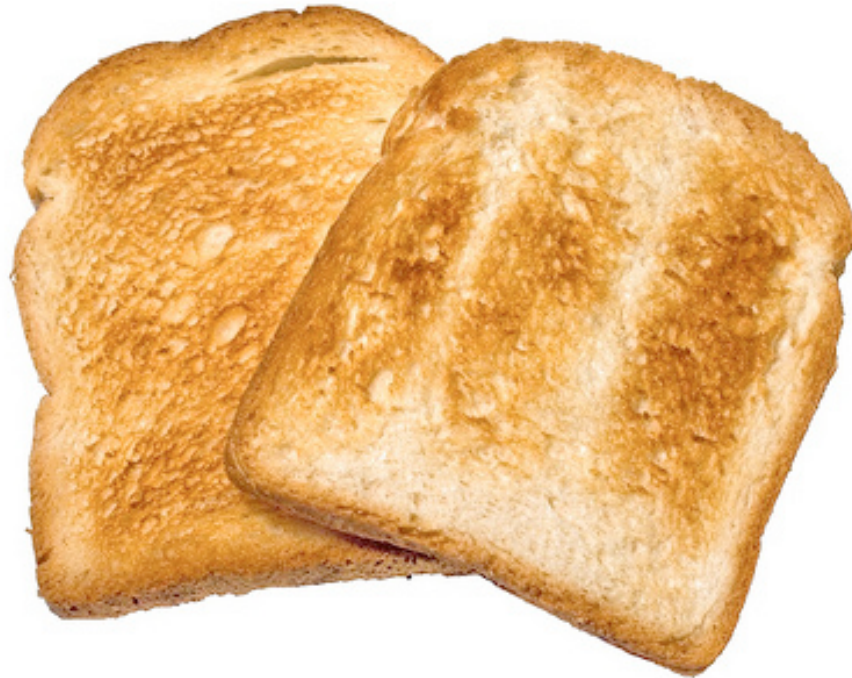


PUE for end users

Are you interested in something more than bread toasting?



Kay Grosskop, Software Improvement Group (SIG), Amsterdam
EASED@BUIIS, Oldenburg 25 April 2013



Software Improvement Group

- Background
 - Spin-off from CWI in 2000, self-owned, independent
 - Strong academic background, innovative, award-winning, profitable
 - Member of the consortium *Green IT Amsterdam* region
 - Presiding the *Knowledge Network Green Software (KNGS)*
 - Initiator of the Software Energy Footprint Lab (SEFLab)
- Activity
 - Management advisory, fact-based
 - Accredited software analysis lab employs analysis tools and models
 - Experienced staff transforms analysis data into advice
- Track record
 - Finance, government, logistics, telecom, manufacturing, energy, ...
 - We analyze over 500 systems each year

PUE : Power usage effectiveness

$PUE = \text{Facility power} / \text{IT power}$

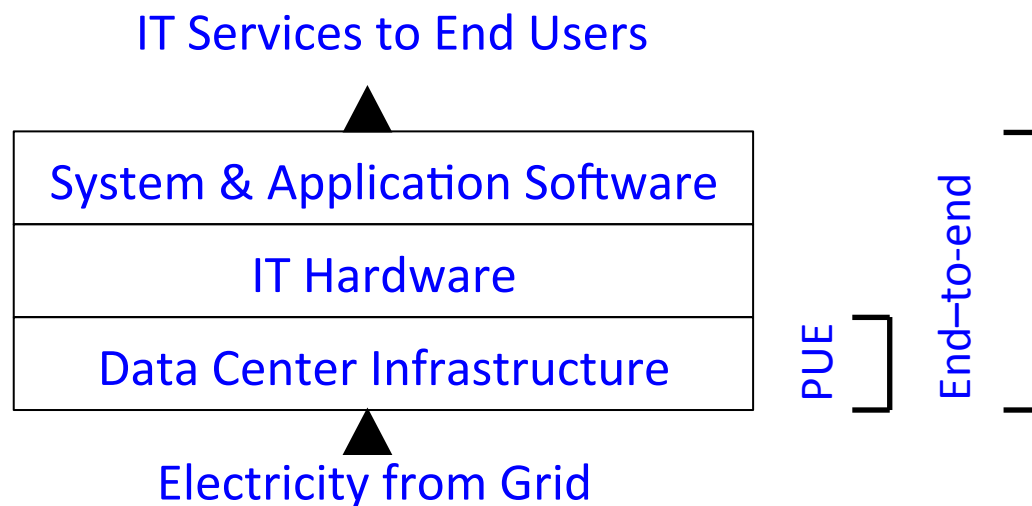
- Widely used
- Simple & understandable for all stakeholders
- Addressed biggest consumers in absolute numbers

Problems with PUE (1)

Addresses only part of the stack

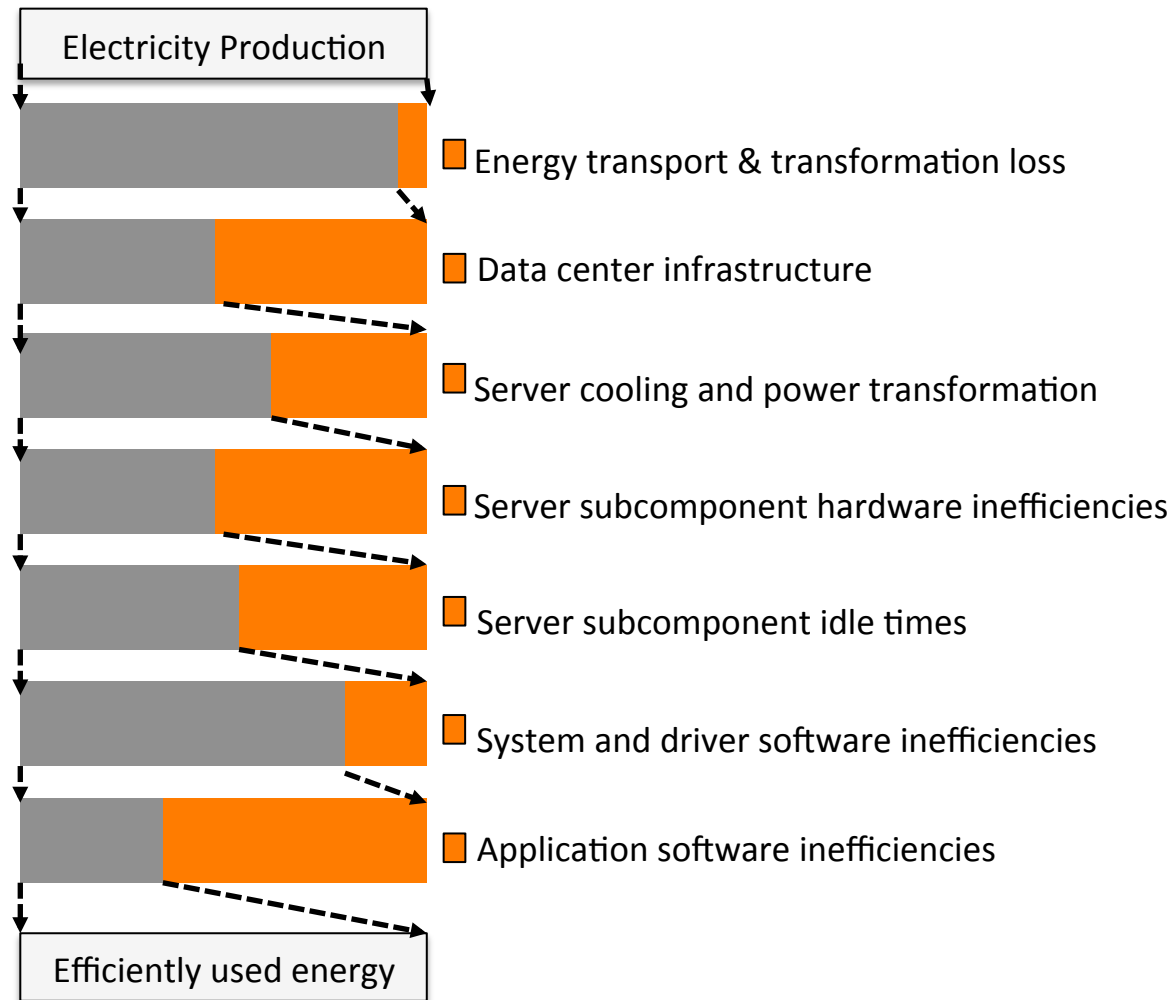
→ No incentive for optimization of 'IT' (hard- & software).

Bread roasters are just fine!



The 'chain of loss'

Look at percentages rather than absolute numbers!



Switch the direction of thinking

- Take into considerations effects of software on energy efficiency of systems
- Reason about energy footprint from a point of view of the functionality that is delivered to the end user
- Switch thinking:

Reducing loss in the
power supply chain



Reducing demand of
Computing resources

Problems with PUE (2)

Unrelated to end-user functionality

→ Not meaningful for **users** and **owners** of applications

How much energy do I need to spent to get a certain functionality?

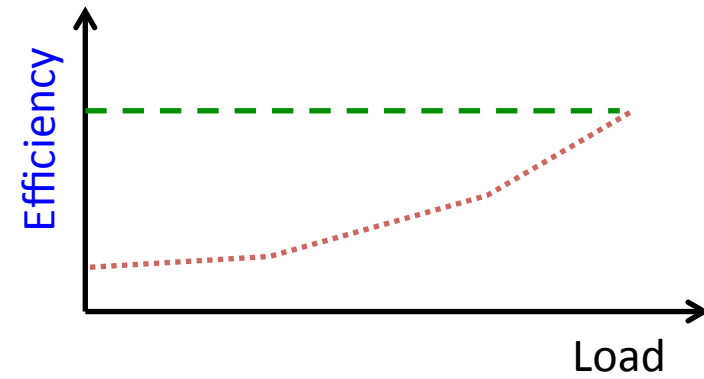
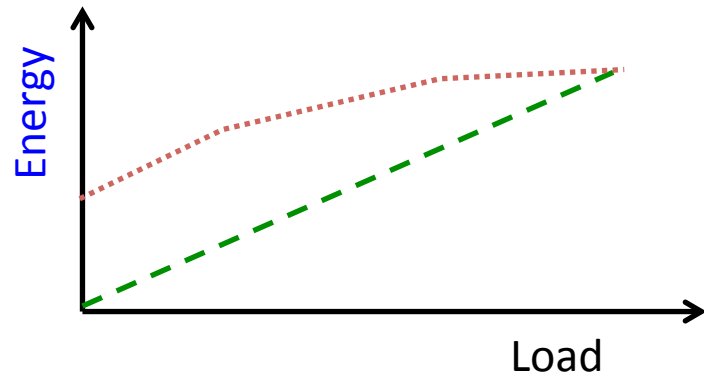
Difficulties for a broader metric:

- *Universal Unit Of Work* is hard to find
 - GreenGrid DCeP proxies (e.g. CPU-cycles) loose the relation to work that is *useful* for the end user
 - Normalizing over ALL tasks and ALL applications with a universal utility function?
- How far are we away from the *absolute optimum*? Is there an equivalent to PUE = 1.

Ideas to address these difficulties

- Use application-centric unit of work
 - Pictures rendered
 - Videos streamed
 - Webpages served
 - Data entities processed
- Look at *Energy Proportionality* rather than absolute efficiency
 - Extend the concept to functional units of work

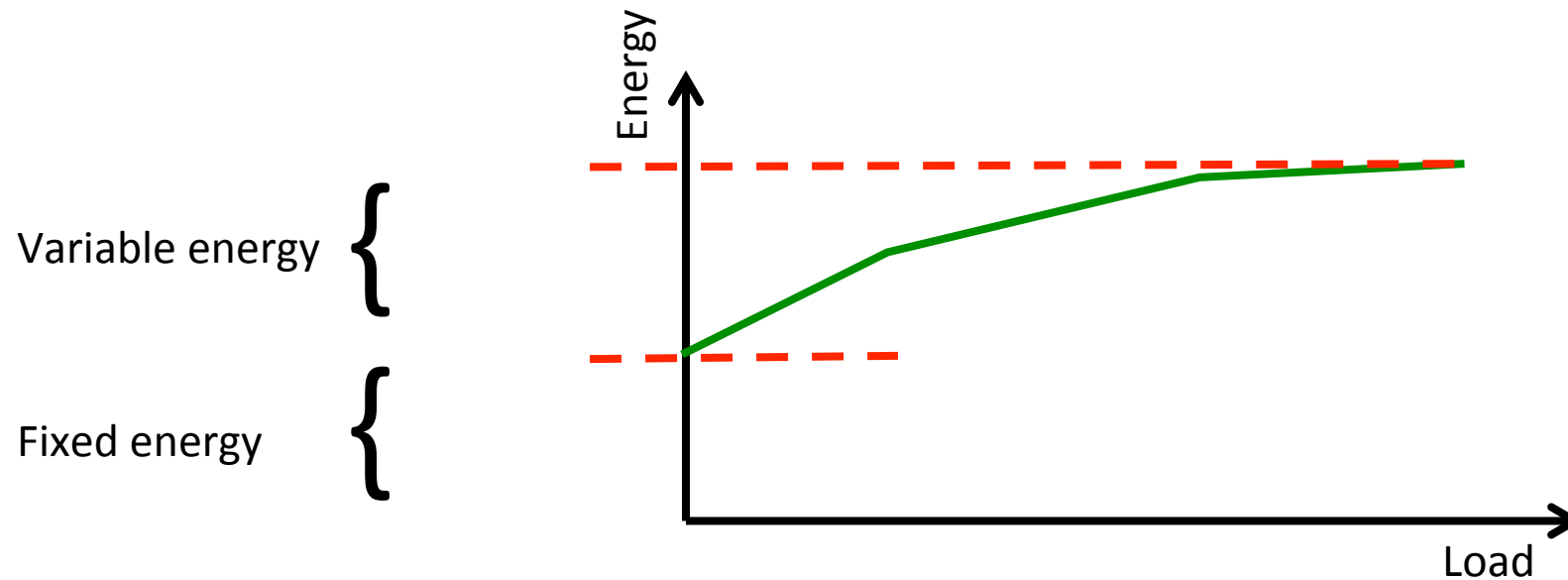
Energy proportionality



- Proportional system
- Typical system

FVER

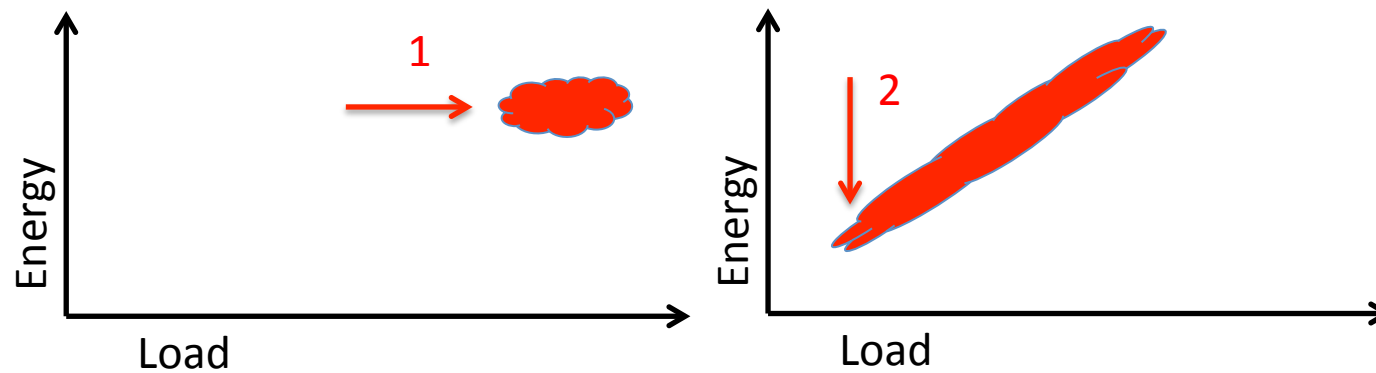
Fixed to Variable Energy Ratio



$$\text{FVER} = 1 + \left(\frac{\text{fixed energy}}{\text{variable energy}} \right)$$

Additional Idea

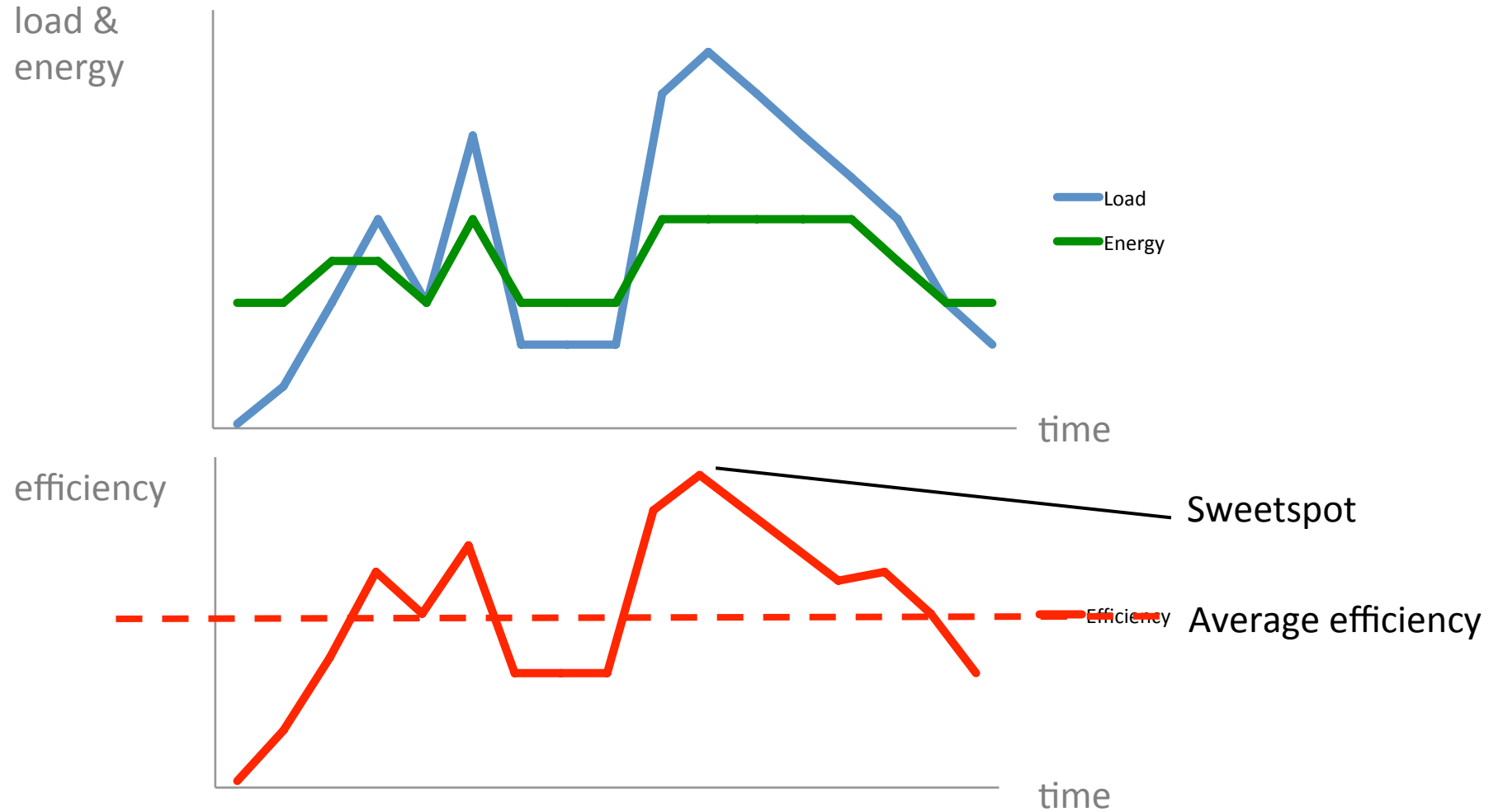
You only need to be flexible if your workload fluctuates



1. You can try to stabilize your workload on a level where your system works efficient
2. You can make the system energy-proportional.

CNS

Consumption Near Sweetspot



$$\text{CNS} = \text{sweetspot efficiency} / \text{average efficiency}$$

Don't get confused

FVER

- Low means good, optimum=1 → like PUE

CNS

- High means good, optimum = 1 (or 100%)

Nice properties of CNS

- Allows comparison between functionally (and technologically) different systems.
- The application relative optimum (sweetspot) is not theoretical, but actually performed by the system.
- Can be applied to applications in a datacenter, but is in principle portable to other setups like a mobile device.

Discussion

PUE

- Addressed biggest inefficiency in absolute numbers

