



UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE  
CENTRO DE CIÊNCIAS EXATAS E DA TERRA  
DEPARTAMENTO DE INFORMÁTICA E MATEMÁTICA APLICADA  
PROGRAMA DE PÓS-GRADUAÇÃO EM SISTEMAS E COMPUTAÇÃO

# **GeOntoQuery - Um Mecanismo de Busca em Bancos de Dados Geográficos Baseado em Ontologias**

**DISSERTAÇÃO DE MESTRADO**

**Renata Fernandes Viegas**

**Prof. Dra. Valéria Gonçalves Soares**

(orientadora)

Natal-RN  
Julho de 2006

# **Livros Grátis**

<http://www.livrosgratis.com.br>

Milhares de livros grátis para download.

UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE  
CENTRO DE CIÊNCIAS EXATAS E DA TERRA  
DEPARTAMENTO DE INFORMÁTICA E MATEMÁTICA APLICADA  
PROGRAMA DE PÓS-GRADUAÇÃO EM SISTEMAS E COMPUTAÇÃO

# **GeOntoQuery - Um Mecanismo de Busca em Bancos de Dados Geográficos Baseado em Ontologias**

**Renata Fernandes Viegas**

Dissertação de Mestrado submetida ao Programa de Pós-Graduação em Sistemas e Computação do Departamento de Informática e Matemática Aplicada da Universidade Federal do Rio Grande do Norte como parte dos requisitos necessários para a obtenção do grau de Mestre em Sistemas e Computação (MSc.).

**Orientadora: Prof. Dra. Valéria Gonçalves Soares**

Natal-RN  
Julho de 2006

Catálogo da Publicação na Fonte. UFRN / SISBI / Biblioteca Setorial  
Especializada do Centro de Ciências Exatas e da Terra – CCET.

Viegas, Renata Fernandes.

GeOntoQuery : um mecanismo de busca em bancos de dados geográficos  
baseado em ontologias / Renata Fernandes Viegas. – Natal, 2006.  
99 f. : il.

Orientador: Valéria Gonçalves Soares.

Dissertação (Mestrado) – Universidade Federal do Rio Grande do Norte. Centro de Ciências Exatas e da  
Terra. Departamento de Informática e Matemática Aplicada. Programa de Pós-Graduação em Sistemas e  
Computação.

1. Banco de dados geográfico – Dissertação. 2. Sistema de informação – Dissertação. 3. Ontologia – Dissertação.  
I. Soares, Valéria Gonçalves. II. Título.

RN/UF/BSE-CCET

CDU 004.6

# **GeOntoQuery - Um Mecanismo de Busca em Bancos de Dados Geográficos Baseado em Ontologias**

**Renata Fernandes Viegas**

Esta Dissertação de Mestrado foi avaliada em 13 de julho de 2006 e considerada aprovada pelo Programa de Pós-Graduação em Sistemas e Computação do Departamento de Informática e Matemática Aplicada da Universidade Federal do Rio Grande do Norte.

Banca Examinadora:

---

Prof. Dra. Valéria Gonçalves Soares  
Orientadora

---

Profa. Dra. Anne Magály de Paula Canuto  
Examinador Interno

---

Prof. Dra. Ana Carolina Salgado  
Examinador Externo

*Dedico este trabalho aos meus pais,  
Antônio e Moema,  
à minha irmã Adriana,  
e ao meu futuro esposo André Luiz.*

## **AGRADECIMENTOS**

A Deus, por tudo que Ele fez e tem feito em minha vida. Por ELE estar sempre ao meu lado, guiando meus passos e orientando minhas decisões. Obrigada Pai do Céu, sem a sua presença, eu nada seria. A Nossa Senhora, Maria, minha Mãezinha do Céu. Pelas bênçãos, intercessões e proteção concedidas. Estar concluindo esse mestrado é uma graça de Maria em minha vida.

Aos meus pais, Antônio e Moema, pelo exemplo de vida que tenho a seguir. Pela construção de uma família perfeita... Por todo amor, cuidados, educação, e apoio, em todos os momentos. E a minha irmã, Adriana, pela felicidade que só ela me traz. Gorda !! Amo você.

Ao meu futuro esposo, André. Meu amor, você é um presente de Deus em minha vida. Obrigada por tudo, pelo amor, carinho, atenção, preocupações, ajudas, vida compartilhada... Sem o seu apoio, não seria possível a conclusão deste mestrado. Palavras nunca expressarão a minha gratidão... Amo você.

À Prof. Valéria Soares, minha orientadora e amiga, que acreditou e me fez acreditar do quanto eu seria capaz, ajudando nos momentos mais difíceis, e conduzindo a nossa parceria para que tudo saísse "perfeito". Obrigada por ter sido uma excelente orientadora.

Ao Prof. Ricardo Amaral, pelos valiosos conceitos de recifes de corais na visão de um geólogo; E a Prof. Liana Mendes, pelos valiosos conceitos na visão de uma bióloga.

À Thais, companheira de apartamento em Natal, companheira de vida. Obrigada amiga por estar sempre ao meu lado, ouvindo minhas alegrias e aflições, compartilhando segredos e felicidades.

A todos as amizades feitas durante o mestrado, em especial a todos do grupo de pesquisa do Projeto Corais de Maracajaú.

## RESUMO

Sistemas de Informação Geográfica (SIG) são ferramentas computacionais usadas para capturar, armazenar, consultar, analisar e imprimir dados geo-referenciados. Os SIG são sistemas multidisciplinares, ou seja, podem ser utilizados por diferentes comunidades de usuários, cada uma com seus interesses e conhecimentos próprios. Desta forma, diferentes visões de conhecimento sobre uma mesma realidade precisam ser combinadas de modo a atender às necessidades de cada comunidade. Este trabalho apresenta um mecanismo que permite que diferentes comunidades de usuários acessem o mesmo banco de dados geográfico sem ter conhecimento de sua estrutura interna. Utilizamos ontologias geográficas para permitir um conhecimento comum e compartilhado a estes diferentes usuários sobre um domínio específico: os recifes de corais. Usando as descrições das ontologias, que representam o conhecimento de diferentes comunidades, mecanismos são criados para tratar tais diferentes conceitos. Definimos uma camada semântica que faz um mapeamento entre as classes equivalentes das ontologias e interage com o banco de dados geográfico, retornando ao usuário as respostas às suas consultas, independente dos termos utilizados.

## **ABSTRACT**

Geographic Information System (GIS) are computational tools used to capture, store, consult, manipulate, analyze and print geo-referenced data. A GIS is a multi-disciplinary system that can be used by different communities of users, each one having their own interest and knowledge. This way, different knowledge views about the same reality need to be combined, in such way to attend each community. This work presents a mechanism that allows different community users access the same geographic database without knowing its particular internal structure. We use geographic ontologies to support a common and shared understanding of a specific domain: the coral reefs. Using these ontologies' descriptions that represent the knowledge of the different communities, mechanisms are created to handle with such different concepts. We use equivalent classes mapping, and a semantic layer that interacts with the ontologies and the geographic database, and that gives to the user the answers about his/her queries, independently of the used terms.

# SUMÁRIO

Capítulo 1 - Introdução .....	1
1.1 - Motivação .....	1
1.2 - Objetivos .....	3
1.3 - Estrutura do Texto .....	4
Capítulo 2 - Fundamentação Teórica e Trabalhos Relacionados .....	5
2.1 - Sistemas de Informação Geográfica .....	5
2.2 - Interação de Dados Geográficos.....	8
2.3 - Ontologia .....	11
2.4 - SIG baseados em Ontologias .....	14
2.5 - Trabalhos Relacionados .....	15
2.6 - Conclusão .....	17
Capítulo 3 - Tecnologias de Desenvolvimento de Ontologias.....	18
3.1 - RDF (Resource Description Framework) .....	18
3.2 - RDF Schema.....	21
3.3 - Linguagens para definição de Ontologias.....	21
3.3.1 - OIL (Ontology Inference Layer) .....	22
3.3.2 - DAM + OIL .....	23
3.3.3 - OWL (Web Ontology Language).....	23
3.3.3.1 - Definição .....	23
3.3.3.2 - Uso e Sintaxe da OWL .....	25
3.4- Editores de Ontologia .....	27
3.4.1 - OIEd.....	28
3.4.2 - OntoEdit.....	28
3.4.3 - Protégé .....	29
3.5 - Implementação de Ontologias .....	31
3.5.1 - Racer .....	31
3.5.2 - Framework Jena .....	31
3.6 - Conclusão.....	33
Capítulo 4 - Arquitetura Proposta.....	34
4.1 - Modelagem Conceitual e Banco de Dados Geográfico.....	34
4.1.1 - Ontologias Geográficas.....	35
4.2 - Construção da Camada Semântica .....	38
4.2.1 - Detalhamento da Camada Semântica .....	39
4.3 Conclusão .....	42
Capítulo 5 - Protótipo da Aplicação e Cenários de Consulta .....	43
5.1 - Protótipo da Aplicação: Domínio de Recifes de Corais .....	43
5.2 - Contextualização do Problema.....	47
5.3 - Ontologias Desenvolvidas.....	49
5.3.1 - Ontologia do Banco de Dados (Geólogo).....	49
5.3.2 - Ontologia Turista.....	54
5.3.3 - Ontologia Biólogo .....	57
5.3.4 - Interação entre as Ontologias .....	59
5.4 - Protótipo da Aplicação .....	62
5.4.1 - Executando uma Consulta Espacial de Distância .....	62
5.4.2 - Exemplos de Inferências .....	68
5.5 - Outros Cenários de Consulta .....	69
5.5.1 - Primeiro Cenário .....	70
5.5.2 - Segundo Cenário de Consulta .....	75
Capítulo 6 - Conclusão e Trabalhos Futuros.....	78
6.1 - Introdução .....	78
6.2 - Contribuições .....	79
6.3 - Trabalhos Futuros .....	80
Glossário .....	81
Anexos.....	82
Código OWL das Ontologias .....	82

Ontologia do Geólogo .....	82
Ontologia do Turista .....	88
Ontologia do Biólogo .....	92
Referências Bibliográficas .....	96

## LISTA DE FIGURAS

Figura 2.1: Arquitetura de um SIG [Câmara <i>et.al</i> 1996] .....	7
Figura 3.1: Representação do modelo RDF em forma de grafo .....	20
Figura 3.2: Representação do RDF na linguagem XML .....	20
Figura 3.3: Exemplo de código OWL.....	27
Figura 3.4: Editor Protégé.....	30
Figura 3.5: Código Java representado um modelo RDF, utilizando a API Jena .....	33
Figura 4.1: Adaptação de uma camada semântica ao SIG de Cunha [Cunha 2005] .....	35
Figura 4.2: Arquitetura Geral do sistema .....	37
Figura 4.3: Módulos da camada semântica.....	39
Figura 4.4: Parte do método utilizado para a pesquisa de semelhanças conceituais .....	41
Figura 5.1: Área de Proteção Ambiental dos Recifes de Corais [Feitoza 2005] .....	44
Figura 5.2: Visão da região de Maracajaú em relação aos ambientes terrestres ....	45
Figura 5.3: Modelagem Conceitual da Divisão Geomorfológica.....	46
Figura 5.4: Região da APARC fotografada .....	46
Figura 5.5: SIG APA dos Corais [Cunha 2005] .....	47
Figura 5.6: Parte da Ontologia $O_{geo}$ .....	50
Figura 5.7: Ontologia dos Componentes Recifais.....	51
Figura 5.8: Ontologia da Região Bêntica .....	52
Figura 5.9: Definição da classe Espécies, da Ogeo. ....	52
Figura 5.10: Continuação da Ontologia da Comunidade de Geólogos, definindo os flutuantes e rotas de barco. ....	53
Figura 5.11: Parte do código OWL, da ontologia $O_{geo}$ .....	54
Figura 5.12: Região dos Parrachos de Maracajaú. ....	55
Figura 5.13: Parte da ontologia $O_{tur}$ , com conceitos referentes à comunidade turista. .....	55
Figura 5.14: Continuação da $O_{tur}$ , definindo os organismos marinhos. ....	56
Figura 5.15: Ontologia sobre atrações turísticas. ....	57
Figura 5.16: Definição de parte da $O_{bio}$ .....	58
Figura 5.17: Parte de componentes da comunidade bêntica, da $O_{bio}$ .....	58
Figura 5.18: Parte de componentes da comunidade pelágica, da $O_{bio}$ .....	59
Figura 5.19: Relacionamento entre as classes das ontologias do geólogo e turista	60
Figura 5.20: Relacionamento entre as classes das três ontologias presentes na aplicação .....	61
Figura 5.21: Protótipo da Interface de consulta pré-definida .....	62
Figura 5.22: Criação dos modelos das ontologias, utilizando a API Jena.....	63
Figura 5.23: Ontologias da comunidade de geólogos e turistas, na procura por termos semelhantes.....	64
Figura 5.24: Código OWL da $O_{tur}$ .....	65
Figura 5.25: Parte do módulo geração de consulta.....	66
Figura 5.26: Resultado da Consulta: "plantas perto de pedras" .....	67
Figura 5.27: Texto explicativo, gerado pelo <i>módulo geração do dicionário</i> .....	68
Figura 5.28: Interface pré-definida de consulta .....	70
Figura 5.29: Parte do código Java, utilizando a API Jena na procura de classes semelhantes. ....	71
Figura 5.30: <i>screenshot</i> do resultado da consulta "selecione as áreas ao redor dos flutuantes e corpos coralíneos".....	73
Figura 5.31: <i>Zoom in</i> na figura da consulta "selecione as áreas ao redor dos flutuantes e corpos coralíneos" .....	74
Figura 5.32: Janela <i>pop-up</i> do dicionário gerado para a consulta em questão.....	74
Figura 5.33: Interface pré-definida da comunidade de biólogos. ....	75
Figura 5.34: Parte do código OWL da ontologia do biólogo. ....	76

Figura 5.35: <i>screenshot</i> do resultado da consulta "selecione as áreas que possuem lama, areia e cascalho". .....	77
Figura 5.36: Janela <i>pop-up</i> do dicionário gerado para a consulta do usuário biólogo .....	77

## **LISTA DE TABELAS**

Tabela 3.1: Exemplo do uso do modelo RDF .....	19
Tabela 5.1: Relacionamentos espaciais.....	66

---

# **Capítulo 1**

## **Introdução**

### **1.1 - Motivação**

Integração da Informação é a combinação de diferentes tipos de informações, em um mesmo *framework*, a serem analisadas, recuperadas e manipuladas a fim de obter um objetivo comum [Fonseca 2001].

Sistemas de Informação Geográfica (SIG) são ferramentas computacionais usadas para capturar, armazenar, consultar, manipular, analisar e imprimir dados geo-referenciados. Em SIG, a integração de informação é dificultada pela grande diversidade de dados que envolvem fatores sintáticos e semânticos. Fatores sintáticos dizem respeito aos dados presentes em um SIG serem provenientes de diferentes fontes de dados (satélites, fotografias aéreas), havendo desta forma, uma dificuldade na integração destes diferentes formatos. Fatores semânticos, por outro lado, dizem respeito ao significado (semântica) dos dados em um SIG. É necessário que o dado tenha não só um nome, mas um significado associado a ele.

A capacidade de integrar múltiplos domínios com diferentes significados para os usuários que irão utilizá-los é chamada de interoperabilidade [Fonseca 2001].

A integração de informação geográfica tem ganhado importância por causa das novas possibilidades de interconexão desta informação em todo mundo. O que antigamente se restringia a simples mapas de papel, hoje se transformou em uma grande quantidade de dados, provenientes de diversas fontes, e utilizados por diversos usuários, de diferentes áreas de conhecimento.

Os SIG são sistemas multidisciplinares, ou seja, podem ser utilizados por diferentes comunidades de usuários, cada uma com seus interesses e conhecimentos próprios. Desta forma, diferentes visões de conhecimento

---

sobre uma mesma realidade precisam ser combinadas de modo a atender às necessidades de cada comunidade.

Pessoas com interesses diferentes identificam de forma também diferente a mesma região geográfica. Um dos pontos básicos destas diferenças é refletido nos nomes dados às entidades do mundo real. Estas entidades geográficas, coletadas e armazenadas em bancos de dados geográficos (BDG), são chamadas de feições na literatura de referência da área [OGC 1999]. Feições geográficas são coletadas e armazenadas nos BDG, que são modelados por um modelo conceitual específico.

Pode-se exemplificar a multidisciplinaridade de um SIG com um cenário onde um usuário procura informações sobre "pedras", mas cujo banco de dados geográfico tem apenas informações armazenadas sobre "rochas", e, neste caso, o retorno dessa busca seria vazio. No entanto, semanticamente o retorno da consulta sobre pedras deveria envolver os resultados sobre rochas. Como resolver isso?

Segundo Fonseca [Fonseca 2001], os sistemas de informação devem ser capazes de entender os modelos que os usuários fazem do mundo e seus significados e de entender também os modelos por trás das fontes de informação. É necessário desenvolver sistemas que ultrapassem as barreiras da informação, mostrando aos diferentes usuários não apenas o dado em si, mas também o seu significado.

Os sistemas atuais deverão ser capazes de resolver a interoperabilidade semântica, onde um fato pode ter mais do que uma descrição. O termo "semântica" é usado para se referir ao significado básico das entidades. Estas entidades são partes de um modelo mental do usuário que representa os conceitos do mundo real, ou mais especificamente, do mundo geográfico [Fonseca 2001].

Atualmente vêm crescendo o estudo de como alcançar a interoperabilidade com a definição de ontologias, funcionando como bases de conhecimento, especificando um vocabulário, relativo a um domínio, e definindo entidades, classes, propriedades, predicados e funções e as relações entre estes componentes.

O termo ontologia é proveniente da Filosofia, e significa "uma especificação formal e explícita que tenta, da melhor forma possível, aproximar a estrutura de mundo definida por uma conceitualização"

---

[Guarino 1998]. Uma ontologia geográfica é uma conceitualização de um fenômeno ou objeto geográfico no mundo real.

As ontologias expressam o formalismo dos conceitos e das relações acerca de um domínio. Pode-se dizer que a ontologia visa desenvolver um conjunto de regras que possibilitem a inferência de forma que a máquina possa, através do acesso a essas regras, abstrair um significado semântico das informações disponibilizadas.

A busca pela semântica de dados geográficos é algo necessário para a interoperabilidade entre SIG.

## **1.2 – Objetivos**

O objetivo principal deste trabalho consiste em prover um mecanismo que permita que diferentes usuários acessem o mesmo Banco de Dados Geográfico (BDG), sem que para isso seja necessário conhecer sua estrutura interna, usando conceitos próprios à sua área de atuação. Isso será feito através de ontologias geográficas e dos relacionamentos entre as mesmas.

A idéia é permitir, em um mesmo BDG, diferentes tipos de informação, na medida em que diferentes comunidades de usuários utilizam o mesmo. Cada comunidade tem diferentes termos e conceitos para se referir a um mesmo objeto geográfico.

O que está de fato implementado no banco de dados é uma visão de apenas uma dessas comunidades. Mas, cada comunidade interage com o banco de dados normalmente com o conhecimento específico de sua área, obtendo os resultados de suas consultas de forma transparente.

Usando a inferência inerente às ontologias, é possível que esses conceitos sinônimos sejam relacionados, melhorando assim a busca no banco de dados.

Para isso, podemos enumerar nossas metas e objetivos em:

- Construção de ontologias geográficas, que reflitam as diferentes visões, de diferentes comunidades de usuários;
- Integrar estas ontologias, através de uma similaridade de termos, presentes nas classes das diferentes ontologias;

- 
- Adaptar à aplicação desenvolvida anteriormente por Cunha [Cunha 2005], uma camada semântica, que irá interagir com as ontologias, permitindo que cada comunidade de usuário realize consultas, usando os termos que ela conhece.

### **1.3 – Estrutura do Texto**

Esta dissertação está dividida nos seguintes capítulos:

- O Capítulo 2 apresenta uma revisão da literatura, contendo toda a fundamentação teórica que serviu como embasamento para realização deste trabalho;
- O Capítulo 3 mostra um levantamento das tecnologias envolvidas para a implementação e desenvolvimento de ontologias;
- O Capítulo 4 mostra a nossa solução de interação das ontologias com o banco de dados, através da inserção de uma camada semântica, que irá prover mecanismos para consultas em BDG através de ontologias;
- O Capítulo 5 mostra um protótipo da aplicação, com resultados de consultas utilizando a camada semântica;
- O Capítulo 6 apresenta as conclusões do nosso trabalho, bem como sugestões para pesquisas futuras.

---

# **Capítulo 2**

## **Fundamentação Teórica e Trabalhos Relacionados**

### **2.1 – Sistemas de Informação Geográfica**

Os Sistemas de Informação Geográfica (SIG) tiveram um acelerado crescimento na década de 80, beneficiando-se dos avanços da microinformática e do estabelecimento de grandes centros de estudo no assunto. No decorrer desta década, ocorreu uma grande difusão do uso de SIG, voltado para cartografia. Nos anos seguintes, a evolução dos SIG deu-se com a incorporação de funções de análise espacial e uso de imagens [Câmara *et.al* 1996]. Atualmente, há uma grande aceitação e crescimento dos SIG nas grandes empresas e centros de pesquisa, sendo utilizados como bibliotecas digitais geográficas, com necessidade de compartilhamento e reuso dos dados, e crescimento de várias pesquisas na área de interoperabilidade entre diversas fontes de dados [Fonseca 2001].

Dados geográficos são dados espaciais (que representam fenômenos com alguma dimensão espacial), cuja dimensão é a superfície terrestre.

Um dado espacial possui características associadas a ele, tornando-o um dado complexo de ser analisado e armazenado [Câmara *et.al* 1996].

São estas:

- Característica não-espacial: representa os atributos textuais do fenômeno, ou seja, nome e atributos específicos do mesmo.
- Característica espacial: o seu geo-referenciamento, sua posição na Terra.
- Características temporais: são importantes, pelo fato dos dados geográficos serem variantes com o tempo.
- Características gráficas: são as representações pictóricas dos dados.

---

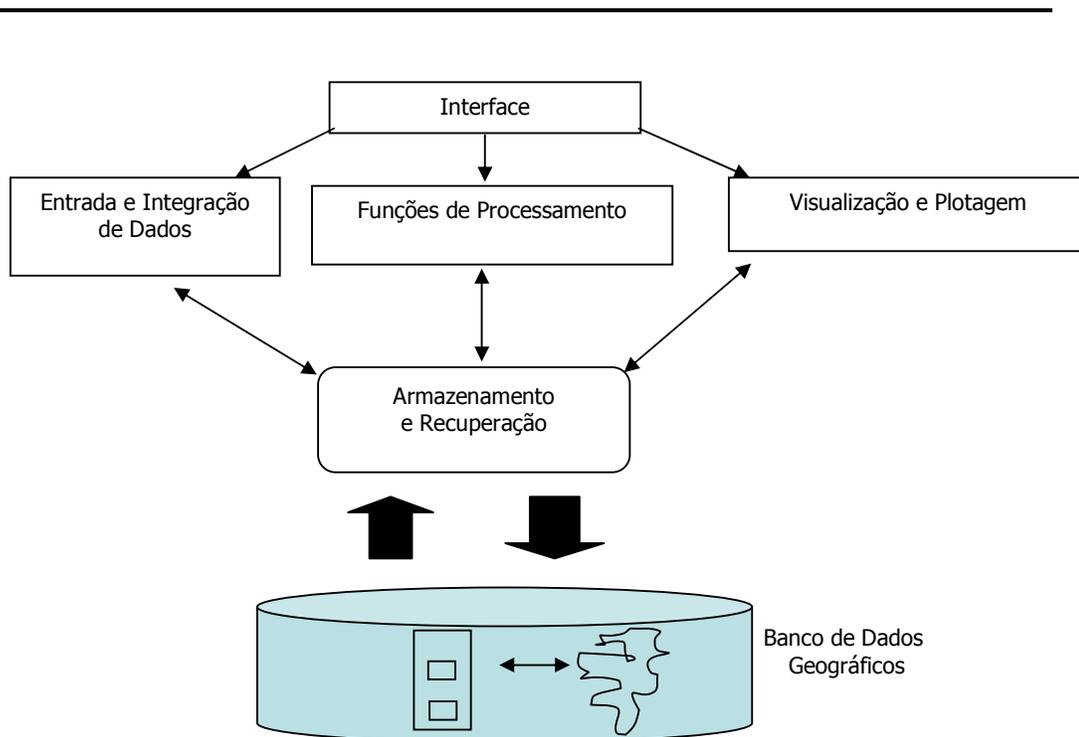
Segundo Güting [Gütting 1994], um banco de dados é espacial se o mesmo gerencia dados que estão relacionados com algum espaço. No caso de SIG, temos o espaço geográfico (superfície terrestre) como espaço utilizado.

Cada objeto encontrado em um SIG pode ser relacionado com alguma localidade na Terra. Os objetos em um SIG são definidos pelas suas posições e pelos múltiplos atributos que descrevem as características do objeto.

Um SIG é composto pelos seguintes elementos: [Câmara *et.al* 1996]

- Interface com o usuário: canal de comunicação entre o usuário e o sistema
- Entrada e integração de dados: captura os dados espaciais de diversas fontes fornecidas pelo usuário, convertendo-os em formato digital.
- Funções de processamento: funções aplicadas aos dados espaciais que foram capturados, facilitando a análise dos mesmos.
- Visualização e plotagem: permite, a partir de elementos selecionados do banco de dados, produzir mapas de saída na tela ou em alguns dispositivos de saída em meio sólido como uma impressora ou *plotter*.
- Armazenamento e recuperação de dados: armazenamento dos dados em um BDG, e todas as operações sobre os dados que são tratados pelo sistema gerenciador de banco de dados.

Cada sistema, em função de seus objetivos e necessidades, implementa estes componentes de forma distinta, mas todos estão usualmente presentes em um SIG, e se relacionam como mostra a Figura 2.1 [Câmara *et.al* 1996].



**Figura 2.1:** Arquitetura de um SIG [Câmara *et.al* 1996]

O espaço geográfico é definido como uma coleção de localizações na superfície da Terra onde ocorrem os fenômenos geográficos. A partir de idéias conceituais de fenômenos geográficos, é possível formalizar a representação do espaço e das propriedades espaciais. Podem-se notar diferentes maneiras do que está acontecendo na superfície terrestre, percebendo o espaço como sendo ocupado por *entidades*, ou imaginando uma variação contínua de um atributo de interesse, ou *campo*, através do espaço, de acordo com alguma função matemática [Soares 2002].

O modelo de campos enxerga o espaço geográfico como uma superfície contínua, sobre a qual variam os fenômenos a serem observados segundo diferentes distribuições. Já o modelo de objetos representa o espaço como uma coleção de entidades distintas e identificáveis, cuja posição pode ser mapeada usando algum sistema de coordenadas [Câmara *et.al* 1996].

---

## 2.2 – Interação de Dados Geográficos

Os SIG são sistemas multidisciplinares, ou seja, podem ser utilizados por diferentes comunidades de usuários tais como geógrafos, biólogos ou geólogos, cada um com seus interesses próprios. Surge então uma área de estudo que podemos chamar de interação da informação geográfica.

Cada comunidade trabalha com informações segundo a sua visão e necessidade de trabalho. Por exemplo, dentro de uma comunidade de biólogos, um corpo de água específico pode ser um lago usado como o habitat para espécies, e, conseqüentemente, pode haver um conceito ou um nome especial para referenciá-lo. Por outro lado, esse mesmo corpo de água pode ser visto de diferentes maneiras, por outras comunidades. Para um departamento de tratamento de água, um lago pode ser visto como uma fonte da água pura; para um cientista ambiental, é um habitat dos animais selvagens; para o departamento de turismo, é um ponto de recreação, atividades e brincadeiras [Fonseca 2001]. Cada comunidade tem seus interesses e denominações próprias para definir um mesmo objeto. Estas diferentes formas de definir conceitos originam as barreiras semânticas [Câmara *et.al* 2005]. Barreiras semânticas aparecem em comunidades com cultura e história diferentes, que interpretam distintamente a realidade geográfica e produzem sistemas conceitualmente heterogêneos.

É necessário desenvolver sistemas que suportem as barreiras semânticas, mostrando aos diferentes usuários não apenas o dado em si, mas também o seu significado. Para isso, devemos ter uma interoperabilidade entre sistemas.

Vários autores têm sua própria definição do que seria interoperabilidade:

“A capacidade de trocar e usar informação, integrando múltiplos domínios, é chamada de interoperabilidade” [Fonseca 2001].

“Interoperabilidade: capacidade de compartilhar informações e processos entre ambientes computacionais heterogêneos, autônomos e distribuídos” [Lima 2004].

“Interoperabilidade pode ser vista como a capacidade que um sistema possui de integrar diferentes visões de informação, de diferentes

---

usuários. Assim, o sistema permitirá que usuários heterogêneos compartilhem informações de forma autônoma e, até mesmo, distribuída – características importantes para definir interoperabilidade entre sistemas” [Yuan 1998].

Para que o intercâmbio de dados espaciais aconteça sem o comprometimento da informação, é necessário um alto grau de interoperabilidade entre SIG [Lima 2004]. É necessário garantir que a transferência de dados entre sistemas heterogêneos seja feita de maneira transparente, e que os dados trocados tenham significado para todos os usuários.

Para resolver o problema da interoperabilidade entre diferentes formatos dos SIG (ou seja, a interoperabilidade sintática), várias alternativas foram propostas, como a criação de padrões como o SDTS (*Spatial Data Transfer Standard*) [Sondheim *et al.* 1999] e o SAIF (*Spatial Archive and Interchange Format*) [Surveys 1994].

O padrão SAIF [Surveys 1994] propõe uma linguagem para especificação de troca de dados, permitindo a definição de *profiles* contendo regras e restrições sobre a linguagem. Seu objetivo é facilitar a tradução de modelos de SIG e de outros padrões para o padrão definido pelo SAIF [Surveys 1994], e facilitar o compartilhamento de dados através de um *profile* comum.

O SDTS [Sondheim *et al.* 1999] é um padrão para transferência de dados espaciais entre diferentes sistemas. Ele especifica construtores de troca, formatos de endereçamento, estrutura e conteúdo para dados georeferenciados.

Padrões para troca de dados são necessários e úteis para a transferência de dados, mas eles não têm a capacidade de transferir o significado associado com o dado. Um interesse crescente no desenvolvimento de um modelo de dados comum tem conduzido a linhas de pesquisa em integração de informação geográfica. Uma das maiores iniciativas é o consórcio OpenGIS [McKee and Buehler 1996], uma associação de desenvolvedores de software e agências governamentais para definir uma série de requerimentos, padrões e especificações para suportar a interoperabilidade entre SIG. Seu objetivo é alcançar especificações abstratas para componentes de software relacionados aos SIG. Mas, nem

---

um padrão de dados nem um modelo de dados comum permitem a transferência do significado da informação [Fonseca 2001].

Embora haja uma busca pela interoperabilidade sintática, ou seja, entre formatos diferentes, que trata de transformações estruturais dos dados, os atuais sistemas de informação devem também ser capazes de resolver a interoperabilidade semântica, que trata da equivalência de significados e modelos, onde um fato pode ter mais que uma descrição. Os problemas semânticos persistem e impedem a interoperabilidade, e são claramente os mais difíceis problemas, em se tratando de informações geográficas. Diferentes visões da realidade geográfica sempre existirão por pessoas com culturas diferentes, pois a própria natureza é complexa e leva a percepções distintas.

Pessoas diferentes, de culturas diferentes, percebem diferentemente uma mesma região geográfica. Um dos principais pontos desta diferença é a nomenclatura dada às entidades do mundo real. Dados sobre estas entidades são coletados e armazenados nos SIG que são concebidos de acordo com um modelo conceitual próprio.

Seria interessante conviver com estas diferentes formas de conhecimento sobre a realidade e tentar criar mecanismos para implementar e combinar diferentes visões, ou seja, representar o conhecimento geográfico no computador buscando interoperabilidade pela equivalência semântica dos conceitos entre sistemas distintos.

Assim, os sistemas de informação deverão ser capazes de entender o modelo que o usuário faz do mundo e seus significados e de entender também os modelos por trás das fontes de informação [Fonseca 2001]. O termo semântica é usado para se referir ao significado básico das entidades. Estas entidades são partes de um modelo mental que representa os conceitos do mundo real, ou mais especificamente, do mundo geográfico [Fonseca 2001].

Para este tipo de integração de informação acontecer, é necessário primeiro explicitar formalizações de conceitos mentais que as pessoas têm sobre o mundo real. Além disso, estes conceitos precisam ser agrupados por comunidades representando os argumentos básicos que existem dentro de cada comunidade [Fonseca 2001]. Uma vez que os modelos mentais

---

forem explicitados e formalizados, mecanismos são criados para generalizar um tipo específico, adicionando especificações de conceitos.

Atualmente vêm crescendo o estudo de como alcançar a interoperabilidade com a definição de ontologias, funcionando como bases de conhecimento, especificando um vocabulário, relativo a um domínio, e definindo entidades, classes, propriedades, predicados e funções e as relações entre estes componentes.

### **2.3 – Ontologia**

O termo ontologia é proveniente da Filosofia, significando a representação da existência através de uma explicação sistemática, como sendo a concepção de tudo aquilo que pode “existir” ou “ser”. Na área de informática, a ontologia começou a ser utilizada pela Inteligência Artificial, como sendo “uma especificação formal e explícita que tenta, da melhor forma possível, aproximar a estrutura de mundo definida por uma conceitualização” [Guarino 1998]. Nessa definição, “formal” significa legível para computadores; “explícita” diz respeito a conceitos, propriedades, relações, funções, que são explicitamente definidos; “conceitualização” diz respeito a um modelo abstrato de algum fenômeno do mundo real [Guarino 1998]

Uma formalização explícita do nosso modelo mental é geralmente chamada de ontologia. A descrição básica das coisas reais no mundo, a descrição de o que seria a verdade, é chamado de Ontologia. Assim, há apenas uma Ontologia, mas várias ontologias. [Fonseca 2001]

As comunidades que oferecem informações e que tem acesso a elas, cada uma tem a sua ontologia. Cada uma destas ontologias pode ser dividida em pequenas ontologias. Seu nível de detalhe relata o nível de detalhe da informação geográfica. A informação pode ser integrada em diferentes níveis de detalhes [Fonseca 2001].

As ontologias podem atuar sobre fontes de dados proporcionando organização e recuperação mais efetiva. Possibilitam uma compreensão comum e compartilhada de um domínio. Desempenham um papel importante no intercâmbio de informações, ao proporcionar estrutura semântica às fontes de dados.

---

As ontologias são muitas vezes confundidas com esquemas de bancos de dados. Semanticamente, elas são mais ricas do que simples esquemas. Um esquema de banco de dados representa o que está armazenado no banco de dados; uma ontologia representa conceitos no mundo. A informação que existe nas ontologias tem que ser adaptada para preencher as classes dos bancos de dados. [Fonseca 2001]

Uma ontologia é uma descrição formal dos conceitos de um domínio, e das propriedades de cada conceito, que descrevem várias características e atributos deste conceito.

As classes compõem o foco da maioria das ontologias. Uma classe é a extensão do conceito de um tipo abstrato, uma estrutura que representa uma entidade simples, que descreve o conteúdo de informação e seu comportamento. Uma classe define a estrutura e o conjunto de operações que são comuns a um grupo de objetos.

Uma instância, ou objeto, representa uma ocorrência individual de uma classe. Enquanto a classe é a definição do tipo, uma instância é a estrutura de dados representada na memória de um computador, e manipulada por um sistema de software.

Uma ontologia possibilita a comunicação sobre um domínio particular, definindo explicitamente o vocabulário de termos básicos, o significado preciso destes termos e as relações entre eles. Assim, modelando formalmente o domínio da aplicação, o entendimento sobre o que está representado no BDG será mais fácil.

O desenvolvimento de uma ontologia é um processo iterativo. Construí-la consiste em aprender e compreender os conceitos e visões relevantes para os diferentes usuários que irão utilizá-las.

Dependendo da aplicabilidade, uma ontologia pode ser descrita em diferentes níveis de abstração [Novello 2002].

- Alto Nível - descreve conceitos gerais como espaço, tempo, assunto, objeto, evento, ação, os quais são independentes de um problema ou domínio particular.
- Domínio – descreve o vocabulário relacionado ao domínio genérico, especializando os termos introduzidos na ontologia de alto nível.

- 
- Tarefa - expressam conceituações sobre a resolução de problemas, independentemente do domínio em que ocorram, isto é, descrevem o vocabulário relacionado a uma atividade ou tarefa genérica.
  - Aplicação – descreve conceitos dependendo de um domínio e tarefa específicos, os quais são freqüentemente especializações das ontologias relacionadas.

Uma ontologia requer o uso de um vocabulário específico para descrever o domínio, que geralmente é feito através de uma especificação formal acerca deste, possibilitando assim o compartilhamento e reuso do conhecimento.

Uma ontologia às vezes é confundida com vários outros termos e estudos. Assim, uma ontologia:

- Não é uma coleção de fatos sobre uma situação específica, mas ela provê todas as entidades semânticas (por exemplo, classes) para descrever essa situação. Uma descrição concreta de uma situação usa esses conceitos para criar exemplos e anotá-los com seus predicados e relações.
- Não é um esquema de banco de dados que descreve categorias e tipos de dados e organização no banco de dados. Um esquema de banco de dados pode ser derivado de uma ontologia pela adição da informação do tipo de dados e traduzindo o formalismo da representação do conhecimento em um paradigma de gerenciamento do banco de dados, como o relacional ou o orientado a objetos. Vice-versa, um esquema de banco de dados pode ser usado inicialmente para estabelecer conceitos para povoar uma ontologia.
- Não é uma taxonomia que conhece apenas relações de superclasses e sub-classes, onde uma ontologia é aberta em muitos tipos de relacionamentos entre conceitos (ex: topológicos, compositivas).
- Não é um vocabulário ou dicionário já que as palavras no dicionário não descrevem necessariamente a hierarquia e a relação entre cada conceito e não são organizadas de forma que suportem inferência computacional. Numa ontologia pode-

---

se seguir um caminho de qualquer conceito para outro através das relações “é um”, por exemplo, ou outras relações.

Para definirmos formalmente uma ontologia, necessitamos de linguagens que auxiliem nessa definição. Diversas linguagens foram e vêm sendo desenvolvidas, entre elas a OIL (*Ontology Inference Layer*) [OIL 2000], DAML (*DARPA Agent Markup Language*) [DAML 2003], DAML + OIL [DAML+OIL 2001], OWL (*Web Ontology Language*) [OWL 2004], entre outras.

A principal característica destas linguagens é a capacidade de representar ontologias em RDF [RDF 2004] e RDF Schema [RDFS 2004], que são arquiteturas já consagradas pela W3C [W3C 2005] para interoperabilidade de dados na Web [Moura 2001].

Além de linguagens, existem também editores de ontologias, ferramentas que facilitam a criação e manipulação das mesmas. Dentre as ferramentas mais citadas na literatura, que estão associadas à criação e manipulação de ontologias, estão a OIEd [OIEd 2003], OntoEdit [OntoEdit 2003] e Protégé [Protégé 2005].

## **2.4 – SIG baseados em Ontologias**

O uso de ontologias traduzidas em componentes ativos do sistema de informação geográfica conduz a um ODGIS (Ontology-Driven Geographic Information Systems) [Fonseca 2001].

ODGIS são construídos usando componentes de software derivados de várias ontologias. Estes componentes de software são classes que podem ser usadas para desenvolver novas aplicações.

Em ODGIS, uma ontologia é um componente, tal como o banco de dados, cooperando para cumprir os objetivos do sistema. O primeiro passo para se construir um ODGIS é especificar as ontologias, e representá-las de modo formal. Navegando através das ontologias, os usuários obtêm informações sobre o conhecimento embutido nos sistemas.

Assim, uma ontologia é a descrição de uma dada realidade com um vocabulário específico, usando um conjunto de premissas de acordo com o sentido intencional das palavras do vocabulário [Fonseca 2001]. O conjunto

---

das ontologias que estão em um SIG faz parte da camada semântica, e esta camada é responsável por prover conhecimento para que o SIG possa adaptar, filtrar ou explicar a informação, armazenada em sua base de dados, para os diferentes tipos de usuários.

Na arquitetura ODGIS, há diferentes níveis de ontologia. Assim, há diferentes níveis de informação e detalhes. O mapeamento das entidades que estão nas ontologias para componentes de software é feito usando orientação a objetos e herança múltipla.

Um ODGIS é composto por um editor de ontologias, por um servidor de ontologias, por ontologias especificadas formalmente e por classes derivadas de ontologias [Fonseca 2001].

O resultado de um ODGIS é um conjunto de especificações formais das ontologias e um conjunto de classes, formando assim a geração de conhecimento. As ontologias armazenadas são administradas por um servidor de ontologias. A comunicação entre o SIG e o servidor de ontologias é feita usando mediadores, responsáveis na extração de informação do SIG e criação das instâncias das classes.

## **2.5 – Trabalhos Relacionados**

Pesquisas sobre interoperabilidade em sistemas de informação são motivadas pelo aumento da heterogeneidade no mundo da informática. Há uma grande quantidade de tipo e significado de informações e há a necessidade de troca dessas informações.

Guarino [Guarino 1998] propôs o uso de ontologias nos sistemas de informação, conduzindo ao desenvolvimento de sistemas de informação dirigidos a ontologias (ODIS). Neste trabalho, não há o uso de ontologias geográficas.

Fonseca [Fonseca 2001] defende o uso de ontologias geográficas para auxiliar na interação semântica dos dados em um SIG. Em seu trabalho, ele define um ODGIS (Sistemas de Informação Geográfica Dirigidos a Ontologias), que são sistemas que utilizam ontologias traduzidas em componentes de software. O resultado de um ODGIS é um conjunto de especificações formais das ontologias e um conjunto de classes, formando a geração de conhecimento. As ontologias definidas em [Fonseca 2001] são

---

ditas geográficas por apresentar um domínio geográfico da aplicação. Informações mais detalhadas sobre o ODGIS foram apresentadas na seção 2.4.

Egenhofer [Egenhofer 2002] reporta a criação da Web Semântica Geo-Espacial: um *framework* para recuperação de informação geo-espacial baseados na semântica de ontologias espaciais. Esse *framework* permite ao usuário recuperar os dados de maneira mais precisa, baseado na semântica associada a estes dados. É proposto o mapeamento entre as classes das ontologias utilizando objetos Java, criando as classes através de herança entre elas e promovendo o reuso de informações.

Trabalhos baseados em ontologias para formulação de consultas também podem ser encontrados em [Dongilli *et. al* 2004], [Franconi and Tessalis 2005]. Em [Dongilli *et. al* 2004] é definido o projeto SEWASIE, que tem como objetivo permitir um acesso uniforme a bases de dados heterogêneas através de uma interação de ontologias. A interface de consulta é formulada de acordo com as ontologias, sobre um domínio de Economia. A interface é disponibilizada em HTML, e o projeto utiliza processamento de linguagem natural para formulação das consultas. As ontologias não são baseadas em um domínio geográfico, e o ponto forte deste trabalho é o processamento de linguagem natural, que permite que o usuário digite o texto que deseja procurar.

Em [Cardoso *et. al* 2005] é apresentada uma arquitetura que une os conceitos da Web Semântica (ontologias e RDF) com técnicas de expressões regulares, com o intuito de realizar recuperação e extração de dados de páginas Web. As tecnologias envolvidas são: expressões regulares (técnicas que permitem identificar e extrair os dados requeridos de arquivos HTML e XML), ontologias e APIs Java (no trabalho foi utilizada a API Jena). Nesta arquitetura, os arquivos são recuperados de páginas Web por agentes de software, e são submetidos a uma análise de *parsers*, identificando e extraindo dados relevantes de arquivos HTML/XML. São usadas ferramentas para navegar sobre a estrutura dos arquivos, tendo acesso a palavras e expressões que são partes dos documentos, para posterior comparação com termos de determinados domínios, descritos em uma ontologia específica. As ontologias são descritas utilizando a linguagem DAML+OIL.

---

Dos trabalhos analisados na literatura que utilizam ontologias, poucos utilizam a teoria do que seriam ontologias geográficas. Os relacionamentos são simples, na sua maioria "ISA" (é um), sem acrescentar nenhuma informação geográfica às ontologias. Em nosso trabalho, uma ontologia é dita geográfica se ela armazena propriedades geográficas (por exemplo, localização), como também relacionamentos geográficos.

Em nossa proposta, não precisamos que as classes herdem informações uma das outras. Utilizamos a linguagem Java, juntamente com a API Jena, que possui métodos específicos para trabalharmos com mapeamento entre as classes. Nossa proposta é fazer a conexão entre as ontologias e o BDG, através das semelhanças conceituais, presentes nas classes das ontologias. Com um único BDG, o custo da aplicação é bem menor, pois só é necessário coletar os dados uma única vez. Além disso, nenhuma das propostas citadas na literatura é baseada em modelagem ambiental e marinha, o que aumenta a dificuldade de coletar a informação e de selecionar aquelas que são mais relevantes.

## **2.6 – Conclusão**

Este capítulo teve como objetivo introduzir os conceitos básicos que foram estudados para a realização deste trabalho.

Com base em o que são sistemas de informação geográfica, e os problemas existentes na interação de dados geográficos, elaboramos uma proposta para interação semântica, baseada em ontologias geográficas (conceitos apresentados neste capítulo).

No próximo capítulo, serão apresentadas as tecnologias necessárias para o desenvolvimento e implementação da nossa proposta de interação semântica.

---

## **Capítulo 3**

### ***Tecnologias de Desenvolvimento de Ontologias***

Para a interação de SIG com ontologias, será necessário utilizar algumas tecnologias e ferramentas disponíveis. Um levantamento de algumas tecnologias existentes será feito nos próximos tópicos, bem como a justificativa pelas tecnologias escolhidas.

Antes de descrever as linguagens e editores existentes para definir ontologias, se faz necessário descrever dois padrões, recomendados pelo W3C, com o objetivo de promover o intercâmbio de metadados desenvolvidos de forma independente, de modo que eles possam ser compreendidos pelo computador, sem o auxílio do discernimento humano. As arquiteturas RDF e RDFS consistem em uma estrutura básica para o processamento de metadados, provendo interoperabilidade entre aplicações que trocam informações, compreendidas pela máquina e entre si na Web [RDF 2004] que descrevem formalmente um determinado recurso da Web através de seus metadados. A criação de padrões, arquiteturas de metadados, ontologias, entre outros projetos conduzidos pelo W3C, têm sido realizados no sentido de buscar a otimização da integração, intercâmbio e compreensão semântica das informações Web [RDF 2004].

#### **3.1 – RDF (Resource Description Framework)**

O *Framework* de descrição de recursos (RDF) é um padrão, tecnicamente recomendado pelo W3C. A definição para o que seja um recurso ainda é assunto de debate. Podemos pensar em recurso como qualquer coisa que se possa identificar [RDF 2004].

O modelo de dados do RDF é composto por [RDF 2004]:

- Recurso (*Resource*): Uma entidade, que pode ser referenciada através de um *Universal Resource Identifier* (URI). URI é um identificador de recursos universal, que identifica unicamente este recurso, podendo o URI ser um sujeito, um objeto ou uma URL (*Uniform Resource Locator*).

- Propriedades (*Properties*): Atributo ou característica que representa um recurso, ou o relacionamento entre recursos.
- Declaração (*Statement*): Uma associação entre uma propriedade (ou o valor desta) e um recurso específico. Uma declaração é dividida em sujeito, predicado e objeto, ou seja, declaração = recurso (sujeito) + propriedades (predicado) + valor da propriedade (objeto).

Um exemplo fictício de como se usa o modelo RDF seria a declaração: “Um recife de coral é animal marinho”, representada pelo recurso <http://www.corais.com.br>.

A Tabela 3.1 ilustra a declaração de um *statement* que descreve o recurso de uma página HTML com a URI <<http://www.corais.com.br>>, cuja propriedade é *animal* e o objeto é *recife de coral*.

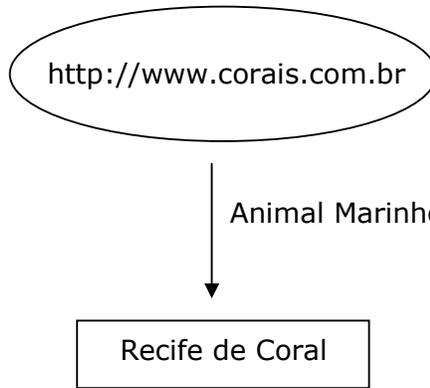
<i>Subject</i> (Recurso)	<i>Predicate</i> (Propriedades)	<i>Object</i> (Literal)
<a href="http://www.corais.com.br">http://www.corais.com.br</a>	É animal	Recife de coral

**Tabela 3.1:** Exemplo do uso do modelo RDF

O modelo de dados RDF fornece uma arquitetura para processamento e intercâmbio de metadados. Este modelo é usado para especificar relacionamentos entre as classes, e fornecer interoperabilidade estrutural. Porém, o modelo RDF não fornece maneiras para declaração de propriedades, e relacionamentos entre estas propriedades [RDF 2004]. Para suprir esta necessidade, tem-se o RDF Schema [RDFS 2004].

A sintaxe do RDF utiliza a linguagem XML e *namespaces* (URI) para associar cada propriedade com o esquema que a define.

O modelo RDF também pode ser representado em forma de grafos. Um exemplo para a mesma sentença: “Um recife de coral é animal marinho” é mostrado na Figura 3.1.



**Figura 3.1:** Representação do modelo RDF em forma de grafo

A sintaxe RDF utiliza a linguagem XML e *namespaces* (URI) para associar cada propriedade com o esquema que a define. Na representação da linguagem XML, a mesma sentença é vista na Figura 3.2.

```
1: <?xml version="1.0"?>
2: <rdf:RDF xmlns: rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
3:   xmlns: s="http://www.corais.com.br/esquema#>
4:   <rdf:Description rdf:about="http://www.corais.com.br/animall.html">
5:     <rdf:type resource=""="http://www.corais.com.br/esquema#animal">
6:     <s:nome>recife de coral</s:nome>
7:   </rdf:Description>
8: </rdf:RDF>
```

**Figura 3.2:** Representação do RDF na linguagem XML

Na Figura 3.2, a linha 1 indica a versão da linguagem XML que está sendo utilizada. Da linha 2 a 4 serão declaradas as variáveis de *namespaces* "rdf:" e "s:" que especificam a localização de um novo vocabulário para representar o recurso de uma forma mais significativa. A identificação do recurso ocorre na linha 4, indicando um URI `<rdf:Description rdf:about="http://www.corais.com.br/animall.html">` que mostrará o recurso a ser descrito seguido da *tag* de abertura `<s:nome>` que declara quem é o animal, cujo valor da *string* objeto é "recife de coral".

O elemento `rdf:Description` é responsável pela descrição do assunto. O elemento `rdf:type` permite definir o modelo objetivando a aplicação dos conceitos de tipagem, provendo uma certa descrição do recurso. No ato da tipagem, é possível indicar o tipo do recurso que será abordado. Esse valor poderá ser numérico, texto, ou do tipo de um recurso

---

já existente. O caracter “#” indica fragmento, e referencia o aparecimento de um componente dentro do recurso. Assim, todas as propriedades que referenciavam o recurso passam a relacionar-se com a parte referenciada. O elemento `rdf:about` é utilizado como identificador do recurso.

### **3.2 – RDF Schema**

O RDF Schema (RDFS) é uma extensão do RDF para oferecer mecanismos para descrição de recursos e relacionamentos entre eles [RDF 2004]. Enquanto que, com RDF, as propriedades podem ser vistas como atributos dos recursos e relacionamentos entre eles, em RDFS pode-se descrever estas propriedades, e definir relações entre elas e outros recursos. O RDFS pode ser entendido como um dicionário que descreve os termos que poderão ser declarados em especificações RDF [RDFS 2004].

O RDFS é especificado como um conjunto de classes, propriedades e restrições entre seus relacionamentos, e essas classes são organizadas de forma hierárquica. Os recursos podem ser instâncias de uma ou mais classes, representados através da propriedade `rdf:type` [RDFS 2004].

A representação esquemática do RDFS tem uma analogia descritiva com relação aos modelos orientados a objetos de outras linguagens, apesar de apresentar certa hierarquia. Mas, pode-se ter uma noção da identificação dos componentes de RDFS com os componentes usualmente utilizados em modelos UML.

### **3.3 – Linguagens para definição de Ontologias**

A escolha da linguagem em que será descrita a ontologia é um dos passos mais importantes de sua construção, já que o poder de expressão de uma linguagem determina quais conhecimentos pode-se capturar no desenvolvimento.

Para construção de ontologias algumas linguagens de programação lógica podem ser utilizadas, entretanto linguagens baseadas em algum formalismo são mais comumente utilizadas.

---

A principal característica dessas linguagens está na capacidade de representar ontologias em RDF [RDF 2004] e RDF Schema [RDFS 2004], arquitetura já consagrada pela W3C [W3C 2005] para interoperabilidade de dados na Web [Moura 2001].

Nos próximos tópicos serão mostradas algumas linguagens para definição de ontologias, e a justificativa pela escolha, neste trabalho, da linguagem OWL.

### **3.3.1 – OIL (Ontology Inference Layer)**

A Ontology Inference Layer (OIL) é uma linguagem para a representação de ontologias que fornece a maior parte das primitivas de modelagem utilizadas em ontologias baseadas em *frames*. Possui semânticas simples, claras e bem definidas, baseadas na lógica de descrição (*Description Logic – DL*), e apresenta suporte para dedução automática [OIL 2000].

A linguagem OIL fornece uma série de construções comuns em ontologias definidas em lógica de descrição, e possui ferramentas de suporte que facilitam a edição e validação de ontologias. Uma ontologia OIL contém descrições para classes, relacionamentos (*slots*) e instâncias [OIL 2000]. Classes podem se relacionar com outras classes através de uma hierarquia (classes/ subclasses) e através de relações binárias estabelecidas entre duas relações. Além disso, restrições de cardinalidade podem ser atribuídas aos relacionamentos [OIL 2000].

A definição de uma ontologia em OIL é constituída de dois componentes: o primeiro, que descreve as características da ontologia (*ontology container*) utilizando-se de descrições do padrão *Dublin Core*; e o segundo (*ontology definitions*) que define o vocabulário particular daquela ontologia [OIL 2000].

OIL contém uma construção simples para modularizar ontologias. Este mecanismo é idêntico ao mecanismo de *namespaces* no XML e XML Schema. Para estruturar representações de ontologias complexas, seria necessário elaborar novos mecanismos, pois a linguagem OIL é atualmente restrita, somente as classes são fornecidas, não possui definições de meta-classes ou indivíduos.

---

### **3.3.2 - DAML + OIL**

A DAML + OIL é uma linguagem de marcação semântica para recursos Web, a qual estende os padrões RDF e RDF Schema [DAML+OIL, 2001].

Como o seu nome já indica, ela surgiu da combinação de outras duas linguagens, DAML-ONT e OIL, resultando em uma linguagem com uma semântica mais bem definida. DAML+OIL é uma linguagem baseada em RDF e em esquemas RDF com rica modelagem de primitivas. Ela é muito similar à especificação original do OIL, mas possui algumas diferenças [DAML+OIL, 2001]. Em particular, DAML+OIL abandonou os ideais originais baseados em *frames* do OIL, mostrando uma sintaxe alternativa para uma lógica de descrição.

Uma ontologia DAML+OIL é composta por vários componentes, sendo alguns opcionais e outros que podem se repetir por diversas vezes durante a definição da mesma [DAML+OIL, 2001].

Uma ontologia DAML+OIL consiste de zero ou mais cabeçalhos (*headers*), seguidos de zero ou mais classes (representadas pelo elemento *class*), propriedades (*property*) e instâncias (*instances*) [DAML+OIL, 2001].

As instâncias das classes e de propriedades são escritas de acordo com a sintaxe RDF e RDF Schema, e a DAML+OIL faz uso de vários tipos de elementos que auxiliam na definição dos conceitos e relações pertencentes ao domínio da ontologia.

### **3.3.3 - OWL (Web Ontology Language)**

#### **3.3.3.1 – Definição**

A OWL (Web Ontology Language) [OWL 2004] é uma linguagem de definição de ontologias, recomendada pela W3C, e é projetada para ser usada por aplicações que precisam processar o conteúdo de informação, em vez de apenas apresentar esta informação. Seu desenvolvimento teve início em março 2002, e consiste em uma revisão da DAML+OIL.

OWL facilita uma melhor interpretação de máquina de conteúdo da WEB, sendo suportada por XML, RDF, e RDF Schema, provendo vocabulário adicional junto com uma semântica formal.

---

A OWL é usada para explicitar as representações de significados de termos em vocabulários, e os relacionamentos entre estes termos [OWL 2004].

Para oferecer a interoperabilidade dos dados, duas condições são necessárias [OWL 2004]:

- Adoção de uma sintaxe comum: habilitando as aplicações a analisarem sintaticamente os dados. O XML provê esta característica de uma sintaxe comum.
- Adoção de meios para a compreensão da semântica: habilitando as aplicações a utilizarem o dado. A OWL provê uma forma padrão de expressar a semântica.

O problema chave em se conseguir interoperabilidade está em ser capaz de reconhecer que as duas partes do dado estão falando sobre a mesma coisa, mesmo que estejam utilizando diferentes terminologias. A OWL pode ser utilizada como uma ponte entre terminologias distintas, de modo que se possa compreender e obter a interoperabilidade dos dados [OWL 2004].

Atualmente, a OWL vem se destacando de outras tecnologias, devido ao fato de:

- *XML* prover uma sintaxe para documentos estruturados, mas não impor nenhuma semântica no significado destes documentos.
- *XML Schema* ser uma linguagem para restringir a estrutura de documentos XML e também estender XML com tipos de dados.
- *RDF* ser um modelo de dados para objetos (recursos) e relações entre eles, provendo uma semântica simples para este modelo de dados, podendo estes serem representados em uma sintaxe de XML.
- *RDF Schema* ser um vocabulário para descrever propriedades e classes de recursos de RDF, com uma semântica para generalização e hierarquias de tais propriedades e classes.
- *OWL* somar mais vocabulário para descrever propriedades e classes, relações entre classes, cardinalidade, igualdade, tipos mais ricos de propriedades, características de propriedades, e classes enumeradas.

---

### 3.3.3.2 – Uso e Sintaxe da OWL

OWL pode ser usada como uma ponte entre terminologias distintas, de modo que se possa compreender estas terminologias, provendo assim a interoperabilidade entre as mesmas [OWL 2004].

A sintaxe de OWL é baseada em XML e RDF, compatível com RDFS. Ela é uma revisão da linguagem DAML+OIL.

A OWL é um conjunto de elementos e atributos XML, com significados padronizados, que são utilizados para definir termos e seus relacionamentos. Ela estende o RDF Schema, ou seja, torna-o mais amplo, acrescentando outros elementos, como, por exemplo, *equivalentClass* e *TransitiveProperty* [OWL 2004].

A OWL possui três sub-linguagens: [OWL 2004]

- OWL *Lite*: Fornece um suporte básico aos usuários que necessitam apenas classificar uma hierarquia de classes e definições simples de propriedades. Por exemplo, a OWL *Lite* possibilita restringir cardinalidade para propriedades somente com os valores 0 ou 1, enquanto que as OWL DL e *Full* não possuem essa restrição. Dessa forma, ela facilita o desenvolvimento de ferramentas que lhe dêem suporte, visto que não apresenta grande complexidade para a implementação de suas funcionalidades.
- OWL DL: Fornece suporte aos usuários que necessitam de uma maior expressividade para o processamento. A OWL DL inclui todas as construções da linguagem OWL.
- OWL *Full*: Suporte mais complexo da linguagem, permitindo classificar e projetar classes com alto grau de complexidade.

A OWL Lite é um subconjunto da OWL DL, que por sua vez é um subconjunto da OWL Full.

Um documento OWL consiste de declarações de *namespaces*, um cabeçalho opcional referente à ontologia, e definições de classes, propriedades e indivíduos [OWL 2004].

Um *namespace* é utilizado para evitar colisões entre vocabulários definidos em documentos distintos, que, porventura, utilizem os mesmos

---

nomes para seus elementos e atributos. Ele é identificado por um URI (Universal Resource Identifier) [OWL 2004]. Ao iniciar uma ontologia OWL, é necessário declarar os *namespaces* que serão utilizados.

Um padrão inicial para uma ontologia informa um conjunto de declarações de *namespaces* XML incluído entre as *tags* `rdf:RDF`. Eles fornecem uma forma de se evitar a ambigüidade na interpretação de identificadores do documento, fazendo com que a ontologia se torne mais legível. O *namespace* é declarado através do atributo `xmlns` [OWL 2004].

Após a declaração de *namespaces*, é possível realizar a inclusão de uma coleção de afirmações acerca da referida ontologia, definindo, assim, o seu cabeçalho. Estas informações são agrupadas dentro da *tag* `owl:Ontology`. Estas *tags* suportam comentários, controle de versão e inclusão de outras ontologias. Através do elemento `owl:Ontology` declara-se uma coleção de metadados OWL acerca do documento [OWL 2004].

A utilização de ontologias depende da capacidade em raciocinar sobre as classes, as instâncias, objetos, indivíduos. Assim, é necessário um mecanismo para a descrição de classes e suas propriedades.

O mais básico dos conceitos acerca de um domínio corresponde às classes, que são as raízes das árvores de taxonomia. Cada objeto na linguagem OWL é um membro da classe `owl:Thing` por padrão, a qual já é predefinida na OWL. Dessa forma, cada classe definida pelo usuário é implicitamente uma subclasse de `owl:Thing` [OWL 2004]. As classes são declaradas através do elemento `owl:Class` e seus nomes são identificados pelo atributo `rdf:ID`. Podemos definir subclasses, com o elemento `rdfs:subClassOf`.

Após definirmos classes e subclasses em OWL, o próximo passo seria a definição de propriedades. As propriedades descrevem características e relacionamentos entre classes. A propriedade é uma relação binária [OWL, 2003]. OWL possui dois tipos de propriedades

- `owl:ObjectProperty`: é utilizada para relacionar uma classe com outra classe.
- `owl:DatatypeProperty`: é utilizada para especificar um tipo de dado, uma expressão literal ou um valor.

---

A Figura 3.3 mostra um exemplo de uma ontologia definida em OWL. Podemos ver declarações de *namespaces* da linha 2 a 7. A ontologia começa propriamente dita na linha 8, com a declaração de classes e subclasses nas linhas 9 e 10, respectivamente.

```
1. <?xml version="1.0"?>
2. <rdf:RDF
3.   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
4.   xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
5.   xmlns:owl="http://www.w3.org/2002/07/owl#"
6.   xmlns="http://www.owl-ontologies.com/unnamed.owl#"
7.   xml:base="http://www.owl-ontologies.com/unnamed.owl">
8.   <owl:Ontology rdf:about=""/>
9.     <owl:Class rdf:ID="Sedimentos">
10.      <rdfs:subClassOf>
11.        <owl:Class rdf:ID="Regiao_Bentica"/>
12.      </rdfs:subClassOf>
13.    </owl:Class>
...

```

**Figura 3.3:** Exemplo de código OWL

Devido a ser uma linguagem que está sendo amplamente utilizada, sua facilidade de uso e semelhança com XML, e por ser um padrão W3C, que suporta RDF, a linguagem OWL foi escolhida para ser utilizada nesse trabalho.

### 3.4– Editores de Ontologia

Editores de ontologia são ferramentas que proporcionam navegação, codificação e modificações de forma a facilitar as tarefas de construção e manutenção de ontologias.

Alguns editores de ontologias oferecem várias funcionalidades, como ferramentas de integração de informações, classificação lingüística e análise para auxiliar a extração de informações de fontes de dados com conteúdo não estruturado.

Os editores de ontologia variam consideravelmente em sua apresentação geral ao usuário. As habilidades de organizar e gerenciar uma ontologia emergente são pontos chave de usabilidade em um editor de

---

ontologias. A apresentação e manipulação conveniente e intuitiva dos conceitos e relações de uma ontologia são essenciais.

A seguir serão mostrados três editores de ontologias, e a justificativa pela escolha, neste trabalho, do editor Protégé.

#### **3.4.1 – OIEd**

O OIEd [OIEd 2002] foi criado pelo grupo de gerenciamento de informação da *University of Manchester*.

É um editor de ontologias de código aberto que permite construir ontologias utilizando as linguagens OIL e DAML+OIL. O modelo do conhecimento do OILED é baseado no do DAML+OIL, mas é estendido pelo uso de representações baseadas em *frames* para a modelagem. Então, OILED oferece um ambiente baseado no paradigma de *frames* para modelagem, enquanto ainda suporta a rica expressividade do DAML+OIL quando requerido [OIEd 2002].

No OILED, classes são definidas em termos de suas superclasses e restrições de propriedades, como axiomas adicionais capturando adicionalmente relacionamentos como a disjunção. O modelo de conhecimento expressivo permite o uso de composições de descrição complexas como regras de filtragem. Isto está em contraste com muitos editores baseados em *frames*, onde estes *frames* anônimos devem ser nomeados antes que sejam usados como modelos [OIEd 2002].

A versão corrente deste editor não provê um ambiente completo para o desenvolvimento de ontologias. Não tem um suporte para o desenvolvimento de ontologias em larga escala, a verificação de consistência e classificação automática da ontologia não são suportadas pelo editor, tendo assim que ser usada outra ferramenta em paralelo para a realização destas tarefas.

#### **3.4.2 - OntoEdit**

Desenvolvida pela Universidade *Karlsruhe*, na Alemanha. Tem como características o desenvolvimento de projeto e manutenção de ontologias, suporte a ontologias multilingual, modelo de representação baseado em *frames*, porém não suporta reutilização [Sure *et.al* 2002].

---

É um ambiente gráfico para edição de ontologias que permite inspeção, navegação, codificação e alteração de ontologias. O modelo conceitual é armazenado usando um modelo de ontologia que pode ser mapeado em diferentes linguagens de programação. As ontologias são armazenadas em bancos de dados relacionais, e podem ser implementadas em XML, *FLogic*, RDF, RDFS e DAML+OIL [Sure *et.al* 2002].

### 3.4.3 – Protégé

O Protégé [Protégé 2005] é uma ferramenta de software integrada usada para desenvolvimento de sistemas baseados em conhecimento. Aplicações desenvolvidas com Protégé são usadas em resolução de problemas relacionados à tomada de decisão, sendo este ambiente caracterizado por oferecer diversos recursos gráficos que facilitam a construção de bases de conhecimento em qualquer área.

Protégé foi desenvolvida, inicialmente na década de 80 pelo departamento de informática médica da *University of Stanford*, sendo amplamente utilizada nessa área nos dias atuais. Ela está em constante desenvolvimento e, atualmente, oferece suporte à Web Semântica através da utilização de um *plugin* para a OWL [Protégé 2005].

A ferramenta Protégé é um software livre e código aberto. Ela fornece diversos *plugins* de modo a lhe dar maiores características e fornecer suporte às necessidades dos programadores. Há a possibilidade de conversão em vários formatos como XML, XMLSchema, RDF, RDFS, *Topic Maps*, UML, OIL, DAML+OIL, OWL, entre outros, fornecendo, assim, maior flexibilidade à aplicação.

O Protégé tem sua instalação livre, é desenvolvido em Java, necessitando, assim, do JVM (Java Virtual Machine) e está disponível para diversas plataformas, caracterizando sua interoperabilidade [Protégé 2005].

Uma ontologia no Protégé consiste basicamente em [Protégé 2005]:

- Classes: são as entidades do domínio a ser modelado;
- Propriedades: Descreve propriedades de classes e instâncias, bem como as relações;
- Instâncias: Declarações de uma determinada classe.

O Protégé conta com um ambiente gráfico e interativo para o projeto de ontologias e bases de conhecimento. Isto ajuda os especialistas no domínio a realizarem suas tarefas. Os desenvolvedores de ontologias podem acessar informações importantes quando necessário, e pode-se usar manipulação direta para se administrar uma ontologia. Controles em árvore permitem uma navegação simples e rápida através da hierarquia de classes.

Em adição a interface com boa usabilidade, pode-se destacar ainda duas qualidades: escalabilidade e extensibilidade. Uma das maiores vantagens da arquitetura do Protégé é que o sistema é construído em uma forma aberta e modular. Esta arquitetura baseada em componentes permite aos construtores do sistema a adição de novas funcionalidades com a criação de *plugins*.

Em termos de variedade de características, especialmente as relativas às interfaces com outros sistemas de informação, o Protégé, versão 3.0, é um dos editores mais completos, além de contar com *plugins*, e suporte à linguagem OWL [Protégé 2005], características que determinaram a escolha pelo uso deste editor nesse trabalho.

A Figura 3.4 mostra a interface do Protégé, com *plugin* OWL na criação da ontologia.

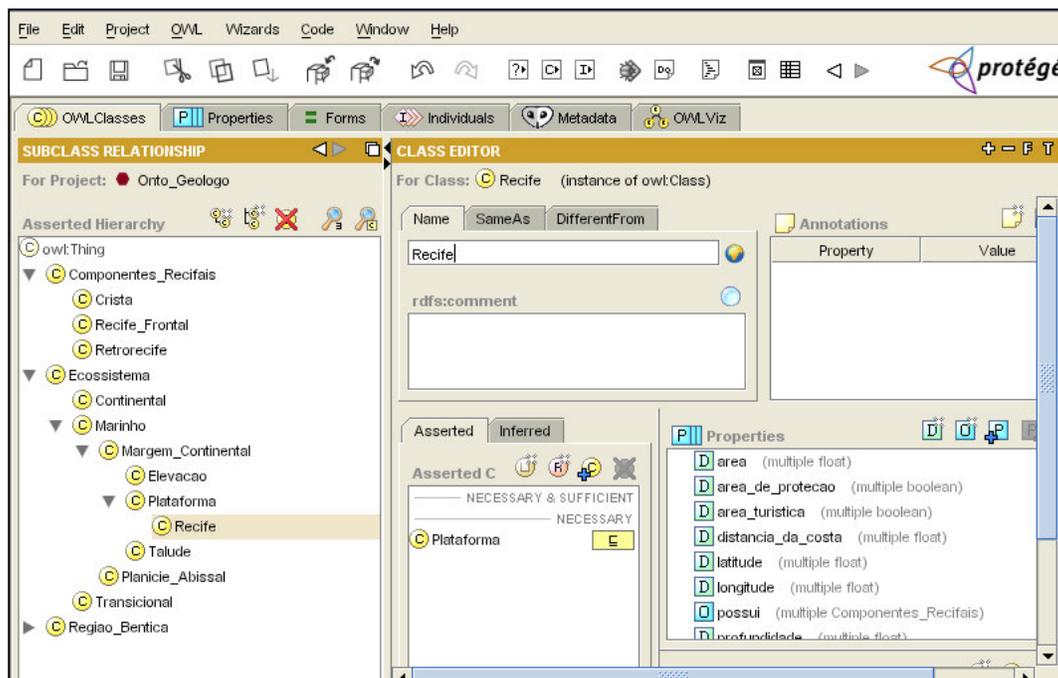


Figura 3.4: Editor Protégé

---

## 3.5 – Implementação de Ontologias

Uma das características das ontologias que são descritas usando o OWL-DL é que elas podem ser processadas por uma máquina de inferência, muitas vezes denominadas de *Reasoner*. Com as ontologias descritas formalmente (através da linguagem OWL DL, e utilizando o editor Protégé), o próximo passo é a validação desta ontologia utilizando um *Reasoner* para checar a consistência das classes na ontologia [Protégé 2005]. Com as ontologias verificadas e inferidas, o passo agora é a implementação de uma camada, que faria o intermédio entre as ontologias, em OWL, e a aplicação. Para isso, será mostrado o framework Jena [Jena 2004], um conjunto de classes Java que possuem métodos implementados para manipulação e tratamento de ontologias.

### 3.5.1 – Racer

*Reasoner* é uma máquina de inferência utilizada para checar a consistência das classes em uma ontologia. Um dos serviços principais oferecidos por uma máquina de inferência é testar se uma classe é ou não uma subclasse de uma outra classe. Executando tais testes em todas as classes de uma ontologia, é possível para um *Reasoner* computar a hierarquia inferida das classes da ontologia. Um outro serviço oferecido é verificar a consistência da ontologia.

Racer [Racer 2005] é um *plugin* de um *Reasoner* que deve ser executado junto com o editor Protégé. A ontologia pode ser “enviada ao *Reasoner*” para automaticamente computar a hierarquia de classificação, e para verificar também a consistência lógica da ontologia, inferindo classes e conceitos provenientes da ontologia.

### 3.5.2 – Framework Jena

Jena é um projeto *open-source* desenvolvido pelo *HP Labs Semantic Web Programme* [Jena 2004]. Consiste em um *framework* Java para construção de aplicações de web semântica.

A API Jena provê classes para lidar com RDF, RDFS e OWL, através da manipulação de grafos que descrevem o conteúdo semântico de um objeto. É possível fazer várias operações nos grafos de ontologias, entre

---

elas: navegar no grafo (procurando pelo URI de um recurso, procurando pelo URI de uma propriedade, listando as propriedades de um recurso e seus valores), alterar o grafo (adicionando recursos e declarações sobre eles), remover recursos (um recurso é removido quando não há triplas que o contenha) consultar o grafo (podemos realizar consultas simples (por triplas)).

A API Jena caracteriza-se por possibilitar a criação e manipulação de grafos RDF, representadas pelos recursos (*resource*), propriedades (*properties*) e literais (*literals*), formando as tuplas que irão dar origem aos objetos criados em JAVA. Assim, esse conjunto de objetos é usualmente chamado de *model*. O *model* pode ser considerado como o conjunto de *statements* que forma o grafo por completo.

Para criação de um modelo, a API Jena dispõe de vários pacotes que permitirão a interação com o RDF. Alguns pacotes [Jena 2004]:

- *com.hp.hpl.jena.datatypes*: Contém um conjunto de interfaces para a descrição de tipos de dados
- *com.hp.hpl.jena.datatypes.xsd*: Provê a definição de tipos de dados compatíveis com Xml Schema
- *com.hp.hpl.jena.ontology*: Pacote composto por um conjunto de interfaces e classes para acesso e manipulação de grafos de ontologias
- *com.hp.hpl.jena.reasoner*: Provê suporte para a inferência das ontologias
- *com.hp.hpl.jena.model*: Contém a interface *model*, que representa um modelo. Um modelo de ontologia é uma extensão do modelo RDF e fornece capacidades para a manipulação de ontologias.

Os modelos de ontologia são criados pelo *Jena ModelFactory*, utilizando a classe *OntModel* proveniente do pacote *jena.ontology*. *OntModel* suporta os tipos de objetos que são esperados em uma ontologia: classes (ou hierarquia de classes), propriedades e instâncias [Jena 2004].

O grafo gerado pela API é obtido através da classe *ModelMem* que, armazenado em memória, fica disponível até a finalização da execução do programa. Assim, durante essa execução, pode-se percorrer e buscar os

---

elementos armazenados para criação, modificação ou exclusão do conteúdo de um arquivo RDF [Jena 2004].

Após a criação de um grafo vazio, para manipulação de um modelo RDF, precisa-se de recursos para constituírem o modelo. Após serem instanciadas as classes que permitem realizar a transformação do documento em memória, os objetos criados terão a flexibilidade de transformar o modelo de acordo com as necessidades do cliente [Jena 2004]. A Figura 3.5 mostra a criação de um modelo RDF, utilizando a API Jena.

```
static String animalURI    = " http://www.corais.com.br ";
static String Nome        = " Recife de coral ";

// criando um modelo vazio
Model modelo = ModelFactory.createDefaultModel();

// criando o recurso
Resource recifeCoral = modelo.createResource(animalURI);

// adicionando a propriedade
recifeCoral.addProperty(animal, Nome);
```

**Figura 3.5:** Código Java representado um modelo RDF, utilizando a API Jena

### 3.6 - Conclusão

Com o crescimento das pesquisas sobre integração de dados geográficos, várias tecnologias, anteriormente ligadas a outras áreas da computação (como por exemplo, inteligência artificial), têm sido incorporadas ao desenvolvimento de BDG, auxiliando no desenvolvimento de aplicações multidisciplinares.

Neste capítulo, apresentamos tecnologias para construção e manipulação de ontologias, que serão usadas no decorrer do capítulo seguinte, onde apresentaremos a proposta de solução de uma camada semântica, que irá utilizar linguagem OWL para construção de ontologias, o *reasoner* Racer para validação das ontologias, e o *framework* Jena, para construção da camada semântica, utilizando métodos e classes Java.

---

# **Capítulo 4**

## **Arquitetura Proposta**

### **4.1 – Modelagem Conceitual e Banco de Dados Geográfico**

Um Banco de Dados Geográfico (BDG) é um banco de dados não-convencional, que além de armazenar dados alfanuméricos, é capaz de armazenar e recuperar dados geográficos.

Através das modelagens conceitual, lógica e física de dados geográficos, os fenômenos e entidades geográficos são selecionados para serem armazenadas no banco de dados. O modelo de dados deve produzir uma visão abrangente da realidade, mostrando todos os fatores envolvidos e todos os relacionamentos, existentes ou não, entre eles. Além disso, estes fatores e relacionamentos deverão ser armazenados no BDG. A interação de SIG com BDG tem o intuito de mudar o desenvolvimento da tecnologia SIG, permitindo a transição dos atuais sistemas monolíticos para uma nova geração de aplicativos geográficos, onde tanto os atributos como as geometrias de dados espaciais sejam gerenciadas pelo Sistema Gerenciador de Banco de Dados [Cunha 2005].

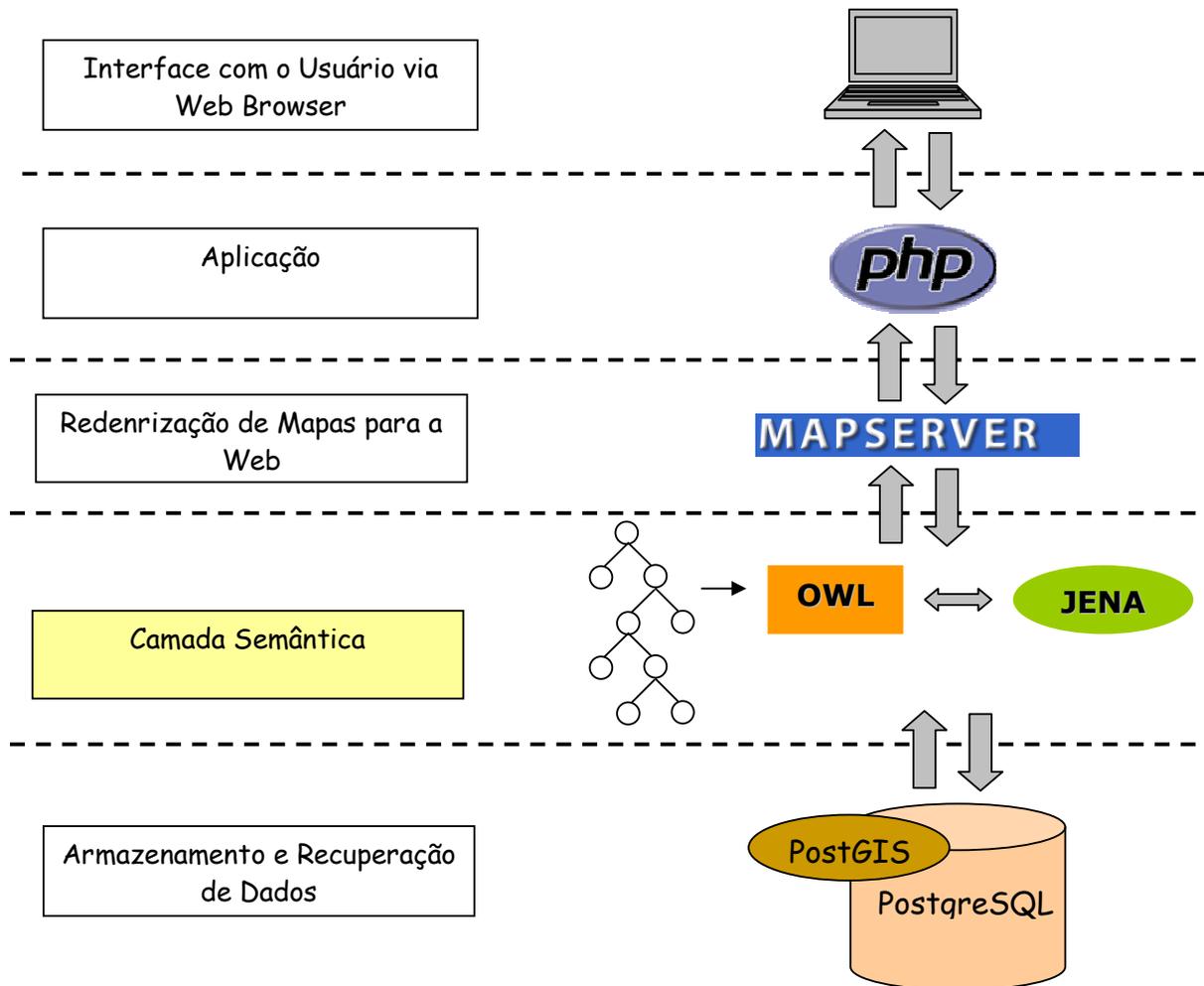
Nossa proposta é de uma solução baseada em ontologias geográficas para a interação das informações. A idéia é propiciar, em um mesmo BDG, diferentes tipos de informação, na medida em que profissionais de áreas distintas utilizam o mesmo. A partir dos conceitos armazenados nas ontologias existentes em nossa aplicação, iremos inferir os conceitos e informações necessárias às consultas, provendo assim informação suficiente às diferentes comunidades de usuários que irão interagir com o sistema [Viegas and Soares 2006].

Para a interação do BDG com as ontologias, desenvolvemos uma camada semântica, que irá interagir com o BDG, e será criada a partir do processamento de ontologias que representam o conhecimento das

---

comunidades de usuários e das ontologias que representam conhecimento do BDG.

A camada semântica é formada pelas ontologias, em OWL, e por classes Java que irão fazer o tratamento e busca destas ontologias. A Figura 4.1 mostra a adaptação de uma camada semântica à aplicação de Cunha [Cunha 2005]. Esta camada irá coletar as consultas inseridas no código PHP e intermediá-las, antes que elas cheguem ao banco de dados.



**Figura 4.1:** Adaptação de uma camada semântica ao SIG de Cunha [Cunha 2005]

#### 4.1.1 - Ontologias Geográficas

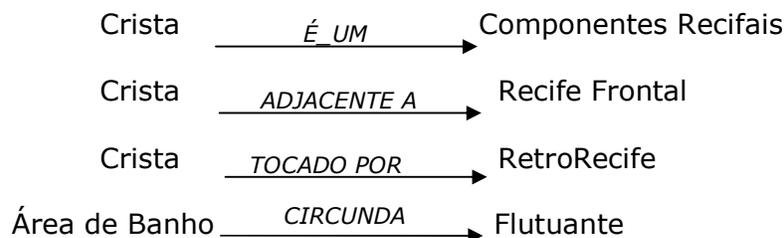
Na vasta literatura da área [Guarino 1998] [Fonseca 2001] [Breitman and Casanova 2005] existem muitas definições a respeito de ontologias. Em algumas destas referências percebe-se a tentativa de se explicar o que é exatamente uma ontologia geográfica. Muitas vezes a mesma é tratada

---

simplesmente como uma ontologia, cujo domínio é geográfico. Mas, neste trabalho, acrescentamos algumas especificações que definem e determinam o escopo de uma ontologia geográfica. Para nós, esta ontologia deve incluir classes típicas de feições geográficas e relações geográficas (topológicas, distância e direção).

Em nosso trabalho, para cada ontologia definida, criamos relacionamentos usando a característica de propriedades, de modo a relacionar espacialmente um conceito com outro, de acordo com as operações básicas definidas no Modelo *9-Intersection* [Egenhofer and Herring 1991] e adotadas na especificação do Consórcio OpenGIS [OGC 1999].

Isto pode ser verificado em alguns exemplos dados a seguir, onde se percebe que um conceito pode se relacionar 'espacialmente' com outro, tais como:



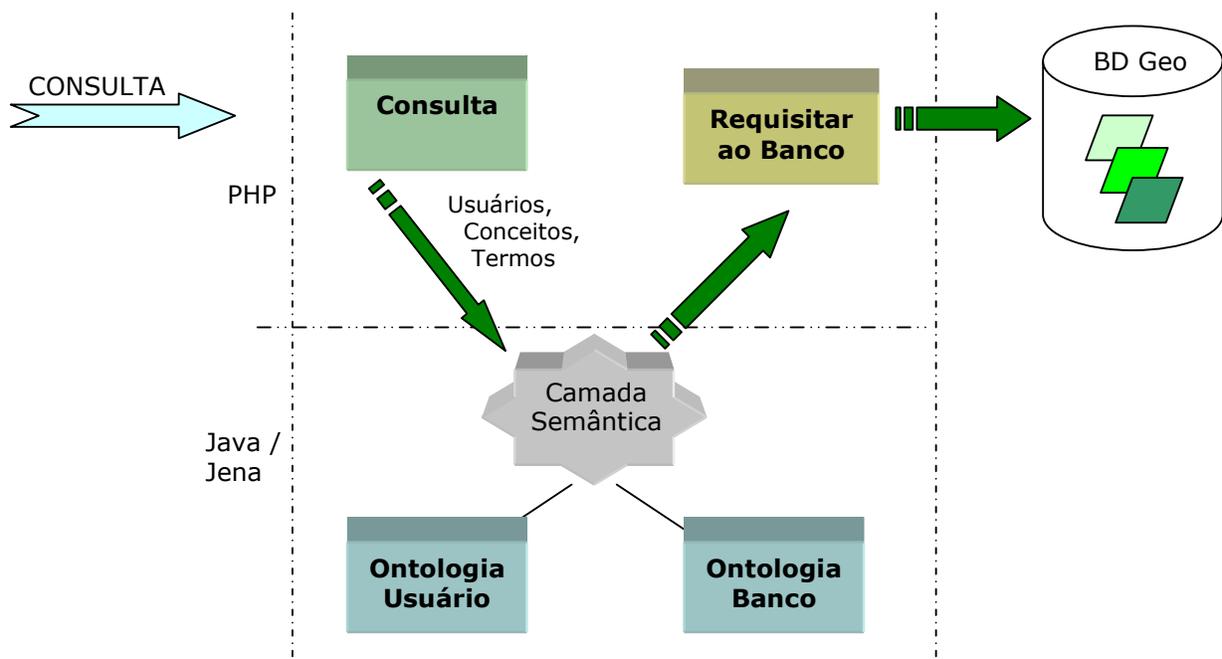
Em geral, os relacionamentos espaciais de topologia definidos nas ontologias possuem um significado ligado aos operadores definidos no modelo *9-Intersection*, como, por exemplo, *Tocado Por* (nas ontologia) refere-se ao operador *Touches* (modelo *9-Intersection*).

Nós não trabalhamos com instâncias nas ontologias, todos os valores dos dados estão armazenados no BDG. No entanto, as informações espaciais dos relacionamentos e propriedades das ontologias podem ser úteis na solução de algumas consultas, como também no complemento de outras.

Através de janelas *pop-up*, a cada consulta submetida, nossa aplicação também fornece um dicionário, contendo informações detalhadas (descritivas e espaciais) sobre as instâncias do BDG envolvidas na mesma, bem como nos seus relacionamentos. Assim, objetivo do uso dos relacionamentos espaciais é poder dar suporte ao usuário em suas consultas, informando a este, conceitos sobre as quais ele está

pesquisando. Além disso, para os usuários cuja ontologia difere da que efetivamente reflete o conteúdo do banco de dados, ele irá receber a equivalência dos termos utilizados em sua consulta com os termos que efetivamente foram retornados em sua consulta, enriquecendo o seu conhecimento com outras áreas.

Uma arquitetura geral do sistema é mostrada na Figura 4.2. As consultas realizadas pelos usuários, antes de chegar ao banco de dados, serão intermediadas pela camada semântica, que irá processar a informação e descobrir os termos semelhantes. Essa intermediação é feita utilizando métodos da API Jena [Jena 2004], que será utilizada para construir os grafos das ontologias. O pacote *jena.ontology* [Jena 2004], possui classes *OntClass* e *OntModel* que permitem que componentes de um grafo sejam acessados de várias maneiras.



**Figura 4.2:** Arquitetura Geral do sistema

O processamento se dá da seguinte maneira: a consulta, com seus termos chave, é coletada na interface e montada para interagir com as ontologias. Montar a consulta significa reconhecer qual o usuário que está

---

interagindo com o sistema no momento, os conceitos e termos que foram utilizados na consulta, e a consulta propriamente dita. Com a consulta em mãos, o próximo passo será procurar os termos e conceitos na ontologia do usuário em questão, e comparar essa ontologia com a ontologia do BDG, procurando por conceitos equivalentes. Através das URLs das ontologias, a API Jena [Jena 2004] será utilizada para construir grafos das ontologias.

## **4.2 – Construção da Camada Semântica**

O primeiro passo para a obtenção da camada semântica é a construção das ontologias de cada comunidade que irá interagir com a aplicação.

A partir da proposta desenvolvida neste trabalho, diversas comunidades de usuários poderão consultar o mesmo BDG de forma transparente, ou seja, sem necessitar ter conhecimento da seu esquema conceitual. Além disso, as ontologias irão auxiliar na inferência de informações (um dos fatores que diferenciam uma ontologia de uma modelagem conceitual). Através dos grafos das ontologias, podemos inferir informações, recuperando uma série de conceitos relevantes.

A criação de uma camada semântica no SIG fará com que cada comunidade de usuário, ao interagir com o sistema, utilize apenas as informações relevantes à sua área de conhecimento e obtenha os resultados que estão efetivamente armazenados no banco. As ontologias ficarão armazenadas nesta camada e irão interagir com o banco de dados, buscando os conceitos necessários. A camada semântica irá atuar na adaptação de informações (adaptar a informação de acordo com o vocabulário do usuário em questão) e também na filtragem de informações (informações que não são úteis para o usuário em questão não serão apresentadas ao mesmo).

Com as ontologias criadas, o próximo passo é o mapeamento entre as classes das diferentes ontologias. No mapeamento de ontologias tem-se como resultado uma estrutura formal com expressões que ligam os termos de uma ontologia aos termos de uma outra ontologia. Fizemos um mapeamento de classes sinônimas, para viabilizar a localização das classes que possuem informações relevantes para atender às consultas requisitadas

---

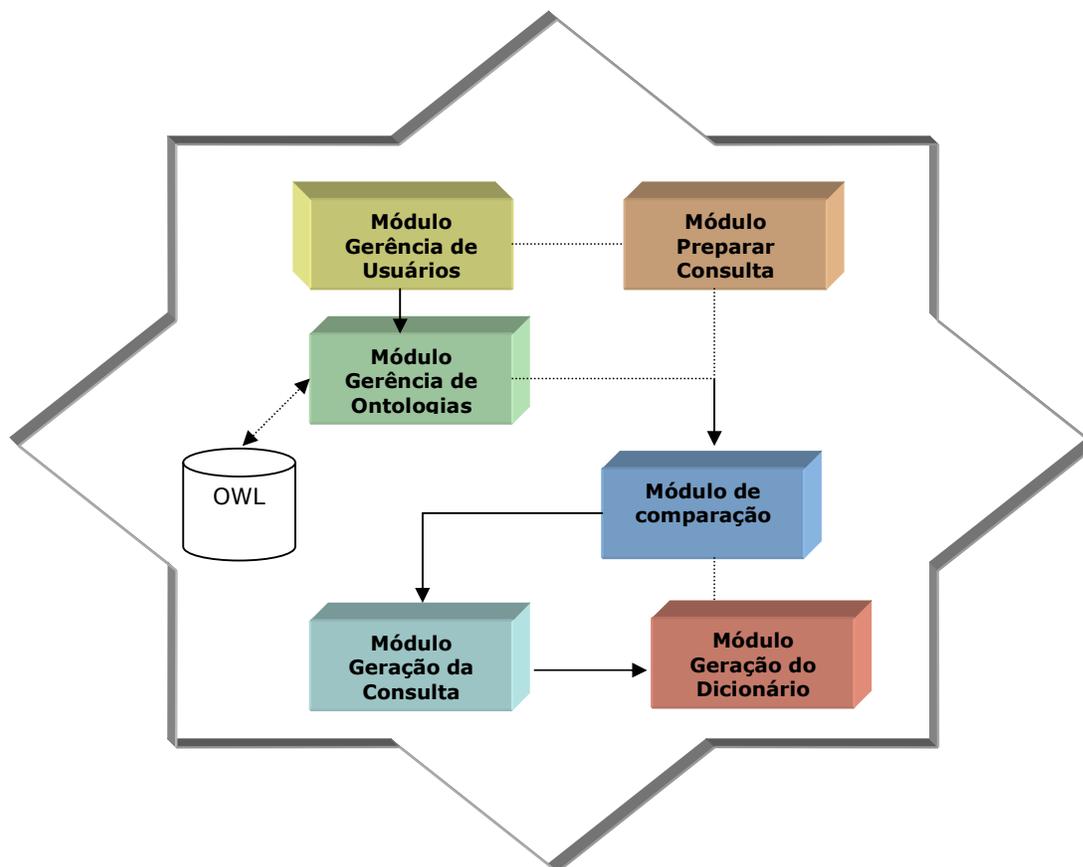
ao BDG. No nosso caso, classes sinônimas são aquelas que possuem conceitos com um mesmo significado, mas que podem estar rotulados com nomes diferentes de acordo com o conhecimento específico de cada comunidade.

Depois das ontologias serem mapeadas, há a necessidade delas serem manipuladas. Ou seja, é necessário percorrer os termos de cada ontologia procurando por conceitos semelhantes. A API Jena [Jena 2004] foi utilizada para construir grafos das ontologias, sendo possível a criação e manipulação de grafos RDF, representadas pelos recursos, propriedades e literais.

A partir de métodos específicos da API Jena, conseguimos manipular e comparar as ontologias. Detalharemos a camada semântica no tópico a seguir.

#### 4.2.1 - Detalhamento da Camada Semântica

Um detalhamento da camada semântica é visto na Figura 4.3. Os módulos desta camada são:



**Figura 4.3:** Módulos da camada semântica

---

➤ **Módulo Gerência de Usuários:**

O primeiro passo para realizar a consulta em nossa aplicação é informar que tipo de usuário está interagindo com o sistema. Para isso, foi incorporado ao sistema desenvolvido por Cunha [Cunha 2005] um *módulo gerência de usuários*.

Com este módulo, o usuário poderá escolher com qual tipo de interface ele deseja interagir com o sistema. Cada comunidade de usuário terá uma interface pré-definida de consulta. Essa interface específica foi criada de modo que o usuário possa interagir com o sistema de forma transparente, utilizando apenas o conhecimento específico à sua área de atuação, contendo apenas os termos relativos à ontologia do usuário em questão.

➤ **Módulo Gerência de Ontologias:**

Com a escolha do tipo de usuário e a interface de consultas pré-definida, será ativado o próximo módulo da camada semântica, que irá ser responsável pela ativação das ontologias. Em nossa aplicação, as ontologias ativadas serão sempre as do usuário em questão, bem como a do BDG.

No momento da ativação, são aplicados métodos da API Jena para construir os grafos (modelos) das ontologias.

A API Jena possui classes de objetos para representar grafos, recursos, propriedades e literais. Um grafo é chamado de modelo, e é representado pela interface *Model*. Esses modelos irão guardar as ontologias ativadas pela aplicação. Com as ontologias modeladas em forma de grafos, podemos fazer uma varredura procurando pelos termos desejados.

➤ **Módulo Preparar Consulta**

No momento em que o usuário informa a maneira pela qual ele quer interagir com o sistema, as ontologias são ativadas e modeladas, e a interface pré-definida de consulta é mostrada.

Com a interface pré-definida, o usuário irá formular sua consulta, marcando aquilo que ele deseja pesquisar. Com isto, o próximo módulo da camada semântica será ativado. Este é o *módulo preparar consulta*. Este módulo irá coletar os termos chave da consulta

---

➤ **Módulo de Comparação:**

Um dos módulos mais importantes de nossa aplicação é o *módulo de comparação*. Ele é o responsável pela busca de termos semelhantes nas ontologias.

A ligação entre os conceitos equivalentes é representada na linguagem OWL através das *tags equivalentClass* e *sameAs*, no entanto há uma diferença entre as duas *tags*. A *tag* `<owl:equivalentClass>` é utilizada para indicar que duas classes são equivalentes se, e somente se, possuírem, precisamente, as mesmas instâncias. Já a *tag* `<owl:sameAs>` é utilizada quando temos diferentes nomenclaturas que se referem a uma mesma classe. Um uso típico da *owl:sameAs* é para unificação de ontologias, ou seja, para dizer que dois indivíduos, definidos em documentos diferentes, são iguais [OWL 2006]. Assim, utilizamos a *tag* `<owl:sameAs>`.

Ao encontrar as *tags* que definem semelhança conceitual entre as classes, o próximo passo do *módulo de comparação* é utilizar métodos da API Jena para tratar estas semelhanças, ligando classes de uma ontologia a outra.

Fizemos uma comparação entre as classes que são sinônimas nas diferentes ontologias. Definimos classes sinônimas como aquelas que possuem conceitos com um mesmo significado, mas rotulados com nomes diferentes; ou com diferenças na escrita como, por exemplo: um termo no plural e outro no singular.

A Figura 4.4 mostra parte do código Java, onde criamos um método para encontrar a semelhança entre uma classe em OWL e o elemento passado como parâmetro (no nosso caso, esse elemento é a palavra chave da consulta do usuário). O método *getSameAs()*, da interface *OntResource*, é utilizado para encontrar as classes semelhantes nas ontologias

```
private String getSameElement (OntModel model, String element) {
    OntResource ontRes = model.getOntResource (element);
    OntResource ontResSA = ontRes.getSameAs ();
    if (ontResSA == null) return null;

    return ontResSA.getLocalName ();
}
```

**Figura 4.4:** Parte do método utilizado para a pesquisa de semelhanças conceituais

---

➤ **Módulo Geração da Consulta:**

O *módulo geração de consulta* irá montar a consulta que será submetida ao banco de dados. Para isto, serão utilizados os termos encontrados na busca pela semelhança, bem como o relacionamento passado na interface de consulta.

➤ **Módulo Geração do Dicionário:**

A consulta será retornada para o usuário, localizando no mapa o que ele pesquisou, além de uma janela explicativa, mostrando todas as informações sobre a sua pesquisa.

O *módulo geração do dicionário* irá montar um texto explicativo, com os termos chave da consulta, fornecendo ao usuário descrições detalhadas sobre cada conceito pesquisado.

### **4.3 Conclusão**

Este capítulo apresentou a construção de uma camada semântica que, aplicada ao sistema de Cunha [Cunha 2005], auxilia na consulta de diferentes comunidades de usuários, que irão interagir com o BDG.

A partir das ontologias construídas, mostramos o funcionamento dos módulos da arquitetura da camada semântica, que utilizam classes Java da API Jena para construir métodos de acesso e comparação das ontologias, e permitir o acesso ao banco de dados, independente dos termos técnicos utilizados pelas diferentes comunidades.

No próximo capítulo serão mostrados cenários de consultas aplicados em um estudo de caso real, com os resultados.

---

## **Capítulo 5**

### ***Protótipo da Aplicação e Cenários de Consulta***

#### **5.1 – Protótipo da Aplicação: Domínio de Recifes de Corais**

Para a validação da camada semântica, iremos utilizar a modelagem conceitual e o banco de dados geográfico desenvolvido em [Cunha 2005]. Inicialmente, no trabalho de Cunha [Cunha 2005] foi elaborado o modelo conceitual dos Recifes de Corais, e o BDG foi desenvolvido tendo como base os Parrachos de Maracajaú, localizado no litoral do Rio Grande do Norte. Foi realizado um estudo da APARC (Área de Proteção Ambiental Estadual dos Recifes de Corais), que foi criada pelo governo do estado do Rio Grande do Norte através do decreto nº. 15.746 de 06 de junho de 2001 e está inserida entre as coordenadas UTM: N-9.443.000 m; S-9.349.m; L-266.000m; O-224.000m [IDEMA 2004]. Também fazem parte da APARC os corais de Cioba e Rio do Fogo, que não foram estudados para a aplicação. A Figura 5.1 ilustra a localização da APARC, destacando a área estudada em detalhes pelo Projeto Corais de Maracajaú.

A modelagem de Cunha [Cunha 2005] assume a divisão da Terra em três grandes grupos principais de ecossistemas: continental, marinho e transicional. Onde, as terras emersas correspondem ao ambiente continental, o ambiente transicional pode ser definido como a área de intercessão dos sistemas continental e marinho, e os ambientes submersos encontrados após o transicional correspondem ao ambiente marinho o que pode ser visto na Figura 5.2. A região de estudo, Parrachos de Maracajaú, é um ecossistema marinho localizado sobre uma região de plataforma continental [Cunha 2005].



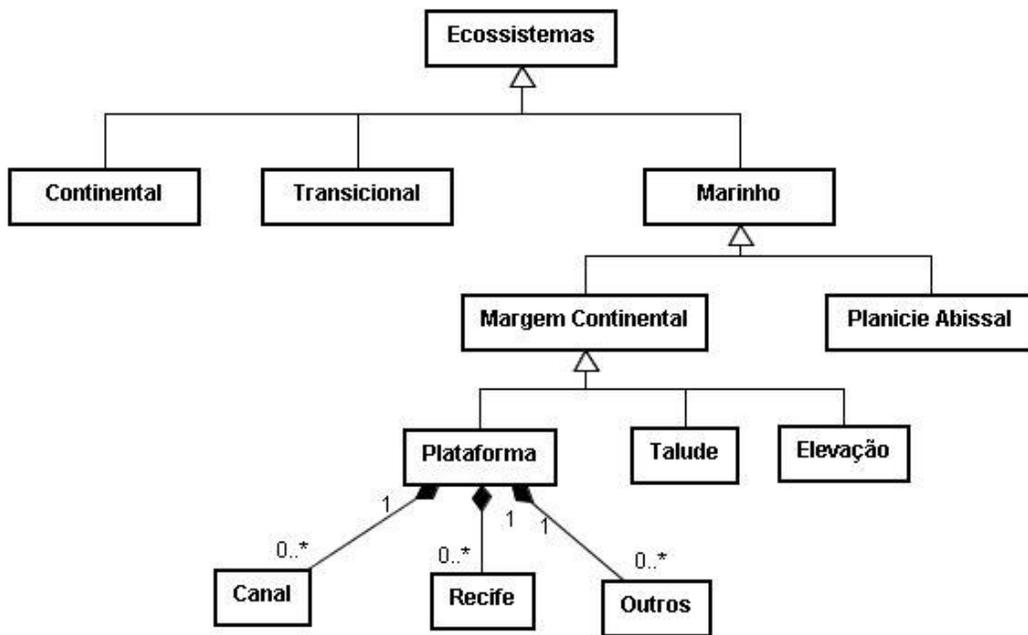
**Figura 5.1:** Área de Proteção Ambiental dos Recifes de Corais [Feitoza 2005]

No modelo conceitual do banco de dados, o ecossistema marinho é dividido com base em critérios geomórficos em: Margem Continental e Planície Abissal. Partindo da margem continental, o modelo preocupa-se em definir somente as zonas de interesse para o escopo do trabalho na região de estudo. Define-se a plataforma continental, onde o diagrama destaca canal e recife, sendo recife a classe de interesse a ser trabalhada e refinada no modelo [Cunha 2005].

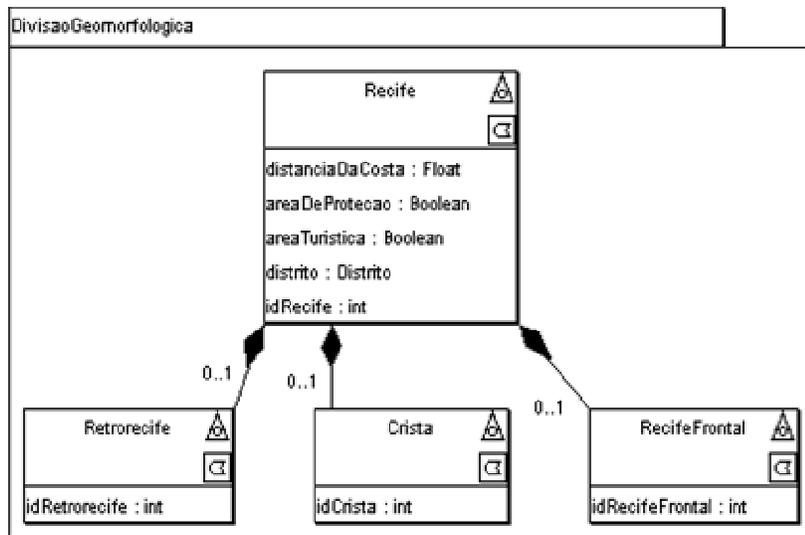
Na divisão geomorfológica, a área é modelada de acordo com o seu formato geomórfico e pode ser classificado em três diferentes zonas: Retrorecife, Crista e Recife. O retrorecife ou "*backreef*" é a região do recife voltada para a costa que normalmente sofre menos com o impacto das ondas devido à proteção exercida pela crista e o recife frontal. A crista do

recife ou "reef crest" é a região, onde o corpo coralíneo fica parcialmente emerso nos períodos de baixa mar. Já o recife frontal ou "fore reef" localiza-se na porção exterior do recife, no lado oposto à costa, absorvendo grande parte da energia das ondas [Cunha 2005]. Cada parte do recife é modelada como um objeto geográfico, com seus atributos e localização.

Podemos ver uma parte da modelagem conceitual na Figura 5.2, onde temos uma visão de alto nível contextualizando os corais de recifes da região de Maracajaú. Outra parte do modelo conceitual pode ser encontrado na Figura 5.3, que mostra o modelo proposto para o corpo do recife de acordo com a sua divisão geomorfológica.



**Figura 5.2:** Visão da região de Maracajaú em relação aos ambientes terrestres



**Figura 5.3:** Modelagem Conceitual da Divisão Geomorfológica

A Figura 5.4 mostra a região APARC fotografada, indicando os locais dos recifes de corais.



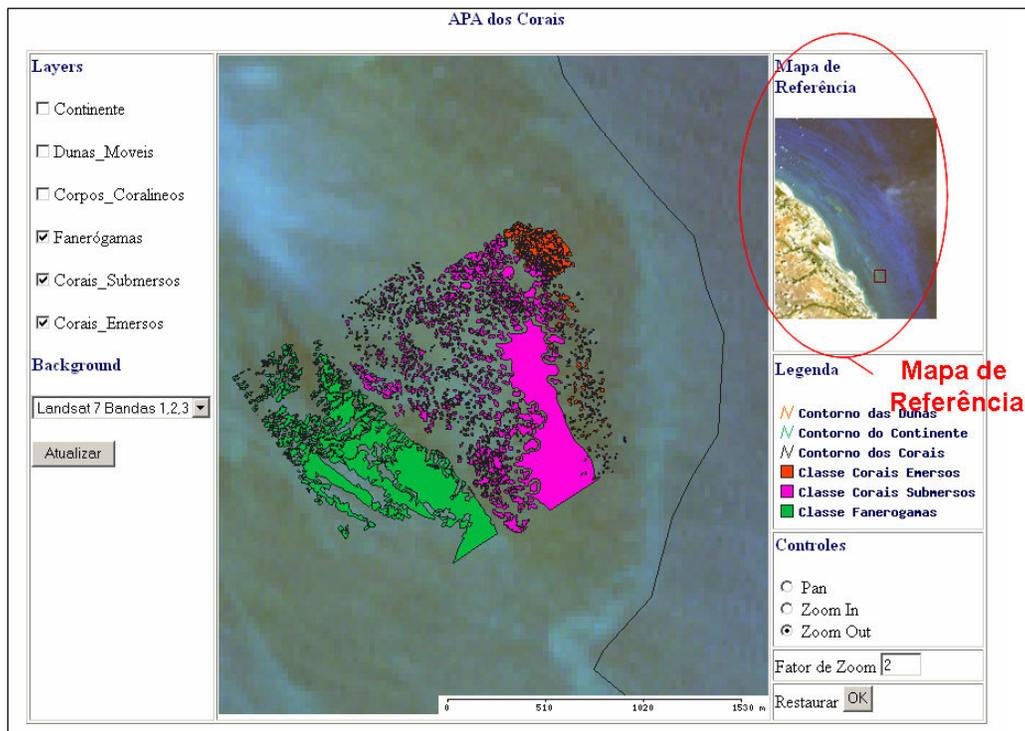
**Figura 5.4:** Região da APARC fotografada

O Sistema desenvolvido por Cunha [Cunha 2005] está baseado nas seguintes tecnologias:

- PostgreSQL/PostGIS: Banco de Dados para armazenamento e recuperação de informações geográficas;
- MapServer: Ferramenta para a redenzificação dos mapas - "map engine";

- PHP: Linguagem especialmente desenvolvida para a construção de aplicações Web.

A Figura 5.5 mostra um "screenshot" ilustrativo da aplicação APA dos Corais [Cunha 2005].



**Figura 5.5:** SIG APA dos Corais [Cunha 2005]

Mais informações sobre o desenvolvimento deste banco de dados, bem como sobre o modelo conceitual desenvolvido para este domínio de aplicação podem ser obtidos em [Cunha 2005].

## 5.2 - Contextualização do Problema

Como foi mostrado anteriormente, o domínio geográfico deste trabalho são os recifes de corais. Este domínio foi escolhido devido à existência de um projeto de pesquisa da Universidade Federal do Rio Grande do Norte que envolve pesquisadores de diferentes áreas de atuação, na região de corais dos Parrachos de Maracajaú [Amaral 2002]. Sob o

---

domínio deste projeto, foram desenvolvidos vários trabalhos acadêmicos dentre os quais o de Cunha [Cunha 2005] e o de Cabral [Cabral 2006].

Devido a existir basicamente três comunidades de usuários dentro deste projeto, foi desenvolvido, através do uso de ontologias, três diferentes visões deste mesmo domínio: a dos geólogos, dos biólogos e a dos turistas.

A necessidade deste projeto inicialmente partiu da comunidade de geólogos, que concentrava grandes conhecimentos sobre a região, mas não tinha conhecimento da área de ciência da computação. Desta forma, a modelagem do banco de dados foi um trabalho árduo precisando de freqüentes entrevistas de modo a se chegar a um consenso. Ao ser implementado o projeto, porém, surgiu a necessidade de se disponibilizar as informações do sistema por meio de uma página Web. Com a página disponível na Web, surgiu a dificuldade de outras comunidades de usuários usarem o sistema e entenderem a terminologia usada pelos geólogos e implementada no banco de dados. A solução dada a este impasse foi então através do uso de ontologias geográficas [Viegas and Soares 2006].

Cada profissional pode ter uma visão diferente sobre os recifes de corais. Um geólogo, ao interagir com o SIG, busca conceitos que interessem à sua área de conhecimento. Por exemplo, a divisão geomorfológica, os habitats bênticos da região, as zonas do ecossistema, os critérios geomórficos, a formação do recife de coral. Já um biólogo pode interagir com o mesmo SIG que o geólogo interage, mas com conceitos diferentes, que interessam somente à sua área de conhecimento. Neste caso, é interessante procurar nos recifes de corais os animais que ali habitam, a formação das algas, os tipos de minerais, etc. E, ainda podemos ter um recife de coral do ponto de vista de um turista. Para o turista, interessam apenas o lazer, as atrações, belezas naturais, possibilidade de mergulhos e navegação, como exemplo.

O recife de coral deste domínio de aplicação é único, mas a interpretação da informação que ele contém pode ser diferente para cada área de conhecimento. Este é um dos grandes problemas quando um mesmo SIG é utilizado por diversos profissionais, de diversas áreas. Um único conceito pode ser definido com diferentes termos. O detalhamento de alguns conceitos específicos pode ser interessante para determinada

---

comunidade de usuários, mas para outra pode não ser. Em alguns casos, os geólogos necessitam de informações que os biólogos não estão interessados. Em outros, ambos necessitam da mesma informação, mas em diferentes formatos.

### **5.3 – Ontologias Desenvolvidas**

Desenvolvemos uma ontologia que representa o conhecimento do BDG, e outras ontologias que especificam a base de conhecimento dos pesquisadores do projeto. Essa ontologia que representa o banco de dados é a visão da comunidade de geólogos sobre os recifes de corais. O BDG [Cunha 2005] utilizado nesta proposta permite recuperar informações dos recifes de corais utilizando apenas os termos dos geólogos que foram implementados no mesmo.

Para este trabalho, construímos três ontologias, referentes às áreas de conhecimento de três comunidades: geólogo (deste ponto em diante será referenciada como (*O<sub>geo</sub>*), turista (*O<sub>tur</sub>*) e biólogo (*O<sub>bio</sub>*)).

Cada comunidade irá possuir suas ontologias, que irão se comunicar umas com as outras através da semelhança conceitual das classes.

#### **5.3.1 - Ontologia do Banco de Dados (Geólogo)**

Para a criação das ontologias, foram analisados vários fatos, levantados na fase da entrevista ao profissional da área em questão, e foram selecionados os conceitos e seus relacionamentos.

As ontologias são formadas por um conjunto de conceitos (classes). Um conceito se relaciona com outro através de uma propriedade. Cada conceito pode ter textos explicativos, que detalham o seu significado (representado em OWL através da *tag* `rdfs:comment`).

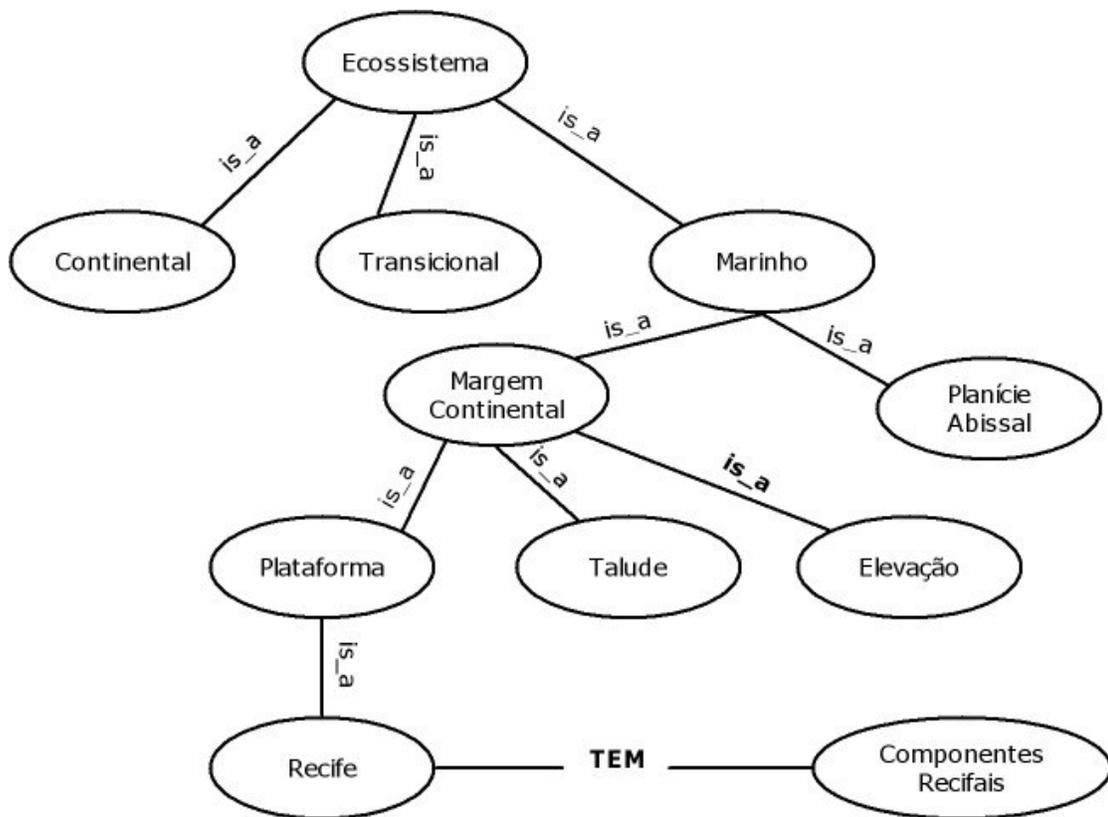
A *O<sub>geo</sub>* representa os conceitos relacionados aos aspectos geológicos, ou seja, conceitos que um geólogo irá procurar no sistema, ao interagir com este.

A *O<sub>geo</sub>* começa definindo o ambiente em que o recife de coral está localizado. Sabe-se que a localização de um objeto geográfico é um fator determinante. A classe `Ecossistema` é a raiz do grafo, e é dividida nas

classes `Continental`, `Transicional` e `Marinho`. Com estes relacionamentos, temos a informação, através do grafo, que o ambiente marinho, por exemplo, é um ecossistema. Continuando a ontologia, temos a divisão geomorfológica do ambiente marinho, em `Margem Continental` e `Planície Abissal`. Podemos ver, assim, que a margem continental é um ambiente marinho. A margem continental é dividida em `Plataforma`, `Talude` e `Elevação`.

O recife, por sua vez, é uma plataforma. Note que este é um caminho para termos uma visão geral da localização do recife.

Podemos ver parte da ontologia `Ogeo` na Figura 5.6.

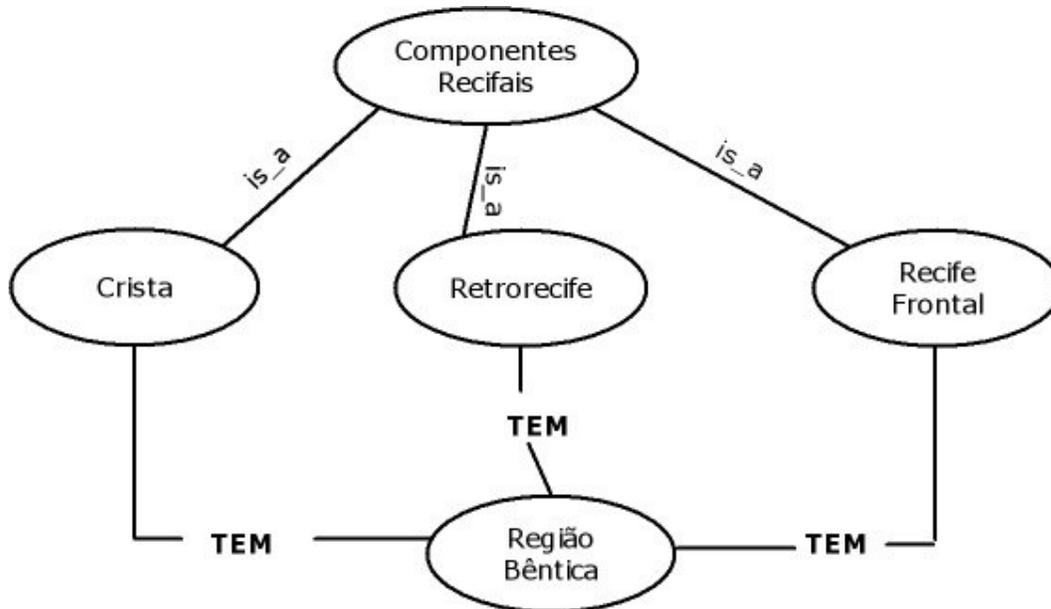


**Figura 5.6:** Parte da Ontologia `Ogeo`

Continuando com nossa ontologia, o recife de coral possui três componentes recifais, separados, que são adjacentes uns dos outros. Essas classes são `Retrorecife`, `Crista` e `Recife Frontal`. A `Crista` é tocada pelo `Recife Frontal`, e este é tocado pelo `Retrorecife`. Assim, podemos notar que

---

a Crista é um componente recifal. Outra parte da **O<sub>geo</sub>** é mostrada na Figura 5.7.

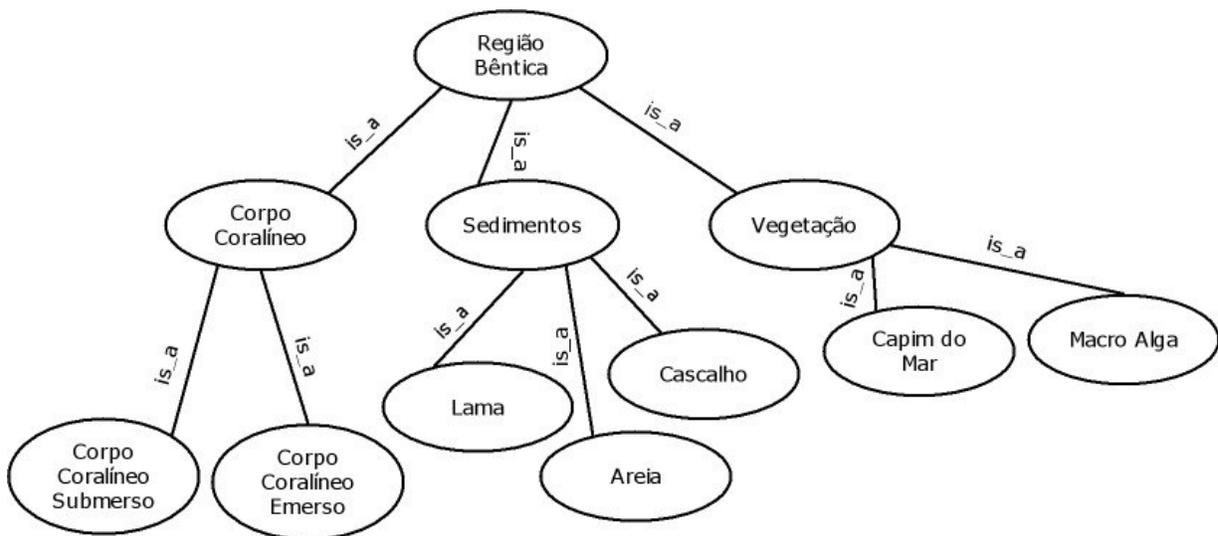


**Figura 5.7:** Ontologia dos Componentes Recifais

O recife é dividido, de acordo com uma divisão bêntica, em Sedimentos, Vegetação Submersa e Corpo Coralíneo. Esta região bêntica, por sua vez, aparece em quaisquer das três partes dos componentes dos recifes, ou seja, na crista, retrorecife e recife frontal.

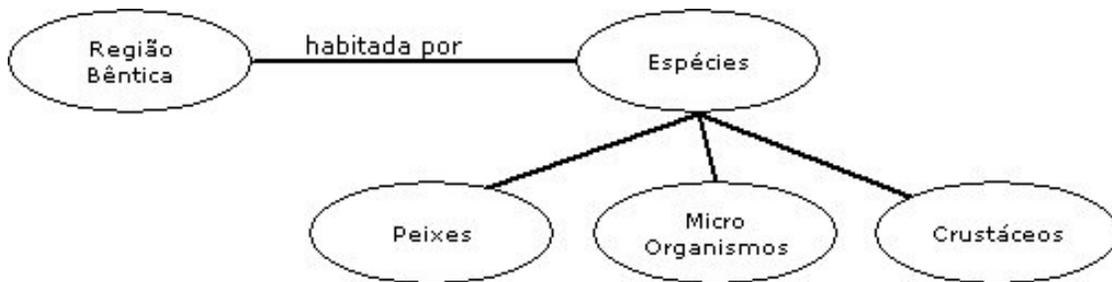
Partindo da classe *Sedimentos*, temos uma divisão de acordo com a granulometria, nas classes *Lama*, *Areia* e *Cascalho*. Podemos inferir, por exemplo, que a lama é um sedimento; e um sedimento é uma região bêntica.

A classe *Vegetação Submersa* é dividida em *Capim do Mar* e *Macro Alga*, e a classe *Corpo Coralíneo*, em *Corpo Coralíneo Submerso* e *Corpo Coralíneo Emerso*. Podemos ver a ontologia da região bêntica da **O<sub>geo</sub>** na Figura 5.8.



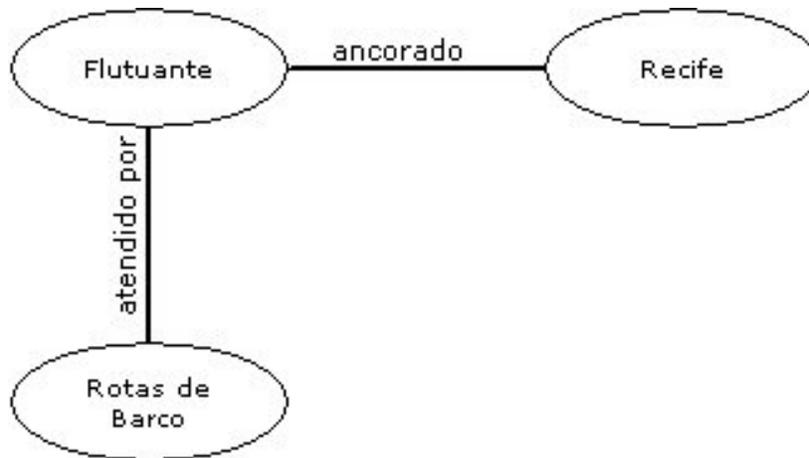
**Figura 5.8:** Ontologia da Região Bêntica

Temos ainda outras classes que estão ligadas ao Recife. A Região Bêntica está ligada com a classe Espécies, através da propriedade *habitado\_por*. Podemos definir as espécies como sendo: crustáceos, peixes e microorganismos. Todas estas espécies *habitam* a região bêntica. Podemos ver estas classes na Figura 5.9



**Figura 5.9:** Definição da classe Espécies, da Ogeo.

Continuando a modelagem das classes na ontologia, temos ainda os objetos que podem estar perto ou presos ao Recife. Na Figura 5.10, modelamos o Flutuante, que está *ancorado* no Recife. O Flutuante é *atendido\_por* Rotas de Barco.



**Figura 5.10:** Continuação da Ontologia da Comunidade de Geólogos, definindo os flutuantes e rotas de barco.

Com a ontologia  $O_{geo}$  completa e com as classes interligadas através de propriedades, podemos coletar informações, unindo os conceitos. Por exemplo, da  $O_{geo}$  podemos ter informações que o Recife é uma plataforma; cada recife possui como componentes a crista, o retrorecife e o recife frontal; cada componente tem uma região bântica, que é uma vegetação submersa, corpo coralíneo e sedimentos; lama, cascalho e areia são sedimentos; capim do mar e macro alga são vegetações. As espécies (algas, crustáceos, entre outros) habitam a região bântica. Podemos também coletar relacionamentos geográficos, como por exemplo: a crista *toca* o recife frontal; o retrorecife *toca* a crista.

Uma parte do código OWL é mostrada na Figura 5.11. Podemos ver, na linha 1, a definição da classe *crista*; na linha 2, o código mostra que a classe *crista* é subclasse da classe *componentes recifais*; as linhas 3 e 4 mostram a *tag* `owl:disjointWith`, que estabelece que um indivíduo que seja um membro de uma classe não pode simultaneamente pertencer a outra classe. Assim, vemos que a classe *crista* é disjunta das classes *recife frontal* e *retrorecife*, ou seja, que se um elemento estiver na classe *crista*, ele não pode estar nem na classe *recife frontal* e nem na classe *retrorecife*. Salientamos que está é uma propriedade da linguagem OWL, e que difere do operador espacial “disjuncto”. As linhas 5, 6, 7 e 8 são linhas de comentário. As linhas 10 e 11 mostram a criação de uma propriedade, referenciando que a *crista* **tem** *região bântica*.

```

1. <owl:Class rdf:about="#Crista">
2.   <rdfs:subClassOf rdf:resource="#Componentes_Recifais"/>
3.   <owl:disjointWith rdf:resource="#Recife_Frontal"/>
4.   <owl:disjointWith rdf:resource="#Retrorecife"/>
5.   <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
6.     >Região, entre o retrorecife e o recife frontal, onde o corpo
7.   coralíneo fica parcialmente emerso nos períodos de baixa mar.
8.   </rdfs:comment>
9. </owl:Class>
10. <owl:ObjectProperty rdf:ID="tem">
11.   <rdfs:domain rdf:resource="#Regiao_Bentica"/>

```

**Figura 5.11:** Parte do código OWL, da ontologia *O<sub>geo</sub>*

### 5.3.2 - Ontologia Turista

A *O<sub>tur</sub>* representa os conceitos relacionados aos aspectos turísticos da região, ou seja, conceitos que irão interessar a comunidade de turistas, ao interagir com o sistema.

Semelhante a *O<sub>geo</sub>*, começamos definindo o ambiente no qual o recife de coral está localizado.

O *Ecosistema* é dividido em *Continente*, *Praia* e *Mar*. No ponto de vista de um turista, no mar encontram-se a *Região de Corais* e as *Piscinas Naturais*. A região de corais é definida como um multipolígono, contendo vários pequenos corais (corpos coralíneos), que unidos formam toda a região. No caso específico dos corais de Maracajaú (escopo deste trabalho), esta região é denominada de Parrachos. As piscinas naturais estão localizadas *dentro da* região de corais, e ao redor de cada corpo coralíneo. Podemos ter uma idéia da definição *Região de Corais* e *Piscinas Naturais* através da Figura 5.12. Nesta figura, podemos ver a região de corais, composta por pequenos corais, e rodeada de piscinas naturais. Também foi adicionado um sinônimo para *Região de Corais* para os mais leigos, com a denominação simples de "pedras".



**Figura 5.12:** Região dos Parrachos de Maracajaú.

Na região de corais, existem Pedras mais Rasas, Pedras Visíveis e Pedras mais Fundas. Para os geólogos, esta divisão corresponde a retrorecife (local que sofre menos impacto das ondas), crista (local onde podemos ver corais emersos) e recife frontal (local que sofre mais impacto das ondas).

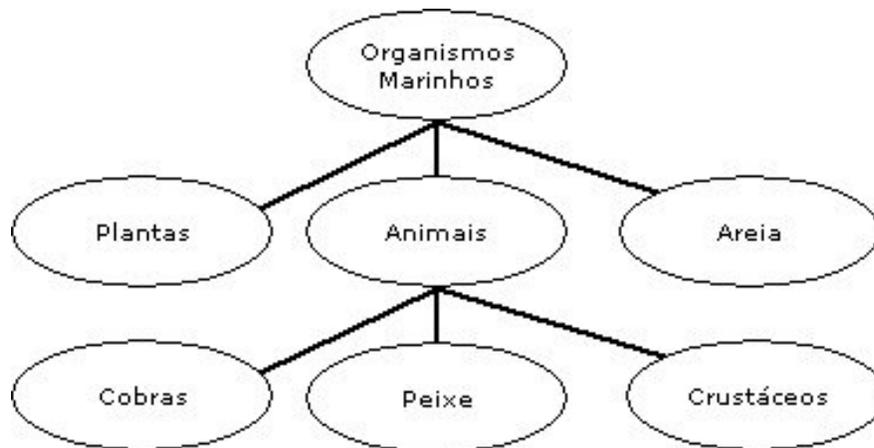
Os Organismos Marinhos habitam a Região de Corais. Estes organismos marinhos correspondem a região bêntica da ontologia dos geólogos. Podemos ver essa primeira parte da O<sub>tur</sub> na Figura 5.13.



**Figura 5.13:** Parte da ontologia O<sub>tur</sub>, com conceitos referentes à comunidade turista.

---

Os organismos marinhos são divididos em Plantas e Algas, Animais Marinhos e Areia. Animais marinhos são os Peixes, Cobras e Crustáceos. Podemos ver a continuação da  $O_{tur}$  na Figura 5.14.



**Figura 5.14:** Continuação da  $O_{tur}$ , definindo os organismos marinhos.

Continuando a definir conceitos referentes à comunidade de turistas, referente aos conceitos das atrações turísticas da região dos corais. Estas atrações foram também definidas em uma segunda parte da Ontologia do Turista. Assim, podemos mostrar ao turista não só conceitos sobre a região de corais, como também atrações turísticas da região.

Definimos uma Atração Turística como sendo Navegar, Mergulhar, Nadar e Banho. Caso o turista deseje navegar, esta classe tem ligação com a classe Rotas de Barco, da  $O_{geo}$ , e serão feitas inferências para unir os conceitos destas ontologias. As classes que estão pontilhadas, na Figura 5.15, dizem respeito às classes pertencentes a outras ontologias, mas que possuem ligação com a  $O_{tur}$ . Continuando a  $O_{tur}$ , caso o turista deseje mergulhar, a ontologia mostrará a Exploração de Organismos Marinhos; nadar, será mostrado a Região sem Corais, e banho será mostrado as Piscinas Naturais e os Flutuantes. Podemos ver esta segunda parte da ontologia referente à comunidade de turistas na Figura 5.15.

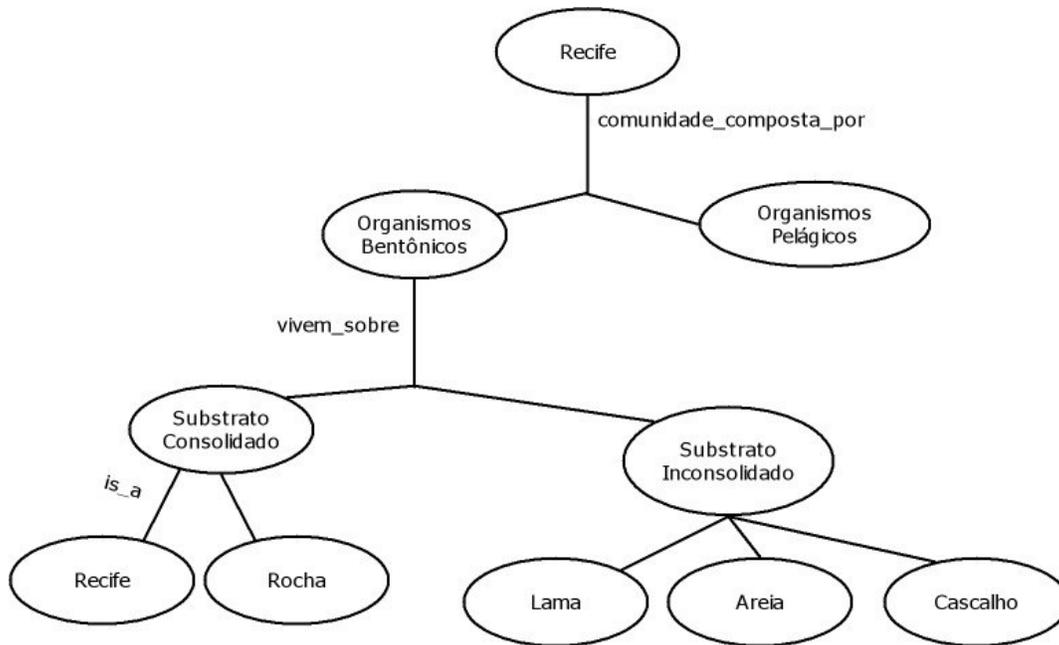


**Figura 5.15:** Ontologia sobre atrações turísticas.

### 5.3.3 - Ontologia Biólogo

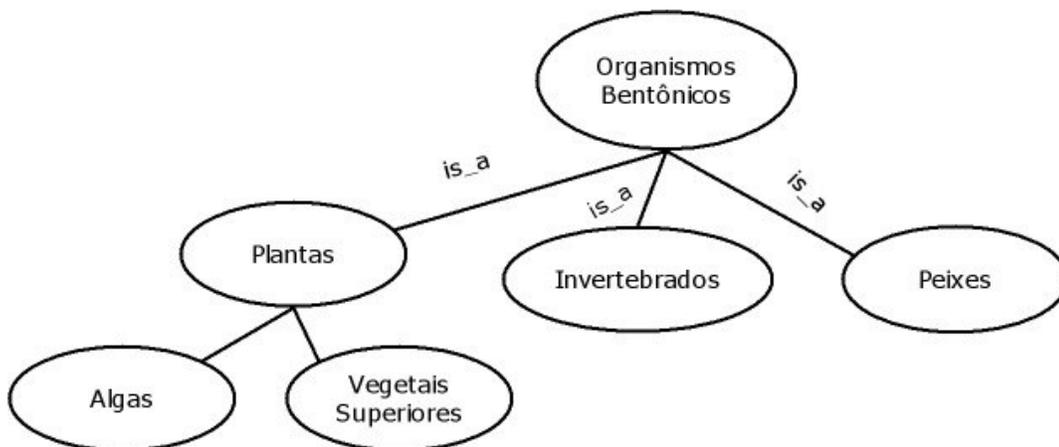
A *O<sub>bio</sub>* representa os conceitos relacionados à visão da comunidade de biólogos, ao interagir com o SIG.

No ponto de vista dessa comunidade, um *Recife* é uma comunidade composta por *Organismos Bentônicos* e *Organismos Pelágicos*; os organismos bentônicos vivem sobre *Substratos Consolidados* e *Substratos não Consolidados*. Podemos ver esse detalhamento na Figura 5.16.

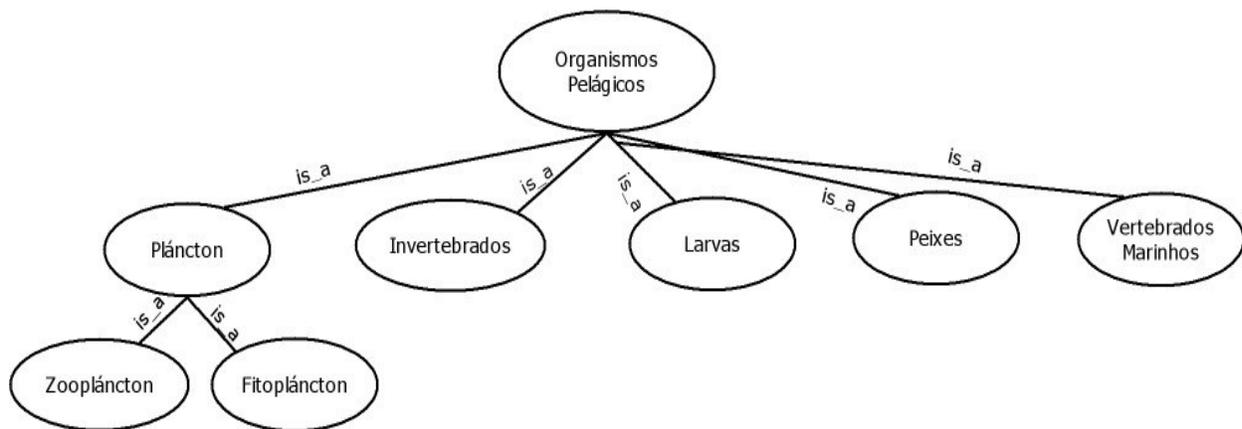


**Figura 5.16:** Definição de parte da *Obio*

Continuando a *Obio*, a classe *Organismos Bentônicos* é dividida em *Plantas*, *Invertebrados* e *Peixes*; e a classe *Organismos Pelágicos*, dividida em *Plâncton*, *Invertebrados*, *Larvas*, *Peixes* e *Vertebrados Marinhos*, como mostram, respectivamente, as Figuras 5.17 e 5.18.



**Figura 5.17:** Parte de componentes da comunidade bêntica, da *Obio*



**Figura 5.18:** Parte de componentes da comunidade pelágica, da **O<sub>bio</sub>**

Todas as ontologias foram criadas com o auxílio do editor Protégé, utilizado a linguagem OWL [OWL 2004].

#### 5.3.4 - Interação entre as Ontologias

Todas as ontologias desenvolvidas e apresentadas nas seções anteriores se relacionam umas com as outras. É importante mostrar quais relacionamentos existem entre elas, de modo a ser mais fácil o entendimento das inferências da camada Semântica.

A Figura 5.19 mostra os relacionamentos entre as ontologias do geólogo (identificada na figura pela cor preta) e a do turista (identificada pela cor vermelha). Percebem-se nesta figura que as classes de uma ontologia se relacionam umas com as outras, como podemos notar, por exemplo, a classe Praia, na **O<sub>tur</sub>** possui relação com a classe Transicional, na **O<sub>geo</sub>**.

A Figura 5.20 mostra os relacionamentos entre as três ontologias (**O<sub>geo</sub>**, **O<sub>tur</sub>**, **O<sub>bio</sub>**). A ontologia do geólogo está identificada com a cor preta, a do turista com a cor vermelha, e a do biólogo, azul. Podemos ver nesta figura, por exemplo, que a classe Sedimentos, da **O<sub>geo</sub>**, possui ligação com as classes Substrato Não Consolidado (**O<sub>bio</sub>**), e Areia (**O<sub>tur</sub>**). Salientamos que ainda há outros relacionamentos, e que colocamos no texto apenas algumas classes para exemplificar.



**Figura 5.19:** Relacionamento entre as classes das ontologias do geólogo e turista

---

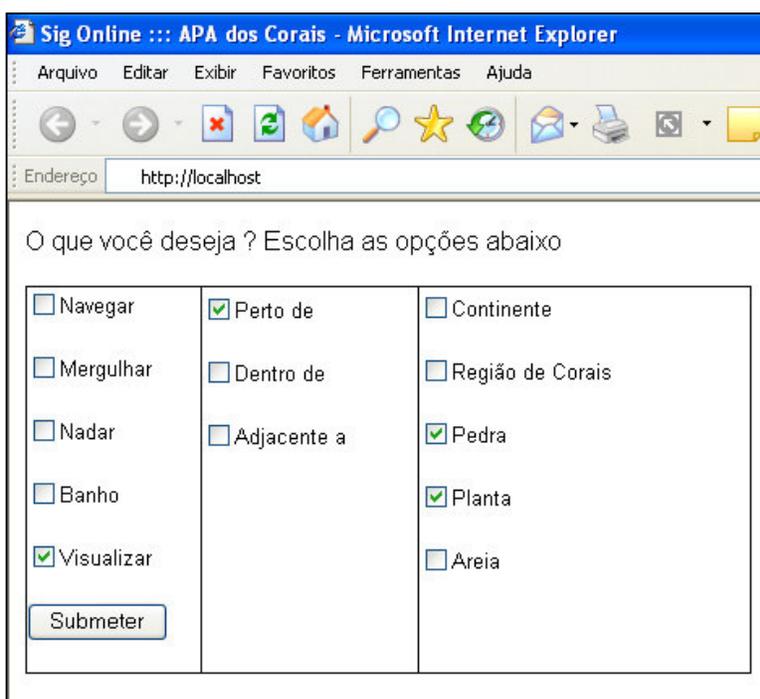
**Figura 5.20:** Relacionamento entre as classes das três ontologias presentes na aplicação

---

## 5.4 – Protótipo da Aplicação

### 5.4.1 - Executando uma Consulta Espacial de Distância

Criamos para este cenário um exemplo onde um usuário da comunidade de turistas faz uma consulta ao nosso sistema: “Quero ver um lugar onde tenha plantas perto de pedras”. Essa escolha do usuário é limitada a uma interface de consulta elaborada previamente, como mostra a Figura 5.21.



The image shows a screenshot of a Microsoft Internet Explorer browser window. The title bar reads "Sig Online :: APA dos Corais - Microsoft Internet Explorer". The address bar shows "http://localhost". The main content area contains the text "O que você deseja ? Escolha as opções abaixo" followed by a form with three columns of checkboxes. The first column has "Navegar", "Mergulhar", "Nadar", "Banho", and "Visualizar" (checked). The second column has "Perto de" (checked), "Dentro de", and "Adjacente a". The third column has "Continente", "Região de Corais", "Pedra" (checked), "Planta" (checked), and "Areia". A "Submeter" button is located at the bottom left of the form.

**Figura 5.21:** Protótipo da Interface de consulta pré-definida

Se esta consulta fosse levada diretamente ao banco de dados, não haveria nenhum retorno para o usuário, pois os termos “planta” e “pedra” não estão inseridos no banco de dados.

Embora estes termos não estejam inseridos no banco de dados, existem as informações “vegetação\_submersa” e “corpo\_coralíneo”, que são termos técnicos, e que possuem semelhança conceitual com os termos “planta” e “pedra”, respectivamente. Estas semelhanças conceituais, bem como as outras semelhanças em nossas ontologias, foram encontradas após entrevistas realizadas com profissionais de cada área de atuação do nosso escopo.

---

Dessa maneira, há a necessidade de que a consulta, antes de chegar ao BDG, seja intermediada por uma camada semântica, que irá processar a informação, descobrindo, por exemplo, que o termo “planta” é equivalente ao conceito do termo “vegetação\_submersa”. O termo “vegetação\_submersa” está armazenado no banco de dados, e, assim, será retornado para o usuário todos os locais que possuem esta vegetação.

Em nosso cenário, vimos que um turista deseja realizar a consulta. Assim, o *módulo gerência de ontologias* irá ativar a  $O_{tur}$  e a  $O_{geo}$ . As ontologias estão armazenadas em um servidor de ontologias, e são acessadas através de suas URLs.

A Figura 5.22 mostra a criação de dois modelos, utilizando a API Jena que possui classes de objetos para representar grafos, recursos, propriedades e literais. Um grafo é chamado de modelo, e é representado pela interface *Model*. Esses modelos irão guardar as ontologias ativadas pela aplicação ( $O_{geo}$  e  $O_{tur}$ ). Com as ontologias modeladas em forma de grafos, podemos fazer uma varredura procurando pelos termos desejados.

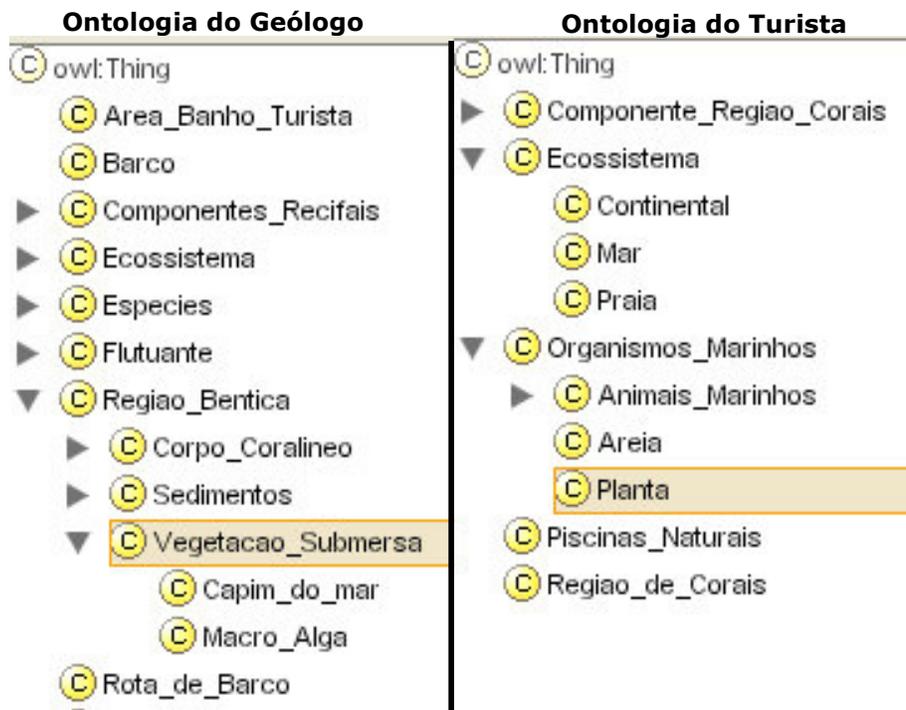
```
1. Model model1 = ModelFactory.createDefaultModel();
2. Model model2 = ModelFactory.createDefaultModel();

3. OntModel OMTurista = ModelFactory.createOntologyModel(spec, model1);
4. OntModel OMGeologo = ModelFactory.createOntologyModel(spec, model2);
```

**Figura 5.22:** Criação dos modelos das ontologias, utilizando a API Jena

No nosso cenário, identificamos como termos chave as palavras “planta” e “pedra”, e o relacionamento “perto de”.

No nosso cenário iremos percorrer as ontologias  $O_{geo}$  e  $O_{tur}$ , vistas na Figura 5.23. Com os dois grafos (modelos) das ontologias disponíveis, a procura será por classes que possuam conceitos semelhantes ao termo utilizado pelo usuário.



**Figura 5.23:** Ontologias da comunidade de geólogos e turistas, na procura por termos semelhantes

Primeiramente, será pesquisado na ontologia do usuário ( $O_{tur}$ ) o termo que ele pesquisou (no nosso cenário, planta e pedra). Ao encontrar estes termos, a pesquisa agora será feita por termos semelhantes a este na ontologia do banco de dados ( $O_{geo}$ ).

Iniciamos a pesquisa na  $O_{tur}$  e detectamos as classes “Planta” e “Pedra” (sinônimo de região de Corais). Nestas classes, a procura agora é por *tags* que definem alguma semelhança com classes de outras ontologias.

Podemos acompanhar parte do código OWL da ontologia  $O_{tur}$  na Figura 5.24. Na linha 3, vemos a utilização da *tag sameAs*, identificando que a classe *Planta* (definida na linha 1) possui semelhança conceitual (linha 3) à classe *vegetação\_submersa*, da  $O_{geo}$  (linha 4). Um código semelhante a este define que a classe *Pedra*, contida na  $O_{tur}$ , é semelhante à classe *corpo\_coralíneo*, contida na  $O_{geo}$ .

```
1. <owl:Class rdf:ID="Planta">
2. <rdfs:subClassOf rdf:resource="#Organismos_Marinhos"/>
3. <owl:sameAs
4. rdf:resource="http://localhost:8080/Onto_Geologo.owl#Vegetacao_Submersa"/>
5. </owl:Class>
```

**Figura 5.24:** Código OWL da Otur

Para a ligação de classes semanticamente sinônimas, usamos a *tag owl:sameAs*, pois iremos definir semelhança de conceitos entre dois indivíduos de ontologias diferentes, que possuem nomes diferentes.

Assim, detectamos que as classes "Planta" e "Pedra", contidas na *O<sub>tur</sub>*, têm o mesmo significado das classes "vegetação\_submersa" e "corpo\_coralíneo", respectivamente, contidas na *O<sub>geo</sub>*. São, assim, classes sinônimas, que possuem a mesma semântica, mas nomenclaturas diferentes.

Com este método, o *módulo de comparação* consegue a informação que a classe "planta" tem semelhança conceitual com a classe "vegetação\_submersa", bem como a classe "pedra" com a classe "corpo\_coralíneo".

No caso do nosso cenário, a semelhança foi encontrada no primeiro nível do grafo. Ou seja, ao encontrar a classe "Planta", foi logo encontrada a semelhança com a classe "Vegetação\_Submersa". Caso esta semelhança conceitual não fosse encontrada neste primeiro nível, o *módulo de comparação* desceria um nível no grafo, procurando por relacionamentos entre classes. Iremos mostrar um exemplo deste caso no próximo exemplo.

A ontologia do geólogo é composta por termos que estão presentes no BDG, ou seja, todas as tabelas contidas no BDG possuem uma classe com o mesmo nome, na ontologia do banco de dados (*O<sub>geo</sub>*).

O *módulo geração de consulta* trabalha com esta informação, bem como com a informação de todos os nomes das colunas de geometria das tabelas do BDG. Para isto, foi criado um arquivo XML, que armazena o nome das tabelas e os nomes de suas respectivas colunas.

Além das ontologias e do arquivo XML com nomes das colunas do BDG, nós mantemos uma lista de semelhanças dos relacionamentos. Ou seja, para cada relacionamento utilizado na consulta, temos a sua

semelhança com algum operador espacial definido no modelo 9-*Intersection*. Para nossa aplicação, o termo "perto de" significa dizer que está a uma distância menor ou igual a 20 metros, por exemplo. A Tabela 5.1 mostra o dicionário de relacionamentos espaciais, que relaciona os termos presentes nas consultas, bem como os termos espaciais semelhantes.

<b>Relacionamento</b>	<b>Operador</b>
Ao redor de	<i>Buffer</i>
Cruzam	<i>Crosses</i>
Comprimento	<i>Lenght</i>
Dentro de	<i>Within</i>
Fora de	<i>Not within ou Disjoint</i>
Intersecta	<i>Intersection</i>
Perto de	<i>Distance &lt;= 20</i>
Sobrepõe	<i>Overlaps</i>
Toca	<i>Touches</i>

**Tabela 5.1:** Relacionamentos espaciais

Podemos acompanhar parte do *módulo geração de consulta* na Figura 5.25. Para montar a consulta a ser submetida ao BDG, este módulo recolhe o nome da tabela e o nome da coluna, substituindo seus valores na montagem da consulta.

```
String coluna = obterNomeColunaGeom(classe);
String tabela = obterNomeTabela(classe);
newLayer.setData(coluna + " FROM ("
    + " SELECT vs.oid AS oid, vs." + coluna + " AS "
    + coluna
    + " FROM " + tabela + " AS vs "
    + " WHERE" + funcaoRelacionamentos()
    + " AS foo USING UNIQUE oid USING SRID = -1");
```

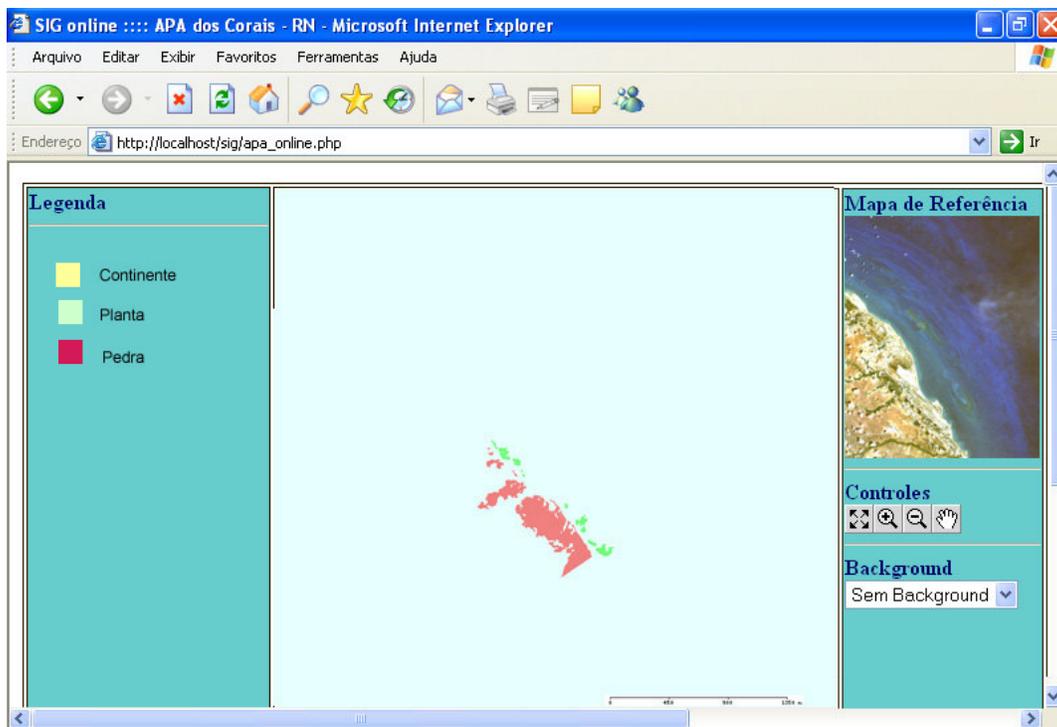
**Figura 5.25:** Parte do módulo geração de consulta.

Desta forma, o *módulo geração de consultas* vai usar a geometria selecionada, pesquisar a lista de relacionamentos e operadores, e vai submeter a seguinte consulta ao BDG:

```
SELECT vs.oid AS oid, vs.geom_vegetacaosubmersa AS  
      geom_vegetacaosubmersa, cc.geom_cabeco AS geom_cabeco  
FROM vegetacao_submersa AS vs, corpo_coralineo AS cc  
WHERE distance (vs.geom_vegetacaosubmersa,cc.geom_cabeco) <=  
      20)  
AS foo USING UNIQUE oid USING SRID = -1
```

O retorno desta consulta desenhará no mapa todas as vegetações que estão a uma distância menor ou igual a 20 metros dos corpos coralíneos.

Podemos ver o resultado desta consulta na Figura 5.26, mostrando o mapa gerado para a consulta. A Figura 5.27 mostra o texto explicativo, gerado pelo *módulo geração do dicionário*.



**Figura 5.26:** Resultado da Consulta: "plantas perto de pedras"

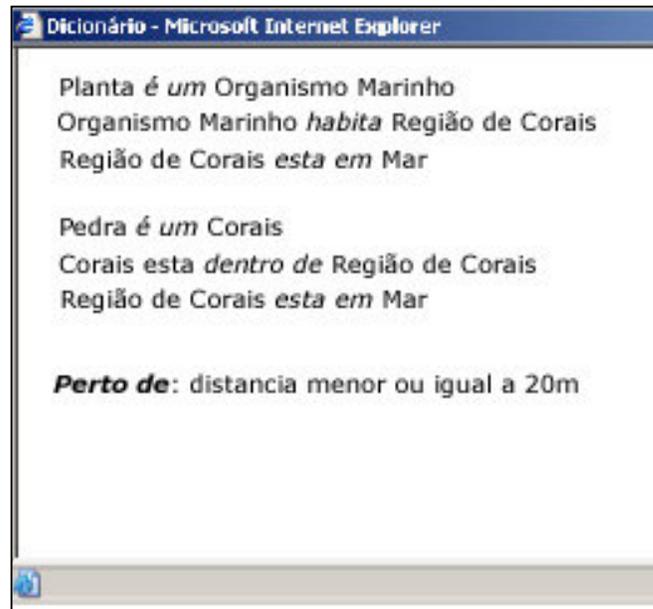


Figura 5.27: Texto explicativo, gerado pelo *módulo geração do dicionário*

#### 5.4.2 – Exemplos de Inferências

No exemplo anterior mostramos a inferência de semelhança conceitual entre as classes “planta” e “pedra”, da ontologia do turista, com as classes “vegetação\_submersa” e “corpo\_coralíneo”, da ontologia do geólogo, respectivamente.

Salientamos que várias outras inferências podem ser realizadas, utilizando as ontologias criadas, ligando as classes que possuem semelhança conceitual.

Por exemplo, se o turista desejar “localizar a região de corais onde as pedras ficam mais visíveis”. De novo esta informação não está armazenada no banco de dados. Mas, como foi dito na ontologia do turista, a idéia de pedras mais visíveis para o turista corresponde exatamente a região de Crista da ontologia do geólogo. Desta forma, após todo o processamento da camada semântica, a seguinte consulta seria levada ao BDG:

```
SELECT vs.oid AS oid, zr.geom_zonarecifal AS geom_zonarecifal  
FROM zona_recifal zr, tipo_zona_recifal tr  
WHERE zr.id_tipo_zona = tr.id_tipo_zona and tr.nome = 'Crista'
```

Ainda com o usuário turista, podemos realizar outras inferências, como a opção: “mergulho”. Se esta opção for selecionada pelo usuário, o

---

sistema irá indicar regiões que sejam mais seguras para este fim, ou seja, que tenham uma maior profundidade. A classe “Mergulhar” possui uma propriedade que se relaciona com a Corpo Coralíneo, da ontologia do geólogo, que apresentem uma maior profundidade, desta forma, a consulta levada ao BDG seria :

```
SELECT vs.oid AS oid, geom_cabeco AS geom_cabeco
FROM corpo_coralineo
WHERE profundidade > 10 (o campo profundidade é medido em metros).
```

Podemos também realizar inferências com o usuário do tipo biólogo. Observemos a consulta: “local onde Fitoplânctons podem ser encontrados”; Esta classe possui semelhança com a classe macro alga (vegetação submersa), do geólogo. Pela hierarquia de relacionamentos, vemos que Fitoplâncton é um Organismo Pelágico que vive no Recife; e desta forma, a consulta levada ao BDG seria :

```
SELECT vs.oid AS oid, geom_vegetacaosubmersa AS
geom_vegetacaosubmersa
FROM vegetacao_submersa vs, tipo_vegetacao_submersa tv
WHERE vs.id_tipo_vegetacao_submersa =
tv.id_tipo_vegetacao_submersa and tv.nome = 'MACROALGA'
```

Outras inferências podem ser realizadas, mas, para que o texto não fique repetitivo, não iremos demonstrar todos os exemplos possíveis nesta dissertação. Devido a limitação de tempo, a interface de consulta não disponibiliza ainda todas as opções possíveis de consulta, mas, o sistema já está realizando todas estas inferências citadas.

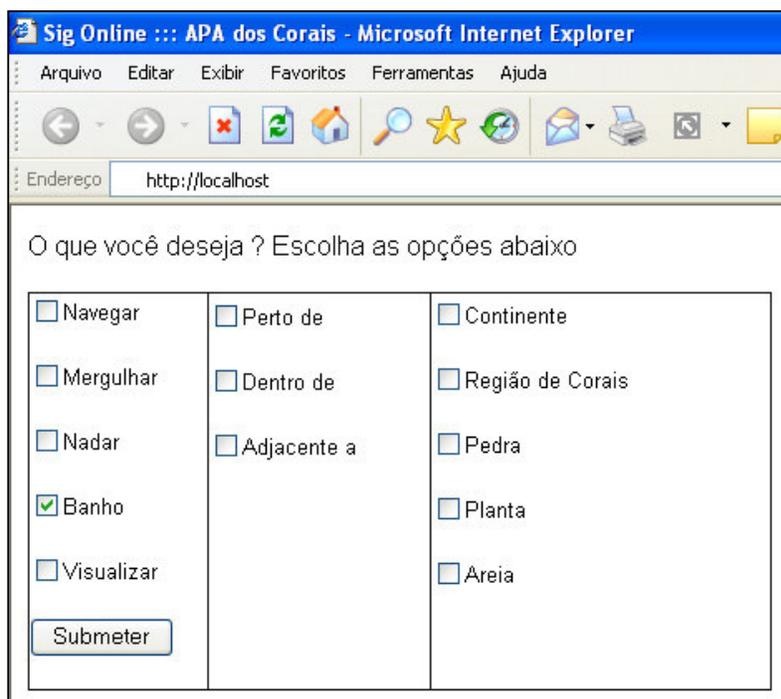
## 5.5 – Outros Cenários de Consulta

Nos próximos tópicos, descreveremos mais dois cenários de consultas, para a validação da camada semântica proposta no capítulo anterior.

---

### 5.5.1 – Primeiro Cenário

Neste cenário, veremos o exemplo de um turista que deseja “localizar uma área para banho nos recifes de corais”; essa informação não está contida no banco de dados. Vemos na Figura 5.28 a interface pré-definida de consultas.



The screenshot shows a Microsoft Internet Explorer window titled "Sig Online :: APA dos Corais". The address bar contains "http://localhost". The main content area displays the text "O que você deseja? Escolha as opções abaixo" followed by a form with three columns of checkboxes. The first column contains "Navegar", "Mergulhar", "Nadar", "Banho" (checked), and "Visualizar". The second column contains "Perto de", "Dentro de", and "Adjacente a". The third column contains "Continente", "Região de Corais", "Pedra", "Planta", and "Areia". A "Submeter" button is located at the bottom left of the form.

**Figura 5.28:** Interface pré-definida de consulta

As ontologias ativadas para esta consulta serão a do geólogo ( $O_{geo}$ ) e a do turista ( $O_{turAtracao}$  e  $O_{tur}$ ).

A palavra chave “Banho” é coletada da consulta do usuário turista.

Inicialmente o sistema procura na ontologia do turista por alguma classe relativa ao termo “banho”. A classe “Banho” é encontrada na  $O_{tur}$ , na ontologia que define as atividades turísticas.

Com a classe “Banho” encontrada, a busca agora será por classes semelhantes, na  $O_{geo}$ . Nenhuma classe semelhante a “Banho” é encontrada na  $O_{geo}$ , neste ponto.

Como a semelhança conceitual não foi encontrada no primeiro nível do grafo, o *módulo de comparação* vai descer um nível no grafo, procurando por relacionamentos entre classes. Desta maneira, a busca

---

continuará, na  $O_{tur}$ , pesquisando se a classe “Banho” possui sub-classes ou relacionamentos com outras classes.

Para realizar esta procura um nível mais “abaixo” do grafo, acompanharemos parte do código na Figura 5.29. Se não for encontrada nenhuma semelhança entre a palavra pesquisada e algum termo na ontologia do banco de dados (linha 1), então a busca será feita por alguma classe que tenha ligação, através de propriedades, com a classe pesquisada (linha 4). O método *obterTermoGeologo* (linha 6) é utilizado para obter a semelhança entre a classe encontrada e alguma classe na ontologia do banco de dados.

```
1. if (palavraRetorno == null) {
2.     ExtendedIterator iter = OMTuristaAt.listObjectProperties();
3.     while (iter.hasNext()) {
4.         ObjectProperty element = (ObjectProperty) iter.next();
5.         if (element.getDomain().getLocalName().equals(palavraChave)) {
6.             obterTermoGeologo(consultaRetorno,
7.                 element.getRange().getLocalName(), tipoOntologia);
8.         }
9.     }
```

**Figura 5.29:** Parte do código Java, utilizando a API Jena na procura de classes semelhantes.

Percorrendo a ontologia  $O_{tur}$ , temos a relação Banho *em* Piscinas Naturais e Banho *ao redor* dos Flutuantes. Ou seja, a classe “Banho” não possui sub-classes (relacionamento ISA), no entanto possui propriedades que ligam esta classe à outras classes na ontologia. A propriedade *em* liga a classe “Banho” (domínio) à classe “Piscina Natural”. Ainda temos a propriedade *ao redor*, que liga a classe “Banho” (domínio) à classe “Flutuante”.

Então, a busca agora será por classes semelhantes a “Piscina Natural” e “Flutuante”.

A classe “Flutuante”, na  $O_{tur}$ , possui semelhança conceitual com a classe “Flutuante”, na  $O_{geo}$ . A classe denominada “Flutuante”, presente na  $O_{geo}$ , refere-se a embarcações de posição fixa, pertencentes a empresas ou instituições, e servem de base de apoio para turistas e pesquisadores.

Temos, neste momento duas informações a respeito da classe Flutuante. A primeira foi retirada da  $O_{tur}$ , informando que Flutuante é

---

*circundado por* Banho; a segunda, retirada da *O<sub>geo</sub>*, é a semelhança conceitual com a classe Flutuante, presente nesta ontologia.

Temos ainda a procura por semelhança conceitual da classe "Piscina Natural", da *O<sub>tur</sub>*. A semelhança é encontrada, com a classe "Piscina Natural", na *O<sub>tur</sub>*. A procura agora é pela semelhança desta classe com alguma classe da ontologia do geólogo; Não é encontrada nenhuma semelhança conceitual, e assim o *módulo de comparação* vai descer mais um nível no grafo, procurando por relacionamentos entre classes. É encontrado o relacionamento, na *O<sub>tur</sub>*, "Piscina Natural" *ao redor de* "Corais"; "Corais" está *dentro de* "Região de Corais". Com isto, temos que cada corpo coralíneo, presente na região de corais, possuem piscinas naturais ao seu redor. "Corpo Coralíneo" está modelado no BDG.

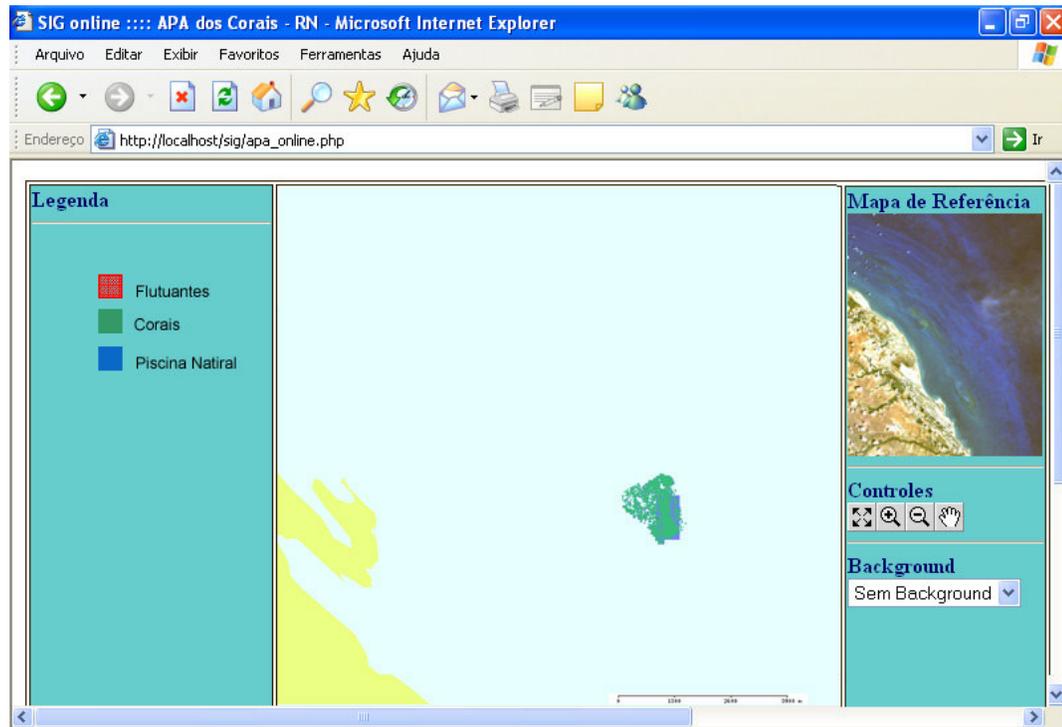
Assim, na informação retirada das nossas ontologias, a área para banho nos recifes de corais são aquelas que estão ao redor dos flutuantes e também ao redor dos corpos coralíneos. "Flutuante" e "Corpo Coralíneo" estão modelados no banco de dados, e através das propriedades, e desta forma, é inferido que a área de banho do turista é uma área ao redor dos flutuantes e ao redor dos corpos coralíneos.

A consulta levada ao banco de dados, após a inferência entre as duas ontologias será: "mostre as áreas ao redor dos flutuantes e ao redor dos corpos coralíneos". A consulta montada é a seguinte:

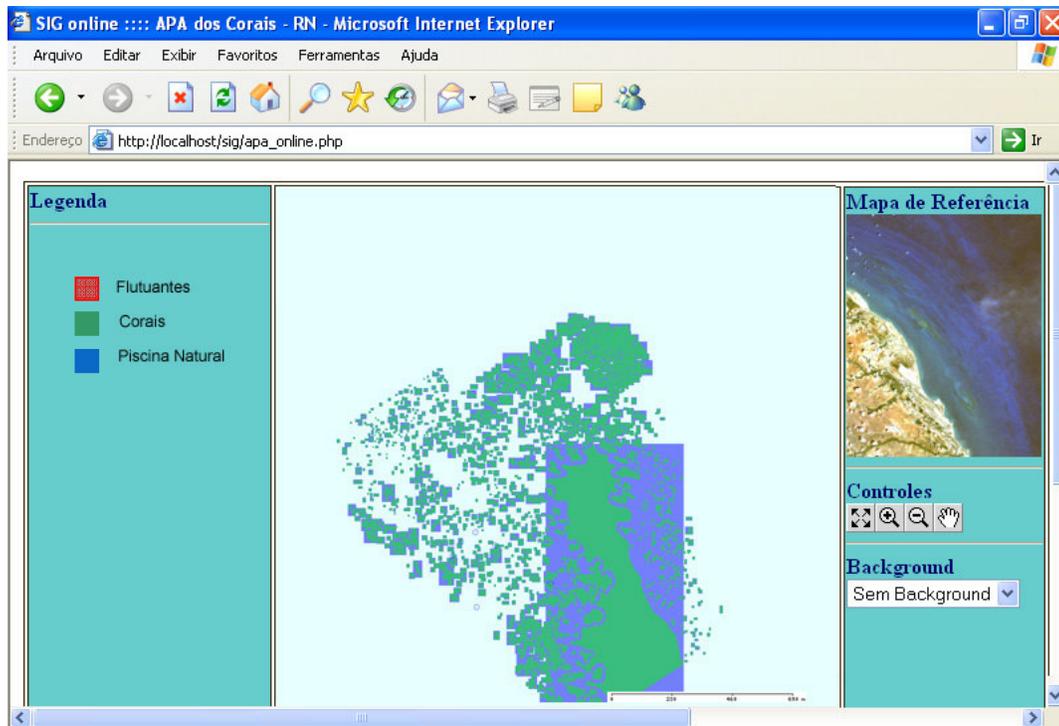
```
SELECT buffer(flutuante.flutuante_geom, 10) AS  
        flutuante_geom, buffer(corpo_coralineo.geom_cabeco,  
        10) AS geom_cabeco  
FROM flutuante, corpo_coralineo  
AS foo USING UNIQUE oid USING SRID = -1
```

Para esta consulta, utilizamos o operador 'buffer'. De acordo com a especificação do manual do PostGIS [PostGis 2006], este operador retorna as geometrias cuja distância à geometria desejada (geom\_flutuante e geom\_cabeco) é menor ou igual a distância passada como parâmetro. Para a nossa aplicação em questão, definimos que a distância ao redor dos flutuantes para uma boa área de banho seria a distância ao objeto geométrico em questão de até 10 metros.

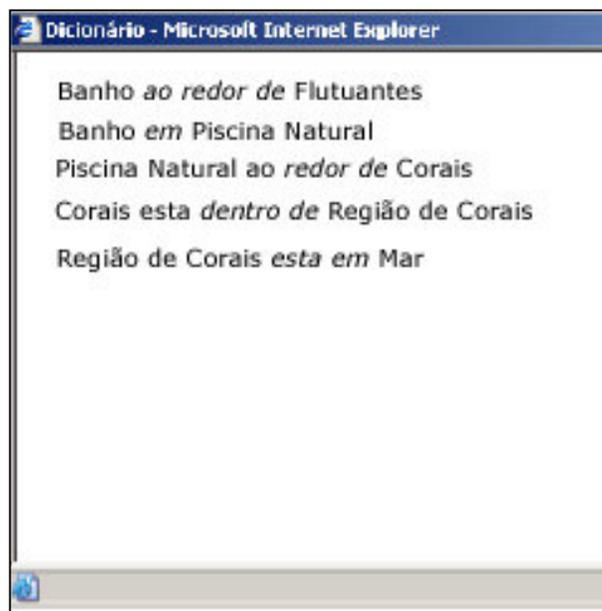
Podemos ver o resultado desta consulta na Figura 5.30. Com as ferramentas de *zoom in* e *zoom out*, presentes na aplicação, pode-se ver mais detalhes da figura em questão, como mostra a Figura 5.31. Na Figura 5.32, podemos ver as informações referentes à consulta do usuário, mostradas através de uma janela *pop-up*.



**Figura 5.30:** *screenshot* do resultado da consulta “selecione as áreas ao redor dos flutuantes e corpos coralíneos”



**Figura 5.31:** *Zoom in* na figura da consulta "selecione as áreas ao redor dos flutuantes e corpos coralíneos"

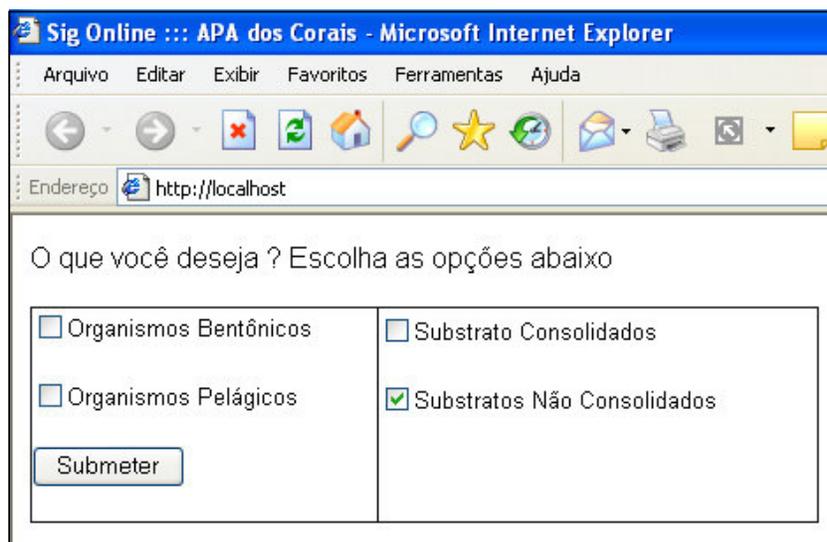


**Figura 5.32:** Janela *pop-up* do dicionário gerado para a consulta em questão.

---

## 5.5.2 – Segundo Cenário de Consulta

Neste cenário, veremos o exemplo de um biólogo que deseja localizar os substratos não consolidados da região dos recifes de corais; essa informação não está contida no banco de dados. Vemos na Figura 5.33 a interface pré-definida de consultas.



**Figura 5.33:** Interface pré-definida da comunidade de biólogos.

As ontologias ativadas para esta consulta serão as *O<sub>geo</sub>* e *O<sub>bio</sub>*. Começa a procura na *O<sub>bio</sub>* pela classe "substrato inconsolidado"; esta classe é encontrada na ontologia do biólogo, e a busca agora é pela semelhança entre classes. A classe "substrato inconsolidado" não possui nenhuma *tag sameAs*, que ligue esta classe a outra classe pela semelhança conceitual.

Como a semelhança conceitual não foi encontrada no primeiro nível do grafo, o *módulo de comparação* vai descer um nível no grafo, procurando por relacionamentos entre classes. Desta maneira, a busca continuará, na *O<sub>bio</sub>*, pesquisando se a classe "substrato inconsolidado" possui sub-classes ou relacionamentos com outras classes.

O *módulo de comparação*, percorrendo o grafo da *O<sub>bio</sub>*, encontra que a classe "substrato inconsolidado" possui sub-classes (relacionamento ISA) com as classes "Areia", "Cascalho" e "Lama". Podemos dizer, com este relacionamento, que Areia é *um* substrato inconsolidado, por exemplo.

---

A busca agora é por alguma semelhança entre estas classes e as classes da ontologia do banco de dados (**O<sub>geo</sub>**). Todas estas classes ("Areia", "Cascalho" e "Lama"), presentes na **O<sub>bio</sub>**, possuem semelhança conceitual com as classes Areia, Cascalho e Lama, respectivamente, contidas na **O<sub>geo</sub>**.

Podemos ver parte do código OWL da **O<sub>bio</sub>** na Figura 5.34. Vemos na linha 1 a definição da classe Areia, que é sub-classe de substrato inconsolidado (linha 2) e possui semelhança conceitual com a classe Areia, presente na **O<sub>geo</sub>** (linhas 3 e 4).

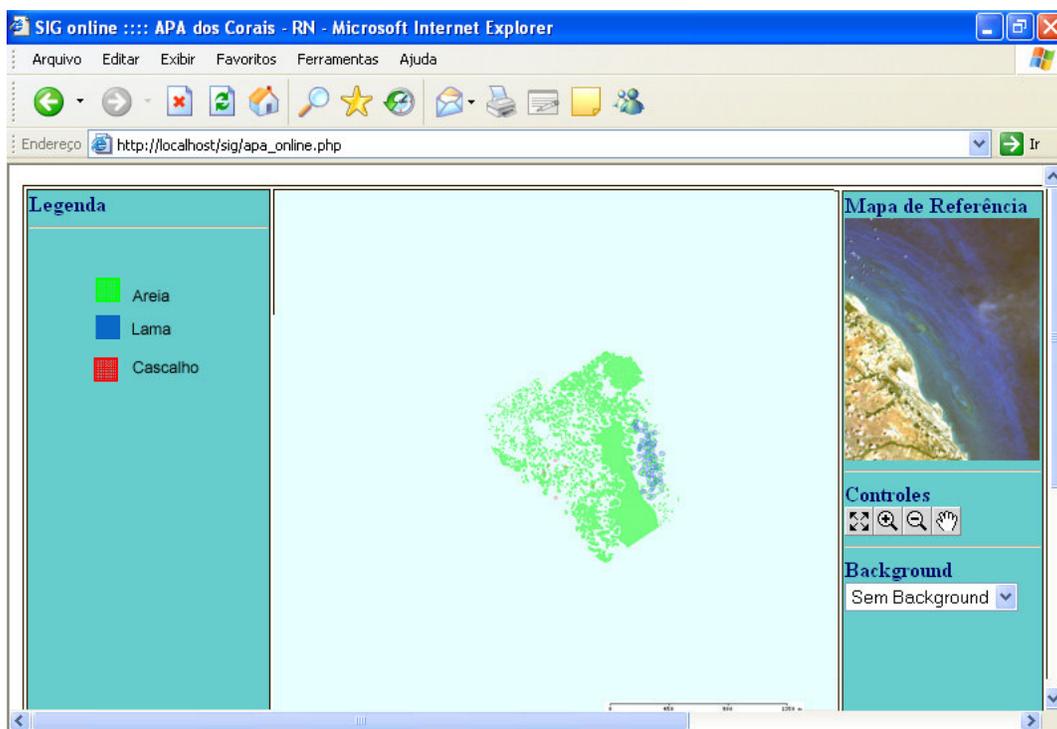
```
1. <owl:Class rdf:ID="Areia">
2.   <rdfs:subClassOf rdf:resource="#Substrato_Inconsolidado"/>
3.   <owl:sameAs
4.     rdf:resource="http://localhost:8080/Onto_Geologo.owl#Areia" >
5.   </owl:sameAs>
6. </owl:Class>
```

**Figura 5.34:** Parte do código OWL da ontologia do biólogo.

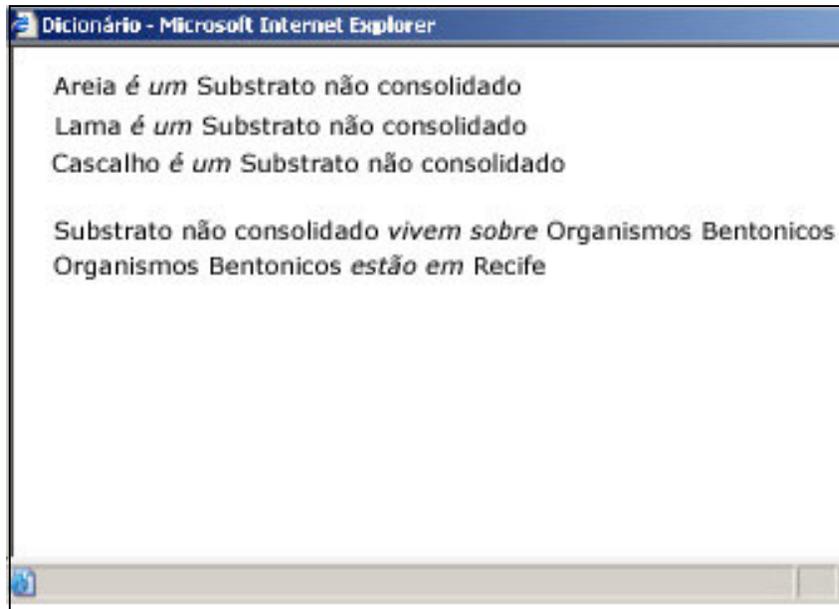
Com a semelhança conceitual encontrada, o próximo passo é montar a consulta a ser levada ao banco de dados. O *módulo geração de consultas* vai submeter ao banco de dados a seguinte consulta:

```
SELECT geom_areia AS geom_areia, geom_lama AS geom_lama,
       geom_cascalho AS geom_cascalho
FROM areia, cascalho, lama
AS foo USING UNIQUE oid USING SRID = -1
```

Esta consulta mostrará no mapa todos os locais que possuem lama, areia e cascalho, como vemos na Figura 5.35. Vemos na Figura 5.36 o dicionário explicativo gerado para esta consulta.



**Figura 5.35:** *screenshot* do resultado da consulta “selecione as áreas que possuem lama, areia e cascalho”.



**Figura 5.36:** Janela *pop-up* do dicionário gerado para a consulta do usuário biólogo

---

# **Capítulo 6**

## **Conclusão e Trabalhos Futuros**

### **6.1 – Introdução**

Este trabalho apresentou um mecanismo baseado em ontologias para permitir que diferentes consultas sejam submetidas a um mesmo banco de dados geográfico. Este mecanismo permite que diferentes comunidades de usuários, através de diferentes ontologias geográficas, acessem a mesma base de dados, sem que para isso seja necessário conhecer a estrutura desta base.

Para isto, nós realizamos uma interação semântica da informação geográfica, através do uso de ontologias e da similaridade semântica entre classes, mas que possuem diferentes nomenclaturas. A ligação entre as classes similares nas ontologias é feita usando propriedades OWL, que ligam estas classes sinônimas.

Foi proposta a construção de uma camada semântica, para a interação com o SIG e o banco de dados geográfico, através de ontologias. As consultas, antes de chegar ao banco de dados, são intermediadas pela camada semântica, que irá processar as classes similares e levá-las ao banco de dados.

Como estudo de caso, desenvolvemos ontologias para o domínio de recifes de corais de acordo com três visões de diferentes comunidades: a comunidade dos geólogos, dos biólogos e a dos turistas.

Cada classe de cada ontologia criada tem uma similaridade com a classe de outra ontologia, mas os termos usados para defini-las são diferentes. As ontologias propostas podem ser usadas como ferramenta de navegação e consulta para o usuário, fornecendo a informação semântica desejada. Além disso, as ontologias podem ser usadas para estabelecer relacionamentos entre diferentes visões do conceito, transportando o significado das idéias que são embutidas na representação de fenômenos geográficos à mente humana.

---

É importante salientar que, devido ao domínio escolhido ser um pouco restrito, não foi possível fazer grandes demonstrações de consultas diferentes. No entanto esta proposta pode ser adaptada a qualquer domínio espacial que seja de interesse multidisciplinar. E, neste caso, o retorno destas novas aplicações com o uso de ontologias, da forma como foi especificada nesta proposta, é muito satisfatório.

Uma limitação do nosso trabalho seria quanto à interface de consulta. Devido ao tempo curto, fizemos uma interface de consulta pré-definida, sem o uso de linguagem natural. Em nosso projeto, o usuário escolhe, dentre uma lista de palavras, o que ele deseja pesquisar. A interface é baseada nas ontologias dos usuários em questão.

## **6.2 – Contribuições**

1. Construção de todas as ontologias, referentes às comunidades em questão referenciadas neste trabalho: geólogo, biólogo e turistas.
2. Definição de relacionamentos espaciais para representar ontologias geográficas;
3. Implementação da camada semântica, com a definição de classes equivalentes entre as diversas ontologias, e a sua integração com o Banco de Dados Geográfico.
4. Criação de uma interface de consulta adaptável ao tipo do usuário, permitindo que o usuário interaja com o SIG utilizando termos específicos de sua área pré-definidos.
5. Tradução da Consulta feita em termos das ontologias para a linguagem SQL;
6. Submissão da Consulta ao BDG e retorno do resultado aos usuários;
7. Disposição de informações descritivas no resultado das consultas de modo a esclarecer o conteúdo do mesmo, como também a respeito de termos técnicos.

---

### **6.3 – Trabalhos Futuros**

Como trabalhos futuros, sugere-se:

1. Elaboração de outros tipos de ontologias, referentes às diferentes comunidades que ainda irão interagir com a aplicação.
2. Aplicação desta camada semântica a outro domínio espacial, para obter uma maior diversidade tanto de consultas, como de ontologias.
3. Melhoria da Interface de Consulta adicionando novas funcionalidades à mesma, de modo que o usuário possa definir cores de saída, e sobrepor resultados. Além disso, permitir uma melhor qualidade no resultado das mesmas.
4. Uso de Linguagem Natural na Construção de Consultas de modo a atender a uma maior gama de consultas.
5. Construção de métodos e aplicação de novas tecnologias para inferência automática dos conceitos nas ontologias.

---

## **Glossário**

- **Parrachos de Maracajaú:** é um ecossistema marinho localizado sobre uma região de plataforma continental que pode ser definida como: Solo marinho raso e aproximadamente horizontal que vai da linha da costa até o topo da quebra de plataforma

- **Organismos Bentônicos:** São organismos que vivem junto ao substrato. Podem viver em fundo de areia, fundos rochosos com presença de algas, fundos de recifes de corais.

- **Organismos Pelágicos:** Organismos que vivem em meio líquido, ou seja, na coluna de água.

- **Substratos Consolidados:** Refere-se ao tipo de fundo; Tudo que é firme, duro. Ex: rochas, recifes.

- **Substratos Não Consolidados:** tudo que é mole, móvel. Ex. areia, lama

- **Invertebrados:** são animais que não apresentam vértebras, como os insetos, polvos, crustáceos, corais, esponjas, estrelas-do-mar, etc.

- **Plâncton:** organismos microscópicos (plantas ou animais – podem ser ovos, larvas, minúsculos crustáceos, que vivem flutuando ao sabor das correntes. Se forem microalgas, será o *fitoplâncton*, se animais – *zooplâncton*).

- **Larvas:** Larvas são estágios juvenis dos animais – larvas de peixes, de ouriços-do-mar; com o passar do tempo elas se modificam e se tornam iguais aos adultos.



---

```

<rdfs:subClassOf>
  <owl:Restriction>
    <owl:someValuesFrom>
      <owl:Class rdf:ID="Crista"/>
    </owl:someValuesFrom>
    <owl:onProperty>
      <owl:ObjectProperty rdf:about="#possui"/>
    </owl:onProperty>
  </owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf>
  <owl:Class rdf:ID="Plataforma"/>
</rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Pesquisa">
  <rdfs:subClassOf>
    <owl:Class rdf:ID="Flutuante"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:about="#Retrorecife">
  <owl:disjointWith>
    <owl:Class rdf:about="#Recife_Frontal"/>
  </owl:disjointWith>
  <owl:disjointWith>
    <owl:Class rdf:about="#Crista"/>
  </owl:disjointWith>
  <rdfs:subClassOf rdf:resource="#Componentes_Recifais"/>
  <rdfs:comment
rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >Região do recife voltada para a costa que normalmente sofre
menos com o impacto das ondas devido à proteção exercida pela
crista e o recife frontal. </rdfs:comment>
</owl:Class>
<owl:Class rdf:ID="MicroOrganismos">
  <rdfs:subClassOf>
    <owl:Class rdf:ID="Especies"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Barco"/>
<owl:Class rdf:ID="Cascalho">
  <rdfs:subClassOf>
    <owl:Class rdf:ID="Sedimentos"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Moreia">
  <rdfs:subClassOf rdf:resource="#Especies"/>
</owl:Class>
<owl:Class rdf:ID="Peixes">
  <rdfs:subClassOf rdf:resource="#Especies"/>
</owl:Class>
<owl:Class rdf:about="#Plataforma">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#Margem_Continental"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Corpo_Coralineo_Emerso">
  <rdfs:subClassOf>
    <owl:Class rdf:ID="Corpo_Coralineo"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:about="#Margem_Continental">

```

---

```

    <rdfs:subClassOf>
      <owl:Class rdf:ID="Marinho"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="Exploracao_Turistica">
    <rdfs:subClassOf>
      <owl:Class rdf:about="#Flutuante"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:about="#Crista">
    <owl:disjointWith>
      <owl:Class rdf:about="#Recife_Frontal"/>
    </owl:disjointWith>
    <rdfs:comment
rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
      >Região, entre o retrorecife e o recife frontal, onde o corpo
coralíneo fica parcialmente emerso nos perfis de baixa
mar.</rdfs:comment>
    <owl:disjointWith rdf:resource="#Retrorecife"/>
    <rdfs:subClassOf rdf:resource="#Componentes_Recifais"/>
  </owl:Class>
  <owl:Class rdf:ID="Algas">
    <rdfs:subClassOf rdf:resource="#Especies"/>
  </owl:Class>
  <owl:Class rdf:ID="Areia">
    <rdfs:subClassOf>
      <owl:Class rdf:about="#Sedimentos"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:about="#Flutuante">
    <rdfs:comment
rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
      >Embarcações de posição fixa, pertencentes a empresas ou
instituições, e servem de base de apoio para turistas e
pesquisadores</rdfs:comment>
  </owl:Class>
  <owl:Class rdf:ID="Lama">
    <rdfs:subClassOf>
      <owl:Class rdf:about="#Sedimentos"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="Corais">
    <rdfs:subClassOf rdf:resource="#Especies"/>
  </owl:Class>
  <owl:Class rdf:about="#Marinho">
    <rdfs:subClassOf rdf:resource="#Ecosistema"/>
  </owl:Class>
  <owl:Class rdf:about="#Corpo_Coralíneo">
    <rdfs:subClassOf rdf:resource="#Região_Bentica"/>
  </owl:Class>
  <owl:Class rdf:ID="Capim_do_mar">
    <rdfs:subClassOf rdf:resource="#Vegetação_Submersa"/>
  </owl:Class>
  <owl:Class rdf:about="#Sedimentos">
    <rdfs:subClassOf rdf:resource="#Região_Bentica"/>
  </owl:Class>
  <owl:Class rdf:ID="Crustáceos">
    <rdfs:subClassOf rdf:resource="#Especies"/>
  </owl:Class>
  <owl:Class rdf:ID="Área_Banho_Turista"/>
  <owl:Class rdf:ID="Planície_Abissal">

```

---

```

    <rdfs:subClassOf rdf:resource="#Marinho"/>
  </owl:Class>
  <owl:Class rdf:ID="Macro_Alga">
    <rdfs:subClassOf rdf:resource="#Vegetacao_Submersa"/>
  </owl:Class>
  <owl:Class rdf:about="#Recife_Frontal">
    <rdfs:comment
rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >Localiza-se na porÃ§Ã£o exterior do recife, ou seja, no lado
oposto Ã costa, absorvendo grande parte da energia das ondas.
</rdfs:comment>
    <owl:disjointWith rdf:resource="#Crista"/>
    <rdfs:subClassOf rdf:resource="#Componentes_Recifais"/>
    <owl:disjointWith rdf:resource="#Retrorecife"/>
  </owl:Class>
  <owl:Class rdf:ID="Elevacao">
    <rdfs:subClassOf rdf:resource="#Margem_Continental"/>
  </owl:Class>
  <owl:Class rdf:ID="Continental">
    <rdfs:subClassOf rdf:resource="#Ecossistema"/>
  </owl:Class>
  <owl:Class rdf:ID="Corpo_Coralineo_Submerso">
    <rdfs:subClassOf rdf:resource="#Corpo_Coralineo"/>
  </owl:Class>
  <owl:Class rdf:ID="Rota_de_Barco"/>
  <owl:ObjectProperty rdf:ID="habitam">
    <owl:inverseOf>
      <owl:ObjectProperty rdf:ID="habitado_por"/>
    </owl:inverseOf>
    <rdfs:domain rdf:resource="#Especies"/>
    <rdfs:range rdf:resource="#Regiao_Bentica"/>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:ID="adjacente_por">
    <owl:equivalentProperty>
      <owl:ObjectProperty rdf:ID="tocado_por"/>
    </owl:equivalentProperty>
    <rdfs:domain rdf:resource="#Recife_Frontal"/>
    <rdfs:range rdf:resource="#Crista"/>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:ID="circunda">
    <rdfs:range rdf:resource="#Flutuante"/>
    <rdfs:domain rdf:resource="#Area_Banho_Turista"/>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:ID="atendido_por">
    <rdfs:range rdf:resource="#Rota_de_Barco"/>
    <rdfs:domain rdf:resource="#Flutuante"/>
    <owl:inverseOf>
      <owl:ObjectProperty rdf:ID="atende"/>
    </owl:inverseOf>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:ID="faz_parte_de">
    <rdfs:domain rdf:resource="#Componentes_Recifais"/>
    <owl:inverseOf>
      <owl:ObjectProperty rdf:about="#possui"/>
    </owl:inverseOf>
    <rdfs:range rdf:resource="#Recife"/>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:ID="possui_barco">
    <rdfs:range rdf:resource="#Barco"/>
    <rdfs:domain rdf:resource="#Rota_de_Barco"/>
  </owl:ObjectProperty>

```

---

```

<owl:ObjectProperty rdf:ID="tem">
  <owl:inverseOf>
    <owl:ObjectProperty rdf:ID="esta_em"/>
  </owl:inverseOf>
  <rdfs:range rdf:resource="#Regiao_Bentica"/>
  <rdfs:domain>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Crista"/>
        <owl:Class rdf:about="#Retrorecife"/>
        <owl:Class rdf:about="#Recife_Frontal"/>
      </owl:unionOf>
    </owl:Class>
  </rdfs:domain>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#esta_em">
  <rdfs:domain rdf:resource="#Regiao_Bentica"/>
  <owl:inverseOf rdf:resource="#tem"/>
  <rdfs:range>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Retrorecife"/>
        <owl:Class rdf:about="#Crista"/>
        <owl:Class rdf:about="#Recife_Frontal"/>
      </owl:unionOf>
    </owl:Class>
  </rdfs:range>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#atende">
  <owl:inverseOf rdf:resource="#atendido_por"/>
  <rdfs:domain rdf:resource="#Rota_de_Barco"/>
  <rdfs:range rdf:resource="#Flutuante"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#possui">
  <rdfs:range rdf:resource="#Componentes_Recifais"/>
  <owl:inverseOf rdf:resource="#faz_parte_de"/>
  <rdfs:domain rdf:resource="#Recife"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="adjacente">
  <rdfs:range rdf:resource="#Recife_Frontal"/>
  <rdfs:domain rdf:resource="#Crista"/>
  <owl:equivalentProperty>
    <owl:ObjectProperty rdf:ID="toca"/>
  </owl:equivalentProperty>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#tocado_por">
  <owl:equivalentProperty rdf:resource="#adjacente_por"/>
  <rdfs:domain rdf:resource="#Crista"/>
  <rdfs:range rdf:resource="#Retrorecife"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="Onto_Geologo_ObjectProperty_13">
  <rdfs:domain rdf:resource="#Crustaceos"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="ancorado">
  <rdfs:domain rdf:resource="#Flutuante"/>
  <rdfs:range rdf:resource="#Recife"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#toca">
  <rdfs:domain rdf:resource="#Retrorecife"/>
  <owl:equivalentProperty rdf:resource="#adjacente"/>
  <rdfs:range rdf:resource="#Crista"/>

```

---

```

</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="navega_em">
  <rdfs:domain rdf:resource="#Barco"/>
  <rdfs:range rdf:resource="#Rota_de_Barco"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#habitado_por">
  <owl:inverseOf rdf:resource="#habitam"/>
  <rdfs:domain rdf:resource="#Regiao_Bentica"/>
  <rdfs:range rdf:resource="#Especies"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="circundado_por">
  <rdfs:range rdf:resource="#Area_Banho_Turista"/>
  <rdfs:domain rdf:resource="#Flutuante"/>
</owl:ObjectProperty>
<owl:DatatypeProperty rdf:ID="area_turistica">
  <rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/>
  <rdfs:domain rdf:resource="#Recife"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="area">
  <rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#float"/>
  <rdfs:domain rdf:resource="#Recife"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="area_de_protecao">
  <rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/>
  <rdfs:domain rdf:resource="#Recife"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="longitude">
  <rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#float"/>
  <rdfs:domain rdf:resource="#Recife"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="latitude">
  <rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#float"/>
  <rdfs:domain rdf:resource="#Recife"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="profundidade">
  <rdfs:domain rdf:resource="#Recife"/>
  <rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#float"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="distancia_da_costa">
  <rdfs:domain rdf:resource="#Recife"/>
  <rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#float"/>
</owl:DatatypeProperty>
</rdf:RDF>

<!-- Created with Protege (with OWL Plugin 1.3, Build 225.4)
http://protege.stanford.edu -->

```

---

- **Ontologia do Turista**

```
<?xml version="1.0"?>
<rdf:RDF
  xmlns="http://localhost:8080/Onto_Turista.owl#"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xml:base="http://localhost:8080/Onto_Turista.owl">
  <owl:Ontology rdf:about=""/>
  <owl:Class rdf:ID="Peixe">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="Animais_Marinhos"/>
    </rdfs:subClassOf>
    <owl:sameAs
rdf:resource="http://localhost:8080/Onto_Geologo.owl#Peixes"/>
  </owl:Class>
  <owl:Class rdf:ID="Mar">
    <owl:sameAs
rdf:resource="http://localhost:8080/Onto_Geologo.owl#Marinho"/>
  <rdfs:subClassOf>
    <owl:Class rdf:ID="Ecosistema"/>
  </rdfs:subClassOf>
</owl:Class>
  <owl:Class rdf:ID="Cobra">
    <owl:sameAs
rdf:resource="http://localhost:8080/Onto_Geologo.owl#Moreia"/>
  <rdfs:subClassOf>
    <owl:Class rdf:about="#Animais_Marinhos"/>
  </rdfs:subClassOf>
</owl:Class>
  <owl:Class rdf:ID="Componente_Regiao_Corais"/>
  <owl:Class rdf:ID="Areia">
    <owl:sameAs
rdf:resource="http://localhost:8080/Onto_Geologo.owl#Areia"/>
  <rdfs:subClassOf>
    <owl:Class rdf:ID="Organismos_Marinhos"/>
  </rdfs:subClassOf>
</owl:Class>
  <owl:Class rdf:ID="Crustaceos">
    <owl:sameAs
rdf:resource="http://localhost:8080/Onto_Geologo.owl#Crustaceos"/>
  <rdfs:subClassOf>
    <owl:Class rdf:about="#Animais_Marinhos"/>
  </rdfs:subClassOf>
</owl:Class>
  <owl:Class rdf:ID="Continental">
    <owl:sameAs
rdf:resource="http://localhost:8080/Onto_Geologo.owl#Continental"/>
  <rdfs:subClassOf rdf:resource="#Ecosistema"/>
</owl:Class>
  <owl:Class rdf:ID="Planta">
    <rdfs:subClassOf rdf:resource="#Organismos_Marinhos"/>
    <owl:sameAs
rdf:resource="http://localhost:8080/Onto_Geologo.owl#Vegetacao_Submersa"/>
  </owl:Class>
  <owl:Class rdf:ID="Pedra">
    <owl:sameAs
```

---

```

    <owl:Class rdf:ID="Corais"/>
  </owl:sameAs>
</owl:Class>
<owl:Class rdf:ID="Pedras_Mais_Fundas">
  <owl:sameAs
rdf:resource="http://localhost:8080/Onto_Geologo.owl#Recife_Frontal"/>
  <rdfs:subClassOf rdf:resource="#Componente_Regiao_Corais"/>
</owl:Class>
<owl:Class rdf:ID="Pedras_Mais_Rasas">
  <rdfs:subClassOf rdf:resource="#Componente_Regiao_Corais"/>
  <owl:sameAs
rdf:resource="http://localhost:8080/Onto_Geologo.owl#Retrorecife"/>
</owl:Class>
<owl:Class rdf:about="#Animais_Marinhos">
  <rdfs:subClassOf rdf:resource="#Organismos_Marinhos"/>
</owl:Class>
<owl:Class rdf:ID="Regiao_de_Corais">
  <rdfs:comment
rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >Região de Corais é um Multipolígono</rdfs:comment>
  <owl:sameAs
rdf:resource="http://localhost:8080/Onto_Geologo.owl#Recife"/>
</owl:Class>
<owl:Class rdf:ID="Praia">
  <rdfs:subClassOf rdf:resource="#Ecossistema"/>
  <owl:sameAs
rdf:resource="http://localhost:8080/Onto_Geologo.owl#Transicional"/>
</owl:Class>
<owl:Class rdf:ID="Pedras_Visiveis">
  <rdfs:subClassOf rdf:resource="#Componente_Regiao_Corais"/>
  <owl:sameAs
rdf:resource="http://localhost:8080/Onto_Geologo.owl#Crista"/>
</owl:Class>
<owl:Class rdf:ID="Piscinas_Naturais"/>
<owl:Class rdf:about="#Corais">
  <owl:sameAs rdf:resource="#Pedra"/>
</owl:Class>
<owl:ObjectProperty rdf:ID="possui_componente">
  <rdfs:range rdf:resource="#Componente_Regiao_Corais"/>
  <owl:inverseOf>
    <owl:ObjectProperty rdf:ID="parte_de"/>
  </owl:inverseOf>
  <rdfs:domain rdf:resource="#Regiao_de_Corais"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="ao_redor_de">
  <rdfs:range rdf:resource="#Corais"/>
  <rdfs:domain rdf:resource="#Piscinas_Naturais"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="Onto_Turista_ObjectProperty_16"/>
<owl:ObjectProperty rdf:ID="habitam">
  <rdfs:domain rdf:resource="#Organismos_Marinhos"/>
  <rdfs:range rdf:resource="#Regiao_de_Corais"/>
  <owl:inverseOf>
    <owl:ObjectProperty rdf:ID="habitado_por"/>
  </owl:inverseOf>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="possui">
  <owl:inverseOf>
    <owl:ObjectProperty rdf:ID="esta_em"/>
  </owl:inverseOf>
  <rdfs:domain rdf:resource="#Mar"/>

```

---

```

    <rdfs:range rdf:resource="#Regiao_de_Corais"/>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:about="#parte_de">
    <rdfs:domain rdf:resource="#Componente_Regiao_Corais"/>
    <rdfs:range rdf:resource="#Regiao_de_Corais"/>
    <owl:inverseOf rdf:resource="#possui_componente"/>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:ID="dentro_de">
    <rdfs:range rdf:resource="#Regiao_de_Corais"/>
    <rdfs:domain rdf:resource="#Corais"/>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:ID="contem">
    <owl:inverseOf>
      <owl:ObjectProperty rdf:ID="esta_contido"/>
    </owl:inverseOf>
    <rdfs:range rdf:resource="#Piscinas_Naturais"/>
    <rdfs:domain rdf:resource="#Mar"/>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:about="#esta_contido">
    <owl:inverseOf rdf:resource="#contem"/>
    <rdfs:range rdf:resource="#Mar"/>
    <rdfs:domain rdf:resource="#Piscinas_Naturais"/>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:about="#habitado_por">
    <owl:inverseOf rdf:resource="#habitam"/>
    <rdfs:domain rdf:resource="#Regiao_de_Corais"/>
    <rdfs:range rdf:resource="#Organismos_Marinhos"/>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:about="#esta_em">
    <owl:inverseOf rdf:resource="#possui"/>
    <rdfs:range rdf:resource="#Mar"/>
    <rdfs:domain rdf:resource="#Regiao_de_Corais"/>
  </owl:ObjectProperty>
  <owl:DatatypeProperty rdf:ID="profundidade">
    <rdfs:domain rdf:resource="#Piscinas_Naturais"/>
    <rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#float"/>
  </owl:DatatypeProperty>
</rdf:RDF>

<?xml version="1.0"?>
<rdf:RDF
  xmlns="http://localhost:8080/Onto_Turista_Atracoes.owl#"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xml:base="http://localhost:8080/Onto_Turista_Atracoes.owl">
  <owl:Ontology rdf:about=""/>
  <owl:Class rdf:ID="Navegar">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="Atracao_Turistica"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="Nadar">
    <rdfs:subClassOf rdf:resource="#Atracao_Turistica"/>
  </owl:Class>
  <owl:Class rdf:ID="Mergulhar">
    <rdfs:subClassOf rdf:resource="#Atracao_Turistica"/>
  </owl:Class>
  <owl:Class rdf:ID="Piscinas_Naturais">

```

---

```

    <owl:sameAs
rdf:resource="http://localhost:8080/Onto_Turista.owl#Piscinas_Naturais
"/>
  </owl:Class>
  <owl:Class rdf:ID="Organismos_Marinhos">
    <owl:sameAs
rdf:resource="http://localhost:8080/Onto_Turista.owl#Organismos_Marinh
os"/>
  </owl:Class>
  <owl:Class rdf:ID="Flutuante">
    <owl:sameAs
rdf:resource="http://localhost:8080/Onto_Geologo.owl#Flutuante"/>
  </owl:Class>
  <owl:Class rdf:ID="Banho">
    <rdfs:subClassOf rdf:resource="#Atracao_Turistica"/>
  </owl:Class>
  <owl:Class rdf:ID="Regiao_de_Corais">
    <owl:sameAs
rdf:resource="http://localhost:8080/Onto_Geologo.owl#Corpo_Coralineo"/
>
  </owl:Class>
  <owl:Class rdf:ID="Rota_de_Barco">
    <owl:sameAs
rdf:resource="http://localhost:8080/Onto_Geologo.owl#Rota_de_Barco"/>
  </owl:Class>
  <owl:ObjectProperty rdf:ID="em">
    <rdfs:domain rdf:resource="#Navegar"/>
    <rdfs:range rdf:resource="#Rota_de_Barco"/>
  </owl:ObjectProperty>
  <owl:ObjectProperty
rdf:ID="Onto_Turista_Atracoes_ObjectProperty_7"/>
  <owl:ObjectProperty rdf:ID="explora">
    <rdfs:range rdf:resource="#Organismos_Marinhos"/>
    <rdfs:domain rdf:resource="#Mergulhar"/>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:ID="ao_redor_de">
    <rdfs:domain rdf:resource="#Banho"/>
    <rdfs:range rdf:resource="#Flutuante"/>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:ID="local_banho">
    <rdfs:range rdf:resource="#Piscinas_Naturais"/>
    <rdfs:domain rdf:resource="#Banho"/>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:ID="disjunto">
    <rdfs:domain rdf:resource="#Nadar"/>
    <rdfs:range rdf:resource="#Regiao_de_Corais"/>
  </owl:ObjectProperty>
  <owl:DatatypeProperty rdf:ID="profundidade">
    <rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#float"/>
    <rdfs:domain rdf:resource="#Mergulhar"/>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="db_info">
    <rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <rdfs:domain rdf:resource="#Rota_de_Barco"/>
  </owl:DatatypeProperty>
</rdf:RDF>

<!-- Created with Protege (with OWL Plugin 1.3, Build 225.4)
http://protege.stanford.edu -->

```

---

- **Ontologia do Biólogo**

```
<?xml version="1.0"?>
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xmlns="http://www.owl-ontologies.com/unnamed.owl#"
  xml:base="http://www.owl-ontologies.com/unnamed.owl">
  <owl:Ontology rdf:about=""/>
  <owl:Class rdf:ID="Molusco">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="Invertebrado"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="Peixes">
    <owl:sameAs>
      <owl:Class rdf:ID="Peixe"/>
    </owl:sameAs>
    <rdfs:subClassOf>
      <owl:Class rdf:ID="Organismos_Bentonicos"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="Rocha">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="Substrato_Consolidado"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="Vegetais_Superiores">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="Plantas"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="Lama">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="Substrato_Inconsolidado"/>
    </rdfs:subClassOf>
    <owl:sameAs
rdf:resource="http://localhost:8080/Onto_Geologo.owl#Lama" >
      </owl:sameAs>
    </owl:Class>
  <owl:Class rdf:ID="Recife"/>
  <owl:Class rdf:ID="Fitoplancton">
    <rdfs:subClassOf rdf:resource="#Plancton"/>
    <owl:sameAs
rdf:resource="http://localhost:8080/Onto_Geologo.owl#Vegetacao" >
      </owl:sameAs>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="Larvas">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="Organismos_Pelagicos"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:about="#Plantas">
    <rdfs:subClassOf rdf:resource="#Organismos_Bentonicos"/>
  </owl:Class>
</rdf:RDF>
```

---

```

    <owl:sameAs
rdf:resource="http://localhost:8080/Onto_Geologo.owl#Vegetacao" >
    </owl:sameAs>
</owl:Class>
<owl:Class rdf:about="#Plancton">
    <rdfs:comment
rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >organismos microsc3picos (plantas ou animais   " podem ser ovos,
larvas, min3sculos crust3ceos, que vivem flutuando ao sabor das
correntes. Se forem microalgas, ser3; o fitopl3ncton, se animais   "
zoopl3ncton. </rdfs:comment>
    <rdfs:subClassOf rdf:resource="#Organismos_Pelagicos"/>
</owl:Class>
<owl:Class rdf:ID="Crustaceo">
    <rdfs:subClassOf>
        <owl:Class rdf:about="#Invertebrado"/>
    </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Esponja">
    <rdfs:subClassOf>
        <owl:Class rdf:about="#Invertebrado"/>
    </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Algas">
    <rdfs:subClassOf rdf:resource="#Plantas"/>
</owl:Class>
<owl:Class rdf:ID="Recife_de_Coral">
    <rdfs:subClassOf rdf:resource="#Substrato_Consolidado"/>
    <owl:sameAs
rdf:resource="http://localhost:8080/Onto_Geologo.owl#Corpo_Coralineo"
>
    </owl:sameAs>
</owl:Class>
<owl:Class rdf:about="#Peixe">
    <owl:sameAs rdf:resource="#Peixes"/>
    <rdfs:subClassOf rdf:resource="#Organismos_Pelagicos"/>
</owl:Class>
<owl:Class rdf:ID="Cascalho">
    <rdfs:subClassOf rdf:resource="#Substrato_Inconsolidado"/>
    <owl:sameAs
rdf:resource="http://localhost:8080/Onto_Geologo.owl#Cascalho">
    </owl:sameAs>
</owl:Class>
<owl:Class rdf:ID="Equinodermos">
    <rdfs:subClassOf>
        <owl:Class rdf:about="#Invertebrado"/>
    </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Areia">
    <rdfs:subClassOf rdf:resource="#Substrato_Inconsolidado"/>
    <owl:sameAs
rdf:resource="http://localhost:8080/Onto_Geologo.owl#Areia" >
    </owl:sameAs>
</owl:Class>
<owl:Class rdf:ID="Invertebrados">
    <owl:sameAs>
        <owl:Class rdf:about="#Invertebrado"/>
    </owl:sameAs>
    <rdfs:subClassOf rdf:resource="#Organismos_Bentonicos"/>
</owl:Class>
<owl:Class rdf:ID="Vertebrados_Marinhos">

```

---

```

    <rdfs:subClassOf rdf:resource="#Organismos_Pelagicos"/>
  </owl:Class>
  <owl:Class rdf:ID="Zooplanton">
    <rdfs:subClassOf rdf:resource="#Plancton"/>
  </owl:Class>
  <owl:Class rdf:about="#Invertebrado">
    <owl:sameAs rdf:resource="#Invertebrados"/>
    <rdfs:subClassOf rdf:resource="#Organismos_Pelagicos"/>
  </owl:Class>
  <owl:ObjectProperty rdf:ID="parte_de">
    <rdfs:range rdf:resource="#Recife"/>
    <owl:inverseOf>
      <owl:ObjectProperty rdf:ID="comunidade_composta_por"/>
    </owl:inverseOf>
    <rdfs:domain>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Organismos_Bentonicos"/>
          <owl:Class rdf:about="#Organismos_Pelagicos"/>
        </owl:unionOf>
      </owl:Class>
    </rdfs:domain>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:ID="vivem_sobre">
    <owl:inverseOf>
      <owl:ObjectProperty rdf:ID="eh_abrigo_para"/>
    </owl:inverseOf>
    <rdfs:domain rdf:resource="#Organismos_Bentonicos"/>
    <rdfs:range>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Substrato_Consolidado"/>
          <owl:Class rdf:about="#Substrato_Inconsolidado"/>
        </owl:unionOf>
      </owl:Class>
    </rdfs:range>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:about="#comunidade_composta_por">
    <rdfs:domain rdf:resource="#Recife"/>
    <owl:inverseOf rdf:resource="#parte_de"/>
    <rdfs:range>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Organismos_Bentonicos"/>
          <owl:Class rdf:about="#Organismos_Pelagicos"/>
        </owl:unionOf>
      </owl:Class>
    </rdfs:range>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:about="#eh_abrigo_para">
    <rdfs:range rdf:resource="#Organismos_Bentonicos"/>
    <owl:inverseOf rdf:resource="#vivem_sobre"/>
    <rdfs:domain>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <owl:Class rdf:about="#Substrato_Consolidado"/>
          <owl:Class rdf:about="#Substrato_Inconsolidado"/>
        </owl:unionOf>
      </owl:Class>
    </rdfs:domain>
  </owl:ObjectProperty>

```

---

</rdf:RDF>

<!-- Created with Protege (with OWL Plugin 1.3, Build 225.4)  
http://protege.stanford.edu -->

---

## **Referências Bibliográficas**

- [Amaral 2002] Amaral R.F. 2002. Mapeamento da Área de Proteção Ambiental dos Recifes de Corais/RN - O Baixo de Maracajaú. Relatório técnico. IDEMA, Jan 2002, 50 p.
- [Breitman and Casanova 2005] Breitman, K. K. and Casanova, M. A. 2005. "Desenvolvimento de Ontologias para Engenharia de Software e Banco de Dados: Um tutorial Prático". Tutorial Apresentado no XIX Simpósio Brasileiro de Engenharia de Software. Uberlândia – MG.
- [Cabral 2006] Cabral, I. P. S. 2006. "Uma abordagem para manipulação de informações geográficas utilizando Scalable Vectors Graphics". Dissertação de Mestrado. Universidade Federal do Rio Grande do Norte.
- [Câmara *et.al* 1996] Câmara G., Casanova M., Hemerly A., Magalhães G., Medeiros C. (1996) Anatomia de Sistemas de Informações Geográficas. In: Escola de Computação, IX. Campinas, SP. Livro On Line. Disponível em: <<http://www.dpi.inpe.br/gilberto/livro/anatomia.pdf>>.
- [Câmara *et.al* 2005] Câmara G., Casanova M., Davis C., Vinhas L., Queiroz G. (2005) Bancos de Dados Geográficos. Editora MundoGeo.
- [Cardoso *et. al* 2005] Cardoso, R. C., Souza, F. F., Salgado, A. C. (2005) Using Ontologies to Prospect Offers on the Web. In: 7th International Conference on Enterprise Information Systems, 2005, Miami. ICEIS 2005 - Software Agent and Internet Computing. Lisboa - Portugal : INSTICC, 2005. v. 4. p. 200-207.
- [Cunha 2005] Cunha, M. (2005) *Modelagem e Implementação de Banco de Dados Geográficos para Apoio a Tomada de Decisão – Caso Maracajaú*. Tese de Mestrado. Universidade Federal do Rio Grande do Norte.
- [DAML 2003] The DARPA Agent Markup Language. Disponível em <<http://www.daml.org>>. Acessado em 19/06/2006.
- [DAML+OIL 2001] DAM + OIL (2001). Reference Description. Disponível em <<http://www.w3.org/TR/daml+oil-reference>> . Acessado em: 19/06/2006.

- 
- [Dongilli et. al 2004] Paolo Dongilli, Enrico Franconi, and Sergio Tessaris. Semantics driven support for query formulation. In Proceeding of the 2004 international workshop on description logics (dl'04), 2004.
- [Egenhofer 2002] Egenhofer, M. J. Toward the Semantic Geospatial Web, in Proceedings of the Tenth ACM International Symposium on Advances in Geographic Information Systems 2002, McLean, Virginia, 2002
- [Egenhofer and Herring 1991] M. J. Egenhofer and J. Herring. (1991). *Categorizing Binary Topological Relations Between Regions, Lines, and Points in Geographic Databases*. Technical Report, University of Maine, Orono.
- [Feitoza 2005] Feitoza, B. M. 2005. Avaliação Rápida da Comunidade de Peixes Recifais da "Área de Uso Turístico Intensivo" No Parracho de Maracajaú, Rio Grande do Norte. Relatório Técnico do Projeto Corais de Maracajaú, UFRN/IDEMA. Natal – RN, 2005.
- [Fonseca 2001] Fonseca, F. (2001) *Ontology-Driven Geographic Information Systems*. Phd Thesis. University of Maine.
- [Franconi and Tessalis 2005] Enrico Franconi and Sergio Tessaris. A unified logical framework for rules (and queries) with ontologies. In *Rule Languages for Interoperability*, 2005.
- [Guarino 1998] Guarino, N. (1998) *Formal Ontology and Information Systems*. In: *Formal Ontology and Information Systems (FOI'S 98)*. Itália, 1998.
- [Gütting 1994] Gütting, R.H. (1994) *An Introduction to Spatial Database Systems*. Special Issue on Spatial Database Systems of the VLDB Journal, Vol 3, Nº 4.
- [IDEMA 2004] IDEMA - Instituto de Desenvolvimento Econômico e Meio Ambiente do RN. 2004. Unidades de Conservação, Subcoordenadoria de Gerenciamento Costeiro (SUGERCO). On Line. Disponível em: <http://www.rn.gov.br/secretarias/idema/apas/recife.asp>. Acesso em 20/05/2006.

- 
- [Jena 2004] JENA – A Semantic Web Framework for Java - Jena 2 Ontology API. (2004). Disponível em <<http://jena.sourceforge.net/ontology>>. Acessado em 19/06/2006.
- [Lima 2004] Lima JR, P. O. (2004). *GEOBR: intercâmbio sintático e semântico de dados espaciais*. Tese de Mestrado. INPE – São José dos Campos.
- [McKee and Buehler 1996] McKee, L. and Buehler, K. (1996) The Open GIS Guide. Open GIS Consortium, Inc, Wayland, MA.
- [Novello 2002] Novello, Taisa Carla. (2002) Ontologias, Sistemas baseados em conhecimento e modelos de banco de dados. Disponível em <[http://www.inf.ufrgs.br/~clesio/cmp151/cmp15120021/artigo\\_taisa.pdf](http://www.inf.ufrgs.br/~clesio/cmp151/cmp15120021/artigo_taisa.pdf)>. Acessado em: 19/06/2006.
- [OGC 1999] Open GIS Consortium, Simple Features Specification for SQL. Revision 1.1. May, 1999. <http://www.opengis.org/>.
- [OIL 2000] Description of OIL (2000). Disponível em <<http://www.ontoknowledge.org/oil>> Acessado em: 19/06/2006.
- [OILED 2003] OilEd - An ontology editor allowing the user to build ontologies using DAML+OIL (2003). Disponível em <<http://oiled.man.ac.uk/>>. Acessado em 19/06/2006.
- [OWL 2004] OWL WEB ONTOLOGY LANGUAGE GUIDE (2004). Disponível em <<http://www.w3.org/TR/owl-guide/>> . Acessado em: 19/06/2006.
- [Protégé 2005] PROTÉGÉ ONTOLOGY EDITOR (2005). Disponível em <<http://protege.stanford.edu/>>. Acessado em: 19/06/2006.
- [Racer 2005] Racer Systems (2005). Disponível em <<https://www.racer-systems.com/>>. Acessado em 19/06/2006.
- [RDF 2004] Resource Description Framework (RDF) (2004). Disponível em <<http://www.w3.org/TR/rdf-concepts/>>. Acessado em: 19/06/2006.
- [RDFS 2004] Resource Description Framework Schema Specification (RDFS) 1.0 (2004). Disponível em <<http://www.w3.org/TR/rdf-schema/>>. Acessado em: 19/06/2006.

- 
- [Soares 2002] Soares, Valéria G. (2002) GeoVisual – Um Ambiente de Consultas Visuais para Bancos de Dados Geográficos. Centro de Informática. Universidade Federal de Pernambuco. Tese de Doutorado.
- [Sondheim *et al.* 1999] Sondheim, M., Gardels, K and Buehler, K. (1999) GIS Interoperability. in: P. Longley, M. Goodchild , D. Maguire and D. Rhind. Geographical Information Systems - Principles and Technical Issues. John Wiley & Sons, New York, pp. 347-358.
- [Sure *et.al* 2002] Sure Y., Erdmann M., Angele J., Staab S. , Studer R., Wenke D. (2002) *OntoEdit: Collaborative Ontology Development for the Semantic Web*. First International Semantic Web Conference 2002. Sardinia, Itália.
- [Surveys 1994] Surveys and Resource Mapping Branch – Ministry of Environment, Lands and Parks, British Columbia – Canadá (1994). Spatial Archive and Interchange Format: Formal Definition 3.1.
- [Viegas and Soares 2006] Viegas, R., Soares, V. (2006). Querying a Geographic Database using an Ontology-Based Methodology. VIII Simpósio Brasileiro de Geoinformática. Campos do Jordão. Novembro de 2006.
- [W3C 2005] W3C - WORLD WIDE WEB CONSORTIUM (2005). Disponível em <<http://www.w3.org>>. Acessado em: 19/06/2006.
- [Yuan 1998] Yuan, M. Development of a global conceptual schema for interoperable geographic information. 1998. Disponível em: <http://bbq.ncgia.ucsb.edu/conf/interop97/program/papers/yuan/yuan.html>. Acessado em: 10/11/2005.

# Livros Grátis

( <http://www.livrosgratis.com.br> )

Milhares de Livros para Download:

[Baixar livros de Administração](#)

[Baixar livros de Agronomia](#)

[Baixar livros de Arquitetura](#)

[Baixar livros de Artes](#)

[Baixar livros de Astronomia](#)

[Baixar livros de Biologia Geral](#)

[Baixar livros de Ciência da Computação](#)

[Baixar livros de Ciência da Informação](#)

[Baixar livros de Ciência Política](#)

[Baixar livros de Ciências da Saúde](#)

[Baixar livros de Comunicação](#)

[Baixar livros do Conselho Nacional de Educação - CNE](#)

[Baixar livros de Defesa civil](#)

[Baixar livros de Direito](#)

[Baixar livros de Direitos humanos](#)

[Baixar livros de Economia](#)

[Baixar livros de Economia Doméstica](#)

[Baixar livros de Educação](#)

[Baixar livros de Educação - Trânsito](#)

[Baixar livros de Educação Física](#)

[Baixar livros de Engenharia Aeroespacial](#)

[Baixar livros de Farmácia](#)

[Baixar livros de Filosofia](#)

[Baixar livros de Física](#)

[Baixar livros de Geociências](#)

[Baixar livros de Geografia](#)

[Baixar livros de História](#)

[Baixar livros de Línguas](#)

[Baixar livros de Literatura](#)  
[Baixar livros de Literatura de Cordel](#)  
[Baixar livros de Literatura Infantil](#)  
[Baixar livros de Matemática](#)  
[Baixar livros de Medicina](#)  
[Baixar livros de Medicina Veterinária](#)  
[Baixar livros de Meio Ambiente](#)  
[Baixar livros de Meteorologia](#)  
[Baixar Monografias e TCC](#)  
[Baixar livros Multidisciplinar](#)  
[Baixar livros de Música](#)  
[Baixar livros de Psicologia](#)  
[Baixar livros de Química](#)  
[Baixar livros de Saúde Coletiva](#)  
[Baixar livros de Serviço Social](#)  
[Baixar livros de Sociologia](#)  
[Baixar livros de Teologia](#)  
[Baixar livros de Trabalho](#)  
[Baixar livros de Turismo](#)