Universidade Federal de Goiás
Instituto de Informática

Alison Carlos Filgueiras

Uma Experiência de Consultas com Palavras-chave em Fontes de Dados Heterogêneas na Web

Goiânia
2013

Universidade Federal de Goiás

Instituto de Informática

Autorização para Publicação de Dissertação em Formato Eletrônico

Na qualidade de titular dos direitos de autor, **AUTORIZO** o Instituto de Informática da Universidade Federal de Goiás – UFG a reproduzir, inclusive em outro formato ou mídia e através de armazenamento permanente ou temporário, bem como a publicar na rede mundial de computadores (*Internet*) e na biblioteca virtual da UFG , entendendo-se os termos "reproduzir" e "publicar" conforme definições dos incisos VI e I, respectivamente, do artigo 5º da Lei nº 9610/98 de 10/02/1998, a obra abaixo especificada, sem que me seja devido pagamento a título de direitos autorais, desde que a reprodução e/ou publicação tenham a finalidade exclusiva de uso por quem a consulta, e a título de divulgação da produção acadêmica gerada pela Universidade, a partir desta data.

Título: Uma Experiência de Consultas com Palavras-chave em Fontes de Dados Heterogêneas na Web

Autor(a): Alison Carlos Filgueiras

Goiânia, 26 de Junho de 2013 .

Alison Carlos Filgueiras – Autor

João Carlos da Silva – Orientador

Auri Marcelo Rizzo Vicenzzi – Co-Orientador

Alison Carlos Filgueiras

Uma Experiência de Consultas com Palavras-chave em Fontes de Dados Heterogêneas na Web

Dissertação apresentada ao Programa de Pós–Graduação do Instituto de Informática da Universidade Federal de Goiás, como requisito parcial para obtenção do título de Mestre em Mestrado em Ciências da Computação.

Área de concentração: Sistema de Informação.

Orientador: Prof. João Carlos da Silva

Co-Orientador: Prof. Auri Marcelo Rizzo Vicenzzi

Goiânia
2013

Alison Carlos Filgueiras

Uma Experiência de Consultas com Palavras-chave em Fontes de Dados Heterogêneas na Web

Dissertação defendida no Programa de Pós–Graduação do Instituto de Informática da Universidade Federal de Goiás como requisito parcial para obtenção do título de Mestre em Mestrado em Ciências da Computação, aprovada em 26 de Junho de 2013 , pela Banca Examinadora constituída pelos professores:

Prof. João Carlos da Silva
Instituto de Informática – UFG
Presidente da Banca

Prof. Auri Marcelo Rizzo Vicenzzi
Instituto de Informática – UFG

Prof. <Nome do membro da banca>
<Unidade acadêmica – Sigla da universidade>

Alison Carlos Filgueiras

Graduou-se em Tecnologia em Processamento de Dados na Universidade Estadual em Goiás em 2002, especializou-se em Web e Sistemas de Informação na Unievangélica Centro Universitário em 2006. Atualmente é Gestor de Tecnologia da Informação do Estado de Goiás, e Gerente de Assuntos Estudantis e Egressos da Pró-Reitoria de Extensão, Cultura e Assuntos Estudantis da Universidade Estadual de Goiás.

À minha incansável e batalhadora genitora.

Agradecimentos

Meus sinceros agradecimentos ao Prof. Dr. João Carlos. Mentor de todas as ideias que permearam esta pesquisa, e que com muita serenidade conduziu as orientações para que ela de desenvolvesse e se desdobrasse nesta dissertação. Agradeço também, de forma especial, ao Prof. Dr. Auri, que disponibilizou tempo e muito trabalho para me auxiliar no desenvolvimento dos artigos, dos softwares e da dissertação. Me sinto honrado de poder fazer parte, se é que assim posso dizer, de um time de pessoas criativas, preparadas, positivas, e que carregam tamanha experiência. Estou certo que levo comigo um aprendizado para a vida, e espero poder repassá-lo da melhor maneira possível, e me inspirando nesses dois mestres. Gostaria de agradecer também aos colegas de mestrado, Adriano, Douglas, Joelias, Alexis e Mariana, que nos momentos difíceis foram cruciais para auxiliarem a manter o foco e a buscar a força necessária para seguir a árdua jornada. Estes colegas que se tornaram grandes amigos. Não poderia esquecer da Adriana, que sempre deu muita força e boas sugestões, do Renan e da Elisabete, que forneceram um apoio grandioso em várias questões relacionadas à pesquisa, e de outros amigos e colegas de trabalho e de estudos que de alguma forma contribuiram, mesmo que com uma pequena palavra de incentivo, para que este projeto se concretizasse. Agradeço ainda, à minha família que sempre me apoiou nessa batalha pelo mestrado. Meus irmãos, sobrinhos, tios, primos. Minha mãe Alice, tão carinhosa e cuidadosa, minha esposa Ju que teve uma participação tão ativa em todo o processo, além de me cobrir com todo amor e dedicação, e à minha mascote e companheira Polly.

Nada acontece por acaso. Ninguém está aqui por acaso. Agradeço por fim, àquele que me concede a dádiva.

"Eu ainda tenho um sonho de que a Web possa ser menos televisão e mais um mar interativo de conhecimento compartilhado."

Berners-Lee,
WebEconomia - Evan I Schwartz p. 28.

Resumo

Filgueiras, Alison Carlos. **Uma Experiência de Consultas com Palavras-chave em Fontes de Dados Heterogêneas na Web**. Goiânia, 2013 . 99p. Dissertação de Mestrado. Instituto de Informática, Universidade Federal de Goiás.

Contexto: Consulta com palavras-chave é um recurso altamente utilizado para recuperação de informações através dos motores de busca disponíveis na Internet. Grande parte da informação existente no mundo, no entanto, não é alcançada pelos processos convencionais de busca por estar armazenada em bancos de dados, na maioria relacionais. A busca integrada de informações de diferentes fontes de dados é explorada por diversos trabalhos, entretanto, não foram encontrados estudos que trouxessem soluções efetivas quando se inclui, dentre essas fontes de dados, bancos de dados relacionais. **Objetivo:** A ênfase deste estudo é apresentar uma solução para recuperação de informação armazenada em fontes de dados heterogêneas, utilizando o protocolo OAI-PMH como mecanismo para viabilizar interoperabilidade. **Método:** Implementação de um sistema que executa consultas por palavras-chave em fontes de dados heterogêneas a partir da coleta de metadados expostos com o protocolo OAI-PMH em provedores de dados. Além disso, é apresentada uma proposta de um *web service* que utiliza métodos públicos para permitir que as informações de bancos de dados relacionais sejam retornadas sem a necessidade de esforços adicionais, tais como conhecimento da estrutura do banco de dados ou uso de SQL. **Resultados:** As simulações produziram o retorno de informações a partir de metadados de objetos digitais e bancos de dados relacionais, obtidos a partir de provedores de dados. A execução de consultas exemplos foi bem sucedida na recuperação de informações em todas as fontes de dados pesquisadas. **Conclusão:** Este trabalho apresenta uma proposta de solução para recuperação de informação armazenada em fontes de dados heterogêneas. A solução proposta mostrou-se viável ao permitir a consulta por palavras-chave em bibliotecas digitais e bancos de dados relacionais utilizando o protocolo OAI-PMH. O *web service* proposto permitiu que informações de bancos de dados relacionais fossem obtidas por aplicações externas, sem que estas necessitem conhecer a estrutura dos bancos de dados consultados ou uma linguagem de consulta como SQL.

Palavras–chave

OAI-PMH, SGBD, Interoperabilidade, Dublin Core

Abstract

Filgueiras, Alison Carlos. **A prototype for querying heterogeneous sources on the web**. Goiânia, 2013 . 99p. MSc. Dissertation. Instituto de Informática, Universidade Federal de Goiás.

Context: keyword research is a highly used feature for retrieval of information through the search engines available on the Internet. Much of the information in the world, however, is not achieved by conventional search to be stored in databases, relational most. The integrated search information from different data sources is explored by several studies, still, no studies were found to bring effective solutions when it includes, among these data sources, relational databases. Objective: The emphasis of this study is to present a solution for retrieval of information stored in heterogeneous data sources using the OAI-PMH as a mechanism to enable interoperability. Method: Implementing a system that runs queries for keywords in heterogeneous data sources from the collection of metadata exposed to OAI-PMH data providers in. Furthermore, the proposal is for a web service that uses public methods to allow information relational databases are returned without the need for additional efforts, such as knowledge of the structure of the database or use SQL. Results: The simulations produced a return of information from metadata of digital objects and relational databases, obtained from data providers. The query execution examples was successful in retrieving information on all data sources surveyed. Conclusion: This work proposes a solution for information retrieval stored in heterogeneous data sources. The proposed solution was feasible to allow consultation by keywords in digital libraries and relational databases using the OAI-PMH. The proposed web service enabled information relational databases were obtained by external applications, without requiring that they know the structure of the databases and a query language like SQL.

Keywords

OAI-PMH, RDBMS, Interoperability, Dublin Core

Conteúdo

Lista de Figuras

Lista de Tabelas

Lista de Algoritmos

Lista de Códigos de Programas

CAPÍTULO 1

Introdução

Este capítulo apresenta ideias introdutórias relacionadas à pesquisa que culminou nesta dissertação. A discussão aqui apresentada tem a intenção de trazer a motivação, objetivos e questões relacionadas ao escopo do trabalho, visando contextualizar o assunto que será desenvolvido no decorrer desta dissertação, e permitir que o leitor tenha uma visão geral do cenário em que se encontra a proposta aqui defendida.

1.1 Contexto e Motivação

A célebre frase que Ted Nelson utilizou durante a concepção do projeto Xanadu, nos anos sessenta, dizia que o "futuro da humanidade está na tela interativa do computador". De fato, meio século depois, nota-se que o presente da humanidade está, em grande parte, dependente desta interação e na infinidade de possibilidades que dela podem surgir. Permitir que o conhecimento humano fosse computado e interligado por algum mecanismo que imitasse o cérebro humano foi um *insight* ainda anterior, por Vannevar Bush, ainda nos anos 40. O *Memex*, como foi idealizado, propunha um mecanismo baseado em associações que supriria as falhas de memória da mente humana e que fosse uma alternativa ao modelo hierárquico de organização de ideias comumente utilizado naqueles tempos.

Após o surgimento de muitos paradigmas e a quebra de tantos outros, a despeito de quanto as tecnologias da informação e comunicação tem evoluído, percebe-se que grande parte das informações de todo o mundo está disposta em repositórios de dados. Uma parcela significativa em forma de documentos (páginas da web, imagens, outros arquivos digitais) e outra, em sistemas gerenciadores de bancos de dados relacionais (SGBDs).

Com o avanço da *Internet*, um volume considerável dessas informações começou a ser disponibilizada por meio de páginas independentes e outros documentos que podem ser acessados por vários mecanismos de busca. No entanto, outra grande parcela de informações não é facilmente alcançada por estar, justamente, armazenada e ser controlada por sistemas gerenciadores bancos de dados relacionais.

Há portanto, grandes esforços da comunidade científica para concepção de melhorias contínuas nas soluções para armazenamento e recuperação de informações de conteúdo digitais. Neste contexto, uma iniciativa chamada *Open Archives Initiative* (LAGOZE; SOMPEL, 2008) teve e tem um papel importante na disseminação da informação científica na *World Wide Web*.

1.2 Justificativa

Em geral, as organizações, sejam elas privadas ou públicas, comerciais ou educacionais, possuem em sua estrutura tecnológica Sistemas Gerenciadores de Bancos de Dados Relacionais. Estes repositórios carregam uma grande parcela de informações que só estão disponíveis para um determinado público, em geral por meio de sistemas específicos que possuem pouca acoplagem com outros sistemas externos. Os proprietários destes bancos de dados, mesmo se quiserem, encontrarão dificuldades para exposição dessas informações na *Internet* total ou parcialmente. De outro lado, as tecnologias de bibliotecas digitais se permitem a recuperação de informação, normalmente com a utilização de metadados, e disponíveis para várias outras tecnologias, como a iniciativa de arquivos abertos *Open Archives Initiative*, que contribuiu sensivelmente para a transformação da comunicação científica, permitindo um potencial alto nível de interoperabilidade entre repositórios digitais(BAPTISTA et al., 2007). Agregar os bancos de dados relacionais à esta possibilidade de interoperabilidade possibilitará um aumento significativo na recuperação de informações nas mais diversas áreas, podendo facilitar inclusive o acesso à informação previsto na Lei Federal 12.527 - Lei do Acesso à Informação, que regula o acesso à informação de todos os órgãos públicos federais, estaduais e municipais.

1.3 O Problema

Os sistemas de bibliotecas digitais permitem a recuperação de informação por meio de palavras-chave. Para este trabalho de recuperação, utilizam-se os metadados, que trazem em si descrições dos objetos digitais armazenados nos repositórios

digitais. Um arquivo em PDF de um livro, por exemplo, poderá ter metadados para expor o autor, a edição, o ano de publicação e o assunto por ele abordado. O padrão de metadados *Dublin Core* (POWELL et al., 2007), que será melhor apresentado nos próximos capítulos, sugere 17 atributos para descrever um objeto digital. A recuperação de informação de bancos de dados relacionais, no entanto, requer que o usuário ou programa tenha, no mínimo, conhecimento do esquema das tabelas, dos atributos a elas pertencentes, uma vez que não há uma forma padrão para nominá-los, e ainda o conhecimento de linguagem de recuperação, como o *SQL*.

Essencialmente, o problema a ser estudado neste trabalho é a recuperação de informações a partir de fontes de dados heterogêneas que tenham alguma relevância para um determinado conjunto de palavras-chave. A estratégia utilizada para resolvê-lo é a construção de um sistema que permite a consulta aos metadados em diferentes fontes de dados, utilizando o protocolo OAI-PMH como interlocutor entre as fontes de dados.

1.4 Objetivos

considerando o contexto e a motivação deste trabalho, e tendo apresentado os problemas a serem estudados, os objetivos deste trabalho são:

- Discutir as tecnologias e iniciativas envolvidas na recuperação de informações de fontes de dados heterogêneas, dando ênfase à utilização do OAI-PMH como protocolo para recuperar metadados na *Web* (LAGOZE; SOMPEL, 2008) e o padrão de metadados *Dublin Core* como padrão para exposição destes dados de diferentes fontes de dados (POWELL et al., 2007).
- Propor uma metodologia de consulta por palavras-chave em diferentes fontes de dados utilizando OAI-PMH como protocolo de interoperabilidade, em que o usuário tenha o mínimo de trabalho na composição da consulta, deixando-o isento de qualquer conhecimento adicional sobre as estruturas de dados envolvidas.
- Apresentar um protótipo de sistema denominado ISSHS *(Information Search System from Heterogeneous Sources)*, que visa permitir o armazenamento, a manutenção e a recuperação de informações a partir de fontes de dados heterogêneas, envolvendo bibliotecas digitais, bancos de dados relacionais e outras fontes de dados.
- Apresentar ao leitor exemplos de interação e resultados obtidos a partir da aplicação do protótipo de sistema apresentado, defendo assim que esta

proposta é viável para um cenário onde seja necessária a recuperação integrada de informação de fontes de dados heterogêneas.

1.5 Organização da Dissertação

A presente dissertação se encontra organizada em seis capítulos, iniciando por este Capítulo de introdução. Os demais capítulos seguem a estrutura abaixo descrita: No Capítulo 2 são apresentados os trabalhos relacionados, seguido pela fundamentação teórica que é apresentada no Capítulo 3. O Capítulo 4 discute a metodologia proposta para a solução. Posteriormente o Capítulo 5 traz detalhes exemplos de interação e resultados. Finalmente, o Capítulo 6 apresenta as conclusões e contribuições obtidas por esta dissertação, abordam a perspectiva dos trabalhos futuros a serem desenvolvidos, e trazem informações sobre a produção bibliográfica realizada no decorrer desta pesquisa.

CAPÍTULO 2

Trabalhos Relacionados

Este capítulo apresenta uma discussão sobre a etapa de revisão bibliográfica, que foi uma etapa essencial para o desenvolvimento desta dissertação. Foi realizado um trabalho de leitura e análise de trabalhos desenvolvidos por pesquisadores atuais e que possuem importância no contexto desta dissertação. Durante a revisão bibliográfica ora realizada, muitos artigos, dissertações, manuais e teses foram analisados, e receberam atenção em questões pontuais que direcionaram as discussões e os objetivos a serem alcançados por este trabalho. No entanto, para não tornar o capítulo exaustivo, alguns aspectos dos trabalhos são apresentados de forma simplificada, e aqueles que se mostraram mais importantes para o desenvolvimento de reflexões sobre questões específicas como, ferramentas, técnicas e algoritmos que adotam a estratégia de busca por palavras-chave, e ou buscam integração entre repositórios heterogêneos, serão discutidos com um maior nível de detalhes.

Ressalte-se que no início desta pesquisa foi realizada uma revisão sistemática (RAMADA et al., 2013) que tinha como principal objetivo a busca do estado da arte na temática *consulta com palavras-chave* visando:

- Identificar estudos relevantes disponíveis acerca de consulta com palavras-chave; e
- Analisar as técnicas desenvolvidas para processar consulta com palavras-chave.

Esta revisão teve como questão primária *"Quais técnicas são utilizadas atualmente para consulta com palavras-chave em banco de dados relacionais?"*. A *string* de busca para consulta nas bases de dados científicas foi: `(key search OR keyword search OR keyword-based search OR key word search) AND (database OR relational database OR DBMS OR RDBMS)`. A consulta, que foi realizada nas principais bases de dados científicas, retornou 323 trabalhos, e após a aplicação dos critérios de exclusão, obteve-se 86 artigos para serem lidos e revisados.

Apesar da revisão sistemática ter sido focalizada em bancos de dados relacionais, foi possível visualizar pela utilização desta ferramenta metodológica algumas lacunas dentro da área temática, dentre as quais, o que diz respeito à integração de fontes de dados heterogêneas.

Mesmo com algumas lacunas, percebe-se no entanto, que integração entre fontes de dados, interoperabilidade e consultas com palavras-chave em fontes heterogêneas na *Web* são temas extremamente explorados pela comunidade científica nas últimas décadas. Sobretudo nas tecnologias de bibliotecas digitais, vários esforços tem sido demandados para a evolução de técnicas e base ferramental que possibilitem que um conjunto cada vez maior de repositórios digitais possam ser alcançados por sistemas de recuperação de informação.

Nota-se, portanto, que a exploração de fontes heterogêneas, em grande parte das mais recentes pesquisas, tem se dedicado a aplicações para recuperação integrada de objetos digitais, sendo apenas uma pequena parte dedicada à integração com dados provenientes de bancos de dados relacionais, que são altamente utilizados e estão, muitas vezes, em uma camada sub-explorada da *Internet*.

A seguir serão apresentados alguns destes trabalhos encontrados no estado da arte da temática aqui pesquisada. Faz-se necessário salientar que estes trabalhos exploram diversas técnicas e trazem propostas significativas para recuperação da informação. Técnicas e estratégias que inspiraram e foram base para a concepção desta dissertação, mas que no entanto apresentam diferenças fundamentais, em especial nas estratégias de solução aqui defendidas.

Fan and Gui (FAN; GUI, 2007) discutem as abordagens de integração de dados na *Web* dando ênfase à classificação da heterogeneidade em dois aspectos: a) heterogeneidade de modelos, onde dados são modelados distintamente; e b) heterogeneidade de esquemas, onde os dados possuem mesmo modelo mas esquemas distintos. Para a construção de uma solução de integração, vários desafios precisam ser levados em consideração: criar um repositório uniforme integrando dados de fontes heterogêneas, ou seja, dados estruturados, semi-estruturados e não estruturados; criar esquemas de dados de fácil compreensão; e expressar as transformações entre fontes de dados e os novos esquemas de dados integrados que foram gerados. Processos de integração de dados, nesta perspectiva, incluem construir, manter e usar um sistema de integração de dados. Sua construção consistirá da extração de dados, transformação dos dados, limpeza dos dados e classificação dos dados. A manutenção consiste em reverter estes dados em *clusters* e armazéns de dados. Por fim, o processo de utilização deve ser direcionado para concepção de aplicações para que

usuário final possa ter condições efetivas de extrair informações relevantes destes dos integrados.

A abordagem descrita em (FAN; GUI, 2007) se apresenta extremamente útil para a construção de sistemas de mineração de dados em determinado contexto. Diferentemente da proposta apresentada nesta dissertação não utiliza de metadados descritivos e nem de algum protocolo para a interoperação entre fontes de dados. Como será possível visualizar na Seção 3.3, a utilização de metadados e a padronização da coleta por meio de um protocolo, como o OAI-PMH, traz diversas vantagens para interoperabilidade entre repositórios, dentre as quais a facilidade de acesso, e consequentemente, a recuperação da informação por meio de aplicações externas, uma vez que são respeitados critérios padrões de armazenamento e recuperação dessas informações (LAGOZE; SOMPEL, 2008).

Telang et al (TELANG et al., 2008) apresentaram um estudo sobre os principais trabalhos presentes no estado da arte da integração entre dados heterogêneos, comparando e abordando várias técnicas existentes, e propondo para o problema da recuperação integrada de informação o *framework* InfoMosaic. Esta proposta de arquitetura permite que, a partir de termos informados de forma livre pelo usuário, seja gerado um conjunto de palavras-chave que é refinada a partir de novas interações *(feedbacks)* com o usuário. A recuperação da informação proposta por esta arquitetura prevê a extração de dados em ambientes heterogêneos, adicionando um nível de semântica a partir de bases de conhecimento.

A proposta descrita em (TELANG et al., 2008) tem motivação semelhante à desenvolvida nesta dissertação, mas contém diferenças fundamentais na propositura da solução e na utilização de técnicas de recuperação. Nesta dissertação, a proposta se baseia em um protocolo para garantir a interoperabilidade entre fontes de dados heterogêneas, e nessa concepção utiliza os metadados como elemento descritivo das fontes de dados, garantindo o trabalho de coleta destes metadados como uma condição *sine qua non* para a eficácia da recuperação integrada das informações dispostas em bibliotecas digitais, bancos de dados relacionais e outras fontes de dados. Outra diferença fundamental, é que a proposta defendida por esta dissertação aborda detalhes metodológicos na construção de um protótipo que leva em consideração os diversos aspectos que envolvem a heterogeneidade de repositórios, sugerindo o OAI-PMH como protocolo de interoperabilidade, onde as consultas podem ser submetidas a fontes heterogêneas, associando-se a solução à utilização de um *middleware*, e com o uso de alguns algoritmos que permitem a recuperação de informações relevantes para a consulta do usuário. Portanto, vai além da propositura de um modelo teoricamente ideal de recuperação integrada de informações.

Li et al. (LI et al., 2008) e Garcia-Alvarado and Ordonez (GARCIA-ALVARADO; ORDONEZ, 2010) trouxeram colaborações e discussões no campo da recuperação de informação de fontes heterogêneas, levando em consideração dados estruturados provenientes de sistemas de bancos de dados relacionais. Ambos sugerem a construção de ferramentas para recuperação de informação em repositórios heterogêneos, e preveem o uso de palavras-chave para realização de consultas.

Os primeiros (LI et al., 2008) propuseram o *EASE*, um mecanismo de busca que permite que usuários acessem facilmente dados heterogêneos (estruturados, semi-estruturados e não estruturados) sem precisar utilizar ferramentas como *XPath*, *XQuery* ou mesmo *SQL*. Propõe uma nova abordagem na construção de índices invertidos para facilitar a busca por palavras-chave, com um novo sistema de *ranking*, baseado na técnica *r-Radius Steiner Graph*. Diferentemente da abordagem utilizada nesta dissertação, este trabalho mantém a estratégia de indexação dos resultados utilizando a abordagem de grafos com a apresentação de resultados baseada na relevância dos dados pesquisados com a utilização de uma nova proposta de *ranking*. O *framework* EASE pode ser acessado pelo endereço `http://dbgroup.cs.tsinghua.edu.cn/EASE/`.

Garcia-Alvarado (GARCIA-ALVARADO; ORDONEZ, 2010) realizou um estudo de recuperação de informação por meio de palavras-chave entre uma base de dados central (SBGBD) e uma coleção de documentos, dando ênfase à granularidade. O cenário explorado pelo trabalho abrange dados estruturados e a associação deles a documentos digitais como arquivos de texto, imagens e outros. O trabalho explora ainda o cenário paradoxal, onde de um lado se tem dados criados e mantidos por rigorosas técnicas que garantem integridade referencial e uma forte estrutura hierárquica, e de outro lado, dados não estruturados e semi-estruturados a partir de documentos que não possuem sequer processo de validação e garantias de corretude, enfatizando a importância dos dados não estruturados e semi-estruturados para o processo de recuperação da informação, principalmente pelo fato destes dados terem a capacidade de descrever os dados estruturados presentes em um banco de dados relacional, muitas vezes em pormenores. Para a solução, o trabalho propõe dois algoritmos para associar um banco de dados central a uma coleção de documentos.

A abordagem utilizada em Garcia-Alvarado (GARCIA-ALVARADO; ORDONEZ, 2010) requer a utilização de uma base de dados como centralizador de informações, inclusive de documentos não estruturados, o que exige uma construção bem definida de entidades no banco de dados para o relacionamento com recursos internos que podem apresentar algumas vantagens em algumas perspectivas de uso dessas informações. No entanto, para a efetiva recuperação de informação em

fontes de dados que podem estar distribuídas em locais geograficamente distintos, não parece ser um modelo melhor do que a organização em metadados que poderão ser coletados por um provedor de serviços que permitem, inclusive, que estes dados possam disponibilizados para outros fins. O padrão *Dublin Core*, que é melhor discutido na Seção 3.4, pressupõe um conjunto padrão de elementos para a exposição de objetos de dados.

Alguns trabalhos estudaram, mais especificamente, questões relacionadas à recuperação de informação em bibliotecas digitais. Dentre os quais citam-se Ward (WARD et al., 2009), que demonstraram que a migração de coleções entre bibliotecas digitais e preservação de arquivos de dados é possível utilizando carregamento automático de dados e metadados em lote utilizando o *OAI-PMH* como protocolo de interoperabilidade. García-Crespo (GARCíA-CRESPO et al., 2011), que apresentaram o sistema *CallimachusDL*, uma biblioteca digital semântica que oferece pesquisa facetada, possibilidades de acesso melhoradas e uma implementação de prova de conceito. Os autores defendem que o *CallimachusDL* representa uma nova abordagem para as bibliotecas digitais, já que integra rede social e elementos multimídia em um repositório com anotações semânticas. Já Vacari et al (VACARI et al., 2010) discutiram sobre vários aspectos de implantação de bibliotecas digitais, apresentando a experiência técnica da Empresa Brasileira de Agropecuária na implementação de repositórios digitais e provedores de serviço *Infoteca*,*Alice*, *SABIIA*, baseados em software livre, para a inserção no modelo de comunicação científica para *Open Access Initiative*.

Os trabalhos que exploram e propõem ferramentas e técnicas de bibliotecas digitais foram muito úteis para esta dissertação. A proposta aqui defendida parte da utilização do protocolo OAI-PMH, cujo principal objetivo é permitir o acesso a repositórios digitais abertos, sendo portanto, um protocolo bastante explorado nos sistemas de bibliotecas digitais. Em relação a essas técnicas e ferramentas, reitere-se que a possibilidade de recuperação de informação de fontes de dados relacionais por meio do OAI-PMH sugere um possível adicionamento de informações, uma vez que muitas instituições possuem muitas informações em bancos de dados relacionais que podem ser expostas para coleta na *Web* sem a necessidade de transformação de informação em objetos digitais. Neste contexto, a proposta apresentada por essa dissertação oferece um avanço significativo.

Focalizando as técnicas de geração de consultas com palavras-chave em bancos de dados relacionais, Bergamaschi et al. (BERGAMASCHI et al., 2010) propuseram o *Keymantic*, um sistema de busca com palavras-chave em bancos de dados relacionais que não requer conhecimento prévio da estrutura do catálogo do banco

de dados. Ele é bastante eficaz em várias situações em que técnicas tradicionais de pesquisa baseada em palavras são inaplicáveis devido a indisponibilidade do conteúdo do banco de dados para a construção dos índices necessários. Bergamaschi (BERGAMASCHI et al., 2011) propusaram posteriormente uma nova técnica para transformar palavras-chave em expressões SQL com base no algoritmo *Munkres Algoritmh*, utilizando significados semânticos. As técnicas estudadas nestes dois trabalhos são voltadas especificamente para consultas em bancos de dados relacionais. Como a solução proposta nesta dissertação requer a busca de informações também de fontes de dados armazenadas em bancos de dados relacionais, são muito pertinentes nas tarefas de geração de palavras-chave para realização das consultas em fontes de dados relacionais, que são melhores detalhadas na Seção 4.4.

Vale ressaltar que não é o intuito deste trabalho discutir ou propor técnicas de construção de consultas com palavras-chave como *Keymantic*. A intenção para esta tarefa é de se utilizar uma ferramenta existente.

De Oliveira (OLIVEIRA, 2010) abordou a integração de um conjunto de bibliotecas digitais, repositórios e outros provedores de dados por meio do protocolo *OAI-PMH* e a recuperação contextualizada de documentos neste repositório integrado em um contexto diretamente ligado ao uso de ontologias. Esta abordagem, que teve colaborações significativas para a proposta de solução apresentada nesta dissertação, trata de integração de repositórios e a utilização de um ambiente colaborativo *wiki*, que contém uma coleção de artigos associados por meio de *hiperlinks*. A diferença essencial está no objeto de recuperação de informação, uma vez que esta abordagem prevê como condição inicial a utilização de ontologias de domínio e não nas palavras-chave geradas a partir da consulta do usuário. Outra diferença significativa está na ênfase que a proposta defendida nesta dissertação dá na recuperação de informações de múltiplos repositórios heterogêneos, inclusive bancos de dados relacionais que são conectados aos provedores de serviço via *Middleware*.

Nessa mesma perspectiva de abordagem semântica, Dos Santos (SANTOS, 2011) propôs uma arquitetura lógica apoiada por busca semântica para recuperação de fontes de informação em repostitórios de metadados. Esta arquitetura prevê uma extensão das funcionalidades de busca e catalogação de fontes de informação no repositório de metadados, utilizando-se de ontologias e anotações semânticas.

Por fim, Da Silva et al. (SILVA et al., 2012) propuseram o *Metadatabase Extractor*, um mecanismo para a promoção da interoperabilidade entre bancos de dados relacionais e outras fontes de informação. Dentre as contribuições deste trabalho, uma em especial trata da criação de uma estrutura de exposição de metadados de bancos de dados relacionais com elementos do padrão *Dublin Core*

e outros elementos adicionais para permitir o acesso a bancos de dados relacionais. Estes metadados estão disponíveis para serem coletados por meio de requisições OAI-PMH. Este trabalho trouxe contribuições significativas para esta dissertação, uma vez que explorou a exposição de metadados de bancos de dados relacionais e a utilização do *Dublin Core* como padrão de metadados e ainda a propositura do protocolo OAI-PMH para permitir a exposição dos metadados descritivos destes bancos de dados relacionais. As discussões apresentadas nesta proposta são utilizadas para implementação de várias etapas da solução proposta nesta dissertação. No entanto, foram acrescentadas várias proposituras como a criação de um *Service Provider* para a coletar metadados, uma vez que sua ausência inviabiliza qualquer recuperação de informação por meio do protocolo OAI-PMH. Outra diferença está no acesso aos dados em etapas posteriores ao processo de coleta. O padrão *Dublin Core* prevê uma referência inequívoca para o acesso ao recurso que é armazenada no elemento `identifier` (POWELL et al., 2007). Nesta proposta, dois elementos adicionais são utilizados *nome de usuário* e *senha* para o caso de um *service provider* necessitar avançar ao nível dos dados. A arquitetura de solução proposta nesta dissertação sugere a utilização de *web service* para permitir que aplicações externas utilizem de uma interface pública que não necessite interferir em políticas de acesso ao banco de dados.

2.1 Considerações Finais

Este capítulo apresentou os trabalhos relacionados ao tema pesquisado nesta dissertação. Tais trabalhos sugerem várias propostas de solução para algum problema específico dentro da temática aqui explorada, quer seja em bancos de dados relacionais ou bibliotecas digitais. Foi possível, após a revisão da literatura, direcionar este trabalho de acordo com a motivação maior que é permitir a recuperação de informações de fontes de dados heterogêneas a partir de palavras-chave do usuário. No próximo capítulo serão apresentados alguns fundamentos para o desenvolvimento desta dissertação, que permitirão ao leitor uma melhor familiaridade com as tecnologias envolvidas na proposta de solução aqui defendida, e consequentemente, um maior conforto para a compreensão dos capítulos posteriores.

CAPÍTULO 3

Fundamentação Teórica

Este capítulo apresenta a base teórica utilizada para o desenvolvimento desta pesquisa, além de permitir ao leitor o entendimento de termos cruciais pertinentes à temática abordada. A Seção 3.1 apresenta fundamentos sobre consultas com palavras-chave. A Seção 3.2 abordará conceitos de bibliotecas digitais. Na seção 3.3 a interoperabilidade será abordada sobre diferentes perspectivas. A Seção 3.4 abordará os metadados e o padrão *Dublin Core*. Na Seção 3.5 serão desenvolvidas discussões sobre *web services* e *middlewares* e por fim, a seção 3.6 tratará conceitualmente do protocolo OAI-PMH.

3.1 Consulta com Palavas-Chave

Teixeira (TEIXEIRA, 1974) afirma que "por palavra-chave entende-se uma palavra ou frase relativamente curta, que descreve, de alguma forma, características daquilo a que se refere." Em outras palavras, palavras-chave geralmente dizem respeito a um assunto discutido, direta ou indiretamente.

Muito comumente utilizada nos motores de busca da web, como o *Google*, *Yahoo* e *Bing*, as consultas com palavras-chave são os recursos mais amplamente explorados por usuários da *Internet*. Os sistemas de busca na *Web*, normalmente, utilizam-se de uma combinação de palavras-chave, e trazem como resultado da busca, uma lista de documentos relevantes àqueles termos de entrada, com algum critério de organização (SAELEE; BOONJING, 2008). Por outro lado, as consultas em sistemas que utilizam bancos de dados relacionais são normalmente tarefas bem mais árduas. Tradicionalmente, se um usuário de bancos de dados deseja realizar uma simples consulta utilizando palavras-chave, precisará, necessariamente, possuir conhececimentos, tanto do *catálogo do banco de dados* quanto de uma linguagem de consulta como a *SQL* ou *XQuery* (CHEN et al., 2011). Tendo em

vista a popularização e o sucesso dos motores de busca na *Web*, torna-se natural esperar que os usuários possam efetivamente ter a mesma qualidade e facilidade nas consultas com palavras-chave também em sistemas de bancos de dados relacionais. Existem duas grandes vantagens no desenvolvimento dessas tecnologias. Primeiro, os usuários casuais ficam isentos de gastar tempo com o aprendizado de uma linguagem estruturada ou esquemas de dados. E segundo, isto permite facilidades de acesso a bases de dados heterogêneas(CHEN et al., 2011).

A recuperação de informações baseada em palavras-chave pode ser usada não apenas para recuperação de dados textuais, mas também para outros tipos de documentos digitais, desde que tenham palavras-chave associadas. Existem diferenças entre esse modelo e os utilizados no sistemas tradicionais de bancos de dados (SILBERSCHATZ et al., 1999):

- Os sistemas de bancos de dados tratam de várias operações que não são consideradas nos sistemas de recuperação de informações. Elas dizem respeito principalmente à característica organizacional dos sistemas de bancos de dados relacionais que requerem uma composição de dados estruturados de forma relativamente complexa, enquanto os sitemas de recuperação de informação se organizam, normalmente, como uma coleção de documentos não estruturados.
- Os sistemas de recuperação de informação lidam com várias questões que não são consideradas adequadamente nos sistemas de banco de dados. Uma dessas questões é a busca aproximada por palavra-chave e da classificação de documentos em graus de relevância ao que se está sendo consultado.

3.2 Bibliotecas Digitais

Com o crescimento estrondoso da *Internet*, assim como a busca crescente por conteúdo, muitos recursos são cada vez mais requisitados na *Web* contemporânea. É notável uma crescente demanda por bibliotecas digitais de forma rápida e para os mais diversos fins. A necessidade de associação das bibliotecas digitais às redes de comunicação, em especial à rede mundial de computadores é clara, a partir do momento em que se percebe a própria natureza das bibliotecas digitais, que é de tornar acessível e, preferencialmente público, um patrimônio documental. Tammaro (TAMMARO; SALARELLI, 2008) afirmam que a centralidade no usuário é um dos elementos que realmente qualificam uma biblioteca digital, e mesmo que se construa a biblioteca digital mais avançada do mundo, com as tecnologias mais sofisticadas, com os documentos mais atraentes e com o catálogo mais eficiente, se ela não for

disponibilizada na *Web*, este usuário precisará dedicar muito tempo para obter uma informação. A *Web* cria no leitor a ilusão de que ele realmente se move, em absoluta liberdade, no universo dos documentos. Uma liberdade sem barreiras, sem estorvos tecnológicos. Liberdade esta que muitas vezes não existe nas bibliotecas 'tradicionais' (TAMMARO; SALARELLI, 2008).

Sayão (SAYãO, 2009) afirma que as bibliotecas digitais surgem num contexto que apresenta de um lado a integração de tecnologias e o barateamento do armazenamento em massa, e do outro, a disponibilidade crescente de conteúdos digitais por um custo economicamente viável, com um fenômeno conhecido como coerência das mídias digitais. Este fenômeno abre uma possibilidade singular para abertura de novos serviços que podem ser concebidos pela integração de objetos digitais heterogêneos. A abertura para a multidisciplinaridade presente nas bibliotecas digitais fortalece um fenômeno interessante que alia esforços e relacionamentos entre diversas áreas do conhecimento para a construção de espaços inter-relacionados (SAYãO, 2009).

Candela et. al. (CANDELA et al., 2007) enfatiza que este é um espaço sinérgico de um grande número de áreas da tecnologia da informação e várias outras disciplinas e campos de pesquisa, como biblioteconomia, ciência da informação, museologia, arquivologia e gestão do conhecimento. No contexto deste trabalho, percebe-se a importância de aliar essas tecnologias e bancos de dados relacionais para o armazenamento e a recuperação da informação, uma vez que há um grande esforço para criação e evolução de tecnologias para bibliotecas digitais, mas sem necessariamente diminuir o uso dos SGBD's. Este fato torna visível que, em muitos casos, essas duas tecnologias podem se completar harmonicamente.

Nas bibliotecas digitais, as informações também são digitais. Os documentos, por sua vez, não poderiam ser diferentes. Tais documentos, conforme (TAMMARO; SALARELLI, 2008) não são mais somente aquilo que é legível na forma da palavra escrita. É fruto da capacidade de quem pesquisa ao interrogar um material. As respostas à interrogação, tornam-se portanto algo que não é fruto da casualidade. O documento digital coloca-se à frente do processo de codificação das mensagens que serão repassadas de autores para indivíduos ou comunidades de pesquisadores (consumidores da informação). O registro desses documentos implica em uma série de dados que precisam ser armazenados para lhe fazerem referência. O ato de pesquisar os documentos digitais requer uma certa decodificação desses dados para certos critérios dos indivíduos que se relacionam nesses processos, quer sejam sistemas informatizados ou pessoas propriamente ditas.

O pesquisador é o verdadeiro artífice do documento em relação ao seu próprio interesse, à própria cultura: dez diferentes análises, dez documentos, mas um único objeto (TAMMARO; SALARELLI, 2008).

Para implementação de sistemas de Bibliotecas Digitais se faz necessário um esforço integrado de serviços que são bem parecidos com aqueles serviços desenvolvidos nas bibliotecas físicas tradicionais. Organização, catalogação, manutenção das coleções, ciclo de vida dos documentos e outras. Nas bibliotecas digitais, no entanto, o recursos não podem ser usados sem tecnologias apropriadas de arquivamento e acesso. E isso se aplica tanto a recursos originalmente digitais quanto aqueles que foram convertidos de formatos analógicos para digitais (TAMMARO; SALARELLI, 2008).

Várias ferramentas e *frameworks* para o desenvolvimento de bibliotecas digitais estão disponíveis, e grande parte é composta por softwares livres, como **E-Prints Fedora**, **DSPACE**, **jOAI**, e **Solr** (VACARI et al., 2010). Abaixo uma breve descrição de algumas dessas tecnologias:

- **E-Prints**: Pioneiro entre os softwares desenvolvidos especificamente para a armazenamento e distribuição de informações científicas, para ser disseminado e utilizado em todo o mundo (BAPTISTA et al., 2007). É uma ferramenta aberta, apropriada para a construção de repositórios institucionais, relativamente fácil de instalar e adaptável às necessidades de qualquer instituição de ensino e pesquisa (OLIVEIRA, 2010).

- **Fedora** (Um acrônimo de *Flexible Extensible Digital Object and Repository Architecture*): trata-se de uma tecnologia *open source* para a implementação de repositórios digitais em geral. Sua arquitetura possui um núcleo que permite criar, gerenciar, armazenar, pesquisar e reusar objetos digitais. Todas suas funções são disponibilizadas por meio de serviços e podem ser acessadas por meio de *web services* (OLIVEIRA, 2010). Em sua arquitetura inclui-se ainda um modelo de relacionamento genérico baseado em *RDF (Resource Description Framework)* que representa relações entre objetos e seus componentes, que permite maior integração semântica. Por se tratar de um software livre, tem sua aplicação em uma grande variedade de sistemas de repositórios digitais como bibliotecas digitais institucionais, sistemas de aprendizado, arquivos digitais, e outros (LAGOZE et al., 2006).

- **DSpace**: Tecnologia que nasceu de uma iniciativa conjunta entre as bibliotecas do *Massachussets Institute of Technology (MIT) e a HP-Labs* em 2002

(BAPTISTA et al., 2007). Baseia-se em um modelo de informação organizacional focado em comunidades e coleções. Portanto, pode refletir todo o conjunto de unidades administrativas de uma instituição. Permite os mais variados tipos de formatos de arquivos digitais, incluindo textos, sons e imagens (OLIVEIRA, 2010).

- **jOAI**: Uma ferramenta *open source*, desenvolvida pela *Digital Learning Sciences (DLS)* da *University Corporation for Atmospheric Research*, voltada para a *Open Archives Initiative* que funciona tanto como provedor de dados quanto coletor de metadados (JOAI..., 2012). Altamente configurável, permite o acesso e coleta em múltiplos repositórios, bem como a especificação de conjuntos *OAI-PMH* (SIDHUNATA et al., 2012). Trata-se e uma aplicação *web* Java, executável em um *servlet container* como o *Apache Tomcat*.

- **Solr**: Uma plataforma de busca *open source* do projeto *Apache Lucene* que permite várias operações como uma poderosa busca textual, busca facetada, clusterização dinâmica e busca geoespacial. Permite também a integração de repositórios e busca distribuída. *Solr* é escrita em Java e funciona como um servidor independente de pesquisa de texto completo dentro de um *servlet container* como o *Apache Tomcat*, utilizando a biblioteca de busca do *Lucene*. Possui *APIs HTTP/XML* e *JSON* que facilita sua acomplagem a aplicações externas (FOUNDATION, 2012).

3.3 Interoperabilidade

O termo interoperabilidade é aplicado de diversas maneiras tanto no campo da Ciência da Computação, da Ciência da Informação, como em outras áreas do conhecimento. Em geral, o termo se aplica a capacidade de operações de troca entre elementos, autônomos ou não, para um determinado fim. Não se trata apenas de um termo associado a assuntos meramentes técnicos ou tecnológicos. Faz necessário considerar diferenças culturais e diferentes percepções de conceitos, ou seja, não se pode considerar apenas apenas uma interoperabilidade tecnológica, mas também uma interoperabilidade semântica (BACKHOUSE et al., 2003).

O Dicionário Online de Cambrigde (ONLINE..., 2013) descreve que interoperabilidade é o grau em que os dois produtos, programas, etc, podem ser utilizados em conjunto, ou a qualidade de um sistema em ser capaz de ser utilizado em conjunto.

Backhouse (BACKHOUSE et al., 2003) afirma que interoperabilidade significa que aplicações podem trocar dados e servicos com consistência e meio efetivo, cobrindo diferentes plataformas de hardware e software. Já em Press (PRESS, 2004), encontramos que interoperabilidade é a habilidade de múltiplos sistemas de informação, com diferentes plataformas de *hardwares* e *software*, estrutura de dados e interfaces, para trocar dados com o mínimo de perda de conteúdo e funcionalidade.

Neste trabalho, observamos a interoperabilidade em três diferentes perspectivas. 1) interoperabilidade no nível dos bancos de dados relacionais; 2) interoperabilidade no nível das bibliotecas digitais; e 3) a interoperabilidade no nível dos metadados.

3.3.1 Interoperabilidade no Contexto dos Bancos de Dados Relacionais

Sistemas de bancos de dados relacionais, em geral, são concebidos para armazenar um conjunto bem particular de dados, atendendo a termos e regras compreendidas por um grupo específico de usuários. Desta forma, existem vários desafios quando se necessita permitir interoperabilidade entre dois ou mais bancos de dados relacionais (TRINH et al., 2007). Dois ou mais bancos de dados relacionais são ditos heterogenêneos quando apresentam diferenças na forma utilizada para modelagem dos dados.

Conflitos ocorrem pelo fato dos sistemas de bancos de dados relacionais usarem diferentes modelos de dados. Estes conflitos podem ser classificados como (PITOURA et al., 1995):

- **Conflitos de identidade:** ocorrem quando o mesmo conceito é representado por diferentes objetos em diferentes bancos de dados.
- **Conflitos de esquemas:** ocorrem quando esquemas de dados representam o mesmo conceito mas não são equivalentes permitindo: a) conflitos de nomes, quando o mesmo nome é usado para diferentes conceitos (homônimos) ou quando o mesmo conceito é descrito por diferentes nomes (sinônimos); e b) conflitos estruturais, quando o mesmo conceito é representado por diferentes construções do modelo de dados, ou quando representadas pelas mesmas construções mas elas possuem diferentes estruturas (diferentes relacionamentos, dependências) ou diferentes comportamentos (operações).
- **Conflitos Semânticos:** ocorrem quando o mesmo conceito é interpretado de forma diferente em diferentes bancos de dados.

- **Conflitos de Dados:** ocorrem quando os valores de dados de um mesmo conceito são diferentes em diferentes bancos de dados.

Além dos vários conflitos apresentados acima, outras questões precisam ser levadas em consideração quanto à interoperabilidade de bancos de dados relacionais. Sistemas Gerenciadores de Bancos de Dados, apesar de atenderem determinados padrões, possuem estratégias diferentes de armazenamento e recuperação de informações que podem interferir em várias tarefas que visem a interoperabilidade entre estes dados. De um ponto de vista tecnológico, estas previsões precisam constar nas nos sistemas responsáveis pela integração entre os bancos de dados. Políticas de segurança também precisam ter atenção especial neste caso, e para tal, os SGBDs apresentam também diferentes estratégias e diferentes tecnologias. Por fim, cabe reafirmar que durante a concepção desta dissertação, não foi identificado um protocolo ou outra tecnologia conhecida, especificamente voltada para interoperabilidade entre bancos de dados relacionais e outras fontes de dados. A interoperabilidade entre bancos de dados pode ser encontrada, por exemplo, nos sistemas federados onde os participantes de uma federação de bancos de dados compartilham seus objetos de dados sem perderem o controle sobre eles.

3.3.2 Interoperabilidade no Contexto das Bibliotecas Digitais

A ideia geral de integração e interoperabilidade em bibliotecas digitais consiste na disponibilização de serviços de recuperação de recursos informacionais heterogêneos, armazenados em diferentes repositórios e servidores na *Web*, utilizando-se de uma interface única (OLIVEIRA, 2010). Para Tammaro & Salarelli (TAMMARO; SALARELLI, 2008) a infra-estrutura para interoperabilidade do acesso à biblioteca digital é a estratégia para o futuro. Isto é, possibilitar que os recursos sejam efetivamente integrados em ambientes interativos e de fácil recuperação para o usuário possibilita um grande avanço na recuperação de informações digitais. O protocolo *OAI-PMH* assim como outros protocolos como o Z39.50 são exemplos de abordagens que permitem a interoperação de bibliotecas digitais em vários níveis, e mostram que do ponto de vista técnico, a interoperabilidade entre bibliotecas digitais é possível, mas tem seus custos (TAMMARO; SALARELLI, 2008).

3.3.3 Interoperabilidade no Contexto dos Metadados

Os metadados estruturados trazem na sua essência características de interoperabilidade. É facilmente notável que a manutenção de uma estrutura padronizada permite que objetos diferentes tenham uma descrição que leve em consideração aspectos comuns, facilitando assim a integração de informações. Interoperabilidade não só exige uma padronização de formatos, mas também um compromisso com os padrões de qualidade definidos pela comunidade (URBAN, 2012).

Haslhofer (HASLHOFER; KLAS, 2010) afirma que a interoperabilidade entre metadadados é um pré-requisito para uniformização de acesso objetos digitais em sistemas de informação múltiplos e heterogêneos. É um conceito que tem definição ampla, alcançando desde capacidades de baixo nível de sistemas de computação em entenderem arquivos, através de habilidades de alto nível em preservar significado da informação que é trocada (URBAN, 2012).

3.4 Metadados e o Padrão Dublin Core

Metadados são normalmente vistos, de forma simplificada, como dados sobre outros dados. Trata-se de informações estruturadas que descrevem, explicam, localizam ou, de alguma maneira, facilitam a recuperação de uma fonte de informação (PRESS, 2004). Apesar de possuírem uma estruturação simplificada, são capazes de carregar informações extremamente relevantes acerca do objeto a que está relacionado, como autor, data de publicação, fonte de publicação, tamanho do documento, ou gênero do documento (livros, discos e outros documentos) (KOWATA, 2011). Em adição, mesmo com aparente simplicidade, o termo *metadata* é utilizado de forma diferente em diferentes comunidades. Assim, a busca por padronização representa grandes esforços da comunidade científica. Uma das principais razões para existência e a utilização desses padrões está relacionada com gerenciamento e disponibilização da informação (DZIEKANIAK, 2007).

Para Press (PRESS, 2004) e Santos (SANTOS, 2011) os metadados podem ser classificados em três tipos: (1) **metadados descritivos** descrevem uma fonte de informação para fins de identificação e recuperação utilizando elementos como título, autor, resumo e palavras-chave; (2) **metadados estruturados** descrevem a organização interna dos objetos e das relações entre eles, o exemplo mais comum é o esquema do banco de dados; e (3) **metadados administrativos**, que apoiam as atividades de gerenciamento do acervo de recursos de informação como controle de permissões de acesso, localização de arquivos e critérios de avaliação da qualidade.

Dublin Core, por sua vez, é um esquema de exposição de metadados para descrição de objetos digitais. Seu principal objetivo é promover interoperabilidade entre metadados a partir de um padrão que sugere informações relevantes comuns (OLIVEIRA, 2010). Por ser construído sobre uma base de princípios simples, *Dublin Core* tem a intenção de ser uma *Língua Franca* que facilita a troca de informações interoperáveis (URBAN, 2012).

O padrão *Dublin Core* possui dois níveis. O **Simples**, que possui 15 elementos básicos, e o nível **Qualificado**, que traz três elementos adicionais referentes a auditoria, direitos autorais e proveniência. A Tabela 3.1 traz os elementos do *Dublin Core* e as informações que os complementam (POWELL et al., 2007).

Kowata (KOWATA, 2011) afirma que o *Dublin Core* pode ser visto como uma pequena linguagem para fazer declarações de uma classe particular de recursos, onde há duas classes de termos: elementos (substantivos) e qualificadores (adjetivos). Sua intenção é trazer uma padronização de forma simples com a possibilidade de adaptação para suprir necessidades adicionais de cada comunidade de usuários (POWELL et al., 2007). Desta forma, o padrão *Dublin Core* apresenta quatro níveis de interoperabilidade conforme demonstra a figura 3.1.

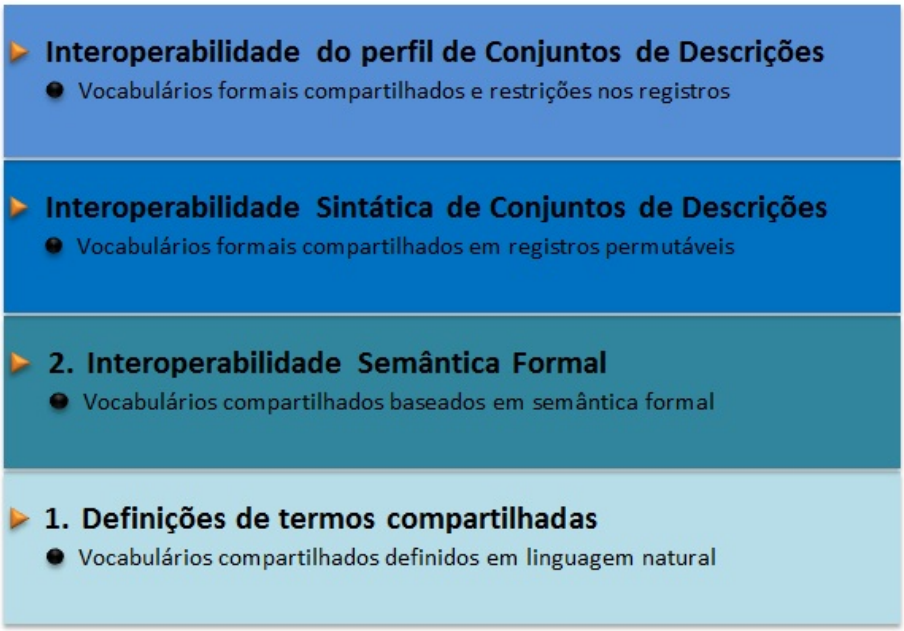

Figura 3.1: *Níveis de interoperabilidade do Dublin Core - baseado em (NILSSON; BAKER, 2009).*

O nível 1 corresponde à utilização de definições em linguagem natural dos termos Dublin Core. No nível 2, há uma correspondência à utilização implícita ou explícita da semântica *RDF* subjacente aos termos DCMI. Dessa forma, qualquer utilização dos termos necessita de ser precisa na sua conformidade com o modelo RDF e com os

Tabela 3.1: *Elementos do Dublin Core (POWELL et al., 2007)*

contributor	
URI:	http://purl.org/dc/elements/1.1/contributor
Recurso:	Contributor
Definição:	Uma entidade responsável por qualquer contribuição para o recurso.
coverage	
URI:	http://purl.org/dc/elements/1.1/coverage
Recurso:	Coverage
Definição:	A cobertura espacial ou temporal de um recurso, a aplicabilidade espacial de um recurso, ou sua jurisdição.
creator	
URI:	http://purl.org/dc/elements/1.1/creator
Recurso:	Creator
Definição:	Uma entidade principal responsável por gerar o recurso.
date	
URI:	http://purl.org/dc/elements/1.1/date
Recurso:	Date
Definição:	Um ponto ou um período de tempo associado ao ciclo de vida do recurso .
description	
URI:	http://purl.org/dc/elements/1.1/description
Recurso:	Description
Definição:	Uma descrição do recurso. (pode ser um resumo, um índice, uma representação gráfica, etc)
format	
URI:	http://purl.org/dc/elements/1.1/format
Recurso:	Format
Definição:	O formato de arquivo, meio físico, ou dimensões do recurso.
identifier	
URI:	http://purl.org/dc/elements/1.1/identifier
Recurso:	Identifier
Definição:	Uma referência inequívoca ao recurso dentro de um dado contexto
language	
URI:	http://purl.org/dc/elements/1.1/language
Recurso:	Language
Definição:	A linguagem de recurso (PT,EN, etc).
publisher	
URI:	http://purl.org/dc/elements/1.1/publisher
Recurso:	Publisher
Definição:	Uma entidade responsável por tornar o recurso disponível
relation	
URI:	http://purl.org/dc/elements/1.1/relation
Recurso:	Relation
Definição:	Um recurso relacionado
rights	
URI:	http://purl.org/dc/elements/1.1/rights
Recurso:	Rights
Definição:	Informações de direitos sobre recurso.
source	
URI:	http://purl.org/dc/elements/1.1/source
Recurso:	Source
Definição:	Um recurso relacionado a partir do qual o recurso descrito é derivado.
subject	
URI:	http://purl.org/dc/elements/1.1/subject
Recurso:	Subject
Definição:	Assunto to recurso (palavras-chave, frases-chave e etc)
title	
URI:	http://purl.org/dc/elements/1.1/title
Recurso:	Title
Definição:	Título. O nome dado ao recurso.
type	
URI:	http://purl.org/dc/elements/1.1/type
Recurso:	Type
Definição:	A natureza ou gênero do recurso.

domínios e contra-domínios dos termos. No nível 3, há exigência de uma utilização explícita do Modelo Abstrato da DCMI nos metadados, onde estes devem estar estruturados com a utilização de noções de Descrição e Conjuntos de Descrições do Modelo Abstrato da DCMI (POWELL et al., 2007). Por fim, o nível 4 fornece um modelo de informação e expressão XML de restrições estruturais num conjunto de descrições, com possibilidade de reutilizações máximas, com elentos como requisitos funcionais, um modelo de domínio e um perfil de conjuntos de descrição para cobertura completa ao conjunto de metadados (NILSSON; BAKER, 2009).

3.5 Web Services

Uma visão simplificada de *web services* é que estes são sistemas *Web* que ofecerem serviços gerais para aplicações remotas sem interações imediatas de usuários finais (TANENBAUM; STEEN, 2006). Esta visão sugere portanto que exista uma interface para o fornecimento deste serviço que permite uma série de operações que podem ser fornecidas de um servidor para um cliente na *Internet* (COULOURIS et al., 2005). Podem ser vistos também como componentes de *software* que podem ser chamados por outros aplicativos (OLIVEIRA, 2010). Tanenbaum & Steen (TANENBAUM; STEEN, 2006) afirmam que um *Web service* nada mais é do que um serviço tradicional, como por exemplo, uma previsão de tempo, serviço de nomeação ou um fornecedor eletrônico, que pode ser oferecido pela *Internet*. O que o torna especial é o fato destes serviços obedecerem a um conjunto de padrões que permitirão que eles sejam localizados e acessados por aplicações clientes que também adotem esses padrões. Isto é, independente da sua estrutura ou tecnologia de hardware e software, quem fornece um serviço e quem o utiliza devem atender as mesmas especificações dialógicas. O transporte de dados nos *web services* é realizado via protocolo HTTP, mas eles possuem protocolos específicos para garantir o processo de localização e fornecimento de serviços.

Como pode ser visualizado na Figura 3.2, o princípio de funcionamento de um *web service* é bastante simples, e bem semelhante aos serviços tradicionais de comunicação entre processos de sistemas distribuídos. Uma aplicação servidora fornece serviços para aplicações clientes. Para que esta operação de consumo seja efetiva será necessária uma padronização, no que se refere à forma como esses serviços são descritos e como podem ser consultados (TANENBAUM; STEEN, 2006). Na arquitetura de um *web service* o serviço de diretório é o componente responsável pelo armazenamento das descrições de serviços, que deve obedecer o padrão UDDI (*Universal Description, Discovery and Integration*) que possui um

layout específico os para descrevê-los por meio de uma linguagem denominada WSDL (*Web Services Definition Language*). A WSDL contém as definições das interfaces fornecidas pelo serviço, que inclui as operações que podem ser executadas e os tipos de dados envolvidos neste processo. Esta descrição forma a base de um acordo entre o cliente e o servidor (fornecedor) (COULOURIS et al., 2005). A comunicação entre fornecedores e clientes é realizada pelo protocolo SOAP (*Simple Object Access Protocol*) que é a estrutura que permite a padronização de grande parte da comunicação entre dois processos (TANENBAUM; STEEN, 2006).

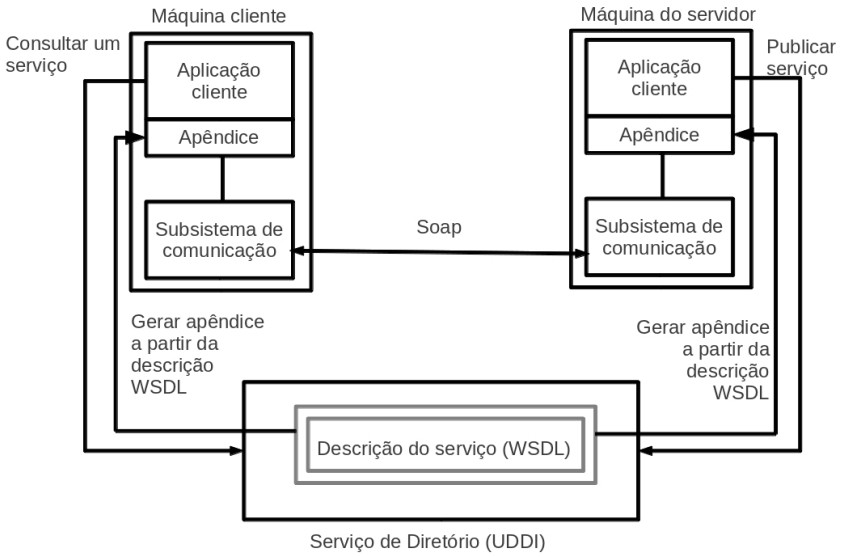

Figura 3.2: *Princípio de um web service (TANENBAUM; STEEN, 2006)*

3.5.1 Middlewares

Para Vikoski (VINOSKI, 2002) *middleware* é o que permite integração. O autor defende que os *middlewares* existem em várias formas por muito anos, em tecnologias como *Customer Information Control System (CICS)*, *Common Object Request Broker Architecture (Corba)*, *Microsoft's Component Object Model (COM)*, *Java 2 Enterprise Edition (J2EE)*, e como mais explorado ultimamente, em forma de *web services*. Em outras palavras, trata-se de uma camada que permite a comunicação entre aplicações distribuídas. Sugere-se portanto, que seja composto de um ou mais softwares e deve garantir um certo nível de transparência entre as aplicações envolvidas (EMMERICH; KAVEH, 2002). No contexto da proposta de solução defendida neste trabalho, a construção de *middleware* se faz necessária para que os dados de bancos de dados relacionais possam ser recuperados em etapa

posterior à coleta de metadados. O *middleware* fornece condições para que aplicações externas possam realizar requisições para os bancos de dados nele conectados por meio de métodos públicos.

3.6 Protocolo OAI-PMH

O protocolo *OAI-PMH (Open Archives Initiative - Protocol for Metadata Harvesting)* é um protocolo que define um mecanismo de coleta e registro de metadados em repositórios digitais. Foi desenvolvido pela *Open Archives Initiative*, entidade que tem por objetivo desenvolver e promover normas facilitem a interoperabilidade e livre circulação de metadados e conteúdos entre diferentes entidades de sistemas de informação (FERROS et al., 2010). Trata-se de um mecanismo que contém poucos obstáculos para a interoperabilidade e para transferência de dados entre repositórios digitais. Pode ser visto como uma interface que permite que um servidor possa expor metadados de objetos residentes para aplicações externas que desejam coletar esses dados, promovendo assim interoperabilidade e extensibilidade (KOWATA, 2011).

Há duas classes de provedores participantes no protocolo *OAI-PMH*: 1) *data providers* que administram os sistemas dão suporte ao *OAI-PMH*; e 2) *service-providers*, que usam a coleta de metadados baseados no *OAI-PMH* como base para construção de serviços de valor agregado (LAGOZE; SOMPEL, 2008).

Os *data providers* atuam na transferência de metadados fornecendo-os aos *service providers*, que por sua vez, vão oferecer outros serviços após a coleta, tais como pesquisa, referência ou estatística, tendo como base as informações recolhidas (FERROS et al., 2010). A comunicação é feita por meio de requisições *HTTP* e são acessíveis por meio de uma *URL*. Além do endereço para a coleta, é necessário a identificação do que será coletado e como a coleta será realizada. Os seis comandos de requisição (verbos) definidos para este processo são (KOWATA, 2011): *Identify, ListIdentifier, ListMetadataFormats, ListSets, ListRecord, GetRecord*. Todas as respostas a todas as requisições são codificadas em XML

Um repositório *OAI-PMH* é um servidor acessível em uma rede que pode processar as seis requisições (verbos) presentes no protocolo *OAI-PMH*. Este repositório é gerenciado por um *data provider* para expor metadados para aplicações clientes que emitem as requisições. Afim de permitir configurações diversas para os repositórios, o *OAI-PMH* compreende três tipos distintos de entidades relacionadas a metadados. ***Resource*** (recurso) que é um objeto o qual os metadados descrevem, por exemplo um arquivo PDF. ***Item*** (item) que é um componente de um repositório pelo qual

os metadados de um recurso podem ser disseminados. E ***Record*** (registro) que é um metadado em um formato específico, retornado em forma de XML em resposta a uma requisição do protocolo (LAGOZE; SOMPEL, 2008).

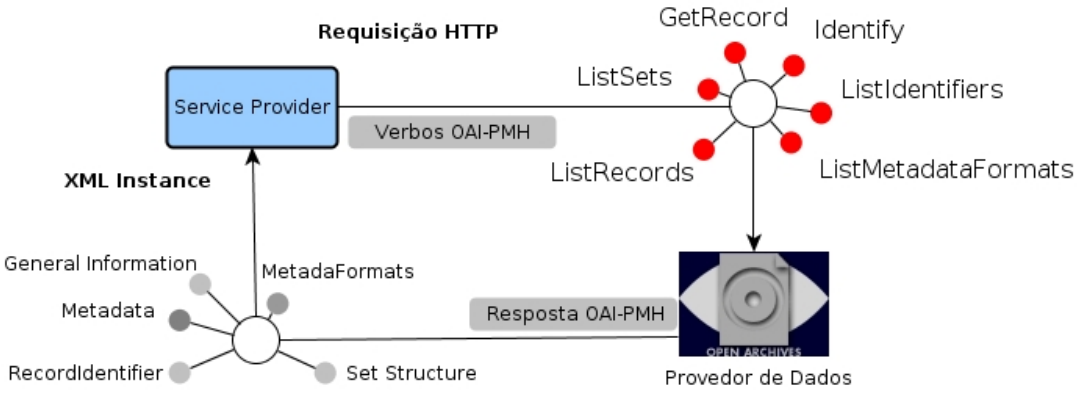

Figura 3.3: *Esquema do OAI-PMH. Baseado em (LAGOZE; SOMPEL, 2008)*

3.6.1 Harversters

Harversters são coletores de metadados, e este serviço de coleta é um processo unilateral pelo qual os provedores de serviços, a partir de uma lista de repositórios disponíveis e registrados na OAI, podem realizar periodicamente uma busca de provedores de dados, colhendo os metadados para exibição de acordo com consultas requisitadas por usuários (GARCIA; SUNYE, 2003). A coleta de metadados pode ser total ou baseada em critérios. Para tanto, existem critérios para coleta que podem ser **baseados em dados (*Data-based*)**, e que são coletados apenas os metadados incluídos e/ou alterados após uma data especificada, e **baseados em conjuntos(*Set-based*)**. O protocolo define *set* como uma estrutura opcional para agrupar itens num repositório para o propósito de uma coleta seletiva de registros. Esta estrutura funciona como um agrupador de objetos que representa a hierarquia do repositório, como por exemplo, um assunto ou uma disciplina dentro de um determinado repositório (GARCIA; SUNYE, 2003) e (LAGOZE; SOMPEL, 2008).

3.6.2 Requisições e Respostas

As requisições (ou verbos) do protocolo *OAI-PMH* são acompanhadas de argumentos de três tipos específicos. a) *required*, requerido, que indica que o argumento deve ser incluído na requisição obrigatoriamente. b) *optional*, opcional, que indica que o argumento pode ser incluído na requisição. e c) *exclusive*, exclusivo, que indica

que o argumento pode ser incluído na requisição, mas deve ser o único. Abaixo são apresentados os comandos de requisição (verbos) do protocolo *OAI-PMH*, acompanhados de uma breve descrição de suas características (LAGOZE; SOMPEL, 2008). Para elucidar a ideia de comunicação do *OAI-PMH*, a figura 3.4 traz um exemplo de resposta em *XML* da requisição *GetRecord*.

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="http://www.rdm.uff.br/lib/pkp/xml/oai2.xsl" ?>
<OAI-PMH xmlns="http://www.openarchives.org/OAI/2.0/"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:schemaLocation="http://www.openarchives.org/OAI/2.0/
                http://www.openarchives.org/OAI/2.0/OAI-PMH.xsd">
        <responseDate>2012-11-30T21:09:09Z</responseDate>
        <request verb="GetRecord" metadataPrefix="oai_dc" identifier="oai:ojs.www.rdm.uff.br:article/43">http://
        <GetRecord>
                <record>
                        <header>
                                <identifier>oai:ojs.www.rdm.uff.br:article/43</identifier>
                                <datestamp>2012-02-07T15:03:15Z</datestamp>
                                <setSpec>rdm:SDB</setSpec>
                        </header>
                        <metadata>
<oai_dc:dc
        xmlns:oai_dc="http://www.openarchives.org/OAI/2.0/oai_dc/"
        xmlns:dc="http://purl.org/dc/elements/1.1/"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:schemaLocation="http://www.openarchives.org/OAI/2.0/oai_dc/
        http://www.openarchives.org/OAI/2.0/oai_dc.xsd">
        <dc:title xml:lang="pt-BR">Direito Comunitário: O Apogeu do Velho Continente e o Longo Caminho da Améric
        <dc:creator>de Mendonça Fróes, Rodrigo Dias Rodrigues</dc:creator>
        <dc:subject xml:lang="pt-BR">Direito Internacional Publico</dc:subject>
        <dc:subject xml:lang="pt-BR">Direito comunitário, Direito Internacional</dc:subject>
        <dc:description xml:lang="pt-BR">O texto aborda a evolução do Direito Comunitário desde o seu surgimento
na América Latina.</dc:description>
        <dc:publisher xml:lang="pt-BR">Revista de Direito dos Monitores da Universidade Federal Fluminense</dc:p
        <dc:publisher xml:lang="en-US">Revista de Direito dos Monitores da Universidade Federal Fluminense (Moni
        <dc:contributor xml:lang="pt-BR"></dc:contributor>
        <dc:date>2009-09-07</dc:date>
        <dc:type xml:lang="en-US"></dc:type>
        <dc:type xml:lang="pt-BR"></dc:type>
        <dc:format>application/pdf</dc:format>
        <dc:identifier>http://www.rdm.uff.br/index.php/rdm/article/view/43</dc:identifier>
        <dc:source xml:lang="pt-BR">Revista de Direito dos Monitores da Universidade Federal Fluminense; n. 6 (2
        <dc:source xml:lang="en-US">Revista de Direito dos Monitores da Universidade Federal Fluminense (Monitor
        <dc:language>pt</dc:language>
        <dc:coverage xml:lang="pt-BR">America Latina</dc:coverage>
        <dc:coverage xml:lang="pt-BR">Histórica</dc:coverage>
        <dc:coverage xml:lang="pt-BR"></dc:coverage>
        <dc:rights>Todos os trabalhos enviados e publicados pela Revista devem ter suas autorias reconhecidas em
de trechos de qualquer tamanho é livre. Ao enviar o seu trabalho, o autor automaticamente abre mão de seus direi
direito de fazer pequenas alterações na formatação do texto.Para mais informações sobre a Licença que utilizamos
formalidades de Direitos Autorais.</dc:rights>
        <dc:rights>Every essay sent and published by the Review must have their authorship recognized in any uti
quotation of any piece, of any size is free. By submitting your essay, the author automatically waive any of it'
ourselves the right to make smal alterations to the formatting of the text. To further information, read the lic
formalities of the author law.</dc:rights>
</oai_dc:dc>
                        </metadata>
                </record>
        </GetRecord>
</OAI-PMH>
```

Figura 3.4: *Instância XML de resposta do verbo GetRecord*

- ***GetRecord*** É utilizado para recuperar registro de metadado individual em um repositório. Dois argumentos são requeridos, o *identifier* que é o identificador único de um item no repositório, e o *metadataPrefix* que especifica o formato dos metadados. Um registro só deve ser devolvido se o formato especificado pelo *metadataPrefix* pode ser divulgada a partir do item identificado pelo valor do argumento identificador. Os formatos de metadados e de um registro específico podem ser recuperados utilizando a requisição *ListMetadataFormats* (LAGOZE; SOMPEL, 2008).

- ***Identify*** Este verbo é utilizado para recuperar informação sobre um determinado repositório. Algumas das informações retornadas são exigidas como parte do *OAI-PMH*. Repositórios também podem utilizá-lo para informações descritivas adicionais. Não possui argumento.

- ***ListIdentifiers*** Trata-se de uma forma abreviada do verbo *ListRecords*, recuperando apenas os cabeçalhos ao invés de todo o registro. Possui cinco argumentos opcionais para permitir coleta seletiva. Os argumentos de limitações de datas *from*, que especifica um limite anterior e *until*, que especifica o limite superior. O argumento *Set*, que dá critérios específicos de conjunto para coleta seletiva. Possui também o argumento requerido *metadataPrefix*, que especifica o formato que devem ser retornados os *heads* (cabeçalhos). E por fim o *Resumption Token*, argumento exclusivo contendo o valor do fluxo de controle retornado anteriormente pelo verbo *ListIdentifiers* que emitiu uma lista incompleta. O uso do argumento *resumptionToken*, faz com que não seja necessário re-informar outros argumentos, como por exemplo o *metadataPrefix*. O uso do *resumptionToken* é retornado pelos *data providers* quanto uma requisição retornou muitos registros e precisou ser dividida.

- ***ListMetadataFormats*** Utilizado para recuperar o formato dos metadados disponíveis de um repositório. Possui um argumento opcional, *identifier* que pode restringir a requisição para o formato disponível para um item específico.

- ***ListRecords*** Este verbo é utilizado para colher registros de um repositório. Afim de promover a colheita seletiva, permite as mesmas opções de argumentos já expostas no verbo *ListIdentifiers*. Esta requisição traz como resultado os registros completos e não somente os cabeçalhos.

- ***ListSets*** A partir deste verbo é possível recuperar a estrutura do conjunto de um repositório. Muito útil para coleta seletiva. Possui apenas um argumento, exclusivo, que é o *resumptionToken*.

Os provedores de serviço *(services providers)* OAI-PMH devem utilizar os verbos acima destacados respeitando sua estruturação e seus argumentos para que se tenha uma coleta efetiva e sem retorno de erros. A figura 3.5 traz uma representação dos verbos OAI-PHM acompanhados de seus argumentos.

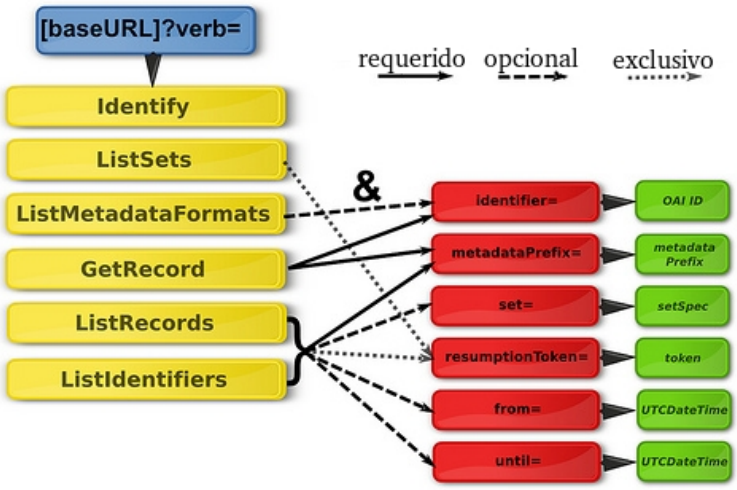

Figura 3.5: *Requisições (Verbos) OAI-PMH e seus argumentos (LAGOZE; SOMPEL, 2008).*

3.7 Considerações Finais

Este capítulo apresentou os conceitos fundamentais, discutindo a base teórica e as tecnologias utilizadas para a construção da proposta de solução defendida por esta dissertação. Foi possível perceber que vários conceitos precisam ser levados em consideração quando se pretende construir uma solução que alcance a interoperabilidade efetiva entre fontes de dados heterogêneas. Em especial para o uso do protocolo OAI-PMH como promotor de interoperabilidade, além de estratégias e ferramentas para armazenamento e coleta de metadados, é preciso levar em consideração regras de padronização dos dados que serão expostos por meio de metadados. O padrão *Dublin Core* atende essa expectativa para os seis verbos de coleta utilizados nas requisições OAI-PMH. O próximo capítulo abordará a metodologia desenvolvida para a construção da proposta de solução. Alguns aspectos tecnológicos discutidos neste capítulo serão novamente visualizados, desta vez dentro de uma perspectiva voltada para o desenvolvimento de um protótipo de software que permita a recuperação de informação de fontes de dados heterogêneas baseada nas palavras-chave resultantes da consulta do usuário.

CAPÍTULO 4

Metodologia Proposta para Solução

Este capítulo tem como objetivo discutir a metodologia utilizada para a construção da solução defendida por esta dissertação. Ressalte-se que um dos principais objetivos desta pesquisa é a implementação de um sistema (protótipo) que demonstre a viabilidade das ideias aqui propostas. O sistema implementado, aqui denominado ISSHS *(Information Search System from Heterogeneous Sources)*, visa promover interoperabilidade entre repositórios digitais, bancos de dados relacionais e outras fontes de dados por meio do protocolo OAI-PMH. Desenvolvido para plataforma Web, a interação com o usuário parte da premissa da simplicidade de comunicação, na qual uma consulta em linguagem natural (palavras-chaves) é submetida e um conjunto de termos é capturado e repassado ao sistema. Este realiza processamentos de limpeza, interpretação e agregação de novos termos (usando uma ontologia) à consulta formulada. Posteriormente, a partir de requisições HTTP emitidas para os provedores OAI-PMH nele registrados, o sistema coleta metadados de fontes de informação consideradas relevantes, de acordo com às palavras-chave da consulta, identificando aquelas que serão consultadas para obtenção do resultado final. Para que este protótipo alcance efetividade, outros sistemas auxiliares precisaram ser desenvolvidos. A Seção 4.1 aborda as tecnologias empregadas no desenvolvimento do protótipo e dos sistemas auxiliares. A Seção 4.2 discute a metodologia utilizada para o desenvolvimento dos sistemas auxiliares que são a base para o funcionamento do protótipo ISSHS. A Seção 4.3 apresentará a metodologia utilizada para o desenvolvimento do protótipo propriamente dito.

4.1 Tecnologias Empregadas

O protótipo do sistema ISSHS foi desenvolvido utilizando tecnologias livres, em especial Java 6. O IDE NetBeans 7.0.1 com o *Servlet Container* Glassfish (HEF-

FELFINGER, 2010). A comunicação entre os módulos é executada, principalmente, pela construção de *Servlets* HTTP, que extendem a classe ***HTTPServlet*** do pacote ***java.http.HttpServlet***. Para manutenção de registros de provedores e repositórios foi utilizado o SGBD MySQL, e para a execução das tarefas de conectividade utilizou-se o ***EclipseLink*** (JPA 2.0) com Hibernate. Para o processamento das palavras-chaves e busca de sinônimos, foi utilizado a JAWS - *Java API for WordNet Searching* e consequentemente, a base lexical WordNet (MILLER, 1995) e (LYLE, 2009). O web service DBot foi desenvolvido utilizando basicamente as mesmas tecnologias, com adição da METRO Web services 2.0 e Jersey 1.8 (RICHARDSON; RUBY, 2007). Para aplicação dos testes do protótipo, um provedor de dados OAI-PMH, baseado no software livre Phpoai2-1.8.0 (OLDENBURG, 2013), foi desenvolvido, e também se conecta a um banco de dados MySQL 5. Uma base de dados de testes contendo informações sobre livros foi adicionalmente desenvolvida, a qual é referenciada no provedor de dados como um recurso, sendo portanto descrita em metadados do padrão Dublin Core, com a referência ao web service DBot no elemento ***Identifier***.

4.2 Sistemas de Apoio ao ISSHS

Para o ciclo de funcionamento do ISSHS houve a necessidade de desenvolvimento de dois sistemas de apoio para oferecer suporte ao protótipo nas tarefas de coleta de metadados, controle de repositórios e acesso a fontes de dados heterogêneas. Estes sistemas são a base para o funcionamento do ISSHS, uma vez que a exposição dos metadados e o controle dos provedores de dados que serão conectados via requisições OAI-PMH dependem destes sistemas.

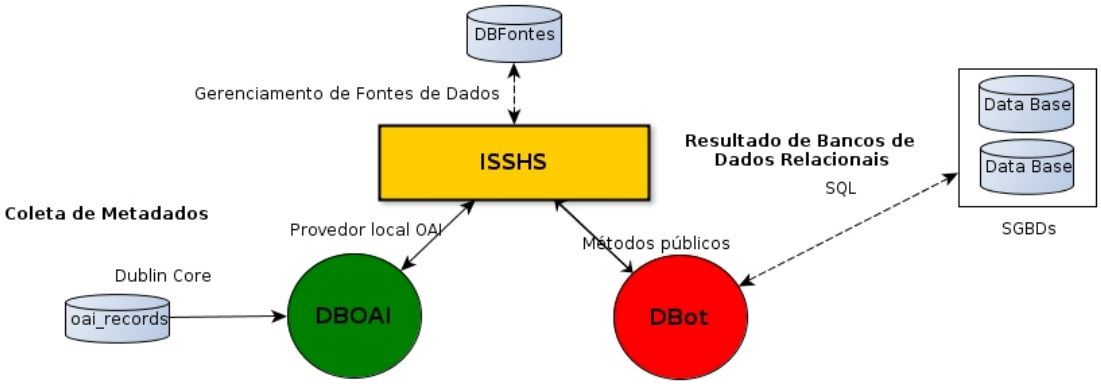

Figura 4.1: *Representação de Comunicação do Sistema ISSHS com os sistemas de apoio*

A figura 4.1 mostra uma representação de comunicação entre os sistemas de apoio aqui discutidos e o protótipo do Sistema ISSHS.

- **DBOAI**: fornece um *Data Provider* OAI-PMH. Isto é, permite que, a partir de requisições OAI-PMH, os arquivos *XML* resultantes da requisição sejam enviados para os requisitantes. O DBOAI conecta-se a uma base MySQL que contém a tabela *oai_records* responsável por armazenar os elementos do padrão *Dublin Core* que serão coletados, e outros atributos de controle. As requisições advindas dos seis verbos OAI-PMH com base nos registros contidos nesta tabela são devolvidas em forma de instâncias *XML* conforme já explicitado na Seção 3.6. A Subseção 4.2.1 detalha a metodologia utilizada na construção deste sistema.
- **web service DBot**: permite a conexão do ISSHS com múltiplas bases de dados e retorna, a partir de palavras-chave informadas via métodos públicos os registros presentes nos bancos de dados relacionais a que o web service estiver conectado, e que fizerem referência a estas palavras-chave. O web service retira dos ISSHS a necessidade de conhecimento da localização e conexão direta com os Sistemas Gerenciadores de Bancos de Dados que irão fornecer dados estruturados para o ISSHS apresentar aos usuários finais.

Outras discussões metodológicas sobre o desenvolvimento destas duas ferramentas são discorridas nas próximas subseções.

4.2.1 Sistema de Apoio DBOAI

Por ser este sistema auxiliar um *Data Provider* OAI-PMH, seu objetivo essencial é permitir respostas a requisições OAI-PMH advindas de um *Service Provider*(LAGOZE; SOMPEL, 2008). O DBOAI está preparado para responder qualquer um dos seis verbos previstos no protocolo OAI-PMH via requisições HTTP, estando conectado a uma base mySQL que contém a tabela *registros_oai*, composta dos elementos presentes no padrão *Dublin Core* conforme especificação já discutida na Seção 3.4.

Com a presença do DBOAI é possível criar repositórios próprios e há uma diferença entre sua concepção e aquelas dos *Data Providers* convencionais. Além de permitir que seja disponibilizada uma base de documentos digitais acessíveis a qualquer *Service Provider*, o DBOAI permite também que outras fontes de dados sejam descritas conforme abordagem de (KOWATA, 2011) e (SILVA et al., 2012), que sugerem uma tabela *TME* (Tabela de Metadados para Exposição) contendo vinte e três atributos. A tabela 4.1 apresenta resumidamente estes atributos. Essa construção se justifica no fato dos repositórios OAI-PMH não oferecerem suporte para fontes de dados advindas de SGBDS.

O sistema DBOAI utiliza estrutura similar à TME, uma vez que o padrão de metadados *Dublin Core* também é representado nesta abordagem, a fim de que as requisições OAI-PMH sejam respondidas corretamente, e atendendo ao que está previsto no protocolo conforme descrição anteriormente explicitada na seção 3.6.

A ideia inicial desta pesquisa era permitir o acesso a metadados de múltiplas e heterogêneas fontes de dados, como ocorre neste sistema. No entanto, percebeu-se a necessidade de avançar um nível a mais para o caso dos dados provenientes de Sistemas Gerenciadores de Bancos de Dados Relacionais, uma vez que há a necessidade de buscar informações além daquelas descritas nos metadados descritivos das fontes de dados.

Tabela 4.1: *Tabela para exposição de metadados (TME) (KOWATA, 2011) e (SILVA et al., 2012)*

serial	provider	url	email	oai_set	dispquery
dc_title	dc_creator	dc_subject	dc_description	dc_contributor	dc_publisher
dc_date	dc_type	dc_format	dc_identifier	dc_source	dc_language
dc_relation	dc_coverage	dc_rights	logindb	passwordb	–

Na abordagem utilizada por Kowata (KOWATA, 2011) e (SILVA et al., 2012), o atributo *dispquery* é usado como um *flag* que indica a disponibilização de conteúdo. Os atributos *logindb* e *passwordb* são elementos de controle para permitir que alguma aplicação externa tenha acesso ao banco de dados descrito nos metadados em questão. Outros atributos nesta mesma proposta possuem os mesmos nomes dos elementos originais do *Dublin Core*, mas na abordagem possuem objetivos diferentes. A tabela 4.1 traz uma descrição completa dos atributos propostos para TME (KOWATA, 2011).

Entretanto, na metodologia proposta por esta pesquisa, os elementos adicionais presentes em (KOWATA, 2011) e (SILVA et al., 2012) não são utilizados, uma vez que o controle de fontes de dados deve permitir que fontes heterogêneas sejam descritas, independentemente de serem sistemas gerenciadores de bancos de dados, ou dados semi-estruturados ou não estruturados. Desta forma o sistema DBOAI funciona sobre uma tabela de controle de fontes de dados denominada *oai_records* que apresenta a estrutura conforme descrição apresentada pela tabela 4.2.

A tabela (entidade) *oai_records* precisa conter as informações suficientes para a coleta de metadados em consonância com o protocolo OAI-PMH, que deve corresponder, no mínimo, a metadados *Dublin Core* (POWELL et al., 2007) de acordo o que foi discutido na Seção 3.4, que apresentou uma descrição detalhada dos elementos deste padrão de metadados, e em (LAGOZE; SOMPEL, 2008), onde encontram-se as descrições completas dos elementos do protocolo OAI-PMH.

Atributo	Descrição
serial	identificador único do registro
provider	nome do provedor de dados OAI-PMH.
url	endereço do provedor de dados OAI-PMH.
enterdate	data de criação do registro em formato *datestamp* (yyyy-mm-dd hh:mm:ss).
oai_identifier	identificador do registro OAI utilizado para refinamentos de coletas.
deleted	*flag* para exclusão lógica de registro.
dc_title	referente ao elemento *title*.
dc_creator	referente ao elemento *creator*.
dc_subject	referente ao elemento *subject*.
dc_description	referente ao elemento *description*.
dc_contributor	referente ao elemento *contributor*.
dc_publisher	referente ao elemento *publisher*.
dc_date	referente ao elemento *date*.
dc_type	referente ao elemento *type*.
dc_format	referente ao elemento *format*.
dc_identifier	referente ao elemento *identifier* .
dc_source	referente ao elemento *source*.
dc_language	referente ao elemento *language*.
dc_relation	referente ao elemento *relation*.
dc_coverage	referente ao elemento *coverage*.
dc_rights	referente ao elemento *rights*.

Tabela 4.2: *Tabela oai_records*

Levando em consideração as características de heterogeneidade propostas por esta pesquisa, o sistema DBOAI fornecerá condições para o ISSHS acessar múltiplos repositórios e fontes heterogêneas que serão descritas nos metadados armazenados no banco de dados *oai_records* por ele mantido. Para o perfeito funcionamento do Sistema, há dois módulos responsáveis pelas tarefas de armazenamento e transferência de dados.

A figura 4.2 apresenta o esquema de comunicação entre os módulos e a perspectiva de funcionamento de suas tarefas. Um usuário administrador se faz necessário para a manutenção dos dados. Isto é, cadastramento dos registros referentes aos objetos digitais não estruturados ou semi-estruturados, como páginas da *Internet*, arquivos de texto e também dados estruturados advindos de bancos de dados relacionais.

O sistema DBOAI possui também um módulo provedor de dados (*data provider*) que funciona de forma automática e sem a interveniência direta de qualquer usuário. Este módulo é responsável por responder os verbos OAI-PMH que lhe são repassadas via requisições HTTP. Ou seja, este módulo devolve, em forma de instâncias *XML*,

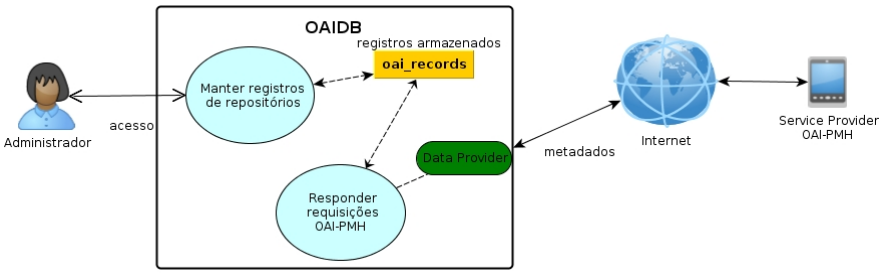

Figura 4.2: *Esquema de funcionamento do sistema DBOAI*

os pedidos dos coletores de metadados denominados provedores de serviço (*service providers*) OAI-PMH. O módulo provedor de dados do DBOAI foi desenvolvido com base no software livre PhpOAI2. (OLDENBURG, 2013)

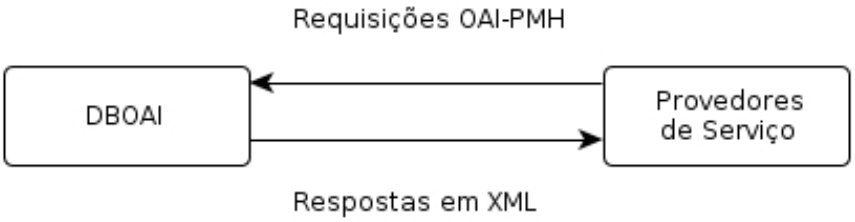

Figura 4.3: *Interação do DBOAI com provedores de serviço OAI-PMH. Inspirado em (LAGOZE; SOMPEL, 2008)*

Tecnologicamente, o sistema DBOAI é independente do ISSHS. Isto significa que ele não está restrito a requisições enviadas somente deste protótipo. O que na prática já se apresenta como uma contribuição desta pesquisa, pois é uma alternativa para que qualquer organização possa expor na *Web* algum repositório digital que poderá ser acessado por aplicações externas que destinarem para ele qualquer um dos verbos OAI-PMH. A Figura 4.3 traz uma representação dessa independência, onde qualquer provedor de serviços OAI-PMH pode realizar a tarefa de coleta de metadados executando qualquer um dos verbos OAI-PMH endereçados ao DBOAI. O código XML 4.1 mostra o retorno de uma requisição OAI-PMH enviada para o DBOAI.

4.2.1.1 Administração do DBOAI

A administração do DBOAI envolve o controle de repositórios digitais, isto é, verificação do conteúdo, acréscimo de metadados, licenciamento do registro, direitos de administração e controle de acesso e o armazenamento e associação dos metadados à localização física dos objetos nos repositórios. (KOWATA, 2011) O Administrador

Código XML 4.1 Parte do código XML de resposta à requisição OAI-PMH para o DBOAI.

```
<?xml version="1.0" encoding="UTF-8"?>
<OAI-PMH xmlns="http://www.openarchives.org/OAI/2.0/"
         xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
         xsi:schemaLocation="http://www.openarchives.org/OAI/2.0/ http://www.
             openarchives.org/OAI/2.0/OAI-PMH.xsd">
 <responseDate>2013-05-24T00:20:41Z</responseDate>
 <request verb="ListRecords" metadataPrefix="oai_dc">http://localhost/phpoai/
     oai2.php</request>
 <ListRecords>
  <record>
   <header>
    <identifier>oai:aName.org:localhost/phpoai</identifier>
    <datestamp>2013-01-01T23:20:00Z</datestamp>
   </header>
   <metadata>
     <oai_dc:dc
       xmlns:oai_dc="http://www.openarchives.org/OAI/2.0/oai_dc/"
       xmlns:dc="http://purl.org/dc/elements/1.1/"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xsi:schemaLocation="http://www.openarchives.org/OAI/2.0/oai_dc/http://
           www.openarchives.org/OAI/2.0/oai_dc.xsd">
  <dc:title>Consulta com Palavras-chave em repositorios heterogeneos</dc:title>
  <dc:creator>Filgueiras, Alison</dc:creator>
  <dc:subject>palavras-chave</dc:subject>
  <dc:subject>bancos de dados</dc:subject>
  <dc:subject></dc:subject>
  <dc:description>Este trabalho visa apresentar um estudo sobre consulta
```

deverá portanto manter o acervo digital dentro dos repositórios, que prevê a escolha de páginas da *Web*, PDFs, e outros documentos e organizá-los de acordo com um determinado critério que poderá ser uma linha temática ou um assunto. Deverá ainda manter os metadados dos objetos digitais de acordo com o padrão *Dublin Core* (MILLER, 1995). Os bancos de dados relacionais presentes no repositório serão descritos da mesma forma dos objetos digitais, que no caso da proposta de solução aqui discutida, terá como identificador o endereço do *middleware* DBot.

A figura 4.4 representa, de forma simplificada, as tarefas principais desenvolvidas pelo Administrador do DBOAI.

4.2.2 O web service DBot

Naturalmente, alguns repositórios digitais heterogêneos podem ser referenciados em provedores OAI-PMH, uma vez que os objetos digitais presentes nestes repositórios possuem uma descrição completa, baseada no padrão Dublin Core, retornada pelo elemento *Identifier* em uma requisição OAI-PMH. No entanto, no caso de registros referentes a bancos de dados relacionais, não há suporte específico dentro do protocolo.

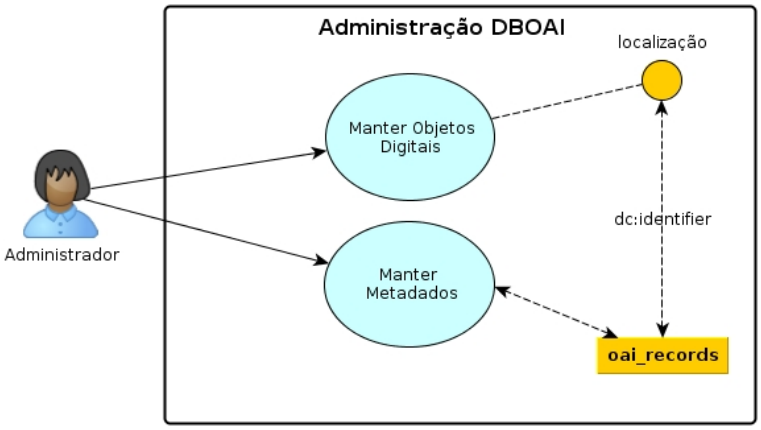

Figura 4.4: *Administração do sistema DBOAI*

Nesta lacuna, como proposta de solução, foi desenvolvido o *web service* DBot, que é um *middleware* responsável pela interlocução entre bancos de dados relacionais e um provedor de serviço, como o ISSHS. Sem o compromisso de discutir mais detalhadamente questões relacionadas à estruturação de *web services* e *middlewares*, é importante ressaltar que as tecnologias aqui envolvidas apresentam condições de garantir acessos na *Internet* com um certo nível de transparência de localização e com capacidade de permitir a interoperabilidade com aplicações clientes respeitando suas características heterogêneas (RICHARDSON; RUBY, 2007).

Por se tratar de um *web service*, o DBot pode ser localizado a partir de um identificador de recurso único (*URI - Uniform Resource Identifier*), e sua interface pública e contratos para utilização desta interface são descritos em XML (OLIVEIRA, 2010) e (W3C Working Group, 2004). A descrição dos serviços é realizada por WSDL (*Web Services Description Language*), permitindo que aplicações clientes e servidoras, fracamente acopladas, possam interagir por meio desses serviços. Esta arquitetura orientada a serviços permite que detalhes sejam ocultados, possibilitando que, independentemente de *hardware*, *software* ou linguagem de programação envolvidos, os serviços descritos possam ser utilizados. Há portanto uma capacidade de uso em conjunto de serviços para execução de operações complexas (W3C Working Group, 2004).

As requisições de serviços enviadas para o DBot são realizadas por meio de chamadas HTTP, sendo os dados transferidos em formato XML, encapsulados pelo protocolo SOAP (*Simple Object Access Protocol*). Permite-se assim que os dados sejam rotulados por meio de *tags* garantindo consistência e robustez à troca de dados (OLIVEIRA, 2010).

DBot possui um método público principal *getResultado* (ver 4.2.3.4) que permite conexão aos bancos de dados conectados ao repositório, sendo responsável pela submissão de consultas aos bancos de dados. Este modelo, além de permitir a busca em diferentes bancos de dados, retira a necessidade da aplicação cliente armazenar senhas ou mesmo ter conhecimento de esquemas.

Na versão atual do web service, o conjunto de termos de entrada, e posteriormente cada palavra-chave adicionada ao conjunto, é enviada para consulta ao banco de dados. A transformação do conjunto de palavras-chave em uma consulta ao banco de dados é executada pelo Analisador de Consulta, que deve ter acesso aos elementos do esquema para realizar a tarefa. O DBot possui outros métodos públicos que permitem que aplicações clientes recebam informações sobre os nomes de atributos e tabelas advindos do esquema dos bancos de dados relacionais conectados. O Método *getDataBaseName* (ver 4.2.3.1) permite que a aplicação cliente receba uma *string* com o nome do banco de dados. O Método *getTables* (ver 4.2.3.2) retorna uma lista das tabelas de um banco de dados, e o método *getTablaFields* (ver 4.2.3.3) retorna uma lista de atributos de uma tabela. O desenvolvimento destes métodos públicos adicionais visam uma evolução de funcionalidades do DBot, no entanto, para o serviço de recuperação de dados via palavras-chave, o ISSHS necessita fazer uso apenas do método *getResultado*.

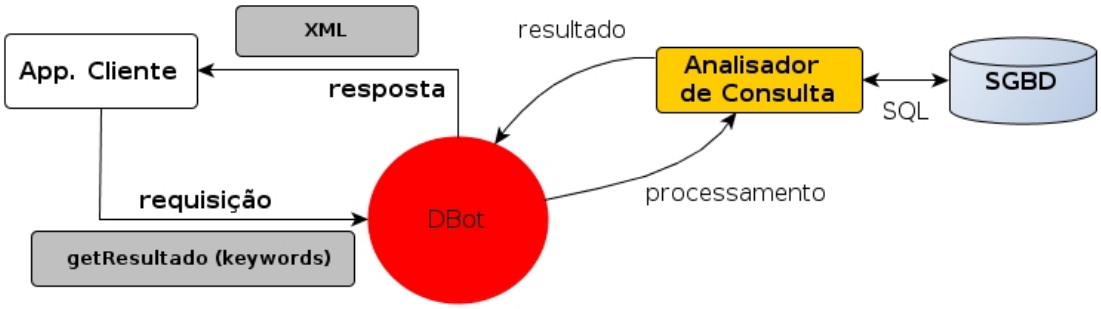

Figura 4.5: *Visão de Funcionamento do Middleware DBot*

Assim como o DBOAI, o *middleware* DBot apresenta características de independência em relação ao ISSHS, uma vez que, estando disponível na *Web*, permite que qualquer aplicação externa possa acessar os bancos de dados estruturados nele conectados, retornando os conteúdos presentes nesses bancos via métodos públicos. A Figura 4.5 traz uma representação do fluxo de requisição e resposta do DBot e a Figura 4.6 traz uma representação do modelo arquitetural do *web service* DBot que demonstra em alto nível sua perspectiva de funcionamento envolvendo aplicações externas e sistemas gerenciadores de bancos de dados nele conectados.

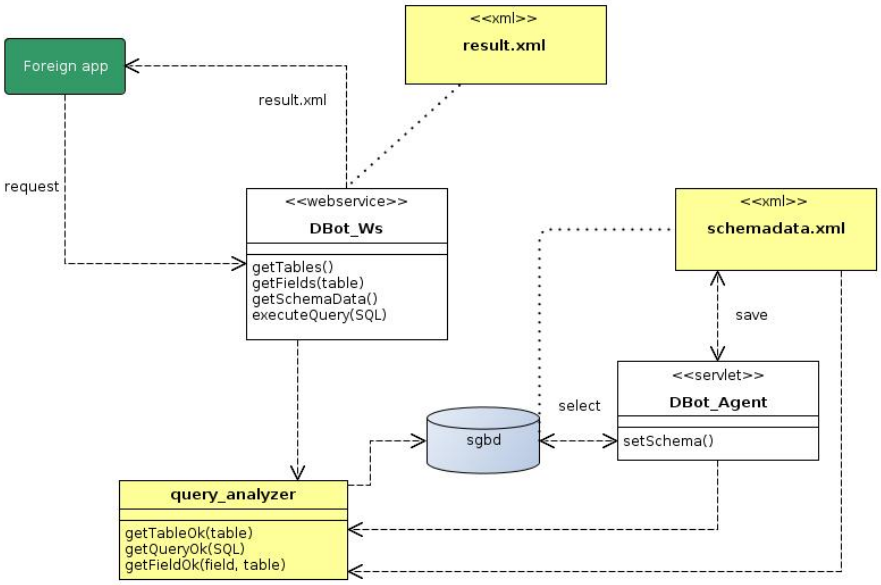

Figura 4.6: *Arquitetura do web service DBot*

Para corresponder a um nível mínimo de semântica, o DBot possui um módulo para o pré-processamento das consultas que serão transformadas em SQL. O **Analisador de Consulta** processa uma série de tarefas com intuito de retirar os termos supérfluos (aqui chamados *stop-words*) e adquirir sinônimos. onde cada um expressa um conceito distinto (MILLER, 1995). Uma classe léxica foi desenvolvida para realizar o trabalho de pré-processamento de consultas e é discutida com maior nível de detalhe na Subseção 4.4.

4.2.3 Interface Pública do DBot

Como já mencionado, os *web services* tornam-se acessíveis por meio de operações públicas. No caso do DBot, estas operações são getDataBaseName, getTables, getTablaFields e getResultado. Abaixo são listadas os parâmetros e características de resposta de cada uma destas operações.

4.2.3.1 GetDataBaseName

Este serviço retorna o nome do banco de dados conectado ao DBot. Para implementação de funcionalidades futuras, pode ser muito útil para que clientes possam escolher em quais bancos se conectar para uma possível busca seletiva de informações.

Argumentos

- nenhum

Retorno

- $< databasename > nomedobancodedados < /databasename >$

4.2.3.2 GetTables

Este serviço retorna um arquivo XML contendo a lista das tabelas presentes em um determinado do banco de dados conectado ao DBot. Espera-se que com este método, funcionalidades futuras possam ser implementadas para a coleta seletiva em uma determinada tabela.

Argumentos

- DataBaseName: *string* contendo nome do banco de dados

Retorno

- $< tablename > nomedatabela < /tablename >$

4.2.3.3 GetTableFields

O arquivo XML retornado por este serviço contém todos atributos presentes em uma tabela de um determinado do banco de dados conectado ao DBot. A expectativa é que com este método, funcionalidades futuras possam ser implementadas para a coleta seletiva em uma determinada tabela, tornando possível acessos a determinados atributos destas tabelas.

Argumentos

- DataBaseName: *string* contendo nome do banco de dados
- TableName: *string* contendo nome da tabela

Retorno

- $< tablename > nomedatabela < /tablename >$
- $< fieldname > nomedoatributo < /fieldname >$
- $< fieldtype > tipodoatributo < /fieldtype >$

4.2.3.4 GetResultado

O método público getResultado fornece o principal serviço do DBot, que é o retorno das consultas a partir de palavras-chave que lhe são passadas como parâmetro. Como já discutido anteriormente, estas palavras-chave de entrada obtém ganhos semânticos e são executadas como consultas SQL para as tabelas dos bancos de dados conectados no DBot. Em desenvolvimento futuro, espera-se que este método permita consultas seletivas. No entanto, na versão em que se encontra, todas as palavras-chave são consideradas como opção de consulta para todas as tabelas.

Argumentos

- palavras_chave: *string* contendo o conjunto de palavras-chave inicial.

Retorno

- $< tablename > nomedatabela < /tablename >$
- $< fieldname > nomedoatributo < /fieldname >$
- $< value > valordoatributoretornado < /value >$

4.2.4 Administração do DBot

Kowata (KOWATA, 2011) afirma que os serviços de administração de dados digitais envolvem verificação de conteúdo, tratamento de informações de metadados e de direitos autorais, enquanto em bancos de dados, além destas tarefas, a administração e controle de acesso se apresentam como tarefas adicionais. Isso é real quando o acesso aos bancos de dados dependem de tecnologias que permitem conexões diretas aos Sistemas Gerenciadores de Bancos de Dados Relacionais. No caso do DBot, os processos de administração não se limitam a políticas de segurança, uma vez que o *web service* permite apenas o acesso por meio de uma interface pública com base em XML e SOAP (OLIVEIRA, 2010). Não existe também qualquer trabalho de transformação e exposição de metadados dos bancos de dados conectados ao DBot, uma vez que as consultas são transformadas em linguagem SQL e submetidas a cada esquema destes bancos de dados. Desta forma, para o administrador do DBot restou a tarefa de escolher quais bancos de dados serão disponíveis para os métodos públicos e manter as configurações necessárias para as conexões nos arquivos de persistência, que utiliza o *Hibernate*. Os processos administrativos do DBot são representados na Figura 4.7.

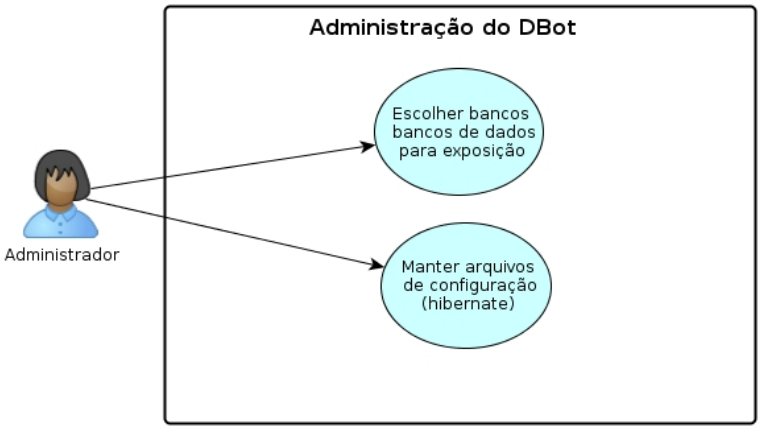

Figura 4.7: *Administração do web service DBot*

4.3 O Protótipo ISSHS - Information Search System from Heterogeneous Sources

O protótipo ISSHS objetiva oferecer uma alternativa para consultar palavras-chave em fontes de dados heterogêneas. Bibliotecas digitais, bancos de dados relacionais, ou outras fontes de informação disponíveis na *Internet*, como por exemplo uma página da *Web*, podem ser alcançadas pelo ISSHS, desde que descritas em metadados e expostas em repositórios OAI-PMH visíveis ao sistema.

O protótipo apresenta uma arquitetura modular conforme representações gráficas presentes nas Figuras 4.8 e 4.9 e recebe um conjunto de termos de um usuário final e após recuperar, de repositórios digitais e bancos de dados relacionais, informações relativas às palavras-chave resultantes da entrada, apresenta em forma de lista ordenada por relevância, para o usuário final, que pode ou não acessar a referência inequívoca trazida pelo sistema.

4.3.1 Arquitetura do Sistema

A arquitetura do ISSHS é constituída com base em uma abordagem modular de interdependências entre os módulos ou subsistemas. A Figura 4.9 apresenta uma representação básica desta arquitetura e a Figura 4.10 uma representação dos pacotes, que apresenta as principais classes e a comunicação entre elas. Ressalte-se, como já discutidos na Seção 4.2, que além dos módulos internos do ISSHS, esta proposta requer a integração com sistemas auxiliares, o que permitirá a efetividade de consulta com palavras-chave em fontes de dados heterogêneas, uma vez que o próprio protocolo OAI-PMH requer a divisão da coleta de metadados em provedores

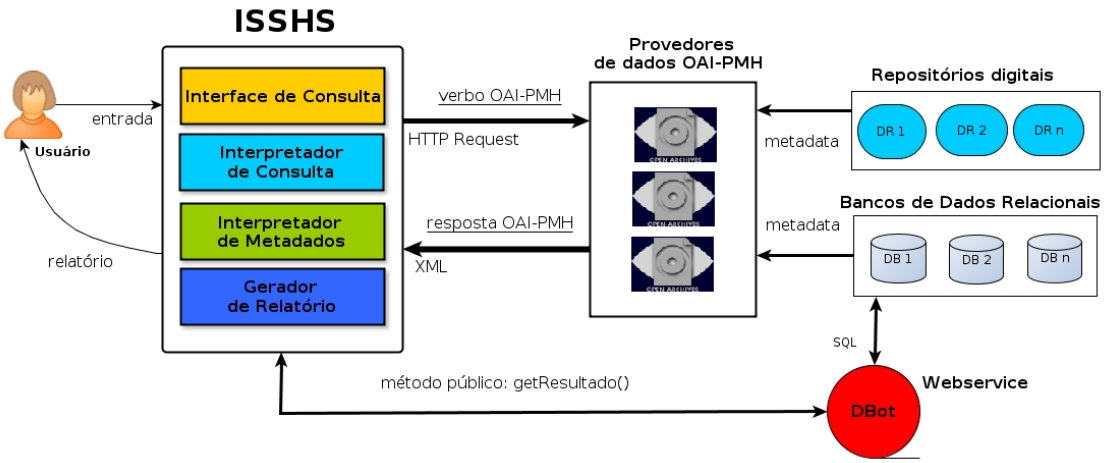

Figura 4.8: *Esquema de Funcionamento do Sistema ISSHS*

de serviço (*services providers*) e provedores de dados (*data providers*) (LAGOZE; SOMPEL, 2008), de acordo com as discussões anteriormente discorridas na Seção 3.6.

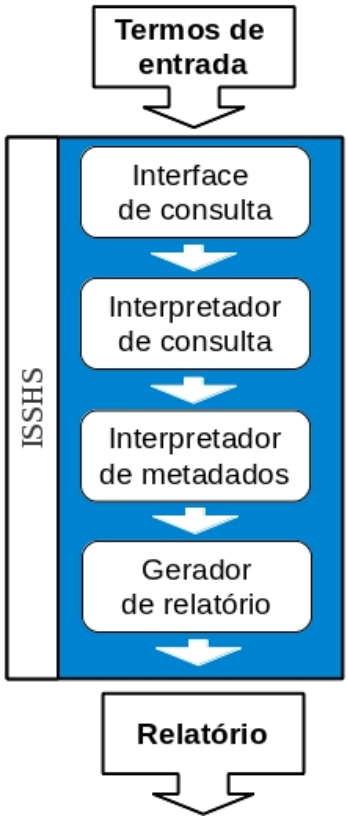

Figura 4.9: *Modelo Arquitetural do Sistema ISSHS*

4.3.2 Módulos do Sistema ISSHS

Por ser a arquitetura do ISSHS orientada à construção de módulos ou subsistemas, observa-se que há algumas vantagens em se estabelecer um *framework* para permitir a comunicação entre esses sistemas. Sommerville (SOMMERVILLE, 2006) afirma que em uma abordagem arquitetural orientada a objetos, os módulos são objetos com estado privado, tendo estes estados operações bem definidas. E tais módulos podem ser implementados como componentes ou processos sequenciais.

Esta estatégia permite, além da abstração em nível de domínio, a possibilidade de reaproveitamento de código para uma aplicação futura. As classes desenvolvidas para cada módulo são independentes e podem ser aplicadas em outro contexto. O ISSHS possui quatro módulos interdependentes, cada um com objetivos bem distintos em relação ao processo de consultar informações de fontes de dados heterogêneas utilizando palavras-chave. Como pode ser visto na representação presente na Figura 4.9, os módulos realizam os processamentos pertinentes e agregam valor à informação desde a entrada informada pelo usuário em forma de termos na Interface de Consulta, até a saída em forma de um relatório gerado pelo Gerador de Relatório, onde se espera um resultado que atende uma ordenação por relevância dos termos pesquisados. Os módulos mais internos do sistema são o Interpretador de Consulta e o Interpretador de Metadados.

Nas próximas subseções estes módulos serão melhores discutidos, e serão apresentados mais detalhes metodológicos e tecnológicos utilizados para seu desenolvimento.

4.3.2.1 Interface de Consulta

A interface de consulta compõe-se de um conjunto de simples de páginas *Web* que interagem diretamente com o usuário final, recebendo dele um conjunto de termos que podem estar em linguagem natural ou podem ser simplesmente palavras soltas em relação a algum assunto que o usuário desejar. Nos módulos subsequentes estes termos resultarão em um conjunto de palavras-chave, no entanto, todo o processamento que acontece após esta interação deve ser transparente ao usuário do sistema, ou seja, para o este usuário final do ISSHS, nada mais do que a Interface de Consulta lhe será visível em primeiro plano. Esta implementação da interface *Web* representa o único momento de intervenção do usuário com o sistema, uma vez que todo processamento posterior é executado automaticamente, isto é, sem interferência humana. Ao ser realizado todo processamento nos demais módulos do

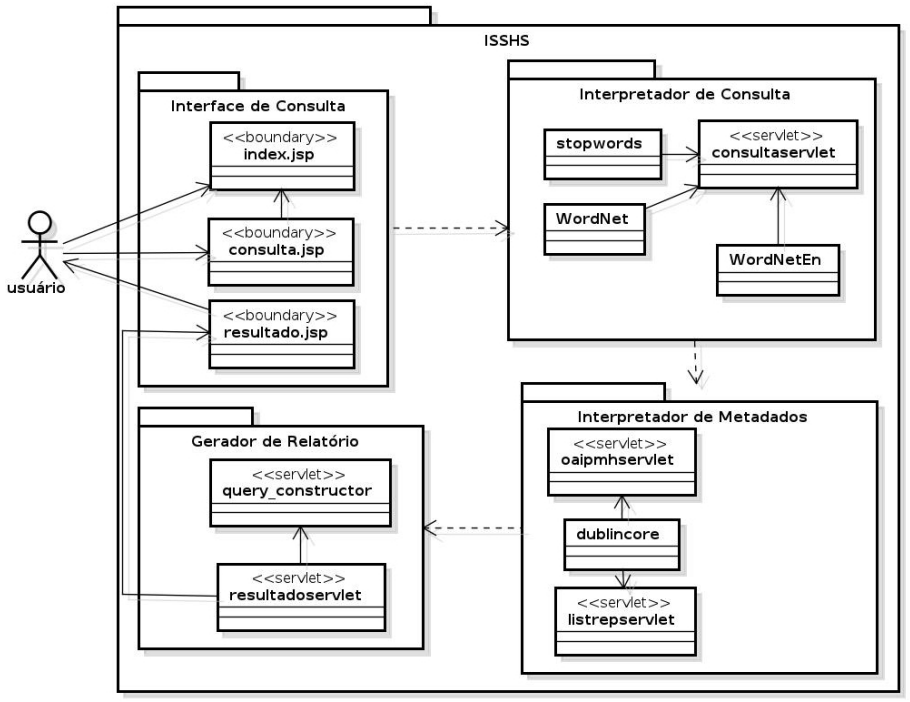

Figura 4.10: *Modelo de Pacotes do Sistema ISSHS*

ISSHS, um resultado é apresentado para o usuário final em forma de relatório que conterá informações organizadas baseadas em um critério de ordenação que objetiva apresentar prioritariamente aqueles registros de maior relevância para os termos informados inicialmente. Estes detalhes são discutidos de maneira mais aprofundada na Subseção 4.3.2.4

Trata-se de uma estratégia não tão peculiar, uma vez que vários motores de busca disponíveis na *Internet* desta mesma maneira o fazem, ao menos na interação inicial, como é o casso do *Google*, *Bing* e *Yahoo*. A figura 4.11 representa a perspectiva de transparência para o usuário final do ISSHS.

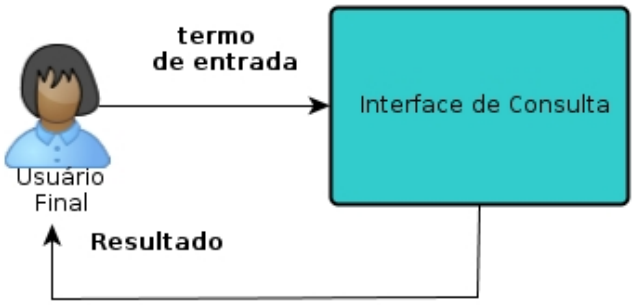

Figura 4.11: *Transparência para o usuário final*

A única verificação realizada pela Interface de Consulta é se o conjunto de termos de entrada está ou não vazio. Se o conjunto de termos não é vazio, independentemente

da maneira como foram informados os termos, estes serão encaminhados para o Interpretador de Consulta, para que ocorram as etapas posteriores do processamento que o Sistema executará dentro de cada módulo específico. A Figura 4.12 demonstra o fluxo deste processamento.

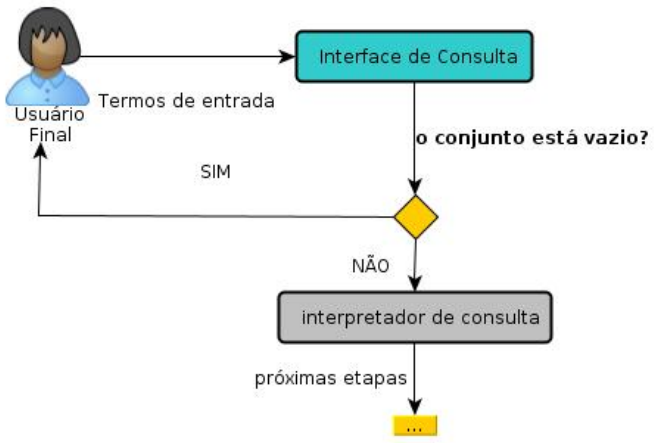

Figura 4.12: *Fluxo da interface de consulta do sistema ISSHS*

4.3.2.2 Interpretador de Consulta

A primeira grande tarefa do Interpretador de Consulta é a preparação dos dados textuais, que segundo Oliveira (OLIVEIRA, 2010) tem o objetivo de selecionar o que melhor expressa o conteúdo, reduzindo a dimensionalidade das informações textuais. Em suma, o módulo Interpretador de Consulta deve captar os termos lidos na interface de consulta, retirar as *stop-words* e realizar alguns tratamentos, para só depois repassá-los em forma de conjunto de palavras-chave para o Interpretador de Metadados. É nesta etapa que os termos de entrada recebem ganhos semânticos através do acréscimo de sinônimos e outros termos relevantes. O conjunto de termos gerado é submetido ao módulo que realizará a interpretação de metadados.

As atividades desempenhadas pelo Interpretador de Consulta passam inicialmente por três etapas (OLIVEIRA, 2010). A primeira é a geração de *tokens*, que é a subdivisão dos termos em um conjunto de palavras que sofrerão algumas modificações (como a transformação das letras em minúsculo) e a retirada de hífens, aspas e outros caracteres que poderão interferir na qualidade de uma consulta posterior. A segunda etapa passa pela remoção dos termos supérfluos, ditos *stop-words*, que são considerados irrelevantes para o sistema. A terceira etapa requer uma normalização morfológica (*stemming*). Um esquema apresentando estas três etapas pode ser visualizado na Figura 4.13.

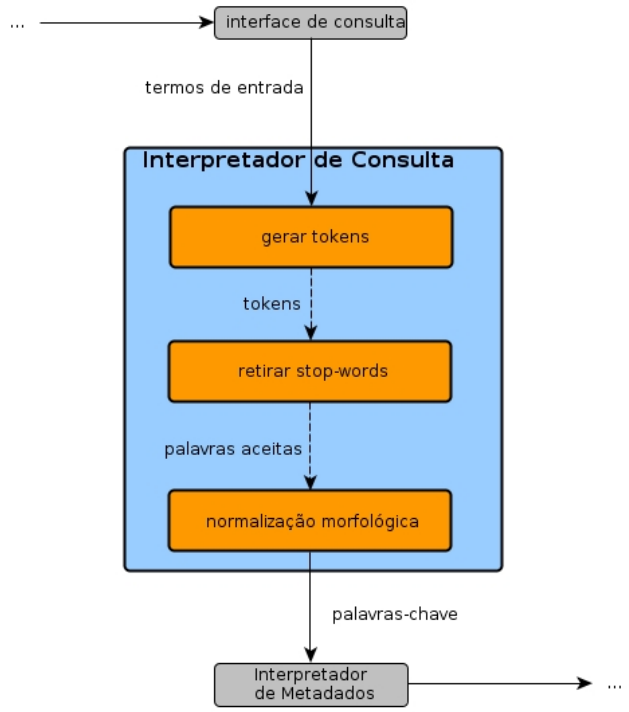

Figura 4.13: *Atividades do Interpretador de Consulta*

A normalização morfológica envolve a busca de sinônimos de cada termo presente no conjunto de palavras-chave repassado para o Interpretador de Consulta. Este tipo de inferência é ser realizado com a utilização do *WordNet*, que é um grande banco de dados léxico da língua inglesa constituido de substantivos, verbos, adjetivos e advérbios agrupados em conjuntos de sinônimos cognitivos, chamados de *(synsets)*, onde cada um expressa um conceito distinto (PRINCETON UNIVERSITY, 2010).

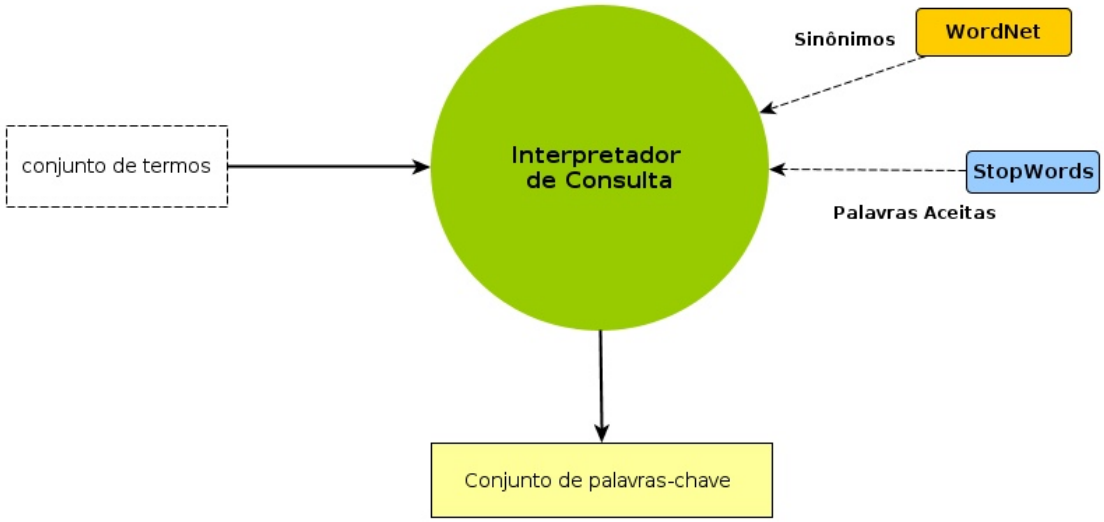

Figura 4.14: *Esquema do Interpretador de Consulta do ISSHS*

Tais tratamentos semânticos são discutidos com um maior nível de detalhe na Subseção 4.4, mas neste ponto é importante ressaltar que em termos de abstração finalística, os *tokens* repassados via Interface de Consulta são enviados para algoritmos que os transformarão em palavras-chave em que, dentre outras modificações, o novo conjunto receberá ganhos semânticos. A Figura 4.14 demonstra este esquema de funcionamento do Interpretador de Consulta. Ao final será repassado para o Interpretador de Metadados o conjunto de palavras-chave gerado pelo processamento de algoritmos pertencentes a este módulo.

Algoritmo 4.1: Algoritmo do Interpretador de Consulta

```
1 Entrada: C = conjunto de termos do usuário
2 Saida: P = conjunto de palavras-chave
3
4 se C não é vazio então
5   realize tratamentos semânticos;
6   envie resultado para Interpretador de Metadados;
7 fimse
```

4.3.2.3 Interpretador de Metadados

Este módulo é responsável por todas as operações que envolvem metadados. Ou seja, é dele a tarefa de coletar e realizar as devidas comparações com as palavras-chave afim de calcular a relevância para a consulta do usuário. A entrada para o Interpretador de Metadados é justamente o conjunto de palavras-chave resultante do processamento do módulo anterior, que realiza as requisições *OAI-PMH* para os repositórios disponíveis para aplicação. As repostas a estas requisições são feitas em *XML*, e o Interpretador de Metadados deve verificar se existem informações relevantes às palavras-chave resultantes dos processamentos anteriores. Isto é, o novo conjunto de palavras que obteve ganhos semânticos com a retirada de termos supérfluos e adicionamento de sinônimos, precisa ser confrontado com os metadados que estão coletados para gerar um conjunto de registros eleitos que será disponibilizado para o usuário final no módulo posterior. Tudo isso levando em consideração conjuntos heterogêneos de repositórios de dados. Visto que, nestes repositórios podem estar presentes documentos digitais (caso seja uma biblioteca digital ou estrutura análoga) ou dados estruturados em forma de tabelas, atributos, relacionamentos (SGBD), os metadados *Dublin Core* são nesta etapa imprescindíveis para a identificação e descrição do repositório, em especial o elemento *MetadataFormat*, que é responsável por informar se o registro obedece o padrão *Dublin Core*. Em caso positivo, trará o valor `oai_dc` na passagem de parâmetros ficando a requisição OAI-PMH com o formato `[requisição]&metadataPrefix=oai_dc`.

Como é nesta etapa que o sistema precisa se conectar com os provedores de dados *Data Providers* OAI-PMH, é necessário o uso de uma lista destes repositórios disponíveis, para que assim possam ser executadas as requisições e efetivado o processo de coleta. Para o armazenamento destes provedores foi desenvolvido um banco de dados denominado *DBFontes*, que possui uma tabela para armazenar as fontes de dados para o ISSHS realizar o processo de coleta de metadados. A manutenção do ***DBFontes*** é uma das tarefas administrativas para o funcionamento deste protótipo. A Figura 4.15 ilustra o funcionamento deste módulo.

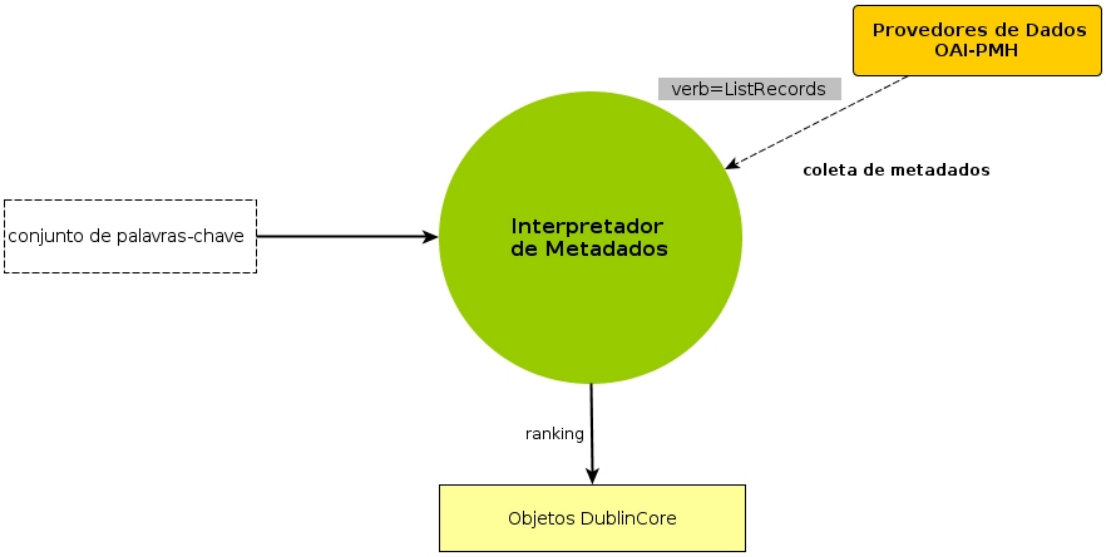

Figura 4.15: *Esquema do Interpretador de Metadados do ISSHS*

A tabela *repositorio*, do banco de dados ***DBFontes***, possui seis atributos, conforme demonstrado na figura 4.16. O atributo *id* deve possuir o identificador do repositório, que é uma *string* contendo essa identificação do provedor que deve ser registrada na Open Archives. O atributo *url* também deve ser do tipo texto e tem o intuito de armazenar o endereço do provedor que será acessado para o processo de coleta de metadados. Os atributos *cidade* e *pais* são informações administrativas para informar a localização geográfica do repositório. O atributo *idioma* deve conter o idioma principal em que estão descritos os metadados no repositório. Essas informações são úteis para uma possível coleta seletiva baseada no idioma de resposta. Por fim, o atributo *selecionado* é o *flag* que permite a habilitação ou desabilitação do repositório para o processo de coleta.

A manutenção de um banco de dados de repositórios OAI-PMH para o ***ISSHS*** é importante para garantir uma abrangência no que diz respeito ao crescimento incremental de fontes de dados que podem ser pesquisadas pelo sistema. Esta característica permite que o ***ISSHS*** interaja com múltiplos repositórios OAI-PMH, o

repositorio					
id	url	pais	cidade	idioma	selecionado
ojs.java2.unesp.br	http://ojs.unesp.br/index.php/revista_proex/oai	Brasil	São Paulo	PT	S
ojs2.periodicos.ufpb.br	http://periodicos.ufpb.br/ojs2/index.php/index/oai	Brasil	João Pessoa	PT	S
...	...	...	...	...	...

Figura 4.16: *Tabela repositorio do banco de dados DBFontes*

que pode ser visto como uma vantagem, pois coletando-se dados de diversas fontes, há um aumento no poder de coleta a partir das palavras-chave presentes nesta etapa. No entanto, vale ressaltar também que as conexões aos provedores de dados são feitas por meio de requisições *HTTP* através de *Servlets*, e por consequência, o número de repositórios pesquisados aumenta sensivelmente o tempo final de resposta da requisição.

A requisição HTTP submetida por este módulo tem o formato `endereco_do_provedor?verb=ListRecords&metadataPrefix=oai_dc` (LAGOZE; SOMPEL, 2008), que significa recuperar e coletar metadados dos provedores de dados endereçados no formato de metadados *Dublin Core*. A resposta de cada requisição é enviada em um arquivo XML contendo os metadados da fonte consultada. A tarefa do Interpretador de Metadados é analisar semanticamente os termos da consulta frente aos metadados obtidos, visando identificar se a fonte de informação contém dados relevantes para responder a consulta do usuário.

O conteúdo dos arquivos XML retornados é acondicionado dentro de objetos que permitem operações de comparação. Tais comparações levam em consideração todos os elementos presentes no conjunto de palavras-chave que foi repassado ao Interpretador de Metadados. Isto quer dizer que, antes mesmo de iniciar qualquer comparação, uma etapa de preparação dos dados coletados se faz necessária. O arquivo XML que é utilizado pelas requisições OAI-PMH, traz as informações em etiquetas que precisam ser identificadas e receber o tratamento necessário para as comparações futuras. O arquivo XML 4.2 apresenta o cabeçalho e parte da lista de registros de uma resposta a uma coleta realizada pela requisição OAI-PMH a partir do verbo *ListRecords*.

A etiqueta `<OAI-PMH>` traz as definições do esquema e do protocolo. Em `<responseDate>` é apresentada a data e horário da requisição. A *tag* `request` traz as informações da requisição, que são o verbo utilizado (*ListRecords*), o formato dos metadados (*metadataPrefix="oai_dc"*) e o endereço do provedor de dados (*data provider*) que antecede o fechamento da *tag* ($http://bdu.univates.br:8080/bdu\_oai/request$). No `<header>` é possível visualizar o idenfificador do registro (`<identifier>oai:www.univates.br/bdu:10737/16</identifier>`), a data de

Código XML 4.2 Cabeçalho resposta de uma requisição OAI-PMH (LAGOZE; SOMPEL, 2008).

```
<OAI-PMH xmlns="http://www.openarchives.org/OAI/2.0/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.openarchives.org/OAI/2.0/ http://www.openarchives.org/OAI/2.0/OAI-PMH.xsd">
<responseDate>2013-06-10T23:52:26Z</responseDate>
<request metadataPrefix="oai_dc" verb="ListRecords">http://bdu.univates.br:8080/bdu_oai/request</request>
<ListRecords>
<record>
<header>
<identifier>oai:www.univates.br/bdu:10737/16</identifier>
<datestamp>2011-12-01T11:26:42Z</datestamp>
<setSpec>hdl_10737_9</setSpec>
</header>
<metadata>
<oai_dc:dc xmlns:oai_dc="http://www.openarchives.org/OAI/2.0/oai_dc/" xmlns:dc="http://purl.org/dc/elements/1.1/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.openarchives.org/OAI/2.0/oai_dc/ http://www.openarchives.org/OAI/2.0/oai_dc.xsd">
<dc:title>
Avaliação econômica e energética de resíduos do setor avícola: análise do impacto ambiental
</dc:title>
<dc:creator>MOERSCHBAECHER, Oto Roberto</dc:creator>
<dc:subject>Embalagens</dc:subject>
```

criação do registro (`<datestamp>2011-12-01T11:26:42Z</datestamp>`) e o identificador do conjunto que este registro faz parte (`<setSpec>hdl_10737_9</setSpec>`).

Após obter a entrada de um arquivo XML de resposta da requisição OAI-PMH, neste caso, *ListRecords*, para acondicionar o conteúdo em um objeto que permita comparações entre *strings*, se faz necessário absorver os conteúdos presentes entre as *tags* `<metadata></metadata>`. Os elementos `Dublin Core` são notados por terem antes da descrição do elemento na etiqueta o prefixo "*dc* :", como pode ser visto na tag `<dc:subject>Embalagens</dc:subject>` do arquivo XML apresentado.

A estratégia de comparação adotada por este protótipo utiliza os elementos `<title>`, que traz o título referente ao recurso coletado, o elemento `<subject>` que apresenta as palavras-chave do recurso, e o elemento `<description>`, que possui o conteúdo do recurso em forma de resumo ou outra forma de descrição. A primeira comparação realizada pelo Interpretador de Metadados leva em consideração o conjunto de termos originais informados pelo usuário. As demais comparações são de acordo com cada ***token*** presente no conjunto de palavras-chave.

Em cada interação dos elementos do conjunto de palavras-chave que são confrontados com os metadados coletados é realizado um processo de adicionamento do registro em uma lista de objetos da classe *DublinCore*. Esta classe possui, além dos elementos do padrão *Dublin Core* tradicionais, um atributo ***ranking*** que recebe o valor calculado pelo algoritmo de processamento dos metadados. Este atributo

é definido pelo método ***setRanking***() da classe *DublinCore*. A primeira comparação realizada verifica se título (***title***), palavras-chave (***subject***) e resumo (***description***) do registro contém o conjunto de termos inicial informado pelo usuário. Em caso afirmativo para ***title***, instancia-se um novo objeto *(*DublinCore) com todos os elementos retornados na coleta e com peso 200 de *ranking*. Os procedimentos para ***subject*** e ***description*** são análogos a este, entanto, as pontuações serão 150 para o primeiro e 100 para o segundo.

As próximas comparações são realizadas com os itens do conjunto de palavras-chave diferenciando os termos aceitos originais daqueles que foram adicionados por serem sinônimos. Em caso de termos originais, o *ranking* somará 6 pontos por palavra-chave encontrada no título, 4 pontos se esta coincidência for nas palavras-chave do registro, e 2 pontos no resumo. Para os sinônimos estes valores são divididos na metade, simbolizando que os termos originais possuem o dobro de valor de cada item de sinônimo localizado, que terão 3, 2 e 1 ponto, respectivamente. O algoritmo 4.2 traz uma ilustração de parte do código em *Java* do processo comparação dos metadados e adicionamento dos pesos para o *ranking*, e o Algoritmo 4.3 e mesma representação em alto nível.

Os algoritmos de *ranking*, segundo (CROFT et al., 2009), são baseados implicitamente em um modelo de recuperação da informação. Eles devem estar relacionados à temática e à relevância com a pesquisa do usuário, sob pena de não funcionarem, em caso contrário. A despeito de tais afirmativas, o mecanismo de ***ranking*** aqui descrito não tem a intenção de apresentar como uma proposta para cálculo de relevância para termos de consulta, uma vez que não faz parte do contexto desta pesquisa teorizar, caracterizar ou explorar tais tecnologias. Nota-se no entanto, que este esquema de pontuação parece corresponder a um nível aceitável de efetividade para as consultas realizadas, mesmo não tendo sido contemplados testes exaustivos para este fim específico.

4.3.2.4 Gerador de Relatório

Por fim, o módulo de resultado, que aqui é denominado Gerador de Relatório, tem duas tarefas básicas. Primeiramente, é elaborado um relatório para ser apresentado ao usuário final com base em um *ranking* de relevância, considerando o conjunto de termos da consulta construído nos módulos anteriores e os metadados recuperados de cada fonte de informação consultada. Caso o recurso recuperado seja referente a um documento digital, o relatório deve permitir sua recuperação. Se o recurso retornado referir-se a um banco de dados, deverá ser permitida uma consulta

Algoritmo 4.2: Algoritmo para comparação do Interpretador de Metadados (em Java)

```
for (int k = 0; k < ldc.size(); k++) {
 DublinCore d = ldc.get(k);
 Util_resultado u = new Util_resultado();
 boolean achou = false;
 if (u.ContemIgnoreAcentos(d.getSubject(), kworiginal)) {

  pontuacao += 150;
  d.setSubject(d.getSubject().replaceAll(busca, "<b>" + kworiginal + "</b>"));
 }
 if (u.ContemIgnoreAcentos(d.getTitle(), kworiginal)) {

      pontuacao += 200;
      d.setTitle(d.getTitle().replaceAll(busca, "<i>" + kworiginal + "</i>"));
   }
   if (u.ContemIgnoreAcentos(d.getDescription(), kworiginal)) {

      pontuacao += 100;
      d.setDescription(d.getDescription().replaceAll(busca, "<b>" + kworiginal + "
          </b>"));
   }
  for (int j = 0; j < palavrasAceitas.size(); j++) {
    String string = palavrasAceitas.get(j).toLowerCase();
    if (d.getSubject().toLowerCase().contains(string)) {

     pontuacao += 4;
    }
    if (d.getTitle().toLowerCase().contains(string)) {

     pontuacao += 6;
    }
    if (d.getDescription().toLowerCase().contains(string)) {
     d.setDescription(d.getDescription().replaceAll("(?i)\\Q" + string + "\\E", "<b
         >" + string + "</b>"));
     pontuacao += 2;

    }
 }

 for (int j = 0; j < Sinonimos.size(); j++) {
  String string = SinonimosEnglish.get(j).toLowerCase();
  if (d.getSubject().toLowerCase().contains(string)){

   pontuacao += 2;
  }
  if (d.getTitle().toLowerCase().contains(string)) {

   pontuacao += 3;
  }
  if (d.getDescription().toLowerCase().contains(string)) {
   d.setDescription(d.getDescription().replaceAll("(?i)\\Q" + string + "\\E", "<b>"
       + string + "</b>"));
   pontuacao += 1;

  }
 }

 d.setRanking(d.getRanking() + pontuacao);
 if (pontuacao > 0) {
  ldcImpressao.add(d);
  registros_econtrados++;
  qtd = d.getOcorrencias(dc.getDescription(), kworiginal);
  pontuacao = 0;
 }
```

ao SGBD, pois o elemento Identifier do registro listado informará, ao invés de uma referência ao objeto digital, o endereço do web service DBot.

Croft (CROFT et al., 2009) afirma que o sucesso nas interações dos sistemas de busca na *Web* dependem drasticamente do usuário compreender os resultados. Dessa forma, nota-se a importância do relatório gerado pelo ISSHS ser simples,

Algoritmo 4.3: Algoritmo para comparação do Interpretador de Metadados em Alto Nível

```
1  //conjunto de termos = texto original informado pelo usuário
2  //palavras-chave = conjunto gerado pelo interpretador de consulta
3  Instancie DC como um novo objeto DublinCore;
4
5  DC recebe dados da coleta de metadados;
6
7  se <title> contém conjunto de termos então
8    adicione pontuação (200) para o ranking;
9  fimse
10
11 se <subject> contém conjunto de termos então
12   adicione pontuação (150) para o ranking;
13 fimse
14
15 se <description> contém conjunto de termos então
16   adicione pontuação (100) para o ranking;
17 fimse
18
19 para cada palavra-chave faça
20
21   se <title> contém palavra então
22     adicione pontuação (6) para o ranking;
23   fimse
24
25   se <subject> contém palavra então
26     adicione pontuação (4) para o ranking;
27   fimse
28
29   se <description> contém palavra então
30     adicione pontuação (2) para o ranking;
31   fimse
32
33   se palavra é sinônimo então pontuação da palavra é divida por (2);
34
35 fimpara
36 se DC possui pontuacao>0 então adicione DC à lista de relatório;
```

inteligível e com características que garantam ao usuário a continuidade das ações após a consulta realizada. Isto é, ele deve ter a clara noção do que está disponível e o que cada elemento retornado quer dizer e quais as opções para novas interações. Este usuário deve provavelmente ter o mínimo de contato com as páginas HTML, com *links* que o possibilite dar saltos em páginas e outros documentos de forma a efetivamente utilizar o conteúdo que é sua necessidade no momento da busca. Assim o relatório lhe é apresentado constando as informações dos registros retornados pelo OAI-PMH, sendo destacados os elementos importantes para a compreensão do conteúdo ali apresentado.

Os dados listados seguirão a ordenação pelo valor do peso (*ranking*) que é retornada pelo método ***getRanking*** de cada objeto ***DublinCore*** presente na lista de registros $java.util.List < DublinCore >$.

O modelo geral de funcionamento do módulo Gerador de Relatório pode ser visto na figura 4.18. A lista de registros de dados coletados de metadados de objetos digitais, bancos de dados relacionais e outras fontes de dados lhe são repassados em forma de objetos instanciados da classe ***DublinCore*** que já possuem um atributo específico para ordenação, retornado pelo método ***getRanking***. Esse peso

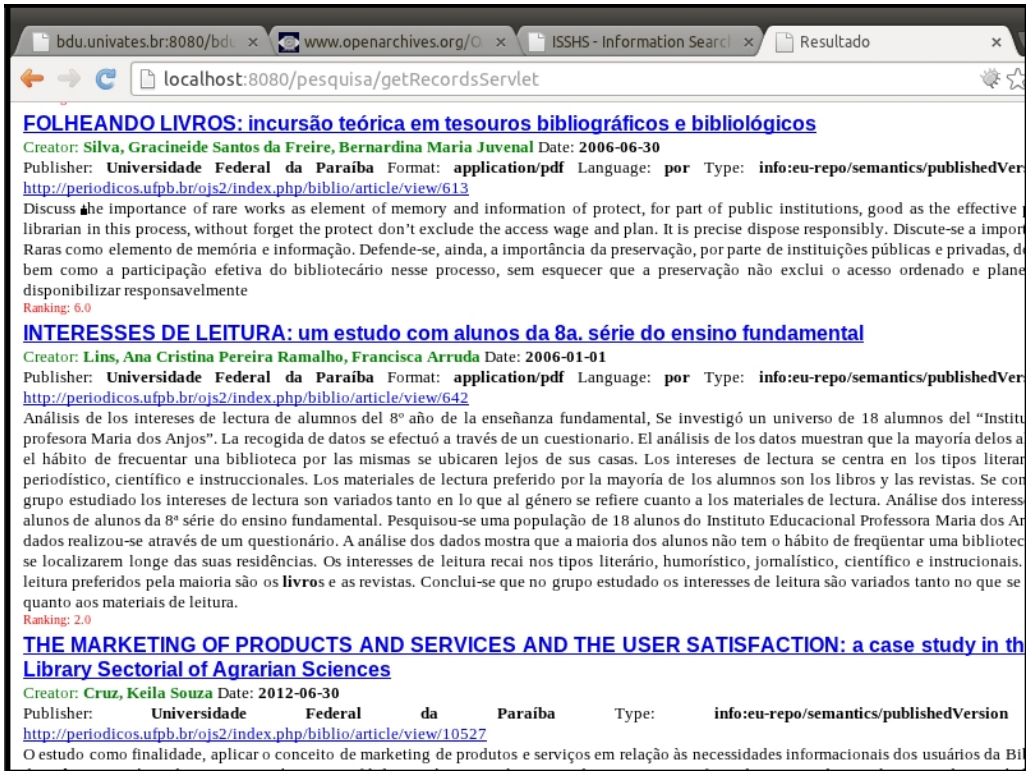

Figura 4.17: *Parte de saída gerada pelo Gerador de Relatório*

de relevância foi calculado na etapa anterior. O objeto traz também uma referência inequívoca e outras informações do padrão *Dublin Core* para serem apresentadas ao usuário final, que pode avançar com a consulta interagindo com um dos recursos apresentados. Esta etapa encerra o ciclo iniciado na Interface de Consulta.

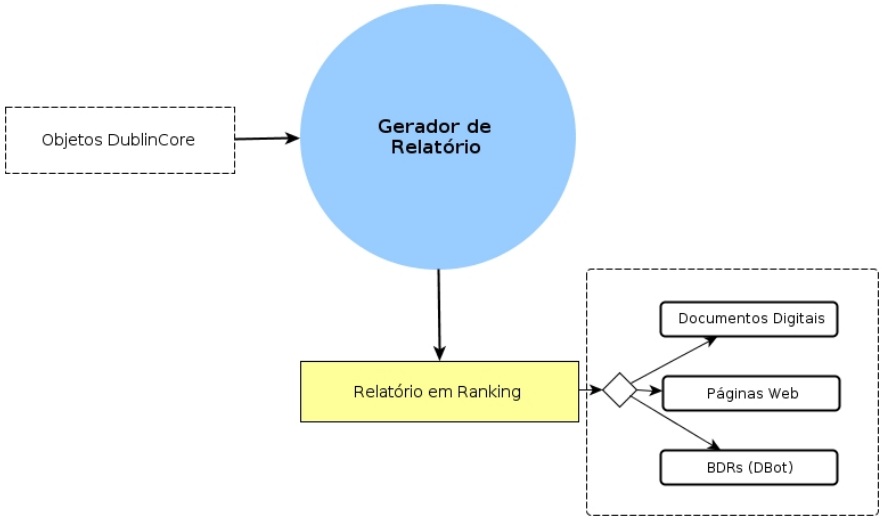

Figura 4.18: *Esquema do Gerador de Relatório*

4.3.2.5 Administração do ISSHS

Assim como as demais ferramentas que fornecem seviços na *Web*, o ISSHS requer uma administração. O papel do Administrador do Sistema ISSHS está associado às tarefas de escolha dos provedores de dados pelos quais será relizada a coleta de metadados, a manutenção dos termos supérfluos que serão retirados durante o processo de *stemming*, e a manutenção de termos da base de dados *Wordnet* mantida para a busca de sinônimos. Abaixo são descritas estas tarefas que estão simbolizadas na Figura 4.19.

- **Manter repositórios OAI-PMH**

  Descrição:
  Neste processo, o administrador insere, altera e exclui registros de repositórios OAI-PMH que estão armazenados na tabela *repositorio* no banco de dados *DBFontes*. Além disso, é possível que o administrador possa indicar quais repositórios estão disponíveis para serem pesquisados. A figura 4.16 mostra detalhes desta tabela.

- **Manter conjunto de termos supérfluos**

  Descrição:
  A ideia desta tarefa é permitir que o administrador possa adicionar e excluir termos da lista de palavras não significativas existentes no arquivo *stopwords.xml* do diretório do servidor do ISSHS. A partir desta lista de palavras, os processos léxicos retiram *stop-words*, que podem ser preposições, conjunções e outras palavras.

- **Manter base Wordnet**

  Descrição:
  A base do WordNet na língua inglesa não requer qualquer processo administrativo, uma vez que é instalada uma versão desta base no servidor e a partir de um pacote java baseado em JAWS (*Java API for Wordnet Searching*) (LYLE, 2009), são realizadas as consultas a esta base (MILLER, 1995). No entanto a base de dados gerada para o ***WordNetBR*** baseada em (SILVA, 2010) permite a inclusão de novos termos, sendo possível que as ações Administrador garantam uma evolução desta base. A conectividade do ***ISSHS*** com esta base de dados é realizada por meio do ***Hibernate***.

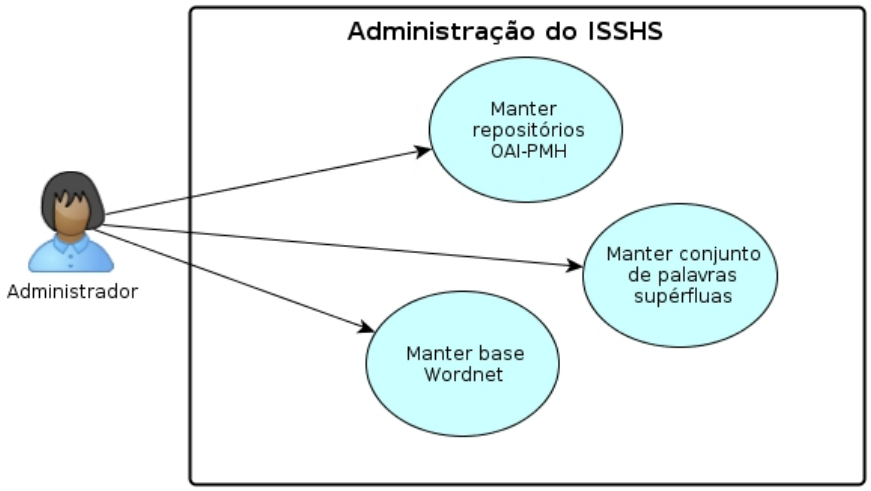

Figura 4.19: *Administração do ISSHS*

4.4 Tratamentos Semânticos

O protótipo do Sistema ISSHS e o *Middleware* DBot necessitam de tratamentos especiais que se referem a semântica dos termos que serão enviados para consulta tanto no nível dos metadados quanto nos dados relacionais, no caso da utilização do *Middleware*. Dessa maneira, a estratégia utilizada para proposta de solução levou em consideração os requisitos comuns nos dois casos, que resultou na construção de um pacote léxico que é utilizado em ambos.

Este pacote, que é demonstrado na Figura 4.20, contém três classes (*StopWord*, *WordNet* e *WordNetEn*) para garantir um nível mínimo de semântica, tanto no DBot, que possui um módulo para o pré-processamento das consultas que serão transformadas em SQL, quanto no caso do ISSHS, que possui um módulo equivalente para o processamento das palavras-chave que serão enviadas ao Interpretador de Metadados.

4.4.1 Processamento de Consulta

O **Analisador de Consulta** no DBot e **Interpretador de Consulta** do ISSHS, processam uma série de tarefas com intuito de retirar os termos supérfluos (aqui chamados *stop-words*) e adquirir sinônimos.

Estes sinônimos vão ser adquiridos com o uso de tecnologias auxiliares. Neste caso, foi utilizado o *WordNet*, que funciona como um grande banco de dados léxico da língua inglesa constituido de substantivos, verbos, adjetivos e advérbios agrupados

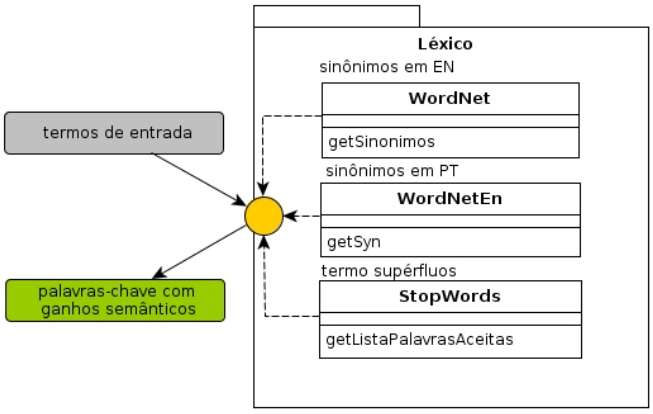

Figura 4.20: *Pacote Léxico*

em conjuntos de sinônimos cognitivos, chamados de *synsets*, onde cada um expressa um conceito distinto (MILLER, 1995).

Como o *Wordnet* é um banco de dados de palavras da língua inglesa, o processamento de palavras em Português não consegue efetivamente corresponder à busca de sinônimos para a grande maioria dos termos. Para isso, neste trabalho foi desenvolvido um banco de dados de termos em Português baseado em (SILVA, 2010) para retornar os sinônimos de palavras em português.

Dessa forma, o processamento de consulta inclui buscas dos termos em inglês e posteriormente em português. Ressalte-se que no escopo deste trabalho não é objetivo a discussão de aspectos léxicos muito menos de solução de características bilinguais para consultas com palavras-chave.

Este processamento, que tem claramente o intuito de adicionar ganhos semânticos, requer a utilização de vários algoritmos que estão presentes no pacote léxico. O Algoritmo 4.4, representa em alto nível o funcionamento destes algoritmos. O Algoritmo 4.5 apresenta o algoritmo, desenvolvido em java, que realiza o adicionamento de sinônimos em Inglês da classe *WordNetEn*.

Algoritmo 4.4: Algoritmo para processamento de consulta

```
1  -entrada- :T = conjunto de termos;
2  -saida- :P = conjunto de palavras-chave;
3  para (i de 1 até tamanho de T) faça
4    token = T[i];
5    se token não está em stop-words então
6      adicione token em P;
7      S=retornar sinônimos de token na classe Wordnet;
8      se S não é vazio então
9        adicione S em P;
10     fimse
11   fimse
12   retorne S;
13 fimpara
```

Algoritmo 4.5: Algoritmo para retornar Lista de Sinônimos (em Java)

```
1 Entrada: C = conjunto de palavras-chave;
2 Saida: Lista de Registros encontrados;
3 R = lista de repositórios OAI-PMH disponíveis;
4 para i de 1 até tamanho de r
5   envie ListRecord para R[i];
6
7 fimpara
```

O algoritmo responsável por esse processo de *stemming* primeiro retira as *stop-words* e executa uma agregação dos termos retornados pela consulta realizada na classe ***Wordnet***. A intenção desse processamento é, por exemplo, para um conjunto de entrada "livros de ciências", descartar a preposição "de" e agregar sinônimos de "livros" e "ciências". Uma saída possível para este processamento poderia ser o conjunto formado pelos dois termos selecionados e adicionados de outros termos como "revistas", "obras", "conhecimentos", etc. A figura 4.21 traz uma representação deste processamento.

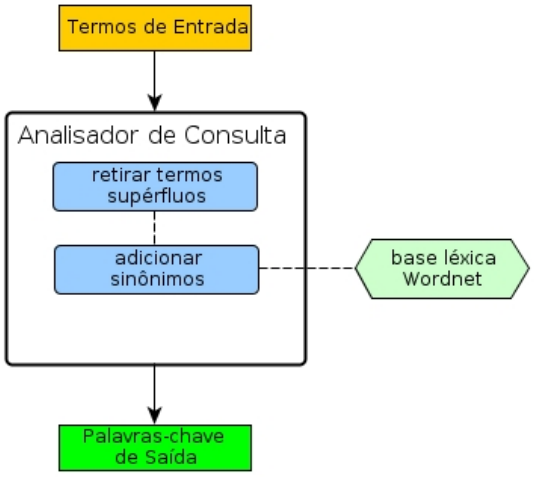

Figura 4.21: *Analisador de Consulta*

4.4.2 A Classe StopWord

A classe *StopWord* trabalha na retirada nos termos supérfluos e também de termos repetidos. Esta classe, que está associada a um arquivo *XML* contendo os termos não significativos é ilustrada no arquivo XML 4.3.

O algoritmo de verificação retira os *tokens* que forem iguais a qualquer termo presente no conjunto das *stop-words* retornadas do *XML*. Os administradores do ISSHS ou do DBot podem alterar essa lista de palavras, que é composta de preposições, artigos e outras classes de palavras.

Código XML 4.3 Arquivo XML contendo lista de termos supérfluos (*stop-words*).

```
<?xml version="1.0" encoding="UTF-8"?>
<stopwords>
  <stopword>de</stopword>
  <stopword>das</stopword>
  <stopword>dos</stopword>
  <stopword>das</stopword>
  <stopword>o</stopword>
  <stopword>as</stopword>
  <stopword>os</stopword>
  <stopword>as</stopword>
  <stopword>uma</stopword>
  <stopword>duma</stopword>
  <stopword>umas</stopword>
  <stopword>dumas</stopword>
  <stopword>pelo</stopword>
  <stopword>pelos</stopword>
  <stopword>pela</stopword>
  <stopword>pelas</stopword>
  <stopword>com</stopword>
  <stopword>nem</stopword>
  <stopword>qual</stopword>
  <stopword>como</stopword>
  <stopword>quanto</stopword>
</stopwords>
```

Algoritmo 4.6: Algoritmo para retirada de termos supérfluos (*stop-words*)

```
-Entrada: C = Conjunto de palavras
-Saida:   S = lista de palavras aceitas;
SW = conjunto de termos supérfluos;
para i de 1 até tamanho de c faça
  token=C[i];
  se token não está em SW então
    adicione token em S;
  fimse
fimpara
retorne S
```

Esta classe conta com dois métodos importantes. O *getListaSW*() que retorna a lista de *stop-words* de uma *string*, e o *getListaPalavrasAceitas*() que retorna as palavras significativas. Existem muitas outras técnicas de seleção de palavras, muitas inclusive utilizando abordagens com maior nível semântico. Este protótipo no entanto, limitou-se a simbolizar, de forma simples, o trabalho da seleção de termos significativos, uma vez que trata-se de um sistema modular que pode a qualquer momento agregar outras técnicas mais eficazes.

4.4.3 A Classe WordNetEn

A classe *WordNetEn* trabalha na proposta de processamento de *stemming*, que conecta-se com a base de dados do WordNet em Inglês, e retorna os sinônimos de cada palavra do conjunto gerado pelo método getListaPalavrasAceitas da Classe StopWord. O desenvolvimento desta Classe *WordNetEn* requer a instalação do Word-

Net no servidor que for utilizar o pacote léxico. Outro *framework* necessário para o processamento de sinônimos é o JAWS (*Java API for WordNet Searching*), que permite que outras aplicações java utilizem seus métodos para buscar essa relação sinonímia, que é a principal relação entre as palavras no WordNet (MILLER, 1995). A algoritmo 4.7 traz um exemplo de implementação de consulta ao WordNet por meio do *framework* JAWS. O método mais importante desta Classe, e responsável pelo retorno dos sinônimos, é o *getSyn*(), que tem como parâmetro uma *string* que contenha uma lista de palavras-chave separadas por vírgula. Outra versão deste método possui, ao invés de uma *string* uma lista do tipo *java.Util.List* contendo essas palavras-chave de entrada. A saída dessa Classe é sempre uma lista do tipo *java.util.List* < *String* >.

Algoritmo 4.7: Algoritmo de utilização do JAWS (LYLE, 2009)

```
1  NounSynset nounSynset;
2  NounSynset[] hyponyms;
3
4  WordNetDatabase database = WordNetDatabase.getFileInstance();
5  Synset[] synsets = database.getSynsets("fly", SynsetType.NOUN);
6  for (int i = 0; i < synsets.length; i++) {
7      nounSynset = (NounSynset)(synsets[i]);
8      hyponyms = nounSynset.getHyponyms();
9      System.err.println(nounSynset.getWordForms()[0] +
10                 ": " + nounSynset.getDefinition() + ") has " + hyponyms.length + "
                       hyponyms");
11 }
```

4.4.4 A Classe WordNet

Esta classe visa atender o mesmo processamento para a busca de sinônimos que agreguem valor semântico ao conjunto de palavras-chave para consulta do usuário. No entanto, diferencia-se da classe *WordNetEn*, pois não há a suporte para língua portuguesa no pacote Wordnet explorado pela classe *WordNetEn*. Dessa forma, para o contexto deste trabalho, foi desenvolvido um banco de dados relacional baseado no banco de dados textual de (SILVA, 2010) que continha uma lista sequencial de termos, sinônimos e antônimos correspondentes. O formato do arquivo segue o padrão apresentado no código XML 4.4. O processo de transformação deste arquivo textual em SQL para ser inserido na tabela *Triplos* lê cada linha do arquivo, seleciona o conteúdo de cada posição e constrói a consulta de inserção pertinente. O algoritmo utilizado para este processamento está descrito em 4.8.

Para o banco de dados relacional em MySQL, denominado *wnbr*, foi criada uma tabela *Triplos* para receber relação entre os termos que está expressa de forma textual. Os atributos desta tabela são descritos abaixo. A Figura 4.22 traz uma

Código XML 4.4 Formato do arquivo *WordnetBR* (SILVA, 2010).

```
[nome_da_categoria_dos_verbetes] verbete1 relacao verbete2
::::::::::::::::::::::::::::::::
Exemplo:
[Adjetivo] sobreexcedido SINONIMO_DE ultrapassado
```

Algoritmo 4.8: Algoritmo para tranformação da base textual *WordNetBR* para SQL.

```
entrada: E = arquivo texto com termos WNBR;
saida: S = arquivo com inserções SQL para tabela Triplo;

enquanto E.eof=falso faça
  localize categoria(do verbete);
  localize verbete1;
  localize relacao;
  localize verbete2;

  query=insert into Triplos(tipo,palavra1,relacao,palavra2)values(categoria,
      verbete1,relacao,verbete2);
  adicione query em S;
retorne S;
```

representação da tabela preenchida com dados de sinônimos e antônimos de verbos, adjetivos e substantivos.

- *tipo*: Tipo.
  Atributo textual que possui a descrição do tipo do elemento que pode ser Adjetivo, Advérbio ou Verbo.

- *relacao*: Relação.
  Atributo textual de tamanho 1 que indica *A* para antônimo e *S* para sinônimo na ligação entre duas palavras.

- *palavra*1: 1ª Palavra.
  Atributo textual de contendo a primeira palavra da associação.

- *palavra*2: 2ª Palavra.
  Atributo textual de contendo a segunda palavra da associação.

Algoritmo 4.9: *Script SQL* para consultar sinônimos em *Triplos*

```
select
  t.* from Triplos as t
where
  (palavra1='palavra' or palavra2='palavra')
  and relacao='S'";
```

triplos			
tipo	palavra1	relacao	palavra2
Adjetivo	à-toa	A	acerbo
Adjetivo	à-toa	S	descuidado
Substantivo	método	S	modo
Verbo	viver	A	falecer
...	...	...	...

Figura 4.22: *Tabela Triplos do banco de dados wnbr*

Verificar se há sinônimos e retorná-los é o processo realizado pelo método getSinonimos dessa classe. Este método possui um parâmetro na sua assinatura, que é uma *string* contendo o conjunto de palavras-chave de entrada. A mesma estratégia do método getSyn da Classe *WordNetEn* foi utilizada, e portanto, há uma outra implementação deste método em que o parâmetro é uma lista *java.util.List* < *String* >, que também possui o mesmo formato de resposta.

4.5 Considerações Finais

O objetivo deste Capítulo foi discutir a metodologia utilizada para a proposta de solução do problema de se recuperar informações de fontes de dados heterogêneas, dentre elas, bancos de dados relacionais, utilizando o protocolo OAI-PMH como interlocutor na interoperabilidade entre os repositórios. A solução aqui proposta passa pela construção de um protótipo, que na prática, é um provedor de serviços (*service provider*) OAI-PMH integrado com dois sistemas de apoio, e um provedor de dados OAI-PMH que permite a exposição de metadados de bancos de dados relacionais, objetos digitais e outros documentos (SILVA et al., 2012). O segundo, um *middleware* capaz de retornar, via métodos públicos, conteúdos de bancos de dados relacionais nele conectados. Discutiu-se também a administração dos sistemas desenvolvidos e os tratamentos semânticos que são importantes no processo de geração de palavras-chave a partir da consulta do usuário. Os resultados do funcionamento desta solução são visualizados no Capítulo seguinte, que discutirá outros detalhes de implementação e trará exemplos de interação realizadas pelos módulos do protótipo ISSHS.

CAPÍTULO 5

Exemplos de Interação e Resultados

No capítulo anterior foram discutidos aspectos da metodologia para proposta de solução do problema de se consultar, em fontes de dados heterogêneas, informações pertinentes a um conjunto de palavras-chave, que informados inicialmente pelo usuário, recebem ganhos semânticos. Foi proposto como solução o protótipo denonimado ISSHS, que utiliza de outras tecnologias para recuperar informações em fontes heterogêneas a partir do protocolo OAI-PMH. Neste Capítulo serão apresentados exemplos de interação nos diversos módulos do ISSHS, bem como serão discutidos e apresentados outros resultados retirados dessas e de outras interações. A Seção 5.1 apresentará de forma cronológica as entradas e saídas hipotéticas em uma interação com o ISSHS, com o intuito de demonstrar o estado da informação e como ela é transformada, dentro de cada módulo, deste o contato inicial com o usuário até a saída final em forma de relatório. A Seção 5.2 apresentará outras nuances e resultados obtidos desta interação. Por fim, a Seção 5.3 trará discussões e outras considerações finais deste capítulo.

5.1 Exemplos de Interação

A interação do usuário com sistema se dá pela interface de consulta. De maneira similar aos motores de busca tradicionais da *Internet*, espera-se que os usuários não despendam muito esforço além de informar uma combinação de termos que podem ser especificados em linguagem natural ou serem palavras soltas a respeito de um determinado tema a ser pesquisado. A Figura 5.1 mostra o início do processo de interação.

Neste exemplo, a suposição é que o usuário final tenha informado a sequência de palavras "`prices of books`", e o sistema ISSHS terá o comportamento descrito a seguir.

1. Interface de Consulta

A Interface de Consulta recebe a consulta do usuário "`prices of books`" e envia o conjunto de palavras para o interpretador de consulta. Isto significa que o usuário informou esta combinação de palavras no campo específico que, conforme pode ser visto na Figura 5.1 se apresenta para o usuário perguntando o que ele está procurando (*"what are you looking for?"*), e este usuário que inicia a tarefa de buscar as informações nas fontes de dados disponíveis para o ISSHS. Este processo de interação inicial com o usuário pretende garantir o máximo de simplicidade.

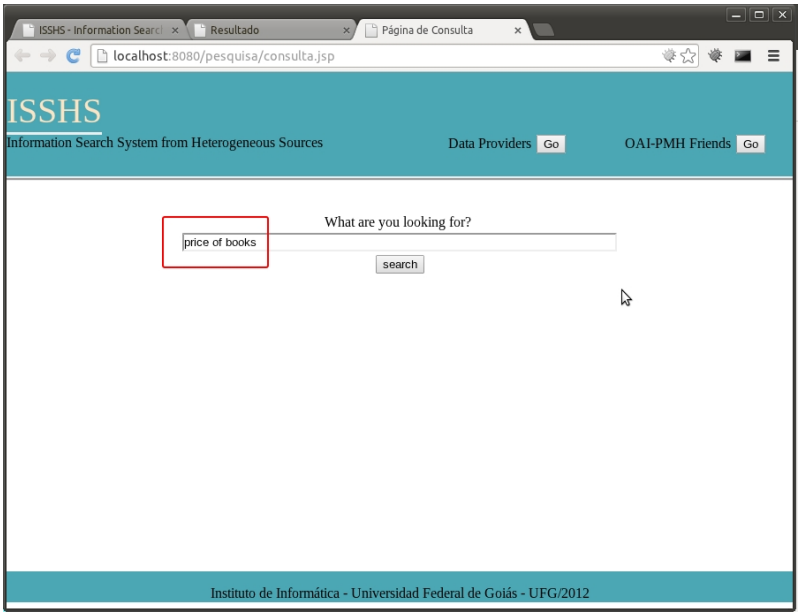

Figura 5.1: *Exemplo de interação: consulta do usuário.*

2. Interpretador de Consulta

O Interpretador de Consulta retira as palavras supérfluas e símbolos não significativos, e adiciona, a partir de consultas à base léxica Wordnet (MILLER, 1995), os sinônimos de cada um dos termos aceitos `[prices,book]`. O termo *of* é considerado supérfluo e portanto precisa ser descartado. O sistema leva em consideração também a palavra no singular, `prices` perde o "s", tornando-se `price` para a busca de sinônimos. Por estar em Inglês, a busca que o Interpretador de Consulta fará na base léxica baseada no *WordNetBr* (SILVA, 2010) não trará nenhum ganho semântico e nem afetará a consulta do usuário.

O conjunto de palavras-chave gerado agora tem a seguinte composição: `[monetary value, price, cost, terms, damage, toll, Price, Le-`

`ontyne Price, Mary Leontyne Price, book, volume, record, record book, script, playscript, ledger, leger, account book, book of account, rule book, Koran, Quran, al-Qur'an, Book, Bible, Christian Bible, Good Book, Holy Scripture, Holy Writ, Scripture, Word of God, Word]`.

Nota-se que a consulta à base WordNet retornou vários termos que incluem substantivos próprios. É notório ainda, que pela necessidade de se analisar cada palavra do conjunto inicial de palavras-chave, a quantidade de termos que comporão um novo conjunto de palavras-chave e seus sinônimos tem um aumento de tamanho significativo. Não há garantias, no entanto, que esses sinônimos façam uma referência precisa daquilo que o usuário deseja buscar. Como é o caso deste exemplo no que diz respeito ao termo `price`, que no contexto da consulta se referia a buscar preços de livros, e que o WordNet retornou como sinônimo `Leontyne Price`, cantora lírica americana. Outro caso visualizado no contexto desta consulta, é dos termos `book`, livro, que retornou como sinônimo a `Bible` (bíblia), livro sagrado do cristianismo. E consequentemente retornou também as palavras `Koran, Quran, al-Qur'an, Koran, Quran, al-Qur'an, Scripture, Word of God e Word` outros substantivos associados à escrituras religiosas. Este novo conjunto de palavras-chave é agora enviado para o interpretador de metadados

3. Interpretador de Metadados

De posse do novo conjunto de palavras-chave o interpretador de metadados envia o comando OAI-PMH *ListRecords* para os provedores de dados presentes na lista de repositórios do sistema. O comando *ListRecords* recupera cada registro em formato XML, e contendo em seu conteúdo, além das definições de esquema do protocolo OAI-PMH, os elementos do padrão *Dublin Core*. As requisições disparadas para cada repositório serão `[endereco_do_repositorio]?verb=ListRecords&metadataPrefix=oai_dc`. Uma das requisições será então `http://bdu.univates.br:8080/bdu_oai/request?verb=ListRecords&metadataPrefix=oai_dc` que retornou o XML contendo as informações de registros presentes neste repoistório. O código 5.1 ilustra parte deste arquivo XML que ao ser utilizado pelo sistema terá seus elementos comparados para verificar ser este registro é relevante ou não para a consulta do usuário. Em caso positivo, um objeto do tipo *Dublin Core* é instanciado e adicionado a uma lista *java.util.List<DublinCore>* para ser repassada ao Gerador de Relatório.

Tais repositórios de metadados fazem referência a objetos digitais e bancos de dados disponíveis. Para cada arquivo XML retornado, verifica-se os elementos

Código XML 5.1 Código XML contendo registro da coleta de metadados.

```
<OAI-PMH xmlns="http://www.openarchives.org/OAI/2.0/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.openarchives.org/OAI/2.0/ http://www.openarchives.org/OAI/2.0/OAI-PMH.xsd">
<responseDate>2013-06-09T23:22:13Z</responseDate>
<request metadataPrefix="oai_dc" verb="ListRecords">http://bdu.univates.br:8080/bdu_oai/request</request>
<ListRecords>
<record>
<header>
<identifier>oai:www.univates.br/bdu:10737/16</identifier>
<datestamp>2011-12-01T11:26:42Z</datestamp>
<setSpec>hdl_10737_9</setSpec>
</header>
<metadata>
<oai_dc:dc xmlns:oai_dc="http://www.openarchives.org/OAI/2.0/oai_dc/" xmlns:dc="http://purl.org/dc/elements/1.1/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.openarchives.org/OAI/2.0/oai_dc/ http://www.openarchives.org/OAI/2.0/oai_dc.xsd">
<dc:title>
Avaliação econômica e energética de resíduos do setor avícola: análise do impacto ambiental
</dc:title>
<dc:creator>MOERSCHBAECHER, Oto Roberto</dc:creator>
<dc:subject>Embalagens</dc:subject>
<dc:subject>Avicultura</dc:subject>
<dc:subject>Reciclagem</dc:subject>
<dc:subject>Polimero</dc:subject>
<dc:description>
Com o objetivo de dimensionar o volume de resíduos sólidos gerados na avicultura, e possíveis formas de aproveitamento por processos de reciclagem ou fontes alternativas energéticas, realizou-se a análise do ciclo de vida (ACV) das embalagens utilizadas nesta atividade pecuária. Este trabalho foi desenvolvido em parceria com a empresa Sadia S.A. que forneceu o grupo de produtos que utiliza em tratamentos terapêuticos e de higienização das ...
</dc:description>
<dc:date>2008-11-20T13:24:45Z</dc:date>
<dc:date>2008-11-20T13:24:45Z</dc:date>
<dc:date>2008-11-20</dc:date>
<dc:date>2008-06-13</dc:date>
<dc:type>Dissertation</dc:type>
<dc:identifier>http://hdl.handle.net/10737/16</dc:identifier>
<dc:language>pt_BR</dc:language>
</oai_dc:dc>
</metadata>
</record>
```

`<title>`, `<subject>` e `<description>`, comparando seus conteúdos ao conjunto original de termos "`prices of book`", e posteriormente a cada item do conjunto de palavras-chave gerado. Para a aplicação deste teste, considerou-se apenas a primeira interação com o provedor de dados, sendo descartados os demais registros alancansáveis pelo uso do atributo `ResumptionToken`. Nos provedores de dados pesquisados foram coletados dados de 949 registros e em 294 houve resultados satisfatórios. Realizando as requisições para os provedores de dados em separado tem-se as seguintes informações.

- **Requisição**: http://periodicos.ufpb.br/ojs2/index.php/index/oai
  Registros retornados: 100

Registros relevantes: 21
Tempo de Resposta: 7498Ms
Tamanho do *Buffer*:352361 caracteres

- **Requisição**: http://revcom2.portcom.intercom.org.br/index.php/mediajornalismo/oai
  Registros retornados: 0
  Registros relevantes: 0
  Tempo de Resposta: 928Ms
  Tamanho do *Buffer*:1154 caracteres

- **Requisição**: http://www.worldsciencepublisher.org/journals/index.php/ACMA/oai
  Registros retornados: 59
  Registros relevantes: 11
  Tempo de Resposta: 4449ms
  Tamanho do *Buffer*:263779 caracteres

- **Requisição**: http://www.icesi.edu.co/revistas/index.php/revista_cs/oai
  Registros retornados: 100
  Registros relevantes: 5
  Tempo de Resposta: 9013ms
  Tamanho do *Buffer*:346485 caracteres

- **Requisição**: http://sedici.unlp.edu.ar/phpoai/oai2.php
  Registros retornados: 100
  Registros relevantes: 12
  Tempo de Resposta: 4472ms
  Tamanho do *Buffer*:127128 caracteres

- **Requisição**: http://www.jistem.fea.usp.br/index.php/jistem/oai
  Registros retornados: 319
  Registros relevantes: 228
  Tempo de Resposta: 26656ms
  Tamanho do *Buffer*:2158786 caracteres

- **Requisição**: http://www2.pucpr.br/reol/index.php/PA/oai/
  Registros retornados: 350
  Registros relevantes: 14
  Tempo de Resposta: 32423ms
  Tamanho do *Buffer*:1003801 caracteres

- **Requisição**: http://ukdw.ac.id/journal-theo/index.php/wacana/oai
  Registros retornados: 15
  Registros relevantes: 1
  Tempo de Resposta: 9399ms
  Tamanho do *Buffer*:33242 caracteres

- **Requisição**: http://localhost/phpoai/oai2.php
  Registros retornados: 6
  Registros relevantes: 2
  Tempo de Resposta: 314ms
  Tamanho do *Buffer*:12053 caracteres

4. O Gerador de Relatório

O Gerador de Relatório deve ordenar uma lista destes resultados encontrados contendo os principais elementos do metadados que descrevem o recurso indicado pelo registro, incluindo um link de acesso. Os registros são listados em ranking que pontua a relevância da fonte de informação com base na consulta original 'prices of book' e nos metadados título, assunto e descrição, seguindo a mesma análise para o conjunto de palavras-chave gerado. Um dos registros apresentados se refere ao objeto digital http://periodicos.ufpb.br/ojs2/index.php/biblio/article/view/9622, que tem como título `'Bibliotecário na Formação dos Leitores em Potencial'`. Um registro referente a um bancos de dados é encontrado, ocasionando uma consulta que será executada pelo *web service* DBot, que apresenta o resultado em forma de tabela, contendo informações do livro `'Metadata and Its Applications in Digital Library'`.

A Figura 5.2 traz parte da lista ordenada apresentada pelo gerador de relatório para o usuário final.

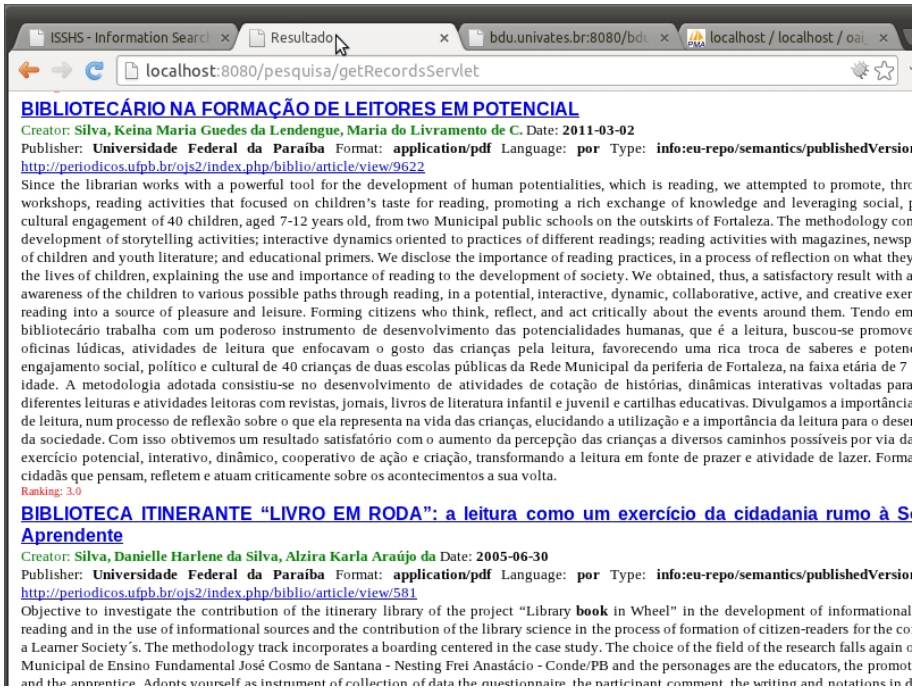

Figura 5.2: *Exemplo de interação: relatório de saída para o usuário.*

5.2 Resultados

Até a data de conclusão deste trabalho, a *Open Archives Initiative* possuía uma lista de 1962 (mil novecentos e sessenta e dois) provedores de dados registrados. Uma requisição HTTP para o endereço `http://www.openarchives.org/Register/ListFriends` retorna um arquivo XML contendo o `ID` e o endereço de cada um dos servidores. Existem ainda provedores de dados OAI-PMH que estão em funcionamento, mas não estão nesta lista de servidores registrados. Como exemplo deste caso, cita-se a *url* `http://www.teses.usp.br/cgi-bin/oai.pl`.

A *Open Archives Initiative* disponibiliza também regras para o registro de provedores de dados. O acesso de um novo provedor é realizado pelo endereço `http://www.openarchives.org/data/registerasprovider.html` e tem a intenção de tornar público e acessível tais provedores, garantir a conformidade desses provedores com o protocolo OAI-PMH, e fornecer um meio para a OAI monitorar a utilização do protocolo, para ações estratégicas futuras (LAGOZE; SOMPEL, 2008). Estes provedores, após serem registrados, serão adicionados à lista de *repositórios amigos* que pode ser requisitada e retorna via linguagem XML.

O código 5.2 apresenta parte deste XML onde `<BaseURLs number='1962'>` representa que existem no momento da requisição 1962 (mil novecentos e sessenta e dois) repositórios registrados. Cada `<BaseURL></BaseURL>` traz em seu interior o

Código XML 5.2 Parte do arquivo XML de retorno do OAI/ListFriends.

```
<BaseURLs number="1962">
<baseURL id="ojs.icesi.edu.co">
http://www.icesi.edu.co/revistas/index.php/revista_cs/oai
</baseURL>
<baseURL id="sedici.unlp.edu.ar">http://sedici.unlp.edu.ar/phpoai/oai2.php</
    baseURL>
<baseURL id="digitalscholarship.uml.edu">http://digitalscholarship.uml.edu/cgi/
    oai2.cgi</baseURL>
<baseURL>
http://bibliotecadigital.icesi.edu.co/biblioteca_digital-oai/request
</baseURL>
<baseURL id="www.qualitative-research.net">
http://www.qualitative-research.net/index.php/fqs/oai
</baseURL>
<baseURL id="biblioteca.aranjuez.es">http://biblioteca.aranjuez.es/i18n/oai/oai
    .cmd</baseURL>
<baseURL>http://dspace.ubu.es:8080/tesis-oai/request</baseURL>
<baseURL id="ojs.delhi.spaceweb.ru">http://psychopharmacology.ru/index.php/PPBN
    /oai</baseURL>
<baseURL>http://dcoll.brandeis.edu/dspace-oai/request</baseURL>
<baseURL id="ojs.geolib.geo.auth.gr">
http://geolib.geo.auth.gr/digeo/index.php/index/oai
</baseURL>
<baseURL>
http://sucra.saitama-u.ac.jp/modules/xoonips/oai.php
</baseURL>
<baseURL>http://84.79.19.79:8080/dspace-oai/request</baseURL>
<baseURL id="caltechcdstr.library.caltech.edu">http://caltechcdstr.library.
    caltech.edu/perl/oai2</baseURL>
<baseURL id="digitalcommons.law.msu.edu">http://digitalcommons.law.msu.edu/do/
    oai/</baseURL>
<baseURL id="place.asburyseminary.edu">http://place.asburyseminary.edu/do/oai/<
    /baseURL>
<baseURL>http://dspace.library.iitb.ac.in/oai/request</baseURL>
<baseURL id="ojs.cinej.pitt.edu">http://cinej.pitt.edu/ojs/index.php/cinej/oai<
    /baseURL>
<baseURL>
http://bfheecsucv.oai.alejandria.biz/cgi-win/be_oai.exe
</baseURL>
<baseURL>http://bibliogeo.ing.ucv.ve/cgi-win/be_oai.exe</baseURL>
<baseURL>http://hermes.bbtk.ull.es/pandora/cgi-bin/oai.exe</baseURL>
<baseURL>
```

`id` e a `url` de cada repositório para que um provedor de serviço (*service provider*) OAI-PMH possa realizar as requisições OAI-PMH.

Para os testes desenvolvidos neste trabalho foram escolhidos nove provedores de dados OAI-PMH disponíveis na Internet. Por se tratarem de requisições HTTP para diferentes servidores em diferentes localizações geográficas, o tempo de resposta varia sensivelmente de um para o outro. O verbo OAI-PMH enviado para cada provedor foi `ListRecords?metadataPrefix=oai_dc`. Este comando retorna um arquivo XML contendo os registros, no formato Dublin Core, presentes no provedor de dados. A Tabela 5.1 apresenta o resultado das requisições aos provedores de dados, informando o endereço do provedor, a quantidade de caracteres do arquivo XML de resposta, o tempo para execução da requisição, em milissegundos, e por fim, a abreviação do país onde se encontra o provedor.

Tabela 5.1: *Resultados de requisições OAI-PMH aos provedores de dados.*

Endereço	Tamanho	Tempo	Local
http://periodicos.ufpb.br/ojs2/index.php/index/oai	347212	11113	BR
http://revcom2.portcom.intercom.org.br/index.php/mediajornalismo/oai	1154	6212	BR
http://www.worldsciencepublisher.org/journals/index.php/ACMA/oai	391	1525	IS
http://www.icesi.edu.co/revistas/index.php/revista_cs/oai	352613	8520	CO
http://sedici.unlp.edu.ar/phpoai/oai2.php	119728	5509	AR
http://www.jistem.fea.usp.br/index.php/jistem/oai	342298	6428	BR
http://www2.pucpr.br/reol/index.php/PA/oai/	1003801	32423	BR
http://ukdw.ac.id/journal-theo/index.php/wacana/oai	33955	10949	ID
http://localhost/phpoai/oai2.php	11643	1135	-

A consulta exemplo submetida aos nove repositórios, sendo um local com referência a documentos digitais e um banco de dados relacional, e oito acessíveis na Internet. Foram encontrados 949 registros relevantes em um tempo um pouco superior a um minuto. Estes nove repositórios foram escolhidos de forma aleatória, contemplando servidores nacionais e internacionais. O provedor de dados local possui objetos digitais e um banco de dados relacional. O banco de dados relacional, que também foi exposto no formato *Dublin Core* faz referência ao *Middleware* DBot conectado ao sistema gerenciador de banco de dados que possui as informações descritas nos metadados coletados no processamento da consulta.

5.3 Considerações Finais

Este Capítulo teve como principal objetivo a discussão de resultados da validação da solução proposta nesta dissertação, além de apresentar exemplos de interação, que se referem a estados de funcionamento do protótipo desenvolvido e aqui apresentado como proposta de solução para o problema de se recuperar informações de fontes de dados heterogêneas, tendo como protocolo de interoperabilidade o OAI-PMH. Apresentou os aspectos referentes aos estados assumidos em cada módulo presente na solução proposta, fazendo referência ao comportamento do sistema em cada um desses estados. Apresentou ainda, resultados e outras informações pertinentes aos repositórios de dados que estão registrados na *Open Arches Initiative* em especial aqueles que estão na lista de repositórios amigos (*ListFriends* do OAI-PMH (LAGOZE; SOMPEL, 2008). As discussões apresentadas neste capítulo são base para as discussões finais desta Dissertação que será apresentado em seguida no Capítulo 6 onde tem-se a oportunidade de descorrer sobre as contribuições, trabalhos futuros, e a produção bibliográfica relacionada a esta pesquisa.

CAPÍTULO 6

Conclusão

O objetivo maior deste trabalho foi propor uma solução para a recuperação de informação de diferentes fontes de dados utilizando o protocolo OAI-PMH como agente interlocutor. Como proposta de solução, foi apresentado o sistema ISSHS que visa permitir justamente que um usuário possa, a partir de conjunto de termos, buscar informações em diferentes repositórios na *Web* em um processo complementamente transparente e sem utilização de esforços extras. Foram apresentados também, como mecanismos auxilires, o desenvolvimento de outros sistemas que trabalhando de forma integrada com o ISSHS, torna possível a recuperação de informação de fontes de dados heterogêneas em uma perspectiva da *Internet* que envolve documentos digitais não estruturados e dados estruturados de bancos de dados relacionais. Estes sistemas auxiliares e o protótipo ISSHS foram discutidos no Capítulo 4.

Desta forma, apresentam-se abaixo as principais contribuições trazidas por este trabalho. Seguidamente serão apresentadas as perspectivas de trabalhos futuros e da produção bibliográfica desenvolvida nesta pesquisa.

6.1 Contribuições

O acesso a informações armazenadas em múltiplas fontes de dados é algo bastante explorado nas comunidades científicas, em especial na Ciência da Computação. Há interesses de várias organizações e pessoas neste assunto e muitas iniciativas e tecnologias, como as desenvolvidas pela a comunidade *Open Archives Inicitive*, tem sido propostas para contribuir neste cenário de troca de informações. De fato, percebe-se um número cada vez maior de recursos, sobretudo aqueles relacionados a bibliotecas digitais, que tem permitido que a comunidade de usuários do mundo inteiro possa buscar informações utilizando-se da *Web*. Entretanto, existe uma grande quantidade de informação invisível em camadas da *Web* que é inalcançável pelos sistemas de

busca. Entre essas informações estão aquelas contidas em sistemas de bancos de dados relacionais, inacessíveis pelos mecanismos tradicionais de busca por palavras-chaves, por necessitarem da execução de processos adicionais entre o usuário e a informação propriamente dita.

Este estudo explorou tecnologias de integração entre diferentes fontes de dados, visando apresentar uma solução para interoperabilidade entre bancos de dados relacionais e bibliotecas digitais utilizando-se do protocolo OAI-PMH. Desta forma, defende-se que a proposta traz algumas contribuições para solucionar o problema de consultas por palavra-chave em fontes de dados heterogêneas.

Nos testes realizados:

1. Recuperação em múltiplas fontes de dados

O protótipo ISSHS mostrou se capaz de recuperar dados de repositórios digitais e também de bancos de dados relacionais, além de permitir a coleta de metadados em múltiplos provedores OAI-PMH.

2. Recuperação em fontes de dados heterogêneas

O *web service* desenvolvido (DBot) mostrou ser capaz de permitir que diferentes fontes de dados, hospedadas em servidores de bancos de dados relacionais, possam ser acessadas por aplicações externas a partir de métodos públicos, sem que tais aplicações tenham conhecimento das estruturas presentes nestes bancos de dados, como o nome de campos ou tabelas.

3. Coleta de metadados de bancos de dados relacionais

Os bancos de dados relacionais, expostos via OAI-PMH a partir de metadados descritivos, foram efetivamente explorados após a coleta destes metadados realizada por um provedor de serviço OAI-PMH, sem a necessidade de permissões especiais, conforme sugestões de outros trabalhos, e nem conhecimento prévio de esquemas de dados ou mesmo o uso de SQL.

4. Exposição de metadados de dados heterogêneos

Os sistemas de apoio desenvolvidos nessa proposta permitiram que bancos de dados relacionais e objetos digitais pudessem descritos e expostos no padrão de metadados *Dublin Core*, sendo possível sua utilização como uma provedor de dados OAI-PMH de fácil utilização.

6.2 Trabalhos Futuros

Alguns aspectos desta proposta necessitam de maior atenção e desenvolvimento futuro. A implementação de critérios semânticos, para análise de consultas frente aos metadados de cada fonte de informação, precisa ser melhor explorada, preferencialmente com o uso de ontologias. Outra possibilidade é aprimorar a recuperação da informação com palavras-chave em bancos de dados relacionais, com técnicas análogas aquelas desenvolvidas em Bergamaschi et al (BERGAMASCHI et al., 2010) e (BERGAMASCHI et al., 2010), para que os resultados das requisições do *web service* DBot tenham ganhos na coleta seletiva, e com outras possibilidades de busca, como o compartilhamento de esquemas. Outra oportunidade vislumbrada é que as consultas aos provedores de dados possam ser executadas em paralelo, com o intuito de melhorar o tempo de resposta destas consultas..

6.3 Produção Bibliográfica

O desenvolvimento desta pesquisa culminou na produção de uma revisão sistemática e de dois artigos científicos que abordaram questões relativas ao tema explorado por esta dissertação. A seguir é apresentada uma lista sucinta com a descrição destes trabalhos.

IX - Congresso de Pesquisa, Extensão e Ensino da Universidade Federal de Goiás - Conpeex/2012 O artigo intitulado "Promovendo Interoperabilidade entre Repositórios Digitais e Bancos de Dados Relacionais por meio do protocolo OAI-PMH" (FILGUEIRAS; SILVA, 2012) apresentado no IX-Conpeex abordou a integração entre bibliotecas digitais e bancos de dados relacionais pelo protocolo OAI-PMH. Analisou questões teóricas de tecnologias de sistemas bibliotecas digitais e bancos de dados relacionais, o padrão *Dublin Core* de metadados e o protocolo OAI-PMH. Apresentou como estudo de caso o problema de integração de informação do sistema ***Pegasus*** da Pró-Reitoria de Extensão, Cultura e Assuntos Estudantis da Universidade Estadual de Goiás, sugerindo cinco passos para a construção de um sistema de exposição e recuperação de metadados.

IADIS International Conference, de 22 a 25 de outubro 2013, em Fort Worth, Texas, USA. Entitulado *"A prototype for querying heterogeneous sources on the web"* (FILGUEIRAS et al., 2013), este artigo foi aceito para submissão na conferência WWW/INTERNET. Ele apresenta as ideias defendidas nesta dissertação, em especial, a propositura de um sistema (protótipo) para consultar informações em

fontes de dados heterogêneas a partir de palavras-chave do usuário. Como resultado, denfedeu-se neste artigo que a solução mostrou-se viável ao permitir a consulta por palavras-chave em bibliotecas digitais e bancos de dados relacionais utilizando o protocolo OAI-PMH, com a utilização de web service que possibilitou que informações de bancos de dados relacionais fossem obtidas por aplicações clientes sem que necessitassem conhecer a estrutura dos bancos de dados consultados ou uma linguagem de consulta como SQL.

Além desses dois artigos, e como bem destaca o Capítulo 2, no início desta pesquisa foi realizada uma revisão sistemática intitulada "Consulta com palavras-chave em Banco de Dados Relacionais: uma revisão sistemática" (RAMADA et al., 2013) que pretendeu identificar e avaliar os estudos relevantes que explorassem o uso de palavras-chave para consulta em bancos de dados relacionais além das técnicas que têm sido aplicadas para consulta com palavras-chave. A revisão se encontra concluída mas ainda não foi publicada.

Bibliografia

BACKHOUSE, J.; HSU, C.; MCDONNELL, A. Toward public-key infrastructure interoperability. *Commun. ACM*, ACM, New York, NY, USA, v. 46, n. 6, p. 98–100, jun. 2003. ISSN 0001-0782. Disponível em: <http://doi.acm.org/10.1145/777313.777345>.

BAPTISTA, A. A.; COSTA, S. M. d. S.; KURAMOTO, H.; RODRIGUES, E. Comunicação científica : o papel da open archives initiative no contexto do acesso livre. *Encontros Bibli: revista eletrônica de biblioteconomia e Ciência da Informação*, v. 12, n. 1, 2007.

BERGAMASCHI, S.; DOMNORI, E.; GUERRA, F.; ORSINI, M.; LADO, R. T.; VELEGRAKIS, Y. Keymantic: semantic keyword-based searching in data integration systems. *Proc. VLDB Endow.*, VLDB Endowment, v. 3, n. 1-2, p. 1637–1640, set. 2010. ISSN 2150-8097. Disponível em: <http://dl.acm.org/citation.cfm?id=1920841%-.1921059>.

BERGAMASCHI, S.; DOMNORI, E.; GUERRA, F.; LADO, R. T.; VELEGRAKIS, Y. Keyword search over relational databases: a metadata approach. In: *Proceedings of the 2011 ACM SIGMOD International Conference on Management of data*. New York, NY, USA: ACM, 2011. (SIGMOD'11), p. 565–576. ISBN 978-1-4503-0661-4. Disponível em: <http://doi.acm.org/10.1145/1989323.1989383>.

CANDELA, L. et al. Setting the foundations of digital libraries: The delos manifesto. *D-Lib Magazine*, v. 13, n. 3/4, 2007. Disponível em: <http://www.dlib.org/dlib-/march07/castelli/03castelli.html>.

CHEN, Y.; WANG, W.; LIU, Z. Keyword-based search and exploration on databases. In: *Proceedings of the 2011 IEEE 27th International Conference on Data Engineering*. Washington, DC, USA: IEEE Computer Society, 2011. (ICDE '11), p. 1380–1383. ISBN 978-1-4244-8959-6. Disponível em: <http://dx.doi.org/10.1109/ICDE.2011.5767958>.

COULOURIS; DOLLIMORE, J.; KINDBERG, T. *Distributed Systems: Concepts and Design (4th Edition) (International Computer Science)*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 2005. ISBN 0321263545.

CROFT, B.; METZLER, D.; STROHMAN, T. *Search Engines: Information Retrieval in Practice*. 1st. ed. USA: Addison-Wesley Publishing Company, 2009. ISBN 0136072240, 9780136072249.

DZIEKANIAK, G. Mapeamento do uso de padrões de metadados por comunidades científicas. In: *Congresse Brasileiro de Biblioteconomia, Documentação e Ciência da Informação*. [S.l.: s.n.], 2007.

EMMERICH, W.; KAVEH, N. Component technologies: Java beans, com, corba, rmi, ejb and the corba component model. In: *Software Engineering, 2002. ICSE 2002. Proceedings of the 24rd International Conference on*. [S.l.: s.n.], 2002. p. 691–692.

FAN, H.; GUI, H. Study on heterogeneous data integration issues in web environments. In: *Wireless Communications, Networking and Mobile Computing, 2007. WiCom 2007. International Conference on*. [S.l.: s.n.], 2007. p. 3755–3758.

FERROS, L. M.; FERREIRA, M.; RAMALHO, J. C. Digitarq e o novo módulo de interoperabilidade oai-pmh. *APBAD*, 2010. Disponível em: <http://hdl.handle.net-/1822/10532>.

FILGUEIRAS, A. C.; SILVA, J. Consulta com palavras-chave em repositórios heterogêneos: Uma visão sobre bibliotecas digitais e bancos de dados relacionais diante da perspectiva de interoperabilidade. *Anais IX Conpeex*, v. 1, p. 9762–9766, 2012.

FILGUEIRAS, A. C.; SILVA, J.; RIZZO, A. A prototype for querying heterogeneous sources on the web. In: *ICWI/2013 (Aceito para publicação)*. [S.l.: s.n.], 2013.

FOUNDATION, A. S. *Solr Tutorial Overview*. 11 2012. Acessado em 25/11/2012. Disponível em: <http://www.lucene.apache.org/solr>.

GARCIA-ALVARADO, C.; ORDONEZ, C. Keyword search across databases and documents. In: *Proceedings of the 2nd International Workshop on Keyword Search on Structured Data*. New York, NY, USA: ACM, 2010. (KEYS '10), p. 2:1–2:6. ISBN 978-1-4503-0187-9. Disponível em: <http://doi.acm.org/10.1145/1868366.1868368>.

GARCIA, P. d. A. B.; SUNYE, M. S. O protocolo oai-pmh para interoperabilidade em bibliotecas digitais. *I Congresso de Tecnologias para Gestão de Dados e Metadados do Cone Sul*, 2003.

GARCíA-CRESPO, Á.; BERBíS, J. M. G.; PALACIOS, R. C.; SáNCHEZ, F. G. Digital libraries and web 3.0. the callimachusdl approach. *Computers in Human Behavior*, v. 27, n. 4, p. 1424–1430, 2011.

HASLHOFER, B.; KLAS, W. A survey of techniques for achieving metadata interoperability. *ACM Comput. Surv.*, ACM, New York, NY, USA, v. 42, n. 2, p. 7:1–7:37, mar. 2010. ISSN 0360-0300. Disponível em: <http://doi.acm.org/10.1145-/1667062.1667064>.

HEFFELFINGER, D. *Java Ee 6 with Glassfish 3 Application Server*. Packt Publishing, Limited, 2010. (Community experience distilled). ISBN 9781849510370. Disponível em: <http://books.google.com.br/books?id=GCFdQ5SxPnQC>.

JOAI Overview: The Java-based Open Archives Initiative Data Provider & Harvester. 11 2012. Acessado em 24/11/2012.

KOWATA, E. T. *Metadados de Bancos de Dados Relacionais: Extração e Exposição com o Protocolo OAI-PMH*. Dissertação (Mestrado) — Universidade Federal de Goias, 2011.

LAGOZE, C.; PAYETTE, S.; SHIN, E.; WILPER, C. Fedora: an architecture for complex objects and their relationships. *Int. J. Digit. Libr.*, Springer-Verlag, Berlin, Heidelberg, v. 6, n. 2, p. 124–138, abr. 2006. ISSN 1432-5012. Disponível em: <http://dx.doi.org/10.1007/s00799-005-0130-3>.

LAGOZE, C.; SOMPEL, H. V. D. *The Open Archives Initiative Protocol for Metadata Harvesting*. [S.l.], 12 2008. Disponível em: <http://www.openarchives.org/OAI-/openarchivesprotocol.html>.

LI, G.; FENG, J.; WANG, J.; ZHOU, L. An effective and versatile keyword search engine on heterogeneous data sources. *Proc. VLDB Endow.*, VLDB Endowment, v. 1, n. 2, p. 1452–1455, ago. 2008. ISSN 2150-8097. Disponível em: <http://dl.acm.org/citation-.cfm?id=1454159% -.1454198>.

LYLE, B. *Java API for WordNet Searching(JAWS)*. 2009.

MILLER, G. A. Wordnet: A lexical database for english. *Communications of the ACM*, v. 38, p. 39–41, 1995.

NILSSON, M.; BAKER, T. *Interoperability Levels for Dublin Core Metadata*. 2009. Http://dublincore.org/documents/2009/05/01/interoperability-levels/.

OLDENBURG, I. for S. N. *PhpOAI 2 Data Provider*. 2013. Disponível em: <http://physnet.uni-oldenburg.de/oai>.

OLIVEIRA, R. R. D. *Recupeção Contextualizada de Documentos Integrados pelo Protocolo OAI-PMH*. Dissertação (Mestrado) — UFG, 2010.

ONLINE Etymology Dictionary. Mar 2013. Disponível em: <http://dictionary.reference-.com/browse/cite>.

PITOURA, E.; BUKHRES, O.; ELMAGARMID, A. Object orientation in multidatabase systems. *ACM Comput. Surv.*, ACM, New York, NY, USA, v. 27, n. 2, p. 141–195, jun. 1995. ISSN 0360-0300. Disponível em: <http://doi.acm.org/10.1145/210376.210378>.

POWELL, A.; NILSSON, M.; NAEVE, A.; JOHNSTON, P.; BAKER, T. *DCMI Abstract Model*. June 2007. DCMI Recommendation.

PRESS, N. *Understanding Metadata*. [S.l.]: National Information Standards Organization Press, 2004. ISBN 1-880124-62-9.

PRINCETON UNIVERSITY. *About WordNet*. 2010. Acessado em 29/11/2012. Disponível em: <http://wordnet.princeton.edu/wordnet>.

RAMADA, M. S.; SILVA, J.; FILGUEIRAS, A. C. *Consulta com palavras-chave em Banco de Dados Relacionais: uma revisão sistemática*. 2 2013.

RICHARDSON, L.; RUBY, S. *Restful web services*. First. [S.l.]: O'Reilly, 2007. ISBN 9780596529260.

SAELEE, J.; BOONJING, V. A metadata search approach to keyword search in relational databases. In: *Proceedings of the 2008 Third International Conference on Convergence and Hybrid Information Technology - Volume 01*. Washington, DC, USA: IEEE Computer Society, 2008. (ICCIT '08), p. 571–576. ISBN 978-0-7695-3407-7. Disponível em: <http://dx.doi.org/10.1109/ICCIT.2008.70>.

SANTOS, V. dos. *Uma arquitetura suportada por busca semântica para recuperação de fontes de informação em repositórios de metadados*. Dissertação (Mestrado) — Universidade Federal Fluminense, 2011.

SAYãO, L. F. Afinal, o que é biblioteca digital? *Revista USP*, scielosibi, p. 6 – 17, 02 2009. ISSN 0103-9989.

SIDHUNATA, H. R.; CROUCHER, J. L.; FRANCES, M. Selective harvesting: Creating and ingesting custom oai-pmh sets. In: *4th eResearch Australasia Conference*. [S.l.: s.n.], 2012.

SILBERSCHATZ, A.; KORTH, H.; SUDARSHAN, S. *Sistema de bancos de dados*. Makron Books, 1999. ISBN 9788534610735. Disponível em: <http://books.google-.com.br/books?id=JXHFAAAACAAJ>.

SILVA, B. C. Dias da. Brazilian portuguese wordnet: A computational linguistic exercise of encoding bilingual relational lexicons. *International Journal of Computational Linguistics and Applications*, v. 1, n. 1-2, p. 137–150, Jan-Dec 2010 2010.

SILVA, J. C. da; KOWATA, E. T.; VINCENZI, A. M. R. Extracting and exposing relational database metadata on the web. *IADIS International Conference WWW/Internet 2012*, 2012.

SOMMERVILLE, I. *Software Engineering: (Update) (8th Edition)*. 8. ed. [S.l.]: Addison Wesley, 2006. Hardcover. ISBN 0321313798.

TAMMARO, A. M.; SALARELLI, A. *A Biblioteca Digital*. [S.l.: s.n.], 2008.

TANENBAUM, A. S.; STEEN, M. v. *Distributed Systems: Principles and Paradigms (2nd Edition)*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 2006. ISBN 0132392275.

TEIXEIRA, I. Uma linguagem de busca para sistemas de recuperação de informação. *Ciência da Informação*, v. 3, n. 1, 1974. ISSN 1518-8353. Disponível em: <http://revista.ibict.br/ciinf/index.php% -/ciinf/article/view/1687/1292>.

TELANG, A.; CHAKRAVARTHY, S.; HUANG, Y. Information integration across heterogeneous sources: Where do we stand and how to proceed? In: DAS, G.; SARDA, N. L.; REDDY, P. K. (Ed.). *COMAD*. Computer Society of India / Allied Publishers, 2008. p. 186–197. ISBN 978-81-8424-370-3. Disponível em: <http://dblp.uni-trier.de/db/conf/comad/comad2008.html>.

TRINH, Q.; BARKER, K.; ALHAJJ, R. Semantic interoperability between relational database systems. In: *Proceedings of the 11th International Database Engineering and Applications Symposium*. Washington, DC, USA: IEEE Computer Society, 2007. (IDEAS '07), p. 208–215. ISBN 0-7695-2947-X. Disponível em: <http://dx.doi.org/10-.1109/IDEAS.2007.40>.

URBAN, R. J. *Principle Paradigms Revisting the Dublin Core 1:1 Principle*. Tese (Doutorado) — University of Illinois, 2012.

VACARI, I.; VISOLI, M. C.; LEITE, F. C. L.; PONTES, S. D. d. C. L. D.; OKAWACHI, M. F.; SIMõES, V. P. M.; GONZALES, L. E.; PRAXEDES, M. G. G. Software livre para implementação de reposiórios digitais e provedores de serviços: Experiência da embrapa informática agropecuária. *JSL 2010*, ISSN 1850-2857, p. 2345, 2010.

VINOSKI, S. Where is middleware? *IEEE Internet Computing*, IEEE Educational Activities Department, Piscataway, NJ, USA, v. 6, n. 2, p. 83–85, mar. 2002. ISSN 1089-7801. Disponível em: <http://dx.doi.org/10.1109/4236.991448>.

W3C Working Group. *Web Services Architecture*. 2004.

WARD, J. H.; TORCY, A. d.; CHUA, M.; CRABTREE, J. Extracting and ingesting ddi metadata and digital objects from adata archive into the irods extension of the nara tpap using the oai-pmh. *e-Science.*, v. 34, 2009.