OpenRTM-aist (Python) - 機能 #3402

③　コンポーネント起動時に事前設定に従いポート同士を接続する機能（同一プロセス）

2015/12/22 09:43 - n-ando

ステータス:	終了	**開始日:**	2015/12/22
優先度:	通常	**期日:**	2016/03/25
担当者:	miyamoto	**進捗率:**	100%
カテゴリ:		**予定工数:**	30.00時間
対象バージョン:	RELEASE_1_2_0		

説明

コンポーネントコンフィギュレーションファイル(rtc.conf)に設定することで、起動時に同一プロセス内コンポーネントのポート同士を接続する機能を実装すること。

関係しているリビジョン

リビジョン 634 - 2016/02/01 13:25 - miyamoto

[incompat,new_func,->RELENG_1_2] add initPreConnection() to Manager. refs #3402

リビジョン 646 - 2016/02/01 19:35 - miyamoto

[incompat,new_func,->RELENG_1_2] add initPreConnection() to Manager. refs #3402

履歴

#1 - 2016/01/14 16:16 - miyamoto

- 期日 を 2016/03/25 にセット

- 担当者 を miyamoto にセット

- 対象バージョン を RELEASE_1_2_0 にセット

- 進捗率 を 0 から 90 に変更

- 予定工数 を 30.00時間 にセット

#2 - 2016/01/14 19:59 - miyamoto

- ファイル test_Manager_new.py を追加

実装

Manager.pyクラスにinitPreConnection関数を実装した。

rtc.confに以下のような形式で記述する事で自動的にポートが接続される。

manager.components.preconnect: RTC0.port0:RTC0.port1(interface_type=corba_cdr&dataport.dataflow_type=pull),RTC2.port2:RTC3.port3()

そのためinitPreConnection関数では以下の作業を行う事で接続するポート名、接続の設定を取得する。

","で区切ることで接続のリストを取得
connectors = str(self._config.getProperty("manager.components.preconnect")).split(",")
→
["RTC0.port0:RTC0.port1(interface_type=corba_cdr&dataport.dataflow_type=pull)", "RTC2.port2:RTC3.port3()"]

connectorsの各要素を"("で区切ることで、ポート名と接続の設定に分割
conn_prop = c.split("(")
→
["RTC0.port0:RTC0.port1","interface_type=corba_cdr&dataport.dataflow_type=pull)"]

conn_prop [1]から")"を削除
conn_prop[1] = conn_prop[1].replace(")","")
→
"interface_type=corba_cdr&dataport.dataflow_type=pull"

conn_prop [0]を":"で分割
comp_ports = conn_prop [0].split(":")
→
["RTC0.port0","RTC0.port1"]

comp_portsの各要素を"."で分割
comp0_name = comp_ports [0].split(".")[0]
→
["RTC0","port0"]

comp0_nameからコンポーネント、ポートオブジェクトを取得
comp0 = self.getComponent(comp0_name)
port0_var = OpenRTM_aist.CORBA_RTCUtil.get_port_by_name(comp0.getObjRef(), comp_ports [0])

conn_prop [1]を"&"で分割
opt_props = conn_prop [1].split("&")
→
["interface_type=corba_cdr","dataport.dataflow_type=pull"]

opt_propsの各要素を"="で分割
temp = o.split("=")
→
["interface_type","corba_cdr"]

プロパティに追加
prop.setProperty("dataport."+temp [0],temp [1])

ポートを接続
OpenRTM_aist.CORBA_RTCUtil.connect(c, prop, port0_var, port1_var)

そしてinitPreConnection関数をactivateManager関数内で呼び出すことでrtc.confに記述した接続が自動的に行われる。
ただし、manager.components.precreateに指定したRTCを生成後に呼び出す必要がある。

テスト

添付したテスト用コードを使用してテストを行った。
テスト用コードは以下の動作を行う。

setUp関数

マネージャ初期化の引数追加
self.dataPortConnectorName = "TestComp20.in:TestComp10.out(interface_type=direct)"
self.servicePortConnectorName = "TestComp10.service:TestComp20.service()"
sys.argv.extend(['-o', 'manager.components.preconnect:'+self.dataPortConnectorName+","+self.servicePortConnectorName])

マネージャ初期化
self.manager = OpenRTM_aist.Manager.init(sys.argv)
self.manager.setModuleInitProc(MyModuleInit)
self.manager.activateManager()

→MyModuleInitではTestComp10、TestComp20を起動する。どちらもインポートin、アウトポートout、サービスポートserviceを保持している

test_PreConnection関数

データポートが接続済みかを確認
ans = OpenRTM_aist.already_connected(inport, outport)
self.assertTrue(ans)

括弧内の設定が反映されたか確認
self.assertEqual(prop.getProperty("dataport.interface_type"),"direct")

サービスポートが接続済みかを確認
ans = OpenRTM_aist.already_connected(provided, required)
self.assertTrue(ans)

#3 - 2016/02/26 18:41 - miyamoto

- ファイル test_Manager_new.py を追加

#3404の実装のために、ポート名の間の記号を":"から"^"に変更した。

manager.components.preconnect:　RTC0.port0^RTC0.port1(interface_type=corba_cdr&dataport.dataflow_type=pull),RTC2.port2^RTC3.port3()

#4 - 2016/03/17 10:58 - miyamoto

- 進捗率 を 90 から 100 に変更

#5 - 2017/08/30 14:19 - n-ando

- ステータス を 新規 から 終了 に変更

ファイル

test_Manager_new.py	7.37 KB	2016/01/14	miyamoto
test_Manager_new.py	7.38 KB	2016/02/26	miyamoto