



# What's New At Robelle



## Adventures with Byte Order

By Neil Armstrong Software Architect

As you may be aware Suprtool has been ported to Linux and dubbed Suprtool/Open. A huge part of this process was re-writing some of the low-level number coercion code. The reason for this was that much of the different number formats and conversions were big endian based, from the original 3000 world. If you are uncertain what endianness means, here is a primer:

### Endianness

The endianness of a computer essentially means the byte order of the data stored has a certain Byte Order. For example from Wikipedia:

*In computing, memory commonly stores binary data by splitting it into 8 bit units (bytes). When a data word uses multiple such units, the order of storing these bytes into memory becomes important. The terms endian and endianness, refer to how bytes of a data word are ordered within memory.*

*Each byte of memory is associated with an index, called its address, which indicates its position. Bytes of a single data word (such as a 32 bit integer datatype) are generally stored in consecutive memory addresses (a 32 bit int needs 4 such locations).*

*Big-endian systems are systems in which the most significant byte of the word is stored in the smallest address given and the least significant byte is stored in the largest. In contrast, little endian systems are those in which the least significant byte is stored in the smallest address.*

The number 168,496,141 on a Big Endian machine like the HP 3000 would be hex \$0A0B0C0D and on a Little Endian machine (like the Intel X86) the double-integer would be \$0D0C0B0A. The byte order is reversed between the two machine types. Please note: the endian-ness applies to the order of the bytes, not to the order of the individual bits. Since Suprtool/MPE and parts of Suprtool for HP-UX assumed Big Endian for some of the numeric conversions, the low level numeric coercion library needed to be re-vamped for Linux.

### Getting Rid of Endian Dependency

During the port to Linux, done on an X86 machine running Red Hat Linux under VMWare, we mapped all the coercions that Suprtool did and wrote new individual routines to do those coercions with minimal little endian dependence. Once the low-level routines were re-written, the rest of the porting was standard fare.

Suprtool / Open

MF Cobol: Out of the box on Linux, MF Cobol expects Integers (even on a Little Endian platform) to be Big Endian. You can tell MF Cobol at the item level that a number is little endian by using comp-5 for the data type, however, we had a customer that didn't want to make that change to millions of lines of Cobol code. They have a reasonable goal: one set of source code for all platforms.

Another customer also indicated that they had similar problems with integers and were converting to comp-5.

#### Enter Suprtool/Open New Feature

So in order to accommodate our customer, we added a new feature to Suprtool, whereby Suprtool would convert output to Big Endian on output for integers (to accommodate MF Cobol) and also convert to Little Endian when reading in files. With this new option, Suprtool always sees little endian numbers internally, so that selection and sort are always accurate.

To turn this feature on, you can put SET ENDIANINT BE in your /opt/robelle/suprmgr file (or create one if it doesn't exist) and Suprtool will do the rest.

Here is an example run of Suprtool, converting a Self-Describing file from Little Endian to Big Endian integers. The Form command of the output file has a "BE" in column 70, but the output has been condensed for display purposes.

```
>in file1sd
>form
  File: file1sd      (SD Version B.00.00)  No linefeeds
  Entry:
    MYBYTE          X10      1
    MYINT           I1       11
    MYDBL           I2       13
    MYQUAD          I4       17
    MYDISP          Z12      25
    MYPACK          P28      37
    MYIEEEE4        E2       51
    MYIEEEE8        E4       55
  Entry Length: 62  Blocking: 1
>set endianint be
>out bige,link
>xeg
IN=15, OUT=15. CPU-Sec=1. Wall-Sec=1.
```

```
>form bige
  File: bige      (SD Version B.00.00)  No linefeeds
  Entry:
    MYBYTE          X10      1
    MYINT           I1       11 BE
    MYDBL           I2       13 BE
    MYQUAD          I4       17 BE
    MYDISP          Z12      25
    MYPACK          P28      37
    MYIEEEE4        E2       51
    MYIEEEE8        E4       55
  Entry Length: 62  Blocking: 1
```

Note the BE beside the Single-integer, Double-Integer and Quad-Integer indicating that those fields are Big Endian format. Suprtool, STExport and Suprlink will always convert those numbers so that Suprtool can consistently use the integers internally as Little Endian, so sorts and if comparisons will work properly.

```
>in file1sd
```

```
>total myint
>if myint=2222
>out $null
>xeq
Totals (WED, NOV 27, 2013, 6:43 AM):
```

```
MYINT 4444+
IN=15, OUT=2. CPU-Sec=1. Wall-Sec=1.
```

```
>in bige
>if myint=2222
>total myint
>out $null
>xeq
```

```
Totals (WED, NOV 27, 2013, 6:45 AM):
```

```
MYINT 4444+
IN=15, OUT=2. CPU-Sec=1. Wall-Sec=1.
```

## Suprtool2

Similarly, Suprtool2 has been enhanced to read and write out these Big Endian Integers. All that needs to happen is you set the following environment variable:

```
export ROBELLE_ENDIAN="BE"
```

and the Cobol interface integers will handled as Big Endian to allow for Cobol programs to not have to be changed to use the COMP-5.

Suprtool not only does it's normal coercions and record selections, but it is starting to handle the concept of endianness on x86 Linux platforms. We've laid the ground work for Suprtool to handle Big Endian and Little Endian numbers across platforms. If you have any interest in this feature or Suprtool in general on all platforms then contact Neil [neil@robelle.com](mailto:neil@robelle.com) or Tammy [tammy@robelle.com](mailto:tammy@robelle.com)

## Dog's life at Robelle

Ever since we had our office in a Langley farmhouse, dogs have been part of our office life. Our first office dog was a big German Shepard with bad hips (as is common with in-bred big dogs). He enjoyed hanging out with the staff or enjoying the sun on our office deck. Bob's office in Anguilla had Flash & Bobo. Flash was a stray island dog that decided to adopt Bob. Bobo was a friendly Doberman who spent a lot of time napping on the sofa in Bob's office.



Flash

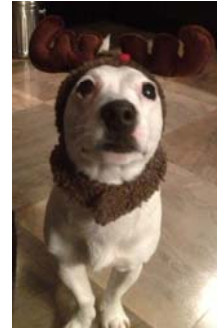


Bobo

Surrey office we had Cleo, a Dalmatian. She was the biggest mooch at coffee & lunch breaks. She always wanted to go to neighboring offices for a visit. Neil's office Milo was reigning king.



**Cleo**



**Milo**

Currently our office dogs are Cassie (Collie), Otis (King Shepard) & Parker (Malinois).



**Cassie**



**Otis**



**Parker**

**Robelle Team including Cassie, Otis & Parker would like to wish you a very Merry Christmas and happy prosperous New Year ...**

