

OpenRTM-aist (Python) - 機能 #4031

manager_nameに指定が無い場合にmanager_<プロセス番号>でスレーブマネージャを起動する

2017/05/02 13:49 - n-miyamoto

ステータス:	終了	開始日:	2017/05/02
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:	RELEASE_1_2_0		
説明 create_component実行時にmanager_nameの指定が無く、実行中のスレーブマネージャでRTCが起動できない場合に、新たにmanager_<プロセス番号>というマネージャ名でスレーブマネージャを起動する。 スレーブマネージャでRTCが起動できない場合、現在はマスターマネージャ上でRTCを起動できるか試みるようになっているが、この機能は削除する。			
関連するチケット: 関連している OpenRTM-aist (Java) - 機能 #4187: manager_nameに指定が無い... 担当 2017/09/08			

関係しているリビジョン

リビジョン 825 - 2017/05/02 16:25 - miyamoto

[compat,->RELENG_1_2] refs #4031

履歴

#1 - 2017/05/02 17:14 - 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

マネージャ名に%pがある場合にプロセスIDと置き換えるように処理を変更しました。

```
def initManager(self, argv):
    (省略)
    self._config.setProperty("manager.instance_name",self.formatString(self._config.getProperty("manager.instance_name"),
```

self._confi

マネージャが起動したかの確認については、スレーブマネージャのインスタンス名がマスターマネージャ側で既知である必要があります。 subprocessモジュールの機能により起動したプロセスのIDは取得できるものの、これはバッチファイルを起動してその中でrtcdを実行しているため、rtcdのプロセスIDは取得できません。このため、あらかじめ登録済みのスレーブマネージャのインスタンス名一覧を保持しておき、新たにmanager_****という名前のマネージャが登録されるかでマネージャが起動したかを判定するようにしました。

```
        guard_slave = OpenRTM_aist.ScopedLock(self._slaveMutex)

        for slave in self._slaves[:]:
            try:
                prof = slave.get_configuration()
                prop = OpenRTM_aist.Properties()
                OpenRTM_aist.NVUtil.copyToProperties(prop, prof)
                name = prop.getProperty("manager.instance_name")

                if re.match(regex, name) and not (name in slaves_names):
                    mgobj = slave
            except:
                self._rtcout.RTC_ERROR("Unknown exception caught.")
                self._rtcout.RTC_DEBUG(OpenRTM_aist.Logger.print_exception())
                self.remove_slave_manager(slave)

        del guard_slave
```

#2 - 2017/08/30 14:20 - n-ando

- ステータスを解決から終了に変更

#3 - 2017/09/08 14:28 - katami

- 関連している 機能 #4187: manager_name に指定が無い場合にmanager_<プロセス番号>でスレーブマネージャを起動する を追加