

Anotação do Manifesto SOA

Comentários e Análises sobre o Manifesto SOA de Thomas Erl

A orientação a serviços é um paradigma que molda o que você faz. A arquitetura orientada a serviços (SOA) é um tipo de arquitetura resultante da aplicação do paradigma de orientação a serviços.

Desde o início, foi entendido que o manifesto era sobre dois temas distintos, porém relacionados: o molde arquitetural orientado a serviços e a orientação a serviços propriamente, paradigmas no qual uma arquitetura é definida. O formato deste manifesto foi modelado a partir do Manifesto Ágil, o que limita o conteúdo de declarações concisas que expressam as ambições, valores e princípios orientadores para a realização dessas. Esse manifesto não é uma especificação de um modelo de referência ou até mesmo um papel em branco sem opção de estabelecer as definições reais, nós decidimos adicionar este preâmbulo, a fim de esclarecer como e por que esses termos são referenciados em outras partes do documento de manifesto.

Estamos aplicando a orientação a serviços...

O paradigma de orientação a serviços é melhor entendido como um método ou uma abordagem para a realização de um estado-alvo, logo definido por um conjunto de benefícios e objetivos estratégicos. Quando aplicamos a orientação a serviços, construímos softwares e arquitetura tecnológicas para apoiar à realização deste estado-alvo. Assim é como se qualifica uma arquitetura de tecnologia como sendo orientada a serviços.

... para ajudar as organizações a entregar valor de negócio sustentável e consistente, com maior agilidade e rentabilidade...

Esta continuação do preâmbulo destaca alguns dos benefícios mais comuns e importantes que se espera da estratégia da computação orientada a serviços. A compreensão desses benefícios ajuda a lançar alguma luz sobre o supracitado estado-alvo que pretendemos perceber como um resultado da aplicação da orientação a serviços.

Agilidade no nível de negócios é comparável à capacidade de resposta de uma organização. Quanto mais fácil e eficaz uma organização consegue responder às mudanças do negócio, mais eficiente e bem sucedida será a sua adaptação nos impactos da mudança (e alavancará ainda mais os benefícios que uma mudança pode trazer).

A orientação a serviços posiciona serviços como bens ativos de TI e espera-se que estes forneçam "valor repetido" ao longo do tempo, excedendo e sobrepondo o

investimento inicial necessário para a sua entrega. O custo-efetividade está ligado primeiramente ao retorno de investimento. De muitas maneiras, o aumento de custo-efetividade caminha lado a lado com o aumento da agilidade, se houver mais oportunidades de reutilização de serviços existentes, então o custo com novas soluções será menor.

O valor de negócio "sustentável" refere-se aos objetivos de longo prazo da orientação a serviços para estabelecer programas de software como serviços, com flexibilidade inerente, para continuamente se compor novas configurações de solução e evoluir para acomodar as necessidades empresariais, que sempre estão em constante mudança.

... Em consonância com as necessidades empresariais em constantes mudanças.

Estas últimas seis palavras do preâmbulo são fundamentais para compreender a filosofia subjacente à computação orientada a serviços. A necessidade de acomodar a mudança nos negócios em uma base contínua é fundamental para o serviço de orientação e considerado um objetivo geral estratégico fundamental.

Através do nosso trabalho, viemos para dar prioridade:

As próximas declarações estabelecem um conjunto de valores, cada qual é expresso como uma prioridade sobre algo que também é considerado de valor. A intenção deste sistema de valores é abordar as escolhas difíceis que precisam ser feitas em uma base regular para que os objetivos estratégicos e benefícios da computação orientada a serviços sejam realizados de forma consistente.

Valor de Negócio à estratégia técnica

Como dito anteriormente, a necessidade de acomodar a mudança nos negócios é um objetivo estratégico global. Portanto, a qualidade fundamental da arquitetura orientada a serviços e de quaisquer programas de software, soluções e eco-sistema que resultam da adoção de orientação a serviços é que eles são orientados pelos negócios. Não se trata de tecnologia de determinar a direção da empresa, trata-se da visão de negócios ditando a utilização da tecnologia.

Essa prioridade pode ter um profundo efeito cascata nas regiões de uma empresa de TI. Introduzindo mudanças em quase todas as partes do ciclo de vida das entregas da TI, em como podemos planejar e fundar soluções de automação, a forma como construímos e governamos-as. Todos os outros valores e princípios do manifesto, de um modo ou de outro, apóiam a realização deste valor.

Objetivos estratégicos à benefícios específicos de um projeto só

Historicamente, muitos projetos de TI focavam exclusivamente na criação de aplicativos projetados especificamente para automatizar o processo de requisitos de negócio que eram específicos de sua época. Isto cumpria necessidades imediatas (táticas), mas como outras aplicações de propósito singular eram criadas, isso resultou em uma empresa de TI cheia de ilhas de lógica de negócios e dados referidos como aplicação de "silos". Como novos requisitos de negócio surgiam, novos silos eram criados ou estabeleciam novos canais de integração entre esses

silos. Como a mudança em negócios surgia, os canais de integração tinha que ser aumentados, novos silos tinham que ser criados, e logo o cenário corporativo de TI tornou-se cada vez mais complicado e oneroso, caro e lento para evoluir.

De muitas maneiras, a orientação a serviços surgiu em resposta a estes problemas. é um paradigma que fornece uma alternativa para projetos específicos de desenvolvimento de aplicações, baseada em silos e desenvolvimento de aplicações de integração, no intuito de, inflexivelmente, priorizar a realização de longo prazo e os objetivos estratégicos de negócios. O estado de destino defendido na orientação a serviço não tem silos de aplicações tradicionais. E mesmo quando os recursos existentes e silos de aplicações existam em ambientes onde a orientação de serviços for aprovada, o estado de destino é aquele em que eles estão harmonizados a qualquer medida do possível.

Interoperabilidade Intrínseca à integração personalizada

Para os programas de software compartilhar dados de dados é preciso que eles sejam interoperáveis. Se os programas de software não são projetados para serem compatíveis, eles provavelmente não serão interoperáveis. Para permitir a interoperabilidade entre programas de software incompatíveis exige-se que eles sejam integrados. A integração é, portanto, o esforço necessário para alcançar a interoperabilidade entre softwares diferentes. Embora muitas vezes necessário, a integração personalizada pode ser cara e demorada e pode levar a construção de arquiteturas frágeis, de difícil evolução. Um dos objetivos da orientação a serviços é minimizar a necessidade de integração personalizada por meio da formulação de programas de software (dentro de um determinado domínio) para que eles sejam nativamente compatível. Esta é uma qualidade que se refere à interoperabilidade intrínseca. O paradigma de orientação a serviço engloba um conjunto de princípios de design específicos que são voltados a estabelecer a interoperabilidade intrínseca a vários níveis. Interoperabilidade intrínseca, como uma característica dos programas de software que reside dentro de um determinado domínio, é fundamental para perceber os benefícios estratégicos, tais como o aumento da relação custo-eficácia e agilidade.

Serviços compartilhados à Implementações de propósito específico

Como já explicado, a orientação a serviços estabelece uma abordagem de projeto composto por um conjunto de princípios de projeto (design patterns). Quando aplicado a uma foco significativo, estes princípios compõe um programa de software em uma unidade de lógica orientada a serviços que podem ser legitimamente chamado de serviço.

Os serviços são equipados com características concretas (como os que permitem a interoperabilidade intrínseca) que apoiam diretamente o estado desejado descrito anteriormente. Uma dessas características é o encapsulamento da lógica de multi-propósito que podem ser compartilhadas e reutilizadas para apoiar a automação de processos de negócios diferentes.

Um serviço compartilhado se estabelece como um ativo de TI que pode repetir valor de negócio, enquanto diminui o gasto e o esforço para oferecer soluções de automação. Embora haja valor em aplicativos tradicionais de propósito singular que resolve os requisitos táticos de negócios, o uso de serviços compartilhados

proporciona maior valor na concretização dos objetivos estratégicos da computação orientada a serviços (que também incluem um aumento da relação custo-eficácia e agilidade).

Flexibilidade a otimização

Esta é talvez a mais ampla das demonstrações priorização de valor, e é melhor visualizada como uma orientação filosófica para a melhor forma de priorizar as várias considerações quando se entrega e evoluem serviços individuais e inventários de serviços.

A otimização refere-se primeiramente ao cumprimento dos ganhos táticos ajustando um projeto ou agilizando sua entrega para atender necessidades imediatas. Não há nada de indesejável nisso, exceto que isso leva aos tão falados ambientes baseados em silos, quando não devidamente priorizados em relação à promoção da flexibilidade.

Por exemplo, a característica de flexibilidade vai além da capacidade dos serviços para efetivamente (e intrinsecamente) compartilhar dados. Para ser realmente sensível às necessidades empresariais em constante mudança, os serviços devem ser flexíveis e podem ser combinados e agregados em soluções compostas.

Ao contrário das tradicionais aplicações distribuídas, que muitas vezes eram relativamente estáticas, apesar do fato de serem modular, composições de serviços precisam ser concebidas com um nível de flexibilidade inerente para permitir aumento constante.

Isso significa que, quando um existente uma mudança no processo de negócio ou quando um novo processo é introduzido, é preciso ser capaz de adicionar, remover e estender os serviços dentro da arquitetura de composição com o esforço (integração) mínima. é por isso que agregabilidade de serviços é um dos principais princípios de projetos de orientação a serviços.

Refinamento evolucionário a busca da perfeição inicial

Há um ponto comum que confunde quando tratamos a "agilidade" em relação à orientação a serviços. Algumas abordagens defendem ganhos imediatos na entrega rápida de programas de software. Isto pode ser considerado "agilidade tática", pois o foco está no tático, de benefício de curto-prazo. A orientação a serviços defende a realização de agilidade em nível organizacional ou empresarial com a intenção de fortalecer a organização como um todo, para ser receptiva a mudanças. Esta forma de agilidade organizacional também pode ser referida como "agilidade estratégica", pois a ênfase está na longevidade de que cada programa de software que oferecemos trabalhe em direção a um estado desejado que promova agilidade no valor estratégico de longo prazo. Para uma empresa de TI permitir a agilidade organizacional, deve-se evoluir em paralelo com o negócio. Nós geralmente não podemos prever como um negócio evoluirá com o tempo e, portanto, não podemos iniciar a criação de serviços perfeitos. Ao mesmo tempo, geralmente há riqueza de conhecimento existente dentro da inteligência de uma organização em negócios existentes que podem ser colhidas durante as fases de análise e modelagem de projetos SOA. Esta informação, juntamente com os princípios de orientação a serviço e metodologias comprovadas, pode nos ajudar a identificar e definir um conjunto de serviços para um nível de operação de um negócio existente hoje

enquanto sendo suficientemente flexível para se adaptar as mudanças de negócio ao longo do tempo.

Ou seja, enquanto nós valorizamos os itens à direita, valorizamos os itens à esquerda ainda mais.

Ao estudar como esses valores são priorizados, ganhamos a introspecção no que distingue a orientação a serviços dos outros paradigmas. Este tipo de percepção pode beneficiar profissionais de TI de diversas maneiras. Por exemplo, pode nos ajudar a estabelecer critérios fundamentais para determinar o quão compatível a orientação a serviços é para uma determinada organização ou empresa de TI. Além disso, pode ajudar a determinar a medida no qual a orientação a serviços pode ou deveria ser adotada.

A apreciação dos valores centrais também pode nos ajudar a compreender os desafios que podem existir para realizar os projetos de SOA com sucesso dentro de certos ambientes. Por exemplo, várias dessas priorizações podem colidir frontalmente com crenças e preferências já estabelecidas. Nesse caso, os benefícios da orientação a serviços precisam ser pesados contra o esforço e o impacto que sua adoção pode ter (e não apenas na tecnologia, mas também na cultura da TI da empresa). Os próximos princípios e guias que fornecemos ajudam a resolver muitos destes tipos de desafios.

Nós seguimos estes princípios:

Até agora, o manifesto estabeleceu uma visão geral, bem como um conjunto de valores associados a cada visão. O restante da declaração é composta por um conjunto de princípios que são fornecidos como orientação para a adesão aos valores e concretização desta visão.

é importante ter em mente que estes são os princípios orientadores específicos para este manifesto. Separadamente, existe um conjunto de princípios de projetos (design principles) que compõem o paradigma de projetos orientados a serviços e há muitos mais práticas documentadas e padrões específicos para o orientação a serviços e arquitetura orientada a serviços.

Respeitar a estrutura social e poder da organização.

Uma das armadilhas mais comuns de projetos SOA está próxima à adoção de uma iniciativa tecnológica centrada. Fazer isso quase sempre leva ao fracasso, porque nós simplesmente não estamos preparados para os inevitáveis impactos organizacionais.

A adoção da orientação a serviços tem a ver com a transformação da forma de automação das empresas. No entanto, independentemente de que tenhamos planos para fazer este esforço para a transformação acontecer, devemos sempre começar com compreensão e apreciação da organização, sua estrutura, seus objetivos e sua cultura. A adoção da orientação a serviços é muito mais uma experiência humana. Ela requer o apoio dos dirigentes e, em seguida, pede uma cultura adaptada a uma atitude estratégica, centrada na comunidade. Temos de reconhecer plenamente e nos planejar para este nível de mudança organizacional, a fim de obter os compromissos necessários a longo prazo, necessários para

alcançar o estado alvo da orientação a serviços. Estes tipos de considerações, não só nos ajudam a determinar a melhor forma de avançar como uma iniciativa de SOA, mas também nos ajudam a definir a adoção mais adequada ao nosso contexto.

Reconhecer que SOA, definitivamente exige mudanças em muitos níveis.

Há um ditado que diz: " Success is being prepared for opportunity." Talvez a principal lição aprendida a partir de projetos SOA realizados até agora é que temos de compreendê-los de forma plena e, em seguida, planejar e preparar-se os níveis e volumes de mudanças trazidas com o resultado da adoção da orientação a serviços. Aqui estão alguns exemplos.

A orientação a serviço muda a forma como construímos soluções de automação de programas de software posicionando-os como ativos de longo prazo na TI com repetíveis valores comerciais. Um investimento inicial é necessário para se criar um ambiente formado por tais ativos e um compromisso constante é necessário para manter e alavancar o seu valor. Assim, desde o principio, são necessárias mudanças de como financiamos, medimos e mantemos sistemas de TI dentro da empresa. Além disso, como orientação a serviços apresenta os serviços posicionados como recursos da empresa, haverá mudanças sobre como nós nos apropriamos das diferentes partes dos sistemas e como regulamos sua concepção e utilização, sem mencionar as mudanças na infra-estrutura necessárias para garantir a contínua escalabilidade e confiabilidade.

O âmbito da adoção de SOA pode variar. Manter os esforços gerenciáveis dentro dos limites significativos.

Um mito comum é de que, a fim de concretizar os objetivos estratégicos da computação orientada a serviços, a orientação a serviços deve ser adotada na empresa inteira. Isto significa, criação e implementação de padrões de projetos (design patterns) e da indústria em toda a empresa de TI de forma a criar um inventário de serviços intrinsecamente interoperáveis de toda a empresa. Embora não haja nada de errado com esse ideal, não é uma meta realista para muitas organizações, especialmente aqueles com grandes equipes de TI.

O âmbito de aplicação mais adequada para qualquer esforço, dado qualquer tipo de adoção de SOA deve ser determinada como resultado de planejamento e análise em conjunto com considerações pragmáticas, como os tão falados impactos nas estruturas organizacionais, áreas de autoridade, e nas mudanças culturais que são trazidas. Estes tipos de fatores nos ajudam a determinar um foco de atuação que é mensurável. Mas para qualquer adoção que resulta num ambiente que eleva a TI em direção ao desejado estado alvo, o escopo tem que ser significativo.

Em outras palavras, deve ser significativamente um cruzamento de silos, para que as coleções de serviços possam ser entregues considerando umas as outras dentro de um limite pré-definido.

Em outras palavras, queremos criar "continentes de serviços" e não temida "ilhas de serviços". Este conceito de construção de inventários de serviços independentemente apropriados e governados dentro de domínios numa mesma TI corporativa, reduz muitos os riscos que são comumente atribuídos aos "big-bang"

de projetos SOA, além disso, diminui o impacto de ambas as mudanças organizacionais e tecnológicas (pois o impacto está limitado a um escopo segmentado e de menor administração). É também uma abordagem que permite a adoção por fases onde inventários de serviços dentro de domínio podem ser estabelecidos a cada momento.

Os produtos e as normas por si só, não vão prover SOA, nem aplicar o paradigma de orientação a serviço para você.

Este princípio aborda dois mitos distintos, porém relacionados. A primeira é que você pode comprar o seu caminho para SOA com produtos de tecnologia moderna, e a segunda é a suposição de que a adoção de padrões da indústria (como XML, WSDL, SCA, etc) naturalmente resultará em arquitetura de tecnologia orientada a serviços.

Empresas privadas e comunidades de padrões de mercado têm crédito com suas tecnologias inovadoras para criação de serviços mediante aos frameworks e plataformas não-proprietárias. Tudo a partir da virtualização de serviços para computação nas nuvens e computação em grid tem ajudado a promover o potencial para a construção de sofisticadas e complexas soluções orientadas a serviços. No entanto, nenhuma dessas tecnologias é exclusiva para SOA.

Você pode facilmente construir sistemas baseada em silos "nas nuvens" como também ter seus próprios servidores privados. Não existe tal coisa como "SOA in a box ", pois para atingir a arquitetura orientada a serviços, a orientação a serviços precisa ser aplicada com sucesso, o que, por sua vez, exige que tudo o que projetamos e construímos seja conduzido por uma única direção, visão, e requisitos do negócio.

SOA pode ser realizado através de uma variedade de tecnologias e padrões.

A orientação a serviços é um paradigma neutro em relação à tecnologia e fornecedores. Arquitetura orientada a serviços é um modelo arquitetônico neutro em relação a tecnologia e fornecedor. A Computação orientada a serviço pode ser vista como uma forma especializada de computação distribuída. Soluções orientadas a serviços podem ser criadas usando apenas as tecnologias e padrões de mercado voltados à computação distribuída.

Enquanto algumas tecnologias (especialmente aquelas baseadas em padrões de mercado) podem aumentar o potencial de utilização de alguns princípios de projetos de orientação a serviços, o que conta é o potencial de cumprimento dos requisitos de negócio, isso é o que vai determinar a escolha mais adequada de tecnologias e padrões da indústria.

Estabelecer um conjunto uniforme de normas e políticas empresariais baseados no mercado, de fato, nos padrões das comunidades.

Padrões de mercado representam especificações de tecnologia não-proprietária que ajudam a estabelecer, entre outras coisas, as características de uma linhagem consistente (como os transportes, interface, formato de mensagem, etc) da

tecnologia da arquitetura. No entanto, o uso de padrões da indústria por si só não garante que os serviços estarão intrinsecamente interoperáveis.

Para dois programas de software serem totalmente compatíveis, convenções adicionais (tais como modelos de dados e políticas) devem ser respeitadas. É por isso que as empresas devem estabelecer e fazer cumprir as normas do projeto. Deixar de padronizar e regulamentar a padronização dos serviços dentro de um determinado domínio rasgará o tecido da interoperabilidade cuja realização é confiada nos benefícios estratégicos. Este princípio não só defende o uso de padrões de projetos da empresa, ele também lembra que, sendo possível e viável, os padrões de projetos personalizados devem se basear e incorporar normas já usadas pelo mercado e pela comunidade em geral.

Manter uniformidade no exterior, permitindo a diversidade no interior.

Federação pode ser definida como a unificação de um conjunto de entidades diferentes. Mesmo permitindo que cada entidade seja independente regida por dentro, todas as entidades concordam em aderir a uma frente comum e unificada.

O surgimento de uma camada de federação que abstrai os detalhes de implementação do serviço é uma parte fundamental da arquitetura orientada a serviços, enquanto a publicação de um conjunto de endpoints representa serviços individuais dentro de um determinado domínio numa forma unificada. Realizar isso geralmente envolve um atendimento unificado baseado na comunicação de padrões de mercado e padrões de projetos. A consistência entre as unidades e os serviços é a chave para realizar interoperabilidade intrínseca.

Uma camada de endpoint federada logo ajuda a aumentar as oportunidades de explorar a diversidade de opções de fornecedores. Por exemplo, um serviço pode precisar ser construído sobre uma plataforma completamente diferente de outros serviços. Enquanto estes serviços mantêm end-points compatíveis, a governança de suas respectivas implementações pode permanecer independente. Isso não só evidencia que os serviços possam ser construídos utilizando diferentes meios de aplicação (tais como EJB, .NET, SOAP, REST, etc), mas também enfatiza que diferentes plataformas intermediárias e tecnologias podem ser utilizadas em conjunto, conforme necessário.

Perceba que este tipo de diversidade vem com um preço. Este princípio não defende a diversificação em si - ele simplesmente recomenda que a seja permitida a diversificação quando ela se justificar, assim tecnologias e plataformas "best-of-breed" podem ser construídas afim de maximizar o cumprimento dos requisitos de negócios.

Identificar os serviços através da colaboração com as partes interessadas de negócios e tecnologia.

A fim de que soluções de tecnologia sejam orientadas ao negócio (business-driven), a tecnologia deve estar em sintonia com o negócio. Portanto, outro objetivo da computação orientada a serviços é alinhar tecnologia e negócios. A fase em que esse alinhamento é realizado inicialmente é durante a análise e modelagem de processos que geralmente precede o desenvolvimento e entrega de serviço.

O ingrediente crítico para realizar a análise orientada a serviços é fazer com que especialistas do negócio e tecnologia trabalhem lado a lado para identificar e definir os serviços candidatos. Por exemplo, especialistas em negócios podem ajudar a definir com precisão os contextos funcionais relativos aos serviços centralizando-se no negócio, enquanto especialistas em tecnologia podem dar contributos pragmáticos a fim de garantir realismo da granularidade e definição conceitual dos serviços em relação a seus eventuais ambientes de implementação.

Maximizar a utilização do serviço, considerando o alcance atual e futuro de utilização.

O escopo de um projeto de SOA pode ser considerado para uma empresa toda ou limitado a um domínio da empresa. Seja qual for o âmbito de aplicação, um limite pré-definido é criado para englobar um inventário dos serviços que precisam ser modelados conceitualmente antes que possam ser desenvolvidos. Em modelar múltiplos serviços em relação uns aos outros, nós estabelecemos um plano de serviços que eventualmente serão construídos. Este exercício de modelagem é crítico quando tentamos identificar e definir os serviços que podem ser compartilhados por diferentes soluções.

Existem diversas metodologias e abordagens que podem ser usadas para realizar as fases de análise orientada a serviços. No entanto, um traço comum entre todos eles é que os limites funcionais de serviços podem ser normalizados para evitar redundância. Mesmo assim, a normalização de serviços não se faz somente para que eles sejam altamente reutilizáveis. Outros fatores entram em jogo, como a granularidade de serviço, a autonomia, a gestão de estado, escalabilidade, modularidade e, no escopo em que a lógica de serviço é suficientemente genérica para que ele possa ser efetivamente reutilizado. Estes tipos de considerações guiadas pela experiência no negócio e na tecnologia oferecem a oportunidade de definir os serviços que capturam a corrente de utilização de requisitos, enquanto, tendo, flexibilidade de se adaptar às mudanças futuras.

Verifique se os serviços satisfazem necessidades e objetivos de negócios.

Como com qualquer coisa, os serviços podem ser usurpados. Quando crescemos e mantemos um portfólio de serviços, sua utilização e efetividade no comprimento de requisitos de negócios precisam ser verificadas e medidas. Ferramentas modernas fornecem várias formas de monitoramento de utilização de serviços, mas existem valores intangíveis que também precisam ser considerados para certificar que os serviços não são usados apenas porque estão disponíveis, mas sim verificar se eles estão verdadeiramente em conformidade com as necessidades de negócio e atendendo expectativas. Isto é especialmente verdade com serviços compartilhados que se baseiam múltiplas dependências. Não só os serviços compartilhados exigem infra-estrutura adequada para garantir a escalabilidade e confiabilidade para todas as soluções que os reutilizam, eles também precisam ser concebidos e estendidos com muito cuidado para garantir que seus contextos funcionais nunca sejam distorcidos.

Evoluir serviços e suas organizações em resposta ao uso real.

Esse princípio orientador lança diretamente a volta do valor da declaração "aperfeiçoamento evolutivo sobre a busca de perfeição inicial", como também o objetivo global de manter um alinhamento dos negócios e da tecnologia.

Nós nunca podemos esperar que confiar em suposições quando se trata de determinar granularidade de serviço, a gama de funções que precisa executar os serviços, ou como os serviços deverão ser organizados em composições. Com base em qualquer medida de análise que somos capazes de realizar, inicialmente, um determinado serviço será atribuído um contexto definido funcional e conterá uma ou mais capacidades funcionais que provavelmente vão envolvê-lo em uma ou mais composições de serviços.

Como os requisitos de negócio do mundo real e as circunstâncias mudam, o serviço pode precisar ser aumentado, ampliado, reformulado, ou talvez até mesmo substituído. Princípios de projetos de orientação a serviços (service orientation design principles) contróem flexibilidade nativa em arquiteturas de serviços de modo que, como programas de software, são flexíveis e adaptáveis à mudança e mutáveis em resposta as demandas do mundo real.

Separe os diferentes aspectos de um sistema que mudam em ritmos diferentes.

O que faz sistemas monolíticos baseados em silos inflexíveis é que mudanças podem ter impacto significativo em sua utilização atual. é por isso que muitas vezes é mais fácil criar novas aplicações baseadas em silos em vez de aumentar ou estender as já existentes.

A lógica por trás da separação de interesses (famosa teoria de engenharia de software) diz que um grande problema pode ser resolvido de forma mais eficaz quando ele é decomposto em um conjunto de pequenos problemas ou preocupações. Aplicando a orientação a serviços na separação de interesses, vamos construir unidades correspondentes da lógica de solução que resolve as preocupações individuais, permitindo-nos agregar as unidades afim de solucionar o problema maior, além de nos dar a oportunidade de agregá-los em diferentes configurações, resolvendo outros problemas.

Além de promover a reutilização de serviços, esta abordagem apresenta numerosas camadas de abstração que ajudam a proteger os sistemas de serviços compreendidos entre os impactos das mudanças. Essa forma de abstração pode existir em níveis diferentes. Por exemplo, se um recurso legado encapsulado por um serviço precisa de ser substituído, o impacto dessa mudança pode ser atenuado, desde que o serviço seja capaz de manter o seu end-point e comportamento funcional original.

Outro exemplo é a separação de lógica agnóstica da lógica não-agnóstica. O primeiro tipo de lógica tem elevado potencial de reutilização, se for multi-propósito torna-se menos propenso a mudanças. Lógica não-agnóstica, por outro lado, representa tipicamente partes de um processo de negócios pai, que muitas vezes são mais voláteis. Separando esses respectivos tipos de lógicas em diferentes camadas de serviços logo se introduz abstrações que permitem, a partir do impacto da mudança, reutilização de serviços enquanto protegem serviços e qualquer solução que os utilize.

Reduzir dependências implícitas e publicar todas as dependências externas para aumentar a robustez e reduzir o impacto da mudança.

Um dos mais conhecidos princípios de design de orientação a serviços é o acoplamento frouxo. Como uma arquitetura de serviços está estruturada internamente e como os serviços relacionam-se com programas que os consome (que pode incluir outros serviços), tudo se resume a dependências que são formadas individualmente movendo partes que são partes da arquitetura de serviços.

Camadas de abstração ajudam a aliviar a mudança evolutiva por localizar os impactos das mudanças em regiões controladas. Por exemplo, dentro das arquiteturas de serviços, service façades (fachadas de serviço) podem ser usadas para abstrair partes de implementação a fim de minimizar o alcance das dependências de implementação.

Por outro lado, contratos técnicos de serviços publicados precisam divulgar as dependências que os serviços consumidores farão quando estão a fim de interagir com o serviço em questão. Por reduzir dependências internas que podem afetar estes contratos técnicos quando mudanças surgirem, evitamos proliferação do impacto dessas alterações nos consumidores do serviço dependentes.

Em cada nível de abstração, organize cada serviço em torno de uma unidade funcional coesa e gerenciável

Cada serviço exige um contexto funcional bem definido que determina a lógica que faz e que não pertencem ao limite funcional do serviço. Determinar o escopo e detalhamento dos limites funcionais de um serviço é uma das responsabilidades mais importantes durante o ciclo de vida do serviço.

Serviços com densa granularidade funcional pode ser demasiadamente inflexível e não eficaz, especialmente se espera que esses sejam reutilizáveis. Por outro lado, o excesso de granularidade funcional dos serviços pode encarecer a infraestrutura de composições de serviço, aumentando muito o número de membros da composição.

Determinar o equilíbrio certo de escopo funcional e granularidade requer uma combinação de expertise de negócios e tecnologia, e ainda exige uma compreensão de como os serviços dentro de um limite determinado se relacionarão.

Muitos dos princípios descritos neste manifesto ajudarão a fazer essa determinação em prol do posicionamento de cada serviço como um ativo da TI, capaz de favorecer uma empresa em direção ao estado alvo em que os benefícios estratégicos da computação orientada a serviços são realizados.

Em última análise, no entanto, será sempre a obtenção do valor real mundo dos negócios, que dita, desde a concepção até a entrega para repetição do uso, o caminho evolutivo de qualquer unidade de funcionalidade orientada a serviços.

Agradecimentos: Eu escrevi o conteúdo desta página no fim de semana de 21-22 novembro de 2009, para o próximo livro: Next Generation SOA que ainda estava em desenvolvimento no momento. Eu gostaria de agradecer Prentice Hall por permitir que o conteúdo fosse publicado abertamente neste site como um complemento ao original SOA Manifesto. Eu também gostaria de agradecer aos

membros do grupo e os membros da comunidade SOAPatterns.org que forneceu feedback e assistência com estas anotações. Muitos dos membros do grupo têm publicado seus próprios artigos, blogs e jornais sobre o Manifesto de SOA e encorajo a todos que os procurem, para mais comentários e insights.

- Thomas Erl

Translator

Eduardo Xavier

[Original SOA Manifesto](#) [PDF](#) [About the Translators](#)

Copyright © 2009-2011