Compliant Control Architecture - Bug #1638
cca.component logger[INFO] reports wrong state transition

09/25/2013 08:39 PM - C. Emmerich

Status:	Resolved		**Start date:**	09/25/2013
Priority:	Normal		**Due date:**	
Assignee:			**% Done:**	100%
Category:			**Estimated time:**	0.00 hour
Target version:	cca0.5			

Description

When configured for log-level INFO, the cca.component-logger reports state changes of the component. But the reported state changes are wrong. They are shifted by one state change request. See the following example:

```cpp
#include <cca/Component.h>

#include <rsc/logging/LoggerFactory.h>
#include <cca/processing/Strategies.h>

using namespace std;
using namespace rsc::logging;
using namespace cca;

class myCCA : public Component {
public:
   myCCA() : Component("myCCA") {
      setProcessingStrategy(Timed::samplerate());
   };

   void onProcess() {
      cout << "myCCA: my state: " << *state << endl;
   }
};

int main(int argc, char **argv) {

   LoggerFactory::getInstance().reconfigure(rsc::logging::Logger::LEVEL_INFO);

   myCCA mycca;
   cout << mycca.print() << endl;

   mycca.onProcess();
   mycca.requestTransition(ComponentState::LOADED());        // <-- logger reports <Execution>
   mycca.onProcess();
   mycca.requestTransition(ComponentState::EXECUTION());     // <-- logger reports <Loaded>
   mycca.onProcess();
   mycca.requestTransition(ComponentState::OFFLINELEARNING()); // <-- logger reports <Execution>
   mycca.onProcess();
   mycca.requestTransition(ComponentState::EXECUTION());     // <-- logger reports <OfflineLearning>
   mycca.onProcess();
   mycca.requestTransition(ComponentState::PERSISTED());     // <-- logger reports <Execution>
   mycca.onProcess();
```

```
      return 0;
    }
```

which results in:

```
    ...
    1380133236383 rsb.transport.spread.spreadconnection [INFO]: connected to spread daemon
     +---- myCCA ----------------------------------------------------------------+
     |  Component State:                                       |
     |     <Execution>                                    |
     |  Processing strategy:                                |
     |     <Timed, on every beat>                             |
     |  Beat Port:                                       |
     |     <std::string> <cfg:inprocess,spread,>(/cca/beat/)           |
     |  Component State Ports:                              |
     |     in:  <cca::ComponentState> <cfg:inprocess,spread,>(/cca/comp/myCCA/statechange/) |
     |     out: <cca::ComponentState> <cfg:inprocess,spread,>(/cca/comp/states/myCCA/) |
     |  No data input ports.                                |
     |  No data output ports.                               |
     +------------------------------------------------------------------------+
     |     ...                                       |
     +------------------------------------------------------------------------+

    myCCA: my state: <Execution>
    1380133236383 cca.component [INFO]: myCCA requestTransition() Successful transition into state <Execution>
    myCCA: my state: <Loaded>
    1380133236384 cca.component [INFO]: myCCA requestTransition() Successful transition into state <Loaded>
    myCCA: my state: <Execution>
    1380133236384 cca.component [INFO]: myCCA requestTransition() Successful transition into state <Execution>
    myCCA: my state: <OfflineLearning>
    1380133236384 cca.component [INFO]: myCCA requestTransition() Successful transition into state <OfflineLearning>
    myCCA: my state: <Execution>
    1380133236384 cca.component [INFO]: myCCA requestTransition() Successful transition into state <Execution>
    myCCA: my state: <Saved>
    ...
```

As you can see, the logger output is always wrong. E.g. for the first statechange: component initially is <Execution> as reported the user-printout and should be switched to <Loaded>. But the next successful transition is reported to <Execution>. And so on. Everything is shifted by one state change request. As a result, the last state change to <Saved> is never reported. The output should look like:

```
    ...
    myCCA: my state: <Execution>
    1380133236383 cca.component [INFO]: myCCA requestTransition() Successful transition into state <Loaded>
    myCCA: my state: <Loaded>
    1380133236384 cca.component [INFO]: myCCA requestTransition() Successful transition into state <Execution>
    myCCA: my state: <Execution>
    1380133236384 cca.component [INFO]: myCCA requestTransition() Successful transition into state <OfflineLearning>
    myCCA: my state: <OfflineLearning>
    1380133236384 cca.component [INFO]: myCCA requestTransition() Successful transition into state <Execution>
    myCCA: my state: <Execution>
```

1380133236384 cca.component [INFO]: myCCA requestTransition() Successful transition into state <Saved>
myCCA: my state: <Saved>
...

This only affects the logger reporting not the actual state change as the user-printouts show.

Hopefully, this example makes clear what I mean, as I found it difficult to explain (and debugging!!!) :D

Associated revisions

Revision 720 - 09/26/2013 11:24 AM - anordman

Fixes log messages on switching component states

   - Logger was reporting wrong state switches, although states were switched correctly

fixes #1638

History

#1 - 09/26/2013 11:17 AM - Anonymous
- *Status changed from New to Feedback*
- *Assignee set to Anonymous*

Which cca version is this?

#2 - 09/26/2013 11:24 AM - Anonymous
- *Status changed from Feedback to Resolved*
- *% Done changed from 0 to 100*

Applied in changeset r720.

#3 - 09/26/2013 11:26 AM - Anonymous
- *Target version set to cca0.5*

log message in both, cca 0.4 and trunk were affected, as long with a second (trace) log message.
Fixed in both versions, thanks for reporting.