

How to Determine CICS Wait Time from CICS Trace

When looking at CICS performance-related issues, sometimes the CICS trace has useful information, and can be used to determine how long CICS was in a wait state, and hence how long it was actively dispatching transactions. This dispatch time is elapsed and not CPU time, although if CICS had 100% access to a CPU during the dispatch time window, it would be.

To quickly calculate wait time, two particular trace entries must be produced - DS 0022 when CICS is about to enter a wait state, and DS 0023 when the wait has completed. There are two ways to get these trace entries:

1. The better way is to install PTF UK79207 for APAR PM53981 if it is not already installed. This PTF causes the two trace entries to be produced when standard tracing level 1 is active (SIT STNTR=1), which most z/VSE customers use.
2. Use SIT override STNTRDS=(1,2) to add level 2 Dispatcher trace. This causes the trace entries to be produced because they were defined as level 2 trace before the PTF, but it adds other level 2 entries that are not useful.

You can use TR=2 output produced from a dump by DFHPD410, or FULL output produced from an auxiliary trace dataset by DFHTU410.

I use Rexx code to calculate wait time as part of a more comprehensive trace analysis, but this is an internal use program. I will show you how to find the data, and you could extract the important lines and write some simple code to calculate the time.

Important CICS Design Notes

You need to understand that CICS uses a number of subtasks to run the work. The most common of these, in reverse dispatching priority order, are:

1. RO - this does things such as loading programs.
2. QR - this runs almost all of the transaction activity, and is the one that whose wait time and dispatch times are significant.
3. SL - this handles the TCP/IP side of the Socket Listener CSOL that runs for the installed TCPIP SERVICES.
4. SO - this performs TCP/IP socket I/O.

The design was made to avoid QR becoming blocked by activities that were synchronous in nature. For example, when you ask z/VSE to load a program, the task must wait until the load has completed. Unlike in OS/390 (now called z/OS), the design of z/VSE does not allow multiple CPUs to be exploited to provide true multi-processing.

In CICS trace output, you will often see trace entries from multiple subtasks, and each subtask will generate DS 0022 and 0023 entries. You need to use only those that are related to QR, which are for "KENUM_0002". The interval between the DS 0022 and 0023 will be at most the ICV value for QR - the Partition Exit interval.

Sample CICS Trace

```
DS 0022 DSDS3  EVENT - VSE_WAIT_ENTRY           (the wait starts now)

TASK-DSTCB KE_NUM-0002 TCB-003F8000 RET-868F0162 TIME-16:43:09.2259076259 INTERVAL-00.0000438125 =000389=
 1-0000 D8D96DE2 E4C2C440                      *QR_SUBD *
 2-0000 00000000                                *.... *

KE 0201 KEDD  ENTRY - FUNCTION(ADD_DOMAIN) DOMAIN_NAME(DFHLD) DOMAIN_TOKEN(00000005) ENTRY_POINT(86909600) DOMAIN_AFFINITY(RO)

TASK-DSTCB KE_NUM-0003 TCB-003F5000 RET-868C9DC8 TIME-16:43:09.2259238134 INTERVAL-00.0000161875 =000390=
 1-0000 00480000 0000000B 00000000 00000000 BE002000 00000000 01000000 C4C6C8D3 *.....DFHL*
 0020 C4404040 00000005 86909600 00000000 00000000 00000000 00000000 *D....f.o.....*
 0040 00000002 00000000                                *..... *

. . .

DS 0023 DSDS3  EVENT - VSE_WAIT_EXIT           (the wait has completed)

TASK-DSTCB KE_NUM-0002 TCB-003F8000 RET-868F0162 TIME-16:43:09.2513588447 INTERVAL-00.0002238125 =000524=
 1-0000 D8D96DE2 E4C2C440                      *QR_SUBD *
 2-0000 40000000                                *... *
```

The first thing to note is that the QR DS 0022 and 0023 trace entries were not adjacent in the output. There was RO load activity in the time gap (showing that RO really is a lower dispatching priority). That does not mean that they are never adjacent.

The wait time in this case is:

16:43:09.2513588447 - 16:43:09.2259076259 = 0.0254512188 seconds.

There will be a small amount of rounding due to the TOD clock units in use by the cpu.

The wait time is not accounted to a CICS task number, it maps to the dispatcher, and hence CICS itself.

Do NOT use the INTERVAL values, this is the elapsed time between adjacent trace entries.

The TCB address is the OS/390 TCB not the z/VSE TCB.