IBM Software Group

Understanding The Impact Of The Network On z/OS Performance

Ed Woods - IBM Corporation

Session 12780

Tuesday, February 5th: 4:30 PM - 5:30 PM

Tivoli software

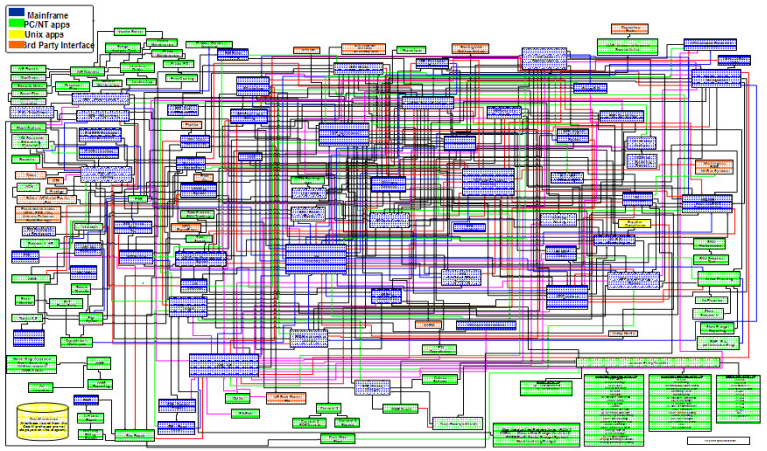
Agenda

- Introduction
- Looking at the application time line
- Examples of mainframe/network interaction
  - Analysis scenarios using commonly available commands
  - Optimization considerations for various subsystems
- Defining a consistent monitoring strategy

The Challenges Of Performance And Availability Management Of Complex Systems

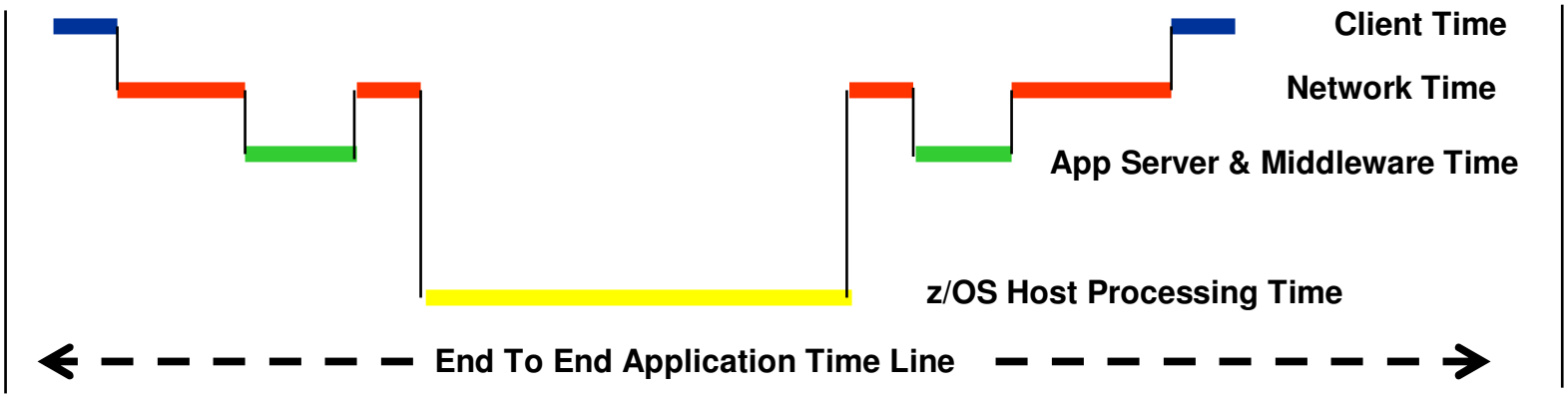
- Most new applications are composite by design
  - Applications cross multiple subsystems and platforms
  - Integration and utilization of multiple core technologies
  - Pose challenges from a management and monitoring perspective

- Common Technical Challenges
  - Multiple platforms
  - Potentially multiple DB systems
  - Middleware considerations
  - One or multiple network hops



- *Is the problem the network, the host, the DB, the client, or somewhere in between?*

The Network And The Application Time Line

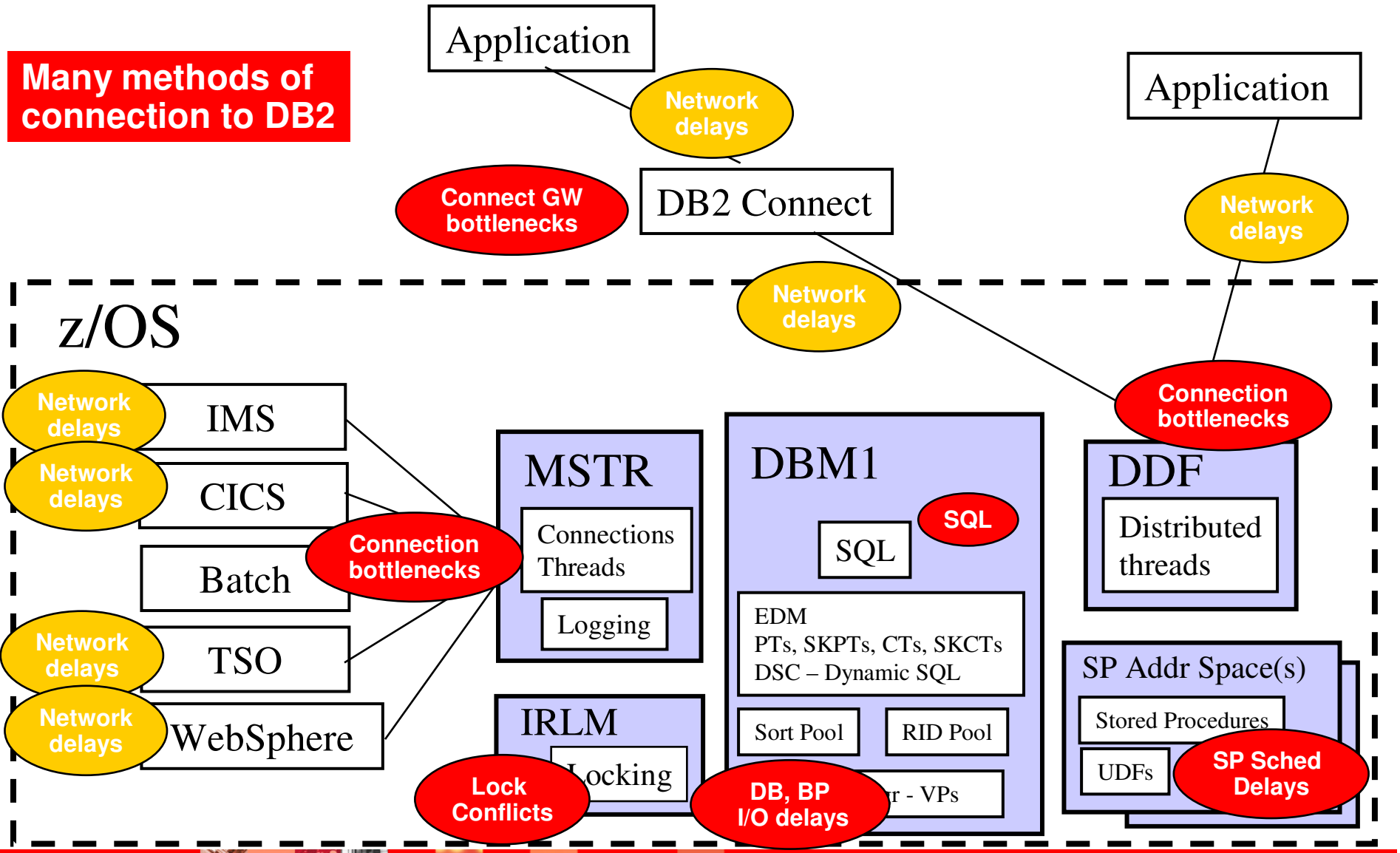


- Portions of response time may reside in any of the following
  - End user client processing, the application server or middleware level, the database, or other aspects of host z/OS application processing
  - Potential for bottlenecks at multiple points
- The network will impact the overall application time line
  - Time is required to send messages across the network
  - Overhead processing, including communication subsystem session management
  - Network hardware, traffic, connections, connection pools

The Impact Of The Network On Critical z/OS Components

- The network has impact on z/OS workload in many ways
- Speed of the network – network congestion and bottlenecks
- Each z/OS application or component subsystem has unique network considerations
  - IMS, DB2, CICS, MQSeries, WebSphere, FTP
- Keep in mind that z/OS application/subsystem configuration and logic may also impact the network
  - Subsystem configuration options and settings impact network interaction
  - Application logic impacts network usage and performance
  - Be aware of the potential impact of SSL and IPSec

DB2 Has Several Potential Performance Bottlenecks



Network Analysis Tools
NETSTAT Command

- NETSTAT reports TCP/IP connections and protocol statistics
- Get status information on connections and statistics on packets sent, packets received, fragmentation, etc…..

```
C:\Documents and Settings\woodse>netstat

Active Connections

  Proto  Local Address          Foreign Address        State
  TCP    IBM-1E47754C52F:4138   demomvs.demopkg.ibm.com:telnet  ESTABLISHED
  TCP    IBM-1E47754C52F:4251   d01ml253.pok.ibm.com:1352  ESTABLISHED
  TCP    IBM-1E47754C52F:4255   demomvs.demopkg.ibm.com:448  ESTABLISHED
  TCP    IBM-1E47754C52F:1035   localhost:1036         ESTABLISHED
  TCP    IBM-1E47754C52F:1036   localhost:1035         ESTABLISHED
  TCP    IBM-1E47754C52F:1920   localhost:3416         ESTABLISHED
  TCP    IBM-1E47754C52F:1920   localhost:3768         ESTABLISHED
  TCP    IBM-1E47754C52F:3416   localhost:1920         ESTABLISHED
  TCP    IBM-1E47754C52F:3417   localhost:3661         ESTABLISHED
  TCP    IBM-1E47754C52F:3661   localhost:3417         ESTABLISHED
  TCP    IBM-1E47754C52F:3661   localhost:3769         ESTABLISHED
  TCP    IBM-1E47754C52F:3768   localhost:1920         ESTABLISHED
  TCP    IBM-1E47754C52F:3769   localhost:3661         ESTABLISHED
  TCP    IBM-1E47754C52F:1097   204.146.166.107:http   CLOSE_WAIT
  TCP    IBM-1E47754C52F:1098   129.42.208.236:https   ESTABLISHED
  TCP    IBM-1E47754C52F:1100   rarcol01.attglobal.net:http  CLOSE_WAIT
  TCP    IBM-1E47754C52F:1187   www.live365.com:http   CLOSE_WAIT
  TCP    IBM-1E47754C52F:1188   ss32.live365.com:http  ESTABLISHED
  TCP    IBM-1E47754C52F:4204   58.mtl-mg05.streamtheworld.net:http  ESTABLISHED
```

NETSTAT command issued from client perspective.

Connection to DB2 on z/OS

NETSTAT Connection Detail

netstat all (port 448)

```
EZZ2350I MVS TCP/IP NETSTAT CS V1R10       TCPIP Name: TCPIP           19:16:24
EZZ2550I Client Name: DSNCDIST                 Client Id: 0000C90E
EZZ2551I Local Socket: 9.39.68.147..448        Foreign Socket: 9.65.73.27..4255

EZZ2552I   Last Touched:        19:14:58          State:               Establsh
EZZ2577I   BytesIn:            0000006973        BytesOut:            0008457981
EZZ2574I   SegmentsIn:         0000003423        SegmentsOut:         0000006614
EZZ2553I   RcvNxt:             3808791478        SndNxt:              2538223807
EZZ2554I   ClientRcvNxt:       3808791478        ClientSndNxt:        2538223807
EZZ2555I   InitRcvSeqNum:      3808784504        InitSndSeqNum:       2529765825
EZZ2556I   CongestionWindow:   0000017349        SlowStartThreshold:  0000002620
EZZ2557I   IncomingWindowNum:  3808824236        OutgoingWindowNum:   2538289289
EZZ2558I   SndWl1:             3808791478        SndWl2:              2538223807
EZZ2559I   SndWnd:            0000065482        MaxSndWnd:           0000131070
EZZ2560I   SndUna:            2538223807        rtt_seq:             2538223753
EZZ2561I   MaximumSegmentSize: 0000001310       DSField:             00
EZZ2563I   Round-trip information:
EZZ2564I     Smooth trip time: 184.000          SmoothTripVariance:  84.000
EZZ2565I   ReXmt:             0000000002        ReXmtCount:          0000000000
EZZ2572I   DupACKs:           0000000284        RcvWnd:              0000032758
EZZ2566I   SockOpt:           8D                TcpTimer:            00
EZZ2567I   TcpSig:            04                TcpSel:              40
EZZ2568I   TcpDet:            EC                TcpPol:              00
EZZ2537I   QOSPolicy:         No
EZZ2542I   RoutingPolicy:     No
EZZ2570I   ReceiveBufferSize: 0000016384        SendBufferSize:      0000065536
EZZ2538I   ReceiveDataQueued: 0000000000
EZZ2539I   SendDataQueued:    0000000000
```

Byte counts

Network segment counts

Network response time info

Retransmission count

An Example - Looking At The Numbers
DB2 Distributed Performance
Statistics Trace Data For The DB2 Subsystem

```
>                    DISTRIBUTED DATA FACILITY STATISTICS
 DFST
+ Collection Interval:  REALTIME                          Start:  06/04 13:40:46
+ Report Interval:       4 sec                              End:  06/04 13:40:49
+
                                 DCDB203    DDF Status                =   ACTIVE
                                  .0%       Dist Allied Threads       =        0
+ Active DBATs                 =        3   Inactive DBATs            =        0
+ DDF Send Rate                =   0K/sec   DDF Receive Rate          =   0K/sec
+ Resync Attempts              =        0   Resync Successes          =        0
+ Cold Start Connections       =        0   Warm Start Connections    =        0
+ DBAT Queued                  =        0   Conversations Dealloc     =        0
+ HWM All DBATs                =        5   HWM Active DBATs          =        5
+ Max DB Access (MAXDBAT)      =      500   HWM Inactive DBATs        =        0

                                 RDA REMOTE LOCS
+
+ Conversations Queued          =        0   Binds for Remote Access  =        0
+ Message Buffer Rows           =   174874   Block Mode Switches       =        0
+ Commits/Remote                =        0   Rollbacks/Remote          =        0
+ Indoubts/Remote               =        0
+
+          Tran      SQL      Row Message      Byte  Commit    Abort    Conv   Blocks
+        ------- ------- ------- ------- --------- ------- ------- ------- -------
+ Sent         0       0  174927    2486  43164569       0       0       0    1746
+ Recv        55    1714       0    2285    238429     363       8      55       0
+
```

Thread creation queues?

HWM of DBAT usage

Is buffering occurring?

Recv – received into DB2
Sent – out to client/apps

Bytes and messages sent and received

Looking At The Application
DB2 Accounting Information Analysis

```
>                          DISTRIBUTED THREAD DETAIL
 PLAN
+ Thread:  Plan=DISTSERV  Connid=SERVER   Corrid=db2bp.exe    Authid=DNET581
+ Dist   :  Type=DATABASE ACCESS, Luwid=G941491B.PC10.090604182432=169
+ Location  :  9.65.73.27
 rsum
+                          Distributed TCP/IP Data
+Location          IP Addr  Port Ctbuser  Srvclsnam  Prod ID  Workstation Name
+----------------- -------- ---- -------- ---------- -------- ------------------
+9.65.73.27        0941491B  448 dnet581  NT         SQL09013 IBM-1E47754C52F
+
+ Transaction name: db2bp.exe
+ TCP/IP Userid:    dnet581
+
+                                    Statistics
+
+ Remote Loc                         Remote Location Luname =
+
+ Protocol Used          =                Conversations Queued    =            0
+ Block Mode Switches    =              0  Message Buffer Rows     =        33280
+ Bind Remote Access     =                 Max Allocated Conv      =            0
+ Conv Allocated         =              0  Conv Deallocated        =            0
+ Indoubt/Remote         =              0  Commit/Remote           =            0
+ Rollback/Remote        =              0
+
+         Tran     SQL      Row Message      Byte  Commit   Abort    Conv  Blocks
+       ------- ------- ------- ------- -------- ------- ------- ------- -------
+ Sent       0       0   33280     270  8457981       0       0       0     260
+ Recv       1     264       0     270    26329       2       0       1       0
+
+                                    ast 2-Phase                    Commit Backo t
+ 2-PHASE COMMIT         Prepare    Agent   Commit Backout  Forget    Resp    Re p
+                        ------- ------- -------- ------- ------- ------- ------
+             Sent             0       0        0       0       0       0      0
+             Recv             0       0        0       0       0       0      0
================================================================================
```

Note the quantity of data being sent

Note row versus message/block counts to determine blocking

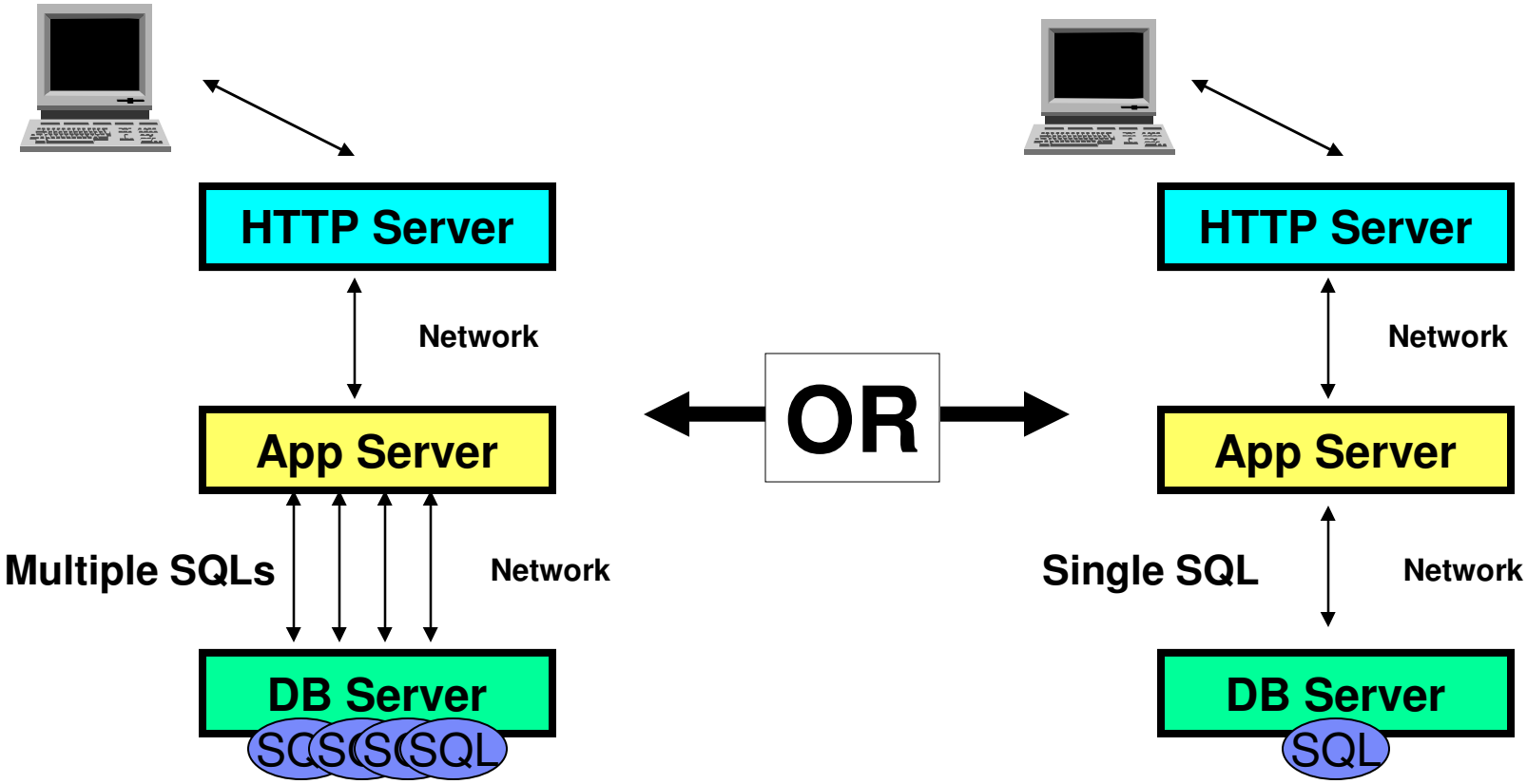
Recv – received into DB2
Sent – out to client/app

 # SQL calls

# of messages sent

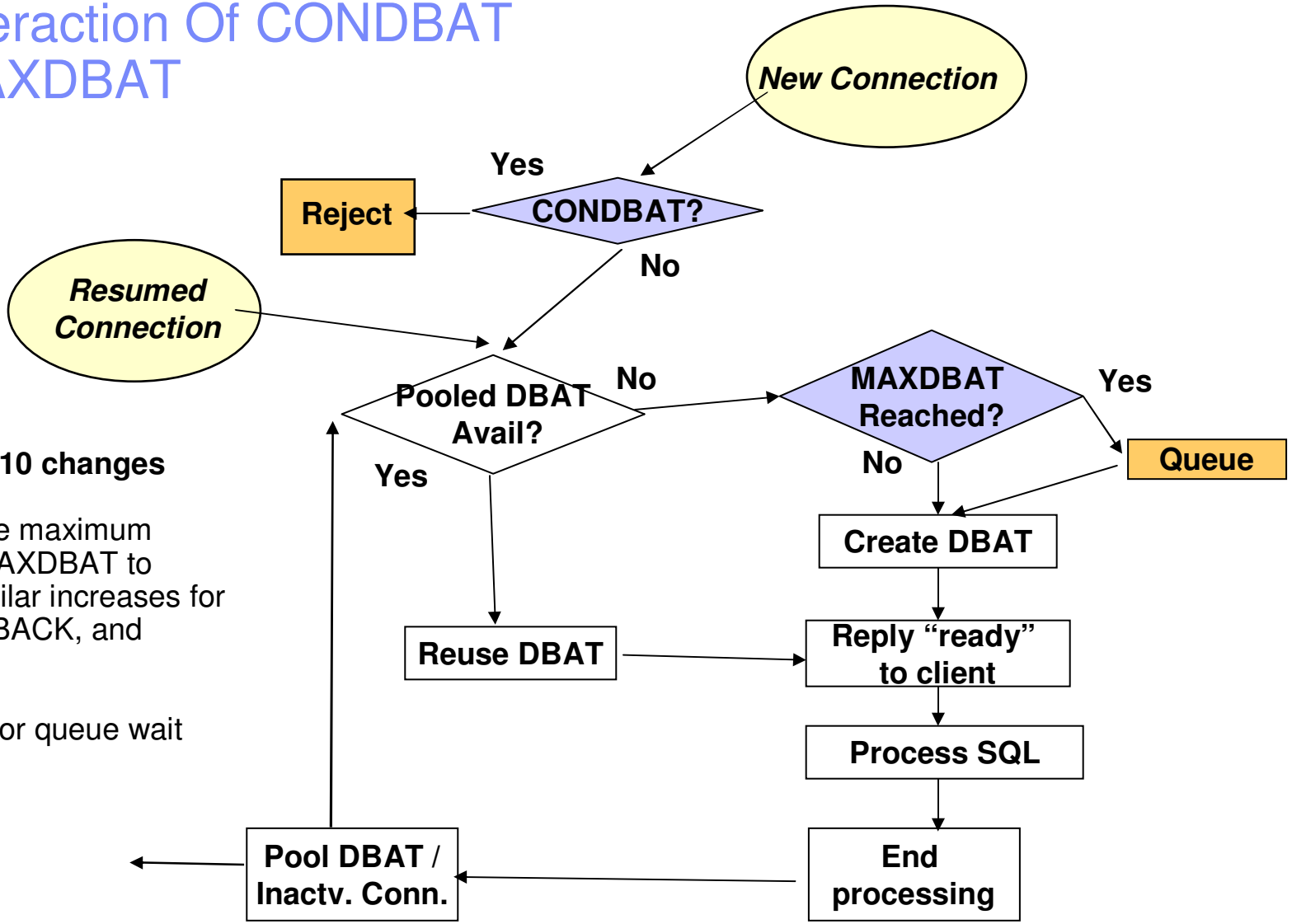
DRDA blocks for queries

Example - Optimize DB2 Applications To Minimize Network Traffic



- Do more with SQL to eliminate redundant back/forth activity
- Crossing more layers will mean more overhead
- Don't put too much business logic in the DB layer

DB2 Subsystem Configuration The Interaction Of CONDBAT And MAXDBAT

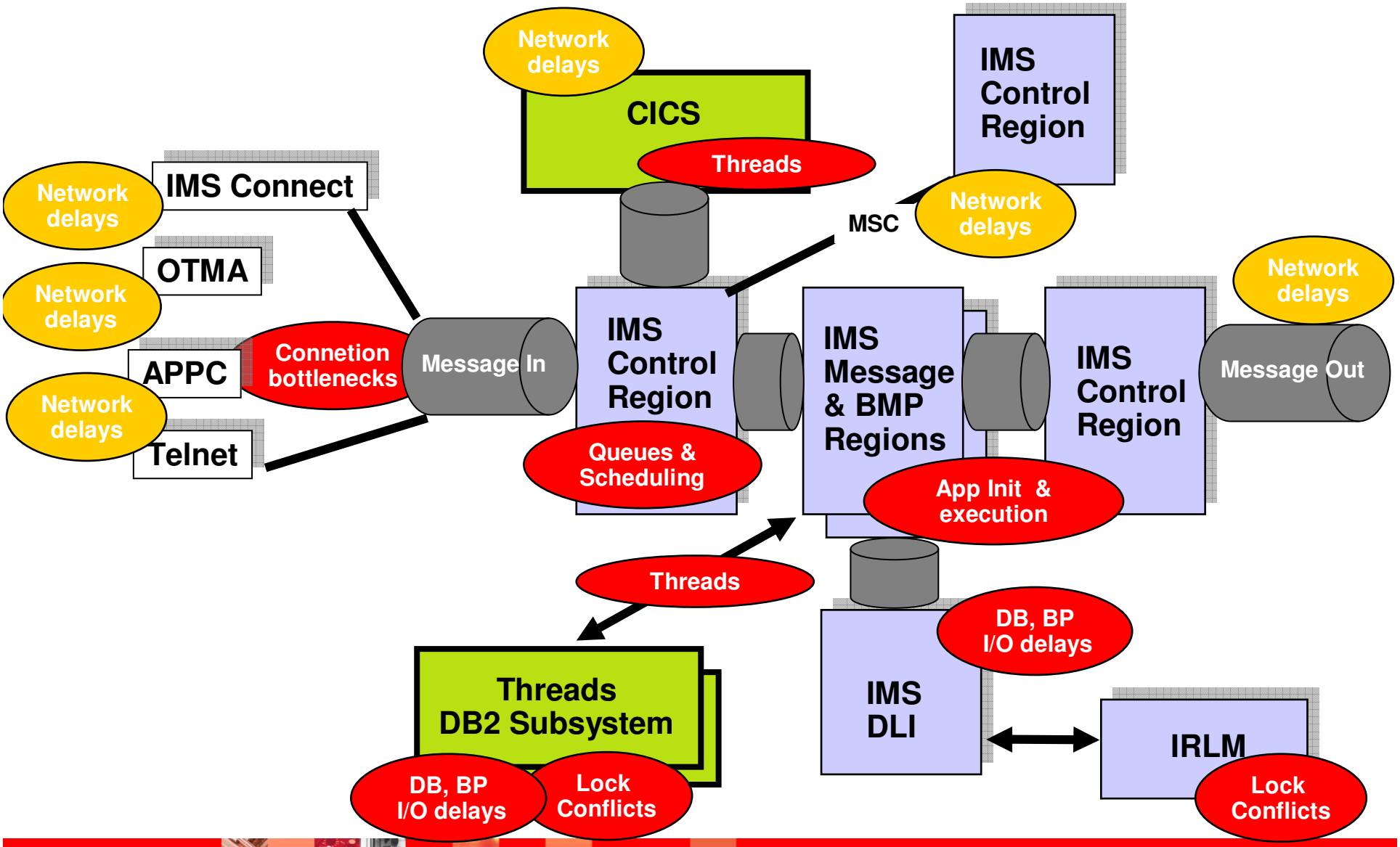


Note – DB2 10 changes

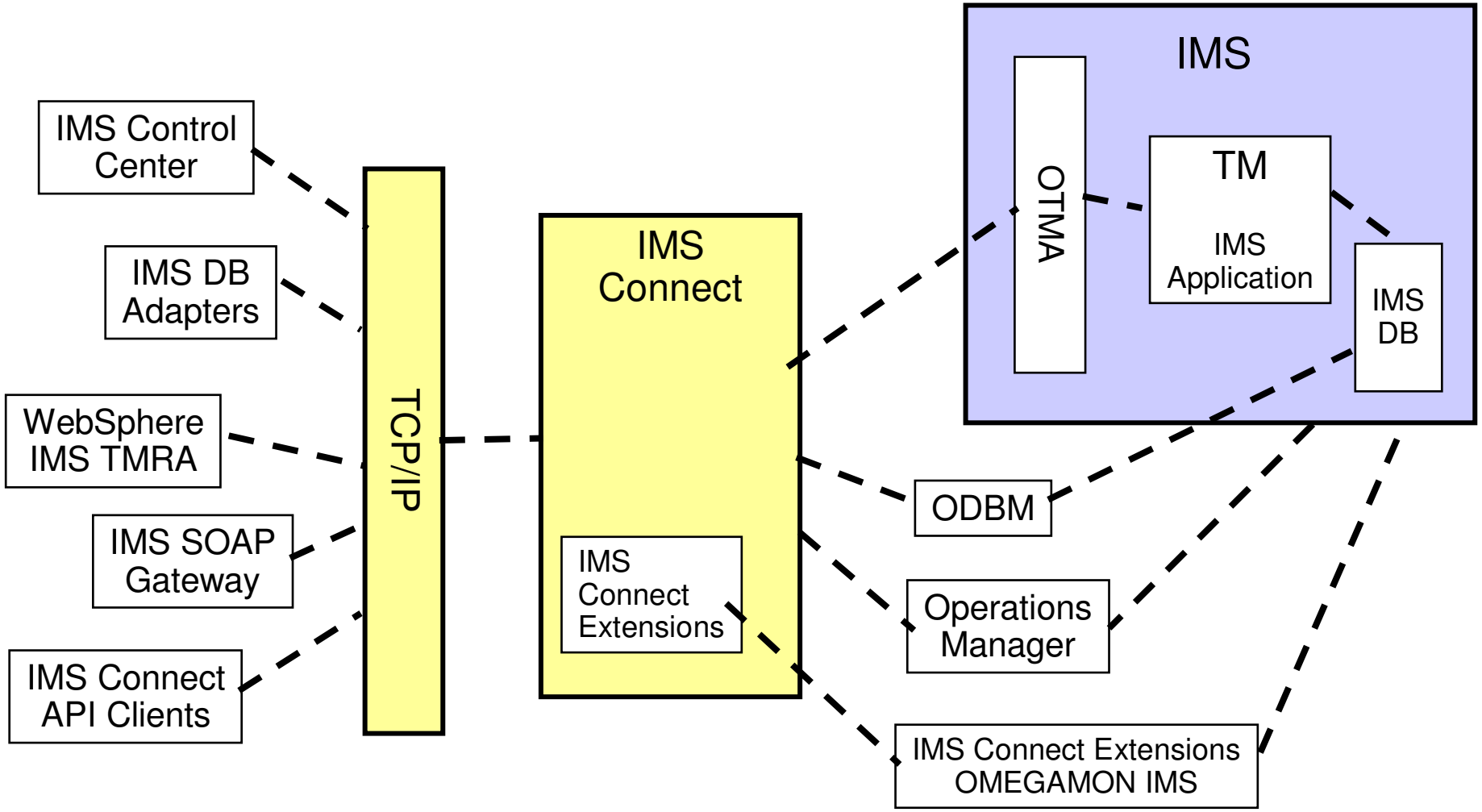
Increases the maximum setting for MAXDBAT to 19999. (Similar increases for IDFORE, IDBACK, and CTHREAD)

New zparm for queue wait time

IMS Has Many Potential Bottlenecks (Including Network)



IMS Connect And The Interaction With TCP/IP



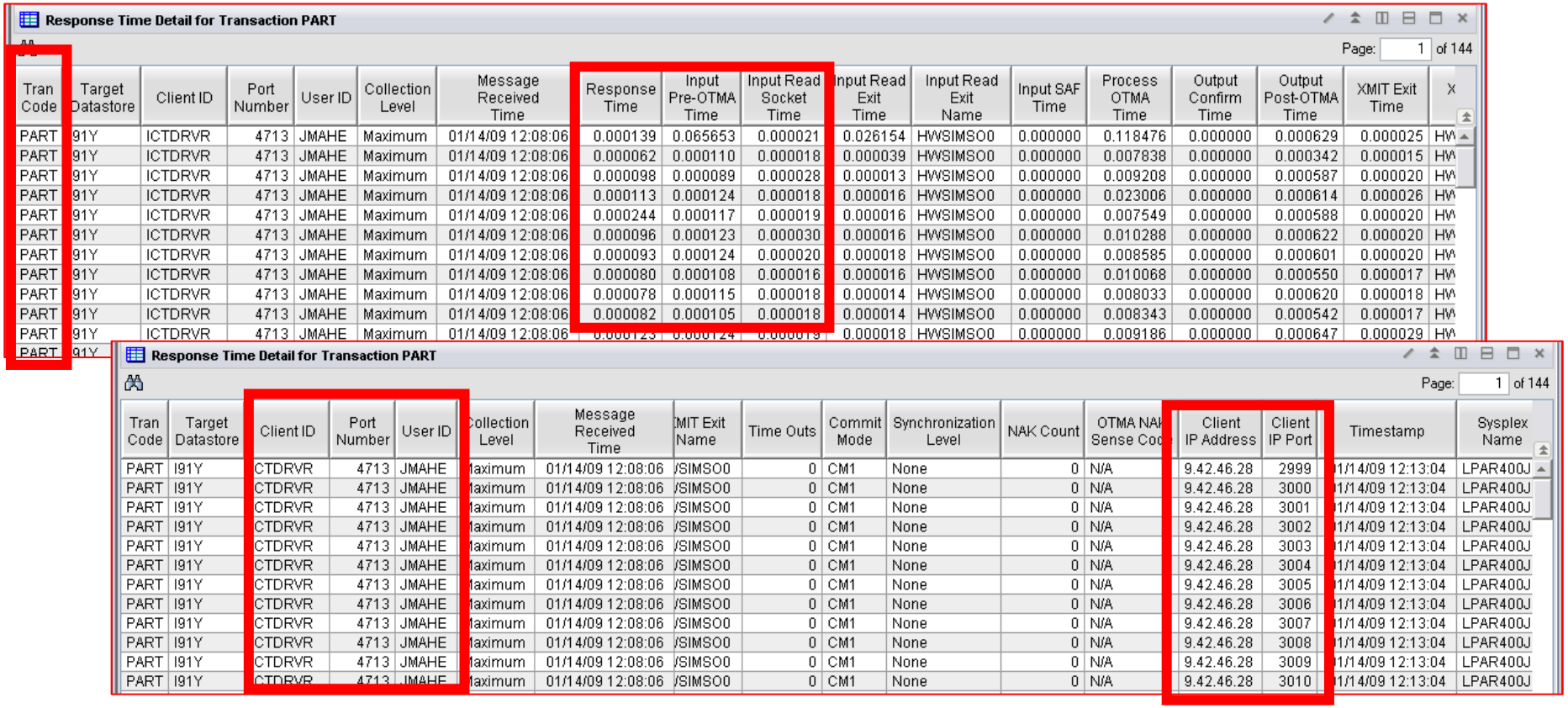
IMS Connect
Types Of Connections & Message Considerations

- IMS Connect – types of connections
  - Non-Persistent socket
    - Closes after each send to the ICON client
  - Transaction socket
    - Close after each transaction or conversation - The default
    - Has connect/disconnect overhead for each message
  - Persistent socket
    - Multiple read/writes for multiple transactions
    - Typically more efficient
    - Will keep the socket open – make sure you have enough sockets
- Message considerations
  - General ROT – use one send for the entire message
  - If doing multiple writes then specify NODELAYACK on PORT statement in z/OS
    - If not specified then may wait up to 300ms for each transmission

IMS Network
Examples Of Relevant Options And Parameters

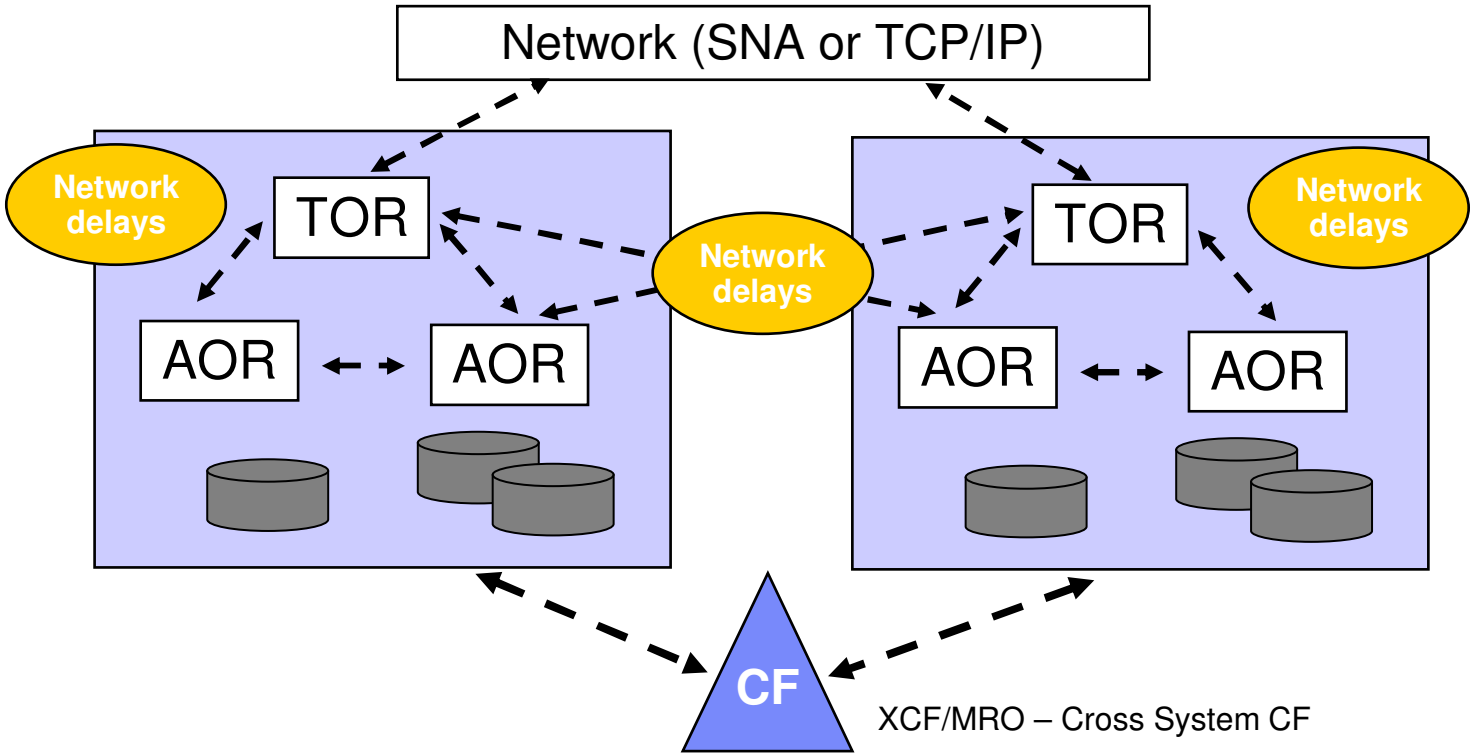
- PROFILE.TCPIP parameters
  - PORT
    - Reserve ports for IMS Connect
      - Include the NODELAYACK parameter for multi-message applications
      - Example benchmark
        - http://www-01.ibm.com/support/docview.wss?uid=swg21079911
  - SOMAXCONN
    - The number of connection requests that can be queued because IMS Connect has not yet issued the accept call - Default setting is 10
- IMS Connect parameter MAXSOC
  - Total number of sockets IMS Connect supports across all ports at the same time
    - Note - USS parameter MAXFILEPROC must be equal to or greater than the value of the IMS Connect parameter MAXSOC
  - IMS Connect issues warning message HWSS0772W when the number of sockets reaches the default warning threshold of 80 percent of MAXSOC

IMS Connect Monitoring
An Example



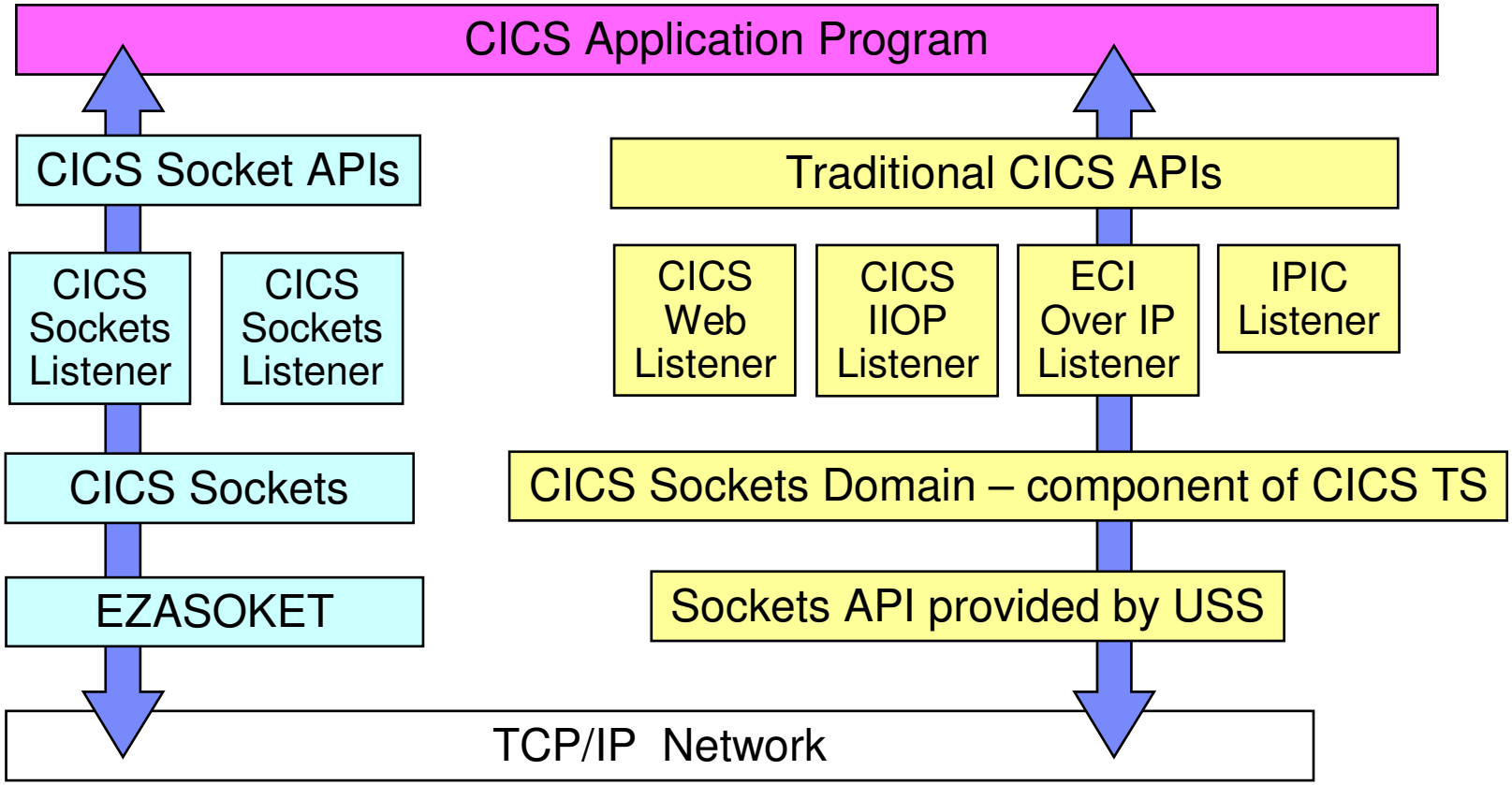
Note – This display requires IMS Connect Extensions And OMEGAMON IMS

The Network Impacts CICS Processing



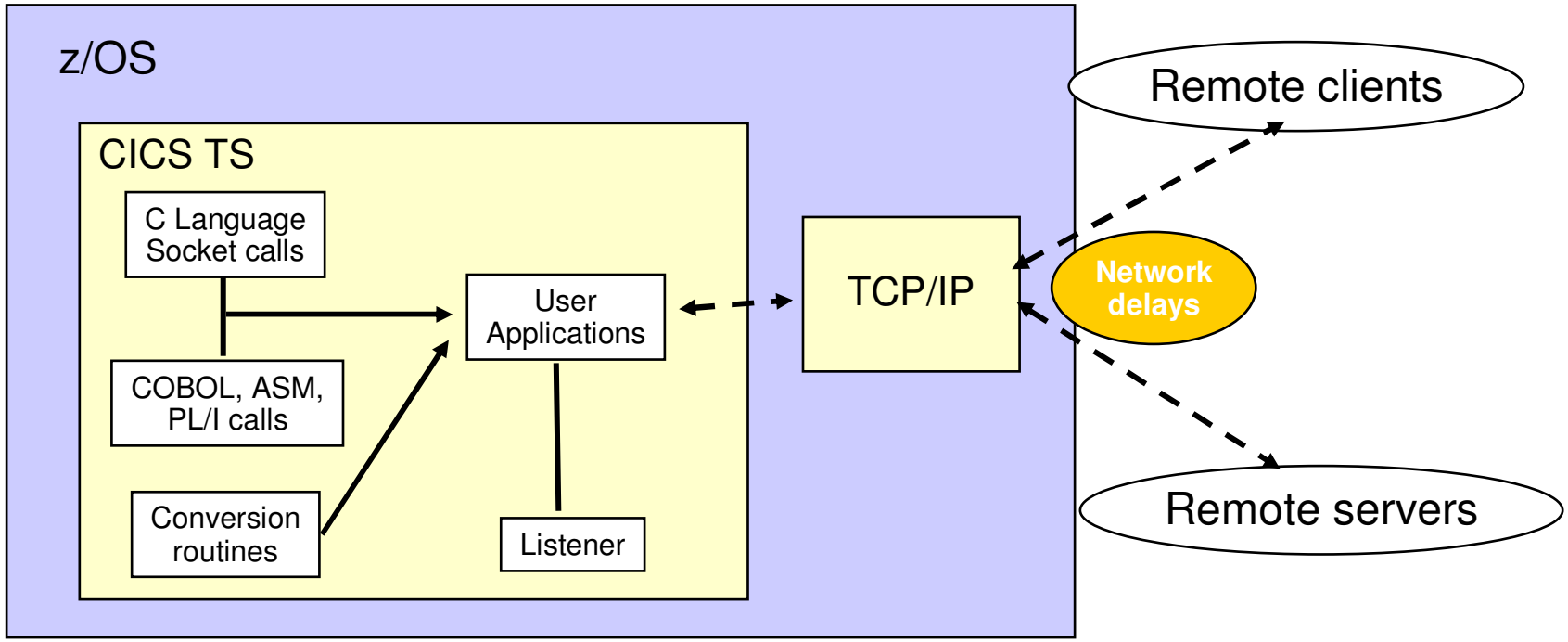
- Network potentially impacts CICS in a variety of ways
  - Connections to CICS – connections via a variety of means
  - Communication within CICS - ISC and MRO
    - InterSystems Communication - system to system, Multi-Region Operation - region to region, and IPIC – IP InterCommunications

CICS Sockets Versus CICS Sockets Domain



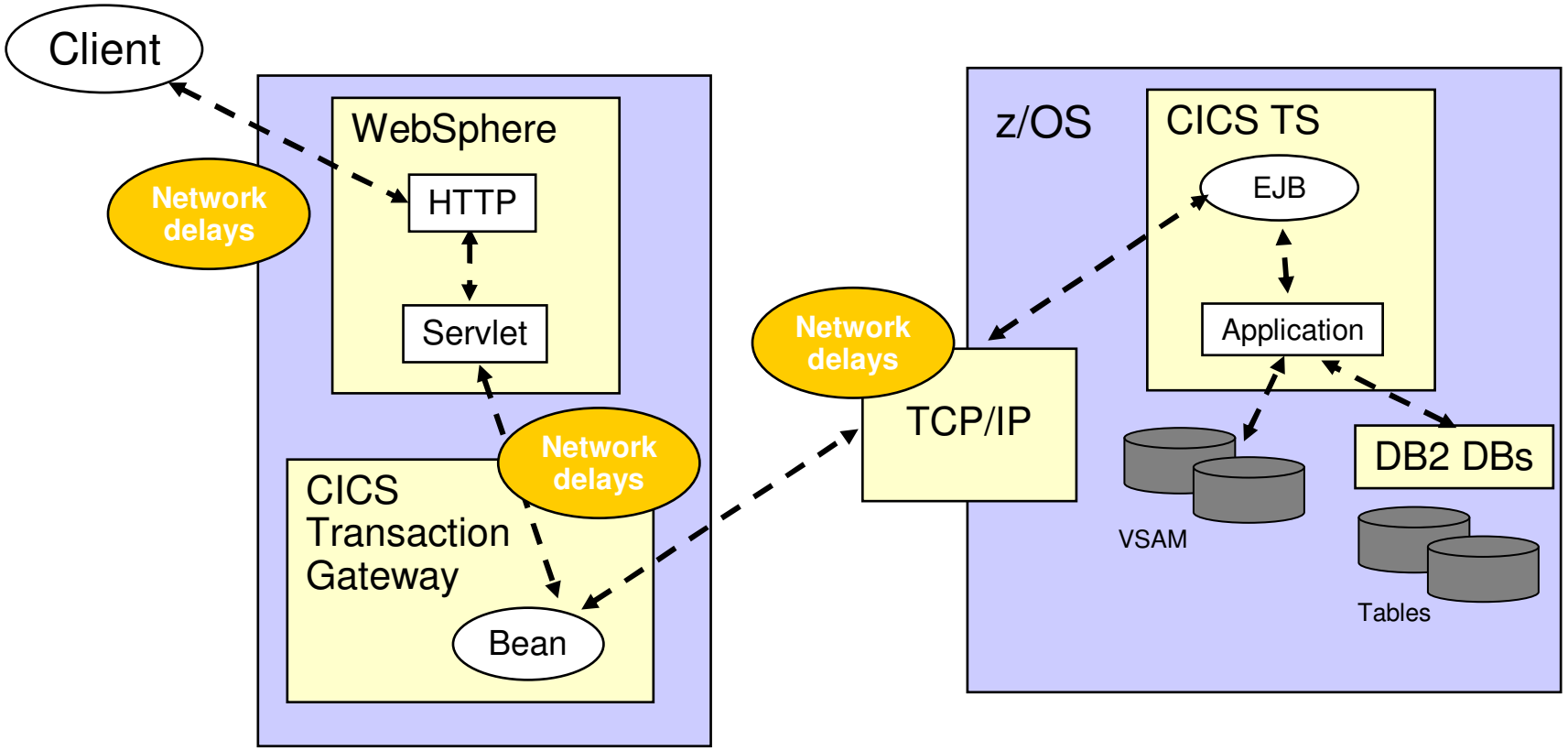
- **CICS Sockets – a component of Communications Server for z/OS**
  - General purpose socket API for use by CICS programmers
- **CICS Sockets Domain – a component of CICS TS**
  - Does not have direct access to the socket
  - Communicates with CICS Socket Domain Services

CICS Socket Interface Example



- Socket API available for C, COBOL, PL/I and ASM applications
- Listener is a CICS transaction
- Conversion routines – ASCII/EBCDIC

A WebSphere Example With CICS Transaction Gateway

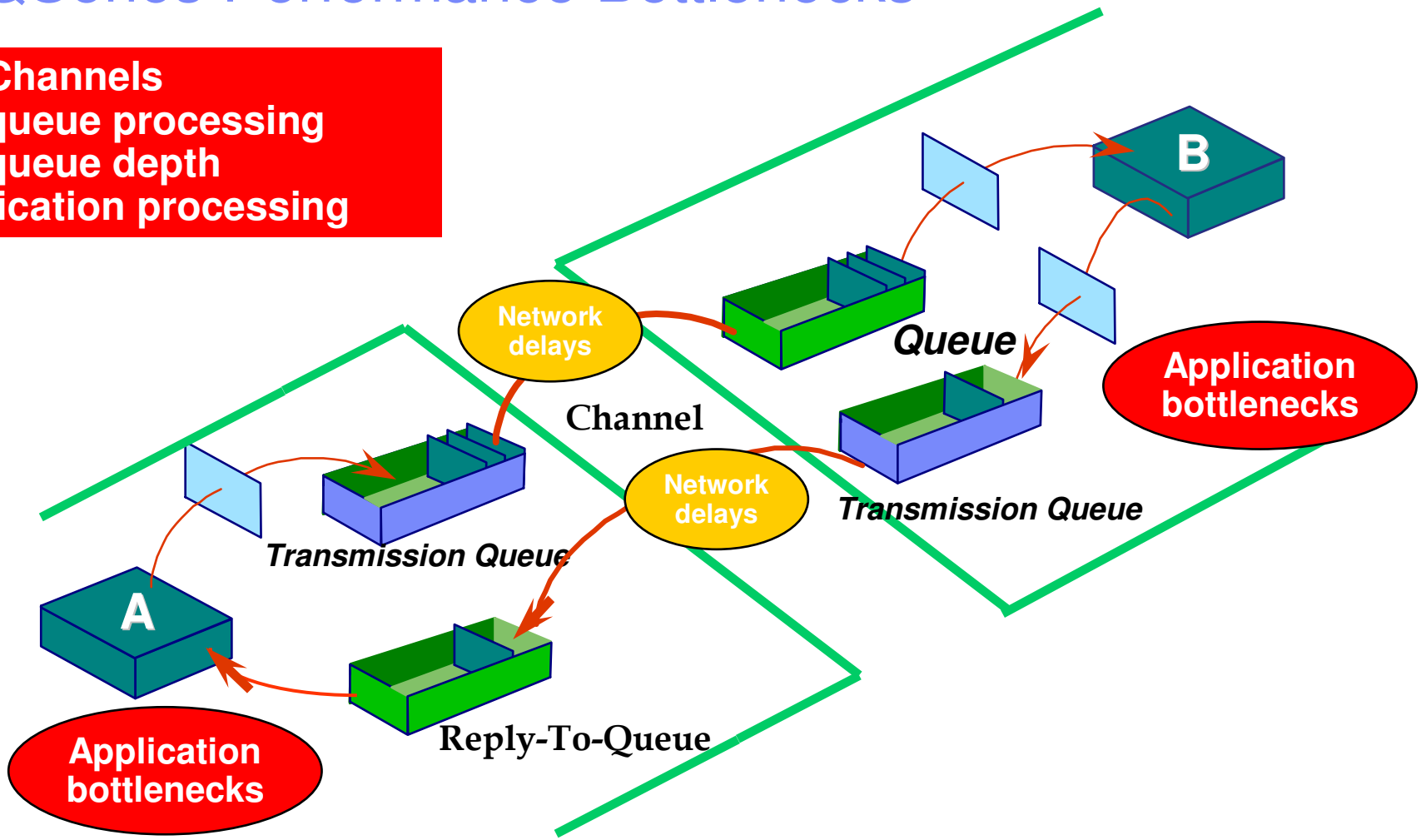


CICS Network
Examples Of Relevant Parameters

- TCPIPSERVICE parameter defines services
  - ECI over TCP/IP (for CICS Clients), IIOP, CICS Web support (HTTP), IPIC (ISC), or a user-defined protocol.
  - For use only with the CICS-provided TCP/IP services, and have nothing to do with the z/OS Communications Server IP CICS Sockets interface
- BACKLOG parameter
  - Specifies the maximum number of inbound TCP/IP connection requests that can be queued in TCP/IP for CICS processing
  - When the maximum number is reached, TCP/IP rejects additional connection requests
- MAXSOCKETS parameter
  - Maximum number of IP sockets that can be managed by the CICS sockets domain
  - If the CICS region userid does not have superuser authority, the maximum possible value is the value of the MAXFILEPROC parameter in SYS1.PARMLIB member BPXPRMxx.
  - MAXSOCKETS and maximum tasks (MXT)
    - Recommendation - MAXSOCKETS should not be a subset of MXT
    - http://www-01.ibm.com/support/docview.wss?uid=swg21596250&myns=swgother&mynp=OCSSGMGV&mync=R

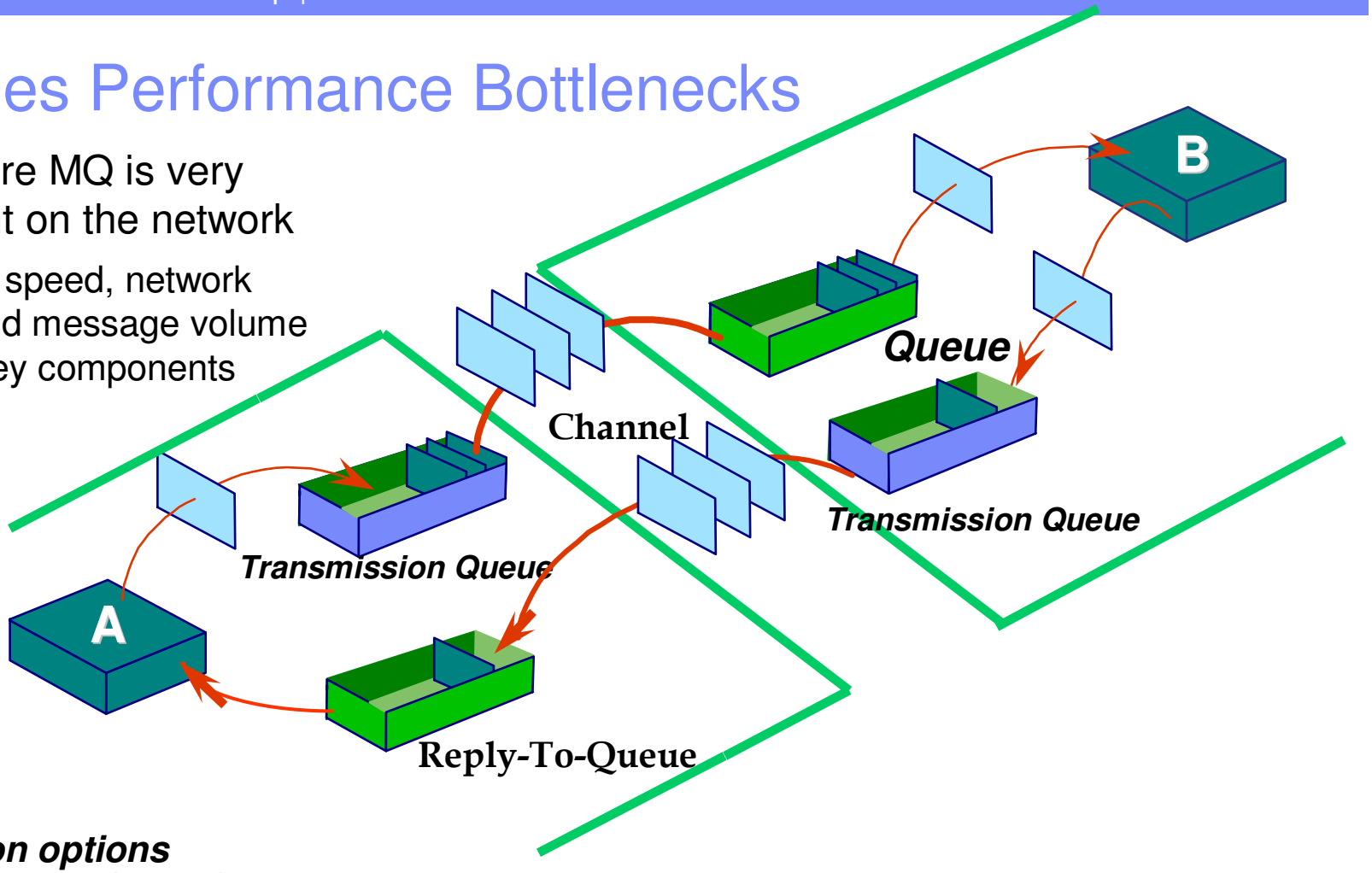
MQSeries Performance Bottlenecks

MQ Channels
MQ queue processing
MQ queue depth
Application processing



MQSeries Performance Bottlenecks

- WebSphere MQ is very dependent on the network
  - Network speed, network traffic and message volume are all key components



Optimization options

Increase network speed

Compress messages - decreases network transmission by reducing the size of the message.

Channel parameters

Batch size defines the maximum number of messages sent within a batch.

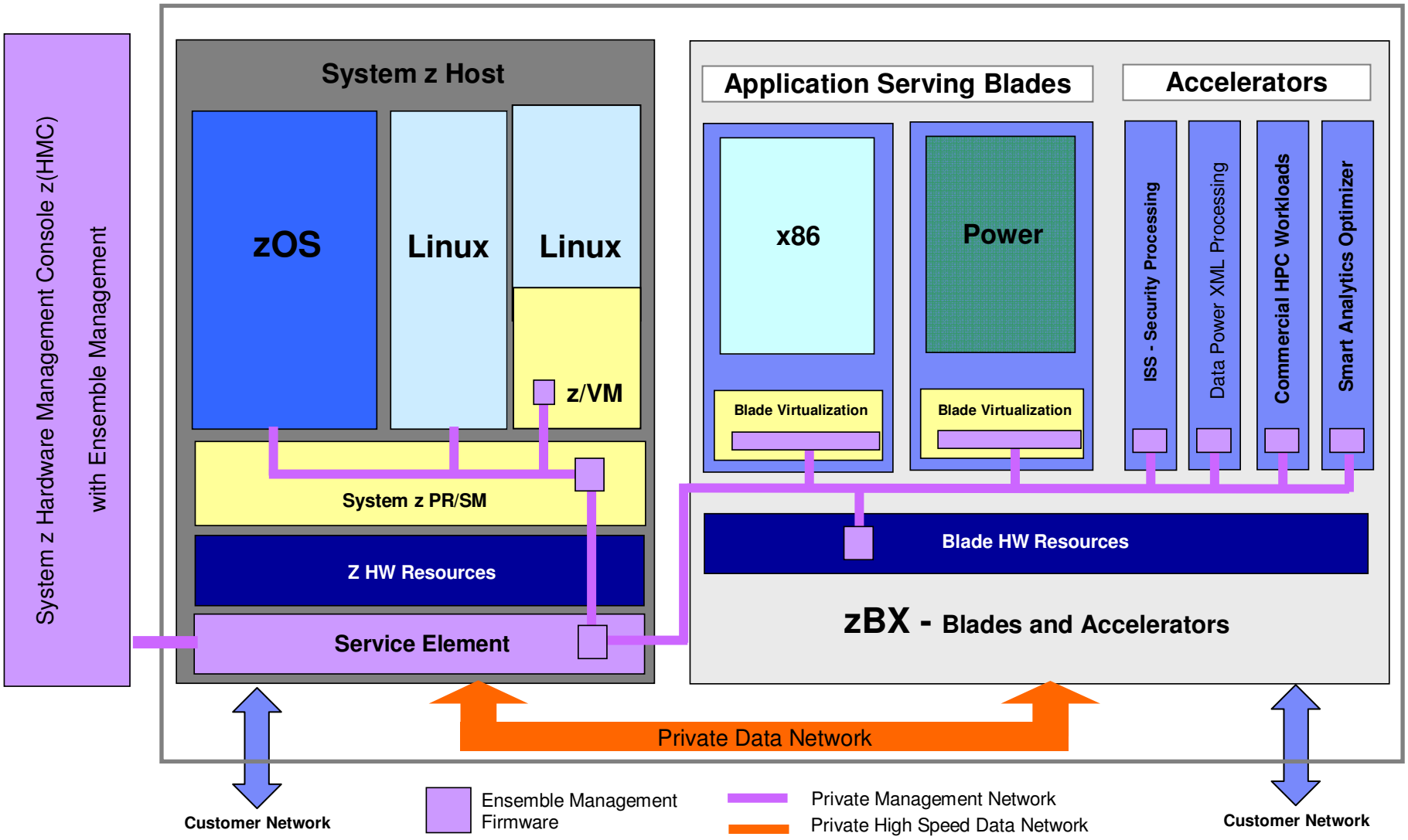
Reduces the amount of channel processing required.

Note – batching for small applications may result in delays and spikes

MQ Series
Configuration/Application Options And Network Impact

- Consider MQCONN and MQPUT patterns
  - MQCONN connects the application program to the MQ queue manager
    - Note - Cost of MQCONN high
  - MQPUT puts a message on a queue that was opened using MQOPEN
  - Similar to DB2 SQL call scenario
    - Consider cost of back and forth activity versus application logic
- Channel parameters
  - Batchsz – defines the maximum number of messages sent within a batch
    - Reduces the amount of channel processing required
  - Channel message compression
    - Some compression can be CPU heavy - how compressible is the data?
- Persistent versus non-persistent messages
  - Persistent messages are written to logs and queue data files
    - May be recovered by the queue manager after restart from failure
    - Persistent messages may have I/O and logging bottlenecks
  - Non-persistent messages are discarded after a failure
- Fast non-persistent messages
  - NPMSPEED – specifies speed at which non-persistent messages are sent

Address Network Latency Concerns With zEnterprise

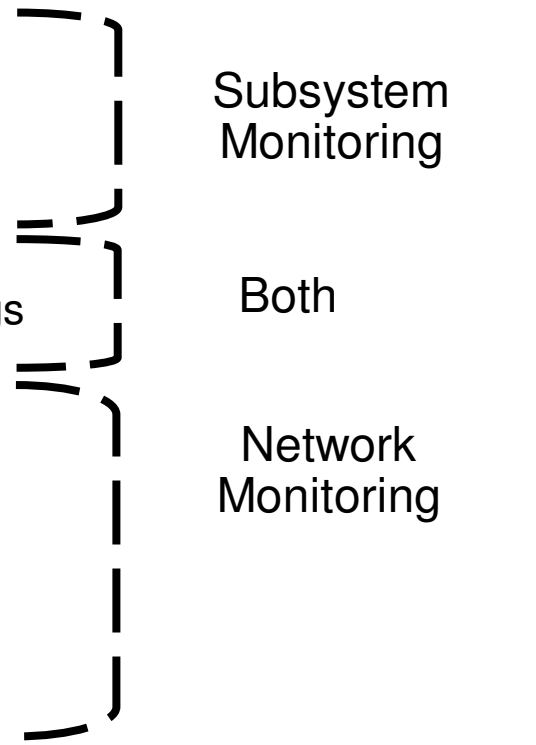


Defining A Monitoring Strategy
Many Factors May Impact Response Time

- Host processing bottlenecks
  - Transaction bottlenecks, application failures/stopped resources, high I/O and poor BP ratios, transaction/message queues, concurrency/lock conflicts
- Network performance
  - Network congestion, data fragmentation, data retransmission
- Network hardware issues
  - Adapter hardware errors, hardware configuration errors, hardware congestion issues
- Application subsystem connection issues
  - Application errors, subsystem configuration errors
- Application issues
  - Application design and logic problems

Defining A Monitoring Strategy
Monitoring At Multiple Levels

- Monitor at the host application subsystem level
  - IMS, CICS, DB2, WebSphere, WebSphere MQ
  - Response time, transaction rates, message rates, queues
- Monitor host application network connection activity
  - Connection activity, connection counts, connection backlogs
- Monitor at the interface level
  - OSA adapters, error counts, fragmentation counts, retransmission counts
- Monitor at the network connection level
  - Response time, traffic counts, error counts, fragmentation counts, retransmission counts
- Integrate host and network monitoring
- Monitor from an end-to-end perspective



Example - Understanding IMS Response Time

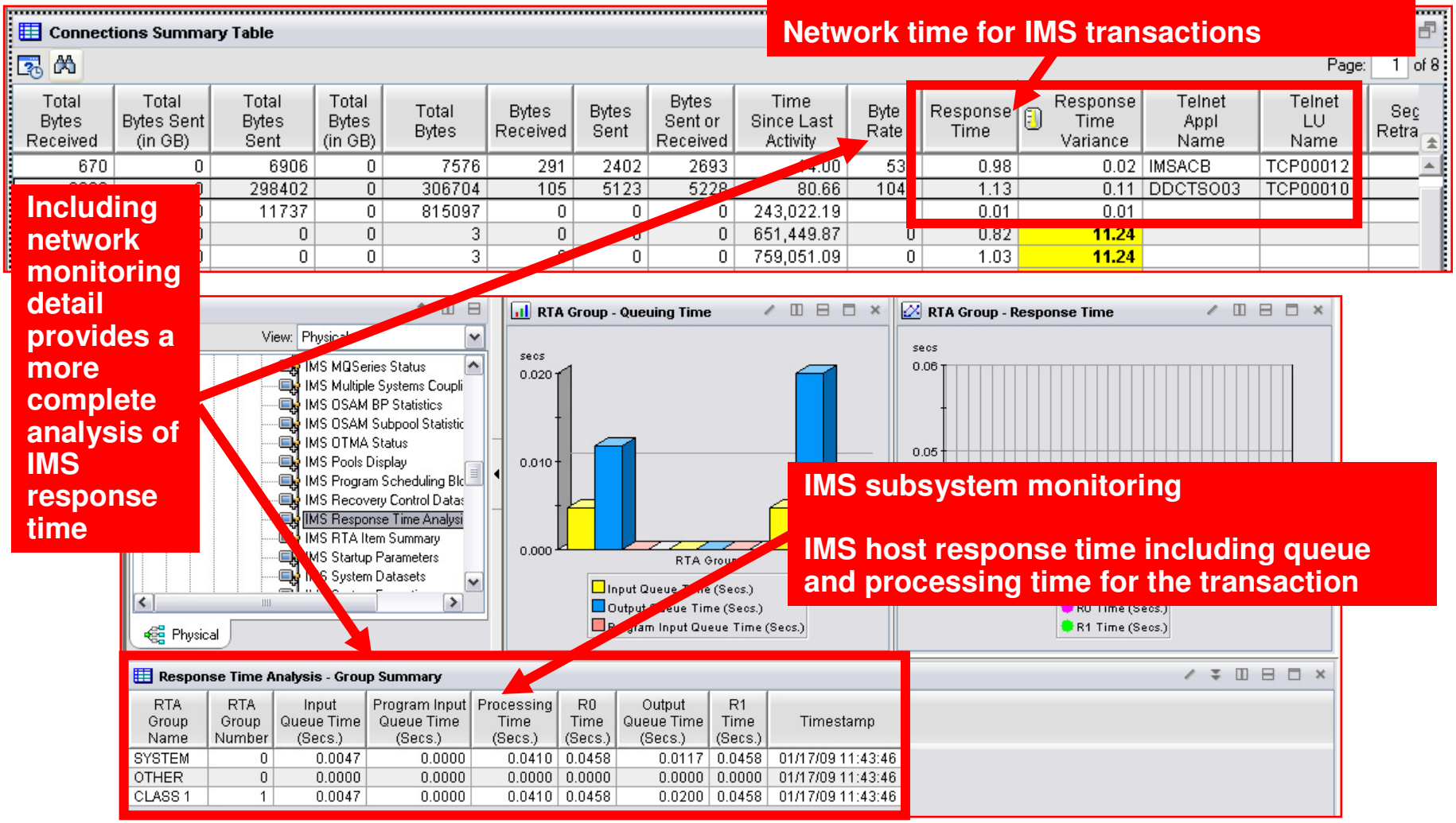
Mainframe network monitoring

Network time for IMS transactions

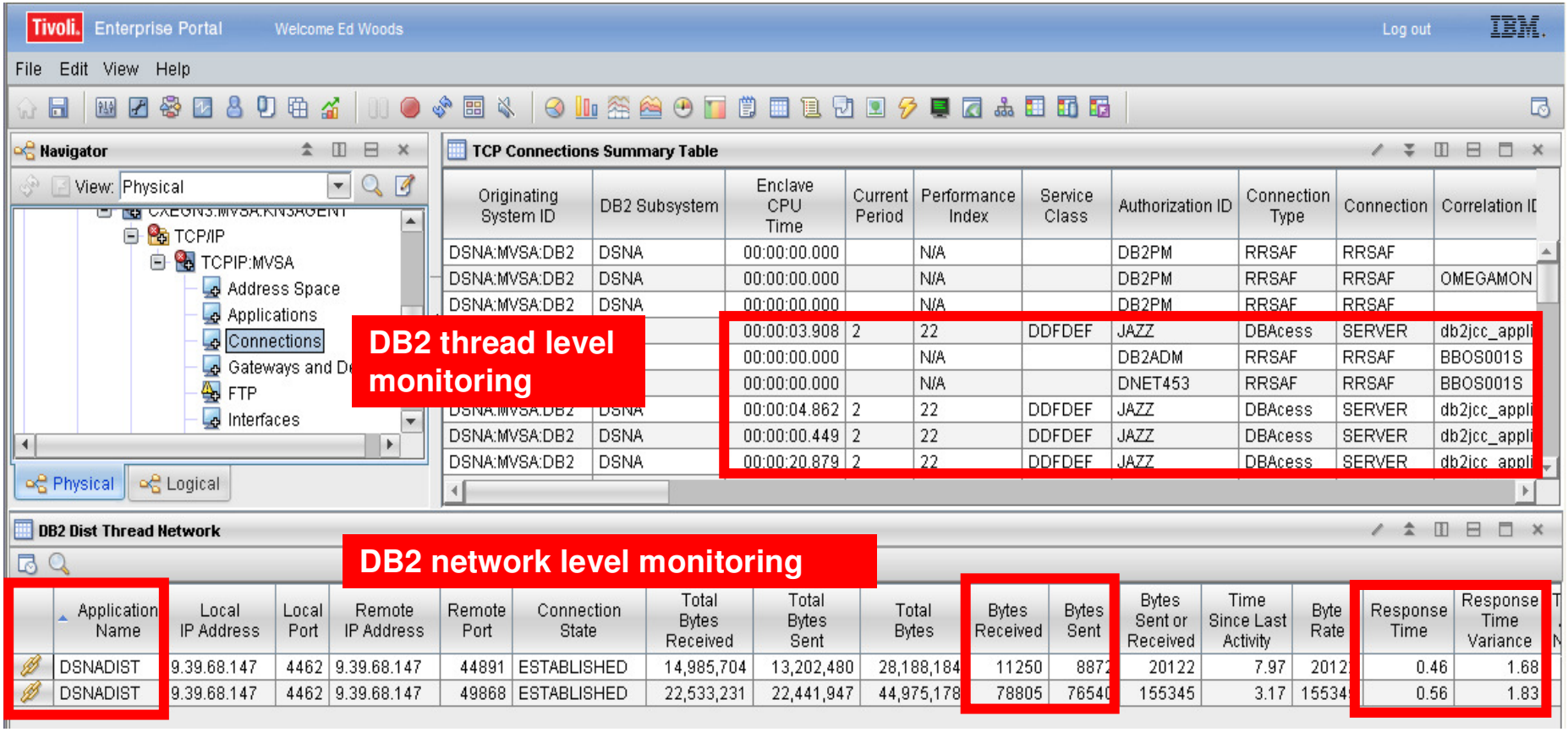
Including network monitoring detail provides a more complete analysis of IMS response time

IMS subsystem monitoring

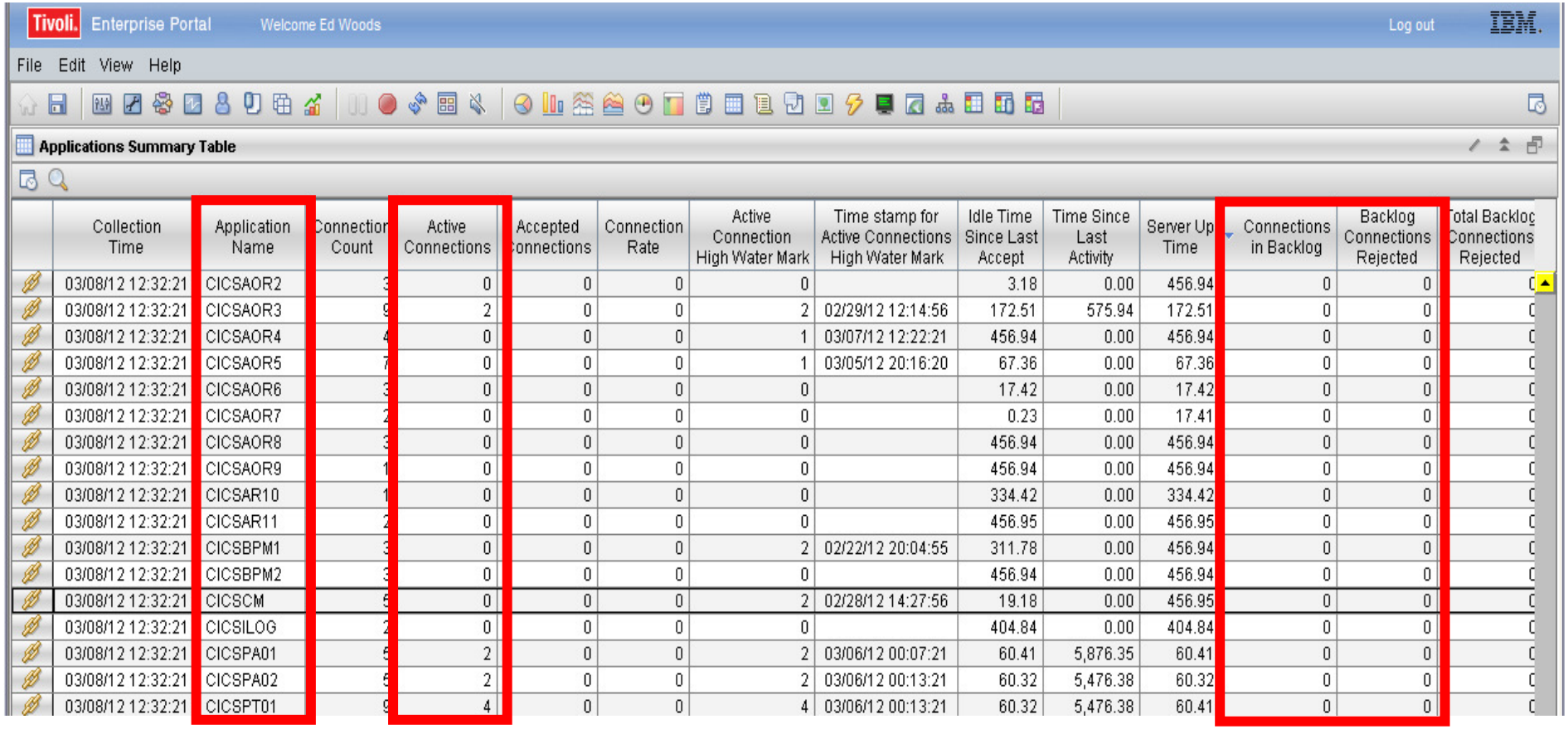
IMS host response time including queue and processing time for the transaction



Another Example
Combining Host And Network Level Monitoring



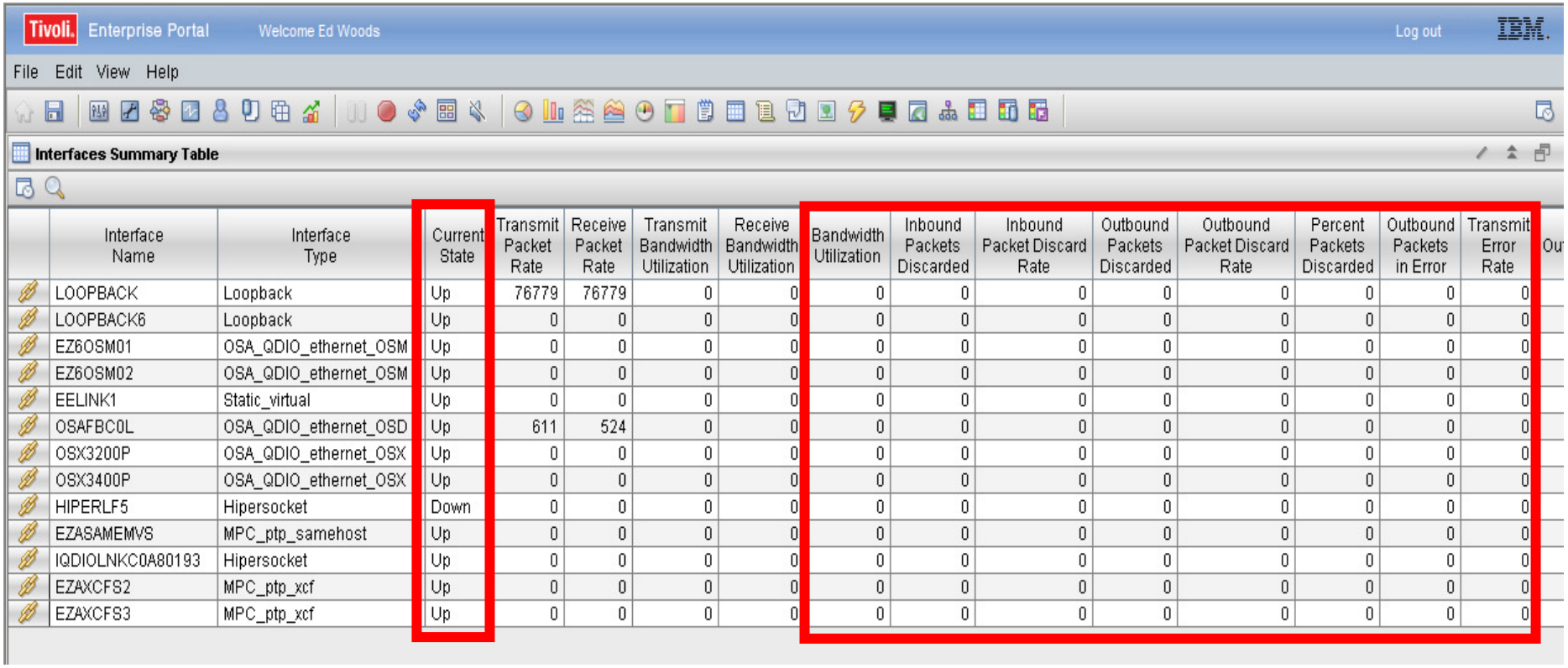
Monitor Host Application Network Connection Activity



Collection Time	Application Name	Connection Count	Active Connections	Accepted Connections	Connection Rate	Active Connection High Water Mark	Time stamp for Active Connections High Water Mark	Idle Time Since Last Accept	Time Since Last Activity	Server Up Time	Connections in Backlog	Backlog Connections Rejected	Total Backlog Connections Rejected
03/08/12 12:32:21	CICSAOR2	3	0	0	0	0		3.18	0.00	456.94	0	0	0
03/08/12 12:32:21	CICSAOR3	9	2	0	0	2	02/29/12 12:14:56	172.51	575.94	172.51	0	0	0
03/08/12 12:32:21	CICSAOR4	4	0	0	0	1	03/07/12 12:22:21	456.94	0.00	456.94	0	0	0
03/08/12 12:32:21	CICSAOR5	7	0	0	0	1	03/05/12 20:16:20	67.36	0.00	67.36	0	0	0
03/08/12 12:32:21	CICSAOR6	3	0	0	0	0		17.42	0.00	17.42	0	0	0
03/08/12 12:32:21	CICSAOR7	2	0	0	0	0		0.23	0.00	17.41	0	0	0
03/08/12 12:32:21	CICSAOR8	3	0	0	0	0		456.94	0.00	456.94	0	0	0
03/08/12 12:32:21	CICSAOR9	1	0	0	0	0		456.94	0.00	456.94	0	0	0
03/08/12 12:32:21	CICSAR10	1	0	0	0	0		334.42	0.00	334.42	0	0	0
03/08/12 12:32:21	CICSAR11	2	0	0	0	0		456.95	0.00	456.95	0	0	0
03/08/12 12:32:21	CICSBPM1	3	0	0	0	2	02/22/12 20:04:55	311.78	0.00	456.94	0	0	0
03/08/12 12:32:21	CICSBPM2	3	0	0	0	0		456.94	0.00	456.94	0	0	0
03/08/12 12:32:21	CICSCM	5	0	0	0	2	02/28/12 14:27:56	19.18	0.00	456.95	0	0	0
03/08/12 12:32:21	CICSILOG	2	0	0	0	0		404.84	0.00	404.84	0	0	0
03/08/12 12:32:21	CICSPA01	5	2	0	0	2	03/06/12 00:07:21	60.41	5,876.35	60.41	0	0	0
03/08/12 12:32:21	CICSPA02	5	2	0	0	2	03/06/12 00:13:21	60.32	5,476.38	60.32	0	0	0
03/08/12 12:32:21	CICSPT01	9	4	0	0	4	03/06/12 00:13:21	60.32	5,476.38	60.41	0	0	0

- Connection activity, connection counts, connection backlogs
  - Look for applications with connection failures and backlogs

Monitor At The Interface Level



	Interface Name	Interface Type	Current State	Transmit Packet Rate	Receive Packet Rate	Transmit Bandwidth Utilization	Receive Bandwidth Utilization	Bandwidth Utilization	Inbound Packets Discarded	Inbound Packet Discard Rate	Outbound Packets Discarded	Outbound Packet Discard Rate	Percent Packets Discarded	Outbound Packets in Error	Transmit Error Rate	Ou
	LOOPBACK	Loopback	Up	76779	76779	0	0	0	0	0	0	0	0	0	0	
	LOOPBACK6	Loopback	Up	0	0	0	0	0	0	0	0	0	0	0	0	
	EZ6OSM01	OSA_QDIO_ethernet_OSM	Up	0	0	0	0	0	0	0	0	0	0	0	0	
	EZ6OSM02	OSA_QDIO_ethernet_OSM	Up	0	0	0	0	0	0	0	0	0	0	0	0	
	EELINK1	Static_virtual	Up	0	0	0	0	0	0	0	0	0	0	0	0	
	OSAFBC0L	OSA_QDIO_ethernet_OSD	Up	611	524	0	0	0	0	0	0	0	0	0	0	
	OSX3200P	OSA_QDIO_ethernet_OSX	Up	0	0	0	0	0	0	0	0	0	0	0	0	
	OSX3400P	OSA_QDIO_ethernet_OSX	Up	0	0	0	0	0	0	0	0	0	0	0	0	
	HIPERLF5	Hipersocket	Down	0	0	0	0	0	0	0	0	0	0	0	0	
	EZASAMEMVS	MPC_ptp_samehost	Up	0	0	0	0	0	0	0	0	0	0	0	0	
	IQDIOLNKC0A80193	Hipersocket	Up	0	0	0	0	0	0	0	0	0	0	0	0	
	EZAXCFS2	MPC_ptp_xcf	Up	0	0	0	0	0	0	0	0	0	0	0	0	
	EZAXCFS3	MPC_ptp_xcf	Up	0	0	0	0	0	0	0	0	0	0	0	0	

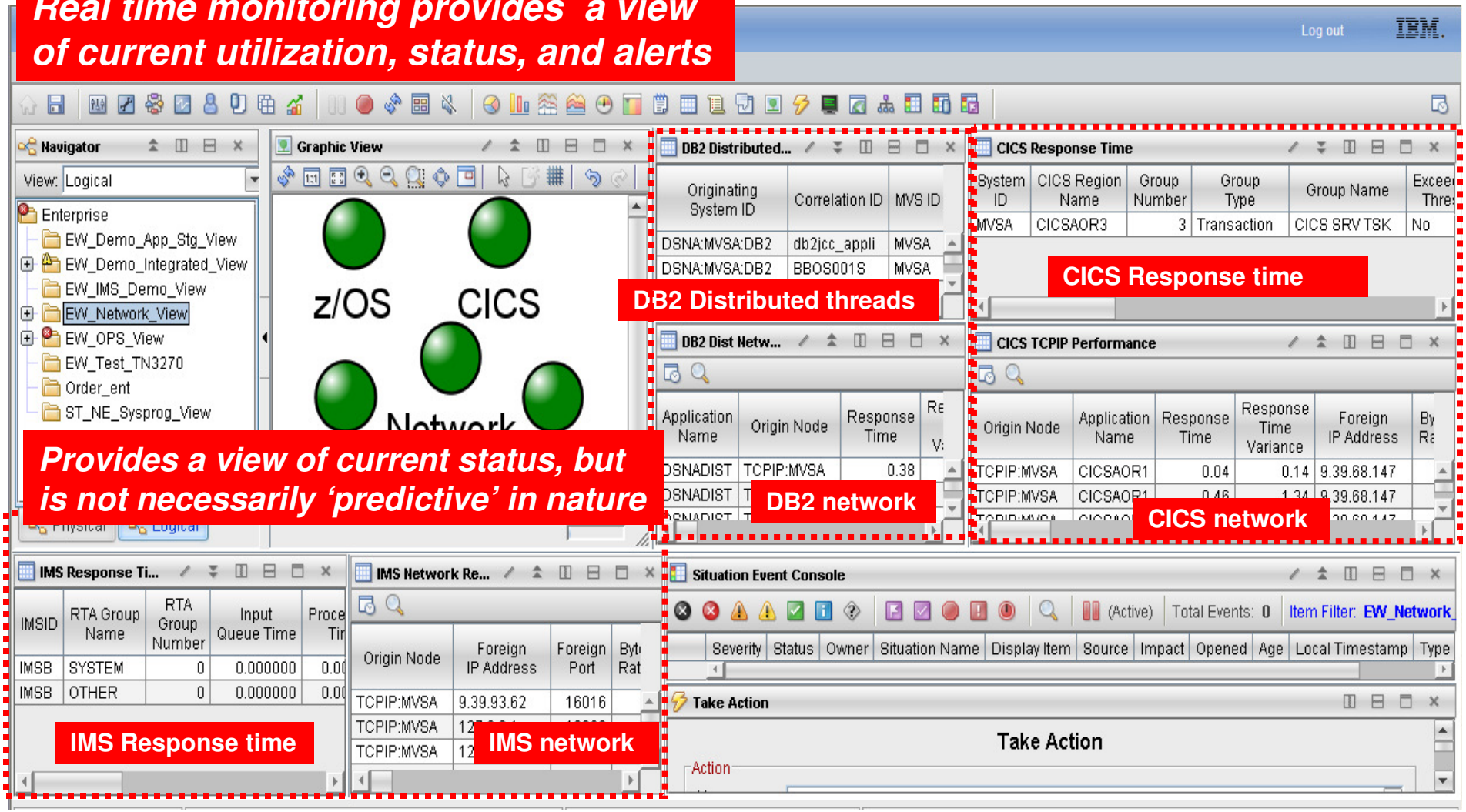
- Monitor for interface status, bandwidth utilization, and errors
- Look for potential problems at the interface level

Dashboard Level Monitoring
Creating An Integrated Performance Interface

- Creating an integrated performance management display allows for the easy inclusion of network detail into various mainframe monitoring displays
- Integrated monitoring takes several forms
  - Integrated displays pulling together performance detail from multiple sources (host and network monitoring)
  - Integrated cross monitoring tool navigation
  - History integrated with real time performance information
  - Integrated alerts, alert correlation, and corrective actions

Dashboard Level Monitoring
Integrate Host And Network Monitoring

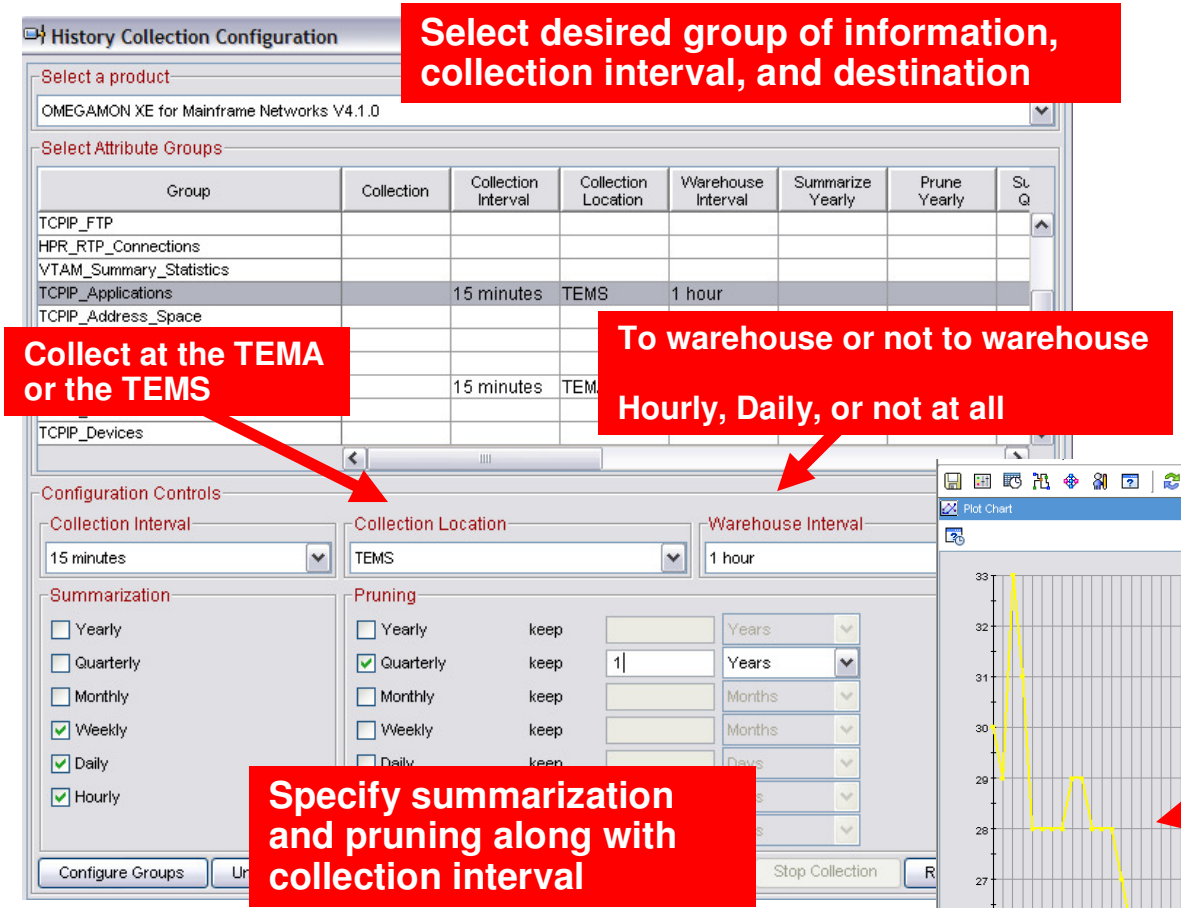
Real time monitoring provides a view of current utilization, status, and alerts

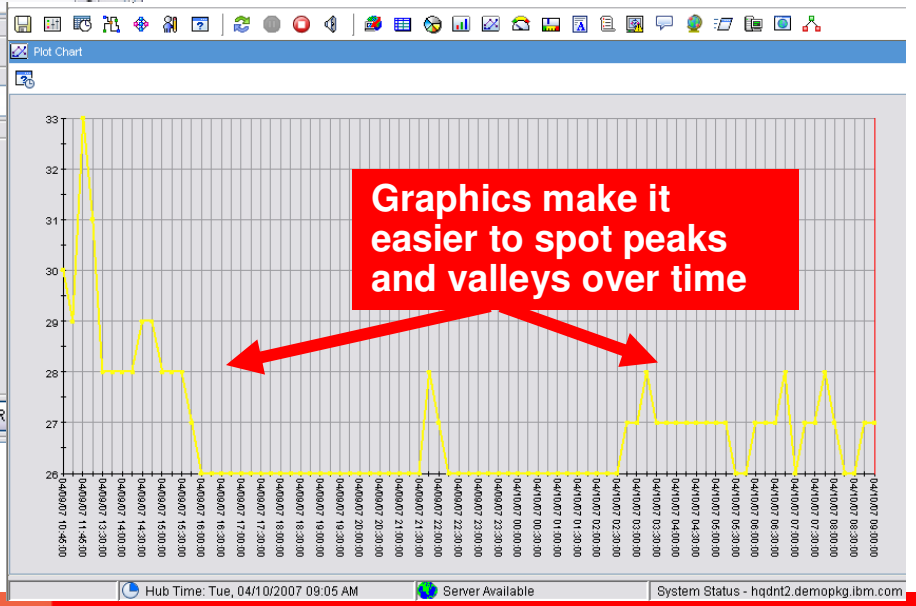


Provides a view of current status, but is not necessarily 'predictive' in nature

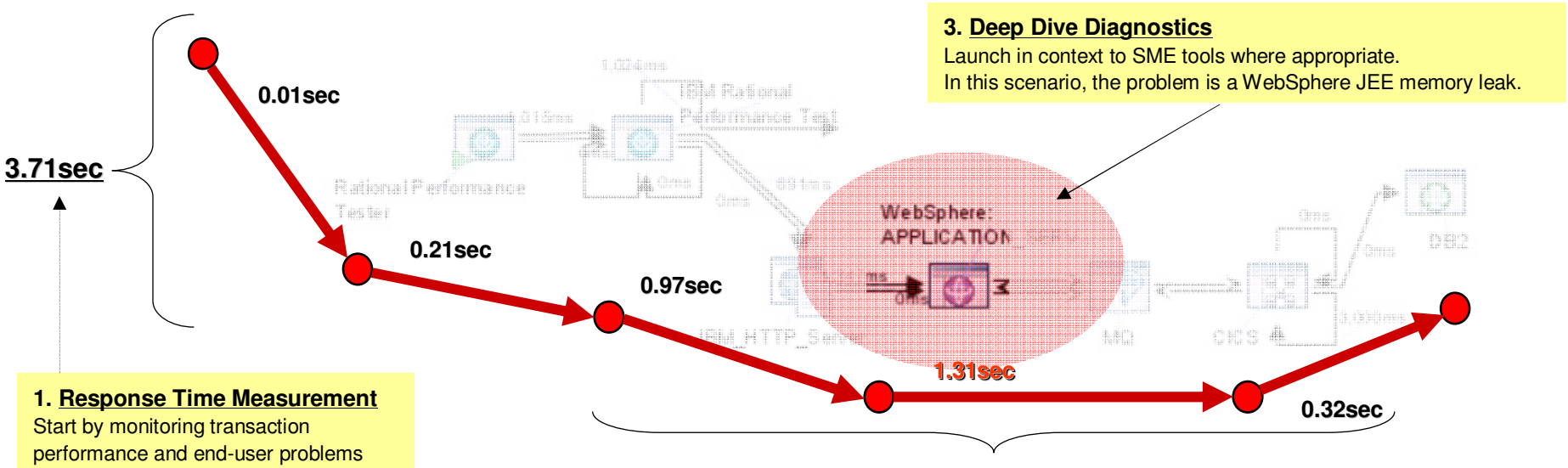
Collect History For Trending And Analysis

- Collect history data at multiple levels
  - System/LPAR level
  - Application level
  - Interface/adapter level
  - Connection level
- Frequency and retention will vary by type
- Useful for trending and alert threshold analysis





End-to-End Monitoring, Tracking and Diagnosis



Transaction Root Cause Analysis

1. **Sense** End User Experience and alert on threshold violation	2. **Isolate** by measuring performance data against baseline through entire infrastructure	3. **Diagnose and repair** through launch-in-context into deep-dive diagnostics

Summary

- The network is an essential part of the overall mainframe application time line
  - Each network application/subsystem has interactions with the network
- It's important to understand how the mainframe interacts with the network
  - Application/subsystem configuration and options
- It is useful to have an integrated monitoring strategy that pulls together core mainframe and network monitoring information
  - Integrated dashboard views, integrated analysis, integrated alerts and automated corrections
  - Defining an end to end analysis strategy

Thank You!

Check Out My Blog

http://tivoliwithaz.blogspot.com