Session 11612

z/OS Workload Manager (WLM)

Workload Management of Transactional Workloads

August 2012

Horst Sinram, *z/OS Workload Management*

IBM Germany Research & Development

Email: sinram@de.ibm.com

Trademarks

The following are trademarks of the International Business Machines Corporation in the United States and/or other countries.

AIX*	DS8000*	Language Environment*	SystemPac*	z10
BladeCenter*	FICON*	Parallel Sysplex*	System Storage	z10 BC
DataPower*	HiperSockets	POWER7*	System z	z10 EC
DB2*	Hyperwap	PrintWay	System z9	z/OS*
DFSMS	IBM*	ProductPac*	System z10	zEnterprise
DFSMSdss	IBM eServer	RACF*	System z10 Business Class	zSeries*
DFSMShsm	IBM logo*	REXX	WebSphere*	
DFSMSrmm	ibm.com	RMF	z9*	
DFSORT	Infiniband*	ServerPac*		
DS6000*	InfoPrint			

* Registered trademarks of IBM Corporation

The following are trademarks or registered trademarks of other companies.

InfiniBand is a registered trademark of the InfiniBand Trade Association (IBTA).
Intel is a trademark of the Intel Corporation in the United States and other countries.
Linux is a trademark of Linux Torvalds in the United States, other countries, or both.
Java and all Java-related trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc., in the United States and other countries.
Microsoft, Windows and Windows NT are registered trademarks of Microsoft Corporation.
UNIX is a registered trademark of The Open Group in the United States and other countries.
All other products may be trademarks or registered trademarks of their respective companies.
The Open Group is a registered trademark of The Open Group in the US and other countries.

Notes:

Performance is in Internal Throughput Rate (ITR) ratio based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput improvements equivalent to the performance ratios stated here.

IBM hardware products are manufactured from new parts, or new and serviceable used parts. Regardless, our warranty terms apply.

All customer examples cited or described in this presentation are presented as illustrations of the manner in which some customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics will vary depending on individual customer configurations and conditions.

This publication was produced in the United States. IBM may not offer the products, services or features discussed in this document in other countries, and the information may be subject to change without notice. Consult your local IBM business contact for information on the product or services available in your area.

All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

Information about non-IBM products is obtained from the manufacturers of those products or their published announcements. IBM has not tested those products and cannot confirm the performance, compatibility, or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Prices subject to change without notice. Contact your IBM representative or Business Partner for the most current pricing in your geography.

This presentation and the claims outlined in it were reviewed for compliance with US law. Adaptations of these claims for use in other geographies must be reviewed by the local country counsel for compliance with local laws.

Agenda

- Introduction
- Some Workload Management Definitions and Metrics
- Execution Delay Services and CICS Management Options
- Enclaves and Subsystem Use of Enclaves
- Defining goals for important workloads
- Routing of Work

Workload Management uses a wider definition of “transaction”

- A-C-I-D criteria irrelevant
- A transaction is a work request that…
  - Has defined start and end times
  - Consumes some resources
  - A set of similar transactions is reported on and managed to a certain performance objective
  - May be served by
    - One or more dispatchable units
    - One or more address spaces
    - One or more subsystem types
    - One or more z/OS systems within a Sysplex
- View of what a transaction is may somewhat deviate across subsystems and monitoring products

Overview of transactional workloads management

Workload	TSO/E	Batch and WLM-managed Initiators (JES)	CICS/IMS	DB2	WAS
Transaction scope	TSO command	Job	Work manager (CICS or IMS) transaction	Enclave	Enclave
WLM Subsystem Type	TSO	JES	STC/JES (regions) CICS, IMS	DDF, DB2	CB
Interfaces used	Sysevents	Sysevents, Queue server	Subsystem Work Manager Services, Execution delay services	Enclave services, execution delay services, queue server	Enclave services, queue server, monitoring-only execution delay services
Managed entity	Address space	Address space / initiator	Address spaces executing same transaction mixes	Enclave / dispatchable unit	Enclave / dispatchable unit
Typically…	Short running	Long running	Short or very short running	Short or long running	Short running

Additional enclave exploiters: SAP, IWEB, TCP/IP, LDAP

Workload Management Definitions and Metrics

Performance objective that which WLM should ensure

- **Three options for specifying performance goals:**
  - *Average* Response time goal
    - E.g. *0.5 sec* for an online transaction or *10 min* for a batch service class
    - Good for transactions with similar response times
    - Stable, end-user relevant goal definition
  - *Percentile* response time goal
    - E.g. *80% of transactions to complete in 1 sec* or less
    - Better suited for transaction with inhomogeneous response time distribution
    - Stable, end-user relevant goal definition
  - Execution velocity goal
    - "Execution velocity" is a measure how fast a piece of work is processed
    - Depends on workload, and H/W, S/W configuration

$$\text{Execution Velocity} = \frac{\text{All Using Samples}}{\text{All Using} + \text{ManagedDelays Samples}} \times 100$$

Business importance when not all goals can be met

- **Importance**
  - (most important, fixed DP): SYSTEM, SYSSTC
  - (dynamic DP range managed by WLM): **1 ,2 ,3, 4, 5**
  - (least important) DISCRETIONARY
  - Defines business importance of work, i.e. which goals are most important, and which goals may be sacrificed if not all the work can meet its goal

DP=Dispatch Priority

z/OS Dispatch Priorities

255	FF	SYSTEM
254	FE	SYSSTC
253 … 249	FD … F9	Not Used
248	F8	Small consumer
247 … 204	F7 … CC	Dynamically Managed Dispatch Priorities
203 … 202	CB … CA	Not Used
201 … 192	C9 … C0	Discretionary Mean Time to Wait Algorithm
191	BF	Quiesce

Used by Imp 1 thru 5 work

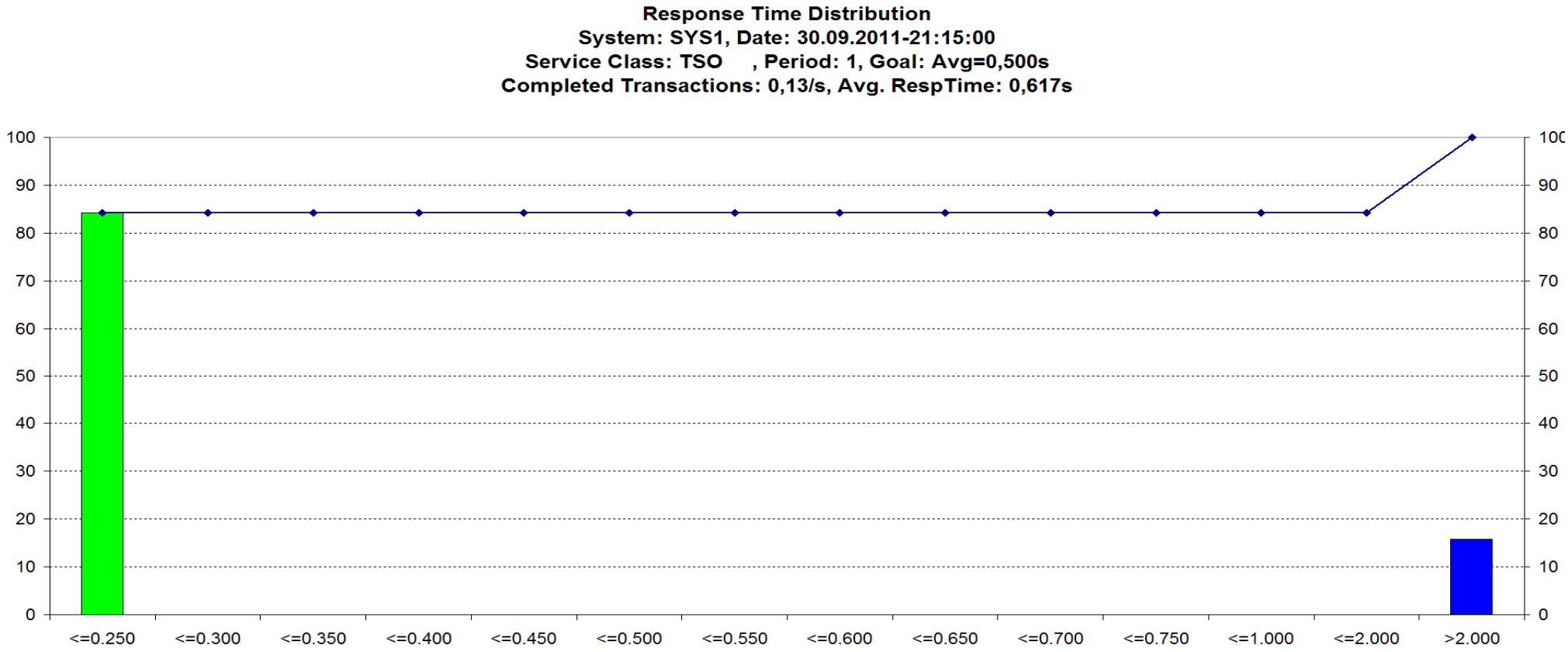
Used by discretionary work

Workload Management Goal Achievement (Performance Index)

- **Goal achievement**
  - Performance Index (PI) is the key metric for goal achievement
  - Defined as $$PerformanceIndex \equiv \frac{ActualPerformance}{DefinedPerformanceGoal}$$
  - Therefore
    - PI<1: Work overachieves goal
    - PI=1: Work achieves goal
    - PI>1: Work misses goal - trigger for WLM to consider some actions
- For velocity and response time goals the PI can be computed easily:
  - Velocity goal $$\mathrm{PI} = \frac{\text{Execution Velocity Goal}}{\text{Achieved Execution Velocity}}$$
  - Average response time goal $$\mathrm{PI} = \frac{\text{Achieved Response Time}}{\text{Response Time Goal}}$$

Average response time goals

- Workload is managed to defined average response times of transaction endings in the period
- Average RT goal achievement may be skewed by small numbers of very long-running transactions
  - But suitable goal type for the first periods of multi-periods service classes



How the PI is computed for percentile goals

- For percentile goals WLM computes 14 discrete response time "buckets"
  - For each transaction ending the count of the respective bucket is incremented
  - Example: Goal = 85% of all transactions completed in 1 sec:

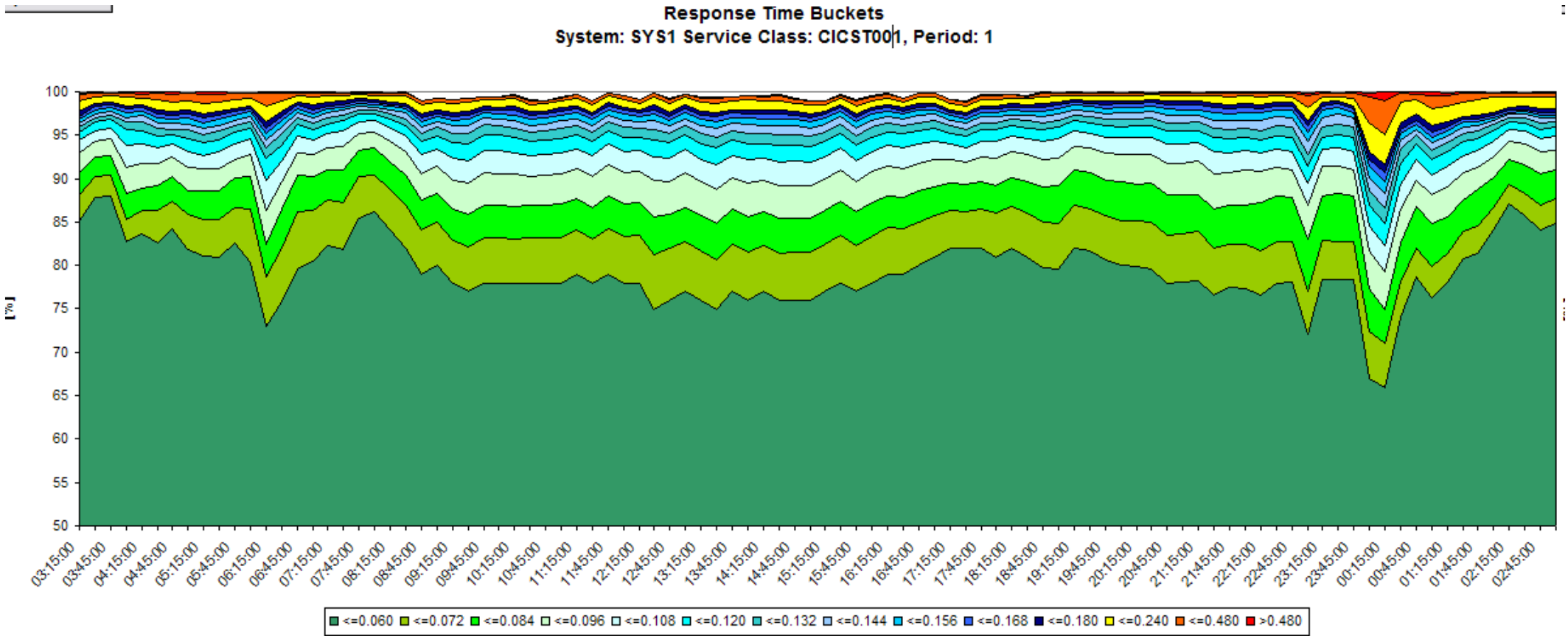
r bucket#	1	2	3	4	5	6	7	8	9	10	11	12	13	14
F_r Fraction of Goal-Response-time	≤0.5	0.6	0.7	0.8	0.9	1.0	1.1	1.2	1.3	1.4	1.5	2.0	4.0	>4.0
Sample percentage of endings	30	10	10	10	10	10	4	6	0	0	0	5	5	5

Accumulated percentage >=85: PI=1.2

Not required to meet goal

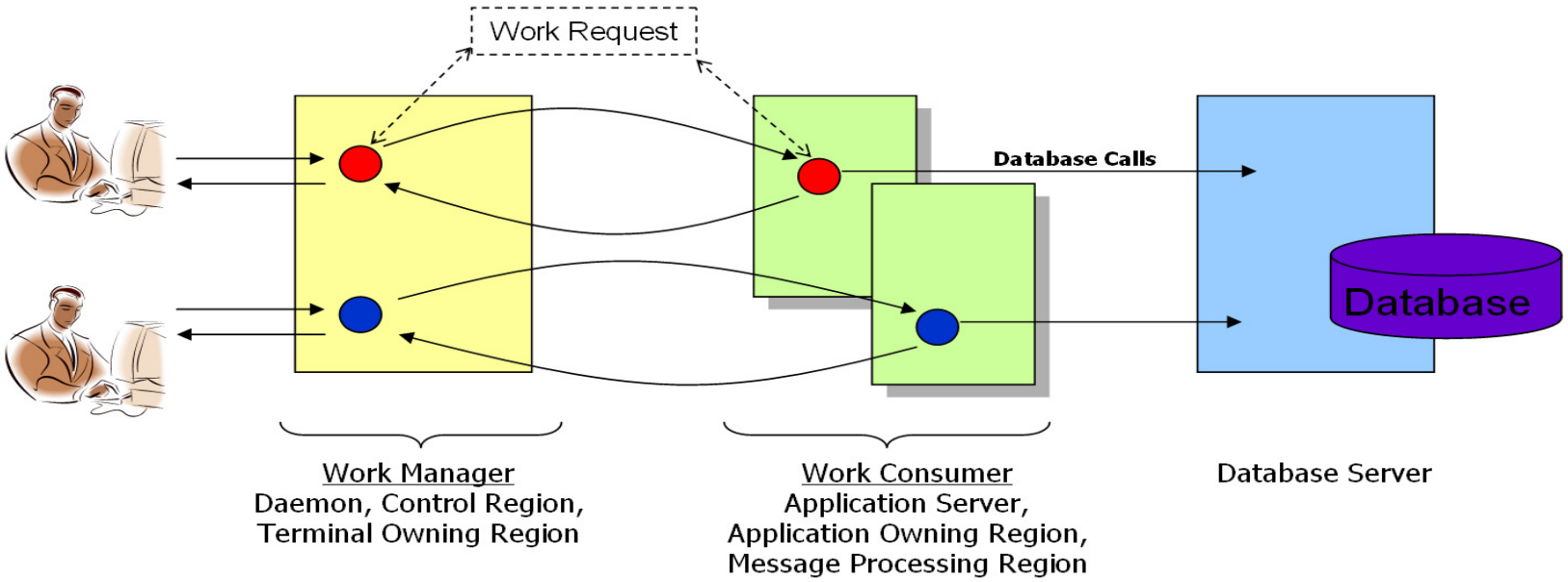
- Smallest possible PI is 0.5 is
- Maximum possible PI is 4.0

Percentile Response Time Distributions



- Percentile RT are the first choice for managing transactions for CICS/IMS, OMVS, TSO, DDF
  - Except for the last period of multi-period service classes (use a velocity goal)
- If the number of transaction endings is very low the PI may appear erratic
  - May not be a real problem, just a low application utilization effect

Work Manager / Consumer concept



- Work request enters in a front end region (Work Manager)
  - Work is classified, transaction bookkeeping tracking begins
  - Usually not very work intensive, but high importance required
  - Used to be not applicable to CICS
- Work manager can pass the work request to other regions (Work Consumer) that process the work request (partially or entirely)
- Work consumers may call database servers for processing of I/O requests
  - DB2, IMS DB, VSAM RLS
- Results are returned to work manager and bookkeeping ends with completion of work request
- Different mechanisms can be uses to implement this model:
  - CICS/IMS: Subsystem work manager services / execution delay services
  - DDF/DB2, WAS: Enclaves
  - Combinations are possible

Agenda

- Introduction
- Some Workload Management Definitions and Metrics
- Execution Delay Services and CICS Management Options
- Enclaves and Subsystem Use of Enclaves
- Defining goals for important workloads
- Routing of Work

WLM CICS Management Overview

Options for managing CICS work

1. CICS managed by response time goals
   - All Regions defined as managed towards TRANSACTION goals
     - Long existing and recommended method
     - Works well for most environments
       - All environments which are not exclusively CICS workload or don't have any problem CICS managed by Region Goals

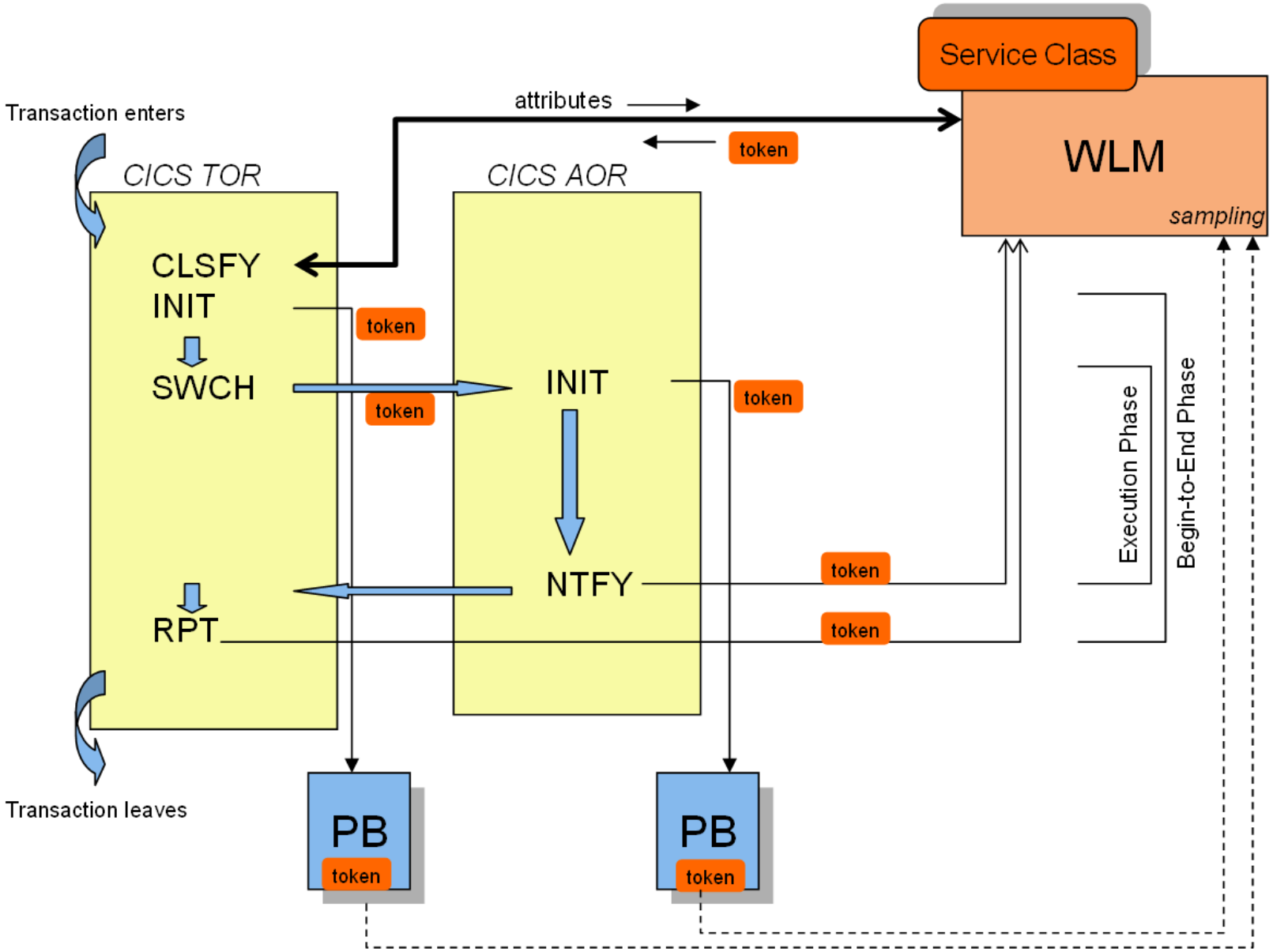
2. If response time goals have not been defined all CICS regions are managed towards REGION goals (exempted from transaction management)
     - Long existing Method
     - Works well for many environments
       - But: Execution velocity goals are more sensitive to hardware and software changes
       - Usually no transaction reporting available

3. CICS managed by BOTH Region and Response Time Goals
   - CICS TORs defined as managed towards BOTH goals
   - CICS AORs defined as managed towards TRANSACTION goals
     - Introduced with OA35428
     - Works well for most environments, too.
     - Also addresses environments where CICS is the main workload and little displaceable capacity exists

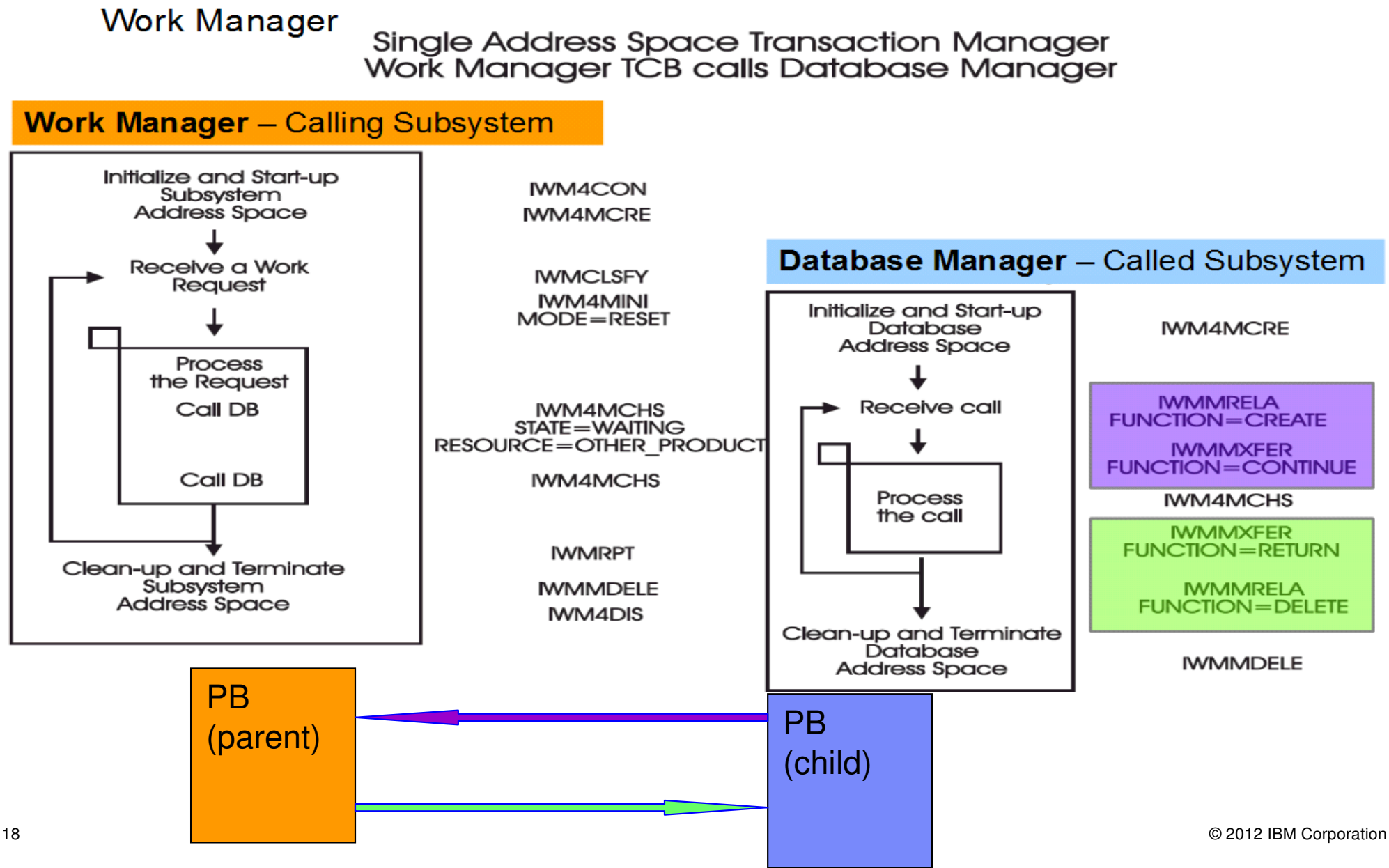
Execution delay monitoring services: The Performance Block

- A control block "Performance Block (PB) plays a key part with execution delay monitoring services
  - Communication vehicle between subsystems and WLM
    - High performance interfaces
  - Subsystems
    - Request to create PB – usually one per thread
    - Control type of PB (e.g. regular, report-only, bufferpool mgmt)
    - Save classification information into the PB
    - Update the PB with work request initiation and completion data
    - Update the PB with address space and dispatchable unit information
    - Update the PB with using/delay states
    - Maintain relationship with parent PB when work request is passed to another address space/component
  - WLM
    - Provides APIs to interface with the PB
    - Samples the PB each 250 ms (can be less)
    - Determines association between service classes and server address spaces: "server topology"
    - Reports on using/delay statistics
    - Reports on response times

Use of execution delay services a CICS TOR/AOR environment



Use of execution delay services an environment with a database manager



Server topologies and internal WLM service classes

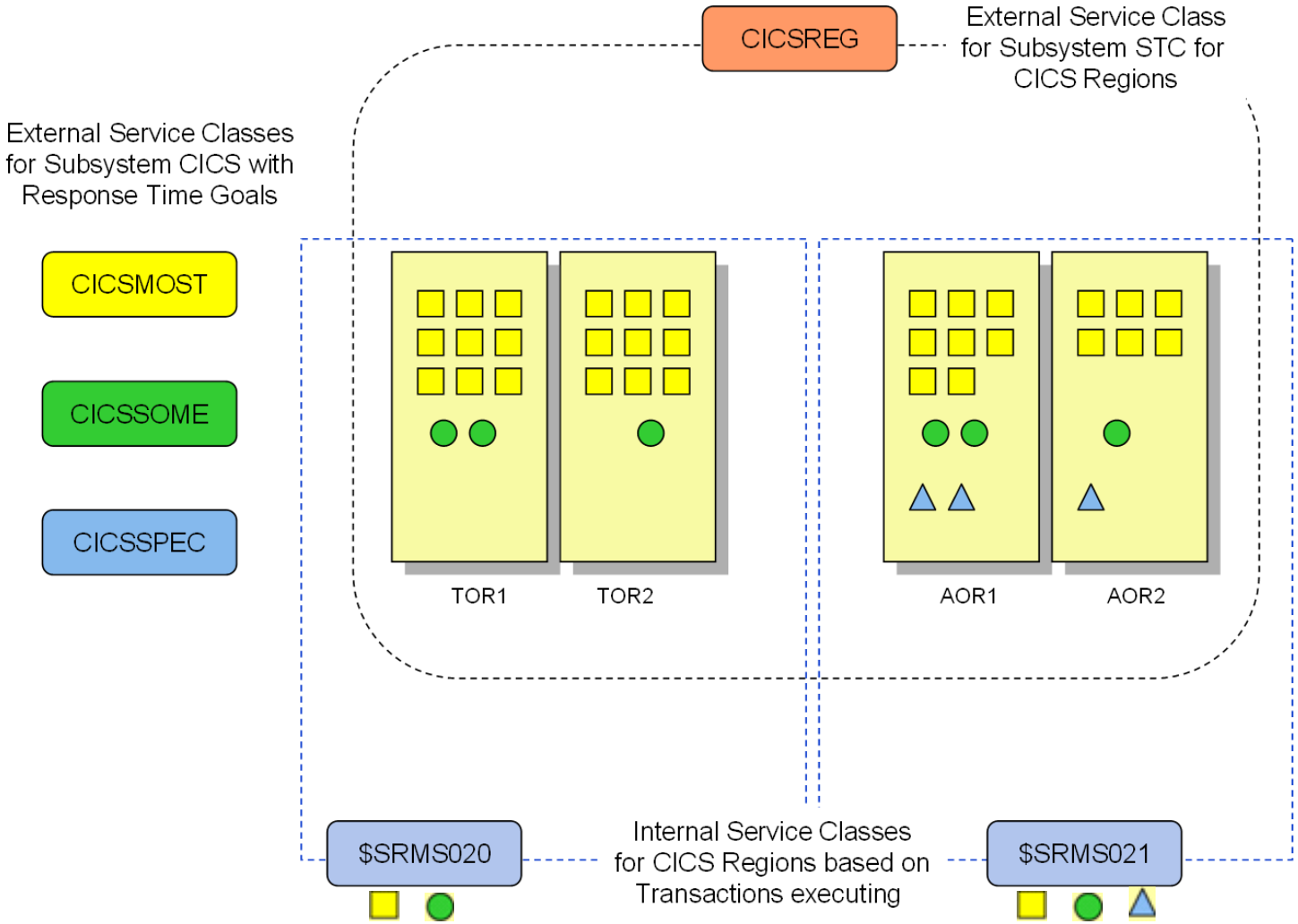
- Usually many server address spaces (TORs, AORs) will exist in a system
  - Server address spaces are classified by itself
  - Executing different transactions that may be classified into different service classes with different (response time) goals
  - WLM needs to manage regions based on the transactions' goals

- For that purpose WLM creates internal service classes

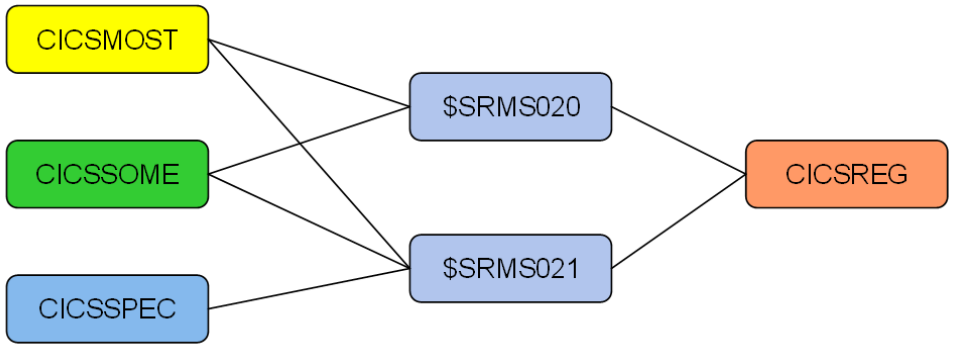
  - Named $SRMSnnn
  - Used to associate sets of server address spaces with sets of external service classes: One internal service class for each set of servers executing the same set of external service classes
  - Need to be continuously reassessed in case transaction mix changes

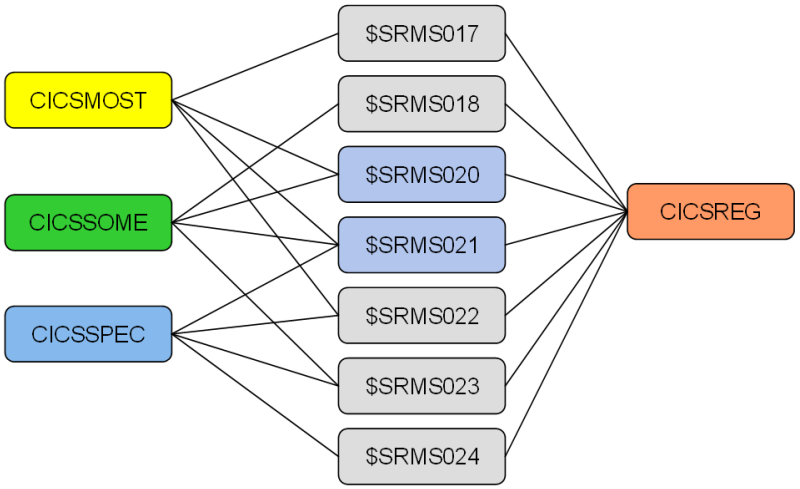
Currently, WLM is not aware of what CICS regions are TORs.

Example with four server AS and three external service classes



Helping server address spaces through internal service classes





- Relationship between external and internal service classes based on previous example
- WLM management:
  - If CICSSPEC goals are missed regions can be helped via $SRMS021
  - If CICSMOST goals are missed
    - Determine which internal SCs contributes most
    - Try helping regions through that internal SC
    - Implicitly helps CICSSOME as well
- However, the number of internal service classes can increase rapidly with the number of external service classes
  - …if transaction topology is unconstrained

Some CICS/IMS management conclusions

- When managing CICS transactions to response times keep the number of external service classes low
  - Ideally no more than two
- Avoid introducing separate external service classes unless these transactions are really significant
  - Transaction service classes for little used transactions can split the sets of servers working on the same relevant transactions
    - These different sets can be managed differently resulting in more heterogeneous response times.
    - A higher number of external service classes usually leads to more volatile topologies.
  - Low consumption internal service classes may be treated as “small consumer” at a DP above other work
- Atypical or long running transactions can potentially be “ignored” by using appropriate percentile goals
- Restricting transaction topology can also help simplifying server topology.

CICS managed by BOTH Region and Response Time Goals Definition

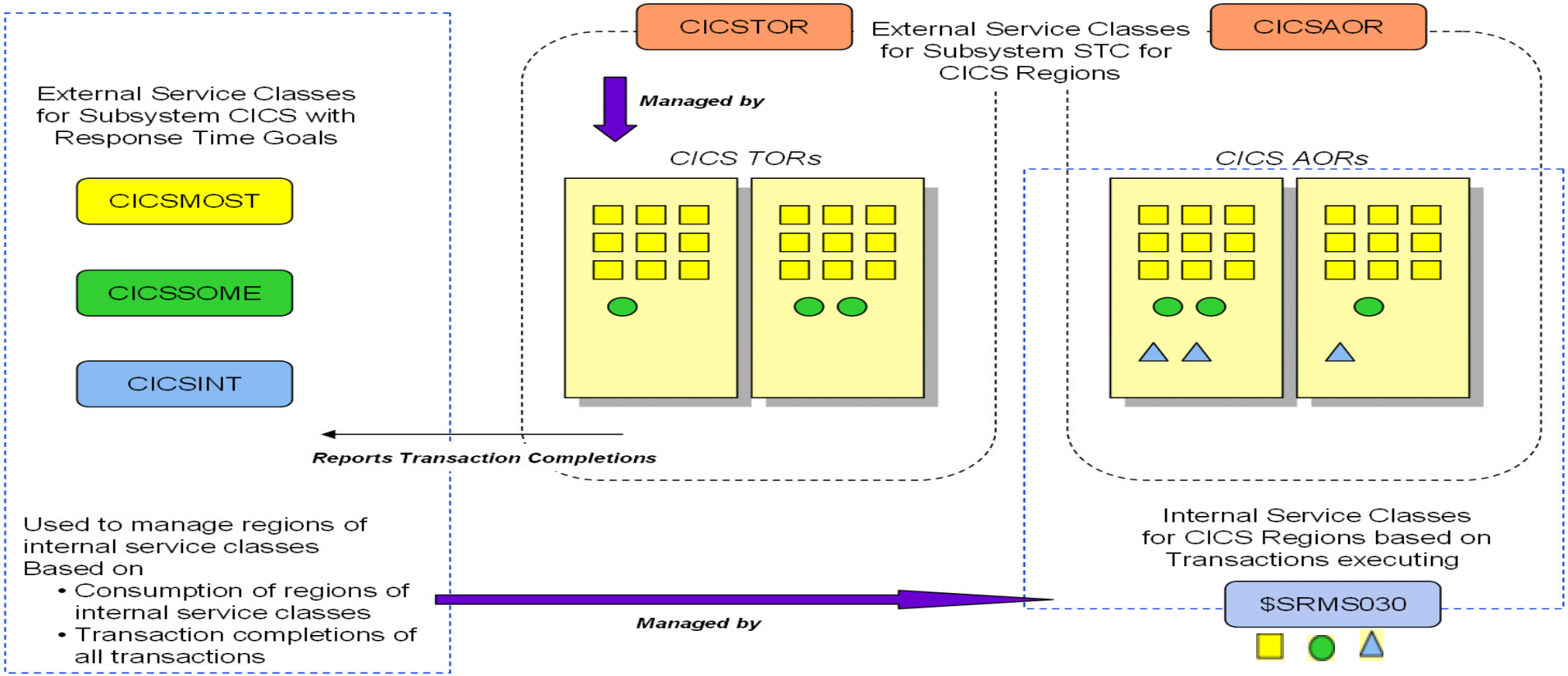
- New "Manage Regions by Goals Of" option in WLM service definition: "BOTH"
  - Available with OA35428 on z/OS R11 and above
  - Use option "BOTH" for TORs
    - Define STC service class for TORs which has a higher importance than the CICS service class with response time goals for the CICS work and AORs
  - Stay with "*Manage Regions Using Goals Of = Transaction*" for AORs.

- Result:
  - WLM will manage the TORs towards the goals of the STC service class
  - And WLM will ensure bookkeeping of transaction completions to the correct CICS response time service class
    - The CICS transactions are managed towards CICS response time goals and the AORs are also managed towards these goals like today

```
  Subsystem-Type  Xref  Notes  Options  Help
 ------------------------------------------------------------------------
                  Modify Rules for the Subsystem Type        Row 1 to 3 of 3
 Command ===> _____________________________________________ Scroll ===> PAGE

 Subsystem Type . : JES          Fold qualifier names?   Y  (Y or N)
 Description  . . . Batch Work ____________________________

 Action codes:   A=After     C=Copy        M=Move     I=Insert rule
                 B=Before    D=Delete row  R=Repeat   IS=Insert Sub-rule
                                                        <=== More
           --------Qualifier--------            Storage    Manage Region
 Action    Type        Name      Start           Critical   Using Goals Of

  ____   1  TN          CICSTOR* ___              NO         BOTH
  ____   1  TN          CICSAOR* ___              NO         TRANSACTION
  ____   1  TN          CICS*    ___              NO         TRANSACTION
******************************* BOTTOM OF DATA ******************************
```

CICS managed by BOTH Region and Response Time Goals

Structure of Service Classes

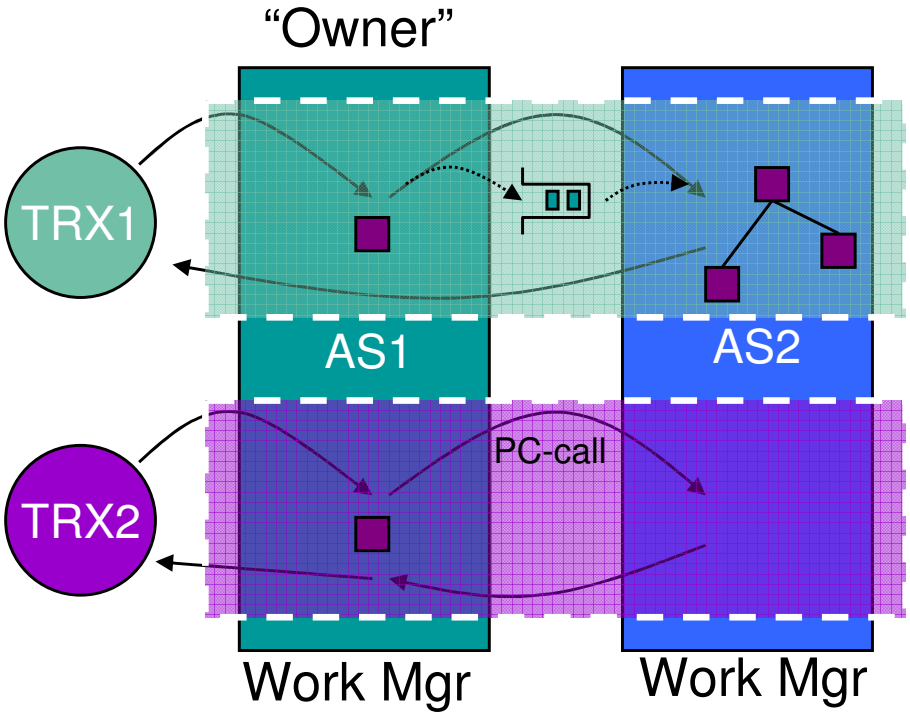


- TORs are now managed towards the goal of the service class CICSTOR
  - They still report their transaction completions for management
- AORs are still managed towards the goals of the CICS service classes and the consumption of the internal service class for the region
- Recommendation: CICSTOR should be defined at a higher importance than the CICS service classes

Agenda

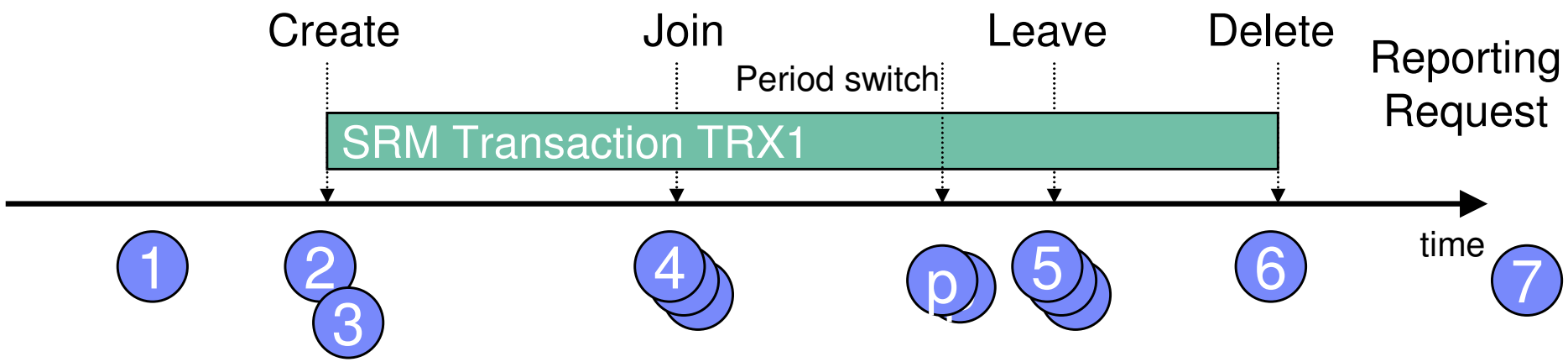
- Introduction
- Some Workload Management Definitions and Metrics
- Execution Delay Services and CICS Management Options
- Enclaves and Subsystem Use of Enclaves
- Defining goals for important workloads
- Routing of Work

What is an enclave?



- Unrelated to what other components call enclave ☺
- A logical construct representing a "business unit of work"
  - Groups one or more units of work running in the same or multiple address spaces
    - Preemptible SRBs
    - Tasks (TCBs)
- Enclave dispatch priority is managed by WLM
- Enclaves do not own storage
- Owned by home address space at time of creation
- Owner can own multiple enclaves at a time

Typical enclave lifecycle



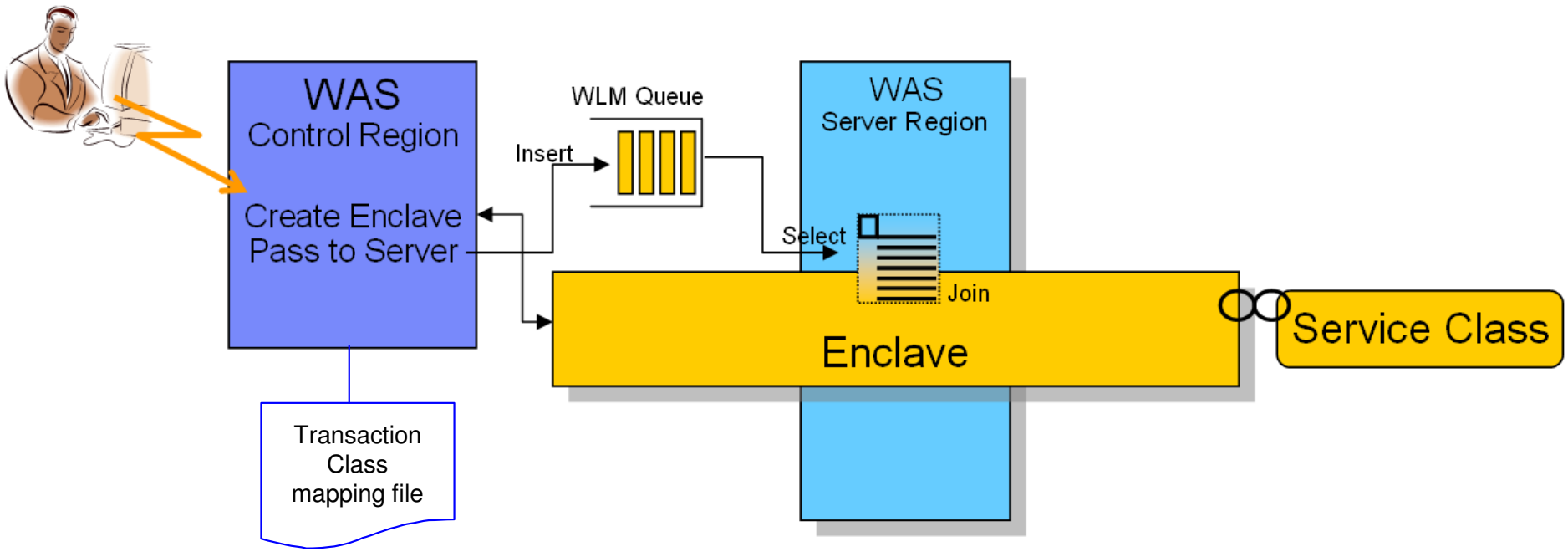
- Exploiting subsystems (DB2, SAP,Websphere)
  - 1 ➢register as work managers to WLM
  - 2 3 ➢Upon arrival of work requests it creates an enclaves and classifies it. Either a preemptible SRB is scheduled to execute the work request or the enclave execution information is passed to a server region
  - 4 ➢The server region dispatchable unit picks up (“joins”) the request.
  - 5 ➢Server tasks leave the enclave.
  - 6 ➢Work manager deletes enclave.

- WLM / SRM
  - p ➢Can directly manage the enclaves independently from the address spaces. Allows giving different execution units of the same address space different dispatch priorities depending on the service class they belong to. Transaction can also undergo period switch**.**
    Manages server storage to meet enclave goals.
  - 7 ➢Provides reports on the enclaves.

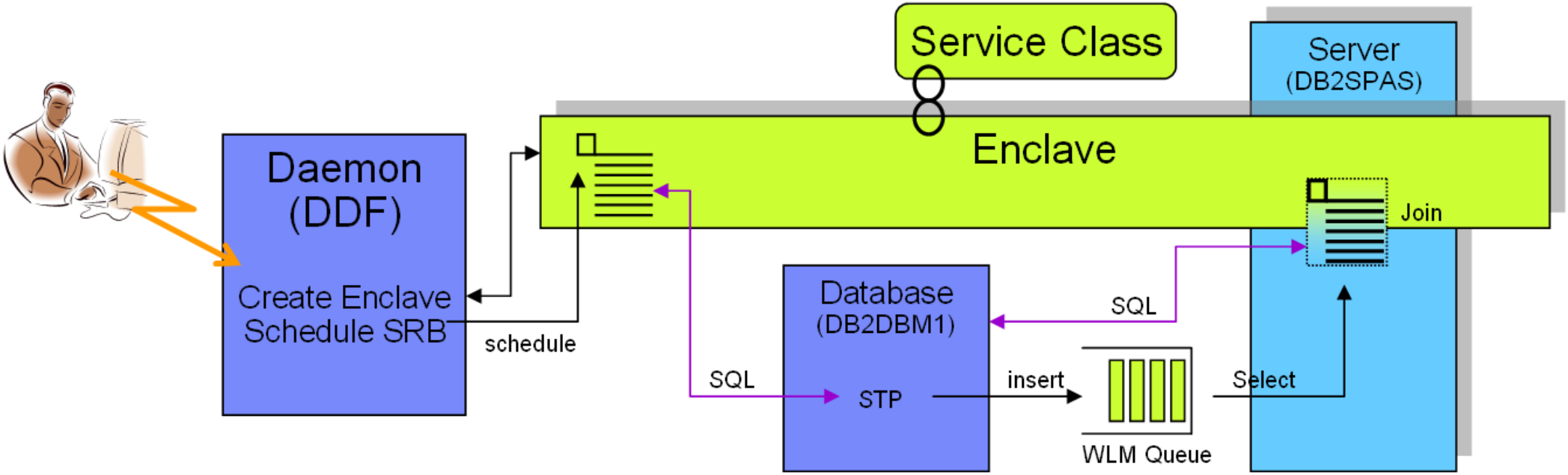
Types of enclaves

Enclave type	Characteristics
Independent	➢Represent a new unit of work in the system. Does not extend a running transaction. ➢Need to be classified through classification rules in WLM Service Definition ➢Executable Units (TCBs and SRBs) join the enclave during execution ▪While joined they are managed towards the goals of the service class into which the enclave has been classified to ➢Allow to manage units of work across multiple address spaces and therefore are closest to represent customer transactions on MVS from a performance management point
Dependent	➢Are a continuation to an existing process on MVS. So the continuation is always tied to the address space under which it is created. Extends creating AS' transaction. ➢Do not require separate classification: Inherit the classification from the address space
Work-dependent	➢ Extension to an independent, dependent, or other work-dependent enclave. **Extends the transaction the creating enclave**. ➢ Run like an independent enclave when created by non-enclave work. Allows control of zIIP offload by entitled products.
Foreign	➢ Are a continuation of a unit of work (enclave) from another system in the same sysplex.

Enclave Management with WebSphere



Enclave Management with DDF



WebSphere Classification Approach

- Control region
  - Under the STC subsystem: High importance, high velocity

- Servant/adjunct
  - Appropriate velocity
  - Importance: Weigh fast (re)start need vs. impact of CPU demand
  - IEAOPT ManageNonEnclaveWord=No/Yes controls how work outside enclaves is being managed (garbage collection, common service routines)
    - With default of NO such work is not fully managed by WLM
    - With YES it is managed to the region's goal

Agenda

- Introduction
- Some Workload Management Definitions and Metrics
- Execution Delay Services and CICS Management Options
- Enclaves and Subsystem Use of Enclaves
- Defining goals for important workloads
- Routing of Work

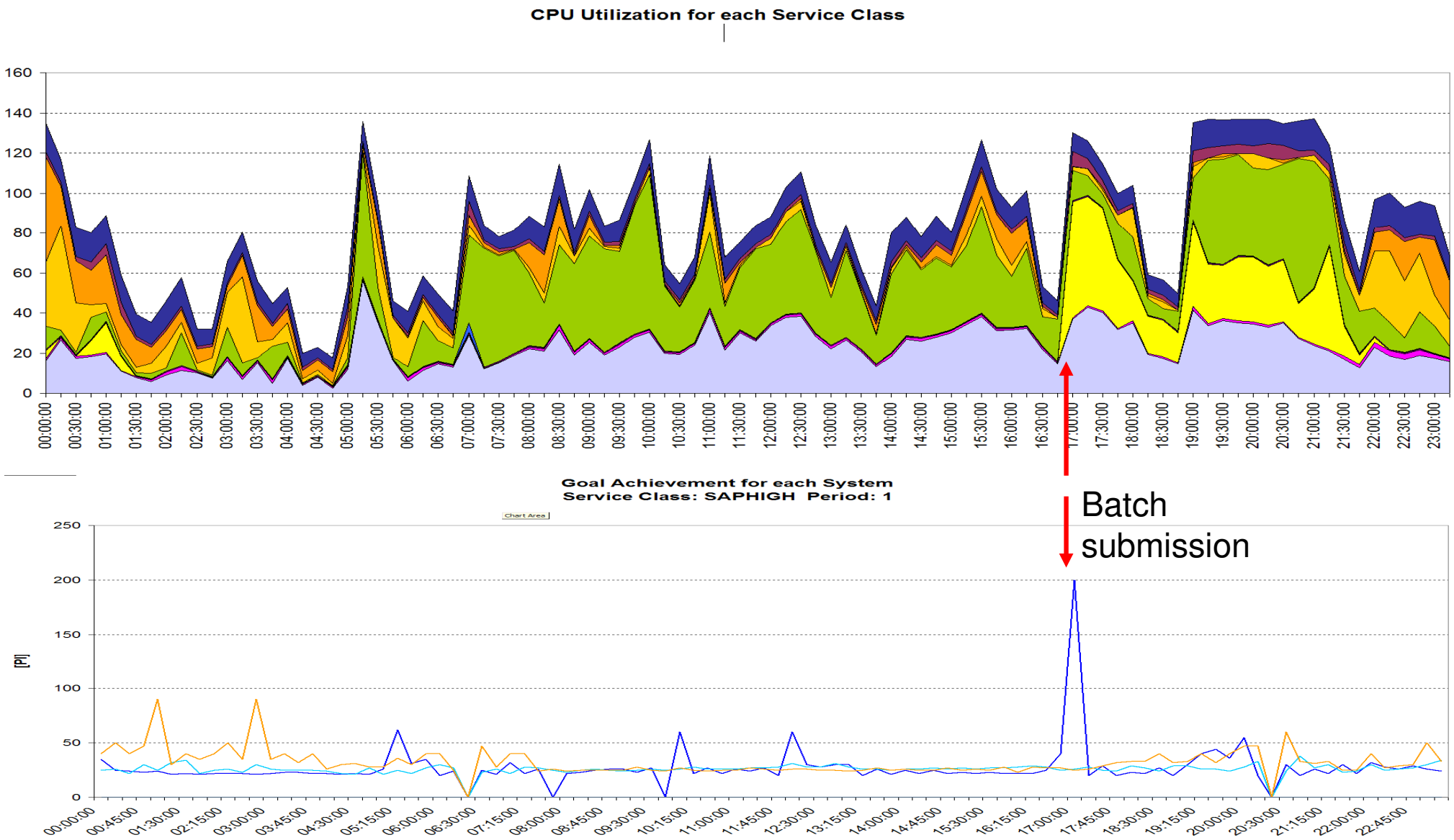
Choosing goals for transactional workloads

- A goal –regardless of what type- should be
  - Achievable
    - Base goal based on peak times (application/system/CPC level)
  - Important goals should be challenging
    - Ideally, target a PI of 1.2 at peak duration

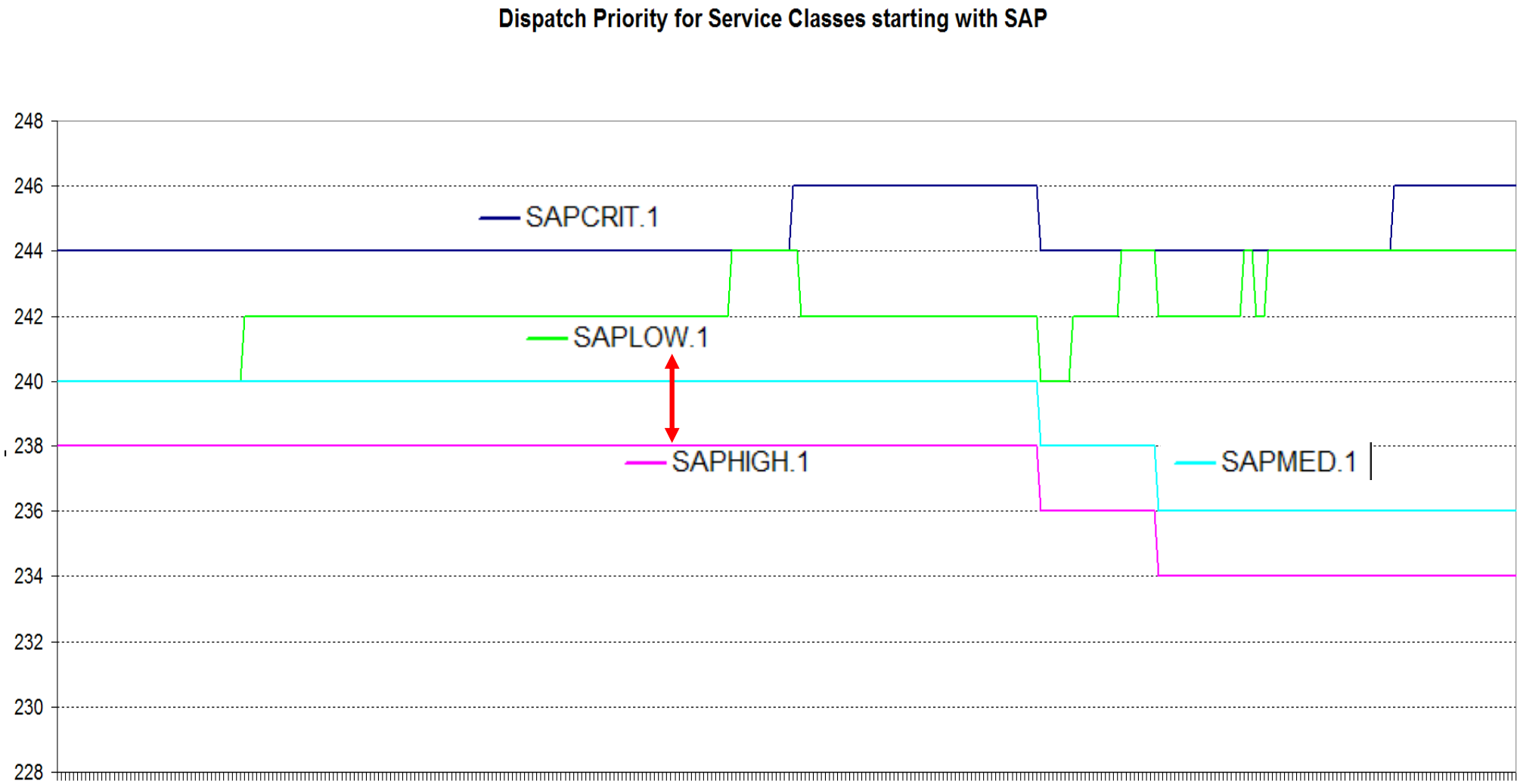
- Common problems are goals that are far too relaxed
  - May not show up as a problem as long as
    - Overall utilization moderate, and
    - No other workloads are injected into the system
  - Frequently seen symptom are impacted goals upon batch submits.
    - Dispatch priority not elevated by WLM because goals are still being for quite some time.

- Following examples show a SAP workload but the problem equally applies to other workloads

Batch submission impacting high importance workloads



Dispatch priority of SAPHIGH too low (below SAPLOW) because the goal far too relaxed



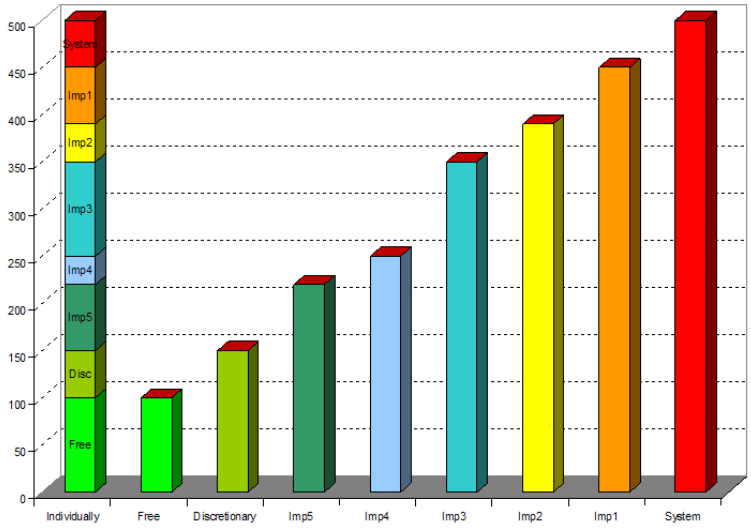
Agenda

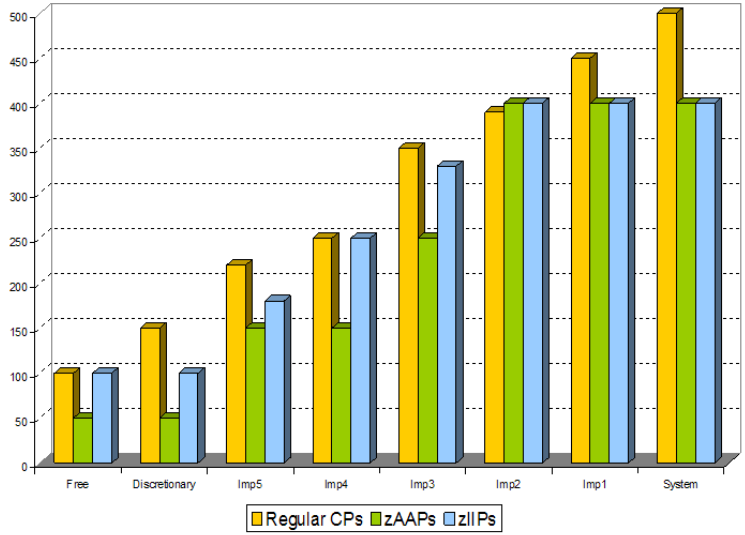
- Introduction
- Some Workload Management Definitions and Metrics
- Execution Delay Services and CICS Management Options
- Enclaves and Subsystem Use of Enclaves
- Defining goals for important workloads
- Routing of Work

Sysplex Routing Services

- Sysplex routing services allow for a proper load balancing of transactions across all systems in a sysplex. They provide support
  - to understand the free or displaceable capacity at each importance level for every system
  - Let exploiting subsystems decide which system and server a work request should be routed to
- Capacity-based routing alone is insufficient. Also need to consider:
  - Server-specific performance data: Performance index, queue time
  - Health state
- For exploiters of specialty processors the processor-type specific capacities are available
  - Optionally include cost factor for different processor types
- Routing is ´done by the subsystem, such as CTG, DB2 Connect, Sysplex Distributor – not WLM itself
  - WLM provides set of services for registered servers to report health state and obtain routing weights

IWMWSYSQ is a simple interface to understand free/displaceable capacity





Server-specific Routing (DB2)

SYS	Avail Cap	**Orig. Server weight**	PI	WLM weight
SYS1	110	**18**	1.3	14
SYS2	100	**16**	0.8	16
SYS3	95	**15**	1.0	15
SYS4	95	**15**	2.0	8
SUM		**64**		53

Notes:

1. With less than perfect health states the weights would also to modified to reflect health state
2. The number of connections is usually not proportional to the WLM weights

धन्यवाद
Hindi

多謝
Traditional Chinese

ขอบคุณ
Thai

Спасибо
Russian

Gracias
Spanish

Thank You
English

Obrigado
Brazilian Portuguese

شكراً
Arabic

多谢
Simplified Chinese

Danke
German

Bedankt
Dutch

Grazie
Italian

Merci
French

நன்றி
Tamil

ありがとうございました
Japanese

감사합니다
Korean