

---

Subject: Re: operator<<= in tree control reference example

Posted by [gprentice](#) on Thu, 06 Jan 2011 10:53:10 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

cbpporter wrote on Wed, 05 January 2011 23:07Well <<= is pretty standard for U++, and comes often in pair with ~, which does the opposite. For controls it is SetValue/GetValue. It is just one of the conventions you need to learn and what contributes to the learning curve of U++. It is the same with standard C++ and << for streams and other conventions.

The reason "=" is not used because it would be confusing. You use "=" between different types, but only if they are near identical/easily convertible. Like different representation of the same data. But an EditField is not a Value. It has a Value. This is why we use <<=, or the "kind of assign operator, but more of s take this value and use it" operator.

Actually I can't even see how that operator<<= works.

edit is declared like this

```
Array<EditString> edit;
```

and the code does

```
edit.Top() <<= n;
```

where n is a String. So the LHS is EditString&. The only operator<<= I can find is in the Ctrl base class

```
const Value& operator<<=(const Value& v) { SetData(v); return v; }
```

but the call to SetData (not SetValue?) is not virtual and just throws the data away - yet the code works ??? I must be missing something?

---