

## Processos de Software

---

- Modelos de processos
- Iteração de processo
- Especificação de software
- Implementação e desenho de software
- Validação de software
- Evolução de software
- Suporte automatizado ao processo

João Araújo – Engenharia de Software

## O Processo de Software

---

- Um conjunto estruturado de actividades para desenvolver um sistema de software
  - Especificação
  - Desenho
  - Validação
  - Evolução
- Um modelo de processo é uma representação abstrata de um processo

João Araújo – Engenharia de Software

# Modelos de processo

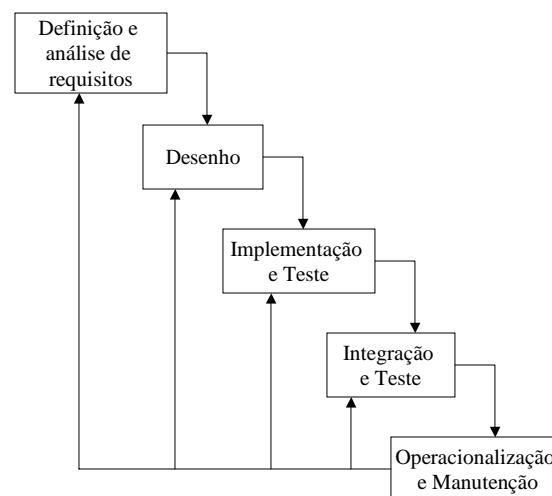
---

- O modelo Cascata
  - Fases distintas e separadas de especificação e desenvolvimento
- Desenvolvimento Evolutivo
  - Especificação e desenvolvimento são intercalados
- Desenvolvimento formal de sistemas
  - Um modelo matemático do sistema é formalmente transformado em uma implementação
- Desenvolvimento baseado em reutilização
  - O sistema é construído a partir de componentes já existentes

João Araújo – Engenharia de Software

# Modelo Cascata

---



João Araújo – Engenharia de Software

## Problemas com o modelo Cascata

---

- Dificuldade em acomodar mudanças depois que o processo está a ser executado
- Partição inflexível do projecto em estágios distintos
- Dificuldade em responder a mudanças dos requisitos
- É mais apropriado quando os requisitos são bem compreendidos

João Araújo – Engenharia de Software

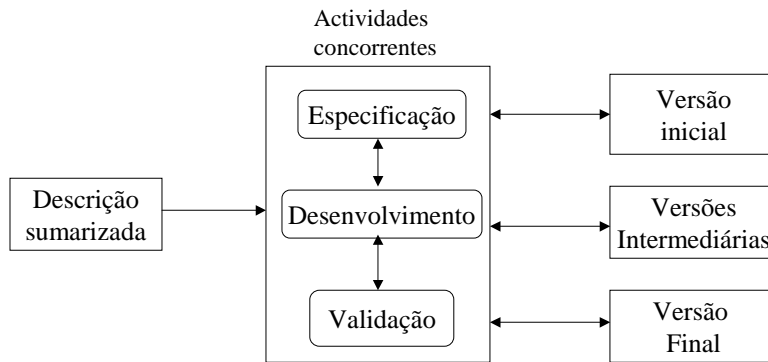
## Desenvolvimento Evolutivo

---

- **Desenvolvimento exploratório**
  - O objetivo é interagir com os clientes e obter um sistema a partir de uma especificação inicial. Deve-se começar com requisitos bem compreendidos
- **Prototipação descartável**
  - O objectivo é compreender os requisitos. Deve-se começar com requisitos mal compreendidos

João Araújo – Engenharia de Software

# Desenvolvimento Evolutivo



João Araújo – Engenharia de Software

# Desenvolvimento Evolutivo

- **Problemas**
  - Falta de visibilidade do processo
  - Os sistemas são mal estruturados
  - Conhecimentos especializados (e.g. em linguagens para prototipação rápida) pode ser preciso
- **Aplicabilidade**
  - Sistemas interactivos de pequeno e médio porte
  - Partes de sistemas de larga escala (e.g. interface do utilizador)
  - Sistemas com tempo de vida curto

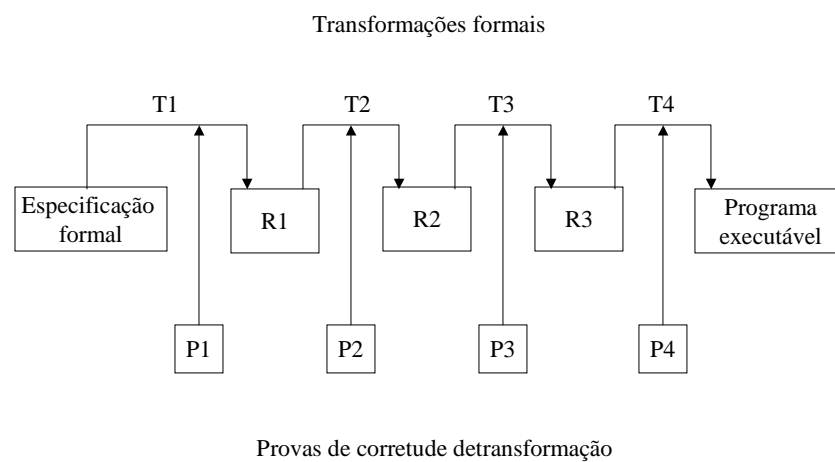
João Araújo – Engenharia de Software

## Desenvolvimento formal de sistemas

- Baseado em transformações sucessivas de uma especificação matemática em um programa executável
- As transformações preservam a corretude, portanto é possível mostrar que o programa está de acordo com sua especificação
- Está incorporado na abordagem 'Cleanroom' de desenvolvimento de software

João Araújo – Engenharia de Software

## Transformações formais



João Araújo – Engenharia de Software

## Desenvolvimento formal de sistemas

---

- Problemas
  - Necessidade de conhecimentos específicos e treino para aplicar a técnica
  - Dificuldade para formalmente especificar alguns aspectos do sistema tais como a interface do utilizador
- Aplicabilidade
  - Sistemas críticos onde aspectos de segurança são cruciais

João Araújo – Engenharia de Software

## Desenvolvimento orientado à reutilização

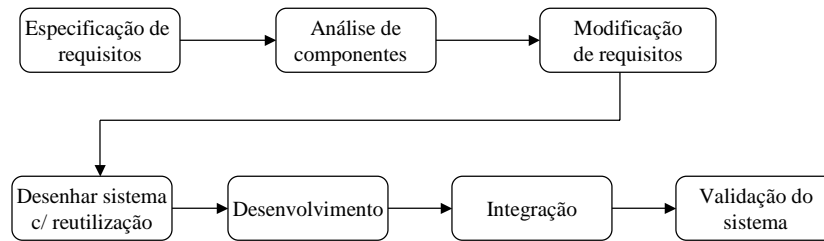
---

- Baseado na reutilização sistemática onde sistemas são construídos a partir de componentes existentes ou sistemas COTS (Commercial-off-the-shelf)
- Estágios do processo
  - Análise de componentes
  - Modificação de requisitos
  - Desenho do sistema com reutilização
  - desenvolvimento e integração
- Esta abordagem tem grande potencial, mas ainda falta mais experiência com seu uso

João Araújo – Engenharia de Software

## Desenvolvimento orientado à reutilização

---



João Araújo – Engenharia de Software

## Iteração no processo

---

- Os requisitos de um sistema sempre evoluem no decorrer de um projecto. Portanto iteração é parte fundamental no processo de desenvolvimento de sistemas de larga escala
- Abordagens
  - Desenvolvimento incremental
  - Desenvolvimento espiral

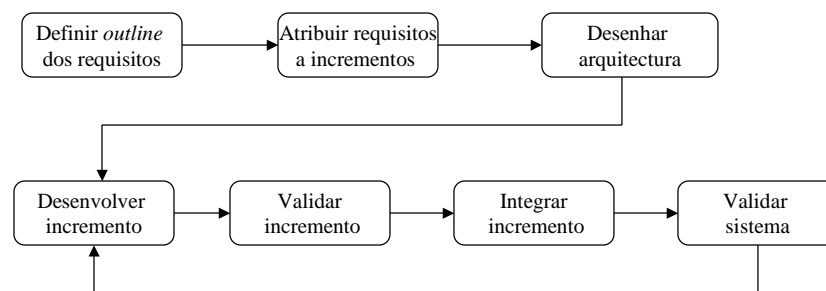
João Araújo – Engenharia de Software

# Desenvolvimento Incremental

- Em vez de entregar os sistema de uma só vez, o desenvolvimento e a entrega é realizada a partir de incrementos onde cada incremento corresponde a uma das funcionalidades do sistema
- É definida uma prioridade entre os requisitos, e os requisitos de prioridade mais alta são incluídos nos incrementos iniciais
- Uma vez que o desenvolvimento de um incremento é iniciado, seus requisitos são congelados

João Araújo – Engenharia de Software

# Desenvolvimento Incremental



João Araújo – Engenharia de Software

## Desenvolvimento Incremental - vantagens

---

- O cliente não tem que esperar até que todo o sistema seja entregue
- Incrementos iniciais agem como um protótipo que ajuda a elicitar os requisitos de incrementos posteriores
- Menor risco de fracasso geral do projecto
- Os serviços de prioridade mais alta são testados exhaustivamente

João Araújo – Engenharia de Software

## Extreme programming

---

- É uma nova abordagem baseada no desenvolvimento e entrega de incrementos de funcionalidade
- Baseia-se num constante melhoramento de código, envolvimento do utilizador, “egoless programming” e testes intensivos

João Araújo – Engenharia de Software

