

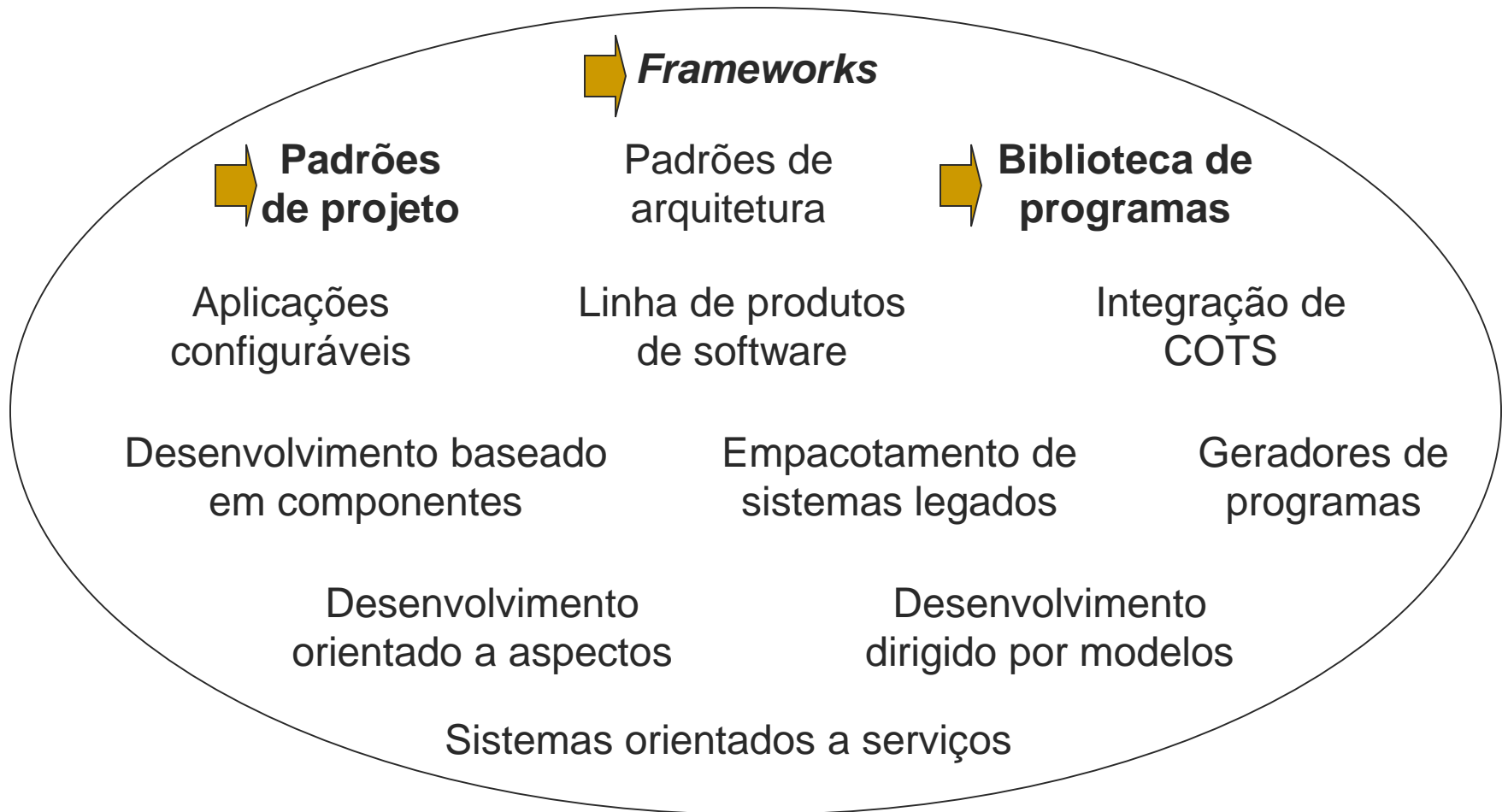


Bibliotecas, Frameworks e Padrões de Projeto

Eduardo Figueiredo

<http://www.dcc.ufmg.br/~figueiredo>

[Panorama de Reutilização]





Bibliotecas e *Frameworks*

[Bibliotecas de Software]

- Bibliotecas implementam serviços que podem ser usados por programas
 - É uma forma comum de reutilização
- Disponibiliza funcionalidades comuns a diferentes tipos de sistemas
 - Converter informação entre formatos conhecidos (e.g., string para inteiro)
 - Acesso a recursos, arquivos, BD, etc.
 - Tipos abstratos de dados: fila, pilha, lista...



[Uso de Biblioteca em Java]

```
import java.util.Vector;

public class Customer {

    String name;
    Vector phoneNumbers = new Vector();

    void removePhoneNumber(String c) {
        phoneNumbers.removeElement(c);
    }

    void addPhoneNumber(String c) {
        phoneNumbers.addElement(c);
    }

    ...
}
```

[*Frameworks*]

- *Frameworks* são aplicações incompletas
 - São formados por interfaces, classes abstratas e classes concretas (OO)
- As classes e interfaces formam uma estrutura genérica
- Um sistema é implementado pela adição de componentes para preencher lacunas
 - Por exemplo, pela implementação das classes abstratas do *framework*



[Tipos de Frameworks]

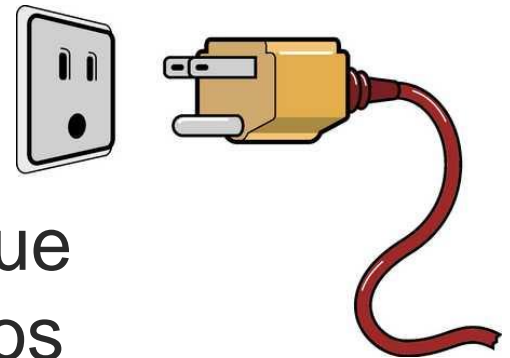
- Frameworks de infra-estrutura
 - Apoiam a criação de infra-estruturas de sistemas, tais como comunicações, interfaces de usuário e compiladores
- Frameworks de integração
 - Apoiam a comunicação e a troca de informações de componentes
- Frameworks de aplicações
 - Apoiam o desenvolvimento de um tipo de aplicações (e.g., aplicações Web)

[Extensão de Frameworks]

- Frameworks são entidades grandes que devem ser estendidas para reutilização

- Exemplos de extensão

- Adição de classes concretas que implementam métodos abstratos
- Adição (sobrescrita) de métodos que implementam comportamento padrão
- Adição de arquivos de configuração (XML)



[Principal Problema]

- *Framework* é normalmente uma entidade grande e complexa
 - Pode levar um longo tempo para entendê-lo e usá-lo efetivamente
 - O desenvolvedor pode querer apenas uma funcionalidade simples





Padrão de Projeto

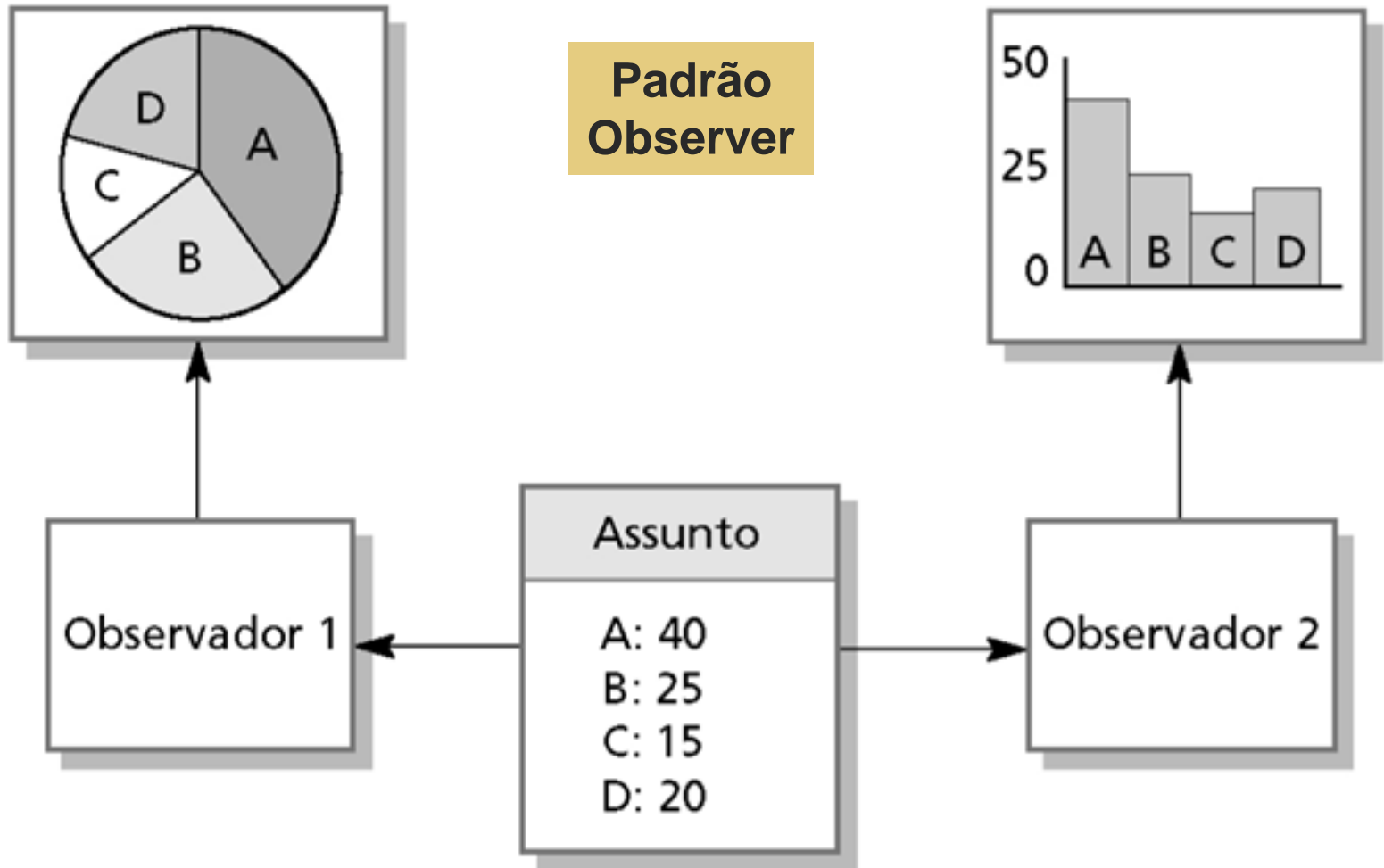
[Padrões de Projeto]

- Um padrão é uma descrição do problema e a essência da sua solução
- Documenta boas soluções para problemas recorrentes
 - Permite a reutilização de conhecimento anterior documentados em boas práticas
- Deve ser suficientemente abstrato para ser reusado em aplicações diferentes

[Elementos de um Padrão]

- Nome
 - Um identificador significativo para o padrão
- Descrição do problema
- Descrição da solução
 - Um *template* de solução que pode ser instanciado em maneiras diferentes
- Consequências
 - Os resultados e compromissos de aplicação do padrão

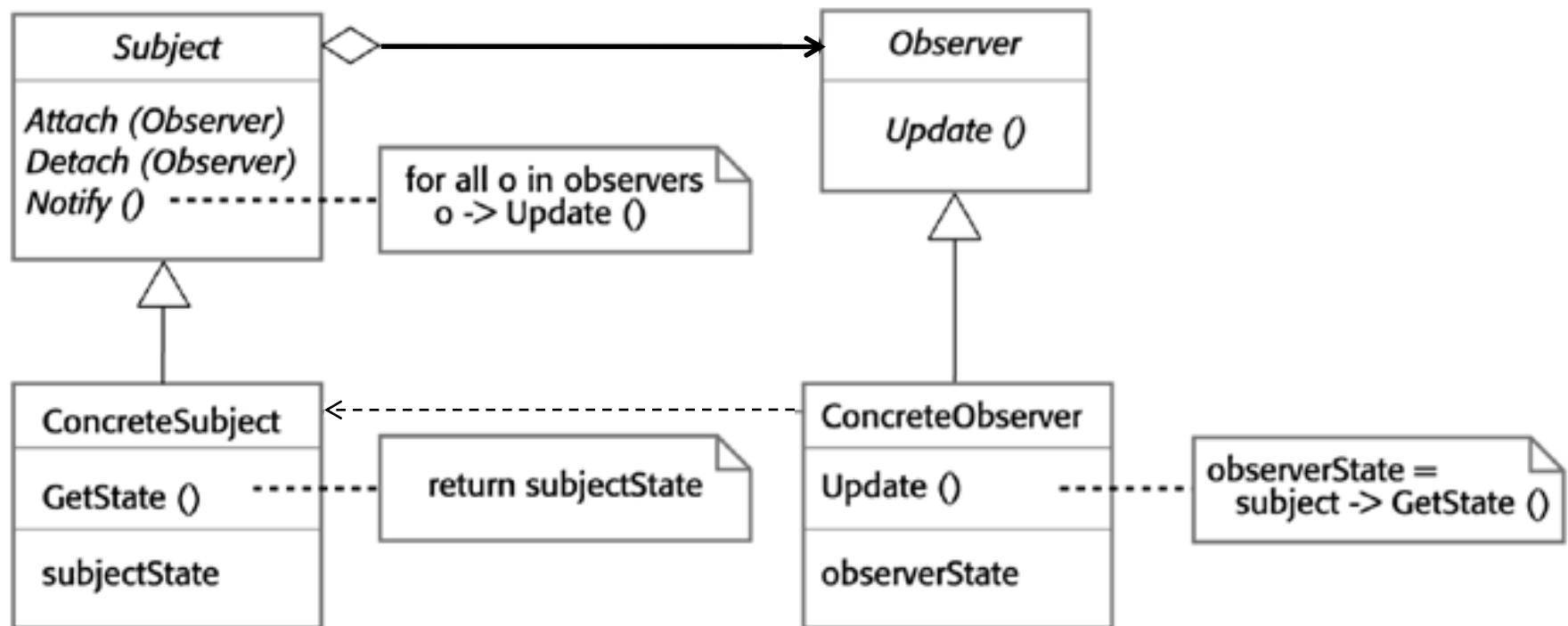
[Exemplo de Problema]



[Padrão Observer]

- Nome
 - Observer
- Descrição do problema
 - Separa o objeto de suas formas de apresentação
- Descrição da solução (próximo slide)
- Consequências
 - Otimizações para melhorar a atualização da apresentação

[Solução do Observer]



[Bibliografia]

- Ian Sommerville. **Engenharia de Software**, 9ª Edição. Pearson Education, 2011.
 - Cap. 16 Reuso de Software (Seções 16.1 e 16.2)