The Computer Measurement Group, commonly called CMG, is a not for profit, worldwide organization of data processing professionals committed to the measurement and management of computer systems. CMG members are primarily concerned with performance evaluation of existing systems to maximize performance (e.g., response time, throughput, etc.) and with capacity management where planned enhancements to existing systems or the design of new systems are evaluated to find the necessary resources required to provide adequate performance at a reasonable cost.

This paper was originally published in the Proceedings of the Computer Measurement Group’s 2005 International Conference.

For more information on CMG please visit www.cmg.org
DB2 requests originate from a variety of diverse sources including batch, CICS, DDF, and SAP. Related DB2 CPU usage can be recorded in RMF, DB2, SMF 30, and other subsystem records. Proper selection and interpretation of these values will vary with transaction source, DB2 environment, product levels, and analysis objectives. Analysts must be careful to include all desired values and avoid multiple counting of the same logical utilization. This paper will discuss the sources and analysis of DB2 CPU metrics and corresponding response times. Examples will include CICS, DDF, and SAP.

INTRODUCTION

The goal of this paper is to provide an overview of DB2 CPU and response metrics for traditional large system performance and capacity analysts in an MVS (ESA, OS/390, z/OS, etc.) environment. Attempts will be made to provide high-level definitions when appropriate. Included at the end are sources of DB2 record layouts, references, and a glossary. Other good definition sources are the glossary in the DB2 Installation Guide for DB2 V8 and DB2 Administration Guide for earlier releases, and your DB2 specialists. Data sources discussed will be primarily from SMF. Specific fields will be referenced but it is always best to validate field names for the DB2 levels installed. Examples shown use real data though they do not necessarily all represent optionally running systems. Obviously good performance is very subjective and varies with objectives.

Most installations have a central performance analysis and capacity planning function (group, person, part of a person) that reports, tracks, and evaluates overall performance and capacity requirements. Skills that exist within the central function are typically traditional mainframe and operating system related. Detail installation and analysis skills tend to exist within separate functional areas for subsystems (e.g. DB2, IMS, and CICS). Reporting typically includes installation totals, systems (e.g. MVS image), subsystems (e.g. TSO, CICS, DB2), and workloads (e.g. accounting, development). The central function should have enough knowledge of installed subsystems to provide system level analysis and reporting. It is also necessary to know when to get appropriate subsystem specialists involved. Teamwork and communication are key factors.

From a high level, not much has changed for many years. MVS and DB2 have been around for a long time, CPU cycles are the most important planning resource, paging is to be minimized (though it is rarely a current issue), and balanced I/O capabilities must be available. At a detail level, change is continuous. MVS and DB2 provide a very flexible evolving architecture, which can cause significant performance data deviations across installations as a result of different levels, setups, applications, and development techniques. As the evolving architecture is exploited, it is necessary to keep reporting systems up to date.
DEFINITIONS

Before getting into data sources and analysis, a couple of important terms will be defined. An underlying assumption is that reporting capabilities will be consistent with these definitions.

Workload

Refers to an identifiable consumer of system resources that produces answers for an end user. Ideally it is reported in terms of department, application, or application for a department. Often reported in terms of delivery vehicle or requestor (e.g. batch, TSO, CICS, IMS). Typically a reporting entity that consists of one or more lower level entities which could be service classes, report classes, accounting codes, DB2 plan names, etc. There should be enough workloads to be able to report major users of resources separately, but not so many that detailed analysis becomes cumbersome. It is logical to have more than one set of workload definitions to be used for different reporting and analysis perspectives.

Transaction

The basic unit of a workload as requested by a user. This could be a batch job or an individual interactive request. It should be a measurement with which a user can identify.

EXPECTATION LEVEL

A key starting point is to have well-defined goals of what represents good performance. If there are no defined goals, performance management can become very emotional. Ideally there should be formal service level agreements backed up by a reporting system. The targets should be in terms that user departments can understand and must also be consistent with available reporting capabilities. Responsiveness, throughput, and application complexity are interrelated factors. It is risky to have a response target without a related workload level, but it is logical to have a workload level without a response objective. Interactive workloads will typically have a response-oriented target. It is important to have response targets consistent with transaction complexity and expected volumes. Batch and other lower priority workloads might have volume targets without associated response objectives. Following are logical target definitions:

Workload X has a target of 95% of transactions completing within 1.5 seconds, as long as no more than 10,000 per hour are executed.

Workload Y has an objective of 1,000,000 transactions per day.

Workload Z has an objective of 5 CPU hours per day. (CPU hours should be normalized to some constant to allow for processors of different power.)

Since available data does not always support desired reporting, some compromise might be necessary. An unfortunate problem is that most response time measurements are internal and do not reflect what the user sees directly. Network tools can sometimes be used to measure total response time but they are not typically granular enough to identify the workload component. A related issue is that transaction counts from monitors might contain internally generated ones that the user does not see. Since this paper is about DB2 metrics and not service level reporting, primary focus will be on DB2 related data.

When expectation levels are not being met there are essentially the following four choices:

Tuning: Often easier said than done. Might be as simple as changing priorities or memory use. Helping one workload frequently hurts another, though some changes might help DB2 and other work. A strategic database index can result in significant I/O reductions, which could also help workloads other than DB2.

Additional Capacity: Must determine what resource(s) will help. Good planning, tracking, and forecasting will allow this to take place before performance degrades below targets.

Rewrite: Rarely a practical choice. If an application is so inefficient that rewrite would help significantly, it should have been caught before production. This is mentioned primarily for completeness.

Change Targets: If the target is not reasonable for an application, it should be changed. This also should have been caught sooner.
Before getting to data, it is important to have an understanding of the basic structure of DB2 and how requestors of service interface with it. While a simplification, DB2 can be thought of as being a super access method. It uses MVS subsystem interface (SSI) protocols and cross-memory services. Data sharing among multiple DB2s on different images is accomplished by using the coupling facility. There can be multiple DB2 subsystems in an MVS image, but most production work will typically be performed by only one. While there is generally no performance advantage to having multiple DB2 subsystems on an image, there is a value to separate development and test DB2 subsystems. A DB2 subsystem (identified by SSID) includes at least three system address spaces: reference Figure 1. A master or system services address space (MSTR or SSAS) provides overall control functions. The database manager (DBM1 or DBAS) contains buffer pools, EDM (environment descriptor management) pool, and supporting code. The EDM pool contains information on currently open databases and transaction (plan) descriptions. The inter-regional resource lock manager (IRLM) provides locking support.

If distributed DB2 is active, there will be a distributed data facilities (DIST or DDF) address space. DDF will be used to refer to the function and DIST to identify the address space. TCP/IP and VTAM are used by DDF. Potential clients include remote DB2 and any system (e.g. Windows, UNIX, or Linux) capable of communicating with DB2. Server application code is executed under control of this address space. Enclaves and preemptible SRBs are facilities added to MVS and exploited by DB2 to enhance performance of parallel (dependent enclaves) processing and DDF (independent enclaves) requests. Enclaves are effectively virtual address spaces. Preemptible SRBs generally behave like TCBs but are much less costly to establish.

There might also be one or more stored procedure address spaces (SPAS) that essentially contain preloaded and initialized DB2 transaction code. This capability was added in DB2 Version 5 in order to provide enhanced client/server performance (DDF); however, it can also be used for local processing.

Local requests for DB2 processing can come from a variety of allied address spaces including CICS, IMS, TSO, and batch. Cross memory services is used for local communication. The structure connecting DB2 with an allied address space is a thread. A thread processes one transaction at a time. In the case of CICS, there can be multiple threads, each of which will have a separate TCB within the CICS address space. Most DB2 application processing is done within the requesting address space and at the priority of the requestor. Parallel processing is accomplished with preemptible client SRBs established within the requesting address space. RRSAF (Recoverable Resource Manager Services Attachment Facility) is used by SAP.

Server (DDF) requests are received from a remote system so there is no local requestor address space. The requestor for reporting and management purposes is DIST. Priorities, enclaves, and number of SPAS address spaces are managed by the workload manager (WLM). It is conceivable that a DB2 might be a server that only processes remote client requests. In that case only the system address spaces would be present. One could make a parallel between that structure and IMS where all processing takes place within a small set of address spaces. Reporting based on WLM service and report classes might be adequate, though resource consumption measurements would not be very granular.
DATA SOURCES

There are numerous sources of DB2 accounting and performance data. Depending on environment and objectives, it might be necessary to use more than one data source. Although RMF is an excellent starting point and source of performance data in general, CPU consumption for local DB2 applications is not recorded where many analysts expect. DB2 accounting data (SMF 101 records) is typically needed to fill this gap.

Traditionally there have been two kinds of CPU time – TCB and SRB, where SRB was not preemptible or WLM managed. With the existence of preemptible SRBs it is best to think of preemptible and non-preemptible tasks. Preemptible includes TCB and preemptible SRB, both of which have similar characteristics relative to WLM. To simplify references in this document, SRB will mean non-preemptible SRB and pre-SRB will be used for preemptible SRB. Unless otherwise indicated CPU time will be the sum of all components for system address spaces, and TCB plus pre-SRB for requestors. Also application TCB in examples will include pre-SRB.

There are a few basic considerations to keep in mind. A high percentage of DB2 CPU time is application, which is reported with the requestor address space. Recall the requestor for server requests is DIST. There are a variety of DB2 system functions, which can be associated with TCB or SRB time in a specific address space (MSTR, DBM1, or IRLM). Typically the only significant value is DBM1 SRB, which is primarily from asynchronous database I/O (database I/O counts are also reported with DBM1). Following is a discussion of several useful data sources with primary focus on CPU time.

RMF

This is often the best source for system address space totals and service class breakdowns. It is easy to observe and track DB2 system functions using R723CCPU and R723CSRMB.

If there is a significant server workload there will probably be separate service classes for the (SPAS) enclaves, so the DIST system address space will be reported separately. Unfortunately the portion of server application CPU time that is not stored procedure is included with DIST SRB, because it really is SRB time. That makes it difficult to accurately measure system DIST time, which fortunately is not typically a significant value.

Report classes can effectively be used for reporting transaction information with flexible grouping rules if the transactions are from CICS, IMS, or a remote system (DDF). Transaction rates and response times can easily be reported for desired groups. For a transaction service class, valuable delay and state metrics are also provided. Resource consumption (CPU, I/O, paging, and memory) is not available for transaction group level data. It is only reported with the requestor address space classes.

For a pure distributed environment, RMF might be adequate. It would easily show all CPU time but workload reporting would not be granular. Workload reporting of transaction counts and response time could be accomplished with report classes.

SMF 30

These can be useful, but granularity is probably a problem. All resource consumption is included (not just DB2). For CICS and IMS it is not possible to report logical workloads, unless region totals are adequate. For batch and TSO this could be a desirable source. Even though resource consumption for other than DB2 processing is included, there is a value to totals by workload. There is only one SMF 30 per address space so enclaves for server work are included with DIST system values. Following are specific CPU time fields of interest:

- SMF30CPT - TCB plus ASR, ENC, & DET
- SMF30SRB - traditional SRB
- SMF30ASR - pre-SRB but not enclave
- SMF30ENC - independent enclave pre-SRB
- SMF30DET - dependent enclave pre-SRB

Note that CPT is logically like "old" TCB time because it includes all pre-SRB time.

SMF 42-6

These records report activity and performance metrics by a job to a dataset. While beyond the scope of this paper, it can be a good data source for database activity and performance reporting.

SMF 100 - System

These are interval records produced for each DB2 SSID. Prior to DB2 V7, they were difficult to synchronize with other SMF interval records. Recording options are controlled with the DB2 DSNTIPB installation panel. Most data is not actually interval but is either a current reading (e.g. active buffers) or a
A wealth of data is available but comparatively little is of direct interest to the traditional analyst. It does include CPU times (TCB and SRB) for DB2 system functions by address space. The SMF 100 IFCID 0001 record contains a QWSA segment for each system address space. Fields of interest are:

- **QWSAPROC**: Identifier of MSTR, DBM1, IRLM, or DIST
- **QWSAEJST**: TCB time
- **QWSASRBT**: SRB time

Note that DIST QWSASRBT includes server application CPU time that is not stored procedure.

### SMF 101 - Accounting

There is loosely one SMF 101 IFCID 0003 record produced for each DB2 transaction. For many types of work (e.g. CICS and IMS) there might be close to a one-to-one relationship with what a user considers a transaction, but it does depend on installation options and programming conventions. A long batch step or TSO session could be recorded in a single record. For parallel processing a record might be recorded for each parallel child task, though typically rollup reporting would aggregate all the children into one record. Starting with DB2 V8, there is an option to rollup RRSAF and DDF transactions for a specified set of identifier fields. There is no interval recording.

Two primary sets of data are provided - application (Class 1) and “In DB2” (Classes 2 and 3). Application includes establishing the connection with DB2 and might include other processing. The term In DB2 is used to refer to activities within DB2 to satisfy application requests. In DB2 Class 2 elapsed and CPU values are a subset of application measurements. Class 1 includes most activity counts (e.g. SQL and buffer pool requests) and Class 3 includes most delays (e.g. I/O and locks).

Some logically interactive work might look like long transactions. RRSAF, which is used by SAP, does not provide logical transaction end prior to DB2 V8.

A second set of accounting data might be available – IFCID 0239 package / DBRM. A package is essentially a precompiled DB2 program. One or more might be used during execution of a transaction. If activated, there will be a QPAC segment for each different package used. Depending on DB2 level and setup these segments might be in separate SMF 101 records. Two types of data are available. Class 7 is In DB2 accounting and Class 8 is delay information. These can provide valuable additional breakdowns of metrics. There is a set of identifiers in QPACPKNM and QPACAAFG is a type of activity flag. Since not all transactions use packages and a transaction might use multiple packages, summarized package data is unlikely to cross-foot with other data.

If rollup records (QWACPARR will be set) are involved and IFCID 0239 is in separate SMF 101, it is necessary to synchronize with the IFCID 0003 record in order to know how many transactions might have been involved. Correlation headers (QWHC) will be identical. Note that you cannot assume that all transactions used all packages.

While much of the accounting data is primarily of interest to the DB2 specialist, many valuable metrics are available. Given post processing code and workload grouping rules, this can be an excellent data source. Specific SMF 101 fields will be discussed in subsequent sections.

### SMF 102 – Performance

There is a large quantity of detail data that can optionally be recorded. A reasonable parallel is to think of it as being GTF for DB2. It does not contain any CPU metrics and will not be discussed further.

### Requestor Measurements

This category includes transaction data from subsystems like CICS and IMS. These sources can be especially good for response times and transaction rates because they might be closer to the user view of a transaction (a CICS or IMS transaction might invoke what looks like multiple DB2 transactions).

With recent levels, CICS Monitoring Facility (CMF) SMF includes several useful DB2 related fields. DB2 CPU time is reported both separately in CMF 259 L8CPUT and included in total CMF 008 USRCPUT, both of which are SMF 110 transaction detail fields. Note that while L8CPUT for DB2 CICS workloads is essentially DB2 CPU, it will include CPU time for any application code using the CICS OPENAPI option.
IMS includes DB2 CPU with application total (DLRTIME) in 07 IMS log accounting record. Parallel child time is not included. Unfortunately IMS log records are difficult to work with for performance reporting. There is no IMS SMF recording. IMS monitors would be useful.

**KEY PERFORMANCE METRICS**

The discussion will now turn to the performance metrics of interest – CPU, transaction counts, and response times. The principal data sources that will be discussed further are RMF and the primary DB2 SMF records (100 & 101), though some other specific fields will be referenced.

**System CPU Usage**

These values are fairly easily reported and tend to be comparatively small in most environments. Recall that DBM1 SRB is typically the only significant value. If any other TCB or SRB time for the primary address spaces (MSTR, DBM1, IRLM) is high or changes significantly, consult your DB2 specialist. RMF report classes are a logical approach for reporting, though SMF 30 and 100 are also good sources.

Depending on the data source, DIST values might be high if there is a large server workload. Remember that DIST is the requestor. RMF produces the easiest to use totals because the enclaves (where most application time will be reported) should be in separate service classes from the DIST address space. Since DDF application time that is not stored procedure is included with DIST SRB (recall it really is SRB time), DIST system times will be DIST TCB and the rest of

**Application CPU Usage**

Most DB2 CPU time is application TCB and pre-SRB and is included with requestor totals. As already mentioned, the best source of DB2 CPU time by workload is SMF 101. Application SRB time cannot be reliably measured so it is not recorded. Fortunately it would not be a very significant value. There are many fields of interest depending on environment. The following IFCID 0003 Class 1 fields must be added to calculate application total:

- QWACEJST - QWACBJST: Home AS End – Start
- QWACSPC: Stored Procedure
- QWACTRTT: Trigger not enclave
- QWACTRTE: Trigger enclave
- QWACUDCP: User defined function

In many environments without DDF work, only the first value will be significant. Server workloads typically use stored procedures so QWACSPC might be large.

The following IFCID 0003 Class 2 fields must be added to determine total In DB2 CPU time. The values are subsets of (but generally not much smaller than) the corresponding application total values.

- QWACAJST: Home AS
- QWACSPTT: Stored Procedure
- QWACTRTT: Trigger not enclave
- QWACTRTE: Trigger enclave
- QWACUDTT: User defined function

The trigger values are not different in the second group and are reported when Class 1 is active. For rollup records use QWACEJST without subtracting QWACBJST.
Figure 3 adds total application CPU to Figure 2. Notice the other values are now dwarfed. A different view of application total is shown in Figure 4. Those times are totaled by type of transaction (based on values of QWHCCN and QWHCATYP), which obviously varies with installation. Using different grouping identifiers, the same chart could have been produced by application or department.

Server Application CPU Usage

As already suggested, analysis of a DB2 that is a server for remote requests presents a bit more complexity. Since at least some application time will be included with system time (remember the amount and where will vary with data source) and all application time will be in SMF 101, there is the potential of double counting or missing something. Figures 5-7 are from an environment with a significant percentage of server work. Figure 5 is based on SMF 30 data. It shows CPU totals (SMF30CPT plus SMF30SRB) for the system address spaces. The largest values are for DIST, which includes all application time.
Figure 6 is based on SMF 100 and 101. The system values show TCB and SRB separately. DIST APPL TCB is the sum of all application CPU fields from SMF 101 for all server transactions. To avoid double counting, the portion of application time that was not stored procedure (SMF 101 Class 1 CPU fields except for QWACSPCP) was subtracted from DIST SRB so it would be net system value. Notice how small the net DIST values are. Also notice how close the corresponding values of the two charts are to each other (they would have been closer if SMF intervals for all sources had been synchronized).

APPL DIST TCB is broken down into logical components in Figure 7. The top and bottom values were the ones double reported (in SMF 101 and with DIST SRB). The two stored procedure values (SP BASE and SP IN DB2) were SPAS enclaves. Specific field sources are:

- **HOME:** QWACEJST-QWACBJST-QWACAJST
- **SP BASE:** QWACSPCP - QWACSPPT
- **IN DB2:** QWACSPPTT
- **IN DB2 (NOT SP):** QWACAJST

### SAP CPU Times

Prior to DB2 V8, it was not possible to report SAP with reasonable workload granularity because the connections used looked like long running transactions. However, it was still possible to do some useful total and average analysis. One big difference compared to many other workloads is that SAP uses DB2 primarily as a database manager, thus the ratio of DB2 reported application time to system time tends to be lower than other environments.

### Transaction Rates

There are numerous choices on what data source to use for transaction counts including RMF, SMF 101, and requestor measurements. A given source often offers multiple metric choices. The definition of a transaction varies with data source and analysis objective. Depending on objective, the best sources might be RMF (service and report classes) or requestor data (e.g. CICS). In cases where a logical transaction invokes multiple DB2 transactions, a decision must be made. If the objective is service level or end user oriented, it can be attractive to work with requestor or RMF data. For DB2 analysis or reporting granularity, SMF 101 data is very useful.

While there is loosely one SMF 101 for each DB2 transaction, there is considerable variation. Multiple records are produced for parallel transactions. While you might only consider parents when counting transactions, the child records are necessary for complete resource consumption analysis. There might be rollup records for parallel children, DDF, or RRSAF. There are several control and count fields to consider when counting transactions. The following table lists some of the important relationships.

<table>
<thead>
<tr>
<th>Record Type</th>
<th>QWACPARR</th>
<th>QWACPCNT</th>
<th>QWACPACE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Normal</td>
<td>OFF</td>
<td>ZERO</td>
<td>ZERO</td>
</tr>
<tr>
<td>Parallel Parent</td>
<td>OFF</td>
<td>CHILD</td>
<td>COUNT</td>
</tr>
<tr>
<td>Parallel Child</td>
<td>OFF</td>
<td>ZERO</td>
<td>TOKEN</td>
</tr>
<tr>
<td>Parallel Rollup</td>
<td>SET</td>
<td>CHILD</td>
<td>TOKEN</td>
</tr>
<tr>
<td>Simple Rollup</td>
<td>SET</td>
<td>TRAN</td>
<td>COUNT</td>
</tr>
</tbody>
</table>

DB2 V8 added the option to provide logical end of transaction recording to take place at commit time. This allows work that is logically interactive, but previously appeared to be long running, to be reported in a more useful manner. Logically changing identifiers are also recognized and reported in the correlation header (QWHC). For much work a commit is really a logical transaction. Current levels of SAP, which uses RRSAF, can benefit from this. Most interactive work (e.g. short CICS and DDF) has only one commit per transaction; however, for some long running work it can be useful to use commit count (QWACCOMM) as tran count. Even prior to V8 that approach could help with high level SAP analysis, though the primary reporting problem was lack of identifier granularity.

In many environments there will not be significant rollup or long running transaction issues, in which case each SMF 101 IFCID 0003 record can be counted as a transaction. An important consideration to determining tran count is that it will probably be needed to calculate response time or average resource use.

QPAC (package / DBRM) data does not contain a logical transaction count so the count is effectively how many transactions used the package. Rollup records can make this complex because QPAC segments might be in SMF 101 IFCID 0239 records separate from other application data. It is necessary to match with the IFCID 0003 record in case rollup was involved. You will then know max potential count because not all of the rollup transactions necessarily used all reported packages.

### Response Times

This is a key metric to use when reporting how effectively service is delivered to a workload but it...
should be reported with a workload level (e.g. transaction rate). Data sources are the same as for rates (above). While it often valuable to have service level response reporting use requestor measurements, the focus here is DB2 measurements. Again there are two primary sets of times – application and In DB2.

Application time is calculated by subtracting QWACBSC (begin) from QWACESC (end). If it is a rollup record do not subtract QWACBSC because QWACESC is the desired accumulation. Average will obviously require division by the transaction count. While application time is useful for some purposes, it often includes wait for work (e.g. CICS and RRSAF).

While they are not necessarily all always present, the following values must be added to calculate In DB2 time:

- QWACASC: Home AS
- QWACSPEB: Stored Procedure
- QWACTRET: Trigger not enclave
- QWACTREE: Trigger enclave
- QWACUDEB: User defined function

The first will always be present when In DB2 values are recorded while the others depend on environment.

Package elapsed time is QPACSCT.

Response Components

There are numerous response components recorded. These are actually parts of response time, which can be used to concentrate effort or money if response objectives are not being met. Included are service times (e.g. CPU), service delays (e.g. DASD I/O), and exception condition delays (e.g. wait for a lock). RMF reports useful WLM state and delay data, but the best source is DB2 accounting records. Newer levels of DB2 add progressively more processing function and related reporting capabilities. DB2 data tends to have good resolution with the sum of the parts being close to elapsed for high priority work.

Be careful when working with parallel transactions. While analysis of parent records will show overall transaction response, child records will provide useful resource consumption and delay information.

Figure 8 is an In DB2 response profile for the average short transaction in a CICS and DDF environment. The chart shows all significant metrics for the environment. SP TCB (stored procedure) is primarily from the DDF transactions. DASD WAIT (QWACAWTI) is wait for database I/O. OTHER THREAD READ (QWACAWTR) is delay due to read I/O performed by another thread. Incomplete prefetch, which is actually performed by DBM1, is the probable reason. OTHER THREAD WRITE (QWACAWTW) is delay due to write I/O performed by another thread. DDF WAIT (QLACCPUL) is delay due to a DDF request sent to another system. OTHER WAIT is the sum of many reported delay and exception conditions, which would not easily fit on this chart. When significant they should be looked at separately. OTHER is the sum of all the components subtracted from response time. Assuming no significant metrics were missed, it typically represents CPU queuing.

A breakdown of OTHER WAIT is shown in Figure 9. It does not include all potential fields. Sources of those shown are:

- SERVICE: QWACAWTE
- QUIESCE: QWACALOG
- LOCK LATCH: QWACAWTL
- DRAIN LOCK: QWACAWDR
- CLAIM: QWACAWCL
- ARCHIVE: QWACAWAR
- PAGE LATCH: QWACAWTP
- GLOBAL MESSAGE: QWACAWTG
- GLOBAL LOCK: QWACAWTJ

There were apparently some workload and environment changes during the periods shown. Breakdowns by workload would be useful to learn more about the changes. Then consultation with a DB2 specialist would be appropriate if it is felt that a problem exists.
Response Profile Choices

There are two basic profile choices – application and IN DB2. Appropriateness will vary with workload and analysis objective. Specific metrics used could also vary with environment. Figures 10 and 11 show the two profiles for a typical CICS workload. The dominate OTHER in Figure 10 is due to wait-for-work, showing that DB2 measured application response time is essentially useless for CICS. For performance, CICS uses threads that stay connected to DB2; however, logical transaction end is recognized and unique identifiers are recorded. All metrics except application response are useful. IN DB2 profile (Figure 11) is the one to use for CICS. The early morning OTHER values are not considered significant because related transaction rate is so low.

Figures 12 and 13 show profiles for a batch workload. Both of these can be useful. IN DB2 of Figure 12 represents Figure 13 totals. Figure 12 OTHER is the combination of batch CPU and I/O outside of DB2, and CPU delay due to higher priority work or other LPARs. Notice that prime time Figure 13 OTHER is relatively larger than Figure 11 values. Since CICS typically executes at a higher priority than batch, it is likely to have less CPU delay.

The profile examples shown thus far used number of related SMF 101 records as transaction count. Figure 14 shows an example of SAP RRSASF prior to DB2 V8, which appear to be long running transactions. To get a more logical view, commit count (QWACCOMM) was used for transaction count divisor. The displayed commit rate is not very smooth because there was no attempt to spread activity over the total elapsed time; however, a reasonable view of the average SAP transaction is provided.
SUMMARY

The purpose of this document is to provide an introduction to the analysis of DB2 CPU and response metrics. While it is important to have a general understanding of how DB2 interfaces with application requests and where data is recorded, detail DB2 knowledge is not necessary. Key DB2 performance and capacity information should be reported with other installation data. Both current period and historical perspectives are important. Many important DB2 metrics for both management reporting and problem warning can be found in RMF, especially if report classes are appropriately defined. Those metrics include response times, transaction rates, and CPU time for system functions. The most important metric that cannot be easily reported with RMF is application CPU time for logical workloads. DB2 accounting records must be processed for that data. If there is significant server work, be sure to avoid double counting CPU time.

Teamwork is an important consideration. As functional areas get more specialized it is easy to get removed from reality. There must be some centralized reporting and tracking function. Data from multiple sources and tools will probably be required. There must also be enough knowledge and coordination to get specialists involved when appropriate. Most subsystems have some idiosyncrasies relative to how they interface and report. It is a matter of learning the key points about the important ones in your environment.

GLOSSARY

Allied Address Space – An AS connected to DB2 that can make DB2 requests.

Application Plan - Effectively a transaction definition. The control structure that is produced during the bind process

EDM Pool - Environmental descriptor management pool. A pool in memory used for database descriptors, application plans, and packages.

Enclave - One or more units of work running in the same or multiple address spaces. Effectively a virtual AS.

In DB2 - Application processing within DB2.

Package - A set of statically bound SQL statements that is available for processing.

Preemptible SRB - TCB like but less expensive to create.

Requestor - Source of “transaction” for accounting purposes.

RRSAF - Recoverable Resource Manager Services Attachment Facility. The DB2 facility that uses MVS Transaction Management and Recoverable Resource Manager Service.

Stored procedure - User-written DB2 application program that can be invoked through use of the SQL CALL statement.

Server - Target of requests from a remote system.

Thread - Connection between DB2 and requesting AS.

Trigger - A set of SQL statements that is stored in a DB2 database and executed when a certain event occurs in a DB2 table.
User-defined function (UDF) - A function that is defined to DB2 and can be referenced in SQL statements.

**DB2 RECORD LAYOUTS**

SMF records layouts are produced by assembling macros located in DB2xxx.SDSNMACS, where xxx is the installed DB2 level. The following will provide layouts for the records discussed:

SMF 100 IDCID 001 System services:
DSNDQWST SUBTYPE=0

SMF 100 IDCID 001 System service:
DSNDQWST SUBTYPE=1

SMF 101 IFCID 0003 & 0239 Accounting:
DSNDQWAS SUBTYPE=ALL

Additional field description information is in DSN810.SDSNIVPD(DSNWMSGS) for V8, and DSNxxx.SDSNSAMP(DSNWMSGS) for older releases. All time fields from these records referenced in this paper are in 8-byte TOD-clock format.

**ACKNOWLEDGEMENTS**

The following trademarks have been used. Any omissions are unintentional:

IBM
RMF
z/OS
MVS
DB2
CICS
IMS
VTAM
SAP
Windows
UNIX
Linux

Peter Enrico, “Focus Enclaves”, Cheryl Watson’s Tuning Letter 1999, N0.2

John Arwe, “Preemptible SRBs”, Proceedings, CMG95, p646

Chuck Hoover, “Tuning DB2 From The Ground Up”, Share Feb 98 Proceedings, Session 1340


Barry Merrill, “Captured DB2 CPU Time”, Cheryl Watson’s Tuning Letter 2005, N0.1 & MXG Listserv December 2004

System Management Facilities, IBM

Resource Measurement Facility Report Analysis, IBM

DB2 Administration Guide, IBM

DB2 Installation Guide, IBM

DB2 PM Report Reference, IBM

PerfMan® for z/OS, The Information Systems Manager, Inc. (ISM)

PerfMan® for DB2, The Information Systems Manager, Inc. (ISM)

CMG Proceedings
Share Proceedings
IDUG Proceedings
Google

**REFERENCES**