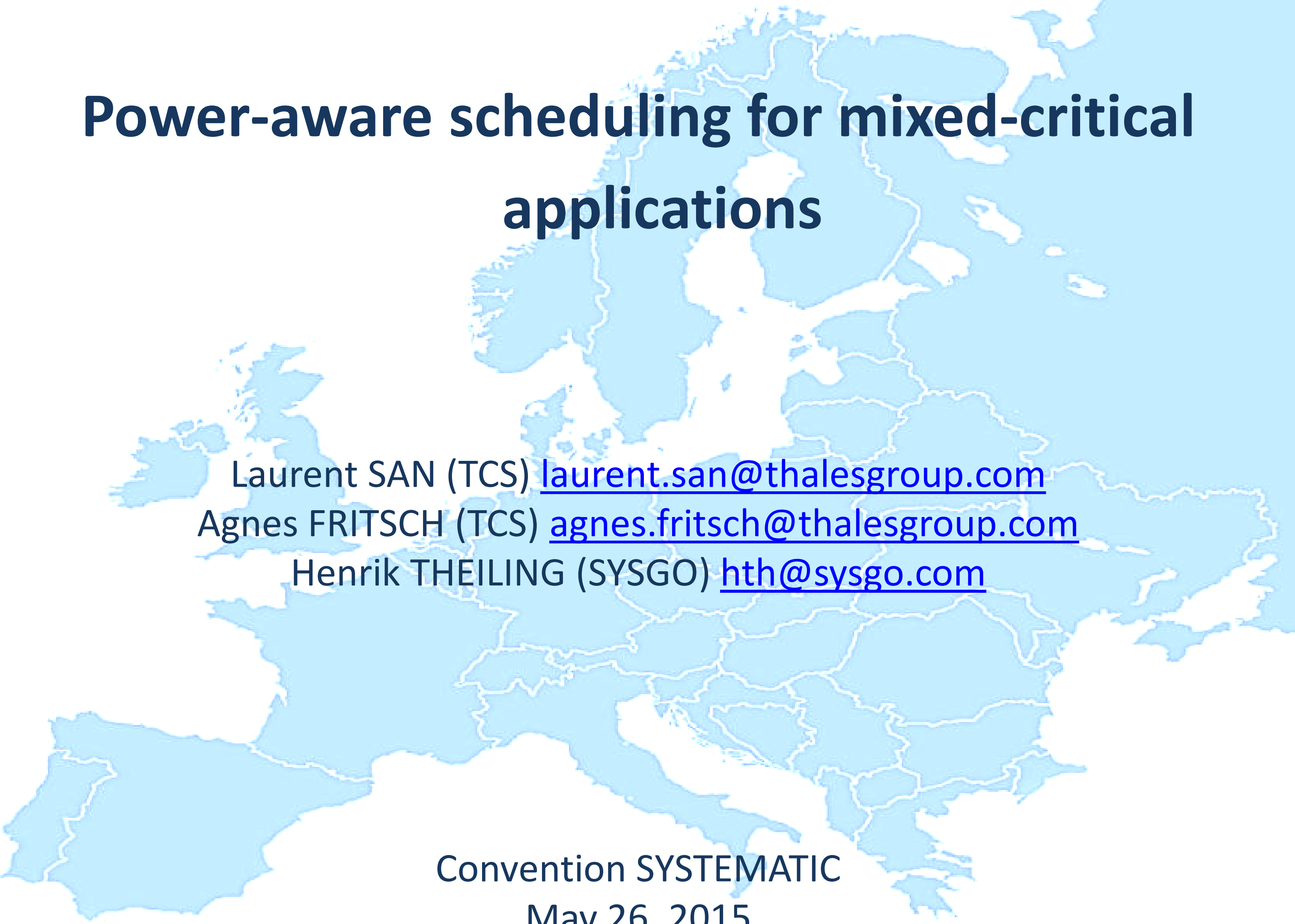
Power-aware scheduling for mixed-critical applications



Laurent SAN (TCS) laurent.san@thalesgroup.com
Agnes FRITSCH (TCS) agnes.fritsch@thalesgroup.com
Henrik THEILING (SYSGO) hth@sysgo.com

Convention SYSTEMATIC
May 26, 2015
Paris, Palais Brongniart

Introduction – TCS motivations

Examples of TCS products requesting the management of mixed-criticality

- An Equipment pointing the antenna to the right direction should not lose the satellite position
- A radio handheld device with advanced applications that should always give the priority to the voice transmission

- All the applications running on the same multicore chip
  - □ Processing SW: non-critical SW
  - □ Control Antenna/voice : critical SW

A COTS multicore platform

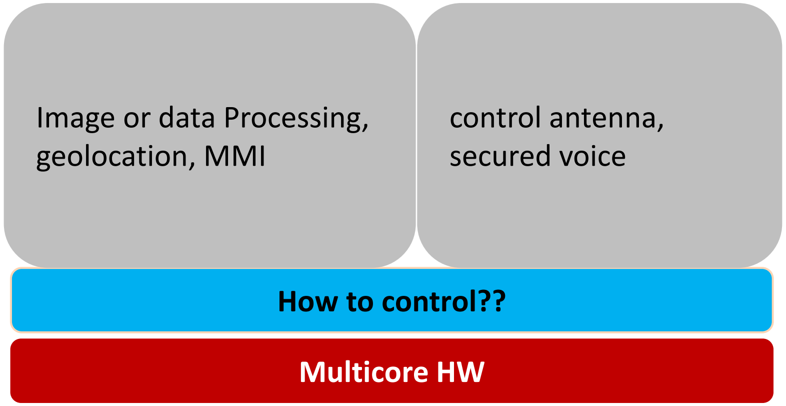


Image or data Processing, geolocation, MMI

control antenna, secured voice

How to control??

Multicore HW

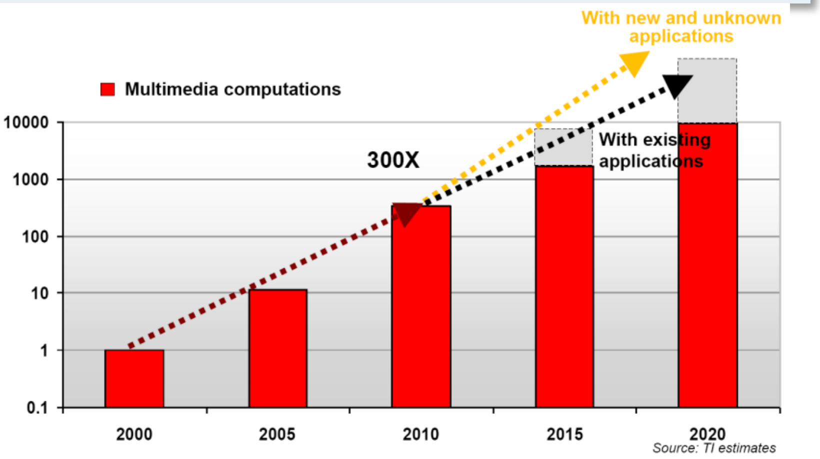
Introduction – TCS motivations

Context & Trends: multi/manycores, DVFS based methods

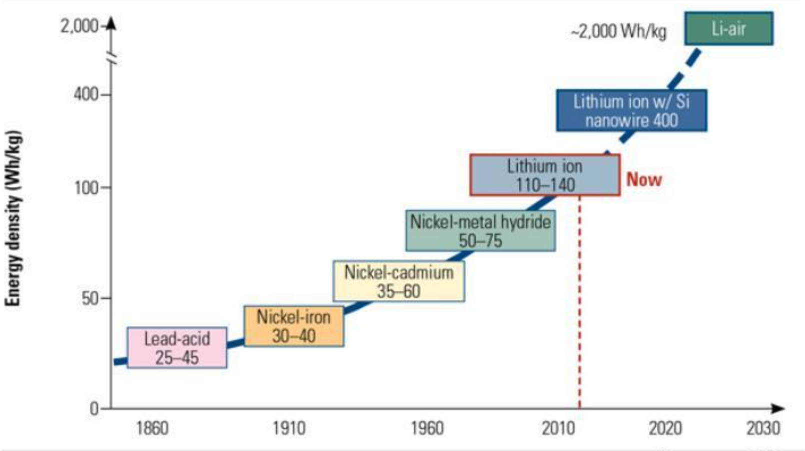
- Ever increasing functionalities & complexity
  - ☐ More performances required!
  - ☐ New SoC architectures are emerging:

  Multicores, GP-GPU, Manycores …

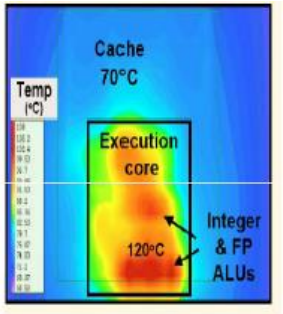
- Battery technology evolution is slow so
  - ☐ Power Management becomes a must!
  - ☐ Thermal dissipation for miniaturized SoCs is a bottleneck : 15°C excess of temperature => MTBF drops by 30% (radio equipment)



Source: TI estimates



Powermag 2013

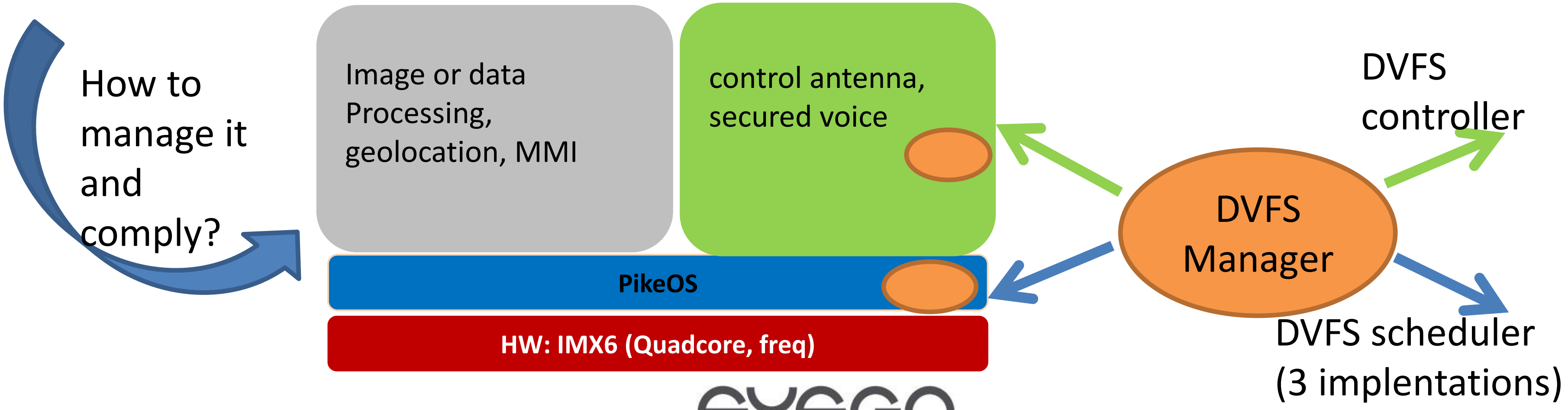


Technical choices and architecture

Context & choice: use COTS multicores, DVFS based methods

- **Running the applications on a COTS quadcore HW**
  - □ **Suitable for handheld devices**
  - □ **Suitable for DVFS mechanisms; DVFS and a QoS driven scheduler are efficient**
  - □ **But all the cores run at the same frequency**
  - □ **The architecture is based on shared resources**

How to guarantee then a system determinism while application load is changing and frequency/voltage are being modified?



How to manage it and comply?

Image or data Processing, geolocation, MMI

control antenna, secured voice

PikeOS

HW: IMX6 (Quadcore, freq)

DVFS Manager

DVFS controller

DVFS scheduler (3 implentations)

Technical choices and architecture

Context & choice: PikeOS considerations for safety and security on multicore

- Safety considerations
  - □ Address all applicable safety standards (DO-178B, IEC 61508 EN 50128)
  - □ Guarantee determinism with respect to timing and resource consumption
- Security considerations
  - □ Support evaluation according to Common Criteria
  - □ Prohibit any information flow between VMs
  - □ Comply to security design patterns (MILS, SKPP)
- Flexible and Extensible
  - □ Supports all important CPU Architectures like x86, PowerPC, ARM, MIPS and Sparc today. Easily portable to new platforms
  - □ Multiple guest operating systems (personalities)
- Very often, compromise with hard real time properties and check the overhead

Low Power scheduling perspectives for mixed critical systems

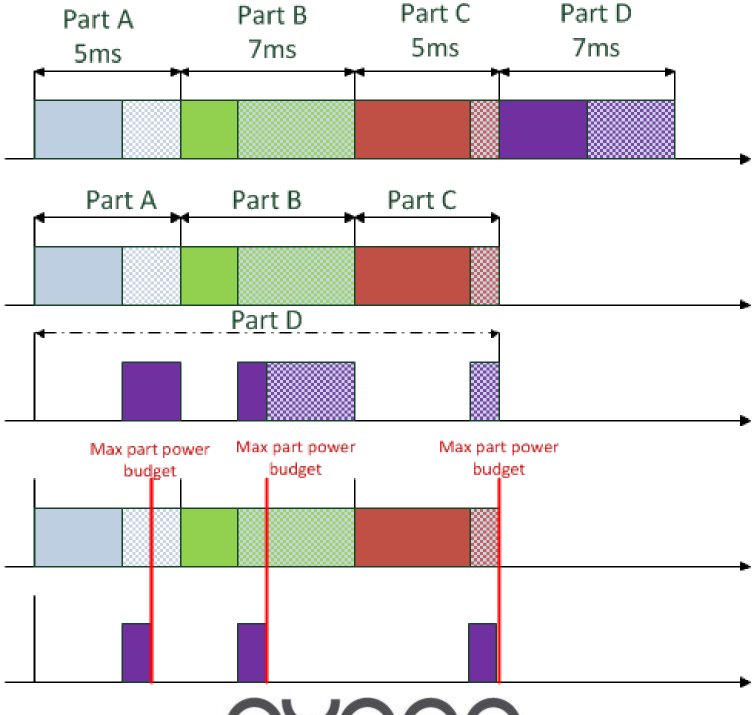
Option 1: Integration as extensions of PikeOS scheduler

- A new parameter must be introduced for the VMs and handled by PikeOS scheduler:
  - □ Max power budget to be consumed for a VMs during its time partition
  - □ It only impacts the background time partition execution duration, and/or low priority VMs

Traditional ARINC653 scheduling

Scheduling with background time partition

Scheduling with background time partition and limited power budget



Low Power scheduling perspectives for mixed critical systems

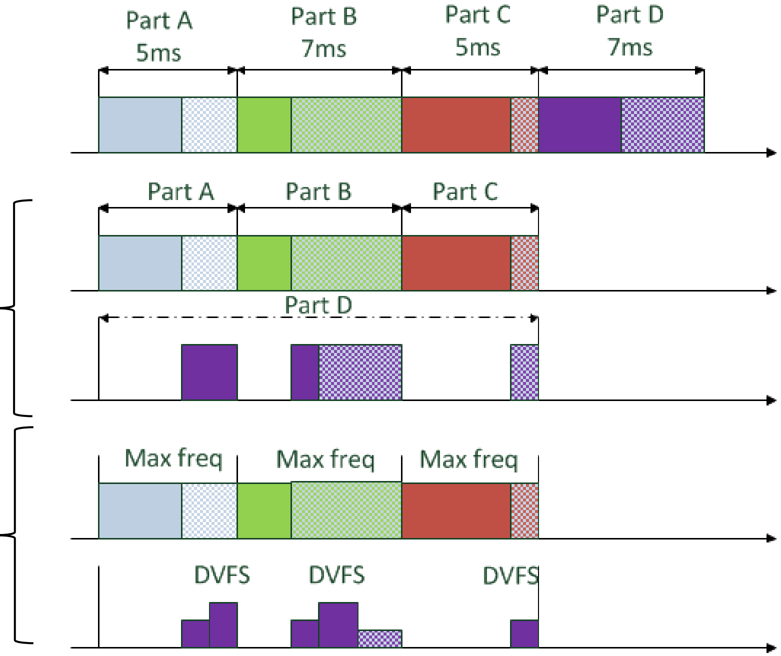
Option 2: Integration at level of user VMs

- Apply DVFS/DPM only on Low Priority VMs
  - □ Low-priority VMs are allocated to background time partition
- A new parameter must be handled by PikeOS scheduler: CPU frequency
  - □ This is required to ensure the right power mode is used when High priority VMs are scheduled

Traditional ARINC653 scheduling

Scheduling with background time partition

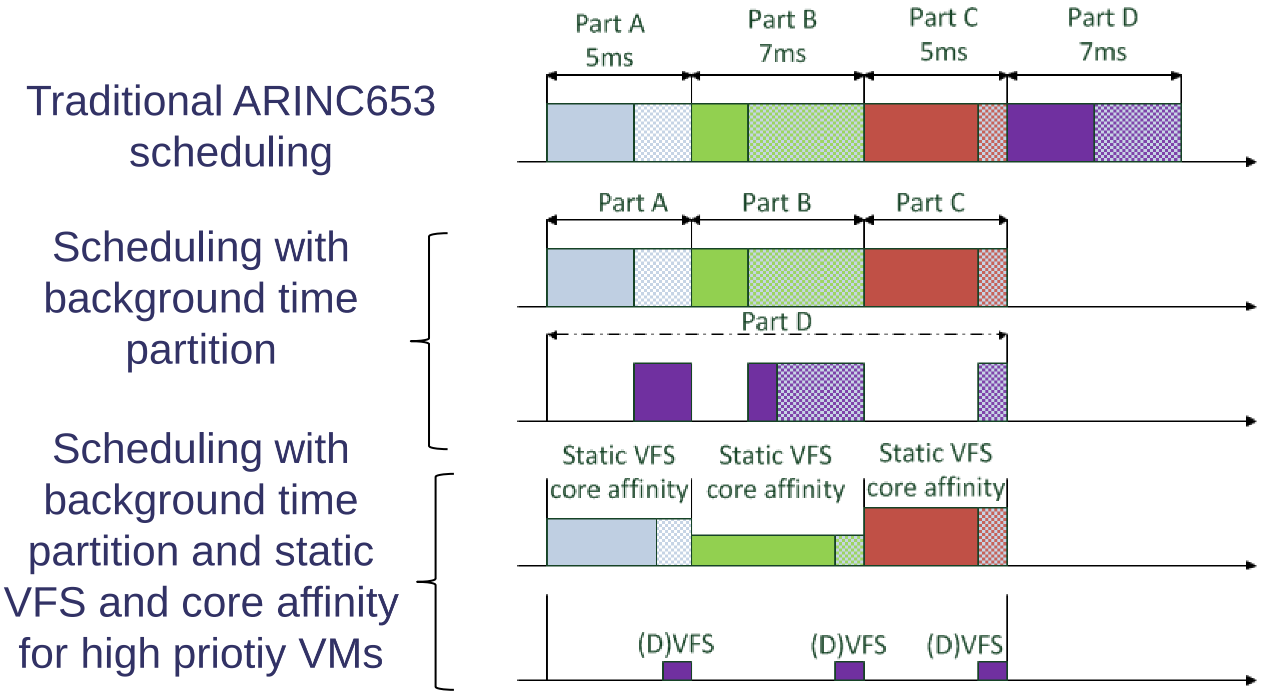
Scheduling with background time partition and DVFS management on this low-priority VMs



Low Power scheduling perspectives for mixed critical systems

Option 3: Integration at the level of PikeOS extensions

- Allocate high performance cores to critical VMs
- Allocate low performance cores to low priority VMs
  - □ The system integrator must provide the number of cores and/or frequencies to be used for each partitions to ensure deterministic behavior of the application



Traditional ARINC653 scheduling

Part A 5ms

Part B 7ms

Part C 5ms

Part D 7ms

Scheduling with background time partition

Part A

Part B

Part C

Part D

Scheduling with background time partition and static VFS and core affinity for high priotiy VMs

Static VFS core affinity

Static VFS core affinity

Static VFS core affinity

(D)VFS

(D)VFS

(D)VFS

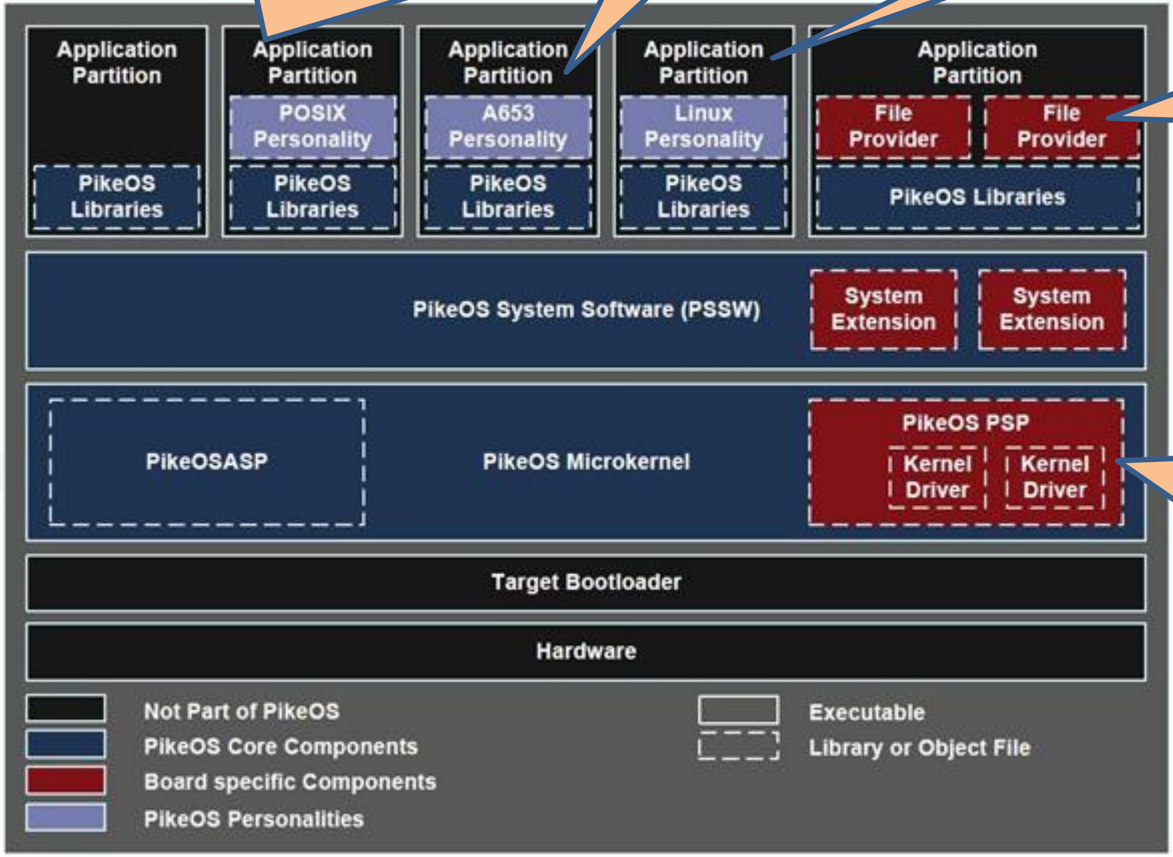
Architectures examples

Native partition, as master, mostly sleeping
Crontrol application (scheduler) here with access
- right to DVFS driver
- Appli partition monitoring access rights

Fixed freq application here with hard real time constraints
No access to DVFS

application running with « soft » real time constraints
No access to DVFS

- 4th partition with DVFS driver
(Easier for first design)



A driver Implementation is required
- if interupts ack are needed
- If too slow at File Provider level

THANK YOU!