IBM Software Group

OMEGAMON XE For DB2

Usage Strategies And Best Practices

Ed Woods

Consulting IT Specialist

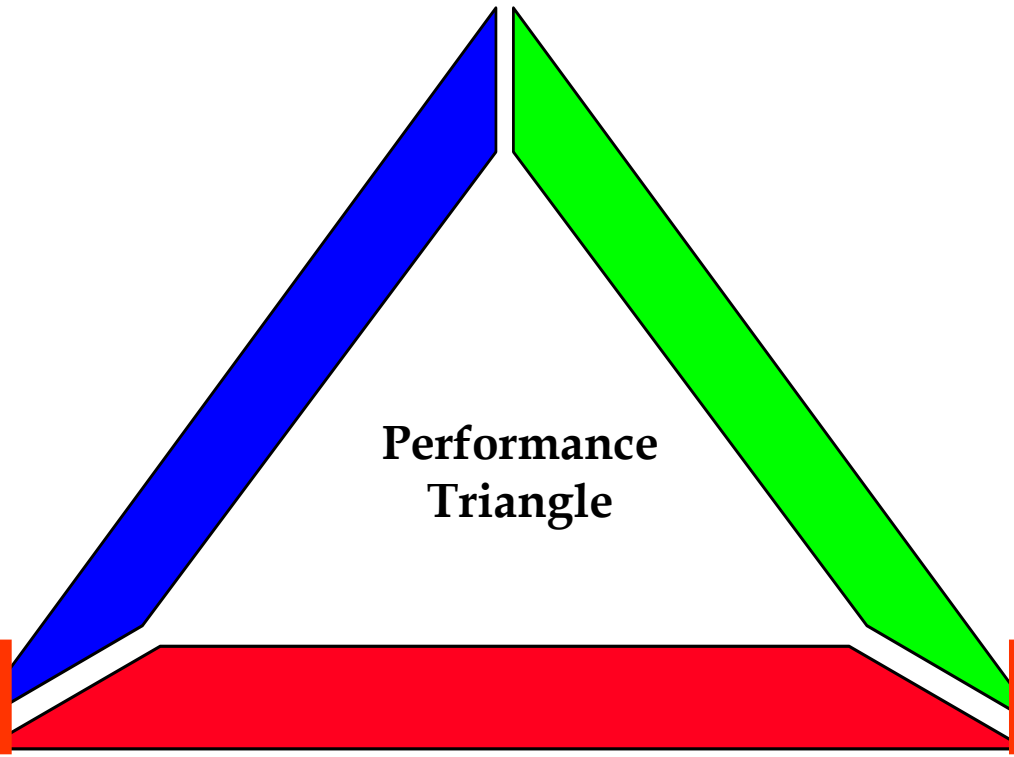
Tivoli software

e business on demand.

OMEGAMON Management Triangle

OMEGAMON DE – Dashboard Edition

High level monitoring
Multi-system view
Cross Platform View



3270 Interface

OMEGAMON XE

Subsystem details
3270 Interface
Historical details

Subsystem level monitoring
Browser And GUI interface
Proactive Alerting & Automation

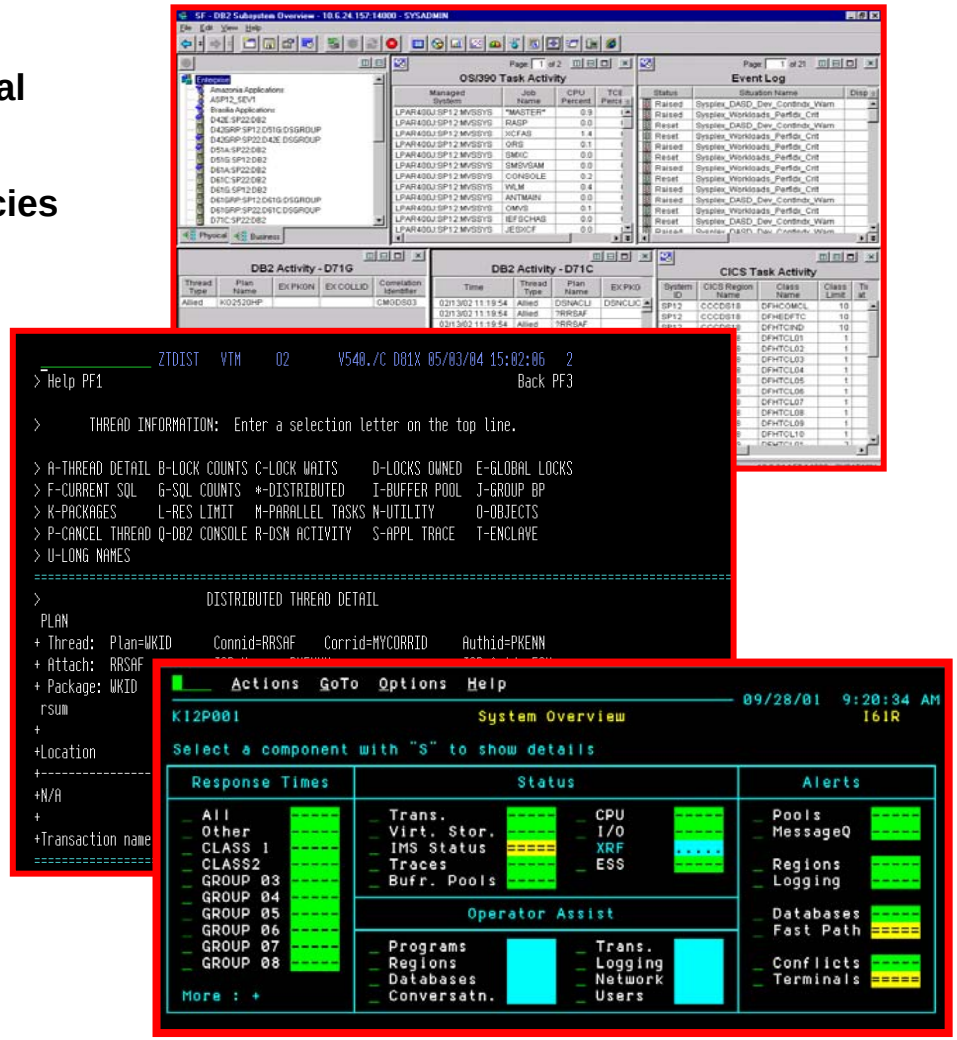
OMEGAMON DB2 Options & Interfaces

- **OMEGAMON XE GUI Interface**
  - Java client or web browser – Tivoli Portal
  - Real time and historical
  - Automation & alerts – Situations & Policies
  - Plex level information (CF, n-way)

OMEGAMON Classic

- 3270 Interface command interface
- Real Time & Historical
- Major & Minor commands
- Exceptions stored in classic profile

- **OMEGAMON CUA**
  - 3270 interface
  - Different views from Classic
  - Real Time & Historical
  - Warning & Critical exception alerts



OMEGAMON XE For DB2 UDB On z/OS
Major Features & Components

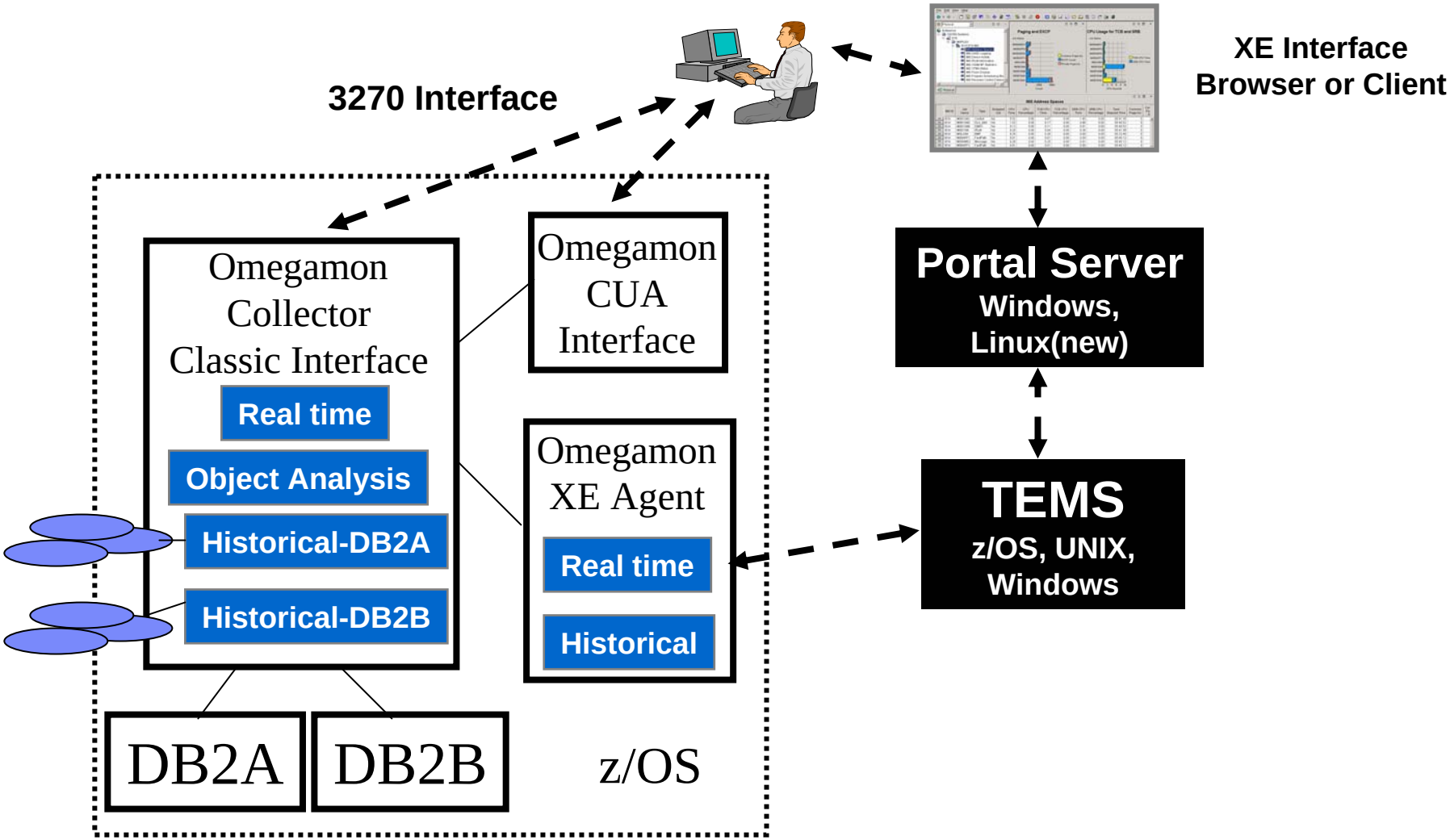
- Real Time Thread Analysis
  - ✓Thread performance (elapsed, CPU, getpage info)
  - ✓Thread Detail (lock detail, SQL detail, plan & package level
  - ✓Triggers, Procedures, & UDFs
- Real Time – DB2 subsystem
  - ✓Virtual Pool & EDM Pool analysis
    - ✓Pool performance
    - ✓Pool snapshot detail
  - ✓Locking & Logging
- Application Trace Facility
  - ✓Detailed performance tracing
- Choice Of Interfaces (XE Interface, 3270 Classic & CUA)
- Object Analysis
  - ✓I/O & getpage analysis
  - ✓Correlate activity by object & applications
- Lock Conflicts
- Near-Term Historical
  - ✓Near-term history online
- Historical Analysis
  - ✓Batch reporting from VSAM, DB2 or SMF
  - ✓XE Historical analysis
- ✓DB2Plex Monitoring View
  - ✓View CF structures
  - ✓Global lock analysis
- Automation capabilities

OMEGAMON XE For DB2 UDB On z/OS
Major Features & Components

GUI = Available in GUI
3270 = Available in 3270

- Real Time Thread Analysis
  - ✓Thread performance (elapsed, CPU, getpage info) GUI & 3270
  - ✓Thread Detail (lock detail, SQL detail, plan & package level 3270
  - ✓Triggers, Procedures, & UDFs
- Real Time – DB2 subsystem
  - ✓Virtual Pool & EDM Pool analysis
    - ✓Pool performance GUI & 3270
    - ✓Pool snapshot detail 3270
  - ✓Locking & Logging GUI & 3270
- Application Trace Facility 3270
  - ✓Detailed performance tracing
- Choice Of Interfaces (XE GUI & 3270 Interface, 3270 Classic & CUA)
- Object Analysis GUI & 3270
  - ✓I/O & getpage analysis
  - ✓Correlate activity by object & applications
- Lock Conflicts GUI & 3270
- Near-Term Historical 3270
  - ✓Near-term history online
- Historical Analysis
  - ✓Batch reporting from VSAM, DB2 or SMF 3270
  - ✓XE Historical analysis GUI
- ✓DB2Plex Monitoring View
  - ✓View CF structures
  - ✓Global lock analysis GUI
- Automation capabilities GUI

OMEGAMON XE For DB2
Components And Architecture



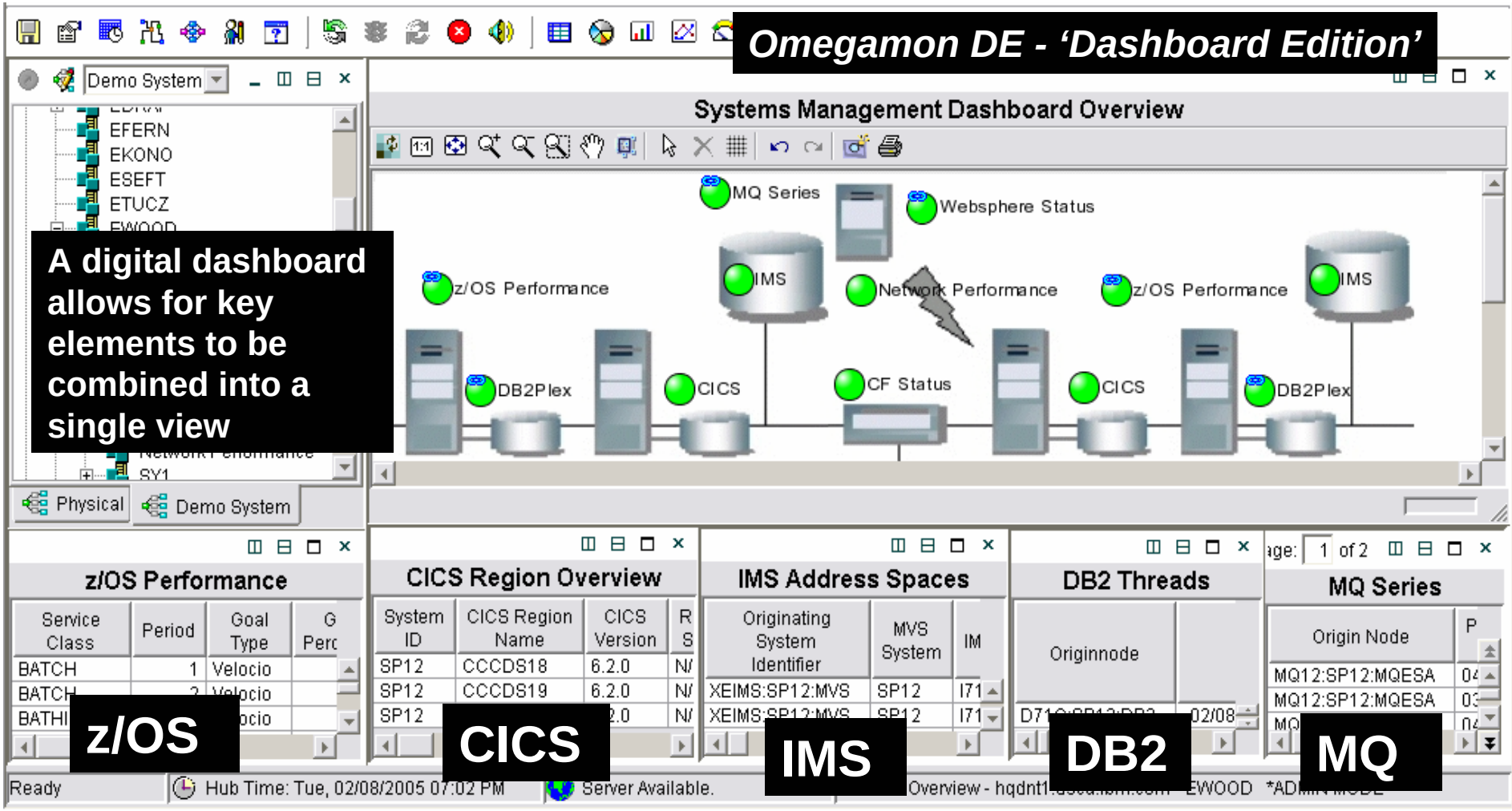
Omegamon DB2
XE GUI Interface versus 3270 – When To Use

- XE GUI Interface strengths and capabilities
  - Customizable high level overview of all DB2 activity
    - Thread activity and subsystem activity
  - Data sharing performance information (CF structures. global lock analysis)
  - Customizable alerts, automation, and corrective actions
- 3270 (Classic & CUA) Interface strengths and capabilities
  - Thread activity detailed analysis
    - Thread detail, timings, detail SQL activity, lock detail and activity
  - Subsystem activity detail
    - Virtual Pool and EDM Pool snapshot and detailed analysis
  - Application Trace Facility
  - Object Analysis (non data-sharing subsystems)
- Historical considerations
  - Omegamon Near Term History – 3270 Interface
  - XE GUI Interface snapshot historical

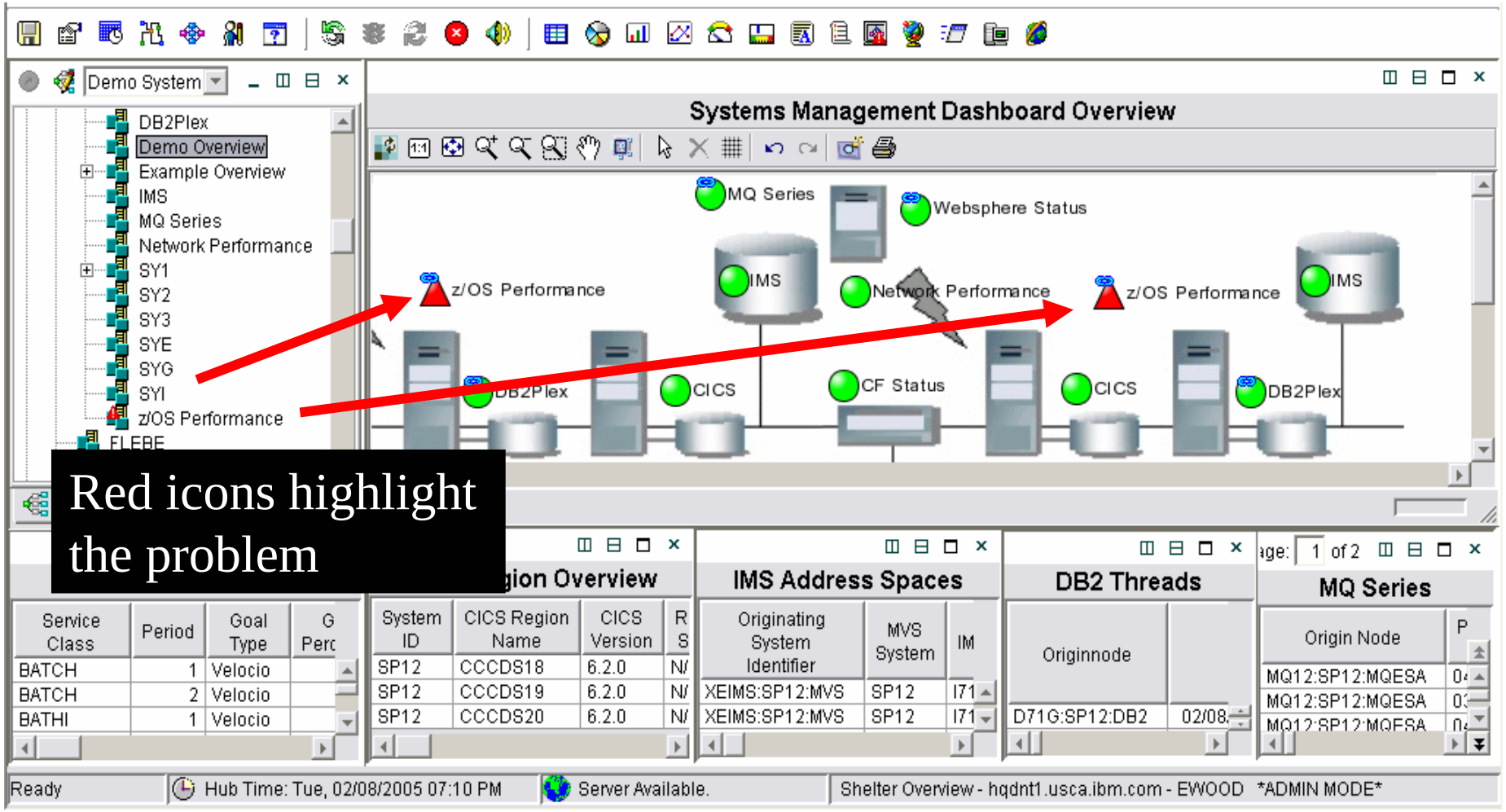
Omegamon XE GUI Interface

Integration, Consolidation, Customization, and Flexibility

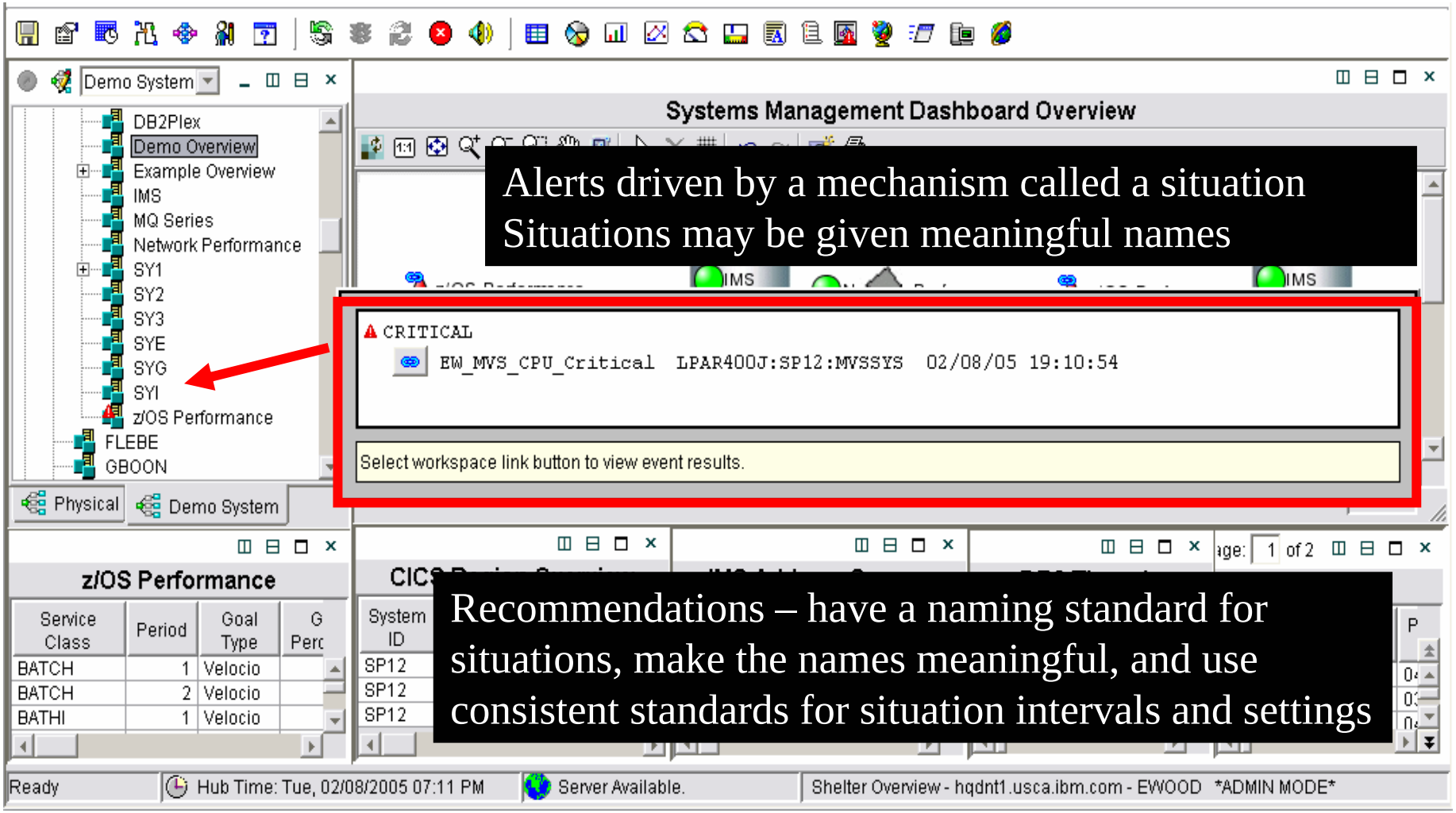
DB2 As Part Of A Bigger Picture



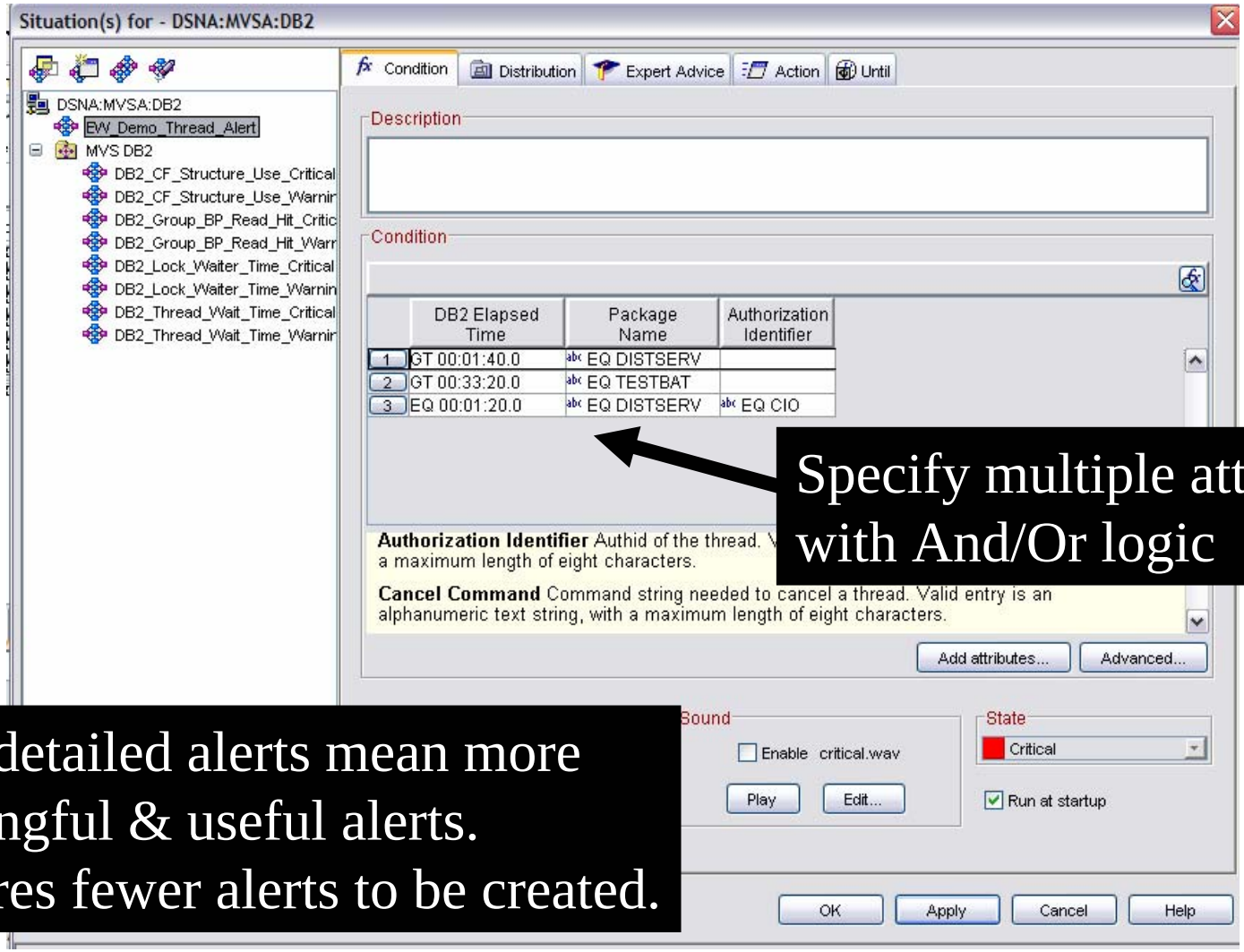
Event Management & Problem Isolation



Event Management & Problem Isolation



Omegamon XE Situations Enable Detailed Alerts



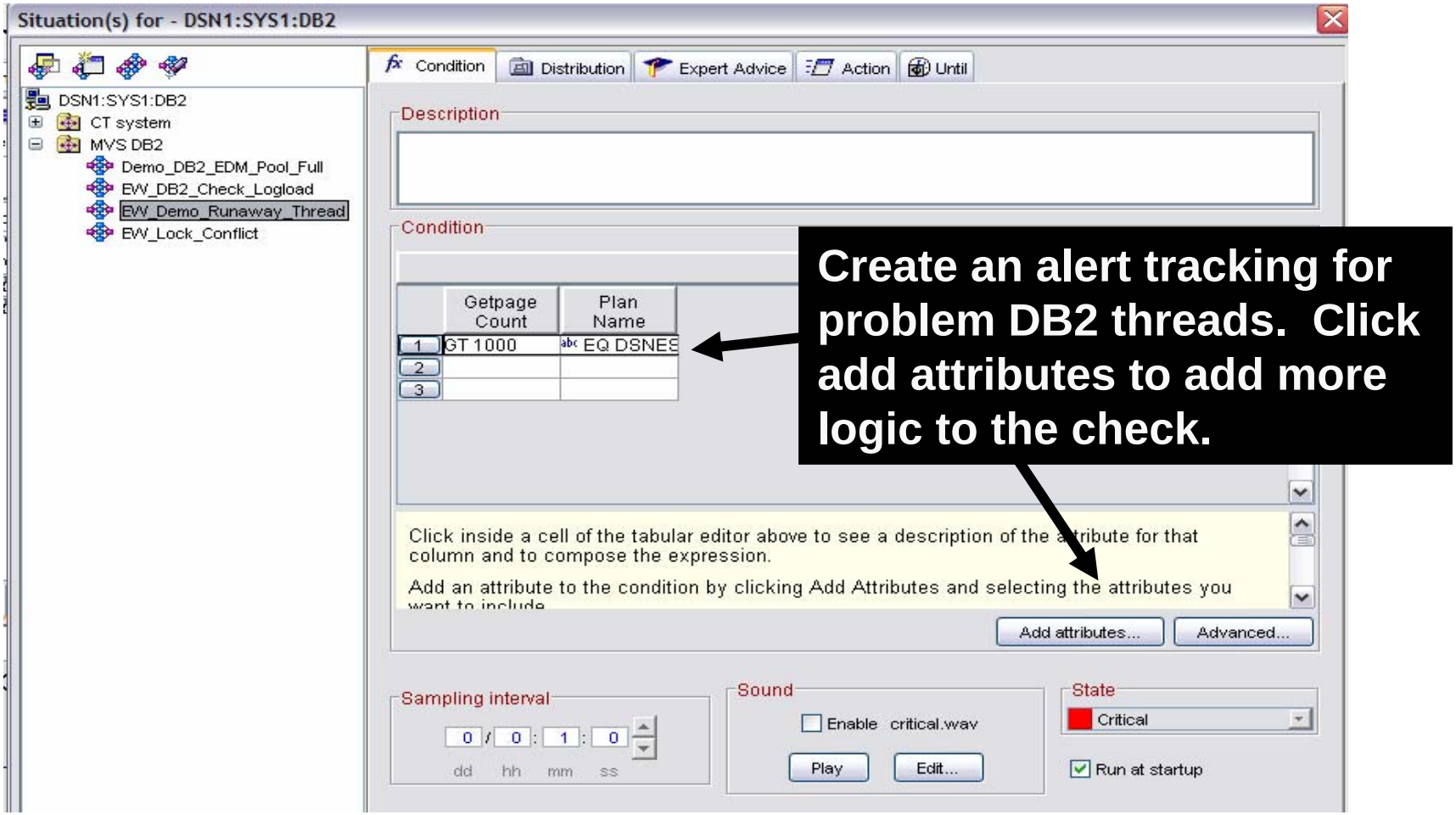
Specify multiple attributes with And/Or logic

More detailed alerts mean more meaningful & useful alerts.
Requires fewer alerts to be created.

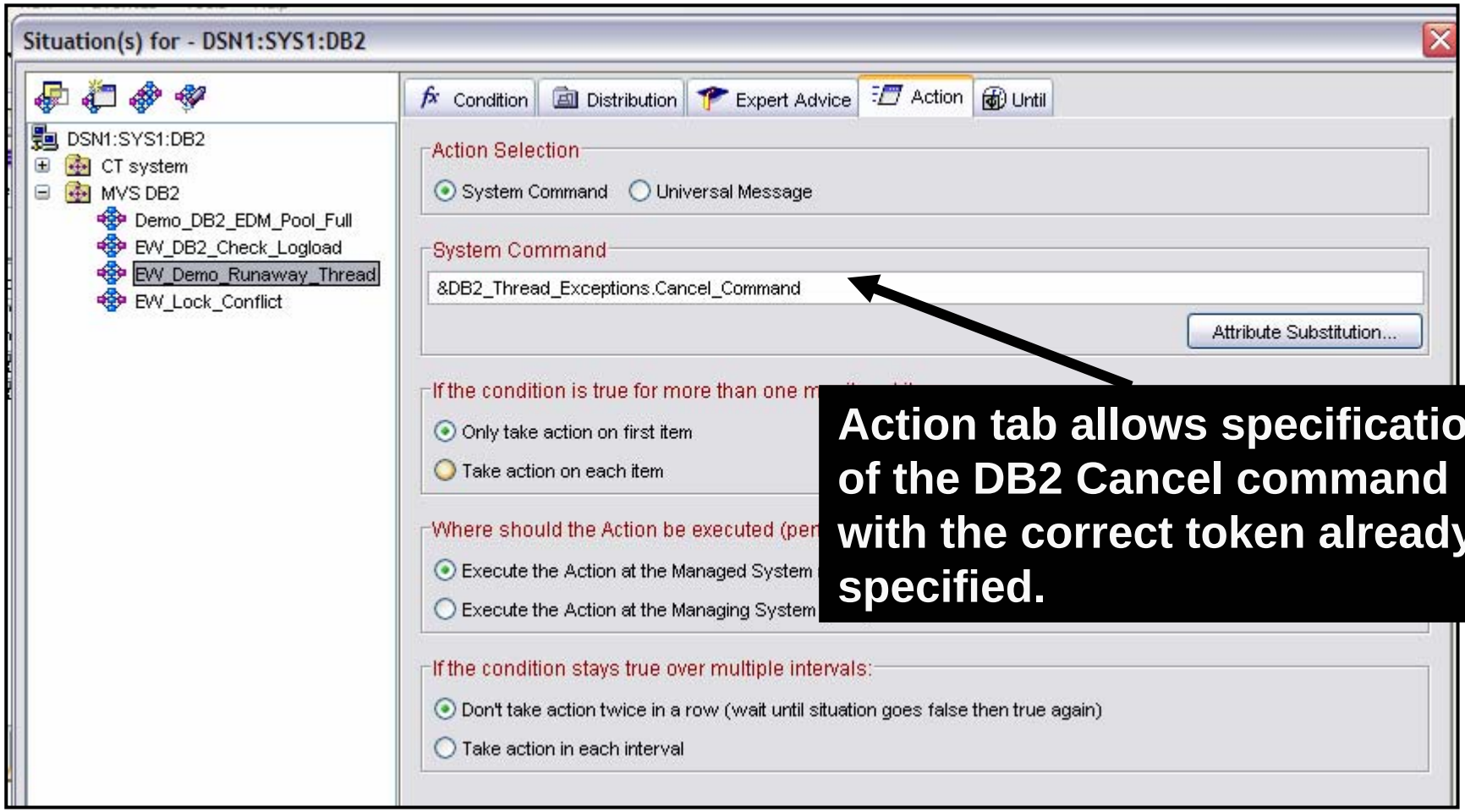
Performance Automation
Automated Corrections

- Monitor problem applications on an ongoing basis
  - Example - Monitor for runaway threads
    - Automate the termination of runaway threads
    - Automated ‘kill’ capability
- Use intelligent situation logic to target problem applications
- Monitor for subsystem issues and automate corrective actions
- No Rexx code or procedural language required

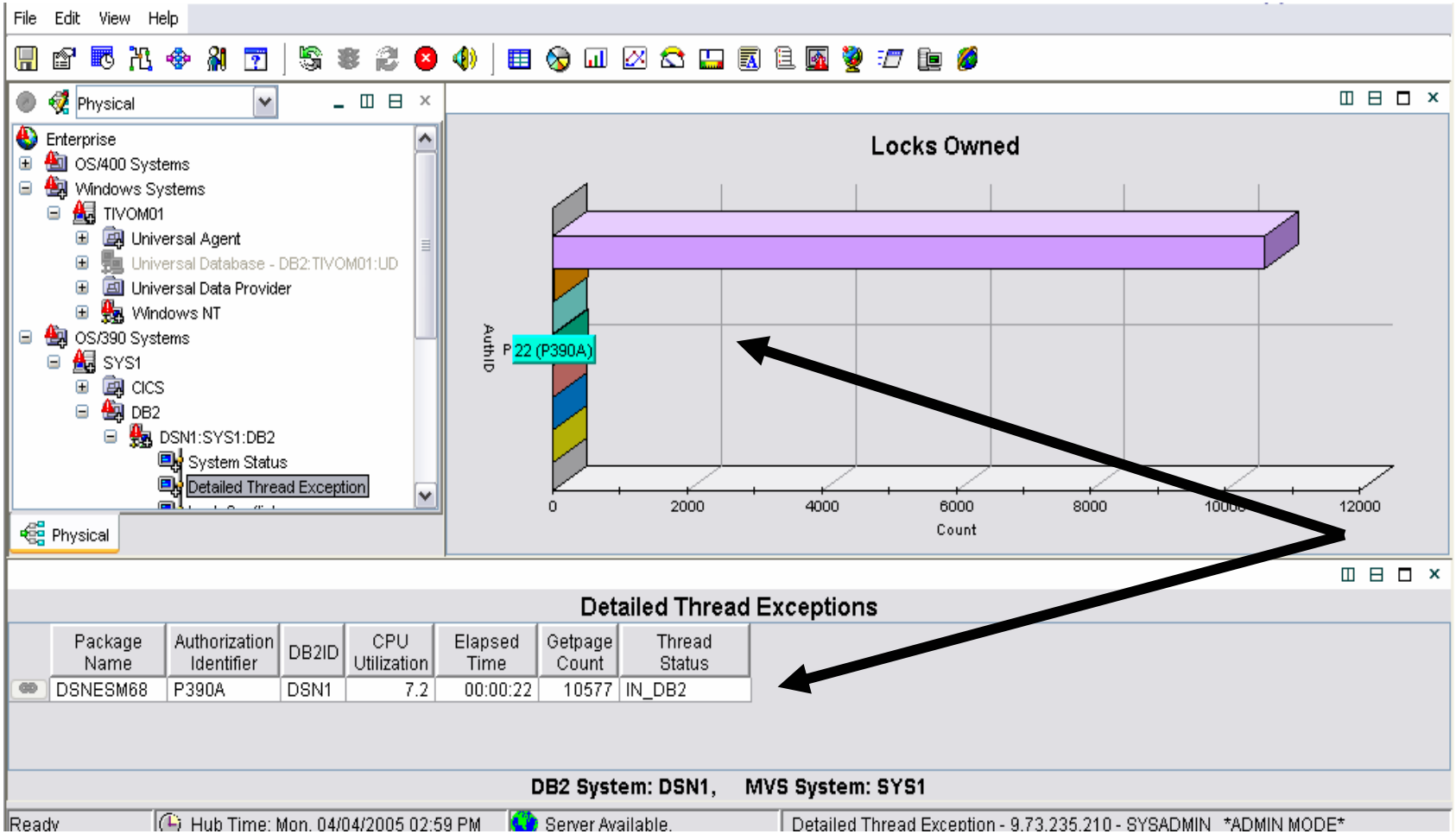
Automated Corrections
Runaway Thread Scenario



Automated Corrections
Specifying The Cancel Command



Monitoring The Problem Thread



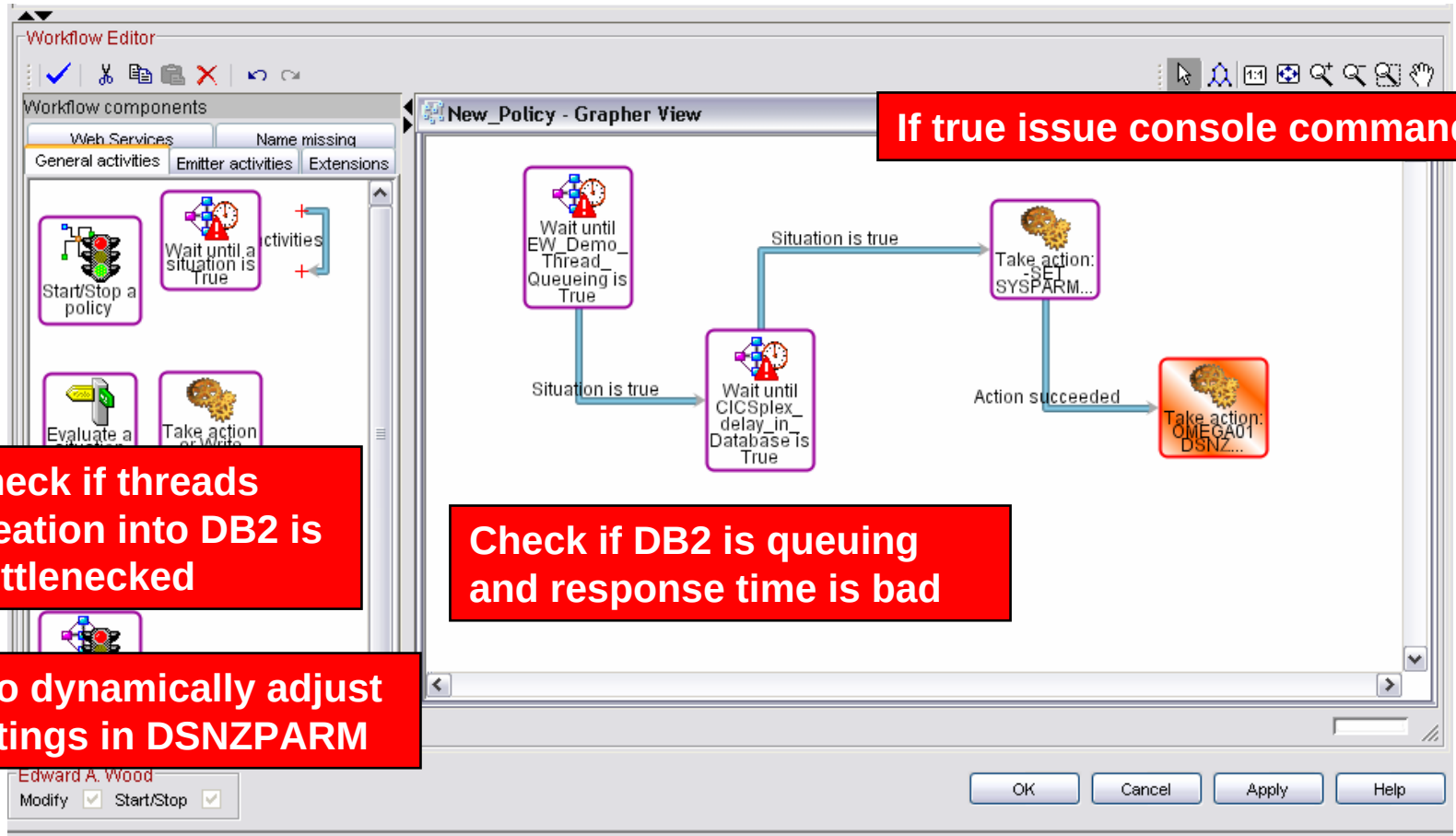
Automated Corrections
The Cancel Command Is Issued

```
  Display  Filter  View  Print  Options  Help
 -------------------------------------------------------------------------------
 SDSF SYSLOG     12.101 SYS1 SYS1 04/04/2005 2W   32267      COLUMNS  38 117
 COMMAND INPUT ===> _                                       SCROLL ===> CSR
STC00625 00000090  CSV002I REQUESTS FOR MODULE KPDCSVG  EXCEED MAXIMUM USE COUNT
STC00625 00000090  CSV002I REQUESTS FOR MODULE KPDCSVG  EXCEED MAXIMUM USE COUNT
STC00625 00000090  CSV002I REQUESTS FOR MODULE KPDCSVG  EXCEED MAXIMUM USE COUNT
STC00625 00000090  CSV002I REQUESTS FOR MODULE KPDCSVG  EXCEED MAXIMUM USE COUNT
STC00625 00000090  CSV002I REQUESTS FOR MODULE KPDCSVG  EXCEED MAXIMUM USE COUNT
STC00625 00000090  CSV002I REQUESTS FOR MODULE KPDCSVG  EXCEED MAXIMUM USE COUNT
STC00625 00000090  CSV002I REQUESTS FOR MODULE KPDCSVG  EXCEED MAXIMUM USE COUNT
STC00625 00000090  CSV002I REQUESTS FOR MODULE KPDCSVG  EXCEED MAXIMUM USE COUNT
STC00625 00000090  CSV002I REQUESTS FOR MODULE KPDCSVG  EXCEED MAXIMUM USE COUNT
STC00625 00000090  CSV002I REQUESTS FOR MODULE KPDCSVG  EXCEED MAXIMUM USE COUNT
STC00625 00000290  -  CANCEL THREAD(556)
STC00023 00000090  DSNV426I  - DSNVCT THREAD '556' HAS BEEN CANCELED
STC00023 00000090  DSN3201I  - ABNORMAL EOT IN PROGRESS FOR USER=P390A 855
     855 00000090  CONNECTION-ID=TSO CORRELATION-ID=P390A JOBNAME=P390A ASID=004
     855 00000090  TCB=008E1798
5 DFS996I *IMS READY*  IVP1
3 ISTEXC200 - DYN COMMANDS MAY BE ENTERED
******************************** BOTTOM OF DATA ********************************
```

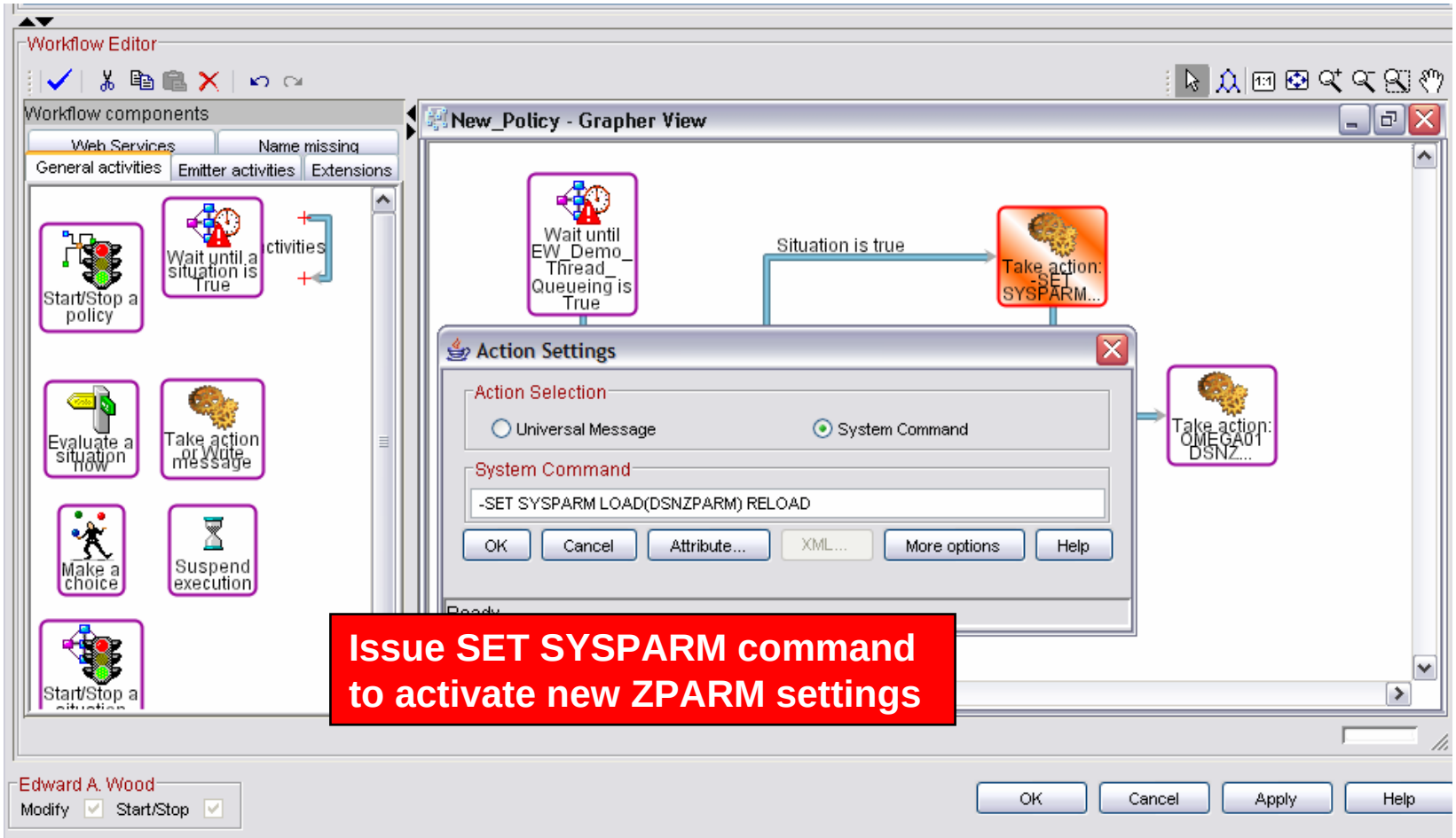
Performance Automation And Policies

- Use Situations for simpler “fire and forget” type of scenarios
- Use Policies for more sophisticated performance automation scenarios
- Automate corrections at machine speed
  - Implement machine speed corrective actions, issue alerts, and allow for later human intervention
- Use for dynamic subsystem management and ‘tweaks’ as the workload changes
  - Not permanent fixes, but to keep the workload running
- Policies allow for correlated automation of composite applications

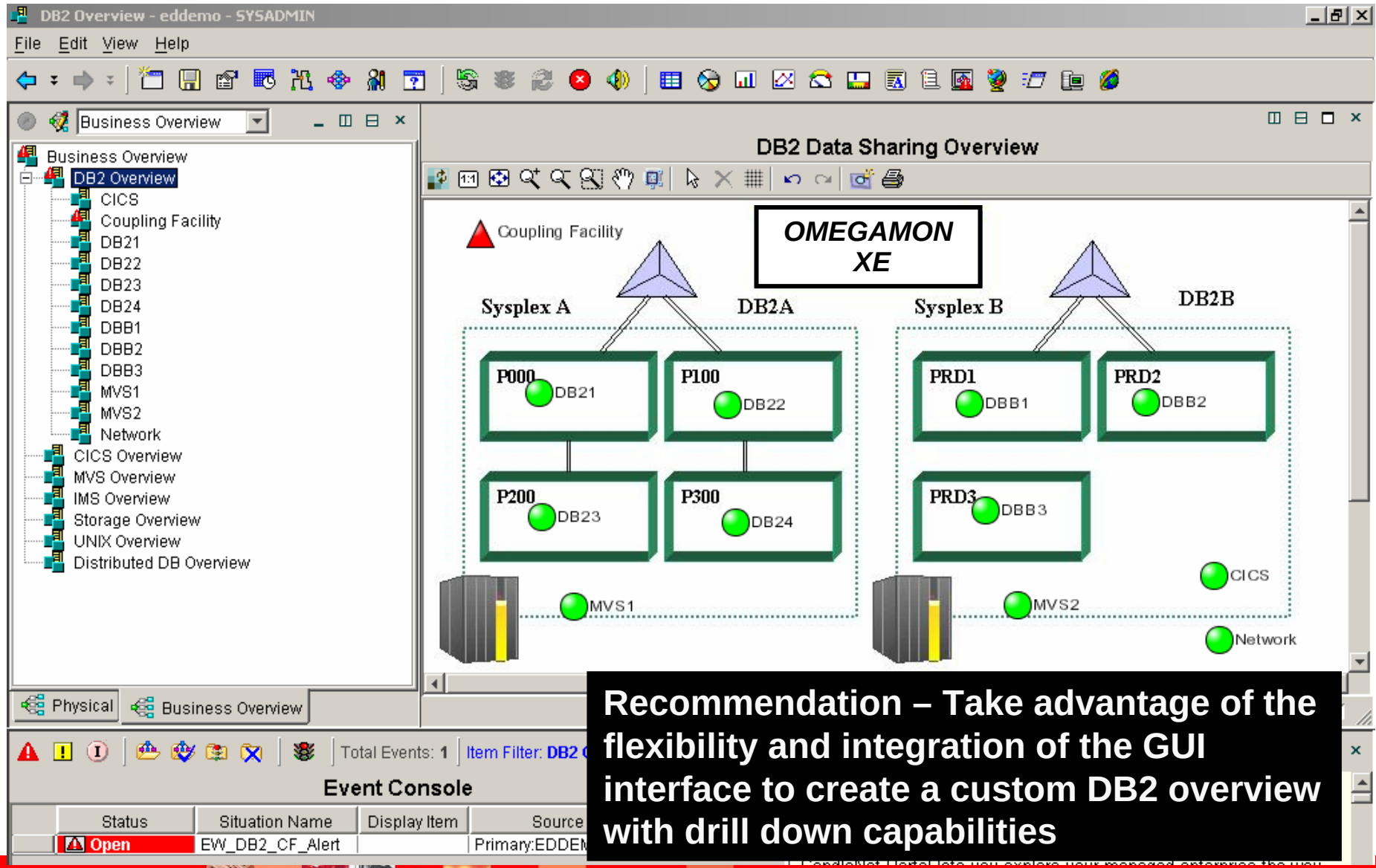
Using Policies For Dynamic Performance Management



Using Policies For Dynamic Performance Management



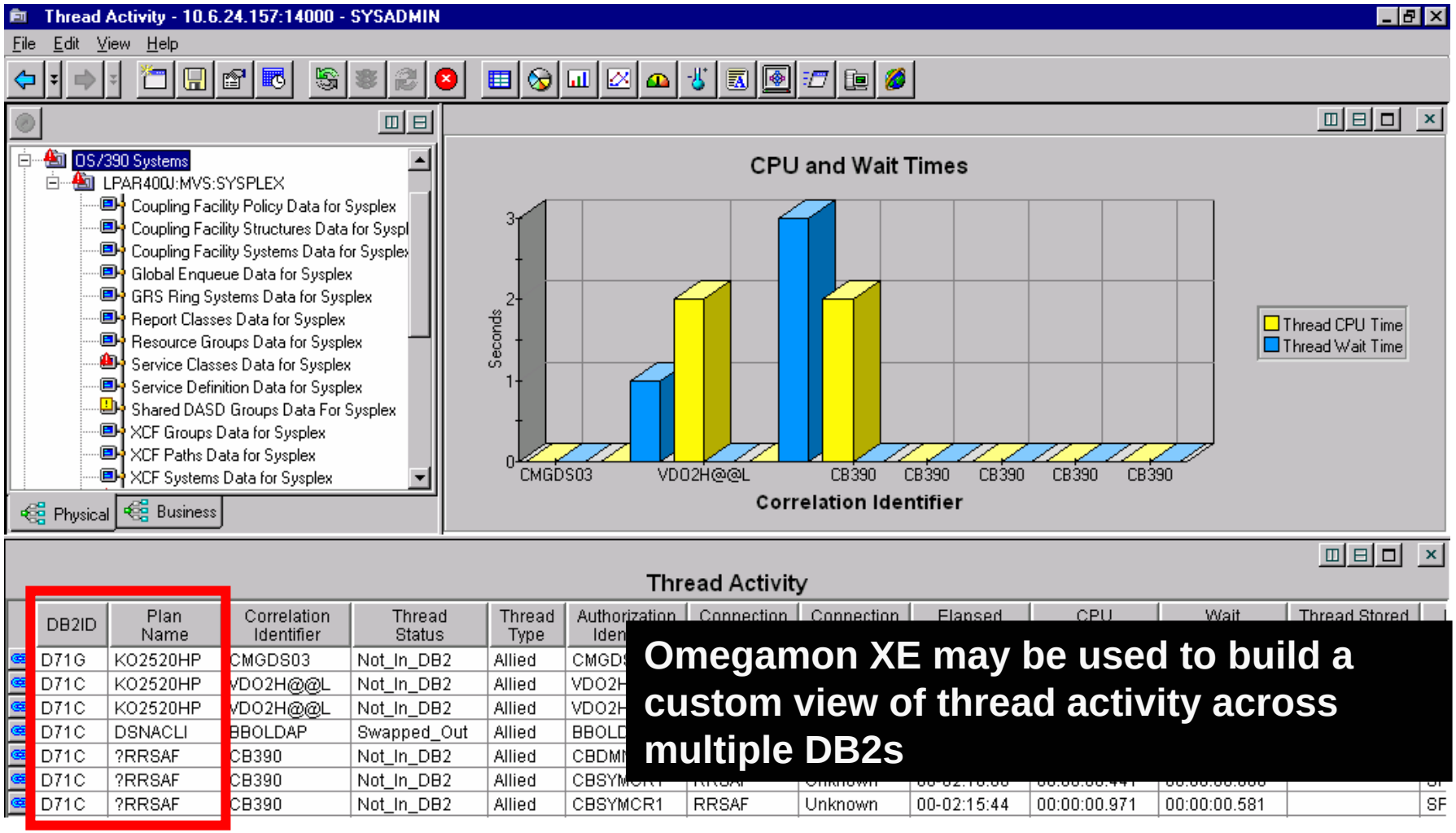
Use Omegamon XE GUI To Build Graphic Overviews



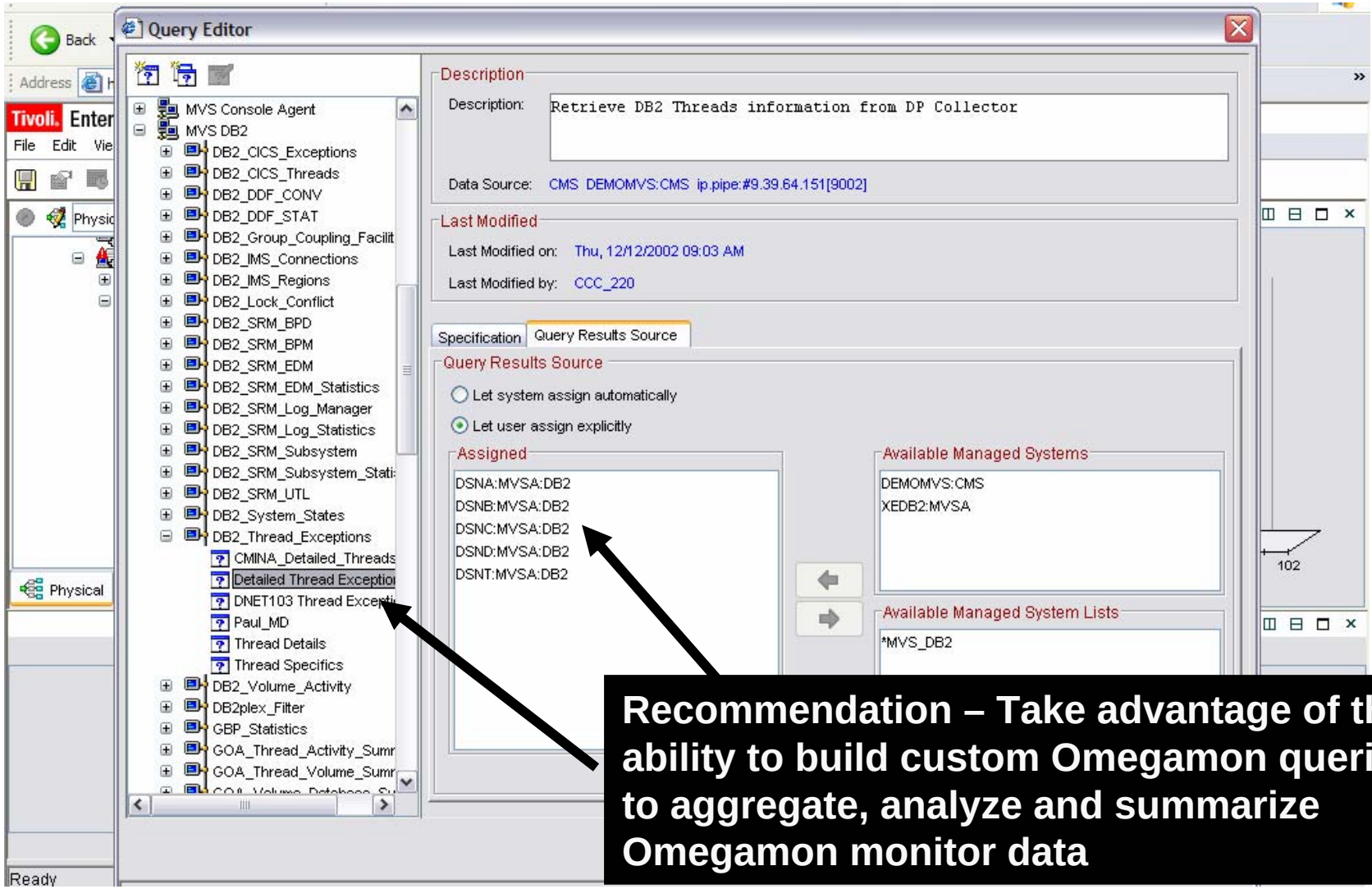
Recommendation – Take advantage of the flexibility and integration of the GUI interface to create a custom DB2 overview with drill down capabilities

OMEGAMON XE For DB2

Use The GUI To See A Global View Of DB2 Activity

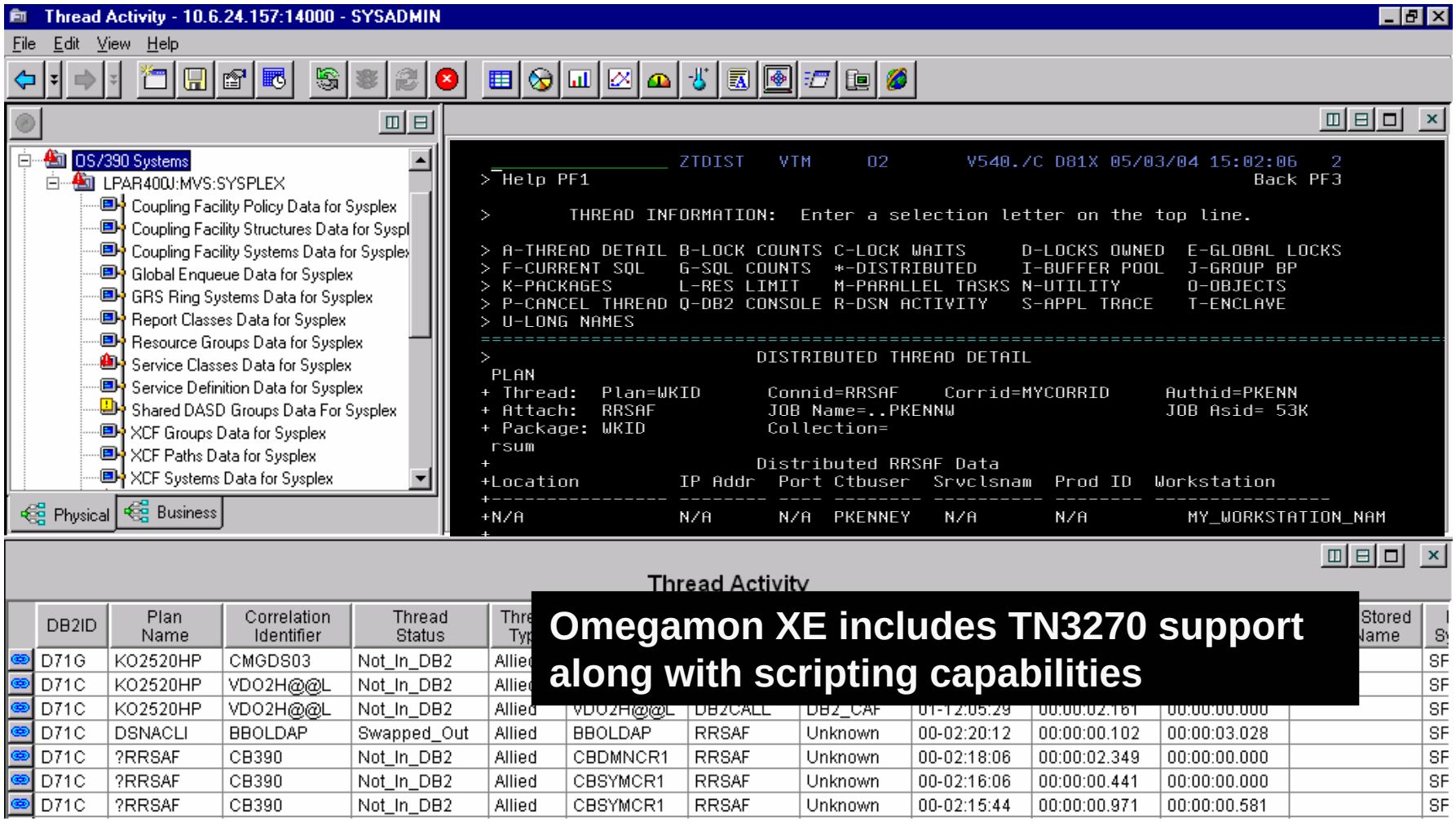


Create Custom Queries To Build Aggregate Views

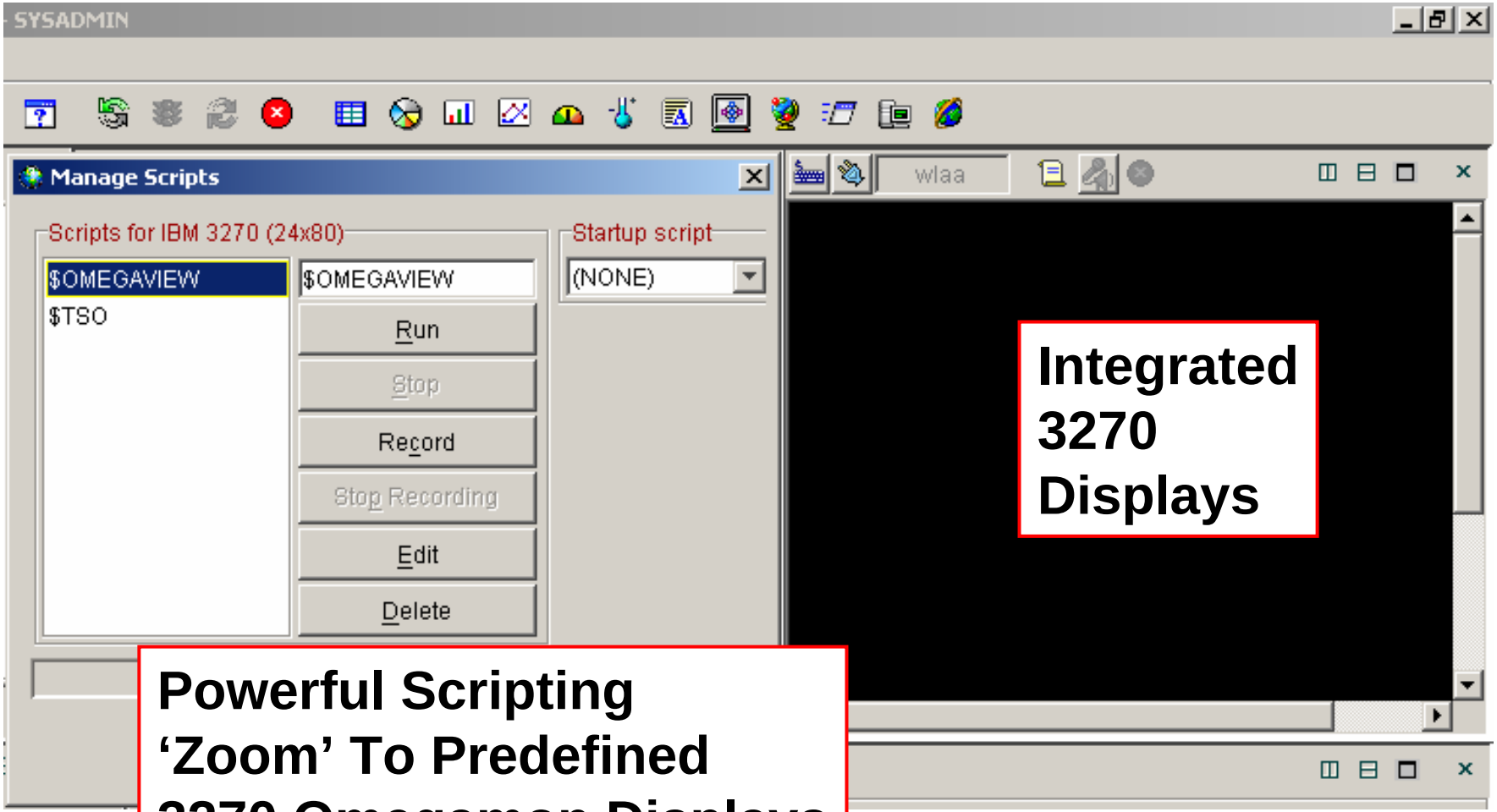


Recommendation – Take advantage of the ability to build custom Omegamon queries to aggregate, analyze and summarize Omegamon monitor data

Integrate 3270 Interface Into XE GUI Displays



Integrate 3270 Into The XE GUI Views



OMEGAMON DB2
Classic 3270 Interface Main Menu
Use 3270 For Detailed Analysis

```
 _________________ ZMENU    VTM     O2       V540./C D81G 02/22/05  6:51:28   2
>  Help/News/Index PF1               Exit PF3                    PF Keys PF5
>
>    Type a selection letter at the left end of the top line and press ENTER.
>
================================================================================
>         OMEGAMON II FOR DB2 CLASSIC INTERFACE -- REALTIME MAIN MENU

_ S  SUMMARY ............. Summary of DB2 activity
_ E  EXCEPTIONS .......... Current or potential system problems
_ T  THREAD ACTIVITY ..... Thread activity information
_ U  THREAD ACTIVITY ..... Thread activity information by Package
_ L  LOCKING CONFLICTS .... Locking conflict information
_ R  RESOURCE MANAGERS .... Resource manager, other DB2 subsystem information
_ A  APPLICATION TRACE .... Trace and view application activity
_ D  DISTRIBUTED DATA ..... Distributed database system information
_ O  OBJECT ANALYSIS ..... Object and Volume information
_ C  MVS CONSOLE ......... MVS console to issue commands and view messages
_ B  DB2 CONSOLE ......... DB2 console to issue
_ M  MISCELLANEOUS ....... Address space informat
_ P  PROFILE ............. Customize OMEGAMON se
_ H  HISTORICAL .......... Online historical inf
_ I  IFCID TRACE ......... Start an IFCID Trace
_ Z  OTHER DB2 ........... Redirect monitoring t
```

Use letter commands to navigate
Select option T to see real time thread activity

Thread Activity Overview

```
 _________________ ZALLT    VTM     O2       V540./C D81G 02/22/05  7:12:19   2
> Help PF1      Back PF3       Up PF7      Down PF8     Sort PF10    Zoom PF11
> T.A
>         THREAD ACTIVITY:  Enter a selection letter on the top line.

> *-ALL   B-TSO   C-CICS   D-IMS       E-BACKGROUND  F-DIST ALLIED  G-DIST DBAC
> H-UTIL  I-INACT J-FILTER K-FUNCTIONS L-STORED PROC M-TRIGGERS     N-SYSPLEX
> O-ENCLAVES
================================================================================
>                        ALL THREADS CONNECTED TO DB2
 THDA
+ *
+ Elapsed      Planname  CPU    Status       GetPg   Update  Commit  Jobname
+ ----------   --------  -----  ------------ ------  ------  ------  --------
+ 02-01:29     KO2520IF  00.0%  NOT-IN-DB2        0       0       0  CXEGA03
+ 02-01:29     KO2520HP  00.0%  NOT-IN-DB2        0       0       0  CXEGA03
+ 02-01:23     KO2520IF  00.0%  NOT-IN-DB2        0       0       0  CXEGA03
+ 02-01:23     KO2520IF  00.0%  NOT-IN-DB2        0       0       0  CXEGA03
+ 01:52:49.6   KO2520IF  00.0%  NOT-IN-DB2        0       0       0  CXEGA03
+ 00:38:44.9   TRANSPLX  00.0%  NOT-IN-DB2     1230     300       0  CCCDS18
+ 00:00:05.7   DEMO1     00.0%  NOT-IN-DB2        4       0       0  DEMOENCL
--------------------------------------------------------------------------------
```

Note high Getpage counts and high CPU%

The thread display may be sorted
To view a specific thread position the cursor and press F11

Example – Analyze Thread Lock Detail

```
 _________________ ZLOCKO   VTM     O2       V540./C D81G 02/22/05  6:39:37   2
> Help PF1               Back PF3              Up PF7                Down PF8

>       THREAD INFORMATION:  Enter a selection letter on the top line.

> A-THREAD DETAIL B-LOCK COUNTS C-LOCK WAITS     *-LOCKS OWNED  E-GLOBAL LOCKS
> F-CURRENT SQL   G-SQL COUNTS  H-DISTRIBUTED    I-BUFFER POOL  J-GROUP BP
> K-PACKAGES      L-RES LIMIT   M-PARALLEL TASKS N-UTILITY      O-OBJECTS
> P-CANCEL THREAD Q-DB2 CONSOLE R-DSN ACTIVITY   S-APPL TRACE   T-ENCLAVE
================================================================================
>                       LOCKS/CLAIMS OWNED BY A THREAD
 PLAN
+ Thread:  Plan=TRANSPLX  Connid=CCCDS18  Corrid=POOLDB210001 Authid=MHANS
+ Attach:  CICS   JOB=CCCDS18   Tran=DB21  Task#=00127  Term=M485  Type=POOL
+ Package: CICSDB21       Collection=
 own
+                         Lock Ownership Information
+      Percent NUMLKUS       = .00        Total  Locks Owned        = 5
+      Total Catalog Locks   = 1          Pageset and Dataset Locks = 1
+      Catalog Pageset Locks = 1          Page/Row Locks            = 0
+      Catalog Page/Row Locks= 0          Directory and Other Locks = 3
+      Bind ACQUIRE option   = USE        Bind RELEASE option       = COMMIT
+      ISOLATION option      = Cursor Stability
+
+      Type    Level        Resource                                 Number
+      ----    -----        ---------------------------------------- ------
+      DTBS        S        DB=DSNDG07                                    1
+      PSET        S        DB=DSNDG07        PS=DSN4K01                  1
+                 IS        DB=DSNDB06        PS=SYSDBASE                 1
+                 IS        DB=DSNDB01        PS=SCT02                    1
+      SKCT        S        Plan=TRANSPLX                                 1
+                                                                    ------
+                                                          Total =       5
+                         Claim Information
+      Type       Class     Resource
+      -------    -----     ------------------------------------------
+      IX         CS        DB=DSNDB06        PS=DSNDCX01
```

See thread level lock activity and locking detail in real time

Enter other letter commands on the command line to view more thread detail

Example
Viewing The Currently Executing SQL Statement

```
 _________________ ZSQL     VTM     O2       V540./C D81G 02/22/05  6:47:38   2
> Help PF1                                                          Back PF3

>       THREAD INFORMATION:  Enter a selection letter on the top line.

> A-THREAD DETAIL B-LOCK COUNTS C-LOCK WAITS     D-LOCKS OWNED  E-GLOBAL LOCKS
> *-CURRENT SQL   G-SQL COUNTS  H-DISTRIBUTED    I-BUFFER POOL  J-GROUP BP
> K-PACKAGES      L-RES LIMIT   M-PARALLEL TASKS N-UTILITY      O-OBJECTS
> P-CANCEL THREAD Q-DB2 CONSOLE R-DSN ACTIVITY   S-APPL TRACE   T-ENCLAVE
================================================================================
>                            SQL CALL BEING EXECUTED
 PLAN
+ Thread:  Plan=TRANSPLX  Connid=CCCDS18  Corrid=POOLDB210001 Authid=MHANS
+ Attach:  CICS   JOB=CCCDS18   Tran=DB21  Task#=00127  Term=M485  Type=POOL
+ Package: CICSDB21       Collection=
 call
+      SQL call is active, call information is as follows :
+
+      Thread Status  = WAIT-REMREQ    SQL Request Type     = STATIC
+      Total SQL Reqs =   274910       SQL Call Type        = FETCH
+      SQL DBRM Name  = CICSDB21       SQL Statement Number = 00073
+
+      DECLARE CUR1 CURSOR FOR SELECT I . DBNAME , I . TBCREATOR , I . TBNAME ,
+       I . NAME , I . CLUSTERING , I . CLUSTERRATIO , I . UNIQUERULE , I . FIR
```

Hit enter to watch screen refresh

Press F8 to see additional SQL text detail

Thread Buffer & I/O Analysis

```
 _________________ ZBUF     VTM     O2       V540./C D81G 02/22/05  6:48:16   2
> Help PF1              Back PF3                 Up PF7                 Down PF8

>       THREAD INFORMATION:  Enter a selection letter on the top line.

> A-THREAD DETAIL B-LOCK COUNTS C-LOCK WAITS     D-LOCKS OWNED  E-GLOBAL LOCKS
> F-CURRENT SQL   G-SQL COUNTS  H-DISTRIBUTED    *-BUFFER POOL  J-GROUP BP
> K-PACKAGES      L-RES LIMIT   M-PARALLEL TASKS N-UTILITY      O-OBJECTS
> P-CANCEL THREAD Q-DB2 CONSOLE R-DSN ACTIVITY   S-APPL TRACE   T-ENCLAVE
================================================================================
>                          THREAD BUFFER POOL ACTIVITY
 PLAN
+ Thread:  Plan=TRANSPLX  Connid=CCCDS18  Corrid=POOLDB210001 Authid=MHANS
+ Attach:  CICS   JOB=CCCDS18   Tran=DB21  Task#=00127  Term=M485  Type=POOL
+ Package: CICSDB21       Collection=
 buf
+ Buffer Pool: BP0
+
+ Getpage Requests       =        494   Failed Getpage Requests    =          0
+ Synchronous Read I/O   =         21   Getpage/Read I/O           =      23.52
+ Page Updates           =        116   Seq Prefetch Requests      =         29
+ List Prefetch Requests =         29   Dynamic Prefetch Requests  =          0
+ Prefetch Pages Read    =          9   Pr
+ Hiperpool Reads        =          0   Fa
```

Look at getpage and I/O counts for threads showing high I/O wait times

Real time views of thread getpage activity broken out by buffer pool

Object Information

Object Analysis Option Shows Thread Getpage & I/O

```
 _________________ ZTOBJ    VTM     O2       V540./C D81G 02/22/05  6:49:24   2
> Help PF1                                                          Back PF3

>       THREAD INFORMATION:  Enter a selection letter on the top line.

> A-THREAD DETAIL B-LOCK COUNTS C-LOCK WAITS     D-LOCKS OWNED  E-GLOBAL LOCKS
> F-CURRENT SQL   G-SQL COUNTS  H-DISTRIBUTED    I-BUFFER POOL  J-GROUP BP
> K-PACKAGES      L-RES LIMIT   M-PARALLEL TASKS N-UTILITY      *-OBJECTS
> P-CANCEL THREAD Q-DB2 CONSOLE R-DSN ACTIVITY   S-APPL TRACE   T-ENCLAVE
================================================================================
>                              OBJECTS USED BY THREAD
 PLAN
+ Thread:  Plan=TRANSPLX  Connid=CCCDS18  Corrid=POOL
+ Attach:  CICS   JOB=CCCDS18   Tran=DB21  Task#=0012
+ Package: CICSDB21       Collection=
 tobj
+                                                    Sync      ------ Prefetch I/O -----
+ Database  Spacenam  Dsn  Volume   Getpage          Read        Seq       List   Dynamic
+ --------  --------  ---  ------   -------          -------   -------   -------   -------
+ DSNDB06   DSNDCX01  001             117               0         0         0         0
+ DSNDB06   SYSDBASE  001              13               0         0         0         0
+ DSNDG07   DSN4K01   001              78               0         0         0         0
================================================================================
```

Note Getpage and I/O information for the thread

Getpages

I/Os

Application Trace Facility (ATF)

- Ability to selectively trace the execution of DB2 applications
- This information will help in your analysis of application flow and resource consumption
- ATF displays several types of information related to the execution of the thread
  - SQL access path at execution time
  - Pageset access and scan information
  - Sort activity
  - Locking information
  - DB2 times by thread, unit of work, program and SQL statement
- Uses DB2 performance traces to gather application information

ATF – Collection Options

```
 _________________ ZATRQ    VTM     O2       V540./C DB8G 08/09/05  7:39:44   2
>         Help PF1                                              Back PF3
> A.A
> *-SPECIFY TRACE     B-VIEW TRACE       C-STOP TRACE       D-SELECT DSN
> E-VIEW DATASET      F-STOP VIEW        G-CREATE VSAM LDS
================================================================================
>                          SPECIFY APPLICATION TRACE
 ATRQ
+ Type DB2 Plan name to be traced. Also, provide additional optional
+ selection information to limit trace output. To save trace records
+ for later viewing you must specify a data set name for DSN
+
:     DSN=_____________________________________________      Data set name
:     TIME=005             Number of mins to trace (001-060)
:     PLANNAME=_________   Plan name or ALL for all active threads
:     AUTHID=_________     DB2 authorization identifier
:     TSOUSER=_________    TSO USERID (TSO foreground app)
:     JOBNAME=_________    Jobname (TSO batch app)
:     CICSTRAN=____        CICS trans id
:     CICSCONN=_________   CICS connection id)
:     PSBNAME=_________    IMS PSB name
:     IMSID=_________      IMS ID of the IMS region
:     LOCKDATA=Y           Collect DB2 lock trace recs? (Y/N)
:     SCANDATA=Y           Collect DB2 scan trace recs? (Y/N)
```

Trace to memory

- Trace data stored in OMEGAMON memory (up to 4MB)
- When user exits OMEGAMON data goes to the ‘bit’ bucket

Trace to Dataset

- Trace data stored in VSAM data set
- Use for later trace analysis and reports
- If no DSN specified, trace data is sent to memory

Application Trace Facility
Event Detail

```
 _________________ ZATD3    VTM     O2       V540./C D81G 02/22/05  7:02:02   2
> Help PF1              Back PF3               Up PF7                 Down PF8

>         APPLICATION TRACE:  Enter a selection letter on the top line.

> A-PROGRAM    B-SQL INDEX    C-SQL DETAIL    D-LOCK DETAIL    *-EVENT DETAIL
================================================================================
>                        APPLICATION TRACE EVENT DETAIL
 ATD3
+ Planname=TRANSPLX   Connid=CCCDS18    Corrid=POOLDB2
+
: Control=NEXT    (FIRST/LAST/NEXT/PREV/+nnnn/-nnnn/Sn
+ Current=000001  Total Number of SQL Calls=008207
+
+ Event Time   TN Event Type            Event Resource Information
+ ------------ -- --------------------- ------------------------------------
+ 06:51:14.954    START OPEN CURSOR     PGM=CICSDB21 STMT=00070
+ 06:51:14.954    START OF SORT
+ 06:51:14.955    START INDEX SCAN      DB=DSNDB06   PS=DSNDXX03
+ 06:51:14.955    LOCK ACQUIRE          DB=DSNDB06   PS=SYSDBASE DPAG S   D=MNL
+                                       PAGE=0000FD
+ 06:51:14.955    START INDEX SCAN      DB=DSNDB06   PS=DSNDKX01
+ 06:51:14.955    END OF INDEX SCAN     DB=DSNDB06   PS=DSNDKX01
+                                       ROWS=0000022 PAGES=0000000
```

Note individual event detail. Start/Stop trace records, with elapsed time

Omegamon 3270
Buffer Pool Snapshot Detail Analysis

```
 _________________ ZBPSN    VTM     O2       V540./C D81G 02/23/05 19:41:33   2
> Help PF1     Back PF3     Up PF7     Down PF8     Sort PF10     Zoom PF11
>
> A-BUFFER POOL    B-GROUP BUFFER POOL    *-BUFFER POOL SNAPSHOT    F-FILTER
==============================================================================
>                      BUFFER POOL SNAPSHOT OPEN PAGESETS
>
 BPSN 0
+ <<< The following BP snapshot was collected on  02/23/05 at 19:41:33.   >>>
+
+                                                  *
+ Pageset             Pageset      Use    Open   VP Pgs   HP Pgs   VP Pgs
+ Name                Type         Count  DS     Current  Current  Changed
+ ----------------    ----------   -----  ----   -------  -------  -------
+ DSNDB01.DBD01       TABLESPACE       0      1       35        0        0
+ DSNDB01.SYSLGRNX    TABLESPACE       0      1       19        0        0
+ DSNDB06.DSNSSX01    INDEXSPACE       0      1       17        0        0
+ DSNDB01.SCT02       TABLESPACE       0      1       14        0        0
+ DSNDB06.SYSSTR      TABLESPACE       0      1       13        0        0
+ DSNDB01.DSNLLX01    INDEXSPACE       0      1       12        0        0
+ DSNDG07.DSN4K01     TABLESPACE       0      1       11        0        2
+ DSNDB06.DSNAPH01    INDEXSPACE       0
+ DSNDB06.DSNDCX01    INDEXSPACE       1
                                       0
```

Use F10 Sort option to sort by VP pages descending

Look for high utilization objects within the pool. F11 zoom to see more detail.

Object Details

```
 _________________ ZBPSD    VTM     O2       V540./C D81G 02/23/05 19:44:08   2
>  Help PF1           Back PF3                 Up PF7                Down PF8
>
================================================================================
>                     BUFFER POOL SNAPSHOT DATASETS
>
 BPSD
+ BP: 0    Pageset Name: DSNDB01.DBD01     Type: TABLESPACE  Open Datasets:   1
+
+ Dataset Name: DB2C71.DSNDBC.DSNDB01.DBD01.I0001.A001
+
+ VP Pages Current        =       35     HP Pages Current             =         0
+ VP Pages Maximum        =     3100     HP Pages Maximum             =         0
+ VP Pages Changed        =        0     VP Pages Changed Maximum     =         0
+ Sync I/O Total Pages    =      189
+ Sync I/O Average Delay  =        3     Sync I/O Maximum Delay       =        19
+ Async I/O Average Delay =        0     Async I/O Maximum Delay      =         0
+ Async I/O Total Pages   =        0     Async I/O Total I/O Count    =         0
================================================================================
```

Note pages used current versus pages high water mark as an indication of thrashing.

Detail found in the 3270 interface is useful for Virtual Pool tuning and analysis

EDM Pool Snapshot

```
 _________________ ZEDSN    VTM     O2       V540./C D81G 02/23/05 20:37:24   2
>  Help PF1                    Back PF3                                Zoom PF11
>  R.C.A
================================================================================
>                          EDM POOL SNAPSHOT SUMMARY
>
 EDSN
+ <<< The following EDM snapshot was collected on 02/23/05 at 20:37:24.   >>>
+
+ EDM              % of       Pages      Count of     Avg Pages     Max Pages
+ Storage Type     Pool       Alloc      Entries      Entry         Entry
+ ------------     ------     -------    --------     ---------     ---------
+ DBDs                .9%        32.0           2          16.0           8.0
+ CTs                 .1%         5.0           5           1.0           1.0
+ PTs                 .0%          .0           0           .0            .0
+ SKCTs               .3%        10.5           5           2.1           4.0
+ CACHE               .0%          .5           2           .2            .2
+ SKPTs               .0%          .0           0           .0            .0
+ SQL CACHE           .0%          .0           0           .0            .0
+ FREE              98.7%      3655.0           1        3655.0        3655.0
================================================
```

F11 to see more detail.

The EDM pool snapshot shows more detail on EDM pool utilization

EDM Pool Detail

```
 _________________ ZEDSS    VTM     O2       V540./C D81G 02/23/05 20:38:13   2
>  Help PF1         Back PF3         Up PF7          Down PF8          Sort PF10
>
================================================================================
>                      EDM SNAPSHOT SKELETON CURSOR TABLES
>
 EDSS
+                    *
+                    Planname       Pages Alloc       Bytes Used
+                    --------       -----------       ----------
+                    DEMO1                  4.0             9176
+                    KO2520AP               1.0             1680
+                    KO2520HP               1.0             1680
+                    KO2520IF                .7             1696
+                    TRANSPLX               3.7            14800
================================================================================
```

This shows detail on the contents showing number of pages within the SKCT portion of the EDM pool

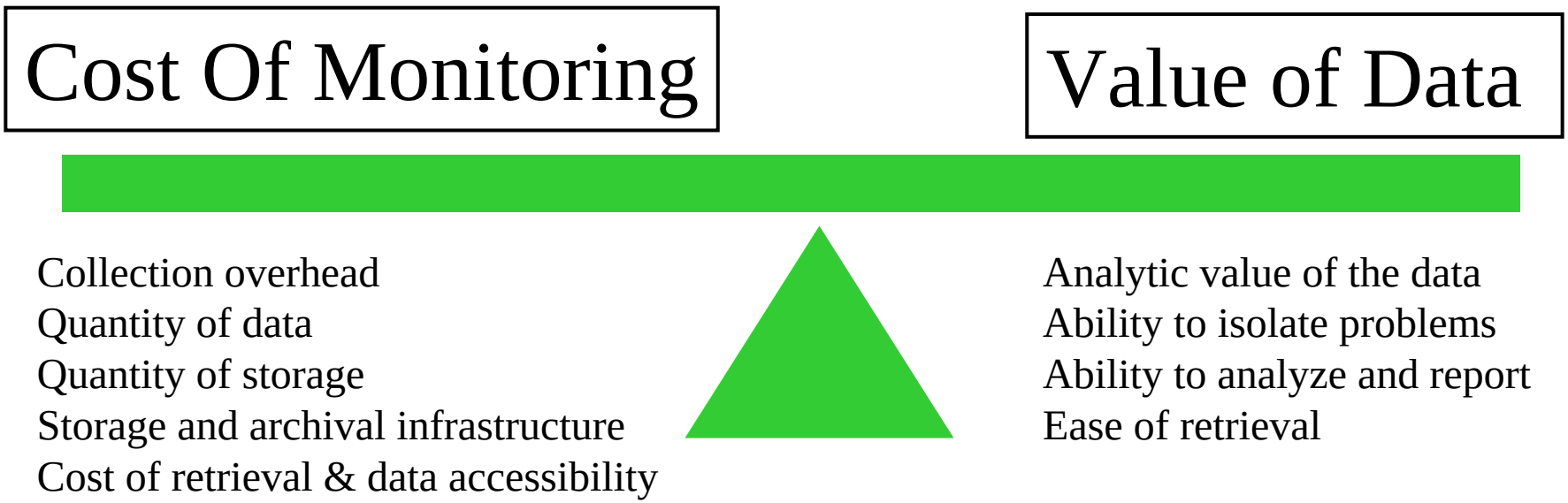
EDM Pool Snapshot Analysis – CUA Interface Example
Dynamic SQL Cache Details

```
 ____  Actions(A)  View(V)  Options(O)  Help(H)
 ______________________________________________________ 09/25/02 11:05:07 AM
 KD2RESCT        Dynamic SQL Cache Statistics                    System:DSN
                                          lines ___1  To   12 of  250
```

Times Exec.	CPU Time	Elapsed Time	Wait Time	Get-Pages	Sync Reads	Sync Writes
_ 408767	00:01:17.614	00:02:42.358	00:01:07.433	1435K	19266	0
_ 241909	00:00:49.065	00:02:38.719	00:01:43.748	1204K	57809	0
_ 182434	00:00:40.717	00:0				
_ 182434	00:01:1_.217	00:0				
_ 98307	00:00:31.77_	00:0				
_ 92177	00:00:13.477	00:0				
_ 39919	00:00:03.769	00:0				
_ 14976	00:00:00.368	00:0				
_ 14972	00:00:00.383	00:0				
_ 11158	00:00:03.304	00:0				
_ 10236	00:00:02.168	00:0				
_ 9753	00:00:04.416	00:0				

```
 ______________________________________________________ 09/25/02 11:05:28 AM
 KD2RESCE        SQL Cache Statistics Detail                      System:DSN

                                          lines ___1  To   15 of   20

    statistics require that monitor class 1 and ifcid 318 be started

          Authorization Id: $EC

 SELECT SID FROM SYSADM.TB_EC_SESSION WHERE SID = ? AND EXPIRES > CURRENT
  TIMESTAMP + 1770 SECONDS FOR FETCH ONLY

 Times Executed                 408997   Synchronous Buffer Reads      19287
 Getpages                      1435987   Rows Examined                817948
 Rows Processed                 397810   Sorts Performed                   0
 Index Scans                    408998   Tablespace Scans                  0
 Parallel Groups Created             0   Synchronous Writes                0
 Elapsed Time             00:02:42.459   CPU Time               00:01:17.660
 Wait for Synch I/O       00:01:00.096   Wait for Lock/Latch    00:00:07.318
 Synch Exec Switch        00:00:00.000   Wait for Global Locks  00:00:00.000
```

Historical Monitoring Considerations And Options

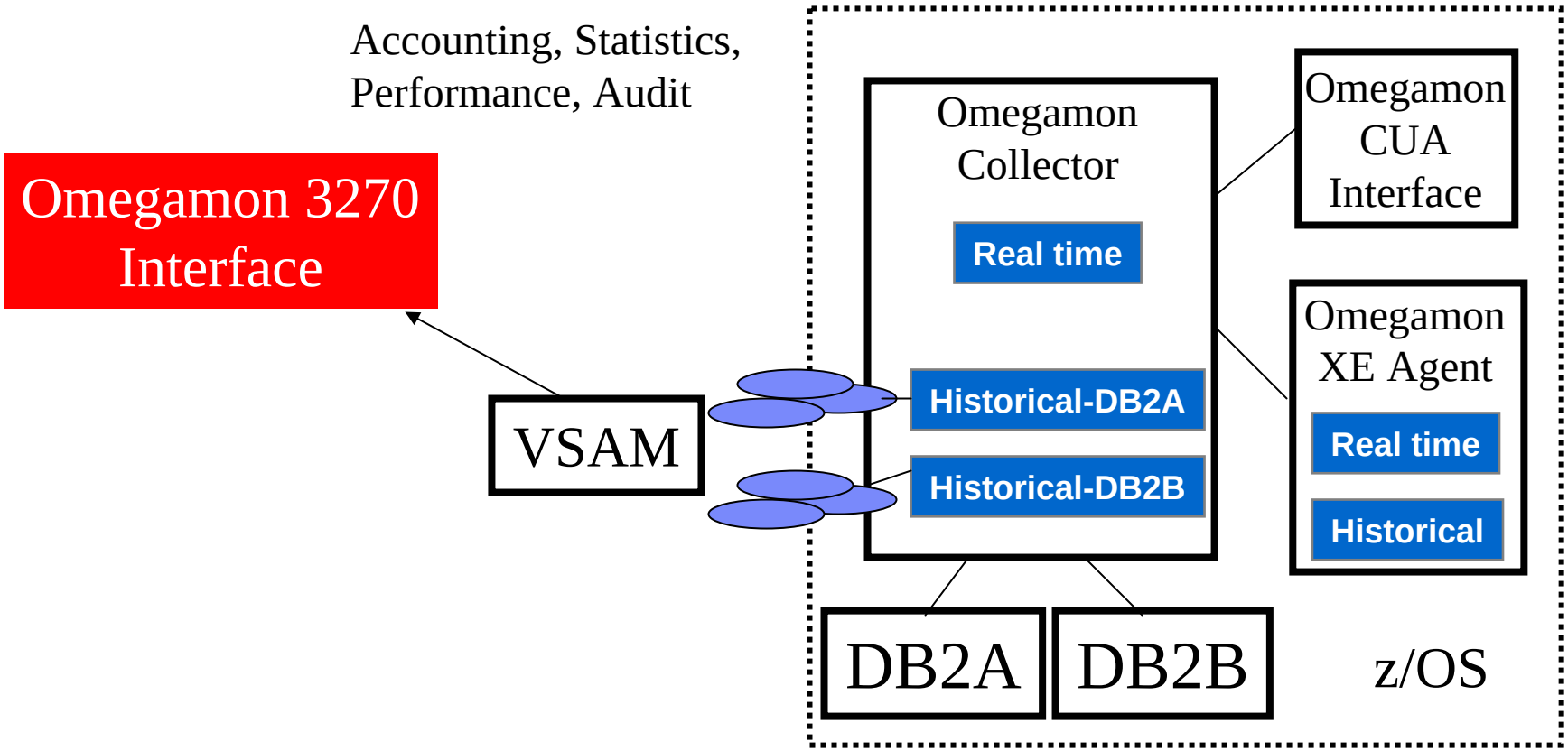


- Historical data reporting and analysis strategies should be built around the concept of the cost of gathering relative to the value of the data gathered

History Data Options
OMEGAMON Provides Historical Analysis Flexibility

- DB2 Accounting data
  - High data volume poses challenges from a gathering, retention, and reporting perspective
  - Trace overhead considerations, DSNZPARM options, CICS RDO options
  - Essential for application analysis
- DB2 Statistics
  - Low cost and quantity of data
  - Important for subsystem tuning - Should always gather this data
- DB2 Performance & Audit traces
  - Event based detail beyond that provided by other sources
  - Cost may be high if many traces and many events traced
  - Recommend using tactically
- Snapshot
  - Overhead typically in monitoring infrastructure gathering and storage
  - Recommend using tactically - don’t snapshot everything

OMEGAMON XE For DB2
Historical Data Gathering Capabilities And Flow
Near Term Historical Data



Near Term Thread History

Easy Access To History Within OMEGAMON Interface

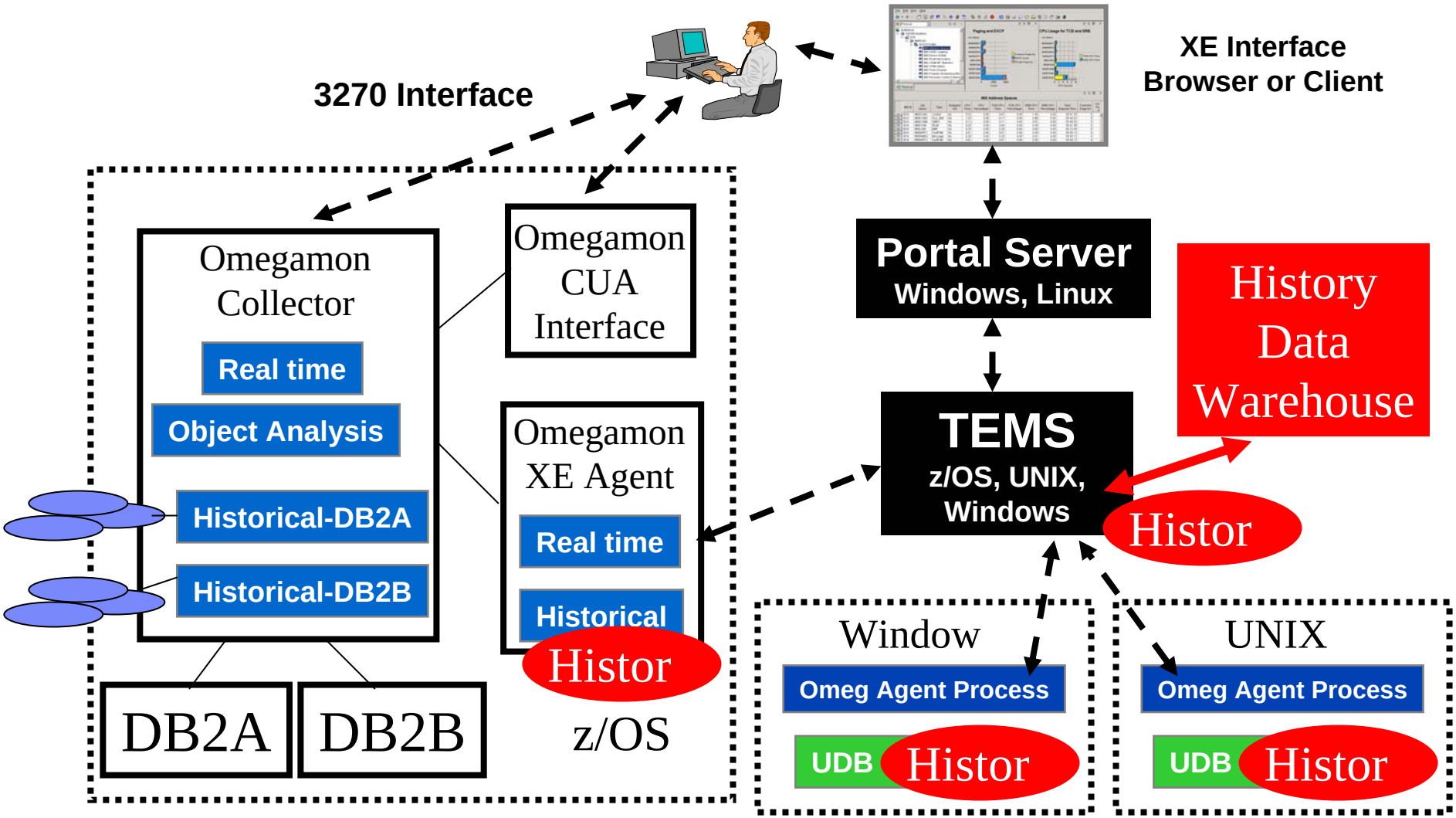
```
 _________________ ZHATACT  VTM     O2       V540./C D81G 02/22/05  7:38:00   3
> Help PF1          Back PF3         Up PF7          Down PF8          Zoom PF11
>
>                    Enter a selection letter on the top line.
>
> *-SUMMARY         B-BUFFER POOL        C-DB2 TIME         D-LOCK/SCAN/SORT
> O-OPTIONS
================================================================================
>                              THREAD HISTORY SUMMARY
 HATH
+ Report Interval:   15 mins                      Start: 02/22 07:15:00.000000
+ Report Filtered:       NO                          End: 02/22 07:29:59.999999
 act
+
+                                    Elapsed  CPU                              Term
+ End Time       Plan     Authid     Time     Time    SQL   Commit Abrt Pkg Status
+ ------------   -------- -------- -------  ------ ----- ------ ---- --- ------
+ 07:25:52.899   TRANSPLX MHANS     3138.42  44.758 1000K      1     0   1
+ 07:22:14.571   DEMO1    CXE12AUR  600.90     .011    11      1     0   1 EOT/AB
================================================================================
```

F11 to see more detail on a thread

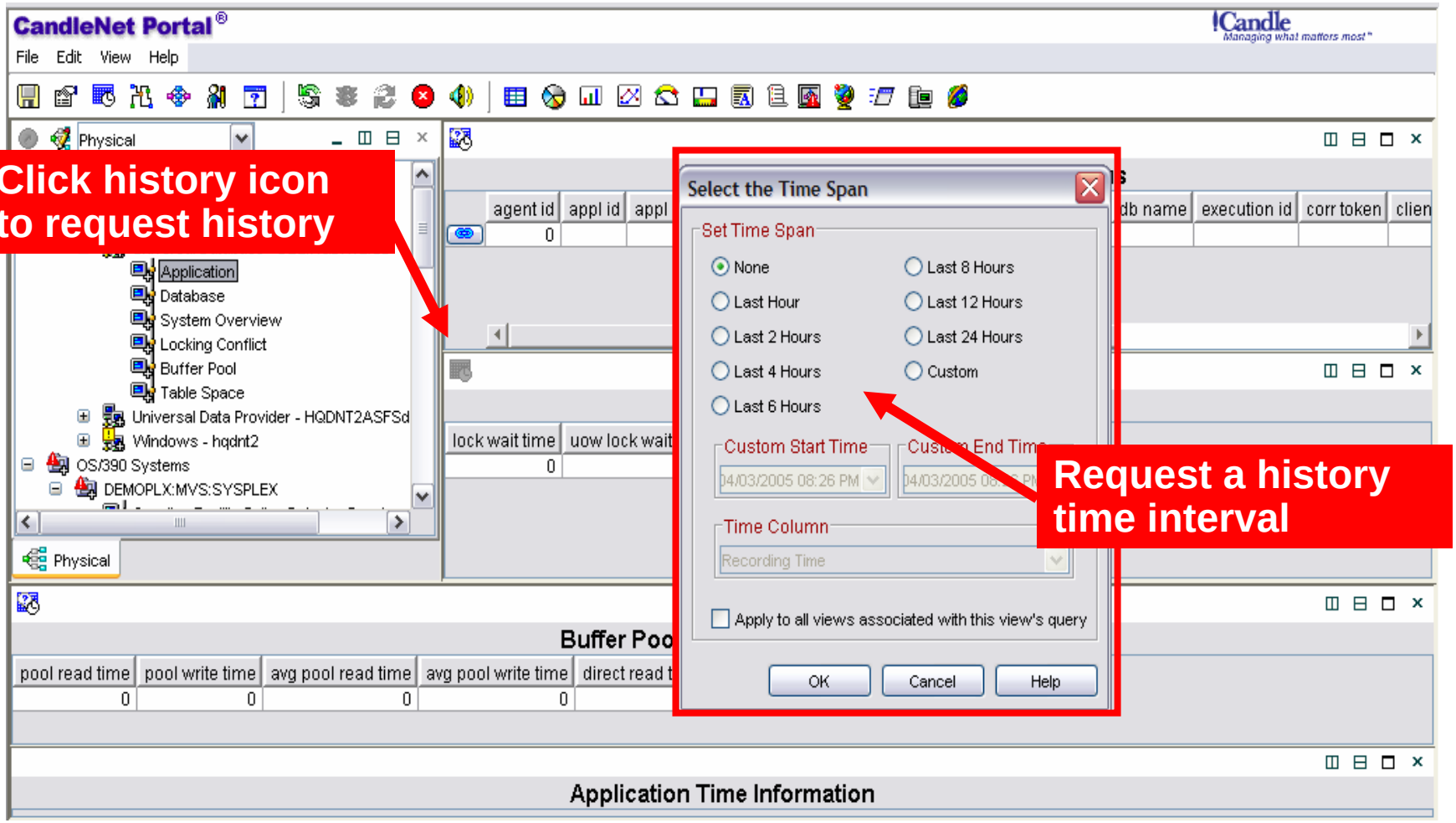
Omegamon allows thread history to be captured and reviewed from within the real time monitor. History is captured in VSAM files.

OMEGAMON XE For DB2

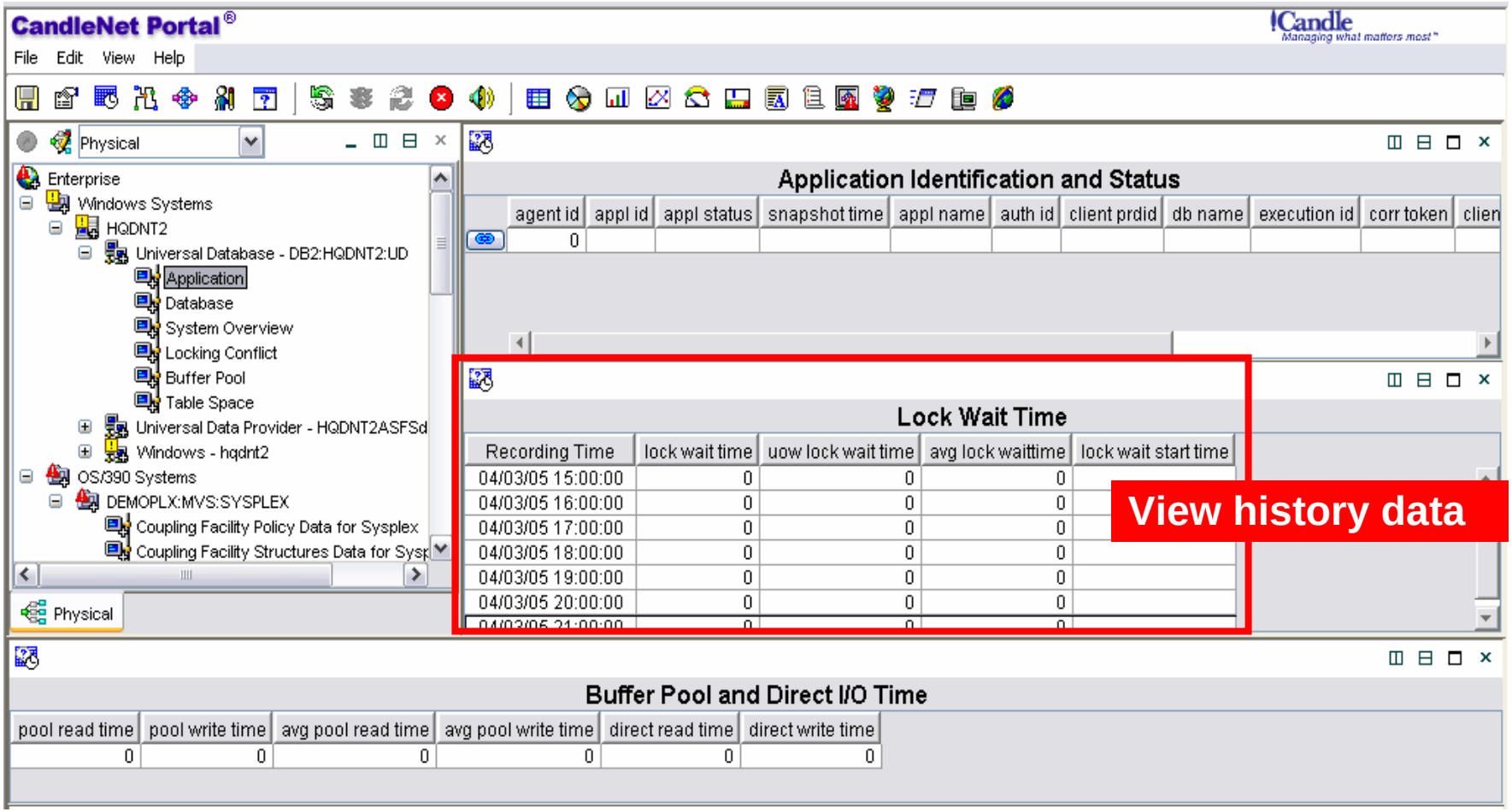
Integrated Historical Support For UDB on z/OS, UNIX, Windows



OMEGAMON XE
Historical Interface



History Integrated Within Real Time XE GUI Interface



Merging The Monitors

A Comprehensive DB2 Performance Monitoring Solution

OMEGAMON XE For DB2 PM
&
OMEGAMON XE For DB2 PE

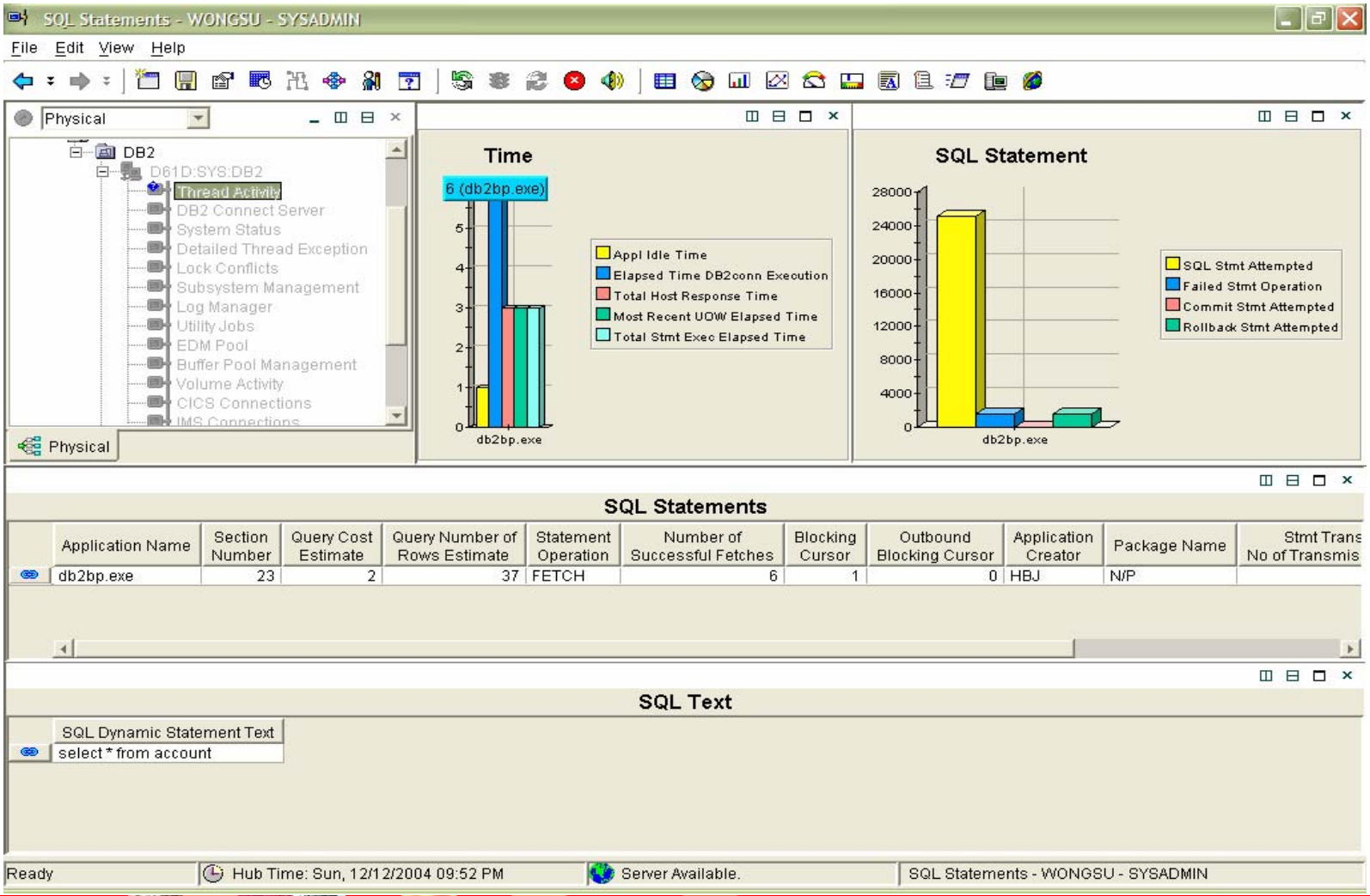
What OMEGAMON XE for DB2 customers get merging with DB2 PE/PM

- DB2 Connect monitoring
- DB2 PM detailed reports and PWH (*)
- Expert analysis using ROT and SQL queries
- Official DB2 IFI API support
- More granular snapshot history

What DB2 PE / PM customers get merging with OMEGAMON

- OMEGAMON near-term history
- Real time object analysis
- OMEGAMON classic 3270 VTAM end user interface
- OMEGAMON XE GUI interface
- One Central Server per LPAR
- The ability to integrate with MVS, CICS, IMS and distributed monitoring

Omegamon XE Connect Monitoring Example - SQL Information



Summary

- Omegamon offers options in terms of interfaces and capabilities
  - XE GUI Interface
    - High level overview, alerting, analysis, automation
  - 3270 Interface – Classic & CUA
    - Deep dive analysis
- Historical options
  - Omegamon Near Term History – 3270 Interface
  - XE GUI Interface snapshot historical
- Merger of monitors – a powerful combined solution
  - Omegamon XE for DB2/PM
  - Omegamon XE for DB2/PE

Thank You!!!