COMP 3400 Mainframe Administration[1]

Christian Grothoff
`christian@grothoff.org`

`http://grothoff.org/christian/`

---

[1]These slides are based in part on materials provided by IBM's Academic Initiative.

UNIVERSITY OF DENVER

Performance

- Speed or Efficiency

- Device Utilization

- Response Time

- System Capacity / Number of Users Supported

- Throughput

- Reliability / Consistency

- Number of Complaints

System Resources Affecting Performance

- Processor

- Memory Hierarchy

- I/O (Disks and Network)

- Attributes:
  - Capacity
  - Utilization
  - Distribution

Attributes that can be Measured

- Processor: Capacity, Usage

- Time: CP system time, virtual machine and virtual processor time

- Memory: Central, Expanded, Paging

- I/O: Bandwidth (Capacity, Usage), Latency

UNIVERSITY OF
DENVER

Performance Tuning

1. Measure

2. Reduce data

3. Analyze data

4. Tune

5. Measure again!

Measuring

The following CP commands help with measuring:

- CP INDICATE
- CP QUERY
- CP MONITOR

CP INDICATE

INDICATE provides us with a snapshot of system activities:

- INDICATE USER: system resource statistics about our own VM

- INDICATE LOAD: current contention for system resources

- INDICATE QUEUES: scheduling information (more on this later)

- INDICATE I/O: disk performance

- INDICATE SPACES: space utilization by address spaces

- INDICATE PAGING: usage of auxiliary storage

CP QUERY

QUERY also provides snapshot information on certain system statistics:

- QUERY FRAMES: storage use (real storage)

- QUERY SXSPAGES: information on pages in execution space (below 2 GB)

CP MONITOR

Collect system performance data in a saved segment for processing by an application program later. To use it, you must:

1. Create a monitor saved segment (of sufficient size)

2. Create a VM with application program that loads the saved segment and connects to the *MONITOR system service to retrieve and process the data

3. Establish a monitor profile (what to monitor)

4. Enter the MONITOR START command

Data Types

MONITOR can collect two types of data:

- Event data — record each time an event occurs

- Sample data — record system status, either once or at a particular frequency

Example Configuration

```
monitor sample enable processor
monitor sample enable storage
monitor enable user userid jordan perkins black
monitor enable i/o device 0120 5140 150
monitor enable i/o type 3380
monitor enable i/o volume pack1 pack2
monitor sample rate 1 second
monitor sample interval 2 minutes
monitor config size 40
monitor sample start
```
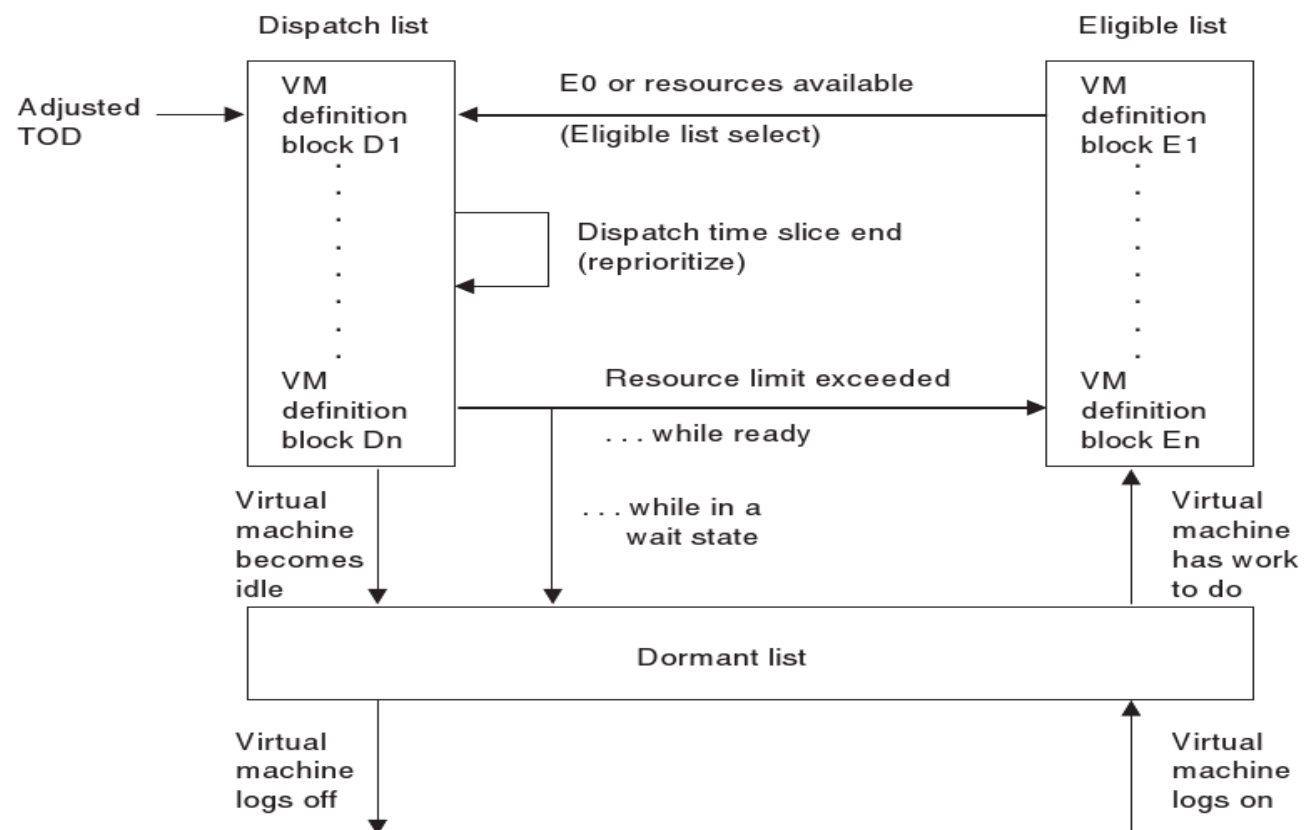
Analyzing Performance Data

This requires understanding:

• the performance problem (always)

• the performance goals (always)

• the application logic (always)

• the operating system logic (sometimes)

• the hardware architecture (sometimes)

$\Rightarrow$ We will focus on z/VM system logic

z/VM Scheduling

User Classes

E0 Guests that do not wait in the eligible list regardless of transaction duration (quick dispatch)

E1 Guests that have just begun a transaction; assumed to be short transactions

E2 Medium-length transactions; guest did not complete during first dispatch

E3 Long running transactions; guest did not complete during second dispatch stay

UNIVERSITY OF
DENVER

Sharing between Guests: NSS and DCSS

- Name Saved Systems (NSS) can be used if multiple guests have the same function; the associated code and data can then be shared

- NSS is often used for operating system kernels (CMS, Linux)

- Discontiguous Shared Segments (DCSS) are similar to NSS in that they can be accessed by multiple guests as part of their virtual storage

- DCSS can not be IPLed

Tools for Analyzing Performance

- z/VM Performance Monitor: `PERFSVM`

- Tivoli Omegamon

Tuning

Tuning may require changes to:

- the application software

- the operating system configuration

- the z/VM configuration

- the hardware

$\Rightarrow$ We will focus on changes to z/VM configuration

CP SET

CP SET can be used to change performance characteristics:

- SET MAXUSERS — limit number of users who can logon

- SET CACHE — enable or disable caching

- SET SHARE — change resource-access priorities

- SET RESERVED — change number of available real storage frames for a VM

- SET THROTTLE — limit I/O operations for a guest

Reactive Analysis

Reactive performance analysis is in response to a problem. Questions to ask include:

- Who is having the problem? A single user, a group of users or all users?

- What type of problem is it? (Data access, response time, logon, abend)

- What is special about those users (what do they run, what did they do, did they do this previously, shared resources)?

- Did anything change in the system recently?

UNIVERSITY OF
DENVER

Example: User using too much CPU

- INDICATE QUEUES EXP

$\Rightarrow$ Display members on dispatch and eligible lists

- INDICATE USER $USERNAME$

$\Rightarrow$ Display details on user activity (CPU usage, I/O usage)

Possible solutions: reboot or terminate guest.

Example: System hang

- Symptom: System hangs for a few minutes, then recovers; users have to log back on

- Cause: CP abends, produces a dump and reIPLs

- Action: research cause of abend in manual; for example, PGT004 indicates lack of paging space on disk

Predictive Analysis

Predictive analysis is monitoring performance data to avoid problems before they occur:

- Graph data on system utilization to recognize trends

- Simulate load to determine system limitations

Measures: CPU usage too high

- Ask developers to improve application performance

- Persuade users to stagger workload

- Reduce number of virtual processors for heavy CPU users

- Use `SET SHARE nn% LIMITHARD` to cap heavy users

- Use `SET SRM DISPBUF` to limit users in dispatch list

Measures: Not enough Storage

- Ask developers to reduce storage requirements

- Reduce the size of virtual machines

- Use `SET SRM STORBUF` to reserve storage for interactive users

- Shortage of storage usually shows in problems with paging!

Measures: Too much paging

- Use `QUERY ALLOC PAGE` to check usage

- Ask developers to improve data locality

- Add more paging space

- Use `SET LDUBUF` to give interactive users better paging resource availability

UNIVERSITY OF
DENVER

Measures: Too much I/O

- Check data on devices to see if it can be spread across more paths (if some devices are more busy than others)

- Use `THROTTLE` to limit heavy users

- Ask developers to improve data locality (if seek time is the issue)

Questions

?