Subject: Converters and Value problem
Posted by mubeta on Thu, 22 Jul 2010 20:44:04 GMT
View Forum Message <> Reply to Message

Hi all,
I am using my own class for converting an integer number in a binary mode:

```
...
Value ConvertBinShort::Scan(const Value& text) const
{
 Value v = UPP::ScanInt((const char*)String(text), NULL, 2);
 if(IsError(v)) return v;
 if(IsNull(v)) return notnull ? NotNullError() : v;
 int m = v;
>>>>>> if(m > 0x7FFF) m = 0xFFFF - m;
 if(m >= minval && m <= maxval) return v;
 return ErrorValue(UPP::Format(t_("Number must be between %X and %X."), minval, maxval));
}
...
```

For every type of number: char, short, int and int64 I have a different class. The same problem for Octal ed Hexadecimal converters.

In fact, I don't found any solution for get the type of the number from Value. Value allow only 32 bit integres, or 64 bit integers. This result in an error when the Scan() method verfy the Min and Max limits.

Now, with my 4 different class: ConvertBinChar; ConvertBinShort, ConvertBinInt and ConvertBinInt64, (and other 4 more for Hex ... for Oct, etc), I am able to "rewind" the resulting number.
This all because Value do not allow char and short numbers, so I am not able to write an unique converter that automatically recognize the numbers limit in byte. (The main problem is when the ScanInt() result in a positive value, instead of negative, and I get an error of Min Max).

Question, (the same from two days ago, but maybe not so clear): can add to the Value also char and short numbers, in the user application?? (All the code will be more light and much more clear than now).

Thanks.