MDM
Performance Must-Gather

This document contains confidential material. It may only be released upon the written consent of IBM and the client.

Table of Contents

Static data:

Hardware

- Overall architecture and components: servers involved (application server, database server), network (Best if shown as a diagram)

For each server:

- System Architecture
  - uname –a
  - vmstat -s
  - vmstat -v
- Disks
  - df –k
  - Raid configuration
  - Any additional network storage information
- Network
  - # of interfaces in use per LPAR/Partition
  - bandwidth at weakest infrastructure link (gigabit?, 100BaseT?, full duplex?)
  - are there any content-routing or load-balancing components implemented?

Software

MDM

- Detailed information of customized transactions
- Log4JProperties (inside properties.jar)
- Extensions that are in use in the server and data model

WAS

- WAS Version (This can be found in the welcome page of the admin console, top right)
- Has a cluster been configured?
- How many WAS instances of MDM?
- How many other applications are installed on the WAS instances with MDM?
- JVM setting
  - Java Version (full output of java -version) and Brand (ie IBM JDK 1.5)
  - Heap size min and max (as set by WAS admin console)
  - Any custom command line arguments provided to the JVM
- Data source definition
  - Oracle Thin Driver version
  - Connection pool setting: mix and max pool size,
  - Prepared statement size

DB2

- DB2 version (db2level)
- Databases involved other than MDM?
- Snapshots from an active run
- Partitioning/Clustering Configuration
- Storage Configuration

Runtime information:

Batch Processors / Client Application:

- Configuration file for batch (if applicable)
- Any relevant logs collected from client or batch
- JVM “verbose garbage collection” output “java –verbosegc”
- CPU and Memory: vmstat with a measurement interval of 60 seconds (if the Batch Processor / Client Application is running on MDM Application server, use the vmstat interval specified in the section below)

MDM / Application Server

- List of transactions used for load run and their frequency of execution
- MDM Application and WAS logs collected (all the logs generated in that directory)
- Performance Monitor Enabled and set to level 2:
  - /IBM/DWLCommonServices/PerformanceTracking/level = 2
  - /IBM/DWLCommonServices/PerformanceTracking/enabled = true
  - /IBM/DWLCommonServices/PerformanceTracking/SuspectProcessing/enabled = true
  - /IBM/DWLCommonServices/PerformanceTracking/PartyMatcher/enabled = true
- JVM “verbose garbage collection” output “java –verbosegc” (there is a check box to enable this in the WAS admin console)
- CPU and Memory: vmstat with a measurement interval of 30 seconds
- I/O: iostat –xtc with a measurement interval of 30 seconds
- WAS PMI should be left at the default level (this has a negligible impact)
- Enable Logging in and Watch the Tivoli Performance Viewer during the run, specifically:
  - DWL/Customer datasource pool size
  - DWL/Customer datasource connections
  - ORB Thread count
- For details on the Tivoli Performance Viewer, please refer to chapter 14.3 in the following IBM Redbook: **http://www.redbooks.ibm.com/abstracts/sg246392.html?Open**

DB2 / DB Server

- Snapshots with and without resets (DB2 LUW)
- CPU and Memory: vmstat with a measurement interval of 30 seconds
- I/O: iostat –xtc with a measurement interval of 30 seconds (AIX and Linux, use equivalent for Solaris)
- Any I/O runtime data from Storage System (SAN)
- APAR reports (Oracle)
- Omegamon long format accounting report (DB2 z/OS):

http://publib.boulder.ibm.com/infocenter/dzichelp/v2r2/topic/com.ibm.db2.doc.admin/dsnagj18.pdf?noframes=true