Agenda

• WLM Overview
• 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
    • DB2 9 for z/OS and Native SQL Procedures
    • WLM Managed Stored Procedures
• WLM and DB2 Buffer Pools
z/OS Workload Management

Arriving Work

Service Goals

Dynamic Workload Distribution

Service Class

CPU
Storage
Tasks
I/O

Report Class

CICS
IMS
TSO
JES
DB2
STC
CB
ASCH
IWEB
OMVS
LSFM
DDF
MQ
WLM Concepts

• Importance
  — Importance of some work relative to other concurrently executing work
  — For most work, importance 1 (IMP 1) is highest and importance 5 (IMP 5) is lowest.
  — WLM applies resources to IMP 1 work first. If IMP 1 work meets its goals, then WLM will apply resources to IMP 2 work, then IMP 3, etc.

• Goal
  — Performance objective stated in business terms. There are four goal types.

• Service Class
  — Set of performance objectives for a class of work
  — A service class can combine goals of different types in multiple periods.

• Classification rules
  — Combination of the subsystem (e.g. STC, CICS, DDF, TSO) and qualifiers, or filters, which WLM uses to assign work to a service class

• 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 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.
    • For example: IMP 2 for QA, IMP 3 for Development, etc.
  – Production DDF transactions should generally be defined with IMP below that of production DB2 address spaces
  – IRLMs should be defined in SYSSTC

• Goal – four goal types in service class definitions
  – **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
  – 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 Importance 5. Appropriate for low priority, long-running work
WLM Importance Levels and DB2, an example

<table>
<thead>
<tr>
<th>SYSTEM</th>
<th>z/OS</th>
</tr>
</thead>
<tbody>
<tr>
<td>SYSSTC</td>
<td>IRLMs</td>
</tr>
<tr>
<td>IMP 1 Highest</td>
<td>DB2PMSTR, DB2PDBM1, DB2PDIST, DB2PWLM</td>
</tr>
<tr>
<td>IMP 2 High</td>
<td>Production DDF txns - First period</td>
</tr>
<tr>
<td>IMP 3 Medium</td>
<td>Production DDF txns - Second period</td>
</tr>
<tr>
<td>IMP 4 Low</td>
<td>Low 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>

- 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
  - Very little service if CPU 100% busy
Assigning Types of Goals

**CICS, IMS or TSO transactions**
E.g. average response time goal
90% complete < 0.7 seconds

**DB2 Address Spaces**
Velocity goal
Exec Vel = 50

**Production DDF Transactions**
E.g. percentile response time goal, multiple periods
Period 1: 90% complete < 0.5 seconds
Period 2: 90% complete < 4 seconds
Period 3: Vel = 20
Period Switch

- 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 2
- 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 3. [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 20, with an importance of 4.
- “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
Identify your workload(s), categorize them.
**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 STC: DB2 Address Spaces</td>
<td>Multiple</td>
</tr>
<tr>
<td>Enclave</td>
<td>Response Time Execution Velocity Discretionary DDF and DB2</td>
<td>Multiple</td>
</tr>
<tr>
<td>CICS/IMS</td>
<td>Response Time</td>
<td>1</td>
</tr>
</tbody>
</table>

**Diagram Notes:**
- TSO: 1
- CB: 2
- ASCH: 1
- CB: 2
- CICS: 3
- DB2: 2
- DDF: 2
- IMS: 3
- IST: 2
- JES: 2
- IWEB: 2
- TCP: 2
- OMVS: 1
- NETV: 2
- LDAP: 2
- MQ: 2
- STC: 1
- SYSH: 1
- SYSH: *
WLM Classification Rules

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

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

- If all IMP 1 service class periods meet their goals, WLM considers IMP 2
- If all IMP 2 service class periods meet their goals, WLM considers IMP 3
- Then IMP 4, then IMP 5
- One change per 10 second interval... across the system.
- 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
- Goals should be realistic
- Loose goals can cause poor performance because WLM sees the goals are met, so takes no action
Performance Index (PI)

- WLM calculates PI for service class period.
  - If PI = 1, service class period met its goal. If PI < 1, it exceeded its goal.

- RMF Workload Activity, Service Class Period Report

---

GOAL: RESPONSE TIME 000.00.300 FOR 95%

- TSO Period 1 exceeded its objectives; PERF INDX (PI) < 1
### Performance Index (PI)

- WLM calculates PI for service class period.
  - If PI > 1, service class period missed its goal.

- RMF Workload Activity, Service Class Period Report

  ![Workload Activity Report](image)

  - TSO Period 2 did NOT meet its objectives; PERF INDX (PI) > 1
Defining DB2 Address Spaces to WLM

DB2 address spaces belong in workload type of STC - started tasks - and default to a service class of SYSSTC. IRLMs should be in SYSSTC.
The rest of the DB2 address spaces should be defined with a velocity goal and appropriate importance. For example, production in IMP 1, development in IMP 2 or 3.
DB2 Address Spaces and Service Class

- xxxxIRLM: Lock Manager
- xxxxMSTR: System Services
- xxxxDBM1: Database Services
- xxxxDIST: Distributed Services
- xxxxWLMx: WLM Managed Stored Procs

SYSSTC

Similar IMP and Velocity Goal; Single Period. Usually in the same service class
Specifying Services Classes - STC

Subsystem Type Selection List for Rules
Row 1 to 14 of 14
Command ===> ___________

Action Codes: 1=Create, 2=Copy, 3=Modify, 4=Browse, 5=Print, 6=Delete, /=Menu Bar

<table>
<thead>
<tr>
<th>Action</th>
<th>Type</th>
<th>Description</th>
<th>Service</th>
<th>Report</th>
</tr>
</thead>
<tbody>
<tr>
<td>__</td>
<td>ASCH</td>
<td>APPC Transaction Programs</td>
<td>ASCH</td>
<td></td>
</tr>
<tr>
<td>__</td>
<td>CB</td>
<td>WebSphere/Component Broker</td>
<td>DDFDEF</td>
<td>RCB</td>
</tr>
<tr>
<td>__</td>
<td>CICS</td>
<td>CICS Transactions</td>
<td>CICS</td>
<td></td>
</tr>
<tr>
<td>__</td>
<td>DB2</td>
<td>DB2 Sysplex Queries</td>
<td>DB2QUERY</td>
<td></td>
</tr>
<tr>
<td>__</td>
<td>DDF</td>
<td>DDF Work Requests</td>
<td>DDFBAT</td>
<td></td>
</tr>
<tr>
<td>__</td>
<td>EWLM</td>
<td>EWLM Subsystem for ESC/ETC</td>
<td>SC_EWLM</td>
<td>REWLMDFT</td>
</tr>
<tr>
<td>__</td>
<td>IMS</td>
<td>IMS Transactions</td>
<td>IMS</td>
<td></td>
</tr>
<tr>
<td>__</td>
<td>IWEB</td>
<td>Web Work Requests</td>
<td>RIWEB</td>
<td></td>
</tr>
<tr>
<td>__</td>
<td>JES</td>
<td>Batch Jobs</td>
<td>BATCHLOW</td>
<td>BATCHDEF</td>
</tr>
<tr>
<td>__</td>
<td>MQ</td>
<td>MQSeries Workflow</td>
<td>RMQ</td>
<td></td>
</tr>
<tr>
<td>__</td>
<td>OMVS</td>
<td>UNIX System Services</td>
<td>SYSSTC1</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>STC</td>
<td>Started Tasks</td>
<td>STC</td>
<td>RSYSDFLT</td>
</tr>
<tr>
<td>__</td>
<td>SYSH</td>
<td>linux</td>
<td></td>
<td></td>
</tr>
<tr>
<td>__</td>
<td>TSO</td>
<td>TSO Commands</td>
<td>TSO</td>
<td>TSO</td>
</tr>
</tbody>
</table>
Specifying Services Classes - STC

- **DB2 in STCHI**
  - DB2 address spaces defined in a Transaction Name Group (TNG)

- **IRLM in SYSSTC**

<table>
<thead>
<tr>
<th>Action</th>
<th>Type</th>
<th>Name</th>
<th>Start</th>
<th>Service</th>
<th>Report</th>
<th>DEFAULTS:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>TNG</td>
<td>DB2AS</td>
<td></td>
<td>STCHI</td>
<td>RDB2AS</td>
<td>STC</td>
</tr>
<tr>
<td></td>
<td>TN</td>
<td>%%%%IRLM</td>
<td></td>
<td>SYSSTC</td>
<td>RIRLM</td>
<td>RYSDFLT</td>
</tr>
<tr>
<td>______ 1</td>
<td>SPM</td>
<td>SYSTEM</td>
<td></td>
<td>SYSTEM</td>
<td>RSYSTEM</td>
<td></td>
</tr>
<tr>
<td>______ 1</td>
<td>SPM</td>
<td>SYSSTC</td>
<td></td>
<td>SYSSTC</td>
<td>RYSSTC</td>
<td></td>
</tr>
<tr>
<td>______ 1</td>
<td>TN</td>
<td>DFS</td>
<td></td>
<td>SYSSTC2</td>
<td>RINETD1</td>
<td></td>
</tr>
<tr>
<td>______ 1</td>
<td>TN</td>
<td>DFSKERN</td>
<td></td>
<td>SYSSTC3</td>
<td>RINETD1</td>
<td></td>
</tr>
<tr>
<td>______ 1</td>
<td>TN</td>
<td>INETD1</td>
<td></td>
<td>SYSSTC4</td>
<td>RINETD1</td>
<td></td>
</tr>
<tr>
<td>______ 1</td>
<td>TN</td>
<td>PMAP</td>
<td></td>
<td>SYSSTC5</td>
<td>RPMAP</td>
<td></td>
</tr>
</tbody>
</table>
WLM: Four Types of DB2 Work

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

DDF work requests

Sysplex Query Parallelism: Queries that DB2 creates by splitting a single, larger
query and distributing it to remote systems in a Parallel Sysplex®

Stored Procedures – specifically WLM managed stored procedures, which run in
WLM Application Environments.
#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
#2 - DDF and Enclave SRBs

**ssnmDIST (DDF)**

- **Enclave SRB**
- **Create Enclave**
- **Schedule SRB**

- **DDF production requests**
- **DDF default requests**

- **STC rules**

- **DDF rules**
  - **RT = 90%, 0.5 sec, IMP 2**

- **DDFPROD**
  - **RT = 5s avg**
  - **Imp = 3**

- **STCHI**
  - **Vel = 50%**
  - **Imp = 1**

- **SMF 30: Common Address Space Work accounting**
- **SMF 72: RMF Workload Activity and Storage Data**
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
  – True SRM transaction
  – Separately classified and managed in service class

• Why do we need enclaves?
Why Do We Need Enclaves?

• DB2 V3
  – All distributed threads ran at priority of \textit{ssnmDIST} address space
    • Often \textit{ssnmDIST} ran at lower priority to protect other production work
  – High priority business requests competed for resources equally with low priority requests
  – Limited local control of SQL from distributed clients
  – Example: Vice President versus Summer Intern
• DB2 V4 introduced enclave support
  – Distributed threads (DBATs) managed separately from \textit{ssnmDIST} address space
  – DBAT joins an enclave and runs at priority of enclave
    • High priority work assigned to service class with aggressive performance objectives
    • Low priority work assigned to appropriate service class with different performance objectives
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
- * see appendix for more info
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 (which is no goal, i.e. discretionary) unless they can be assigned 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
Enclaves Can Use Multiple Periods

- **PERIOD 1**
  - 90% in 0.5 sec; IMP = 2
  - DUR=300

- **PERIOD 2**
  - 90% in 4 sec; IMP = 3
  - DUR=600

- **PERIOD 3**
  - Ex Vol = 20
  - IMP = 4

**Response time and Velocity measures**

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

- The **DURation** 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 DDFPROD
What is a DDF Transaction?

Threads: ZPARM CMTSTAT = Inactive

- DRDA unit-of-work 1
  - Queue time
  - Execution time
  - Idle
- DRDA unit-of-work 2
  - Queue time
  - Execution time

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
- **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
- Goals should be attainable when system not under stress
  - If not, then **Performance Indexes (PI) > 1** may affect workload balancing (as of z/OS 1.7 and APAR PK03045)
**DDF Classification Rules, example**

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

![DDF Classification Rules Example](image)

**Subsystem Type:** DDF  
**Fold qualifier names:** Y (Y or N)  
**Description:** DDF work requests

**Action codes:**  
- A = After  
- B = Before  
- C = Copy  
- D = Delete row  
- M = Move  
- I = Insert rule  
- IS = Insert Sub-rule  
- More = More

<table>
<thead>
<tr>
<th>Action</th>
<th>Type</th>
<th>Qualifier</th>
<th>Start</th>
<th>Service</th>
<th>Report</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SI</td>
<td>DB9A</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>PC</td>
<td>TRX*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>AI</td>
<td>TOTOA*</td>
<td>56</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>UI</td>
<td>PAOLOR3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SI</td>
<td>D9C*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>UI</td>
<td>PAOLOR3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PC</td>
<td>DB2J*</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Defaults:**  
- DDFDEF  
- DDFDEF  
- DDFPROD  
- RSSL  
- DDFTOT  
- RNISANTI  
- DDFDEF  
- DDFTEST  
- DDFPROD

**Bottom of Data**

- F1 = Help  
- F2 = Split  
- F3 = Exit  
- F4 = Return  
- F7 = Up  
- F8 = Down  
- F9 = Swap  
- F10 = Left  
- F11 = Right  
- F12 = Cancel
## 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_WRKSTNNNAME - 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
#3 - DB2 Parallel Query

**Query CP Parallelism**

- Host 1: Starts as Client SRB, then switches to enclave SRB to become zIIP eligible.
- PARTITIONED TABLESPACE
- Complex query originates here.

**Sysplex Query Parallelism: DB2 Data Sharing**

- Host 1
- Host 2
- Host 3
- Portions of complex query arrive on participant systems, classified under "DB2" rules, and run in enclave SRBs.
#4 - DB2 Stored Procedures

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

Task
- Creates dependent enclave
- Continuation of transaction CHARLIE

KEY:
- TCB
- Enclave SRB
WLM Managed DB2 Stored Procedures

- Stored Procedures run in WLM managed server regions in an application environment
- Distributed requests (DDF)
  - DBM1 processes SQL request under existing Enclave
- Local requests (e.g. CICS, batch, TSO, etc.)
  - DBM1 creates a dependent Enclave
- DBM1 inserts work request into WLM work queue
  - Available task (TCB) in server region selects the request and processes it under the Enclave
- Beware loose goals!
  - WLM may not start more server regions if PI <= 1
Native SQL Procedures (beginning with DB2 9 for z/OS)

The SQL procedure logic runs in the DBM1 address space

Execution from remote thread eligible for zIIP at same percentage as DDF Enclave SRB
V9 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
Resources

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

• Special thanks to:
  – Hugh Smith, DB2 Development
  – Glenn Anderson, IBM Learning Services
Disclaimer

© Copyright IBM Corporation 2010. All rights reserved.
U.S. Government Users Restricted Rights - Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

THE INFORMATION CONTAINED IN THIS PRESENTATION IS PROVIDED FOR INFORMATIONAL PURPOSES ONLY. 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. IN ADDITION, THIS INFORMATION IS BASED ON IBM’S CURRENT PRODUCT PLANS AND STRATEGY, WHICH ARE SUBJECT TO CHANGE BY IBM WITHOUT NOTICE. IBM SHALL NOT BE RESPONSIBLE FOR ANY DAMAGES ARISING OUT OF THE USE OF, OR OTHERWISE RELATED TO, THIS PRESENTATION OR ANY OTHER DOCUMENTATION. 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 ANY AGREEMENT OR LICENSE GOVERNING THE USE OF IBM PRODUCTS AND/OR SOFTWARE.

IBM, the IBM logo, ibm.com, CICS, DB2, IMS, z/OS, and Parallel Sysplex and 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.
Appendix

Additional information about WLM and enclaves.
SRBs and TCBs

- SRB and TCB are fundamental to the MVS operating system. They represent different execution modes - different ways to use CPU resources.
  - **TCB = Task Control Block**
    - Lower priority
    - Longer lasting
    - Interruptible
    - Executes within an address space, and most application code runs in TCB mode
    - DB2 examples: SQL (INSERT, UPDATE, DELETE, etc.) including Stored Procedures, UDF, etc.
  - **SRB = Service Request Block**
    - High priority
    - Short duration
    - Not interruptible
    - Generally SRBs are performed on your behalf by the system or a subsystem
    - DB2 examples: asynchronous writes, prefetch
  - Most of DBM1 and MSTR is SRB mode processing. Most CICS transactions and TSO or batch jobs execute in TCB mode.
Enclaves and Enclave SRBs

- Relatively new to MVS
- Enclaves were invented to handle DDF (distributed) workload and to allow WLM to manage distributed threads with different priorities.
- Enclaves represent a business unit of work and can include TCBs and SRBs that execute in multiple address spaces.
- Enclave SRBs behave like TCBs and are interruptible (we usually say 'preemptible').
Enclave Services: A Dispatching Unit

• Standard dispatching
  – Dispatchable units (DUs) are the TCB and the SRB
    • TCB runs at dispatching priority of address space and is preemptible
    • SRB runs at supervisory priority and is non-preemptible

• Advanced dispatching units
  – Enclave
    • Anchor for an address space-independent transaction managed by WLM
    • Can comprise multiple DUs (TCBs and Enclave SRBs) executing across multiple address spaces
  – Client SRB
    • Created and executed like an ordinary SRB but runs with client (scheduler) dispatching priority and is pre-emptible
  – Enclave SRB
    • Created and executed like an ordinary SRB but runs with Enclave dispatching priority and is pre-emptible

• Enclave Services enable a workload manager to create and control enclaves
WLM Interfaces DB2 Uses

- Enclave management
  - CREATE, DELETE, JOIN, LEAVE
  - Independent (DDF) and Dependent (when external stored procedures called from an allied thread)
  - Workdependent – used for parallel query zIIP offload
    * WLM APAR OA26104 installs support
    * APAR PK76676 (V8) and PK87913 (V9) will utilize WORKDEPENDENT enclaves to offload parallel query processing to zIIPs under original task or DDF enclave goals
    * If DB2 apars not applied, independent enclaves are created in SUBSYS=DB2 which then requires classification rules
  - EWLM response time monitoring
  - zIIP support
- Child SRBs
  - Primarily for parallelism tasks
- Sysplex Routing (Data Sharing)
  - Register/deregister (DDF startup/shutdown/parameter change)
  - Determine where to route work (Create server list)
    * FUNCTION=SELECT (pre z/OS 1.7) – displaceable capacity only
    * FUNCTION=SPECIFIC – displaceable capacity influenced by performance index and queuing of served enclaves
What is a WLM Transaction?

- A WLM transaction represents a WLM "unit of work"
  - basic workload entity for which WLM collects a resource usage value
  - foundation for statistics presented in workload activity report
  - represents a single subsystem "work request"
- Subsystems can implement one of three transaction types
  - Address Space:
    - WLM transaction measures all resource used by a subsystem request in a **single address space**
    - Used by JES (a batch job), TSO (a TSO command), OMVS (a process), STC (a started task) and ASCH (single APPC program)
  - Enclave:
    - Enclave created and destroyed by subsystem for each work request
    - WLM transaction measures resources used by a single subsystem request across **multiple address spaces and systems**
    - Exploited by "new workload" subsystems - Component Broker(WebSphere), DB2, DDF, IWEB, MQSeries, LDAP, NETV, TCP
  - CICS/IMS Transactions
    - Neither address space or enclave oriented - special type
    - WLM transaction measures resource used by a single CICS/IMS transaction program request
More on Enclave SRBs

• An Enclave SRB is primarily for distributed threads. When a thread comes into DDF, DDF works with WLM to create an enclave. The SQL requests for that thread execute under that enclave. WLM will manage resources for that enclave. So most of the distributed requests may be eligible for zIIP processing.

• If the distributed thread issues a CALL statement to start a Stored Procedure, that Stored Procedure will execute in TCB mode, but will remain associated with the enclave. Thus the Stored Procedure will NOT be eligible for zIIP processing. Such things as COMMIT, and passing result sets from the Stored Procedure execute under enclave SRB.
What are enclave SRBs?

- z/OS dispatches DB2 work in either TCB (Task Control Block) mode if request is local or SRB (Service Request Block) mode if request is distributed. Under these modes of operation the parallel tasks are assigned the same importance as the originating address space.

- Pre-emptible enclaves are used to do the work on behalf of the originating TCB or SRB address space. Enclaves are grouped by common characteristics and service requests and since they are pre-emptible, the z/OS dispatcher (and WLM) can interrupt these tasks for more important ones (i.e. manage a transaction end-to-end). There are two types of pre-emptible SRBs: client SRBs and enclave SRBs.

- If the DB2 for z/OS V8 (or beyond) request is coming in over distributed (i.e. DRDA over TCP/IP) then that work is executed in enclave SRBs.

- If the request is coming over local/ native connection, then that work is dispatched between TCBs, client SRBs, and enclave SRBs (star schema queries and index maintenance now use enclave SRBs)

- So...... regarding the zIIP, only the enclave SRB work (not the client SRB work) is eligible to be redirected to the zIIP.

- DB2 knows how its work is dispatched and directs z/OS to dispatch (redirect) a portion of the eligible work to the zIIP.
DB2 Distributed: DSNZPARM CMTSTAT

- DRDA connections use *inactive connection* support
  - DSNZPARM: CMTSTAT=INACTIVE
  - Upon commit, DBAT marked in DISCONN state (*pooled*) and connection becomes inactive
  - New unit of work (UoW) request from any user connection will cause DBAT in pool to be associated with that connection
  - After 200 state switches, DBAT is purged
  - After POOLINAC of time in pool, DBAT is purged
- If DBAT cannot be pooled at commit, it behaves like an active connection
  - DBAT remains associated with the connection
  - DBAT cannot be used by other requests
  - Enclave remains and transaction includes think time
Path of the Distributed SRB (V8 and beyond)
DBAT Is Pooled When:

- No WITH HOLD cursors are open
  - temporary condition which can be resolved by client closing cursor and issuing another commit
  - ODBC/CLI/JDBC/… client has a default of WITH HOLD
    - Change default via db2cli.ini file set to CURSORHOLD=0
- No Declared Global Temporary Tables exist on the connection
  - temporary condition which can be resolved by the application dropping the table and then issuing another commit
- No package (stored procedure, trigger, UDF, or nonnested task) with KEEP_DYNAMIC YES bind option has been accessed
  - permanent condition until thread is terminated or ROLLBACK has been issued
- No LOB locators held