

Seven New Year's Resolutions for Mainframe Performance and Capacity People

Scott Chapman

Enterprise Performance Strategies, Inc.

Scott.Chapman@EPStrategies.com

<https://www.pivotor.com>

<https://www.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:

© Enterprise Performance Strategies, Inc. All rights reserved. No part of this material may be reproduced, distributed, stored in a retrieval system, transmitted, displayed, published or broadcast in any form or by any means, electronic, mechanical, photocopy, recording, or otherwise, without the prior written permission of Enterprise Performance Strategies. To obtain written permission please contact Enterprise Performance Strategies, Inc. Contact information can be obtained by visiting <http://www.epstrategies.com>.

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

Welcome to 2026!

- 2025 is in the rearview mirror... let's look ahead to 2026!
- This presentation is a few goals for you to help make your mainframe environment even better in 2026
 - Mostly from a performance and capacity perspective
 - Plus a couple that maybe can help you or others in the mainframe community



Seven Resolutions



- Record useful data
- Periodically review your reports
- Revisit your WLM Goals
- Understand your software cost drivers
- Engage with the community
- Optimize your SRB
- Learn something about AI

#1: Engage with the Mainframe Community



You can learn a lot from talking to people



- Attend a conference and talk to other attendees
 - Yes, this costs money, as does the mainframe environment: shouldn't your company want the people taking care of that expensive environment to be as knowledgeable as possible?
- Online list servs (You might want to continue subscribing as a daily digest to limit the number of emails you see per day!)
 - IBM-Main <https://listserv.ua.edu/cgi-bin/wa?A0=ibm-main>
 - Specifically for performance people, MXG-L: https://www.mxg.com/frame_mxgl.asp
 - If Discord is your thing, see System Z Enthusiasts <https://discord.gg/sze>
 - IBM community groups: <https://community.ibm.com/community/user/groupz>
- Seek out local mainframe users
 - Schedule a meet-up?
 - If you're in Central Ohio: email me about Central Ohio Mainframe Users Group

#2: Record Useful Data



Are you recording the data you should?



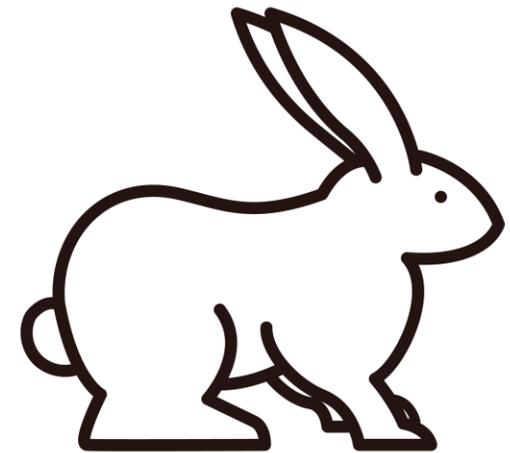
- See my SMF Recommendations presentation from the November webinar
 - Available on our website at <https://www.pivotor.com/content.html>
- Record your SMF/RMF data at no longer than a 15 minute interval
- Make sure you're recording these record types:
 - 113 – Hardware Instrumentation Services
 - Most are now recording these, but make sure you have the extended counters enabled
 - 99 – *At least* subtypes 6, 10, 11, 12, 14
 - But really record them all, except 13 which is undocumented and somewhat voluminous
 - 98 – IBM recommendation is 5 second interval, I'm now on board with that
 - See HFTSINTRVL
 - 42 – I expect most shops are recording these, but have you looked at them?
 - More detailed view of I/O than you just get from the 74s

Bonus Best Practice



- Record business metrics related to your mainframe applications
 - Customer calls
 - Account activity (open, close, orders, whatever)
 - Widgets produced
 - Website inquiries
 - Whatever business action drives mainframe application activity
- The application will be recording these activities one way or the other
 - Talk to the application team about how to get these metrics out of the database
- Would be nice to see these metrics hour by hour but daily is a fine place to start
- Note to Pivotor customers: talk to us about ingesting and reporting on this data for you (usually no additional charge)

#3: Review your reports



Daily report review

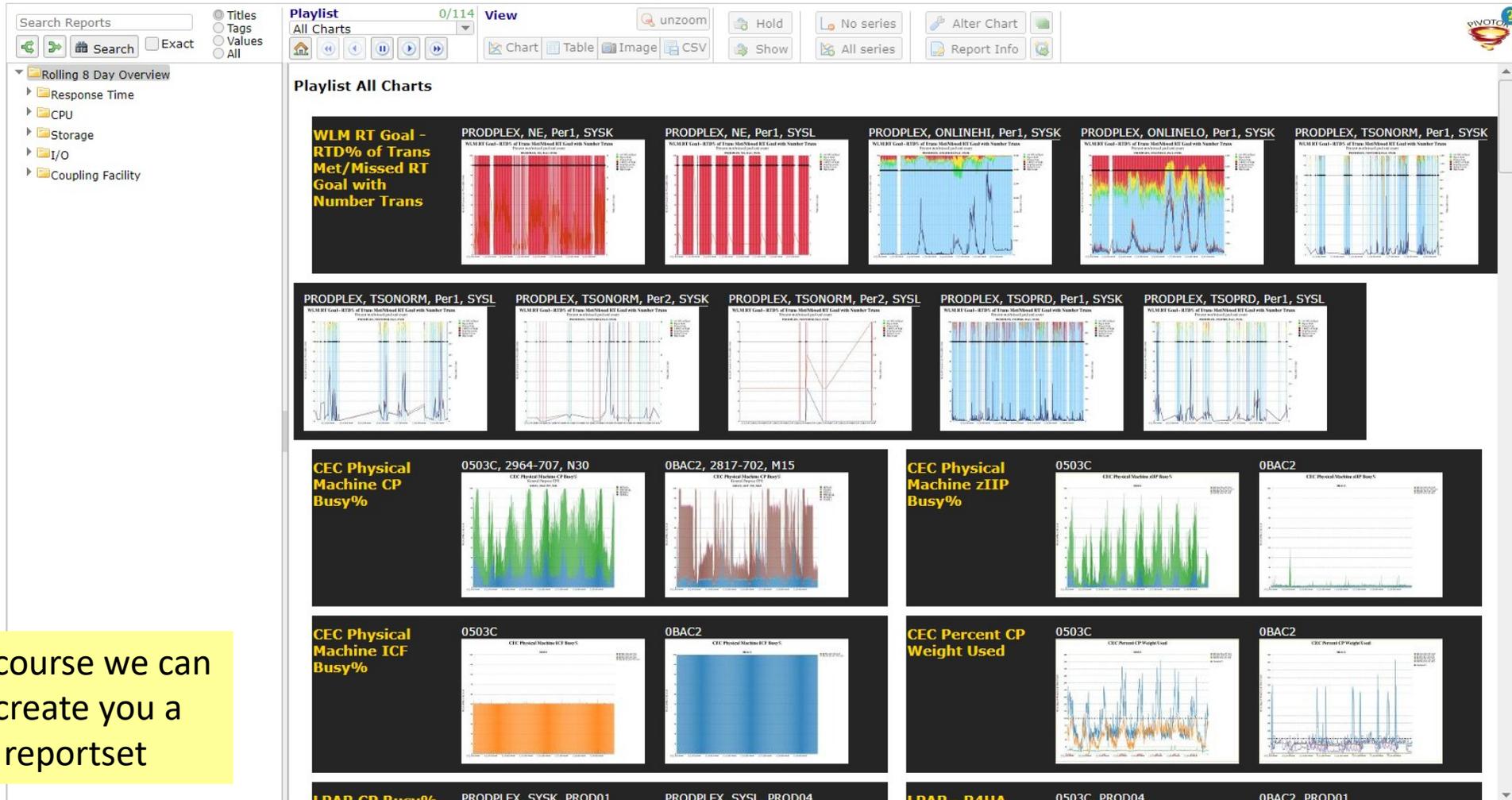


- If you have responsibility for performance / capacity, resolve to spend at least 5 minutes every day looking at the reports from yesterday
 - This seems obvious to me, but I know there are sites that are not doing this!
 - By looking at the data every day you'll gain a better appreciation for how your environment works
 - Don't be afraid to chase down some rabbit trails
- My top reports I'd always look at:
 - Application response time
 - CEC utilization
 - Business metrics

For Pivotor Customers



- Consider the thumbnail view of the rolling 8 day reports

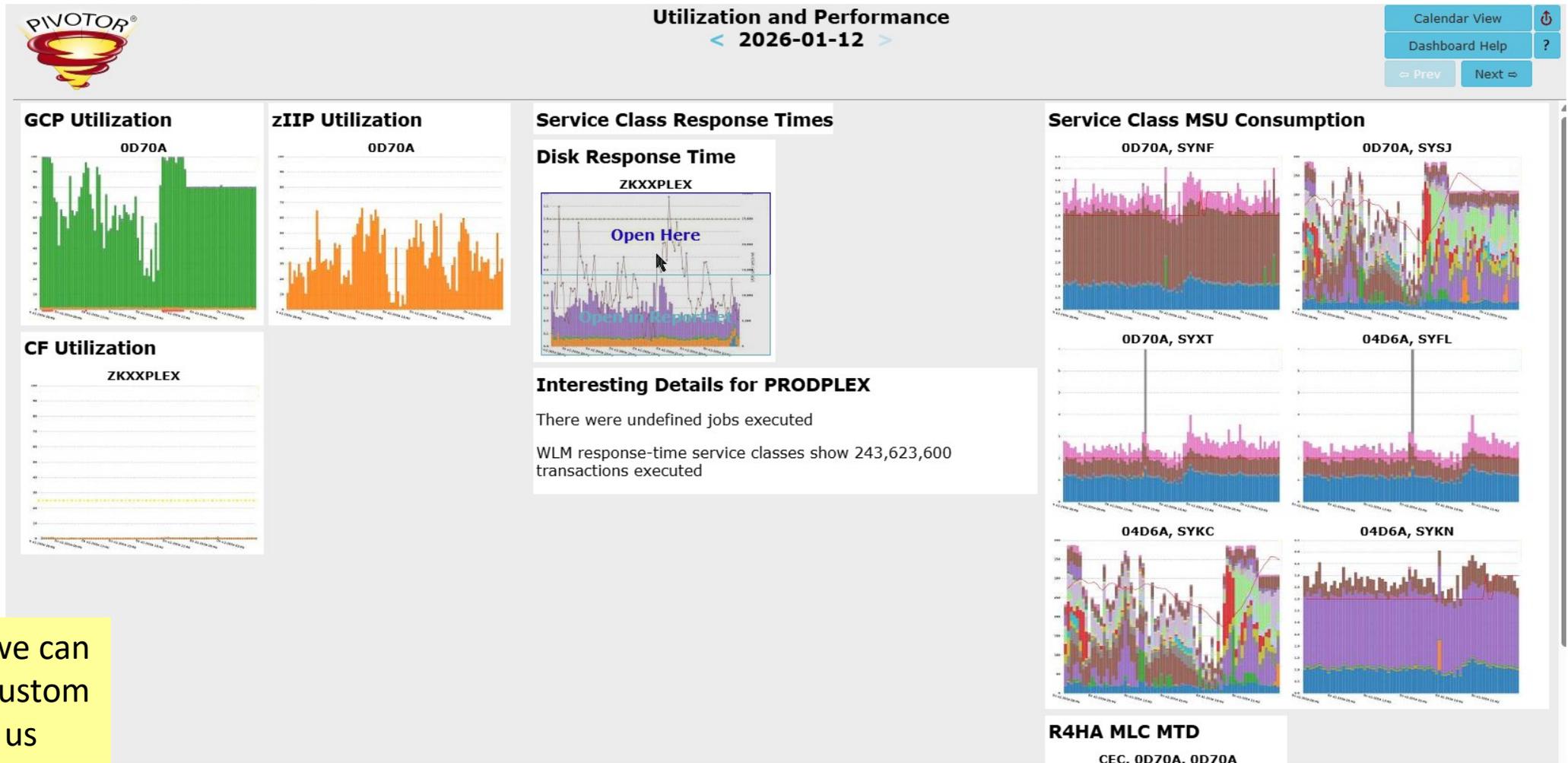


And of course we can always create you a custom reportset

For Pivotor Customers



- Consider Dashboards



And of course we can always create custom dashboards for us

Pivotor customers: Exception reports



- Consider looking at the System Health Check reports for charts flagged as exceptions
 - The same charts will be similarly flagged in the rolling 8 day reports, but of course that reportset has a limited number of reports in it, so you may miss some exceptions that might be flagged in the other more verbose reportsets

A screenshot of the Pivotor web interface. The top navigation bar includes a search box, a playlist dropdown set to "All Charts" (0/554), and various view and chart manipulation buttons like "unzoom", "Hold", "No series", "Alter Chart", "Report Info", "Show", and "All series". The left sidebar shows a tree view of report categories under "z/OS Health Check", including Processor Analysis, MSU Usage Analysis, Storage / Memory Analysis, SMF 30 Address Space Analysis, WLM Analysis, Virtual Storage Activity Analysis, and Environment Summary. The main content area displays a "Report Exception Overview" section with a plus sign and a "Report Playlists" section with a minus sign. Under "Report Playlists", there are four items: "All Charts (197 reports, 554 charts)", "Charts Warranting Investigation Due to Exception Counts (2 reports, 2 charts, more details)", "All Charts with Exceptions (4 reports, 7 charts, more details)", and "Evaluating WLM Velocity Goals (4 reports, 33 charts, more details)". A red arrow points to the "Charts Warranting Investigation Due to Exception Counts" link.

Search Reports Exact

Titles
 Tags
 Values
 All

Playlist 0/2 View unzoom Hold No series Alter Chart Show All series Report Info

- z/OS Health Check
 - Processor Analysis
 - MSU Usage Analysis
 - Storage / Memory Analysis
 - SMF 30 Address Space Analysis
 - WLM Analysis
 - Virtual Storage Activity Analysis
 - Environment Summary

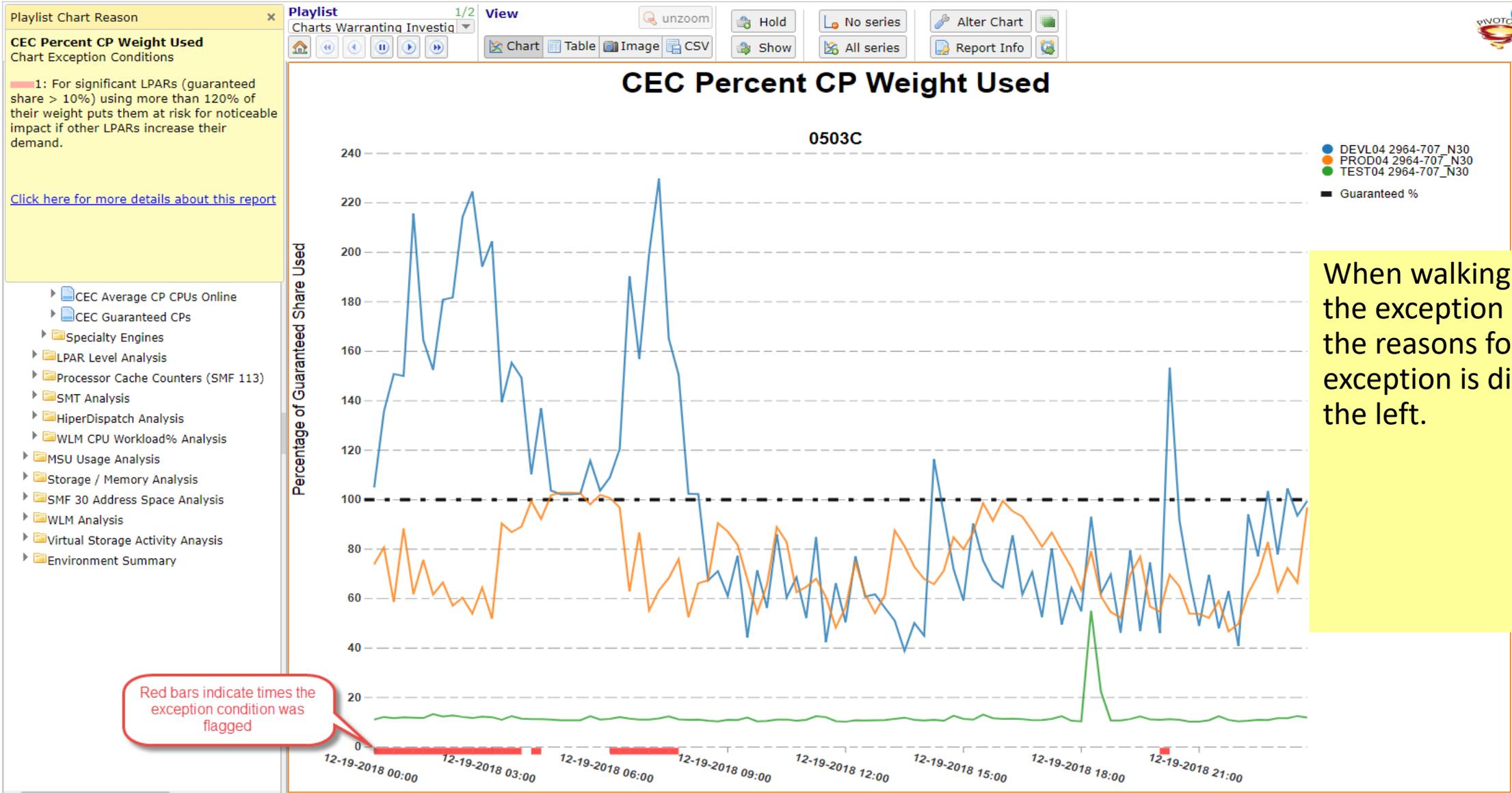
Playlist Charts Warranting Investigation Due to Exception Counts

CEC Percent CP Weight Used 0503C

LPAR CP Busy%, MVS CP Busy%, Workload CP Busy%, and Capture Ratio PRODPLEX, SYSK, PROD01

You may very well know exactly what's going on from the thumbnails

If you want to customize the thresholds to avoid flagging certain things that are explicitly set for your situation, let us know as the thresholds can be changed on a plex by plex basis.



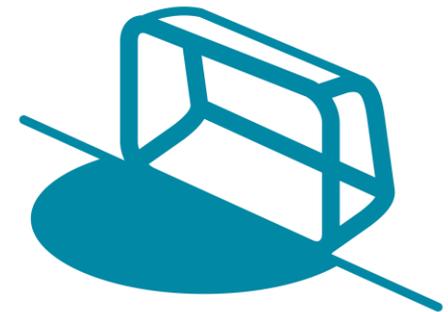
When walking through the exception playlist, the reasons for the exception is displayed on the left.

Don't have daily reporting?



- We do sometimes encounter sites that have effectively no daily reporting
- How much are you paying for your mainframe environment?
 - Investing in a little bit of reporting to better manage it seems like a good idea
- Consider:
 - Pivotor (easy, comprehensive, inexpensive)
 - Pivotor **Free** Tier (easy, important reports for 1 system for 1 week)
 - Existing reporting product that's under-utilized
 - Roll-your-own (may be “no additional cost” but at some effort)
 - RMF Spreadsheet Reporter
 - Mainframe Data Services for Spark (“optional” service and support cost, some effort)
 - Home-grown code to read SMF (not really recommended when there's low/no cost options)

#4: Revisit your WLM Goals



Revisit your WLM goals when:



- Any hardware changes
 - Migration to new hardware
 - Upgrade to existing hardware, e.g. adding a processor, changing processor model
- Some system configuration changes
 - E.G. changing weights or CPs online
- Most software changes (system or application)
 - Hopefully these have little impact, but that's not always the case
- Before implementing R4HA management / capping
- If it's been more than 6-12 months since you looked at them
 - Because probably something has changed in that time!

Goals for WLM Goals



- Goals should be reasonable and in the Goldilocks zone
 - Not too hard, not too easy
 - Just missing a goal is often better than beating it
- Goals that are too easy are one of the most common issues we see when reviewing WLM
- WLM PI
 - < 1 means work is doing better than its goal
 - 1 means work is exactly meeting its goal
 - >1 means work is doing worse than its goal
- PI values from about 0.90 to about 1.3 (maybe even a little more) are good
 - If a service class period's PI is regularly very much outside that range, then you might need to re-evaluate it
- Examine your goal attainment over time and at times of particular interest

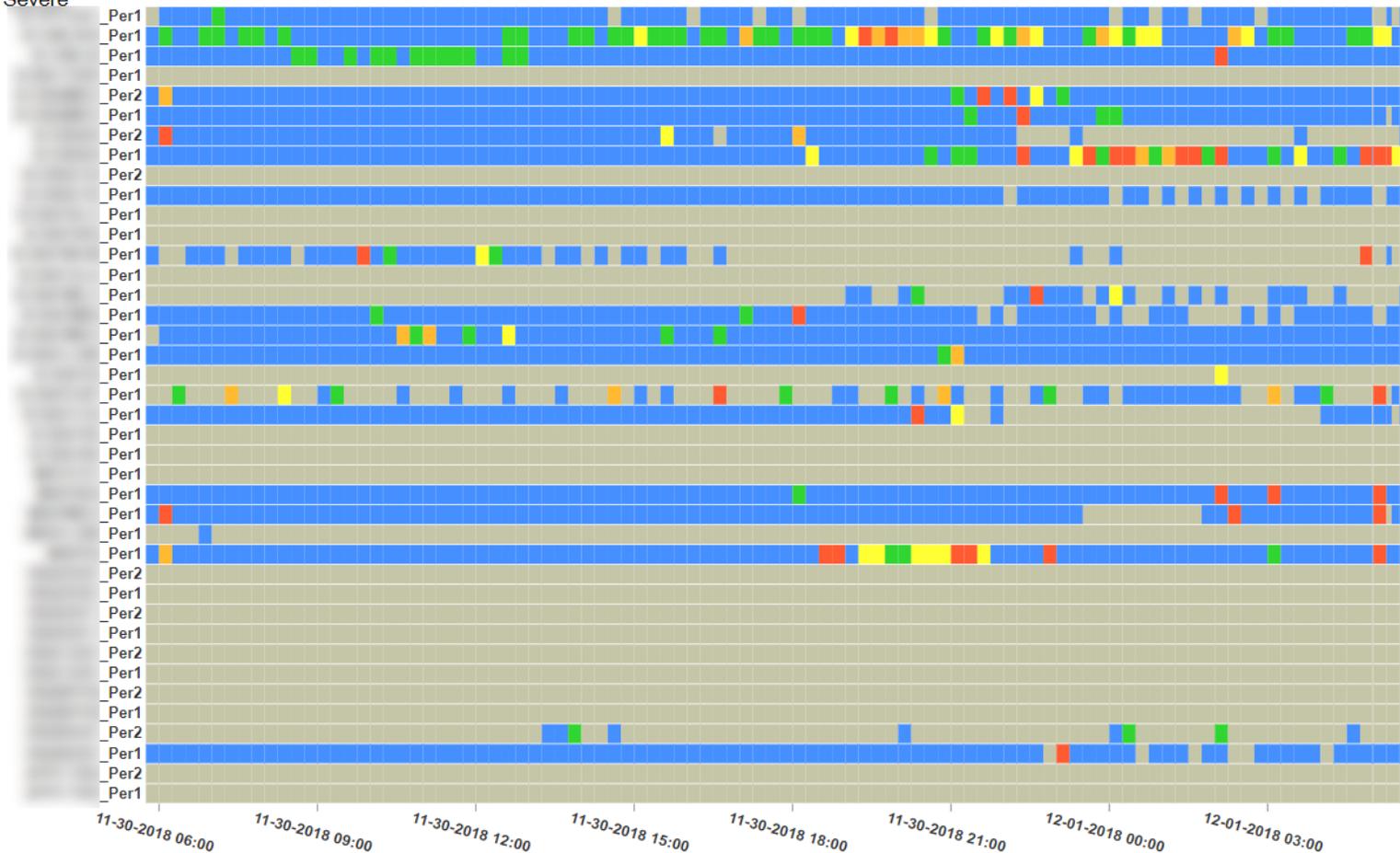
Possibly too easy goals example



- ≤ 0: Zero
- ≤ 0.81: Over Achieving
- ≤ 1.1: Met
- ≤ 1.4: Fair
- ≤ 1.99: Warning
- higher: Severe

WLM PI - PI Heat Chart for Service Class Periods

PRODPLEX, SYSA (1 of 2)



Note all the blue here which indicates service class periods with PIs ≤ 0.81

Possibly too difficult goals



- ≤ 0: Zero
- ≤ 0.81: Over Achieving
- ≤ 1.1: Met
- ≤ 1.4: Fair
- ≤ 1.99: Warning
- higher: Severe

WLM PI - PI Heat Chart for Service Class Periods

PPLEX, SYSB



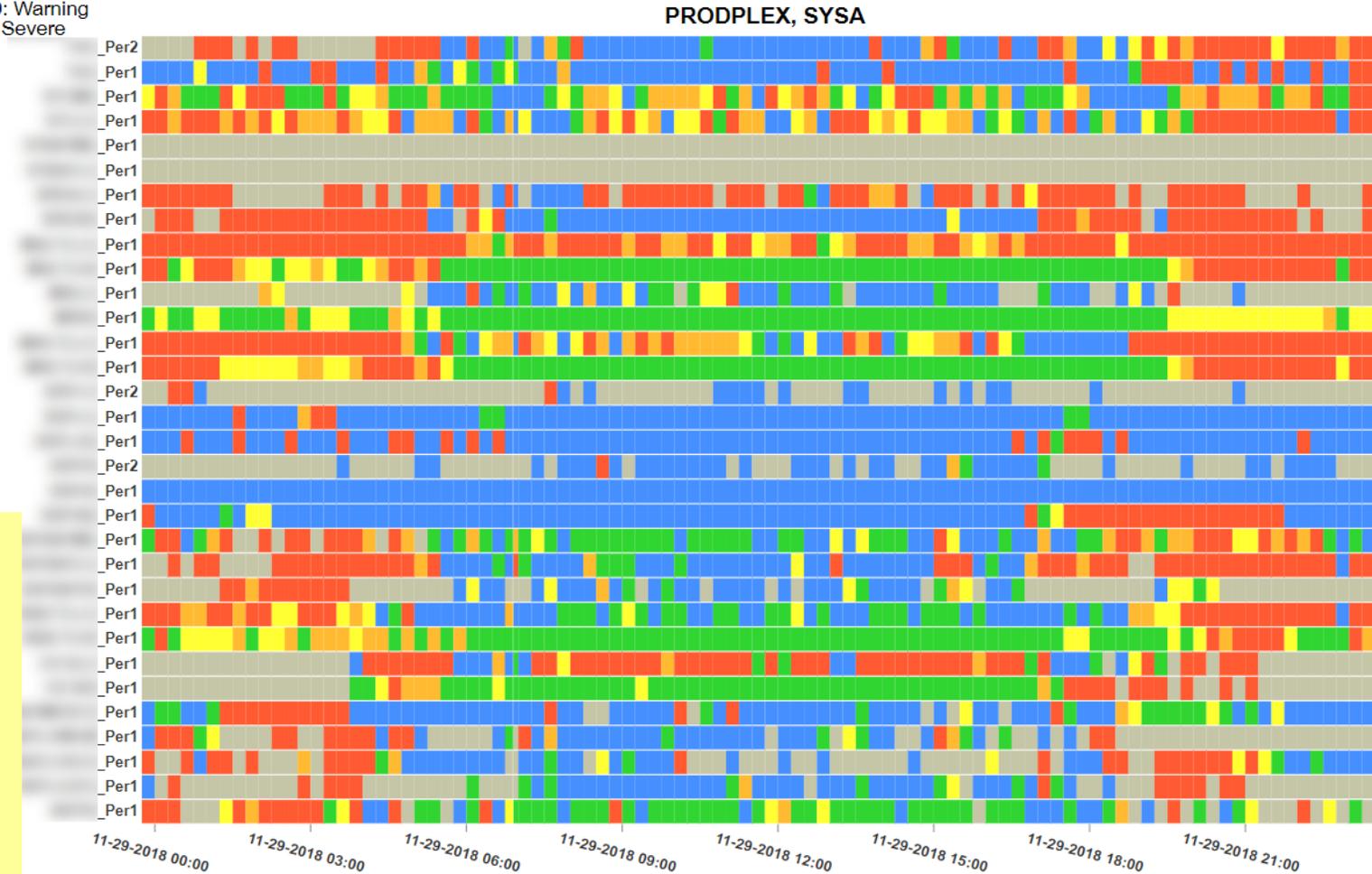
Lots of red here, plus not too many active SCPs (which may be ok, or may be an opportunity for better managing the work)

Mixture



- ≤ 0 : Zero
- ≤ 0.81 : Over Achieving
- ≤ 1.1 : Met
- ≤ 1.4 : Fair
- ≤ 1.99 : Warning
- higher: Severe

WLM PI - PI Heat Chart for Service Class Periods



A lot more green and yellow here makes me happier. This is a fairly common situation where a goal may not be perfect for all hours of the day.

If you need help with reviewing WLM...



- If you're a Pivotor customer:
 - We can discuss in detail in a annual review or even ahead of one of those
 - You're always welcome to send us an email with your specific questions
 - performance.questions@epstrategies.com
- See Peter's presentations on <https://www.pivotor.com/content.html>
 - z/OS Wworkload Manager: Revisiting Goals Over Time
 - WLM – Deciding Between Velocity Goals and Response Time Goals
 - WLM Management of CICS and IMS Workloads
- Upcoming webinar: “Wait? Do we need to reevaluate our WLM Goals?”
- Attend Peter's 4 day “WLM Performance and Re-evaluating Goals” class
- Request a **free** cursory review (generally about a 90 minute phone call)

#5: Understand Your Software Cost Drivers



Software Cost Importance



- Performance, capacity, and software cost are all inter-related today
- IBM MLC costs might be able to be reduced by restricting performance
 - Question is how much of a performance hit can you take: zero, a little, or a lot?
 - Better question is how much of a performance hit makes business sense?
- You might be able to improve performance by increasing software costs
 - E.G. by raising existing caps or by adding more capacity
 - What is the business benefit of increasing performance?
- You might be able to reduce software cost by increasing another cost
 - I.E. increasing installed capacity can in some cases improve performance and efficiency and lower the variable software costs
 - May be difficult if you have software costs based on your full installed capacity

Answers You Should Know



- Are you under R4HA or TFP for IBM MLC?
 - If under R4HA: when are your peaks?
 - If TFP: are you tracking to above or below baseline?
- What is your **incremental** per-MSU / per-MSUhour MLC cost?
 - Not average, incremental!
- Do you have an ELA in place with IBM that fixes your MLC cost?
 - Are you tracking above or below the ELA spend?
 - **I.E. if you reduce your R4HA will it actually reduce real dollars spent?**
 - If yes: Performance tuning & reducing the R4HA can save real money!
 - If no: Is there room to raise your caps to improve performance (if necessary)?
- What full-capacity ISV / IBM OTC contracts are in place and what are their limits?
- If you have more than one machine, are you using sysplex aggregation?

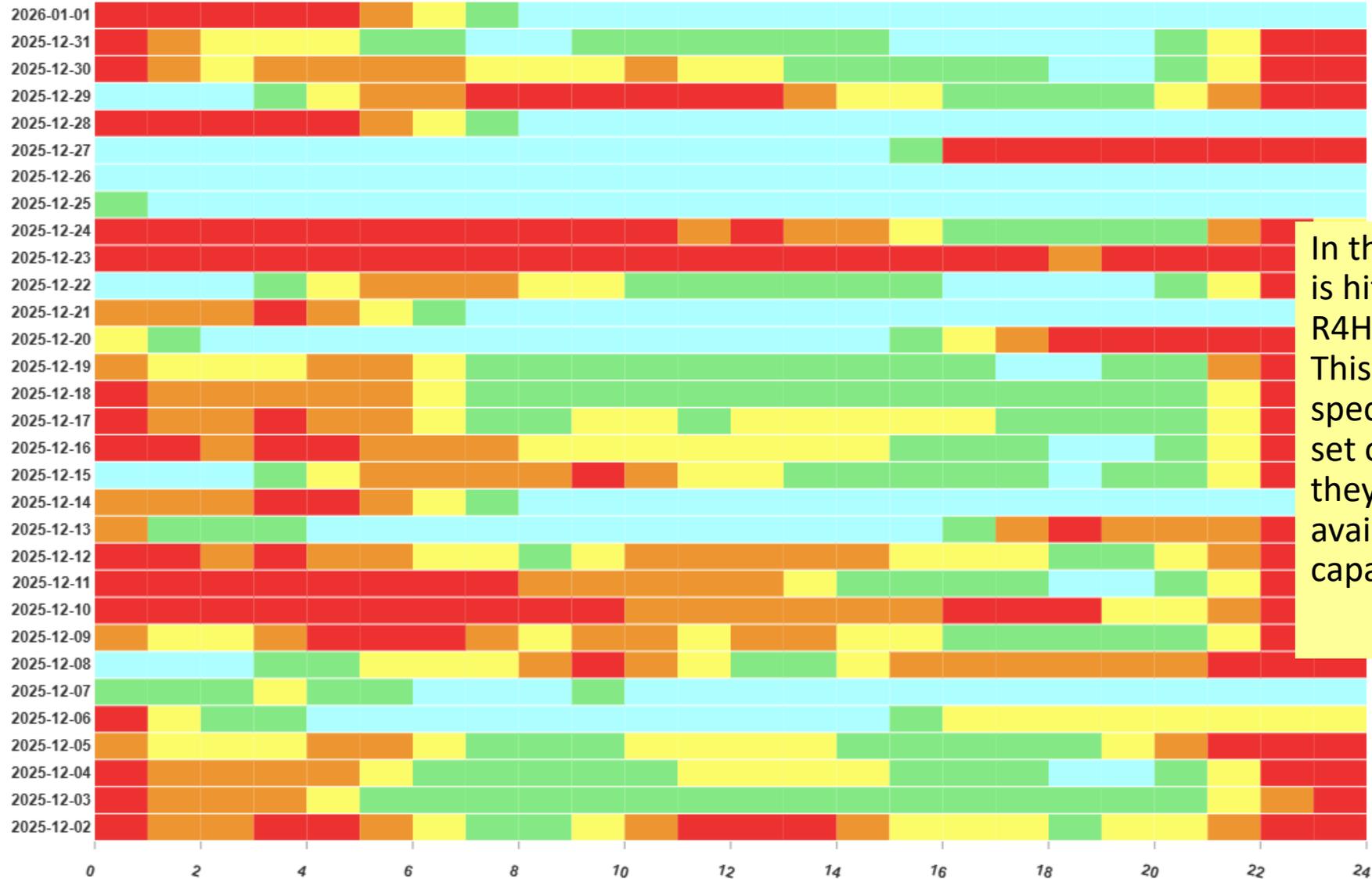
Understanding these answers is a basic precursor to any MLC cost reduction effort

- ≤ 60:
- ≤ 80:
- ≤ 90:
- ≤ 99:
- higher: Peak

Hourly Estimated Billable R4HA Usage

Percent of Peak R4HA

CEC, 0D70A, 0D70A

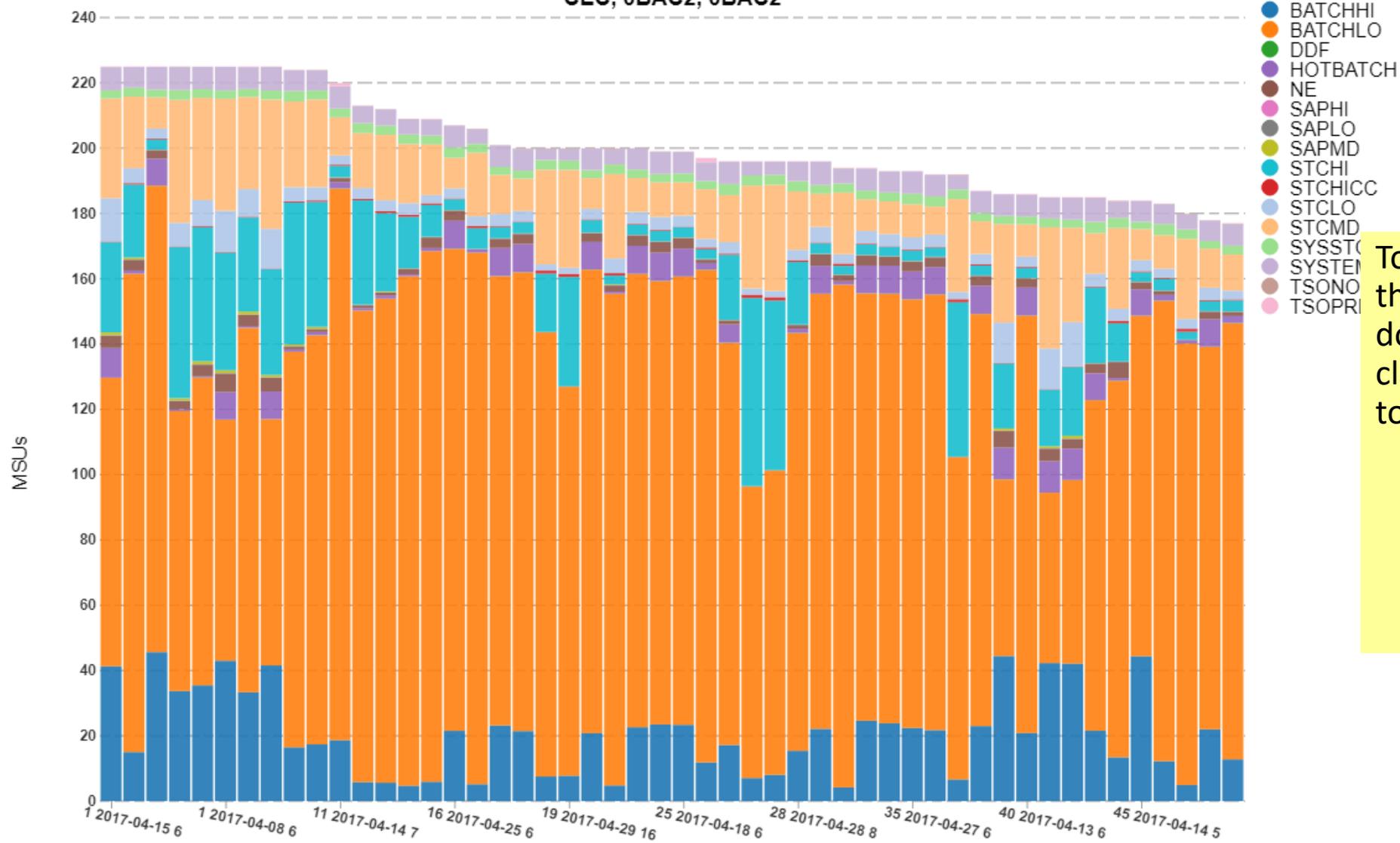


In this example, the CEC is hitting it's monthly R4HA peak most nights. This could be due to a specific cap they have set or could be that they're using all of the available machine capacity.

Top Service Class Workloads

Estimated Billable MSUs

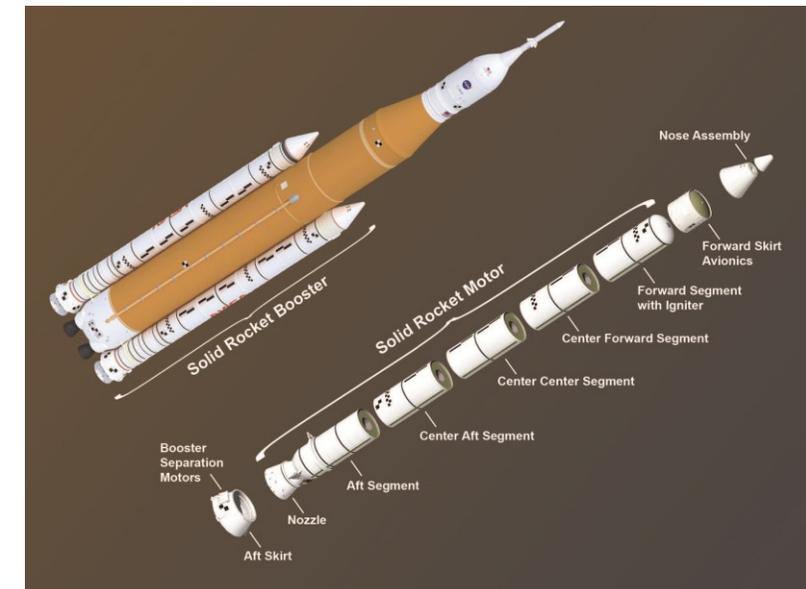
CEC, 0BAC2, 0BAC2



Top 50 intervals, with the R4HA MSUs broken down by the service classes that contributed to them.

#6: Optimize Your System Recovery Boost

A lot of sites aren't using this as much as they could



System Recovery Boost



- Overall goal: provide additional capacity and performance to better recover from certain planned or unplanned events
 - Shutdown faster
 - Startup faster
 - Faster DR site switch
 - Faster sysplex recovery
 - Faster subsystem restart
- Requires z15 or later machine (91% of our customers)
 - + z/OS 2.3 and up on z15, z/OS 2.4 and up on z16
- No additional charge for basic System Recovery Boost
 - Optional extra cost item not needed by vast majority of customers

Types of boost



- Speed Boost: *GP engines run as full speed engines*
 - Applicable only to sub-capacity systems
 - I.E. this is of no use to 8561-7xx and 8562-Z0x systems
 - Only boosted LPARs run as full speed!
- zIIP Boost: *zIIPs will run work not eligible to run on zIIPs*
 - I.E. **any** workload may be dispatched to a zIIP
 - Reserved (but physically available) zIIPs brought online to the boosted LPAR
 - Note that zIIPs always run full speed
 - What if you want more zIIPs for zIIP boost?



Boost classes = triggering events



Class	Where	Duration	WLM Work Routing
Shutdown	Single system	30 minutes	Routes work away from boosted system
IPL	Single system	60 minutes	Routes work to boosted system
Recovery Process	Multiple systems	<=5 minutes	No change

● Recovery Processes Boost

- CF data sharing member recovery
 - Triggered by disconnection from lock structure while locks were held
- Sysplex partitioning
 - System removed from a sysplex
- CF structure recovery
 - Structure rebuild, duplex failover, or reduplexing
- HyperSwap Recovery
 - Recovery from storage controller failover

New with z16:

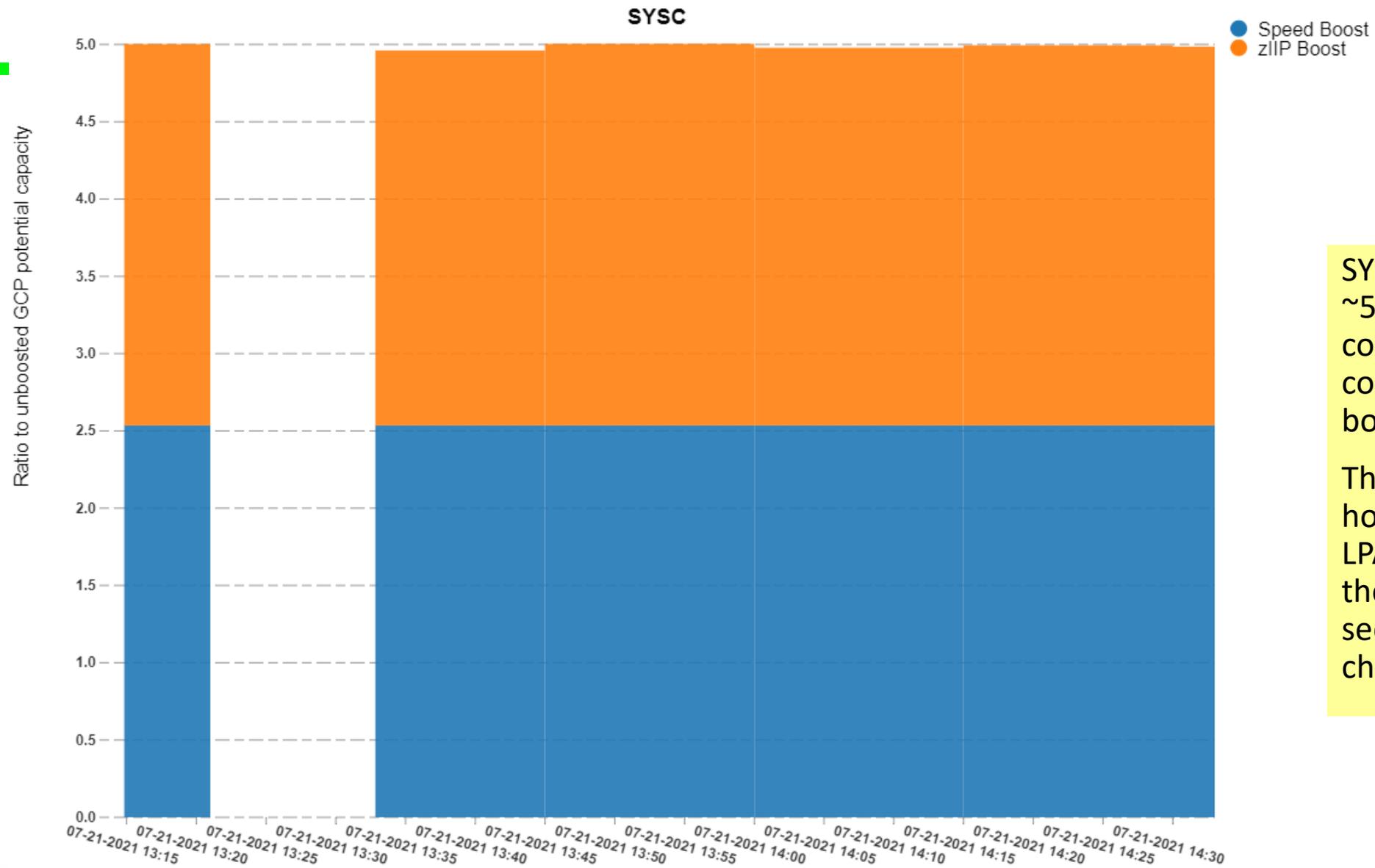
- SVC Dump
- HyperSwap Configuration Load
- **Middleware Region Startup**
- **Dynamic I/O Activation**

More people should be using more SRB



- Boosts you get by default, without really taking action:
 - IPL
 - Recovery process for things like sysplex partitioning
- **You must take action to get boosts for:**
 - Shutdown
 - Middleware Recovery
 - SVC Dump boosts
- We're not seeing much of the latter—why??
 - If you're on TFP, all the more reason to think about this
- The extra capacity available to a boosted LPAR can be significant!

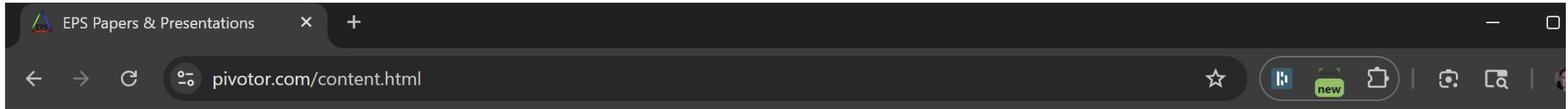
Boost Potential Capacity Ratio



SYSC could have done ~5x the GP work compared to what it could have done unboosted.

This report considers how much the other LPARs are using hence the slight variations you see in the ratios on each chart.

See also <https://pivotor.com/content.html>



Enterprise Performance Strategies Inc.

Creators of Pivotor®

- Home
- Pivotor
- Workshops
- Consulting
- Webinars
- Tools & Resources ▾
- About ▾

- [Download](#) [Scott Chapman - High, Medium, Low: Understanding how HiperDispatch Influences Performance](#)
- [Download](#) [Peter Enrico - Intensity! Understanding the Concepts and Usage of Intensity Measurements](#)

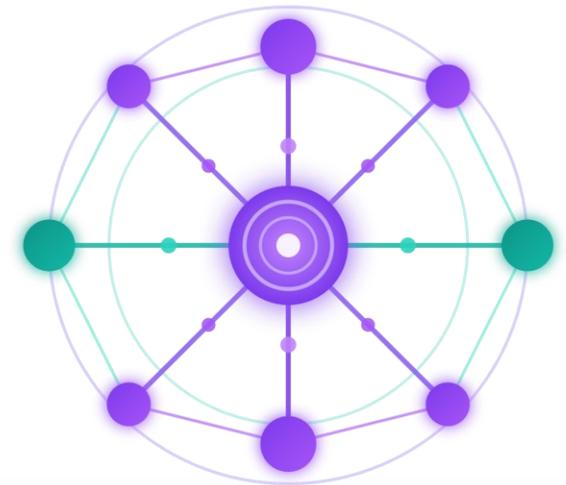
- WLM to HTML tool
- Our Speaking Schedule
- Our Presentations
- Our YouTube Channel
- Free Cursory Review
- Free Performance Reporting

2022

- [Download](#) [Scott Chapman - WLM Management of DDF Work: What can you do and what has changed?](#)
- [Download](#) [Scott Chapman - Understanding System Recovery Boost's Impact on Performance and Performance Reporting](#)
- [Download](#) [Peter Enrico - Key Reports to Evaluate z16 Processor Caches](#)



#7: Learn Something About AI



AI: Not just the next word



- The AI systems have gotten much more sophisticated and helpful
- Reasoning models can build a list of goals and work towards those goals
- Improved memory functions lets AI remember between sessions
- Skills, Tools, and MCPs help agentic AIs discover details and do real work

- AI is far from perfect and doesn't have your human judgement but...
- If you are a “knowledge worker” and expect to be working past the next couple of years, you need to start thinking about learning how to use AI
 - Using AI well is also a skill
 - You may not be replaced by an AI, but rather by somebody who is good at using AI

I've been experimenting...



- We've been using Windsurf for a couple of years for coding assistance
 - Pretty much always good, sometimes amazingly so
 - Sometimes writes more defensive code than I would have written
 - Occasionally avoiding problems I hadn't considered
 - Always a faster typer than I am (and I'm pretty fast)
- Shift in thinking from “how long will it take me to add this” to “the AI will probably generate a good first draft for review and testing in almost no time”
 - Much less programming, still testing/verifying
 - If you ask for the wrong thing, that may be what you get!



Personal AI Assistant



- Claude Code has a somewhat unique design that makes it well suited for supporting a customized and customizable “personal AI assistant”
- See: [https://github.com/danielmiessler/Personal AI Infrastructure](https://github.com/danielmiessler/Personal_AI_Infrastructure)
 - Still learning how best to use it
 - Best run under Linux, MacOS, or Windows Subsystem for Linux
 - Terminal user interface didn't work well for me directly in Windows
 - It can write/execute PowerShell script, but I know bash far better so I can keep a better eye on it
 - Similar but with *many more* security concerns: OpenClaw <https://openclaw.ai/>
 - I do not recommend giving an AI “carte blanche” to your email, calendar, Amazon account, etc.!
- I tried it initially with a personal project for monitoring our solar equipment
 - It helped work through and select the necessary infrastructure
 - Eventually I gave it (non-sudo!) access to my home linux server
 - It installed and configured multiple packages on its own
 - When I pointed out problems it would investigate, present alternatives, and implement the fix once I told it which way I wanted to go
 - I learned a bit about some things from it along the way, but not as much as if I spent more hours working on it (but I didn't spend more hours working on it!)

Example of me working a problem with my AI assistant “Blaise”*

Are the buckets named for the end of the interval or the start of the interval? E.G. in `ltss_daily` I see buckets of `"2026-01-13 19:00:00-05"` what time period does that represent? Specify it in my local timezone of -5 hours (NewYork)

• Good question - TimescaleDB `time_bucket` returns the start of the interval.

So `2026-01-13 19:00:00-05` represents:

	EST (your timezone)
Start	Jan 13, 7:00 PM
End	Jan 14, 7:00 PM

Why 7 PM? The daily bucket aligns to UTC midnight by default:

- UTC midnight (00:00) = 7:00 PM EST (previous day)
- So "January 14th UTC day" displays as "January 13th 7 PM EST"

This means your "daily" buckets are actually capturing 7 PM to 7 PM EST, not midnight to midnight in your local time.

If you want midnight-to-midnight EST buckets, the aggregate would need to use an offset:

```
time_bucket('1 day', time, INTERVAL '-5 hours')
```

I found the daily summary statistics confusing, although I thought I knew what was going on.

Want me to recreate the daily aggregate with EST-aligned buckets? It would make the data more intuitive - bucket `2026-01-13 00:00:00-05` would represent all of January 13th in your local time.

* I named it Blaise after Blaise Pascal then retconned to “Bespoke Learning And Intelligence Support Engine”

Don't trust that the AI's first answer is the best!

Does that need to change then during time change?

- Exactly the right concern. A hardcoded `-5` hours offset would be wrong during daylight saving time (EDT = UTC-4).

Better solution: TimescaleDB 2.0+ supports timezone-aware bucketing:

```
time_bucket('1 day', time, 'America/New_York')
```

This automatically handles DST transitions - it always buckets to local midnight whether you're in EST or EDT.

Let me check if your TimescaleDB version supports this:

- ```
Bash(ssh -i /home/scott/.ssh/id_ed25519 blaise@helium.we3 "docker exec timescaledb psql -U postgres -d ltss -c \"SELECT extversion FROM pg_extension WHERE extname =...)\nL extversion\n-----\n 2.24.0\n(1 row)\nL Running PostToolUse hooks... (1/2 done)
```

I appreciate the version check!

- ```
Bash(ssh -i /home/scott/.ssh/id_ed25519 blaise@helium.we3 "docker exec timescaledb psql -U postgres -d ltss -c \"SELECT time_bucket('1 day', time, 'America/New_York'...)\nL bucket | entity_id | avg_value\n-----+-----+-----\n  2026-01-15 00:00:00-05 | sensor.envoy_482534029601_current_power_production_log | 2.55\n... +5 lines (ctrl+o to expand)\nL Running PostToolUse hooks... (1/2 done)
```

- Version 2.24.0 supports it. See the difference - now buckets show `00:00:00-05` (local midnight):

Bucket	Represents
2026-01-15 00:00:00-05	All of Jan 15 (midnight to midnight EST/EDT)

The timezone parameter automatically adjusts for DST - during summer it would show `00:00:00-04`.

Options:

1. Recreate the aggregates with 'America/New_York' timezone – cleaner going forward
2. Keep current UTC-aligned aggregates – consistent but offsets from local midnight
3. Add new aggregates alongside existing ones – query whichever makes sense

I like that it came up with reasonable options, but I want more clarification...

Want me to recreate `ltss_hourly` and `ltss_daily` with timezone-aware bucketing? I'd drop the old ones and create new ones with the same names.

* Cooked for 31s

► If we take option #1 will they repopulate with the detailed data we currently have in `ltss`?

● Yes – continuous aggregates can be refreshed from existing raw data. Let me check what time range we have in `ltss`:

```
Bash(ssh -i /home/scott/.ssh/id_ed25519 blaise@helium.we3 "docker exec timescaledb psql -U postgres -d ltss -c \"SELECT min(time) as earliest, max(time) as latest, c...")
┌          earliest          |          latest          | total_rows
-----+-----+-----
 2026-01-04 20:33:13.777136-05 | 2026-01-15 20:46:02.765021-05 |      214242
(1 row)
```

● Yes. You have ~11 days of raw data (Jan 4 – Jan 15, 214K rows). After recreating the aggregates, we'd run:

```
CALL refresh_continuous_aggregate('ltss_hourly', '2026-01-04', '2026-01-16');
CALL refresh_continuous_aggregate('ltss_daily', '2026-01-04', '2026-01-16');
```

This recomputes the aggregates from the raw `ltss` data with the new timezone-aligned buckets.

The process:

1. Drop old aggregates (`ltss_hourly`, `ltss_daily`)
2. Create new ones with `time_bucket('1 day', time, 'America/New_York')`
3. Refresh to backfill from existing raw data
4. Set up refresh policies for ongoing data

Note that it developed a plan! I told it to go ahead and do it

No data loss – the raw `ltss` table is untouched.

Want me to proceed?

```
● Bash(ssh -i /home/scott/.ssh/id_ed25519 blaise@helium.we3 "docker exec timescaledb psql -U postgres -d ltss -c \"DROP MATERIALIZED VIEW IF EXISTS ltss_hourly CASCADE...")
  L DROP MATERIALIZED VIEW
  DROP MATERIALIZED VIEW
  NOTICE: drop cascades to table _timescaledb_internal._hyper_2_2_chunk
  NOTICE: drop cascades to table _timescaledb_internal._hyper_3_4_chunk
  L Running PostToolUse hooks... (1/2 done)
● Bash(ssh -i /home/scott/.ssh/id_ed25519 blaise@helium.we3 "docker exec timescaledb psql -U postgres -d ltss -c \"
  CREATE MATERIALIZED VIEW ltss_hourly...)
  L CREATE MATERIALIZED VIEW
  L Running PostToolUse hooks... (1/2 done)
● Bash(ssh -i /home/scott/.ssh/id_ed25519 blaise@helium.we3 "docker exec timescaledb psql -U postgres -d ltss -c \"
  CREATE MATERIALIZED VIEW ltss_daily...)
  L CREATE MATERIALIZED VIEW
  L Running PostToolUse hooks... (1/2 done)
● Bash(ssh -i /home/scott/.ssh/id_ed25519 blaise@helium.we3 "docker exec timescaledb psql -U postgres -d ltss -c \"CALL refresh_continuous_aggregate('ltss_hourly', '20...")
  L CALL
  CALL
  L Running PostToolUse hooks... (1/2 done)
● Bash(ssh -i /home/scott/.ssh/id_ed25519 blaise@helium.we3 "docker exec timescaledb psql -U postgres -d ltss -c \"
  SELECT add_continuous_aggregate_policy('ltss_hourly...')
  L add_continuous_aggregate_policy
  -----
  1004
  (1 row)
  L Running PostToolUse hooks... (1/2 done)
```

There was more steps in here that I haven't shown. But note that it's logging into the server, using docker to issue commands against timescaledb in a container.

```
● Bash(ssh -i /home/scott/.ssh/id_ed25519 blaise@helium.we3 "docker exec timescaledb psql -U postgres -d ltss -c \"SELECT bucket, round(avg_value::numeric, 2) as avg,...)
L      bucket      | avg | sample_count
-----+-----+-----
  2026-01-15 20:00:00-05 | 0.00 |          1
... +5 lines (ctrl+o to expand)
L Running PostToolUse hooks... (1/2 done)
```

● Done. Aggregates recreated with timezone-aware bucketing.

✓ Results:

Daily - now shows local midnight:

Bucket	Represents	Avg Production
2026-01-15 00:00:00-05	All of Jan 15 EST	2.55 kW
2026-01-14 00:00:00-05	All of Jan 14 EST	0.46 kW
2026-01-13 00:00:00-05	All of Jan 13 EST	1.10 kW

Hourly - also aligned to local hours:

Bucket	Avg Production
2026-01-15 17:00:00-05	0.10 kW
2026-01-15 16:00:00-05	2.33 kW

Refresh policies configured:

- Hourly: refreshes every hour
- Daily: refreshes every day

These will automatically adjust for DST - during summer the timestamps will show -04 offset but still represent local midnight/hours.

At the end, a nice summary of was done. I also had it write documentation about how the whole process works.

Final AI notes



- I'm not crazy: I didn't give it sudo access
- Human expertise and judgement still quite useful
 - AI can make good decisions but may not make the optimal decision by itself
 - Humans don't always make optimal decisions by themselves either!
 - Understanding the broader system design may be challenging for AI due to context length limits
 - Regardless, velocity of IT stuff will accelerate
- I don't know how this is going to impact our jobs as z/OS performance analysts, but it seems like it will
- I think having access to an AI assistant will be a key productivity aid
 - Tony Stark's Jarvis doesn't seem far-fetched today!
- If you haven't seen it: <https://shumer.dev/something-big-is-happening>

Resolution Summary

