IBM DB2 for z/OS and Workload Manager Intersection: Understanding the Basics

March 8, 2012

Mark Rader, IBM
mrader@us.ibm.com
Agenda

• Workload Manager (WLM) Overview
• How WLM affects DB2
  – Defining DB2 Address Spaces to WLM
  – WLM: Four Types of DB2 work
    • Local Attach
    • DDF and Enclaves
      – What is an enclave?
      – Classifying DDF work
    • Sysplex Query Parallelism
    • Stored Procedures and Application Environments
      – External stored procedures in WLM managed address spaces
      – DB2 9 for z/OS and native SQL procedures
  – Other WLM interaction with DB2
    • Autonomic DB2 buffer pool sizing
    • Latch contention
WLM is the priority and resource manager for z/OS and, therefore, for DB2 on z/OS
DB2 and Workload Manager

• WLM manages DB2 address spaces
  – DB2 subsystem address spaces: MSTR, DBM1, IRLM, DIST
  – DB2 stored procedure address spaces for external stored procedures
  – How WLM manages these address spaces can affect DB2 application performance

• WLM manages DB2 workflow
  – Priority and performance of allied tasks that call DB2
    • CICS, IMS, batch, TSO, WebSphere, MQSeries
  – DB2 distributed and stored procedure workload

• DB2 professional should have a basic understanding of WLM
WLM Terminology

- A Service Definition
  - Consists of one or more Service Policies
- A Service Policy
  - Contains several Workloads
  - One Service Policy is active at a time in an LPAR or Parallel Sysplex
- Each Workload
  - Consists of one or more Service Classes
- Each Service Class
  - Has at least one Period and each Period has one Goal
- A Goal may be one of five types:
  - System, Average Response Time, % Response Time, Execution Velocity, Discretionary
  - A Goal may have a Duration
- Address spaces and transactions are assigned to service classes by Classification Rules
WLM Concepts – Service Class and Classification

• Classification
  – Assignment of incoming work to a service class, and optional report class
  – Based on a wide variety of filters, or qualifiers

• Service Class
  – Set or group of related work
    • Production CICS, IMS, and DB2 address spaces might be in same service class: STCHI or PRODHI
    • Separate Report Classes can report on CICS, IMS, DB2
  – A service class can combine goals of different types in multiple periods
    • A Period is the combination of Importance (IMP), Goal and Duration
WLM Classification Rules

WLM assigns work to a service class based on qualifiers that apply to the subsystem from which the work arrived.

- Subsystems
  - CICS
  - IMS
  - TSO
  - JES
  - DB2
  - STC
  - CB
  - ASCH
  - IWEB
  - OMVS
  - LSFM
  - DDF
  - MQ

- Arriving Work
- Qualifiers
  - accounting info
  - collection name
  - connection type
  - correlation info
  - LU name
  - netid
  - package name
  - old PGN
  - plan name
  - priority
  - procedure name
  - subsystem instance
  - subsystem parameter
  - transaction class
  - transaction name
  - userid

- WLM
- Service goals
- Report Class
  - CPU
  - Storage
  - Tasks
  - I/O
- Service Class
Subsystems Types Used for Classification

- Subsystems follow one of three transaction type models
- Need to understand how this affects the value of figures shown in workload activity report
  * SYSH is used for LPAR load balancing

<table>
<thead>
<tr>
<th>Transaction Type</th>
<th>Allowable Goal Types</th>
<th>Allowable # of Periods</th>
</tr>
</thead>
<tbody>
<tr>
<td>Address space oriented</td>
<td>Response Time, Execution Velocity, Discretionary</td>
<td>Multiple</td>
</tr>
<tr>
<td>Enclave</td>
<td>Response Time, Execution Velocity, Discretionary</td>
<td>Multiple</td>
</tr>
<tr>
<td>CICS/IMS</td>
<td>Response Time</td>
<td>1</td>
</tr>
</tbody>
</table>

Subsystems follow one of three transaction type models:
- Address space oriented
- Enclave
- CICS/IMS

SYSH is used for LPAR load balancing.
WLM Concepts – Importance

• For most work, importance 1 (IMP 1) is highest and importance 5 (IMP 5) is lowest.
• WLM applies resources to IMP 1 first.
• If IMP 1 work meets its goals, then WLM will apply resources to IMP 2 work, then IMP 3, etc.
• Some service trickles down to DISCRETIONARY
• SYSTEM and SYSSTC are internal service classes for system tasks and have the highest dispatching priorities
• SYSOTHER is the default service class for unclassified work and runs at a DISCRETIONARY goal
WLM Concepts – Goal Types

• System
  – SYSTEM and SYSSTC service classes have fixed dispatching priorities

• Execution Velocity, ‘velocity goal’
  – Velocity goals are intended for work for which response time goals are not appropriate, such as address spaces or long running jobs
  – How fast work should run relative to other work requests when ready, without being delayed for CPU, storage, or I/O
  – Expressed as a number, e.g. 60 or 40
    • Value of 60 means ‘ready’ work will run 60% of the time

• Average response time, including queue time and execution time

• Percentile response time, reduces impact of outliers
  – E.g. 90% of transactions complete within 0.7 seconds

• Discretionary – appropriate for low priority, long-running work
WLM Concepts and DB2

• Importance
  – Production DB2 address spaces (MSTR, DBM1, DIST, WLM) should be defined with Importance 1 (IMP 1)
  – Non-production DB2 address spaces in a production LPAR should be defined with lower importance: IMP > 1.
    • Consider relative to other production work
  – Production DDF transactions should generally be defined with IMP below that of production DB2 address spaces
  – IRLMs should be defined in SYSSTC

• Goals for DB2 work
  – **System** - IRLM in SYSSTC
  – **Velocity** goals are appropriate for started tasks or long-running work
    • DB2 address spaces should have velocity goals and only a single period in the service class (MSTR, DBM1, DIST, WLMx)
  – Response time goals are appropriate for transactions, including most DDF work
    • **Percentile response time** – e.g. 90% complete in 0.5 seconds
    • **Average response time** – e.g. average response time is 0.5 seconds
  – **Discretionary**: below IMP 5. Not appropriate for DB2 work
WLM Importance Levels and DB2, an example

- Importance 1 is highest priority after SYSSTC
- DB2 address spaces should have velocity goals and a single period defined
- Non-production DB2s could be IMP 2 or IMP 3 or IMP 4 if in same LPAR (or Parallel Sysplex) with production DB2
- Discretionary work gets service after all other importance levels
  - Not appropriate for DB2 address spaces
  - Not recommended for DB2 work
  - Very little service if CPU 100% busy

<table>
<thead>
<tr>
<th>Importance Level</th>
<th>Service Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>SYSTEM</td>
<td>z/OS</td>
</tr>
<tr>
<td>SYSSTC</td>
<td>IRLMs</td>
</tr>
<tr>
<td>IMP 1 Highest</td>
<td>DB2PMSTR, DB2PDBM1, DB2PDIST, DB2PWLMx</td>
</tr>
<tr>
<td>IMP 2 High</td>
<td>Production DDF txns</td>
</tr>
<tr>
<td>IMP 3 Medium</td>
<td>Low priority work</td>
</tr>
<tr>
<td>IMP 4 Low</td>
<td>Lowest priority work</td>
</tr>
<tr>
<td>IMP 5 Lowest</td>
<td>Lowest priority work</td>
</tr>
<tr>
<td>DISCRETIONARY</td>
<td></td>
</tr>
<tr>
<td>SYSOTHER</td>
<td>Default service class</td>
</tr>
</tbody>
</table>
Service Class: Assigning Types of Goals - example only

**CICS, IMS or TSO transactions**

E.g. average response time goal
Transactions complete < 0.7 seconds

**Production DDF Transactions**

Percentile response time goal, single period

**DB2 Address Spaces**

Velocity goal
Exec Vel = 50
Single period

**Non-production DDF**: response time goals in first period, response time or velocity in second period
Period 1: 90% complete < 0.5 seconds
Period 2: 90% complete < 4 seconds
Period 3: Vel = 40
### Service Class: Period Switch – example

<table>
<thead>
<tr>
<th>PERIOD 1</th>
<th>PERIOD 2</th>
<th>PERIOD 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>DUR = 300</td>
<td>DUR = 600</td>
<td>IMP = 5</td>
</tr>
<tr>
<td>IMP = 3</td>
<td>IMP = 4</td>
<td>VEL = 40</td>
</tr>
<tr>
<td>R/T = 90% in 0.5 sec</td>
<td>R/T = 90% in 4 sec</td>
<td></td>
</tr>
</tbody>
</table>

- All transactions start in Period 1
  - WLM manages the transactions in period 1 to the percentile response time goal of 90% completing in half a second, with an importance of 3.
- Transactions that accumulate 300 service units (DUR = 300) before completing migrate to Period 2 (a new service class period)
  - WLM manages the transactions in period 2 to the goal of 90% completing in 4 seconds, with an importance of 4. [That is, 90% of those that did not complete in period 1.]
- Transactions that accumulate 900 service units (DUR 300 + DUR 600) before completing migrate to Period 3 (a new service class period).
  - WLM manages the transactions in period 3 to a velocity goal of 40, with an importance of 5.

- “Service units” is a hardware independent measure of CPU consumption. If your transaction consumes 1000 service units on a z9, it should consume 1000 service units on a z196.
Service Class Example

• Several goal types defined into periods

<table>
<thead>
<tr>
<th>Action</th>
<th>#</th>
<th>Duration</th>
<th>Imp.</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>300</td>
<td>3</td>
<td>90% complete within 00:00:00.500</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>600</td>
<td>4</td>
<td>90% complete within 00:00:04.000</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>________</td>
<td>5</td>
<td>Execution velocity of 40</td>
</tr>
</tbody>
</table>
WLM Managed Delays

• WLM can only affect work by adjusting these resources:
  – Processor (dispatching priority)
  – Non-paging DASD I/O (IOSQ, subchannel pending, control unit queue)
  – Storage (paging, swapping)
  – Tasks (multi-programming level, server address space creation, batch initiation)

• WLM cannot manage, for example:
  – User delay (coffee breaks)
  – Network delay
WLM Concepts: Performance Index (PI)

- 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

<table>
<thead>
<tr>
<th>PI Condition</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>The PI equals 1</td>
<td>The work in the period is meeting its goal exactly</td>
</tr>
<tr>
<td>The PI is less than 1</td>
<td>The work is doing better than its goal</td>
</tr>
<tr>
<td>The PI is more than 1</td>
<td>The work is missing its goal</td>
</tr>
</tbody>
</table>
2. Compare reality with goal (as stated in WLM policy). Performance Index (PI) > 1 gets attention.

3. Change priorities, based on delays (donor/receiver).

4. Wait 10 seconds for the effect.

1. Measure delays (again, again & again...)

- Set accurate goals
  - Goals should correspond to how your business runs
  - Goals should be realistic
  - Loose goals can cause poor performance because WLM sees the goals are met, so takes no action

- Goals should correspond to how your business runs
WLM Service Class Periods

• WLM heuristic behavior is applied to service class periods
• WLM can effectively manage 25-30 active service class periods
  – If you have more than 30 active service class periods, WLM may not be able to adjust resources for all of them when the system is busy
  – It is when the system is busy that you want WLM to adjust resources to meet your business goals

• “Loose” goals are performance goals that are too easily achieved
  – Service class periods with loose goals are likely to have a PI < 1, so WLM will always perceive they are meeting their goals.
  – Service class periods with loose goals may have a PI < 0.7, in which case they may become a donor
Defining DB2 Address Spaces to WLM

• DB2 address spaces are started tasks
  – To WLM, the DB2 address spaces have a subsystem type of “STC”
• IRLMs should be defined in service class SYSSTC
• Remaining DB2 address spaces should be assigned to a service class with a single period, a velocity goal and appropriate importance. For example,
  – Production: IMP 1
  – QA, Development and/or Test in same LPAR/Sysplex:
    • IMP > 1 (i.e. lower importance)
    • Adjust based on other production work, such as production batch
  – DB2 address spaces include ssnmMSTR, ssnmDBM1, ssnmDIST and ssnmWLMx for stored procedures
WLM: Four Types of DB2 Work

• 1: DB2 work that originates in another local subsystem:
  – Examples: CICS, IMS, TSO

• 2: DDF work requests
  – DDF requests use enclave SRBs

• 3: Sysplex Query Parallelism
  – Queries that DB2 creates by splitting a single, larger query and distributing it to other members of the data sharing group in a Parallel Sysplex® (PSX)

• 4: Stored Procedures
  – WLM managed stored procedures, which run in WLM Application Environments, are external stored procedures or DB2 V8 SQL Procedures
  – Native SQL Procedures in DB2 9 for z/OS
Type 1 - Local Attach

- DB2 SQL activity runs under dispatchable unit of invoker
  - IMS, CICS, TSO, Batch, etc.
  - Inherited classification class of invoker
  - Priority and management of home unit
  - Service attributed back to invoker
Type 2 - DDF and Enclave SRBs

ssnmDIST (DDF)

- Enclave SRB
- PC-call to DBM1

- Create Enclave
- Schedule SRB

STCHI

Vel = 50
Imp = 1

SMF 30: Common Address Space Work accounting
SMF 72: RMF Workload Activity and Storage Data

DDF production requests

- DDF default requests

DDF rules

- DDFPROD
  - RT = 90%, 0.5 sec, IMP 2

- DDFDEF
  - RT = 5s avg
  - Imp = 3

PC-call to DBM1

STC rules
What is an Enclave?
What is an Enclave?

The Enclave at Sheridan Pointe
What is an Enclave?

2010 Buick Enclave CX
What is an Enclave?

• A "business transaction" without address space boundaries
  – Two types: dependent and independent
  – System or sysplex scope
• Dependent enclaves
  – Logical extension of an existing address space transaction
  – Inherits service class from its owner's address space
• Independent enclave (e.g. DDF)
  – True SRM transaction
  – Separately classified and managed in service class
• Why do we need enclaves?
**Enclave Characteristics**

- Created by an address space (AS)
  - the "owner"; DDF for DBATs
- One AS can own many enclaves
- One enclave can include multiple dispatchable units (SRBs/tasks) executing concurrently in multiple address spaces (the "participants")
  - Enclave SRBs are preemptible, like tasks
  - All its dispatchable units are managed as a group
- Many enclaves can have dispatchable units running in one participant address space concurrently
- RMF produces separate Type72 SMF records for independent enclaves
  - Both Type72 and Type30 records produced for address spaces

SMF 30: Common Address Space Work accounting
SMF 72: RMF Workload Activity and Storage Data
Classifying DDF Work

• Define service classes and appropriate goals for DDF work
• DDF Classification Defaults
  – Defaults apply if you do not provide any classification rules for DDF work
  – Enclaves default to the SYSOTHER service class (i.e. discretionary goal) unless they can be assigned to a service class
• Managing DDF Work (Enclaves)
  – All goals are permitted
  – Transactions are subject to period switch
  – WLM manages an enclave with its own dispatching priority, etc.
  – Production DDF work:
    • Generally importance (IMP) below that of DB2 address spaces
    • Consider a single period goal
Enclaves Can Use Multiple Periods

- **PERIOD 1**
  - 90% in 0.5 sec; IMP = 3
  - DUR = 300
- **PERIOD 2**
  - 90% in 4 sec; IMP = 4
  - DUR = 600
- **PERIOD 3**
  - Ex Vol = 40
  - IMP = 5

**Response time** and **Velocity** measures

- Dispatch Priority
- Working Set
- I/O Qing
- Multi Pgm Level

- The **DUR**ation value defines period length in service units
- Within a service class, periods can use different goals, goal types, and importance
- In this example: Service class DDFTEST
What is a DDF Transaction?

Threads: ZPARM CMTSTAT = Inactive

<table>
<thead>
<tr>
<th>DRDA unit-of-work 1</th>
<th>DRDA unit-of-work 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Queue time</td>
<td>Queue time</td>
</tr>
<tr>
<td>Execution time</td>
<td>Execution time</td>
</tr>
<tr>
<td>Idle</td>
<td>Idle</td>
</tr>
</tbody>
</table>

Threads: ZPARM CMTSTAT = Active

Database thread is active from creation until termination

Enclave transaction managed by SRM exhibits conversational behavior
What Goals Should I Use?

• **CMTSTAT=INACTIVE** and **DBAT** is pooled (Connection inactive)
  – DDF creates one enclave per active interval
  – Response times do not include user think time
  – Response time goals and multiple periods can be used
  • But multiple periods with different importance can lead to issues with locks and latches

• **CMTSTAT=ACTIVE**
  – DDF creates one enclave for the life of the thread
  – Enclave response time includes user think time
  – Response time goals should not be used
  – Multiple periods should not be used
CMTSTAT = INACTIVE

• What if the DBAT cannot be pooled at COMMIT?
  – If only reason is KEEPDYNAMIC = YES
    • Same as previous slide: one enclave per interval, response times do not include user think time, response time goals are appropriate
  – If because CURSOR WITH HOLD, DGTT or LOB LOCATOR:
    • Thread stays active after COMMIT
      – Subject to period switching
      – Subject to idle thread timeout (IDTHTION in DSNZPARM)
    • Percentile response time goals may be appropriate
• What about DB2 10 for z/OS High Performance DBATs?
  – DBAT remains active after commit, but enclave deleted, accounting record cut, and idle thread timer reset
  – One enclave per interval, response time goals are appropriate
DDF Classification Rules, example

- Classification by Subsystem Instance (SI), Process Name (PC - application program), Accounting Information (AI), and Userid (UI)

<table>
<thead>
<tr>
<th>Action</th>
<th>Type</th>
<th>Name</th>
<th>Start</th>
<th>DEFAULTS:</th>
<th>Service</th>
<th>Report</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SI</td>
<td>DB9A</td>
<td></td>
<td>DDFDEF</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>PC</td>
<td>TRX*</td>
<td></td>
<td>DDFDEF</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>AI</td>
<td>T00A*</td>
<td>56</td>
<td>DDFPROD</td>
<td>RSSL</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>UI</td>
<td>PAOLR3</td>
<td></td>
<td>DDFTOT</td>
<td>RNISANTI</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>SI</td>
<td>D9C*</td>
<td></td>
<td>DDFDEF</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>UI</td>
<td>PAOLR3</td>
<td></td>
<td>DDFTEST</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>PC</td>
<td>DB2J*</td>
<td></td>
<td>DDFPROD</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Note: DDF Classification Rules example shows how to classify data based on different parameters such as Subsystem Instance, Process Name, Accounting Information, and Userid. The table illustrates rules that can be applied to classify data accordingly.*
## DDF Work Classification Attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accounting Information</td>
<td>AI</td>
<td>Can be passed from a DB2 Client via Client Information APIs</td>
</tr>
<tr>
<td>Correlation Information</td>
<td>CI</td>
<td>DB2 Connect assigns application program name by default but application can set via Client Information APIs</td>
</tr>
<tr>
<td>Collection Name</td>
<td>CN</td>
<td>Collection name of the first SQL package accessed by the DRDA requester in the unit of work</td>
</tr>
<tr>
<td>Connection Type</td>
<td>CT</td>
<td>Always 'DIST' for DDF server threads</td>
</tr>
<tr>
<td>Package Name</td>
<td>PK</td>
<td>Name of the first DB2 package accessed by the DRDA requester in the unit of work</td>
</tr>
<tr>
<td>Plan Name</td>
<td>PN</td>
<td>'DISTSERV' for DDF server threads accessed via DRDA requesters unless requester is another DB2 for z/OS, then requester's PLAN name</td>
</tr>
<tr>
<td>Procedure Name</td>
<td>PR</td>
<td>Name of the procedure called as the first request in the unit of work</td>
</tr>
<tr>
<td>Process Name</td>
<td>PC</td>
<td>Client application name by default but can be set via Client Information APIs</td>
</tr>
<tr>
<td>Subsystem Collection Name</td>
<td>SSC</td>
<td>Usually the DB2 data sharing group name</td>
</tr>
<tr>
<td>Subsystem Instance</td>
<td>SI</td>
<td>DB2 server's MVS subsystem name</td>
</tr>
<tr>
<td>Sysplex Name</td>
<td>PX</td>
<td>Name assigned to sysplex at IPL</td>
</tr>
<tr>
<td>Userid</td>
<td>UI</td>
<td>DDF server thread's primary AUTHID</td>
</tr>
<tr>
<td>Subsystem Parameter</td>
<td>SPM</td>
<td>Beginning in V8: the concatenation of client userid and workstation</td>
</tr>
</tbody>
</table>

- Not very useful
- Widely used, some limitations
- Most granular control; See next page.
Workload Classification Attributes - Client

• ODBC/CLI/VB/ADO ... applications
  – Use SQLSetConnectionAttr on:
    • SQL_ATTR_INFO_ACCTSTR - accounting string (AI)
    • SQL_ATTR_INFO_APPLNAME - application name (PC)
    • SQL_ATTR_INFO_USERID - client userid
    • SQL_ATTR_INFO_WRKSTNNAME - client workstation name

• Non-ODBC… use sqleseti Administrative API function

• IBM Data Server Driver for JDBC and SQLJ (Type 2 or Type 4 connectivity)
  – Use methods against connection class instance
    • setClientUser, setClientApplicationInformation, setClientWorkStation, setClientAccountingInformation
Type 3 – Sysplex Query Parallelism; WLM sees as “DB2”

Complex query originates here

Portions of complex query arrive on participant systems, classified under "DB2" rules, and run in enclave SRBs

Sysplex Query Parallelism: DB2 Data Sharing

Host 1

Host 2

Host 3

PARTITIONED TABLESPACE
**Type 4 - DB2 External Stored Procedures**

- **Task**
  - Listens for requests coming from outside of the system
  - Creates independent enclave
  - Schedules enclave SRB

- **DB2DIST**
- **DB2DBM1**
- **DB2WLM**
- **CHARLIE (TSO)**

- **Enclave A**
- **Enclave B**
- **Enclave C**

- **Task**
  - Creates dependent enclave
  - Continuation of transaction CHARLIE

- **KEY:**
  - TCB
  - Enclave SRB

- **Stated Procedures execute as TCBs in WLM-managed address space(s)**
  - TCBs not zIIP-eligible
Native SQL Procedures (beginning with DB2 9 for z/OS)

The SQL procedure logic runs in the DBM1 address space
Enclave SRB mode; DB2WLMx not involved, no TCB schedule delay

Execution from remote thread eligible for zIIP at same percentage as DDF Enclave SRB
WLM Considerations – Example: Stored Procedures

• The original assumption
  – All work requests inserted by DB2 (example – Stored Procedures) were independent requests

• The reality
  – Procedures may recursively call other procedures
  – The processing may be inter-dependent

• The newer logic
  – DB2 tells WLM about dependent stored procedure requests
  – WLM gives dependent requests priority
    • WLM may make adjustments, if needed
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 SRB
  - Enclave SRB work may be directed to the zIIP
- Certain types of DB2 work may take advantage of zIIP, including
  - DRDA - Queries that access DB2 for z/OS via DRDA over TCP/IP
  - Complex parallel queries
  - DB2 utilities for index maintenance
    - LOAD, REORG, and REBUILD
  - DB2 V10 – Sequential prefetch eligible for zIIP processor
- WLM and new enclave structures to manage zIIP related workload – work dependent enclave
About Work-dependent Enclaves

• Extension to an independent, dependent, or other work-dependent enclave
  – Extends the transaction creating the enclave.
• Allows control of zIIP offload by entitled products.
DB2 V9 for z/OS: WLM Automatic Buffer Pool Size Adjustment

• PK75626 enables capability as well as the WLM delay monitoring support
• Requires z/OS 1.9 and above with WLM APARs OA18461 and OA32631 applied
• Triggered when buffer pool is defined or altered with AUTOSIZE(YES)
  – VPSIZE at the time of AUTOSIZE setting governs the possible size range for buffer pool
    • e.g. VPSIZE(10000) would allow WLM to adjust its size from 7500 to 12500
    • WLM will only request the alteration if new size within range (minimum size adjustment is 64)
    • WLM can decrease size when real storage demand affected
• Buffer pools adjusted based on WLM goal attainment of service classes that buffer pool size affects, e.g. lots of random I/O.
  – This is a WLM policy adjustment decision
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.
Resources

• Redbooks
  – *DB2 9 for z/OS: Distributed Functions*
    • SG24-6952-01
  – *System Programmer’s Guide to: Workload Manager*
    • SG24-6472-03
Questions

???
Acknowledgements

• Special thanks to:
  – Hugh Smith, IBM DB2 Development
  – Glenn Anderson, IBM Learning Services
  – Ed Woods, IBM Tivoli
  – Bill Schray, IBM ATS
  – Brad Snyder, IBM ATS
  – Kathy Walsh, IBM ATS
Acknowledgements and Disclaimers:

**Availability.** References in this presentation to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates.

The workshops, sessions and materials have been prepared by IBM or the session speakers and reflect their own views. They are provided for informational purposes only, and are neither intended to, nor shall have the effect of being, legal or other guidance or advice to any participant. While efforts were made to verify the completeness and accuracy of the information contained in this presentation, it is provided AS-IS without warranty of any kind, express or implied. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, this presentation or any other materials. Nothing contained in this presentation is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software.

All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics may vary by customer. Nothing contained in these materials is intended to, nor shall have the effect of, stating or implying that any activities undertaken by you will result in any specific sales, revenue growth or other results.

© Copyright IBM Corporation 2012. All rights reserved.

- U.S. Government Users Restricted Rights - Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

IBM, the IBM logo, ibm.com, CICS, DB2, IMS, WebSphere, and Parallel Sysplex are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at “Copyright and trademark information” at www.ibm.com/legal/copytrade.shtml

Other company, product, or service names may be trademarks or service marks of others.
Information Management Communities

- On-line communities, User Groups, Technical Forums, Blogs, Social networks, and more
  - Find the community that interests you…
    - World of DB2 for z/OS http://db2forzos.ning.com/
    - Information Management ibm.com/software/data/community
    - Business Analytics ibm.com/software/analytics/community
    - International DB2 User Group www.idug.org

- IBM Champions
  - Recognizing individuals who have made the most outstanding contributions to Information Management, Business Analytics, and Enterprise Content Management communities
    - ibm.com/champion
Useful DB2 for z/OS URLs

- DB2 for z/OS Website [http://www-01.ibm.com/software/data/db2/zos/]

- Latest Whitepapers
  - Business Value of DB2 10 – Julian Stuhler
  - A Matter of Time: Temporal Data Management
  - Why DB2 for z/OS is BETTER than Oracle RAC?


- Upcoming Conferences/Events
  - IDUG DB2 Tech Conference AG  Denver  - 14th-16th May 2012
  - IDUG 10 Migration Planning Workshop Denver  - 13th May 2012
Top DB2 for z/OS e-Communities

- World of DB2 for z/OS - 1700+ members  [http://db2forzos.ning.com/]
- DB2 10 LinkedIn - 1000+ members  [http://linkd.in/IBMDB210]
- DB2 for z/OS What’s On LinkedIn – 2000+ members  [http://linkd.in/kd05LH]
- DB2 for z/OS YouTube  [http://www.youtube.com/user/IBMDB2forzOS]
- WW IDUG LinkedIn Group - 2000 +members  [http://linkd.in/IDUGLinkedIn]
# DB2 10 is Here!
Customers seeing reduced costs, simplified workloads through proven technology

<table>
<thead>
<tr>
<th>Reduced Costs</th>
<th>Simplified Workloads</th>
<th>Proven Technology</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;Based on the performance metrics from our controlled test environment, we see a significant amount of CPU and Elapsed time savings. This release has many features that will help bring down our operating costs.”</td>
<td>“With DB2 10 able to handle 5-10 times as many threads as the previous version, the upgrade will immediately give the bank some much-needed room for future workload growth while simultaneously reducing their data sharing overhead.”</td>
<td>“Every single SQL statement we have tested has been better or the same as our current optimal paths – we have yet to see any significant access path regression. We had to spend a lot of time tuning SQL with DB2 9, but we expect that to disappear when we upgrade to DB2 10.”</td>
</tr>
<tr>
<td>Morgan Stanley DB2 Team</td>
<td>Paulo Sahadi - Senior Production Manager, Information Management Division at Banco do Brasil</td>
<td>Philipp Nowak, BMW DB2 Product Manager</td>
</tr>
<tr>
<td>&quot;We are particularly interested in the performance improvements due to the potential CPU reductions that we realized during our DB2 10 Beta testing. Our early testing has shown out-of-the-box processing cost reductions of between 5% - 10% and for some workloads as high as 30%. Potential cost savings of this magnitude cannot be ignored given today’s business climate.”</td>
<td>“The new temporal functionality in DB2 10 for z/OS will allow us to drastically simplify our date-related queries. In addition, we’ll be able to reduce our storage costs by using cheaper storage for inactive rows and reduce our processing cost by having DB2 handle data movement more efficiently than the custom code we’ve written to do the same work in the past”</td>
<td>The new audit capabilities in DB2 10 will allow tables to be audited as soon as they are created, which is an obvious benefit for the business and will reduce costs and simplify our processes”</td>
</tr>
<tr>
<td>Large Global Bank</td>
<td>Large Insurance Company</td>
<td>Guenter Schinkel - Postbank Systems AG</td>
</tr>
</tbody>
</table>