

OpenRTM-aist (Python) - バグ #3978

shutdownOnNoRtcs関数で終了する場合にエラーが表示される

2017/03/07 17:45 - n-miyamoto

ステータス:	終了	開始日:	2017/03/07
優先度:	通常	期日:	
担当者:	n-miyamoto	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			
shutdownOnNoRtcs関数で終了する際に以下のエラーが表示されるので修正する。			
Traceback (most recent call last): File "/usr/local/lib/python2.7/dist-packages/OpenRTM_aist/CorbaNaming.py", line 157, in isAlive if self._rootContext._non_existent(): File "/usr/local/lib/python2.7/dist-packages/omniORB/CORBA.py", line 561, in _non_existent return self._obj.nonExistent()			

履歴

#1 - 2017/03/07 17:52 - n-miyamoto

- ステータスを新規から解決に変更
- 進捗率を0から100に変更

発生箇所は、NamingManagerのネームサーバ確認、再登録処理の部分。

```
def update(self):  
    (省略)  
    if not self._names[i].ns.isAlive():
```

この部分がORBの終了後に呼び出されるとエラーが表示される。
そのため、Managerのshutdown関数内でタイマを先に終了するように変更した。

```
def shutdown(self):  
    self._rtcout.RTC_TRACE("Manager.shutdown()")  
    self._listeners.manager_preShutdown()  
    self.shutdownTimer()  
    self.shutdownComponents()  
    self.shutdownNaming()  
    self.shutdownORB()  
    self.shutdownManager()
```

ただ、shutdownOnNoRtcs関数はタイマのスレッドで呼び出しているため、タイマのスレッドの終了待ちはできず、ネームサーバ確認、再登録処理が実行されてしまう。

そのため、shutdownOnNoRtcs関数では新たにスレッドを生成してshutdown関数を呼び出すように変更した。

```
def shutdownOnNoRtcs(self):  
    (省略)  
    if len(comps) == 0:  
        self.shutdown_thread = threading.Thread(target=self.shutdown)  
        self.shutdown_thread.start()
```

これでエラーは表示されなくなりました。

#2 - 2017/03/19 08:13 - n-ando

- ステータスを解決から終了に変更

確認しました。ありがとうございました。