# Monitoring: Become More Proactive Through Effective Historical Analysis

Ed Woods
Consulting IT Specialist
IBM Corporation

Session 9633
Tuesday, August 9th
1:30-2:30 PM

# Agenda

- How can you use history to improve your real-time monitoring strategy?
- Are you getting the most from your investment in monitoring and management solutions?
- How can you leverage history to improve your overall performance and availability?
- What are the most effective ways to use history to solve common problems?
- What are the optimal ways to collect historical information?
- How can you use history to become more proactive with real-time monitoring and management?

# The Importance Of History Data
# Why History Can Be Essential To Your Overall Strategy

- Not all problems or events can be seen and analyzed in real time
  - Inevitably some analysis will need to be done after the fact using such functions as Near Term History, snapshot history, or report analysis
- History provides an understanding of what happened in the past
  - History of application performance and response time
    - CICS/IMS response time, DB2 thread activity, z/OS batch activity
  - History of resource utilization and resource issues
    - DASD, memory/paging, CPU, pools
  - History of alerts and issues
    - What alerts fired and how often
- History can be used to help visualize what may happen in the future
  - Analysis of the past to help anticipate potential future issues/bottlenecks
- Use history to make real-time monitoring more effective and meaningful
  - Use historical information to make real time alerts more accurate and relevant
  - Include history in custom real time workspaces

# Historical Data Collection Considerations


***Monitoring And History Collection Trade-off***

- Historical data collection varies in cost and quantity
  - CPU, memory, and software process cost of collection
  - Cost of data storage and retention
  - Cost of retrieval and post processing
  - Ease of review and analysis
- Some historical data will be more relevant and useful than other data
  - Consider the context, nature, and meaningfulness of the data

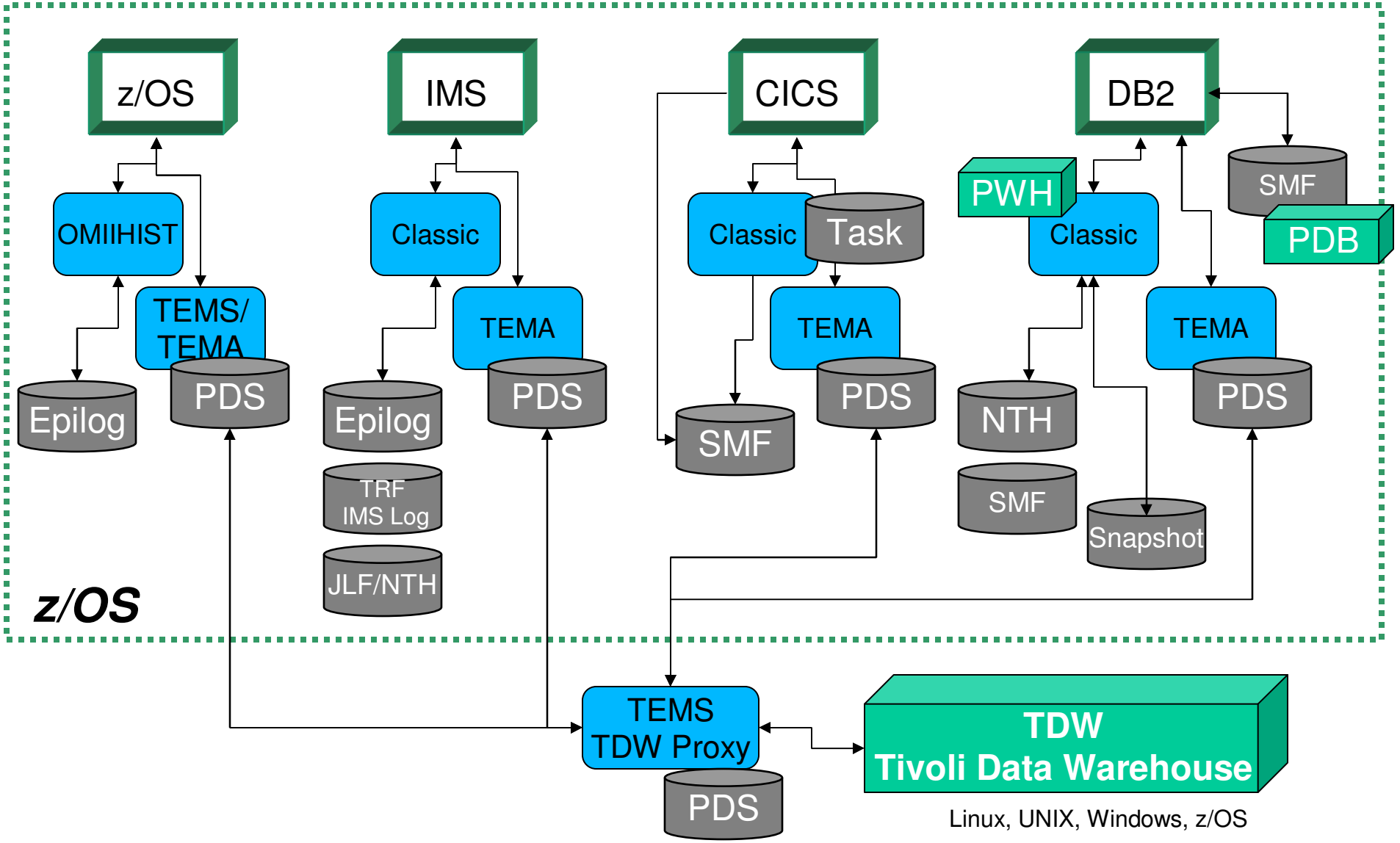
# Types Of Historical Monitoring Data

- Know the nature and characteristics of the history data being collected
- Detail data
  - Data that documents/measures detail of a specific event
  - Often high quantity data and the most detailed for analysis
  - May pose the greatest challenge in terms of cost, retention, post processing
  - Examples – DB2 Accounting records in Near Term History, CICS Task History, IMS Near Term History
- Summary data
  - Data that summarizes underlying detail data
  - Either an aggregation or an averaging of underlying detail records
  - May be useful for longer term trending and analysis
  - Reduces quantity of data and reduces cost of retention, post processing
  - Less detail may mean less diagnostic value
  - Examples – Summary data in Tivoli Data Warehouse, summary DB2 trace data

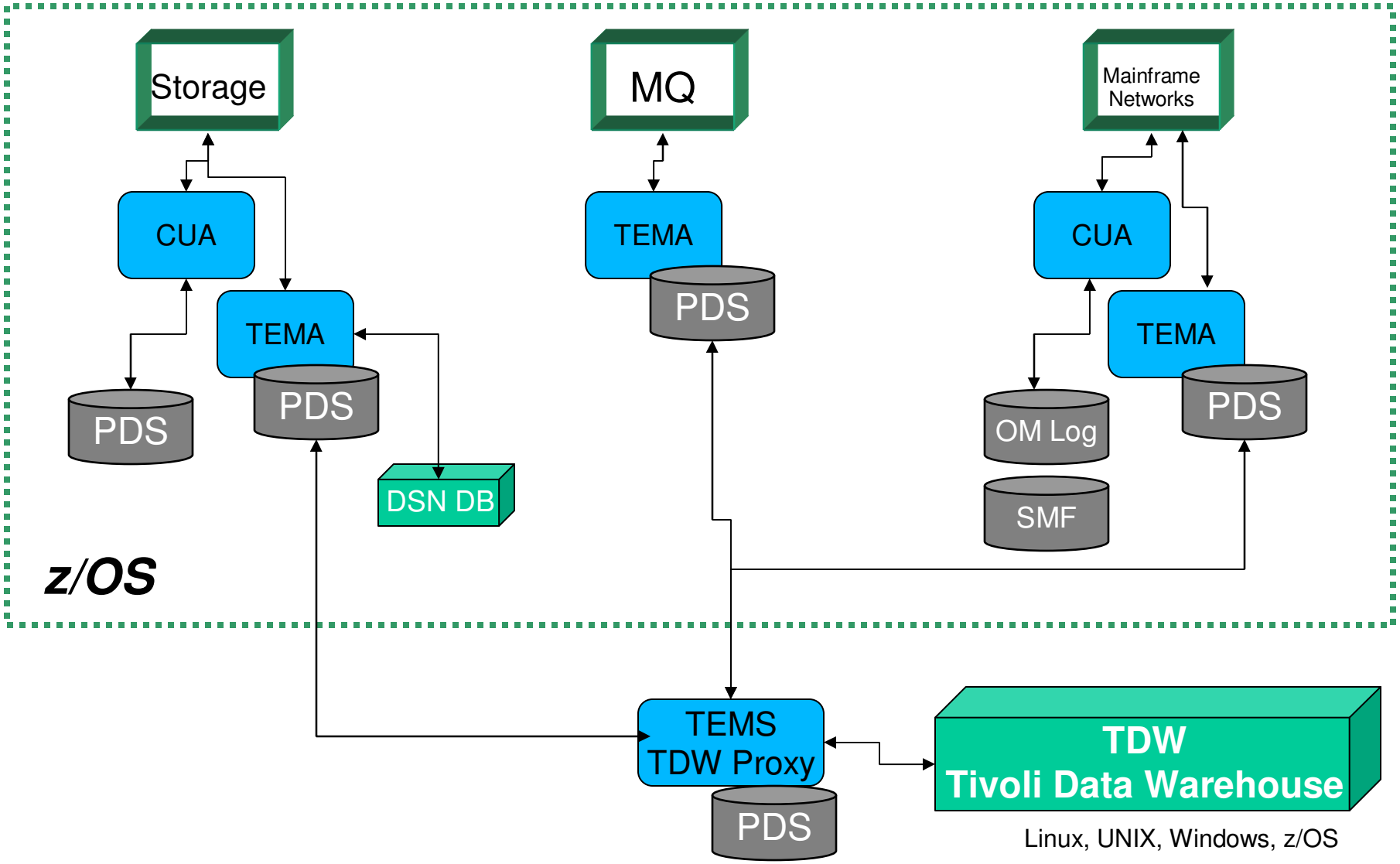
# Types Of Historical Monitoring Data - continued

- Interval data
  - History data that includes an encapsulation of one or multiple events within a specified time interval
  - The data will include all activity within that given time interval
  - Useful for problem analysis and trending analysis
  - Examples – DB2 statistics records in Near Term History, Epilog IMS or z/OS history
- Snapshot data
  - Typically a point in time snapshot of activity
  - Snapshots are usually based on a specified time interval
  - Snapshots may be taken of types of history (detail, summary, or interval)
  - Snapshots will show activity at time of the snapshot, but may/may not reflect activity between snapshots
  - Useful for problem analysis and trending analysis
  - Useful as an aid in setting alert thresholds
  - Examples – OMEGAMON DB2 PE GUI snapshot history, Tivoli Data Warehouse snapshot history

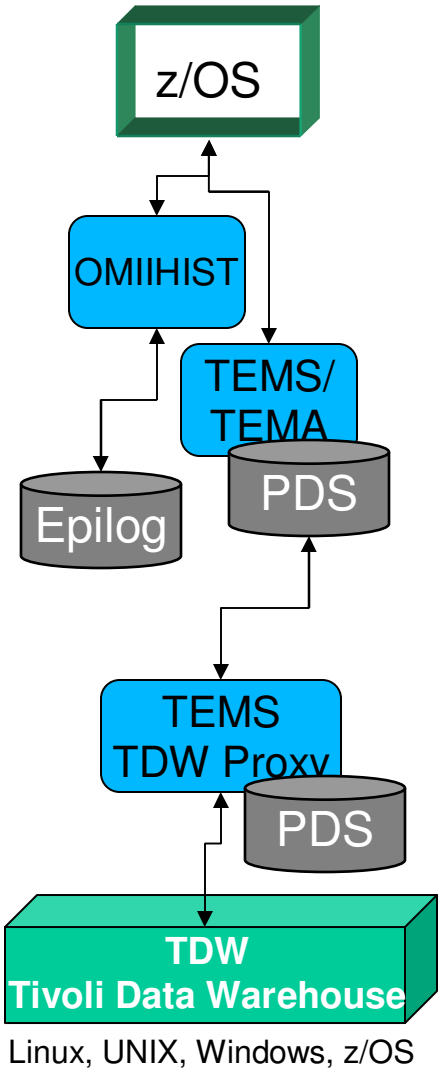
# OMEGAMON History Collection Options



# OMEGAMON History Collection Options - continued



# OMEGAMON XE For z/OS History




- OMEGAMON XE on z/OS provides history data in the form of Epilog history
  - Service levels (elapsed times and response times)
  - Resource utilization data
  - Degradation data (bottleneck analysis of z/OS workload)
- Epilog history may be accessed via CUA interface, batch jobs, ISPF command interface
  - Sample batch reporter JCL is in *hilev*.RKANSAMU(KEPPROC)
- OMEGAMON XE on z/OS provides snapshot history data and supports the Tivoli Data Warehouse (TDW)
  - Data is stored in the TEMA/TEMS in the Persistent Data Store (PDS)
  - Data may be optionally sent to the TDW
  - Data may be summarized and pruned using the TDW
- Cost of collection relative to value
  - Epilog – low cost ⟷ Useful for problem analysis
  - Snapshot – low cost ⟷ Useful for trending

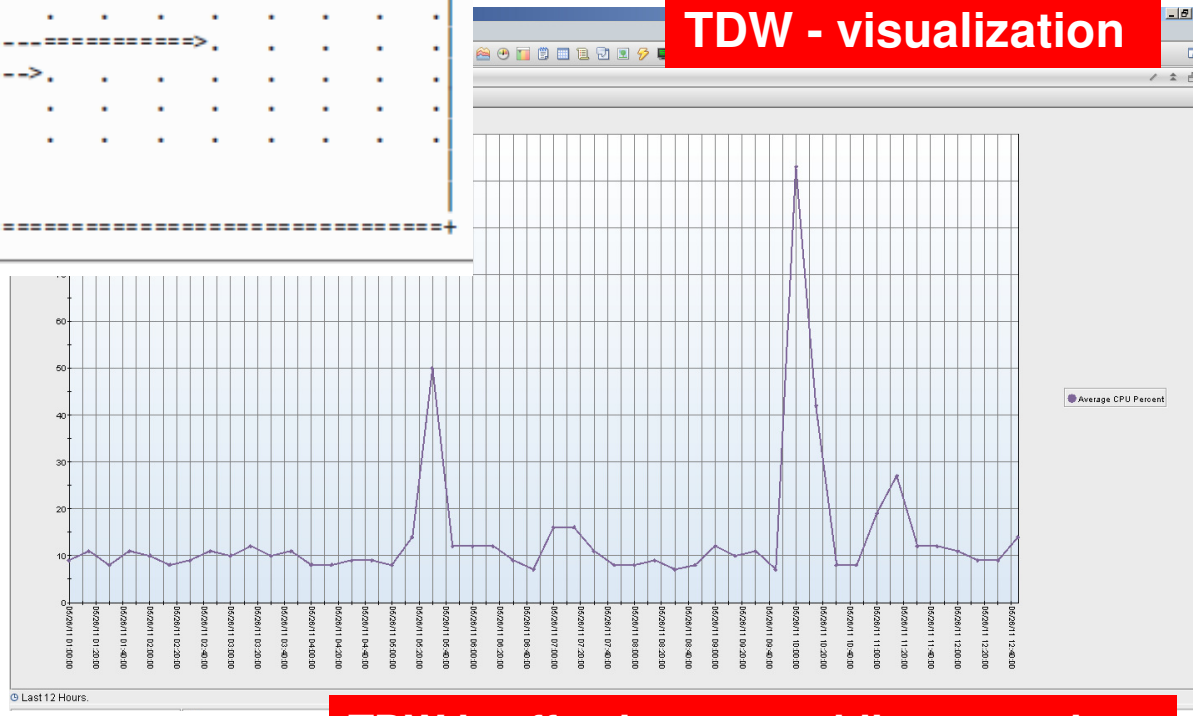
# The Value Of OMEGAMON z/OS History

**Epilog - historical problem analysis**

```
+=============================================================================+
| Job = PAYROLL          JES Number = 2544            Job Steps =   3 /   6   |
| Job Class = T          Account Number = 035G        Input Queue = 48.10 S   |
| From  09:12 To 09:14 On 08/01/99                    Elap =    1:33 M SYSA   |
+-----------------------------------------------------------------------------+
|WAIT_Reason_____________Time____%_|0___1___2___3___4___5___6___7___8___9___0|
|Using CPU              0.09 S    .1|.   .   .   .   .   .   .   .   .   .   .|
|Swapped With WTOR     52.65 S  59.9|-----------==========>.   .   .   .   .|
|Tape Mount Pending    25.05 S  28.5|----------->.   .   .   .   .   .   .   .|
|ECB Wait               5.01 S   5.7|--> .   .   .   .   .   .   .   .   .   .|
|Tape  IEFU84  522 Que  5.01 S   5.7|--> .   .   .   .   .   .   .   .   .   .|
|Job Elapsed Time       1:27 M                                                |
+=============================================================================+
```

**Epilog history provides historical bottleneck analysis data correlated with resource and usage data**

**TDW - visualization**

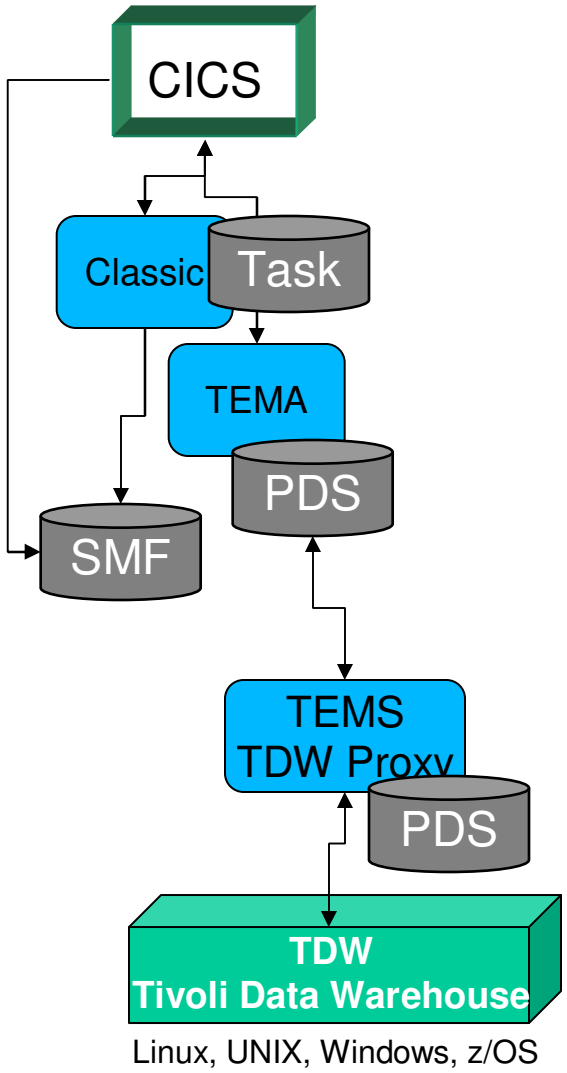



**Use Epilog for historical problem analysis**

**Use TDW for highlighting peaks and trending utilization (use as input for setting alert thresholds)**

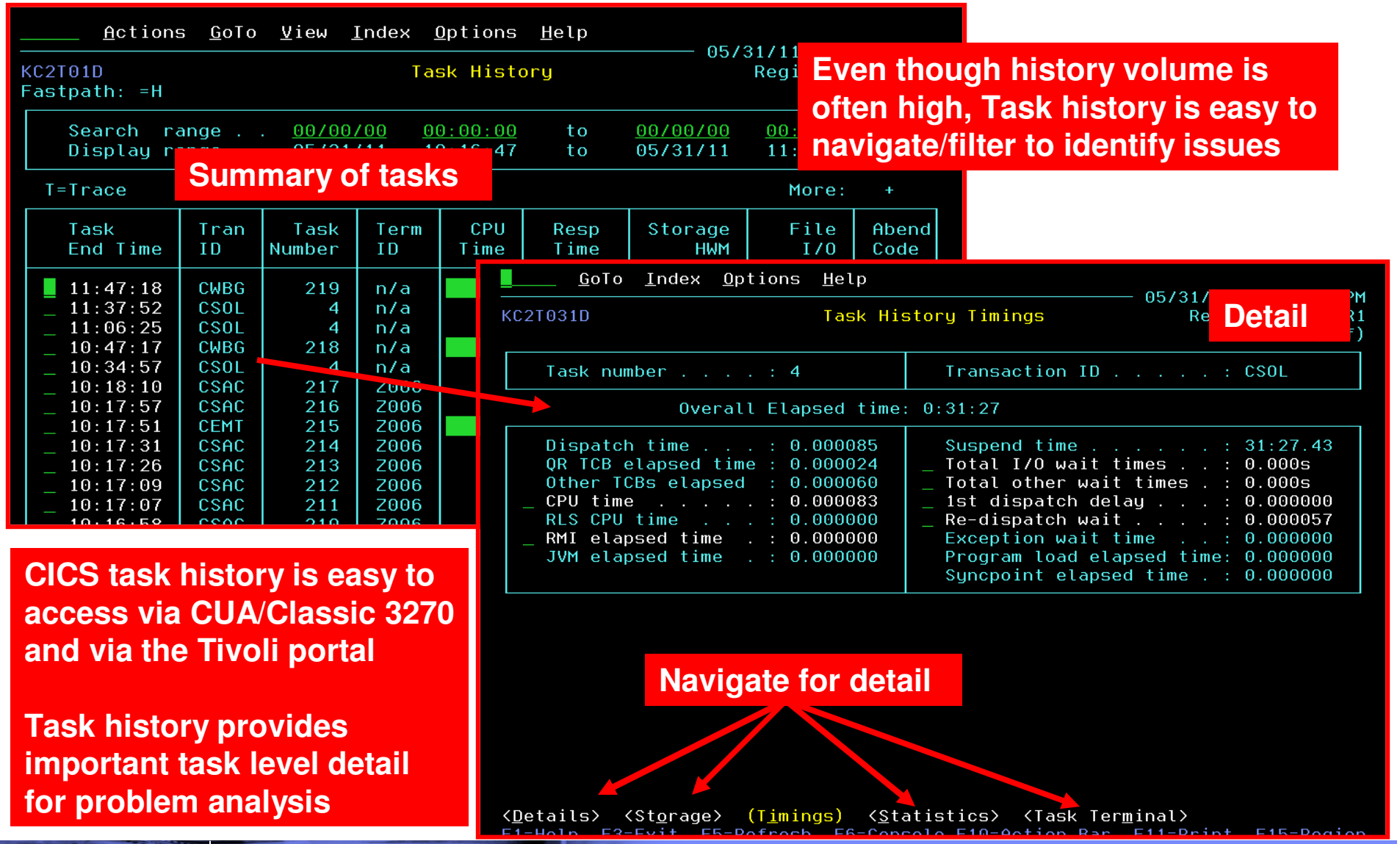
**TDW is effective at providing snapshot data for trend analysis**

# OMEGAMON XE For CICS Provides History Options For History Detail, Near Term Detail And Trending




- Task history (also known as Online data viewing) provides detailed CICS transaction level history
  - Detailed transaction-level data stored in an ONDV task history file (wraparound VSAM file – one per CICS region)
  - Easy to access/filter – very good detail
- SMF 110 records
  - SMF 110 subtype 1 records - CICS task level data
  - CICS Statistics data (SMF 110, subtypes 2, 3, 4, 5)
    - Information collected on an interval basis and/or end of day
  - Note – OMEGAMON may optionally add additional detail to SMF 110 records
- OMEGAMON CICS provides snapshot history data and supports the Tivoli Data Warehouse (TDW)
  - Data is stored in the TEMA/TEMS in the PDS and optionally sent to the TDW
  - Data may be summarized and pruned using the TDW
- Cost of collection relative to value
  - Task history – low to moderate ⟷ Useful for problems
  - SMF 110s – typically low ⟷ Important for reporting
  - Snapshot – typically low ⟷ Useful for trending

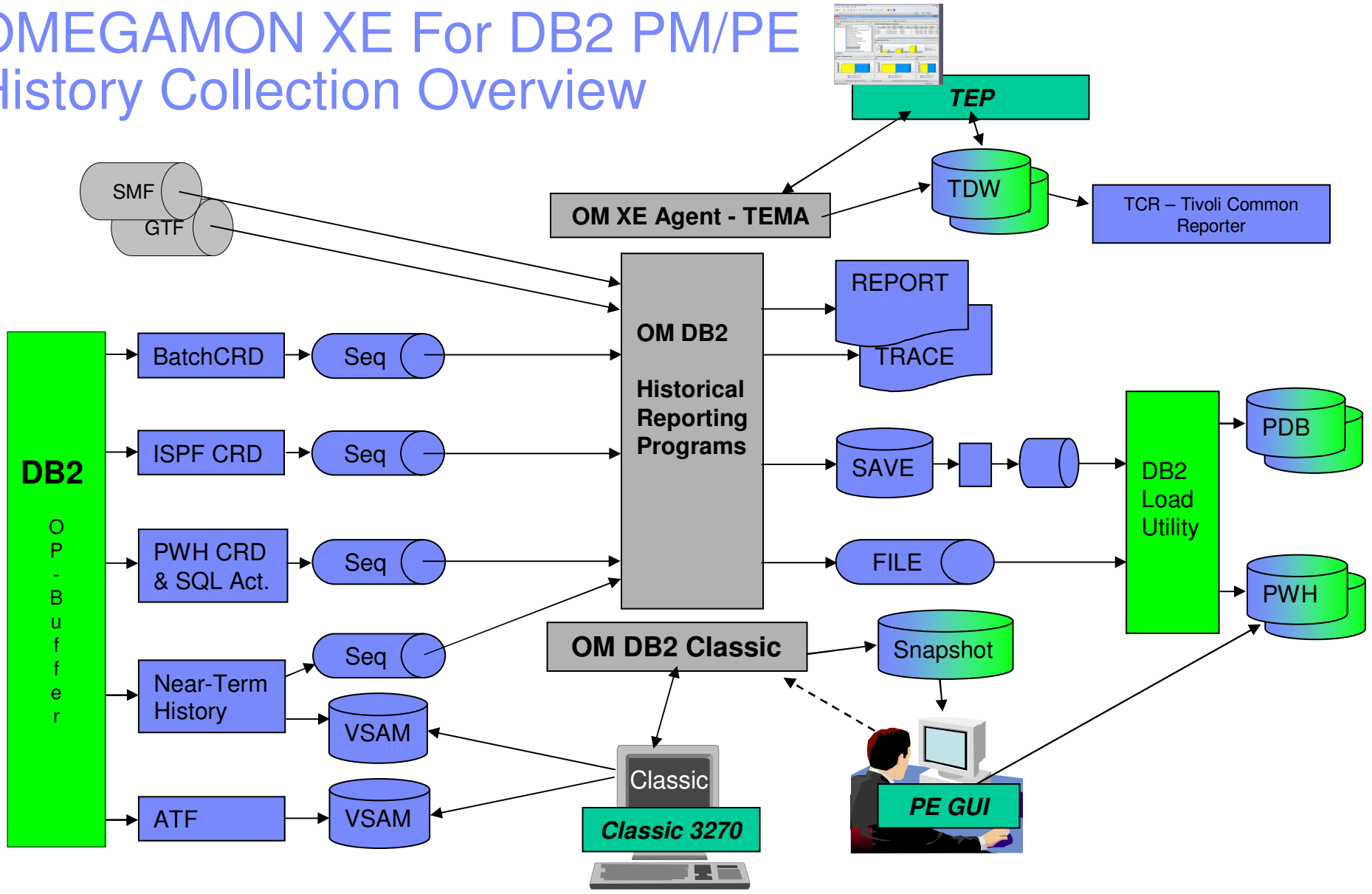
# OMEGAMON CICS Task History – Valuable For Problem Analysis Access Via 3270 Interface And The TEP




**CICS task history is easy to access via CUA/Classic 3270 and via the Tivoli portal**

**Task history provides important task level detail for problem analysis**

# OMEGAMON XE For DB2 PM/PE History Collection Overview



# OMEGAMON DB2 Near Term Thread History

## Easy Access To History Within OMEGAMON Classic Interface

**View thread level data (Accounting detail or summary)**

**View DB2 Statistics data (interval data)**

```
./C DSNC 07/25/09  9:24:57   3
                    Down PF8          Zoom PF11
>
>                  Enter a selection letter on the top line.
>
> *-BY PLAN       B-BY AUTHID       C-BY PLAN,AUTHID       D-BY AUTHID,PLAN
> O-OPTIONS
================================================================================
>                             THREAD HISTORY BY PLAN
 HAGP
+ Report Interval:   15 mins                          St
+ Report Filtered:       NO
 plan
+      Thread Summary Not Available, Data Collecte
+
+                                      DLk/  In-DB2  In-DB2   In-DB2          GetP/
+ Plan      Thrds Commit Abrt  DML     TOut  Elap Tm CPU Tm  Wait Tm  Getpage RIO
+ -------- ----- ------ ---- ----- ---- ------- ------- -------  ------- -----
+ ASNQA910      9      90    0  1486    0      .5     .15      .4     1225  1225
+ ASNQC910      1      10    0    20    0      .0     .00      .0       54  54.0
+ DISTSERV     15      49   30   117    0      .2     .04      .2      210 210.0
================================================================================
```

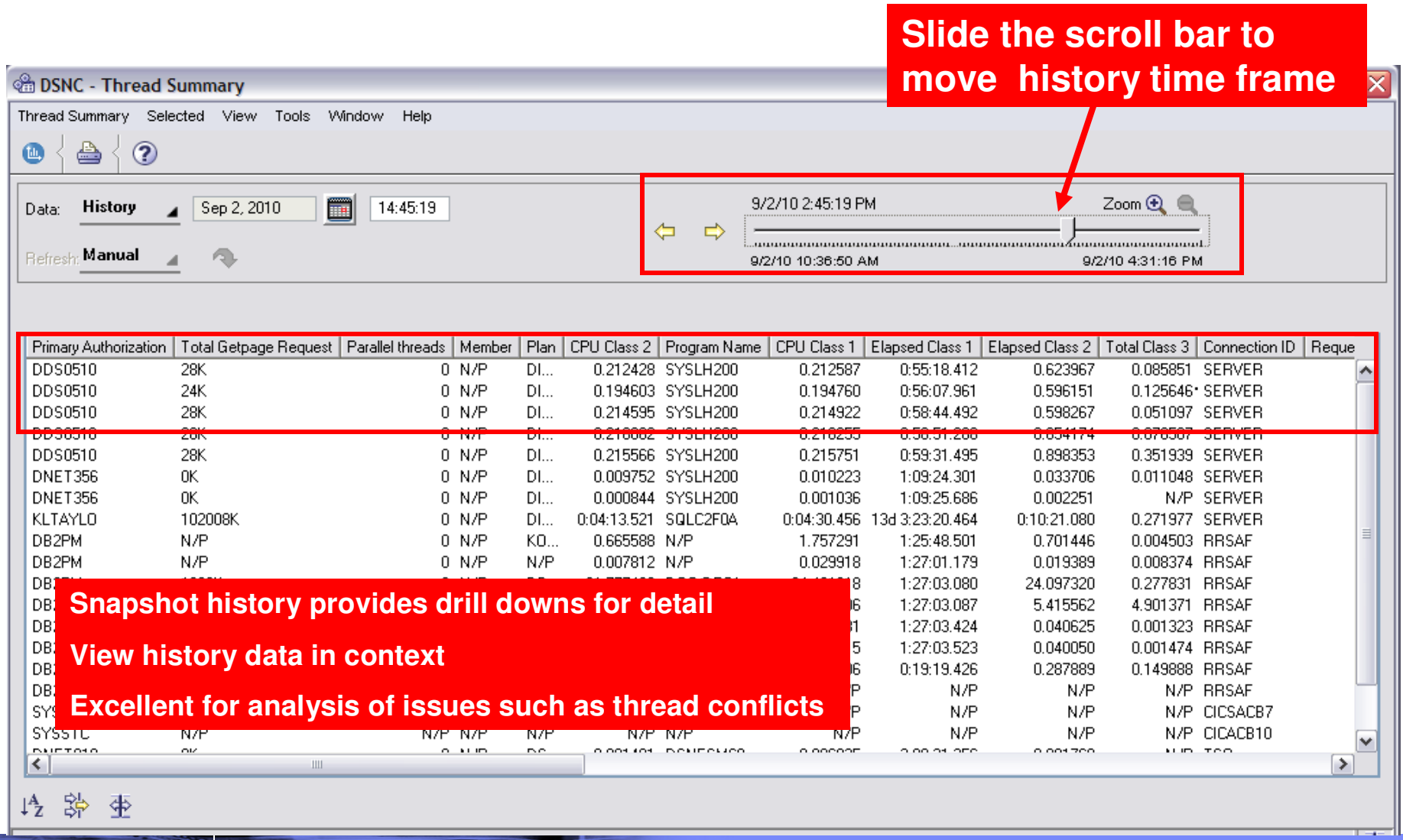
**Near Term History stored in VSAM files for easy access from Classic interface.**

**Many filter/view options.**

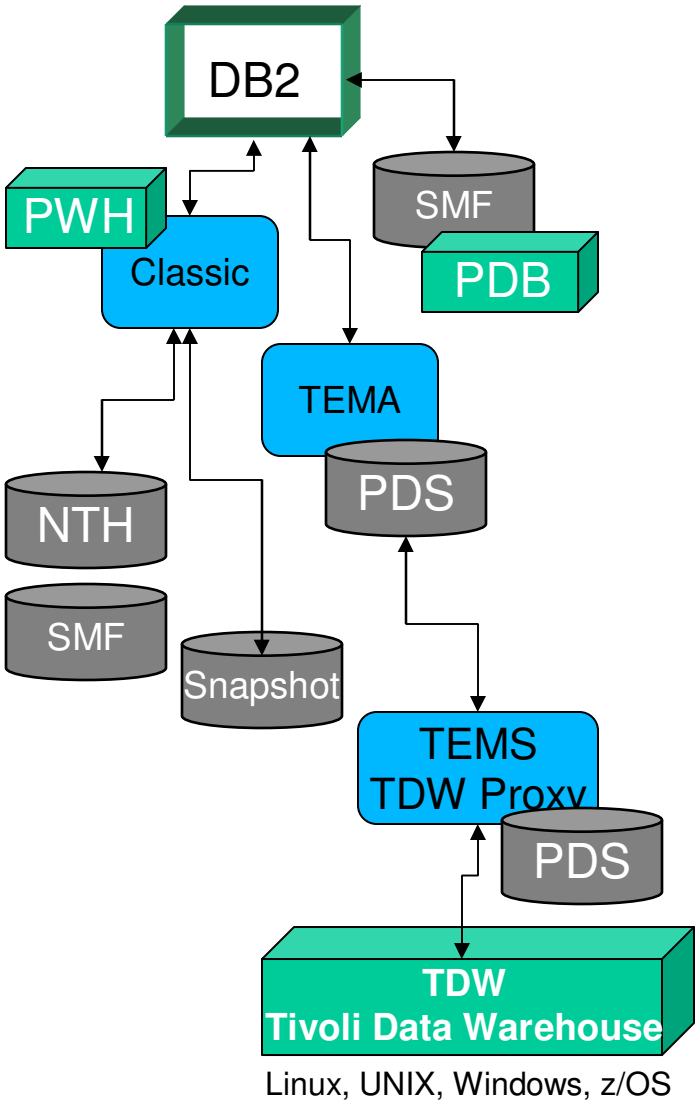
**F11 to see more detail on a specific thread**

**NTH is highly detailed, easy to filter and access via 3270 interface, and is highly useful for problem analysis**

# OMEGAMON DB2 Snapshot History Scrollable And Easy To Access Via PE GUI



# OMEGAMON XE For DB2 History Collection Summary

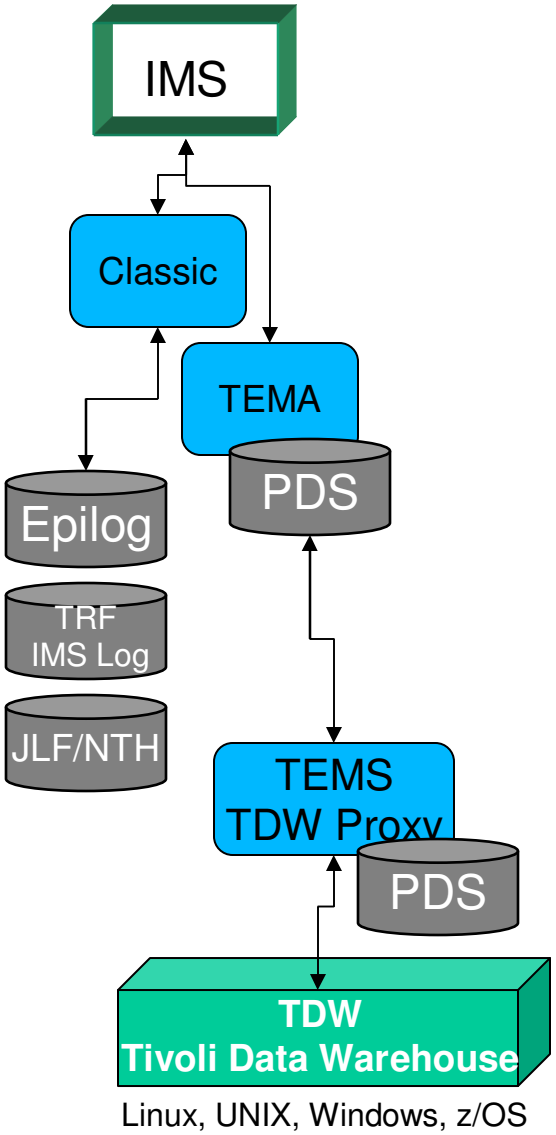



- Near Term History
  - Accounting (thread detail) and Statistics stored in a set of VSAM files – primary access via 3270 interface
  - Very detailed – useful for problem analysis
- Performance Warehouse
  - DB2 trace data (Accounting, Statistics, Performance) stored in DB2 tables
  - Collection and objects managed by OMEGAMON
  - Detail and quantity of data is variable
- Performance Database
  - DB2 trace data (Accounting, Statistics, Performance) stored in DB2 tables
  - Collection and objects managed by user
  - Detail and quantity of data is variable
- Snapshot history – PE GUI
  - Snapshots on a user defined interval
  - Easy to view and navigate via the PE GUI interface
- TDW snapshot history (different from PE GUI)
  - Use PDS and TDW infrastructure as other OMEGAMONs

# OMEGAMON XE For DB2 History Collection Options Considerations And Recommendations

- Near Term History (NTH)
  - Detailed history data that is easily accessible
  - NTH is often the most costly to collect in most shops
  - Cost of collection – moderate to high  Value – usually high
- Performance Warehouse
  - Detail of data and cost of collection varies depending upon user requirements
  - General recommendation – use when desired for lower cost/quantity data
- Performance Database
  - For higher quantity/detail requirements
  - Provides more manual control for higher volumes of history data collection
- Snapshot history – PE GUI
  - Easy to access and low cost to collect – requires the PE GUI
  - A low cost alternative to NTH  limitations of snapshot data collection
- TDW snapshot history
  - Cost of collection - low
  - Useful for trending analysis, not as detailed NTH or PE GUI snapshot

# OMEGAMON XE For IMS History



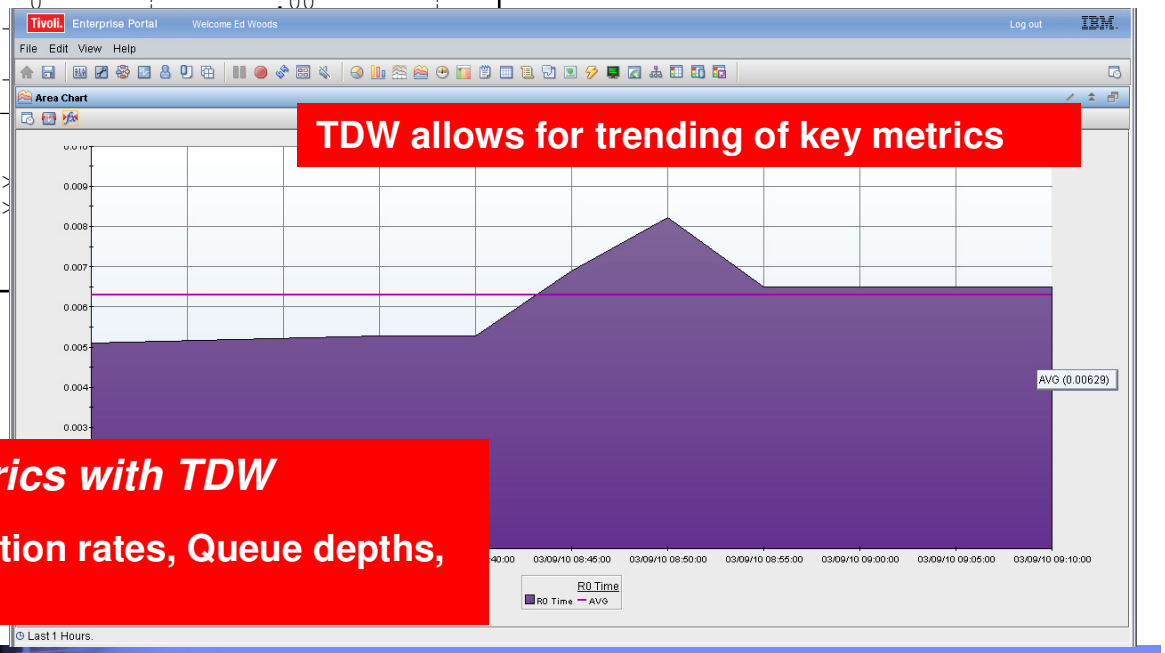

- Epilog provides IMS history
  - Service levels (response times), resource utilization data, and degradation data (bottleneck analysis of IMS workload)
  - Detail is limited - interval/group based
  - Cost – low ⟷ Value – moderate
- Near Term History (NTH)
  - Transaction detail history (tran level/call level detail)
  - Easy to access via 3270 Classic interface
  - Data collected to Journal Logging Facility (JLF)
  - Cost – moderate ⟷ Value – moderate to high
- Transaction Reporting Facility (TRF)
  - DB call level detail and summary data written to IMS log
  - Useful for chargeback and some performance analysis
  - Cost – high ⟷ Value – limited use/requires batch
- Snapshot history data and the Tivoli Data Warehouse (TDW)
  - Data is stored in the TEMA/TEMS in the PDS and optionally sent to the TDW
  - Data may be summarized and pruned using the TDW
  - Cost of collection is low – data is useful for trending analysis

# OMEGAMON IMS Historical Data Collection Alternatives

**Epilog data is interval based**

**Useful for analysis of bottlenecks relative to overall system workload**

```
EPILOG/IMS V420   09/29/09   7:13   Mode: PAGE
CMD==>
*****************************************
+=======================================
¦ Transaction Group =    1        Symbolic Na
¦ Period:  05:59 to 06:14 on 09/28/09                    Elap =  14:4
+-------------------------------------------------------------------
¦                          RESPONSE TIME DATA                      ¦
+-------------------------------------------------------------------
¦Response_Component_____Avg. Rsp. Time_____Trans. Count_____Rate (per min.)  ¦
Input Queue          ¦     0.00 S       ¦          0       ¦        .00       ¦
¦Pgm Input Queue     ¦     0.05 S       ¦         56       ¦       3.81       ¦
¦Processing          ¦     0.73 S       ¦         55       ¦       3.74       ¦
¦Response time 0     ¦     0.77 S       ¦         55       ¦       3.74       ¦
¦Output Queue        ¦     0.00 S       ¦          0       ¦        .00       ¦
¦Response time 1     ¦     0.00 S       ¦          0       ¦        .00       ¦
+-------------------------------------------------
¦                            DEGRADATION DATA
+-------------------------------------------------
¦Competing_State_________Time_____%_¦0___1___2___
¦MVS Waits             0.24 S   14.3¦----->  .
¦ PGM Fetch I/O      (0.24) S (14.3)¦----->  .
¦IMS Waits             0.49 S   28.6¦---------->
¦ Iswitched to CTL   (0.49) S (28.6)¦---------->
```

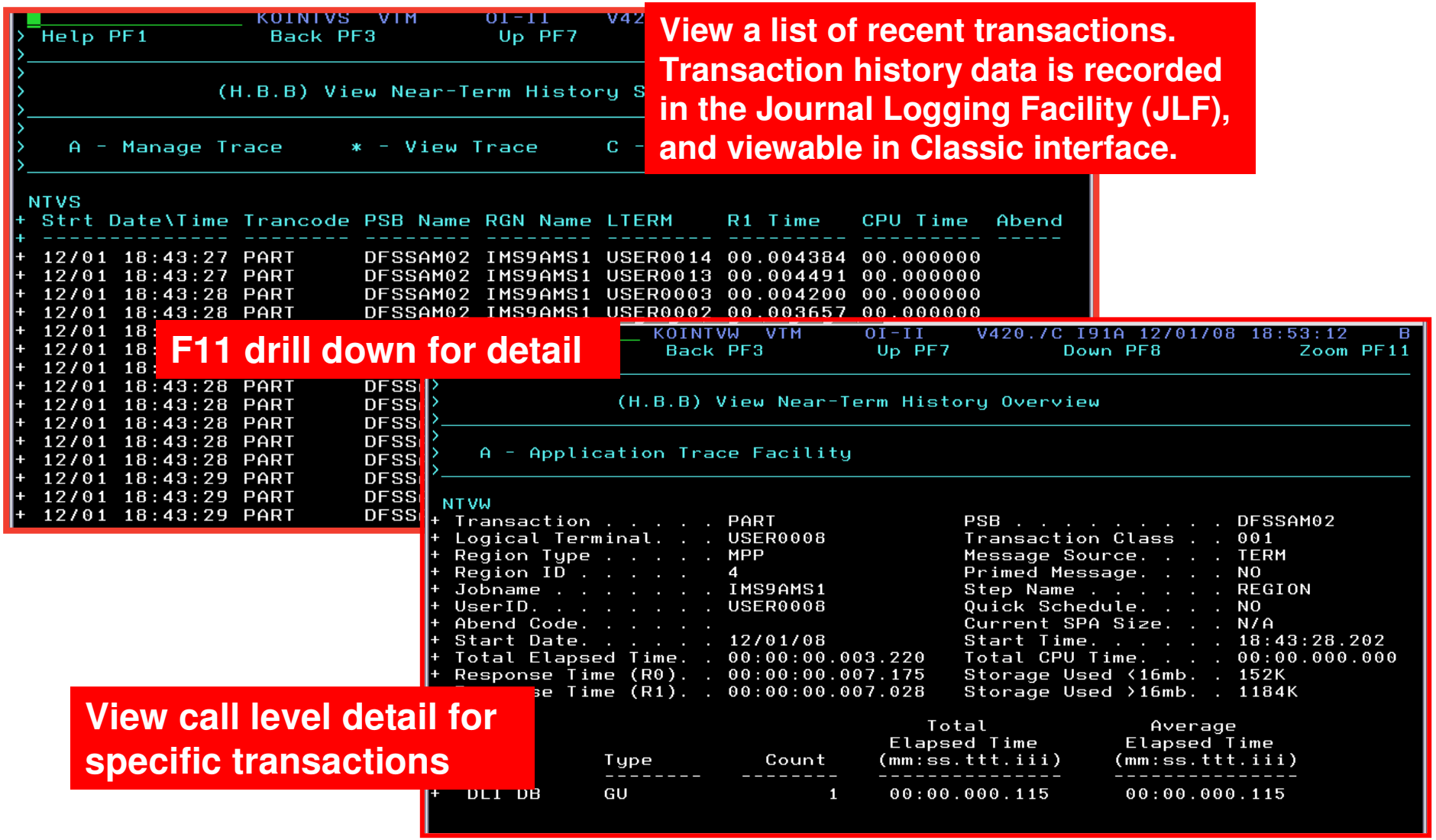



**TDW allows for trending of key metrics**

***Trend key IMS performance metrics with TDW***

**Response time, Bottlenecks, Transaction rates, Queue depths, Buffer/pool performance**

# Near Term History Of IMS Transactions Useful For Problem Analysis

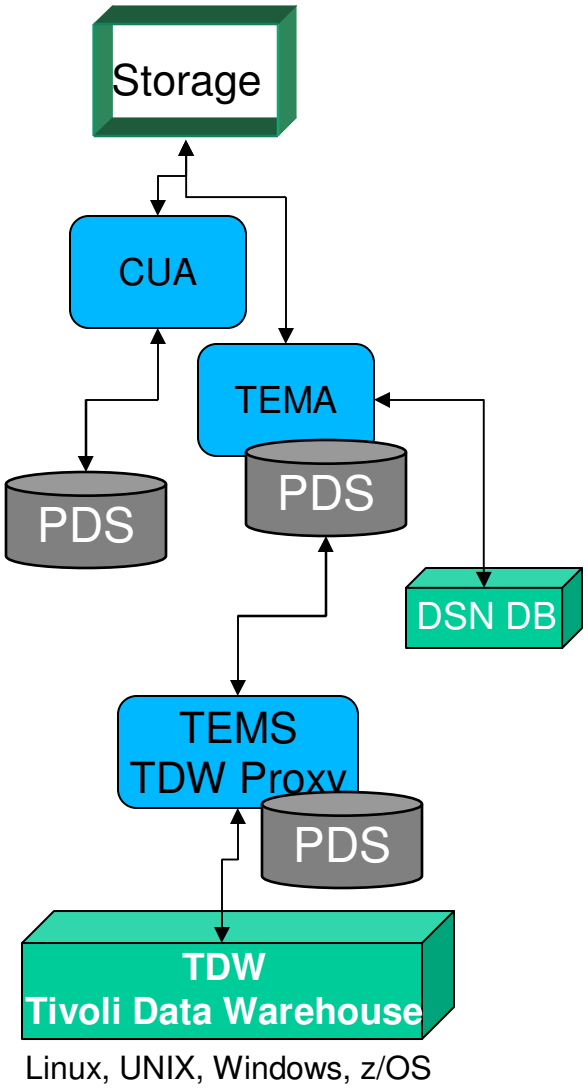



**View a list of recent transactions. Transaction history data is recorded in the Journal Logging Facility (JLF), and viewable in Classic interface.**

**F11 drill down for detail**

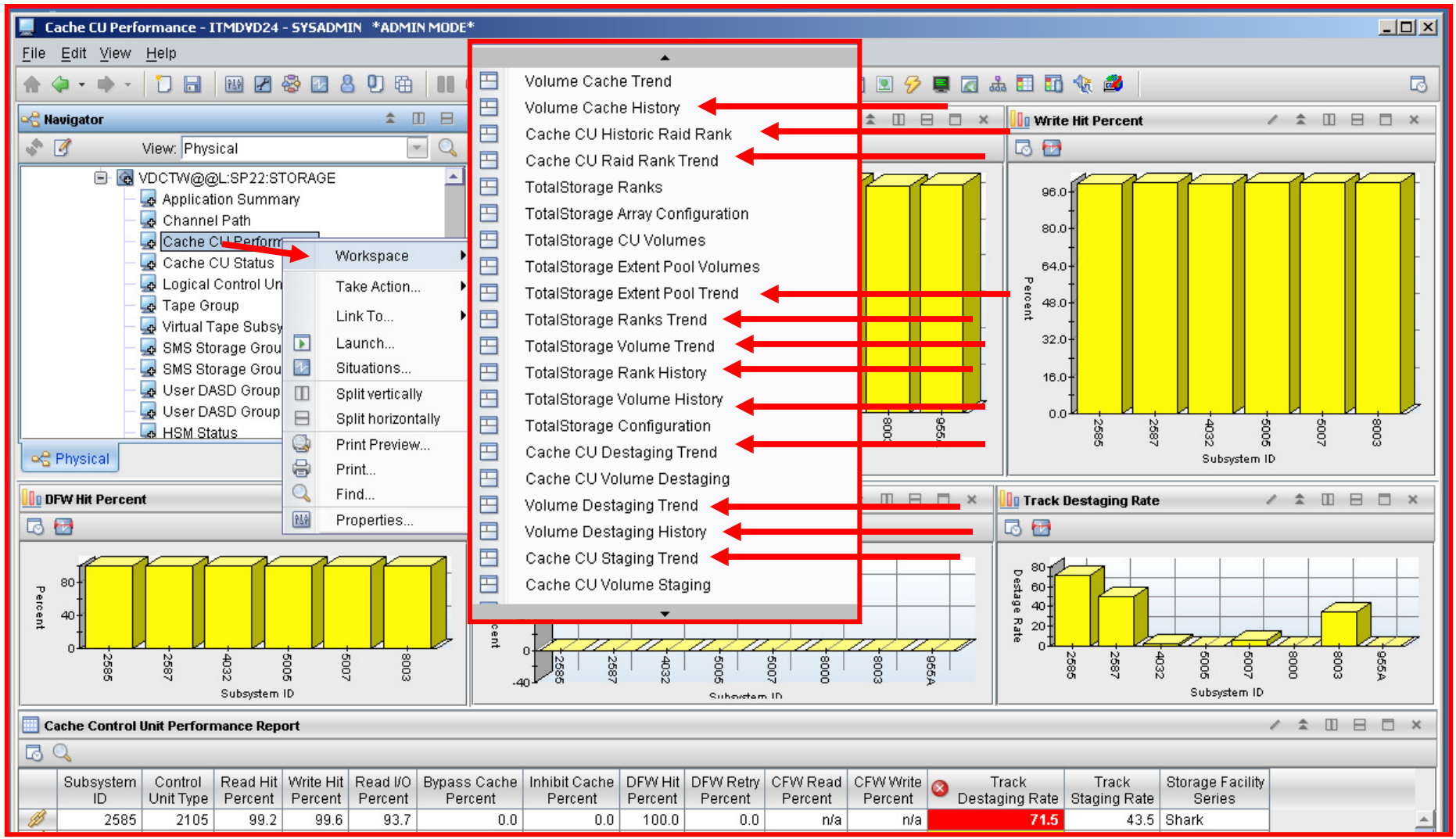
**View call level detail for specific transactions**

# OMEGAMON XE For Storage History

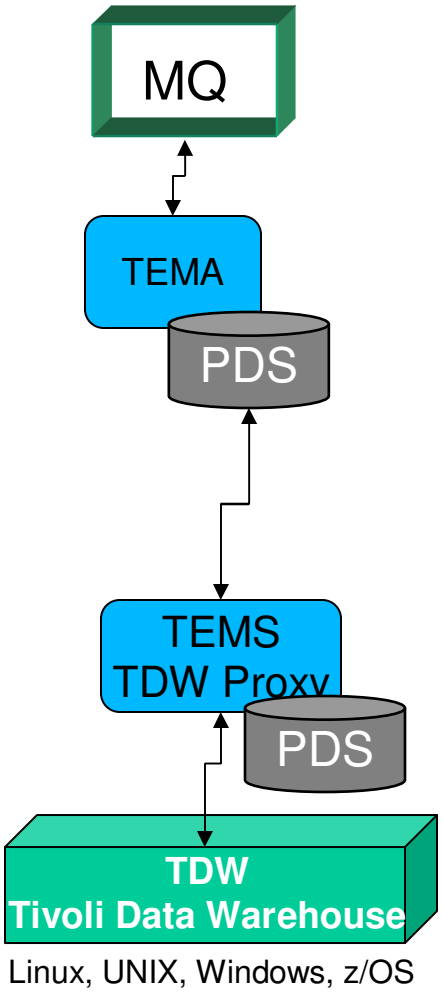



- OMEGAMON XE For Storage makes extensive use of the Persistent Data Store (PDS) for data collection
- PDS data may be accessed by both the CUA 3270 and Tivoli Enterprise Portal interfaces
- OMEGAMON Storage provides numerous product provided Tivoli Enterprise Portal history workspaces
- Cost of collection
  - Potentially high since many shops may have thousands of devices to gather information about
  - Observe best practices for OMEGAMON Storage monitoring
    - Avoid redundant monitoring of devices
    - Group related devices and use wild cards to set options
    - Consider options carefully when monitoring at the application and data set level
  - When defining history in the TEP and TDW consider quantity of data being collected
    - Number of devices, controllers, data sets, applications
- Value can be high, but so can cost

# OMEGAMON XE For Storage Provides Trending/History Information At Several Levels

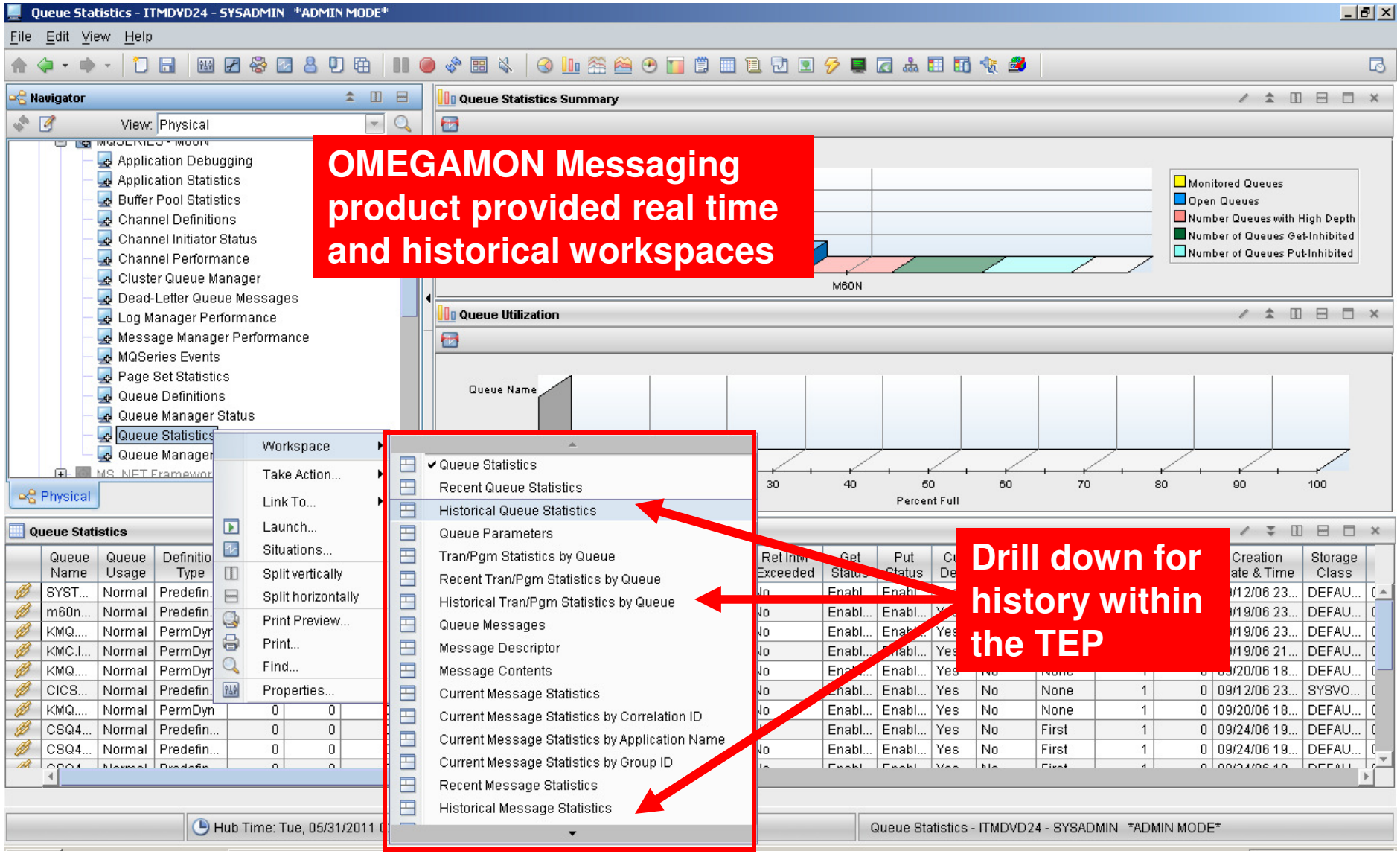


# OMEGAMON XE For Messaging History




- OMEGAMON XE For Messaging provides snapshot history data and supports the Tivoli Data Warehouse (TDW)
  - Data is stored in the TEMA/TEMS in the Persistent Data Store (PDS)
  - Data may be optionally sent to the TDW
  - Data may be summarized and pruned using the TDW
- OMEGAMON XE For Messaging provides many history workspaces out of the box
  - Examples of product provided workspaces include
    - Queue statistics, tran/program statistics by queue, Message statistics, Page set statistics, Message manager performance, Log manager performance, Channel performance
- Snapshot data is easy to access within the Tivoli Portal
  - Cost of collection is low ⟷ value is moderate to high

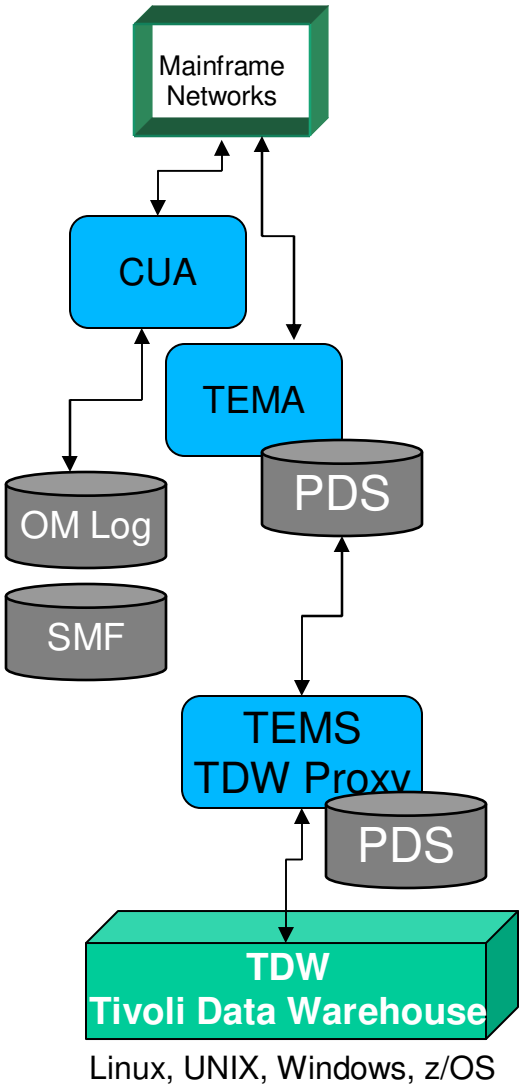
# Accessing OMEGAMON Messaging History Data




**OMEGAMON Messaging product provided real time and historical workspaces**

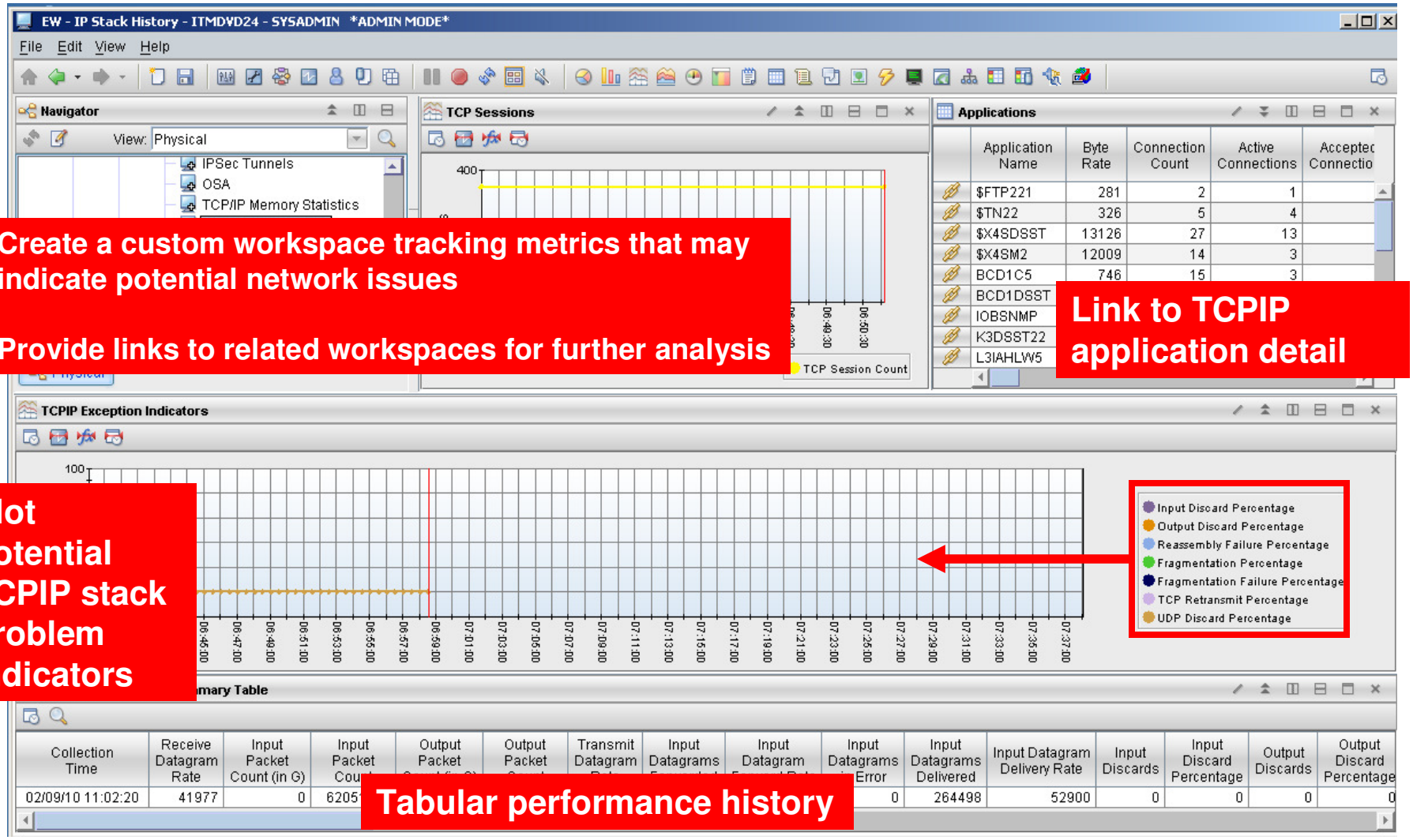
**Drill down for history within the TEP**

# OMEGAMON XE For Mainframe Networks History




- OMEGAMON XE for Mainframe Networks provides snapshot history data and supports the Tivoli Data Warehouse (TDW)
  - Data is stored in the TEMA/TEMS in the Persistent Data Store (PDS)
  - Data may be optionally sent to the TDW
  - Data may be summarized and pruned using the TDW
- When configuring history in the TEP/TDW
  - Be aware of relative number of rows per snapshot and snapshot frequency when specifying collection
    - Example- application level versus connection level history
- OMEGAMON XE For Mainframe Networks provides trending history log in the CUA 3270 interface
  - Data is logged and viewable in CUA
  - Recommendation - the most current information is in the Tivoli Portal, therefore focus history efforts in the TEP
- Cost of collection relative to value
  - CUA log – typically low ⟷ Limited data – use TEP
  - Snapshot – typically low ⟷ Useful for trending/analysis

# OMEGAMON Mainframe Networks Example
# A Custom Workspace Showing Network Problem Indicators




**Create a custom workspace tracking metrics that may indicate potential network issues**

**Provide links to related workspaces for further analysis**

**Link to TCPIP application detail**

**Plot potential TCPIP stack problem indicators**

**Tabular performance history**

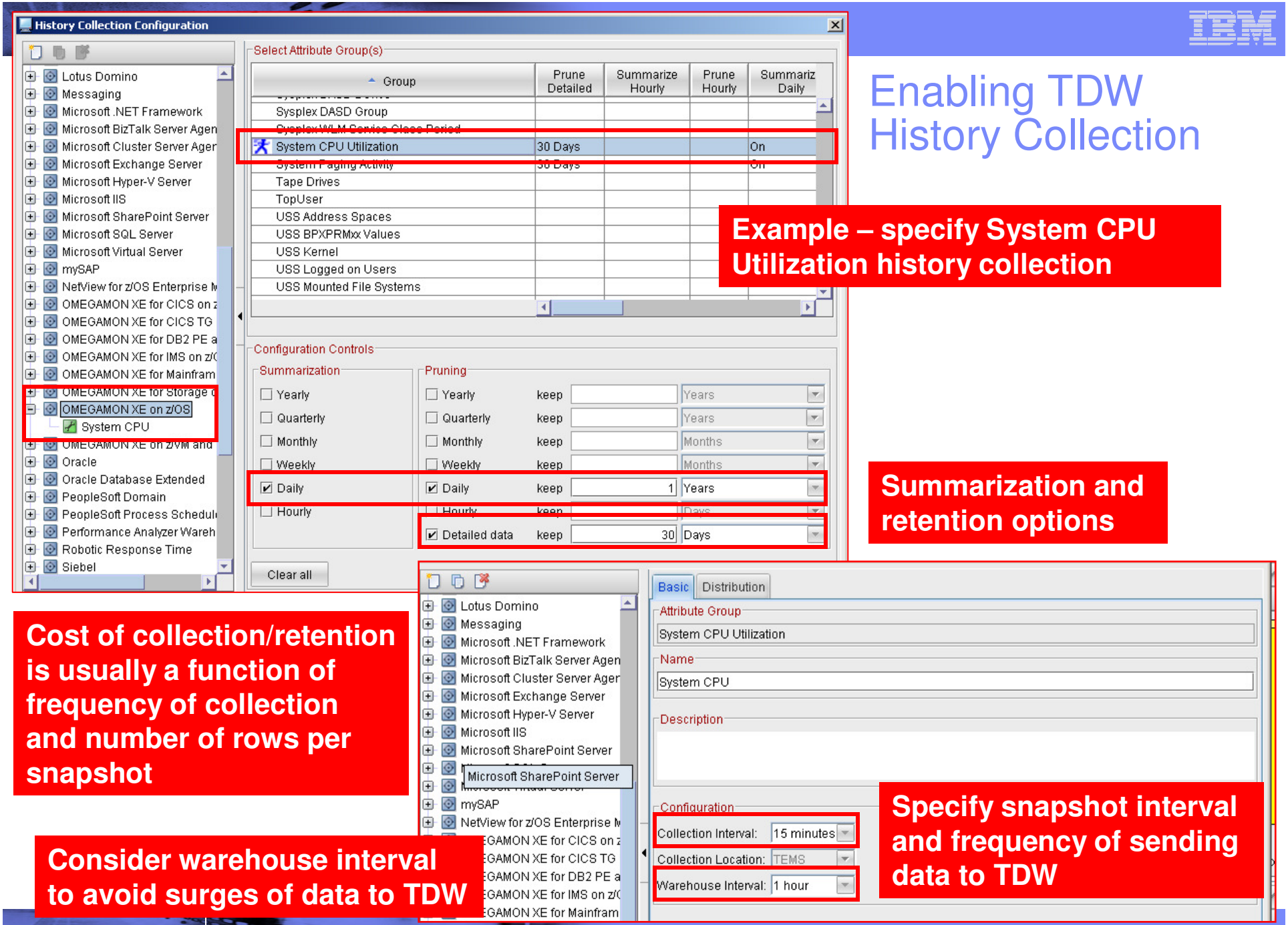
# Using History To Become More Proactive

- A strategy to be more proactive
  - ***Visualize - Control - Automate***
- Use history data to improve the visualization of system activity and resource utilization
  - Use history data to identify peaks/valleys/bottlenecks
  - Use trending and visualization to identify potential issues
- Use history to improve control
  - Customize workspaces, views and navigation
  - Identify and isolate issues and take corrective actions
- Use history to improve automation
  - Improve alerts by making situation thresholds more accurate and relevant
    - History data can be used as a reference point to make sure threshold levels in situations reflect real problems

# Considerations For Collecting Tivoli Data Warehouse Snapshot History Data

- Avoid the “turn on everything” method
  - Turning on everything will result in a fire-hose of information that will potentially obscure useful information, waste space, increase cost of collection, and slow down data recall
- Project potential volume of history being collected
  - Warehouse projection worksheet provides a means to calculate
  - Here is a link to documentation for the tool:
    - http://publib.boulder.ibm.com/infocenter/tivihelp/v15r1/index.jsp?topic=/com.ibm.itm.doc_6.2.2fp1/ch2.3warehousecon.htm
- Consider options for history data retention
  - Many deploy TDW with DB2 on a Linux/Unix/Windows type platform to collect and house data
  - You may optionally store your history data on DB2 on z/OS
    - Requires DB2 on z/OS at the V9 level, or above
    - Here is a link to a white paper that goes through the setup of TDW on DB2 on z/OS:
      - http://www-03.ibm.com/support/techdocs/atsmastr.nsf/5cb5ed706d254a8186256c71006d2e0a/b327c2b1683071e28625786400634a7f/$FILE/TDW_DB2_ZOS_Considerations.pdf

# Enabling TDW History Collection




**Example – specify System CPU Utilization history collection**

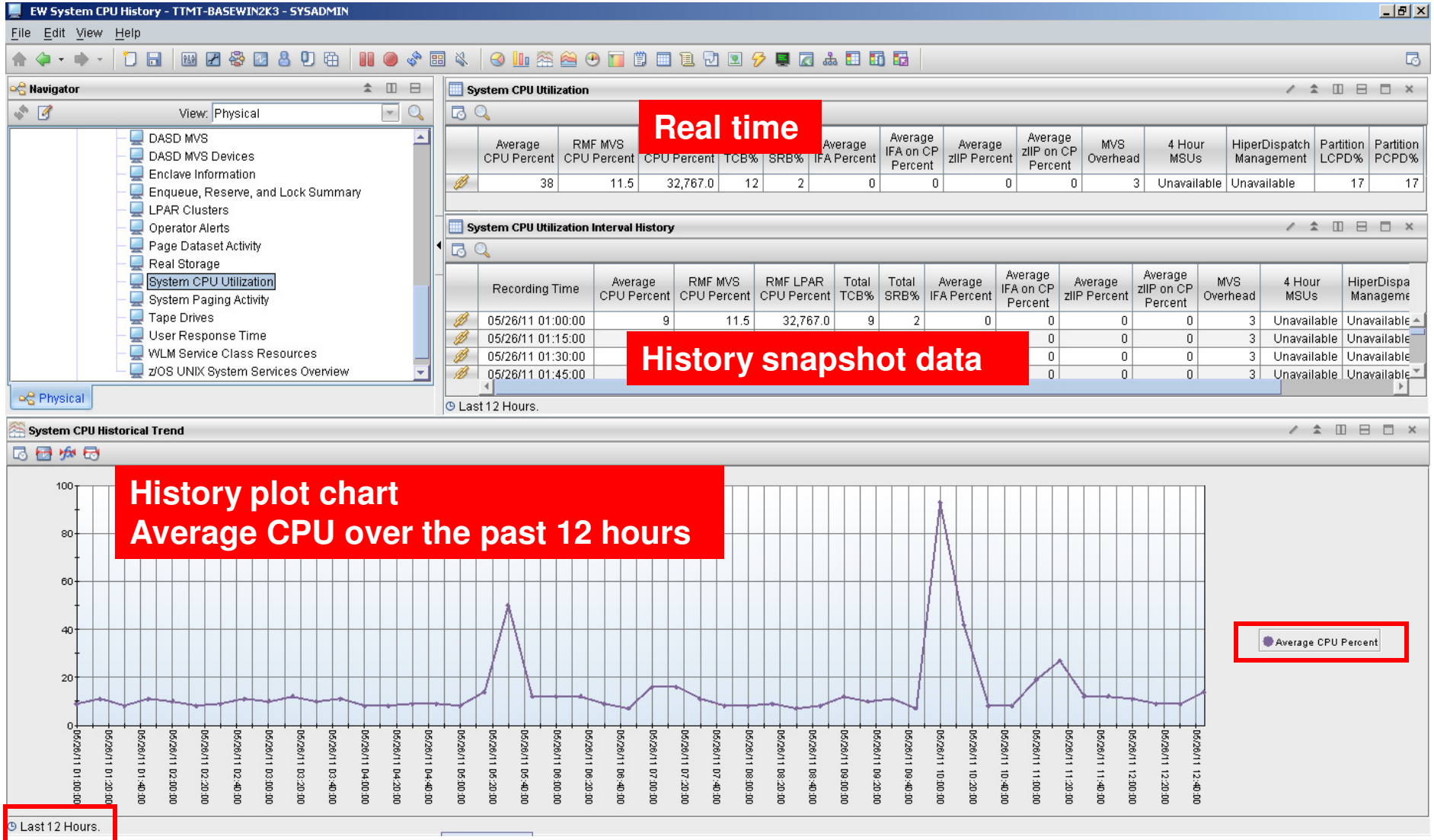
**Summarization and retention options**

**Cost of collection/retention is usually a function of frequency of collection and number of rows per snapshot**

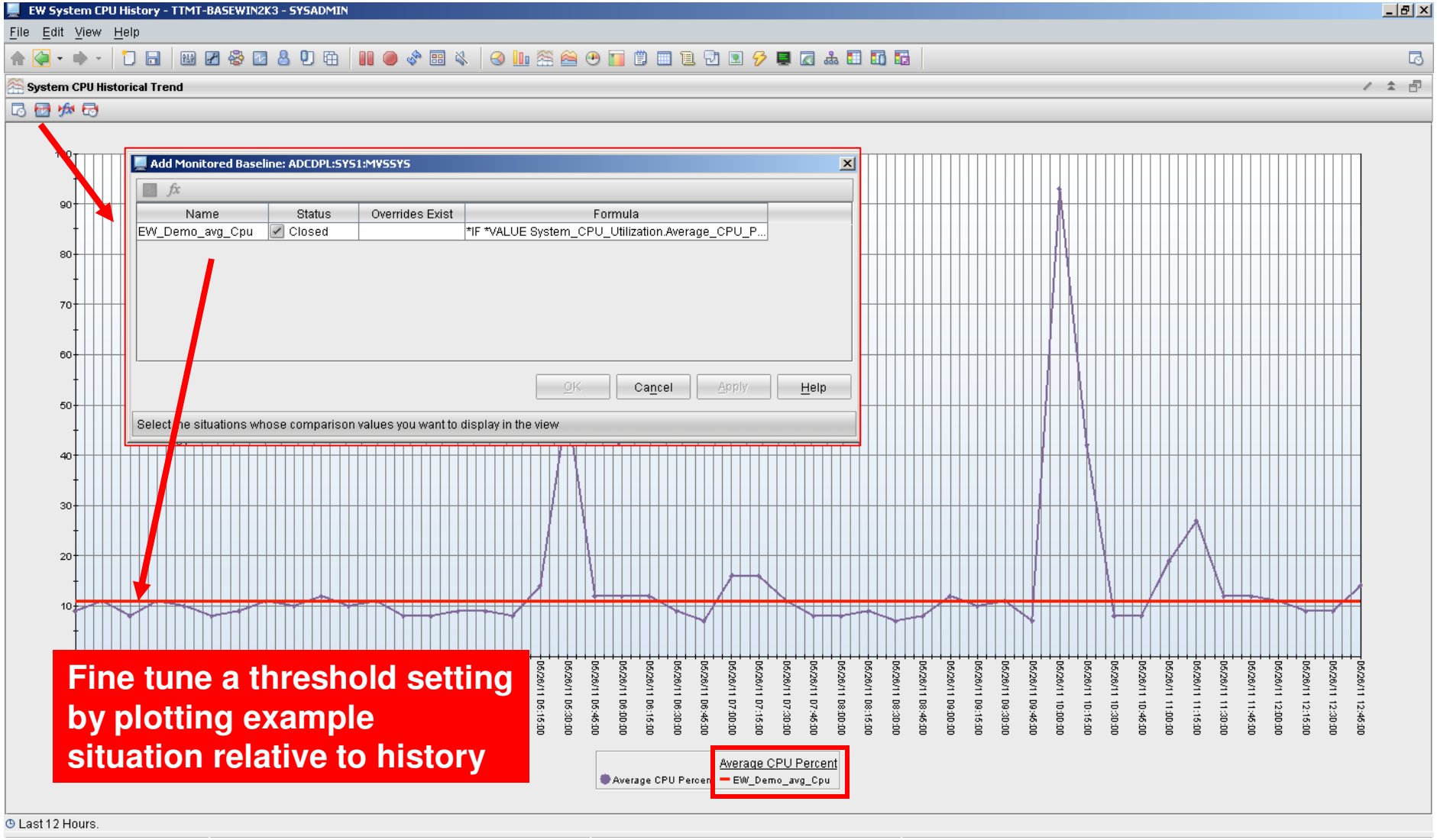
**Specify snapshot interval and frequency of sending data to TDW**

**Consider warehouse interval to avoid surges of data to TDW**

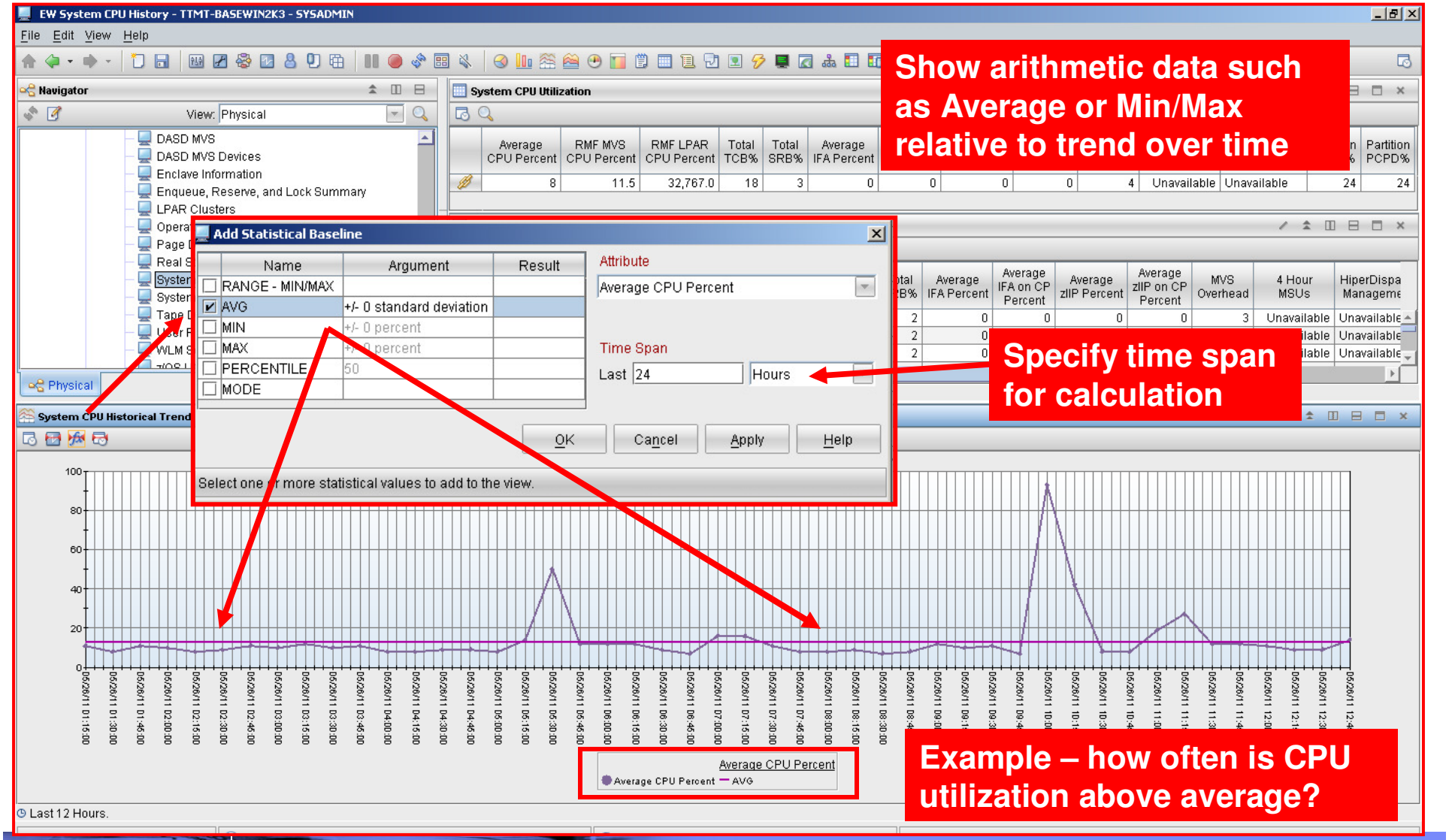
# Example - Use The TEP To Create A Custom Workspace As A Starting Point For Historical Data Analysis



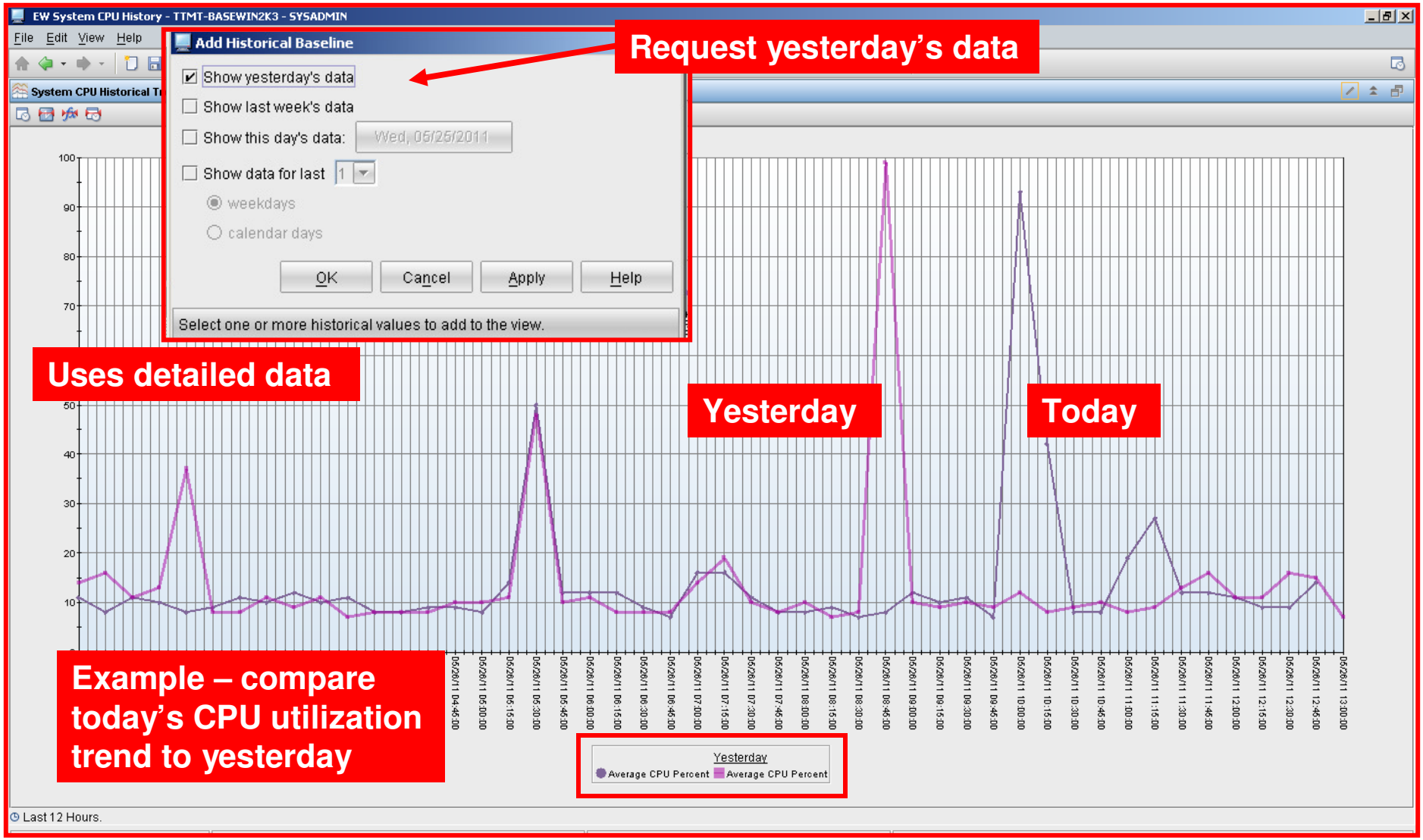
# Use A Situation To Track A Monitored Baseline Help Determine Where To Set A Threshold Level



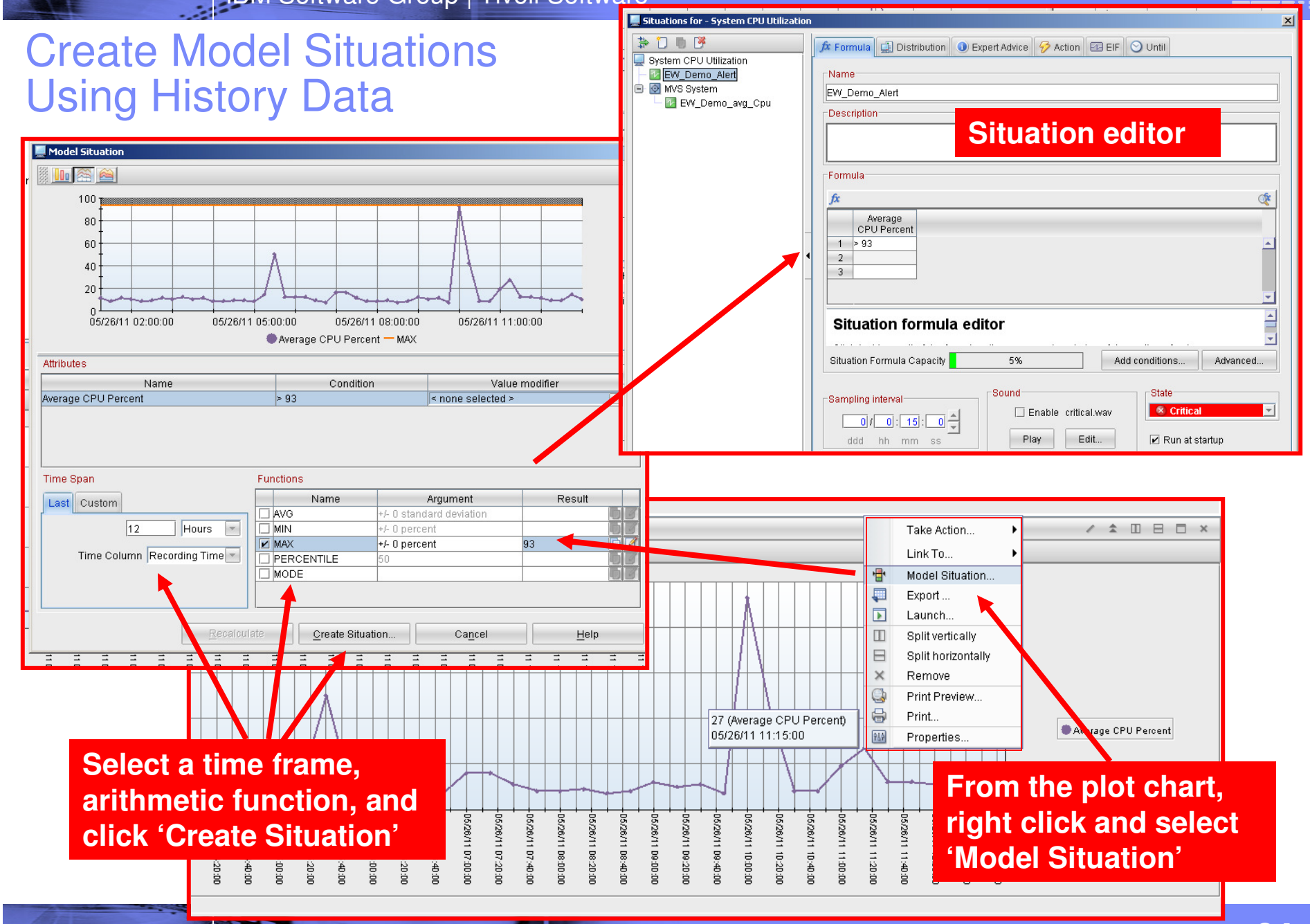
# Use Arithmetic Functions To Trend History



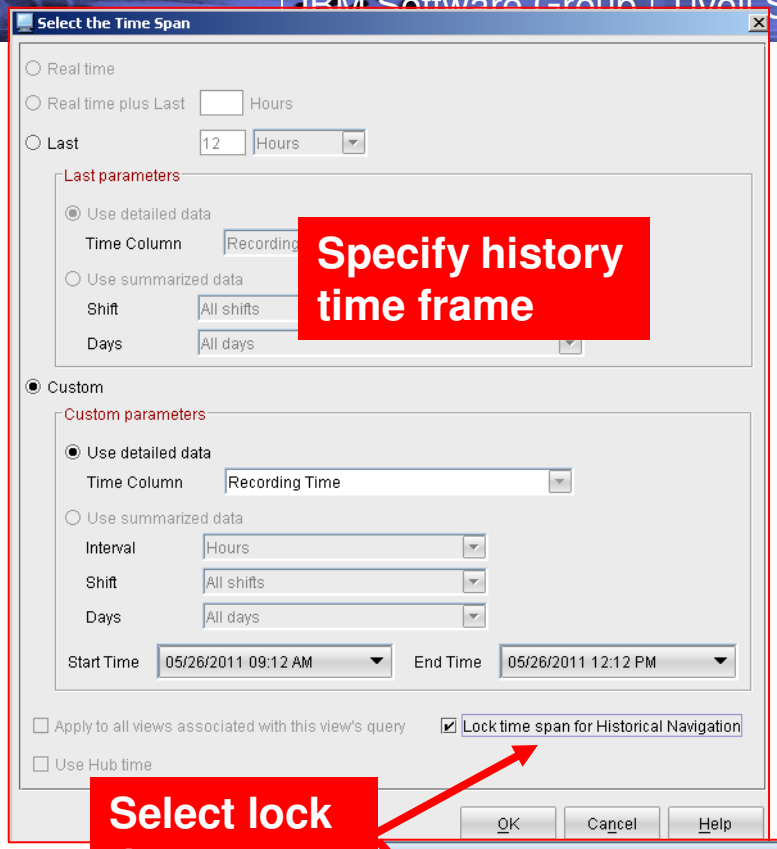
# Use Historical Baseline To Compare Past Trends To Current Trends



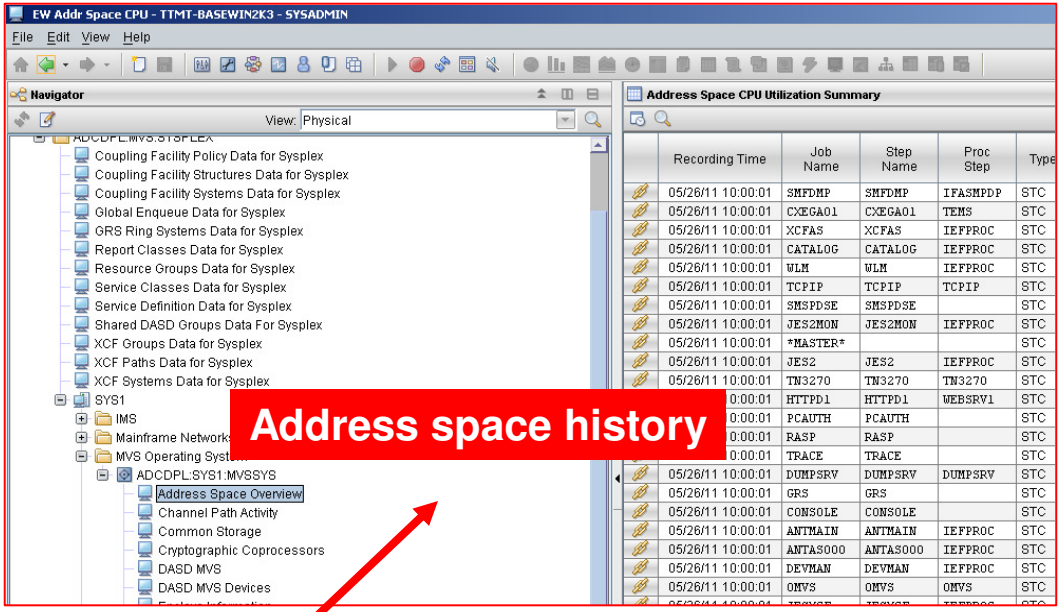
# Create Model Situations Using History Data



# History Links And Drill Down

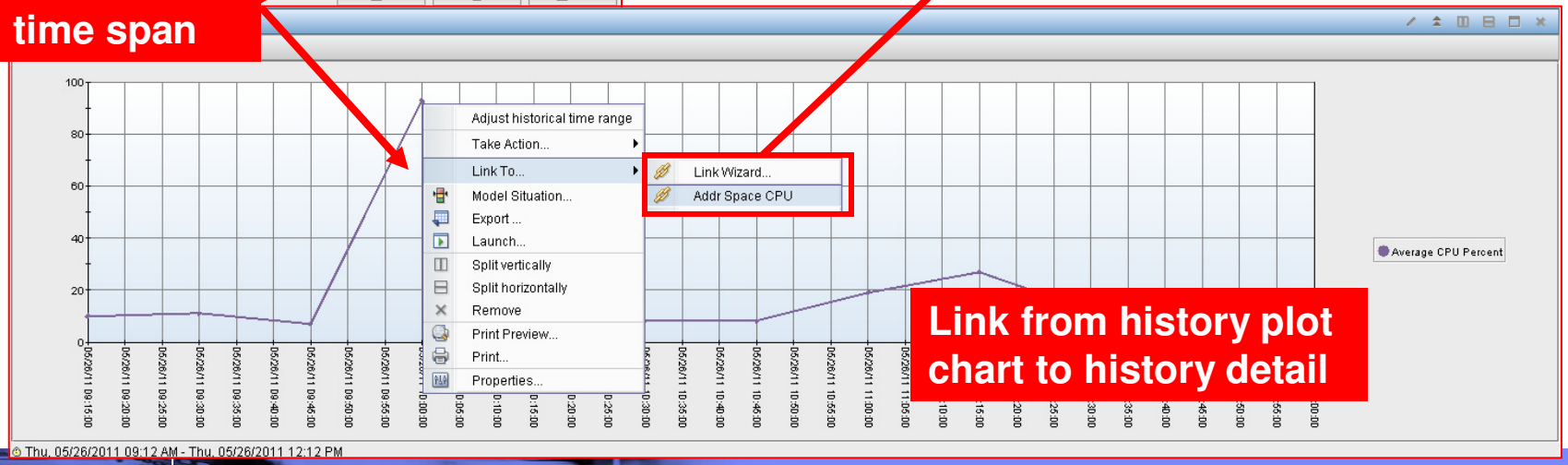



**Specify history time frame**




**Address space history**

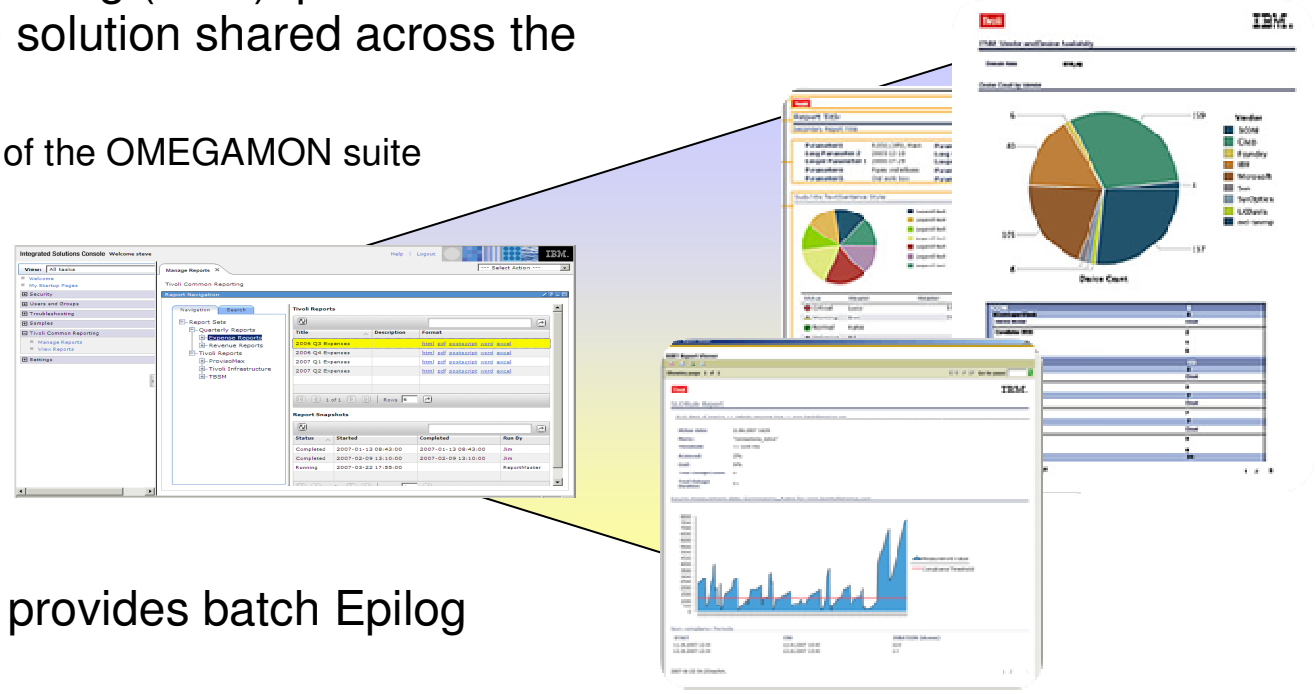
**Select lock time span**




**Link from history plot chart to history detail**

# Historical Reporting Options

- Tivoli Common Reporting (TCR) provides a consistent reporting solution shared across the Tivoli Portfolio
  - TCR included as part of the OMEGAMON suite



- OMEGAMON z/OS provides batch Epilog reporting
- OMEGAMON IMS provides batch Epilog reporting
- OMEGAMON CICS provides some basic reports
- OMEGAMON DB2 provides a detailed/robust best-of-breed historical reporting suite

# Improve Your Analysis

## Additional Components To Consider For Historical Data Collection And Analysis

- Tivoli Decision Support For z/OS
  - Generate customized reports to communicate system performance, capacity management, resource availability and cost allocation information
  - Collects data, such as SMF, CICS, IMS performance data
  - Provides a central data repository (DB2) and integrates with the Tivoli Portal
  - Integrates with a variety of Tivoli solutions
- IMS Performance Analyzer
  - Provides robust reporting and information on IMS system performance for monitoring, tuning, managing service levels, analyzing trends, and capacity planning
  - Expands the reporting options beyond what's available with OMEGAMON IMS
- CICS Performance Analyzer
  - Comprehensive performance reporting and analysis for CICS, including use of DB2, WebSphere MQ, IMS, and z/OS System Logger
  - Evaluate CICS system efficiency, eliminate system bottlenecks and proactively tune system performance
  - Expands the reporting options beyond what's available with OMEGAMON CICS

# Summary And Conclusions

- Each OMEGAMON monitoring solution offers history along with real time data collection
- Each OMEGAMON has it's own unique considerations specific to history collection and the data that is available
- History data collection is a classic trade-off of cost versus benefit
  - In some scenarios history data collection can be costly
  - Understand the costs versus the benefits
- History is essential to solve problems after the fact
- History is useful to make monitoring more proactive
  - Historical trending to identify peaks/valleys/issues
  - Historical data analysis to optimize alerts and thresholds

# Check Out My Blog

## http://tivoliwithaz.blogspot.com

# Thank You!