White Paper

Pre-Deployment Application Performance Analysis Using Cisco Application Analysis Solution

Networked applications are now the backbone of nearly every business activity. When an application does not perform well, it costs money. For businesses to maintain a competitive edge and ensure integrity with customers and vendors, IT professionals today require tools to predict, avoid, and mitigate costly network and application disruptions.

SUMMARY

Cisco® Application Analysis Solution (AAS) allows you to visualize, diagnose, and perform high-level predictive analysis of an application's performance. Useful for both pre-deployment analysis and for troubleshooting existing applications, this tool helps IT groups to analyze and optimize the interactions between the network, servers, and applications through offline modeling. It bases analyses on actual packet traces collected from the production network.

Cisco AAS is central to the Cisco Network Application Performance Analysis (NAPA) solution, which is a comprehensive set of tools and services that provides information about application and network performance. More information on the Cisco NAPA Solution can be found at http://www.cisco.com/go/napas.

This document discusses the challenges and workflow details for pre-deployment analysis. In a pre-deployment environment, Cisco AAS assists in rolling out new or updated applications by analyzing both the impact the network will have on the performance of the new application as well as the impact the new application will have on the network.

PLANNING AHEAD, PRE-DEPLOYMENT CHALLENGE

A company is testing the deployment of a new Oracle application on its network. This application has been successfully tested in the lab and is now being deployed to a pilot site connected via a metropolitan area network (MAN). The MAN circuit has a bandwidth of 10 Mbps and a latency of 3 to 4 milliseconds (ms).

The ultimate goal is to deploy this application over the company's Frame Relay WAN. However, users at the pilot site are complaining of slow response times.

PRE-DEPLOYMENT SOLUTION WORKFLOW

Evaluating the performance of an application prior to deploying it over the network requires the following workflow:

1. Profile application response time

    - Import application packet data

    - Visualize the application

2. Identify performance issues and determine root cause

    - Examine source of delays

    - Diagnose bottlenecks

    - Perform quantitative analysis

3. Analyze application response time under varying network conditions

- Visualize application response time
- Predict response times for pilot sites

4. Suggest steps for improving application response time

- Virtually re-code application
- Visualize effect of code changes

Profile Application Response Time

Captured files will be opened and analyzed using the Cisco AAS Application Characterization Environment, or "ACE." ACE is an enabling technology for visualizing and analyzing application performance problems. The input to Cisco AAS is packet data. Cisco AAS reads packet traces that can be captured from a Cisco Network Analysis Module (NAM) or using capture agents that come with Cisco AAS. The capture agent can be installed on Windows, UNIX, and Linux operating systems.

Import Application Packet Data

To evaluate the user's experience with the application, trace captures are taken simultaneously at the client and server. In this scenario, a Cisco NAM installed in a Cisco Catalyst® 6500 Series Switch on the same LAN segment is used to capture the server transaction while the capture agent is used for the client packet trace. Both captures are filtered by host name to isolate the application of interest; in this case the application is Oracle.

Visualize the Application

The first step to analyzing the performance problem experienced by the pilot user is to view the application trace in the Data Exchange Chart (see Figure 1). The Data Exchange Chart shows the application and network packets between the client and database server during the time the transaction was captured. The client and database server are called "tiers."

Figure 1.   Data Exchange Chart Showing Tiers and Total Response Time
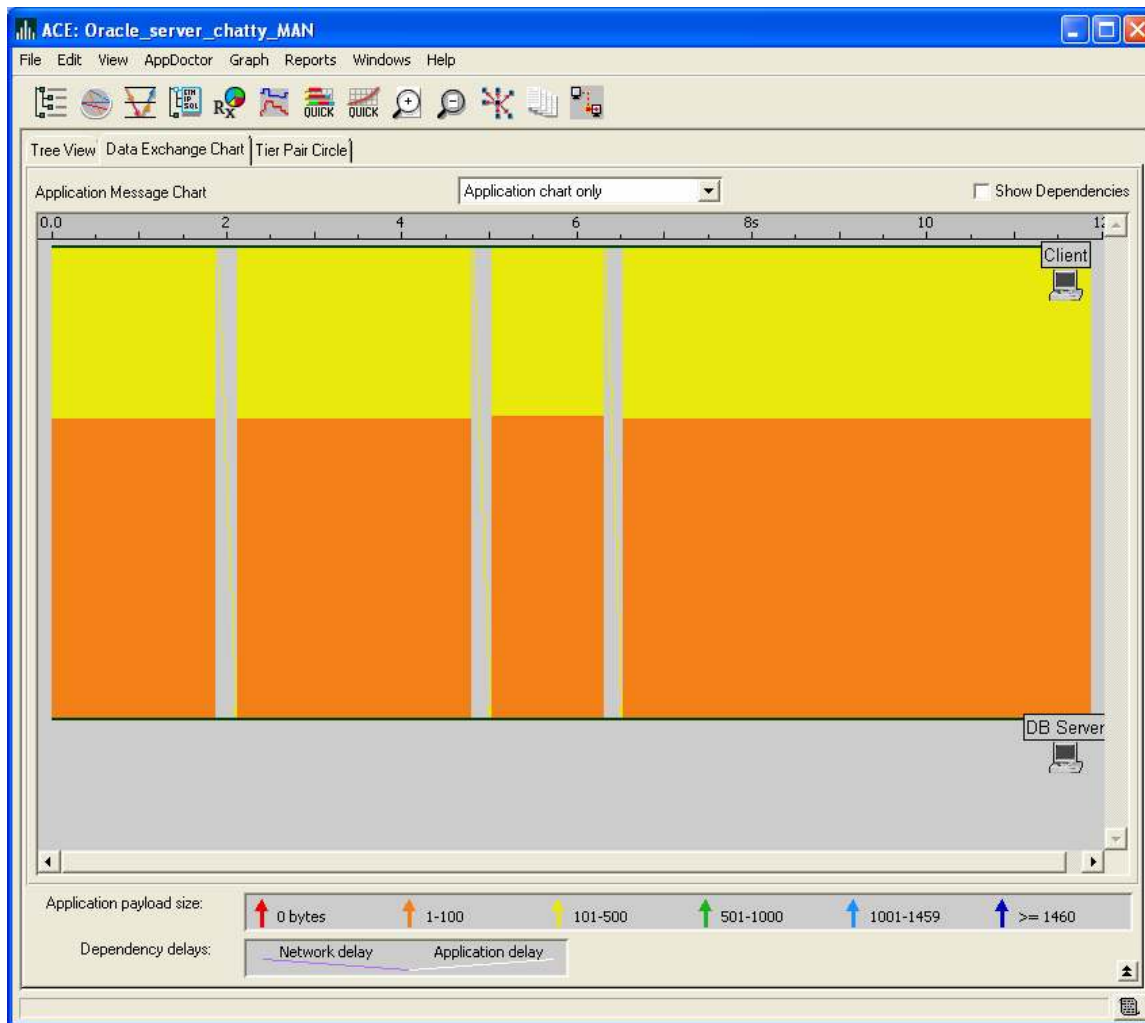
The Data Exchange Chart visually characterizes the application trace:

- The timeline on the top graphs shows the total response time of the application task. The response time for this application is almost 12 seconds.
- A horizontal line represents each application tier: client and database server.
- Message request and response times are shown between each tier.
- The messages are colored by their application payload size. The legend is shown at the bottom of the window.
- Each individual application message is represented by an arrow that starts at the source tier and ends at the destination tier (see Figure 2). The arrow color represents the number of application bytes in the message. Each colored bar represents a group of messages; the amount of color represents the percentage of messages that fall into each application-payload size category.

Figure 2.   Data Exchange Chart Showing Each Individual Application Message
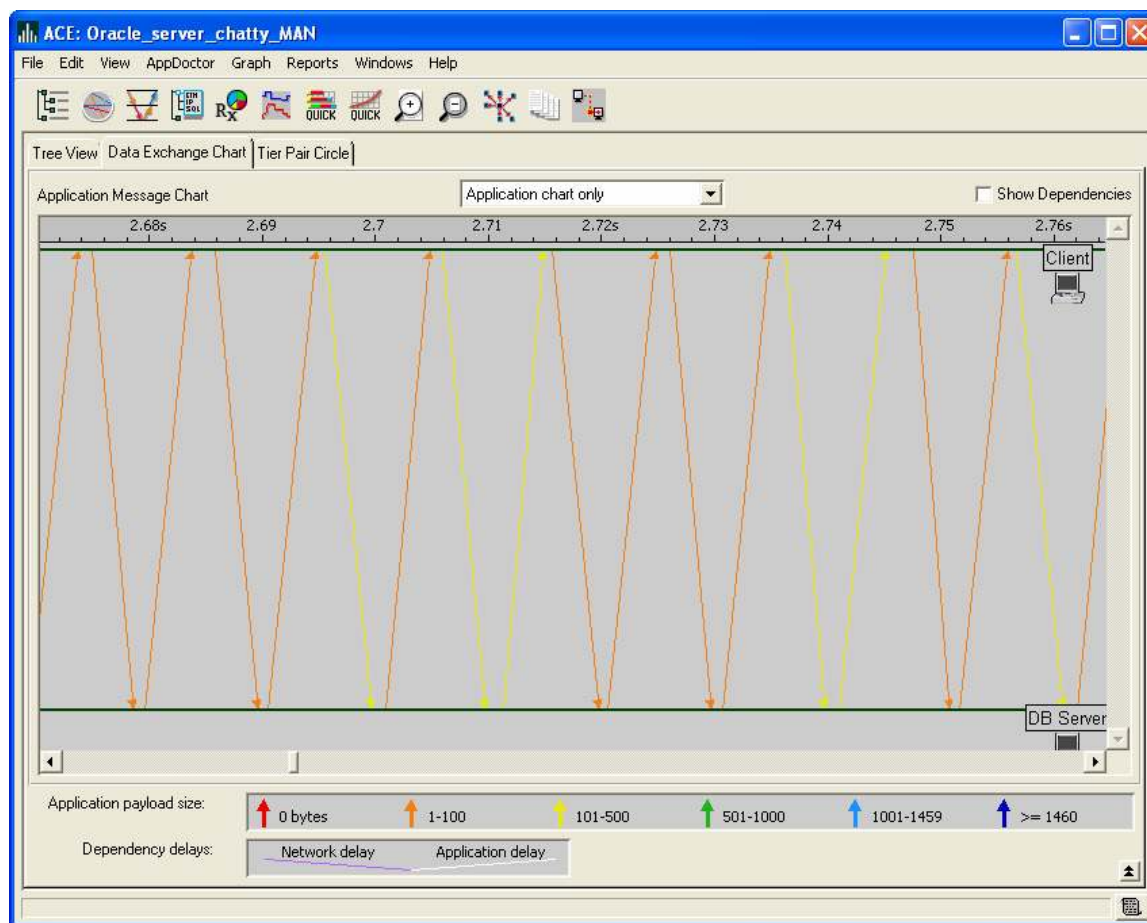
More than 50 percent of the messages are orange, meaning the payload size for each message ranges between 1 and 100 bytes. You can immediately conclude that the application is sending many small messages and the total application response time is nearly 12 seconds.

Identify Performance Issues and Determine Root Cause

After determining that the application is transmitting many small packets and the total response time is significant, the next step is to further understand the impact that small packet size has on the overall response time.

Examine Sources of Delays

The sources of delay are summarized in convenient diagrams. Thresholds for key application statistics are used to generate informative reports that characterize problems. The underlying technology in Cisco AAS that performs these functions is called AppDoctor. Using AppDoctor in the Cisco AAS tool, the Summary of Delays chart shows you how individual delays contribute to the total application response time. AppDoctor's diagnosis will then identify the bottlenecks. Finally, more quantitative analysis can be performed looking at AppDoctor's statistics.

The Summary of Delays chart (Figure 3) breaks down the individual delay components and their effect on the total application response time, including:

- Tier processing delay—The total time it takes to process the application at each tier. This includes CPU processing time and user think time.
- Latency—Delay due to latency in the network. Latency is the time for one bit to be transmitted across the network.
- Bandwidth delay—Delay caused by the limited bandwidth of the network.
- Protocol and congestion delay—Delay as a metric of network restriction to packet flow. It may be caused by:
  - Packet queuing in the network or
  - Flow-control mechanisms imposed by network protocols. TCP, for example, has several built-in flow-control mechanisms including TCP window resizing.

Figure 3. Summary of Delays, Including Client-to-Database-Server Latency
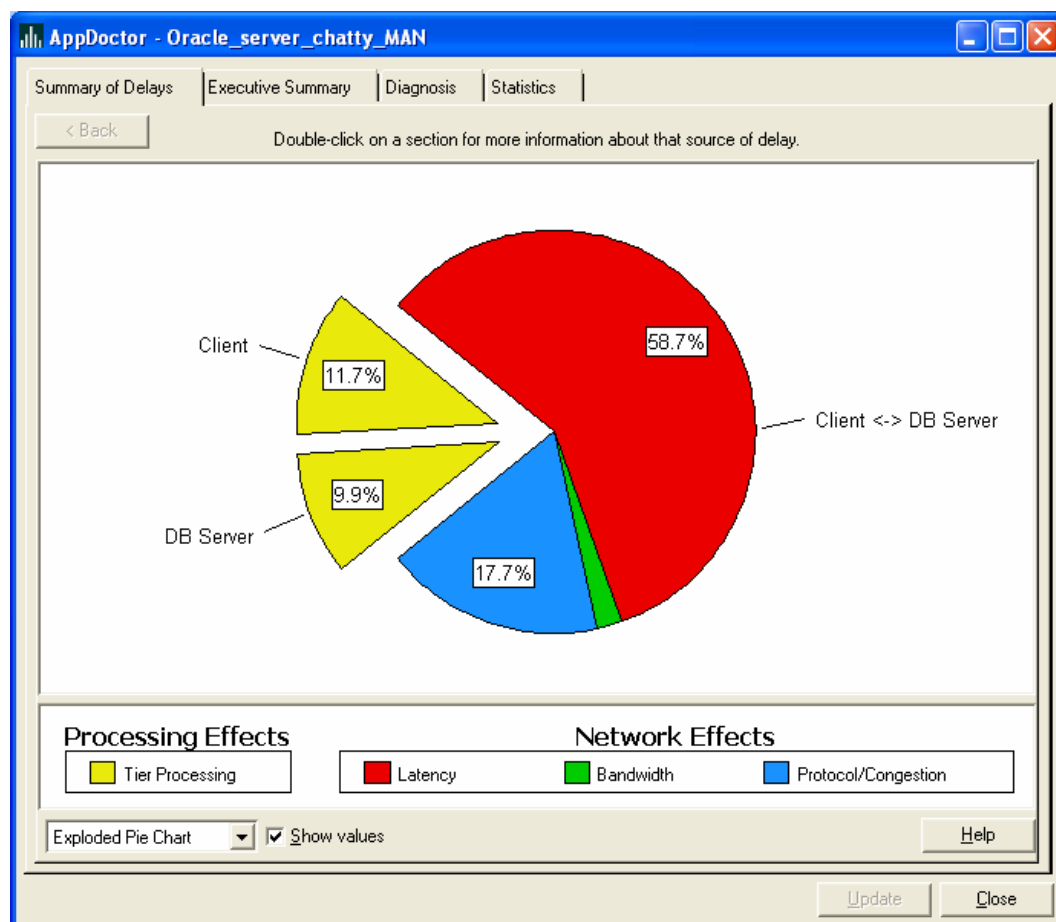
The Summary of Delays shows that network latency accounts for about 60 percent of the total application response time. Processing time and bandwidth do not significantly impact response time. Therefore adding bandwidth or server capacity will not improve the user's experience in this scenario.

Diagnose Bottlenecks

AppDoctor Diagnosis (see Figure 4) provides a more detailed view of the potential bottlenecks affecting this transaction as well as recommendations on how to improve the application. AppDoctor Diagnosis tests the current transaction against issues that often cause performance problems for network-based applications, grouped by category. Values that exceed specific, user-configurable thresholds are marked as bottlenecks or potential bottlenecks.

Figure 4. Bottlenecks Identified by AppDoctor Diagnosis
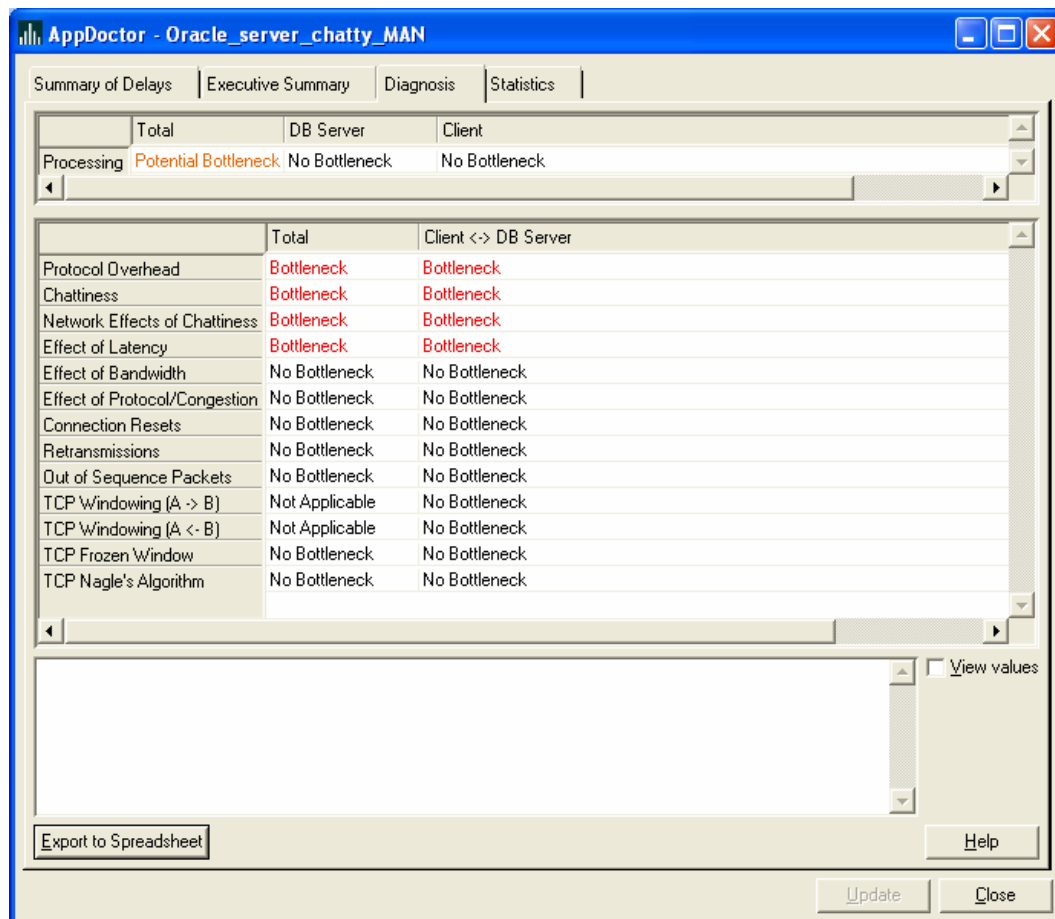
The diagnosis shows that this application is very chatty, meaning many small messages are sent between the client and database server. The communication overhead is significant for this application because each message has a delay due to network latency.

For Chattiness, AppDoctor recommends sending more application data per application turn (that is, when an application changes the direction of the data flow), which would decrease the impact of latency on the overall application response time.

For Effect of Latency, AppDoctor recommends moving the tiers closer together to reduce the propagation delay caused by geographical constraints.

Perform Quantitative Analysis

The AppDoctor feature in Cisco AAS can provide further details on application transactions through summary statistics including:
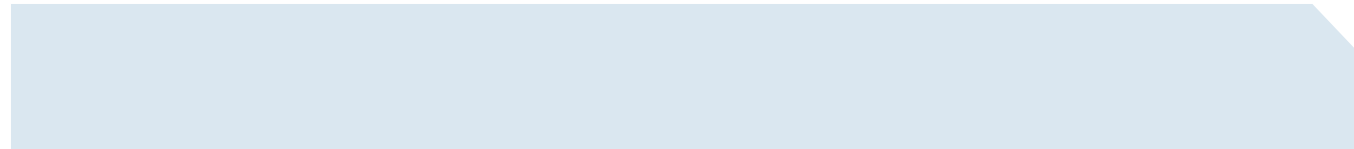
- Amount of time each tier spent processing the transaction
- Number of messages sent between application tiers
- Amount of data sent between application tiers
- Average network packet size
- Amount of data loss including packet retransmissions

Figure 5. Application Transaction Statistics
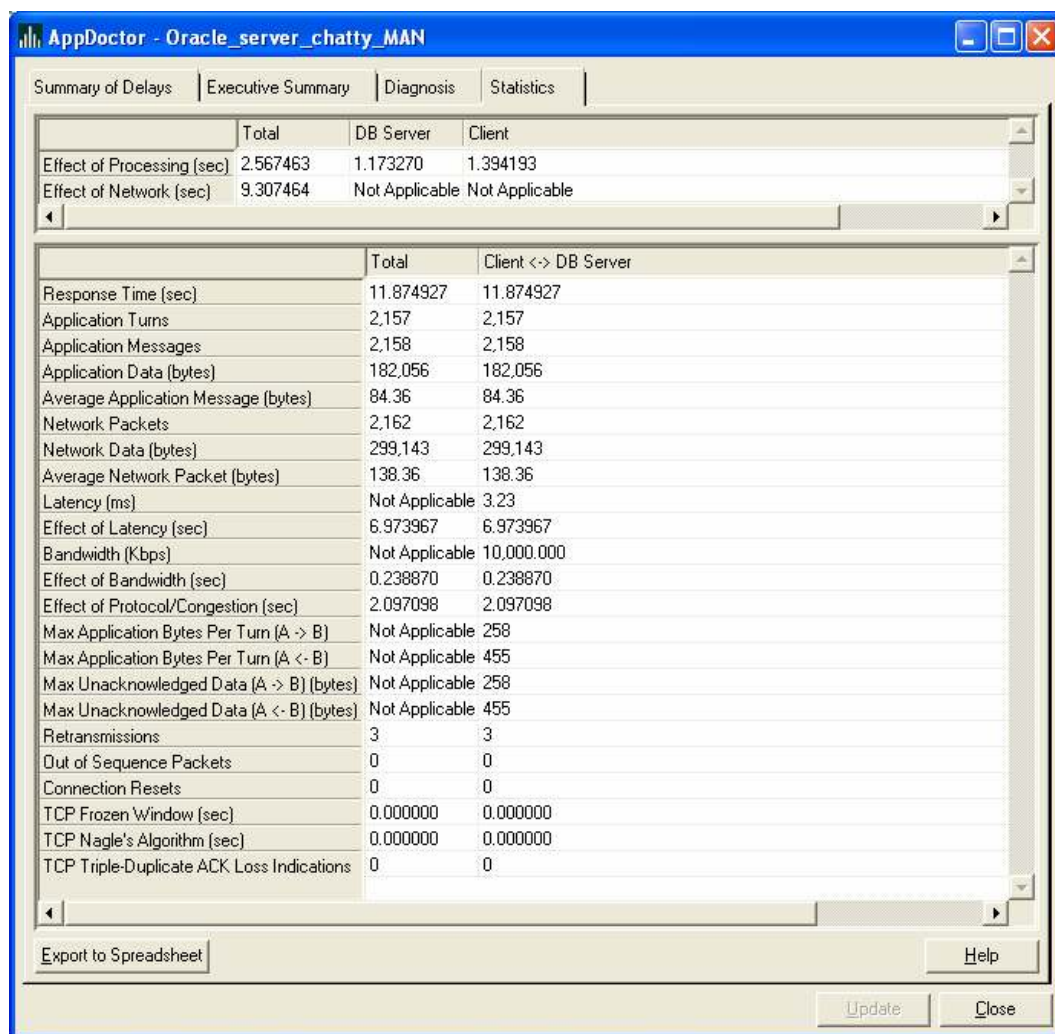
Notice the response time for this application is 11.87 seconds, which corresponds to the value that you saw previously in the Data Exchange Chart.

Several statistics are relevant to this study. This report examines two in particular: the number of application turns and the maximum data exchanged in each turn.

Each change in direction is called an application turn because the application changes the direction of data flow. Applications with many application turns are generally considered chatty and are sensitive to network delay. The sensitivity occurs because each message must be received at a tier before the corresponding response can be sent; as a result, each message is affected by network latency. Notice that the application used 2157 turns to exchange 182,056 bytes of data.

This confirms the previous conclusion in which chattiness and the network effect due to latency is the primary cause of the poor response time.

Also note that the effect of latency is 6.973 seconds. The effect of latency alone, aggravated by the 2157 applications turns, accounts for about 6.97 seconds or 58.7 percent of the total transaction response time. Because latency is largely a product of geographical distance and network hops, adding bandwidth will have a minimal effect on the response time. To minimize the effect of latency on this application, you can reduce the latency on the circuit by moving the tiers geographically closer together or reduce the number of application turns.

Once the application performance problem has been identified, Cisco Application Analysis Solution can be used to virtually re-code the application and reconfigure the network to determine the optimal solution.

Analyze Application Response Time Under Varying Network Conditions

The QuickPredict and QuickRecode functions in Cisco AAS are used for doing predictive studies that justify changes to network infrastructure or application behavior respectively.

QuickPredict is an analytic simulation mechanism that enables you to test the performance of an application quickly under different network conditions. You can test possible network upgrades to evaluate the impact they might have on application performance.

Visualize Application Response Time and Predict Response Times for Pilot Sites

The QuickPredict bar chart, shown at the bottom of Figure 6, allows high-level predictive analysis on application tasks. You can change network characteristics such as: bandwidth, latency, packet loss, link utilization, and TCP window size, and plot on a bar graph the impact of these factors on the end-to-end response time.

Figure 6.   Use QuickPredict to Evaluate Network Changes on Application Performance
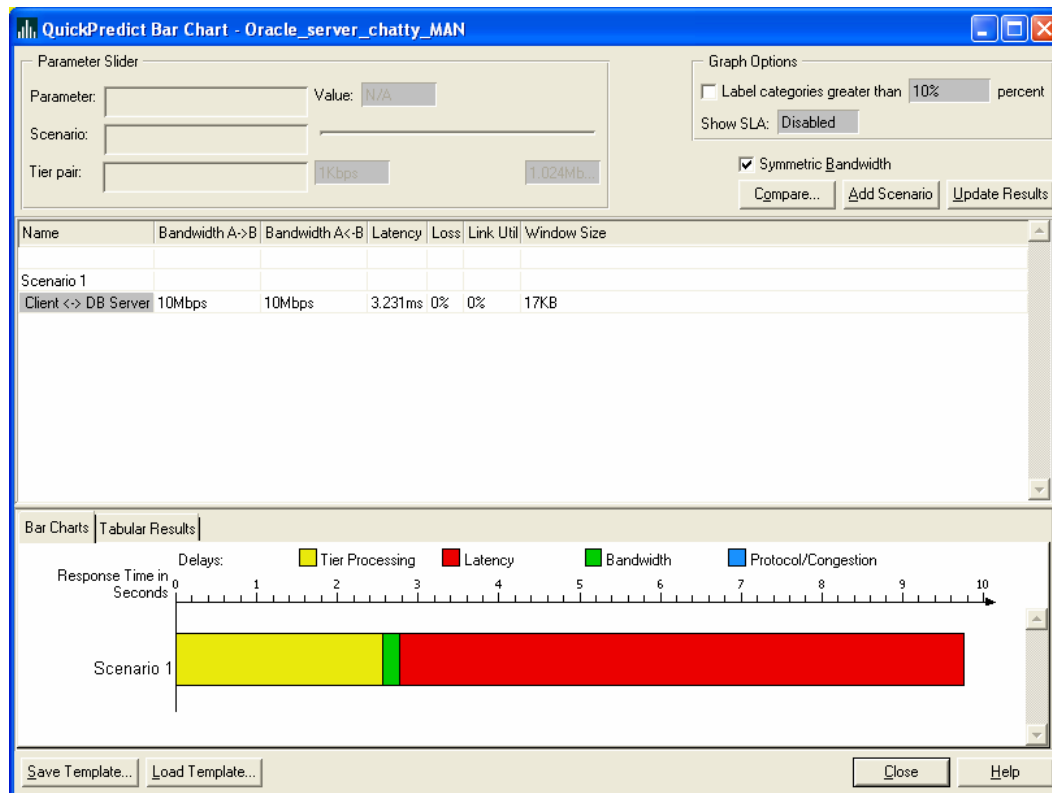

Different scenarios can be visualized and compared on different graphs; you can create and predefine a set of variables into templates.

As noted earlier, the original traces file was captured in the pilot environment; all remote users are connected through a 10 Mbps MAN network, with latency between 3 and 4 ms. The subject transaction takes 11.87 seconds to complete. Cisco AAS determined the major application bottleneck is due to chattiness or small application messages, and is aggravated by network latency.

The ultimate goal is to deploy this application over a Frame Relay WAN. Target sites include New York, Washington DC, and Sydney, Australia. The latencies are 20 ms, 30 ms, and 175 ms respectively (Figure 7).

Figure 7.   Locations and Latencies
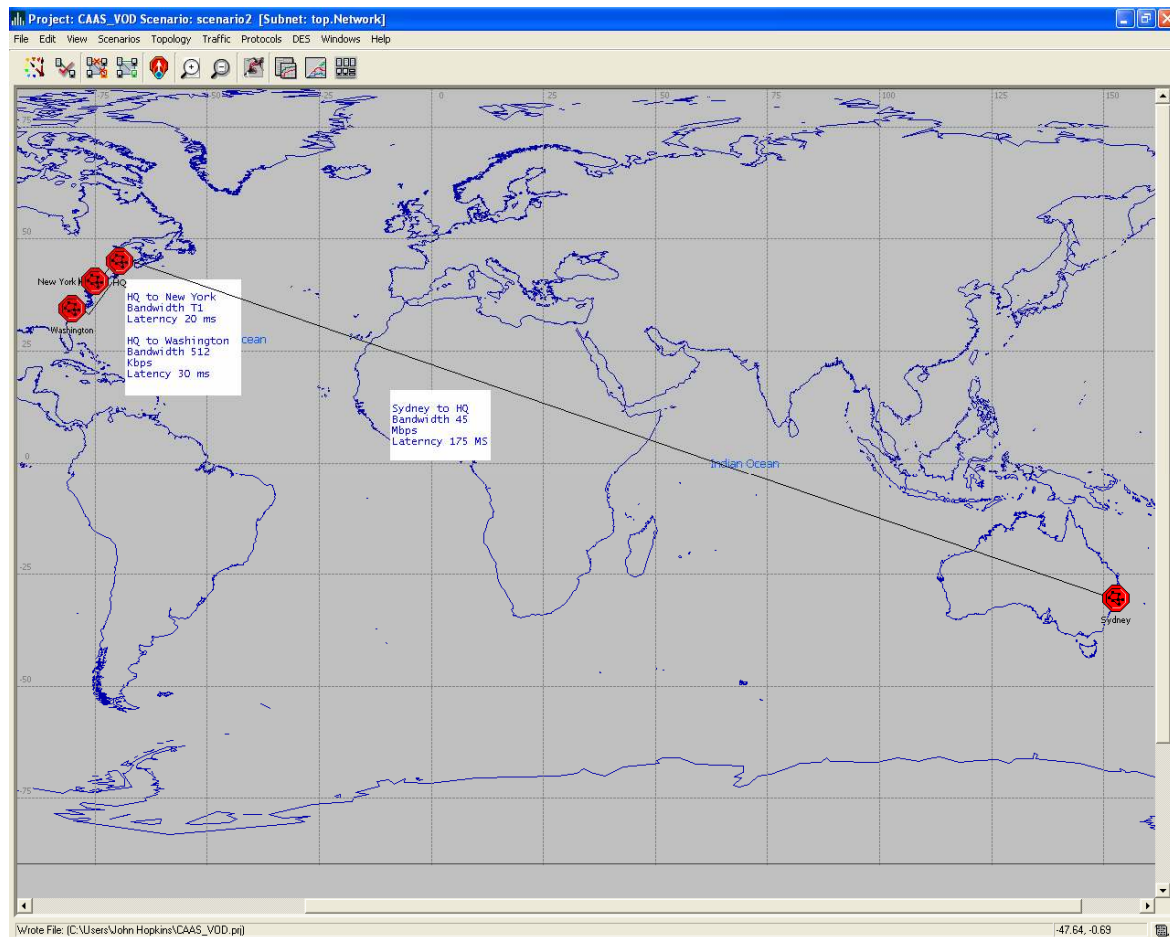
The QuickPredict bar chart shows the baseline transaction and response time. By adding scenarios for each pilot site (New York, Washington DC, and Sydney) and using "update results" you can see the expected application response time for each site. The bar chart (Figure 8) shows that the existing application is very sensitive to latency on the network. Because the transaction sends data back and forth over the network more than 2100 times, changes to latency have dramatic effect. If you deploy this application over the WAN, users accessing the database server over high-latency links will experience unacceptable application response times.

Figure 8.   Baseline Transactions and Latency
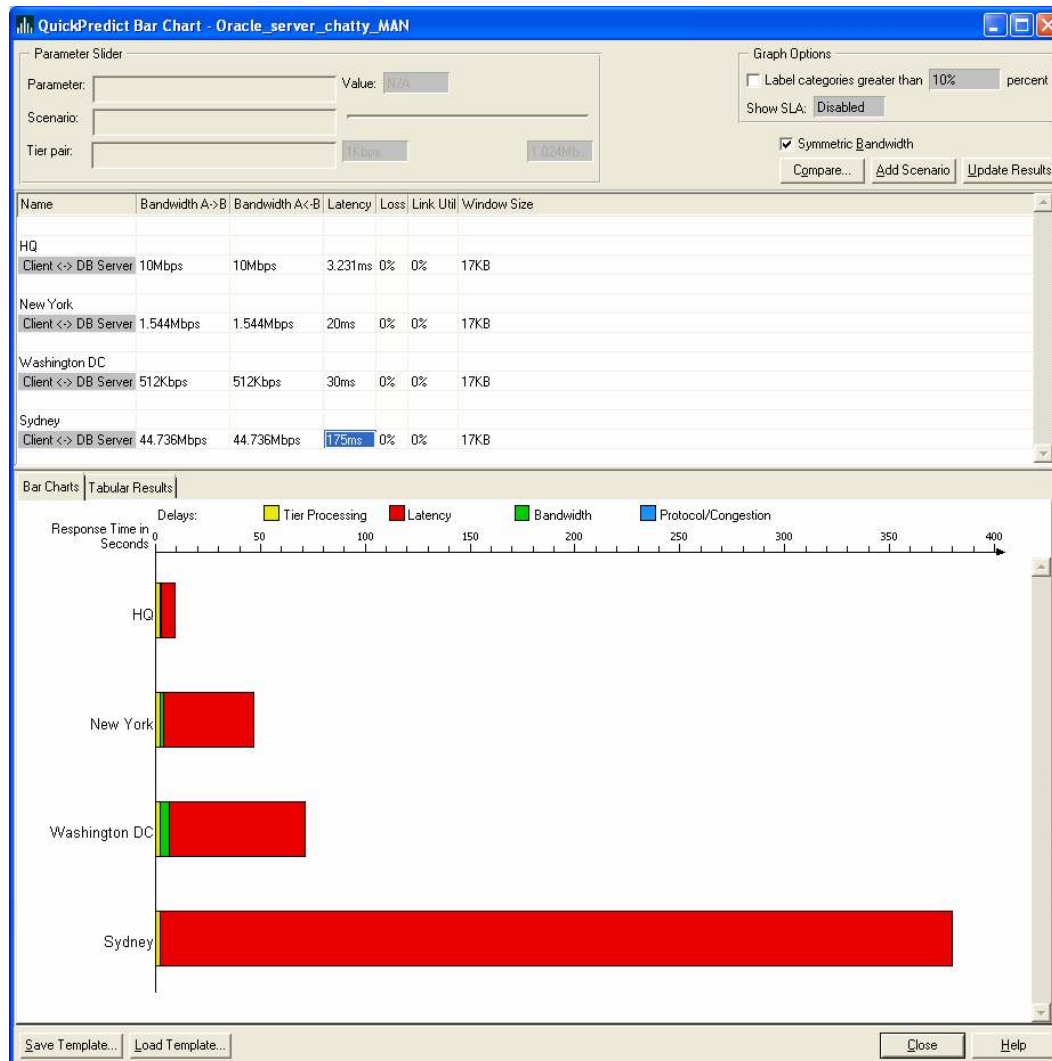
Suggested Steps for Improving Application Response Time

Initially, Cisco AAS determined that the application response time was more than 11 seconds, and the users at the pilot sites were complaining of slow response times. AppDoctor determined the slow response time was primarily due to application chattiness which was aggravated by the effect of network latency. Although the latency between client and database server was 3 to 4 ms, that delay became significant because it was experienced for each of the many application turns.

QuickPredict results further confirmed the analysis, showing that users at the proposed pilot sites will encounter unacceptable response times over the WAN due to expected network latency.

In this case, the best solution is to rewrite the application so that it transfers fewer, larger messages, therefore reducing the effect of latency on the total response time. QuickRecode can be used to virtually recode the application and analyze the impact on the total response time.

Virtually Recode Application and Visualize the Effect of Code Changes

You can use QuickRecode to study the effect that application flow changes have on performance. QuickRecode lets you edit parts of the application including the number of application turns, how much data is transmitted in each turn, and the expected processing time at each tier. Using this approach, you can see the effect of making specific changes to an application without changing the actual code. You can then analyze the new "recoded" application using QuickPredict to determine if the application response time for each pilot site has improved.

QuickRecode is used to minimize application chattiness. This is replicated in the real application by changing the way Oracle accesses the data.

To start, select a group of messages to edit. In this case all the messages between the client and Oracle server are selected and "QuickRecode Selected Items" is applied (Figure 9).

Figure 9.   Use QuickRecode to Minimize Chattiness
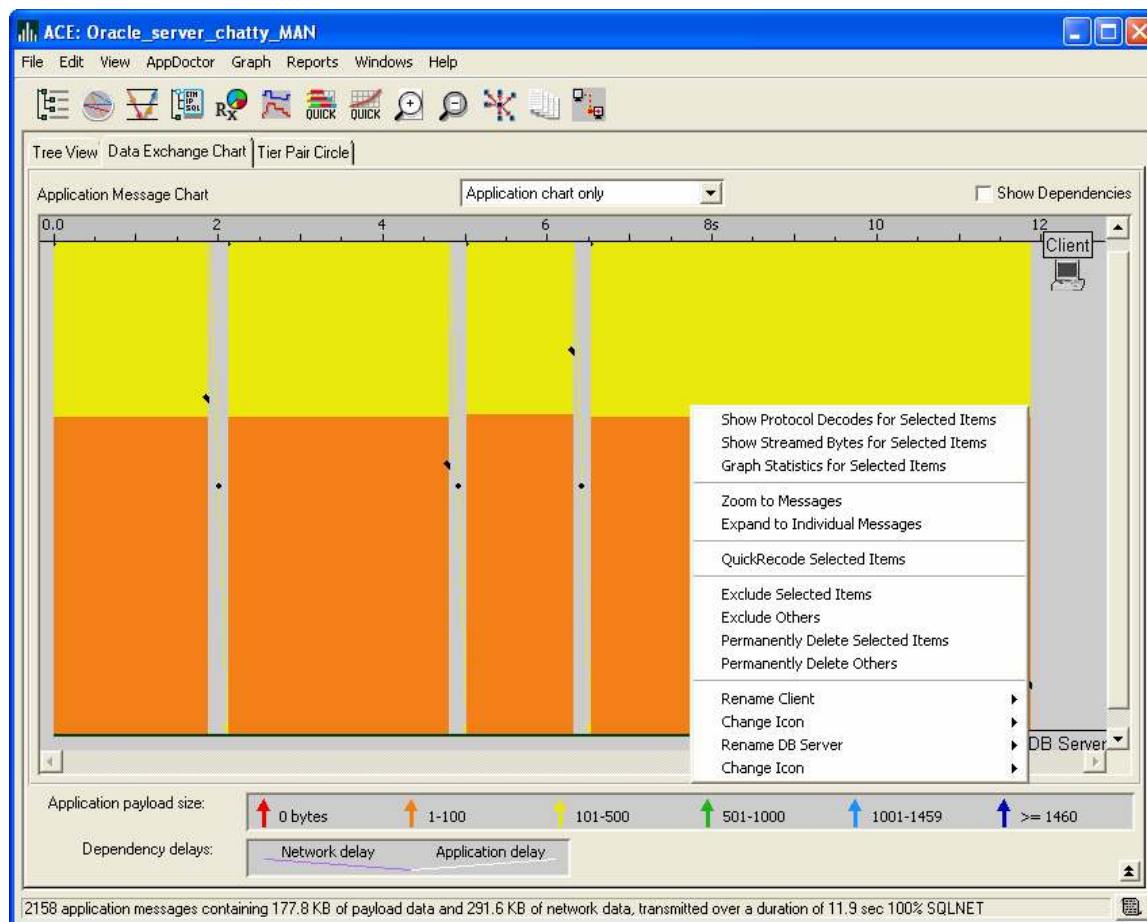
Next, you can modify the behavior of the application. The number of application turns is reduced from 2157 to 200 to simulate a change to the database access (Figure 10).

Figure 10.   Simulate Change in Application Behavior
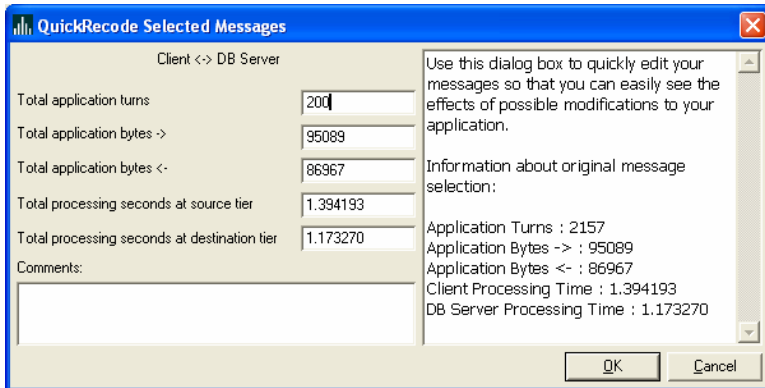
A red band appears around the selected messages in the Data Exchange Chart and the group changes color to indicate that is has been edited (Figure 11).

Figure 11.   Data Exchange Chart Reflects Modified Application
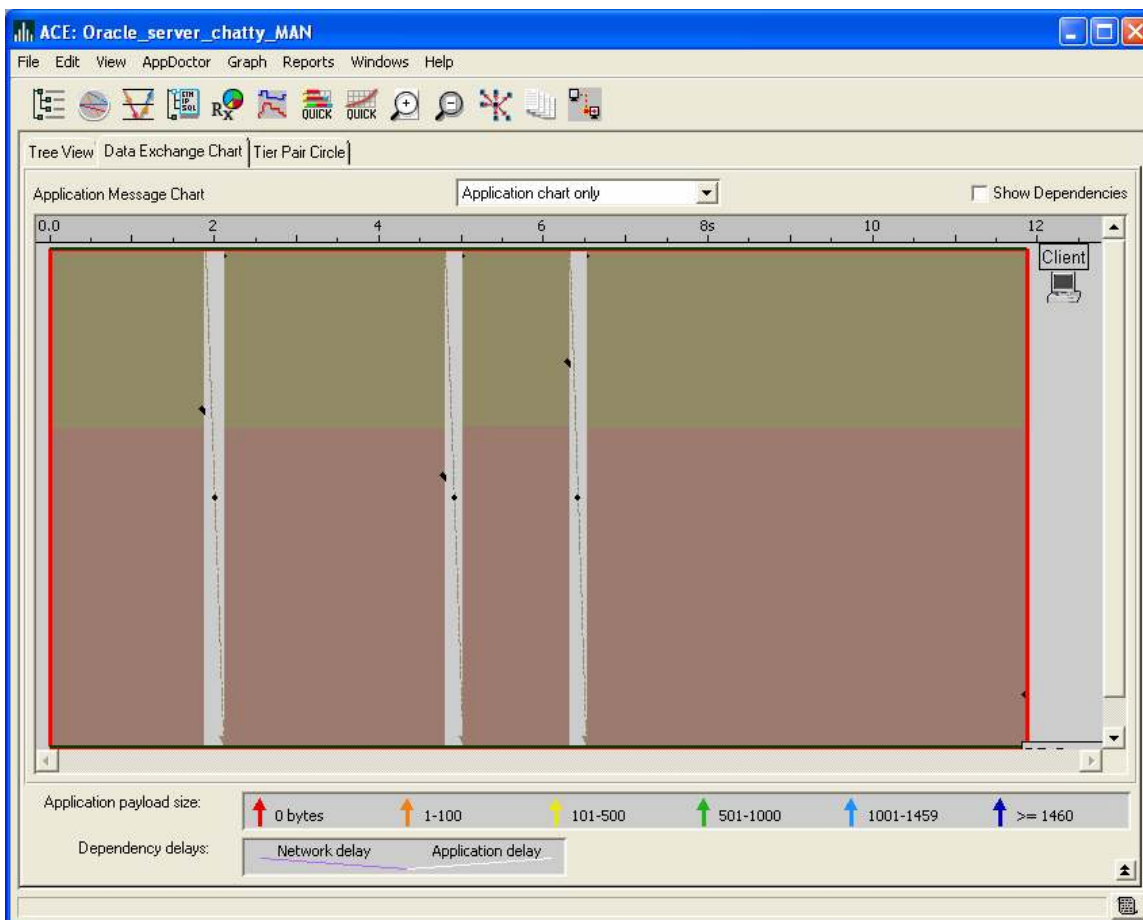
QuickPredict shows how the "recoded" application behaves over different network conditions (Figure 12). The bottom bar chart is the original application without QuickRecode. The top chart represents the recoded or hypothetical modified application. Because the recode reduced the number of application turns, the network latency and overall response times are reduced. The recoded application has a response time of about 3.4 seconds. This data can be used to justify the expense of recoding the application.

Figure 12. QuickPredict Bar Graph Shows Reduced Latency Based on Simulated Application Change
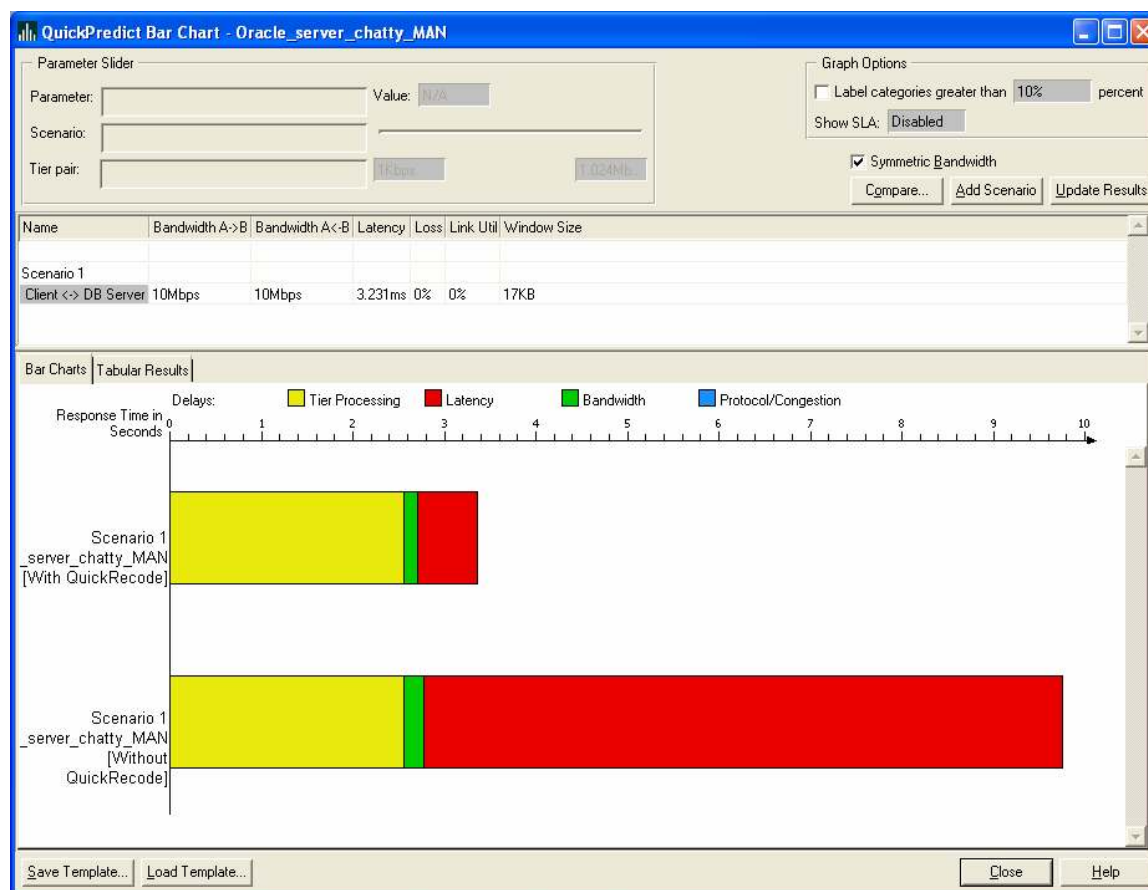
To summarize, the objective of this study was to evaluate the performance of a new application prior to full deployment. Users at the pilot sites were complaining of poor response times. The transaction was captured using the Cisco Network Analysis Module (NAM) blade in the data center and a Cisco AAS capture agent installed on the client desktop. Cisco AAS showed the response time for this specific transaction was approximately 12 seconds. The Summary of Delays chart showed latency was the major contributor of delay; the diagnostic chart showed the application was a bottleneck for chattiness, with too many small requests and responses between the client and database server. QuickRecode was used to show the effects of software recoding, allowing for larger applications messages. The results from QuickRecode showed improved application response time by approximately 6 seconds. This data can be used to justify the expense of recoding the application.

Cisco Application Analysis Solution is a vital tool for diagnosing and solving application performance problems. It can be used to predict application performance before deployment and to troubleshoot problems of production applications. For more information about Cisco AAS, contact your Cisco representative or visit http://www.cisco.com/en/US/products/ps6362/index.html

CISCO SYSTEMS