SHARE
EDUCATE ‣ NETWORK ‣ INFLUENCE

Pinpointing Transient Performance Problems with SMF 98 & 99

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

Creators of Pivotor®

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

Contact, Copyright, and Trademark Notices

Questions?

Send email to Peter at Peter.Enrico@EPStrategies.com, or visit our website at http://www.epstrategies.com or http://www.pivotor.com.

Copyright Notice:

Trademarks:

Abstract

- **Using High Frequency SMF 98 and 99 records**

  - Today the standard SMF interval is usually about 15 minutes. In the world of z, 15 minutes is an eternity. So much can happen, and a 15-minute average can hide important bursts of activity and events. This presentation will discuss the high frequency performance measurements of the SMF 98 and 99 records. These z/OS and WLM records assist during performance debugging and analysis. For example, the SMF 98 contain lock and resource usage measurements, and 99 records contain a wealth of information related to WLM algorithm decisions.

  - Peter Enrico will provide an intro to the measurements of the SMF 98 and SMF 99 records, as well as show some very practical uses for these records and a number of performance insights these records will provide.

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
- MSU Reductions
  - Engagements designed specifically to reduce your MSU consumption
- Information
  - We present around the world, bi-weekly webinars, 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
  - October 3-7, 2022
- WLM Performance and Re-evaluating Goals
  - September 12-16, 2022
- Parallel Sysplex and z/OS Performance Tuning
  - February 7-8, 2023
- Also… please make sure you are signed up for our free monthly z/OS educational webinars! (email contact@epstrategies.com)

Like what you see?

- The z/OS Performance Graphs you see here come from Pivotor™
- If you don't see them in your performance reporting tool, or you just want a free cursory performance 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

EPS presentations this week

What	Who	When	Where
PSP: z/OS Performance Tuning - Some Top Things You May Not Know	Peter Enrico Scott Chapman	Tue 1:15	Delaware A
Planning Your Next Mainframe Processor Upgrade	Scott Chapman	Tue 2:45	Franklin C
z/OS Performance Risk Management: Easy Things To Do To Reduce the Risk of Bad Performance	Scott Chapman	Wed 10:30	Franklin C
Pinpointing Transient Performance Problems with SMF 98 & 99	Peter Enrico	Thu 8:00	Franklin A
WLM's Algorithms - How WLM Works	Peter Enrico	Thu 1:15	Franklin C

Transient Performance Problems

A performance related problem that last for a short time.

Useful transient performance SMF records

- Today you should be recording the SMF 98 and most SMF 99 records
  - These records record in sub-minute intervals (e.g. 2, 5, or 10 seconds)
  - Not as much detailed data as in RMF/CMF, but very useful for zeroing in on transient performance problems and evaluating performance on those short intervals
- Many sites have not enabled SMF 98 (but should)
- Many sites have 99s excluded due to IBM recommendations from 1995
  - May have been some validity to those recommendations then, but times and hardware capacity have changed!

SMF 98/99 records to Include

- SMF 98 High-frequency Throughput Statistics (HFTS)
  - IBM recommendation is to record on 5 second interval
    - Can use 5, 10, 15, 20, 30 or 60 seconds
    - 5 second interval is about 400MB-500MB/system/day
- SMF 99 SRM/WLM details
  - Our minimum recommended subtypes: 6, 10, 11, 12, 14
    - These will be around 50-150MB/system/day
  - Subtype 1, 2, and 3 can be quite useful, but can be more voluminous
    - These can be 1-1.5GB/system/day
  - Pivotor customers: send them if you're collecting them!
  - Subtype 13 is fairly voluminous and is undocumented "IBM use only"
    - 150-200MB/system/day
    - We recommend you turn off SMF 99, subtype 13s

```
In SMFPRMxx:

HFTSINTVL(15)
```

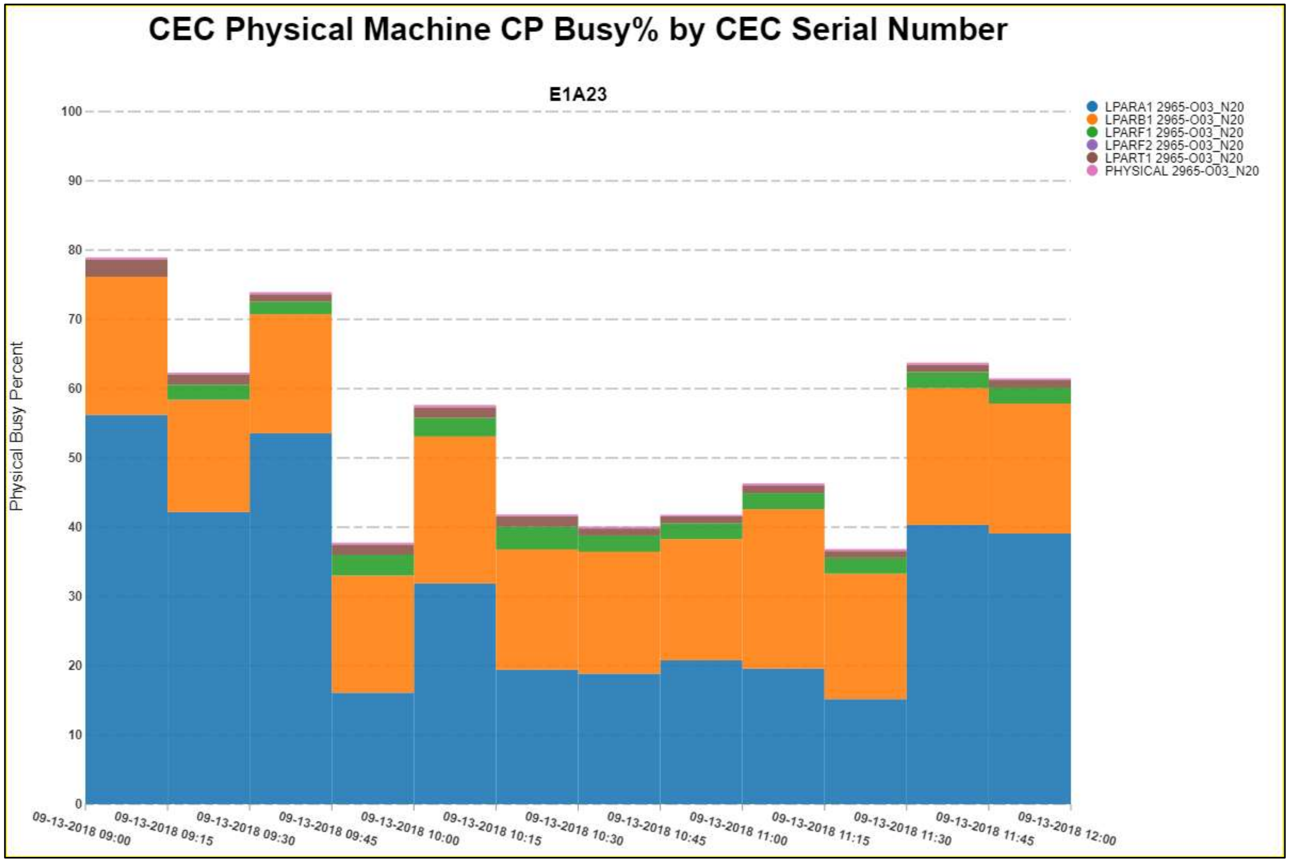
None of these records represent data you will look at every day, but it's nice to have them available when you need them!

Be cautious getting ‘wrapped up’ in reporting of SMF 98s & 99s real-time

None of these records represent data you will look at every day, but it will be nice to have these records available when you need them!

They are most useful for investigating transient performance issues.

Classic CEC Utilization Transient Performance Problem



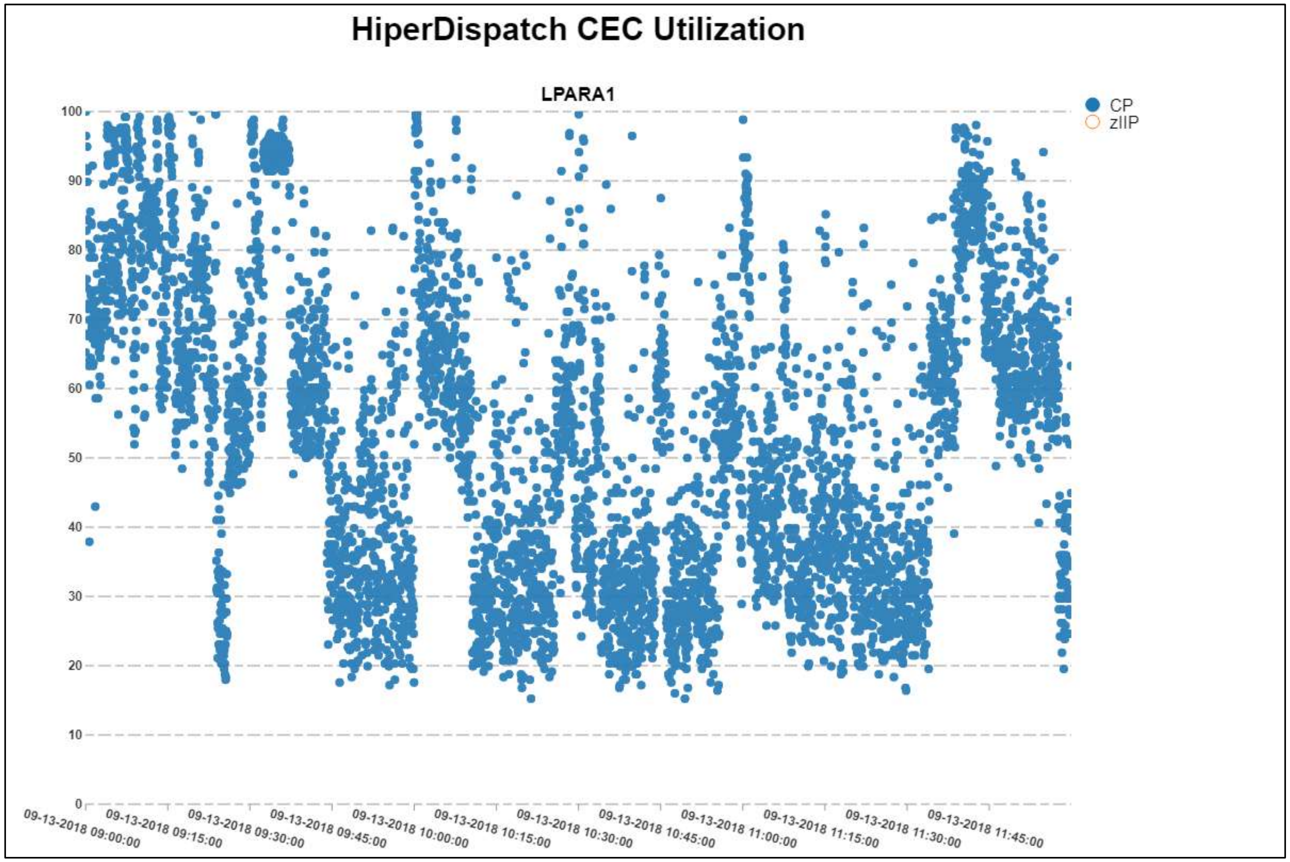
<u>Problem Statement:</u>

System Seemed to Freeze / Stall / things too a long time, but we have lots of available capacity

This is just a standard view of CEC Utilization, here we've narrowed in to just 3 hours in the morning, where it doesn't appear there's really any capacity concerns.

This chart is generated from data that comes from the SMF 70 records. In this example, the measurement intervals are 15 minutes.

Classic CEC Utilization Transient Performance Problem



High Frequency CEC Utilization:

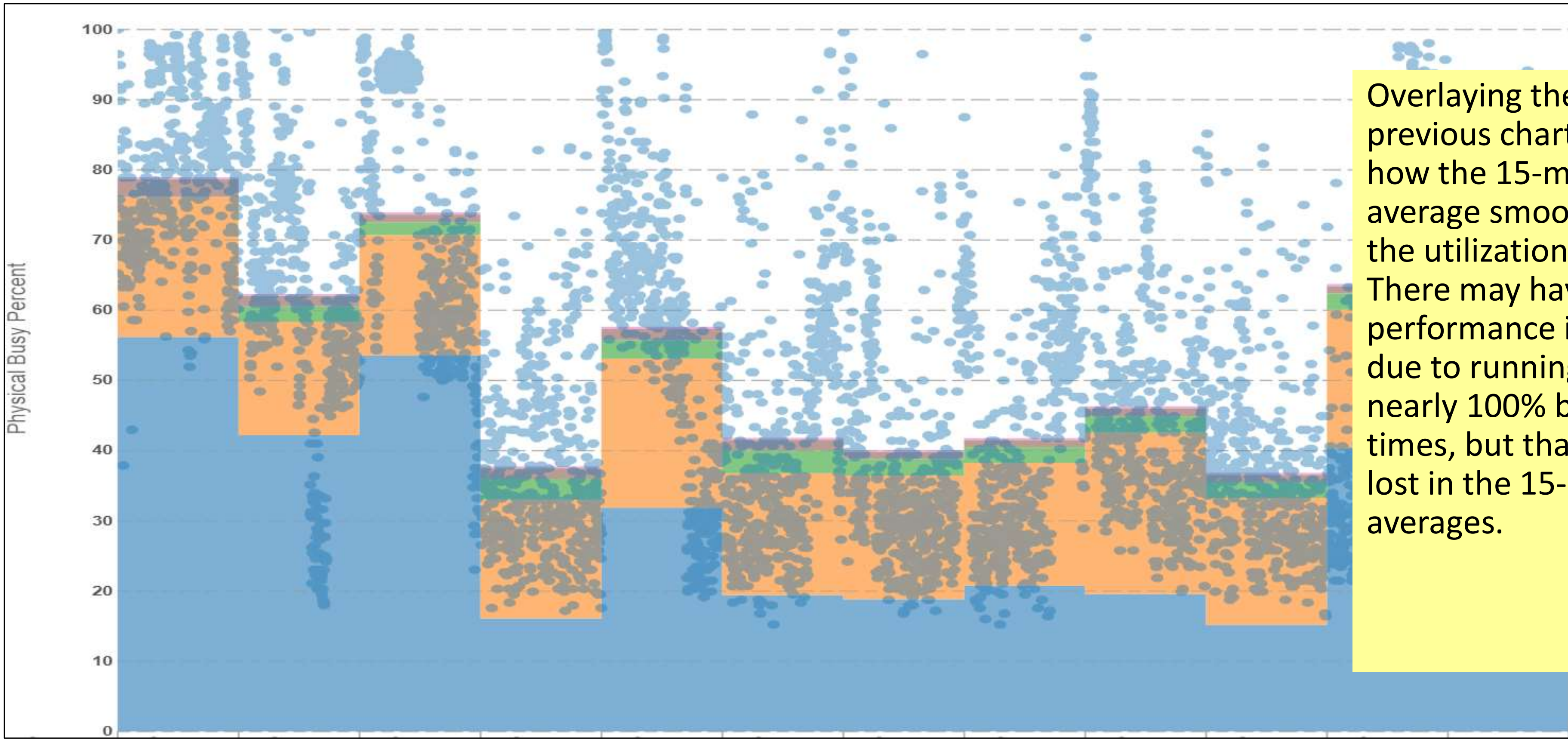
This also is a CEC utilization chart for the same 3 hours as the previous chart.

This data comes from the from the SMF 99.12 HyperDispatch records.

The CEC utilization is at 2-second measurement interval.

Note that this tells a different story than the 15-minute RMF intervals.

Classic CEC Utilization Transient Performance Problem



Overlaying the two previous charts shows how the 15-minute average smooths out the utilization levels. There may have been performance issues due to running at or nearly 100% busy at times, but that's really lost in the 15-minute averages.

SMF 98 Records

High Frequency Throughput Statistics

Overview of SMF 98 Record

- Relatively new SMF record that contains performance information for the z/OS supervisor component about the workload and its significant jobs
  - Includes metrics such as
    - Utilization
    - Concurrency
    - Efficiency
    - Contention and Queuing

```
In SMFPRMxx:

HFTSINTVL(15)
```

- Parameter in SMFPRMxx
  - HFTSINTVL - specifies the time interval, in seconds, for writing SMF type 98 records
    - Supported values are 5, 10, 15, 20, 30, and 60 seconds
    - Make sure to also enable TYPE(98)
    - When specified, SMF type 98 records are collected every five seconds for one minute each hour, at 0, 15, 30, and 45 minutes past each hour.
    - For all other minutes during each hour, SMF type 98 records are written at the interval specified by the HFTSINTVL parameter.

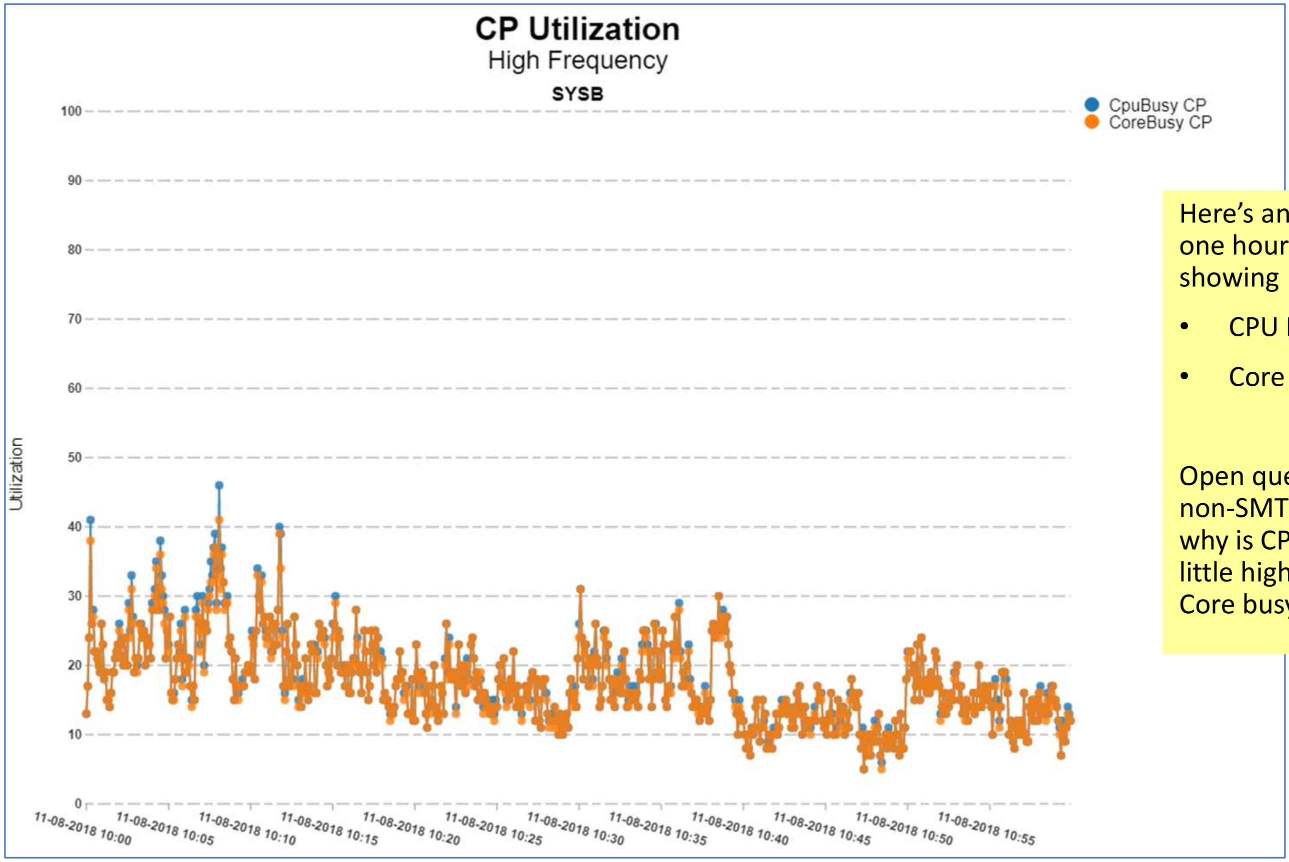
  - NOHFTSINTVL parameter disables the HFTS interval and prevents the collection of type SMF type 98 records

- Can be somewhat voluminous
  - But not as voluminous as the SMF 99 records, and only a fraction of the DB2 or CICS SMF transaction records (i.e. SMF 101 and SMF 110)

Overview of SMF 98 Record cont…

- Examples of measurements include:
  - Processor configuration information, and resultant HiperDispatch pooling
  - Processor utilizations broken down by HiperDispatch pools
  - Insights into latent demand
    - Such as average dispatch per wait (by HiperDispatch pool and engine type)
  - Work unit priority bucket section contains data about work unit priority buckets
    - A priority bucket is a collection of work aggregated across a range of dispatch priorities
    - (1=High, 2=Med, 3=Low, 4=Discretionary)
  - Spin and suspend lock processing insights
    - Additional insights of contributing address spaces
- The true usage of this data is not yet known.
  - It seems IBM put this data for giggles and kicks, but there is good performance analysis usage
  - SMF 98 is sure to provide great insights into processor demands on a much more granular level

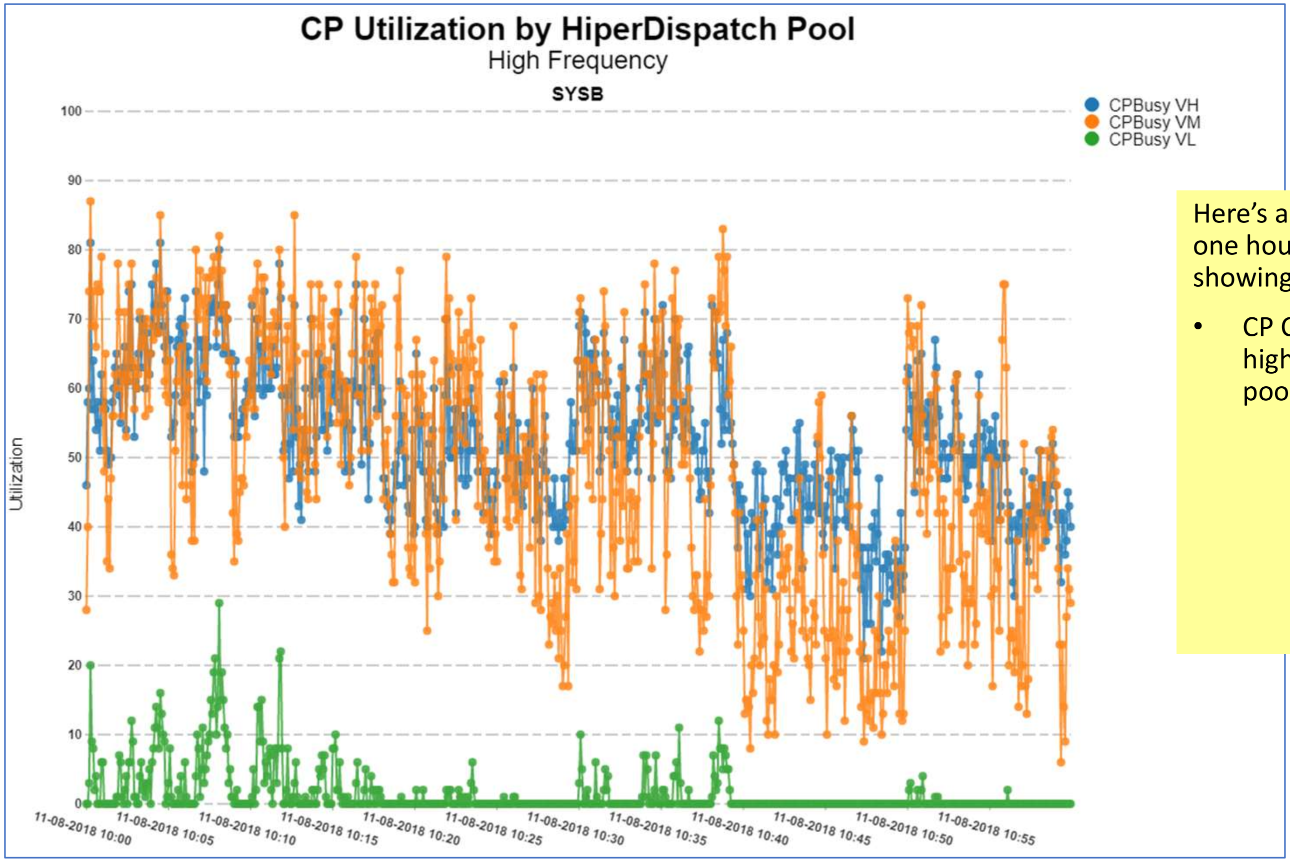
SMF 98 Report Example



Here's an example of one hour of data showing

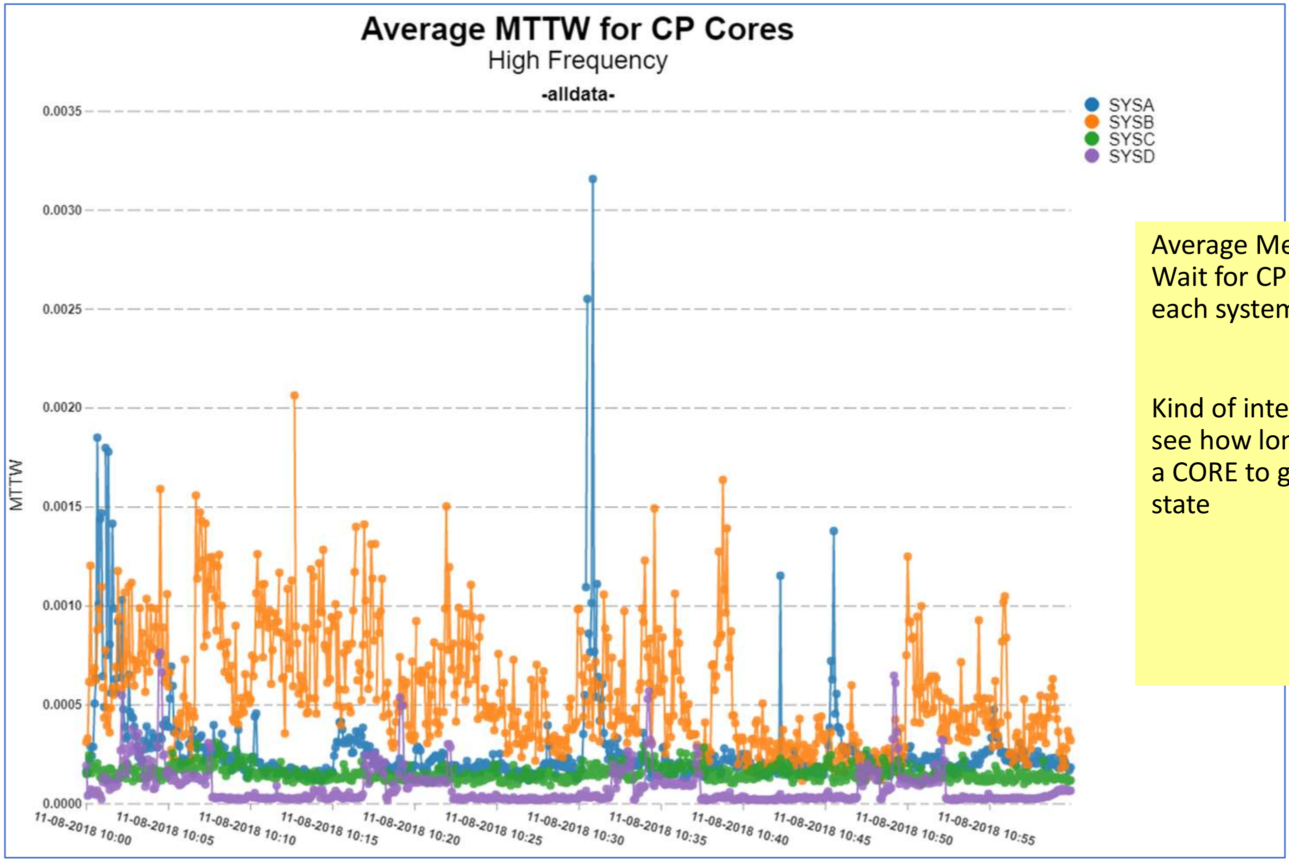
- CPU Busy
- Core Busy

Open question: In a non-SMT environment, why is CP CPU busy a little higher than CP Core busy?



Here's an example of one hour of data showing

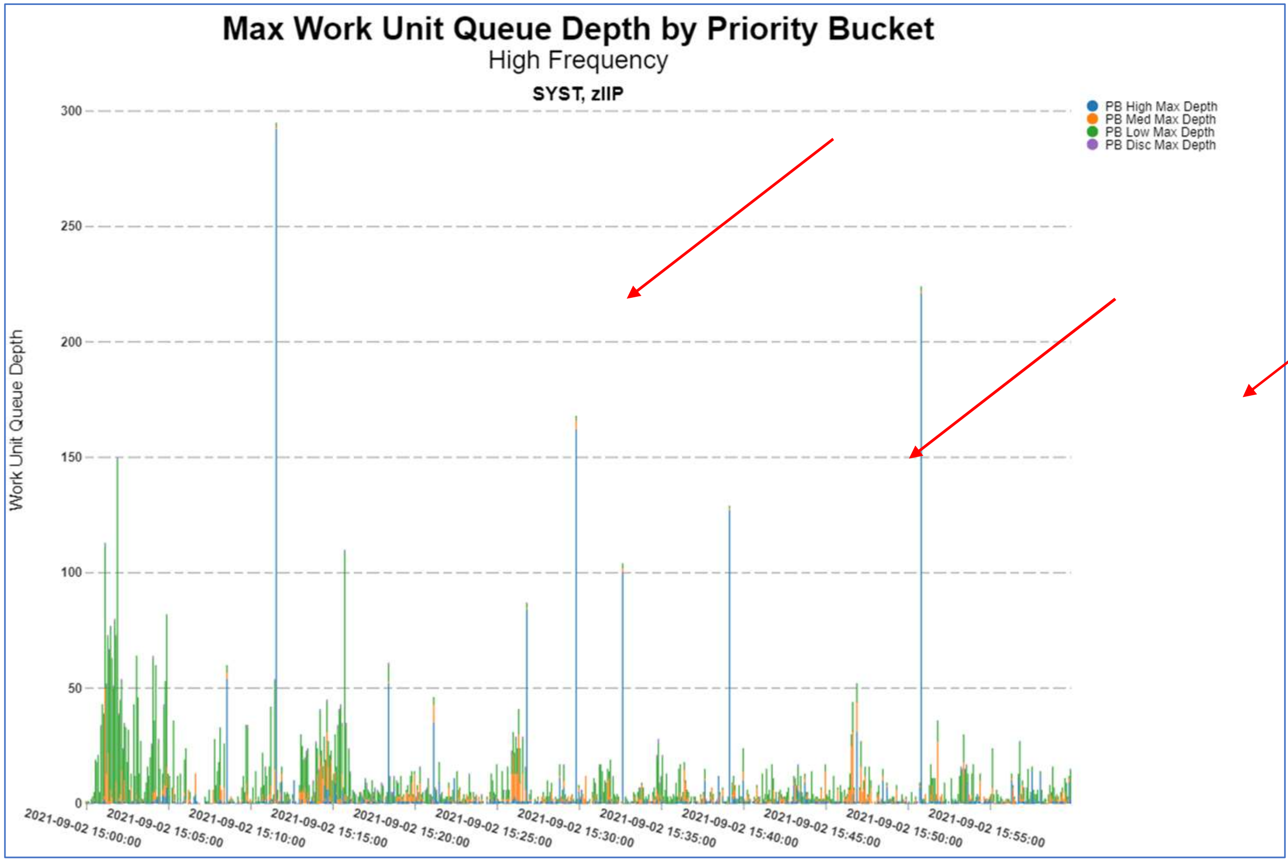
- CP CPU busy by higher dispatch pool



Average Mean Time to Wait for CP Cores for each system

Kind of interesting to see how long it take for a CORE to go into a wait state

Use SMF 98 to look at dispatch queue depths



Priority bucket statistics

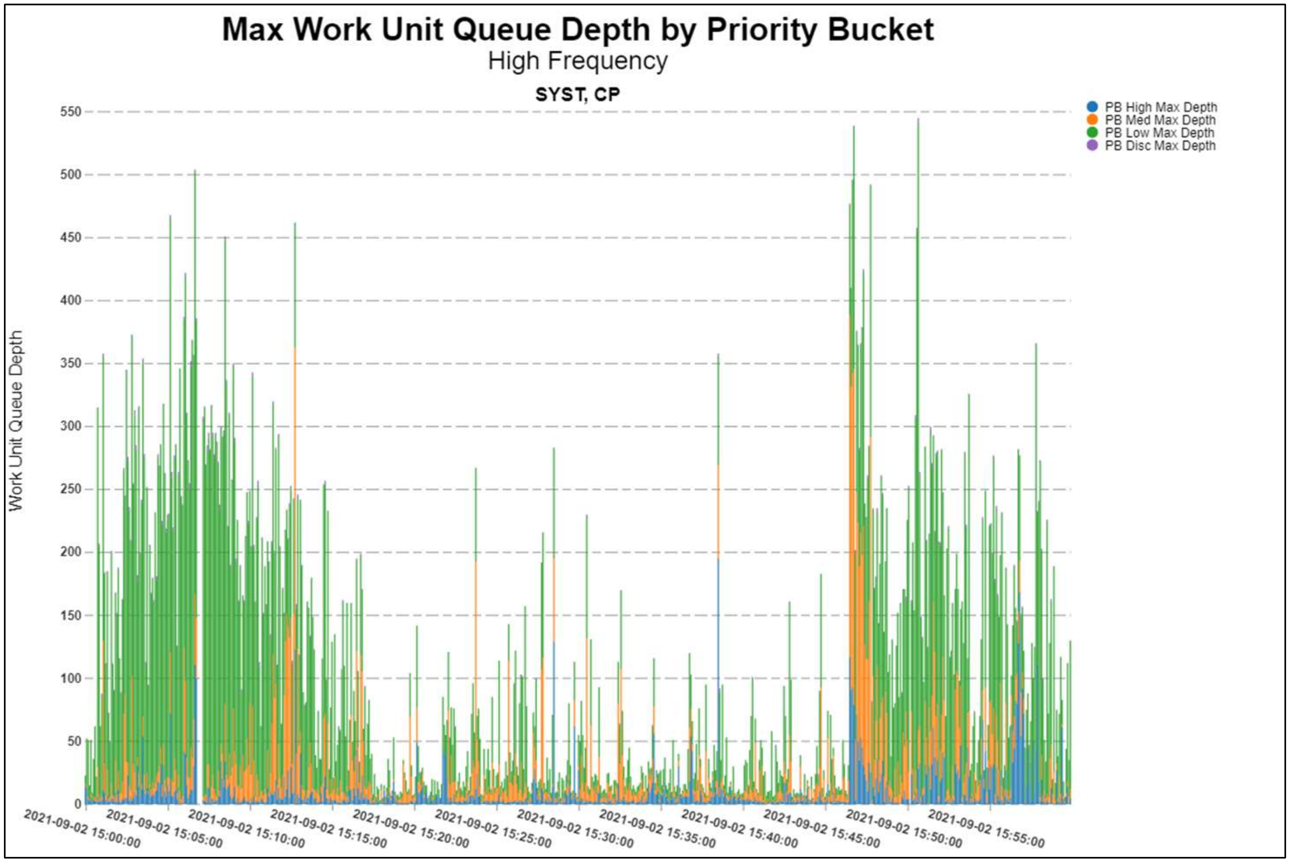
(1=High, 2=Med, 3=Low, 4=Discretionary)

On this system there was a great deal of cross over from the zIIP engines to the CPs engines, but the zIIP utilizations were relatively low. Most of the crossover occurred for high importance workloads.

In this case we see the dispatching queue build up every 5 seconds, and we see there were bursts of zIIP activity in high importance workloads.

This sort of pattern of activity helps us understand that maybe ZIIPAWMT is a good mechanism to throttle back crossover.

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.

How the SMF 98 records can be useful

- Contain low level diagnostic information that persons with detailed knowledge of z/OS internals can obtain detailed data from when a system or resource anomaly is detected.
  - Or another way of saying this….
  - Not sure if this data is usable by the average z/OS performance analyst

- Most of the data is meaningless unless you know what is normal
  - And normal can be different:
    - For different environments
    - Different times of the day
    - Different workloads
    - Different levels of the operating system, hardware, microcode, etc.
    - Different machine configurations

  - So normal in your environment can be different from day to day

- Great input for problem debug
  - But still need quite a lot of knowledge of z/OS internals
  - Little you can do to act on the data real time.

How customers can use the SMF 98 records

- Note that the SMF 98 measurements are not fully understood
  - Even our talks with IBM about these records leads us to believe they do not fully understand either
  - The next few statements may be modified in the future

- As of right now, for the average customer can use the SMF 98 records in the following ways:
  - To maybe report on spikes in CPU usage, queues, delays and events, and other patterns of activity
    - To produce a lot of pretty charts and reports
    - To feed the Splunks and Sparks of the world data to produce pretty charts (but most likely unnecessary)

  - To maybe understand if certain locking events are regularly happening to certain address spaces or software

  - To have some 'gee whiz' numbers and charts

- No doubt there is some interesting data in the SMF 98 records
  - But finding practical day-to-day use is still be explored

SMF 99 Records

Overview of SMF 99 Subtypes

- Subtype 1
  - System level measurement data used for decision input
  - Trace of WLM actions
  - Written every 10 seconds (i.e. policy adjustment interval)
- Subtype 2
  - Service class period measurement data used for decision input
  - Written every 10 seconds (i.e. policy adjustment interval)
- Subtype 3
  - Service class period plot data
  - Written every 10 seconds (i.e. policy adjustment interval)
- Subtype 4
  - Service class device cluster information
  - Written every 10 seconds (i.e. policy adjustment interval)
- Subtype 5
  - Data about monitored address spaces
  - Written every 10 seconds (i.e. policy adjustment interval)

Overview of SMF 99 Subtypes cont…

- Subtype 6
  - Service class period settings and measurements
  - Written every 10 seconds (i.e. policy adjustment interval)
- Subtype 7
  - Enterprise Storage Server ® (ESS) with Parallel Access Volumes (PAVs)
  - Written every 30 seconds (i.e. 3 policy adjustment intervals)
- Subtype 8
  - Information about LPAR CPU management
  - Written every 10 seconds (i.e. policy adjustment interval)
- Subtype 9
  - Information about dynamic channel path management
  - Written every 10 seconds (i.e. policy adjustment interval)
- Subtype 10
  - Information about dynamic processor speed changes
  - Written when speed changes

Overview of SMF 99 Subtypes cont…

- Subtype 11
  - Information about Group Capacity Limits
  - Written every 5 minutes
- Subtype 12
  - HiperDispatch interval data
  - Written every 2 seconds (i.e. policy adjustment interval)
- Subtype 13
  - HiperDispatch IBM internal use only (so undocumented)
  - *And very voluminous!*
- Subtype 14
  - HiperDispatch topology data
  - Written every 5 minutes

SMF 99 Recommendations

- Consider regularly collecting the following SMF 99 subtypes
  - Subtype 6 - Service class period settings and measurements
  - Subtype 11 - Information about Group Capacity Limits
  - Subtype 12 - HiperDispatch interval data
  - Subtype 14 - HiperDispatch topology data

  - Collectively these records typically produce about 40MiB/system/day
  - They contain the most interesting and useful data of the 99s

- Records to collect for problem periods of time, or when doing a study to better understand WLM decision making
  - Subtype 1 - System level measurement and trace data used for decisions
  - Subtype 2 - Service class period measurement data used for decision input
  - Subtype 3 - Service class period plot data
  - Subtype 5 - Data about monitored address spaces

  - Then call Peter Enrico and / or Scott Chapman to process with Pivotor

SMF 99.6

SMF 99.6 Overview

- Subtype 6
  - Service class period settings and measurements
  - Written every 10 seconds (i.e. policy adjustment interval)
  - The purpose of this subtype is to record the WLM controls that are set for each service class period

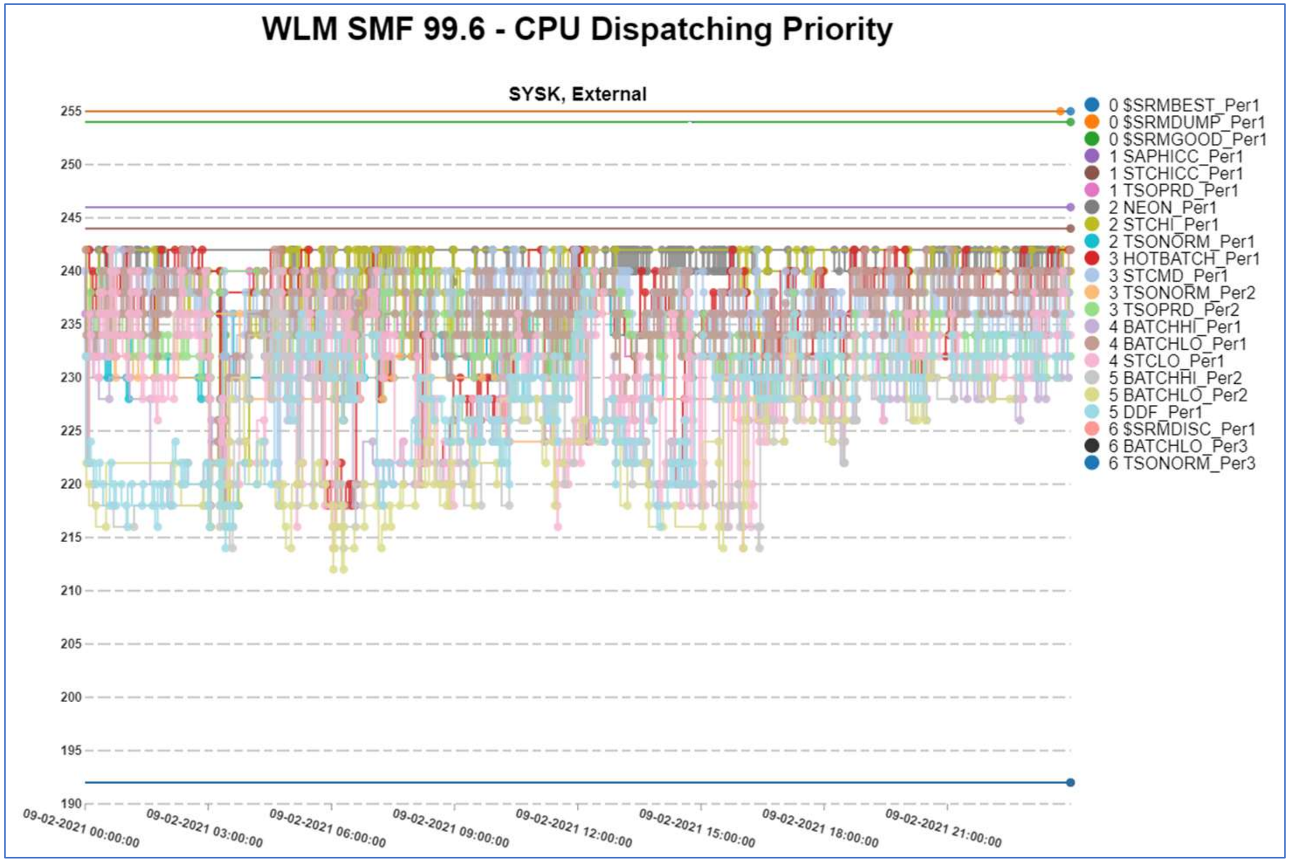
- It is recommended that SMF 99.6 record be turned on
  - Typically, about 40MiB/system/day

- Key data in the SMF 99.6 includes
  - Service class, service class period, and goal information
  - Performance Indexes – both local and Sysplex PIs
  - CPU and I/O dispatching priorities
  - CPU service consumption (CP / zIIP / zAAP)
  - MPL in-targets and out-targets
  - Storage isolation and protection
  - For $SRMSxxx periods – the external service class period(s) served

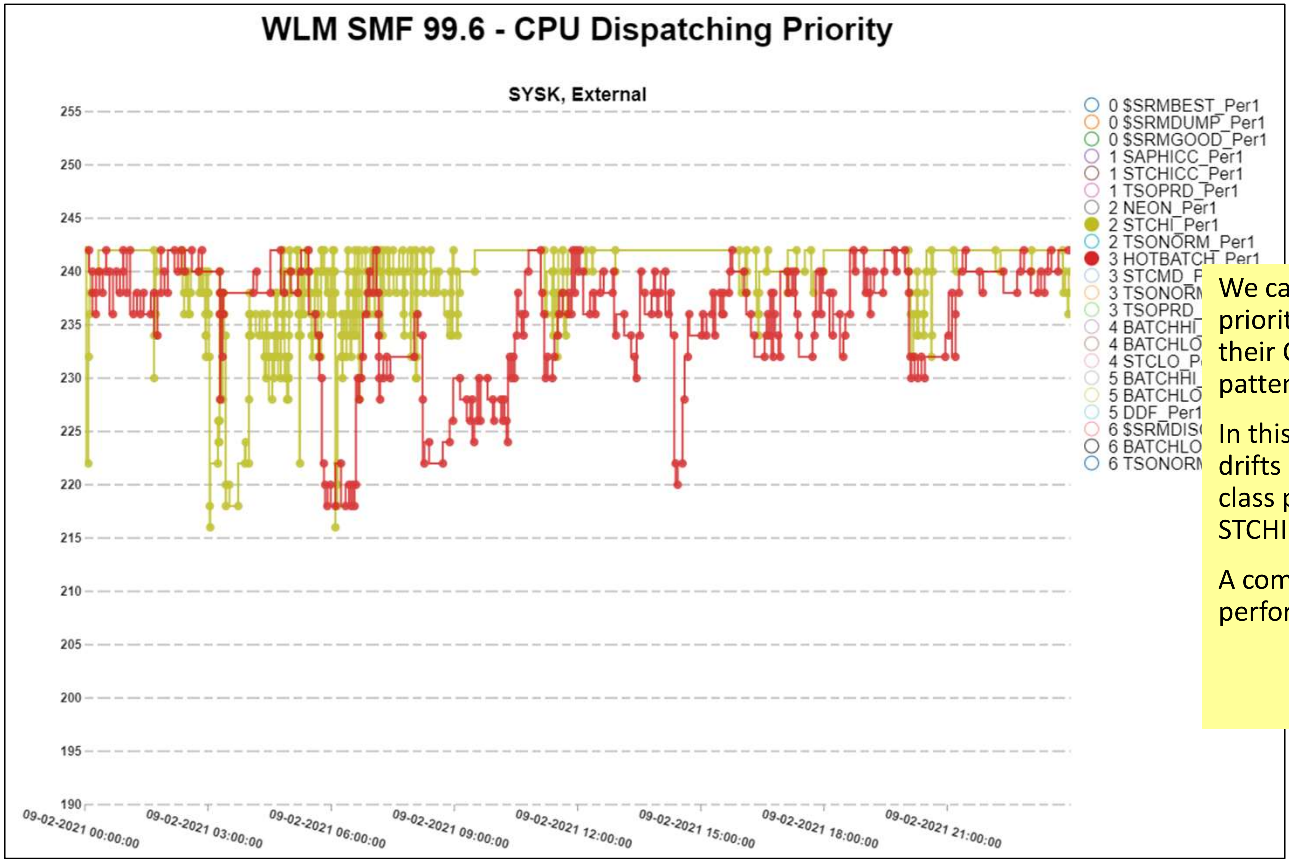
How can the SMF 99.6 record be used?

- The SMF 99.6 record is helpful for answering the following questions:
  - Over time, what is the assigned dispatching priorities of each service class period?
  - How do the priorities change over time?
  - Relative to the goal value and importance level, is the assign priority as desired?
  - How much service is accumulated by each period every 10 seconds?
  - How much service is accumulated at CPU priorities above, below, and at the priority of the service class period being studied?
  - What is the relationship between the local PI and the Sysplex PI?
    - Is the Sysplex PI delaying WLM from helping a period missing its local PI?
  - What is server / served relationship between external periods and internal periods?
  - What types of protections are in place for large storage intensive workloads?

SMF 99.6 CPU Dispatching Priority – Every 10 Seconds



SMF 99.6 CPU Dispatching Priority – Every 10 Seconds

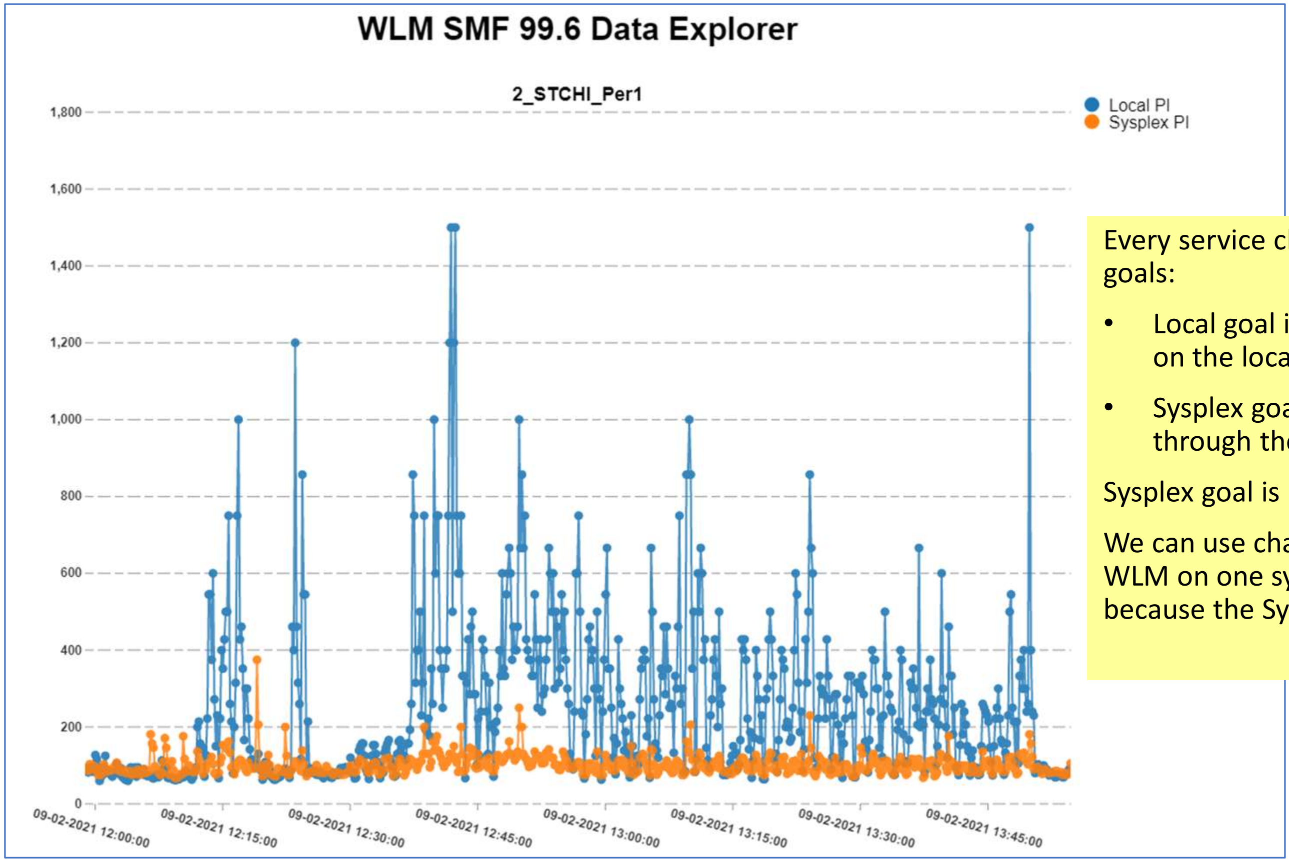


We can look at the dispatching priority of various workloads to see their CPU dispatching priority pattens.

In this example, we see extreme drifts download in CPU DP for service class periods for HOTBATCH and STCHI.

A common objective of z/OS performance tuning is predictability.

SMF 99.6 Sysplex and Local PIs – Every 10 Seconds



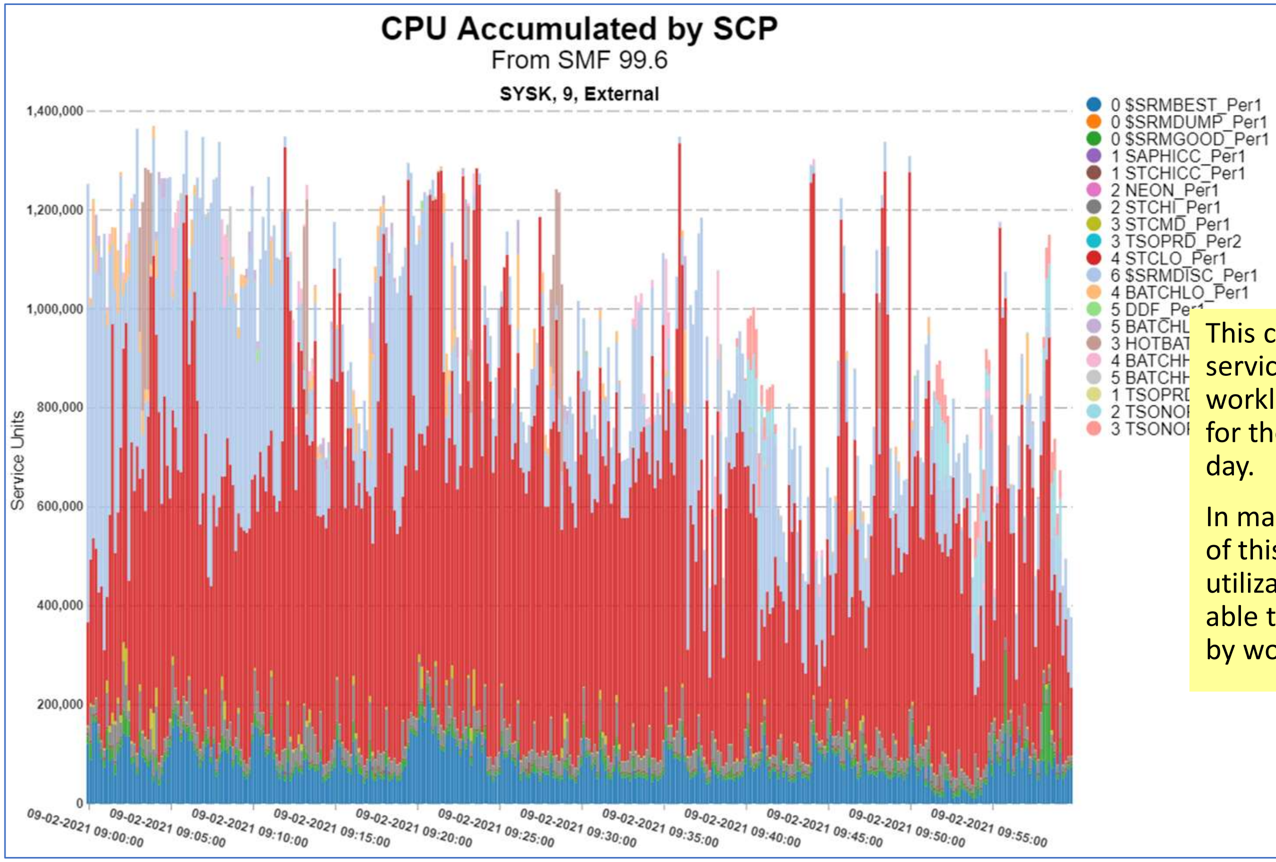
Every service class period actually has 2 goals:

- Local goal is what needs to be achieved on the local system
- Sysplex goal is what needs to be achieved through the Sysplex.

Sysplex goal is preferred over local goal.

We can use charts like this to debug why WLM on one system is not helping STCHI because the Sysplex goal is being met.

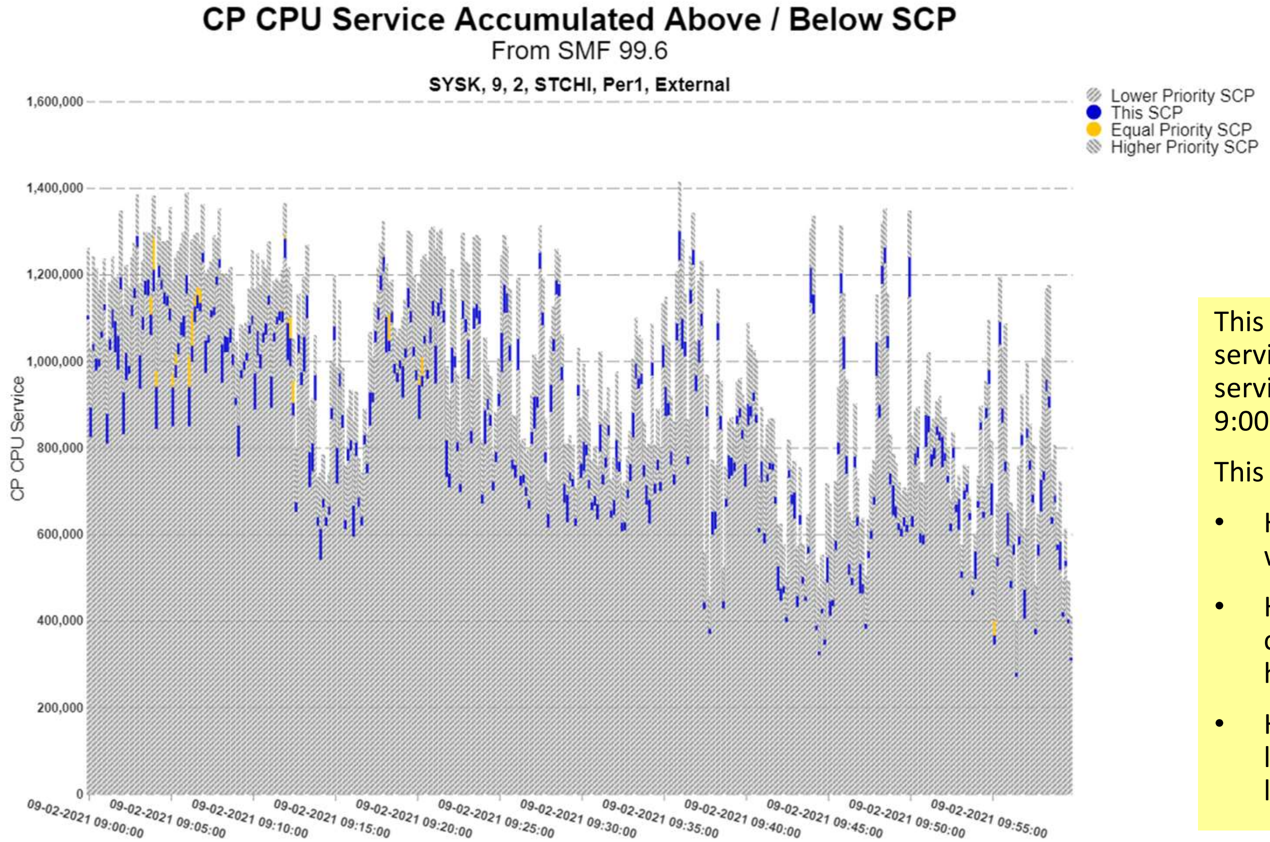
SMF 99.6 Service Consumed by Period – Every 10 Seconds



This chart shows the CPU service consumed by the workloads every 10 seconds for the 9:00am hour of the day.

In many ways, you can think of this chart almost like a CPU utilization chart. You will be able to see bursts of activity by workloads.

SMF 99.6 Service Consumed above / below – Every 10 Seconds

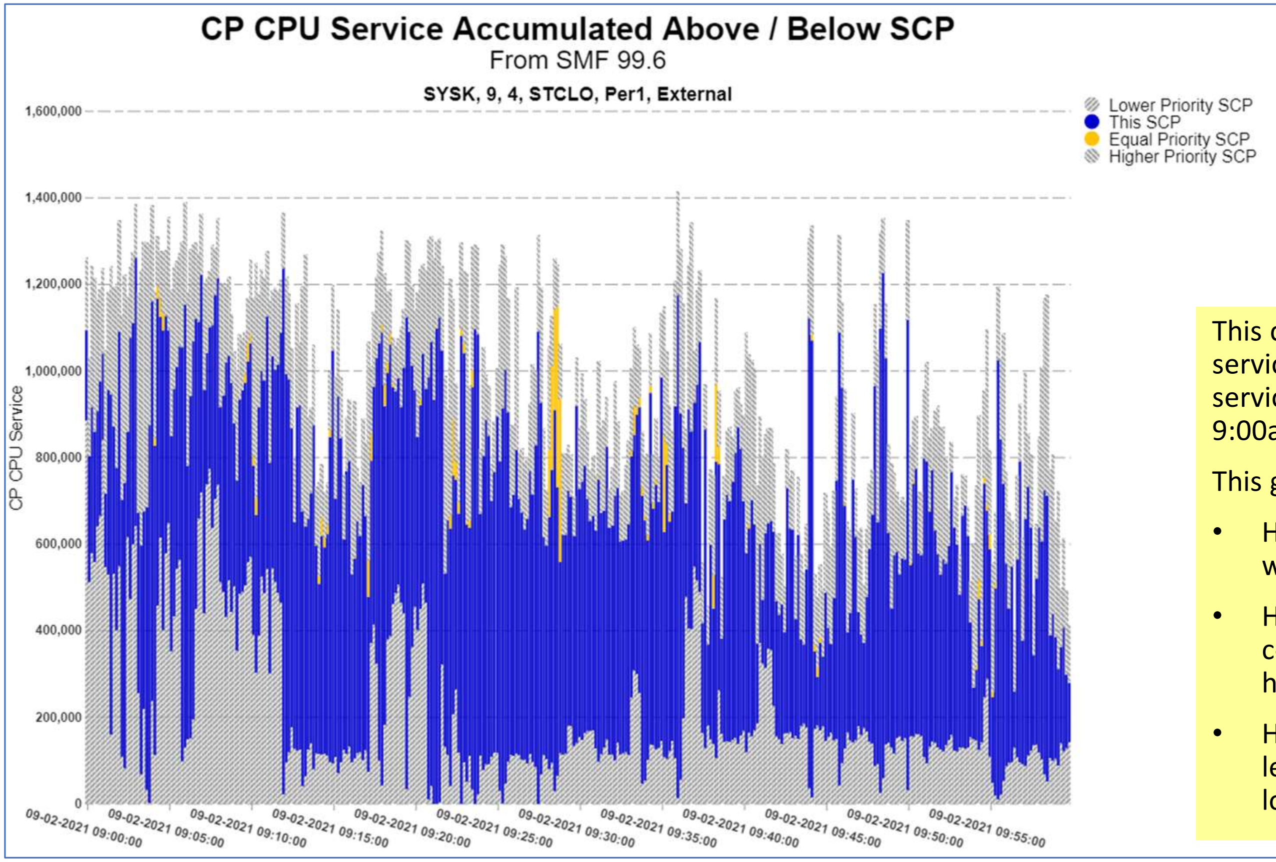


This chart isolates the CPU service consumed for the STCHI service class period for the 9:00am hour.

This gives us a feel for

- How much CPU service a workload consumed
- How much CPU was consumed by work running higher CPU DPs
- How much CPU this workload left for workloads running at lower CPU DPs

SMF 99.6 Service Consumed above / below – Every 10 Seconds

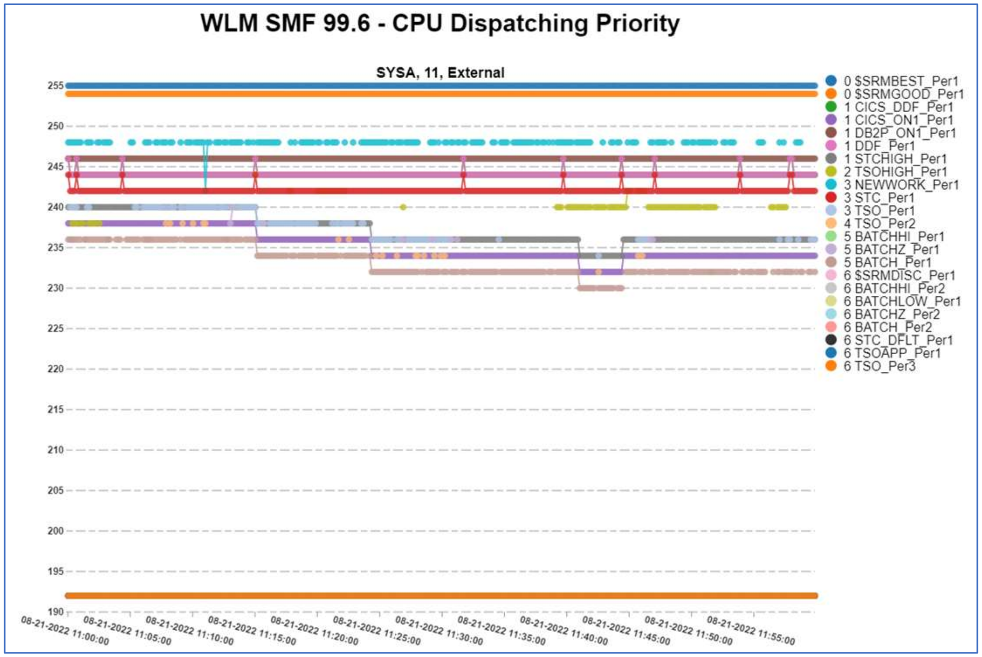


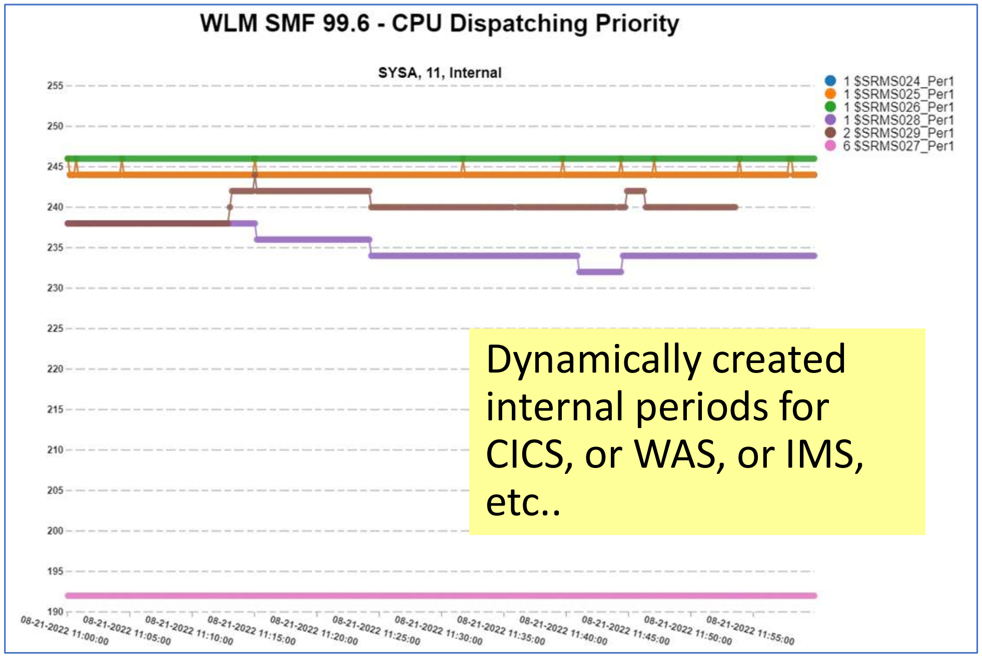
This chart isolates the CPU service consumed for the STCHI service class period for the 9:00am hour.

This gives us a feel for

- How much CPU service a workload consumed
- How much CPU was consumed by work running higher CPU DPs
- How much CPU this workload left for workloads running at lower CPU DPs

Insights into Internal and External Period





SMF 99.12

SMF 99.12 Overview

- Subtype 12
  - HiperDispatch interval data
  - Written every 2 seconds (i.e. HiperDispatch interval)
  - The purpose of this subtype is to record the factors that influence HiperDispatch parking and un-parking of processors

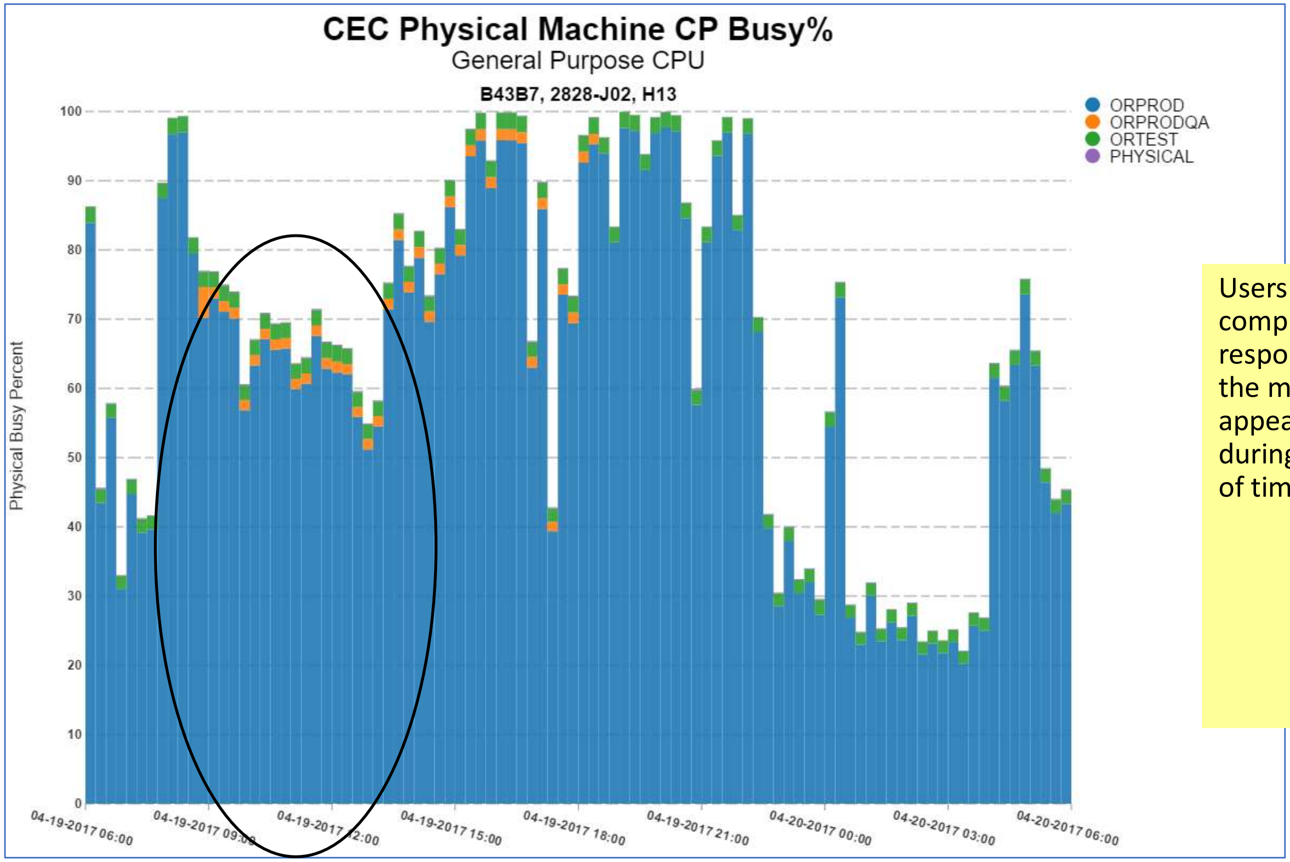
- It is recommended that SMF 99.12 record be turned on

- Key data in the SMF 99.12 includes
  - LPAR level configuration information relevant to HiperDispatch
    - Example: LPAR share, LPAR capacities, SMT enablement, etc.
  - Processor utilizations (current and projected)
  - Pooling of Vertical Highs, Vertical Mediums, Vertical Lows
  - Capacity used / available to each pool (VHs, VMs, VLs)
  - Guaranteed shares to VHs, VMs, VLs
  - CPU displaced by parking and un-parking

Using the SMF 99.12 record

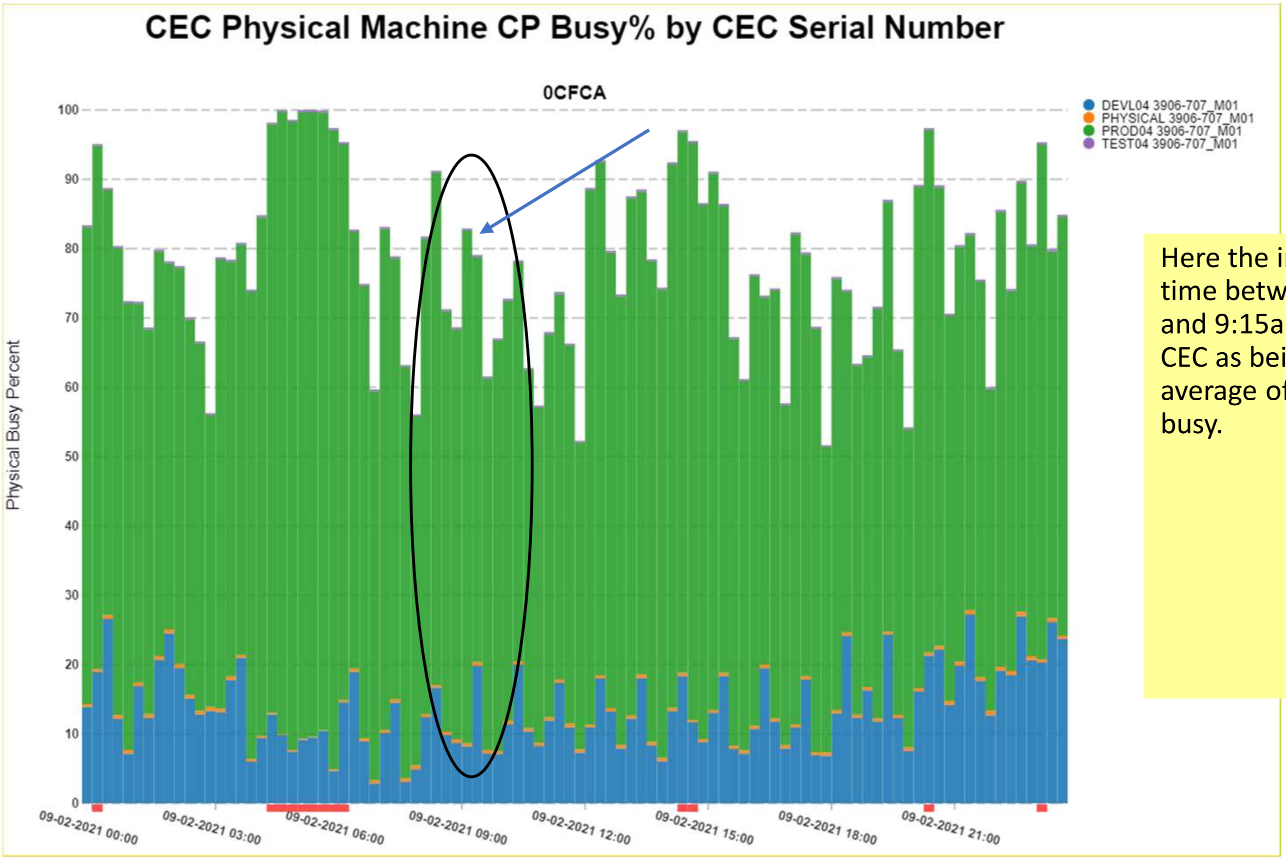
- The SMF 99.12 record is helpful for answering the following questions:
  - Over time, what is the LPAR configuration, and did it change?
  - What is the logical processor pooling for the LPAR?
  - Did the pooling change due to a configuration change or due to capping?
  - What is the parking and un-parking of the logical processors?
  - What is the utilization of the processors?
    - Remember, this is every 2 seconds, so much more granularity than SMF 70 data.
  - What may be inhibiting the un-parking of a processor?
  - What are the effects of capping on the decisions of parking and un-parking processors?

SMF 70 – A look at physical machine utilization

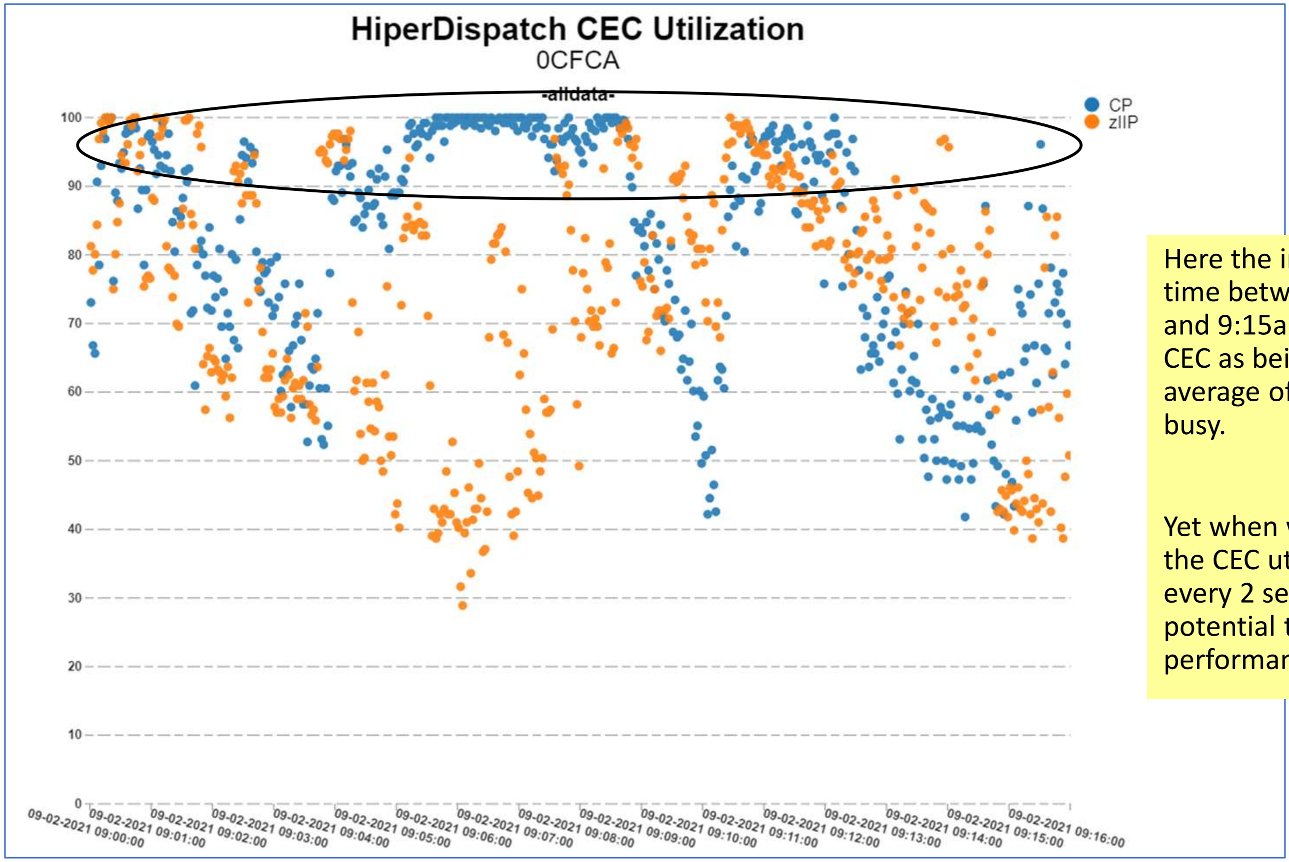


Users are calling up complaining of poor response times, but the machine does not appear very busy during problem period of time.

SMF 70 – A look at physical machine utilization



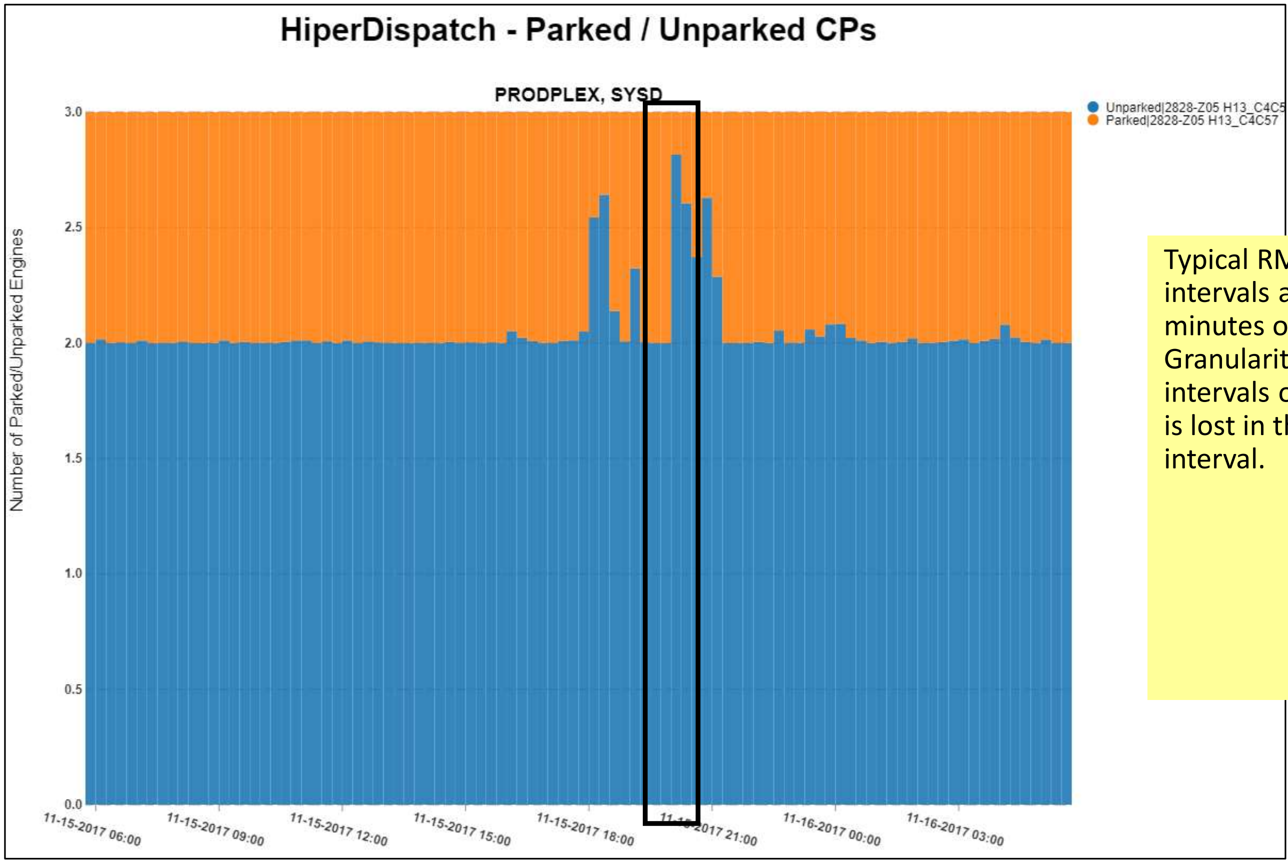
Here the interval of time between 9:00am and 9:15am shows this CEC as being an average of 82.89% busy.



Here the interval of time between 9:00am and 9:15am shows this CEC as being an average of 82.89% busy.

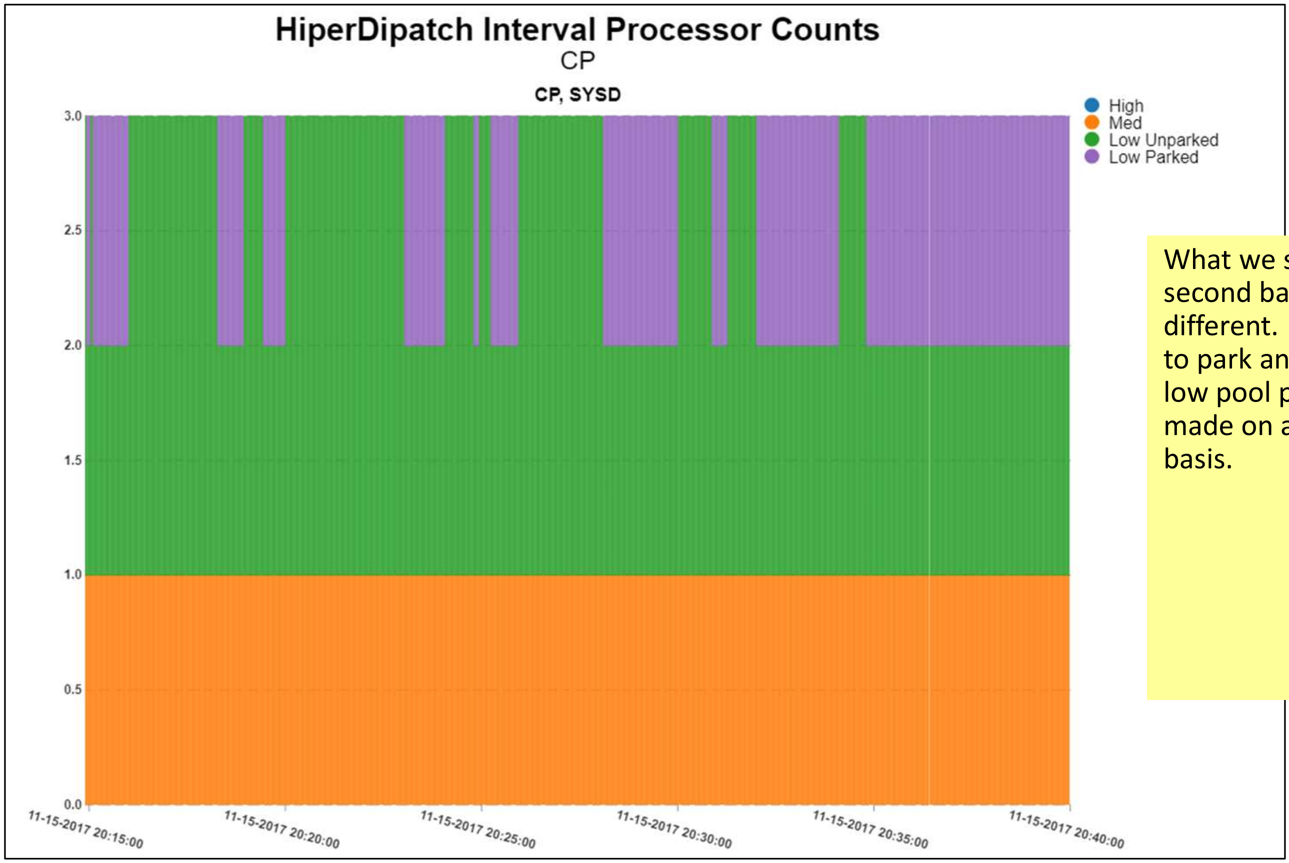
Yet when we look at the CEC utilization every 2 seconds we see potential transient performance issues.

HiperDispatch Pooling, and Parking/Unparking



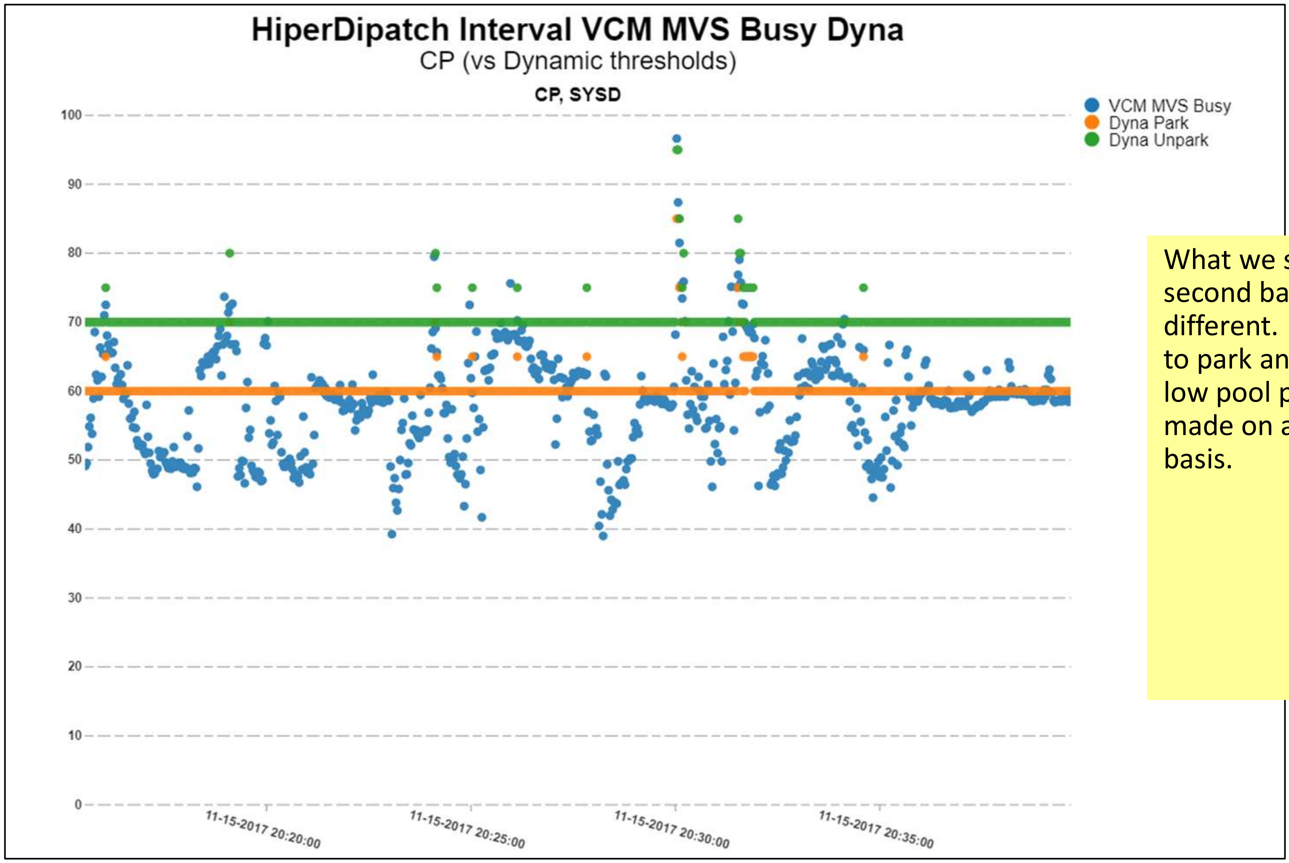
Typical RMF/CMF intervals are 15 minutes or less. Granularity of decision intervals of 2 seconds is lost in the 15 minute interval.

HiperDispatch Pooling, and Parking/Unparking



What we see on a 2 second basis is very different. The decision to park and un-park of low pool processors is made on a 2 second basis.

HiperDispatch Pooling, and Parking/Unparking



What we see on a 2 second basis is very different. The decision to park and un-park of low pool processors is made on a 2 second basis.

SMF 99.1

SMF 99.1 and SMF 99.2 – PA and RA Decisions

- There are two primary phases of WLM algorithms

- Policy Adjustment (PA)
  - Done approximately every 10 seconds (AKA 'PA interval')
  - Objectives include:
    - Summarize state of system and resources
    - Help work meet goals by setting resource controls
    - Housekeep resource controls that may be out of date

- Resource Adjustment (RA)
  - Done approximately every 2 seconds (AKA 'RA interval')
  - Objectives include:
    - improve efficiency of system resources
    - avoided if at the expense of goals

WLM Policy Adjustment – 'The Loop'

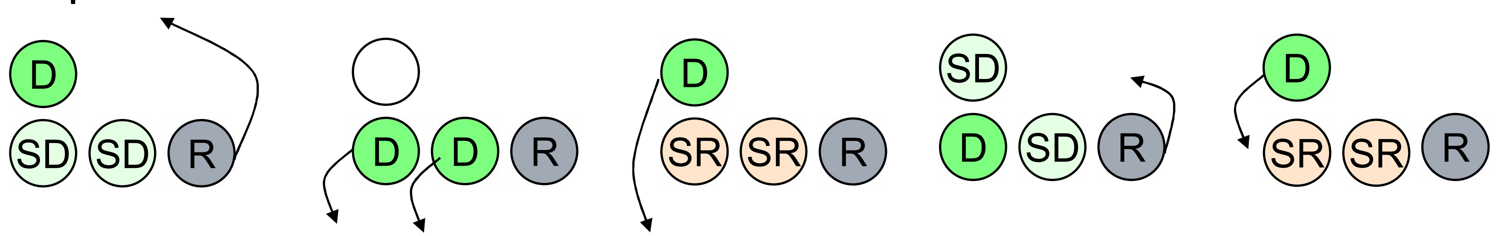
- Summarize data for state of the system and workloads
- Select a receiver period (highest importance missing goal the most)
- Find the receiver's largest bottleneck
  - Determine fix for receiver's bottleneck
    - Determine if needed resources can be gotten from unused resources
    - Find donor(s) of resource that receiver needs
    - Assess effect of reallocating resources from donor(s) to receivers
    - If allocation has both net and receiver value
      Then commit change
      Else don't make change
  - If reallocation was done
    then jump to Exit and allow change to be absorbed
  - If reallocation was not done
    then try to fix receiver's next largest bottleneck
- If cannot help receiver
  then look for next receiver (highest importance missing goal the most)
- Exit
  - Housekeep current set of controls

Receivers and Donors

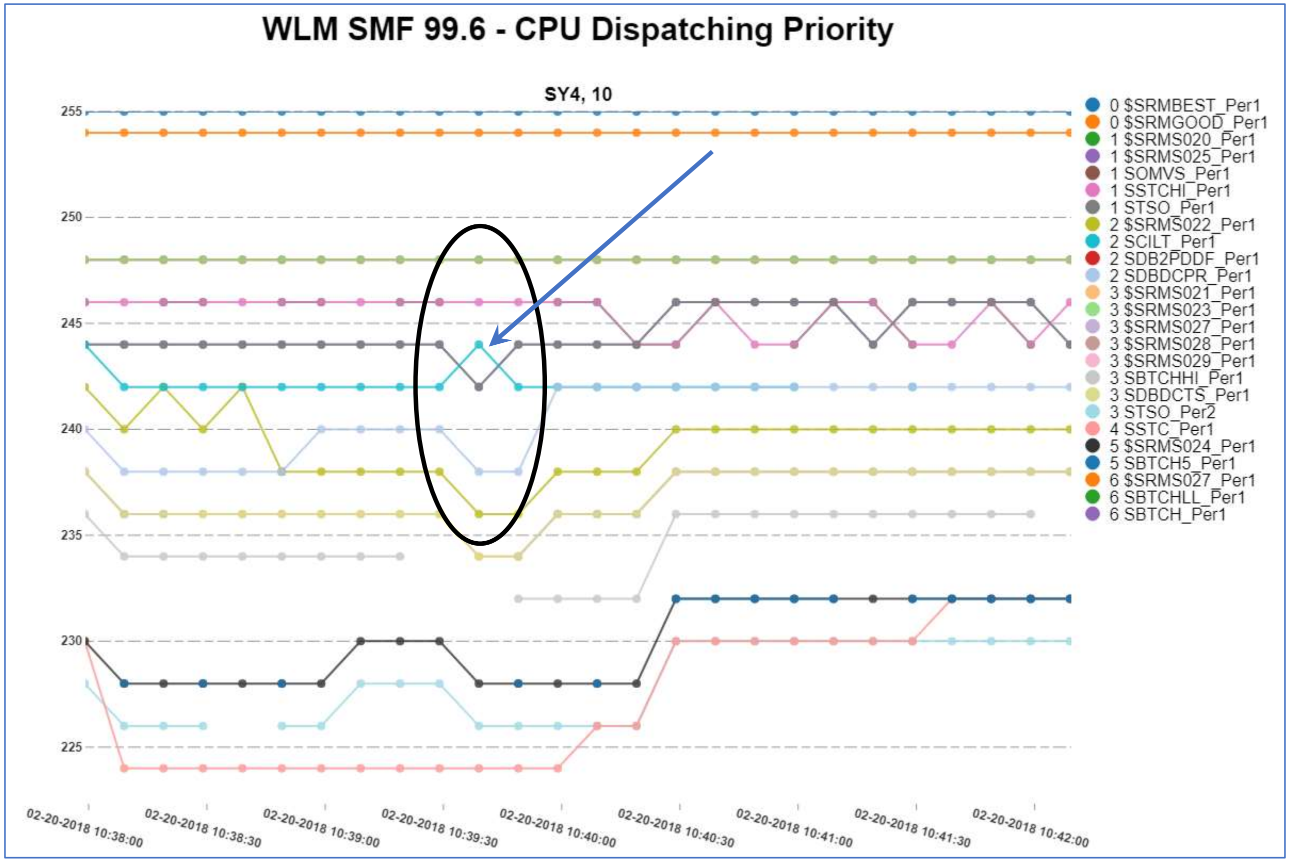
- Receiver
  - Service class period to potentially 'receive' resources
  - WLM will help only one receiver during each policy adjustment interval
    - Goal Receiver - Period with goal that needs help
    - Resource Receiver - Period to give the resources to in order to help the goal receiver
    - Secondary Receiver - Period helped indirectly due to an action to help the goal receiver
- Donor
  - Service class period to potentially 'donate' resources to help receiver
  - WLM may take from multiple donors during each policy adjustment interval
    - Goal Donor - Period whose goals may be impacted by resource donation
    - Resource Donor - Period to donate resources
    - Secondary Donor - Period that donates indirectly when receiver is helped

Example of WLM Decisions – CPU DP

- Dispatching priority adjustments
  - Objective: Increase Receiver's CPU using, or decrease Receiver's CPU delay
  - Interesting concepts:
    - Wait-to-Using ratio - ratio of CPU delay samples to CPU using samples (change in ratio used to determine change in CPU delay)
    - Maximum demand
      - Theoretical maximum percentage of total processor time a period can consume if it had no CPU delay
    - Achievable maximum demand
      - Percentage of total processor time a service period is projected to consume, taking into account demand of all higher work
  - Some possible actions

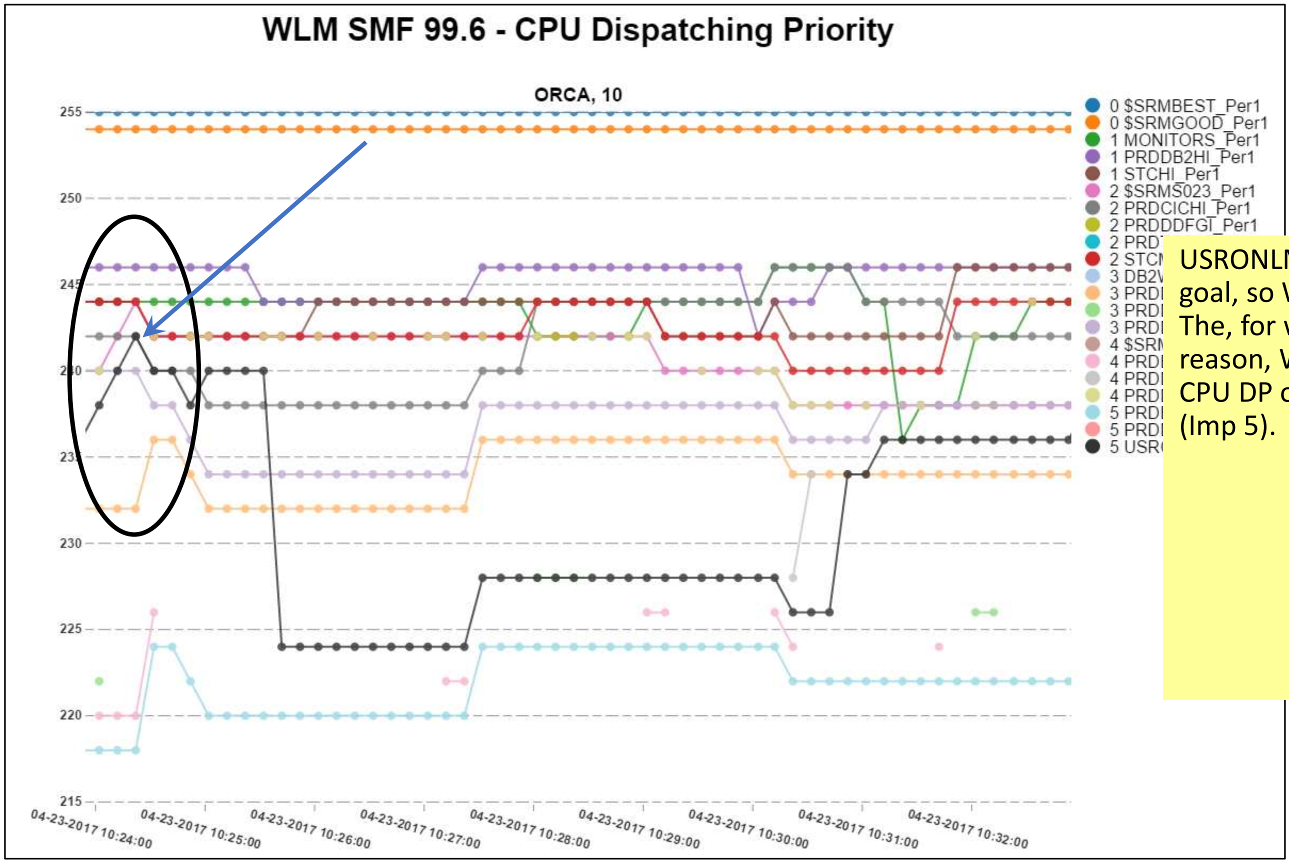


SMF 99.6 CPU Dispatching Priority – Every 10 Seconds



SMF 99.1
- Example of WLM Actions Trace

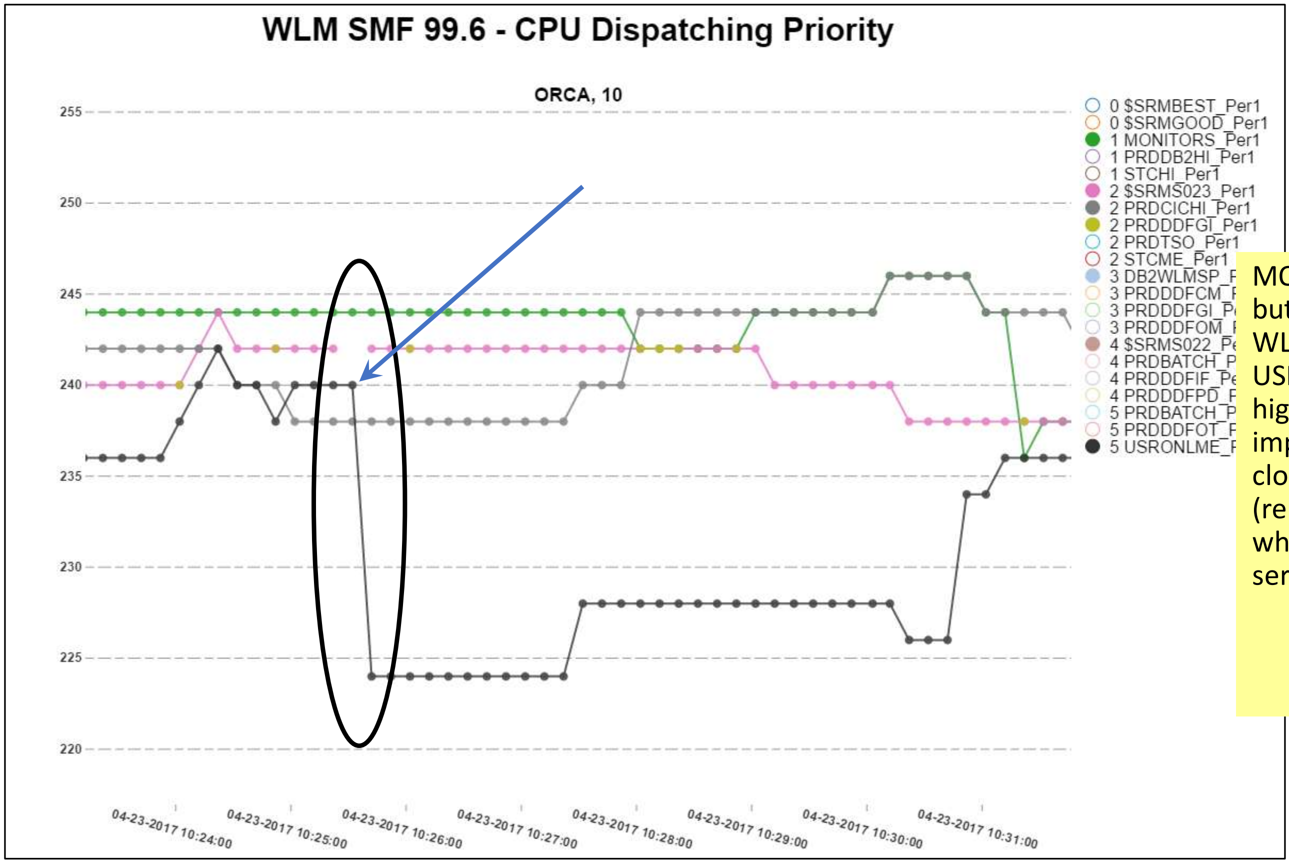
system	Time	PA Int	RA Int	Code	Code	Explain	Local P	Sysple	Perioc	Service Class
SY4	10:39:39	142	173	270	PA_REC_CAND	Policy adjustment, receiver candidate selected.	50	50	1	SCILT
SY4	10:39:39	142	173	308	PA_DONOR_PERIOD	Policy adjustment, donor period.	0.64	0.94	1	STSO
SY4	10:39:39	142	173	308	PA_DONOR_PERIOD	Policy adjustment, donor period.	0.78	1.09	1	SSTCHI
SY4	10:39:39	142	173	525	HSK_UNBUNCH_PRTY	Housekeeping, unbunch priorities.	0.01	0.14	1	$SRMS024
SY4	10:39:39	142	173	525	HSK_UNBUNCH_PRTY	Housekeeping, unbunch priorities.	0.01	0.14	1	SBTCH5
SY4	10:39:39	142	173	525	HSK_UNBUNCH_PRTY	Housekeeping, unbunch priorities.	0.5	5.5	1	SOMVS
SY4	10:39:39	142	173	525	HSK_UNBUNCH_PRTY	Housekeeping, unbunch priorities.	0.54	0.9	2	STSO
SY4	10:39:39	142	173	525	HSK_UNBUNCH_PRTY	Housekeeping, unbunch priorities.	0.64	0.94	1	$SRMS025
SY4	10:39:39	142	173	525	HSK_UNBUNCH_PRTY	Housekeeping, unbunch priorities.	0.64	0.94	1	STSO
SY4	10:39:39	142	173	525	HSK_UNBUNCH_PRTY	Housekeeping, unbunch priorities.	0.74	1.09	1	$SRMS021
SY4	10:39:39	142	173	525	HSK_UNBUNCH_PRTY	Housekeeping, unbunch priorities.	0.74	1.09	1	SDBDCTS
SY4	10:39:39	142	173	525	HSK_UNBUNCH_PRTY	Housekeeping, unbunch priorities.	0.76	0.97	1	SBTCHHI
SY4	10:39:39	142	173	525	HSK_UNBUNCH_PRTY	Housekeeping, unbunch priorities.	1.1	0.8	1	$SRMS022
SY4	10:39:39	142	173	525	HSK_UNBUNCH_PRTY	Housekeeping, unbunch priorities.	50	50	1	SCILT
SY4	10:39:39	142	173	525	HSK_UNBUNCH_PRTY	Housekeeping, unbunch priorities.	60	0.85	1	SDBDCPR
SY4	10:39:39	142	173	530	PA_PMDO_DON	Policy adjustment, assess moving primary processor donor down to occupied priority.	0.64	0.94	1	STSO
SY4	10:39:39	142	173	530	PA_PMDO_DON	Policy adjustment, assess moving primary processor donor down to occupied priority.	0.78	1.09	1	SSTCHI
SY4	10:39:39	142	173	531	PA_PCC_DON_VIOLTN	Policy adjustment, moving the donor to the receivers priority violates CPU critical rules.	0.78	1.09	1	SSTCHI
SY4	10:39:39	142	173	532	PA_PCC_BLKR_IS_DON	Policy adjustment, cannot move the blocker up because it is the donor.	0.78	1.09	1	SSTCHI
SY4	10:39:39	142	173	580	PA_PMD_SEC_DON	Policy adjustment, assess moving secondary processor donor down.	0.64	0.94	1	$SRMS025
SY4	10:39:39	142	173	580	PA_PMD_SEC_DON	Policy adjustment, assess moving secondary processor donor down.	0.78	1.09	1	$SRMS020
SY4	10:39:39	142	173	620	PA_PMUO_REC	Policy adjustment, assess moving primary processor receiver up to occupied priority.	50	50	1	SCILT
SY4	10:39:39	142	173	635	PA_PMUUB_REC	Policy adjustment, assess moving primary processor receiver up to unoccupied priority between donor and receiver's current priorities.	50	50	1	SCILT
SY4	10:39:39	142	173	750	PA_PRO_INCP_REC	Policy adjustment, increase priority for receiver.	50	50	1	SCILT



USRONLME misses goal, so WLM helps it. The, for what ever reason, WLM raises CPU DP of USRONLME (Imp 5).

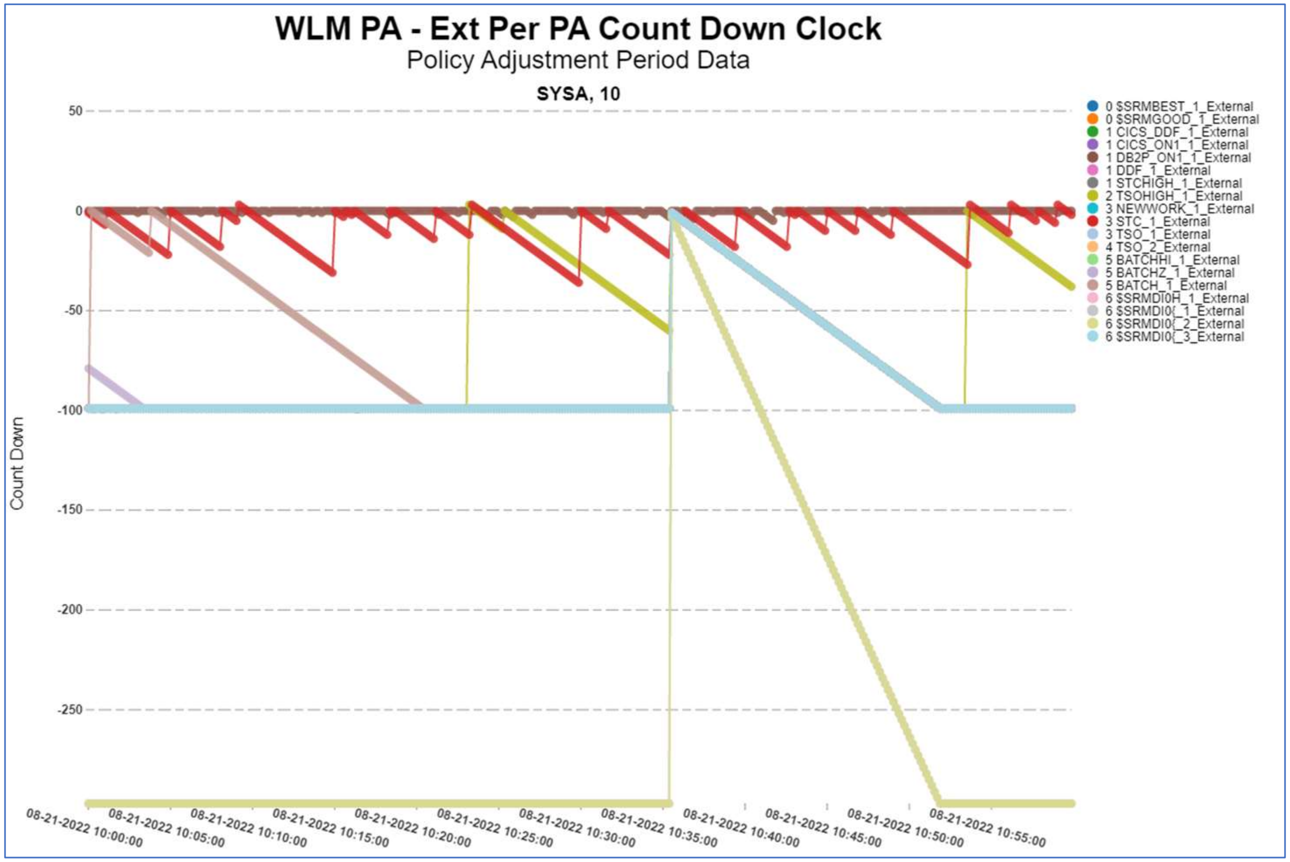
SMF 99.1
- Example of WLM Actions Trace

SMFDateTime	PA Inteval	RA Interval	Trace Code	Code	Job	Local PI	Sysplex PI	Service Class	Period
4/23/17 10:24:03 AM	175	124	270	PA_REC_CAND		131	131	USRONLME	1
4/23/17 10:24:03 AM	175	124	975	PA_SDO_DONFAIL_SPC		110	110	PRDDDFGI	1
4/23/17 10:24:03 AM	175	124	975	PA_SDO_DONFAIL_SPC		70	70	PRDDDFOM	1
4/23/17 10:24:03 AM	175	124	975	PA_SDO_DONFAIL_SPC		27	27	STCME	1
4/23/17 10:24:03 AM	175	124	975	PA_SDO_DONFAIL_SPC		110	110	PRDDDFPD	1
4/23/17 10:24:03 AM	175	124	308	PA_DONOR_PERIOD		40	40	STCHI	1
4/23/17 10:24:03 AM	175	124	880	PA_PRO_RDON_CAND		40	40	STCHI	1
4/23/17 10:24:03 AM	175	124	620	PA_PMUO_REC		131	131	USRONLME	1
4/23/17 10:24:03 AM	175	124	620	PA_PMUO_REC		131	131	USRONLME	1
4/23/17 10:24:03 AM	175	124	620	PA_PMUO_REC		131	131	USRONLME	1
4/23/17 10:24:03 AM	175	124	651	PA_PMU_SPC_NXT_DP		110	110	PRDDDFPD	1
4/23/17 10:24:03 AM	175	124	940	PA_PRO_UNC_DON		40	40	STCHI	1
4/23/17 10:24:03 AM	175	124	940	PA_PRO_UNC_DON		40	40	STCHI	1
4/23/17 10:24:03 AM	175	124	940	PA_PRO_UNC_DON		40	40	STCHI	1
4/23/17 10:24:03 AM	175	124	740	PA_PRO_INCP_DON		110	110	PRDDDFPD	1
4/23/17 10:24:03 AM	175	124	740	PA_PRO_INCP_DON		110	110	PRDDDFPD	1
4/23/17 10:24:03 AM	175	124	740	PA_PRO_INCP_DON		110	110	PRDDDFPD	1
4/23/17 10:24:03 AM	175	124	780	PA_PRO_INCP_SC		110	110	PRDDDFPD	1
4/23/17 10:24:03 AM	175	124	780	PA_PRO_INCP_SC		110	110	PRDDDFPD	1
4/23/17 10:24:03 AM	175	124	780	PA_PRO_INCP_SC		110	110	PRDDDFPD	1
4/23/17 10:24:03 AM	175	124	750	PA_PRO_INCP_REC		113	113	USRONLME	1
4/23/17 10:24:03 AM	175	124	750	PA_PRO_INCP_REC		113	113	USRONLME	1
4/23/17 10:24:03 AM	175	124	750	PA_PRO_INCP_REC		113	113	USRONLME	1



MONITORS missing goal, but WLM cannot help. WLM then notices that USRONLME is running high relative to its importance level and too close to DB2SPWLMSP (related to $SRMS023) which probably are serving PRDCICHI

Does WLM stop trying? No!

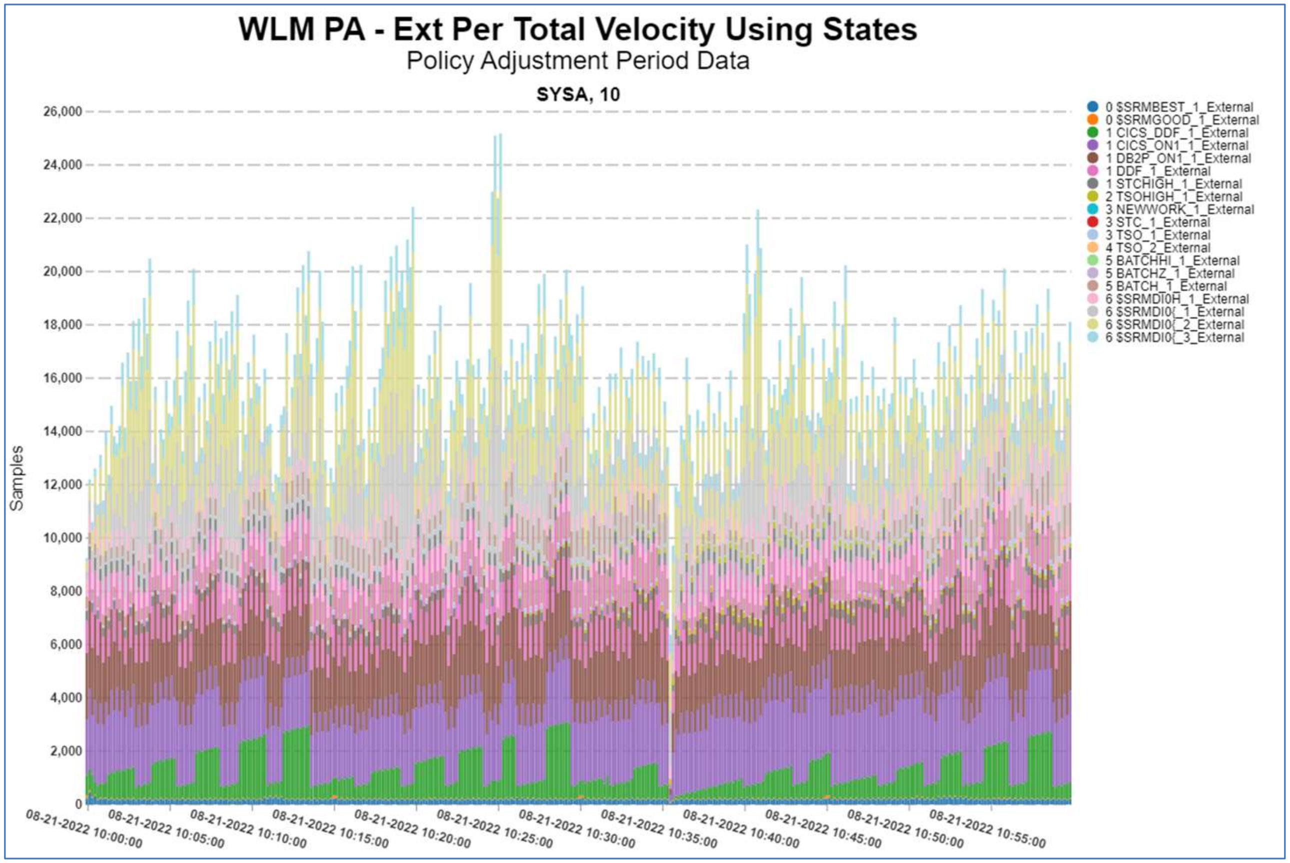


If WLM cannot meet the goal for work it never stops 'trying'.

There are cases when WLM may 'ignore' work for up to 30 seconds.

This chart shows the countdown chart for each period.

Lots of Samples and CPU data from history

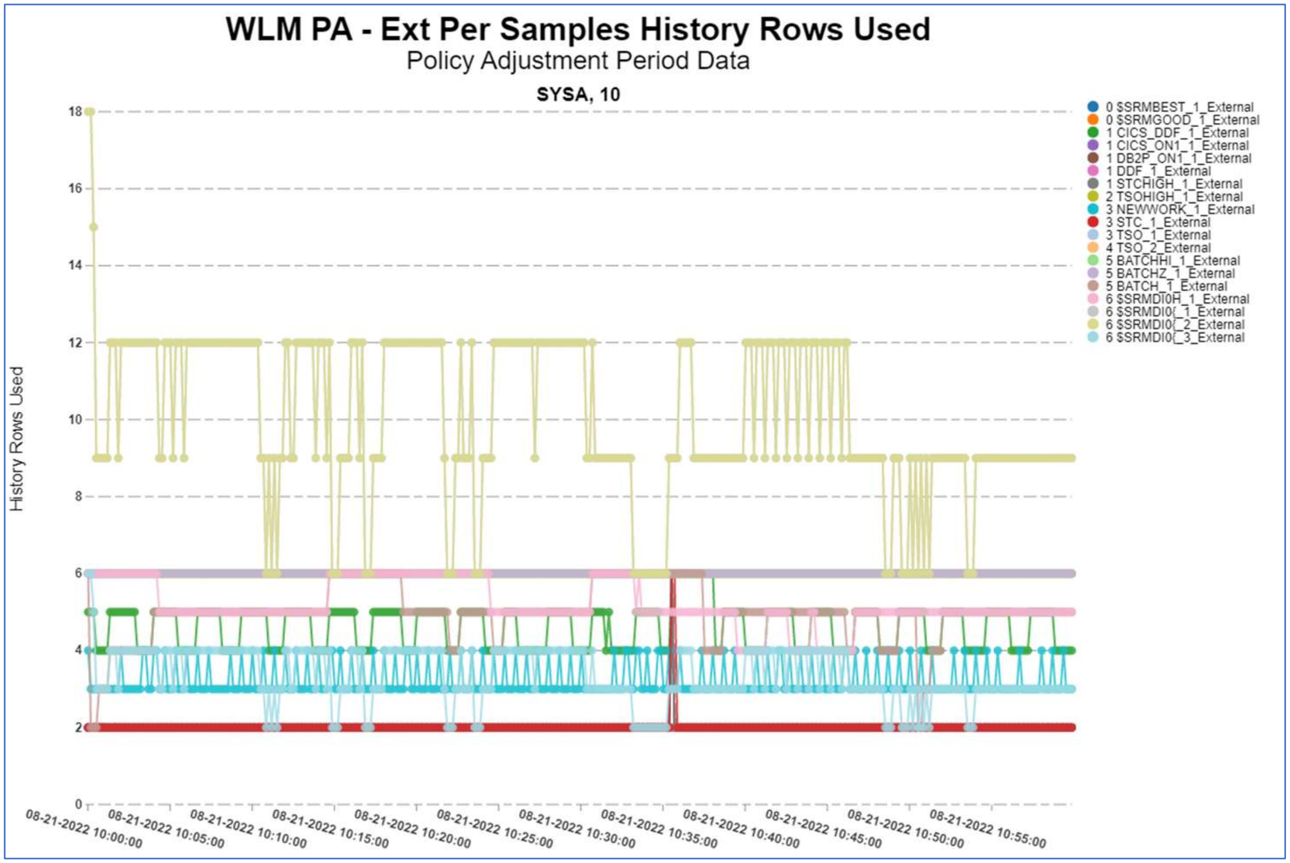


SMF 99.2 shows Using and delay samples on a 10 second basis.

BEWARE!

The samples reported are not just what was accumulated over the last 10 seconds.

Instead, samples based on rows of history used.



SMF 99.2 shows Using and delay samples on a 10 second basis.

BEWARE!

This chart shows the number of rows used when accumulating sample history

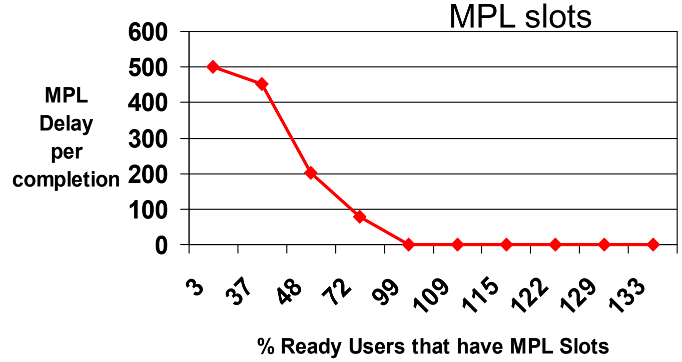
SMF 99.3

WLM maintains a series of plots

- Plots used to track how well work is being processed
- Some of the plots include
  - Period Paging Rate Plot
  - Period MPL Delay Plot
  - Period Ready User Average Plot
  - Period Swap Delay Plot
  - Proportionate Aggregate Speed Plot
  - Queue delay Plot
  - Queue ready user average Plot
  - Active server instance Plot
  - Others…

MPL Plot Example:

- shows how response time may improve by increasing MPL slots
- shows how response time may degrade by reducing MPL slots



Summary – SMF 98 and 99s

- There is a lot more in the SMF 98 and 99 records
  - And I hope I gave you the spirit of what is in there

- There is lots of great information in the SMF 98 and 99 records
  - But you need some knowledge of z/OS internals
  - And a good understanding of WLM

- Recommendation
  - It is probably a waste of processing to process this data on a regular basis
    - Unless you want a lot of pretty charts
  - However, during performance problem debug they can be extremely valuable

- But ALWAYS free to ask me to look at the SMF 98 and 99 data with you
  - Email me: peter.enrico@epstrategies.com
  - We will walk through the data together

Questions?