Subject: Differences between IDL's floats and Java's floats - a problem
Posted by necr on Thu, 13 Nov 2003 12:09:04 GMT
View Forum Message <> Reply to Message

I am currently porting some IDL code across to java, and I've run into a couple of snags with the different ways IDL and Java deal with numbers. Most of these I have sorted, with trips to the APIs and a fair bit of googling. However, there is one I haven't managed to sort out yet.

IDL and Java appear to load floating point numbers from a file in very different ways.

In IDL, I can read seperate bytes into memory like so:

```
myvar=0b & readu, 10, myvar
```

(where 10 is the filehandle of the open file which is being read).
This will result in an unsigned byte being retrieved.

I could read the byte in with java using a DataInputStream like so:

```
byte a = input.readUnsignedByte();
```

(where input is my open DataInputStream).

Both these pieces of code would give the same result if run on the same byte in a file

To read in a float in idl, I would use:

```
readu, 10, floatvar
```

(the floatvar would not previously have been set to anything).

And in java:

```
float f = input.readFloat();
```

However, running these two pieces of code will not result in the same float being given out.

My Example

I read in four bytes from a file, both in IDL and Java. Both systems give me the results 0, 64, 206, 67.

If I run the float code on these same four bytes though, IDL will give me 412.50 (the value I want), while Java will give 5.951465E-39 - clearly not the number I'm looking for!

I have tested this on both a Sun and a Windows machine, and have received the same results.

So, has anyone got any ideas as to why this is happening? And more importantly, does anyone know what I can do to get the same float value being loaded in Java?

Thankyou in advance for any help you can give me.

Neil
