LE Implementation and Configuration (a user's perspective)

Karl De Vore – WAVV 2007

kdevore@co.lake.il.us

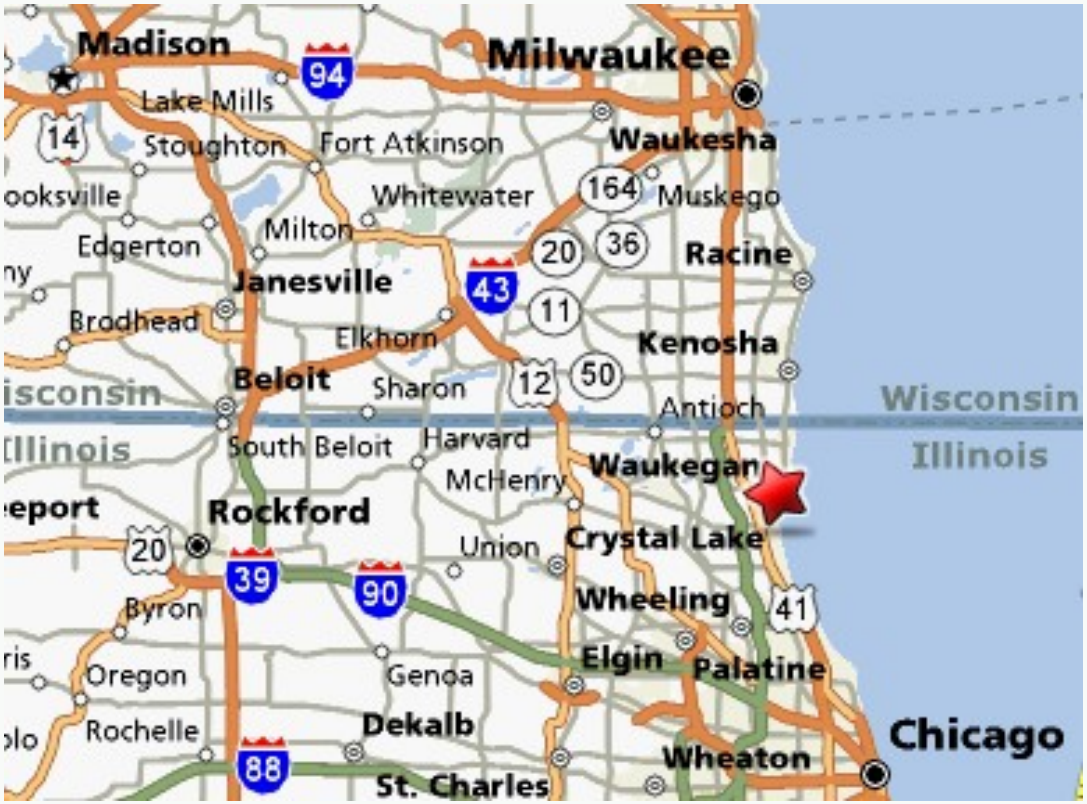
Agenda

- Lake County
- Lake County Environment
- LE/VSE General Requirements
- Compatibility
- Installation of LE/VSE
- Installation of COBOL/VSE
- Applying Maintenance
- Pre-customization Requirements
- General Customization Rules
- To Fix or not to Fix...
- Customization of LE/VSE Run-time Environments
- Selected LE/VSE run-time options
- New neat stuff with LE/VSE
- Customization of COBOL/VSE Compiler Options
- CICS, DL1 and DB2 Dependencies
- Suggestions
- Recommended reading and addition reference resources

Lake County

Demographics (2000 U.S. Census)

- Population 644,356
- Caucasian 80.1%
- African American 6.9%
- Asian 3.9%
- Other 7.0%
- Two or more races 2%
- Hispanic/Latino (may be of any race) 14.4%
- Median household income:
- Lake County $66,973
- Chicago $38,625
- Illinois $46,590
- U.S. $41,994



Lake County Environment

Hardware: z/890 – 220 SHARK Array

Software: z/VM 5.1 z/LINUX

z/VSE 3.1.1 +

z/VSE 3.1.1 Guests:

Prod1 VSE: 1 CICS/TS partition - DB2 Guest sharing and VSAM

Prod2 VSE: 2 CICS/VSE 2.3 - DL/1 databases

Test1 VSE: 3 CICS/TS partitions - 2 DB2 Guest sharing

1 UDB/DB2 POF

Lake County Environment

Some of the things that make us different:

NO IUI on any of the systems

NO VSE libraries in VSAM SPACE

NO DOSRES or SYSWK1

DB2 Guesting since 1993

Flavors of Cobol over the years:

DOS/VS COBOL 1.3.1 - ????

VS COBOL II – 1.3.2 – (24 bit) - VSE/ESA 1.2.x – 1991

VS COBOL II – 1.4.0 – (31 bit) - VSE/ESA 1.3.x – 1993

COBOL/VSE 1.1.0 and LE/VSE 1.4.1 – VSE/ESA 2.3.x – 1998

COBOL/VSE 1.1.1 and LE/VSE 1.4.4 – z/VSE 3.1.1 - 2006

LE/VSE General Information

Conforming Languages:	C	Rel.1
	COBOL/VSE	Rel.1
	PL/1	Rel.1
Non-comforming:	DOS/VS COBOL	
	VS COBOL II	
	DOS PL/1	
	DOS/VS RPGII	

Compatibility

Many DOS/VS Cobol and VS COBOL II programs will run without recompiling in compatibility mode (LE/VSE Run-time Migration Guide)

In compatibility mode:

No use of 4-digit dates

No exploitation of LE/VSE functionality

No Interlanguage communication capabilities

LE/VSE Compatibility for Object and Phase:

“A phase will run in compatibility mode with any LE/VSE that is equivalent to or higher than the level used to link-edit them” (your mileage may vary dependent on where you are coming from)

Installation of LE/VSE

With z/VSE:

LE/VSE now installs as part of the base during z/VSE installation in PRD2.SCEEBASE

LE/VSE Installation verification

The doc states that a 2 Meg partition is adequate

Consider using a 4 Meg or larger partition, but keep it below the line

Don’t load anything into the SVA at this point

STOP after the LE/VSE verification

Installation of COBOL/VSE

Decide on your installation library

Do ***not*** put COBOL/VSE into PRD2.PROD if you have or are planning to install DOS/COBOL - install it elsewhere!

COBOL/VSE Installation verification

The doc states that a 2 Meg partition is adequate

Consider using a 4 Meg or larger partition, but keep it below the line

Again, don’t load anything into the SVA at this point

STOP with the completion of verification

Applying Maintenance

Review the PSP buckets for LE/VSE and COBOL/VSE

Search → Servicelink PSP

"upgrade" name for VSE is zVSE311

"subset" name for LE/VSE = IBMLANG/81K

COBOL/VSE = IBMCOBOL/18M

Read all INFO APARS

Applying Maintenance

Backup your entire system

Types of backups I like:

FASTCOPIES of DOSRES - SYSWK1

STANDALONE backup of IJSYSRS

BACKUP of PRD1

BACKUP of PRD2

BACKUP of IJSYSHF (intentionally separate)

Apply maintenance (buckets)

Re-run your verification programs AFTER applying maintenance

Pre-Customization Requirements

Read these manuals:

LE/VSE Run-Time Migration Guide (SC33-6687-00)

LE/VSE Customization Guide (SC33-6682-06)

LE/VSE Programming Guide (SC33-6684-04)

In addition, you'll want to also look at these manuals:

COBOL/VSE Migration Guide (GC26-8070-00)

COBOL/VSE Programming Guide (SC26-8072-02)

COBOL/VSE Language Ref. (SC26-8073-02)

COBOL/VSE and VisualAge COBOL MLE for VSE Installation and customization Guide (SC26-8071-01)

General Customization Rules

To run LE/VSE or COBOL/VSE successfully

Rule #1	Review the “Summary of changes” in the LE/VSE Customization Guide
Rule #2	Take an inventory of applications and vendor products
Rule #3	Everybody should do their homework
Rule #4	Systems Programmers and Application Development should “willingly indulge” in productive conversation
Rule #5	If it ain’t broke - don’t fix it!

Customization considerations

Custom LIOCS modules (ie: old programs that have “reserved clause” that were written when 1403 printers were in vogue)

Custom User Exits

SVA Utilization and customization

Shipped loadlists:	$SVACEE	LE/BASE
	$SVAEDCM	LE/VSE recommended “C” routines
	$SVAIGZM	Recommended COBOL routines
	$SVAIGZ	Eligible COBOL Run-time

There are additional “optional” loadlists for the languages above, including PL/1

Customization considerations

CICS/TS SVA=YES

Is coded in the SIT for CICS/TS

Required RDO entries are predefined

Loads both $SVACEE and $SVAEDCM

COBPACKS and their purpose in life:

If a Cobol routine is not loaded into a COBPACK, then it gets loaded into the partition at runtime. The intent is to share routines across the system.

Years ago this made a big difference, but with today's machine speeds and real storage support, it becomes a lesser concern.

Customizing considerations

Two types of COBPACKS: GENERAL and ENVIRONMENT specific

GENERAL COBPACK

IGZCPAC (IGZWEPAC.Z – PRD2.SCEEBASE)

ENVIRONMENT COBPACKS

IGZCPCO (IGZWEPCO.Z – PRD2.SCEEBASE – BATCH)

IGZCPCC (IGZWEPCC.Z – PRD2.SCEEBASE - CICS)

Customization considerations

Member inclusion:

If you customized a COBPACK, ALL included routines need to be either 31 or 24 bit. If you inadvertently include a 24 bit routine in a 31 bit COBPACK, it will reside in 24 bit storage (SVA). Make sure you link edit it with AMODE=31, RMODE=ANY

A PARTIALLY loaded COBPACK will require less virtual storage but will be slower because more routines have to be loaded into the partition

A FULLY loaded COBPACK will run faster as no routines are loaded and it is reentrant, but more virtual storage will be used.

Appendix D of the LE/VSE Customization Guide documents the COBPACKS

To Fix or not to Fix...

That is one of the questions you need to ask before you start changing LE run-time or compiler options

In LE/VSE:

Fix is denoted by (NONOVR) following the option

DEBUG=((OFF),NONOVR) verses DEBUG=((OFF),OVR)

In COBOL/VSE:

Fix is denoted by a “*” in front of the option

ADV=*NO verses ADV=NO

Customization of LE/VSE run-time environments

As of VSE/ESA 2.5.x

Only 1 library (PRD2.SCEEBASE) is used now and specific configuration members are used to build the runtime options

VSE/ESA 2.6.x and thereafter…

There is an option that can be set in each assembly for the Online and Batch run-times jobstreams, that will allow automatic inclusion of those respective phases in the SVA.

All run-time source members (.a books) use the CEEXOPT macro

Coding rules for CEEXOPT macro are outlined in Chapter 2 of the LE/VSE Customization Guide (pages 18 through 25)

Customization of LE/VSE run-time environments

Effected source books and associated JCL:

CEEDOPT.A – BATCH (code all options)	CEEWDOPT.Z
CEECOPT.A – CICS (code all options)	CEEWCOPT.Z
CEEUOPT.A - Application specific	CEEWUOPT.Z

Option Search (highest to lowest):

Options found in CEEDOPT.A or CEECOPT.A

An Assembler user exit

Options found on the EXEC card

Options in CEEUOPT

Customization of LE/VSE run-time environments

Gotcha's:

*When ever you install a new release of LE/VSE or apply maintenance, **ALWAYS** pull down the new ".A" members, recode your settings and reassemble*

Beware of Case Sensitivity

Options and suboptions must be uppercased. Only suboptions that are strings can be specified in mixed-case or lowercase. For instance:

Okay: MSGFILE=(SYSLST) or MSGFILE(syslst)

Not okay: TRACE=((off,4K,DUMP,LE=0),OVR)

Customization of LE/VSE run-time environments

Gotcha's:

CICS

IF you were running an earlier version of LE/VSE previously

AND you are continuing to run CICS/VSE 2.3

AND you are using MACRO entries

THEN There are additional LE/VSE entries that you will have to add to your MACRO tables

With LE/VSE 1.4.4 +, CICS definitions are only shipped via the CSD. The following ".z" books contain these entries:

CEECCSD.Z	LE/VSE BASE (Mandatory)
IGZCCSD.Z	COBOL (optional)
IBMCCSD.Z	PL/1 (optional)
EDCCCSD.Z	C (Mandatory)
EDCUCSD.Z	C/VSE Code Converter (optional)

Customization of LE/VSE run-time environments

And now…

for something completely different…

There is a quick reference guide beginning on page 62 of the LE/VSE Customization Guide that outlines the LE/VSE run-time options. The nitty-gritty begins on page 70.

Not all LE/VSE run-time options are discussed in this presentation

The LE/VSE options discussed in the following foils will have underlines under the suboption(s), if that is the default as shipped by IBM

ABTERMENC=((ABEND),OVR)

Requires TRAP(ON) or TRAP(ON,MAX)

Should be set to ABTERMENC((ABEND)),OVR) for DB2 and DL/1

(ABEND = activates BACKOUT, RETCODE does nothing)

ALL31=((ON),OVR) - CICS

ALL31=((OFF),OVR) - BATCH

Turn this option on (for CICS) if you link with AMODE=31 as there are some performance improvements. You should not have any problems as long as any programs that there XCTL, LINK or CALLED, are also linked as 31 bit

ANYHEAP and all ‘dem HEAPS

All the heap commands define increments of storage

Take the defaults and leave ‘em alone unless there is a reason to mess with them. You’ll know…

CBLOPTS=((ON),OVR)

Specifies how parameters are acknowledged on the // EXEC card

// EXEC PGM=prog-name,PARM='*program-arguments*,/run-time-options'

CBLOPTS=((OFF),OVR)

// EXEC PGM=prog-name,PARM='run-time-options/*program-arguments*'

CHECK=((OFF),OVR)

Works with the SSRANGE COBOL/VSE Compiler option

Consider CHECK ON for Test and OFF for Production

Lake County runs with CHECK=((ON,OVR) in production

DEPTHCONDLMT=((10),OVR)

Might have to increase if you have lots of embedded "called programs" in a LE/VSE enclave

MSGfile=((SYSLST),OVR) - Batch

MSGfile=((CESE),OVR) - CICS

Where the output goes from RPTOPTS and RPTSTG

TEST/NOTEST=((ALL,*,PROMPT, ‘ ‘,),OVR)

Pertains specifically to the DEBUG tool

Recommend NOTEST for production - ***not*** a performance option

RETZERO=((OFF),OVR)

Pertains only to Batch and Forces “Return code 0” –

RPTOPTS=((OFF),OVR)

ON generates a report of “run-time options” in effect

Watch out if you turn this on for CICS

Recommend OFF for both Batch and CICS

RPTSTG=((OFF),OVR)

ON generates a report on “storage” utilization during a LE enclave

Watch out if you turn this on for CICS

Recommend OFF for both Batch and CICS

Example:

// EXEC, SIZE=128K,PARM=‘/RPTOPTS(ON),RPTSTG(ON)’

RTEREUS=((OFF),OVR)

Leave it OFF - else change all STOP RUNS to GOBACKS

STORAGE=((00,NONE,NONE,32K),OVR) - BATCH

STORAGE=((00,00,00,0K),OVR) – CICS*

Watch Out! - Unless you can guarantee that all the fields in Working Storage are initialized –

Second and third suboptions are the “fill values” used when storage is allocated or freed

The first suboption is “heap_alloc_value” and controls whether the fields defined in the Working Storage section of a COBOL program are initialize the system.

*Lake County has run with STORAGE=((00,00,00,0K) for years (The Ain’t broke, don’t fix it rule…)

Performance Recommendation, run with the defaults:

Batch: STORAGE=((00,NONE,NONE,32K),OVR)

CICS: STORAGE=((00,NONE,NONE,0K),OVR)

TERMTHDACT **Sets the level of information that is produced when LE/VSE percolates a condition of severity 2 or greater**

TERMTHDACT=((TRACE,LSTQ,,0),OVR) **- CICS**

TERMTHDACT=((DUMP,,0),OVR **- BATCH**

Shipped defaults: CICS: TERMTHDACT=((TRACE,MSGFL,0,OVR)

BATCH: TERMTHDACT=((TRACE,,0),OVR)

Flavors:	(QUIET)	- generates no message
	(MSG)	- generates message and a dump containing the trace table
	(TRACE)	- same as MSG, but includes trace back info
	(DUMP)	- same as TRACE, produces a LE/VSE DUMP
	(UADUMP)	- same as DUMP – and – a SYSTEM DUMP based on the JCL OPTION DUMP setting

Watch out ! “The use of UADUMP with database managers might bypass any backout processing that could be required” (DL/1). See LE/VSE Customization Guide page 108

TRAP=((ON,MAX),OVR)

Do you want backout to occur with ABTERMENC? Better turn TRAP ON…

Must be in effect “in order for applications to run successfully”

UPSI=((00000000),OVR)

Beware - This isn’t your COBOL UPSI – this is JOB CONTROL

New neat stuff with LE/VSE

Interactively Process LE/VSE run-time: CICS 2.3 and CICS/TS

CLER transaction

IBM supplied and in RDO. Displays and allows “on the fly” modifications to certain LE/VSE run-time options

NEWC transaction

IBM supplied and in RDO. Allows you to change and load all your LE/VSE options via CEEWCOPT. If CEECOPT was loaded into the SVA, you must issue another SET SDL prior

ROPC Transaction

IBM supplied and in RDO. Allows you to print LE/VSE CICS-WIDE run-time options to your z/VSE console

Customization of COBOL/VSE Compiler Options

Modify IGYWEOP1.Z

ALLOWCBL=YES

Allows both CBL and PROCESS cards

AWO=NO

Change to YES for blocked V-mode sequential files

BUF=4K

Leave it alone initially

CMPR2=NO

Default provides ANSI85 support - YES provides ANSI74 support

COMPILE=NOC(S)

Change COMPILE=NOC(E) unless you want “E” levels linked!

DATA=31

Recommend this for CICS

Doesn’t make any difference if program is linked as 24 bit

DYNAM=NO

Required for CICS - Can get around this by coding:

01 MY-PROG PICx(8) VALUE ‘MYPROG’

PROCEDURE DIVISION:

CALL MY-PROG using….

FASTSRT=NO

Change to YES for “Using and Giving”

FLAG=I

Consider other options such as FLAG=(I,W)

LIB=NO

Change to YES if you have any COPYBOOKS

LIST=NO

Change to YES for assembler output

LITCHAR=QUOTE

Everyone else uses APOST

MAP=NO

YES gives you a Data Division Map

NUMPROC=NOPFD

NUMPROC=MIG for DOS/COBOL compatibility

NAME=NO

Change to NAME=NOALIAS for CATALR compatibility

OFFSET=NO

Yes produces a condensed Procedure Division listing

OPT=NO

OPT=STD is probably what you want

OPT=FULL deletes code!

RENT=NO

YES is required for CICS programs

RMODE=AUTO

NORENT	- program links as 24 bit
RENT	- program links as 31 bit

SEQ=YES

Set it to NO!

SIZE=MAX

Watch out and do your own testing! BUFSIZE considerations?

SSRANGE=NO

Probably YES for TEST - NO for PRODUCTION

TEST=NO

Buy lots of DASD if you turn on Full function

Flavors:	NOTEST
	NONE,NOSYM
	NONE,SYM
	ALL,SYM

TRUNC=STD

TRUNC=BIN

For CICS, SQL, PL/1 and other S/370 format binary data

TRUNC=OPT

Watch out!

Lake County’s defaults

```
IGYCOPT
    ADV=NO,
    COMPILE=NOC(E),
    FLAG=(I,W),
    LIB=YES,
    LITCHAR=APOST,
    MAP=YES,
    NUMPROC=MIG,
    OFFSET=YES,
    OPT=STD,
    RENT=YES,
    SEQ=NO,
    SIZE=2560K,
    SSRANGE=YES,
    TEST=NO,
    TRUNC=BIN
END   IGYCDOPT
```

CICS, DL/1 and SQL(DB2) Dependencies

CICS Dependencies:

Compiler requirements: NOCMPR2, RENT, NODBCS, NODYNAM

Decide on DATA(24) or DATA(31)

AMODE/RMODE requirements

Preprocessor requirements: ANSI85 or COBOL2 or COBOL3(CICS/TS)

Initialization of applications working storage via the STORAGE LE/VSE run-time option?

DL/1 Dependencies:

Review APAR PN67649 introducing full 31 bit support and INFO APAR II08400

Should not have any problems with pure DOS/COBOL programs

Remove DLZBPJRA

Change entry point CBLCALLA to DLITCBL

Insure that LE ABTERMENC run-time option is set to ABEND and not RETCODE

SQL(DB2) Dependencies:

No problems encountered

Compiles and LE run-time environment is totally transparent with existing COBOL II programs

No changes in SQL “includes” ARI… during LNKEDT

Insure that LE ABTERMENC run-time option is set to ABEND and not RETCODE

Cannot address any issues pertaining to DOS/COBOL SQL/DB2 applications

Suggestions

If you are running multiple versions of COBOL, consider separate VSE sublibs for each flavor of COBOL and LE/VSE

Always insert the LE/VSE sublib ahead of any language library in a LIBDEF chain unless instructed otherwise

Recommend the use of a VSE/CICS monitor if you are going to make changes to compiler or LE/VSE run-time option(s)

Modify and change only 1 option at a time so you can measure the change

No one LE/VSE run-time option or COBOL/VSE compiler option will kill performance

Temper the combination of options in effect for both environments

Recommended reading and additional references

LE/VSE Run-Time Migration Guide (SC33-6687-00)

LE/VSE Customization Guide (SC33-6682-06)

LE/VSE Programming Guide (SC33-6684-04)

COBOL/VSE Migration Guide (GC26-8070-00)

COBOL/VSE Programming Guide (SC26-8072-02)

COBOL/VSE Language Ref. (SC26-8073-02)

COBOL/VSE and VisualAge COBOL MLE for VSE Installation and customization Guide (SC26-8071-01)

Nice link off the z/VSE homepage containing a lot of good information:

http://www-03.ibm.com/servers/eserver/zseries/zvse/support/le.html