

CENTRO UNIVERSITÁRIO FEEVALE

MARCO AURÉLIO BRAUN

MODELO DE INTEGRAÇÃO ENTRE DIFERENTES SOFTWARES
APLICATIVOS POR UM BARRAMENTO DE SERVIÇOS
CORPORATIVOS BASEADO EM UMA ARQUITETURA
ORIENTADA A SERVIÇOS

(Título Provisório)

Anteprojeto de Trabalho de Conclusão

Novo Hamburgo, abril de 2009.

MARCO AURÉLIO BRAUN

marco.aurelio.braun@gmail.com

MODELO DE INTEGRAÇÃO ENTRE DIFERENTES SOFTWARES
APLICATIVOS POR UM BARRAMENTO DE SERVIÇOS
CORPORATIVOS BASEADO EM UMA ARQUITETURA
ORIENTADA A SERVIÇOS

(Título Provisório)

Centro Universitário Feevale
Instituto de Ciências Exatas e Tecnológicas
Curso de Sistemas de Informação
Anteprojeto de Trabalho de Conclusão

Professor orientador: Alexandre de Oliveira Zamberlam

Novo Hamburgo, abril de 2009.

RESUMO

Durante anos, as empresas desenvolvedoras de software geraram uma incontável quantidade de sistemas, baseados nas mais diversas linguagens de programação e arquiteturas. Esses sistemas são o resultado da demanda gerada a partir da evolução dos negócios corporativos. Muitas empresas ainda mantêm sistemas antigos em pleno funcionamento, convivendo com softwares aplicativos¹ baseados em tecnologias mais modernas. Uma alternativa que pode ser aplicada na integração de todos esses softwares é a *Arquitetura Orientada a Serviços - Service Oriented Architecture* (SOA). SOA combina regras de negócios entre aplicações através de serviços intercambiáveis. Um componente ou ferramenta integrante de SOA é o *Barramento Corporativo de Serviços – Enterprise Service Bus* (ESB). ESB tem a capacidade de fazer com que sistemas heterogêneos comuniquem-se de forma transparente. Sendo assim, este trabalho tem o objetivo de estudar, definir o funcionamento e gerar um modelo de um ESB. Dessa forma, servir de referência às empresas desenvolvedoras de software, promovendo o reuso de componentes (linhas de código) e combinando recursos já existentes, no intuito de dar mais agilidade às mesmas, com objetivo de que atinjam uma vantagem estratégica.

Palavras-chave: SOA. ESB. Integração de Sistemas. Arquitetura de Software.

¹ Software Aplicativo é um conjunto de comandos, instruções ou ordens elaborado pelo cliente e/ou usuário para o computador cumprir, visando resolver problemas e desenvolver atividades ou tarefas específicas (FURTADO, 2002, p.29).

SUMÁRIO

MOTIVAÇÃO.....	5
OBJETIVOS.....	8
METODOLOGIA.....	9
CRONOGRAMA	11
BIBLIOGRAFIA	13

MOTIVAÇÃO

Nos últimos anos, a tecnologia da informação sofreu fortes e rápidas mudanças. As empresas desenvolvedoras de software geraram uma infinidade de sistemas, a partir das mais variadas tecnologias (linguagens de programação, arquiteturas...). “O atual panorama corporativo é composto de sistemas que são o resultado da evolução dos negócios e da tecnologia com o passar dos anos.” (CUMMINS, 2002, p.2).

Os sistemas legados² utilizam-se das mais variadas formas de integração, com o uso de tecnologias diferentes. Esse cenário é assim descrito:

“Após muitos anos de desenvolvimento de aplicações de computador em um panorama tecnológico em evolução, a maioria das empresas está carregada de computadores e aplicações que estão interligados através de uma variedade de mecanismos específicos.” (CUMMINS, 2002, p.XIII).

Essa diversidade acaba por determinar um forte acoplamento³ entre as aplicações, com uma estrutura muito rígida de comunicação. Para cada interligação entre aplicações, existe uma tecnologia, forma ou meio diferentes. Isso acaba por tornar-se de difícil manutenção e depende de um despendimento desnecessário de recursos.

“Os grandes sistemas utilizam plataformas diferentes, linguagens de programação diferentes (e paradigmas de programação) e até middleware⁴ diferente. Eles são uma bagunça de mainframes, clientes SAP, bancos de dados, aplicações J2EE, pequenos motores de regras, e assim por diante. Em outras palavras, eles são heterogêneos.” (JOSUTTIS, 2008, p. 13).

Além dos sistemas legados, desenvolvidos por uma mesma empresa, há também os casos em que empresas diferentes têm a necessidade de realizar a integração entre seus aplicativos, para atingir um objetivo em comum, que é a satisfação da necessidade de um cliente em comum. “SOA inclui práticas e processos que são baseados no fato de que redes de

² Sistemas que estão em plataformas de software antigas, mas continuam em funcionamento

³ Alta dependência entre softwares aplicativos integrados

⁴ Programa de computador que faz a mediação entre outros softwares

sistemas distribuídos não são controlados por um único proprietário. Equipes diferentes, departamentos diferentes ou até empresas diferentes podem gerenciar os sistemas diferentes.” (JOSUTTIS, 2008, p. 13).

Desta forma, uma empresa escreve seus softwares aplicativos alicerçados em uma determinada plataforma e em uma linguagem de programação específica, além de manter os dados dos mesmos em um banco de dados em particular. Na maioria dos casos, essas características não são reproduzidas em um todo pela outra empresa. Mesmo que haja semelhanças de plataforma, linguagem ou banco de dados, a maneira de como esses dados estão estruturados são diferentes. Inclusive, existem cadastros que são repetidos em dois ou mais aplicativos, não podendo assim ser compartilhados.

Uma alternativa que pode ser estudada é a *Arquitetura Orientada a Serviços - Service Oriented Architecture (SOA)*. SOA pode ser conceituada como “um estilo de arquitetura de sistemas de informação que habilita a criação de aplicações que são construídas com a combinação de acoplamento leve com serviços intercambiáveis.” (SANTOS, 2007, p. 3).

Ainda de acordo com Santos (2007), “estes serviços conversam entre si, baseados em uma definição formal, que é independente da plataforma e da linguagem de programação.”. Ou seja, SOA permite a integração entre sistemas heterogêneos. Segundo Josuttis (2008), “a abordagem SOA aceita a heterogeneidade”. Ele continua, afirmando que “essa é uma das idéias-chave de SOA e ela pode dar à SOA o poder para iniciar uma revolução”.

Um dos objetivos de SOA é atingir um acoplamento fraco entre os aplicativos. Conforme em (JOSUTTIS, 2008),

“O acoplamento fraco é o conceito de minimizar as dependências. Quando as dependências estão minimizadas, as modificações têm os efeitos minimizados e os sistemas ainda executam quando partes deles estão quebradas ou indisponíveis. Minimizar as dependências contribui para tolerância às falhas e flexibilidade, que é exatamente dos que nós precisamos”.

Josuttis (2008) afirma que SOA é baseada em três conceitos técnicos principais: serviços, interoperabilidade através de um barramento corporativo de serviços e acoplamento fraco.

Uma das definições para *Barramento Corporativo de Serviços – Enterprise Service Bus (ESB)*, é

“Um barramento corporativo de serviços (ESB) é a infra-estrutura que possibilita a alta interoperabilidade entre os sistemas distribuídos para serviços. Ele torna mais

fácil distribuir processos corporativos através de múltiplos sistemas utilizando diferentes plataformas e tecnologias.” (JOSUTTIS, 2008, p. 7).

Também em (JOSUTTIS, 2008), é enaltecida a importância contemporânea do ESB, quando é escrito que

“Embora a idéia de um barramento de TI seja bastante antiga, recentemente tem havido um renascimento desse conceito. Isso se iniciou com a introdução do *barramento corporativo de integração das aplicações* (barramento EAI), o que posteriormente foi substituído pelo *barramento corporativo de serviços* (ESB)”.

Finalmente, pretende-se com este trabalho, contribuir na forma de material teórico e prático sobre os temas abordados, que são de um referencial bibliográfico um tanto escasso. No meio acadêmico o assunto é novo e pode ser uma desafiadora fonte de pesquisa. Além do que, este estudo pode ser de relevante valia às empresas desenvolvedoras de software, ajudando a minimizar os investimentos e esforços para realizar a integração entre softwares legados. Pode ainda contribuir na promoção do reuso e compartilhamento de componentes, combinando recursos de maneira rápida, dando uma resposta ágil a uma exigência de mercado, possibilitando a elas a obtenção de uma vantagem estratégica. Assim, dentro desse contexto apresentado, têm-se como exemplo, sistemas de folha de pagamento e controle de frequência de funcionários (ponto eletrônico).

OBJETIVOS

Objetivo Geral

Projetar um software (*middleware*) de Barramento de Serviços Corporativos, baseado em uma Arquitetura Orientada a Serviços para integração dos sistemas de folha de pagamento e controle de frequência de funcionários (ponto eletrônico).

Objetivos específicos

- Apresentar as formas e métodos de integração de softwares aplicativos;
- Descrever a Arquitetura Orientada a Serviços;
- Detalhar a arquitetura de um Barramento de Serviços Corporativos;
- Contribuir, dessa forma, com material teórico e prático sobre esses temas;
- Relatar estudos e/ou trabalhos correlatos que tiveram sucesso;
- Projetar um modelo de integração de softwares aplicativos, visando contribuir com uma empresa desenvolvedora de software, para a solução dos problemas relativos à integração de seus aplicativos e com softwares de terceiros.

METODOLOGIA

A fim de se atingir com êxito os objetivos propostos, devem ser executadas as atividades relacionadas a seguir. Essas atividades estão divididas em duas etapas distintas.

1. Elaboração do Trabalho de Conclusão I

1.1. Pesquisa de referências bibliográficas para embasamento deste anteprojeto, bem como de todo o trabalho de conclusão. Utilizar-se-á como fonte de pesquisa livros, dissertações, monografias, artigos e eventualmente conteúdos de sites da internet, desde que de fontes confiáveis;

1.2. Preparação, escrita e entrega do anteprojeto;

1.3. Redação do referencial teórico, a partir de pesquisas e estudos sobre:

1.3.1. A Arquitetura Orientada a Serviços (SOA);

1.3.2. Barramento de Serviços Corporativos (ESB);

1.3.3. Formas alternativas de integração de softwares aplicativos;

1.3.4. Trabalhos Correlatos;

1.4. Via *Unified Modeling Language* (UML) – Linguagem de Modelagem Unificada, realizar o detalhamento inicial da empresa desenvolvedora de software, a fim de compreender o processo utilizado atualmente na integração;

1.5. Redação e entrega do TC I;

2. Elaboração do Trabalho de Conclusão II

2.1. Via *Unified Modeling Language* (UML) – Linguagem de Modelagem Unificada, realizar a:

2.1.1.Finalização do detalhamento da área/problema de uma empresa desenvolvedora de software, a fim de compreender o processo utilizado atualmente na integração de sistemas;

2.1.2.Descrição da arquitetura do processo atual de integração;

2.1.3.Modelagem e descrição da nova arquitetura proposta para o processo de integração (aspectos estruturais e funcionais);

2.2. Avaliação do modelo proposto;

2.3. Redação e entrega do Trabalho de Conclusão II.

CRONOGRAMA

Trabalho de Conclusão I

Etapa	Meses				
	Mar	Abr	Mai	Jun	Jul
Pesquisa de referências bibliográficas					
Redação e entrega do Anteprojeto					
Revisão bibliográfica sobre SOA					
Revisão bibliográfica sobre ESB					
Revisão bibliográfica sobre formas de integração de softwares aplicativos					
Pesquisa sobre trabalhos correlatos					
Instalação e testes de ferramentas de modelagem UML					
Detalhamento inicial da área/problema de uma empresa desenvolvedora de software					
Redação e entrega do TC I					

Trabalho de Conclusão II

Etapa	Meses			
	Ago	Set	Out	Nov
Finalização do detalhamento da área/problema de uma empresa desenvolvedora de software				
Descrição da arquitetura do processo atual de integração				
Modelagem e descrição da nova arquitetura proposta				
Modelagem de aspectos estruturais: Mapeamento Objeto Relacional – Diagramas de Classes e Pacotes; Modelo ER				
Modelagem de aspectos funcionais: Diagramas de Casos de Uso e Casos de Uso Estendidos				
Modelagem de aspectos funcionais: Diagramas de Atividades para complementar os fluxos dos Casos de Uso Estendidos				
Modelagem de aspectos funcionais: Diagramas de Colaboração/Sequência para complementar os métodos (serviços) nos Diagramas de Classes				
Avaliação do modelo proposto, através do desenvolvimento de um protótipo				
Redação e entrega do TC II				

BIBLIOGRAFIA

BENEDETE JUNIOR, Antonio Carlos. **Roteiro para a definição de uma arquitetura SOA utilizando BPM**. Monografia apresentada à Escola Politécnica da Universidade de São Paulo para obtenção do Título de MBA em Tecnologia da Informação. São Paulo, 2007.

CHAPPELL, David. **Enterprise Service Bus – Theory in Practice**. O’Reilly Media, 2004.

CUMMINS, Fred A. **Integração de sistemas: EAI (Enterprise application integration): arquiteturas para integração de sistemas e aplicações corporativas**. Rio de Janeiro: Campus, 2002.

FURTADO, Vasco. **Tecnologia e Gestão da Informação na Segurança Pública**. Rio de Janeiro: Garamond, 2002.

JOSUTTIS, Nicolai M. **SOA na Prática. A Arte da Modelagem de Sistemas Distribuídos**. Rio de Janeiro: Alta Books, 2008.

PARDAL, Miguel Filipe Leitão. **Segurança de aplicações empresariais em arquiteturas de serviços**. Dissertação para obtenção do Grau de Mestre em Engenharia Informática e de Computadores. Universidade Técnica de Lisboa. Instituto Superior Técnico. Lisboa, Portugal: 2006.

SANTOS, Alfredo Luiz dos. **Integração de Sistemas com Java**. Rio de Janeiro: Brasport, 2007.