Pontifícia Universidade Católica
do Rio de Janeiro

ENG1000 – Introdução à Engenharia

Aula 01 – Processo de Desenvolvimento de Software

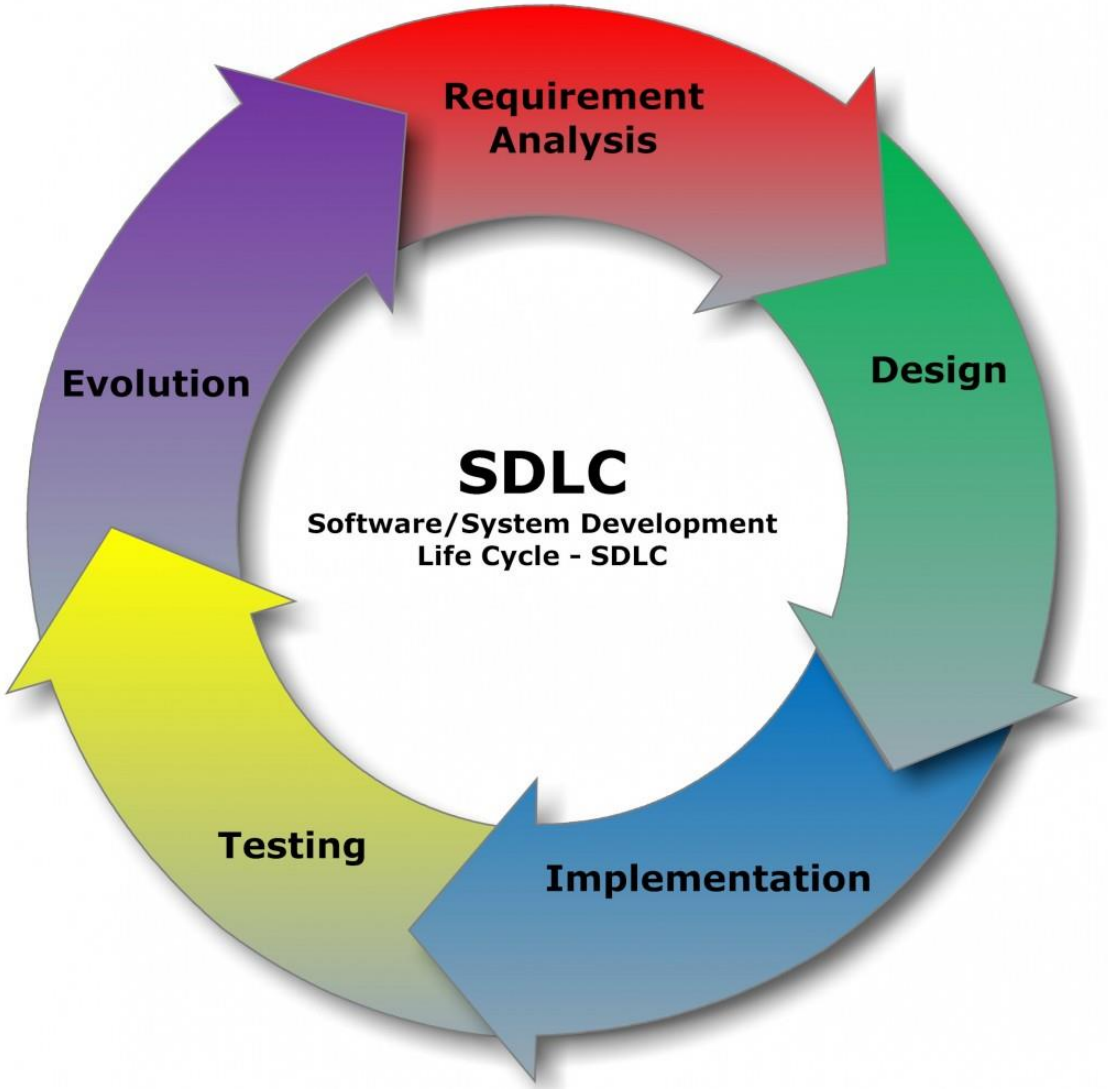
Edirlei Soares de Lima
<elima@inf.puc-rio.br>

visionlab
visualization laboratory

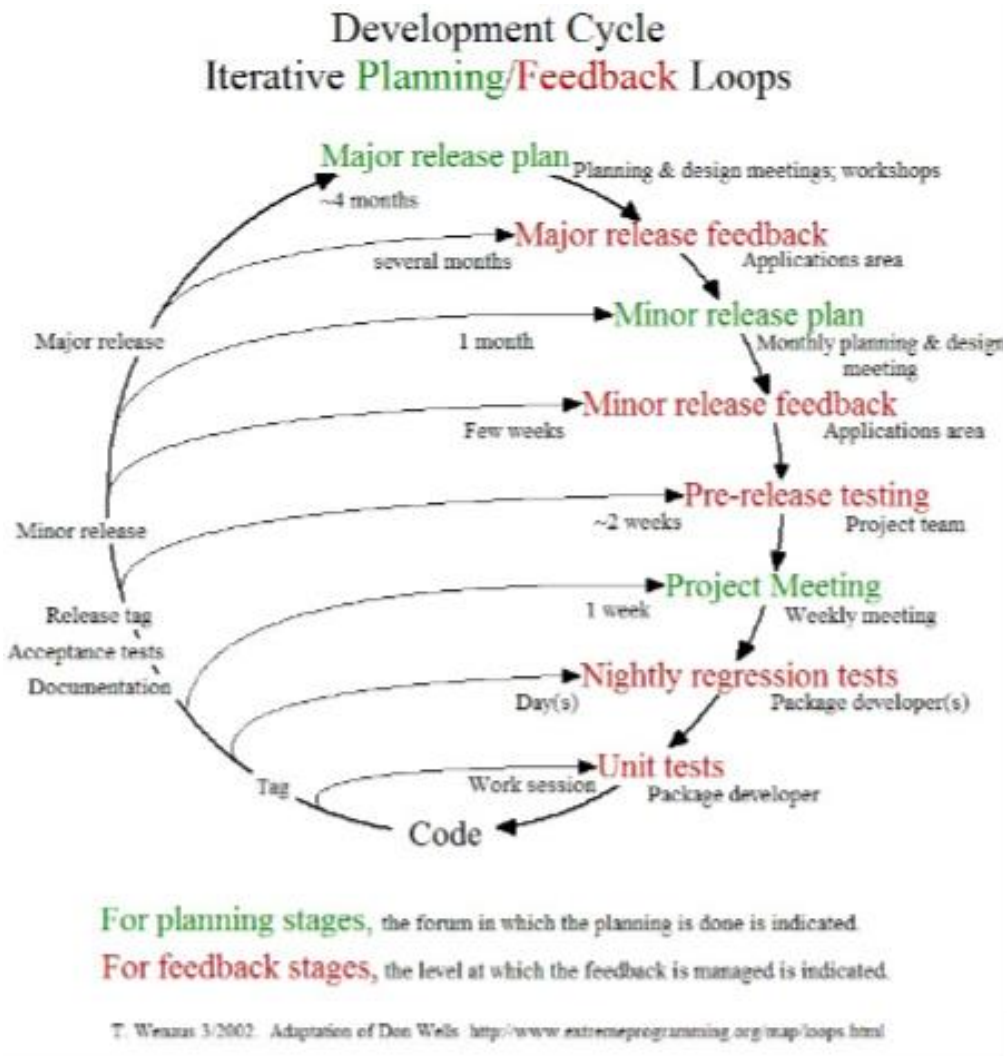
Processo de Software

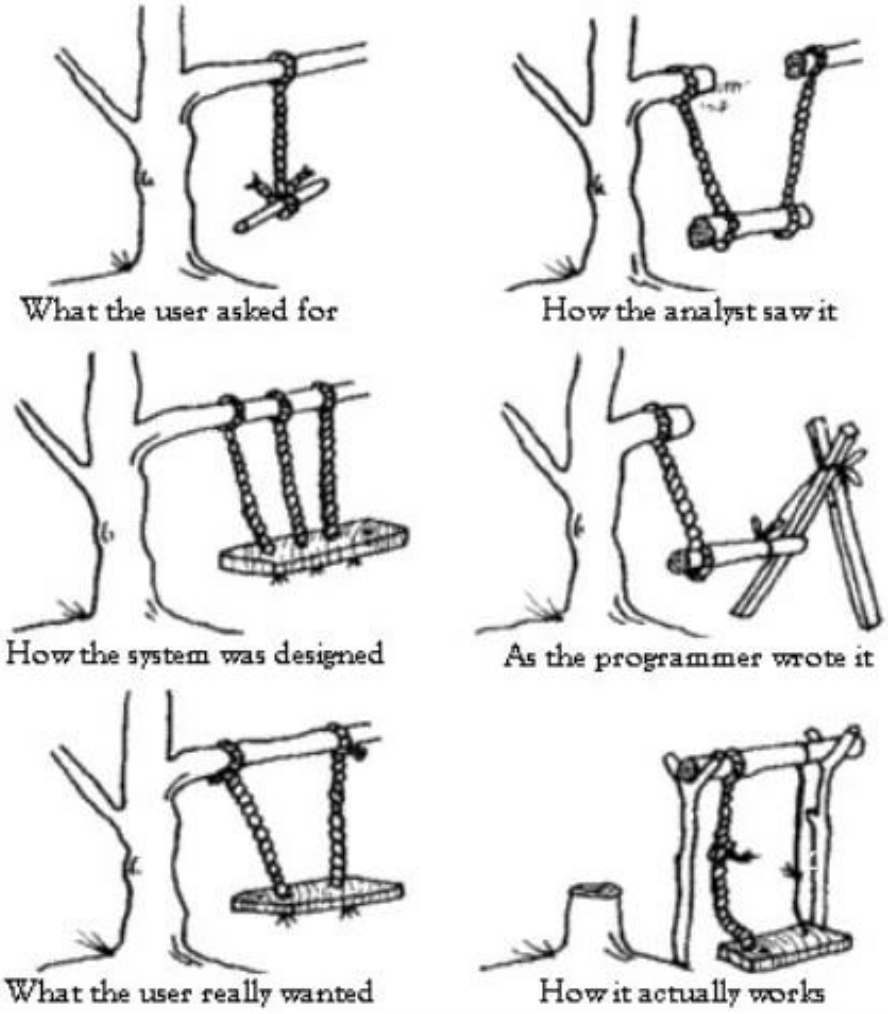
- O processo de software consiste em um conjunto estruturado de atividades necessárias para **construir um sistema de software**:
  - **Especificação** - o que o sistema deve fazer e as restrições de desenvolvimento;
  - **Desenvolvimento** - produção do sistema de software;
  - **Validação** - avaliar se o software produzido confere com o esperado;
  - **Evolução** - mudar o software em resposta às novas necessidades.

Ciclo de Desenvolvimento



Ciclo de Desenvolvimento



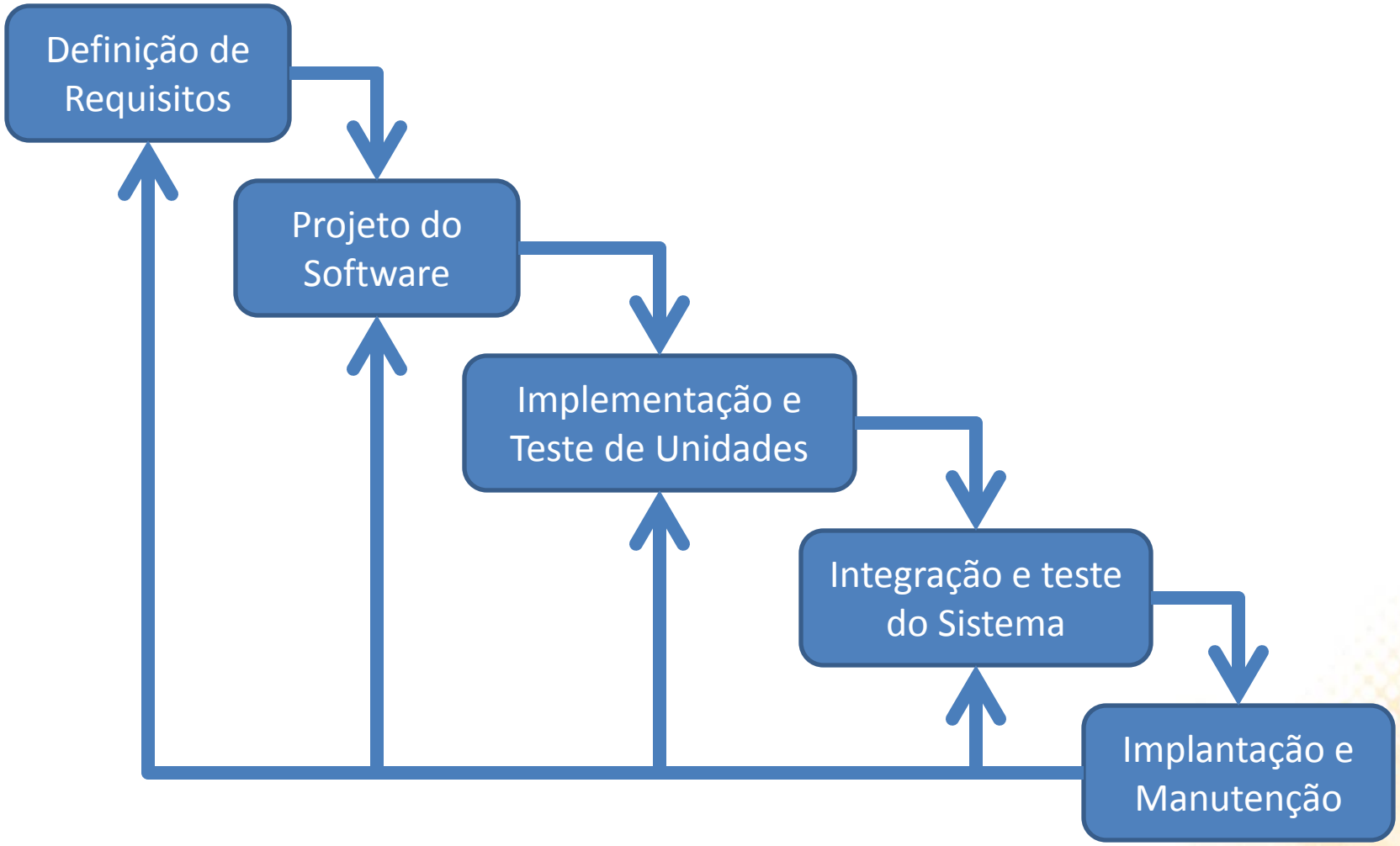


Modelo de Processo de Software

- Um modelo de processo de software é uma representação abstrata de um processo. Ele apresenta uma descrição do processo a partir de uma dada perspectiva.

- Alguns modelos:
  - **Waterfall (Modelo em Cascata):** Fases distintas e separadas de especificação e desenvolvimento;
  - **Desenvolvimento evolucionário (Modelo de Prototipação):** Especificação e desenvolvimento são inter-relacionadas;
  - **Desenvolvimento incremental:** Processo dividido em etapas que produzirão incrementalmente o sistema até a sua versão final;
  - **Desenvolvimento baseado em reuso:** O sistema é montado a partir dos seus componentes;

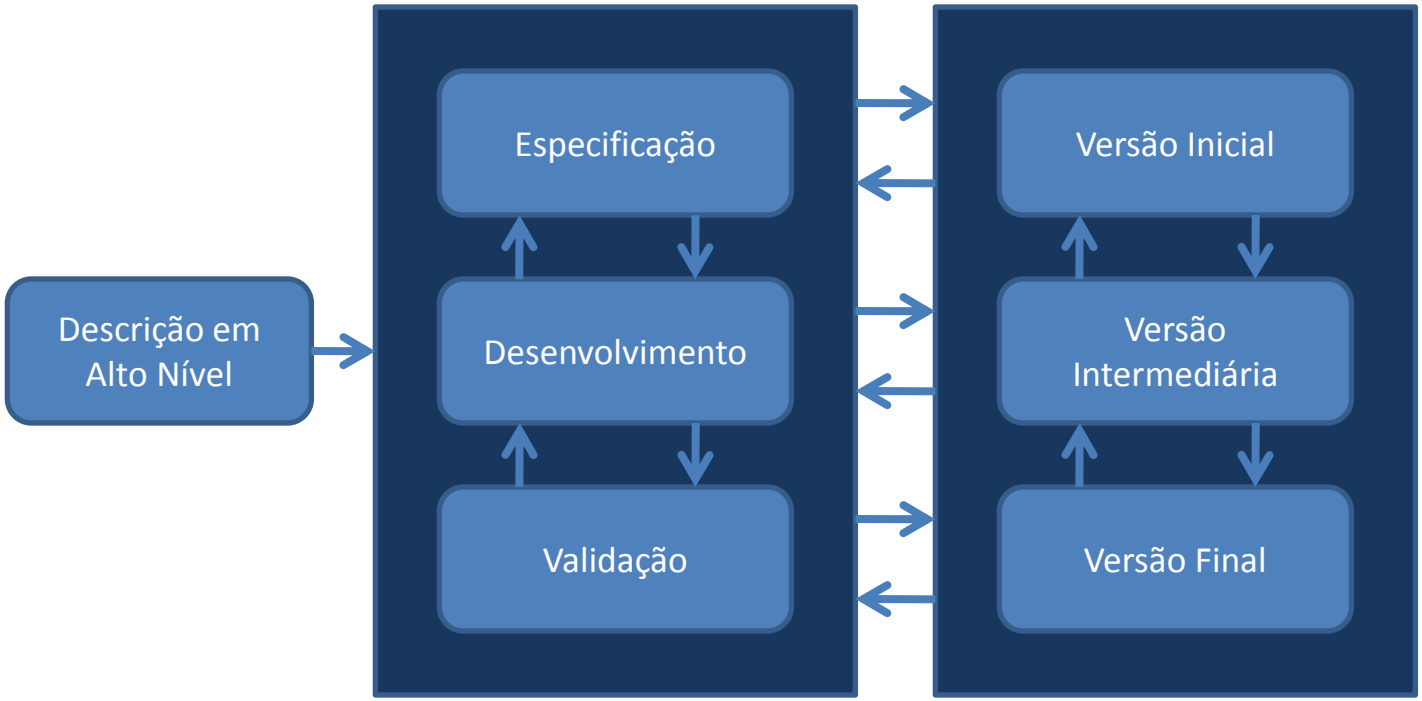
Modelo em Cascata



Modelo em Cascata

- **Problemas do Modelo em Cascata:**
  - A principal desvantagem do modelo em cascata é a dificuldade em se acomodar mudanças uma vez que o processo se iniciou - uma fase deve terminar antes que a fase seguinte possa se iniciar;
  - Apropriado apenas quando os requisitos são claros desde o início do projeto;
  - Poucos sistemas possuem requisitos estáveis;
  - Ainda assim, cerca de 40% de todos os projetos de software utilizam este modelo!

Modelo de Prototipação



Modelo de Prototipação

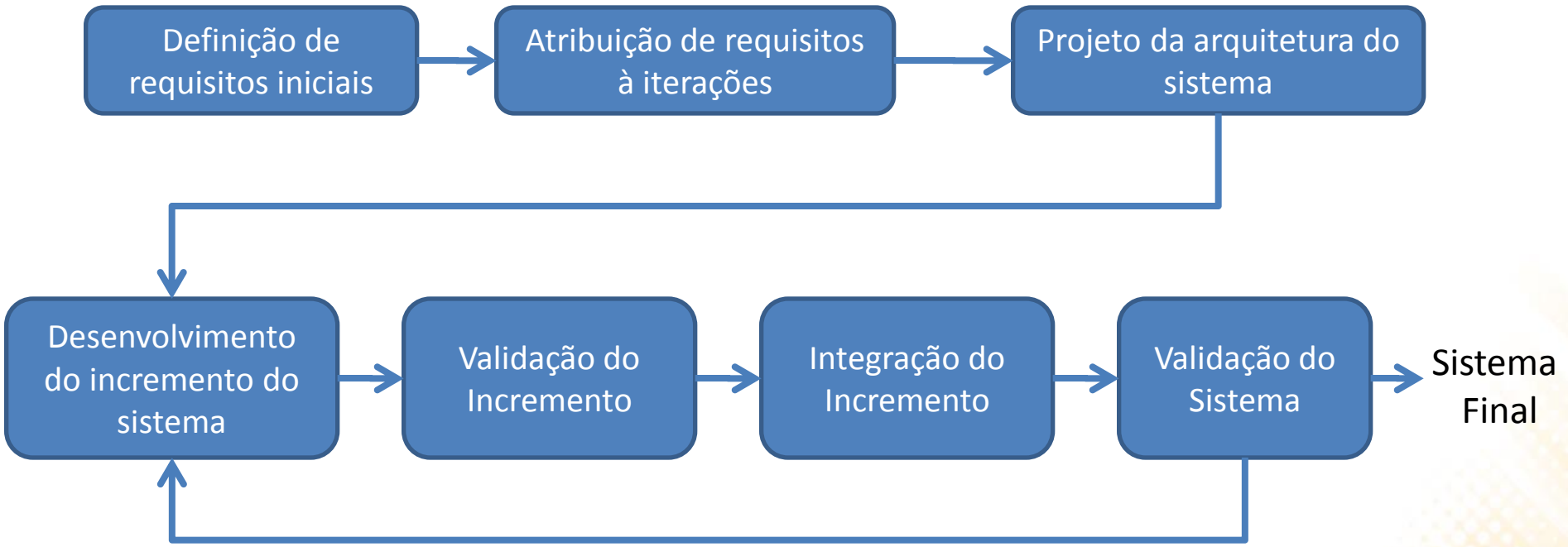
- **Desenvolvimento Exploratório:**
  - O objetivo é trabalhar com os clientes para criar iterativamente um sistema final a partir de uma especificação inicial;
  - Deve-se iniciar o processo com um conjunto de requisitos muito bem entendidos;
  - Novas características são adicionadas a medida que vão sendo propostas pelo usuário
- **Protótipo Descartável:**
  - Tem como objetivo o entendimento dos requisitos do sistema

Modelo de Prototipação

- **Problemas:**
  - Falta de visibilidade;
  - Sistemas possuem geralmente uma estrutura pobre;
  - Habilidades em linguagens de prototipação rápida podem ser necessárias.

- **Aplicabilidade**
  - Em projetos de pequenos e de médio tamanho;
  - Em partes de sistemas mais complexos (ex: interfaces do usuário);
  - Em programas de curto ciclo de vida.

Modelo de Desenvolvimento Incremental



Modelo de Desenvolvimento Incremental

- **Vantagens:**
  - Uma parte usável do sistema é entregue ao cliente a cada iteração (incremento);
  - Incrementos iniciais podem ser usados como protótipos para clarificação de requisitos;
  - Baixo risco de falha geral do projeto;
  - Os sub-sistemas de mais alta prioridade tendem a passar por testes mais intensos.

- **Problemas:**
  - Número de iterações não pode ser definido no início do processo;
  - O fim do processo não pode ser previamente definido.

Métodos Ágeis

- Baseados no modelo incremental, porém mais “leves” e centrados no ponto de vista das pessoas envolvidas:
  - Cada fase demora dias e não semanas;
  - Envolvidos ficam presentes numa mesma sala;
- Enfatizam trabalho no software como uma medida primária de progresso:
  - Utiliza feedback ao invés de planejamento como mecanismo primário de controle;
  - Disponibilização regular de versões do software;

Métodos Ágeis - Exemplos

- **Extreme Programming (XP):**
  - Fases pequenas e rápidas (alguns dias);
  - Testes são automatizados: metas para desenvolvimento;
  - Programação feita em duplas;
  - Projeto e arquitetura surgem por refactoring;

- **SCRUM:**
  - Usado no gerenciamento de projetos de software;
  - Ciclos formados por várias interações;
  - Breves reuniões diárias (daily scrum);

Métodos Ágeis

- **Aplicabilidade:**
  - Mais adequados quando os requisitos estão emergindo e mudando rapidamente;
  - Mais adequados para projetos com pequenos times, em torno de 20 pessoas;
  - Não são aplicáveis em sistemas críticos;

Arquitetura de Software

- O processo de projeto para identificação de **subsistemas** do sistema principal e a definição da comunicação destes subsistemas chama-se de **projeto arquitetural**.

- O resultado deste processo é a descrição da **arquitetura do software**.

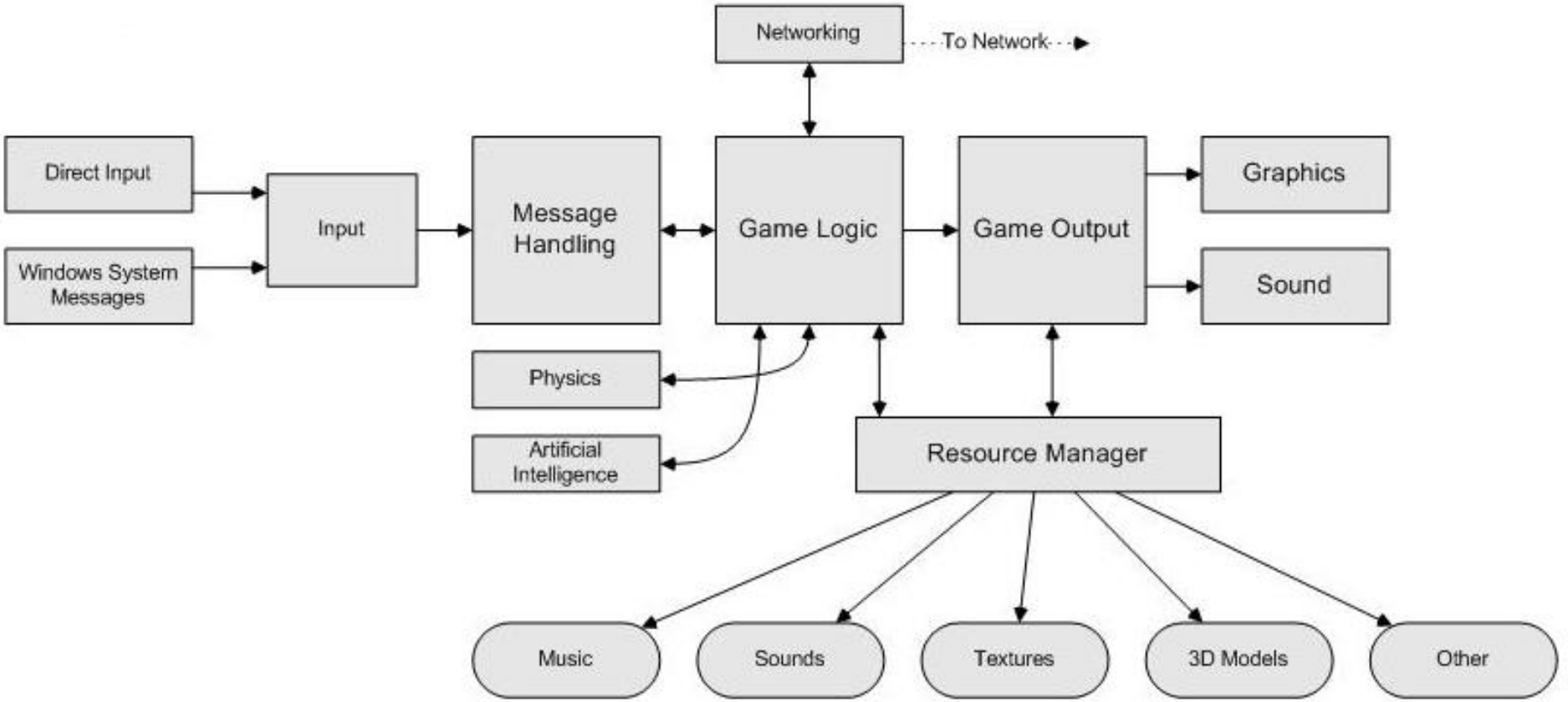
- **Decomposição modular:**
  - Os subsistemas identificados são decompostos em módulos.

Arquitetura de Software

- Um módulo é um componente de um sistema que dá suporte, através de um conjunto de serviços, para outros componentes, mas não é considerado como um sistema em separado.

- Os módulos de um sistema podem ser projetados de varias formas:
  - Cliente-Servidor
  - Camadas
  - Pipes e Filtros
  - Plugins

Exemplo de Módulos



Bibliografia

- Sommerville, I. **Engenharia de Software**; Prentice Hall: Addison-wesley, 2003.