



CPU Critical: A Modern Revisit of a Classic WLM Option



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



Creators of Pivotor®

Peter Enrico & Scott Chapman

Enterprise Performance Strategies, Inc.

performance.questions@epstrategies.com



● **WLM and CPU Critical Control**

Buckle up, folks! As of z/OS 3.1, all WLM importance level 1 service class periods will be treated as CPU critical. Why buckle up? Because this change is one of the rare, incompatible changes made by IBM, and has the potential to negatively affect the WLM management of your environment. Luckily, there is an option to disable this incompatible change. However, there is a parameter to optionally disable this setting, if you feel the incompatible default may negatively impact the way WLM manages your workloads.

During this webinar, Scott Chapman and Peter Enrico will provide a review of the WLM CPU critical control and discuss the changes made to z/OS 3.1 that, by default, causes all importance level 1 service class periods to be treated as CPU critical. They will also feature some Pivotor[®] reports to help you determine whether you will be positively or negatively affected by this change.

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
 - May 12 – May 16, 2025 (4 days)
- Parallel Sysplex and z/OS Performance Tuning
 - July 15-16, 2025 (2 days)
- Essential z/OS Performance Tuning
 - September 22-26, 2025 (4 days)
- 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 Spring / Summer 2025 webinars are as follows:
 - *Back to basics - Processor Consumption Analysis*
 - *WLM and CPU Critical Control*
 - *Pivotor Pointers*
 - *Back to Basics - Evaluating Latent Demand*
 - *Understanding SMF 98 Locking Measurements*
 - *Overseeing z/OS Performance Management With Your Outsourcer*
 - *Standard Measurements when Monitoring Transactions*
 - *Processor Comparison Discussion*
 - *z/OS Performance Management in an AI World*
 - *Understanding z/Architecture Processor Topologies*
 - *SMF 99 WLM Decision Making Traces*
 - *Understanding SMF 98 Address Space Consumption Measurements*
- If you want a free cursory review of your environment, let us know!
 - We're always happy to process a day's worth of data and show you the results
 - See also: <http://pivotor.com/cursoryReview.html>

Like what you see?



- If you want a free cursory review of your environment, let us know!
 - We're always happy to process a day's worth of data and show you the results
 - See also: <http://pivotor.com/cursoryReview.html>
- We also have a **free** Pivotor offering available as well
 - 1 System, SMF 70-72 only, 7 Day rolling retention
 - That still encompasses over 100 reports!
- We offer free bi-monthly z/OS performance educational Webinars
 - Sign our guestbook to get invited
 - Or send an email to
 - Peter.Enrico@EPStrategies.com



Presentation Overview



- z/OS 3.1 is defaulting all importance 1 workloads to “Implicit Long-Term CPU Protection”, aka “CPU Critical”
 - This will likely lead to dispatching priority changes on most systems.
 - Some environments will be significantly impacted
- This presentation
 - Refresher of z/OS CPU Dispatching and Dispatching Priorities
 - Refresher of WLM CPU controls and algorithms
 - Refresher of CPU Critical Control
 - Overview of new Implicit CPU Protection in z/OS 3.1

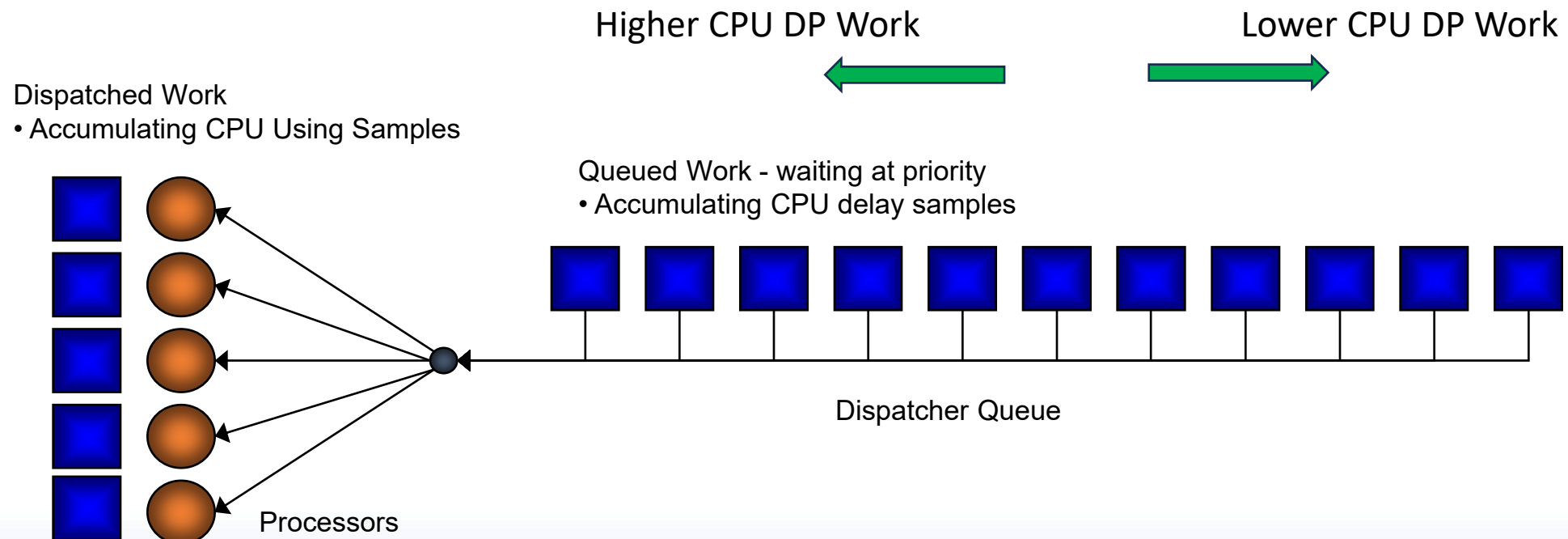


Refresher : CPU Dispatching and Dispatching Priorities

Defining CPU Dispatching Priorities



- Dispatching priorities determine the order in which operations are executed by the CPU
 - All address spaces and enclaves are assigned a CPU dispatching priority
 - The dispatching priority order is determined by WLM based on WLM goals
 - Work queued to use the CPU is placed onto the dispatching queue in CPU dispatching priority order

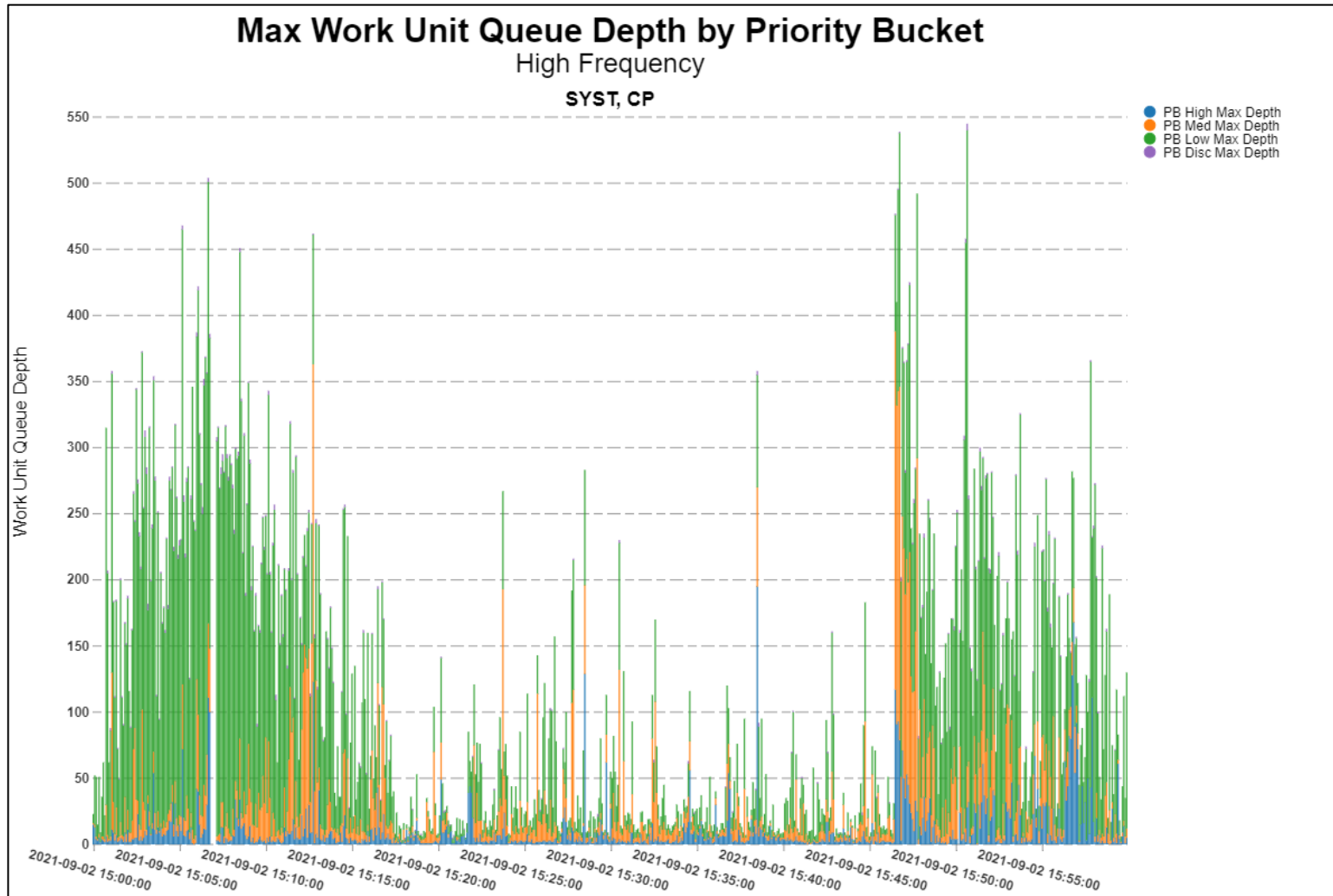


Reason 1: Why we care about CPU DPs



- Reason 1: Prioritization of CPU access to each workload
 - Example: Higher importance work should run at higher CPU DPs
- Although not a hard and fast rule, to meet goals, higher importance work tends to receive higher CPU DPs from lower importance work
- Relative importance does not translate to relative CPU DPs
 - A higher importance goal could have a lower CPU dispatch priority than a lower importance goal
 - CPU Critical control does influence this
 - Says lower importance work will never have same or higher DP as work identified as CPU critical

Use SMF 98 to look at dispatch queue depths



Priority bucket statistics

(1=High, 2=Med, 3=Low, 4=Discretionary)

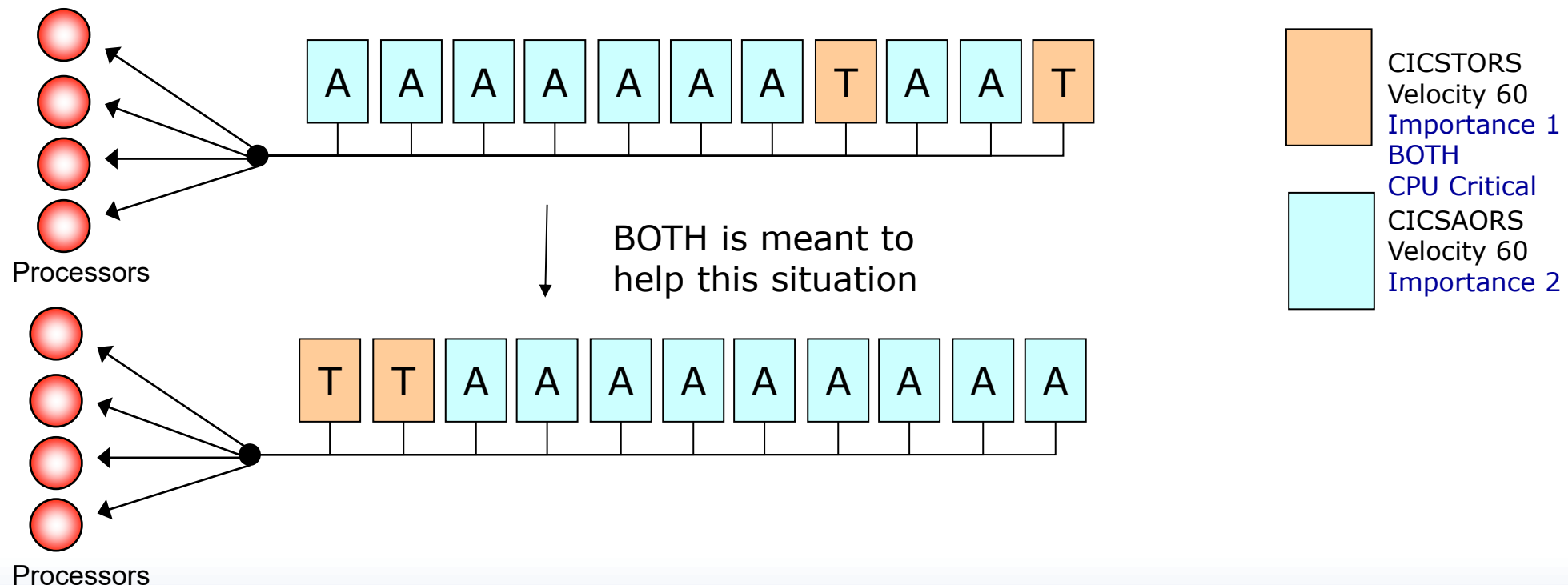
This chart is related to the previous chart, but for CP engines rather than zIIP engines.

We see that although the dispatching queues are longer. The displaceable work is lower importance. Thus, crossover of higher importance will displace this lower importance work.

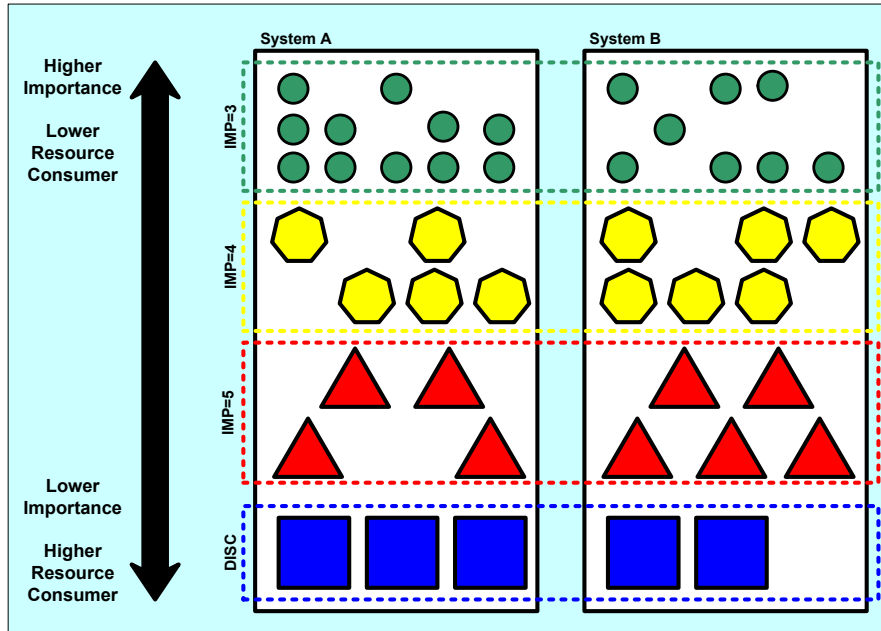
Reason 2: Why we care about CPU DPs



- Reason 2: To assist when there are feeder effects
 - Example: Certain workloads feed other workloads to progress work
 - IRLM -> DB2 (DBM1,MSTR,DIST) -> CICS TOR -> CICS AOR

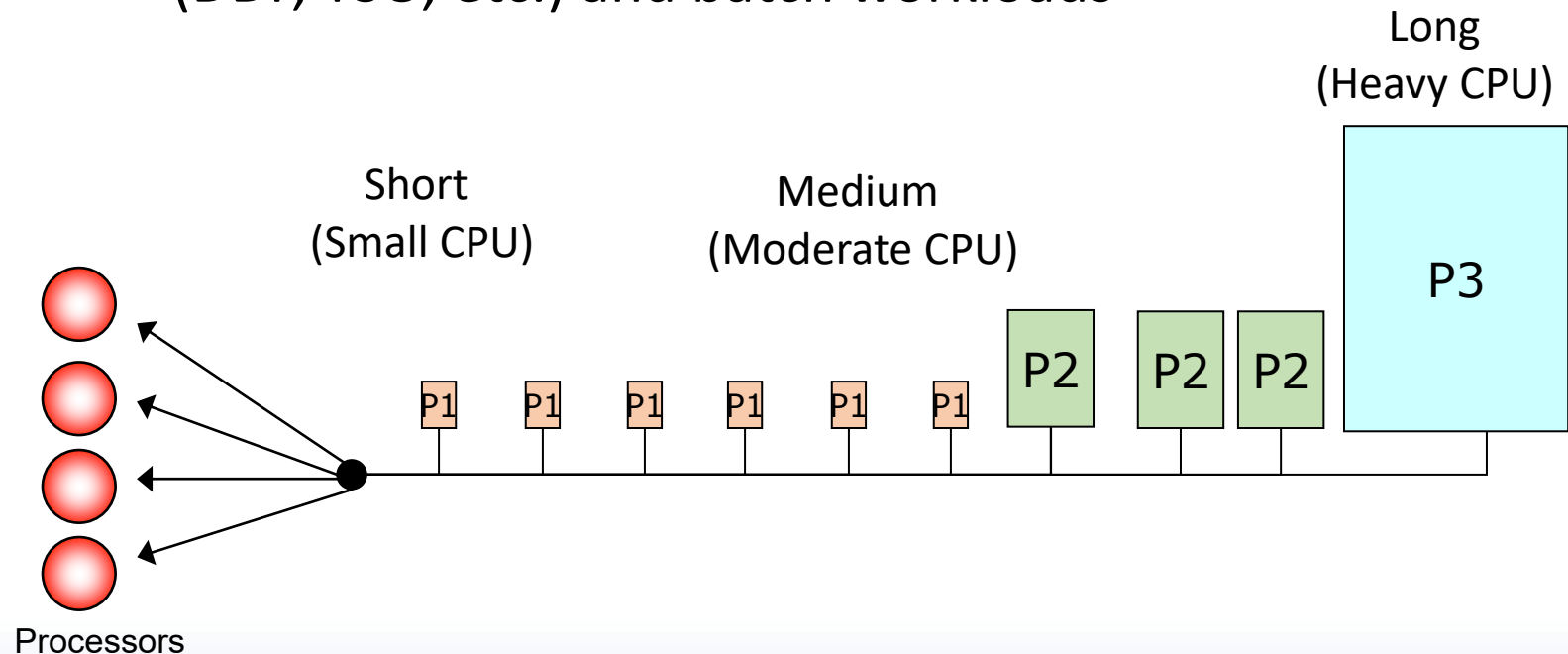


Reason 3: Why we care about CPU DPs



Reason 3: Workload efficiencies

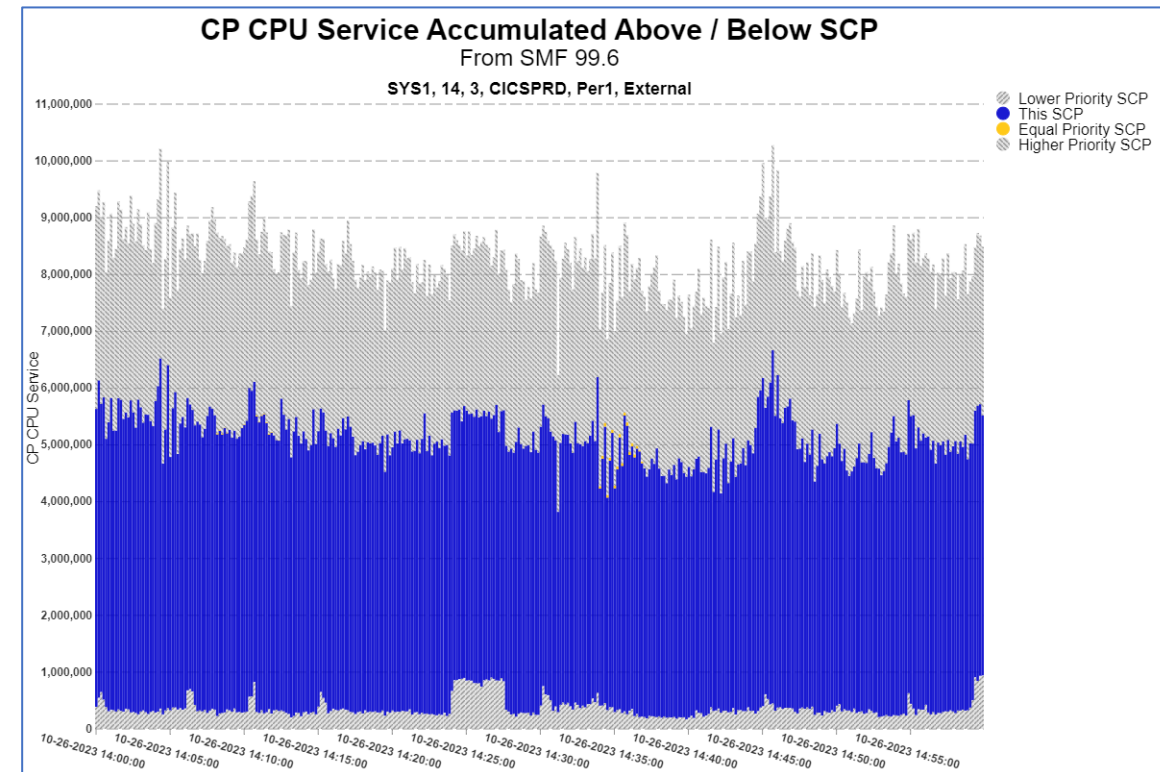
- Example: Get the shorter work (light CPU consuming work) on the CPU first to get it 'in and out' as quickly as possible
- This is extraordinarily effective with interactive (DDF, TSO, etc.) and batch workloads



Reason 4: Why we care about CPU DPs



- Reason 4: Managing goals and optimize processors by addressing processor delays
 - Helps address latent demand
 - Correct CPU dispatching priority helps manage and alleviate latent demand and processor delays
 - Example: Over initiation of batch, many more CICS regions than number of CPUs
- Also, processor cache efficiencies
 - Interactive workloads tend to use processor caches less efficient
 - Batch workloads tend to use processor caches more efficiently

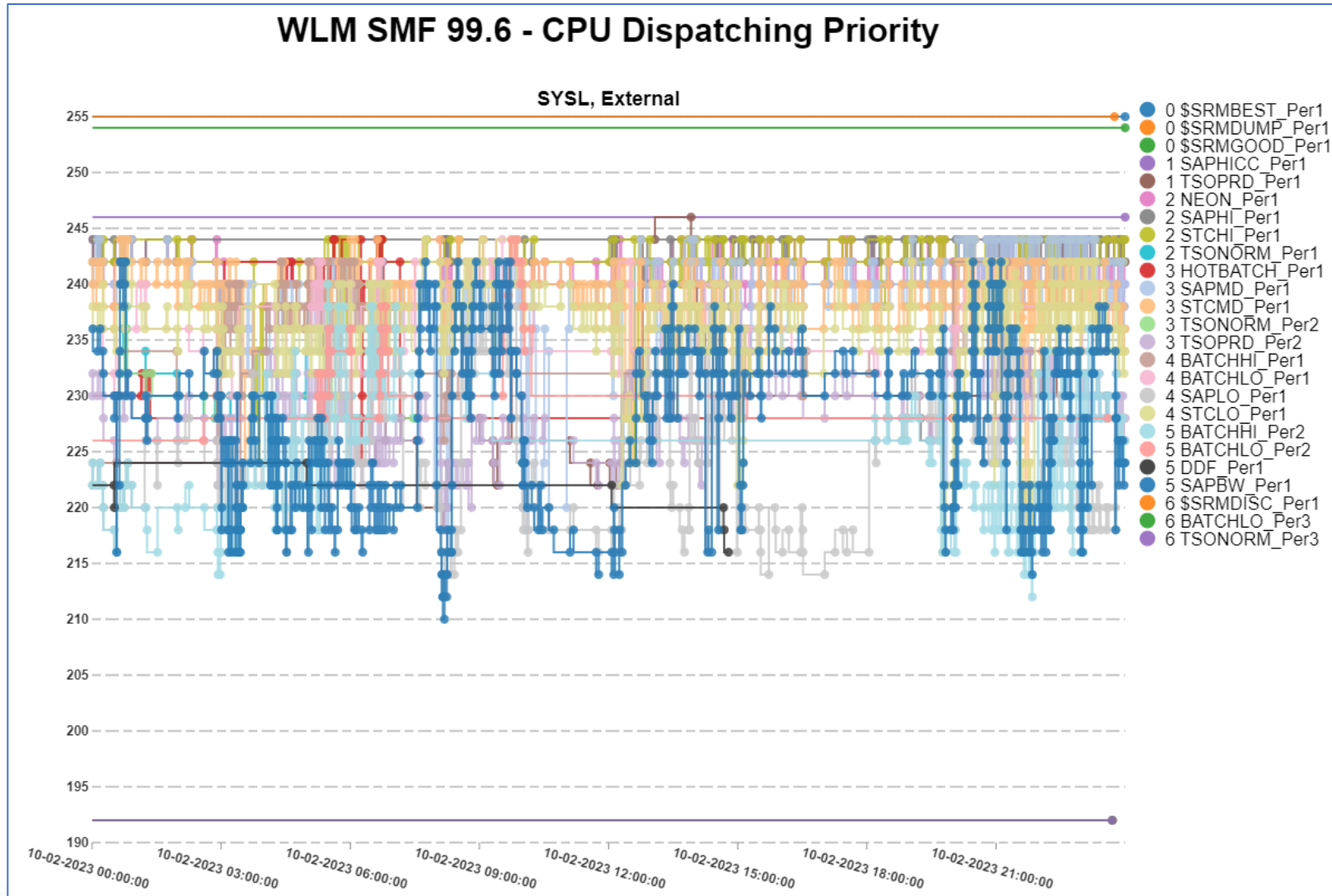


Assigning CPU Dispatching Priorities



- WLM sets dispatching priority for service class periods.
- All address spaces in a service class period have the same base dispatching priority
 - Note: there are 'internal periods' and 'external periods'
 - There is also something called promotion (discussed later)
- Multiple service class periods may have the same base dispatching priority.
- Unbunching
 - There is a concept in the WLM algorithms called 'unbunching'
 - After a dispatching priority change, service class periods may be remapped to different dispatching priorities such that there is an unoccupied priority between each occupied priority.

SMF 99.6 CPU Dispatching Priority – Every 10 Seconds



WLM service class
period CPU dispatching
priority for 24 hours.
Shows the CPU
dispatching priority
order every 10 seconds

CPU Promotion: Another reason for changing CPU DPs:



- WLM briefly ‘promotes’ individual work units to higher CPU dispatching priorities to boost the access to the CPU for very short periods of time
 - Concept:
 - If a unit of work is regularly demanding CPU and possibly holding a resource, then WLM may briefly promote the unit of work to a higher CPU DP to allow it to run for a short period of time in hopes of the work unit releasing the resource
- Reasons for promotion:
 - ENQ - Promoted by enqueue management because the work held a resource that other work needed.
 - BLK - Promoted to help blocked workloads
 - LCK - Promoted to shorten the lock hold time of a local suspend lock held by the work unit
 - SUP - Promoted by the z/OS supervisor to a higher dispatching priority than assigned by WLM
 - CRM - Promoted by chronic resource contention management because the work held a resource that other work needed

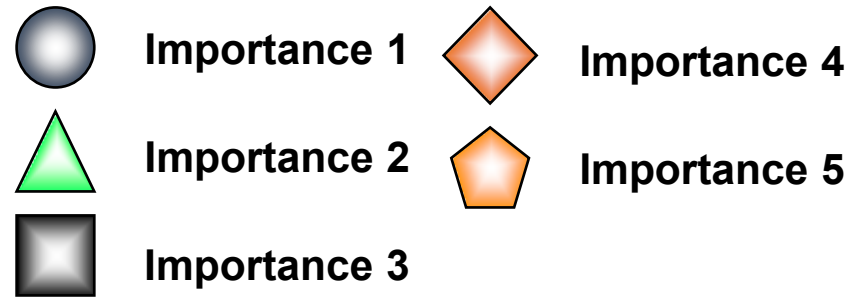
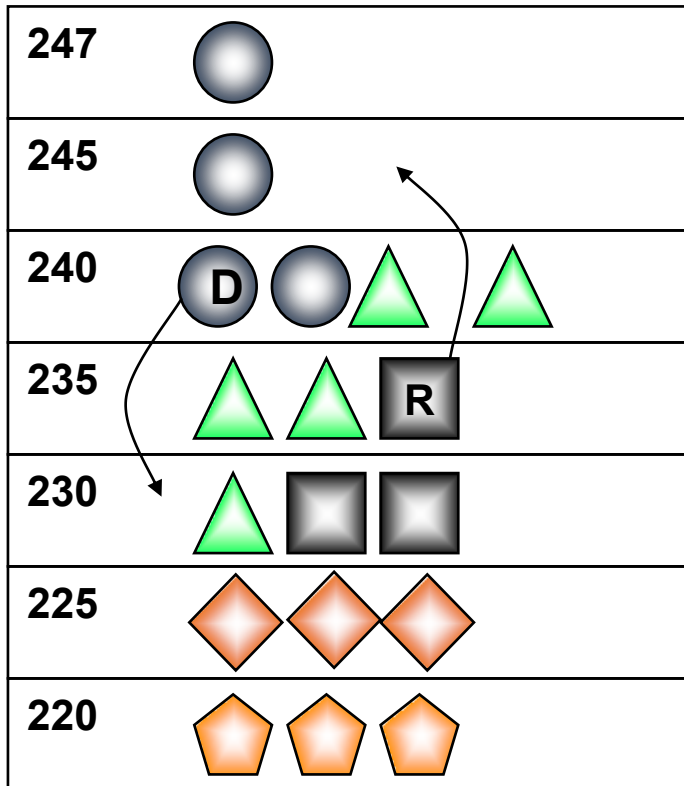


WLM CPU Critical Control

Overview of WLM CPU Critical Control



- Because some installations are concerned that WLM will not react fast enough for high priority work, WLM has a 'CPU Critical Control'



- With well set predictable goals, DPs tend to be ordered by importance
- If work is missing its goal WLM may decide to adjust its DP equal or above a higher importance period
- The problem occurs when this lower importance period starts to consume more CPU and causes the higher importance period to miss its goal
- WLM will recognize this condition and fix it, but WLM can be slow to react

Note: To make the point, just a few priorities between DP 203 and DP247 are shown.

CPU Critical aka Long-term CPU Protection



- Long-time option in your WLM service definition
- Enabled by setting YES for CPU Critical on a Service Class
 - Must be a single-period SC and cannot be discretionary
- Ensures that the CPU Critical SC always has a dispatching priority that's greater than the DP of lower importance service class periods
- Note some small amount of lower-importance work may still get higher DP:
 - Due to promotion for locks, resource contention, etc.
 - Small consumers
- General recommendation has been to avoid this option
 - Allows WLM to make better decisions about balancing overall throughput to best meet the goals of all work

! **Important:** The use of these options limits WLM's ability to manage the system. This may affect system performance and/or reduce the system's overall throughput.

i **Note:** Implicit CPU Critical for importance 1 work can impact the CPU distribution to lower importance work. Ensure that the goals are appropriate given their importance level. Evaluate the current distribution of CPU at different importance levels, especially those that are covered by CPU Critical to ensure that they have consistent CPU demands.

Setting CPU Critical Control



- CPU Critical is set at the service class level
 - Can be set for address space, enclave, or transaction-oriented work
 - To help ensure that critical work will have a higher CPU DP than lower importance work

```
Service-Class  Xref  Notes  Options  Help
-----
                                         Modify a Service Class                               Row 1 to 2 of 2
Command ==> _____

Service Class Name . . . . . : STCHI
Description . . . . . Important non-system Started Tsk
Workload Name . . . . . STC          (name or ?)
Base Resource Group . . . . . _____ (name or ?)
Cpu Critical . . . . . YES          (YES or NO)

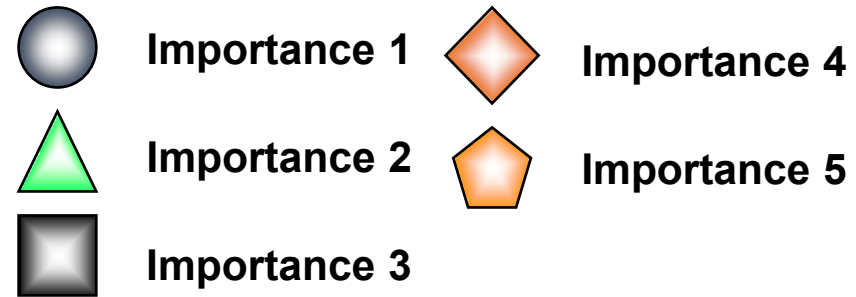
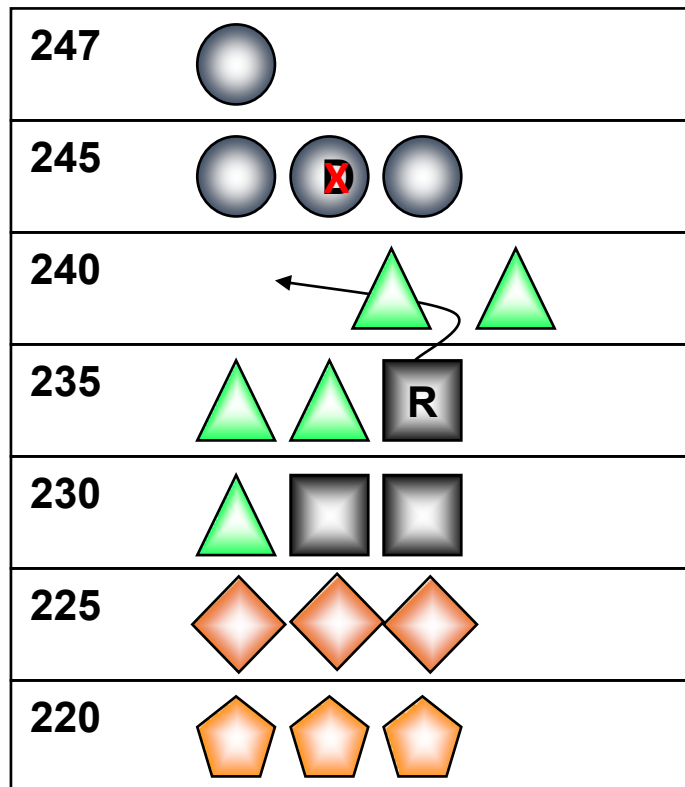
Specify BASE GOAL information.  Action Codes: I=Insert new period,
E=Edit period, D=Delete period.

      ---Period---  -----Goal-----
Action #  Duration  Imp.  Description
-----
   1      1          1  Execution velocity of 60
***** Bottom of data *****
```

Effect of WLM CPU Critical Control



- With CPU Critical set for the importance 1 workloads, all lower importance workloads will have a lower dispatching priority



- In the case where that one Importance 1 workload is overperforming and the importance 3 workload is underperforming, now it can't shift those workload DPs around as much to help the importance 3 workload.
- The R workload may be helped somewhat by moving it up to DP 240, but maybe it's not enough because that Importance 1 donor is really consuming a bunch of CPU (and it could suffer some CPU delay and still meet its goal)

Note: To make the point, just a few priorities between DP 203 and DP247 are shown.



Implicit CPU Protection in z/OS 3.1

z/OS 3.1 Implicit Long Term CPU Protection



- New option for “Implicit” Long-Term CPU Protection
 - In other words, CPU Critical without having to specify it on every SC definition
- Default is “On” for importance 1 service class periods
 - Optional, but “Off”, for importance 2 and lower service class periods
 - only assigned to single-period classes and the first period of multi period service classes.
- **We think “On” for importance 1 workloads is a bad default**
 - Could significantly change the dispatching priority of work in the system
 - Goes against historical practices of not changing defaults that change behavior
- DP/Importance inversions are common
 - I.E. Lower Importance work running with a DP above higher importance work
 - Not all such inversions are problematic
 - Not all importance 1 work really should be importance 1

IEAOPTxx – CCImp and CCImpBoost



- Two new parameters for z/OS 3.1
 - CCImp=0|1|2 (CCImp applies when no boost is active)
 - CCImpBoost=0|1|2 (CCImpBoost applies while the boost is in effect)
 - 0
 - CPU protection is not assigned to any work unless CPU Critical has been explicitly assigned for the service class in the active WLM policy.
 - 1|2
 - Specifies the importance level up to which CPU protection is implicitly assigned to the work when CPU Critical has not been assigned for the service class in the active WLM policy.
- Examples:
 - CImp=0 and CCImpBoost=1 suppresses implicit CPU protection but enables implicit CPU protection when a boost is in effect for importance 1 work.
 - CCImp=1 and CCImpBoost=1 assigns implicit CPU protection for any importance 1 work.

From the IBM z/OS 3.1 Documentation

(z/OS 3.1 MVS Planning Workload Management SC34-2662)



*“Beyond explicitly setting the CPU Critical option for single-period service classes of any importance except discretionary, CPU protection is implicitly assigned **for the first period** of any service class of importance 1 (and importance 2 when a boost is in effect).”*

One (and only?) benefit of this: effects multi-period workloads

*“If you want to modify the importance level for automatically setting CPU protection, or even disable it, use the **CCImp** and **CCImpBoost** parameters in the **IEAOPTxx** member.” (See next slide)*

*“**Note:** Implicit CPU Critical for importance 1 work can impact the CPU distribution to lower importance work. Ensure that the goals are appropriate given their importance level. Evaluate the current distribution of CPU at different importance levels, especially those that are covered by CPU Critical to ensure that they have consistent CPU demands.”*

We did an analysis of 116 systems

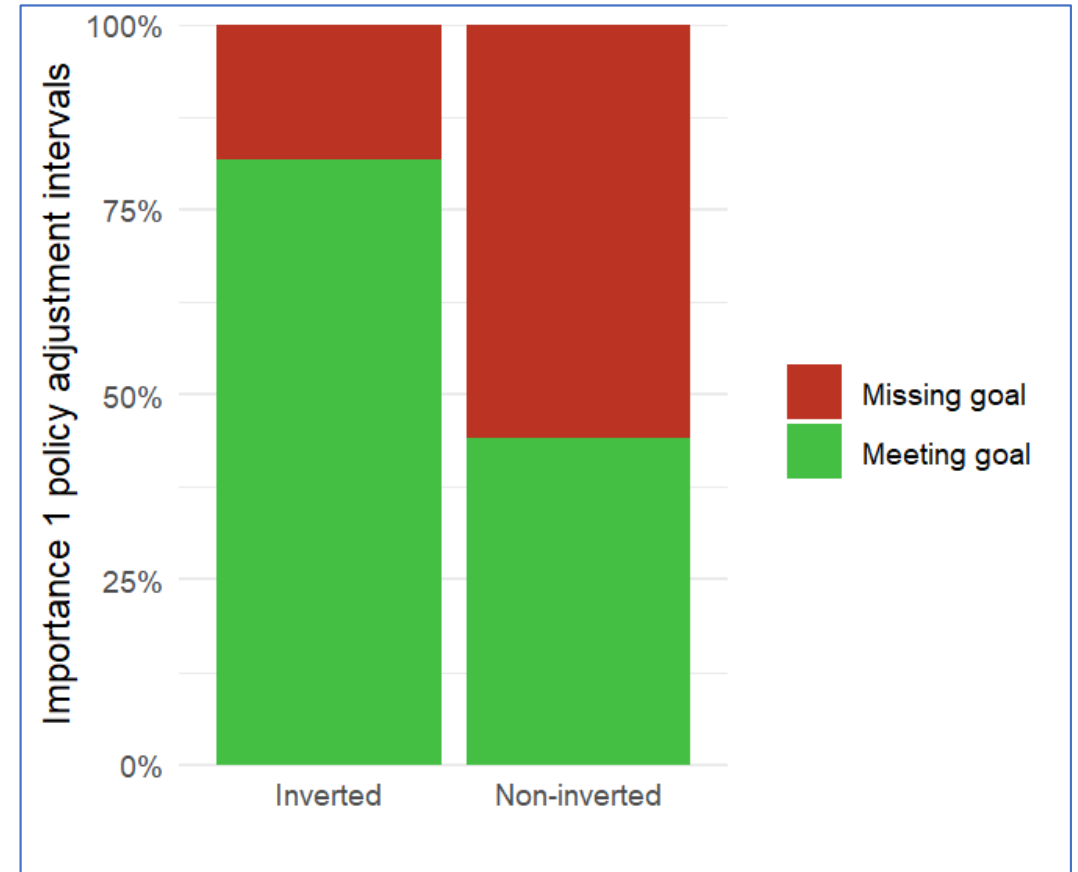


- Covered a variety of sizes from large to small, “IPO” to “Prod”, a couple of dozen customers
- Evaluated a day’s worth of 99.6 data from each system (over 17M records)
- Created 2 new metrics to help understand the risk/benefit:
 - For SCPs that would be bumped down:
Inversion Risk Ratio – relative amount of CPU that would move above the SCP
 - For SCPs that would move up in priority:
Protection Benefit Ratio – relative amount of CPU that would move below this SCP
 - Higher numbers means more potential risk/benefit
 - Can be very high if there’s a relatively large difference in the consumption of the workloads

Effectiveness of z/OS pre-3.1 WLM



- z/OS pre-3.1 WLM behavior sufficiently protects importance 1 workloads and properly balances workload performance across all importance levels
- Study shows Importance 1 workloads which are running at a lower priority than lower-importance workloads are already meeting their goals more often than those running above all lower importance work (82% to 44%)
- It is important to note the limitations of this metric, as goals are not always assigned in accordance with actual needs and expectations



Notes: Inversion Risk Ratio



- The ratio of CPU service of importance 1 work to lower-importance work that is running at a higher dispatching priority (excluding work at DP 248)
 - Basically: more importance 1 work is about to move above a particular SCP, means that SCP is at greater risk
 - E.G. you have SCP A (importance 2) consuming 50 SUs that's running above importance 1 work consuming 5000 SUs. SCP A has an inversion risk ratio of 100.
 - Higher numbers = more risk, but even ratios < 1 represent some risk
 - Risk = Might not get as ready access to CPU, might suffer more CPU delay

Notes: Protection Benefit Ratio



- The ratio of CPU service of lower-importance work that is running at a higher dispatching priority (excluding work at DP 248) to importance 1 workload
 - Basically: more lower importance work is about to move below an importance 1 SCP, means that importance 1 SCP might get a bigger benefit
 - E.G. you have SCP Z at importance 1 and consuming 80 SUs but it's running below lower importance work that's consuming 4000 SUs. SCP Z has a Protection Benefit Ratio of 50.
 - Higher numbers = more potential benefit, but even ratios < 1 represent some possible benefit.
 - Benefit = might get easier/faster access to CPU, might see less CPU delay

Histograms

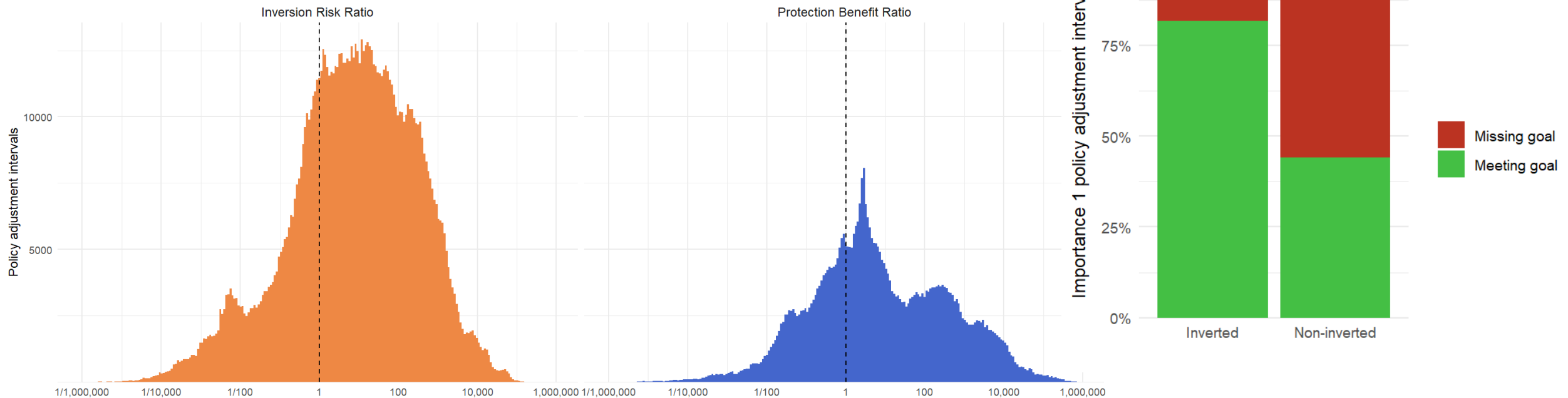


- The next slide has two histograms:
 - On the left, the Inversion Risk Ratio across all such service class periods which are running above an importance 1 workload in the 116 systems. (Total ~1 million service class policy adjustment intervals)
 - On the right, the Protection Benefit Ratio, across all importance 1 SCPs running below lower-importance workloads in the 116 systems. (Total, ~434,000 service class policy adjustment intervals)
- There are instances of very high ratios in both cases, but overall, there's more SCPs at risk than they are that stand to benefit.
 - And most of SCPs that might benefit are already meeting their goals
 - For those that have set goals appropriate to the workload's actual performance needs, the risks may very well outweigh any potential benefits

Study findings

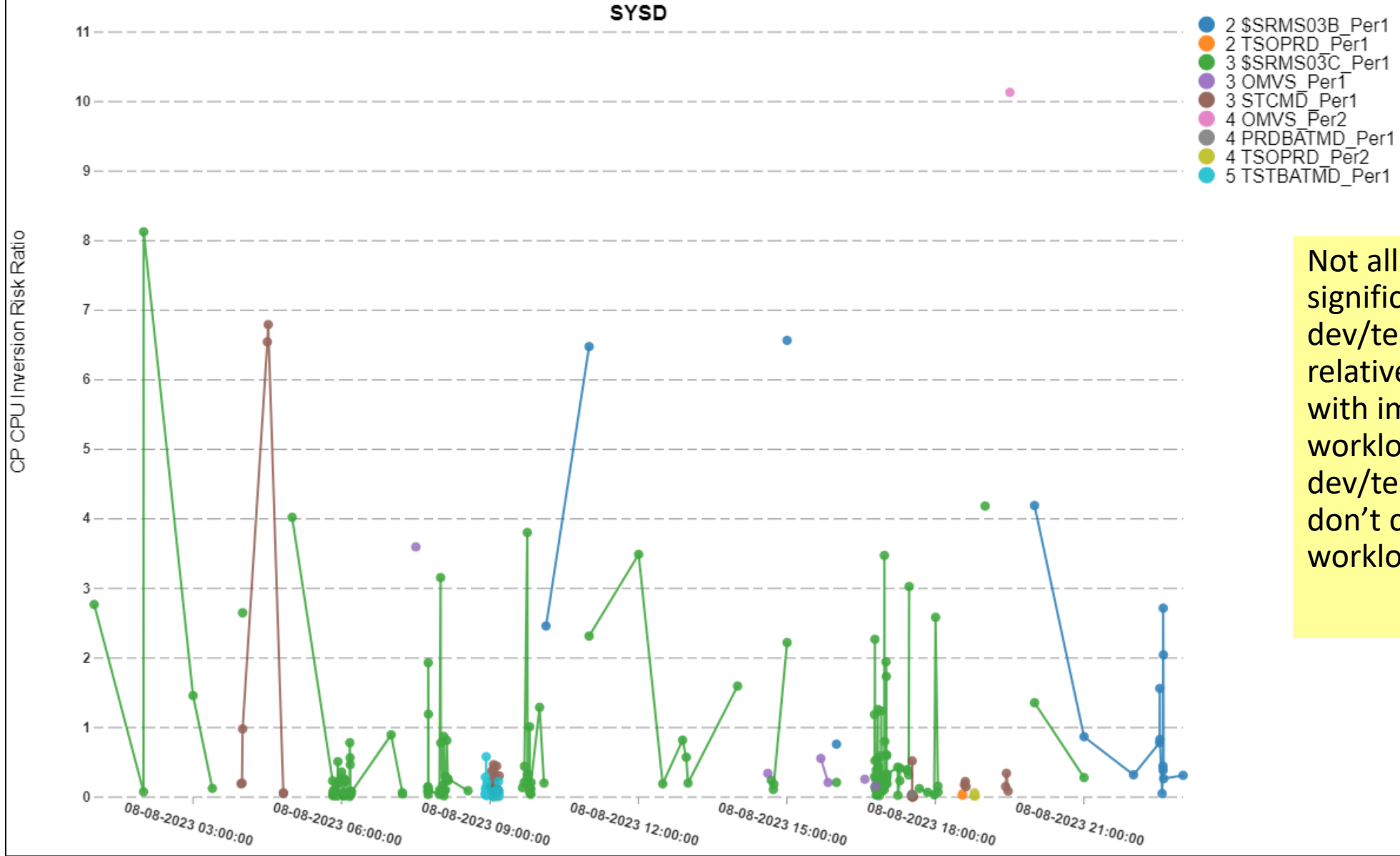


- 78% of systems had at least one interval with an inversion
- 39% of systems had inversions in at least 25% of their intervals
- 82% of “Inverted” Importance 1 workloads were meeting their goal



CP CPU Protection Inversion Risk Ratio

Posed by Importance 1 Workloads Running Below This



Not all systems will have significant risk. This dev/test system has relative few inversion with importance 1 workloads. And it's dev/test: maybe you don't care if those workloads suffer more.



CP CPU Protection Inversion Risk Ratio Posed by Importance 1 Workloads Running Below This

SYSP



Some production systems are considerably more complicated!

Here part of the issue is that there's an importance 1 SC consistently running at low DPs. That might be because it has a poor goal. But if everything is running "ok", maybe it's not actually a bad goal.

The dumbing down of WLM



- In the age of AI, it is strange that IBM is choosing to dumb down WLM
 - Customers could already choose to do this... so why the new forced default?
 - Also, the disablement is in the IEAOPTxx rather than the WLM service definition
- Defaulting all importance 1 workloads to CPU Critical may have significant impacts
 - “Fixing” these inversions is likely to have a larger negative impact on the lower importance workload than it will a positive impact on the importance 1 workloads
- Making this the new default disregards:
 - The practice of avoiding changing defaults that would change the behavior of the system
 - Long-standing recommendations to generally avoid the use of CPU Critical
- It is not correct to assume that all work is classified to the proper importance and given a proper goal that is consistent with the business and technical requirements, but it’s also not correct to assume goals were improperly set as well.

Our thoughts



- We don't see the need for this change
 - A significant part of the premise of WLM was that it would manage dispatching priorities and could intelligently move them in possibly counter-intuitive ways to better balance throughput for diverse workloads
 - If you want, you can make all importance 1 work CPU Critical today (at least for single-period SCs)
- We'd recommend turning this off for z/OS 3.1 and wish that was the default
- If you want to go to z/OS 3.1 with it on, we might suggest
 1. Evaluate which workloads are at risk
 2. Before z/OS 3.1, incrementally add CPU Critical to importance 1 workloads
 - If something goes wrong, you can back out your change and z/OS 3.1 doesn't get the blame
- We do sometimes recommend CPU Critical, but it's an exception, not the rule
- Again: emerging area of study, we might refine our recommendations over time

If you want more details...



- White paper available:
Summary of Potential Impact of Implicit Long-Term CPU Protection
 - For a copy, please send an email to performance.questions@epstrategies.com
- **Special thanks to Ethan Chapman for his statistical and R expertise!**
- **QUESTIONS?**