Green Software Engineering – Substance or fad?

Hans Christian Benestad

June 30, 2009

Ph.D. Trial Lecture

UNIVERSITETET I OSLO
[simula . research laboratory]
· by thinking constantly about it ·

What does it mean to go green?

Acid Rain
Effects felt through the food chain

Toxic Waste
Man's poisonous byproducts

To counteract the threats to our environment

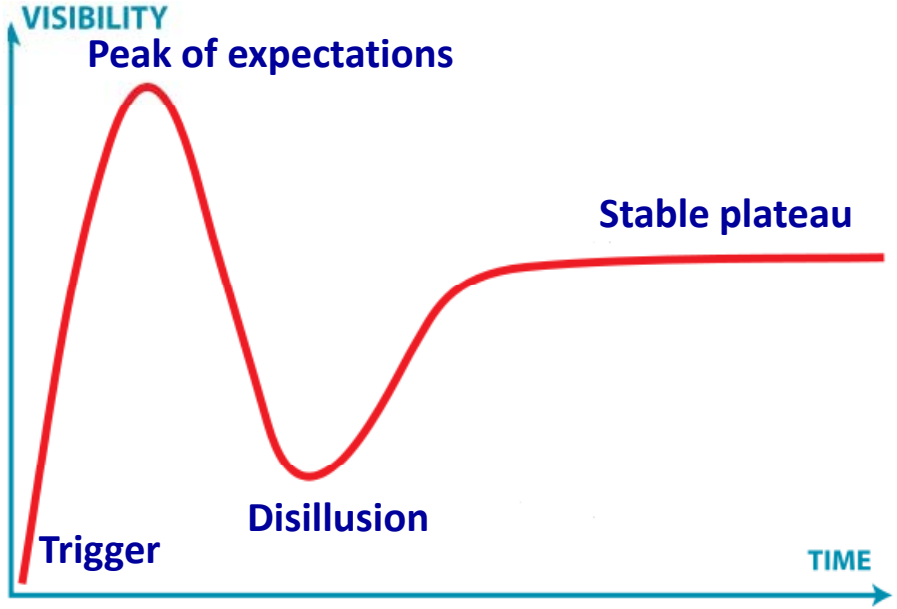
Deforestation
Deliberate by man

Air Pollution
Harming our atmosphere.

Ozone Depletion
Losing the Earth's protective layer

[National Geographic ,2009]

Going green are by many regarded a hype



[Generic hype curve, adapted from Gartner Group]

A hype or a product of a hype is not necessarily without substance

Hypes can be problematic, but are also powerful

Commercially exploited

Hampers nuanced debates

May end in disillusion

Global awareness

Change people's behaviour

A definition of GSE, based on definitions of *Software Engineering* and *Going green*

Software engineering is the application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software…

+

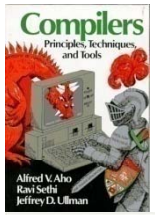
To counteract the threats to the environment

An approach to developing, operating and maintaining software that recognizes and counteracts threats to the environment

How can SE counteract threats to the environment?

Develop Toyota Prius power management software

Reduce CPU cycles generated by SW code



Dispose less hardware

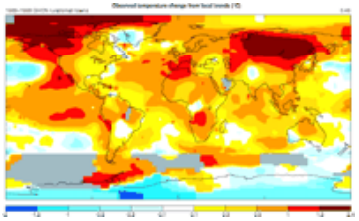
Reduce energy consumption in computing centers

Remove need for printouts and post

Downloadable books

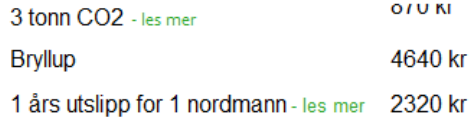
Cloud computing

Software to predict climate changes



Develop video conferencing
Use video conferencing

Websites for trading CO2 quotas



Use OS with better power management

Optimized air traffic control

SW tools for distributed development

Can software engineers contribute to a greener planet?

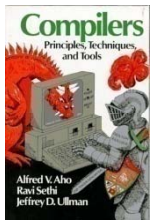
Software-intensive technology plays a key role in counteracting threats to the environment




Should development of such technology be denoted Green Software Engineering?

"An approach to developing, operating and maintaining software that recognize and counteract threats to the environment"

Reduce the environmental impact of the software products and processes in themselves

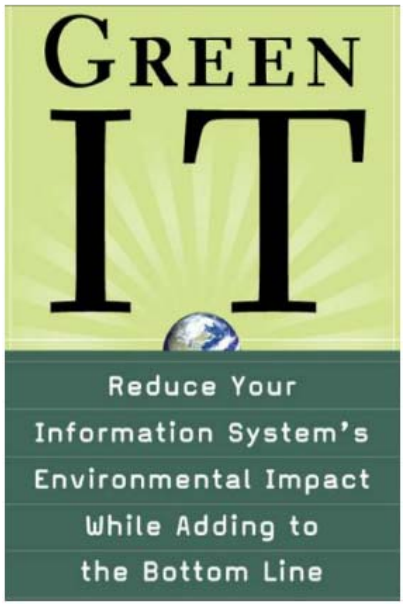


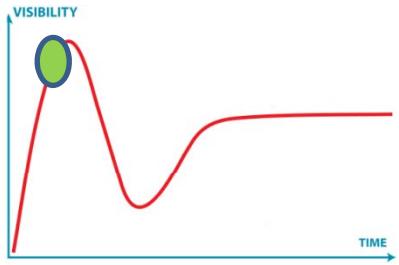




Green software engineering is one enabler for *Green IT*

The ICT industry emits 2% of global CO2 emissions, most resulting from the power consumption of PCs, servers and cooling systems. [Gartner Group 2007]





improve energy efficiency

lower greenhouse gas emissions

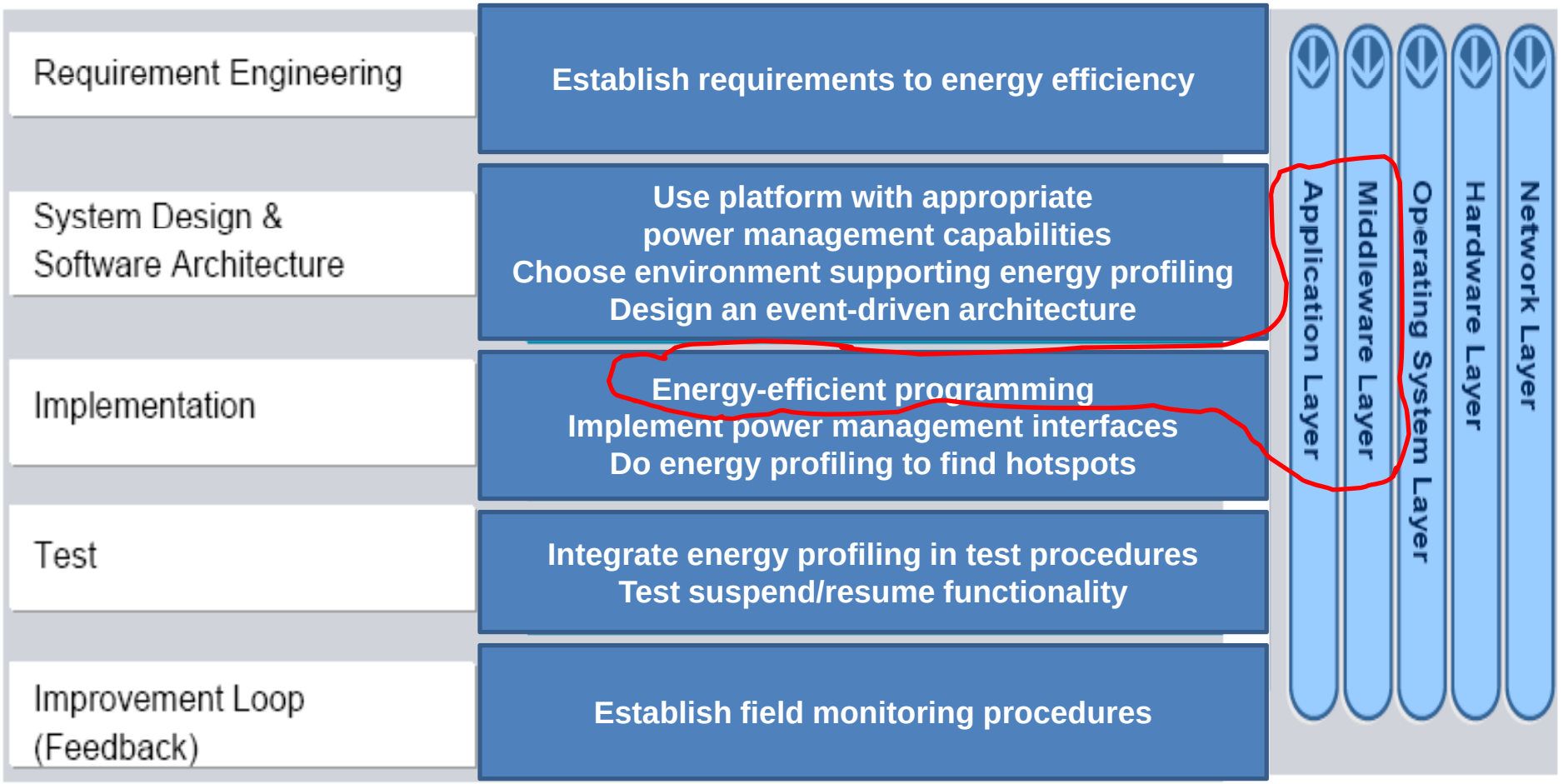
use less harmful materials

encouraging reuse and recycling

[Murugesan, IEEE IT Professional 2008]

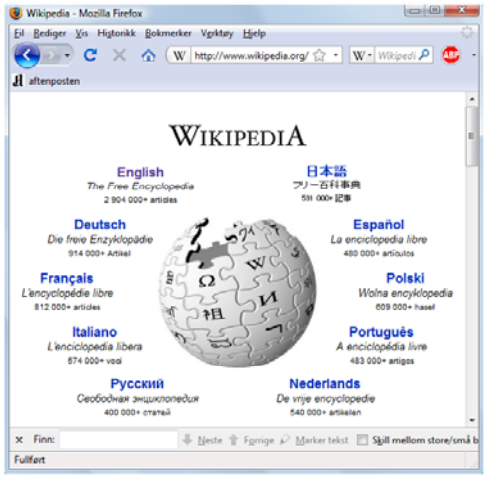
What can software engineers do?

Energy efficiency crosscuts the system architecture and the software engineering process



[GSE, Gerald Kaefer, Siemens AG, presented at ICSE 2009]

Coding style is not always irrelevant to energy efficiency



13 http requests to handle pictures

Assume 100 million views per day, and conservative estimates of other relevant parameters

Annual CO2 emissions of 500 kg

A simple change in the code would eliminate the emission

```
<FilesMatch "\.(png|gif)$">
Header set Expires "Thu, 15 Apr
2010 20:00:00 GMT"
</FilesMatch>
```

Creating software that fits in a *cloud computing* model

Enterprise X users

Enterprise Y users

....

Enterprise N users

Today's computing

Scaled for max usage
Available resources are typically idle

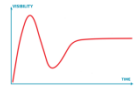
Enterprise X users

Enterprise Y users

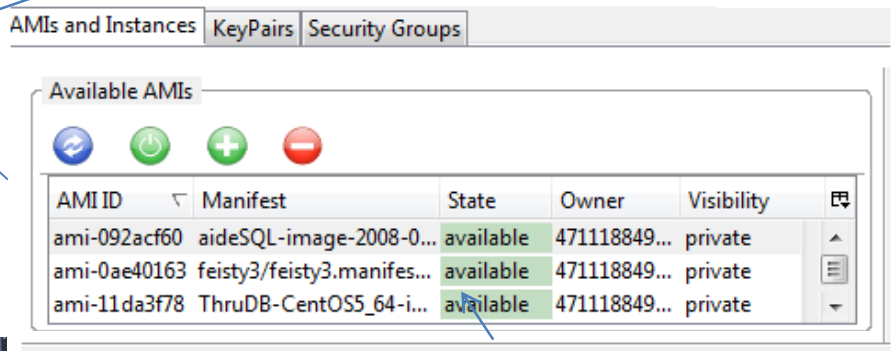
....

Enterprise N users

Cloud computing



Internet



Dynamically scalable infrastructure services

Overall, unbeaten resource utilization

In design of embedded systems, energy consumption is a critical factor.

Thermal design

Power supply design

Weight

Cost

Operating time

Models to predict and optimize energy consumption are well established

Adapt to IT systems?

Engineering more eco-friendly business processes

Challenge any need for printing, including pdf generation

Two examples
Topic for more research?

Limit the need for transportation of humans and objects

Resolving and trading off conflicting goals are core challenges of software engineering

Cost vs. functionality

Efficiency vs. usability

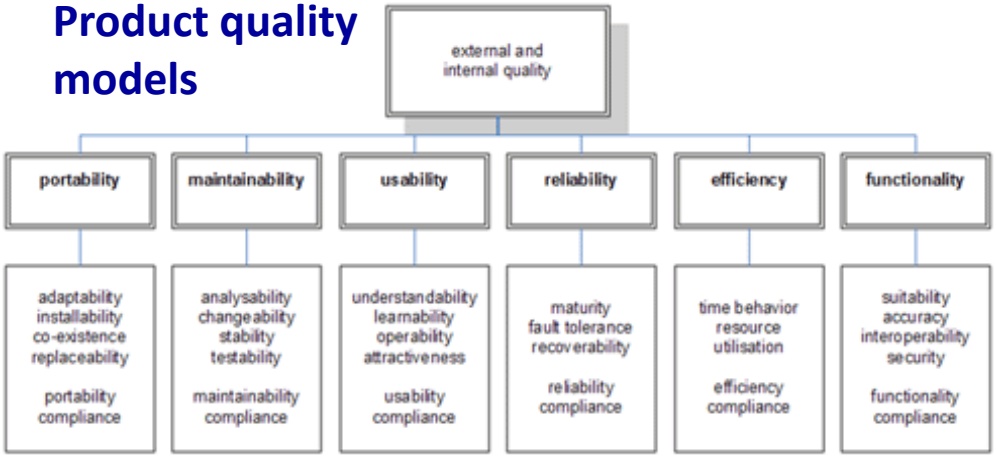
Deployment architecture vs. evolvability

Going green vs. *

Going green can be viewed as new generic goal for SE, possibly in conflict with other goals

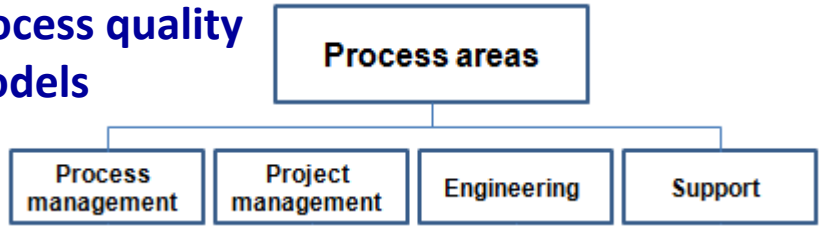
Integration of green qualities into existing SE frameworks?

Product quality models



ISO 9126

Process quality models

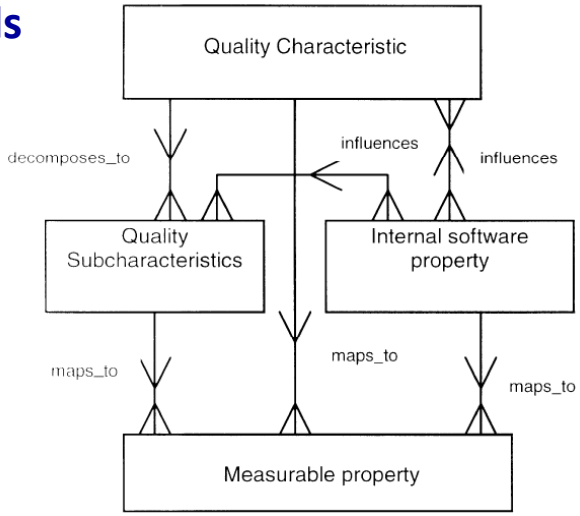


CMMI

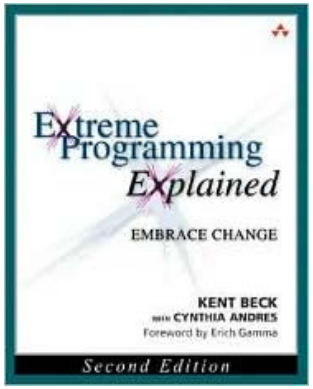
S3M

ISO 12207

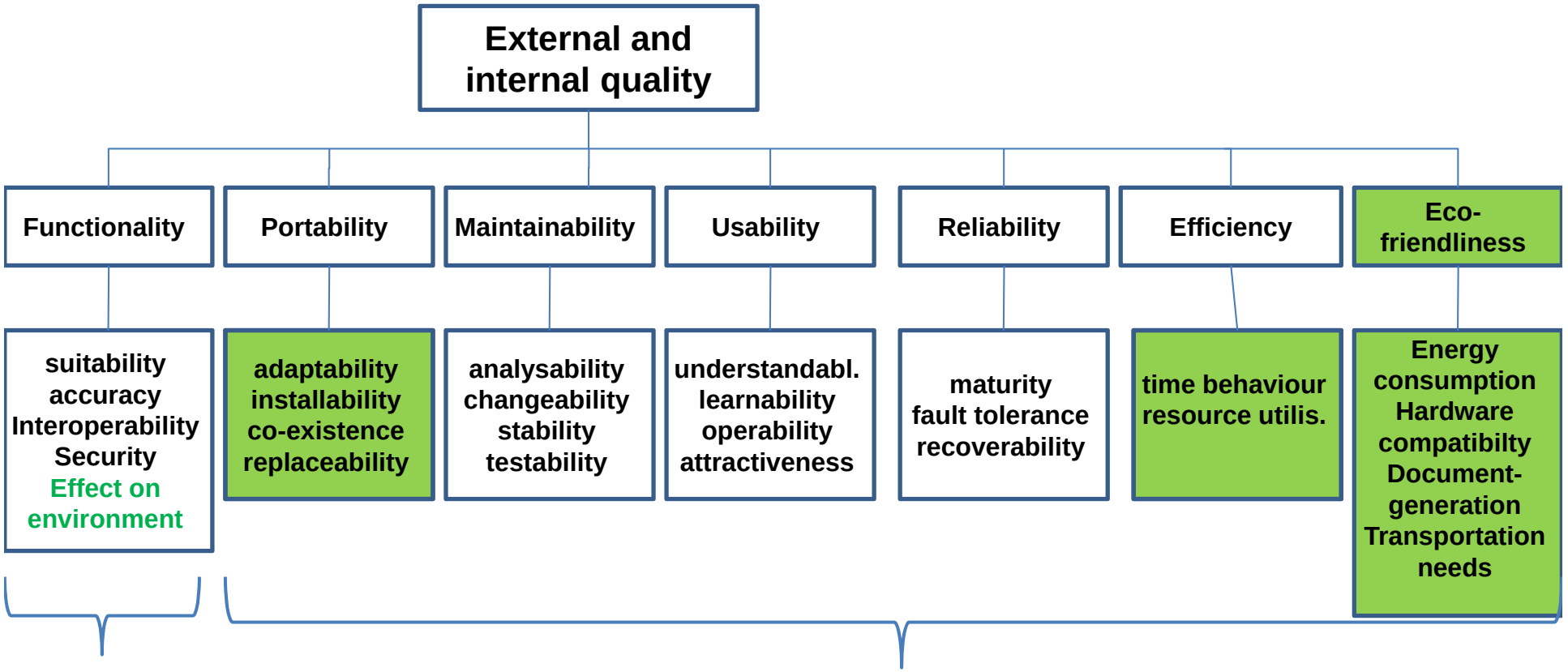
Meta quality models



SE Methods an practices

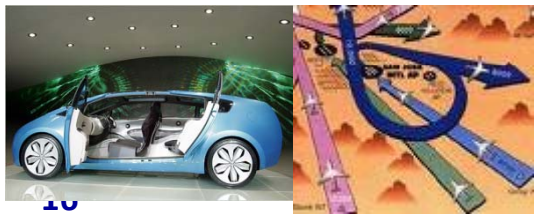


Integrating eco-friendliness into product quality models

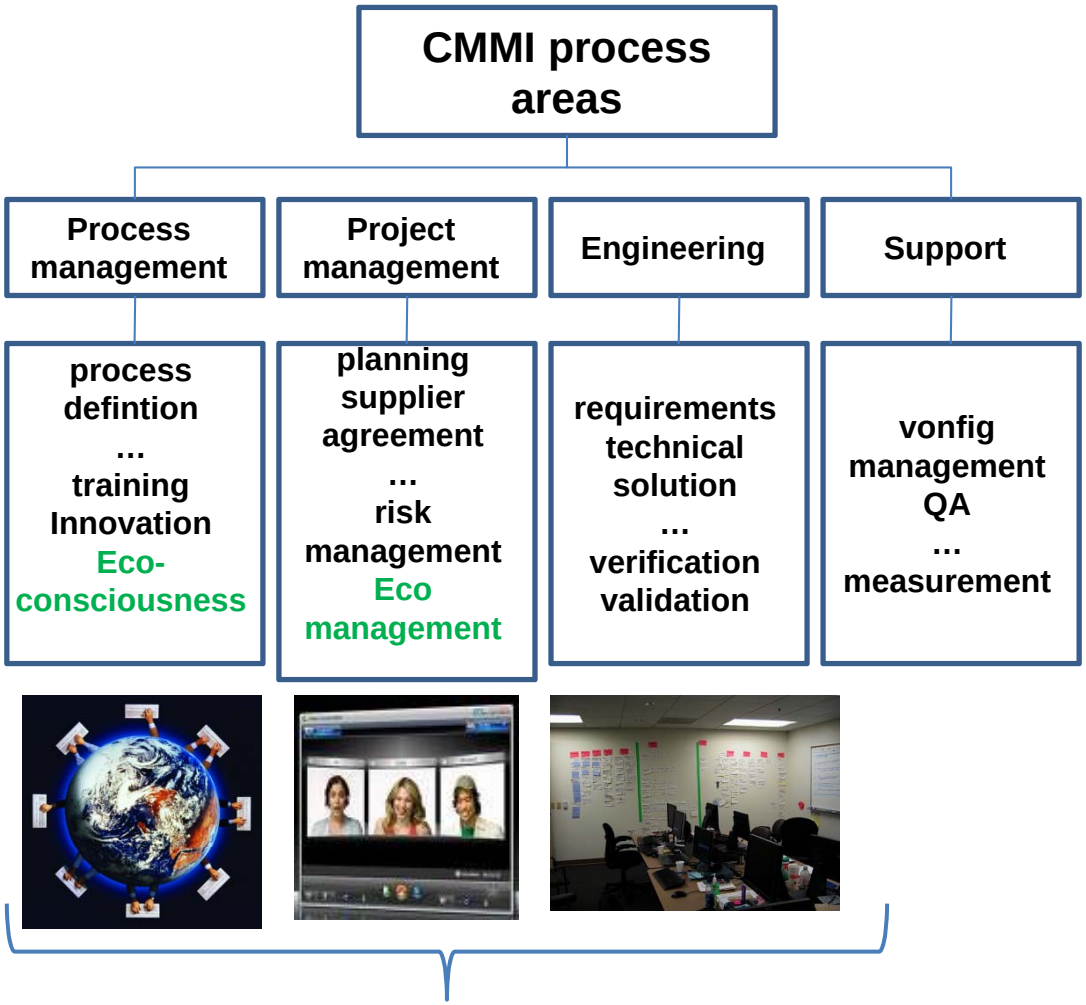


Environmental effect part of functional design

Environmental side-effects of the product



Integrating eco-friendliness into software process models



Working more environmental friendly

CMMI

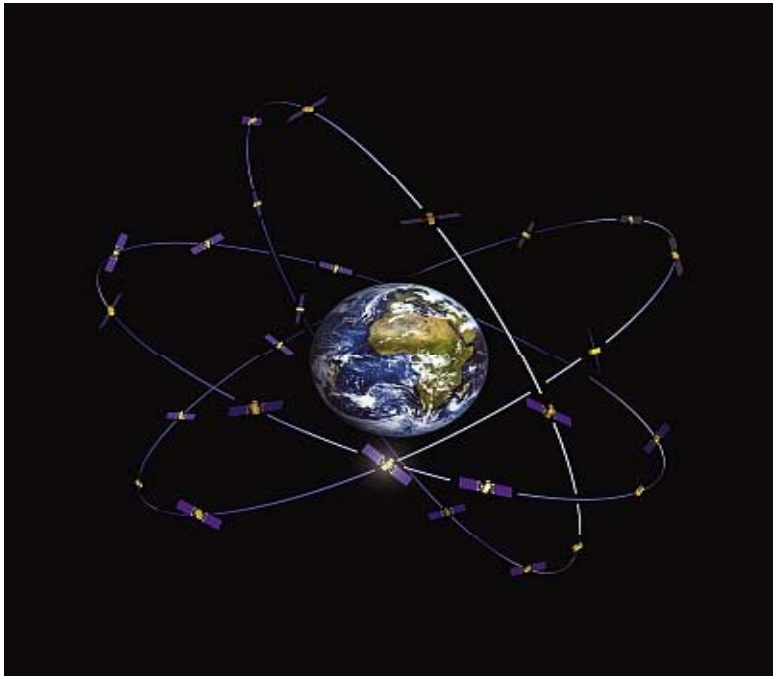
Or development methods

”Agile values”

Individuals and interactions	over	Process and tools
Working software	over	Comprehensive documentation
Customer collaboration	over	Contract negotiation
Responding to change	over	Following a plan

Going green over **Neglecting the environment**

Two prospects for the future



Building a giant cloud computing centre in the Norwegian mountains*

Rich access to cool and secure mountain halls

Stable political climate and power supply

99% of our electricity is hydroelecticity

Financial strength

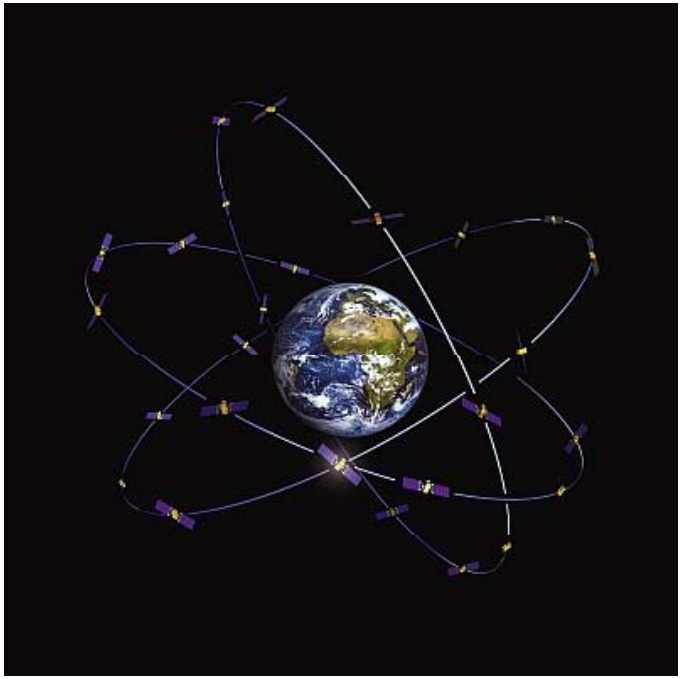
Would this make *the cloud* an attractive option for the largest IT users (public sector, finance) worldwide?

*[IKT-Norge/Teknisk Ukeblad]

Improved global navigation satellite systems will provide dependable and accurate positioning services

Right human or object at the right time and place

Small and inexpensive receivers

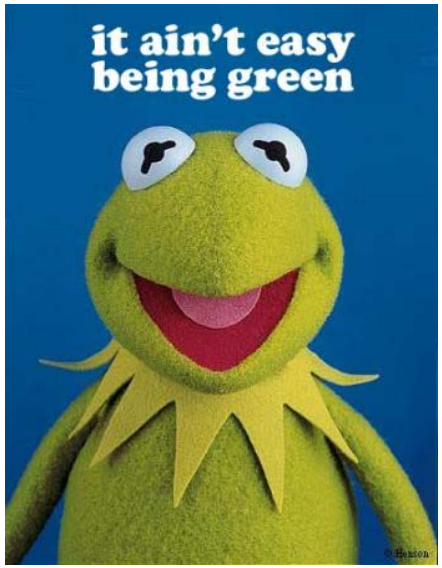


Opens a huge market for positioning-based services, most of them software intensive

New challenges for software engineering?

A new sub-profession for software engineers?

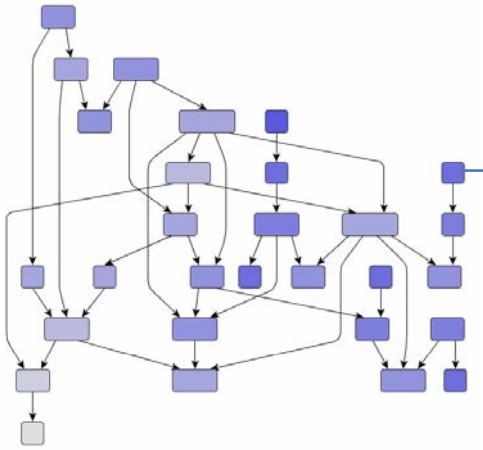
Conclusions



Software engineers have immense opportunities for counteracting environmental threats

Green software engineering is best understood as elements within existing product and process frameworks

…and research opportunities



A better taxonomy for green elements in SE

A framework for analyzing costs and benefits of green elements in SE

Energy consumption models for IT systems

Industry survey of current and future acceptance of green elements in software engineering