



## **Auditoria e Qualidade de Sistemas**

Prof. Edgard Davidson C. Cardoso



Qualidade de Software

# INTRODUÇÃO

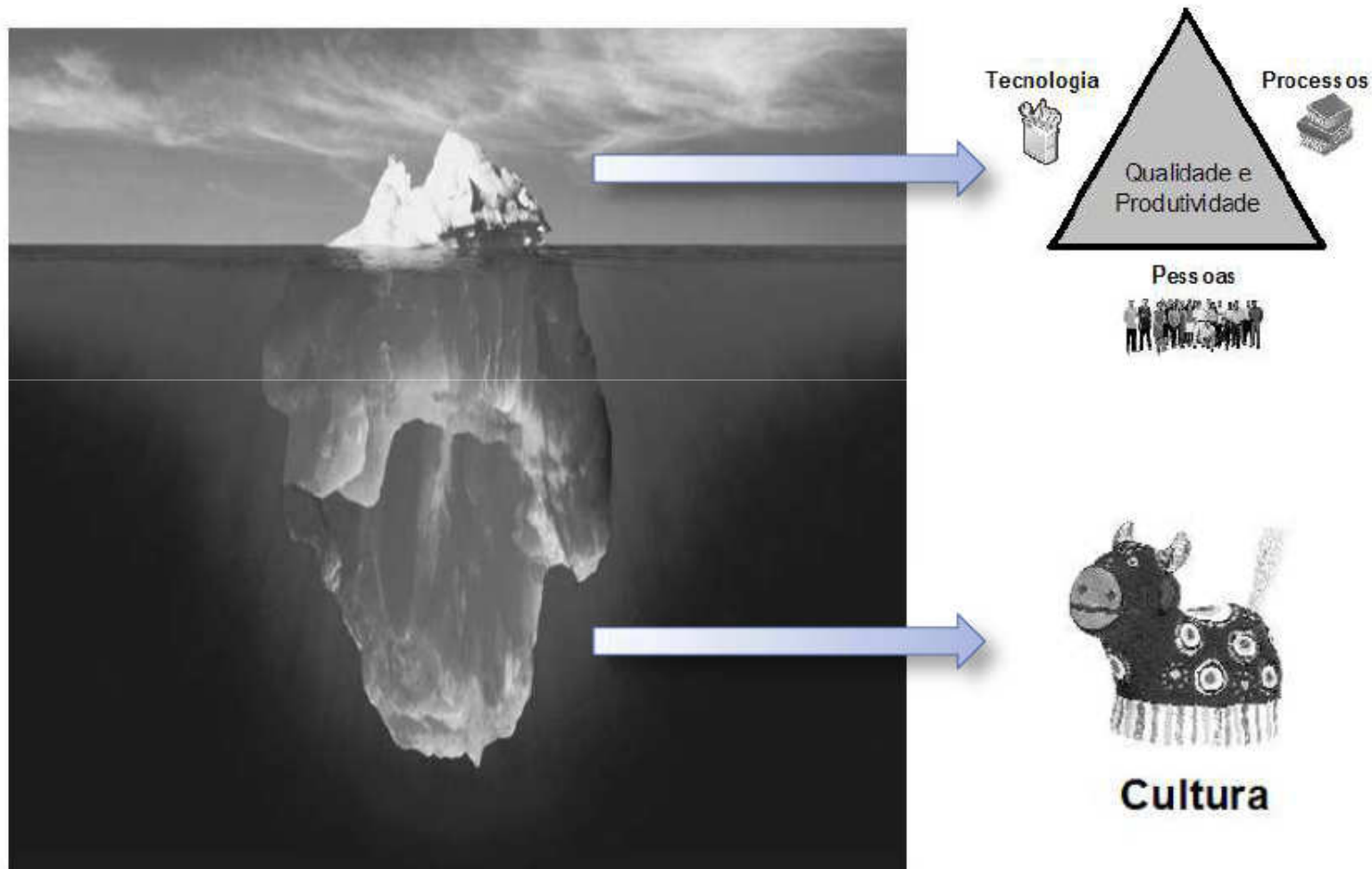


# O Ambiente de Projeto de Software





# O Ambiente de Projeto de Software





# Característica dos componetes

- Conhecimento e habilidades
- Motivação e comprometimento
- Reconhecimento
- Crescimento

Pessoas



- Personalidade coletiva
- Risco X Segurança
- Etiqueta
- O “jeito de ser” da organização

Cultura



- Disciplina e coordenação
- Gerenciamento
- Padronização
- Institucionalização

Processos



- Produtividade
- Controle
- Eficiência
- Automação

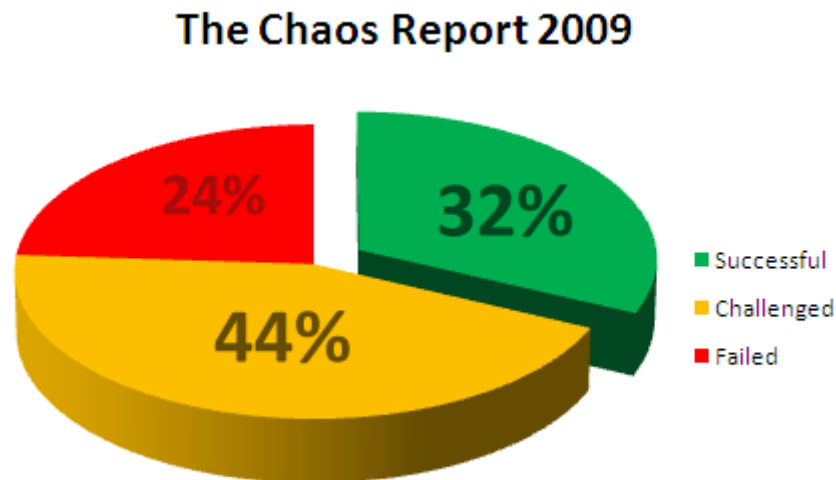
Ferramentas





# A Verdade Sobre os Projetos

- O Standish Group vem, há mais de uma década, realizando estudos em volta dos resultados dos projetos de software ao redor do mundo. O resultado destes estudos é um relatório batizado de Chaos Report;





# Se Fabricássemos Aviões



**32 %**  
Pousariam sem  
Problemas



**44 %**  
Pousariam com  
Problemas



**24 %**  
Cairiam



# Chaos Report

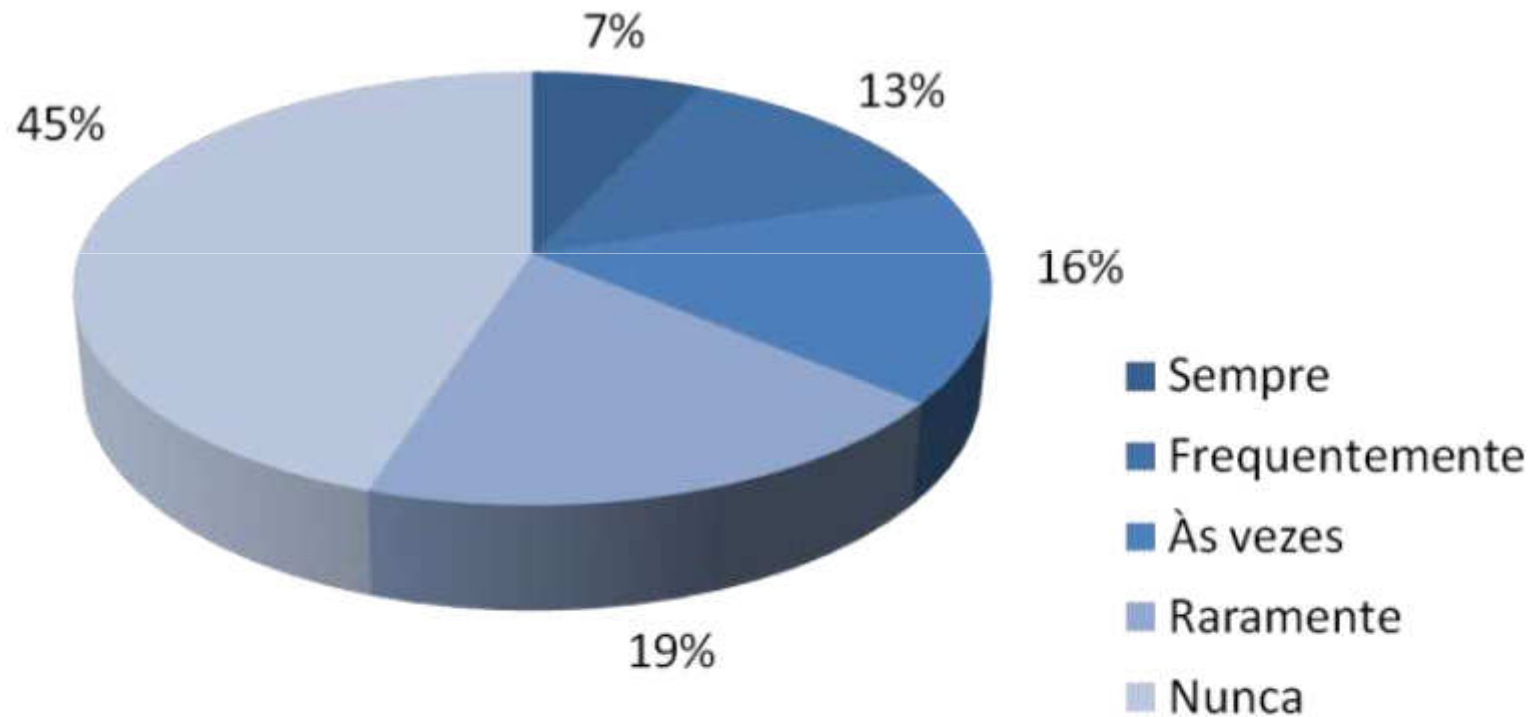
- Segundo o Standish Group quais os principais fatores para um número ainda tão alto de projetos que não alcançam seu objetivo?
- A vasta maioria dos projetos de software falha por falta de clareza – sobre funções pessoais, responsabilidades e requisitos – e também por inabilidade para acompanhar o que ocorre em cada um dos diferentes passos do ciclo de vida da aplicação.





# Uso de funcionalidades

## Média de uso de funcionalidades de sistemas



Standish Group, 2002



# O que é Qualidade?

- O que um determinado produto apresenta para considerarmos que o mesmo tem qualidade? Ex.: Carro
- **Qualidade é um conceito relativo.** Diversos aspectos são levados em conta. No caso de um automóvel, fatores como conforto, segurança, desempenho, beleza e custo têm estreita relação com a qualidade.



# O que é Qualidade?

- Qualidade está fortemente relacionada à conformidade com os requisitos.
- O que é “conformidade em relação a requisitos”? observado x especificado.
- Pode haver problemas na observação.
- Pode haver problemas na especificação.



# O que é Qualidade?

- Qualidade diz respeito à satisfação do cliente.
- Requisitos são especificados por pessoas e com o objetivo de satisfazer outras pessoas.
  - Uma especificação depende das escolhas feitas (clientes alvo).
  - Pode haver problemas na especificação.



# O que é Qualidade de Software?

- Conferência da NATO (1968) – Crise de Software





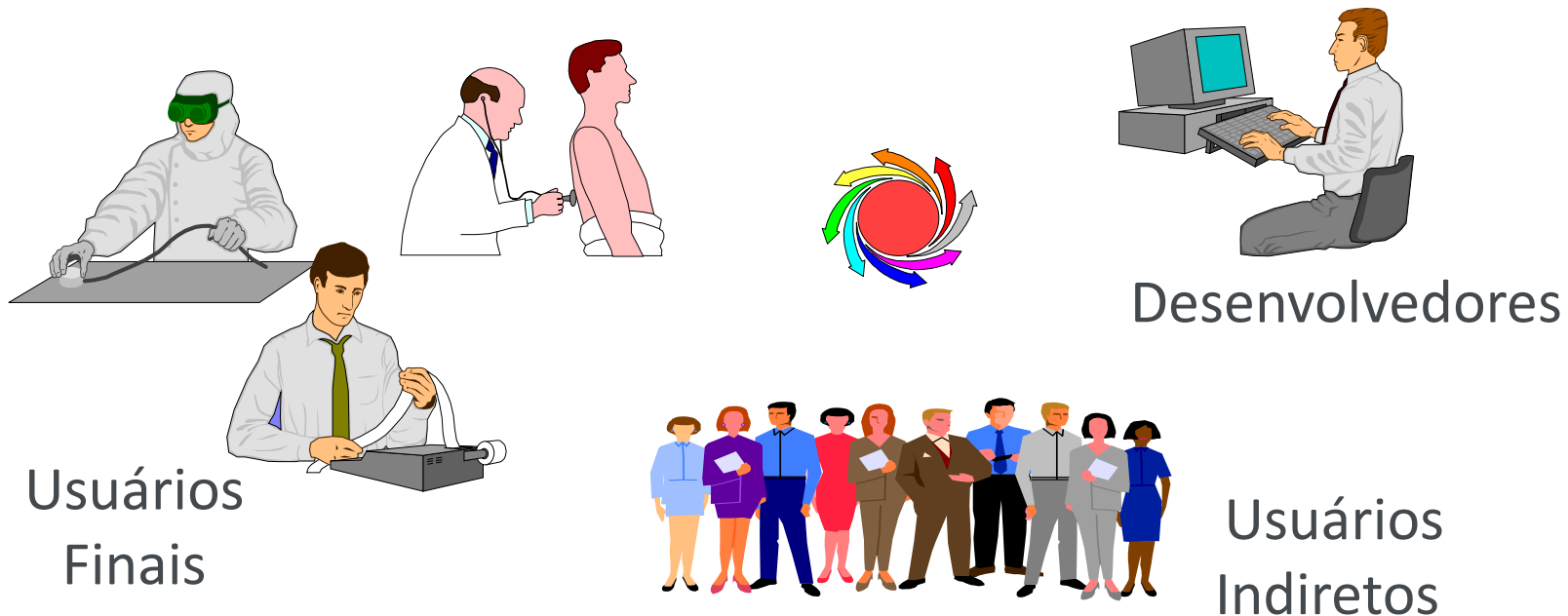
# O que é Qualidade de Software?

- Problemas detectados:
  - Cronogramas não observados.
  - Projetos abandonados.
  - Módulos que não operam corretamente quando combinados.
  - Programas que não fazem exatamente o que era esperado.
  - Sistemas tão difíceis de usar que são descartados.
  - Sistemas que simplesmente param de funcionar.
- Passados quase 40 anos, o que mudou?



# O que é Qualidade de Software?

Conjunto de características a serem satisfeitas em um determinado grau, de modo que o software satisfaça às necessidades de seus usuários.





# Qualidade do Produto x Qualidade do Processo de Software

- Qualidade do produto de software não se atinge de forma espontânea.
- A qualidade do produto depende fortemente da qualidade do processo de desenvolvimento.





# Qualidade do Processo de Software

- Um bom processo não garante que os produtos produzidos são de boa qualidade, mas é um indicativo de que a organização é capaz de produzir bons produtos.



# Qualidade do Processo de Software

- Motivação para a busca da Qualidade do Processo de Software:
  - Aumento da qualidade do produto.
  - Diminuição do retrabalho.
  - Maior produtividade.
  - Redução do tempo para atender o mercado (*time to market*).
  - Maior competitividade.
  - Maior precisão nas estimativas.



# Qualidade do Processo de Software

- A implantação de um Programa de Qualidade começa pela definição e implantação de um processo de software.
- **Questão:** Por onde começar? O que considerar na definição de processos de software?



# O que é Processo de Software?

- Processos de Software devem estabelecer:
  - atividades a serem realizadas durante o processo, sua estrutura e organização (decomposição e precedência), incluindo a definição de um modelo de ciclo de vida quando pertinente (ex.: processo de desenvolvimento);
  - artefatos requeridos e produzidos por cada uma das atividades do processo;
  - procedimentos (métodos, técnicas, roteiros e padrões) a serem adotados na realização das atividades;
  - recursos necessários (humanos, hardware e software) para a realização das atividades.



# Abordagem Prescritiva ou Tradicional

- Processo de Desenvolvimento de Software Tradicional
  - Baseado em:
    - ISO 12207
    - SPICE
    - CMMI
    - MPS-BR
    - PMI
    - UP
    - RUP
    - Praxis



# Abordagem Tradicional

## Estrutura do Processo de Software

- Processo de Software
  - Sub-processos (Desenvolvimento, Manutenção, etc)
    - Atividades
      - Sub-atividades
      - Pré-atividades
      - Artefatos
        - » Insumos
        - » Produtos
      - Recursos
        - » Humanos
        - » Software
        - » Hardware
      - Procedimentos
        - » Métodos
        - » Técnicas
        - » Roteiros



# Abordagem Tradicional

## Exemplo de Processo de Software

- Processo de Software
  - Processo de Desenvolvimento
    - Análise e Especificação de Requisitos
  - Projeto
    - Projeto Arquitetural
      - » Pré-atividade: Análise e Especificação de Requisitos
      - » Insumo: Documento de Especificação de Requisitos
      - » Produto: Documento de Arquitetura do Sistema
      - » Recurso Humano: Projetista de Software
      - » Ferramenta de Software: Ferramenta de Modelagem UML
      - » Técnica: Modelagem de Objetos
    - Projeto Detalhado
  - Implementação
  - Testes
  - Implantação



# Abordagem Tradicional

## Como definir um processo?

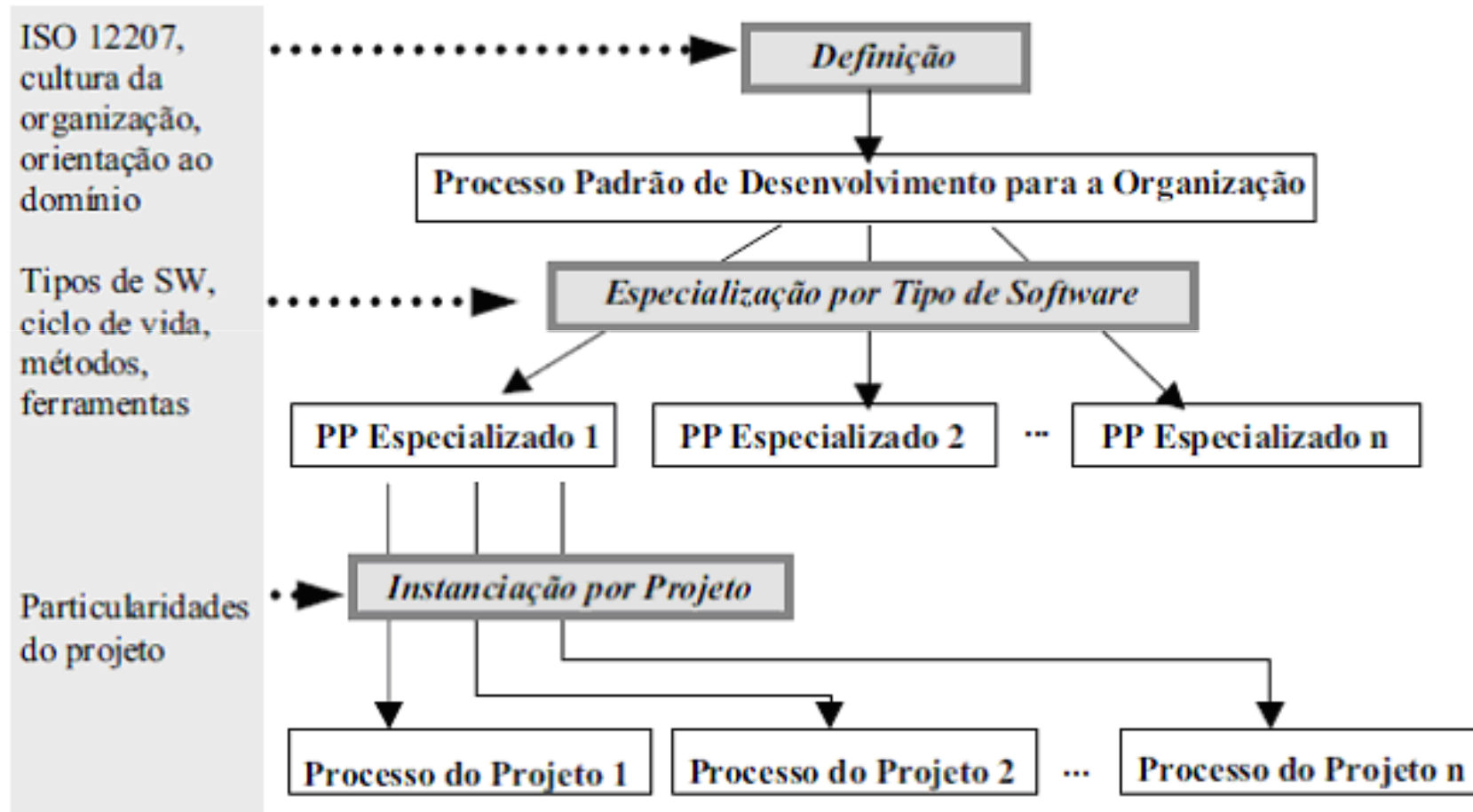
- Processos têm de ser definidos caso a caso, levando-se em consideração as características específicas do projeto em questão: equipe, domínio de aplicação, tipo de software, tecnologias a serem adotadas, restrições de negócio (cronograma, custo, qualidade) etc.
- Apoio de Normas e Modelos de Qualidade de Processos de Software.

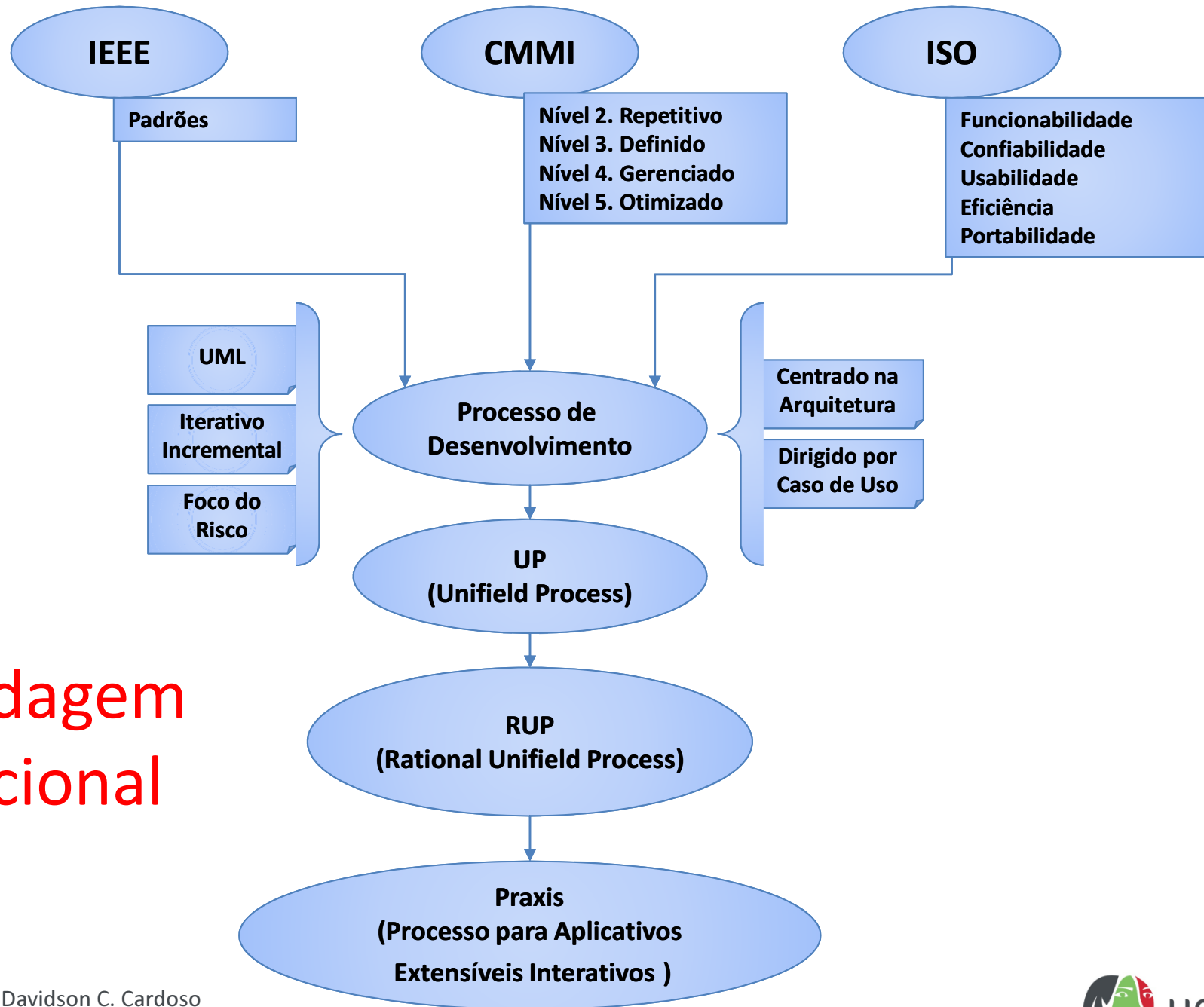




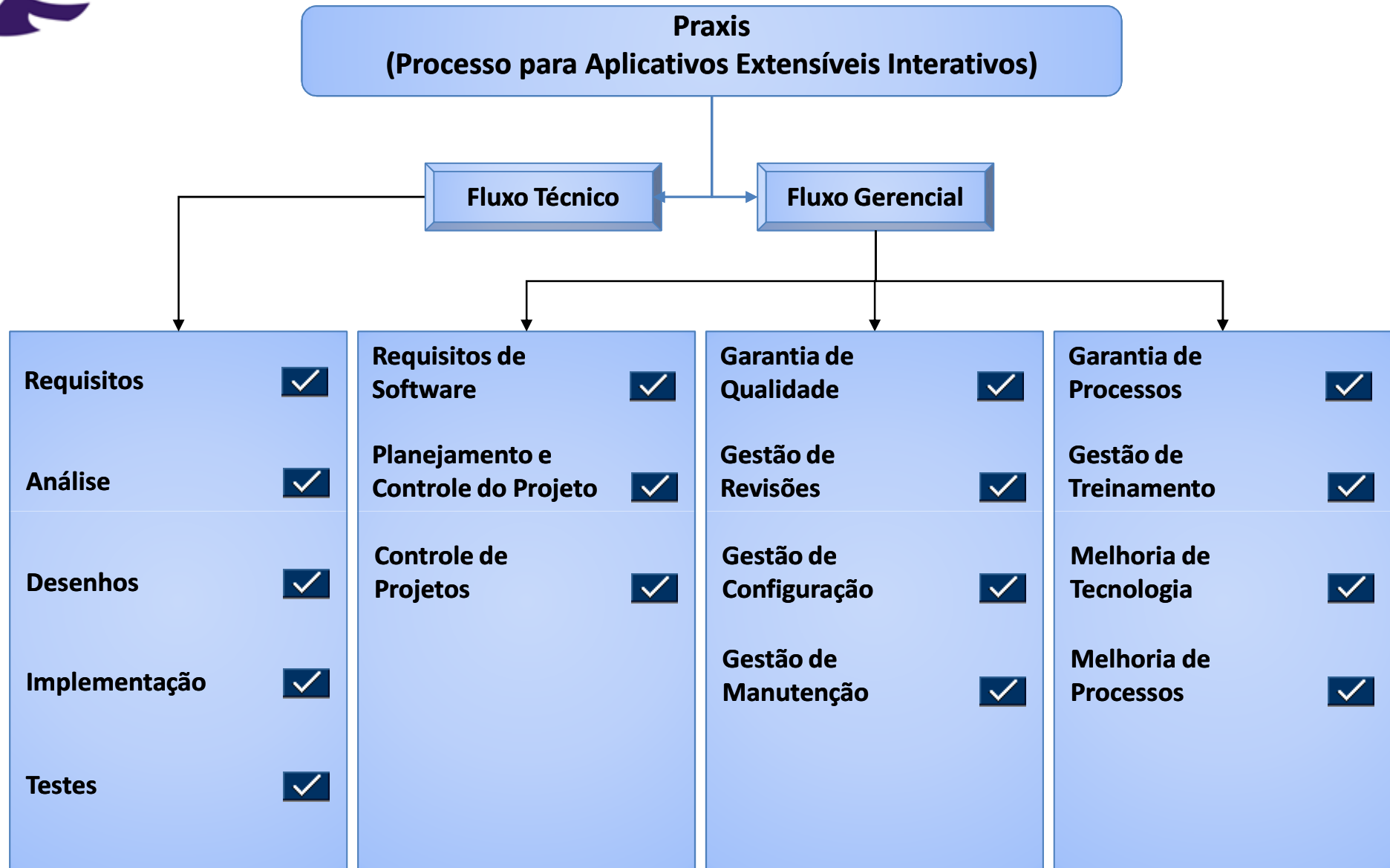
# Abordagem Tradicional

## Como definir um processo?





**Abordagem  
Tradicional**



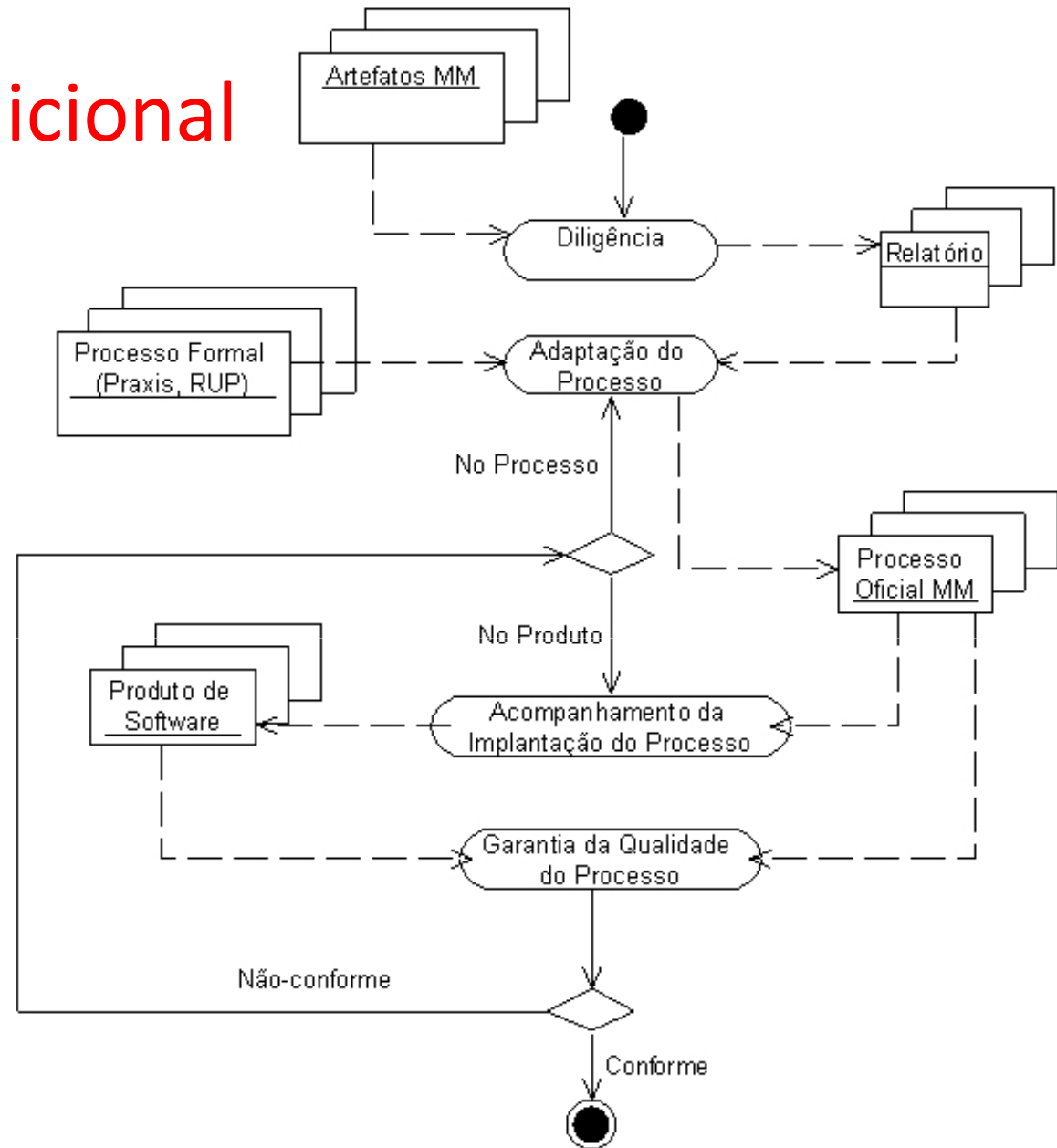
## Abordagem Tradicional

Prof. Edgard Davidson C. Cardoso



# Abordagem Tradicional

Definindo um Processo de Desenvolvimento de Software customizado para sua organização





# Abordagem Tradicional

## Qualidade de Software

### Projeto

- PMI – PMBOK
- ISO 10006

### Processo

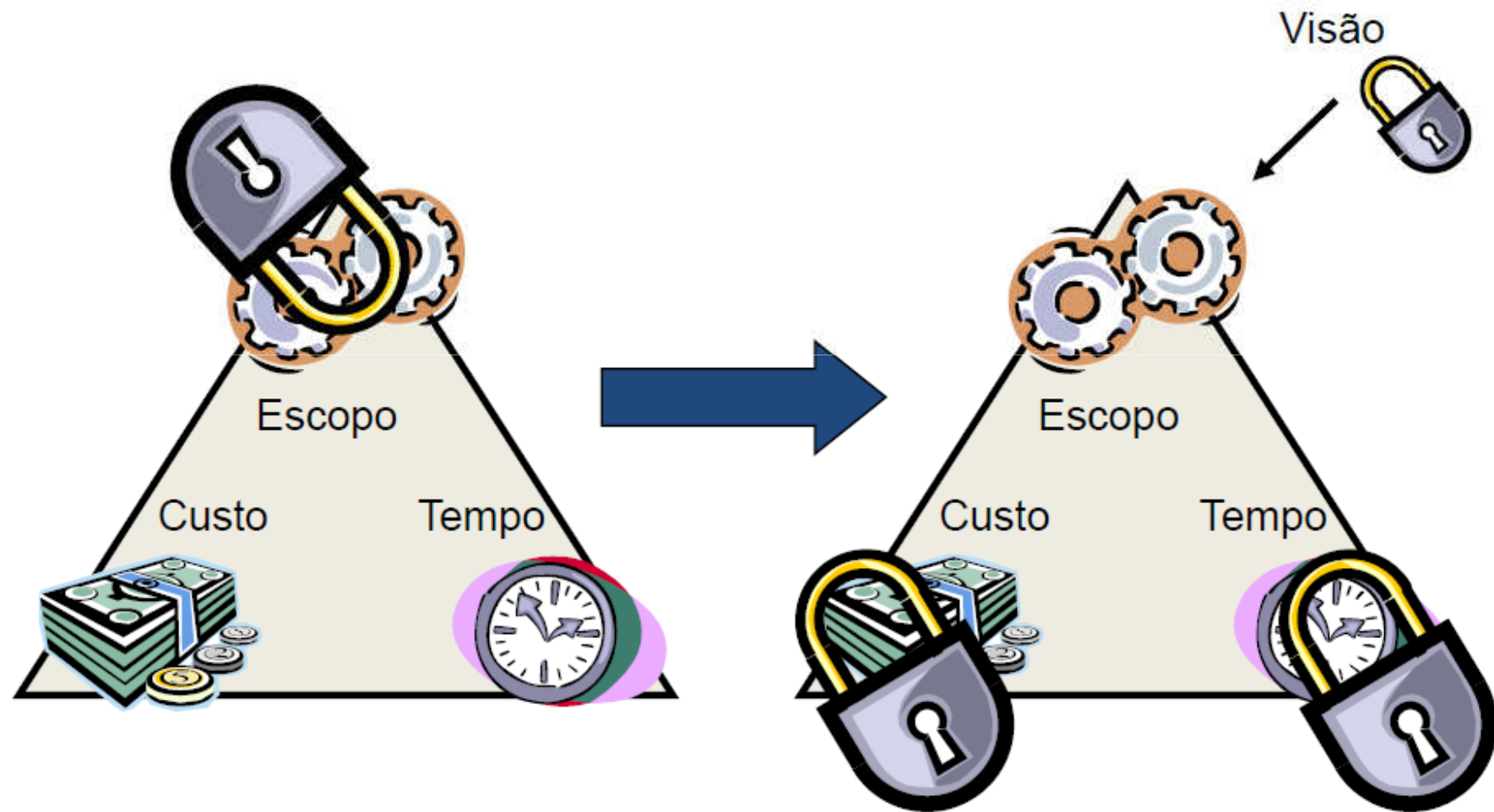
- ISO 12207, 15504
- CMMI, MPS-BR

### Produto

- ISO 9126, NBR 13596
- ISO 12119



# Processo Tradicional x Ágil





# Tradicional, Plan Driven, Definido

- Os processos baseados no modelo *waterfall* (cascata) buscam organizar o desenvolvimento de produtos no formato de linha de produção;
- Isto se parece com projetos da área de TI?





# Abordagem Ágil

- Métodos Ágeis de Desenvolvimento de Software
  - Baseado em:
    - Scrum
    - XP
    - TDD
    - FDD
    - DDD
    - Lean





# Abordagem Ágil

## O Manifesto Ágil

“Estamos descobrindo maneiras melhores de desenvolver software fazendo-o nós mesmos e ajudando outros a fazê-lo. Através desse trabalho, passamos a valorizar:

**Indivíduos e interação entre eles** mais **que** processos e ferramentas  
**Produto em funcionamento** mais **que** documentação abrangente  
**Colaboração com o cliente** mais **que** negociação de contratos  
**Responder a mudanças** mais **que** seguir um plano

Ou seja, mesmo havendo valor nos itens à direita, valorizamos mais os itens à esquerda.”

*<http://agilemanifesto.org>*



# Abordagem Ágil

## Desenvolvimento Iterativo





# Abordagens Ágeis mais Difundidas

## Scrum

- É uma abordagem ágil para o gerenciamento de projetos. Fornece práticas que ajudam gerentes a tornar mais dinâmico e gerenciável o ambiente de desenvolvimento de software.

## XP (eXtreme Programming)

- É uma abordagem ágil para a engenharia de projetos de software. Como o próprio nome diz, é extremamente focada no desenvolvimento, e tem como principal característica a programação em par.

## FDD (Feature Driven Development)

- É uma abordagem ágil para a engenharia de projetos de software. Defende o desenvolvimento de um modelo abrangente no início do projeto pelo qual as funcionalidades do sistema serão descobertas e desenvolvidas.