



IBM Software Group

IBM WebSphere® Data Interchange V3.3

WebSphere Data Interchange Service Director Trace



@business on demand.

© 2007 IBM Corporation

This presentation will review the WebSphere Data Interchange Service Director Trace.

Objectives

- Determine when to use the WDI Service Director Trace
- WDI Service Director Overview
- Describe the WDI Service Director Trace
- Reference.

The presentation will discuss the WebSphere Data Interchange Service Director Trace and give an overview of the Service Director.

WDI 3.3 Service Director Trace

- When to use the Service Director Trace?
 - ▶ Usually a request from L2 support
 - ▶ Unable to isolate the problem using other diagnostics
- Non-Data Transformation
- Does not trace Data Transformation
 - ▶ Will trace the Utility
 - ▶ Data Transformation uses limited SD calls.
 - Use TRACELEVEL() perform keyword
 - Writes to EDIDTTRC dataset, TSQ, or environment variable

The service director trace is usually generated via a WebSphere Data Interchange L2 support request. This would be requested for a problem with any non-data transformation function that cannot be isolated or recreated using other diagnostics.

Section

Service Director Over view

WDI 3.3 Service Director

- WDI Service Director
 - ▶ Provides access to WDI services via API.
 - ▶ Call/return semantics for invoking services
 - ▶ Execution time binding
 - ▶ Cross language communication between WDI and your application.
 - ▶ Environment dependent services

The Service Director is a set of routines which provide access to WebSphere Data Interchange services and a facility for defining those services. Simply this is an application programming interface (API). The functions provided by the Service Director include:

Call/return semantics for invoking services.

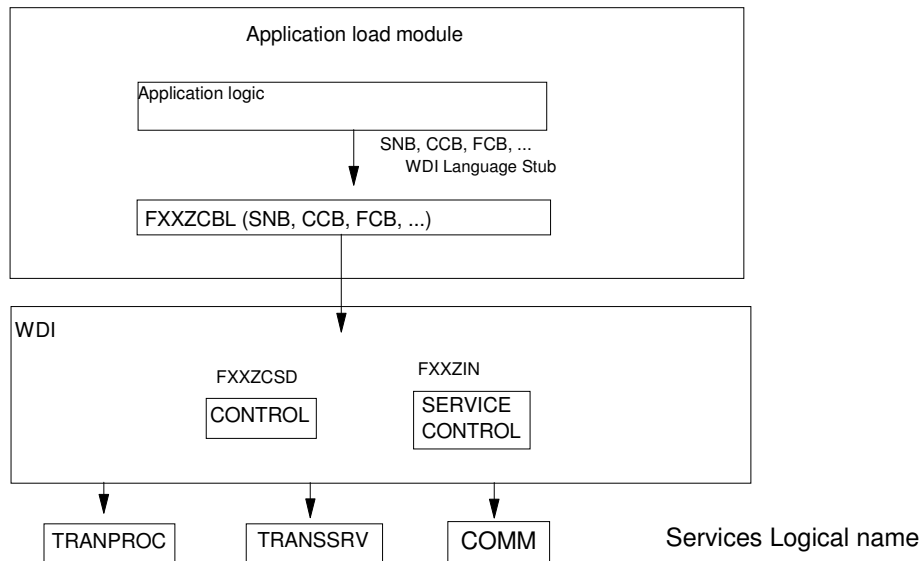
Execution time binding. The services are not linked with the calling application.

Cross language communication between DataInterchange supported languages.

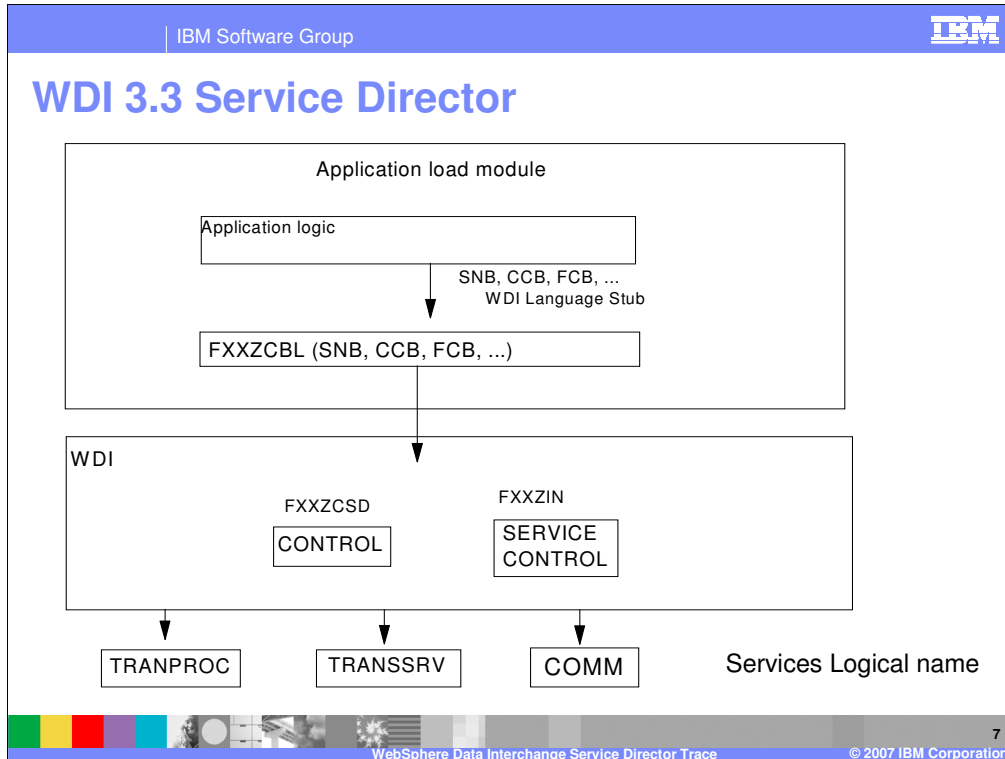
(For example, a COBOL application may request a service written in "C").

Environment dependent services.

WDI 3.3 Service Director



The Service Director is the "control point" and provides an access to all common services from the application. The application programmer, when requesting a service, utilizes a logical name. The logical name references the physical name of the service requested. From the application program viewpoint the services appear as "logical services". This means that the application calls the service by an assigned name with no regard to where the actual "physical" service resides or what its real physical name is. The Service Director handles all calls to the services and routes the requests on behalf of the application.



By using logical names, the application program is given the ability to perform "dynamic calls". Functions can be changed/modified/enhanced with no impact to the application. Once the service director is given control its actions are controlled by a service table. This table contains service names, locations, flags, indicators, etc.

The service director is given control by the application issuing a call to a **language stub** and passing it the required information (requested service, common block address, etc.). The service director will then map the logical to physical name, then load and pass control to the service. When the service has completed processing, control returns to the service director and then to the application program. The application can then examine the return code, data buffers, etc., and continue processing.

WDI 3.3 Service Director

- API Languages
 - ▶ FXXZASM Service Director Asm language stub
 - ▶ FXXZC Service Director "C" /370 language stub
 - ▶ FXXZCBL Service Director COBOL language stub
 - ▶ FXXZPLI Service Director PL/I language stub

The load library distributed with WDI contains a load module for each of the stub programs. These programs are linked with the application program requesting the service. These stub programs are pieces of code that load and transfer control to WDI when a request is made. WDI is not physically part of the application load module. The distributed load library should be part of the //SYSLIB for the linkage editor.

WDI 3.3 Service Director

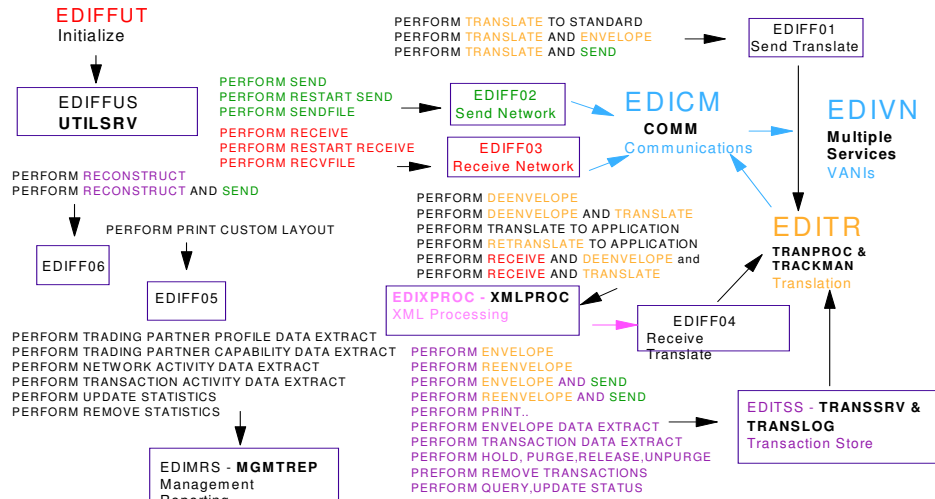
- SNB - Service Name Block.
 - ▶ The SNB contains the name of the service being requested and the number of parameters being passed to that service. The service director updates this control block with an index to allow fast access to the service if that service is requested multiple times. The first access to a service requires a search to resolve the logical service name to the physical service module name. Subsequent calls for the same service using the same SNB will bypass the search and use the index in the SNB to locate the requested service.
- CCB - Common Block
 - ▶ There is a single DataInterchange Common Block (CCB) per application and it contains data and pointers needed by the service director as well as return information set by the called service. A single CCB must be used by all services and facilities called by the application.
- FCB - Function Block
 - ▶ Function block contains a 2 byte block length and a the 2 byte function code within the service being requested.



WebSphere Data Interchange assumes that all parameters are pointers to control blocks. Any remaining parameters have no meaning to the service director but are parameters that will simply be forwarded to the requested service.

WDI 3.3 Service Director

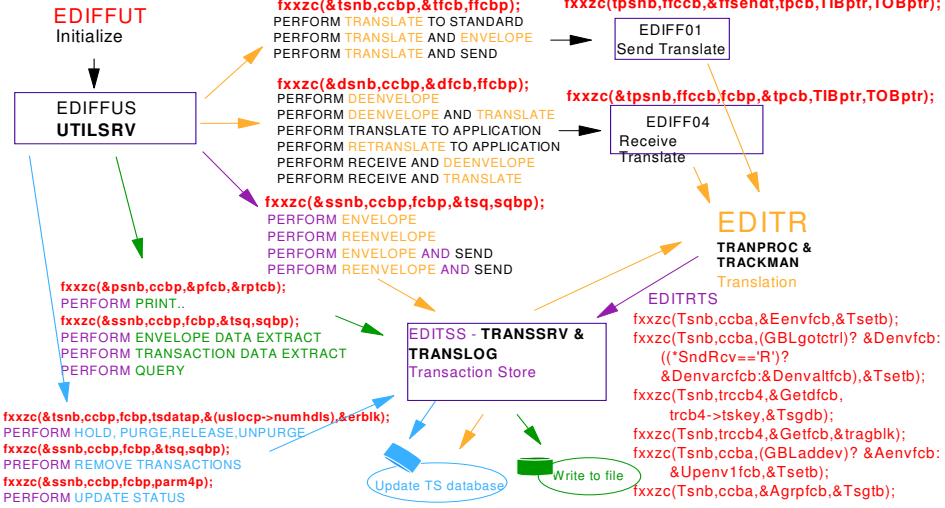
Start WDI Utility (EDIFFUT) from batch JCL



This is a diagram of the WebSphere Data Interchange Utility and key services used.

WDI 3.3 Service Director

Start WDI Utility (EDIFFUT) from batch JCL



This is a diagram of the WebSphere Data Interchange Utility service calls for translation and the document store.

Section

Service Director Trace

WDI 3.3 Service Director Trace

- Contains relevant information for each type of service invoked
- z/OS - Writes to EDITRACE data set if allocated
- CICS – Writes to EDITRACE TSQ if defined.
- Other platforms – Writes to EDITRACE if environment variable is defined.

The WebSphere Data Interchange Service Director trace is a wraparound storage trace of all services invoked through the Service Director. The Service Director uses a 32 KB buffer of storage that is filled with trace entries whenever services are requested through the Service Director. These trace entries consist of information that is relevant to each type of service that is invoked. Each time the Service Director reaches its wrap point (the 32 KB buffer is full), the Service Director writes the trace to the dump data set if either of the following situations is true:

For z/OS, the EDITRACE dump data set is allocated. In this case, the trace is written to this data set. For CICS, an EDITRACE temporary storage queue is defined. In this case, the trace is written to the CICS dump data set.

Section

z/OS and CICS

WDI 3.3 Service Director Trace

- Basic Entries
 - ▶ Trace header
 - ▶ Trace end
 - ▶ Service call
 - ▶ Service return
 - ▶ Service Director Environmental Services call



The trace contains five basic entry types when calling the Service Director Environmental Services.

WDI 3.3 Service Director Trace

- Subentries
 - ▶ Service Director initialization
 - ▶ Add service
 - ▶ Delete service
 - ▶ Getmain
 - ▶ Freemain
 - ▶ Set anchor
 - ▶ Get anchor

The trace contains nine subentry types when calling the Service Director Environmental Services.

WDI 3.3 Service Director Trace

Service Director Trace

Character View - PF11 or Right 50 command

Time of Day

Save area of caller

INITSERV - EDIZINIT called with SD initialization

```

Menu Utilities Compilers Help
Browse WINTERS.EDI.TRACE Line 00000000 Col 041 120
Command ==> Scroll ==> CSR
***** Top of Data *****
TIME 112337 DATE 00266 ID = 026 CPUID = 7A3450759672 PAGE 00000001

43F6 ILC 02 INTC 0033

7D00 13A37828 13A01598 00000000 00000000 *.uc..v.(.v.\.t....q.....*
1112 00010002 B4AF14EC E9A61701 0000B350 *INITSERVlv+.....Zw....6*
52CE 000C0003 B4AF14EC F4D8DA03 13A37DC8 *ENVSERV ..(lv.....4Q...t'H*
D9E5 C5C4C9E9 C9D5C9E3 00000001 00000000 *SETANCHRINITSERVEDIZINIT.....*
0000 00000003 B4AF14EC F4DB6A03 5B5B5B5B *ENVSERV .....4...$$$*
52F8 03E70003 B4AF14EC F4DB9503 13A37DC8 *DATABASElv.}lv.8.X.....n.t'H*
0000 00000003 B4AF14ED 28924703 5B5B5B5B *DATABASE.....k. $$$*
537A 00C80003 B4AF14ED 2892F603 13A37DC8 *EDITS 1..lv...H.....k6.t'H*
7F56 000A0004 B4AF14ED 3840BC03 13A38240 *ENVSERV ...(l.....tb *
0000 00000000 00000000 13A01230 000001C4 *GETMAIN .....D*
4040 E3E5E3C2 D3C1D5C3 13A01230 00000000 *SETANCHREDITS TVTBLANC.....*
0000 00000004 B4AF14ED 38425203 5B5B5B5B *ENVSERV ..... $$$*
8906 000A0004 B4AF14ED 38431103 13A3A4D0 *ENVSERV ...(l.i.....tu)*
0000 00000000 00000000 13ADC890 00001770 *GETMAIN .....H.....*
0000 00000004 B4AF14ED 38482803 5B5B5B5B *ENVSERV ..... $$$*
8A54 005B0004 B4AF14ED 38497003 13A3A4D0 *PROFLESV1.Y.l....$. ....tu)*
E9F0 005B0005 B4AF14ED 42E7EC06 13A3A5D0 *PROSERV 1.nq1.Z0.$.....X...tv)*
0DB2 005A0006 B4AF14ED 6D8E0604 13A3AE98 *ENVSERV ...(l.....!.....t.g*
0000 00000006 B4AF14ED 6D8FC904 5B5B5B5B *ENVSERV .....I. $$$*

```

A service call is made when a service is invoked through the WebSphere Data Interchange Service Director. The service logical name "INITSERV" is used for initialization.

WDI 3.3 Service Director Trace

Service Director Trace

Dump View - PF10 or Left 50 command

INITSERV - EDIZINIT called with SD initialization

Service Name

Entry Address

Return Address

Function Code - FCB
1 = init

Call nesting level

Time of day

```

Menu Utilities Compilers Help
*****
BROWSE WINTERS.EDI.TRACE Line 00000000 Col 001 080
Command ==> Scroll ==> CSR
*****
JOB WINTERSP STEP EXDBZ TIME 112337 DATE 00266 ID = 026
*****
PSW AT ENTRY TO SNAP 078D1000 800543F6 ILC 02 INTC 0033
-STORAGE
13A4C2E0 13A4C300 13A53FC0 13A522E0 00007D00 13A37828 13A01598 00000000 000000
13A4C300 C9D5C9E3 E2C5D9E5 93A54E60 80051112 00010002 B4AF14EC E9A61701 00000B
13A4C320 C5D5E5E2 C5D9E540 000506C0 93A552CE 000C0003 B4AF14EC F4D8DA03 13A37
13A4C340 E2C5E3C1 D5C3C8D9 C9D5C9E3 E2C5D9E5 C5C4C9E9 C9D5C9E3 00000001 000000
13A4C360 C5D5E5E2 C5D9E540 00000000 00000000 00000003 B4AF14EC F4DB6A03 5B5B5
13A4C380 C4C1E3C1 C2C1E2C5 93A615D0 93A552F8 03E70003 B4AF14EC F4DB9503 13A37
13A4C3A0 C4C1E3C1 C2C1E2C5 00000000 00000000 00000003 B4AF14ED 28924703 5B5B5
13A4C3C0 C5C4C9E3 E2404040 93AC7D90 93A5537A 00C80003 B4AF14ED 2892F603 13A37
13A4C3E0 C5D5E5E2 C5D9E540 000506C0 93AC7F56 000A0004 B4AF14ED 3840BC03 13A38
13A4C400 C7C5E3D4 C1C9D540 00000000 00000000 00000000 00000000 13A01230 000000
13A4C420 E2C5E3C1 D5C3C8D9 C5C4C9E3 E2404040 E3E5E3C2 D3C1D5C3 13A01230 000000
13A4C440 C5D5E5E2 C5D9E540 00000000 00000000 00000004 B4AF14ED 38425203 5B5B5
13A4C460 C5D5E5E2 C5D9E540 000506C0 93AC8906 000A0004 B4AF14ED 38431103 13A3A
13A4C480 C7C5E3D4 C1C9D540 00000000 00000000 00000000 00000000 13ADC890 00001
13A4C4A0 C5D5E5E2 C5D9E540 00000000 00000000 00000004 B4AF14ED 38482803 5B5B5
13A4C4C0 D7D9D6C6 D3C5E2E5 93ADE8A0 93AC8A54 005B0004 B4AF14ED 38497003 13A3A
13A4C4E0 D7D9D6C6 C5D9E540 93AF9598 93ADE9F0 005B0005 B4AF14ED 42E7EC06 13A3A
13A4C500 C5D5E5E2 C5D9E540 000506C0 93B40DB2 005A0006 B4AF14ED 6D8E0604 13A3A
13A4C520 C5D5E5E2 C5D9E540 00000004 00000000 00000006 B4AF14ED 6D8FC904 5B5B5

```

The service call information in the trace contains the service logical name, entry address which is the address to which control is transferred, return address which is the address to which control is transferred when returning from this module, the service function code which is the code indicating the requested function from the component being called, the call nesting level, and time-of-day clock value.

WDI 3.3 Service Director Trace

Service Director Trace - PF10 or Left 50

```

Menu Utilities Compilers Help
#####
BROWSE WINTERS.EDI.TRACE CHARS 'TRANSSRV' found
Command ==> Scroll ==> CSR
0000 00000004 B4AF14EF 57B77803 5B5B5B5B *ENVSERV .....$$$$*
8AEA 00A10004 B4AF14EF 57B8EC03 13A3BEC0 *TRANSSRV1C..l.....t.* TRANSSRV - EDITSS
787A 000D0005 B4AF14EF 6F797406 13A3C140 *ENVSERV ...{lv.....?.....tA * Transaction Store
D9E5 E3E2E2D9 E5C1D5C3 00000000 00000000 *GETANCHRTRANSSRVTSRVANC.....*
0000 00000005 B4AF14EF 6F7A3106 5B5B5B5B *ENVSERV .....?.....$$$$*
787A 000A0005 B4AF14EF 6F7A5C06 13A3C140 *ENVSERV ...{lv.....?*.tA *
0000 00000000 00000000 13BDA2A8 000008C8 *GETMAIN .....sy..H*
D9E5 E3E2E2D9 E5C1D5C3 13BDA2A8 00000000 *SETANCHRTRANSSRVTSRVANC..sy...*
0000 00000005 B4AF14EF 6F7C6806 5B5B5B5B *ENVSERV .....?@.....$$$$*
3A20 00B40005 B4AF14EF 6F7C906 13A3C090 *ENVSERV ...{lC.....?@i..t{.*
0000 00000005 B4AF14EF 6F7CDC06 5B5B5B5B *ENVSERV .....?@.....$$$$*
3B9E 00450005 B4AF14EF 6F7D8506 13A3C090 *DATABASE1w..}lC.....?'e..t{.*
0000 00000005 B4AF14EF 78014601 5B5B5B5B *DATABASE.....$$$$*
8586 006E0005 B4AF14EF 78023401 13A3C3A8 *ENVSERV ...{lvef.>.....tCy*
0000 00000005 B4AF14EF 78066101 5B5B5B5B *ENVSERV ...../.....$$$$*
3E24 00020005 B4AF14EF 78083401 13A3C090 *SYNCSERV1s..lC.....t{.*
3D4E 03E60006 B4AF14EF 78098C01 13A3C3A8 *DATABASE1w..}ls.+W.....tCy*
0000 00000006 B4AF14EF 79B10603 5B5B5B5B *DATABASE.....$$$$*
0000 00000005 B4AF14EF 79B18003 5B5B5B5B *SYNCSERV.....$$$$*
0000 00000004 B4AF14EF 79B1E603 5B5B5B5B *TRANSSRV.....W$$$$*
BF80 000F0004 B4AF14EF 79B2E003 13A3BD20 *PROFLESV1.Y.l.....\..t...*
E9F0 000F0005 B4AF14EF 79B40903 13A3BE18 *PROSERV 1.ngl.20.....t...*
0000 00000005 B4AF14EF 7CE56C01 5B5B5B5B *PROSERV .....@V$.$$$$*
0000 00000004 B4AF14EF 7CE61F01 5B5B5B5B *PROFLESV.....@W.....$$$$*
C732 000A0004 B4AF14EF 7CEBF101 13A3BE68 *ENVSERV ...{lvG.....@.l..t...*
0000 00000000 00000000 13C42000 00007000 *GETMAIN .....D.....*
0000 00000004 B4AF14EF 7CFAAE01 5B5B5B5B *ENVSERV .....@.....$$$$*

```

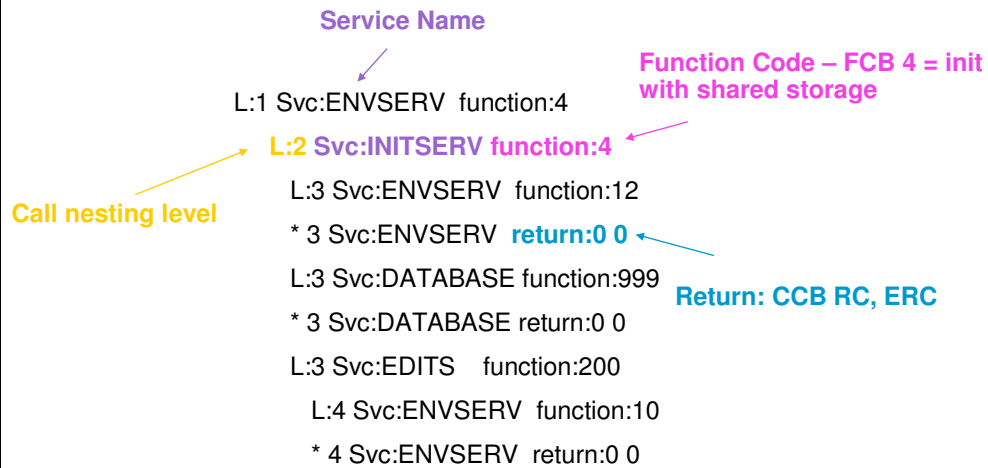
Document Store services logical name is "TRANSSRV".

Section

Other Platforms

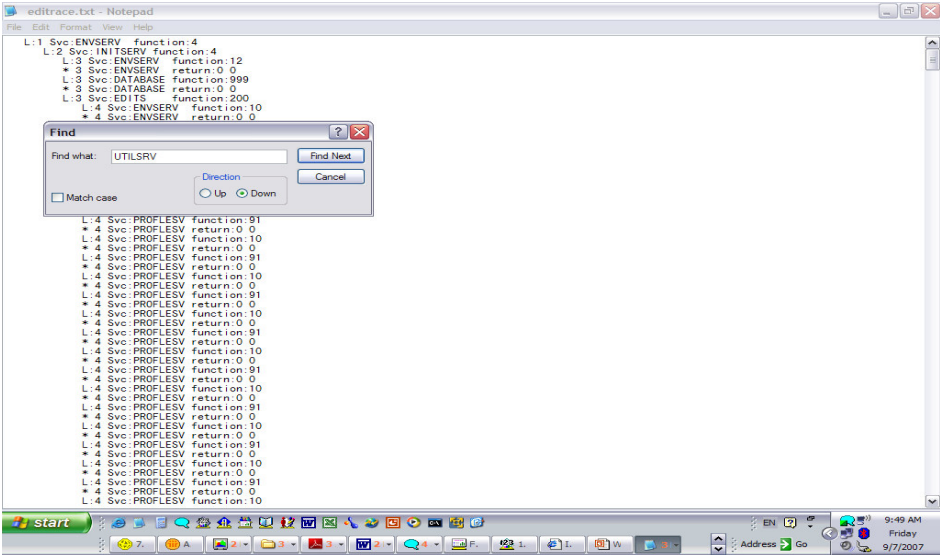
WDI 3.3 Service Director Trace

INITSERV – EDIZINIT called with SD initialization



A service call is made when a service is invoked through the WebSphere Data Interchange Service Director. The service logical name “INITSERV” is used for initialization. The service call information in the trace contains the service logical name, the service function code which is the code indicating the requested function from the component being called, and the call nesting level.

WDI 3.3 Service Director Trace



The screenshot shows a Notepad window titled "editrace.txt - Notepad" containing a Service Director trace. The trace consists of several lines of log entries, including:

```
L:1 Svc:ENVSESV function:4
L:2 Svc:INITSESV function:4
L:3 Svc:ENVSESV function:12
  * 3 Svc:ENVSESV return:0 0
L:3 Svc:DATABASE function:999
  * 3 Svc:DATABASE return:0 0
L:3 Svc:EDITS function:200
L:4 Svc:ENVSESV function:10
  * 4 Svc:ENVSESV return:0 0
```

A "Find" dialog box is open over the trace, with "UTILSRV" entered in the "Find what:" field. The "Direction" is set to "Down". The "Match case" checkbox is unchecked. The trace continues with many entries for "Svc:PROFLESV" with various function and return values.

At the bottom of the screenshot, the text "WebSphere Data Interchange Service Director Trace" and "© 2007 IBM Corporation" is visible, along with the page number "31".

The Service Director is used with the WebSphere Data Interchange API and also is used internally with Utility perform commands. Looking at the trace you can do a find command for the logical name "UTILSRV". This is the WebSphere Data Interchange Utility.

WDI 3.3 Service Director Trace

```

editrace.txt - Notepad
File Edit Format View Help
L:3 Svc:QSAMHDL function:1
L:4 Svc:ENVSERV function:10
* 4 Svc:ENVSERV return:0 0
L:4 Svc:ENVSERV function:90
* 4 Svc:ENVSERV return:0 0
L:4 Svc:ENVSERV function:73
* 4 Svc:ENVSERV return:0 0
L:4 Svc:ENVSERV function:10
* 4 Svc:ENVSERV return:0 0
* 3 Svc:QSAMHDL return:0 0
* 3 Svc:QSAMHDL function:2
L:3 Svc:QSAMHDL return:0 0
* 3 Svc:QSAMHDL function:3
* 3 Svc:QSAMHDL return:0 0
L:3 Svc:QSAMHDL function:2
* 3 Svc:QSAMHDL return:0 0
L:3 Svc:TRANPROC function:131
L:4 Svc:ENVSERV function:10
* 4 Svc:ENVSERV return:0 0
L:4 Svc:ENVSERV function:10
* 4 Svc:ENVSERV return:0 0
L:4 Svc:ENVSERV function:10
* 4 Svc:ENVSERV return:0 0
L:4 Svc:ENVSERV function:10
* 4 Svc:ENVSERV return:0 0
L:4 Svc:ENVSERV function:10
* 4 Svc:ENVSERV return:0 0
L:4 Svc:ENVSERV function:10
* 4 Svc:ENVSERV return:0 0
L:4 Svc:ENVSERV function:10
* 4 Svc:ENVSERV return:0 0
L:4 Svc:ENVSERV function:160
* 4 Svc:ENVSERV return:0 0
L:4 Svc:ENVSERV function:110
* 4 Svc:ENVSERV return:0 0
L:4 Svc:EDITS function:400
* 4 Svc:EDITS return:0 0
L:4 Svc:EDITS function:400
* 4 Svc:EDITS return:0 0
L:4 Svc:EDITS function:400
* 4 Svc:EDITS return:0 0
L:4 Svc:ENVSERV function:10
* 4 Svc:ENVSERV return:0 0
L:4 Svc:DATABASE function:504
* 4 Svc:DATABASE return:0 0
L:4 Svc:EDITS function:400
* 4 Svc:EDITS return:0 0
L:4 Svc:EDITS function:400
* 4 Svc:EDITS return:0 0
L:4 Svc:DATABASE function:3
* 4 Svc:DATABASE return:0 0
L:4 Svc:ENVSERV function:160

```

TRANPROC - EDITR
Translation Service

FCB function code 131 =
Production Send

The results of the find for WebSphere Data Interchange Translation service. The function code indicates this is a translation service function code 131 which is a production send.

WDI 3.3 Service Director Trace

```
editrace.txt - Notepad
File Edit Format View Help
L:2 Svc:OSAMHDL function:10
  L:3 Svc:ENVSERV function:90
    * 3 Svc:ENVSERV return:0 0
  * 2 Svc:OSAMHDL return:0 0
L:2 Svc:ENVSERV function:72
  * 2 Svc:ENVSERV return:0 0
* 1 Svc:UTILSRV return:8 384
L:1 Svc:ENVSERV function:90
  * 1 Svc:ENVSERV return:0 0
L:1 Svc:ENVSERV function:90
  * 1 Svc:ENVSERV return:4 0
L:1 Svc:ENVSERV function:90
  * 1 Svc:ENVSERV return:0 0
L:1 Svc:ENVSERV function:90
  * 1 Svc:ENVSERV return:0 0
L:1 Svc:ENVSERV function:90
  * 1 Svc:ENVSERV return:0 0
L:1 Svc:ENVSERV function:90
  * 1 Svc:ENVSERV return:4 0
L:1 Svc:ENVSERV function:90
  * 1 Svc:ENVSERV return:0 0
L:1 Svc:ENVSERV function:90
  * 1 Svc:ENVSERV return:0 0
L:1 Svc:ENVSERV function:90
  * 1 Svc:ENVSERV return:0 0
L:1 Svc:ENVSERV function:2
  * 1 Svc:ENVSERV return:0 0
L:2 Svc:ERRORS function:-1
  L:3 Svc:ENVSERV function:13
    * 3 Svc:ENVSERV return:0 0
  L:3 Svc:ENVSERV function:11
    * 3 Svc:ENVSERV return:0 0
  * 2 Svc:ERRORS return:0 0
L:2 Svc:EDITS function:-1
  L:3 Svc:ENVSERV function:11
    * 3 Svc:ENVSERV return:0 0
  L:3 Svc:ENVSERV function:11
    * 3 Svc:ENVSERV return:0 0
  L:3 Svc:ENVSERV function:11
    * 3 Svc:ENVSERV return:0 0
  L:3 Svc:ENVSERV function:11
    * 3 Svc:ENVSERV return:0 0
  L:3 Svc:ENVSERV function:11
    * 3 Svc:ENVSERV return:0 0
  L:3 Svc:ENVSERV function:11
    * 3 Svc:ENVSERV return:0 0
  * 2 Svc:EDITS return:0 0
L:2 Svc:TRANPROC function:-1
  L:3 Svc:ENVSERV function:11
    * 3 Svc:ENVSERV return:0 0
  L:3 Svc:ENVSERV function:11
    * 3 Svc:ENVSERV return:0 0
  L:3 Svc:ENVSERV function:11
    * 3 Svc:ENVSERV return:0 0
```

ENVSERV - FXXZCSD
FCB function code 2 = Terminate

FCB function code 11 = Free main

This example shows various services called during the service director termination.

Reference

- More information can be found in the WDI V3.3 Messages and Codes Guide.



More information can be found in the WebSphere Data Interchange Version 3.3 Messages and Codes Guide.

Trademarks, copyrights, and disclaimers

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

IBM	CICS	IMS	WMO	Tivoli
IBM (logo)	Cloudscape	Informix	OS/390	WebSphere
e/Logo/business	DB2	iSeries	OS/400	xSeries
AIX	DB2 Universal Database	Lotus	pSeries	zSeries

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are registered trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel, ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a registered trademark of Linus Torvalds.

Other company, product and service names may be trademarks or service marks of others.

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements and/or changes in the product(s) and/or program(s) described herein at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectual property rights, may be used instead.

Information is provided "AS IS" without warranty of any kind. THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (e.g., IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

© Copyright International Business Machines Corporation 2006. All rights reserved.

Note to U.S. Government Users - Documentation related to restricted rights-Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract and IBM Corp.