Tópicos - Seminário SOA

Arquitetura Orientada Serviço

Pra dia 27/09

Conceitualizar SOA ok

Falar de integração vs SOA (esta no livro pag.187)

Padrões de projeto SOA

Exemplo de um Projeto SOA

Beneficios ok

<https://www.infoq.com/br/news/2009/01/SOAPatternsBook>

<http://www.devmedia.com.br/vantagens-e-desvantagens-de-soa/27437>

<http://www.celtainformatica.com.br/noticias/o-que-e-soa-e-por-que-usa-la>

O que é SOA?

SOA (Servico Oriented Architecture) não é um serviço. Muitos acreditam que seus sistemas são baseados em SOA somente pelo fato de possuírem web services. SOA é, antes de tudo, um conceito.   
É mais fácil definir o que não é SOA, SOA não é produto, não é solução, não é uma tecnologia. SOA é um estilo de projeto, ele guia a criação e uso de serviços durante o ciclo de desenvolvimento.

Afinal, por que usar SOA?

As organizações precisam responder de forma rápida às oportunidades de mercado, dentro das empresas existem diferentes aplicações para serem realizadas suas atividades, estas aplicações precisam se comunicar com o objetivo de simplificar o negócio tornando-os mais produtivos e ajudando na competitividade do mercado.   
O grande uso de serviços levou a grandes avanços nas integrações entre aplicações. E é aí que entra SOA, pois ela é mais flexível e capaz de suportar serviços independente da plataforma e protocolo pelos quais estão sendo distribuídos.   
SOA tem como propósito tratar requisitos com baixo acoplamento, desenvolvimento realizado em padrões, mapear sistemas de informação para todos seus fluxos de processos, gerenciar transações, tratar da existência de sistemas com múltiplas plataformas e sistemas legados.   
O objetivo de SOA é permitir com que as organizações realizem seus negócios e tenham vantagens por meio da combinação de inovação e governança eficaz.

Vantagens e desvantagens

Vantagens

Uma das principais vantagens de se ter esse tipo de arquitetura é a reutilização de serviços o que aumenta e produtividade, alinha melhor o negócio e facilita a gerencia.   
Outras vantagens:

* Reutilização: O serviço pode ser reutilizado para outras aplicações.
* Produtividade: Com o reuso, a equipe de desenvolvimento pode reutilizar serviços em outros projetos, diminuindo o tempo de desenvolvimento.
* Flexibilidade: Isolando a estrutura de um serviço as mudanças são feitas com maior facilidade.
* Manutenibilidade: Com baixo acoplamento, facilita a manutenção dos serviços.
* Alinhamento com o negócio: A área de negócio visualiza os processos alinhados com a tecnologia.
* Interoperabilidade: Disponibilizar serviços independentemente da plataforma e tecnologia.
* Integração: A integração com outros serviços, aplicativos e sistemas legados.
* Governança: Gerenciamento nos processamentos de negócio.
* Padronizado: É baseado no uso de padrões.
* Abstração: Serviço totalmente abstraído da sua implementação.

Desvantagens

Geralmente, a maior preocupação na implementação dessa arquitetura é a segurança.   
Outras desvantagens:

* Complexidade: Uma grande quantidade de serviços precisa ser gerenciada.
* Performance: A performance depende do servidor onde o serviço está publicado, como também da rede.
* Robustez: Caso uma exceção acontecer não tem como reverter o processo.
* Disponibilidade: Uma queda na rede ou no servidor deixa todos os serviços indisponíveis.
* Testabilidade: O debug no serviço é um problema para os desenvolvedores.
* Segurança: Os serviços estão disponíveis na rede, qualquer aplicativo pode consumir esse serviço, os dados são trafegados pela rede podendo ser interceptados.

Integração vs. SOA

No catalogo de design patterns SOA, muitos dos padrões nele contidos não tem-se um detalhamento de como se implementar, mas sim uma forma de modelagem. Isso, tem muito, pelo motivo que SOA não diz respeito a detalhes de como implementar. E é aí que muitos dos detalhes de como implementar SOA se confunde com técnicas de integração de sistemas, porque faz com que SOA tenha que usar técnicas de integração para ser melhor entendida. Apesar disso, muitas técnicas aplicáveis não são aplicáveis em SOA, isso faz que elas compartilhem algumas características sem serem equivalentes.