Understanding Page Faults and Their Influence on Uncaptured Time

Peter Enrico

Email: Peter.Enrico@EPStrategies.com

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

Creators of Pivotor®

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:

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

Abstract

- **Understanding Page Faults and Their Influence on Uncaptured Time**

- This webinar is a back-to-basic webinar about memory management and page faults. Recently, there has been talk about the influence of first reference page faults and their influence on uncaptured time and capture ratios. Since uncaptured time translates to MSU consumption, it is always interesting to explore what contributes to these MSUs. Besides, what is the difference between a demand page fault and a first reference page fault? After attending this webinar with ***Peter Enrico***, you will be more familiar with z/OS memory management, page faults, and uncaptured times.

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
  - February 19-23, 2024
- Parallel Sysplex and z/OS Performance Tuning
  - August 20-21, 2024
- Essential z/OS Performance Tuning
  - October 7-11, 2024
- Also… please make sure you are signed up for our free monthly z/OS educational webinars! (email contact@epstrategies.com)

Like what you see?

- Free z/OS Performance Educational webinars!
  - The titles for our Summer / Fall 2024 webinars are as follows:
    - ✓ *What a z/OS Guy Learned About AWS in 10 Years*
    - ✓ *Advantages of Multiple Period Service Classes*
    - ✓ *Understanding z/OS Connect Measurements*
    - ✓ *WLM and SMF 99.1 – System Measurements Deeper Dive*
    - ✓ *WLM and SMF 99.2 – Service Class Period Measurements Deeper Dive*
    - ✓ *Optimizing Performance at the Speed of Light: Why I/O Avoidance is Even More Important Today*
    - ✓ *Understanding MVS Busy % versus LPAR Busy % versus Physical Busy %*
    - ✓ *Rethinking IBM Software Cost Management Under Tailored Fit Pricing*
    - *Understanding Page Faults and Their Influence on Uncaptured Time*
    - *Response Time Goals: Average or Percentiles?*
    - *Understanding and Using Enclave*

- 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?

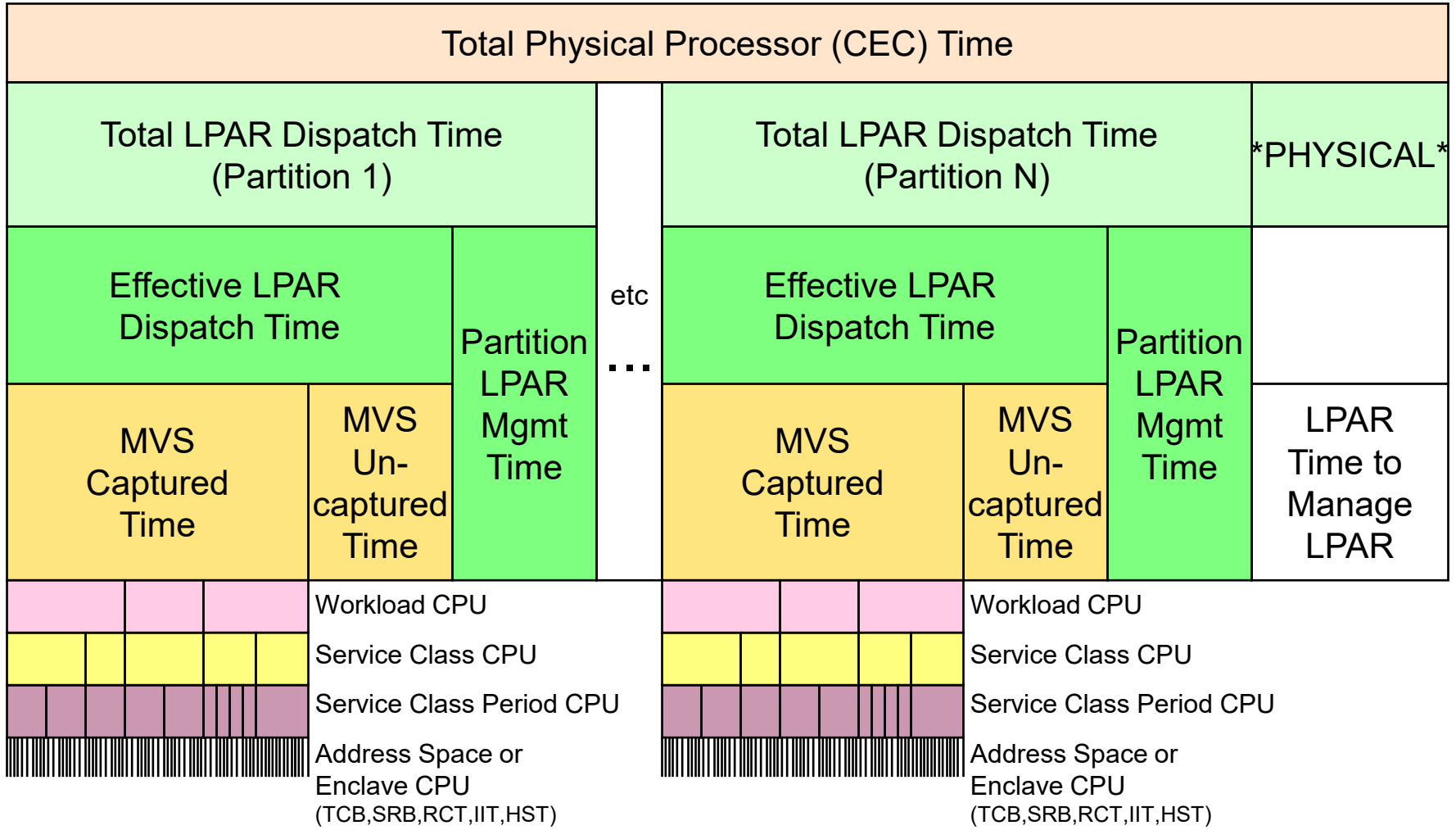
- The z/OS Performance Graphs you see here come from Pivotor

- 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!

Breaking Down GCP CPU Consumption

- Total utilization of the CEC = total dispatch time of all the LPARs + time spent to manage LPARs (*PHYSICAL*)
- Note Uncaptured time within an LPAR
  - Time that z/OS cannot directly attribute to a workload



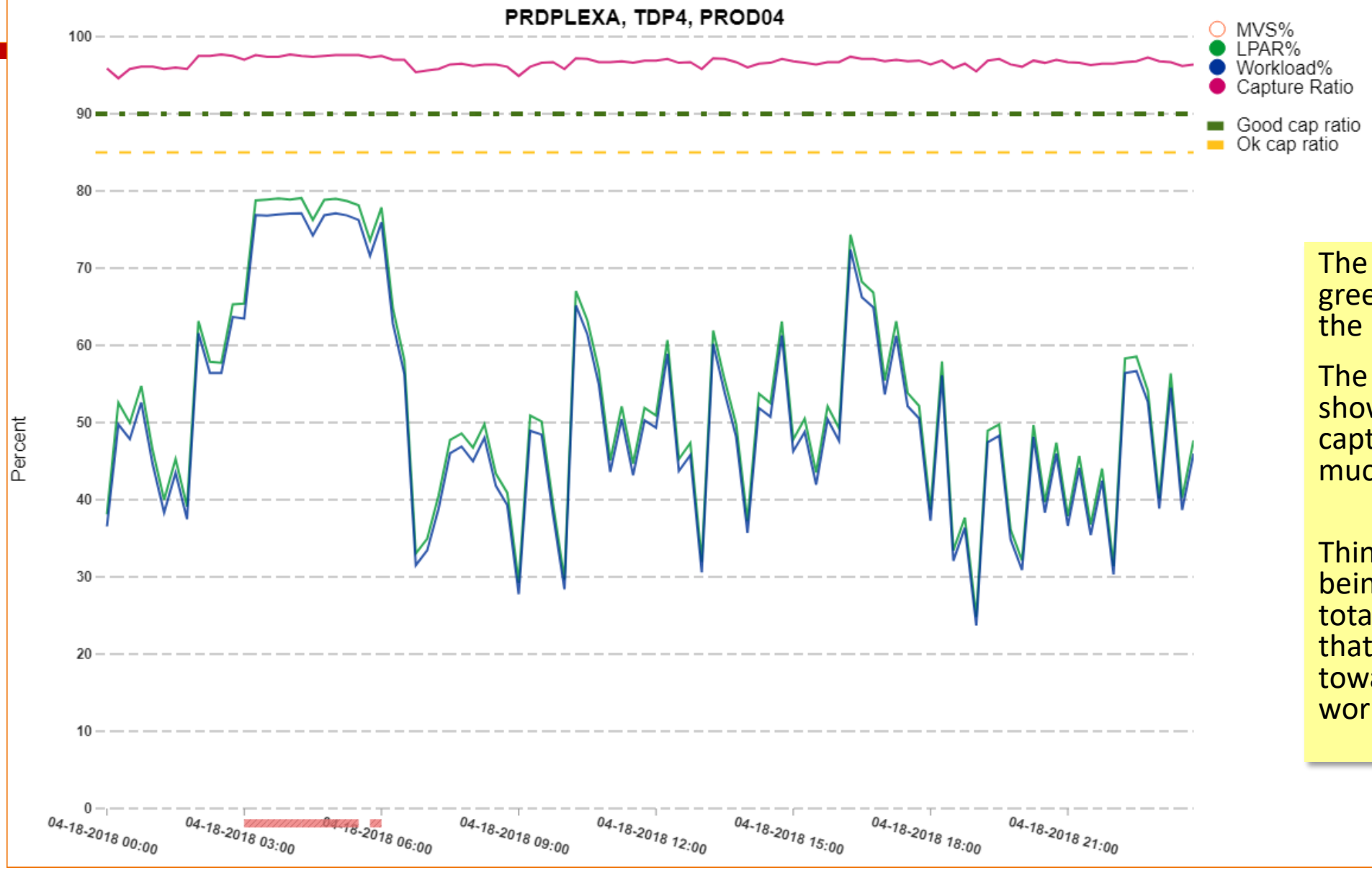
Capture Ratio

- Captured time
  - CPU time accounted for, and attributed, to particular workloads
  - Example: Sum of GCP CPU time for all WLM service class periods in the type 72 records
- Capture Ratio = $\frac{Captured\ time}{Effective\ dispatch\ time}$

Target Capture Ratio: >90%

- Uncaptured time = $(Captured\ time) - (Effective\ dispatch\ time)$
- Some uncaptured time is normal and is because z/OS cannot directly attribute some system-related work to a specific workload
  - Periodic system management
  - Interrupts
  - Contention
  - Paging
  - SLIP traps and system traces
  - Inefficient allocation process
  - First reference page faults

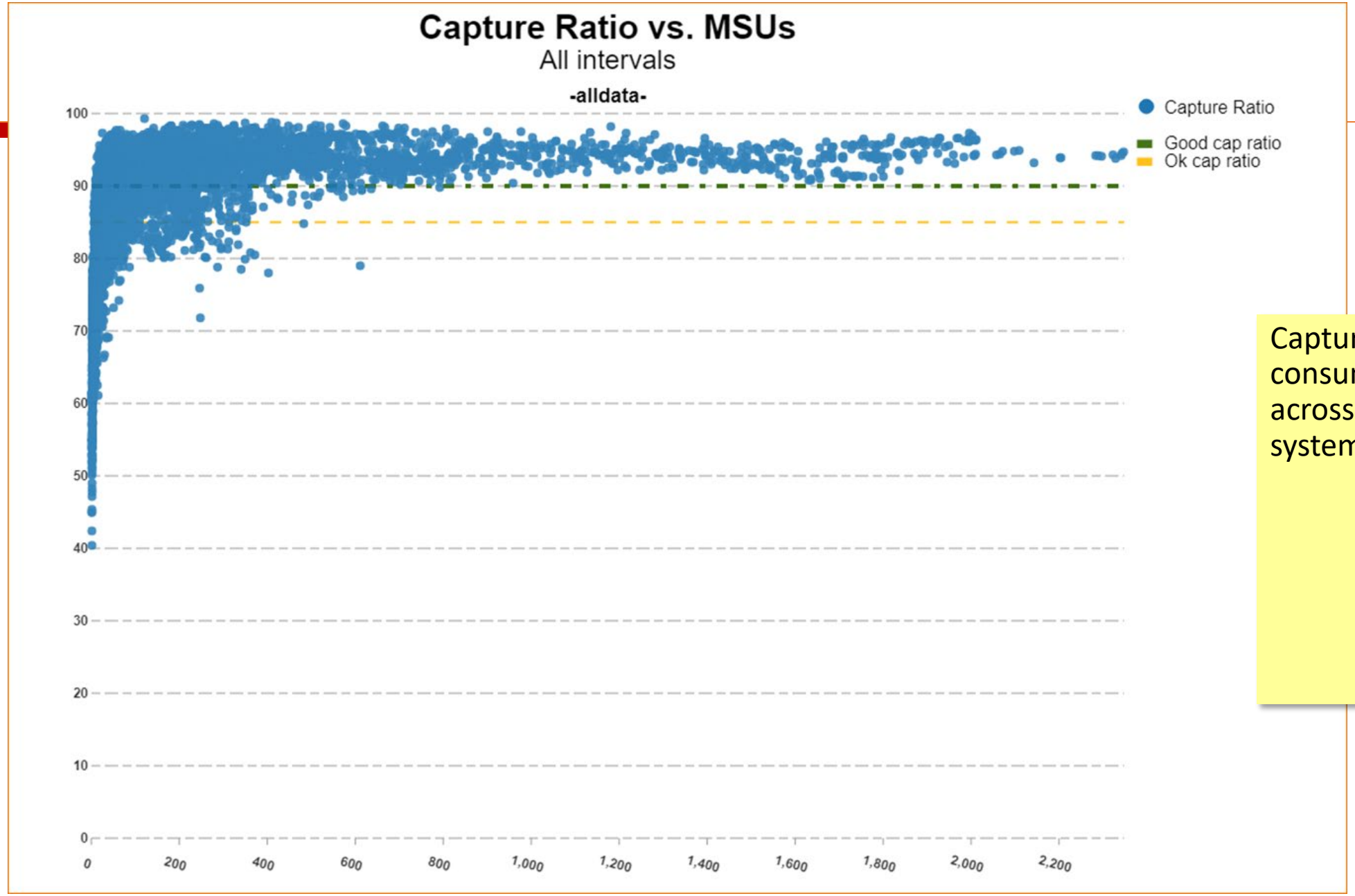
LPAR CP Busy%, MVS CP Busy%, Workload CP Busy%, and Capture Ratio



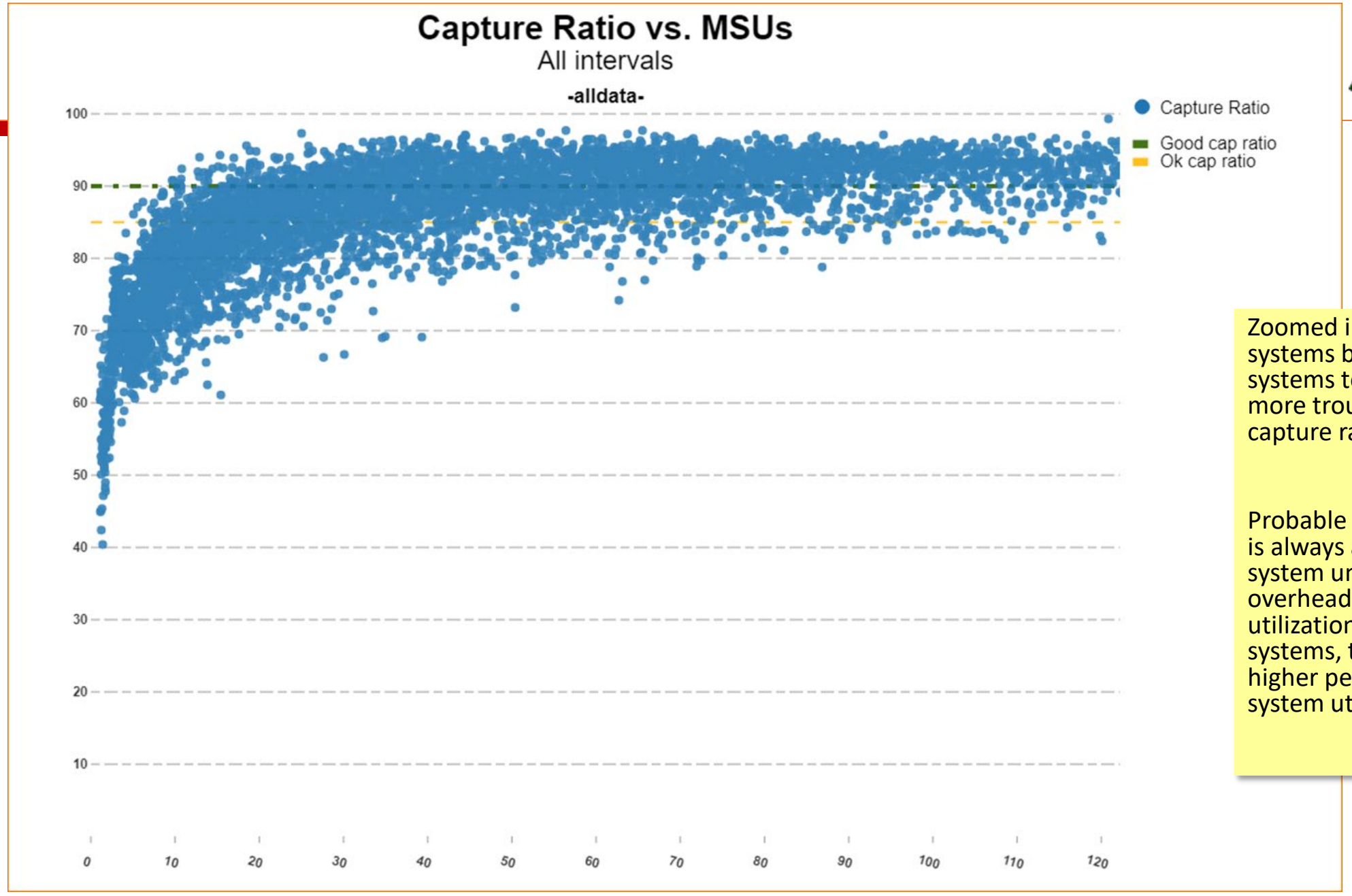
The space between the green and the blue lines is the uncaptured utilization.

The Capture Ratio is also shown. This system's capture ratio is pretty much ideal.

Think of capture ratio as being the percentage of total CPU time consumed that is accounted for towards a particular workload



Capture ratio vs. MSUs consumed in interval across many different systems



Zoomed in to see smaller systems better: very small systems to do seem to have more trouble achieving high capture ratios.

Probable reason is that there is always a fixed amount of system uncaptured time (i.e. overhead). Thus, at lower utilizations or smaller systems, this time is usually a higher percentage of the system utilization.

Evaluating Capture Ratio

- If capture ratio is above 90%: no concern, probably little tuning opportunity
- Some systems clearly run relatively low capture ratios all the time
  - These seem to be mostly lightly loaded systems
- If a capture ratio is regularly below 85% and the system is a significant contributor to your software costs, I would look into it
  - Look for old traces or SLIP traps that may have been left enabled
  - Do you have RMPTTOM set too low in IEAOPTxx?
    - May be able to increase it above default for lightly used systems
  - Look for paging activity
  - Look for contention
  - Unfortunately, tracking down uncaptured time is usually difficult
    - Because it is “uncaptured”
  - Do you have large SMS pools?

First Reference Page Faults Decrease Capture Ratios

What is a first reference page fault?

- Demand Page Faults
  - Typically, virtual frames are backed by real storage
  - Assumes the page was previously reference and populated for it to get stolen
  - If there is stress on storage, a real frame could be paged out to auxiliary storage
  - When that frame is re-referenced, this is known as a demand page fault
  - Demand Page Fault:
    - When a referenced page of virtual storage is not backed by a frame in central storage, a page fault occurs. This requires z/OS to retrieve the page from auxiliary storage and bring it into central storage.

- First Reference Page Fault
  - When a referenced page of virtual storage is not ***YET*** backed by a real frame in central storage, a first reference page fault occurs
  - Assumes the page was never previously reference and populated for it to get stolen
  - It is the 1st reference page fault
    - Finds the real frame and fills in the segment table
    - Drives Dynamic Address Translation (DAT), and the real frame is associated with the virtual address

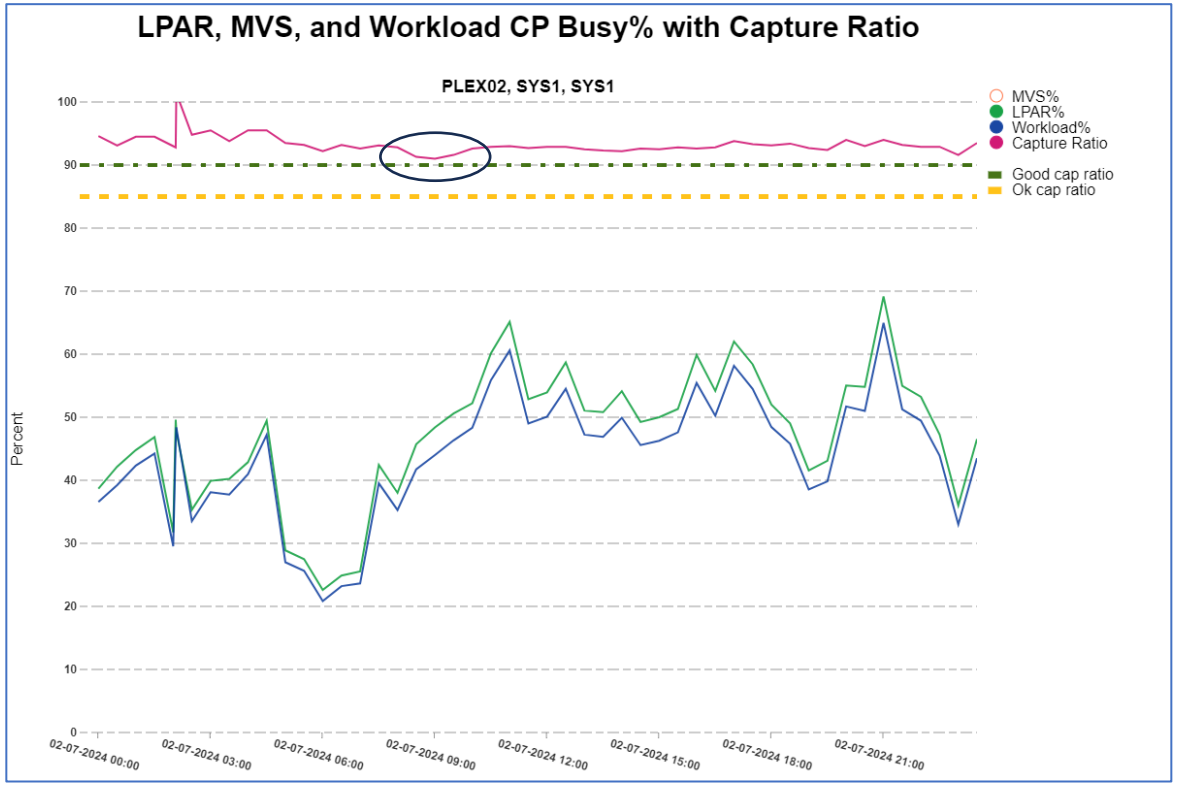
Capture Ratios and 1st Referenced Page Faults

- IBM WCS says that 1st Reference Page Faults contribute to uncaptured times
  - And that 1st Reference Page Fault rates above 100,000 per second should be considered problematic

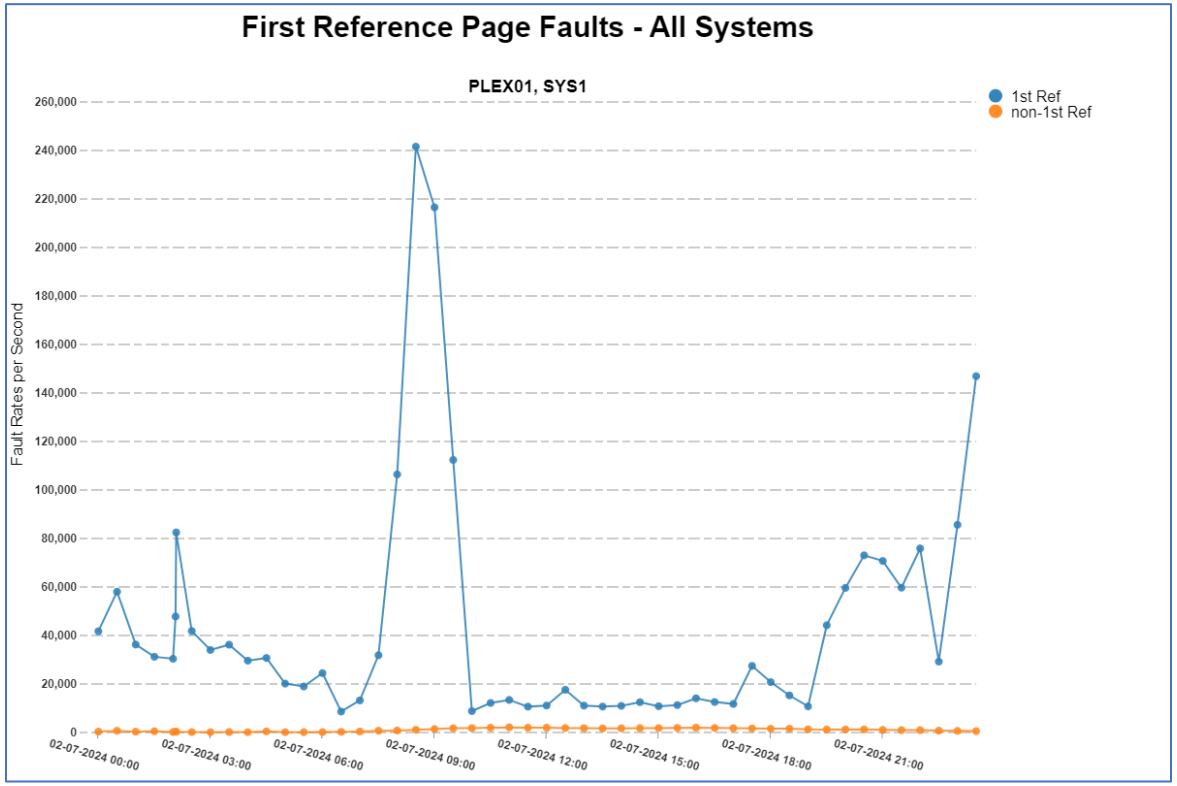
- Comments:
  - Probably a modern-day issue since storage areas and getmains can be super large
  - There is not much that can be done by customers to alleviate 1st Reference Page Faults
    - Perhaps recode applications to get less storage?
    - However, correlating them to capture ratios can be helpful to explain some of the uncaptured times
  - So many things contribute to uncaptured times, that is tough to see the direct correlation
  - Just understand this, and if investigating low capture ratios, then consider analyzing your 1st reference page faults to *maybe* help explain.

Example : Tough to see any correlation

Capture Ratios for System

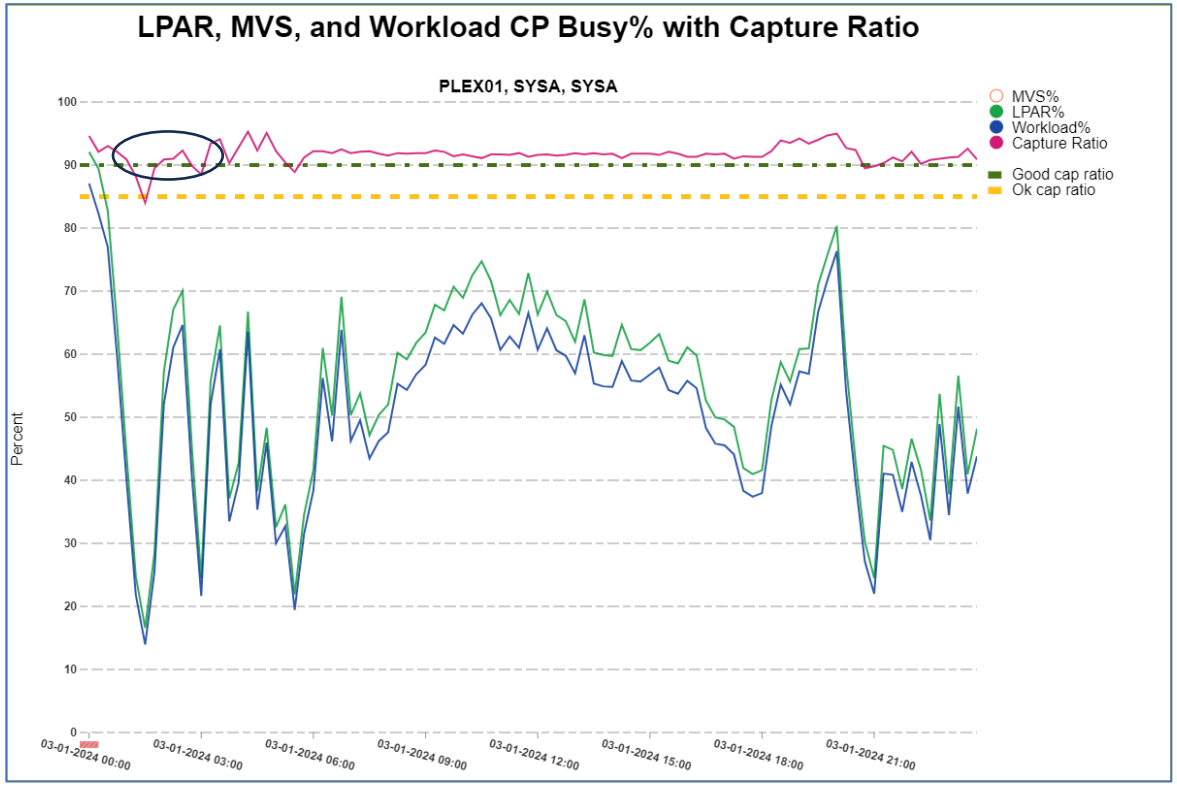


1st Reference Page Fault Rates

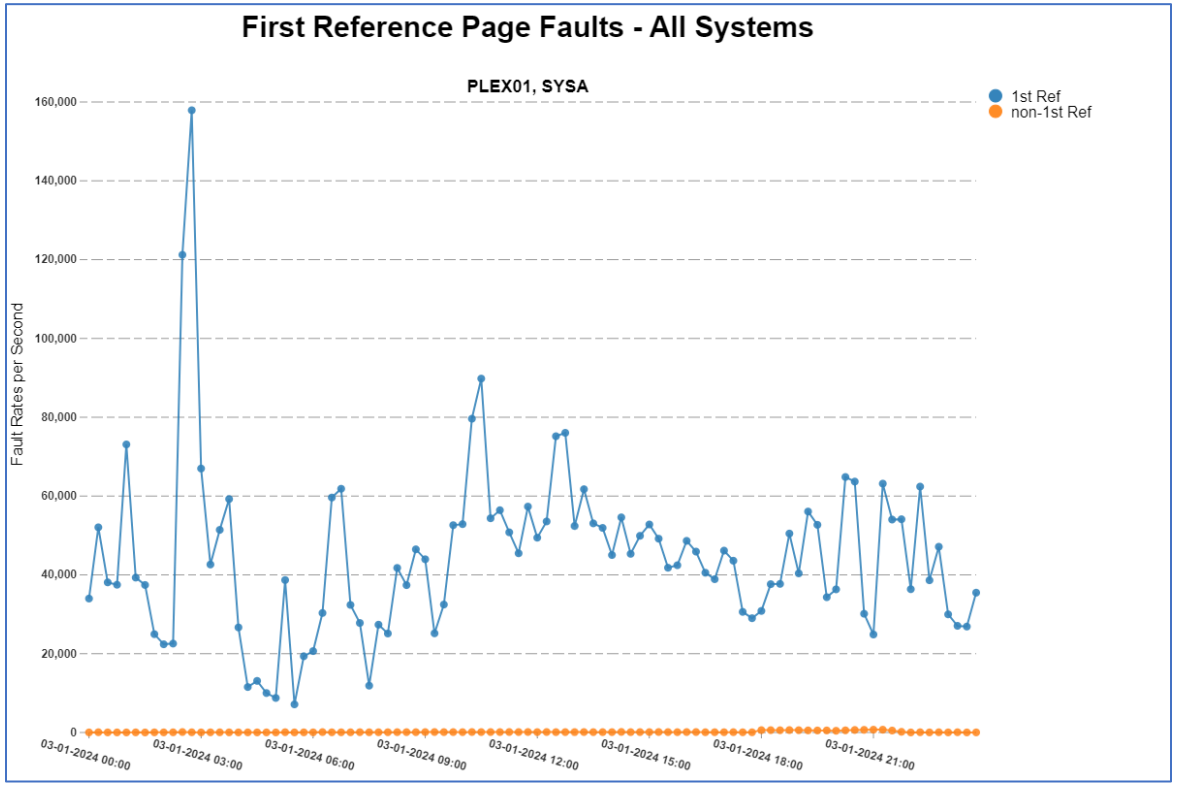


Example : Tough to see any correlation

Capture Ratios for System



1st Reference Page Fault Rates



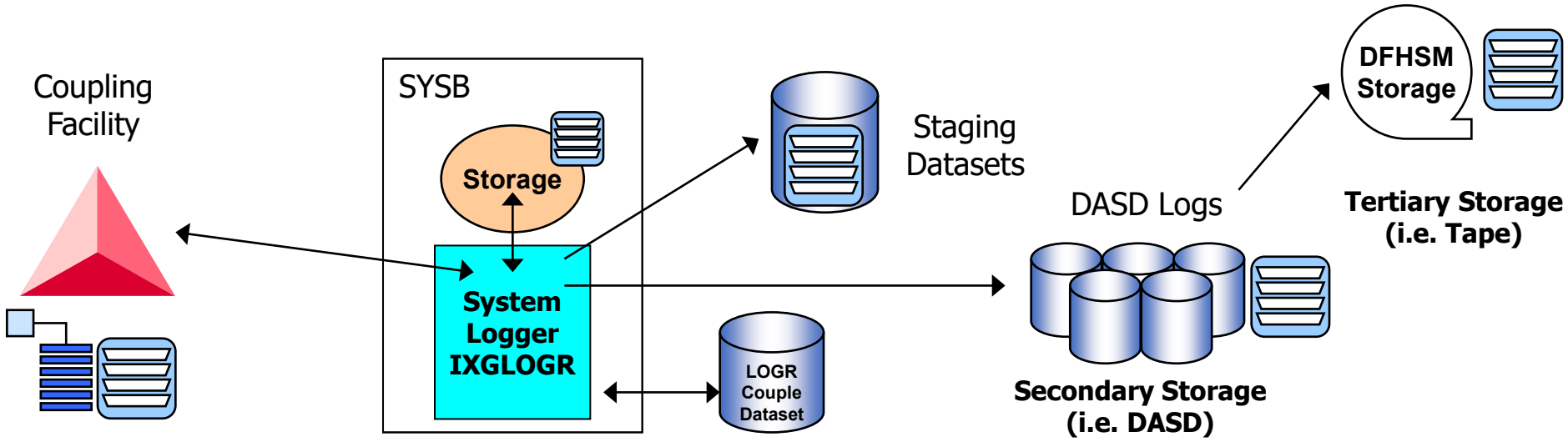
New z/OS System Logger IXGCNFxx Parameter

KEEPLOCALBUFFERS(NO | YES)

Targeted to alleviate the uncaptured time due to 1st reference page faults

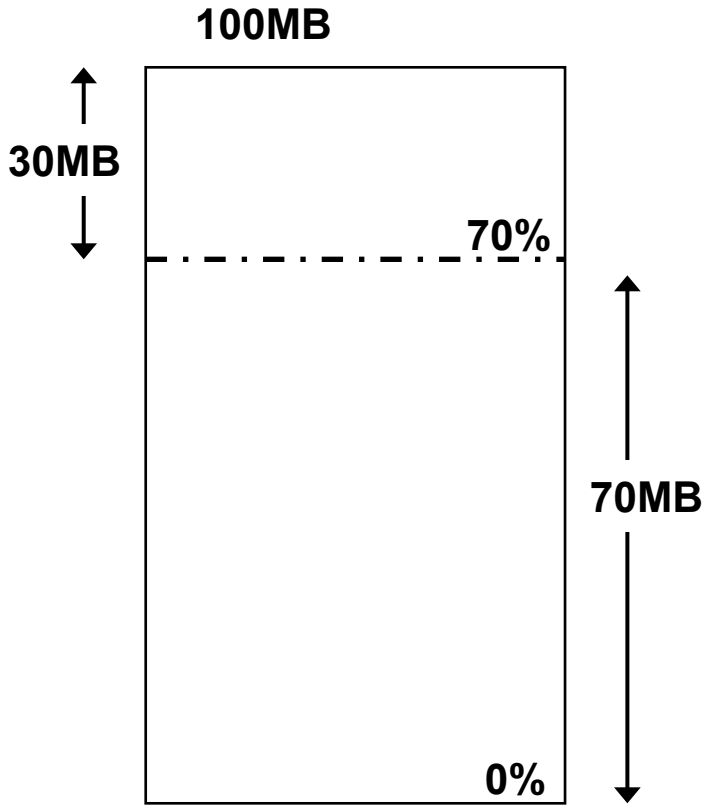
Introduction to z/OS System Logger

- z/OS System Logger - Component of z/OS that provides logging services
  - IXGLOGR – key system address space for logger functions
  - Interim Storage - Primary storage used to hold the log data that has not yet been offloaded
    - What 'interim storage' is depends on how the log stream has been setup
    - Examples of include central storage (via a data space), Coupling Facility, Staging data sets
  - Secondary Storage - generally DASD
  - Tertiary Storage – generally Tape medium



Logstream Memory Usage

- Data is written to memory and duplexed to either a staging dataset or CF structure
  - The point is, the memory required by logger is directly related to the size of your staging datasets and CF structures
  - Some Logstreams, such as those of SMF, could required large amounts of storage



- General reminders unrelated to this lecture:
  - Active type log streams
    - Goal is to never have offloads to disk
    - If do have offloads, then want to keep some data in interim storage
    - HIGHOFFLOAD never higher than 80%
    - LOWOFFLOAD should be between 60% and 75%
  - Funnel type
    - Start with HIGHOFFLOAD between 70% and 80%
    - LOWOFFLOAD should be zero
    - Key considerations
      - Calculate rate it takes to write data to the log stream
      - Calculate number of megabytes above HIGHOFFLOAD
      - For the log stream in question, calculate how long it takes to write that amount of data
      - Make sure I/O for offload can keep up with the write rate

New system Logger IXGCNFxx Parm

(APAR OA63551)

- PROBLEM DESCRIPTION:
  - New function to reduce page faults caused by IXGWRITE requests that were submitted after a log stream offload occurred.
- RECOMMENDATION:
  - Delays in completing IXGWRITE requests can occur as a result of page faults associated with system logger local buffers used by IXGWRITE processing.
- Comments
  - A new IXGCNFxx parmlib option will be introduced to keep the real frames that back the local buffers when the storage for the local buffers are freed after a log stream offload.
  - Keeping the real frames reduces page faults that will occur when the local buffers are reused during subsequent IXGWRITE requests. This will result in an increase of real storage associated with the System Logger address space.

New IXGCNFxx KEEPLOCALBUFFERS Parm

KEEPLOCALBUFFERS(NO | YES)

- Specifies whether the system will request to keep the real frames backing the local buffers used as interim storage when it is freed. Keeping the real frames reduces page faults that will occur when the local buffers are reused during subsequent IXGWRITE requests.

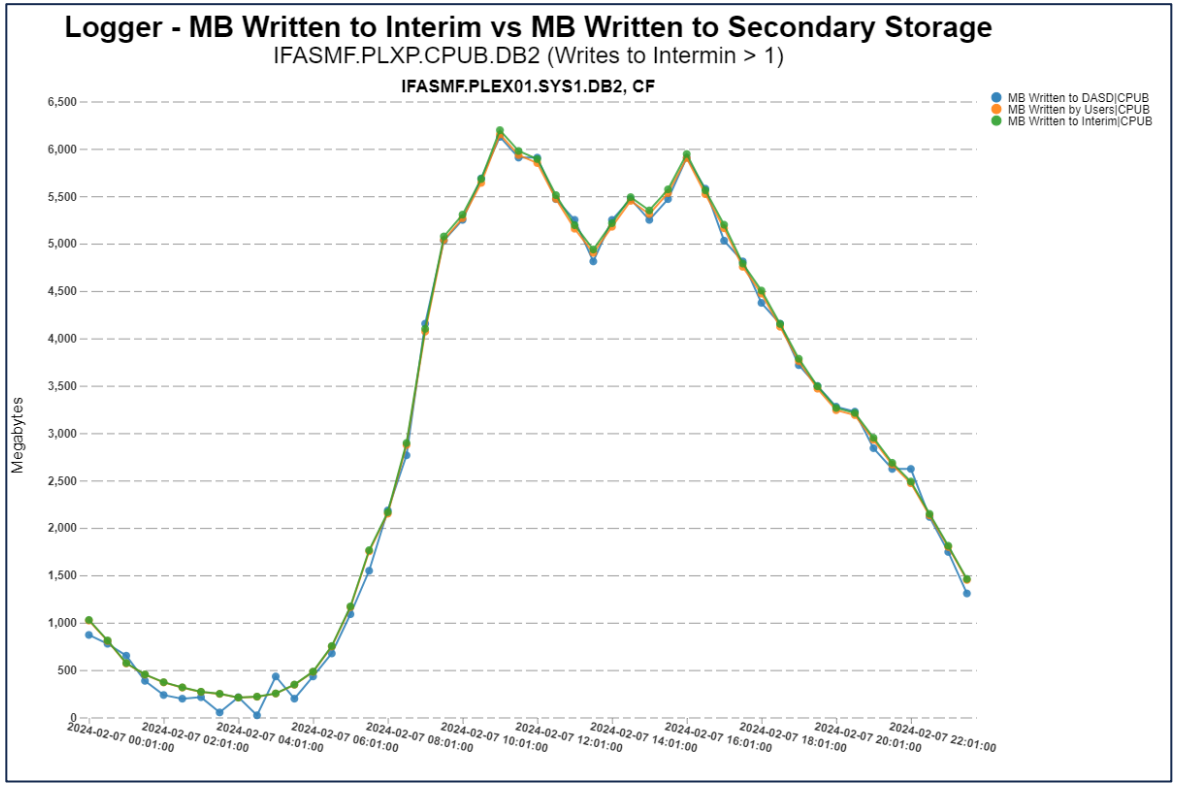
- Note: Local buffers are data space areas associated with the system logger address space, IXGLOGR. Specifying KEEPLOCALBUFFERS(YES) may result in systems experiencing increased paging.
- Evaluate your real memory requirements to ensure unacceptable paging does not occur by reviewing the amount of real memory consumed by the system logger address space, IXGLOGR.

- The following options are possible:
  - NO - Indicates that the system will not keep the real frame used to back local buffers when the buffer storage is freed.
  - YES - Indicates that the system will request to keep the real frame used to back local buffers when the buffer storage is freed.
- You can use the DISPLAY LOGGER,IXGCNF,MANAGE command to view the parameter settings for configuring the system logger.

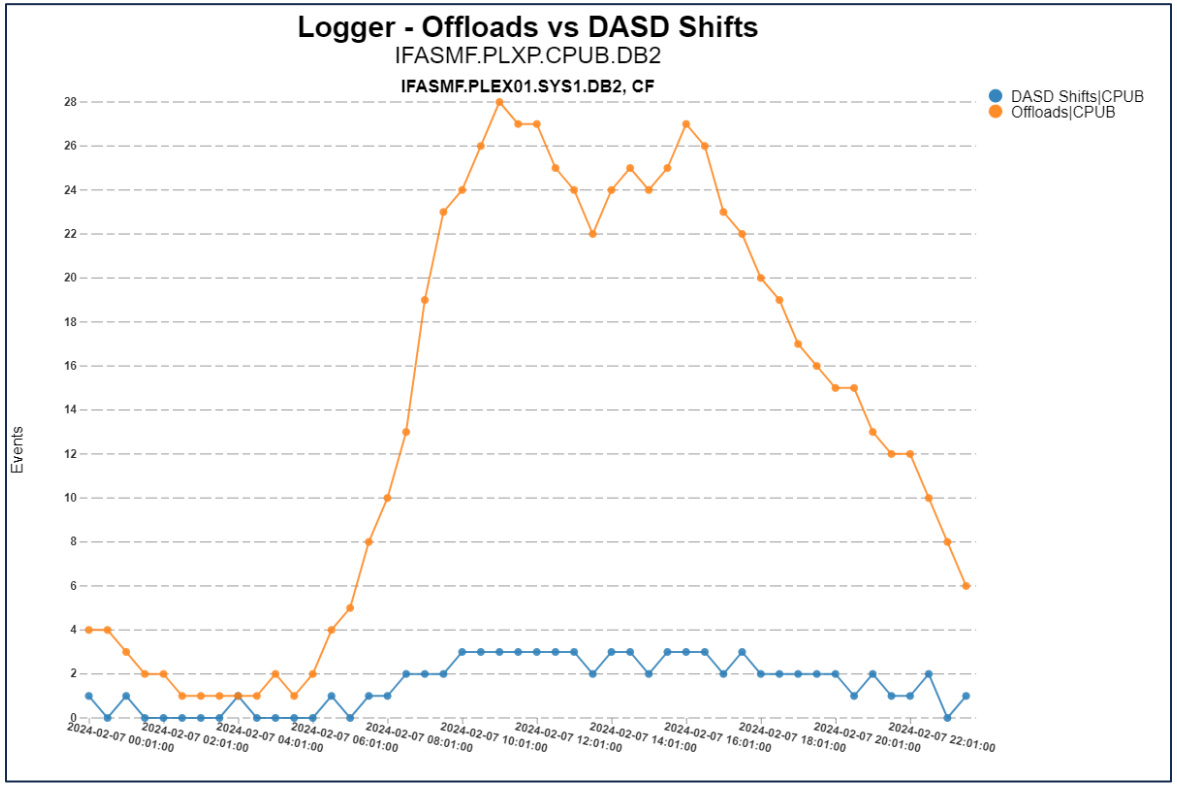
- Default: NO

Example: Logger Offloads of SMF

MBs of SMF offloaded

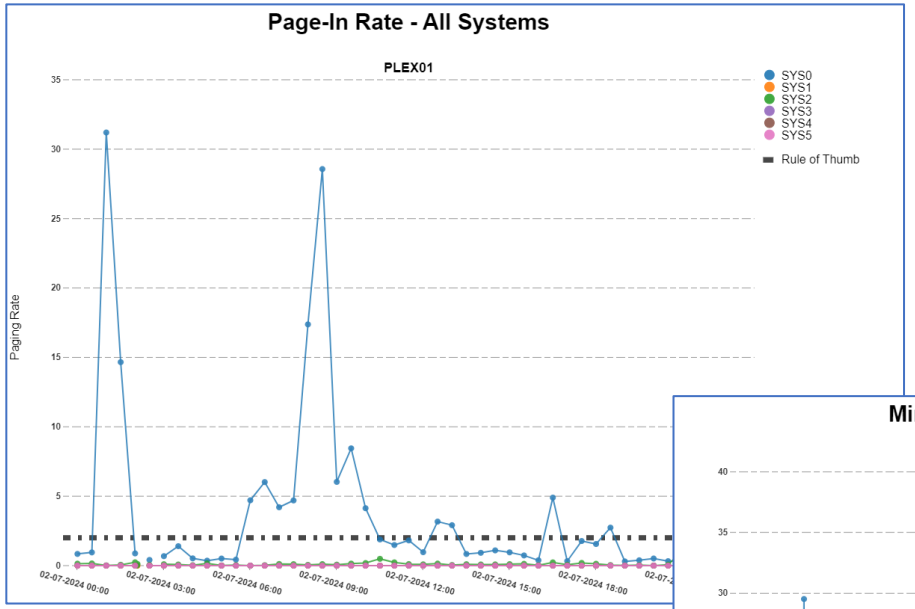


Number of offloads

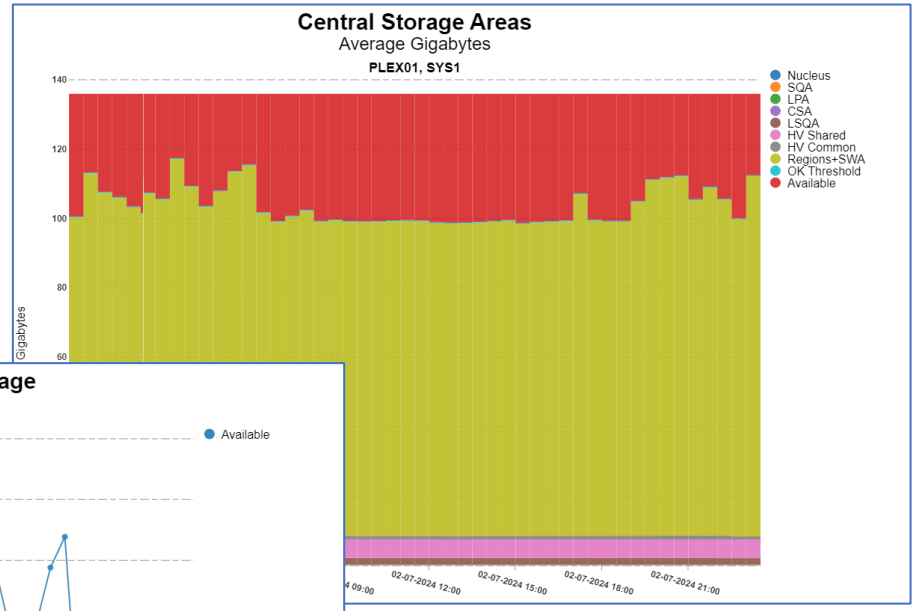


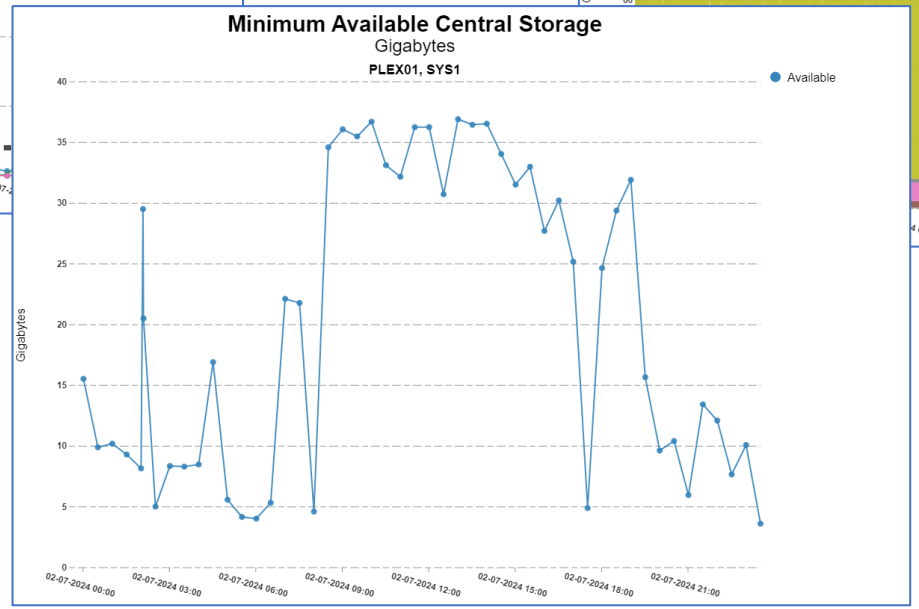
Example: Logger Offloads of SMF

Paging



Average and Min Storage Available





Questions?