

ステータス:	終了	開始日:	2009/03/02
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

症状

複合RTCに子RTCを追加(RTSystemEditorにて追加)した場合、元の複合RTCで公開していたポートが重複して公開されてしまう。

原因

PeriodicECSharedComposite.cppのupdateDelegatedPorts()の処理にて

```
set_difference(oldPorts.begin(), oldPorts.end(),
               newPorts.begin(), newPorts.end(),
               std::back_inserter(removedPorts));
set_difference(newPorts.begin(), newPorts.end(),
               oldPorts.begin(), oldPorts.end(),
               std::back_inserter(createdPorts));
```

という処理があります。

最初のset_difference()では、oldPortsの要素の内、newPortsに含まれない要素を抜き出しますので、例えば、oldPortの要素がAで、newPortsの要素がA,Bの場合、該当する要素が存在しないため、removedPortsには何も追加されません。

解決策

- 1. ポートを公開する場合
set_intersection()を使用する事で"重複する部分を抽出する"事が出来るようですので、removedPortsの抽出には、set_intersection()を使用するように変更すると良いかと思います。
- 2. 全てのポートを非公開とする場合(newPortsの要素が0の場合)
oldPortsの要素をremovedPortsとする。

newPortsの要素が0か否かに応じて上記1,2の処理を別ける必要があるかと思います。

一応、私のところで下記の変更を行いましたところ、複合RTCへの子RTCの追加でのポート重複問題は改善されました。

```
$ svn diff PeriodicECSharedComposite.cpp
Index: PeriodicECSharedComposite.cpp
=====
--- PeriodicECSharedComposite.cpp      (リビジョン 1230)
+++ PeriodicECSharedComposite.cpp      (作業コピー)
@@ -433,13 +433,19 @@
     std::vector<std::string> removedPorts; // oldPorts - interPorts
     std::vector<std::string> createdPorts; // newPorts - interPorts
```

```

-         set_difference(oldPorts.begin(), oldPorts.end(),
-                         newPorts.begin(), newPorts.end(),
-                         std::back_inserter(removedPorts));
-         set_difference(newPorts.begin(), newPorts.end(),
-                         oldPorts.begin(), oldPorts.end(),
-                         std::back_inserter(createdPorts));

+         if (newPorts.size() == 0) {
+             removedPorts = oldPorts;
+         }
+         else {
+             set_intersection(oldPorts.begin(), oldPorts.end(),
+                             newPorts.begin(), newPorts.end(),
+                             std::back_inserter(removedPorts));
+             set_difference(newPorts.begin(), newPorts.end(),
+                             oldPorts.begin(), oldPorts.end(),
+                             std::back_inserter(createdPorts));
+         }
+
+         RTC_VERBOSE(("old      ports: %s", ::coil::flatten(oldPorts).c_str()));
+         RTC_VERBOSE(("new      ports: %s", ::coil::flatten(newPorts).c_str()));
+         RTC_VERBOSE(("remove  ports: %s", ::coil::flatten(removedPorts).c_str()));

```

履歴

#1 - 2009/03/04 17:00 - n-ando

- ステータスを新規から終了に変更
- 担当者をn-andoにセット
- 進捗率を0から100に変更

updateDelegatedPorts()の処理自体は問題なし
addPorts()で以下の間違いがあった。

```

-         std::vector<std::string>::iterator pos =
-             std::find(portlist.begin(), portlist.end(), port_name);
+         if (pos == m_expPorts.end())
+             if (pos == portlist.end())
+             {

```

これを修正することで、configurationの変更によるポートの公開・非公開切り替えが正常に動作するようになった。