

Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas  
Prof. Felipe Scheidt – IFPR – Campus Foz do Iguaçu

# Frameworks

Introdução

# Objetivo

- Utilização de frameworks e de componentes de software reutilizáveis visando a otimização do processo de desenvolvimento de sistemas, diminuindo a redundância da geração de código.

# Ementa

- Conceito de **framework**;
- componentes de software reutilizáveis;
- vantagens do uso de frameworks;
- principais frameworks para desenvolvimento web;
- frameworks para persistência de dados;
- frameworks baseado em ações;
- frameworks baseados em componentes;

# O que são frameworks?

- Programação é difícil, complexa e repetitiva
- Precisamos de formas de reaproveitar código
- Frameworks: capturam funcionalidades comuns a várias aplicações.
- As aplicações devem ter algo em comum, ou seja, pertencer a um mesmo domínio de problema.
  - Exemplo: aplicações web.

# O que são frameworks?

- Um framework é um *esqueleto (estrutura)*, onde a aplicação define o *core* de operações através do preenchimento do framework (esqueleto).
- O esqueleto possui código que ligam suas partes, porém o trabalho mais importante é realizado pela aplicação.
- fornece funcionalidades genéricas.
- Pode ser seletivamente modificado através de código escrito pelo usuário, inserindo assim código específico da aplicação.

# Um estrutura para que?



# O que são frameworks?

- Um framework provê uma solução para uma família de problemas semelhantes.
- Usando um conjunto de classes e interfaces que mostra como decompor a família de problemas
- E como objetos dessas classes colaboram para cumprir suas responsabilidades
- O conjunto de classes deve ser flexível e extensível para permitir a construção de várias aplicações com pouco esforço, especificando apenas as particularidades de cada aplicação
- Observe que um framework é uma aplicação *incompleta*.
- Ao receber um framework, seu trabalho consiste em prover os pedaços que são específicos para sua aplicação

# Princípio de hollywood

- Um framework impõe um modelo de colaboração ao qual você deve se adaptar
- Já que a comunicação entre objetos já está definida, o projetista de aplicações não precisa saber quando chamar cada método: é o framework que faz isso
- O framework é usado de acordo com o Princípio de Hollywood ("*Don't call us, we'll call you*")
  - É o framework que chama o código da aplicação (que trata das particularidades dessa aplicação)



# Exemplo AWT

- Exemplo do Hollywood Principle:
  - Modelo de eventos em Java/AWT
- AWT é um framework
- No código abaixo  
mouseClicked() e mousePressed() são chamados pelo framework (AWT)

# Exemplo AWT

```
public class MeuMouseListener implements MouseListener {
    public void mouseClicked(MouseEvent event) {
        ...
    }
    public void mousePressed(MouseEvent event) {
        ...
    }
    ...
}
...
MeuMouseListener mouseListener = new MeuMouseListener();
JButton meuBotão = new JButton("clique aqui");
// O seguinte método estabelece a interação entre o objeto
// meuBotão e o objeto mouseListener
meuBotão.addMouseListener(mouseListener);
}
```

# Características de frameworks

- Um framework deve ser reusável
  - É o propósito final!
  - Para ser reusável, deve primeiro ser *usável*
    - Bem documentado
    - Fácil de usar
- Deve ser extensível
  - O framework contém funcionalidade abstrata (sem implementação) que deve ser completada
- Deve ser de uso seguro
- Deve ser eficiente
  - Devido a seu uso em muitas situações, algumas das quais poderão necessitar de eficiência
- Deve ser completo
  - Para endereçar o domínio do problema pretendido

# Exemplos de frameworks

- Frameworks para desenvolvimento web
  - php: cake, laravel, codeigniter
  - java: struts, primefaces
- Frameworks para desenvolvimento GUI
  - Swing, AWT
- Frameworks de persistência
  - Hibernate, TopLink

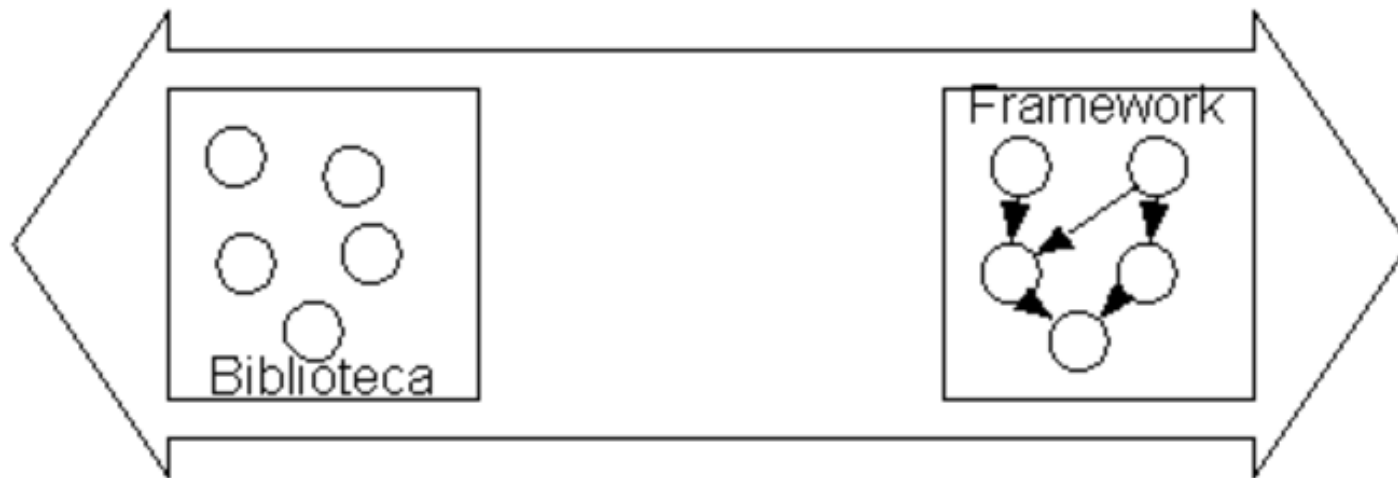
# Frameworks vs. Bibliotecas

- Importante distinguir estes dois tipos de recursos, muito utilizados em projetos de software.
- Uma biblioteca realiza atividades específicas, um conjunto de funcionalidades bem definidas.
- Exemplos:
  - jQuery;
  - commons IO;
  - JDBC;

# Bibliotecas

- É apenas uma coleção de rotinas ou classes.
- O propósito por detrás é apenas reutilização de código, i.e., obter acesso ao código que já foi escrito por outros desenvolvedores. As rotinas normalmente definem operações específicas em uma área de domínio.
- Por exemplo, bibliotecas: compressão de arquivos, manipulação de imagens, utilitários de string, expressão regular, matemática

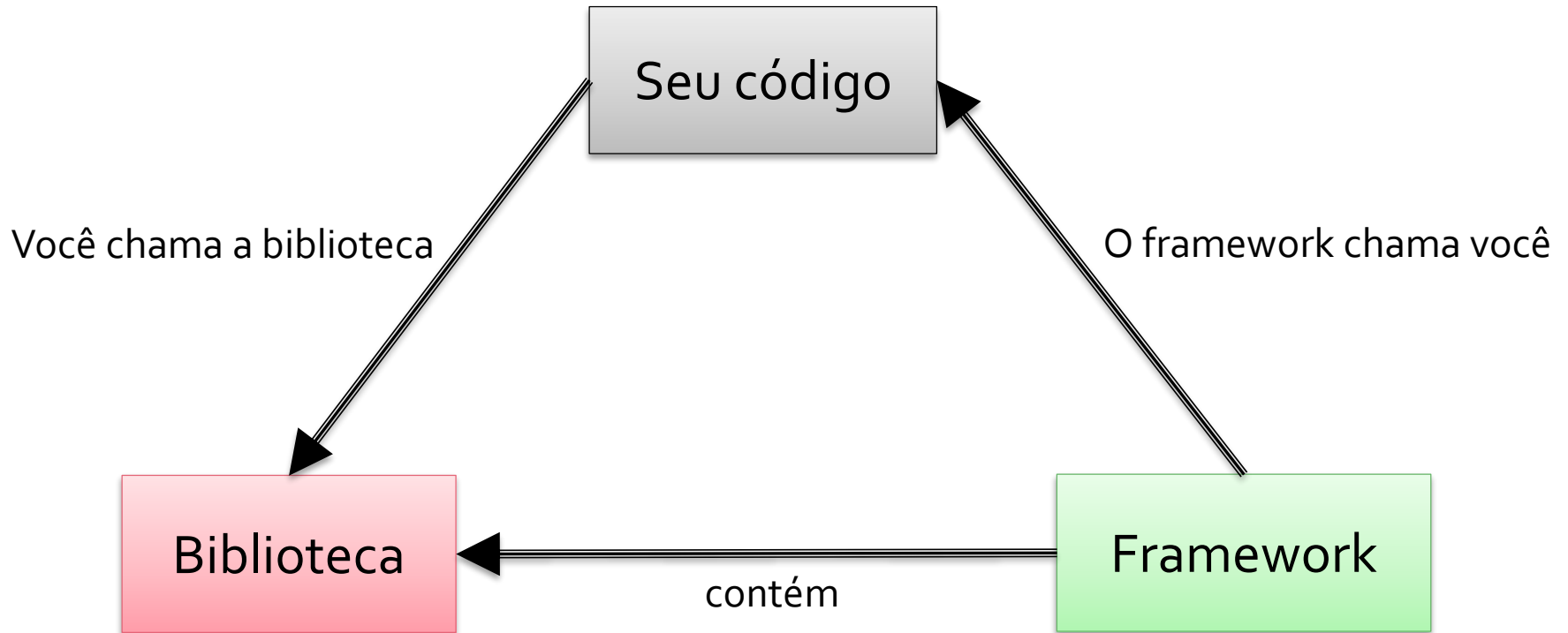
# Frameworks vs. Bibliotecas



- Classes instanciadas pelo cliente
- Cliente chama funções
- Não tem fluxo de controle predefinido
- Não tem interação predefinida
- Não tem comportamento default

- Customização com subclasse ou composição
- Chama funções da "aplicação"
- Controla o fluxo de execução
- Define interação entre objetos
- Provê comportamento default

# Frameworks vs. Bibliotecas





# Vantagens

- Segundo (BOG 2002) o uso de frameworks apresenta 3 vantagens:
  - Não gastar tempo desnecessário modelando elementos já existentes
  - Convergir todos os esforços para modelar o que é específico do projeto
  - Utilizar frameworks em diferentes projetos, garantindo com isso que as equipes partam da mesma base já conhecida e amplamente testada.

# Padrões de projetos

- Estão presentes em frameworks, bibliotecas.
- Um padrão descreve um problema que ocorre com frequência propondo uma solução genérica e reutilizável.
- Contém o somatório das melhores práticas desenvolvidas e pensadas ao longo de anos com experiências em desenvolvimento de software.

# Atividade

- Pesquise sobre frameworks web para java.
- Liste pelo menos 5 frameworks
  - Cite vantagens e desvantagens de cada um
- Com base na sua pesquisa, qual desses frameworks você recomendaria? Argumente sobre essa escolha.

# Referências

---

- <http://martinfowler.com/bliki/InversionOfControl.html>
- <http://www.dsc.ufcg.edu.br/~jacques/cursos/map/html/frame/oque.htm>
- [https://en.wikipedia.org/wiki/Software\\_framework](https://en.wikipedia.org/wiki/Software_framework)