Subject: About Serialization
Posted by Tom1 on Thu, 01 Nov 2018 10:41:09 GMT
View Forum Message <> Reply to Message

Hi Mirek,

A quick question here: Is Serialization guaranteed to retain binary compatibility over the years to come and across platforms? I.e. is it safe to store data in files created by Serialize() so that they can be read from those same files in the coming years with programs created with future versions of U++?

Best regards,

Tom

Subject: Re: About Serialization
Posted by mirek on Fri, 02 Nov 2018 08:08:45 GMT
View Forum Message <> Reply to Message

Tom1 wrote on Thu, 01 November 2018 11:41Hi Mirek,

A quick question here: Is Serialization guaranteed to retain binary compatibility over the years to come and across platforms? I.e. is it safe to store data in files created by Serialize() so that they can be read from those same files in the coming years with programs created with future versions of U++?

Best regards,

Tom

Well, this one is interesting... Originally, it was not meant to be (like 15 years ago...), but now it definitely is. As long as you play right with Serialize (e.g. carefully version for the future), it is compatible across platforms and future proof.

Mirek

Subject: Re: About Serialization
Posted by Tom1 on Fri, 02 Nov 2018 09:32:35 GMT
View Forum Message <> Reply to Message

Thanks Mirek!

Sounds promising. Is the following the correct and complete documentation to follow?

https://www.ultimatepp.org/srcdoc$Core$Tutorial$en-us.html#S ection_2_3

Best regards,

Tom

Subject: Re: About Serialization
Posted by mirek on Fri, 02 Nov 2018 13:59:39 GMT
View Forum Message <> Reply to Message

Well, that is rather tutorial than docs, but yes, it gives the general idea.

Subject: Re: About Serialization
Posted by Tom1 on Fri, 14 Dec 2018 10:34:58 GMT
View Forum Message <> Reply to Message

Hi Mirek,

I would like to serialize 'Vector<Any> vec;', but Any does not have Serialize() available. How to proceed to accomplish this?

Best regards,

Tom

Subject: Re: About Serialization
Posted by mirek on Fri, 14 Dec 2018 11:58:06 GMT
View Forum Message <> Reply to Message

Tom1 wrote on Fri, 14 December 2018 11:34Hi Mirek,

I would like to serialize 'Vector<Any> vec;', but Any does not have Serialize() available. How to proceed to accomplish this?

Best regards,

Tom

That is basically impossible for really generic case...

If the set of classes stored in Any is fixed, you can do that testing this fixed set of types, encoding

the type when being stored and storing using types's Serialize, then on loading creating proper class based on encoded type and again Serialize it back.

Mirek

Subject: Re: About Serialization
Posted by Tom1 on Fri, 14 Dec 2018 12:53:35 GMT
View Forum Message <> Reply to Message

Hi,

This is exactly what I just did as a workaround while waiting for your input on the subject. (I thought there might be a nicer way.) Anyway, it works just fine!

Thanks and best regards,

Tom

Subject: Re: About Serialization
Posted by mirek on Fri, 14 Dec 2018 12:55:30 GMT
View Forum Message <> Reply to Message

Tom1 wrote on Fri, 14 December 2018 13:53Hi,

This is exactly what I just did as a workaround while waiting for your input on the subject. (I thought there might be a nicer way.) Anyway, it works just fine!

Thanks and best regards,

Tom

P.S.: Maybe you could rather put that data in Value? Value has full interfaces that allow to serialize any custom Value types..

Subject: Re: About Serialization
Posted by Tom1 on Fri, 14 Dec 2018 13:04:08 GMT
View Forum Message <> Reply to Message

That sounds interesting.. I need to dig a little deeper with that idea!

While Any has worked great for my mixed set of classes, it is a bit complex to code with those Is<>() / Get<>() interfaces. Value with my own classes might be a nicer way to deal with this.

Thanks,

Tom