THE DEVELOPER'S CONFERENCE

Trilha – .Net

Bruno Gouvêa Roldão
Arquiteto de Soluções

Agenda

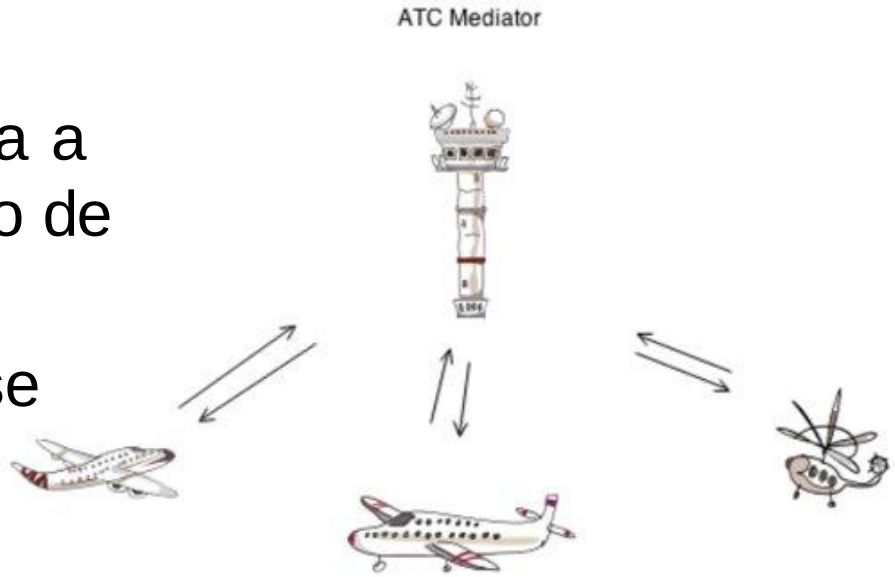
THE DEVELOPER'S CONFERENCE

- Mediator
  - O que é?
  - O que não é?
  - Pros/Contras
- Padrões de Projeto
  - Command
  - Domain Event
  - Validator
- Mediatr .Net
  - Como funciona
  - Conceitos Básicos

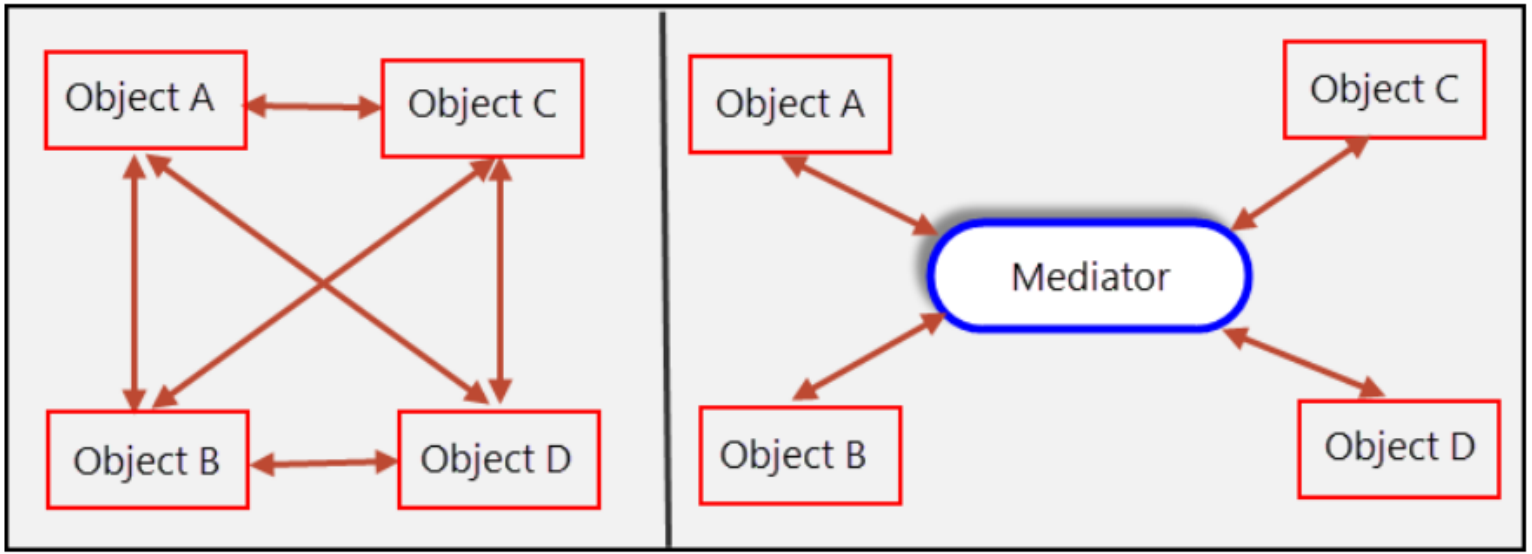
Mediator

THE DEVELOPER'S CONFERENCE

- O que é?
  - Um padrão de projetos;
  - Um objeto que encapsula a interação de um conjunto de outros objetos;
  - Impede que os objetos se refiram uns aos outros diretamente.



Mediator


Object A
Object C
Object B
Object D
Object A
Object C
Mediator
Object B
Object D

Mediator

THE DEVELOPER'S CONFERENCE

- O que **não** é?
  - Mágico;
  - Uma bala de prata;
  - Uma alternativa à injeção de dependência.

Mediator

THE DEVELOPER'S CONFERENCE

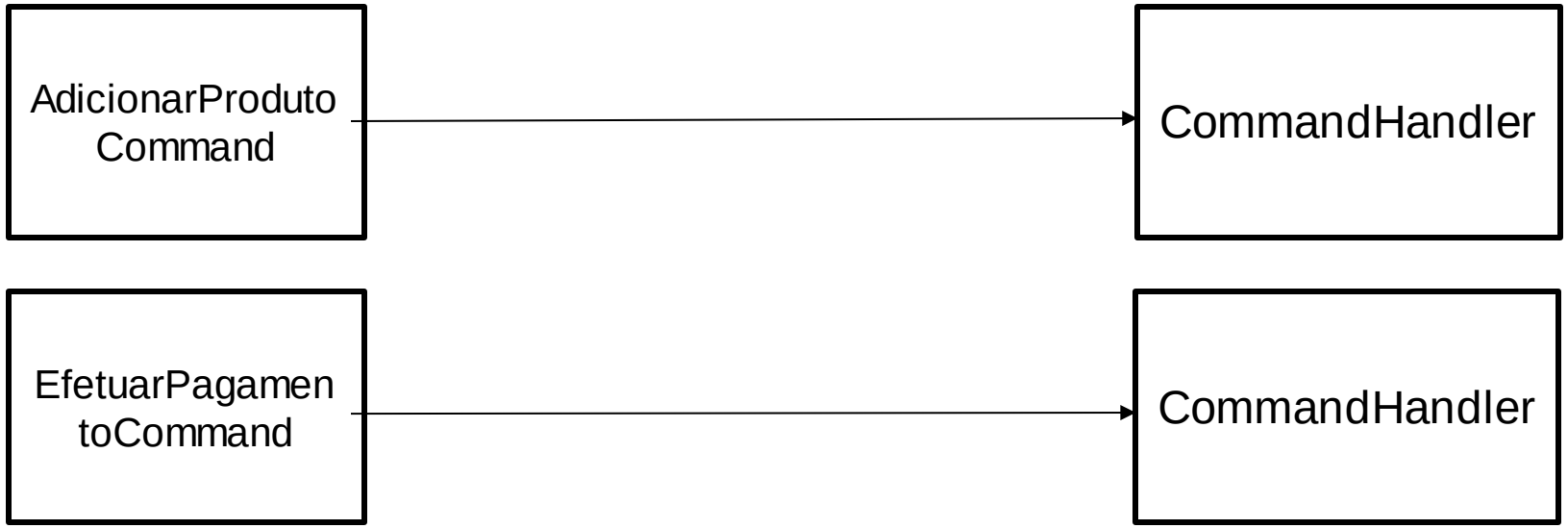
- Prós
  - Desacoplamento
    - Objetos;
    - Camadas da aplicação.
  - Reuso de componentes
  - Responsabilidade única
- Contras
  - Uso excessivo pode gerar confusão

Padrões de Projeto

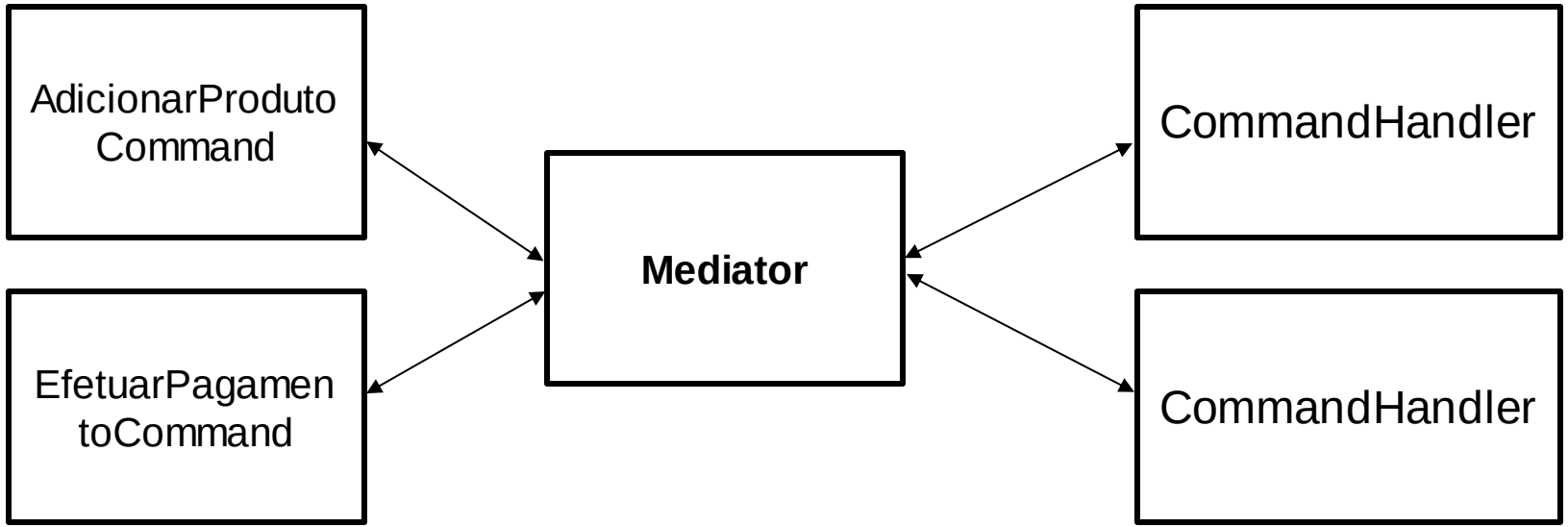
THE DEVELOPER'S CONFERENCE

- Command
  - Encapsula uma solicitação como um objeto.
  - Este objeto é processado por um handler.
  - Provê um feedback deste processamento através de um Command Result

Command


AdicionarProduto Command
CommandHandler
EfetuarPagamen toCommand
CommandHandler

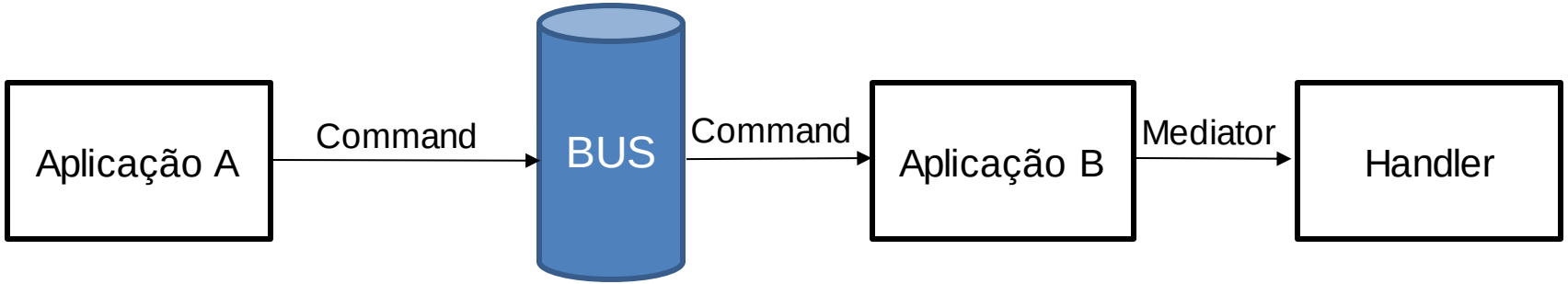
Command


AdicionarProduto Command
EfetuarPagamento Command
Mediator
CommandHandler
CommandHandler

Padrões de Projeto

THE DEVELOPER'S CONFERENCE

- Command
  - Pode ser utilizado como uma mensagem para serviços de mensageria.



Padrões de Projeto

THE DEVELOPER'S CONFERENCE

- Domain Event
  - Deve ser utilizado em conjunto com DDD.
  - Ações no seu domínio geram eventos que influenciam em outros contextos da aplicação.
  - O evento é processado por um handler, mas não espera um feedback.

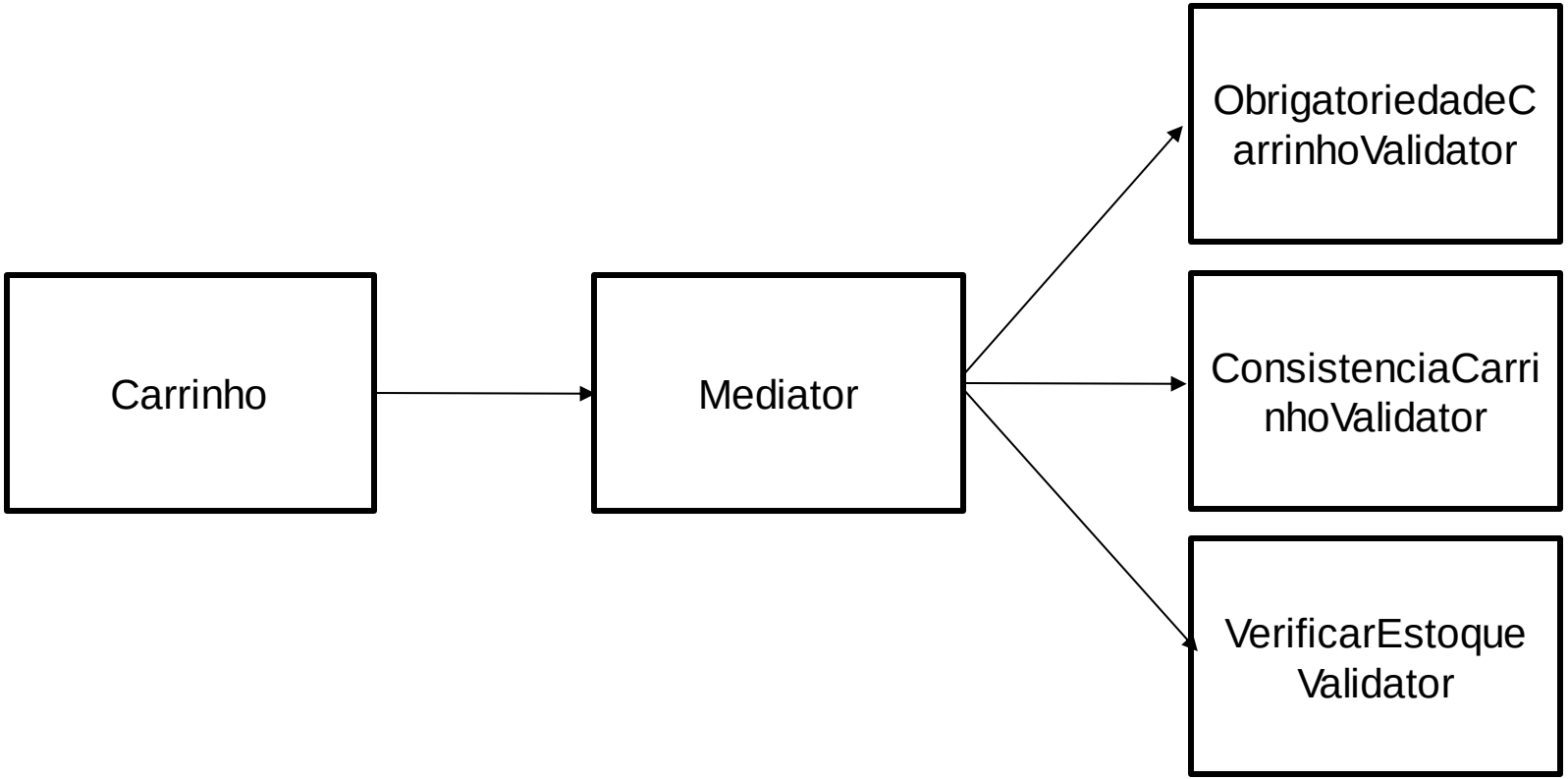
Domain Event

Padrões de Projeto

THE DEVELOPER'S CONFERENCE

- Validator
  - Um objeto possui múltiplos contextos de validação.
  - Estas validações devem ser reaproveitáveis para outros objetos do mesmo tipo.

Validator


Carrinho
Mediator
ObrigatoriedadeCarrinhoValidator
ConsistenciaCarrinhoValidator
VerificarEstoqueValidator

Padrões de Projeto

THE DEVELOPER'S CONFERENCE

- Validator
  - Pode ser executado em conjunto com outros padrões como Domain Notification, Speciffication, etc.
  - A execução dos validators pode ser síncrona ou assíncrona.
  - Geralmente utilizado em integrações.

MediatR .Net

THE DEVELOPER'S CONFERENCE

- Como funciona
  - Provê interfaces pré definidas para encapsular os objetos.
  - Faz a injeção de dependência **automática** de todos os objetos os quais ele vai mediar no setup da aplicação.
  - Necessita receber o Assembly de todos os projetos que possuem objetos a serem intermediados.

MediatR .Net

THE DEVELOPER'S CONFERENCE

- Conceitos Básicos
  - Request/Response:
    - Despachado para apenas um handler

```
public class Ping : IRequest<string> { }
```

```
public class PingHandler : IRequestHandler<Ping, string> {
    public Task<string> Handle(Ping request, CancellationToken cancellationToken) {
        return Task.FromResult("Pong");
    }
}
```

```
var response = await mediator.Send(new Ping());
Debug.WriteLine(response); // "Pong"
```

MediatR .Net

THE DEVELOPER'S CONFERENCE

- Conceitos Básicos
  - Notification:
    - Despachado para multiplos handlers (sem retorno)

```
public class Ping : INotification { }
```

```
public class Pong1 : INotificationHandler<Ping> {
    public Task Handle(Ping notification, CancellationToken cancellationToken) {
        Debug.WriteLine("Pong 1");
        return Task.CompletedTask;
    }
}
public class Pong2 : INotificationHandler<Ping> {
    public Task Handle(Ping notification, CancellationToken cancellationToken) {
        Debug.WriteLine("Pong 2");
        return Task.CompletedTask;
    }
}
```

```
await mediator.Publish(new Ping());
```

DEMO

THE
DEVELOPER'S
CONFERENCE

THE DEVELOPER'S CONFERENCE