

---

Subject: Re: U++ 2017 beta

Posted by [mirek](#) on Wed, 04 Jan 2017 14:47:47 GMT

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

---

cbpporter wrote on Wed, 04 January 2017 15:38mirek wrote on Wed, 04 January 2017 15:49cbpporter wrote on Wed, 04 January 2017 11:19

But I'll focus on strings first. Won't have probably time for anything more.

Here is a mockup of what I'm thinking:

Trouble with that is that it requires "special treatment" for String - debugger needs to be aware that you are at String.

Frankly, the ideal solution would be calling 'AsString' for values somehow, but that aint so easy unfortunately...

Yeah, that's exactly what I'm doing. Added special logic based on type name. Small price to pay I think for the added benefits. And not just String. A few Core classes should be handled like this too, like Vector, which has abysmal debugging. This isn't 1995 anymore, we can't have it like it is. Plus, the debugger should be aware of a few more other things, like Moveable. Remove all unnecessary information.

Think of the most fancy and elegant scripting language possible. Something people love to use and once they start to use it, become extremely loyal to it, like Python or Ruby. Now imagine how streamlined and useful debugging can look in those (there is support for full time run-time reflection and eval). That's what I want in U++. A String shouldn't have len, ptr, s, wptr, w, q and what not. What are even those? I know what they are, but I don't want to. Nobody does! The debug output should be readable for a String by a baby.

We can't do stuff like in the scripting languages, but we can hard code the debugger to pretty print a couple of hand picked types if in debug mode and if they have the proper layout.

Been there - that is what micio with MI2 debugger was trying. For me, it was never really usable. But maybe you will have a better luck :)