

---

Subject: Re: ValueArray behaviour / inconsistantcy / BUG?

Posted by [mirek](#) on Thu, 21 Oct 2010 21:12:41 GMT

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

---

kohait00 wrote on Wed, 20 October 2010 04:45

now the question is, how is ValueArray supposed to be used:

is it supposed to be used as

- 1) Content of a RichValueRep? (Value.cpp:142: RichValue<ValueArray>::Register())
- 2) as a selfsustained Value interface/data container, just same as RawValue / RichValue?

as far as i got it to understand, ValueArray is kind of both. an extended Value implementation and meant to be in container data of a normal value, but offers the ValueArray::Data : Value::Void implementation, which is sort of Value domain.

now converting a ValueArray to a Value takes over the ValueArray::Data, serilizing it properly. deserializing it creates a RichValueRep<ValueArray>, which is inconsistent.

It was only a bug.

When using Value, you should not care about RichValueRep as long as you are not adapting another type to be Value compatible.

ValueArray is simply array of Values with O(1) copy, which is Value compatible. Nothing more, nothing less.

As far as that "ValueArray::Data" issue, it is simply optimization, which implements ValueArray without RichValueRep helper. The bug was that we were not careful enough and Serialization supposed that ValueArray Value compatibility IS done using RichValueRep.

---