

IBM

IBM Systems & Technology Group

VSE access to remote relational databases

Linux
on
zSeries



z/VSE

z/VM

IBM zSeries and System z

Wilhelm Mild
IBM Germany
zvse@de.ibm.com

© 2006 IBM Corporation

IBM zSeries Systems and Technology Group

IBM

Trademarks

The following are trademarks of the International Business Machines Corporation in the United States and / or other counties.

<p>CICS*</p> <p>DB2*</p> <p>DB2 Connect</p> <p>DB2 Universal Database</p> <p>e-business logo*</p> <p>Enterprise Storage Server</p> <p>HiperSockets</p>	<p>IBM*</p> <p>IBM logo*</p> <p>IMS</p> <p>Intelligent Miner</p> <p>Multiprise*</p> <p>MQSeries*</p> <p>OS/390*</p> <p>S/390*</p> <p>SNAP/SHOT*</p>	<p>Virtual Image Facility</p> <p>VM/ESA*</p> <p>VSE/ESA</p> <p>z/VSE</p> <p>VisualAge*</p> <p>VTAM*</p> <p>WebSphere*</p> <p>xSeries</p> <p>z/Architecture</p> <p>z/VM</p> <p>zSeries</p> <p>System z</p>
--------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

* Registered trademarks of IBM Corporation

The following are trademarks or registered trademarks of other companies.

LINUX is a registered trademark of Linus Torvalds

Tivoli is a trademark of Tivoli Systems Inc.

Java and all Java-related trademarks and logos are trademarks of Sun Microsystems, Inc., in the United States and other countries

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



Microsoft, Windows and Windows NT are registered trademarks of Microsoft Corporation.

SET and Secure Electronic Transaction are trademarks owned by SET Secure Electronic Transaction LLC.

Intel is a registered trademark of Intel Corporation.

ACUCORP is a registered Trademark of ACUCORP Corporation

2

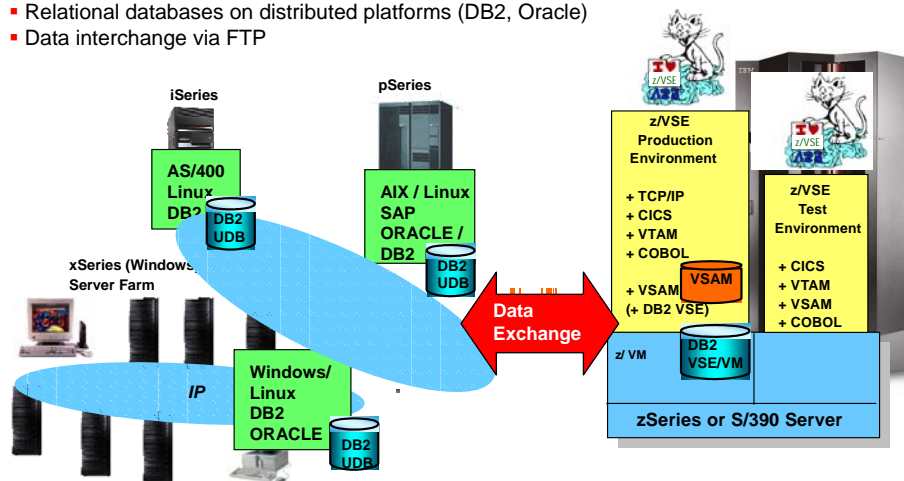



DB2 possible solution scenarios

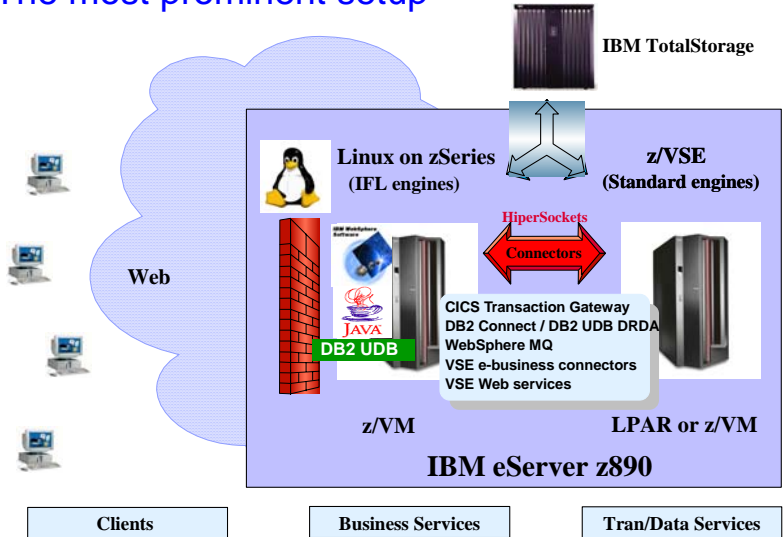
- **DB2 Scenarios with VSE**
 - DB2 Server and 'DB2 Client' to access DB2 UDB
- **VSE Applications access DB2 UDB**
 - SQL access to DB2 UDB on Linux on zSeries
- **VSE/VSAM Access to DB2 UDB**
 - VSAM applications, to access DB2 UDB

Typical VSE Customer Environment

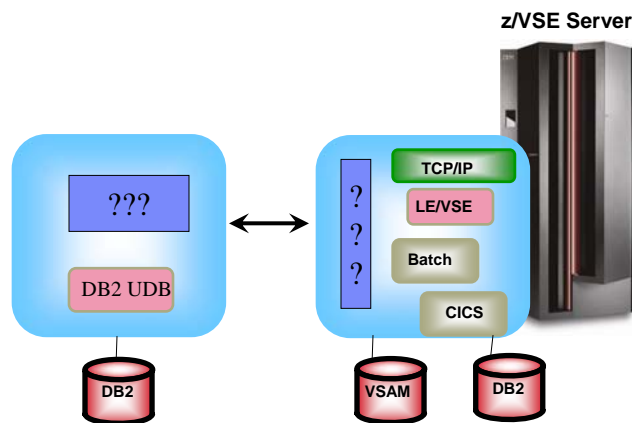
- Various different servers (zSeries, pSeries, iSeries, xSeries, and competitive)
- VSAM data on VSE (few DB2 environments)
- Relational databases on distributed platforms (DB2, Oracle)
- Data interchange via FTP



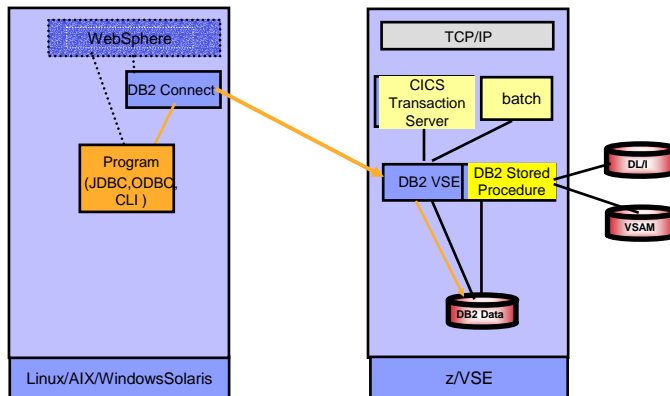
z/VSE with Linux on zSeries The most prominent setup



VSE applications and DB2 UDB on Linux on zSeries

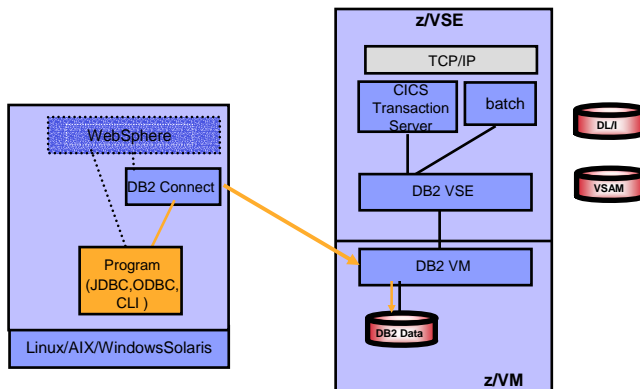


Relational Access to DB2 VSE



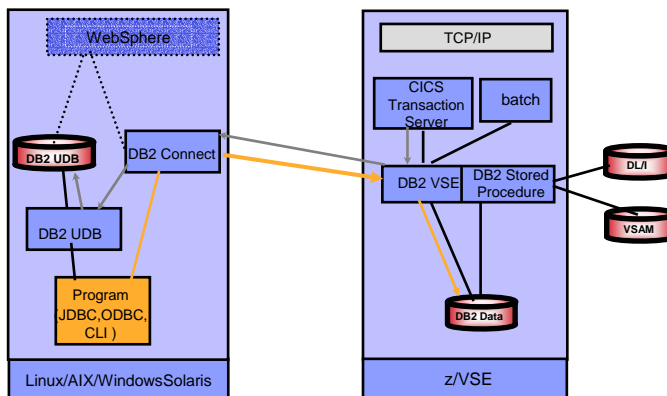
- ▶ Remote access of DB2 VSE via DB2 Connect
- ▶ Integration of non relational VSE data (i.e. VSAM, DLI) with remote DB2 logic via Stored Procedures

DB2 VSE/VM Guest sharing scenario, access to DB2 VM from Remote applications



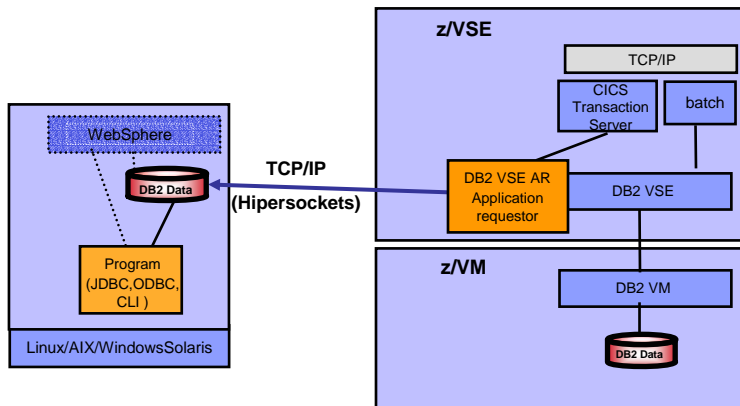
- ▶ VSE applications access DB2 VM via Guest sharing
 - ▶ DB2 server for VM/VSE must be installed on both platforms
 - ▶ DB2 Databases (data) on VM only
- ▶ Remote access of DB2 VM via DB2 Connect
- ▶ No Stored procedure access to VSE data possible

Integration of DB2 UDB with DB2 VSE in Remote applications



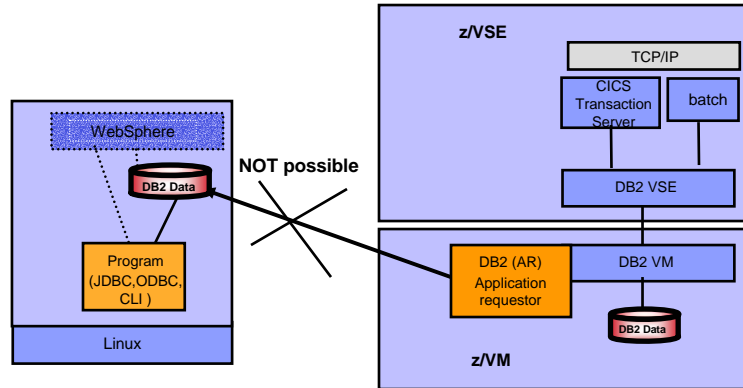
- ▶ Remote access of DB2 VSE via DB2 Connect
- ▶ Integration of non relational VSE data with DB2 logic via Stored Procedures
- ▶ Remote programs can access DB2 UDB and DB2 VSE at the same time.

DB2 VSE applications to access, remote DB2 UDB on Linux



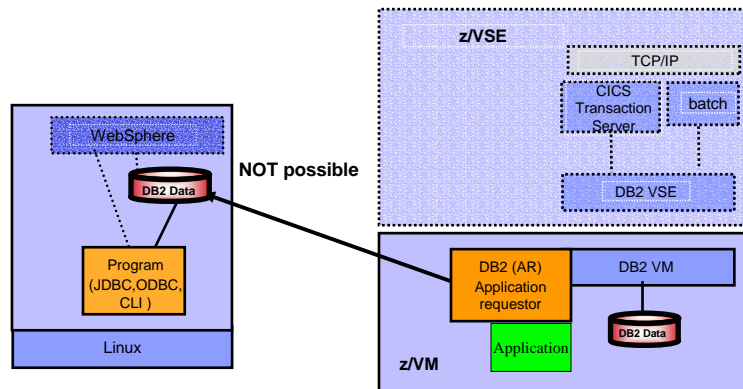
- ▶ VSE applications access remote DB2 UDB data via DB2 Application requestor in VSE
- ▶ VSE applications access DB2 VM via Guest sharing
 - ▶ DB2 must be installed on both platforms
 - ▶ DB2 Databases on VM only (no data on DB2 VSE)

DB2 VSE applications to access, remote DB2 UDB on Linux (NOT possible via DB2 AR in VM)



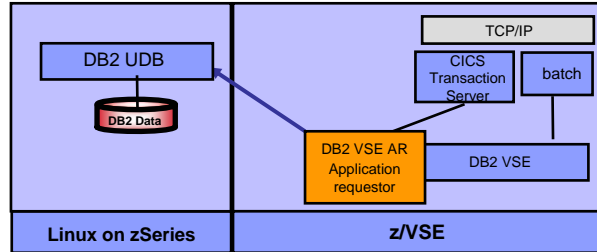
- ▶ VSE application access DB2 VM via Guest sharing
 - ▶ DB2 must be installed on both platforms
 - ▶ DB2 Databases on VM only
- ▶ From VSE applications NO remote access via VM Application requestor possible
 - ▶ DB2 VM Application Requestor (AR) can not be a gateway to DB2 UDB on Linux

DB2 VM applications to access, remote DB2 UDB on Linux



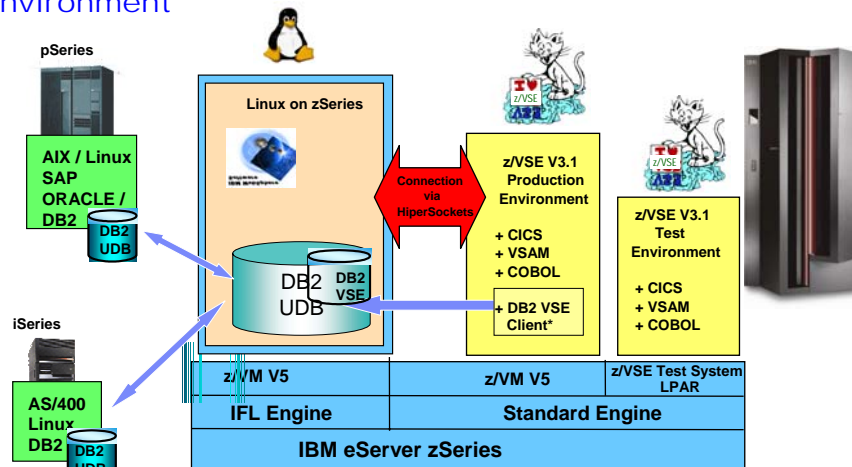
- ▶ VM application accesses DB2 UDB on Linux

DB2 VSE applications to access, remote DB2 UDB on Linux on zSeries - Special OFFERING



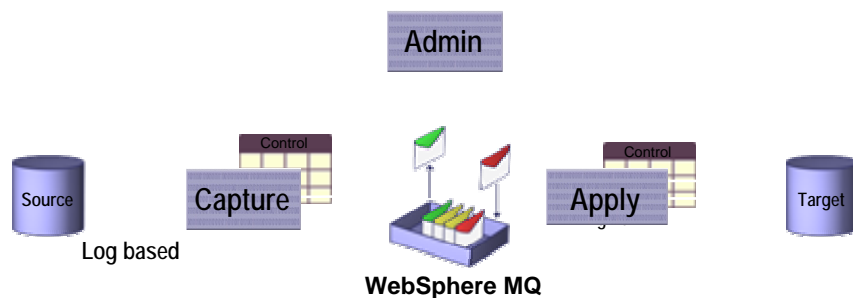
- ▶ Original Price Model: License for DB2 VM/VSE AND DB2 UDB for Linux
- ▶ PRPQ: P10154 (Ordering Nr: 5799-HAQ)
 - ▶ Reduced License for DB2 VSE Client only - if NO data on VSE
 - ▶ Full Price for DB2 UDB on Linux on zSeries
- ▶ Special Price for DB2 UDB for Linux on zSeries
- ▶ Note:
 - ▶ Both Products are needed because of the Programming interface and precompiler
 - ▶ On VSE the SQL language that can be used is the DB2 VSE SQL Language – because of precompiler

DB2 Scenarios – with DB2 UDB on Linux a modern environment



(*) DB2 VSE Client – the client functionality only, can be obtained with [PRPQ P10154](#)

Q Replication Architecture between DB2 UDBs



- Each message represents a transaction
- Highly parallel apply process
- Differentiated conflict detection and resolution
- Integrated infrastructure for replication and publishing
- Staged availability of heterogeneous support

DB2 VSE and DB2 UDB on Linux on zSeries

Why use DB2 UDB on Linux on zSeries with VSE Core applications

- Modern environment in DB2 UDB on Linux on zSeries
- Existence of lots of tools for:
 - database management
 - Optimization and Tuning
 - Data analysis (Warehouse, Mining, OLAP)
- ASCII environment – easy integration with distributed DB2 UDBs
- Consolidation of DB2 UDB databases from distributed platforms
- **Note: DB2 CONNECT is not needed on Linux on zSeries**

DB2 VSE and DB2 UDB on Linux on zSeries

Why use DB2 UDB on Linux on zSeries with VSE Core applications

- VSE applications access to DB2 UDB on Linux via HiperSockets
 - reliable network – no wires
 - fast network (memory copy speed)
 - transparent

- Core applications on VSE (CICS and batch):
 - can be used unchanged with considerations of EBCDIC – ASCII code pages (i.e. sorts with low values)
 - can show performance degradations if mass single row processing is done – these applications might need adaptations

- **Note: DB2 CONNECT is not needed on Linux on zSeries**

Environment and Database design

Configuration for CICS applications and remote DB2 UDB database

- VSE environment
 - configure DB2 VSE database directory
 - configure ARISDIRD (IP, port, DBname of remote database)
 - enable DRDA code (batch and online)
 - configure ARIS74LD (batch), ARIS745D (AR)
 - new transaction in CICS to bind packages (CBND) to remote AS (done during program preparation)

- zLinux environment
 - configure database manager on DB2 UDB zLinux
 - change some DBM parameters to allow implicit connect from within CICS
 - configure VSE batch and ISQL options (create remote packages)
 - ARIISQL for ISQL and ARIDSQL for Batch

- **Note: DB2 CONNECT is not needed on Linux on zSeries**

Environment and Database design

Configuration for CICS applications and remote DB2 UDB database

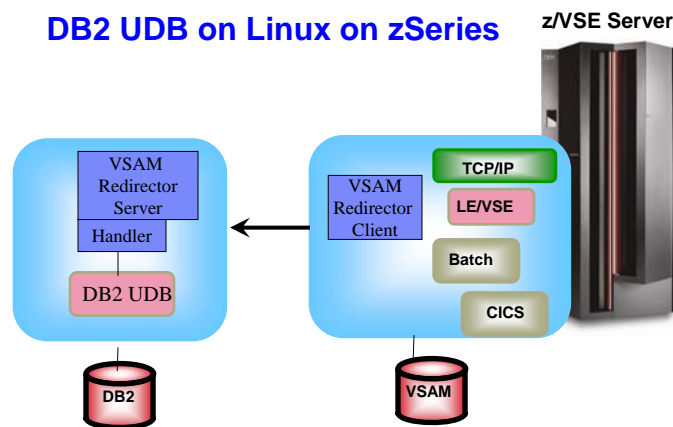
Application considerations:

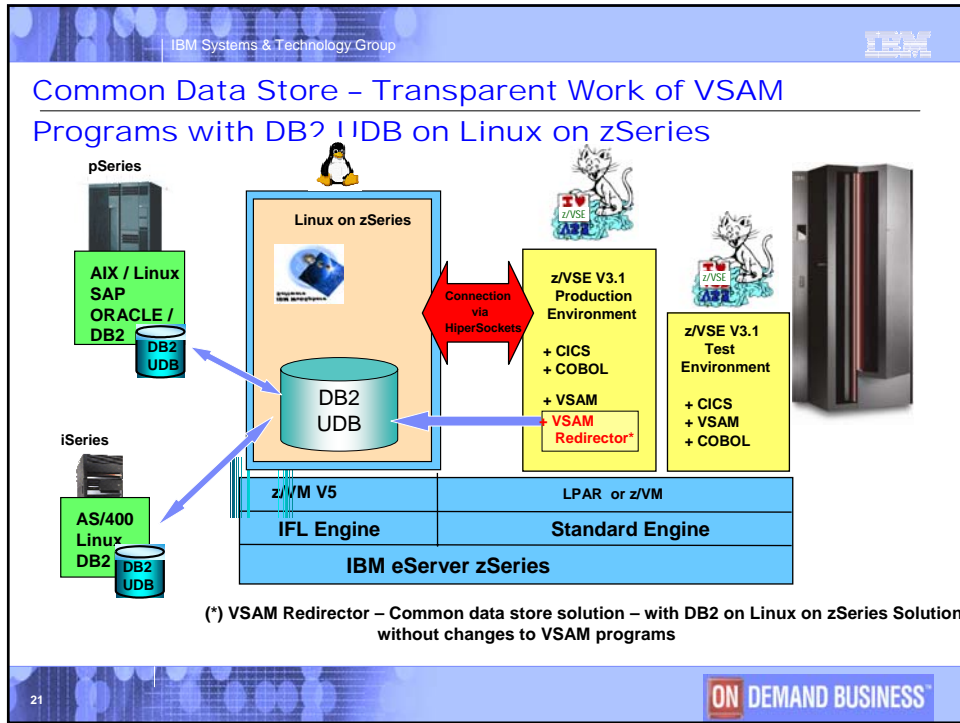
- migrate tables from DB2 VSE to DB2 UDB zLinux
 - UDB export/import options
 - use of federated DB2 UDB options and a cursor application

- existing CICS/DB2 VSE applications
 - no changes to the source code required (except Code page issues)
 - the SQL precompile creates new packages on the remote DB2 UDB)

- existing VSE batch DB2 VSE applications
 - no changes to source code required
 - adapt CONNECT statements to access remote DB2 UDB

VSE/VSAM applications access to DB2 UDB on Linux on zSeries



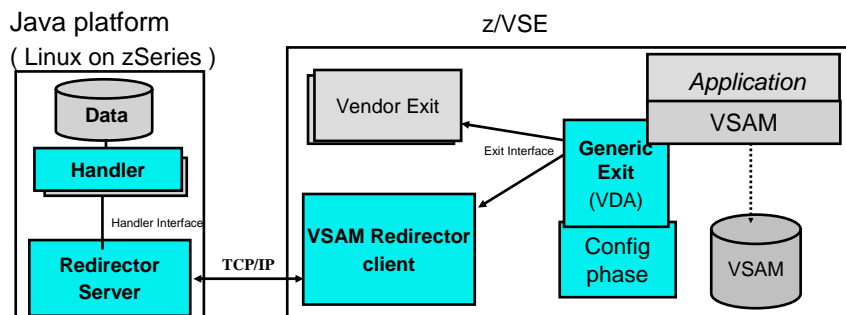


- IBM Systems & Technology Group
- ## VSE/VSAM Redirector
- **Benefits**
 - ▶ no changes to existing applications
 - ▶ access remote data from VSE applications
 - ▶ synchronization of VSAM data and remote data (database)
 - ▶ transparent for batch and CICS
 - ▶ offers new options for incremental
 - ▶ processing
 - ▶ FTP
 - ▶ journaling
- 22 **ON DEMAND BUSINESS**

VSAM Redirector types of operation

- (1) Real time VSAM to DB2 access
 - a) synchronization (two phase commit of VSAM and DB2)
- (2) VSE local data collection for VSAM
 - a) Capture Exit and Incremental FTP, processing
 - b) MQ Exit and MQ Series solutions

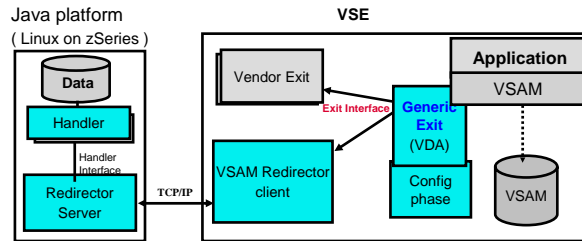
VSE/VSAM Redirector - functional view



- Redirector Components:
 - Generic Exit is based on VSAM Data Access Exit (VDA)
 - Config phase - redirection properties
 - Redirector client
 - Redirector server
 - Handler

VSE/VSAM Redirector - Components

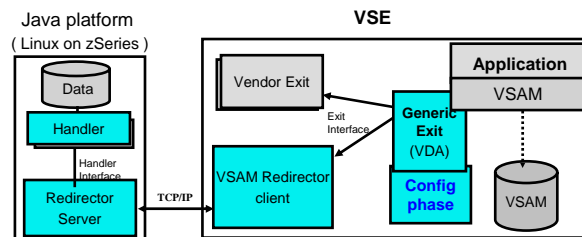
Generic Exit



- ▶ **Generic Exit**
 - ▶ based on VSAM Data Access Exit (VDA) IKQVEX01
 - ▶ VSAM requests will be intercepted (i.e. OPEN,CLOSE,GET,PUT,POINT...)
- ▶ The exit is called twice for each VSAM request.
 - ▶ before request execution
 - ▶ after request finished
- ▶ RC=0 from exit, normal VSAM processing continues
- ▶ RC=4 from exit, skip all physical VSAM access, return to caller with RC=0
- ▶ Config phase is used for decisions of further processing
- ▶ Open (documented) **Exit interface**
- ▶ To use Generic Exit, no changes to VSAM programs are required.

VSE/VSAM Redirector - Components

Configuration phase



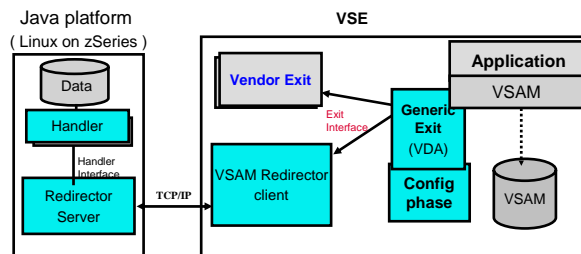
- ▶ **Config phase**
 - ▶ contains parameters for the processing decision
 - ▶ cluster and catalog name
 - ▶ whether VSAM file should or not be redirected
 - ▶ destination specification like IP address, PORT, handler,
 - ▶ option string containing i.e. database name, userid, password
 - ▶ skeleton in ICCF Library 59 (SKRD CFG) is provided
 - ▶ allows integration of existing VDA Exit phases

Information in Configuration phase

- ▶ **CATALOG**= VSAM catalog name for the file to be redirected. (wildcard allowed) (CLUSTER=* - required)
NOTE: If the master catalog is redirected, you might not be able to startup your VSE/ESA system!
- ▶ **CLUSTER**= VSAM cluster name.(wildcard allowed).
- ▶ **EXIT**= Name of the exit phase or 'IESREDIR'.
 - ▶ **EXIT=phasename** - the phase with this name will be invoked (integration of VDA exit)
 - ▶ **EXIT='IESREDIR'** - the redirector client will be invoked with additional parameters
- ▶ **OWNER**= Name of the primary VSAM data owner (VSAM or REDIRECTOR)
 - ▶ **OWNER = REDIRECTOR** - No VSAM access is done for this VSAM cluster, all requests will be redirected
 - ▶ **OWNER = VSAM**: Dual processing occurs (both VSAM processing and redirecting requests are done)
- ▶ Options *for both OWNER modes*:
- ▶ **IP**= IP address VSAM Redirector Server to handle the requests
- ▶ **PORT**= (optional, default is 2387) Port number the VSAM Redirector Server
Standard port is 2387 is assigned by the Internet Assigned Numbers Authority (IANA).
- ▶ **HANDLER**= Name of the Java class to be started, which represents the request handler
- ▶ **OPTIONS**= A string with arbitrary, options.
- ▶ Options *for OWNER=VSAM*:
- ▶ **IGNOREERROR**= NO|YES (opt, default NO) If set to YES, no error is set if redirector server is unreachable.
- ▶ **PUTREQONLY**= NO|YES (opt, default is NO). If set to YES , only INSERT, UPDATE and DELETE requests are redirected. (useful for synchronisation purposes, excluding the requests POINT and GET).

VSE/VSAM Redirector - Components

Vendor Exit

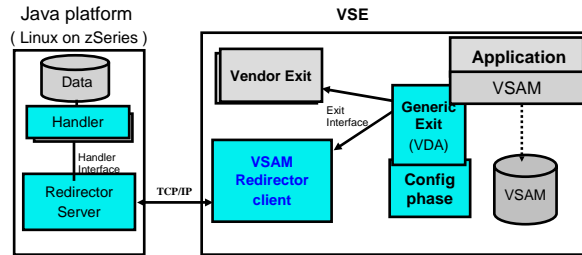


- ▶ **Vendor Exit**
 - ▶ user (or vendor) written phase for data collection/transformation
 - ▶ has to comply with the documented **Exit Interface**

Note: No chaining of Vendor Exit with VSAM Redirector client supported

VSE/VSAM Redirector - Components

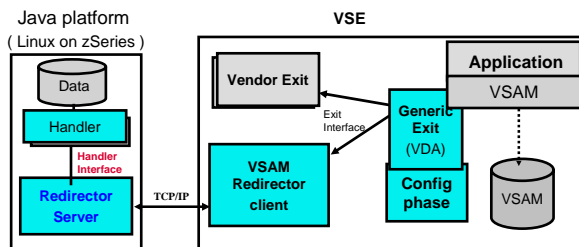
VSAM Redirector client



- ▶ VSAM Redirector client
 - ▶ running on VSE in user's program partition
 - ▶ component implemented as reentrant SVA phase
 - ▶ responsible for TCP/IP connection/session handling with Redirector server

VSE/VSAM Redirector - Components

Redirector server



- ▶ VSAM Redirector server
 - ▶ implementation: 100% Java
 - ▶ command line interface to interact with server (start/stop/status)
 - ▶ Open **Handler Interface**
 - ▶ responsible for TCP/IP connection with Redirector client
 - ▶ responsible for data translations (EBCDIC - specified Code Page)
 - ▶ configured via `VSAMRedirectorServer.properties` file
 - ▶ calls the handler for data processing (OPEN,...,CLOSE method)
 - ▶ generates (translates) error messages for VSAM

VSE/VSAM Redirector - Components

Redirector server (configuration)

- ▶ `VSAMRedirectorServer.properties` file used to configure Redirector server
 - ▶ `messages` = on|off
 - ▶ on: print server messages
 - ▶ off: quiet mode, messages will not be shown
 - ▶ `listenport` = portnumber
 - ▶ TCP/IP portnumber on which the server listens
 - ▶ `maxconnections` = number
 - ▶ number of maximum parallel connections allowed, from clients
 - ▶ `codepagetranslator` = `com.ibm.vse.server.DefaultTranslator`
 - ▶ the codepage translator class to be used to convert Strings from EBCDIC into ASCII and back (`get` method from `VSAMRequestInfo.class`)
 - ▶ `tracelevel` = number (currently 0 - 2)

Note: Individual translators can be written and implemented via VSAM Redirector properties file (`codepagetranslator`).

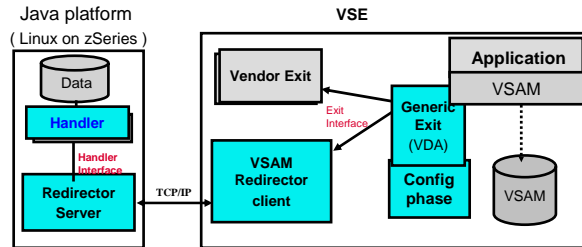
VSE/VSAM Redirector - Components

Redirector server (translator)

- ▶ Redirector server has 'data translation' functions implemented in the `VSAMRequestInfo.class`
 - ▶ `VSAMRequestInfo.class` contains methods to get and set data areas in the communication area. 'offset' plus 'length' of the record area and these methods are used to transform the data into:
 - ▶ `String`
 - ▶ `Number (Signed/Unsigned)`
 - ▶ `Packed Decimal`
 - ▶ `Binary bytes`
- These methods (see Javadoc) are called from `handler` if necessary.

VSE/VSAM Redirector - Components

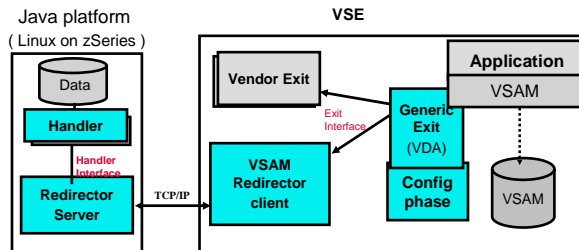
Handler



- ▶ **Handler**
 - ▶ file system specific
 - ▶ must reside in the directory of Redirector server
 - ▶ sample implementations: 100% Java
 - ▶ multiple handlers per system possible
 - ▶ **Handler interface** allows individual handler for proprietary/standard file systems and platforms
 - ▶ VSE/ESA V2.6/2.7 contains a sample implementation for a relational handler (DB2) and an HTML handler

VSE/VSAM Redirector

Working with remote data



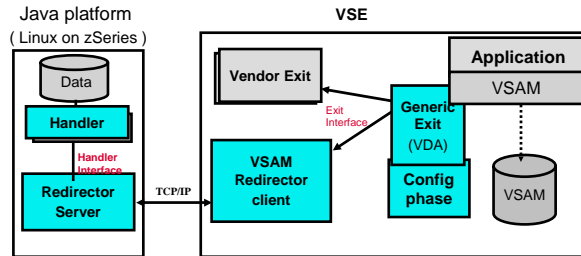
Working with data residing on another platform OWNER=REDIRECTOR
 No VSAM access any more (after OPEN).
 All requests are redirected to VSAM Redirector client, which connects to the VSAM Redirector server. VSAM Redirector server performs the request via the proper handler.

Note:

1. The redirected VSAM file must exist and have at least one dummy record.
2. You cannot chain exits if any one of these exits has OWNER set to REDIRECTOR.

VSE/VSAM Redirector

Synchronisation

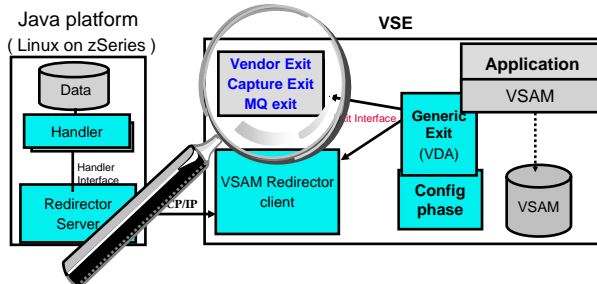


Synchronizing existing VSAM data with a remote file system (OWNER=VSAM)
 The programs perform a VSAM access and redirected access.
 Each VSAM request causes two invocations of VSAM Redirector Client, giving the chance to commit or rollback depending on VSAM request completion.

Note:
 PUTREQONLY=YES allows to make changes only simultaneously on 2 sides, READs are done from VSAM.

VSAM Data collection / transformation / journaling on VSE

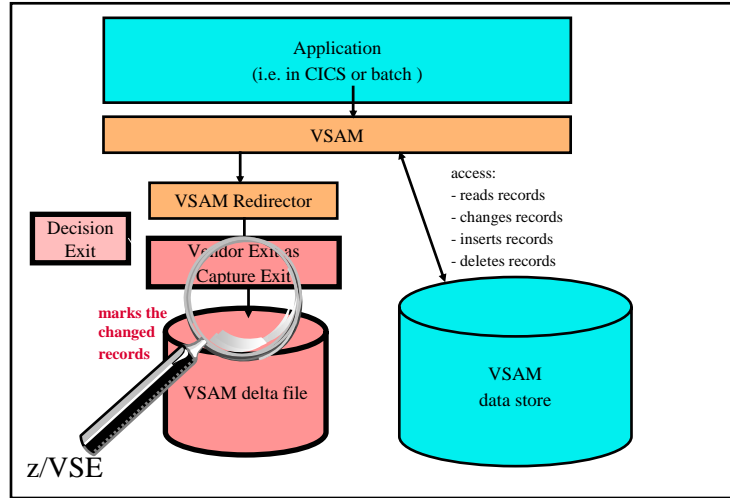
Vendor Exit



- ▶ Vendor Exit
 - ▶ user (vendor) written phase for data collection/transformation
 - ▶ has to comply with the documented Exit Interface

Note: No chaining of Vendor Exit with VSAM Redirector client supported

Architectural View



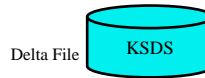
Journaling



Rekord 1	inserted
Rekord 2	inserted
Rekord 3	inserted
Rekord 2	updated
Rekord 1	deleted
Rekord 3	updated
Rekord 4	inserted
Rekord 1	inserted
Rekord 2	updated
Rekord 4	updated
Rekord 4	deleted

or

cumulative

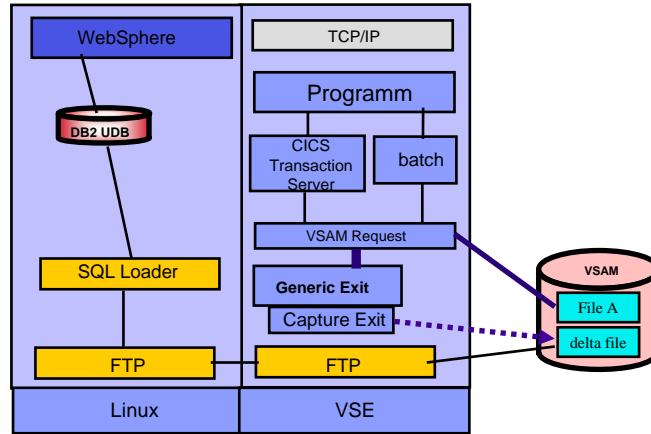


key

Rekord 1	inserted
Rekord 2	updated
Rekord 3	updated
Rekord 4	deleted

☞ The last version only of a changed VSAM record is stored into the delta file

Reduce network traffic, save time

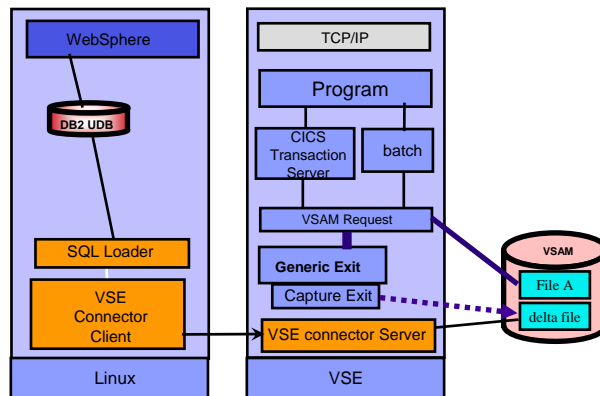


- ▶ Collect the changed records in a separate VSAM file
 - ▶ Possibility of cleansing
- ▶ FTP – as before, with a much smaller file

Capture Exit and Java-Based Connector

Reduce network traffic, save time

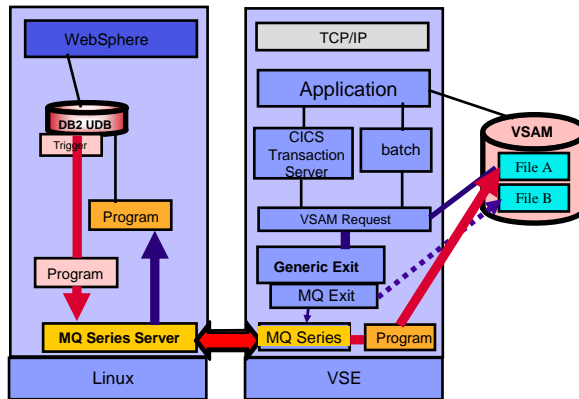
- ▶ Collect the changed records in a separate VSAM file
 - ▶ Possibility of cleansing
- ▶ Connector Client reads the delta file and inserts them into a database
- ▶ Transparent Journaling of data changes if wanted



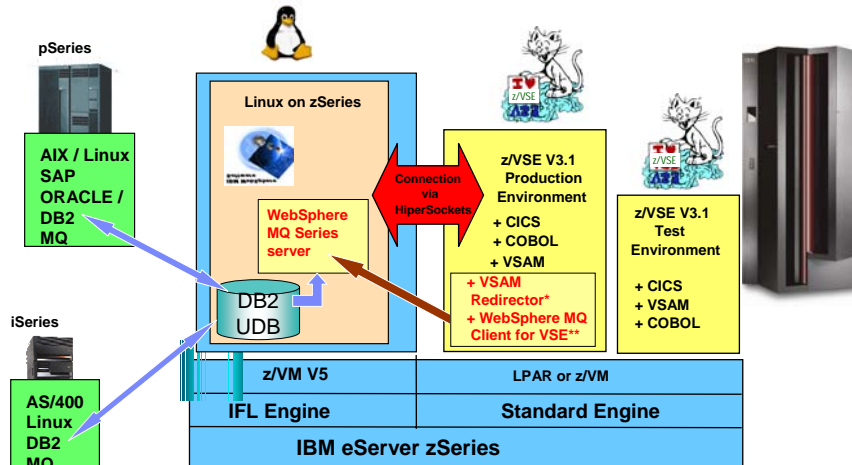
MQ Exit

Integration of VSE Application with MQ Series

► enablement for MQ Series w/o changing existing applications.



WebSphere MQ Series Solutions with z/VSE



(*) VSAM Redirector + Redirector MQ Exit allows MQ Solutions without changes to VSAM programs
 (**) WebSphere MQ Client for VSE is brand-new and free of charge

More information about DB2 UDB and DB2/VSE

- **Summary of DB2 Planning and Customization Tasks (VSE)**
http://publibz.boulder.ibm.com/cgi-bin/bookmgr_OS390/BOOKS/iespie41/10.4.5
- **Enabling the DB2 Server for VSE**
http://publibz.boulder.ibm.com/cgi-bin/bookmgr_OS390/BOOKS/iespie41/10.4.4
- **Customizing Tasks for DB2 Server for VSE (DB2-Based Connector)**
http://publibz.boulder.ibm.com/cgi-bin/bookmgr_OS390/BOOKS/IESWUE41/HDRINDB2BC
- **DB2 - Resolve Frequent Problems**
<http://publib.boulder.ibm.com/infocenter/db2help/index.jsp?topic=/com.ibm.db2.udb.doc/conn/c0005607.htm>
- **DB2 Universal Database (UDB)**
<http://www.ibm.com/software/data/db2/udb/>
- **Moving Data from DB2/VSE&VM to DB2 UDB**
<http://www-306.ibm.com/software/data/db2/vse-vm/support.html>

Additional Information

- z/VSE Home Page
<http://www.ibm.com/servers/eserver/zseries/zvse/>
- z/VSE Solutions and Utilities
<http://www-1.ibm.com/servers/eserver/zseries/zvse/solutions/>



- e-business Solutions for VSE/ESA SG24-5662
- e-business Connectivity for VSE/ESA SG24-5950
- CICS Transaction Server for VSE/ESA
 CICS Web Support SG24-5997-00
- *Handbook For WebSphere 5*
 (*Connectors to z/OS and VSE*) SG24-7042

We appreciate your comments on z/VSE: zvse@de.ibm.com