VSE Connectors Workshop

Basic Setup of VSE e-business Connectors on Windows



Wilhelm Mild Ingo Franzki zVSE@de.ibm.com

© Copyright IBM Corporation 2007

Trademarks

References in this publication to IBM products or services do not imply that IBM intends to make them available in every country in which IBM operates. Consult your local IBM business contact for information on the products, features, and services available in your area.

AIX*, APPN*, CICS*, CICS/VSE*, CICS, DB2*, DB2 Connect, DB2 Universal Database, DFSORT, DRDA*, e-business logo*, Enterprise Storage Server, FlashCopy, HiperSockets, IBM*, IBM logo*, IBM eServer, iSeries, Language Environment*, MQSeries*, Multiprise*, pSeries, S/390*, S/390, System z, Parallel Enterprise Server, TotalStorage, VSE/ESA, z/VSE, VTAM*, WebSphere*, xSeries, z/OS, z/VM, zSeries and Distributed Relational Database Architecture are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both.

Linux is a registered trademark of Linus Torvalds.

Java and all Java-related trademarks and logos are trademarks or registered trademark of Sun Microsystems, Inc.

UNIX is a registered trademark in the United States and other countries, licensed exclusively through The Open Group.

Microsoft, Windows, Windows NT, Visual Basic and the Windows flat logo are Trademarks of Microsoft Corporation.

Other trademarks and registered trademarks are the properties of their respective companies.

IBM hardware products are manufactured from new parts, or new and used parts. Regardless, our warranty terms apply. This equipment is subject to all applicable FCC rules and will comply with them upon delivery.

Information concerning non-IBM products was obtained from the suppliers of those products. Questions concerning those products should be directed to those suppliers.

All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

(C) Copyright IBM Corporation 2007 All Rights Reserved.

z/VSE e-business Connectors

Real time access to various VSE resources is implemented using Connector technologies to embed the VSE services. These implementations are needed by today's heterogeneous IT environments. The Connector technology implements a software component on the remote system and an access component on the VSE host.



The VSE e-business Connectors included in VSE are platform independent because the remote software component is written in Java. These e-business Connectors are fully compatible with WebSphere technologies, and provide real time access to:

- •VSAM
- •Power queues
- Librarian
- •Console
- •ICCF
- •DL/I (VSE 2.7)

In Addition to the functions of these Connectors, the applications running on VSE/ESA 2.6 and newer can:

- •Transparently access remote data
- •Synchronize different data stores

•Use the Virtual Tape support delivered with VSE/ESA 2.6 and newer

By using Java technologies these Connectors enable the integration of VSE data into distributed processes and Web transactions in a heterogeneous environment and exploit the advanced functions of the IBM WebSphere Application Server.

Overview of possibilities with Java-Based Connector



- ► Access to VSE Resources with Java technologies.
- ► VSE Connector client is part of VSE/ESA 2.5 and newer
- ► VSAM file access via batch or CICS (for mapped files)
- Integration of VSAM data with Web Application Server using VSAM JDBC driver

Workshop objective

- Setup of Java-Based Connector for VSAM access
- VSE Navigator Setup and usage



To be able to work with VSE e-business Connectors to access VSAM files we have to setup:

♦On z/VSE

1. VSE Connector server

The VSE Connector server is the software component handling incoming requests from remote VSE e-business Connector programs.

*****On the Java platform

(i.e. Windows or Linux on zSeries)

1. VSE Connector Client

The VSE Connector Client contains a Java class library with java beans which contain functions for the connection and session management and data access functions for the various resources on VSE.

2. Map the VSAM files

To access VSAM data via the VSE e-business Connectors, the structure of a VSAM record must be defined – that's the mapping step.

Chapter 1. Setup Connector Server on VSE



The Connector Server is the listener in VSE for incoming requests from a VSE Connector client.

The VSE Connector server doesn't need special setup

 The VSE Connector Server was setup and started in your VSE Lab installation.

VSE is ready for incoming requests from VSE Connector Client .

Note: Steps how to setup and start the VSE Connector server are described in <u>Appendix A</u>.

Chapter 2. Software prerequisites for Windows



STEP2.1: Verification if Java environment installed

To install the VSE Connector Client, a Java Virtual Machine (JVM) must be installed in Windows.

The JVM can be installed in different flavors.

•To just run Java programs, the JRE 1.3.x or later is needed (Java Runtime Environment),

•to develop/compile Java programs, JDK 1.3.x or higher is needed (Java Developer Kit, which includes the JRE).

To verify if a Java Virtual Machine is installed, open a Command prompt and enter command:

java -version

You should see something like:

Java version "1.4.2"

Java(TM) 2 Runtime Environment, Standard Edition

If the messages above are shown go to Chapter 3.

STEP2.2 Install a Java Environment

If following message (or similar) is shown:

java' is not recognized as an internal or external command, operable program or batch file.

then your system has no Java virtual machine (Runtime Environment) installed or it can not be found in the path.

To install a Java Virtual machine download the code from IBM:

http://www.ibm.com/developerworks/java/

or download a SUN Version from http://www.sun.com Install the downloaded JDK 1.4.x. or later.

Chapter 3. Setup VSE Connector Client



Download of VSE Connector Client code

The download of VSE Connector Client for this workshop was already done from the VSE homepage.

The Steps are described in **Appendix B**

STEP3.1 Install VSE Connector client

Use a Windows command prompt and change to the Connector Client directory to invoke the installation batch file.

Enter: C:

cd C:\conntmp\connector

install.bat (with command java install you get the same result)

The installer will guide you trough the installation process of the VSE Connector client. The default directory where the VSE Connector Client will be installed is **C:\vsecon**

The VSE connector client consists of:

- a Java class library (Java Beans) with the Connector functions
- a detailed HTML documentation about the functions and possibilities
- concepts for development, deployment and implementation
- a lot of commented and ready to run samples

STEP3.2 Verify settings of CLASSPATH and VSECON

To have access to the functions of VSE Connector client you need to update the CLASSPATH and the environment variable VSECON.

Verify the VSECON variable :

In a Windows command prompt enter: :

 set VSECON
 It must show the installation path of VSE Connector Client: C:\vsecon

To update the Classpath for the Windows system, update it in

START – Settings - Control panel – System - Andvanced Tab - Environment Variables in the SYSTEM Variables

Close the dialogs with **OK** (not with Cancel) to activate the settings. Open a **new** command prompt to verify the settings (set VSECON).

Verify the CLASSPATH:

In a Windows command prompt enter:

set CLASSPATH

The classpath must contain following files:

.;%VSECON%\VSEConnector.jar;%VSECON%\ibmjsse.jar; %VSECON%\cci.jar

To update the Classpath for the Windows system, update it in **START – Settings** - **Control panel – System** - **Andvanced Tab - Environment Variables**

Close the dialogs with **OK** (not with Cancel) to activate the settings. Open a **new** command prompt to verify the settings.

Chapter 4. Verify installation of Connector Client

STEP4.1: Verify VSE Connector Client documentation

To verify that the VSE Connector Client is installed properly,

open the VSE Connector Client HTML Documentation (VSEConnectors.html) in the <vsecon> root directory

START -> Programs -> VSE Connectors -> VSEConnectors.html

The Main HTML page will be opened.

You will see the main page of the online documentation of VSE e-business Connectors which is part of the VSE Connector Client



STEP4.2: Verify that VSE Connector Client can work with z/VSE resources

Make sure the VSE Connector server is started on VSE. To see how to start VSE Connector server see *Setup Connector Server on VSE* at the beginning of this workshop.

From the main local HTML page of VSE Connectors, (see <u>STEP4.1</u>) Click:

Applications in section Programming Concepts in right frame. Click on Examples -> How to work with Librarian Objects

The commented source code explains the steps of this program.

The link **LibrApiExample.java** shows the entire source code of the program. The program accesses VSE Libraries via the VSE Java-Based Connector and retrieves all the Library names from VSE, the sub libraries for PRD2 and the members for Library PRD2.CONFIG.

All samples are stored on your PC in the VSE Connector Client directory: *C:\vsecon\samples*

Make sure you know the *IP* address of the VSE system, the VSE userid and password to use.

On a Windows command prompt navigate to the C:\vsecon\samples directory

Start the program by typing:

LibrApiExample.bat

Use VSE IP: 192.168.23.11

Use the userid:**TExx** and password:**teamxx** (**xx- is team nr.**) The system will give you the results like shown:

🖾 C:\WINNT\System32\cmd.exe 📃	
LibrListener: listAdded(), member = SRU\$SYS.PROC	
LibrListener: listAdded(), member = MILD.PROFDIT1	
LibrListener: listAdded(), member = SYSA.PROFDIT1	
LibrListener: listAdded(), member = ATCSTR00.SAVE	
LibrListener: listAdded(), member = IJBDAT.SAVE	
LibrListener: listAdded(), member = VTMMDL.SAVE	
LibrListener: listAdded(), member = VTMNSNA.SAVE	
LibrListener: listAdded(), member = IESLIBDF.Z	
LibrListener: listAdded(), member = IESPLGIN.Z	
LibrListener: listAdded(), member = IESUSERS.Z	
LibrListener: listAdded(), member = IESVCSRU.Z	
LibrListener: listAdded(), member = JMSTART.Z	
LibrListener: listAdded(), member = JMSTOP.Z	
LibrListener: listAdded(), member = JMSTPNXT.Z	
LibrListener: listEnded()	
Number of members in CONFIG : 47	
First member in CONFIG : DTSECTXS.A	
Number of records: 394	
Logical record length: 80	
Creation date: Mon Oct 29 15:28:42 CET 2001	
Last update: Mon Feb 17 14:56:37 CET 2003	
Member format is text	
DTSECTXS.A downloaded.	
	▶ //.

11

Accessing VSAM data from remote systems



- Using the same technology, Java-Based Connector, the access to VSAM files in VSE can be realized using the VSE Connector client component and interface with it in different ways:
- ► With the Java beans provided by VSE Connector client directly
- ► Via a VSAM JDBC driver
- ► Via a script interface
- The access can then be done via batch or via CICS
- Various samples and descriptions are in VSE Connector client online documentation <u>STEP 4.1</u>
- ► In all cases, because VSAM access is record based, a <u>mapping</u> must be done prior to can access VSAM data from remote 12

Mapping of a VSAM cluster



VSE/VSAM Record structure (i.e. from EMPPROG.COBOL Copy book).

Mapping characteristics:

- ► No changes to VSAM data
- Mapping information stored in a repository in VSAM (VSE.VSAM.MAPPING.DEFS)
- Possible data types: STRING, binary, signed number, unsigned number, packed data
- Multiple maps and views (subset of map fields) supported
- Mapping of VSAM cluster can be done using:
 - IDCAMS RECMAP command on z/VSE,
 - VSE Navigator or a self written Java program
 - Maptool standalone Java program,

► We will use the Maptool in this workshop

<u>Chapter 5. Mapping a VSAM file</u> for access through batch or CICS

To have access via Connectors to a VSAM file, the structure of the VSAM record must be defined. This allows the Connector to translate the different types of fields within a VSAM record from the VSE EBCDIC format into the ASCII format on the remote platform. The definition of a set of fields for a VSAM record is called the **map** (similar to a relational table definition). The elements of a map are **columns**. A subset of the columns can be grouped in a **view**, were a view refers to columns of a map rather then having definitions of the columns. Multiple maps and views can be defined for a VSAM cluster.

STEP5.1 Mapping specification for a VSAM cluster

The VSAM cluster used is FLIGHT.ORDERING.FLIGHTS.TEAMxx

The Map **FLIGHTS_MAP** we have to define is based on the VSAM cluster with following record structure. This structure is the base for our column definitions.

These definitions are typically done in **COBOL copy books**.

We will use the **Flights Cobol Copy** book and the Java tool, called **Maptool** to define the map in VSE.

Offset	Length	Туре	Кеу	Field Name	Description
0	4	UNSIGNED	yes	FLIGHT_NUMBER	Flight Number
4	20	STRING	no	START	Start
24	20	STRING	no	DESTINATION	Destination
44	5	STRING	no	DEPARTURE	Departure (hh:mm)
49	5	STRING	no	ARRIVAL	Arrival (hh:mm)
54	4	UNSIGNED	no	SEATS	Seats
58	4	STRING	no	RESERVED	Seats reserved
62	4	PACKED	no	PRICE	Price
66	20	STRING	no	AIRLINE	Airline

Make sure you know the IP address of the VSE system, userid and password.

Create a Map using Maptool and a COBOL Copybook

STEP5.2 Mapping of FLIGHT.ORDERING.FLIGHTS.TEAMxx using Maptool

The Map for a VSAM file can be defined using different ways as you know from the first part of this workshop.

We use here, the Standalone Java tool called Maptool which you have to download and install from the VSE home page:

http://www-1.ibm.com/servers/eserver/zseries/zvse/

Click on "**Downloads**" Download it and unzip it – no install process needed.

To start the Maptool, use a Windows command prompt window and enter:

C: cd C:\maptool run.bat

You will see a window like this:



Import the Cobol Copy Book *Flights.cb* with the definitions of the VSAM record.

Click:

Import a map from

COBOL Copybook

Click on Start. Navigate to directory C:\maptool Click on the Cobol Copybook file *Flights.cb* Click: **Open** The generated mapping definitions will be shown. As description the lines from the Cobol Copy book are generated.

Now changes could be done if needed (different Column names than the Field names in Cobol Copy book, change the map or parts of a VSAM record by deleting some columns, or have another description for the columns).

Verify and compare the definitions of the fields with the definitions in Step 5.1.

🐐 MapToolGui - F:\VSE\vsecon27\maptool\Flights.cb 🛛 🗐 💶 🗙						
Fieldname	Length	Offset	Туре		Description	
FLIGHTNUM	4	0	UNSIGN	05 FLIGHTNUME	BER PIC 9(8) CC)MP. 0000
START	20	4	STRING	05 START	PIC X(20).	00006000
DESTINATION	20	24	STRING	05 DESTINATION	N PIC X(20).	00007000
DEPARTURE	5	44	STRING	05 DEPARTURE	PIC X(5).	0008000
ARRIVAL	5	49	STRING	05 ARRIVAL	PIC X(5).	00009490
SEATS	4	54	UNSIGN	05 SEATS	PIC 9(8) COMP.	00010000
RESERVED	4	58	STRING	05 RESERVED	PIC 9999.	00011000
PRICE	4	62	PACKED	05 PRICE	PIC 9(6) COMP-3.	00012000
AIRLINE	20	66	STRING	05 AIRLINE	PIC X(20).	00013000
Field options Map options						
<u>A</u> ppend	<u>E</u> dit <u>D</u> elete		Loc <u>C</u> har	okup a field nge map info	Export Insert <u>m</u> ap	Import <u>n</u> ew Exit

This definitions can now be exported in different formats, in XML or as Java structure or as definitions for the VSE system – that's what we need. To export the definitions to VSE and create there a map do:



Click:



The window below is shown asking you for the authorization ID, and parameters for VSE.

🕌 Export a	map to VSE host	a ×
Host : Port	192.168.23.11	: 2893
User	TExx	
Password	*****]
Catalog	VSESP.USER.CATA	LOG
Cluster	IGHT.ORDERING.F	LIGHTS
Мар	FLIGHTS_MAP	
<u>_</u>	<u>K</u>	ancel
		•

At this point the map can be generated for access trough **batch** <u>(STEP</u> <u>5.3</u>) or for access trough CICS (<u>STEP 5.4</u>).

The difference is only the map name:

Map for batch access:

VSESP.USER.CATALOG/FLIGHT.ORDERING.FLIGHTS.TEAMxx/FLIGHTS_MAP

The Lab covers the access trough batch and in Appendix E you'll find the steps for the access trough CICS

Define the map in the next Steps.

STEP5.3 Generate the map for batch access



Specify the parameter, (**xx** – is your team number) VSE IP: **192.168.23.11** Port: **2893**

VSE user: **TExx** VSE Password: **teamxx**

Catalog : click ···· on the right and select, or type VSESP.USER.CATALOG

Cluster: click and select or type FLIGHT.ORDERING.FLIGHTS.TEAMxx

Map: FLIGHTS_MAP Click: OK

Now the map is defined in the specified VSE system and enables access to the VSAM file in <u>batch</u> mode.

Chapter 6. Graphical interface to z/VSE, VSE Navigator

VSE Navigator is an application based on the VSE Connector Client. A wide range of functions of the VSE e-business Connectors, are incorporated in the VSE Navigator.

With this graphical interface, VSE system resources and data can be accessed, displayed, changed and monitored.

In STEP3.3 and 3.4 you downloaded it and unzipped it from FTP server. *To run the Navigator, the Connector client must be installed on your workstation. Go to next steps to install and run it.*



Note: You can also download the VSE Navigator from the **VSE Home page**:

http://www-1.ibm.com/servers/eserver/zseries/zvse/

and the "Downloads" link on the left side

For the Lab the Navigator was downloaded already.

STEP 6.1: Install VSE Navigator

To install VSE Navigator, use a Windows command prompt Enter: **C**:

cd C:\conntmp\navi install.bat

NOTE: Install Navigator in the same directory with VSE Connector client: *C:\vsecon*

Navigator can be used to work with multiple VSE systems at the same time from a graphical interface.

STEP 7.2: First configuration of VSE Navigator

✓ At the first start of VSE Navigator you'll be asked for settings.

To start it, in a command prompt enter:

C:

cd C:\vsecon run.bat

You can set your look and feel, local directories and are asked for local applications.

✓ Local applications

✓ For Browser enter: C:\Program Files\Internet Explorer\IEXPLORE.EXE

✓ For file compare entry, type **windiff**

These settings can be changed later on at any time.

Now the Navigator is ready and the next step is to define the VSE systems That Navigator should have access to.

STEP 6.3: Define host and access it

✓ After Navigator is started define a VSE system to it:

click: Configuration

Hosts

enter Name you'd like to give to your VSE in *Description* enter the IP address: **192.168.23.11** and userid: TExx xx – is your team number click **SAVE CLOSE** Right click on the Host Icon and then **Connect**

Enter the *password* (team<u>xx</u>) and then OK.

Make sure VSE Connector Server (STARTVCS) is running on VSE.

I f you expand the Host folder you get a window like the one below. Right click on the host Folder shows various functions available.



STEP 6.4: Display a VSAM file with Navigator

To look at the same VSAM file you worked with the Java Program, after You started VSE Navigator and connected to the VSE system: Click:

- expand VSAM Folder
- expand VSESP.USER.CATALOG folder
- expand the cluster FLIGHT.ORDERING.FLIGHTS.TEAM<u>xx</u>
- right click on the MAP FLIGHTS_MAP
- click: Display VSAM data

At this time, you used the same function as with the Java program, with the advantage of the graphical possibilities of VSE Navigator.

In the view below if you right click on a **ROW** and then **change**, you are able to do changes based on individual columns direct in the VSAM file.

🖶 ¥SE/ESA Navigator - vseworkshop									a_o×
File Edit Selected Configuration Functions He	lp	5							
	00	🧕 💷 💷 🗕							
🕸 VSE/ESA Navigator 🔺	FLIGHT_NU.	. START	DESTINATION	DEPARTURE	ARRIVAL	SEATS	RESERVED	PRICE	AIRLINE
🗄 📶 Datajoin	34	Munich	Stuttgart	17:10	18:20	50	23	120	Lufthansa
FFDemo	123	Atlanta	Los Angeles	14:00	18:30	300	134	250	Delta Airlines
E − VSE-z800	284	Atlanta	Miami Beach	07:40	11:20	250	156	180	United Airlines
E-W VSEFlex	1234	Stuttgart	Atlanta	09:25	20:30	180	155	400	Lufthansa
E VSEFRAN2	4711	New York	Atlanta	12:00	15:30	300	45	250	Delta Airlines
E- vseredb									2010/01/2010/01/2010
E- VSETP									
🗉 🖬 vsewm									
E seworkshop									
🕀 😨 Librarian									
E-III POWER									
E-C VSAM									
E- #VSAM#CICS.DBDCCICS									
E-D VSAM.MASTER.CATALOG									
E- D VSESP.USER.CATALOG									
E-O CICS.CSD									
E-CICS.DBDCCICS.DFHDMFA									
E-CICS.DBDCCICS.DFHDMFB									
E-CICS.DUMPA									
E-CICS.DUMPB									
E− C CICS.GCD									
Den dicsilico	1								
E-CICS.RSD									
E-CICS.TD.INTRA									
DEFAULT.MODEL.ESDS.SAM									
DFHTEMP									
E-O DOS.WORKFILE.SYS001.RECC									
DOS.WORKFILE.SYS002.RECC									
E-B FLIGHT.ORDERING.FLIGHTS.TE									
- FLIGHT.ORDERING.FLIGHTS.TE									
E-O FLIGHT.ORDERING.FLIGHTS.TE									
E-B FLIGHT.ORDERING.FLIGHTS.TE									
F- C FLIGHT ORDERING FLIGHTS TE									
F- C FLIGHT ORDERING FLIGHTS TE									
F- C FLIGHT ORDERING FLIGHTS TE									
FLIGHT ORDERING FLIGHTS TE									
E-O FLIGHT ORDERING FLIGHTS TE									
E-O FLIGHT ORDERING FLIGHTS W									
F- ELIGHTS MAR									
									1
5 row(s) received									

In Appendix F you can setup a java sample to access the same VSAM file.

Summary

Major Steps to install VSE Connector Client on a workstation

✓ install Java Runtime Environment (JRE) or
 ✓ Java Developer Kit (JDK)
 ✓ free download from SUN or IBM
 ✓ version 1.3.x or later
 http://www.ibm.com/developerworks/java/

download VSE Connector client
 http://www-1.ibm.com/servers/eserver/zseries/zvse/downloads/#vsecon

✓ install VSE Connector client✓ at a command prompt enter: java install

✓ download VSE Navigator
 http://www-1.ibm.com/servers/eserver/zseries/zvse/downloads/#navi
 ✓ install VSE Navigator (see <u>Appendix F</u>)
 ✓ at a command prompt enter: java install

Appendix A. Setup Connector Server on VSE



Windows

z/VSE

The Connector Server is the listener in VSE for incoming requests from a VSE Connector client.

STEPA.1: Setup correct TCP/IP system ID for VSE Connector server

 ✓ TCP/IP for VSE/ESA has a system ID (default is 00), specified in TCP/IP startup Job (the default name is TCPIP00):

// EXEC IPNET, SIZE=IPNET, PARM='ID=00', INIT=.....

✓ To use the TCP/IP services from another partition (i.e. VSE Connector Server) this partition has to 'know' the system ID of TCP/IP. This is specified in the VSE Connector server startup job as follows:

// OPTION SYSPARM='nn' - where nn is the system ID.

★ **Recommendation**: Copy the skeleton SKVCSSTJ from ICCF lib 59 in your primary library, adjust and submit it.

★ It's the startup job of VSE Connector server.

STEPA.2: Start VSE Connector server

At the VSE console enter:

r rdr, STARTVCS

several messages will appear on the console. The server is ready for e-business when the following messages are shown:

IESC1002I FINISHED STARTUP OF VSE CONNECTOR SERVER IESC1003I WAITING FOR CONNECTIONS OF CLIENTS...

Now, VSE is ready for incoming requests from VSE Connector Client.

24

Appendix B: <u>Download VSE Connector Client</u> <u>from the Internet</u>



Open the VSE Homepage with a web browser:

http://www-1.ibm.com/servers/eserver/zseries/zvse/ Click on "Downloads" on the left side .

The page opened contains the VSE Connector Client and the tools that can be downloaded for free.

In the upper left corner you see **VSE Connector Client** which represents the remote component of the e-business Connectors.

VSE Connector Client
Updated: 03/2005
→ Detailed Description
→ Documentation
→ Latest APARs and PTFs
→ Frequently asked guestions
↓ Installation
Note: This tool is part of VSE Connectors (5686-CF7-35): IESINCON.W in PRD1.BASE
APAR level: z/VSE 3.1: GA, VSE/ESA 2.7/2.6: PQ88809

🛃 Download now (zip, 7.2MB)

Please note that the PTF mentioned here must be applied on z/VSE to allow VSE Connectors to work properly.

Click on:

Download now

Download it to a temporary directory (i.e c:\conntmp) and unzip it.

Appendix C: Connectors Security

To access VSE resources from remote platforms, the VSE security applies.

The security is checked using Racroute requests.

Using a Basic Security manager or an Extended Security Manager,

allows the global or more granular check of accesses to VSE

resources. Connectors can work using secured (SSL) connections.



Appendix D: Define a file in CICS

Logon to your VSE system using the icon on your desktop:

🛛 <mark>_</mark> Session E - [32 x 80]		
File Edit View Communication Actions Windo	w Help	
🖻 🔁 🛃 🛃 🔳 🔳 🚵		
z/VM ONLINE		
0000000	ccc 777777777 0000000	
999999999	666 777777777 222222222	
999 999 E 999 999 E	666 777 222 222 666 F / S 777 222	
999 999 6	566 777 222	
999999999	56666666666666666666666666666666666666	
999 E 999 E	566 666 777 222 566 666 777 222	
999 6	666 666 777 222 666 666 777 222	
999	66666666 777 22222222	
	IBM WSC z/VM	
	Version 4 Release 3.0	
Fill in your USERID and PASS	SWORD and press ENTER	
(Your password will not appe	ear when you type it)	
PASSWORD ===>	Session E - [24 x 80]	8_D×
COMMAND ===> d vse27_		
MA a		
💬 Connected to remote server/host 9.82.1.89 usir		
	Enter the character of your selection and press the ENIER key	1:
	A DBDCCICS B PRODUCTOS	
	C OLDCICS	
	E	
	F	
	==>	
	M <u>A</u> e	19/008
	$ {\mathfrak G}^{\rm Q} $ [Connected to remote server/host 9.82.1.89 using port :]IBM 4019 LaserPrinter on LPT1:	11.

On Command line Enter: d vse310 A On the CICS logon screen enter:

User: **TE**<u>xx</u> (xx- is your team number 00 - 10)

Password: team<u>xx</u>

You are now in the Interactive User Interface (IUI) of VSE.



Press: **PF6** (to go into the CICS command mode Hit **CEDA def File (FLIGH<u>xx</u>)**

🖲 <mark>5</mark> Session A - [24 x 80]		
File Edit View Communication	Actions Window Help	
🖻 🖻 🛍 🚮 🖷	» 🐏 🛍 🌆 🚾 🚵 📾	
DEF FILE		
OVERTYPE TO MO	DDIFY	CICS RELEASE = 0411
CEDA DEFine	File()	
File	==> FLIGHxx	
Group	==> VSESPG	
DEScription	==> Flight file defini	tion in CICS
VSAM PARAMETE	ERS	
DSNAme	==> FLIGHT.ORDERING.FL	IGHTS.TEAMxx
Password	==>	PASSWORD NOT SPECIFIED
Lsrpoolid	==> 01	1-15 None
Catname	==> VSESPUC	
DSNSharing	==> Noreqs	Noreqs Allreqs Modifyreqs
STRings	==> 001	1-255
Nsrgroup	==>	
SHr4access	==> Key	Key Rba
REMOTE ATTRIE	BUTES	
REMOTESystem	n ==>	
REMOTEName	==>	
RECORDSize	==>	1-32767
+ Keylength	==>	1-255
MESSAGES: 2 S	SEVERE	
		SYSID=CIC1 APPLID=DBDCCICS
PF 1 HELP 2 CON	M 3 END 6 CR	SR 7 SBH 8 SFH 9 MSG 10 SB 11 SF 12 CNCL
M <u>A</u> a		02/022
Connected to remote server/h	nost tn3270 using port 23	IBM 4019 LaserPrinter on LPT1:

Enter the required parameters.

Hit **PF8** to go to the next screens and type **YES** to all Operations:

🛡 <mark>-</mark> Session A - [24 x 80]		
File Edit View Communication	Actions Window Help	
🖻 🖻 🏝 📠 🔳	🤌 ف ا 🔜 🚵 🖮 🖮	
OVERTYPE TO MO	DIFY OR PRESS ENTER TO	EXECUTE CICS RELEASE = 0411
CEDA DEFine	File(FLIGHXX)	
+ INITIAL STATU	S	
STAtus	==> Enabled	Enabled Disabled Unenabled
Opentime	==> Firstref	Firstref Startup
BUFFERS		
DAtabuffers	==> 00002	2-32767
Indexbuffers	==> 00001	1-32767
DATATABLE PAR	AMETERS	
Table	==> No	No Cics User
Maxnumrecs	==>	16-16777215
DATA FORMAT		
RECORDFormat	==> V	V F
OPERATIONS		
Add	==> yes	No Yes
Browse	==> yes	No Yes
DELete	==> yes	NoYes
REAd	==> Yes _	Yes No
+ Update	==> yes	No Yes
		SYSID=CIC1 APPLID=DBDCCICS
PF 1 HELP 2 COM	3 END 6 CR	SR 7 SBH 8 SFH 9 MSG 10 SB 11 SF 12 CNCL
M <u>A</u> a		20/026
Gi Connected to remote server/ho	ost tn3270 using port 23	IBM 4019 LaserPrinter on LPT1:

Press: ENTER to complete the definition

To activate the definition **install FLIGH<u>xx</u>** in the group you specified in the definition:



Appendix E: Use the access through CICS



After you defined the map for batch access you could now use the same definitions to define a map for CICS access. But for CICS we want to use a map with fewer columns defined.

The Map **FLIGHTS** should contain the following field definitions:

Offset	Length	Туре	Кеу	Field Name	Description
0	4	UNSIGNED	yes	FLIGHT_NUMBER	Flight Number
66	20	STRING	no	AIRLINE	Airline
4	20	STRING	no	START-AIRPORT	Start
24	20	STRING	no	DESTINATION	Destination
62	4	PACKED	no	PRICE	Price

Generate a map for CICS access (continued)

Your Window in Maptool should look like this:

🚔 MapToolGui - F:\VSE\vsecon27\maptool\Flights.cb 🔤 📃 🗆 🗶					
Fieldname	Length	Offset	Туре	Description	
FLIGHTNUM	4	0	UNSIGN	05 FLIGHTNUMBER PIC 9(8) COMP.	0000
START	20	4	STRING	05 START PIC X(20).	00006000
DESTINATION	20	24	STRING	05 DESTINATION PIC X(20).	00007000
DEPARTURE	5	44	STRING	05 DEPARTURE PIC X(5).	0008000
ARRIVAL	5	49	STRING	05 ARRIVAL PIC X(5).	00009490
SEATS	4	54	UNSIGN	05 SEATS PIC 9(8) COMP.	00010000
RESERVED	4	58	STRING	05 RESERVED PIC 9999.	00011000
PRICE	4	62	PACKED	05 PRICE PIC 9(6) COMP-3.	00012000
AIRLINE	20	66	STRING	05 AIRLINE PIC X(20).	00013000
Field options			Map optic	ons	
Append Edit Insert Delete		ikup a field Export	Import <u>n</u> ew		

You need to delete the Columns: Departure, Arrival, Seats, Reserved

To delete a Field definition click on it and then click DELETE

To change a field content, click on the field and do the change.

Generate a map for CICS access (continued)

After finishing the changes, click:

Export

VSE VSAM MAP

Export

The window below is shown asking you for the authorization and parameters for VSE.

≜ Export a	map to VSE host 🛛 🖉 📉 🗙				
Host : Port	192.168.23.11 : 2893				
User	TExx				
Password	*****				
Catalog	#VSAM.#CICS.DBDCCICS				
Cluster	FLIGH01				
Мар	FLIGHTS				
<u>O</u> K <u>C</u> ancel					
Disconnected.					

Specify the parameter, (**xx** – is your team number) VSE IP: **192.168.23.11** Port: **2893**

VSE user: **TExx** Password: **teamxx**

Catalog : click on the right and select, or type #VSAM.#CICS.DBDCCICS

Cluster: click ... and select or type FLIGHxx

Map: FLIGHTS Click: OK

Now the map is defined in the specified VSE system and enables access to the VSAM file via the <u>CICS</u> with the ID DBDCCICS.

Access FLIGHTS.ORDERING.FLIGHTS via CICS

Since most of VSAM files are active in a CICS environment, to update them via Connectors, the VSAM cluster must be defined with Share Option 4 or the access with the Connectors must be done via CICS.

We'll use the same sample **VsamDisplayExample.java** from <*vsecon*>*samples**com**ibm**vse**samples*

To can have access via CICS the file must be defined in CICS . The file FLIGHTS.ORDERING.FLIGHTS.TEAMxx is defined in CICS as **FLIGH**<u>xx</u> How these definitions can be done is described in <u>Appendix D</u>.

The map must be defined for this access (Step 5.4) – specific here is the catalog name for Connector access which has a name of: #VSAM.#CICS.<CICSAPPLID>

In our case the CICS used is the one having APPLID DBDCCICS.

All samples are stored on your system in the directory: <vsecon>\samples The Java source code is in: <vsecon>\samples\com\ibm\vse\samples

The Map **FLIGHTS** defined in <u>STEP 5.3</u> contains the following field definitions:

Offset	Length	Туре	Кеу	Field Name	Description
0	4	UNSIGNED	yes	FLIGHT_NUMBER	Flight Number
66	20	STRING	no	AIRLINE	Airline
4	20	STRING	no	START-AIRPORT	Start
24	20	STRING	no	DESTINATION	Destination
62	4	PACKED	no	PRICE	Price

Modify Java program VsamDisplayExample for access trough CICS

Exercise: The name of the VSAM file and map must be modified.

The source program is in C:\vsecon\samples\com\ibm\vse\samples

- •Edit program VsamDisplayExample.java (i.e. Notepad)
 - •In a Windows command prompt enter:

C:

cd C:\vsecon\samples\com\ibm\vse\samples Notepad VsamDisplayExample.java

• Do the following changes:

•Catalog name, *catName* : **#VSAM.#CICS.DBDCCICS**

the File name, *fileName* : FLIGH<u>xx</u> where xx is your team number
the Map name, *mapName*: FLIGHTS

• Save the modified source.

•Compile the changed program

To compile the source in the command prompt enter:

cd C:\vsecon\samples

javac com\ibm\vse\samples\VsamDisplayExample.java

Note: the compile must be done from the C:\vsecon\samples directory because all samples belong to the Java package com.ibm.vse.samples

If the compile returns no errors, execute the program, enter:

C:\vsecon\samples\VsamDisplayExample.bat

C:\WINNT\System32\cmd.exe	_ 🗆 ×
Getting records from FLIGHT.ORDERING.FLIGHTS Records in file FLIGHT.ORDERING.FLIGHTS : Record 0: FLIGHT_NUMBER (Key) : 34 START : New York DESTINATION : Atlanta DEPARTURE : 12:35 ARRIVAL : 14:45 SEATS : 180 RESERVED : 24 PRICE : 250 AIRLINE : Delta Airlines	▲
Record 1: FLIGHT_NUMBER (Key) : 1234 START : Muenchen DESTINATION : New York DEPARTURE : 08:30 ARRIVAL : 12:00 SEATS : 250 RESERVED : 21 PRICE : 1200 AIRLINE : Lufthansa	×
•	• /

Appendix F: Connector Sample to access VSAM data in batch

Access VSAM data via Java-Based Connector using the Java sample program VsamDisplayExample

All samples are stored in the VSE Connector Client directory: *C:\vsecon\samples*

The Java source code is in: C: \vsecon\samples\com\ibm\vse\samples

We will work with VsamDisplayExample.java

The program displays the content of a mapped VSAM file.

Start the program. In a Windows command prompt enter:

C:

cd C:\vsecon\samples

VsamDisplayExample.bat

You got an error. That's because of a wrong name of the VSAM file. In next step you'll correct it .

C:\WINNT\system32\cmd.exe	
mild Please enter password: sel2us	<u> </u>
Creating connection and USE system	
Getting records from FLIGHT.ORDERING.FLIGHTS.DEMO	
Exception in thread "main" com.ibm.vse.connector.NotFoundException:	Error: The r
fource valar. User. Chintody rulight. ONDERING. Fulghts. Denov rulights_mhr	has not been
Fund of can not been accessed.	
Command ID- 42000004	
at com ibm use connector IISEConnection evecute(IInknoun Source	
at com ibm.vse.connector ll@ConnectionLandla evenue/llanoum	Courses
at com. ibm. vse. connector. NSConnect Ionnanute.execute(Unknown)	source/
at com. 10m. vse. connector. USES ystem. execute volknown source/	
at com. ibm. vse. connector. VsEvsamhap.refreshvonknown source/	Courses)
at com. 10m. 0se. connector. USECsammap.getrr1maryReyr1e1us(onkn	own sources
at com. ibm. vse. connector. vsEconnection.execute(onknown sourc	6/ 0>
at com.ibm.vse.connector.VSEConnectionHandle.execute(Unknown	Source)
at com.inm.vse.connector.vsEsystem.execute(unknown Source)	
at com.ibm.vse.connector.VSEVsamCluster.selectRecords(Unknow	n Source?
at com.ibm.vse.connector.VSEVsamCluster.selectRecords(Unknow	n Source?
at com.ibm.vse.samples.VsamDisplayExample.main(VsamDisplayEx	ample.java:1
22)	
F: VSE/vsecon27/samples/pause	
Press any key to continue 🔤	•

Modify Java program VsamDisplayExample

Exercise: The name of the VSAM file and the map must be modified.

The source program is in C:\vsecon\samples\com\ibm\vse\samples

Edit program VsamDisplayExample.java (i.e. Notepad)

In a Windows command propmt enter:

C: cd C:\vsecon\samples\com\ibm\vse\samples Notepad VsamDisplayExample.java

• Do the following changes:

• the file name *fileName* to: FLIGHT.ORDERING.FLIGHTS.TEAMxx

where xx is your team number

• the map name mapName to: FLIGHTS_MAP

• Save the modified source.

• Compile the changed program

To compile the source in the command prompt enter:

cd C:\vsecon\samples

javac com\ibm\vse\samples\VsamDisplayExample.java

Note: the compile must be done from the C:\vsecon\samples directory because all samples belong to the Java package com.ibm.vse.samples

If the compile returns no errors, execute the program, enter:

C:\vsecon\samples\VsamDisplayExample.bat



Additional Information

• z/VSE Home Page

http://www.ibm.com/servers/eserver/zseries/zse/

 e-business Connectors User's Guide SC33-6719 http://www-1.ibm.com/servers/eserver/zseries/zvse/documentation/#conn

•VSE Connectors: Components, tools http://www.ibm.com/servers/eserver/zseries/zvse/downloads

•z/VSE modern solutions http://www.ibm.com/servers/eserver/zseries/zvse/solutions



•e-business Connectivity for VSE/ESA	SG24-5950
•e-business Solutions for VSE/ESA	SG24-5662
Servlet and JSP Programming	SG24-5755
 Linux Web Hosting with WebSphere, DB2, and Domino 	SG24-6007
 WebSphere V5 for Linux on zSeries Connectivity Handbook 	SG24-7042

zVSE@de.ibm.com