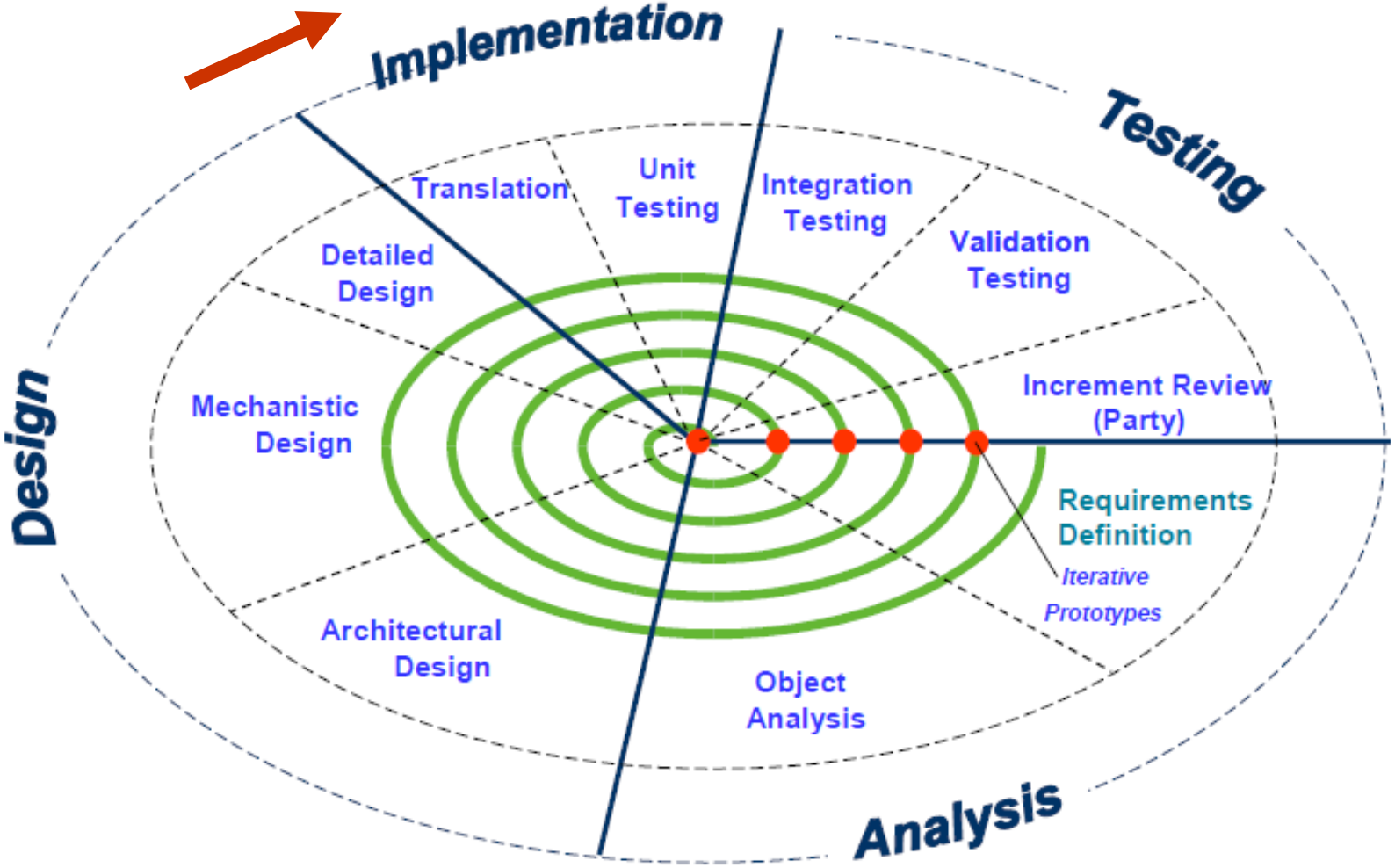
Engenharia de Software

Implementação

Luís Morgado

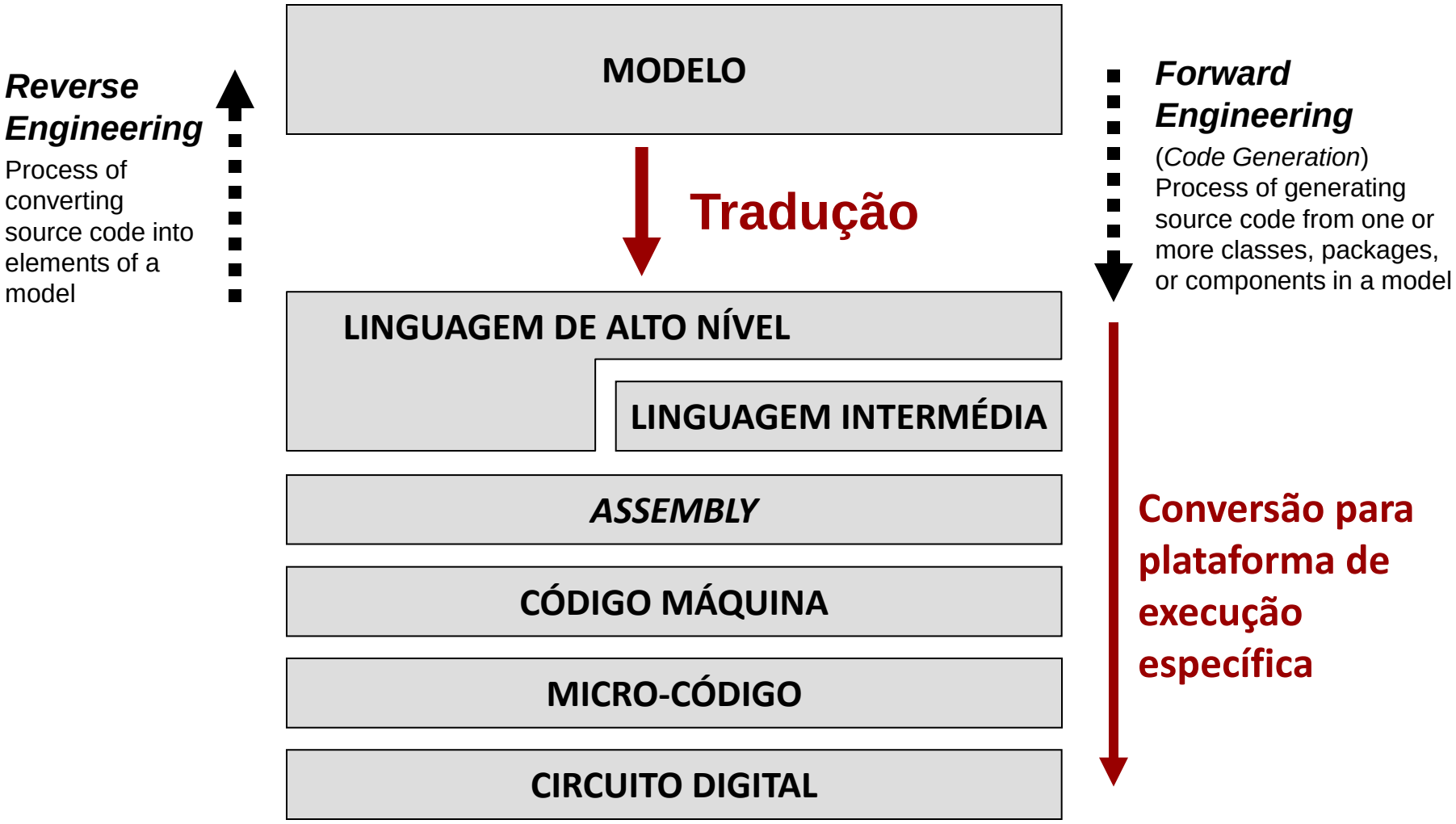
Instituto Superior de Engenharia de Lisboa
Departamento de Engenharia de Electrónica e Telecomunicações e de Computadores

Processo de desenvolvimento



[Douglass, 2006]

Níveis de Especificação



Níveis de Especificação

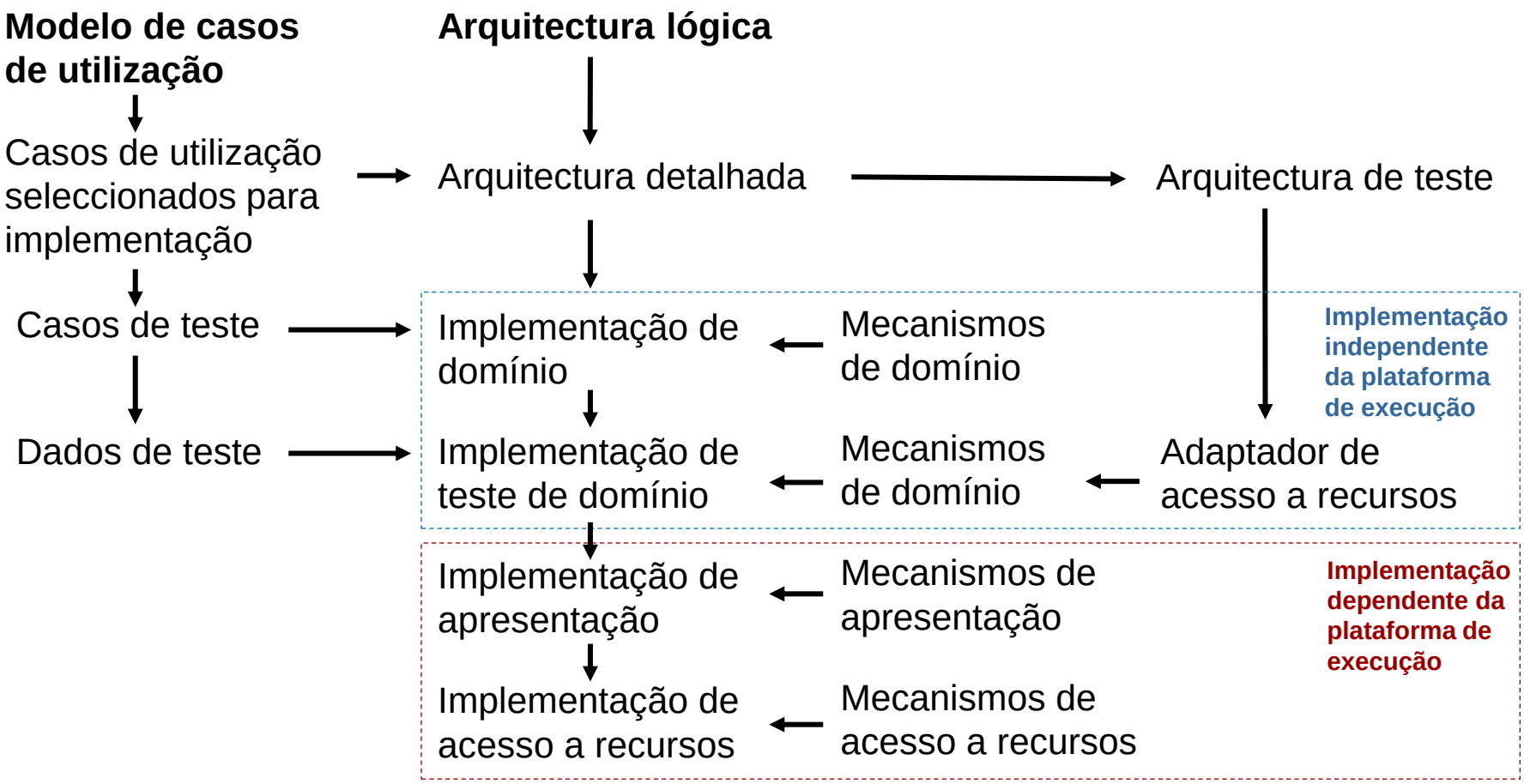
- **Três níveis principais de especificação**
  - **Independente do Modelo Computacional (CIM)**
    - Também designado ***Modelo de Negócio*** ou de ***Domínio***, descreve o contexto de utilização do sistema e o seu comportamento e características esperadas

  - **Independente da Plataforma de Execução (PIM)**
    - Descreve o sistema com tanto detalhe quanto possível de forma **independente da plataforma de execução**

  - **Específico da Plataforma de Execução (PSM)**
    - Descreve a concretização do sistema para uma **plataforma de execução específica**

Processo de Implementação

Implementação modular e incremental

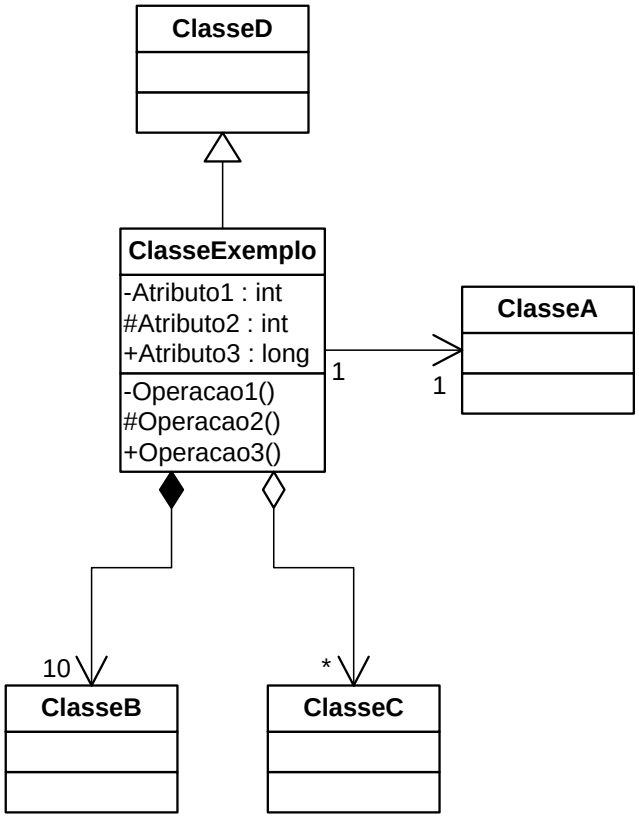


Conversão Modelo – Implementação

Análise	Projecto	Implementação
Objecto de Análise	Classe ou Mecanismo	Uma ou mais classes
Comportamento de um objecto	Operação	Método
Atributo (classe)	Atributo (classe)	Variável estática da classe
Atributo (instância)	Atributo (instância)	Variável da instância
Associação	Associação (detalhada)	Referência
Interacção entre objectos	Mensagem/Evento	Chamada de um método
Caso de utilização	Caso de Utilização (detalhado) Realização de Caso de Utilização	Sequência de chamadas de métodos
Subsistema	Subsistema	Ficheiro ou conjunto de ficheiros

Implementação de Estrutura

Exemplo

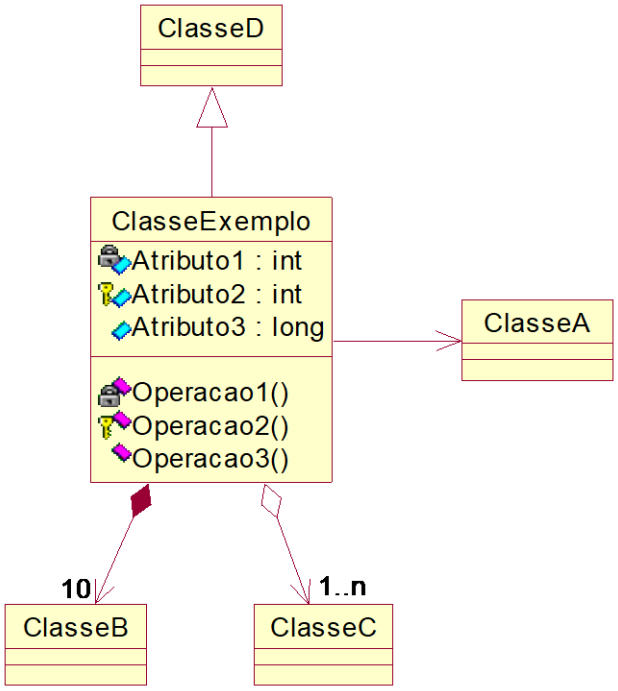


Visio for Enterprise Architects 2005 Beta
(*Visual Studio 2005 TeamSystem Beta*)

```
public class ClasseExemplo : ClasseD
{
     private int Atributo1;
     protected int Atributo2;
     public long Atributo3;

     private ClasseA the_A;
     private ClasseB [] the_B = new ClasseB[10];
     private System.Collections.ArrayList the_C;

     public void Operacao3()
     {
     }

     protected void Operacao2()
     {
     }

     private void Operacao1()
     {
     }
}
```

Implementação de Estrutura

Exemplo

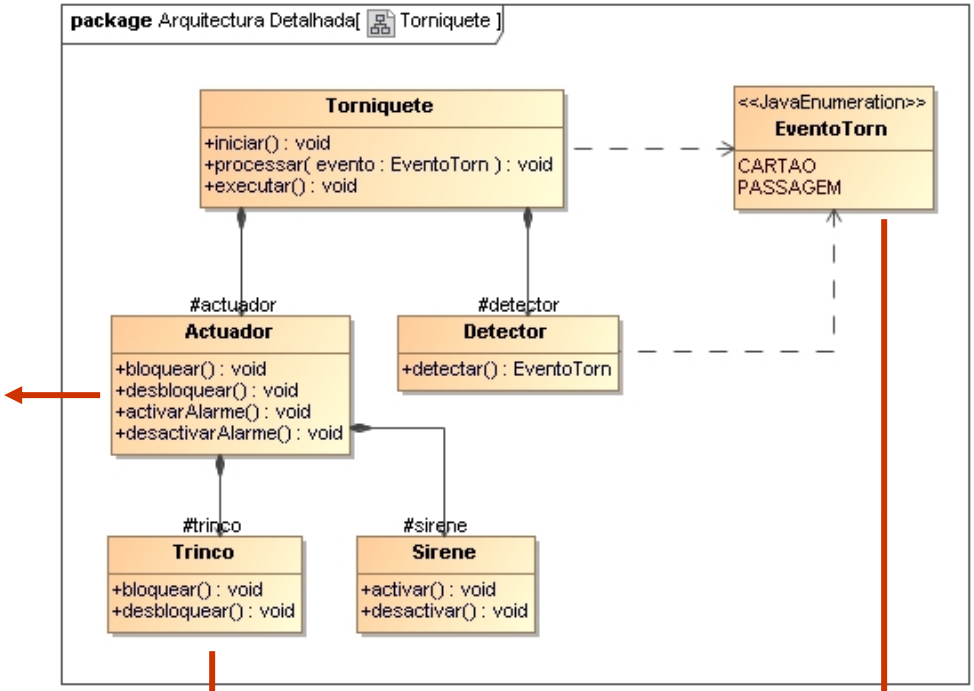


IBM Rational Rose

```
class ClasseExemplo : public ClasseD
{
  private:
     int Atributo1;

  protected:
     int Atributo2;

  public:
     long Atributo3;

  private:
     void Operacao1 ();

  protected:
     void Operacao2 ();

  public :
     void Operacao3 ();

  private:
      ClasseA *the_ClasseA;
      ClasseB the_ClasseB[10];
      UnboundedSetByReference<ClasseC> the_ClasseC;
 };
```

Exemplo: Implementação de Estrutura

```
public class Actuador
{
    protected Sirene sirene = new Sirene();
    protected Trinco trinco = new Trinco();

    public void bloquear() {
        trinco.bloquear();
    }

    public void desbloquear() {
        trinco.desbloquear();
    }

    public void activarAlarme() {
        sirene.activar();
    }

    public void desactivarAlarme() {
        sirene.desactivar();
    }
}
```




```
public class Trinco
{
    public void bloquear() {
    }

    public void desbloquear() {
    }
}
```

```
public enum EventoTorn
{
    CARTAO,
    PASSAGEM
}
```

Estrutura

- Tradução sem ambiguidade

Bibliografia

[Pressman, 2003]

R. Pressman, *Software Engineering: a Practitioner's Approach*, McGraw-Hill, 2003.

[Gamma et al., 1995]

Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides, *Design Patterns: Elements of Reusable Object-Oriented Software*, Addison-Wesley, 1995.

[Shaw & Garlan, 1996]

M. Shaw, D. Garlan, *Software Architecture: Perspectives on an Emerging Discipline*, Prentice-Hall, 1996.

[Vernon, 2013]

V. Vernon, *Implementing Domain Driven Design*, Addison-Wesley, 2013.

[Parnas, 1972]

D. Parnas, *On the Criteria to Be Used in Decomposing Systems into Modules*, Communications of the ACM 15-12, 1968.

[Kruchten, 1995]

F. Kruchten, *Architectural Blueprints - The "4+1" View Model of Software Architecture*, IEEE Software, 12-6, 1995.

[Booch *et al*., 1998]

G. Booch, J. Rumbaugh, I. Jacobson, *UML User Guide*, Addison-Wesley, 1998.

[Booch, 2004]

G. Booch, *Software Architecture*, IBM, 2004.