

OpenRTM-aist (C++) - バグ #839

PortBase::notify_connect 見直し

2009/07/31 14:56 - n-ando

ステータス:	終了	開始日:	2009/07/31
優先度:	通常	期日:	
担当者:	n-ando	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

PortBase::notify_connect() の仕様にはあいまいな点がある。
notify_connect()内で呼び出している、以下の3つの関数について処理が一貫していない。

- publishInterfaces() : エラー処理あり、エラーの場合リターン
- connectNext() : エラー処理あり、エラーの場合リターン
- subscribeInterfaces() : エラー処理なし、エラーを無視して続行

connect()およびnotify_connect()シーケンスとして、関連するPortのConnectorProfileの一貫性を保つ観点と、途中でエラーが出た場合には、connect()関数内できちんとdisconnect()されるため、notify_connect()のカスケード呼び出しは、エラーの無に関わらず完了させるべきである。

履歴

#1 - 2009/07/31 19:11 - 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

notify_connect() では、主な処理として

- publishInterfaces()
- connectNext()
- subscribeInterfaces()

を順に行うが、関連する全ポートのConnectorProfileの整合性を保つには、notify_connect()のカスケード呼び出しを完了する必要がある。したがって、これらの処理でエラーが発生しても3つの処理すべてを行ったうえでエラーコードを返すような処理にすべきである。以上の方針に従って、notify_connect()の処理を以下のように書き換えた。

```
ReturnCode_t PortBase::notify_connect(ConnectorProfile& connector_profile)
    throw (CORBA::SystemException)
{
-
    RTC_TRACE(("notify_connect()"));
+
    ReturnCode_t retval[] = {RTC::RTC_OK, RTC::RTC_OK, RTC::RTC_OK};
+
    // publish owned interface information to the ConnectorProfile
-   ReturnCode_t retval(RTC::RTC_OK);
-   retval = publishInterfaces(connector_profile);
-   if (retval != RTC::RTC_OK) return retval;
+   retval[0] = publishInterfaces(connector_profile);
+   if (retval[0] != RTC::RTC_OK)
+   {
+       RTC_ERROR(("publishInterfaces() in notify_connect() failed.));
+   }

    // call notify_connect() of the next Port
-   retval = connectNext(connector_profile);
-   if (retval != RTC::RTC_OK) return retval;
+   retval[1] = connectNext(connector_profile);
+   if (retval[1] != RTC::RTC_OK)
+   {
+       RTC_ERROR(("connectNext() in notify_connect() failed.));
+   }

    // subscribe interface from the ConnectorProfile's information
```

```

-     retval = subscribeInterfaces(connector_profile);
+     retval[2] = subscribeInterfaces(connector_profile);
+     if (retval[2] != RTC::RTC_OK)
+     {
+         RTC_ERROR(("subscribeInterfaces() in notify_connect() failed."));
+     }

    RTC_PARANOID(("%"d connectors are existing",
                  m_profile.connector_profiles.length()));

-     CORBA::Long index;
-     index = findConnProfileIndex(connector_profile.connector_id);
-
+     Guard gurad(m_profile_mutex);
+     CORBA::Long index(findConnProfileIndex(connector_profile.connector_id));
+     if (index < 0)
+     {
+         CORBA_SeqUtil::push_back(m_profile.connector_profiles,
@@ -215,7 +236,15 @@
+         m_profile.connector_profiles[index] = connector_profile;
+         RTC_PARANOID(("Existing connector_id. Updated."));
+     }
-     return retval;
+
+     for (int i(0), len(sizeof(retval)/sizeof(ReturnCode_t)); i < len; ++i)
+     {
+         if (retval[i] != RTC::RTC_OK)
+         {
+             return retval[i];
+         }
+     }
+     return RTC::RTC_OK;
+ }

```