IBM Education Assistance for z/OS V2R2

Item: SDUMP CTRACE
Element/Component: BCP/SDUMP

Material current as of May 2015

Agenda

- Trademarks
- Presentation Objectives
- Overview
- Usage & Invocation
- Presentation Summary
- Appendix

Trademarks

- See url http://www.ibm.com/legal/copytrade.shtml for a list of trademarks.

Presentation Objectives

- Describe the purpose of SDUMP CTRACE support
- Describe how to invoke and use SDUMP CTRACE

Overview

- Problem Statement / Need Addressed
  - The serviceability of SVC dump (SDUMP) needs improvement
    - SDUMP hung conditions
    - SDUMP delays / long dump times
- Solution
  - Exploit Component Trace (CTRACE) services to improve SDUMP diagnostic data
- Benefit / Value
  - Improved SDUMP serviceability
  - Reduce the need for problem recreates

Usage & Invocation

- Overview – Component Trace service
- Planning for SDUMP Component Tracing
  - Trace buffers
  - Select the trace options
  - Decide where to collect the trace records
  - Create CTIDMPxx parmlib member
  - Use TRACE CT command to start, stop, or modify Sdump Ctrace
- Obtaining the SDUMP Component Trace
- Verifying SDUMP Component Tracing
- Viewing the SDUMP Component trace data

Usage & Invocation... (Overview – Component Trace Service)

- Overview – Component Trace Service
  - Provide a way for MVS components to collect problem data about events
  - Each component that uses Component Trace service has to set up its own trace in a way that provides the unique data needed for the component
  - Trace data is used to see how the component was running and to aid in debugging the component problems

Usage & Invocation... (Planning for SDUMP Component tracing)

- Trace Buffers
  - Location: 64-bit Common Storage Area
  - Default size: 4MB
  - Range: 4MB – 32MB
  - Size is set by CTIDMPxx parmlib member or TRACE CT command
  - Buffer size determines whether you get all the records needed for diagnosis; when the buffer is full, the system wraps the buffer, overwriting the oldest records
  - Size can be changed after IPL

Usage & Invocation... (Planning for SDUMP Component tracing)

- Select the trace options
  - OPTIONS parameter
    - Set by CTIDMPxx and REPLY for TRACE CT command
    - Available trace options:

      ALL – trace everything
      MINIMUM – trace errors
    - Trace options revert to minimum event tracing when the operator turns the trace off
    - Most SDUMP error occurrences are not reproducible. Option 'ALL' is recommended

Usage & Invocation... (Planning for SDUMP Component tracing)

- Decide where to collect the trace records
  - In the SVC dump
  - In a trace data set or sets
    - External writer is supported, which allows trace buffers to be written to a trace data set on DASD or tape.

Usage & Invocation... (Planning for SDUMP Component tracing)

- Create CTIDMPxx parmlib member
  - Use the CTIDMPxx parmlib member to specify tracing options for the Sdump Component Trace
  - Parameters you can specify in a CTIDMPxx parmlib member:
    - ON / OFF
    - BUFSIZE
    - OPTIONS
    - WTR
    - WTRSTART / WTRSTOP
  - IBM supplied CTIDMP00 parmlib member in SYS1.PARMLIB
  - Default member CTIDMP00 contents:

```
TRACEOPTS
    ON
    OPTIONS('ALL')
    BUFSIZE(4M)
```

Usage & Invocation... (Planning for SDUMP Component tracing)

- Create CTIDMPxx parmlib member...
  - Decide if IBM-supplied CTIDMP00 meets the needs of your installation.
  - **Default** Tracing is activated using CTIDMP00 during DUMPSRV address space initialization.
  - If parmlib does not contain CTIDMP00 at initialization, BUFSIZE 4M and OPTIONS Minimum will be used at initialization.
  - Tracing options specified in the CTIDMP00 parmlib member can be overridden by another CTIDMPxx parmlib member identified on a TRACE CT command using PARM parameter
    - For example, have TRACE CT command in a COMMNDxx parmlib member

      COM='TRACE CT,ON,COMP=SYSDUMP,PARM=CTIDMPxx'

Usage & Invocation... (Planning for SDUMP Component tracing)

- Create CTIDMPxx parmlib member...
  - WTRSTART(membername)
    - Identifies a member containing source JCL for a started task that the system uses to start the SDUMP component trace external writer and to open the data sets that the writer uses.
  - WTR(membername|DISCONNECT)
    - Connects or disconnects the component trace external writer and the trace.
    - The member name identifies the member that contains the source JCL that invokes the external writer.
    - The membername in the WTR parameter must match the membername in the WTRSTART parameter.
    - WTR(DISCONNECT) disconnects the writer and the trace.

Usage & Invocation... (Planning for SDUMP Component tracing)

- Create CTIDMPxx parmlib member...
  - WTRSTOP(membername)
    - Identifies the member containing source JCL for a started task that the system used to start the Sdump component trace external writer
    - The system also closes the data sets the writer used.

See *z/OS MVS Initialization and Tuning Reference*, chapter 27 for CTncccxx (component trace parameters)

Usage & Invocation... (Planning for SDUMP Component tracing)

- TRACE CT command
  - Use the TRACE CT command to start, stop, or modify the SDUMP CTRACE
  - SDUMP Component Trace name - SYSDUMP
  - Command syntax:

TRACE CT,ON,COMP=**SYSDUMP**[,PARM=*mem*]

OFF,

nnnnM,

TRACE CT,WTRSTART=*membername*[,WRAP|NOWRAP]

TRACE CT,WTRSTOP=*membername*[,FLUSH|NOFLUSH]

Usage & Invocation... (Planning for SDUMP Component tracing)

- Specifying TRACE CT options
  - In response to a TRACE CT command without the PARM parameter, the system prompts you to specify the component trace options you want with message ITT006A
  - Use the REPLY command to specify each option, OPTIONS, or WTR
  - You can issue the DISPLAY TRACE command before entering a TRACE CT command to verify what options are supported
  - REPLY command syntax for specifying TRACE CT options:

    R id[,OPTIONS=(*name*)]

    [,WTR=(*membername*|DISCONNECT)]

    [,END]

See *z/OS MVS System Commands* for information about the TRACE CT command

Usage & Invocation... (Planning for SDUMP Component tracing)

- TRACE CT command Examples:
  - Change buffer size to 8M:

    TRACE CT,8M,COMP=SYSDUMP

  - Change options using PARM=

    TRACE CT,ON,COMP=SYSDUMP,PARM=CTIDMPxx

  - Change the OPTION to 'ALL'

    TRACE CT,ON,COMP=SYSDUMP

    *08 ITT006A SPECIFY OPERAND(S) FOR TRACE CT COMMAND.
    R 8,OPTIONS=(ALL),END

Usage & Invocation... (Planning for SDUMP Component tracing)

- TRACE CT command Examples...
  - Turn off SDUMP CTRACE:

    TRACE CT,OFF,COMP=SYSDUMP

  - Start Ctrace External Writer

    TRACE CT,WTRSTART=WWWTR1

  - Connect SDUMP CTRACE to the External Writer

    TRACE CT,ON,COMP=SYSDUMP

    *07 ITT006A SPECIFY OPERAND(S) FOR TRACE CT COMMAND.
    R 7,WTR=WWWTR1,END

Usage & Invocation... (Planning for SDUMP Component tracing)

- TRACE CT command Examples...
  - Disconnect SDUMP CTRACE from the External Write

    TRACE CT,ON,COMP=SYSDUMP

    *08 ITT006A SPECIFY OPERAND(S) FOR TRACE CT COMMAND.

    R 8,WTR=DISCONNECT,END

  - Stop the External Writer

    TRACE CT,WTRSTOP=WWWTR1

Usage & Invocation... (Obtaining the SDUMP Component Trace)

- Obtaining a SDUMP Component Trace
  - In the SVC dump
    - SDUMP trace data is captured at the end of SDUMP capture phase and dumped at the end of the dump.
    - If the trace data is needed at other times, a new dump must be requested to collect the SDUMP trace data.
  - In a trace data set
    - Refer to *z/OS MVS Diagnosis: Tools and Service Aids*, Chapter 12 Component Trace, for details on “Request Writing Component trace data to trace data sets”

Usage & Invocation... (Verifying SDUMP Component Tracing)

- Verifying SDUMP Component tracing - DISPLAY TRACE command

```
DISPLAY TRACE,COMP=SYSDUMP

IEE843I 14.44.25  TRACE DISPLAY      FRAME LAST   F      E   SYS=S7A0
  COMPONENT      MODE BUFFER HEAD SUBS
 ------------------------------------------------------------
  SYSDUMP        ON    0004M
     ASIDS       *NOT SUPPORTED*
     JOBNAMES    *NOT SUPPORTED*
     OPTIONS     ALL
     WRITER      *NONE*
```

Usage & Invocation... (Verifying SDUMP Component Tracing)

- Verifying SDUMP Component tracing - DISPLAY TRACE command

```
DISPLAY TRACE,COMP=SYSDUMP

IEE843I 14.44.25  TRACE DISPLAY      FRAME LAST   F      E   SYS=S7A0
  COMPONENT      MODE BUFFER HEAD SUBS
 ------------------------------------------------------------
  SYSDUMP        MIN   0004M
     ASIDS       *NOT SUPPORTED*
     JOBNAMES    *NOT SUPPORTED*
     OPTIONS     MINIMUM

     WRITER      *NONE*
```

Usage & Invocation... (Verifying SDUMP Component Tracing)

- Verifying SDUMP Component tracing - DISPLAY TRACE command

```
DISPLAY TRACE,COMP=SYSDUMP

IEE843I 14.44.25  TRACE DISPLAY      FRAME LAST   F      E   SYS=S7A0
  COMPONENT      MODE BUFFER HEAD SUBS
 ------------------------------------------------------------
  SYSDUMP        ON    0004M
     ASIDS       *NOT SUPPORTED*
     JOBNAMES    *NOT SUPPORTED*
     OPTIONS     ALL
     WRITER      WWWTR1
```

Usage & Invocation... (Viewing the SDUMP Component trace data)

- Viewing the SDUMP Component trace data
  - Use IPCS subcommand CTRACE

```
IPCS CTRACE COMP(SYSDUMP) SHORT
IPCS CTRACE COMP(SYSDUMP) FULL
```

Usage & Invocation... (Viewing the SDUMP Component trace data)

IPCS CTRACE COMP(SYSDUMP) SHORT

```
COMPONENT TRACE SHORT FORMAT

COMP(SYSDUMP)

**** 01/22/2015

SYSNAME   MNEMONIC  ENTRY ID    TIME STAMP      DESCRIPTION

-------   --------  --------  ---------------  -------------

S7A0      WtDSVSST  0000007F  22:23:29.882181  Wait for DSVSSTECB to be post

S7A0      DmpStrtd  00000002  22:28:08.433737  Sdump started

S7A0      ClrStor   00000029  22:28:08.433753  Storage cleared for new dump

S7A0      EntyTSPR  00000013  22:28:08.433754  Entry at IEAVTSPR

S7A0      EntyTSPR  00000013  22:28:08.433759  Entry at IEAVTSPR

S7A0      SdumpPml  00000004  22:28:08.433760  Sdump PList after calling TSPR

S7A0      DumpSupp  00000006  22:28:08.433773  Dump is not being suppressed

S7A0      SchSD2TR  00000026  22:28:08.433774  Schedule TSD2 for SNAPTRC

S7A0      RTCTLUCT  0000002F  22:28:08.434757  Increment RTCTLUCT

S7A0      ScheTSD2  00000082  22:28:08.434757  Schedule SRB to TSD2
```

Usage & Invocation... (Viewing the SDUMP Component trace data)

IPCS CTRACE COMP(SYSDUMP) FULL

```
COMPONENT TRACE FULL FORMAT

COMP(SYSDUMP)

**** 01/22/2015

SYSNAME   MNEMONIC  ENTRY ID    TIME STAMP      DESCRIPTION

-------   --------  --------  ---------------  -------------

S7A0      WtDSVSST  0000007F  22:23:29.882181  Wait for DSVSSTECB to be post

    ASID..... 0005      IssueMod. IEAVTSST  TCB...... 004DFD90

    RetnAddr. 89C0258A

S7A0      DmpStrtd  00000002  22:28:08.433737  Sdump started

    ASID..... 0029      IssueMod. IEAVTSDX  TCB...... 004F8240

    RetnAddr. 8122A4BE

S7A0      ClrStor   00000029  22:28:08.433753  Storage cleared for new dump

    ASID..... 0029      IssueMod. IEAVTSDX  TCB...... 004F8240

    RetnAddr. 8122A590

...........
```

Usage & Invocation... (Viewing the SDUMP Component trace data)

```
S7A0      EntyTSPR  00000013  22:28:08.433754  Entry at IEAVTSPR
    ASID..... 0029      IssueMod. IEAVTSPR  TCB...... 004F8240
    RetnAddr. 812304BC
    RTSDFNCD. 0001
S7A0      SnapTrcE  0000000C  22:28:08.436861  SNAPTRC end
    ASID..... 0005      IssueMod. IEAVTSD2  TCB...... 00000000
    RetnAddr. 863FF9E2
    RetCode.. 00000000
    SDTTCH... 7F5E3000
S7A0      ENQHeld   0000002B  22:28:08.437346  ENQ Obtained
    ASID..... 0005      IssueMod. IEAVTSXS  TCB...... 004FCC98
    RetnAddr. 8641110A
    QName.... SYSIEA01
    RName.... SDUMPENQ
```

Usage & Invocation... (Viewing the SDUMP Component trace data)

```
S7A0      CalExTbl  00000062  22:28:08.465305  Call to Exit Table routine
    ASID..... 0005      IssueMod. IEAVTSDU  TCB...... 004FCC98
    RetnAddr. 863D859C
    ExitName. x4...AHLYSDMP 13108 HBB77A0..\.
S7A0      ExitRtn   0000005F  22:28:08.469057  Return from Exit Table routine
    ASID..... 0005      IssueMod. IEAVTSDU  TCB...... 004FCC98
    RetnAddr. 863D864C
    RetCode.. 00000000
S7A0      NonDynEx  0000005E  22:28:08.472842  Non dynamic Exit
    ASID..... 0005      IssueMod. IEAVTSDU  TCB...... 004FCC98
    RetnAddr. 863D88CC
    ExitName. x4...IEAVTRSX  2014.073...}.{{..
    ExitAddr. 854EFB10
    ExitType. Early Global Exit
    HASID.... 0005
```

Usage & Invocation... (Viewing the SDUMP Component trace data)

```
S7A0      DPLatDW0  00000014  22:28:08.609845  DPL, SDW1 at entry of IEAVTDW0

    ASID..... 0005      IssueMod. IEAVTDWT  TCB...... 004DF8B0

    RetnAddr. 863E1F74

           DPL:

    +0000  C4D7D340  00000000  00000000  01FDEB68  | DPL ............ |

    +0010  01FD9000  000000F5  00000000  10000000  | .......5........ |

    +0020  00000000  00000000  00000000  00000000  | ................ |

    +0030  00000000  80008200  000000E5  00000001  | ......b....V.... |

    +0040  00001000  000D2000  00000000  00000000  | ................ |

    +0050  00000000  00000000  00000000  00000000  | ................ |

     ...
```

Usage & Invocation...

- Messages

IEA053I ***name-of-function:*** **Service** ***name-of-service*** **failed with RC:***retcode* **RSN:***rsncode*

Explanation: A service was invoked which could not process the request. This message records this error.

In the message text:

Name-of-function : The name of the function that invoked the service.

Name-of-service : The name of the service that failed.

Retcode: The return code from the service that failed.

Rsncode: The reason code from the service that failed.

System action: This failing service, along with the function that invoked that service, govern the action that will be taken. In some cases, an ABEND may be issued because the function can not continue without the service.

Operator response: Notify the system programmer.

System programmer response: Search problem reporting data bases for a fix for this problem. If no fix exists, contact the IBM Support Center.

Module: Various. Refer to ***name-of-function*** as the detecting module.

R**outing code: 10**

Descriptor code: 4

Usage & Invocation...

IEA054I SDUMP CTRACE definition failed using CTIDMP00. RC=*rc*, RSN=*rsn*

Explanation: The system could not define the SDUMP component trace using the CTIDMP00 parmlib member.

In the message text:

rc The return code provided by the CTRACE DEFINE macro

rsn The reason code provided by the CTRACE DEFINE macro

System action: The system will attempt to define the SDUMP component trace without the CTIDMP00 parmlib member.

Operator response: Contact the system programmer.

System programmer response: If the return and reason codes refer to a CTIDMP00 parmlib member error, correct the member and have the operator either re-IPL or use the TRACE CT command to use the corrected member. Else search the problem reporting data bases for a fix for the problem. If no fix exists, contact the IBM Support Center.

Module: IEAVTSCI

R**outing code:** -

Descriptor code: 4

Usage & Invocation...

IEA055I Component trace option *optname* is not valid.

Explanation: The system encountered an incorrect option in the CTIDMPxx parmlib member or the

TRACE CT command.

In the message text:

Optname : The specified option that is incorrect.

System action: The system does not start the requested component trace. Processing continues with the next option specified.

Operator response: Contact the system programmer.

System programmer response: Examine the options near the indicated character string for a misspelling or other error. **Correct the error before reissuing the TRACE CT command.**

Module: IEAVTSSRI

R**outing code:** -

Descriptor code: 5

Presentation Summary

- The purpose of SDUMP CTRACE support
- How to invoke and use SDUMP CTRACE

Appendix

- Publications:

SA38-0673 z/OS V2R2.0 MVS System Messages, Vol 6 (GOS-IEA)

GA32-0905 z/OS V2R2.0 MVS Diagnosis: Tools and Service Aids