RMF BASICS
by Cheryl Watson

This is an introduction to RMF, Resource Measurement Facilities It can be used as an overview of the functions and facilities of RMF. This material has been extracted from my class, "Advanced Tuning with SMF & RMF," which provides a much more detailed look at RMF, including parameter settings, analysis of Monitor I reports, use of Monitor II (foreground and background), and use of Monitor III. The course provides the most detailed examination of RMF that is available and is applicable to system programmers, capacity planners and performance analysts. Please call 1-800-553-4562 for a 1993 course brochure and schedule

RMF BASICS is also intended as a companion and supplement to my monthly MVS newsletter, ***Cheryl Watson's TUNING Letter***. Further information about any of the parameters can be found in the IBM RMF Reference & User's Guides listed in the last section. Please call for a free copy of the newsletter and past tables of contents. **Please feel free to make as many copies of RMF BASICS as you need at your installation.**

History

The first use of RMF was to provide MVS-wide measurements on the primary resources - CPU, I/O and Storage. RMF Monitor I provides this capability today. A second use of RMF was to provide this system-wide information online (under TSO or a dedicated terminal), as well as providing address space information. That addition was provided with RMF Monitor II. The latest addition to RMF was the introduction of Monitor III, an interactive monitor that shows contention bottlenecks and utilization of system resources by address space, performance group or subsystem.

RMF writes records to SMF and can be processed in background using the RMF Post Processor or any number of vendor products. The most common products in use include MXG from Merrill Consultants, Inc. (SAS-based), MICS from Legent Corporation (SAS-based), SLR (Service Level Reporter) from IBM and several user-written programs from various user-group tapes, such as the CBT tape or SHARE Mods tape. Boole & Babbage sells a product called CMF, which replaces RMF. It produces the same RMF record types, but has its own reporting program and produces additional records. This article will use the RMF Post Processor for examples since it is available to most users. A CMF/RMF cross-reference is also available.

Monitor I

The most important component of RMF is Monitor I. It is normally started at IPL and collects data about the system resources. It collects samples and measurements every "cycle", a user-defined sampling period that's normally set to 1 second. Periodically, RMF takes these measurements, determines minimums, maximums and averages, and writes one or more records to SMF. The frequency of writing these records is an "interval" and is normally 15 or 20 minutes, although intervals of 5, 10, 30 or 60 are also used. Keep in mind that the time-stamp on the SMF record is the time that the record was moved to the SMF buffer and can differ from that actual end of the interval by as much as 2 or 3 seconds depending on the dispatch priority of RMF and the other workloads on the system.

The record types produced by RMF, Monitor I, are:

70 - CPU Activity
71 - Paging Activity
72 - Workload Activity (Performance Group Data)
(subtype 0)
73 - Channel Activity
74 - Device Activity
75 - Page and Swap Data Set Activity
76 - Trace Activity
77 - Enqueue Activity
78 - (subtype 1, 3) - IOQueue Activity
(subtype 2) - Virtual Storage Activity

Starting Monitor I

A member of SYS1.PARMLIB called ERBRMFxx (where "xx" is defined by the user) is first built containing the parameters to run Monitor I. An operator command is either placed in IEACMD00 or COMMND00 (or simply entered by the operator) in the form: "S RMF.A,,,MEMBER(xx)". If ERBRMF00 is used, the "MEMBER" parameter is not needed. The ".A" on the start command is referred to as the session id for RMF. It can be any value chosen by the installation. Many sites use "RMF.RMF". RMF will then start the RMF address space and will normally start Monitor I. There is an parameter (NOZZ) to delay starting Monitor I, but it is seldom used

Monitor I Parameters

The following parameters are used to control Monitor I RECORD, REPORT, SYSOUT, (NO)OPTIONS, CYCLE, INTERVAL, SYNC, STOP, CHAN, CPU, DEVICE, ENQ, IOQ, PAGESP, PAGING, TRACE, VSTOR, WKLD, and EXITS. Here's a brief run-down on each parameter and my recommendations. The full details are contained in the IBM "RMF Monitor I & Monitor II Reference and User's Guide".

RECORD/NORECORD - records to SMF. Since this is the primary purpose of Monitor I, you would normally specify "RECORD".

NOREPORT/REPORT(REALTIME)/REPORT(DEFER) - specifies whether RMF will dynamically allocate and produce reports while it's running Since all of the data is recorded to SMF for later processing by the RMF Post Processor, there's seldom a reason to request dynamic reports, so "NOREPORT" is most often specified and recommended.

SYSOUT(x) - If you specify REPORT, this parameter defines the SYSOUT class for the reports. Normally not needed.

OPTIONS/NOOPTIONS - defines whether the parameters are listed at RMF start-up and whether the operator can override the parms Your choice depends on the experience of the operators and

your security requirements.

INTERVAL(nnnM) - defines the number (nnn) of minutes between writing of SMF records Data is accumulated every CYCLE (see below) and recorded to SMF every INTERVAL The most common intervals are 15 or 20 minutes Smaller intervals, such as 5 or 10 minutes provide a more accurate picture of the system, but result in more overhead. Larger intervals, such as 30, 45 or 60 minutes cause less overhead, but don't provide accurate measures of system resources, especially when evaluating averages or determining peaks This is the most important of all parameters and I would recommend using 15 minutes

CYCLE(nnnn) - number of milliseconds between samples. This parm is normally left to the default of "1000" (1 second), but should be used in conjunction with the INTERVAL parameter. You should attempt to collect at least 800 to 1000 samples per interval to obtain a valid statistical measurement. A cycle of 1000 for a 15 minute interval results in 900 samples and is just fine However a cycle of 1000 for 5 minutes only produces 300 samples, so should be reduced to about 333 ms (1/3 second) to produce 900 samples during the 5 minute interval Likewise, an interval of 60 minutes (which is often used to reduce overhead) can use a cycle of 5000 (5 seconds) since averages won't be very accurate anyway (and the 5 second cycle also results in less overhead)! The 60 minute interval will accurately collect totals, such as total CPU time and total TSO transactions, despite the cycle time.

SYNC(nnm) - denotes that intervals are to be synchronized at a specific minute of the hour SYNC(0M) with an INTERVAL(15M) will result in SMF records written at 8:00, 8:15, 8:30 and 8:45. MICS logic requires that SYNC(59M) be used, since they extract the hour for summarization The reason for this is that, due to the timer logic, it is possible when using SYNC(0M) to obtain records at 7:59:54 on one day and 8:00.02 on the next day. Selecting records from 8:00am might miss the data for the first day. For that reason, any system is better off using SYNC(59M). You'd get records written at 7:59, 8:14, 8:29, 8:44 and 8:59 You would never extract the wrong time period. SYNC is a must on a shared DASD system where you want to look at DASD devices from both systems. I'd recommend SYNC(59M) RMF 4.3 (scheduled for 3/93) will allow synchronization with SMF and will expand the specification for SYNC.

NOSTOP/STOP(nnnM/H) - specifies when Monitor I is to be stopped Since I'd never want it stopped, I'd always use NOSTOP.

CHAN - indicates that channel busy information is to be collected and written to SMF type 73 records This is necessary to collect data for balancing channels

CPU - specifies that CPU and PR/SM activity is to be kept and written to SMF type 70 records This is probably one of the most important records to collect

DEVICE(option,option,..)/NODEVICE - defines that device activity is to be kept The "options" specify which devices are kept. CHRDR (character reader), COMM (communications devices), DASD (good old DASD), GRAPH (graphic devices), TAPE (tape drives), UNITR (unit record), NMBR(nnnn) - device numbers These records (type 74) can be very large, especially if you collect communication devices which include CTCs, local terminals and communication controllers. I'd recommend either "DASD" or "DASD,TAPE" and nothing else unless you have a specific reason to collect other devices.

PAGESP - collects information on page and swap datasets and is needed to help you balance and tune your paging configuration This parameter produces a type 75 record for each page and swap dataset

IOQ(option,option...)/NOIOQ - collects logical control unit information for the same type of devices as shown on the DEVICE option. A type 78 record is produced with different formats depending on whether the CPU is a 3090/9000 or other machine. Recommended parm is IOQ(DASD)

VSTOR(S)/VSTOR(D,jobname,jobname,...)/NOVSTOR - provides virtual storage information at a system-wide basis "VSTOR(S)" and/or at a detailed (D) job level. The system-wide data can tell you when CSA, SQA or PLPA is growing, while the job-level information can show the use of virtual storage by a job The system information is always useful and the job-level information can be turned on for one day to obtain useful information for your online systems or other "loved ones" VSTOR produces a type 78 record.

PAGING - specifies that paging, swapping and expanded storage measurements are to be collected and recorded in the type 71 record Since this provides the primary measurements for storage analysis, this data should always be kept

WKLD(list)/NOWKLD where "list" = (NO)PERIOD, (NO)GROUP, (NO)RANGE, (NO)OBJ, (NO)DOMAIN, (NO)SYSTEM, (NO)TIME - collects information by workloads performance group periods, performance groups, range of performance groups, service objectives, domains, time slice groups and total system. The fields in the list indicate the lowest level of detail to be collected. Since this information is required to analyze SRM parameters, you should keep the lowest level or "WKLD(PERIOD)". Type 72 records are produced, one per performance group.

TRACE(fieldname)/NOTRACE - defines fields that are to be tracked using the RMF trace facility. This facility provides more detailed information than normal Monitor I reports for certain fields. This is only of use if you are doing detailed analysis of the data and can normally be set at NOTRACE until you're ready for tracing. Tracing produces type 76 records, one per trace field. This is an extremely powerful facility in RMF and should be investigated.

ENQ(DETAIL)/ENQ(SUMMARY,majname,minname)/NOENQ - collects enqueue contention information. Analysis of enqueue contentions can result in elimination or reduction of delays due to conflicts. The DETAIL option provides jobnames that are holding or waiting on resources, while the SUMMARY option only indicates the name of the resource and the length of contention On large systems, the DETAIL information can be very extensive Some installations never look at this data, while others review it daily. Turn on the level of detail that you're willing to analyze. I've often turned on "ENQ(SUMMARY)" until I see that I need more detail for analysis, then turn on "ENQ(DETAIL)"

In summary, the following parameters are typically used for Monitor I collection.

RECORD	CPU
NOREPORT	DEVICE(DASD,TAPE)
SYSOUT(A)	PAGESP
OPTIONS	IOQ(DASD)
INTERVAL(15M)	VSTOR(S)
CYCLE(1000)	PAGING
SYNC(59M)	WKLD(PERIOD)
NOSTOP	NOTRACE
CHAN	ENQ(SUMMARY)

Monitor II

Monitor II provides a different type of RMF session. It collects much of the same data as Monitor I, but additionally collects information about address spaces. It can be run under TSO or can be run as a background session that records SMF records, produces reports or both It's normally run under TSO to obtain current address space activity or system status (sort of a Monitor I online!) It's run as a background session when data needs to be collected over a longer time period, such as an hour, prime shift or a day

I've used Monitor II as an online monitor to determine why jobs were swapped out, how efficient the DASD devices are, what enqueues are impacting performance and how TSO response times are doing. In background, I've used Monitor II to provide storage usage for determining storage isolation parameters, validating CPU utilization (at every minute rather than 30 minute Monitor I interval), tracking paging and swapping during prime time and evaluating a single job as to its CPU, I/O and storage usage

A word about "other" online monitors. Many installations have ignored both Monitor II and Monitor III because they have purchased other online monitors, such as Omegamon from Candle Corporation, CMF from Boole & Babbage, and The Monitor for MVS from Landmark, to name a few. Although Monitor II and III contain fewer reported fields, their ability to record data to SMF over long intervals, such as one hour or one day, make them valuable tools, at no extra charge!

Although you may not use Monitor II as an online monitor, it can be an excellent tool for post-processing Let's assume that Monitor I data is being collected every 20 minutes and the "average" CPU busy is 50% during the 20 minutes. If it was 100% busy for 10 minutes and 0% busy for 10 minutes, the average is still 50% The Monitor I data won't show the peak and minimum utilization, but an hour-long Monitor II session recording every 5 minutes **could** show you the peaks and valleys. It can also be an invaluable tool in determining system paging parameters for SRM.

At least give it a try!

RMF Monitor II produces a type 79 SMF record with several subtypes (each subtype produces a separate record).

1 - ASD/ASDJ - Address Space State Data
2 - ARD/ARDJ - Address Space Resource Data
3 - SRCS - System Resources
4 - SPAG - System Paging
5 - ASRM/ASRMJ - Address Space SRM Data
6 - SENQR - Device Reserve Data
7 - SENQ - Enqueue Contention
8 - TRX - Transaction Data
9 - DEV - Detailed Device Activity
10 - DDMN - Detailed Domain Activity
11 - PGSP - Page & Swap Dataset Activity
12 - CHP - Channel Path Activity
13 - IOQ - Logical Control Unit Data for non-3090
14 - IOQ - Logical Control Unit Data for 3090

Starting Monitor II

Under TSO, either at the TSO READY prompt or option 6 of ISPF, enter "RMFMON" It may have been renamed or disabled at your installation if you can't get to it See the section later on "Monitor II Under TSO". Starting a background session requires an operator command "F RMF.A,S yy,MEMBER(xx)" will normally start a background session, where "RMF A" is the active RMF session, "xx" is the suffix on an ERBRMFxx member of SYS1.PARMLIB containing Monitor II parameters, and "yy" is any 2-character identifier that you specify This is handy to later stop the session with "F RMF.A,P yy". An alternate way to run Monitor II is to a dedicated local 3270 terminal. You start a session with: "F RMF.A,S 14C,MEMBER(xx)" where "14C" is the terminal device address.

Monitor II Parameters

Before starting a background session, you should specify parameters in an ERBRMFxx member. The following parameters are the same used for Monitor I OPTIONS, RECORD, SYSOUT, STOP and SYSOUT Additional parameters include.

SINTV(nnn) - the number of seconds in the interval for this Monitor II session. Most of the time, this is a much smaller interval than Monitor I although it doesn't need to be. The default is 30 seconds which is very frequent and can cause some high overhead. Most people simply use it to obtain more detailed data than available from Monitor I. A typical SINTV value might be 2 minutes or 10 minutes. It really depends on the particular need for the data. As in Monitor I, larger intervals result in less overhead and less accuracy, while smaller intervals produce more records with more overhead, but greater detail and therefore higher accuracy.

DELTA/NODELTA - much of the address space data provides transaction totals, such as total TCB time and total SRB time. You might be more interested in what happened between intervals (such as the TCB time and SRB time during the last 2 minutes). "DELTA" provides just the values accumulated during the interval.

ASD(class,status,domain)/NOASD/ASDJ(job1,...) - collects address space information on storage usage, swapping and dispatch priority. "class" can be A (all), B (batch and STC) or T (TSO) "status" is A (all), I (active - swapped in). "domain" is A (all) or nnn (domain number). This data can be used to determine why a job is swapped out, what dispatch priority it's running at, the number of swaps since it started and SRM workload priority. ASD provides data on all address spaces, while ASDJ collects data on selected address spaces.

ARD(class,status,domain)/NOARD/ARDJ(job1...) - collects even more address space information giving paging rates, EXCP rates, CPU times and device connect time. The parameters are the same as ASD.

ASRM(class,status,domain)/NOASRM/ASRMJ(job1,...) - collects even more address space information giving service unit measurements, transaction counts, active and resident times (so you can determine the length of time swapped out). The parameters are the same as ASD. If you're **really** evaluating a job, use of ASDJ, ARDJ and ASRMJ are recommended.

TRX(subname,nnn,nnn:nnn/ALLPGN)/NOTRX - collects response time and transaction rate for one or more performance groups ("nnn") within a subsystem ("subname" can be TSO, STC or BAT) This is primarily used to check on TSO response time.

SENQ(S/D/A,sysname/E,sysname/majname,minname)/NOSENQ - this provides Monitor I "ENQ" data and is normally used to determine enqueues as they occur.

SENQR(ALLVSER/volser)/NOSENQR - provides reserve contention, similar to data provided by Monitor I "ENQ" data.

SRCS/NOSRCS - provides a snapshot of the major system indicators, such as available frame queue, UIC, CPU utilization, swap location and storage usage. This is often useful to see how the indicators vary over a very short period of time, such as one or two minutes

SPAG/NOSPAG - provides as summary of paging and swapping activity as well as central and expanded storage indicators,

such as UIC, migration age, and migration rate. This data can be invaluable when trying to determine valid SRM parameters.

DEV(list/VOLSER(aaaaaa,...))/NODEV - this collects a subset of the Monitor I "DEVICE" parameter, but it's the most important subset: response time (with its components) and activity rate. This can be used online to detect poorly tuned devices or as a background session to determine true peaks and valleys for the device statistics.

PGSP(PAGE/SWAP)/NOPGSP - provides Monitor I "PAGESP" data under Monitor II to help you manage your page and swap datasets.

CHANNEL/NOCHANNEL - identifies channel path activity and is identical to the Monitor I "CHAN" collection.

DDMN/NODDMN - provides domain constraints and domain activity showing min, max, target and current MPLs, average ready users and the domain workload. This can be invaluable for setting domain constraints in SRM.

IOQUEUE(list/NUMBER(lcu,lcu:lcu))/NOIOQUEUE - where "list" is the same as Monitor I "IOQ" parameter. This collects logical control unit data similar to Monitor I.

You may want to get a sample of each report. The following parameters would collect a half hour summary with at least two sets of measurements. There is a small overhead to this collection, so try it first during non-peak hours! Once you get familiar with the reports, you could select just the data that you need:

RECORD	-or-	NORECORD
NOREPORT	-or-	REPORT(DEFER)
SYSOUT(A)		
OPTIONS		SPAG
SINTV(15M)		TRX(TSO,ALLPGN)
CYCLE(1000)		SENQ(S)
SYNC(0M)		SENQR(ALLVSER)
STOP(30M)		DDMN
CHANNEL		ASD(A,I,A)
SRCS		ARD(A,I,A)
DEV(DASD,TAPE)		ASRM(A,I,A)
PGSP		IOQUEUE(DASD)

Use NORECORD and REPORT(DEFER) if you want to simply produce a set of reports. Use RECORD and NOREPORT if you want to record to SMF and use the Post Processor to produce the reports.

Monitor II Under TSO

Monitor II under TSO can be used as an online monitor to determine contention or problems on the system as well as a training tool. I've used it to find non-swappable address spaces (or so I thought) that were swapping thousands of times per day. I've also used it to determine storage isolation parameters for my CICS systems and to determine SRM domain constraints. You should at least be familiar with this tool. So logon now and experiment. "RMFMON" is the command that will get you into RMF Monitor II.

A menu is displayed and you can select from the menu. Most of the parameters used for a Monitor II background session can be used in foreground: ASD, ASDJ, ARD, ARDJ, ASRM, ASRMJ, TRX, SENQ, SENQR, TRX, SRCS, SPAG, DEV, DEVV (to select specific volumes), PGSP, CHAN, DDMN, and IOQUEUE. Additional commands under TSO:

M - displays the menu

MM - displays default operands

D ON/OFF - turn delta mode on and off

H ON/OFF - turn hardcopy on and off to get reports

T x,y - set automatic timed update (refresh)

F - frame through the report (similar to scroll down)

P - print this report

Z - end the session (you'll forget this one and agonize on how to get OUT of RMFMON!)

RMF Post Processor

The RMF Post Processor is a program that produces reports from the RMF Monitor I and II records. The JCL is fairly simple and consists of:

```
//STEP1   EXEC PGM=ERBRMFPP
//MFPINPUT  DD DSN=smf-dataset,DISP=SHR
//SYSIN     DD *
--control statements--
/*
```

Other DD statements can be specified for the reports, but can be extremely difficult to predict (the ddnames are dynamically created), so most people ignore them. The post processor (called PostP from here on), reads the SMF file and produces reports based on the control statements. Please keep in mind that the PostP doesn't sort the input file and will produce reports based on the sequence of the input file. SMF files that are created by concatenating several SMF recording files are often out of sequence.

There are five types of reports that can be produced: interval, duration, summary, plot and exception reports. The interval reports display the lowest level of detail and provide one report per Monitor I interval (per 15 minute period for example). The duration reports look similar to the interval reports but contain multiple intervals. A duration report for one hour would contain the data for four 15-minute intervals. Average fields would be calculated as the average over the hour and total fields would contain the totals for the hour. A summary report prints one line per interval with some of the more important data elements on each line. Elements like CPU Busy, DASD rate, maximum and average TSO users are listed. Summary reports can be used to get a quick glimpse of the system throughout a period of time, such as day shift or a 24-hour day. Plot reports produce a graphical display of one element showing the value of the data element on a horizontal bar graph. Exception reports list exceptions to thresholds defined by the report requestor. Exception reporting is one of the most powerful facilities of the PostP and I'd recommend looking at it.

Control statements define selection of the input records as well as a list of reports that are desired. Control statements include the following:

DATE(yyddd,yyddd) - indicates the dates for the records to be included If not specified, all dates on the file are used.

xTOD(hhmm,hhmm) - specifies the time of day for selection of data from the input file. **ETOD, PTOD, RTOD, STOD** (exception reports, plot reports, interval/duration reports and summary reports) define starting and ending times for selection of data from the input file. If the starting time is less than the ending time, then all data between the two time-stamps is selected If the starting time is greater than the end time (such as RTOD(2000,0800)), then data starting at 8 00pm in the evening until 8:00am the next day is selected. Be careful of the RMF manual - the diagram is wrong! This also means that if you want to report on weekday processing

Monday at 8:00am to Friday at 5:00pm, you'll need to run three passes of the report - one for Monday (0800,2400), one for mid-week (0000,2400) and one for Friday (0000,1700). Also keep in mind that this selection is based on the **ending** timestamp of the interval A record written at 8:00am is REALLY for the period of 7:45 to 8:00!

EXITS/NOEXITS - define whether user exits are to be invoked. Very few installations use RMF PostP exits

SYSID(xxxx) - used to select a single system id from the input file if multiple SMF files have been merged.

SYSOUT(x) - specifies the default sysout class for the reports.

Duration Reports:

DINTV(hhmm) - specifies the amount of time for summarization to create a duration report For example, if the INTERVAL is 15 minutes, a DINTV(0100) would summarize four interval records to report a single hour's worth of data.

Plot Reports:

PTOD(hhmm,hhmm) - time of day for plot reports.

MAXPLEN(nnn) - maximum number of lines of a plot report.

PINTV(hhmm) - indicates the range of intervals to be summarized on a single line of the plot.

PLOTS(BATCH,CH(nn),CPUID(n),CSAA,CSAFP,CUBDL(nnn), DAC(nnn),DCON(nnn),DRT(nnn),ESMR,HUIC(nn),IOAC(nnn), PAGE,PTES,SEV,SQAA,SQAE,SQAFP,STC,SWA,TPAG,TRA,TSO) - defines the data fields to plot. Use the manual to determine the contents of each report (for example: BATCH, STC and TSO plot the maximum number of address spaces of each type).

Summary Report:

STOD(hhmm,hhmm) - time of day range for the Summary Report. Most typical is (0000,2400) or (0800,1600).

SUMMARY(INT/TOT)/NOSUMMARY -SUMMARY(INT) displays one line per recording interval with a total line while SUMMARY(TOT) only displays the single total line.

Exception Reports:

EXCEPT(name(condition,LE/GE,threshold)) - allows you to define some exceptions for the PostP to identify and report. "name" is an identifying name that you invent to describe the exception. "condition" is one of a set of specific data elements that RMF can identify and "threshold" is the value that you want to test. For example: EXCEPT(HIGHTSO(MAXTSO,GE,50)) would produce a line on the exception report whenever the number of logged on TSO users met or exceeded 50. It would tell you the time of day, the exception that triggered the line and the actual number of users. This can be a very powerful tool (in other words, let the computer do your work!). The manual defines the list of valid conditions.

EXRPTS(name(report,report,...)) - defines which Monitor I reports are to be produced when an exception occurs "name" defines the exception defined with the EXCEPT parameter. For example. EXRPTS(HIGHTSO(CPU,WKLD)) would produce the CPU Activity Report and the Workload Report whenever the exception specified by HIGHTSO occurred (i e when the number of TSO users exceeded 50).

Duration/Interval Reports:

RTOD(hhmm,hhmm) - defines the selection times for the reports. If RTOD(0800,0900) is specified without DINTV, and the INTERVAL is 15 minutes, you'll get about four 15-minute reports, depending on the time-stamp in the record. The same selection times specified with DINTV(0100) would produce a single one-hour report

REPORTS(ALL/CHAN/CPU/DEVICE/ENQ/IOQ/PAGESP/ PAGING/TRACE/VSTOR/WKLD/
(Monitor I)
ASD/ARD/ASRM/CHANNEL/DDMN/DEV/IOQUEUE/PGSP/ SENQ/SENQR/SRCS/SPAG/TRX)
(Monitor II)

provides selection of the reports for Monitor I and Monitor II. Simply list the type(s) of report to be produced.

A common set of parameters that I use to get a snapshot of a system is as follows:

```
          //PRINT   EXEC PGM=ERBRMFPP
          //MFPINPUT DD   DSN=input SMF file,DISP=SHR
          //MFPMSGDS DD   SYSOUT=A
          //SYSIN    DD   *
==>        DATE(yyddd,yyddd)   /* select a single day */
           REPORTS(ALL)
==>        RTOD(1000,1015)     /* pick a peak 10-15 min period*/
           SUMMARY(INT,TOT)
           STOD(0000,2400)     /* 24-hour period */
==>        SYSID(xxxx)         /* omit to get all systems */
           SYSOUT(A)           /* report SYSOUT class */
          /*
```

NOTE: The RTOD(hhmm,hhmm) should be at least as long as your RMF recording interval. That is, if your recording interval is one hour, use RTOD(1000,1100). That's (hhmm,hhmm).

RMF Monitor III

RMF Monitor III is primarily a contention monitor that shows where the primary delays are. It is a very useful tool, however, to determine how well or how poorly the system or subsystems are running. If you haven't used Monitor III in a while, please review it under MVS/XA 2.2 or MVS/ESA, where there have been some impressive improvements. If you want to get started, here's some summary information to get you going under MVS/XA 2.2 or MVS/ESA. (These are easier because Monitor III is run under TSO and uses standard dialog manager interface).

Starting Monitor III

Monitor III collects data and stores it in an internal table in the address space. If you have defined some VSAM history datasets, Monitor III will periodically store sections of the table to the VSAM file(s). If no files are specified, the table is simply overlaid with new data. The use of the VSAM files is to provide historical data, either for the last hour, the last day or the last week. Because of the volume of data, most installations only store 3 to 4 days online. This example will use no files - it's just to get you access to the reports.

Define an ERBRMFxx member with parameters as described in the next section. Start Monitor III with an operator command of: "F RMF,A,S III,MEMBER(xx)" After Monitor III has been collecting data, you can sign onto Monitor III under TSO by entering RMFWDM A menu is displayed and you can go from there. If you didn't specify STOP on your parameters, you can end Monitor III collection with "F RMF A,P III"

Monitor III Parameters

CYCLE, OPTIONS, STOP, SYNC, SYSOUT - similar to Monitor I.

DATASET(suboptions) - indicates the name of the VSAM history datasets. To bypass the use of datasets, omit this parameter or use the default of. DATASET(STOP)

MINTIME(seconds) - similar to the Monitor I interval. Every CYCLE, Monitor III collects data, then every MINTIME it will perform calculations and store the data in the table. The default of 100 seconds may cause unneeded overhead while you're just getting familiar with the report, so you might increase this to 300 seconds (5 minute intervals) for testing.

RESOURCE(*JES2/*JES3) - identifies the primary JES system.

Understanding Monitor III

Monitor III has introduced a whole new set of terms, many of them a little confusing at first. Without going into great detail, let me summarize just a few terms:

Resource - Monitor III monitors the resources and determines if a user is using or waiting on a resource. A user can be "using" either the CPU (PROC) or a device (DEV). A user can be "waiting" on the CPU (PROC), a device (DEV), storage (STOR), an enqueue resource ENQ), HSM, JES, an operator reply (MSG) or a mount (MNT).

WORKFLOW% - indicates how well the job is getting its needed resources. A WORKFLOW% of 100% means that the job is not delayed for anything and is getting all of its resources. A WORKFLOW% of 20% means that there is a lot of delay for the job

DELAY% - indicates how much a job is delayed for any resource. The higher the DELAY%, the longer a job is delayed. The DELAY% is made up of delays for each of the resources, so you can determine where the major delay is. That is, if a job has a DELAY% of 60%, with 40% for DEV, then the major delay is I/O. However, if the major delay is PROC, then the job is simply far down on the dispatch queue and someone else is getting the CPU. This is an excellent place to determine if online systems are being impacted by high paging rates (the DELAY% would show large delays due to paging).

Summary

This overview of RMF is a review of some of the basic parameters needed to collect and report this very valuable data. Try to get familiar with the parameters and collection techniques of RMF

RMF RELEASES

o RMF V3.3 for MVS/SP 2.1.3
- Manuals - GC28-1398 - General Information.
LC28-1138 - Reference & User's Guide.

o RMF V3.4.0/3.4.1/3.5 for MVS/SP 2.1.3 to 2.2
- Manuals - LC28-1556 - Monitor I & II Reference & User's Guide.
LC28-1557 - Monitor III Reference & User's Guide
SC28-1558 - SAM User's Guide.
LC28-1559 - SAM Diagnosis Reference.
SX22-0009 - Monitor I & II Summary.
SX22-0010 - Monitor III Summary.

o RMF V4.1/4.1.1 for MVS/SP 3.1.0/3.1.0e
- Manuals - GC28-1028 - General Information
GX22-0012 - Ordering Guide
GC28-1415 - Programming Interfaces.
LY28-1007 - Monitor I & II Reference & User's Guide.
LY28-1008 - Monitor III Reference & User's Guide.
SX23-0023 - Monitor I & II Summary.
SX23-0024 - Monitor III Summary
GC28-1020 - RMF Messages & Codes
LY28-1001/4 - RMF PLM, Vol. 1 & 2
LY28-1005 - RMF Data Areas.

o RMF V4 1 2 for MVS/SP 3 1 3
- Manuals - GC28-1020 - RMF Messages and Codes
GC28-1028 - RMF General Information
LY28-1007 - RMF Monitor I & II Reference & User's Guide
LY28-1008 - RMF Monitor II Reference & User's Guide
GC28-1022 - RMF Program Directory
GC28-1415 - RMF Directory of Program Interfaces
GX22-0012 - RMF Publications Ordering Guide
SX23-0023 - RMF Monitor I & II Reference Summary
SX23-0024 - RMF Monitor III Reference Summary
SC28-1019 - SAM User's Guide
LY28-1009/1015/1001-1005 - RMF Diagnosis, PLM, Data Areas

o RMF V4.2/4.2.1 for MVS/SP 4.1/4.2
- Manuals - GC28-1020 - RMF Messages and Codes
GC28-1028 - RMF General Information
LY28-1007 - RMF Monitor I & II Reference & User's Guide
LY28-1008 - RMF Monitor II Reference & User's Guide
GC28-1058 - RMF User's Guide
GC28-1120 - RMF Program Directory
GX22-0012 - RMF Publications Ordering Guide
SX23-0023 - RMF Monitor I & II Reference Summary
SX23-0024 - RMF Monitor III Reference Summary
LY28-1009/1015/1303 - RMF Diagnosis, PLM, Data Areas