Suprtool 6.7 for HP-UX:

Change Notice

by Robelle Solutions Technology Inc.

Robelle Solutions Technology Inc.
Suite 372, 7360 137 Street
Surrey, BC Canada V3W 1A3

Phone: 604.501.2001
Support: 289.480.1060
E-mail: sales@robelle.com
E-mail: support@robelle.com
Web: www.robelle.com

Contents

Introducing Suprtool Version 6.7 6

Overview 6
Highlights in Suprtool 6.7 6
Highlights in Suprtool 6.6 6
Highlights in Suprtool 6.5 6
Highlights in Suprtool 6.4 7
Itanium Dynamic Loading 8
Compatibility 8
Itanium Compatibility 8
CPU Serial Number (uname) 8
Documentation 8

Installation 9

Overview 9
Installation Instructions 9
Installation Assistance 9

Enhancements in Version 6.7 10

Introduction 10
$SPLIT Function 10
Set QuoteChar 10
Set SmartSplit 10
$Dynamic Loading of Eloquence Routines 11
Support for Eloquence 8.4 12

Enhancements in Version 6.6 13

Introduction 13
STExport 13
INDENT 13
Header and Tail 14
$CHANGE Function 15
Oracle Add 15

Enhancements in Version 6.5 16

Introduction 16
$Change Function 16

Enhancements in Version 6.4 17

Introduction 17
Rport 17

Enhancements in Version 6.3 18

Introduction ... 18
Set SDItemNoExt ... 18

Enhancements in Version 6.2 19

Introduction ... 19
Rport Set Pagelen ... 19
Rport Subtotal ... 19
Rport Set Maxlen ... 19
Rport Comma ... 19
Rport Hide ... 20
System Commands ... 20
$Month($stddate()) ... 20
Rport ... 20
$BOM and $EOM ... 22
String Function Checks ... 22
Sdlinux for Linux ... 23
Set Limits Override On | Off ... 23

Enhancements in Version 6.1 25

$Length ... 25
$LENGTH (Works on Byte-type and numeric fields) ... 25
If Usage: ... 25
Extract Usage: ... 25
$ROBELLE variable ... 25
Set EditSignNeutral ... 25
STEXPORT ... 26
SUPRLINK ... 26

Enhancements in Version 6.0 27

$List NOSAMETO ... 27
BackwardChain ... 27
$INRECNUM ... 27
$LEADZEROZ ... 27
$LEADZEROB ... 28
You can clean it up with the following: ... 28
$JUSTIFYL ... 29
$JUSTIFYR ... 29
$RESPACE ... 30
SDLINUX UTILITY ... 31

Bugs Fixed 34

Bugs Fixed In Suprtool 5.8.11 ... 34
Bugs Fixed In Suprtool 5.8.10 ... 34
Bugs Fixed In Suprtool 5.7.11 ... 34
Bugs Fixed In Suprtool 5.7 ... 34
Bugs Fixed In Suprtool 5.6.11 ... 34

Introducing Suprtool Version 6.7

Overview

Use Suprtool/UX to read, select, and sort data from Oracle, Allbase and Eloquence databases and data files with fixed-length records. Suprtool/UX is designed to be similar to Suprtool for MPE while providing necessary HP-UX features.

Suprlink/UX provides high-speed data-file linking based on a sort key. Use STExport to convert fields in a self-describing input file into an output file that can be imported into different applications.

Highlights in Suprtool 6.7

- The $split function has been improved with two set commands set quotechar and set smartsplit. Set quotechar aids in removing quotes from byte type csv fields. Set Smartsplit, in conjunction with set quotechar detects and ignores the split character inside of quotes.
- Suprtool dynamic loading of Eloquence libraries has been improved.
- Suprtool supports Eloquence version 8.4 but does not take advantage of blob support.

Highlights in Suprtool 6.6

- STExport/UX/Open now supports Extended SD information.
- STexport has improved buffer handling, and the amount of overhead has been improved by over 50%.
- Stexport has new features in the Json command. Namely, Indent and Header and Trailer options to add data to the output file.
- The Oracle Add command parsing has been improved. It would fail if there were more than 62000 tables in the given database/ownername.
- The $CHANGE function has been improved to update data directly in the case of an UPDATE.

Highlights in Suprtool 6.5

- Suprtool has a new function called $change which will change byte type field data from one string to another.

Highlights in Suprtool 6.4

- Rport spacing and alignment has been improved for subtotal and total lines. (All Platforms)
- Rport spacing of detail lines has been improved to allow for a space after each byte field. (All Platforms)
- Suprtool has a new command called set sdinbypass on, allows the reclen, lf syntax on an input file if it is self-describing. This was done to allow some customers to not have to change scripts. (Linux Only)
- Input files with Bigendian data are converted to Little endian about three times faster in all cases. (Linux Only)
- Suprtool when running with set ffisbe on and set endianint BE on a sort operation with extract commands would see suprtool incorrectly convert the data from LE to BE twice. (Linux Only Fixed in 6.3.50 Build 6)
- Suprtool when there was an input source of an SQL table and sorting a field with an extract of fields, and set ffisbe on and set endianint, log and Ieee to BE would incorrectly convert the data to Big Endian on the Input side. (Linux Only. Fixed in 6.3.50 Build 6)
- An input of a non-sd file with defines and extracts would assume for BigEndian to Little Endian as a Self-Describing File. (Linux Only. 6.3.50 Build 2)
- A subsequent Suprtool task with large key sorts would fail when the previous step was a step that accessed an Oracle database and the Oracle database was still open. (Linux Only. Suprtool 6.3.01 Build 2)
- Suprtool no longer requires that you need to extract a field from a Flat file (ffisbe on) or Self-Describing file in order for comands like if, sort, dup none keys total field, and total, to correctly convert the data from Big Endian to Little Endian. (Linux Only. 6.3.01 Build 14)
- Suprtool / Open didn't handle double logical numbers over 2 billion when converting to ascii in the list command. (Linux Only. Fixed in 6.3 Build 2)
- Sorting with certain files would not allocate enough memory space to complete the sort. (Linux Only. Fixed in 6.3 Build 5)
- An if $lookup would not work all times with certain data fields if the data field was Big Endian. (Linux Only. Fixed in 6.3 Build 5)
- A Chain with multiple values specified in line with byte or logical field types would fail. (Linux Only)
- Output,ascii and Stexport, conversion of double integers with decimal places would incorrectly convert, numbers where the value was less than the number of decimal places. (Linux Only.)
- Output, ascii and Stexport would incorrectly handle single logical conversions for some values. (Linux Only)
- Output,ascii would incorrectly pad uneven byte output files with a null on the first record. (Build 14)
- Output,ascii of a packed field would default to outputting a sign on all fields, when the MPE and HP-UX versions would default to no sign unless negative. (Fixed in 6.3.50 Build 4)

Itanium Dynamic Loading

SHLIB_PATH

By default and without ROBELLE_DYN_LOAD set to Y, Suprtool for HP-UX will look for the libraries in SHLIB_PATH.

LD_LIBRARY_NAME

You can change Suprtool to pay attention to LD_LIBRARY_PATH and use dlopen to load libraries by setting the environment variable:

```
export ROBELLE_DYN_LOAD =Y.
```

It is generally advisable to use DLOPEN or the ROBELLE_DYN_LOAD option when using the Itanium version.

Compatibility

Fastread option is now set to Off for Suprtool for HP-UX. The Dynamic loading version of Suprtool the option is set to off.

Itanium Compatibility

There are a couple of differences in the Itanium version of Suprtool, which you need to be aware of.

The DBEdit module is not available in the Itanium versions of Suprtool. This module is invoked by the EDIT command.

The layout for the Verify command is slightly different then the PA-RISC versions of Suprtool.

CPU Serial Number (uname)

This program runs only on CPUs whose serial numbers have been encoded (the "uname" on HP-UX). If it fails to run and you get an "invalid HPSUSAN" error message, contact Robelle for assistance, via support@robelle.com or the support number at 1.289.480.1060.

Documentation

The user manual contains the full description of all the Suprtool suite of products including Dbedit, Suprlink, STExport, and Suprtool2, as well as usage tips and commands for each. The manuals are up-to-date with all the latest changes. To see only the changes in the latest version, see the "What's New" section of the manual.

Installation

Overview

The following instructions describe the installation process of a new Suprtool release. The new version overwrites an existing version of Suprtool on your HP-UX system.

Installation Instructions

All Robelle software is available via a download link which you can get to by filling out this form:

http://www.robelle.com/forms/request-prod-release.php

If you don't know your Account number please provide enough information for us to find you in our database.

You can find the HP-UX download install instructions here:

http://www.robelle.com/downloads/install-sxprod.html

Installation Assistance

If you have any questions or run into any problems, please call us. Technical support is available on weekdays from 8 a.m. to 4 p.m. Pacific time at 1.289.480.1060. Technical support can also be obtained via e-mail at: support@robelle.com If your new version of software will not run, you can call technical support by calling the support number, or you can typically easily run extend with the disaster option to tide you over until business hours. Instructions for this are available at:

http://www.robelle.com/disaster/

You can download our manuals and Change Notices in various formats from:

http://www.robelle.com/library/manuals/.

Enhancements in Version 6.7

Introduction

Suprtool is constantly being updated with new features. The following section describes the new enhancements to Suprtool since Suprtool 6.6

$SPLIT Function

The $Split function has two new options in how data is processed, specifically Set QuoteChar and Set SmartSplit, which work together to process "byte-type" fields in a csv file. Set quotechar, defines what quote character was used to in creating the CSV file so that Suprtools split function will know when it is in quotes. When the Split function knows this, it will automatically remove the quotes.

Set QuoteChar

The Set QuoteChar, tells the Suprtool Split function what Quote character surrounds byte type fields in a CSV file. If you need to define the double quote as the Quote character you define by surrounding the double quote with single quotes and vice versa.

The commands:

```
   $set QuoteChar '"'  {define double quote as quote character}
   $Set QuoteChar "'"  {define single quote as quote character}
```

Set SmartSplit

Set SmartSplit On, in conjunction with Set QuoteChar, will allow the Split Function to ignore any occurrence of a Split Character inside the defined Quote Character.

Impact of Set QuoteChar '"' and Set SmartSplit On.

```
>Set QuoteChar '"'
>Set SmartSplit On
```

Examples

When doing a typical $split operation, you may not get the results you're looking for using the default method:

```
>in alana
>def record,1,80
>def name,1,30
>def bytenum,1,6
>ext name=$split(record,first,",")
>ext bytenum=$trim($split(record,",",1,","))
>list
>xeq

>IN ALANA.NEIL.GREEN (0) >OUT $NULL (0)
NAME            = "Neil Armstrong"
BYTENUM         = "1234"

>IN ALANA.NEIL.GREEN (1) >OUT $NULL (1)
NAME            = "Peanut
BYTENUM         = dadog"
```

With Set Smartsplit on and defining the Quote character, $split can do some extra work for you. Specifically, with Smartsplit on, the function can pass over the defined split character and treat it like text. The quotechar tells smartsplit when your in data that is byte AND what character remove.

```
>set smartsplit on
>set quotechar '"'
>in alana
>def record,1,80
>def name,1,30
>def bytenum,1,6
>ext name=$split(record,first,",")
>ext bytenum=$trim($split(record,",",1,","))
>list
>xeq
>IN ALANA.NEIL.GREEN (0) >OUT $NULL (0)
NAME            = Neil Armstrong
BYTENUM         = 1234

>IN ALANA.NEIL.GREEN (1) >OUT $NULL (1)
NAME            = Peanut,dadog
BYTENUM         = 4567
```

This helps the $SPLIT function do more work and often eliminate another pass thru the data.

$Dynamic Loading of Eloquence Routines

The routines and order of the loading of the dynamic loading of libimage3k and libeqdb. Suprtool will try to load the routines in the following order.

```
libimage3k.sl
lib/libimage3k.sl
/opt/eloquence/current/lib/libimage3k.sl
/opt/eloquence/8.4/lib/libimage3k.sl
/opt/eloquence/8.3/lib/libimage3k.sl
/opt/eloquence/8.2/lib/libimage3k.sl
/opt/eloquence/8.1/lib/libimage3k.sl
/opt/eloquence/8.0/lib/libimage3k.sl
```

The first two calls are how they have always been called and this relied on the setting of SHLIB_PATH or LD_LIBRARY_PATH, which often allows the library to be loaded.

From there we attempt to load the full name of the library, however, we first try to load using the directory name called current. We found over the years that to test or

try a newer version of Eloquence, we wanted a way to point to a new directory or a different directory. We use a symbolic link to point to a directory that we use for whatever the current version we want to use. We found it easier to do this to control and test new versions.

Symbolic Link to Current

In order to create a symbolic link to point to a one of the versions we want as the "current" version we simply do:

```
ln -s 8.1 current
```

This creates a symbolic link to make the current directory to point to 8.1 and have all directories under 8.1.

Run suprtool with -lw (loader warnings)

You can review, the loading that suprtool tries and when it gets resolved. In the case below, it is resolved at the time we use the current directory which we setup with the symbolic link:

```
suprtool -lw
SUPRTOOL/UXia/Copyright Robelle Solutions Technology Inc. 1981-2025.
(Version 6.7 Internal)  Wed, Feb 19, 2025,  1:55 PM  Type H for help.
Build 1
Loader Warning eloq_findsym_load shl_load of libimage3k.sl
Loader Warning eloq_findsym_load shl_load of hpux32/libimage3k.sl
Loader Warning eloq_findsym_load shl_load of
current/hpux32/libimage3k.sl
Loader Warning eloq_findsym_load shl_load of libeqdb.sl
Loader Warning eloq_findsym_load shl_load of hpux32/libeqdb.sl
Loader Warning eloq_findsym_load shl_load of
current/lib/hpux32/libeqdb.sl
```

Support for Eloquence 8.4

Suprtool has been tested with Eloquence 8.4 and it is supported, however, datasets with blobs will not be extracted from unless the blob is not included in the extract.

Enhancements in Version 6.6

Introduction

Suprtool is constantly being updated with new features. The following section describes the new enhancements to Suprtool since Suprtool 6.5

STExport

STExport, HP-UX and Open versions now supports extended SD information. STExport will use the SD information if the input self-describing file was created with Set SDExtname On with Suprtool, and Set Sdextname is set to on inside STExport.

STExport, all versions, has improved buffer handling and the amount of overhead has been significantly reduced.

STExport, has three new features in the JSON command. Specifically, INDENT, HEADER and TRAILER.

INDENT

The Indent command allows you to specify an integer, in which STExport will indent the data portion of the JSON output the number of spaces specified.

The Indent option does not impact the Header or Tail lines or the Object Lines.

The commands:

```
$in jsonin.stexptst
$json oneperline indent 12
$json Object "This is a test"
$out *
$xeq
```

Will produce data that looks as such:

```
{"This is a test":
         [{"CHAR-FIELD":"11111",
         "INT-FIELD":1111,
         "DBL-FIELD":11111,
         "PACKED-FIELD":11111,
         "PACKED*-FIELD":11111,
         "QUAD-FIELD":11111,
         "ID-FIELD":1,
         "LOGICAL-FIELD":1111,
         "DBLLOG-FIELD":11111,
         "ZONED-FIELD":11111
         }]
}
```

Header and Tail

The Header and Tail options of the JSON command, allow you to maintain two files, whereby the contents of the files are added to the Json stream, at the beginning and end of the output file.

The header and tail files are created to be fixed length files with a record size of 256 bytes.

The options of the Header and Tail command are as follows:

```
JSON HEADER CREATE <filename>
JSON HEADER ADD "This is a string"
JSON HEADER DELETE <filename>
JSON HEADER SETFNAME <filename>
```

There four options for the Header and Tail commands tell STExport to do the following:

Create <filename>

The create action opens the specified file name, if the file exists, it will erase the contents of the file.

ADD "string"

Allows you to add lines to the specified Header or Tail file, by specifying a string to add the the Head or Tail file.

Delete <filename>

Deletes the specified filename, for either the Header or Tail File specified.

SetFname

Sets the name of the header or Tail file and confirms that it exists.

Examples

The following STExport code shows using the new features and the impact on the Output.

```
input jsonin
JSON oneperline indent 12 header create file24h tail create file24t
JSON TAIL ADD "        }"
JSON TAIL ADD "}"
JSON HEADER ADD '{'
JSON HEADER ADD '  "Recreation_Membership_File": {'
JSON HEADER ADD '       "description": "Recusers_File_Ver_1",'
JSON HEADER ADD '       "fileType": "ACTIVITIES",'
JSON HEADER ADD '       "DataType": "Membership",'
JSON HEADER ADD '       "fullFileMember": "Y",'
JSON HEADER ADD '       "MemberActivity":'
output file24a
exit
```

would produce the following:

```
{
 "Recreation_Membership_File": {
      "description": "Recusers_File_Ver_1",
      "fileType": "ACTIVITIES",
      "DataType": "Membership",
      "fullFileMember": "Y",
      "MemberActivity":
           [{"CHAR-FIELD":"11111",
           "INT-FIELD":1111,
           "DBL-FIELD":11111,
           "PACKED-FIELD":11111,
           "PACKED*-FIELD":11111,
           "QUAD-FIELD":11111,
           "ID-FIELD":1,
           "LOGICAL-FIELD":1111,
           "DBLLOG-FIELD":11111,
           "ZONED-FIELD":11111
           }]
       }
}
```

$CHANGE Function

The $CHANGE function has been improved to update data directly in the case of an UPDATE.

Oracle Add

The parsing of the Oracle Add command would fail if the database based on User/Owner had more than 62000 tables.

Enhancements in Version 6.5

Introduction

Suprtool is constantly being updated with new features. The following section describes the new enhancements to Suprtool since Suprtool 6.4

$Change Function

Suprtools $change function returns a string that was changed by the $change function which, takes three parameters. The first is the field, that contains the data you want to change, the from string and the to string, which is to be enclosed in quotes.

Sample Extract Usage:

```
extract Name=$change(name,"Neal","Neil")
```

In the example above the first string of Neal, is the from string, and the to string is Neil.

If you need to change a phrase with a double quote then you can enclosed the strings in a single quote:

```
extract xml_input=$change(xml_input,'"LP"','LP')
```

The primary use of the $change function is to update fields using the extract command, however, you can use it in the if command, however, the field you are changing will appear in the output file as the changed record. Note the examples below:

```
if $change(name,"Neal","Neil")="Neil"
if ("Neil"=$change(name,"Neal","Neil"))
```

A reminder that if you start the if equation with the constant string first, you need to encase the complete equation in brackets.

Enhancements in Version 6.4

Introduction

Suprtool is constantly being updated with new features. The following section describes the new enhancements to Suprtool since Suprtool 6.3

Rport

Rport has had the spacing re-vamped for Header, Detail, Subtotal and Total lines. The spacing has been improved to insure that fields are easier to read and align properly.

Enhancements in Version 6.3

Introduction

Suprtool is constantly being updated with new features. The following section describes the new enhancements to Suprtool since Suprtool 6.2

Set SDItemNoExt

Non-MPE versions of Suprtool didn't update the Item information, decimals and date like the MPE version does, when an extract command wasn't entered. You can turn this off with set sditemnoext off, but the default is on to have it behave in the same manner as the MPE version.

Enhancements in Version 6.2

Introduction

Suprtool is constantly being updated with new features. The following section describes the new enhancements to Suprtool since Suprtool 6.1.10.

Rport Set Pagelen

Rport will Page Break by default at 24 lines, you can set a Page Break to any integer value greater than 10. When Rport reaches the number of lines for a page the Titles and headings are re-printed and Rport continues on printing, detail and subtotal lines.

Rport Subtotal

Rport will follow all levels of sort specified in the Self-Describing file, and will produce a subtotal line when each level break is made.

Rport Set Maxlen

Rport will by default calculate the maximum size of a field, including, commas, decimals and dollar signs. The previous version just took into account, decimals. This is on by default, you can just use the old calculation by adding set maxlen off to your rportmgr file.

Rport Comma

Rport has a new command called comma, where you can specify a fieldname or list of fieldnames to tell Rport to add commas to a given numeric field, when it is reported on and converted to ascii. The same rules will apply for a given fields subtotal and total field. This command can only be entered after the input command.

Rport Hide

Rport has a new command called Hide which when given a fieldname will remove that field from the report. The report will be adjusted accordingly and also the Hide command must be entered after the input command.

System Commands

If the user id isn't found or a shell name isn't returned the Open version of Suprtool suite of products will assume a shell of /bin/bash.

$Month($stddate())

The combination of nested $stddate within $month would handle months with 31 days incorrectly if the from date was a month with less that 31 days and the to month was a month with 31 days.

Rport

All versions of Suprtool, MPE, HP-UX (PA-Risc and Itanium), and Suprtool / Open have a new component called Rport, which is a simple report writer.

The code below represents one of the more complicated reports, and shows some of the attributes that you can control. Rport works off of self-describing files and is available on MPE, HP-UX (PA-Risc and Itanium) and on all Linux platforms that Suprtool/Open runs on.

The Title command divides the lines into quadrants, primarily, left, center and right and two lines are supported. The heading column option prints the Heading information specified in each column. You can also specify Heading Fieldnames, where Rport will divide the field name into each column, and you can do your own layout for each Heading Line:

```
head 1 Add " CharIntege     Double      Packed     Packed*                Quad"
head 1 Add "    IdLogica      DblLog Zoned"
head 2 add "Field Field      Field       Field       Field               Field"
head 2 add " Field Field       Field Field"
```

The Size command tells Rport the maximum size of the column for the data. By default Rport picks the maximum size that a field will be when converted to Ascii, as shown in the following table:

Field Format	Output Size
I1, J1	6 bytes
I2, J2	11 bytes
I3, J3	16 bytes
I4, J4	20 bytes
K1	5 bytes
K2	10 bytes

E2	12 bytes
E4	23 bytes
R2	12 bytes
Z*n*	*n*+1 bytes
P*n*	*n* bytes

The integer specified beside the fieldname is the length for the field to use for the report. If the data overflows the field will be filled with asterisks. The Subtotal and Total commands are pretty self-explanatory.

```
in file1.rporttst
title line 1 left RPNAME "SampReport"
title line 1 center RPCOMPANY "Robelle Solutions"
title line 1 right rppageno tag "Page No : "
title line 2 left rpdate tag "Date : "
title line 2 right rpruntime tag "Run Time : "
head 1 col "Char"
head 2 col "Field"
head 1 col "Integer"
head 2 col "Field"
head 1 col "Double"
head 2 col "Field"
head 1 col "Packed"
head 2 col "Field"
head 1 col "Packed*"
head 2 col "Field"
head 1 col "Quad"
head 2 col "Field"
head 1 col "Id"
head 2 col "Field"
head 1 col "Logical"
head 2 col "Field"
head 1 col "DblLog"
head 2 col "Field"
head 1 col "Zoned"
head 2 col "Field"
size int-field 8
size dbl-field 8
size packed-field 7
size packed*-field 7
size quad-field 7
size logical-field 7
size dbllog-field 7
size zoned-field 8
subtotal int-field dbl-field packed-field packed*-field quad-field id-field
subtotal logical-field dbllog-field zoned-field
total int-field dbl-field packed-field packed*-field quad-field id-field
total logical-field dbllog-field zoned-field
out REPORT01
xeq
```

The above code creates the following report. The report shows that it can handle most data types (the field names are the data types) and provides a simple layout and automatically aligns the subtotals and the total lines. The subtotaling works off the sorted field, (in this case char-field) and prints a subtotal line each time the char-field changes.

```
SampReport              Robelle Solutions              Page No : 1
```

```
Date : 20191024                                  Run Time : 12:45
 Char  Integer   Double  Packed Packed*   Quad   Id  Logical   DblLog   Zoned
Field   Field    Field   Field  Field    Field Field   Field     Field    Field
11111   11.11   111.11 +111.11 +111.11  111.11   1   11.11    111.11   111.11
     11.11  111.11 +111.11 +111.11  111.11   1   11.11    111.11   111.11
22222   22.22   222.22 +222.22 +222.22  222.22   1   22.22    222.22   222.22
22222   22.22   222.22 +222.22 +222.22  222.22   2   22.22    222.22   222.22
     44.44  444.44 +444.44 +444.44  444.44   3   44.44    444.44   444.44
33333   33.33   333.33 +333.33 +333.33  333.33   1   33.33    333.33   333.33
33333   33.33   333.33 +333.33 +333.33  333.33   2   33.33    333.33   333.33
33333   33.33   333.33 +333.33 +333.33  333.33   3   33.33    333.33   333.33
     99.99  999.99 +999.99 +999.99  999.99   6   99.99    999.99   999.99
44444   44.44   444.44 +444.44 +444.44  444.44   1   44.44    444.44   444.44
44444   44.44   444.44 +444.44 +444.44  444.44   2   44.44    444.44   444.44
44444   44.44   444.44 +444.44 +444.44  444.44   3   44.44    444.44   444.44
44444   44.44   444.44 +444.44 +444.44  444.44   4   44.44    444.44   444.44
    177.76 1777.76 +1777.76 +1777.76 1777.76  10  177.76   1777.76  1777.76
55555   55.55   555.55 +555.55 +555.55  555.55   1   55.55    555.55   555.55
55555   55.55   555.55 +555.55 +555.55  555.55   2   55.55    555.55   555.55
55555   55.55   555.55 +555.55 +555.55  555.55   3   55.55    555.55   555.55
55555   55.55   555.55 +555.55 +555.55  555.55   4   55.55    555.55   555.55
55555   55.55   555.55 +555.55 +555.55  555.55   5   55.55    555.55   555.55
     277.75 2777.75 +2777.75 +2777.75 2777.75  15  277.75   2777.75  2777.75
     611.05 6111.05 +6111.05 +6111.05 6111.05  35  611.05   6111.05  6111.05
```

$BOM and $EOM

Suprtool has two new subfunctions called $BOM and $EOM which are now available in the $stddate and $month functions. The $bom returns the beginning of the month date, "01" and the $eom, returns the end of the month for the given date.

For Example, the $stddate function will return the converted curdate, with either the beginning of the month for the day or the end of the month for the given curdate.

```
in file131
item curdate,date,ddmmyyyy
ext curdate
ext curbom=$stddate(curdate,$bom)
ext cureom=$stddate(curdate,$eom)
```

You can also use $stddate to convert a date that is already in ccyymmdd format to the beginning of the month or the end of the month respectively.

You can also use $bom and $eom as part of the $month function even if the date is not in ccyymmdd format.

```
Item curdate,date,ddmmyyyy
ext curdate
ext curbom=$month($stddate(curdate),+6,$bom)
ext cureom=$month($stddate(curdate),+6,$eom)
```

String Function Checks

Some of the String functions in Suprtool did not have proper length checks in place to insure that the an overflow wouldn't occur for specific functions. The length for most string fields to be checked is 4095 bytes. Suprtool will now throw a more generalized error of:

```
Error: Field in function is greater than 4095 bytes
```

If a function is longer than the 4095, the field is potentially corrupted and/or truncated.

Sdlinux for Linux

SDLinux is a utility that will convert the Self-Describing information, such that Suprtool/Open can read the file. Specifically it is meant to convert the .sd file from an HP-UX system to be read on Linux natively.

Sdlinux, will change the sd version field which tells Suprtool/Open the endianness of the sd file.

SD Version	Endianness	Extended Names
B.00.00	BIG (HP-UX) LITTLE(LINUX)	NO
B.00.01	BIG (HP-UX) LITTLE(LINUX)	YES
B.00.02	BIG	NO
B.00.03	BIG	YES

The sdlinux utility has four options, -f, -r, -h and –d. Only one option can be specified at any given time:

Option	SD Version	Integer Field
-f	B.00.00 becomes B.00.02 and B.00.01 becomes B.00.03	Any integer/logical field gets flagged as BE.
-r	B.00.02 becomes B.00.00 and B.00.03 becomes B.00.01	Any integer/ logical field gets the Endianness flag turned off.
-h	B.00.02 becomes B.00.00 and B.00.03 becomes B.00.01	No effect.
-d	No effect.	Any integer/logical field gets the Endianness flag turned off.

The sdlinux utility gets run with the above options and a filename for an argument as in:

```
./sdlinux '-ffilename'
./sdlinux '-rfilename'
./sdlinux '-hfilename'
./sdlinux '-dfilename'
```

The filename specified just needs to be the data file name not the sd filename. Regardless, sdlinux will figure out what to do if the .sd extension is in the filename argument.

Set Limits Override On | Off

Set Limits OverRide tells Suprtool for MPE to ignore any subsequent Set Limits TableSize command. This was added since the new MPE table size is now 2 GB and

works well without any limit. It was also added as Suprtool 4.4 and lower did not calculate the Set Limits Tablesize limit properly. So if a user had Set Limits Tablesize 25, Suprtool would allow for a table greater than that old limit, while the new version of the table would stop with "Table Full" as prescribed by the command.

Overriding the TableSize, allows for customers to set in a global suprmgr file and not have to search thru code to remove the specific command.

Enhancements in Version 6.1

$Length

Suprtool has a new function called $length which returns a double integer number that is the length of a given byte or numeric field. For byte type fields Suprtool starts at the right most byte position to look for any non-space character and calculates the length of the data. For numeric fields Suprtool will convert the number to display and starting on the left will search for any non-zero character to establish the length.

Below is an excerpt from the manual, that describes how it can be used:

$LENGTH (Works on Byte-type and numeric fields)

Purpose of the $length frunction is to return the length of the specified byte field as if the field has had the $rtrim function applied or the position of the rightmost non-space character.

If Usage:

```
if $length(NAME) >= 15
```

Extract Usage:

```
Define LenField,1,4,double
extract LenField=$length(byte-field)
```

$ROBELLE variable

The $ROBELLE variable on HP-UX and Open platforms has been expanded to 55 characters for the $ROBELLE directory name when you are installing outside of /opt/robelle. You use the $robelle variable to tell suprtool and all of the programs that come with suprtool where to find where it is installed.

Set EditSignNeutral

The $edit function converts data from numeric to byte and applies an edit mask to format the data. Neutral data is not considered to be signed so if you have neutral

data, a positive sign will not be applied to the edit mask. With Set EditSignNeutral turned on, Suprtool will consider the Neutral data to be positive and therefore print out the "+" or "DR", depending on the edit-mask used.

```
Set EditSignNeutral On
```

STEXPORT

STExport now supports 8196 bytes on the input, and the output file.

SUPRLINK

Suprlink now supports 8196 bytes on the input file, link file and 16384 bytes on the output file.

Enhancements in Version 6.0

$List NOSAMETO

The List command now has a NOSAMETO option to turn off the SAMETO feature.

BackwardChain

The Set command, Backwardchain when turned on will tell the Chain command to do a Backward Chained read.

$INRECNUM

The if / extract commands can now utilize a new function called $INRECNUM, which allows you to use the input record number in certain tasks. For example, the task below would find record number 11.

```
In somefile
If $inrecnum=11
```

You can also, utilize the $inrecnum function in the extract command:

```
In somefile
Def recnum,1,4,double
Ext recnum=$inrecnum
```

The $inrecnum function was designed to find records especially in the instance where duplicate records are in a particular data source and cannot be isolated by any other means.

$LEADZEROZ

The if / extract commands can now utilize a new function called $LEADZEROZ, which allows you to add leading zeroes to a specific display field. This was designed specifically for the extract command and fixing up data but can be used in the if command as well.

$NUMBER is capable of fixing up numbers, but the new $LEADZEROZ function is more lightweight and simply adds leading zeroes, and has an option to justify right as shown below. The source data looks like this:

```
/PRINT LEADZERO
1
 2
  3
   4
    5
     6
      7
       8
      9
10
12
12345
220
```

You can format with the following:

```
>IN LEADZERO.SUPRTEST
>DEF A,1,12,DISPLAY
>EXT A=$LEADZEROZ(A,J)
>out *
>xeq
000000000001
000000000002
000000000003
000000000004
000000000005
000000000006
000000000007
000000000008
000000000009
000000000010
000000000012
000000012345
000000000220
IN=13, OUT=13. CPU-Sec=1. Wall-Sec=1.
```

The $LEADZEROZ function cannot fix issues like commas and decimal places in a display field, this can be handled by the $number function.

$LEADZEROB

The if / extract commands can now utilize a new function called $LEADZEROB, which allows you to add leading zeroes to a specific byte field. This was designed specifically for the extract command and fixing up data but can be used in the if command as well. The data looks like this:

```
/PRINT LEADZERO
1
 2
  3
   4
    5
     6
      7
       8
      9
10
12
12345
220
```

You can clean it up with the following:

```
>IN LEADZERO.SUPRTEST
>DEF A,1,12,byte
>EXT A=$LEADZEROB(A,J)
>out *
>xeq
000000000001
000000000002
000000000003
000000000004
000000000005
000000000006
000000000007
000000000008
000000000009
000000000010
000000000012
000000012345
000000000220
IN=13, OUT=13. CPU-Sec=1. Wall-Sec=1.
```

$JUSTIFYL

The if / extract commands can now utilize a new function called $JUSTIFYL, which allows you to left justify text to the left side of a field.

```
>in leadzero
>def b,1,12,byte
>ext b=$justifyl(b)
>out *
>xeq
1
2
3
4
5
6
7
8
9
10
12
12345
220
IN=13, OUT=13. CPU-Sec=1. Wall-Sec=1.
```

$JUSTIFYR

The if / extract commands can now utilize a new function called $JUSTIFYR, which allows you to right justify text to the right side of a field.

```
>in leadzero.suprtest
>def a,1,12,byte
>ext a=$justifyr(a)
>out *
>xeq
           1
           2
           3
           4
           5
           6
           7
           8
           9
          10
          12
       12345
         220
IN=13, OUT=13. CPU-Sec=1. Wall-Sec=1.
```

$RESPACE

The if / extract commands can now utilize a new function called $RESPACE, which allows you to fixup byte data that has multiple spaces in between text. For example your data looks like this:

```
>in respace.suprtest
>def text,1,40
>ext text
>out *
>xeq
this  is  a  test
 this is a        test
this  is  a   test
this         is a test
this   is   a   test
this is a test
 please note this   is   a     test
this  is   a   test
this    is      a      test
this is a test
       this   is   a   test
 this is a test
this    is    a    test
this is a test
dummy   record
this is a  not   test
silly   record
IN=30, OUT=30. CPU-Sec=1. Wall-Sec=1.
```

It can easily be fixed up and converted to what is shown below:

```
>in respace.suprtest
>ext text=$respace(text,J)
>out *
>xeq
this is a test
this is a test
this is a test
this is a test
this is a test
this is a test
please note this is a test
this is a test
this is a test
this is a test
this is a test
this is a test
this is a test
this is a test
dummy record
this is a not test
silly record
IN=30, OUT=30. CPU-Sec=1. Wall-Sec=1.
```

SDLINUX UTILITY

Suprtool now has a utility called sdlinux, which will help convert an HP-UX self-describing file which will have Big Endian numbers, in the Self-Describing information and in the integers in the data file.

Sdlinux, is a small utility that will change the sd version field which tells Suprtool/Open the endianness of the sd file.

SD Version	Endianness	Extended Names
B.00.00	BIG (HP-UX) LITTLE(LINUX)	NO
B.00.01	BIG (HP-UX) LITTLE(LINUX)	YES
B.00.02	BIG	NO
B.00.03	BIG	YES

The sdlinux utility has four options, -f, -r, -h and –d. Only one option can be specified at any given time:

Option	SD Version	Integer Field
-f	B.00.00 becomes B.00.02 and B.00.01 becomes B.00.03	Any integer/logical field gets flagged as BE.
-r	B.00.02 becomes B.00.00 and B.00.03 becomes B.00.01	Any integer/ logical field gets the Endianness flag turned off.
-h	B.00.02 becomes B.00.00 and B.00.03 becomes B.00.01	No effect.

-d	No effect.	Any integer/logical field gets the Endianness flag turned off.

The sdlinux utility gets run with the above options and a filename for an argument as in:

```
./sdlinux '-ffilename'
./sdlinux '-rfilename'
./sdlinux '-hfilename'
./sdlinux '-dfilename'
```

The filename specified just needs to be the data file name not the sd filename. Regardless, sdlinux will figure out what to do if the .sd extension is in the filename argument.

Please see the following example how you can move data from HP-UX, to Linux natively without having to export to ascii. On HP-UX, you simply create a self-describing file as you normally would:

```
Base mydb
Get mydataset
Out outfile,link
Xeq
```

Then you can use sdlinux to make some quick changes to the Self-describing information, specifically the version in the header and the integers and logicals get updated with the big endian flag being turned on:

```
sdlinux 'foutfile'
```

You can then ftp the data file and the sd file over to your Linux box:

```
ftp linuxbox.robelle.com
user
password
put outfile outfile
put outfile.sd outfile.sd
quit
```

You can then reverse the effects of the –f option with the –r option:

```
sdlinux -routfile
```

Then on Linux you can read the sd file natively even though the sd information has big endian information and the data file can be read with bigendian integers and logicals.

```
>in outfile
>form
    File: outfile     (SD Version B.00.02)  Has linefeeds
       Entry:                     Offset
          CHAR-FIELD           X5      1
          INT-FIELD            I1      6                                BE
          DBL-FIELD            I2      8                                BE
          PACKED-FIELD         P12    12
          PACKED*-FIELD        P12    18
          QUAD-FIELD           I4     24                                BE
          ID-FIELD             I1     32                                BE
          LOGICAL-FIELD        K1     34                                BE
          DBLLOG-FIELD         K2     36                                BE
          ZONED-FIELD          Z5     40
          FILLER               X36    45
    Entry Length: 80  Blocking: 1
>num 3
>list
>xeq
>IN outfile (0) >OUT $NULL (0)
CHAR-FIELD      = 11111          INT-FIELD       = 1111
DBL-FIELD       = 11111          PACKED-FIELD    = +11111
PACKED*-FIELD   = +11111         QUAD-FIELD      = 11111
ID-FIELD        = 1              LOGICAL-FIELD   = 1111
DBLLOG-FIELD    = 11111          ZONED-FIELD     = 11111
FILLER          =
```

Bugs Fixed

Bugs Fixed In Suprtool 5.8.11

Add Command. The Add command would fail if a Table in an Oracle database accessible by a given username had more than 2.1 billion entries on Oracle 11 and higher.

Bugs Fixed In Suprtool 5.8.10

Banner. The opening banner at runtime would display the wrong day and day of month. This was only an issue in Suprtool 5.8.

Bugs Fixed In Suprtool 5.7.11

Set CleanChar. Set CleanChar to a single character, after a Set CleanChar "<null>" would not come into effect due to the special null flag not being reset. This was in both Suprtool and STExport.

Bugs Fixed In Suprtool 5.7

STExport/Suprlink. PA_RISC versions of Suprlink and STExport would fail when attempting to report an error on an OUTPUT file if the user specified a directory that didn't exist or the user didn't have write permission to.

Bugs Fixed In Suprtool 5.6.11

Output , Display Command. Suprtool for Itanium and Suprtool/Open would not properly convert negative, single and double integers when output/,display is invoked

Quad Integer Input Parsing. Suprtool/Open would not parse a negative number into a quad integer container.