

ステータス:	終了	開始日:	2011/06/29
優先度:	通常	期日:	
担当者:		進捗率:	100%
カテゴリ:		予定工数:	0.00時間
対象バージョン:			
説明			
株式会社 小田桐様からの報告			
onInitializeやonActivated、onExecuteなどが例外を投げたとき、登録しているPostComponentActionListenerが呼び出されません。onExecuteなどが戻り値でエラーを返した場合は、PostComponentActionListenerが呼び出されるため、例外を投げた場合でもPostComponentActionListenerは呼び出された方が良いと思います。			

関係しているリビジョン

リビジョン 2195 - 2011/07/01 00:53 - n-ando

Post-component-action callbacks were not called when component-actions throw exception. Now post-component-action callbacks are called with RTC::ERROR return value when exception is thrown. refs #2178

リビジョン 2195 - 2011/07/01 00:53 - n-ando

Post-component-action callbacks were not called when component-actions throw exception. Now post-component-action callbacks are called with RTC::ERROR return value when exception is thrown. refs #2178

リビジョン 2195 - 2011/07/01 00:53 - n-ando

Post-component-action callbacks were not called when component-actions throw exception. Now post-component-action callbacks are called with RTC::ERROR return value when exception is thrown. refs #2178

履歴

- #1 - 2011/06/29 22:13 - kurihara
- プロジェクト を OpenRTM-aist から OpenRTM-aist (C++) に変更
- #2 - 2011/07/01 00:53 - n-ando
- ステータス を 新規 から 終了 に変更
- 進捗率 を 0 から 100 に変更

onXXX()で例外が発生した場合も、戻り値RTC::ERRORとしてコールバックが呼ばれるように修正した。

```
--- RTOBJECT.cpp      (revision 2191)
+++ RTOBJECT.cpp      (working copy)
@@ -780,23 +780,23 @@
     {
         preOnInitialize(0);
         ret = onInitialize();
         postOnInitialize(0, ret);
         std::string active_set;
         active_set = m_properties.getProperty("configuration.active_config",
                                                                                                     "default");
         if (m_configsets.haveConfig(active_set.c_str()))
         {
             m_configsets.update(active_set.c_str());
         }
     }
```

```

-         else
-         {
-             m_configsets.update("default");
-         }
-     }
-     catch (...)
-     {
-         return RTC::RTC_ERROR;
+     ret = RTC::RTC_ERROR;
-     }
+     std::string active_set;
+     active_set = m_properties.getProperty("configuration.active_config",
+                                           "default");
+     if (m_configsets.haveConfig(active_set.c_str()))
+     {
+         m_configsets.update(active_set.c_str());
+     }
+     else
+     {
+         m_configsets.update("default");
+     }
+     postOnInitialize(0, ret);
+     return ret;
+ }

@@ -816,12 +816,12 @@
-     {
-         preOnFinalize(0);
-         ret = onFinalize();
-         postOnFinalize(0, ret);
-     }
-     catch (...)
-     {
-         return RTC::RTC_ERROR;
+     ret = RTC::RTC_ERROR;
-     }
+     postOnFinalize(0, ret);
+     return ret;
+ }

@@ -841,12 +841,12 @@
-     {
-         preOnStartup(ec_id);
-         ret = onStartup(ec_id);
-         postOnStartup(ec_id, ret);
-     }
-     catch (...)
-     {
-         return RTC::RTC_ERROR;
+     ret = RTC::RTC_ERROR;
-     }
+     postOnStartup(ec_id, ret);
+     return ret;
+ }

@@ -866,12 +866,12 @@
-     {
-         preOnShutdown(ec_id);
-         ret = onShutdown(ec_id);
-         postOnShutdown(ec_id, ret);
-     }
-     catch (...)
-     {
-         return RTC::RTC_ERROR;
+     ret = RTC::RTC_ERROR;
-     }
+     postOnShutdown(ec_id, ret);
+     return ret;
+ }

@@ -893,12 +893,12 @@
-     m_configsets.update();
-     ret = onActivated(ec_id);
-     m_portAdmin.activatePorts();
-     postOnActivated(ec_id, ret);

```

```

    }
    catch (...)
    {
-       return RTC::RTC_ERROR;
+       ret = RTC::RTC_ERROR;
    }
+   postOnActivated(ec_id, ret);
    return ret;
}

@@ -919,12 +919,12 @@
    preOnDeactivated(ec_id);
    m_portAdmin.deactivatePorts();
    ret = onDeactivated(ec_id);
-   postOnDeactivated(ec_id, ret);
    }
    catch (...)
    {
-       return RTC::RTC_ERROR;
+       ret = RTC::RTC_ERROR;
    }
+   postOnDeactivated(ec_id, ret);
    return ret;
}

@@ -944,12 +944,12 @@
    {
        preOnAborting(ec_id);
        ret = onAborting(ec_id);
-       postOnAborting(ec_id, ret);
    }
    catch (...)
    {
-       return RTC::RTC_ERROR;
+       ret = RTC::RTC_ERROR;
    }
+   postOnAborting(ec_id, ret);
    return ret;
}

@@ -969,13 +969,13 @@
    {
        preOnError(ec_id);
        ret = onError(ec_id);
-       m_configsets.update();
-       postOnError(ec_id, ret);
    }
    catch (...)
    {
-       return RTC::RTC_ERROR;
+       ret = RTC::RTC_ERROR;
    }
+   m_configsets.update();
+   postOnError(ec_id, ret);
    return ret;
}

@@ -995,12 +995,12 @@
    {
        preOnReset(ec_id);
        ret = onReset(ec_id);
-       postOnReset(ec_id, ret);
    }
    catch (...)
    {
-       return RTC::RTC_ERROR;
+       ret = RTC::RTC_ERROR;
    }
+   postOnReset(ec_id, ret);
    return ret;
}

@@ -1023,12 +1023,12 @@
    if (m_readAll) {readAll(); }
    ret = onExecute(ec_id);

```

```

        if (m_writeAll) { writeAll();
-         postOnExecute(ec_id, ret);
        }
        catch (...)
        {
-         return RTC::RTC_ERROR;
+         ret = RTC::RTC_ERROR;
        }
+         postOnExecute(ec_id, ret);
        return ret;
    }

@@ -1050,12 +1050,12 @@
        preOnStateUpdate(ec_id);
        ret = onStateUpdate(ec_id);
        m_configsets.update();
-         postOnStateUpdate(ec_id, ret);
        }
        catch (...)
        {
-         return RTC::RTC_ERROR;
+         ret = RTC::RTC_ERROR;
        }
+         postOnStateUpdate(ec_id, ret);
        return ret;
    }

@@ -1075,12 +1075,12 @@
    {
        preOnRateChanged(ec_id);
        ret = onRateChanged(ec_id);
-         postOnRateChanged(ec_id, ret);
        }
        catch (...)
        {
-         return RTC::RTC_ERROR;
+         ret = RTC::RTC_ERROR;
        }
+         postOnRateChanged(ec_id, ret);
        return ret;
    }

```