



Fabrício Cabral <fabriciofx@gmail.com>

[dotnetarchitects] Dúvidas básicas OO e DDD

8 messages

@fmatheus07 <felipe_matheus@hotmail.com>

Wed, Dec 28, 2011 at 12:19 PM

Reply-To: dotnetarchitects@googlegroups.com

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

Boa tarde pessoal,

Sou iniciante em OO e DDD e andei lendo um pouco sobre o assunto, os padrões e talz. Porém ainda não consigo montar nada apenas com a teoria, parece que esta faltando uma linha para ligar tudo que venho lendo.

Vou enumerar algumas dúvidas minhas e se alguém puder ajudar fico agradecido :).

1) Quanto ao repositório, na arquitetura sugerida em DDD (Infra, Dominio, Aplicação, UI), coloco meus repositórios na camada de infra ou no Dominio? Caso infra, como eu resolvo a dependencia multua entre Dominio e Infra uma vez que a entidade esta no dominio e sua tarefas que necessitam de armazenar um estado estão na infra....

2) Programar para interface, utilizar o TEM-UM ao inves de É-UM me parece uma ótima solução, mas não consigo ver aplicação real para essa abordagem, sempre que eu tento montar um esquema de estrategia que não seja comportamento de patos me perco :/. Estou apanhando em um aplicação fake que estou construindo para estudar.

Imaginem uma classe produto que possui alguns atributos como (custo médio, custo atual, saldo em estoque, quantidade mínima, código, descrição, origem {comprado, fabricado}, estrutura caso fabricado, peso, etc. Acredito que na realidade esta classe será um super classe que agrega varias outras classes, podendo ser instanciada como ProdutoFabricado ou como ProdutoComprado, mas não sei como fazer isso usando as boas praticas, sempre peço em alguma coisa. Será que alguém poderia me ajudar com esse exemplo prático, pra ver se no tranco eu pego hehehe.

3) Vi muitas coisas sobre diferenças entre o Dominio e o Model do MVC, no entanto só consigo pensar em uma coisa: Retrabalho :/, sempre acabo praticamente reescrevendo a entidade que esta no domino no model.

4) Por fim, No EF4, se eu precisar incluir um campo novo em uma determinada tabela eu preciso atualizar o banco, o edmx, a entidade e depois aplicar a regra de negócio nova ?

Acho que eu estou fazendo isso da maneira errada hehehe..

[]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

Para mais opções visite o grupo em <http://groups.google.com/group/dotnetarchitects?hl=pt-br>

Denis Ferrari <denis.sisinf@gmail.com>

Wed, Dec 28, 2011 at 6:22 PM

Reply-To: dotnetarchitects@googlegroups.com

To: dotnetarchitects@googlegroups.com

Oi Felipe,

Seguem alguns links que vão te ajudar a entender a aplicação dos conceitos (eu acho):

Espero que ajude. :)

Abraços!

Denis Ferrari - "Faça pouco, faça sempre e faça direito"

www.heroisdati.com

Sites

www.denisferrari.com

www.minhacarreira.com

Em 28 de dezembro de 2011 12:19, @fmatheus07 <felipe_matheus@hotmail.com> escreveu:

Boa tarde pessoal,

Sou iniciante em OO e DDD e andei lendo um pouco sobre o assunto, os padrões e talz. Porém ainda não consigo montar nada apenas com a teoria, parece que esta faltando uma linha para ligar tudo que venho lendo.

Vou enumerar algumas dúvidas minhas e se alguém puder ajudar fico agradecido :).

1) Quanto ao repositório, na arquitetura sugerida em DDD (Infra, Dominio, Aplicação, UI), coloco meus repositórios na camada de infra ou no Dominio? Caso infra, como eu resolvo a dependencia multua entre Dominio e Infra uma vez que a entidade esta no dominio e sua tarefas que necessitam de armazenar um estado estão na infra....

<http://www.heroisdati.com/persistencia-plugavel/>

2) Programar para interface, utilizar o TEM-UM ao inves de É-UM me parece uma ótima solução, mas não consigo ver aplicação real para essa abordagem, sempre que eu tento montar um esquema de estrategia que não seja comportamento de patos me perco :/. Estou apanhando em um aplicação fake que estou construindo para estudar.

Imaginem uma classe produto que possui alguns atributos como (custo médio, custo atual, saldo em estoque, quantidade mínima, código, descrição, origem {comprado, fabricado}, estrutura caso fabricado, peso, etc. Acredito que na realidade esta classe será um super classe que agrega varias outras classes, podendo ser instanciada como ProdutoFabricado ou como ProdutoComprado, mas não sei como fazer isso usando as boas praticas, sempre peço em alguma coisa. Será que alguém poderia me ajudar com esse exemplo prático, pra ver se no tranco eu pego hehehe.

<http://www.heroisdati.com/herocast-aquele-das-tres-regras-da-poo/>

<http://www.heroisdati.com/herocast-aquele-sobre-injecao-de-dependencia-di/>

<http://www.heroisdati.com/herocast-abrir-pendencia-e-notificar-por-e-mail/>

3) Vi muitas coisas sobre diferenças entre o Dominio e o Model do MVC, no entanto só consigo pensar em uma coisa: Retrabalho :/, sempre acabo praticamente reescrevendo a entidade que esta no domino no model.

<http://www.heroisdati.com/herocast-07-domain-model-x-view-model-e-automapper/>

4) Por fim, No EF4, se eu precisar incluir um campo novo em uma determinada tabela eu preciso atualizar o banco, o edmx, a entidade e depois aplicar a regra de negócio nova ?

Acho que eu estou fazendo isso da maneira errada hehehe..

[]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
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>

Fabrizio Cabral <fabriciofx@gmail.com>
To: dotnetarchitects@googlegroups.com

Wed, Dec 28, 2011 at 7:21 PM

Felipe,

vou procurar tentar responder pelo que eu sei, ok (se eu disser alguma besteira, o pessoal pode ficar à vontade em me corrigir!
:D)

2011/12/28 @fmatheus07 <felipe_matheus@hotmail.com>

Boa tarde pessoal,

Sou iniciante em OO e DDD e andei lendo um pouco sobre o assunto, os padrões e talz. Porém ainda não consigo montar nada apenas com a teoria, parece que esta faltando uma linha para ligar tudo que venho lendo.

Vou enumerar algumas dúvidas minhas e se alguém puder ajudar fico agradecido :).

1) Quanto ao repositório, na arquitetura sugerida em DDD (Infra, Dominio, Aplicação, UI), coloco meus repositórios na camada de infra ou no Dominio? Caso infra, como eu resolvo a dependencia multua entre Dominio e Infra uma vez que a entidade esta no dominio e sua tarefas que necessitam de armazenar um estado estão na infra....

Meus repositórios são uma interface de alguma implementação. Por exemplo, eu crio a interface para um RepositorioProdutos, mas posso ter uma implementação para guardar os dados em um BD (RepositorioProdutosBD) ou guardar em um arquivo XML (RepositorioProdutosXML). Estas implementações devem implementar a interface RepositorioProdutos.

Então, faço assim: a interface (RepositorioProdutos) faz parte da camada d domínio, mas as implementações (RepositorioProdutosBD ou RepositorioProdutosXML) faz parte da minha camada de infraestrutura).

2) Programar para interface, utilizar o TEM-UM ao inves de É-UM me parece uma ótima solução, mas não consigo ver aplicação real para essa abordagem, sempre que eu tento montar um esquema de estrategia que não seja comportamento de patos me perco :/. Estou apanhando em um aplicação fake que estou construindo para estudar.

Acredito que o que está por trás deste pensamento é o chamado "acoplamento". Quando você utiliza herança (É-UM) você aumenta o acoplamento entre duas classes, enquanto que uma agregação ou composição (TEM-UM) o acoplamento é bem menor).

Imagem uma classe produto que possui alguns atributos como (custo médio, custo atual, saldo em estoque, quantidade mínima, código, descrição, origem {comprado, fabricado}, estrutura caso fabricado, peso, etc. Acredito que na realidade esta classe será um super classe

que agrega varias outras classes, podendo ser instanciada como ProdutoFabricado ou como ProdutoComprado, mas não sei como fazer isso usando as boas praticas, sempre peço em alguma coisa. Será que alguém poderia me ajudar com esse exemplo prático, pra ver se no tranco eu pego hehehe.

Eu não sei se entendi muito bem o seu exemplo, mas acho que ele pareceu meio vago. A única coisa que difere um ProdutoFabricado de um ProdutoComprado é o atributo origem? O que é esse atributo? Uma outra classe? Apenas um valor simples? ProdutoFabricado e ProdutoComprado possui comportamentos diferentes? A lógica do negócio é diferente entre ambas?

3) Vi muitas coisas sobre diferenças entre o Dominio e o Model do MVC, no entanto só consigo pensar em uma coisa: Retrabalho :/, sempre acabo praticamente reescrevendo a entidade que esta no domino no model.

Por que isso? Não dá pra reaproveitar a mesma classe?

4) Por fim, No EF4, se eu precisar incluir um campo novo em uma determinada tabela eu preciso atualizar o banco, o edmx, a entidade e depois aplicar a regra de negócio nova ?
Acho que eu estou fazendo isso da maneira errada hehehe..

Aqui eu peço ajuda aos universitários... :)

--
--fx

Felipe Matheus <felipe_matheus@hotmail.com>
Reply-To: dotnetarchitects@googlegroups.com
To: dotnetarchitects@googlegroups.com

Wed, Dec 28, 2011 at 11:55 PM

Fabricio e Denis,

Muito obrigado pela atenção (:

Quanto a questão 1, fica bem claro qto falta eu ganhar experiência em OO, a solução estava logo ali nos princípios básicos hehehe.

Questão 2, realmente é para eu tentar aplicar testes unitários. Já que vou fazer uma aplicação para estudos vou tentar englobar tudo o que for possível de tecnologias e técnicas.

No meu exemplo, ProdutoComprado é totalmente diferente de ProdutoFabricado.

Um produto comprado, possui um fornecedor, um grupo de estoque, saldo, custo, etc.

Já um produto fabricado tem matéria prima, seu custo é composto pelo custo de suas matérias, tenho que considerar como ele será fabricado (estrutura de produto: Produto A é fabricado com 1 produto B e 2 produtos C).

Questão 3, Eu vou dar uma lida no site do Denis que parece ter um conteúdo bem legal, se ainda continuar com as dúvidas eu volto a perturbar vocês.

[]s e Boas festas a todos!

De: dotnetarchitects@googlegroups.com [mailto:dotnetarchitects@googlegroups.com] **Em nome de** Fabrício Cabral

Enviada em: quarta-feira, 28 de dezembro de 2011 20:22

Para: dotnetarchitects@googlegroups.com

Assunto: Re: [dotnetarchitects] Dúvidas básicas OO e DDD

[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>

Fabio Kobuti <kobuti@gmail.com>

Thu, Dec 29, 2011 at 9:34 AM

Reply-To: dotnetarchitects@googlegroups.com

To: dotnetarchitects@googlegroups.com

Felipe,

Quanto a questão de aprender OO, tem um livro beeeeeem bacana que elucida bem a questão. É o Head First Object-Oriented Analysis and Design.

Ele apresenta de maneira fácil e bem divertida de entender. Na amazon tá beem em conta o livro msm.

[Quoted text hidden]

—

=====

Fabio K Cavalcante

11 6317-0983

[Quoted text hidden]

Fabrizio Cabral <fabriciofx@gmail.com>

Thu, Dec 29, 2011 at 1:02 PM

To: dotnetarchitects@googlegroups.com

Olá Felipe,

2011/12/28 Felipe Matheus <felipe_matheus@hotmail.com>

Fabricio e Denis,

Muito obrigado pela atenção (:

De nada. :)

Quanto a questão 1, fica bem claro qto falta eu ganhar experiência em OO, a solução estava logo ali nos princípios básicos hehehe.

Sinceramente, não acho que essa sua 1ª pergunta tenha sido trivial. Muito pelo contrário: pelo que eu já andei vendo na net ela já gerou *muitas* discussões nos fóruns e nas listas. Inclusive, era uma dúvida pertinente que eu também tinha.

Questão 2, realmente é para eu tentar aplicar testes unitários. Já que vou fazer uma aplicação para estudos vou tentar englobar tudo o que for possível de tecnologias e técnicas.

No meu exemplo, ProdutoComprado é totalmente diferente de ProdutoFabricado.

Um produto comprado, possui um fornecedor, um grupo de estoque, saldo, custo, etc.

Já um produto fabricado tem matéria prima, seu custo é composto pelo custo de suas matérias, tenho que considerar como ele será fabricado (estrutura de produto: Produto A é fabricado com 1 produto B e 2 produtos C).

E o que existe de comum entre ProdutoComprado e ProdutoFabricado? Veja que para poder melhor modelar OO, é preciso conhecer melhor o domínio do seu problema. Dois objetos iguais, mas em domínios diferentes, podem necessitar de implementações diferentes. Por isso é MUITO importante analisar o contexto do seu problema para definir qual a melhor solução.

No seu caso, eu *suspeito* que ProdutoComprado e ProdutoFabricado são dois objetos diferentes, mas que, eventualmente, podem apresentar um mesmo comportamento. Se for isso, uma sugestão seria criar uma interface (por exemplo, Produto) e mandar as classes ProdutoComprado e ProdutoFabricado implementar esta interface.

Questão 3, Eu vou dar uma lida no site do Denis que parece ter um conteúdo bem legal, se ainda continuar com as dúvidas eu volto a perturbar vocês.

Ler ajuda muito. Mas eu recomendaria você começar com "a base", como por exemplo, o livro DDD do Eric Evans, os livros da série Head First e os livros do Martin Fowler.

[]'s

—

—fx

Antonio Pedro <aplнас@hotmail.com>

Thu, Dec 29, 2011 at 1:10 PM

Reply-To: dotnetarchitects@googlegroups.com

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

@Fabrício

DDD não é a base, para alguém conseguir entender o livro do Evans, ela deve ter uma boa base de OO e padrões de projeto.

Recomendo para quem está começando ler o livro Utilizando Padrões e UML do Larman:
(http://www.amazon.com/Applying-UML-Patterns-Introduction-Object-Oriented/dp/0131489062/ref=ntt_at_ep_dpt_1). Estude principalmente sobre padrões GRASP.

Att,

Antônio Nascimento

Date: Thu, 29 Dec 2011 13:02:21 -0300
Subject: Re: [dotnetarchitects] Dúvidas básicas OO e DDD
From: fabriciofx@gmail.com
To: dotnetarchitects@googlegroups.com
[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]

Robson Ramos <robsonramosabiao@gmail.com>
Reply-To: dotnetarchitects@googlegroups.com
To: ".Net Architects" <dotnetarchitects@googlegroups.com>

Thu, Dec 29, 2011 at 1:20 PM

"Head First: Object-Oriented Analysis and Design." +1

Att,

Robson Ramos

On 29 dez, 14:10, Antonio Pedro <apl...@hotmail.com> wrote:
> @Fabrício
> DDD não é a base, para alguém conseguir entender o livro do Evans, ela deve ter uma boa base de OO e padrões de projeto.
>
> Recomendo para quem está começando ler o livro Utilizando Padrões e UML do Larman:
> (http://www.amazon.com/Applying-UML-Patterns-Introduction-Object-Oriented/dp/0131489062/ref=ntt_at_ep_dpt_1). Estude principalmente sobre padrões GRASP.
> Att,
> Antônio Nascimento
>
> Date: Thu, 29 Dec 2011 13:02:21 -0300
> Subject: Re: [dotnetarchitects] Dúvidas básicas OO e DDD
> From: fabriciofx@gmail.com
> To: dotnetarchitects@googlegroups.com
>
> Olá Felipe,
>
> 2011/12/28 Felipe Matheus <felipe_math...@hotmail.com>
>
> Fabrício e Denis,
> Muito obrigado pela atenção (:)
>
> De nada. :)
> Quanto a questão 1, fica bem claro qto falta eu ganhar experiência em OO, a solução estava logo ali nos princípios básicos hehehe.
>
> Sinceramente, não acho que essa sua 1ª pergunta tenha sido trivial. Muito pelo contrário: pelo que eu já

andei vendo na netela já gerou *muitas* discussões nos fóruns e nas listas.

> Inclusive, era uma dúvida pertinente que eu também tinha.

>

> Questão 2, realmente é para eu tentar aplicar testes unitários. Já que vou fazer uma aplicação para estudos vou tentar englobar tudo o que for possível de tecnologias e técnicas.

> No meu exemplo, ProdutoComprado é totalmente diferente de ProdutoFabricado.

> Um produto comprado, possui um fornecedor, um grupo de estoque, saldo, custo, etc.

> Já um produto fabricado tem matéria prima, seu custo é composto pelo custo de suas matérias, tenho que considerar como ele será fabricado (estrutura de produto: Produto A é fabricado com 1 produto B e 2 produtos C).

>

> E o que existe de comum entre ProdutoComprado e ProdutoFabricado?Veja que para poder melhor modelar OO, é preciso conhecer melhor domínio do seu problema. Dois objetos iguais, mas em domínios

> diferentes, podem necessitar de implementações diferentes. Porisso é MUITO importante analisar o contexto do seu problema para definir qual a melhor solução.

> No seu caso, eu *suspeito* que ProdutoComprado e ProdutoFabricado

> são dois objetos diferentes, mas que, eventualmente, podem apresentarum mesmo comportamento. Se for isso, uma sugestão seria criaruma interface (por exemplo, Produto) e mandar as classes ProdutoComprado

> e ProdutoFabricado implementar esta interface.

>

> Questão 3, Eu vou dar uma lida no site do Denis que parece ter um conteúdo bem legal, se ainda continuar com as dúvidas eu volto a perturbar vocês.

>

> Ler ajuda muito. Mas eu recomendaria você começar com "a base", como por exemplo, o livro DDD do Eric Evans, os livros da série Head First eos livros do Martin Fowler.

>

> []'s

> --

> --fx

>

> --

>

> 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

[Quoted text hidden]