Revisiting WLM Options for the Modern Mainframe

Scott Chapman

Enterprise Performance Strategies, Inc.

Scott.chapman@EPStrategies.com

Contact, Copyright, and Trademarks

Questions?

Send email to performance.questions@EPStrategies.com, or visit our website at https://www.epstrategies.com or http://www.pivotor.com.

Copyright Notice:

Trademarks:

Enterprise Performance Strategies, Inc. presentation materials contain trademarks and registered trademarks of several companies.

The following are trademarks of Enterprise Performance Strategies, Inc.: **Health Check®, Reductions®, Pivotor®**

The following are trademarks of the International Business Machines Corporation in the United States and/or other countries: IBM®, z/OS®, zSeries®, WebSphere®, CICS®, DB2®, S390®, WebSphere Application Server®, and many others.

Other trademarks and registered trademarks may exist in this presentation

EPS: We do z/OS performance…

- Pivotor - Reporting and analysis software and services
  - Not just reporting, but analysis based reporting based on our expertise
- Education and instruction
  - We have taught our z/OS performance workshops all over the world
- Consulting
  - Performance war rooms: concentrated, highly productive group discussions and analysis
- Information
  - We present around the world and participate in online forums

z/OS Performance workshops available

During these workshops you will be analyzing your own data!

- Essential z/OS Performance Tuning
  - Milwaukee WI, June 10-14, 2019
- Parallel Sysplex and z/OS Performance Tuning
  - Via the internet, November 12-14, 2019
- WLM Performance and Re-evaluating Goals
  - Virginia Beach VA, October 21-25, 2019

Like what you see?

- The z/OS Performance Graphs you see here come from Pivotor™ but should be in most of the major reporting products

- If not, or you just 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 retention
  - That still encompasses over 100 reports!

All Charts (132 reports, 258 charts)
All charts in this reportset.

Charts Warranting Investigation Due to Exception Counts (2 reports, 6 charts, more details)
Charts containing more than the threshold number of exceptions

All Charts with Exceptions (2 reports, 8 charts, more details)
Charts containing any number of exceptions

Evaluating WLM Velocity Goals (4 reports, 35 charts, more details)
This playlist walks through several reports that will be useful in while conducting a WLM velocity goal an

Agenda

- Historical Perspective
- WLM Basics Quick Review
- Service Coefficients
- I/O Management
- Discretionary Goal Management

Historical Perspective

WLM History

- WLM was originally developed in the early 90s
  - Released with MVS 5.1 in 1994: making it 25 years old now!
- Replaced IPS/ICS management ("compatibility mode")
  - Avoid having system administrator micromanaging individual task dispatching priorities and instead give the system a policy that defines the importance of the work and how the work should perform
  - This is still a pretty radical concept in terms of managing system performance!
- Most shops converted to WLM management ("goal mode") by early 2000s
  - That's still going on 20 years ago!
  - Goal mode was required in z/OS 1.3, released in 2002

See Peter's 25 Year WLM Retrospective presentation

What was different 20-25 years ago?

- The mainframe has changed dramatically in the last 20-25 years!
  - First CMOS machine: 9672-R11: 696 SU/sec
  - Last bipolar machine: 9021-711: 3,018 SU/sec
  - *Smallest* z14: 3907-A01: 4,509 SU/sec
  - Full speed z15: 8561-701: 103,488 SU/sec

Single Engine SU Ratings

- A few GBs of memory was a very large machine in the early 90s
  - Minimum z14 ZR1 memory is 64GB (z15 minimum = 512GB!)
- IBM RAMAC Array DASD introduced in 1994
  - IBM ESS "Shark" was introduced in 1999

total DASD shipments are expected to increase 23% to 900TB this year and then rise another 33% to 1200TB in 1995

ComputerWorld Dec94

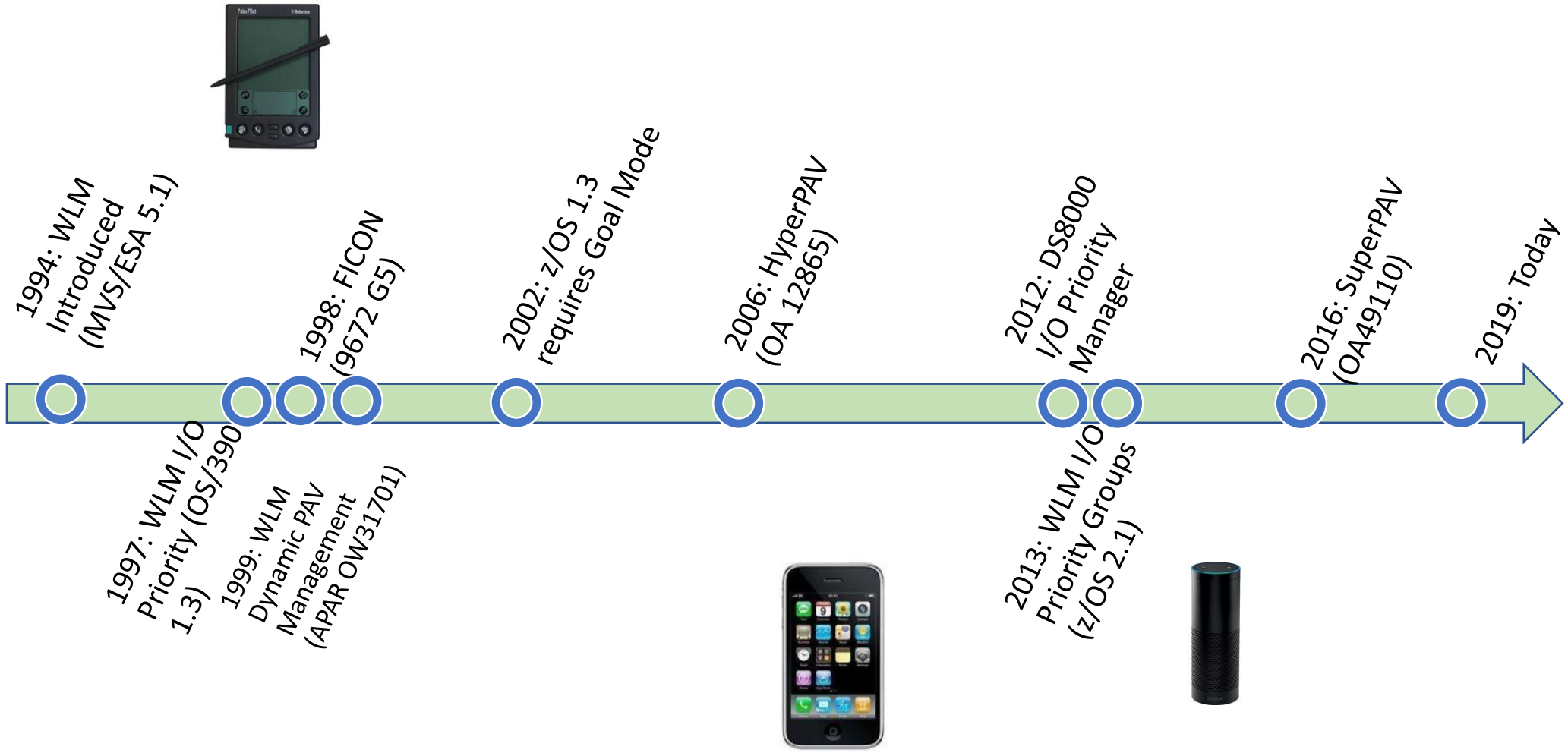
- Although ESCON was available, Parallel channels were still prevalent in 1994
  - Today: ESCON no longer available: FICON Express8 and FICON Express16
  - Only 1 concurrent I/O per ESCON channel vs. multiple for FICON

Alamo paid 'in the $3-and-change range [per MB]'

ComputerWorld Dec94

Recommendations always need to be revisited as technology changes

WLM & I/O Features Timeline



WLM Basics Quick Review

A very brief overview

Service Classes (SC)

- Service Classes define work with similar:
  - Work types
  - Performance goals
  - Resource requirements
  - Importance to the installation
- A service class consists of:
  - Service class name
  - Service class description
  - Period(s)
    - Performance goal and importance
    - Durations
  - Resource group name
- Service class can only be associated with one workload
- Can define up to 100 service classes (since OS/390 1.3)

Work is managed by Service Class Period

COWPBAT Service Class

Period 1
Goal = Velocity 15
Importance 4
RGRP = FENCED

PRODTSO Service Class

Period 1 – 500 SU
Goal = RT 0.5 sec, 95%
Importance 2
RGRP =

Period 2 – 1500 SU
Goal = RT 1.5 sec, 90%
Importance 3
RGRP =

Period 3
Goal = Velocity 31
Importance 4
RGRP =

Service Class Periods (SCP)

- WLM manages work at the Service Class Period level
- Service Classes have 1 to 8 periods
  - Most SCs will be single-period (generally don't need more than 2 or 3)
  - Each period has its own goal, importance, and duration
- Duration: Amount of resource consumed, measured in service units
  - As consumption exceeds period duration work moves to the next period
- Why use multiple periods in a SC?
  - Protect and promote "trivial" transactions
  - Prevent resource-intensive transactions from impacting other work
    - Don't let the hogs trample the humming birds!

PRODTSO Service Class

Period 1 – Duration 500
Goal = RT 0.1 sec, 95%
Importance 2

Period 2 – Duration 15000
Goal = RT 1.0 sec, 90%
Importance 3

Period 3
Goal = RT 3.0 sec, 80%
Importance 4

Service Coefficient/Options

- Service coefficients: scale service units from raw values
- Options turn on or off features:
  - I/O Priority Management: WLM will set I/O priorities separate from CPU priorities
  - I/O Priority Groups: Make sure some SCs always have higher I/O priority
  - Dynamic Alias Tuning: WLM manages PAVs based on goals
  - Deactivate Discretionary Goal Management: Don't help discretionary

```
Coefficients/Options  Notes  Options  Help
------------------------------------------------------------------------
              Service Coefficient/Service Definition Options
Command ===> ____________________________________________________________

Enter or change the Service Coefficients:

CPU  . . . . . . . . . . . . . . 1.0      (0.1-99.9)
IOC  . . . . . . . . . . . . . . 0.5      (0.0-99.9)
MSO  . . . . . . . . . . . . . . 0.0000   (0.0000-99.9999)
SRB  . . . . . . . . . . . . . . 1.0      (0.0-99.9)

Enter or change the service definition options:

I/O priority management  . . . . . . . . YES  (Yes or No)
Enable I/O priority groups . . . . . . . YES  (Yes or No)
Dynamic alias tuning management  . . . . YES  (Yes or No)
Deactivate Discretionary Goal Management NO   (Yes or No)
```

Commonly used coefficients and options today

Service Coefficients

WLM & SRM's Concept of Service

- One of the basic functions of WLM is to monitor the dynamic performance characteristics of all address spaces and enclaves under its control
  - WLM needs to ensure distribution of system resources under its control
- When transactions run on z/OS, they use resources
  - Processor / CPU to execute instructions
  - Memory (page frames in central storage)
  - I/O operations
- The standardized measurement of these resources is known as *service units*
- At the start of every new transaction, service unit count starts out as zero
  - Then as transaction runs the service accumulates until the end of the transaction

SUs and Period Transitions

- For Service Classes with multiple periods, work transitions between periods as it consumes SUs

WLM Service Units

- CPU Service Units
  - Task (TCB) and preemptible SRB execution time multiplied by SU/sec constant
  - Also includes time used by address spaces in cross memory mode
- SRB Service Units
  - Non-preemptible Service Request Block CPU time for local and global SRBs, multiplied by SU/sec
  - Also includes time used by address spaces in cross memory mode
- I/O Service Units (also known as IOC)
  - Measurement of individual dataset I/O activity and JES spool reads and writes for all datasets associated with an address space or enclave
  - Calculated using I/O block (EXCP) counts and device connect time
- Storage Service Units (also known as MSO)
  - Measurement of central storage usage, but scaled to attempt to bring in line with CPU capacity
    - Calculated as (Central Storage Page Frames) x (CPU SU) x 1/50
  - Does not include central storage frames used by the caller while referencing the private area of a target address space

Service Consumption

$$\text{Service} = \left(\begin{array}{l} (\text{CPU SDC} * \text{CPU Service Units}) \\ + (\text{SRB SDC} * \text{SRB Service Units}) \\ + (\text{IOC SDC} * \text{IOC Service Units}) \\ + (\text{MSO SDC} * \text{MSO Service Units}) \end{array} \right)$$

- Service consumed is used by WLM for:
  - Period switching (period duration is in service units)
- Note that this is sometimes called “weighted” service units
- “Unweighted” service units recorded in some places, usually relative to CPU consumption
  - Using CPU & SRB coefficient of 1.0 makes weighted and unweighted CPU/SRB service units the same

Multi-Period Service Class Example

Reminder: duration is resource consumption

```
          ---Period---  ---------------------Goal---------------------
Action  #  Duration   Imp.  Description
  __
        1  2000        2    90% complete within 00:00:00.250
        2  3000        3    90% complete within 00:00:00.500
  __    3              4    80% complete within 00:00:01.000
  __
******************************** Bottom of data ********************************
```

Reminder: goals are elapsed times

- Generally want the period durations and elapsed time goals to make sense relative to each other
  - Unless the last period is a “penalty” period
- If the SRM constant is 10,000 SUs/sec:
  - 2000/10000 = 0.2s of CPU time so that makes sense relative to 0.25s ET
  - Duration’s CPU time doesn’t have to calculate to less than the ET, but if it’s far above the ET, there’s a greater chance there will be transactions in the period that have no chance of meeting the goal
- But I/O and MSO will impact period aging as well: does that make sense?

Why do you want to switch periods?

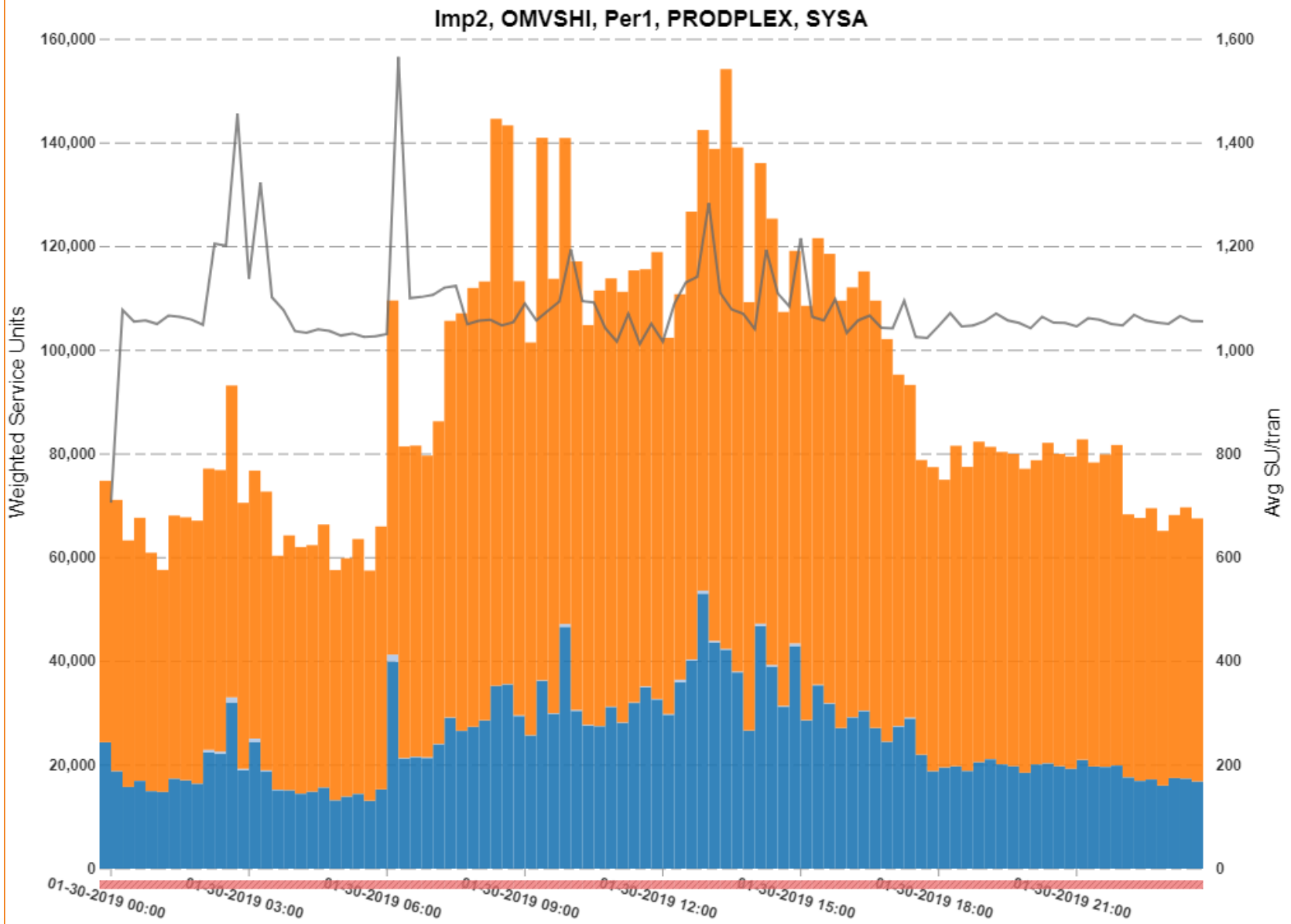
- Give priority to the trivial transactions: get them in and out really quickly
  - For example: TSO ISPF page down
- Prevent heavy-weight transactions from impacting the normal transactions
  - For example: TSO interactive compile

- Main thing we're trying to control here is the dispatching priority
- These concepts also apply to other transactional work, such as DDF
  - The concept of a "penalty period" for DDF SCs that include user adhoc SQL (e.g. QMF) may be very beneficial
  - Multi-period service classes commonly used for: TSO, DDF
  - Multi-period service classes sometimes used for: OMVS and Batch

Should period aging be based on I/O?

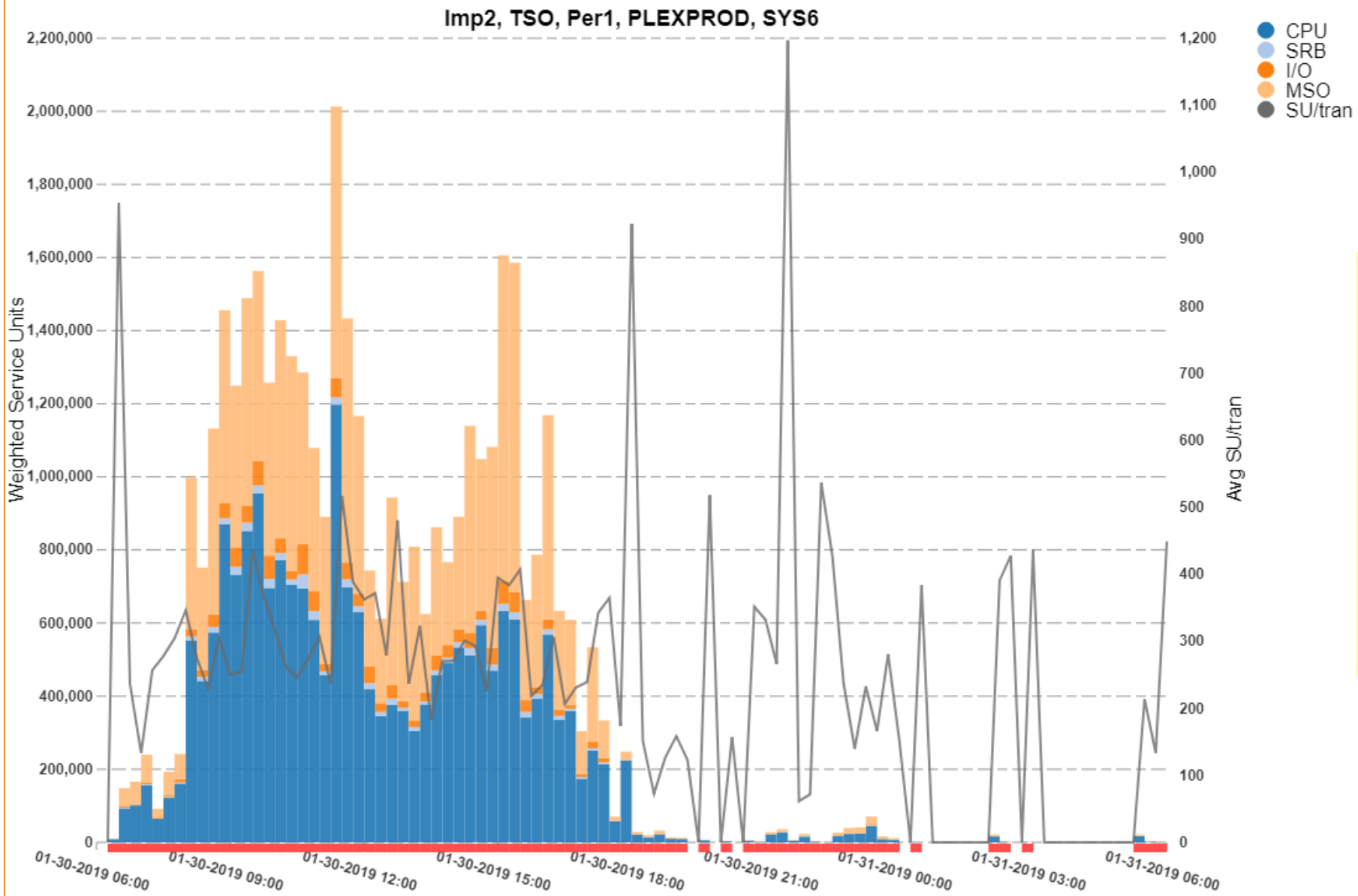
- Including I/O service is probably not necessary
  - Most trivial transactions are doing trivial I/O, so whether or not to include I/O service is largely immaterial for those transactions
  - For aging to later periods, if the transaction is doing a bunch of I/O, then it's giving up the CPU regularly so its CPU impact to other workloads is going to be limited
- DDF service class periods don't accumulate I/O service
  - They accumulate I/O time and delay, but not IO Service Units
- Many (but not all) TSO transactions won't have I/O
- Excluding I/O (and MSO) service would make it easier to relate period durations to potential CPU consumption

WLM Multiple Period - Service Units Consumed by Type



In this particular case, the IOC SDC is 0.5 (a common value), and work is aging out of period 1 primarily because of I/O, not CPU consumption. Even if the SDC was 0.1, I/O would significantly drive period aging.

WLM Multiple Period - Service Units Consumed by Type



In this particular case the MSO SDC is set to 0.001 but you can see that MSO SUs are still quite significant relative to the CPU SU.

New SDC Recommendations

```
Enter or change the Service Coefficients:

CPU  . . . . . . . . . . . . . . . 1.0      (0.1-99.9)
IOC  . . . . . . . . . . . . . . . 0.0      (0.0-99.9)
MSO  . . . . . . . . . . . . . . . 0.0000   (0.0000-99.9999)
SRB  . . . . . . . . . . . . . . . 1.0      (0.0-99.9)
```

- CPU and SRB Coefficients of 1 avoid confusion between raw and scaled SUs
- MSO recommendation has been 0 for a very long time: *even small MSO SDCs can result in relatively large MSO SUs*
- New recommendation of 0 for IOC keeps period aging focused on CPU consumption and makes it easier to set good period durations
- IBM has said as of z/OS "version after 2.4" the above will be enforced
  - So you have a couple of years, but we don't see any reason to wait

Impact of Changing Coefficients

- IOC & MSO of 0 will ensure period aging based on CPU consumption
- Recorded SU amounts will change in SMF 30 and 72 records
  - Only really a concern if you're actually using those values for some reporting
- If changing CPU/SRB coefficient from 10 to 1: cut all period durations to 10% of existing durations
  - *Assuming existing durations are good of course!*
- When changing I/O (or MSO) SDC to 0.0: *may* need to alter durations
  - For many service classes, the impact may be minimal or non-existent
  - If you don't change durations, likely more transactions will complete in period 1 and period 1 will accumulate more CPU
    - This could be a good & desired thing, especially for transactional workloads
  - For Pivotor customers: review the SU Consumption reports under WLM Multi-Period Analysis
  - For everyone else: we'd be happy to do a free cursory review with you!

What do we see in WLM policies today?

- A bit over 90% of WLM policies use CPU/SRB of 1
  - Meaning a bit under 10% are still using 10 and should change to 1
- About 85% use MSO of 0
  - Meaning about 15% should change to 0
- About 2% use IOC of 0
  - Meaning 98% should change to 0
  - ~45% use 0.1 with CPU of 1
  - ~45% use 0.5 with CPU of 1 or 5 with CPU of 10
  - Remainder use 1 with CPU of 1
- Very rarely do we see other combinations

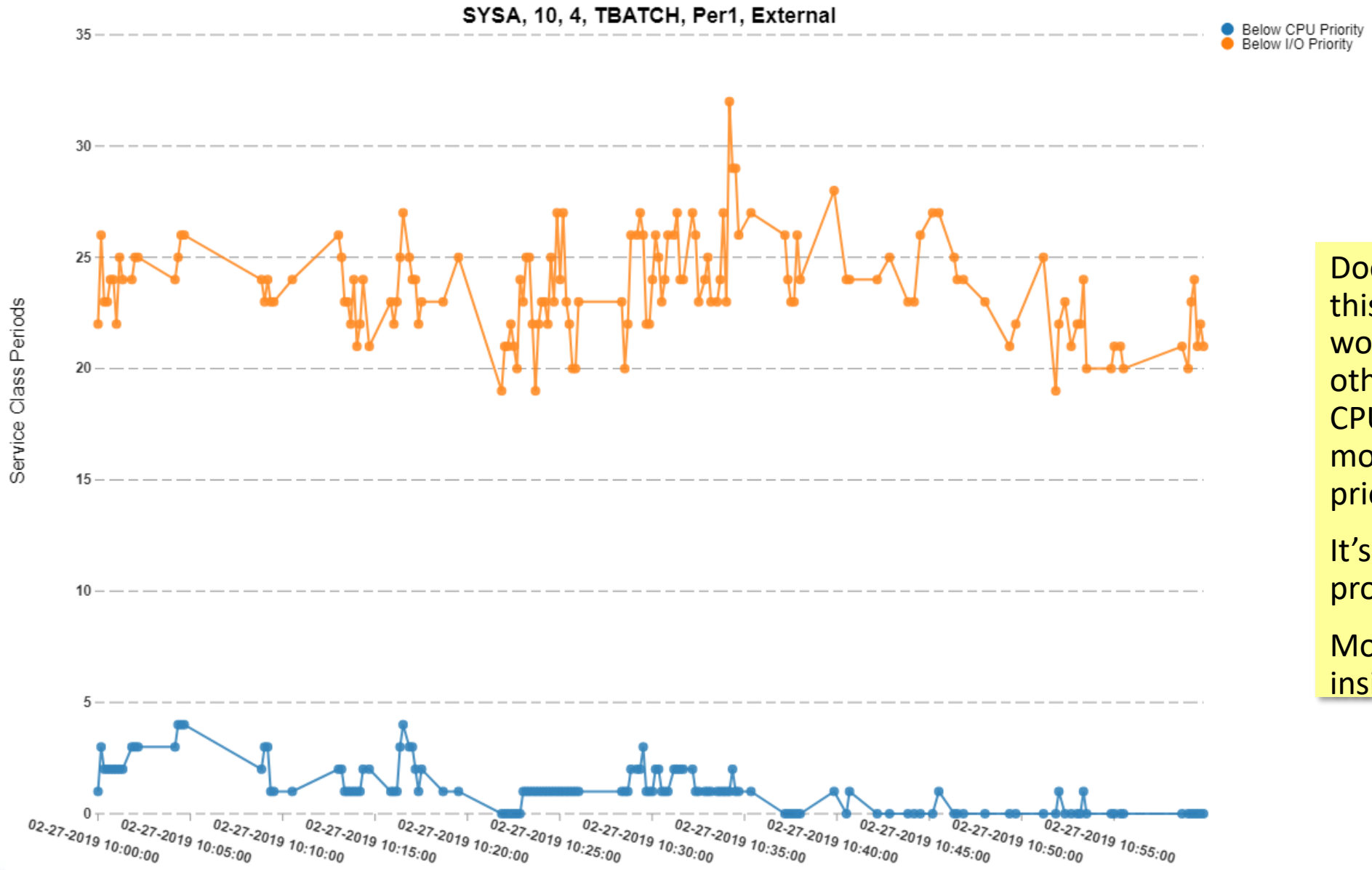
I/O Management

I/O Priority Management Introduction

- I/O Priority Management was added in 1997
- I/O queueing was a significant issue!
  - ESCON channel could only be servicing a single I/O at a time
  - Without PAVs, a volume could only be servicing a single I/O at a time
- Goal was to manage I/O priority separately from CPU priority
  - Without I/O Priority Management, I/O priority = CPU priority
- Higher priority I/Os could move to the front of the I/O queue
  - Help I/O-limited work without impacting the work's CPU dispatching priority
- I/O priority also passed to the DASD controller
  - So can influence queueing within the controller as well

SCPs With Priorities Below This One

From SMF 99.6



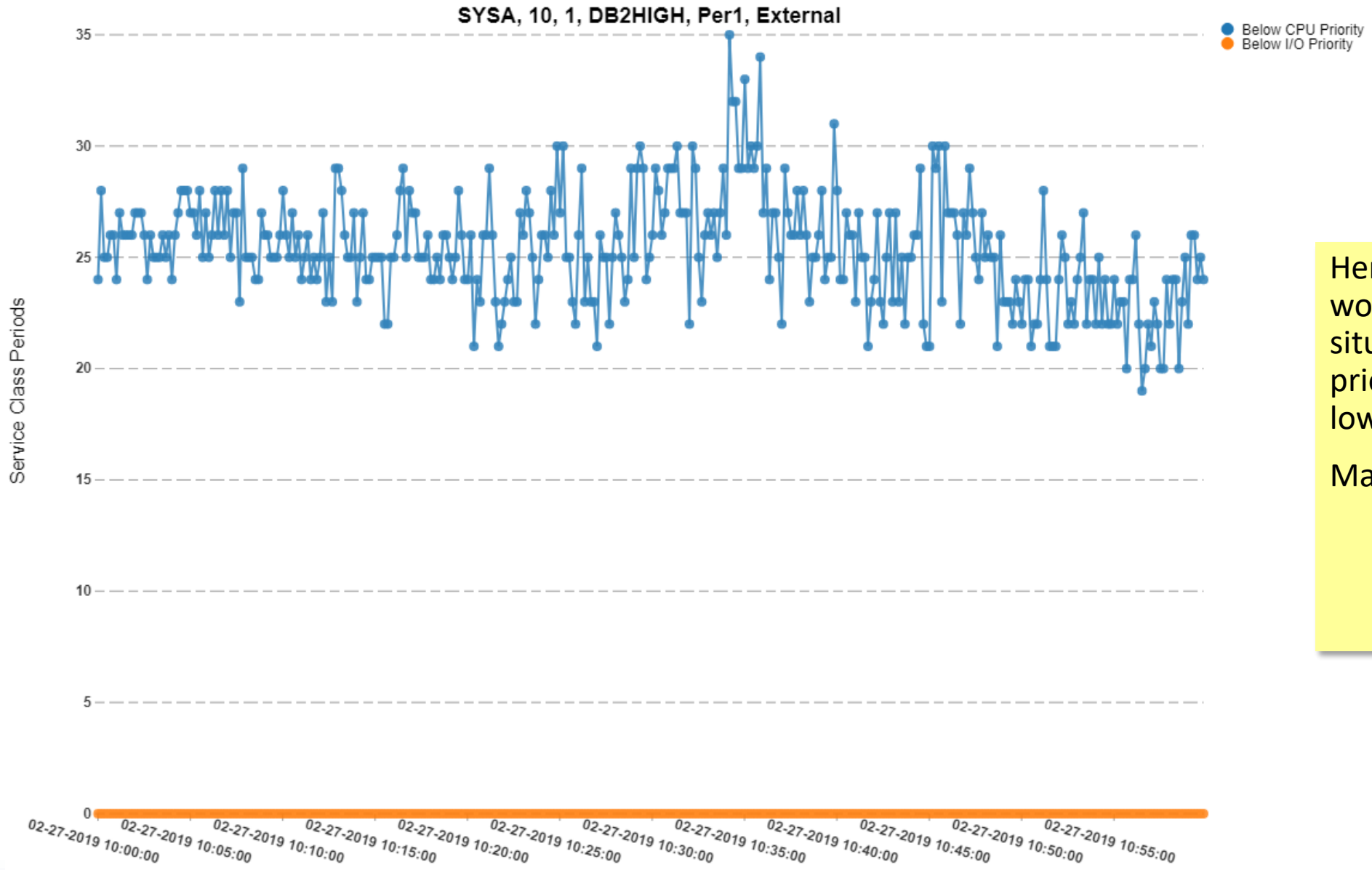
Does it make sense that this importance 4 workload is below most other SCPs in terms of CPU priority, but above most in terms of I/O priority?

It's possible, but probably unlikely.

Most likely: it's insignificant either way.

SCPs With Priorities Below This One

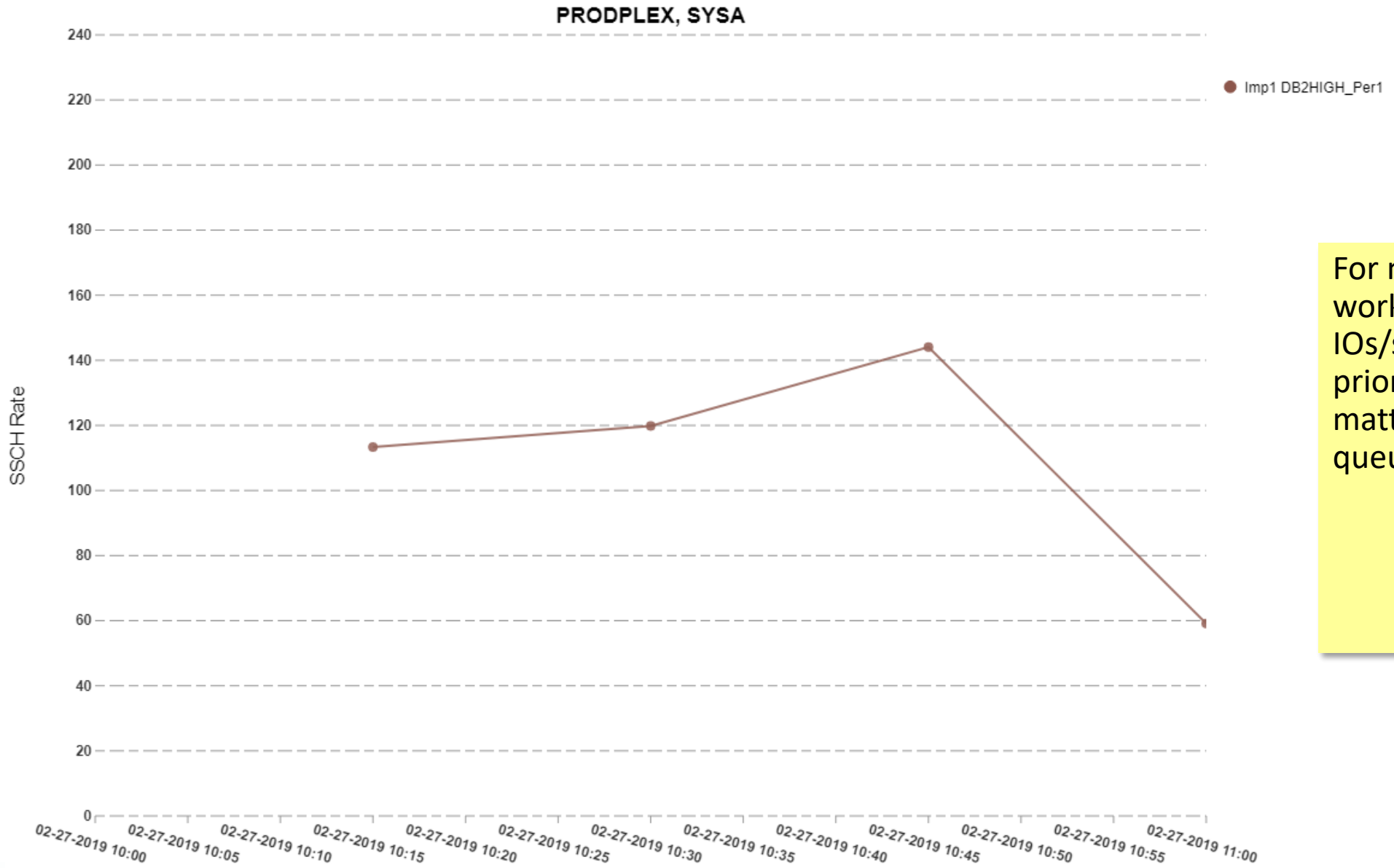
From SMF 99.6



Here's an importance 1 workload in the opposite situation: with its I/O priority apparently as low as it can go.

Maybe it doesn't matter.

WLM DASD - I/O SSCH Rate by Service Class Period Over Time



For most of the hour the work was doing over 100 IOs/sec but its I/O priority doesn't really matter if there's no queueing.

What Changes with I/O Priority Management

- I/O Priority may be different from CPU Dispatching Priority
- Velocity calculation is changed to include I/O Using and I/O Delay "samples"

$$\frac{CPU\ Using + I/O\ Using}{CPU\ Using + I/O\ Using + CPU\ Delay + I/O\ Delay + Paging\ Delay + Swapping\ Delay + MPL\ Delay}$$

  - I/O using and delay samples aren't really samples but rather calculated values based on I/Os performed, disconnect, connect time, I/O queue time
- Enables the other two I/O options:
  - Enable I/O Priority Groups
  - Dynamic Alias Tuning Management
- Allows DS8K I/O Priority Manager *with* software support
  - Still can use it without software support, if need be

Parallel Access Volumes

- PAVs are additional device addresses that point to the same volume, which allows z/OS to initiate more than one simultaneous I/O to the volume
- Initially was "static" – had to specify PAVs for each volume in the IODF
- Dynamic PAV (introduced 1998) lets WLM manage the PAVs, moving aliases to base addresses as required to reduce I/O queue time
- HyperPAV (2006) took WLM out of the equation: an alias address is assigned for each I/O then returned to the "free" pool for the LCU
  - WLM Dynamic alias tuning management no longer needed
- SuperPAV (2016) extends HyperPAV such that the free pool is across multiple LCUs

If Hyper/Super PAV is not turned on
(HYPERPAV=XPAV or YES in IECIOSxx) find out why!

Vast majority (but not all) customers do have HyperPAV or SuperPAV licensed on their storage subsystem

I/O Priority Groups (2013)

- Enabling I/O Priority Groups allows you to specify a service class an I/O priority of "HIGH"
  - SCs with "HIGH" I/O priority will always have an I/O priority higher than SCs with "NORMAL" (default) I/O priority
- Introduced to deal with the issue that WLM only builds "device sets" that maps service classes to devices every 10 minutes
  - So workload using new device and accumulating I/O delay may not get it's I/O priority increased for 10 minutes
- Changes I/O priority ranges
  - I/O priority group high = I/O priority F8 to FD
  - I/O priority group normal = F3 to F7
  - Without I/O priority groups enabled, but with I/O priority management = F9 to FD
  - So if you enable I/O priority groups, do so for all sysplexes connected to the DASD

Most sites aren't using I/O priority groups, and probably don't need to

DS8000 I/O Priority Manager

- *Licensed* DS8K LIC feature
- Can be with or without software support; to enable software support:
  - Must have I/O Priority Management enabled in WLM
  - Must set STORAGESERVERMGT=YES in IEAOPTxx
- *When a RAID rank is overloaded, throttles* certain I/Os to that rank based on "Performance Group" of the I/O
  - DS8K maps WLM PI, Goal and Importance to Performance Groups
  - Work exceeding its WLM goal will be more likely to be throttled
- Note that throttling only occurs when a RAID rank is saturated
  - Fewer sites have I/O saturation problems today than in the past
    - I/O subsystems are even more robust today: all flash arrays not uncommon
    - EasyTier attempts to balance and improve rank performance over time
    - Large memory helps eliminate I/O

Most sites probably don't need this, but if you think you do, read and understand the doc and review your WLM policy

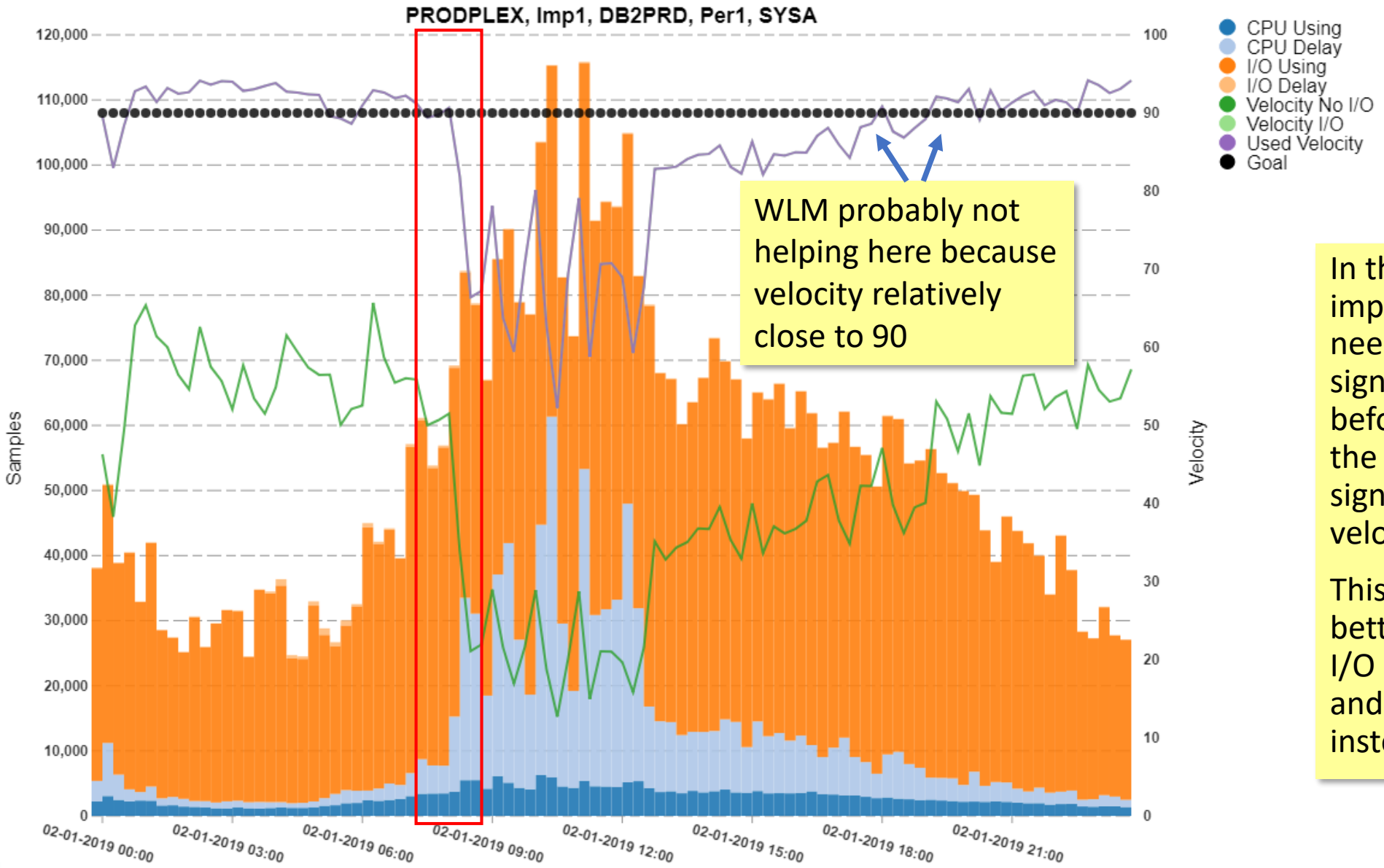
See also DS8000 I/O Priority Manager Redpaper: http://www.redbooks.ibm.com/abstracts/redp4760.html

Reviewing…

- Most sites probably don't need DS8K I/O Priority Manager
- Most sites not using and probably don't need I/O Priority Groups
- Dynamic alias management capability made moot by HyperPAV
- HyperPAV (or SuperPAV) and FICON channels eliminate much queueing on the host
- Improved technology (SSD, Flash, EasyTier, faster processors) makes control unit queuing more difficult as well

- Is there a need to manage I/O priority separate from CPU priority?
  - Probably in most cases, I/O priority being same as CPU priority is fine
- Does it make sense to include I/O in the velocity calculations?
  - Maybe not…

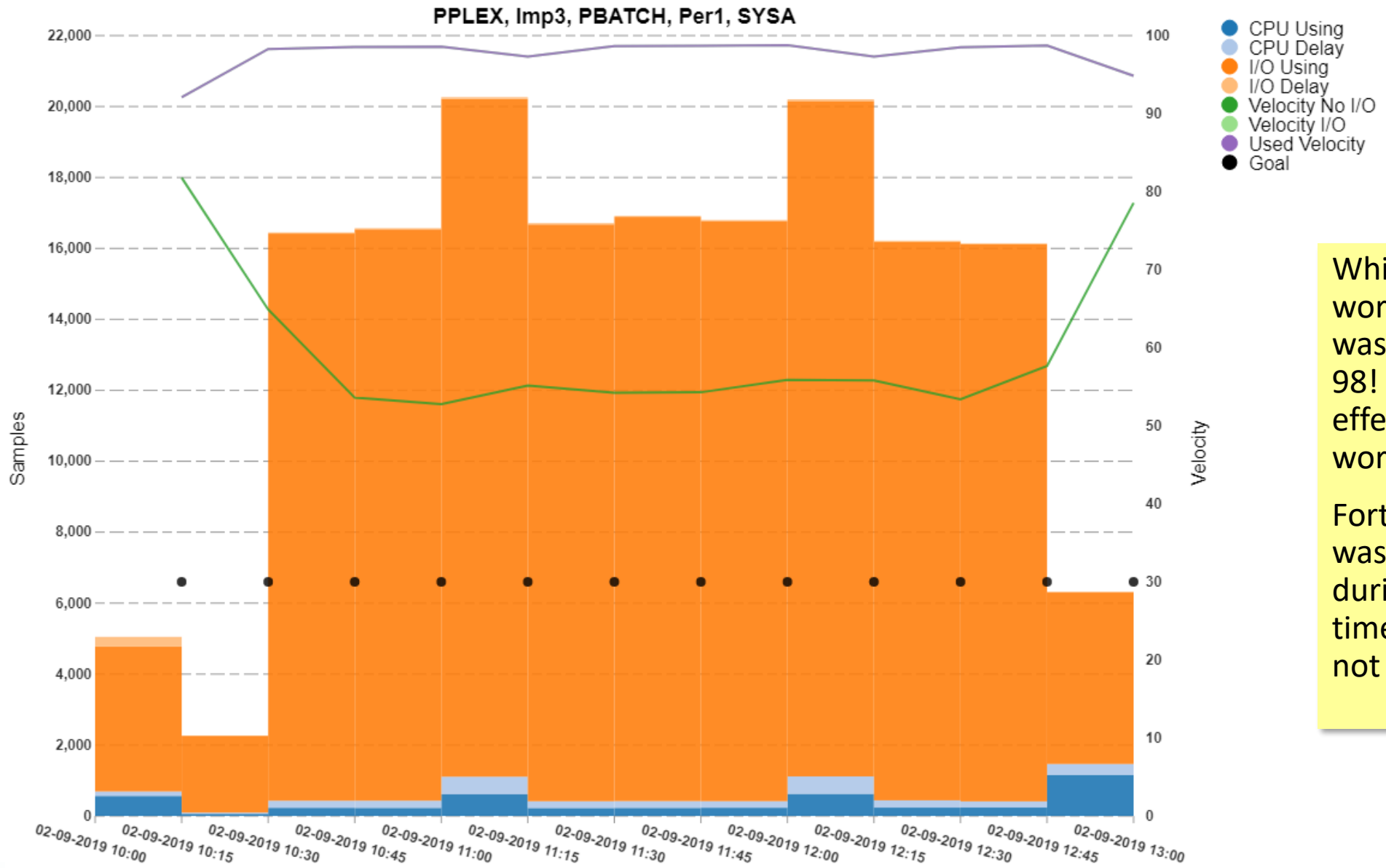
CPU & I/O Contribution to Velocity



In this case, this importance 1 workload needs to suffer significant CPU delays before those overcome the I/O using to significantly alter the velocity.

This workload might be better protected without I/O Priority management and with a goal of ~60 instead of 90.

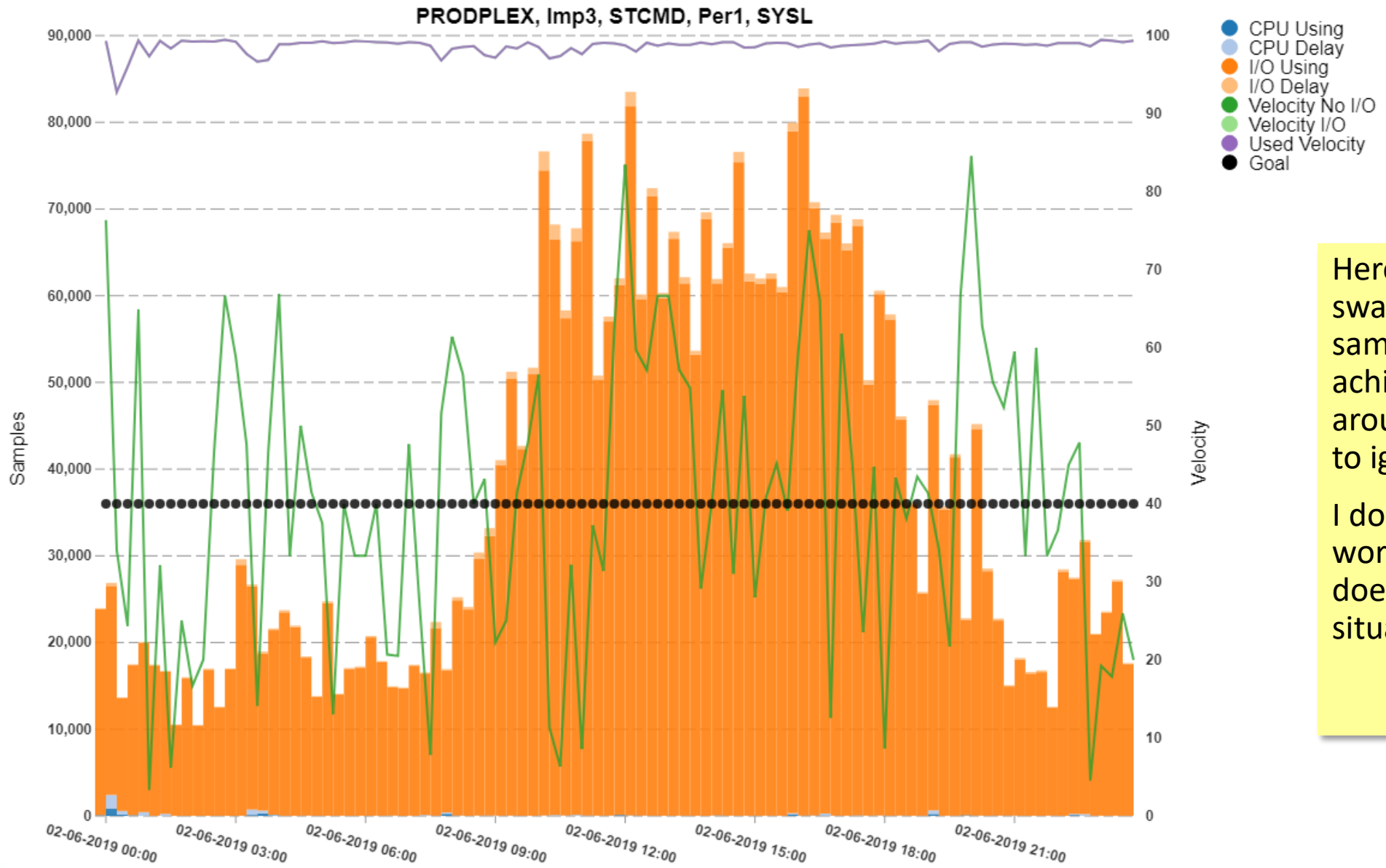
CPU & I/O Contribution to Velocity



While this batch workload was running, it was getting a velocity of 98! WLM has no room to effectively manage this work.

Fortunately the work wasn't suffering much during this particular timeframe, but that may not always be the case.

CPU & I/O Contribution to Velocity



Here I/O has completely swamped STCMD's CPU samples. With an achieved velocity of around 97, WLM is sure to ignore this work.

I don't know what this work actually is, but it doesn't seem like a good situation.

Danger of I/O Priority Management

- We're seeing more and more of these situations where service classes have unnaturally high velocities due to I/O Using
- This may allow the work to suffer significant CPU delay before WLM intervenes
- Goal or achieved velocities of around 90 should be suspect
  - WLM will only make changes that are projected to be significant

What to do?

- Petition IBM to change how I/O using and delay is calculated to bring it better in line with modern I/O capabilities
  - This is probably the best answer, but not likely to be a quick resolution
  - Ideally I'd like a "sample" coefficient that would allow us to adjust the I/O sample counts, much like we can (could?) adjust the I/O service units
- Disable I/O Priority Management in WLM to get I/O out of velocity calculations

Assuming you have HPAV/XPAV

  - This also disables other things (see previous slides), but you probably don't care
  - If you do this, you will need to change velocity goals that have a significant I/O component
    - Pivotor customers: See CPU & I/O Contribution to Velocity charts
  - WLM will probably start paying more attention to more workloads, and the CPU delay for some workloads may smooth out

New I/O Priority Mgmt Recommendation

```
Enter or change the service definition options:

I/O priority management  . . . . . . . . . No   (Yes or No)
Enable I/O priority groups . . . . . . . . No   (Yes or No)
Dynamic alias tuning management  . . . . No   (Yes or No)
```

- In ***most*** cases today the benefits and features enabled by I/O Priority Management do not outweigh the issues caused by artificially high velocities
- For probably 10% or fewer, these features may have value and I/O priority management may need to be retained
  - If you have issues with I/O skewing velocity and can't turn off I/O PM, petition IBM for a better solution
  - In some cases may be able to redistribute work between SCs to improve the situation

What do we see in WLM policies today?

- About 90% have I/O Priority Management enabled (about 10% have it disabled)
  - We now think that those percentages should probably be the opposite
  - Does having separate priorities provide actual value or does it just seem like a good idea because it was 20 years ago?
  - If you're using the DS8K I/O Priority manager, are you really saturating I/O ranks?
    - If so, is throttling some work the correct answer?
- About 75% have Dynamic Alias Management enabled
  - For most this is a moot point either way
- About 90% have I/O Priority Groups disabled and/or have no service classes specifying I/O Priority HIGH
  - This may be about right, but for the 10% that are using it, we might question how much value this is actually providing

Discretionary Goal Management

New option to consider

Helping Discretionary Work

- In order to better help discretionary work get a chance to run, certain service class periods may become donors to discretionary work:
  - Velocity goal of 30 or less or response time goal > 1 minute
  - PI <= 0.81
  - Not part of a resource group
- In some cases, velocity goals of 31 (instead of 30) have been used to avoid becoming a donor
- But if you disable I/O priority management and velocities go down and so should have a goal of 10 or 20, a goal of 31 might not be ideal

Deactivate Discretionary Goal Management

- New service definition option (z/OS 2.3)
- Defaults to “NO” – existing behavior
- Set to “YES” to keep over-achieving low-velocity periods from donating to discretionary work
  - Recommendation: enable if you need, but otherwise leave “NO”
  - This should rarely be a problem

```
Enter or change the service definition options:

 I/O priority management  . . . . . . . . NO   (Yes or No)
 Enable I/O priority groups . . . . . . . NO   (Yes or No)
 Dynamic alias tuning management  . . . . NO   (Yes or No)
 Deactivate Discretionary Goal Management NO   (Yes or No)
```

Summary

New / Current Recommendations

- Service Coefficients
  - 1, 0, 0, 1 – Use zero for I/O and MSO to ensure period aging is based on CPU
- I/O Priority Management
  - No – Avoid skewing velocities with I/O using samples
- Deactivate Discretionary Goal Management
  - Set to YES (deactivate) if needed

```
Coefficients/Options  Notes  Options  Help
------------------------------------------------------------------------
              Service Coefficient/Service Definition Options
Command ===> ____________________________________________________________

Enter or change the Service Coefficients:

CPU  . . . . . . . . . . . . . . 1.0      (0.1-99.9)
IOC  . . . . . . . . . . . . . . 0.0      (0.0-99.9)
MSO  . . . . . . . . . . . . . . 0.0000   (0.0000-99.9999)
SRB  . . . . . . . . . . . . . . 1.0      (0.0-99.9)

Enter or change the service definition options:

I/O priority management  . . . . . . . . NO   (Yes or No)
Enable I/O priority groups . . . . . . . NO   (Yes or No)
Dynamic alias tuning management  . . . . NO  (Yes or No)
Deactivate Discretionary Goal Management NO   (Yes or No)
```