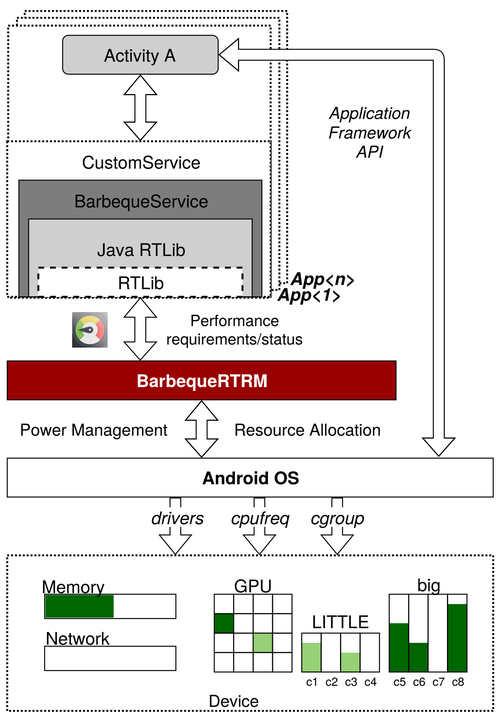
Application-level Run-Time Power/Performance Management for Android Devices

Michele Zanella, Giuseppe Massari, William Fornaciari

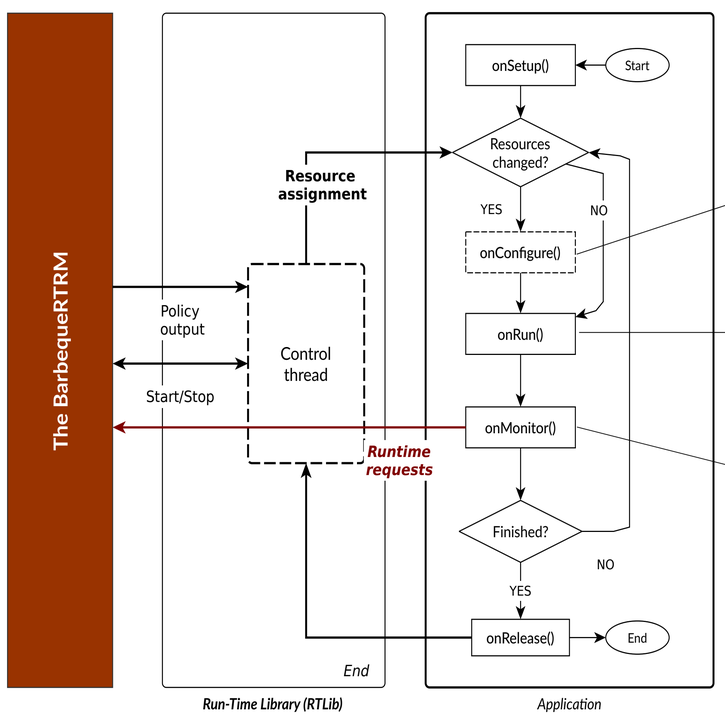
{name.surname}@polimi.it

The BarbequeRTRM and the Adaptive Execution Model (AEM)

Run-time resource management framework, supporting homogeneous and heterogenenous systems, spanning from High-Performance Computing (HPC) to embedded multi-core based devices [1]. Recent (unreleased yet) developments include the porting of the framework to Android, and the support of Fog/Edge computing platforms [2]. The development of the framework is also supported by several EU funded research projects [5] [6].



Porting the framework to Android:
- Java wrappers
- Linux control groups (cgroup) enabled on target device




```
@Override
protected void onConfigure(int awm_id)
   throws RTLibException {

   int freq, cpu_quota;
   // Frequency of the assigned core
   freq = getAssignedResources(
         RTLibResourceType.FREQ);
   // Amount of CPU quota
   cpu_quota = getAssignedResources(
         RTLibResourceType.PROC_ELEMENT);
   ...
}
```

```
@Override
protected void onRun()
   throws RTLibException {

   if (finished)
      throw new RTLibException(
         RTLibExitCode.RTLIB_EXC_WORKLOAD_NONE);

   /*  Core processing... */

   ...
}
```

```
@Override
protected void onMonitor()
   throws RTLibException {

   if (...) {
      // Setting upper bound power/performance
      RTLibConstraint constraint =
         new RTLibConstraint(
            awm_id,
            RTLibConstraintType.UPPER_BOUND,
            RTLibConstraintOperation.CONSTRAINT_ADD);
      setAWMConstraints(constraint);
   }
}
```

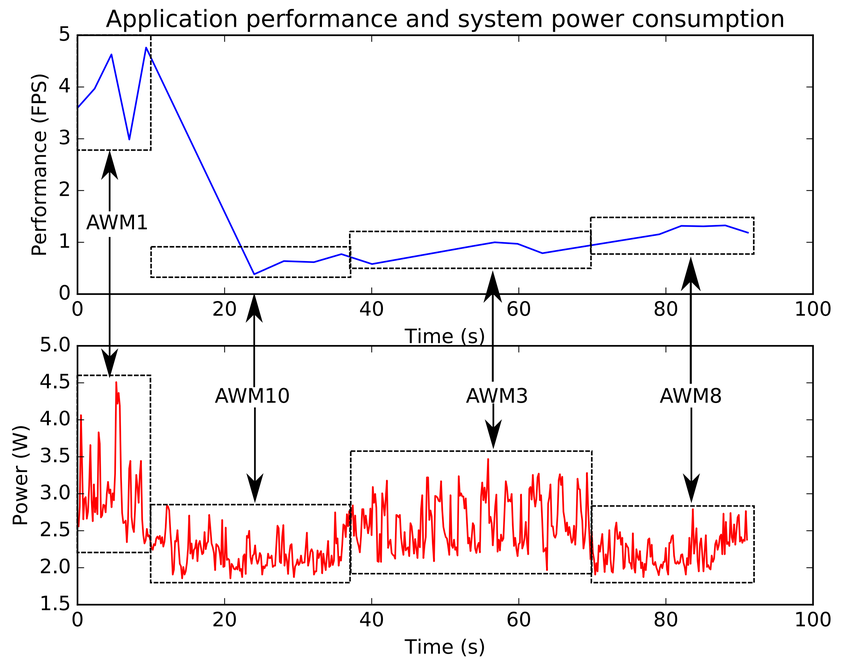
Android-inspired application execution model based on the callback methods. The application execution is synchronized with the BarbequeRTRM resource allocation process. The application is aware of the resources allocated and can negotiate its Application Working Mode (AWM), or explicitly ask for running in a lower power consumption mode (reduced assigned amount of CPU and other resources).

Experimental execution of an AEM-integrated benchmark

Profiling of the application with construction of the set of Application Working Modes ("RECIPE") with respect to the target device (*Hikey 960*).
For each AWM, the amount of CPU, the frequency setting, the average system power consumption and the performance estimation are reported.
Since the benchmark is single core, all the AWMs are related to a single core profiling activities.
In detail, AWMs 1-5 are related to a big core, while AWMs 6-12 are related to the little one.
From AWMs 1 to 10, application is forced into a specific core by setting a specific cpuset cgroup.
In AWMs 11 and 12 only the CPU quota is reduced through the cgroup.

AWM	Cluster	Frequency(Hz)	CPU Quota(%)	Power(W)	Performance(f/sec)
1	big	2362000	100	3.0926	2.7996
2	big	2112000	100	2.7884	2.5008
3	big	1805000	100	2.5849	2.1320
4	big	1421000	100	2.5322	1.7442
5	big	903000	100	2.2458	1.1310
6	LITTLE	1844000	100	2.5831	1.3713
7	LITTLE	1709000	100	2.5203	1.2126
8	LITTLE	1402000	100	2.0705	0.8635
9	LITTLE	999000	100	2.2378	0.6527
10	LITTLE	533000	100	2.2142	0.3122
11	LITTLE	533000	50	2.2508	1.02125
12	LITTLE	533000	25	2.2180	1.0913

The application varies its requirements at runtime. Thanks to the AEM and the runtime managed execution it can cooperatively reduce the system power consumption, by notifying the performance level needed, given application-specific evaluations. This allows us to go beyond a single system-wide "low-power mode". The Figure below shows an experimental scenario using the *Image Effect* benchmark of the *MobileXPRT2015* suite. It highlightes the application requests for different AWMs during the execution and related changes in performance/power consumptions level.



References

[1] P. Bellasi, G. Massari, and W. Fornaciari. Effective runtime resource management using linux control groups with the BarbequeRTRM Framework. ACM Trans. Embed. Comput. Syst, 2015
[2] M. Zanella, G. Massari, and W. Fornaciari. Enabling Run-Time Managed Distributed Mobile Computing. In proceedings of the 9th Workshop and 7th Workshop on Parallel Programming and RunTime Maangement Techniques and Design Tools and Architectures for Multicore Embedded Computing Platforms, 2018.
[3] BarbequeRTRM Open-Source Project: https://bosp.deib.polimi.it
[4] HEAPLab Politecnico di Milano: www.heaplab.deib.polimi.it
[5] H2020 MANGO: www.mango-project.eu
[6] H2020 RECIPE: www.recipe-project.eu