



Universidade Federal de Pernambuco
Centro de Informática
Graduação em Ciência da Computação

Proposta Trabalho de Graduação

**“Um Mecanismo de Monitoramento e Seleção de
Serviços Baseado em Atributos de Qualidade para uma
Arquitetura Orientada a Serviços”**

*Proposta inicial apresentada ao Programa de Graduação
em Ciência da Computação do Centro de Informática da
Universidade Federal de Pernambuco como requisito par-
cial para obtenção do grau de Bacharel em Ciência da
Computação.*

Orientador: Nelson Souto Rosa (nsr@cin.ufpe.br)

Aluno: Tarcisio Coutinho da Silva (tcs5@cin.ufpe.br)

Recife, Setembro/2011

Sumário

1 Proposta	2
1.1 Contexto	2
1.2 Objetivos	4
2 Cronograma	5
2.1 Programação das Atividades	5
Referencias Bibliográficas	6

1

Proposta

1.1 Contexto

Arquitetura Orientada a Serviços, ou do inglês *Service-Oriented Architecture* (SOA), tem surgido ao longo dos últimos anos como uma das abordagens preferidas para construção de sistemas [1]. Com objetivos alinhados aos objetivos do paradigma SOC (*Service-Oriented Computing*), SOA possibilita a criação de novas aplicações com maior coerência, rapidez e diminuição nos custos, tudo isso com excelente aproveitamento do legado [1].

Baseada em padrões abertos e na visão onipresente da Internet, SOA tem como princípio fundamental a idéia de serviços como unidades que representam módulos do negócio ou funcionalidades da aplicação [1] [2]. Na visão de SOA, um serviço é um componente que implementa uma função de negócio. Ele pode responder a requisições ocultando os detalhes de sua implementação e é descrito através de contratos que expressem seu objetivo e suas capacidades. Buscando maior agilidade, flexibilidade e reuso de componentes pré-existentes no desenvolvimento de aplicações, diversas empresas de tecnologia da informação vêm adotando SOA na construção de seus sistemas.

Nesse contexto, temos a plataforma de serviços OSGi (*Open Services Gateway initiative* [3]). OSGi provê suporte à modelagem e desenvolvimento de sistemas modulares na linguagem Java [4]. A idéia básica é resolver o problema de criar softwares monolíticos, ou seja, softwares projetados sem modularidade, facilitando a reutilização e manutenção de componentes, o que torna a solução mais robusta, barata e confiável [5].

OSGi introduz um modelo de programação orientada a serviço, o que alguns autores apresentam como *SOA in a Virtual Machine* [4], separando de fato a interface da implementação. Isso mostra um grande potencial para a construção de aplicações orientadas à serviço.

Outra tecnologia que compartilha vários princípios de SOC é a tecnologia de *Web*

Services. *Web Services* fornecem uma forma padrão de interoperabilidade entre diferentes aplicações, capaz de executar em uma variedade de plataformas e *frameworks*. [6]

Assim, em uma arquitetura orientada a serviços, módulos podem consumir serviços providos por tecnologias diferentes, apenas identificando o serviço requerido e realizando o *binding*. Estabelecendo assim um contrato entre o consumidor e o provedor do serviço [7].

Porém, uma tarefa complexa neste cenário é garantir que os contratos a nível de serviço entre o consumidor e provedor são realmente respeitados. Além disto, quando for caracterizada uma quebra de contrato, onde o provedor não cumpre com suas responsabilidades, pode ser necessário selecionar dinamicamente um serviço equivalente que realmente cumpra com os requisitos desejados.

Neste contexto, o monitoramento dos serviços aliado à análise dos dados monitorados, surge como uma possível solução para este problema de garantia de cumprimento de contrato entre provedores e consumidores de serviços.

1.2 Objetivos

O objetivo deste trabalho de graduação é desenvolver um mecanismo de monitoramento e seleção de serviços, capaz de capturar e armazenar dados relativos as métricas de qualidade de serviço (QoS) monitorados. A partir da obtenção destes dados, realizar a seleção de serviço requeridos.

As métricas de QoS monitoradas são requeridas pelo consumidor do serviço, através da definição de contratos de serviço junto ao provedor. A partir da análise desses contratos em relação aos dados obtidos do monitoramento, o provedor mais adequado será escolhido dentre os possíveis candidatos.

2

Cronograma

2.1 Programação das Atividades

Atividade	Setembro	Outubro	Novembro	Dezembro
Estudar SOA, OSGi e iPojo	×	×		×
Implementar Interceptadores	×	×		
Estudar DOSGi		×		×
Implementar Seleção de Serviços		×	×	
Elaborar Monografia			×	×

Tabela 2.1 Tabela das Atividades Programadas

Referências Bibliográficas

- [1] T. Erl and Safari Tech Books Online (Online service). *Soa: principles of service design*. Prentice Hall, 2008.
- [2] Boris Lublinsky. Defining soa as an architectural style, 2007. <http://www.ibm.com/developerworks/library/ar-soastyle>. Último acesso em Setembro/2010.
- [3] O.S.G. Alliance. Osgi service platform, core specification, release 4, version 4.1. *OSGi Specification*, 2007.
- [4] R.S. Hall and D. Savage. *Osgi in action*, 2010.
- [5] J. Davis and Safari Tech Books Online. *Open source SOA*, volume 8. Manning, 2009.
- [6] W3C. Web services activity, 2002. <http://www.w3.org/2002/ws/>. Último acesso em Setembro/2010.
- [7] Qusay H. Mahmoud. *Service-oriented architecture (soa) and web services: The road to enterprise application integration (eai)*, 2005. <http://www.oracle.com/technetwork/articles/javase/soa-142870.html>. Último acesso em Setembro/2010.