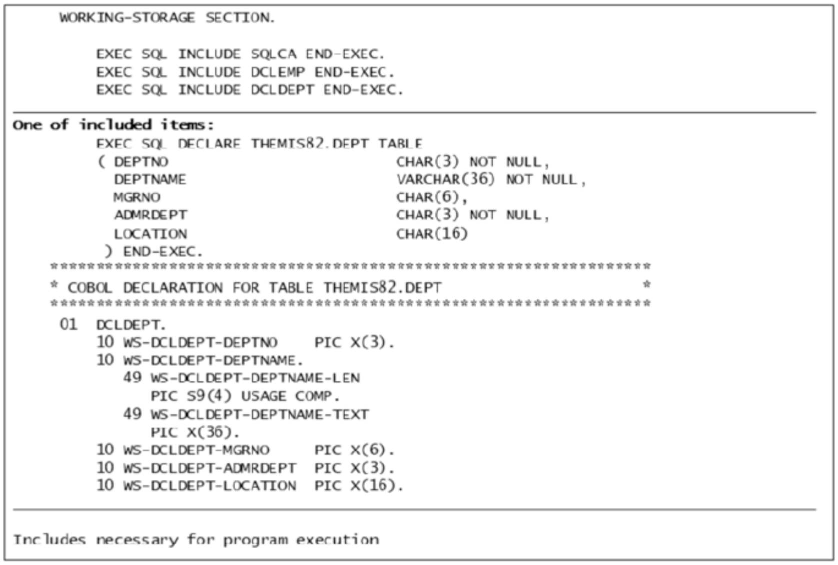
Virtual CICS user group: Newsletter 48

Welcome to the Virtual CICS user group newsletter. The Virtual CICS user group at www.fundi.com/virtualcics is an independently-operated vendor-neutral site run by and for the CICS user community.

Virtual CICS user group presentation

The latest webinar from the Virtual CICS user group was entitled, “The CICS/Db2 interface”, and was presented by Tom Dunlap, CTO at Themis.



```
     WORKING-STORAGE SECTION.

         EXEC SQL INCLUDE SQLCA END-EXEC.
         EXEC SQL INCLUDE DCLEMP END-EXEC.
         EXEC SQL INCLUDE DCLDEPT END-EXEC.
```

One of included items:

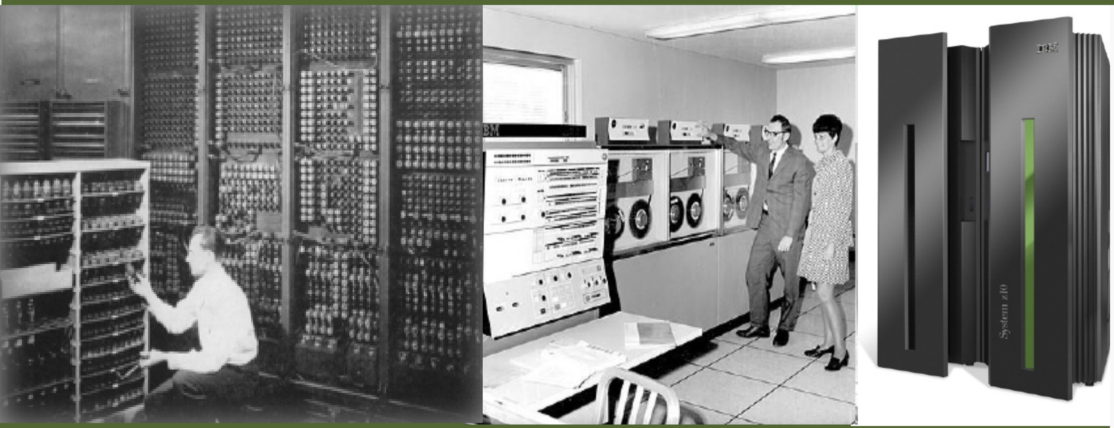
```
         EXEC SQL DECLARE THEMIS82.DEPT TABLE
         ( DEPTNO                         CHAR(3) NOT NULL,
           DEPTNAME                       VARCHAR(36) NOT NULL,
           MGRNO                          CHAR(6),
           ADMRDEPT                       CHAR(3) NOT NULL,
           LOCATION                       CHAR(16)
          ) END-EXEC.
    *******************************************************************
    * COBOL DECLARATION FOR TABLE THEMIS82.DEPT                       *
    *******************************************************************
     01  DCLDEPT.
         10 WS-DCLDEPT-DEPTNO    PIC X(3).
         10 WS-DCLDEPT-DEPTNAME.
            49 WS-DCLDEPT-DEPTNAME-LEN
               PIC S9(4) USAGE COMP.
            49 WS-DCLDEPT-DEPTNAME-TEXT
               PIC X(36).
         10 WS-DCLDEPT-MGRNO     PIC X(6).
         10 WS-DCLDEPT-ADMRDEPT  PIC X(3).
         10 WS-DCLDEPT-LOCATION  PIC X(16).
```

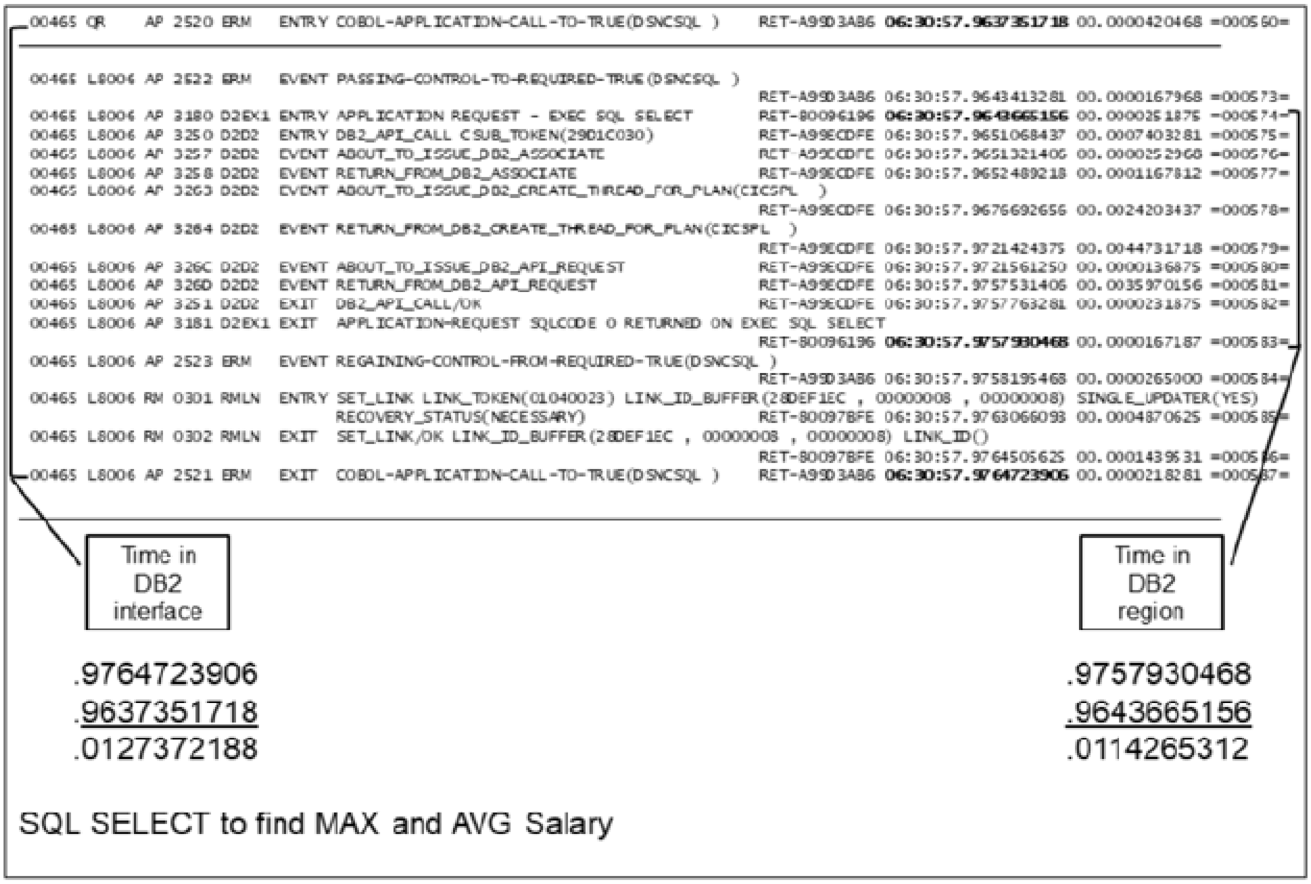
Includes necessary for program execution

Figure 1: DB2 SQL commands in COBOL programs

Tom has been working with CICS since its first release in April 1969. Throughout his career with CICS, he has developed applications, worked as a system programmer and administrator, and lead architect for commercial software products that worked in CICS, and provided both applications and systems-level training to many corporations. The extent of his coding exposure in CICS includes, COBOL, Assembler, PL/I, C, and Java. Beyond his CICS background, he has worked with z/OS (and its predecessors back to OS MFT) and many related components at both the applications and systems levels. This operating system experience includes efforts as a lead architect for commercial software products. Also included during his career is extensive work with DB2, performance and capacity planning efforts, large

Contents:





```
00465 QR    AP 2520 ERM   ENTRY COBOL-APPLICATION-CALL-TO-TRUE(DSNCSQL )    RET-A99D3AB6 06:30:57.9637351718 00.0000420468 =000560=

00465 L8006 AP 2522 ERM   EVENT PASSING-CONTROL-TO-REQUIRED-TRUE(DSNCSQL )
                                                                          RET-A99D3AB6 06:30:57.9643413281 00.0000167968 =000573=
00465 L8006 AP 3180 D2EX1 ENTRY APPLICATION REQUEST - EXEC SQL SELECT     RET-80096196 06:30:57.9643665156 00.0000251875 =000574=
00465 L8006 AP 3250 D2D2  ENTRY DB2_API_CALL CSUB_TOKEN(29D1C030)         RET-A99ECDFE 06:30:57.9651068437 00.0007403281 =000575=
00465 L8006 AP 3257 D2D2  EVENT ABOUT_TO_ISSUE_DB2_ASSOCIATE              RET-A99ECDFE 06:30:57.9651321406 00.0000252968 =000576=
00465 L8006 AP 3258 D2D2  EVENT RETURN_FROM_DB2_ASSOCIATE                 RET-A99ECDFE 06:30:57.9652489218 00.0001167812 =000577=
00465 L8006 AP 3263 D2D2  EVENT ABOUT_TO_ISSUE_DB2_CREATE_THREAD_FOR_PLAN(CICSPL  )
                                                                          RET-A99ECDFE 06:30:57.9676692656 00.0024203437 =000578=
00465 L8006 AP 3264 D2D2  EVENT RETURN_FROM_DB2_CREATE_THREAD_FOR_PLAN(CICSPL  )
                                                                          RET-A99ECDFE 06:30:57.9721424375 00.0044731718 =000579=
00465 L8006 AP 326C D2D2  EVENT ABOUT_TO_ISSUE_DB2_API_REQUEST            RET-A99ECDFE 06:30:57.9721561250 00.0000136875 =000580=
00465 L8006 AP 326D D2D2  EVENT RETURN_FROM_DB2_API_REQUEST               RET-A99ECDFE 06:30:57.9757531406 00.0035970156 =000581=
00465 L8006 AP 3251 D2D2  EXIT  DB2_API_CALL/OK                           RET-A99ECDFE 06:30:57.9757763281 00.0000231875 =000582=
00465 L8006 AP 3181 D2EX1 EXIT  APPLICATION-REQUEST SQLCODE 0 RETURNED ON EXEC SQL SELECT
                                                                          RET-80096196 06:30:57.9757930468 00.0000167187 =000583=
00465 L8006 AP 2523 ERM   EVENT REGAINING-CONTROL-FROM-REQUIRED-TRUE(DSNCSQL )
                                                                          RET-A99D3AB6 06:30:57.9758195468 00.0000265000 =000584=
00465 L8006 RM 0301 RMLN  ENTRY SET_LINK LINK_TOKEN(01040023) LINK_ID_BUFFER(28DEF1EC , 00000008 , 00000008) SINGLE_UPDATER(YES)
                                RECOVERY_STATUS(NECESSARY)                RET-80097BFE 06:30:57.9763066093 00.0004870625 =000585=
00465 L8006 RM 0302 RMLN  EXIT  SET_LINK/OK LINK_ID_BUFFER(28DEF1EC , 00000008 , 00000008) LINK_ID()
                                                                          RET-80097BFE 06:30:57.9764505625 00.0001439531 =000586=
00465 L8006 AP 2521 ERM   EXIT  COBOL-APPLICATION-CALL-TO-TRUE(DSNCSQL )    RET-A99D3AB6 06:30:57.9764723906 00.0000218281 =000587=
```

.9764723906
.9637351718
.0127372188

.9757930468
.9643665156
.0114265312

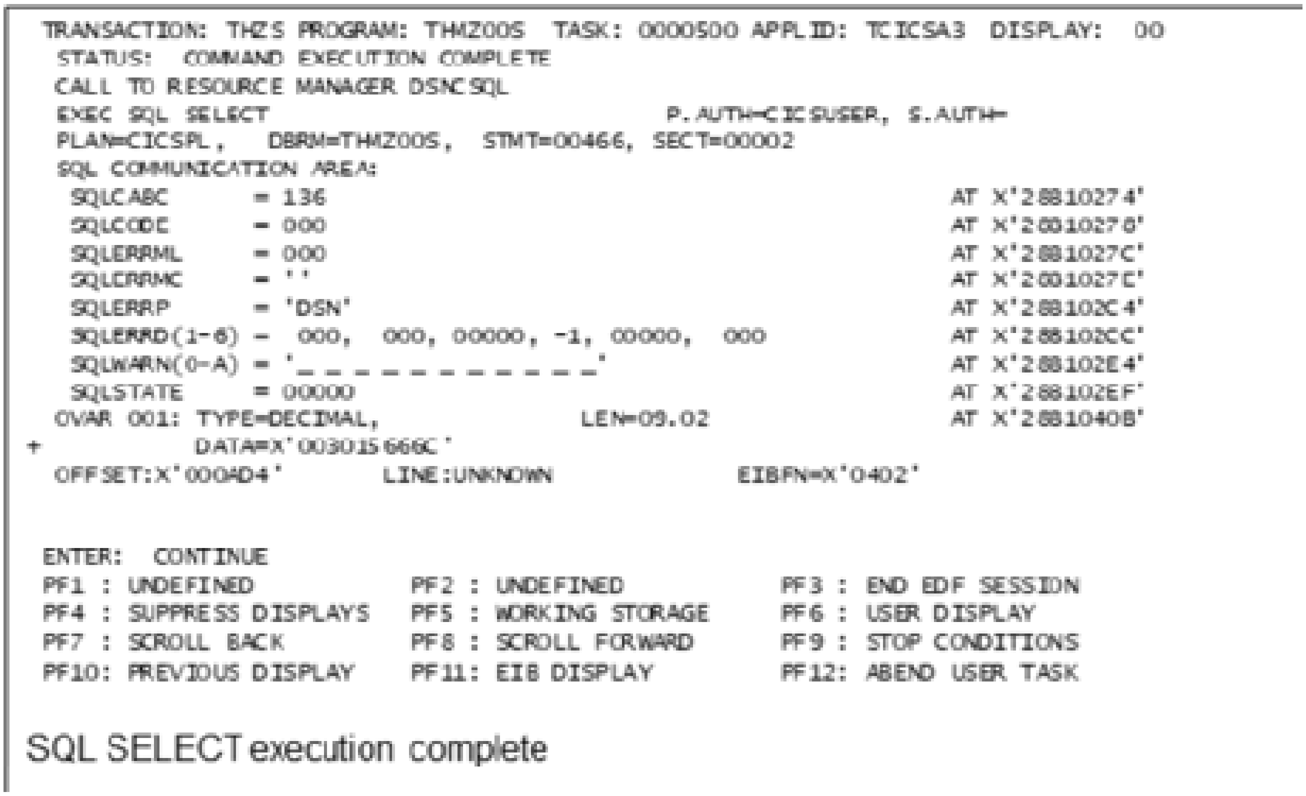
SQL SELECT to find MAX and AVG Salary

Figure 2: Trace for DB2 commands in applications

system integration efforts, and problem solving for large corporate application and systems environments. For the past 20 years he has also been involved with work on distributed environments which include UNIX, Linux, and Windows. These efforts include working with products like TXSeries (CICS on distributed systems), DB2, IBM MQ, plus many other supportive products. These efforts also include the use of application development tools like IBM RAD and IDz, CICS Explorer, DB2 Data Studio, plus the IBM Problem Determination Tools.

Tom Dunlap started his presentation with a look at application programs. He said that each application program that executes using SQL statements requires a few basic inclusions of generated code. The DECLARE statements have been generated using the DCLGEN option of the DB2 Interactive panels. The SQLCA is distributed as part of the DB2 product libraries. The DCLEMP and DCLDEPT includes represent the DB2 tables being used by the application. The prime reason for using the DB2 INCLUDE statements is due to the SQL preprocessor being invoked before the COBOL compile. If you utilize the integrated DB2 processor, as they do, you could use a standard COPY statement in place of the INCLUDE.

Figure 1 includes one of the included DECLARE statements. This SQL statement is required to provide the table name and column names for use by the application. This statement also provides the DB2 data type definitions for the columns of the table. Also, as part of the DCLGEN



```
TRANSACTION: THZS PROGRAM: THMZ005  TASK: 0000500 APPLID: TCICSA3  DISPLAY:  00
 STATUS:  COMMAND EXECUTION COMPLETE
 CALL TO RESOURCE MANAGER DSNCSQL
 EXEC SQL SELECT                              P.AUTH=CICSUSER, S.AUTH=
 PLAN=CICSPL,   DBRM=THMZ005,  STMT=00466, SECT=00002
 SQL COMMUNICATION AREA:
  SQLCABC      = 136                                               AT X'28810274'
  SQLCODE      = 000                                               AT X'28810278'
  SQLERRML     = 000                                               AT X'2881027C'
  SQLERRMC     = ''                                                AT X'2881027E'
  SQLERRP      = 'DSN'                                             AT X'288102C4'
  SQLERRD(1-6) =  000,  000, 00000, -1, 00000,  000                AT X'288102CC'
  SQLWARN(0-A) = '_ _ _ _ _ _ _ _ _ _ _'                           AT X'288102E4'
  SQLSTATE     = 00000                                             AT X'288102EF'
 OVAR 001: TYPE=DECIMAL,             LEN=09.02                     AT X'2881040B'
+          DATA=X'003015666C'
 OFFSET:X'000AD4'       LINE:UNKNOWN            EIBFN=X'0402'


ENTER:  CONTINUE
PF1 : UNDEFINED          PF2 : UNDEFINED          PF3 : END EDF SESSION
PF4 : SUPPRESS DISPLAYS  PF5 : WORKING STORAGE    PF6 : USER DISPLAY
PF7 : SCROLL BACK        PF8 : SCROLL FORWARD     PF9 : STOP CONDITIONS
PF10: PREVIOUS DISPLAY   PF11: EIB DISPLAY        PF12: ABEND USER TASK
```

SQL SELECT execution complete

Figure 3: CEDF screen for DB2 commands

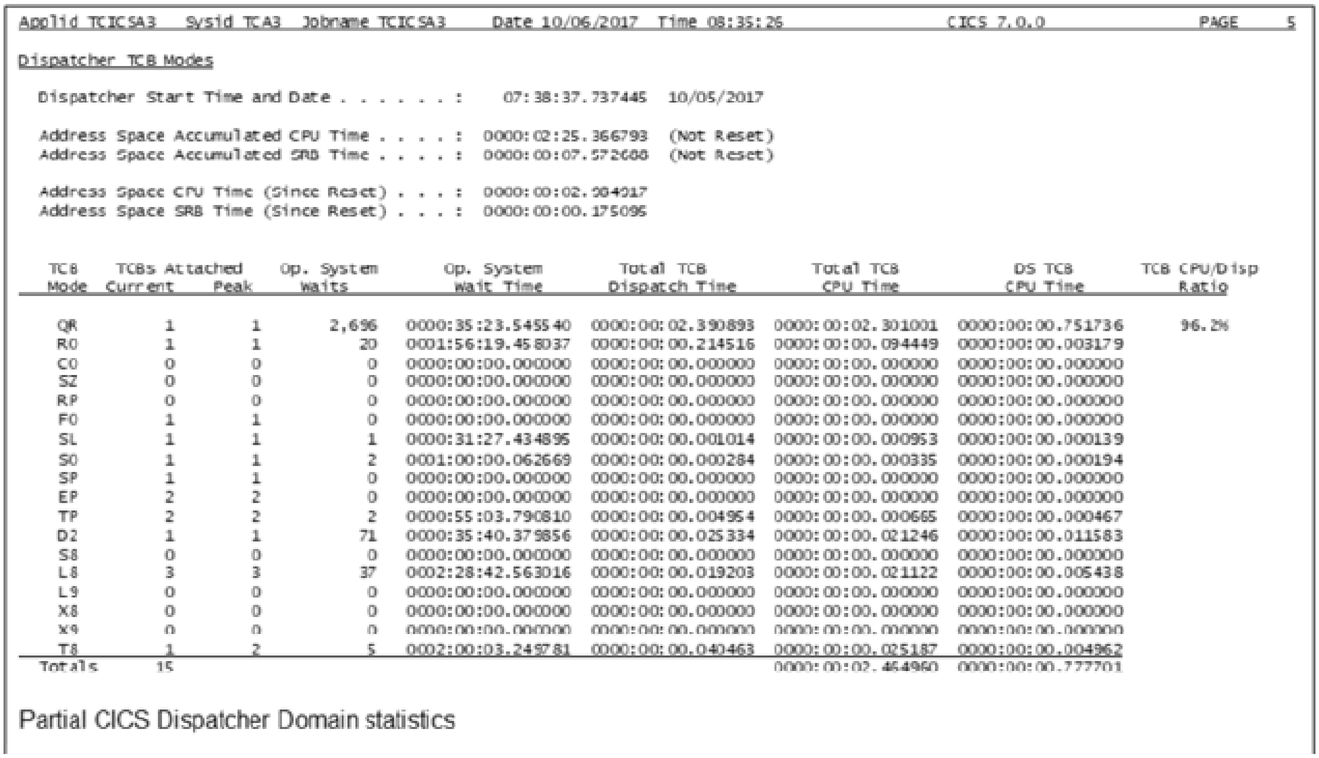
output, there is a COBOL data structure representing the columns of the table and their data types. This data area can be used on the SQL statements to supply the “host variables” to accept the data being processed. One special style of definitions is related to DB2 VARCHAR data type, where you will notice the COBOL structure contains a half-word binary length field along with the character data attribute. This length contains the actual number of significant characters within the character attribute.

Tom then went on to look at other examples.

Tom next took a look at trace for the application program. Figure 2 contains the portion of trace entries representing the SQL SELECT statement to find the average and maximum SALARY. The bracketed entries on the left represent the total time spent in the CICS DB2 interface. Most of this time (processor time) will actually be captured and reported on the DB2 region side. This is because control is transferred from the DB2 interface to DB2 as a secondary address space very quickly. Tom has always classified the time spent between “ERM ENTRY” and “ERM EXIT” as time when the application has turned control over to DB2 and entered a wait state.

The bracketed entries on the right represent the total time spent in the DB2 region itself. The “D2EX1 ENTRY” trace entry occurs just before giving control to DB2. The “D2EX1 EXIT” occurs just after DB2 give control back to the CICS DB2 interface. This time is considered to be solely done on behalf of the DB2 region and the CICS application is in a wait state for its completion.

While the times shown in Figure 2 might seem to be relatively low, on their small mainframe they have executed around 1.5 million



```
Applid TCICSA3   Sysid TCA3  Jobname TCICSA3     Date 10/06/2017  Time 08:35:26            CICS 7.0.0             PAGE     5

Dispatcher TCB Modes

  Dispatcher Start Time and Date . . . . . . :    07:38:37.737445  10/05/2017

  Address Space Accumulated CPU Time . . . . :  0000:02:25.366793  (Not Reset)
  Address Space Accumulated SRB Time . . . . :  0000:00:07.572688  (Not Reset)

  Address Space CPU Time (Since Reset) . . . :  0000:00:02.504017
  Address Space SRB Time (Since Reset) . . . :  0000:00:00.175096


   TCB    TCBs Attached    Op. System       Op. System        Total TCB          Total TCB            DS TCB          TCB CPU/Disp
  Mode  Current    Peak     Waits           Wait Time       Dispatch Time        CPU Time            CPU Time            Ratio

   QR        1        1       2,696   0000:35:23.545540  0000:00:02.390893  0000:00:02.301001  0000:00:00.751736       96.2%
   R0        1        1          20   0001:56:19.458037  0000:00:00.214516  0000:00:00.094449  0000:00:00.003179
   C0        0        0           0   0000:00:00.000000  0000:00:00.000000  0000:00:00.000000  0000:00:00.000000
   SZ        0        0           0   0000:00:00.000000  0000:00:00.000000  0000:00:00.000000  0000:00:00.000000
   RP        0        0           0   0000:00:00.000000  0000:00:00.000000  0000:00:00.000000  0000:00:00.000000
   F0        1        1           0   0000:00:00.000000  0000:00:00.000000  0000:00:00.000000  0000:00:00.000000
   SL        1        1           1   0000:31:27.434895  0000:00:00.001014  0000:00:00.000953  0000:00:00.000139
   S0        1        1           2   0001:00:00.062669  0000:00:00.000284  0000:00:00.000335  0000:00:00.000194
   SP        1        1           0   0000:00:00.000000  0000:00:00.000000  0000:00:00.000000  0000:00:00.000000
   EP        2        2           0   0000:00:00.000000  0000:00:00.000000  0000:00:00.000000  0000:00:00.000000
   TP        2        2           2   0000:55:03.790810  0000:00:00.004954  0000:00:00.000665  0000:00:00.000467
   D2        1        1          71   0000:35:40.379856  0000:00:00.025334  0000:00:00.021246  0000:00:00.011583
   S8        0        0           0   0000:00:00.000000  0000:00:00.000000  0000:00:00.000000  0000:00:00.000000
   L8        3        3          37   0002:28:42.563016  0000:00:00.019203  0000:00:00.021122  0000:00:00.005438
   L9        0        0           0   0000:00:00.000000  0000:00:00.000000  0000:00:00.000000  0000:00:00.000000
   X8        0        0           0   0000:00:00.000000  0000:00:00.000000  0000:00:00.000000  0000:00:00.000000
   X9        0        0           0   0000:00:00.000000  0000:00:00.000000  0000:00:00.000000  0000:00:00.000000
   T8        1        2           5   0002:00:03.249781  0000:00:00.040463  0000:00:00.025187  0000:00:00.004962
  Totals    15                                                              0000:00:02.464960  0000:00:00.777701
```

Partial CICS Dispatcher Domain statistics

Figure 4: DB2 interface statistics

instructions to complete this SELECT statement.

Tom went on to look at trace in more detail.

Tom next looked at CEDF screens for the DB2 interface in order to show the use of CEDF to stop on SQL statements in a similar fashion to the EXEC CICS commands. In Figure 3, we can see the completion of the SQL SELECT statement execution. We do not see the complete statement but can directly tie back to the compile listing by the STMT= option. We can see the first data variable being returned (OVAR 001) in packed decimal format. If we were to place the cursor on the plus sign below OVAR 001 and hit enter, we could see all the data variables being returned.

We also see most of the attributes from the SQLCA including the most important ones like SQLCODE and SQLSTATE, which indicate successful completion of the statement.

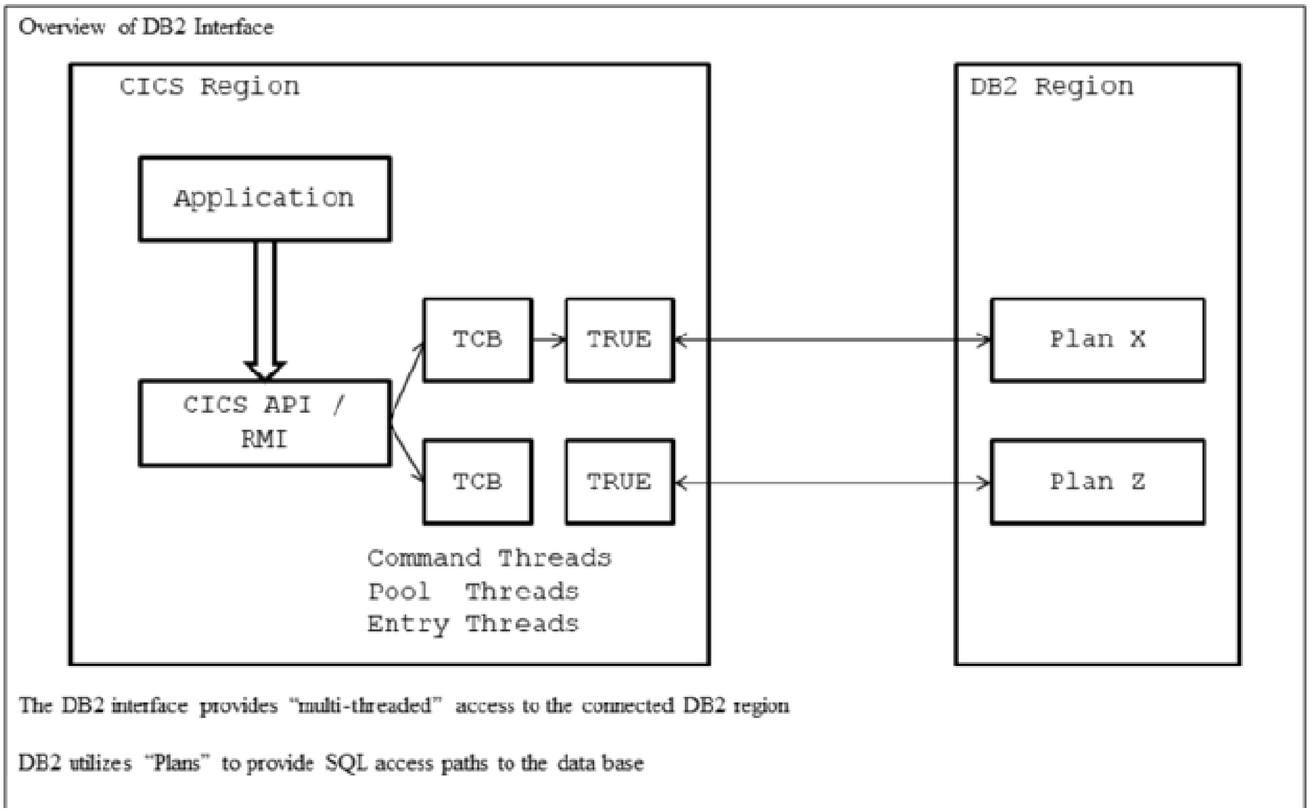
Tom looked at more examples.

Tom Dunlap then turned his attention to statistics for the DB2 interface. Figure 4 shows a set of CICS DS domain statistics. While not directly associated with the CICS DB2 interface, the timings for the L8 and L9 TCBs do include the amount of CPU and WAIT time consumed by the DB2 interface. Tom then showed lots of other examples.

Tom next looked at DB2 resource definitions. Figure 5 provides an overview of the CICS DB2 interface. It is implemented as a TRUE (Task-Related User Exit) in CICS, which is provided by the CICS product itself. Any support or enhancements will be provided by the CICS Development groups.

The CICS DB2 interface is designed to provide a “multi-threaded” interface with the connected DB2. CICS can establish a connection with only one DB2 region at a

Overview of DB2 Interface



The DB2 interface provides "multi-threaded" access to the connected DB2 region

DB2 utilizes "Plans" to provide SQL access paths to the data base

Figure 5: DB2 resource definitions

time. This is controlled by the DB2CONN resource.

There will be three different style of threads managed by the interface:

1 Command threads – on which any DB2 command will be issued

2 Pool threads – which will be utilized for applications not associated with a DB2ENTRYresource

3 Entry threads – used in support of applications associated with a specific DB2ENTRY resource

Pool threads will consume more processor time since they will have to perform DB2 “sign-on” function with each request. This is because the thread may be used by different applications. If many applications are using pool threads, this can add as much as 4-7 % overhead in processing these requests.

Entry threads will be defined and dedicated to a given application to ensure better thread reuse and reduce the overhead required to process the DB2 requests. Only one command thread is required since CICS will typically issue only one DB2 command at a time.

There will be defined “Plans” used to allow DB2 to choose access paths to be used with these requests. This is especially true for Pool and Command threads.

Tom explained the parameters that can be coded, with examples of values used.

Tom also described using SVC dump information for the DB2 interface, more DSNC commands (for example to issue a DB2 command to display the current threads from the DB2 point of view), and DB2 resource definitions with complete CEDA screens.

A copy of Tom Dunlap’s presentation is available for download from the Virtual CICS user group Web site at https://itech-ed.com/CICS/CICSDB2interfaceJan19.pdf.

You can see and hear the whole user group meeting on YouTube at https://youtu.be/SPoZq5W2DHo.

Meeting dates

The following meeting dates have been arranged for the Virtual CICS user group:

- On 5 March 2019, Ezriel Gross, CEO, Circle Software will be discussing “Efficient management of online resources using CICS Configuration Manager”.
- The following meeting is on 14 May 2019, when Tom Dunlap, CTO at Themis, Inc will be back with us. He will be discussing, “The CICS / MQ interface”.

We will be using Citrix GoToMeeting for the user group meetings.

Recent CICS articles

OSGi Demystified: 5.2 – Declarative Services: A Tutorial by Alexander D Brown on CICS DevCenter (29 January 2019). You can find the article at: https://developer.ibm.com/cics/2019/01/29/osgi-demystified-5-2-declarative-services-sample/

OSGi Demystified: 5.1 – Declarative Services: A Tutorial by Alexander D Brown on CICS DevCenter (29 January 2019). You can find the article at: https://developer.ibm.com/cics/2019/01/29/osgi-declarative-service-tutorial/

Getting started with Node.js in CICS by Mark Cocker on CICS DevCenter (11 January 2019). You can find the article at: https://developer.ibm.com/cics/2019/01/11/getting-started-with-node-js-in-cics/

Understanding zIIP Usage in CICS by Ian Burnett in *IBM Systems Magazine* (January 2019). You can find the article at: http://ibmsystemsmag.com/mainframe/administrator/cics/ziip-usage/

CICS - Still Powering the Planet: A New Release by Nick Garrod in *Enterprise Tech Journal* (2018 Issue 5). You can find the article at: http://ourdigitalmags.com/publication/index.php?i=546656#{%22issue_id%22:546656,%22numpages%22:1,%22page%22:10

System Programmers: An introduction to Node.js in CICS by Mark Cocker on CICS DevCenter (8 January 2019). You can find the article at: https://developer.ibm.com/cics/2019/01/08/system-programmers-intro-to-node-js-in-cics/

Node.js Developers: an introduction to Node.js in CICS in CICS by Mark Cocker on CICS DevCenter (8 January 2019). You can find the article at: https://developer.ibm.com/cics/2019/01/08/node-js-developers-intro-to-nodejs-cics/

Enterprise Architects: an introduction to Node.js in CICS by Mark Cocker on CICS DevCenter (8 January 2019). You can find the article at: https://developer.ibm.com/cics/2019/01/08/enterprise-architects-intro-to-node-js-in-cics/

How to recreate the source of Program List Tables (PLTs) by Darren Beard on CICS DevCenter (20 December 2018). You can find the article at: https://developer.ibm.com/cics/2018/12/20/recreate-the-source-of-program-list-tables/

Defining your own actions for policy system rules by Andy Wharmby on CICS DevCenter (20 December 2018). You can find the article at: https://developer.ibm.com/cics/2018/12/20/defining-your-own-actions-for-policy-system-rules/

CICS TS V5.5 – our best release yet, now generally available! by Sam Catling on CICS DevCenter (14 December 2018). You can find the article at: https://developer.ibm.com/cics/2018/12/14/cics-ts-v5-5-release/

CICS Explorer 5.5 GA – new tools to analyze your CICS resources and definitions by Dave Nice on CICS DevCenter (13 December 2018). You can find the article at: https://developer.ibm.com/cics/2018/12/13/cics-explorer-5-5-ga-new-tools-to-analyze-your-cics-resources-and-definitions/

Arcati Mainframe Yearbook 2019

Many members of the Virtual CICS user group contributed to this year's user survey in the Arcati Mainframe Yearbook.

If you want to get your hand on this one-stop shop for mainframers, the Yearbook is free to download from the Arcati Web site here: http://www.arcati.com/newyearbook19.

Like us on Facebook

#VirtualCICS

About the Virtual CICS user group

The Virtual CICS user group was established as a way for individuals using IBM's CICS TS systems to exchange information, learn new techniques, and advance their skills with the product.

The Web site at www.fundi.com/virtualcics provides a central point for coordinating periodic meetings (which contain technically-oriented topics presented in a webinar format), and provides articles, discussions, links, and other resources of interest to IBM CICS practitioners. Anyone with an interest in CICS is welcome to join the Virtual CICS user group and share in the knowledge exchange.

To share ideas, and for further information, contact trevor@itech-ed.com.

The Virtual CICS user group is free to its members.