



# Assuring z/OS Environments with Customizable, Realtime Service Dashboards

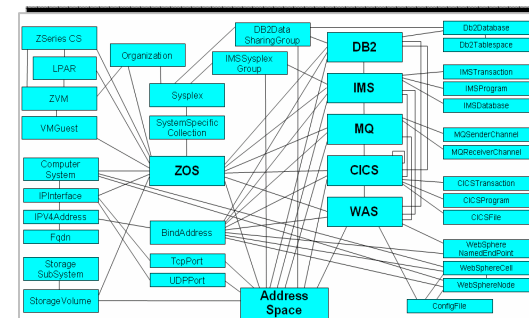
Clayton Ching  
Senior Product Manager  
IBM Business Service Management  
[CChing@us.ibm.com](mailto:CChing@us.ibm.com)

## Agenda

- Challenges Facing Operations
- The Value of Dashboards
- Overview of Tivoli Dashboards
  - Managing Virtualized Environments
  - SOA
- Getting Started with Out of the Box:
  - Discovery for z/OS
  - State and status Collection
- Defining Key Performance Indicators

# Common Challenges Facing Today's Operations

- Complexity of z/OS based Applications & Services:
  - One or more sub subsystems: CICS, IMS, MQSeries...
  - Supporting one or more Business Service(s)
  - Dependencies can span host and distributed environments
- Difficulty isolating root cause of problems within z/OS, and across host & distributed environments
- Limited insight into the impact of z/OS application & infrastructure problems on services, customers and revenue.
- No easy way to track delivery against key operational indicators and SLA commitments



*Operations teams require realtime visibility & intelligence for more effective decision making!*

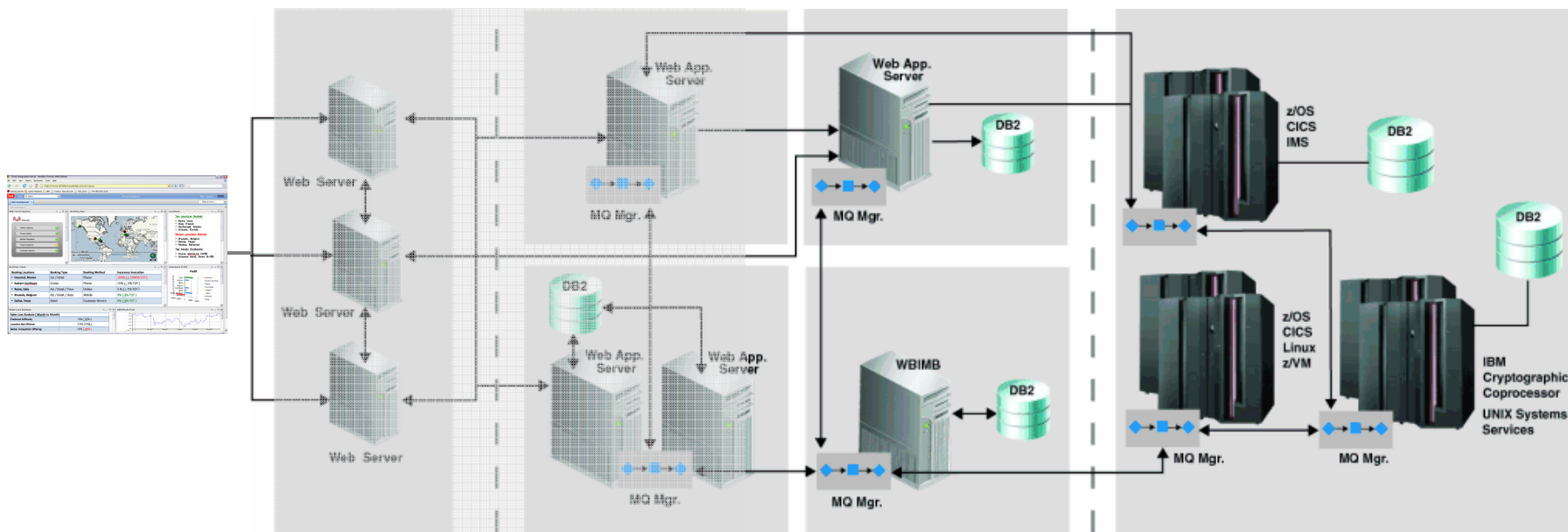
# Operations Must Watch Many Management Products/Consoles



## Distributed Resources

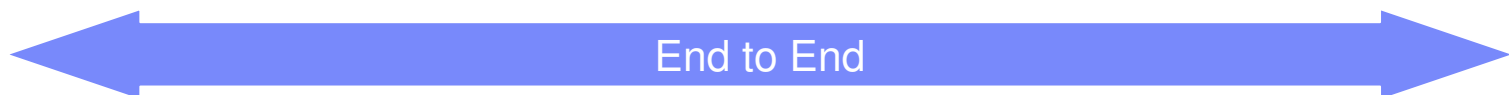
## Transactions

## Mainframe Resources



- In today's environment applications span End-to-End
- A variety of tools to help manage these applications

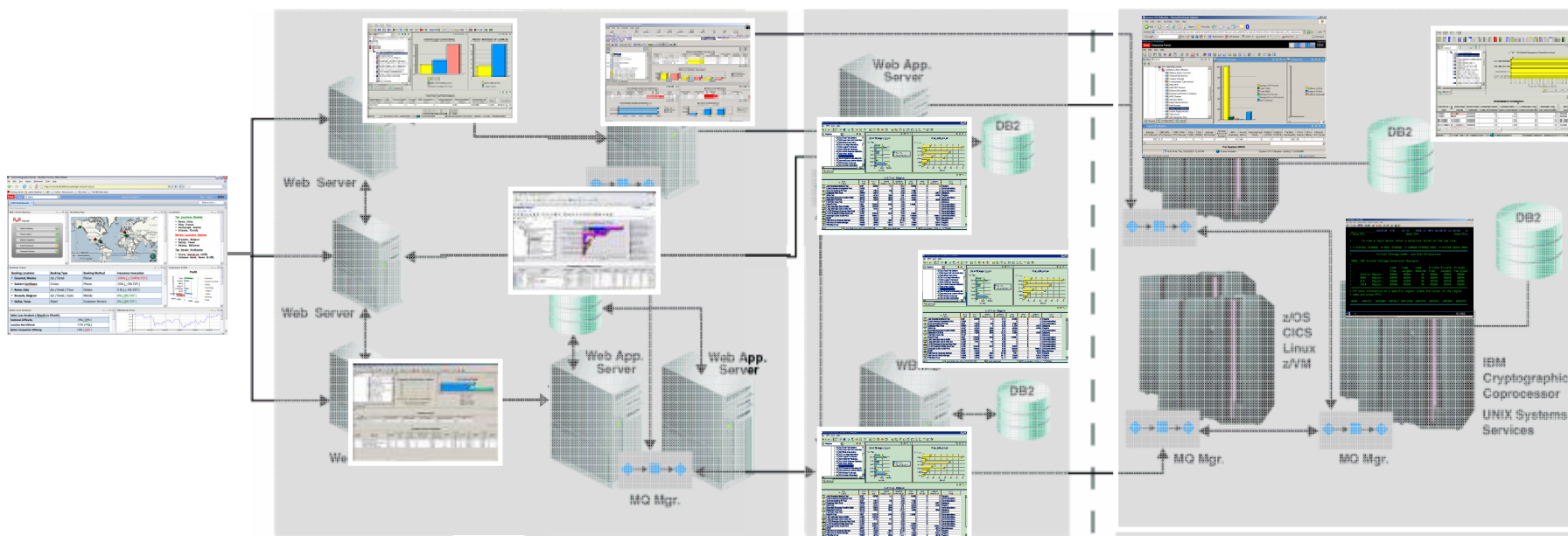
# Operations Must Watch Many Management Products/Consoles



## Distributed Resources

## Transactions

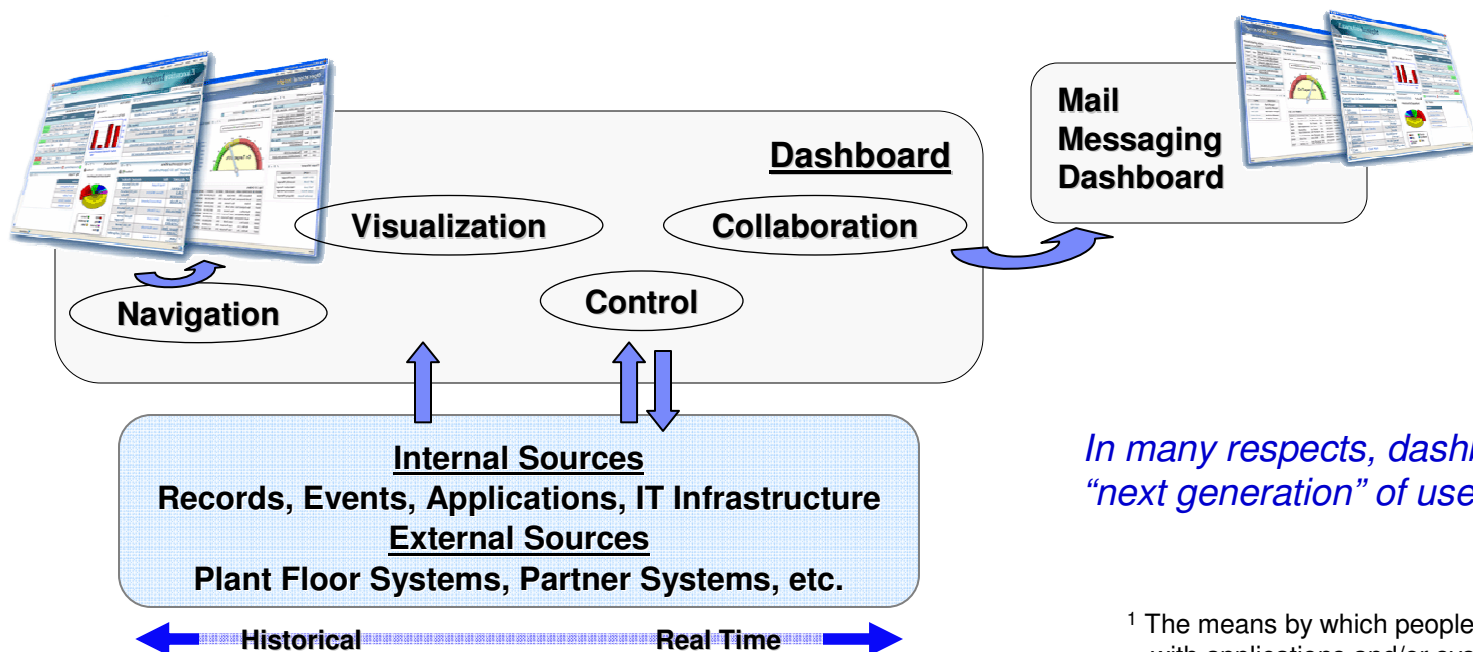
## Mainframe Resources



- In today's environment applications span End-to-End
- A variety of tools to help manage these applications
- When an event is received they have no idea of the impact to the business

# The Value of Dashboards: Visibility & Intelligence

- Present complex information in a simple, easy to understand & timely fashion;
  - *Insights that are easy to gain and still relevant!*
- Navigate a user through the varied sources and detail of that information;
  - *Insights that are deep!*
- Permit a user to take action based on that information – communicate it context to stakeholders or exercise control on operational applications, systems.
  - *Insights that result in collaboration, decisions, and responses!*



*In many respects, dashboards are the “next generation” of user interface.*

<sup>1</sup> The means by which people interact with applications and/or systems.



# Tivoli Customizable Realtime Dashboards

## Role-based dashboards

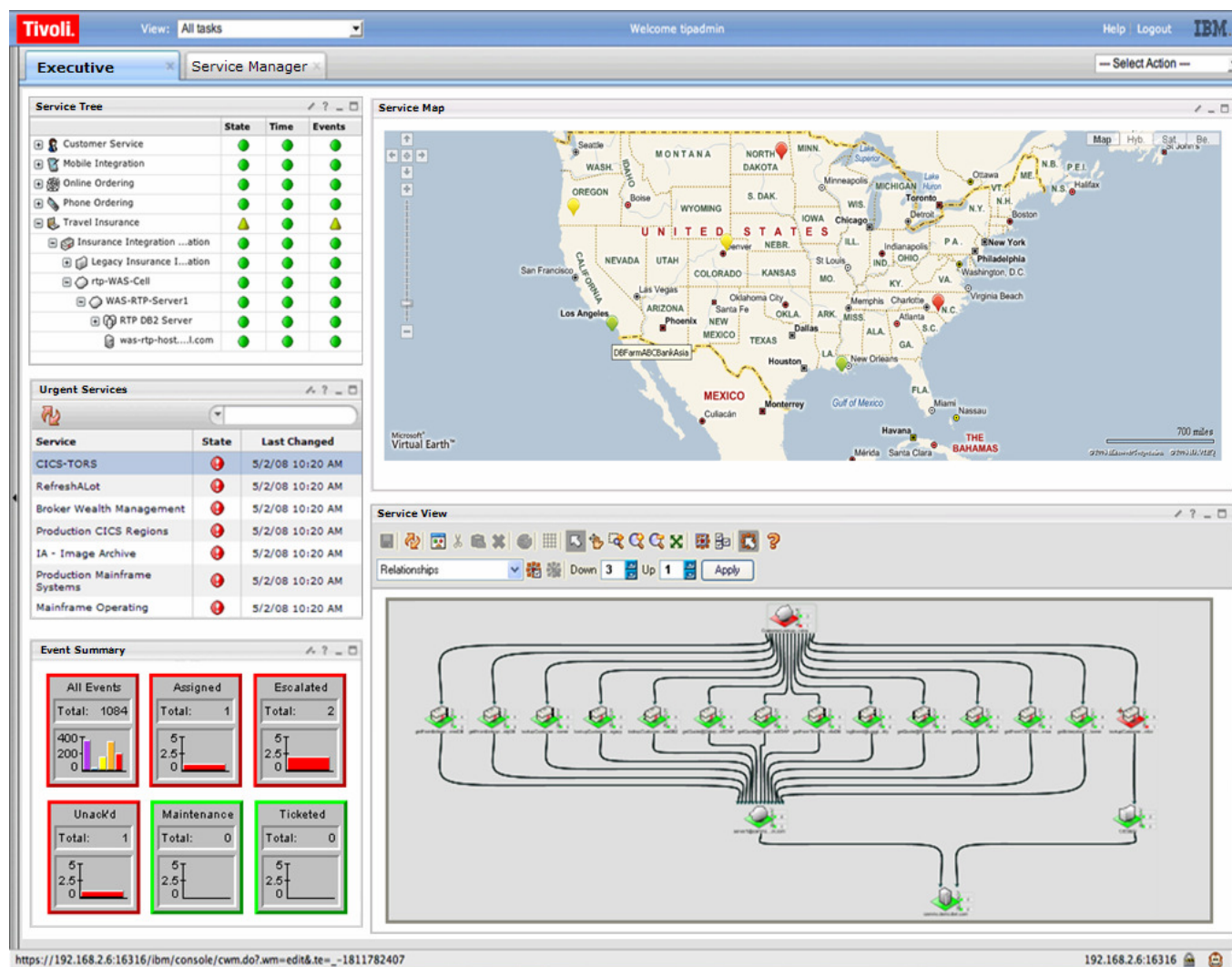
- Customizable/sharing common context
- Web 2.0/Mash-ups (IBM & 3<sup>rd</sup> party)
- Launch in context views & automations.
- Realtime & Historical reporting across KPIs, event & performance.
- Mobile Support

## Distributed & Mainframe

- Visibility across both
- Manage from either
- SOA & Virtualization
- Supports IPv4 & v6

## High Scalability/Availability

- Split UI & Engine
- Self-monitoring
- Failover



# Integrated Visibility & Intelligence

Low level Diagnostics

- Technical Support staff
- Systems Programmers

→ OMEGAMON Classic

```

Session B - [24 x 80]
File Edit View Communication Actions Window Help
KOISTOR VTM OI-II V550./C IMP1 03/29/07 11:18:55 B
> Help PF1 Back PF3 Zoom PF11
> To view a topic below, enter a selection letter on the top line.
> *-VIRTUAL STORAGE B-REAL STORAGE C-COMMON STORAGE AREA D-SYSTEM QUEUE AREA
=====
> Virtual Storage Under- and Over-Allocations
VMEM IMS Virtual Storage Constraint Analysis
+
+ Lsqa Lsqa Lsqa Private Private Private
+ Free Largest Assured Free Largest Top block
+ Control Region : 6580k 6580k 0k 6580k 6580k 6580k
+ DBRC Region : 8944k 8924k 0k 8924k 8924k 8924k
+ DLS Region : 8340k 8324k 0k 8372k 8324k 8324k
+ IRLM Region : 8540k 8524k 0k 8524k 8524k 8524k
=====
> For more information on a specific region, place the cursor on the region
> name and press PF11.
RGNA IMP1CTL IMP1DBR IMP1DLI IMPT1IRM IMP1FP2 IMP1FP1 IMP1MP1 IMP1FP3
=====
MA b 01/002
Connected to remote server/host.pthomo1 using lu/pool.TCPO.1005 and port 23
  
```



# Integrated Visibility & Intelligence

## Infrastructure Views

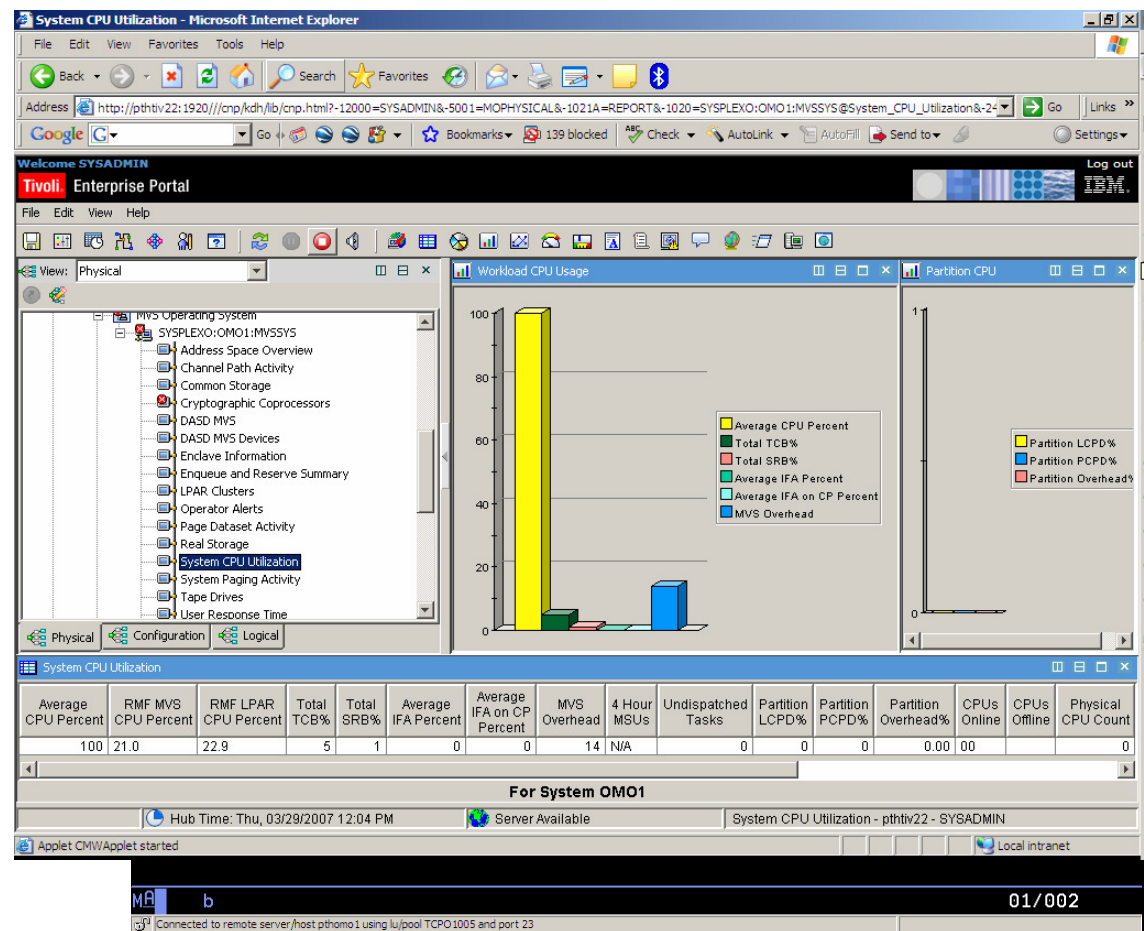
- Operations staff
- Technical Support staff

→ ITM & OMEGAMON XE

## Low level Diagnostics

- Technical Support staff
- Systems Programmers

→ OMEGAMON Classic



# Integrated Visibility & Intelligence

## Business Component Views

- Application Support staff
- Help Desk staff
- Operations staff

→ Tivoli Business Service Manager for z/OS

## Infrastructure Views

- Operations staff
- Technical Support staff

→ ITM & OMEGAMON XE

## Low level Diagnostics

- Technical Support staff
- Systems Programmers

→ OMEGAMON Classic

The screenshot displays the Tivoli Business Service Manager for z/OS interface. The main window is titled "Service Manager" and shows several key components:

- JVM Activity:** A 3D bar chart showing time allocated to tasks in seconds.
- Event Summary:** A dashboard with six event categories: All Events (Total: 1094), Assigned (Total: 1), Escalated (Total: 2), Unack'd (Total: 1), Maintenance (Total: 0), and Ticketed (Total: 0). Each category has a small bar chart.
- Transaction Summary:** A table listing transactions with columns for Inal, Task Number, Wait Type, Elapsed Time, CPU Time, Program ID, DB2 Correlation Identifier, PSB Name, and Recovery Token.
- Service View:** A hierarchical tree diagram showing the relationship between various services and components.
- SLA Details:** A table showing SLA status for various services, including Service Name, Best Case %, Downtime, TimeLeft, Twin, and Penalty.
- System Overview:** A summary table at the bottom showing system metrics such as Average CPU Percent, RMF MVS CPU Percent, RMF LPAR CPU Percent, Total TCB%, Total SRB%, Average IFA Percent, Average IFA on CP Percent, MVS Overhead, 4 Hour MSUs, Undispatched Tasks, Partition LCPD%, Partition PCPD%, Partition Overhead%, CPUs Online, CPUs Offline, and Physical CPU Count.

At the bottom of the interface, there is a status bar with the text "For System OMO1", "Hub Time: Thu, 03/29/2007 12:04 PM", "Server Available", "System CPU Utilization - pthtiv22 - SYSADMIN", and "Local intranet".

# Integrated Visibility & Intelligence

## High Level Business Views

- Business Managers
- End Users

→ Tivoli Business Service Manager for z/OS

## Business Component Views

- Application Support staff
- Help Desk staff
- Operations staff

→ Tivoli Business Service Mngner for z/OS

## Infrastructure Views

- Operations staff
- Technical Support staff

→ ITM & OMEGAMON XE

## Low level Diagnostics

- Technical Support staff
- Systems Programmers

→ OMEGAMON Classic

The screenshot displays the Tivoli Business Service Manager for z/OS interface. It features several key components:

- Service Tree:** A hierarchical view of services and their performance metrics.
 

Service	State	Infrastructure State	% Throughput vs. Baseline	Response Time	Historical Baseline	Total Tickets
EquityTrader	Green	Green	93%	463	432	125
London	Green	Green	92%	545	505	30
ET_CancelOrder	Red	Red	50%	122	61	0
ET_ChangeOrder	Green	Green	113%	125	141	0
ET_ExecuteBuyOrder	Yellow	Yellow	77%	127	98	0
ET_ExecuteSellOrder	Green	Green	122%	69	84	18
ET_GetQuote	Green	Green	150%	12	18	12
ET_Login	Green	Green	114%	90	102	0
New York	Green	Green	100%	373	374	38
Tokyo	Green	Green	88%	472	418	57
ExchangeTrading	Green	Green	62%	615	381	107
OnlineBanking	Green	Green	82%	424	349	14
- Event Summary:** A dashboard showing counts for various event categories:
 

Category	Total
All Events	1084
Assigned	1
Escalated	2
Unack'd	1
Maintenance	0
Ticketed	0
- Transaction Summary:** A table listing individual transactions with details such as Inal, Task Number, Wait Type, Elapsed Time, CPU Time, Program ID, DB2 Correlation Identifier, PSB Name, and Recovery Token.
 

Inal	Task Number	Wait Type	Elapsed Time	CPU Time	Program ID	DB2 Correlation Identifier	PSB Name	Recovery Token
00081	Interval	00:21:38.06	00:00:15.09	CICSD21	ENTROB210002	n/a	BA1C6FA590A224	
00046	TaskCtrl	07:31:59.85	00:00:00.00	KOCSR2ZZ	n/a	n/a	BA1C66C8C0B93531	
00030	MQSeries	07:32:03.78	00:00:00.00	KOCJO620	n/a	n/a	BA1C66C8C0B93531	
00029	MQSeries	07:32:03.78	00:00:00.00	CSQCTASK	n/a	n/a	BA1C66C535E11Bf	
00028	Terminal	07:32:03.78	00:00:00.01	CSQCAMON	n/a	n/a	BA1C66C50C157F7	
00025	Terminal	07:32:04.04	00:00:00.01	DFHZNAC	n/a	n/a	BA1C7CF590481A6	
00024	Terminal	07:32:04.79	00:00:00.03	DFHSHSY	n/a	n/a	BA1C66C41827E9E	
00023	Database	07:32:11.61	00:00:00.00	DFHD2EX2	n/a	n/a	BA1C66BD89851Bf	
- Service View:** A graphical representation of service relationships and dependencies, showing a hierarchy from 'Stock Trader' down to various underlying services like 'TradeApp' and 'TradeSubApp'.
- SLA Details:** A table providing specific Service Level Agreement (SLA) information.
 

Service Name	Best Case %	Downtime	TimeLeft	Twin	Penalty
Stock Trader	99.991	00:03:47s	00:56:12s	12-Aug-08	1538.12
Tradeapp	97.736	00:05:47s	00:22:15s	18-Aug-08	2404.97

The interface also includes a system status bar at the bottom showing 'For System OMO1', 'Hub Time: Thu, 03/29/2007 12:04 PM', and 'System CPU Utilization - pthtv22 - SYSADMIN'.



# Dashboarding for your Z/OS environment

The screenshot displays the Tivoli Service Administration web interface. The top navigation bar shows the user is logged in as 'Netcool Administrator' and is viewing 'Service Administration'. The main interface is divided into several sections:

- Service Navigation (Left Panel):** A tree view showing the hierarchy of services. Under 'Service Component Repository', the 'z/OS' folder is expanded, showing sub-objects like 'PERFPLEX:LP11:MVSSYS', 'LP11-z/OS', 'PERFPLEX', 'PLEX1401:0140:MVSSYS', '0140-z/OS', and 'PLEX1401'. These three items are circled in red, with a text box pointing to them containing the labels: 'z/OS OS Object', 'z/OS Computer System', and 'z/OS Sysplex'.
- Service Viewer (Right Panel):** Displays a hierarchical diagram of the 'SCR\_Servers\_zOS' service. At the top is a 'Servers' node, which connects to a 'z/OS' node. This 'z/OS' node branches into three intermediate nodes: 'PLEX1401:0140:MVSSYS', 'PERFPLEX:LP11:MVSSYS', and 'wlag.tivlab.ra...m.com'. Each of these further branches into specific system nodes: 'PLEX1401', '0140-z/OS', 'PERFPLEX', 'LP11-z/OS', and '-z/OS'.
- Event Summary (Bottom Left):** A chart titled 'Number of Events by Service and Severity for zOS'. The Y-axis is 'Number Of Events' (0 to 1). The X-axis is labeled 'PERFPLEX'. A legend indicates event severities: Clear (green), Indeterminate (purple), Warning (blue), Minor (yellow), Major (orange), and Critical (red).
- Service Details (Bottom Right):** A tabbed interface with 'SLA', 'Events', and 'Rules' tabs. The 'Events' tab is active, showing a URL: [http://cvtwin53.tivlab.raleigh.ibm.com:8080/RawEvents\\_32](http://cvtwin53.tivlab.raleigh.ibm.com:8080/RawEvents_32) and a table with columns: Node, Summary, AlertKey, Class, Manager, and S.

# Dashboarding for your CICS Environment

The screenshot displays the Tivoli Service Administration console. The top navigation bar shows the user is logged in as 'Netcool Administrator' and is in the 'Service Administration' section. The left-hand 'Service Navigation' pane shows a tree view of the 'Service Component Repository' with 'CICS' selected under 'Application Servers'. A red oval highlights 'CICST11A-LP11', which is labeled as the 'CICS Subsystem'. Below this, a list of other CICS instances is visible, including LP11-PERFPLEX, LP11-z/OS, and various LP11 instances (CICST11B-LP11 through CICST11E-LP11, and CMAST11-LP11).

The main 'Service Viewer' pane displays a hierarchical diagram of the 'SCR\_BSM\_CICS' service. At the top is the 'Application Servers' node, which connects to a 'CICS' node. This 'CICS' node is further connected to a 'LP11-PERFPLEX' node, which in turn connects to several 'PERFPLEX' nodes. The diagram illustrates the service's architecture and dependencies.

At the bottom left, an 'Event Summary' chart titled 'Number of Events by Service and Severity for CICS' is shown. The chart has a legend with categories: Clear (green), Indeterminate (purple), Warning (blue), Minor (yellow), Major (orange), and Critical (red). The x-axis lists various CICS instances, and the y-axis represents the 'Number Of Events'.

The bottom right section of the interface shows 'Service Details' for the selected service, with tabs for 'SLA', 'Events', and 'Rules'. A URL is provided: [http://cvtwin53.tivlab.raleigh.ibm.com:8080/RawEvents\\_16](http://cvtwin53.tivlab.raleigh.ibm.com:8080/RawEvents_16). Below this is a table with columns for 'Node', 'Summary', 'AlertKey', 'Class', 'Manager', and 'S'.

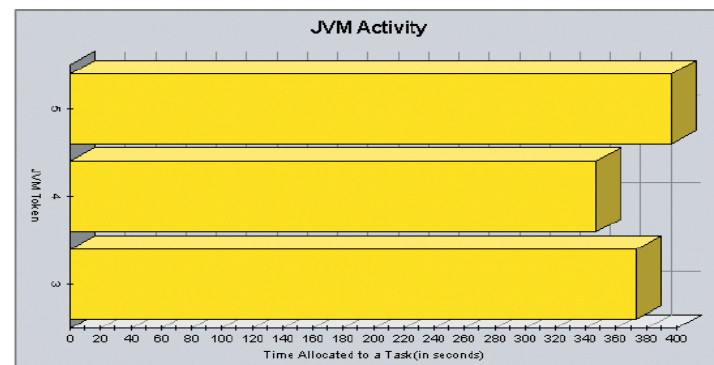
# Integrated Drill Down to Detailed Intelligence

Launch in Context from:

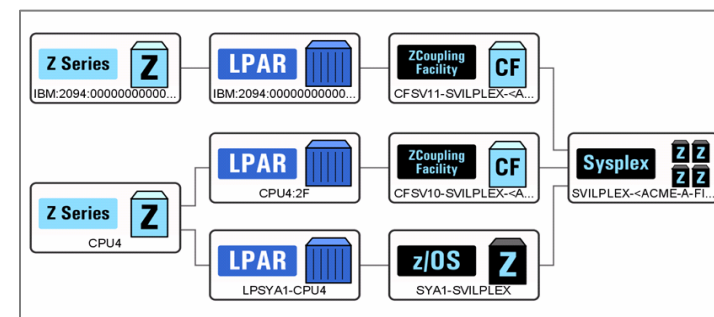
- Dependency Model
- Event Views
- SLA Views
- Custom Maps

Directly to detailed views:

- Transactions
- Performance
- Configuration Details
- Change Reports



Transaction Analysis									
inal	Task Number	Wait Type	Elapsed Time	CPU Time	Program ID	DB2 Correlation Identifier	P&B Name	Recovery Token	
	00081	Interval	00:21:38.06	00:00:15.09	CICSD821	ENTRDB210002	n/a	BA1CC6FA590A224	
	00046	TaskCntl	07:31:59.85	00:00:00.00	KOCSR2ZZ	n/a	n/a	BA1C66C8C0B3534	
	Lnk to DB2 Threads	TaskCntl	07:31:59.85	00:00:00.00	KOCSR2ZZ	n/a	n/a	BA1C66C8C0B3534	
	00030		07:32:03.78	00:00:00.00	KOCJO620	n/a	n/a	BA1C66C500EFCF1	
	00029	MQSeries	07:32:03.78	00:00:00.00	CSQCTASK	n/a	n/a	BA1C66C535EE1BF	
	00028		07:32:03.78	00:00:00.00	CSQCAMON	n/a	n/a	BA1C66C50C157F7	
	00025	Terminal	07:32:04.04	00:00:00.01	DFHZNAC	n/a	n/a	BA1C7FC590481A6	
	00024		07:32:04.79	00:00:00.03	DFHSHSY	n/a	n/a	BA1C66C41827E9E	
	00023	Database	07:32:11.61	00:00:00.00	DFHD2EX2	n/a	n/a	BA1C66BD89851B4	
	00034	MPC	07:32:14.00	00:00:00.00	DFHCPMP	n/a	n/a	BA1C66A8023100	



# Managing Service Oriented Architectures

The screenshot displays the Tivoli Service Manager interface with the following components:

- Service Tree:** A hierarchical list of services including StockTrader, OnlineTrade, Databases, TradeApp, and TradeService.
- Service View:** A graphical diagram showing the relationships between services, such as StockTrader depending on OnlineTrade, which in turn depends on Databases and TradeApp.
- Recent Performance History:** A line graph showing the quality of the StockTrader service over time.
- Revenue by City:** A bar chart comparing financial metrics (Dollars, Growth, Profit) across five cities: Boston, New York, Chicago, Miami, and Los Angeles.
- Service Details:** A table showing SLA status for various services.

Entity:	SLA	Events		Rules	
View:	Best Case %	Downtime	TimeLeft	Twin	Penalty
SLAStatus_14	View: SLA Status	DataSource: NCOMS			
Service Name	Best Case %	Downtime	TimeLeft	Twin	Penalty
Stock Trader	99.991	00:03:47s	00:56:12s	12-Aug-08	1538.12
Tradeapp	97.736	00:05:47s	00:22:15s	18-Aug-08	2404.97

Drill down to SOA services & servlets

Access Business Events, KPIs & views from SOA platforms & middleware

Auto-discovery and mapping of SOA-based business services & processes (BPEL)

SLA Tracking by service and sub-service components including best case, downtime, penalty...



# Managing Service Oriented Architectures

The screenshot displays the Tivoli Service Manager interface. On the left, the 'Service Tree' shows a hierarchy of services including StockTrader, OnlineTrade, Databases, TradeApp, and TradeService. The 'Recent Performance History' chart shows a fluctuating quality score for StockTrader. The 'Revenue by City' bar chart compares Dollars, Growth, and Profit across five cities. The 'Operation Flow' diagram illustrates a sequence of operations: lookupCustomerclient, lookupCustomer, getFromCICSService, and lookupCustomericss.

Rapidly identify the service impact on SOA-based business services, processes or transactions.

Pinpoint Service-impacting root cause.

LIC to Operation flow diagrams & details.

# Relational Value Across Tivoli & Third Party Tools...



## Integrated Visualization & Navigation

*Web 2.0 interface with launch in context across IBM tools and 3<sup>rd</sup> party views.*



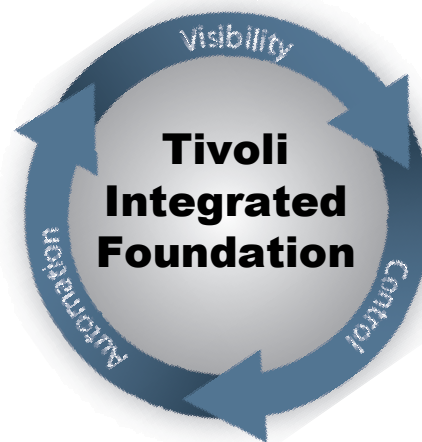
## Process Automation

*Common platform for cross-product integration, processes, & task / runbook automation.*



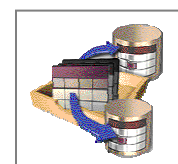
## Integrated Security

*Secure single sign-on across products.*



## Open Reporting Ecosystem

*Out of the box and custom reports leveraging IBM data warehouse and 3<sup>rd</sup> party data sources.*



## Common Data Warehouse

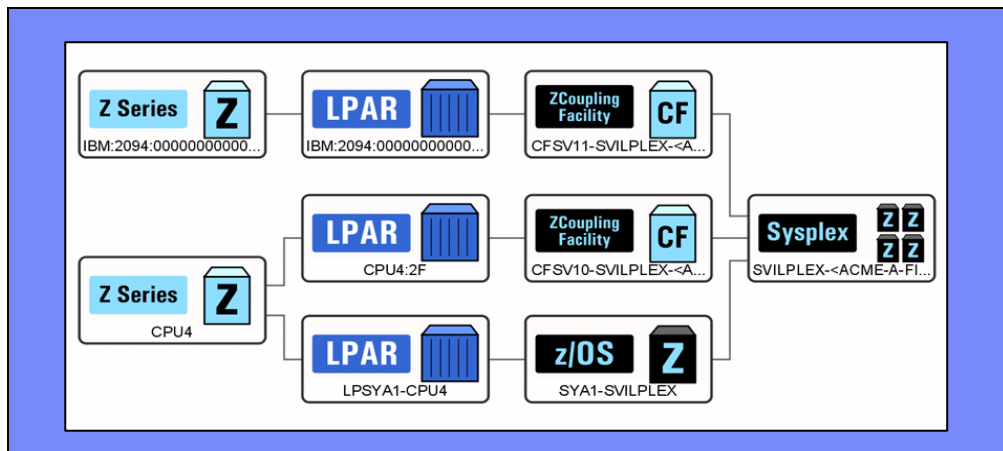
*Stores IBM and 3<sup>rd</sup> party event, performance, & business data for predictive analytics.*

**Only Tivoli has the integrated & extensible foundation to manage across Tivoli & 3<sup>rd</sup> Party**

# Tivoli Discovery for Automated Mapping of Dependencies

## Breadth of discovery:

- Mainframe
- Distributed
- SOA
- Virtualization
- Storage
- Network
- Security

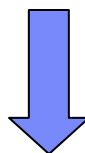


## Topology Mapping:

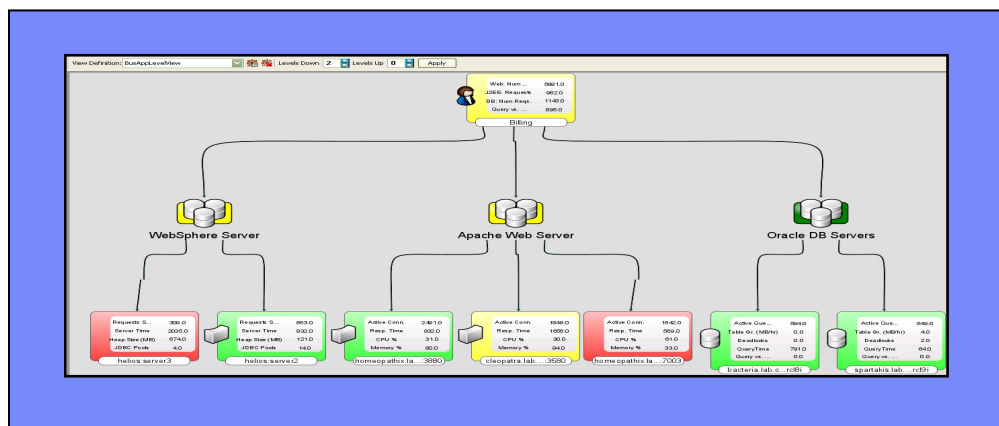
- Periodic
- Partial/Full
- Manual

## Configuration Details & Change History

*Cross tier application maps  
Configuration changes*



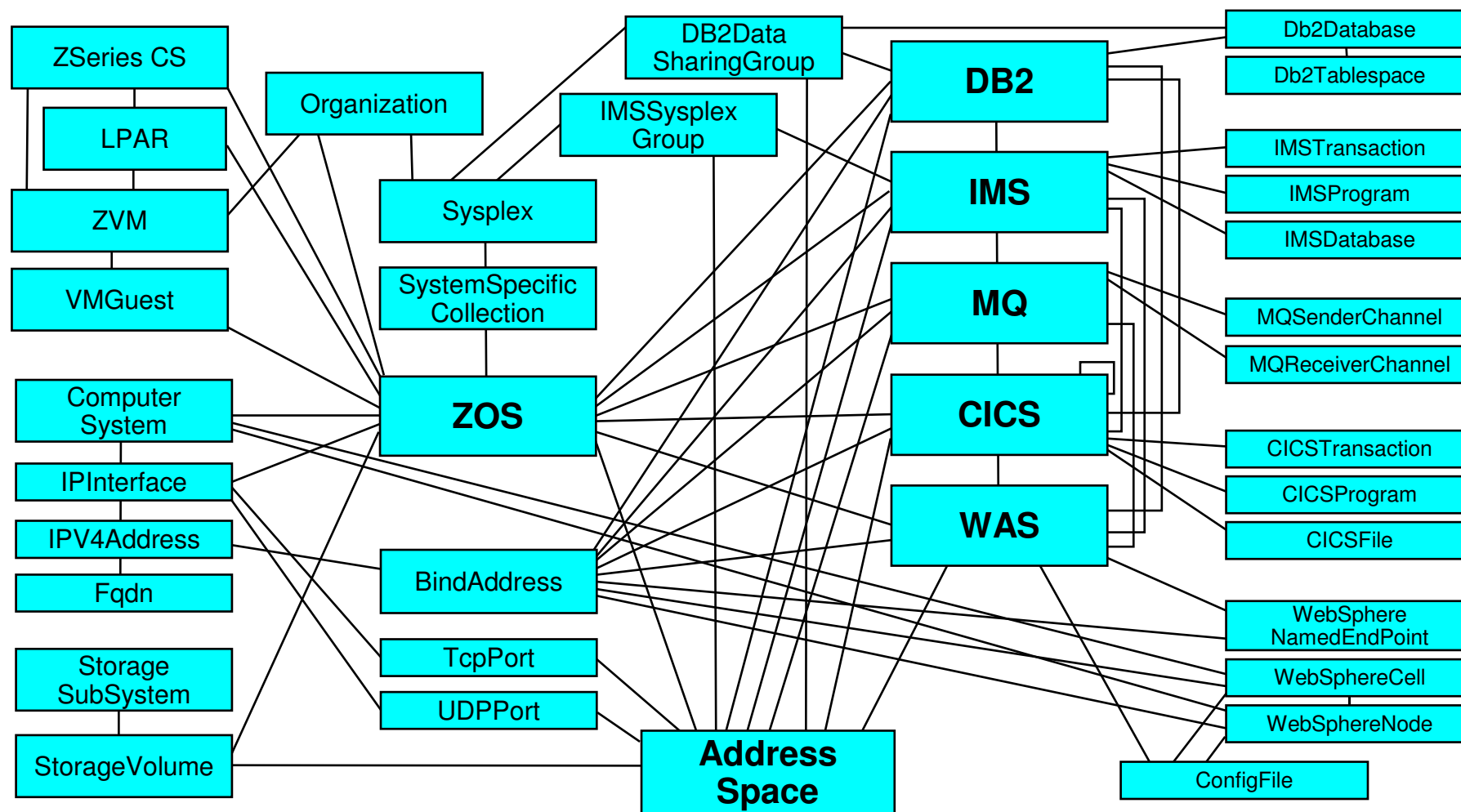
*Launch in context to  
configuration details panels*



## Sources of Dependency Data for z/OS

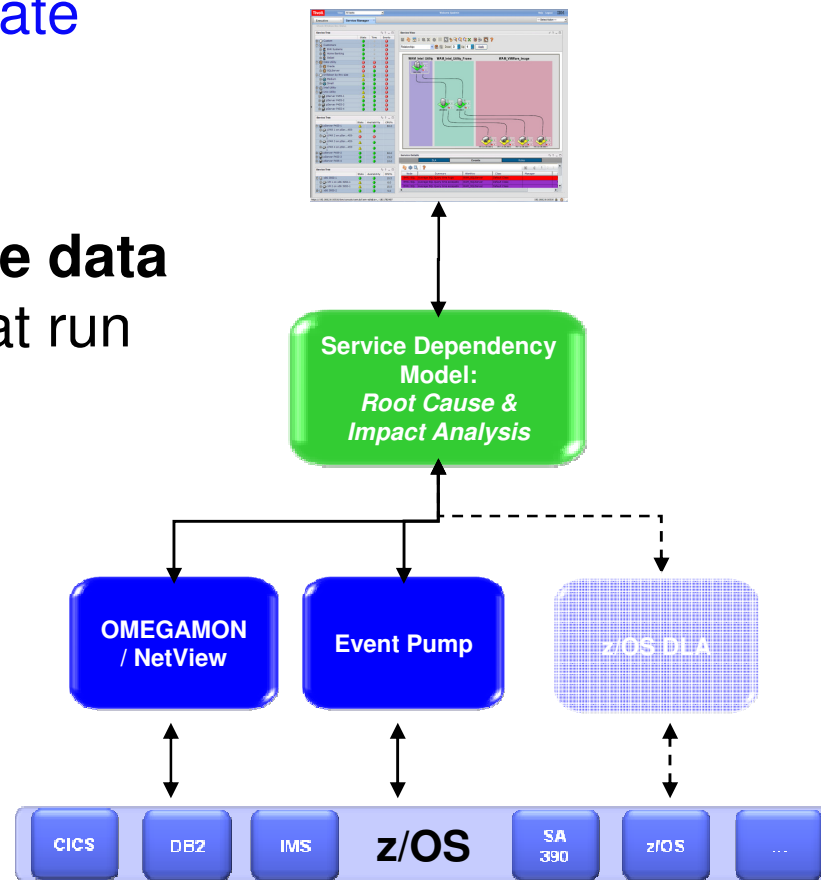
- Discovery Library Adapters (DLAs)
  - On OPAL – OPEN PROCESS APPLICATION LIBRARY
  - Discover Resources and Relationships
  - Aim for zero prerequisites
  - Creates an XML file that is in Identity Markup Language (iDML) format, which conforms to the Common Data Model (CDM)
  
- z/OS DLA – [Out-of-the-box Discovery of z/OS Dependencies](#)
  - Discovers z/OS Hardware and z/OS Details
  - Address Spaces
  - Subsystems: DB2, IMS, MQ, CICS, WAS
  
- TMS DLA
  - Discovers Tivoli Monitoring Services resources
  - All Managed Systems: including Distributed Agents and OMEGAMON XE mainframe agents. Logical Groupings.
  
- IBM Tivoli NetView for z/OS DLA
  - Discovers System z IP Managed Element data

# Breadth of Discovery & Connections



## Collect State and Status: Event Pump for z/OS

- Out-of-the-box Collection of z/OS State and Status
- Runs on z/OS (mainframe)
- Retrieves **state** and **status resource data** for various subsystems and tools that run on z/OS, such as:
  - z/OS
  - Base CICS
  - CA OPS/MVS
  - IMS\*
  - DB2\*
  - SA/390\*
- *\*requires NetView for z/OS 5.x*
- Direct Support for OMEGAMON XE
- Any External Distributed Sources







ERROR: undefined  
OFFENDING COMMAND: f'~  
STACK: