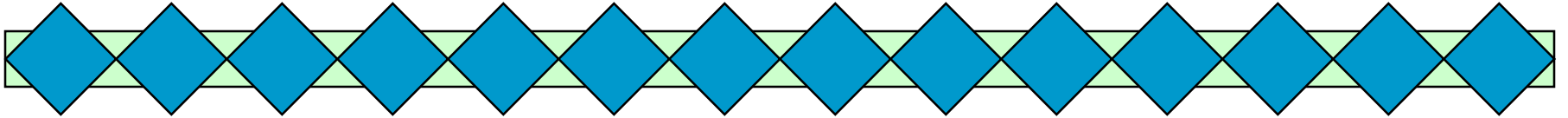


Frameworks



Profa. Dra. Rosana Teresinha Vaccare Braga



Laboratório de Engenharia de Software



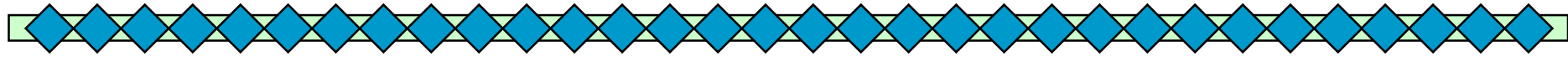
**Instituto de Ciências Matemáticas e de Computação
Universidade de São Paulo
São Carlos – SP**

Frameworks

◆ Definições:

- Aplicação semi-completa reutilizável que, quando especializada, produz aplicações personalizadas (Johnson & Foote, 1988) ou
- Coleção de classes abstratas e concretas e a interface entre elas, representando o projeto de um sub-sistema (Pree, 1995)

Frameworks

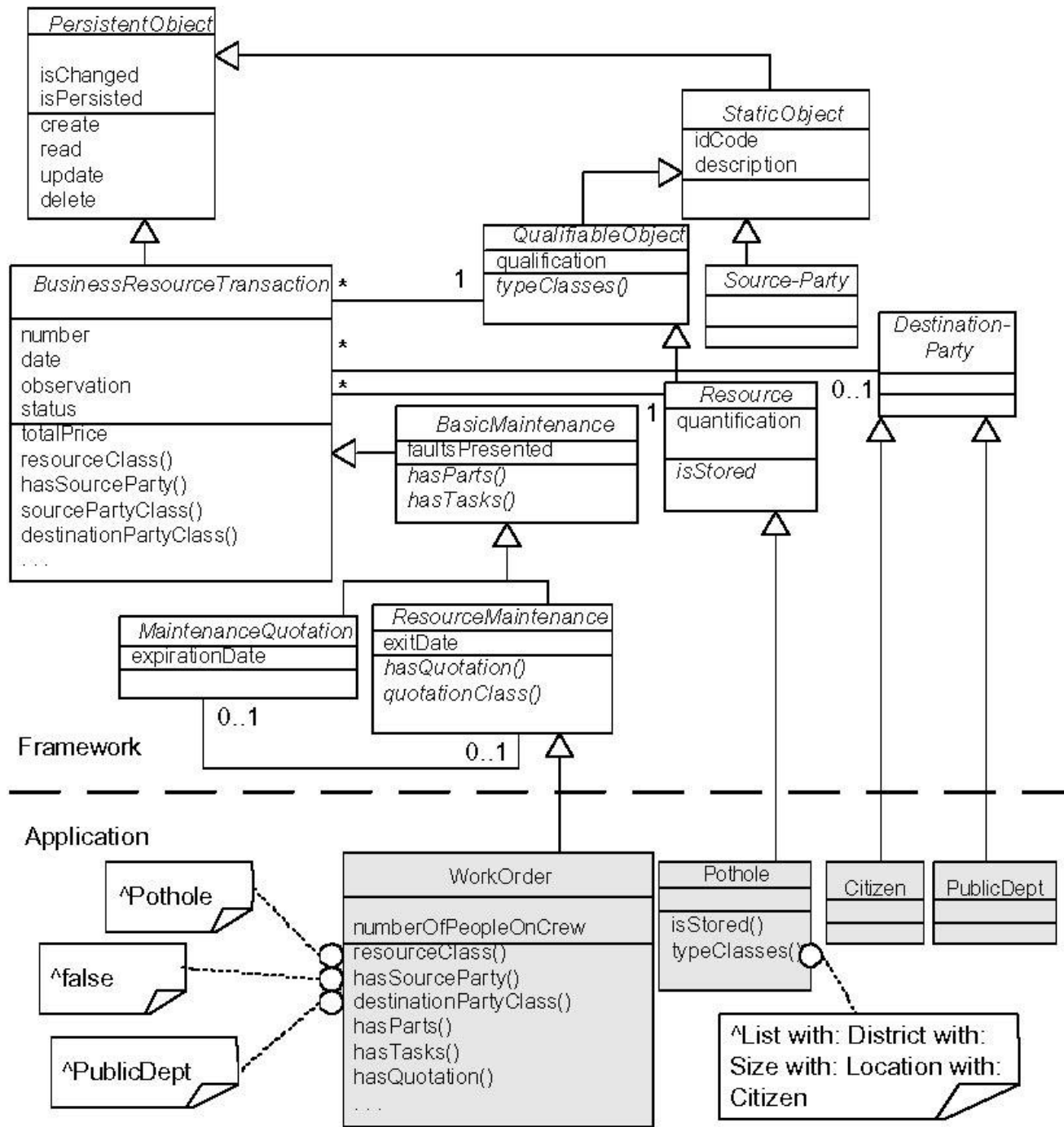


◆ Hot-Spots

- Representam as partes do framework de aplicação que são específicas de sistemas individuais
- São projetados para serem genéricos - podem ser adaptados às necessidades da aplicação

◆ Frozen-Spots

- Definem a arquitetura geral de um sistema de software - seus componentes básicos e os relacionamentos entre eles
- Permanecem fixos em todas as instanciações do framework de aplicação



Tipos de Frameworks

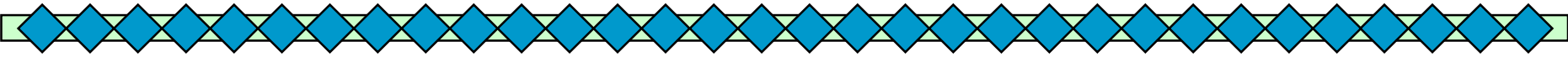
◆ Framework **caixa branca**:

- reuso por herança e associação dinâmica
- deve-se entender detalhes de como o framework funciona

◆ Framework **caixa preta**:

- reuso por composição ou definição de interfaces para os componentes.
- deve-se entender apenas a interface do cliente

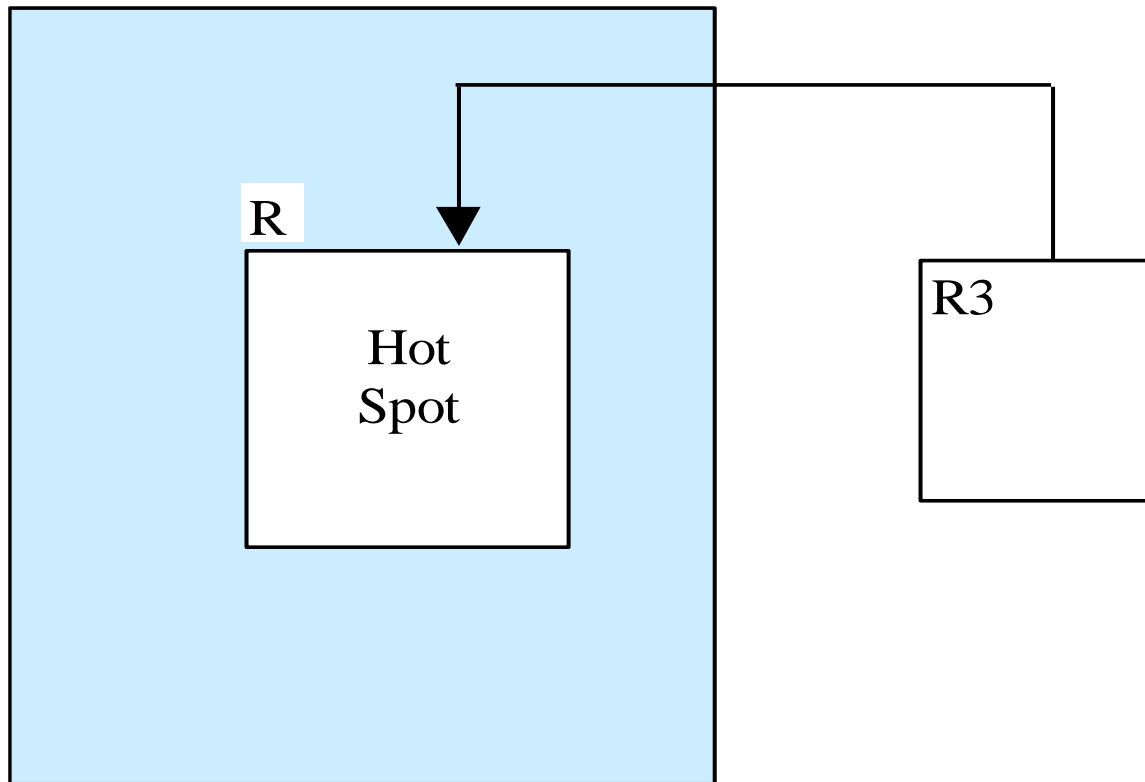
Tipos de Frameworks



- ◆ Framework **caixa cinza**:
 - combinação do caixa branca e do caixa-preta
 - reuso por herança, associação dinâmica e definição de interfaces
 - Levantamento realizado por Yassin e Fayad em 1999: 55% dos frameworks caixa-cinza, 30% dos frameworks caixa-branca e 15% dos frameworks caixa-preta

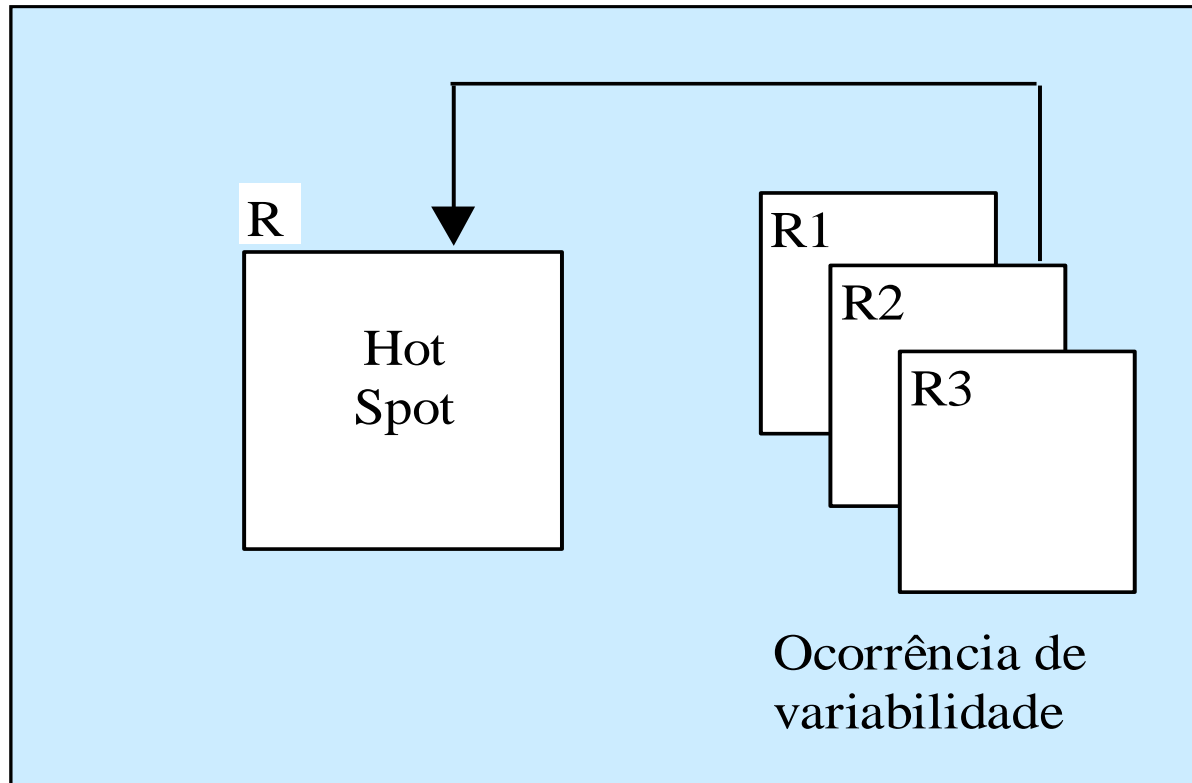
Tipos de Frameworks

Framework Caixa Branca



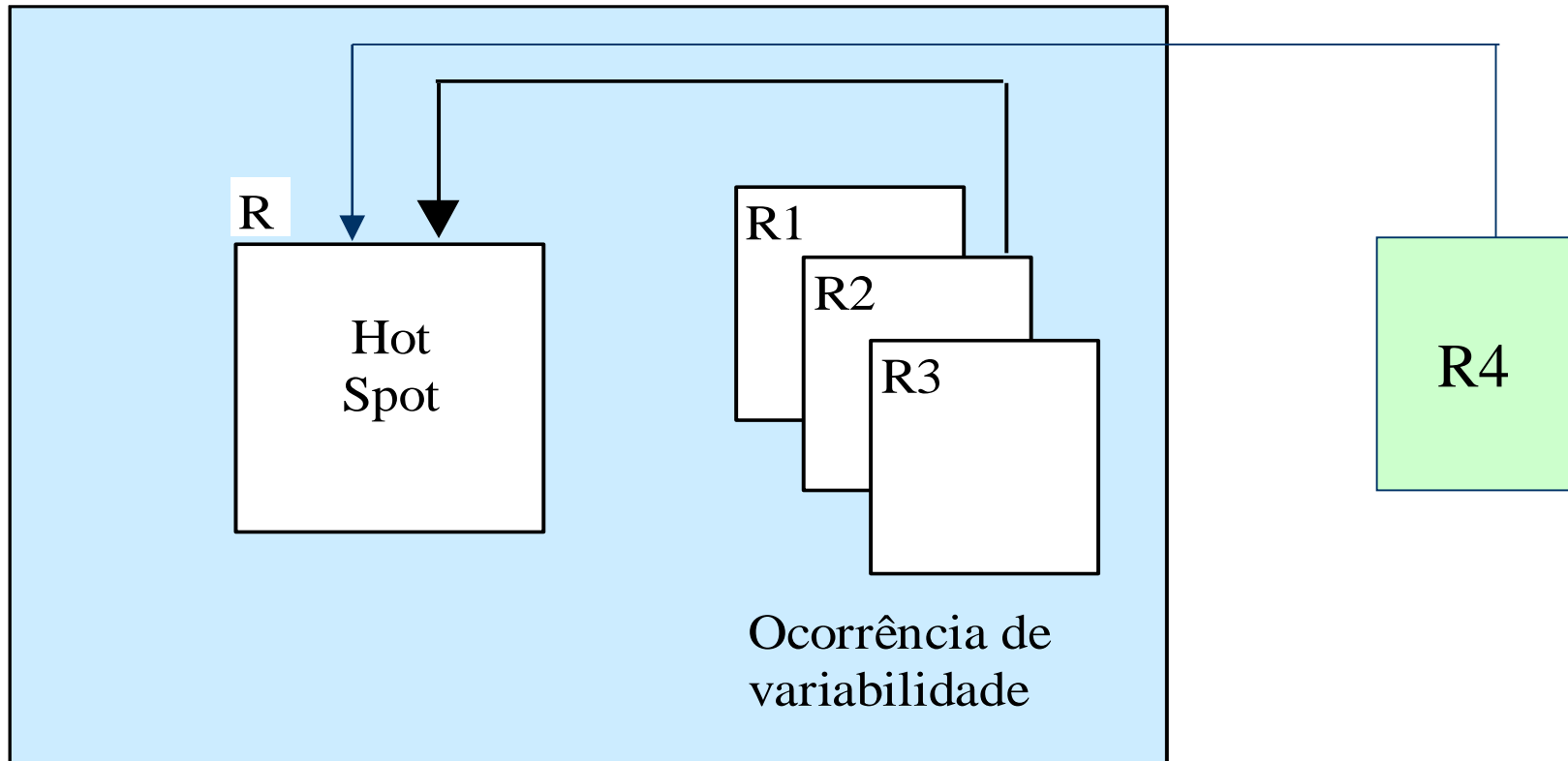
Tipos de Frameworks

Framework Caixa Preta



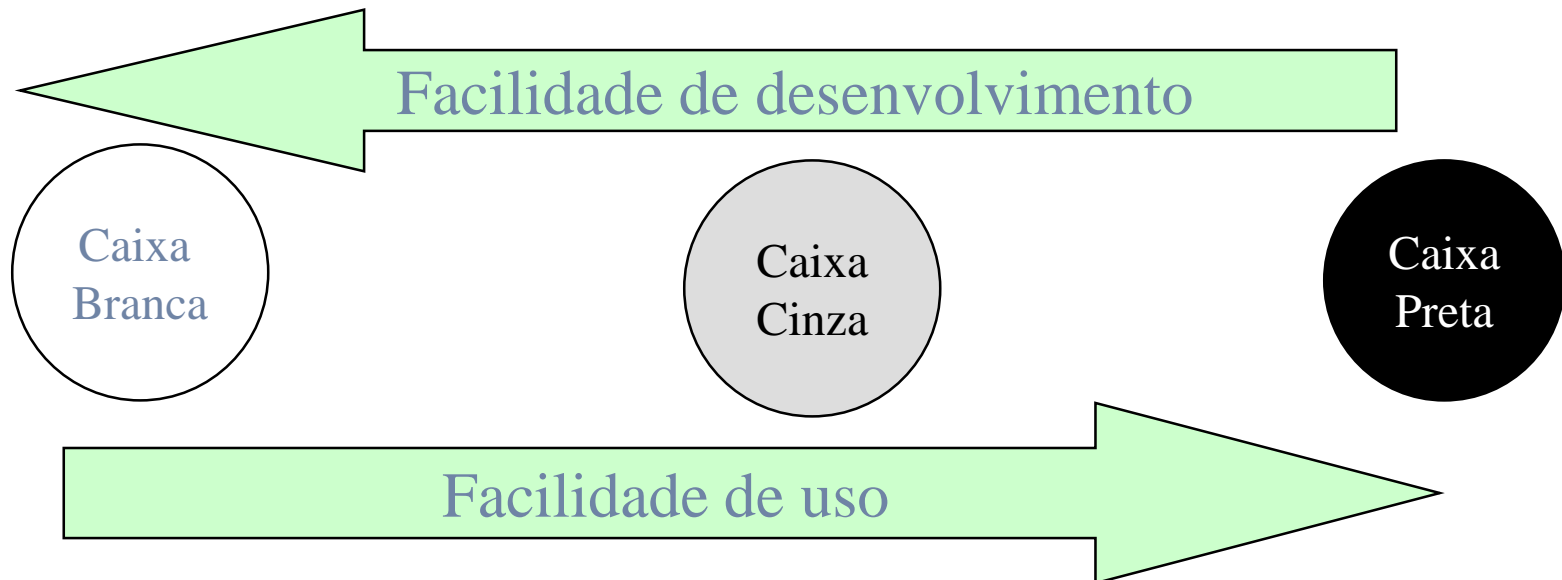
Conceitos Básicos

Framework Caixa Cinza



Tipos de framework

- ◆ framework caixa branca é mais fácil de projetar
- ◆ framework caixa preta é mais fácil de usar
- ◆ frameworks caixa-branca evoluem para se tornar mais caixa preta



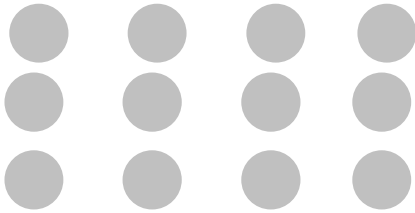
Frameworks

Inversão de Controle

- ◆ **Biblioteca de subrotinas**
 - Programa do usuário chama código reusado
 - Usuário projeta estrutura do programa
- ◆ **Framework**
 - Código reusado chama programa do usuário
 - Estrutura do programa determinada principalmente pelo código reusado

Frameworks

Biblioteca



- Conjunto de classes instanciadas pelo cliente
- Cliente chama funções
- Fluxo de controle não pré-definido
- Interação não pré-definida
- Não tem comportamento “default”

Framework



- Cuida da personalização através de subclasses
- Chama funções do cliente
- Controla o fluxo de execução
- Define a interação dos objetos
- Tem comportamento “default”

Classificação de Frameworks de aplicação

◆ Frameworks de Infra-estrutura do Sistema

- simplificam o desenvolvimento da infra-estrutura de sistemas portáteis e eficientes,
- exemplos: sistemas operacionais, comunicação, interfaces com o usuário e ferramentas de processamento de linguagem
- em geral são usados internamente em uma organização de software e não são vendidos a clientes diretamente

Classificação de Frameworks de aplicação

◆ Frameworks de Integração de Middleware

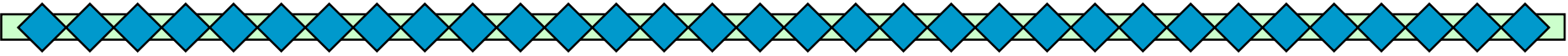
- Middleware: camada de software entre a rede e as aplicações, que provê serviços como identificação, autenticação, autorização, diretórios e segurança
- Frameworks de middleware: usados em geral para integrar aplicações e componentes distribuídos.
- projetados para melhorar a habilidade de desenvolvedores em modularizar, reutilizar e estender sua infra-estrutura de software para funcionar “sem costuras” em um ambiente distribuído
- exemplos: Object Request Broker (ORB), middleware orientado a mensagens e bases de dados transacionais

Classificação de Frameworks de aplicação

◆ Frameworks de Aplicação Empresarial

- voltados a domínios de aplicação mais amplos e são a pedra fundamental para atividades de negócios das empresas.
- exemplos: telecomunicações, aviação, manufatura e engenharia financeira.
- são mais caros para desenvolver ou comprar, mas podem dar um retorno substancial do investimento, já que permitem o desenvolvimento de aplicações e produtos diretamente

Como construir frameworks

- 
- ◆ Frameworks são abstrações: as pessoas fazem generalização a partir de exemplos concretos
 - ◆ A concepção de código reutilizável requer iteração
 - ◆ Frameworks embutem conhecimento do domínio
 - ◆ Os clientes do framework são os programadores de aplicações

Como construir frameworks

◆ Para generalizar:

- encontre coisas com nomes diferentes mas que sejam as mesmas
- parametrize para eliminar diferenças
- quebre coisas grandes em coisas pequenas, de forma que componentes similares possam ser encontradas
- categorize coisas similares

Uma linguagem de padrões para construção de frameworks



- ◆ “Evolving frameworks”

[Roberts e Johnson 1998]

- ◆ Seus padrões descrevem um caminho comum que pode ser percorrido pelos construtores de frameworks.

Uma linguagem de padrões para construção de frameworks

Three Examples

White-Box Framework

Black-Box Framework

Component Library

Hot-spots

Pluggable Objects

Fine-grained Objects

Visual Builder

Language Tools

Construa um programa gráfico que permita a especificação dos objetos que fazem parte da aplicação e de sua inter-conexão

Visual Builder



◆ Context:

- Você agora tem um framework “black-box”.
- Pode, assim, construir uma aplicação apenas conectando objetos de classes existentes.
- O comportamento da sua aplicação é totalmente determinado por como esses componentes são ligados.
- Uma aplicação consiste de duas partes: a primeira diz, através de um “script”, como conectar os objetos e depois acioná-los. A segunda dá o comportamento dos objetos individualmente.
- O framework fornece a maioria da segunda parte e o programador deve fornecer a primeira

Visual Builder



◆ Problema

- O “script” de conexão é muito semelhante de uma aplicação para outra. Apenas os objetos específicos são diferentes.

◆ Forças

- Os componentes são complicados e difíceis de entender e gerar
- construir ferramentas é caro
- Especialistas no domínio raramente são programadores

Visual Builder

◆ Solução

- Construa um programa gráfico que permita a especificação dos objetos que vão compor a sua aplicação e de como são interligados. Ele deve gerar o código para a aplicação a partir da sua especificação

◆ Explicação dos princípios (*Rationale*)

- Como o código é basicamente apenas um “script”, a ferramenta pode gerá-lo automaticamente
- Tornará também o framework mais fácil de usar.
- Assim, especialistas de domínio podem criar aplicações apenas manipulando imagens na tela
- Raramente novas classes precisam ser adicionadas ao framework

Visual Builder

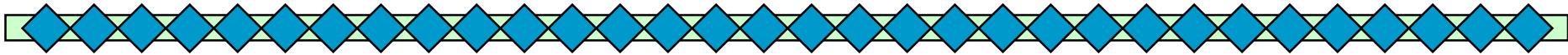
◆ Implementação

- em alguns casos a especificação pode ser feita através de caixas de diálogo e browsers.
- Usualmente, todavia, há necessidade de desenhar um gráfico para especificar domínios mais complexos. Um framework para editores gráficos como “HotDraw” pode então ser usado.

◆ Exemplos

- No Visualworks 1.0, o usuário pode “pintar” a GUI em uma área. A aplicação é criada a partir da descrição gráfica da GUI.

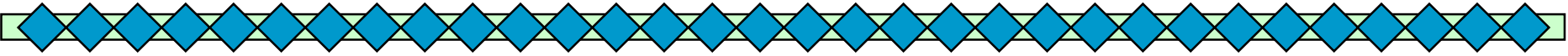
Visual Builder



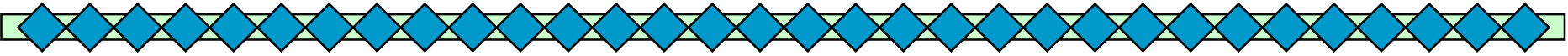
◆ Related Patterns:

- Agora que a linguagem visual já foi desenvolvida, você precisará de ferramentas de linguagem, assim como qualquer outra linguagem.

Exemplos de Frameworks

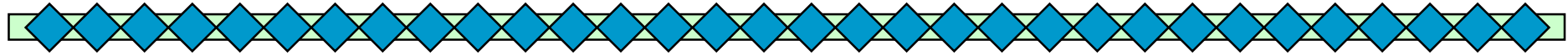
- 
- ◆ A maioria dos frameworks existentes é para domínios técnicos tais como interfaces com o usuário ou distribuição
 - MacApp: framework para aplicações Macintosh: free
 - LAMA: framework para desenvolvimento de software: free
 - Amulet: framework para para interface gráfica: free
 - Lisa Toolkit
 - Smalltalk Model View Controller
 - QT – interface gráfica
 - Hibernate, iBatis: persistência
 - Struts, Spring, .NET : Web
 - RMI – Objetos Distribuídos
 - EJB???

Exemplos de Frameworks

- 
- ◆ A maioria dos frameworks para aplicações específicas não é de domínio público
 - ET++, Interviews, ACE
 - Microsoft Foundation Classes (MFC) e DCOM
 - JavaSoft's RMI
 - Implementações do OMG's CORBA
 - IBM San Francisco: aluguel pelo tempo que durar o desenvolvimento da aplicação específica
 - PrismTech BOF e CORBA: frameworks para indústria e manufatura - preços entre US\$ 200 e US\$10.000
 - HBOC: framework para área médica: preço acima de 1 milhão de dólares.

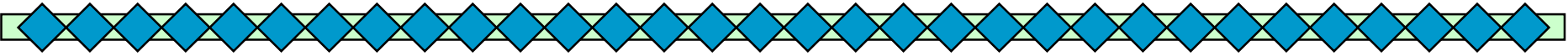
Padrões X Frameworks

X Geradores de Aplicação

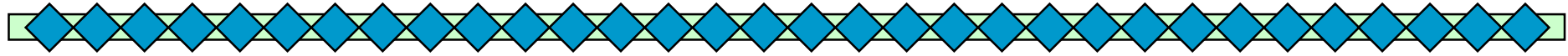


- ◆ Comparação de Frameworks com Aplicações
- ◆ Comparação de Padrões com Frameworks
- ◆ Comparação de Frameworks com Bibliotecas de programação
- ◆ Comparação de Frameworks com Geradores de Aplicação

Frameworks X Aplicação

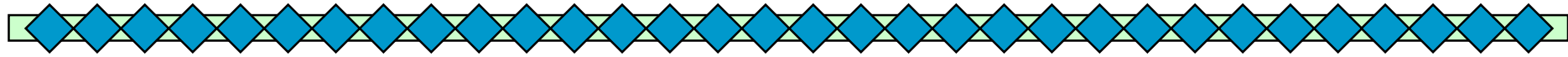
- 
- O desenvolvimento de **uma aplicação** pode ser feito com base em **mais de um framework**. Por exemplo, no desenvolvimento de uma ferramenta CASE, o framework “Model-View-Controller” pode ser usado para desenvolver a interface e o framework “Hotdraw” para desenvolver a parte de elaboração de diagramas.

Padrões X Frameworks



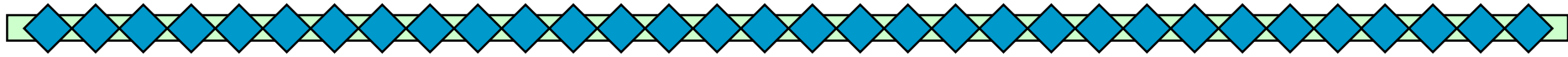
- Padrões são menores do que frameworks
 - Frameworks contém padrões mas não o inverso.
 - Padrões são formados por duas ou três classes e frameworks por muitas classes

Padrões X Frameworks



- Padrões são mais abstratos do que frameworks:
 - Um framework é um software executável, enquanto padrões representam conhecimento e experiência sobre software
 - Frameworks têm natureza física enquanto que padrões têm natureza lógica.
 - Frameworks são programados para ser executados e reusados diretamente, enquanto que padrões devem ser estudados e programados a cada uso

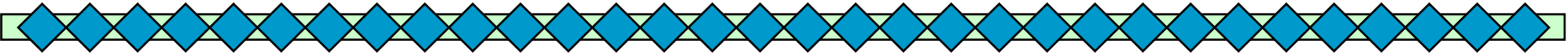
Padrões X Frameworks - cont.



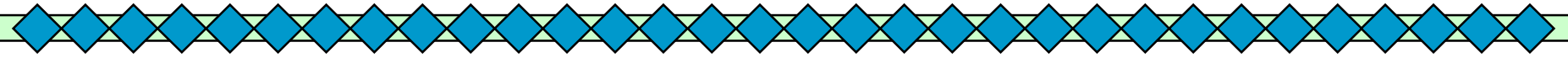
- Padrões são menos especializados do que frameworks:

- Frameworks são, em geral, desenvolvidos para um domínio de aplicação específico enquanto que padrões (de projeto) podem ser usados em diversos tipos de aplicação
 - Ressalva: frameworks de infra-estrutura podem ser usados em diversos tipos de aplicação
- Apesar de haver padrões mais especializados, eles não são capazes de ditar a arquitetura de uma aplicação.
 - Ressalva: padrões arquiteturais são capazes de mostrar a arquitetura global do sistema, embora não as implemente.

Frameworks X Biblioteca de programação

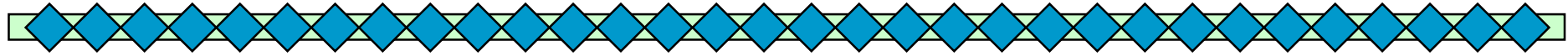
- 
- Em uma biblioteca de programação o desenvolvedor é responsável pela chamada de componentes e fluxo de controle do programa enquanto que em um framework há a inversão de papéis: o programa principal é reutilizado e o desenvolvedor decide quais componentes são chamados e deriva novos componentes. O framework determina a estrutura geral e o fluxo de controle do programa.

Construtores Visuais X Geradores de Aplicação



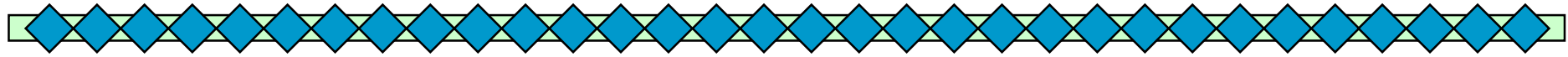
- ◆ Gerador de aplicação é um utilitário que, a partir de uma especificação de alto nível de um problema implementável, transforma automaticamente essa especificação na implementação do problema
- ◆ Construtores Visuais podem ser construídos a partir de um framework caixa-preta, pois os frameworks caixa-preta facilitam:
 - a especificação de uma aplicação com uma figura
 - a geração de código a partir de uma figura
- ◆ Linguagens de programação visuais permitem a não-programadores construir aplicações

Frameworks X Gerador de Aplicações



- Frameworks implicam o uso da orientação a objetos enquanto geradores de aplicação são independentes do paradigma de desenvolvimento.
- Ambos envolvem análise de domínio para sua construção, na qual diferentes aplicações dentro do mesmo domínio são estudadas e posteriormente generalizadas.
- Nos frameworks o código é herdado enquanto em geradores de aplicação o código é gerado.
 - Com isso, no framework todo o código é levado quando se usa o framework; já no gerador de aplicação só se leva o código relativo às funcionalidades específicas da aplicação

Frameworks X Gerador de Aplicações - cont.



- Ambos resultam em código que integrará uma aplicação específica
- Em ambos pode-se adicionar novo código à aplicação, que pode servir de retro-alimentação para uso futuro.
- Nos geradores de aplicação é necessário fornecer uma especificação de todo o sistema em alto nível para obter o produto final, enquanto no framework são apenas informados os pontos variáveis, com possível criação de novas classes, para a instanciação do framework para uma aplicação específica

Frameworks

