Examining a Sybase Application

Key Concept—This example shows how AppTransaction Xpert diagnoses and visualizes application and network problems; it is not a step-by-step tutorial. If you have experience with AppTransaction Xpert, you can recreate this study by following the instructions in Recreate the Example. The screen images in this example were captured while running AppTransaction Xpert in Windows with both the AppTransaction Xpert Decode Module (ADM) and the Wireshark external decoder installed. If you are working on Linux, or do not have ADM and Wireshark installed, some screens might look different.

In this network, the client accesses a Sybase application server over a 100 Mbps LAN. This transaction is a download of 15 days of network data that is approximately 1.5 MB. It takes 10 seconds for this transaction to complete.

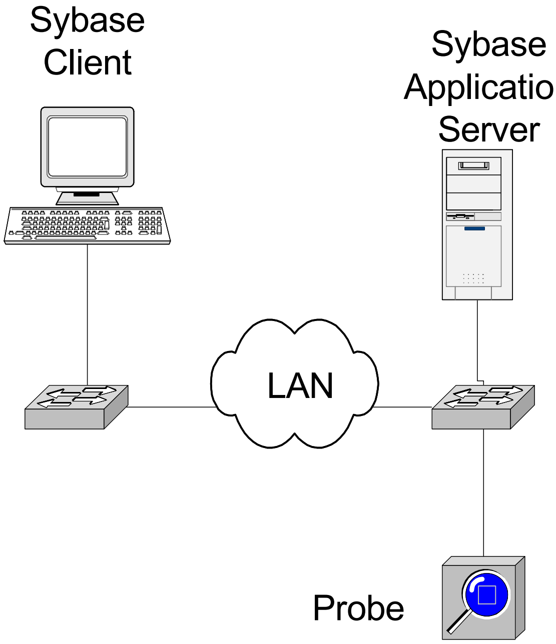
Possible causes for the slow response time fall into three general categories:

- network bottlenecks
- server bottlenecks
- poor application design

This study identifies the cause of the performance problem and recommends possible solutions.

To capture a typical transaction, a probe was placed on the switch to which the application server is connected.

Figure 9-1 Network Diagram

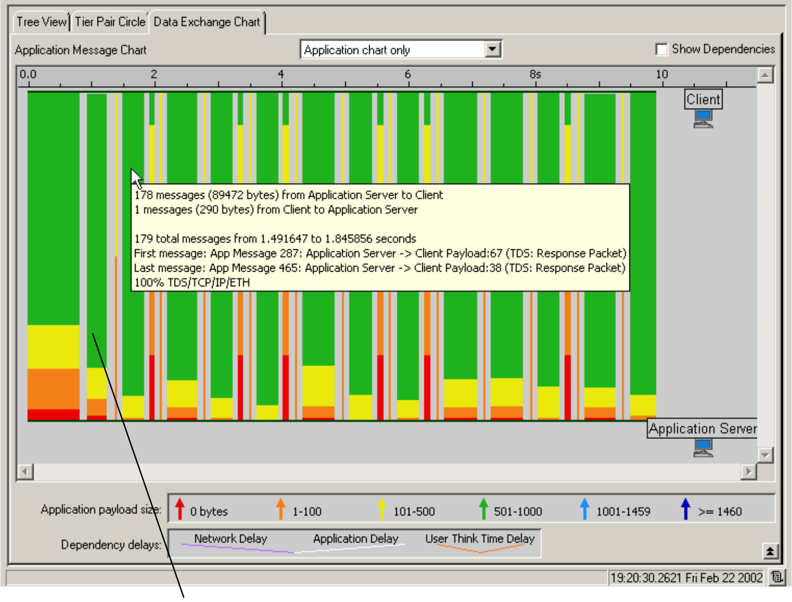


A packet trace capture was taken while the client executed a download from the application server. This capture was then opened in AppTransaction Xpert.

Diagnosis

The Data Exchange Chart (DEC) shows the data transferred between tiers on a time line. The colors of the application messages represent the size of the messages. Each group’s color represents a histogram of message sizes. For this application, this chart tells us that the response time is about 10 seconds and that the majority of the data being transferred is between 501-1000 bytes (green).

Figure 9-2 Data Exchange Chart (Application View)



178 messages (89472 bytes) from Application Server to Client
1 messages (290 bytes) from Client to Application Server

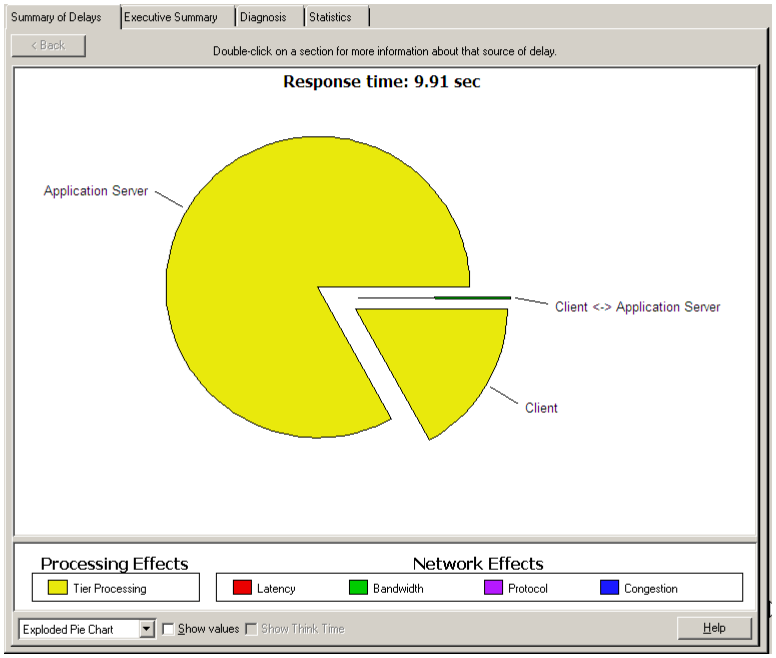
179 total messages from 1.491647 to 1.845856 seconds
First message: App Message 287: Application Server -> Client Payload:67 (TDS: Response Packet)
Last message: App Message 465: Application Server -> Client Payload:38 (TDS: Response Packet)
100% TDS/TCP/IP/ETH

Colored bars represent a histogram of message sizes

AppDoctor’s Summary of Delays is used to identify the factors that contribute to delay. The chart shows that a majority of the delay is caused by processing on the application server.

It is interesting to note that the network-related delays are minimal compared to the application delays.

Figure 9-3 AppDoctor Summary of Delays

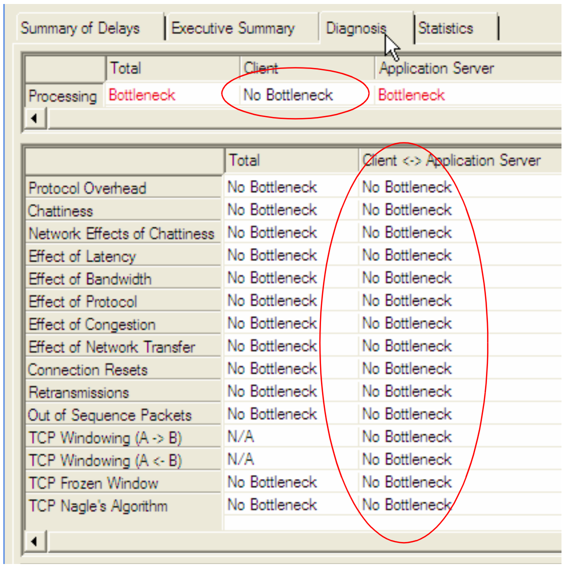


Double-click on a section for more information about that source of delay.

AppDoctor's Diagnosis provides a more granular analysis of the application. Diagnosis identifies bottlenecks and potential bottlenecks for many common causes of application performance problems.

From the Diagnosis, we can see a problem with the application server. Diagnosis confirms that the primary bottleneck is due to processing delay on the application server. Processing delay may be due to CPU load, excessive processing demands, disk access, or other hardware-related problems. It is important to note there are no *network*-related bottlenecks.

Figure 9-4 Diagnosis for the Sybase Transaction

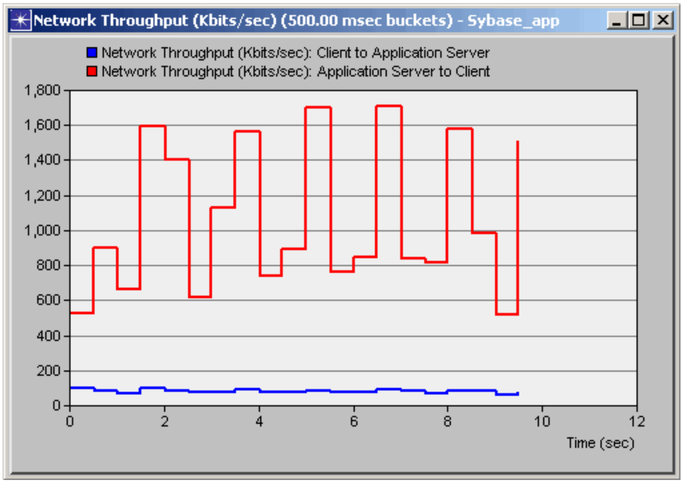


	Total	Client	Application Server
Processing	Bottleneck	No Bottleneck	Bottleneck

	Total	Client <-> Application Server
Protocol Overhead	No Bottleneck	No Bottleneck
Chattiness	No Bottleneck	No Bottleneck
Network Effects of Chattiness	No Bottleneck	No Bottleneck
Effect of Latency	No Bottleneck	No Bottleneck
Effect of Bandwidth	No Bottleneck	No Bottleneck
Effect of Protocol	No Bottleneck	No Bottleneck
Effect of Congestion	No Bottleneck	No Bottleneck
Effect of Network Transfer	No Bottleneck	No Bottleneck
Connection Resets	No Bottleneck	No Bottleneck
Retransmissions	No Bottleneck	No Bottleneck
Out of Sequence Packets	No Bottleneck	No Bottleneck
TCP Windowing (A -> B)	N/A	No Bottleneck
TCP Windowing (A <- B)	N/A	No Bottleneck
TCP Frozen Window	No Bottleneck	No Bottleneck
TCP Nagle's Algorithm	No Bottleneck	No Bottleneck

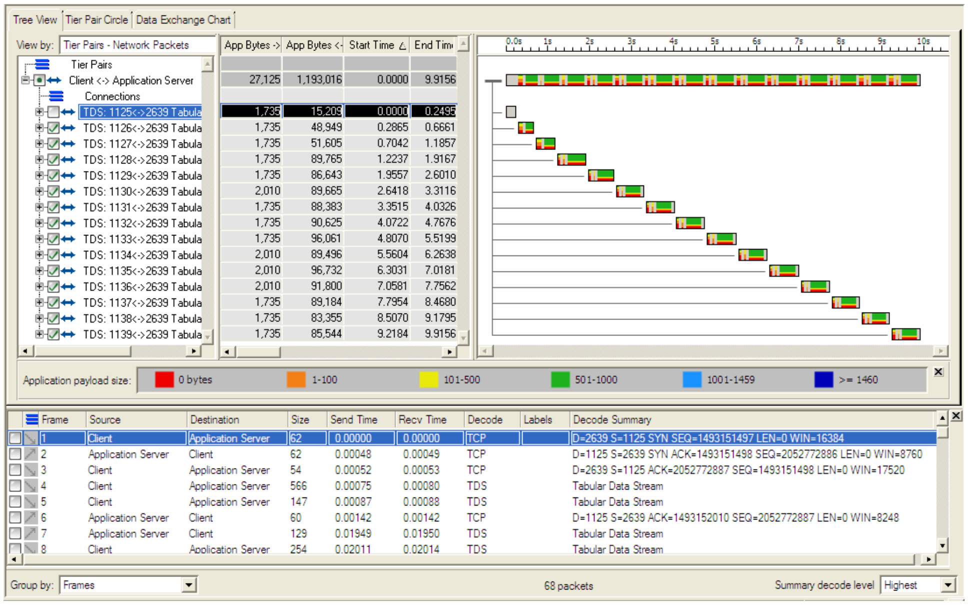
Graphing statistics can help us understand the application's behavior and its interaction with the network. In the following graph, we can see that the network throughput during this download averages approximately 1 Mbps, a small percentage of the total bandwidth.

Figure 9-5 Network Throughput Graph



In addition to the Diagnosis performed above, AppTransaction Xpert represents the transaction in terms of its connections. The associated statistics for each connection and a graphical representation in a Gantt chart can be seen in the Tree View.

Figure 9-6 Tree View of the Sybase Application



View by: Tier Pairs - Network Packets

	App Bytes ->	App Bytes <-	Start Time	End Time
Tier Pairs				
Client <-> Application Server	27,125	1,193,016	0.0000	9.9156
Connections				
TDS: 1125<->2639 Tabula	1,735	15,209	0.0000	0.2495
TDS: 1126<->2639 Tabula	1,735	48,949	0.2865	0.6661
TDS: 1127<->2639 Tabula	1,735	51,605	0.7042	1.1857
TDS: 1128<->2639 Tabula	1,735	89,765	1.2237	1.9167
TDS: 1129<->2639 Tabula	1,735	86,643	1.9557	2.6010
TDS: 1130<->2639 Tabula	2,010	89,665	2.6418	3.3116
TDS: 1131<->2639 Tabula	1,735	88,383	3.3515	4.0326
TDS: 1132<->2639 Tabula	1,735	90,625	4.0722	4.7676
TDS: 1133<->2639 Tabula	1,735	96,061	4.8070	5.5199
TDS: 1134<->2639 Tabula	2,010	89,496	5.5604	6.2638
TDS: 1135<->2639 Tabula	2,010	96,732	6.3031	7.0181
TDS: 1136<->2639 Tabula	2,010	91,800	7.0581	7.7562
TDS: 1137<->2639 Tabula	1,735	89,184	7.7954	8.4680
TDS: 1138<->2639 Tabula	1,735	83,355	8.5070	9.1795
TDS: 1139<->2639 Tabula	1,735	85,544	9.2184	9.9156

Application payload size: 🟥 0 bytes 🟧 1-100 🟨 101-500 🟩 501-1000 🟦 1001-1459 🟦 >= 1460

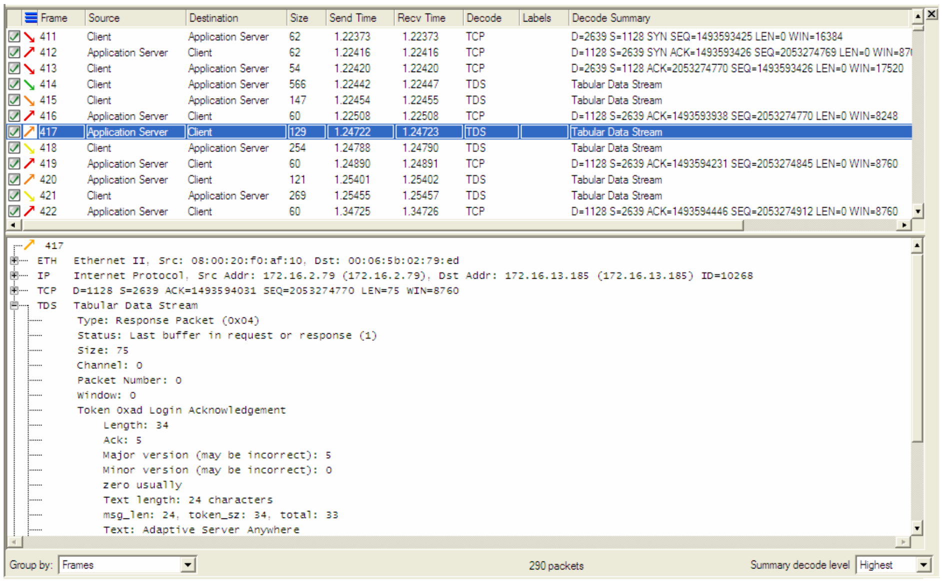
Frame	Source	Destination	Size	Send Time	Recv Time	Decode	Labels	Decode Summary
1	Client	Application Server	62	0.00000	0.00000	TCP		D=2639 S=1125 SYN SEQ=1493151497 LEN=0 WIN=16384
2	Application Server	Client	62	0.00048	0.00049	TCP		D=1125 S=2639 SYN ACK=1493151498 SEQ=2052772886 LEN=0 WIN=8760
3	Client	Application Server	54	0.00052	0.00053	TCP		D=2639 S=1125 ACK=2052772887 SEQ=1493151498 LEN=0 WIN=17520
4	Client	Application Server	566	0.00075	0.00080	TDS		Tabular Data Stream
5	Client	Application Server	147	0.00087	0.00088	TDS		Tabular Data Stream
6	Application Server	Client	60	0.00142	0.00142	TCP		D=1125 S=2639 ACK=1493152010 SEQ=2052772887 LEN=0 WIN=8248
7	Application Server	Client	129	0.01949	0.01950	TDS		Tabular Data Stream
8	Client	Application Server	254	0.02011	0.02014	TDS		Tabular Data Stream

Group by: Frames 68 packets Summary decode level Highest

In this study, the client requested 15 days of data. For each connection, the associated statistics and Gantt chart are available. The Gantt chart illustrates the duration of and activity within each download in relation to the total transaction time. Most of the data downloads take about 0.7 seconds.

By expanding the connections, we can view more detailed information about the frames in the transaction. AppTransaction Xpert decodes the frames, showing the protocol commands that are being executed.

Figure 9-7 Expanded Protocol View of Fourth Connection



Frame	Source	Destination	Size	Send Time	Recv Time	Decode	Labels	Decode Summary
☑ 411	Client	Application Server	62	1.22373	1.22373	TCP		D=2639 S=1128 SYN SEQ=1493593425 LEN=0 WIN=16384
☑ 412	Application Server	Client	62	1.22416	1.22416	TCP		D=1128 S=2639 SYN ACK=1493593426 SEQ=2053274769 LEN=0 WIN=87
☑ 413	Client	Application Server	54	1.22420	1.22420	TCP		D=2639 S=1128 ACK=2053274770 SEQ=1493593426 LEN=0 WIN=17520
☑ 414	Client	Application Server	566	1.22442	1.22447	TDS		Tabular Data Stream
☑ 415	Client	Application Server	147	1.22454	1.22455	TDS		Tabular Data Stream
☑ 416	Application Server	Client	60	1.22508	1.22508	TCP		D=1128 S=2639 ACK=1493593938 SEQ=2053274770 LEN=0 WIN=8248
☑ 417	Application Server	Client	129	1.24722	1.24723	TDS		Tabular Data Stream
☑ 418	Client	Application Server	254	1.24788	1.24790	TDS		Tabular Data Stream
☑ 419	Application Server	Client	60	1.24890	1.24891	TCP		D=1128 S=2639 ACK=1493594231 SEQ=2053274845 LEN=0 WIN=8760
☑ 420	Application Server	Client	121	1.25401	1.25402	TDS		Tabular Data Stream
☑ 421	Client	Application Server	269	1.25455	1.25457	TDS		Tabular Data Stream
☑ 422	Application Server	Client	60	1.34725	1.34726	TCP		D=1128 S=2639 ACK=1493594446 SEQ=2053274912 LEN=0 WIN=8760

```
417
ETH   Ethernet II, Src: 08:00:20:f0:af:10, Dst: 00:06:5b:02:79:ed
IP    Internet Protocol, Src Addr: 172.16.2.79 (172.16.2.79), Dst Addr: 172.16.13.185 (172.16.13.185) ID=10268
TCP   D=1128 S=2639 ACK=1493594031 SEQ=2053274770 LEN=75 WIN=8760
TDS   Tabular Data Stream
       Type: Response Packet (0x04)
       Status: Last buffer in request or response (1)
       Size: 75
       Channel: 0
       Packet Number: 0
       Window: 0
       Token 0xad Login Acknowledgement
           Length: 34
           Ack: 5
           Major version (may be incorrect): 5
           Minor version (may be incorrect): 0
           zero usually
           Text length: 24 characters
           msg_len: 24, token_sz: 34, total: 33
           Text: Adaptive Server Anywhere
```

Group by: Frames — 290 packets — Summary decode level: Highest

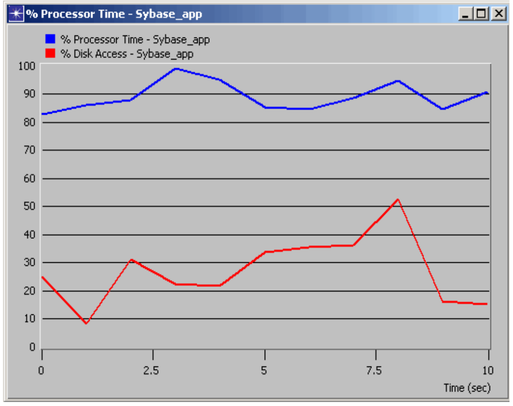
This transaction performs the 15-day data transfer sequentially: the server responds to the client's request by sending one day of data at a time. Each day of data is carried in a separate connection.

Analysis of this transaction determined that the network was not a factor in the delay. The application server is the cause of the Sybase application's poor performance. AppTransaction Xpert diagnosed the bottleneck for this application transaction as processing delay on the application server.

Conclusion

There are several ways to decrease the application response time. One solution is to upgrade the CPU processor on the server. The following graph shows the performance meter on the application server during the download. The CPU bar indicates that there is a heavy load on the server. Upgrading the processor should decrease the load and the processing delay.

Figure 9-8 Snapshot of Performance Meter on Server During the Transaction



Another solution is to have application developers rewrite the application to download the data in parallel, not sequentially. If the transaction were executed in parallel, the total transaction time might decrease.

Using AppTransaction Xpert, you can troubleshoot poor application performance and pinpoint the potential causes.

Recreate the Example

From the AppTransaction Xpert System window, choose **File > Open Model > Transaction Analyzer…** and select **Sybase_app** (in *<reldir>*\sys\examples\AppTransaction Xpert\examples) or perform the following procedure.

Procedure 9-1   Recreate the Example

1 Open the **Sybase_app.enc** packet trace in AppTransaction Xpert.
(**File** > **Open Packet Trace(s)** > **In Transaction Analyzer (Single Capture)…**)

2 Rename the tiers as follows:
(In the DEC, right-click on the tier icon and choose **Rename** > **Edit…**)

- Rename 172.16.13.185 to Client
- Rename 172.16.2.79 to Application Server

3 Set the bandwidth to 100,000 Kbps between tiers.
(**AppDoctor** > **Refine Network Effects**…)

4 When creating the Application Throughput graph, use a bucket width of 500 ms.

End of Procedure 9-1