Changing TCP Window Sizes

1 In the QuickPredict Control dialog box, click the **Advanced…** button to display the additional parameters shown in the following figure.



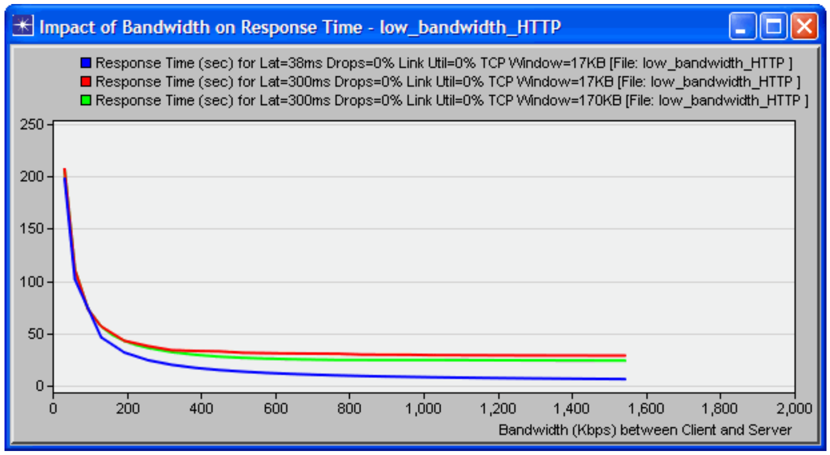
The expanded QuickPredict Control dialog box provides access to five different network parameters whose values can be plotted as a range on the X Axis.

2 Click **Add Curve(s)**.

➥ A third (green) curve is added to the graph.

3 Move the slider for the **TCP Window Size** to the far right (170 KB).

➥ The graph automatically updates the green curve to show the change in the response time.



4 Compare the red (17 KB TCP Window Size) and green (170 KB TCP Window Size) curves. When using a 1.544 Mbps connection, increasing the TCP Window to 170 KB decreases the response time over the satellite link.

5 Try other settings and ranges in QuickPredict.

6 Close all the windows.

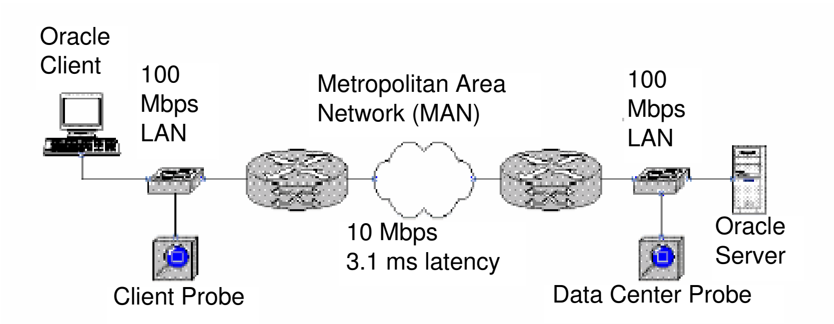
End of Procedure 3-4

Use QuickRecode

QuickRecode adds to the capability of QuickPredict. QuickRecode lets you study how changes to the application flow affect performance. In the previous example, QuickPredict simulated changes to network parameters only. This example shows how changes to the application can solve performance problems.

This part of the tutorial uses a transaction that was captured from the topology shown in the following two-tier Oracle application.

Figure 3-2 MAN Network Diagram



Clients access the Oracle Server in the data center through a high bandwidth, low-latency, Metropolitan Area Network (MAN). The application response time to the client is 12 seconds.

Packet trace captures were taken simultaneously at the client and server; both captures were filtered by host name to isolate the application of interest. These captures were merged and synchronized to obtain the best analysis of delays at each tier and the network.

The merged packet trace file was used in a previous tutorial where the results showed that the application was not efficient at exchanging data between tiers. It was determined that there was little that could be changed in the network to make the application faster.

This tutorial proposes and tests virtual changes to the application. First we are going to analyze the application using AppDoctor and QuickPredict. Then we will test the proposed changes using QuickRecode and QuickPredict.

Analyze the Oracle Application

To start, you will open the model and analyze the Oracle application.

Procedure 3-5   Analyzing the Oracle Application

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

➥ The **Open** dialog box appears.

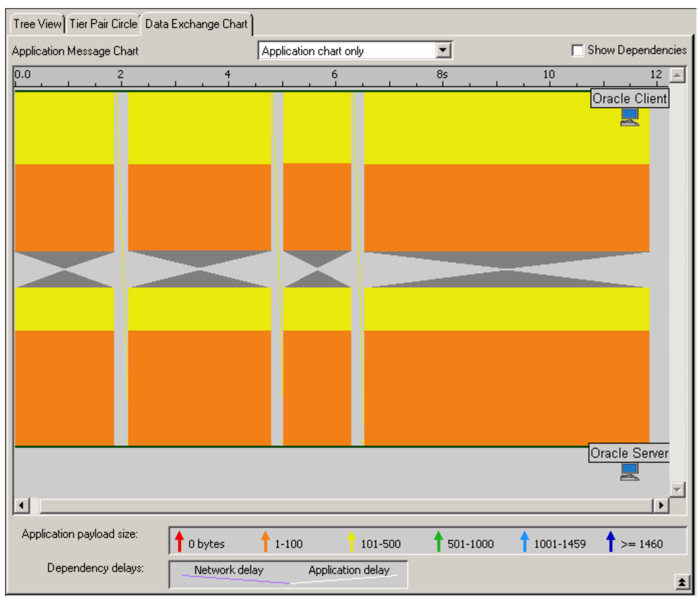
2 Select **Transaction Analyzer** in the “Files of type” menu (Windows) or Filters menu (Linux) if it is not already selected. Then navigate to the **examples** folder using the following path:

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

3 Select **Oracle_two_tier_app_over_MAN** from the list of files, then click **Open**.

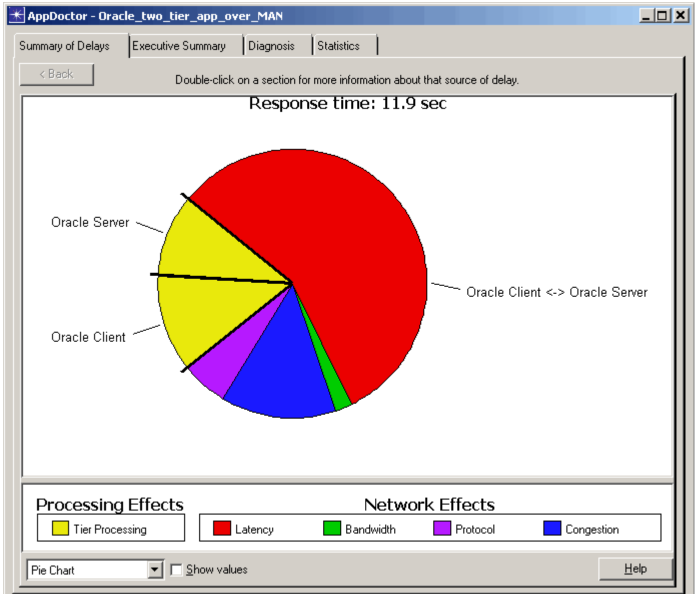
If the “Getting Started Analyzing an Application Transaction” window appears, click **Close**.

4 Click the **Data Exchange Chart** tab if it is not already selected.



5 Select **AppDoctor** > **Summary of Delays (AppDoctor Analysis)**.

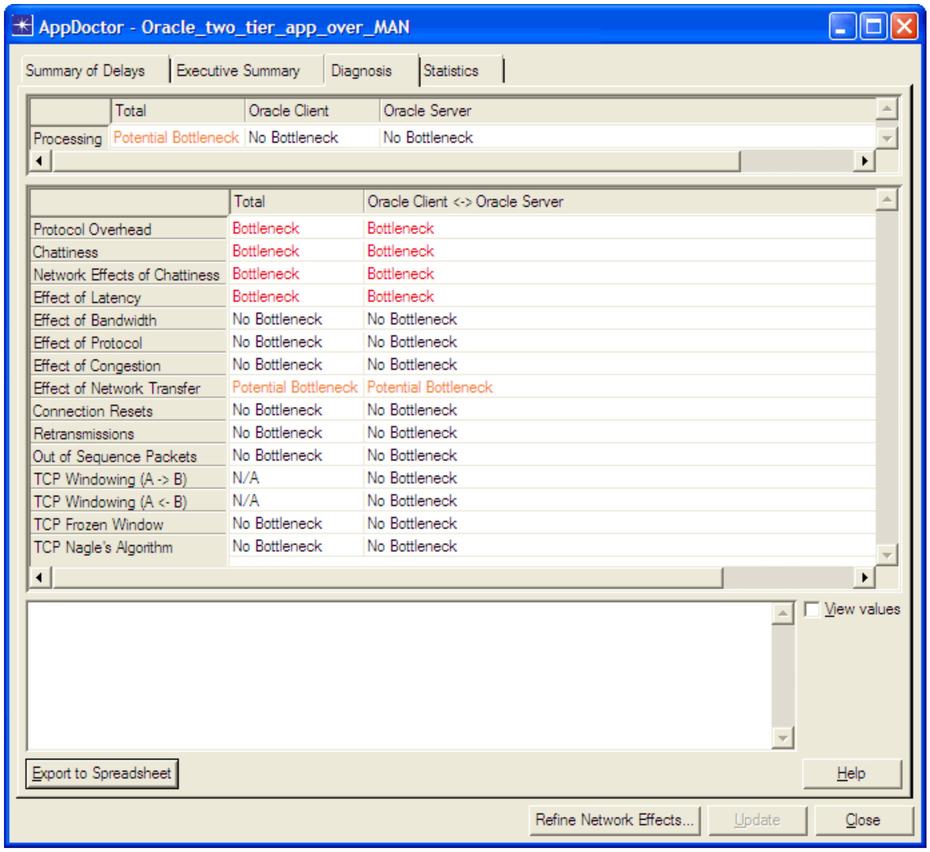
➥ The Summary of Delays shows that Latency Delay (the red slice) is about 60 percent of the total application response time.



To view more information about a category, place your cursor in a section of the pie chart until a tooltip appears.

6 Click the **Diagnosis** tab.

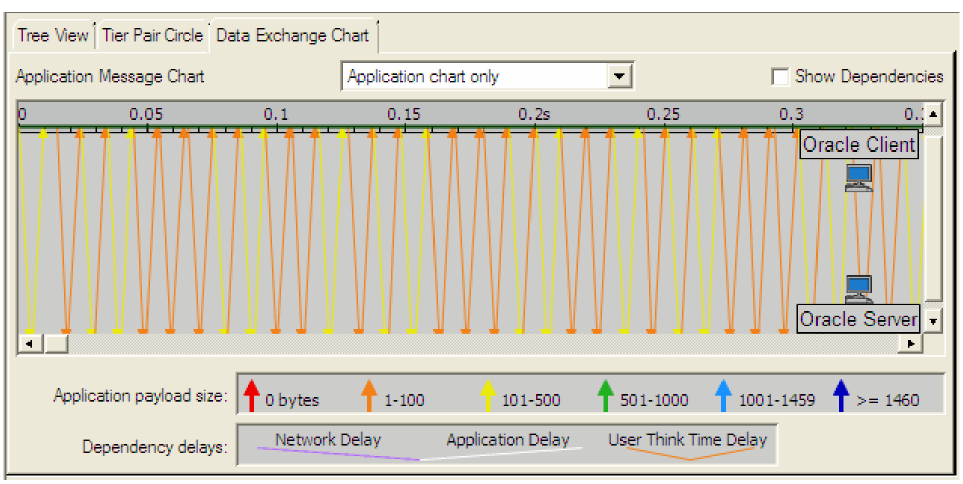
➥ The Diagnosis shows that this application is chatty and that the cost of the chattiness is high on this network.



7 Close the AppDoctor dialog box.

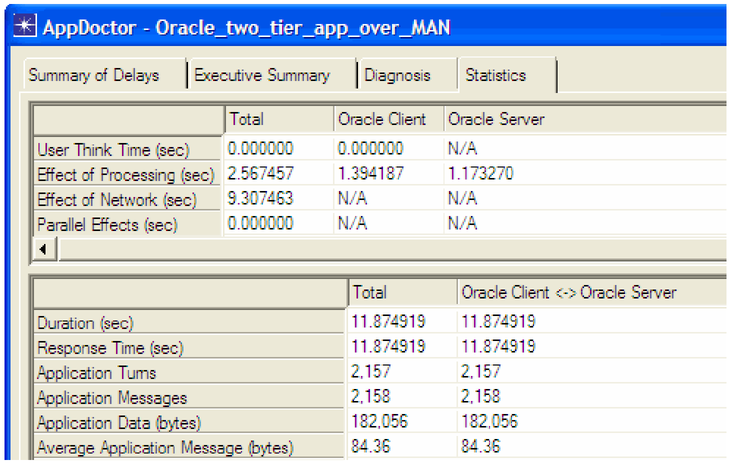
8 Select **View** > **Set Visible Time Range**. Set the **Start Time** to **0.0** and the **End Time** to **0.3**. Click **OK**.

➥ The Data Exchange Chart zooms to the beginning of the transaction.



The chart highlights the inefficient exchange of messages in this application. You can see the repetitive back-and-forth path of each tiny request and response (called an application turn in AppTransaction Xpert).

9 Select **AppDoctor** > **Statistics** to see the total number of application turns.



	Total	Oracle Client	Oracle Server
User Think Time (sec)	0.000000	0.000000	N/A
Effect of Processing (sec)	2.567457	1.394187	1.173270
Effect of Network (sec)	9.307463	N/A	N/A
Parallel Effects (sec)	0.000000	N/A	N/A

	Total	Oracle Client <-> Oracle Server
Duration (sec)	11.874919	11.874919
Response Time (sec)	11.874919	11.874919
Application Turns	2,157	2,157
Application Messages	2,158	2,158
Application Data (bytes)	182,056	182,056
Average Application Message (bytes)	84.36	84.36

This application has 2,157 turns. Each turn causes a delay to the application as it travels across the network. Although the latency is only 3.1 ms in this example (as shown in the MAN Network Diagram), it causes about 60 percent of the 12 second response time.

10 Close the **AppDoctor** dialog box.

End of Procedure 3-5

How sensitive is this application to the additional latency of a Wide Area Network?

You can use QuickPredict to understand the application behavior under different network conditions.

Evaluate the Impact of Latency

To evaluate the impact of latency, complete the following procedure.

Procedure 3-6 Evaluating the Impact of Latency

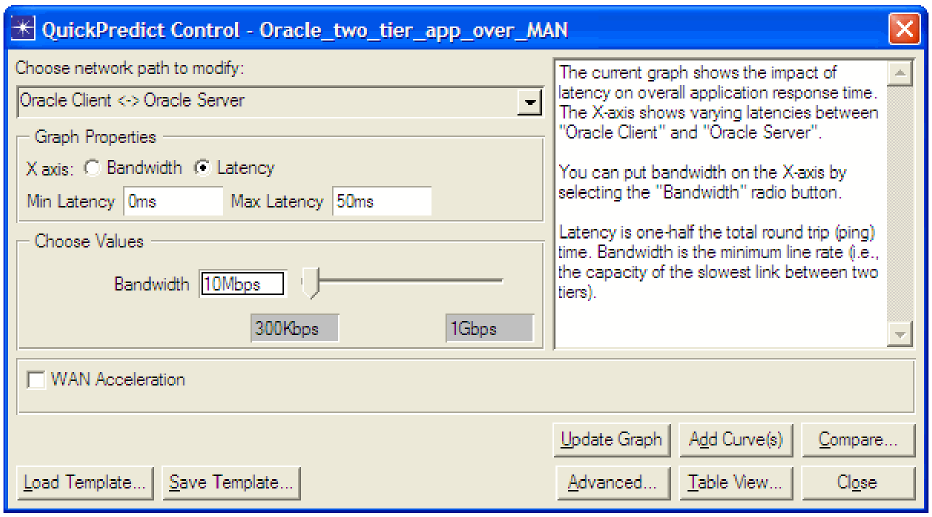
1 Select **Simulation** > **QuickPredict**.

➥ The QuickPredict Control dialog box and a graph appear.

2 Click the **Latency** radio button to change the value of the **X Axis** to **Latency**.

3 Verify that the **Min Latency** is set to **0 ms** and that the **Max Latency** is set to **50 ms**.

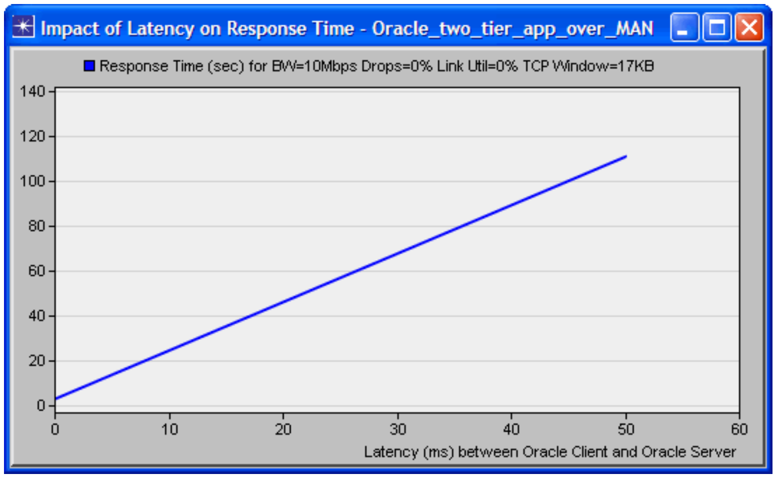
4 Set the Bandwidth value to **10Mbps**.



5 Click **Update Graph**.

➥ The QuickPredict graph is updated.

The graph shows that the existing application is very sensitive to latency on the network. Because the transaction sends data back and forth over the network more than 2,100 times, small changes to latency have a dramatic effect. If you deployed this application over a WAN circuit with 50 ms of delay, the total response time would be about 110 seconds.



6 Close the **QuickPredict Control** dialog box.

7 Close the QuickPredict graph.

End of Procedure 3-6

Visualize the Effect of Code Changes

QuickRecode lets you edit parts of a Transaction Analyzer model manually. You can then predict the behavior of your hypothetical application using QuickPredict or event-based simulation. Using this approach, you can see the effect of making specific changes to an application without changing the actual application code.

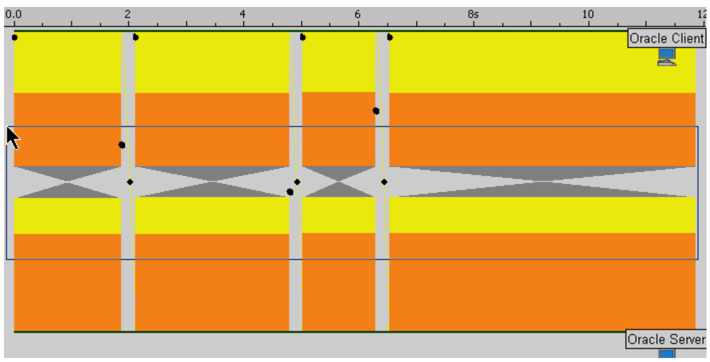
In this example, you use QuickRecode to minimize the chattiness of this application. This is replicated in the real application by changing the way Oracle accesses the data from the server.

Procedure 3-7 VIsualize the Effect of Code Changes

1 Right-click in the background of the **Data Exchange Chart** and select **Full Zoom**.

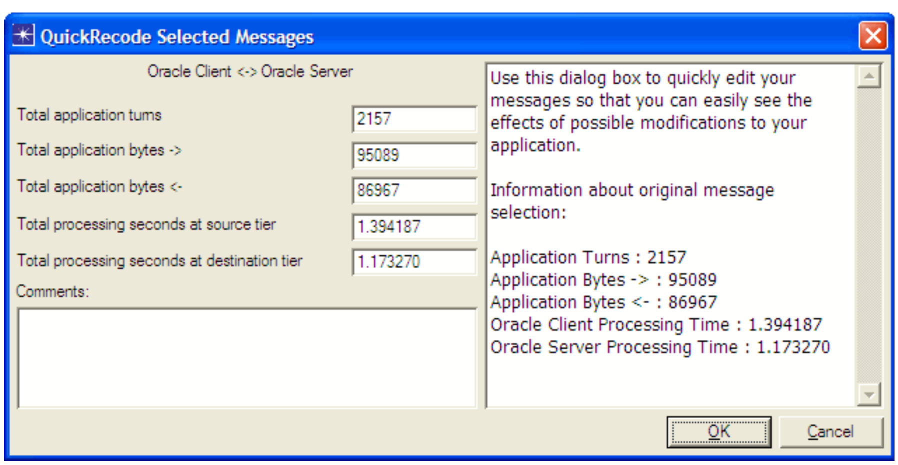
2 Select all messages between the client and the server: use the mouse to draw a box across all the messages.

➥ The groups in the chart are selected.



3 Right-click on one of the selected groups and choose **QuickRecode Selected Items**.

➥ The **QuickRecode Selected Messages** dialog box appears.



Use this dialog box to modify the behavior of the application transaction. You can modify five primary controls:

- total number of application turns (**Total application turns**)
- amount of application data in each direction (**Total application bytes ->** and **Total application bytes <-**)
- processing delay for each tier (**Total processing seconds at source tier** and **Total processing seconds at destination tier**)

Changing these values affects the response time of the application. By changing the number of application turns from 2157 to 200, you will simulate a change to the database access. QuickPredict will show how this “recoded” application behaves over different network conditions.

3.1 Change the **Total application turns** to **200**.

3.2 Click **OK** to accept this value and close the dialog box.

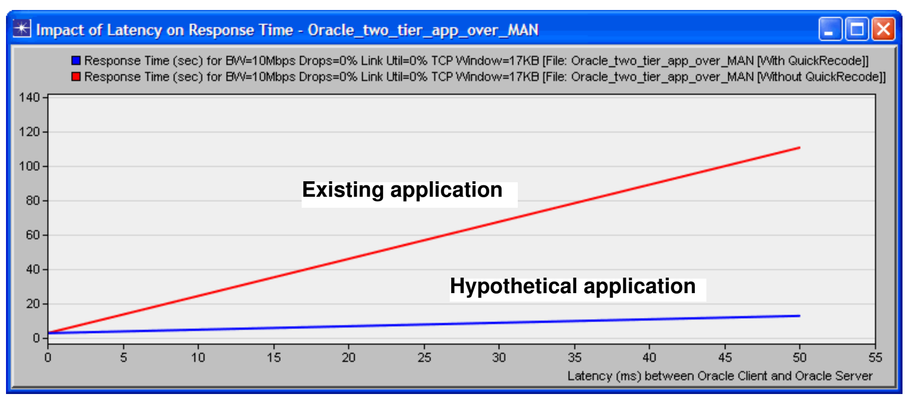
➥ A red band appears around the selected messages in the Data Exchange Chart and the group changes color to indicate that it has been edited.

3.3 Select **Simulation > QuickPredict**.

➥ The **QuickPredict Control** dialog box appears.

3.4 Click **Update Graph**.

➥ A QuickPredict graph with two curves appears. The red curve shows the behavior of the existing application over a range of latencies from 0 to 50 ms. The blue curve shows the response time of the hypothetical application with the recode.



Because the recode reduced the number of application turns, the network latency and overall response time are reduced. At 50 ms of latency, the existing application has a 110-second response time. The recoded application has a response time of about 15 seconds. This reduction can justify the expense of recoding the application.

4 Close the file.

End of Procedure 3-7

Conclusion

This tutorial showed how to use QuickPredict and QuickRecode to predict the response time of an application.

In the first example, you tested a network upgrade. In the second example, you tested a change to the application. Both examples show how AppTransaction Xpert can predict the effect of change on application performance.

The techniques used in this tutorial can help you to:

- Demonstrate the Return on Investment (ROI) of a specific upgrade
- Determine how an application performs before you deploy it in a production network