

**José Carlos Lazzeri**

lazeri@soatrack.com.br é profissional da área da Tecnologia da Informação, com formação em Tecnologia do Processamento de Dados, Sistemas de Informação e Engenharia da Computação. É autor do livro *Arquitetura Orientada a Serviços – Fundamentos e Estratégias*.

SOA

modelagem de serviços com BPMN

Nas arquiteturas SOA, o serviço é o elo que aproxima a Tecnologia da Informação dos objetivos organizacionais. As técnicas de modelagem têm o desafio de preservar este diferencial de SOA, materializando nos negócios seu efeito inovador. Uma estratégia equivocada no tratamento do serviço torna SOA apenas um pouco mais do mesmo e as frustrações logo tendem a aparecer. Na modelagem com BPMN, a aproximação se dá através de um artefato de modelagem consistente, adequado às duas perspectivas de visão e próprio para garantir a precisão computacional.

Neste artigo, o autor demonstra como modelar serviços utilizando fluxos de processos em BPMN. São percorridos os procedimentos de modelagem, desde a captura dos requisitos até a provisão, dando ao leitor uma visão ampla dos benefícios desta promissora estratégia.

A arquitetura SOA conquistou grande notoriedade e hoje é aceita como uma alternativa robusta para o alinhamento da Tecnologia da Informação (TI) às estratégias de negócios. Em um contexto econômico drasticamente modificado pela crise, isto se torna ainda mais relevante. SOA possui várias características importantes. A flexibilidade para acomodar a dinâmica dos negócios é uma das que se destaca. SOA alinha-se aos negócios apoiando-se no conceito de serviço, através do qual traz uma nova forma para representar as capacidades em prover ou requerer tecnologia da informação dentro dos modelos de negócio modernos.

Técnicas avançadas em modelagem de serviços devem preservar o diferencial de SOA, capturando a essência da visão de negócios e refletindo em toda sua extensão dentro dos sistemas computacionais. Artefatos de modelagem pouco estruturados nem sempre permitem isto, mostrando-se falhos na representação dos preceitos essenciais para a estabilidade dos sistemas. A modelagem de serviços feita com o auxílio de fluxos de processo em BPMN tem demonstrado extraordinário vigor, facilitando a representação de muitos aspectos de negócio que escapam aos recursos de modelagem convencionais.

Este artigo reproduz um cenário de modelagem de serviços utilizando a linguagem BPMN. São percorridos os principais procedimentos envolvidos nas atividades de modelagem, com detalhes da captura das referências de negócio e sua transposição para sistemas computacionais. Nos tópicos finais, são representadas as formas de provisão, apoiadas por tecnologia de componentes EJB.

Arquitetura SOA

A arquitetura SOA (Service Oriented Architecture) é o modelo que descreve a disposição e os princípios que regem os elementos para a provisão e o intercâmbio de serviços entre entidades provedoras e requisitantes.

As entidades provedoras capturam os anseios dos requisitantes e os modelam em abstrações que sintetizam suas necessidades: os serviços.

O modelo arquitetônico SOA tem adquirido muitas faces, acomodando inúmeras visões. Em seu estágio atual, consolida qualidades herdadas de modelos anteriores, especialmente dos objetos distribuídos e componentes, como: distribuição, reutilização, contratos, estilo modular, baixo acoplamento e capacidade de integração. Estas são as características básicas de SOA, as mais aparentes e, várias vezes, aquelas que prevalecem em muitas implementações. Em sua visão avançada, SOA se coloca em um patamar ainda mais elevado, acrescentando alternativas inovadoras e fomentando sofisticadas formas de negócio.

A preservação das características inovadoras trazidas por SOA é um desafio que se inicia na formulação das estratégias de sua adoção dentro de cada empresa e culmina nos princípios de engenharia de software apropriados para torná-los realidade nos sistemas computacionais. Em um segmento que tenta se aprimorar, buscando maior produtividade e redução de custos, a fuga de modelos proprietários, optando-se por padrões tecnológicos bem estabelecidos é uma estratégia que deve prevalecer.

A aparência ideal de um modelo avançado para SOA deve enfatizar a tendência à alta decomposição e ao baixo acoplamento. A decomposição deve estar voltada para o aspecto modular não exclusivamente dos elementos tecnológicos, mas também dos elementos de negócio. O baixo acoplamento deve estar voltado para garantir a suficiente flexibilidade do modelo em absorver a dinâmica dos mercados e a interação com uma geração de públicos de estilo de vida eminentemente digital.

O serviço é o centro de atenções de um modelo SOA. A noção de serviço possui a semântica necessária para disciplinar as trocas de mercado entre provedores e requisitantes, dando suporte inovador às múltiplas interrelações de negócio em um contexto extensamente globalizado.

Serviços

O serviço é a representação abstrata do trabalho a ser provido em uma relação específica entre entidades provedoras e requisitantes. Este conceito de serviço vai além dos limites meramente tecnológicos, que são insuficientes para evidenciar as riquezas de SOA. Ele permite delimitar a capacidade de provisão, a necessidade potencial do requisitante e a unidade de troca entre estes agentes nos mercados.

A insuficiência semântica é uma das fraquezas mais visíveis dos modelos orientados a serviços. Os conceitos de SOA flutuam entre escassas abordagens acadêmicas e fortes direcionamentos de mercados. Neste cenário de incertezas a melhor defesa para uma estratégia de vida longa é ancorá-la em padrões, habilitando o modelo para atuar em um panorama de negócios sem fronteiras, utilizando linguagens aceitas universalmente.

No tratamento dos serviços em um modelo padronizado, o suporte à modelagem, descrição, orquestração e implementação da provisão não deve desconsiderar os padrões BPMN, Web Services, WS-BPEL e JEE. O uso destes

padrões cria as fundações para o amplo espectro de possibilidades que SOA pode propiciar.

A semântica contida no conceito de serviço que utilizamos adere perfeitamente a estes padrões, permitindo transitar com precisão pelas várias atividades de um processo de desenvolvimento renovado, sem colidir com barreiras conceituais intransponíveis. Muitos outros conceitos existentes não nos dão esta mesma convicção.

Modelando negócio com metodologia de negócio

Uma das mais notórias fraquezas das estratégias convencionais de modelagem de sistemas é a de tentar modelar negócio com a técnica de modelar sistemas. Metodologias baseadas em “use cases”, extensamente utilizadas no tratamento de requisitos, são insuficientes para capturar a riqueza dos modelos de negócio, criando bases frágeis, que propagam deficiências por todo o processo de desenvolvimento. Ao prevalecer as técnicas de sistemas, o aperfeiçoamento do modelo de negócio fica em segundo plano e as lacunas somente aparecem em momento de operacionalização. As consequências disto se refletem na baixa qualidade dos sistemas, no alto custo de manutenção e nas constantes insatisfações.

Não se pode confundir artefatos de modelagem de sistemas com artefatos de modelagem de negócio, nem pretender que estes possam ser substituídos. Na modelagem do negócio, a ênfase é o aprimoramento do modelo para que possa ser operado da forma mais otimizada possível dentro da organização. As técnicas aplicadas têm que focar neste objetivo. Transformá-lo em sistema deve ser um passo seguinte.

A representação visual como técnica de modelagem facilita a abstração e os problemas tendem a ser compreendidos com mais facilidade. Se estas abstrações forem representadas em artefatos estruturados, de precisão e possíveis de serem manuseados pelo computador, os avanços serão inquestionáveis, o nível de consistência dos modelos de negócio se modificará, influenciando diretamente na qualidade dos sistemas. O fluxo do processo é um artefato que preenche estas exigências e suporta modelar negócio indo além da mera representação estática dos requisitos.

Ao decidir-se por um novo modelo arquitetônico, é também um momento propício para tentar atenuar as distâncias entre TI e os negócios, considerando novas opções. SOA não expressará suas verdadeiras possibilidades sem uma abordagem que resguarde a essência dos negócios e a reflita nos processos automatizados. A ótica restritiva das técnicas de modelagem de sistemas não é suficiente para isto.

BPMN

Um dos aspectos presentes no avanço e amadurecimento de um segmento industrial é a proliferação de padrões. Modelos proprietários levam à formação de monopólios, altos custos e engessamento da evolução.

A linguagem BPMN (Business Process Management Notation) faz parte de uma nova geração de linguagens e se estabelece vigorosamente como padrão no tratamento de processos de negócio. A BPMN acrescenta várias qualidades, tanto em termos de negócio quanto para a renovação das estratégias de modelagem de sistemas.

Em termos de negócio, a linguagem demonstra ser de ótima aceitação e de

fácil aprendizado para um público que não considera o tecnicismo como ponto focal. Os modeladores de negócio se sentem confortáveis para expressar seus conceitos e necessidades de forma rica e consistente. Seu formato visual é intuitivo e, sendo padronizado, facilita a comunicação. BPMN coloca à disposição dos modeladores de negócio um artefato adequado à representação do objeto em análise: o negócio. Se contrastada com as técnicas de levantamento de requisitos comuns, muitas vezes colocada como a única alternativa para compreender o modelo de negócio, pode-se dizer que o diferencial foge às possibilidades de comparação.

Em termos da modelagem de sistemas, a BPMN nos permite a passagem formalizada e mais completa dos modelos de negócio para TI, diminuindo as imprecisões das diferentes formas de comunicação utilizadas por estes dois públicos. Sendo BPMN um padrão, os artefatos da modelagem de negócio podem ser mapeados para modelos de sistemas de forma assistida por computador, elevando ainda mais os níveis de precisão. A especificação contempla as regras de mapeamento para WS-BPEL (Web Services-Business Process Execution Language), mas isto é apenas uma das várias possibilidades que podem ser exploradas.

A BPMN torna a modelagem de sistemas menos dependente da “invenção”, pois está presente em várias fases do desenvolvimento, guiando a modelagem com artefatos formais, muito mais adequados ao uso do computador. A aplicação dos princípios de engenharia fica reforçada e o uso de estratégias de transformação de modelos, como a MDA (Model Driven Architecture), muito mais efetivo.

Um dos diferenciais de BPMN é que seu padrão visual permite representar modelos em diversos níveis de abstração, possibilitando adequar a representação visual ao escopo de debate e ao tipo de público participante. Ao facilitar a transição da visão de negócio para a visão de sistemas, BPMN auxilia SOA no essencial de sua proposição de aproximar-se dos negócios, dando respostas condizentes com papel reservado à tecnologia da informação nos dias atuais.

Coreografia e orquestração em serviços

O suporte à execução de processos está cada vez mais presente nos modelos arquitetônicos orientados a serviço. Os fluxos de processos nos permitem reforçar o aspecto modular embutido na noção de serviço e cumprem um papel singular em momento de execução, perfazendo coreografia e orquestração.

A coreografia trata da descrição do comportamento colaborativo observável que ocorre na interação com os serviços. Está diretamente ligada ao comportamento previsto no sistema, definindo a sequência e as condições sob as quais os múltiplos agentes atuam em cooperação. A coreografia normalmente aborda as interações para troca de mensagens sob o ponto de vista do usuário.

A orquestração, por sua vez, define a sequência e as condições em que as invocações ocorrem no âmbito dos serviços, intervindo no padrão de troca de mensagens aplicado para que o serviço seja provido.

Em uma estratégia de desenho de aplicação é importante manter a orquestração em uma camada independente, preservando um dos aspectos essenciais de SOA para lhe garantir flexibilidade: o baixo acoplamento.

A orquestração pode ser feita de várias formas, inclusive através de um desenho de interação baseado em MVC (Model View Controller). A orquestração através de WS-BPEL tem nos mostrado ótimos atrativos. Embora eventuais inquietações trazidas pela mudança de paradigma, com o uso de WS-BPEL os

padrões de manuseio de serviço ficam reforçados e a separação da orquestração inteiramente preservada.

Uso do fluxo do processo em BPMN na modelagem do serviço

Os avanços nos padrões de tratamento de processos com BPMN e WS-BPEL criaram um novo momento para o gerenciamento de processos, possibilitando que estejam cada vez mais presentes nos sistemas automatizados.

A modelagem de serviços através da BPMN coloca o fluxo do processo como âncora para uma arquitetura SOA diferenciada, aquela com perfil para resguardar a proximidade com a visão de negócio.

Como veremos nos tópicos a seguir, a linguagem BPMN nos permite capturar a essência da visão de negócio e transpô-la com precisão para os modelos de serviços, gerando artefatos robustos que criam uma base sólida para a modelagem dos sistemas.

Cenário de modelagem de serviços

A partir deste ponto, este artigo explora um cenário de modelagem de serviços com o uso de BPMN. Nele demonstramos os princípios de modelagem aplicando as técnicas diretamente em um exemplo. Isto facilitará a visualização dos efeitos progressivos produzidos sobre um contexto real de negócio.

Serão percorridas as atividades de modelagem de serviços sobre um tema de negócio bastante familiar, de fácil compreensão para o leitor, pois existem inúmeras referências na internet. O cenário do exemplo é o da comercialização de produtos (possivelmente através de um canal eletrônico) no qual provedor e requisitante interagem através de serviços. O exemplo explorará uma parte deste modelo de negócio, que julgamos suficiente para as demonstrações pretendidas. Obviamente, não se pretendeu descrever todos os seus processos. Um fluxo descrito em BPMN representará a pequena cadeia de valor destacada para o exemplo, a partir da qual serão efetuados os procedimentos de modelagem dos serviços até a provisão.

Modelo de negócio

A primeira atividade da modelagem de sistemas para arquiteturas orientadas a serviços não pode dispensar a construção de um consistente modelo de negócio. Iniciar a modelagem dos serviços, ancorada em um modelo de negócio frágil, significaria propagar inconsistências por todo o processo de desenvolvimento. Já neste primeiro momento, a BPMN atua para fortalecer o método de engenharia, diferenciando-se em relação a outras abordagens e criando as bases para um modelo final de alto nível de precisão.

No modelo de negócio, o modelador captura a essência daquilo que a empresa deseja oferecer para seus clientes. Através do fluxo do processo, ele representa como esse modelo de negócio será executado. Muitas técnicas de confecção de modelos de negócio poderão ser aplicadas. Sua forma de execução representada através de um fluxo, porém, dá a oportunidade imediata para que sejam aplicadas todas as técnicas de BPM (Business Process Management), aprimorando o modelo até que a forma mais otimizada para os negócios seja encontrada. Com isto, garante-se também que estará sendo passado um artefato robusto para os trabalhos de automação.

Modelagem do processo de negócio

A partir dos insumos contidos no modelo de negócio são aplicadas as técnicas de modelagem para a produção dos fluxos dos processos. A exigência essencial é a de que esteja representado em um formato padrão, capaz de ser convertido em artefatos de sistemas e suporte representar os requisitos de negócio com completude e precisão. A linguagem BPMN detém estas qualidades e será aplicada para gerar o modelo que será a base para todas as atividades de modelagem.

Neste cenário de demonstração, o fluxo do processo de negócio, em seu primeiro estágio, representará a forma de execução do modelo de negócio “compra de produtos” em alto nível de abstração. Neste ponto inicial, o foco da modelagem é estritamente o de encontrar a melhor performance do processo sob a ótica de negócio, ainda sem qualquer preocupação com as restrições dos sistemas computacionais.

A sequência dos elementos do fluxo mostra o trabalho executado por cada atividade sobre seus objetos de entrada, produzindo uma saída. A atividade, portanto, representa a ação que acontece sobre alguma coisa, de modo que seus nomes devem procurar explicitar a natureza do trabalho executado. Resumidamente, a formação dos nomes através de um verbo, mais complemento e sujeito indeterminado, nos dá uma boa representação do trabalho que a atividade representa.

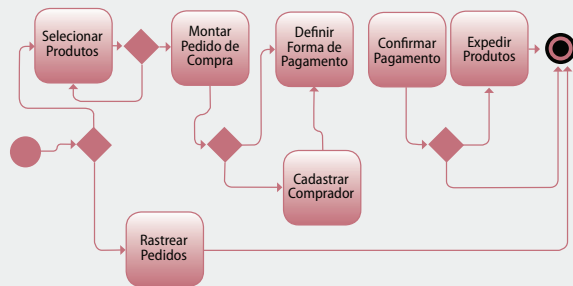


Figura 1. Processo de compra modelado.

Análise do fluxo

- A figura1 contém uma rede de elementos de fluxo em BPMN, que descrevem como o modelo de compra de produtos será executado dentro da organização.
- O fluxo representa a síntese daquilo que o modelador de negócio deseja comunicar para seus públicos. Note que além de informar a natureza do trabalho que ocorrerá dentro do modelo, ele consegue transmitir também a dinâmica de como estas ações ocorrerão.
- Podemos perceber que as bases deste simples modelo de negócio estão bastante explícitas, dispensando uma documentação adicional para sua compreensão geral.
- Mesmo com poucos elementos, é possível concluir de que tema se trata e quais as funcionalidades do negócio que o modelador pretende tratar. A riqueza embutida nos elementos de fluxo permite sintetizar o conhecimento, que em outro formato poderia ocupar vários parágrafos de descrição.
- Os nomes das atividades cumprem um papel importante, pois foram definidos de forma a traduzir a semântica do trabalho efetuado por cada uma delas, dando ainda mais capacidade ao modelo de transmitir informação. Nos próximos passos, conforme o modelo for sendo refinado, esta capacidade tende a se encorpar ainda mais.

Um aspecto a se observar é o de que o fluxo possivelmente estaria inserido em um conjunto de outros processos e normalmente seria originado como consequência de um momento anterior, no qual os modelos de negócio da organização estariam em nível de abstração ainda mais alto. Em modelos complexos, estes níveis formarão grandes cadeias, até o desmembramento adequado à perfeita compreensão. Ou seja, a modelagem de processos dentro de uma empresa não se inicia em um processo específico de compra, estoque, pagamento, ou coisa deste nível, e sim nos processos mais macros da organização.

Outro aspecto importante a se notar é o de que, neste primeiro instante, o modelador de negócio preocupou-se em expressar sua necessidade de negócio, sem condicionar-se demasiadamente aos aspectos lógicos do fluxo. Se assim não se proceder, o fluxo tende a tornar-se complexo já nas primeiras versões e os princípios de negócios podem ficar em segundo plano, ocultando lacunas, com péssimas consequências para os sistemas. É importante que nesta fase prevaleça a visão de negócio. A precisão da lógica de fluxo não é dispensada, mas não deve sobrepor-se aos objetivos de captura dos princípios de negócio da forma mais completa possível.

Preparação do processo para automação

Um fluxo de processo de negócio, em seu formato inicial, enfatiza os aspectos de negócio e a forma mais otimizada de execução daquele modelo dentro da organização. No momento que este fluxo se depara com as restrições das técnicas de modelagem de sistemas, as dissonâncias entre esses dois mundos certamente se evidenciarão. A conformação do fluxo neste primeiro momento de modelagem dificilmente estará totalmente ajustada às exigências de um modelo computacional ou às peculiaridades de uma estratégia de engenharia. Este fluxo deve então ser preparado para isto. A atividade de “Preparação do processo para a automação” sintetiza os procedimentos de harmonização do cenário de negócios com o de sistemas, buscando atribuir ao fluxo do processo uma conformação pertinente às técnicas que serão aplicadas daí para frente. Em uma arquitetura SOA, vale dizer, submeter-se a um conjunto de técnicas em que encontre correspondência com os preceitos dos modelos orientados serviço.

Delimitação dos serviços

A metodologia de preparação deve ser adequada aos princípios de uma engenharia de software consistente. Aplicamos aqui a metodologia constante do livro “Arquitetura Orientada a Serviços – Fundamentos e Estratégias”. Ela nos remete facilmente a um modelo consistente com os padrões de orientação para serviço mais presentes no mercado de TI atual (Web Services, WS-BPEL, EJB etc.): resume-se em balancear o fluxo do processo, fazendo com que cada atividade do fluxo corresponda a exatamente um serviço em um modelo SOA padronizado. Em relação à visão de negócio, significa identificar, através de uma atividade BPMN, cada serviço que o modelador de negócio pretende prover para seus potenciais clientes. A “loja de comercialização de produtos” e os “compradores” serão os provedores e requisitantes principais envolvidos nos cenários deste modelo de negócio. Uma observação pertinente a ser feita neste ponto é a seguinte: algumas complexidades características dos sistemas computacionais, como o tratamento extensivo de erros, segurança, peculiaridades de canais, técnicas de interação humana, formato de mensagens etc., estão sendo abstraídas, evitando assim sobrecarregar o fluxo com informações não imprescindíveis para o tema em demonstração.

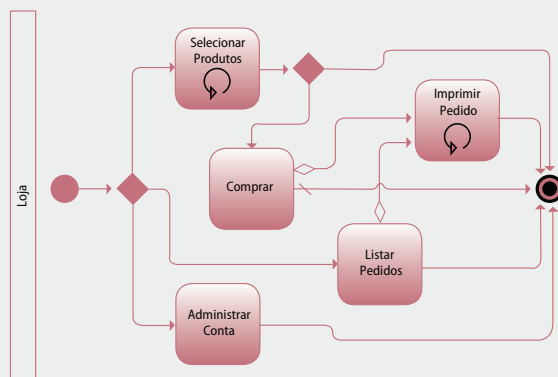


Figura 2. Processo preparado para automação.

Na preparação do processo os princípios da orientação para serviços e as nuances computacionais colocaram suas condições, de modo que os elementos do fluxo, embora não tenham perdido o ideal de negócio, passaram a refletir também estas características. No fluxo da etapa anterior, o modelador de negócio havia se preocupado apenas em descrever o processo de compra da melhor forma para o negócio. Possivelmente aplicou técnicas de BPM para otimizá-lo, mas sem preocupar-se ainda com as questões da modelagem orientada para serviço. Note que agora cada atividade está em um formato que poderá ser executado como uma parte específica e independente do trabalho, gerando resultados bem definidos para os potenciais requisitantes. Ou seja, a noção de serviço vai tomando forma em termos de elemento de fluxo em BPMN. A visão de negócio vai se aproximando da visão orientada para serviços, mesmo antes da aplicação das técnicas específicas de modelagem de sistemas.

Análise do fluxo

- O fluxo da figura 2 mostra agora as atividades que representam os cinco serviços que o provedor irá oferecer aos potenciais requisitantes. A metodologia ajustou o fluxo, sintetizando-o em atividades do tipo tarefa, cada uma em um nível de abstração apropriado para que equivalha precisamente a um único serviço.
- O pool denominado “Loja” foi adicionado para identificar o único participante do modelo distinguido até aqui.
- Algumas atividades tiveram seus nomes ajustados para melhor encaixar-se às mudanças feitas no fluxo e a síntese do trabalho que passaram a representar. Posteriormente, cada tarefa será implementada como um serviço em arquitetura SOA.
- Os procedimentos necessários para efetuar o trabalho previsto em cada tarefa constam do modelo de negócio, ainda como documentação e serão transpostos posteriormente para o formato de provisão.
- A metodologia que delimitou os serviços considerou o trabalho que cada atividade irá efetuar. A tarefa “Pesquisar Produtos” do fluxo anterior foi modelada como um serviço, representado por uma atividade que contém um marcador de “loop” da BPMN, evidenciando que poderá ser executada iterativamente sem que o fluxo se mova para frente.
- O cadastramento do comprador também foi desacoplado do fluxo de compra para um serviço independente, através da atividade “Administrar Conta”. Se esta pretender efetuar todas as regras comuns de administração de contas, como alterações, exclusões etc., deverá ser desmembrada para representar também estes serviços.
- Uma tarefa de “Imprimir Pedido”, contida nas regras, foi destacada como um serviço independente, e poderá ser executado iterativamente. Este serviço foi condicionado a uma precedência de ações que passam por

outros serviços deste modelo de negócio. Em outro cenário, ele poderá ser precedido de outros serviços específicos, sem perder a autonomia. Isto é bastante comum em cenários complexos onde se busca eliminar a redundância de procedimentos já na modelagem do processo.

- As demais tarefas do fluxo anterior foram absorvidas pelos serviços e aparecerão no momento em que estes forem desmembrados em operações.
- Um aspecto interessante de se notar é o de que a noção de serviço se materializa sob ótica de negócio. Não houve qualquer condicionante computacional influenciando na delimitação destes. Como a noção de serviço deriva dos preceitos de negócio, isto se torna natural. Se na prática do leitor isto não vem acontecendo, vale revisar o conceito de serviço que está sendo aplicado, pois ele pode estar tentando refletir apenas funções computacionais, distante dos princípios de SOA.

Modelagem da orquestração

Na modelagem da orquestração são definidas a sequência e as condições em que as invocações ocorrerão. As tarefas são revisadas para aproximarem-se das exigências das técnicas de modelagem, neste caso, das tecnologias orientadas a serviço que serão aplicadas. A realidade presente nos sistemas existentes na organização também deverá ser considerada.

Criação dos subprocessos dos serviços

O procedimento inicial é a aplicação do recurso de “sub-process” da linguagem BPMN, visando desmembrar o serviço em operações e dar autonomia tecnológica aos elementos que o representam. Esta técnica também facilitará a transposição mecânica de cada um dos serviços para os modelos de automação de forma independente.

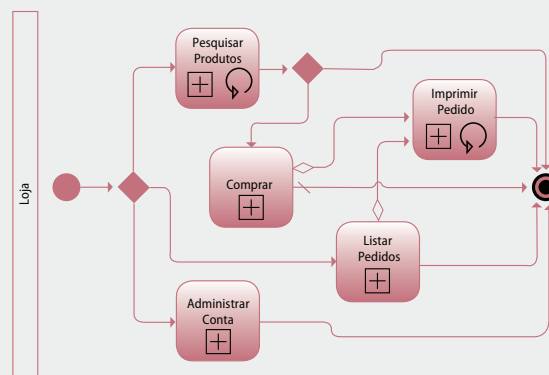


Figura 3. Subprocessos dos serviços.

Análise do fluxo

- As atividades do fluxo anterior foram transformadas em subprocessos. O marcador da BPMN (pequeno quadrado com sinal de adição dentro) visível no elemento de fluxo denota esta condição. Observação: este recurso permite também que as ferramentas de modelagem usem-no para expandir e “colapsar”, facilitando a visualização.
- Cada subprocesso continua representando um dos cinco serviços, bem delimitados em termos visuais e independentes em termos de elementos de fluxo. No momento, todos se encontram na forma colapsada, por consequência, seus elementos internos (as operações) não estão visíveis.

- Vale frisar mais uma vez que é possível perceber claramente a noção de partes expressadas no fluxo. Pelos seus nomes, percebe-se que se tratam de porções de trabalho bem delimitadas, que gerarão resultados perceptíveis, exclusivos e completos para o requisitante.
- Os nomes das atividades permanecem conforme sua regra de formação, pois ainda estamos tratando de fluxo de processo. Quando os serviços forem descritos na WSDL (Web Service Description Language) deverão ter seus nomes convertidos para representarem o produto resultante das ações do provedor na troca de mensagens com o requisitante. Uma combinação de substantivos, o primeiro referindo-se ao objetivo e os seguintes à natureza e ao escopo, formarão boas combinações. Neste caso, teremos: “Pesquisa de Produtos”, “Compra”, “Administração de Conta” etc.

Modelagem das operações dos serviços

Uma questão delicada na lida com a orientação para serviços é a escolha das abordagens corretas para a composição do desenho arquitetônico que será adotado. Se isto não estiver bem resolvido, o modelo SOA criado poderá tornar-se progressivamente mais parecido com os outros modelos que já existiam anteriormente. No final, poderá não passar de uma fila de mensagem recondicionada em um Enterprise Service BUS ou conjunto de Web Services empacotando módulos de código maquiados de serviço, sem qualquer diferencial representativo para o modelo computacional e os negócios da organização. Na conformação dada ao serviço, estas questões tomam forma e ditam as características que o modelo irá ter. Uma destas questões diz respeito a como as regras de negócio do serviço serão modeladas. Nas técnicas de descrição de serviços com o padrão WSDL, o serviço é descrito em fases distintas, que modelam cada troca de mensagem específica: as operações. Ou seja, o serviço não é uma peça monolítica de corpo único, ele é formado de partes que acomodam as complexidades das regras de negócio executadas. Muitos autores abdicam deste fundamento, e tentam contornar esta situação sugerindo a ideia de “serviço atômico” e “serviço composto”, tomando o modelo acoplado e saindo fora dos padrões. Em nossa prática, o uso do padrão WSDL tem se demonstrado preciso e flexível, não havendo razão para não aplicá-lo. Com ele, as definições de “serviço atômico” ou “composto” são totalmente desnecessárias e o modelo se torna muito mais simples.

A operação é a representação abstrata de uma ação suportada pelo serviço. Representa uma interação entre o provedor e o requisitante para uma troca de mensagem. A existência deste elemento na WSDL permite que um serviço seja provido sem limitação na troca de mensagens. Na prática, percebemos o uso intensivo deste recurso, pois a provisão dos serviços normalmente demanda várias trocas de mensagens. Sem isto, teríamos que recorrer a artifícios que não se encaixariam nos padrões disponíveis. Para identificar as operações, usamos nomes que reflitam a ação produzida pela parte do trabalho que será provida. Possivelmente será formado por um verbo no infinitivo mais o complemento verbal, nesta ordem.

A modelagem das operações é feita também com o apoio da BPMN. Isto permitirá o mapeamento do modelo de negócio diretamente para o modelo de execução em WS-BPEL, preservando a produtividade da técnica e a precisão. Para isto, cada subprocesso será desmembrado em outras tarefas, cada uma equivalendo a uma operação da interface do serviço. Este trabalho é feito com base em metodologias apropriadas, como as constantes do livro “Arquitetura Orientada a Serviços – Fundamentos e Estratégias”, ou outra conforme a escolha do leitor. Várias questões importantes e muitas vezes sutis se ligam a esta

parte da modelagem, inclusive aquelas diretamente relacionadas ao conceito de serviço e às técnicas de modelagem de domínio possíveis de serem aplicadas. As nuances deste cenário exigirão do leitor o conhecimento de alguns fundamentos avançados de SOA. A separação precisa dos domínios entra neste escopo e deve ser claramente discernida. Estas questões serão evidenciadas progressivamente neste artigo.

Modelagem das operações do serviço “Compra”

O serviço “Compra” foi o escolhido para ser usado nos procedimentos de modelagem que aplicaremos a seguir. Ele contém os melhores elementos para aplicar a técnica de modelagem que pretendemos demonstrar. Os demais serviços seriam submetidos ao mesmo método. Este serviço estava representado no fluxo da figura anterior pelo subprocesso “Comprar”, na forma colapsada, de modo que os elementos em seu interior não estavam visíveis. Na figura 4, o fluxo está no formato expandido, mostrando um conjunto de elementos representando as várias operações do serviço e a forma como foram encadeadas pela estratégia de orquestração.

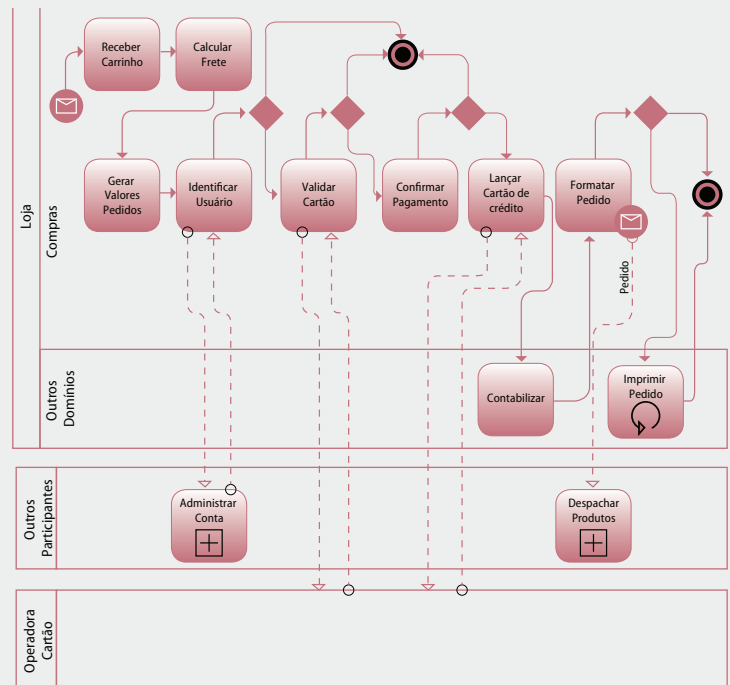


Figura 4. Serviço “Compra” desmembrado em operações.

Análise do fluxo

- O desmembramento do subprocesso gerou várias novas atividades, que podem ser vistas na figura 4, cada uma agora equivalente a uma operação. Mesmo aquelas que se encontram na forma de subprocesso, como “Administrar Conta” etc., para o serviço “Compra”, são vista como uma operação.
- Cada operação representa uma fase dentro do serviço, que acomoda uma parte de regras de negócio de alta coesão e não deve estar acoplada a outras partes. O contraste com o serviço é facilmente perceptível, pois são partes que, isoladamente, não são suficientes para produzir algo de útil para o requisitante.
- Como as tarefas representam operações, seus nomes transmitem a ideia de partes do trabalho provido no serviço.
- A tarefa inicial do processo, denominada “Receber Carrinho”, é dispa-

rada pelo evento que contém a mensagem com os itens colocados pelo comprador no “carrinho de compras”. A modelagem definiu que o serviço “Pesquisa de Produtos” (visível no fluxo da figura 3) componha esta mensagem ao percorrer seu fluxo iterativamente e, ao ser completado, valide a disponibilidade dos produtos e gere o evento previsto.

- As tarefas seguintes modelam as operações que tratam os valores envolvidos no pedido de compra, considerando as características dos produtos e o CEP de entrega. Os valores são então gerados para serem apresentados ao comprador.
- A tarefa “Identificar Usuário” acessa os dados da conta do comprador através da troca de mensagem com outro participante, eventualmente uma certificadora que faz a administração das contas dos usuários. Isto foi evidenciado deslocando-se a tarefa para um “pool” específico. A complexidade do processo de identificação não é conhecida, pois pertence a outro participante. Poderia ser o simples registro dos dados básicos do comprador, ou envolver todas as atividades de administração de contas. A mensagem de retorno define se o processo de compra será encerrado ou continua.
- A operação de validação do cartão de crédito também troca mensagens com outro participante, representado em outro pool. Seus processos não são conhecidos, de modo que as mensagens se dirigem diretamente à borda do pool.
- A operação de confirmação do pagamento formata os dados do pedido para capturar a decisão do comprador. Com a confirmação do pagamento, uma operação para a troca de mensagens com a operadora de cartões será invocada na sequência. Se o pagamento não for confirmado, o serviço é encerrado.
- A operação “Contabilizar” invoca uma tarefa pertencente ao domínio Contabilidade. Ela está deslocada para a “lane” “Outros domínios”, pois trata de regras de negócio que não pertencem ao domínio “compra”. Esta tarefa pertence, portanto, a outro processo. Neste caso, está sendo apenas referenciada através de uma tarefa do tipo “reference”. Este recurso da BPMN permite representar atividades no fluxo sem gerar redundância de implementação.
- A operação “Formatar Pedido” formata e registra o pedido de compra, emitindo um evento de mensagem direcionado ao participante “Distribuidor”. Este dará andamento ao pedido e entregará o produto ao comprador. As complexidades relativas às “operações com mercadorias” ficam por conta deste participante.
- A provisão do serviço encaminha-se para seu encerramento. Opcionalmente uma operação de impressão poderá ser invocada iterativamente para formatar o documento e imprimir. É uma operação de outro domínio, referenciada no processo de compra. Possivelmente será reutilizada em vários outros fluxos.
- O recurso de “swimlane” da BPMN foi usado para a separação de elementos na modelagem, facilitando a compreensão e a visualização de responsabilidades. Foi usado o mecanismo de “pool” para separar as entidades “Loja”, “Outros Participantes” e “Operadora de Cartão”. Estes possivelmente fora dos domínios da organização. Visto que serviço é uma abstração, sua delimitação flutua conforme o modelo de negócio determina. Na troca de mensagem da entidade “Loja” com a “Operadora de Cartão” e a “Distribuidora”, por exemplo, a “Loja” atua como requisitante dos serviços oferecidos por aqueles participantes e os invoca para compor outro serviço, destinado ao requisitante “comprador”.
- **Observação:** as atividades de interação humana não foram integralmente representadas, visto ser uma parte que envolve extensas especificidades e, inclusive, pelo fato de ainda não terem suporte robusto da BPMN e da WS-BPEL.

Contraste entre serviço e operação

O exemplo nos mostra com clareza o contraste entre a noção de serviço e de operação. O serviço se caracteriza pela presença da entidade provedora, provendo algo que atende a uma necessidade do requisitante. As operações, por sua vez, representam as consecutivas trocas de mensagens entre o provedor e o requisitante, modeladas de modo a permitir que a estratégia de orquestração se concretize. Embora as operações sejam partes bem delimitadas, não se confundem com o serviço, pois isoladamente não são suficientes para prover aquilo que o requisitante demanda, nem aquilo que o provedor deseja colocar à disposição dos potenciais requisitantes. De um modo geral, o requisitante não estaria interessado em invocar qualquer uma das operações do fluxo “Comprar” de forma isolada, pois sua necessidade, abstraída no escopo do serviço, não estaria satisfeita. Nos sistemas de informação encontramos várias peças modulares, como componentes, programas, classes, funções, subrotinas etc. O fato de poderem ser chamadas individualmente, ou de serem eventualmente empacotadas em um Web Service, não lhes promove a serviço, nem lhes acrescenta as capacidades demandadas por SOA. É importante que o leitor faça esta distinção, pois ao contrário poderá criar um modelo SOA sob bases falsas, reproduzindo coisas que já existem sem nada de novo acrescentar.

Separação de domínios

Alguns princípios de modelagem de sistemas não devem ser esquecidos. Um destes é a separação de domínios, que nos dá a condição de modelarmos sem que as regras de negócio de um tema se misturem às de outros. A separação de domínio é uma questão crítica também na modelagem de serviços. Uma má interpretação desta questão implicará em equívocos de importantes consequências. Na análise do fluxo do exemplo, podemos perceber uma clara distinção entre a noção de serviço e outros recursos usados para modular sistemas. A distinção se situa na forma com que estes diferentes recursos são modelados para abordar uma classe de problema específica (um domínio). Na modelagem de serviços, os domínios ficam distribuídos nas operações, mantendo a separação. Se fossem juntadas em um único bloco, produziriam um acoplamento notável entre temas completamente diferentes. Neste simples exemplo, se o serviço “Compra” formasse um módulo de software único, estaríamos misturando pelo menos estes domínios: compra, pessoa, meios de pagamento, contabilidade, estoque etc. com extenso acoplamento e graves consequências para a manutenção.

Modelagem e implementação da provisão

Uma vez delimitados os serviços, suas operações e a forma de orquestração, no dirigimos para os procedimentos de modelagem e implementação da provisão, em que as abstrações do trabalho representado nos serviços deverão ser tratadas para acomodar as regras de negócio do modelo.

Provisão de serviços

Modelar a provisão de serviços significa reportar-se aos mais clássicos desafios da modelagem de sistemas de informação, com destaque para o tratamento da regra de negócio de forma modular e flexível, evitando acoplamento.

Os requisitos de um modelo transacional de alta capacidade também não devem ser esquecidos, de modo que a tecnologia escolhida é uma das decisões

importantes a serem tomadas. Nos últimos anos, presenciamos o fantástico avanço e maturidade do middleware JEE (Java Enterprise Edition). Seu robusto suporte aos ambientes distribuídos de alto desempenho nos dá uma forte referência para uma estratégia de provisão. A tecnologia de componentes EJB (Enterprise Java Beans) oferece ótima oportunidade para implementar a provisão de serviços de forma modular, utilizando-se de um padrão de mercado consolidado. Componentes não dependem de SOA, inversamente, porém, modelar a provisão de serviços em SOA sem componentes, apenas empacotando programas em Web Services, pode ser um movimento de alto risco, pois a provisão poderá ficar dispersa em plataformas proprietárias e obsoletas e a credibilidade do modelo será progressivamente enfraquecida. A modelagem da provisão tem como atividade central o tratamento da lógica de negócio contida nas operações dos serviços. Nesta fase, o conjunto de operações normalmente será representado por um conjunto de estereótipos da linguagem UML, extraídos do fluxo em BPMN. Outros tantos elementos poderão ser extraídos mecanicamente do fluxo, como as entradas e saídas das atividades para representar os atributos das classes de domínio, modelo de dados, diagramas de sequência etc. Qualquer técnica de modelagem poderá ser aplicada. Como a provisão será baseada em componentes, a Análise de Domínio se mostra bastante apropriada.

Modelagem e implementação dos componentes

Em um modelo que coleta requisitos baseado em fluxo de processo, a lógica de negócio, em seu alto nível de abstração já estará explicitada nos elementos do fluxo. O passo seguinte é modelar os componentes tratando as regras de negócio contidas no conjunto de operações. Note que a metodologia aplicada na modelagem dos serviços – que definiu o conjunto de operações – já se encarregou de garantir que cada operação pertença a um único domínio. Definir como estas operações estarão agrupadas em componentes, porém, demanda uma atividade de modelagem que considere, inclusive, as fronteiras do que se pretende modelar e a possibilidade de já existirem sistemas implementados anteriormente. Não está no escopo deste artigo discutir modelagem de componentes, de modo que apenas representaremos visualmente o conjunto de Enterprise Beans originados no exemplo. Mesmo sem passar pelo crivo da modelagem, porém, o leitor não terá dificuldade em visualizar que cada componente trata de um conjunto de operações com grande afinidade em termos de classe de problema, transparecendo a alta coesão de cada módulo e, possivelmente, a ausência de qualquer acoplamento. É possível visualizar que cada componente acomoda um conjunto de operações afins, sem misturar os domínios.



Figura 5. Classes dos componentes EJB.

Os componentes acomodam em suas interfaces os métodos correspondentes a cada operação do serviço “Compra”. Note que modelo de provisão está completamente desacoplado do serviço, e cada método do componente mantém-se livre para outras recombinações em qualquer outro serviço, materializando a flexibilização que deve prevalecer em um modelo SOA.

As interfaces expressam uma estratégia padronizada de modelagem, possivelmente a aplicação do padrão “Façade” ou equivalente. Cada método da fachada foi modelado segundo um conjunto de padrões, tratando do fluxo que aciona as classes de domínio que contêm as instruções de linguagem para a execução das regras de negócio. Após codificados, os componentes EJB devem ser publicados para que possam ser acessados pelos mecanismos do padrão Web Services. As ferramentas dão amplo suporte a esta tarefa, criando os artefatos necessários, como as Service Endpoint Interface (SEI) onde estarão declarados os métodos que o cliente poderá invocar sobre o serviço e o mecanismo de proxy para que seja acessado remotamente. A Listagem 1 nos mostra o código Java da interface remota do componente “Pedido”, implementado sob o padrão EJB. Os outros componentes têm conformação semelhante.

Listagem 1. Código Java da interface do componente “Pedido”.

```

package br.com.loja.ejb;
import java.util.Vector;
import java.util.Collection;

/**
 * Remote interface for br.com.loja.ejb.PedidoBean.
 */
public interface Pedido extends javax.ejb.EJBObject {

    /**
     * @param Calcula o valor do frete considerando as
     * características dos produtos e o CEP.
     * @return Collection
     * @throws java.rmi.RemoteException
     */
    public Collection calcularFrete(Vector carrinho)
        throws java.rmi.RemoteException;

    /**
     * @param Gera os valores que compõe o pedido para
     * apresentação ao comprador.
     * @return Collection
     * @throws java.rmi.RemoteException
     */
    public Collection gerarValoresPedido(Vector pedido)
        throws java.rmi.RemoteException;

    /**
     * @param Formata o pedido, gera os documentos e emite
     * mensagem ao distribuidor.
     * @return Collection
     * @throws java.rmi.RemoteException
     */
    public Collection formatarPedido(Vector ordemCompra)
        throws java.rmi.RemoteException;
}
  
```


Listagem 2. WSDL do serviço "Compra".

```

<?xml version="1.0" encoding="UTF-8"?><wsdl:definitions
<wsdl:portType name="Compra">
  <wsdl:operation name="calcularFrete">
    <wsdl:input message="impl:calcularFreteRequest"
      name="calcularFreteRequest"/>
    <wsdl:output message="impl:calcularFreteResponse"
      name="calcularFreteResponse"/>
  </wsdl:operation>
  <wsdl:operation name="formatarPedido">
    <wsdl:input message="impl:formatarPedidoRequest"
      name="formatarPedidoRequest"/>
    <wsdl:output message="impl:formatarPedidoResponse"
      name="formatarPedidoResponse"/>
  </wsdl:operation>
  <wsdl:operation name="lançarCartaoCredito">
    <wsdl:input message="impl:lançarCartaoCreditoRequest"
      name="lançarCartaoCreditoRequest"/>
    <wsdl:output message="impl:lançarCartaoCreditoResponse"
      name="lançarCartaoCreditoResponse"/>
  </wsdl:operation>
  <wsdl:operation name="validarCartao">
    <wsdl:input message="impl:validarCartaoRequest"
      name="validarCartaoRequest"/>
    <wsdl:output message="impl:validarCartaoResponse"
      name="validarCartaoResponse"/>
  </wsdl:operation>
  <wsdl:operation name="gerarValoresPedido">
    <wsdl:input message="impl:gerarValoresPedidoRequest"
      name="gerarValoresPedidoRequest"/>
    <wsdl:output message="impl:gerarValoresPedidoResponse"
      name="gerarValoresPedidoResponse"/>
  </wsdl:operation>
  <wsdl:operation name="identificarUsuario">
    <wsdl:input message="impl:identificarUsuarioRequest"
      name="identificarUsuarioRequest"/>
    <wsdl:output message="impl:identificarUsuarioResponse"
      name="identificarUsuarioResponse"/>
  </wsdl:operation>
  <wsdl:operation name="confirmarPagamento">
    <wsdl:input message="impl:confirmarPagamentoRequest"
      name="confirmarPagamentoRequest"/>
    <wsdl:output message="impl:confirmarPagamentoResponse"
      name="confirmarPagamentoResponse"/>
  </wsdl:operation>
  <wsdl:operation name="imprimirPedido">
    <wsdl:input message="impl:imprimirPedidoRequest"
      name="imprimirPedidoRequest"/>
    <wsdl:output message="impl:imprimirPedidoResponse"
      name="imprimirPedidoResponse"/>
  </wsdl:operation>
  <wsdl:operation name="contabilizar">
    <wsdl:input message="impl:contabilizarRequest"
      name="contabilizarRequest"/>
    <wsdl:output message="impl:contabilizarResponse"
      name="contabilizarResponse"/>
  </wsdl:operation>
</wsdl:portType>
<wsdl:service name="CompraService">
  <wsdl:port binding="impl:CompraSoapBinding" name="Compra">
    <wsdl:soap:address
      location="http://localhost:9080/LojaEJBHttpRouter/services/Compra"/>
  </wsdl:port>
</wsdl:service>
</wsdl:definitions>

```

Descrição dos serviços em WSDL

Os serviços são descritos em documentos WSDL, que é o formato padrão de descrição de serviços em Web Services. O serviço "Compra" foi descrito através deste padrão, gerando o documento WSDL da Listagem 2.

Observações sobre a Listagem 2

- Foram suprimidas as seções do documento WSDL que tratam dos tipos, mensagens e binding, visto serem extensivas e pouco relevantes para a abordagem em questão.
- Para manter a compatibilidade com a versão atual do WS-BPEL, foi utilizado a WSDL 1.1. Nesta versão, a tag "portType" correspondente à tag "interface" da WSDL 2.0.

A figura a seguir nos mostra uma das seções importantes de um documento WSDL, a interface do serviço. Ela contém todas as operações por ele suportadas. Note que ordem de execução e quais operações serão executadas no provimento do serviço não são descritas neste tipo de documento. Isto é definido no mecanismo de orquestração, dando ainda mais flexibilidade ao modelo.

Compra			
calcularFrete			
input	caminho		Vector
output	calcularFreteReturn		ArrayOfXSDAnyType
formatarPedido			
input	ordemCompra		Vector
output	formatarPedidoReturn		ArrayOfXSDAnyType
lançarCartaoCredito			
input	ordemCompra		Vector
output	lançarCartaoCreditoReturn		boolean
validarCartao			
input	cartao		int
output	validarCartaoReturn		string
gerarValoresPedido			
input	pedido		Vector
output	gerarValoresPedidoReturn		ArrayOfXSDAnyType
identificarUsuario			
input	emailAddress		string
output	identificarUsuarioReturn		ArrayOfXSDAnyType
confirmarPagamento			
input	pedido		Vector
output	confirmarPagamentoReturn		ArrayOfXSDAnyType
imprimirPedido			
input	pedido		Vector
output	imprimirPedidoReturn		string
contabilizar			
input	ordemCompra		Vector
output	contabilizarReturn		boolean

Figura 6. Interface da WSDL do serviço "Compra".

Implementação da orquestração

Uma vez disponível a descrição do serviço de forma padronizada, o mecanismo de orquestração poderá então mapeá-lo para a provisão. Estando o fluxo do processo em BPMN é fácil gerar o fluxo de orquestração, bastando mapeá-lo diretamente para WS-BPEL. Na configuração do fluxo em WS-BPEL, as atividades (tipo invoke) são ligadas às operações disponíveis na interface do serviço, conforme a sequência de execução prevista. As entradas e saídas das atividades devem encontrar compatibilidade com as mensagens do serviço. Existem inúmeras estratégias para a composição de mensagens. Neste exemplo, simplificamos esta parte e consideramos que as mensagens passadas entre os elementos de fluxo coincidem com a assinatura e o retorno dos métodos dos componentes. Em muitos casos, por inúmeras razões, isto não será a melhor alternativa.

Comparando paradigmas

Comparando as representações dos componentes e dos serviços, podemos perceber um contraste evidente. Enquanto aqueles são modelados segundo a coesão determinada pelas regras de negócio do domínio, estes são modelados pela noção da unidade de troca em uma relação entre o provedor e seus requisitantes. Podemos observar também que o serviço preconiza um fluxo de processo que rege a sequência e as condições sob as quais cada operação é invocada. O mecanismo de orquestração é o encarregado de efetuar esta tarefa. No presente caso, a especificação WS-BPEL dá o suporte a este trabalho, definindo como as inúmeras complexidades da execução deverão ser tratadas. Esta característica do serviço mostra que ele percorre os domínios horizontalmente, de forma perpendicular aos componentes.

Considerações finais

Transformação de modelos

Durante todo o processo de modelagem dos serviços aplicado neste artigo prevaleceram os padrões e a estratégia de transformação de modelos. A passagem de um modelo para outro é viabilizada pela combinação de várias técnicas, as quais tornam possível aplicar as transformações no decorrer de todo o processo de desenvolvimento. A MDA é a principal referência quanto se trata de estratégias baseadas em transformação de modelos. Sem um conjunto bem alinhado de padrões, porém, não se materializa, pois os modelos tendem a apresentar lacunas que interrompem a sequência de transformações. Neste artigo vimos que a evolução dos modelos inicia-se em uma estratégia de modelagem de serviços, quando progressivamente o modelo de negócio vai tomando forma adequada à automação. No momento da passagem de BPMN para WS-BPEL, produziu-se o primeiro artefato de sistemas. Este artefato somente tornou-se útil para um modelo de automação pelo fato de o processo em BPMN ter sido ajustado para esta finalidade. Nesta estratégia de modelagem, estes são os artefatos básicos. A partir destes a transformação continua com o suporte da MDA e das ferramentas, até o último módulo específico de implementação em Java.

A efetividade dos padrões

Um aspecto visível nos exemplos é o fato de que as peças se encaixam naturalmente, sem a necessidade de arranjos e reconceituações. O uso dos padrões, desde a modelagem de negócio, é a âncora para este sucesso. Sem o uso dos padrões, o desenvolvedor depara-se com barreiras a todo o momento, tendo que fazer escolhas que poderão desviá-lo do melhor caminho, consumindo tempo e descaracterizando SOA.

Percepção do baixo acoplamento

Uma das características importantes de SOA é o baixo acoplamento. As atenções para isto se iniciam na estratégia do desenho arquitetônico, culminando nas formas de implementação. O acoplamento é uma das limitações dos modelos monolíticos. Em um modelo orientado a serviço com suporte a processos de negócio, o baixo acoplamento se materializa mais facilmente. O fluxo do processo forma uma rede que abstrai a conexão entre os elementos, mantendo-os livres para serem recombinados, dando maiores possibilidades para absorverem a dinâmica dos negócios. No exemplo do serviço “Compra”, se percebe que qualquer operação poderia ser facilmente removida do fluxo e substituída por uma nova forma de provisão. Se, por exemplo, a empresa

decidir oferecer outras formas de pagamento bastaria substituir a operação atual, apontando para o novo método na interface do componente. Não é difícil imaginar o efeito positivo que um modelo com estas qualidades pode produzir em um ambiente empresarial de grande estatura, em que as interrelações entre as regras de negócio tomam grande complexidade. Neste artigo, tivemos a oportunidade de constatar esta característica na prática.

Materializando a aproximação de negócios com SOA

Os requisitos de negócio foram capturados em forma de fluxo de processo com BPMN. O fluxo foi submetido à metodologia de modelagem e balanceado para que representasse os serviços que o modelador pretendia prover. Este é um primeiro momento de materialização da aproximação entre as visões de negócio e SOA. Em outros momentos, na sequência da aplicação da metodologia, o modelo de negócio foi tomando a forma de automação, mas sem perder suas características essenciais, visto que estavam talhadas no fluxo do processo. O aspecto modular foi aplicado já na formatação dos elementos de negócio, não diretamente sobre os elementos tecnológicos. Os módulos de negócio foram acomodados em vários paradigmas: componentes, operações e serviços e foram implementados de forma a manterem-se livres para as recombinações que o negócio demandar. Nesta estratégia de modelagem de serviços, a aproximação dos negócios com SOA se dá de forma suave e estruturada, materializando-se em artefatos condizentes com a visão das duas perspectivas, evitando que as lacunas perdurem até a implementação.

Resposta final

As maiores dificuldades que temos percebido em relação à SOA é a insuficiência semântica e a falta de uma metodologia apropriada para a modelagem dos serviços. As metodologias convencionais não se mostram efetivas. Inúmeras interpretações são disponíveis atualmente, muitas apenas acomodam forças de mercado sem se mostrarem muito coerentes. Em muitos casos os verdadeiros princípios da orientação para serviços se perdem pelo caminho e o resultado obtido é algo pouco animador. Isto tem provocado frustrações e dúvidas sobre as verdadeiras contribuições que uma Arquitetura Orientada a Serviços poderia dar, produzindo algum descrédito sob seu real valor. Neste artigo, percorremos uma estratégia apoiada em vários padrões. A metodologia de modelagem de serviços com BPMN mostrou no resultado final sua capacidade de inovação, produzindo um modelo preciso, flexível, em total conformidade com as tecnologias modernas e de grande efetividade na convergência das visões de negócios e TI. As iniciativas de mudança dispõem desta opção. **MU**

Referências

- Lazzeri, José Carlos • *Arquitetura Orientada a Serviços – Fundamentos e Estratégias*. Rio de Janeiro, Editora Ciência Moderna Ltda, 2009
- *Business Process Model and Notation (BPMN), Version 1.2* •
OMG Document Number: formal/2009-01-03
Object Management Group (OMG)
- *Web Services Description Language (WSDL), 1.1*
W3C Note 15 March 2001
The World Wide Web Consortium (W3C)
- *Web Services Business Process Execution Language, Version 2.0*
OASIS Standard 11 April 2007
Organization for the Advancement of Structured Information Standards (OASIS)