IPAACA - Bug #493

IPAACA-CPP batch payload update does not work correctly when changing multiple values in a dictionary > level 1

2016-12-07 21:48 - Hendrik Buschmeier

Status:	Resolved		**Start date:**	2016-12-07
Priority:	Normal		**Due date:**	
Assignee:	Ramin Yaghoubzadeh		**% Done:**	100%
Category:			**Estimated time:**	0.00 hour
Target version:				

Description

The following batch payload update of a map {"test": {"1": "hello"}, {"2": "world"}, {"3": "!"}}.

```
ipaaca::IU::ptr iu = ipaaca::IU::create("test");
iu->payload()["hello"] = std::map<std:string, std::string>{{"1", "hello"}, {"2", "world"}, {"3", "!"}};
ob->add(iu);
{
    ipaaca::Locker locker(iu->payload());
    iu->payload()["test"]["1"] = "hallo";
    iu->payload()["test"]["2"] = "welt";
}
```

Results in the IU {"test" : {"2": "welt"}} instead of {"test": {"1": "hallo"}, {"2": "welt"}, {"3": "!"}}.

IPAACA-Python shows the correct behaviour:

```
import ipaaca
ob = ipaaca.OutputBuffer("test")
iu = ipaaca.IU("test")
iu.payload["test"] = {"1": "hello", "2": "world", "3": "!"}
ob.add(iu)
with iu.payload:
    iu.payload["test"]["2"] = "hallo"
    iu.payload["test"]["3"] = "welt"
```

A minimal working example is attached.

History

#1 - 2016-12-07 21:50 - Hendrik Buschmeier

As a workaround, the following can be used:

```
iu->payload()["test"] = std::map<std:string, std::string>{
    {"1", "hallo"},
    {"2", "welt"},
    {"3", iu->payload()["test"]["3"]} // <- needs to be re-included
};
```

#2 - 2016-12-07 23:25 - Ramin Yaghoubzadeh

- Status changed from New to Feedback

- % Done changed from 0 to 80

Considered fixed in 219bc8 (branch rsb0.14), tests pending.
(Windows binary package to follow suit soon.)

In locked mode, PayloadEntryProxies are now preferably generated from cached information, for the lock-owning thread only (other threads still only read form the last known globally known state).

ATTENTION: general note: non-Locked writes are by design not thread-safe. Make sure to use Locker frames around all code that might be called from more than one thread at the same time and write to the same IUs (e.g. IU event handlers that themselves update existing IUs)!

Hendrik Buschmeier wrote:

> The following batch payload update of a map {"test": {"1": "hello"}, {"2": "world"}, {"3": "!"}}.
>
> [...]
>
> Results in the IU {"test" : {"2": "welt"}} instead of {"test": {"1": "hallo"}, {"2": "welt"}, {"3": "!"}}.

#3 - 2016-12-08 01:35 - Ramin Yaghoubzadeh

> (Windows binary package to follow suit soon.)

For testing in Windows (and since the deps did not change) please simply replace your SOA_REPO_DIR\ipaaca-windows with the "ipaaca-windows" repo from the SCS server.

#4 - 2016-12-08 15:34 - Ramin Yaghoubzadeh

- Status changed from Feedback to Resolved

- % Done changed from 80 to 100

Initial tests / applications working. Considered RESOLVED until any contrary evidence might open another issue ;)

Files