

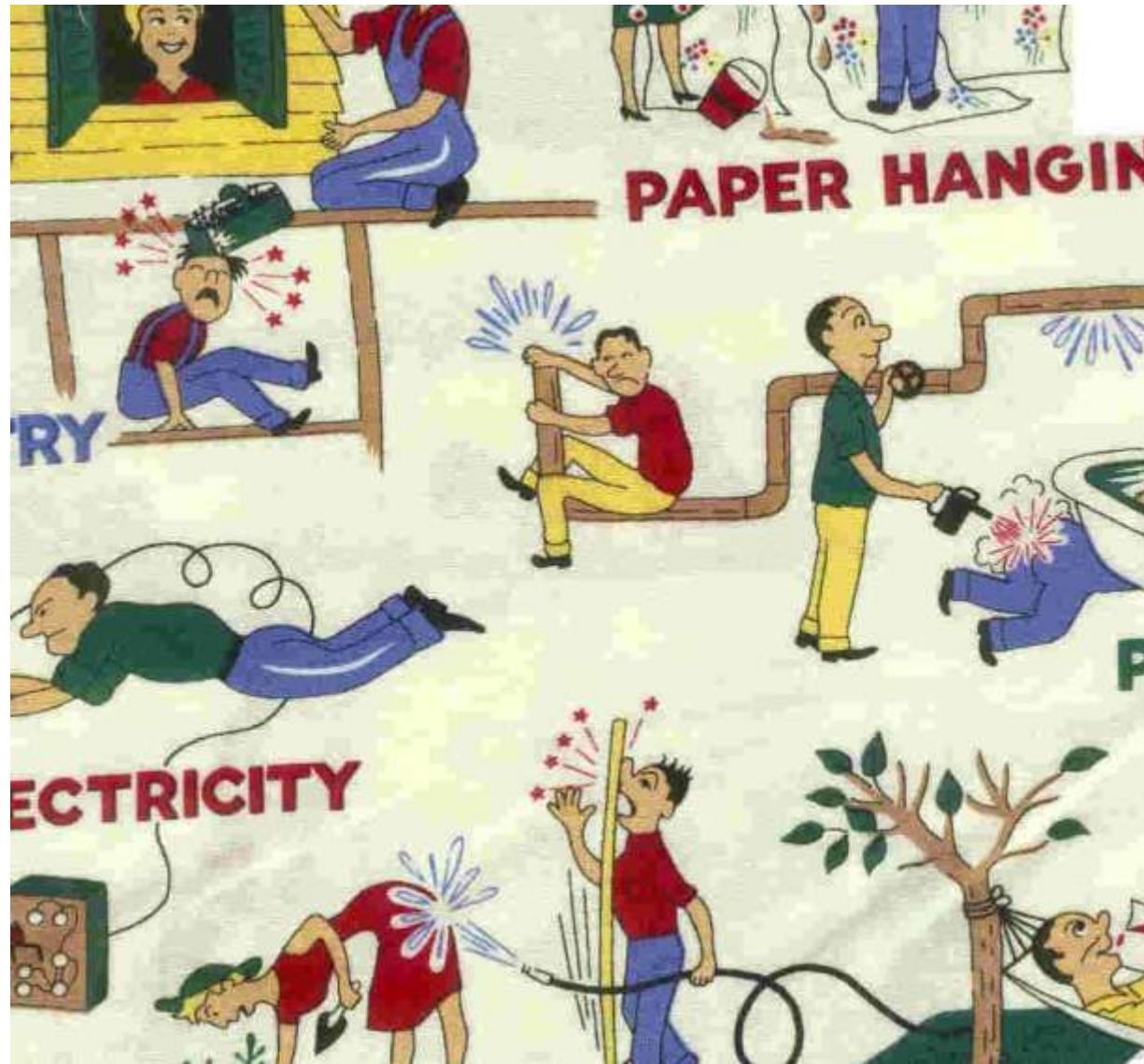
desenvolvimento de software em
indústria, comunidades
acadêmicas e científicas

uma fábrica de software?...

joa@ufrpe.br

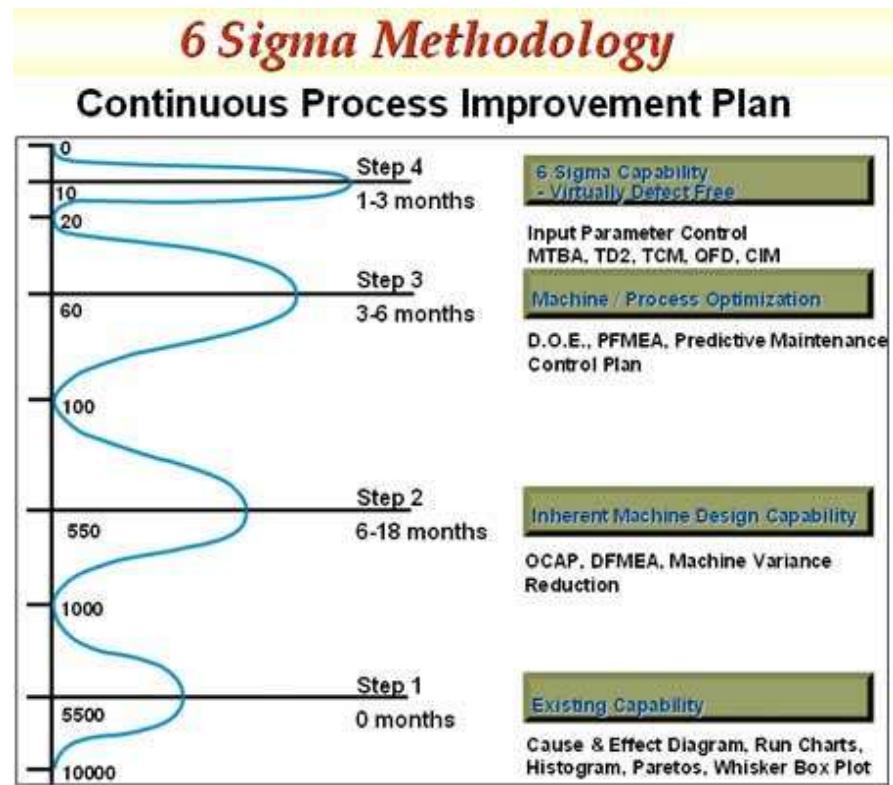
silvio@cesar.org.br

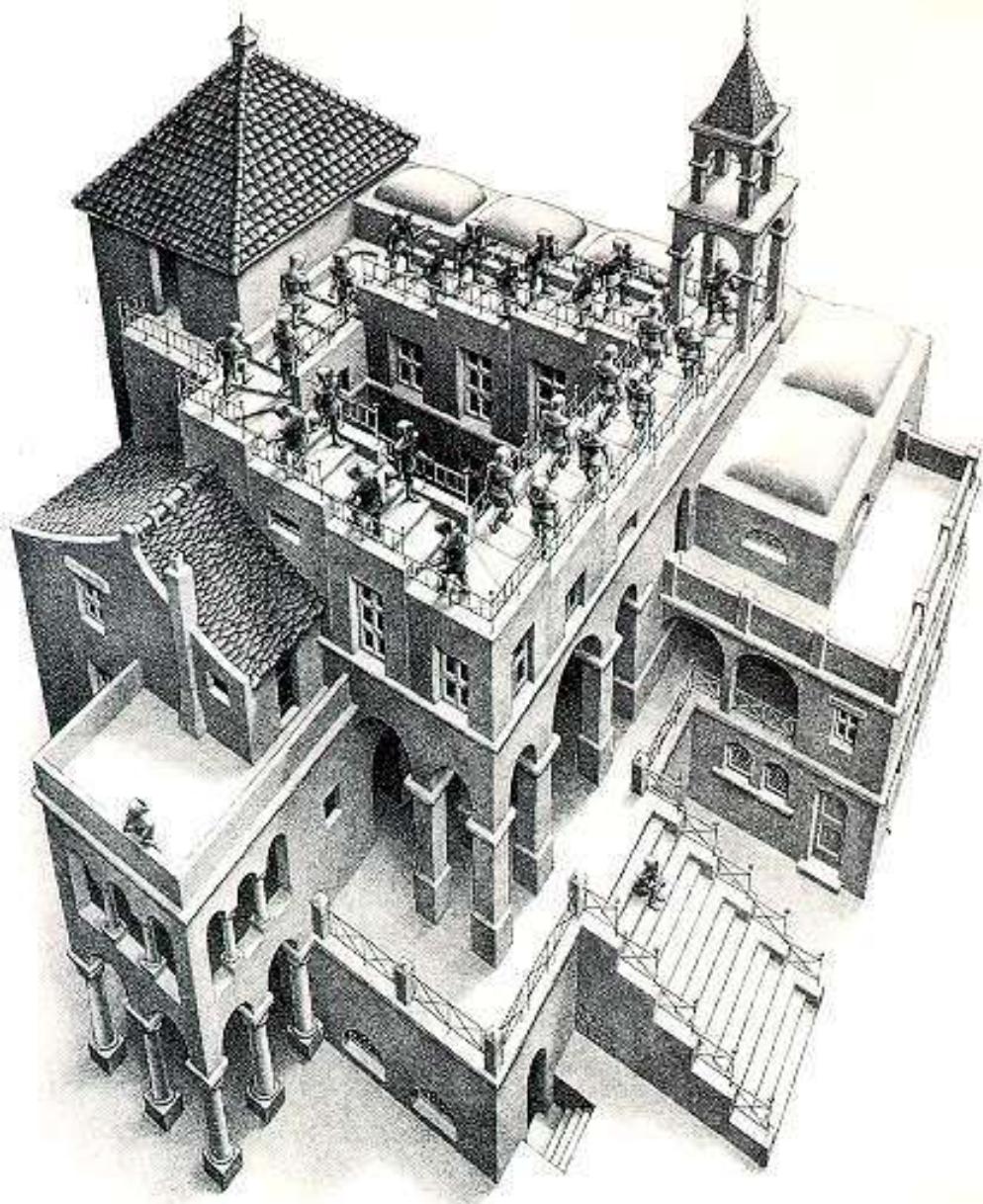
laboratórios de desenvolvimento...



Produção de Software: histórico

- 60's Fábrica de Software
- 70's Kanban
- 70's Just In Time
- 80's SIGMA
- 90's CMM
- 00's Fábrica de Software... de novo?





fábricas de
software

princípios
conceitos
ilusões

fábrica de sw na prática

- Operação profissional
- Processo de desenvolvimento transparente
- Retorno rápido ao cliente
- Alta produtividade
 - Ferramentas e processos padronizados
- Alta qualidade
 - Dados históricos, previsibilidade e análise de risco
- Reusabilidade de código

[1968 G&E, R. W. Berner]

[2003 IEEE Computer Vol.36 Num.3, B. Boehm]

[2005 IEEE Software Vol. 22 Num 2, PostModern SW Design]

Last modified August 10, 2003

<http://www.atai.org/softwarewar.png>

Software Wars

Empire Strikes Back

Free Software (Open Source)



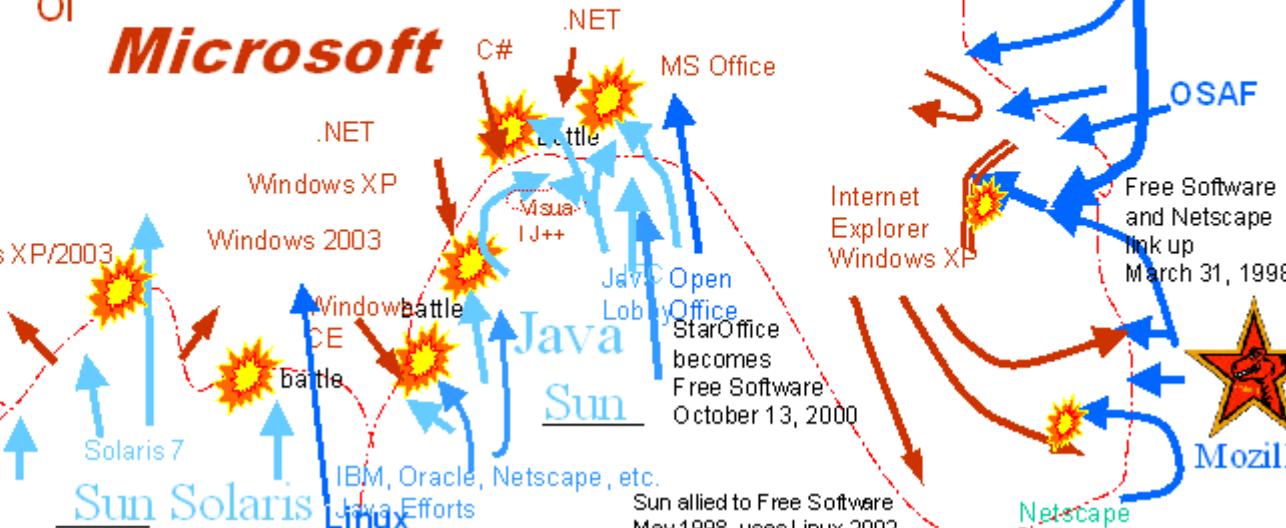
GNU Linux



Perl
Python
Apache



Microsoft



Sun allied to Free Software
May 1998, uses Linux 2002

Netscape





Cathedral and Bazaar

e Desenvolvimento de Software
segundo Eric Raymond, 1997

- Alguns projetos são como *Cathedrals*
 - altamente centralizados em poucas pessoas que decidem projeto e implementação
 - para fazer parte deve-se aceitar as definições

<http://catb.org/~esr/writings/cathedral-bazaar/cathedral-bazaar/>

Outros são como *Bazaars* (sw livre)

- Sem planejamento detalhado mas orientados!
- Linux Kernel (www.linux.org):
“hierarchical”
 - Linus Torvalds, “The Benevolent Dictator”
- APACHE Foundation (www.apache.org):
“meritocracy”
 - Para fazer parte você deve ter colaborado continuamente em projetos da fundação
- GCC (gcc.gnu.org): **“steering Committee”**



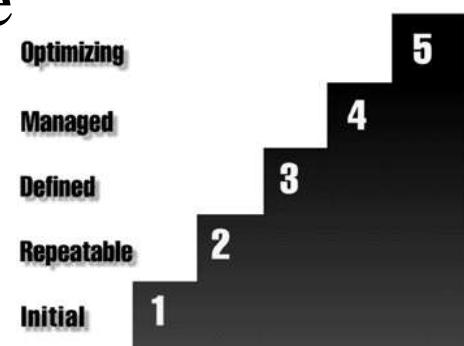
Conceitos

The factory is an organization inhabited by people engaged in a common effort, work is organized one way or the other, standardization is used for coordination and formalization, and systematization is important, but there will be several options for the design of a particular software factory

[Aaen, Botcher, Mathiassen, Software Factories, 1997]
http://www.cin.ufpe.br/~in953/papers/Software_Factories_17.pdf

fábricas de software: 4 estratégias...

- Japonesa (1981,1987)
 - SWB – Software Work Bench
- Européia (1991,1992)
 - ISDE – Integrated Software Development Environment
- Norte-Americana – experiência produção (1989,1993)
- Norte-Americana – níveis de maturidade (1990,1993)



Estratégia Japonesa



- Aumento de produtividade e qualidade de desenvolvimento e manutenção
- Estratégia baseada em infra-estrutura: física, organizacional e ferramental
- Uso de métricas
- Metodologia padronizada para todos os projetos
- Reuso em todas as fases
- Foco em tecnologia

Estratégia Européia



- Ambientes integrados de desenvolvimento orientados a cliente (**IDEs**)
- Estratégia orientada a ferramentas: padronização de componentes, adaptação de processo
- Sem métricas
- Metodologia adaptada por projeto
- Sem reuso
- Foco em Tecnologia
(Organizacional... ISO-9000)

Estratégia Norte Americana 1



- **Baseada em Componentes**
- Maior eficácia de processos, menos re-trabalho e mais reuso
- Estratégia de melhoria contínua baseada na experiência adquirida
- Sem métricas
- Metodologia adaptada por projeto
- Reuso
- Sem foco em tecnologia

(...Ágeis...XP – Extreme Programming)

Estratégia Norte Americana 2



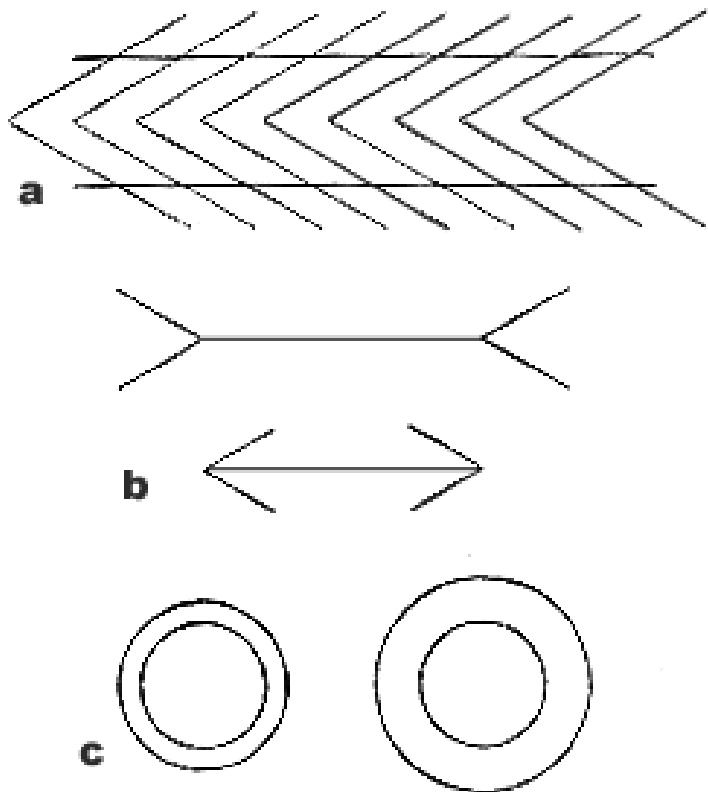
- Processo eficaz, previsível, confiável e auto-melhorável
- Estratégia de melhoria em etapas, níveis de maturidade
- Uso de métricas
- Metodologia adaptada por projeto
- Pouco reuso
- Pouco foco em tecnologia

(Processos... RUP, CMM...)

...considerações...

The four approaches are important contributions towards this goal. At the same time the approaches individually may lead to unfortunate illusions. Learning from the relative strengths and weaknesses between the approaches may help us avoid becoming victims of these illusions

...e ilusões...



**fábrica de software ≠
produção em massa**

**padronização, formalização,
especialização, controle ≠
produção de software com
qualidade**

o que fazer?

building a software factory...

[Making the software factory work... 1990-1999]

<http://www.cin.ufpe.br/~in953/papers/MakingTheSoftwareFactoryWorkLessonsFromADecadeOfFactory.pdf>

1. to define a detailed software development process
2. staff members were given extensive training in the new process
3. process specification separated from process execution
4. data collection and analysis
 1. interviews
 2. software process assessments
 3. process attributes for each project
 4. configuration management system
 5. project tracking data



lessons learned...

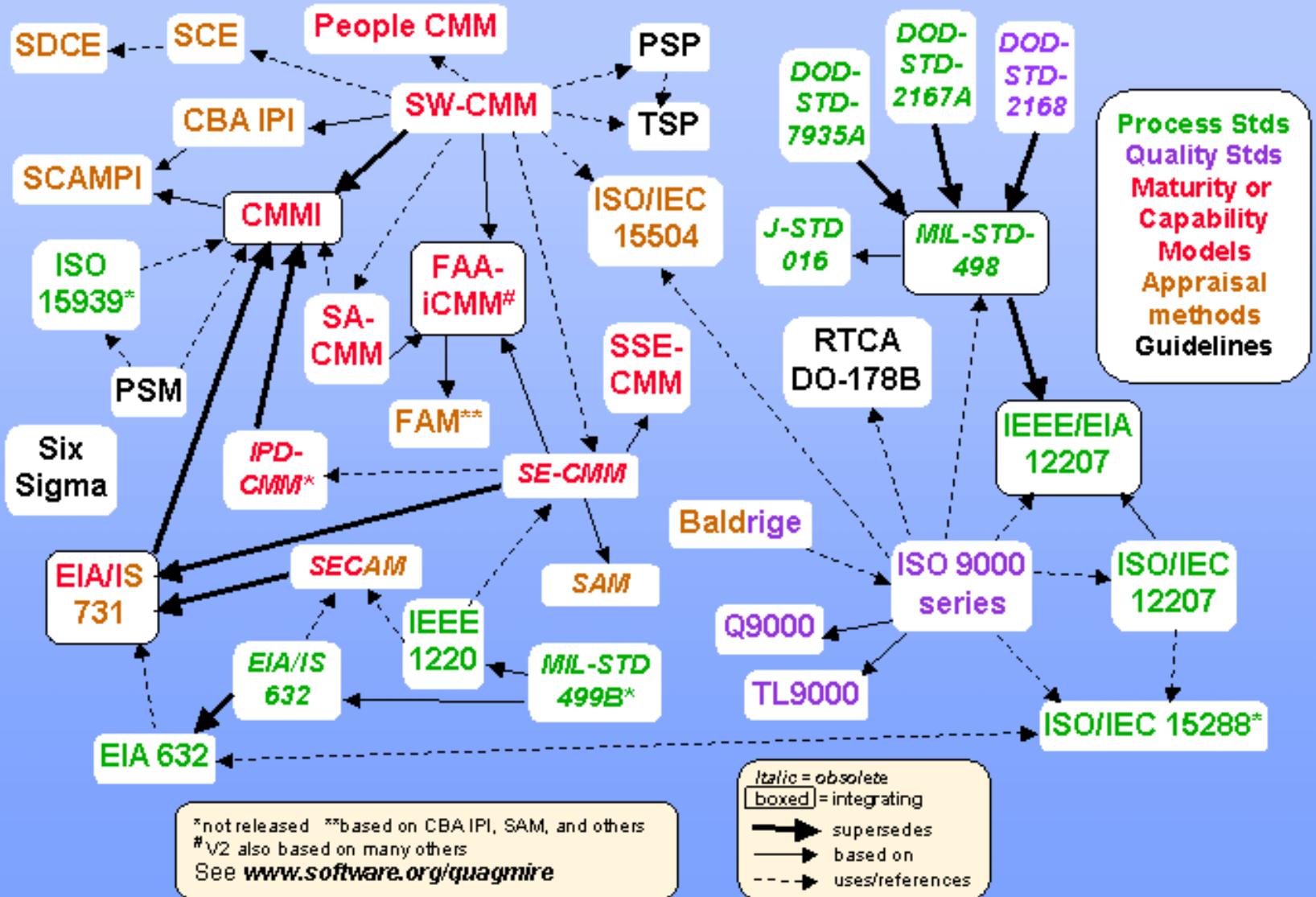
- separating function of process engineer from developer
- job rotation between process engineering and product development

building a software factory...

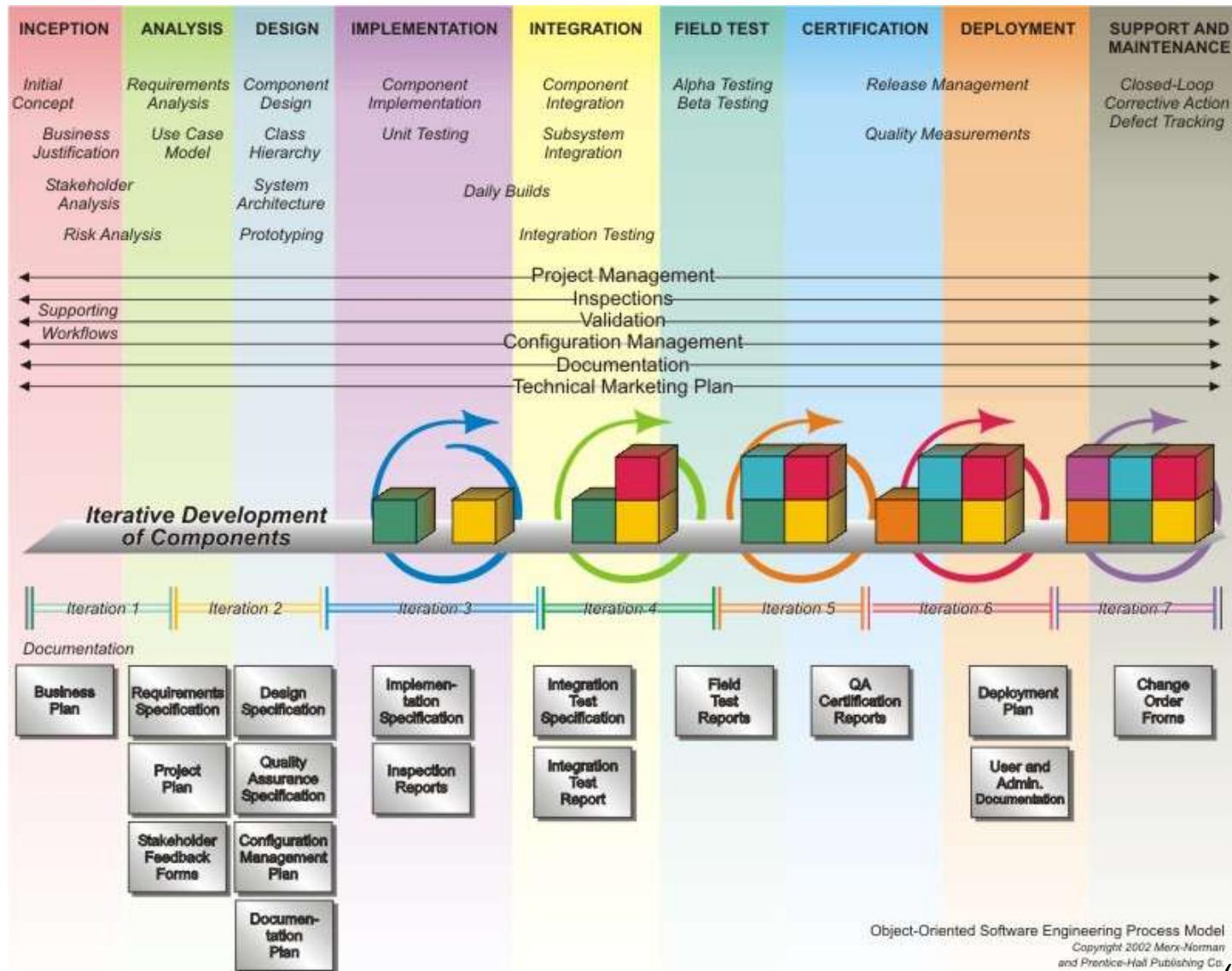
framework:

1. ... process x team
2. ... requirements x architecture
3. ... metrics x uncertainty
4. ... tools x process
5. ... communities
6. ...

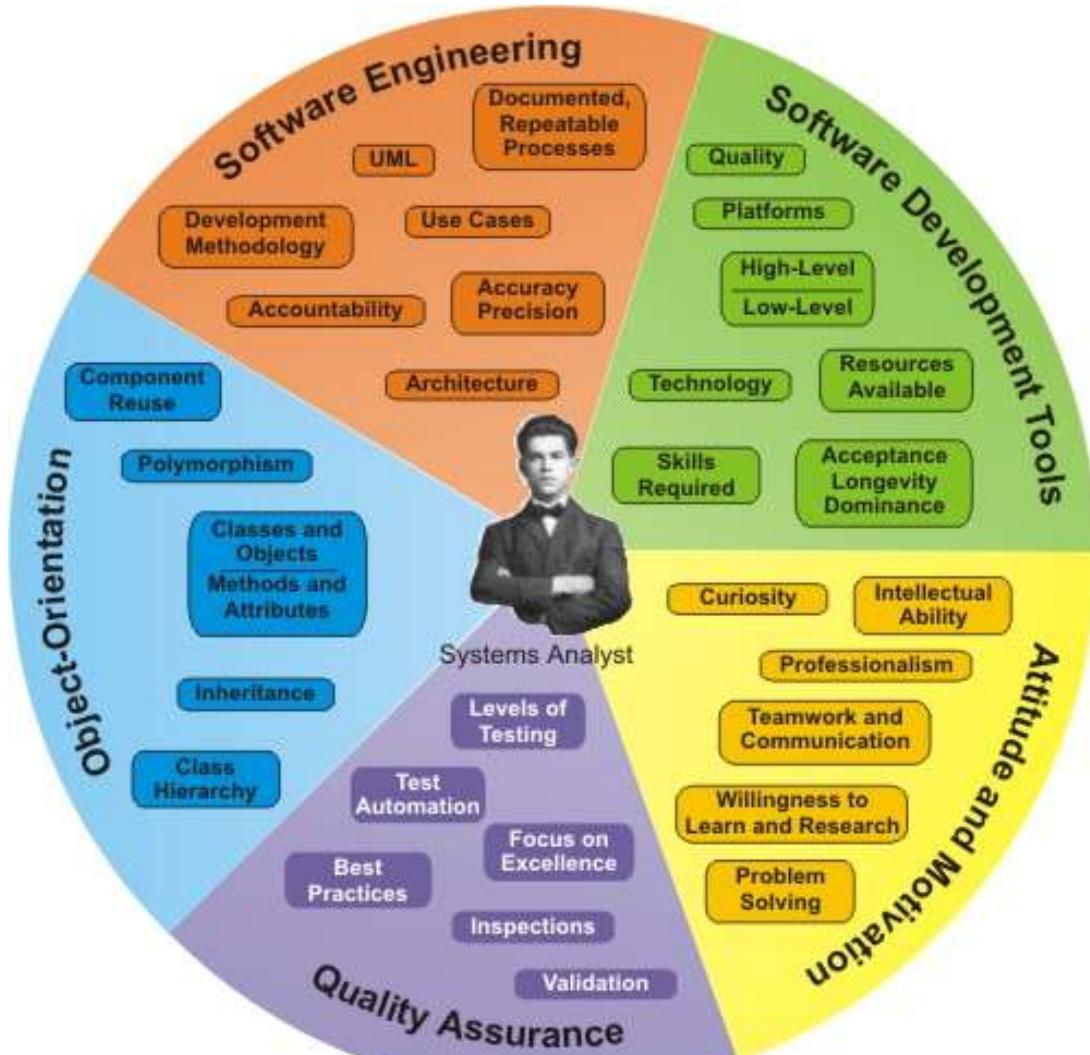




o PROCESSO é complexo!



e as pessoas?...



What is institutionalization?



No, not that kind!

o que “sabemos” aqui?

- TODO o pressman ou sommerville (*Software Engineering*)
- ou seja:
 - requisitos, especificações, refinamento, validação e verificação, métodos, técnicas, linguagens e ferramentas, componentes, reutilização, manutenção, modelagem de processos, qualidade, reengenharia,, verificação, validação e teste... além de programação!...
 - {faltaria o quê?}

o que fazer...

- montar uma fábrica de software?
- como? em 4 meses?

SIMULARE
by USINA - United Software Infrastructure Alliance

Project home

If you were [invited](#) and [logged in](#), you could join this project.

Summary: A framework to allow execution of OO simulations in Java.

Category: Licenses
License: GNU Public License
Owner(s): alexE12, lucard, msp, apaf007, tito

Description

Xsimulare is a Java based framework for simulating environmental systems by modelling simulation items and their relationships.

XSimulare is formed by a group of B.M.Sc. and Ph.D.) Computer Science students from Universidade Federal de Pernambuco, Brazil taking Software Engineering classes with the following objectives:

- Apply Software Engineering concepts, techniques and tools;
- Analyze interactions and disconnections between Software Engineering and Open Source Development;
- Define, create and evaluate an open source software factory (within a real world open source software development) generating a detailed report of the experiment.

CIn - UFPE

Pós-Graduação em Ciência da Computação
(Mestrado e Doutorado)

IN953 - Engenharia de Software: fabricas de software livre

(1º Semestre/2004)
(1º Semestre/2003)

Última atualização: Mie 3 Jun 7 14:39:55 BRT 2004

Objetivos da Disciplina

project highlights:

USINA - United Software Infrastructure Alliance

[Project Home](#) [Project Members](#) [Announcements](#) [Planning](#) [Success](#) [Projects](#)

Projects

We are not accepting proposals while validating the first stable Usina's artifacts release. By now we are executing pilot projects to validate the Usina's process and artifacts, below we list the testing projects:

Project name	Brief description	Status
Simulare pilot	This is the first project built using the Usina's process. It is a testing project to verify process overheads and accuracy.	Finished
Project's Artifacts (templates, code and directory structures)	You can use it as a guide to start your new project.	Ongoing
Simulare final	The final project instance using the development process specified by Usina.	Ongoing

Rules and Tools for software evolution planning and Management 2001...

- Continuing change
- Increasing complexity
- Continuing growth
- 10 regras... 1 princípio.. de incerteza!...



the real world outcome of any software execution is inherently uncertain with the precise area of uncertainty also not knowable