

6 Trabalhos Relacionados

Apresentamos a seguir uma análise comparativa de trabalhos relacionados ao problema exposto.

6.1. ***Find and Bind***

Karastoyanova [10] propõe a adaptação da composição por meio de um mecanismo denominado '*Find and Bind*', implementado como extensão do WS-BPEL. Este mecanismo envolve:

- Procura por Web Services que satisfaçam ao *portType* especificado no elemento de chamada subsequente (*<invoke>*, *<receive>*, *<reply>*)
- Seleção de um destes Web services;
- Associação da chamada a este.

A procura pelos Web Services é feita no registro de serviços UDDI. A seleção pode ser feita por critérios funcionais ou não-funcionais, ou ainda com base na descrição semântica do serviço.

Esta abordagem visa superar a falta de flexibilidade das implementações do WS-BPEL citadas no artigo, uma vez que nestas as chamadas têm que incluir referências a instâncias específicas.

Nossa proposta é bem semelhante. A seleção com base nos critérios funcionais é feita antes, na camada de aplicação, a partir da definição da composição e dos registros dos fornecedores. Esta seleção resulta em um ou mais Web services a serem chamados, e o número de respostas desejadas é um parâmetro (mais de um é possível por meio de agregação de resultados). Todos os Web services selecionados são acionados, quando então são processados os critérios técnicos. Por meio de abstração e de um bloco de processamento paralelo, a composição é executada para qualquer número de fornecedores, sem necessitar de alteração em seu código ou intervenção do usuário. O SAP XI comporta a adição de novos fornecedores sem necessidade de reiniciar o sistema.

6.2. A-ActiveBPEL

A fim de tornar as chamadas da composição adaptativas, Lins [11] realizou uma modificação da *engine open source* ActiveBPEL, com a introdução das classes *adaptiveInvoke* e *informWS*. De maneira semelhante ao trabalho anterior, estas classes são executadas antes da chamada ao Web Service propriamente dita, selecionando Web services candidatos para execução no repositório. O Web service é escolhido com base em um atributo associado a ele no registro, que pode ser definido livremente.

Devido ao fraco acoplamento, esta implementação permite adicionar ou remover serviços ao repositório com o sistema ativo, modificando as chamadas sem necessidade de reinício. Por modificar a diretiva de baixo nível, é transparente para o WS-BPEL, mas afeta todas as chamadas indiscriminadamente.

A arquitetura que apresentamos não depende de modificações nos componentes *standard*, e também apresenta fraco acoplamento por utilizar mensagens entre a aplicação e o mediador. Nossa aplicação foi preparada com rotinas de recuperação em caso de erros. Se os Web services escolhidos não retornarem respostas válidas, o sistema automaticamente tentará um novo conjunto, até que a operação seja bem-sucedida ou não exista mais nenhum fornecedor que satisfaça aos critérios da composição.

6.3. Composição Dinâmica baseada em Padrões de Projeto

Martins [12] aponta duas características importantes para uma boa composição de serviços:

- Receber notificação de novos serviços publicados no diretório;
- Escolher um dentre vários serviços disponíveis para composição.

Em seu trabalho, Martins observou que estas características podem ser obtidas pelos padrões de projeto *Observer* e *Strategy*²/*MVC*. Em lugar de uma consulta única ao repositório, a aplicação recebe notificações sobre a atualização deste continuamente.

² também conhecido como *Policy*

Um agente efetua coleta de informações sobre os serviços por meio de chamadas de teste.

Para maior autonomia, foi adotado um mecanismo baseado em ontologias para descoberta de serviços. Assim, o usuário precisa especificar apenas a qual ontologia o serviço a ser usado deve pertencer.

A adaptação é feita pelo agente, em tempo de execução.

Comparativamente, o trabalho de Martins é o que apresenta a solução mais geral. A fase de descoberta baseia-se em ontologia, caracterizando uma composição semântica de Web services. O uso desta tecnologia provê um alto grau de abstração e flexibilidade na definição da composição dos serviços, conforme vêm demonstrando os trabalhos do SWC³. A fase de seleção é baseada em estatísticas de tempo de resposta, mas pode ser adaptada para outros critérios. Por fim, a chamada a um serviço deixa de ocorrer quando este se torna lento, em favor de outros serviços semanticamente equivalentes.

Nossa proposta de mediação flexível endereça as mesmas questões, mas com uma abordagem diferente. Em lugar de ontologias formais, utilizamos listas de atributos na composição. Ambas as propostas efetuam a seleção de serviços equivalentes, dados os critérios da composição. Em nossa proposta não há coleta de tempos de processamento, mas neste ponto fomos além: todos os serviços possíveis são acionados, e apenas os primeiros a responder são considerados. Naturalmente, este mecanismo funcionará bem para consultas, mas não para transações. Porém, o fato dele ser aplicável em todos os estudos de caso apresentados nos trabalhos citados mostra o seu valor. Com relação à chamada dos serviços, criamos um registro que suspende os fornecedores com Web services lentos por um tempo pré-definido, e automaticamente tenta um novo conjunto de fornecedores. Nenhuma requisição do usuário é perdida devido a este esquema, que não figura em nenhum dos trabalhos analisados.

³ <http://challenge.semanticweb.org/>

6.4. Comparativo das arquiteturas apresentadas

	Find and Bind	A-ActiveBPEL	Composição Dinâmica	Mediação flexível
Registro	UDDI	Arquivo XML	Diretório Ativo	Repositório SAP XI
Consulta ao registro	A cada execução	A cada execução	A cada modificação	A cada execução
Descoberta de serviços	Baseada no <i>portType</i>	Baseada em atributo fixo no registro	Baseado em ontologia	Baseada em atributo fixo e no cadastro de fornecedores
Seleção	Implementação em aberto	Primeiro registro	Histórico de tempo de resposta	Histórico de consultas ou pedidos, menor tempo de resposta
Seleciona mais de um serviço	Não	Não	Não	Sim, agrega resultados
Em caso de indisponibilidade do serviço selecionado	Retorna erro	Retorna erro	Retorna erro	Suspende novas chamadas temporariamente e seleciona o próximo
Em caso de lentidão do serviço selecionado	Não trata	Não trata	Reflexo na média de tempo de resposta	Só considera as n primeiras respostas

Tabela 4 - Comparativo de trabalhos com arquiteturas relacionadas

6.5. *Contracting Interface*

A fase 3 do *Semantic Web Services Challenge* inclui um cenário de descoberta de serviços, onde um fornecedor deve ser selecionado com base em suas anotações semânticas. Esta descoberta possivelmente envolveria várias interações com determinado serviço, dependendo do grau de especificidade do requerimento de compra. Por exemplo, além de indicar o produto a ser encontrado, o requerimento pode citar um preço máximo ou data limite para entrega.

Neste contexto, a equipe do DERI [13] apresentou uma proposta baseada na criação de uma *contracting interface* para os Web Services. Por meio dela, um mediador pode solicitar informações mais específicas sobre determinado serviço ou produto, bem como suas condições de fornecimento. Este mecanismo

habilita ao mediador obter as informações de que precisa para seleção por meio da troca de mensagens com o fornecedor, sem a necessidade de carregar previamente todas as informações necessárias, e sem intervenção humana.

Esta abordagem depende da definição de uma ontologia comum, mas é bastante promissora num cenário onde o número de fornecedores e produtos é elevado.

Nossa proposta de mediação flexível depende de intervenção humana com relação a critérios que não foram previstos e cadastrados originalmente. Por exemplo, se o solicitante deseja adquirir um triturador de papel automático, ele provavelmente não terá à sua disposição uma composição dedicada este tipo de produto. Mais ainda, ele pode ter como requisito a marca e a voltagem do equipamento. É possível, porém impraticável, que os atributos de fornecedores cadastrados incluam tal informação. Com a adoção do mecanismo proposto pelo DERI, após algumas interações, o mediador conseguiria realizar esta seleção.