

System z Expo

October 13 – 17, 2008 – Las Vegas, Nevada



Session Title: Workshop: How to Access z/VSE data and services

Session ID: zEO53

Speaker Name: Ingo Franzki & Wilhelm Mild

Authorized
IBM | Training



© 2008 IBM Corporation

Trademarks

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

Not all common law marks used by IBM are listed on this page. Failure of a mark to appear does not mean that IBM does not use the mark nor does it mean that the product is not actively marketed or is not significant within its relevant market.

Those trademarks followed by ® are registered trademarks of IBM in the United States; all others are trademarks or common law marks of IBM in the United States.

For a complete list of IBM Trademarks, see www.ibm.com/legal/copytrade.shtml:

*, AS/400®, e business(logo)®, DBE, ESCO, eServer, FICON, IBM®, IBM (logo)®, iSeries®, MVS, OS/390®, pSeries®, RS/6000®, S/30, VM/ESA®, VSE/ESA, WebSphere®, xSeries®, z/OS®, zSeries®, z/VM®, System i, System i5, System p, System p5, System x, System z, System z9®, BladeCenter®

The following are trademarks or registered trademarks of other companies.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries. Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.

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 trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

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

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

ITIL is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.

IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency, which is now part of the Office of Government Commerce.

* All other products may be trademarks or registered trademarks of their respective companies.

Notes:

Performance is in Internal Throughput Rate (ITR) ratio based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput 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 improvements equivalent to the performance ratios stated here.

IBM hardware products are manufactured from new parts, or new and serviceable used parts. Regardless, our warranty terms apply.

All customer examples cited or described in this presentation are presented as illustrations of the manner in which some customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics will vary depending on individual customer configurations and conditions.

This publication was produced in the United States. IBM may not offer the products, services or features discussed in this document in other countries, and the information may be subject to change without notice. Consult your local IBM business contact for information on the product or services available in your area.

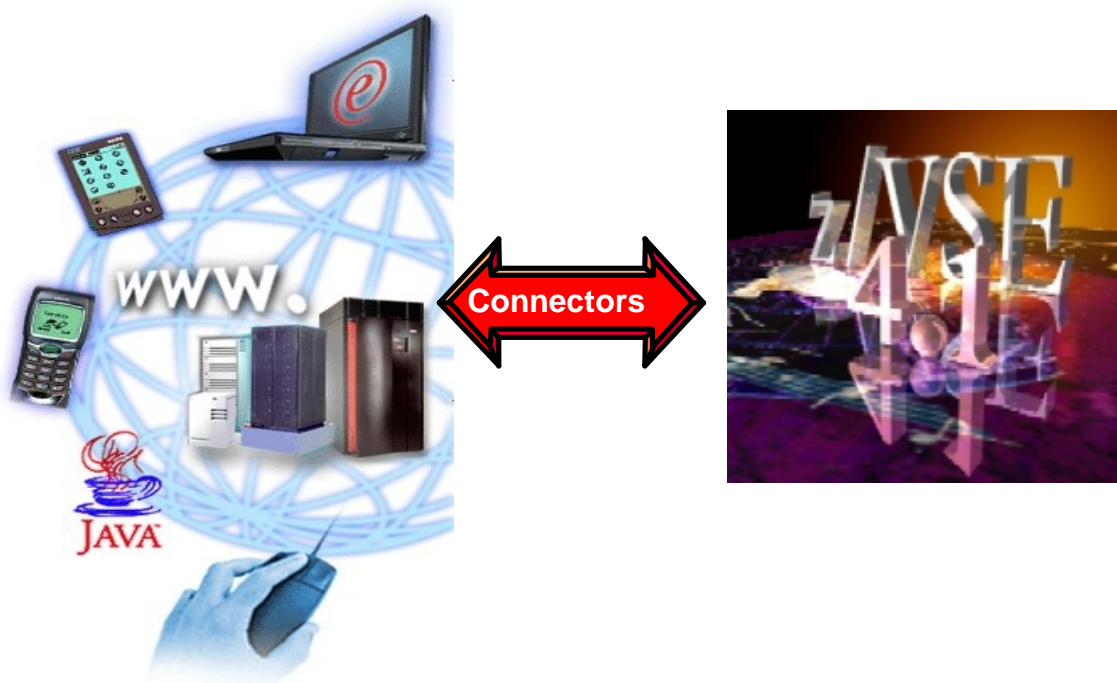
All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

Information about non-IBM products is obtained from the manufacturers of those products or their published announcements. IBM has not tested those products and cannot confirm the performance, compatibility, or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Prices subject to change without notice. Contact your IBM representative or Business Partner for the most current pricing in your geography.

VSE Connectors Workshop

Setup of z/VSE e-Business Connectors and VSAM Redirector on Windows



Wilhelm Mild, IBM
Ingo Franzki, IBM

Contact: zVSE@de.ibm.com

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 z9Parallel 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.

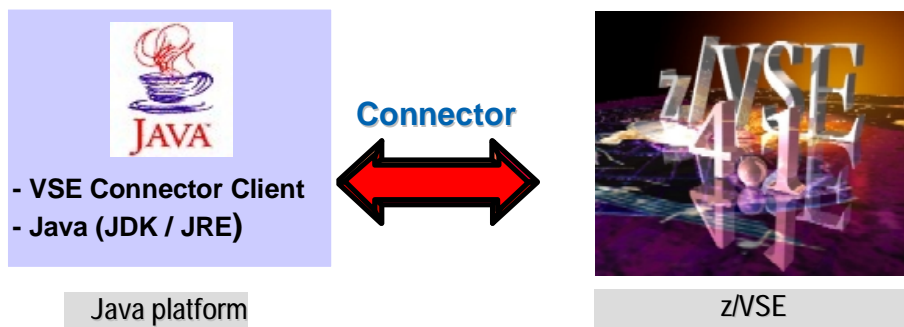
All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

© Copyright IBM Corporation 2007
All Rights Reserved.

z/VSE e-Business Connectors Overview

The z/VSE e-Business Connectors provide real time access to various VSE resources like VSAM, VSE/POWER, LIBR, ICCF and the VSE Console. Having access to VSE resources is required in today's heterogeneous IT environments.

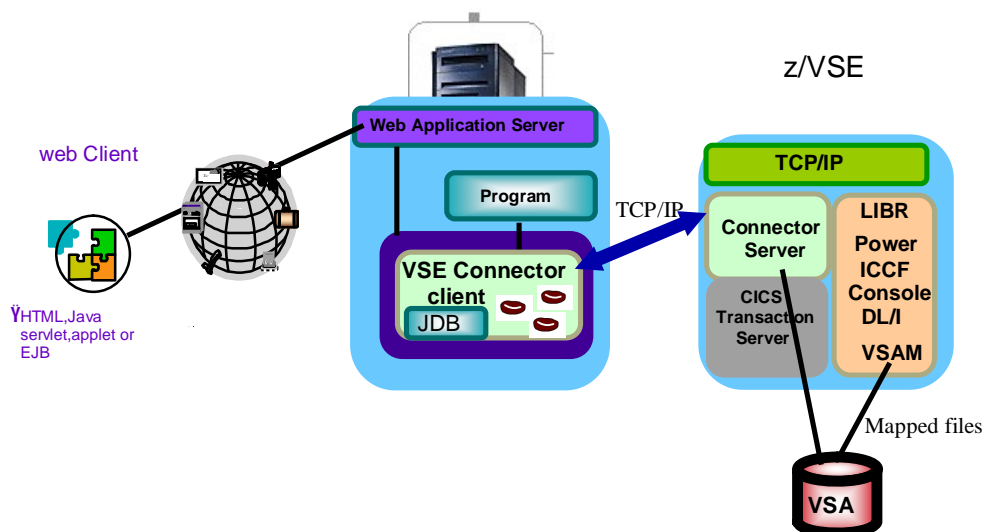
The z/VSE e-Business Connectors consists of 2 software components, one on the distributed systems and one on the VSE system:



The VSE e-business Connectors included in VSE are platform independent because the remote software component is implemented in Java. The e-business Connectors are fully compatible with WebSphere technologies.

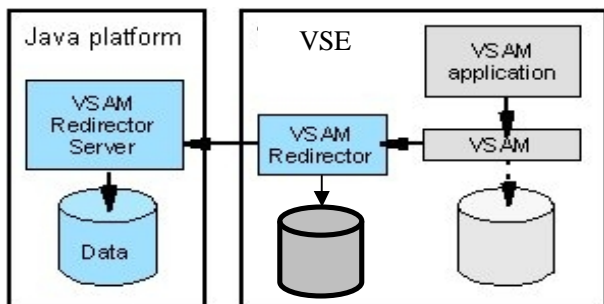
Using the e-business Connectors you can have real time access to:

- VSAM
 - access can be done through batch or CICS
 - access can be done using SQL through the VSAM JDBC driver
- VSE/POWER queues
- Librarian
- Console
- ICCF
- DL/I



VSAM Redirector Overview

The VSAM Redirector allows you to redirect all requests issued against a certain VSAM file to another file system or database on another (Java-enabled) platform. In this case VSE is the client, while the server part runs on another Java-enabled platform like Windows or Linux on System z. Furthermore the VSAM Redirector can be extended by using a Vendor Exit to do additional VSE operations (i.e. data quality checks) for a VSAM request.

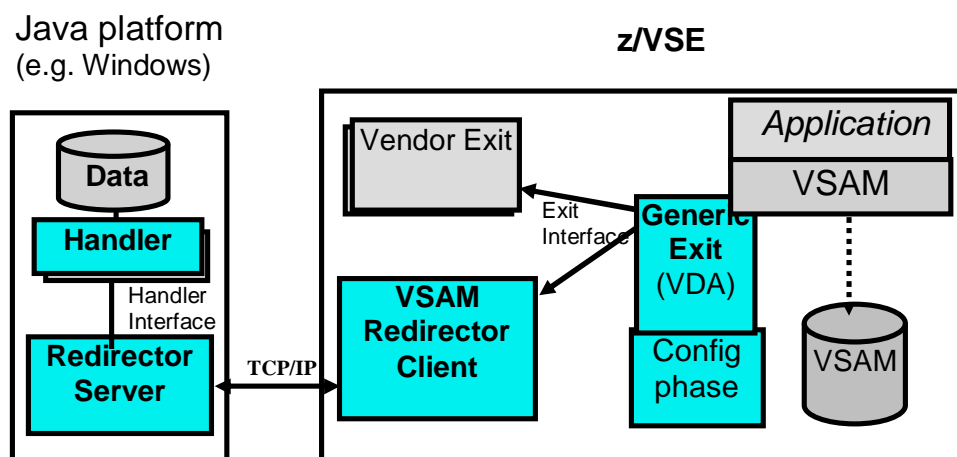


For example you could use the VSAM Redirector to synchronize a VSAM file with a DB2 database running on a Windows or Linux on System z. By using the VSAM Redirector your existing VSAM programs doesn't require any revisions.

The VSAM Redirector makes use of a so called VSAM Data Access Exit (VDA). This exit intercepts all VSAM requests. The exit gets control at OPEN, CLOSE, and for all data requests (GET, PUT, INSERT, etc.).

A configuration table is used to define which VSAM files are being "redirected". The VSAM Redirector Client on VSE forwards all requests against a redirected file to the VSAM Redirector Server which then handles the request.

The VSAM Redirector Server is a Java application. It can run on any Java enabled platform.



Workshop objectives

Implementation of z/VSE e-Business Connectors and VSAM Redirector:

1. Install VSE Connector Client:

In the first part of the workshop we will install the VSE Connector Client package on your workstation and verify the connectivity between your workstation and VSE.

2. Access a VSAM file from your workstation

This part implements access from your workstation to a VSAM file on VSE. This includes the mapping of the VSAM record. By defining the mapping, we enable the VSE Connectors to translate the data fields from EBCDIC to ASCII or from other VSE specific data formats to a Java format.

3. Usage of VSE Navigator:

In this part you install and use the VSE Navigator tool. The VSE Navigator is Windows Explorer like tool, which allows you to easily browse through the VSE file systems and resources.

4. Installation and configuration of VSAM Redirector

This part implements VSAM Redirector. We will use an existing sample application FFST that accesses a VSAM file. Without changing the application, we will setup VSAM Redirector to let the application access a DB2 database, which is running on your workstation. This includes the configuration of VSAM Redirector, creation of the tables in the database and loading the data from VSE into the database.

Chapter 1.) Software prerequisites for Windows

Step 1.1: 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 1.4.x or later is needed (Java Runtime Environment)
- To develop/compile Java programs, JDK 1.4.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 the following command:

```
java -version
```

You should see something like:

```
Java version "1.4.2"  
Java(TM) 2 Runtime Environment, Standard Edition
```

If you see messages like above, please continue with **Error! Reference source not found..**

Step 1.2: 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.

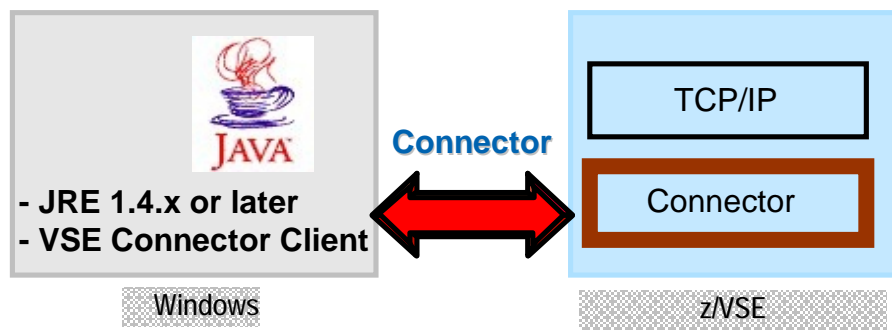
To install a Java Virtual Machine download the code from IBM:

<http://www.ibm.com/developerworks/java/> or download a SUN Version from <http://www.sun.com>.

After downloading, you need to install the downloaded JDK 1.4.x. or later.

Chapter 2.) Setup VSE Connector Server on VSE

This chapter sets up the server part of the Java-based connector.



The VSE Connector Server is the server part of the Java-based connector. It is the TCP/IP listener that sits there and waits for clients to connect. The server provides access to different kind of VSE resources:

- VSAM
 - access can be done through batch or CICS
 - access can be done using SQL through the VSAM JDBC driver
- VSE/POWER queues
- Librarian
- Console
- ICCF
- DL/I

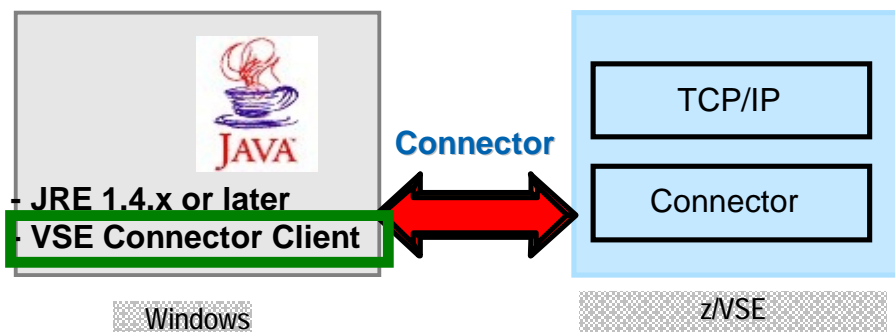
The VSE Connector Server comes pre-installed as part of the VSE installation and does not need any special configuration on VSE. Per default, there is a startup job placed in the VSE reader queue called STARTVCS.

The VSE Connector Server was already setup and started prior to the workshop. It is ready for incoming requests from VSE Connector Clients.

Please see Appendix A on page 32 for more details about how to tailor the startup job and other configuration members.

Chapter 3.) Setup VSE Connector Client

This chapter installs and sets up the client part of the Java-based connector.



The VSE Connector Client package comes as part of the VSE installation as a W-book. However, you can also download the latest version from the VSE homepage.

The download of VSE Connector Client was already done prior to the workshop and copied to your workstation into “C:\conntmp\connector”.

Please see Appendix B on page 33 for more details about how to download the VSE Connector Client package.

Step 3.1: Install the VSE Connector Client

Open a Windows command prompt and change to the directory where the installation package for the VSE Connector Client is located (C:\conntmp\connector). Invoke the installation batch script setup.bat to start the installation.

```
Enter: C:
       cd C:\conntmp\connector
       setup.bat           (with command java -jar setup.jar you get the same result)
```

A graphical installer will show up and will guide you through the installation process of the VSE Connector Client.

Note: Please install the VSE Connector Client into “C:\vsecon\”, not in the default directory suggested by the installer.

The VSE Connector Client consists of:

- Ÿ A Java class library (Java Beans) – with the connector functions
- Ÿ A detailed HTML documentation about the functions and possibilities
- Ÿ Concepts for development, deployment and implementation
- Ÿ A lot of ready to run samples including source code.

Step 3.2: Verify CLASSPATH and VSECON environment variables

To have access to the functions of VSE Connector Client, some environment variable needs to be set correctly:

- CLASSPATH - The path where Java is searching for class files
- VSECON - The directory where VSE Connector Client has been installed into

Verify the VSECON environment variable:

Open a Windows command prompt and enter:

```
SET VSECON
```

It should display the installation directory of VSE Connector Client: **C:\vsecon**

To update the VSECON environment variable on Windows, open the “System” applet in the Control Panel:

**START à Settings à Control Panel à System
à Advanced Tab à Environment Variables**

Look in the SYSTEM Variables for the variable named VSECON and change its value accordingly. Close the dialogs with OK (not with Cancel) to activate the settings. Open a **new** command prompt to verify the changes (SET VSECON).

Verify the CLASSPATH environment variable:

Open a Windows command prompt and enter:

```
SET CLASSPATH
```

It should at least contain the following entries separated by semi colons ‘;’:

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

Note: instead of %VSECON% there could also be the full qualified path (e.g. C:\vsecon\). This is also OK.

To update the CLASSPATH environment variable on Windows, open the “System” applet in the Control Panel:

**START à Settings à Control Panel à System
à Advanced Tab à Environment Variables**

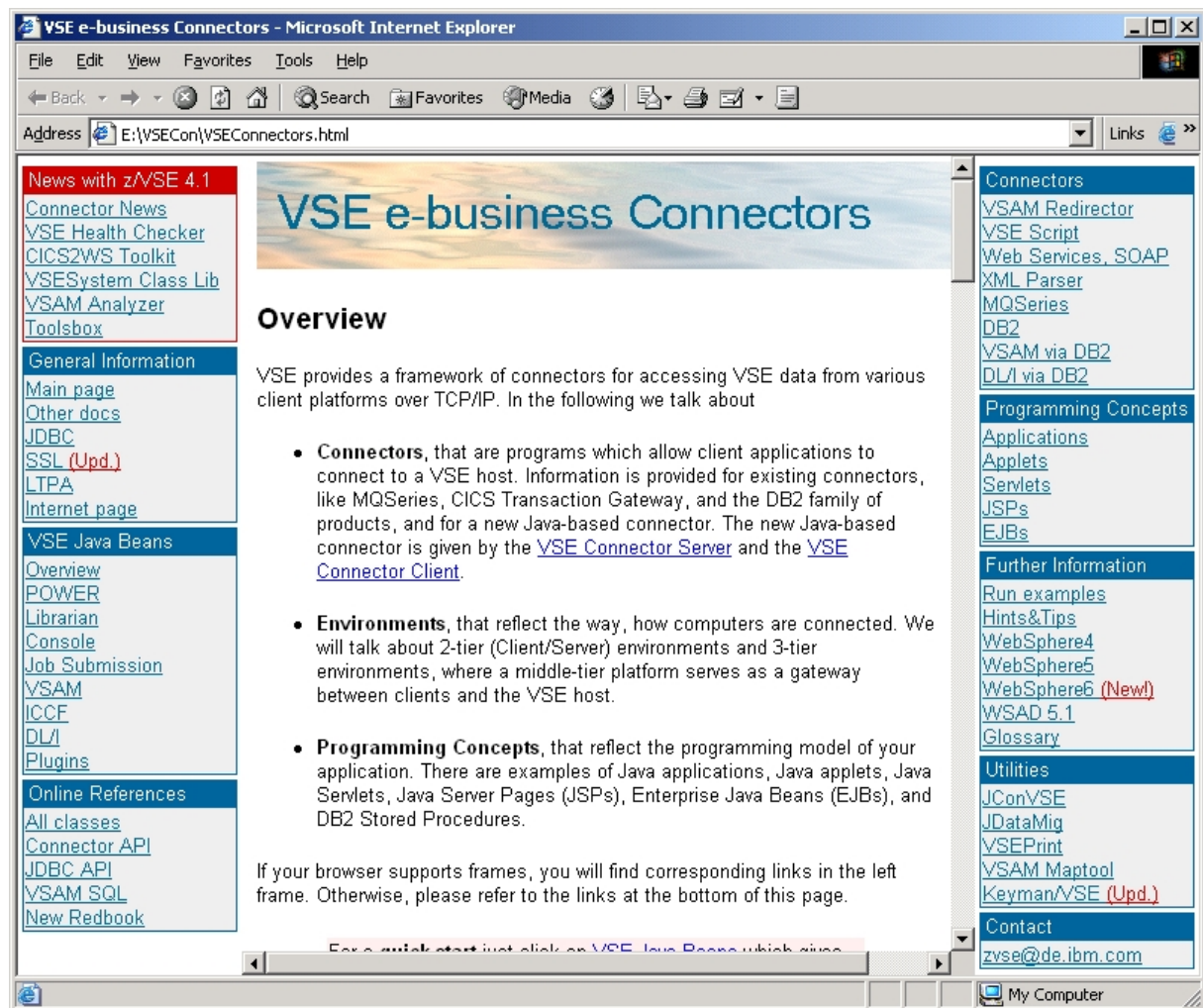
Look in the SYSTEM or USER Variables for the variable named CLASSPATH and change its value accordingly. Close the dialogs with OK (not with Cancel) to activate the settings. Open a **new** command prompt to verify the changes (SET CLASSPATH).

Step 3.3: Verify installation of VSE Connector Client

To verify that the VSE Connector Client is installed properly, open the VSE Connector Client online documentation (VSEConnectors.html) in the installation directory (C:\vsecon\). Alternatively, you can open the online documentation using the Windows start menu:

START à Programs à VSE Connector Client à Online Doc and Samples

The Main page of the VSE Connector Client documentation will be opened in a web browser:



You may browse through the documentation by clicking on the various links on the left and right side.

Step 3.4: Verify that the VSE Connector Client can work with VSE resources

Before you try to connect to VSE, make sure the VSE Connector Server is started on VSE. To see how to start the VSE Connector Server, please see Chapter 2.).

From the VSE Connector Client online documentation (see Step 3.3:) click “**Applications**” in section Programming Concepts on the right side. Then click on “**Examples**” à “**How to work with Librarian Objects**”.

You will see the fully explained source code of one of the sample programs that are part of the package. The link “**LibrApiExample.java**” shows the entire source code of the program. This sample 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 member names for library PRD2.CONFIG.

All samples are stored on your PC in the VSE Connector Client directory:

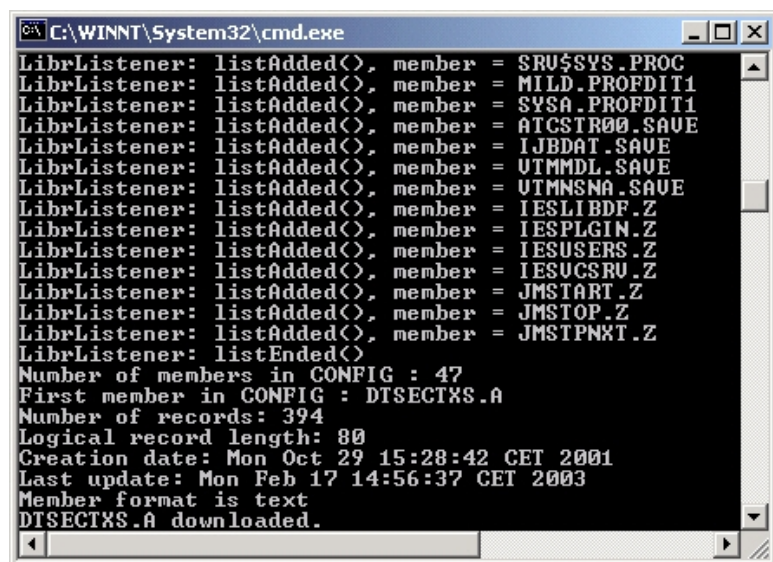
C:\vsecon\samples

Open a Windows command prompt and navigate to the **C:\vsecon\samples** directory. Then start the sample program by typing:

LibrApiExample.bat

Use VSE IP: **192.168.23.11** (Note: the IP address may be different)
 Use the user-id: **TExx** (where xx is your team number)
 Password: **teamxx** (where xx is your team number)

If you see output like this, the sample has successfully accessed the VSE system. If you get an error message or exception, see the message text for more information.

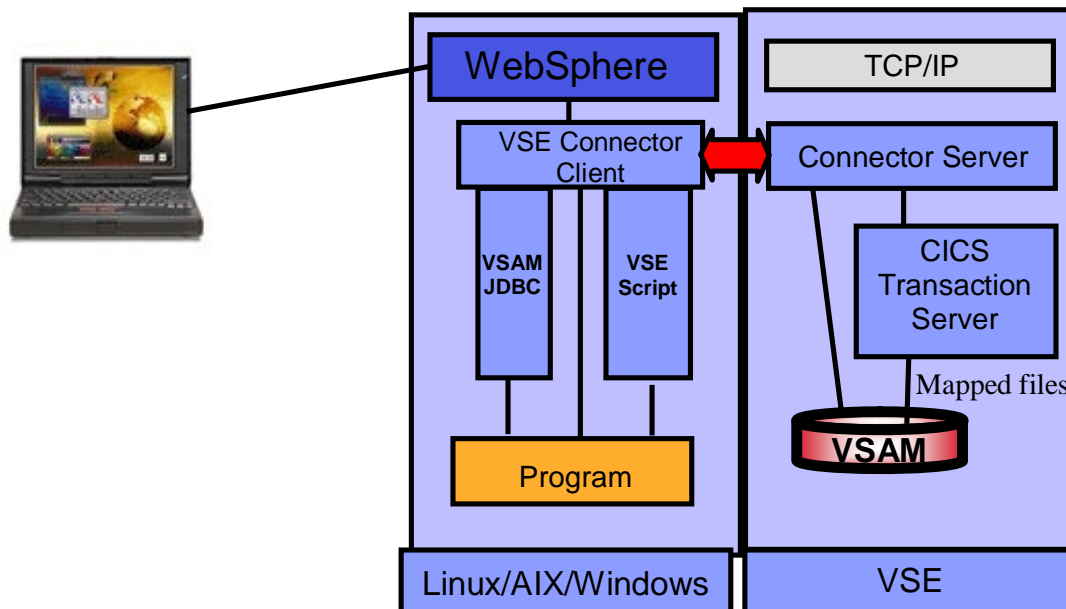


```

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 = UTMMDL.SAVE
LibrListener: listAdded(), member = UTMNSNA.SAVE
LibrListener: listAdded(), member = IESLIBDF.Z
LibrListener: listAdded(), member = IESPLGIN.Z
LibrListener: listAdded(), member = IESUSERS.Z
LibrListener: listAdded(), member = IESUCSRU.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.
  
```

Chapter 4.) Accessing VSAM data from remote systems

This chapter shows how the Java-based Connector can be used to access VSAM data.



Using the same technology (Java-based Connector) as in the sample we have used before, real time access to VSAM files in VSE can be realized in different ways:

- Using the Java classes provided by the VSE Connector Client directly
- Using a VSAM JDBC driver via SQL statements
- Using a script interface for non-Java applications

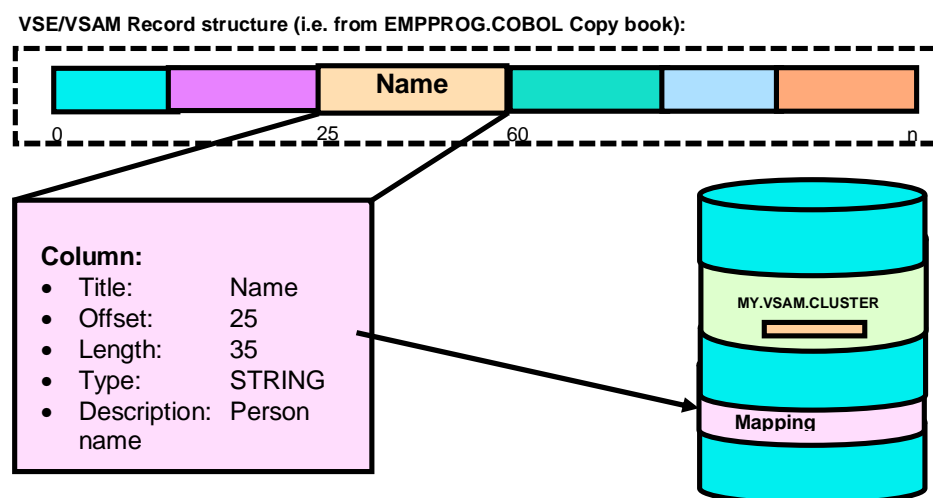
The VSAM access in VSE can then be done directly by the VSE Connector Server in batch or through a CICS rein that already owns the file(s) to be accessed.

There are also various samples for accessing VSAM files contained in VSE Connector Client online documentation, see Step 3.4: .

In all cases, because VSAM access is record based, a mapping must be defined prior to accessing VSAM data from a remote system. By defining the mapping, you enable the VSE Connectors to translate the data fields from EBCDIC to ASCII or from other VSE specific data formats like packed decimals to a Java format.

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 **fields**. A subset of the fields can be grouped in a **view**, were the fields in a view refers to (points to) fields of a map rather than having its own definitions of the field. Multiple maps and views can be defined for a VSAM cluster.

Mapping the record structure of a VSAM cluster:



Mapping characteristics:

- No changes to the VSAM data itself. The mapping definition is just meta data.
- The mapping information is stored in a system wide repository in VSAM (IESMAPD - VSE.VSAM.MAPPING.DEFS)
- Possible data types: STRING, BINARY, SIGNED and UNSIGNED numbers, PACKED and ZONED decimals
- Multiple maps and views (subset of map fields)per VSAM file ca be defined

The mapping of VSAM a cluster can be done using:

- IDCAMS RECMAP command on z/VSE
- VSE Navigator or a self written Java program
- VSE Maptool

We will use the Maptool in this workshop for defining the mapping.

Step 4.1: Record mapping for the sample VSAM file

In the workshop we use a sample VSAM cluster called **FLIGHT.ORDERING.FLIGHTS.TEAMxx**. The cluster belongs to a imaginary application that allows to book flights. This VSAM cluster have already been defined and filled with data prior to the workshop. To be able to access this cluster a map (**FLIGHTS_MAP**) has to be defined. The record structure is usually given by a **COBOL copybook** that is used in the applications that use the file. We will use such a COBOL copybook to define the map using the VSAM Maptool.

The VSAM Maptool has already been downloaded and copied to your workstation into directory C:\maptool\ prior to the workshop.

The VSAM Maptool can be downloaded from the VSE Homepage: <http://www.ibm.com/servers/eserver/zseries/zvse/downloads/#maptool>. After downloading the ZIP file, simply extract the ZIP file into a new directory (e.g. C:\maptool\). No further installation is required. Make sure the CLASSPATH contains the VSE Connector Client JAR files.

The following table shows the record structure:

| Field name | Offset | Length | Type | Key | Description |
|---------------|--------|--------|----------|-----|--------------------------|
| FLIGHT_NUMBER | 0 | 4 | UNSIGNED | Yes | Flight number |
| START | 4 | 20 | STRING | No | Start City/Airport |
| DESTINATION | 24 | 20 | STRING | No | Destination City/Airport |
| DEPARTURE | 44 | 5 | STRING | No | Departure time (hh:mm) |
| ARRIVAL | 49 | 5 | STRING | No | Arrival time (hh:mm) |
| SEATS | 54 | 4 | UNSIGNED | No | Total number of seats |
| RESERVED | 58 | 4 | STRING | No | Number of seats reserved |
| PRICE | 62 | 4 | PACKED | No | Price in US \$ |
| AIRLINE | 66 | 20 | STRING | No | Airline |

The COBOL copybook for this file looks as follows (Flights.cb):

```

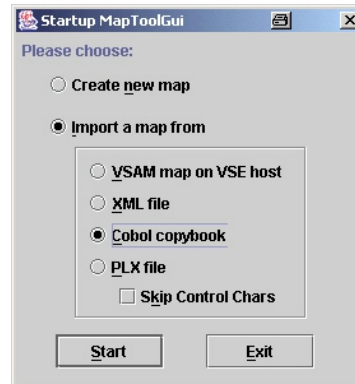
03 FLIGHTS-MAP .
   05 FLIGHTNUMBER      PIC 9(8) COMP .
   05 START              PIC X(20) .
   05 DESTINATION        PIC X(20) .
   05 DEPARTURE          PIC X(5) .
   05 ARRIVAL            PIC X(5) .
   05 SEATS              PIC 9(8) COMP .
   05 RESERVED           PIC X(4) .
   05 PRICE              PIC 9(6) COMP-3 .
   05 AIRLINE            PIC X(20) .

```


To start the VSAM Maptool, open a Windows command prompt and change to the maptool directory C:\maptool\ and run the run.bat script:

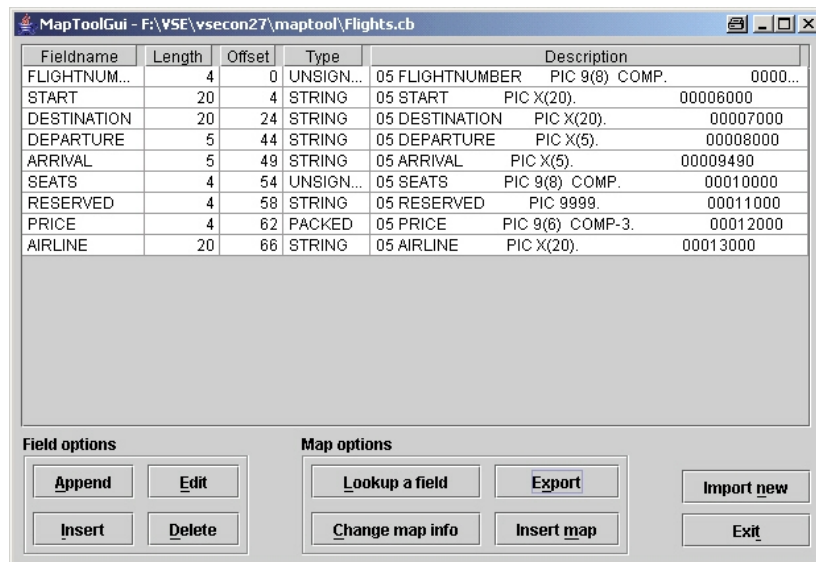
Enter: **C:**
 cd C:\maptool
 run.bat

You will see a window like this:



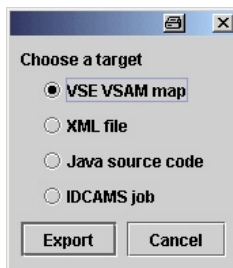
Select **“Import a map from”** à **“Cobol copybook”** and click **“Start”**. Navigate to **“C:\maptool\”** and select the file **“Flights.cb”** and click **“Open”**.

The COBOL copybook will be imported and automatically converted into a VSAM mapping. The description of each field shows the original COBOL definitions from the copybook. If needed, you can now do changes to the mapping (e.g. use different field names, add or delete fields, etc.). Verify and compare the definitions of the fields with the definitions on the previous page.

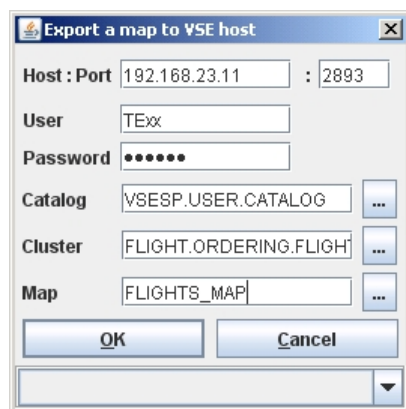


These definitions can now be exported in different formats: as XML, as Java source code or as definitions for the VSE system. We will export the mapping directly to the VSE system.

Click: **Export**
à VSE VSAM Map
Export



The dialog box as shown below is now asking for the IP address or hostname of VSE, the VSE user-id and password, as well as the names of the VSAM catalog and cluster you want to create the map for.



After you have entered the VSE user-id and password, you can use the “...” buttons on the right to brose for catalog and cluster.

Create the map for the following cluster:

VSE IP: **192.168.23.11** (the IP address may be different)
 User-id: **TExx** (where xx is your team no)
 Password: **teamxx** (where xx is your team no)
 Catalog: **VSESP.USER.CATALOG**
 Cluster: **FLIGHT.ORDERING.FLIGHTS.TEAMxx** (where xx is your team no)
 Map: **FLIGHTS_MAP**

At this point you have generated the map for access trough batch. You can additionally define the map for access trough CICS also. The difference is only the names of the catalog and cluster:

Catalog: **#VSAM.#CICS.DBDCICIS** (name of CICS region)
 Cluster: **FLIGHTS** (name of the file in CICS FCT)
 Map: **FLIGHTS_MAP**

This workshop covers the access trough batch only.

Step 4.2: Run a sample program to access VSAM data

All samples are stored on your PC in the VSE Connector Client directory:

C:\vsecon\samples

Open a Windows command prompt and navigate to the **C:\vsecon\samples** directory. Then start the sample program by typing:

VsamDisplayExample.bat

Use VSE IP: **192.168.23.11** (Note: the P address may be different)
 Use the user-id: **TExx** (where xx is your team number)
 Password: **teamxx** (where xx is your team number)

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

```

C:\WINNT\system32\cmd.exe
mild
Please enter password:
aq12ws
Creating connection and USE system ...
Getting records from FLIGHT.ORDERING.FLIGHTS.DEMO...
Exception in thread "main" com.ibm.use.connector.NotFoundException: Error: The resource USESP.USER.CATALOG/FLIGHT.ORDERING.FLIGHTS.DEMO/FLIGHTS_MAP has not been found or can not be accessed.
ErrorCode: 00020003
CommandID: 42000004
at com.ibm.use.connector.USEConnection.execute(Unknown Source)
at com.ibm.use.connector.USEConnectionHandle.execute(Unknown Source)
at com.ibm.use.connector.USESystem.execute(Unknown Source)
at com.ibm.use.connector.USEUsanMap.refresh(Unknown Source)
at com.ibm.use.connector.USEUsanMap.getPrimaryKeyFields(Unknown Source)
at com.ibm.use.connector.USEConnection.execute(Unknown Source)
at com.ibm.use.connector.USEConnectionHandle.execute(Unknown Source)
at com.ibm.use.connector.USESystem.execute(Unknown Source)
at com.ibm.use.connector.USEUsanCluster.selectRecords(Unknown Source)
at com.ibm.use.connector.USEUsanCluster.selectRecords(Unknown Source)
at com.ibm.use.samples.VsamDisplayExample.main(VsamDisplayExample.java:122)
F:\VSE\vsecon27\samples>pause
Press any key to continue . . . .
  
```

Step 4.3: Modify the sample program

In the previous step, the execution of the sample program VsamDisplayExample showd an error. This is because the sample assumes that a specific VSAM cluster is existent, which is not the case in this workshop. We will now modify the source code of the sample program to correct the name of the VSAM file and the map.

The source code of the sample program is located in

C:\vsecon\samples\com\ibm\vse\samples

Edit the file **VsamDisplayExample.java** (i.e. using Notepad):

In a Windows command prompt

Enter: C:

cd C:\vsecon\samples\com\ibm\vse\samples

Notepad VsamDisplayExample.java

Do the following changes:

- Change the value for variable **fileName** to:
“**FLIGHT.ORDERING.FLIGHTS.TEAMxx**” (where xx is your team number)
- Change the value for variable **mapName** to: “**FLIGHTS_MAP**”

Exit the editor and save the modified source.

Now we need to compile the program. To compile the source in the command prompt

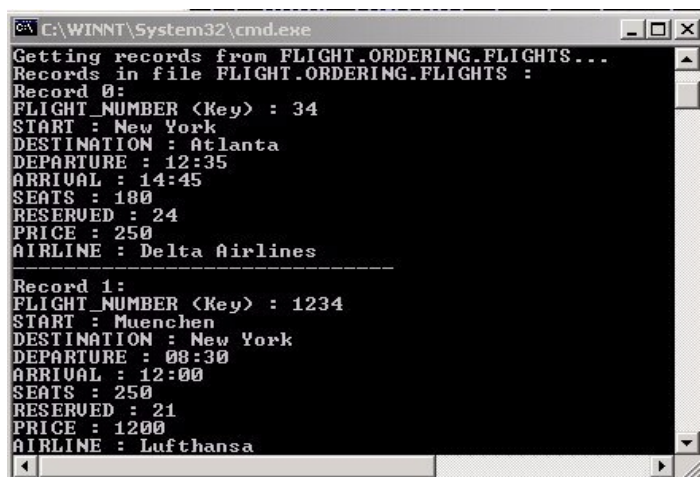
Enter: `cd C:\vsecon\samples`
`javac com\ibm\vse\samples\VsamDisplayExample.java`

Note: The compile must be done from the **C:\vsecon\samples** directory because all samples belong to the Java package **com.ibm.vse.samples**. Java class names are **case sensitive!**

If the compile returns no errors, execute the program:

Enter: `VsamDisplayExample.bat`

You should now see a display like this:



```
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
```

Chapter 5.) Install and use the VSE Navigator

The VSE Navigator is a graphical user interface for VSE 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 easily be accessed, displayed, changed and monitored, without the need for deep VSE knowledge. The VSE Navigator can be used to work with multiple VSE systems at the same time from a graphical interface.

The VSE Navigator can be downloaded from the VSE homepage. The download of VSE Navigator was already done prior to the workshop and copied to your workstation into “C:\conntmp\navi”.

Please see Appendix B on page 33 for more details about how to download the VSE Navigator package.

Step 5.1: Install VSE Navigator

To install VSE Navigator, open a Windows command prompt:

```
Enter:      C:
           cd C:\conntmp\navi
           install.bat
```

A graphical installer will show up and will guide you through the installation process of the VSE Navigator.

Note: It is recommended to install the VSE Navigator in the same directory as the VSE Connector Client: **C:\vsecon**

Step 5.2: Configuration of VSE Navigator

The basic configuration of VSE Navigator is done when you start it the first time.

To start VSE Navigator open a Windows command prompt and enter:

```
Enter:      C:
           cd C:\vsecon
           run.bat
```

Alternatively you can start it using the Windows start menu:

START à Programs à VSE Navigator à VSE Navigator

When you start VSE Navigator the first time, it will show a series of dialogs to let you configure VSE Navigator. It lets you configure settings like the look and feel, local directories and local applications.

For the workshop, please select the following local applications:

- Browser: C:\Program Files\Internet Explorer\IExplore.exe
- File compare tool: Windiff

These settings can be changed later on at any time. If you do not know which local application to use, leave it empty or accept the default.

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

Step 5.3: Configure the host(s) to connect to

You need to configure the host systems, you want to access with VSE Navigator.

Click: Configuration Hosts

Create a new Host profile, or choose an existing profile to change.

Enter the name you'd like to give to your VSE in the Description field (e.g. **VSE410**).

Enter the IP address or hostname of the VSE system: **192.168.23.11**.

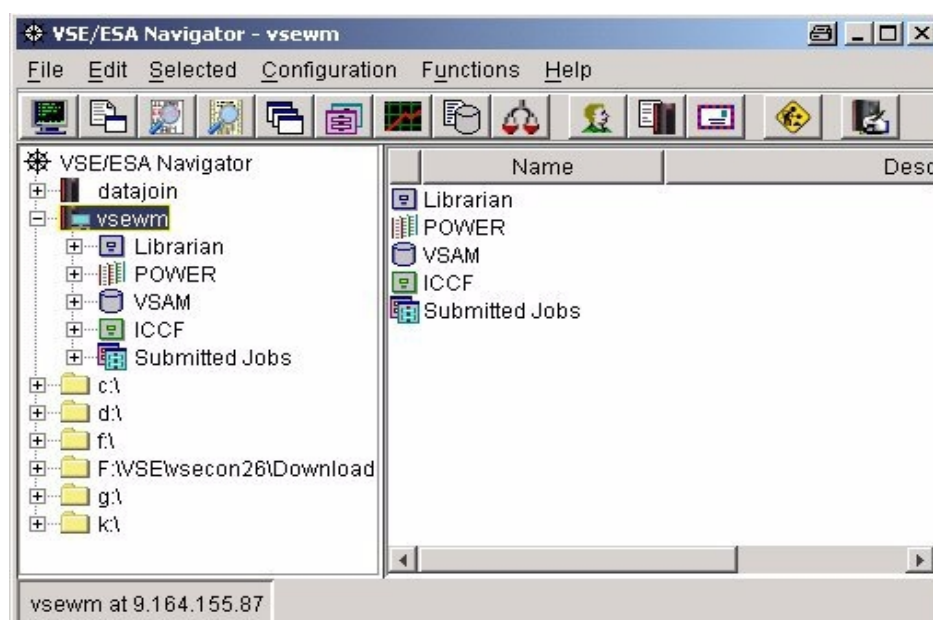
Enter the user-id: **TExx** (where xx is your team number).

Click SAVE

Click CLOSE

Now you should see an icon in the tree on the left side for your host. Right click on the host icon and click "**Connect**". Enter the password (teamxx) and press OK. If the connection was successfully, the icon should have changed to indicate that you are connected.

If you expand the host folder you get a window like the one below. Right click on a tree node (folder) shows various functions available for this node.



Step 5.4: Display a VSAM file with VSE Navigator

In this step, we will display the contents of the same VSAM cluster you worked with the Java sample program.

- Expand the VSAM Folder
- Expand the “VSESP.USER.CATALOG” folder
- Expand the cluster “FLIGHT.ORDERING.FLIGHTS.TEAMxx”
- Right click on the map “FLIGHTS_MAP”
- Click: **Display VSAM data**

At this time, VSE Navigator uses the same function under the covers as used with the Java sample program. This time the data is displayed in a graphical way, which makes it much more user friendly.

To change the content of a row, right click a row in the table on the right and select “Change”. You can do changes based on individual fields of a record in the VSAM file.

| FLIGHT_NO. | START | DESTINATION | DEPARTURE | ARRIVAL | SEATS | RESERVED | PRICE | AIRLINE |
|------------|-----------|-------------|-----------|---------|-------|----------|-------|-----------------|
| 34 | Munich | Stuttgart | 17:10 | 18:20 | 50 | 23 | 120 | Lufthansa |
| 123 | Atlanta | Los Angeles | 14:00 | 18:30 | 300 | 134 | 250 | Delta Airlines |
| 284 | Atlanta | Miami Beach | 07:40 | 11:20 | 250 | 156 | 180 | United Airlines |
| 1234 | Stuttgart | Atlanta | 09:25 | 20:30 | 180 | 155 | 400 | Lufthansa |
| 4711 | New York | Atlanta | 12:00 | 15:30 | 300 | 45 | 250 | Delta Airlines |

You may now also play with other functions provided by VSE Navigator.

Keep in mind, any function provided by VSE Navigator, can also be done in a self written Java program. This means, you can include that functionality in your own programs, tools or web applications.

Chapter 6.) Setup VSAM Redirector

In this chapter we will setup VSAM Redirector. We will use an existing sample application FFST that accesses a VSAM file. Without changing the application, we will setup VSAM Redirector to let the application access a DB2 database, which is running on your workstation. This includes the configuration of VSAM Redirector, creation of the tables in the database and loading the data from VSE into the database.

The VSAM Redirector package comes as part of the VSE installation as a W-book. However, you can also download the latest version from the VSE homepage. The download of VSAM Redirector Server was already done prior to the workshop and copied to your workstation into “C:\vsamredir”.

Please see Appendix B on page 33 for more details about how to download the VSAM Redirector Server package.

Step 6.1: Install VSAM Redirector Server on Windows

Open a Windows command prompt and change to the directory where the installation package for the VSAM Redirector Sever is located (C:\vsamredir). Invoke the installation batch script install.bat to start the installation.

Enter: **C:**
cd C:\vsamredir
setup.bat (with command **java -jar setup.jar** you get the same result)

A graphical installer will show up and will guide you trough the installation process of the VSAM Redirector Server.

Note: Please install the VSAM Redirector Server into “C:\ **vsamredir** \”, not in the default directory suggested by the installer.

The VSAM Redirector Server consists of:

- The VSAM Redirector Server code
- Detailed HTML documentation about the functions and possibilities
- Sample handlers, one for relational database DB2 and one for HTML
 - The HTML handler allows, to collect data inserted into a VSAM cluster, on a HTML page
 - The DB2 handler allows the synchronization of a VSAM cluster with a remote DB2 database or to redirect all VSAM requests to a DB2 database
- Documentation how to develop your own handlers.

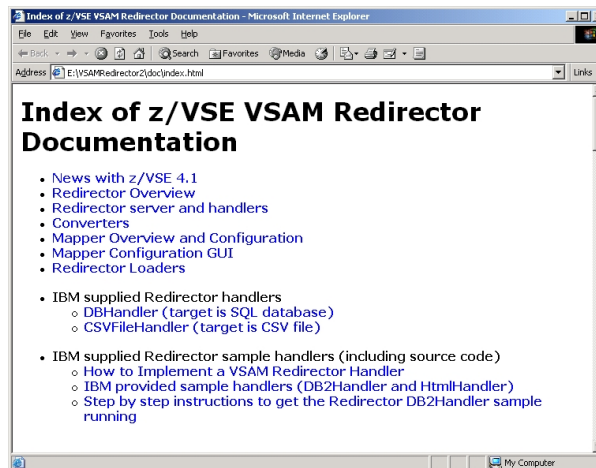
Note: The VSE Redirector Server does NOT require the VSE Connector Client or any other VSE Connector component to be installed.

Step 6.2: Verify the VSAM Redirector Server installation

To verify that the VSAM Redirector Server was installed properly, open the HTML Documentation using the windows start menu:

**START à Programs à
VSE VSAM Redirector Server à
Online Help**

The Main HTML page will be opened:



Step 6.3: Start and operate VSAM Redirector Server

To start the VSAM Redirector Server, open a Windows command prompt:

Enter: **C:**
 cd C:\vsamredir
 run.bat

```

C:\WINNT\system32\cmd.exe
Licensed Materials - Property of IBM
(C) Copyright IBM Corp. 1998, 2000. All Rights Reserved.

US Government Users Restricted Rights -
Use, duplication or disclosure restricted by
GSA ADP Schedule Contract with IBM Corp.

VSAMRedirectorServer starting...
Loaded converter for type: PACKED
Loaded converter for type: DATETIME
Loaded converter for type: INTEGER
Loaded converter for type: FLOAT
Loaded converter for type: FIXEDTEXTNUMBER
Loaded converter for type: BIT
Loaded converter for type: BINARY
Loaded converter for type: HEXCHAR
Loaded converter for type: STRING
Loaded converter for type: FLOATTEXTNUMBER
Loaded converter for type: ZONED
Loaded converter for type: S2V
Loaded converter for type: TOD
Loaded 13 converters.
Aug 22, 2007 10:33:34 AM - Listening socket created on port 2387
Enter 'quit' to stop the server
Aug 22, 2007 10:33:34 AM - Waiting for connections...

```

Once started, the VSAM Redirector Server can be operated using the following commands:

- status - shows the status of the server
- stop x | all - stop client with number 'x' (show in status) or stop all clients
- quit - stop all clients and exit the server

Please leave the VSAM Redirector Server running; we will need it later on in this workshop. If you have stopped the server, please restart it.

Chapter 7.) Setup VSE to redirect VSAM requests

In this chapter we will setup VSE to redirect VSAM requests fore a particular VSAM file to a database running on your workstation.

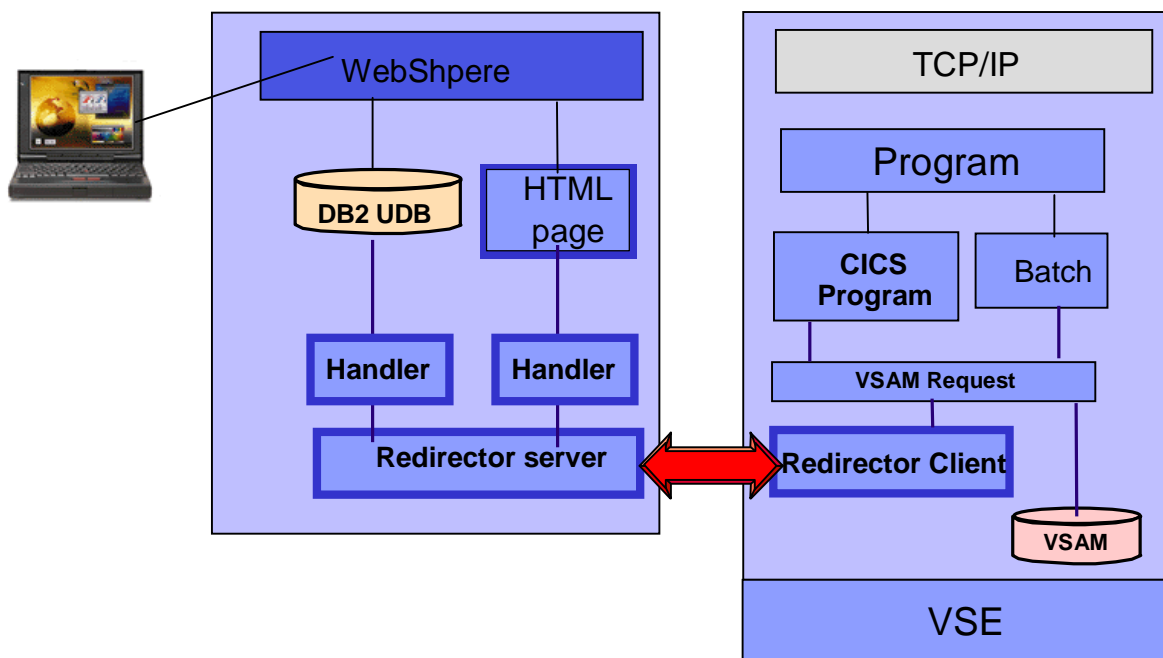
Initial state:

Assume you have a VSAM file that contains data (i.e. FFSTORES.DEMO.CLUSTER.TEAMxx)

Assume furthermore you have applications that work with that VSAM file (i.e. FFST a CICS Transaction accessing FFSTORES)

Goal:

Redirect all VSAM requests coming from applications on VSE to this VSAM file to a DB2 UDB database running on your workstation. This allows you to access the DB2 data with existing VSE applications (without changing them), and to easily share the data with other applications in the distributed environment.



Relational database tables have columns with specific characteristics. To be able to move VSAM data into a database table, we must know the internal structure and format of the VSAM records. Therefore we need a mapping containing all fields of a VSAM record including their position in the record, its length and data type.

The easiest way to create such a mapping is to use **VSAM Maptool** or **VSE Navigator** by importing an existing COBOL copybook. We have done that already in this workshop in chapter Chapter 4.) .

The following table shows the record structure of the sample file:

| Field name | Offset | Length | Type | Key | Description |
|------------|--------|--------|----------|-----|-------------------------------|
| STOREID | 0 | 6 | STRING | Yes | Store Number |
| STORENAME | 6 | 25 | STRING | No | Name of the store |
| LOCSTREET | 31 | 25 | STRING | No | Street name |
| LOCCITY | 56 | 25 | STRING | No | City name |
| LOCZIP | 81 | 10 | STRING | No | ZIP code of the city |
| LOCCOUNTRY | 91 | 25 | STRING | No | Country name |
| LOCREP | 116 | 20 | STRING | No | Name of contact person |
| STOREN1 | 136 | 4 | UNSIGNED | No | Internal store number 1 |
| STOREN2 | 140 | 4 | UNSIGNED | No | Internal store number 2 |
| LDATE | 144 | 10 | STRING | No | Date |
| WEBPIC1 | 154 | 20 | STRING | No | Filename of picture 1 for web |
| WEBPIC2 | 174 | 20 | STRING | No | Filename of picture 2 for web |
| ACODE | 194 | 10 | STRING | No | Access code |
| FILLER | 204 | 6 | STRING | No | unused |

The COBOL copybook for this file looks as follows:

```

03 FSTIO-MAP .
   05 STORE-ID           PIC X(6) .
   05 STORE-NAME        PIC X(25) .
   05 LOC-STREET        PIC X(25) .
   05 LOC-CITY          PIC X(25) .
   05 LOC-ZIP           PIC X(10) .
   05 LOC-COUNTRY       PIC X(25) .
   05 LOC-REP           PIC X(20) .
   05 VAL1              PIC 9(8) COMP .
   05 VAL2              PIC 9(8) COMP .
   05 DATE              PIC X(10) .
   05 WEB-PIC1          PIC X(20) .
   05 WEB-PIC2          PIC X(20) .
   05 A-CODE            PIC X(10) .
   05 FILLER            PIC X(6) .

```

The VSAM Redirector provides a tool called "Create DB2 Tables" that requires the mapping information to be supplied in an XML file. The mapping definitions can be exported from VSE Navigator or from VSAM Maptool into an XML file (i.e. ffstores.xml). The "Create DB2 Tables" tool reads the mapping and automatically creates database tables according to the mapping.

For this workshop all required DB2 tables has already been created in DB2 on your workstation:

- FFSTORES_MAP - contains the mapping information
- FFSTORES - contains the data that has been loaded from VSAM

Please see Appendix D on page 35 for more details on how to use the “Create DB2 Tables” tool, and how to load the VSAM data into the database table.

Step 7.1: Configure the VSAM Redirector Client on VSE

The configuration of VSAM Redirector Client on VSE is done using a configuration member that has to be assembled and cataloged as a PHASE. Based on these settings the VSAM Redirector Client will be invoked for requests on a particular VSAM file or not.

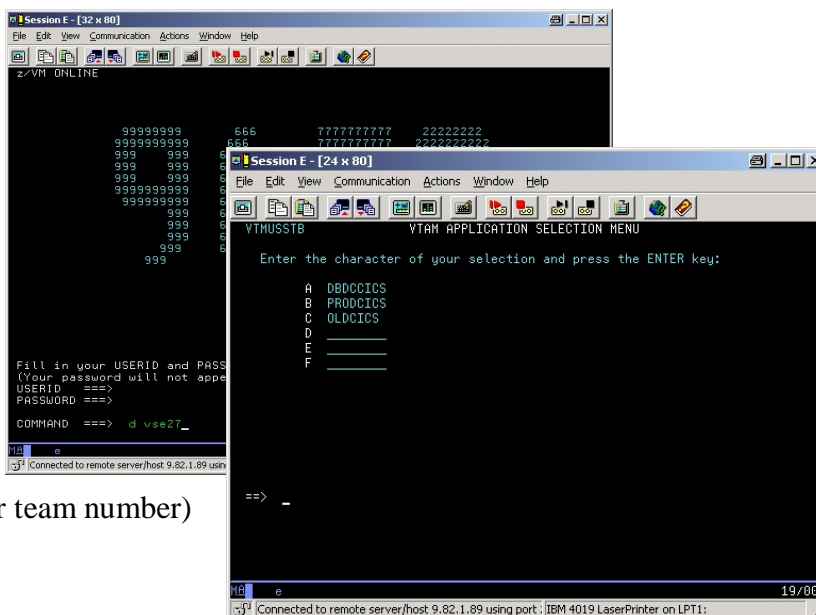
Logon to your VSE system using the 3270 icon on your desktop:

On Command line enter:

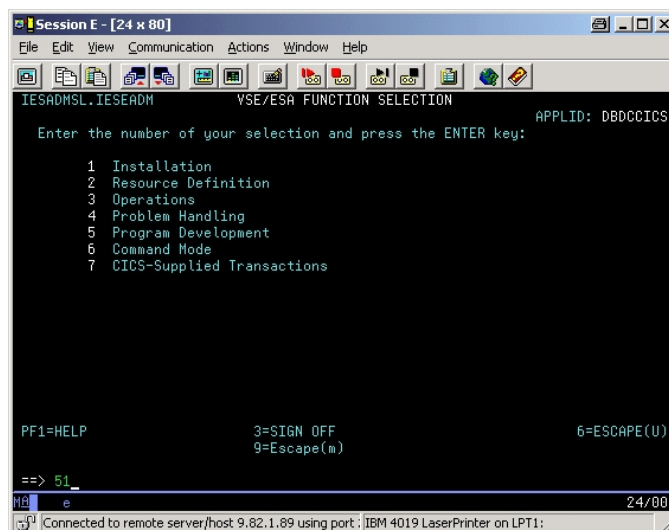
D VSE410 (on z/VM only)
A

On the logon screen enter:

User: **TExx** (xx- is your team number)
Password: **TEAMxx**



You are now in the Interactive Interface main panel of VSE.



Enter: **51**
(to go into your primary development library)

```

Session E - [24 x 80]
File Edit View Communication Actions Window Help
IESLIBI PROGRAM DEVELOPMENT LIBRARY

ENTER the numbers of the Libraries you want to access

PRIMARY..... 21          You may read and write in this library
SECONDARY.... _____  You may only read from this library

SPECIFY optionally

PREFIX..... _____    To list only members whose names begin
                           with the specified prefix

SELECT which library you want to display

OPTION..... 1             1 = Primary Library
                           2 = Secondary Library

PF1=HELP      2=REDISPLAY  3=END      4=RETURN

MA e 06/023
Connected to remote server/host 9.82.1.89 using port: IBM 4019 LaserPrinter on LPT1:

```

Hit the enter key to see the members in your primary library.

To configure the VSAM Redirector Client, we have to edit and regenerate the configuration phase (IESRDCFG.PHASE). In this configuration table we have to define which VSAM cluster should be redirected to which remote location, and optionally decide which requests.

For configuring the VSAM Redirector Client a skeleton in ICCF Library 59 is shipped, called **SKRDCFG**. For the workshop, it was copied to your primary library prior to the workshop. The definitions for the redirected file were extracted into member **RDCFGxx** (where xx is your team number). This was done to be able to build one configuration table containing the settings from all attendees of this workshop.

We want to redirect the requests for:

VSAM cluster: **FFSTORES.DEMO.CLUSTER.TEAMxx**
In VSAM catalog: **VSESP.USER.CATALOG**

Edit skeleton **RDCFGxx** and change the settings in your configuration member.

Enter: **1 <enter>**

(in the left margin of **RDCFGxx**)

```

Session E - [24 x 80]
File Edit View Communication Actions Window Help
IESLIBI PRIMARY LIBRARY PAGE 1 of 1

PRIMARY (READ/WRITE): 21          PREFIX:
OPTIONS:  1 = EDIT   2 = CHANGE  3 = PRINT  4 = COPY   5 = DELETE
           6 = RENAME 7 = SUBMIT  8 = COMPILE 9 = DISPLAY

OPT MEMBER NAME  NEW NAME  NEW LIB  LAST ACCESSED  OWNER  PASSW  PRIVATE
 1  RDCFG01      _____  _____  03/10/2004  MILD
-  SKRDCFG      _____  _____  03/10/2004  MILD

PF1=HELP      2=REFRESH    3=END      4=RETURN      6=ADD MEMBER
                9=SORT.DATE 10=SORT.NAME 11=SORT.SIZE 12=LIST QUEUE

LOCATE MEMBER/LIST QUEUE PREFIX ==> _____ MEMBER PREFIX (PF2) ==> _____

MA e 10/022
Connected to remote server/host 9.82.1.89 using port: IBM 4019 LaserPrinter on LPT1:

```

Attention: The contents of them member RDCFGxx is case sensitive!
Enter **CASE M** in the command line to switch to mixed case mode before editing.

Change the settings as shown below. Please keep track of the mixed case format.

```

IESRDENT CATALOG='VSESP.USER.CATALOG' ,           X
          CLUSTER='FFSTORES.DEMO.CLUSTER.TEAMxx' ,   X
          EXIT='IESREDIR' ,                          X
          OWNER=REDIRECTOR ,                         X
          IP='<ip-addr>' ,                            X
          HANDLER='com.ibm.vse.db2handler.DB2Handler' , X
          OPTIONS='dburl=jdbc:db2:<database>;        X
                   dbtable=<table>;                X
                   maptable=<maptab>;map=<mapname>;   X
                   dbuser=<userid>;dbpassword=<password>'

```

Replace **<ip-addr>** with the **IP address of your workstation**. Enter **"ipconfig"** in a Windows command prompt to find out which IP address has been assigned to your workstation.

```

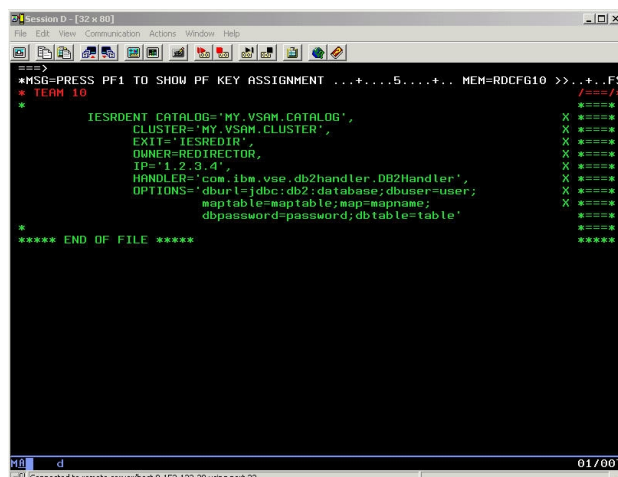
Use  <userid>=db2admin
     <password>=db2admin
     <database>=SAMPLE
     <table>=DB2ADMIN.FFSTORES
     <maptab>=DB2ADMIN.FFSTORES_MAP
     <mapname>=FFSTORES

```

One of the most important parameter is OWNER:

- **OWNER=VSAM** means the all READ requests will be done from the VSAM cluster, but INSERTs, UPDATEs and DELETEs will be done to VSAM and to the remote site. That means synchronization takes place.
- **OWNER=REDIRECTOR** means ALL requests are redirected to the remote site. No local VSAM processing is none anymore.

The HANDLER parameter specifies the Java class that should be invoked for this VSAM cluster by the VSAM Redirector Server on Windows.



```

Session 0 - [32 x 80]
File Edit View Communication Actions Window Help
====>
*MSG=PRESS PF1 TO SHOW PF KEY ASSIGNMENT ...+...5...+. MEM=RDCFG10 >> ..+. .FS
* TERM 10
*
      IESRDENT CATALOG='MY_VSAM_CATALOG' ,           X
            CLUSTER='MY_VSAM_CLUSTER' ,             X
            EXIT='IESREDIR' ,                       X
            OWNER=REDIRECTOR ,                     X
            IP='1.2.3.4' ,                          X
            HANDLER='com.ibm.vse.db2handler.DB2Handler' , X
            OPTIONS='dburl=jdbc:db2:database;dbuser=user; X
                     maptable=maptable;map=mapname;   X
                     dbpassword=password;dbtable=table'
*
**** END OF FILE ****

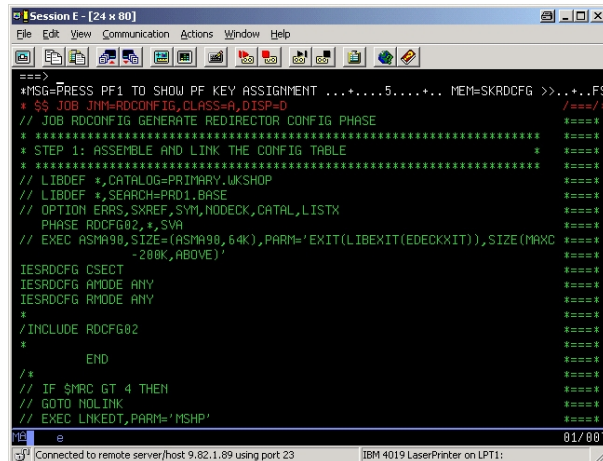
```

Press PF3 to exit the editor and save the changes.

In addition to the member RDCFGxx that you have just edited, there is a skeleton named **SKRDCFG** that you should use to compile your configuration.

For the workshop, we have modified this skeleton to not generate the system wide configuration table. However, you need to use the skeleton to compile your configuration and check for any compile errors. Once the generation was successful, we can pick up your configuration and include it into the system wide configuration we use for this workshop.

Enter: 1 <Enter> in front of SKRDCFG
 (if you want to edit it and look at it)
 Hit: PF3 (to save and exit)



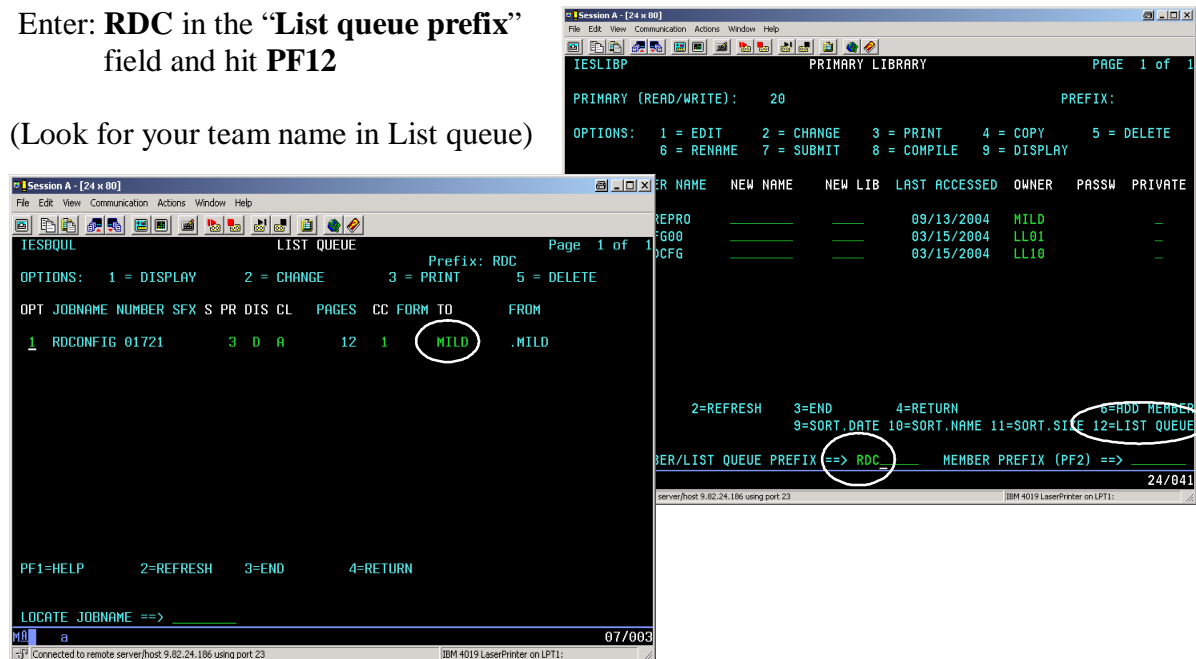
Submit **SKRDCFG** and check for RC=0000 (this is only to test for errors)

Enter: 7 <Enter> (in front of SKRDCFG)

The compile job is now submitted to VSE/POWER. Next, look into the POWER List queue to check for completion of the job and its return code:

Enter: **RDC** in the “List queue prefix” field and hit **PF12**

(Look for your team name in List queue)

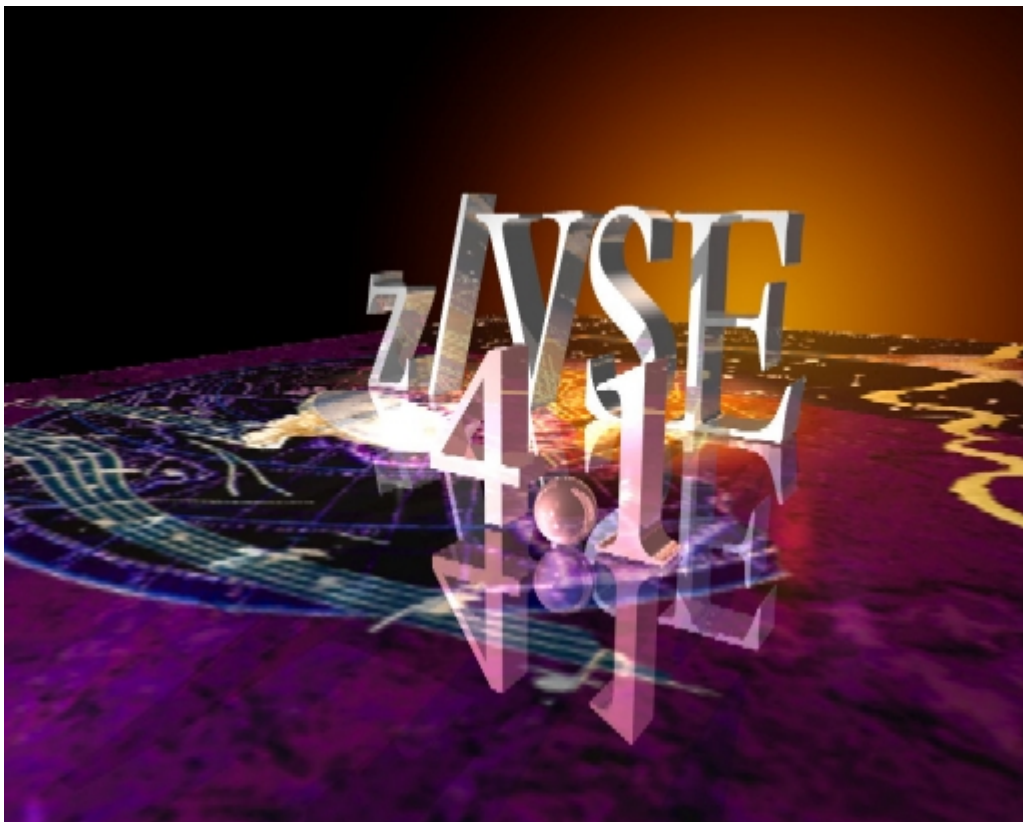


In the List queue dialog, enter 1 to display the entry and PF12 to go to bottom.

Look for **RC=0000** and tell the instructor to let him activate your definitions.

Congratulation

You have completed the workshop.



Appendix A: Setup Connector Server on VSE

The VSE Connector Server is the server part of the Java-based connector. It is the TCP/IP listener that sits there and waits for clients to connect. The server provides access to different kind of VSE resources.

The VSE Connector Server comes pre-installed as part of the VSE installation and does not need any special configuration on VSE.

Tailor startup job for VSE Connector Server

Per default, there is a startup job placed in the VSE reader queue called STARTVCS. If you have a need to modify that startup job, please copy the skeleton SKVCSSTJ from ICCF library 59 into your primary library and edit it.

Specify the correct TCP/IP system ID

Every instance of TCP/IP for VSE has its own system ID (default is 00) which is 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 the TCP/IP stack to use. This is specified in the VSE Connector server startup job as follows:

```
// OPTION SYSPARM='nn'   where nn is the system ID of TCP/IP
```

Start the VSE Connector Server

To start the VSE Connector Server, use the startup job (STARTVCS) that has been placed in the VSE reader.

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 Clients.

Appendix B: Download the VSE Connector packages

The VSE Connector Client package as well as the VSAM Redirector Server package comes as part of the VSE installation as a W-book. However, you can also download the latest version from the VSE homepage. Some Tools like VSAM Maptool can only be obtained from the VSE Homepage.

<http://www.ibm.com/servers/eserver/zseries/zvse/downloads/>

This page opened contains the VSE Connector Client, VSAM Redirector Server, VSAM Maptool as well as many other tools and components. For every component or tool, you see a little box similar to the following:

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

To download, click on: **Download now**

Download the file into a temporary directory (i.e c:\conntmp) and unzip it. For installation, please follow the Installation link as part of the box.


VSE Connector Client

Updated: 03/2005

- [Detailed Description](#)
- [Documentation](#)
- [Latest APARs and PTFs](#)
- [Frequently asked questions](#)
- ↓ [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

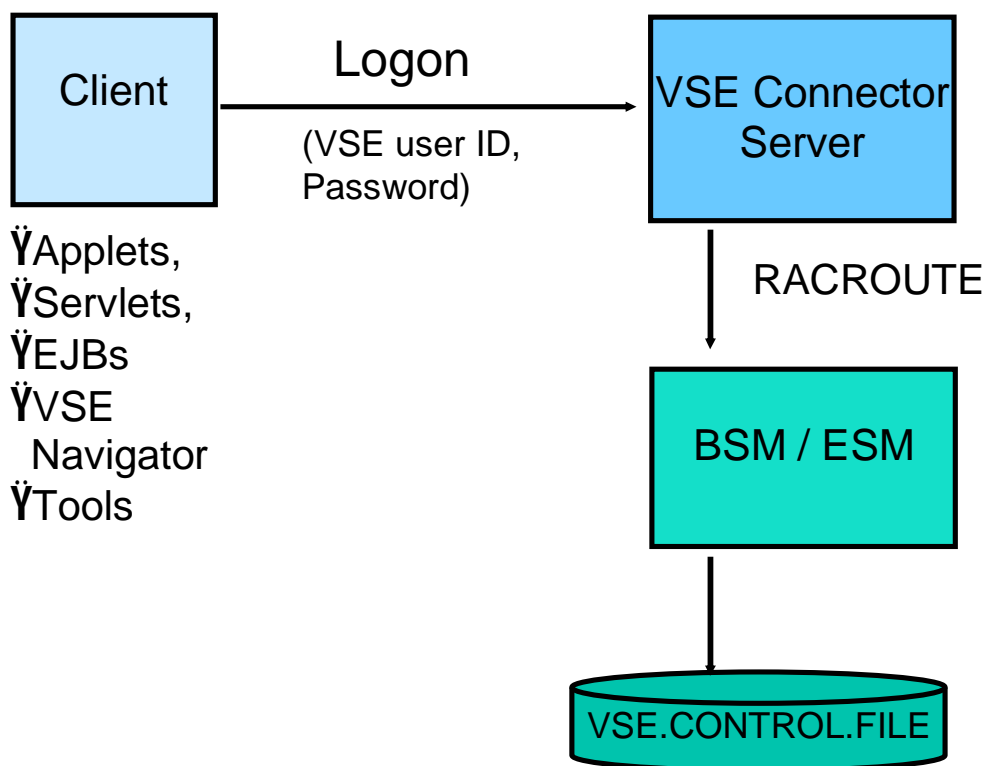
 [Download now](#) (zip, 7.2MB)

Some of the components are packaged using a graphical installer that shows some dialogs that guide you through the installation process. Other components come in a ZIP file that you simply need to extract into a new folder.

Appendix C: Connector Security

To access VSE resources from remote platforms, the VSE security rules apply. Before a client can access any resource on VSE, it has to sign on using a valid VSE user-id and password. Whenever the client tries to access a protected VSE resource, the permissions are checked using RACROUTE requests through a security manager (BSM or ESM). Using a Basic Security manager or an Extended Security Manager allows the global or more granular control of permissions for accessing VSE resources.

In addition, the Java-based Connector can work using secured (SSL) connections.



Appendix D: Create the DB2 tables and load VSAM data

Relational database tables have columns with specific characteristics. To be able to move VSAM data into a database table, we must know the internal structure and format of the VSAM records. Therefore we need a mapping containing all fields of a VSAM record including their position in the record, its length and data type.

The easiest way to create such a mapping is to use **VSAM Maptool** or **VSE Navigator** by importing an existing COBOL copybook. We have done that already in this workshop in chapter Chapter 4.)

The VSAM Redirector provides a tool called "Create DB2 Tables" that requires the mapping information to be supplied in an XML file. The mapping definitions can be exported from VSE Navigator or from VSAM Maptool into an XML file (i.e. ffstores.xml). The "Create DB2 Tables" tool reads the mapping and automatically creates database tables according to the mapping.

Create the tables in the DB2 database

To continue with the next steps:

- You must have installed a database management system (i.e. DB2).
- You must have a JDBC Driver for it (i.e. db2java.zip in <db2>\java\).
- You must be able to connect to the database (user-id/password)
- Have the VSAM mapping information in an .xml file

Make sure the JDBC driver is contained in the CLASSPATH of the system or the "Create DB2 Tables" tool. You may have to edit the **create.bat/create.cmd/create.sh** batch files as well as the **run.bat/run.cmd/run.sh** batch files and add the JDBC Driver to the CLASSPATH:

```
set CLASSPATH=.;xerces.jar;<db2>\java\db2java.zip;%CLASSPATH%
```

Also the **PATH** must be setup to contain the DB2 libraries. This is usually done automatically when installing DB2.

Start the "Create DB2 Tables" utility (**create.bat/create.sh** or START → All Programs → VSAM Redirector → Create DB2 Tables).

Enter following parameters when prompted by the tool:

- **XML Filename:** name of the full path XML File (i.e. C:/ffstores.xml)
- **DB url:** the JDBC URL for your database, i.e. jdbc:db2:sample
- **DB user:** a user id you use for accessing the database (i.e. workshop)
- **DB password:** db users password (i.e. workshop)
- **DB table name:** name of the data table, i.e. FFSTORES. This table will contain the VSAM data later on.

- **Map table name:** Enter the name of the table that contains the mapping information, (structure of VSAM record) i.e. FFSTORES_MAP. You can use the same map table for several clusters.
- **Map name:** Enter the name of the map, that will be the selection key in the 'Map table' i.e. FFSTORES. This name is independent from the map defined with the VSE Navigator.
- **DB system:** Enter 1 for DB/2, 2 for Oracle.
- The tool now imports the XML file and connects to the database.
- Next, the map info table is created (FFSTORES_MAP).

Enter **Yes** to continue.

- The data table is created (FFSTORES).

Enter **Yes** to continue.

- **Cluster Type:** Enter **1** (KSDS without AIX).
- **Primary key field:** Enter the name of the key field: **STOREID**.

Press enter to create the table and indexes.

Load data into the DB2 table

Before we can work with data in that database, we have to load/migrate once the whole data from VSAM into the database. To do this we define a 'twin' cluster, with the same characteristics as the one with the data, redirect this cluster and copy (REPRO) the original cluster into the redirected one.

In the workshop we use the already existing cluster FFSTORES.DEMO.CLUSTER to repro the data into the redirected cluster FFSTORES.DEMO.CLUSTER.TEAMxx.

Use the IDCAMS REPRO function to copy the contents of FFSTORES.DEMO.CLUSTER into the redirected FFSTORES.DEMO.CLUSTER.TEAMxx. You can see job FFREPRO in your primary library:

```
* $$ JOB JNM=REPRO,CLASS=A,DISP=L
// JOB REPRO COPY FILE
// DLBL COPYIN, 'FFSTORES.DEMO.CLUSTER',,VSAM,CAT=VSESPUC
// DLBL COPYOUT, 'FFSTORES.DEMO.CLUSTER.TEAMxx',,VSAM,CAT=VSESPUC
// EXEC IDCAMS,SIZE=AUTO
  REPRO INFILE (COPYIN) -
        OUTFILE (COPYOUT) -
        NOREUSE
/*
/&
* $$ EOJ
```

During the copy process, the VSAM Redirector Client will get control (at open time of FFSTORES.DEMO.CLUSTER.TEAMxx) and will connect to the VSAM Redirector Server and to the database. In case this fails, a VSAM open error message will be displayed.

To verify the data in the DB/2 table, issue the following SQL statement in a DB2 command window, or use the DB/2 Control Center:

```
SELECT * FROM FFSTORES
```

Additional information

- z/VSE Home Page
<http://www.ibm.com/servers/eserver/zseries/zvse/>
- e-business Connectors User's Guide SC33-6719
<http://www.ibm.com/servers/eserver/zseries/zvse/documentation/#comm>
- VSE Connectors: Components and Tools
<http://www.ibm.com/servers/eserver/zseries/zvse/downloads>
- VSE solutions
<http://www.ibm.com/servers/eserver/zseries/zvse/solutions>

IBM **Redbooks** International Technical Support Organization **and publications:**

- z/VSE Basics SG24-7436
- e-business Connectivity for VSE/ESE 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
- CICS Transaction Server for VSE SG24-5997
– CICS Web support
- WebSphere V5 for Linux on zSeries SG24-7042
Connectivity Handbook
- zJournal Articles about z/VSE and SOAP:
<http://www.zjournal.com/index.cfm?section=searchresults>

Contact:

You can reach us at the following e-Mail address:

zVSE@de.ibm.com