



**HAL**  
open science

## **BitWatts: A Process-level Power Monitoring Middleware**

Maxime Colmant, Mascha Kurpicz, Pascal Felber, Loïc Huertas, Romain  
Rouvoy, Anita Sobe

► **To cite this version:**

Maxime Colmant, Mascha Kurpicz, Pascal Felber, Loïc Huertas, Romain Rouvoy, et al.. BitWatts: A Process-level Power Monitoring Middleware. Middleware - Poster session, Dec 2014, Bordeaux, France. 2014, 10.1145/2678508.2678529 . hal-01078825v2

**HAL Id: hal-01078825**

**<https://inria.hal.science/hal-01078825v2>**

Submitted on 8 Dec 2014

**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.

# BitWatts: A Process-level Power Monitoring Middleware

Maxime Colmant, Mascha Kurpicz, Pascal Felber, Loïc Huertas, Romain Rouvoy, Anita Sobe

## Motivation

- Create software-defined power meters
- Identify the largest power consumers
- Provide critical indicators
- Make informed decisions (heuristics, power capping)
- Provide architecture-agnostic solutions
- Limited number of power-aware interfaces (RAPL)

## Metrics

### Hardware Performance Counter (HPC):

- Representative and accurate metrics
- Mostly available on modern processors

### Criteria selection:

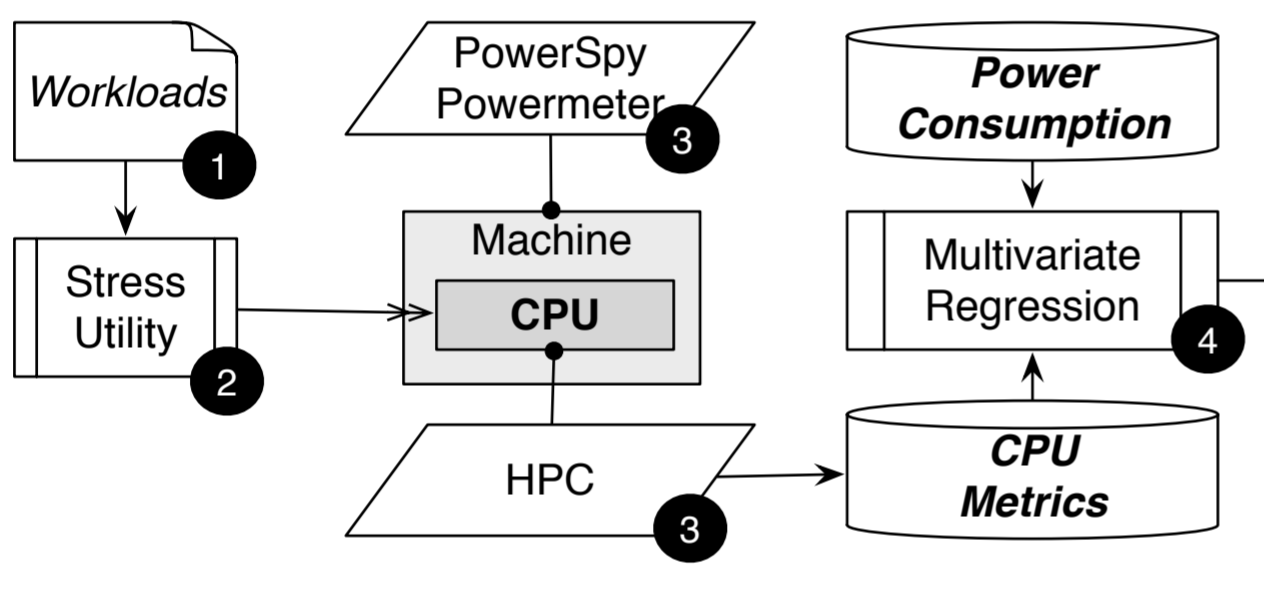
- Counter availability per CPU
- Monitoring overhead
- Best fit under several workloads

### Selected HPC:

- instructions ( $i$ ), cache-references ( $r$ ), cache-misses ( $m$ )

## BitWatts Middleware

### Learning the CPU Power Model

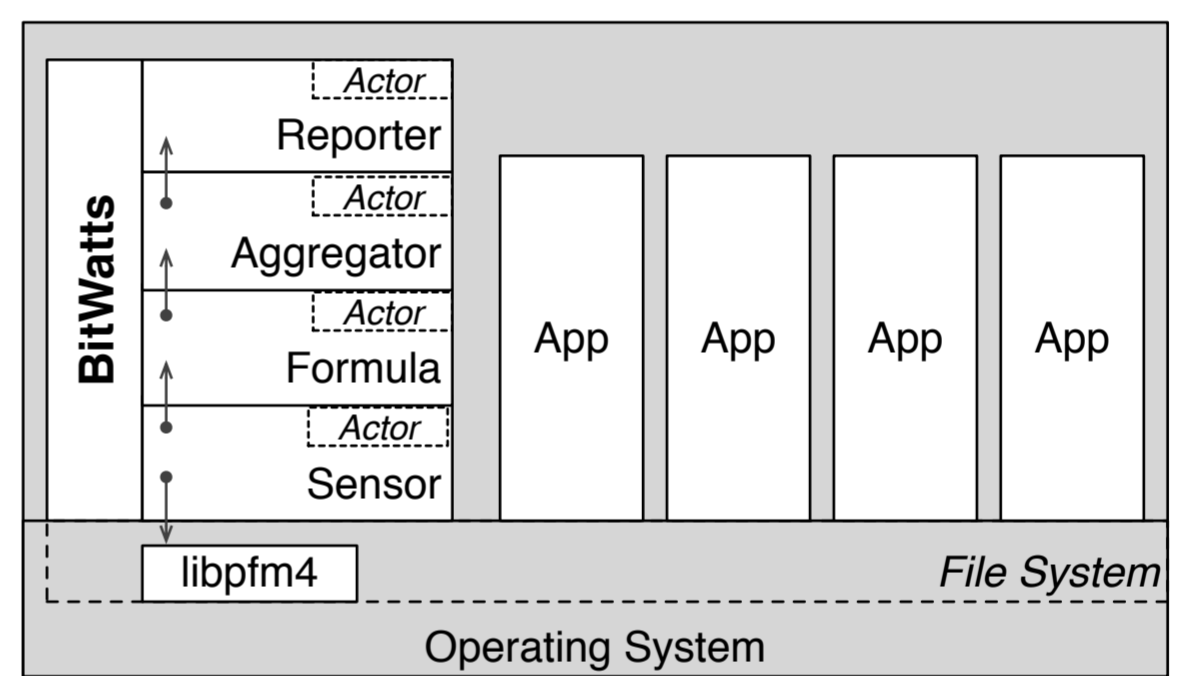


### Power Model

$$Power_{Xeon} = 90.23 + \sum_{\substack{f=1.6, \\ f \in Freq}}^{2.93} Power_f$$

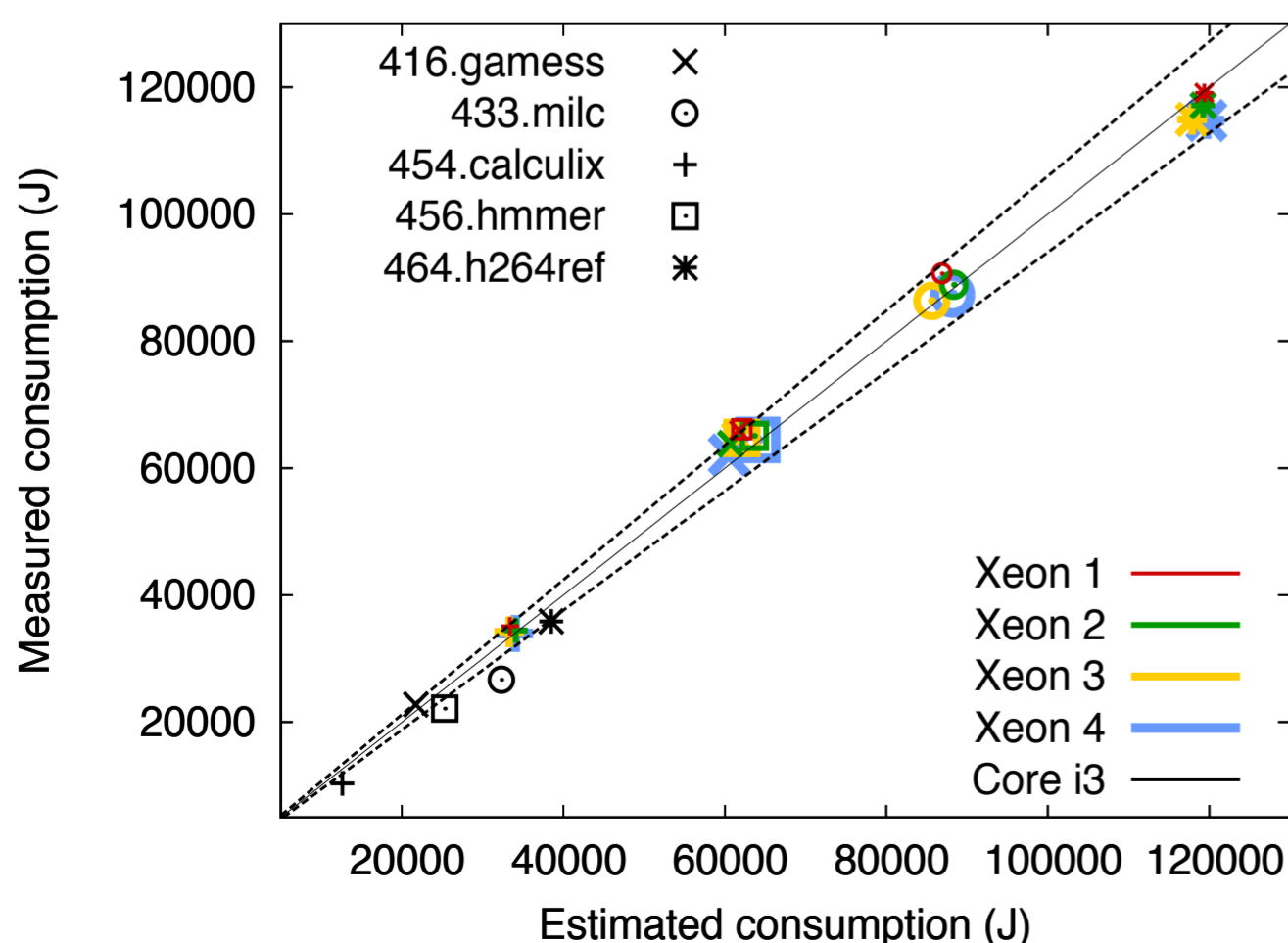
$$Power_{2.93} = \frac{5.37 \cdot i}{10^9} + \frac{7.67 \cdot r}{10^8} + \frac{3.23 \cdot m}{10^7}$$

### Architecture



- Actor programming model (Scala / Akka)
- Modular & Scalable Middleware
- Real-time power estimation (10Hz-100Hz)

## Preliminary experiment



## Conclusion

### A Middleware to build software-defined power meters

- High-level power API
- Processor-agnostic solution
- Power-model inference by sampling
- Real-time power estimation

### Outlook

- Support AMD, ARM
- Support virtualized environments
- Identify automatically the HPCs
- Heuristics, power capping