Software and systems engineering

Paulo Borba
Informatics Center
Federal University of Pernambuco

pauloborba.cin.ufpe.br

What is software and systems engineering?

Engineering is…

the application of science to the design, building and use of machines, construction, etc.

The Concise Oxford Dictionary

System =
Software +
People + Data +
Hardware
(computers, sensors,
phones, watches,
drones, etc.)

Our society depends on software systems

They are the basis of essential services, communication, entertainment, etc. and drive innovation

Every year, billions of lines of code (LOC) are created or modified

Volvo car > 50 MLOC
A380 > 100 MLOC

Software and systems
engineering is concerned with…

Quality
and
Productivity

Professional software and systems development

http://www.home-dzine.co.za/diy/diy-doghouse.htm

http://silviarangel.wix.com/fotografa

It's not about small versus large

http://transmissionsmedia.com/the-inexplicable-precision-in-the-construction-of-the-great-pyramid-at-giza/

Most software today is very much like an Egyptian pyramid with millions of bricks piled on top of each other, with no structural integrity, but just done by brute force and thousands of slaves.

Alan Kay

http://en.wikipedia.org/wiki/Alan_Kay

unprofessional | ʌnprə'fɛʃ(ə)n(ə)l |

adjective

below or contrary to the standards expected in a particular profession

focus
on how to do it?
vs
on how to do it right?

Should we be concerned that we might be viewed as an over-paid, over-privileged elite that does not care enough about the damage that our work can cause?

Be Gracious. Leon J. Osterweil.

https://dl.acm.org/citation.cfm?id=3203100

Ultimately, we need to assure ourselves and our society that our software has been made as sound and robust as feasible so that failures are not attributable to our own carelessness, recklessness, or laziness.

Your Software Dwells in the House of Tomorrow, Too.
Leon J. Osterweil.
http://doi.acm.org/10.1145/3041765.3041769,

So our main goal is…

- Software and systems **quality**
- Software and systems development and operation **productivity**
  - costs and deadlines

Software and systems engineering

Paulo Borba
Informatics Center
Federal University of Pernambuco

pauloborba.cin.ufpe.br

What is the software crisis?

Practical impact of quality and productivity

- Companies competitiveness
- Better and safer products, smaller costs (long term)
- Attraction of new companies to local ecosystems
  - investments, more taxes

Not always achieved,
almost never easy!

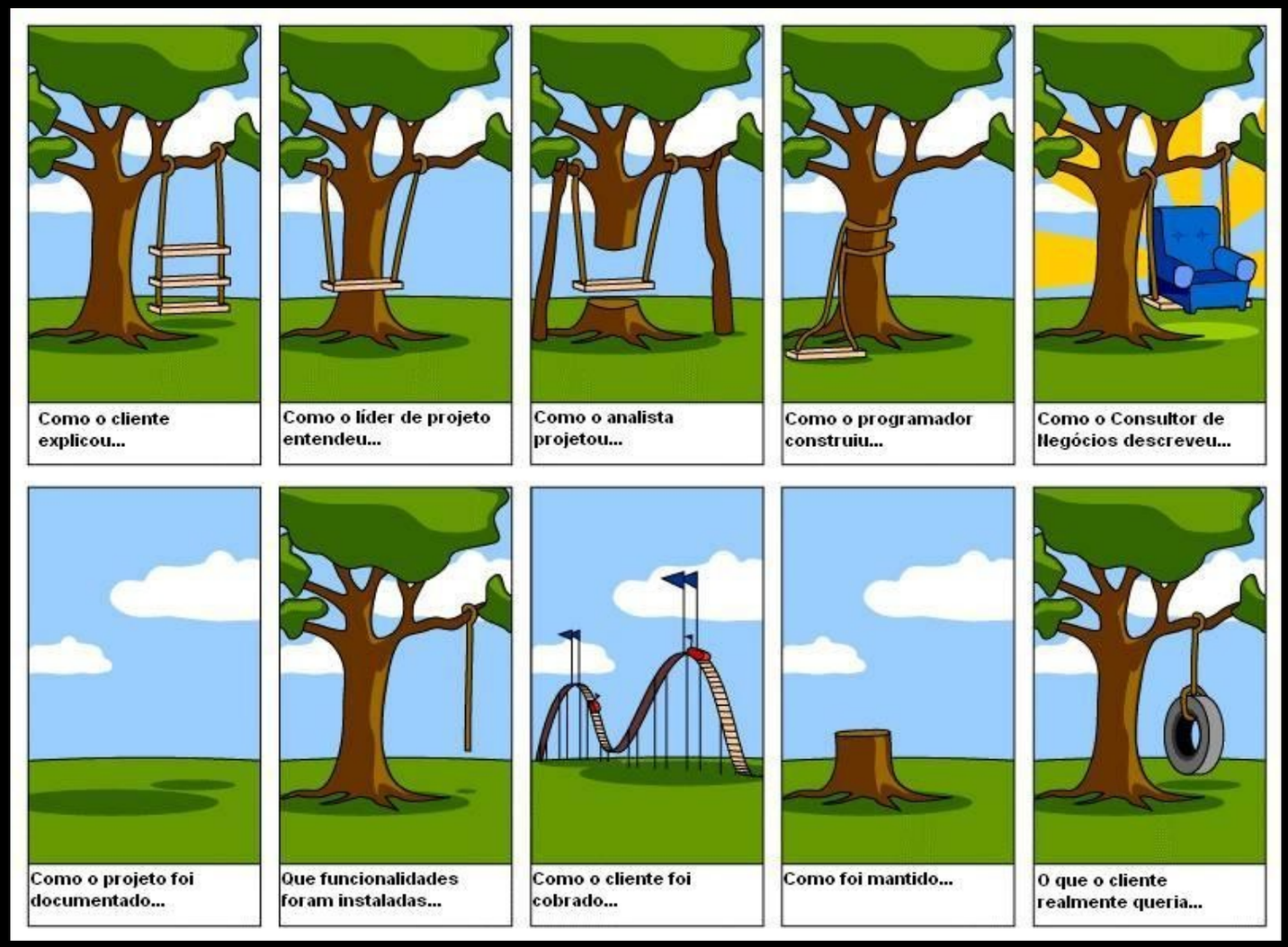
Software development issues, (crisis? since 1968!)

- Project cancellations
- Development time and cost go well beyond the estimative
- Systems do not work as planned
- Difficult reuse and maintenance

No silver bullet!

Essential causes of software issues

- Increasing systems complexity
- Formalization difficulties and costs



Accidental causes of software issues

- People lack skills and experience
- Poor process and practices
- Lack of proper languages and tools
  - little synthesis
- Weak organisational structure
  - poor management
  - conflicts
  - conflict ethics and business values
- Too many project constraints

Teams that don't achieve share these deficiencies

Handling the crisis

- 50: compilers, OS
- 60: SE (management, formal, testing), OO, databases
- 80: no silver bullet, configuration management
- 90: maturity model, processes, patterns
- 70:
  - information hiding/ modules
  - top-down, stepwise refinement
  - incremental builds (today's MVP)
  - inspections
  - requirements verification and validation

Software and systems engineering

Paulo Borba
Informatics Center
Federal University of Pernambuco

pauloborba.cin.ufpe.br

What is software

quality?

Software quality factors, focus on ethics and business values

- ✓ Reliability
  - ✓ Correctness
  - ✓ Robustness
- ✓ Extensibility
- ✓ Reusability
- ✓ Compatibility (backward)
- ✓ Portability

More factors, internal and external

- ✓ Performance
- ✓ Usability
- ✓ Scalability
- ✓ Flexibility
- ✓ Integrity, privacy and security
- ✓ Fault tolerance
- ✓ Safety

For example, why would those factors be important for a store automation system?

Software and systems engineering

Paulo Borba
Informatics Center
Federal University of Pernambuco

pauloborba.cin.ufpe.br

What is software productivity?

Productivity aspects, assuming constant functionality and quality

- Reduced development cost
  - Consuming company wishes to invest little in software
  - Producing company should offer “inexpensive software"
- Reduced development time
  - Quick support and attention to market needs

“Inexpensive software”

Not only a result of lower development costs, but also of the cost distribution among a number of clients

Reuse, extensibility and flexibility are important factors for achieving such distribution

Trade-offs between quality and productivity

- Investing too much in quality can reduce productivity in the short term
- Neglecting quality can impact productivity even in the short term
- Professional ethics should not be part of the trade-off
  - careful with software that is ethically non-neutral

To obtain software
quality and productivity,
teams have to deal with
those issues in a
professional way!

Take notes,

now!

To do after class

- Read Uma longa noite aprendendo, and Chapter 1 from the textbook
- Read Software Engineering Code of Ethics and Professional Practice
- Watch “Na rota do dinheiro sujo”, episode “Emissões mortais”, and Software powers the world
- Choose your team
- Subscribe to the course calendar
- Study questions from previous exams

Questions from previous exams

- Indique dois fatores de qualidade de software e explique porque eles são importantes.
- Indique quais dos seguintes termos correspondem a fatores de qualidade de software: corretude, robustez, produtividade e extensibilidade. Defina e explique brevemente porque cada um dos fatores que você indicou é importante.

Software and systems engineering

Paulo Borba
Informatics Center
Federal University of Pernambuco

pauloborba.cin.ufpe.br