
Suprtool 6.1 for HP e3000:

Change Notice

by Robelle Solutions Technology Inc.



Program and manual copyright © 1981-2019 Robelle Solutions Technology Inc.

Permission is granted to reprint this document (but not for profit), provided that copyright notice is given.

Qedit and Suprtool are trademarks of Robelle Solutions Technology Inc. Oracle is a trademark of Oracle Corporation, Redwood City, California, USA. Other product and company names mentioned herein may be the trademarks of their respective owners.



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

Table of Contents

Introducing Suprtool Version 6.1	5
Introduction	5
Highlights in Suprtool 6.1	5
Highlights in Suprtool 6.0	5
Highlights in Suprtool 5.9	6
Known Problems	6
Compatibility.....	6
CPU Serial Number and CPU Name (HPSUSAN & HPCPUNAME).....	7
Documentation	7
Installation	8
Overview	8
Ecometry Users	8
Installation Instructions	8
Installation Assistance.....	8
Enhancements in Version 6.1	9
Introduction	9
\$Length.....	9
\$LENGTH (Works on Byte-type and numeric fields).....	9
If Usage:.....	9
Extract Usage:.....	9
\$ROBELLE variable	9
Set EditSignNeutral.....	10
STEXPORT.....	10
SUPRLINK	10
Enhancements in Version 6.0	11
Introduction	11
\$List NoSameTo	11
BackwardChain	11
Extents	11
SET INITNUMEXTENT number.....	11
SET MAXNUMEXTENT number	12
SET SORTTEXTMEM ON	12
SET SORTMEMADDR number	12
\$INRECNUM.....	12
\$LEADZEROZ	13
\$LEADZEROB	14
You can clean it up with the following:	14

\$JUSTIFYL.....	14
\$JUSTIFYR.....	15
\$RESPACE	15
Enhancements in Version 5.9	17
\$Month	17
Excel Command	17
Json Output.....	19
Multiple Json Commands	20
Bugs Fixed	21
Bugs Fixed In Suprtool 5.7.11	21
Bugs Fixed In Suprtool 5.7.10	21

Introducing Suprtool Version 6.1

Introduction

Suprtool provides fast access to your data on the HP e3000. With Suprtool, you can perform many necessary DP functions easily, with just a few simple commands. At Robelle we are constantly working on your enhancement requests so that we can include them every year when we release an updated version of Suprtool.

Highlights in Suprtool 6.1

- Suprtool has a new function called \$length that will return a double integer value of the length of a byte and/or numeric field.
- Suprtool has a new option called Set EditSignNeutral which tells the \$edit function to treat Neutral zoned and packed fields to be positive and treated as such by the \$edit function.
- Suprlink now supports 8192 bytes on both the input file and the link file.
- Suprlink now supports 16384 bytes on the output file in Suprlink
- Stexport now supports 8192 bytes on both the input file and output files.

Highlights in Suprtool 6.0

- The List command now has a NOSAMETO option to turn off the SAMETO feature.
- Set Backwardchain On, will cause the Chain command to do a backward chained read.
- Suprtool has some new Set Commands such as, InitNumExtent, MaxNumExtent, SortExtMem and SortMemAddr, which can be used to reduce the number of Extents used in some files and internal sort files.
- STExport has a new set command called Set Excel Leadzero On which tells the Excel command to add leading zeroes to the fields specified in the Excel Preserve command.
- Suprtool has a new function available to the if/extract commands called \$inrecnum, which expects a double integer result.

- Suprtool has a new function called \$leadzeroz, which will add leading zeroes to a display field and will optionally justify the field.
- Suprtool has a four new string handling functions, specifically, \$justifyl, \$justifyr, \$leadzerob and \$respace.

Highlights in Suprtool 5.9

- A new \$month function in the if / extract option allows you to add or subtract a number of months from a given date.
- STExport has a new command called Excel command which allows you to format a field in a format that allows leading zeroes or spaces to be preserved.
- A new option called JSON will output SD data in Java Script Object Notation.
- Suprtools new banner would show the incorrect day and day of week in certain circumstances. This was cosmetic only.
- Suprtool previously may have found “bad” records in jumbo master datasets under certain conditions.
- Suprtool/Open now has the option to read MySQL databases. We are looking for feedback on where to take this feature.

Known Problems

Set Itemlock does not work in one particular case.

Compatibility

Suprtool/iX is compatible with MPE V, MPE XL, and MPE/iX, including MPE/iX 6.5, 7.0 and MPE/iX 7.5.

Suprtools default prefetch value has been changed from 2 to zero.

Ksam64 files were not assumed to be Self-describing like CM KSAM and KSAMXL files when it was the input file and had labels.

A warning will print if a non-regular SD file is loaded into a Table with the data option.

An error will print if an extract command attempts to reference data when loaded from table that in turn was loaded when using a non-regular SD file, such as KSAM/XL, KSAM64 and CM KSAM.

CPU Serial Number and CPU Name (HPSUSAN & HPCPUNAME)

Suprtool and its associated products run only on CPUs whose serial numbers have been encoded ("showvar hpsusan" on MPE/iX). They also may be encoded with your HPCPUNAME values. If it fails to run and you get an "invalid HPCPUNAME" or "invalid HPSUSAN" error message, contact Robelle for assistance.

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, or see the change notice.

You can download our manuals and change notices in various formats and you can order printed (hardcopy) manuals from our web site at:

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

Installation

Overview

We have two types of Installations, Tape and Download. We provide links to the Installation instructions on our web site for each type of Install.

Ecometry Users

Users of the Ecometry application need to use the installation instructions supplied by Ecometry Corporation. Those instructions have extra steps for updating the Ecometry code accounts.

You will find the Ecometry version of these instructions at <http://www.robelle.com/ecometry/>

Types of installations, tape, download combined qedit and suprtool tape.

Installation Instructions

If you received a production release tape from Robelle, then you can install using the instructions on this web page:

<http://www.robelle.com/support/install/tape/stprod.html>

If you downloaded the Suprtool production release from our web site then the instructions can be found here for the ftp download method:

<http://www.robelle.com/downloads/install-stprod-ftp.html>

and here for the WRQ download method:

<http://www.robelle.com/downloads/install-stprod-wrq.html>

Installation Assistance

If you have any questions about the upgrade process 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 289.480.1060 or you can e-mail your question to support@robelle.com.

Enhancements in Version 6.1

Introduction

Suprtool is constantly being updated with new features. The following section describes the new enhancements to Suprtool since Suprtool 6.0

\$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 function 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

Introduction

Suprtool is constantly being updated with new features. The following section describes the new enhancements to Suprtool since Suprtool 5.9.10

\$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.

Extents

Suprtool for MPE, now has features designed to help control and reduce the number of extents used in Output files and some internal files.

SET INITNUMEXTENT number

Set InitNumExtent, when specified with a number between 1 and 32 and will use that number on the creation of the Output file, in the Initial Extents parameter.

This can help with extent allocation depending on your systems disc space and fragmentation. Keep in mind if you set InitNumExtent to 1, and MaxNumExtent is also set to 1, all of the disc space required for the output file will attempt to be allocated at once.

When a value between 1..32 is specified for InitNumExtent, and InitExtents is turned on, the InitExtents flag will be turned off.

SET MAXNUMEXTENT number

Set MaxNumExtent Set MaxNumExtent, when specified with a number between 1 and 32, will use that number on the creation of the Output file.

This can help with extent allocation depending on your system disc space and fragmentation.

Keep in mind if you set InitNumExtent to 1, and MaxNumExtent is also set to 1, all of the disc space required for the output file will attempt to be allocated at once.

It is also important to note that if MaxNumExtent is set to any value between 1..32, and the InitExtents flag is on, the InitExtents flag will be turned off.

SET SORTEXTMEM ON

Set SortExtMem, when turned on reduces some of the space used for memory mapped file used for sorting. The algorithm for the amount of space used is the flimit of the file plus 10 million bytes.

You can increase the adder, by entering a number between 10 million and 2billion with the set SortMemAddr command.

Turning this option on may cause errors indicating that sort overflowed or hit a bounds condition. You can increase the amount of space again by using set SortMemAddr command.

SET SORTMEMADDR number

Set SortMemAddr accepts a number between 10 million and 2billion which will add space to a memory mapped file used during the sort process and is to be used with Set SortExtMem, when it is turned on and the default adder value is not enough to prevent a Bounds condition.

\$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.

describes the new enhancements to Suprtool since Suprtool 5.9

\$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
00000000220
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
00000000220
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 *
>xeg
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 *
>xeg
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
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.

```


Enhancements in Version 5.9

\$Month

The if / extract commands can now utilize a new function called \$Month, which will add a given number of months to a given date in the format of ccyyymmdd or yyyyymmdd.

For Example:

```
In somefile
Item mydate,date,ccyyymmdd
Def targetdate,1,4,double
Ext targetdate=$month(mydate,+4)
```

The above task will take the field mydate and add four months to it. Suprtool will check if the date is valid and adjust the date within reason. For example if the given month for mydate has 31 days and the day is 31, and the month mydate becomes when the date is added to has only 30 days. The date will be adjusted to have the 30th for the day.

Excel Command

The Excel command can be used to produce columns of data that when imported will preserve spaces or leading zeroes.

```
EXCEL PRESERVE <fieldname>
```

Example

STExport can generate columns that are imported into Excel in such a way that leading zeroes are preserved. While the format produced is not traditional CSV, the format will produce a field in the form:

```
= "00055555"
```

This form when imported into Excel will preserve the leading zeroes. In order to invoke this format the Excel command has very simple syntax:

```
$in fileexcel
$col fixed
$quote double
$zero leading
$excel preserve newchar int-field
$out *
$xeq
```

These simple commands will generate a file that will have the usually formatted fields as well as some fields formatted specifically for preserving spaces and leading zeroes in Excel.

The result of such an STExport task will look as follows:

```
= " 1111 ", = " 0111", 0000011111, +0000001111
= " 1111 ", = " 0222", 0000022222, +0000002222
```

Json Output

The JSON command specifies STExport to generate Json output. Use the JSON to produce Java Script Object Notation documents for either Internet or Intranet applications.

JSON

OBJECT "string"

ONEPERLINE

Example

STExport can generate JSON output with just a few commands.

```
$input file1sd
$JSON
$output myJSON
$xeq
```

These four simple commands will generate a file that can be read by various applications. The result of such an STExport task will look as follows:

```
[{"CHAR-FIELD":"11111","INT-FIELD":1111,"ZONED-FIELD":11111}]
```

Object

The Object option allows the JSON data to be wrapped in a specific Object description.

```
JSON Object "Json object"
```

Looks like this:

```
{"Json object":
[{"CHAR-FIELD":\ "11111",
 \ "INT-FIELD"\ :1111,
 \ "DBL-FIELD"\ :11111,
 \ "PACKED-FIELD"\ :+1111,
 \ "PACKED.-FIELD"\ :+11111,
 \ "QUAD-FIELD"\ :11111,
 \ "ID-FIELD"\ :1,
 \ "LOGICAL-FIELD"\ :111,
 \ "DBLLOG-FIELD"\ :11111,
 \ "ZONED-FIELD"\ :11111
}]
}
```

Note that the example of the Output has one field per line with data. Normally this would have to be specified via the command line but the data is shown this way simply due to space constraints.

OnePerLine

For files that have many fields you may want to consider using the OneLine option of the JSON command:

```
JSON OnePerLine
```

STExport will put each field and data on one line with the appropriate beginning and end notation.

```
[{"CHAR\ -FIELD": "11111",  
  "INT\ -FIELD": 1111,  
  "DBL\ -FIELD": 11111,  
  "PAC\KED\ -FIELD": +11111,  
  "PAC\KED\ .-FIELD": +11111,  
  "QUAD\ -FIELD": 11111,  
  "ID\ -FIELD": 1,  
  "LOG\ICAL\ -FIELD": 1111,  
  "DBL\LOG\ -FIELD": 11111,  
  "ZONED\ -FIELD": 11111  
}]
```

Multiple Json Commands

You can enter multiple JSON commands per task to set the JSON options you require.

```
$in file1sd  
$JSON Object "Json object"  
$JSON OnePerLine  
$out *  
$xeq
```

An example of the output generated by the above commands is as follows:

```
{"Json object":  
  [{"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  
  ]  
}
```

Bugs Fixed

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 5.7.10

Table, Data and Hold. Suprtool would lose track of Table data field information, on tables held with the hold option in subsequent tasks.