



**HAL**  
open science

## Process-level Power Estimation in Multi-core Architectures

Maxime Colmant, Romain Rouvoy, Lionel Seinturier

► **To cite this version:**

Maxime Colmant, Romain Rouvoy, Lionel Seinturier. Process-level Power Estimation in Multi-core Architectures. Compas, Jun 2015, Lille, France. hal-01171704

**HAL Id: hal-01171704**

**<https://inria.hal.science/hal-01171704>**

Submitted on 9 Jul 2015

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Copyright

# Process-level Power Estimation in Multi-core Architectures \*

Maxime Colmant, Romain Rouvoy, Lionel Seinturier

\* Published article: Colmant (M.), Kurpicz (M.), Huertas (L.), Rouvoy (R.), Felber (P.) et Sobe (A.). – Process-level Power Estimation in VM-based Systems. – In *European Conference on Computer Systems (EuroSys)*, Bordeaux, France, avril 2015

## Motivation

### Problem

- Hard to identify the largest power consumers
- Power meters are not fine-grained enough
- Limited number of processor-agnostic solutions
- Limited number of power-aware interfaces

### Vision

- Software-defined power meters
- Support for all CPU features
- Provide critical indicators

## Metrics

### Hardware (HW) Performance Counters

- Representative and accurate metrics
- Mostly available on modern processors

### Criteria selection

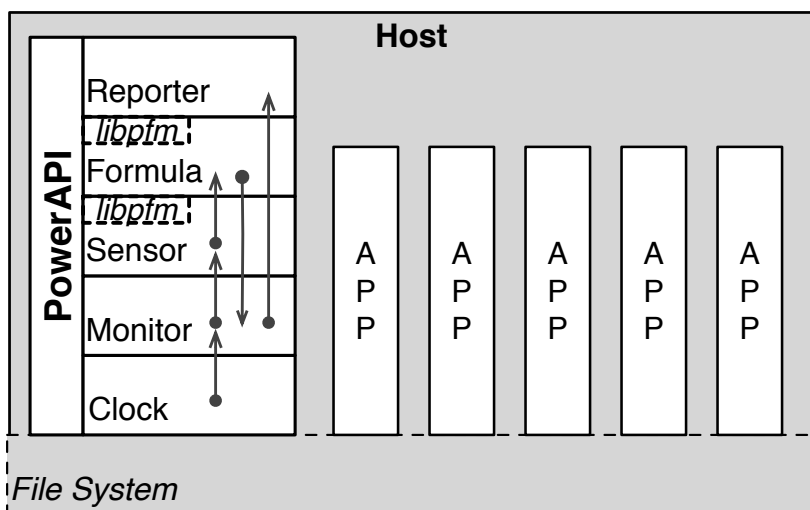
- Counter availability per CPU
- Monitoring overhead
- Best fit under several workloads

### Selected HW Performance Counters

- CPU\_CLK\_UNHALTED:THREAD\_P ( $uc$ )
- CPU\_CLK\_UNHALTED:REF\_P

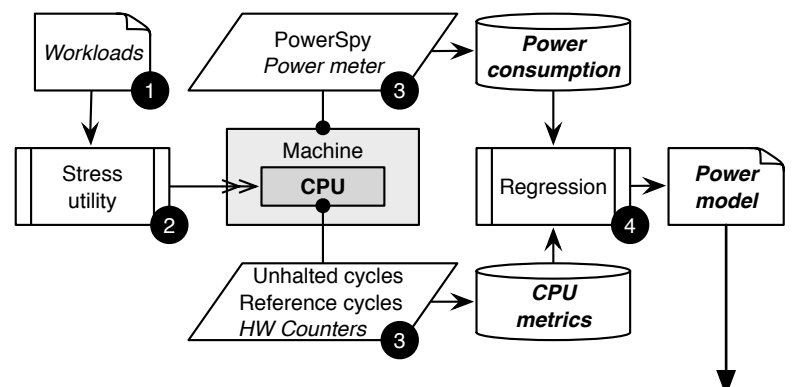
## PowerAPI

### Implementation



Available online:  
<http://powerapi.org>

### Learn the CPU power model

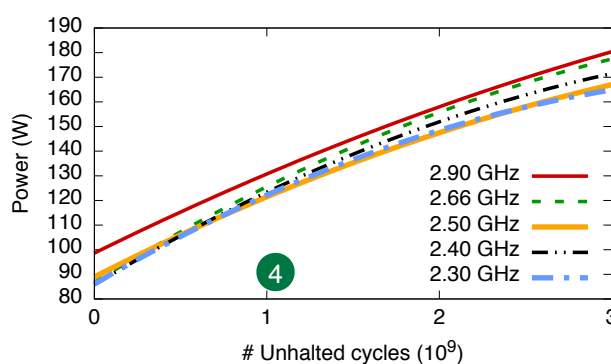
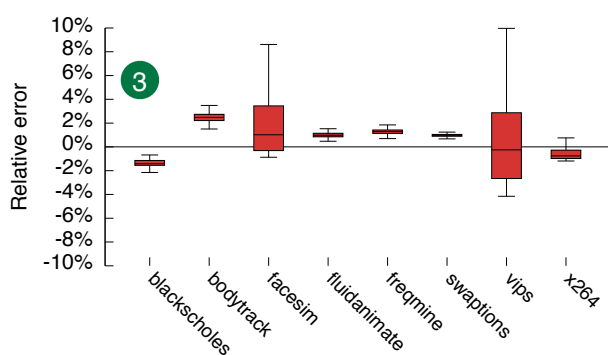
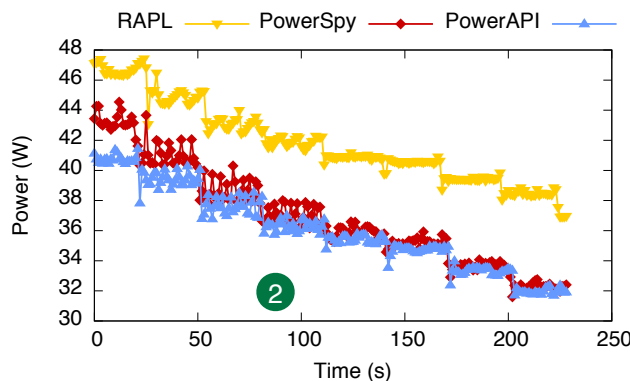
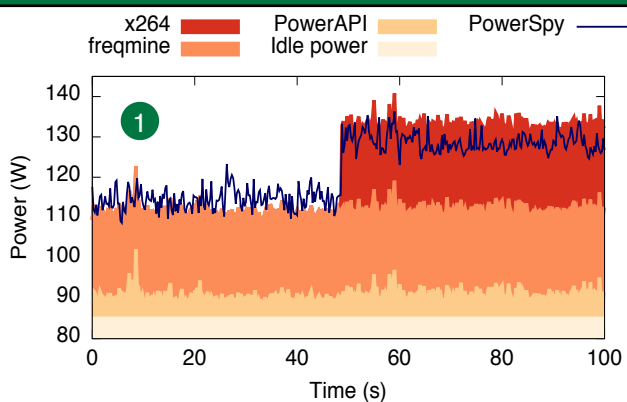


### Power Model

$$P_{host}(f) = P_{idle}(f) + \sum_{pid \in PIDs} P_{cpu}(f, uc_{pid}^1 \dots uc_{pid}^N)$$

$$P_{cpu}(f, uc_{pid}^1 \dots uc_{pid}^N) = \sum_{n=1}^N P_f(uc_{pid}^n)$$

## Validation



- 1 Process-level power consumption of PowerAPI, x264, and freqmine on a Xeon W3520 processor
- 2 Decreasing load of stress on a i3 2120 processor, compared to RAPL
- 3 Relative error distribution of the PARSEC benchmarks on a Xeon W3520 processor
- 4 Power models for the highest frequencies on a Xeon W3520 processor