IBM i Performance Tools for Application Developers

Dawn May – dmmay@us.ibm.com
@DawnMayiCan

for Business

IBM

Performance Disclaimer

- "it depends …"

- Performance information and recommendations in this presentation are based on measurements, analysis, and projections in a controlled environment for specific performance workloads.

- Your results may vary significantly and are dependent on the application and configuration.

- This information is provided along with general recommendations for you to better understand system performance.

- Information is provided *AS IS* without warranty of any kind.

Definitions

- **Interactive work** – Generally 5250 online transaction processing (OLTP)

- **Batch work** – non-interactive workloads

- **Commercial Processing Workload** (CPW) – Workloads which have a relatively large amount of I/O compared to computation

- **Disk arms** – generally one disk arm per disk drive. More disk arms generally results in improved I/O performance

- **Disk capacity** – the amount of disk storage space

- **Memory paging and faulting** – the movement of data in and out of memory

- **Wait Accounting** – the ability to determine what a job is doing when it is not running

Definitions...

- **Measurement** - The collection of performance metrics

- **Transaction** - A basic unit of work

- **Workload** - An application that can drive load on a system

- **Benchmark** - A specific workload with specific environment settings

- **Metric** - a value that is measured to gain insight into performance

- **Response Time** - The average observed time to complete a transaction

- **Utilization** - The percent of time that a resource is busy

- **Throughput** - The rate at which transactions are completed

- **Capacity** - The maximum throughput of a system

Agenda

- Brief Overview

- Introduction to IBM i Wait Accounting

- Performance Data Collectors
    - Collection Services
    - Job Watcher
    - Performance Explorer

- Performance Data Visualization and Diagnostics
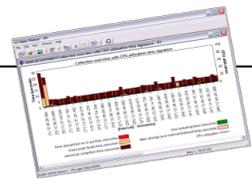   - Performance Data Investigator
    - iDoctor

- Examples

Keep Current on PTFs

It's always good practice to keep current on the latest fixes from IBM

- PTFs address defects
- PTFs introduce new capabilities

  - IBM i Technology Refresh Updates

  - IBM i Group PTFs
    - Database
    - Performance tools
    - Java
    - HTTP Server
      - HTTP Server Group PTF for latest Navigator for i functionality

  - PTFs for performance data collectors
    - Collection Services, Job Watcher, Disk Watcher, Performance Explorer

IBM i Performance Tools

iDoctor Suite
- Job Watcher
- PEX Analyzer
- Heap Analyzer
- VIOS Investigator

Green Screen
- Performance Tools Reports
- System commands
- Performance Explorer

PM for Power Systems

System i Navigator
- Perf Tools GUI
- DB/SQL Monitor
- Active Jobs
- System Status
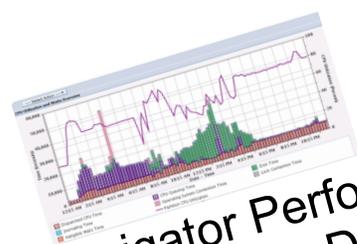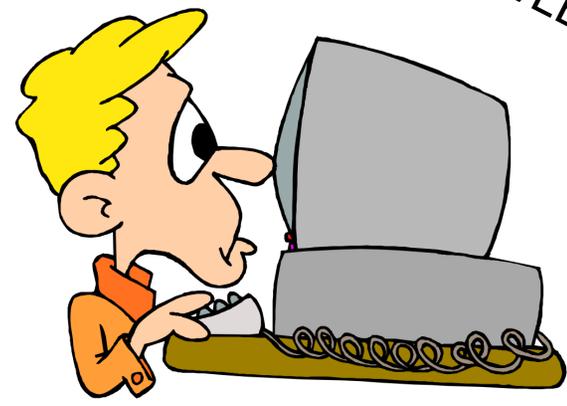- Graph History

PTDV

Collection Services

Job Watcher

Disk Watcher

WLE

Management Central

Third Party Tools

Performance Explorer

Navigator Performance
- Investigate Data (PDI)
- Batch Model
- System Monitors

Web Performance Advisor
Web Performance Monitor

Navigator for i

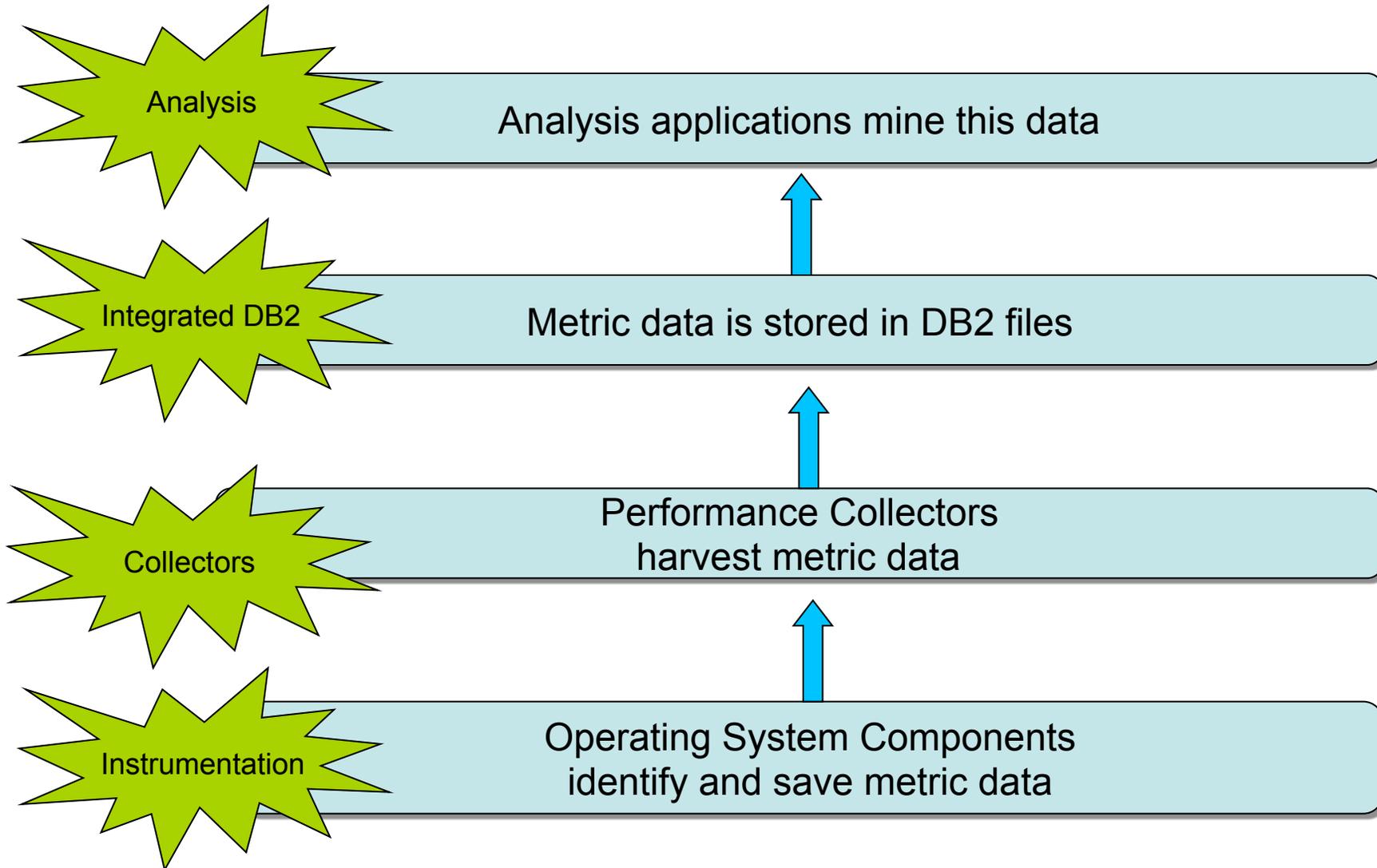# Performance Instrumentation and Data Collection

The ⓘ for Business Advantage

- IBM **develops the software stack**, top to bottom
  - Instruments the software with performance metrics
  - Performance metrics are component-specific

- IBM develops the **performance data collectors** that harvest those performance metrics

- IBM i has an **integrated database** – DB2
  - This is a BIG DEAL
  - Performance data is stored in the database automatically
    - No "add on" application is necessary – it's all in the Operating System

  - Applications mine the performance data in the DB2 files using SQL

IBM i has the best performance instrumentation and data collection capabilities in the industry!

Performance Instrumentation and Data Collection

Analysis — Analysis applications mine this data

Integrated DB2 — Metric data is stored in DB2 files

Collectors — Performance Collectors harvest metric data

Instrumentation — Operating System Components identify and save metric data
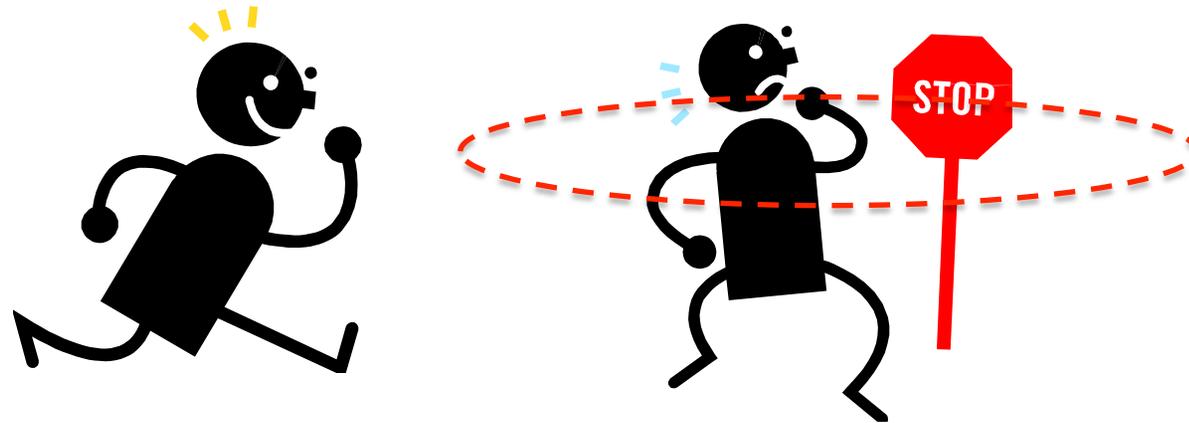
Introduction to Wait Accounting

Performance Fact:

"All computers wait at the same speed"

What is Wait Accounting?

Wait Accounting = the ability to determine what a job is doing when it is not running

i Exclusive!! Patented technology built into IBM i.

Wait Accounting Overview

When a job is not running (using CPU), it is waiting
- *But what is it waiting for?*

Waits may be normal, some waits are not normal
- Wait Accounting helps to determine what the wait is and if it is a problem

IBM i has instrumented most of the key wait conditions
- Wait information is automatically collected by **Collection Services** and **Job Watcher**

Wait States

Wait information is tracked for each job, thread and task on system

A job/thread/task is in one of three states:

Using CPU

- "Dispatched CPU"
  Assigned to a virtual processor so it can begin
  execution of instructions

Waiting for CPU

- "CPU Queuing"
  Ready to use processor, but waiting for it to
  become available

Waiting for something else...

- Idle waits
- **Blocked waits**

These waits are typically the most
interesting waits to focus on

Wait Accounting - Buckets

Wait Buckets = "Wait condition groups" instrumented in the operating system.

- Buckets can then be **analyzed** to determine where a job is spending it's time (running or waiting)

- Categorized into **32** buckets

- Buckets found in both **Collection Services** and **Job Watcher** data

- Waits can be viewed at a **system-level** or at an **individual job/ thread/task level**

  - Can also be grouped by generic job name, subsystem, user profile, pool ID, etc.

32 Wait Buckets (6.1 and beyond)

1. **Time dispatched on a CPU**
2. **CPU queuing**
3. Reserved
4. Other waits
5. **Disk page faults**
6. **Disk non-fault reads**
7. **Disk space usage contention**
8. **Disk operation start contention**
9. **Disk writes**
10. **Disk other**
11. **Journaling**
12. **Semaphore contention**
13. **Mutex contention**
14. **Machine level gate serialization**
15. **Seize contention**
16. **Database record lock contention**
17. **Object lock contention**

18. **Ineligible waits**
19. **Main storage pool contention**
20. Classic Java™ user including locks (to 6.1)
    → **(7.2) Journal save while active**
21. Classic Java JVM (up to 6.1)
22. Classic Java other (up to 6.1)
23. Reserved
24. Socket transmits
25. Socket receives
26. Socket other
27. IFS
28. PASE
29. Data queue receives
30. Idle/waiting for work
31. Synchronization Token contention
32. **Abnormal contention**

http://www.ibm.com/developerworks/ibmi/library/i-ibmi-wait-accounting/
http://public.dhe.ibm.com/services/us/igsc/idoctor/Job_Waits_White_Paper_61_71.pdf
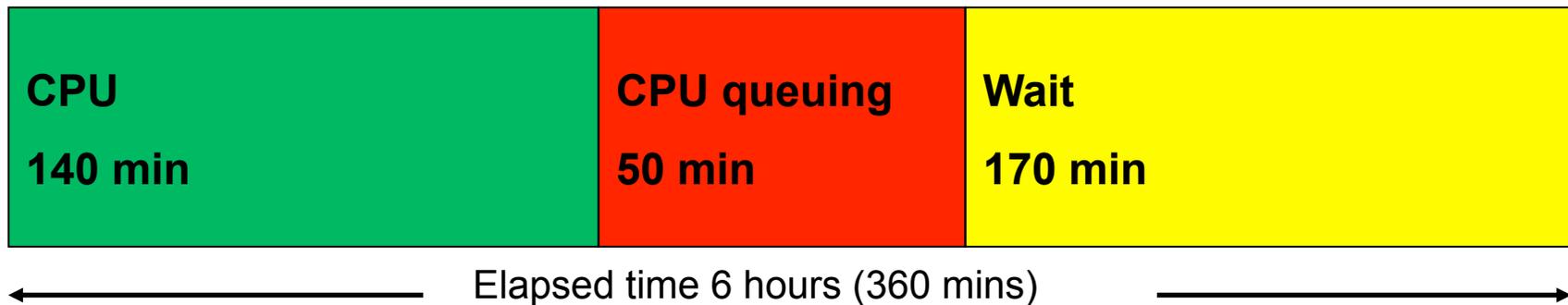
Wait Accounting – "Run-wait" signature

Applying the concepts of wait accounting, we are now able to identify the amount of time the thread/task was running and the time the thread/task was waiting.
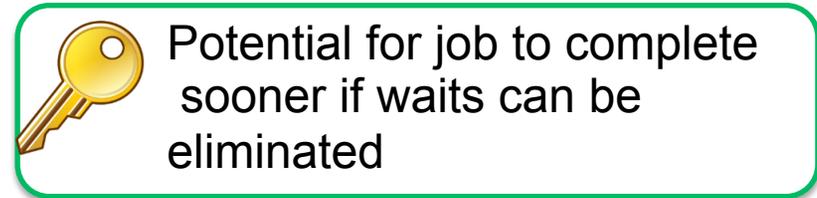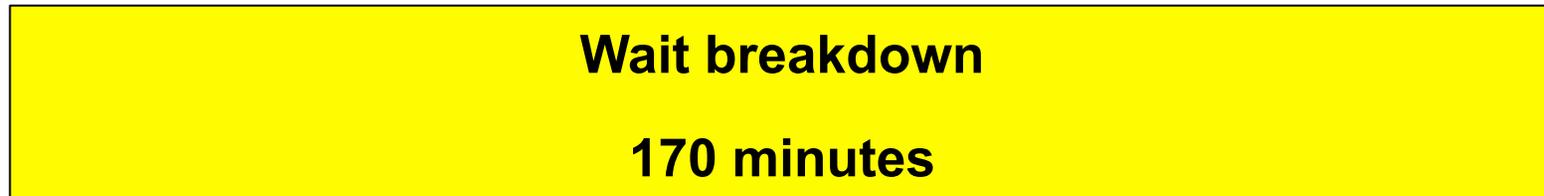
Consider the following:

Batch job with total run time of 6 hours

Run-wait signature

CPU 140 min	CPU queuing 50 min	Wait 170 min

← Elapsed time 6 hours (360 mins) →

Wait Accounting – "Run-wait" signature

Wait breakdown

170 minutes

Potential for job to complete sooner if waits can be eliminated

Disk Page Fault Time 100 mins	Disk Writes 30 mins	Journal 15 mins	DB Record Locking 15 mins	Object Lock 10 mins

Now you can start asking questions such as:
- Are my pool sizes appropriate? What objects is the faulting occurring on?
- Is the write cache being overrun? Is the application forcing writes out synchronously?
- Are all the journals optimally configured? Are unnecessary objects being journaled?
- Am I locking records or objects unnecessarily?

18

Metrics related to components of wait time

	Disk reads	Disk writes	Record Locks	Journal
Total count	3,523	17,772	355	5,741
Total time	42 sec	73 sec	45 sec	44 sec
Avg time per wait	0.012 sec	0.004 sec	0.126 sec	0.007 sec

Disk Page Faults Counts	Disk Page Faults Time (Seconds)
20498	26.37
7969	10.83
9080	15.86
1738	2.55
3124	1.22
1071	0.65
2013	0.94
3133	1.19
2939	1.36
1056	0.55

- Tools capture this level detail
- Useful to know both counts and time

Why Developers should leverage Wait Accounting!!

- Helps you understand application characteristics
  - Is it CPU bound? I/O bound?

- Helps you to understand where to focus your effort and investment
  - Is there a bottleneck on CPU, Memory, I/O, Contention time?
  - Invest resources where greatest benefit will be

- Can offer insight into potential performance issues before end-users are affected
  - Can leverage aspects of wait accounting in test environment
    - Eliminate surprises
    - Identify bottlenecks that prevent scaling

- Provides valuable clues to help analyze performance issues as they arise

- Instrumentation part of base IBM i operating system, IBM tools available to help you analyze

Common Waits that Applications use

- Disk Waits
- Semaphores, Mutexes, Synchronization Tokens
- Journaling
- Database record locks
- Object locks
- Sockets

A few other things to know about waits…

- Some waits are "expected" and others "unexpected"

- If waits can be reduced or eliminated, CPU can be used more efficiently

- One wait may be reduced/eliminated, only to have another wait surface

- Likely won't be able to remove all wait times

- When is a wait "bad"?
  - Is there a business impact? Are users complaining?
  - ***"It depends"*** but waits more than 25% of run time may need additional analysis

Tools for analyzing Wait Accounting information

Wait Accounting - IBM i Collectors

- **Collection Services**
  - Collects data automatically 24 X 7 at specified intervals (typically 5 or 15 minutes)
  - System and job level data
  - Starting point!

- **Job Watcher**
  - Needs to be started/stopped (typically 5 or 10 second intervals)
  - Additional detailed data such as call stacks, object waited on, holder
 - Frequently needed to solve performance issues

Performance Data
## *Collection vs Visualization*

- The performance data ***collection*** capability is built into the operating system
  - **Everyone** can collect all types of IBM i performance data

- The ***visualization*** of the data may require additional products

  - Visualization is the display of charts or tables of performance data

  - The Performance Data Investigator is part of the operating system
    - **Everyone** can visualize Collection Services data
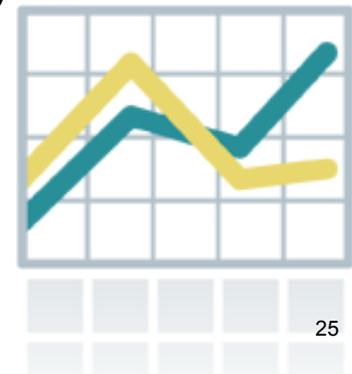
Two Graphical Analysis Tools

- **Performance Data Investigator – Job Watcher**
  - Requires 57xx-PT1 – Job Watcher feature
  - Geared to *average* user
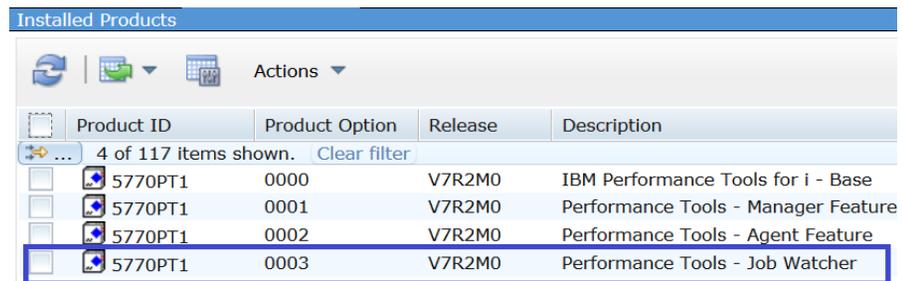
	Product ID	Product Option	Release	Description
	5770PT1	0000	V7R2M0	IBM Performance Tools for i - Base
	5770PT1	0001	V7R2M0	Performance Tools - Manager Feature
	5770PT1	0002	V7R2M0	Performance Tools - Agent Feature
	5770PT1	0003	V7R2M0	Performance Tools - Job Watcher

Installed Products — Actions — 4 of 117 items shown. Clear filter

- **iDoctor – Job Watcher**
  - IBM i Service offering, yearly license by serial number
  - Geared to *advanced* user

Wait Accounting IBM i Graphical Analysis Tools

- Two powerful graphical tools to help make your analysis more efficient and productive:

## Performance Data Investigator (PDI)

- Component in IBM Navigator for i (browser-based)
- Nothing to install, can view Collection Services for "free"
- http://www.ibm.com/developerworks/ibmi/library/i-pdi/index.html

IBM iDoctor for IBM i

- Microsoft Windows based client
- Requires Job Watcher yearly license to see Collection Services data (IBM Service offering)
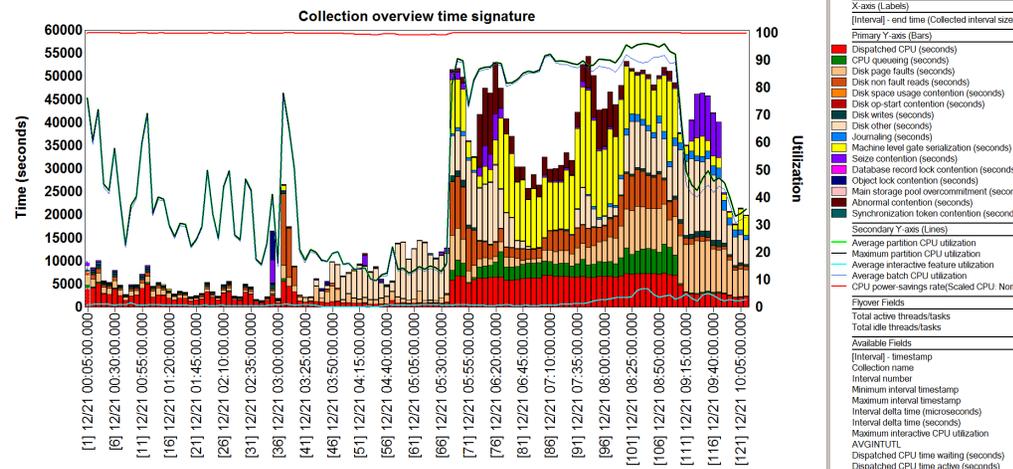- https://www-912.ibm.com/i_dir/idoctor.nsf

Wait Accounting IBM i Graphical Analysis Tools

- Both GUI tools sit on top of same rich IBM i instrumentation, but not equivalent in presentation and function
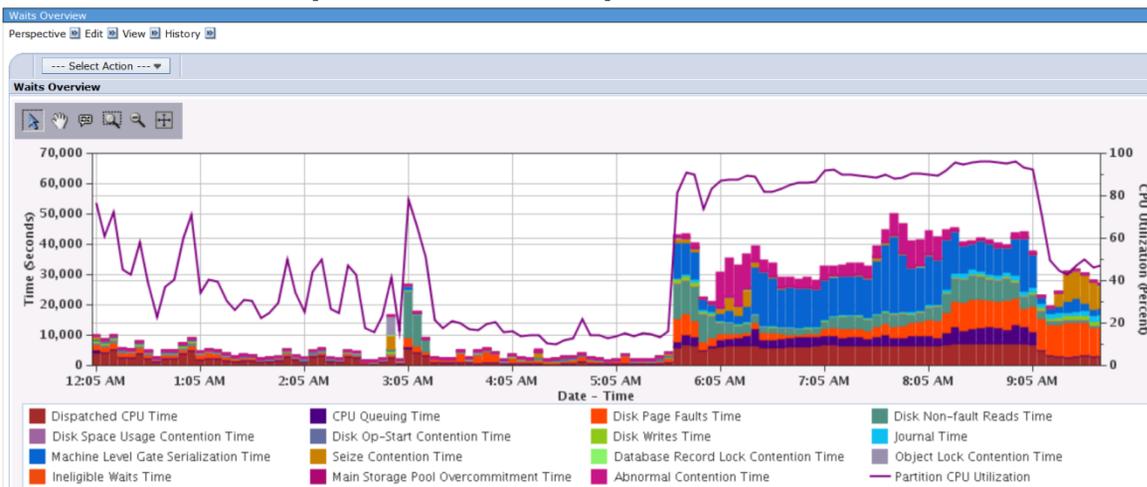
← PDI

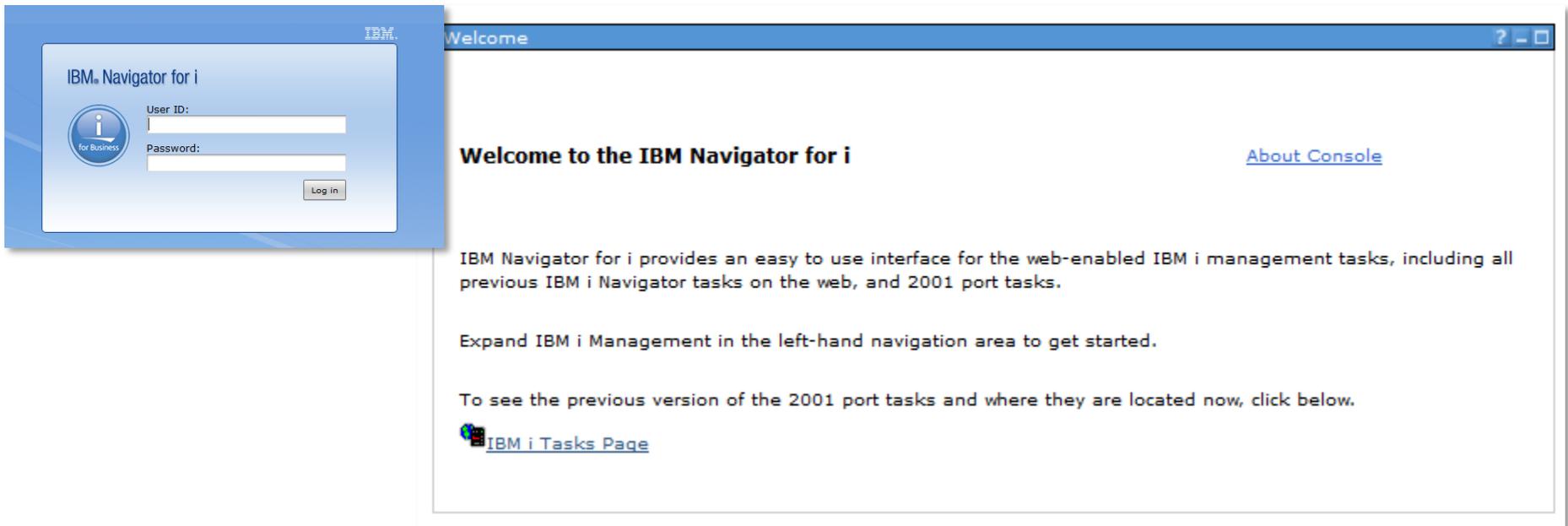iDoctor ->

Wait Accounting Analysis Strategy

- Understand the "big picture" first
  - Understand overall partition characteristics first and where system bottlenecks may be that affecting your application
  - Typically done using **Collection Services** data
  - Drill down to job level
  - Waits can be analyzed in various useful ways:
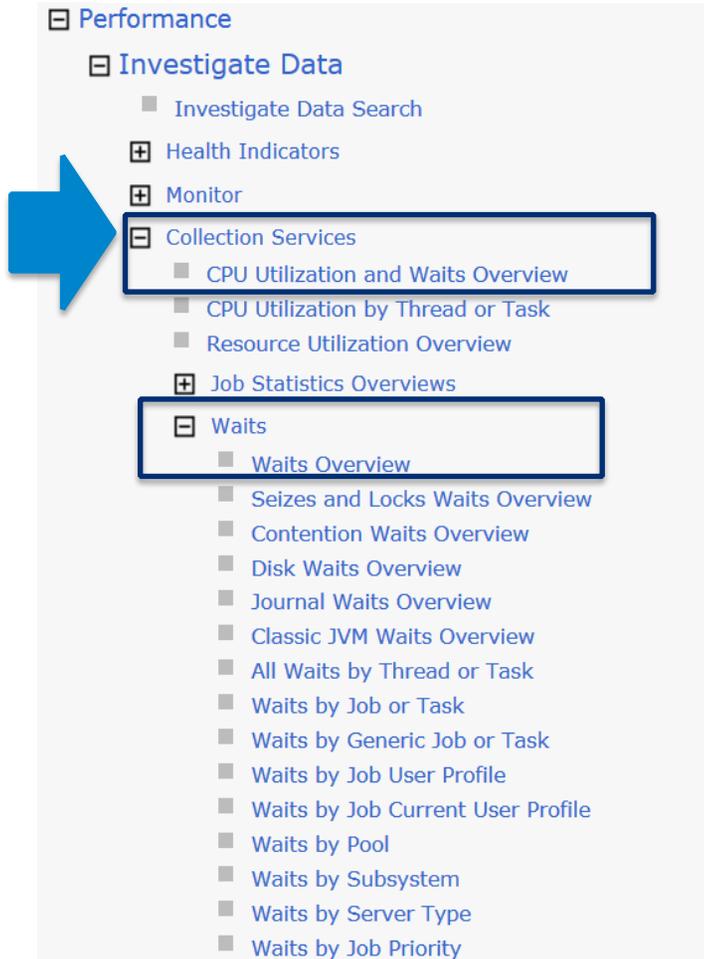
    - Waits by Job or Task
    - Waits by Generic Job or Task
    - Waits by Job User Profile
    - Waits by Job Current User Profile
    - Waits by Pool
    - Waits by Subsystem
    - Waits by Server Type
    - Waits by Job Priority

- Continue detailed analysis at a Job Level using **Job Watcher**
  - Narrow focus to interesting timeframes / jobs
  - Many more job level details available

Using Performance Data Investigator (PDI)

- IBM Navigator for i is the Web console for managing IBM i
    - Has much of the function as System i Navigator
        - but with a browser user interface

    - Simply point your browser to http://systemname:2001

IBM Navigator for i

User ID:

Password:

Log in

Welcome

Welcome to the IBM Navigator for i   About Console

IBM Navigator for i provides an easy to use interface for the web-enabled IBM i management tasks, including all previous IBM i Navigator tasks on the web, and 2001 port tasks.

Expand IBM i Management in the left-hand navigation area to get started.

To see the previous version of the 2001 port tasks and where they are located now, click below.

IBM i Tasks Page

PDI Wait Accounting Perspectives - Where to start

Performance
  Investigate Data
    Investigate Data Search
    ⊞ Health Indicators
    ⊞ Monitor
    ⊟ Collection Services
        CPU Utilization and Waits Overview
        CPU Utilization by Thread or Task
        Resource Utilization Overview
      ⊞ Job Statistics Overviews
      ⊟ Waits
          Waits Overview
          Seizes and Locks Waits Overview
          Contention Waits Overview
          Disk Waits Overview
          Journal Waits Overview
          Classic JVM Waits Overview
          All Waits by Thread or Task
          Waits by Job or Task
          Waits by Generic Job or Task
          Waits by Job User Profile
          Waits by Job Current User Profile
          Waits by Pool
          Waits by Subsystem
          Waits by Server Type
          Waits by Job Priority

Collection

Collection Library   Collection Name
QPFRDATA   Q201000002 (*CSFILE) - Jul 20, 2015 12:00:02 AM

Performance -> Investigate Data -> Collection Services:

Option 1: **CPU Utilization and Waits Overview**
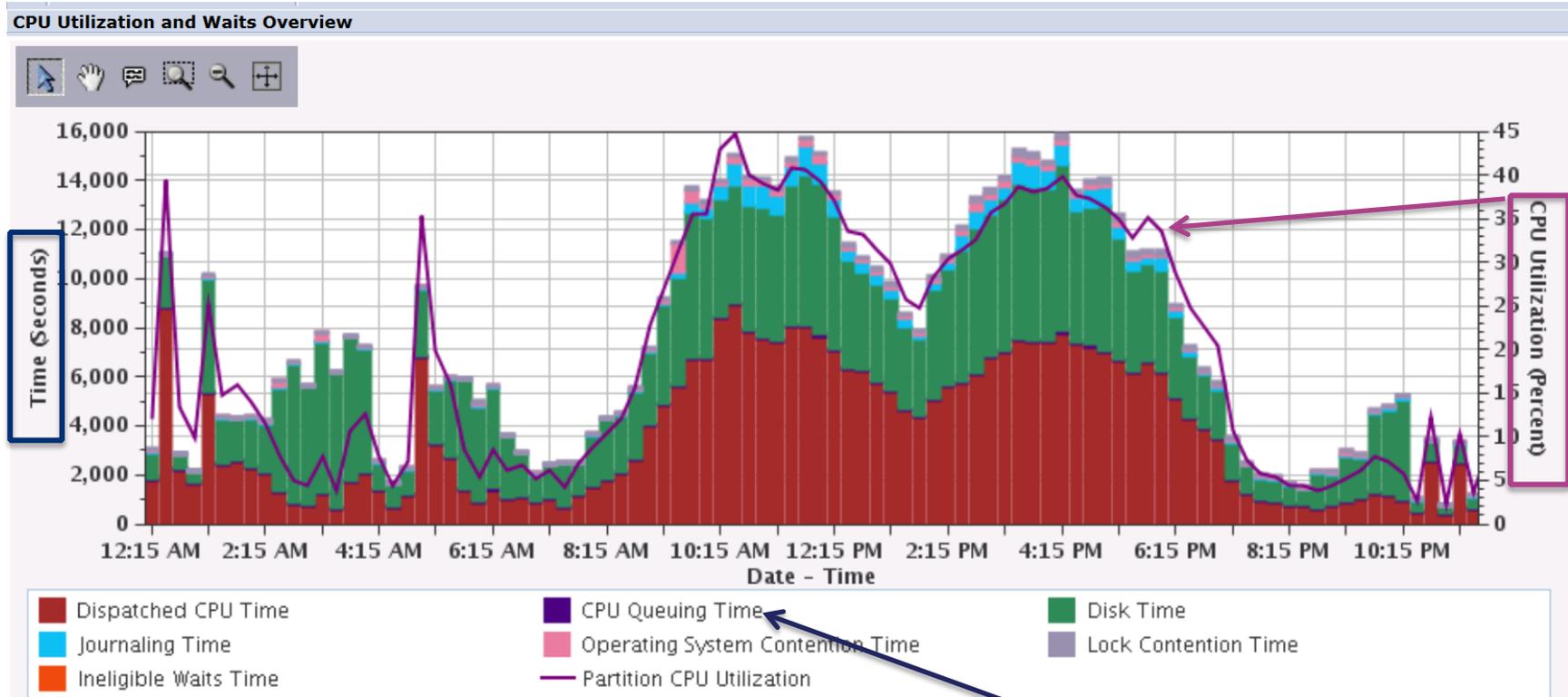– Combines related waits into higher level buckets

Option 2: **Waits Overview**
– All individual "blocked" wait buckets shown

System "run-wait" signature -> CPU Utilization and Waits Overview

CPU Utilization and Waits Overview

- A chart that combines CPU utilization as well as the wait buckets can be very beneficial in assessing the health of your partition
- In this chart, we can see that the majority of the time, the jobs were spending time in CPU as well as in Disk. Minor amounts of Journal wait time and operating system contention time are also present.

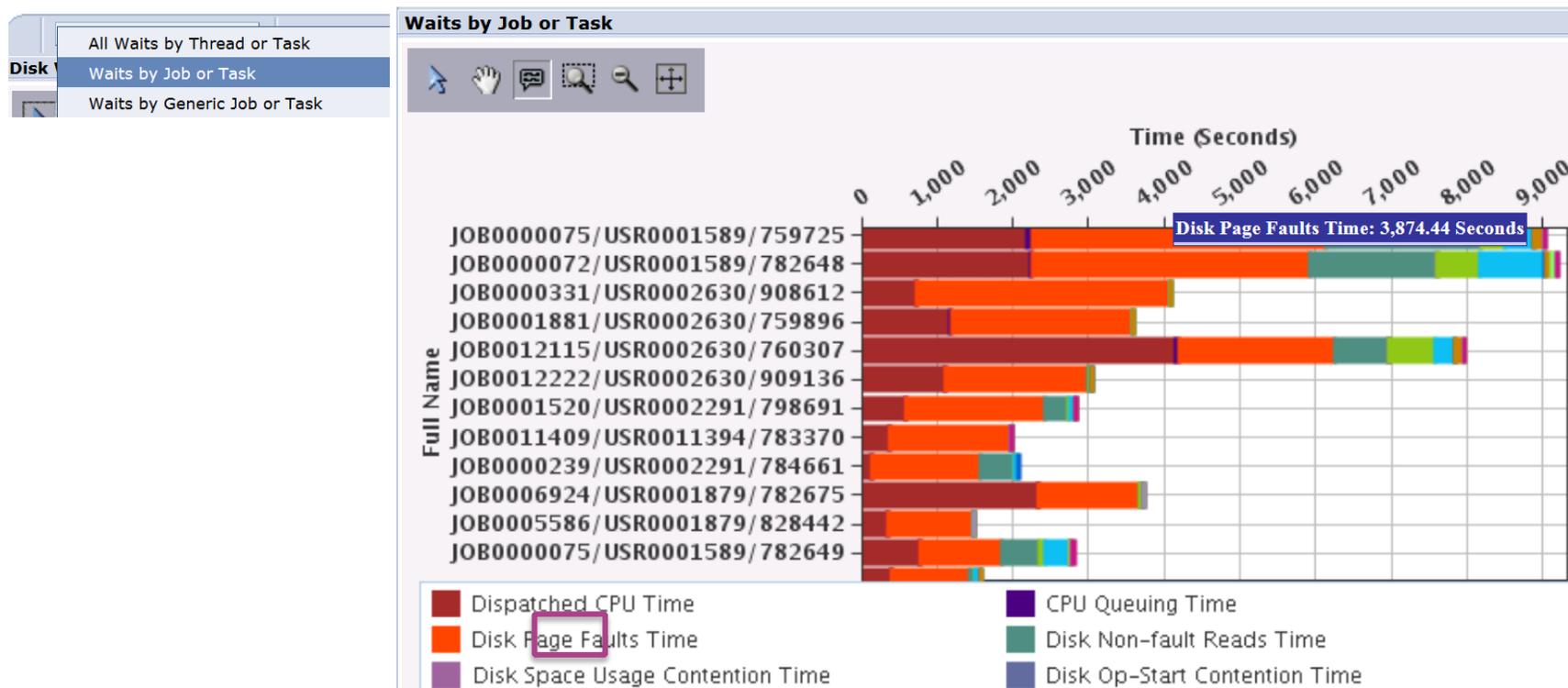# CPU Utilization and Waits Overview – drilldown analysis

Because Disk wait time was fairly significant, drilldown to Disk Waits Overview to further examine the detailed waits contributing to this time:
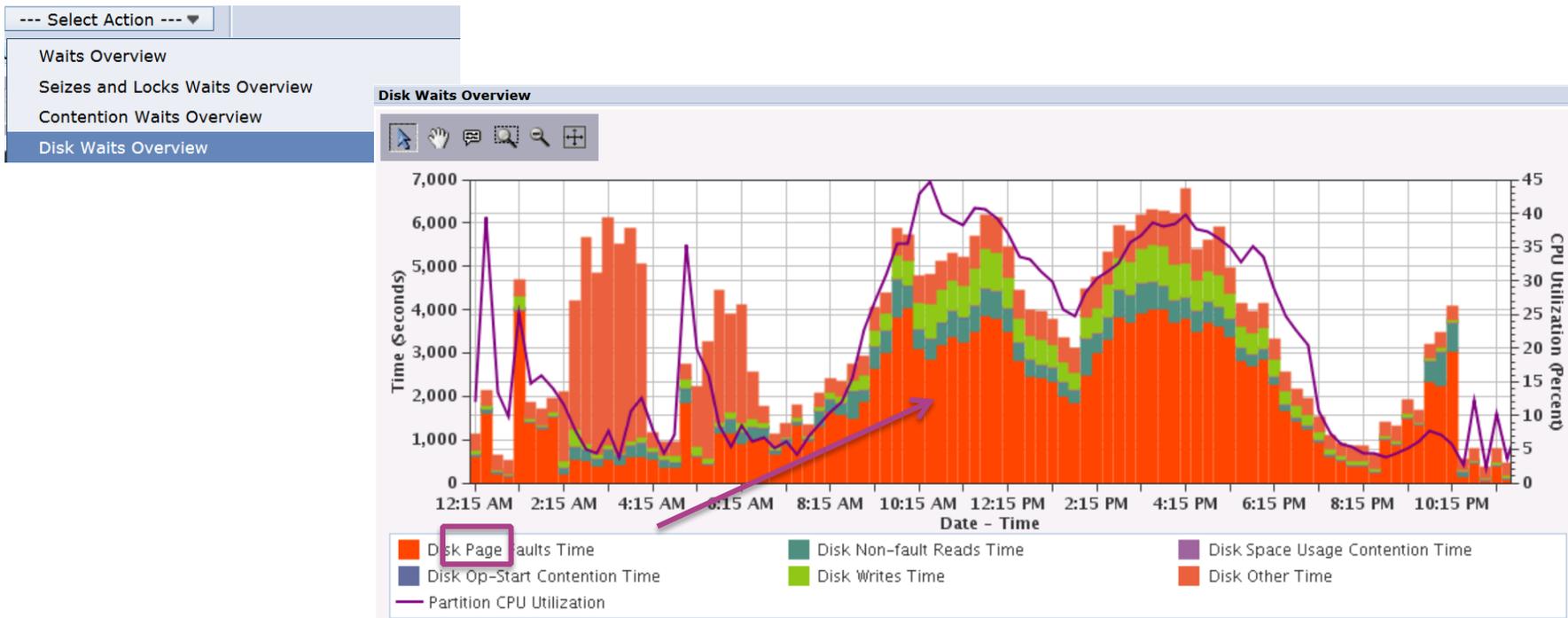
- Can now see that Disk Page Fault time is the biggest contributor to Disk Time. (A job needed something in memory, it wasn't there, had to do an I/O to bring it into memory before job could continue running).
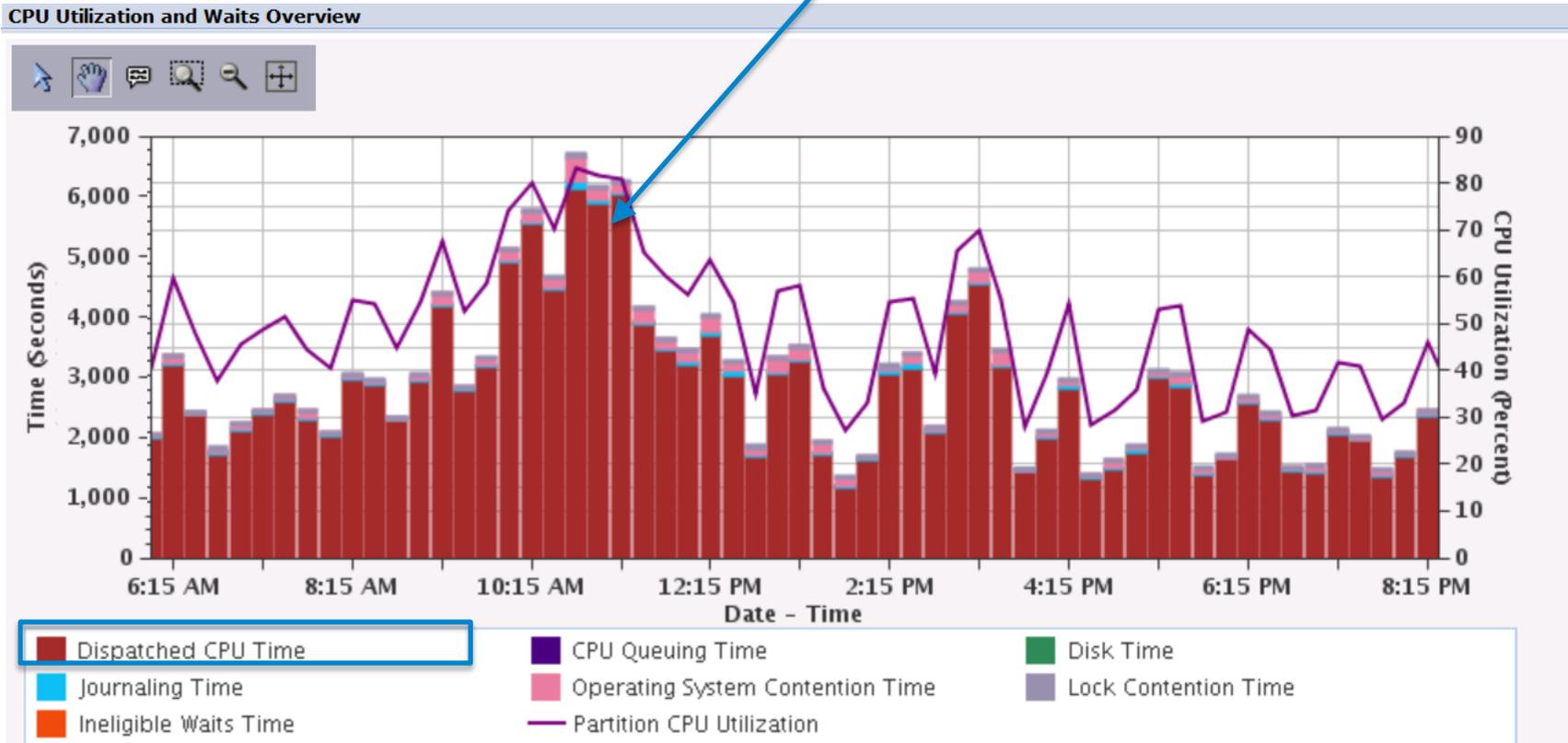
Waits by Job or Task

The next question likely would be which job(s) are incurring this wait time. Drilling down further, we can see the list of jobs incurring this wait time:

- This type of chart can also be used to understand a job(s) "run-wait" signature.

Efficient System with Little Waiting

Primarily Dispatched CPU Time

CPU Utilization and Waits Overview

Legend:
- Dispatched CPU Time
- CPU Queuing Time
- Disk Time
- Journaling Time
- Operating System Contention Time
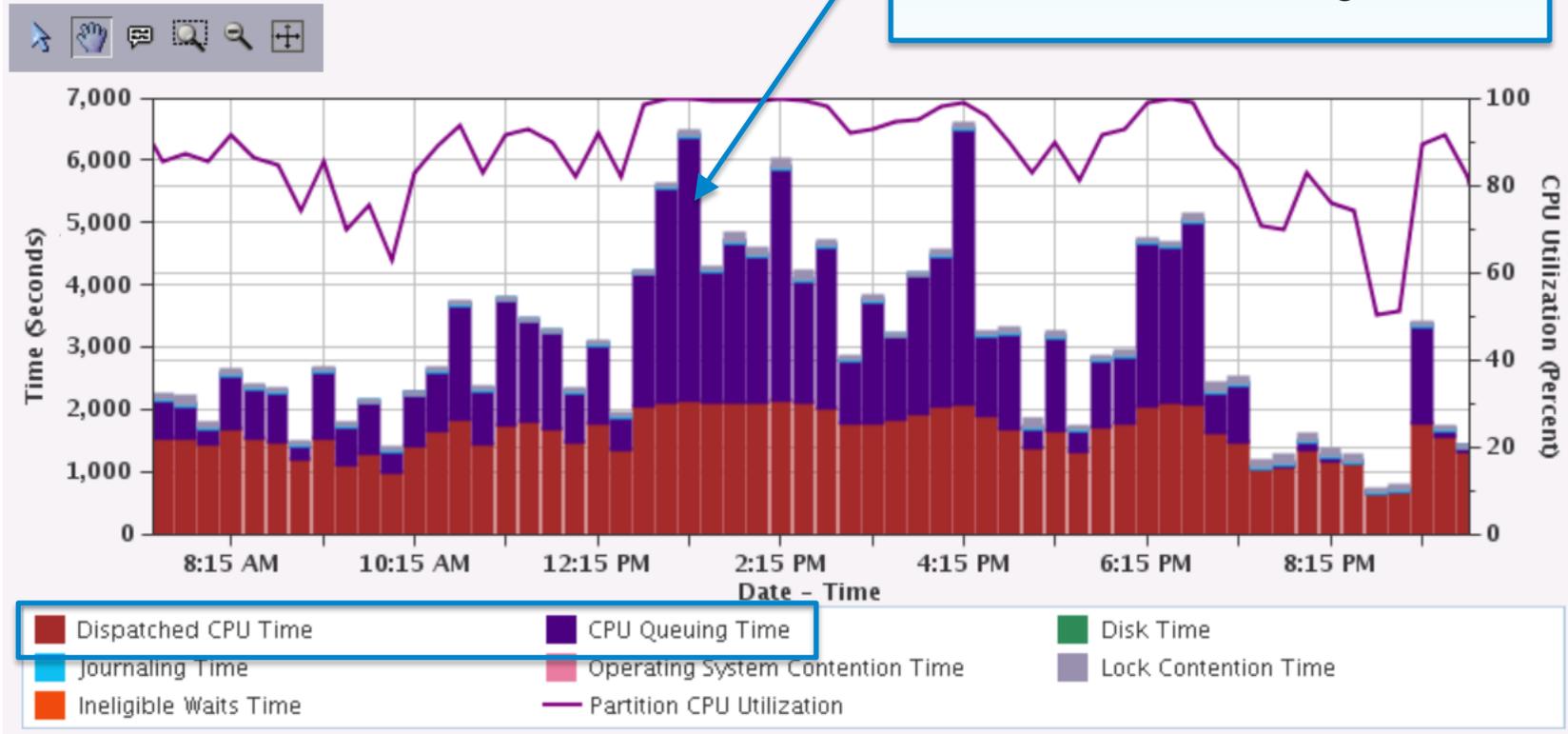- Lock Contention Time
- Ineligible Waits Time
- Partition CPU Utilization

Processor Bound System

Dispatched CPU
+
CPU Queuing

CPU Utilization and Waits Overview

I/O Bound System

Disk Time

CPU Utilization and Waits Overview
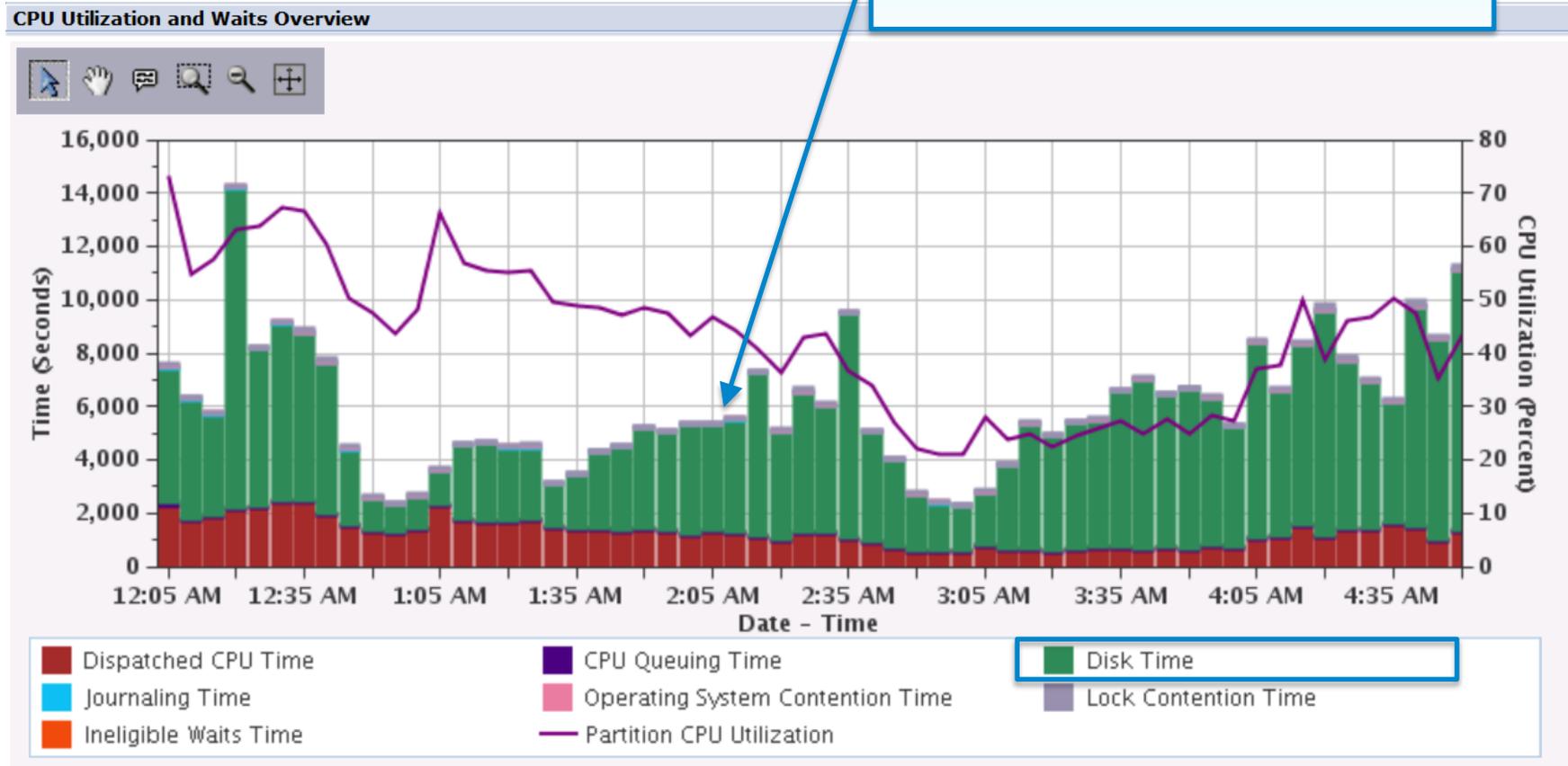
I/O – Further Investigation

Disk Page Faults Wait Time

Disk Waits Overview

Legend:
- Disk Page Faults Time
- Disk Op-Start Contention Time
- Partition CPU Utilization
- Disk Non-fault Reads Time
- Disk Writes Time
- Disk Space Usage Contention Time
- Disk Other Time
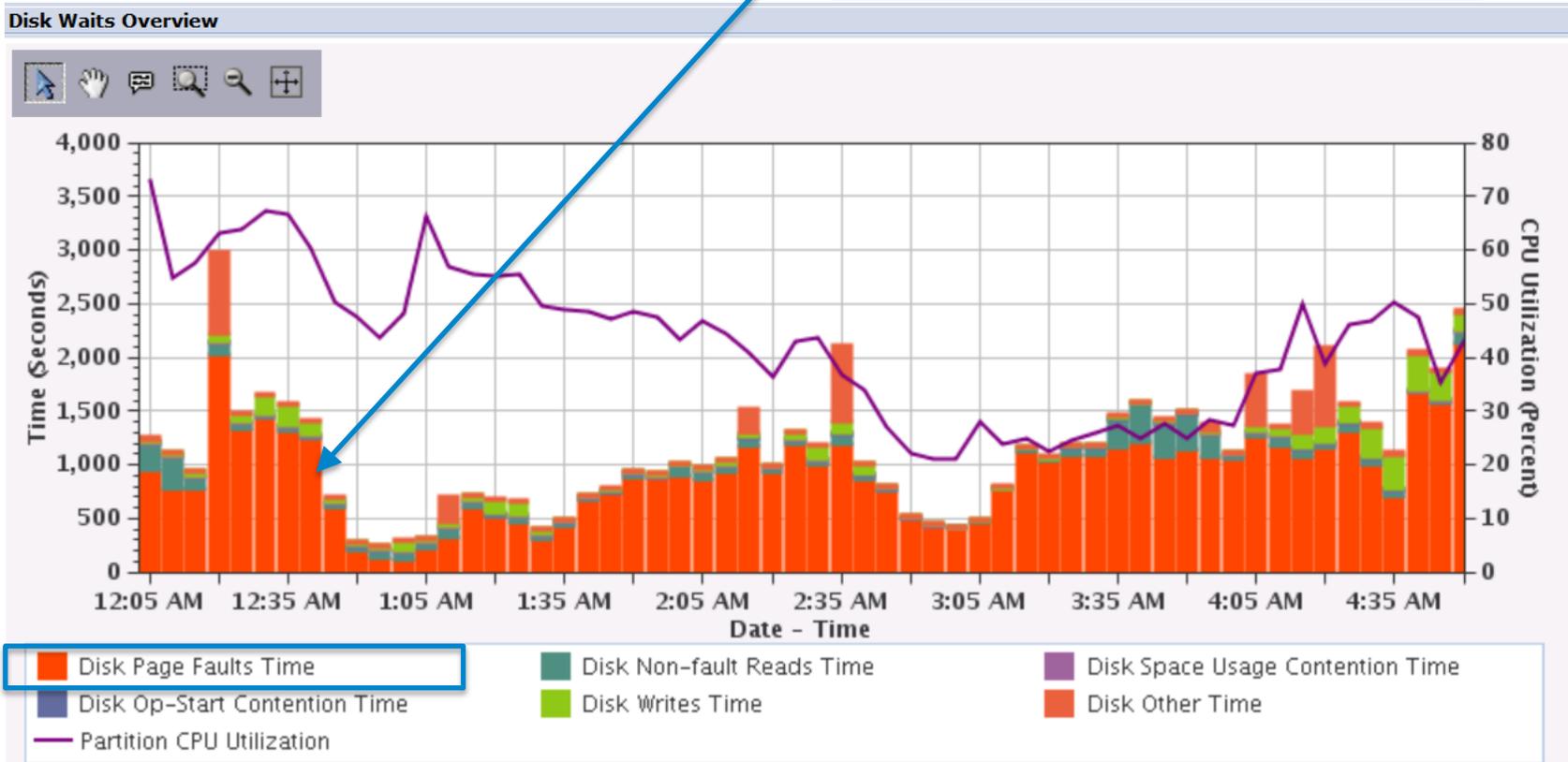
Lock Contention Time Bottleneck

Lock Contention wait time

CPU Utilization and Waits Overview

Legend:
- Dispatched CPU Time
- CPU Queuing Time
- Disk Time
- Journaling Time
- Operating System Contention Time
- Lock Contention Time
- Ineligible Waits Time
- Partition CPU Utilization
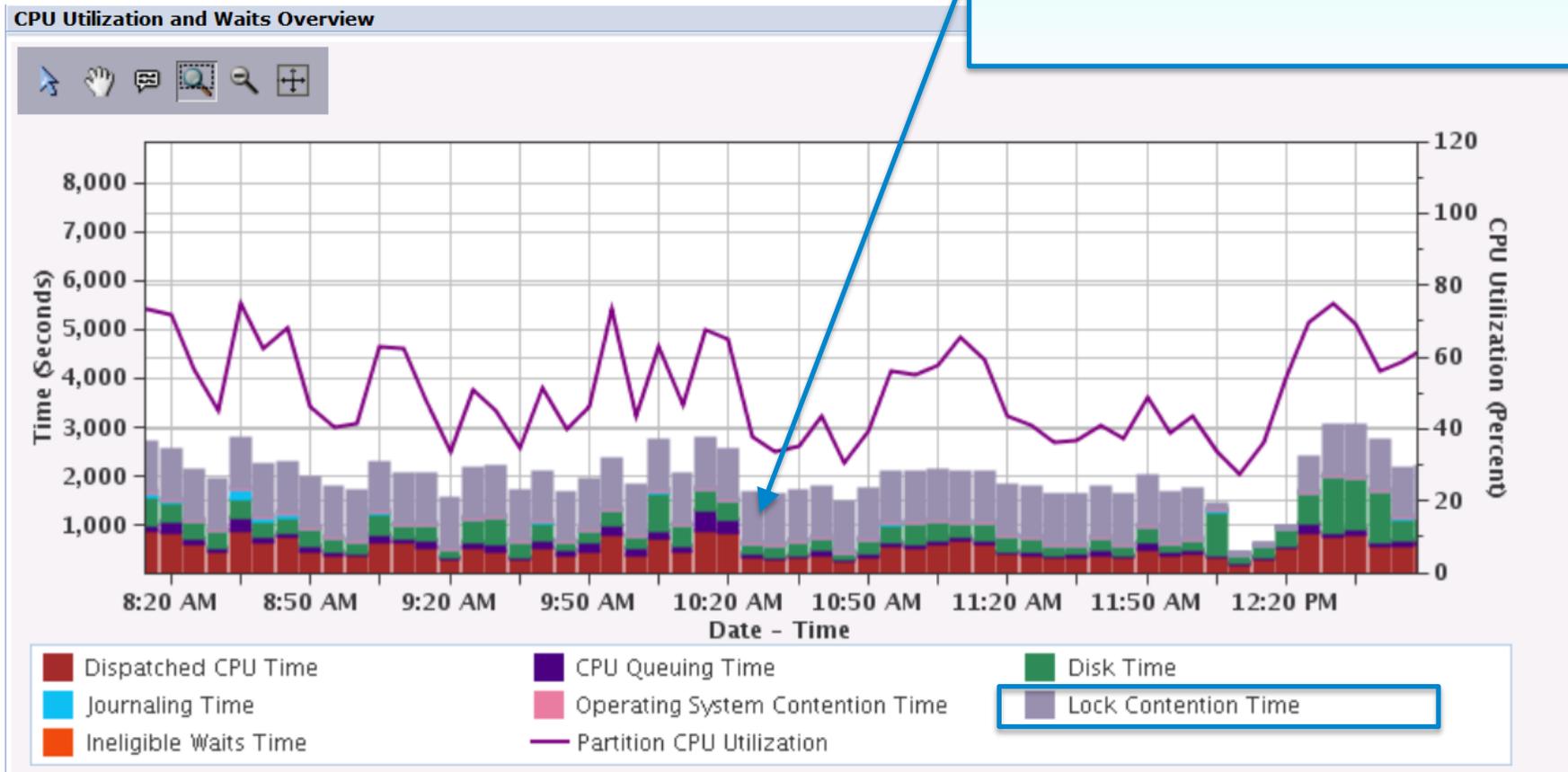
Job Watcher data is typically needed to solve lock related issues.

Lock – Further Investigation

Object Lock Contention time

--- Select Action --- ▼

CPU U

Waits Overview

Seizes and Locks Waits Overview

Contention Waits Overview

Seizes and Locks Waits Overview
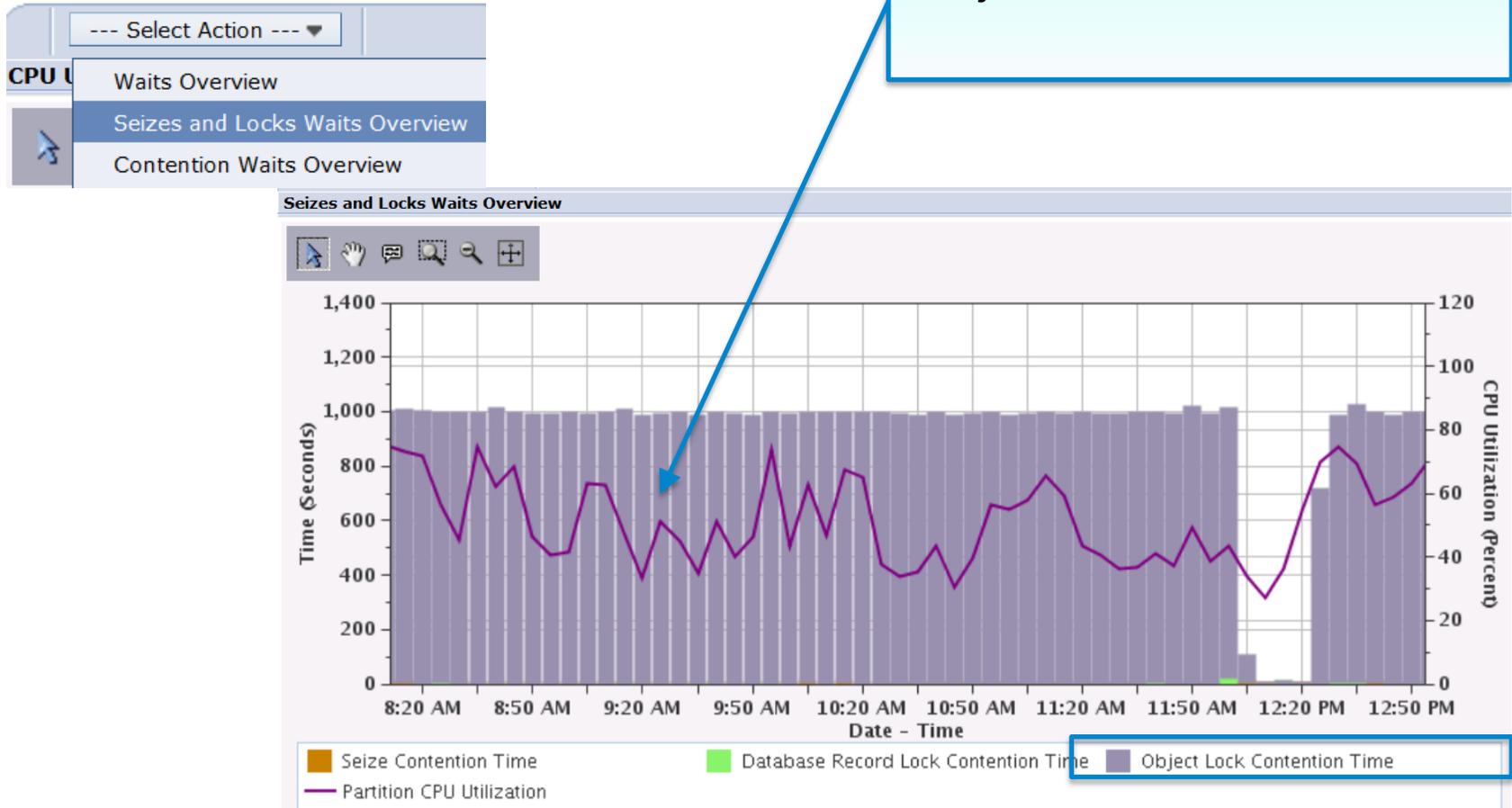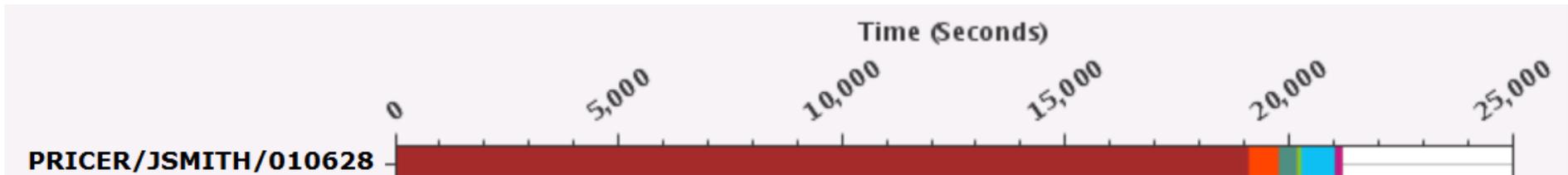
Job Watcher data will show object waited on, the holder, and call stacks for both the waiter and the holder (example shown later on...)
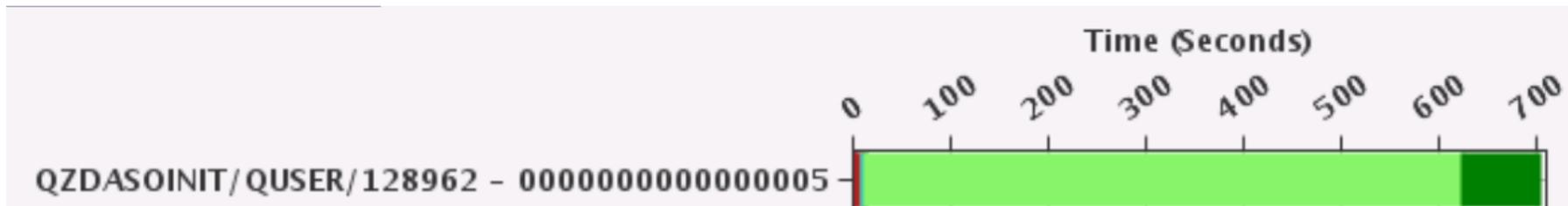
Wait Accounting at a Job Level

? Would this job benefit from additional memory?

Time (Seconds)

```
              0      5,000    10,000    15,000    20,000    25,000
PRICER/JSMITH/010628 ─[██████████████████████████████████████]──
```

■ Dispatched CPU Time	■ CPU Queuing Time	■ Disk Page Faults Time
■ Disk Non-fault Reads Time	■ Disk Space Usage Contention Time	■ Disk Op-Start Contention Time
■ Disk Writes Time	■ Journal Time	■ Machine Level Gate Serialization Time
■ Seize Contention Time	■ Database Record Lock Contention Time	■ Object Lock Contention Time
■ Socket Receives Time	■ Main Storage Pool Overcommitment Time	■ Abnormal Contention Time

Wait Accounting at a Job Level

? Would this job benefit from additional memory? CPU? Disk?

Time (Seconds)

0   100   200   300   400   500   600   700

QZDASOINIT/QUSER/128962 – 0000000000000005

■ Dispatched CPU Time	■ CPU Queuing Time	■ Disk Page Faults Time
■ Disk Non-fault Reads Time	■ Disk Space Usage Contention Time	■ Disk Op-Start Contention Time
■ Disk Writes Time	■ Journal Time	■ Machine Level Gate Serialization Time
■ Seize Contention Time	■ Database Record Lock Contention Time	■ Object Lock Contention Time
■ Socket Receives Time	■ Main Storage Pool Overcommitment Time	■ Abnormal Contention Time

Wait Accounting at a Job Level

? Would this job benefit from an improved I/O subsystem?

Time (Seconds)

100 200 300 400 500 600 700

CUSTMRPT02/EDGE/128871 – 00000000000000D2

■ Dispatched CPU Time	■ CPU Queuing Time	■ Disk Page Faults Time
■ Disk Non-fault Reads Time	■ Disk Space Usage Contention Time	■ Disk Op-Start Contention Time
■ Disk Writes Time	■ Journal Time	■ Machine Level Gate Serialization Time
■ Seize Contention Time	■ Database Record Lock Contention Time	■ Object Lock Contention Time
■ Socket Receives Time	■ Main Storage Pool Overcommitment Time	■ Abnormal Contention Time

Job Watcher - Additional Benefits

- Collects **more detailed** performance data than Collection Services
  - Call Stacks
  - SQL Statements
  - Additional wait accounting information:
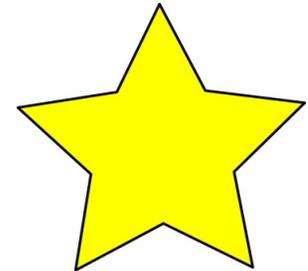   - Objects being waited on
    - Holder of object

- More **frequent intervals** (seconds)

- Need to start/stop Job Watcher
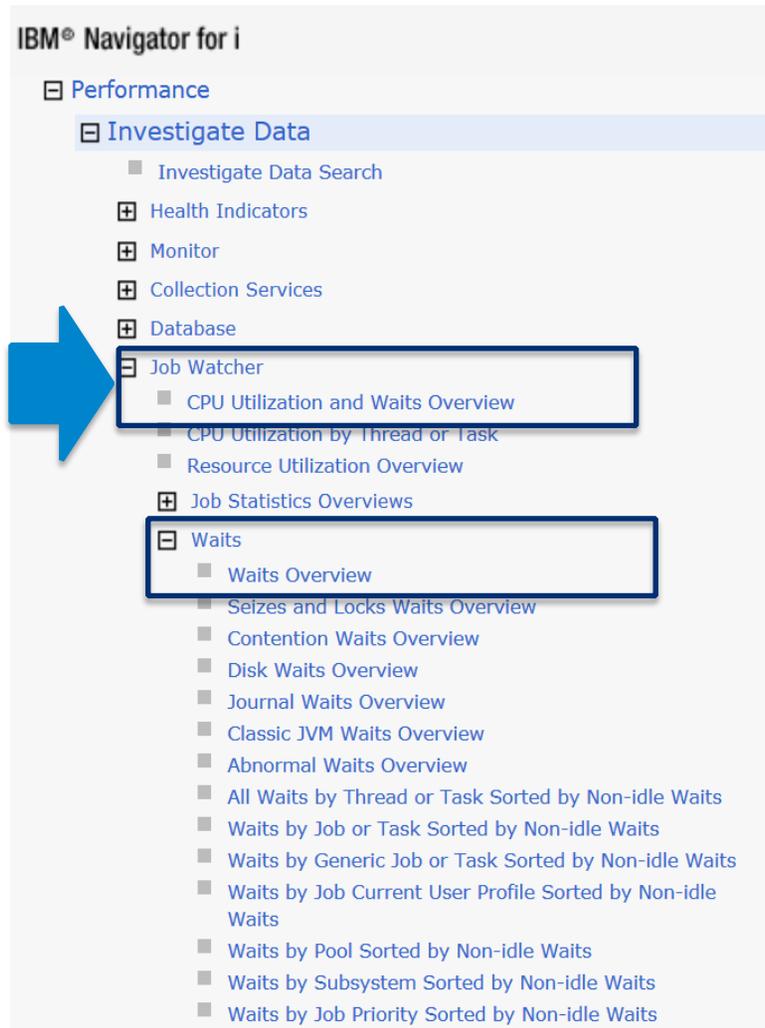 - Navigator for i, iDoctor, green screen commands

- To see charts in PDI, need Performance Tools LPP Job Watcher option (chargeable) or iDoctor Job Watcher license for viewing in iDoctor

Job Watcher - Holders versus Waiters

- IBM i keeps track of who is holding a resource, and if applicable, who is waiting to access that resource
  - A **Holder** is the job/thread/task that is holding the serialized resource
  - A **Waiter** is the job/thread/task that wants to access the serialized resource

- IBM i also maintains call stacks for every job/thread/task

- The combination of
  - **Who** - holders and waiters … *who has it? who wants it?*
  - **What** – object being waited on
  - **How** - call stacks

  provides a very powerful solution for analyzing wait conditions

Job Watcher – Where to Start

IBM® Navigator for i

Performance
  Investigate Data
    Investigate Data Search
    ⊞ Health Indicators
    ⊞ Monitor
    ⊞ Collection Services
    ⊞ Database
    ⊟ Job Watcher
      CPU Utilization and Waits Overview
      CPU Utilization by Thread or Task
      Resource Utilization Overview
      ⊞ Job Statistics Overviews
      ⊟ Waits
        Waits Overview
        Seizes and Locks Waits Overview
        Contention Waits Overview
        Disk Waits Overview
        Journal Waits Overview
        Classic JVM Waits Overview
        Abnormal Waits Overview
        All Waits by Thread or Task Sorted by Non-idle Waits
        Waits by Job or Task Sorted by Non-idle Waits
        Waits by Generic Job or Task Sorted by Non-idle Waits
        Waits by Job Current User Profile Sorted by Non-idle Waits
        Waits by Pool Sorted by Non-idle Waits
        Waits by Subsystem Sorted by Non-idle Waits
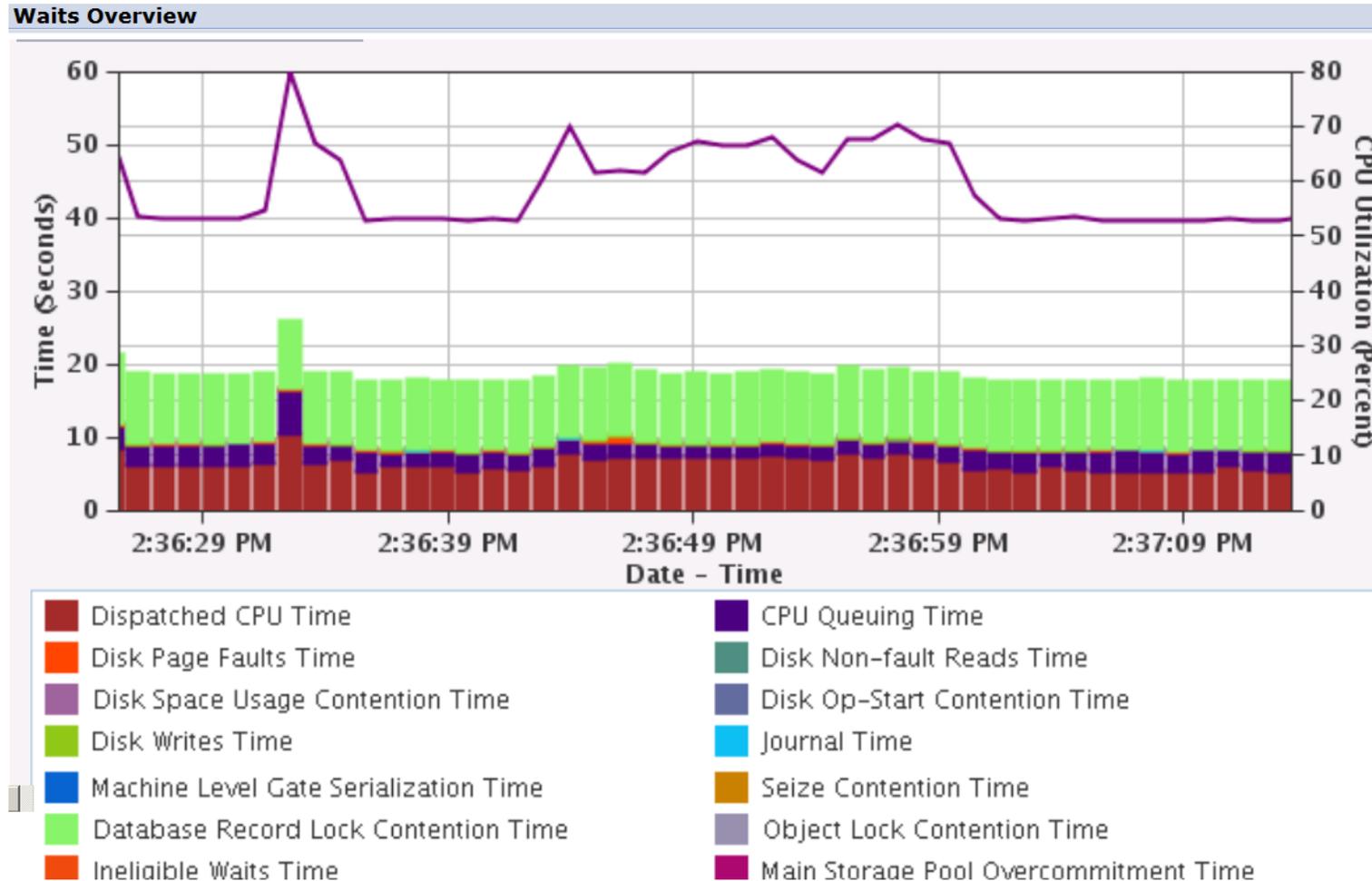   Waits by Job Priority Sorted by Non-idle Waits

Performance -> Investigate Data -> Job Watcher:

Option 1: **CPU Utilization and Waits Overview**
– Combines related waits into higher level buckets

Option 2: **Waits Overview**
– All individual "blocked" wait buckets shown

Notice similar perspectives available as Collection Services

Job Watcher – Waits Overview

Waits Overview

Legend:
- Dispatched CPU Time
- CPU Queuing Time
- Disk Page Faults Time
- Disk Non-fault Reads Time
- Disk Space Usage Contention Time
- Disk Op-Start Contention Time
- Disk Writes Time
- Journal Time
- Machine Level Gate Serialization Time
- Seize Contention Time
- Database Record Lock Contention Time
- Object Lock Contention Time
- Ineligible Waits Time
- Main Storage Pool Overcommitment Time

Notice same wait buckets, but more granular intervals

Job Watcher – Additional Interval Details

Thread or Task Details

Job information:	QZDASOINIT/QUSER/128962 - 0000000000000005	Priority	20
Current user profile:	LISAW	Pool:	2
Object waited on:	INVENTORY INVENTORY	Type description:	PHYSCIAL FILE MBR - DATA PART
Wait duration:	581 milliseconds	Segment type description:	DB PHYSICAL FILE MEMBER RECORDS
Current or last wait:	DB record lock: update	Wait object library:	None detected this interval
Holding job or task:	QZDASOINIT/QUSER/128890	Interval timestamp:	Jan 3, 2014 2:36:28 PM
SQL client job:	None detected this interval	Interval (1 to 684):	< 174 >

Show Holder

Call Stack

| --- Select Action --- ▼			
Call Level	**Program**	**Module**	**Procedure**
1			qutde_block_tra
2			longWaitReceive__9QuCounterFR12RmprReceiverPv
3			DBLockConflict__15RmslDBHashClassFR11RmslPlm
4			rmslDBHLock__FR11RmslPlmpLad
5			getLockWithWait__18DbpmUpdateResourcede
6			getLock__18DbpmUpdateResourcead
7			getRowLock__18DbpmUpdateResourceFCUlRC9Dbp
8			execute__18DbpmUpdateLockNodeFR13DbpmQuer
9			vPositionNextAndExecute__18DbpmUpdateLockNod
10			positionNextEntryAndFetchOutline__17DbpmReadO

SQL Statement

☑ Include Host Variables

SELECT QUANTITY FROM WAREHSE42.INVENTORY WHERE ID=*DATA FORMAT ERRORTITY FROM WAREHSE42.INVENTORY WHERE ID=? FOR UPDATE

Wait Accounting - Recommendations: Be proactive!

- Use the rich IBM i wait accounting instrumentation found in
  - Collection Services & Job Watcher
 - Use PDI or iDoctor to view/analyze

- Understand your partition's "run-wait" signature and normal patterns
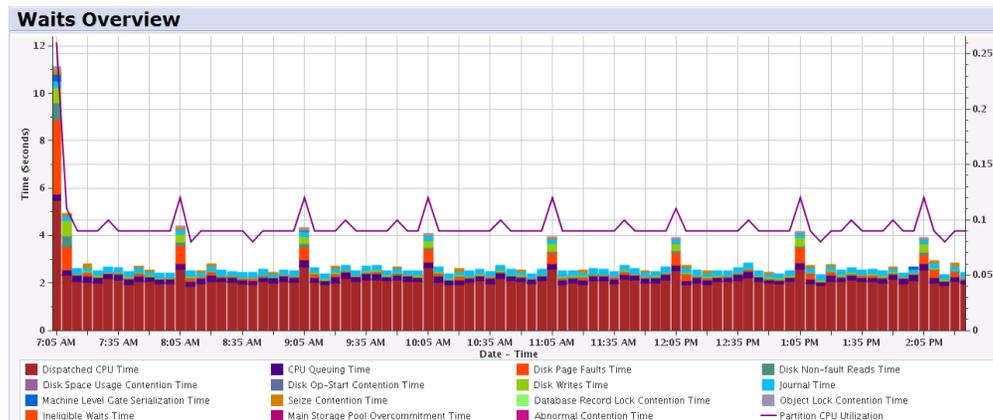
- Identify bottlenecks

Recommendations: Be proactive!

- Keep a **baseline**
  - Collection Services (Job Watcher data is also nice to have)
    - Weekly, end-of-month, end-of-year
    - Prior to any hardware, software, configuration related change
- A baseline provides a **reference point**
  - It is the expected performance characteristics over a defined period of time
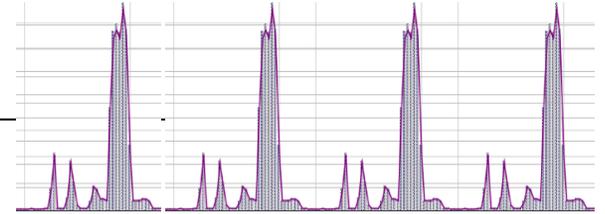 - Having one makes it easier to recognize changes and its effect
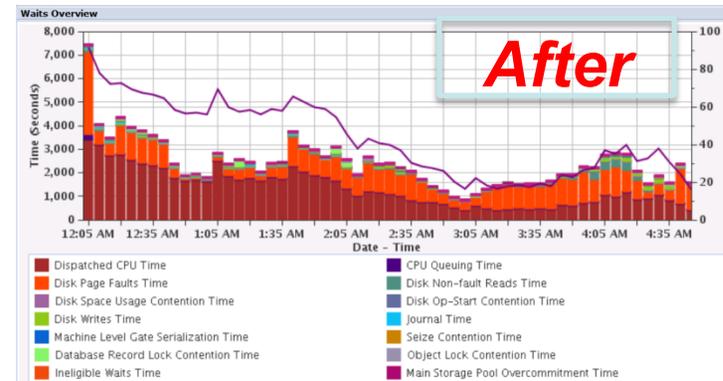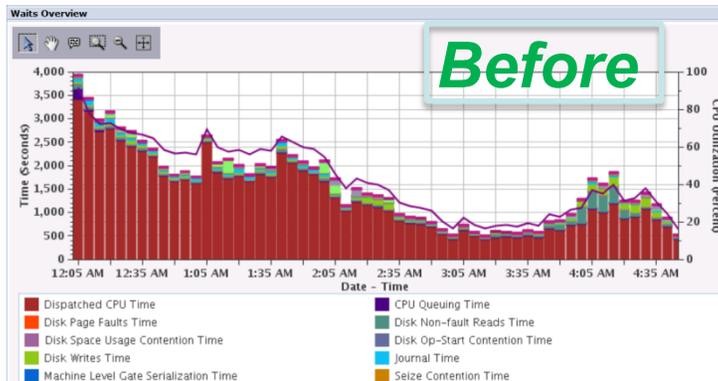
- **<u>Wait bucket information can make it easier to determine what has changed!
  Both at a partition level as well as an individual job level</u>**

IBM i Performance Data Collectors

Performance Data Collection Architecture

- Collection Services

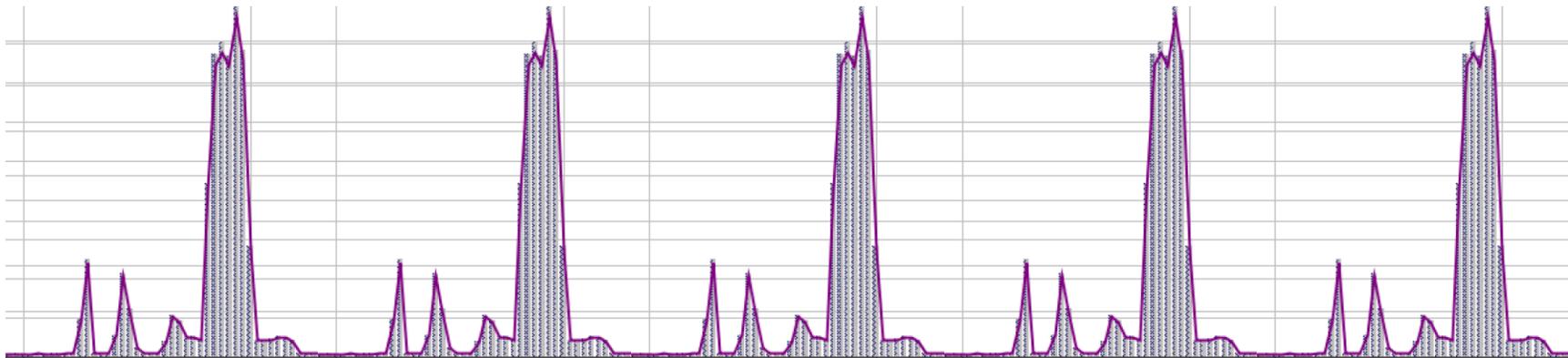- Job Watcher

- Disk Watcher

- Performance Explorer
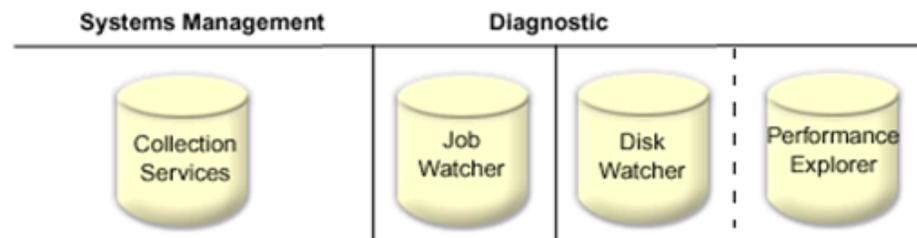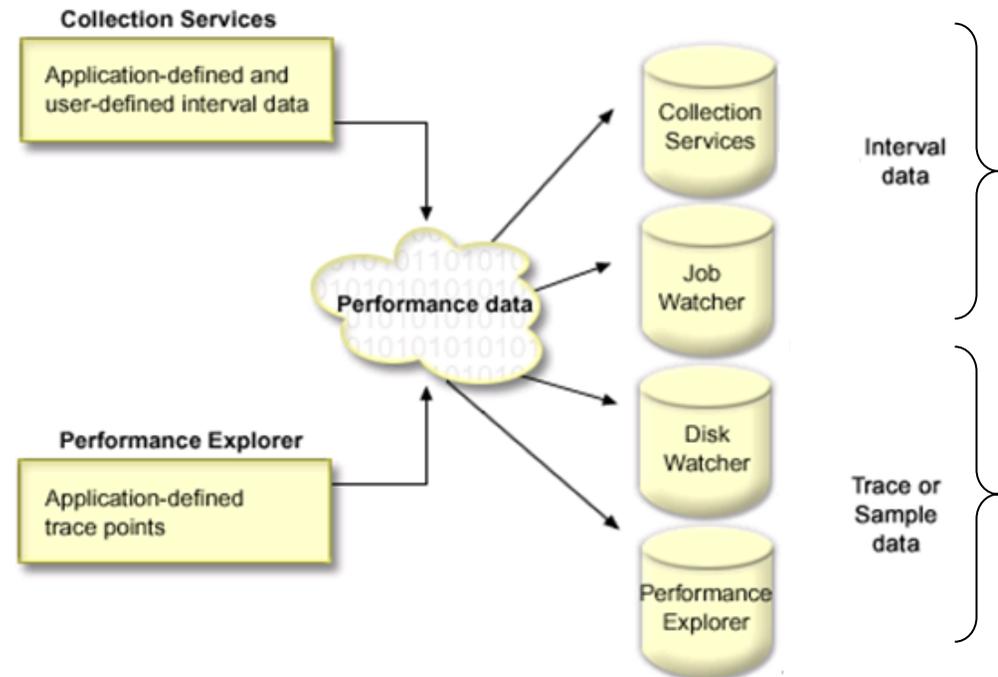
Collect System-wide Performance Data

Collect Performance Data 24/7

- If something goes wrong, you have data that will help **analyze** the problem, **fix** it, and **prevent** it from happening in the future

- If you can't solve the problem, you have information that makes it easier for IBM Support to **solve the problem faster**

- To provide a **reliable baseline** so you can **understand the impact** that a software, network, or environmental **change** had on the performance of your system

- To provide historical information that enables you to **plan for future growth** based on real trends, not guesses.

Patterns in Performance Data

- Performance data typically has patterns

  - Daily, weekly, monthly, yearly

  - Understand your typical patterns

  - Recognize change

Job Watcher

- Job Watcher returns real-time information about a selected set of jobs, threads, or LIC tasks

- Job Watcher collects additional types of data that Collection Services does not, as well as more frequent intervals
  - Job Watcher has more overhead than Collection Services

- Data collected by Job Watcher includes
  - Wait times
  - CPU
  - I/O activity
  - Call Stacks
  - SQL statements
  - Communications statistics
  - Activation Group statistics

Run Job Watcher when you need detailed performance data for diagnostic purposes.

There are clients that run Job Watcher 24x7 to always have diagnostic data available.

Need to manage the data carefully.

Job Watcher

- Job Watcher collects **more detailed** performance data than Collection Services and at **more frequent intervals**
  - CPU and I/O (like Collection Services)
  - **Call Stacks**
  - SQL Statements
  - Detailed Wait information:
    - **Objects being waited on**, even records number of files
    - **Holder of object**
- Job Watcher **does not collect everything** that Collection Services collects.
  - It does not always collect information about every thread
    - Thread must use CPU during interval
    - Thread must exist for entire interval
  - It does not collect memory pool or detailed I/O statistics

- Data is written to DB2 files

Job Watcher Usage Tips

- Use Job Watcher when you need detailed performance data to **resolve a problem**

  - Typically problem has been scoped first by Collection Services

- For problem determination Job Watcher can be run on **specific jobs**

  - **Caution:** When using Job Watcher on specific job(s), you may not get detailed Holder information

- Multiple collections can be run at the same time

- Need to manage the amount of data collected
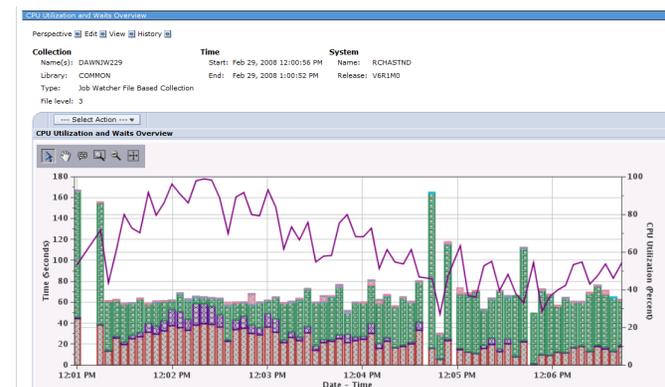
Basic Job Watcher Data Collection Steps

1. Create the Job Watcher definition
    – Or use one of the IBM-supplied definitions

2. Start the Job Watcher collection

3. Let it run until the problem has occurred

4. Stop the Job Watcher collection

5. Analyze the data

There are times when you may want to run Job Watcher continuously

# How Do I Run Job Watcher with the Commands?
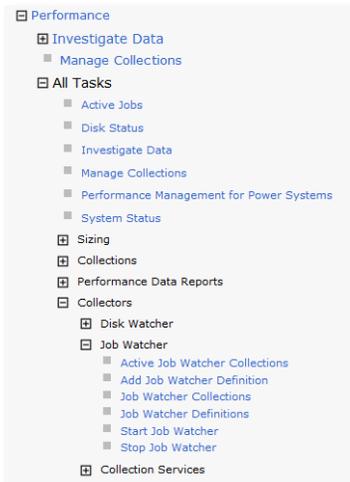
- **CL Commands**
  - Add Job Watcher Definition (ADDJWDFN) to define the collection
    - Identifies the performance data that is to be collected

  - Remove Job Watcher Definition (RMVJWDFN) to remove a definition
    (Note: Job Watcher Definitions can only be displayed through the GUI)

  - Start Job Watcher (STRJW) to start the collection
  - End Job Watcher (ENDJW) to end the collection (optionally)

IBM-Supplied Job Watcher Definitions

- Several pre-defined Job Watcher definitions are available
  - The main difference is the sample intervals
    - Q1SEC* – 1 second intervals with call stacks, another to include SQL
    - Q10SEC* –10 second intervals with call stacks, another to include SQL
    - Q5SEC* – 5 second intervals with call stacks, another to include SQL
- Recommendations:
  - Collect with Call Stacks and SQL
  - Use 10 second intervals for general analysis (Q10secsql)
  - Use 5 second intervals for complex or intermittent issues, or for contention related problems (Q5secsql)
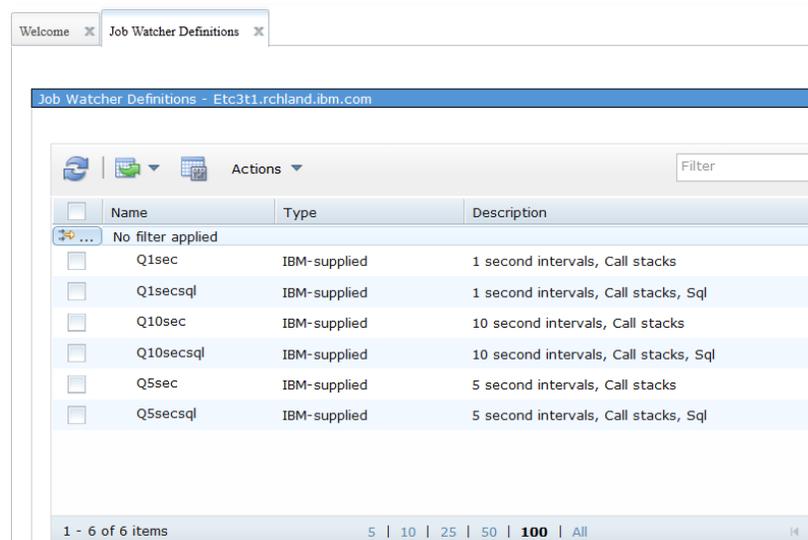
Job Watcher Authority Requirements

- Commands:
    1. You must have service (*SERVICE) special authority
        - Change User Profile to add ***SERVICE** authority to create Job Watcher Definitions or to Start Job Watcher
    2. -**OR**- Be authorized to the Job Watcher function of the operating system
        - *Change Function Usage (CHGFCNUSG) command, with a function ID of QIBM_SERVICE_JOB_WATCHER can be used to change the list of users that are allowed to use this command.*

        CHGFCNUSG FCNID(QIBM_SERVICE_JOB_WATCHER) USER(<usrprofile>) USAGE(*ALLOWED)

- Definitions:
    – Additional authority is needed to see the definitions for each as they are shipped *PUBLIC *EXCLUDE. To see the definitions shipped in Job Watcher, users will need authority to the QAPYJWDFN file in QUSRSYS

https://www.ibm.com/developerworks/community/wikis/home?lang=en#!/wiki/IBM%20i%20Technology%20Updates/page/Authority

Performance Explorer

- Performance Explorer helps identify the causes of performance problems that cannot be resolved using one of the other performance data collectors

  – Collects more detailed information about a specific application, program, or resource

- Performance Explorer is typically used for two main reasons:

  – Detailed performance trace data is needed to identify the performance problem
  – Analyzing the performance of applications

- Performance Data Investigator supports profile collections only
- iDoctor is required for advanced PEX Analysis

Performance Explorer

- Performance Explorer is the most sophisticated IBM i performance tool
  - Can collect the details of every I/O operation, every task switch
  - Hundreds of events collected
  - Thus, most complex to use
  - More overhead

- Typically, problem has been scoped by other tools first

- Generally used by IBM performance analysis experts

Except…..

Performance Explorer – "TPROF" usage

- "Trace-profile" is a fairly easy, and fairly "light-weight" PEX collection that can be useful to application developers, especially when trying to diagnose high CPU issues
  - Provides CPU usage at a program/module/procedure level
  - Make sure you have latest PEX PTFs applied
    http://www-01.ibm.com/support/docview.wss?uid=nas8N1012020
  - Can be run over subset of jobs
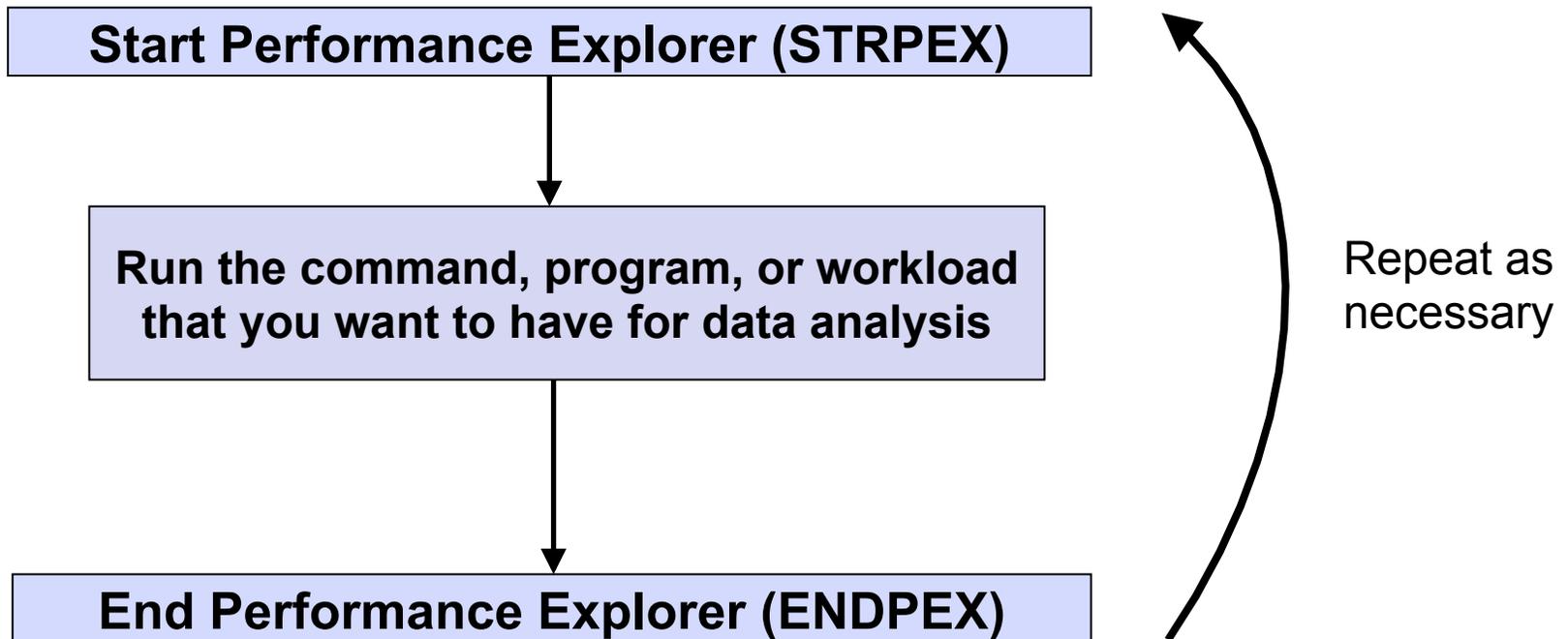
Steps:

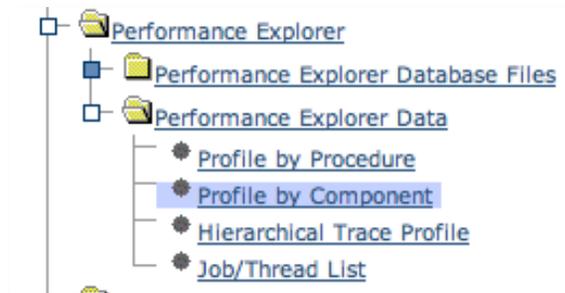1. Add a PEX definition:

```
ADDPEXDFN DFN(TPROF)
    TYPE(*TRACE)
    JOB((*ALL *ALL))
    TASK(*ALL)
    MAXSTG(4000000)
    INTERVAL(1)
    TRCTYPE(*SLTEVT)
    SLTEVT(*YES)
    MCHINST(*NONE)
    BASEVT((*PMCO *NONE *FORMAT2))
```

Performance Explorer – "TPROF" usage

2. Collect data

Start Performance Explorer (STRPEX)

Run the command, program, or workload that you want to have for data analysis

End Performance Explorer (ENDPEX)

Repeat as necessary

3. Analyze data


```
□─ 🗂 Performance Explorer
   ■─ 📁 Performance Explorer Database Files
   □─ 🗂 Performance Explorer Data
      ●─ Profile by Procedure
      ●─ Profile by Component
      ●─ Hierarchical Trace Profile
      └─ ● Job/Thread List
```

Performance Explorer TPROF reports – PDI

Profile by Procedure

Perspective ▣ Edit ▣ View ▣ History ▣

--- Select Action --- ▾

Program Name	Module Name	Procedure Name	Component	Hit Count
CFTSMPI		#cftsmpi	SLIC Common Functions	332(48.61%)
STRHU		do_copyMemoryLarge	SLIC String Functions	94(13.76%)
DBRSQMN		#dbrsqmn	SLIC Database	85(12.45%)
CUSTOMER CUSTOMER		#DBXFMP2	MI Other	45(6.59%)
READER	READER	READER	MI Other	27(3.95%)
DBPM2010		sExecute__42VariableLen	SLIC Database	6(0.88%)
SMMUTLH		trimRangeForRead__14Sn	SLIC Storage Management	6(0.88%)
HvString		HvString	SLIC Hypervisor	4(0.59%)
SMMSSUBH		findStealablePage__20Sm	SLIC Storage Management	4(0.59%)
QDBGETM	QDBGETM	QDBGETM	XPF Database Other	4(0.59%)

Profile by Component

--- Select Action --- ▾

Select		Total	Component	Procedure Name	Hit Count
☐	▾	Total			683(100%)
☐	▸		SLIC Common Functions		335(49.05%)
☐	▾		SLIC Database		118(17.28%)
☐				#dbrsqmn.#dbrsqmn	85(12.45%)
☐				sExecute__42VariableLeng	6(0.88%)
☐				sExecute__14HashOperatic	3(0.44%)
☐				sExecute__17PackedDivide	2(0.29%)
☐				sSad__19VariableLengthFic	2(0.29%)
☐				vPositionNextAndExecute__	2(0.29%)

Job/Thread List

--- Select Action --- ▾

Select		Name	Task/Thread Name	Cumulative CPU Time in Seconds	Active Time in Seconds	Hit Count
☐	▾	Total		143.50993	221051.77	509401459333(100%)
☐	▾	QDFTJOBD		68.429	123.51712	443134332927(86.99%)
☐	▾	WEBADM		68.429	123.51712	443134332927(86.99%)
☐	▾	QDFTJOBD/WEBADM/012		68.429	123.51712	
☐		0000000000000004	MITHREAD	68.429	123.51712	
☐	▸	QZDASOINIT		8.869434	1354.2184	
☐	▸	QPADEV0005		0.462739	123.545746	
☐	▸	ADMIN1		0.10104	5105.88	
☐	▾	ADMIN2		0.101269	9519.301	
☐	▾	QLWISVR		0.101269	9519.301	

Hierarchical Trace Profile

--- Select Action --- ▾

Select		Name	Hit Count
☐	▾	Total	683(100%)
☐	▸	LIC	606(88.73%)
☐	▾	DB Code Burst	45(6.59%)
☐	▾	CUSTOMER CUSTOMER	45(6.59%)
☐	▾	#DBXFMP2	45(6.59%)
☐		Amount of elements that did not match the filter: 1	0(0%)
☐	▸	MI ILE	27(3.95%)
☐	▸	MI OPM	4(0.59%)
☐	▸	Unknown	1(0.15%)

Watches

- Watches provide a programmatic interface to be notified when the following occur:
  - Message
  - Licensed Internal Code Log (LIC Log)
  - Problem Activity Log Entry (PAL entry)

- Start Watch (STRWCH) command or API (QSCSWCH)
- End Watch (ENDWCH) command or API (QSCEWCH)

- When the condition being watched occurs, your program gets control and you can take any action you want
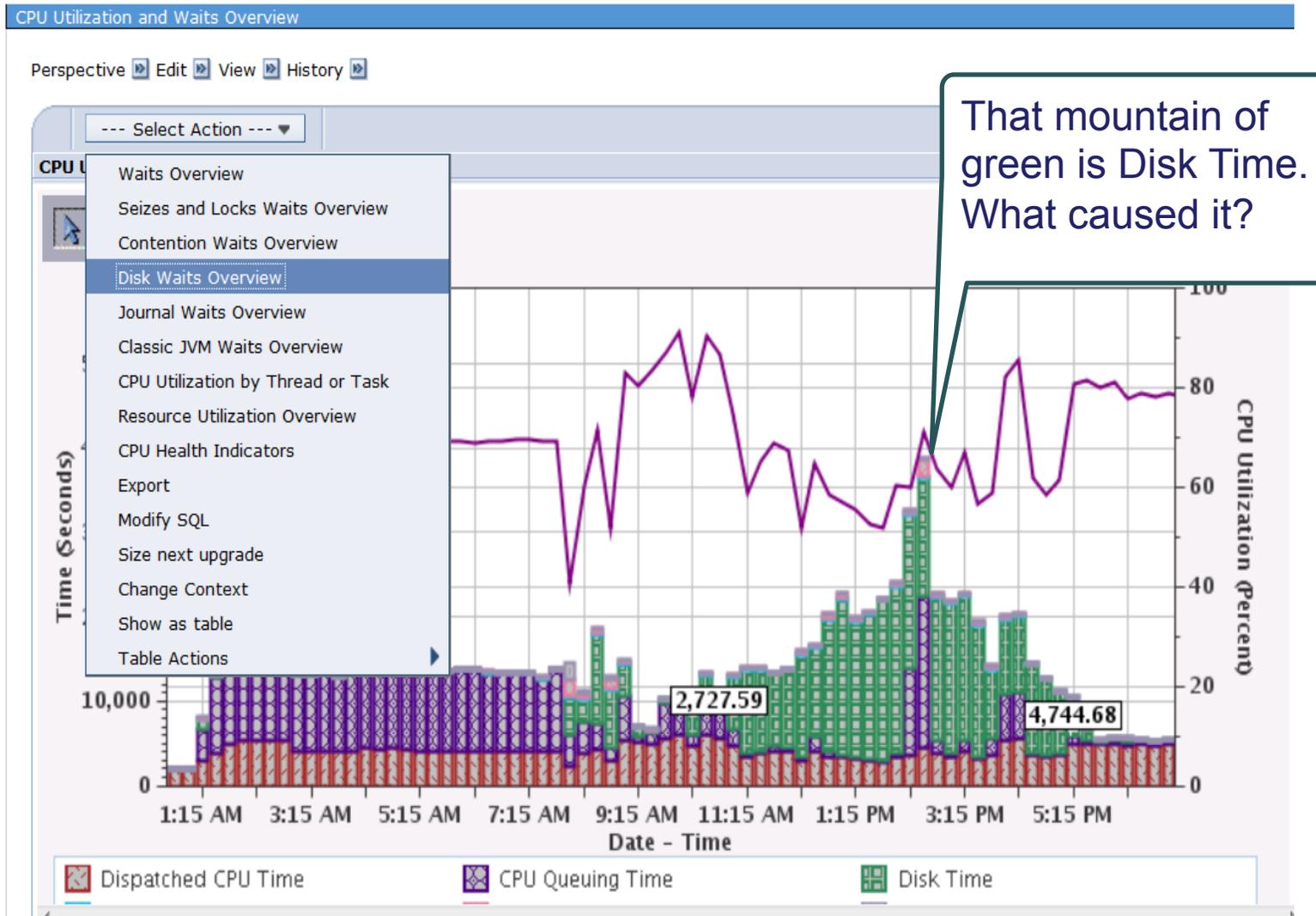
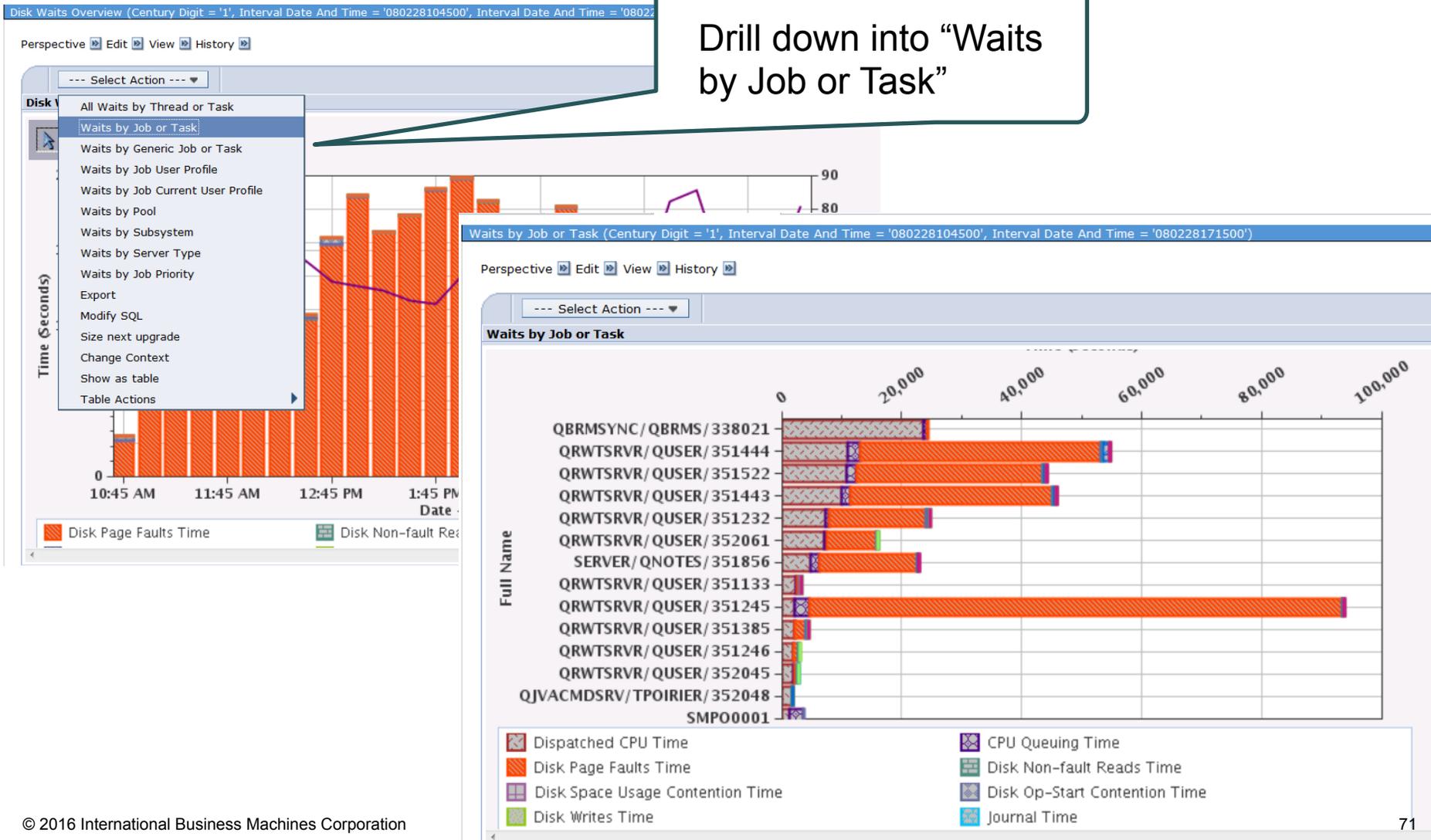http://www.ibmsystemsmag.com/Blogs/i-Can/Archive/i-can-automate-monitoring-with-watches/

Examples

What is causing disk wait time?

Let's Look at the Disk Waits

We see it's faulting…. let's find out who did it

Drill down into "Waits by Job or Task"

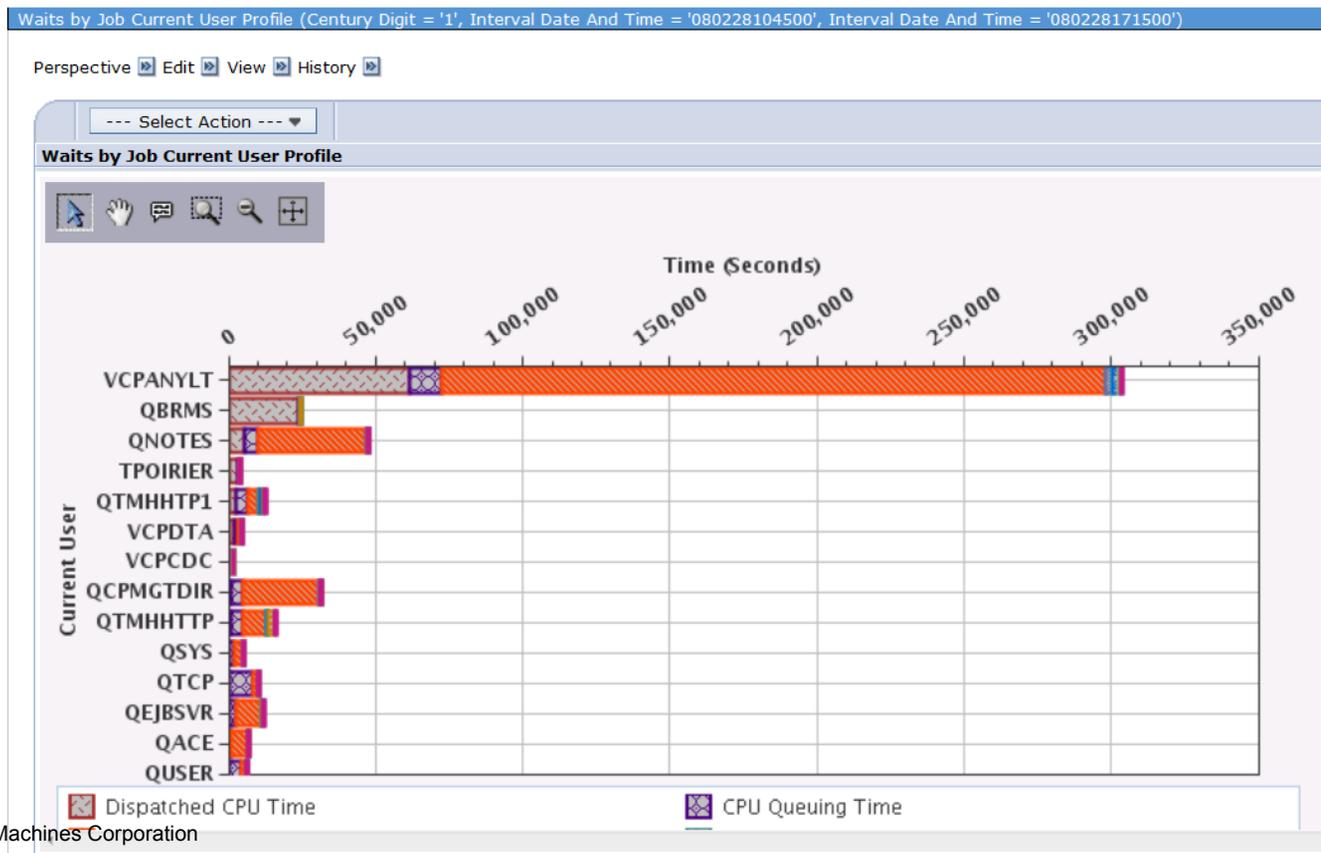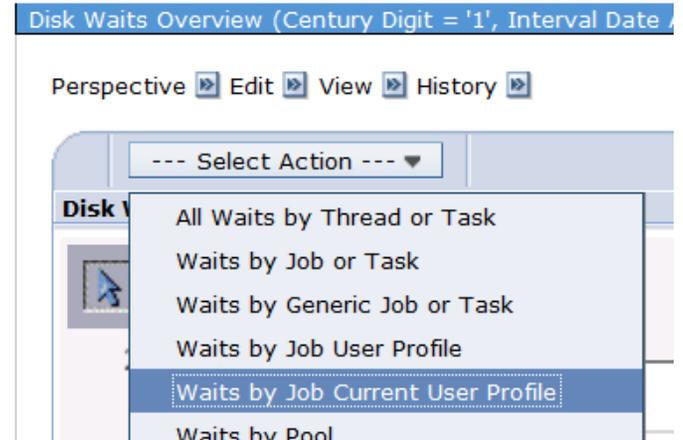# Let's find out who the user is

We now have several clues:
We know the jobs - QRWTSRVR - **DDM/DRDA server jobs**
We know the time - early afternoon
We know the user profile - **QUSER**

But QUSER isn't helpful. We need the job's **current user profile**

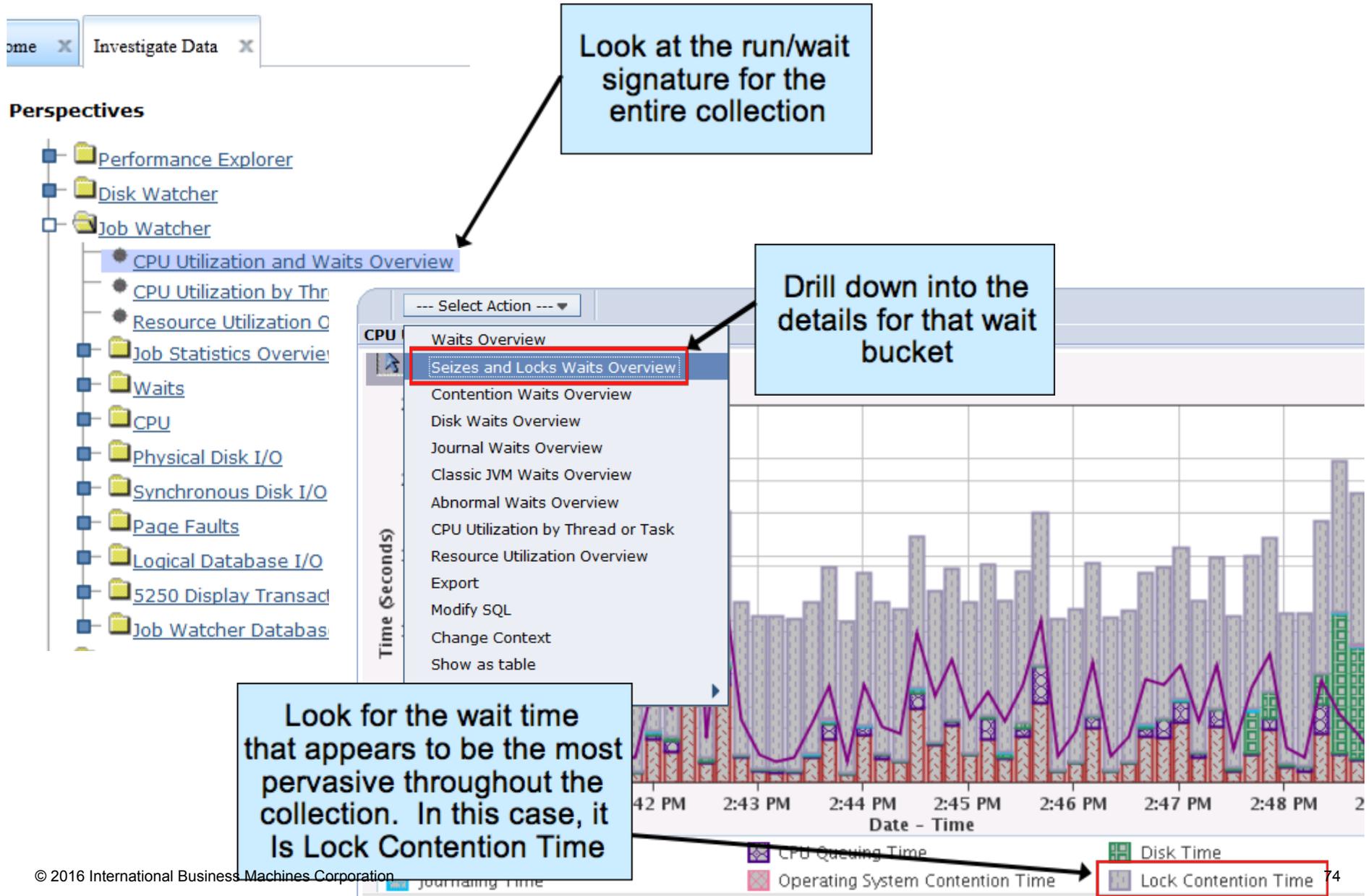**Waits by Job Current User Profile** shows us **VCPANYLT** is the guilty party
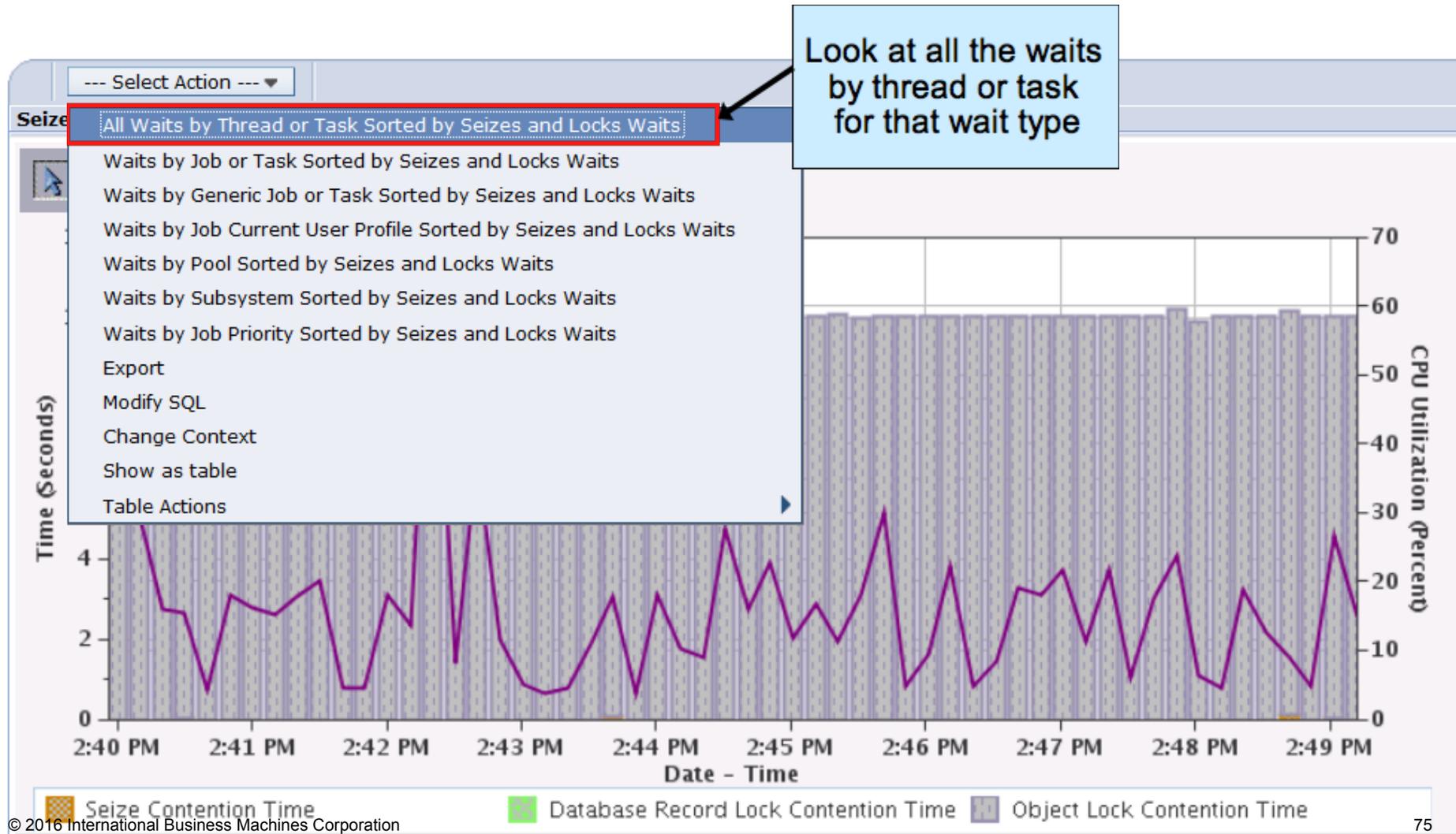
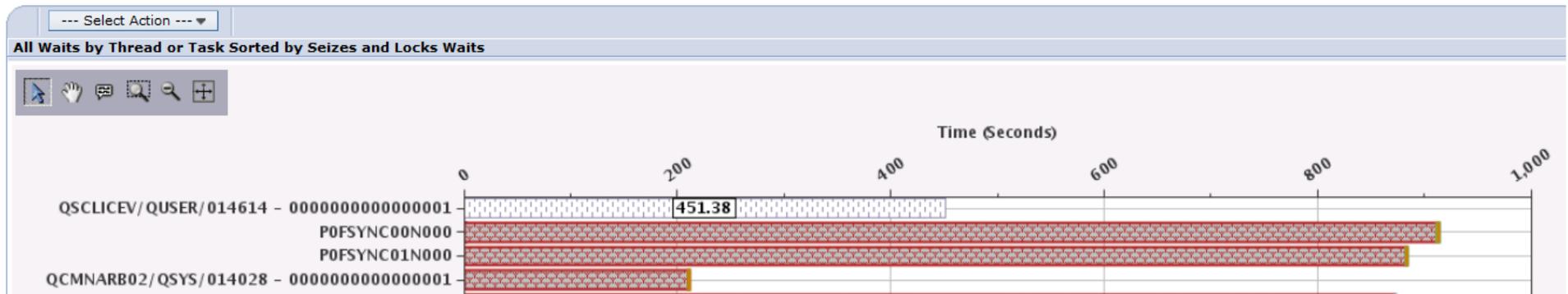# Viewing Waits with Job Watcher

Example of Object Lock Contention

Job Watcher: CPU Utilization and Waits Overview

IBM

Look at the run/wait signature for the entire collection

Perspectives

- Performance Explorer
- Disk Watcher
- Job Watcher
  - CPU Utilization and Waits Overview
  - CPU Utilization by Thr
  - Resource Utilization O
  - Job Statistics Overview
  - Waits
  - CPU
  - Physical Disk I/O
  - Synchronous Disk I/O
  - Page Faults
  - Logical Database I/O
  - 5250 Display Transact
  - Job Watcher Databas

--- Select Action --- ▼

CPU

- Waits Overview
- Seizes and Locks Waits Overview
- Contention Waits Overview
- Disk Waits Overview
- Journal Waits Overview
- Classic JVM Waits Overview
- Abnormal Waits Overview
- CPU Utilization by Thread or Task
- Resource Utilization Overview
- Export
- Modify SQL
- Change Context
- Show as table

Drill down into the details for that wait bucket

Look for the wait time that appears to be the most pervasive throughout the collection. In this case, it Is Lock Contention Time

Time (Seconds)

42 PM   2:43 PM   2:44 PM   2:45 PM   2:46 PM   2:47 PM   2:48 PM   2

Date – Time

- CPU Queuing Time
- Operating System Contention Time
- Disk Time
- Lock Contention Time

74

Seizes and Lock Waits Overview →All Waits by Thread or Task…

Look at all the waits
by thread or task
for that wait type

--- Select Action --- ▼

Seize

All Waits by Thread or Task Sorted by Seizes and Locks Waits

Waits by Job or Task Sorted by Seizes and Locks Waits

Waits by Generic Job or Task Sorted by Seizes and Locks Waits

Waits by Job Current User Profile Sorted by Seizes and Locks Waits

Waits by Pool Sorted by Seizes and Locks Waits

Waits by Subsystem Sorted by Seizes and Locks Waits

Waits by Job Priority Sorted by Seizes and Locks Waits

Export

Modify SQL

Change Context

Show as table

Table Actions ▶

70 — 60 — 50 — 40 — 30 — 20 — 10 — 0

CPU Utilization (Percent)

Time (Seconds) 4   2   0

2:40 PM   2:41 PM   2:42 PM   2:43 PM   2:44 PM   2:45 PM   2:46 PM   2:47 PM   2:48 PM   2:49 PM

Date - Time

Seize Contention Time   Database Record Lock Contention Time   Object Lock Contention Time

All Waits by Thread or Task →
All Waits for One Thread or Task

--- Select Action --- ▼

All Waits by Thread or Task Sorted by Seizes and Locks Waits

Time (Seconds)

| | 0 | 200 | 400 | 600 | 800 | 1,000 |

QSCLICEV/QUSER/014614 – 0000000000000001 — 451.38
P0FSYNC00N000
P0FSYNC01N000
QCMNARB02/QSYS/014028 – 0000000000000001 —

> Select the job with the object lock contention time. Look at all waits for that one thread or task

--- Select Action --- ▼

All W

| All Waits for One Thread or Task |
| All Waits by Thread or Task Sorted by Seize Contention |
| All Waits by Thread or Task Sorted by Database Record Lock Contention |
| All Waits by Thread or Task Sorted by Object Lock Contention |
| Export |
| Modify SQL |
| Change Context |
| Show as table |
| Table Actions ▶ |

200

451.38

QCMNARB02/QSYS/014028 – 0000000000000001 —

All Waits by Thread or Task → Interval Details

Display the interval details for that thread or task

--- Select Action --- ▼

All W...

Interval Details for One Thread or Task

Waits Overview

Export

Modify SQL

Change Context

Show as table

Table Actions ▶

Select an interval where the wait is displayed by clicking on it

Time (Seconds)

8

6

4

10.03

Interval Details

Interval Details for One Thread or Task (Interval Number = '47', Initial Thread Task Count = '2545')

Perspective ▣ Edit ▣ View ▣ History ▣

Collection
Name(s): JWOBJLOCKC
Library: COMMON2
Type: Job Watcher File Based Collection
File level: 3

Time
Start: Dec 13, 2007 2:40:08 PM
End: Dec 13, 2007 2:55:21 PM

System
Name: ISZ1LP13
Relea

Thread or Task Details

Job information:	QSCLICEV/QUSER/014614 - 0000000000000001	Interval:
Current user profile:	QSYS	Priority:
Object waited on:	WATCHEVENTSPACE	Pool:
Holding job or task:	QZRCSRVS/QUSER/014097	Interval timestamp:

The information about the object waited on and who is holding the lock to that object can be found here. The call stack is below. The call stack can give an idea of where to look to find the root cause of the problem. Very powerful!!

[Show Holder]

Call Stack

--- Select Action --- ▼

Call Level	Program	Module	Procedure	
1			qutde_block_trace	
2			longWaitReceive__9QuCounterFR12RmprReceiverPvQ2_8TDQSEnum4EnumUlCPv	
3			lockConflict__17RmslHHTEntryBlockFP11RmslPlmpLKIP12RmslLKIEntryPvP6TDTaskR14RmslRcvrHolderQ2_8TDQSEn	
4			obtainHold__17RmslHoldHashTableFR11RmslPlmpLKIR12RmslLKIEntryP6TDTaskR19RmslBlockDescriptorR14RmslRcv	
5			rmslHLock__FR11RmslPlmpLKI	
6			rmsllockspace__FR8SpacePtrXT17GenericMiTemplate_RUci	

7.2 …More Information!

Interval Details for One Thread or Task (Interval = '97', Initial Thread Task Count = '3959433')

Perspective ⏩ Edit ⏩ View ⏩

Collection   **Time**   **System**

Name(s): DAWNJW2   Start: Mar 12, 2008 8:42:26 AM   Name: RCHASTND

Library: COMMON72   End: Mar 12, 2008 9:42:33 AM   Release: V6R1M0

Type: Job Watcher File Based Collection

File level: 8

Thread or Task Details

Job information:	QSQSRVR/QUSER/437123 – 000000000000013	Priority	50
Current user profile:	VCPANYLT	Pool:	2
Object waited on:	JUDGE JUDGE	Type description:	DB2 ACCESS PATH
Wait duration:	10 milliseconds	Segment type description:	MACHINE INDEX RADIX4 SECONDARY
Current or last wait:	161/SFt	Wait object library:	None detected this interval
Holding job or task:	None detected this interval	Interval timestamp:	Mar 12, 2008 8:51:05 AM
SQL client job:	None detected this interval	Interval (1 to 710):	< 97 >

[Show Holder]

More "Thread or Task Details" details

Easily navigate from one interval to the next

More PDI Examples

Power Systems

Java Perspectives in Collection Services

Java
- IBM Technology for Java Memory Overview
- IBM Technology for Java Memory by Job
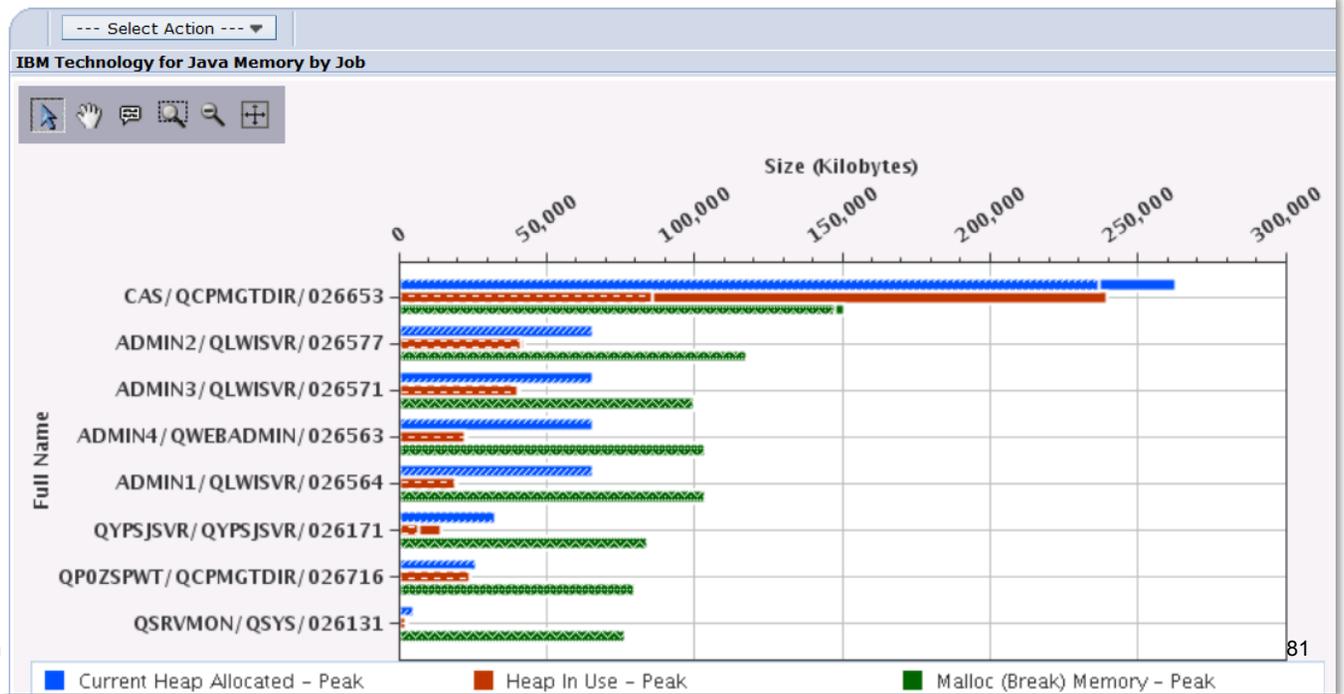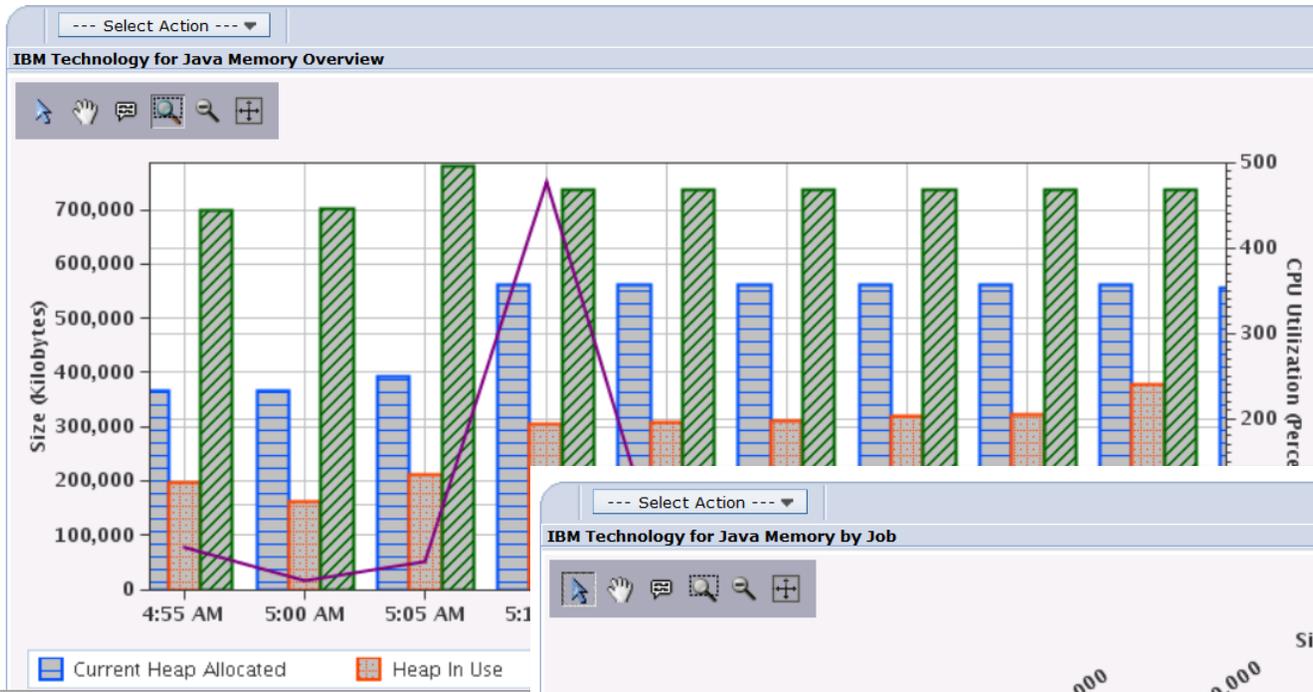
Find that job using a lot of heap…

81

Java Perspectives

Drilldown for one job -
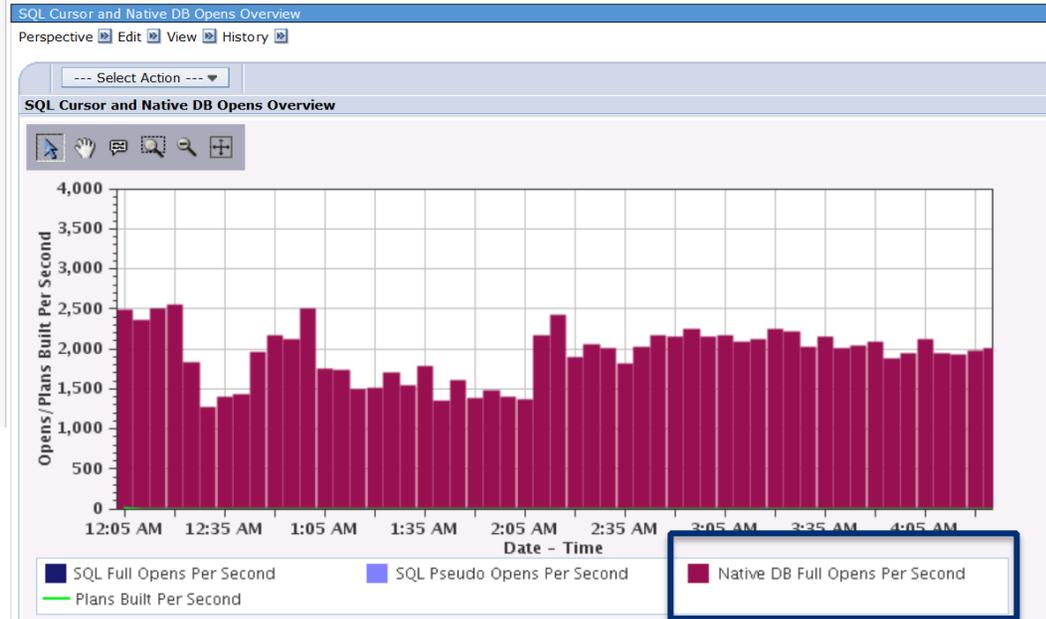Look at the heap and memory usage over time for one selected job.
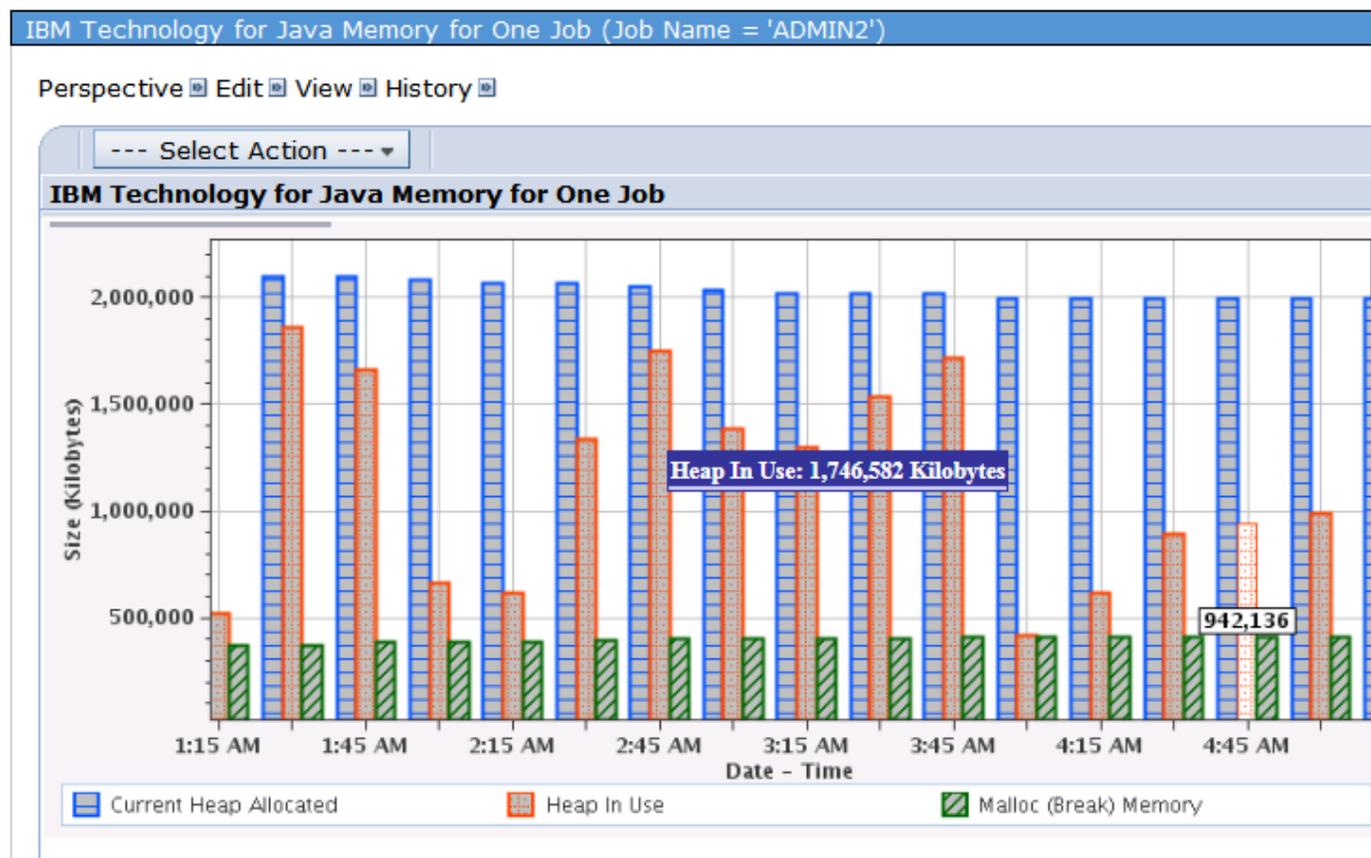
IBM Technology for Java Memory for One Job

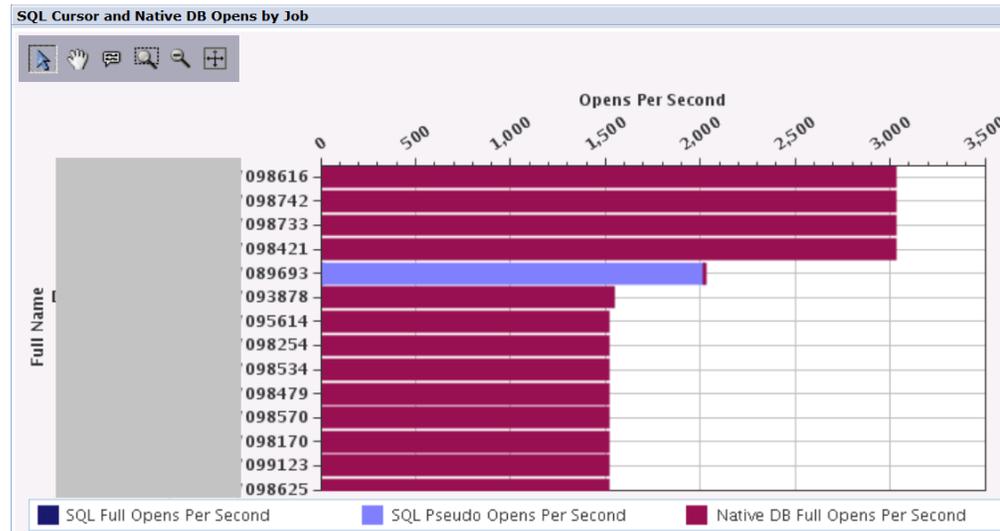# Database Full Opens

Full Opens are expensive resource-wise

⭐ General recommendation is to keep Native Full Opens per second < 1000

Next, find jobs
doing full opens…

Database Full Opens

In an RPG program, full opens are caused by the use of SETON *LR instead of RETRN. Avoid LR if possible. Not setting on LR (in OPM programs) will keep the program in memory, keeps file open and pointers set, retains variable values, etc. Also, avoid *NEW for the ACTGRP

Shared file opens are far less expensive than full file opens. They consume less CPU, less storage and are faster than full opens. There are some implications of changing full file opens to shared file opens, but implementation of shared opens is typically easier to manage and implement than changing SETON LR to RETRN in RPG programs. The following links provide explanation, usage and considerations of using shared ODPs:

Sharing database files in the same job or activation group http://pic.dhe.ibm.com/infocenter/iseries/v7r1m0/topic/dbp/rbafosfile.htm?resultof=%22%4f%44%50%22%20%22%6f%64%70%22%20

Open considerations for files shared in a job or an activation group
http://pic.dhe.ibm.com/infocenter/iseries/v7r1m0/topic/dbp/rbafoopenc.htm

Input/output considerations for files shared in a job or an activation group
http://pic.dhe.ibm.com/infocenter/iseries/v7r1m0/topic/dbp/rbafoiocon.htm

Close considerations for files shared in a job or an activation group
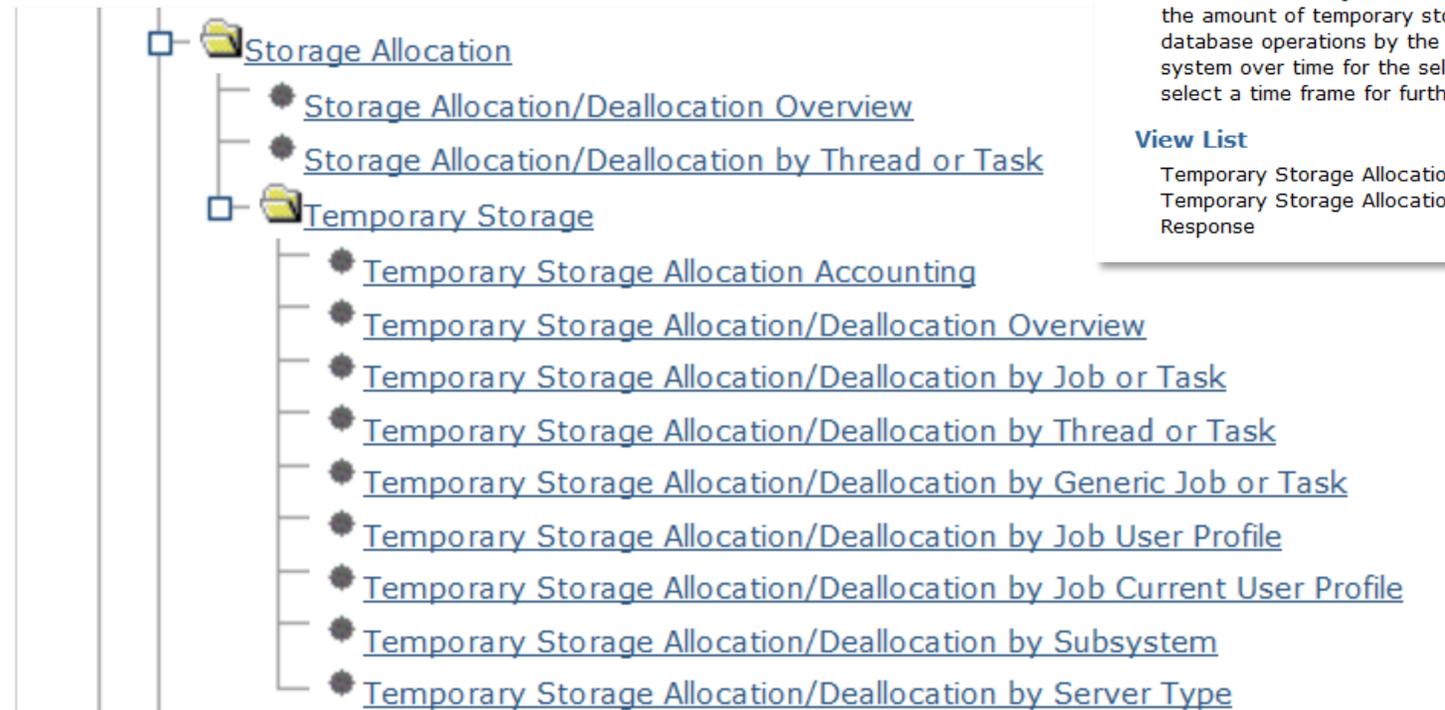http://pic.dhe.ibm.com/infocenter/iseries/v7r1m0/topic/dbp/rbafoclose.htm

Temporary Storage Allocation/Deallocation perspectives
Storage Allocation Perspectives

Where is my temporary storage going?

Expand Collection Services
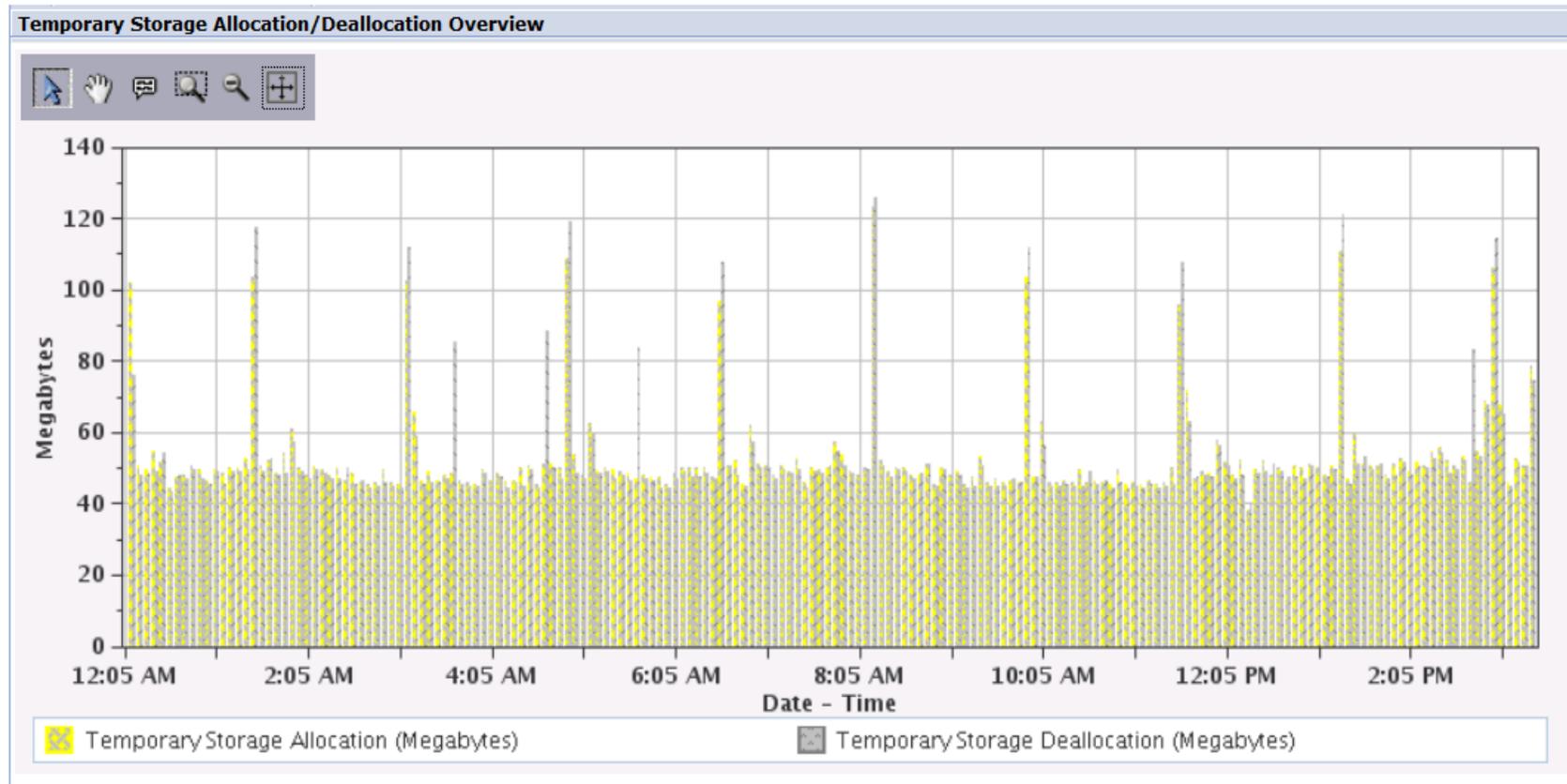
Selection

Name

Temporary Storage Allocation Accounting

Description

This chart shows the amount of temporary storage charged to active and ended jobs, the amount of user temporary storage, and the amount of temporary storage used for database and non database operations by the IBM i operating system across the system over time for the selected collections. Use this chart to select a time frame for further detailed investigation.
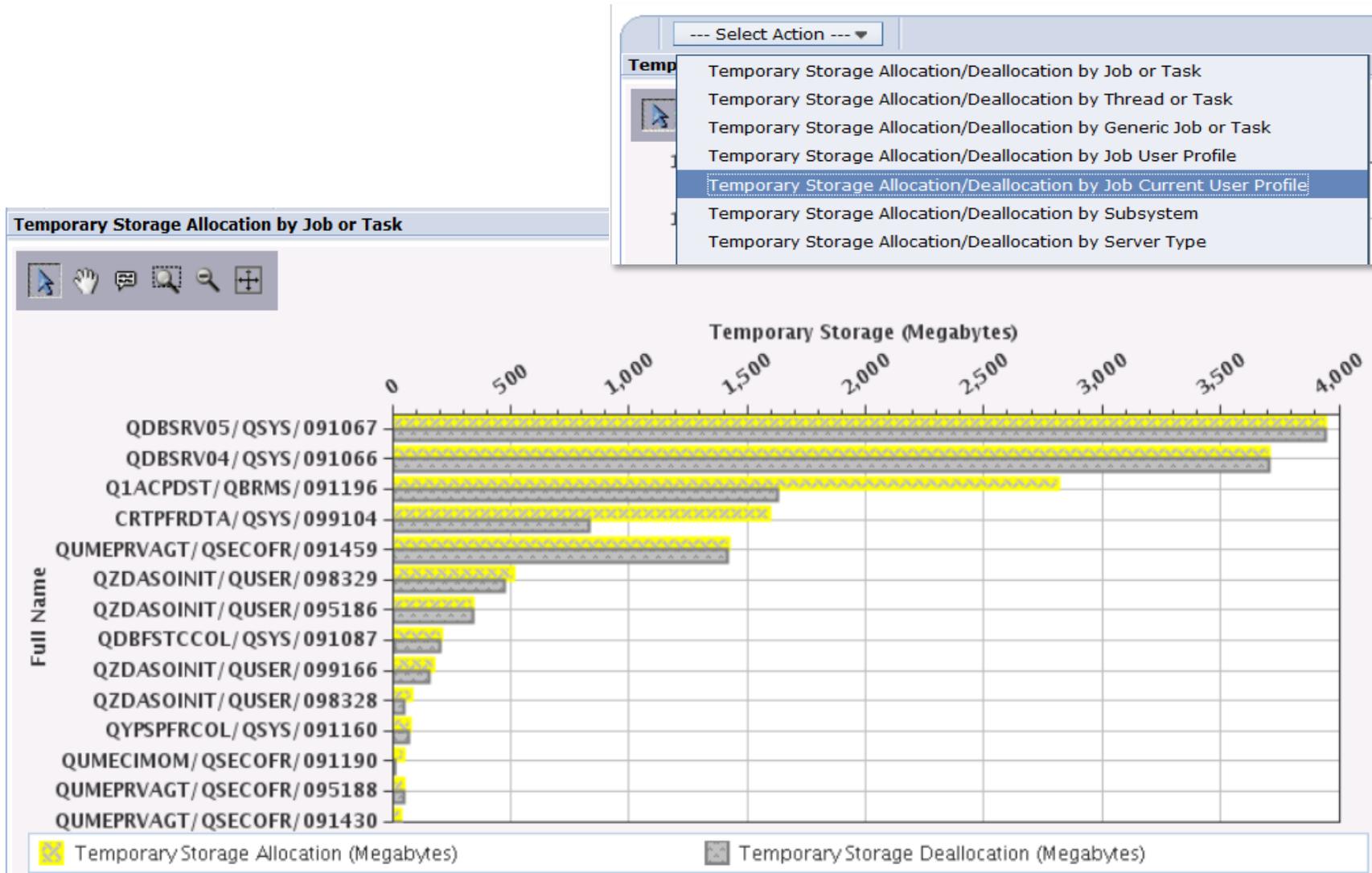
View List

Temporary Storage Allocation Accounting and SQL Statements
Temporary Storage Allocation Accounting and Disk Average Response

- Storage Allocation
  - Storage Allocation/Deallocation Overview
  - Storage Allocation/Deallocation by Thread or Task
- Temporary Storage
  - Temporary Storage Allocation Accounting
  - Temporary Storage Allocation/Deallocation Overview
  - Temporary Storage Allocation/Deallocation by Job or Task
  - Temporary Storage Allocation/Deallocation by Thread or Task
  - Temporary Storage Allocation/Deallocation by Generic Job or Task
  - Temporary Storage Allocation/Deallocation by Job User Profile
  - Temporary Storage Allocation/Deallocation by Job Current User Profile
  - Temporary Storage Allocation/Deallocation by Subsystem
  - Temporary Storage Allocation/Deallocation by Server Type

Temporary Storage Allocation / Deallocation Overview

Generally, allocations and deallocations following a similar pattern

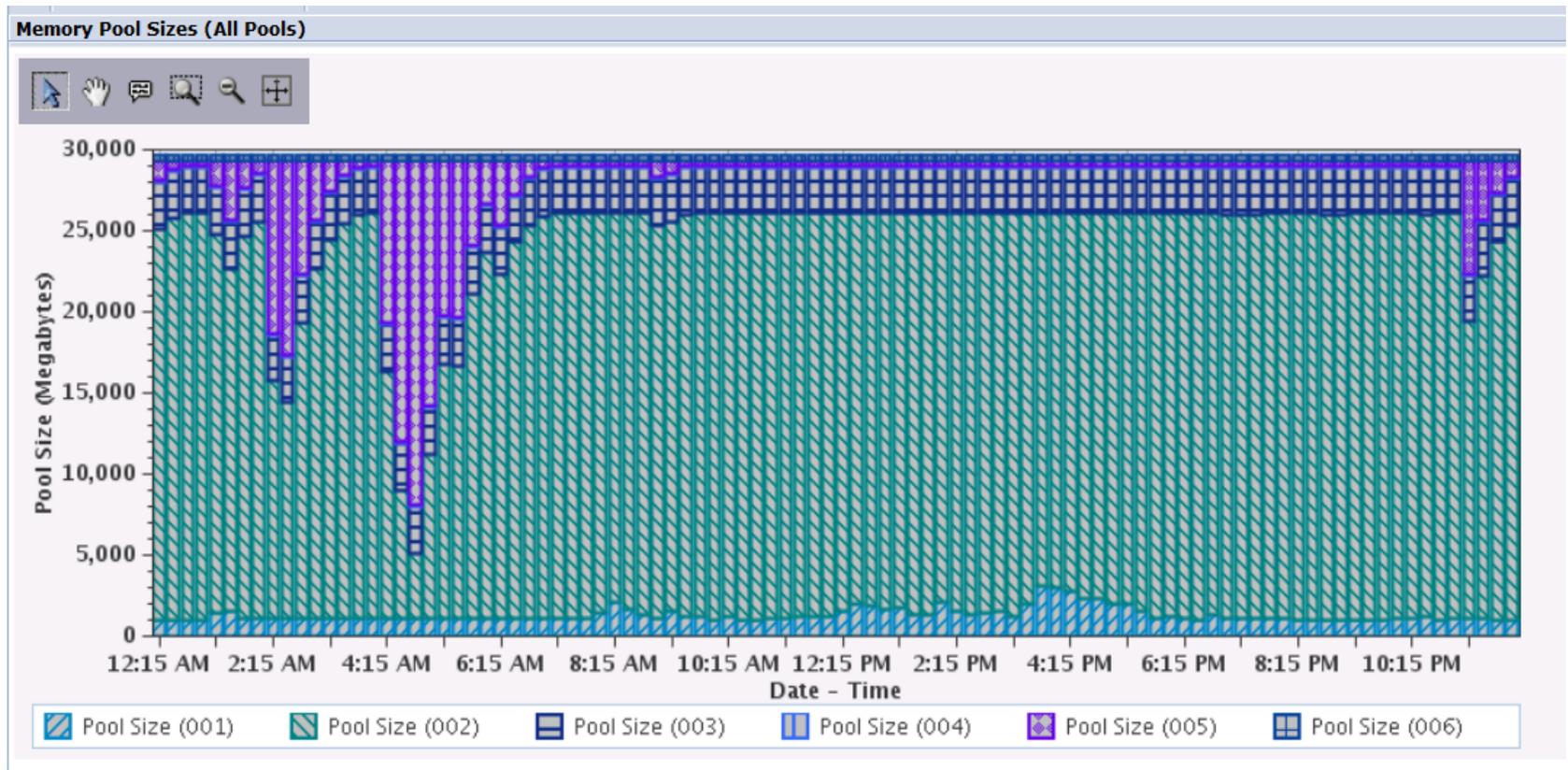# From an overview perspective, drill down to more detail

--- Select Action --- ▼

Temp

| Temporary Storage Allocation/Deallocation by Job or Task |
| Temporary Storage Allocation/Deallocation by Thread or Task |
| Temporary Storage Allocation/Deallocation by Generic Job or Task |
| Temporary Storage Allocation/Deallocation by Job User Profile |
| Temporary Storage Allocation/Deallocation by Job Current User Profile |
| Temporary Storage Allocation/Deallocation by Subsystem |
| Temporary Storage Allocation/Deallocation by Server Type |

Temporary Storage Allocation by Job or Task

Temporary Storage (Megabytes)

0   500   1,000   1,500   2,000   2,500   3,000   3,500   4,000

Full Name:
QDBSRV05/QSYS/091067
QDBSRV04/QSYS/091066
Q1ACPDST/QBRMS/091196
CRTPFRDTA/QSYS/099104
QUMEPRVAGT/QSECOFR/091459
QZDASOINIT/QUSER/098329
QZDASOINIT/QUSER/095186
QDBFSTCCOL/QSYS/091087
QZDASOINIT/QUSER/099166
QZDASOINIT/QUSER/098328
QYPSPFRCOL/QSYS/091160
QUMECIMOM/QSECOFR/091190
QUMEPRVAGT/QSECOFR/095188
QUMEPRVAGT/QSECOFR/091430

⬚ Temporary Storage Allocation (Megabytes)   ⬚ Temporary Storage Deallocation (Megabytes)

What has the performance adjuster been doing to my pools?

- Collection Services allows you to look *backward in time*

Memory
- Memory Pool Sizes and Fault Rates
- Memory Pool Activity Levels
- DB and Non-DB Page Faults

What does the faulting look like when I was testing?

IBM iDoctor for i

- Product developed by the IBM Rochester Support Center for deep, detailed performance analysis

- Major components
  - Job Watcher
    - Job Watcher
    - Collection Services Investigator
    - Disk Watcher
    - Plan Cache Analyzer - graphical analysis of the system's SQL Plan Cache
  - PEX Analyzer
  - ~~Heap Analyzer – Classic JVM heap analysis~~
  - VIOS Investigator

- http://www-912.ibm.com/i_dir/idoctor.nsf
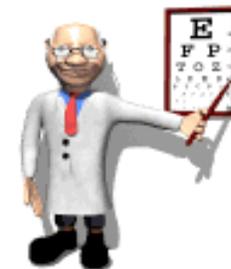
IBM iDoctor for i
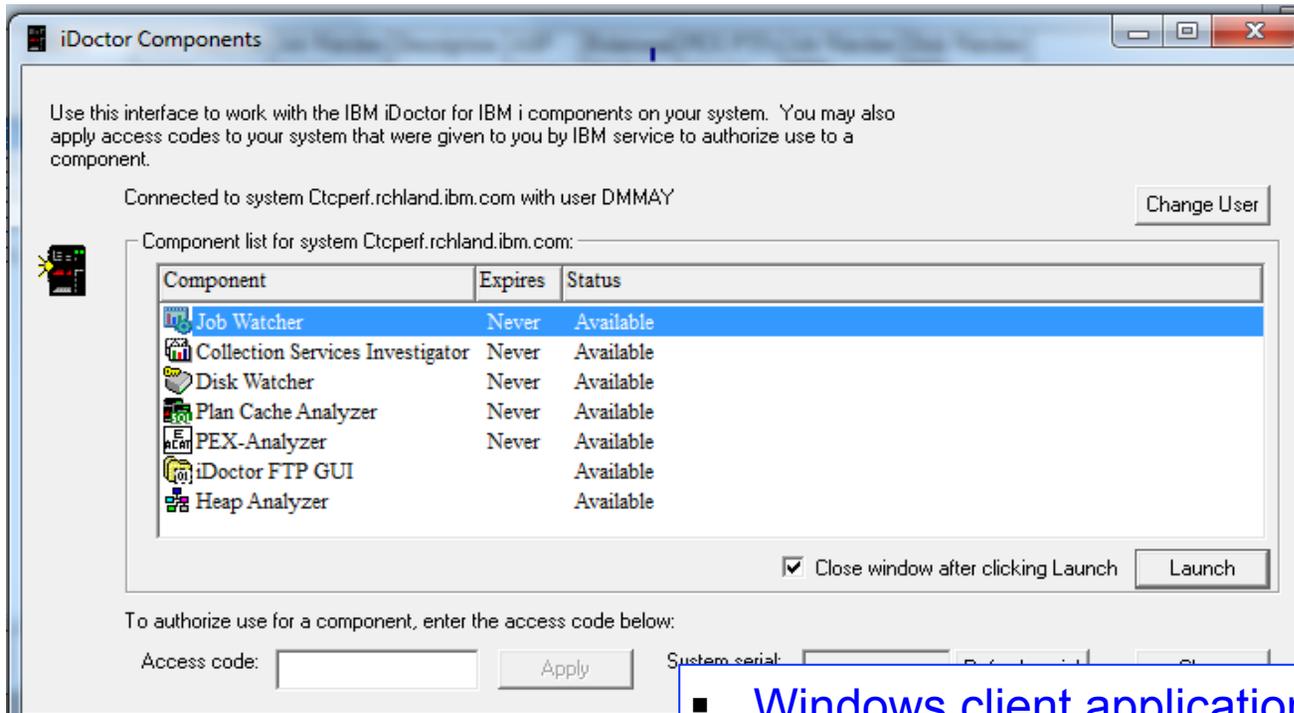
- **Windows client application**
  https://www-912.ibm.com/i_dir/idoctor.nsf

- **Detailed performance analysis and diagnostics**

iDoctor versus Performance Data Investigator

- You have two graphical interfaces for performance data analysis…
    - Which should you use?   It depends….

Feature	iDoctor	PDI
Interface	Windows client	Browser
Wait Analysis	Yes	Yes
Collection Services	Yes	Yes
Job Watcher	Yes	Yes
Disk Watcher	Yes	Yes
Performance Explorer	Yes	Profile collections only
Database	Yes	Yes
Job Watcher Monitors	Yes	No
Customizable	Yes	Yes
User Defined graphs and queries	Yes	Yes
Update Frequency	Monthly	Twice Yearly
Support	Defect only	Standard SWMA
Chargeable	Yearly license	·Collection Services at no additional charge with i ·Disk Watcher, Database, and Performance Explorer included with base PT1 product ·Job Watcher is an additional option of PT1 and has an additional charge
Experimental Features	Yes (e.g., VIOS Investigator)	No
Multinational language support	No	Yes

developerWorks®

Leverage the latest enhancements
The IBM i operating system and related software products are frequently enhanced. Visit this wiki to learn more.

Updates Info IBM i 7.2
TR
Technology Refresh
IBM i 7.1 Enhancements

[Access technology updates]

IBM i Technology Updates - by IBM i product or subject matter

Backup Recovery and Media Services (BRMS)
Collaboration and Social for i (Lotus)
DB2 for i (Database)
General IBM i operating system
Hardware and Firmware (including Technology Refresh content)
IBM i Access Client Solutions
IBM Integrated Web Services for i
Integration with BladeCenter and System x
Java on IBM i
Navigator
Performance Tools
PowerHA SystemMirror for i
Systems Director for i
Web Integration on i

IBM

Performance Tools

☺ Like | Updated today at 12:17 PM by ShaunaRollings | Tags: performance_tools

[Page Actions ▾]

This section contains information about the most recent enhancements to IBM i Performance Tools. This topic includes Performance data collection tools, the performance components of IBM Navigator for i and the Performance Tools LPP (licensed program 57xxPT1)

Performance Data Collectors

Collection Services, Disk Watcher, Job Watcher, and Performance Explorer are the primary pe
IBM i. Other performance related tools include: Batch Model, Work with System Activity (WRKS
(DMPMEMINF), and Analyze Command Performance (ANZCMDPFR).

Performance on the Web

The Performance components of IBM Navigator for i include the **Investigate Data** task which
Investigator (PDI) and the **Manage Collections** task used to manage performance collecti
web-based GUI interface for Collection Services, Job Watcher and Disk Watcher.

Performance Tools LPP (57xxPT1)

Performance Tools is a licensed program product that contains additional performance tools.
Tools Reports. More information on this licensed program is contained in the IBM Knowledge (

Performance on the web

☺ Like | Updated 2/12/15 by lrpowell | Tags: collection_manager, pdi, performance_data_investigator

[Page Actions ▾]

Performance Tools GUI:

The performance components in **IBM Navigator for i** include **Performance Data Investigator (PDI)**, Performance **Collection Manager** and web-based GUI interfaces for Collection Services, Job Watcher and Disk Watcher.

Getting Started:

The main page for Performance Tools and this sub-page "Performance on the web" provide enhancement information. For specific enhancement by topic, see Enhancements and New Perspectives.

The Resources sub-page contains a significant resource list. A good place to start for learning PDI is to document titled "Getting started with the Performance Data Investigator".

PTFs:

PTFs for these functions are part of the set of PTFs for IBM Navigator for i. They are listed in the table below, grouped by date of release. Check against the PTFs listed for IBM Navigator for i.

Performance Task Enhancements (Release Date)	7.2 PTFs	7.1 PTFs - 5770SS1	6.1 PTFs	PTF Description	Notes
Fall 2014 (Dec 2014)	SF99713 level 5 or higher - not yet available *Use PTFs (SP2):* • SI53771 • SI53772	SF99368 level 31 or higher - not yet available *Use PTFs (SP11):* • SI53777 • SI53776)	SF99115 level 42 - not yet available *Use PTFs (SP15):* • SI53773 • SI53774 • SI53775)	HTTP Group PTF *(includes but not limited to:* • Common PTF • Navigator for i • IBM i Navigator tasks on the Web PTF (this PTF is only needed on 6.1)	The Navigator for i PTFs are shipped in the HTTP group, and it is recommended that you keep current on this PTF group as well as: 1. Java group PTF 2. Database Group PTF F 3. Performance Tools Group PTF

93

IBM i Performance FAQ
a MUST read!

http://www.ibm.com/common/ssi/cgi-bin/ssialias?
subtype=WH&infotype=SA&appname=STGE_PO_PO_USEN&htmlfid=POW03102USEN&attachment=POW03102USEN.PDF
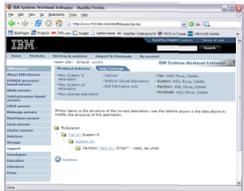
IBM **Power Systems Performance**
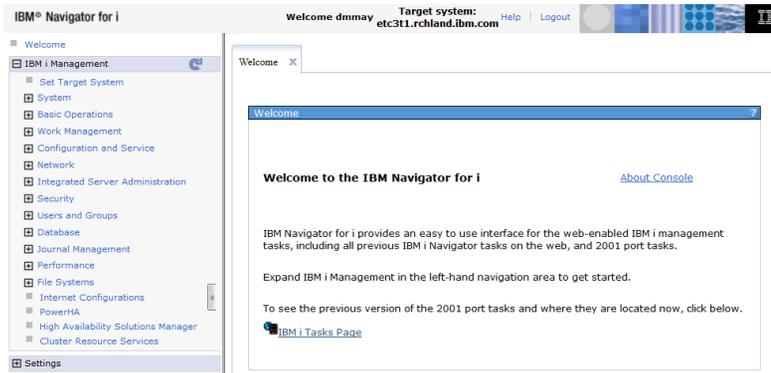
IBM

IBM i on Power - Performance FAQ
October 12, 2015

Now it all makes sense!

PM for Power Systems

WorkLoad Estimator

IBM Navigator for i

Green Screen
- Performance Explorer
- Performance Tools Reports
- System commands

Collection Services
Health Indicators
Monitors
Database
Job Watcher
Disk Watcher
Perf Explorer
Batch Model

| Collection Services |
| Disk Watcher |
| Job Watcher |
| Performance Explorer |

i for Business
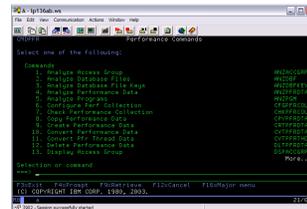
ithankyou

www.ibm.com/power/i

References

Updates **IBM i 7.2**
TR
Technology Refresh
IBM i 7.1 Enhancements

Leverage the latest enhancements

The IBM i operating system and related software products are frequently enhanced. Visit this wiki to learn more.

Access technology updates

IBM i Technology Updates - by IBM i product or subject matter

Backup Recovery and Media Services (BRMS)
Collaboration and Social for i (Lotus)
DB2 for i (Database)
General IBM i operating system
Hardware and Firmware (including Technology Refresh content)
IBM i Access Client Solutions
IBM Integrated Web Services for i
Integration with BladeCenter and System x
Java on IBM i
Navigator
Performance Tools
PowerHA SystemMirror for i
Systems Director for i
Web Integration on i

Performance Tools

Like | Updated today at 12:17 PM by ShaunaRollings | Tags: performance_tools

Page Actions ▾

This section contains information about the most recent enhancements to IBM i Performance Tools. This topic includes Performance data collection tools, the performance components of IBM Navigator for i and the Performance Tools LPP (licensed program 57xxPT1).

Performance Data Collectors

Collection Services, Disk Watcher, Job Watcher, and Performance Explorer are the primary pe
IBM i. Other performance related tools include: Batch Model, Work with System Activity (WRKS
(DMPMEMINF), and Analyze Command Performance (ANZCMDPFR).

Performance on the Web

The Performance components of IBM Navigator for i include the **Investigate Data** task which
Investigator (PDI) and the **Manage Collections** task used to manage performance collecti
web-based GUI interface for Collection Services, Job Watcher and Disk Watcher.

Performance Tools LPP (57xxPT1)

Performance Tools is a licensed program product that contains additional performance tools.
Tools Reports. More information on this licensed program is contained in the IBM Knowledge (

Performance on the web

Like | Updated 2/12/15 by lrpowell | Tags: collection_manager, pdi, performance_data_investigator

Page Actions ▾

Performance Tools GUI:

The performance components in **IBM Navigator for i** include **Performance Data Investigator (PDI)**, Performance **Collection Manager** and web-based GUI interfaces for Collection Services, Job Watcher and Disk Watcher.

Getting Started:

The main page for Performance Tools and this sub-page "Performance on the web" provide enhancement information. For specific enhancement by topic, see Enhancements and New Perspectives.

The Resources sub-page contains a significant resource list. A good place to start for learning PDI is to document titled "Getting started with the Performance Data Investigator".

PTFs:

PTFs for these functions are part of the set of PTFs for IBM Navigator for i. They are listed in the table below, grouped by date of release. Check against the PTFs listed for IBM Navigator for i.

Performance Task Enhancements (Release Date)	7.2 PTFs	7.1 PTFs - 5770SS1	6.1 PTFs	PTF Description	Notes
Fall 2014 (Dec 2014)	SF99713 level 5 or higher - not yet available *Use PTFs (SP2):* • SI53771 • SI53772	SF99368 level 31 or higher - not yet available *Use PTFs (SP11):* • SI53777 • SI53776)	SF99115 level 42 - not yet available *Use PTFs (SP15):* • SI53773 • SI53774 • SI53775)	HTTP Group PTF *(includes but not limited to:* • Common PTF • Navigator for i • IBM i Navigator tasks on the Web PTF (this PTF is only needed on 6.1)	The Navigator for i PTFs are shipped in the HTTP group, and it is recommended that you keep current on this PTF group as well as: 1. Java group PTF 2. Database Group PTF 3. Performance Tools Group PTF

PDI Enhancements via PTFs

1. IBM i developerWorks

2. Technology Updates

3. Performance Tools

4. Performance on the Web

You will find a list of enhancements by timeframe with links to the details.

IBM i Performance on developerWorks

Leverage the latest enhancements
The IBM i operating system and related software products are frequently enhanced. Visit this wiki to learn more.

Access technology updates

- developerWorks

- Performance Tools
    - Additional performance tools resources
    - Performance on the Web
    - Performance Data Collectors

- Forum

- IBM i Performance Data Investigator

- IBM i Performance Data Investigator – Edit Perspectives

- IBM i Wait Accounting

- How to use the Batch Model performance tool

IBM i Web Sites with Performance Information

- IBM Knowledge Center
  - 7.1
  - 7.2

- IBM i Performance Management
  This web site has a lot of GREAT references and papers – see the resources tab

- Performance Management for Power Systems

- IBM Workload Estimator

- iDoctor

- Job Waits Whitepaper

Steve Will
Posts
Bio

You and i

You and i *IBM i Trends and Strategies*

http://www.ibmsystemsmag.com/Blogs/You-and-i/

Dawn May

i Can

i Can *Technical Tips for i*

http://www.ibmsystemsmag.com/Blogs/i-Can/

Mr. Modern-i-zation Rowe

https://www.systemideveloper.com/blogs/?q=blog/6

DB2 for i

DB2 for i

http://db2fori.blogspot.com/

Power Systems

i Can

For a simple list of all blogs on one page -
"i Can" Blog of Blogs

i Can Technical Tips for i

Performance Management on IBM i Web Site

http://www-03.ibm.com/systems/power/software/i/management/performance/index.html

IBM Systems > Power Systems > Software > IBM i > System management >

Performance management on IBM i

| Overview | Tools | Performance Explorer | Resources |

Find what you need

Performance Data Collectors

There are four collectors on IBM i that collect performance related data and store the information in database files, each having their own unique characteristics: Collection Services, IBM i Job Watcher, IBM i Disk Watcher, and Performance Explorer.

Performance Data Investigator (PDI)

Use the Investigate Data task found in the web-based IBM Systems Director Navigator for i to view and analyze the data collected from any of the four data collectors found on IBM i. This powerful tool allows you the ability to work with the data interactively in chart or table form.

iDoctor for IBM i

A family of products (including Job Watcher, PEX Analyzer, and Heap Analysis Tools for Java) focused on assessing the overall health of a system by providing automated analysis on a variety of performance related data.

PM for Power Systems

A tool that can automatically collect system utilization information and can produce regular reports which show the utilization and growth trends of your system.

Performance and Scalability Services

Plan and prepare for changes in the data center when using the IBM i operating system on Power Systems hardware with help from IBM Systems Lab Services and Training. Whether

What Happened to the PCRM?

- Performance Capabilities Reference Manual – "*PCRM*"

- Was *THE* reference manual for all things related to IBM i performance considerations
    - Content was carried forward but not always updated

- Beginning in 2014, the PCRM only covers **CPW information**
    - Updates for new hardware models and CPW ratings
    - Older versions are still available for download

- Use other sources for IBM i performance information:
    - The IBM i Performance FAQ
    - Papers under the *resources* section on the Performance Management site
    - Knowledge Center
    - developerWorks

See Performance Management Resource Library at …

http://www.ibm.com/systems/i/advantages/perfmgmt/resource.html

Performance management on IBM i

| Overview | Tools | Performance Explorer | Resources |

General performance resources

- IBM i 7.2 Performance Capabilities Reference - June 2014 (781.29 KB)
- IBM i 7.2 Performance Capabilities Reference - April 2014 (771.33 KB)
- IBM i 7.1 Performance Capabilities Reference - August 2013 (1.42 MB)
- IBM i 7.1 Performance Capabilities Reference - February 2013(1.42 MB)
- IBM i 7.1 Performance Capabilities Reference - November 2012(1.41 MB)
- IBM i 7.1 Performance Capabilities Reference - October 2012(1.40 MB)
- IBM i 7.1 Performance Capabilities Reference - April 2012(1.37 MB)
- IBM i 7.1 Performance Capabilities Reference - October/April 2011(1.38 MB)
- IBM i 7.1 Performance Capabilities Reference - April 2011(1.53 MB)

Performance education resources

- → IBM Systems Training
- → Performance Analysis and Capacity Planning for IBM i

Performance papers and articles

- IBM i on Power - Performance FAQ (889KB)
- Under the Hood: POWER7 Logical Partitions(2.54 MB)
- Under the Hood: Of POWER7 Processor Caches(412.68 KB)
- POWER7 Virtualization Best Practice Guide(201KB)
- Performance Implications of POWER7 Model 780's TurboCore Mode(670.08 KB)
- Of Numa on POWER7 in IBM i(579.97 KB)
- What's This Multi-Core Computing Really?(19.56 KB)
- Performance Value of Solid State Drives using IBM i(169.56 KB)
- DS5300 Performance Results in IBM i Power Systems Environment(116.91 KB)
- Of GigaHertz and CPWs(44.10 KB)
- LPAR Performance on Power Systems with POWER4, POWER5 and POWER6(191.37 KB)
- Simultaneous Multi-Threading on POWER7 Processors(51.75 KB)
- ↪ Get Microsoft® Word® Viewer

System i WebSphere and Java tuning resources

Contact IBM

- ✉ Email IBM
- → Find a Business Partner
- ☎ Call IBM: 1-866-883-8901
  Priority code: 101AR13W

Browse Power Systems

- Hardware
- Operating systems
- System software
- Solutions
- Migrate to Power
- Advantages

- → Community
- → Success stories
- → News
- → Support & services
- → Resources
- → Education

Sizing tools

Tools and resources for estimating and sizing computing resources needed to run your business.

- → IBM Systems Workload Estimator
- → IBM Systems Energy Estimator
- → IBM Benchmark Center

Additional information

- → BRMS Performance Tuning of Online Domino Backups
- → BRMS TSM Client Performance
- → iSeries Navigator Performance Tips and Techniques
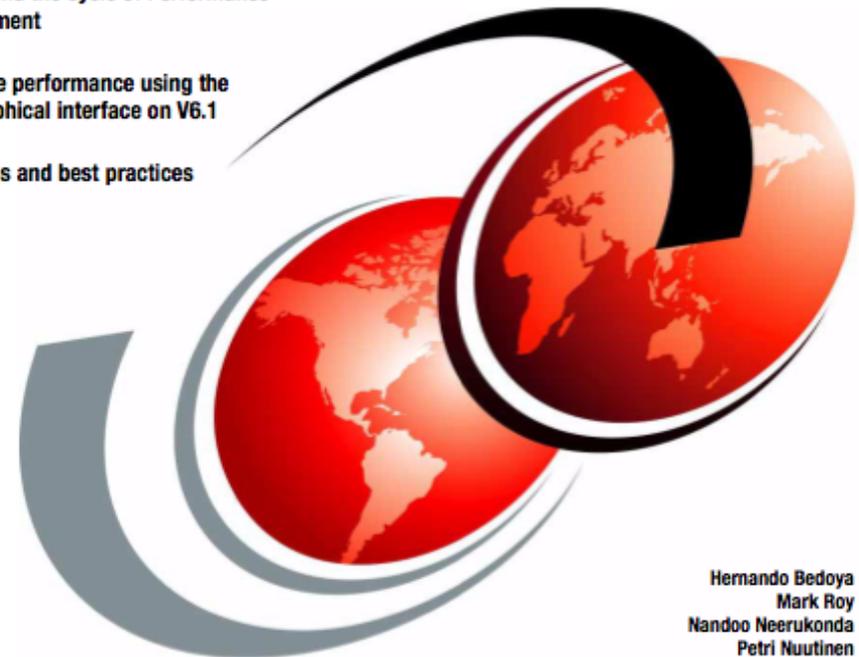
Refer to these

IBM

A **Red**books publication!

End to End Performance Management on IBM i

Understand the cycle of Performance Management

Maximize performance using the new graphical interface on V6.1

Learn tips and best practices

Hernando Bedoya
Mark Roy
Nandoo Neerukonda
Petri Nuutinen

http://www.redbooks.ibm.com/redbooks/pdfs/sg247808.pdf
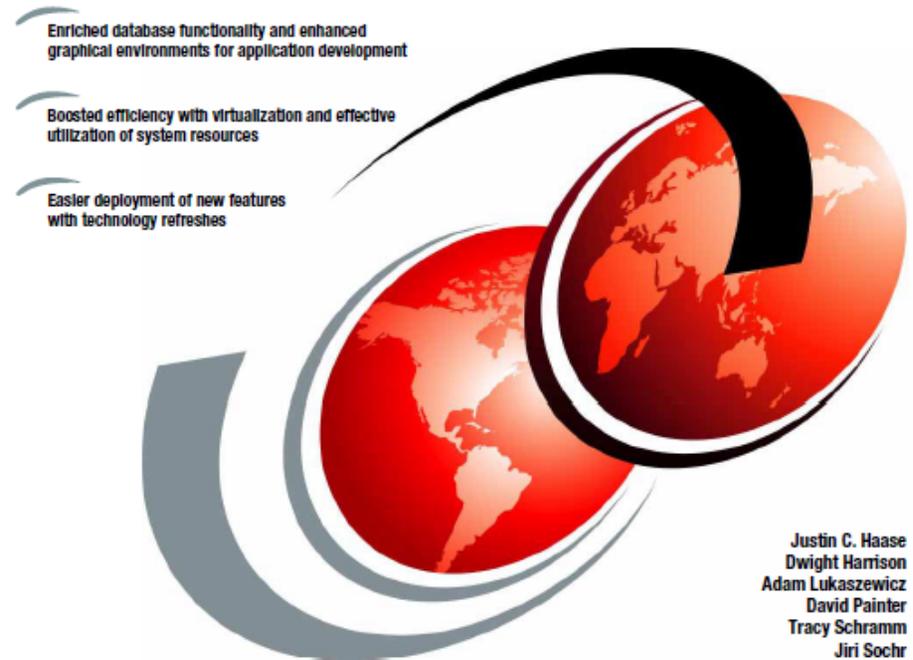
ibm.com/redbooks

Redbooks

IBM i 7.1 Technical Overview with Technology Refresh Updates

Covers the 7.1 content through Technology Refresh 7

Chapter 6 – Performance Tools

Chapter 17, Section 6 – Performance in Navigator for i

IBM i 7.2 Technical Overview with Technology Refresh Updates

Covers the 7.2 content through Technology Refresh 1

Draft Document for Review December 10, 2014 2:51 pm   SG24-8249-00

IBM i 7.2 Technical Overview with Technology Refresh Updates

Section 2.8 – Performance

Section 8.6.7 – Job level SQL stats in Collection Services

Covers new functions and enhancements through IBM i 7.2 TR1

Easy to use web-based system management

Integrated Data-Centric approach

Redbooks and Redpapers on IBM i Performance Tools

- IBM i 7.1 Technical Overview with Technology Refresh Updates
- IBM i 7.2 Technical Overview with Technology Refresh Updates
- Application and Program Performance Analysis Using PEX Statistics
- Best Practices for Managing IBM i Jobs and Output (and a few other special tips)
- i5/OS Diagnostic Tools for System Administrators: An A to Z Reference for Problem Determination

The following redbooks are a bit dated but still have some useful information.

- IBM Systems Director Navigator for IBM i (Chapter 9)
- IBM eServer iSeries Performance Management Tools
- A Systems Management Guide to Performance Management for System i and System p servers
- Sizing IBM i5/OS Work on IBM System i5 Partitions
- Managing OS/400 with Operations Navigator V5R1 Volume 5: Performance Management
- IBM iDoctor iSeries Job Watcher: Advanced Performance Tool
- IBM eServer iSeries Systems Management Handbook

Articles

- IBM Systems Magazine, IBM i – "Power Systems Monitoring With PowerVP", January 2014

- IBM Systems Magazine, IBM i – "Customizing a Perspective in Performance Data Investigator", August 2011

- IBM Systems Magazine, IBM i – "Investigating the Investigator", May 2010

- IBM Systems Magazine, IBM i - "Sky High Performance ", Aug 2009

- SystemiNetwork - "Performance Data Investigator Consolidates Functions in One Place", June 2009

- SystemiNetwork - "IBM Systems Director Navigator for i: Performance Tasks Overview", June 2009

- IBM Systems Magazine, IBM i – "A Command Performance", Nov 2008

- IBM Systems Magazine, IBM i - "Introducing IBM Systems Director Navigator for i5/OS", Aug 2008

- IBM Systems Magazine, IBM i – "A Collective Effort", Nov 2006

- IBM Systems Magazine, IBM i - "Mission: Performance Management", Nov 2006

Articles on Job Watcher

- "Web Power"

- Introduction to Job Watcher Green Screen Commands

- Top 10 Hidden iDoctor Gems

- Using iDoctor for iSeries Job Watcher to Determine Why Jobs Wait

Articles on Disk Performance

- A New Way to Look at Disk Performance
  http://www.ibmsystemsmag.com/ibmi/administrator/performance/A-New-Way-to-Look-at-Disk-Performance/
- Analyzing Disk Watcher Data
  http://www.ibmsystemsmag.com/ibmi/tipstechniques/systemsmanagement/Analyzing-Disk-Watcher-Data/
- Using Wait State Accounting to Determine Disk Performance
  http://iprodeveloper.com/systems-management/using-wait-state-accounting-determine-disk-performance
- Understanding Disk Performance, Part 2: Disk Operation on i5/OS
  http://iprodeveloper.com/systems-management/understanding-disk-performance-part-2-disk-operation-i5os
- Understanding Disk Performance Metrics
  http://iprodeveloper.com/systems-management/understanding-disk-performance-metrics
- Planning for Solid State Drives
  http://ibmsystemsmag.blogs.com/i_can/2012/01/planning-for-solid-state-drives.html
- Moving Data to Solid State Drives
  http://ibmsystemsmag.blogs.com/i_can/2013/03/moving-data-to-solid-state-drives.html
  http://www.ibmsystemsmag.com/ibmi/storage/disk/data_ssd/
- Customer use of SSDs
  http://www-912.ibm.com/s_dir/slkbase.NSF/DocNumber/592252201
- A Look at System i Integrated DASD Configuration and Performance under i5/OS
  - Redpaper REDP-3919-00
  http://www.redbooks.ibm.com/abstracts/redp3919.html

Systems Management References

- Navigator for i on developerWorks
  https://www.ibm.com/developerworks/mydeveloperworks/wikis/home?lang=en#/wiki/IBM%20i%20Technology%20Updates/page/IBM%20Navigator%20for%20i

- IBM Application Runtime Expert
  http://www-03.ibm.com/systems/power/software/i/are/index.html
  http://www.ibm.com/developerworks/ibmi/library/i-applicationruntime/index.html

- Uncovering Application Runtime Expert – IBM i 7.1
  http://www.redbooks.ibm.com/abstracts/redp4805.html?Open

- Web Performance Advisor
  http://www.ibmsystemsmag.com/ibmi/administrator/performance/Web-Performance-Advisor-Helps-Solve-Mysteries/

- IBM Systems Director
  http://www-03.ibm.com/systems/software/director/

  http://pic.dhe.ibm.com/infocenter/director/pubs/index.jsp

- IBM Tivoli Monitoring
  http://www-01.ibm.com/support/knowledgecenter/SSTFXA_6.3.0.2/com.ibm.itm.doc_6.3fp2/welcome.htm?lang=en
- IBM Tivoli Monitoring Agent for IBM i
  IBM Tivoli Monitoring IBM i OS Agent Reference Version 6.3 Fix Pack 2.pdf

Special notices

This document was developed for IBM offerings in the United States as of the date of publication. IBM may not make these offerings available in other countries, and the information is subject to change without notice. Consult your local IBM business contact for information on the IBM offerings available in your area.

Information in this document concerning non-IBM products was obtained from the suppliers of these products or other public sources. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. Send license inquires, in writing, to IBM Director of Licensing, IBM Corporation, New Castle Drive, Armonk, NY 10504-1785 USA.

All statements regarding IBM future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

The information contained in this document has not been submitted to any formal IBM test and is provided "AS IS" with no warranties or guarantees either expressed or implied.

All examples cited or described in this document are presented as illustrations of the manner in which some IBM products can be used and the results that may be achieved. Actual environmental costs and performance characteristics will vary depending on individual client configurations and conditions.

IBM Global Financing offerings are provided through IBM Credit Corporation in the United States and other IBM subsidiaries and divisions worldwide to qualified commercial and government clients. Rates are based on a client's credit rating, financing terms, offering type, equipment type and options, and may vary by country. Other restrictions may apply. Rates and offerings are subject to change, extension or withdrawal without notice.

IBM is not responsible for printing errors in this document that result in pricing or information inaccuracies.

All prices shown are IBM's United States suggested list prices and are subject to change without notice; reseller prices may vary.

IBM hardware products are manufactured from new parts, or new and serviceable used parts. Regardless, our warranty terms apply.

Any performance data contained in this document was determined in a controlled environment. Actual results may vary significantly and are dependent on many factors including system hardware configuration and software design and configuration. Some measurements quoted in this document may have been made on development-level systems. There is no guarantee these measurements will be the same on generally-available systems. Some measurements quoted in this document may have been estimated through extrapolation. Users of this document should verify the applicable data for their specific environment.

Revised September 26, 2006

Special notices (cont.)

IBM, the IBM logo, ibm.com AIX, AIX (logo), AIX 5L, AIX 6 (logo), AS/400, BladeCenter, Blue Gene, ClusterProven, DB2, ESCON, i5/OS, i5/OS (logo), IBM Business Partner (logo), IntelliStation, LoadLeveler, Lotus, Lotus Notes, Notes, Operating System/400, OS/400, PartnerLink, PartnerWorld, PowerPC, pSeries, Rational, RISC System/6000, RS/6000, THINK, Tivoli, Tivoli (logo), Tivoli Management Environment, WebSphere, xSeries, z/OS, zSeries, Active Memory, Balanced Warehouse, CacheFlow, Cool Blue, IBM Systems Director VMControl, pureScale, TurboCore, Chiphopper, Cloudscape, DB2 Universal Database, DS4000, DS6000, DS8000, EnergyScale, Enterprise Workload Manager, General Parallel File System, , GPFS, HACMP, HACMP/6000, HASM, IBM Systems Director Active Energy Manager, iSeries, Micro-Partitioning, POWER, PowerExecutive, PowerVM, PowerVM (logo), PowerHA, Power Architecture, Power Everywhere, Power Family, POWER Hypervisor, Power Systems, Power Systems (logo), Power Systems Software, Power Systems Software (logo), POWER2, POWER3, POWER4, POWER4+, POWER5, POWER5+, POWER6, POWER6+, POWER7, System i, System p, System p5, System Storage, System z, TME 10, Workload Partitions Manager and X-Architecture are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries.

A full list of U.S. trademarks owned by IBM may be found at: http://www.**ibm.com**/legal/copytrade.shtml.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.
AltiVec is a trademark of Freescale Semiconductor, Inc.
AMD Opteron is a trademark of Advanced Micro Devices, Inc.
InfiniBand, InfiniBand Trade Association and the InfiniBand design marks are trademarks and/or service marks of the InfiniBand Trade Association.
Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.
IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency which is now part of the Office of Government Commerce.
Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.
Linear Tape-Open, LTO, the LTO Logo, Ultrium, and the Ultrium logo are trademarks of HP, IBM Corp. and Quantum in the U.S. and other countries.
Linux is a registered trademark of Linus Torvalds in the United States, other countries or both.
Microsoft, Windows and the Windows logo are registered trademarks of Microsoft Corporation in the United States, other countries or both.
NetBench is a registered trademark of Ziff Davis Media in the United States, other countries or both.
SPECint, SPECfp, SPECjbb, SPECweb, SPECjAppServer, SPEC OMP, SPECviewperf, SPECapc, SPEChpc, SPECjvm, SPECmail, SPECimap and SPECsfs are trademarks of the Standard Performance Evaluation Corp (SPEC).
The Power Architecture and Power.org wordmarks and the Power and Power.org logos and related marks are trademarks and service marks licensed by Power.org.
TPC-C and TPC-H are trademarks of the Transaction Performance Processing Council (TPPC).
UNIX is a registered trademark of The Open Group in the United States, other countries or both.

Revised December 2, 2010
Other company, product and service names may be trademarks or service marks of others.