Latest Features in Suprtool

The latest and greatest features

Variable Substitution
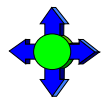
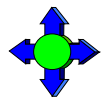$total

$counter

$Clean

For Techies

References

Development is a continuous process

We are always working on new features

New version every month or so

We constantly improve and work on our products. Suprtool is no exception. There are always new features being planned, designed, worked on and implemented. This section is to give you up-to-the-minute progress report on some of the latest features in Suprtool.

For Techies

References

Variable Substitution

MPE version has had Variable Substitution for two years

- Internal MPE routine
- Same as used by the CI

HP-UX now has the same feature

- Suprtool functions take precedence
- $stddate, $total, $date, $today are not replaced
- Variable must be set and exported prior to running program
- Suprtool command line, $read function
- STExport and Suprlink as well
- Must have Set Varsub On

For Techies

Suprtool, STExport and Suprlink on MPE have had Variable Substitution for over two years now. We have recently added Variable Substitution for the three programs on HP-UX. The feature is invoked in the same manner, >Set Varsub On. However, you cannot have an environment variable resolved if it has the same name as a Suprtool function, such as $date, $stddate etc.

References

$counter function

Sequential number function

Allows you to retain original output order

> get morder
> def mycount,1,4,double
> ext mycount=$counter
> ext orderno
> out myfile,link
> xeq

For Techies

For years, Suprtool has had the ability to output a record number to an output file with the num option of the output command:

```
>in mysdfile
>out myfile,num,data
```

The above could would generate an output file called myfile, however, you would lose the SD information and you can only put the number at the beginning or the end of the data. Suprtool now has a counter function that allows you to place a $counter anywhere in the output record as well as preserve the SD information.

```
>in mysdfile
>def mycount,1,4,double
>ext field1
>ext field2
>ext mycount=$counter
>out myfile,link
>xeq
```

References

The file myfile will be self-describing and contain the fields field1, field2 and mycount. The field mycount is defined as a double integer, since this is the only field type that the $counter function can use. Each record will have a unique ascending number starting at one.
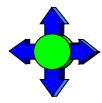
Clean my data

$clean function in Suprtool

- clean "^9"
- Set cleanchar " "
- update
- extract mybytefield=$clean(mybytefield)

The Clean command is used to tell Suprtool which characters it needs to look for in a given byte type field. For example:

clean "^9","^10","."

tells Suprtool to replace all occurrences of the tab character (Decimal 9), LineFeed (Decimal 10) and periods to whatever the Clean character is set to.

The Clean command takes both, decimal notation and the character itself. However, it is probably most convenient to use the Decimal notation for the characters that you wish to clean. The Decimal notation is indicated by the caret "^" character. By default, Suprtool replaces any of the characters specified in the clean command with a space. You can change the replacement character with the following set command:
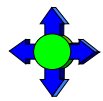
>set CleanChar "."

This command sets the replacement character to a period. You call the clean function the same way you normally use other functions available to If and Extract. For example:

>extract address1=$clean(address1)

For Techies

References

Clean Example

```
>base mydb,1,;
>get customer
>clean "^9","^10","^0","^7"
>set cleanchar " "
>update
>ext address(1) = $clean(address(1))
>ext address(2) = $clean(address(2))
>ext address(3) = $clean(address(3))
>xeq
```

	For Techies
The above task will look at the three instances of address and replace all occurrences of the tab, linefeed, null and bell characters with a space.	
	References