

Universidade de Brasília - UnB
Faculdade UnB Gama - FGA
Engenharia de Software

**Uma proposta para conversão de metadados
Dublincore para RDF: Integração para
bibliotecas de Participação Social**

Autor: Luiz Fernando de Freitas Matos
Orientador: Prof. Dr. Paulo Roberto Miranda Meirelles

Brasília, DF
2014



Luiz Fernando de Freitas Matos

**Uma proposta para conversão de metadados Dublincore
para RDF: Integração para bibliotecas de Participação
Social**

Monografia submetida ao curso de graduação em (Engenharia de Software) da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em (Engenharia de Software).

Universidade de Brasília - UnB

Faculdade UnB Gama - FGA

Orientador: Prof. Dr. Paulo Roberto Miranda Meirelles

Brasília, DF

2014

Luiz Fernando de Freitas Matos

Uma proposta para conversão de metadados Dublincore para RDF: Integração para bibliotecas de Participação Social / Luiz Fernando de Freitas Matos. – Brasília, DF, 2014-

85 p. : il. (algumas color.) ; 30 cm.

Orientador: Prof. Dr. Paulo Roberto Miranda Meirelles

Coorientador:

Trabalho de Conclusão de Curso – Universidade de Brasília - UnB
Faculdade UnB Gama - FGA , 2014.

1. Linked Data. 2. Metadados. I. Prof. Dr. Paulo Roberto Miranda Meirelles. II. . III. Universidade de Brasília. IV. Faculdade UnB Gama. V. Uma proposta para conversão de metadados Dublincore para RDF: Integração para bibliotecas de Participação Social

CDU 02:141:005.6

Luiz Fernando de Freitas Matos

Uma proposta para conversão de metadados Dublincore para RDF: Integração para bibliotecas de Participação Social

Monografia submetida ao curso de graduação em (Engenharia de Software) da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em (Engenharia de Software).

Trabalho aprovado. Brasília, DF, 01/12/2014:

**Prof. Dr. Paulo Roberto Miranda
Meirelles**
Orientador

Prof. Dr. Fernando William Cruz
Convidado 1

**Prof. Msc. Ronald Emerson Scherolt
da Costa**
Convidado 2

Brasília, DF
2014

Agradecimentos

Agradeço primeiramente a Deus, pela iluminação, conhecimento, sobretudo saúde e várias outras coisas para superação das dificuldades.

Agradeço a minha família, principalmente aos meus pais, João Alves e Célia Maria, por todo apoio incondicional, carinho, conselhos, educação, e muitas outras coisas dadas durante todas as etapas da minha vida. Todo esse apoio foi responsável pelas oportunidades geradas e permitir realizar o meu futuro. Aos meus irmãos Karla Joyce e João Paulo por toda a amizade, forças e apoio prestados durante todos esses anos.

Agradeço pela orientação do professor Prof. Dr. Paulo Meirelles pela constâncias das orientações do trabalho de conclusão de curso. Agradeço também pelas oportunidades acadêmicas e profissionais oferecidas durante esse ano e por ter repassado o seu conhecimento técnico durante as disciplinas sobre Métricas de Código-Fonte, git, Ruby, licenças de software, metodologia de trabalho, entre outros, fator fundamental para o meu crescimento profissional.

Agradeço ao Prof. Dr. Fernando Willian Cruz, por todo conhecimento repassado durante as disciplinas da graduação de Engenharia de Software. Também agradeço pela aceitação em participar do Projeto do Participa.br e mostrar que é possível ser diferente fazendo um trabalho correto. Pelo orientação em meu trabalho, permitindo explorar horizontes que eu não havia explorado ainda, estendendo ainda mais minha aprendizagem em diferentes aspectos da Engenharia de Software e Ciência da Informação.

Agradeço a todos os membros que fazem parte do laboratório LAPPIS, assim como os companheiros da Engenharia de Software, que apesar de não estar citando os nomes, a troca de experiências e amizades feitas durante nesses últimos anos foram vitais para o meu amadurecimento e ganho de forças para conclusão de mais uma etapa.

Agradeço a todos os membros da informática da Justiça Federal, que durante o período de estágio me ajudaram a acordar e acreditar novamente que era possível chegar nessa etapa.

Resumo

A participação da sociedade civil em decisões governamentais é um dos principais mecanismos de consolidação da democracia de um país. Dentro do poder executivo, essa participação popular ocorre por meio dos mecanismos formais (dentre eles conselhos, conferências, mesas de diálogo, fóruns, etc.), o qual foram instituídos a partir da Constituição de 1988. Através do surgimento de novas tecnologias e mecanismos formais, encontrar informações sobre os mesmos vem se tornando cada vez mais difícil, uma vez que são descentralizados e desatualizados. A partir dessas necessidades em 2014 surgiu o Participa.br, que é a rede de participação social do governo federal, que busca ser um canal digital centralizado contendo as principais informações relativas de participação social. Essas informações serão disponibilizadas através de uma biblioteca digital. Este trabalho propõe-se para a integração de um software livre para biblioteca digital com um *framework* de redes sociais. Para este trabalho, o portal Participa.br foi usado como base. Ele utiliza o software livre Noosfero como um provedor de serviços e os mecanismos formais de participação social como provedores de dados, cuja interoperabilidade ocorre por meio do protocolo OAI-PMH. Dessa forma, ao final desse trabalho foi desenvolvido um *plugin* para realizar a interoperabilidade utilizando o OAI-PMH, apresentando os resultados utilizando os conceitos de Web Semântica, através do uso de RDFs.

Palavras-chave: bibliotecas digitais, participação social, oaipmh, triplificação, rdf, web semântica.

Abstract

Social participation in government decisions is a major consolidation mechanism of democracy in a country. In the executive power, that public participation occurs through formal mechanisms (including advice, conferences, roundtables, forums, etc.), which were established by the Constitution of 1988. With the creation of new technologies and formal mechanisms, find information about them is becoming increasingly difficult, since they are decentralized and outdated. From these problems, the Participa.br was created in 2014. The Participa.br is the network of social participation of the federal government, which seeks to be a centralized digital channel containing the main information for social participation. This information will be made available through a digital library. This degree monograph proposes to integrate an open source digital library with a framework of social network. For this work, the portal Participa.br was used as a base. It uses free software Noosfero as a service provider and the formal mechanisms of social participation as data providers, whose interoperability occurs through the OAI-PMH protocol. Therefore, a plug-in was developed to perform this interoperability using OAI-PMH protocol, presenting the results using the concepts of the Semantic Web, by using RDFs.

Keywords: digital libraries, social participation, oai-pmh, rdf, triplification, semantic web.

Lista de ilustrações

Figura 1 – Componentes de um objeto digital	26
Figura 2 – Modelo conceitual para projeto de bibliotecas digitais	27
Figura 3 – Identificação dos principais componentes de uma biblioteca	28
Figura 4 – Entregáveis de uma biblioteca	29
Figura 5 – Exemplo de escrita bibliográfica em formato Dublin Core	31
Figura 6 – Finalidade do Omeka	34
Figura 7 – Funcionamento dos provedores de serviços	38
Figura 8 – Arquitetura de funcionamento do OAI-PMH	40
Figura 9 – Tela Inicial do Participa.br.	47
Figura 10 – Arquitetura de funcionamento do Noosfero	50
Figura 11 – Relacionamentos entre perfis, ambientes e domínios	50
Figura 12 – Relacionamento entre os tipos de perfis	51
Figura 13 – Relacionamento entre os tipos de conteúdo do Noosfero	51
Figura 14 – Funcionamento de plug-ins no Noosfero	53
Figura 15 – Modelo de comunicação OAI-PMH entre o portal e os mecanismos	58
Figura 16 – Espaço de colaboração semântico e social no Participa.br	59
Figura 17 – Novos conteúdos adicionado no Noosfero	61
Figura 18 – Tela para adição de nova biblioteca	62
Figura 19 – Tela para adição de novo set	62
Figura 20 – Tratamento dos sets para apresentar aos usuários	64
Figura 21 – Lista de objetos digitais	66
Figura 22 – Detalhes de um objeto digital	66
Figura 23 – Painel de configuração do Virtuoso no Noosfero	67
Figura 24 – Painel de inserção de <i>Namespaces</i>	68
Figura 25 – Painel de mapeamento dos metadados	69
Figura 26 – Resultado da triplicação da biblioteca	71

Lista de tabelas

Tabela 1 – <i>Namespaces</i> utilizados para triplicação dos dados	81
Tabela 2 – Elementos análogos utilizados para mapeamento dos termos em Dublin Core	82

Lista de abreviaturas e siglas

AGPL	GNU Affero General Public License
DC	Dublin Core
HTML	HyperText Markup Language
HTTP	Hipertext Transfer
IBICT	Instituto Brasileiro de Informação em Ciência e Tecnologia
MVC	Model-View-Controller
OAI	Open Archives Initiative
OAI-PMH	Open Archives Initiative Protocol for Metadata Harvesting
Participa.br	Portal de Participação Social
PNUD	Programa das Nações Unidas para o Desenvolvimento
RDF	Resource Description Framework
Set	Grupos de assuntos relacionados
SERPRO	Serviço Federal de Processamento de Dados
SPARQL	SPARQL Protocol and RDF Query Language
TDD	Test Driven Development
URI	Identificador de Endereços na Internet
XML	Extensible Markup Language.

Sumário

1	Introdução	19
1.1	Problemas	20
1.2	Objetivos	21
1.2.1	Objetivo geral	21
1.3	Metodologia	21
1.4	Organização do Trabalho	22
2	Bibliotecas Digitais	25
2.1	Definição	25
2.2	Arquitetura das Bibliotecas Digitais	27
2.3	Padrão de metadados para catalogação dos dados	30
2.4	Softwares para criação de Biblioteca Digital	31
2.5	Considerações	35
3	Protocolos de interoperabilidade e disponibilização de informações	37
3.1	Iniciativa <i>Open Archive Initiative</i> (OAI)	37
3.1.1	Provedor de dados	37
3.1.2	Haverster	38
3.1.3	Provedor de serviços	38
3.1.4	Protocolo OAI-PMH	39
3.2	Linked Data	41
3.2.1	RDF - Resource Description Framework	42
3.2.2	Ferramentas para armazenamento de RDF	43
3.3	Considerações	45
4	Plataforma de Participação Social	47
4.1	Participa.br	47
4.2	Noosfero: A Plataforma Utilizada no Participa.br	48
4.2.1	Arquitetura do núcleo	49
4.2.2	Arquitetura de <i>plug-ins</i>	52
4.3	Considerações	53
5	Conversão de dados Dublincore para RDF	55
5.1	A biblioteca digital de participação social	55
5.2	Escolha do protocolo de interoperabilidade	56
5.3	Adaptando as ferramentas para a arquitetura proposta	59
5.4	O plugin OAI-PMH para o Noosfero	60
5.4.1	Adaptando a arquitetura do Noosfero	60

5.4.2	Desenvolvimento do haverster	61
5.4.3	Desenvolvimento da triplificação	65
5.5	Considerações	71
6	Conclusão	73
6.1	Contribuições	75
6.2	Limitações	76
6.3	Trabalhos Futuros	76
	 Apêndices	 79
	APÊNDICE A – Metadados utilizados para triplificação	81
	 Referências	 83

1 Introdução

Desde o início da década de 1990 até os dias de hoje a participação popular em diversas frentes vem sendo amparada e institucionalizada em diversos países pelo mundo, tornando-os verdadeiras democracias representativas (JACOBI, 2001). No Brasil esse fato ocorreu ainda no ano de 1988, onde a Constituição 1988¹ define o Brasil como um Estado Democrático de Direito, atribuindo um novo modelo para gestão pública que trouxe a participação popular como o exercício pleno da cidadania, fazendo com que o cidadão tenha conscientização que ele faz parte da democracia, buscando melhorias do bem estar social para todos (SCUASSANTE, 2009).

A participação popular pode ser compreendida como um processo no qual homens e mulheres se descobrem como sujeitos políticos, exercendo os direitos políticos (ALMEIDA, 2011). Após 2003, o governo passou a enxergar essa iniciativa popular como um mecanismo de concepção, execução e manutenção de novas políticas públicas. Antigamente todas as medidas que era tomadas era definidas apenas por técnicos e ministros do governo (REPÚBLICA, 2010). Uma das forma de participação previstas em lei é através dos mecanismos formais de participação criados na Constituição de 1988. Atualmente, existem uma gama de mecanismos formais de participação, sejam eles conselhos, conferências, ouvidorias, mesa de diálogos, entre outros. Todos eles possuem uma função semelhante: dar voz para o povo demandar, criticar, sugerir, elogiar, fazer políticas públicas, etc. Apesar da Constituição ter criado esses mecanismos, ainda fica claro que a população desconhece o poder de seus direitos, como por exemplo, muitos cidadãos ainda desconhecem a existência dos mesmos (ROCHA, 2011). Parte desse desconhecimento da população é em fator das informações desses mecanismos formais de participação estarem de forma desorganizada e descentralizada na internet.

Pensando nisso, foi criado o Participa.br², que é a Plataforma Nacional de Participação Social, desenvolvido a partir do software livre de redes sociais Noosfero. O Participa.br é um ambiente virtual de participação social, que promove o diálogo entre a administração pública e a sociedade civil, promovendo ações que ajudam a fortalecer os mecanismos de participação. Uma dessas ações é a criação de uma biblioteca digital contendo os principais artefatos digitais (documentos, áudios, vídeos, entre outros) referentes aos mecanismos formais de participação.

Atualmente a participação da sociedade civil na tomada de decisões políticas, construção políticas públicas, demandando soluções para os problemas encontrados no cotidiano e na construção coletiva de soluções possa já é uma realidade. Hoje, faz-se

¹ Disponível em: <http://www.planalto.gov.br/ccivil_03/constituicao/constituicao.htm>

² Maiores informações em: <<http://www.participa.br>>

necessário adotar medidas para que todos meios de participação consolidados nos períodos atuais, estejam compatíveis com a realidade encontrada no mundo tecnológico (redes sociais, participação social através do computador, conferências virtuais, entre outros).

1.1 Problemas

Embora exista uma quantidade enorme de informações sobre participação social e mecanismos formais de participação, a maioria delas possuem informações desatualizadas, principalmente as que estão disponibilizadas em catálogos impressos e em sites. Outro fator a ser considerado é o dinamismo presente nos mecanismos de participação³, provocando o surgimento constante de novidades, entre eles: *(i)* surgimento de novos mecanismos a cada ano; *(ii)* mudanças de nomes de gestores; e *(iii)* mudança de endereços e telefone. Por isso, se faz necessário a criação de um canal centralizado, onde estarão disponíveis todas as informações sobre os mecanismos formais de participação.

Esse canal foi proposto pela consultoria de Ciência da Informação do PNUD⁴ (Termo de Referência BRA/12/018), que propõe a criação de uma biblioteca digital contendo os principais documentos relacionados aos mecanismos de participação social. No entanto essa biblioteca além de ser um ambiente fora do contexto do Noosfero do Participa.br (principalmente por ser desenvolvida em outra ferramenta), ainda conterà documentos que vão agregar pouco valor aos seus usuários.

Com base nisso, trazendo os artefatos digitais para dentro do Participa.br, pode aumentar o interesse dos usuários nesses documentos, além disso pode-se pensar na utilização de uma abordagem baseada em web semântica para gravação/apresentação desses artefatos. Com a inserção de aspectos semânticos nesses documentos, o ambiente pode inferir ou reconhecer outros sites relacionados à participação social, apresentando novos metadados automaticamente para um artigo ou página que está sendo visualizado no momento do uso, criando um serviço especializado que trás maior interesse dos usuários, além de permitir que motores de busca possam localizar esses documentos de forma mais facilitada.

Com bases nesses problemas, foram descritas algumas perguntas que serão utilizadas para direcionar o trabalho.

- **Q1** - Como as bibliotecas digitais garantem a atualização das informações presentes nos mecanismos formais de participação social?

³ Exemplo disso é em relação aos Conselhos, que no total trinta e cinco, desses, dezenove Conselhos foram criados no ano 2003 a 2013 e outros dezesseis foram reformulados com o objetivo de ampliar a participação social (REPÚBLICA, 2010).

⁴ Programa das Nações Unidas para o Desenvolvimento

- **Q2** - Bibliotecas digitais descentralizadas potencializam os mecanismos formais de participação social em rede?
- **Q3** - Como o Noosfero pode ser integrado com uma biblioteca digital?
- **Q4** - Como integrar os dados presentes nas bibliotecas digitais com namespaces e padrões de catalogação já propostas?

1.2 Objetivos

1.2.1 Objetivo geral

O objetivo deste trabalho de conclusão de curso foi o desenvolvimento de um *plugin* para o Noosfero, que realize a colheita (*harvesting*) de todas as informações presentes na biblioteca digital de participação social para dentro do Participa.br e que apresente-as utilizando aspectos semânticos.

Objetivos específicos

- **OE1** - Adaptação do Noosfero⁵, uma ferramenta para criação de redes sociais, para implementar o protocolo OAI-PMH. Esse protocolo vai garantir que as informações sobre os mecanismos formais de participação presentes nas bibliotecas digitais sejam visualizados na própria ferramenta Participa.br.
- **OE2** - Adaptar o Noosfero para armazenar triplas RDF referentes aos artefatos (atas, normas, regimentos, entre outros) dos mecanismos formais de participação.
- **OE3** - Permitir uma visualização mais detalhada dos artefatos referentes aos mecanismos formais de participação dentro do Noosfero;

1.3 Metodologia

Do ponto de vista metodológico, realizou-se uma pesquisa teórica inicial buscando entender o que é a participação social, e o que são e como funcionam os mecanismos formais de participação presenciais e virtuais, além de exemplos sobre alguns mecanismos de participação virtual já consolidados.

Após isso, realizou-se um estudo inicial sobre o tema na literatura concernente (em especial nas publicações disponibilizadas pelo IPEA⁶) e nos conteúdos disponibilizados nos sites dos mecanismos de participação social.

⁵ Mais informações em: <<http://noosfero.org>>

⁶ O IPEA (Fundação Instituto de Pesquisa Econômica Aplicada) é um organização pública vinculada a Secretaria de Assuntos Estratégicos da Presidência da República. O IPEA é responsável por fornecer

A partir do levantamento inicial das necessidades a serem resolvidas, criou-se um catálogo, contendo as informações sobre os principais mecanismos formais de participação, a fim de fazer um levantamento inicial para criação de uma biblioteca digital, utilizado para homologar o plug-in a ser desenvolvido.

A fim de manter uma solução mais duradoura para o problema da catalogação e organização de conteúdos, percebeu-se que a solução mais adequada é a proposição de uma biblioteca digital, na qual as informações que eventualmente sejam alteradas num determinado Conselho, Conferência ou Ouvidoria possam ser automaticamente compartilhadas por todos e devidamente atualizadas no portal Participa.br. Essa biblioteca digital está sendo desenvolvida e é produto de uma consultoria entre o PNUD através do termo de referência BRA/12/018.

Ainda com base nos problemas levantados, percebeu-se a necessidade de integração dessa biblioteca com o Noosfero presente no Participa.br, melhorando o acesso dos documentos para os usuários, além de permitir futuramente que os mecanismos formais possam utilizar as ferramentas de participação que o portal de participação federal fornece.

Em seguida, foram feitos estudos sobre plataformas de software livre voltadas para bibliotecas digitais e foram encontradas algumas soluções que poderiam ser adotadas no projeto, as quais estão listadas na seção 2.4 deste trabalho. Com base nas ferramentas e suas características, percebeu-se que todas as opções haviam suporte ao protocolo OAI-PMH, que será visto na seção 3, logo foi estudada soluções e bibliotecas (código) na linguagem Ruby, que ajudassem no desenvolvimento desse plug-in de integração entre o Participa.br e a biblioteca.

Do mesmo modo que foi feito no estudo de softwares de bibliotecas, realizou-se um estudo de plataformas de software livre voltadas para o armazenamento de RDFs (*Resource Description Framework*), que são candidatas a serem utilizadas para armazenamento de dados referentes aos mecanismos formais de participação, assim como soluções na linguagem Ruby que auxiliassem na integração dessa ferramenta com o Noosfero. Essa abordagem semântica, permite que os dados e documentos referentes aos mecanismos formais possam ser catalogados de maneira mais ampla, permitindo que os motores de busca, além dos próprios usuários possam encontrar os artefatos mais facilmente.

1.4 Organização do Trabalho

Este trabalho está dividido dentro de 6 outros capítulos.

No 2º capítulo é apresentado um conceito sobre o funcionamento das bibliotecas digitais, assim como exemplos de ferramentas que podem ser utilizadas;

suporte técnico e institucional às ações governamentais para a formulação e reformulação de políticas públicas e programas de desenvolvimento brasileiros.

No 3º será apresentado um estudo sobre os protocolos de interoperabilidade utilizado para disponibilizar os conteúdos da biblioteca digital dentro do Participa.br, assim como os conceitos de *Linked Data*, utilizados para apresentar os dados de maneira mais completa no Participa.br.

Já No 4º capítulo será apresentado o Participa.br, a plataforma virtual para participação social, assim como um estudo sobre o funcionamento da arquitetura e como funciona o Noosfero, a plataforma livre para criação de redes sociais que foi utilizado pelo Participa.br.

No 5º capítulo será apresentado a arquitetura de funcionamento da biblioteca de participação social, assim como toda a aplicação do conteúdo proposto nesse trabalho, logo, todo o desenvolvimento do plugin e aplicação dos conceitos.

O trabalho se encerra com a conclusão, assim como as restrições encontradas. Também é feita uma análise sobre os levantamentos, além de uma discussão e trabalhos futuros, que podem incrementar esse trabalho.

2 Bibliotecas Digitais

Nesse capítulo serão apresentados conceitos sobre bibliotecas digitais, uma das soluções encontradas para catalogação dos mecanismos formais de participação.

2.1 Definição

Uma biblioteca digital não é simplesmente uma coleção digital com ferramentas para manutenção das informações. Ela é composta por um ambiente que possui coleções, serviços e pessoas apoiando um ciclo de vida de disseminação, uso e preservação do dado, informação e conhecimento.

De acordo com a definição provida pela ARL¹, uma biblioteca digital é uma entidade que possui as seguintes características:

- É ligada à tecnologia, permitindo acesso à vários de seus recursos;
- A biblioteca não é uma entidade isolada, mas sim interligada com outras bibliotecas (Através do uso de protocolos de interoperabilidade como Z39.50 ou OAI-PMH);
- Um dos seus objetivos é ser universalmente acessível, logo serve a vários tipos de usuários;
- As coleções das bibliotecas digitais não estão limitadas somente a documentos impressos, mas sim a artefatos digitais que não podem ser distribuídos em meios impressos (áudio, vídeo, mapas digitais, entre outros)

Arms (2000) ressalta a importância de um bom projeto de interface para uma biblioteca digital, já que bons projetos de interfaces melhoram a usabilidade entre o usuário e a biblioteca digital, conseqüentemente reduzindo o esforço necessário para compreender a arquitetura da informação dos objetos dentro de uma biblioteca digital.

Um objeto digital é a unidade básica de informação de uma biblioteca digital, em conjunto com os seus metadados e demais atributos. Existem padrões estabelecidos para a definição de cada artefato, como a proposição feita por (KURAMOTO, 2014), e que está ilustrada na Figura 1.

¹ A Association of Research Libraries (ARL) é uma organização sem fins lucrativos, composta por várias instituições entre as principais: Associação de Universidades Americanas (AAU), a Associação Canadense de Bibliotecas de Pesquisa (CARL) e SPARC Europa. Essa entidade tem o objetivo de promover pesquisas, recomendar padrões e integrar as instituições envolvidas. Maiores informações <<http://www.arl.org/>>

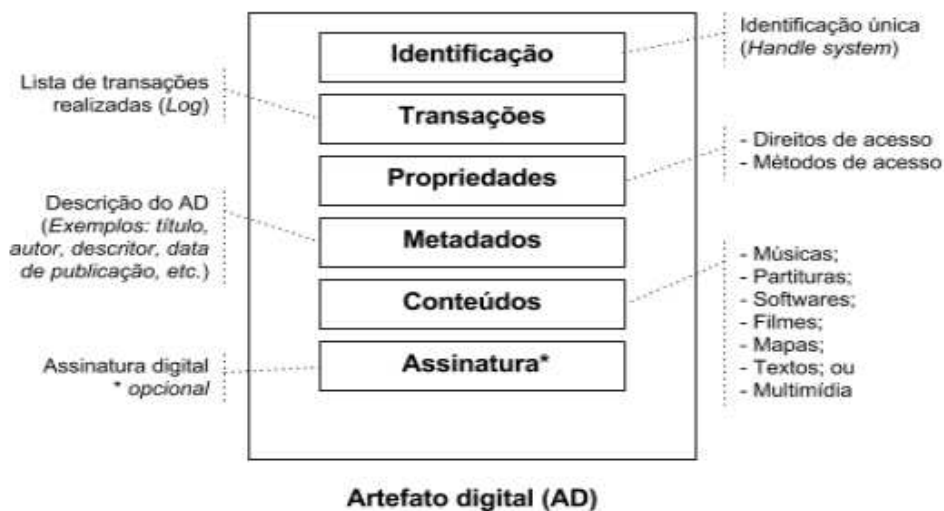


Figura 1 – Componentes de um objeto digital. Adaptado de Kuramoto (2014)

Ainda de acordo com a Figura 1, os artefatos digitais devem possuir: (i) uma assinatura digital como forma de certificação do produto; (ii) uma identificação do tipo de conteúdo que possui; (iii) os metadados descritivos, estruturais e administrativos; (iv) os direitos de acesso; (v) um controle de acessos e, finalmente (vi) uma identificação única. Abaixo é descrito cada um desses componentes:

- **Identificação** - Cada item tem que possui um identificador único, que é uma propriedade que distingue um objeto dos demais. Esse identificador pode ser um campo incremental, algum número de identificação único (CPF ou CNPJ), as iniciais do artefato (*slugs*), entre outros.
- **Transações** - É responsável por identificar e guardar em um arquivo de *log* (registros), número de acessos a determinado artefato digital, quais itens do artefato digital foram acessados, etc.
- **Propriedades** - Elas identificam quais são as diretivas de acesso para determinado artefato digital, no caso o artefato pode ser visualizado publicamente ou ter uma lista de determinados usuários que podem acessar o mesmo. Também identifica as permissões de controle a esse item artefato (ler, editar, deletar). A propriedade identifica também qual será o caminho utilizado para se chegar a determinado artefato digital e como é feita a divisão do mesmo.
- **Metadados** - Os metadados descrevem as características de um artefato digital ².

² Visto mais detalhadamente adiante, na seção 2.3

- **Conteúdo** - O conteúdo é algo que faz parte de um artefato digital. Um artefato digital pode ser composto de vários tipos de conteúdo (áudio, vídeo, imagem, textos, etc.).
- **Assinatura** - A assinatura digital garante a autenticidade daquele artefato digital. Ele garante que o item acessado possui veracidade em suas informações, além de permitir que aquele objeto digital está com suas informações integras (nada foi perdido).

2.2 Arquitetura das Bibliotecas Digitais

De acordo com o que está apresentado na Figura 2, o projeto de interface é parte integrante do modelo conceitual do sistema, juntamente com o projeto funcional que especifica as funções a serem oferecidas aos usuários e ao projeto dos metadados associados aos dados que especificam a estrutura e a organização e descrição do conteúdo (ARMS, 2000, pp. 143–145)(FERREIRA; SOUTO, 2006, pp. 190).

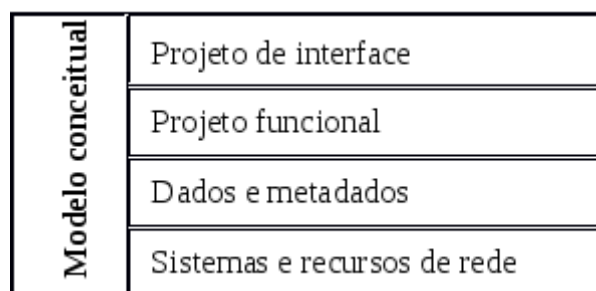


Figura 2 – Modelo conceitual para projeto de bibliotecas digitais. Extraído de Arms (2000, p. 144)

Ainda de acordo com esse o modelo ilustrado pela Figura 2 faz-se documentar todas as fases de implementação uma biblioteca completa. Apesar dessas documentação ficar escondida dos usuários, elas ajudam a criar um sistema com os módulos (busca, navegação, catalogação e demais) totalmente integrados, além de ajudar os usuários com uma sistema mais harmônico e robusto, conseqüentemente, tendo uma confiança na utilização do sistema pelo usuário. As Figuras 3 e 4 representam os sistemas compostos pela arquitetura e os entregáveis de uma arquitetura, respectivamente.

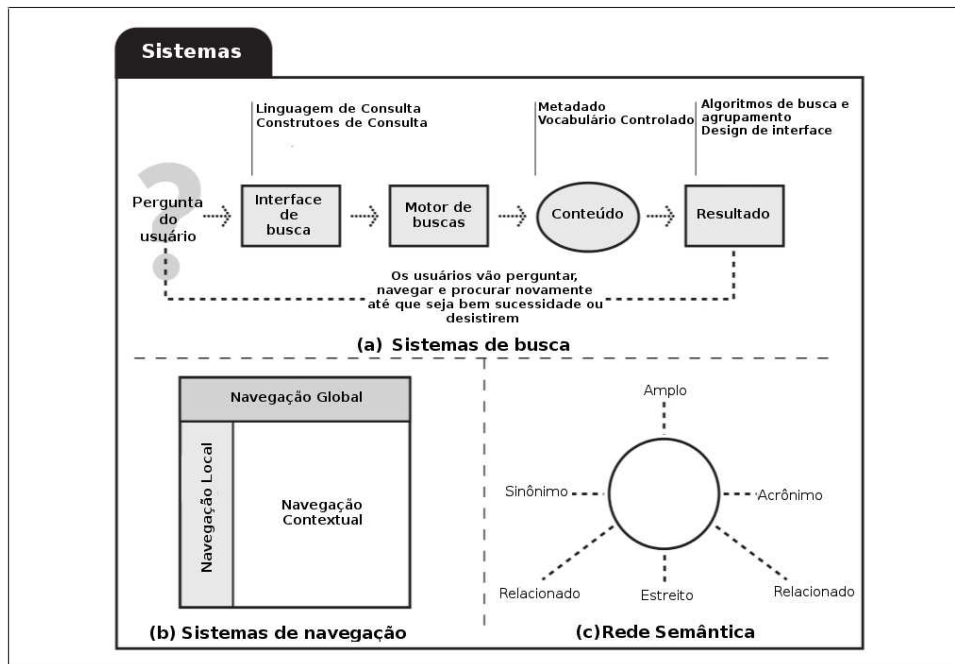


Figura 3 – Identificação dos principais componentes de um site. Adaptado de Rosenfeld e Morville (2002)

O diagrama ilustrado pela Figura 3, representa a arquitetura de informação utilizado em uma biblioteca. Se faz necessário ter uma documentação sobre a arquitetura de informação do sistema para os usuários terem um melhor entendimento, facilitando a interação do usuário com a biblioteca.

Um sistema de busca não é composto somente por uma interface bonita ou um bom motor de busca, mas por todo um conjunto integrado. O usuário tem que ser capaz de pesquisar qualquer tipo de conteúdo e ordena-lo conforme o seu gosto, e os resultados precisam ser apresentados de forma agradável. Portanto, na sua estrutura interna é preciso garantir que os resultados relativos a uma busca (expressa por um atributo de consulta, chamada de *query*) sejam compatíveis com as demandas dos usuários e, ao mesmo tempo, que esses resultados sejam apresentados segundo uma ordem de importância e também de maneira agradável no sentido de interface.

Para isso é necessário catalogar os artefatos digitais, pois definindo os metadados adequados facilitam o processo de busca. Adotando os padrões de metadados aceitos pela comunidade, como o Dublin Core (que será tratado na seção 2.3), facilita a integração dessa biblioteca com outros contextos e sites de buscas, como é o caso do Google.

A ampliação do sistema para permitir um ambiente semântico permitem buscas mais refinadas, pois permitem que se façam inferências entre objetos correlatos e ajudam a máquina a entender as pesquisas (*queries*) dos usuários com mais facilidade.

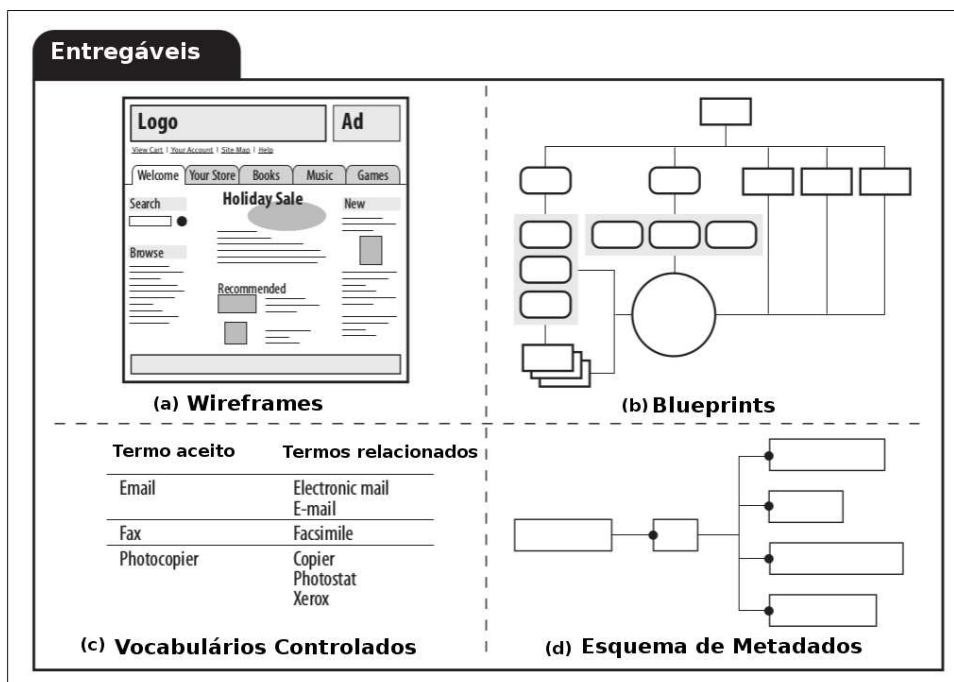


Figura 4 – Entregáveis de uma biblioteca. Adaptado de Rosenfeld e Morville (2002)

A Figura 4 mostra os entregáveis de uma biblioteca, que são utilizados para um entendimento melhor do sistema a ser produzido. Esses entregáveis são compostos pelos (i) *Blueprints*, também chamados de mapa do site são uma espécie do funcionamento de toda a arquitetura de informação presente no site. Eles mostram as relações de caminho entres as páginas e seus componentes. (ii) Os Wireframes descrevem uma página individual, mostrando todos os conteúdos e arquitetura da informação relacionados com a página através de um protótipo. Esse protótipo pode dizer quais informações podem ser informadas para o usuário e servir de insumo para o sistema final. (iii) Os vocabulários controlados permitem que um item seja encontrado através de palavras relacionadas, isso ajuda o cliente em interfaces de busca. Um exemplo de vocabulário controlado é um tesouro³. (iv) O esquema de metadados, explicado mais detalhadamente na seção 2.3 ajudam a catalogar melhor os objetos digitais, fazendo com que a indexação de busca seja melhor.

Na primeira etapa deste trabalho, foi feito um primeiro protótipo de uma biblioteca digital, levantando alguns metadados iniciais com base em um catálogo de mecanismos formais. Esse processo catalogação gerou um primeira versão do esquema de metadados⁴. Entendemos também que apesar de não ser necessariamente um Wireframe, também foi feito um primeiro protótipo da biblioteca digital de participação social, com o objetivo de entender o problema. Esse protótipo pode ser entendido com um Wireframe, já que

³ Maiores informações em: <<http://pt.wikipedia.org/wiki/Tesouro>>

⁴ Disponibilizado no produto 01 da consultoria de Ciência da Informação de Cruz e Meirelles (2014). Informações em: <<https://docs.google.com/file/d/0Bz8tVudv1bR6T0ZrTWRxclJZM0k>>

apresenta uma interface inicial onde há a exibição de conteúdos para os usuários.

2.3 Padrão de metadados para catalogação dos dados

Na biblioteconomia, o metadado é dado estruturado que compartilha diversas características similares para a catalogação e descrevem as características de um determinado recurso informacional. Um registro de metadados consiste em um número predefinido de elementos que representam atributos específicos de um objeto e cada elemento pode estar associado a um ou mais valores (FEITOSA, 2006). Os esquemas de metadados devem possuir (independente da área de conhecimento) um número limitado de elementos, o nome de cada elemento e o significado de cada elemento (SOUZA; VENDRUSCULO; C, 2000).

Dentre os vários benefícios na utilização de metadados na web, pode-se citar os principais (CRUZ; MEIRELLES, 2014): *(i)* são estruturados e formam a base para o desenvolvimento de sistemas de busca, *(ii)* podem ser convertidos para outros formatos, ou seja, interoperar com diferentes protocolos de busca e recuperação, *(iii)* tornam mais fácil a extração de conteúdo por engines de busca (SEOs) e *(iv)* facilitam o gerenciamento do sistema de informação (com os metadados administrativos) porque ajudam a avaliar quando os recursos devem ser revistos ou removidos da base de dados.

Em relação agrupamento dos elementos de metadados de um recursos computacional (classificação), podem ser classificados em:

- **Descritivos** Passíveis de uso por sistemas de busca. Por exemplo o título, a descrição, URI, autor, idioma, formato de arquivo;
- **De assunto** (Relacionados à atividade de indexação) descrevem o conteúdo do documento, tais como as palavras-chave, os códigos e sistemas de classificação, termos do tesouro ou cabeçalho de assuntos; e
- **Administrativos** que facilitam a organização e administração do sistema de informação. Exemplos incluem o responsável pela manutenção do documento, data de adição do documento, data de expiração, dentre outros.

Do ponto de vista do grau de dificuldade de uso dos padrões, os metadados, podem ser classificados em formatos de bandas um, dois e três (FEITOSA, 2006). Dentro dessa classificação em banda, podemos encontrar vários graus de detalhamento e formalidades na catalogação dos recursos. Em grau de detalhamento, os metadados são classificados por (ALVES, 2004): *(i)* metadados simples, utilizados para descrever elementos de maneira mais simples (como é o caso das metatags utilizadas no padrão HTML); *(ii)* padrões de metadados mais estruturados, que oferecem um descrição mínima de catalogação de

um recurso, como é o caso do Dublin Core; e (iii) metadados complexos com um alto grau de formalidade e detalhes, que permitem descrever um recurso de um domínio bem específico e necessita de um alto grau de ajuda de especialistas, como é o caso do MARC 21, utilizado em áreas da biblioteconomia.

O Dublin Core é interessante porque envolve um conjunto de dados padronizados para descrição de recursos na web. Os metadados descritos pelo padrão Dublin Core são definidos como um conjunto de elementos de metadados planejados para facilitar a descrição de recursos eletrônicos.

Eles se caracterizam por sua utilidade e flexibilidade na representação dos dados e contém 15 elementos na sua composição básica⁵, conforme exemplo descrito na Figura 5. Esse padrão é acordo conceitual sobre termos de fácil uso e não necessita ser especializado no processo de catalogação para descrição dos recursos na web (CRUZ; MEIRELLES, 2014).

```
Dublin Core Example
Title="Metadata Demystified"
Creator="Brand, Amy"
Creator="Daly, Frank"
Creator="Meyers, Barbara"
Subject="metadata"
Description="Presents an overview of
metadata conventions in
publishing."
Publisher="NISO Press"
Publisher="The Sheridan Press"
Date="2003-07"
Type="Text"
Format="application/pdf"
Identifier="http://www.niso.org/
standards/resources/
Metadata_Demystified.pdf"
Language="en"
```

Figura 5 – Exemplo de escrita bibliográfica em formato Dublin Core

Na Web Semântica, os metadados oferecem trazem maiores significados aos objetos digitais, agregando um maior valor na catalogação dos dados, de modo que recuperação dos dados pelos usuários é facilitada;

2.4 Softwares para criação de Biblioteca Digital

Visto a importância das bibliotecas digitais nos mais variados contextos, inclusive dentro da participação social, nesta seção, foram documentadas e descritas as principais

⁵ A descrição de cada metadado está disponível no endereço: <<http://dublincore.org/>>

ferramentas em software livre⁶ pesquisadas e utilizadas para auxiliar o desenvolvimento desse trabalho.

Greenstone

Greenstone é um conjunto de softwares para a construção e distribuição de coleções de bibliotecas digitais. Ele organiza a informação, publicando-a na Internet ou em CD-ROM. Greenstone é produzido pela Universidade de Waikato, desenvolvido e distribuído em cooperação com a UNESCO e a ONG Human Info.

O objetivo do software Greenstone é capacitar os usuários, especialmente em universidades, bibliotecas e outras instituições de serviço público, para construir suas próprias bibliotecas digitais. Através de plugins, Greenstone pode importar documentos digitais em formatos de texto incluindo HTML, JPG, TIFF, MP3, PDF, vídeo e documentos.

Também é possível a utilização do protocolo OAI-PMH, que é um protocolo desenvolvido pela Open Archives Initiative, que é responsável por coletar metadados em diversos outros arquivos (repositórios) que são disponibilizados pela Internet, que será melhor estudado na seção 3.1.

O GreenStone possui a vantagem de ser altamente customizável, tanto no quesito de disponibilização de uma interface web compatível com o desejo do administrador, quanto na disponibilização de itens de acordo com o especificado na ferramenta de criação de uma biblioteca digital. Esse software é capaz de exportar os dados que compõe a biblioteca (textos, XML, entre outros) para uma mídia de CD/DVD-ROM, ou mesmo, exportado para um arquivo que pode ser utilizado para ser importado novamente em outro ambiente que possua o GreenStone.

Os plugins são utilizados para visualização de documentos e extração de metadados de um determinado tipo de arquivo. Cada tipo de arquivo tem um plugin específico que pode ser utilizado para abrir o mesmo no greenstone, assim como preencher os metadados automaticamente na parte de enriquecer, basta somente que o usuário direcione o plugin para ler os metadados nos arquivos que foram obtidos. Por exemplo, alguns plug-ins juntamente com o arquivo que ele abre são: (i) BibTexPlugin – Extensão que manuseia arquivos com a extensão .bibtex; (ii) CSVPlugin – Extensão que manuseia arquivos com a extensão .csv separados por vírgulas, cada título pode ser definido como um metadado; e (iii) HTMLPlugin – Arquivos .HTML (hipertexto) são abertos com esse plugin. Através das meta tags especificadas pela W3C (meta name), podemos fazer o greenstone obter os registros de metadados diretamente dos arquivos. Vale ressaltar que existem muitas outras extensões que estão documentadas na wiki oficial do greenstone⁷.

⁶ Um software livre permite aos usuários utilizarem, estudarem, modificarem e realizar a distribuição do mesmo, sem restrições e permitindo aos usuários futuros utilizarem de forma igual.

⁷ Maiores informações em: <<http://wiki.greenstone.org/wiki/index.php/Plugins>>

Uma das vantagens do GreenStone é possível a utilização de metadados, que são dados sobre dados, ou seja, são informações que caracterizam um determinado dado. Existem várias especificações de metadados, criadas pelos mais diversos tipos de finalidades, entre eles, o Dublin Core, o DLS e o EX⁸ quando o dado é importado para dentro da biblioteca digital.

Esses dados geralmente são preenchidos automaticamente e não são editáveis. Também podemos utilizar os plugins para preenchimentos desses metadados, basta apenas apontar para o local onde esses metadados estão especificados. Após esse apontamento, basta o usuário criar a biblioteca e esses metadados serão automaticamente exportados para a parte de enriquecimento de metadados.

Omeka

Omeka⁹ é uma plataforma para disponibilização e manutenção de bibliotecas digitais bem flexível, isto é, ele se adapta muito bem para sua finalidade, seja ele museu, biblioteca de imagens, livros, lugares (bibliotecas sobre guerras), entre outros. O Omeka é escrito em PHP¹⁰ e sua instalação é bastante simples, bastando apenas que a máquina do cliente possua o PHP 5 instalado, juntamente com o banco de dados MySQL¹¹ e um servidor HTTP (como o *Apache*¹² ou *nginx*¹³). O Omeka não possui uma finalidade específica como o GreenStone, Dspace, entre outros. Na Figura 6 podemos visualizar que o Omeka caminha na intersecção de três ferramentas, entre elas: Sistema de gerenciamento de conteúdo (*Web Content Management* ou CMS¹⁴), Repositório de arquivos digitais e coleções (biblioteca digital) e sistemas para gerenciamento de museus.

O Omeka possui uma interface muito intuitiva e de fácil personalização, isso se deve ao fato de ser construído para pessoas que não são experientes na utilização de computadores. Diferentemente do GreenStone, o Omeka não possui uma interface offline para inserção de dados, já que todas as funções são realizadas dentro do site.

Sua arquitetura possui o padrão MVC (*Model-View-Controller*)¹⁵. O Omeka também é extensível através de plug-ins, ou seja, é possível estender novos temas ou fun-

⁸ Importado automaticamente sobre o arquivo, extraindo informações como nome, tamanho, data de criação, entre outros

⁹ Disponível em <<http://www.omeka.org>>

¹⁰ Disponível em: <<http://www.php.net>>

¹¹ Disponível em: <<http://www.mysql.org>>

¹² Informações em: <<http://www.apache.org>>

¹³ Maiores Informações: <<http://www.nginx.org>>

¹⁴ Maiores Informações em: <http://pt.wikipedia.org/wiki/Sistema_de_gerenciamnto_de_conte%C3%BAdo>

¹⁵ Segundo Silva (2012) é um padrão que sugere uma arquitetura de software dividida em componentes, viabilizando com clareza o desenvolvimento de um código organizado e enxuto, e posteriormente, a reciclagem e manutenção do sistema sem dificuldade e com segurança. Porém, a independência dos componentes só será atingida se houver uma organização do sistema em camadas para garantir a escalabilidade, eficiência e a reusabilidade

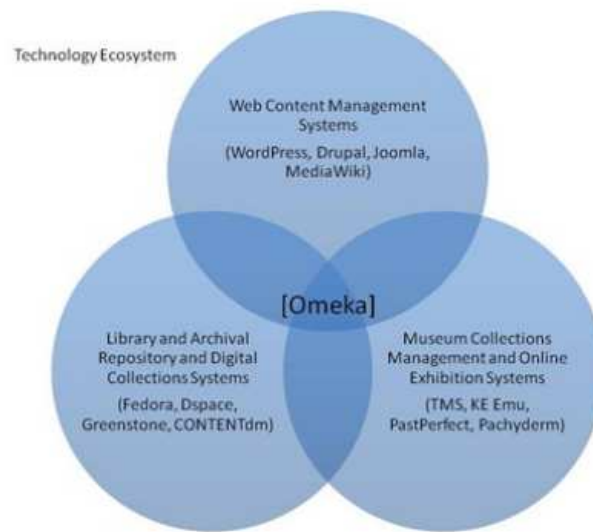


Figura 6 – Objetivo do Omeka. Extraído de <<http://www.omeka.org>>

cionalidades que são desenvolvidas pela comunidade, de forma bastante simples: basta colocar o tema dentro da pasta */themes* (a partir da raiz do Omeka), ou no caso de um *plug-in*, existe uma pasta */plugins*, onde estão localizados todos os *plug-ins*. Após esse passo, basta logar como administrador na interface do Omeka e ativar os *plug-ins* ou o tema. Mais detalhes de extensibilidade é visto na seção 4.2.2.

Uma das vantagens do desenvolvimento de uma arquitetura extensível através de *plugins* é que novas funcionalidades podem ser incluídas sem a necessidade da interferência do desenvolvedor no código fonte do core do Omeka, uma exemplo de uma nova funcionalidade que foi incluída é a compatibilidade com o protocolo OAI-PMH. Para o desenvolvimento de *plugins* e temas, o Omeka fornece toda uma documentação detalhada para o desenvolvimento.

DSpace

DSpace¹⁶ é um software de código fonte aberto que fornece facilidades para o gerenciamento de acervo digital, utilizado para implementação de repositórios institucionais. Suporta uma grande variedade de tipo de documentos, tais como: livros, teses e dissertações, fotografias, filmes, áudio, e outros. Os documentos são organizados em comunidades e coleções.

O DSpace é disponibilizado livremente às instituições de investigação, que pode ser livremente adaptado e expandido funcionalmente, nos termos da BSD *Open source license*. Suas possibilidades de uso incluem: (i) capturar e descrever documentos digitais de acordo com um *workflow* adaptável aos processos específicos de uma comunidade,

¹⁶ Informações adicionais em: <<http://www.dspace.org/>>.

(ii) distribuir os documentos digitais da instituição na Web, possibilitando a pesquisa e obtenção de cópias aos utilizadores, e (iii) preservar os documentos digitais a longo prazo.

O DSpace aceita todas as formas de materiais digitais, incluindo arquivos de texto, imagem, vídeo e áudio, o que possibilita custodiar os mais variados tipos de conteúdos, tais como, livros, artigos, relatórios técnicos, *working papers*, artigos de conferências, e-teses, conjuntos de dados (estatísticos, geoespaciais, etc.), programas de computador, modelos e simulações visuais, etc. Como forma de se adaptar às necessidades específicas de cada instituição e dos seus departamentos, as possibilidades de “customização” do DSpace incluem, não só, a definição de *workflows* “à medida”, mas também, a especificação de regras de utilização e formatos digitais suportados.

O DSpace permite a aplicação de variadas técnicas que garantam a segurança dos documentos digitais submetidos ao repositório. Algumas destas técnicas consistem na realização de cópias de segurança, espelhamento e atualização da infra-estrutura física (i.e., a migração de um suporte físico obsoleto para outro mais atual). Além disso, a cada item é atribuído um identificador único de forma a assegurar a sua recuperação na ocorrência de uma migração de dados.

O repositório do MIT conta com um mecanismo de aconselhamento aos fornecedores de conteúdos para que a documentação depositada seja fornecida nos formatos mais adequados à sua preservação em longo prazo. Os administradores de cada comunidade têm a possibilidade de limitar o acesso aos conteúdos, quanto ao nível do item submetido, quanto ao nível da coleção. Para a pesquisa e recuperação dos itens, o processo de submissão de documentos ao DSpace permite a sua descrição usando uma versão qualificada do vocabulário de metadados Dublin Core.

O DSpace foi desenvolvido em linguagem Java e é suportado por um conjunto de ferramentas de código aberto (*open source*), tais como: PostgreSQL¹⁷, Tomcat¹⁸ e o Lucene (motor de pesquisa)¹⁹. Outros softwares necessários são o Maven ant²⁰ e ant-optionals²¹.

2.5 Considerações

Como visto nessa seção, foram levantados os principais conceitos para entendimento sobre o funcionamento das bibliotecas digitais, assim como as principais ferramentas de bibliotecas digitais candidatas a serem utilizadas na biblioteca digital de participação social. Tendo em vista as ferramentas, baseado nas experiências dos consultores do

¹⁷ Informações em: <<http://www.postgresql.org/>>

¹⁸ Disponível em: <<http://tomcat.apache.org/>>

¹⁹ Maiores Informações: <<http://lucene.apache.org/>>

²⁰ Informações em: <<http://ant.apache.org/>>

²¹ Maiores informações em: <mvnrepository.com/artifact/ant/ant-optional>

Participa.br, além de outros fatores a ferramenta escolhida foi o Dspace. Essa discussão será apresentada no capítulo [5](#).

3 Protocolos de interoperabilidade e disponibilização de informações

Nesse capítulo será discutido as principais formas de interoperabilidade entre os objetos digitais de uma biblioteca digital.

3.1 Iniciativa *Open Archive Initiative* (OAI)

A iniciativa OAI (*Open Archive Initiative*), surgiu a partir em uma Convenção de Santa Fé no Novo México, no final de 1999, com o objetivo de "desenvolver e promover padrões de interoperabilidade que visam facilitar a disseminação eficiente de conteúdo."(OAI)

A iniciativa OAI desenvolveu padrões de interoperabilidade e novas tecnologias que permitiram uma maior facilidade no acesso aos arquivos presentes nos repositórios digitais, principalmente na área acadêmica, como por exemplo: revistas acadêmicas, dissertações, artigos, entre outros. Essas tecnologias e padrões foram essenciais na comunicação acadêmica, pois devido a eles, os usuários podem ter acessos conteúdos disponibilizados em outros repositórios.

Kuramoto (2014) afirma que a iniciativa OAI procura democratizar o acesso à informação, já que facilitando-o através da concepção de novos repositórios livres, permitindo que qualquer pessoa possa acessar as informações que estão nestes repositórios.

Uma das maiores contribuições da iniciativa OAI foi a elaboração do protocolo OAI-PMH, que faz com que os participantes desta possam compartilhar seus metadados com outros repositórios. No prosseguimento desse capítulo é apresentado os principais conceitos para entendimento desse procolo, que será utilizado para integrar a biblioteca digital de participação social com o Participa.br.

3.1.1 Provedor de dados

Os provedores de dados mantêm repositórios de arquivos digitais que implementam o protocolo OAI-PMH como forma de expor os metadados de seus metadados. Eles podem oferecer uma interface para os usuários, disponibilizando acesso a seus documentos e mídias digitais, no entanto, não é obrigatório.

Atualmente, a iniciativa OAI mantém um repositório¹ com as principais entidades

¹ Disponível em: <<http://www.openarchives.org/Register/BrowseSites>>

que fornecem repositórios digitais compatíveis com o protocolo OAI-PMH. Até a data de escrita deste trabalho, já estavam cadastradas cerca de 2458 entidades. Vale lembrar também, que as entidades que se cadastram para serem listadas por esse repositório, tem que ser compatíveis com a versão 2.0 do protocolo OAI-PMH, que será visto mais abaixo.

3.1.2 Haverster

São aplicações ou softwares que utilizam a interface modelada pelo protocolo OAI-PMH para coletar metadados (OLIVEIRA, 2009). Eles coletam todos os metadados disponibilizados pelos Provedores de Dados e importam para os Provedores de Serviços, que podem disponibilizá-los na forma de novos serviços, utilizando os metadados coletados como insumo.

3.1.3 Provedor de serviços

Os provedores de serviços utilizam *haversters* para realizar a colheita dos metadados expostos pelos provedores de dados. Os provedores de serviços realizam requisições utilizando o protocolo OAI-PMH para provedores de dados e usa os metadados coletados como base para a construção de serviços de maior valor agregado, com uma interface mais amigável para os seus usuários (OLIVEIRA, 2009).

Na imagem ilustrada pela Figura 15, é representado o funcionamento dos provedores de serviços dentro do protocolo OAI-PMH. Nessa representação, o provedor de serviço requisita através de uma requisição HTTP, todos os metadados de seus provedores de dados, esses retornam a essa requisição devolvendo todos os seus metadados através do formato XML.



Figura 7 – Funcionamento dos provedores de serviços. Extraído de (OLIVEIRA, 2009)

Através dessa representação, os provedores de serviços podem oferecer buscas mais refinadas estes metadados, além de disponibilizá-los em formas de outros serviços, agregando mais valor ao repositório.

De acordo com o repositório da iniciativa OAI (*Registered Service Providers*²), o número de provedores de serviços é bastante reduzido, fato esse, que faz com que os provedores de serviços sejam pouco divulgados.

3.1.4 Protocolo OAI-PMH

O protocolo OAI-PMH (*Open Archives Initiative Protocol for Metadata Harvesting*) foi lançado no primeiro semestre de 2001. A versão atual, 2.0, foi disponibilizada em julho de 2002. Ele se consolidou como o principal protocolo utilizado em bibliotecas e repositórios digitais em todos os lugares do mundo, isso foi possível já que o OAI-PMH permite proporcionar visibilidade e integração de informações, com custos acessíveis à realidade de países em desenvolvimento³, como é o caso do Brasil (GARCIA; SUNYE, 2003).

O OAI-PMH possui algumas vantagens, entre elas: (i) o protocolo fornece o suporte para ser colhido pelo Google, Bing, Yahoo, entre outros, permitindo uma maior visibilidade para o repositório e (ii) existem repositórios que podem ser importados para a disponibilização em contextos mais específicos, por isso o OAI-PMH é amplamente utilizado no mundo acadêmico.

Com base nesse protocolo, as requisições são obtidas através de uma requisição HTTP, contendo um *timestamp* (data-hora) do momento da requisição, e caso seja necessário ela pode ser restringida a conjuntos definidos⁴ pelo requisitante, desta forma cada requisição vai exigir somente o mínimo de dados vindo do repositório, diminuindo o tráfego de rede oriundo da transmissão de metadados. No envio das respostas, os provedores de dados são obrigados a fornecer metadados no formato XML, além disso o solicitante pode pedir diversos esquemas de metadados, desde que o repositório ofereça suporte⁵. A imagem ilustrada pela Figura 8 representa o esquema de comunicação realizada no Dublin Core.

² Disponível em: <<http://www.openarchives.org/service/listproviders.html>>

³ O protocolo OAI-PMH não exige custos elevados com o desenvolvimento de tecnologias, assim como uma estrutura física de alto desempenho que possam manter toda a infraestrutura.

⁴ Também chamado de *sets*, são agrupamentos de artefatos digitais relacionados (contendo o mesmo assunto)

⁵ Os metadados mais utilizados no protocolo OAI-PMH são os Dublin Core, pois oferecem validação durante a resposta em XML

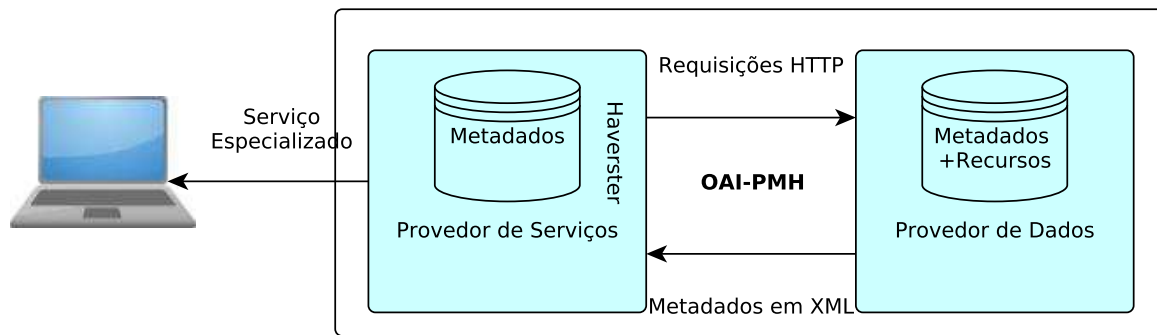


Figura 8 – Arquitetura de funcionamento do OAI-PMH.

Adaptado de <<http://en.wikipedia.org/wiki/OAI-PMH>>

O protocolo OAI-PMH não foi projetado para suportar consultas e/ou pesquisas de seus usuários. Caso o usuário tenha o objetivo de obter registros por autor, assunto, ou outras palavras chaves, ele não vai obter sucesso. Em vez disso, o protocolo consiste em requisições e respostas que identificam e descarregam todos os registros de metadados de uma vez só.

Através de um conjunto de verbos especificados, o OAI-PMH permite que um *haverster* possa interagir com um provedor de dados. O objetivo desses verbos é obter todos os metadados vindos dos provedores de dados na primeira solicitação, e incluir novos registros e alterações feitas nos provedores de dados nas colheitas subsequentes. Com base nesses verbos é possível identificar os tipos de metadados fornecidos e suportados por cada provedor de dados, além de fornecer parâmetros opcionais como por exemplo, timestamps, que ajudam a realizar a obtenção de registros seletivo.

De fato, apesar de parecer complexo, os seis verbos especificados no protocolo OAI-PMH mantêm sua simplicidade. Os interessados em uma explicação detalhada sobre os verbos, respostas, além dos parâmetros de requisição de cada verbo, o site da iniciativa OAI possui toda a documentação detalhada⁶. Em termos gerais, os verbos são os seguintes:

- **Identify:** Recupera as informações gerais do provedor de dados, entre elas: nome do provedor de dados, *URL* ou *site* do repositório, versão do protocolo OAI-PMH, *e-mail* do mantenedor do repositório.
- **GetRecord:** Recupera um único registro do repositório, bastando apenas informar o tipo de metadado, além do *oai_dc* (Dublin Core) e o identificador do registro (*Identifier*) do objeto a ser obtido;

⁶ Disponível em: <<http://www.openarchives.org/OAI/2.0/openarchivesprotocol.htm>>

- **ListMetadataFormats:** Recupera os formatos de metadados suportados e fornecidos no repositório (geralmente o padrão é o Dublin Core: `oai_dc`);
- **ListRecords:** Responsável por fazer o *harvesting* (colheita) dos metadados do repositório, ele utiliza alguns parâmetros opcionais, onde pode-se obter os registros baseada na data (timestamp) ou em conjunto de assuntos relacionados (sets). Necessário informar o tipo do metadado;
- **ListIdentifiers:** É um verbo que realiza uma versão abreviada do verbo ListRecords, retornando apenas os cabeçalhos dos registros; e
- **ListSets:** É um verbo que trás a respostas de todos os grupos que possuem objetos digitais com assuntos relacionados.

Vários softwares são compatíveis com o protocolo OAI-PMH, incluindo o GreenStone, Dspace, Fedora, GNU E-prints, entre outros. No caso de softwares que não são compatíveis com o protocolo, pode-se desenvolver soluções a partir de *plug-in*, como é o caso do Omeka e transformá-lo em um provedor de serviços.

O Instituto Brasileiro de Informação em Ciência e Tecnologia (IBICT) é o principal incentivador para a adoção do protocolo OAI-PMH pelas instituições brasileiras. O IBICT também incentiva no uso da ferramenta de biblioteca digital DSpace como ferramenta padrão para criação de bibliotecas digitais, além de realizar alguns estudos que ajudam na criação de repositórios nacionais (OLIVEIRA, 2009).

3.2 Linked Data

Nesta seção, falaremos sobre o conceito de *Linked Data*, que é uma idéia nova introduzida por Tim Berners-Lee que busca formas de estruturar e apresentar os dados na Web, utilizando alguns conceitos já documentados pela Web Semântica.

Com o crescimento exponencial da Web tornou-se difícil indexar as informações. A linguagem HTML, que popularizou a Web e é utilizada pela maioria dos sites, não possui recursos que permitam atribuir significado à informação.

A Web Semântica, que é visto por muitas pessoas como o futuro da internet (Web 3.0), tem o objetivo de buscar uma internet mais robusta através do conhecimento conectado, dessa maneira os dados significativos e informações podem ser legíveis e inferidas através de pessoas e máquinas. Imaginado por Tim Berners-Lee (2001)⁷, o seu objetivo é fazer a transição de uma web de documentos e arquivos não estruturados em uma "teia de

⁷ Disponível em: <http://esw.w3.org/topic/SweoIG/TaskForces/CommunityProjects/LinkingOpenData>

dados", na qual se pode navegar e pesquisar dados não apenas com base em palavras-chave sintáticas, mas também com base em significado semântico.

Existem quatro princípios fundamentais do Linked Data definidos por Lee (2009), que são:

1. Utilizar URIs⁸ para nomear as coisas (documentos, arquivos, mídias, etc.);
2. Utilizar URIs fornecidas por HTTP, para que os usuários possam encontrar os recursos com mais facilidade;
3. Quando alguém acessar o URI fornecido, trazer informações úteis utilizando os padrões já documentados (SPARQL ou RDF); e
4. Incluir acesso (*links*) para outros URIs, para que os usuários possam encontrar diversos assuntos.

3.2.1 RDF - Resource Description Framework

Através da utilização do Linked Data, são utilizados alguns padrões e modelos de linguagens como por exemplo: o Resource Description Framework (RDF) e o Web Ontology Language (OWL), ambos fornecem uma base para a criação de vocabulários que podem ser usados para descrever as entidades no mundo e como eles estão relacionadas (MCGUINNESS, 2004). O RDF é tido como uma sigla comum dentro da comunidade web semântica porque forma um dos blocos de construção básicos para a formação da rede de dados semânticos.

O objetivo geral na utilização do RDF é a descrição dos dados e dos metadados por meio de “triplas”. Essas triplas informam os recurso-propriedade-valor (ou também sujeito-predicato-objeto) de um determinado artefato, gerando uma forma mais coesa para catalogação e acesso aos padrões de metadados. Vocabulários ou *namespaces* são conjunto de elementos disponibilizados através de uma IRI (*Internationalized Resource Identifier*), que possui uma coleção de classes, além de suas propriedades utilizada para descrever objetos. No exemplo 3.1 é representado um trecho de código estruturado em RDF (DING, 2005).

Listing 3.1 – Exemplo de código RDF

```
<?xml version="1.0" encoding="utf-8"?>
  <rdf:RDF xmlns:contact="http://www.w3.org/2000/10/
    swap/pim/contact" xmlns:eric="http://www.w3.org/
    People/EM/contact#" xmlns:rdf="http://www.w3.org
    /1999/02/22-rdf-syntax-ns#">
```

⁸ Identificador Uniforme de Recursos (URI) é um conjunto de caracteres utilizado para identificar recursos pela internet

```
<rdf:Description rdf:about="http://www.w3.org/People/EM/contact#me">
  <contact:fullName>Eric Miller</contact:
    fullName>
</rdf:Description>
```

Na abordagem acima, vemos que o autor especifica o *Namespace* já na primeira linha (<http://www.w3.org/2000/10/swap/pim/contact>), esse *Namespace* contém os metadados que vão ajudar a descrever o elemento em questão. Após isso, temos a descrição de um dos recursos (sujeito) do *Namespace*, que no caso é o recurso 'me', que especifica os dados de uma pessoa. Para especificar uma propriedade, o autor seleciona uma das propriedades (predicado) daquele recurso, que no caso é o metadado *contact:fullName* (nome completo) adicionando um valor (objeto) de Eric Miller para aquela propriedade.

É importante ressaltar que um valor pode ter várias propriedades de vários recursos diferentes, criando uma rede de grafos que criam uma rede maior de significados para aquele valor, ajudando os softwares a indexarem melhor o mesmo.

Uma das vantagens do Linked Data é que qualquer um é livre para publicar o seu vocabulário (BERRUETA; PHIPPS, 2008), que por sua vez pode ser conectado a outros vocabulários existentes de forma que os softwares e máquinas possam dar sentido as informações inferindo de forma semântica, e conseqüentemente tratando-os de maneira mais inteligente (SOUZA; ALVARENGA, 2004).

Com base na triplificação de RDF, as classes e propriedades podem ser ligadas de um vocabulário para os de outro, definindo assim mapeamentos entre vocabulários relacionados.

3.2.2 Ferramentas para armazenamento de RDF

As ferramentas de armazenamento de triplas RDF⁹, também chamadas de *triple store* possuem estruturas, que podem ser utilizadas para armazenar e consultar dados RDF. As ferramentas fornecem mecanismos para armazenamento persistente e acesso de grafos RDF.

Foram levantadas algumas ferramentas candidatas que poderão ser utilizadas para o armazenamento dos RDFs oriundos da triplificação dos dados sobre os mecanismos formais de participação, no entanto é importante ressaltar que todas as ferramentas são livres e seus códigos são disponibilizados para que a comunidade possa evoluir.

⁹ Essas ferramentas seriam sistemas de gerenciamento de banco de dados no caso de armazenamento de dados comuns

Apache Jena

O Apache Jena¹⁰ é um *framework* desenvolvido em Java para construção de aplicações utilizando conceitos da web semântica. Usando JDBC¹¹, ele também pode ser vinculada a vários bancos de dados relacionais existente, como MySQL ou PostgreSQL. O Jena fornece suporte para o armazenamento de RDF, permitindo inferências e consultas a essas bases de dados RDF usando um banco de dados SQL (facilitando o desenvolvimento das aplicações).

Virtuoso

Virtuoso¹², é uma triple store disponível em licença aberta, mas também existe uma versão fechada (comercial). A versão aberta do Virtuoso, também chamada de OpenLink Virtuoso fornece suporte a uma API de conexão, que envia os dados através de HTTP, SOAP, além de vários outros formatos. Ele suporta execução de consultas em SPARQL juntamente com SQL para consultar dados RDF armazenados dentro do banco de dados.

Sesame

O Sesame¹³ é um *framework* para o armazenamento de triplas RDF, podendo fazer inferências e consultas a uma base de dados RDF. Ele combina as mesmas características encontrada no Apache Jena fornecendo uma API para conexão de maneira mais simplificada.

O Sesame também fornece um motor para realizações de inferência de RDF, além de disponibilizar um servidor web e *endpoint* SPARQL. Assim como os demais sistemas, o Sesame fornece suporte para diversos servidores como MySQL, PostgreSQL, MongoDB, CountDB, entre outros, além de possuir suporte a diversas linguagens, o que facilita no desenvolvimento de novas aplicações. O Sesame também possui suporte para trabalhar com outros sistemas de armazenamento de RDF.

Triplify

O Triplify¹⁴ fornece suporte para construção de aplicativos Web "semantificados". A diferença do Triplify para as demais soluções é que ele um plugin para aplicativos Web. Desse modo, o triplify converte todo o banco de dados relacionais de aplicação em estru-

¹⁰ Informações em: <<https://jena.apache.org/>>

¹¹ Java Database Connectivity ou JDBC é uma API utilizada em códigos em Java que fazem o envio consultas SQL para qualquer banco relacional. Maiores informações em: <<http://jdbc.postgresql.org>>

¹² Disponível em: <<http://virtuoso.openlinksw.com>>

¹³ Maiores informações em: <<http://rdf4j.org/>>

¹⁴ Disponível em: <<http://triplify.org/>>

turas semânticas codificadas, disponibilizando todo o conteúdo da aplicação disponível como RDF, JSON ou Linked Data.

Uma das vantagens no uso do Triplify é que ele é muito leve (cerca de somente 500 linhas de código), fato esse por se tratar de um *plug-in*. Ele é originalmente desenvolvido em PHP, mas existem variações desenvolvidas por terceiros para Ruby, Java, entre outros. Devido a sua rápida configuração, em cerca de poucas horas é possível "semantificar" a aplicação.

O Triplify transforma as aplicações Web "normais" em "semântica" de maneira mais fácil que as demais opções, oferecendo suporte para formato de bases de próxima geração (RDFa), e possui suporte para semântica baseada em pesquisas na Web.

3.3 Considerações

[Souza e Alvarenga \(2004\)](#) mostram que por meio das ontologias e dos metadados utilizados, compartilhados e validados entre comunidades, é possível a criação de novas metodologias para análise e catalogação de documentos automaticamente, permitindo classificá-los de maneira automática ou semi-automática. Dessa forma é importante utilizar a abordagem de catalogação dos dados dos mecanismos formais de participação em RDF, pois assim será possível ter documentos que interessem os seus usuários.

A partir da triplificação dos dados é necessário guardá-los em uma Triple Store, e dentre as ferramentas utilizadas, o Virtuoso se tornou a melhor opção por oferecer uma maior integração com a linguagem Ruby, através de uma biblioteca na linguagem Ruby chamada RDF-Virtuoso, que oferece várias funções já pré estabelecidas, que ajudam no desenvolvimento da triplificação. Esse assunto será tratado melhor no capítulo 5.

4 Plataforma de Participação Social

Neste capítulo, será apresentado o Participa.br - Plataforma Federal da Participação Social é um espaço colaborativo para participação social, escuta das demandas do povo e um espaço de dialogo da sociedade com o governo. Ele será utilizado como um case para o desenvolvimento do plug-in. A imagem ilustrada pela Figura 9 apresenta a página inicial do Participa.br.



Figura 9 – Tela Inicial do Participa.br.

4.1 Participa.br

O Participa.br tem como um dos seus principais objetivos, a criação de um canal de comunicação e participação entre cidadãos e gestores públicos. O Participa.br é uma ação do Gabinete Digital, que é uma iniciativa online do Governo Federal para ampliar o acesso do cidadão à informação pública, serviços, prestação de contas e participação popular nas decisões (DIGITAL, 2013). O Gabinete Digital foi anunciado pela presidenta Dilma Rousseff como resposta às manifestações ocorridas em todo território brasileiro em

junho do ano de 2013, e uma de suas principais funções é realizar uma comunicação direta com a sociedade através das redes sociais.

No Participa.br, o cidadão pode se juntar a comunidades com temáticas do seu interesse, como por exemplo, educação, saúde, inclusão digital, entre outras. O Participa.br, permite também que participante crie uma comunidade com o tema de seu interesse, caso não exista. Isso é importante para que os usuários possam utilizar a ferramenta para debater temas que lhe interessem, isto torna a ferramenta mais atrativa. Para isso basta apenas que o usuário crie uma comunidade e alguém que administre ou modere a criação de comunidades na ferramenta do Participa.br aprove a mesma.

Uma das características do Participa.br é servir, além de um espaço de consulta pública, também como uma ferramenta para formulação de novas políticas públicas. Quem acessava a página inicial Participa.br durante os dias 19 de março de 2014 a 17 de abril de 2014, participava de um questionário para definição sobre os direitos e princípios fundamentais para garantir o futuro democrático e orientar a governança na internet. O participante respondia 3 perguntas, cada uma com um par de propostas, sem limites de respostas. A cada pergunta o usuário votava na opção que se concordava mais, caso não houvesse nenhuma proposta que ele mais se identificasse ele poderia sugerir novas propostas. O resultado dessa consulta foi enviado como uma Carta Proposta para o Comitê Gestor da Internet, o qual era o gestor do Arena NET Mundial, um evento realizado durante os dias 22 a 24 de abril de 2014 na cidade de São Paulo, com o objetivo de discutir ideias para garantir uma internet livre, colaborativa e democrática ¹.

Quando o usuário acessa o Participa.br, ele só pode acessar as ferramentas de participação social, se o mesmo estiver registrado na plataforma. O registro é bem simples, necessitando apenas alguns dados como nome de usuário, senha, nome, *e-mail*, entre outros. Após o registro um e-mail é enviado para o usuário, este contém o link para ativação do usuário no Participa.br (para evitar criação de spans de usuário). Após a ativação o cidadão já pode fazer comentários e contribuições na plataforma, personalizar o seu perfil, divulgar sua opinião em consultas públicas, manter o seu *blog*, participar e/ou criar comunidades, etc.

4.2 Noosfero: A Plataforma Utilizada no Participa.br

A Secretaria Geral da Presidência da República ²(SGPR) por meio do projeto Participa.br, buscava criar um legado dentro de um software, e ser pioneira em relação a evolução de um software livre por uma entidade governamental. Durante o período de escolha da ferramenta que seria o núcleo do portal de Participação Social, foi procurado

¹ Maiores informações em: <<http://www.participa.br/arena>>

² Disponível em: <<http://www.secretariageral.gov.br/>>

uma ferramenta que além de satisfazer as necessidades demandadas para a construção do Participa.br, também seguisse os princípios do Software Livre (visto na seção ??). Ou seja, na medida que o Participa.br vai sendo desenvolvido, sua plataforma de desenvolvimento, o Noosfero (discutido no seção 4.2 também acompanha esse desenvolvimento, algo que ainda é pioneiro nas esferas governamentais. No Participa.br esse fato é importante pois as contribuições realizadas na ferramenta, ajudam no desenvolvimento tanto do Participa.br, como da ferramenta em si utilizada.

O Noosfero é uma plataforma livre para criação de redes sociais, que está disponível sob licença AGPL³ V3. O Noosfero tem como objetivo permitir a criação de uma rede social por parte de um usuário, onde o mesmo pode modificá-lo e personalizá-lo de acordo com suas necessidades. No entanto, o Noosfero também pode ser adaptado para ser utilizado como um Sistema de Gerenciamento de Conteúdo (CMS), ou até mesmo funcionando em portais de economia solidária.

O Noosfero é escrito na linguagem de programação *Ruby* versão 1.9.2 e utiliza *framework* para escrita de aplicações web *Ruby on Rails* versão 3.2. Utiliza também os padrões arquiteturais de software *Model-View-Controller* (MVC), que foi discutido na seção 2.4.

4.2.1 Arquitetura do núcleo

Detalhar alguns dos padrões arquiteturais utilizado no Noosfero é importante para um maior entendimento sobre os elementos essenciais de sua arquitetura e como eles se relacionam, ajudando a evoluir o software de maneira correta, deixando de lado detalhes desnecessários e irrelevantes. Sendo assim, entender a arquitetura de funcionamento do Noosfero é um dos passos iniciais para evolução do Participa.br, e conseqüentemente atingir os objetivos desta monografia.

Na Figura 10, é apresentado a arquitetura básica de funcionamento do Noosfero. Ainda nessa Figura, o solicitante através do seu navegador *web* (*browser*) envia uma requisição ao Noosfero. Essa requisição é tratada pelo Varnish⁴, que repassa ao Apache⁵, que funciona como um redirecionador para o servidor *web* do Rails, chamado de Thin⁶. Após isso o *Action Dispatcher* trata a requisição e envia ela para a controladora responsável.

Como o Noosfero foi desenvolvido em 2007, o esquema de funcionamento de classes possui uma pequena influência da rede social que havia mais usuários na época, o Orkut⁷. Podemos citar como exemplo dessa influência, a aparência e disposição a disposição de

³ Licença de software GNU Affero General Public License

⁴ Maiores informações em: <<https://www.varnish-cache.org/>>

⁵ Disponível em: <<http://www.apache.org>>

⁶ Disponível em: <<https://github.com/macournoyer/thin>>

⁷ Disponível em: <<http://www.orkut.com.br>>

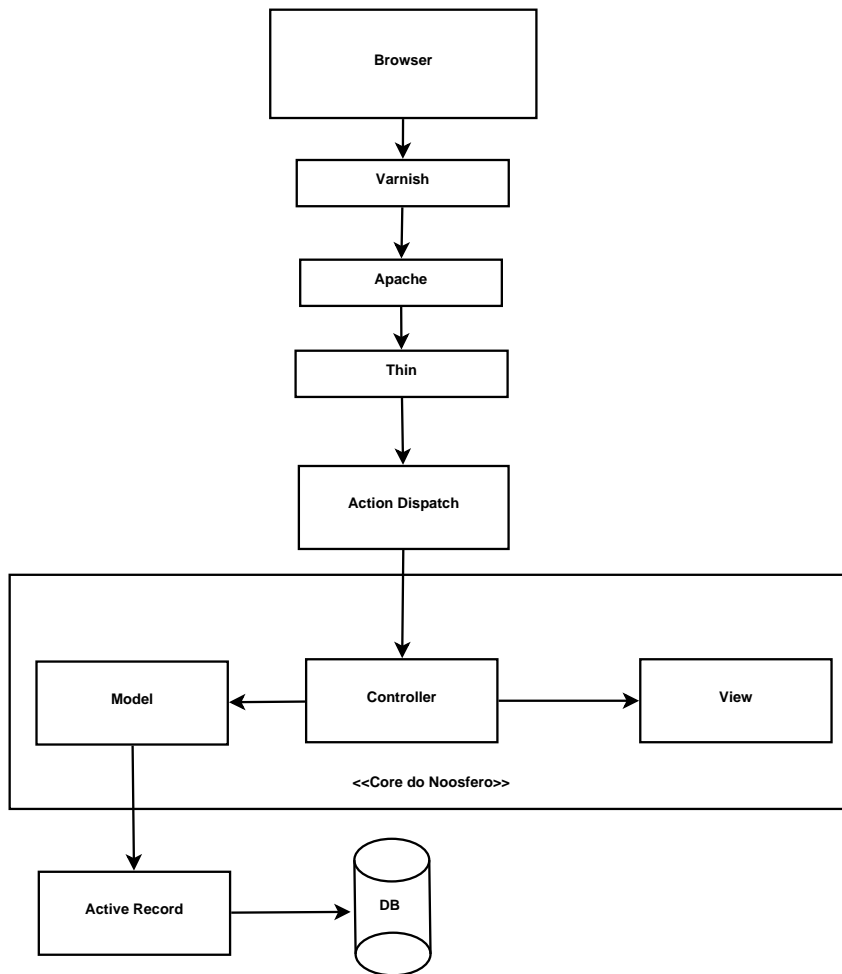


Figura 10 – Arquitetura de funcionamento do Noosfero

alguns botões, o esquema de comunidade, funcionamento de perfis, entre outros. Na Figura 11, é explicado o funcionamento geral do Noosfero.

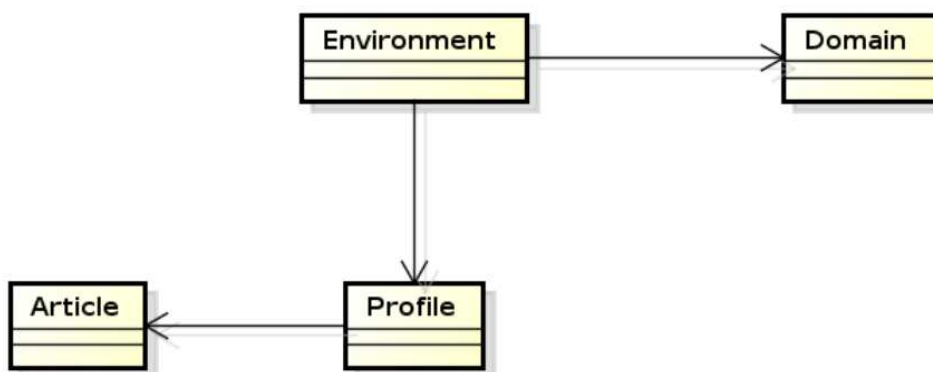


Figura 11 – Relacionamentos entre perfis, ambientes e domínios

Através do suporte múltiplos ambientes dentro de uma única instalação fornecido pelo *Ruby on Rails*, é possível ter um ou mais ambientes de rede social utilizando uma

única instalação do Noosfero, basta que o ambiente criado tenha um domínio (*Domain*) diferente do primeiro ambiente. No Noosfero, existe uma entidade chamada Perfil (*Profile*), essa entidade é uma abstração de outras três outras entidades: Pessoas (*Person*), Comunidade (*Community*) e Empreendimento (*Enterprise*), como é visto na Figura 12. A abstração consiste isolar elementos em comum dentro de uma classe pai, enquanto as especialidades são responsabilidade das classes filhas.

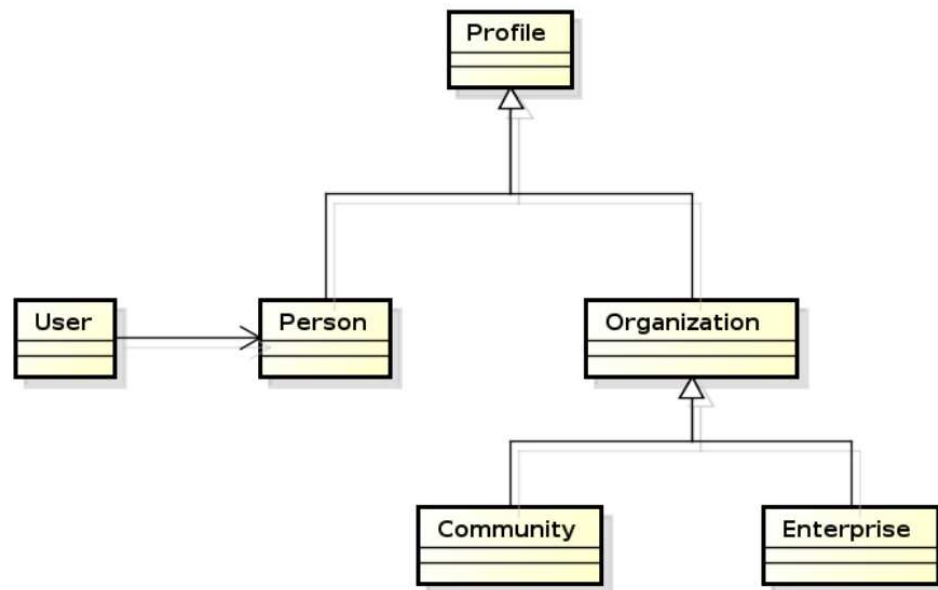


Figura 12 – Relacionamento entre os tipos de perfis

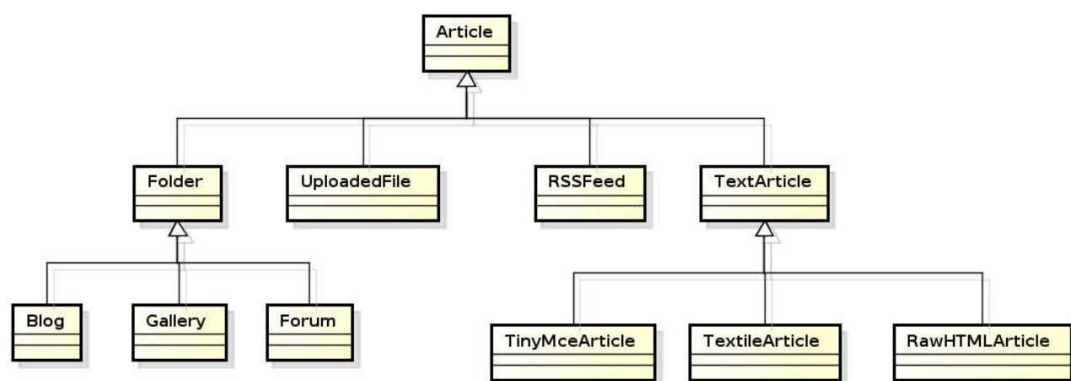


Figura 13 – Relacionamento entre os tipos de conteúdo do Noosfero

Ainda na Figura 12 é encontrada outra abstração no código do Noosfero acontece na classe chamada de organização, que possui o comportamento comum entre comunidades (*community*) e (*enterprise*). A classe usuário (*user*) possui todas as lógicas e implementações relativas a autenticação do usuários, enquanto a classe Person possui lógicas relacionadas a rede social (amigos, manutenção de perfil, etc.), tornando o código mais fácil de ser entendido. Na Figura 13 é representado os tipos de conteúdo disponíveis no

Noosfero como artigos de texto, fóruns, pastas, artigo usando HTML, *blogs* de conteúdo, galerias de imagens, arquivos em geral e feeds de notícias.

4.2.2 Arquitetura de *plug-ins*

O Noosfero também utiliza uma arquitetura extensível através do uso de *plug-ins*⁸. Essa arquitetura é interessante, pois como dito anteriormente, o Noosfero implementa a arquitetura de múltiplos ambientes presente no arcabouço *Ruby On Rails*. Cada um desses ambientes possuem comportamentos que requerem implementações diferentes.

Essa arquitetura encoraja a criação de um software modular, ou seja, a criação de módulos específicos para cada função diminuindo o acoplamento, aumentando a coesão e definindo bem as responsabilidades de cada módulo (ROZANSKI; WOODS, 2005).

Uma das grandes vantagens em se criar uma aplicação com arquitetura extensível através de *plug-ins*, é o fato de o desenvolvedor criarem novas funcionalidades sem ter acesso ao código fonte do software, apenas é necessário somente a API⁹ que é fornecida pelo fabricante do software. Apesar disso no Noosfero os plugins são mantidos na linguagem *Ruby*, através do arcabouço *Ruby on Rails* e acoplados juntamente com o código principal (núcleo), dentro de uma pasta “*plugins*”, isso ajuda no controle de qualidade do código. Quando houver a necessidade de alterar o código do Noosfero, os testes dos *plug-ins* são executados para verificar se as mudanças não afetaram seu funcionamento¹⁰.

O funcionamento dos *plug-ins* é inspirado no paradigma de orientação a eventos, onde núcleo do Noosfero dispara um evento durante a execução de alguma função, e os *plug-ins* interessados nesse evento vão sobrescrever o método do core do Noosfero tratando o evento de acordo com sua implementação, funcionando como o padrão de projeto *Composite*. Os eventos que são chamados pelo pelo Noosfero e implementados em outras partes são chamados de “*hotspots*”. A Figura 14 resume esse funcionamento.

⁸ Um plugin é um software utilizado para incrementar funcionalidades a outros programas maiores. Maiores informações em: <<http://pt.wikipedia.org/wiki/Plug-in>>

⁹ Acrônimo *Application Programming Interface*, que são padrões definidos por um software para utilização de seus recursos sem a necessidade de entrar na implementação do código fonte do software

¹⁰ Maiores informações em: <<http://noosfero.org/Development/Plugins>>

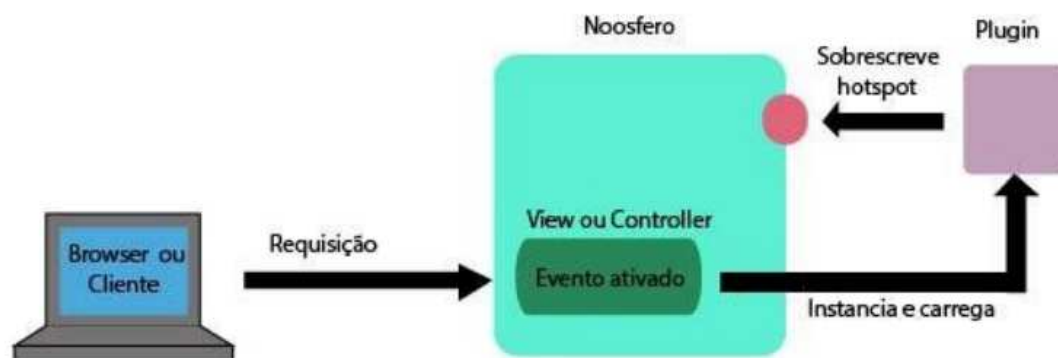


Figura 14 – Funcionamento dos *plug-ins*. Extraído de Meirelles et al. (2014)

4.3 Considerações

Como visto no início desse capítulo, o Participa.br será utilizado como um *case* desse projeto. No entanto é importante ressaltar que até o presente momento da escrita desse trabalho, a biblioteca digital de participação social não estava disponível, logo os testes foram feitos em outros repositórios OAI-PMH de bibliotecas disponibilizados dentro do repositório de provedores de serviços da iniciativa OAI¹¹.

Tendo em vista que o desenvolvimento de novas funcionalidades podem alterar totalmente o funcionamento do Noosfero, é interessante que a solução desenvolvida neste trabalho não cause impacto dentro das redes que utilizam esse software. Através do estudo dos *plug-ins*, que trás uma imagem da arquitetura do núcleo Noosfero é possível o desenvolvimento de novas funcionalidades sem impactar dentro do mesmo, fato esse que foi decisivo na escolha do desenvolvimento ser em *plug-in*, que será visto no capítulo 5.

¹¹ Maiores informações em: <<http://www.openarchives.org/Register/BrowseSites>>

5 Conversão de dados Dublincore para RDF

Neste capítulo será mostrado como foi feito a integração entre a biblioteca digital de participação social, além da arquitetura escolhida para a comunicação, mostrando os passos para desenvolvimento do *plug-in* OAI-PMH para o Noosfero.

5.1 A biblioteca digital de participação social

Para entender como funciona os mecanismos de participação, foi feito e um levantamento (catálogo) inicial das informações sobre os principais mecanismos formais de participação social ¹, dentre eles, foram escolhidos conselhos, conferências e ouvidorias, a fim de verificar os principais problemas enfrentados. Após isso foi verificado que (i) a falta de atualização dos conteúdos dos mecanismos formais; (ii) desconhecimento das pessoas em relação aos mecanismos formais; (iii) falta de padronização em relação aos documentos, entre outros eram os principais problemas relacionados a esses mecanismos.

Dentre esses problemas que foram citados, o principal deles é que encontrar as informações referentes sobre os principais conselhos é uma tarefa bastante dispendiosa, já que os mecanismos formais de participação não disponibilizam informações atualizadas ou até mesmo não disponibilizam suas informações. No site do Conselho Nacional de Saúde² há links para os documentos sobre as atas de reuniões, deliberações, calendários, legislações, assim como todos os seus dados institucionais como membros, estruturas, dentre outras, enquanto o Conselho Curador do FGTS ³ possui apenas uma página com seus membros, sem informações de documentos, resoluções, entre outros. Esses fatores além de ocasionar, pode estar relacionado com o pouco interesse público nesses artefatos.

Pensando em mitigar isso, foi realizado através das consultorias do PNUD (Termo de Referência BRA/12/018), principalmente na de Ciência da Informação, uma proposta para criação de uma biblioteca digital, contendo os principais artefatos disponibilizados pelos mecanismos formais de participação social. Com base na experiência dos consultores, análise de repositórios existentes, além das recomendações dadas pelo IBICT, foi escolhido a ferramenta Dspace para a construção desta biblioteca digital. Nessa biblioteca, será catalogado os principais artefatos relacionados aos mecanismos formais de participação social, apresentando-os de forma mais organizada para os usuários.

No entanto, no ponto de vista do usuário leigo, não é interessante disponibilizar

¹ Disponibilizado no produto 01 da consultoria de Ciência da Informação de Cruz e Meirelles (2014). Informações em: <<https://docs.google.com/file/d/0Bz8tVudv1bR6T0ZrTWRxclJZM0k>>

² Disponível em: <<http://conselho.saude.gov.br/>>

³ Informações em: <http://www.fgts.gov.br/quem_administra.asp>

outra ferramenta para acesso das informações sobre os mecanismos formais, além de que, ferramentas como Dspace possui uma interface um pouco complicada, principalmente quando não foi realizado um bom trabalho de documentação da arquitetura de informação (o que não foi o caso da biblioteca do Participa.br). Pensando também em uma maneira de ajudar as pessoa que serão responsáveis por manter as informações referente sobre os mecanismos formais de participação, já que na maneira atual, é feito cadastro dos artefatos dentro da biblioteca de participação social e depois dentro do Participa.br, gerando trabalho duplo.

A partir disso, pode-se utilizar de toda arquitetura de informação desenvolvida e disponibilizada pelo Participa.br, assim como seus serviços e ferramentas de participação. Com base nisso, é possível disponibilizar através do Participa.br serviços especializados, abrangendo todos os objetos digitais das diversas bibliotecas dos mecanismos de participação social, agregando maior valor aos objetos digitais e conseqüentemente promovendo esses canais de participação, incentivando a população a participar socialmente das ações tomadas pelo governo.

5.2 Escolha do protocolo de interoperabilidade

Para transformar o Participa.br em um provedor de serviços, é necessário que exista uma maneira de interoperar⁴ o Noosfero, e a ferramenta de bibliotecas digitais Dspace. Mas como visto na seção 4.2.2, a arquitetura do Noosfero permite que novas funcionalidades sejam inseridas dentro de sua arquitetura através de plug-ins, sem que alterem o núcleo (*core*) da aplicação.

A escolha dessa implementação ser feita em um plugin no Noosfero, deve-se ao fato da necessidade de transformar o Participa.br em um repositório de objetos digitais oriundos de uma biblioteca digital seja exclusiva do Participa.br neste primeiro momento, não obrigando o Participa.br, além de outras instâncias que utilizam o Noosfero a utilizarem essa intercomunicação. Outro fator a ser considerado é que nada impede de outras instâncias utilizem esse plug-in em outras ocasiões futuramente.

Analisando as alternativas possíveis de se realizar essa interoperabilidade, foram encontradas as seguintes soluções:

- **Acesso direto ao banco de dados** - Deste modo, o Noosfero consultaria toda a base de dados do Dspace⁵ buscando todos os objetos digitais.

⁴ Capacidade de dois ou mais componentes de software cooperarem, a despeito das diferenças de linguagens, interfaces e plataforma de execução (WEGNER, 1996)

⁵ É importante ressaltar que tanto o Noosfero como o Dspace usam PostgreSQL como sistema gerenciador de banco de dados

- **WebService** - Através da API de REST⁶ fornecida pelo Dspace, é possível obter comunidades, coleções, itens e bitstreams (arquivos).
- **Z39-50** - É um protocolo de comunicação entre cliente/servidor designado para permitir a recuperação de documentos, dados bibliográficos, imagens, entre outros arquivos multimídia.
- **OAI-PMH** - Como visto na seção 3.1.4, o OAI-PMH é um protocolo cliente/servidor que utiliza seis verbos para colheita dos metadados de arquivos, textos, imagens, vídeos, entre outros.

Baseado nas três alternativas, o protocolo OAI-PMH se mostra a melhor e mais flexível opção dentre as três citadas anteriormente.

De acordo com que foi visto na seção 3.1.4, o OAI-PMH se mostra um protocolo muito mais flexível que o REST, já que é bem documentado e não é dependente exclusivamente de uma ferramenta, que é o caso do Dspace. Outro fator a ser considerado é que diferentemente do REST que a obtém todos os dados a cada requisição realizada, o OAI-PMH permite uma colheita seletiva, isto é, através de uma variável que contém a data e hora da última colheita, na próxima requisição é possível obter somente os registros que mudaram desde a última colheita, evitando o gasto de tráfego de rede desnecessário.

Já em relação ao acesso direto no banco de dados, não é uma boa prática fazer o Noosfero consumir acessar o banco de dados do Dspace diretamente, pois pode acontecer que no mesmo tempo que o Noosfero consume os objetos digitais do banco do Dspace, o mesmo pode estar sendo utilizado para cadastrar mais objetos digitais, causando perda de dados, lentidão no acesso a base, entre outros. Além de ter os mesmos problemas do REST, como ser uma solução específica para o Dspace e também não conter nenhum controle sobre o que foi consumido.

É importante ressaltar que o OAI-PMH também foi escolhido em relação ao Z39.50⁷, pois permite escalabilidade e uma maior liberdade de troca de informações e uma interoperabilidade com baixo grau de envolvimento entre as partes, preservando, dessa forma, a autonomia de cada um dos canais de participação, permitindo também um baixo fluxo de dados entre todas as partes, já que a colheita é feita somente dos dados necessários.

Com base nas colocações acima, foi desenvolvido um plug-in de comunicação OAI-PMH para que o DSpace converse com o Noosfero, cuja a arquitetura de comunicação está resumido na Figura 15.

⁶ O REST (*Representational State Transfer* ou Transferência de Estado Representativo é um estilo de arquitetura para sistemas de distribuição hipermedia distribuídos (FIELDING, 2000).

⁷ Maiores informações em: <<http://pt.wikipedia.org/wiki/Z39.50>>

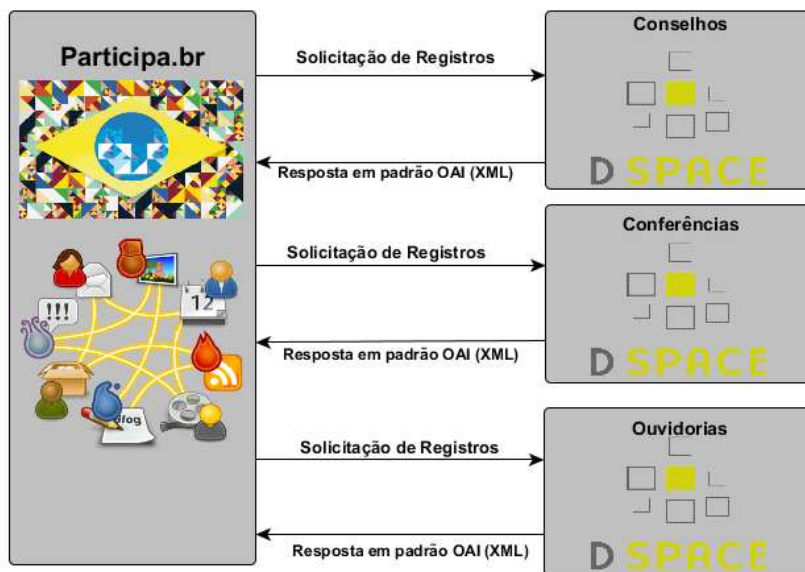


Figura 15 – Modelo de comunicação OAI-PMH entre o portal e os canais de participação.
Adaptado de Oliveira (2009)

Com base na imagem ilustrada pela Figura 15, o portal Participa.br será um provedor de serviços, através de um módulo da biblioteca digital desenvolvido através de um *plugin* no Noosfero. Com base nesse *plugin*, ele será capaz de interagir com os provedores de dados e coletar metadados essenciais sobre cada um desses provedores de dados, por meio do protocolo de *harvesting* OAI-PMH, visto na seção 3.1. Dessa forma, o Participa poderá auxiliar os canais de participação na organização de seus conteúdo e, ao mesmo tempo, poderá ser um provedor de informações atualizadas sobre os diversos canais de participação, uma vez que as colheitas de dados nos provedores podem ser feitas em qualquer instante desejado.

Na Figura 16, tendo em vista da utilização do portal Participa.br é feito dentro dentro dos conceitos da web 2.0 que é caracterizada por meio da interatividade com o usuário por meio de mecanismos descentralizados e interoperáveis, como por exemplo, blogs, hubs de participação, comentários por parágrafos, dentre outros. Utilizando conceitos presentes da Web Semântica, esses mecanismos de interatividade são aprimorados com a inserção de significados aos conteúdos de informação, permitindo maior interoperabilidade dentro dos canais de comunicação e entre eles, formando um espaço de informação social e semântico (CRUZ; MEIRELLES, 2014). Por isso, uma das iniciativas é fornecer os dados sobre os mecanismos formais de participação de maneira semântica, de maneira que facilite a recuperação dos dados, uma vez que estes terão um significado e um valor bem definidos.

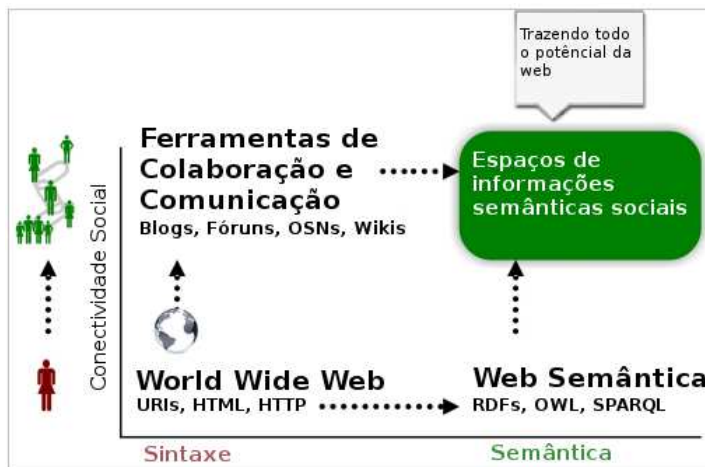


Figura 16 – Espaço de colaboração semântico e social no Participa.br. Adaptado de (KRUK, 2006)

É importante ressaltar que a implantação dessa arquitetura não será obrigatória pelos mecanismos formais de participação. Embora existam alguns conselhos que não possuem infra-estrutura necessária para instalação de uma biblioteca própria, assim como outros que não possuem pessoal necessária para manter uma, para esses casos existe a biblioteca digital de participação social, além do próprio Participa.br, que através de ferramentas de CMS (visto na seção 4.2.1) pode-se manter documentos de forma bastante simples.

5.3 Adaptando as ferramentas para a arquitetura proposta

Ainda neste capítulo, foi visto que o Dspace foi escolhido como o software para o desenvolvimento da biblioteca de participação social. De acordo com a seção 2.4, o Dspace fornece suporte para o protocolo OAI-PMH 2.0 (mais atual) como padrão. Caso ainda não esteja disponível, a instalação consiste apenas em colocar o arquivo *dspace-oai.war* dentro da pasta */webapps* localizada na instalação do Dspace. Após isso é necessário gerar uma nova versão do Dspace (*build*), que ele automaticamente compilará o servidor OAI⁸. Para acessar o Dspace, basta acessar o link */dspace-oai* a partir da url base da biblioteca.

Como visto na seção 4.2, o Noosfero é uma framework para criação de redes social e CMS, logo, ele não possui nenhuma afinidade ou funcionalidades que assemelhariam ele com os softwares de bibliotecas digitais. No entanto, para a criação de um ambiente onde a participação social seja os centros das atenções, é necessário que os dados, documentos, atas, relatórios, entre outros, dos mecanismos formais de participação estejam presentes

⁸ O processo de instalação completo está disponível nos manuais e documentação do Dspace, disponível no endereço: <<https://wiki.duraspace.org/display/DSPACE/Use+the+OAI-PMH+interface>>

dentro do Participa.br, aproveitando a arquitetura do Participa.br, além de agregar mais valor aos serviço.

Para viabilizar a construção desse módulo OAI-PMH no Noosfero, existem alternativas como a gema (*gem*) OAI⁹, que implementam várias funcionalidades entre provedores de dados e serviços pertencentes ao padrão do OAI (como por exemplo os tratamentos dos verbos).

Já para triplificação dos dados, foi feito um estudo(apresentado no capítulo 3) sobre as principais ferramentas candidatas a compor a *triple store*. Com base nesse estudo, foi escolhido o Virtuoso, pelo fato dele oferecer maneiras facilitadas que permitem conecta-lo mais facilmente à linguagens, como por exemplo *Ruby*. A literatura também mostra que o Virtuoso é uma das melhores triple stores existentes. Existe um módulo (*gem*) que é responsável por realizar essa integração com o Virtuoso, assim como no Ruby OAI, essa biblioteca trás um *endpoint SPARQL* para dentro da aplicação *Ruby*.

Um fator a ser considerado é o uso de gemas (*gems*) na implementação em vez da criação de soluções manuais. Do ponto de vista de desenvolvimento de software a utilização dessas gemas diminui o reuso e a criação de código duplicado, além de estar contribuindo em softwares de terceiros, fato esse que é uma das justificativas para a utilização de software livre.

5.4 O plugin OAI-PMH para o Noosfero

Nessa seção será apresentado o plug-in para adaptação do Noosfero para obtenção dos metadados dos artefatos presentes na biblioteca digital de participação, utilizando o protocolo OAI-PMH. Além disso também é apresentado o desenvolvimento da triplificação e apresentação desses metadados referentes aos mecanismos formais de participação. É importante ressaltar que, nas próximas seções que vão tratar do desenvolvimento, serão mostrado somente as partes essenciais do código, que realizam funções bem específicas, e que todo o código deste plug-in está disponível em um repositório *Git*, através do Gitlab¹⁰.

5.4.1 Adaptando a arquitetura do Noosfero

Para aplicar a arquitetura proposta, foram feitos alguns levantamentos e propostas. No entanto, levando em consideração a experiência dos desenvolvedores do SERPRO, além de consultores da Secretária Nacional da Juventude foi feito uma proposta de integração seguindo a ideia discutida no fórum de discussão de bugs e demandas de desenvolvimento

⁹ Maiores Informações em: <<https://github.com/code4lib/ruby-oai>>

¹⁰ Disponível em: <<https://gitlab.com/luiz/oai-pmh-plugin/>>. Até o momento da escrita desse trabalho foram realizados 5 commits no repositório, sendo 4 na última semana, já que foi preferível fazer um único commit com as funcionalidades estáveis, enquanto os demais são correções de *bugs* e refatoração de código.

do Participa.br¹¹. A Figura 17 apresenta as mudanças feitas na arquitetura do Noosfero, para suportar a arquitetura proposta.

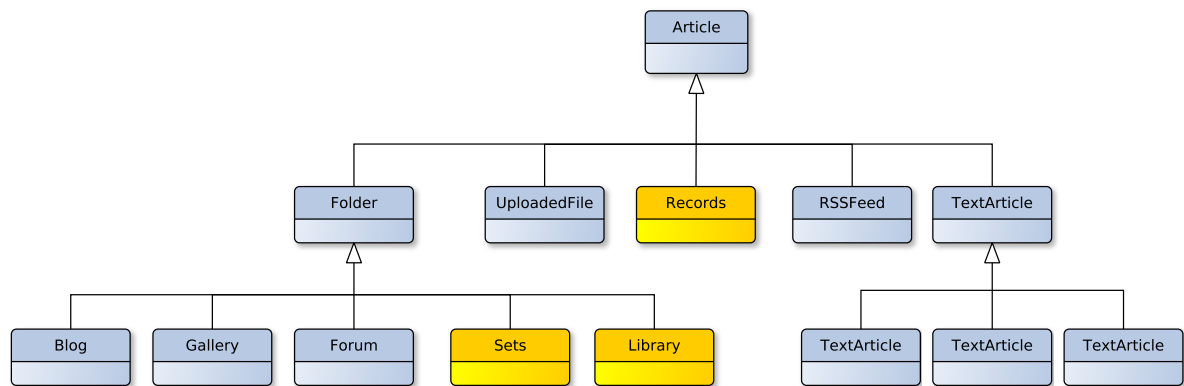


Figura 17 – Novos conteúdos adicionado no Noosfero. Adaptado de Bucher (2013)

Ainda de acordo com a Figura, foi criado um novo tipo de artigo (*Article*), chamado de `OaiPlugin::Library`, que contém as informações de uma instância de servidor OAI-PMH. Esse novo tipo de conteúdo herda as características de pasta (*Folder*). Essa herança permite que o conteúdo `OAI::Library` obtenha os mesmos comportamentos de `Folder`, necessitando rescrever somente funções necessárias.

Também foi criado um conteúdo chamado `OaiPlugin::Set`, que são grupos de assuntos relacionados. Assim como o `OaiPlugin::Library`, o `OaiPlugin::Set` herda as características de uma pasta (*Folder*) no Noosfero, assim permite que sejam guardados vários objetos digitais com um assunto relacionado dentro de um mesma pasta. No entanto, é importante ressaltar que esse conteúdo depende muito da organização da biblioteca, caso a biblioteca esteja desorganizada, conseqüentemente na hora que for feito o *Haversting*, a desorganização vai ser refletida dentro do Noosfero.

A cada metadado coletado é criado um novo tipo de conteúdo, chamado de `Oai::Record`. Esse conteúdo herda as características de um artigo de texto no Noosfero, no entanto ele foi adaptado para que fosse suportada a adição de metadados.

5.4.2 Desenvolvimento do haverster

Nesta subseção serão apresentados os passos relativos para que o usuário possa fazer o *haversting* de todos os metadados de uma biblioteca para dentro da rede social Noosfero. Também é apresentado as requisições e respostas vindas do repositório.

O usuário participante da rede social cria uma comunidade ou é participante de uma, desde que seja administrador dessa comunidade. O usuário entra no painel de con-

¹¹ Maiores informações em: <<https://gitlab.com/participa/noosfero/issues/210>>

figuração da comunidade e adiciona um novo tipo de conteúdo chamado de "Biblioteca". Dentro desse conteúdo, o usuário indica um título e uma descrição, além da instância do servidor OAI-PMH que será feita a colheita (*Haversting*) dos metadados. A imagem ilustrada pela Figura 18 indica essa sequência.

Figura 18 – Tela para adição de nova biblioteca

Após salvar, o usuário é direcionado para biblioteca criada, onde existe um botão para o usuário "Adicionar Sets", que são grupos de artefatos que possuem assuntos correlatos. A imagem ilustrada pela Figura 19 representa essa tela.



Figura 19 – Tela para adição de um novo set

Após selecionado, o Noosfero através da gema *Ruby OAI*, faz uma requisição do verbo *list sets* no repositório. O código 5.1 abaixo possui a principal função que chama esse verbo.

Listing 5.1 – Código da requisição *list_sets*

```
def get_all_sets_from
```



```
@client.list_sets
end
```

O servidor OAI-PMH devolve todos os *sets* para o Noosfero em formato XML, que está representado no código 5.2.

Listing 5.2 – Resultado da requisição *list_sets*

```
<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>2014-11-21T01:47:42Z</responseDate>
    <request verb="ListSets">http://cadair.aber.ac.uk/dspace
      -oai/request</request>
    <ListSets>
      <set>
        <setSpec>com_2160_1</setSpec>
        <setName>Lifelong Learning</setName>
      </set>
      <set>
        <setSpec>com_2160_2</setSpec>
        <setName>Information Studies</setName>
      </set>
      <set>
```

Através de um tratamento no Noosfero é possível retornar para o usuário todos os *Sets* disponíveis de maneira mais agradável, permitindo eles selecionarem o(s) set(s) que deseja(m) fazer o *Haversting* de maneira mais intuitiva. Devido o fato do conteúdo biblioteca utilizar uma herança do conteúdo pasta, permite que o comportamento seja herdado, sendo possível que o usuário escolha vários tipos de *Sets*. A Figura 20 ilustra o resultado desse tratamento.

Quando o usuário seleciona o(s) set(s) que ele deseja fazer o *Haversting*, o Noosfero faz a conexão no repositório escolhido e utiliza o verbo "*list records*", passando o *set* escolhido como um dos seus parâmetros, conforme será visto no código 5.3.

Listing 5.3 – Código da requisição *list_records*

```
def list_records(spec)
  @client.list_records(:set => spec)
end
```

The highlighted fields are mandatory.

Title (*)

Parent folder:

License

Description:

OAI SETS LIST

Check/Uncheck All Unlocked	<input type="checkbox"/>
Addysg a Dysgu Gydol Oes / Education & Lifelong Learning	<input type="checkbox"/>
Astudiaethau Gwybodaeth / Information Studies	<input type="checkbox"/>
Astudiaethau Theatr, Ffilm a Theledu / Theatre, Film & Television Studies	<input type="checkbox"/>
Celf / Art	<input type="checkbox"/>
Cyfrifiadureg / Computer Science	<input type="checkbox"/>
Cymraeg / Welsh	<input type="checkbox"/>
Daearyddiaeth ac Astudiaethau Daearyddol / Geography & Earth Sciences	<input type="checkbox"/>
Gwleidyddiaeth Ryngwladol / International Politics	<input type="checkbox"/>

Figura 20 – Tratamento dos *sets* para apresentar aos usuários

Após isso o repositório devolve todos os *Records* em formato XML. Para visualização dos dados no Noosfero, foi feito todo o tratamento desse XML utilizando um módulo do *Nokogiri*¹² chamado de *XPath*, para fazer toda a extração dos metadados do XML e gravando no conteúdo "*Record*". Essa extração de metadados é visualizada no código 5.4

Listing 5.4 – Extração de metadados dos records obtidos

```
def self.split_record(record, record_metadata)
  metadata_xml = Nokogiri::XML(record_metadata.metadata.to_s)
  metadata_xml.remove_namespaces!

  record.creator = metadata_xml.xpath('//creator').text
```

Após a extração, todos os dados são guardados dentro de um novo objeto *Oaipmh-Plugin::Record*, que posteriormente é feito a persistência dentro do banco de dados do Noosfero.

É importante ressaltar que foi feito um laço de repetição para extrair os 15 itens que compõe o Dublincore. Também existem itens que possui mais de um metadado do mesmo tipo (por exemplo: dois ou mais *identifiers*). Para esses casos, como será visto no

¹² Maiores informações em: <http://www.nokogiri.org/>

código 5.5, foi necessário utilizar um laço para extração de cada metadado e adicionar um caractere de limite, que poderá ser removido posteriormente.

Listing 5.5 – Extração de vários metadados do mesmo tipo

```
metadata_xml.xpath('//description').each { |d|
  description.concat(d.text)
  description << "|"
}
```

No entanto, foi visto que os atributos (campo no banco dados) de um artigo no Noosfero não foram suficientes para armazenar todos os metadados oriundos dos objetos digitais. Logo, foi feita uma migração¹³ adicionando os atributos referentes ao Dublin Core dentro do Noosfero.

A cada objeto digital que é feito a colheita (*Haversting*), o seus metadados são gravados dentro de um novo objeto `Oai::Record`.

Logo após a conclusão do *haversting* dos metadados, os mesmos são apresentados em uma lista na ordem decrescente de seu nome (*title*) conforme imagem ilustrada pela Figura 21. Se o usuário deseja obter maiores informações, basta o mesmo selecionar o título do objeto digital que o mesmo será direcionado para uama tela contendo maiores detalhes, que está ilustrado pela Figura 22.

5.4.3 Desenvolvimento da triplificação

Nesta subseção será apresentada como é realizado a triplificação de dados, atendendo uma necessidade que foi levantada nesse trabalho.

Como visto na seção 5.2, um dos fatores para a utilização do Virtuoso como ferramenta *triple store* a ser utilizada, foi sua grande compatibilidade com as linguagens de programação, em especial a linguagem Ruby, que está presente no Noosfero. Sua instalação é bastante simples¹⁴.

Após instalado é necessário informar dentro do Noosfero a *url (link)* de onde o Virtuoso foi instalado¹⁵. Também é necessário informar um usuário com permissões de escrita no Virtuoso, pois o *haverster* faz a inclusão dos metadados diretamente em SPARQL nessa ferramenta. Foi desenvolvido um painel dentro do menu de administração dos *plug-*

¹³ Através de uma *migration* é possível evoluir o esquema do banco de dados, adicionando novos campos e tabelas, alterando tabelas já existentes ou até mesmo excluir tabelas e campos, sem a necessidade de entrar diretamente no banco de dados. Maiores informações em <http://api.rubyonrails.org/classes/ActiveRecord/Migration.html>

¹⁴ Nas principais distuições Linux, o Virtuoso já é disponível em pacote, bastando apenas instalá-lo na máquina. Após instalado o Virtuoso é automaticamente configurado pelos gerenciadores de pacotes

¹⁵ Em instalações padrões está localizado em <http://localhost:8890>

[Go back to Conselho Nac...](#)

[Edit folder](#) [Delete](#) [New article](#) [Upload files](#) [Report abuse](#)

HANES A HANES CYMRU / HISTORY & WELSH HISTORY

A Concise History of France




Authors: Price, Roger

Description: This book provides a clear and well-informed guide to French history from the early Middle Ages to the present - from Charlemagne to Chirac. It offers the most up-to-date and comprehensive study of French history available. Among the book's central themes are the relationship between state and so...

Subjects:

Date Issued: 2005

"A Glorious Morn"?: Methodism and the Rise of Evangelicalism in Wales, 1735-1762'



Authors: Jones, David Ceri

Description:

Subjects:

Date Issued: 2008

Figura 21 – Lista de objetos digitais

[Go back to Hanes a Hane...](#)

[Edit folder](#) [Delete](#) [New article](#) [Upload files](#) [Report abuse](#)

A CONCISE HISTORY OF FRANCE

Viewed 2 times

Authors: Price, Roger

URI: [dspace: 2160/1309](https://dspace.org/2160/1309)

Date Issued: 2005

Abstract

This book provides a clear and well-informed guide to French history from the early Middle Ages to the present - from Charlemagne to Chirac. It offers the most up-to-date and comprehensive study of French history available. Among the book's central themes are the relationship between state and society, the impact of war and such crucial questions as who possessed political power and how this power was used. It takes account of the great figures in French history, among which Philip Augustus, Henri IV, Louis XIV, Robespierre, Napoleon, de Gaulle and also the findings of social historians concerned with ordinary people, social structures, beliefs, and economic activity. This second edition, substantially re-written to take account of recent research, includes a new chapter on contemporary France - a society and political system in crisis as a result of globalisation, international terrorism, racial tension and a loss of confidence in political leaders.

Description

Price, Roger, A Concise History of France (Cambridge: Cambridge University Press, 2005), pp.xiii+491 RAE2008

Figura 22 – Detalhes de um objeto digital

ins, onde é possível o usuário realizar essas configurações. Esse painel está ilustrado na Figura 23.

Figura 23 – Painel de configuração do Virtuoso no Noosfero

Após a configuração, o Noosfero realiza a conexão no Virtuoso através do código . Esse método que está exemplificado no código 5.6 é chamado somente uma vez, no início do método de haversting dos metadados. É importante notar que o método de autenticação é *digest* para futuras conexões com outras instâncias do Virtuoso, pois na maneira que foi feita nesse trabalho, o Virtuoso está autênticando apenas pela URI.

Listing 5.6 – Conexão do Noosfero com Virtuoso

```
def initialize(environment, oai_server)

  @environment = environment
  @oai_server = URI(oai_server.to_s)
  @settings = Noosfero::Plugin::Settings.new(environment,
    OaipmhPlugin)
  @virtuoso = RDF::Virtuoso::Repository.new(
    environment.virtuoso_url,
    update_uri:environment.virtuoso_sparql_auth,
    username: environment.virtuoso_username,
    password: environment.virtuoso_password,
    auth_method: 'digest')

  @settings.namespace_management.each do |namespace|
    "@namespace = RDF::StrictVocabulary.new('#{namespace[:uri]}')"
```

end

Para catalogação dos dados, foram levantados as principais ontologias (vocabulários), que permitem um maior detalhamento dos elementos do Dspace. Algumas dessas ontologias foram levantadas através do trabalho de Konstantinou et al. (2014), que ressalta a importância de criar um catálogo com as principais ontologias utilizadas para mapear os objetos do Dspace. Algumas ontologias foram levantadas com a ajuda do site LOV (*Linked Open Vocabulary*)¹⁶, que possui vários conjuntos de vocabulários abertos criando uma grande nuvem de *linked data*, conforme visto na imagem ???. A Tabela 1 é apresentada no apêndice A.

Pensando em uma maneira de facilitar a vida do usuário, foi desenvolvido um painel para que o mesmo possa inserir novos vocabulários sem a necessidade de modificar o código fonte da aplicação, conforme está ilustrado na Figura 24.

The screenshot shows a web interface titled "NAMESPACE MANAGEMENT". It contains a table with two columns: "Prefix" and "URI". Each row in the table has a text input field for the prefix, a text input field for the URI, and a "Remove" button. At the bottom of the interface, there are three buttons: "New", "Save", and "Cancel".

Prefix	URI	
DC	http://purl.org/dc/terms/	Remove
cc	http://creativecommons.org/ns#	Remove
cito	http://purl.org/spar/cito/	Remove
bibo	http://purl.org/ontology/cito/	Remove
fabio	http://purl.org/spar/fabio/	Remove
frbr	http://purl.org/vocab/frbr/core#	Remove
frbre	http://purl.org/vocab/frbr/extend	Remove
frbrer	http://flastandarts.info/ns/fr/frbr/	Remove
isdb	http://flastandarts.info/ns/isdb/el	Remove
lvont	http://lexvo.org/ontology#	Remove
mrel	http://id.loc.gov/vocabulary/relat	Remove
openmv	http://purl.org/net/opmvns#	Remove
prism	http://prismstandart.org/namespa	Remove
prv	http://purl.org/net/provenance/ns	Remove
rdarel	http://rdvocab.info/RDARelations	Remove
schema	http://schema.org/	Remove

Figura 24 – Painel de inserção de *Namespaces*

Ainda de acordo com o trabalho realizado por Konstantinou et al. (2014) e do site LOV, foram obtidas todos os termos de outras ontologias que possuem propriedades semelhantes aos termos do Dublincore, necessários para catalogar os dados dos mecanismos formais de participação. É importante ressaltar que a utilização de todas essas propriedades de cada ontologias pode ser dispendioso, mas em virtude de testar o *plug-in*

¹⁶ Maiores informações em: <<http://lov.okfn.org>>

relevou-se em utilizá-las. Essas propriedades são referidas na Tabela 2, que está localizada no apêndice A.

Assim como foi feito acima, também foi desenvolvido um painel para que o usuário possa utilizar outras propriedades em um metadado ou até mesmo mais de uma propriedade dentro de um mesmo metadado, possibilitando que o usuário utilize as propriedades que ele mesmo desejar. Porém é alertado que para funcionar é necessário que o vocabulário da propriedade desejada esteja devidamente cadastrada. Na Figura é apresentada ?? essa tela.

Metadata	Attributes
dc:creator	DC.creator, mrel:cre, mrel:aut, frbr:P2009, frbr:cre
dc:subject	DC.subject, frber:P2023, frbrer:P2025, frbrer:P20
dc:description	DC.description, bibo.shortDescription, isbd:P1037-f
dc:title	DC.title, frber:P3001, frber:P3008, isbd:P1004, is
dc:publisher	DC.publisher, mrel:pbl, frber:P3056, isdb:P1017, si
dc:contributor	DC.contributor, mrel.*, schema.contributor
dc:format	DC.format, frber.P3023
dc:source	DC.source
dc:relation	DC.relation, frber.P2043-P21109, rdarel.relatedW
dc:coverage	DC.coverage, shema.contentLocation
dc:rights	DC.rights, cc:license, prism.copyright, dcterms:access
dc:identifier	DC.identifier, bibo.asin, bibo.coden, bibo.doi, bibo.ean
dc:date	DC.date, frbrer.P3003, frbrer.P3010
dc:language	DC.language, frbrer.P3011, lvont.language, schema.
dc:type	DC.type, prism.genre, schema.genre

Save Cancel

Figura 25 – Painel de mapeamento dos metadados

Como foi dito na seção 3.2.1, a inserção de novos vocabulários pode permitir uma maior ligação entre os metadados, permitindo que a máquina consiga fazer mais inferências, consequentemente trazendo mais informações relevantes de um determinado metadado. Após a configuração da tela ilustrada pela Figura 24 é inserido novos vocabulários dentro do Noosfero, utilizando uma extensão da biblioteca RDF.rb, que permite a inserção de novos vocabulários, o código 5.7 exemplifica essa sequência.

Listing 5.7 – Adição de novos *namespaces*

```
"@namespace = RDF::StrictVocabulary.new("#{namespace[:uri]}")
"
```

Nessa função, cada prefixo configurado no painel da Figura 24 é gravado dentro de um novo Vocabulário. Para cada *namespace* configurado dentro no Noosfero é repetido o

mesmo código, como é visto na seção abaixo 5.8 .

Listing 5.8 – Adição de cada *namespaces*

```
@bibo = RDF::StrictVocabulary.new("http://purl.org/ontology/
  bibo/")
```

A cada metadado que é feito o *haverst* e posteriormente salvo no banco de dados, é chamada uma função que realiza essa triplificação através do endpoint SPARQL do Virtuoso configurado anteriormente. Essa função realiza uma conexão no Virtuoso através dos parâmetros configurados na tela apresentada pela Figura 23, inserindo cada um dos metadados dentro de um grafo RDF no formato de sujeito-predicado-objeto. O código 5.9 apresenta a principal chamada para inserção dos dados dentro do Virtuoso, que é feita através de SPARQL.

Listing 5.9 – Função de inserção de triplas RDF em SPARQL no Virtuoso

```
def triplify(item)
  _identifier = extract_uri(item.identifier)
  URI.join(@oai_server, "#").to_s

  item.map(&:metadata['noosfero'])
  ach do |value|
    ontology_proprety = metadata[:attribute].split(", ")
    ontology_proprety.each do |proprety|
      query = RDF::Virtuoso::Query.insert_data([RDF::URI.
        new(proprety_identifier), RDF::URI.new("@"+#{
          proprety}"), value]).graph(RDF::URI.new(graph))
    end
  end
  tuoso.insert(query)
```

Para cada iteração é adicionado uma tripla dentro de um grafo RDF, que vem da URL da biblioteca digital, conforme código 5.10. No caso da biblioteca utilizada como exemplo, o identificador é um registro no Handle System¹⁷.

Listing 5.10 – Exemplo de inserção de *query* em SPARQL no Virtuoso

```
virtuoso.insert(RDF::Virtuoso::Query.insert_data([RDF::URI.
  new("http://www.participa.br/exemplo/_ata"), RDF::URI.new(
  "http://purl.org/dc/terms/title"), dc.title]).graph("http
  ://www.biblioteca.com/"))
```

¹⁷ O Handle System permite uma forma eficiente, extensível para serviços de identificadores exclusivos e persistentes de objetos digitais

Ainda nessa função, podemos observar que o sujeito, composto pelo metadado *identifier*, que é responsável por fornecer o identificador único para cada tripla RDF. O predicado é formado pelos *namespaces* e seus mapeamentos, configurado através da Figura 24. Já o valor ou objeto é formado pelo valor do metadado que foi feito o Haverst.

Por fim, na Figura 26 é ilustrado que através de uma query em SPARQL (mostrada no código 5.11) dentro do endpoint SPARQL do Virtuoso é possível obter todo resultado da triplicação e *haversting*, com cada metadado contendo o seu namespace e respectivo valor.

sujeito	predicado	objeto
http://hdl.handle.net/2160/1257	http://purl.org/dc/terms/title	Welsh Reflections: Y Drych and America 1851-2001
http://hdl.handle.net/2160/1257	http://purl.org/dc/terms/creator	Jones, William
http://hdl.handle.net/2160/1257	http://purl.org/dc/terms/creator	Jones, Ied Gruffydd
http://hdl.handle.net/2160/1257	http://purl.org/dc/terms/description	Jones, A. and Jones, W. (2001). Welsh Reflections: Y Drych and America 1851-2001. Llandysul: Gomer Press. RAE2008
http://hdl.handle.net/2160/1257	http://purl.org/dc/terms/date	2008-11-25T17:13:50Z
http://hdl.handle.net/2160/1257	http://purl.org/dc/terms/date	2001-08
http://hdl.handle.net/2160/1257	http://purl.org/dc/terms/identifier	Jones, W & Jones, L G 2001, Welsh Reflections: Y Drych and America 1851-2001, Gomer Press.
http://hdl.handle.net/2160/1257	http://purl.org/dc/terms/identifier	1 84323 021 6
http://hdl.handle.net/2160/1257	http://purl.org/dc/terms/identifier	PURE: 86898
http://hdl.handle.net/2160/1257	http://purl.org/dc/terms/identifier	dspace: 2160/1257
http://hdl.handle.net	http://purl.org	http://hdl.handle.net/2160/1257

Figura 26 – Resultado da triplicação dos objetos digitais da biblioteca

Listing 5.11 – Query em SPARQL de obtenção dos resultados

```
SELECT * WHERE {
  GRAPH <http://cadair.aber.ac.uk/dspace-oai/request/#>
    { ?sujeito ?predicado ?objeto }
}
```

5.5 Considerações

Como visto, utilizar o protocolo OAI-PMH para haversting dos metadados permite maior flexibilidade em relação a disponibilidade informações. Para este trabalho foi utilizado como exemplo a biblioteca da universidade de Aberystwyth (cidade do País de Gales situada no Reino Unido). Sua seleção foi devido a organização nos conteúdos dos seus artefatos. Nessa biblioteca é possível obter registros por grupos, além dos metadados estarem bem documentados e catalogados, o que será um ambiente próximo da biblioteca de participação social que ainda não está disponibilizada. Outro fator relevante deu-se em

fato do número de artefatos dessa biblioteca, que possui aproximadamente 12000 itens, assim é possível realizar o haversting dos metadados.

É importante ressaltar que o desenvolvimento desse plug-in, se difere de uma solução que também está sendo desenvolvida pelo SERPRO ¹⁸. Uma das diferenças é que no plug-in presente neste trabalho de conclusão de curso, além dele realizar a colheita de todos os metadados de um grupo de assuntos, ele salva todos esses metadados dentro do Noosfero (utilizando dois bancos de dados simultaneamente). De fato, salvando todos esses metadados dentro do Noosfero, além de poder utilizar todas as ferramentas de *CMS* e redes sociais que o mesmo provêm, ainda é possível utilizar ferramentas que estão sendo desenvolvidas dentro do Participa.br, como por exemplo, o comentário por parágrafo. Do ponto de vista nas principais recomendações no uso de banco de dados, utilizar dois bancos de dados diferentes para gravação dos mesmos dados gera redundância, no entanto, para este trabalho foi utilizado essa abordagem por se tratar de um protótipo.

¹⁸ O repositório dos códigos está disponibilizado em:
<https://gitlab.com/participa/noosfero/tree/virtuoso_integration>

6 Conclusão

Ao longo deste trabalho de conclusão de curso, realizamos atividades que passam pelas principais áreas de conhecimento da Engenharia de Software, desde concepção até o desenvolvimento do plug-in, como visto neste trabalho. No entanto, um dos grande aprendizados nesse trabalho é a possibilidade visualizar a integração entre a engenharia de software com conceitos de outras áreas, neste trabalho utilizamos conceitos da Ciência da Informação, através das bibliotecas digitais, uso metadados, entre outros, além de trazer conceitos que serão utilizados no futuro para prática, como foi o caso da Web Semântica, através do uso de Linked Data e triplificação de dados.

Se tratando em protocolos de interoperabilidade, apesar de sua idade (12 anos) o protocolo OAI-PMH ainda é uma opção viável na integração de repositórios digitais, já que:

- API bem documentada, simples e consistente.
- Facilidade em fazer harvesting, através de um *crawler* (no caso deste trabalho, foi um *plug-in*).
- Amplitude em sua utilização mesmo nos dias atuais, permitindo um maior tempo de suporte pela comunidade em fóruns e listas de e-mail.
- Implementar o OAI-PMH não é uma tarefa difícil, como visto, existem *gems* (*Ruby*) e outras bibliotecas nas mais diversas linguagens que auxiliam na construção da implementação. No entanto manter um repositório é mais custoso, pois necessita de uma boa organização dos dados.

É importante ressaltar, que da maneira que o protocolo OAI-PMH foi implementado, não há uma maneira de fazer uma via de fluxo de informações de "full-duplex" ou "mão dupla" (possibilidade de atualizar os dados de um objeto digital dentro do provedor de serviços) utilizando esse protocolo. De certa forma, faz sentido em manter o protocolo funcionando desta maneira, já que em termos de segurança e autenticidade da informação, podemos deixar a interface do provedor de dados para realizar tais atualizações. No entanto, a única forma de realizar essa funcionalidade seria fazer o Noosfero conectar diretamente no banco de dados do Dspace atualizando as informações diretamente lá, o que não fária muito sentido, já que é a própria função do Dspace.

Sobre o desenvolvimento do plug-in utilizando a Webservice, que foi feito pelo SERPRO ele possui a vantagem de possuir suporte da linguagem Ruby, diferentemente

do OAI-PMH, que é necessário uma gema (*gem*). Por um lado, esse plug-in de Webservice desenvolvido para o Noosfero é preso totalmente ao Dspace; por outro lado, o plug-in OAI-PMH, que, como visto, é um protocolo documentado, pode ser utilizado nos mais diversos softwares de bibliotecas digitais, além de qualquer outro software que use o protocolo.

Porém, o protocolo OAI-PMH possui uma grande desvantagem em relação ao Webservice, que é o fato dele não ser um protocolo de recursos e sim somente de metadados. Isso dificulta bastante a disseminação dos documentos (pdfs, docs, imagens, etc), fato esse que para ser ajustado necessita de realizar trabalho de programação para ajustar o link do artefato diretamente na biblioteca digital a ser escolhida.

De fato, as duas opções possuem suas vantagens e desvantagens, mas uma solução ideal seria utilizar a grande flexibilidade que o protocolo OAI-PMH fornece, com a robustez de um Webservice utilizando REST, permitindo o haverst completo de documentos e seus metadados para dentro do Noosfero.

Ressaltamos que a triplificação em RDF é um primeiro passo para disponibilização de conteúdos em uma internet mais inteligente. Existem mais técnicas, como por exemplo, a extração de palavras chaves de um documento, análise do texto baseando-se em sentimentos, utilização de linguagem natural para pesquisas, além da confecção de motores de buscas robustos que permitem buscar esses documentos de maneira muito mais facilitada e que traz maior valor para os usuários.

Na introdução foram levantadas algumas questões que auxiliaram no seguimento do trabalho.

- **Q1** - Como as bibliotecas digitais garantem a atualização das informações presentes nos mecanismos formais de participação social?
- **Q2** - Bibliotecas digitais descentralizadas potencializam os mecanismos formais de participação social em rede?

Na questão 1 e 2, como foi estudado no capítulo 2, uma biblioteca digital bem desenvolvida no quesito de uma boa arquitetura de informação trás maior facilidade para o usuário. Consequentemente, quanto maior o movimento pela comunidade de usuários, maior será a motivação dos mecanismos formais colocarem suas informações atualizadas. E também através do uso de bibliotecas digitais pode ser criado um padrão, para que os mecanismos formais sejam incentivados a disponibilizarem seus conteúdos, de forma que as pessoas efetivamente conheçam as funções de cada mecanismo social de participação.

- **Q3** - Como o Noosfero pode ser integrado com uma biblioteca digital?

O Noosfero pode ser integrado com uma biblioteca digital através de várias alternativas (OAI-PMH, Z39.50, Z39.99, webservice, crawling, acesso direto no banco

de dados, entre várias outras). Dentro dessas alternativas, exploramos quatro delas que julgamos mais importantes, mas o protocolo OAI-PMH permite que o Noosfero seja integrada com qualquer outro software de biblioteca digital existente, desde que este implemente o protocolo OAI-PMH. O uso desse protocolo ainda trás mais vantagens, pois permite que futuramente seja desenvolvido um repositório no próprio Noosfero, permitindo que novos serviços utilizem o conteúdo do Noosfero em outros serviços.

- **Q4** - Como integrar os dados presentes nas bibliotecas digitais com as namespaces e padrões de catalogação já propostas?

Como foi dito na seção 3.2.1, uma das funções do RDF é disponibilizar a informação de maneira interligada. Através de Vocabulários ou *namespaces* é possível utilizar coleções de classes pertencentes a outras ontologias propostas, como é o caso da FOAF, SCHEMA, DCTERMS, entre outras. No quesito de desenvolvimento, através do RDF.rb é possível adicionar novas *Namespaces* através da função `RDF::URI`, `RDF::Vocabulary` ou `RDF::StrictVocabulary`. Para suportar isso, foi feita uma interface dentro do próprio Noosfero, que permite os usuários escolherem os vocabulários assim como os atributos que pode utilizar para triplificação.

No desenvolvimento feito nesse trabalho, foi visto que o Participa.br tornou-se um provedor de serviços, sendo capaz de realizar o haversting dos metadados das bibliotecas digitais, desde que utilizem o protocolo OAI-PMH. No entanto, como foi visto é necessário alguns ajustes para realizar e apresentar o haversting de maneira correta, além de contar com a organização dos artefatos dentro da biblioteca. Em versões futuras do *plug-in*, pode ser feito uma automatização para que o usuário possa adaptar os metadados a serem mostrados dentro do Noosfero.

Ao final deste trabalho, julgamos que o plug-in é um protótipo funcional, no entanto, não foi possível de verificar o funcionamento do mesmo no ambiente de produção do Participa.br, já que até o presente momento da escrita deste trabalho de conclusão, a biblioteca digital de participação social ainda não estava disponibilizada. Porém, quando a biblioteca estiver disponível basta somente habilitar o protocolo OAI-PMH dentro da biblioteca e ativar o plug-in no Noosfero.

6.1 Contribuições

No final deste trabalho de conclusão concretizou-se as seguintes contribuições, levantadas na introdução deste trabalho:

- **C1** - Adaptação do Noosfero para implementar o protocolo OAI-PMH.

- **C2** - Adaptar o Noosfero para armazenar triplas RDF referentes aos artefatos (atas, normas, regimentos, entre outros) dos mecanismos formais de participação.
- **C3** - Permitir uma visualização mais detalhada dos artefatos referentes aos mecanismos formais de participação dentro do Noosfero.

6.2 Limitações

Trabalhar com Linked Data utilizando Ruby é custoso, já que a comunidade Ruby que trabalha com a Web Semântica não é muito grande, mas atualmente existem opções muito interessantes como a existência de uma gem chamada RDF.RB, que se assemelha RdfLib do Python ou até mesmo a RDF-Virtuoso.rb. No entanto, é difícil de perceber que a maior parte do trabalho em torno da Web Semântica é feito em Java (Apache Jena), e muitas tecnologias com grande potencial futuro, como Ruby on Rails ou Groovy on Grails, são aparentemente esquecidas, já que tutoriais nessas linguagens são aparentemente desatualizados e conseqüentemente errados.

Outro fator limitante é em relação ao próprio OAI-PMH. Existem repositórios que utilizam a versão estendida do formato de metadados Dublin Core, que foi visto na seção 2.3. Os metadados estendidos acrescentam muito na quantidade total de metadados e quando é feito o *Haversting* dessas informações, todos os metadados que estão relacionados são agrupados no mesmo grupo de metadados, por exemplo. Isso dificulta bastante, principalmente no metadado *Identifier*, que teoricamente é o identificador único de cada objeto digital, quando é feito o *Haversting* todos os metadados relacionados com ele são agrupados dentro de um só, por exemplo, em uma biblioteca existe os seguintes metadados: *dc.identifier.citation*, *dc.identifier.issn*, *dc.identifier.other*, *dc.identifier.uri*, logo após o *haversting* todos foram agrupados dentro de *identifier*, dificultando bastante o prosseguimento deste trabalho, visto que para utilização desse plug-in em outros ambiente é necessário realizar um pequeno ajuste que permita buscar os metadados corretamente.

6.3 Trabalhos Futuros

Pensando em deixar uma perspectiva de trabalhos futuros que podem ser feitos com base nesse assunto, foi feito um levantamento dos principais pontos de melhoria. Entre eles, citamos:

- Busca de metadados refinada.

Implementar uma busca para permitir os usuários pesquisarem as triplas RDF de forma bastante facilitada, sem a necessidade da utilização de queries em SPARQL,

já que entendemos que o ambiente do Participa.br, assim como outros são acessados por pessoas leigas em computação.

- Utilização do DelayedJob do Noosfero para o *Haversting* de metadados.

A gema DelayedJob permite que tarefas mais longas sejam executadas em segundo plano, sem a necessidade de deixar o Noosfero lento. No ambiente que foi testado, não obtivemos nenhum problema com a execução do plug-in, no entanto pensando em ambientes de produção, a melhor alternativa seria deixar o *Haversting* em segundo plano, aumentando o desempenho do Noosfero.

- Obtenção de documentos de forma automática.

Atualmente utilizando somente o OAI-PMH, o mesmo não permite a colheita dos documentos, diferentemente da REST API do Dspace. A ideia seria utilizar uma arquitetura híbrida, que permita o *haversting* dos metadados utilizando o OAI-PMH e a importação de documentos utilizando o *bitstream*, da API REST do Dspace. Para outras bibliotecas, existem outras soluções como no Omeka, permite a importação dos documentos via um Webservice também. O Greenstone não possui uma solução que permita essa integração.

- Permitir a colheita de metadados utilizando a versão estendida do Dublin Core.

Como dito nas limitações, há um grande problema entre o *haversting* e a utilização da versão estendida dos metadados Dublin Core. Em uma abordagem através do desenvolvimento de um protocolo que utilize a versão estendida do Dublin Core, o desenvolvimento será universal entre as bibliotecas, visto que não seria necessário adaptar a coleta de metadados para um biblioteca em específico.

- Salvar e disponibilizar os metadados em somente um banco de dados.

Na implementação atual é utilizado dois bancos de dados: (i) Virtuoso para os dados da bibliotecas já triplicados; e (ii) PostgreSQL para os dados vindos através do *haversting* da biblioteca digital que é apresentado no Noosfero. Essa solução gera muita redundância entre os dados, já que os mesmos dados estão gravados em dois bancos de dados diferentes. Para versões futuras, pode ser desenvolvida essa solução utilizando somente o banco de dados do Virtuoso para a apresentação dos metadados dentro do Noosfero, dispensando a necessidade de gravá-los dentro do PostgreSQL, diminuindo essa redundância de dados.

Apêndices

APÊNDICE A – Metadados utilizados para triplificação

Neste apêndice é apresentado as principais ontologias e propriedades utilizadas para triplificação dos dados em RDF. Na tabela 1 é apresentada as principais ontologias que serão utilizadas para triplificação dos dados referentes aos mecanismos formais de participação social.

Tabela 1 – *Namespaces* utilizados para triplificação dos dados

NOME	SIGLA	URI
IFLA's FRBRer Model	frbrer	http://iflastandards.info/ns/fr/frbr/frbrer/
International Standard Bibliographic Description (ISBD)	isbd	http://iflastandards.info/ns/isbd/elements/
Lexvo.org Ontology	lvont	http://lexvo.org/ontology#
MARC Code List for Relators	mrel	http://id.loc.gov/vocabulary/relators/
Open Provenance Model Vocabulary	opmv	http://purl.org/net/opmv/ns#
PRISM: Publishing & Requirements for Industry Standard Metadata	prism	http://prismstandard.org/namespaces/basic/2.0/
Provenance Vocabulary Core Ontology	prv	http://purl.org/net/provenance/ns#
RDA Relationships for Works, Expressions, Manifestations, Items	rdarel	http://rdvocab.info/RDARelationshipsWEMI
Schema.org	schema.org	schema http://schema.org/

Já na tabela 2, possui as propriedades levantadas que podem ser utilizadas para catalogação dos mecanismos formais ou qualquer outro objeto digital oriundo de uma biblioteca digital.

Tabela 2 – Elementos análogos utilizados para mapeamento dos termos em Dublin Core

ELEMENTO DSPACE	TERMO SEMELHANTE
dc.contributor	DC.contributor, mrel.*, schema.contributor
dc.coverage	DC.coverage, shema.contentLocation
dc.creator	DC.creator, mrel.cre, mrel.aut, frbr.P2009, frbr.creator, schema.author, schema.creator
dc.date	DC.date, frbrer.P3003, frbrer.P3010
dc.description	DC.description, bibo.shortDescription, isbd.P1037-P1046, isbd.P1064-P1068, isbd.P1073, isbd.P1078-P1079, isbd.P1086-1087, isbd.P1090-P1101, isbd.P1123-1124, isbd.P1136, rdarel.descriptionOfWork, schema.description, schema.comment
dc.format	DC.format, frber.P3023
dc.identifier	DC.identifier, bibo.asin, bibo.coden, bibo.doi, bibo.eanucc13, bibo.gtin14, bibo.handle, bibo.identifier, bibo.lccn, bibo.oclcnum, bibo.pmid, bibo.upc, fabio.hasDigitalArticleIdentifier, fabio.hasArXivId, fabio.hasCODEN, fabio.hasHandle, fabio.hasNationalLibraryOfMedicineJournalId, fabio.hasPPII, fabio.hasPubMedCentralId, fabio.hasPubMedId, fabio.hasURL, frbrer.P3028, frbrer.P3031, isbd.P1032, isbd.P1154, prism.doi, prism.url, schema.url
dc.language	DC.language, frbrer.P3011, lvont.language, schema.inLanguage
dc.publisher	DC.publisher, mrel.pbl, frber.P3056, isdb.P1017, schema.publisher
dc.relation	DC.relation, frber.P2043-P21109, rdarel.relatedWork
dc.rights	DC.rights, cc.license, prism.copyright
dc.source	DC.source
dc.subject	DC.subject, frber.P2023, frbrer.P2025, frbrer.P2020, frbrer.P2031, frbrer.P2033, frbrer.P2035, frbrer.P2037, frbrer.P2039, frbrer.P2041, frbr.subject, prism.keyword, schema.about, schema.keywords
dc.title	DC.title, frber.P3001, frber.P3008, isbd.P1004, isdb.P1012, isdb.P1026, schema.headline
dc.type	DC.type, prism.genre, schema.genre

Referências

- ALMEIDA, G. A. de. Participação e controle social na garantia dos direitos humanos. *Curso de Formação de Conselheiros em Direitos Humanos*, 2011. Acesso em 11 de junho de 2014. Disponível em: <<http://www.dhnet.org.br/dados/cursos/dh/cc/2/participacao.htm>>. Citado na página 19.
- ALVES, R. C. V. Web semântica: uma análise focada no uso de metadados. 2004. Citado na página 30.
- ARMS, W. *Digital Libraries*. [S.l.]: MIT Press, 2000. (Digital libraries and electronic publishing). ISBN 9780262261340. Citado 2 vezes nas páginas 25 e 27.
- BERRUETA, D.; PHIPPS, J. Best practice recipes for publishing rdf vocabularies. In: W3C WORKING GROUP NOTE. 2008. Acesso em 21 de setembro de 2014. Disponível em: <<http://www.w3.org/TR/swbp-vocab-pub/>>. Citado na página 43.
- BRASIL. Constituição da república federativa do brasil. *Diário Oficial da União*, Brasília, out 1988. Acesso em 04 de maio de 2014. Disponível em: <http://www.planalto.gov.br/ccivil_03/constituicao/constituicao.htm>. Citado na página 19.
- BUCHER, D. C. *Rede de colaboração social para universidades brasileiras: um estudo de caso de implantação e desenvolvimento distribuído de uma plataforma livre na Universidade de Brasília*. Tese (Doutorado) — Universidade de Brasília – Faculdade UnB Gama, 2013. Citado na página 61.
- CRUZ, F. W.; MEIRELLES, P. R. M. *Projeto BRA/12/018 – Levantamento de informações e proposta de catalogação e sistematização de conteúdos sobre Participação Social*. 2014. Citado 5 vezes nas páginas 29, 30, 31, 55 e 58.
- DIGITAL, G. *Presidenta Dilma apresenta novo Portal Brasil*. 2013. Acesso em 15 de junho de 2014. Disponível em: <<http://www.brasil.gov.br/governo/2013/09/dilma-anuncia-reformulacao-do-portal-brasil>>. Citado na página 47.
- DING, L. Tracking rdf graph provenance using rdf molecules. *UMBC Tech Report TR-CS-05-06*, 2005. Citado na página 42.
- FEITOSA, A. *Organização Da Informação Na Web : Das Tags À Web Semântica*. [S.l.]: Thesaurus, 2006. (Ciência da informação e da comunicação). ISBN 9788570625687. Citado na página 30.
- FERREIRA, S. M. S. P.; SOUTO, P. C. do N. *A Interface do Usuário e as Bibliotecas Digitais. Publicação de artigos feita pelo IBICT com o título “Bibliotecas Digitais: Saberes e Práticas”*. 2006. Acesso em 28 de abril de 2014. Citado na página 27.
- FIELDING, R. T. *Architectural Styles and the Design of Network-based Software Architectures*. Tese (Doutorado) — University of California, 2000. Citado na página 57.
- GARCIA, P. de A. B.; SUNYE, M. S. O protocolo oai-pmh para interoperabilidade em repositórios digitais. *I Congresso de Tecnologias para Gestão de Dados e Metadados do Cone Sul*, Paraná, 2003. Citado na página 39.

JACOBI, P. *Políticas sociais locais e os desafios da participação cidadina*. 2001. Acesso em 28 de abril de 2014. Citado na página 19.

KONSTANTINOU, N. et al. Exposing scholarly information as linked open data: Rdfizing dspace contents. In *The Electronic Library*, 2014. Citado na página 68.

KRUK, S. R. *Digital Libraries of the Future: Use of Semantic Web and Social Bookmarking to support E-learning in Digital Libraries*. 2006. Acesso em 20 de maio de 2014. Disponível em: <<http://www.slideshare.net/skruck/digital-libraries-of-the-future-use-of-semantic-web-and-social-bookmarking-to-support-elearning-in-digital-libraries>>. Citado na página 59.

KURAMOTO, H. *Bibliotecas Digitais: que nível de interoperabilidade? Seminário sobre Conteúdos Digitais na Internet, Rio de Janeiro*. 2014. Acesso em 19 de maio de 2014. Disponível em: <<https://cg-conteudos.cgi.br/seminario/arquivos/Seminario-CGI.br-Kuramoto-rj.ppt>>. Citado 3 vezes nas páginas 25, 26 e 37.

LEE, T. B. *Linked-data design issues*. 2009. W3C design issue document. [Http://www.w3.org/DesignIssues/LinkedData.html](http://www.w3.org/DesignIssues/LinkedData.html). Citado na página 42.

MCGUINNESS, V. H. D. Owl web ontology language - w3c recommendation. In: INSTITUTO DE PESQUISA ECONÔMICA APLICADA. 2004. Acesso em 19 de setembro de 2014. Disponível em: <<http://www.w3.org/TR/owl-features>>. Citado na página 42.

MEIRELLES, P. et al. *Evolução da Plataforma do Software Público Brasileiro*. 2014. Acesso em 20 de novembro de 2014. Disponível em: <http://fga.unb.br/articles/0000/6636/napsol_SPB_2014_10_16.pdf>. Citado na página 53.

OLIVEIRA, C. L. d. C. Renan Rodrigues de. Implementação de interoperabilidade entre repositórios digitais por meio do protocolo oai-pmh. 2009. Citado 3 vezes nas páginas 38, 41 e 58.

REPÚBLICA, S. N. d. A. S. Secretária Geral da Presidência da. *Conselhos Nacionais*. first. [S.l.]: Secretaria Geral da Presidência da República, 2010. Citado 2 vezes nas páginas 19 e 20.

ROCHA, J. C. A participação popular na gestão pública no brasil. *Jus Navigandi*, may 2011. Acesso em 06 de maio de 2014. Disponível em: <<http://jus.com.br/artigos/19205>>. Citado na página 19.

ROSENFELD, L.; MORVILLE, P. *Information Architecture for the World Wide Web*. [S.l.]: O'Reilly, 2002. (O'Reilly Series). ISBN 9780596000356. Citado 2 vezes nas páginas 28 e 29.

ROZANSKI, N.; WOODS, E. *Software Systems Architecture: Working With Stakeholders Using Viewpoints and Perspectives*. [S.l.]: Addison-Wesley Professional, 2005. ISBN 0321112296. Citado na página 52.

SCUASSANTE, P. M. A participação popular, prevista na constituição federal de 1988, garante efetivamente a realização do estado democrático de direito? *Âmbito Jurídico*, nov 2009. Acesso em 05 de maio de 2014. Citado na página 19.

-
- SILVA, V. M. da. Revisão sistemática da evolução mvc na base acm. *15º Concurso de Trabajos Estudiantiles*, 2012. Citado na página 33.
- SOUZA, M. I. F.; VENDRUSCULO, L. G.; C, M. G. Metadados para a descrição de recursos de informação eletrônica: utilização do padrão dublin core. Abril 2000. Citado na página 30.
- SOUZA, R. R.; ALVARENGA, L. A web semântica e suas contribuições para a ciência da informação. Abril 2004. Citado 2 vezes nas páginas 43 e 45.