#SHAREorg

SHARE
Technology · Connections · Results

Omegamon XE for Storage – Hints and Tips to Improve Performance and Usage

Deborah Reynolds
IBM Corporation

August 12, 2013
Session 14121

SHARE in Boston

Agenda

- **Maximizing Performance**
  - Monitoring Activities
  - Demand on System Resources
  - Understanding of Data Collection
- **Product Usage Guidelines**
  - Defining masks and Using wildcards
  - Data Collection Usage
- **Best Practices**
  - Situations and Policies
  - Storage Toolkit
  - Data Collection Parameters

Maximizing Performance

- **Reduce the impact of monitoring activities**
  - Review collection intervals and data filtered
  - Use masking to reduce the data collected
  - Adjust collection intervals
- **Demand on system resources**
  - Improve the Performance of ITM by reducing the demand on system resources
- **Make collection of historical data more efficient**
  - Understand short vs/long term historical data collection
  - Review the types of data to collect as well as the interval settings

Maximizing Performance – Monitoring Activities

- Use masks to reduce the amount of resources required to retrieve data (exclude non-critical devices)

- Specify appropriate collection intervals to collect sufficient information. It is a balance between using resources and getting the information on resources.

Maximizing Performance – Demand on System Resources

- **Reduce steady-state demand**
  - reviewing and limiting processes that use resources
  - Demand on System Resources
  - Historical Data Collection
- **Reduce user demand**

Maximizing Performance – Reduce Steady-state Demand

- **Reviewing and limiting processes** that use resources all of the time such as background collectors, historical collection, auto-started situations and policies.
  - Determine critical vs noncritical thresholds
    - For less critical events define an interval for the threshold
    - Production vs nonproduction
    - One size does not fit all
  - Tune Thresholds
    - Use method that uses less system resources. Not all methods use the same resources.
    - Omegamon XE uses Boolean logic which provides for more complex situations.

Maximizing Performance – Reduce Steady-state Demand

- **Reviewing and limiting processes…**
  - Disable data collection for noncritical resources
  - Configure data collection intervals
    - Understand how intervals control data collection.
  - Situation monitoring
    - Omegamon XE for Storage comes with many pre-built situations. Only turn on/activate the ones that are necessary

Maximizing Performance – Reduce Steady-state Demand

- ***Example 1:***
  - The Navigator view is like a tree with leaves.
  - Each leaf has a name such as Application Summary, Cache CU Performance, etc.
  - Leaf names link to workspaces. Workspaces contain numerous columns of data gathered by a single data collector.
  - Right click a leaf to view all situations associated with the leaf. All situations on a leaf use the same data collector.
  - All situations are grouped if the same interval setting is used for all situations.
  - *However, if different interval settings are used, then the data collector is called for each situation on the leaf*

Maximizing Performance – Reduce Steady-state Demand

- ***Example 2:***
  - All situations with the same interval setting are scheduled together
  - Generally 4 or less situations are grouped together (have 2 conditions as most situations do)
  - More than 4 situations or complex situations will require more data collections
  - You can apply four warning situations for one interval and four critical situations for another higher interval to save on resources
  - You must restart the HUB TEMS in order to see the benefits of combining (or recombining) situations

Maximizing Performance – Reduce User Demand

- Each time a user requests data, a demand is made on system resources especially tabular views that return large volumes of data.
- When a workspace is opened or refreshed, a query is run.
  - Customize queries to filter out nonsignificant data - reduces memory
  - Limit the number of rows and columns. Default queries return all columns.
  - Apply same query to multiple views in a workspace (one query per table in a workspace)
  - Set auto-refresh to a long interval or turn off

Maximizing Performance – Reduce User Demand

- ***Example:***
  - Use the Dataset Attribute Database feature to write a query that targets data sets on a specific control unit.
  - Refine your query so that it retrieves only monitoring information about space utilization.
  - Substantially reduce the amount of data that the query returns and reduce demand on system resources:
    - By limiting the query to the specific control unit
    - By specifying the attribute that you want to monitor

- It is more efficient to use queries than View filters.

Maximizing Performance – Understanding Historical Data Collection

- **Short Term history**
  - Stored in the Persistent Data Store files usually 24 hours
  - Short-term historical data is used to investigate and determine the nature of problems that arise
  - Data is stored on z/OS
- **Long term history**
  - Stored in the Tivoli Data Warehouse
  - Used to analyze trends and determine workload balance
  - Data is stored in a relational database (DB2, Oracle, Microsoft SQL)
  - Requires Warehouse Proxy Agent

Maximizing Performance – Understanding Historical Data Collection

- **Sample Historical Collection Intervals**
  - Realtime - 5 minutes
  - Short term - 15 minutes
  - Long term - 1 hour

Agenda

- Maximizing Performance
  - Monitoring Activities
  - Demand on System Resources
  - Understanding of Data Collection
- **Product Usage Guidelines**
  - Defining masks and Using wildcards
  - Data Collection Usage
- Best Practices
  - Situations and Policies
  - Storage Toolkit
  - Data Collection Parameters

Product Usage Guidelines

- **Masks**
  - Keep amount of data small
  - Use unique masks
  - Limit time to collect data
  - For data set masks, clear the Space Data option if space data is not required or if the last reference date is not required
  - For more information on masking, see *IBM Tivoli OMEGAMON XE for Storage on z/OS: User's Guide*
- **Wildcards**
  - Do not use wildcard characters in the first qualifier

Product Usage Guidelines

- **Historical Data**
  - For data set groups, collect only data that you need
  - Use an appropriate data collection rate
    (5, 15, 30,60, or 1440 minutes)

- **Optimize the collection of cache statistics**
  - For shared DASD environments that share a HUB TEMS:
    - Use one LPAR to collect cache space statistics
    - Turn off cache space collection on all other LPARs

Agenda

- Maximizing Performance
  - Monitoring Activities
  - Demand on System Resources
  - Understanding of Data Collection
- Product Usage Guidelines
  - Defining masks and Using wildcards
  - Data Collection Usage
- **Best Practices**
  - Situations and Policies
  - Storage Toolkit
  - Data Collection Parameters

Best Practices – Situations

- **Situations** are queries run on a specified interval and report results to the TEMS Situation Monitor.

  The Situation Monitor forwards alerts to the TEPS.

- **Situation Distribution**
  - Only run critical situations from those provided
  - Distribute certain situations to only one z/OS image in a shared environment
    - Those with space attributes, cache status or performance attributes, hardware related

Best Practices – Situations

- **Situation Take Action**
  - Be careful with Take Action on volume and data situations. Can produce unexpected results. Don't want a situation true for 500 volumes at once and kick off 500 tasks.
  - Run situations on DASD groups and not all volumes at once
  - Time slice situations
  - Know your environment

- **Situation Sampling Interval**
  - Set sampling interval to the collection interval of the data
  - Data set attributes database is collected only once every 24 hours

Best Practices – Policies

- **Policies** allow you to combine Boolean logic with multiple situations and actions to be taken.
  - Very flexible and powerful.

Best Practices – Storage Toolkit

- The Storage Toolkit is a feature rich function available only in Omegamon XE for Storage
- Interfaces with DFDSS and HSM to issue commands
- With V420 IF4, Situation definitions have a Storage Toolkit Take Action facility.
  - This allows for Volume actions, Dataset actions, Commands, and Batch jobs when a Situation becomes true

Best Practices – Data Collection Parameters

- DASD Space and Fragmentation Collection
- VTS Collection
- Dataset Performance Collection
- Dataset Statistics Collection
- Collection Parameters Information

Best Practices – DASD Space and Fragmentation Collection

- Collect space data on one image in a shared DASD environment

- Use exclude list RKANPARU(KDFDEVSU) for system volumes (Page, JES and SYSRES)

Best Practices – VTS Collection

- Collect VTS data on one image only

Best Practices – Dataset Performance Collection

- Dataset level I/O monitoring involves the TEMS address space which handles the data collected by the I/O exits. This collection of data needs to be tuned for the environment.

- Decide which volumes need to be monitored.

  Set appropriate thresholds for each volume

Best Practices – Dataset Performance Collection

- Collecting dataset level I/O can be done 2 ways:

  By a fixed interval or by exceptions during degraded performance.

  - Sample Count can be set from 1 - 99.

    If set to 1, every I/O is monitored

    If set to greater than 1, not every I/O is monitored reduceing resources (e.g. 5 is every 5 I/Os is monitored)

  - Exception monitoring uses 2 values:
    - MSR exception value (response time threshold)
    - Global trip count (number of exceptions that must occur before logging of exception)

Best Practices – Dataset Statistics Collection

- Data Set Groups
  - Do not mask high level index
  - Do not collect data for same dataset or same space data multiple times
- Data Set Attributes Database ad hoc queries
  - Include volser in predicate
  - Include dataset name or mask
  - Include volser in situations

Best Practices – Collection Parameters Information

- Monitoring Sampling Intervals
- Historical data collection
- DASD Device Monitoring
- PARMGEN Fields

Best Practices – Monitor Sampling Intervals

- **Cache Statistics**
  - Controls the frequency of data collections for cached volumes and control units (1-999 secs)
  - Set to greater than zero on one LPAR per shared DASD environment. Zero does not disable the collection as documented in the manuals. Set the other LPARs to 999.

- **Cache reset interval**
  - 0-999 minutes or RMF

Best Practices – Monitor Sampling Intervals

- **DASD response time**
  - Controls how often response time statistics for DASD volumes are collected (0-999 secs)

- **DASD space and fragmentation**
  - Controls how often to retrieve space and fragmentation statistics for DASD volumes. (0-99 intervals or RMF)

Best Practices – Monitor Sampling Intervals

- **Tape monitoring interval**
  - 0-999 secs or OFF

- **Application volumes and datasets**
  - Controls how often to rebuild the list of volumes and data sets of the applications that it monitors. (0-999 secs)

Best Practices – Monitor Sampling Intervals

- **Example:**
  - Monitor sampling intervals:

Cache statistics:	300	(5 minutes)
Cache reset interval:	RMF	
DASD response time:	900	(15 minutes)
DASD space/fragmentation	2	(30 minutes}
Tape monitoring interval:	900	(15 minutes}
Application vols/datasets:	300	(5 minutes)

Best Practices – Historical Data Collection

- **Dataset collection enabled and DASD collection enabled?**
  - Writes information about the DASD and data sets that it monitors to IBM Tivoli Enterprise Monitoring Server's persistent data store (PDS)
  - Y/N - set to N to disable
- **Application collection enabled?**
  - Y/N - set to N to disable

Best Practices – DASD Device Monitoring

- You can specify one of three monitoring methods for each volume

- **Method 1**
  - Track every I/O transaction to every data set on a volume and maintain statistics for each I/O transaction.
  - Tracking and maintaining statistics for every I/O transaction to a volume causes more resource consumption.
  - Using response time information to selectively turn on data set tracking for the volume is relatively inexpensive.
  - Tracking any data set activity to the volume uses resources.

Best Practices – DASD Device Monitoring

- **Method 2**
  - Track every *n*th I/O transaction to a volume based on a specified sample count and maintain statistics only for each sample observed.
  - This method provides a reasonable overview, while reducing the demand on system resources that is required to monitor every I/O transaction.

Best Practices – DASD Device Monitoring

- **Method 3**
  - Based upon a specified MSR (millisecond response time) threshold, only track I/O transactions to the volume that exceed the threshold.
  - This method is a cost-effective way to track data set statistics when the volume is not meeting your response time objectives.
  - It ensures that critical statistics are maintained during periods of response time degradation without using system resources when response times are satisfactory

Best Practices – DASD Device Monitoring

- MSR exception trip count
  - Number of MSR exceptions on a device for every 100 I/O events. Applies to MSR monitored devices (1-99)
- For groups of volumes or selected volumes
  - Specify VOLSER, VOLSER pattern or device address or device address range
- Monitor Status
  - Set method of monitoring (ON or MSR)
- Sample Cnt/MSR
  - Sample count or MSR threshold

Best Practices – DASD Device Monitoring

- DASD devices can be excluded on the EXCLUDE DASD DEVICE FROM MONITORING panel.
  - *Exclude as many volumes as possible.*

Best Practices – DASD Device Monitoring

- ***Example:***
  - DASD collection optons:

    Enable SMS storage class name collection? Y

    MSR exception trip count: 51

  - For Volsers/Patterns

    Monitor Status: MSR

    Sample Cnt/MSR: 25 (response time MS)

These settings will monitor every I/O but until an I/O MSR exceeds 25ms 51 times in this sampling interval, the I/O is not recorded as a problem.

Parmgen Fields
DASD Device Monitoring

```
** DASD Device Monitoring:
** Entries for RKANPARU(KDFDSCIN) member:
*KDF_FM                    BEGIN        * Table begin *
*KDF_FM01_ROW              01
*KDF_FM01_VOL              ""
*KDF_FM01_FIRST_DEV        ""
*KDF_FM01_LAST_DEV         ""
*KDF_FM01_MON_STAT         ON
*KDF_FM01_SAM_CNT           1
*KDF_FM                    END          * Table end *
KDF_STG_CLAS_COLL           Y
KDF_MSR_TRIP_CNT           51
```

Parmgen Fields
DASD Device Monitoring

```
** DASD Device Monitoring Exclude List:
** Entries for RKANPARU(KDFDEVSU) member:
*KDF_FX                    BEGIN          * Table begin *
*KDF_FX01_ROW              01
*KDF_FX01_VOL              ""
*KDF_FX01_FIRST_DEV        ""
*KDF_FX01_LAST_DEV         ""
*KDF_FX                    END            * Table end *
```

Parmgen Fields
Data Collection Options

*** Data Collection Options:**

KDF_MON_CACHE_STATS_INTV	900
KDF_MON_CACHE_RESET_INTV	RMF
KDF_MON_DASD_RESP_INTV	900
KDF_MON_SPACE_FRAG_INTV	4
KDF_MON_TAPE_INTV	900
KDF_MON_APPL_VOLS	300
KDF_SMF_NUM	0
KDF_SMF_INTV	OFF
KDF_SMF_IO_CNT_THRSH	25
KDF_HIS_DASD	N
KDF_HIS_DSN	N
KDF_HIS_APP	N

References

- IBM Tivoli OMEGAMON XE for Storage on z/OS, Version 5.1.0, *Tuning Guide*- **SC27-4380**
- IBM Tivoli Monitoring OMEGAMON XE Performance Guidelines, Mike Goodman, Dec 2010
- A Storage Admin's Guide to Restful Nights and Productive Days: OMEGAMON XE for Storage, Vickie Dault, SHARE Session 12008, 08/07/12