



**SELCOPY C++ Version (SLC) 3.40 New Features**

8 Merthyr Mawr Road, Bridgend, Wales UK CF31 3NH

Tel: +44 (1656) 65 2222  
Fax: +44 (1656) 65 2227

**CBL Web Site - <http://www.cbl.com>**

This document may be downloaded from <http://www.cbl.com/documentation.php>

# Contents

<b>SELCOPY C++ Version (SLC) Release 3.40 New Features.....</b>	<b>1</b>
Documentation Notes.....	1
<b>Overview.....</b>	<b>2</b>
Platforms.....	2
Background.....	2
The Future.....	2
Program Naming.....	2
SELCOPY C++ Version Advantages.....	2
SELCOPY BAL Version Advantages.....	2
Recent History.....	3
<b>Important Changes.....</b>	<b>4</b>
Footing has Latest Change Date.....	4
CMS/VSAM Keyed read of ESDS or RRDS.....	4
<b>Fixes.....</b>	<b>5</b>
Fix 01 - CISIZE for VSAM reported (MVS or CMS).....	5
Fix 02 - POS RBA for VSAM corrected.....	5
Fix 03 - IF field NUMERIC gave ERROR 005.....	5
Fix 04 - Problems with numeric literals in quotes.....	5
Fix 05 - Parameters passed to a Subroutine.....	6
Fix 06 - LOG REPLY into a DCL CHZ var not terminated.....	6
Fix 07 - CHOP statement gave RC=8 for CHZ destination DCL var.....	6
Fix 08 - Regular Expressions.....	6
Fix 09 - RETCMS for CP command.....	6
<b>New Features and Other Changes.....</b>	<b>7</b>
Interactive Debug.....	7
Additional SORT orders for DIR/DIRDATA input.....	7
FILL param with INI=str for CHA var on DCL.....	8
DCL CHZ vars initialised to null by default.....	8
<b>Messages.....</b>	<b>9</b>
ERROR Messages - Control Statement Analysis.....	9
ERROR Messages - Selection Time.....	9
WARNING and Information Messages in Summary.....	9

# SELCOPY C++ Version (SLC) Release 3.40 New Features

---

## Documentation Notes

---

Information in this new feature document (CBL Ref SNF340c001) applies to the multi-platform C++ version of the mainframe SELCOPY Batch product. The C++ version of SELCOPY is referenced as SLC throughout this document which details changes introduced to SELCOPY (SLC) 3.40 Build 001 since SELCOPY (SLC) 3.30 Build 001. Note that these changes have not yet been included in the full SLC Language Reference manual.

Installation packages for the latest GA releases of SLC for each supported platform are available via the CBL web site (<http://www.cbl.com>) and ftp server (<ftp://ftp.cbl.com>).

Product packages are removed when no longer in support.

The **SELCOPY C++ version** for z/OS (MVS) and z/VM (CMS) operating systems is provided as an executable load module, SLC, which is included as part of the **SELCOPY Product Suite**, available for download and install from the CBL web site **SELCOPY Downloads** page (URL: <http://www.cbl.com/selcdl.php>). Corrective service is provided in the form of new build levels via z/OS SMP/E SYSMODs or a VM/CMS VMARC software update bundle.

The **SELCOPY C++ version** for IBM i, UNIX and Windows operating systems is provided as downloadable .TGZ and .ZIP archive files from the CBL web site **SELCOPY Downloads** page.

The following publication documents operation of the SELCOPY program for the C++ version. It is available in Adobe Acrobat PDF format from the CBL web site **Documentation** page:

- **SELCOPY C++ (SLC) Language Reference 3.40 User Manual**

Copyright in the whole and every part of this document and of the SELCOPY system and programs, is owned by Compute (Bridgend) Ltd, whose registered office is located at 8 Merthyr Mawr Road, Bridgend, Wales, UK, CF31 3NH, and who reserve the right to alter, at their convenience, the whole or any part of this document and/or the SELCOPY Product Suite system and programs.

No reproduction of the whole or any part of the SELCOPY system and programs, or of this document, is to be made without prior written authority from Compute (Bridgend) Ltd.

At the time of publication, this document is believed to be correct. CBL do not warrant that upward compatibility will be maintained for any use made of this program product to perform any operation in a manner not documented within the user manuals.

The following generic terms are used throughout this document:

<b>MVS</b>	-	IBM z/OS, OS/390, MVS/ESA, MVS/XA, MVS/SP, OS.
<b>CMS</b>	-	IBM z/VM, VM/ESA, VM/XA, VM/SP.
<b>AIX</b>	-	IBM AIX
<b>DEC</b>	-	HP Tru64
<b>HPX</b>	-	HP HP-UX
<b>LNX</b>	-	Linux x86 (RHEL or SLES)
<b>LNZ</b>	-	z/Linux (RHEL or SLES)
<b>SCO</b>	-	UnXis (SCO) UnixWare
<b>SUN</b>	-	SUN Sparc Solaris
<b>WNT</b>	-	Microsoft Windows x86 (NT, Vista, 7, Server 2008)
<b>AS/400</b>	-	IBM i, i5/OS, OS/400
<b>UNIX</b>	-	AIX, DEC, HPX, LNX, LNZ, SCO and SUN.
<b>PC</b>	-	x86 servers or workstations running Microsoft MS-DOS or Windows.
<b>ALL</b>	-	AIX, DEC, HPX, LNX, LNZ, SCO, SUN, MVS, CMS, AS/400 and WNT.

# Overview

---

Overview on the evolution of the C++ version of the SELCOPY Batch utility product.

---

## Platforms

---

This document applies to the following platforms:

- UNIX. (For supported platforms please see [Documentation Notes](#) above.
  - Microsoft Windows.
  - IBM Mainframe z/OS (MVS) and z/VM (CMS).
- 

## Background

---

SELCOPY for the IBM mainframe, first released in 1971, is written in BAL (Basic Assembler Language) and is ongoing.

SELCOPY for the various UNIX and PC platforms and the, first released in 1996, is written in C++ for the benefit of portability between platforms. It has essentially the same syntax as the BAL version.

Certain features in the C++ version now more than justify its release to mainframe users, giving them many benefits, in particular, the ability to read lists from the SELCOPY Interactive environment.

---

## The Future

---

The BAL version for the mainframe will ultimately be phased out, but not until the C++ version has been adapted to call BAL subroutines for all the critical, cpu-intensive, parts of the product. Thus the raw power of the BAL version will be retained, with the C++ overhead apparant only in the control card analysis at start up.

---

## Program Naming

---

The IBM Mainframe C++ version will be known as **SLC**.

The IBM Mainframe BAL version will remain as **SELCOPY**.

The IBM i, UNIX and Windows C++ versions may remain as SELCOPY, or be renamed to SLC, depending on the installation's preference.

---

## SELCOPY C++ Version Advantages

---

- Reads lists as provided by SELCOPYi. e.g. Vols, DataSets, Members, Queues, Allocs.
  - Command line invocation can provide all control statements on the PARM field.
  - Case insensitive compare.
  - Reverse scan.
  - CSV (Comma Separated Variables) support.
  - TYPE=C and mixed TYPE arithmetic.
  - INCLUDE statement.
  - CVDATE statement for converting date formats.
  - HEX offsets supported.
  - DECLARE variables
  - Multiple fields on a PRINT statement.
  - CHANGE statement.
  - RGX giving the power of unix-type Regular Expressions for IF and CHANGE statements.
- 

## SELCOPY BAL Version Advantages

---

- Raw speed.
  - DB2, IMS, ADABAS support.
-

## Recent History

Platform	SELCOPY (SLC) Release	Build Level	Latest Change	Publish Date	Comments
MVS Mainframe	3.00	001	2010/12/01 23:16	2010/12/16	GA. s300 Product Suite dist
WNT Windows	3.00	004	2011/03/01 22:32	2011/03/01	GA.
WNT Windows	3.00	006	2011/05/25 22:28	2011/06/08	GA.
LNx Linux	3.00	007	2011/06/17 11:38	2011/06/22	GA.
ALL SNF300c	3.00	007		2011/06/24	GA. Documentation.
MVS Mainframe	3.00	009	2012/03/04 21:28	2012/03/15	GA. s310 Product Suite dist
MVS Mainframe	3.10	001	2012/04/10 17:51	2012/04/12	GA. (sysmod)
MVS Mainframe	3.10	004	2012/09/21 17:10	2012/09/25	GA. (sysmod)
CMS Mainframe	3.10	004	2012/09/12 17:48	2012/09/20	GA.
WNT Windows	3.10	004	2012/09/12 17:48	2012/09/20	GA.
ALL SNF310c	3.10	004		2012/09/24	GA. Documentation.
For ease of reference, all new features in the SNF300c are repeated in SNF310c. Thus, SNF310c covers all new features, up to SLC 3.10 Build 004, that are not yet officially published in the SELCOPY manual.					
CMS Mainframe	3.10	005	2012/12/13 15:12	2012/12/19	GA.
HPX HP-UX	3.10	006	2013/01/27 21:58	2013/02/08	GA.
LNx Linux	3.10	007	2013/05/10 12:06	2013/05/10	GA.
MVS Mainframe	3.10	008	2013/05/27 17:12	2013/06/21	GA.
CMS Mainframe	3.10	008	2013/05/27 17:12	2013/06/21	GA.
WNT Windows	3.10	008	2013/05/27 17:12	2013/06/21	GA.
LNx Linux	3.10	008	2013/05/27 17:50	2013/06/21	GA.
MVS Mainframe	3.20	001	2013/10/19 22:09	2013/10/22	GA. s320 Product Suite dist
CMS Mainframe	3.20	001	2013/10/19 22:09	2013/10/22	GA.
WNT Windows	3.20	001	2013/10/19 22:09	2013/10/22	GA.
LNx Linux	3.20	001	2013/10/19 22:09	2013/10/22	GA.
ALL SNF320c	3.20	001		2013/10/23	GA. Documentation.
MVS Mainframe	3.30	001	2015/02/19 14:40	2015/04/22	GA. s330 Product Suite dist
CMS Mainframe	3.30	001	2015/02/19 14:40	2015/04/22	GA.
WNT Windows	3.30	001	2015/02/19 14:40	2015/04/22	GA.
LNx Linux	3.30	001	2015/02/19 14:40	2013/10/22	GA.
ALL SNF340c001	3.30	001		2015/04/22	GA. Documentation.
MVS Mainframe	3.40	001	2017/10/07 12:20	2017/11/03	GA. s340 Product Suite dist
CMS Mainframe	3.40	001	2017/10/07 12:20	2017/11/03	GA.
WNT Windows	3.40	001	2017/10/07 12:20	2017/11/03	GA.
LNx Linux	3.40	001	2017/10/07 12:20	2017/11/03	GA.
ALL SNF340c001	3.40	001		2017/11/03	GA. Documentation.

# Important Changes

---

There are 2 important changes which can give different results from earlier releases of SLC, the C++ version of selcopy:

- **Footing has Latest Change Date**
- **CMS/VSAM Keyed read of ESDS or RRDS (VM/CMS platform ONLY.)**

These IMPORTANT CHANGES are discussed below.

---

## Footing has Latest Change Date

2015/07/22 s330\_002

### All platforms.

Note that while documented here as IMPORTANT, the change to the footing will only affect those applications where the SELCOPY listing is post-processed, so for most users the change is quite harmless.

The footing line at the bottom of SLC's listing file has been changed to include the latest change date of that version of the product. e.g.

```
** SELCOPY/WNT 3.30.002 Licensed by Compute (Bridgend) Ltd +44 (1656) 652222 & 656466 **
```

becomes:

```
** SELCOPY/WNT 3.30.002 2015/08/03 Compute (Bridgend) Ltd +44 (1656) 652222 & 656466 **
```

---

## CMS/VSAM Keyed read of ESDS or RRDS

2015/04/09 s330\_001

### ONLY for VM/CMS platform.

For the VM/CMS platform only, using the IBM-supplied CMS/VSAM interface for accessing VSE/VSAM files, which is now "Out of Support" by IBM, the keyed read of an ESDS or RRDS, by RBA or REC respectively, no longer points to the requested VSAM record, resulting in the selcopy message ERROR 544.

Keyed read of a KSDS still operates correctly.

The problem, under review by IBM, ref no: PMR 38722,999,866, which was opened on 2015/02/04, remains outstanding at 2017/10/25.

# Fixes

---

---

## Fix 01 - CISIZE for VSAM reported (MVS or CMS)

2015/08/06 s330\_002

**ONLY for MVS and CMS platforms.**

The CISIZE and FSIZE values for a VSAM file are now reported in the summary by SLC, in the same way as the mainframe BAL version, SELCOPY.

Note that the FSIZE (File Size) reported is the file size at OPEN time, which will not take into account any insertions or deletions made during the SLC run.

e.g.

```
SUMMARY..
SEL-ID      SELTOT      FILE      BLKSIZE  LRECL      FSIZE  CI    DSN
-----
          2          17  READ  TESTE      120    71 U      17   512 ESDS.TEST.FILE
```

Previously the values reported by SLC were blank.

---

## Fix 02 - POS RBA for VSAM corrected

2015/08/22 s330\_002

**ONLY for MVS and CMS platforms.**

The RBA value of the latest VSAM record read on earlier releases of SLC was wrong.

Note that the RBA value is held as an 8-byte binary number within SLC's storage, but, for compatibility with older architecture, POS RBA still refers to the start of the junior 4-bytes of the RBA. To reference the 8-byte RBA, please use POS RBA-4 instead.

e.g.

```
print from 4 at RBA   type=b format='      z,zzz,zzz,zz9'    ' = RBA Length 4.'
print from 8 at RBA-4 type=b format='zzz,zzz,zzz,zzz,zz9'    ' = RBA Length 8.'
```

---

## Fix 03 - IF *field* NUMERIC gave ERROR 005

2015/09/13 s330\_002

ERROR 005 (PARAM ARGUMENT REQD) was erroneously issued when the field to be tested was coded before the NUMERIC keyword, (synonym NUM), and the NOT keyword was not used.

e.g.

```
if  DCLvar      NUMERIC      * Failed with ERROR 005.
if  8 at 101    NUM          * Failed with ERROR 005.

if          NUMERIC  DCLvar    * Accepted ok.
if          NUM      8 at 101  * Accepted ok.
if  DCLvar    NOT    NUM      * Accepted ok.
if  8 st 101  NOT    NUM      * Accepted ok.
```

---

## Fix 04 - Problems with numeric literals in quotes

2015/09/20 s330\_0022

A variety of errors have been fixed relating to the use of quoted numeric literals which form part of an expression.

Quoted numeric literals are used simply to tolerate punctuation chars, such as commas and blanks, in order to improve readability.

e.g.

```
POS '1,000'+1 = 'XYZ' * Gave ERROR 277 - DATA FOLLOWS IMMED AFTER TERMINATING QUOTE.
```

---

**Fix 05 - Parameters passed to a Subroutine**

---

2016/04/13 s330\_002

A variety of errors have been fixed, where the data type of a parameter passed to the subroutine was not properly established by the subroutine. This includes errors involving nested subroutines, where a parameter passed to the parent subroutine is then passed as a parameter to a nested subroutine.

---

**Fix 06 - LOG REPLY into a DCL CHZ var not terminated**

---

2016/10/01 s330\_004

When a message to the operator requires a REPLY and the reply area coded is a DCL var, which is defined as a CHZ (a Null Terminated String, synonyms: NTS, CHAZ, CHARZ), the necessary 'X'00' terminator for the CHZ var was not written to the REPLY area, resulting in residual data corrupting the reply data.

---

**Fix 07 - CHOP statement gave RC=8 for CHZ destination DCL var**

---

2016/12/09 s330\_004

If one or more of the destination vars for a CHOP statement were coded as DCL CHZ vars, (Null Terminated Strings, synonyms: NTS, CHAZ, CHARZ), the current length of the CHZ var, rather than its maximum length, was used to determine if overflow had occurred, resulting in an erroneous RC=8 when the new value was larger than the previous.

---

**Fix 08 - Regular Expressions**

---

2017/01/09 s330\_004

A variety of errors have been fixed, where the RGX (Regular Expression) used on an IF or CHANGE statement, is more advanced and complicated. Typically, an RGX involving nested alternations, often containing wild strings.

---

**Fix 09 - RETCMS for CP command**

---

2017/07/13 s330\_007

**ONLY for the VM/CMS platform.**

For the VM/CMS platform, SLC supports issuing a CP command in the same way as supported by SELCOPY, the BAL version.

However, the return code from the CP command did not get set in the 4-byte binary storage allocated within SLC's storage which the user can reference as POS RETCMS.



# New Features and Other Changes

---

## Interactive Debug

2017/04/02 s330\_005

**ONLY for MVS and CMS platforms.**

The Selcopy Debugger for the C++ version (SDC) is a large topic and is documented separately in the SELCOPYi pages, because it is only available for the CMS and MVS platforms.

See: [SELCOPY Product Suite 3.40 New Features](#)

Briefly, SDC allows the user to start SLC under the control of SELCOPYi in a windowed environment on a 3270, interactively displaying the SLC control statements in 1 window, the output listing in another window, and the WORKAREA storage in another.

Optionally, the user may open other windows to display:

- the LOG output
- the @ pointers
- a WATCH window for DCL vars and fields
- any addressable storage area

The PF keys may be used to step through the control statements 1 at a time, while observing the changes to the data displayed, or break points may be set and the job run until a break point, or EOJ, is reached.

The control statements can then be modified in the control window and the job rerun, keeping all the other windows still active.

Amazingly useful when a complex job inexplicably keeps giving the wrong result.

---

## Additional SORT orders for DIR/DIRDATA input

2015/08/06 s330\_002

The SORT parameter for DIR or DIRDATA input on a READ statement has been enhanced to support additional sort orders which provide the reverse order of the original standard sort orders.

The sort orders now supported are:

Standard sort orders		Additional sort orders	
Code	Meaning	Code	Meaning
D	Date descending	T	Date ascending
S	Size descending	Z	Size ascending
N	Name ascending	M	Name descending
E	Extn ascending	X	Extn descending
P	Path ascending	H	Path descending

e.g.

```
read IFIL dsn=ABCD.* DIRD SORT=t into=DCLvar fill=' ' * By DATE ascending.
```

---

## FILL param with INI=str for CHA var on DCL

---

2016/02/08 s330\_002

For a DCL statement defining a CHA variable, the FILL (synonym PAD) parameter may be coded with a single byte char or hex argument, which is then used to populate every remaining character of the CHA variable declared that does not form part of the INI string coded.

The INI and FILL params may also be coded on DCL vars that overlay a previously defined var, or even a combination of consecutive vars, using the `POS=varname+offset` syntax.

Initialisation of DCL vars is done in the order they are defined, so beware of unintentionally overlaying a previous INI with different data by using INI and FILL on a DCL which uses POS to overlay an existing DCL var.

However, this can be used to advantage, by filling a var with one CHA, then overlaying part of it with a different CHA. e.g.

```
dcl ABC cha (50) INI='- ' FILL='- '
dcl ABC2 cha (10) INI 'o' FILL='o' POS ABC+20 * 2nd INI of same data.

* Giving:
* "-----o-----"
```

---

## DCL CHZ vars initialised to null by default

---

2016/09/29 s330\_004

DCL CHZ vars, (Null Terminated Strings, synonyms: NTS, CHAZ, CHARZ), when first introduced, were by default initialized to all blanks up to its maximum length.

Because a Null Terminated String should contain only data, it is misleading to fill it with blanks by default when no initialization string was coded by the user on the DCL statement.

The default initialization for DCL CHZ vars has therefore been changed to the null string, where the null terminator (X'00') terminates the zero length data.

# Messages

---

## ERROR Messages - Control Statement Analysis

---

**E225 DOUBLE NEGATION ON IF STMT**  
e.g.

```
IF NOT 8 AT 101 NOT NUMERIC * Error.
```

**E226 DCL POS/LEN EXCEEDS WORKLEN**

When declaring a DCL var for a field which overlays data within the work area, the field may not overlap the size of the work area defined by the WORKLEN parameter on an OPTION stmt.  
e.g.

```
OPT WORKLEN 200
DCL ABC1 CHA(10) POS=192 * Error 226.
DCL ABC2 CHA(10) POS=191 * Would be ok.
DCL ABC3 CHA( 9) POS=192 * Would be ok.
```

**E227 DCL POS/LEN EXCEEDS CURRENT DCL ALLOCATION**

When declaring a DCL var for a field which overlays data that has been previously declared within the dynamic DCL work area, the field may not overlap the size of the dynamic DCL work area accumulated so far.  
e.g.

```
DCL IKEY CHA( 10 ) * Cust key.
DCL INAM CHA( 30 ) * Name.
DCL IADDR1 CHA( 30 ) * Address line 1.
DCL IADDR2 CHA( 30 ) * Address line 2.
DCL IADDR3 CHA( 30 ) * Address line 3.

DCL IREC CHA(131 ) POS IKEY * Error 227.
DCL IREC CHA(130 ) POS IKEY * Would be ok.
```

**E228 M/F DEBUG STMT DEFINITION ERROR**

An interactive debug session has given an error when calculating the start and end columns of the current statement. This is not normally a user error. Please report the problem to CBL. Thank you.

**E229 ... E269 RESRVD**

All message numbers between E229 and E269 (inclusive) are reserved.

---

## ERROR Messages - Selection Time

---

**E627 M/F DEBUG - UNSUPPORTED DCL DATA TYPE**

While running SLC under interactive debug, (available only on the mainframe, CMS or MVS), a DCL var has been encountered with an unsupported data type.

**E628 RESRVD**

Message number E628 is reserved.

---

## WARNING and Information Messages in Summary

---

Changes to the text of the information messages that may occur on the summary are:

None.

---