
Subject: Re: FIX: UPP::Scan returns int64 for INT_V
Posted by [kohait00](#) on Sun, 17 Apr 2011 12:57:42 GMT
[View Forum Message](#) <> [Reply to Message](#)

could you provide the link to the thread? couldnt find it..

i understand, that for best/secure performance ConvertInt should be able to keep the *parsed* data type as complete as possible (int64, if enough info provided from string), even if 'later' in usage the int64 is downgraded to int.

but then, the problem should be solved in EditInt, because, at most, there, one could expect to have a true int value returned. but it is not easy, since EditInt is a EditMinMax<> typedef.

i took a look in the code again. the int64 seems to be there to be able to perform range test after conversion, and return ErrorValue. there is also the solution to the problem. if no error, the range is ok and an int can be generated.

```
Value ConvertInt::Scan(const Value& text) const {  
    Value v = UPP::Scan(INT_V, text);  
    if(IsError(v)) return v;  
    if(IsNull(v)) return notnull ? NotNullError() : v;  
    int64 m = v;  
    if(m >= minval && m <= maxval) return int(m);//v; <=====  
    return ErrorValue(UPP::Format(t_("Number must be between %d and %d."), minval, maxval));  
}
```