WLM Turns 30! : A Retrospective and Lessons Learned

Peter Enrico

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

Creators of Pivotor®

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 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

Abstract

WLM Turns 30! : A Retrospective and Lessons Learned

This year marks the 30th anniversary of the announcement and availability of WLM. Let's celebrate with a retrospective of why this feature of MVS was a game changer in the world of computer performance management. What were the motivating factors, evolution of WLM, and lessons learned?

This presentation will be given by Peter Enrico, who was an original member of the WLM algorithms design and development team. The more senior attendees will listen nostalgically. Newer performance professionals will gain insights to the evolution and maturity of WLM.

This is going to be a great 30-year review and will be full of insights.

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
- **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!

EPS presentations this week

What	Who	When	Where
60 Years of Pushing Performance Boundaries with the Mainframe	Scott Chapman	Sun 17:00	Neptune D
Introduction to Parallel Sysplex and Data Sharing	Peter Enrico	Mon 13:15	Pomona
Macro to Micro: Understanding z/OS Performance Moment by Moment	Scott Chapman	Mon 15:45	Neptune D
WLM Turns 30! : A Retrospective and Lessons Learned	Peter Enrico	Tue 10:30	Neptune D
PSP: z/OS Performance Spotlight: Some Top Things You May Not Know	Peter Enrico Scott Chapman	Tue 13:00	Pomona
More/Slower vs. Fewer/Faster CPUs: Practical Considerations in 2024	Scott Chapman	Tue 14:15	Neptune D
z16 SMF 113s – Understanding Processor Cache Counters	Peter Enrico	Wed 13:15	Pomona

What it means to be 30

- We often celebrate landmark anniversaries such as 30 years

- 30th anniversaries are just moments in time during which we tend to
  - Look back and reflect from where we came
  - How we got here
  - Step back and assess the current state of affairs
  - Look into the future

- MVS 5.1 – First official release of the MVS Workload Manager (WLM)
  - Available 1994

The Conception of WLM

Everything has a beginning…

- If this is the 30th anniversary of WLM, then when was the conception?
- The furthest back that I traced WLM was to the following paper:

  An Automated Workload Manager for MVS
  - Author: Bernie Pierce (IBM)
  - 1981 Internal IBM paper
- In my humble opinion, nearly every discussion of the history of WLM, z/OS, and z/OS performance management should pay tribute to **Bernie Pierce**
- In the context of this presentation, Bernie is the father of WLM
  - The names of 'Bernie Pierce' and 'WLM' and z/OS performance are synonymous

Who was Bernie Pierce?

- Interested in learning more about Bernie and WLM? Suggested reading:
  - Any of Bernie's many papers or presentations
    - Specifically, 1995 CMG paper titled:
      The Evolution of the SRM to the Workload Manager in MVS V5
  - Bernie's many patents

- On a personal note,
  Bernie was my IBM mentor and friend

Bernie Pierce

- In my humble opinion, z/OS is alive and well today due in a very large part to Bernie Pierce
  - Many view Bernie as the best performance designer there was

Bernie Pierce 1998 CMG (www.cmg.org) A. A. Michelson Award recipient

- In 1998 CMG recognized Bernie Pierce with the A.A.Michelson Award

- I specifically note this here since I heavily refer to Bernie's acceptance speech in this presentation
  - Full text of Bernie's acceptance speech is available on our website with permission of June Pierce and the Pierce family

https://www.epstrategies.com/BerniePierce.pdf

CMG President Thomas Dennison awarding Bernie Pierce the 1998 A.A. Michelson award

In addition to Bernie, Some really smart people…

- In the early 1990s a lot of people contributed to the design and development of the first few releases of the workload manager (WLM)

- In addition to Bernie Pierce, a shout out to some key technical designers:
  - It is always risky to name names since some of the first key MVS technical designers may not be included or not want to be acknowledged, but my mind always gravitates to IBMers:

    Cathy Eilert, Gary King, Peter Yocom, John Arwe, Dave Emmes, Jeff Aman, Gus Kassimis

    (and this list does not even mention the many developers, testers, bench-markers, level-2 support, project managers, documentation folks, or folks working in subsystem products such as CICS, DB2, IMS, etc.)

  - Many others followed in their footsteps, most notable (in my mind) is Robert Vaupel

My connection to WLM

- For those of you that do not know much about me, before I left IBM, I was a member of the original WLM algorithm's design and development team

- In the late 1980s and early 1990s I had been working in the RMF design and development group doing SMF measurement architecture
  - Sometime in 1991 IBM made the decision to move all RMF development to IBM Germany
  - I was out of an assignment!

- About 30 minutes after the announcement to move RMF was made, I received a phone call from Bernie Pierce telling me there was a new team forming to develop something call the MVS Workload Manager.
  Bernie asked me if I would be interested in joining the WLM algorithms development team.

- Thank-you Bernie!

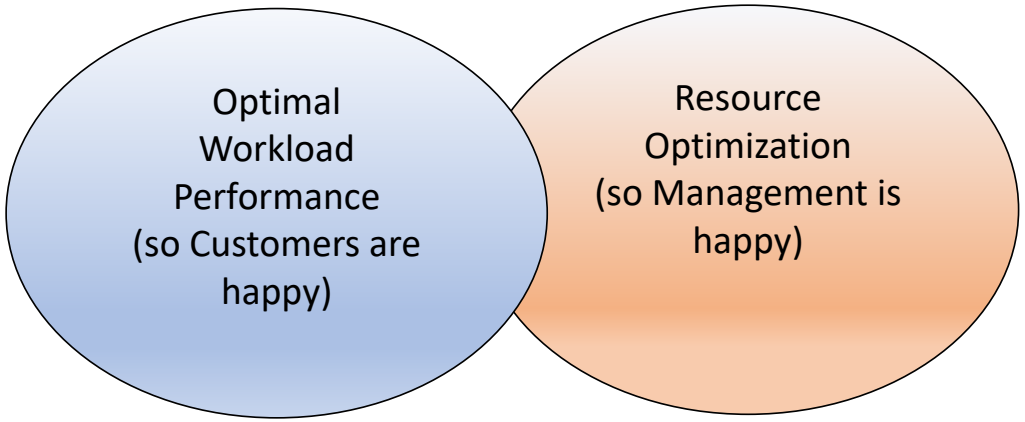
Most of the IBM MVS 5.3 WLM Team (about 1995 or 1996)

Why a Workload Manager?

What finally got WLM into the budget for development?

Workload Management 101

- Performance analysts, capacity planners, and a 'workload manager' have two primary responsibilities



- The problem is that these tend to be two conflicting objectives
  - The challenge is to find the optimal balance
  - This becomes even more difficult on systems with a variety of ever-changing workloads that are part of a multisystem environment

Prior to MVS 5.1

- SRM: the system resource manager of MVS that controls access to system resources
- Before WLM the systems resources were controlled by a static set of controls which today is known as ‘compatibility mode’
  - Parmlib IEAICSxx
  - Parmlib IEAIPSxx
- Without going into too much detail, just know the controls were **awful**
  - For those of you that are really interested, Google search
    - **MVS/ESA Initialization and Tuning Guide** (GC28-1828)
    - **MVS/ESA Initialization and Tuning Reference**
    - Make sure you look at the MVS/ESA version of these manuals and not the newer z/OS versions
- BTW – In general, the MVS and z/OS Initialization and Tuning Guide were, and still are, amazing resources for the research of z/OS performance

IEAIPSxx and IEAICSxx members of parmlib

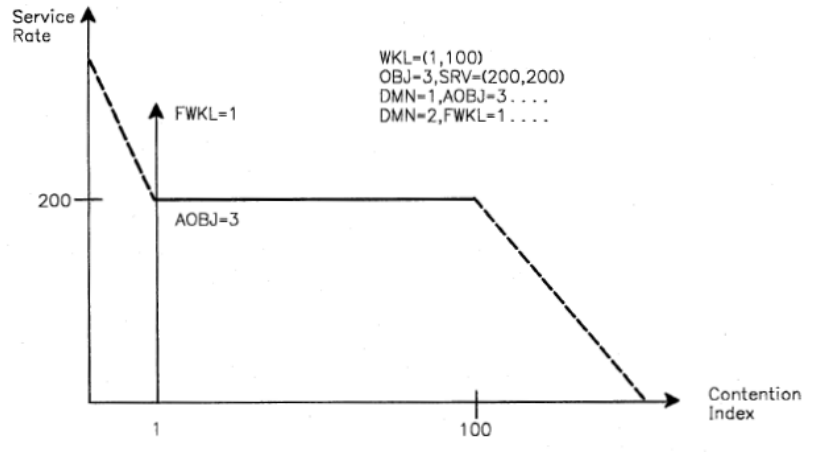
- The IPS and ICS static controls allowed system programmers to tell the system how the work was to be assigned system resources

IPS concepts:

- Domains
- Service definition coefficients
- Service rates
- Performance objectives and workload levels
- Interval service values
- Dispatching priorities
- I/O priorities
- Storage isolation
- Response-throughput bias
- TSO response time control
- Domain importance
- Performance periods
- Performance groups

ICS - Installation control specification concepts:

- Performance group assignment for control
- Performance group assignment for reporting



Bernie's ***CMG A.A.Michelson*** award acceptance speech

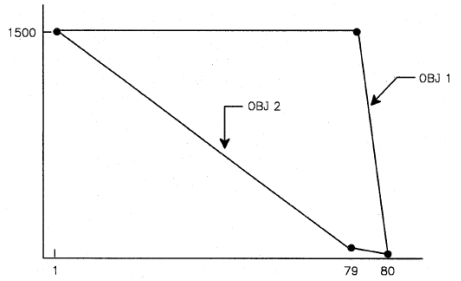

Bernie Pierce

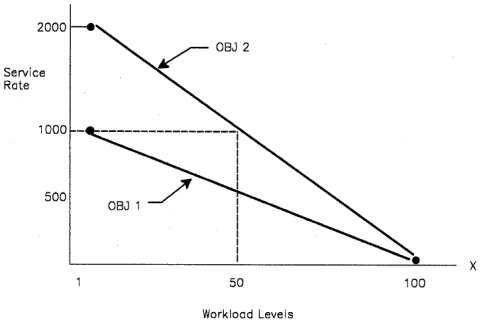
- Excerpt describing a major rewrite of SRM and the 'parameter factory' (which occurred in late 1970s and early 1980s)
  - Full text available on our website with permission of June Pierce https://www.epstrategies.com/BerniePierce.pdf
  - *"The new SRM was improved because it was now more controllable but some of its decisions were quite questionable in my opinion. I soon had my opportunity to apply my opinions and Little's Law to the new improved SRM as a member of the development team. I worked there for several years. I like to call this time in the parameter factory. We could produce parameters like you would not believe. We provided two different means to set dispatching priorities and then added time slicing on top for the advanced user. We added storage isolation, new load balancers, I/O priorities and logical swap controls just to name a few. It was really getting out of hand."*

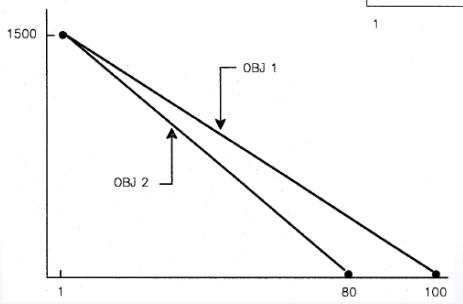
Old SRM parameters

- There were static switches and objective curves

```
APGRNG (0-15)                              /* ALL DISP PRTY IN APG    */
PVLDP = F54                                /*PRIVILEGED USER DPRTY    */
IOQ = PRTY                                 /*                         */
CPU= 10.0,IOC = 5.0,MSO = 3.0,SRB = 10.0   /*                         */
/*                                                                    */
/*                                                                    */
/* OBJ1---(DEFAULT) - A STEEP SLOPE                                   */
/*            (USED WHERE EXCHANGE SWAP CONTROL IS NOT REQUIRED       */
/*            TO EQUALIZE RESPONSE TIMES AND THROUGHPUT).             */
/* OBJ2---MODERATE SLOPE                                              */
/*            (USED TO EVENLY DISTRIBUTESERVICE TO LONG BATCH         */
/*            AND LONG TSO TRANSACTIONS).                             */
/* OBJ3&4 - ONE SLOPE IS TWICE THE OTHER                              */
/*            (TO GIVE TSO LONG TWICE THE SERVICE RATE OF BATCH       */
/* OBJ5---A HORIZONTAL LINE                                           */
/*            (USED FOR IMPORTANT WORK TO PREEMPT OTHER BATCH)         */
/* OBJ6---UNFAVORABLE SLOPE                                           */
/*            (USED FOR UNIMPORTANT WORK THAT SHOULD BE DELAYED)      */
/*                                                                    */
WKL=(1,50,99,100)                                                     */
OBJ=2,SRV=(2000,*,0)                     /*LONG BATCH AND LONG TSO   */
OBJ=3,SRV=(2000,*,*,0)                   /* DOBJ FOR TSO             */
OBJ=4,SRV=(1000,*,*,0)                   /* DOBJ FOR BATCH           */
OBJ=5,SRV=(2000)                         /*HOT BATCH                 */
OBJ=6,SRV=(2000,1,0)                     /*LOW PRIORITY BATCH        */
```



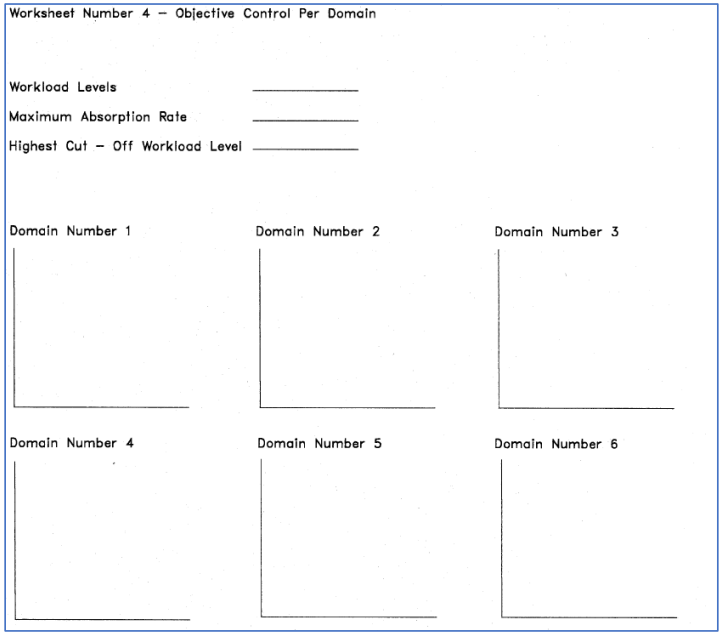




Old SRM parameters

- There were domains, performance group and periods

```
/*ALL DOMAINS HAVE DEFAULTED MINIMUM AND MAXIMUM MPL (1,255)          */
/*BUT EXPLICITLY INDICATE THEIR CONTENTION INDEX ALGORITHM.           */
/*                                                                   */
/*                                                                   */
DMN=1,DOBJ=4                              /*BATCH                    */
DMN=2,FWKL=128                            /*SHORT AND MEDIUM TSO     */
DMN=3,DOBJ=3                              /*LONG TSO                 */
/*                                                                   */
/*                                                                   */
/* DOMAINS PROVIDES ACCESSIBILITY TO MAIN STORAGE.                   */
/* ISV (DEFAULT lOOK) ENSURES RESIDENCY IN MAIN STORAGE FOR MOST,    */
/* BUT ALLOWS EXCHANGE SWAPS FOR LONG TSO TRANSACTIONS.              */
/* DURATIONS ALLOW CONTROL PARAMETERS TO CHANGE AS TRANSACTIONS AGE.*/
/* RESPONSE THROUGHPUT BIAS FAVORS RESPONSE FOR ALL                  */
/* OBJECTIVES ARE USED AS DESCRIBED ABOVE.                           */
/*                                                                   */
/*                                                                   */
PGN=1,  (DMN=1,DP=M2,DUR=30K)                 /*BATCH - SHORT       */
        (DMN=1,DP=M2,OBJ=2)                   /*----- - LONG        */
PGN=2,  (DMN=2,DP=F34,DUR=200)                /*TSO   - SHORT       */
        (DMN=u,DP=F32,DUR=800)                /*----- - MEDIUM      */
        (DMN=3,DP=M2,0BJ=2,ISV=lOK)           /*----- - LONG        */
PGN=3,  (DMN=1,DP=F30,0BJ=5)                  /*HOT BATCH           */
PGN=4,  (DMN=l,DP=Ml,OBJ=6)                   /*LOW PRTY BATCH      */
```


Worksheet Number 4 – Objective Control Per Domain

Workload Levels _____

Maximum Absorption Rate _____

Highest Cut – Off Workload Level _____

Domain Number 1 | Domain Number 2 | Domain Number 3

Domain Number 4 | Domain Number 5 | Domain Number 6

And you had worksheets to figure it out

- And there were worksheets
- All so you could
  - Ensure workload performance
  - Resource optimization
- And these parms were not even the worst parts!

Performance Group Number		Domain Number	Objective	Dispatching Priority DP	TSDP	TSGRP	I/O Priority	Interval Service Value	Duration	RTB	UNT
Defaults →		See Note 1	OBJ–1	M0	None	Domain Number	IOP–DP or –TSDP	ISV–100K	None	0	S
1 Batch Default	Period 1										S
	2										
	3										
	4										
	5										
	6										
	7										
	8										
2 TSO Default	Period 1										
	2										
	3										
	4										
	5										
	6										
	7										
	8										

N/A indicates that assigning a PGN is not meaningful

Subsystem	Transaction Class/Userid/Name	Control PGN[1]	Optional Control PGNs	Report PGN
TSO	TRXNAME=	N/A	N/A	
	USERID=			
	ACCTINFO=			
STC	TRXNAME=			
	TRXNAME=			

The controls were static!

- In addition to being complex, the IEAIPSxx and IEAICSxx parameters were static controls to tell the MVS operating system how to apply resources to the workloads rather than a desired goal for the work

- The problem with complex static controls?
  - Difficult to regularly adapt to ever changing workloads, machines, peaks and valleys, general availability of resources, etc.
    - Most customers used a single set of controls or controls for each shift… but still, they were static
  - Performance analysts spent countless hours either updating the controls, settling for the old controls even in new circumstances, or toggling among some preset definitions
  - Nothing was dynamic
  - Add to this new performance analysts trying to figure out what the previous performance person was trying to do. Nothing was transparent.

How did the MVS platform get into this mess?

- What makes workload management on MVS so unique is the diversity of workloads that other platforms did not have, and still do not have
  - Workloads that have unique objectives, resource requirements, and importance levels
  - Example:
    - Interactive workloads versus background workloads
    - Long running system address spaces doing work on behalf of system and short system requests
    - Long running server address spaces processing part of a transaction before handing the rest to another server address space
    - Long transactions and short transactions
    - Nighttime versus daytime versus month end versus…
    - Etc.

- Management of S/390's diverse workloads require diverse controls
  - If the workloads were all the same, then would probably need just a small number of parameters to control (but this is not the case for the MVS platform)

Bernie's ***CMG A.A.Michelson*** award acceptance speech


Bernie Pierce

- Excerpt describing his 1981 paper conceiving the WLM
  - Full text available on our website with permission of June Pierce
    https://www.epstrategies.com/BerniePierce.pdf

  - *"In 1981 I proposed that the SRM could be significantly less complex using response time as a primary external with internal algorithms that would adjust resource priorities or allocations such as dispatching priority, storage isolation limits etc. based on these simpler goals. The adjustments would be based primarily on profiles of delay as a function of resource allocation (or priority) obtained by state sampling. The proposal defined the vision of the Workload Manager (WLM). The effort had considerable risk and was quite expensive. Although SRM was often cited as an example of MVS complexity, there was no compelling reason to invest; the proposal was put on the shelf for a number of years."*

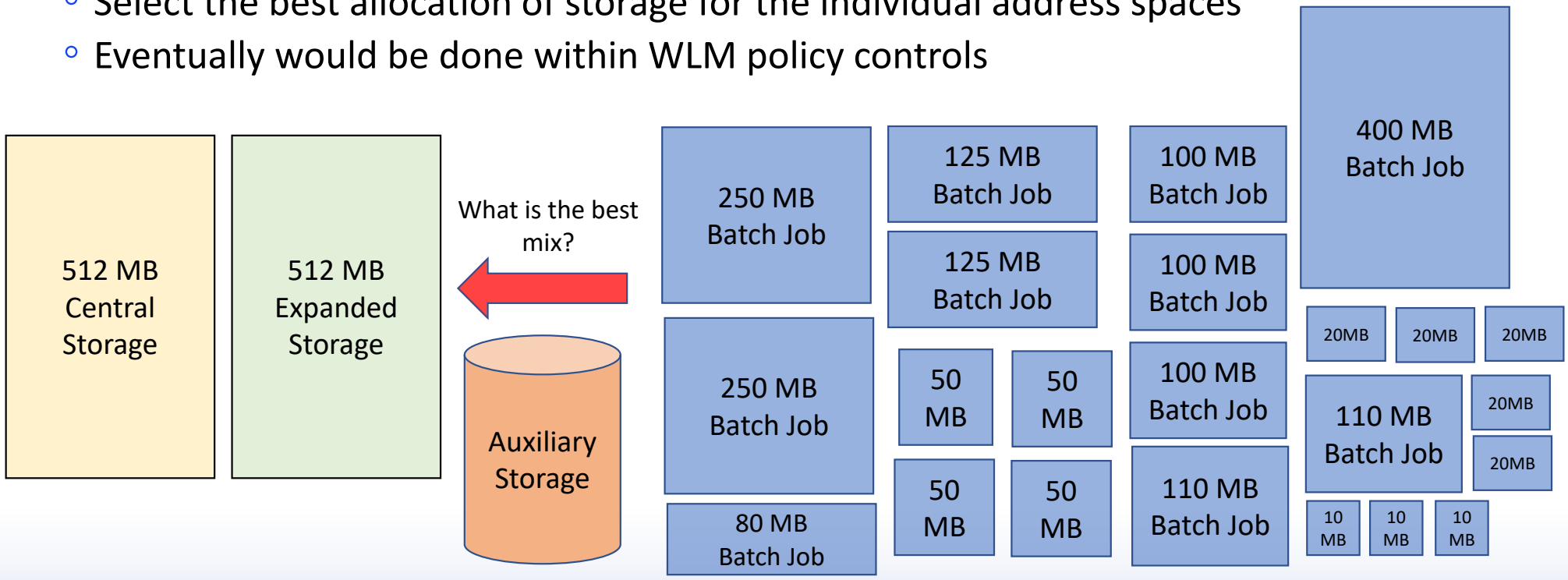
Then in the early 1990s something happened…

- Bernie Pierce and Cathy Eilert invented, designed, and developed a new MVS performance feature titled Working Set Management (WSM)
  - MVS 4.2 was a significant pre-WLM release of MVS since it introduced algorithms for Working Set Management
  - z/OS 2.3 z/OS MVS Initialization and Tuning Guide
    - *Working Set Manager: SRM automatically determines the best mix of work in the multiprogramming set (MPS) and the most productive amount of central storage to allocate to each address space within MPL constraints.*
- *Fun comment from John Arwe (WSM designer/developer):*
  - *"WSM made it into MVS 4.2, by the skin of its teeth. The SRM team lead escalated Bernie to the lab director to get WSM stopped/moved to MVS 4.3 because it wasn't following the usual MVS development process. You want to talk about iterative development before it was Kool? Over a period of 8 weeks in 1990, we averaged 2 iterations **per working day** from design through native performance test, on kernel-level code."*

John Arwe

Overview of Working Set Management (WSM)

- Select the best mixture of work for the multiple programming set to avoid storage thrashing and optimize storage
  - Select the best allocation of storage for the individual address spaces
  - Eventually would be done within WLM policy controls



Bernie's ***CMG A.A.Michelson*** award acceptance speech


Bernie Pierce

- Excerpt describing Working Set Management
  - Full text available on our website with permission of June Pierce
    https://www.epstrategies.com/BerniePierce.pdf

  - *"**Catherine Eilert** and I invented "working set management" to address these complicated issues. The invention allowed the SRM to "plot" paging behavior as a function of pages resident in one or more levels of the hierarchy. Productive CPU processing ability at a particular storage allocation was also plotted. This allowed SRM to control working set in central to avoid the damaging effects of pure LRU [Least Recently Used] which reduced the working sets of well behaved applications as it allowed large working sets with no value to large applications. The selection of address spaces to enter the multiprogramming set was also heavily influenced by the knowledge of the characteristics of applications. "*

Bernie's ***CMG A.A.Michelson*** award acceptance speech


Bernie Pierce

- Excerpt describing role of WSM to justify IBM investment into WLM
  - Full text available on our website with permission of June Pierce
    https://www.epstrategies.com/BerniePierce.pdf

  - *"The working set management effort proved very valuable; it served as the proof of concept of the proposal initially offered in 1981 that SRM controls could be simplified by adding significant additional profiling information and heuristic algorithmic approaches to resource management. The parallel Sysplex initiative provided the business case, dramatically increasing the complexity of the MVS environment. The Workload Manager was a very ambitious project. My primary role in the implementation was in application characteristic profiling and heuristic algorithms collaborating with Gary King and Catherine Eilert. An overview of the WLM approach can be understood from my 1995 CMG paper "The Evolution of the SRM to the Workload Manager in MVS V5" published in the Winter 95 Transactions.*

What was delivered in the first releases of WLM?

Note: This presentation is not about WLM and its controls

For more information on WLM there are lots of papers, presentations, manuals, and articles available

But please consider attending my WLM workshop for a productive week of work learning about WLM, analyzing your own measurements and WLM service definition.

MVS 5.1 introduced WLM

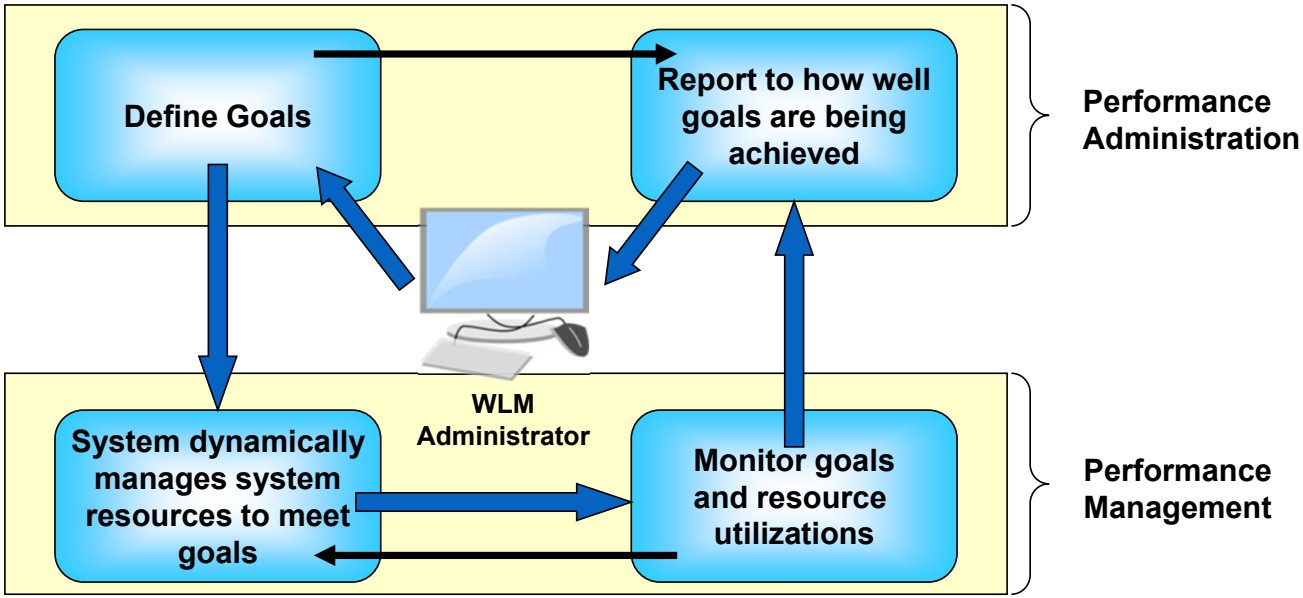
- With the z/OS Workload Manager (WLM), installations:
  - Define performance goals and assign a business importance to each goal
  - System determines how much resource should be given to work to meet goals

- John Arwe pointed out to me that WLM was one of the few major enhancements in MVS 5.1 that benefited all existing customers.

  This was not true for Parallel Sysplex.

  MVS 5.1 was also the parallel Sysplex release which was geared towards biggest leading edged customers and required a significant $$$ investment in hardware

  WLM was for everyone!



WLM introduced the externals which we are well familiar with today

- Users can edit their WLM service definition via ISPF (now also z/OSMF)
  - To tell WLM what you want the work to achieve and not how to run it

- MVS 5.1 introduced concepts of
  - Service Definition
  - Policies
  - Service classes and periods
  - Report classes
  - Classification rules
  - Goals
    - Velocity goals
    - Average Response time goals
    - Percent Response time goals
    - Discretionary goals
  - Resource groups
  - Performance Indexes
  - And much more

```
File  Utilities  Notes  Options  Help
------------------------------------------------------------------------------
Functionality LEVEL025 Definition Menu WLM Appl LEVEL025
Command ===> ____________________________________________________________

Definition data set . . : none

Definition name . . . . . ________ (Required)
Description . . . . . . . ________________________________

Select one of the following options.
___ 1. Policies                              12. Tenant Resource Groups
    2. Workloads                             13. Tenant Report Classes
    3. Resource Groups
    4. Service Classes
    5. Classification Groups
    6. Classification Rules
    7. Report Classes
    8. Service Coefficients/Options
    9. Application Environments
   10. Scheduling Environments
   11. Guest Platform Management Provider
```

Example of z/OS 2.3 WLM ISPF panel

MVS 5.1 Delivered Base Algorithmic Functionality

- SRM, which controls access to system resources, still exists with WLM
  - SRM controls are now dynamically set by WLM algorithms
- Internally to WLM there are a series of algorithms that are used to ensure the installations performance objectives are met

  Policy adjustment – algorithms to help the workloads meet their assigned goals

  Resource adjustment – algorithms to help optimize the use of system resources

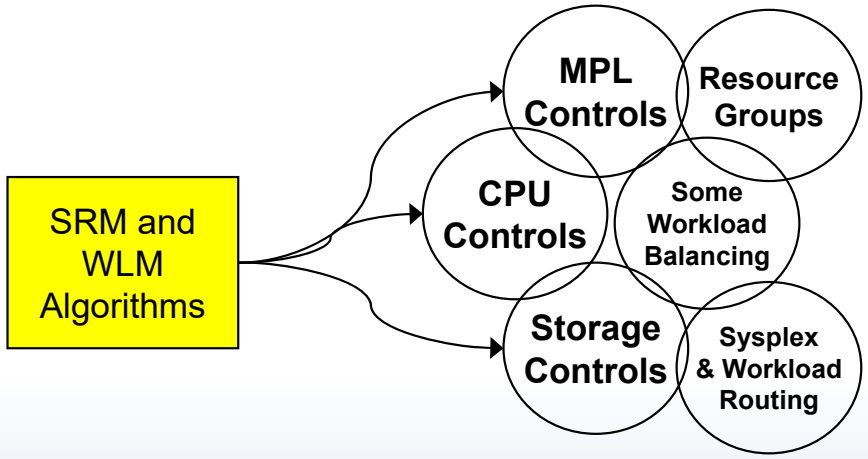
Processor:
- Dispatching priority
- Resource group capping

Sysplex:
- Workload balancing
- Routing suggestions

Working Set Manager
- Work within policy controls

Central Storage:
- MPS slots
- Logical swap slots
- Protected frames

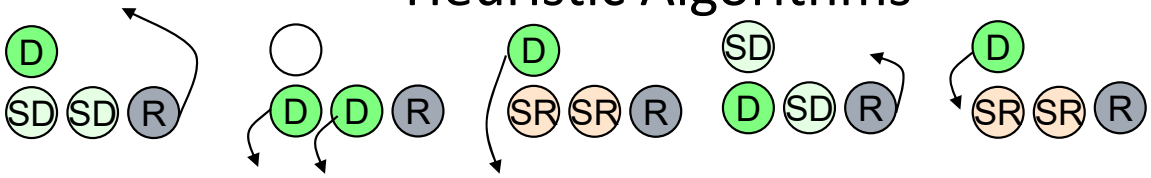
Expanded Storage:
- Swap slots
- Protected frames
- LRU access
- Space available access



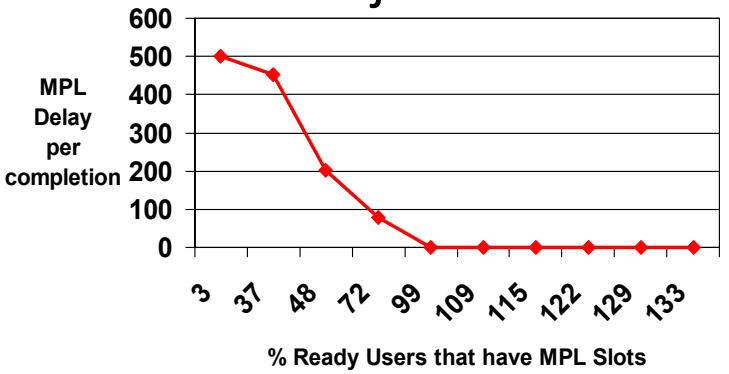
So, if you looked inside WLM…

- You would see some interesting data structures used by the heuristic algorithms to dynamically manage the workloads and system resources

Heuristic Algorithms



Projection Plots



Lots of Formulas

$$\frac{\text{CPU Using} + \text{I/O Using}}{\left(\begin{array}{l}\text{CPU Using} + \text{I/O Using} + \text{CPU Delay} + \text{I/O Delay} \\ + \text{Paging Delays} + \text{Swapping Delays} + \text{MPL Delays}\end{array}\right)}$$

Histories

Last 10 Seconds
Last 30 seconds
Last 60 seconds
Last 5 minutes
Last 15 minutes
Previous 15 minutes

Distributions

Bucket	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Width	<=50%	60%	70%	80%	90%	100%	110%	120%	130%	140%	150%	200%	400%	>400%
Trans Count	0	85	240	365	260	100	50	20	25	20	25	0	0	0

Question: How do you develop heuristic algorithms?

- The algorithms design and development team was kept small
  - The mindset at the time was to keep the team small to ensure the communication of a tight working group
  - Initially, the MVS 5.1 algorithms development team consisted of about 5 or 6 developers supported by several performance bench markers and testers
    - The team grew in the months before release
  - Fun quote from John Arwe: *"When they recruited for WLM in what became 5.1, Cathy and Bernie talked about us like we were a separate company."*
  - Quote from me (Peter Enrico):
    "All of us had a sense we were doing something special."
    - We also worked a lot of hours!
  - The externals design and development team was much larger
    - APIs, ISPF panels, doc, console commands, couple dataset, WLM address space, policy activation, etc.


John Arwe


Peter Enrico

Question: How do you test heuristic algorithms?

- Designing and coding heuristic algorithms is hard enough…
  - but how do you know they are working?
    Are they effectively managing work towards goals while optimizing resources?
  - Especially considering an environment like MVS with so many diverse workloads, workload requirements, resource constraints, etc., these need to work
  - How do you automate something like WLM and then validate it is working?
- Bernie Pierce and Cathy Eilert created the ***Near Type 1 Prototype Process***
  - Type 1 is code that is just about 'ready to ship',
    but '**Near Type 1 Prototype**' methodology means to treat code under development like 'Type 1' code
- In most ways, testing was like most standard code dev cycles, but key differentiators included:
  - Experiments designed to stress diverse workloads being managed together
  - Early in the dev cycle testing is done on actual hardware rather than simulated environments
  - Performance testing was done extremely early and continuous
  - Customer input was key

How do you test heuristic algorithms?

- Traditional testing was done to shake out code defects, but examples of performance test cases to shake out algorithm issues included scenarios such as:

Diverse Workload Scenarios

- All traditional multiple period TSO
- All traditional batch
  - Single and multiple period
- All tradition CICS
  - WLM managed and non-WLM managed
- All tradition IMS
  - WLM managed and non-WLM managed
- Mixture of TSO with Batch
- Mixture of Batch and CICS
- Mixture of TSO and Batch and CICS
- Etc.

Resource and Control Variations

- Easy paging environments
- Hard paging environments
- Everything running at equal importance
- Easy goals and hard goals
- CPU constrained environments
- Working set management
- Etc.

Bernie's ***CMG A.A.Michelson*** award acceptance speech

- Excerpt describing looking back at complexity of coding a WLM
  - Full text available on our website with permission of June Pierce
    https://www.epstrategies.com/BerniePierce.pdf

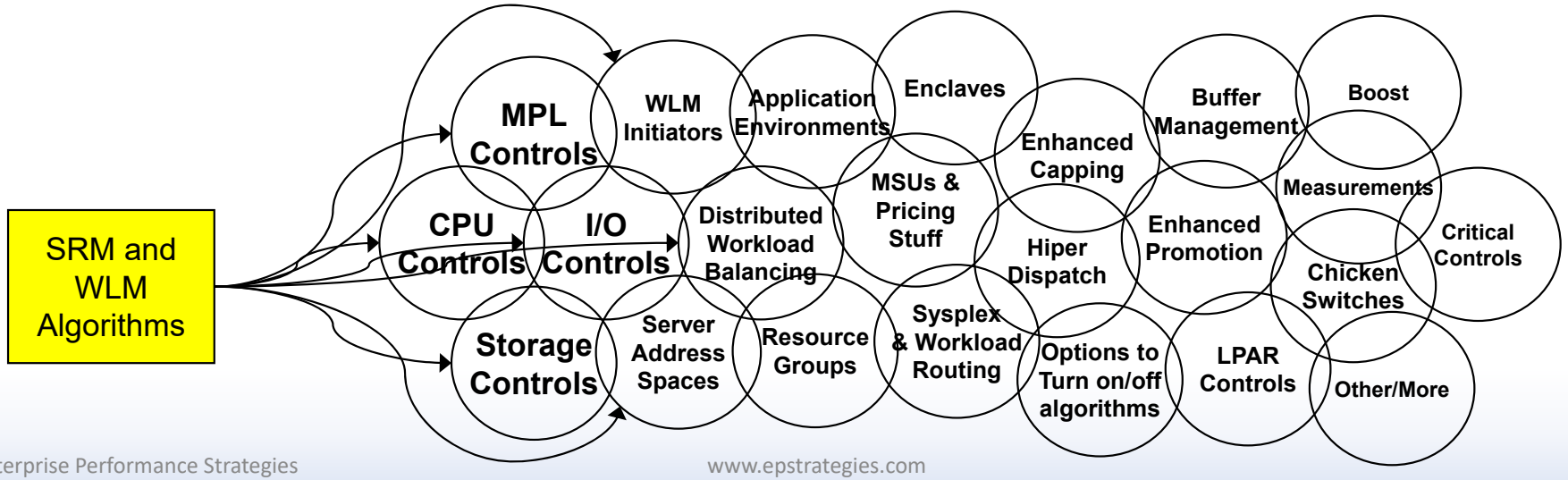
Bernie Pierce

  - *"I have an even greater awareness of the complexity of the challenge accepted to develop the Workload Manager. As I said earlier, I proposed the concept of WLM in 1981 and it was considered too expensive. In retrospect, the management that declined the risk was correct; my estimate was about one order of magnitude less than the eventual implementation. The tools and the processes we employed for software development in 1981 were rudimentary compared to those used to develop WLM. That's two big strikes against my idea. We needed to develop the tools, the processes and the people to build function like WLM or datasharing. We needed to grow as an industry."*

What has been some of the great WLM successes?

The WLM of today

- The original objective was fewer and simpler externals, and WLM delivered a basic set of algorithms for management of system resources such as CPU, memory, and some logical resources
  - The MVS 5.1 platform introduced an autonomic infrastructure
  - First release was then followed by basic management of I/O resources, enclaves, and scalable servers
  - Once this infrastructure was solidified, then WLM could do things such as WLM inits
  - Eventually carried over to PR/SM management and enhanced capping



Thoughts on success

- John Arwe (past WLM designer/ developer / really smart guy):
  - The success is that it really did - drastically - simplify system management of all but the corner cases. Some of those corner cases are/were important, and led to later changes like the "critical" flags I designed in 2000 (once again, so late in the cycle that I had to get Fellow-level approval over the objections of the release manager), but this was a "don't let the perfect be the enemy of the good" situation. Goal mode removed fully 2/3 of existing OPT parameters, the entire ICS, and a lot of historical conflicting *cruft like RTO from the IPS.

    * cruft :badly designed, unnecessarily complicated, or unwanted code or software.

John Arwe

- Scott Chapman (industry z/OS performance expert):
  - Converting to goal mode was the first major thing I did when I joined the systems group. The change went smoothly, so I'd say WLM was successful at making it "easy" to manage diverse workloads without learning and understanding lots of confusing parameters!

Scott Chapman

What were some of the disappointments?

IEAOPTxx still has too many 'chicken switches'

- Parameters in parmlib member IEAOPTxx allow customers to enable, disable, or change a variety of constants that affect the decisions of WLM and SRM
  - Not all algorithms work equally for all customers, so, some constants may need some tweaking

  My dear friend Bob Rogers (who many of you know) likes to call these: **Chicken Switches**

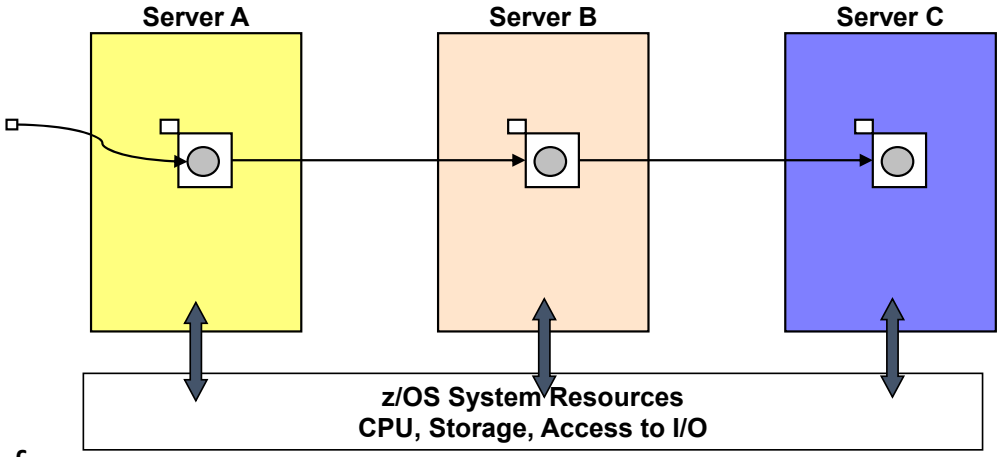
Bob Rogers

- Example: Maybe the following parms should really be policy based (That is… what do you want to happen rather than a 'switch'?)
  - ERV=*xxxxxx*
  - RMPTTOM=*xxxxxx*
  - CPENABLE=(*xxx*,*yyy*)|SYSTEM
  - IIPHONORPRIORITY= YES|NO
  - ZIIPAWMT=*xxxxx*
  - BLWLINTHD, BLWLTRPCT
  - FULLPRESYSTEM=YES|NO
  - HIPERDISPATCH=YES|NO
  - Etc.

Lack of Extensive Subsystem and Vendor Exploitation

- Not many IBM and vendor products took advantage of WLM services to externalize or classify their transactions for WLM management
  - Key IBM WLM exploiters include CICS, DB2, IMS, WAS
    - Even then, CICS and IMS never decided to exploit enclaves
  - A small number of non-IBM products did, but only a few

- Instead, many server address spaces are still 'black box' address spaces
  - That is, we know transactions are running in them, but know nothing about the individual transactions or their performance
  - Instead, address spaces are managed based on their goal and not based on the goals of transactions they are serving
  - Most vendors that do exploit enclaves only did so for the benefit of running on a zIIPs rather than for true WLM transaction management



And there was some WLM back pedaling

- In subsequent release there is usually some amount of back pedaling to undo some algorithms, or to add additional 'chicken switches'

- Here are a few backpedals that quickly come to mind:
  - Originally allowed up to 999 service classes and only 100 report classes. Yikes!
  - PERFORM= parameter in JCL was brought back because it was discovered jobs used it
  - Storage Isolation parameters have since been added to IEAOPTxx
  - SYSSTC1 to SYSSTC5 were introduced and quickly disabled (proud to say I led the charge)
  - Optional enablement server management
  - CPU critical (but changed again in z/OS 3.1)
  - Storage critical
  - Disable zIIP crossover at the service class level
  - Management of locking was never addressed (although usage of promotion was expanded)
  - Discretionary goal management
  - EWLM (probably biggest failure)
  - z/OS 3.1 now defaults usage of the CPU critical control for Imp 1 work

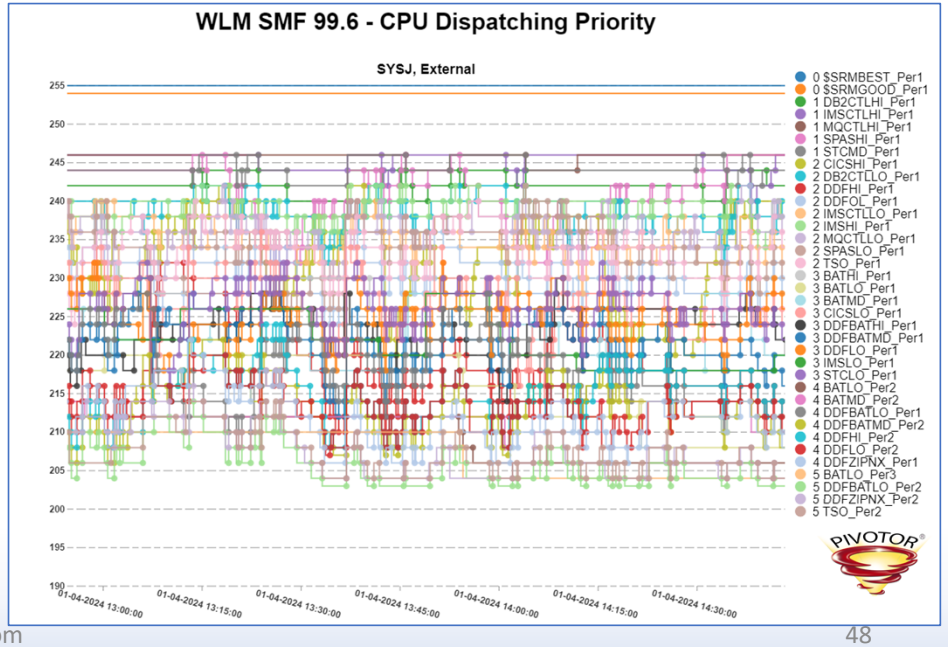
What was the influence of WLM on z/OS performance management?

WLM is still relevant

- WLM allowed professionals to think at a higher level since WLM does the detailed level analysis
  - One does not need to be a rocket scientist to manage z/OS performance
  - Performance analysts can spend time on the bigger and more interesting questions

Think about it…

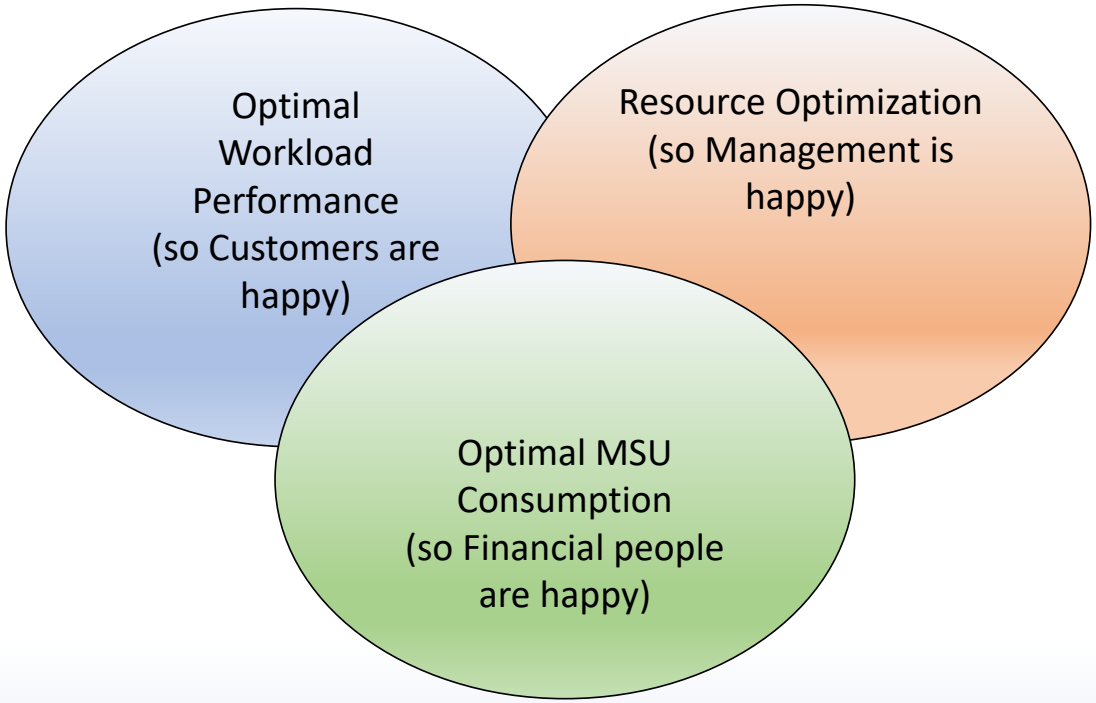
We can have a CPU constraint environment running a variety of ever-changing workloads throughout the day (nighttime batch, daytime online, interactive uses, etc.), and if the policy is setup correctly, WLM will dynamically manage the resources to ensure the most important work receives the resources they need while optimizing the hardware.



What is a little different today…

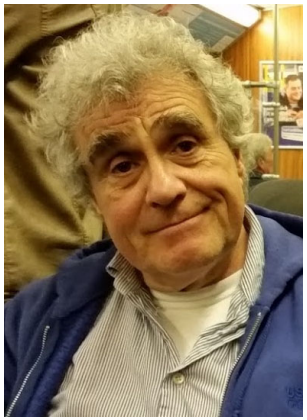
- Today, performance analysts, capacity planners, and a 'workload manager' have now three primary responsibilities

Note: Managing a Rolling 4 Hour Average (R4HA) while managing workloads being capped is only possible with a WLM



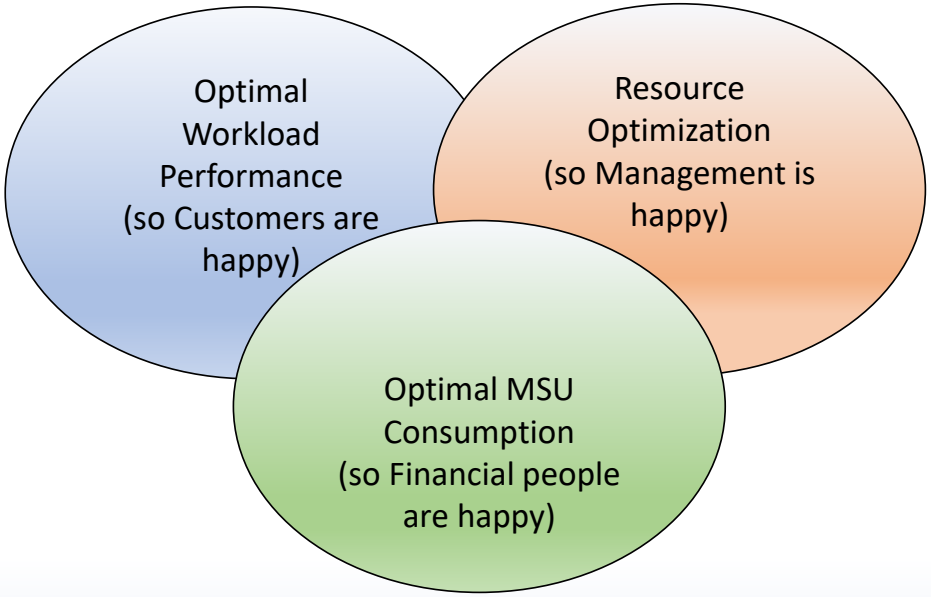
What is the legacy and future of WLM?

The future of WLM



Bob Rogers

Tomorrow's performance problems are created by yesterday's performance solutions!
(Bob Rogers – z/OS Philosopher)



- WLM's future depends on 3 key factors:
  - How customer workloads, applications, and environments evolve
  - How the z platform and resources evolve
  - How IBM and vendor pricing schemas evolve

The legacy of WLM

- WLM has been available now for 30 years
  - It is still working…
  - It is still evolving…
  - And it is still relevant
  - This alone is a great testament to Bernie Pierce, as well as all the other WLM inventors, designers, developers, testers, project managers, writers, and systems analysts
- In my opinion, WLM has been, and will continue be a key part of continued success of the z/OS platform
- Thank-you Bernie!

Bernie Pierce

Thank you!

- There is so much more about those first days of I want to tell you about
  - I wish we had more time!
  - There are so many fun stories to tell.
  - It was a special time in IBM MVS development, and one of the highlights of my career

- I want to thank the Pierce family for their support during my research, and for their allowing me to use Bernie Pierce's image and words

- During this presentation I talked to a wide array of people, but I specifically want to thank the following individuals for their unique and historical insights:
  - Bob Rogers
  - John Arwe
  - Scott Chapman
  - Robert Vaupel