Uma arquitetura para consultas a repositórios de biodiversidade na Web

Este exemplar corresponde à redação final da Dissertação devidamente corrigida e defendida por Luiz Celso Gomes Jr. e aprovada pela Banca Examinadora.

Campinas, 18 de Maio de 2007.

Profa. Dra. Claudia Bauzer Medeiros
(Orientadora)

Dissertação apresentada ao Instituto de Computação, UNICAMP, como requisito parcial para a obtenção do título de Mestre em Ciência da Computação.

Substitua pela ficha catalográfica

(Esta página deve ser o verso da página anterior mesmo no caso em que não se imprime frente e verso, i.é., até 100 páginas.)

Substitua pela folha com as assinaturas da banca

Instituto de Computação
Universidade Estadual de Campinas

Uma arquitetura para consultas a repositórios de biodiversidade na Web

Luiz Celso Gomes Jr.[1]

Abril de 2007

Banca Examinadora:

- Profa. Dra. Claudia Bauzer Medeiros (Orientadora)
- Prof. Dr. Marcos André Gonçalves
  DCC - UFMG
- Prof. Dr. Geovane Cayres Magalhães
  IC - UNICAMP
- Profa. Dra. Islene Calciolari Garcia (Suplente)
  IC - UNICAMP

[1]Este trabalho teve o apoio da CAPES e do projeto WeBios

Then the sky spoke to me in language clear,
familiar as the heart, than love more near.
The sky said to my soul, "You have what you desire!

"Know now that you are born along with these
clouds, winds, and stars, and ever moving seas
and forest dwellers. This your nature is.

"Lift up your heart again without fear,
sleep in the tomb, or breathe the living air,
this world you with the flower and with the tiger share".

Passion – Kathleen Raine

Resumo

A vida na Terra forma uma ampla e complexa rede de interações que alguns especialistas estimam conter até 80 milhões de espécies diferentes. Abordar o tema biodiversidade é essencialmente um esforço distribuído. Uma instituição de pesquisa, seja qual for seu tamanho, é capaz de lidar com apenas uma pequena fração desta variedade. Portanto, para conduzir pesquisas ecologicamente relevantes, é preciso coletar porções de informação sobre espécies e seus habitats em um grande número de instituições e correlacioná-las usando conhecimento geográfico, biológico e ecológico.

A distribuição e a heterogeneidade inerentes aos dados de biodiversidade impõem diversos desafios, por exemplo, como encontrar informação relevante na Web, como resolver divergências sintáticas e semânticas e como processar vários predicados ecológicos e espaciais. Esta dissertação apresenta uma arquitetura que explora avanços em interoperabilidade de dados e tecnologias da Web semântica para tratar destes desafios. A solução se baseia em ontologias e anotação de repositórios para prover compartilhamento e descoberta de dados, estimulando a pesquisa colaborativa em biodiversidade. Um protótipo usando dados reais implementa parte da arquitetura.

Abstract

Life on Earth forms a broad and complex network of interactions, which some experts estimate to be composed of up to 80 million different species. Tackling biodiversity is essentially a distributed effort. A research institution, no matter how big, can only deal with a small fraction of this variety. Therefore, to carry ecologically-relevant biodiversity research, one must collect chunks of information on species and their habitats from a large number of institutions and correlate them using geographic, biologic and ecological knowledge.

Distribution and heterogeneity inherent to biodiversity data pose several challenges, such as how to find relevant information on the Web, how to solve syntactic and semantic heterogeneity, and how to process a variety of ecological and spatial predicates. This dissertation presents an architecture that exploits advances in data interoperability and semantic Web technologies to meet these challenges. The solution relies on ontologies and annotated repositories to support data sharing, discovery and collaborative biodiversity research. A prototype using real data has implemented part of the architecture.

Agradecimentos

Mudar de cidade, deixar o trabalho, me distanciar dos amigos, voltar à rotina de livros e provas, encarar um novo grupo e nova área de pesquisa... Há pouco mais de dois anos tudo parecia muito incerto e intimidante. O tempo passou rápido e hoje eu não poderia estar mais satisfeito com aquela decisão – tudo graças a pessoas fantásticas que tenho a sorte e a honra de ter ao meu lado, às quais tributo mais profundas admiração e gratidão.

Professora Claudia Medeiros, (i) por ter me admitido como orientando, (ii) por ter me proporcionado o melhor ambiente de trabalho que alguém pode desejar, (iii) pela paciência, muitas vezes posta à prova, (iv) pelos exemplos de competência e responsabilidade acadêmica, (v) por ter me ensinado a escrever textos científicos com clareza, rigor e uma porção de algarismos romanos, (vi) por me incentivar a buscar sempre o melhor e, sobretudo, (vii) por ter sempre se preocupado com meu presente e futuro e ter feito tanto para que eu tenha a chance de obter o melhor deles.

Professor Ricardo Torres, pela competente atuação como mentor e coordenador no projeto WeBios, por ter sido para mim uma referência como pesquisador, e pelos conselhos amigos em momentos muito importantes.

Professor Sérgio Freitas, ex-orientador, eterno amigo. Pelos conselhos sempre sensatos, pelo apoio constante e abnegado nas esferas acadêmica, profissional e pessoal e, acima de tudo, por ter despertado em mim o interesse pela pesquisa científica.

Amigos do LIS, pelo estimulante ambiente de trabalho, pelas construtivas discussões técnicas, pelo altruísmo incondicional e pelos incontáveis momentos de alegria proporcionados pela companhia de vocês. Alan, André, Andréia, Carla, Cristiano, Danilo, Evandro, Fábio, Gilberto, Jaudete, Javier, Leonardo, Luciano, Nielsen, Ricardo, Rodrigo, Senra, novatos e transientes. Tenho enorme admiração pela competência e generosidade de vocês e enorme orgulho pelo grupo unido que somos. Será difícil ficar longe de vocês.

Aloir e Layne, Fabiano e Kaylla, Hélder e Edna, Júlio, Nélio, Thiago Meirelles, amigos de longa data. Pelo apoio, camaradagem e as ótimas lembranças que levo sempre comigo. É difícil ficar longe de vocês.

Marília, Thiago Thomes e Dorothy, por terem me recebido tão bem em Campinas e pela amizade neste tempo todo.

Helen, minha irmãzinha doce, amiga e conselheira. Por tudo que você é, não é, será, ou não. Te amo muito gatinha.

Meu pai, pelo apoio e confiança.

Professores Thomas Lewinsohn e Paulo Prado pelos fascinantes ensinamentos sobre ecologia e os desconsertantes caprichos da Mãe Natureza.

Professores e funcionários do IC, pela simpatia inabalável, competência e boa vontade.

Capes e Microsoft Research, pelo importante suporte financeiro.

Pessoas abnegadas que trabalham nos diversos projetos de software de código aberto usados na validação das idéias desenvolvidas nesta dissertação. A liberdade propiciada por estes projetos é fundamental para a pesquisa acadêmica.

Sumário

v

Resumo **vi**

Abstract **vii**

Agradecimentos **viii**

1 Introdução **1**

2 Conceitos e trabalhos correlatos **3**

2.1 Pesquisas em biodiversidade e o sistema WeBios 3

2.1.1 Pesquisas em biodiversidade 3

2.1.2 Relacionamentos e interações de espécies 4

2.1.3 Sistemas computacionais e pesquisas em biodiversidade 6

2.1.4 Compartilhamento de dados de biodiversidade 7

2.1.5 Dados geográficos e pesquisas em biodiversidade 9

2.1.6 O Sistema WeBios 9

2.2 Semântica e Dados 11

2.2.1 Definição e evolução da semântica no gerenciamento de dados . . . 11

2.2.2 Web semântica 14

2.2.3 Ontologias na Web semântica 15

2.3 Interoperabilidade e dados geográficos 18

2.3.1 Sistemas geográficos e o OGC 18

2.3.2 Serviços Geográficos 20

2.4 Processamento e otimização de consultas 25

2.4.1 Visão geral 25

2.4.2 Processamento de consultas distribuídas 27

2.4.3 Processamento de consultas espaciais 28

3 Arquitetura proposta **30**
3.1 Requisitos da arquitetura 30
3.2 Especificação da arquitetura 32
3.2.1 Visão geral da arquitetura 32
3.2.2 Arquitetura do serviço de processamento de consultas 33
3.2.3 Processamento de consultas 35
3.3 Exemplo de processamento de consulta 40

4 Aspectos de implementação **48**
4.1 Repositórios de dados 48
4.2 Interfaces de consulta 49
4.3 Serviço de processamento de consultas 51
4.4 Exemplo de processamento de uma consulta 53

5 Conclusões e extensões **60**
5.1 Contribuições 60
5.2 Extensões 62

Bibliografia **65**

Lista de Tabelas

2.1 Exemplos de campos do padrão Darwin Core 7
2.2 Comparação de construtores disponíveis em diferentes linguagens de modelagem de conhecimento e dados (baseada em [25]) 13

3.1 Resolução de ramos-folha de acordo com o tipo 38

Lista de Figuras

2.1 Interações entre espécies, baseada em [60] 5
2.2 Arquitetura do sistema WeBios 10
2.3 Distribuição de tecnologias e conceitos de acordo com o espectro semântico. 12
2.4 Pilha de tecnologias na arquitetura da Web semântica (baseada em [31] e [45]) 14
2.5 Exemplos de elementos que podem ser representados em RDF (a), RDF Schema (b) e OWL (c) 16
2.6 Exemplo de consulta SPARQL (os prefixos das URIs foram omitidos) . . . 18
2.7 Resposta SPARQL (os prefixos das URIs foram omitidos) à consulta da Figura 2.6 sobre os dados da Figura 2.5. 18
2.8 Exemplos de features (obtidos em [64]) 20
2.9 Exemplo de documento GML 21
2.10 Exemplo de requisição de feature 23
2.11 Exemplo de utilização de serviços Web geográficos (inspirado em [66]) . . . 25
2.12 Fases do processamento de consultas (retirada de [50]) 26

3.1 Visão geral das interações entre os elementos da arquitetura 33
3.2 Interações entre os elementos internos e externos da arquitetura do serviço 34
3.3 Exemplos de registros do catálogo de repositórios 35
3.4 Partes das ontologias taxonômica/ecológica e geográfica utilizadas (inspiradas nas ontologias do projeto SPIRE) 35
3.5 Fases do processamento de consultas 36
3.6 Consulta de exemplo em SPARQL equivalente a "retorne todos os registros de ocorrência de espécies e que são predadas pela mosca *Adaina Bipunctata* que foram encontradas *dentro da Mata Atlântica Paulista*" 41
3.7 Seqüência de configurações do grafo da consulta de exemplo (Figura 3.6) em iterações consecutivas do algoritmo de processamento. As setas se referem à semântica dos predicados e não implicam em orientação no grafo. Os prefixos das URIs foram omitidos. 42

3.8 Parte de uma consulta WFS para a obtenção de certas espécies numa determinada área . . . 45
3.9 Resultado GML para a consulta WFS contendo os dados de ocorrência de espécies . . . 46

4.1 Seleção dos dados utilizados no protótipo . . . 49
4.2 Interface de consultas de dados de ocorrência . . . 50
4.3 Interface de consultas SPARQL . . . 51
4.4 Fases do processamento de consultas no protótipo . . . 52
4.5 Consulta de exemplo em SPARQL equivalente a "obter todos os registros de espécies predadas por tephritídeos" . . . 53
4.6 Ontologias utilizadas no protótipo . . . 54
4.7 Resultado da consulta de exemplo retornado pelo catálogo (há um abuso da notação XML para namespaces para reduzir o tamanho das URIs) . . . 55
4.8 Consulta para obtenção dos repositórios . . . 56
4.9 Resultado da consulta para obtenção dos repositórios . . . 57
4.10 Consulta WFS para obtenção dos dados de ocorrência de espécies . . . 58
4.11 Resultado em GML retornado para a interface de consulta (vários elementos do padrão Darwin Core foram omitidos) . . . 59

Capítulo 1

Introdução

As pesquisas em biodiversidade procuram compreender a diversidade da vida e propor meios para preservá-la. A diversidade biológica tem impacto direto no equilíbrio ambiental e desempenha papel fundamental em diversos campos científicos (como no desenvolvimento de novas drogas). Especialistas demandam contínuo e crescente engajamento nesta área, sobretudo ao se considerar a atual taxa de extinção de espécies, dez mil vezes maior que em eras anteriores [92]. Fatores como estes têm motivado um grande volume de pesquisa em biodiversidade nas últimas décadas.

Estudos em biodiversidade se baseiam em diversos tipos de modelos para definir fatores como riqueza de espécies, abundância, endemismo, distribuição e diversas outras variáveis. Para compilar estes modelos, dados de ocorrência de espécies devem ser obtidos em diversas instituições e combinados com outros tipos de dados, como dados filogenéticos descrevendo relações evolutivas, dados taxonômicos para nomenclatura, dados que descrevem correlações ecológicas entre espécies, ou dados geográficos definindo as condições do habitat [51].

Estes dados são produzidos e analisados em um contexto complexo. Especialistas estimam que o número de espécies no planeta pode chegar a 80 milhões – apenas uma pequena porcentagem destas espécies já foi identificada [92]. O número de interações entre espécies é algumas ordens de grandeza maior, e seres vivos são encontrados nos mais diversos ambientes. Todos estes fatores impõem desafios para o compartilhamento, obtenção, integração e análise dos dados associados às pesquisas [44].

Tipicamente, sistemas de informação de biodiversidade provêem acesso a registros de ocorrência de espécies gerenciados por museus de história natural, grupos e instituições de pesquisa. Um registro de ocorrência de espécie armazena dados sobre alguma observação ou coleta de seres vivos, incluindo informações sobre a classificação taxonômica, responsáveis pela coleta, local e demais condições da coleta. Sistemas que lidam com este tipo de dado freqüentemente permitem buscas baseadas nos valores literais dos campos

dos registros e em alguns casos disponibilizam para o usuário alguma interação de natureza geográfica, como definição da área de interesse ou geração de mapas de distribuição [39]. Outros projetos disponibilizam dados sobre classificação taxonômica, filogenética ou função ecológica de espécies [71, 87]. Além disto, os dados geográficos necessários às pesquisas podem ser obtidos em portais geográficos na Internet [7]. Existe porém a demanda por alternativas que considerem todos estes tipos de dados simultaneamente, sob um ambiente computacional unificado, flexibilizando a especificação de consultas, obtenção e análise de dados referentes aos diversos conceitos envolvidos nas pesquisas em biodiversidade.

O objetivo da dissertação é propor uma arquitetura que preencha esta lacuna. A arquitetura proposta integra conceitos da Web semântica e padrões de interoperabilidade para permitir que pesquisadores elaborem consultas baseadas em predicados taxonômicos, ecológicos e geográficos. Isto permite a elaboração de consultas com predicados que usam termos associados à semântica das aplicações, incluindo interações entre as espécies. A proposta tem como cenário de aplicação grandes repositórios de museus e coleções mundiais bem como coleções locais.

Esta dissertação está associada ao projeto de eScience WeBios, que é uma parceria entre pesquisadores do Instituto de Computação e do Instituto de Biologia da UNICAMP. O objetivo do projeto é oferecer a cientistas que trabalham com questões ambientais e de biodiversidade ferramentas que permitam consultas multimodais a dados heterogêneos.

As principais contribuições desta dissertação são:

- A proposta de uma arquitetura que permite consultas baseadas em predicados taxonômicos, ecológicos e geográficos a dados de biodiversidade que estejam disponíveis em repositórios na Web;
- A análise e aplicação de conceitos da Web semântica e padrões de interoperabilidade no contexto das pesquisas em biodiversidade;
- A implementação parcial da arquitetura proposta, com base em dados reais fornecidos pelo Instituto de Biologia da UNICAMP.

O restante do texto está organizado da seguinte forma: O Capítulo 2 apresenta trabalhos correlatos e conceitos básicos sobre biodiversidade, Web semântica, padrões de interoperabilidade e processamento de consultas. O Capítulo 3 descreve a arquitetura proposta, detalhando os elementos constituintes e estratégias de processamento de consultas. O Capítulo 4 descreve os protótipos que foram implementados para a validação da proposta. Por fim, o Capítulo 5 traz as considerações finais e possíveis extensões para o trabalho.

Capítulo 2

Conceitos e trabalhos correlatos

Este capítulo apresenta conceitos básicos usados na dissertação envolvendo biodiversidade, aspectos semânticos no tratamento de dados, interoperabilidade e processamento de consultas. Estes conceitos motivam e representam a fundamentação teórica da arquitetura que será apresentada no Capítulo 3.

2.1 Pesquisas em biodiversidade e o sistema WeBios

2.1.1 Pesquisas em biodiversidade

A pesquisa em biodiversidade é um assunto de grande destaque. O termo *biodiversidade* se refere à medida da diversidade relativa entre organismos presentes em diferentes ecossistemas. Segundo [92], biodiversidade pode ser identificada em três níveis: diversidade de espécies, diversidade genética e diversidade de habitats. Preservar a biodiversidade nos ecossistemas é importante por dois motivos básicos: manutenção do equilíbrio ecológico e perpetuação do banco de dados genético das espécies [92].

A manutenção do equilíbrio ecológico evita, por exemplo, explosões populacionais que levam ao aparecimento de pragas; a pesquisa do material genético de uma espécie e suas manifestações nos indivíduos pode auxiliar o desenvolvimento de medicamentos e outras substâncias úteis. Indo além, o conteúdo genético de um espécime pode ser visto como um livro, muitas vezes reescrito e revisado, contendo informações sobre como sobreviver no planeta [22]. Grande parte das pesquisas médicas se baseia na hipótese de que estudando outras espécies é possível encontrar soluções para problemas humanos. Preservar a biodiversidade é preservar este valioso volume de informações e garantir a sua evolução.

A extinção de espécies causada pela destruição de habitats cresce a uma taxa que alguns avaliam como sendo 10.000 vezes maior que a taxa anterior à intervenção hu-

mana [92]. Isto mostra a urgência com a qual devem ser tomadas medidas que garantam manutenção da biodiversidade.

As pesquisas em biodiversidade buscam entender melhor o ecossistema e desenvolver formas de preservá-lo. Grande parte do trabalho nesse domínio se concentra em três frentes: a identificação de espécies, o estudo de suas distribuições geográficas e a compreensão dos relacionamentos das espécies com seu meio. Estas frentes estão relacionadas a três disciplinas da biologia [92], respectivamente: sistemática, biogeografia e ecologia. A sistemática é o ramo da biologia que emprega taxonomias e estudos de similaridades para compreender a evolução de grupos de organismos, identificá-los e classificá-los. A biogeografia [13] corresponde ao estudo científico da distribuição dos organismos no presente e no passado. Por fim, a ecologia [46] é a disciplina que explora os relacionamentos dos organismos entre si e entre o ambiente.

2.1.2 Relacionamentos e interações de espécies

As pesquisas em ecologia lidam com dois tipos básicos de interações: entre espécies e entre espécies e seus habitats. Estas interações moldam a distribuição das espécies e determinam a sua diversidade.

As principais interações interespecíficas, ou processo elementares, entre pares de espécies incluem competição, predação e mutualismo [60]. Estes tipos de interação implicam algum tipo de efeito direto na população de uma espécie participante sempre que há variação na população da outra. Interações mais complexas podem ser derivadas dos processos elementares. As *cadeias tróficas*, por exemplo, definem padrões de fluxo de nutrientes em uma seqüência de espécies organizadas conforme suas interações de presa-predador. A Figura 2.1, baseada em [60], apresenta algumas interações interespecíficas destacando o efeito do aumento da população de uma espécie sobre a população da outra. A figura mostra, por exemplo, que numa interação de predação um aumento no número de indivíduos da espécie predadora (1) implica em uma diminuição no número de indivíduos da espécie predada (2).

Interações interespecíficas formam uma extensa e intrincada rede. Havens (*apud* [60]), trabalhando no lago Okeechobee, Flórida, estimou em cerca de 25 mil o número de interações entre as aproximadamente 500 espécies identificadas na área. A despeito da ampla disponibilidade de dados publicados sobre interações interespecíficas, este tipo de informação em geral não é incluído nas implementações de bancos de dados para sistemas de biodiversidade.

A segunda forma de interação estudada pela ecologia, entre espécies e o meio ambiente, é determinante na definição da distribuição, abundância e diversidade das espécies. Estas interações direcionam processos evolutivos e podem até mesmo modificar o próprio

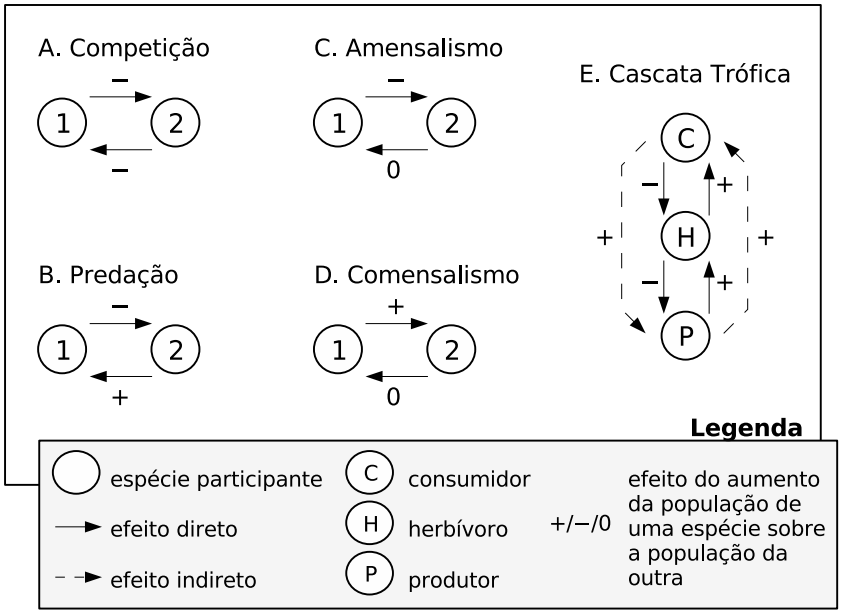

Figura 2.1: Interações entre espécies, baseada em [60]

ambiente [13]. Aqui, há a importante noção de *nicho* de uma espécie, que pode ser definido como um conjunto de variáveis ambientais que são relevantes para a sobrevivência dos organismos [81]. Muitas destas variáveis se referem a condições geográficas, como clima, altitude e solo. Portanto, a análise de dados geográficos é crítica em estudos de biodiversidade, como será visto na Seção 2.1.5.

Registros de ocorrência de espécies desempenham um papel importante neste contexto. Tais dados permitem estudos sobre padrões de distribuição de espécies (como em [72]) que possibilitam um melhor direcionamento para os esforços de conservação [61]. Registros de ocorrência de espécies descrevem procedimentos de coleta ou observação de seres vivos. Devem, por exemplo, conter informações sobre a classificação dos indivíduos em questão, a identificação dos pesquisadores responsáveis, a descrição da forma de coleta ou observação e a definição de sua localização geográfica. Estima-se que os museus de história natural do mundo contenham mais de três bilhões de registros de ocorrência de espécies, estimulando diversas iniciativas para a disponibilização destes dados [39].

Embora não possam ser classificados como interações, as relações taxonômicas entre espécies desempenham um papel chave na maioria dos estudos em biologia [92]. As classificações taxonômicas, introduzidas por Linnaeus no século XVIII, não representam papéis ecológicos; elas remetem a ligações históricas entre as linhagens das espécies e suas relações evolutivas. Na prática, muitas vezes as classificações taxonômicas são decorrentes de convenções históricas ou conveniência, o que leva a classificações redundantes e incompatíveis [38]. A unidade taxonômica básica é o *táxon*, que pode estar associado a

qualquer nível do sistema de classificação. Por exemplo, *Adaina bipunctata* é um táxon referente a uma espécie de uma mosca, enquanto *Tephritidae* é o táxon equivalente à sua família.

2.1.3 Sistemas computacionais e pesquisas em biodiversidade

Há diversos desafios para a realização de pesquisas em biodiversidade, desde a identificação das espécies à integração dos dados coletados. Com a evolução e disseminação da computação, diversos sistemas computacionais foram propostos e são utilizados para auxiliar nos diversos processos envolvidos [26, 11].

A identificação de espécies é um desafio chave e a base para toda pesquisa em biodiversidade. A dimensão do problema que se deve enfrentar, relacionada com a totalidade das espécies em questão, é um grande complicador. Wilson [92] estimou o número total de espécies de plantas, animais e microorganismos conhecidos pela ciência em cerca de 1,4 milhão. Porém, a estimativa do número total de espécies existentes está entre 10 e 80 milhões, dependendo do método estatístico utilizado [92]. Isto mostra o quanto ainda é desconhecido e dá uma noção da complexidade envolvida no processo de identificação de espécies. Diversas pesquisas utilizam programas de computador para auxiliar este processo. Entre as técnicas empregadas estão redes neurais e sistemas especialistas [26], que analisam características do espécime para identificação, e consultas baseadas em conteúdo de imagens, como em [86].

A integração dos dados coletados é, em geral, baseada nas propriedades espaciais e, portanto, Sistemas de Informação Geográfica (SIGs) e bancos de dados geográficos são essenciais. Estes permitem, por exemplo, estudos como o de [61], que identifica pontos prioritários de biodiversidade com o objetivo de direcionar adequadamente os recursos das iniciativas de preservação ambiental. Outra possibilidade é a previsão de pontos de ocorrência de espécies, como em [72].

A simulações de espécies e habitats representam outra área onde computadores são empregados. Autômatos celulares e IBMs (Individual-based Models) são exemplos de técnicas que são utilizadas neste contexto [11] e que possibilitam estudos como [47].

Esta dissertação está associada ao contexto dos sistemas empregados no armazenamento, gerenciamento e compartilhamento de dados envolvidos nas pesquisas em biodiversidade. As primeiras iniciativas de criação deste tipo de sistema na internet correspondem ao desenvolvimento de portais ou catálogos centralizados para os dados. Matias [53] faz uma extensa revisão destes sistemas para o contexto das pesquisas ambientais e de biodiversidade. Uma alternativa aos catálogos e portais centralizados que vem recebendo destaque nos últimos anos é o desenvolvimento infra-estruturas descentralizadas para compartilhamento de dados de biodiversidade. Esta é a estratégia adotada na ar-

Campo	Descrição	Exemplo
ScientificName	Táxon de mais baixo nível no qual o organismo foi identificado	*Ctenomys sociabilis* (Genus + SpecificEpithet)
CollectingMethod	O nome ou breve descrição do método ou protocolo usado na coleta	armadilha de raios UV, rede de arrastão
Collector	Nome(s) do(s) coletor(es)	Erica P. Anseloni
DecimalLatitude	Latitude do local no qual o organismo foi coletado, em graus decimais	23, 41

Tabela 2.1: Exemplos de campos do padrão Darwin Core

quitetura proposta nesta dissertação. Outros sistemas neste domínio são descritos na próxima seção.

2.1.4 Compartilhamento de dados de biodiversidade

Por conta da magnitude dos elementos envolvidos nas pesquisas em biodiversidade (milhões de espécies, interações e habitats), os cientistas e instituições nesta área tendem a focar seus esforços em porções menores do problema, se especializando em determinados táxons ou reduzindo a extensão geográfica das pesquisas. O compartilhamento de dados é fundamental para a realização de pesquisas mais abrangentes, possibilitando a análise de diversos tipos de espécies e ampliando o alcance geográfico. Há diversas iniciativas para possibilitar e promover o compartilhamento de dados de biodiversidade na Internet. Qualquer iniciativa neste sentido precisa lidar com questões relacionadas a representação e integração dos dados.

A abordagem típica para tratar problemas de interoperabilidade de dados se baseia na utilização consensual de padrões de representação. Diversos padrões foram propostos para representar diferentes tipos de dados associados às pesquisas em biodiversidade. Para dados de coleções ou observação de espécies, há o Darwin Core [85] (e suas variantes) e o ABCD (Access Biological Colections Data) [84]. Metadados ecológicos podem ser representados na Ecological Metadata Language (EML) [55]. O trabalho desta dissertação se concentrou no padrão Darwin Core para representação de registros de ocorrência de espécies. A Tabela 2.1 apresenta alguns exemplos de elementos contidos na especificação do padrão Darwin Core.

O passo posterior à utilização de padrões é a criação de mecanismos efetivos para possibilitar conectividade e compartilhamento dos dados. Esta área também é rica em

iniciativas, tanto em padrões de conectividade quanto em projetos para compartilhamento de dados entre instituições. Estas iniciativas acompanham tendências em diversas áreas, motivadas pela disseminação da Internet e necessidades de mecanismos de compartilhamento de dados pelas comunidades. Por exemplo, a comunidade de bibliotecas digitais investe na iniciativa OAI (Open Archive Initiative) [62] para mecanismos de compartilhamento e anotação de recursos.

DiGIR (Distributed Generic Information Retrieval) [24] é um protocolo que disponibiliza um ponto único de acesso a repositórios de dados distribuídos. Ele é baseado em HTTP, XML e UDDI para, respectivamente, transporte, representação e publicação de dados ou serviços. Implementações na área de biodiversidade que utilizam DiGIR estão localizadas principalmente na América do Norte, enquanto a União Européia começa a utilizar o padrão BioCASE [32]. Baseado em HTTP e XML, distingue-se do DiGIR por permitir que o provedor de dados selecione um esquema conceitual (tipicamente o ABCD) [39]. Há ainda a iniciativa TAPIR, patrocinada pelo GBIF (Global Biodiversity Information Facility) [35], para integração dos protocolos DiGIR e BioCASE.

A partir da adoção e especificação de padrões de conectividade, diversas iniciativas foram propostas para integração e compartilhamento dos dados provenientes de diferentes instituições. O Species Analyst [83], que utiliza o padrão DiGIR, é um exemplo. Ele é um projeto de pesquisa para desenvolvimento de padrões e ferramentas para acesso a bancos de dados de observações e coleções de museus de história natural na Web. Diversos projetos se baseiam no Species Analyst, como o MaNIS (Mammal Networked Information System) [52] e o GBIF.

No contexto da Web semântica (Seção 2.2.2), o projeto SPIRE [71] coordena o desenvolvimento e publicação de ontologias de conceitos taxonômicos, ecológicos e de modelagem de nicho. O projeto também disponibiliza ferramentas para consulta às ontologias e construção de cadeias tróficas.

Apesar da existência de diversos projetos para compartilhamento de dados de biodiversidade e do aparecimento de projetos para construção de ontologias neste domínio, ainda não há propostas que unifiquem estas duas vertentes. A proposta desta dissertação, como será visto no Capítulo 3, emprega ontologias num contexto de busca por dados de biodiversidade, flexibilizando as consultas possíveis. Desta forma é possível elaborar consultas envolvendo predicados taxonômicos e ecológicos definidos em ontologias. A arquitetura permite também a utilização de predicados geográficos. A relação entre dados geográficos e pesquisas em biodiversidade é descrita na próxima seção.

2.1.5 Dados geográficos e pesquisas em biodiversidade

A importância dos dados geográficos para as pesquisas em biodiversidade é destacada em [13]. Tipicamente, a posição geográfica é associada a registros de coleta ou observação de espécimes. O padrão Darwin Core confirma esta importância com a disponibilização de campos para registro de latitude e longitude. Já as áreas de ocorrências de espécies precisam ser representadas como polígonos georefencidados. Na variação do Darwin Core criada pelo OBIS (Ocean Biogeography Information System) [63] polígonos podem ser representados em GML (Geographic Markup Language) [65], a linguagem de representação de dados geográficos do OGC (Open Geospatial Consortium - ver Seção 2.3.1).

Dados geográficos também são importantes para a análise e estimativas de distribuição de espécies [72, 82] e para identificação de fatores que ameaçam a biodiversidade (sobretudo os ligados à ação humana [13]). Além disto, os SIGs desempenham um papel fundamental na integração dos dados das pesquisas e acompanham os pesquisadores desde o planejamento das atividades em campo à posterior inserção dos dados coletados. A geração de mapas temáticos e sua disponibilização são também considerados fator chave para as pesquisas em biodiversidade [39]. Neste trabalho, Guralnick et al. descrevem os desafios associados à construção de serviços globais de mapeamento em biodiversidade e sintetizam os esforços existentes neste contexto.

2.1.6 O Sistema WeBios

O projeto WeBios [58] é um proposta conjunta de pesquisadores do Instituto de Biologia e do Instituto de Computação da UNICAMP. Seu objetivo é prover um sistema que possibilite a especificação de consultas multimodais sobre fontes heterogêneas de dados de biodiversidade. Estas fontes incluem fotos (de seres vivos ou seus habitats), dados geográficos (mapas das regiões onde os seres foram encontrados), ontologias e metadados específicos de domínio (como descrições de habitats, ecossistemas e relações ecológicas).

O sistema permitirá que pesquisadores que lidam com questões relacionadas a ecossistemas e biodiversidade elaborem consultas combinando predicados baseados em conteúdo de imagens, predicados espaciais e predicados que explorem informações armazenadas em ontologias e metadados. Os dados obtidos nestas consultas contribuirão para uma melhor compreensão de questões ligadas à variedade de espécies, suas interações e seus habitats. Embora a especificação do projeto seja genérica, o sistema está sendo validado com dados de moscas (*Diptera*) e plantas hospedeiras fornecidos pelos pesquisadores do Instituto de Biologia.

A Figura 2.2 apresenta a arquitetura do sistema WeBios. O sistema é baseado em serviços Web que são agrupados em quatro camadas distintas: armazenamento, serviços de suporte, serviços avançados e aplicação cliente. Intuitivamente, uma consulta na interface

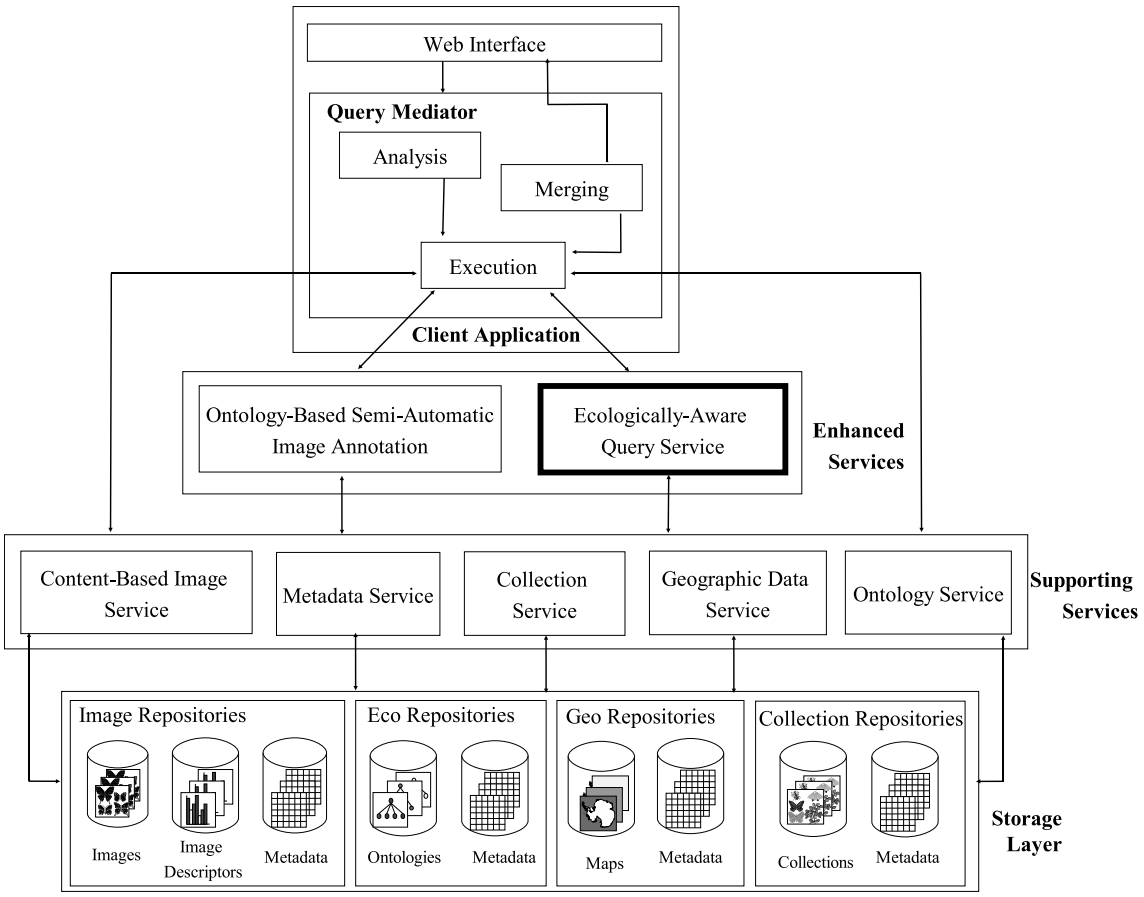

Figura 2.2: Arquitetura do sistema WeBios

Web é traduzida em uma série de novas requisições pelo mediador de consultas. As requisições são encaminhadas para as camadas de serviços avançados e serviços de suporte. Os serviços obtêm dados dos repositórios na camada de armazenamento, processam estes dados e retornam os resultados para o mediador, que integra os dados obtidos e retorna a resposta final à interface Web.

A pesquisa descrita nesta dissertação se concentra no serviço destacado na Figura 2.2: o serviço de consultas envolvendo relacionamentos ecológicos. O serviço de ontologias também desempenha um papel importante na criação, composição e gerenciamento das ontologias usadas no serviço de consultas. Este serviço de ontologias está associado a uma outra dissertação [21] em andamento no Instituto de Computação e não será abordado neste texto.

O serviço de consultas envolvendo relacionamentos ecológicos é classificado na arquitetura do sistema WeBios como um serviço avançado. Neste nível se encontra também o serviço de anotação semi-automática de imagens baseado em ontologias. Este serviço, associado à implementação do sistema OntoSAIA [33], utiliza um sistema de anotação colaborativa de imagens baseado em ontologias e busca por conteúdo de imagens para obtenção de imagens relevantes a partir de uma consulta.

2.2 Semântica e Dados

2.2.1 Definição e evolução da semântica no gerenciamento de dados

Nas últimas décadas, enquanto as pesquisas em bancos de dados avançavam na direção de prover mais semântica na modelagem dos dados e refletir esta semântica nos mecanismos de armazenamento, a comunidade de inteligência artificial desenvolvia técnicas adequadas à representação do conhecimento sobre conceitos e fenômenos do mundo real. Estas duas vertentes convergem no cenário atual de milhões de computadores interconectados numa rede global, onde dados precisam ser compartilhados e aplicações precisam ser interoperáveis. A demanda crescente por mecanismos que permitam interpretação semântica a dados e aplicações na Internet culminou na iniciativa para criação da Web semântica.

A Figura 2.3, retirada de [20], categoriza diversas tecnologias utilizadas para modelagem de dados com base na sua capacidade de expressar conteúdo semântico. Na figura, os pontos de destaque (como taxonomia e modelo conceitual) denotam elementos que introduziram conceitos importantes para expressividade semântica (como, respectivamente, "é subclassificação de" e "é subclasse de").

O modelo relacional (MR) [16] revolucionou as pesquisas em bancos de dados por se basear numa forte fundamentação teórica e possibilitar a aplicação de técnicas de oti-

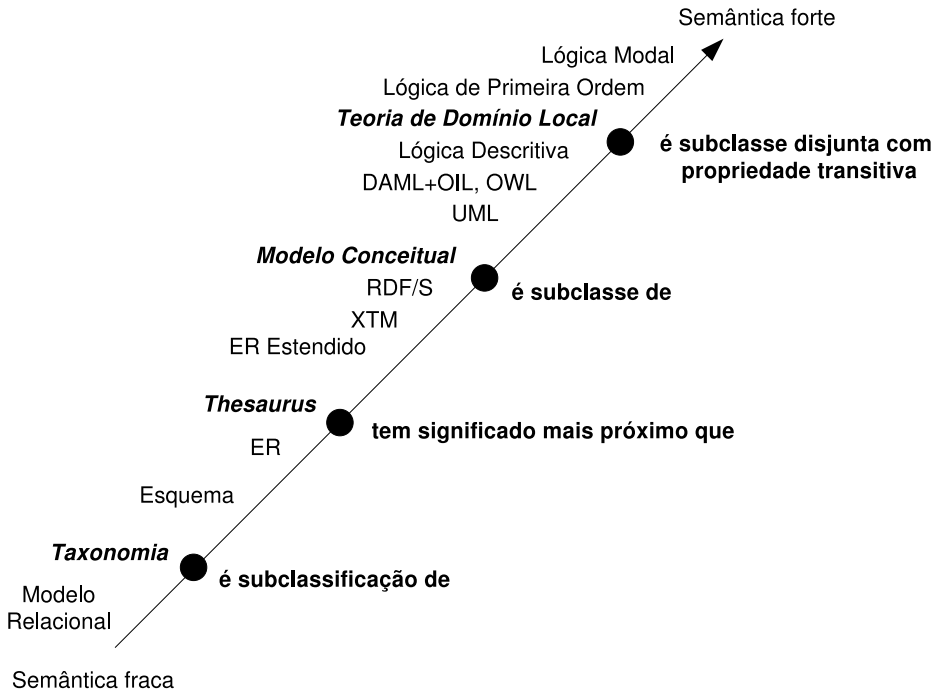

Figura 2.3: Distribuição de tecnologias e conceitos de acordo com o espectro semântico.

mização que viabilizam o gerenciamento de grandes volumes de dados. O modelo de entidade-relacionamento (ER) foi a proposta adotada para complementar o modelo relacional. A modelagem de dados no ER define entidades do domínio em questão, seus relacionamentos e restrições sobre estes relacionamentos. ER é, portanto, uma evolução semântica sobre MR. O mapeamento do ER para o MR é simples e em pouco tempo os desenvolvedores de sistemas de gerenciamento bancos de dados relacionais adotaram mecanismos para garantia de integridade nos relacionamentos (como *constraints* e *triggers*).

Bancos de dados dedutivos e bancos de dados orientados a objetos representam avanços no sentido de agregar mais semântica à especificação e manipulação dos dados. Bancos de dados dedutivos são bancos de dados que utilizam linguagens descritivas baseadas em lógica para expressar construções e consultas [30]. Bancos de dados orientados a objetos incorporam diversos conceitos amplamente utilizados em linguagens de programação orientadas a objetos, como hierarquia de classes e herança. Estas propostas influenciaram inovações nos SGBDs relacionais, como a introdução do modelo objeto-relacional e de consultas recursivas no SQL 1999.

No campo da inteligência artificial, uma das áreas de pesquisa diz respeito ao desenvolvimento de linguagens formais que representem conhecimento sobre o mundo. As características desejáveis para uma linguagem de representação de conhecimento são: ex-

Tipo de Construtor	Subtipo de Construtor	Construtor	RDF	RDF Schema	OWL	BD Relacional	BD Orientado a Objetos	Frame	Lógica Descritiva
Classe	Definição	Classe		X	X		X	X	X
		Classe Enumerada			X				O
		Restrição			X			O	X
		Interseção de			X			O	X
		União de, Complemento de			X			O	X
	Axioma	Subclasse de		X	X		X	X	X
		Igualdade			X			O	O
		Disjunto de			X			O	X
Relação	Definição	Propriedade	X	X	X	X	X	O	X
		Domínio, Abrangência		X	X			O	O
		Subpropriedade de		X	X				
	Axioma	Funcional (Inversa)			X	X			
		Igualdade, Inverso de			X				
		Transitiva, Simétrica			X				
Instância	Definição	Tipo	X	X	X		X	X	X
	Axioma	(Des)Igualdade			R		O	O	O

Legenda: X – Suportado **O** – Suporte Opcional **R** – Suportado com Restrições

Tabela 2.2: Comparação de construtores disponíveis em diferentes linguagens de modelagem de conhecimento e dados (baseada em [25])

pressividade, possibilidade de inferência de conhecimento e eficiência computacional [20]. As primeiras tentativas de representação de conhecimento utilizavam as redes semânticas [73], simples estruturas de grafos que demandavam estratégias ad hoc para inferência de conhecimento. A partir das redes semânticas evoluíram as representações baseadas em frames [59], utilizando um formalismo declarativo que permite a organização dos conceitos em múltiplas hierarquias. O avanço seguinte nas técnicas de representação de conhecimento foi a introdução de lógica descritiva, que permite uma formalização lógica do conhecimento, provê métodos de inferência robustos e é fundamentada em uma forte base teórica [20]. A linguagem OWL, atual padrão para representação de conhecimento, é baseada nos conceitos de frames e lógica descritiva e será analisada na próxima subseção.

A Tabela 2.2, baseada em [25], compara os construtores disponíveis em algumas linguagens de modelagem de conhecimento e dados. A tabela mostra, por exemplo, que OWL suporta a definição do axioma "Disjunto de" (equivalente a *disjointWith* na definição da linguagem) na construção de classes.

Diversos trabalhos unificam conceitos de bancos de dados e representação de conhecimento nas chamadas bases de conhecimento. Exemplo são os projetos MADS, DOGMA e KAON, descritos em [17].

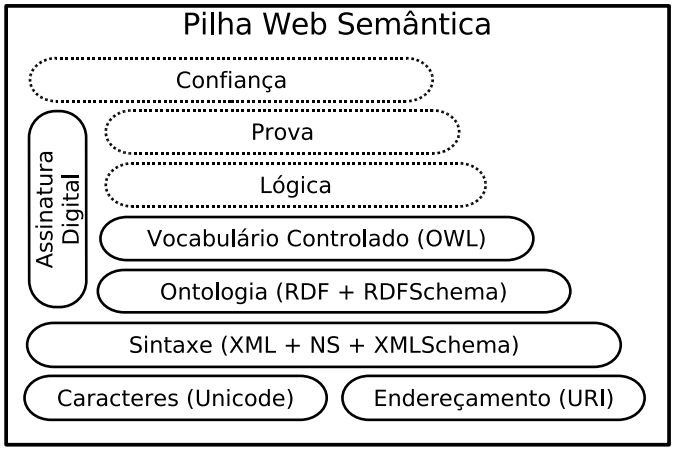

Figura 2.4: Pilha de tecnologias na arquitetura da Web semântica (baseada em [31] e [45])

2.2.2 Web semântica

A Web semântica é uma iniciativa coordenada pela W3C (World Wide Web Consortium) [91] cujo objetivo é ampliar a autonomia de agentes (módulos, aplicações ou serviços) na Internet permitindo que eles possam, automaticamente, encontrar parceiros e que tenham liberdade para iniciar e manter negociações de forma a atingir suas metas [9]. Para que a autonomia desejada seja possível, tais agentes necessitam de (i) protocolos padrão para interoperabilidade, permitindo que agentes desenvolvidos por diferentes organizações e utilizando diversas tecnologias possam se comunicar; e (ii) mecanismos que possibilitem a inferência de diversas características de possíveis parceiros, de forma com que um agente possa obter as informações necessárias para decidir-se por iniciar ou não o processo de negociação.

As tecnologias utilizadas para interoperabilidade entre agentes na Internet são agrupadas sob o conceito de *serviços Web*. Eles representam um conjunto de protocolos e padrões que permitem que aplicações sejam localizadas, descritas e acessadas de forma homogênea na Internet. Este texto não cobrirá detalhes da especificação dos serviços Web. Para mais informações, recomendamos a leitura de [3] e [20].

A Figura 2.4, baseada em [31] e [45], apresenta a pilha de tecnologias associadas às especificações da Web semântica. Na figura, cada nível é construído com base nas definições do nível anterior. A eXtensible Markup Language (XML) [90] provê uniformidade sintática aos dados, é baseada na codificação de caracteres Unicode e utiliza URIs (Universal Resource Identifiers) para endereçamento. XML Schema provê estrutura aos dados definidos em XML, definindo tipos, elementos, constantes e cardinalidades. RDF (Resource Description Framework) e RDF Schema são responsáveis por relacionar recursos (como documentos, imagens, pessoas ou serviços) e provêem o primeiro nível de expressi-

vidade semântica para a especificação de *ontologias*. Uma maior expressividade semântica é obtida a partir do uso de OWL (Web Ontology Language). RDF, RDF Schema, OWL e tecnologias associadas serão descritas na próxima subseção. Os demais níveis da pilha (Lógica, Prova e Confiança) ainda não possuem implementação consensual e não serão abordados.

2.2.3 Ontologias na Web semântica

A capacidade de inferir características de dados e serviços, essencial no âmbito da Web semântica, deve estar apoiada numa bem estruturada base de informações sobre os sujeitos da inferência. Para isto, a Web semântica faz uso intensivo de ontologias [20, 78]. Derivadas das pesquisas sobre representação do conhecimento, ontologias no contexto da Web semântica são definidas como uma representação codificada dos termos e conceitos usados para descrever um domínio do conhecimento [20, 17]. Ontologias devem definir um vocabulário compartilhado por uma comunidade e devem permitir que novo conhecimento seja inferido a partir dos elementos descritos [25].

A linguagem padrão para codificação de ontologias na Web semântica é a OWL. Como a base para a construção de ontologias é o relacionamento de conceitos, OWL foi especificada sobre a estrutura da linguagem RDF.

RDF [49] é uma linguagem de propósito geral para representação e correlação de recursos na Web. Neste contexto, o termo *recurso* compreende qualquer elemento que possa ser representado por uma URI (Universal Resource Identification), como documentos, pessoas ou serviços. O modelo RDF define triplas que contêm um sujeito, um predicado e um objeto. Um conjunto de triplas forma um grafo RDF, no qual sujeitos e predicados correspondem a nós e predicados correspondem a arestas. A Figura 2.5a apresenta um grafo que pode ser representado em RDF contendo, por exemplo, o sujeito *Tamanduá Bandeira* e o predicado *predadorDe* relacionando o sujeito ao objeto *Formiga*.

RDF Schema [12] provê à RDF um vocabulário controlado que possibilita a criação de classes e definição de propriedades. Com isto, é possível criar hierarquias de classes e hierarquias de propriedades, bem como a definição de instâncias para as mesmas. No exemplo da Figura 2.5b, os novos construtores introduzidos pelo RDF Schema permitem representar a classe *Animal* e estabelecer *Tamanduá Bandeira* e *Formiga* como suas instâncias.

OWL [56] é a linguagem padrão especificada pelo W3C para criação de ontologias na Web semântica. A linguagem procura aliar alta expressividade, possibilitando rica inferência lógica, e decidibilidade computacional, evitando a possibilidade de especificação de paradoxos que originem inferências indeterminâveis [20]. Ela integra conceitos de representação de conhecimento baseada em *frames* e lógica descritiva. OWL estende

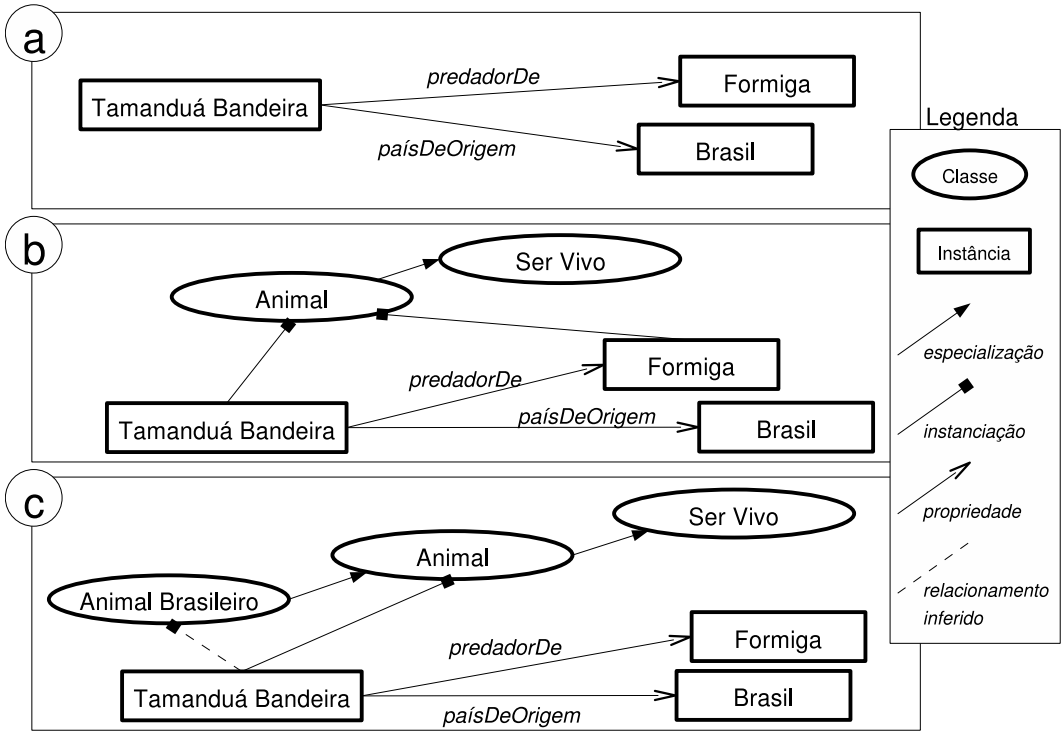

Figura 2.5: Exemplos de elementos que podem ser representados em RDF (a), RDF Schema (b) e OWL (c)

RDF Schema permitindo a definição de, por exemplo, classes a partir de construções lógicas (como união e interseção), transitividade e cardinalidade das propriedades [6]. Na Figura 2.5c, a classe *Animal Brasileiro* pode ser definida através de uma expressão de lógica descritiva que define a classe como sendo formada por todas as instâncias da classe *Animal* que possuem a propriedade *paísDeOrigem* apontando para a instância *Brasil*. A inferência sobre esta definição classifica a instância *Tamanduá Bandeira* como instância da classe *Animal Brasileiro*. Procedimentos de inferência em OWL são realizados por máquinas de inferência, como RACER [40] e Pellet [80].

Avanços significativos para a implementação da Web semântica estão sendo realizados, sobretudo no contexto da pesquisa em e-Science [78]. A partir da crescente disponibilidade de dados em OWL/RDF surge a necessidade de mecanismos para edição, armazenamento, manipulação e consulta destes dados.

A principal ferramenta para edição de ontologias é o Protégé [36], projeto de código aberto da Stanford University. Protégé permite a invocação de máquinas de inferência diretamente do ambiente de desenvolvimento. No campo do armazenamento e manipulação de ontologias, Jena [42] vem se destacando. Jena é um projeto de código aberto iniciado nos laboratórios de pesquisa da HP e oferece diversas possibilidades para manipulação de ontologias. Jena disponibiliza APIs para dados em RDF e OWL, permite o armazenamento dos dados em arquivos de sistema ou bancos de dados relacionais e é capaz de processar consultas SPARQL.

SPARQL (sigla definida recursivamente como SPARQL Protocol And RDF Query Language) [77] é o padrão em desenvolvimento pela W3C para consulta de dados em RDF/RDF Schema. SPARQL define uma linguagem de consultas e um padrão para representação dos dados de resposta. Consultas SPARQL são especificadas a partir de triplas RDF (sujeito, predicado e objeto) que são analisadas pelo processador de consultas. Além dos elementos básicos do RDF, uma tripla pode conter também variáveis, que são expandidas durante o processamento da consulta. Conjunções e disjunções de triplas compõem a cláusula WHERE da consulta. As variáveis que se deseja retornar devem ser incluídas na cláusula SELECT. A Figura 2.6 apresenta uma consulta SPARQL feita aos dados representados na Figura 2.5b. O resultado desejado é um conjunto de *animais* que são classificados como *predadores de* alguma espécie. Na consulta, o símbolo "?" define as variáveis enquanto ":" define elementos contidos na ontologia da Figura 2.5b. A palavra reservada "a" representa relacionamentos de instanciação (*is a*). A Figura 2.7 mostra o resultado da consulta, codificado em XML de acordo com a especificação da SPARQL. No exemplo, apenas a instância *Tamanduá Bandeira* atende às especificações da consulta. O código contém um cabeçalho (elemento head) que lista as variáveis de retorno especificadas na cláusula SELECT (no caso, apenas a variável *predador*). Cada resultado (elemento result) contém elementos resultantes do casamento entre as restrições

```
SELECT ?predador
WHERE { ?predador a :Animal .
        ?predador :predadorDe ?especie
}
```

Figura 2.6: Exemplo de consulta SPARQL (os prefixos das URIs foram omitidos)

```
<?xml version="1.0"?>
<sparql . . . >
  <head>
    <variable name="predador"/>
  </head>
  <results . . . >
    <result>
      <binding name="predador">
        <uri> . . . #Tamandua_Bandeira</uri>
      </binding>
    </result>
  </results>
</sparql>
```

Figura 2.7: Resposta SPARQL (os prefixos das URIs foram omitidos) à consulta da Figura 2.6 sobre os dados da Figura 2.5.

especificadas na cláusula WHERE e as variáveis da cláusula SELECT.

2.3 Interoperabilidade e dados geográficos

2.3.1 Sistemas geográficos e o OGC

Um Sistema de Informação Geográfica (SIG) é um sistema automatizado que possibilita a captura, modelagem, manipulação, recuperação, análise e apresentação de *dados geográficos*: dados que representam objetos e fenômenos em que a localização geográfica é uma característica inerente à informação e indispensável para analisá-la [95, 19]. Exemplos de aplicações que envolvem análise e planejamento sobre dados geográficos incluem o planejamento urbano, a definição de estratégias agrícolas, a localização de pontos de exploração mineral, o controle de epidemias e a análise de ecossistemas para preservação

ambiental.

No contexto das aplicações SIG, o mundo real é freqüentemente modelado segundo duas visões complementares: o modelo de campos e o modelo de objetos. O modelo de campos (em geral implementado segundo a chamada estrutura *raster*) enxerga o mundo como uma superfície contínua, sobre a qual os fenômenos geográficos a serem observados variam segundo diferentes distribuições (e.g. pressão atmosférica). Cada camada corresponde a um tema diferente (e.g. vegetação, solo). Já o modelo de objetos (em geral implementado usando representação vetorial) representa o mundo como uma superfície ocupada por objetos identificáveis, com geometria e características próprias e que existem independentemente de qualquer definição (e.g. um edifício ou um rio). As discussões da utilização apropriada de uma ou outra abordagem geraram o chamado "debate raster-vetor" [57].

Neste cenário de aplicações diversificadas e de maneiras muitas vezes conflitantes de modelagem do mundo real, surge a necessidade da especificação e utilização de padrões de interoperabilidade de dados e sistemas. Além disto, a disseminação dos sistemas na Web amplia a disponibilização de dados geográficos na rede. Nos últimos anos, indústria e academia têm se engajado num esforço conjunto com o objetivo de definir padrões que permitam o compartilhamento e reuso de dados e a integração de sistemas geográficos na Web.

O Open Geospatial Consortium (OGC) [70] é uma organização internacional que lidera o desenvolvimento de padrões para serviços geoespaciais e foi criado em resposta aos problemas de interoperabilidade disseminados na indústria. Sua visão é de um mundo onde todos possam se beneficiar de informações geográficas disponibilizadas para diversas redes, aplicações e plataformas [66]. Os padrões definidos pelo OGC têm o objetivo de simplificar o compartilhamento, comercialização e reuso de dados e sistemas geográficos na Web [64].

O modelo de referência especificado pelo OGC utiliza o conceito de *feature* geográfica como o ponto de partida para a modelagem de informação geoespacial [66]. Feature é a unidade fundamental de informação espacial e pode ser definida como uma abstração de um fenômeno do mundo real (ISO 19101) [64], correspondendo à noção de objeto espacial. Uma feature geográfica é uma feature associada a uma localização relativa à Terra.

Uma instância de uma feature corresponde a um fenômeno discreto; instâncias individuais são agrupadas em classes com características comuns: os *feature types*. Por exemplo, o fenômeno "Lagoa do Taquaral" (discreto) pode ser agrupado com outros fenômenos similares num feature type "Lagoa". Features podem ser definidas recursivamente, o que leva a grandes variações de granularidade. Por exemplo, dependendo da aplicação em questão, qualquer um dos itens da Figura 2.8 poderia ser representado como uma feature [64]. Cada elemento da figura corresponde a um fenômeno diferente, que pode ser

Um segmento de estrada entre intersecções consecutivas	Uma auto-estrada consistindo de vários segmentos de estrada	Uma estrada segmentada dinamicamente
Uma imagem de satélite georeferenciada	Um único pixel da imagem mencionada à esquerda	Uma rede de drenagem
A sobreposição das temperaturas em um mapa meteorológico	Uma malha triangular irregular	Um conjunto de contornos de magnitude de eventos sísmicos

Figura 2.8: Exemplos de features (obtidos em [64])

materializado em features de diferentes tipos.

A GML é uma linguagem baseada em XML para modelagem, transporte e armazenamento de informações geográficas, compreendendo propriedades espaciais e não espaciais das *features* geográficas. Sua especificação define a sintaxe XML Schema da linguagem, mecanismos e convenções que provêem um framework aberto e independente de fabricante para a definição de esquemas e objetos de aplicações geoespaciais [65]. A Figura 2.9 mostra um exemplo de documento GML codificando a geometria e demais atributos de uma ecorregião do Brasil. Na figura, o atributo *the_geom* contém as coordenadas que definem a geometria da região enquanto os atributos textuais *ECORREGION* e *MHT_NAME* descrevem o tipo e descrição da ecorregião.

Nos últimos anos, as tecnologias SIG evoluíram do tradicional modelo de sistemas *stand-alone*, nos quais os dados espaciais são fortemente acoplados aos sistemas que os criaram, para um modelo distribuído baseado em serviços Web geográficos providos de forma independente, especializados e interoperáveis [1]. O OGC especifica diversos padrões para serviços geográficos que viabilizam a criação de aplicações de acordo com este novo modelo.

2.3.2 Serviços Geográficos

Os serviços geográficos podem ser agrupados em três categorias [1, 66]:

- Serviços de dados: em geral acoplados a repositórios específicos, oferecem acesso a dados customizados. Exemplos incluem *Web Map Service* (WMS), que produz mapas de duas dimensões a partir de dados geoespaciais; *Web Coverage Service*

```
<gml:featureMember>
  <lis:wwf_eco fid="wwf_eco.9224">
    <lis:the_geom>
      <gml:MultiPolygon srsName=" . . . epsg.xml#4326">
        <gml:polygonMember>
          <gml:Polygon>
            <gml:outerBoundaryIs>
              <gml:LinearRing>
                <gml:coordinates . . . >
                  -3.31322472 -68.54711916,
                  -3.31118146 -68.54258721 . . .
                </gml:coordinates>
              </gml:LinearRing>
            </gml:outerBoundaryIs>
          </gml:Polygon>
        </gml:polygonMember>
      </gml:MultiPolygon>
    </lis:the_geom>
    <lis:ECOREGION>Purus varzea</lis:ECOREGION>
    <lis:MHT_NAME>Tropical and subtropical
      moist broadleaf forests
    </lis:MHT_NAME>
    . . .
  </lis:wwf_eco>
</gml:featureMember>
```

Figura 2.9: Exemplo de documento GML

(WCS), que provê acesso sob demanda a informações geoespaciais não renderizadas para renderização no cliente; e *Web Feature Service* (WFS), que permite que os clientes obtenham dados geoespaciais codificados em GML.

- Serviços de processamento: provêem operações para o processamento ou transformação segundo parâmetros especificados pelo usuário. Tais serviços podem prover funções genéricas de processamento como projeções e conversão de coordenadas, rasterização e vetorização, sobreposição de mapas, manipulação de imagens, detecção de objetos ou classificação de imagens. O modelo de referência do OGC [66] distingue serviços de apresentação (Portrayal Services), subconjunto dos serviços de processamento.

- Serviços de registro e catalogação: permitem que usuários e aplicações classifiquem, registrem, descrevam, busquem, mantenham e acessem informações sobre serviços Web geográficos.

Entre os padrões atualmente desenvolvidos e recomendados pelo OGC estão o WFS e o WMS, que utilizam GML para a codificação dos dados. Estes padrões são empregados nesta dissertação, sendo por isto descritos em detalhes.

O WFS (Web Feature Service) é uma especificação do OGC cujo objetivo é permitir que clientes obtenham dados geoespaciais codificados em GML a partir de múltiplos servidores de features [69]. Os Web Feature Services permitem operações de atualização, consulta e descoberta de features geográficas. Clientes de um WFS acessam dados destas features através da submissão de requisições que especificam as features desejadas [66]. A Figura 2.10 mostra um exemplo de requisição de feature que obtém todas as instâncias de ecorregiões (feature type *wwf_eco*) dentro de uma certa região geográfica (no caso, a geometria do Brasil). A cláusula *Filter*, que equivale à cláusula WHERE em SQL, permite a definição de restrições geográficas (como *Within* na figura), de casamento de strings, ou algébricas.

A especificação da interface de um WFS define cinco operações [69]: *GetCapabilities*, *DescribeFeatureType*, *GetFeature*, *Transaction* e *LockFeature*. *GetCapabilities* retorna metadados do serviço. Mais especificamente, indica quais feature types são providos e quais operações são suportadas em cada feature type. *DescribeFeatureType* retorna a descrição da estrutura de um feature type. *GetFeature* retorna instâncias de features, sendo que o cliente pode especificar quais propriedades devem ser obtidas e também restringir a consulta espacialmente ou através de atributos não espaciais. *Transaction* recebe as requisições de transações, compostas por operações que modificam features. Por fim, *LockFeature* permite o bloqueio de uma ou mais instâncias de features, garantindo o suporte a transações serializáveis.

```
<?xml version="1.0"?>
<wfs:GetFeature xmlns:topp="http://www.openplans.org/topp"
  xmlns:wfs="http://www.opengis.net/wfs"
  xmlns="http://www.opengis.net/ogc"
  xmlns:gml="http://www.opengis.net/gml"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  service="WFS" version="1.0.0" outputFormat="GML2"
  xsi:schemaLocation=" . . . /WFS-basic.xsd">
  <wfs:Query typeName="lis:wwf_eco">
    <Filter>
        <Within>
          <PropertyName>the_geom</PropertyName>
          <gml:Polygon>
            <gml:outerBoundaryIs>
              <gml:LinearRing>
                <gml:coordinates decimal="." cs="," ts=" ">
                  -44.69500732,-1.81777787
                  -44.48972702,-1.9866668
                  -44.65493393,-2.32368088
                  -44.45083618,-2.14638901 . . .
                </gml:coordinates>
              </gml:LinearRing>
            </gml:outerBoundaryIs>
          </gml:Polygon>
      </Within>
    </Filter>
  </wfs:Query>
</wfs:GetFeature>
```

Figura 2.10: Exemplo de requisição de feature

A partir de 2003, o OGC adotou o projeto GeoServer [37] como sua implementação de referência para WFS. O objetivo do projeto é prover uma implementação dos padrões WFS e WMS com código aberto e disponibilizada livremente. O GeoServer visa possibilitar a criação de uma rede em que usuários possam pesquisar e navegar pelos dados espaciais.

A especificação WMS (Web Map Service) do OGC padroniza a forma com que os clientes requisitam mapas. Cada requisição utiliza como parâmetros nomes de camadas e outros itens como o tamanho do mapa a ser retornado e o sistema de referenciamento espacial a ser utilizado para sua montagem [67].

A especificação da interface WMS define três operações: *GetCapabilities*, *GetMap* e *GetFeatureInfo* (opcional). *GetCapabilities* retorna metadados do serviço e dos dados disponíveis; *GetMap* retorna um mapa de acordo com os parâmetros especificados pelo cliente; e *GetFeatureInfo* retorna informações a respeito de uma feature específica mostrada no mapa. As operações dos Web Map Services podem ser disparadas por um navegador Web padrão através da submissão de requisições no formato de URLs [67, 66].

O modelo de referência OpenGIS do OGC [66] propõe ainda os serviços geográficos WCS, CPS, SCS, Geocoder, Gazetter e Geoparser. O WCS (Web Coverage Service) disponibiliza features contínuas (ou coberturas) não renderizadas que estão associadas ao modelo de campos (em oposição ao modelo de objetos utilizados no WFS). O CPS (Coverage Portrayal Service) estende a interface do WMS e é utilizado para renderização de features obtidas em um WCS. O SCS (Sensor Collection Service) visa acesso e manipulação de dados obtidos por sensores que podem ser encapsulados em coberturas ou features de medição. O Geocoder Service permite a associação de palavras, termos ou códigos em strings de texto a suas features geográficas – como na conversão de um endereço em uma localização geográfica. O Gazetteer Service é um caso especial de Geocoder Service, onde cada serviço está associado a um vocabulário de identificadores. O Geoparser Service processa documentos digitais para identificação de palavras-chave e frases que possuem conteúdo espacial. Além do WMS e WFS, apenas o WCS já é uma especificação adotada pelo OGC segundo a versão 0.1.3 do seu modelo de referência [66].

A Figura 2.11 mostra um exemplo de utilização de serviços Web geográficos num contexto de biodiversidade. Um biólogo utiliza seu navegador Web ① para obter a imagem de um mapa a partir de um visualizador de mapas ② como o MapServer [88]. Este mapa sobrepõe uma foto de satélite de uma área específica com um mapa da distribuição de uma determinada espécie. No exemplo, a foto de satélite com as características desejadas é requisitada ao WMS ③, que a obtém do repositório de imagens ④ e a processa de acordo com os parâmetros definidos pelo cliente (como limites de cobertura e formato de codificação da imagem gerada). Já o mapa de distribuição da espécie é obtido do WMS ⑤ a partir de uma consulta ao banco de dados geográfico ⑦ e também se baseia nos parâmetros definidos pelo cliente. Para incluir uma nova região de ocorrência da espécies,

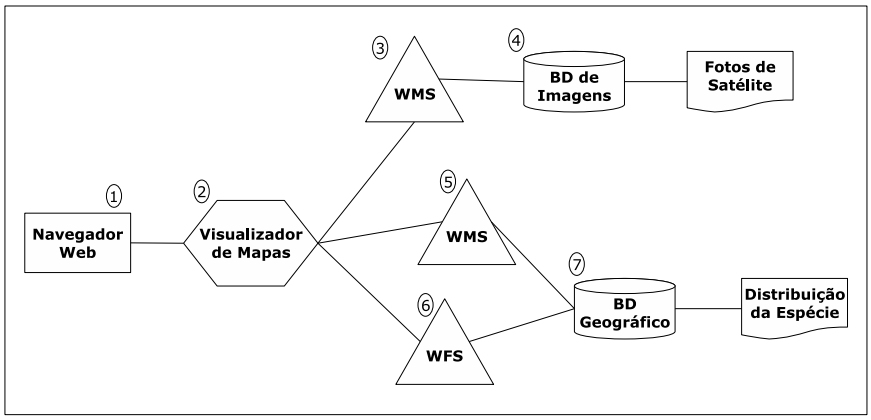

Figura 2.11: Exemplo de utilização de serviços Web geográficos (inspirado em [66])

o biólogo utiliza o visualizador de mapas ② para definir a região desejada bem como as propriedades desta nova instância de feature. A nova feature é então inserida no banco de dados geográfico ⑦ através de uma transação no WFS ⑥.

2.4 Processamento e otimização de consultas

Os sistemas de gerenciamento de bancos de dados atuais oferecem ao usuário linguagens não-procedurais (como SQL [43] ou OQL [2]) para a definição de consultas. Nestas linguagens, o usuário especifica *o que* deve ser encontrado, em vez de especificar *como* deve ser encontrado [94]. De fato, o usuário nem sequer precisa saber como ou onde os dados estão armazenados nestes sistemas. Além de simplificar a especificação das consultas, esta abordagem possibilita o desenvolvimento de diversos mecanismos de otimização das consultas e armazenamento dos dados nos sistemas. Esta seção descreve o processamento de consultas em SGBDs, técnicas de otimização (incluindo consultas geográficas) e aspectos relacionados a processamento de consultas em bancos de dados distribuídos.

2.4.1 Visão geral

O objetivo do processamento de uma linguagem de consulta não-procedural é obter um plano de execução que possa ser aplicado ao conjunto de dados de interesse para a obtenção dos resultados. Técnicas de otimização de consultas são aplicadas durante o processamento com o objetivo de tornar o plano de execução mais eficiente. A Figura 2.12 mostra a seqüência típica de fases para o processamento de consultas em um SGBD, que são descritas a seguir [30, 50].

A Parser: Analisa a consulta e a traduz para um representação interna (usualmente

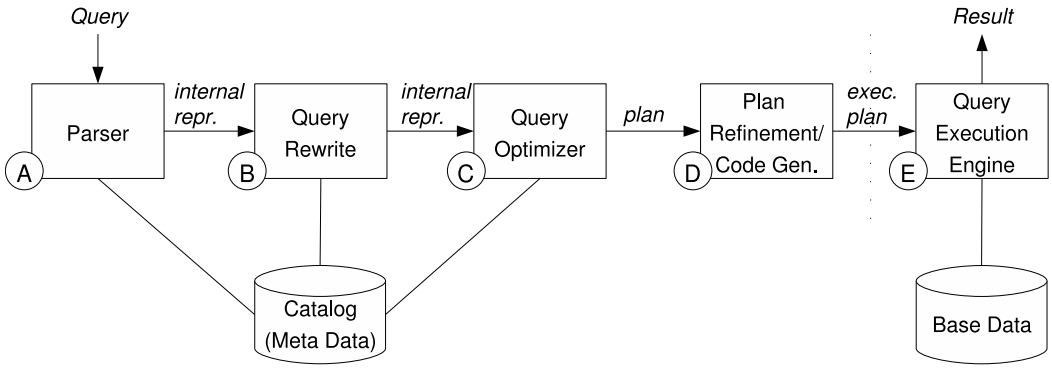

Figura 2.12: Fases do processamento de consultas (retirada de [50])

uma árvore ou um grafo) que facilite o processamento das fases seguintes.

B Query Rewrite: Transforma a consulta empregando otimizações que não dependem do estado físico dos dados, como eliminação de predicados redundantes e simplificação de expressões.

C Query Optimizer: Realiza as otimizações que dependem do estado físico dos dados, como disponibilidade de índices e tamanho das tabelas. Esta fase produz diversos planos alternativos para que um deles seja escolhido de acordo com o modelo de estimativa de custos.

D Plan Refinement/Code Generation: Transforma o plano produzido pelo otimizador em um código que possa ser executado.

E Query Execution Engine: Executa o código gerado na fase anterior. O mecanismo de execução implementa os operadores de acesso aos dados usados no código.

Em um SGBD, o catálogo armazena informações sobre o esquema dos dados, detalhes de armazenamento e estatísticas de ocupação. Estes dados são utilizados pelo processador na análise (Fase A), reescrita (Fase B) e otimização das consultas (Fase C).

Na fase de otimização de consultas, o processador utiliza heurísticas para transformar a árvore ou grafo de consulta construído nas fases A e B em uma estrutura equivalente que proporcione um processamento mais eficiente [30]. A abordagem típica utiliza uma árvore como representação interna da consulta. Uma árvore de consulta é uma estrutura que corresponde a uma expressão de álgebra relacional. Nesta estrutura, as folhas da árvore representam relações (tabelas) de entrada enquanto os nós internos representam operações de álgebra relacional (como seleções e junções). A fase de otimização aplica regras de transformação para organizar as operações de forma a minimizar os custos da consulta.

Outra abordagem para representação e otimização de consultas utiliza grafos [30]. Esta abordagem é interessante para tratamento de consultas em cálculo relacional ou linguagens similares (como QUEL [96]). A otimização dos grafos de consulta utiliza uma técnica chamada decomposição de consulta [94] que, basicamente, subdivide o grafo obtendo e resolvendo consultas mais simples até que a consulta original tenha sido completamente resolvida. O processador de consultas aplica heurísticas de otimização na subdivisão do grafo e na escolha das variáveis (relações) para substituição por tuplas obtidas na base de dados.

Há duas diferenças básicas entre a abordagem que utiliza uma árvore e a que utiliza um grafo para representar as consultas. Primeiramente, o grafo de consulta não impõe uma ordem de execução das operações, diferentemente do que acontece com a árvore. Além disto, o processamento de consultas baseado em grafo não segue os passos apresentados na Figura 2.12. Ao final da fase de otimização, em vez de produzir um plano de execução, o processador produz o resultado final da consulta. Isto porque o processo de substituição de variáveis obtém iterativamente as tuplas necessárias para o processamento da consulta.

2.4.2 Processamento de consultas distribuídas

Diversos fatores despertaram um grande interesse em pesquisas relacionadas a bancos de dados distribuídos nas últimas décadas. Exemplos são a distribuição geograficamente esparsa das organizações globais, a necessidade de integração interorganizacional de dados e sistemas legados, e a possibilidade de se evitar os altos custos e a falta de flexibilidade de grandes bases de dados centralizadas [50]. Avanços em tecnologias de comunicação, hardware e software fazem dos bancos de dados distribuídos uma alternativa viável atualmente. Prova disto é a inclusão de tecnologias que possibilitam distribuição nos SGDBs comerciais mais populares.

Diversas decisões devem ser tomadas na implementação de bancos de dados distribuídos, como sobre o que distribuir (apenas dados ou dados e processamento) e como distribuir os dados (distribuir o esquema ou distribuir tuplas de uma certa tabela). A partir da definição destes pontos, os desafios levantados estão associados à escolha de boas estratégias para o processamento de consultas, gerenciamento de transações e garantia de integridade. Este trabalho se concentra nos aspectos relacionados ao processamento das consultas.

As fases de processamento de consultas apresentadas na Figura 2.12 também se aplicam aos SGDBs distribuídos [50]. Porém, a distribuição introduz novos níveis de complexidade neste cenário. Por exemplo, o otimizador de consultas num contexto distribuído precisa decidir em quais pontos do sistema cada operação da consulta deve ser executada de forma a aumentar a eficiência do processo. Se a arquitetura do sistema de banco de

dados é cliente-servidor, uma decisão deve ser tomada com relação a quando utilizar os recursos do cliente em tarefas como materialização de junções ou *cache* de dados. O modelo de estimativa de custos, utilizado na otimização da consulta, deve considerar também os novos custos de comunicação introduzidos.

No casos de bancos de dados distribuídos em WANs (Wide Area Networks) como a Internet, a construção de modelos de custo é complicada porque envolve fatores de difícil previsão. O tempo de resposta para uma consulta pode variar sensivelmente por conta de variações na qualidade da comunicação associadas, por exemplo, a sobrecargas em alguns pontos da rede. Para lidar com este problema, Urhan et al. propõem uma técnica chamada *query scrambling* [89, 4]. O objetivo da técnica é reorganizar o plano da consulta em tempo de execução, adaptando-o em resposta a atrasos inesperados no tempo de resposta da execução dos operadores. Query scrambling é composta de duas fases diferentes. Na primeira, chamada de reagendamento, quando o processador detecta um atraso em um operador, ele reorganiza a ordem de execução no plano para que um outro operador seja escolhido para execução. A segunda fase, chamada síntese de operadores, é executada quando há ocorrência de atrasos mas todas as possibilidades de reagendamento da primeira fase foram exploradas. A síntese de operadores cria novos operadores, como, por exemplo, a junção de duas relações não existente no plano original.

2.4.3 Processamento de consultas espaciais

Os bancos de dados espaciais apresentam três diferenças básicas quando comparados aos bancos de dados tradicionais [79]: (i) não existe um conjunto fixo de operações usados como base para a avaliação das consultas; (ii) o banco é composto por grande volume de objetos complexos, que não podem ser ordenados de forma simples; e (iii) os algoritmos que testam predicados espaciais são computacionalmente caros, fazendo com que os custos de processamento muitas vezes dominem os custos de I/O. Diversas técnicas foram propostas para tratar estes problemas, algumas, especialmente as associadas ao contexto desta dissertação, serão discutidas a seguir.

Bancos de dados espaciais tipicamente permitem a utilização de predicados topológicos (e.g. overlap, inside) na especificação das consultas. Porém, não existe um formalismo padrão que associe o conjunto de possíveis predicados a suas respectivas interpretações semânticas. Isto faz com que o conjunto de predicados utilizado seja definido de acordo com o domínio da aplicação [15]. Independentemente da subjetividade da interpretação semântica dos predicados, as relações topológicas são restritas ao plano bidimensional, o que limita o número de possíveis relações entre os objetos. Egenhofer e Herring [29] categorizaram todas as relações possíveis entre pontos, linhas e áreas no modelo chamado 9-intersection. O modelo constrói a matriz de todas as interseções possíveis entre inte-

riores, bordas e exteriores de elementos espaciais. A partir desta matriz são obtidas as relações semanticamente consistentes e não redundantes. No caso de duas regiões geográficas, as relações obtidas são *disjoint*, *meet*, *contains*, *covers*, *inside*, *coveredBy*, *equal* e *overlap*. Estas relações podem ser combinadas para obtenção de conjuntos menores de operações adequados ao domínio da aplicação [15], como é feito em [14].

Algumas técnicas de otimização de consultas geográficas são baseadas no modelo 9-intersection. Por exemplo, é possível determinar o número mínimo de testes necessários para verificar cada relação entre dois objetos espaciais [15], o que otimiza o algoritmo responsável pelo teste. Considerando como exemplo a relação A *inside* B onde A e B são regiões espaciais, para verificar a validade da relação basta testar se a interseção entre a borda de A e o interior de B é não-vazia e se a interseção entre a borda de A e a borda de B é vazia. Neste caso, o algoritmo dispensa as outras sete interseções do modelo.

Uma outra abordagem, também baseada no modelo 9-intersection, é chamada de composição de relações binárias [27, 76]. O formalismo desenvolvido nesta técnica possibilita responder questões como "dado três objetos, A, B e C, e duas relações topológicas $A\ \mathrm{r_i} B$ e $B\ \mathrm{r}_j C$, qual a relação topológica $A\ \mathrm{r}_k C$?". A partir deste tipo de inferência é possível (i) integrar informações topológicas coletadas independentemente, (ii) detectar inconsistência em dados espaciais e (iii) simplificar, numa fase de pré-processamento, consultas espaciais compostas por restrições espaciais complexas [27], restritas no entanto a duas dimensões.

Resumo

Este capítulo apresentou conceitos utilizados na dissertação, associados à pesquisa em biodiversidade e sistemas computacionais para apoiá-la. Detalha problemas relativos à integração de dados e à interoperabilidade na Web, com ênfase na Web semântica. Por fim, apresenta uma visão geral de processamento de consultas em bancos de dados, incluindo soluções para dados geográficos, base para sistemas de biodiversidade.

O próximo capítulo apresenta a arquitetura proposta para processamento de consultas de biodiversidade, baseada nestes conceitos. O capítulo aborda a implementação da arquitetura, que utiliza padrões para representação e consulta de ontologias (OWL, RDF, SPARQL) e padrões para interoperabilidade de dados geográficos (GML e WFS).

Capítulo 3

Arquitetura proposta

Este capítulo apresenta a proposta central desta dissertação: uma arquitetura para o processamento de consultas a dados de biodiversidade armazenados em repositórios na Web. O objetivo é permitir que pesquisadores obtenham dados de ocorrência de espécies especificando consultas que envolvam predicados ecológicos (e.g. relações de predatismo aplicáveis às espécies desejadas), taxonômicos (e.g. especificação da ordem ou família das espécies) e geográficos (e.g restrição sobre a distribuição geográfica).

A arquitetura proposta se baseia em diversos conceitos e tecnologias apresentados nos capítulos anteriores. Os padrões definidos no contexto da Web semântica (Seção 2.2) são usados para a representação (ontologias), codificação (OWL/RDF) e consulta (SPARQL) dos dados. Para interoperabilidade dos dados de biodiversidade, a especificação emprega os padrões Darwin Core (Seção 2.1), GML e WFS (Seção 2.3.1).

3.1 Requisitos da arquitetura

A arquitetura proposta deve atender a diversos requisitos relacionados ao domínio das pesquisas em biodiversidade. Estes requisitos são descritos a seguir:

Usabilidade: A arquitetura deve ser adequada tanto ao cientista que a utilizará para obter os dados para sua pesquisa quanto para as instituições que desejam compartilhar seus dados. Cientistas precisam de diferentes ferramentas para tratar aspectos distintos de suas pesquisas. A arquitetura deve, portanto, possibilitar a inclusão de novas interfaces de interação para atender às necessidades do usuário. Sob a óptica das instituições que desejam compartilhar seus dados, a arquitetura deve simplificar o processo de publicação dos dados e ser tolerante em relação à diversidade tecnológica, inclusive na implementação dos bancos de dados.

Integração e flexibilidade na especificação dos dados: Os dados associados às pesquisas em biodiversidade são obtidos por instituições e grupos distintos, processados e armazenados de diferentes maneiras com inúmeras granularidades espaciais, temporais e amostrais. A arquitetura deve especificar mecanismos que possibilitem a integração destes dados heterogêneos. Deve ainda ser capaz de lidar com as freqüentes mudanças na classificação dos conceitos.

Inclusão de predicados geográficos e ecológicos: Dados geográficos são indispensáveis em pesquisas em biodiversidade. A arquitetura deve, portanto, ser capaz de integrar dados geográficos distintos e ser capaz de analisar predicados espaciais sobre os mesmos. Além disto, deve permitir a especificação de predicados ecológicos, uma necessidade neste domínio, raramente providos por sistemas de biodiversidade.

Para atender alguns dos requisitos descritos anteriormente, diversas tecnologias foram adotadas na especificação da arquitetura. Estas tecnologias são descritas a seguir:

Serviços Web: A utilização de serviços Web promove autonomia de escolha da tecnologia de implementação utilizada em cada parte da arquitetura. A arquitetura proposta utiliza padrões de serviços Web para (i) implementação do serviço de processamento de consultas, (ii) comunicação entre o serviço de consultas e as interfaces de consulta e (iii) comunicação entre o serviço de processamento de consultas e os repositórios onde os dados de biodiversidade estão armazenados.

Ontologias: Ontologias representam uma forma eficiente de capturar e contextualizar conceitos de um domínio em uma representação formal. Além disto, ontologias facilitam a construção e evolução destas representações de forma colaborativa. Ontologias são utilizadas na arquitetura para (i) codificar conceitos do domínio (como conceitos taxonômicos ou ecológicos) e (ii) descrever o conteúdo dos repositórios de dados. A arquitetura adota a linguagem OWL para a codificação dos conceitos do domínio e a linguagem SPARQL para a formulação de consultas sobre os conceitos especificados pelas ontologias.

Padrões geográficos: Os dados de biodiversidade tratados na arquitetura são os dados que contêm componentes geográficas. A arquitetura utiliza os padrões do consórcio OGC para representar e compartilhar os dados geográficos. Os repositórios disponibilizam os dados através de serviços WFS que, por padrão, retornam os dados em GML. Os dados retornados pelos repositórios são processados e compostos no serviço de processamento de consultas e também são retornados em GML para as interfaces de consulta.

Darwin Core: Os dados de ocorrência de espécies, dados primários nas pesquisas em biodiversidade, são representados na arquitetura usando o padrão Darwin Core 2. Os repositórios que disponibilizam dados de ocorrência de espécies devem, portanto, adequar o esquema de disponibilização dos dados (no caso, o esquema da *feature* do serviço WFS) ao padrão Darwin Core 2.

3.2 Especificação da arquitetura

Esta seção apresenta a arquitetura proposta. Ela emprega (i) ontologias de domínio como um modelo global para os dados compartilhados e (ii) padrões de interoperabilidade para acesso aos repositórios remotos. Seguindo esta abordagem, as consultas submetidas se referem a conceitos presentes nas ontologias de domínio. Os dados nos repositórios são tratados como instâncias dos conceitos das ontologias. É papel do serviço de consultas identificar os repositórios que possam conter instâncias dos conceitos especificados nas consultas.

3.2.1 Visão geral da arquitetura

A arquitetura é composta por três elementos: (i) interfaces de consulta, por meio das quais usuários elaboram consultas para a recuperação de dados de biodiversidade, (ii) um serviço de processamento de consultas, que processa as consultas recebidas das interfaces e (iii) repositórios distribuídos, de onde o serviço obtém os dados. A Figura 3.1 apresenta estes elementos e suas interações. As interfaces de consulta (centro inferior da figura) são aplicações construídas com objetivos específicos (e.g. predição de ocorrência de espécies, estabelecimento de prioridades para conservação) e distintos perfis de usuários (e.g. cientistas, estudantes). As consultas dos usuários são traduzidas pelas interfaces em SPARQL e encaminhadas para o serviço de processamento de consultas.

O serviço de processamento de consultas (centro da figura) é o principal elemento da arquitetura. Seu papel é encontrar os dados nos repositórios Web apropriados, processar estes dados e retornar os resultados ao usuário. Os repositórios (cantos inferiores da figura) são publicados por grupos e instituições de pesquisa. Existem dois tipos de repositórios: os que armazenam dados de ocorrência de espécies e os que armazenam os demais objetos geográficos usados na pesquisa em biodiversidade, como lagos, países e biomas. Todos os repositórios devem ser compatíveis com a especificação WFS, o que padroniza as interfaces e provê meios para a aplicação de filtros geográficos na obtenção dos dados. Os dados de ocorrência de espécies devem ser compatíveis com o Darwin Core. O esquema GML do resultado das consultas foi extendido para acomodar este padrão.

A figura mostra exemplos de dados publicados pelas instituições: dados de ocorrência

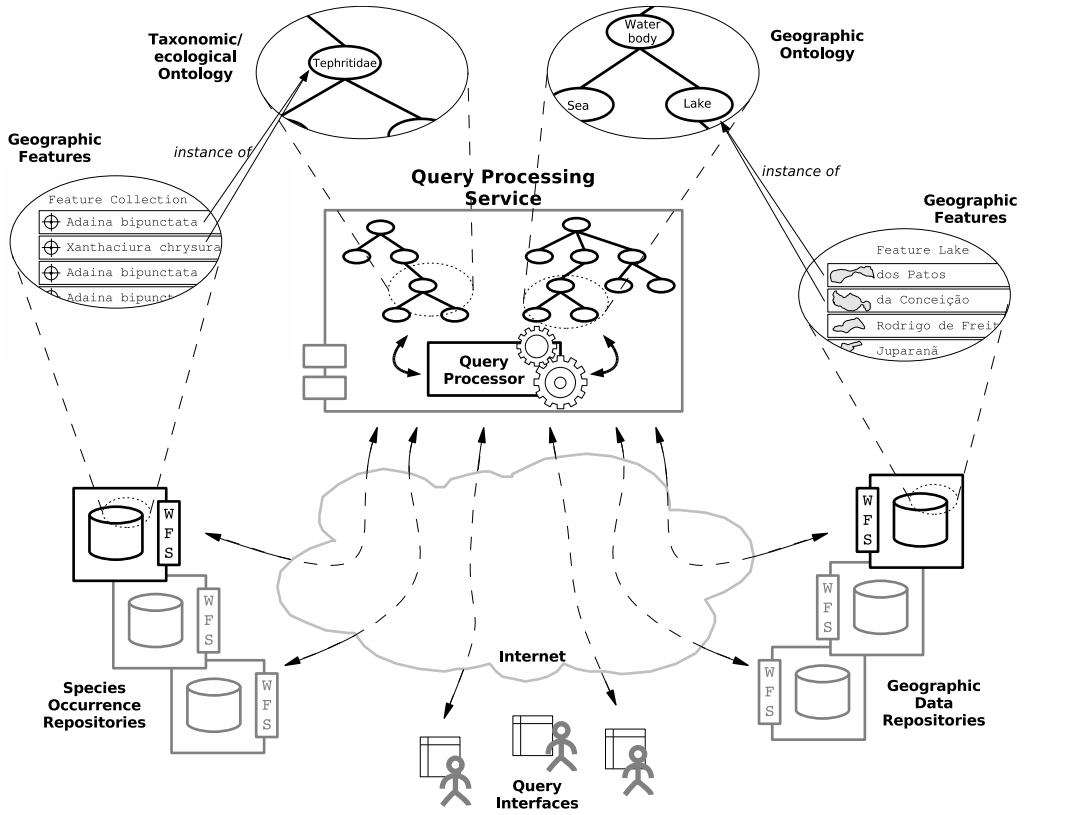

Figura 3.1: Visão geral das interações entre os elementos da arquitetura

(à esquerda), são instâncias que especificam, para cada observação registrada, os campos do padrão Darwin Core, como localização geográfica ou a metodologia usada. Os dados publicados nos repositórios geográficos precisam obrigatoriamente ter ao menos os campos de descrição e geometria espacial (associada a coordenadas geográficas).

A figura também destaca as ontologias utilizadas para o processamento dos predicados e expansão dos termos da consulta. A ontologia taxonômica e ecológica (taxo-ecológica) está à esquerda. Sua visão expandida mostra o conceito *Tephritidae* (a família de insetos que inclui as moscas da fruta). A ontologia à direita é a ontologia geográfica, com os conceitos *Corpo D'água* e *Lago* na visão expandida. Como ilustrado pelas setas entre as visões detalhadas, os registros contidos nos repositórios são considerados instâncias dos conceitos das ontologias.

3.2.2 Arquitetura do serviço de processamento de consultas

A Figura 3.2 mostra os elementos internos básicos do serviço de processamento de consultas e suas interações com outras partes da arquitetura. O processador de consultas recebe

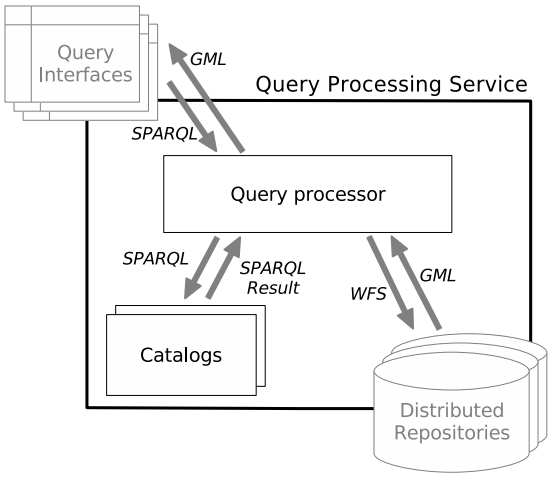

Figura 3.2: Interações entre os elementos internos e externos da arquitetura do serviço

consultas SPARQL das interfaces de consulta. O desenvolvimento de interfaces está fora do escopo deste trabalho. Porém, como destacado no Capítulo 2, muitas aplicações em biodiversidade exibem e processam dados geográficos. Por conta disto, os dados retornados pelo serviço às interfaces são encapsulados em um arquivo GML (como mostra a Figura 3.2).

O processamento das consultas requer a análise de informações internas, que são armazenadas em catálogos. O termo "catálogo" é empregado na arquitetura em analogia aos mecanismos padrão para processamento de consultas em SGBDs, onde catálogos armazenam informações a respeito do esquema e da alocação dos dados [30]. Na arquitetura proposta, os catálogos armazenam as ontologias utilizadas pelo serviço e registram os repositórios disponíveis. Eles são consultados pelo processador em tarefas como expansão de termos da consulta e busca por repositórios de dados. Os dois catálogos contidos no serviço de processamento são o catálogo de ontologias e o catálogo de repositórios.

O *catálogo de repositórios* atua como um índice para as fontes de dados de biodiversidade na Web. Ele contém registros de instituições e grupos de pesquisa reconhecidos como publicadores de dados para biodiversidade. Como lista a Figura 3.3, cada registro é composto por quatro campos principais: a classificação do repositório (type), a URI da fonte de dados, o retângulo geográfico envolvente (*bounding box*) e um conjunto de anotações referentes às ontologias do repositório de ontologias. O campo *type* indica se o repositório contém informações sobre ocorrência de espécies ou fenômenos geográficos. O retângulo envolvente define a região geográfica sobre a qual o repositório contém dados. As anotações de ontologias qualificam o conteúdo do repositório.

O *catálogo de ontologias de domínio* armazena as ontologias de conceitos taxo-ecológicos e geográficos. Seu conteúdo é provido e gerenciado pelas comunidades de pesquisa. Ele

Type	URI	Bbox	HasDataAbout
occurrence	http://plants.org/wfs	-46,-18 -43,-16	Chromolaena_squalida, Mikania_purpurascens
occurrence	http://flies.org/wfs	-47,-12 -42,-15	Tephritidae
occurrence	http://flowers.org/wfs	-43,-16 -27,-18	Asteraceae
geographic	http://ibge.gov.br/wfs	-74,4 -26,-35	State
geographic	http://ibama.gov.br/wfs	-74,4 -33,-35	LandBiome

Figura 3.3: Exemplos de registros do catálogo de repositórios

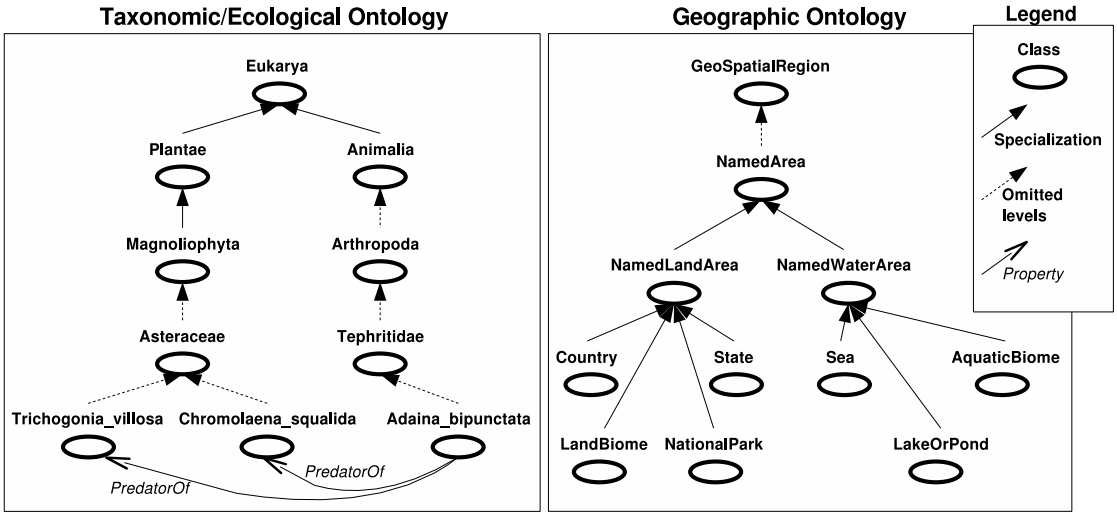

Figura 3.4: Partes das ontologias taxonômica/ecológica e geográfica utilizadas (inspiradas nas ontologias do projeto SPIRE)

é utilizado, por exemplo, para expandir as consultas contendo predicados ecológicos. A Figura 3.4 mostra fragmentos das ontologias utilizadas, que foram inspiradas pelas ontologias desenvolvidas pelo projeto SPIRE [71]. A figura mostra, por exemplo que *Adaina bipunctata* (uma espécie de mosca) é uma subclasse de *Tephritidae* que preda espécies de *Chromolaena squalida* (uma planta da família das margaridas). A ontologia geográfica também é inspirada pelo projeto SPIRE e organiza hierarquicamente diversos fenômenos geográficos, como *Corpo D'água* e *Lago*.

3.2.3 Processamento de consultas

A Figura 3.5 mostra a seqüência de fases no processamento das consultas. O processador recebe uma consulta SPARQL, transforma esta consulta em um grafo (fase A) que é analisado e resolvido (fase B). O resultado é então integrado (fase C) e retornado em um arquivo GML para a interface de consulta.

A arquitetura proposta é fortemente baseada em estruturas de ontologias, que devem

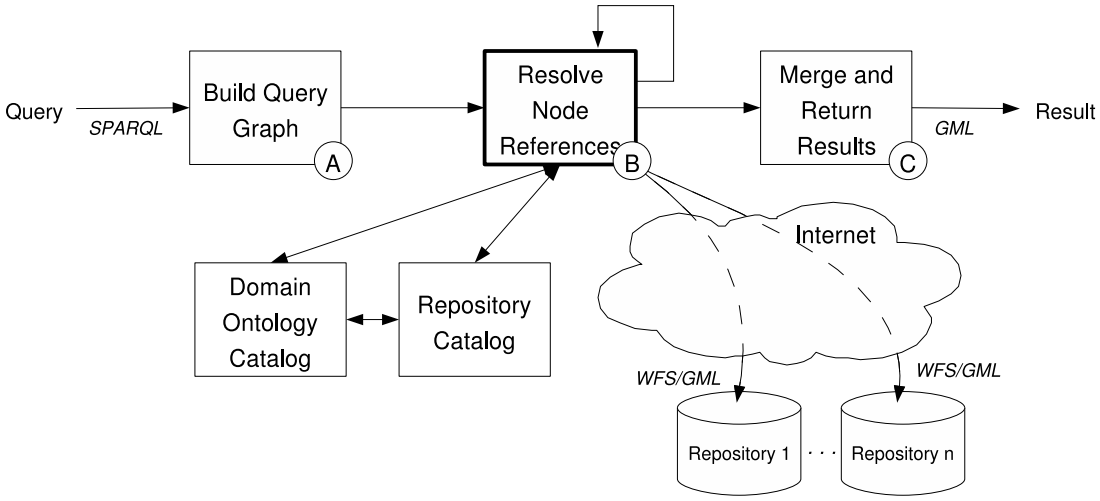

Figura 3.5: Fases do processamento de consultas

ser analisadas em diversos passos do processamento das consultas. Por conta disto, a solução para execução das consultas emprega estruturas para processamento de ontologias – i.e., grande parte dos resultados intermediários são utilizados para criar, casar e expandir grafos. A idéia básica é a seguinte: A consulta do usuário, em SPARQL, é transformada em um grafo que é processado de forma a combinar inferências usando ontologias e busca de registros de coletas em repositórios Web. Estas buscas são auxiliadas por informação contida nos catálogos do sistema, que possibilitam resolver predicados taxo-ecológicos (via máquinas de inferência) e geográficos (combinando informação de ontologias), buscando registros de coleta e demais dados geográficos nos repositórios designados pelo catálogo de repositórios. As três fases principais são:

A Construção do Grafo da Consulta: Constrói um grafo correspondente à consulta SPARQL de entrada, após análise sintática da expressão. O grafo gerado nesta fase é uma materialização direta do grafo expresso implicitamente na consulta: num grafo de consulta $G(V, E)$ para uma consulta Q, (i) $u \in V \Leftrightarrow u$ é sujeito ou predicado de Q e (ii) $(u, v) \in E \Leftrightarrow$ existe em Q um predicado aplicado entre o sujeito correspondente a u e o objeto correspondente a v. Na construção do grafo os vértices e as arestas são rotulados com os respectivos rótulos expressos na consulta (e.g. URI).

B Resolução das Referências dos Nós: Processa iterativamente o grafo da consulta, resolvendo elementos indefinidos. Primeiramente, os catálogos internos são analisados; em seguida, requisições WFS são enviadas aos repositórios Web apropriados para recuperação dos registros necessários. O resultado é o grafo da consulta reduzido às variáveis selecionadas na consulta original, contendo os dados obtidos nos

repositórios.

C Composição e Retorno dos Resultados: Processa o grafo retornado na fase anterior, extraindo os dados requisitados pelo usuário e traduzindo-os em GML para compor o arquivo resultado.

Algorithm 1 Processa ramos-folha

Require: grafo G
Ensure: todos os nós do grafo são resolvidos
1: **while** G contém ramos-folha a resolver **do**
2: $r \Leftarrow$ ramo não resolvido de maior prioridade
3: **if** *prioridade*$(r) = 1$ **then** {r pode ser resolvido localmente}
4: atualiza grafo de consulta
5: **else if** *prioridade*$(r) = 2$ **then** {resolução de r requer dados do catálogo}
6: resolve usando dados do catálogo
7: aplica resultados ao grafo de consulta, atualizando as prioridades de acordo com o resultado
8: **else** {resolução de r requer dados dos repositórios}
9: simplifica predicados espaciais
10: determina repositórios para consultar
11: monta e envia consultas WFS
12: aplica resultados ao grafo de consulta, atualizando as prioridades de acordo com o resultado
13: **end if**
14: **end while**

A Fase B é a mais complexa e é subdividida em diversos passos, de acordo com o Algoritmo 1. O algoritmo é aplicado iterativamente aos *ramos-folha* do grafo até que o grafo esteja completamente resolvido. O qualificador ramo-folha é empregado aqui para descrever conjuntos compostos por um vértice de grau 1 (folha), sua aresta incidente e seu vértice vizinho (tratado aqui como *base* do ramo). Formalmente, um ramo-folha r em um grafo de consulta $G(V, E)$ é definido como

$$r = \{(u, v, (u, v)) : u, v \in V \wedge (u, v) \in E \wedge grau(u) = 1\} \tag{3.1}$$

O Algoritmo 1 é adequado a grafos de consulta conexos e acíclicos (o que os classifica como árvores [93]). Uma extensão para tratamento de grafos desconexos e cíclicos é objeto de trabalhos futuros. O algoritmo não trata também casos de dados redundantes, conflitantes ou problemas de comunicação. A especificação de mecanismos robustos em relação a estes casos é parte de trabalhos futuros.

Prioridade	Tipo	Modelo de Ramo	Exemplo	Estratégia de Resolução	Resultado
1	Elemento vazio		?x geo:within []	substitui o ramo por elemento vazio	
1	Variável com predicado geográfico (relação topológica)		?x geo:within geo:Bahia	adiciona relação topológica ao conjunto de restrições	
2	Variável com predicado taxo-ecológico		?x te:predadorDe te:gato	resolve usando catálogo e substitui o ramo pelo conjunto resultante	
2	Conjunto com predicado taxo-ecológico		[te:cão,te:gato] sub te:felino	resolve usando catálogo e usa resultado para filtrar o conjunto	
2	Variável com predicado geográfico (taxonomia)		?x sub geo:Corpo D'agua	resolve usando catálogo e substitui o ramo pelo conjunto resultante	
2	Conjunto com predicado geográfico (taxonomia)		[geo:Bahia,geo:Itú] sub geo:Estado	resolve usando catálogo e usa resultado para filtrar o conjunto	
3	Instância geográfica		geo:Bahia a geo:Estado	obtém o registro em um repositório usando o id da instância e o adiciona ao grafo	
4	Variável de instância (geográfica ou ecológica)		x? a te:canino	otimiza restrições geográficas e obtém os registros nos repositórios, criando o conjunto resultante	

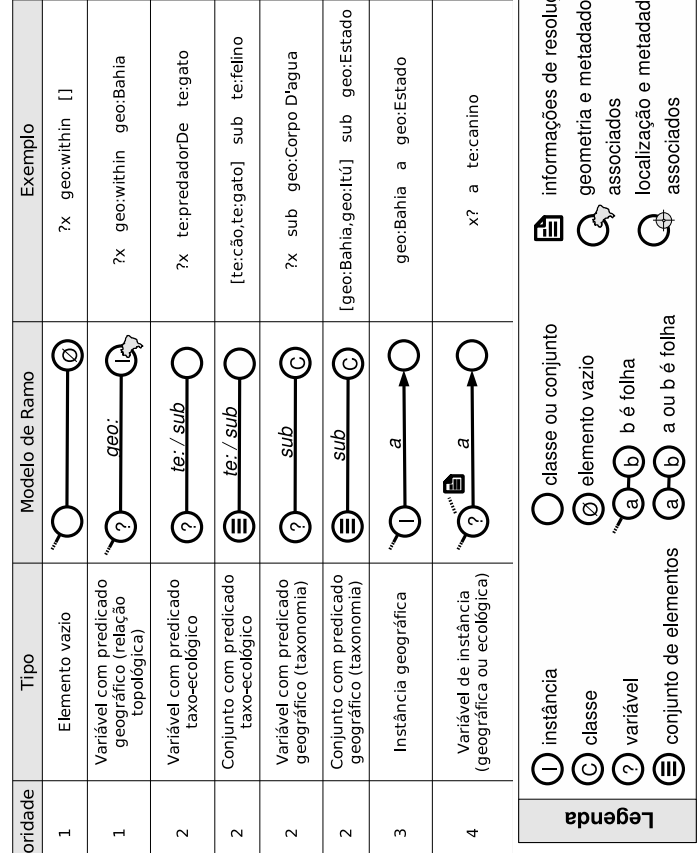

Tabela 3.1: Resolução de ramos-folha de acordo com o tipo

A resolução de um ramo-folha consiste em analisar o predicado correspondente à sua aresta, processar o predicado de acordo com o objeto correspondente à folha, e aplicar os resultados à base do ramo. Ao final do processamento de um ramo, sua folha é eliminada e sua base contém o resultado do processamento. O algoritmo utiliza as informações mostradas na Tabela 3.1 para a resolução dos ramos-folha. A tabela mostra a prioridade de resolução de acordo com diferentes tipos de ramos (1 é a prioridade mais alta). A prioridade está associada à forma como os ramos serão resolvidos. Os ramos de prioridade 1 são resolvidos localmente, os ramos de prioridade 2 envolvem dados do catálogo e os ramos de menor prioridade (3 e 4) envolvem dados dos repositórios remotos. O objetivo da divisão dos tipos de ramos em prioridades é postergar a resolução dos ramos de resolução mais difícil. Desta forma, quando um ramo de resolução difícil é resolvido, o processador pode ter à disposição mais restrições, resolvidas em ramos anteriores, para diminuir número de registros obtidos. A Tabela 3.1 mostra ainda modelos gráficos que descrevem cada tipo, exemplos de triplas correspondentes, um breve resumo da estratégia de resolução e um modelo gráfico do resultado final. Por exemplo, a terceira linha da tabela diz respeito a um ramo descrito como variável com predicado taxo-ecológico. Este tipo de ramo tem prioridade de resolução 2. A tabela mostra um exemplo de tripla que geraria um ramo como este, no caso contendo uma variável (?x), uma predicado ecológico (te:predadorDe) e um sujeito fictício (te:gato). A estratégia de resolução deste tipo de ramo, resumida na tabela, consiste em buscar no catálogo os conceitos que satisfazem ao predicado e usá-los para compor o conjunto resultado. Graficamente, o resultado do processamento é representado na tabela por um conjunto de elementos (no caso, conceitos) com o restante do grafo omitido.

Toda consulta contém variáveis de retorno (especificadas na cláusula SELECT). Durante a montagem do grafo de consulta (Fase A) as variáveis de retorno são marcadas. Sempre que um ramo contendo uma variável de retorno é resolvido, o nó resultante é marcado com o identificador da variável de retorno. O algoritmo é repetido até que o grafo contenha apenas nós marcados como variável de retorno. Ramos contendo apenas variáveis de retorno precisam ser preservados para a montagem da resposta à consulta, sendo portanto tratados como tendo prioridade $= \infty$ e nunca resolvidos. Os passos do algoritmo são explicados a seguir.

Obtém ramo de maior prioridade (linha 2): Dos ramos-folha presentes no grafo de entrada, escolhe um dos de maior prioridade, de acordo com a Tabela 3.1. A variável r recebe o ramo escolhido e é resolvida nos passos seguintes da iteração corrente do algoritmo.

Atualiza grafo de consulta (linha 4): Para os ramos de prioridade igual a 1, a resolução consiste em simples manipulações do grafo. Estes ramos são resolvidos

antes por não demandarem obtenção de informações nos catálogos ou repositórios.

Resolve usando dados do catálogo (linha 6): Para os ramos de prioridade igual a 2, a resolução consiste em obter as informações necessárias diretamente do catálogo de ontologias.

Simplifica predicados espaciais (linha 9): Para os ramos de prioridade maior do que 2, é preciso obter os dados necessários nos repositórios. Se um ramo de entrada neste passo possuir restrições geográficas, elas devem ser otimizadas com o objetivo de diminuir o volume de dados retornados. Predicados redundantes podem ser excluídos [76] e interseções entre regiões podem ser efetivadas. Um estudo aprofundado sobre quais otimizações podem ser feitas neste passo foi deixado como trabalho futuro. Os passos subseqüentes de algoritmo consideram que os predicados geográficos foram pré-processados para restringir a extenção espacial das consultas submetidas aos repositórios.

Determina repositórios para consultar (linha 10): Requisita ao catálogo de repositórios uma lista de repositórios que podem prover instâncias de interesse à resolução do ramo corrente. Esta lista é obtida a partir do casamento do conteúdo do ramo com o tipo e a anotação ontológica no catálogo.

Monta e envia consultas WFS (linha 11): Monta consultas WFS adaptadas para os repositórios identificados no passo "Obtém repositórios". Submete assincronamente as consultas especificadas aos repositórios.

Aplica resultados ao grafo de consulta (linhas 7 e 12): Traduz para a representação do grafo os resultados das consultas ao catálogo de ontologias ou aos repositórios.

3.3 Exemplo de processamento de consulta

Esta seção detalha o processamento de uma consulta na arquitetura. Como se verá no Capítulo 4, apenas parte dos módulos foi implementada.

Seja a seguinte consulta: "retorne todos os registros de ocorrência de espécies que são predadas pela mosca *Adaina Bipunctata* e que foram encontradas *no interior da Mata Atlântica Paulista*". Esta consulta contém predicados ecológicos (predadas), taxonômicos (*Adaina Bipunctata*) e espaciais (no interior da). A Figura 3.6 apresenta o código correspondente em SPARQL. No código, os prefixos *te* e *geo* designam, respectivamente, a ontologia taxo-ecológica e a ontologia geográfica, usadas para processar a consulta. O prefixo *sr* representa predicados topológicos (e.g. within, overlaps, intersects – definidos em

```
PREFIX te: <http:// . . . /webios/taxo_eco.owl#>
PREFIX geo: <http:// . . . /webios/geographic.owl#>
PREFIX sr: <http:// . . . /webios/spatial_relation.owl#>
SELECT ?occurrence
WHERE { te:Adaina_Bipunctata te:predatorOf ?species .
        ?occurrence a ?species .
        ?occurrence sr:within geo:Sao_Paulo .
        geo:Sao_Paulo a geo:State .
        ?occurrence sr:within geo:Atlantic_Rainforest .
        geo:Atlantic_Rainforest a geo:Biome
}
```

Figura 3.6: Consulta de exemplo em SPARQL equivalente a "retorne todos os registros de ocorrência de espécies e que são predadas pela mosca *Adaina Bipunctata* que foram encontradas *dentro da Mata Atlântica Paulista*"

[68] e baseadas nas relações topológicas formalizadas em [28]). Os predicados taxonômicos e ecológicos são processados com base nas ontologias. As relações espaciais são tratadas pelo processador de consultas na construção de filtros para a obtenção dos dados nos repositórios. As fases do processamento da consulta de exemplo são as seguintes (usando as fases da Figura 3.5 e o Algoritmo 1, descritos na Seção 3.2.3):

Fase A – Construção do Grafo de Consulta: A Figura 3.7(1) mostra o grafo para a consulta de exemplo. Predicados são arestas, enquanto sujeitos e objetos são nós do grafo. A variável de retorno selecionada (*?occurrence*) aparece destacada.

Fase B – Resolução das Referências dos Nós: Os próximos ítens descrevem iterações sucessivas do Algoritmo 1. A Figura 3.7 mostra a representação do grafo durante o processamento. Cada iteração corresponde a um elemento da figura: por exemplo, a Figura 3.7(4) mostra o estado inicial da iteração 4.

Iteração 1: A Figura 3.7(1) mostra o grafo de consulta em seu estado original. A figura destaca os três ramos-folha presentes no grafo, todos candidatos à resolução nesta iteração. O ramo à esquerda na figura, que descreve um predicado ecológico aplicado a uma variável, possui prioridade 2 (segundo a Tabela 3.1). Este ramo é escolhido para resolução, uma vez que os demais candidatos possuem prioridade mais baixa (3, segundo a tabela). A resolução do ramo escolhido consiste em uma consulta ao catálogo de ontologias. De acordo com a ontologia taxo-ecológica da Figura 3.4, as espécies que satisfazem o predicado da consulta são *Chromolaena squalida* e *Trichogonia villosa*. O algoritmo constrói um conjunto que agrupa as

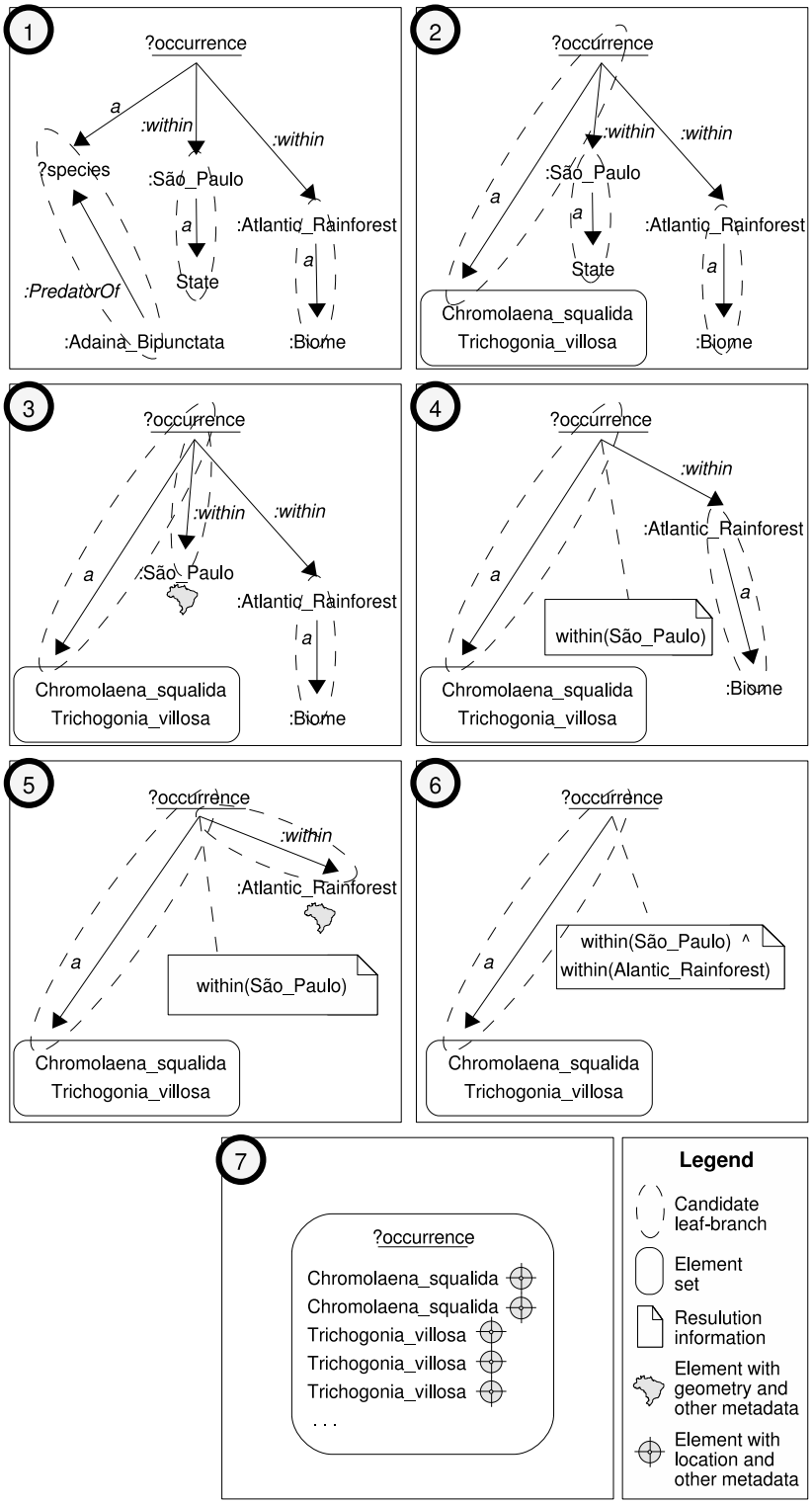

Figura 3.7: Seqüência de configurações do grafo da consulta de exemplo (Figura 3.6) em iterações consecutivas do algoritmo de processamento. As setas se referem à semântica dos predicados e não implicam em orientação no grafo. Os prefixos das URIs foram omitidos.

espécies resultantes para substituir no grafo o ramo resolvido (como mostra a Figura 3.7(2)). Esta estratégia corresponde à descrição da terceira linha da Tabela 3.1.

Iteração 2: Dos três ramos destacados, o do centro e o da direita, ambos representando instâncias geográficas, possuem mais alta prioridade (3) – o ramo à esquerda possui prioridade igual a 4 e é desconsiderado. O algoritmo escolhe um dos ramos de prioridade 3 para resolução nesta fase, no caso, o ramo do centro. A resolução deste ramo requer uma busca nos repositórios para a obtenção da geometria do estado do São Paulo. Para tal, é preciso consultar no catálogo os repositórios que podem conter esta informação. Obtidos os repositórios, o processador monta uma consulta WFS para obter o registro correspondente ao estado de São Paulo. A iteração 6, na seqüência do texto, apresenta uma descrição mais detalhada do processo de obtenção de dados em repositórios, incluindo exemplos de códigos intermediários.

Iteração 3: Nesta iteração o ramo do centro equivale a uma variável com predicado geográfico. Este ramo tem prioridade 1 e é o escolhido para resolução. A resolução deste tipo de ramo é trivial, se resumindo à associação da restrição geográfica ao nó da variável.

Iteração 4: Nesta iteração há apenas dois ramos. O ramo da direita é o de maior prioridade (3) e sua resolução é semelhante à da iteração 2.

Iteração 5: O ramo à direita na Figura 3.7(5) é resolvido nesta iteração, de forma similar à apresentada na iteração 3.

Iteração 6: Nesta iteração só resta um ramo a ser resolvido, equivalente a uma variável de instância ecológica. Este ramo possui a prioridade mais baixa, 4, conforme a Tabela 3.1. Sua resolução consiste em obter nos repositórios remotos os dados de ocorrência de espécies com base no conjunto de espécies identificadas e nas restrições geográficas. A seguir, detalhes dos passos do Algoritmo 1 são descritos para esta iteração.

Simplifica predicados espaciais: Uma possível otimização dos predicados geográficos presentes (within(São_Paulo) $\wedge$ within(Atlantic_Rainforest)) consiste em calcular a interseção das geometrias de São Paulo e Atlantic Rainforest formando um novo predicado within para a região resultante. Este procedimento potencialmente reduz a extensão espacial para a qual a consulta deve ser processada.

Obtém repositórios: O processador de consultas deve verificar no catálogo de repositórios a localização dos repositórios de ocorrência que contêm instâncias das espécies em questão (*Chromolaena squalida* e *Trichogonia villosa*). Dados os registros de repositórios descritos na Figura 3.3, os repositórios são *plants.org* e

flowers.org. Para aumentar a seletividade, os repositórios são também filtrados pela extensão dos seus retângulos envolventes de acordo com o contexto da consulta. Neste caso, apenas repositórios com retângulos envolventes sobrepondo ou dentro da geometria correspondente à Mata Atlântica Paulista são considerados. Este exemplo de consulta supõe que os repositórios *plants.org* e *flowers.org* satisfazem o filtro geográfico.

Monta e envia consultas WFS: A Figura 3.8 apresenta a consulta WFS montada para a obtenção de dados de ocorrência das espécies requisitadas (nomes científicos *Chromolaena squalida* e *Trichogonia villosa* na figura) dentro de uma dada área (que é definida pelo elemento *polygon* que contém as coordenadas que definem a região espacial da Mata Atlântica Paulista). As consultas montadas são submetidas para os repositórios correspondentes (*plants.org* e *flowers.org*). No exemplo, envia-se a consulta mostrada na Figura 3.8 para o repositório plants.org.

Aplica resultados ao grafo de consulta: Os dados recuperados no passo anterior são integrados ao grafo da consulta.

Iteração 7: Nesta iteração todas as variáveis estão resolvidas e o algoritmo retorna para a execução da Fase C. A Figura 3.7(7) mostra a configuração final do grafo de consulta. A variável de retorno *?ocurrence* foi completamente resolvida, se transformando em um conjunto contendo instâncias de ocorrência de espécies, associadas à informação de localização e demais metadados (de acordo com o padrão Darwin Core) que foram obtidos nos repositórios.

Fase C – Composição e Retorno dos Resultados: Integra, para as variáveis selecionadas na consulta (no caso, *?occurrence*), seus conteúdos (no caso, um conjunto de ocorrências das espécies *Chromolaena squalida* e *Trichogonia villosa*), codificando-os em GML e retornando o resultado. A Figura 3.9 mostra um documento GML que responde à consulta de exemplo. Ele indica, por exemplo, que a instância de *Trichogonia villosa* encontrada foi coletada por *Adriana M. Almeida* e *Umberto Kubota* no ponto (-44.7196,-23.3099).

Resumo

Este capítulo apresentou a arquitetura de consultas a dados de biodiversidade proposta nesta dissertação. Os principais requisitos para uma arquitetura desta natureza, bem como as tecnologias adotadas para atendê-los, foram listados. O capítulo descreveu os elementos contidos na arquitetura e a estratégia de processamento das consultas – incluindo um exemplo de processamento. As propostas foram validadas através da cons-

```
<wfs:GetFeature . . . >
  <wfs:Query typeName="plantsorg:species">
    <Filter>
      <And>
        <Or>
          <PropertyIsEqualTo>
            <PropertyName>ScientificN</PropertyName>
            <Literal>Chromolaena_squalida</Literal>
          </PropertyIsEqualTo>
          <PropertyIsEqualTo>
            <PropertyName>ScientificN</PropertyName>
            <Literal>Trichogonia_villosa</Literal>
          </PropertyIsEqualTo>  . . .
        </Or>
        <Within>
          <PropertyName>the_geom</PropertyName>
          <gml:Polygon> . . .
                <gml:coordinates . . . >
                  -46.469289,-18.895586
                  -44.87035,-18.66422 . . .
                </gml:coordinates> . . .
          </gml:Polygon>
        </Within>
      </And>
    </Filter>
  </wfs:Query>
</wfs:GetFeature>
```

Figura 3.8: Parte de uma consulta WFS para a obtenção de certas espécies numa determinada área

```
<wfs:FeatureCollection . . . > . . .
  <gml:featureMember>
    <lis:webios fid=webios.4">
      <lis:the_geom> . . .
            <gml:Point>
              <gml:coordinates . . . >
                -44.7196,-23.3099
              </gml:coordinates>
            </gml:Point> . . .
      </lis:the_geom>
      <lis:ScientificName>Trichogonia_villosa
      </lis:ScientificName>
      <lis:Collector>Adriana M. Almeida, Umberto Kubota
      </lis:Collector>
    </lis:webios>
  </gml:featureMember>
  <gml:featureMember>
    <lis:webios fid=webios.6">
      <lis:the_geom>  . . .
            <gml:Point>
              <gml:coordinates . . . >
                -44.8341,-23.2024
              </gml:coordinates>
            </gml:Point> . . .
      </lis:the_geom>
      <lis:ScientificName>Chromolaena_squalida
      </lis:ScientificName>
      <lis:Collector>Erica P. Anseloni, J.C. Silva
      </lis:Collector> . . .
    </lis:webios> . . .
  </gml:featureMember>
</wfs:FeatureCollection>
```

Figura 3.9: Resultado GML para a consulta WFS contendo os dados de ocorrência de espécies

trução de protótipos para alguns elementos da arquitetura. O próximo capítulo descreve estas implementações.

Capítulo 4

Aspectos de implementação

Este capítulo apresenta os protótipos implementados e as tecnologias utilizadas para a validação da proposta da arquitetura de processamento de consultas a dados de biodiversidade. A arquitetura proposta no Capítulo 3 é composta por três elementos básicos: (i) repositórios de dados distribuídos, (ii) interfaces de consultas e (iii) o serviço de processamento de consultas. Os protótipos foram implementados com o objetivo de testar e validar aspectos específicos de cada elemento. A implementação definitiva exigiria integrar e estender tais protótipos, o que foi deixado para trabalhos futuros.

4.1 Repositórios de dados

A arquitetura especifica que os repositórios de dados devem ser compatíveis com o padrão WFS. O servidor utilizado para a publicação dos dados foi o GeoServer [37], projeto de código aberto que implementa os padrões WFS e WMS do OGC.

O GeoServer é capaz de publicar dados nos padrões do OGC a partir de diversas fontes (como arquivos ShapeFile, GML ou bancos de dados). O SGBD geográfico escolhido para armazenar os dados foi o PostGIS/PostgreSQL. O PostGIS [75] é uma extensão do banco de dados objeto-relacional PostgreSQL para gerenciamento de dados espaciais.

Os dados utilizados na implementação deste protótipo foram obtidos da base de dados do Laboratório de Interações Inseto-Planta do Instituto de Biologia da UNICAMP e da base de dados do projeto FishBase [34]. A Figura 4.1 mostra seleções dos dados utilizados no protótipo. Cada tupla corresponde a um registro de ocorrência de espécie. A Figura 4.1a contém registros de peixes obtidos na base de dados do projeto FishBase. A Figura 4.1b contém registros de plantas hospedeiras obtidos na base de dados do Instituto de Biologia. Por exemplo, a primeira linha da Figura 4.1b apresenta um registro da espécie de planta *Chromolaena odorata* coletada por U. Kubota et al. Além destes atributos, os registros também contêm atributos descrevendo a data, local e demais condições da

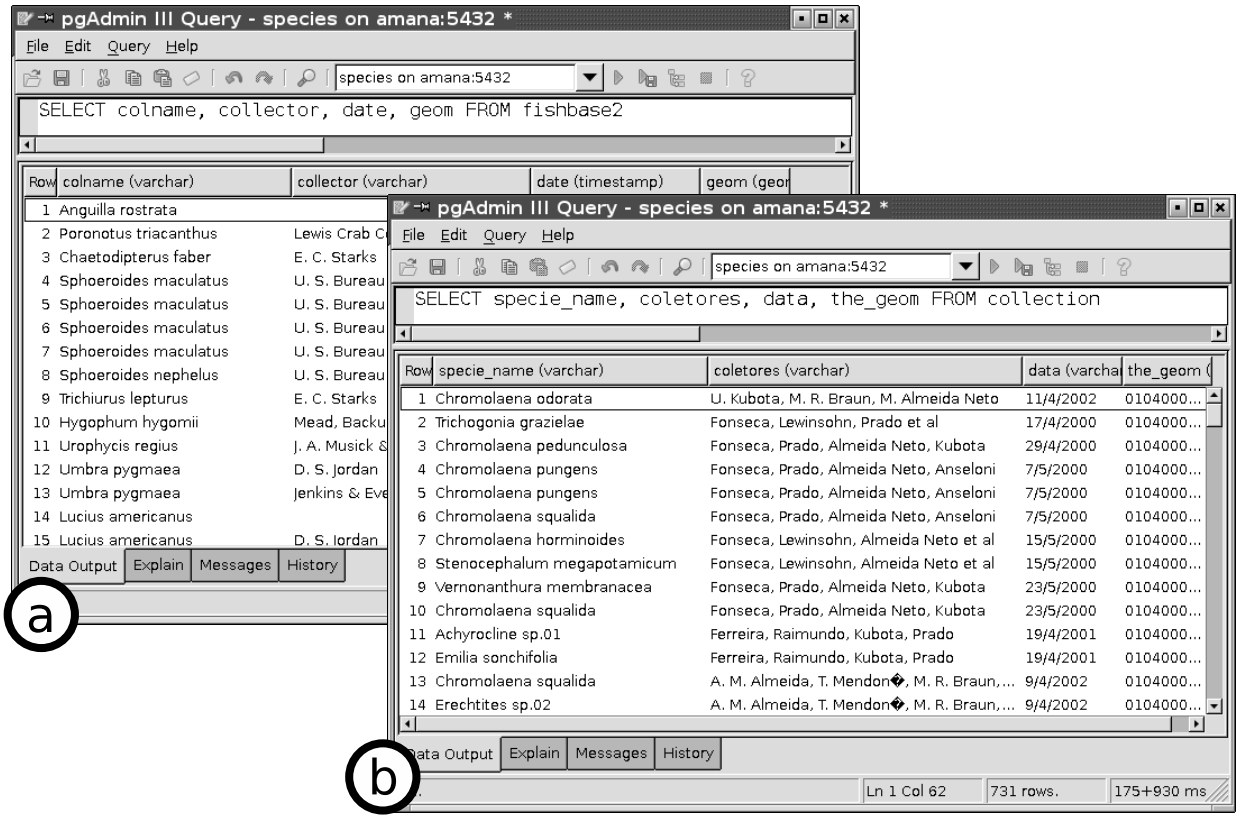

Figura 4.1: Seleção dos dados utilizados no protótipo

coleta.

4.2 Interfaces de consulta

Duas interfaces foram implementadas para testar diferentes aspectos da proposta. A primeira interface é uma aplicação Web para consulta e exibição de dados de ocorrência de espécies usando os padrões WFS e WMS. A segunda é uma aplicação desktop para consultas SPARQL e exibição dos resultados.

A primeira interface visou testes de integração dos padrões WFS e WMS. A implementação utilizou JSP (Java Server Pages). A Figura 4.2a mostra os parâmetros que podem ser selecionados na consulta (bounding box geográfico e espécie-alvo). A Figura 4.2b mostra o mapa resultante da consulta, obtido através de uma consulta WMS. Clicando-se nas regiões marcadas no mapa, o usuário obtém a lista de espécies identificadas na área, no formato GML (Figura 4.2c).

A interface de consultas SPARQL foi desenvolvida para auxiliar os testes do protótipo do serviço de consultas. A implementação foi feita na linguagem Java e utilizou a tecno-

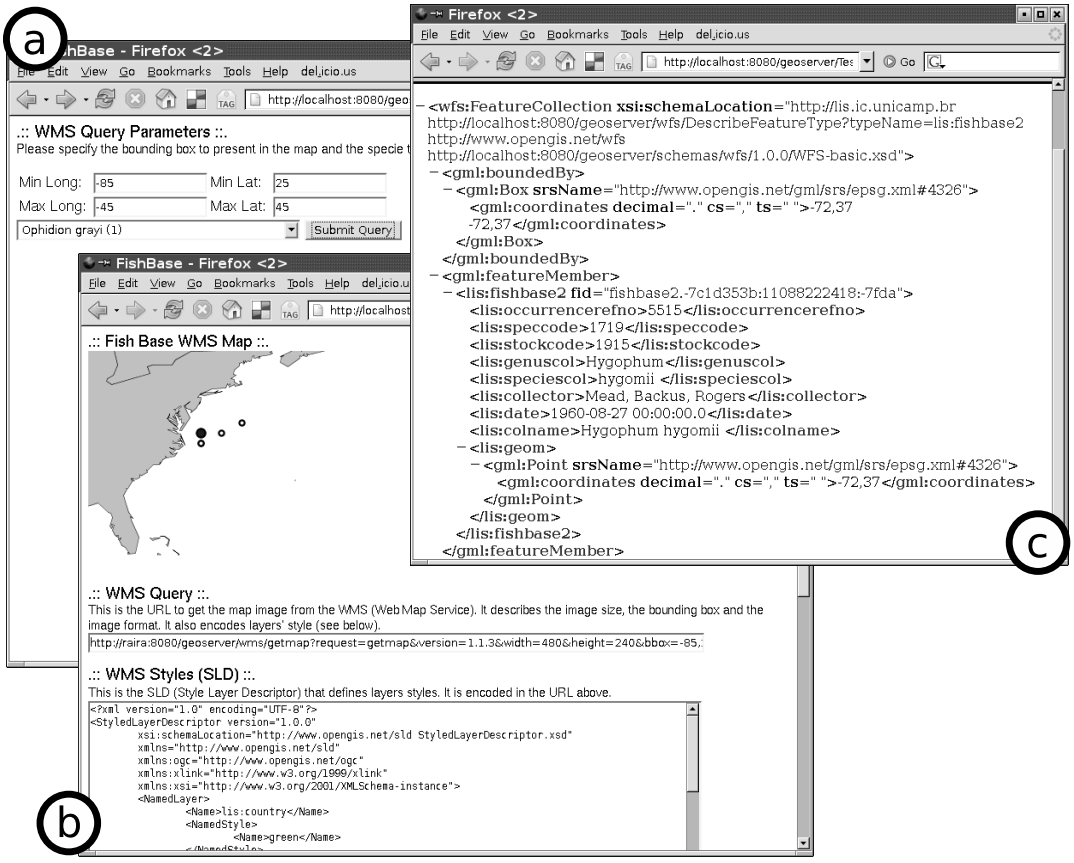

Figura 4.2: Interface de consultas de dados de ocorrência

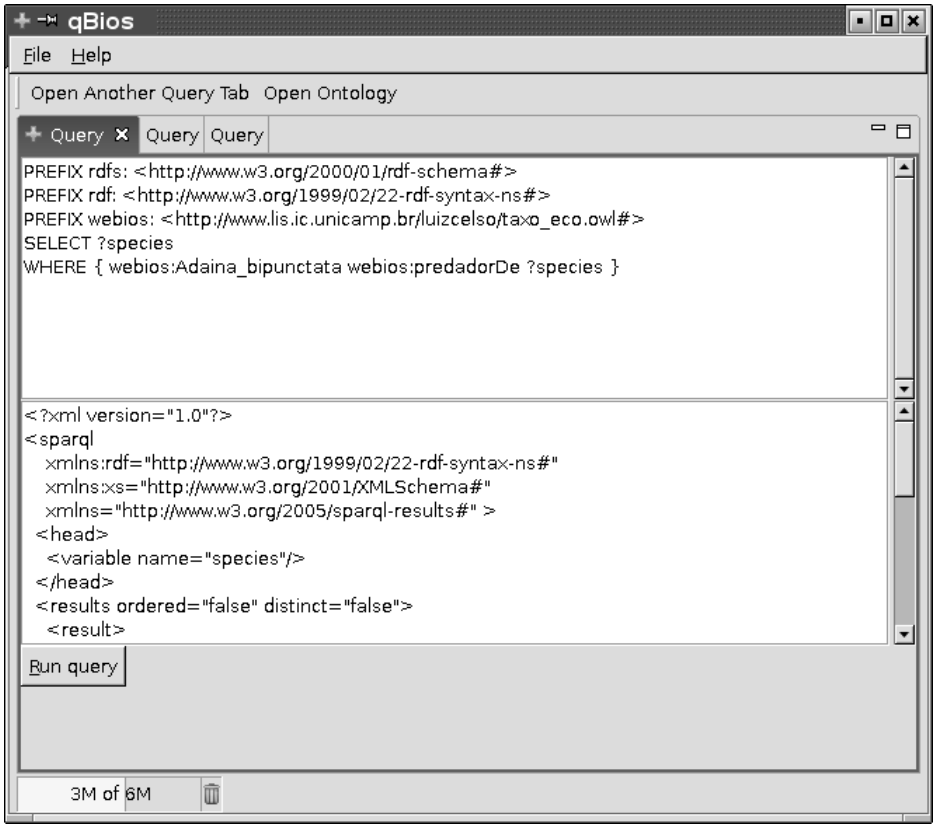

Figura 4.3: Interface de consultas SPARQL

logia RCP (Rich Client Platform) [54] do projeto Eclipse. A Figura 4.3 mostra a interface desenvolvida. Na parte superior as consultas são incluídas (múltiplas abas para consultas podem ser criadas). O resultado da execução das consultas é apresentado em XML na parte inferior da figura.

4.3 Serviço de processamento de consultas

A implementação do protótipo do serviço de processamento de consultas considerou um subconjunto da especificação: as consultas SPARQL contêm apenas predicados taxonômicos e ecológicos. Com esta restrição, foi possível desenvolver os testes sem a necessidade de se implementar ou alterar um processador SPARQL. O framework Jena foi utilizado nos passos intermediários do processamento das consultas. A restrição também simplifica o processamento das consultas, como mostra a Figura 4.4.

Supõe-se que toda consulta tem como resultado um conjunto de espécies. Como não há predicados geográficos nem instâncias indefinidas na consulta, as fases do processamento se

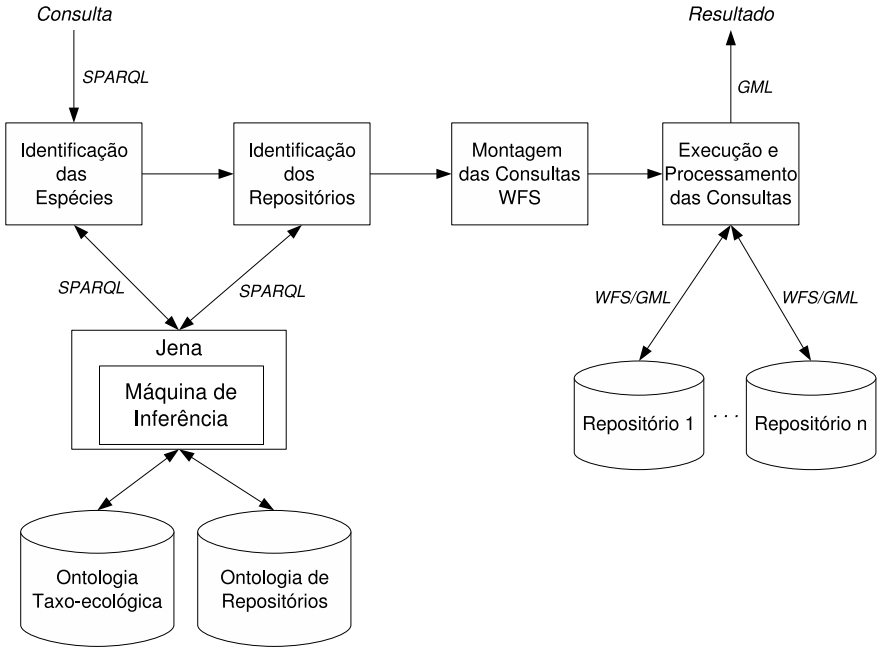

Figura 4.4: Fases do processamento de consultas no protótipo

resumem a (i) Identificação das Espécies, (ii) Identificação dos Repositórios, (ii) Montagem das Consultas e (iv) Execução e Processamento das Consultas. Estas fases são descritas a seguir:

- Identificação das Espécies: A consulta SPARQL é enviada ao processador de consultas do Jena, que foi configurado com uma máquina de inferência para processamento de propriedades transitivas para possibilitar a resolução dos predicados taxonômicos.
- Identificação dos Repositórios: A partir das espécies retornadas na fase anterior, obtém os repositórios que podem conter os dados necessários.
- Montagem das Consultas WFS: Monta as consultas WFS que serão enviadas para os repositórios.
- Execução e Processamento das Consultas: Submete as consultas WFS e compõe os resultados no arquivo GML de resposta.

Para simplificar o processamento na elaboração do protótipo, o catálogo de repositórios foi implementado como uma ontologia de repositórios. A Figura 4.6 (à direita) mostra a ontologia de repositórios que foi implementada em OWL (omitindo os dados dos repositórios, como url e feature type). Na figura, três repositórios são representados como instâncias da classe *Repositório*. Os repositórios são associados a classes da ontologia taxo-ecológica, indicando os conceitos para os quais são capazes de prover dados.

```
PREFIX webios: <http://www.lis.ic.unicamp.br/taxo_eco.owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
SELECT ?especie
WHERE { ?predador webios:predadorDe ?especie .
        ?predador rdfs:subClassOf webios:Tephritidae
}
```

Figura 4.5: Consulta de exemplo em SPARQL equivalente a "obter todos os registros de espécies predadas por tephritídeos"

4.4 Exemplo de processamento de uma consulta

Esta seção apresenta a execução de uma consulta, considerando as fases descritas na Seção 4.3 e mostrando consultas e resultados intermediários. A consulta mostrada na Figura 4.5 é utilizada como exemplo. A consulta equivale a "obter todos os registros de espécies predadas por tephritídeos" e envolve um predicado taxonômico (família *Tephritidae*) e um predicado ecológico (predador). A Figura 4.6 à esquerda apresenta a ontologia taxo-ecológica utilizada. Os relacionamentos ecológicos são os mesmos da Figura 3.4 e foram omitidos. Atualmente, a ligação destas fases não é automatizada e os resultados parciais devem ser transferidos manualmente entre os protótipos desenvolvidos.

- Identificação das Espécies: Considerando as simplificação impostas para a elaboração do protótipo, não há necessidade de pré-processamento da consulta de entrada. A consulta é, portanto, encaminhada diretamente para processamento pelo Jena. O resultado da execução no Jena é um conjunto de espécies que atendem à especificação da consulta, como mostra a Figura 4.7.

- Identificação dos Repositórios: A partir das espécies retornadas na fase anterior, uma nova consulta SPARQL é construída para obtenção dos repositórios que podem conter as espécies em questão. A Figura 4.8 mostra a consulta SPARQL construída para obtenção dos repositórios. Esta consulta é processada pelo processador do Jena. O resultado (Figura 4.9) é um conjunto de repositórios que podem conter dados sobre as espécies de interesse.

- Montagem das Consultas WFS: Utiliza as espécies retornadas na fase anterior para construir as as consultas WFS que serão enviadas para os repositórios. A Figura 4.10 mostra a consulta WFS enviada para o repositório flowers.org (oitava linha).

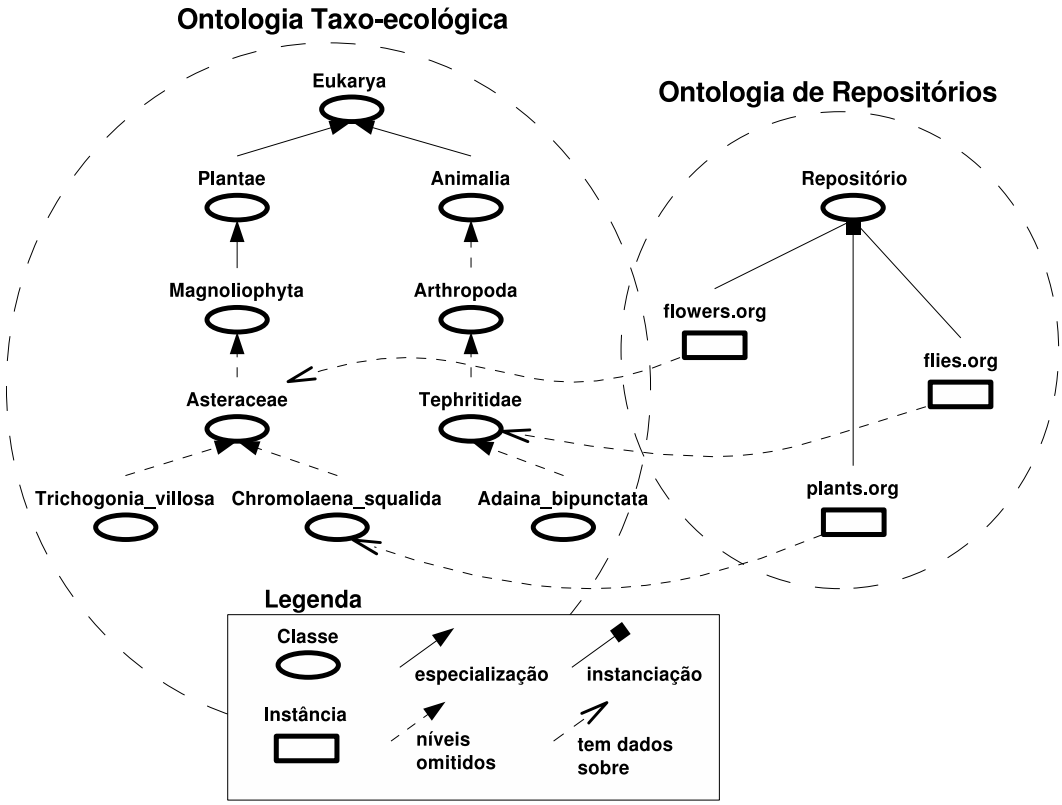

Figura 4.6: Ontologias utilizadas no protótipo

```
<?xml version="1.0"?>
<sparql
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:xs="http://www.w3.org/2001/XMLSchema#"
    xmlns:webios="http://www.lis.ic.unicamp.br/taxo_eco.owl#"
    xmlns="http://www.w3.org/2005/sparql-results#" >
  <head>
    <variable name="especie"/>
  </head>
  <results ordered="false" distinct="false">
    <result>
      <binding name="especie">
        <uri>webios:Trichogonia_villosa</uri>
      </binding>
    </result>
    <result>
      <binding name="especie">
        <uri>webios:Chromolaena_squalida</uri>
      </binding>
    </result>
  </results>
</sparql>
```

Figura 4.7: Resultado da consulta de exemplo retornado pelo catálogo (há um abuso da notação XML para namespaces para reduzir o tamanho das URIs)

A consulta obtém todos os registro da feature fo.occurrence cujos nomes científicos sejam *Trichogonia villosa* ou *Chromolaena squalida*.

- Execução e Processamento das Consultas: As consultas geradas na fase anterior são enviadas para os respectivos repositórios. Os resultados das consultas são então concatenados para gerar a resposta final ao usuário. A Figura 4.11 mostra, em GML, a resposta gerada a partir das consultas.

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX webios: <http://www.lis.ic.unicamp.br/taxo_eco.owl#>
SELECT DISTINCT ?repositorio ?url ?feature
WHERE
{
   UNION
   { ?repositorio a webios:Repositorio .
      ?repositorio webios:temDadosSobre ?super .
      webios:Trichogonia_villosa rdfs:subClassOf ?super .
      ?repositorio webios:hasURL ?url .
      ?repositorio webios:subjectFeature ?feature
   }
   UNION
   { ?repositorio a webios:Repositorio .
      ?repositorio webios:temDadosSobre ?super .
      webios:Chromolaena_squalida rdfs:subClassOf ?super .
      ?repositorio webios:hasURL ?url .
      ?repositorio webios:subjectFeature ?feature
   }
}
```

Figura 4.8: Consulta para obtenção dos repositórios

```
<?xml version="1.0"?>
<sparql
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:xs="http://www.w3.org/2001/XMLSchema#"
    xmlns:webios="http://www.lis.ic.unicamp.br/taxo_eco.owl#"
    xmlns="http://www.w3.org/2005/sparql-results#" >
  <head>
    <variable name="repositorio"/>
    <variable name="url"/>
    <variable name="feature"/>
  </head>
  <results ordered="false" distinct="true">
    <result>
      <binding name="repositorio">
        <uri>webios:plants.org</uri>
      </binding>
      <binding name="url">
        <literal datatype="http://www.w3.org/2001/XMLSchema#string">
             http://plants.org/wfs</literal>
      </binding>
      <binding name="feature">
        <literal datatype="http://www.w3.org/2001/XMLSchema#string">
             po:occurrence</literal>
      </binding>
    </result>
    <result>
      <binding name="repositorio">
        <uri>webios:flowers.org</uri>
      </binding>
      <binding name="url">
        <literal datatype="http://www.w3.org/2001/XMLSchema#string">
             http://flowers.org/wfs</literal>
      </binding>
      <binding name="feature">
        <literal datatype="http://www.w3.org/2001/XMLSchema#string">
             fo:occurrence</literal>
      </binding>
    </result>
  </results>
</sparql>
```

Figura 4.9: Resultado da consulta para obtenção dos repositórios

```
<wfs:GetFeature service="WFS" version="1.0.0"
  outputFormat="GML2"
  xmlns:topp="http://www.openplans.org/topp"
  xmlns:wfs="http://www.opengis.net/wfs"
  xmlns:ogc="http://www.opengis.net/ogc"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wfs . . . ">
  <wfs:Query typeName="fo:occurrence">
    <ogc:Filter>
      <ogc:Or>
          <PropertyIsEqualTo>
                <PropertyName>ScientificN</PropertyName>
                <Literal>Trichogonia_villosa</Literal>
          </PropertyIsEqualTo>
          <PropertyIsEqualTo>
                <PropertyName>ScientificN</PropertyName>
                <Literal>Chromolaena_squalida</Literal>
          </PropertyIsEqualTo>
       </ogc:Or>
    </ogc:Filter>
    </wfs:Query>
</wfs:GetFeature>
```

Figura 4.10: Consulta WFS para obtenção dos dados de ocorrência de espécies

```
<gml:featureMember>
  <lis:IBUNICAMP fid="webios.1">
    <lis:the_geom>
      <gml:MultiPoint srsName=". . . epsg.xml#4326">
        <gml:pointMember>
          <gml:Point>
            <gml:coordinates . . . >
              -43.93889968,-19.15027749
            </gml:coordinates>
          </gml:Point>
        </gml:pointMember>
      </gml:MultiPoint>
    </lis:the_geom>
    <lis:ScientificN>Chromolaena_squalida</lis:ScientificN>
  </lis:IBUNICAMP>
</gml:featureMember>
<gml:featureMember>
  <lis:IBUNICAMP fid="webios.2">
    <lis:the_geom>
      <gml:MultiPoint srsName=" . . . epsg.xml#4326">
        <gml:pointMember>
          <gml:Point>
            <gml:coordinates decimal="." cs="," ts=" ">
              -44.09833337,-20.03103653
            </gml:coordinates>
          </gml:Point>
        </gml:pointMember>
      </gml:MultiPoint>
    </lis:the_geom>
    <lis:ScientificN>Trichogonia_villosa</lis:ScientificN>
  </lis:IBUNICAMP>
</gml:featureMember>
. . .
```

Figura 4.11: Resultado em GML retornado para a interface de consulta (vários elementos do padrão Darwin Core foram omitidos)

Capítulo 5

Conclusões e extensões

5.1 Contribuições

O trabalho apresentado nesta dissertação se concentrou nos desafios relacionados ao compartilhamento, integração e a obtenção dos dados utilizados nas pesquisas em biodiversidade. As dificuldades encontradas neste contexto estão relacionadas à grande quantidade de conceitos envolvidos e ao aspecto distribuído da especificação dos mesmos. Este cenário demanda soluções flexíveis, que se adaptam às constantes alterações nas classificações dos conceitos e, ao mesmo tempo, preservam a autonomia das instituições de pesquisa no gerenciamento dos seus dados.

Para tratar os problemas apresentados, esta dissertação propõe uma arquitetura para consultas a dados de biodiversidade baseada em padrões da Web semântica e do consórcio OGC. A arquitetura, descrita no Capítulo 3, é composta de três categorias de elementos básicos: um serviço de processamento de consultas, interfaces de consulta e repositórios de dados. Os conceitos taxonômicos, ecológicos e geográficos são representados por ontologias armazenadas no serviço de processamento de consultas. Como destacado por Cullot et al. [17], as ontologias apresentam um certo aspecto colaborativo, onde os conceitos podem evoluir continuamente através de interações com a comunidade de usuários. As ontologias são codificadas em OWL e o serviço de processamento recebe consultas especificadas em SPARQL. Ressalta-se ainda que a opção por serviços distribuídos reflete as práticas e necessidades reais do domínio, como ressaltado na introdução deste texto. Sistemas de biodiversidade dependem fortemente de informações disponíveis em museus ou grandes coleções de coletas (como o Smithsonian), contendo dezenas de milhões de registros, o que torna impossível a manutenção dos registros em um local centralizado.

Na arquitetura, os repositórios que armazenam os dados de ocorrência de espécies e demais dados geográficos são disponibilizados com base no padrão WFS. Isto homogeneíza as interfaces de acesso aos dados e possibilita que o processador de consultas especifique

filtros geográficos na obtenção dos dados. A obrigatoriedade da utilização do padrão WFS não impõe restrições às estratégias das instituições com respeito ao armazenamento dos dados. As implementações do padrão WFS disponíveis tipicamente possibilitam a utilização de diversas tecnologias de armazenamento, incluindo bancos de dados geográficos e tradicionais, arquivos Shapefile ou GML. Além disto, a utilização do padrão não implica necessariamente em aumento de custos ou complexidade de implantação. Além de diversas implementações comerciais dos padrões do OGC (que são listadas em [70]), várias alternativas de código aberto estão à disposição [74]. Exemplos são os projetos GeoServer [37], Deegree [23] e MapServer [88]. O processo de instalação do GeoServer, por exemplo, não toma mais que alguns minutos, mesmo para pessoas sem experiência prévia com a aplicação. Este custo para adequação à arquitetura existe em todas as propostas de compartilhamento descentralizado de dados de biodiversidade, o que mostra que a opção tomada está de acordo com a tendência dos usuários e sistemas da área.

A associação entre o conteúdo dos repositórios e as ontologias de domínio fica a cargo do catálogo de repositórios do serviço de processamento de consultas. O acoplamento entre o nível conceitual (ontologias) e o nível das instâncias (registros dos repositórios) é um acoplamento fraco, possibilitando que operações de inclusão e exclusão de repositórios, bem como alterações nas classificações das ontologias tenham baixo impacto sobre o funcionamento do sistema.

Seguindo as recomendações e tendências do W3C para a Web semântica, OWL foi escolhida como linguagem de especificação das ontologias e SPARQL foi escolhida como linguagem de consulta. As duas linguagens se mostraram adequadas à representação e manipulação dos conceitos em alto nível. Porém, para a obtenção de uma granularidade maior nas consultas, como em "obter as espécies encontradas em lagos de profundidade menor que 1m", as linguagens se tornam pouco práticas e demandariam algumas extensões. Este é um problema freqüentemente associado a linguagens de representação de ontologias que, em geral, não são capazes de representar objetos complexos [17].

As principais contribuições desta dissertação são, portanto:

- A proposta de uma arquitetura que permite consultas baseadas em predicados taxonômicos, ecológicos e geográficos a dados de biodiversidade em repositórios na Web;
- A análise e aplicação de conceitos da Web semântica e padrões de interoperabilidade no contexto das pesquisas em biodiversidade;
- A implementação parcial da arquitetura proposta com base em dados reais fornecidos pelo Instituto de Biologia da UNICAMP.

5.2 Extensões

Possíveis extensões para a arquitetura proposta podem ser divididas em dois grupos: as relativas ao detalhamento do funcionamento de elementos da arquitetura e as relativas a novas funcionalidades.

Os seguintes trabalhos futuros podem ser destacados no que tange a arquitetura:

- Algoritmo de processamento de consultas: O processamento das consultas na arquitetura envolve conceitos relacionados a consultas em grafos [5], consultas geográficas [79] e consultas distribuídas [50]. Diversos trabalhos neste campo foram analisados no decorrer da pesquisa mas, a princípio, não foi possível aplicar tais técnicas de processamento diretamente. Estudos futuros podem permitir a representação das consultas com base em estruturas de árvore, como no processamento de consultas em álgebra relacional. Esta é a estratégia sugerida em [18] e permite que técnicas de otimização bem estabelecidas no campo dos bancos de dados relacionais sejam aplicadas no processamento das consultas SPARQL.

- Otimização de consultas geográficas: Existem diversas propostas nesta área. Algumas não se aplicam à arquitetura porque pressupõem um maior controle dos mecanismos de acesso e de armazenamento físico dos dados. É o caso dos índices de objetos geográficos e do processamento de consultas em duas fases [79], por exemplo. Outras propostas [76, 27], se focam em um nível mais alto de abstração, tratando a semântica das relações expressas nas consultas. Estes últimos provavelmente são as estratégias mais adequadas à implementação na arquitetura. É preciso, porém, conduzir um estudo aprofundado da aplicabilidade destas técnicas no contexto distribuído da arquitetura.

- Interação entre os pesquisadores e o catálogo de ontologias de domínio: Os pesquisadores devem ser capazes de criar e editar as ontologias de domínio usadas na arquitetura. A interação dos pesquisadores pode ser através de ambientes de edição colaborativa de ontologias (como em [8]) ou utilizar recursos do serviço de ontologias em desenvolvimento no projeto WeBios.

- Tratamento de redundância e conflito de fontes de dados em repositórios: Nos casos em que mais de um repositório pode fornecer registros de uma dada instância, o processador de consultas deve ser capaz de decidir qual a melhor fonte para o dado. Este tipo de problema pode ser tratado através da inclusão de diretivas de especificação de autoridades que, por exemplo, imponham que apenas um órgão oficial do governo possa fornecer dados sobre estados brasileiros. Outra possibilidade é se basear em classificações baseadas em *feedback* dos usuários para se decidir qual o melhor repositório em situações de conflito.

- Tratamento de exceções: Diversos fatores podem comprometer a execução de uma consulta, como problemas de comunicação ou inatividade de repositórios. A arquitetura deve prever tais fatores e especificar mecanismos para a notificação dos erros. Para lidar com problemas de atraso na comunicação, as técnicas de *query scrambling* podem ser adaptadas ao processamento das consultas.

- Estudo de eficiência do método proposto: O processamento de predicados taxonômicos e ecológicos envolve análise de transitividade de relacionamentos. Esta análise apresenta alto custo computacional, e pode se tornar impraticável em grandes volumes de dados. Faz-se necessário, portanto, o desenvolvimento de testes de desempenho, análise de alternativas eficientes para processamento de transitividade ou, em útima instância, limitar o uso destes predicados nas consultas.

Com relação à inclusão de novas funcionalidades na arquitetura, algumas possibilidades são:

- Pré-processamento de sinônimos: A utilização de técnicas de expansão de termos (e.g. [41]), ou eliminação de ambigüidade (e.g. [48]) nas consultas permite uma interação mais flexível com os usuários. Neste caso, seria possível, por exemplo, elaborar consultas baseadas em nomes populares de espécies e encontrar regiões geográficas que tiveram seus nomes alterados com o tempo.

- Distribuição do processamento das consultas: Em muitos casos os projetos de compartilhamento de dados de biodiversidade se organizam em hierarquias que refletem a classificação taxonômica dos seres. É o caso dos projetos MaNIS [52] e FishBase [34], focados em níveis mais altos de classificação, e de muitos outros que se focam em subgrupos taxonômicos. Talvez seja possível explorar esta hierarquia na distribuição do processamento das consultas. Desta forma, se um serviço recebe uma consulta que contém partes associadas a mamíferos, estas partes seriam encaminhadas para os serviços do projeto MaNIS, que por sua vez encaminharia quaisquer outras subdivisões da consulta a níveis inferiores.

- Cache de instâncias e relações: O serviço de consultas pode empregar técnicas de cache de resultados de consultas [50] para armazenar instâncias muito usadas (como Estado de São Paulo ou Rio Tietê) e relações topológicas (como Rio Tietê *corta* São Paulo). Estas estratégias tendem a agilizar as consultas mas podem também introduzir problemas como o retorno de dados desatualizados.

- Procedência dos dados: No contexto das pesquisas científicas, é importante que os pesquisadores tenham controle e conhecimento sobre a procedência dos dados. Este

tipo de recurso é essencial para a replicação dos experimentos e comprovação dos resultados. Para atender tal demanda, o processador de consultas deve incorporar mecanismos que registrem a origem dos dados e permitam a reexecução das consultas.

- Aplicação em outros domínios: A arquitetura proposta foi baseada nas necessidades e singularidades associadas ao contexto das pesquisas em biodiversidade. Porém, os mecanismos empregados são em grande parte genéricos e poderiam ser aplicados em outros contextos. Esta hipótese necessita de estudos de caso para sua validação.

- Consultas de alta granularidade: A arquitetura permite a elaboração de consultas baseadas nos conceitos das ontologias de domínio e o processador de consultas faz o mapeamento entre os conceitos e os registros dos respositórios (instâncias). Porém, não foi especificado um mapeamento para os atributos dos registros. Tal mepeamento permitiria consultas de maior granularidade, como obter registro de ocorrências de espécies coletadas por um dado pesquisador. Este mapeamento poderia ser feito usando os descritores dos atributos como predicados nas consultas SPARQL (e.g. ?especie :coletor "nome"), já que os registros de ocorrência utilizam o padrão Darwin Core para a definição dos atributos. Um problema desta abordagem está relacionado com os registros de dados geográficos, onde é difícil impor um padronização de atributos (um lago pode ter um atributo *profundidade* enquanto um país tem um atributo *população*).

- Mecanismos de catalogação de repositórios: O catálogo de repositórios proposto foi especificado com os requisitos mínimos necessários para o processamento das consultas. Esta especificalção pode se extendida para se adequar a padrões como o UDDI [3] ou para flexibilizar os esquemas dos repositórios registrados, como em [10].

- Integração da arquitetura ao sistema WeBios: Diversos módulos do sistema WeBios estão em fase de especificação ou construção. Por conta disto a arquitetura proposta, mais especificamente o serviço de processamento de consultas, não foi totalmente integrado ao WeBios. Após a integração, o serviço de processamento de consultas terá como cliente exclusivo o mediador de consultas de WeBios, que filtrará as consultas de alto nível do usuário e encaminhará as porções das consultas envolvendo precidados taxo-ecológicos e geográficos para o serviço proposto. Além disto, o armazenamento e processamento das ontologias de domínio será baseado no serviço de ontologias, o que permitirá a obtenção e integração de ontologias provindas de diversos grupos de pesquisa.

Referências Bibliográficas

[1] N. Alameh. Chaining Geographic Information Web Services. *IEEE Internet Computing*, 7:22 – 29, 2003.

[2] A. M. Alasqur, S. Y. W. Su, and H. Lam. OQL: a Query Language for Manipulating Object-Oriented Databases. In *Proc. Int'l. Conf. on Very Large Data Bases*, page 433, Amsterdam, The Netherlands, August 1989.

[3] G. Alonso, F. Casati, H. Kuno, and V. Machiraju. *Web Services - Concepts, Architectures and Applications*. Springer Verlag, 2004.

[4] L. Amsaleg, M. J. Franklin, A. Tomasic, and T. Urhan. Scrambling Query Plans to Cope with Unexpected Delays. In *Fourth International Conference on Parallel and Distributed Information Systems (PDIS '96)*, pages 208–219, Los Alamitos, Ca., USA, December 1996. IEEE Computer Society Press.

[5] R. Angles and C. Gutierrez. Survey of Graph Database Models. Technical Report TR/DCC-2005-10, Computer Science Department, Universidad de Chile, 2005.

[6] G. Antoniou and F. van Harmelen. Web Ontology Language: OWL. In *Handbook on Ontologies*, pages 67–92. Springer, 2004.

[7] N. Athanasis, K. Kalabokidis, M. Vaitis, and N. Soulakellis. The Emerge of Semantic Geoportals. In *OTM Workshops*, volume 3762 of *Lecture Notes in Computer Science*, pages 1127–1136. Springer, 2005.

[8] J. Bao, Z. Hu, D. Caragea, J. Reecy, and V. Honavar. A Tool for Collaborative Construction of Large Biological Ontologies. In *DEXA Workshops*, pages 191–195. IEEE Computer Society, 2006.

[9] T. Berners-Lee, J. Hendler, and O. Lassila. The Semantic Web. *Scientific American*, 284(5):34–43, 2001.

[10] S. Bowers, K. Lin, and B. Ludäscher. On Integrating Scientific Resources through Semantic Registration. In *Proc SSDBM*, pages 349–352, 2004.

[11] B. Breckling and H. Reuter. Analysing biodiversity: the necessity of interdisciplinary trends in the development of ecological theory. *Poiesis & Praxis: International Journal of Technology Assessment and Ethics of Science*, 3(1 - 2):83–105, October 2004.

[12] D. Brickley and R. V. Guha. RDF Vocabulary Description Language 1.0: RDF Schema. World Wide Web Consortium, Recommendation REC-rdf-schema-20040210, February 2004.

[13] J. Brown and M. Lomolino. *Biogeography*. Sinauer Associates, 1998.

[14] E. Clementini, P. DiFelice, and P. van Oosterom. A Small Set of Formal Topological Relationships Suitable for End-user Interaction. In *Proc Third Intl. Symp. Spatial Databases - SSD*, pages 277–295, 1993.

[15] E. Clementini, J. Sharma, and M. J. Egenhofer. Modeling topological spatial relations: Strategies for query processing. *Computers and Graphics*, 18(6):815–822, 1994.

[16] E. F. Codd. A Relational Model of Data for Large Shared Data Banks. *Comm. ACM*, 13(6):377–387, June 1970.

[17] N. Cullot, C. Parent, S. Spaccapietra, and C. Vangenot. Ontologies: A contribution to the DL/DB debate. In *Proc First Intl. Work. Semantic Web and Databases - SWDB*, pages 109–129, 2003.

[18] R. Cyganiak. A Ralational Algebra for SPARQL. Technical Report HPL-2005-170, Hewlett-Packard Development Company, L.P., 2005.

[19] G. Câmara, M. A. Casanova, A. S. Hemerly, G. C. Magalhães, and C. M. Bauzer Medeiros. *Anatomia de Sistemas de Informação Geográfica*. 10a Escola de Computação, 1996.

[20] M. C. Daconta, L. J. Obrst, and K. T. Smith. *The Semantic Web : A Guide to the Future of XML, Web Services, and Knowledge Management*. Wiley, 2003.

[21] J. Daltio and C. B. Medeiros. Um Servidor de Ontologias para apoio a Sistemas de Biodiversidade. In *Proc. V Workshop de Teses e Dissertações em Banco de Dados (XXI SBBD)*, pages 71–76, Florianópolis, SC, Brasil, October 2006.

[22] R. Dawkins. *The Ancestor's Tale: A Pilgrimage to the Dawn of Evolution*. Houghton Mifflin, 2004.

[23] deegree Project. deegree - Building Spatial Data Infrastructures based on Free Software. http://www.deegree.org (accessed February 26, 2007).

[24] Distributed Generic Information Retrieval (DiGIR). DiGIR website. http://digir.net (accessed February 26, 2007).

[25] L. Ding, P. Kolari, Z. Ding, S. Avancha, T. Finin, and A. Joshi. Using Ontologies in the Semantic Web: A Survey. TechReport TR-CS-05-07, UMBC, July 2005.

[26] M. Edwards and D. R. Morse. The potential for computer-aided identification in biodiversity research. *Trends in Ecology & Evolution*, 10:153–158, 1995.

[27] M. J. Egenhofer. Deriving the composition of binary topological relations. *Journal of Visual Languages and Computing*, 5(1):133–149, 1994.

[28] M. J. Egenhofer and R. Franzosa. Point-Set Topological Spatial Relations. *International Journal of Geographical Information Systems*, 5(2):161–174, 1991.

[29] M. J. Egenhofer and J. Herring. Categorizing binary topological relationships between regions, lines, and points in geographic databases. Technical report, University of Maine, Department of Surveying Engineering, 1991.

[30] R. Elmasri and S. B. Navathe. *Fundamentals of Database Systems, 2nd Edition.* Benjamin/Cummings, 1994.

[31] R. Fileto. *The POESIA Approach for the Integration of Data and Services in the Semantic Web.* PhD thesis, IC-UNICAMP, 2003.

[32] The Biological Collection Access Service for Europe (BioCASE). BioCASE website. http://www.biocase.org (accessed February 26, 2007).

[33] R. B. Freitas and R. S. Torres. OntoSAIA: Um Ambiente Baseado em Ontologias para Recuperação e Anotação Semi-Automática de Imagens. In *1o Workshop em Bibliotecas Digitais - proc. XX Simpósio Brasileiro de Bancos de Dados*, 2005.

[34] R. Froese and D. Pauly. FishBase website. http://www.fishbase.org (accessed February 26, 2007).

[35] Global Biodiversity Information Facility (GBIF). GBIF website. http://www.gbif.org (accessed February 26, 2007).

[36] J. H. Gennari, M. A. Musen, R. W. Fergerson, W. E. Grosso, M. Crubezy, H. Eriksson, N. F. Noy, and S. W. Tu. The evolution of Protege: an environment for

knowledge-based systems development. *International Journal of Human-Computer Studies*, 58(1):89–123, 2003.

[37] GeoServer Project. GeoServer web site. http://geoserver.sourceforge.net (accessed February 26, 2007).

[38] H. Charles J. Godfray. Challenges for taxonomy. *Nature*, 417(6884):17–19, May 2002.

[39] R. Guralnick and D. Neufeld. Challenges Building Online GIS Services to Support Global Biodiversity Mapping and Analysis: Lessons from the Mountain and Plains Database and Informatics project. *Biodiversity Informatics*, 2:56–69, August 08 2005.

[40] V. Haarslev and R. Moller. Racer: A Core Inference Engine for the Semantic Web, October 13 2003.

[41] H. H. Hochmair. Ontology Matching for Spatial Data Retrieval from Internet Portals. In *First Intl. Conf. GeoSpatial Semantics - GeoS*, volume 3799 of *Lecture Notes in Computer Science*, pages 166–182. Springer, 2005.

[42] HP Labs. Jena website. http://jena.sourceforge.net/ (accessed February 26, 2007).

[43] International Organization for Standardization. *ISO/IEC 9075-1:1999: Information technology — Database languages — SQL — Part 1: Framework (SQL/Framework).* pub-ISO:adr, 1999.

[44] M. B. Jones, MM. P. Schildhauer, O. J. Reichman, and S. Bowers. The New Bioinformatics: Integrating Ecological Data from the Gene to the Biosphere. *Annual Review of Ecology, Evolution, and Systematics*, 37:519–544, December 2006.

[45] G. Z. Pastorello Jr. Publicação e Integraçãoo de Workflows Científicos na Web. Master's thesis, IC-UNICAMP, 2005.

[46] M. C. Molles Jr. *Ecology: Concepts and Applications.* Mcgraw-Hill College, 2001.

[47] B. Kerr, M. A. Riley, M. W. Feldman, and B. J. M. Bohannan. Local dispersal promotes biodiversity in a real-life game of rock-paper-scissors. *Nature*, 418(6894):171–174, July 2002.

[48] E. Klien, M. Lutz, and W. Kuhn. Ontology-based discovery of geographic information services- : An application in disaster management. *Computers, environment and urban systems*, 30:102–123, 2006.

[49] G. Klyne and J. J. Carroll. Resource Description Framework (RDF): Concepts and Abstract Syntax. http://www.w3.org/TR/2004/REC-rdf-concepts-20040210 (accessed February 26, 2007), February 2004.

[50] D. Kossmann. The State of the art in distributed query processing. *ACM Comput. Surv*, 32(4):422–469, 2000.

[51] A. Magurran. *Ecological Diversity and Its Measurement*. Princeton University Press, 1988.

[52] Mammal Networked Information System (MaNIS). MaNIS website. http://manisnet.org (accessed February 26, 2007).

[53] Sandro Matias. Processamento de consultas ao banco de dados de biodiversidade do BIOTA. Master's thesis, Instituto de Computação - UNICAMP, 2000.

[54] J. McAffer and J. Lemieux. *Eclipse Rich Client Platform: Designing, Coding, and Packaging Java Applications*. Addison Wesley Professional, 2005.

[55] P. McCartney and M. Jones. Using XML-encoded Metadata as a Basis for Advanced Information Systems for Ecological Research. In *Proc. 6th World Multiconference Systemics, Cybernetics and Informatics*, 2002.

[56] D. McGuinness and F. van Harmelen. 2004 OWL Web Ontology Language Overview. W3c recommendation, World-Wide-Web Consortium, http://www.w3.org/TR/owl-features/, 2004.

[57] C. B. Medeiros and F. Pires. Databases for GIS. *SIGMOD Record*, 23(1):107–115, 1994.

[58] C.B. Medeiros, R.S. Torres, A.X. Falcao, T. Lewinsohn, and P.I. Prado. WeBIOS Project. http://www.lis.ic.unicamp.br/projects/webios (accessed February 26, 2007).

[59] M. Minsky. A framework for representing knowledge. In P. H. Winston, editor, *The Psychology of Computer Vision*, pages 211–277. McGraw-Hill, 1975.

[60] P. Morin. *Community Ecology*. Blackwell Science, 1999.

[61] N. Myers, R. A. Mittermeier, C. G. Mittermeier, G. A. Fonseca, and J. Kent. Biodiversity hotspots for conservation priorities. *Nature*, 403:853–858, 2000.

[62] Mark Needleman. The Open Archives Initiative. *Serials Review*, 28(2):156–158, 2002.

[63] Ocean Biogeographic Information System OBIS. OBIS website. http://www.iobis.org (accessed February 26, 2007).

[64] OGC. Topic 5 Features. http://portal.opengeospatial.org/files/?artifact_id=890 (accessed February 26, 2007), 1999.

[65] OGC. Geography Markup Language (GML) 3.0. https://portal.opengeospatial.org/files/?artifact_id=7174 (accessed February 26, 2007), December 2003.

[66] OGC. OGC Reference Model. http://portal.opengis.org/files/?artifact_id=3836 (accessed February 26, 2007), September 2003.

[67] OGC. Web Map Service (WMS) 1.3. http://portal.opengis.org/files/?artifact_id=5316 (accessed February 26, 2007), August 2004.

[68] OGC. Filter Encoding Implementation Specification 1.1.0. http://www.opengeospatial.org/standards/filter (accessed February 26, 2007), May 2005.

[69] OGC. Web Feature Service (WFS) Implementation Specification. http://portal.opengis.org/files/?artifact_id=8339 (accessed February 26, 2007), May 2005.

[70] Open Geospatial Consortium Inc. (OGC). OGC website. http://www.opengeospatial.org (accessed February 26, 2007).

[71] C. Parr, A. Parafiynyk, J. Sachs, L. Ding, S. Dornbush, T. Finin, D. Wang, and A. Hollander. Integrating ecoinformatics resources on the semantic web. In *WWW '06: Proc 15th international conference on World Wide Web*, pages 1073–1074, 2006.

[72] A. Peterson. Predicting species' geographic distributions based on ecological niche modeling. *The Condor*, 103:599–605, 2001.

[73] M. Quillian. *Semantic Memory*. MIT Press, 1968.

[74] P. Ramsey. The State of Open Source GIS. http://www.refractions.net/white_papers/oss_briefing/2005-02-OSS-Briefing.pdf (accessed February 26, 2007), February 2005.

[75] Refractions Research. PostGIS Web site. http://postgis.refractions.net (accessed February 26, 2007).

[76] M. A. Rodríguez, M. J. Egenhofer, and A. D. Blaser. Query Pre-processing of Topological Constraints: Comparing a Composition-Based with Neighborhood-Based Approach. In *Proc SSTD*, volume 2750 of *Lecture Notes in Computer Science*, pages 362–379. Springer, 2003.

[77] A. Seaborne and E. Prud'hommeaux. SPARQL Query Language for RDF. W3C working draft, W3C, October 2006. http://www.w3.org/TR/2006/WD-rdf-sparql-query-20061004/.

[78] N. Shadbolt, T. Berners-Lee, and W. Hall. The Semantic Web Revisited. *IEEE Intelligent Systems*, 21(3):96–101, 2006.

[79] S. Shekhar and S. Chawla. *Spatial Databases - A Tour*. Prentice Hall, 2002.

[80] E. Sirin and B. Parsia. Pellet: An OWL DL Reasoner. In Volker Haarslev and Ralf Möller, editors, *Description Logics*, volume 104 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2004.

[81] B. Smith and A. Varzi. Surrounding Space – The Ontology of Organism-Environment Relations. *Theory in Biosciences*, 121:139–162, 2002.

[82] J. Soberón and A. Peterson. Biodiversity informatics: managing and applying primary biodiversity data. *Philosophical Transactions of the Royal Society B*, 359:689–698, 2004.

[83] Species Analyst project. Species Analyst website. http://speciesanalyst.net (accessed February 26, 2007).

[84] Taxonomic Databases Working Group (TDWG). ABCD Schema 2.0. http://www.bgbm.org/TDWG/CODATA/Schema (accessed February 26, 2007).

[85] Taxonomic Databases Working Group (TDWG). Darwin Core 2 Review. http://darwincore.calacademy.org (accessed February 26, 2007).

[86] R. S. Torres. *An Environment for Managing Images and Spatial Data for Biodiversity Application Development*. PhD thesis, IC-UNICAMP, 2004. Orientadores C. B. Medeiros e A. Falco.

[87] Tree of Life Web Project. Tree of Life website. http://www.tolweb.org (accessed February 26, 2007).

[88] University of Minnesota. MapServer website. http://mapserver.gis.umn.edu (accessed February 26, 2007).

[89] T. Urhan, M. J. Franklin, and L. Amsaleg. Cost-based query scrambling for initial delays. In Laura Haas and Ashutosh Tiwary, editors, *Proceedings of the 1998 ACM SIGMOD International Conference on Management of Data: June 1–4, 1998, Seattle, Washington, USA*, volume 27(2) of *SIGMOD Record (ACM Special Interest Group on Management of Data)*, pages 130–141, pub-ACM:adr, 1998. ACM Press.

[90] W3C. Extensible Markup Language (XML) 1.1. http://www.w3.org/TR/xml11/ (accessed February 26, 2007), 2006.

[91] World Wide Web Consortium (W3C). W3C website. http://www.w3.org (accessed February 26, 2007).

[92] E. Wilson. *Biological Diversity: The Oldest Human Heritage*. New York State Museum, 1999.

[93] R. J. Wilson and J. J. Watkins. *Graphs - An Introductory Approach*. Wiley, 1990.

[94] E. Wong and K. Youssefi. Decomposition - A Strategy for Query Processing. *ACM Trans. on Database Sys.*, 1(3):223–241, September 1976.

[95] M. F. Worboys. *GIS: A Computing Perspective*. Taylor&Francis, 1995.

[96] K. Youssefi and E. Wong. Query Processing in a Relational Database Management System. In *Proc. Int'l. Conf. on Very Large Data Bases*, page 409, Rio de Janeiro, October 1979.