

# A Utilização da API SOFA para o Desenvolvimento de uma Aplicação de Web Semântica: Um Estudo de Caso Envolvendo as Ontologias Estado, Região e Cidade

Jordan Pauleski Juliani<sup>1</sup>, Raphael Winckler de Bettio<sup>2</sup>, Luis Henrique Bogo<sup>3</sup>, Alejandro Martins Rodrigues<sup>4</sup>, Heleno Fulber<sup>5</sup>

<sup>1,2,3,4,5</sup> Programa de Pós-Graduação em Engenharia e Gestão do Conhecimento, Universidade Federal de Santa Catarina, Trindade, Florianópolis, SC, Brasil  
{jordan, bogo, raphael, hfulber}@egc.ufsc.br, martins@vias.org.br

**Resumo** - Um dos principais problemas da web atual é a recuperação da informação contida nela. Isso acontece porque essa informação não está estruturada nem segue nenhum padrão, sendo que a única preocupação é estética, ou seja, que essa informação esteja organizada em páginas feitas para humanos entenderem. Para dar significado as páginas surgiu a web semântica, que através das ontologias consegue mapear o conhecimento embutido nos documentos. Este artigo demonstra a aplicação dos conceitos da web semântica através do uso da tecnologia SOFA (Simple Ontology Framework API) no âmbito de estado, região e cidade.

Palavras-chave: Web Semântica, Ontologias, SOFA.

## API SOFA Use to develop an Application of Semantic Web: A Case Study Including the Ontology State, Region and City.

**Abstract** - One of the main problems of the current web is the recovery of the information contained in. This happens because this information is not structuralized and no follows one standard, being that the only concern is aesthetic, or either, that this information is organized in pages made to human to understand. To give meant to the pages appeared semantic web, that through the ontologias obtains to chart the knowledge inlaid in documents. This article demonstrates a application of the concepts of semantic web through the use of technology SOFA (Simple Ontology Framework API) in the scope of state, region and city.

Key-words: Semantic Web, Ontologies, SOFA.

Received June 27, 2005 / Accepted August 12, 2005

### Introdução

A Web Semântica é uma extensão da Internet atual onde a informação é distribuída com significado bem definido, possibilitando com que computadores e pessoas trabalhem de maneira mais cooperada. Neste sentido, uma série de pesquisadores e parceiros industriais buscam a definição de padrões e tecnologias através dos quais os dados disponíveis na Internet possam ser definidos e "linkados" de maneira a serem usados para a descoberta, automação, integração e reuso através de aplicações de modo mais efetivo. Por meio desse desenvolvimento a web

alcançará todo o seu potencial, oferecendo um ambiente onde dados podem ser compartilhados e processados a partir de ferramentas automáticas [1] [2].

Um exemplo da aplicação da web semântica pode ser apresentado através de uma situação onde se deseja comparar preços e escolher de vaso de flores que cresçam melhor na sua região, ou procurar por peças de reposição de um automóvel Volvo 740. A informação desestruturada que deve responder a estas perguntas devem certamente estar na web, mas não de forma utilizável pelos computadores, no sentido de que ainda necessita-se de uma

pessoa para discernir o significado da informação e a sua relevância de acordo com o que se deseja.

O presente trabalho apresenta uma aplicação dos conceitos da web semântica através do uso da tecnologia SOFA (Simple Ontology Framework API)<sup>1</sup>, que corresponde a uma API (*application program interface*) desenvolvida através da linguagem java com o foco de oferecer uma linguagem de especificação de conhecimento baseada em ontologias para a recuperação de informação. A aplicação desenvolvida abrange os conceitos de estado, região e cidade, por meio dos quais os usuários podem recuperar informações de uma cidade, por exemplo, através dos relacionamentos estabelecidos entre os estados, regiões e cidades cadastradas de acordo com a ontologia definida, a partir de um processo de inferência implementado pela API utilizada. Para tanto, são apresentados o conceito da Web Semântica, a API SOFA e o estudo e caso.

### Web Semântica

A web semântica corresponde a um conjunto de informações relacionadas entre si, de forma a facilitar o seu processamento pelos computadores, em uma escala global. Pode-se entender a web semântica como uma maneira eficiente de representar dados na Internet ou em um banco de dados relacionados globalmente [3].

A web semântica objetiva definir o significado da informação contida em documentos da internet de forma a possibilitar com que um computador possa, através de agentes de software, ter acesso às informações mais relevantes desses documentos e seja capaz de "entender" seu significado, podendo processá-las de maneira automática [4]. A especificação das informações, nesse contexto, é realizada através de marcações semânticas, onde os dados relevantes são classificados de um com um vocabulário de termos com significado definido, bem como são especificadas as relações entre esses dados.

O gráfico a seguir apresenta a arquitetura da web semântica destacando cada um dos seus componentes.

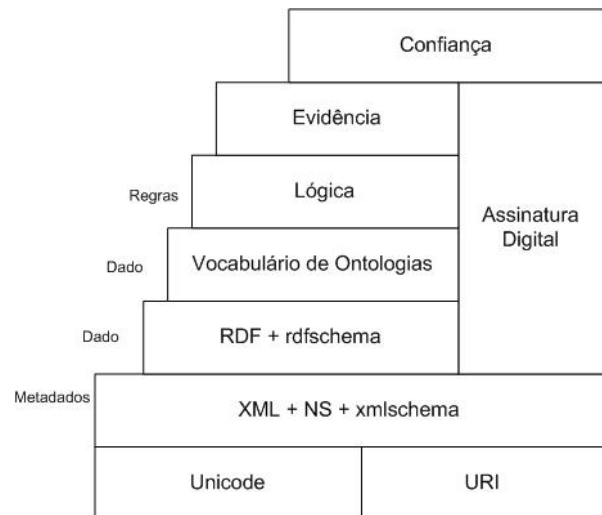


Figura 1: Arquitetura da web semântica

- Unicode e URI: Respectivamente, define os símbolos que podem ser usados para descrever metadados e o modo identificar unicamente recursos na Web
- XML + NS + XMLSchema: Fundamentam a representação e dados e definição dos seus tipos.
- RDF + RDFSchema: Possibilita definir a estrutura dos metadados.
- Ontology Vocabulary: Permite estabelecer as relações entre os metadados. Destaca-se que nessa camada a linguagem OWL (*Web Ontology Language*), desenvolvida para a descrição de ontologias e baseada na linguagem RDF.
- Logic: Possibilita definir regras para o tratamento da informação descrita nas camadas anteriores.
- Proof: Verifica a consistência dos dados acessíveis na Web Semântica.
- Digital Signature: garante a procedência de um documento ou definição.
- Trust: corresponde a última camada que visa permitir a criação de agentes de software que trabalhem apenas com dados confiáveis na Web Semântica.

### SOFA

SOFA (Simple Ontology Framework API), é uma API construída em java com o propósito de proporcionar uma linguagem de especificação de conhecimento baseada em ontologias. Pode ser aplicada em projetos de web semântica, recuperação de informação, bases de conhecimento etc. SOFA providencia um modelo abstrato e simples independente da especificação de representação utilizada pela ontologia, o que possibilita a utilização de diversas linguagens tanto isoladas quanto em conjunto.

A API SOFA providencia as seguintes funcionalidades:

<sup>1</sup> Informações sobre a API disponíveis em: [sofa.projects.semwebcentral.org/dev.html](http://sofa.projects.semwebcentral.org/dev.html)

- Um modelo de objeto abstrato para representação de ontologias;
- Um mecanismo de inferência genérico;
- Ferramentas para gravação de ontologias para diversos modelos;
- Mecanismo para interoperabilidade de ontologias;
- Ferramentas para representação externa de ontologias (diversas linguagens);
- Mecanismo para validação de ontologias;
- Ferramentas para busca de ontologias.
- Entre outras, as principais vantagens da utilização da API são:
- Redução do tempo e custo de desenvolvimento por tratar-se de uma implementação das tarefas mais comuns relativas ao processamento de ontologias.
- Aumento da flexibilidade do software desenvolvido, pois trata-se de um modelo abstrato de tratar com ontologias.

## Estrutura da API

### Ontology Object Model Interfaces

São representados por uma série de classes java capazes de representar uma ontologia e proporcionar as tarefas quanto a utilização das mesmas como: criação de instâncias, criação de conceitos, inferência, etc.

### Serialization modules

Possibilita a serialização, ou gravação em disco, da ontologia criada utilizando-se como padrão diversas linguagens de armazenamento como: OWL, DAML+OIL, RDF+RDF-Schema ou N-Tripes, conforme apresentado no esquema abaixo.

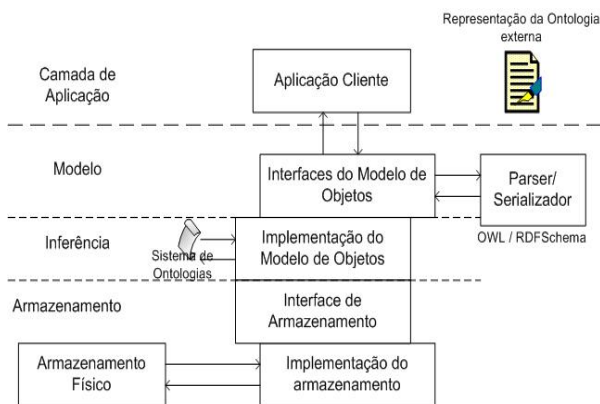


Figura 2: Estrutura da API

## Estudo de Caso

O objetivo do caso de uso implementação é possibilitar uma maior explicitação das capacidades da API SOFA.

Para atingir esse objetivo, a ontologia abaixo representada foi implementada seguindo os padrões da API e como resultado final um conjunto de resultados foi inferido.

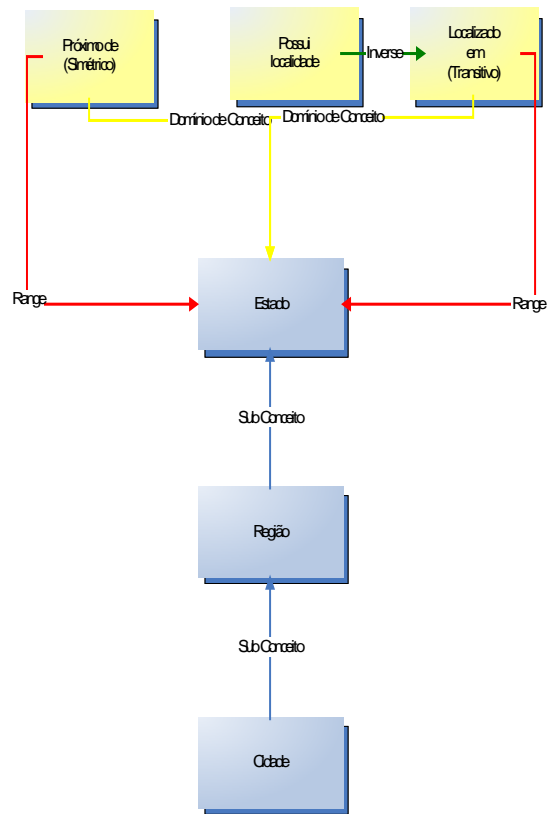


Figura 3: Ontologia implementada

O Objetivo da ontologia é representar de uma maneira simplificada a organização geográfica de um estado.

A representação da ontologia foi efetuada de duas maneiras, sendo a primeira utilizando-se o próprio modelo abstrato disponível e o segundo utilizando-se OWL.

Representação utilizando-se o Modelo Abstrato:

```

Concept          estado          =
onto.createConcept("Estado");
Concept          regioao         =
estado.createSubConcept("Regiao");
Concept          cidade         =
regiao.createSubConcept("Cidade");

Relation          localizadoem    =
estado.createRelation("LocalizadoEm");
localizadoem.addRange(estado);
localizadoem.setTransitive(true);

Relation          possuilocalidade =
regiao.createRelation("PossuiLocalidade");

```

```

    possuiLocalidade.setInverseOf(localizadoem,
true);

    Relation      proximode      =
regiao.createRelation("ProximoDe");
    proximode.addRange(estado);
    proximode.setSymmetric(true);

```

Representação utilizando-se OWL

```

<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF      xml:base="http://geografia.com/onto#"
xmlns="http://geografia.com/onto#"
xmlns:owl="http://www.w3.org/2002/07/owl#"
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-
syntax-ns#"
xmlns:rdfs="http://www.w3.org/2000/01/rdf-
schema#">
  <owl:Ontology/>
  <owl:TransitiveProperty rdf:ID="LocalizadoEm">
    <rdfs:domain rdf:resource="#Estado"/>
    <rdfs:range rdf:resource="#Estado"/>
  </owl:TransitiveProperty>
  <owl:Class rdf:ID="Regiao">
    <rdfs:subClassOf rdf:resource="#Estado"/>
  </owl:Class>
  <owl:SymmetricProperty rdf:ID="ProximoDe">
    <rdfs:domain rdf:resource="#Regiao"/>
    <rdfs:range rdf:resource="#Estado"/>
  </owl:SymmetricProperty>
  <owl:Class rdf:ID="Estado"/>
  <owl:ObjectProperty rdf:ID="PossuiLocalidade">
    <rdfs:domain rdf:resource="#Regiao"/>
    <owl:inverseOf
rdf:resource="#LocalizadoEm"/>
  </owl:ObjectProperty>
  <owl:Class rdf:ID="Cidade">
    <rdfs:subClassOf rdf:resource="#Regiao"/>
  </owl:Class>
</rdf:RDF>

```

Independentemente da representação da ontologia as instâncias foram criadas seguindo as seguintes informações:

- Santa Catarina é um Estado;
- Santa Catarina possui duas Regiões sendo elas, Litoral e Interior;
- Xanxerê é uma cidade e está localizada na região Interior;
- Chapecó é uma cidade e está localizada na região Interior e é próxima de Xanxerê;
- Florianópolis é uma cidade e está localizada na região Litoral.

A representação das instâncias foi feita utilizando-se apenas o modelo abstrato, mas poderia ser serializada utilizando-se o OWL, utilizando-se para isto os recursos da própria API.

Criando as Instancias no modelo abstrato

```

Thing      sc      =
estado.createInstance("SantaCatarina");
Thing interior = regiao.createInstance("Interior");
Thing litoral = regiao.createInstance("Litoral");

```

```

interior.set(localizadoem,sc);
litoral.set(localizadoem,sc);

```

```

Thing      xanxere      =
cidade.createInstance("Xanxere");
Thing      chapeco      =
cidade.createInstance("Chapeco");
Thing      floripa      =
cidade.createInstance("Florianopolis");

```

```

xanxere.set(localizadoem,interior);
chapeco.set(localizadoem,interior);
chapeco.set(proximode,xanxere);
floripa.set(localizadoem,litoral);

```

As inferências realizadas e seus respectivos resultados foram os seguintes:

Inferência	Resultado
Cidades em Santa Catarina	Chapecó, Florianópolis e Xanxerê
Cidades próximas a Xanxerê	Chapecó
Localização de Florianópolis	Litoral, Santa Catarina

Código utilizado para executar as inferências acima apresentadas:

1	<pre> for (Iterator i = sc.list(possuiLocalidade, Thing.INCLUDE_ALL).iterator(); i.hasNext();) {     Thing thing = (Thing)i.next();     if ( thing.isInstanceOf(cidade,false) )     {         System.out.println(thing.getId());     } } </pre>
2	<pre> for (Iterator i = xanxere.list(proximode, Thing.INCLUDE_ALL).iterator(); i.hasNext();) {     System.out.println(((Thing)i.next()).getId()); } </pre>
3	<pre> for (Iterator i = floripa.list(localizadoem, Thing.INCLUDE_ALL).iterator(); i.hasNext();) {     System.out.println(((Thing)i.next()).getId()); } </pre>

As instâncias e as inferências estão representadas graficamente abaixo:

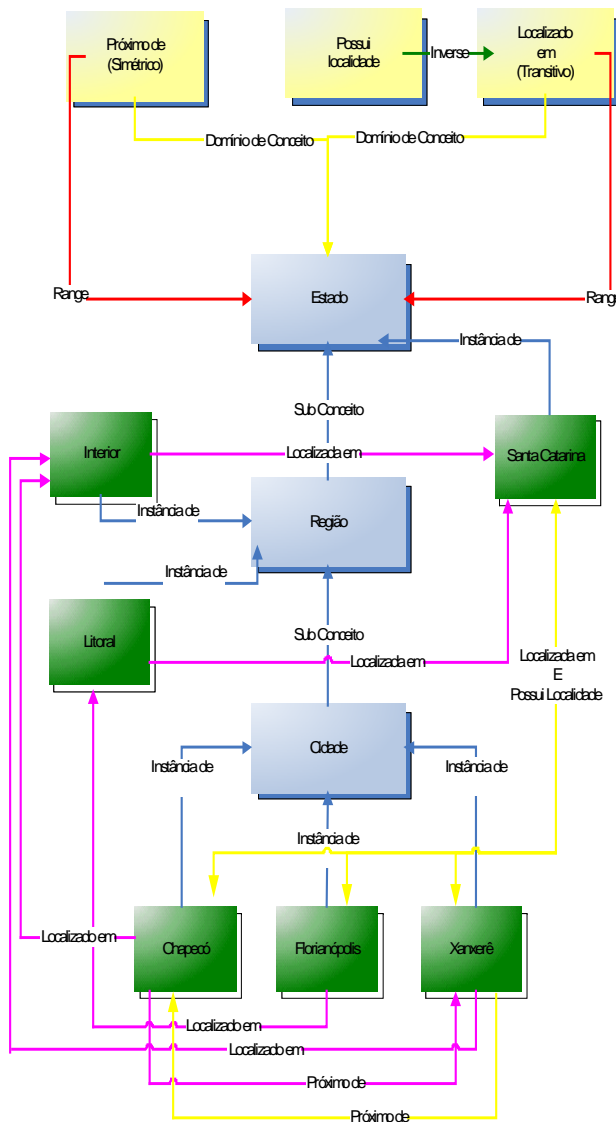


Figura 4: Instâncias e referências criadas

### Considerações Finais

O trabalho de pesquisa realizado aponta para o desenvolvimento de uma série de recursos tecnológicos fundamentais para a consolidação de novas pesquisas e a implementação de aplicações, principalmente na web, no sentido do emprego efetivo dos conceitos embutidos na proposta da web semântica. Tratando-se especificamente da API utilizada neste estudo de caso destacam-se as seguintes observações:

- A API SOFA destaca-se pela facilidade de uso e emprego no desenvolvimento de aplicações de web semântica;
- O processo de criação de ontologias é relativamente complexo (complexidade inerente ao próprio processo), porém é facilitado através do uso da API SOFA;
- É mais simples implementar uma ontologia usando o modelo abstrato proposto pela API

SOFA e depois gerar a representação em OWL, do que fazer o contrário;

- Possui uma limitação: um conceito não pode ser transitivo e simétrico ao mesmo tempo, fato que pode comprometer a criação de ontologias mais complexas.

### Referências

[1] BERNERS-LEE, Tim e MILLER, Eric. **The Semantic Web lifts off**. Disponível em: [http://www.ercim.org/publication/Ercim\\_News/enw51/berners-lee.html](http://www.ercim.org/publication/Ercim_News/enw51/berners-lee.html), 2002. Data de acesso: 08/03/2005.

[2] BERNERS-LEE, Tim. **The Semantic Web. A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities**. Disponível em: <http://www.scientificamerican.com/article.cfm?articleID=00048144-10D2-1C70-84A9809EC588EF21&catID=2>, 2001. Data de acesso: 15/03/2005.

[3] PALMER, Sean. **The Semantic Web: An Introduction**. Disponível em: <http://infomesh.net/2001/swintro/>, 2001. Data de acesso: 16/03/2005.

[4] SILVA, George Henrique. **Construção de Agentes Inteligentes para a Web Semântica**. Disponível em: <http://www.linux.ime.usp.br/~cef/mac499-04/monografias/ghsilva/>, 2003. Data de acesso: 20/03/2005.