

RTミドルウェア
SUMMER CAMP 2013

RTM on Androidの紹介



2013年 7月30日

株式会社セック
開発本部 第四開発部

川口 仁



RTM on Android 開発の狙い

RTC開発者／利用者の
裾野拡大！！

これに伴うRTミドルウェア
の普及促進！！



Android に RTM を載せるとは？

■ そもそもRTMとは

- RTCを作成するために
- RTCがRTCとして動作するために
- 存在するミドルウェア
- . . . としてのソフトウェア
- 実行単位はRTMではなくRTC

■ Android上でRTCを作成する？

- そのようなシーンは想像できない！

■ Android上で動作するRTCとは？

- 単なるAndroidアプリの1形態に過ぎない

Androidアプリとして
RTCを作成し

Android端末上で
上記RTCが動作する

ことを可能にすること

RTM on android
株式会社セック

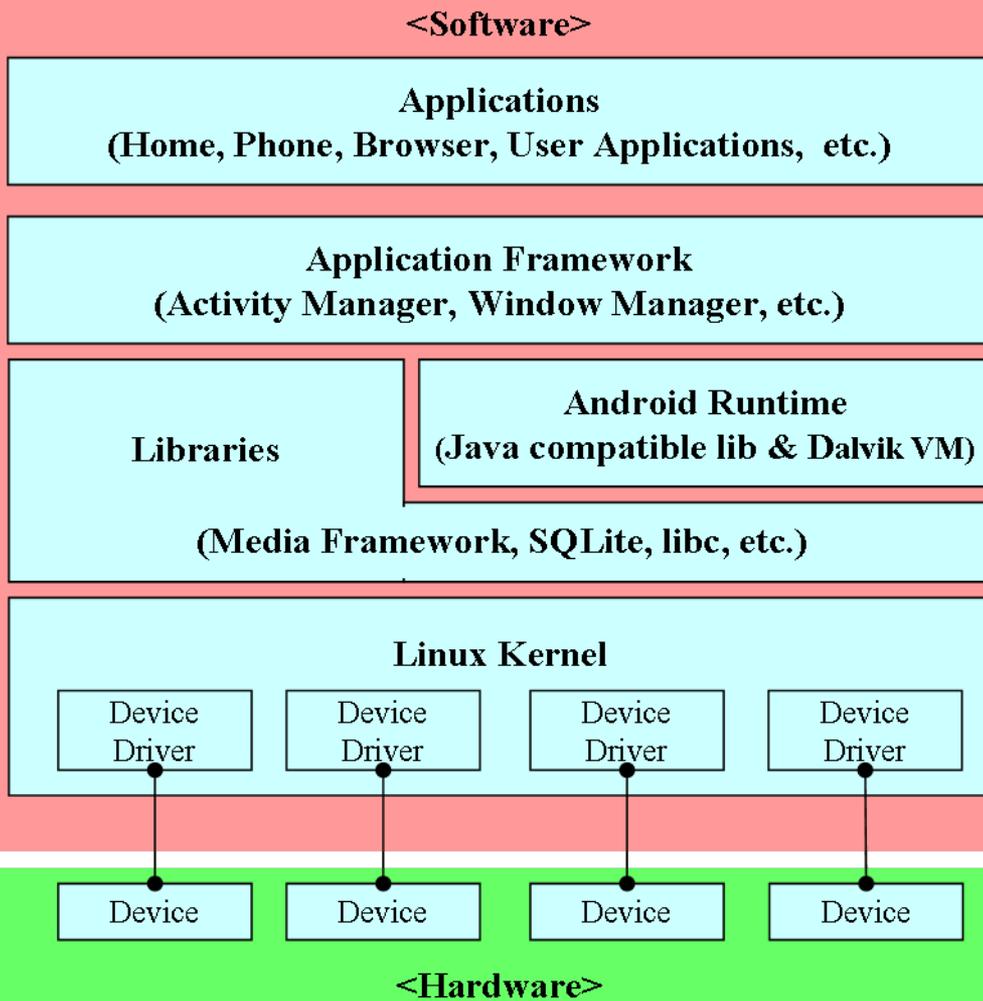


RTM on Android の効能

- RTミドルウェアやRTCの知識が乏しくても、通常のAndroidアプリケーション開発スキルを持つ人であれば、容易にRTCの開発が可能となる
- RTM on Androidを利用して開発したRTCは、ごく普通のAndroidアプリケーションとして扱うことができ、実行時のシステムリソースへの負荷も軽い
- RTM on Androidを利用して開発したRTCは、OMGにより標準化されたRTC標準仕様Ver1.0に従った軽量RTCのOpenRTM拡張モデルとしての基本的な振る舞いに対応し、他プラットフォーム上のRTCともシームレスに相互接続できる



Androidアーキテクチャとアプリケーション



■ Android Runtimeに専用VM (Dalvik VM)

- 各アプリケーションは、それぞれが独立したLinuxのプロセス
- 各アプリケーションに一つのDalvik VMインスタンスが対応して動作

■ 一般的Androidアプリケーションは、UIを持つコンポーネントであるActivityとして実装される

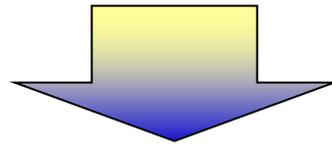
- Activityは画面の最上位に表示される場合にのみ動作

■ UIを持たず、バックグラウンドで継続的に動作するServiceも代表的なコンポーネント単位



RTCとAndroidにおけるコンポーネントの対応

- RTCは、必ずしもUIを必要としない
- 他RTCとの通信は継続的に実施できる必要がある



- **RTM on Android**では、RTCがAndroid上でServiceとして動作
 - UIが必要な場合は別途Activityを必要なだけ追加
 - これら全体で一つのアプリケーション単位 (apk)
 - 一つのAndroid端末上にて
複数のRTCを同時にActiveな状態で稼動可能



CORBA と OpenRTM の実装 . . .

■ RTM on Android は OpenRTMのAndroid実装



■ CORBA通信が必須

Androidプラットフォームでは未サポート

■ OpenRTM実装も必須

■ システムリソースへの負荷を 極力かけないように独自実装

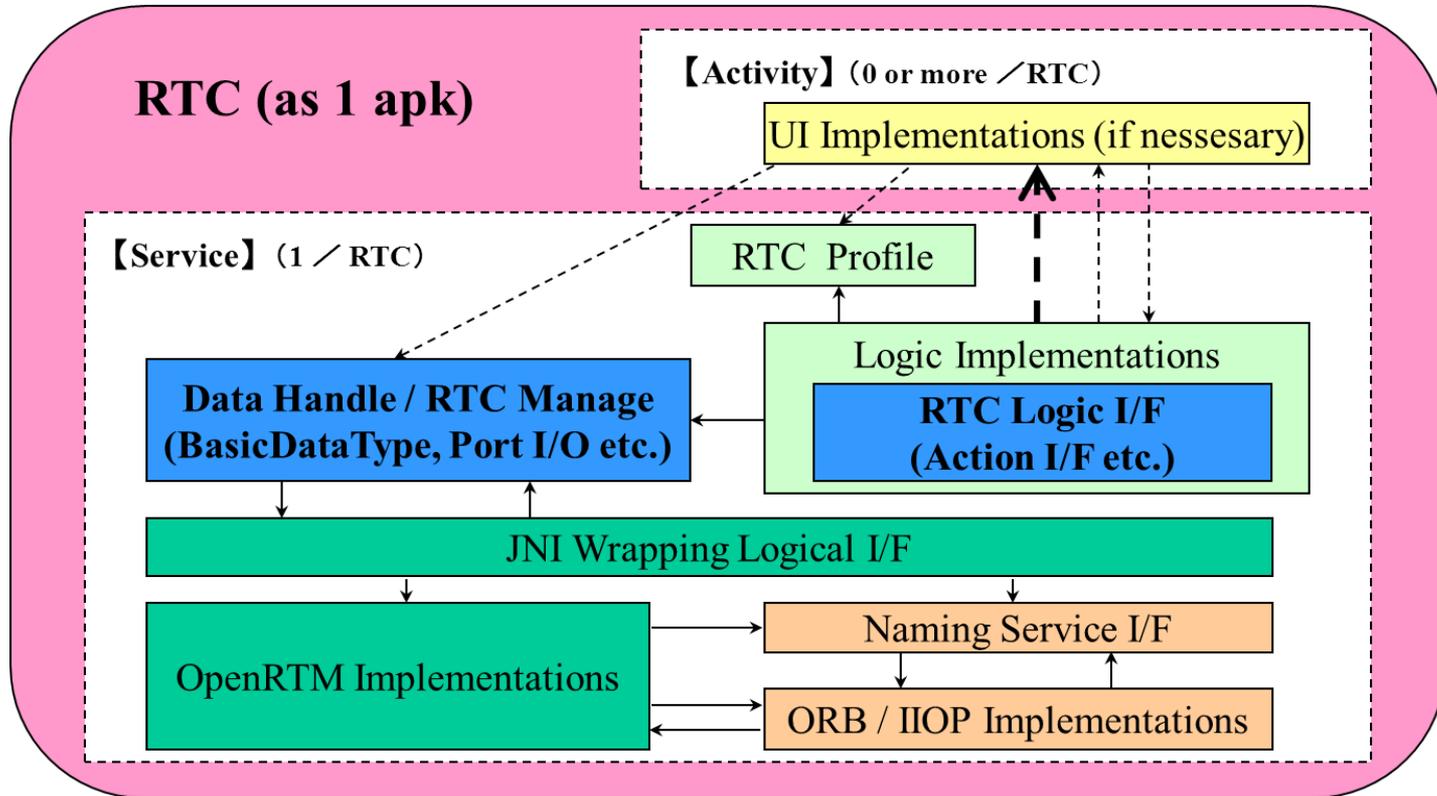


... ネイティブCライブラリとした

- 軽量CORBAのRtORBをAndroidにポーティング
 - RtORBは、独立行政法人産業技術総合研究所の原功主任研究員が開発を進めている
- OpenRTM実装は、IDLファイルからCソースを生成し、最低限のロジックを実装
 - OpenRTM-aistに含まれているOpenRTM.idlなどの産総研の成果を一部利用
- Google社により提供されているNDKを利用し、Cのネイティブライブラリである.so形式として生成／組み込み



RTM on Android のアーキテクチャ



- : Porting from RtORB(C)
 - : RTM Basement(C)
 - : RTM on Android(Java)
 - : RTC User Program from Template(Java)
 - : Android App I/F (Intent)
 - : function call / refer
- (Both dashed lines express arbitrary things.)



RTM on Android でのRTC開発

- **ごく普通のAndroidアプリ開発手法と同じ**
 - Google社により提供されているEclipse用SDKを利用
 - アプリケーション単位にAndroidプロジェクトを作成
 - ソースの編集からビルドまでを実施
 - RTCとしての動作はシンプルなAPI記述で実現可能

A screenshot of the Eclipse IDE. The left pane shows the project structure for 'SimpleIO', with 'MyRTC_Profile.java' selected under the 'src' directory. The right pane shows the source code of 'MyRTC_Profile.java', which defines static variables for profile information and a static map of profiles. The code includes fields for name, implementation ID, type, description, version, vendor, category, and execution rate, along with a map of profiles for different threads.

```
20 public static final String default_name_server = "192...
21 public static final String name = "SimpleInOut";
22 public static final String implementation_id = "Simple
23 public static final String type = "DataFlowComponent"
24 public static final String description = "Sample RTC c
25 public static final String version = "1.0";
26 public static final String vendor = "Systems Engineer
27 public static final String category = "Sample";
28
29 public static final float execute_rate = 10.0F; // 1
30 // PLEASE EDIT RIGHT COLUMN --- end ---
31
32 // DO NOT DELETE STATIC MAP DECLARATION BELOW
33 public static Map< String, String > profiles = new Has
34 put( RTCThread.NAME_SERVER, MyRTC_Profile.default_
35 put( RTCThread.NAME, MyRTC_Profile.name );
36 put( RTCThread.IMPLEMENTATION_ID, MyRTC_Profile.in
37 put( RTCThread.TYPE, MyRTC_Profile.type );
38 put( RTCThread.DESCRPTION, MyRTC_Profile.descript
39 put( RTCThread.VERSION, MyRTC_Profile.version );
40 put( RTCThread.VENDOR, MyRTC_Profile.vendor );
41 put( RTCThread.CATEGORY, MyRTC_Profile.category )
42 put( RTCThread.EXECUTION_RATE, String.valueOf( MyF
43
44
45
46
```



RTM on Android使用上の注意点

■ 画面オフや回転への注意

- Androidバージョン2系では、画面オフ時や回転時にonDestroy()やonCreate()が内部で呼ばれてしまう
- このため、RTCの起動・終了方法によっては、意に反して連動して終了してしまう危険性がある
- 対処方法
 - 画面レイアウト定義への指定にて画面オフ抑止
 - マニフェストへの指定にて回転を抑止

■ omni-ORBとの親和性への注意

- omni-ORBは、デフォルトで一定時間後にタイムアウト処理が走る
- RTM on Androidが採用しているRtORBはこれに未対応
- 対処方法
 - omni-ORBを利用するRTCのコンフィグレーション指定にて、タイムアウトを抑止

■ RT SystemEditorとの親和性への注意

- RTM on Androidを利用したRTCは、RT SystemEditor上での操作感が悪く、RT SystemEditorが無応答に陥ることもある
- 対処方法
 - Connectorを自作したり、rtshellを使う



デモ (ROBOCON Magazine連載で開発したアプリ)

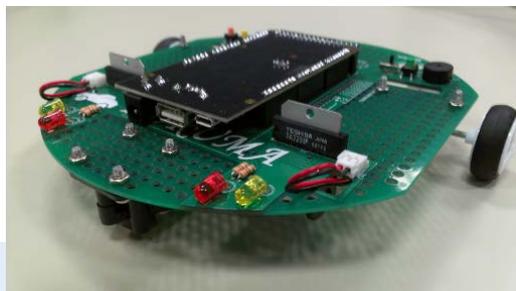


ROBOCON Magazine
(オーム社)
2012年7月号
～ 2013年1月号に連載

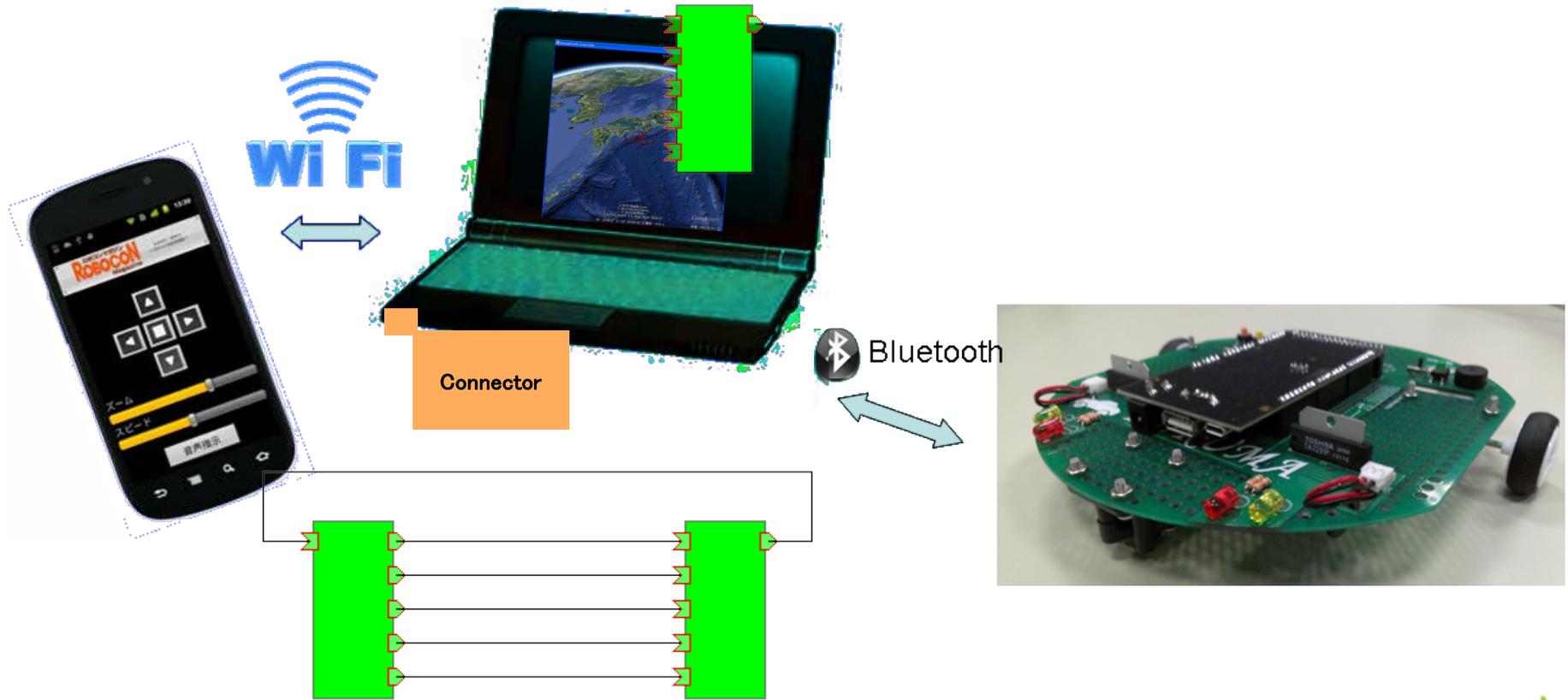
■ RTM.NETで実装したWindows PC上で動作するRTCをAndroid端末からコントロールするRTC

■ コントロール対象

- Google Earth内を移動するバーチャル飛行体
- Arduino用移動台車KURUMA Shield (ピルクス社)



デモシステム構成イメージ



KURUMA Shield コントロール時の接続





Realtime@net

セックはリアルタイム技術専門のソフトウェア会社です

