Software is Energy

Lean software for Green Computing

maurizio.morisio@polito.it

Politecnico di Torino, Italy

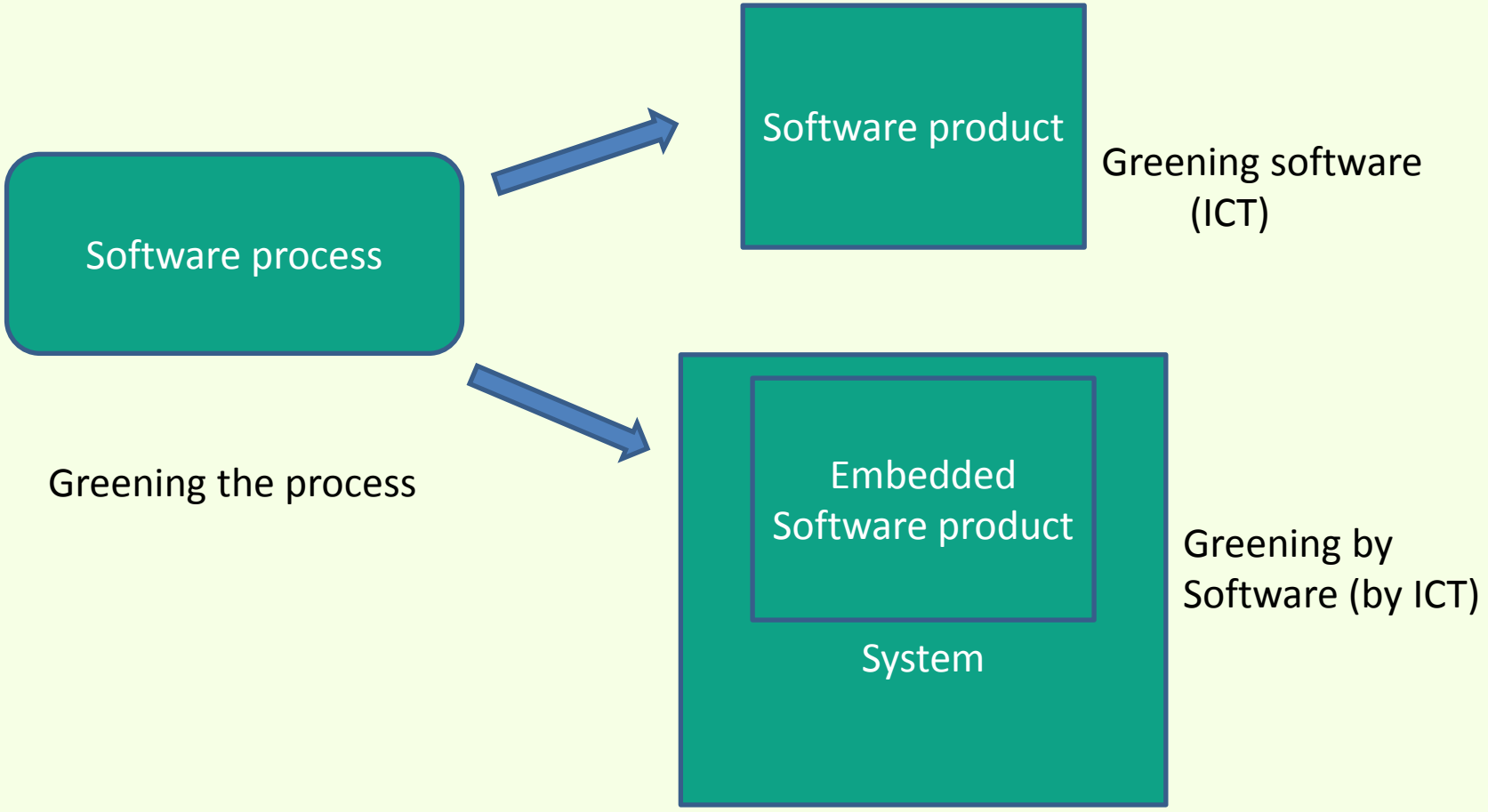
Kudos

- Luca Ardito
- Giuseppe Procaccianti
- Antonio Vetrò

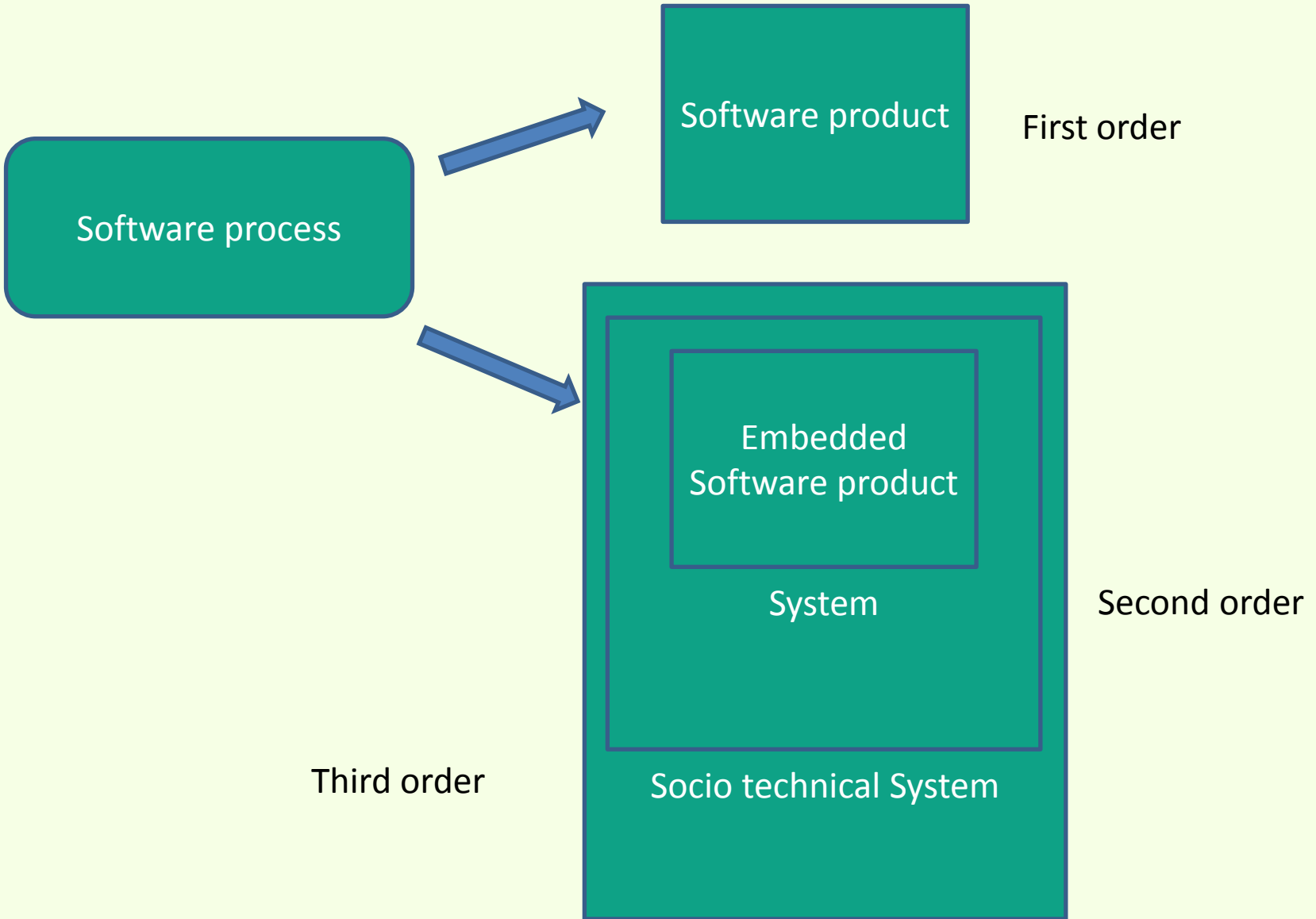



Roadmap
Guidelines
Facts
Concepts

Green what?


Software process
Software product
Greening software (ICT)
Greening the process
Embedded Software product
System
Greening by Software (by ICT)

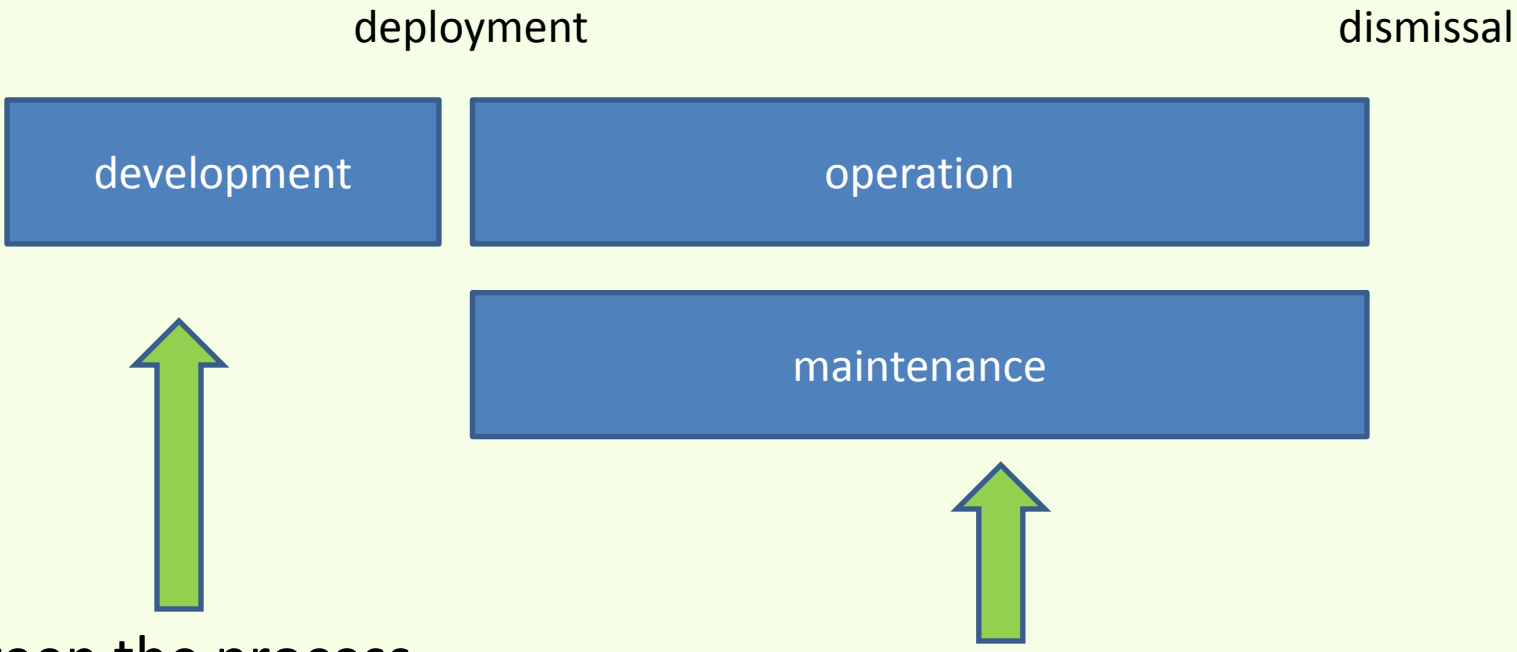
Effects


Software process
Software product
First order
Embedded
Software product
System
Second order
Socio technical System
Third order

Greening ‘by’ software

- Smart grids
- Smart cities
- Smart cars
- Smart factories
- Smart ..

Greening ‘the’ software



Green the process
(LEAN and GREEN)

zero order effect

Green the software

First order effect

Green, what?

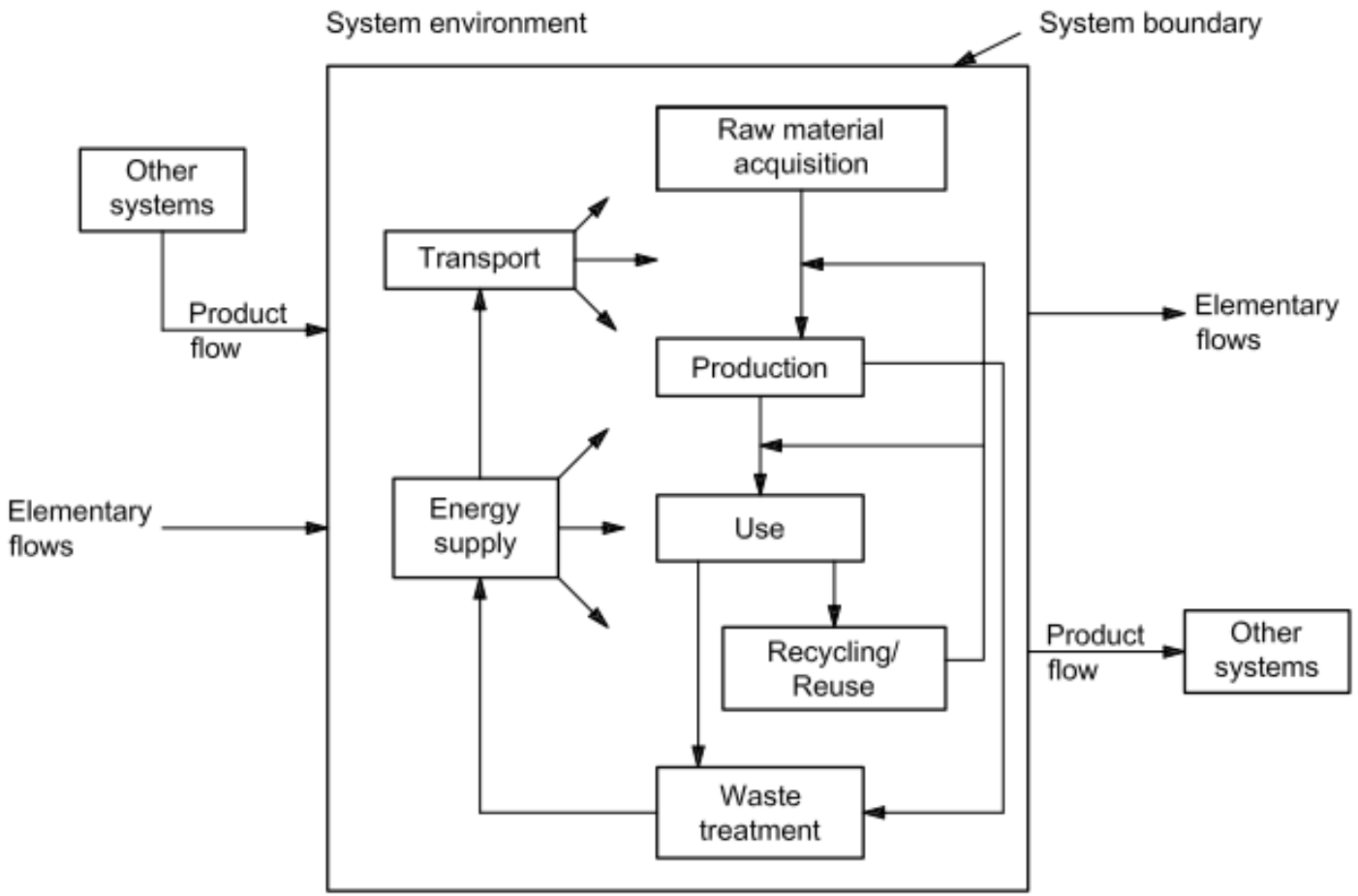
What does

software

mean exactly?

- Energy Consumption / Waste / Gas emissions
  - ISO 14040, LCA
- Sustainability
  - ISO 25010 (was ISO 9126) extended

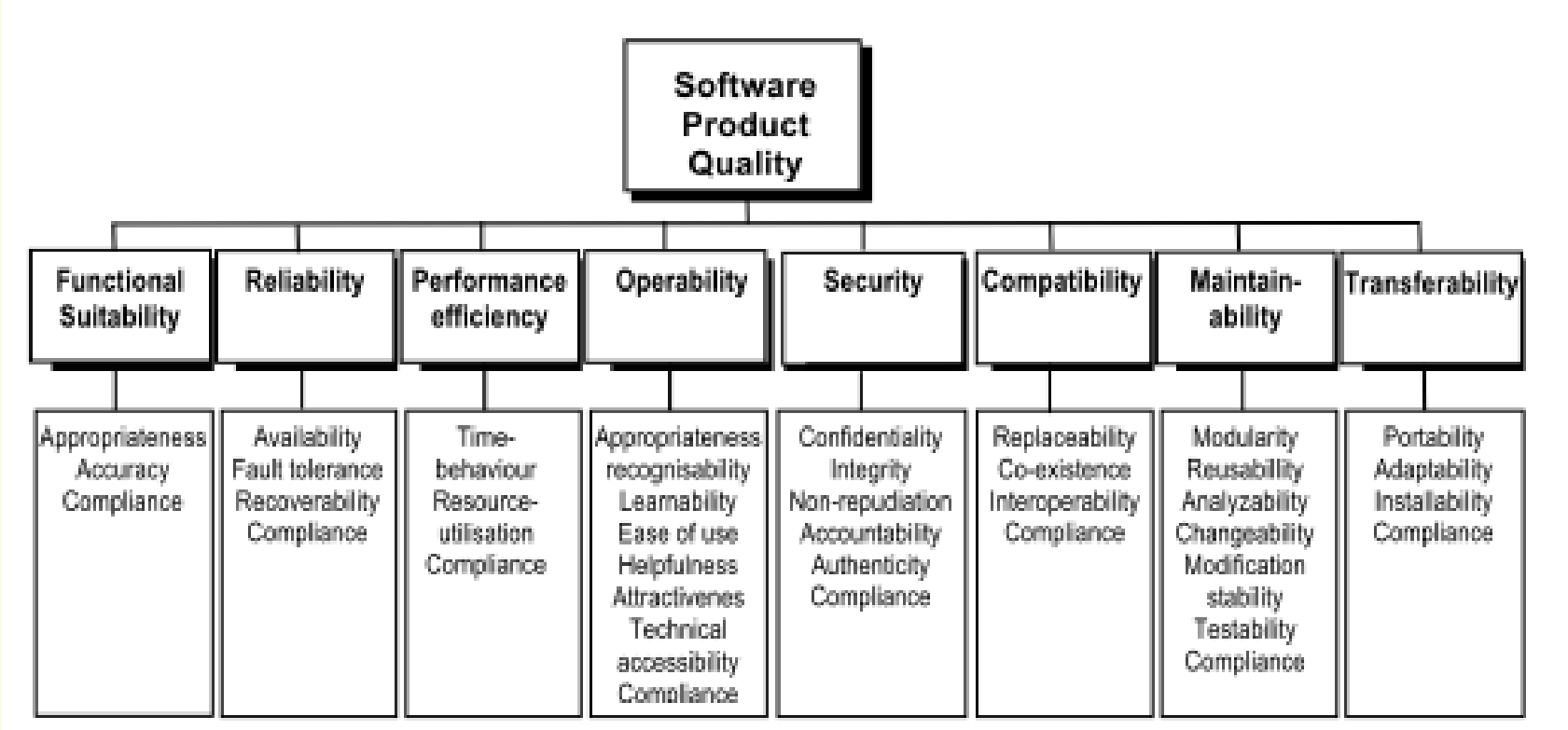
Life Cycle Assessment – ISO 14040



Sustainability

- “meeting the needs of the present without compromising the ability of future generations to meet their own needs” [UN report]
- Dimensions
  - Economic
  - Social
  - Environmental
  - Technical
  - (Human)

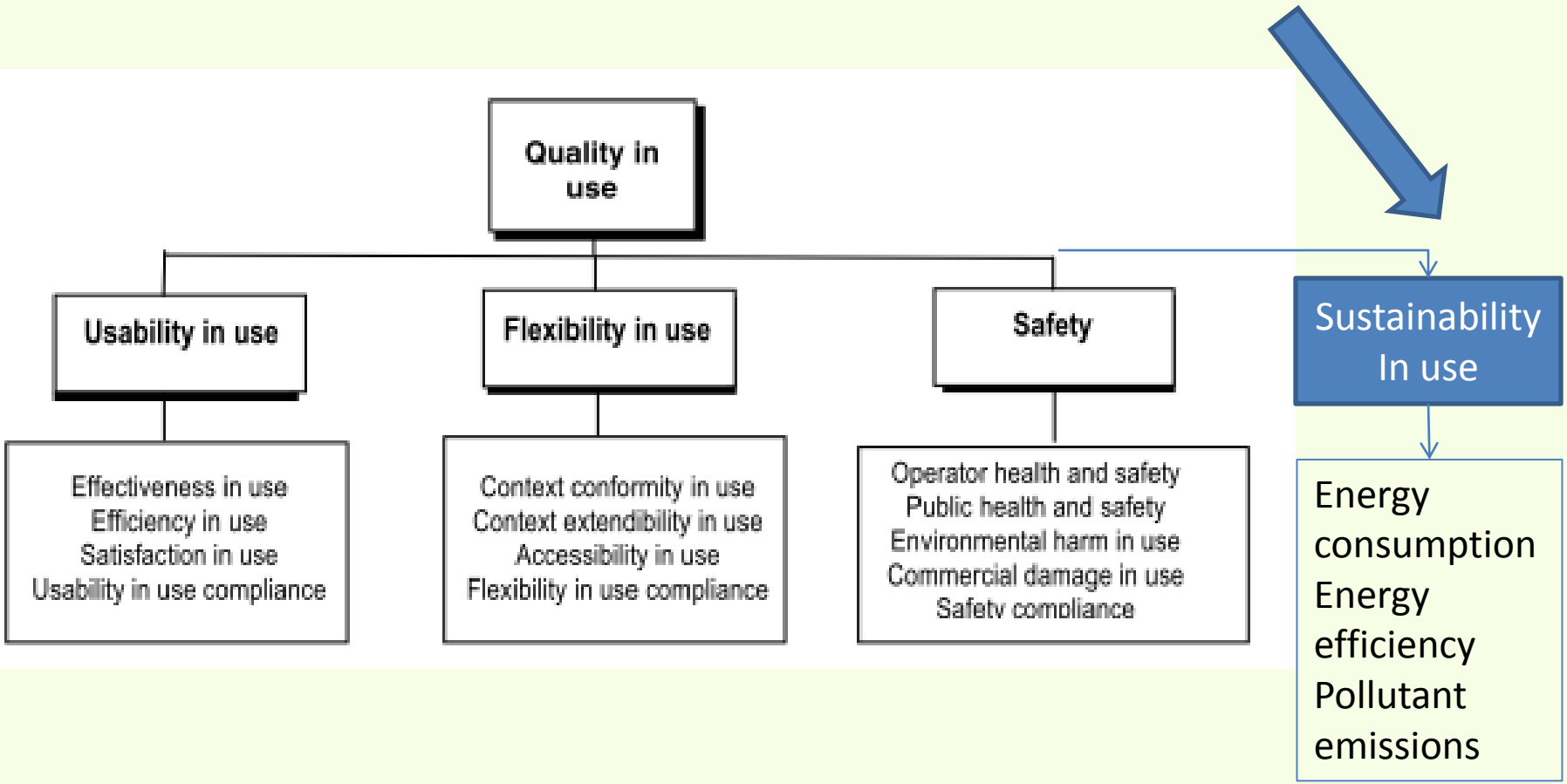
ISO 25010



Energy efficiency

Ardito L., Procaccianti G., Vetro' A., Morisio M.
Introducing Energy Efficiency into SQALE

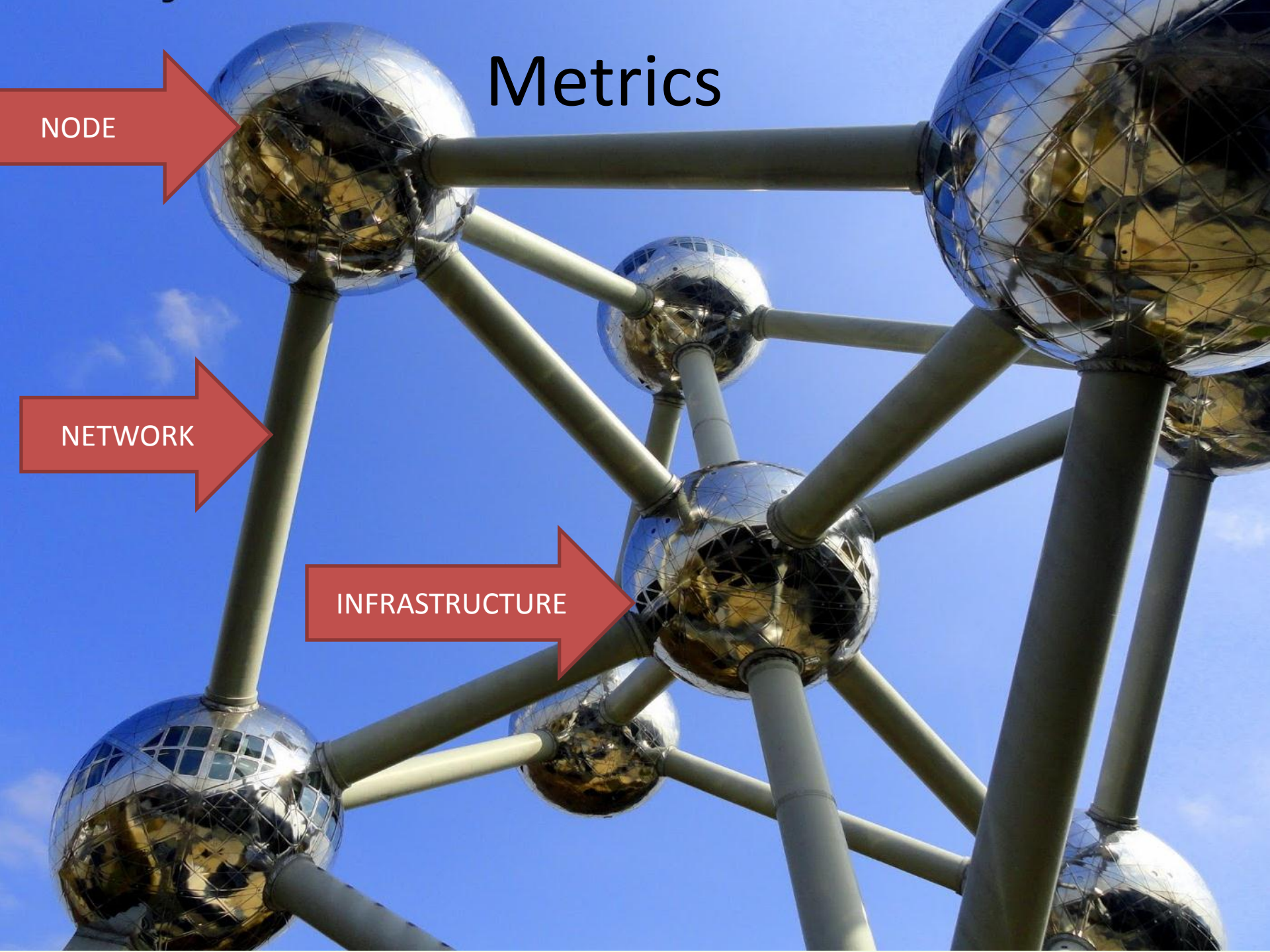
ISO 25010


Quality in use
Usability in use
Flexibility in use
Safety
Sustainability In use
Effectiveness in use
Efficiency in use
Satisfaction in use
Usability in use compliance
Context conformity in use
Context extendibility in use
Accessibility in use
Flexibility in use compliance
Operator health and safety
Public health and safety
Environmental harm in use
Commercial damage in use
Safety compliance
Energy consumption
Energy efficiency
Pollutant emissions

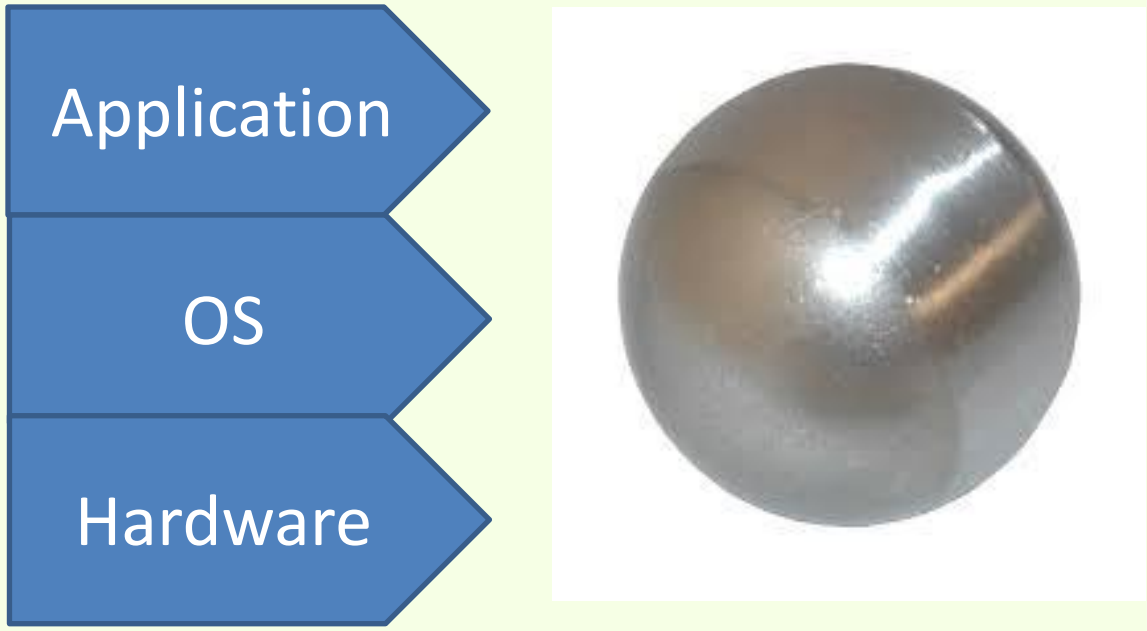
Metrics

- Sustainability
  - Environmental dimension
    - Energy [Joule]
    - Power [Watt]
    - Efficiency: useful energy / total energy
    - Productivity: computational work / energy

Ardito L., Morisio M.
Green IT - available data and guidelines for reducing energy consumption in IT Systems
SUSTAINABLE COMPUTING, Vol.4, pp.24-32, ISSN:2210-5379


Metrics
NODE
NETWORK
INFRASTRUCTURE

Node



Metrics – node level

- Power, Energy:
  - Watt, Joule
- Productivity:
  - MFLOPS / Watt
- Power, energy (node/application):
  - power / energy used by application
- Productivity (node/application):
  - sorted records / Joule
- Power (node/OS):
  - power used by OS

Metrics – network level

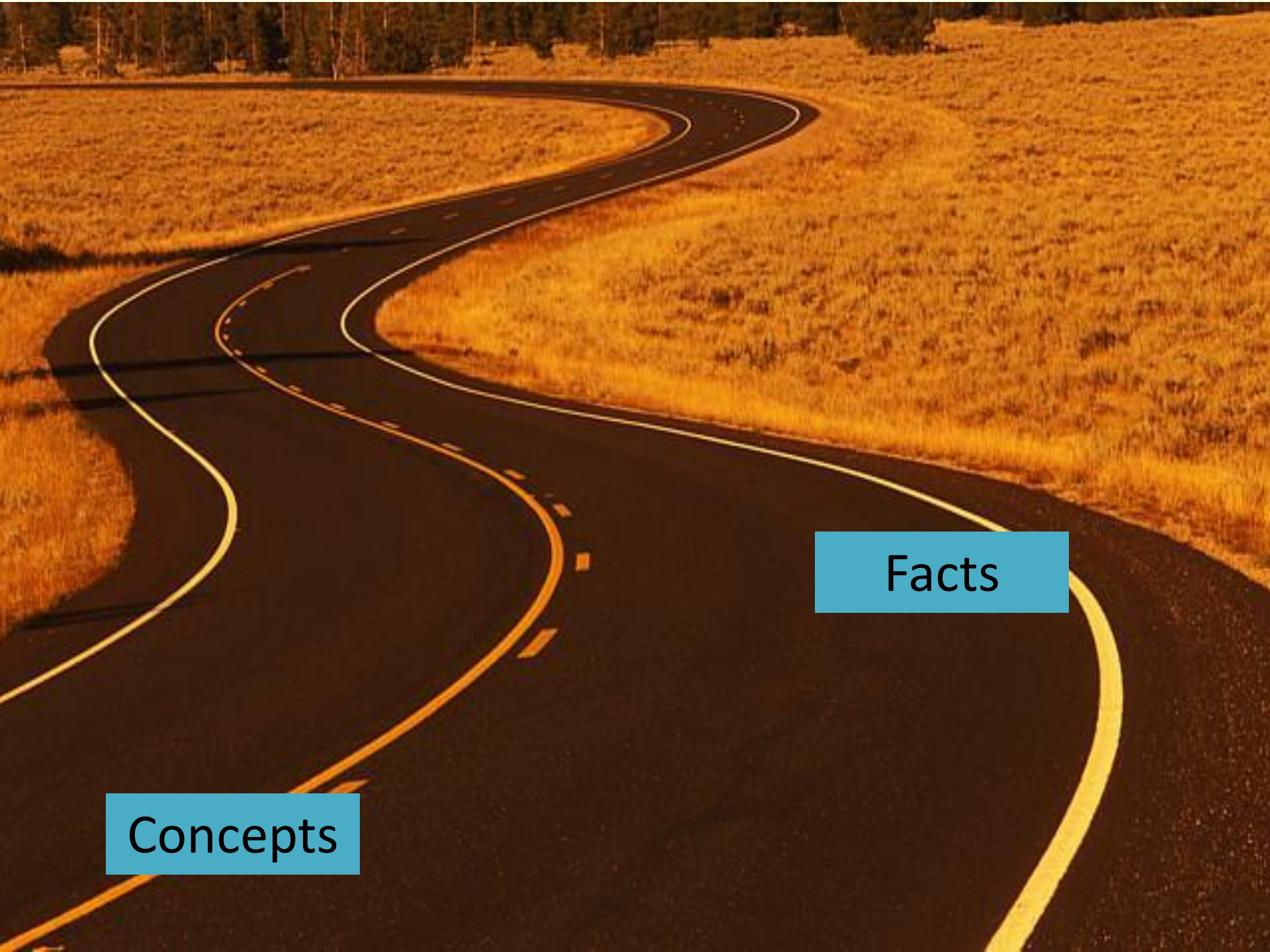
- Efficiency (network):
  - Energy (full - idle)/energy(full)
- Productivity (network):
  - KB transferred / Joule

Metrics – infrastructure level

- Productivity (data center):
  - useful work / energy
- Efficiency (data center):
  - power used for storage /total power used

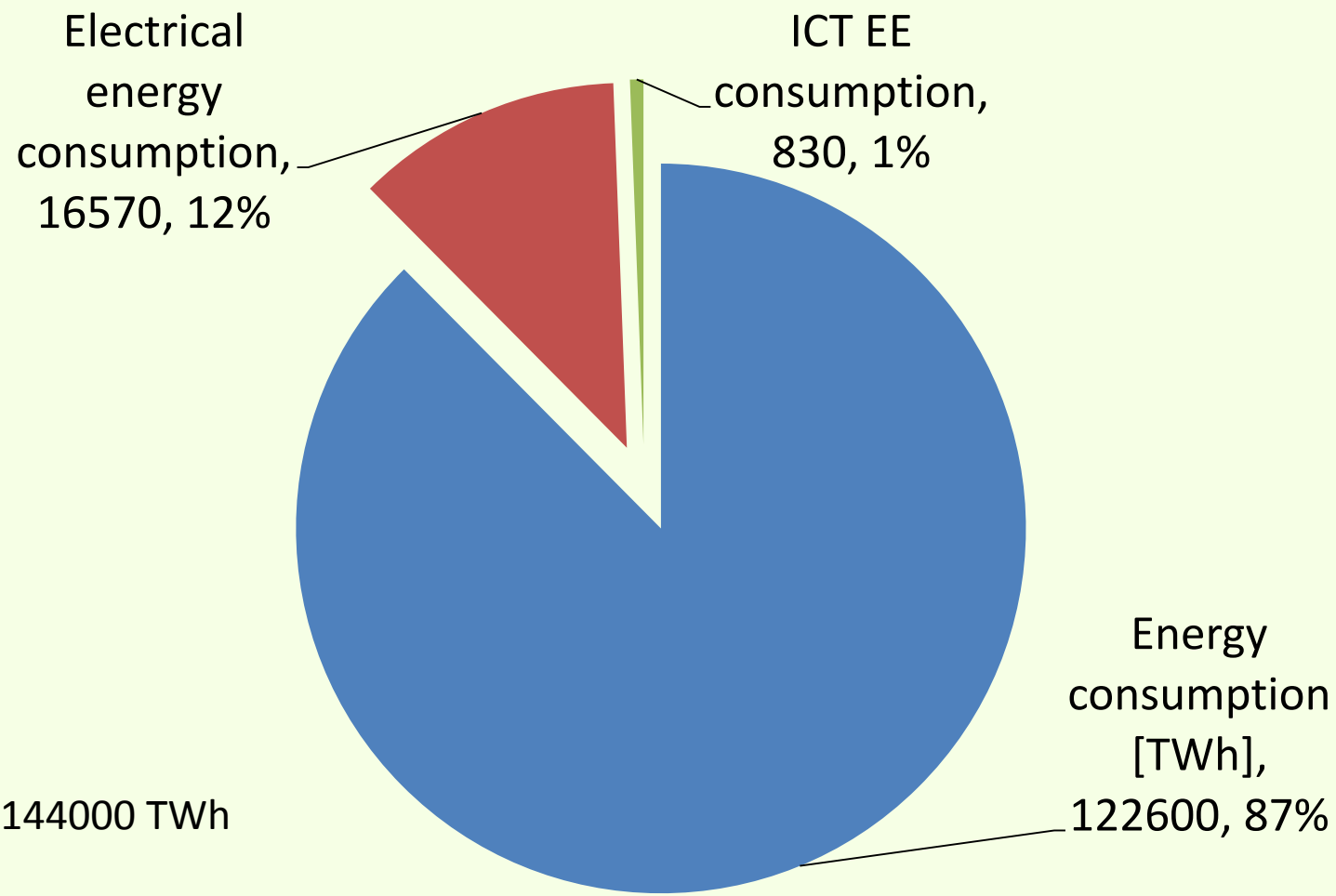
Summary

- (Zero), first, second, third level effects
- Green? Sustainable?
- No established general model
  - Suggestion, first level
    - 25010 extended with sustainability in use,
    - metrics like energy, power, efficiency, productivity
    - At node/network/infrastructure level
  - LCA to be included
    - Assumption: operation phase counts most


Facts
Concepts

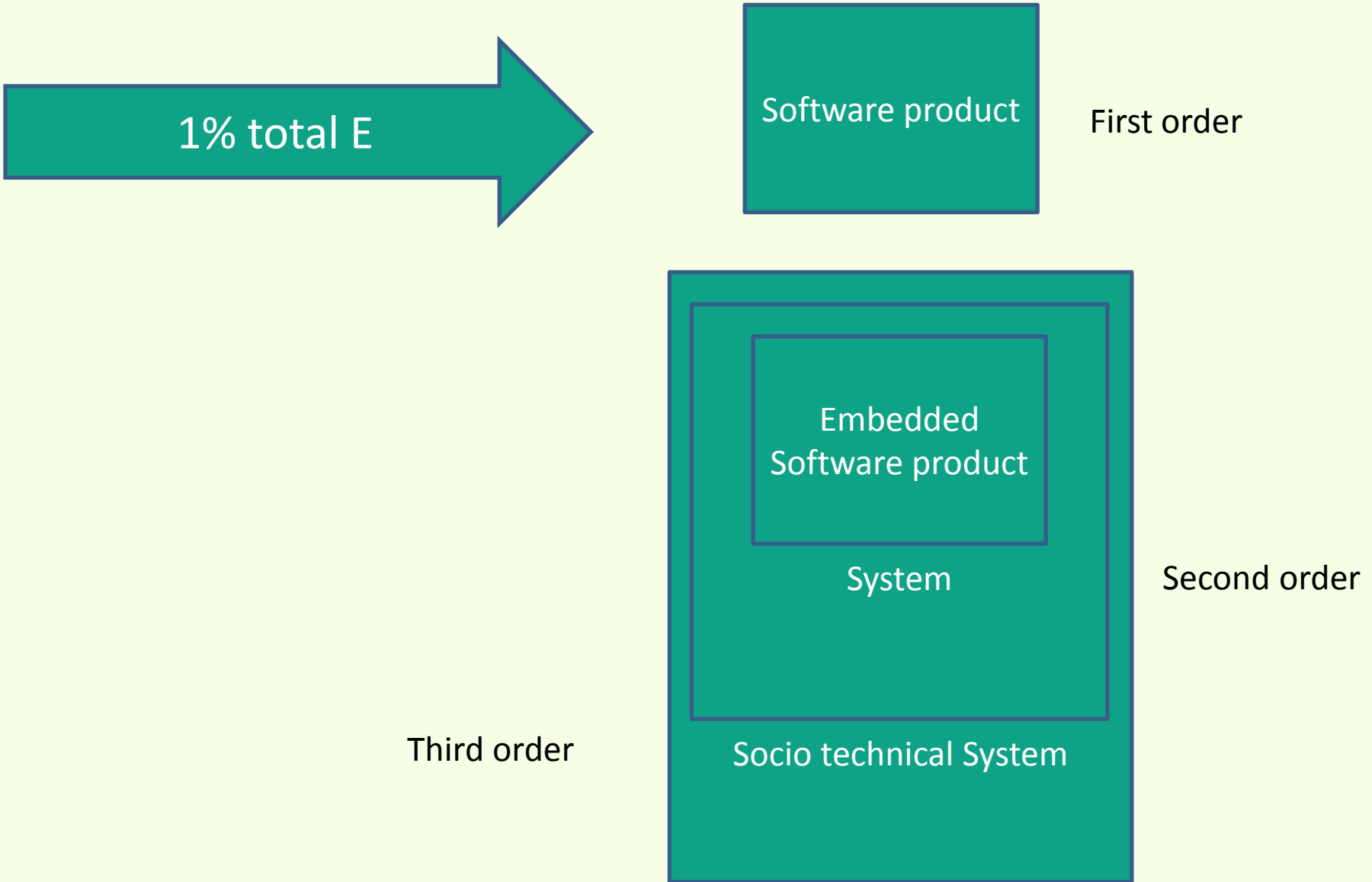
Facts

Energy consumption (2007)

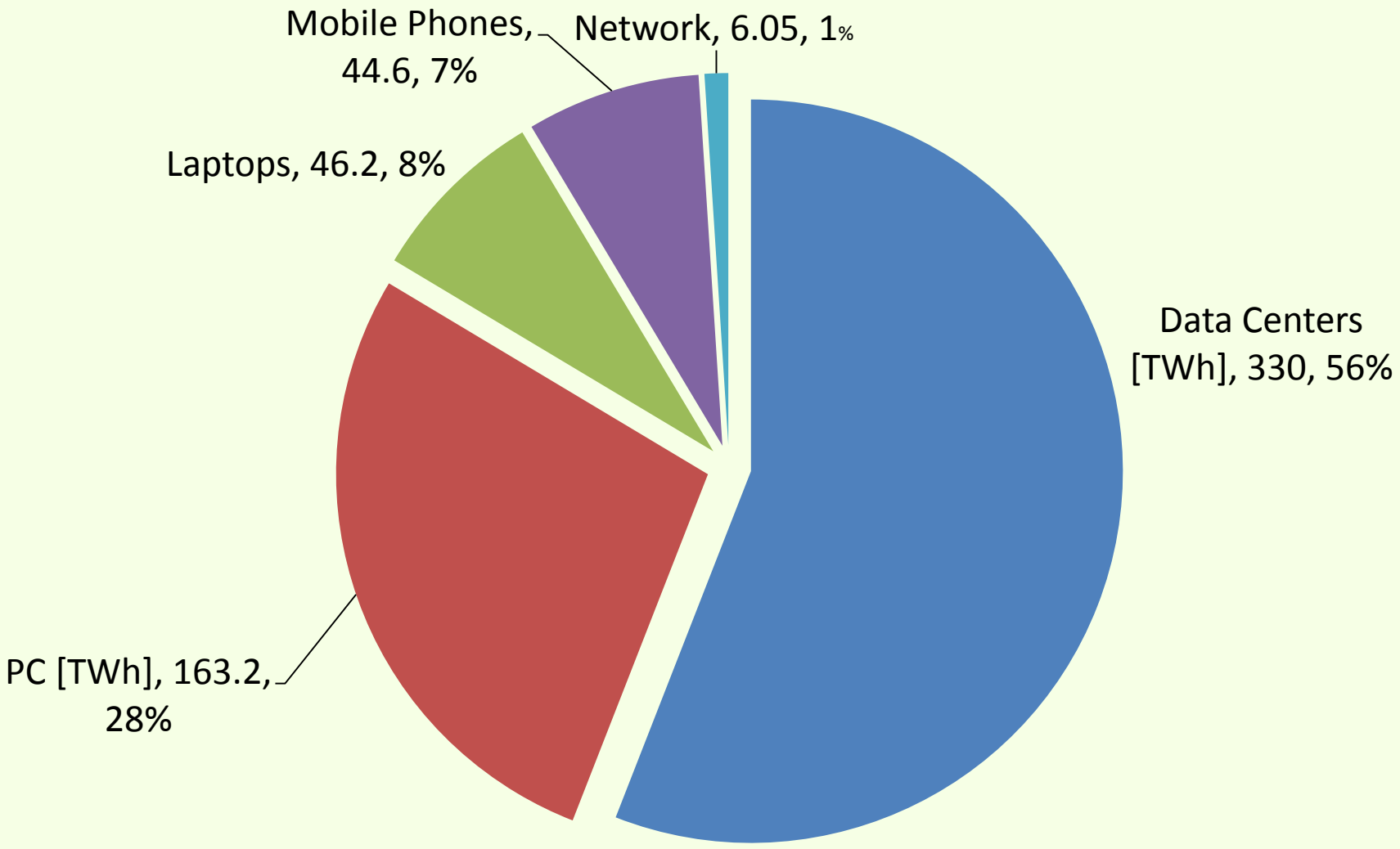


TOTAL: 144000 TWh

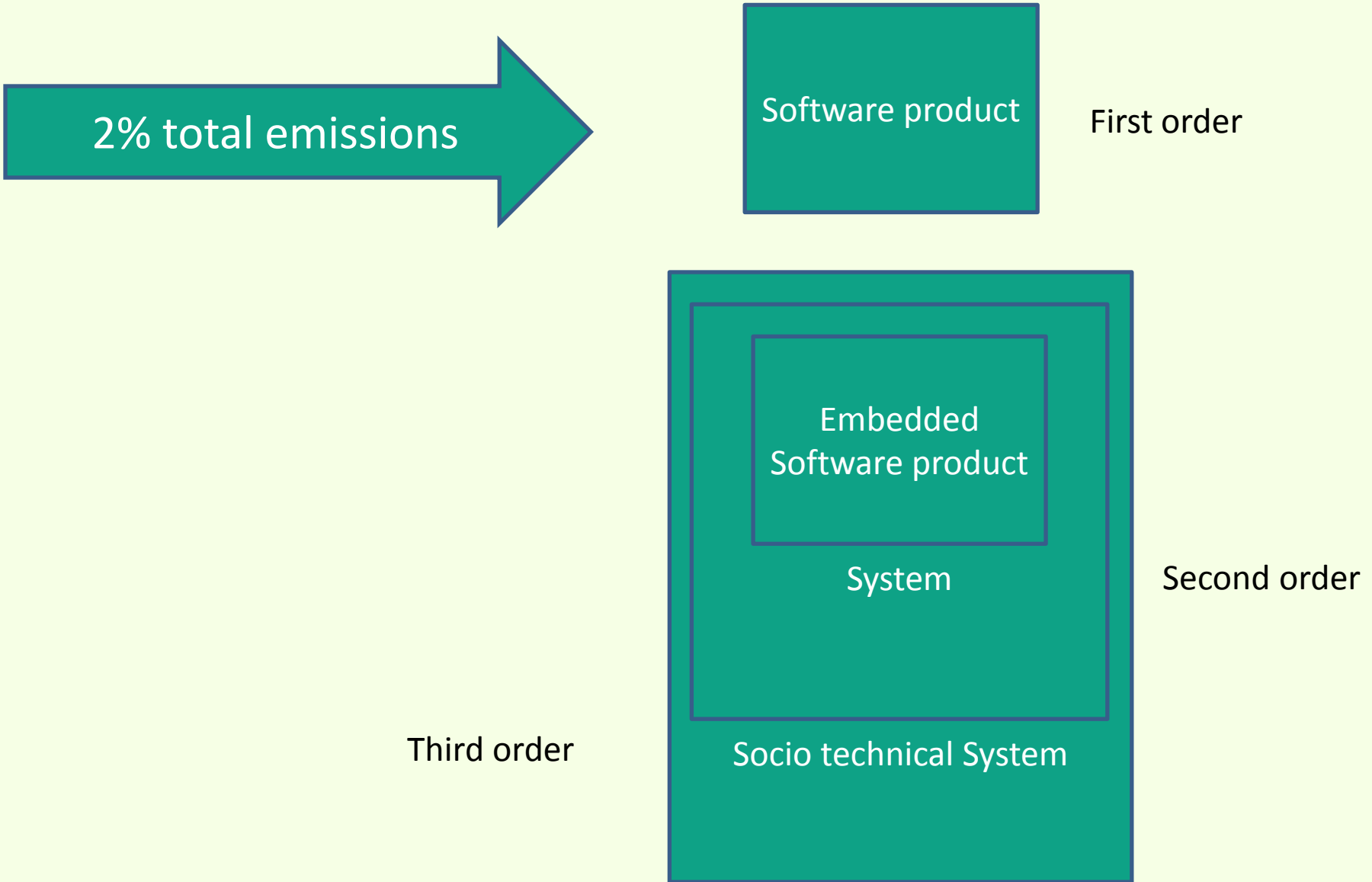
Energy consumption (2007)



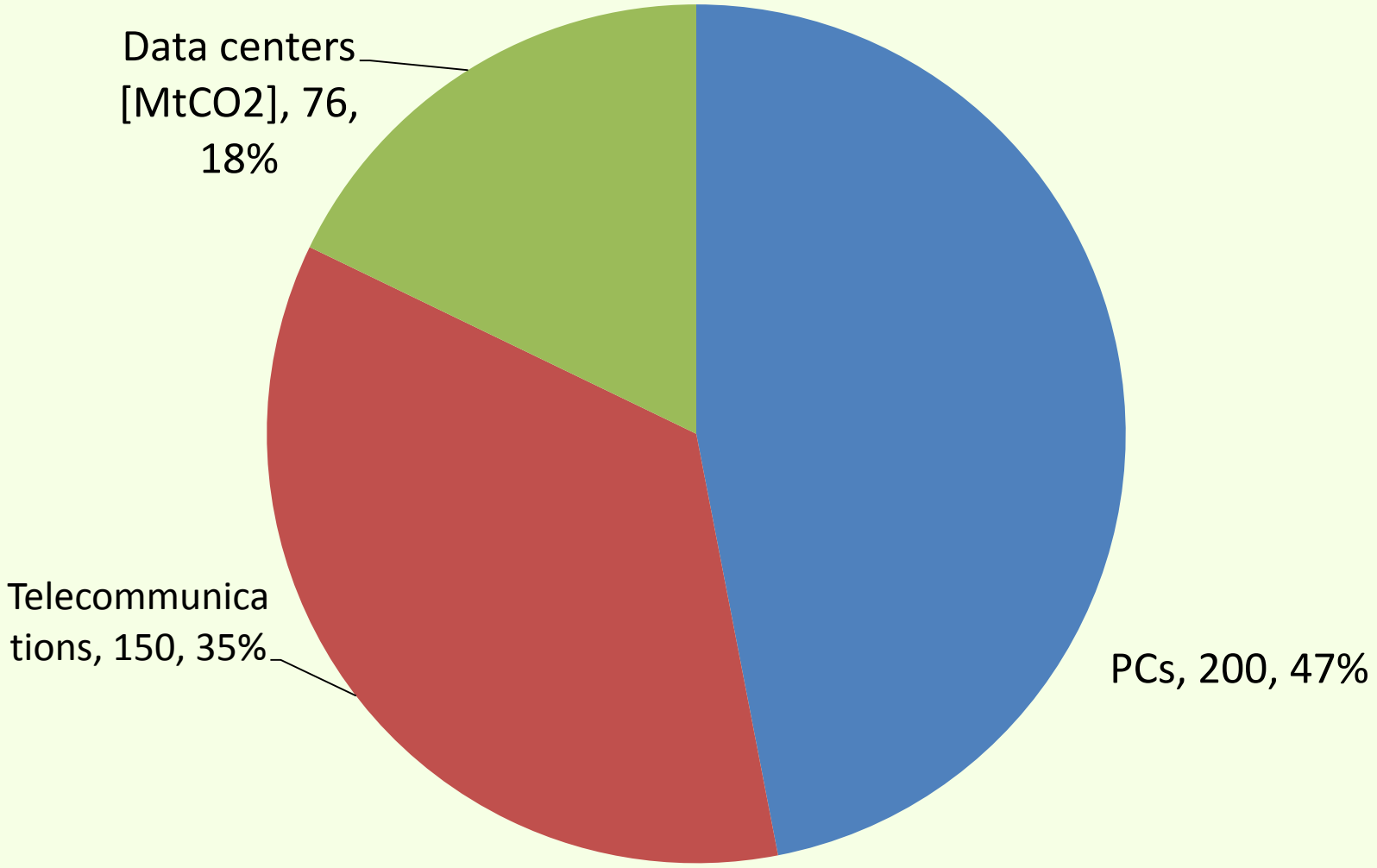
Energy consumption within ICT



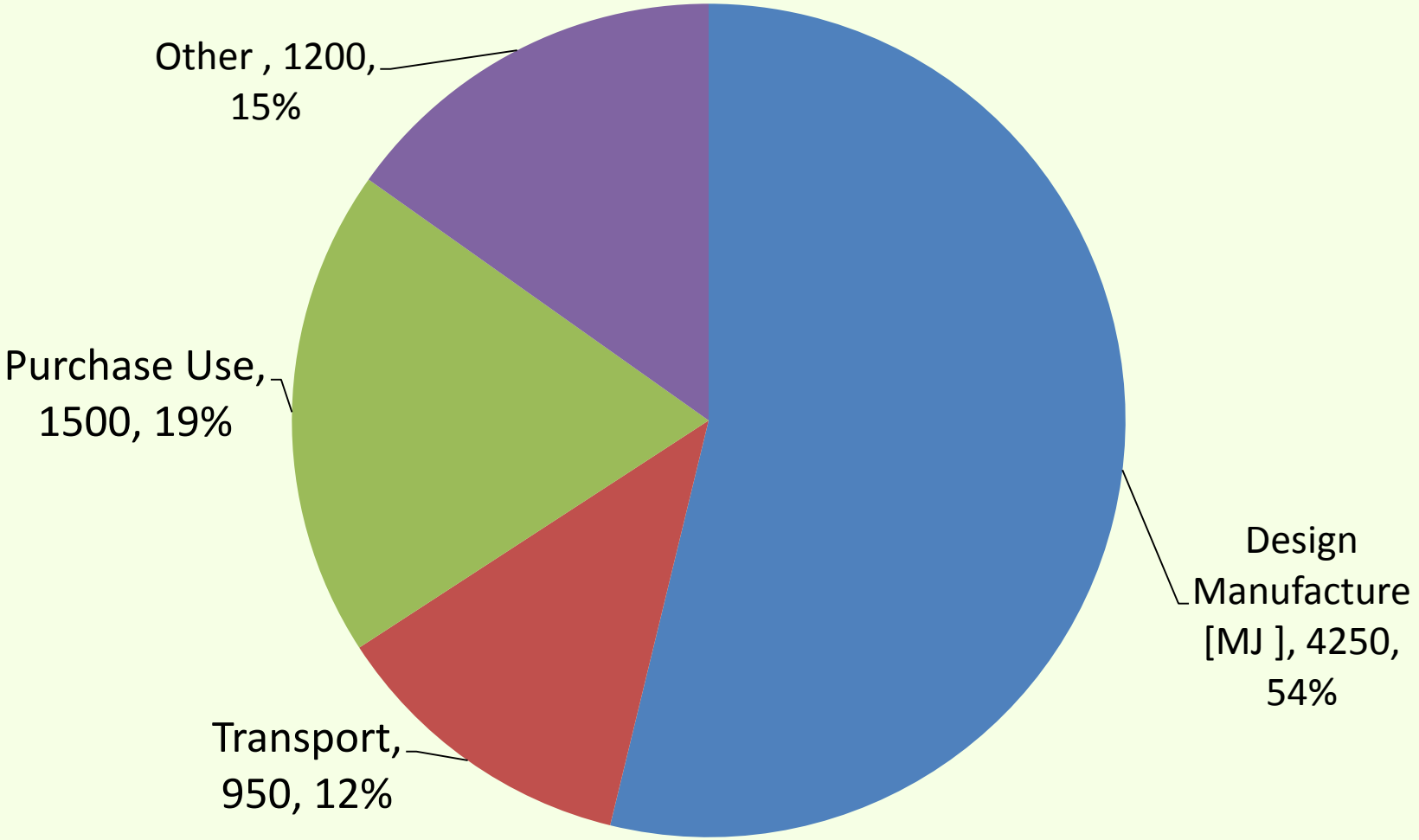
CO_2 emissions



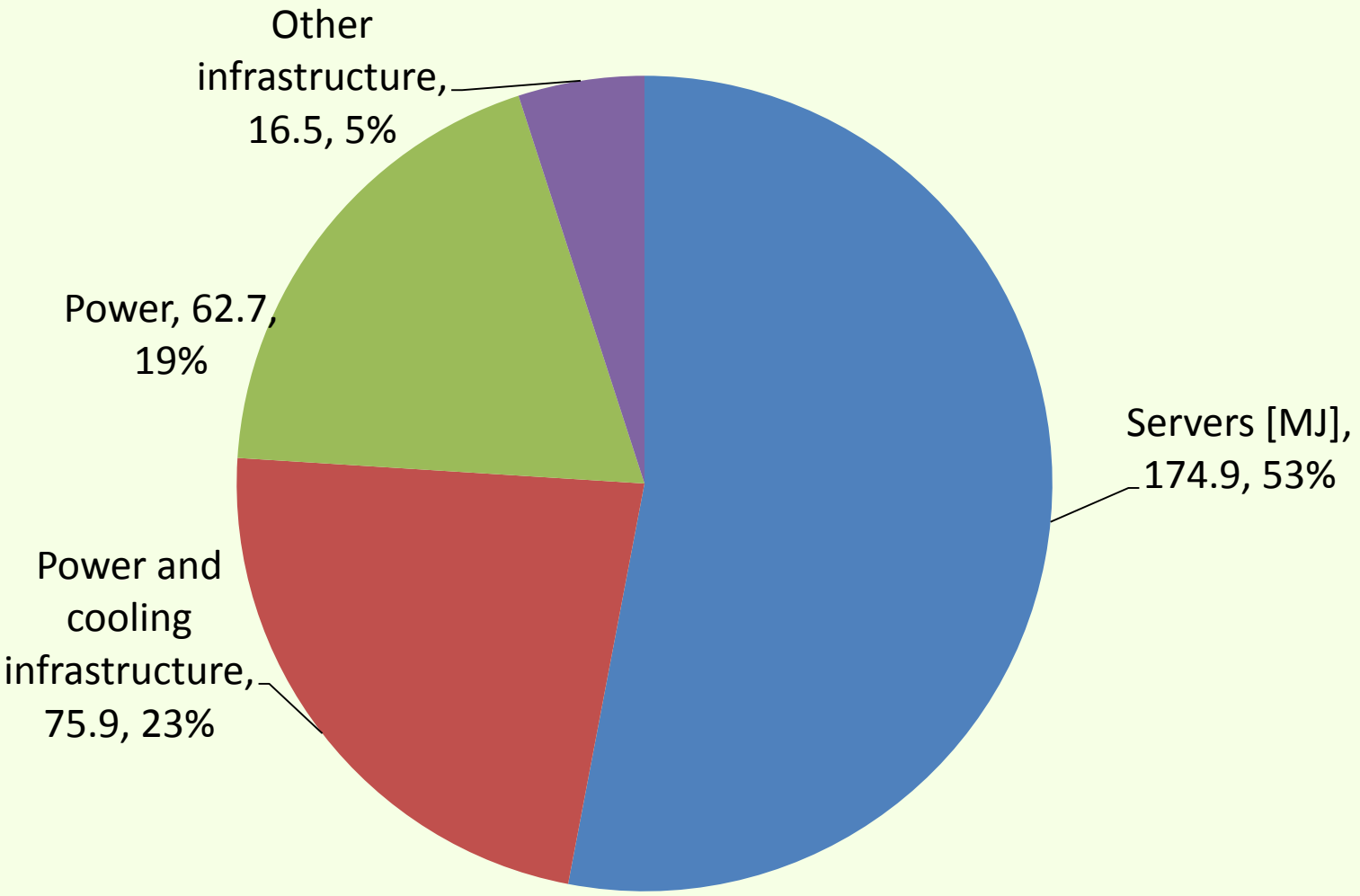
CO2 emissions, ICT



Lifecycle analysis - PC



Energy– data centers



Component analysys

- Desktop PC

  (1) CPU; (2) Hard Disk Drive; (3) Screen & GPU; (4) Network; (5) Memory.

- Mobile phones

  (1) Screen &GPU; (2) CPU; (3) Network; (4) Hard Disk Drive; (5) Memory

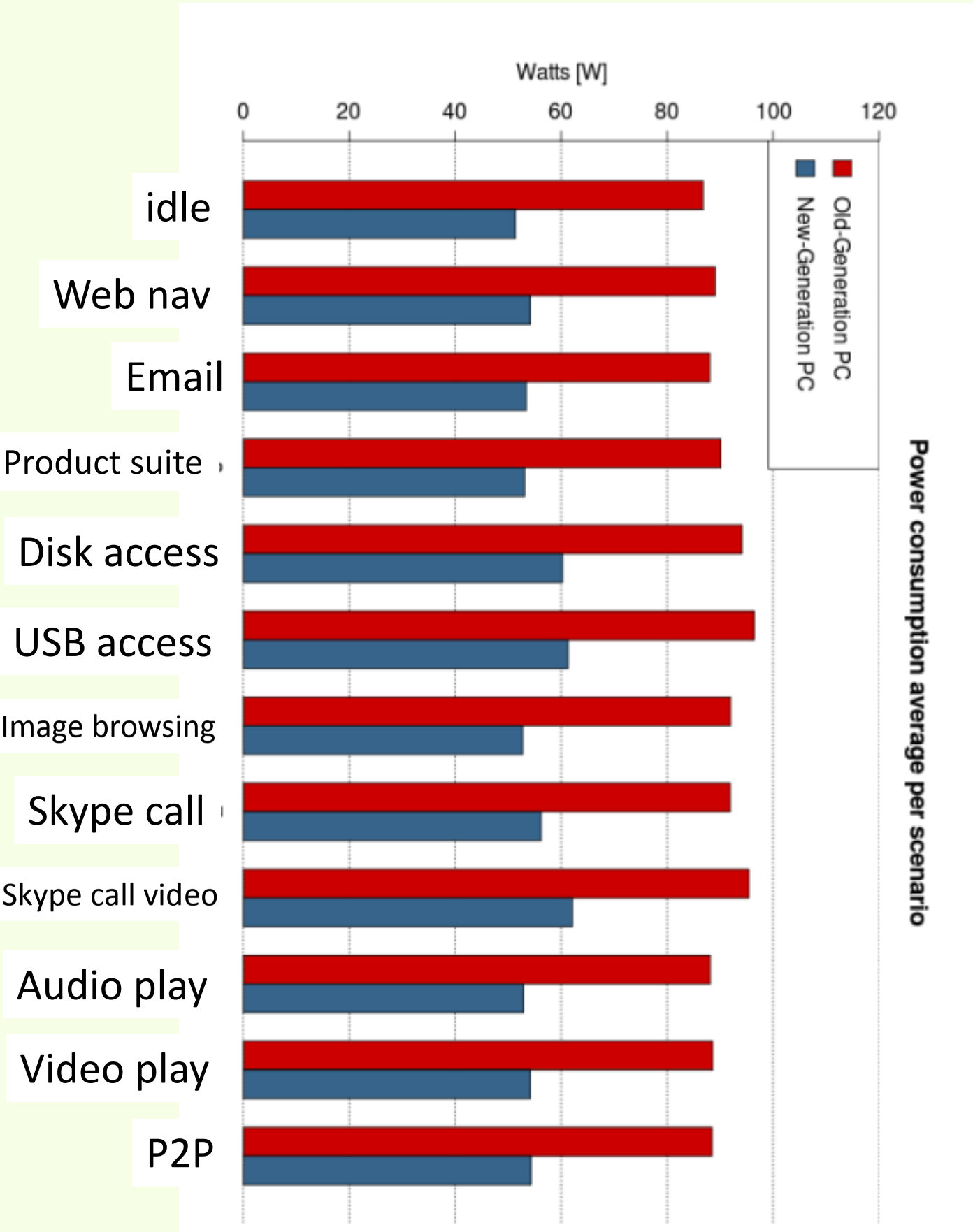
Pang C., Hindle A., Adams B., Hassan A.
What do programmers know about software energy consumption?
IEEE Software

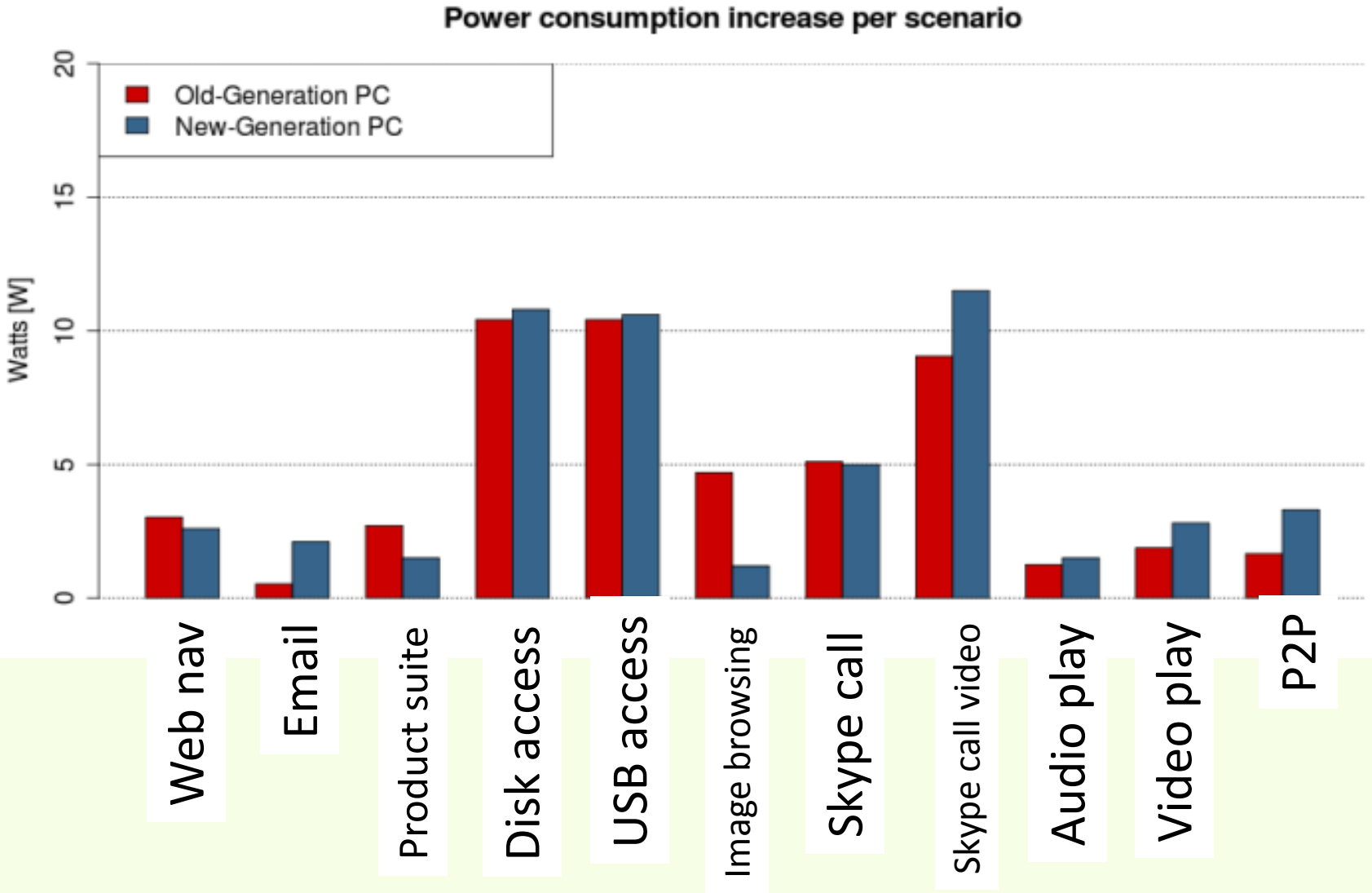
ICT footprint

- Small in %
- Big in absolute numbers
- Increasing trend

- Worthwhile to work for reducing it

Software is energy?


Watts [W]
0
20
40
60
80
100
120
idle
Web nav
Email
Product suite
Disk access
USB access
Image browsing
Skype call
Skype call video
Audio play
Video play
P2P
Old-Generation PC
New-Generation PC
Power consumption average per scenario


Power consumption increase per scenario
Old-Generation PC
New-Generation PC
Watts [W]
0
5
10
15
20
Web nav
Email
Product suite
Disk access
USB access
Image browsing
Skype call
Skype call video
Audio play
Video play
P2P

- Desktop PC: up to 20% increase power consumption from idle
- Small data center: up to 40% increase
- Mobile phones: up to 80% increase

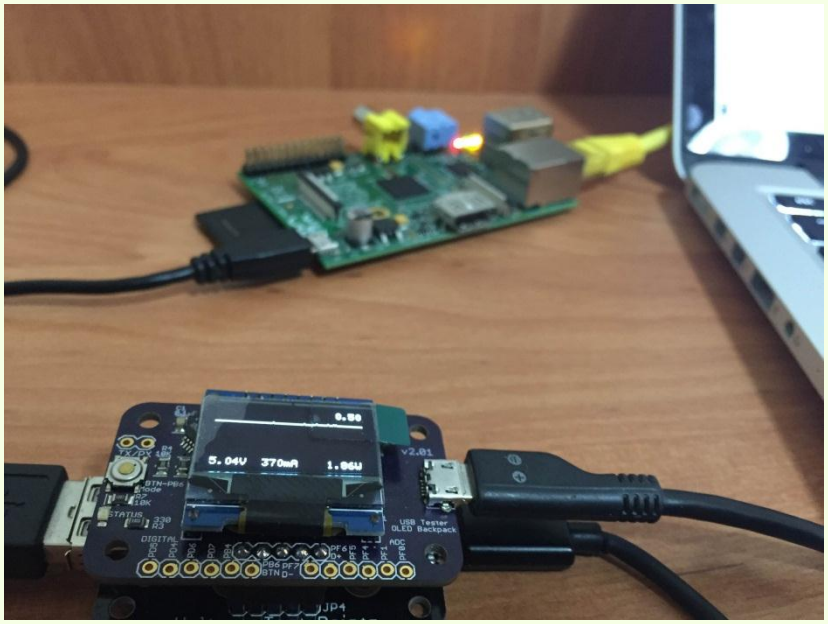
Procaccianti G., Vetro' A., Ardito L., Morisio M., **Profiling Power Consumption on Desktop Computer Systems**

Vetro' A., Ardito L., Morisio M., Procaccianti G. (2011), **Monitoring IT Power Consumption in a Research Center: Seven Facts**.

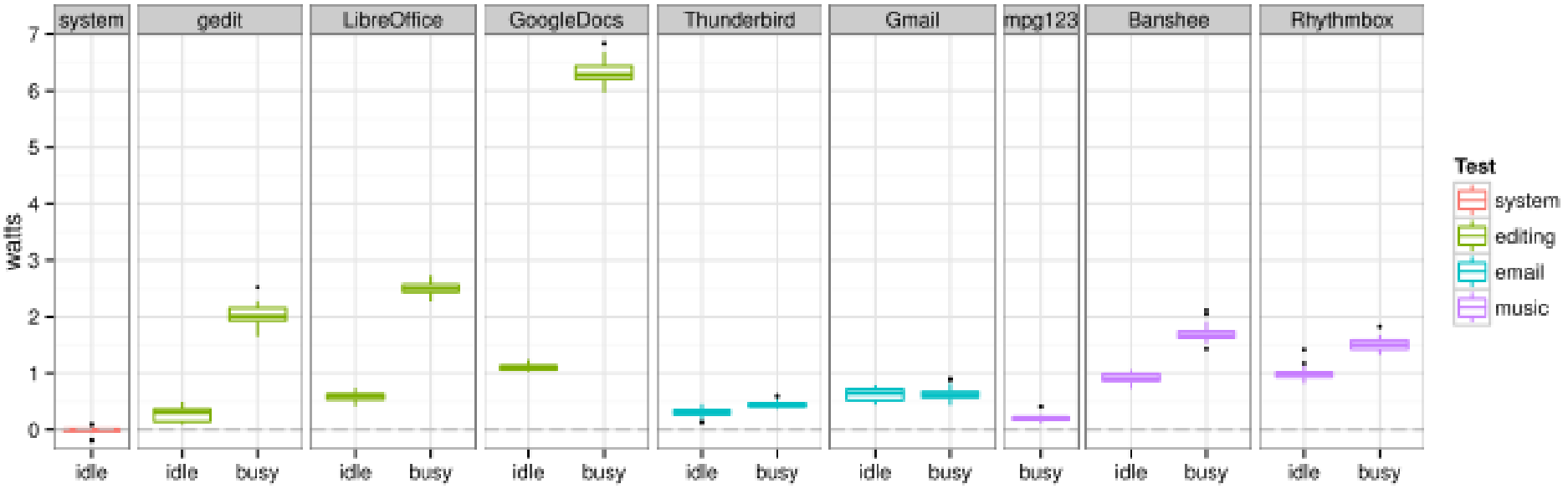
Ardito L., Procaccianti G., Torchiano M., Migliore G. **Profiling Power Consumption on Mobile Devices.**

Measuring workbench

- Instantaneous V, A
- Sampled at 10 - 250Khz
- Repeated sequences, stat analysis



PC, application families



Zhang C, Hindle A., German D

The impact of User Choice on Energy Consumption

IEEE Software

Issues

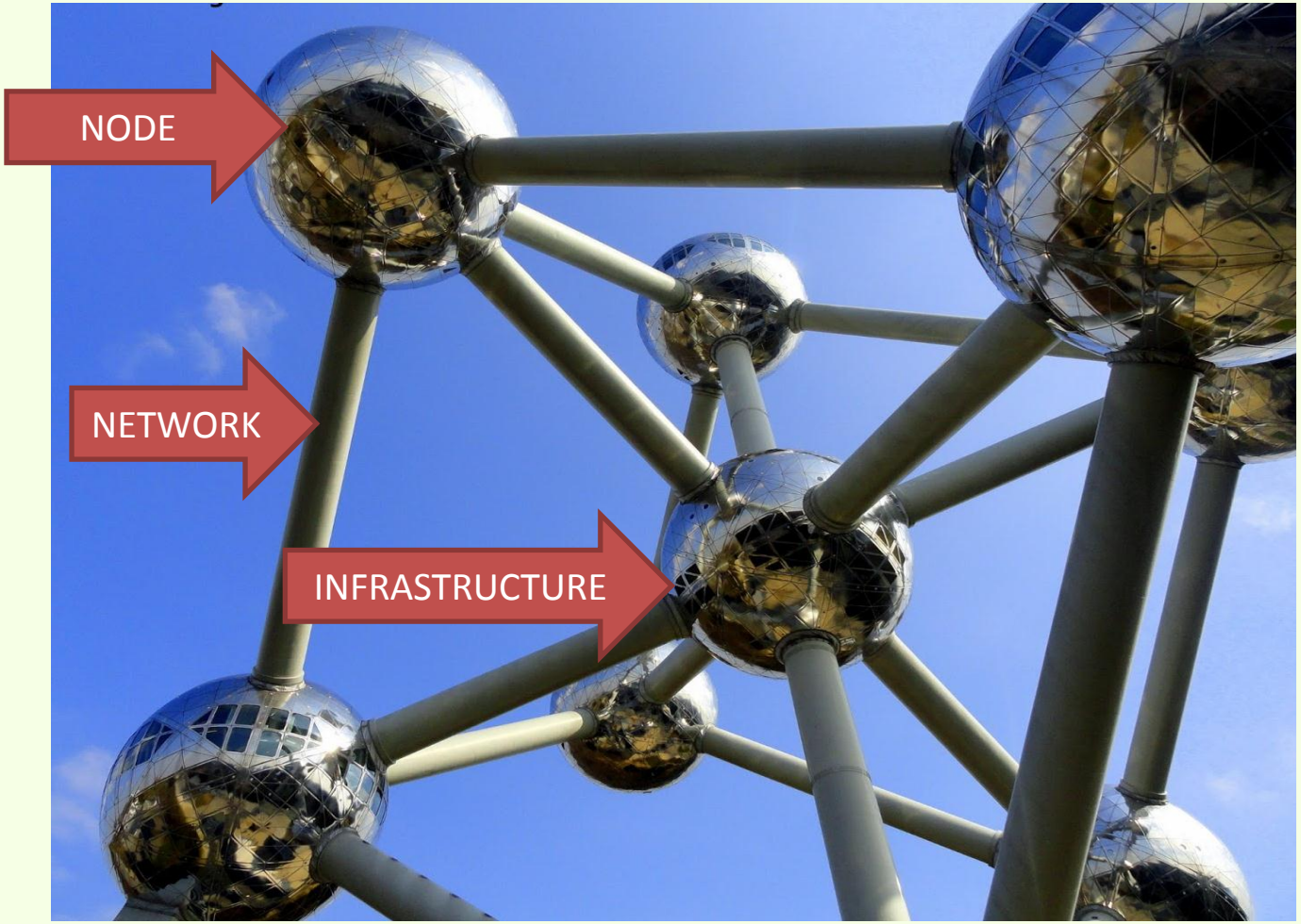
- Definition of scenarios of usage
- Measurement and effect of context
  - (In)dependence of hardware
  - (In)dependence of other applications

Summary

- Small % consumption of ICT, but huge in absolute number, and increasing
- Servers first, then PCs, mobile phones
- In lifecycle, manufacturing matters more
- Application consumption can be measured, and has impact


Guidelines
Facts
Concepts

Guidelines



Guidelines – node /application level

- Efficient UI design
  - To minimize time (energy) to accomplish a task
- Event based programming
  - No polling, no idle resources
- Low level programming
  - Virtual machines, high level programming may be energy inefficient
- Batch I/O
  - Economy of scale. OS can power down IO devices when not used

Guidelines – Node/ application level

- Allocate data / computation where more energy efficient
  - Cfr deploy on cloud
- Data redundancy and migration
  - May reduce energy efficiency
- Adapt/ scale QoS to energy availability
- Use energy models
  - To adapt / optimize behavior of application

Guidelines – Node / app level

- ‘Energy smell’: an implementation choice that makes the software execution less efficient
- Effect <1%

```
int dead_local_store(int x)
{
        int constant_a = x;
        constant_a = 3;
        return constant_a+x;
}

int no_dead_local_store(int x)
{
        int constant_a = 3;
        return constant_a+x;
}
```

Vetro’ A., Ardito L., Procaccianti G., Morisio M. **“A Definition, implementation and validation of Green Code Smells - An exploratory study on an embedded system.”**

Guidelines – Node/OS level

- Provide energy management services / API
  - Cfr energy models for applications
- Optimize use of devices
  - Require collaboration from device drivers / device manufacturers
- Use compiler optimization
- Use only required services and background processes

Guidelines – Node/ hardware level

- Power down / optimize use of peripherals
- Use special purpose hardware
- Use dynamic power management capabilities
  - ACPI
- Devices provide energy consumption data

Guidelines – network level

- Lower data traffic
- Optimize protocols on energy consumption

Guidelines – infrastructure level

- Deploy applications on the cloud
  - Virtualization, less hardware, less consumption
  - Worse response times
- Load balancing
  - Distribute load on resources (CPU, storage ..)
  - Less powerful hardware needed
- Make information about consumption available
  - For adapting energy behaviour

Guidelines - summary

- Adaptation
  - feedback loops on energy /power
  - availability of energy information
  - models for energy behavior
  - scenarios of energy usage
    - Works already at OS – device level, to be extended upwards
- System thinking
  - Allocation of data/computation in function of (system) energy consumption


Roadmap
Guidelines
Facts
Concepts

Roadmap

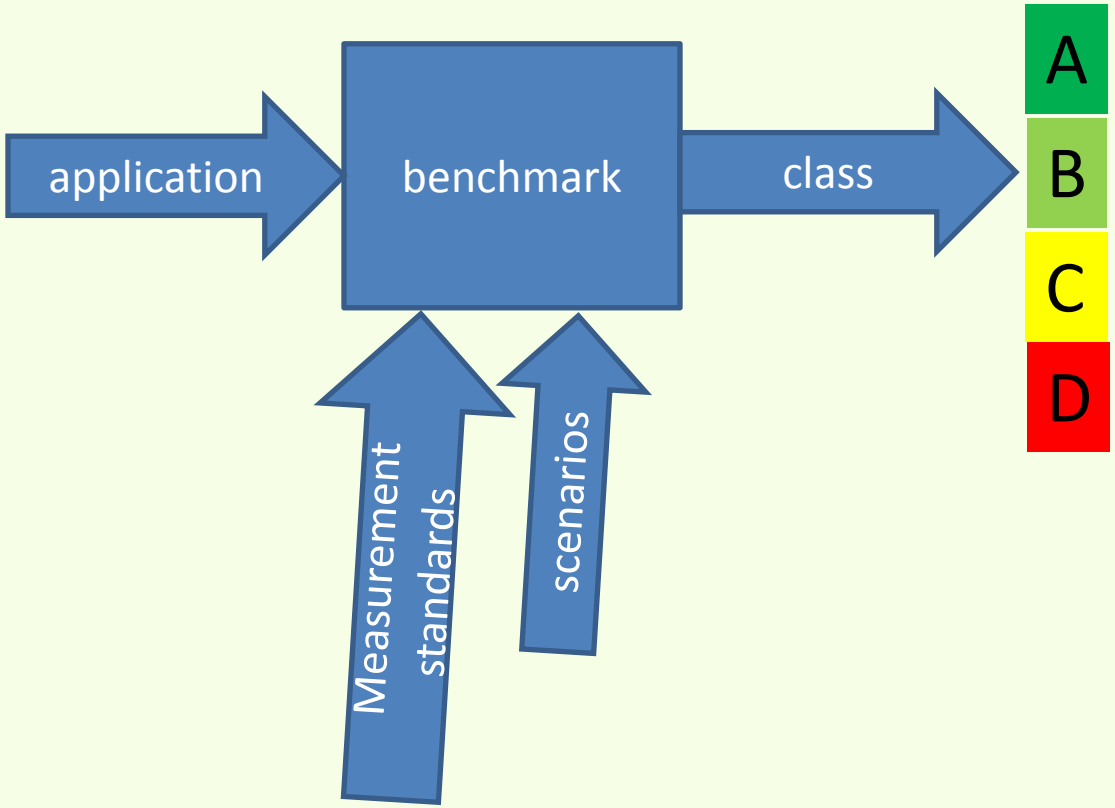
Research goals

- Concepts
  - Greening software or by software???
  - Agreed upon high level model (25010 ..)
- Facts
  - Productivity and efficiency figures
  - Application level
- Guidelines
  - More detailed
  - With context (AKA patterns and antipatterns)
  - With quantified effects

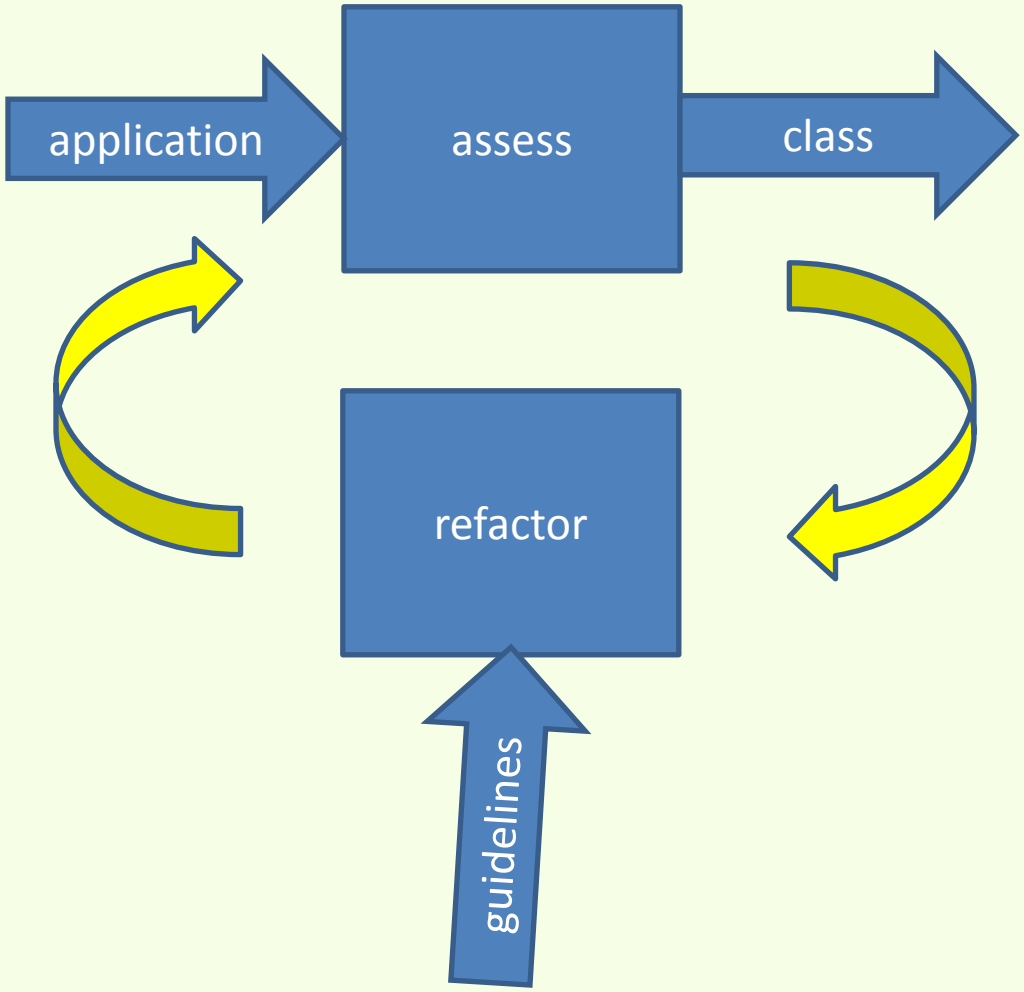
Research goals

- Guidelines
  - Availability of energy / power / usage information at all levels
    - Hardware, OS, application, function
  - Definition and validation of energy models at all levels
  - Self adaptation, at all levels
    - System level, layered
  - Benchmarking

Software Energy Labels



Software Energy Labels



Well..

- Interest in domain is raising
  - Workshops, journals..
- ‘Greenness’ as (yet another) NF requirement (aka safety security) probably to stay
- Lot of work to be done..
  - Define guidelines (good practices for software energy efficiency)
  - Quantify actions (guidelines) in terms of results
  - Trace energy efficiency to process / product