



DB2: Understanding DB2 Usage and Measurement of WLM Enclaves

Peter Enrico

Email: Peter.Enrico@EPStrategies.com

Enterprise Performance Strategies, Inc.

3457-53rd Avenue North, #145

Bradenton, FL 34210

<http://www.epstrategies.com>

<http://www.pivotor.com>

Voice: 813-435-2297

Mobile: 941-685-6789



z/OS Performance
Education, Software, and
Managed Service Providers



Creators of Pivotor®

Abstract



DB2: Understanding DB2 Usage and Measurement of WLM Enclaves

- WLM Enclaves have been available for about 28 years, but enclaves, and associated measurements, are still widely misunderstood. DB2 makes heavy usage of enclaves, and with the expansion of DB2 DBATs and High Performance DBATs, using and interpreting the WLM measurements to evaluate DB2 workloads has become both more important and more difficult.
- During the presentation Peter Enrico will demystify WLM enclaves, DB2's usage of enclaves, and also demystify associated measurements used to evaluate WLM management of DB2 workloads.

EPS: We do z/OS performance...



- **Pivotor** – z/OS performance reporting and analysis software and services
 - Not just SMF reporting, but analysis-based reporting based on expertise
 - www.pivotor.com
- **Education and instruction**
 - We teach our z/OS performance workshops all over the world
 - Want a workshop in your area? Just contact me.
- **z/OS Performance War Rooms**
 - Intense, concentrated, and highly productive on-site performance group discussions, analysis and education
 - Amazing feedback from dozens of past clients
- **MSU Reduction Exercises**
 - The goal is to reduce the MSU consumption of your applications and environment
- **Information**
 - We present around the world and participate in online forums
 - <https://www.pivotor.com/content.html>
<https://www.pivotor.com/webinar.html>



z/OS Performance workshops available



During these workshops you will be analyzing your own data!

- WLM Performance and Re-evaluating Goals
 - April 28 - May 2, 2025
- Parallel Sysplex and z/OS Performance Tuning
 - July 15 - 16, 2025
- Essential z/OS Performance Tuning
 - September 22 - 26, 2025
- Also... please make sure you are signed up for our free monthly z/OS educational webinars! (email contact@epstrategies.com)



Like what you see?



- Free z/OS Performance Educational webinars!

- The titles for our Summer / Fall 2024 webinars are as follows:

- ✓ *What a z/OS Guy Learned About AWS in 10 Years*
- ✓ *Advantages of Multiple Period Service Classes*
- ✓ *Understanding z/OS Connect Measurements*
- ✓ *WLM and SMF 99.1 – System Measurements Deeper Dive*
- ✓ *WLM and SMF 99.2 – Service Class Period Measurements Deeper Dive*
- ✓ *Optimizing Performance at the Speed of Light: Why I/O Avoidance is Even More Important Today*
- ✓ *Understanding MVS Busy % versus LPAR Busy % versus Physical Busy %*
- ✓ *Rethinking IBM Software Cost Management Under Tailored Fit Pricing*
- ✓ *Understanding Page Faults and Their Influence on Uncaptured Time*
- ✓ *Response Time Goals: Average or Percentiles?*
- *Understanding and Using Enclave*

- This is our last webinar for the 2024 season

- Please send Peter and/or Scott emails with suggestions for 2025 webinars
- We would love to hear your suggestions

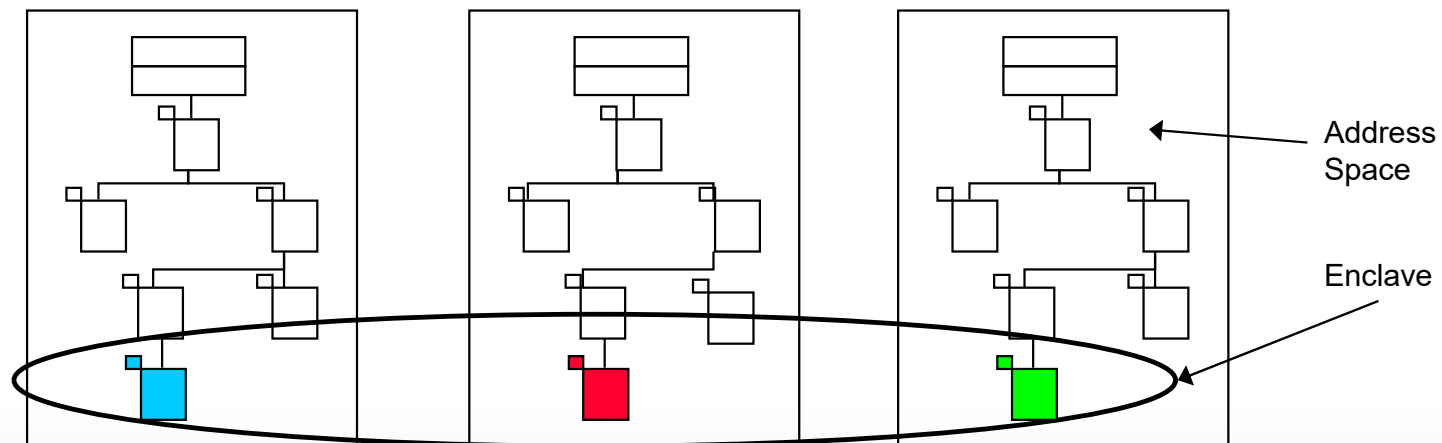


Introduction to Enclaves

Definition of Enclave



- Very different than address spaces, but similar in concept
- An enclave represents a transaction that can span multiple dispatchable units of work (SRBs and TCBs) in one or more address spaces, and reported and managed as a single unit



Objectives of Enclaves



- Objective 1
 - Manage work separately from home address space
- Objective 2
 - Manage work separately from other work running in same address space
- Objective 3
 - Manage work against other work running in other address spaces
- Objective 4
 - Separately monitor and report enclave activity
- Objective 5
 - Allow distributed work running in multiple address spaces to be viewed and managed as a single unit of work

Enclave Objective 1

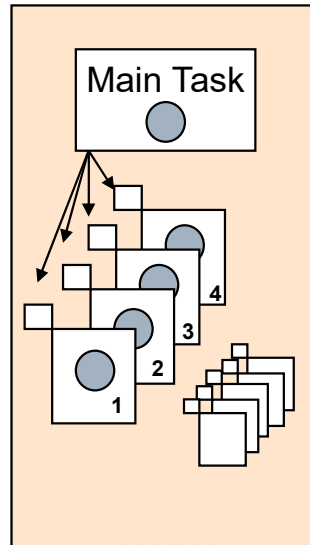
Objective 1 - Manage work separately from home address space

- Prior to enclaves, management of the home address space dictated the qualities of service all the units of work running in that address space received
- With enclaves, the units of work running in the address space can be given a different quality of service than the address space itself

CPU Dispatching Priority

All tasks have the same CPU Dispatch Priority and managed at Address Space level.

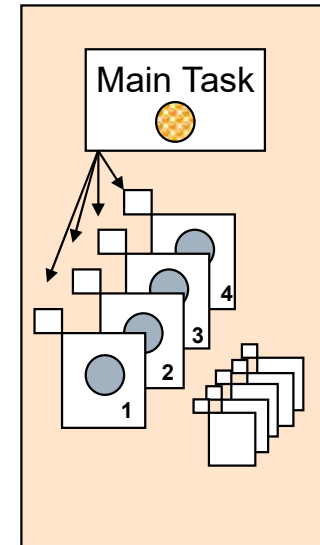
Address Space Without Enclaves



CPU Dispatching Priority

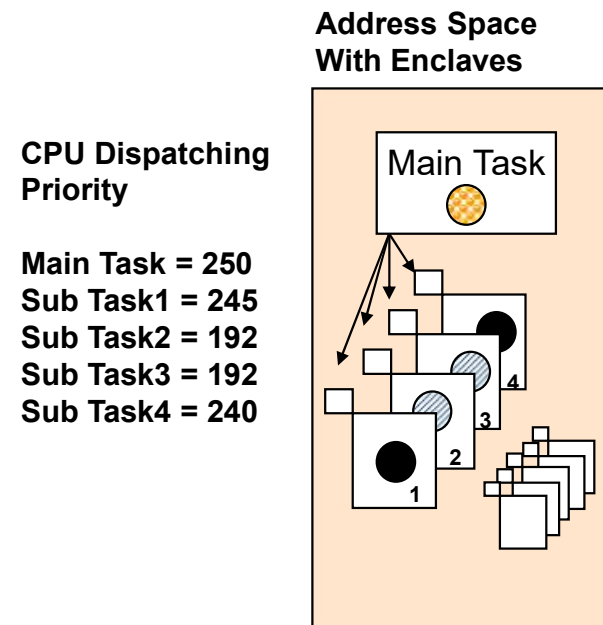
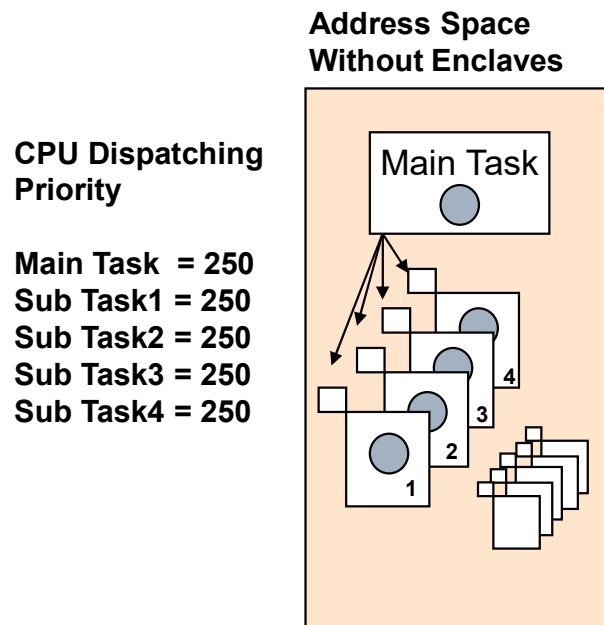
Main Task = 250
All other task could have a difference Dispatch Priority

Address Space With Enclaves



Enclave Objective 2

- Objective 2 - Manage work separately from other work running in same address space
 - Different types of work running in the same server address space can be assigned to difference service classes, periods, and even have different goals



Enclave Objective 3

- Objective 3 - Manage work against other work running in other address spaces
 - Work running in enclaves can be assigned different goals and importance levels than unlike work running in other address spaces.

CPU Dispatching Priority

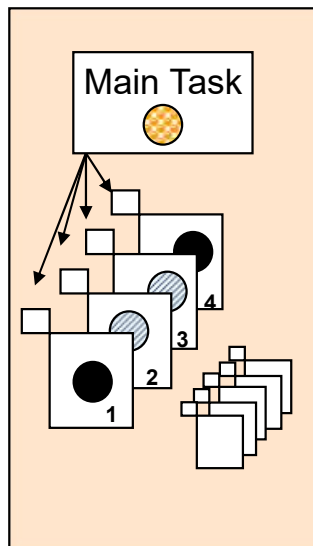
Main Task = 250
Sub Task1 = 248 (PRODQ)
Sub Task2 = 192 (TESTQ)
Sub Task3 = 192 (TESTQ)
Sub Task4 = 240 (PRODOTHR)

Service Class
 • PRODQ
 • AVG RT 1, 95%
 • Imp 1

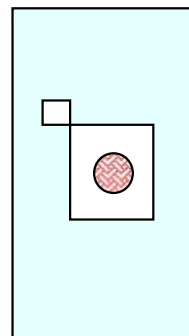
Service Class
 • TESTQ
 • Discretionary

Etc.

Address Space With Enclaves

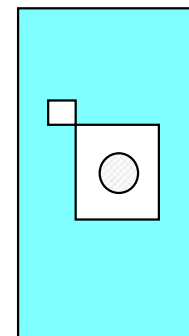


TSO Period 1 - CPU DP 240



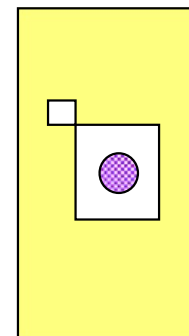
Service Class
 • PRODTSO
 • AVG RT .5, 95%
 • Imp 3

Test Batch - CPU DP 192



Service Class
 • TESTBAT
 • Discretionary
 • n/a

CICS Regions - CPU DP 248



Service Class
 • CICSPROD
 • Velocity 50
 • Imp 2

Enclave Objective 4

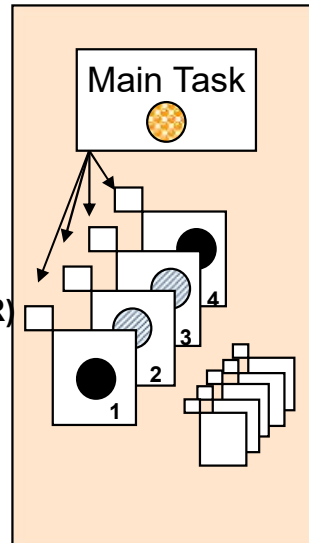
Objective 4 – Separately monitor and report enclave activity

- Like address spaces, we can monitor enclave activity to understand enclave performance and resource consumption

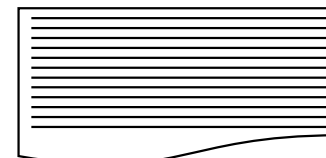
CPU Dispatching
Priority

Main Task = 250
Sub Task1 = 248 (PRODQ)
Sub Task2 = 192 (TESTQ)
Sub Task3 = 192 (TESTQ)
Sub Task4 = 240 (PRODOTHR)

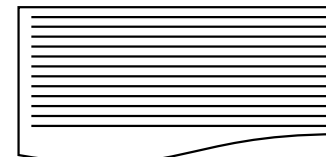
Address Space
With Enclaves



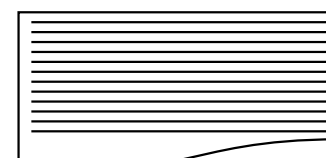
RMF Report
Service Class: ADDRSPAC
PI = 1.5



RMF Report
Service Class: PRODQ
PI = .5



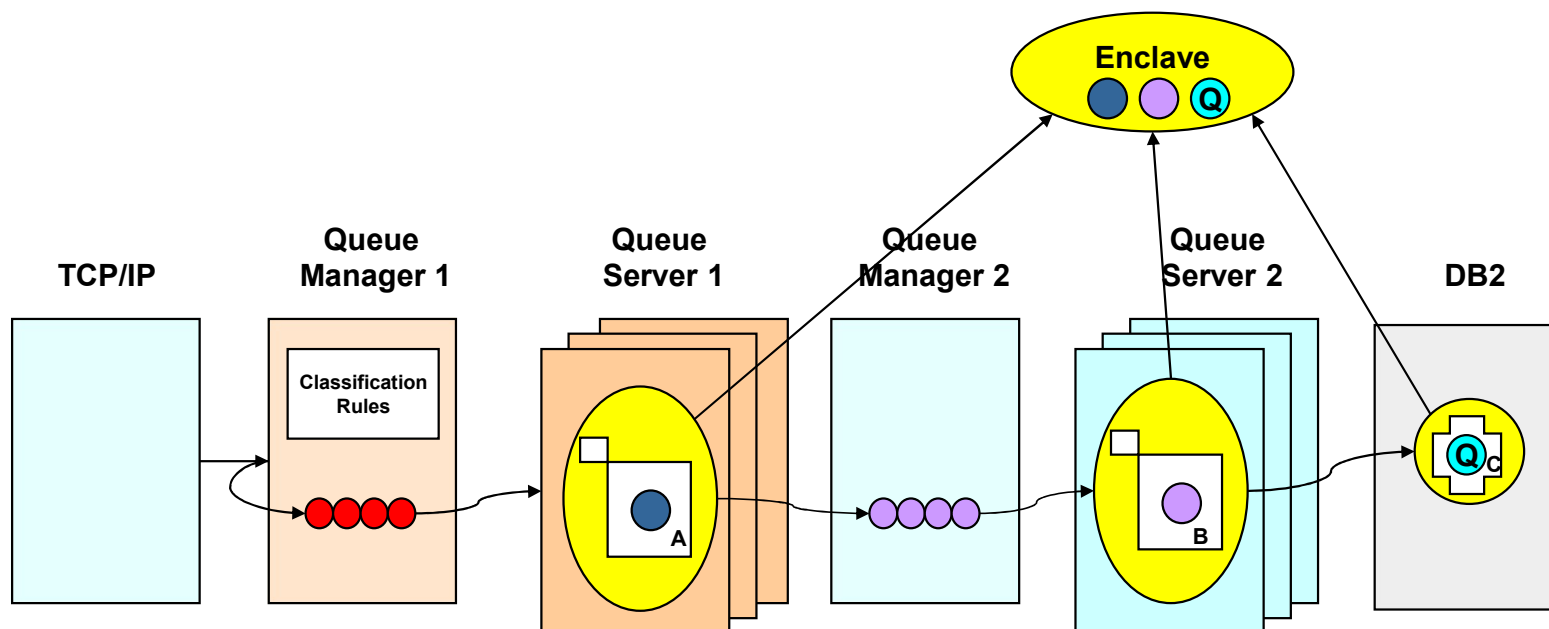
RMF Report
Service Class: TESTQ
PI = n/a



RMF Report
Service Class: PRODOTHR
PI = 1.2

Enclave Objective 5

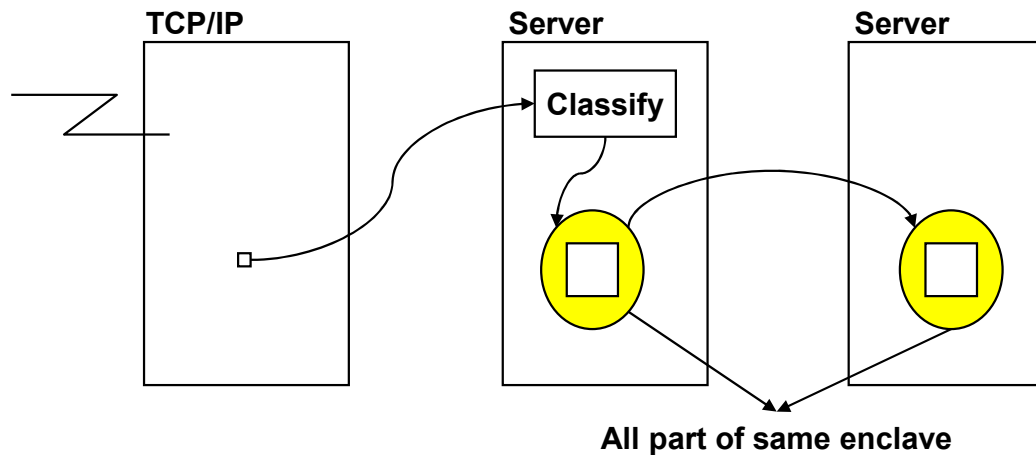
- Objective 5 – Allow distributed work running in multiple address spaces be looked at and managed as a single unit
 - All the units of work shown in the below diagram are parts of a single transaction
 - The transaction spans multiple address spaces
 - In this example, they are all part of one enclave



Types of Enclaves - Independent

□ Independent Enclave

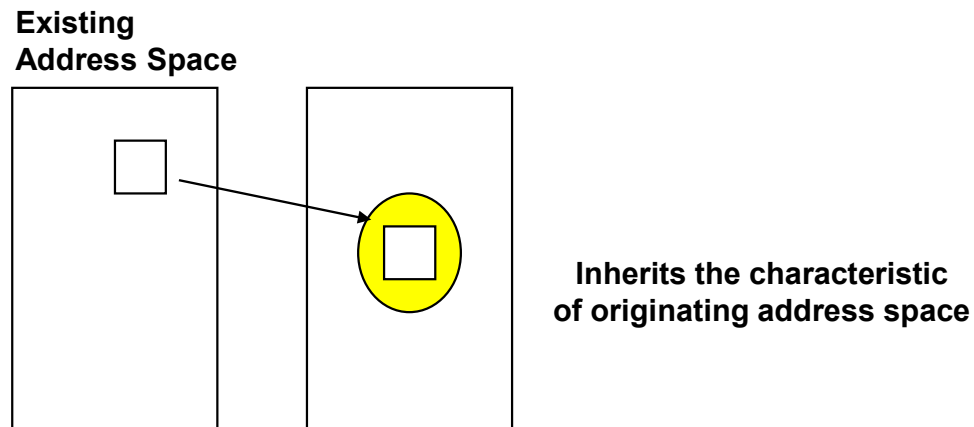
- Represents a complete independent transaction (ie. a true SRM transaction)
 - Usually, a transaction that is arriving from the network
- Separately classified and managed in a service class
- Transaction starts, enclave created, classified, runs



Types of Enclaves - Dependent

□ Dependent Enclave

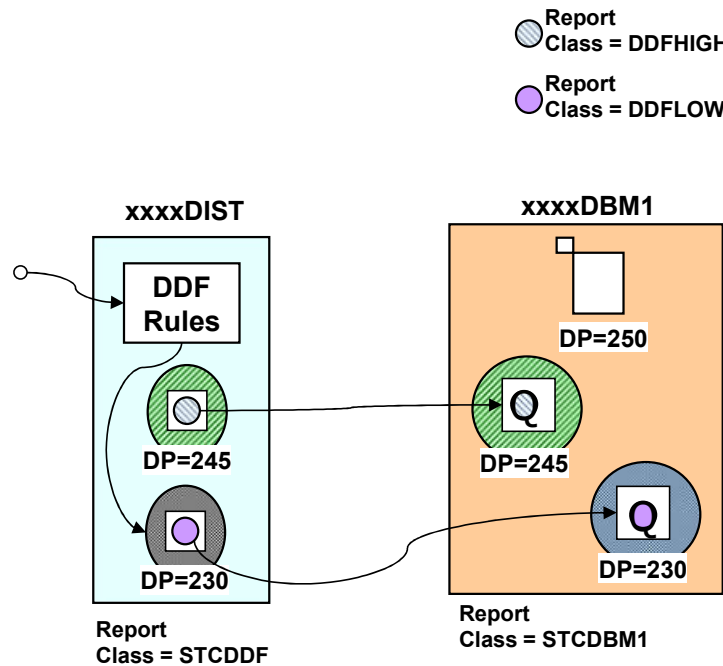
- Represents a transaction that is a continuation of an already existing transaction
- Starts in an address space and 'spawns' into an enclave
- Inherits the service class of the original transaction
- Accumulated service treated as though it were accumulated by the requesting address space



DDF's Usage of Enclaves

DDF and Independent Enclaves with zIIP Engines

- When zIIP engines are configured SMF30ENC is the enclave CPU time that was qualified for zIIP but was prevented from running on zIIP
 - Ran on CP engines
- zIIP time is zIIP eligible time and not included in SMF30ENC
 - Time on zIIP + zIIP eligible but ran on CP



SMF 72
STCDDF
CPU=1%

SMF 72
STCDBM1
(for these transactions)
CPU=1%

SMF 72
DDFLOW
CPU=26.25%
zIIP=33.75

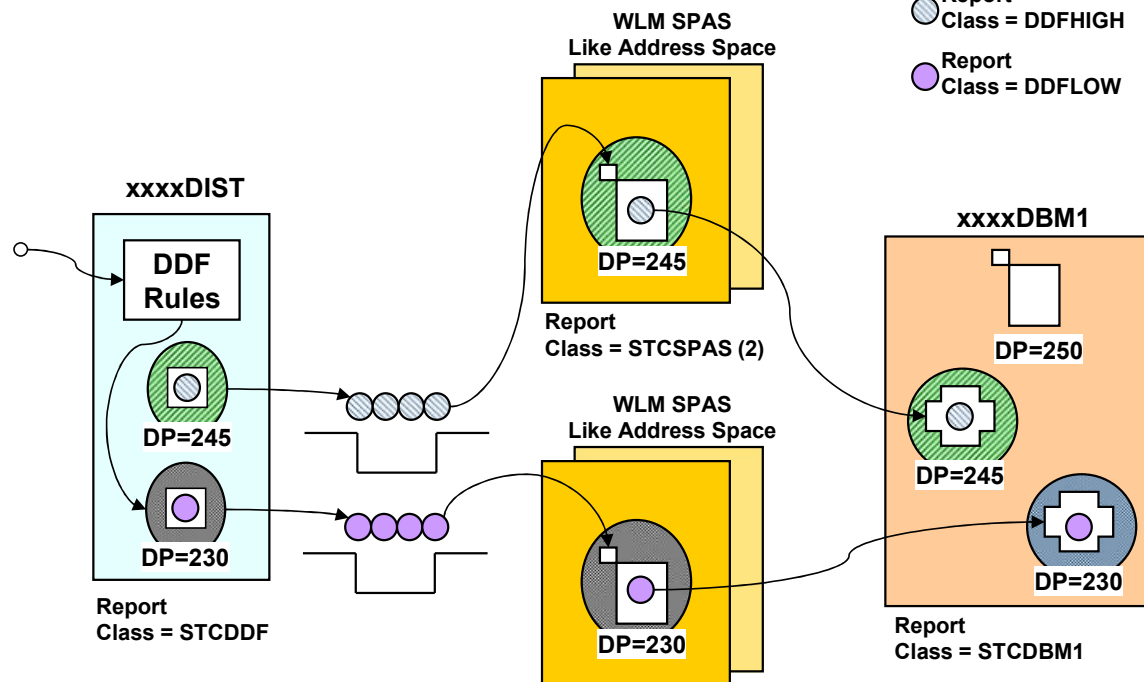
SMF 72
DDFHIGH
CPU=8.75%
zIIP=11.25%

SMF 30
DDF
CPU=36%
ENC=35%
zIIP=45%

SMF 30
DBM1
CPU=1%
ENC=0%
zIIP=0%

AE CPU Times and Queue Times

- ❑ CPU times can be confusing
 - Charged back to address space that created the enclave (usually queue manager)
- ❑ Response times can be confusing
 - Includes queue times which are accumulated whenever transaction queued



SMF 72
STCDDF
CPU=1%
QT=0

SMF 72
STCDBM1
CPU=1%
QT=0

SMF 72
STCSPAS
CPU=2%
QT=0

SMF 72
DDLOW
CPU=60%
QT=x

SMF 72
DDFHIGH
CPU=20%
QT=y

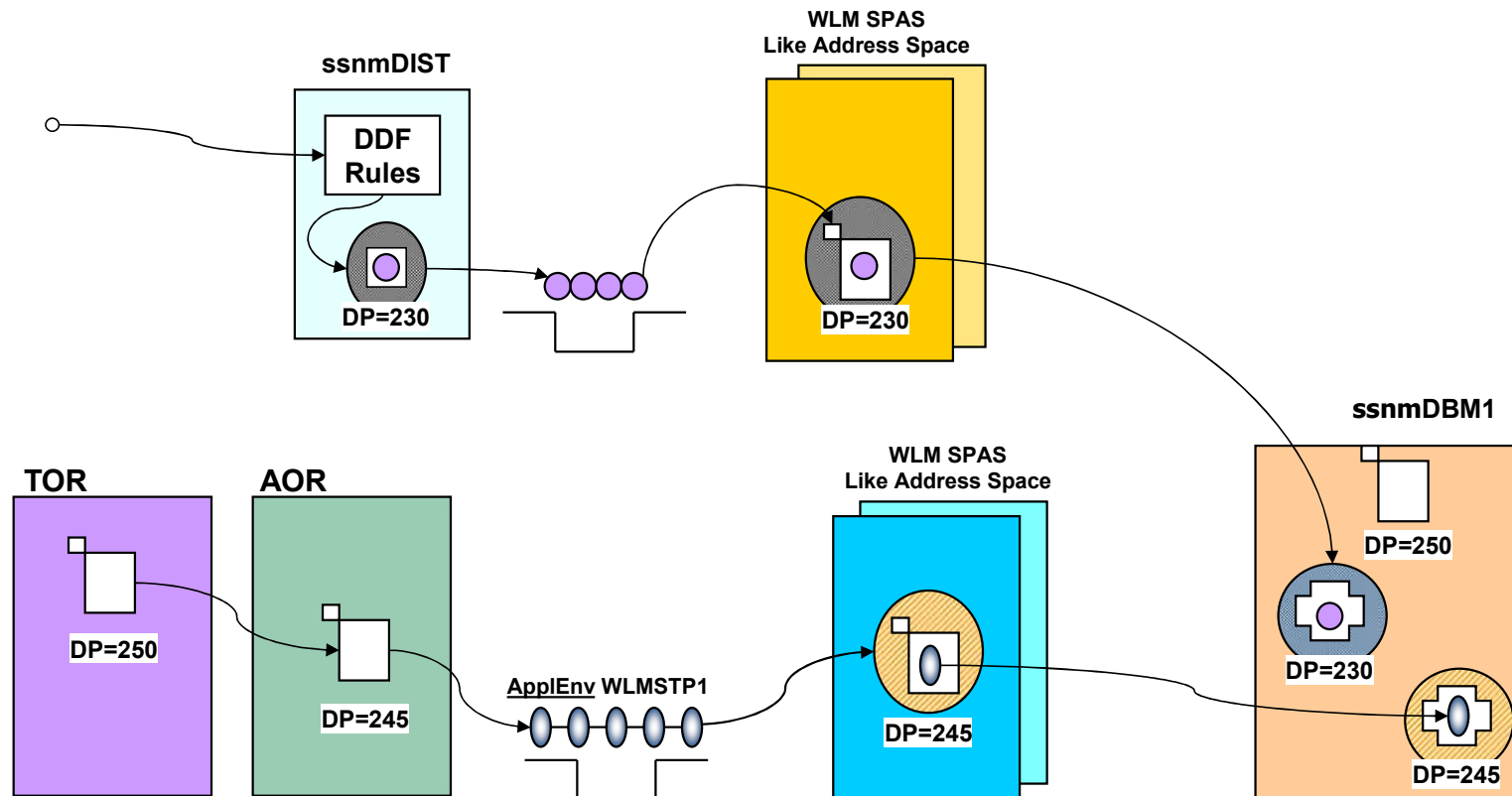
SMF 30
DDF
CPU=81%
ENC=80%

SMF 30
DBM1
CPU=1%
ENC=0%

SMF 30s
SPAS
CPU=1%
ENC=0%

DB2 Stored Procedures and AEs

- Stored procedure client could be IMS, CICS, TSO, Batch, Started Tasks, Distributed DB2
 - In this example, a stored procedure being called by CICS is an example of a dependent enclave



DDF Management Recommendations

- Don't be afraid of using multiple DDF Service Classes!
- Make liberal use of Report Classes (e.g. by authid)
 - Helps determine what application is doing how many transactions and consuming how much CPU: all from the SMF 72 records without having to look at the voluminous 101s
 - For things like QMF with a large number of adhoc users, may look for correlation IDs
- Treat your default DDF Service Class like batch
 - Default should generally not be "like online users"
 - Also helps catch new DDF exploiters to get them classified to a good RC
- Consider 2-3 period SCs, except for work that is well-known and well-behaved
 - 3rd period may be a "penalty period" for those outliers
- Consider RT goals for first 1 or 2 periods
 - Generally easier to describe, monitor, and relate to application performance

SUs and Period Transitions



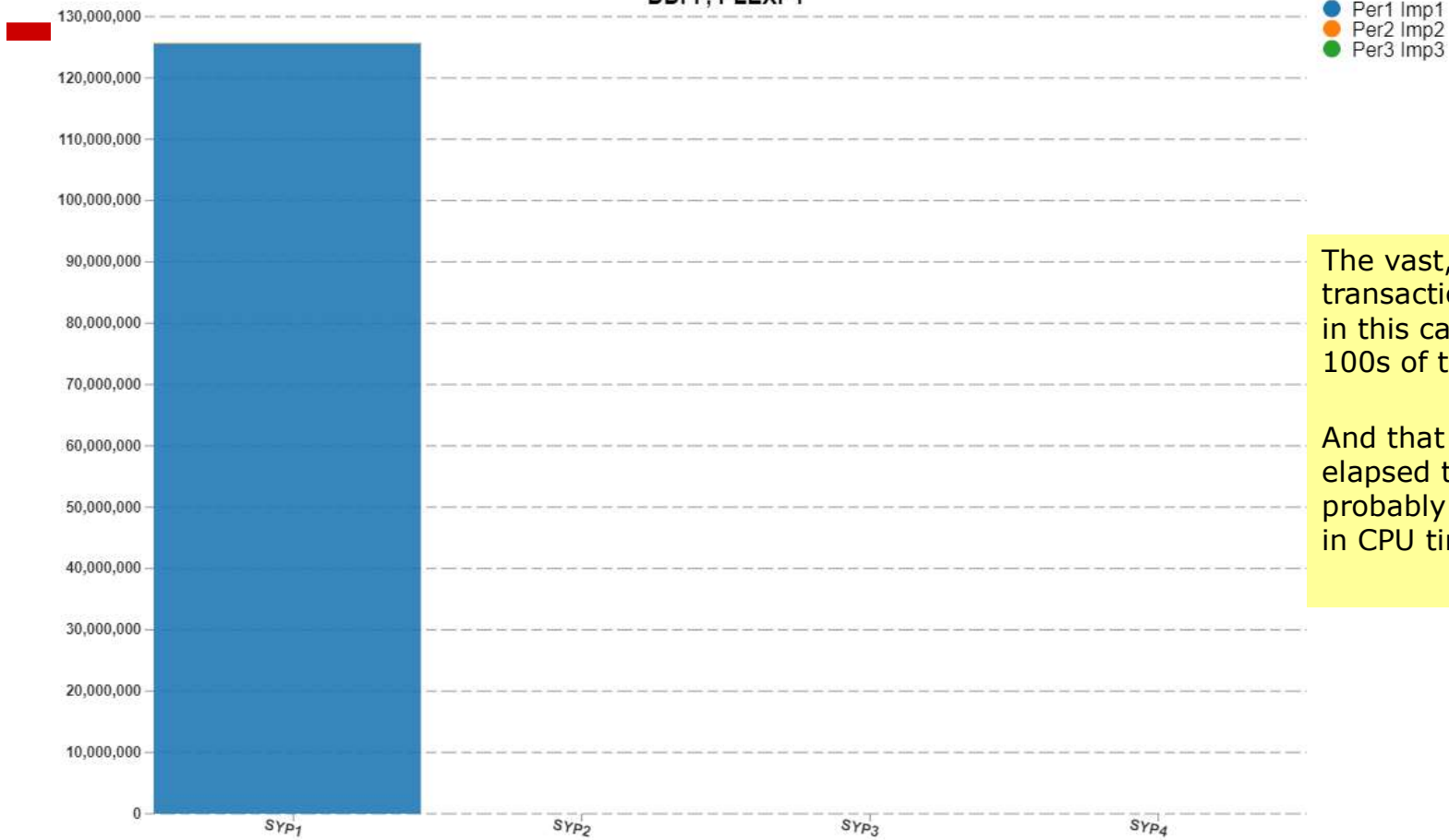
- For Service Classes with multiple periods, work transitions between periods **as it consumes resources (CPU)**



- Each period has its own importance and goal
- So we can automatically adjust the management of long-running work
 - E.G. lower the importance and relax the goal so the hogs don't trample the hummingbirds
 - Very useful when a workload contains a mix of light and heavy work

WLM Multiple Period - Total Ended Transactions All Intervals

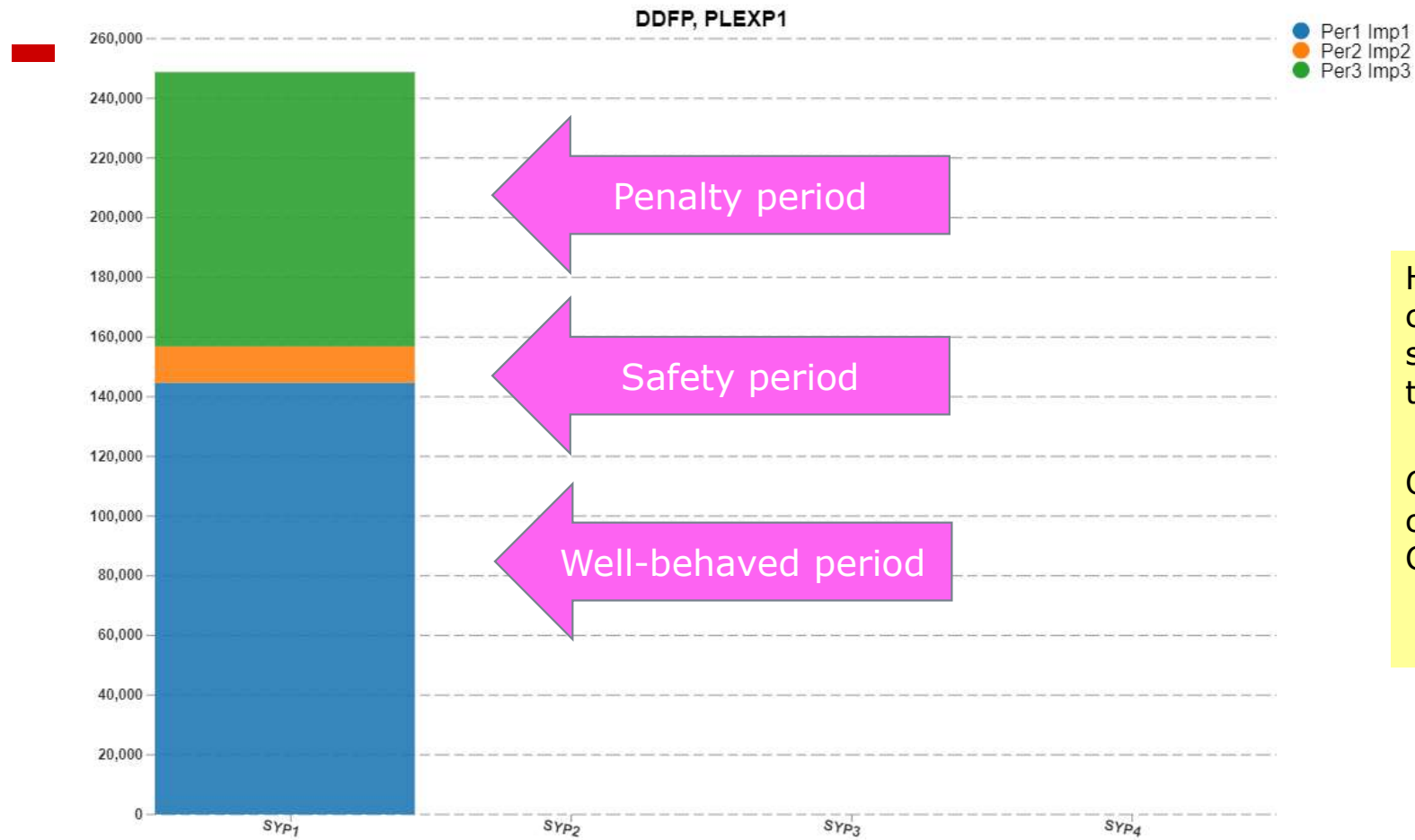
DDFP, PLEXP1



The vast, vast majority of transactions end in period 1. (But in this case, there may be 10s-100s of thousands in P2/P3.)

And that 100,000x increase in elapsed time from P1 to P3 probably has a similar correlation in CPU time.

WLM Multiple Period - Total CPU Seconds All Intervals



Here we see indeed P3 consumed a significant portion of the overall DDFP CPU.

Often, we'll see P3 consuming even more CPU than P1.



New WLM Enclave Measurements

Geared towards 'enclave bundled transactions' in response to High Performance DBATs

Upending >25 Years of DDF Management



Enclaves allow the *management of individual transactions* flowing through address spaces, something that simply has never been possible before. Since MVS is aware of and has access to each transaction, they can be classified individually and most importantly *each transaction is subject to period switch*. ~~This means that you can separate out the long running CPU killers from the shorter requests in the same manner that most installations already employ to control batch, by period level controls.~~

Each enclave is a single transaction, which starts when the enclave is created and ends when the enclave is deleted. ~~DDF creates an enclave for an incoming request when it detects the first SQL statement and deletes the enclave at SQL COMMIT, thus a DDF enclave transaction consists of a single SQL COMMIT scope.~~

~~In WLM goal mode, all goal types are valid for enclaves.~~

This is now the situation if your DDF work uses High Performance DBATs!
(With DB2 APAR PH34378)

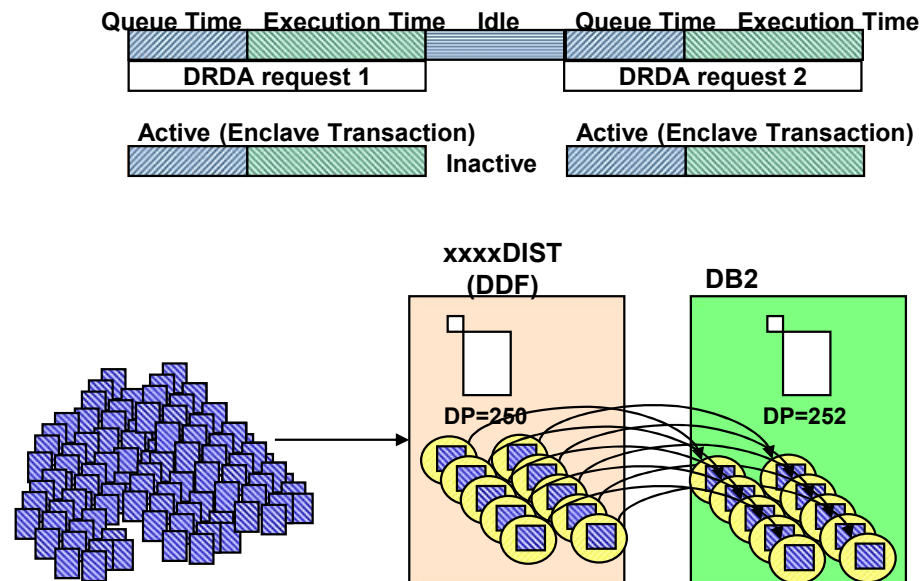
Technically: multi-period service classes and RT goals still are allowed but they will not work as expected!
This could be a surprise when you apply the DB2 maintenance that made this change.

Traditional / Typical DB2 DDF Enclaves



- DDF transactions are managed by WLM as independent enclaves
 - The life of the enclave is the life of the transaction (i.e. 1 tran = 1 ended enclave)

- In this case, 200 transactions come in, and for each transaction an enclave is created
- When each transaction ends, the enclave for that transaction is deleted, and an ended transaction and response time are posted to the WLM Service Class period
- WLM response time goals are best since each transaction is measured and considered by WLM

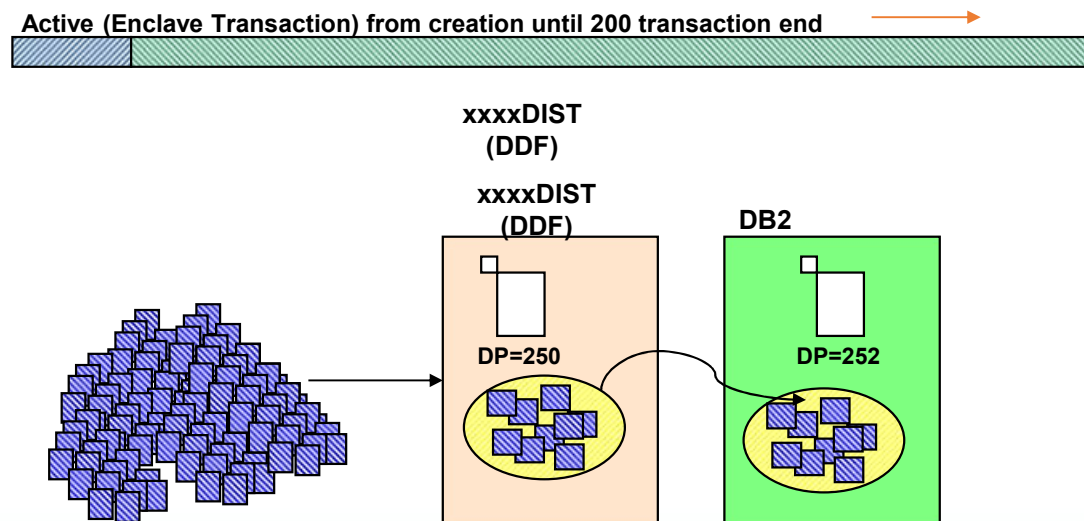


DBAT DB2 DDF Enclaves w/ PH34378



- The DBAT transactions still run in independent enclaves
 - But now the life of the enclave is for as long as up to 200 transactions in the enclave, or 120 seconds
 - The transactions in the enclave are known as ‘bundled transactions’.
 - Thus, you may have 200 ended bundled transactions, but from a WLM point-of-view, the 200 bundled transactions are part of a single (i.e. 1) longer running enclave (up to 500 with DB2 V13)

- In this case, when a new thread is needed, a new thread is created and associated with a new enclave.
- That thread and associated enclave remain active for up to 200 transactions come in, and all are associated with a single enclave.
- After the last of the 200 transactions end, enclave is deleted, and one ended transaction is posted to the WLM service class period. Response time is from start of first transaction to end of last ended transaction
- The many transactions running in the enclave are known as ‘bundled transactions’



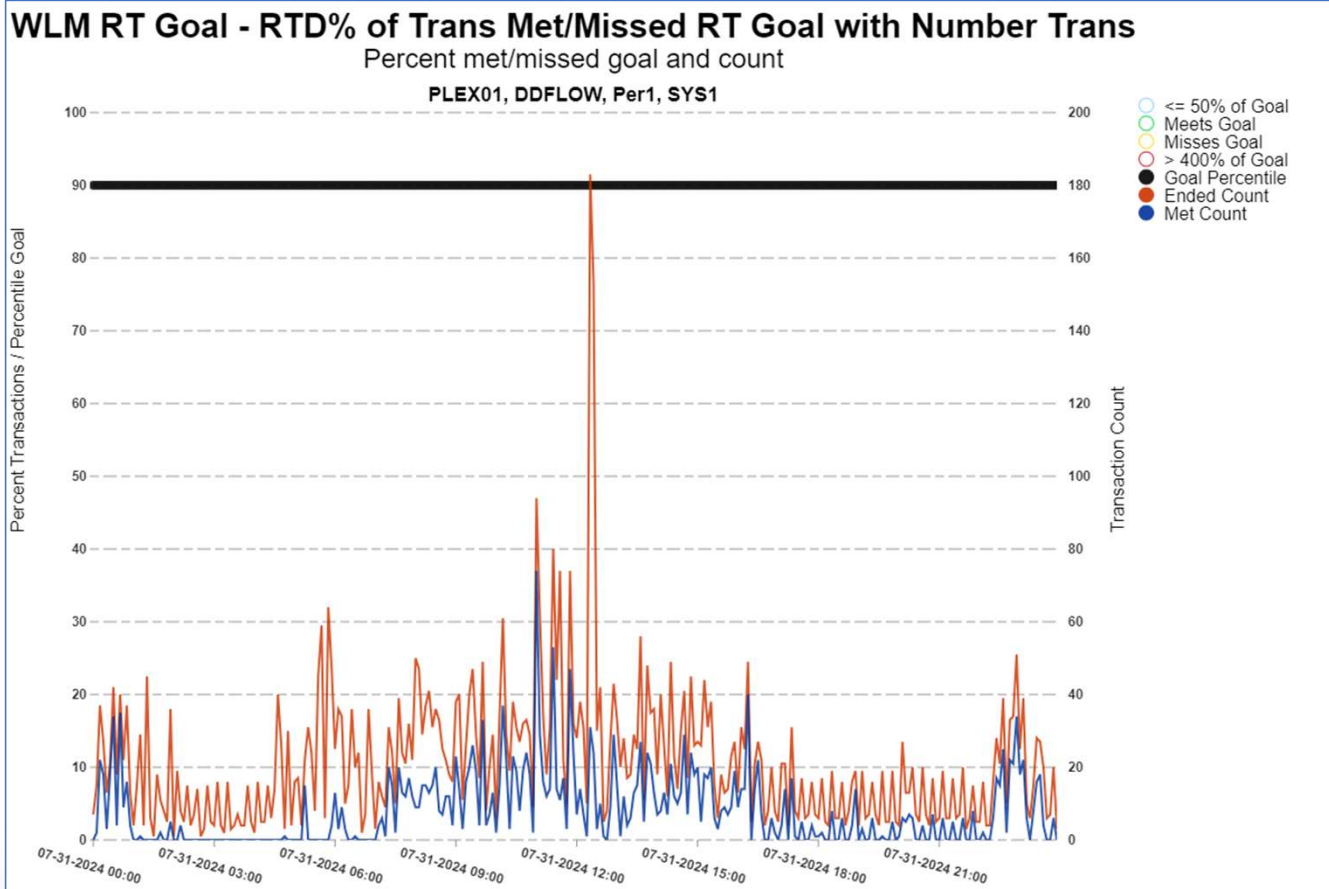
New SMF fields in support of Enclaves



- New SMF 72, subtype 3 SMF fields in support of enclaves
 - Fields are useful to help investigate if DB2 High Performance DBATs are being used

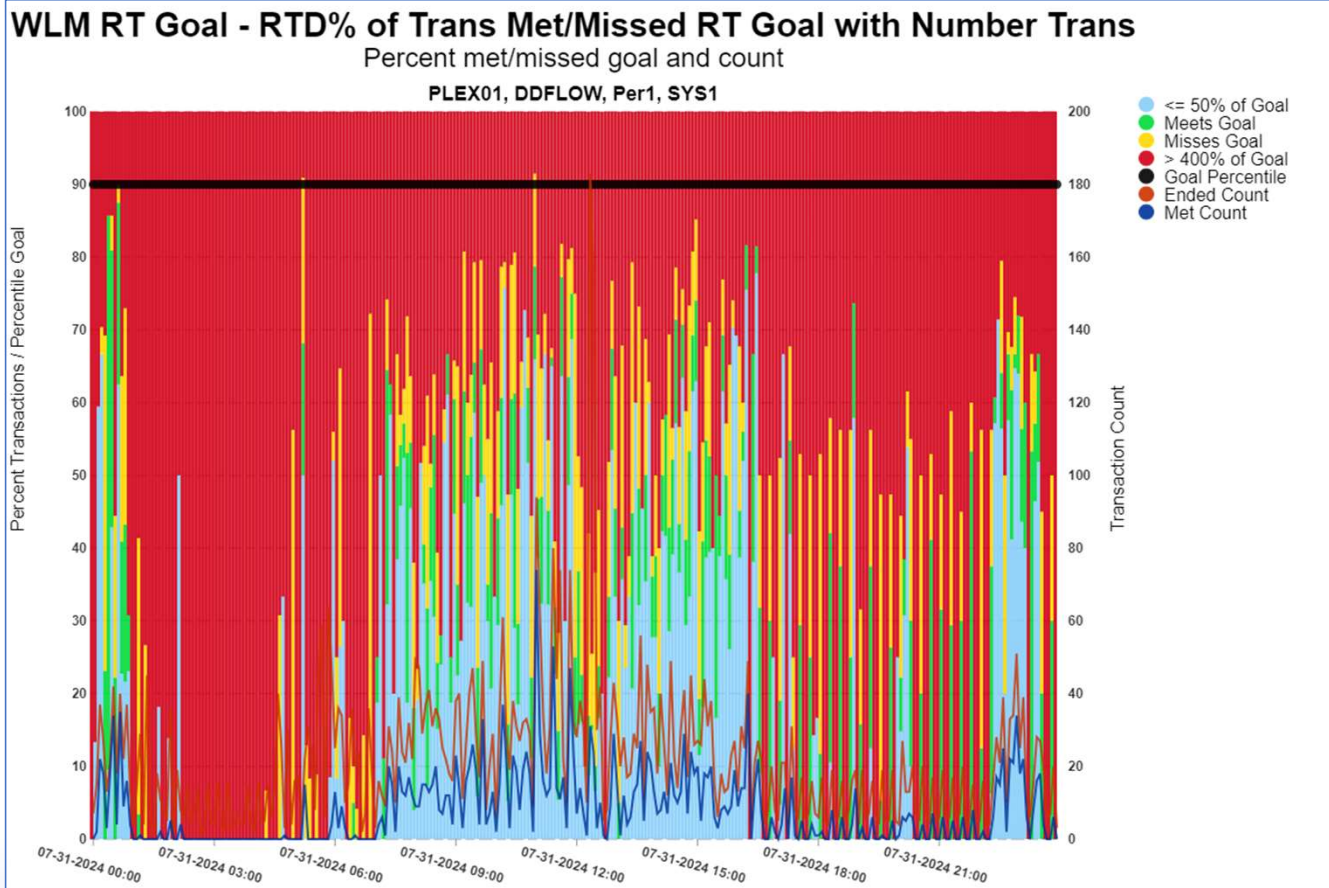
Field Name	Description
R723ENCTRXNUM	Number of subsystem transactions processed within enclaves.
R723ENCTRXCALLS	Number of times transaction data has been reported by subsystem work managers when deleting an enclave. When zero, no transaction data for enclaves has been provided by the subsystem work manager.
R723ENCTRXET	Total execution time, in microseconds, for all subsystem transactions reported in R723ENCTRXNUM
R723ENCTRXETS	Sum of squared execution times, in microseconds, for all subsystem transactions reported in R723ENCTRXNUM.

Example: DDFLOW Response Time Distribution



- DDFLOW has a response time goal of 90% complete in 0.3 seconds
- Notice that we have a fair low number of ended transactions
- Notice the large gap between number of ended transactions and the number that met the goal value.

Example: DDFLOW Response Time Distribution



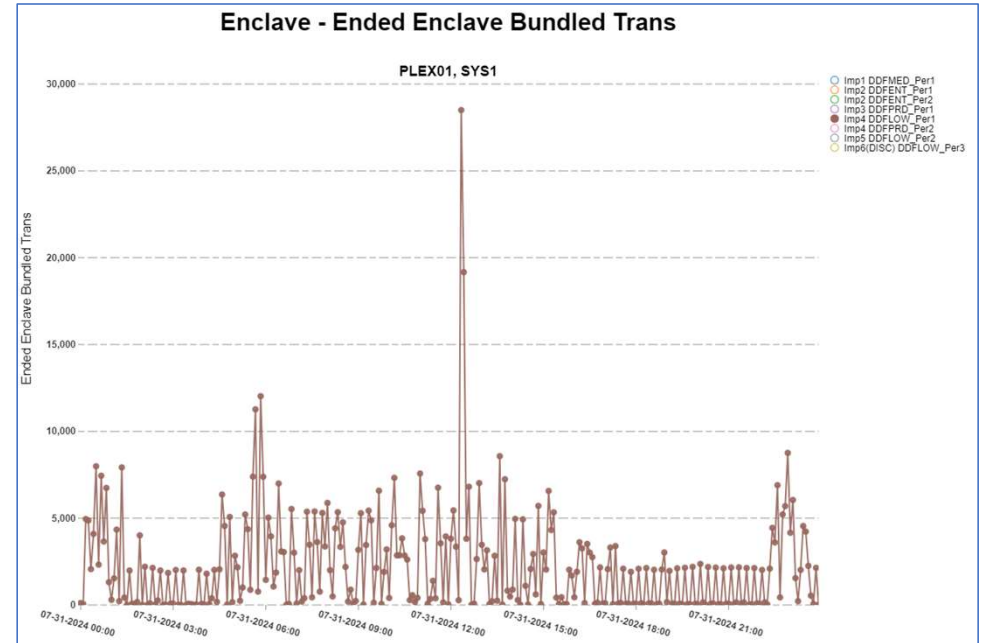
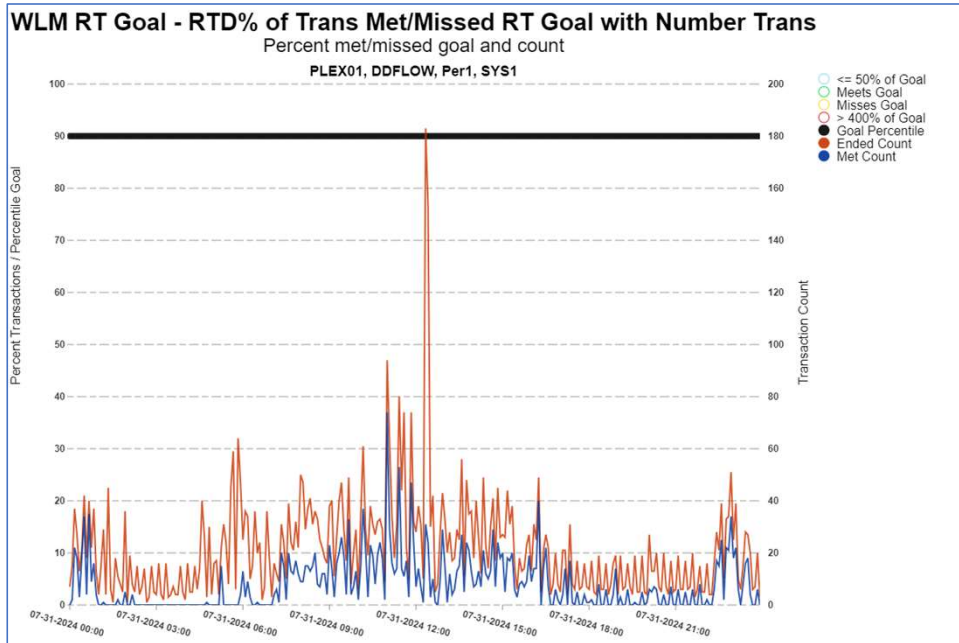
- DDFLOW has a response time goal of 90% complete in 0.3 seconds
- Added to this chart is the percentage of transactions that met and missed the goals
- Not that a large percentage of transactions missed the goal value of 0.3 seconds

DDFLOW Ended Transactions



Ended WLM Transactions (Enclaves)
(What WLM uses for RT goals)

Ended Bundled Transactions
(Not used by WLM for goal management)

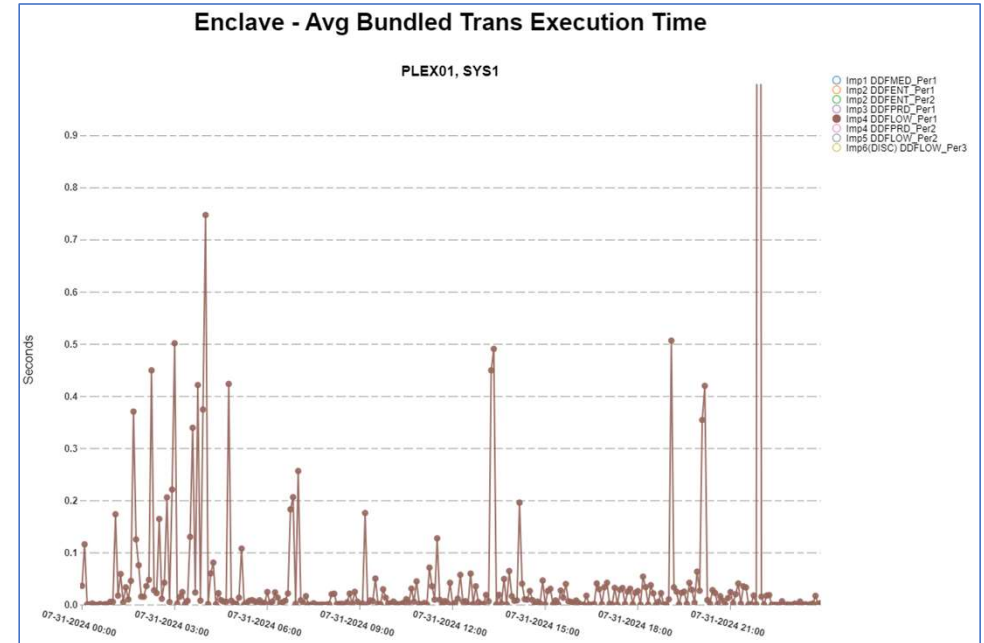
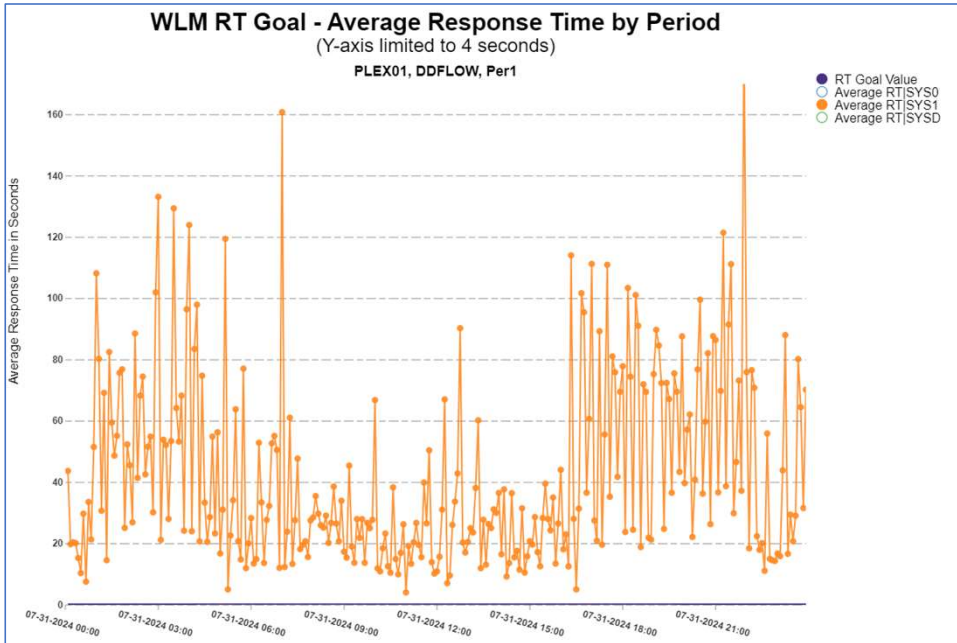


Example: DDFLOW Response Time



Average Response Time of Enclaves
(Used for goal management)

Average Response Time of Bundled Transactions
(Not used for goal management)



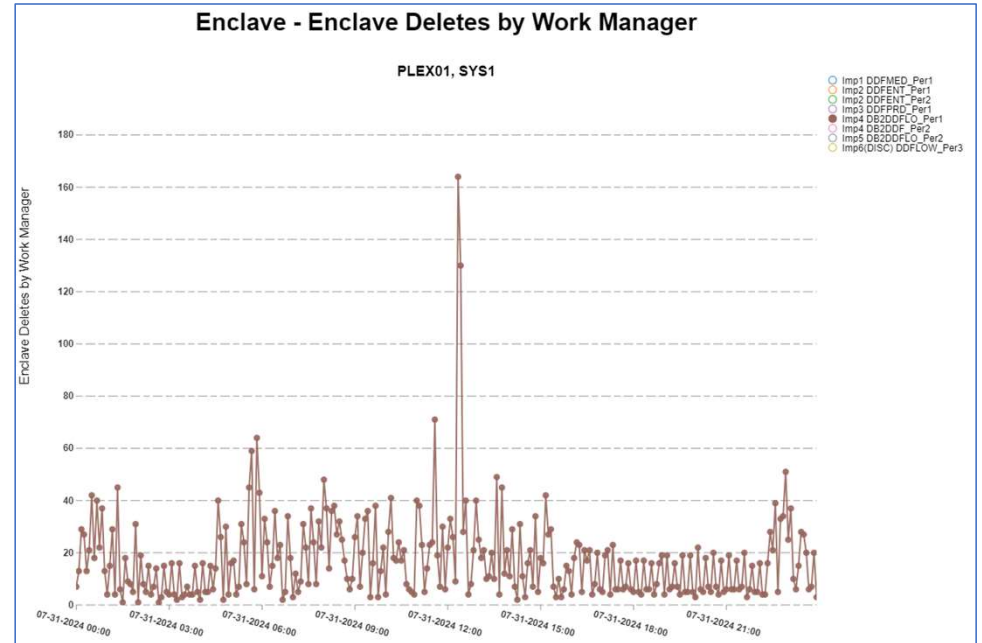
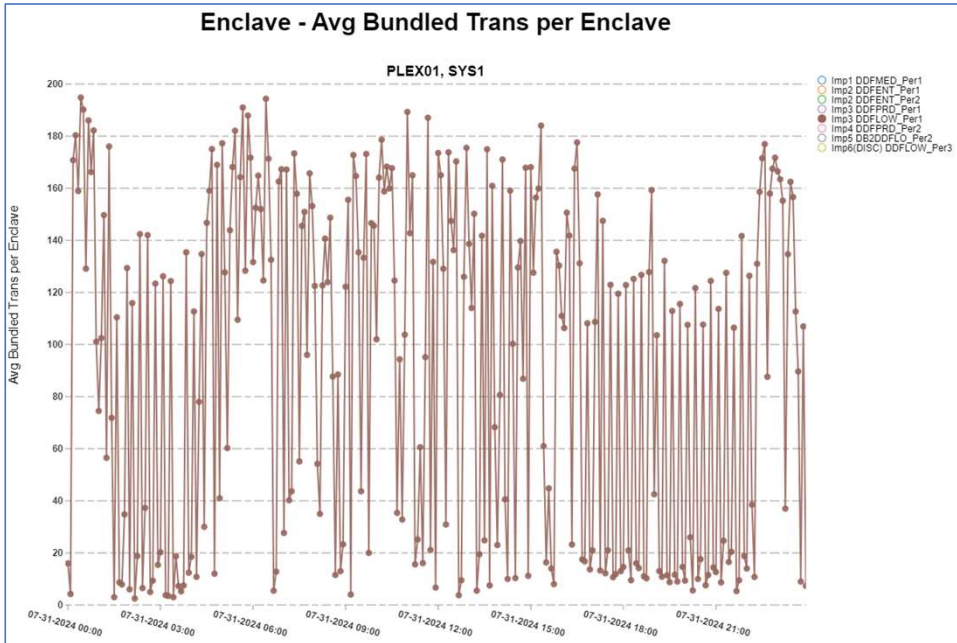
Example: DDFLOW Enclaves for Bundled Transactions



Average Bundled Trans per Enclave
(Note the 200 limit)

Deleted Enclaves for Bundled Trans

(An ended enclave is one WLM ended transaction even though up to 200 DBAT transactions ran in each enclave)



Possible Approaches to DDF Service Classes



- Understand your applications usage of DDF and HiPerf DBATS
 - Are enclaves being used to run bundled transactions?
 - Use the new measurements listed. If non-zero than you are using bundled transactions
- If running HiPerf DBATS, update your WLM setup for DDF
 - Best solution:
 - Work with the DB2 folks to segregate HiPerf DBATs away from other DDF transactions into separate service classes. DBAT transactions should be assigned a velocity goal
 - Mediocre solution:
 - In mixed environments, covert all your DDF service class period goals to velocity
 - Worst solution
 - Do nothing and just use your existing response time goals for the mixed DDF.
 - Adjust goals accordingly
 - Peter's solution
 - Convert the service class running both regular DDF transactions and HiPerf DBATs transactions into a multiple period service class with both periods at the same importance
 - Period 1: Importance x with short duration with a response time goal
 - Period 2: Importance x, with velocity goal



Questions?