

## OpenRTM-aist (Python) - バグ #3649

### onRateChangedコールバックが呼び出されない問題

2016/10/04 17:59 - n-miyamoto

ステータス:	終了	開始日:	2016/10/04
優先度:	通常	期日:	2016/10/31
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

Python版のver.1.1.1以降で実行周期を変更してもonRateChangedコールバックが呼び出されていない問題の調査、修正を行う。

補足：  
C++版についてもtrunkのソースコードはPython版と同じ実装なのでonRateChangedコールバックが呼び出されない問題が発生している。

#### 関係しているリビジョン

リビジョン 721 - 2016/10/13 15:17 - miyamoto

[compat,bugfix] refs #3649

#### 履歴

#1 - 2016/10/04 18:26 - n-miyamoto

- 担当者を n-miyamoto にセット

- 進捗率を 0 から 50 に変更

onRateChanged関数内にprint文を書いてRTCの実行周期を変更してみたのですが何も表示されませんでした。  
on\_rate\_changed関数がどこかで呼び出される必要があると思うのですが、調べた限りではどこでも呼び出されていません。  
以前のバージョンではPeriodicExecutionContextのsetRate関数内で呼び出しています。

```
def set_rate(self, rate):
    self._rtcout.RTC_TRACE("set_rate(%f)", rate)
    if rate > 0.0:
        self._profile.rate = rate
        self._usec = long(1000000/rate)
        if self._usec == 0:
            self._nowait = True

    for comp in self._comps:
        comp._sm.on_rate_changed()
    return RTC.RTC_OK
return RTC.BAD_PARAMETER
```

現在の実装ではsetRate関数内でExecutionContextBaseクラスのsetRate関数を呼び出している。

解決方法としては、まずExecutionContextWorkerクラスに以下のonRateChanged関数を追加する。

```
#
# @if jp
# @brief ExecutionContext の実行周期変更
#
# ExecutionContext の実行周期を変更した際にRTCのonRateChangedコールバックを呼び出す
#
# @return ReturnCode_t 型のリターンコード
#
# @else
# @brief
#
# @return The return code of ReturnCode_t type
#
# @endif
# RTC::ReturnCode_t stop(void);
def onRateChanged(self):
    self._rtcout.RTC_TRACE("onRateChanged()")
```

```
guard = OpenRTM_aist.ScopedLock(self._mutex)
ret_ = RTC.RTC_OK
for comp in self._comps:
    state = comp.onRateChanged()
    if state != RTC.RTC_OK:
        ret_ = state
del guard
return ret_
```

そしてExecutionContextBaseクラスのsetRate関数でonRateChanged関数を呼び出せばRTCのコールバックも呼び出される。

```
def setRate(self, rate):
    self._rtcout.RTC_TRACE("setRate(%f)", rate)
    ret_ = self._profile.setRate(self.onSettingRate(rate))
    if ret_ != RTC.RTC_OK:
        self._rtcout.RTC_ERROR("Setting execution rate failed", rate)
    return ret_
```

```
ret_ = self.onSetRate(rate)
if ret_ != RTC.RTC_OK:
    self._rtcout.RTC_ERROR("onSetRate(%f) failed", rate)
return ret_
```

```
ret_ = self._worker.onRateChanged()
if ret_ != RTC.RTC_OK:
    self._rtcout.RTC_ERROR("onSetRate(%f) failed", rate)
return ret_
```

```
self._rtcout.RTC_INFO("setRate(%f) don", rate)
return ret_
```

#2 - 2017/02/01 09:19 - n-miyamoto

- ステータスを新規から解決に変更
- 進捗率を50から100に変更

#3 - 2017/03/19 08:14 - n-ando

- ステータスを解決から終了に変更