z/OS Workload Manager and CICS\TS MXT

March 9, 2006 **By: Eugene S. Hudders**

CICS/TS Performance Tuning Using C\TREK Course

This seminar covers a lot of tuning material and will require 8 hours a day. The total time for the seminar is 40 hours.

The followings topics included in this course are:

- **Introduction to CICS Performance Tuning**
- **Using Operating System Information to Tune CICS/TS**
- **Tuning CICS/TS Processor Cycles**
- **Tuning Real Storage in CICS/TS**
- **Tuning Virtual Storage in CICS/TS**
- **Tuning CICS/TS Transaction Controls**
- **Tuning On Line VSAM Files**
- **Tuning NSR Files in CICS/TS**
- **Tuning CICS/TS LSR Buffer**
- **Tuning the Index CISZ**
- **Reviewing VSAM Free Space**
- **Reviewing the DB2 Interface**

If you wish to attend this course, please send us an email at ctrek@actpr.com

CICS/TS being an address space within the z/OS system is subject to analysis by the Workload Manager (WLM) to determine the dispatching priority to be assigned to the region. The two basic options available to the z/OS system programmer to assign the operating system dispatching priority for CICS/TS is to either use region controls through the use of velocity goals or transaction controls through the use of average response time or percentile goals. Transaction goals provide the CICS/TS the capacity to be able to obtain an operating system dispatching priority based on a response oriented objective such as 95% of the transactions have to have ½ second or less response time. This type of goals is more oriented toward Service Level Agreements (SLA) that are used in many installations today. It is important to understand that WLM adjusts the dispatching priority of the CICS/TS region and not the individual transactions running within CICS/TS.

WLM requires a control block structure that can be used to measure the individual events that occur within the CICS/TS region. These control blocks are called Performance Blocks (PB) which are x'100' bytes long (ELSQA) and a Performance Block Extension (PBDE) that are X'40' bytes long (ECSA). So, the total space required for these control blocks is just over 300 bytes. The Performance Blocks are created regardless of which WLM objective is used for CICS/TS. The storage for these control blocks comes out of z/OS storage areas not within the CICS/TS dynamic storage areas. The number of PBs is dependent on the MXT value assigned to CICS/TS. MXT refers to user tasks. So, the number of PBs assigned is equivalent to MXT + 1. If MXT is 100, then there would be a total of 101 PBs created. CICS system tasks are not included in this calculation. Instead, CICS/TS will assign an additional 20 to cover system tasks (Figure 1). However, a very large MXT may have CICS allocating 10% of the MXT value for system tasks. For example, the total number of PBs created for a region that has an MXT value of 100 would be 121 PBs (101 for user tasks and 20 for system tasks).

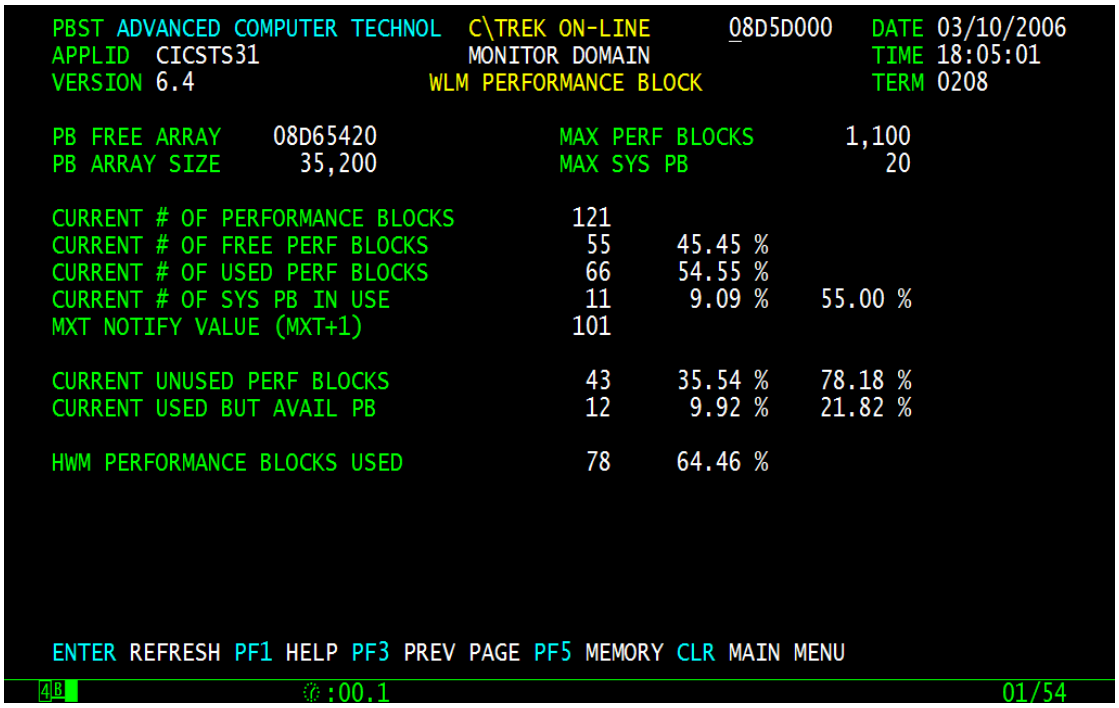

Figure 1. Workload Manager PB Summary

The amount of storage allocated is generally not a major issue but can be quite significant when using large MXT values. So, if you define an MXT value of 999, then you would be allocating almost 400 KB of z/OS storage for PBs for this CICS/TS system. Raising the MXT value via CEMT will also result in a corresponding increase of PBs. However, lowering the MXT value does not result in a reduction of PBs. Once acquired, PBs remain in the CICS/TS system until re-cycled.

The PBs are scanned occasionally by WLM. The frequency of the scan varies with the WLM objective being used. If region controls are used (velocity goals), then every PB is scanned every 2.5 seconds. If transaction controls are used such as, Average Response or Percentile goals, then the PBs are scanned every 250 milliseconds. Therefore, there is major difference in CPU overhead between the two objectives. More importantly, over allocation of the MXT value can represent wasted CPU cycles for unused PBs. Excessive MXT slots result in extra CPU use when analyzing unused PBs. It is important to note that this overhead CPU is not reflected in the CICS/TS region. The excess CPU is registered in the WLM address space. The amount of overhead is greater in installations that have many CICS address spaces.

What exactly does WLM inspect in the PB? WLM is interested when a PB completes in order to identify the appropriate response time and transaction termination. During the WLM scan, each PB contains the state associated with the transaction (Figure 2). A summary of the state information can be found in the RMF reports that provide a breakdown percentage of the type of

waits that occurred in the CICS/TS system during the reported interval. The PB state summary in the RMF report can be used to diagnose bottlenecks in the system (Figures 3 and 4). Some of the PB states are as follows:

- PB is active
- PB is inactive
- PB is waiting on an I/O or I/O activity
- PB waiting on a timer
- PB waiting on an undefined resource
- PB code is unknown

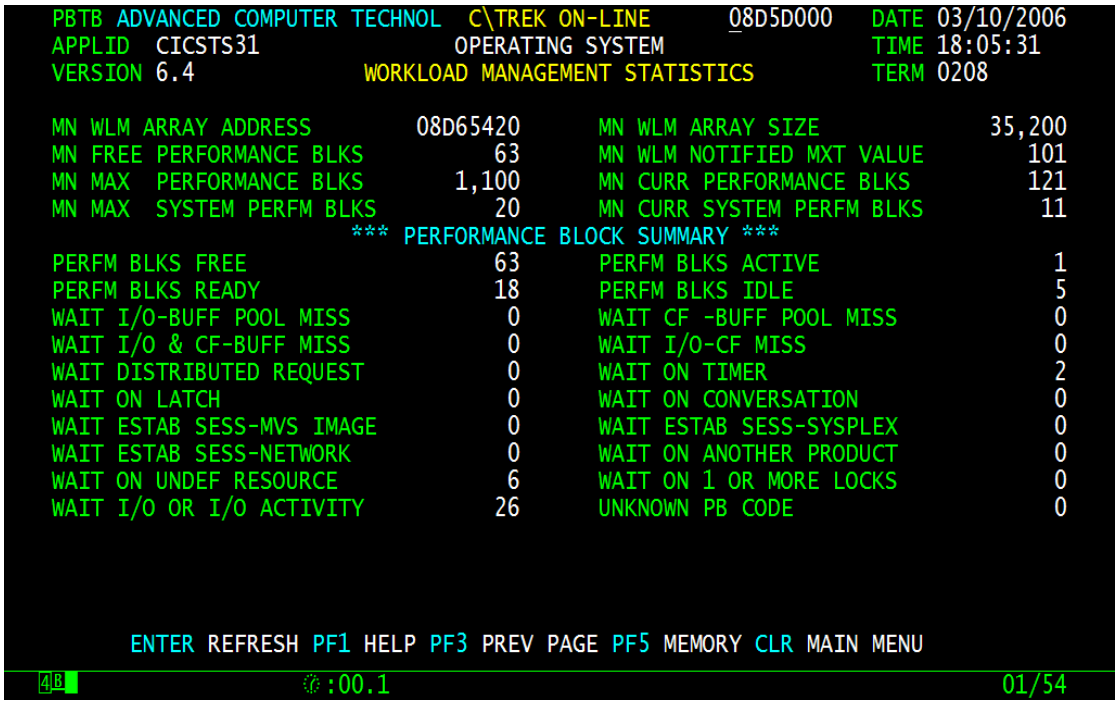

Figure 2. WLM PB Statistics

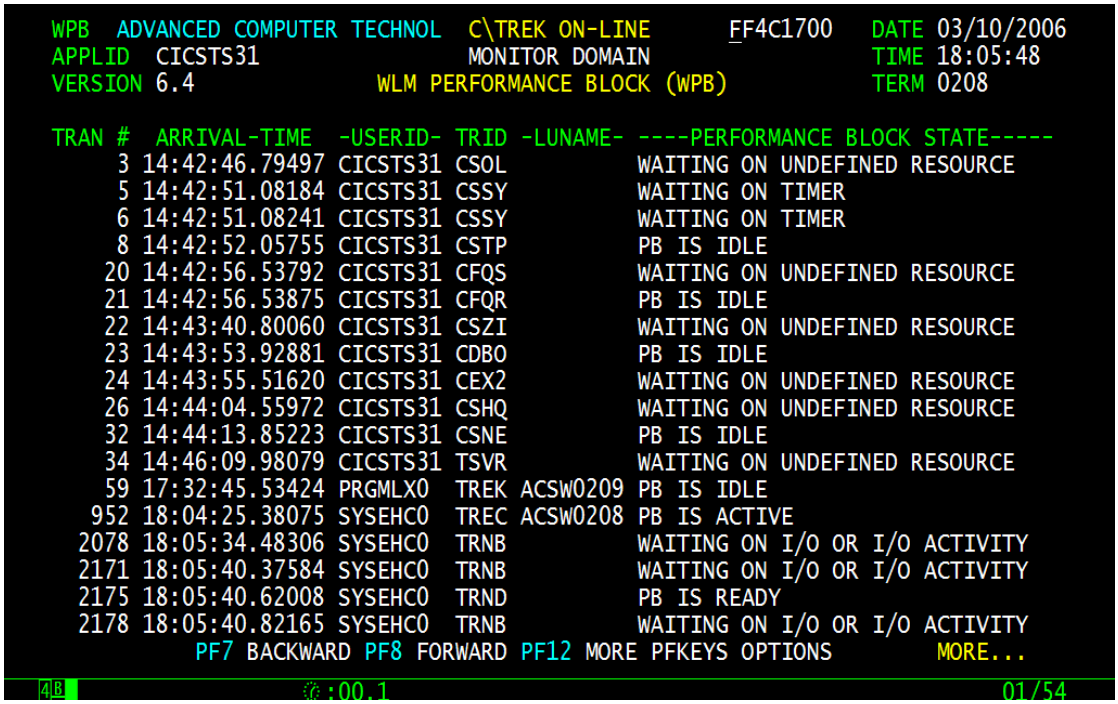

Figure 3. Task PB State

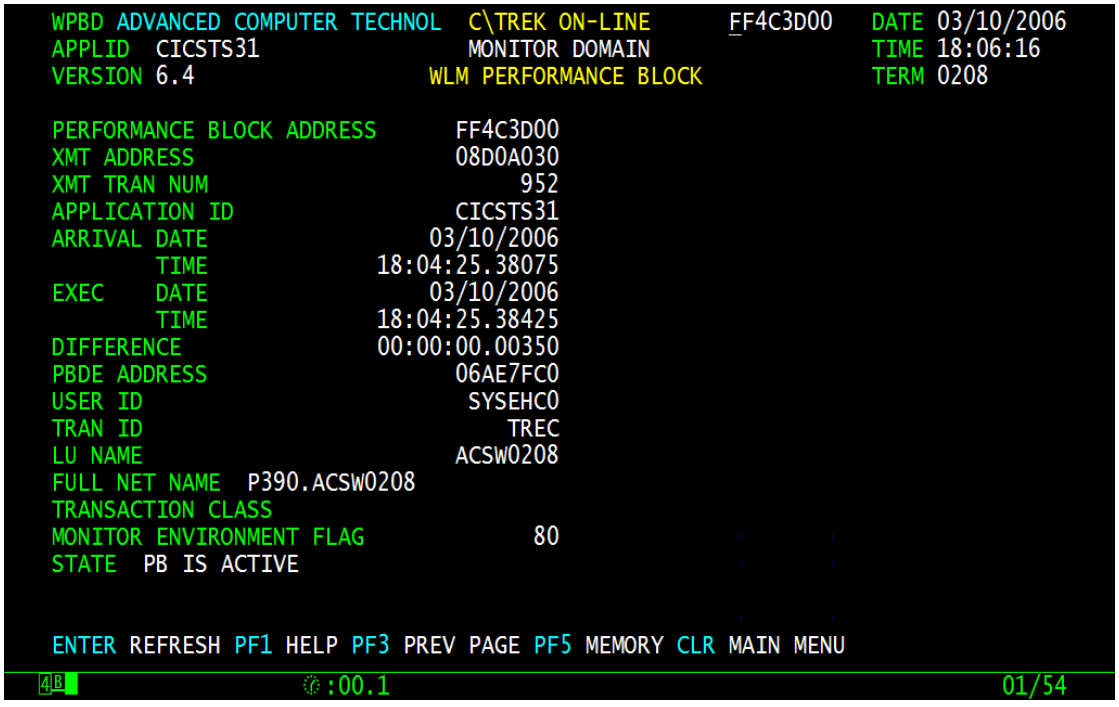

Figure 4. Task PB Detail Information

Many installations define several CICS/TS regions for each production region. For example, it is not unusual to have a test region and a quality assurance region for each production region. Many of these regions are identical in specifications, that is, the same MXT is specified for each region. This practice has a tendency of increasing the WLM overhead for the test and quality assurance regions. Unfortunately, the MXT settings may be required to ensure similar environments when testing. The price to be paid is higher CPU overhead for the test regions with a high MXT value and a low peak usage. In addition, remember that test and quality assurance regions have high periods of inactivity. WLM continues to monitor these PBs regardless of whether or not the region is inactive. It is important to note that transaction controls should not be used for test or quality assurance regions because there can be long periods of inactivity that may distort the WLM measurements and dispatch priority recommendations.

Is there a guideline for setting the MXT value? We use the information obtained regarding the number of used PBs to try an estimate an appropriate MXT value. It is just as important not to under allocate the MXT value as it is to over allocate it. Therefore, we try to allocate a MXT value where the HWM of PBs used falls in the range of 60 to 70% of the MXT allocated. The idea is to have expansion capacity to accommodate for unexpected transaction peaks and/or growth without over allocating that can result in wasted CPU cycles. The simple formula is:

% Used = (HWM PBs / MXT) * 100

If the result of the formula is a higher than 70%, then the MXT value should be increased until it falls in the range. If the result falls below 60%, then the MXT value should be adjusted downward until it falls in the range. You will need several days of MXT data in order to get a global average. This article explored the importance of setting the MXT value to a figure that does not adversely generate CPU overhead by setting the MXT value too high but at the same time does not create artificial bottlenecks by setting the MXT figure too low.

If you have any questions, comments or request for more information, please contact us:

Address: C\TREK Corporation
PO BOX 560069
MONTVERDE FL 34756

E-mail: ctrek@actpr.com

Website: www.ctrekcorp.com

Phone: (407) 469-3600 C\TREK Corporation
(787) 756-5620 Advanced Computer Technology

Fax: (787) 756-5150

Support:: (787) 397-4150
(321) 297-5838
(787) 462-0406