Suprtool 6.1 for AMXW:

Change Notice

by Robelle Solutions Technology Inc.

Contents

Bugs Fixed   22

Introducing Suprtool Version 6.1

Overview

Use Suprtool/AMXW to read, select, and sort data from Oracle, Allbase and Eloquence databases and data files with fixed-length records. Suprtool/AMXW is designed to be similar to Suprtool for MPE while providing necessary HP-UX features. It is designed to work with AMXW an MPE environment from Speedware that runs on other platforms.

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

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.

- The $ROBELLE variable has been expanded to 58 characters.

- 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

- Set AMXW Perfwrite on, would cause Suprtool to fail with fserr 40 when using output,append.

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

- Set SDEXTNAME on now prints a warning in Suprtool for AMXW, since it is not supported in that version.

- Suprtool for AMXW will test for both long fieldnames and short fieldnames when parsing the $lookup function.

- Set Backwardchain On, will cause the Chain command to do a backward chained read.

- 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.

- Suprtool for AMXW now has a limit of 512 fields.

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.

- 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.

- Suprtools new banner would show the incorrect day and day of week in certain circumstances. This was cosmetic only.

Known Problems

There are no known problems at this time. If you have any questions or concerns or feedback, please feel free to e-mail me at: neil@robelle.com

Itanium vs PA_RISC

There are two types of machines that HP has that run HP-UX, the older machines are based on the PA_RISC chip and the new machines that use the Itanium (and Itanium 2 chips). There are also program files that are specific to each chip. PA_RISC programs can run on both PA_RISC and Itanium machines, however, Itanium programs can only run on Itanium machines.

Any program that runs on Itanium must utilize libraries that match its own program type. Therefore if a PA_RISC program runs on Itanium and looks for a library, it must find/load the PA_RISC version of the libraries that it needs and conversely an Itanium program must load the libraries specific to its type.

How to Tell what Machine I am on?

There are many ways to tell what kind of machine you are on, here are a couple:

```
uname -m
getconf MACHINE_MODEL
```

In each case the Itanium machine will typically say ia64 somewhere in the result of the above two commands. PA_RISC machines typically say "9000" somewhere in the string.

What is the Program File Type?

Since there are two types of machines there are also at least two types of program files. Suprtool is primarily either PA_RISC 1.1 or ELF-32 / IA64 which is also known as Itanium. To determine the type of program file you can use the file command:

Itanium
```
file suprtool
suprtool:       ELF-32 executable object file - IA64
```

PA_RISC
```
file suprtool
suprtool:       PA-RISC1.1 executable dynamically linked -not stripped
```

The PA_RISC binary can run on both platforms but you need to have PA_RISC libraries for Eloquence or Oracle if you are using the Oracle module. An Itanium binary will not run on a PA_RISC machine and will fail with "cannot execute".

Of course we've made a provision to identify if you are running the ia version of Suprtool in the banner and verify command with the ia after the UX :
```
SUPRTOOL/UXia/Copyright Robelle Solutions Technology Inc. 1981-2009.
(Version 5.2 Internal)  MON, JUN  8, 2009, 11:26 AM  Type H for help.
```

PA_RISC Loading

Suprtool by default looks in the libraries in any of the directories named in SHLIB_PATH. For example to insure that Suprtool resolves the library loads you can set the SHLIB_PATH system wide in your /etc/profile file in the following manner:
```
 export SHLIB_PATH=/opt/eloquence6/lib/pa11_32:/opt/oracle/lib
```

If you do not have the SHLIB_PATH variable set to a value where the libraries can be found, Suprtool will then try to load from their default names locations. For Eloquence the default location is :/opt/eloquence6/lib/pa11_32.

The Oracle interface will first try to load libclntsh.so and then attempt to load libclntsh.sl from $ORACLE_HOME/lib32/libclntsh.sl.

Itanium Loading

You can change Suprtool to pay attention to LD_LIBRARY_PATH and use dlopen to load libraries, you can set 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. The PA_RISC version of Suprtool can use either type of loading, the default or the dlopen method invoked with ROBELLE_DYN_LOAD.

Compatibility

Suprtool for AMXW would use the file size stored in the SD label as opposed to the actual physical file size that AMXW uses.

```
::FILE FILE80;REC=-80,,F,ASCII
::supramxw
SUPRTOOL/AMXW/Copyright Robelle Solutions Technology Inc.1981-2007.
(Version 5.0)  TUE, JAN 08, 2008,  2:42 PM  Type H for help.
>I STFILE
>DEF ABCXYZ,1,11
>EXT ABCXYZ
>O FILE80,LINK
>X
IN=26, OUT=26. CPU-Sec=1. Wall-Sec=1.


>FO FILE80
File: FILE80     (SD Version B.00.00)  No linefeeds
Entry:                    Offset
    ABCXYZ               X11      1
Limit: 1023  EOF: 26  Entry Length: 11  Blocking: 1
>E
::LISTF FILE80,2


ACCOUNT=  AMXWQA     GROUP=  FDE


FILENAME  CODE  -----------LOGICAL RECORD----------- ----SPACE----
                SIZE  TYP       EOF      LIMIT R/B   SECTORS #X MX


FILE80    1084   82B  FA         26       1023   1        9  1  *
```

So what happens is if an output link file is created with a file equation making the file bigger, any subsequent files that are created from the resulting output file are actually made smaller as you can see below.

```
::supramxw
SUPRTOOL/AMXW/Copyright Robelle Solutions Technology Inc.1981-2007.
(Version 5.0)  TUE, JAN 08, 2008,  2:42 PM  Type H for help.
>i FILE80
>O FILE80A
>X
Warning:  Using Output FILE80A,Link
IN=26, OUT=26. CPU-Sec=1. Wall-Sec=1.
>E
::LISTF FILE80A,2
ACCOUNT=  AMXWQA     GROUP=  FDE



FILENAME  CODE  -----------LOGICAL RECORD----------- ----SPACE----
                SIZE  TYP       EOF      LIMIT R/B   SECTORS #X MX


FILE80A   1084   11B  FA         26       1023   1        2  1  *
```

 Suprtool for AMXW now honours the physical record size and the FILE80A file will now have a record size of 80.

The HP-UX and AMXW versions of AMXW no longer have Set varsubcompat on as the default. The default is now off and can be turned on with the set command or the –cv option.

Previously, Suprtool would treat an "R" type field in an Eloquence database as an R type while the data inside Eloquence is stored and treated as IEEE therefore incorrect results would occur with coercions and arithmetic operations. Suprtool now by default maps all Real and Long fields to their respective IEEE fields. You can change Suprtool back to the previous behaviour with Set RealMap Off.

A form command will still show the fields as being "R2" or "R4", but internally Suprtool will treat as IEEE, which is how Eloquence stores and treats the numbers.

Consequently STExport will now support items in SD files that are condidered R type, however it correctly will map them to IEEE.

Suprtool/UX 5.0 is now compatible with all versions of HP-UX 11.2x, as well as Oracle version 8 and higher.

Suprtool/UX version 4.8.02 and lower is compatible with HP-UX 9.0 and higher and Oracle 7.

Fastread option is now set to On for Suprtool for HP-UX the regular version. The Dynamic loading version of Suprtool has a default value of off to start.

Suprtool 5.2 is necessary for use with Oracle 11 and case sensitive passwords.

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 1-800-453-8970.

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.

You can download our manuals and Change Notices in various formats and order printed (hardcopy) manuals from our web site at:

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

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

There are typically two main types of installations. You can find the Suprtool for AMXW regular install instructions here:

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

You can find the Suprtool for AMXW build on your system instructions on the link below. You will need this install if you are going to require the KSAM option, Dynamic Image loading or Omnidex support, or intend to run on Itanium:

http://www.robelle.com/downloads/install-amxwbuildprod.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 page someone from 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/

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 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

Introduction

Suprtool is constantly being updated with new features. The following section describes the new enhancements to Suprtool since Suprtool 5.9

BackwardChain

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

List Command

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

$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
```

Which can be fixed in the following manner:

```
RUN SUPRTOOL.PUB.ROBELLE

SUPRTOOL/Copyright Robelle Solutions Technology Inc. 1981-2017.
(Version 6.0 Internal)  Mon, Feb 27, 2017,  3:48 PM  Type H for help.

>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.
```

Enhancements in Version 5.9

Introduction

Suprtool is constantly being updated with new features. The following section describes the new enhancements to Suprtool since Suprtool 5.8.10

$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 ccyymmdd or yyyymmdd.

For Example:

```
In somefile
Item mydate,date,ccyymmdd
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 filexcel
$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:

```
="     11111 ",=" 01111", 0000011111,+00000011111
="     11111 ",=" 02222", 0000022222,+00000022222
```

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
}]
}
```

MySQL Access

Suprtool/Open now has the ability to read MySQL databases via the Open and Select commands. We are looking for alpha, beta testers and feedback specifically on how to treat certain data items. Please e-mail Neil Armstrong at neil@robelle.com to request a trial of this software.

Bugs Fixed

Bugs Fixed In Suprtool 5.9.11

Perfwrite. Set AMXW Perfwrite On would cause Suprtool to fail when doing output somefile,append.

Bugs Fixed In Suprtool 5.9

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.