



Fabrício Cabral <fabriciofx@gmail.com>

DDD: Repositório em uma hierarquia - onde ele deve estar localizado?

13 messages

Fabrício Cabral <fabriciofx@gmail.com>
To: UML-BR@yahoogrupos.com.br

Sun, Oct 9, 2011 at 12:45 PM

Olá todos,

trago uma pergunta que pode parecer um tanto ingênua, mas que eu eu fiz em uma outra lista, e tá dando muito pano pra manga.

A pergunta é: dada a seguinte hierarquia: classes A, B e C, onde A é abstrata, e B e C concretas e herdando de A, a quem eu devo associar uma classe de repositório (para fazer a persistência dos dados)? Associo a classe A ou devo ter dois repositórios, um pra cada classe concreta, no caso B e C?

Agradeço todo e qualquer comentário.

Atenciosamente,

—

—fx

Fabrício Cabral <fabriciofx@gmail.com>
To: ".Net Architects" <dotnetarchitects@googlegroups.com>

Sun, Oct 9, 2011 at 6:42 PM

[Quoted text hidden]

—

—fx

Daniel Moreira Yokoyama <moreira.yokoyama@gmail.com>
Reply-To: dotnetarchitects@googlegroups.com
To: dotnetarchitects@googlegroups.com

Sun, Oct 9, 2011 at 7:26 PM

O que governa a decisão pela criação dos repositórios não é a hierarquia das classes, mas sim as Agregações.

Se você está falando de DDD, e tem o livro em casa, releia os capítulos de Aggregates e Repositories.

Não importa quem herda de quem, no contexto do domínio, a herança entre classes não impera. O que imperam são as agregações, entidades e objetos de valor do modelo.
Atenciosamente,

Daniel Moreira Yokoyama.
@dmyoko
<http://twitter.com/dmyoko>

Blogs:
Getting Sharper (C#, Arquitetura de Software e outras coisas mais)
<http://gettingsharper.wordpress.com/>

Eis o Cristo... e jaz o Cristão (Cristianismo sem Religiosidade)
<http://eisocristo.wordpress.com>

Stay Sharp!

[Quoted text hidden]

[Quoted text hidden]

--

Você recebeu esta mensagem porque faz parte do grupo .Net Architects hospedado no Google Groups.
Para postar envie uma mensagem para dotnetarchitects@googlegroups.com
Para sair do grupo envie uma mensagem para dotnetarchitects+unsubscribe@googlegroups.com
Para mais opções visite o grupo em <http://groups.google.com/group/dotnetarchitects?hl=pt-br>

--

Você recebeu esta mensagem porque faz parte do grupo .Net Architects hospedado no Google Groups.
Para postar envie uma mensagem para dotnetarchitects@googlegroups.com
Para sair do grupo envie uma mensagem para dotnetarchitects+unsubscribe@googlegroups.com
Para mais opções visite o grupo em <http://groups.google.com/group/dotnetarchitects?hl=pt-br>

Alessandro de Souza <alessandro.de.souza@gmail.com>

Sun, Oct 9, 2011 at 10:28 PM

Reply-To: dotnetarchitects@googlegroups.com

To: dotnetarchitects@googlegroups.com

Matou.... é isso aí....

[Quoted text hidden]

Vinicius Quaiato <vinicius.quaiato@gmail.com>

Mon, Oct 10, 2011 at 9:58 AM

Reply-To: dotnetarchitects@googlegroups.com

To: dotnetarchitects@googlegroups.com

adorei a parte do "releia" :D

[Quoted text hidden]

Fabício Cabral <fabriciofx@gmail.com>

Mon, Oct 10, 2011 at 3:28 PM

To: dotnetarchitects@googlegroups.com

Olá Daniel,

obrigado pelo seu feedback. No entanto, acredito que fui muito sucinto ao descrever a minha dúvida. Deixar contextualizar em um exemplo, para ficar mais claro. Suponha que eu tenha três classes: Pessoa, PessoaFisica e PessoaJuridica. Pessoa é uma classe abstrata e as classes PessoaFisica e PessoaJuridica são classes concretas e herdam de Pessoa. As classes PessoaFisica e PessoaJuridica possuem suas próprias propriedades, além daquelas obviamente herdadas pela classe Pessoa. A classe Pessoa, por ser uma entidade, possui uma propriedade chamada id, que identifica cada objeto de maneira única. As classes PessoaFisica e PessoaJuridica, por herdarem da classe Pessoa, e por também serem entidades, também possuem o mesmo id.

Com o objetivo de realizar a persistência destes objetos *sem* usar um ORM, vejo as seguintes possibilidades:

1. Criar uma classe Repositório associado a classe Pessoa, onde esta poderá receber instâncias da classe PessoaFisica ou PessoaJuridica. No entanto, a dúvida é, como fazer da maneira "correta" a persistência das propriedades exclusivas destas duas classes;
2. Criar uma classe Repositório associado a cada classe concreta, no caso uma para classe PessoaFisica e outra pra PessoaJuridica.

Alguns sugeriram utilizar a abordagem (2) e outros tantos a (1), mas não explicaram como eu faria para persistir a informação exclusiva de cada classe, isto é, *não* queria ter que fazer:

```
class RepositorioPessoa
{
    void add(Pessoa p)
    {
        if (p is PessoaFisica)
        {
            // persista as propriedades X e Y
        } else if (p is PessoaJuridica)
        {
            // persista as propriedades I e J
        }
    }
}
```

Assim, não sei qual seria a solução mais adequada a este problema. Agradeço qualquer sugestão para a melhor resolução.

Atenciosamente,

2011/10/9 Daniel Moreira Yokoyama <moreira.yokoyama@gmail.com>

O que governa a decisão pela criação dos repositórios não é a hierarquia das classes, mas sim as Agregações.

Se você está falando de DDD, e tem o livro em casa, releia os capítulos de Aggregates e Repositories.

Não importa quem herda de quem, no contexto do domínio, a herança entre classes não impera. O que imperam são as agregações, entidades e objetos de valor do modelo.
Atenciosamente,

—

—fx

Daniel Moreira Yokoyama <moreira.yokoyama@gmail.com>

Mon, Oct 10, 2011 at 3:45 PM

Reply-To: dotnetarchitects@googlegroups.com

To: dotnetarchitects@googlegroups.com

Cara, leia meu primeiro e-mail pois a resposta que eu daria agora é idêntica àquela.

Em tempo: Pessoa não é uma entidade do seu domínio (e eu aposto que PessoaFisica e PessoaJuridica também não são... mas vou deixar que você decida isso por você mesmo).

Atenciosamente,

Daniel Moreira Yokoyama.
@dmyoko
<http://twitter.com/dmyoko>

Blogs:

Getting Sharper (C#, Arquitetura de Software e outras coisas mais)
<http://gettingsharper.wordpress.com/>

Eis o Cristo... e jaz o Cristão (Cristianismo sem Religiosidade)
<http://eisocristo.wordpress.com>

Stay Sharp!

[Quoted text hidden]

[Quoted text hidden]

--

Você recebeu esta mensagem porque faz parte do grupo .Net Architects hospedado no Google Groups.
Para postar envie uma mensagem para dotnetarchitects@googlegroups.com
Para sair do grupo envie uma mensagem para dotnetarchitects+unsubscribe@googlegroups.com
Para mais opções visite o grupo em <http://groups.google.com/group/dotnetarchitects?hl=pt-br>

[Quoted text hidden]

JR <agcjunior@gmail.com>

Mon, Oct 10, 2011 at 4:23 PM

Reply-To: dotnetarchitects@googlegroups.com

To: ".Net Architects" <dotnetarchitects@googlegroups.com>

Fabricio,

Concordo com o Daniel.

Se você está usando o padrão repositório como o Eric Evans sugere no seu famoso livro sobre DDD é provável que Pessoa, PessoaFisica e PessoaJuridica não façam parte do Domínio da sua aplicação. Talvez Cliente, Funcionário, Fornecedor, Aluno, etc. Tente pensar dessa forma e crie seus repositórios usando como base a raiz das agregações.

[]s

--

Você recebeu esta mensagem porque faz parte do grupo .Net Architects hospedado no Google Groups.
Para postar envie uma mensagem para dotnetarchitects@googlegroups.com
Para sair do grupo envie uma mensagem para dotnetarchitects+unsubscribe@googlegroups.com

[Quoted text hidden]

marciovmartins86 <marciovmartins@hotmail.com>

Tue, Oct 11, 2011 at 2:12 PM

Reply-To: UML-BR@yahoogrupos.com.br

To: UML-BR@yahoogrupos.com.br

Isto será um padrão para todas as classes que herdarem de A? Se for, é melhor fazer em associação com A e automaticamente serão passadas para B, C e qualquer outra classe nova que venha a surgir.

Agora, se for algo peculiar apenas a B ou C, é melhor fazer uma classe nova e apenas as classes que necessitam dessa funcionalidade herdarem isto.

-- Em UML-BR@yahoogrupos.com.br, Fabrício Cabral <fabriciofx@...> escreveu

[Quoted text hidden]

> [As partes desta mensagem que não continham texto foram removidas]
>

| [através de email](#) | [Responder através da web](#) | [Adicionar um novo tópico](#)
Mensagens neste tópico (2)

ATIVIDADE NOS ÚLTIMOS DIAS: [Novos usuários](#) 7 |
[Visite seu Grupo](#)

O amor não cai do céu. Comece sua procura agora mesmo!

Y! Ofertas. Conheça e tenha descontos de 50% em suas compras.

Todas as ofertas dos clubes de compras em um só lugar. Clique aqui.



Trocar para: [SÃ Text](#)o, [Resenha DiÃria](#) • [Sair do grupo](#) • [Termos de uso](#)

Fabrcio Cabral <fabriciofx@gmail.com>
To: UML-BR@yahoogrupos.com.br

Tue, Oct 11, 2011 at 2:34 PM

OlÃ MÃrcio,

eu acho que a minha explicaÃÃo nÃo foi muito boa. Permita-me explicar melhor e contextualizar o problema:

Suponha que eu tenha trÃs classes: Pessoa, PessoaFisica e PessoaJuridica. Pessoa Ã uma classe abstrata e as classes PessoaFisica e PessoaJuridica sÃo classes concretas e herdam de Pessoa. As classes PessoaFisica e PessoaJuridica possuem suas prÃprias propriedades, alÃm daquelas obviamente herdadas pela classe Pessoa. A classe Pessoa, por ser uma entidade, possui uma propriedade chamada id, que identifica cada objeto de maneira Ãnica. As classes PessoaFisica e PessoaJuridica, por herdarem da classe Pessoa, e por tambÃm serem entidades, tambÃm possuem o mesmo id.

Com o objetivo de realizar a persistÃncia destes objetos *sem* usar um ORM, vejo as seguintes possibilidades:

1. Criar uma classe RepositÃrio associado a classe Pessoa, onde esta poderÃ receber instÃncias da classe PessoaFisica ou PessoaJuridica. No entanto, a dÃvida Ã, como fazer da maneira "correta" a persistÃncia das propriedades exclusivas destas duas classes;
2. Criar uma classe RepositÃrio associado a cada classe concreta, no caso uma para classe PessoaFisica e outra pra PessoaJuridica.

Alguns sugeriram utilizar a abordagem (2) e outros tantos a (1), mas nÃo explicaram como eu faria para persistir a informaÃÃo exclusiva de cada classe, isto Ã, *nÃo* queria ter que fazer:

```
class RepositorioPessoa
{
    void add(Pessoa p)
    {
        if (p instanceof PessoaFisica)
        {
            // persista as propriedades X e Y
        } else if (p instanceof PessoaJuridica)
        {
            // persista as propriedades I e J
        }
    }
}
```

Assim, nÃo sei qual seria a soluÃÃo mais adequada a este problema. AgradeÃo qualquer sugestÃo para a melhor resoluÃÃo.

[]'s

2011/10/11 marciovmartins86 <marciovmartins@hotmail.com>

Isto será um padrão para todas as classes que herdam de A? Se for, é melhor fazer em associação com A e automaticamente serão passadas para B, C e qualquer outra classe nova que venha a surgir.

Agora, se for algo peculiar apenas a B ou C, é melhor fazer uma classe nova e apenas as classes que necessitam dessa funcionalidade herdem isto.

—
—fx

Marcelo Elias Del Valle <mvallebr@gmail.com>

Sun, Nov 6, 2011 at 8:04 PM

Reply-To: UML-BR@yahoogrupos.com.br

To: UML-BR@yahoogrupos.com.br

Fabício,

Você pode persistir (ou recuperar) as informações dessas classes de várias maneiras. Eu usaria uma implementação diferente para cada classe, pois o número de especializações pode subir e você provavelmente irá persistir de forma diferente as classes filhas da classe pai. Talvez a melhor alternativa fosse um repositório único para as classes filhas e outro para a classe pai.

Acho que é nisso o que eu pensaria para tomar essa decisão, em como vou persistir os objetos. Contudo, se você usar um único repositório para persistir todas as classes, o que pode acontecer é você ter de colocar IFs dentro dos métodos, no caso de a persistência de cada um vir a ser diferente.

Abraços,
Marcelo.

Em 11 de outubro de 2011 14:34, Fabício Cabral <fabriciofx@gmail.com> escreveu:

> **

[Quoted text hidden]

> 2011/10/11 marciovmartins86 <marciovmartins@hotmail.com>

>

> > **

>

> >

> >

> >

> > Isto será um padrão para todas as classes que herdam de A? Se for, é

> melhor

> > fazer em associação com A e automaticamente serão passadas para B, C e

> > qualquer outra classe nova que venha a surgir.

> >

> > Agora, se for algo peculiar apenas a B ou C, é melhor fazer uma classe

> > nova e apenas as classes que necessitam dessa funcionalidade herdem isto.

> >

> —

> —fx

>

> [As partes desta mensagem que não continham texto foram removidas]

>

>

>

>

>

>

>

>

Marcelo Elias Del Valle

<http://mvalle.com>

[As partes desta mensagem que não continham texto foram removidas]

Links do Yahoo! Grupos

<*> Para visitar o site do seu grupo na web, acesse:

<http://br.groups.yahoo.com/group/UML-BR/>

<*> Para sair deste grupo, envie um e-mail para:

UML-BR-unsubscribe@yahoogrupos.com.br

<*> O uso que você faz do Yahoo! Grupos está sujeito aos:

<http://br.yahoo.com/info/utos.html>

Vagner santos <vjsantosuniban@gmail.com>

Wed, Nov 9, 2011 at 2:11 PM

Reply-To: UML-BR@yahoogrupos.com.br

To: UML-BR@yahoogrupos.com.br

Boa tarde,

Gostaria de saber se é correto colocar em uma especificação logica um erro a ser corrigido da aplicação, ou o correto seria abrir um chamado para o mesmo!

Em 6 de novembro de 2011 21:04, Marcelo Elias Del Valle

<mvallebr@gmail.com> escreveu:

> Fabrício,

>

> Você pode persistir (ou recuperar) as informações dessas classes de
> várias maneiras. Eu usaria uma implementação diferente para cada classe,
> pois o número de especializações pode subir e você provavelmente irá
> persistir de forma diferente as classes filhas da classe pai. Talvez a
> melhor alternativa fosse um repositório único para as classes filhas e
> outro para a classe pai.

> Acho que é nisso o que eu pensaria para tomar essa decisão, em como vou
> persistir os objetos. Contudo, se você usar um único repositório para
> persistir todas as classes, o que pode acontecer é você ter de colocar IFs
> dentro dos métodos, no caso de a persistência de cada um vir a ser
> diferente.

>

> Abraços,

> Marcelo.

>

> Em 11 de outubro de 2011 14:34, Fabrício Cabral <fabriciofx@gmail.com>

[Quoted text hidden]

> > 2011/10/11 marciovmartins86 <marciovmartins@hotmail.com>

> >

> > > **

> >

> > >

> >

> > > Isto será um padrão para todas as classes que herdarem de A? Se for, é

> > melhor

> > > fazer em associação com A e automaticamente serão passadas para B, C e

> > > qualquer outra classe nova que venha a surgir.

> >

> > > Agora, se for algo peculiar apenas a B ou C, é melhor fazer uma classe

Vagner J Santos
Analista de Sistemas
Empresa Totvs
Cel : (11)6242-4357

[As partes desta mensagem que não continham texto foram removidas]

* 3 *

[| através de email](#) | [Responder através da web](#) | [Adicionar um novo tópico](#)
Mensagens neste tópico (5)

ATIVIDADE NOS ÚLTIMOS DIAS: **Novos usuários 1** | [Visite seu Grupo](#)

O que é melhor do que pagar pouco por algo que custa muito? Acesse Já.

Encontre a pessoal ideal no Y!Econtros. Comece sua procura agora mesmo!



Trocar para: [SÃ³ Texto](#) • [Resenha DiÃ¡ria](#) • [Sair do grupo](#) • [Termos de uso](#)

_____? _____* _____? _____

RUTEMBERG ALMEIDA <ruthemberg@bol.com.br>
Reply-To: UML-BR@yahoogrupos.com.br
To: UML-BR@yahoogrupos.com.br

Wed, Nov 9, 2011 at 9:15 PM

Depende, se o motivo do seu trabalho for prover melhorias no sistema, e entre ela a correção desse defeito

É importante lembrar que em ambos os casos a documentação deverá existir, sendo que no primeiro caso haverá uma especificação nova, e no outro, uma alteração na especificação do sistema (se houver).

RUTHEMBERG
cel 19 8809-3039
cel 11 8430-0065

[Quoted text hidden]

_____*, _____*, _____*

ATIVIDADE NOS ÚLTIMOS DIAS: [Novos usuários](#) 1 | [Visite seu Grupo](#)

Y!Encontros. O amor não vai cair do céu, então vá atrás dele!

Trocar para: [SÃ³ Texto](#), [Resenha DiÃ¡ria](#) • [Sair do grupo](#) • [Termos de uso](#)

_____? _____? _____?