Understanding The Interaction Of z/OS Workload Manager And DB2

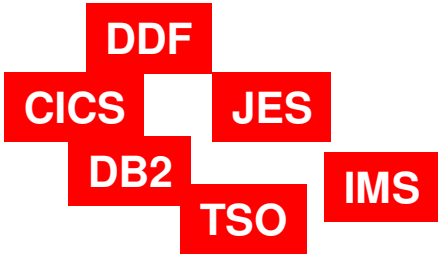
Ed Woods
IBM Corporation

Friday, March 4, 2011
Session 8851

Agenda

- Workload Manager Overview
- Important WLM Concepts And Terminology
- How DB2 Exploits Workload Manager
- Understanding WLM and how it impacts DB2
- WLM/DB2 Usage Recommendations
- Summary

Why Workload Manager (WLM)?



Goal Prioritization

What does z/OS need to accomplish objectives?
How important is the workload to the business?

- Complexity of systems has increased dramatically
  - In the past priority managed by PARMLIB options
  - Too much work to 'micro-manage' each z/OS system
  - Multiple LPARs, Data sharing, large sophisticated workloads
- Workload Manager improves the performance management process
  - Prioritize workload based upon goals and business objectives
  - Let the system optimize and prioritize resource management
  - Make sure that the most important workload gets the appropriate resource

DB2 And Workload Manager

- Workload Manager (WLM) is the priority and resource manager for z/OS and implicitly for DB2 as well
- Workload Manager manages CPU, I/O, and memory resources as needed by the workload
- WLM manages critical DB2 address spaces
  - DB2 subsystem address spaces
  - DB2 Stored Procedure address spaces
- These resources have an impact on how DB2 applications perform
- Workload Manager is used to manage DB2 workflow
  - DB2 Distributed and DB2 Stored Procedure workload
  - Priority and performance of allied address spaces (CICS, IMS, batch, TSO) that call DB2
- Important to have a basic understanding of WLM

Workload Manager
Terminology

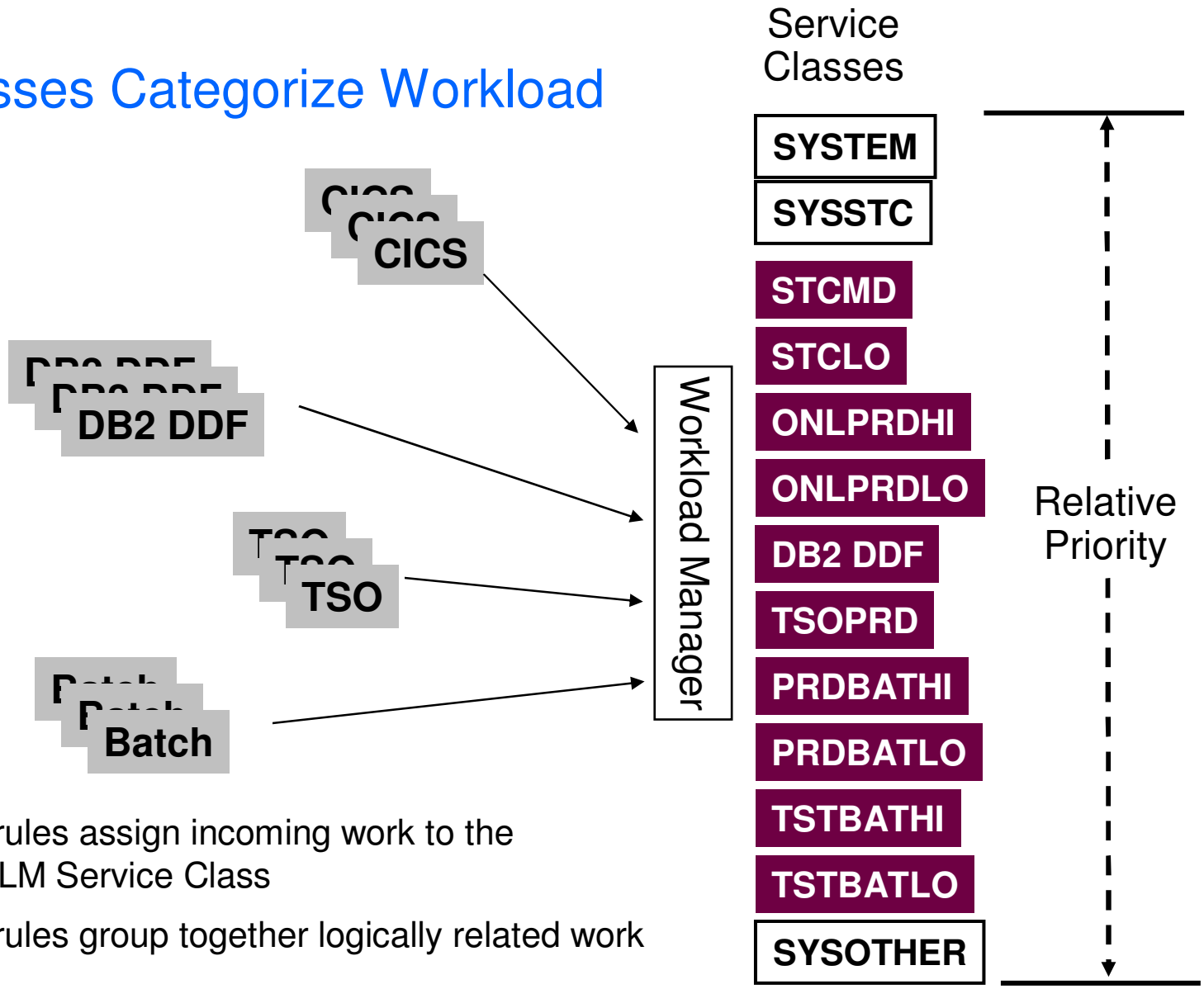
- A SERVICE DEFINITION consists of one or more SERVICE POLICIES
- A service policy contains several WORKLOADS
- Each workload consists of one or more SERVICE CLASSES
- Each service class has at least one PERIOD and each period has one GOAL
- There are five types of goals
  - System, Average Response Time, % Response Time, Execution Velocity, Discretionary
- Address spaces and transactions are assigned to service classes by CLASSIFICATION RULES

SERVICE DEFINITION
SERVICE POLICIES
WORKLOADS
SERVICE CLASSES
PERIOD
GOAL
CLASSIFICATION RULES

Key WLM Concepts

- WLM is built upon two key concepts
  - ***DEFINITION*** - WLM provides mechanisms to categorize, prioritize, and manage workload
    - These are the service definitions managed by the WLM dialogs
  - ***FEEDBACK*** - The components managed by WLM provide information (samples) to help WLM determine how well it is doing
    - These are feedback mechanisms provided by each component, subsystem, and operating system

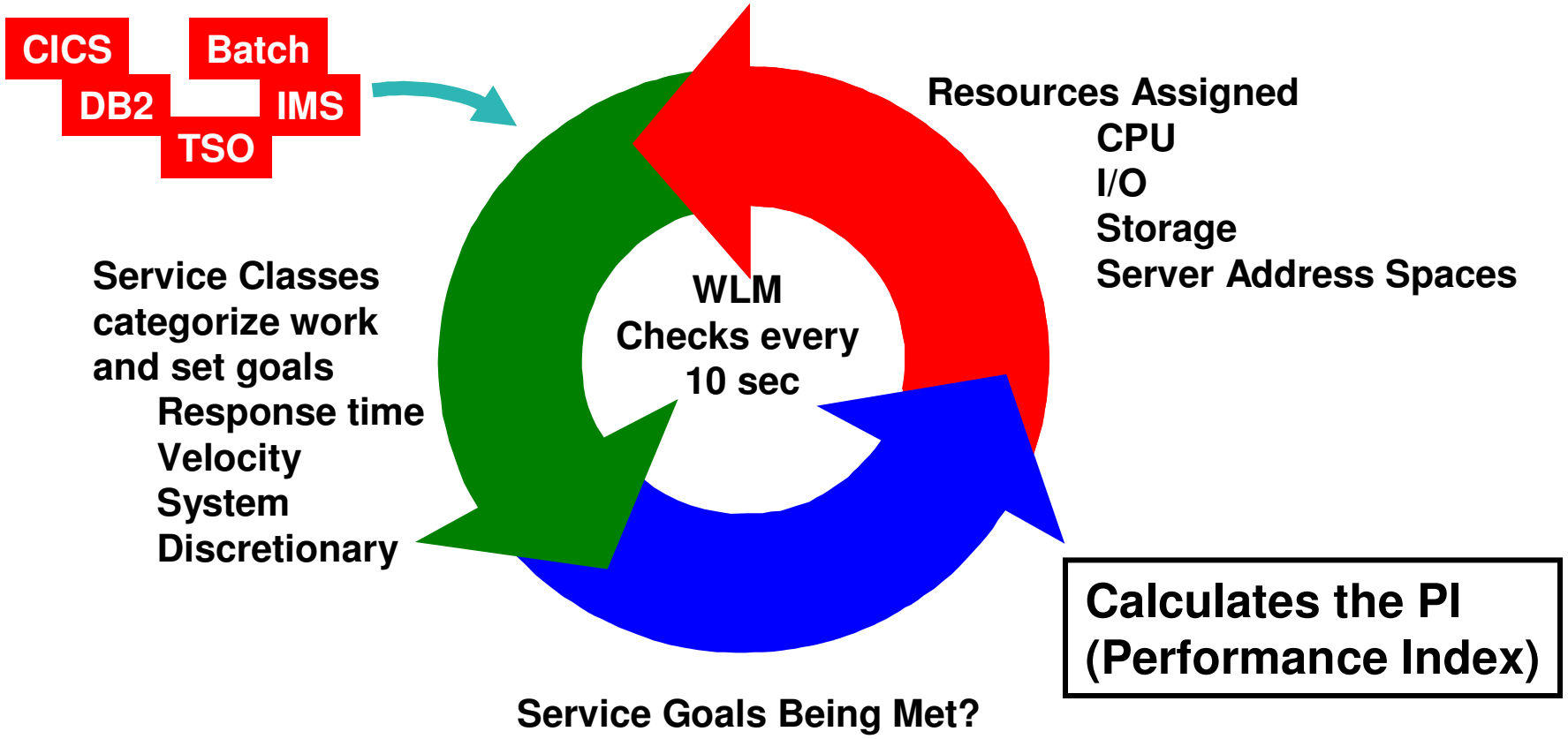
WLM
Service Classes Categorize Workload



- Classification rules assign incoming work to the appropriate WLM Service Class
- Classification rules group together logically related work

Workload Manager Service Classes And Goals

- z/OS resources assigned based upon goals defined in WLM



Understanding WLM Goals
The Performance Index

- Service Class periods are compared by calculating a Performance Index (PI) for each
- PI gives WLM a common way to track how well the work is doing regardless of goal type
- Importance parameter
  - Defined as part of the Service Class - 1 (high) to 5 (low)
  - Assigned to a Service Class Period
  - A way to prioritize critical goals
  - For work at the same importance level, WLM attempts to equalize the PIs

• **The PI equals 1**	**=> The work in the period is meeting its goal exactly**
• **The PI is less than 1**	**=> The work is doing better than its goal**
• **The PI is more than 1**	**=> The work is missing its goal**

The Importance Of Importance

- Importance parameter
  - A way to prioritize critical goals
- Not analogous to Dispatching Priority
  - What is the importance of achieving the goal?
- WLM attempts to meet importance 1 goals first, and so on…….
- Helps WLM determine donors and receivers of resources
  - Donors – workload that can give up resource
  - Receivers – workload that needs resource

Types Of WLM Goals
Velocity Goals

- The percentage of time workload is ready and able to run, and is not delayed for lack of resources
  - Example - Velocity of 50 means that 50% of the time resources should be available for work to run
- Velocity goals measure of acceptable delay based on samples
  - High velocity goals (example 90) in general are unreasonable
  - Use relatively higher velocity goals for DBM1 and SSAS
  - At first glance easy to set
- Velocity goals actually requires more analysis
  - Systems, environments, and workloads will change over time
  - Velocity goals require regular review
  - As systems change, velocity goals may need ‘fine-tuning’

Types Of WLM Goals
Response Time Goals

- Average response time
  - Average response time for a given set of transactions
  - Include queue time and execution time
- Percentile response time
  - Percentile of transactions that need to complete within a desired response time
  - Reduces the impact of ‘outliers’
- Rule of thumb
  - Work should have at least 10 completions in a 20 minute time frame to have adequate samples
- Consider Response time goals where possible for DB2 workloads
  - DDF requests, even batch jobs

Special Service Classes SYSTEM, SYSSTC, & SYSOTHER

- SYSTEM
  - For selected high priority system address spaces
  - Get highest CPU and I/O dispatching priority in system
- SYSSTC
  - For selected high priority started tasks and workload
  - Second highest priority behind SYSTEM
  - Place very high importance workload items here
    - DB2 Example - place IRLM here
- SYSOTHER
  - Unclassified work falls here
  - Bottom of the resource food chain

Specifying The WLM Objectives

```
File  Utilities  Notes  Options  Help
------------------------------------------------------------------------
Functionality LEVEL011          Definition Menu          WLM Appl LEVEL013
Command ===> ____________________________________________________________

Definition data set  . . : none

Definition name  . . . . . DEMO       (Required)
Description  . . . . . . . No overrides

Select one of the
following options. . . . . ___  1.   Policies
                               2.   Workloads
                               3.   Resource Groups
                               4.   Service Classes
                               5.   Classification Groups
                               6.   Classification Rules
                               7.   Report Classes
                               8.   Service Coefficients/Options
                               9.   Application Environments
                               10.  Scheduling Environments
```

All the various WLM constructs on z/OS are defined using the WLM ISPF dialogs

- WLM provides an ISPF interface to define and manage the WLM Service Definition
- Note – z/OS Management Facility V1.12 provides a new management interface

Workload Manager As A DB2 Priority Mechanism

```
  Subsystem-Type  Xref  Notes  Options  Help
------------------------------------------------------------------------------
                Create Rules for the Subsystem Type          Row 1 to 5 of 5

Subsystem Type . . . . . . . . DDF     (Required)
Description  . . . . . . . . . Example DB2
Fold qualifier names?  . . . . Y   (Y or N)

Enter one or more action codes: A=After  B=Before  C=Copy  D=Delete
M=Move I=Insert rule IS=Insert Sub-rule  R=Repeat

          -------Qualifier-------------              -------Class--------
Action    Type        Name      Start                Service      Report
                                                DEFAULTS: PRDBATCH     _________
 _____  1  SI          DB2A      ___                  PRDBATCH     _________
 _____  2    CN          ONLINE    ___                PRDONLIN     _________
 _____  2    UI          SYSADM    ___                PRDONLIN     _________
 _____  2    PK          QMFOS2    ___                TSTQUERY     _________
 _____  1  SI          DB2B      ___                  TESTUSER     _________
```

Examples of Thread Attributes

- **AI (Accounting Information)**
- **CI (Correlation Information)**
- **CN (Collection Name)**
- **CT (Collection Type)**
- **LU (LU Name)**
- **NET (Net ID)**
- **PK (Package Name)**
- **PN (Plan Name)**
- **SI (Subsystem Instance)**
- **UI (Userid)**

Thread attributes in WLM allow for considerable granularity in the classification of DB2 workloads into the appropriate Service Class

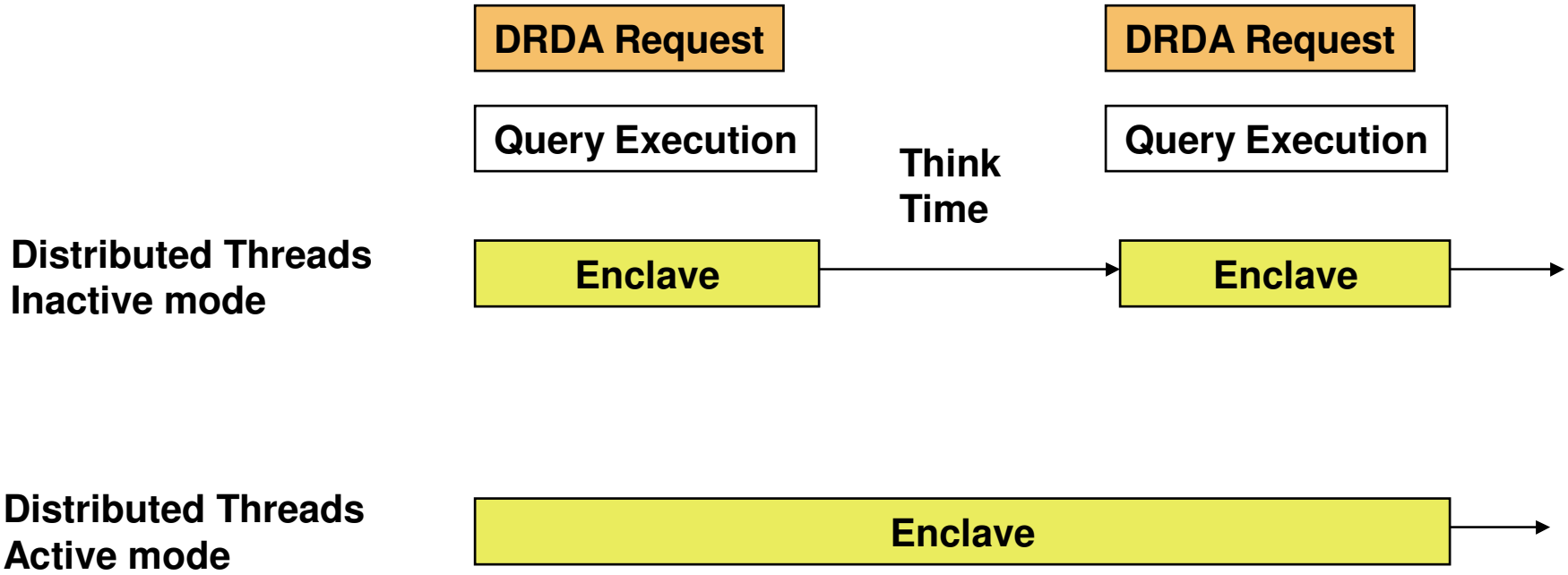
DB2 Workload Priority

How Does WLM Assign Priority To DB2 Workload?

- The priority of the DB2 workload will vary depending upon the origin of the workload
- DB2 workload originating from a local application (IMS, CICS, TSO, Batch)
  - Priority is inherited from the invoking application
  - This applies to Stored Procedures invoked locally
- DB2 Distributed requests (DDF)
  - Priority controlled by DDF Service Class definitions
- DB2 Stored Procedure request via DDF
  - Priority controlled by Service Class definitions

DDF WLM Goal Considerations

- DB2 Distributed thread options control enclave creation and how DB2 may interact with WLM
  - Impacted by KEEPDYNAMIC options, cursor with hold, and zparm settings
  - This impacts whether to use velocity or response time goals

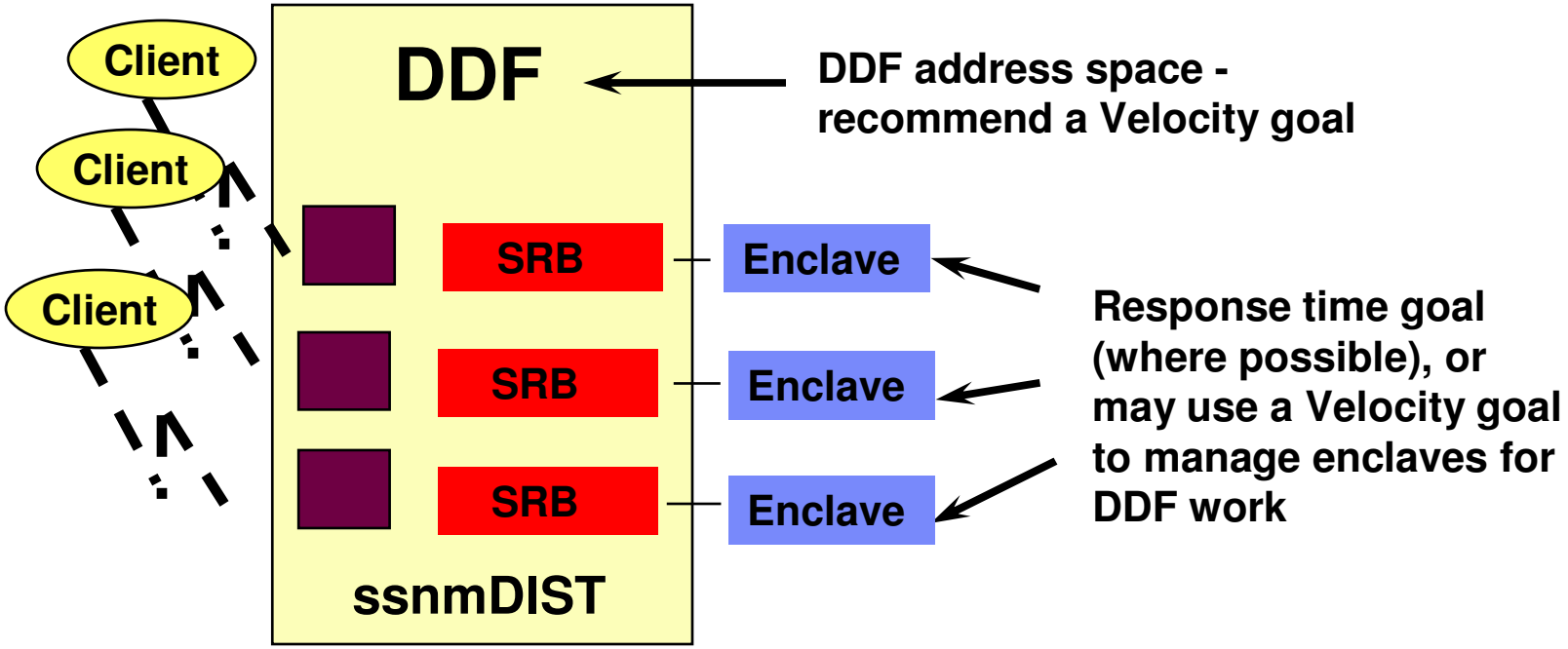


About Enclaves

- Enclaves represent a "business unit of work"
- Enclaves are managed separately from the address space
- Enclaves can include multiple SRBs/TCBs
  - Can span multiple address spaces
  - Can have many enclaves in a single address space
  - Assigned by WLM to a service class
- Enclaves are managed separately from the address space
- DB2 exploits the WLM Enclave interface
  - DB2 workload may create/delete the enclave, join an enclave, etc.

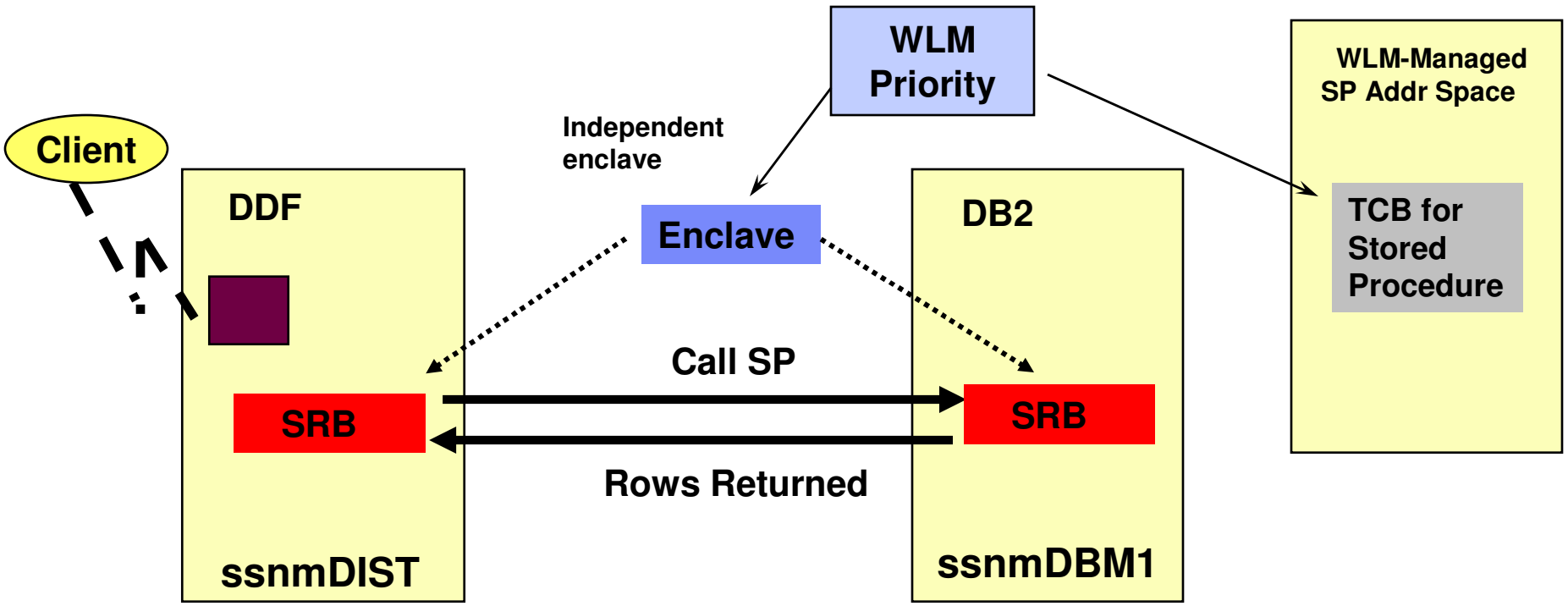
DDF Goal Considerations

- DDF workload may call for a combination of Velocity and Response time goals
  - DDF address space versus DDF workload



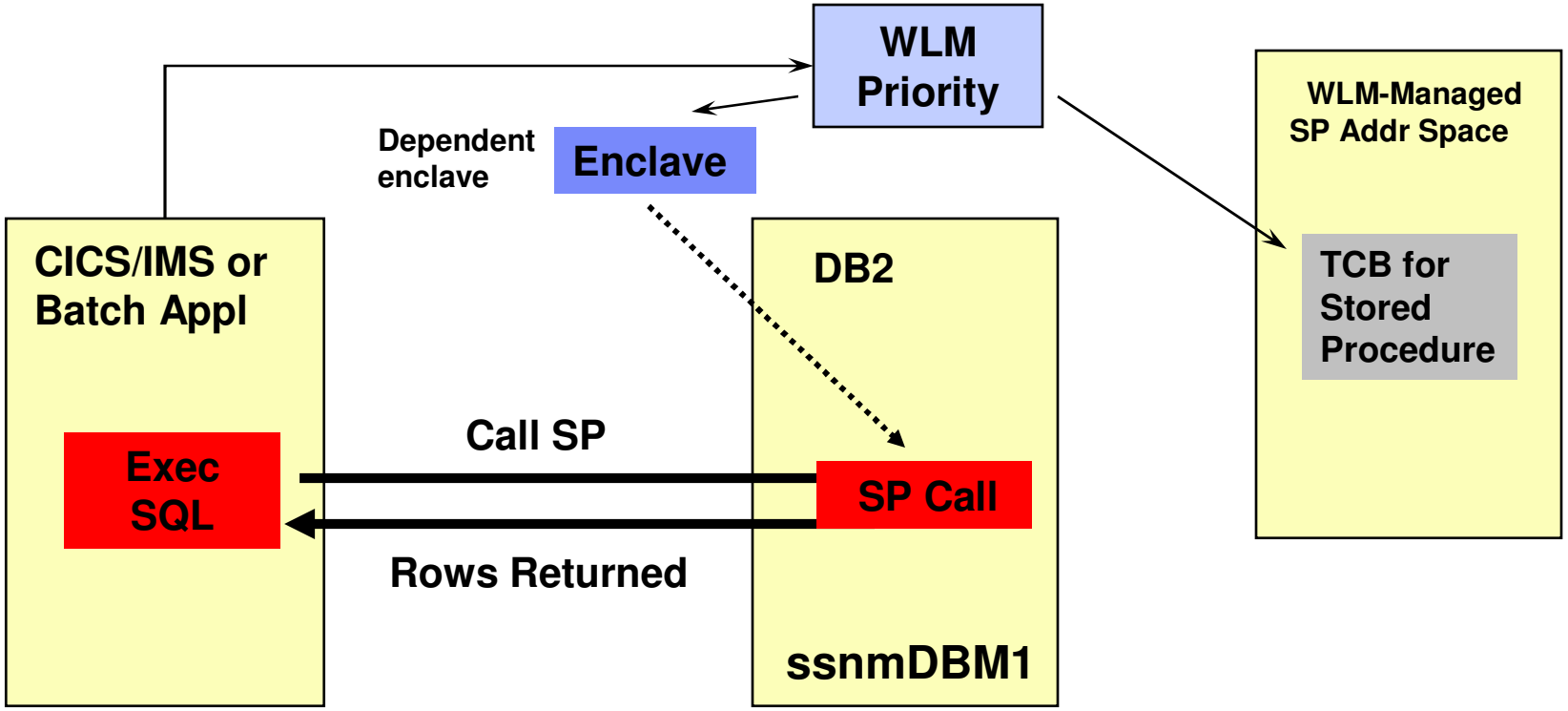
DDF Stored Procedure Priority

- When a Stored Procedure is called from DDF thread
  - DB2 references the enclave created for the DDF request for Stored Procedure
  - Stored Procedure priority is the priority of the DDF request

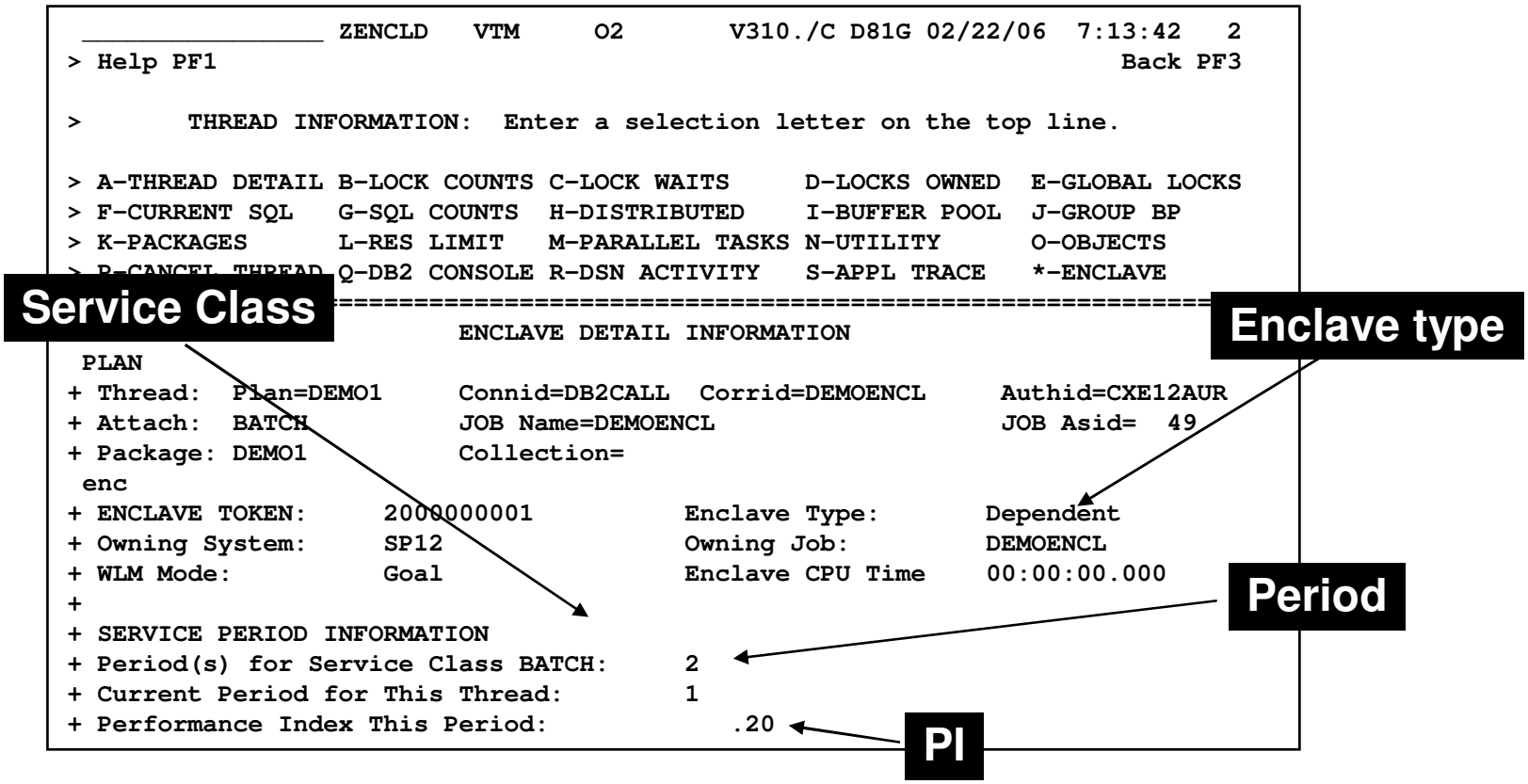


Stored Procedure Priority
Called From A Local Application

- When a Stored Procedure is called from an application on z/OS
  - DB2 creates an enclave for use by the Stored procedure
  - Stored Procedure priority is the priority of the calling application address space



Service Classes And Thread Priorities



- Depending upon how an enclave is created (local allied address space or via DDF) controls what service class, etc that is assigned to a given thread

Intelligent Resource Director - IRD

- Extends the concept of goal-oriented resource management
  - Allows the grouping of system images resident on the same physical server into an "LPAR cluster"
  - Gives WLM the ability to manage resources, both processor and DASD I/O, not just in one single image but across the entire cluster of system images
- LPAR weight management
  - Manages the “weight” of an LPAR and the number of CPs for an LPAR
  - LPAR weight is part of WLM CPU delay analysis
- Dynamic Channel Path management
  - Lets WLM move channel paths from one I/O control unit to another
- Channel Subsystem I/O Priority Queuing
  - Allows WLM to assign a priority to an I/O request
  - Channel subsystem may use a priority managed queue as opposed to FIFO queue
    - Complementary to other I/O queuing mechanisms in the I/O subsystem

How DB2 Interacts With WLM To Assign I/O Priorities

- DB2 informs z/OS about which address space's priority is to be associated with a particular I/O request
  - WLM handles the management of the request

Table 8-1 How read I/O priority is determined

Request type	Synchronous reads	Prefetch reads
Local	Application's address space	Application's address space
DDF or sysplex query parallelism (assistant only)	Enclave priority	Enclave priority

Table 8-2 describes to which enclave or address space DB2 is associated with the I/O write requests.

Table 8-2 How write I/O priority is determined

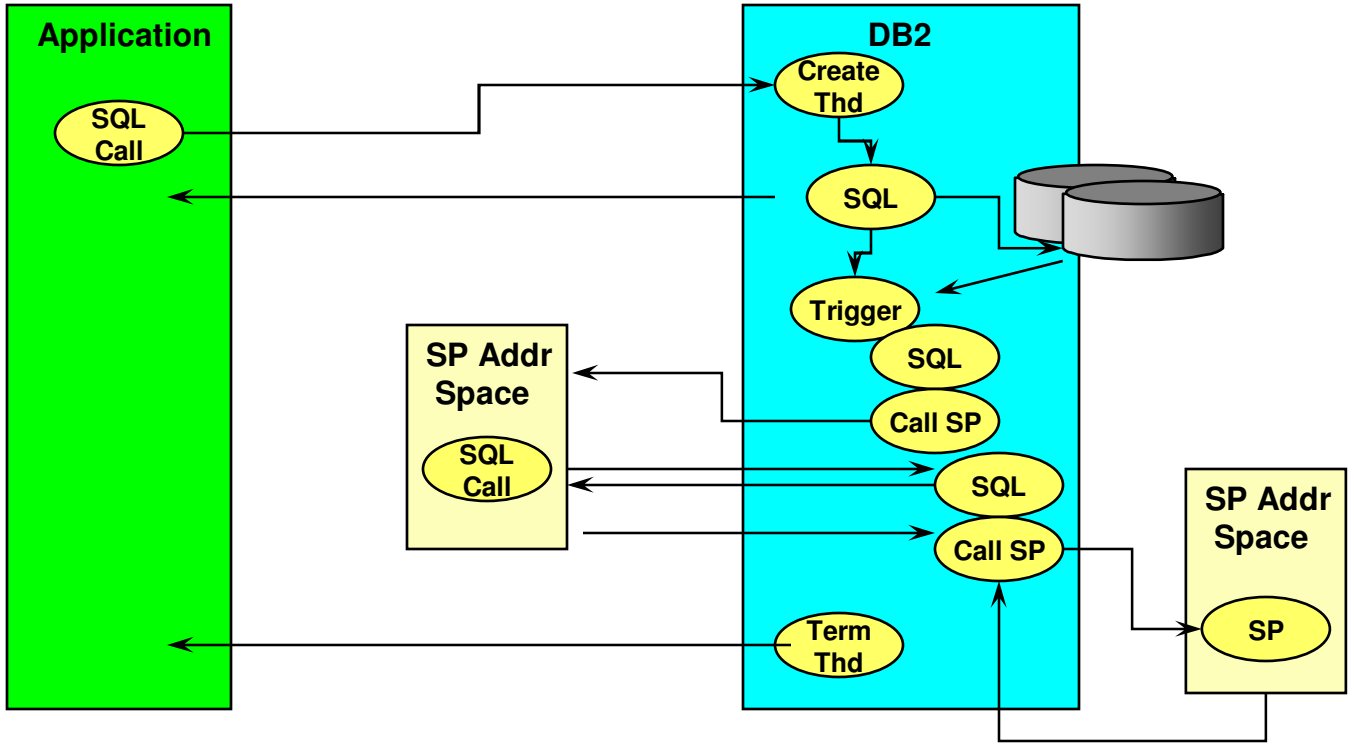
Request type	Synchronous writes	Deferred writes
Local	Application's address space	ssnmDBM1 address space
DDF or sysplex query parallelism (assistant only)	DDF address space	ssnmDBM1 address space

Table from SG24-6472-03

WLM Contention Management

- WLM Contention Management helps addressing chronic or long lasting contention situations
  - WLM provides interfaces to allow resource managers (for example – DB2) to signal contention situations
  - WLM has had the ability to promote (increase the DP) for a short duration to resolve the issue
- DB2 example scenario
  - Lock/latch contention in DB2 may impact performance
  - Often contention may be resolved with a short boost of resource
  - DB2 may notify WLM if a contention occurs
  - WLM may optionally raise the priority for the holder to complete the work
- WLM can promote units of work for longer periods of time, and promote them to the priority of the highest-priority units of work waiting for a resource they are holding.

WLM Considerations For Nested Stored Procedure Requests



- Triggers, Stored Procedures, and UDFs actions may be nested, sometimes multiple layers of nesting
- DB2 tells WLM about dependent stored procedure requests
  - WLM may give dependent requests priority, if needed
  - WLM may start server regions more aggressively, if needed

DB2 And zIIP Processors

- Work on z/OS may have all or a portion of its resource usage on an enclave Service Request Block
  - Enclave SRB work may be directed to the zIIP
- Certain types of DB2 work may take advantage of zIIP
  - DRDA - Queries that access DB2 for z/OS V8/V9 via DRDA over a TCP/IP connection are dispatched within z/OS in enclave SRBs. z/OS directs a portion of this work to the zIIP
  - Complex parallel queries
  - DB2 utilities for index maintenance
    - LOAD, REORG, and REBUILD
- WLM and new enclave structures to manage zIIP related workload – work dependent enclave

It's Back
WLM Managed DB2 Virtual Pools - PK75626

- DB2 9 for z/OS support for z/OS WLM buffer pool management capability is now available for use
- Capability is activated for a buffer pool when a buffer pool is defined or altered by the ALTER BUFFERPOOL command with the AUTOSIZE option set to YES.
  - Each time DB2 processes an ALTER BUFFERPOOL command against a buffer pool that has the AUTOSIZE(YES) attribute, DB2 will register the buffer pool with WLM.
  - DB2 will calculate a maximum size and a minimum size for the buffer pool
    - Maximum size will be calculated to be 1.25 times the initial size
    - Minimum size will be calculated to be 0.75 times the initial size.
- As a DB2 thread executes, DB2 will report to WLM delays occurred due to a DB2 buffer pool I/O wait.
- When WLM has to decide on a policy adjustment
  - If any relevant service classes have not met their goals, WLM analyzes the delays against the service class
  - If a large portion of the delays are due to buffer pool I/O waits for a particular buffer pool, WLM may trigger an alter to the size of the buffer pool.
    - When this occurs, a DB2 message, DSNB555I, will be issued
  - When DB2 deletes a buffer pool because it is no longer needed, it will deregister the buffer pool from WLM.
- A buffer pool could be decreased in size if WLM observes that available real storage is being severely over-committed.

DB2 DDF Considerations Things To Note

- For DDF inactive threads
  - Consider a two-period service class with a response time goal where 80-90% of the transactions complete in first period
- For DDF active threads
  - Consider velocity goals and use a single-period service class
- Look for overly simplistic Service Class definitions
  - Example – type DDF and nothing more than DB2 subsystem name
    - Does little to exploit the ability of WLM to prioritize DB2 workloads
    - Some workloads will inherently be more important than others
- Look for Distributed workloads that run longer than expected but use little resource
  - Indicative of workload that drops into less than optimal Service Classes – example SYSOTHER

DB2 Velocity Goals
Setting Optimal Goals

- Use Velocity goals for always running and long running work
- Use a Velocity goal for the DB2 DDF address space
  - DDF address space has internal tasks that govern thread creation that should have high performance goal
- Use a Velocity goal for
  - DB2 address spaces (SSAS and DBM1)
  - CICS and IMS regions (if not using response time goals)
- Velocity goals at first glance seem easy to set
  - Require more ongoing review
  - Should be validated as the operating environment changes – changes to operating system, hardware, and workload

DB2 Response Time Goals
Setting Optimal Goals

- Use Response Time goals when possible
  - Less need for ongoing maintenance and review
  - WLM will manage resources dynamically to achieve goals
- Response Time goals work well for certain types of DB2 workloads
  - DB2 Distributed workloads in e-business and WebSphere transactional type workloads
  - Transactional type workloads in general including distributed workloads that invoke Stored Procedures
  - Repetitive workloads that have multiple events for WLM to measure and manage

WLM
General Recommendations

- KISS method (Keep It Simple Stupid)
  - Avoid overly complex implementations
  - Avoid overly simplistic standards
    - Example – a service class and/or application environment for each application
  - Too many Application Environments and Service Classes can result in an over abundance of server address spaces
    - Increases number of queues that WLM must manage
- Avoid too many service classes
  - WLM analyzes service classes in a round-robin manner
  - Too many and WLM is unable to manage them all effectively

What's Improved In z/OS V1.12

- z/OS Management Facility (z/OSMF) V1.12 (5655-S28) provides an improved GUI management interface for z/OS
- A new system management task, Workload Management (policy editor), can simplify the creation, modification, and review of z/OS WLM service definitions
- A new application, Sysplex Status and Monitoring Desktops tasks, can provide real-time status of resources of all your servers, sysplexes, as well as Linux images from one location
- The Configuration Assistant for the z/OS Communications Server (available since z/OSMF V1.11) is updated with support for IP security
- The Incident Log capability (available since z/OSMF V1.11)

Summary

Workload Manager As The Priority Manager Of DB2

- Workload Manager (WLM) is the priority and resource manager for z/OS and DB2
- WLM manages critical DB2 address spaces
- These resources have an impact on how DB2 applications perform
- Workload Manager is used to manage DB2 workflow
- Important to have a basic understanding of WLM
- WLM is constantly being enhanced to provide new features and functions

Check Out My Blog

http://tivoliwithaz.blogspot.com