What is Work Management?
More User

Single User Environment

Application

OS

Multi User Environment

Application

OS

... Performance? Security? Availability?
More Applications

- App 1
- App 2
- ... (App N)
- OS 1
- OS 2
- OS N

Reality is more complex!


Windows Deployment Example: Multiple Server Tiers
Work Management

Running Multiple Applications in a single OS Instance

App 1   App2   ...    App N

OS

Performance
(or how to share resources)

Security  Lecture 4

Availability

Do not confuse work management and LPAR!
Process Scheduling
Introduction to Scheduling

- Bursts of CPU usage alternate with periods of I/O wait
  - a CPU-bound process
  - an I/O bound process

Source: Andrew S. Tanenbaum: Modern Operating Systems 2th Edition
Scheduling Algorithm Goals

All systems
Fairness - giving each process a fair share of the CPU
Policy enforcement - seeing that stated policy is carried out
Balance - keeping all parts of the system busy

Batch systems
Throughput - maximize jobs per hour
Turnaround time - minimize time between submission and termination
CPU utilization - keep the CPU busy all the time

Interactive systems
Response time - respond to requests quickly
Proportionality - meet users’ expectations

Real-time systems
Meeting deadlines - avoid losing data
Predictability - avoid quality degradation in multimedia systems

Source: Andrew S. Tanenbaum: Modern Operating Systems 2th Edition
As a process executes, it changes **state**

- **new**: The process is being created
- **running**: Instructions are being executed
- **waiting**: The process is waiting for some event to occur
- **ready**: The process is waiting to be assigned to a processor
- **terminated**: The process has finished execution

Source: Silberschatz, Galvin and Gagne: Operating System Concepts - 7th Edition
Preemptive and Non-preemptive Scheduling

• CPU scheduling decisions may take place, if a process
  (1) switches from running to waiting state
  (2) switches from running to ready state
  (3) switches from waiting to ready state
  (4) terminates

• When scheduling take place only under (1) or (4) the scheduling scheme is non-preemptive or preemptive.

• Examples:
  ▪ non-preemptive: Windows 3.1
  ▪ preemptive: Windows 95 or later, Mac OS X

Source: Andrew S. Tanenbaum: Modern Operating Systems 2th Edition
Context Switch

• When CPU switches to another process, the system must save the state of the old process and load the saved state for the new process
• Context-switch time is overhead; the system does no useful work while switching
• Time dependent on hardware support

Source: Silberschatz, Galvin and Gagne: Operating System Concepts - 7th Edition
Process Control Block (PCB)

- Information associated with each process:
  - Process state
  - Program counter
  - CPU registers
  - CPU scheduling information
  - Memory-management information
  - Accounting information
  - I/O status information

Source: Silberschatz, Galvin and Gagne: Operating System Concepts - 7th Edition
Single Process Execution

- CPU–I/O Burst Cycle – Process execution consists of a cycle of CPU execution and I/O wait

- CPU burst distribution

Source: Silberschatz, Galvin and Gagne: Operating System Concepts - 7th Edition
CPU Switch From Process to Process

Efficient process scheduling?

Source: Silberschatz, Galvin and Gagne: Operating System Concepts - 7th Edition
Scheduling Criteria

- **CPU utilization** – keep the CPU as busy as possible
- **Throughput** – # of processes that complete their execution per time unit
- **Turnaround time** – amount of time to execute a particular process
- **Waiting time** – amount of time a process has been waiting in the ready queue
- **Response time** – amount of time it takes from when a request was submitted until the first response is produced, **not** output (for time-sharing environment)

Source: Silberschatz, Galvin and Gagne: Operating System Concepts - 7th Edition
Scheduling in Interactive Systems

A scheduling algorithm with four priority classes

Source: Andrew S. Tanenbaum: Modern Operating Systems 2th Edition
Scheduling in Batch Systems

Three level scheduling

Source: Andrew S. Tanenbaum: Modern Operating Systems 2th Edition
Example: IBM i

aka i5/OS, OS/400
5.1 Operator’s View
Job

Qualified job name

000578  AS24xx  DSP01

Job number  User name  Job name
Job Types

- **System jobs**
  - Spooling

- **User jobs**
  - Autostart
  - Interactive
  - Batch

- Communication
The Life Cycle of a Job

Submit → Job queue → Subsystem → Memory pool → Output
Work Management Objects

- OS/400
- SUBSYSTEM
- JOB THREAD
- SYSTEM VALUES
- NETWORK ATTRIBUTES
- SBSD
- WORK ENTRIES
- ROUTING ENTRIES
- POOLS
- JOBD
- CLASS
- PROGRAM
## System Values

<table>
<thead>
<tr>
<th>Name in iSeries Navigator</th>
<th>Description of system value</th>
<th>Name in character-based interface</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Dynamically adjust job priorities of interactive jobs</strong></td>
<td>Sets the job priorities of interactive jobs</td>
<td>QDYNPTYADJ</td>
</tr>
<tr>
<td><strong>Dynamically adjust job priorities within priority bands</strong></td>
<td>Sets job priorities within priority bands to be dynamically adjusted.</td>
<td>QDYNPTYS CD</td>
</tr>
<tr>
<td><strong>Automatically adjust memory pools and activity levels</strong></td>
<td>Specifies when to automatically adjust memory pools and activity levels.</td>
<td>QPFRADJ</td>
</tr>
<tr>
<td><strong>Maximum eligible threads</strong></td>
<td>Specifies the maximum number of eligible threads.</td>
<td>QMAXACTLVL</td>
</tr>
<tr>
<td><strong>Machine memory pool size</strong></td>
<td>Specifies the size of the machine memory pool.</td>
<td>QMCHPOOL</td>
</tr>
<tr>
<td><strong>Base memory pool minimum size</strong></td>
<td>Specifies the minimum base memory pool.</td>
<td>QBASPOOL</td>
</tr>
<tr>
<td><strong>Base memory pool maximum eligible threads</strong></td>
<td>Specifies maximum number of eligible threads.</td>
<td>QBASACTLVL</td>
</tr>
<tr>
<td><strong>Move interactive jobs to base pool at end of time slice</strong></td>
<td>Specifies whether to move interactive jobs to base pool at the end of the time slice or not.</td>
<td>QTSEPOOL</td>
</tr>
</tbody>
</table>
Network Attribute

- **Current System Name**: ABC
- **Local Network ID**: APPN
- **APPN Node type**: *ENDNODE
- **Alert Status**: *OFF
- **Alert Primary Focal Point**: *NO
- **Message Queue/Obj. Dist.**
  - Output Queue/Obj. Dist.
  - QSYS/QSYSOPR
  - QGPL/QPRINT
- **Action for DDM Request**: *OBJAUT
- **PC Support Request Access**: *OBJAUT
Job Queue

Job queue:
- JOB1 (2)
- JOB2 (1)
- JOB3 (4)
- JOB4 (2)
- JOB5 (1)
- JOB6 (4)

Job queue priority:

1. JOB2
   - JOB5
2. JOB1
   - JOB4
3. JOB3
   - JOB6
4. 

...
Job Description

Display Job Description

<table>
<thead>
<tr>
<th>Job description:</th>
<th>Library:</th>
<th>System: HYDRA</th>
</tr>
</thead>
<tbody>
<tr>
<td>QDFTJOBD</td>
<td>QGPL</td>
<td></td>
</tr>
</tbody>
</table>

- **User profile**: *RQD
- **CL syntax check**: *NOCHK
- **Hold on job queue**: *NO
- **End severity**: 30
- **Job date**: *SYSVAL
- **Inquiry message reply**: *RQD
- **Job priority (on job queue)**: 5
- **Job queue**: QBATCH
- **Library**: QGPL
- **Output priority (on output queue)**: 5
- **Printer device**: *USRPRF
- **Output queue**: *USRPRF

More...

Press Enter to continue.

F3=Exit  F12=Cancel

Message logging:
- **Level**: 4
- **Severity**: 0
- **Text**: *NOLIST
User Profile

Display User Profile - Basic

User profile . . . . . . . . . . . . . . : AS2401

Storage information:
  Maximum storage allowed . . . . . . . . : *NOMAX
  Storage used . . . . . . . . . . . . . . : 312
  Storage used on independent ASP . . . . : *NO
  Highest scheduling priority . . . . . . . : 3
  Job description . . . . . . . . . . . . . : QDFTJOBD
  Library . . . . . . . . . . . . . . . . : QGPL
  Accounting code . . . . . . . . . . . . . :
  Message queue . . . . . . . . . . . . . : AS2401
    Library . . . . . . . . . . . . . . . : QUSR SYS
  Message queue delivery . . . . . . . . : *NOTIFY
  Message queue severity . . . . . . . . : 00
  Output queue . . . . . . . . . . . . . : AS24OUTQ
    Library . . . . . . . . . . . . . . . : AS2401
  Printer device . . . . . . . . . . . . . :
    *WRKSTN

Press Enter to continue.

F3=Exit   F12=Cancel
Job Processing

SBMJOB

Job queue

Subsystem pool

Batch

Interactive

Output queue

WRITER

PRINTER

WRKJOBQ

WRKACTJOB

WRKOUTQ

WRKWTR

WRKUSRJOB

WRKSBMJOB

WRKSPLF
Subsystems

A specialized environment for handling a certain type of work or function, such as:

- Interactive
- Batch
- Spooling
- Communications
Storage Pools

- Logical Areas of Main Storage for Jobs:
  - POOL SIZE sets amounts of storage
  - Noncontiguous pages
  - Deallocated when set to *NOSTG (No storage)
  - PAGING is restricted to pool
  - ACTIVITY LEVEL determines how many job threads may have main storage
  - Two types of pools: SHARED AND PRIVATE

<table>
<thead>
<tr>
<th>Pool Type</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>*SPOOL POOL</td>
<td>4</td>
</tr>
<tr>
<td>*INTERACT POOL</td>
<td>3</td>
</tr>
<tr>
<td>*BASE POOL</td>
<td>2</td>
</tr>
<tr>
<td>*MACHINE POOL</td>
<td>1</td>
</tr>
</tbody>
</table>
Subsystem Description (1/4)

Display Subsystem Description

Subsystem description: QINTER
Status: ACTIVE

System: ABC
Library: QSYS

Select one of the following:
1. Operational attributes
2. Pool definitions
3. Autostart job entries
4. Work station name entries
5. Work station type entries
6. Job queue entries
7. Routing entries
8. Communications entries
9. Remote location name entries
10. Prestart job entries

30. All of the above

Selection or command =>

Operational Attributes

<table>
<thead>
<tr>
<th>Maximum jobs in subsystem</th>
<th>*NOMAX</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sign-on display file</td>
<td>QSYS/QDSIGNON</td>
</tr>
</tbody>
</table>

Pool Definitions

<table>
<thead>
<tr>
<th>Pool ID</th>
<th>Storage Size (KB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>*BASE</td>
</tr>
<tr>
<td>2</td>
<td>*INTERACT</td>
</tr>
</tbody>
</table>
Subsystem Description (2/4)

Display Subsystem Description
Subsystem description: QINTER System: ABC
Status: ACTIVE Library: QSYS

Select one of the following:
1. Operational attributes
2. Pool definitions
3. Autostart job entries
4. Work station name entries
5. Work station type entries
6. Job queue entries
7. Routing entries
8. Communications entries
9. Remote location name entries
10. Prestart job entries
30. All of the above

Selection or command
=>

WORKSTATION Entries
<table>
<thead>
<tr>
<th>Type</th>
<th>Jobd</th>
</tr>
</thead>
<tbody>
<tr>
<td>*ALL</td>
<td>*USRPRF</td>
</tr>
<tr>
<td>*USRPRF</td>
<td></td>
</tr>
</tbody>
</table>

Job Queue Entries
<table>
<thead>
<tr>
<th>Jobq</th>
</tr>
</thead>
<tbody>
<tr>
<td>QBATCH</td>
</tr>
</tbody>
</table>

PRESTART Job Entries
<table>
<thead>
<tr>
<th>Prog</th>
<th>Jobd</th>
</tr>
</thead>
<tbody>
<tr>
<td>PROG01</td>
<td>JOBDAPPC</td>
</tr>
</tbody>
</table>

AUTOSTART Job Entries
<table>
<thead>
<tr>
<th>Job</th>
<th>Jobd</th>
</tr>
</thead>
<tbody>
<tr>
<td>START</td>
<td>STARTJD</td>
</tr>
</tbody>
</table>

Communication Entries
<table>
<thead>
<tr>
<th>Dev</th>
<th>Jobd</th>
</tr>
</thead>
<tbody>
<tr>
<td>*ALL</td>
<td>*USRPRF</td>
</tr>
</tbody>
</table>
Subsystem Description (3/4)

Display Subsystem Description

Subsystem description: QINTER
Status: ACTIVE
System: ABC
Library: QSYS

Select one of the following:
1. Operational attributes
2. Pool definitions
3. Autostart job entries
4. Work station name entries
5. Work station type entries
6. Job queue entries
7. Routing entries
8. Communications entries
9. Remote location name entries
10. Prestart job entries

30. All of the above

Selection or command
=>

Routing Entries

<table>
<thead>
<tr>
<th>SEQNBR</th>
<th>CMPVAL</th>
<th>CLASS</th>
<th>PGM</th>
<th>POOL ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>QCMDB</td>
<td>QBATCH</td>
<td>QCMD</td>
<td>1</td>
</tr>
<tr>
<td>50</td>
<td>QCMDI</td>
<td>QINTER</td>
<td>QCMD</td>
<td>2</td>
</tr>
<tr>
<td>9999</td>
<td>*ANY</td>
<td>QBATCH</td>
<td>QCMD</td>
<td>1</td>
</tr>
</tbody>
</table>

Class Information

<table>
<thead>
<tr>
<th>QBATCH</th>
<th>QINTER</th>
</tr>
</thead>
<tbody>
<tr>
<td>Run Priority</td>
<td>50</td>
</tr>
<tr>
<td>Time Slice</td>
<td>5000 ms</td>
</tr>
<tr>
<td>Purge</td>
<td>*NO</td>
</tr>
<tr>
<td>Wait Time</td>
<td>120 sec</td>
</tr>
</tbody>
</table>
Subsystem Description (4/4)

Display Subsystem Description
Subsystem description: QINTER System: ABC
Status: ACTIVE Library: QSYS

Select one of the following:
1. Operational attributes
2. Pool definitions
3. Autostart job entries
4. Work station name entries
5. Work station type entries
6. Job queue entries
7. Routing entries
8. Communications entries
9. Remote location name entries
10. Prestart job entries

30. All of the above

Selection or command
==>________________________________________

Job queue entries in subsystem description

<table>
<thead>
<tr>
<th>JOBQ</th>
<th>JOBQA</th>
<th>JOBQB</th>
<th>JOBQC</th>
</tr>
</thead>
<tbody>
<tr>
<td>SEQNBR</td>
<td>10</td>
<td>20</td>
<td>30</td>
</tr>
<tr>
<td>MAXACT</td>
<td>*NOMAX</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>MAXPTY1</td>
<td>*NOMAX</td>
<td>*NOMAX</td>
<td>*NOMAX</td>
</tr>
<tr>
<td>MAXPTY9</td>
<td>:</td>
<td>:</td>
<td>:</td>
</tr>
</tbody>
</table>

Jobs 1 to 5 have equal priority.

Which jobs run concurrently if the subsystem description has:
- MAXJOBS = 2
- MAXJOBS = 3
- MAXJOBS = 4
- MAXJOBS = *NOMAX

?
Batch Job Routing

SBMJOB CMD(CALL PRPROC) RTGDTA(QCMDB) JOB(PAY) JOBQ(QBATCH)

Routing Entry

SEQNBR  CMPVAL  PGM  CLASS  POOL
9999    *ANY    QCMD  QBATCH  1

Max Jobs - *NOMAX
Job Queue Entry
Job Queue: QBATCH
Max Act: 1
Max Pty(n): *NOMAX

Run Priority
CLASS: QGPL/QBATCH

Call PRPROC
JOB MSGQ
PAY

QSYS/QCMD
*BASE

PRPROC

PAY

JOBQ
QBATCH

SUBSYSTEM
MONITOR
PROGRAM

QBATCH
Subsystem Description

Max Jobs - *NOMAX
Job Queue Entry
Job Queue: QBATCH
Max Act: 1
Max Pty(n): *NOMAX
Interactive Job Routing

QINTER
Subsystem Description

Attributes
Storage Pools
MAXJOBS(*NOMAX)

<table>
<thead>
<tr>
<th>ID</th>
<th>Storage</th>
<th>Sign-on Display File (QSYS/QDSIGNON)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>*BASE</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>*INTERACT</td>
<td></td>
</tr>
</tbody>
</table>

Work Station Entry
Job Description
*USRPRF

Routing Entry
SEQNBR  CMPVAL  PGM  Class  Pool
50      QCMDI  QCMD  QINTER  2

User Profile = XYZ
PASSWORD = XXXXXX
INLPGM = USERLIB/USERPGM
JOBD = USERLIB/USERJOBD

User Name
Job Queue
Job Priority
Output Priority
Routing Data (QCMDI)
Request Data
Initial Library List
More . . . .

CLASS
QGPL/QINTER

Run Priority

QSYS/QCMD
USERPGM

*INTERACT
5.2 System’s View

Job Structure and Task Dispatching
Three-level Hierachy of Work

- **Process**
  - MI System object called process-control space (PCS)
  - Tie together the resources for program invocation
  - Consists of
    - A executable program (can be shared)
    - Program’s data
    - State informations
  - Could also be defined as a program in execution

- **Job**
  - OS/400 object that contains a process structure to manage system resources

- **Thread**
  - A portion of a process
  - A process may have more than one thread
  - Each thread consists of
    - Set of registers representing the state of the processor
    - A control stack
    - Automatic program storage (local variables)
  - System resources allocated to the process are shared with other threads within the process
Permanent Job Structure Objects

WCBT
- WCBTE
- JOBMSGQ
- SCB
- JOBQ
- LDA

WCBTE
- JOBMSGQ
- SCB
- SBSID
- LDA
- PCS

JOB on JOBQ
JOB active in Subsystem

SCB
Job Message Queue
Spool Control Block

LDA
Local Data Area

PCS
Process Control Space

Job Message Queue
Spool Control Block
Local Data Area

WCBT – Work Control Block Table
WCBTE – Work Control block Table Entry

Module 5
© Copyright IBM Corporation 2008
Temporary Job Structure Objects

WCBT

SCB

JOBMSGQ

LDA

QTEMP

Process Access Group

Control

Field

File

Process Control Space

Static Storage

A

B

Thread Control Space
(for initial thread)

O

D

P

Thread Control Space
(for secondary thread)

O

D

P

Thread Control Space
(for secondary thread)

TDE

ODP – Open Data Path
TDE – Task Dispatching Element
Open Data Path (ODP)

Job X

Open A

UFCB – User Field Control Block

Create
Duplicate
Open Data Path Sharing

SHARE(*NO)

Job X

Pgm #1
Open file A
Call Pgm #2

Job Y

SHARE(*YES)

Pgm #2
Open file A
Return

Module 5
© Copyright IBM Corporation 2008
Thread-control Space (TCS)

- are temporary SLIC objects
- will be recycled at process level or at systems level when process ends

---

Source: Frank G. Soltis, Fortress Rochester, Fig. 13.5
Task Dispatching Queues

TDQ description  Pointer

TDE description  Pointer  Priority

TDE description  Pointer  Priority

TDE description  Pointer  Priority

TDE description  Pointer  Priority

Highest  Priority  Lowest
Send/Receive Queues

- Data structure used as “mailbox” for tasks
- SEND MESSAGE enqueued a SRM
- Execute a RECEIVE MESSAGE operations to obtain a message (dequeuing)
  - If no message is available the task has to wait

SRQ – Send/Receive Queue
SRM – Send/Receive Message
RECM – Receive Message
SENDM – Send Message

SRQ
  --->
  SRM
    --->
    KEY1
    --->
    SRM CHAIN
    --->
    MSG
    --->
    TDE CHAIN
    --->
    TDE

SRM
  --->
  KEY2
  --->
  SRM CHAIN
  --->
  MSG

PRIORITY
TDE CHAIN
STATUS
Key Comparison
If RECM Is Unsatisfied

Instruction Stream

TDE CHAIN

PRIORITY TDE CHAIN STATUS

TDQ

TDE

PRIORITY TDE CHAIN STATUS

SRQ

SRM

KEY1 SRM CHAIN TDE CHAIN

SRM

KEY2 SRM CHAIN MSG

TDE

INSTR

PRIORITY TDE CHAIN STATUS

TDE

PRIORITY TDE CHAIN STATUS

TDE

PRIORITY TDE CHAIN STATUS

TDE
SENDM (1/2)
Task States

- **Active**: TDE CHAIN
  - PRIORITY
  - TDE CHAIN
  - STATUS

- **Ready**: TDE CHAIN
  - PRIORITY
  - TDE CHAIN
  - STATUS

- **Wait**: SRM CHAIN
  - KEY1
  - SRM CHAIN
  - MSG

- **Wait**: SRM CHAIN
  - KEY2
  - SRM CHAIN
  - MSG

- **Wait**: TDE CHAIN
  - PRIORITY
  - TDE CHAIN
  - STATUS

- **Wait**: TDE CHAIN
  - PRIORITY
  - TDE CHAIN
  - STATUS
States of Tasks

1. Initiate task (start work)
2. Run task (dispatch task)
3. Suspend task (work done)

1. Preempt task (by another task)
2. Wait (for I/O or another task)
3. Signal (task is done waiting)
Thread States and Transitions

Long Wait State

Act Level Open

W-I

YES

W-A

NO

I-A

Ineligible State

Active State Activity Levels

A-W

Prime Task Dispatching Queue

Send/Receive Queue

Other Jobs Ineligible

TSE

A-A

A-I

1 2 3

NO
### Observing Thread State Transitions

#### Work with System Status

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th>Active --&gt;Wait --&gt; Active --&gt;</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>System</td>
<td>Pool</td>
<td>Reserved</td>
<td>Max</td>
<td>Active</td>
<td>Wait</td>
<td>Inel</td>
</tr>
<tr>
<td>Pool</td>
<td>Size (M)</td>
<td>Size (M)</td>
<td>Active</td>
<td>Inel</td>
<td>Inel</td>
<td>Inel</td>
</tr>
<tr>
<td>1</td>
<td>3.61</td>
<td>2.26</td>
<td>+++</td>
<td>7.5</td>
<td>.0</td>
<td>.0</td>
</tr>
<tr>
<td>2</td>
<td>3.97</td>
<td>0</td>
<td>10</td>
<td>.4</td>
<td>.0</td>
<td>.0</td>
</tr>
<tr>
<td>3</td>
<td>.23</td>
<td>0</td>
<td>3</td>
<td>.0</td>
<td>.0</td>
<td>.0</td>
</tr>
<tr>
<td>4</td>
<td>8.19</td>
<td>0</td>
<td>25</td>
<td>88.4</td>
<td>.0</td>
<td>.0</td>
</tr>
</tbody>
</table>

#### Auxiliary storage:

- CPU used: 61.7%
- DB capability: 00:06:30
- Jobs in system: 494
- % perm addresses: 3258 M
- % temp addresses:
  - Permanent: 3.063 M
  - Temporary: 3.962 M
- % system ASP: 18.6657
- System ASP: 3258 M
- % DB capability: 00:06:30
- Elapsed time: 00:06:30
- Current unprotect used: 437 M
- Maximum unprotect: 437 M

#### Type changes (if allowed), press Enter.

**Bottom**

- 4=Prompt
- F5=Refresh
- F9=Retrieve
- F10=Restart
- F11=Display pool data
- F12=Cancel
- F14= Work with subsystems
- F24=More Keys
Active-to-Wait

Prime TDQ

Long Wait State

Activity Levels

Active State

PAG
Wait-to-Ineligible

Long Wait State

Activity Levels

1. JOBD
2. JOBB
3. JOBC

Active State

Ineligible State

Act Level Open?

NO

YES

W-I
Wait in Activity Level

Long Wait State

Activity Levels

Active State

Prime TDQ

Send/Receive Queue

JOBA

JOBB

JOBC
Dynamic Priority Scheduling

New algorithms on RISC systems

- No job can monopolize the CPU
  - Jobs not using CPU have their DELAY COST increased
- Low-priority jobs have a chance to progress
- iSeries server optimization for batch
- Interactive job priority lowered when batch demand is high

Can be turned on/off

- QDYNPTYSCD and QDYNPTYADJ
Delay Cost Curves

Delay Cost

Time (Millions of cycles)

Priorities 10-16

Priority 0-9

Priorities 17-22

Priorities 23-35

Priorities 36-46

Priorities 47-51

Priorities 52-89

Priorities 90-99
Multi-Processor Considerations (1/2)

Dispatching the first n top TDEs on the TDQ?

STOP

Cache Affinity and Performance?!
Multi-Processor Considerations (2/2)

Task Dispatching is based on priority, cache affinity and eligibility

Three bits in a TDE

- Eligibility bit
  - one bit for every processor
  - indicates that a task is eligible to run on on corresponding processor
- Active field
  - one bit for every processor
  - indicates on which processor the task is currently running
- Affinity field
  - one bit for every processor
  - Indicates on which processor the task was currently running