

UNIVERSIDADE FEDERAL DO RIO DE JANEIRO  
INSTITUTO DE MATEMÁTICA  
DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO

DOUGLAS FONSECA ALVES PARANHOS  
EDUARDO DE OLIVEIRA ANDRADE

**DATA-TRIPLIFY: UM MÓDULO PARA TRIPLIFICAÇÃO DE  
DADOS NO WORDPRESS**

Profa. Maria Luiza Machado Campos, Ph. D.  
Orientadora

## **Data-triplify: um módulo para triplificação de dados no WordPress**

Douglas Fonseca Alves Paranhos

Eduardo de Oliveira Andrade

Projeto Final de Curso submetido ao Departamento de Ciência da Computação do Instituto de Matemática da Universidade Federal do Rio de Janeiro como parte dos requisitos necessários para obtenção do grau de Bacharel em Ciência da Computação.

Apresentado por:

---

Douglas Fonseca Alves Paranhos

---

Eduardo de Oliveira Andrade

Aprovado por:

---

Profa. Maria Luiza Machado Campos, Ph.D.  
(Orientadora)

---

Alan Freihof Tygel, M.Sc.

---

Profa. Giseli Rabello Lopes, D.Sc.

---

Prof. Sergio Manuel Serra da Cruz, D.Sc.

---

Kelli de Faria Cordeiro, D.Sc.

## **AGRADECIMENTOS**

Agradeço a toda minha família, que me apoiou durante todo o tempo de construção deste trabalho, bem como meus amigos e professores.

Agradecimento mais que especial para meus pais, Maria Auxiliadora e Lincoln por todo o apoio não só durante este trabalho mas sempre. Agradeço também aos orientadores deste trabalho, Maria Luiza e Alan Tygel pela imensa paciência e compreensão sempre.

Agradecimento especial também para meus amigos que sempre estiveram comigo André Santos e Luan Garrido. Agradeço também ao meu amigo que me ajudou neste projeto Eduardo Andrade.

Douglas Fonseca Alves Paranhos

## **AGRADECIMENTOS**

Agradeço a minha família, amigos, namorada e todos os outros que me apoiaram ou contribuíram de alguma forma para me dar a força necessária para a conclusão deste projeto.

Agradeço aos professores, funcionários e outros estudantes da UFRJ que estiveram presentes durante esta caminhada para a conclusão do bacharelado.

Agradeço ao meu amigo que dividiu este projeto comigo, Douglas, por todo o esforço realizado e por acreditar até o final na conclusão do trabalho.

Agradeço aos orientadores por estarem conosco neste processo e nos dando as informações necessárias para o desfecho do mesmo.

Agradeço a todas as pessoas que disponibilizaram conteúdos essenciais para o projeto, mesmo sem conhecê-las.

Muito obrigado por tudo, pessoal.

Eduardo de Oliveira Andrade

## RESUMO

PARANHOS, D. F. A.; ANDRADE, E. O. **Data-triplify: um módulo para triplificação de dados no WordPress.** Rio de Janeiro, 2016. Trabalho de Conclusão de Curso (Bacharel em Ciência da Computação) – Departamento de Ciência da Computação, Instituto de Matemática, Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2016.

A maioria dos dados atualmente se apresentam de uma forma pouco estruturada e escalável. Diante do aumento da quantidade de informações, é preciso que os dados estejam organizados de tal maneira que sejam capazes de lidar com o crescimento e manipulação destas informações. Com o advento da Web Semântica, novos métodos e grafos são responsáveis por uma estruturação que provê uma integração maior dos dados. Para que os dados interligados e a Web Semântica se tornem mais presentes, são necessárias iniciativas que popularizem estas ideias. A existência de programas que auxiliem na triplificação dos dados ainda é bastante limitada e muitas vezes de difícil manuseio por parte dos usuários. A proposta deste trabalho busca contribuir neste sentido, através de um *plug-in* que facilite o processo de triplificar os dados por meio de uma interface amigável e bastante intuitiva. Este *plug-in* foi desenvolvido para uma ferramenta bastante conhecida, o WordPress. O projeto "Proprietários do Brasil" é utilizado como exemplo de aplicação e mostra o que este trabalho é capaz de contribuir, incluindo o processo de triplificar qualquer tipo de dado em uma base de dados no WordPress e retornar o resultado em diversos formatos distintos.

**Palavras-chave:** Dados Ligados Abertos, Web Semântica, Processo de Triplificação de Dados Ligados Abertos, WordPress, Plug-in

## **ABSTRACT**

PARANHOS, D.; ANDRADE, E. O. **Data-triplify: a module for triplification of data in WordPress.** Rio de Janeiro, 2016. Undergraduate Conclusion Course Work (Bachelor of Computer Cience) – Computer Science Department, Mathematics Institute, Federal University of Rio de Janeiro, Rio de Janeiro, 2016.

Most of the data currently present a loosely structured and scalable manner. Faced with the increased amount of information, it is necessary that the data is organized in such a way that they are able to cope with the growth and manipulation of this information. With the advent of the Semantic Web, new methods and graphs are responsible for a structure that provides greater integration of data. For the linked data and the Semantic Web become more present, initiatives are necessary to popularize these ideas. The existence of programs to assist in the data triplification is still quite limited and often unwieldy for users. The purpose of this study aims to contribute in this direction, through a plug-in that facilitates the process of data triplify over a friendly and intuitive interface. This plug-in was developed for a well-known tool, WordPress. The project "Proprietários do Brasil" is used as a sample application and serves as a demonstration for what this work is able to contribute, including the process of data triplify for any type of data in a database in WordPress and return the results in several different formats.

**Keywords:** Linked Open Data, Semantic Web, Linked Open Data Triplification Process, WordPress, Plug-in

## **LISTA DE ABREVIATURAS E SIGLAS**

CKAN - *Comprehensive Knowledge Archive Network*  
CMS - *Content Management System*  
CSV - *Comma-separated values*  
ETL - *Extract, Transform and Load*  
ETL4LOD - *Extract, Transform and Load for Linked Open Data*  
FOAF - *Friend of a Friend*  
GRDDL - *Gleaning Resource Descriptions from Dialects of Languages*  
INDA - Infraestrutura Nacional de Dados Abertos  
JCE - Joomla Content Editor  
LD - *Linked Data*  
LOD - *Linked Open Data*  
NPG - *Nature Publishing Group*  
OGD - *Open Government Data*  
ODS - *Operational Data Store*  
OO - *Organization Ontology*  
OWL - *Web Ontology Language*  
PDF - *Portable Document Format*  
PF - Polícia Federal  
POWDER - *Protocol for Web Description Resources*  
RDDL - *Resource Directory Description Language*  
RDF - *Resource Description Framework*  
RDFa - *Resource Description Framework in Attributes*  
RDFS - *Resource Description Framework Schema*  
RIF - *Rule Interchange Format*  
ROV - *Registered Organization Vocabulary*  
R2RML - *RDB To RDF Mapping Language*  
SKOS - *Simple Knowledge Organization System*  
SPARQL - *SPARQL Protocol and RDF Query Language*

UE - União Européia

URI - *Universal Resource Identifier*

W3C - *World Wide Web Consortium*

XML - *Extensible Markup Language*

## **LISTA DE FIGURAS**

Figura 1 - data.gov e data.gov.uk, os sites de OGD norte-americano e britânico que estimularam o Portal Brasileiro de Dados Abertos .....	16
Figura 2 - dados.gov.br, o site de OGD do governo brasileiro .....	17
Figura 3 - Exemplo simples de uma ontologia sobre controle e propriedade .....	21
Figura 4 - Esquema mostrando as diferenças entre a <i>Web</i> e a <i>Web Semântica</i> .....	22
Figura 5 - Propriedade da classe <i>post</i> .....	24
Figura 6 - Ontologia para o tipo <i>whoowns_owner</i> .....	25
Figura 7 - Pequeno exemplo de triplas RDF como um modelo de dados em grafo.....	28
Figura 8 - Um grafo RDF grande exibido no <i>software</i> RDF Gravity .....	30
Figura 9 - A tela de buscas de temas do WordPress e suas opções de filtros .....	35
Figura 10 - Linhas de saída após a transformação da ferramenta Kettle sobre uma planilha do banco de dados do Proprietários do Brasil .....	41
Figura 11 - Exibição da interface do módulo que permite a realização das consultas SPARQL43	
Figura 12 - Esquema do processo de ETL4LOD .....	46
Figura 13 - Esquema do processo do <i>plug-in</i> .....	46
Figura 14 - Módulo de URL base, escolha de tipo de <i>post</i> e correspondências do <i>plug-in</i> ....	47
Figura 15 - Informação de inexistência de tipo pelo <i>plug-in</i> .....	48
Figura 16 - Dados retornados do tipo <i>post</i> .....	48
Figura 17 - Parte do módulo onde são definidas as correspondências.....	49
Figura 18 - Link para os dados no formato JSON-LD .....	50
Figura 19 - Módulo de <i>upload</i> de arquivo CSV do <i>plug-in</i> .....	51
Figura 20 - Erro na linha 1 durante <i>upload</i> de arquivo .....	51
Figura 21 - Erro na correspondência durante <i>upload</i> de arquivo .....	51
Figura 22 - Módulo de cadastro de prefixos e URI's do <i>plug-in</i> .....	52
Figura 23 - Dados de saída no formato JSON-LD .....	54
Figura 24 - Dados de saída no formato XML .....	55
Figura 25 - Dados de saída no formato RDF .....	55
Figura 26 - Inserção da URL e escolha de um post-type pelo usuário.....	57
Figura 27 - Inserção da URI e definição das correspondências pelo usuário .....	58
Figura 28 - Resultado do exemplo de aplicação no formato JSON-LD do tipo <i>post</i> .....	59
Figura 29 - Resultado do exemplo de aplicação no formato XML do tipo <i>post</i> .....	60
Figura 30 - Resultado do exemplo de aplicação no formato RDF do tipo <i>post</i> .....	61
Figura 31 - Escolha da URL e tipo de post-type a ser procurado .....	62
Figura 32 - Entrada das correspondências equivalentes entre o banco e a ontologia .....	63
Figura 33 - Resultado do exemplo de aplicação no formato JSON-LD do tipo <i>whoowns_owner</i> .....	64
Figura 34 - Resultado do exemplo de aplicação no formato XML do tipo <i>whoowns_owner</i> ..	65
Figura 35 - Resultado do exemplo de aplicação no formato RDF do tipo <i>whoowns_owner</i> ..	65

# Sumário

1	Introdução.....	12
1.1	Motivação e objetivos.....	12
1.2	Desafio dos dados abertos no Brasil.....	15
1.2.1	Proêmio do Portal Brasileiro de Dados Abertos.....	15
1.2.2	Opiniões e exemplos.....	17
1.3	Estrutura da monografia .....	18
2	Princípios da <i>Web Semântica</i> .....	19
2.1	Conceitos fundamentais da <i>Web Semântica</i> .....	19
2.2	Ontologias.....	21
2.2.1	Conceito .....	21
2.2.2	Ontologias propostas.....	23
2.2.2.1	Ontologia para o tipo <i>post</i> .....	23
2.2.2.2	Ontologia para o tipo <i>whoowns_owner</i> .....	24
2.2.3	Exemplos das principais ontologias.....	25
2.3	RDF .....	27
2.3.1	Definição.....	27
2.3.2	Triplificação.....	28
2.3.3	Banco de dados e os formatos JSON, XML e RDF.....	31
3	Revisão da literatura.....	34
3.1	WordPress.....	34
3.1.1	História.....	34
3.1.2	Visão Geral .....	34
3.1.3	Estrutura/Organização dos dados.....	36
3.2	Kettle .....	36
3.3	Drupal .....	40
3.4	<i>Plug-ins</i> disponíveis .....	43
3.5	Tecnologias utilizadas .....	44
4	A elaboração do <i>plug-in</i> .....	45
4.1	Especificação .....	45
4.2	Processo .....	45
4.3	Utilização .....	46
4.4	Implementação .....	52
4.5	Saídas.....	54
5	Exemplos de aplicação .....	56
5.1	Tipo <i>post</i> .....	56
5.1.1	URL base e post-type.....	56
5.1.2	URI e suas correspondências .....	57
5.1.3	Resultados .....	58
5.1.3.1	Resultado em JSON-LD.....	58
5.1.3.2	Resultado em XML .....	59
5.1.3.3	Resultado em RDF .....	60
5.2	Tipo “ <i>whoowns_owner</i> ”.....	61
5.2.1	URL base e post-type.....	61
5.2.2	URI e suas correspondências .....	62

5.2.3 Resultados .....	63
5.2.3.1 Resultado em JSON-LD.....	64
5.2.3.2 Resultado em XML .....	64
5.2.3.3 Resultado em RDF .....	65
6 Conclusão .....	66
Referências .....	67

# 1 Introdução

Os aspectos introdutórios que serão levantados nesta seção incluem a motivação e objetivos deste projeto, considerações sobre os desafios envolvendo dados abertos interligados e a história de alguns casos de sucesso pelo mundo.

## 1.1 Motivação e objetivos

A política no Brasil tem sido cada vez mais desacreditada. Considera-se que há muita corrupção no meio político, e essas opiniões são corroboradas por operações da Polícia Federal que desmontam esquemas de lavagem de dinheiro, evasão de divisas, desvio de verbas públicas, tráfico de influências, improbidade administrativa entre tantos outros, como são os casos da Operação Chacal<sup>1</sup>, Operação Satiagraha<sup>2</sup> e Operação Sanguessuga<sup>3</sup>. Além destas operações, tivemos diversas CPIs com o mesmo propósito, como a CPI dos Correios<sup>4</sup>, CPI do Mensalão<sup>5</sup> e CPI do Cachoeira<sup>6</sup>. As recentes denúncias de corrupção na Petrobras também têm ajudado a manter o clima de desconfiança. Estas denúncias estão sendo investigadas por diversas operações, como a Lava Jato<sup>7</sup> da PF, CPI e CPMI da Petrobras<sup>8</sup>.

Dante deste quadro conturbado, temos a mobilização de diversas pessoas, com a criação de sites estimulando o combate à corrupção e manifestações nas ruas, como as de Junho de 2013<sup>9</sup>. Vale a pena ressaltar que não são movimentos apenas contra a corrupção, mas também pela reivindicação de direitos como educação, saúde e melhores salários para trabalhadores e trabalhadoras.

É sabido que estes alicerces vêm de um sistema capitalista que é dominante faz muito tempo e que passamos a adotá-lo como se fosse o padrão (SINGER, 2002). Diante de um quadro onde a tendência é o crescimento da desigualdade, medidas para amenizar as disparidades consequentes do sistema são necessárias.

O esclarecimento dos indivíduos pode acarretar uma sociedade com maior senso crítico, aumentando as chances de contorno de problemas de submissão e

---

1 <http://www2.camara.leg.br/camaranoticias/noticias/125734.html>

2 [https://pt.wikipedia.org/wiki/Operação\\_Satiagraha](https://pt.wikipedia.org/wiki/Operação_Satiagraha)

3 [https://pt.wikipedia.org/wiki/Escândalo\\_dos\\_Sanguessugas](https://pt.wikipedia.org/wiki/Escândalo_dos_Sanguessugas)

4 [https://pt.wikipedia.org/wiki/CPMI\\_dos\\_Correios](https://pt.wikipedia.org/wiki/CPMI_dos_Correios)

5 [https://pt.wikipedia.org/wiki/Escândalo\\_do\\_Mensalão](https://pt.wikipedia.org/wiki/Escândalo_do_Mensalão)

6 [https://pt.wikipedia.org/wiki/CPMI\\_do\\_Cachoeira](https://pt.wikipedia.org/wiki/CPMI_do_Cachoeira)

7 <http://www1.folha.uol.com.br/poder/2014/11/1548049-entenda-a-operacao-lava-jato-da-policia-federal.shtml>

8 <http://legis.senado.leg.br/comissoes/comissao?codcol=1801>

9 [https://pt.wikipedia.org/wiki/Protestos\\_no\\_Brasil\\_em\\_2013](https://pt.wikipedia.org/wiki/Protestos_no_Brasil_em_2013)

estímulo de crescimento do próprio país. Por intermédio do conhecimento e da educação que sozinha não é capaz de modificar a sociedade, mas sem ela sequer teremos alguma transformação (FREIRE, 2000), temos a base para as mudanças que precisamos.

O projeto Proprietários do Brasil<sup>10</sup> é uma iniciativa que visa “mapear” as pessoas detentoras de influência por intermédio de ações de empresas, não necessariamente poder econômico. Outra questão importante, é o financiamento privado de campanha, que é a forma oficial com que as empresas influenciam o governo. Com isso, busca-se informar as pessoas da necessidade do movimento da reforma política, cuja principal bandeira é o fim do financiamento privado e uma maior transparéncia, característica fundamental para nossa democracia.

Apesar da relevância, o Proprietários do Brasil é apenas um caso de uso para este trabalho. A proposta pode incluir outras bases além do Proprietários do Brasil. A questão de sua escolha foram as características que se associaram com nossa motivação.

Logo, uma maneira bastante viável de estender essa transparéncia e facilitar o acesso aos dados sobre esse mapeamento, se dá com a utilização da *Web Semântica*. A disponibilização das informações dada pela *Web Semântica* ocorre com significados adicionais, de forma a contextualizá-los e torná-los interpretáveis por máquinas, permitindo que agentes e pessoas possam trabalhar em cooperação (BERNERS-LEE et al., 2001). As iniciativas de publicação de dados em formato de triplas RDF, no que se cunhou chamar de *Linked Open Data* (LOD), teve definição pelo portal Europeana como uma forma de publicação aberta de dados estruturados. Nesta forma, temos representações distintas de uma mesma informação podem ser encontradas e ligações entre os recursos são estabelecidas, ou seja, dados fornecem informações sobre outros dados de maneira interligada.<sup>11</sup>

A proposta da *Web Semântica* atinge nossos objetivos, diante da ideia de disponibilidade e transparéncia para todos e ainda mais, para países em desenvolvimento como o nosso. Como definiu o próprio Comitê Gestor da Internet no Brasil, “o objetivo final é ser capaz de interpretar os dados de maneira uniforme entre as diferentes organizações e plataformas envolvidas na troca de dados”.

---

10 <http://proprietariosdobrasil.org.br/>

11 <http://labs.europeana.eu/api/linked-open-data-introduction>

Os vocabulários ou ontologias são detalhados posteriormente na subseção 2.2 e foram parte fundamental da concepção do trabalho. Serviram como motivação aqueles que apresentaram termos consistentes com nossos exemplos de aplicação.

Como a sugestão para este projeto final foi baseada principalmente na capacidade de procurar atingir um grande número de usuários, analisamos diferentes ferramentas para conseguir o que desejávamos. Um leque de opções boas e gratuitas poderia satisfazer nossa necessidade e traçamos nossas metas fazendo uso das que consideramos mais interessantes.

Aproveitando a falta de recursos referentes à LOD disponibilizados para a ferramenta WordPress e da necessidade de diversos usuários trabalharem com estes tipos de dados, o que será bastante abordado neste trabalho, e percebendo sua crescente utilização para diversos fins, este trabalho objetiva a criação de um *plug-in* para WordPress.

Depois da instalação do *plug-in*, o usuário poderá realizar o processo de triplificação dos dados armazenados da base desejada. Para isto, é preciso definir os parâmetros necessários. O processo ocorre através de uma interface amigável no WordPress que facilita o uso por parte dos administradores de *blogs*.

## **1.2 Desafio dos dados abertos no Brasil**

Será abordado nesta subseção os obstáculos que precisam ser superados. O portal nacional de dados abertos também será brevemente descrito com algumas de suas características e por fim, opiniões serão levantadas sobre o tema e exemplos que vão embasar a importância de levar esta questão para frente.

### **1.2.1 Proêmio do Portal Brasileiro de Dados Abertos**

Os desafios em nosso país são enormes. A questão da transparência traz à tona outros aspectos que precisam ser melhorados e um processo de mudança de cultura. É preciso toda uma infraestrutura para que estes dados sejam disponibilizados da melhor maneira possível, o que necessita de recursos financeiros, pessoas para trabalhar e uma burocracia muito grande. As informações que são vistas no site Proprietários do Brasil por exemplo, precisaram de um grande esforço para serem obtidas. É preciso debater constantemente com os órgãos e agências que possuem estes dados. Através desta atitude, procurar formas de sobrepujar os entraves que as nossas próprias instituições delimitam.

Outro ponto é o processo de mudança de cultura. No caso, estamos falando sobre aumentar o acesso da população em verificar estes dados abertos interligados de governo. Sem o estímulo, de nada adianta ter os dados se eles não serão visualizados. Logo, é preciso conscientizar os cidadãos sobre a importância do tema. Com isto, eles poderão usufruir melhor dos serviços que o seu governo fornece e exigir os compromissos que o Estado deve cumprir. Assim, teremos cidadãos mais informados, que possam contribuir mais com a sociedade em que vivem, ajudar no combate à corrupção e passar a entender e aceitar melhor os órgãos governamentais e o trabalho dos políticos em geral.

Diante do referido quadro, temos o Portal Brasileiro de Dados Abertos que foi estimulado pelos casos de sucesso que ocorreram nos governos norte-americano e britânico (Figura 1) na procura da democratização do acesso aos dados e informações. Mesmo com o nosso conjunto de dados sendo quantitativamente inferior, o Brasil já deu importantes passos. Em novembro de 2011 tivemos uma lei sancionada, a Lei de Acesso à Informação (12.527/2011) cujo propósito é de regulamentar o direito constitucional de acesso das pessoas às informações públicas e desde então, a INDA (Infraestrutura Nacional de Dados Abertos) vem sendo desenvolvida para publicar e oferecer busca e acesso aos dados públicos brasileiros.

## The home of the U.S. Government's open data

Here you will find data, tools, and resources to conduct research, develop web and mobile applications, design data visualizations, and more.

GET STARTED  
SEARCH OVER 125,051 DATASETS

Manufacturing & Trade Inventories & Sales

## Figura

The screenshot shows the DATA.GOV.UK homepage. At the top, there is a navigation bar with links for Home, Data, Apps, Interact, and a search bar. Below the navigation is a large graphic featuring the letters 'NII' in a large, bold font. To the right of the letters, there is a section titled 'NOTIFICATION OF PURDAH' with a portrait of a man. Below this, there are several smaller sections: 'The National Information Infrastructure exemplars work' (with a photo of a woman), 'What this means for DGU' (with a photo of a person), 'Four challenges for the future of Open Data' (with a photo of a house icon), 'ODUG on the NII' (with a photo of a woman), 'Public servants, open data wants you!' (with a photo of a person), and 'Housing Land Availability Assessment' (with a photo of a house icon). At the bottom of the page, there is a footer with links for FAQ, Code of conduct, Accessibility, Cookies policy, Privacy, Moderation policy, Contact, Terms and conditions, API, and About.

I - data.gov e data.gov.uk, os sites de OGD norte-americano e britânico que estimularam o Portal  
Fonte: Print screen do site data.gov<sup>12</sup> e data.gov.uk<sup>13</sup>

Podemos verificar no site brasileiro de OGD (Figura 2), que já temos diversos grupos de dados, entre eles, educação, trabalho, saúde, etc. Também se nota a disponibilização dos dados em formatos não proprietários, como XML, CSV, ODS, entre outros. Ainda há uma parte relativa a aplicativos e serviços que foram desenvolvidos com o uso de dados abertos que são baseados nos três pilares do governo aberto, como consta no site: transparência, credibilidade e prestação de contas e participação. Qualquer pessoa que tenha desenvolvido ou conheça algum aplicativo, pode informar o dados.gov.br. O site então, poderá disponibilizá-lo na sua lista de aplicativos.

12 <https://www.data.gov/>

13 <https://www.data.gov.uk/>

**Figura**

The screenshot shows the homepage of the **dados.gov.br** portal. At the top, there's a navigation bar with links like 'Ir para o conteúdo', 'Ir para o menu', 'Ir para a busca', 'Ir para o rodapé', 'Participe', 'Serviços', 'Legislação', and 'Canais'. Below the header, there's a search bar with placeholder text 'Pesquisa ...' and a button 'PESQUISAR'. A message indicates 'em 497 conjuntos de dados com 4355 recursos ([o que é isto?](#))'. The main content area has several sections: 'Dados em destaque' (with links to 'Compras públicas do governo federal', 'Reclamações do Consumidor.gov.br', and 'SICONV - Convênios e Contratos de Repasse da Administração ...'); 'Publicações mais recentes' (listing documents from various ministries); 'Dados por etiqueta' (a tag cloud including PGI, Indicador, INDE, Previdência Social, Benefícios, Benefícios Previdenciários, Contribuintes, UF, GFIP, Equipamentos públicos, federal, INEP, GPS, Receitas, SP, IBGE, fornecedores, avaliação educação, and Vegetação); and 'Notícias' (with links to news articles from the Ministry of Planning and the Ministry of Justice).

Fonte: Print screen do site [dados.gov.br](http://dados.gov.br)<sup>14</sup>

### 1.2.2 Opiniões e exemplos

Para corroborar o que já foi escrito e ter uma boa perspectiva diante do que pode ser alcançado com os dados abertos, podemos analisar algumas opiniões e exemplos. Houve *workshops* no estado de São Paulo patrocinados pelo governo britânico em março de 2015 sobre LOD. Vale a pena mencionar que os britânicos possuem o maior portal de dados abertos do mundo. Porém, o que desejamos saber são soluções para o nosso país e a Eleanor Stewart, uma das figuras mais importantes no assunto de transparência e transformação digital, citou nestes *workshops* um problema comum entre os diversos casos de abertura de dados<sup>15</sup>.

Seriam estas informações realmente benéficas para a sociedade. Parecem úteis, porém, como afirmar o quanto relevante podem ser? O que Eleanor Stewart acabou dizendo diretamente é que se procurarmos razões para abrir os dados, poderemos nunca os disponibilizar, então, deveríamos simplesmente fazer isto e pronto.

A questão de abrir os dados sem um motivo bem elucidado, traz um ponto de desconfiança para muitas pessoas. É preciso realizar uma transformação no modo

14 <http://dados.gov.br>

15 <http://br.okfn.org/2015/04/08/dados-abertos-estado-cidadao-ou-beltrao-2-0/>

de vermos os dados. Os dados abertos são um fator fundamental e que já obtiveram sucesso na Grã-Bretanha, por exemplo. Redução em custos de viagens e saúde são alguns dos serviços que foram melhorados pelo OGD britânico e esperamos obter as mesmas vantagens no Brasil.

Yasodara Córdova, membro do W3C (principal organização responsável pelos padrões adotados na *Web*), criticou a nossa política de dados abertos<sup>16</sup>, dizendo que não há esforço nem uma estratégia para abertura dos dados. Entretanto, reconheceu melhorias apesar de lentidão demasiada, além de pouco planejamento. Para fechar, temos o exemplo de acessos que aumentaram em cerca de 90% de 2012 até 2013 no portal brasileiro, dados.gov.br. É possível sim, esperar um futuro melhor acerca de LOD no nosso país.

A proposta do Proprietários do Brasil e do *plug-in* é de acrescentar mais um meio e ferramenta para esta disseminação de informações da base nacional. Isto seria dado com o uso de casos bem-sucedidos e fazendo da nossa própria maneira, de acordo com nossa sociedade. Esta oferta do livre uso dos dados para os cidadãos, então, poderá trazer melhorias para o país e ser um novo exemplo.

### **1.3 Estrutura da monografia**

Este trabalho está organizado em seis seções. Como verificou-se, a primeira seção foi introdutória, mostrando a motivação, portais de dados abertos, opiniões, exemplos e a melhora que procuramos com este trabalho. Na segunda seção, temos uma explicação dos princípios fundamentais relacionados à *Web Semântica* e suas aplicações.

A terceira seção é relacionada ao WordPress e suas nuances. Já a quarta seção, mostra como o *plug-in* foi desenvolvido e uma aplicação prática para ele. A quinta seção, corresponde aos nossos exemplos de aplicação. Para finalizar, a sexta e última seção apresenta as conclusões que esta monografia nos permite ter.

---

<sup>16</sup> <http://convergenciadigital.uol.com.br/cgi/cgilua.exe/sys/start.htm?infoid=35892>

## **2 Princípios da *Web Semântica***

Nesta seção serão abordados alguns conceitos da *Web Semântica*, o formato RDF e suas propriedades, além do papel de ontologias e vocabulários. Encerrando, serão vistas as dificuldades e como ocorre o processo de triplificação dos dados.

### **2.1 Conceitos fundamentais da *Web Semântica***

A *Web* é usada predominantemente com o propósito de buscar e usar informação, procurar e manter contato com pessoas, conhecidas ou não (BERNERS-LEE et al., 2001). Os principais problemas da *Web* são:

- Grande quantidade de resultados com uma baixa precisão. Quando se busca por algo, as vezes se obtém um número expressivo de páginas com pouca ou

nenhuma relevância, e um número baixíssimo de páginas que contenham as informações que o usuário está realmente buscando.

- Nenhum resultado ou resultado ineficiente. Em uma minoria dos casos, pode ser que, por ser muito específico ou muito particular a informação, a consulta não tenha resultados, ou as páginas que contenham o resultado não foram retornadas.
- Os resultados dependem do vocabulário. Às vezes, as palavras chave usadas na pesquisa, não representem realmente o que a busca procura, ou a informação procurada está colocada em outros termos, diferentes dos usados.
- A informação procurada pode estar espalhada por várias páginas da Web diferentes, tendo que ser coletados vários pedaços de informação por diferentes páginas e depois serem juntados para se obter a resposta desejada.

Sugere-se que, para resolver esses problemas não simplesmente se aplique técnicas de processamento, técnicas que usem *machine learning*, mas sim uma revolução na Web: a Web Semântica.

Ela é uma ampliação da Web, onde as pessoas e computadores poderão trabalhar juntos, de forma cooperativa. Busca-se atribuir um sentido as informações publicadas na Web, interligando-as e facilitando o entendimento pelas pessoas e computadores.

A Web Semântica tem como meta acrescentar o conteúdo disponibilizado dos documentos da Web em dados. Estes, através de uma arquitetura Web, poderiam ser acessados e estariam interligados, assim como os documentos já são atualmente.

Como um exemplo bem simplificado, poderíamos desejar saber a proporção de mulheres no Brasil que estão trabalhando em cargos políticos. Vamos supor que tivéssemos dois documentos na Web: um que informasse a população feminina no país e outro que mostrasse o nome dos políticos em atividade. Estes documentos disponíveis em DOC e outro em PDF. Para obter o que queremos, teríamos que cruzar estas informações. Porém, sabemos que nestes arquivos não seria uma tarefa viável a realização do cruzamento entre ambos, pelas diferenças dos próprios

formatos. Outro problema que pode acontecer é a perda de precisão da contagem. Algumas vezes temos nomes similares para ambos os sexos, como Dominique. Além disto, neste caso, mesmo copiando as informações nos arquivos PDF e DOC para uma ferramenta que possa procurar analisar e ser capaz de exibir algumas informações (Excel, por exemplo), não seria possível. Apenas com os nomes, a tarefa seria bastante árdua e pouco prática. Bem provável inclusive, de ter que ser feita manualmente.

O que foi citado no parágrafo anterior, representa como funciona a *Web* muitas vezes quando tentamos interligar informações. A *Web Semântica* resolveria o problema, disponibilizando a informação de outra forma, onde seria possível a ligação dos dados. As pessoas teriam a possibilidade de construir vocabulários e definirem regras para a interoperabilidade destes dados. Para cada político, já haveria o seu sexo como informação e seria possível interligar com a população do tipo feminina em tipos de dados padronizados que teriam como saída o que desejamos saber.

O comportamento da *Web Semântica* se dá como um banco de dados abrangente, com dados de várias partes do mundo. A consulta de todos estes dados precisa ser feita por algum tipo de linguagem, esta linguagem é a SPARQL.

Outra linguagem importante também, é a OWL<sup>17</sup>. Hoje em dia é a linguagem recomendada pela própria W3C, que é responsável pela criação dos vocabulários ou ontologias que servem para organizar os dados de um domínio qualquer. O raciocínio sobre os dados em cima destas regras estabelecidas é denominado inferência.

Particularmente, entre as duas linguagens que foram descritas, para consultas e vocabulários, a linguagem OWL é a mais importante para a criação do *plug-in* que é o objetivo deste projeto e claro, da própria ontologia que serve como base.

## 2.2 Ontologias

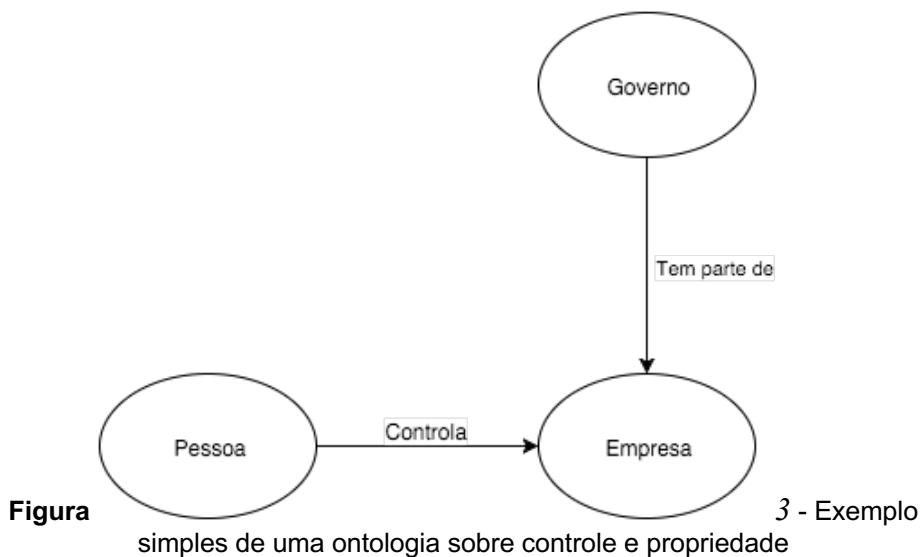
Quando se fala em *Web Semântica*, é de suma importância conceituar o que são as ontologias. Nesta subseção veremos a sua importância para a concepção das modelagens do universo dos dados abertos e interligados, que é de interesse deste projeto. Começaremos pela própria definição de ontologia.

---

17 <https://www.w3.org/2001/sw/wiki/OWL>

## 2.2.1 Conceito

A Web Semântica tem por base as ontologias ou vocabulários. Uma ontologia é uma modelagem de alto nível de como as entidades, ou classes, se relacionam entre si, em um determinado domínio. Segundo a definição de T.R. Gruber, "Uma ontologia é uma especificação explícita e formal da conceituação". Temos então, um exemplo de ontologia na Figura 3:

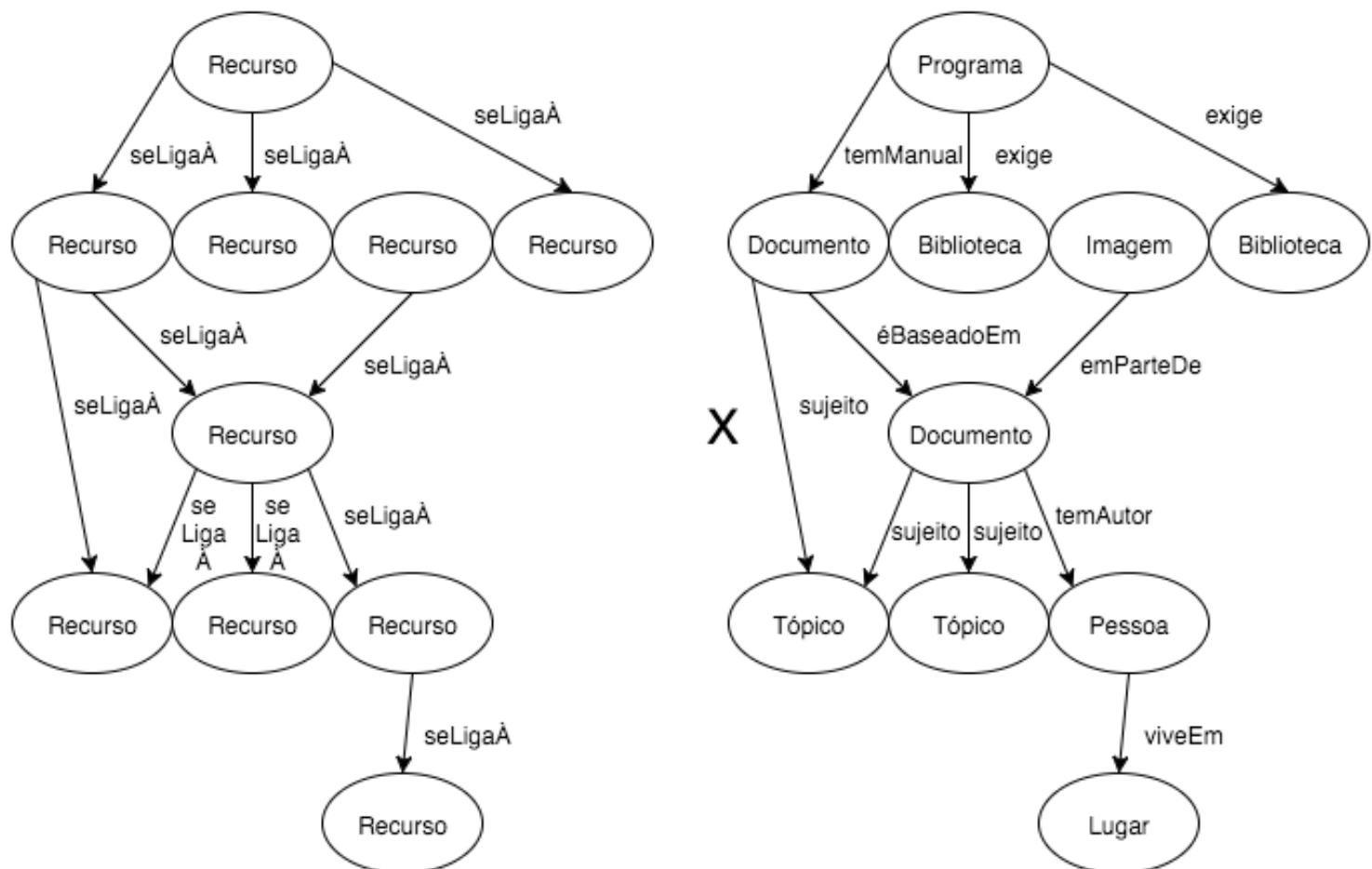


Uma entidade representa um tipo na vida real. “Pessoa”, do exemplo acima, no geral, seria uma entidade, enquanto a “Pessoa” de algo específico, seria uma instância de “Pessoa” portanto uma instância, ou um indivíduo, da entidade “Pessoa”.

Além de classes, tem-se atributos e relacionamentos. Atributos são propriedades ou características que cada classe tem, e pode compartilhar com outras classes. Relacionamentos são como cada classe se comunica uma com a outra, no exemplo acima, a classe “Pessoa” se relaciona com a classe “Empresa” através do relacionamento “Controla”, ou seja, “Pessoa” é um elemento, que por sua vez, é parte de uma “Empresa” que é parcialmente propriedade de “Governo”.

A representação da informação dada pela Web e Web Semântica possui algumas diferenças que são apresentadas na Figura 4:

**Figura**  
**Web x Web Semântica**



**Fonte:** Retirado do *blog* de Natzir Turrado. Traduzido pelos autores<sup>18</sup>

Além de ontologia, temos outras formas de classificação e hierarquização das informações como Vocabulário Controlado, Glossário, Taxonomia e Tesauro:

- Vocabulário controlado: É uma lista de termos explicitamente enumerados. Todos os termos em um vocabulário controlado devem ter uma definição única e não redundante.
- Glossário: É a lista de termos do vocabulário controlado com o significado de cada termo claro, sendo assim, algo semelhante à um dicionário.
- Taxonomia: É uma coleção de vocabulários organizados em uma estrutura hierárquica. Cada termo está em uma ou mais relações pai-filho com outros termos da mesma taxonomia. Podem haver diferentes tipos de

18 <http://www.analistaseo.es/posicionamiento-buscadores/seo-semanitico/>

relacionamentos pai-filho em uma taxonomia (Ex.: parte-todo, tipo-instância), entretanto as boas práticas sugerem apenas um tipo de relação pai-filho de um mesmo pai.

- Tesauro: É uma coleção de vocabulários interligados. Tesauro além de ter as relações pai-filho de uma taxonomia, contém relações associativas, ou seja, um tesauro tem muito mais expressividade que uma taxonomia.

## 2.2.2 Ontologias propostas

Foram elaboradas duas ontologias, uma para cada tipo de *post*: o próprio *post* derivado do WordPress e *whoowns\_owner*, do Proprietários do Brasil.

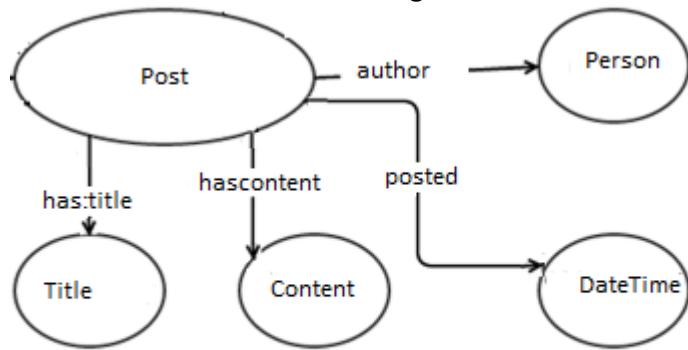
### 2.2.2.1 Ontologia para o tipo *post*

Uma ontologia simples foi modelada para este caso (Figura 5): o *post* tem um autor que no banco de dados corresponde à coluna “post\_author” e na ontologia corresponde ao “foaf:person”, já que o autor é uma pessoa. O *post* também tem uma data de postagem que na ontologia foi mapeado como “npg:dateTime” visto que a data de postagem é uma coluna do tipo “dateTime”. O NPG é uma empresa de publicações internacionais, o que inclui revistas e conteúdos acadêmicos, principalmente da área médica. Apesar disto, foi criada a NPG Core Ontology<sup>19</sup> que foi utilizada neste trabalho e pode servir para assuntos que divergem da medicina. Além destes dois campos, o *post* também possui um título, o que foi traduzido para a ontologia como “dc:title” e por último, mas não menos importante o que reflete o conteúdo do *post* na ontologia é dado por “owl:hasValue”.

---

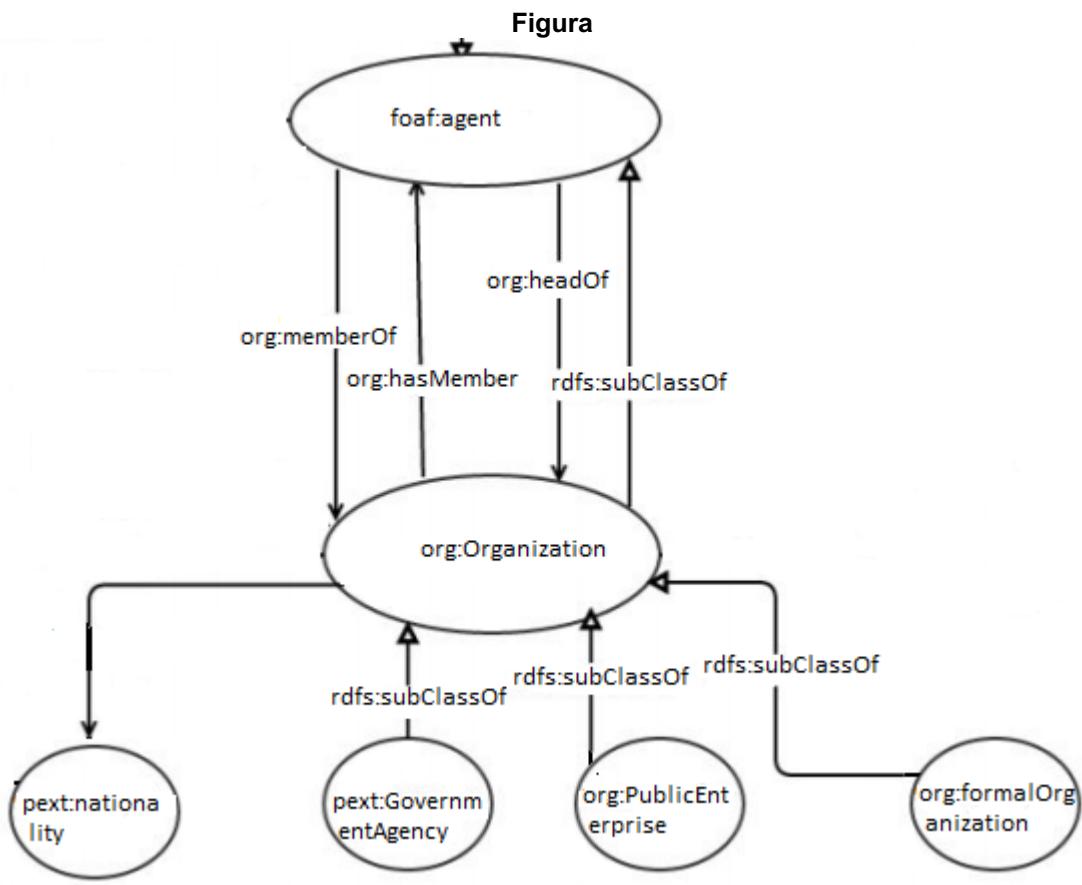
<sup>19</sup> <http://data.nature.com/downloads/latest/ttl/npg-core-ontology.ttl>

**Figura**



### 2.2.2.2 Ontologia para o tipo whoowns\_owner

A ontologia do tipo em questão tem as organizações (org:Organization), que podem ser do tipo empresas públicas (org:publicEnterprise), uma agência governamental (pext:GovernmentAgency) ou uma empresa (org:formalOrganization). As organizações possuem uma nacionalidade (pext:nationality). Organizações são agentes (foaf:Agent) mas um agente não necessariamente é uma organização. E por fim, um agente é o chefe ou quem detém o poder na organização (org:headOf) e agentes podem ser membros de organizações (org:memberOf). Além disto, organizações também podem ter agentes (org:hasMember), conforme a Figura 6:



### 2.2.3 Exemplos das principais ontologias

Não é preciso criar uma ontologia totalmente nova, o recomendado é fazer uso das ontologias existentes, de seus termos e apenas criar o que for preciso e que não esteja disponível na Web. Abaixo, temos alguns exemplos de ontologias importantes e que foram usadas para a criação deste projeto:

#### a) FOAF<sup>20</sup>

A ontologia *Friend of a Friend*, foi criada para a construção de relacionamentos entre pessoas e informações. Basicamente, esta ontologia tem uma abordagem que procura fazer com que os mais diferentes sites possam fazer uma descrição de algo maior em diversas partes também diferentes. Os usuários então, podem manter um certo nível de controle sobre suas informações. Segue um exemplo desta ontologia:

```

<foaf:Group>
  <foaf:name>ILRT staff</foaf:name>
  <foaf:member>
  
```

<sup>20</sup> <http://xmlns.com/foaf/spec/>

```

<foaf:Person>
    <foaf:name>Martin Poulter</foaf:name>
    <foaf:homepage
rdf:resource="http://www.ilrt.bris.ac.uk/aboutus/staff/staffprofile/?search=plmlp"/>
    <foaf:workplaceHomepage
rdf:resource="http://www.ilrt.bris.ac.uk/">
</foaf:Person>
</foaf:member>
</foaf:Group>

```

### b) ORG<sup>21</sup>

A *Organization Ontology* procura representar informações relativas as organizações e suas estruturas, inclusive organizações governamentais. Ela é bem geral e pode ser tanto estendida quanto especificada para diversos fins. Segue um exemplo desta ontologia:

```

<http://example.com/DfE> a org:Organization;
    skos:prefLabel "Department for Education"@en .
<http://example.com/DCSF> a org:Organization;
    skos:prefLabel "Department for Children Schools and
Family"@en .
<http://example.com/regorgMay2010> a org:ChangeEvent;
    rdfs:comment "Post-election re-organization and
rebranding"@en ;
    org:originalOrganization <http://example.com/DfE> ;
    org:resultingOrganization <http://example.com/DCSF> .

```

### c) RDFS<sup>22</sup>

*RDF Schema* é uma extensão da ontologia básica do RDF. Alguns mecanismos novos são utilizados para a descrição de alguns grupos de recursos e os relacionamentos entre eles. Segue um exemplo desta ontologia<sup>23</sup>:

---

21 <http://www.w3.org/TR/vocab-org/>

22 <http://www.w3.org/TR/rdf-schema/>

23 <http://www.cambridge semantics.com/semantic-university/rdfs-introduction>

```

csi:MyNewPersonClass
    rdfs:label "My Person Class" .

csi:controls
    rdf:type rdfs:Property ;
    rdfs:label "Controls" ;
    rdfs:domain csi:MyNewPersonClass ;
    rdfs:range csi:MyNewPersonClass

csienterprise:TestS.A
    rdf:type foaf:Person ;
    rdf:type csi:MyNewPersonClass ;
    rdfs:label "Test Corporation" ;

csi:people:Lee
    rdf:type csi:MyNewPersonClass;
    csi:controls csienterprise:TestS.A .

```

## 2.3 RDF

Após a introdução sobre *Web Semântica* e ontologias, é substancial falarmos sobre o RDF. Segue nesta parte do texto a definição do RDF e a sua relevância.

### 2.3.1 Definição

A sigla RDF significa “*Resource Description Framework*”, ou seja, o RDF é um *framework* que descreve recursos presentes na *Web* de uma forma feita para os computadores. Para as pessoas, não é um formato fácil de compreender, não foi criado com este propósito. Ele é escrito basicamente em XML, sendo o seu uso recomendado pela W3C.

O formato RDF permite definir as informações na *Web* com precisão e integrá-las. A padronização através do RDF é capaz de descrever diversos exemplos úteis que serão processados pelos computadores, como<sup>24</sup>:

- Produtos na *Web*, incluindo disponibilidade e preço.
- Horários de eventos da *Web*.
- Conteúdo para ferramentas de procura.

---

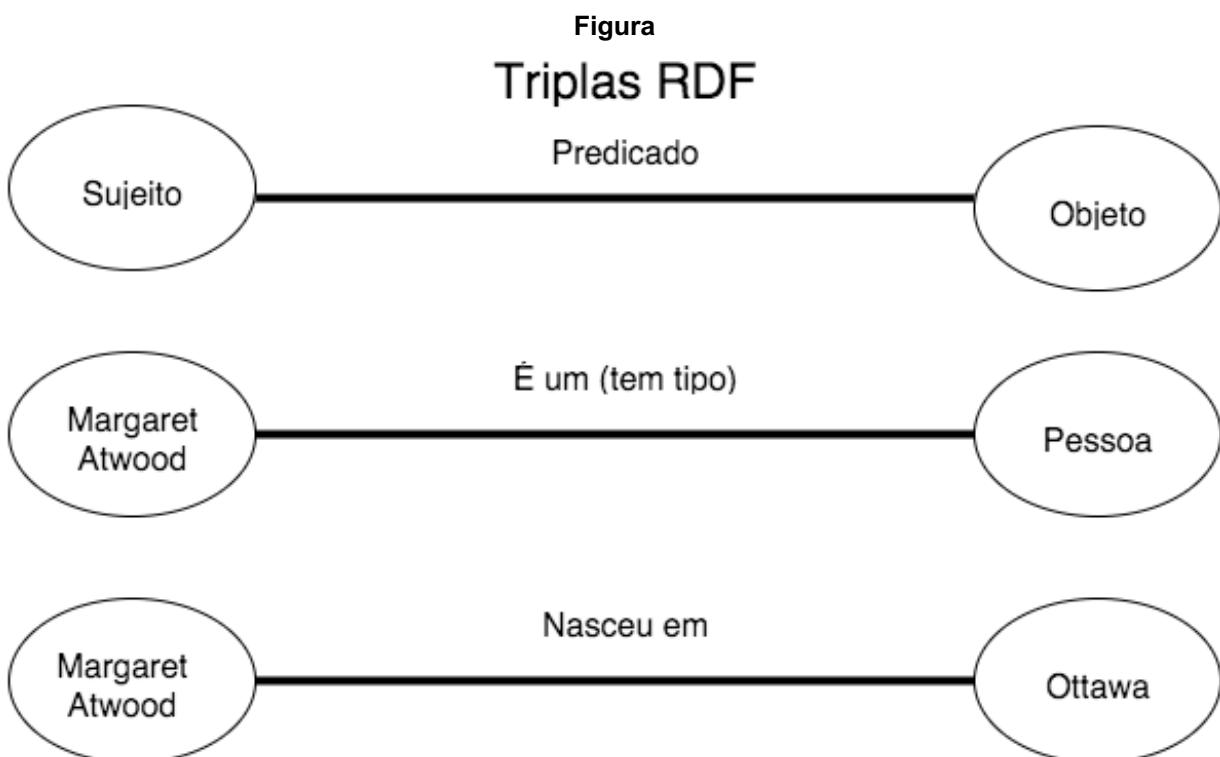
<sup>24</sup> [http://www.w3schools.com/webservices/ws\\_rdf\\_intro.asp](http://www.w3schools.com/webservices/ws_rdf_intro.asp)

- Informações sobre páginas da Web (autor, criação, data de modificação e conteúdo).

Também é importante mencionar que existe o RDFS, uma extensão do vocabulário básico do RDF. O RDFS é utilizado com a linguagem OWL e o SKOS, responsável por fornecer um padrão para a esquematização dos sistemas de organização de conhecimento. O RDFS serve para representar os vocabulários e dar suporte a ontologias, que serão vistas adiante, no decorrer desta subseção e de forma mais específica.

### 2.3.2 Triplificação

Para uma melhor concepção e visualização do processo de triplificação dos dados, podemos observar na Figura 7:



**Fonte:** Retirado do blog InsideOLITA. Traduzido pelos autores<sup>25</sup>

- O sujeito da tripla é o recurso que queremos descrever. Ele identifica o objeto que está sendo referido pela tripla. No caso, é dado pelo nome de uma pessoa chamada “Margaret Atwood”.

---

<sup>25</sup> <http://www.accessola2.com/olita/insideolita/wordpress/?p=60281>

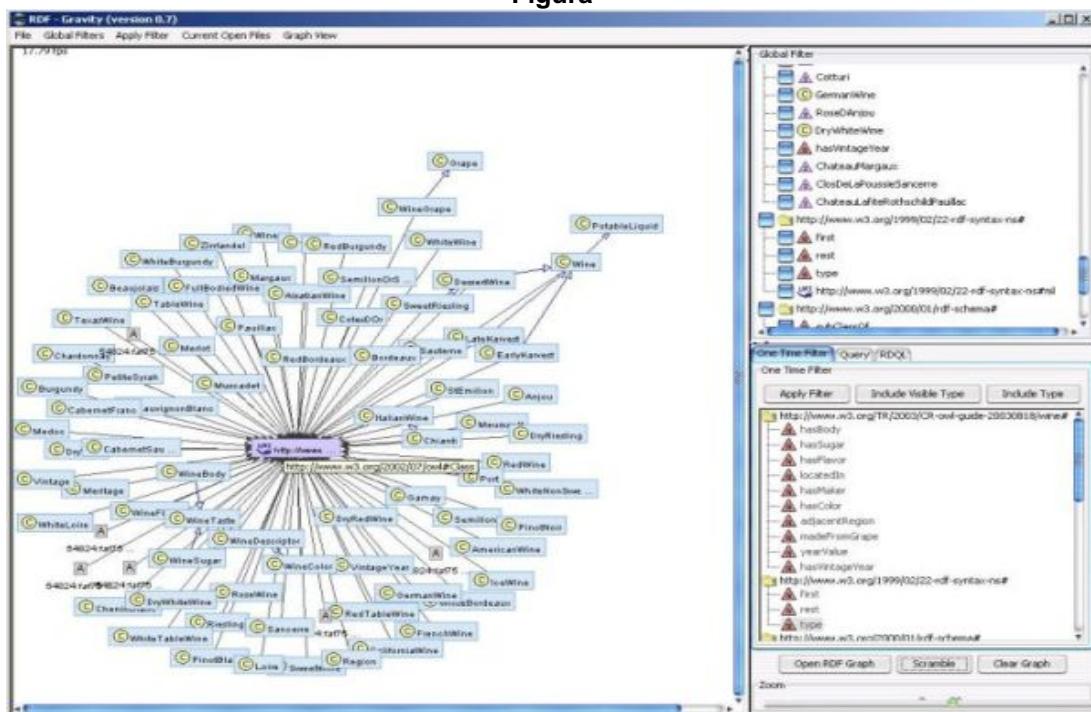
- O predicado consiste em representar a relação entre o sujeito e o objeto. Ele define a parte dos dados no objeto que vamos atribuir um valor. No caso, é dado por “É um (tem tipo)” e “Nasceu em”.
- Por último, objeto é o próprio valor de um determinado tipo de dado. No caso, é dado por “Pessoa” e “Ottawa”.
- Simplificando em um grafo, temos o sujeito como o nó fonte, o predicado como a legenda da aresta que conecta o nó fonte e o nó destino e o objeto como o nó destino.

Complementando, POWERS (2003) define as triplas RDF com as seguintes características:

- Cada tripla necessariamente é formada por um sujeito, predicado e objeto.
- Cada tripla é completa e exclusiva.
- Cada tripla é uma 3-tupla, formada por um sujeito, predicado e objeto. O sujeito pode ser uma referência URI ou um *blank node* (bnode), que é um nó representando um recurso RDF para qual uma URI ou literal não é atribuído. O predicado tem que ser uma referência URI e o objeto pode ser uma referência URI ou um bnode, assim como o sujeito.
- Cada tripla pode ser unida a outras triplas e mesmo que o modelo se torne muito complexo, cada uma das triplas manterá seu significado exclusivo.

Apesar do exemplo relativamente fácil de compreender da Figura 7, existem diversas dificuldades envolvendo o processo de triplificação de dados:

Figura



Fonte: Artigo sobre RDF<sup>26</sup>

Umas dessas dificuldades é a complexidade que os modelos podem atingir, como o exemplo da Figura 8, tornando alguns casos complicados de se ver até para aqueles que não são leigos. Outro problema é a forma com que os modelos são montados. Eles podem ser feitos através de um algoritmo que automatize o processo, o que necessitaria de um algoritmo muito inteligente (até hoje nada se conseguiu) ou manualmente, o que seria extremamente trabalhoso em alguns casos e inviável.

Outro ponto, é o benefício que alguns sites conseguiriam com os metadados. Não é muito fácil de mensurar o custo/benefício devido ao trabalho que será necessário para as conversões dos dados e com as poucas ferramentas disponíveis. Algoritmos inteligentes que automatizassem a tarefa seriam ideais, mas nenhum grande êxito foi obtido até agora.

Questões de padronização também podem atrapalhar e muito. Sites diferentes podem apresentar detalhes que atrasam e complicam o processo de triplificação dos dados. Um determinado site pode conter o número do volume de uma revista em forma de algarismo e outro, pode ter todas as informações similares, porém com

26 [http://swui.semanticweb.org/swui06/papers/Karger/Pathetic\\_Fallacy.html](http://swui.semanticweb.org/swui06/papers/Karger/Pathetic_Fallacy.html)

este número em forma de uma cadeia de caracteres, como cinquenta e seis, por exemplo. Seria preciso definir um padrão geral e isto é complicado de se aplicar diante do imenso volume de dados que temos hoje de diversos locais e culturas diferentes, que fazem o uso de diversas maneiras distintas de representação.

Desafios matemáticos como o problema de NP-completo, desafios lógicos que precisam lidar com incertezas, avaliação de custos, e outras questões. Muito já foi feita para melhorar o cenário e tentar resolver estas questões, mas ainda falta um longo caminho para o processo de triplificação dos dados ser melhor e realizado de uma forma mais simplificada.

### **2.3.3 Banco de dados e os formatos JSON, XML e RDF**

É importante salientar a existência dos bancos de dados que armazenam as triplas RDF. Através deles podemos extrair informações úteis e realizar diversas consultas. Abaixo, temos dois exemplos importantes e que foram analisados para a criação deste projeto:

- DBpedia: é uma base de dados bem grande, cuja proposta é converter para a Web Semântica uma grande fatia dos dados da Wikipedia. Consultas bem específicas podem ser feitas nesta base e ele é bastante referenciado por outros bancos, quando tratamos de RDF.
- GeoNames: aberto e público. O GeoNames é um banco de dados muito grande, sendo referência para dados geográficos em todo mundo.

O padrão RDF possui diversos formatos. Os formatos escolhidos para as saídas neste trabalho, foram o JSON-LD, XML e RDF.

Existem outros formatos, Turtle e N-Triples mas o JSON-LD por ter uma base do *Javascript*, foi um facilitador para o processo de triplificação, já que a ferramenta WordPress foi utilizada. Além disso, o JSON é uma linguagem difundida na *Web* e que podemos trabalhar juntamente com o PHP. Sendo o JSON-LD voltado para dados interligados, fortalece ainda mais o motivo de escolha. Mesmo caso para o XML, que é uma linguagem simples e que as pessoas estão acostumadas a trabalhar na *Web* faz muito tempo. Porém a interpretação do XML como RDF precisa de certo esforço. Por último, o RDF/XML que é chamado de apenas RDF por alguns e também neste trabalho. Ele é uma sintaxe baseado em XML e voltado para RDF.

Os bancos de dados mencionados fazem uso dos diversos formatos, mas neste

projeto criamos o processo apenas para as saídas em JSON-LD, XML e RDF. Os outros formatos ficaram como propostas para trabalhos futuros. Abaixo, alguns exemplos de saídas relativas ao *plug-in* nestes formatos mencionados:

a) JSON-LD<sup>27</sup>

```
{  
  "@context": {  
    "name": "http://xmlns.com/foaf/0.1/name",  
    "homepage": {  
      "@id":  
        "http://xmlns.com/foaf/0.1/workplaceHomepage",  
      "@type": "@id"  
    },  
    "Person": "http://xmlns.com/foaf/0.1/Person"  
  },  
  "@id": "http://me.markus-lanthaler.com",  
  "@type": "Person",  
  "name": "Markus Lanthaler",  
  "homepage": "http://www.tugraz.at/"  
}
```

b) XML<sup>28</sup>

```
<owlx:Class owlx:name="WineDescriptor"  
owlx:complete="false" />  
<owlx:Class owlx:name="WineColor" owlx:complete="false">  
  <owlx:Class owlx:name="#WineDescriptor" />  
</owlx:Class>  
<owlx:ObjectProperty owlx:name="hasWineDescriptor">  
  <owlx:domain owlx:class="Wine" />  
  <owlx:range owlx:class="WineDescriptor" />  
</owlx:ObjectProperty>  
<owlx:ObjectProperty owlx:name="hasColor">
```

---

27 <https://en.wikipedia.org/wiki/JSON-LD>

28 <http://www.w3.org/TR/owl-xmlysyntax/apd-example.html>

```

<owlx:range owlx:class="WineColor" />
</owlx:ObjectProperty>
<owlx:SubPropertyOf owlx:sub="hasColor">
  <owlx:ObjectProperty owlx:name="hasWineDescriptor" />
</owlx:SubPropertyOf>
```

c) RDF<sup>29</sup>

```

<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-
syntax-ns#" 
           xmlns:dc="http://purl.org/dc/elements/1.1/" 
           xmlns:ex="http://example.org/stuff/1.0/">
  <rdf:Description rdf:about="http://www.w3.org/TR/rdf-
syntax-grammar">
    dc:title="RDF1.1 XML Syntax">
    <ex:editor>
      <rdf:Description ex:fullName="Dave Beckett">
        <ex:homePage
          rdf:resource="http://purl.org/net/dajobe/" />
      </rdf:Description>
    </ex:editor>
  </rdf:Description>
</rdf:RDF>
```

---

29 <http://www.w3.org/TR/rdf-syntax-grammar/>

### **3 Revisão da literatura**

Nesta seção são apresentadas revisões sobre o WordPress e a ferramenta Kettle, o que inclui ETL4LOD. Também é visto outro CMS diferente do WordPress, o Drupal. *Plug-ins* que já estavam disponíveis no repositório do WordPress foram analisados e por último, são vistas as tecnologias utilizadas no desenvolvimento do *plug-in* deste trabalho.

#### **3.1 WordPress**

O WordPress é uma importante plataforma que foi utilizada como base para a criação do *plug-in*, que é a essência deste projeto. Apresentaremos o porquê de sua importância, junto com a sua história e visão geral.

##### **3.1.1 História**

*Content Manager System (CMS)* é a classe de ferramentas à qual pertence o WordPress. O CMS consiste em um *framework* de *website* previamente estruturado onde é possível administrar, gerar e arquivar conteúdos de maneira dinâmica. No caso do WordPress, a administração da maior parte destes conteúdos ocorre de uma maneira bastante facilitada para os usuários convencionais da *Web*. As atualizações do ambiente podem ser realizadas de forma rápida, um dos motivos principais para o seu uso<sup>30</sup>.

O b2/cafelog é o antecessor do WordPress e foi uma ferramenta criada para a construção de *blogs*. Foi escrito em PHP para ser utilizado junto com o MySQL. O seu desenvolvedor, Michael Valdrighi, agora ajuda no desenvolvimento do WordPress. No ano de 2003 que o WordPress apareceu de fato, como união do trabalho realizado pelo empresário Matt Mullenweg e Mike Little, que procuraram usar o código base do b2/cafelog. Foi constatado em 2009, pelo sistema *Market Share Report* que o WordPress se tornou uma das marcas mais renomadas considerando sistemas de gerenciamento de códigos abertos. Hoje em dia, o sistema é capaz de gerenciar vários *blogs* ao mesmo tempo em apenas uma instalação e ser multiusuário<sup>31</sup>.

##### **3.1.2 Visão Geral**

O WordPress possui duas formas de uso para os usuários escolherem. A

---

30 [https://pt.wikipedia.org/wiki/Sistema\\_de\\_gerenciamento\\_de\\_conte%C3%BAdo](https://pt.wikipedia.org/wiki/Sistema_de_gerenciamento_de_conte%C3%BAdo)

31 <http://pt.wikipedia.org/wiki/Wordpress>

primeira forma consiste na hospedagem em que o WordPress executa na plataforma de código aberto que é suportada pela empresa *Autommatic* e com algumas limitações (WordPress.com). Pagando certo valor, podemos obter acesso a alguns recursos que antes não eram possíveis. Já a outra, consiste em baixar e realizar a instalação pelas mãos do próprio administrador (WordPress.org). Ambas as formas são grátis (porém com algumas limitações, como mencionado anteriormente), entretanto há diferenças entre as duas, a principal e mais importante delas para este trabalho é a de que o WordPress.org permite a instalação de *plug-ins* e o WordPress.com, não.

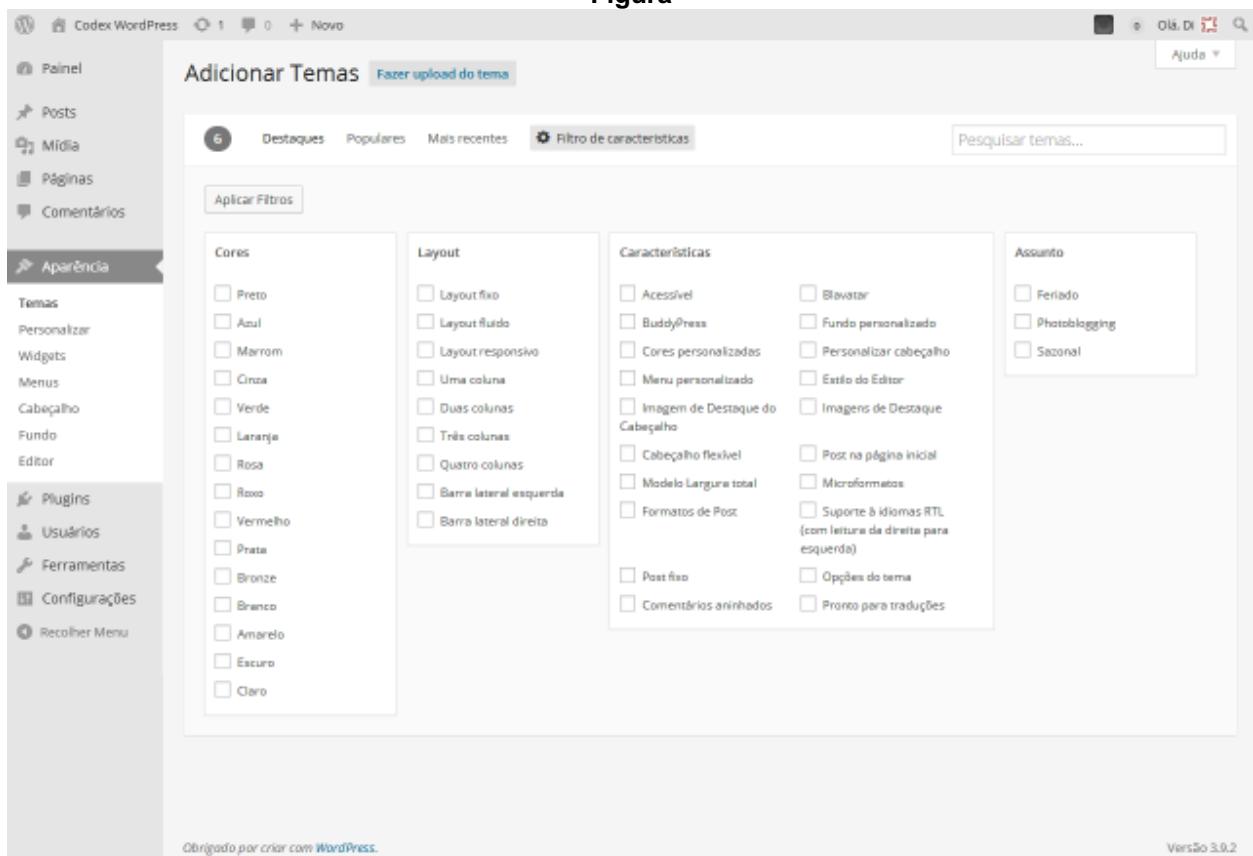
Como o objetivo deste trabalho é criar um *plug-in* que facilite o processo de criar dados, será abordada a forma em que o usuário está encarregado de baixar o WordPress e instalar em seu servidor. O administrador de *blogs* no WordPress tem a possibilidade de criar *posts*, *plug-ins*, além de *themes*.

Um *post* é um artigo postado em um *blog* e em alguns *blogs*, esses artigos são limitados. Assim como acontece na rede social e servidor de *microblogging*, *Twitter*, há um limite de 140 caracteres e cada *post* do *blog* também é chamado de *tweet*. Em outros casos, não há restrições, tais como *Facebook* e o próprio WordPress.

*Themes* são temas, que podem ser aplicados em seus *blogs*, com objetivos como tornar o *blog* mais bonito, mais fácil de ser usado. Os temas basicamente fornecem diversos estilos *front end* e separam o conteúdo da apresentação. Cores, *templates*, estilos para *posts* nos *blogs*, tudo isso faz parte do que um tema pode oferecer.

Na Figura 9 pode-se ver a tela para se adicionar temas novos, verifica-se que é possível customizar cores, *layouts*, assuntos, etc.

**Figura**



### 3.1.3 Estrutura/Organização dos dados

O WordPress tem um esquema de dados próprio, porém neste projeto, somente duas dessas tabelas foram necessárias, as tabelas *post* e *postmeta*, pois são essas que contém as informações dos *posts*.

A tabela *post* guarda diversas informações dos *posts*, tais como autor, data da postagem, título, entre outros, inclusive o tipo do *post*. Já a tabela *post\_meta* possui o id do *post* a qual ele pertence, o *meta\_key* e o *meta\_value*, que são específicos para aquele *post*.

## 3.2 Kettle

O Kettle é um poderoso programa que lida com ETL. Além da integração de dados, pode servir para análise de negócios, *Big Data*, etc. É um software baseado 100% em Java, com uma comunidade bastante ativa e com diversos investidores de peso, o que demonstra a importância e capacidade do programa. O trabalho consistiu em saber que transformações seriam necessárias para se aplicar sobre as

planilhas em formato CSV, trazendo-as para o universo da nossa ontologia e dos dados abertos. Concluindo, teríamos uma triplicação dos dados que poderia apoiar o processo do *plug-in*.

Obtivemos uma planilha proveniente do caso de uso do Proprietários do Brasil. Então, um novo arquivo de transformação do Kettle (\*.ktr) foi criado. Importante mencionar que utilizamos um Kettle estendido com novos *steps*<sup>32</sup>. *Step* nada mais é do que um passo que integra o processo. Depois disso, importamos a planilha “assoc.csv” através do *step* “CSV file input”. Neste *step*, conseguimos obter todos os campos da planilha para trabalhá-los posteriormente. Logo após, criamos diversas conexões com outros *steps* para completar a transformação. Segue abaixo o detalhamento destes *steps*, em ordem:

- “Add constants” (conectado com “CSV file input”). Este *step* consiste em adicionar certos campos a nossa planilha, no caso adicionamos exatamente três: “publicBody”, “formalOrganization” e “Person”. A explicação do porquê destes campos está na descrição da ontologia desenvolvida para este projeto.
- “Select Values” (conectado com “Add constants”). Utilizado apenas para alterar o acento presente no campo “Tipo de Nó” para “Tipo de No”, já que o Kettle não exibe corretamente as acentuações de acordo com a nossa língua (segue o padrão americano). “Select values” seleciona apenas os campos desejados para passar para os *steps* que seguem. Fizemos a seleção de todos, inclusive os que foram criados com “Add constants”.
- “Formula” (conectado com “Select Values”). Esta fórmula teve o propósito de alterar as *strings* “Empresa”, “Estado” e “Pessoa” presentes no campo “Tipo de No” para “EmpresaNome”, “EstadoNome” e “PessoaNome”, meramente para um uso mais adequado na triplificação dos dados da planilha para nossa ontologia.
- “Ajustando nacionalidade” (conectado com “Formula”). Como o nome já diz, este *step* altera as *strings* do campo “Nacionalidade”, de acordo com o país da empresa privada, pública ou pessoa. Realizamos este procedimento para facilitar a visualização e o uso dos dados, procurando formar um certo padrão. Exemplo: “brasileiro” -> “Brasil”.

---

32 <https://github.com/rogersmendonca/ETL4LOD>

- “Switch / Case” (conectado com “Ajustando nacionalidade”): A condição para este caso, é que a *string* seja igual a “EmpresaNome”. Se não for, vai para o step “*Dummy (do nothing)*”. Sendo igual, segue para o step “*Data Property Mapping Empresa*”.
- “Switch / Case 2” (conectado com “Ajustando nacionalidade”): A condição para este caso, é que a *string* seja igual a “PessoaNome”. Se não for, vai para o step “*Dummy (do nothing)*”. Sendo igual, segue para o step “*Data Property Mapping Pessoa*”.
- “Switch / Case 3” (conectado com “Ajustando nacionalidade”): A condição para este caso, é que a *string* seja igual a “EstadoNome”. Se não for, vai para o step “*Dummy (do nothing)*”. Sendo igual, segue para o step “*Data Property Mapping Estado*”.
- “*Dummy (do nothing)*” (conectado com “Switch / Case”, “Switch / Case 2” e “Switch / Case 3”). “Step vazio”. Serve apenas para que uma condição possa cair em um estado onde nada acontece.
- “*Data Property Mapping Empresa*” (conectado com “Switch / Case”). Step proveniente da versão do Kettle com “*Linked Data BR*”. Nele, colocamos o campo que contém a URI do sujeito, tipos RDF da linha de saída da transformação, o predicado (“*DataProperty*”) e o campo com valor do objeto. Aqui inserimos o que foi relativo a Empresa, com tipo RDF da linha “formalOrganization”, predicados “hasName”; “hasCNPJ”; “Nacionalidade”; “hasIncome” e respectivamente, os campos com valor do objeto “EmpresaNome”; “CNPJ”; “Nacionalidade”; “Receita” (de “assoc.csv”).
- “*Data Property Mapping Pessoa*” (conectado com “Switch / Case 2”). Step proveniente da versão do Kettle com “*Linked Data BR*”. Nele, colocamos o campo que contém a URI do sujeito, tipos RDF da linha de saída da transformação, o predicado (“*DataProperty*”) e o campo com valor do objeto. Aqui inserimos o que foi relativo a Pessoa, com tipo RDF da linha “Person”, predicados “hasName”; “hasCPF”; “Nacionalidade”; “hasIncome” e respectivamente, os campos com valor do objeto “PessoaNome”; “CNPJ”; “Nacionalidade”; “Receita” (de “assoc.csv”).
- “*Data Property Mapping Estado*” (conectado com “Switch / Case 3”). Step

proveniente da versão do Kettle com "Linked Data BR". Nele, colocamos o campo que contém a URI do sujeito, tipos RDF da linha de saída da transformação, o predicado ("DataProperty") e o campo com valor do objeto. Aqui inserimos o que foi relativo a Estado, com tipo RDF da linha "publicBody", predicados "hasName"; "hasCPF"; "Nacionalidade"; "hasIncome" e respectivamente, os campos com valor do objeto "EstadoNome"; "CNPJ"; "Nacionalidade"; "Receita" (de "assoc.csv").

- "scape" (conectado com "Data Property Mapping Empresa", "Data Property Mapping Pessoa" e "Data Property Mapping Pessoa"). Mesma coisa do step "Add Constants". Adiciona apenas um campo chamado "scape", que mais para a frente é usado para formatar a saída das triplas.
- "String operations" (conectado com "scape"). Remove alguns caracteres especiais dos campos de sujeito e objeto oriundos das steps de "Data Property Mapping".
- "Replace in string" (conectado com "String operations"). Substitui acentos, sinais e outras coisas que possam atrapalhar na saída das triplas.
- "Formula 2" (conectado com "Replace in string"). A formula aqui presente serve para formatar corretamente a saída das triplas e permitir que no final da transformação, possamos colocar um step para um endpoint. Tudo é passado para um novo campo, chamado "tripla".
- "Select tripla" (conectado com "Formula 2"). Selecionamos então, o campo "tripla" (similar ao step "Select values").
- "Sparql Update Insert" (conectado com "Select tripla"). Final da transformação. Exportamos os dados formatados para um "Sparql Update Endpoint", inserindo a URL do mesmo, nome de usuário e senha.

A saída que se obteve de "assoc.csv" pode ser vista na Figura 10:

**Figura**

```
1 tripla
2 <http://trabalho.orgcon.ufrj.br/resource/1> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://trabalho.orgcon.ufrj.br/proprietarios/formalOrganization> .
3 <http://trabalho.orgcon.ufrj.br/resource/1> <http://trabalho.orgcon.ufrj.br/proprietarios/hasName> "BOMBRIL HOLDINGSA" .
4 <http://trabalho.orgcon.ufrj.br/resource/1> <http://trabalho.orgcon.ufrj.br/proprietarios/hasCNPJ> "257846000101" .
5 <http://trabalho.orgcon.ufrj.br/resource/1> <http://trabalho.orgcon.ufrj.br/proprietarios/Nacionalidade> "Brasil" .
6 <http://trabalho.orgcon.ufrj.br/resource/1> <http://trabalho.orgcon.ufrj.br/proprietarios/hasIncome> "8" .
7 <http://trabalho.orgcon.ufrj.br/resource/2> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://trabalho.orgcon.ufrj.br/proprietarios/formalOrganization> .
8 <http://trabalho.orgcon.ufrj.br/resource/2> <http://trabalho.orgcon.ufrj.br/proprietarios/hasName> "Isolux Infraestrutura S.A." .
9 <http://trabalho.orgcon.ufrj.br/resource/2> <http://trabalho.orgcon.ufrj.br/proprietarios/hasCNPJ> "13407617000178" .
10 <http://trabalho.orgcon.ufrj.br/resource/2> <http://trabalho.orgcon.ufrj.br/proprietarios/Nacionalidade> "Brasil" .
11 <http://trabalho.orgcon.ufrj.br/resource/2> <http://trabalho.orgcon.ufrj.br/proprietarios/hasIncome> "2" .
12 <http://trabalho.orgcon.ufrj.br/resource/3> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://trabalho.orgcon.ufrj.br/proprietarios/formalOrganization> .
13 <http://trabalho.orgcon.ufrj.br/resource/3> <http://trabalho.orgcon.ufrj.br/proprietarios/hasName> "ANDRADE GUTIERREZ TELECOMUNICACAO E PARTICIPACOES S.A." .
14 <http://trabalho.orgcon.ufrj.br/resource/3> <http://trabalho.orgcon.ufrj.br/proprietarios/hasCNPJ> "11616840000108" .
15 <http://trabalho.orgcon.ufrj.br/resource/3> <http://trabalho.orgcon.ufrj.br/proprietarios/Nacionalidade> "Brasil" .
16 <http://trabalho.orgcon.ufrj.br/resource/3> <http://trabalho.orgcon.ufrj.br/proprietarios/hasIncome> "4924" .
17 <http://trabalho.orgcon.ufrj.br/resource/4> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://trabalho.orgcon.ufrj.br/proprietarios/formalOrganization> .
18 <http://trabalho.orgcon.ufrj.br/resource/4> <http://trabalho.orgcon.ufrj.br/proprietarios/hasName> "AESP L PASA" .
19 <http://trabalho.orgcon.ufrj.br/resource/4> <http://trabalho.orgcon.ufrj.br/proprietarios/hasCNPJ> "1917705000130" .
20 <http://trabalho.orgcon.ufrj.br/resource/4> <http://trabalho.orgcon.ufrj.br/proprietarios/Nacionalidade> "Brasil" .
21 <http://trabalho.orgcon.ufrj.br/resource/4> <http://trabalho.orgcon.ufrj.br/proprietarios/hasIncome> "9836" .
22 <http://trabalho.orgcon.ufrj.br/resource/5> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://trabalho.orgcon.ufrj.br/proprietarios/formalOrganization> .
23 <http://trabalho.orgcon.ufrj.br/resource/5> <http://trabalho.orgcon.ufrj.br/proprietarios/hasName> "ALL-AMERICA LATINAL LOGISTICAS S.A." .
24 <http://trabalho.orgcon.ufrj.br/resource/5> <http://trabalho.orgcon.ufrj.br/proprietarios/hasCNPJ> "2387241000160" .
25 <http://trabalho.orgcon.ufrj.br/resource/5> <http://trabalho.orgcon.ufrj.br/proprietarios/Nacionalidade> "Brasil" .
26 <http://trabalho.orgcon.ufrj.br/resource/5> <http://trabalho.orgcon.ufrj.br/proprietarios/hasIncome> "3173" .
27 <http://trabalho.orgcon.ufrj.br/resource/6> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://trabalho.orgcon.ufrj.br/proprietarios/formalOrganization> .
28 <http://trabalho.orgcon.ufrj.br/resource/6> <http://trabalho.orgcon.ufrj.br/proprietarios/hasName> "ALUPAR INVESTIMENTOS S/A" .
29 <http://trabalho.orgcon.ufrj.br/resource/6> <http://trabalho.orgcon.ufrj.br/proprietarios/hasCNPJ> "8364948000138" .
30 <http://trabalho.orgcon.ufrj.br/resource/6> <http://trabalho.orgcon.ufrj.br/proprietarios/Nacionalidade> "Brasil" .
31 <http://trabalho.orgcon.ufrj.br/resource/6> <http://trabalho.orgcon.ufrj.br/proprietarios/hasIncome> "1165" .
32 <http://trabalho.orgcon.ufrj.br/resource/7> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://trabalho.orgcon.ufrj.br/proprietarios/formalOrganization> .
33 <http://trabalho.orgcon.ufrj.br/resource/7> <http://trabalho.orgcon.ufrj.br/proprietarios/hasName> "AMPLA INVESTIMENTOS SERVICOS S/A" .
34 <http://trabalho.orgcon.ufrj.br/resource/7> <http://trabalho.orgcon.ufrj.br/proprietarios/hasCNPJ> "7089905000156" .
35 <http://trabalho.orgcon.ufrj.br/resource/7> <http://trabalho.orgcon.ufrj.br/proprietarios/Nacionalidade> "Brasil" .
36 <http://trabalho.orgcon.ufrj.br/resource/7> <http://trabalho.orgcon.ufrj.br/proprietarios/hasIncome> "2627" .
37 <http://trabalho.orgcon.ufrj.br/resource/8> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://trabalho.orgcon.ufrj.br/proprietarios/formalOrganization> .
38 <http://trabalho.orgcon.ufrj.br/resource/8> <http://trabalho.orgcon.ufrj.br/proprietarios/hasName> "ANDRADE GUTIERREZ CONCESSOES S.A." .
39 <http://trabalho.orgcon.ufrj.br/resource/8> <http://trabalho.orgcon.ufrj.br/proprietarios/hasCNPJ> "1161684000108" .
```

10 - Linhas de saída após a transformação da ferramenta Kettle sobre uma planilha do banco de

Como é possível notar, não é um processo trivial que qualquer usuário

conseguiria realizar. O Kettle foi o primeiro contato dos autores deste trabalho com ETL, sendo essencial para a ideia do *plug-in* e suas nuances. A planilha utilizada foi parte do caso de uso do Proprietários do Brasil e os termos bem próximos da ontologia elaborada. Porém, a proposta do *plug-in* diverge do Kettle no sentido de facilitar o uso dos administradores de *blogs* do WordPress e alcançar um número elevado de usuários.

### 3.3 Drupal

A plataforma Drupal consiste em uma ferramenta de código aberto para gerenciamento de conteúdo para diversos sites na Internet. A sua comunidade é bem grande, ativa e motivada. O Drupal é utilizado por mais de um milhão de sites em todo o mundo, incluindo *blogs* pessoais, para políticos e até mesmo, sites governamentais e corporativos. Além disso, pela sua similaridade com o WordPress, tivemos um auxílio maior em nossa procura pelas funcionalidades para o nosso *plug-in*.

Diferentemente do WordPress, ao invés de *plug-ins*, temos módulos no Drupal. Eles são basicamente desenvolvidos através da linguagem PHP também. Como esperado, eles servem para customizar e estender as funcionalidades do Drupal. A instalação dos mesmos não tem complicações, mas grande parte dos módulos

dependem de outros módulos, logo é preciso observar este detalhe para que eles possam funcionar. Acima dos módulos, existem os chamados “projetos”, que nada mais são do que um conjunto de módulos.

Consultamos sobre LOD no Drupal com a versão 7-3.1. Com o MySQL, PHP e Apache 2 rodando localmente, testes foram realizados. Antes da versão 7 do Drupal, o principal projeto para LOD era o *Linked Data*, que permitia a criação de consultas *Linked Data* e podia apresentar os resultados de diversas maneiras. Isso significava que era possível incluir de forma dinâmica conteúdo de diversas bases com DBpedia, Data.gov, GeoNames, etc<sup>33</sup>.

Já na versão 7, outro projeto passou a ser utilizado: SPARQL Views<sup>34</sup>. Este projeto faz uso de outro, chamado *Views* para exibir os dados formatados da maneira que for desejada. Basicamente, o SPARQL *Views* além do que foi dito anteriormente, utiliza a estrutura do *Views* para acessar localmente e remotamente os SPARQL *endpoints*, onde se realizam as consultas.

Para as consultas serem realizadas, é preciso habilitar “*Clean URLs*”. Isto é feito com pequenas alterações no Apache e habilitando “mod\_rewrite”, que nos permite modificar solicitações de URL de entrada, de maneira dinâmica, com base em regras de expressão regular, ou seja, podemos mapear estruturas de URLs quaisquer do jeito que quisermos. No caso do Drupal, teríamos uma URL exemplo como esta: “<http://www.exemplo.com/?q=node/42>”. Habilitando “*Clean URLs*”, ficaria da seguinte maneira: “<http://www.example.com/node/42>”. Com esta modificação, podemos fazer a consulta SPARQL corretamente pelo Drupal.

Mais uma vez, outro projeto é necessário, o SPARQL que precisa do SPARQL *Views* para realizar as consultas (através do módulo SPARQL API), como exibido na Figura 11. Como o SPARQL lida com o formato RDF, necessitamos do projeto RDF Extensions que foi desenvolvido como uma nova versão para o Drupal 7 do apenas RDF que foi feito para versões anteriores do Drupal. Este projeto contém uma interface de usuário para mapeamento RDF, fornece serialização em RDF/ XML, *NTriples*, entre outros.

Sem o módulo Entity API, o RDF *Extensions* não funciona. Simplificando, o *Entity API* fornece um jeito unificado de lidar com as entidades e suas propriedades, o RDF

---

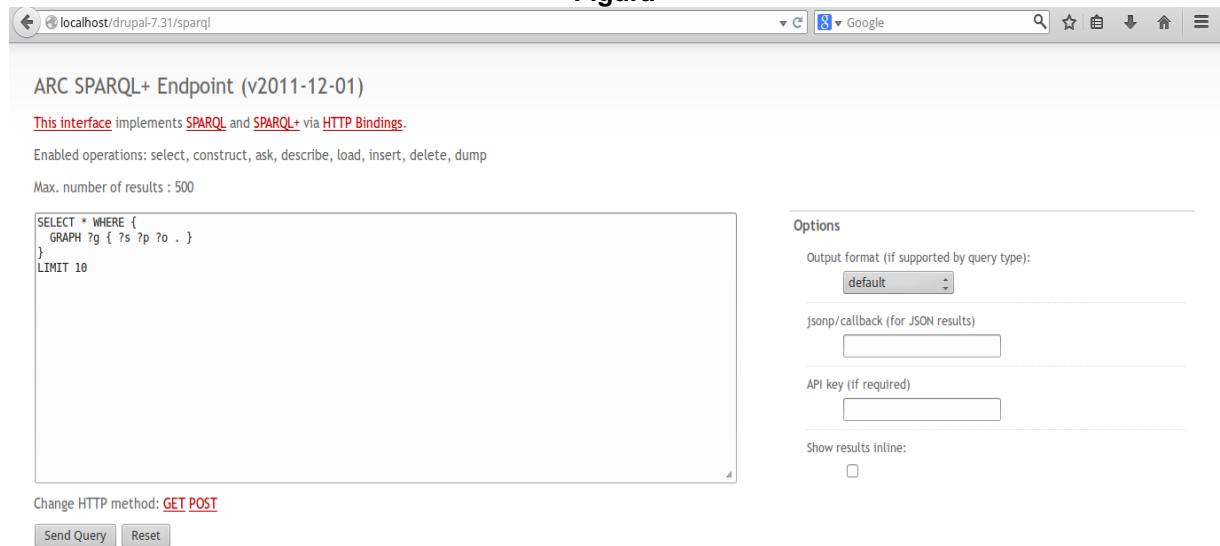
33 [https://www.drupal.org/project/linked\\_data](https://www.drupal.org/project/linked_data)

34 [https://www.drupal.org/project/sparql\\_views](https://www.drupal.org/project/sparql_views)

*Extensions*. Outro módulo importante tratando de RDF e SPARQL *endpoints*, é o *ARC2 Store*, que consiste em um banco de dados RDF para consultas SPARQL. De certa maneira, podemos dizer que cada *ARC2 Store* pode ser configurado para exibir um SPARQL *endpoint* em uma URL de nossa escolha.

Não se pode esquecer o *Ctools*, que é uma suíte básica (um projeto também) com um conjunto de APIs e ferramentas para os desenvolvedores e serve como base para diversos módulos que já foram descritos, como o *ARC2 Store*.

**Figura**



Podemos dizer então, que a nossa pesquisa por questões que envolviam LOD no Drupal para um aproveitamento posterior no WordPress, além da visão geral que obtivemos, acabou nos retornando alguns detalhes que deixaram mais claro o caminho que seguiríamos.

Definimos no projeto do *plug-in* no WordPress que ele precisava de algum tipo de consulta, o que procuramos no Drupal. Porém, o tipo de consulta encontrada foi em SPARQL mas acabamos por utilizar uma aplicação *RESTful*, pois assim o usuário teria mais facilidade (o que está de acordo com o princípio deste projeto, que inclui a elaboração de uma interface menos complexa) de acessar os dados.

Em relação ao *ARC2 Store*, não foi observada a necessidade de implementar um banco de triplas para o projeto do *plug-in*. O *plug-in* lida essencialmente com

transformações e não armazenamento, as saídas são geradas no mesmo momento em que o usuário selecionar para salvar as opções inseridas. Posteriormente, pode-se conferir as saídas via URL para cada formato desejado. Por último, o módulo RDF *Indexer* (que é na verdade um módulo posterior ao módulo SPARQL *Endpoint*), e que serve para a indexação de dados RDF para as entidades de um site qualquer, nos indicou outra funcionalidade. Na verdade, foi uma referência para que utilizássemos o LOV em nosso *plug-in* (o que de certa forma, já era esperado).

### **3.4 Plug-ins disponíveis**

Procurou-se por *plug-ins* no repositório do WordPress com funcionalidades similares em termos de LD. Foram estabelecidas comparações com nosso *plug-in* e então, descrevemos o que diferencia o *plug-in* deste trabalho dos outros que já estavam disponíveis:

a) *Ancient World Linked Data for WordPress*

Este *plug-in* permite ao usuário utilizar URI's relevantes ao “Mundo Antigo”, adicionando elementos visuais e funcionalidades ao WordPress do usuário. É bastante diferente do *plug-in* proposto, sendo que na verdade, possui somente o fato de usar URI's, no caso do *Ancient World Linked Data for WordPress* é, além do propósito do conceito envolvendo URI, mostrar imagens, outros elementos de interface, etc. Não possui o processo de triplificação.

b) *wp-linked-data*

Fornece ao usuário a opção de triplificar os dados dos *posts* de seu *blog* do WordPress. Entre os *plug-ins* semelhantes ao *plug-in* proposto é o mais parecido, entretanto não oferece a funcionalidade de estabelecer as correspondências entre colunas do banco e predicados das triplas. Não permite a escolha da URI do tipo (no *wp-linked-data* vem uma URI mutável somente em partes), e também não oferece os dados em formato RDF ou JSON-LD. O tipo triplificado é somente *post* e nenhum outro.

c) *Link Data From Another DB*

Este *plug-in* apenas conecta os dados de dois bancos de dados MySQL (banco utilizado pelo WordPress). Além disso, não utiliza URI's e triplas. Nada do conceito de LD é abordado por este *plug-in*.

d) *WP Data Cube*

Realiza a extração de dados estatísticos de tabelas provenientes do formato CSV e executa a transformação para o formato de triplas. Apresenta também uma interface para consultas SPARQL.

Prefixos podem ser armazenados no WP *Data Cube* e esta característica foi de grande utilidade para nosso caso. Entretanto, nosso *plug-in* não apresenta a mesma proposta de lidar com dados estatísticos e nem consultas SPARQL. No final, constatou-se que o WP *Data Cube* não possui tantas similaridades.

### **3.5 Tecnologias utilizadas**

Nesta subseção ocorre a descrição sobre as tecnologias que foram usadas no desenvolvimento do *plug-in* e a importância de cada uma delas:

a) PHP

*HiperText Preprocessor* (PHP) é a linguagem que os *plug-ins* para WordPress utilizam. É uma linguagem *server-side*, própria para criar páginas *Web*. Segundo o site IEEE, no ano de 2014 é a sétima linguagem mais usada do mundo. [1]

Até a versão PHP4, era uma linguagem exclusivamente procedural, entretanto a partir dessa versão passou a ter a possibilidade de ser orientada à objetos, além de procedural.

b) JQuery

JQuery é uma biblioteca Javascript, também segundo [1], é a sexta linguagem mais utilizada do mundo. Javascript é uma linguagem *client-side*, e a biblioteca JQuery facilita bastante o desenvolvimento com Javascript, pois já tem muitas funções úteis prontas para serem usadas.

Javascript se comunica com o servidor via conexões AJAX, o *plug-in* faz conexões desse tipo para que ele envie os dados triplificados à aplicação RESTful.

Importante ressaltar que toda a parte visual do *plug-in* foi feita utilizando esta linguagem.

## **4 A elaboração do *plug-in***

Esta seção apresenta a proposta de solução para o problema e antecede os exemplos de aplicação demonstrados neste trabalho. Aqui veremos a especificação acerca do *plug-in* e o processo criado, com detalhes sobre seu uso, implementação e saídas retornadas.

### **4.1 Especificação**

No WordPress temos *plug-ins*. *Plug-ins* são como “aplicativos” que o usuário pode customizar seu WordPress, adicionando funcionalidades. Existem uma grande quantidade de *plug-ins* que o usuário pode simplesmente “importar” para seu WordPress, para isso basta que ele entre como administrador e vá na parte de *plug-ins*, escolha o *plug-in* que deseja adicionar em seu WordPress e ativá-lo.

Outra opção possível é o *download* dos *plug-ins* pelo repositório do WordPress, diretamente pelo *browser*<sup>35</sup>. É possível criar um registro de usuário e fazer login, podendo assim, realizar *upload* de *plug-ins* também.

Existe também a possibilidade de o usuário criar seu próprio *plug-in*. Para criá-lo, o mesmo precisa ser programado na linguagem PHP, ter todos os arquivos necessários para o *plug-in* compactados em um \*.zip, ir na página de *plug-ins*, fazer *upload* do arquivo e ativá-lo.

Para este projeto, foi preciso elaborar um escopo com as funcionalidades necessárias. A proposta consistiu no desenvolvimento de um *plug-in* para WordPress que atendesse a demanda dos administradores dos blogs no Wordpress. Este *plug-in* deveria facilitar o processo de triplificação dos dados, podendo fazer o uso de ontologias existentes e acrescentando novas também. Fora isto, o usuário poderia escolher ao término do processo, o formato de saída que desejasse dentre os mais conhecidos envolvendo LOD. Antes de qualquer tipo de ação, lidamos primeiro com a ferramenta Kettle que trabalha com ETL4LOD. Depois, procurou-se fazer um levantamento de soluções já existentes em outro CMS na Web, o Drupal, para traçarmos melhor o que desenvolveríamos e as funcionalidades similares que poderíamos utilizar.

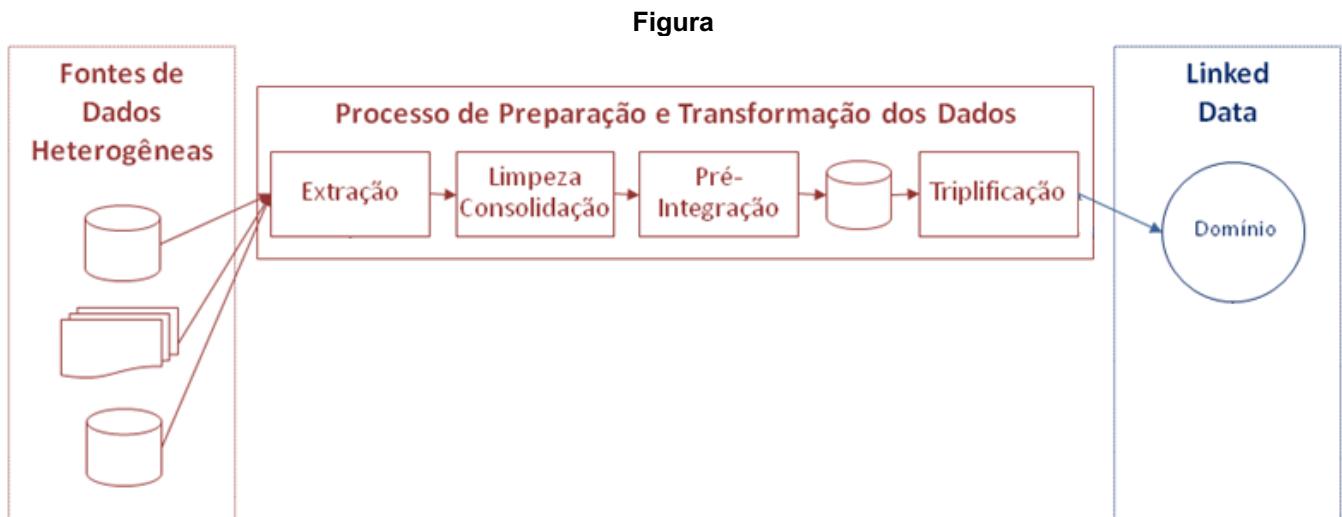
### **4.2 Processo**

Por meio de todos os estudos e exemplos vistos que antecederam o

---

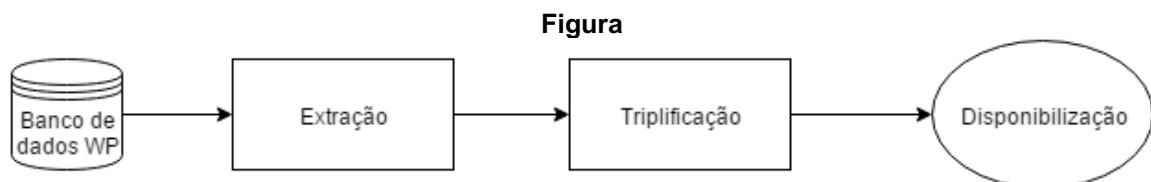
<sup>35</sup> <https://wordpress.org/plugins/>

desenvolvimento do *plug-in*, criou-se um diagrama simples para representar o nosso processo. Este diagrama foi construído principalmente com base no ETL4LOD, implementado na ferramenta Kettle. Podemos ver na Figura 12, uma visão geral de um processo de ETL4LOD:



**Fonte:** Retirado do site do grupo GRECO da UFRJ. Modificado pelos autores<sup>36</sup>

O nosso exemplo feito através da ferramenta Kettle seguiu o processo de ETL4LOD da Figura 12. O *plug-in* apresenta o seguinte esquema, conforme a Figura 13:



A fontes de dado do *plug-in* é o próprio banco de dados atrelado com o WordPress e no final, ocorre apenas a disponibilização dos dados triplificados, sem nenhum tipo de armazenamento.

#### 4.3 Utilização

O *plug-in* auxilia o usuário para disponibilizar os dados de seus *posts* do WordPress de uma forma interligada com outras bases, tornando assim os dados

<sup>36</sup> <http://greco.ppgi.ufrj.br/lodbr/index.php/principal/etl4linkedprov/>

mais acessíveis.

Para ter acesso ao *plug-in* e instalá-lo, o usuário deve entrar na parte de administração do WordPress, ir em “*Plugins -> Add new*” procurar por “*data triplify*” e instalar o *plug-in*.

Depois de instalado o usuário vai em “*Settings -> Data-Triplify*” e tem acesso à página inicial do sistema, que possui três módulos:

- a) URL base, escolha de tipo de *post* e correspondências (amarelo)

**Figura**

The screenshot shows a user interface for configuring a URL base. At the top, there's a text input field labeled "Enter the triplified data's URL basis:" containing the value "http://localhost/wordpress/". Below this, a section titled "Already triplified Post-types:" lists several items in blue boxes: "abc", "page", "portfolio", and "post". Underneath these, another section titled "Search a Post-type and manually triplify it:" contains a list of various post types with checkboxes next to them. The listed types include: attachment, nav\_menu\_item, page, portfolio, post, revision, slide, whoowns\_bndes\_proj, whoowns\_ed\_candidate, whoowns\_ed\_donation, and whoowns\_owner. A blue "Browse" button is located at the bottom left of this list.

Nesta parte, como demonstra a Figura 14, o usuário seleciona a url base em que ele acessará para obter os dados no formato especificado. Nesse exemplo acima, ele entraria em uma url que comece com <http://localhost/Wordpress/abcdef> para acessar aos seus dados.

Este valor é armazenado na tabela *wp\_options* com os valores das colunas *option\_name* e *option\_value* iguais a “*triplify\_url\_base\_dados*” e valor digitado neste campo, neste caso “*abcdef*”, respectivamente.

Há também a facilidade de o usuário selecionar apenas no tipo e ir direto para seus dados já triplificados no formato JSON-LD, no caso da Figura 14, o usuário já triplificou os tipos *abc*, *page*, portfólio, *post*, e apenas clicando em um desses *links*, ele é redirecionado para [http://localhost/Wordpress/abcdef/tipo\\_clicado](http://localhost/Wordpress/abcdef/tipo_clicado). Importante de ser mencionado, caso o tipo não exista no banco de dados do WordPress, o usuário é redirecionado e lá verifica uma mensagem como da Figura 15, informando

que o tipo não existe.



Caso o tipo exista, temos os dados em si, como exemplifica a Figura 16, mostrando os dados do tipo *post*.



Para obter um resultado semelhante ao encontrado na Figura 16, o usuário configura quais colunas do WordPress devem ser representadas nos dados triplificados e qual verbo ele se tornará, para isso ele vai para a última parte do módulo de URL base e escolha de tipo de *post* e correspondências, que é onde ele define as referidas correspondências, conforme é mostrado pela Figura 17:

**Figura**

The screenshot shows a user interface for mapping field names to URIs. At the top, there is a text input field labeled "Type the URI basis of the posts:" containing the value "URI-teste-diferente". Below this, a note says "Enter the correspondences and check the checkbox if the shown result of this column is an URI:". A detailed note below states: "Check the option 'not important' if it's correspondence isn't important, so it should not be shown in the final result. Select 'not important' or leave the correspondences with default values has the same results". A list of 9 mappings is provided, each consisting of a checkbox followed by a field name and a URI template, and a "not important" checkbox:

- 1  ID =>   not important
- 2-  post\_author =>   not important
- 3-  post\_date =>   not important
- 4-  post\_date\_gmt =>   not important
- 5-  post\_content =>   not important
- 6-  post\_title =>   not important
- 7-  post\_excerpt =>   not important
- 8-  post\_status =>   not important
- 9-  comment\_status =>   not important

Primeiramente o usuário define a URI base para que o *post* seja identificado unicamente nos dados triplificados futuramente. Isto se dará pela URI base que ele escolher neste campo adicionado de um “/” + id do *post*, por exemplo, no caso da Figura 17. Vamos supor que sejam 5 *posts* com id's 1, 2, 3, 4 e 5, ao dispor os dados no final. Feito isto, cada *post* teria seu identificador único, sua URI como *URI-teste-diferente/seu\_id*, onde *seu\_id* está entre 1 e 5.

Esse processo de salvamento se dá da seguinte maneira no banco de dados do WordPress: o valor dessa URI base é salvo na tabela *wp\_options*, no campo *option\_value* enquanto o tipo ao qual corresponde àquela URI base é armazenado na coluna *option\_name*. Como este valor pode assumir um nome relativamente comum, como *post* ou *page* por exemplo, ele não é simplesmente armazenado com seu valor puro e simplesmente, ele é armazenado da seguinte forma: é concatenação da string “#triplifier\_uri\_base\_dados#” com o tipo em si (*post*, *page*, entre outros).

Pode-se perceber que temos tuplas numeradas na página. São tantas tuplas quanto colunas existentes referentes ao tipo em questão. O primeiro *checkbox*, mais à esquerda, deve ser marcado se, e somente se, o predicado a ser relacionado com a coluna deve ser uma URI, isso se dá pelo fato de a biblioteca JSON-LD mostrar diferentemente simples *strings* de URI's.

Imediatamente ao lado direito, verifica-se o nome da coluna em si, como se encontra no WordPress. Logo em seguida, está o *input text* para o usuário definir o

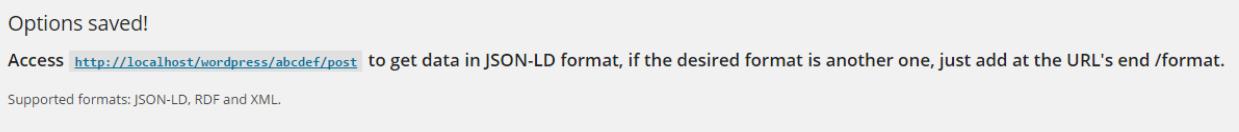
verbo a ser mostrado. Este campo possui um *autocomplete* com predicados mais conhecidos e mais difundidos, como foaf, dc, rdf e rdfs.

E finalmente, como há no mínimo 23 colunas (colunas da tabela `wp_posts`), pode ser que para o usuário a maior parte dessas colunas não seja interessante, então para que a tela fique mais simples, legível e menos poluída, o usuário pode marcar “*not important*” fazendo com que aquela coluna não apareça na tela e sinalizando que não quer que essa coluna esteja na disposição final dos dados. Caso o usuário não marque esta opção, porém tenha deixado “*correspondence*” ou vazio algum *input text*, o *plug-in* também entende que aquela coluna é para ser desconsiderada.

Feitas todas as correspondências, basta ao usuário selecionar “*Save Options*”, no fim da página para que se depare com uma tela com a mensagem como a da Figura 18 e os dados estejam prontos. Ela contém um *link* direto para os dados no formato JSON-LD, e informa ao usuário em quais formatos, e como acessá-los, estão disponíveis os dados, até a versão atual são XML e RDF, além do JSON-LD.

Esses dados são salvos na tabela `wp_triplify_configurations`, que possui 5 colunas: id, tipo, coluna, valor\_correspondente e uri. “Id” é simplesmente a chave primária da tabela. “Tipo” refere-se ao tipo ao qual aquela “substituição” de coluna pelo valor\_correspondente deve ser feita. “Coluna” e “valor” são autoexplicativos. Uri é um booleano, indicando se o predicado que será exposto por este verbo que está em valor\_correspondencia será uma URI ou não, sendo “true” em caso afirmativo e “false”, caso contrário. A uri\_value é uma *string* onde, caso o valor de uri seja “true” o usuário pode preencher esta coluna com alguma URI, caso seja necessário para o objeto das triplas do resultado final, se esta coluna for deixada em branco, nada será acrescentado ao objeto de cada tripla.

### Figura



b) *Upload* de arquivo CSV (verde):

### Figura

Upload a CSV file with the needed configurations to triplify:

The file must be like:

Must be semicolon separated (;)  
First line must contain all the types that will be triplified  
Second line must contain the URI basis respectively  
All the following lines must have, in the following order: The column, it's intended correspondence and if the shown value is an URI or not

Nenhum arquivo selecionado

O módulo de *upload* de arquivo CSV serve para fazer exatamente a mesma tarefa que o módulo de URL base e escolha de tipo de *post* e correspondências, entretanto poupa trabalho do usuário, pois ele pode criar um arquivo com todas as informações relevantes e já registrar todas as configurações rapidamente, e mais do que isso, permite que configure vários tipos de uma vez só.

O arquivo do formato CSV deve ter as seguintes especificações, assim como visto na Figura 19:

1. O delimitador deve ser ponto e vírgula (;).
2. A primeira linha deve conter todos os tipos a serem configurados.
3. A segunda linha deve conter todas as URI's base, de cada tipo da primeira linha, respectivamente.
4. As seguintes linhas devem conter respectivamente a coluna, a correspondência a ser mostrada no resultado final, podem ser tantas linhas quanto número de colunas dos tipos sendo configurados.
5. A primeira e a segunda linhas devem ter o mesmo número de colunas, caso contrário dispara um erro como mostrado na Figura 20.
6. Da segunda linha em diante, se todas as linhas não tiverem três ou quatro colunas cada uma, também é lançado um erro, exemplificado pela Figura 21.

### Figura

Upload failed! Line 1, with post-types is empty, or maybe it has two semicolons in a row.

### Figura

Upload failed! Line 3 has incorrect format.

### c) Cadastro de prefixos e URI's (rosa)

**Figura**

Prefixes already in records:

dc foaf grddl owl rdf rdfs teste xsd

Add new prefixes. If the specified prefix already is on database, it's URI only will be updated. Type the prefix without colon ':'

Prefix	URI	Save
--------	-----	------

Esta parte do *plug-in* serve para, quando os dados forem disponibilizados, venham com prefixos corretos. O usuário cadastra os prefixos e suas URI's correspondentes, para que, no resultado final os prefixos estejam corretos. O *plug-in* já vem com alguns pré-cadastrados, que são: dc, foaf, owl, rdf, rdfs e xsd.

O WP *Data Cube*, já comentado neste trabalho, possui uma funcionalidade semelhante, onde são armazenados prefixos, então, para comodidade do usuário, o *plug-in* procura pela tabela onde estão armazenados esses prefixos oriundos do WP *Data Cube*, e caso existam, os copia para a tabela *wp\_triplify\_prefixes*, que é onde ficam salvos os prefixos que o *plug-in* utiliza.

A estrutura da tabela *wp\_triplify\_prefixes* é muito simples, possui apenas dois campos, prefixo e URI, onde prefixo é a chave primária, já que um mesmo prefixo não pode ter duas URI's diferentes.

O usuário deve entrar com o prefixo sem os dois pontos (:), por exemplo, não deve cadastrar "foaf:person" nem "foaf:", e sim apenas "foaf" (Figura 22). Caso o prefixo que ele esteja tentando cadastrar já exista no banco, apenas sua URI será alterada.

#### 4.4 Implementação

O *plug-in* é dividido em diversos arquivos e cada um com sua respectiva relevância e funcionalidade dentro do sistema:

##### a) Render.php

Este arquivo é o responsável por quase todas as partes de visualização do sistema: Dependendo do parâmetro *POST* recebido ele mostra a respectiva parte.

Caso o parâmetro recebido esteja em "postType" ele então faz uma busca pelo tipo recebido e mostra na tela. Caso o parâmetro recebido seja "triplify-csv-file" o sistema sabe que o usuário está querendo subir as configurações de

correspondências entre colunas do banco e predicados das triplas do resultado final de uma forma automatizada, através de um arquivo, então ele instancia uma nova classe, que possui um arquivo próprio, a "ReadCSVFile" que será explicada mais a frente. E por último, mas não menos importante, o parâmetro "salvar\_prefixos", que simplesmente pega os itens a serem salvos e faz uma busca no banco, caso o valor já exista, ele simplesmente atualiza, caso contrário ele o salva. Este arquivo também é responsável pelas funções java nas partes citadas acima, tais como *autocomplete*, etc.

b) Triplify.php

Este arquivo é o arquivo "inicial" do sistema: Ele adiciona a opção de acessar o plug-in em si no menu do administrador, é responsável também pelo recebimento e ação de funções ajax, além de criar as tabelas no banco que o sistema utiliza, a "wp\_triplify\_configurations" e a "wp\_triplify\_prefixes".

c) Type\_text.php

Cada arquivo nesta subseção é responsável por receber o formato inserido na URL para a saída dos dados e mostrá-los na tela de acordo com seu tipo.

d) TextJson.php

Arquivo que mostra os dados no formato JSON.

e) TextRDF.php

Arquivo que mostra os dados no formato RDF.

f) TextXML.php

Arquivo que mostra os dados no formato XML.

g) ReadCSVFile.php

Arquivo responsável pela validação e salvamento dos dados inseridos no sistema via arquivo CSV.

h) PrefixColumnUri.php

Arquivo simplesmente com uma classe que "agrupe" propriedades de um mesmo registro a ser inserido ou recuperado da tabela wp\_triplify\_configurations para simplesmente facilitar e organizar melhor o desenvolvimento.

i) Functions.php

Arquivo com simples funções de manipulação de cadeias de caracteres, também conhecidos por *strings*, tais como retirada de caracteres especiais, acentos,

etc.

j) Uninstall.php

Arquivo com procedimentos a serem feitos ao se desinstalar o *plug-in*, tais como deletar tabelas criadas pelo *plug-in*, etc.

#### 4.5 Saídas

Importante ressaltar que todas as saídas foram testadas, com o intuito de verificar se seu formato estava correto e todos passaram. JSON-LD foi usado o *playground* do json-ld.org (<http://json-ld.org/playground/index.html>), RDF foi usado RDF validador do w3 (<http://www.w3.org/RDF/Validator/>) e XML foi usado o validador do w3Schools ([http://www.w3schools.com/xml/xml\\_validator.asp](http://www.w3schools.com/xml/xml_validator.asp)).

Vale lembrar também que, por vezes, as saídas possuíam o caractere “&”, como por exemplo a empresa Standard & Poor's, o que é um problema visto que esse caractere possui significado em alguns desses formatos, portanto para validação do formato nos validadores mencionados, este caracter em especial foi substituído por um simples “E”. Os resultados são retornados ao usuário via um *browser* qualquer, sendo possível visualizar a URL na barra de endereço.

a) JSON-LD

Segue um exemplo de uma saída neste formato na Figura 23:

## Figura

localhost/wordpress/abcdef/post/

("@context": {"rdf": {"@id": "http://www.w3.org/1999/02/22-rdf-syntax-ns#"}, "@type": "@id"}, "dc": {"@id": "http://purl.org/dc/elements/1.1/"}, "@graph": [{"@id": "URI-teste-diferente/1", "dc:abstract": "1", "dc:accessRights": "8", "dc:tableOfContents": "Welcome to WordPress. This is your first post. Edit or delete it, then start blogging!"}, {"@id": "URI-teste-diferente/29042", "dc:abstract": "29042", "dc:accessRights": "8", "dc:tableOfContents": "**Terminada a Copa do Mundo é hora de voltar à realidade***Publicado em <a href='http://odia.ig.com.br/noticia/opiniao/2014-07-11/joao-roberto-lopes-pinto-os-donos-do-rio.html' target='\_blank'>O Dia, 11/07/2014</a> Rio - Terminada a Copa do Mundo é hora de voltar à realidade. Pesquisa 'Quem são os proprietários do Brasil?', no Instituto Mais Democracia e da Cooperativa EITA, aponta fortes indícios de formação de cartel nas obras da Copa e Olimpíadas no Rio, envolvendo Odebrecht, Andrade Gutierrez, OAS e Camargo Corrêa. As "quatro irmãs", maiores empreiteiras do país, participam individualmente ou de modo consorciado nas dez maiores obras na cidade, orçadas em quase R\$ 30 bilhões. São elas: Porto Maravilha: reforma do Maracanã e entorno: PAC Favelas; corredores Transcarioca e Transolímpica; Arco Metropolitano; Recuperação das Lagoas da Barra e Jacarepaguá; Parque Olímpico; Vila dos Atletas; Linha 4 do Metrô; e VLT do Centro. A Odebrecht está em nove das dez obras. Já a Andrade Gutierrez está em oito, a OAS em sete e a Camargo Corrêa em quatro. Além desta presença ostensiva que sugere uma espécie de rodízio, chama a atenção que elas estão juntas em obras como o Arco Metropolitano e a Transolímpica. Estão igualmente no Consórcio VLT Carioca, responsável pela obra do VLT no Centro do Rio. Neste caso, uma licitação sem concorrentes, em que às quatro empreiteiras se juntou a Riopar Participações, controladora das empresas da família Barata, que domina o transporte público na cidade. Irregularidades já vieram à público nas licitações relativas à exploração do Complexo do Maracanã, envolvendo Odebrecht, OAS e Andrade Gutierrez; e no PAC Favelas, compreendendo Odebrecht, Andrade Gutierrez e Queiroz Galvão. A formação de cartel visando fraudar licitações gera sobrepreço nas obras, bem como o encarecimento dos serviços para a população, como de resto tem acontecido na cidade. Resta indagar quem banca este jogo? A resposta já se sabe: a banca pública, seja por meio de recursos orçamentários, isenções fiscais, seja pelo financiamento generoso dos bancos públicos, notadamente do BNDES. Cabe, enfim, perguntar qual seria o interesse da banca pública? Entre as eleições de 2002 e 2012, somente as "quatro irmãs" despejaram meio bilhão de reais nas campanhas eleitorais. No Estado do Rio, o PMDB é de longe o partido mais beneficiado, com R\$ 6,3 milhões, mas que a soma dos quatro seguentes: PT, PSDB, PV e DEM. Porém os repasses podem ser ainda maiores em anos não eleitorais. Em 2013, somente a Odebrecht repassou R\$ 11 milhões dos R\$ 17 milhões arrecadados pelo PMDB. Esta na hora de acabar com esta jogatina: embora não seja remédio, o financiamento público de campanhas eleitorais e dos partidos, proibindo contribuições de pessoas jurídicas é um caminho. **João Roberto Lopes Pinto é professor da Escola de Ciência Política da Unirio***

## b) XML

Segue um exemplo de uma saída neste formato na Figura 24:

## Figura

```
<?xml version="1.0"?><posts><post xmlns:dc="http://purl.org/dc/elements/1.1/"><URI>URI-teste-diferente1</URI><dc:abstract>1</dc:abstract><dc:accessRights>1</dc:accessRights><dc:tableOfContents>Welcome to WordPress. This is your first post. Edit or delete it, then start blogging!</dc:tableOfContents><post><post xmlns:dc="http://purl.org/dc/elements/1.1/"><URI>URI-teste-diferente29042</URI><dc:abstract>29042</dc:abstract><dc:accessRights>8</dc:accessRights><dc:tableOfContents>Terminada a Copa do Mundo é hora de voltar à realidade Publicado em O Dia, 11/07/2014 Rio - Terminada a Copa do Mundo é hora de voltar à realidade. Pesquisa 'Quem são os proprietários do Brasil?', no Instituto Mais Democracia e da Cooperativa EITA, aponta fortes indícios de formação de cartel nas obras da Copa e Olimpíadas no Rio, envolvendo Odebrecht, Andrade Gutierrez, OAS e Camargo Corrêa. As "quatro irmãs", maiores empreiteiras do país, participam individualmente ou de modo consorciado nas dez maiores obras na cidade, orçadas em quase R$ 30 bilhões. São elas: Porto Maravilha: reforma do Maracanã e entorno: PAC Favelas; corredores Transcarioca e Transolímpica; Arco Metropolitano; Recuperação das Lagoas da Barra e Jacarepaguá; Parque Olímpico; Vila dos Atletas; Linha 4 do Metrô; e VLT do Centro. A Odebrecht está em nove das dez obras. Já a Andrade Gutierrez está em oito, a OAS em sete e a Camargo Corrêa em quatro. Além desta presença ostensiva que sugere uma espécie de rodízio, chama a atenção que elas estão juntas em obras como o Arco Metropolitano e a Transolímpica. Estão igualmente no Consórcio VLT Carioca, responsável pela obra do VLT no Centro do Rio. Neste caso, uma licitação sem concorrentes, em que às quatro empreiteiras se juntou a Riopar Participações, controladora das empresas da família Barata, que domina o transporte público na cidade. Irregularidades já vieram à público nas licitações relativas à exploração do Complexo do Maracanã, envolvendo Odebrecht, OAS e Andrade Gutierrez; e no PAC Favelas, compreendendo Odebrecht, Andrade Gutierrez e Queiroz Galvão. A formação de cartel visando fraudar licitações gera sobrepreço nas obras, bem como o encarecimento dos serviços para a população, como de resto tem acontecido na cidade. Resta indagar quem banca este jogo? A resposta já se sabe: a banca pública, seja por meio de recursos orçamentários, isenções fiscais, seja pelo financiamento generoso dos bancos públicos, notadamente do BNDES. Cabe, enfim, perguntar qual seria o interesse da banca pública? Entre as eleições de 2002 e 2012, somente as "quatro irmãs" despejaram meio bilhão de reais nas campanhas eleitorais. No Estado do Rio, o PMDB é de longe o partido mais beneficiado, com R$ 6,3 milhões, mas que a soma dos quatro seguentes: PT, PSDB, PV e DEM. Porém os repasses podem ser ainda maiores em anos não eleitorais. Em 2013, somente a Odebrecht repassou R$ 11 milhões dos R$ 17 milhões arrecadados pelo PMDB. Esta na hora de acabar com esta jogatina: embora não seja remédio, o financiamento público de campanhas eleitorais e dos partidos, proibindo contribuições de pessoas jurídicas é um caminho. João Roberto Lopes Pinto é professor da Escola de Ciência Política da Unirio

```



## c) RDF

Segue um exemplo de uma saída neste formato na Figura 25:

## Figura

<?xml version="1.0" encoding="UTF-8"?><rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" xmlns:dc="http://purl.org/dc/elements/1.1/"><rdf:Description rdf:about="URI-teste-diferente1"><dc:abstract>1</dc:abstract><dc:accessRights>1</dc:accessRights><dc:tableOfContents>Welcome to WordPress. This is your first post. Edit or delete it, then start blogging!</dc:tableOfContents></rdf:Description><rdf:Description rdf:about="URI-teste-diferente29042"><dc:abstract>29042</dc:abstract><dc:accessRights>8</dc:accessRights><dc:tableOfContents>Terminada a Copa do Mundo é hora de voltar à realidade Publicado em O Dia, 11/07/2014 Rio - Terminada a Copa do Mundo é hora de voltar à realidade. Pesquisa "Quem são os proprietários do Brasil?", do Instituto Mais Democracia e Cooperativa EITA, aponta fortes indícios de formação de cartel nas obras da Copa e Olimpíadas no Rio, envolvendo Odebrecht, Andrade Gutierrez, OAS e Camargo Corrêa. As "quatro irmãs", maiores empreiteiras do país, participam individualmente ou de modo consorciado nas dez maiores obras na cidade, orçadas em quase R\$ 30 bilhões. São elas: Porto Maravilha, reforma do Maracanã e entorno, PAC Favelas, corredores Transcarioca e Transolímpica, Arco Metropolitano, Recuperação das Lagoas da Barra e Jacarepaguá, Parque Olímpico, Vila dos Atletas, Linha 4 do Metrô, e VLT do Centro. A Odebrecht está em nove das dez obras. Já a Andrade Gutierrez está em oito, a OAS em sete e a Camargo Correa em quatro. Além desta presença ostensiva que sugere uma espécie de rodizio, chama a atenção que elas estão juntas em obras como o Arco Metropolitano e a Transolímpica. Estão igualmente no Consórcio VLT Carioca, responsável pela obra do VLT no Centro do Rio. Neste caso, uma licitação sem concorrentes, em que as quatro empreiteiras se juntou a Riopar Participações, controladora das empresas da família Barreto, que domina o transporte público na cidade. Inregularidades já vieram à público nas licitações relativas à exploração do Complexo do Maracanã, envolvendo Odebrecht e OAS; na Recuperação das Lagoas da Barra e Jacarepaguá, implicando Odebrecht, OAS e Andrade Gutierrez; e no PAC Favelas, compreendendo Odebrecht, Andrade Gutierrez e Queiroz Galvão. A formação de cartel visando fraudar licitações gera sobrepreço nas obras, bem como o encarecimento dos serviços para a população, como de resto tem acontecido na cidade. Resta indagar quem banca este jogo? A resposta já se sabe: a banca pública, seja por meio de recursos orçamentários, isenções fiscais, seja pelo financiamento generoso dos bancos públicos, notadamente do BNDES. Cabe, enfim, perguntar qual seria o interesse da banca pública? Entre as eleições de 2002 e 2012, somente as "quatro irmãs" despejaram meio bilhão de reais nas campanhas eleitorais. No Estado do Rio, o PMDB é de longe o partido mais beneficiado, com R\$ 6,3 milhões, mas que a soma dos quatro seguintes: PT, PSDB, PV e DEM. Porém os repasses podem ser ainda maiores em anos não eleitorais. Em 2013, somente a Odebrecht repassou R\$ 11 milhões dos R\$ 17 milhões arrecadados pelo PMDB. Está na hora de acabar com esta jogatina: embora não seja remédio, o financiamento público de campanhas eleitorais e dos partidos, proibindo contribuições de pessoas jurídicas é um caminho. João Roberto Lopes Pinto é professor da Escola de Ciência Política da Unirio</dc:tableOfContents></rdf:Description><rdf:Description rdf:about="URI-teste-diferente26983"><dc:abstract>26983</dc:abstract><dc:accessRights>9</dc:accessRights>



## 5 Exemplos de aplicação

Esta seção apresenta duas aplicações da ferramenta e seus fluxos de execução. Para os dois tipos diferentes, *post* e *whoowns\_owner*, foi aplicado o processo de triplificação com o *plug-in* proposto.

O tipo *post* é derivado do próprio banco de dados do WordPress, cujo tamanho é em torno de 40 MB. No caso do Proprietários do Brasil, o banco de dados corresponde a cerca de 80 MB (somente foi utilizada uma parte do banco, a parte que interessou para este trabalho) e contém o tipo *whoowns\_owner*. Após a execução do *plug-in*, a quantidade de dados gerados e exibidos na saída pode variar conforme as opções escolhidas pelo usuário durante o processo.

Para os dois tipos são feitos o mesmo procedimento: é escolhida a URL base onde o usuário deverá acessar para recuperar os dados triplificados, bem como o tipo. Logo após a URI base do tipo é definida e são feitas as correspondências, os mapeamentos das colunas do banco com os predicados que cada coluna assumirá no resultado final. Por fim, são exibidos os resultados do procedimento em seus três formatos possíveis: JSON-LD, XML e RDF.

### 5.1 Tipo *post*

Conforme abordado na subseção 3.1.3, os conteúdos no WordPress possuem um tipo. O tipo *post* é um dos tipos mais comuns no WordPress e ele foi escolhido

para mostrar que qualquer tipo, desde que elaborada uma ontologia para ele, pode ser triplificado utilizando a ferramenta proposta.

### 5.1.1 URL base e post-type

Escolhemos uma URL base de exemplo “dados-triplificados”, portanto a URL base final para a recuperação dos dados será “<http://localhost/wordpress/dados-triplificados>”. Logo abaixo são listados todos os tipos existentes no banco de dados do WordPress, e é selecionado o tipo que desejamos triplificar (*post*), conforme mostra a Figura 26.

Pode-se notar também que a parte de “tipos já triplificados” está vazia, pois o processo de triplificação nunca havia sido feito antes nesta instalação do WordPress.

**Figura**

Enter the triplified data's URL basis:

Already triplified Post-types:

Search a Post-type and manually triplify it:

- attachment
- nav\_menu\_item
- page
- portfolio
- post
- revision
- slide
- whoowns\_bndes\_proj
- whoowns\_ed\_candidate
- whoowns\_ed\_donation
- whoowns\_owner

Browse

### 5.1.2 URI e suas correspondências

Nesta etapa do processo o usuário deve estabelecer a URI base do tipo a ser

triplificado bem como as correspondências das tabelas no banco para os predicados a constarem nas triplas.

Escolhemos uma URI base de exemplo: “<http://uriexemplo.com/>” (Figura 27). Este valor será adicionado no resultado como sendo sobre quem, ou o que, cada instância se refere. Por exemplo, supondo que o id do *post* é 1, no resultado será acrescentado o seguinte predicado: “`rdf:about`”:“<http://uriexemplo.com/1>”, sendo o formato JSON-LD tomado como exemplo.

Em sequência, através da ontologia elaborada como base, são feitas as correspondências: a coluna “`post_author`” deverá ser transformada em “`foaf:person`”, já que o autor do *post* é uma pessoa, sua data “`post_date`” é traduzida para “`npg:dateTime`”, o conteúdo “`post_content`” possui a correspondência “`owl:hasValue`” e o título do *post* passaria a ser “`dc:title`”. Clicando no botão de salvar opções, as configurações feitas pelo usuário são salvas, conforme explicitado na seção 4.

**Figura**

You are searching for post

Type the URI basis of the posts:

http://www.uriexemplo.i

Enter the correspondences and check the checkbox if the shown result of this column is an URI:

Check the option "not important" if it's correspondence isn't important, so it should not be shown in the final result  
Select "not important" or leave the correspondences with default values has the same results

2- <input type="checkbox"/> <code>post_author =&gt;</code>	<code>foaf:Person</code>	<input type="checkbox"/> not important
3- <input type="checkbox"/> <code>post_date =&gt;</code>	<code>npg:dateTime</code>	<input type="checkbox"/> not important
5- <input type="checkbox"/> <code>post_content =&gt;</code>	<code>owl:hasValue</code>	<input type="checkbox"/> not important
6- <input type="checkbox"/> <code>post_title =&gt;</code>	<code>dc:title</code>	<input type="checkbox"/> not important

**Save options**

### 5.1.3 Resultados

Esta subseção mostra os resultados obtidos em seus três formatos possíveis:

JSON-LD, XML e RDF. Todos os exemplos de saída que constam nesta subseção foram validados e aprovados por seus devidos validadores.

### 5.1.3.1 Resultado em JSON-LD

Este formato tem suas duas partes principais divididas da seguinte maneira: o contexto, ou cabeçalho, onde estão as ontologias que foram usadas no resultado e estão dentro do símbolo “@context”. Por exemplo, se nos dados triplificados tiver algum termo da ontologia “foaf”, dentro desse “@context” haverá necessariamente a explicitação da URI da ontologia, no caso: “foaf”:“{@id”：“http://xmlns.com/foaf/0.1”}.

O símbolo “@id” serve, dentro deste formato para dizer que é uma URI (Figura 28). A segunda parte são os dados em si, triplificados, definidos pelo símbolo “@graph”, voltando ao exemplo do *post* de id 1, este é seu resultado final:

```
{"rdf:about": "http://www.uriexemplo.com//1", "foaf:Person": "1", "npg:dateTime": "2015-01-14 01:07:18", "owl:hasValue": "Welcome to WordPress. This is your first post. Edit or delete it, then start blogging!", "dc:title": "Hello world!"}, com os valores estabelecidos pelo usuário e seus respectivos conteúdos.
```

Figura

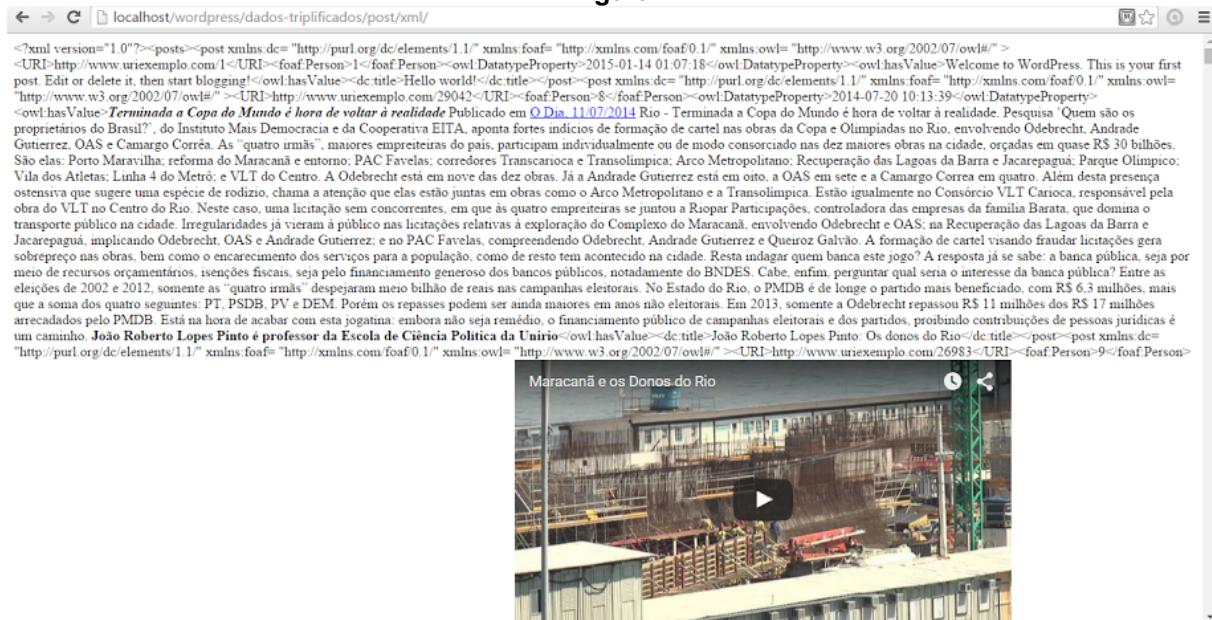


### 5.1.3.2 Resultado em XML

O formato em XML possui algumas diferenças quanto ao formato JSON-LD. Primeiro ele vem com <?xml version="1.0"?> no topo, indicando que é o formato XML. Segundo que todos os *posts* vêm dentro da tag <posts>...</posts>, e cada *post* individual vem dentro da tag <post>...</post>. E por último, cada *post* vem com seu próprio contexto (Figura 29), portanto, voltando novamente ao exemplo de *post* com ID = 1, o resultado é este:

```
<post xmlns:dc= "http://purl.org/dc/elements/1.1/" xmlns:foaf=
"http://xmlns.com/foaf/0.1/" xmlns:owl= "http://www.w3.org/2002/07/owl#" xmlns:rdf=
"http://www.w3.org/1999/02/22-rdf-syntax-ns#"> <rdf:about>
http://www.uri exemplo.com/1 </rdf:about> <foaf:Person> 1 </foaf:Person>
<npg:dateTime> 2015-01-14 01:07:18 </npg:dateTime> <owl:hasValue> Welcome to
WordPress. This is your first post. Edit or delete it, then start
```

Figura



blogging!</owl:hasValue> <dc:title> Hello world! </dc:title> </post>.

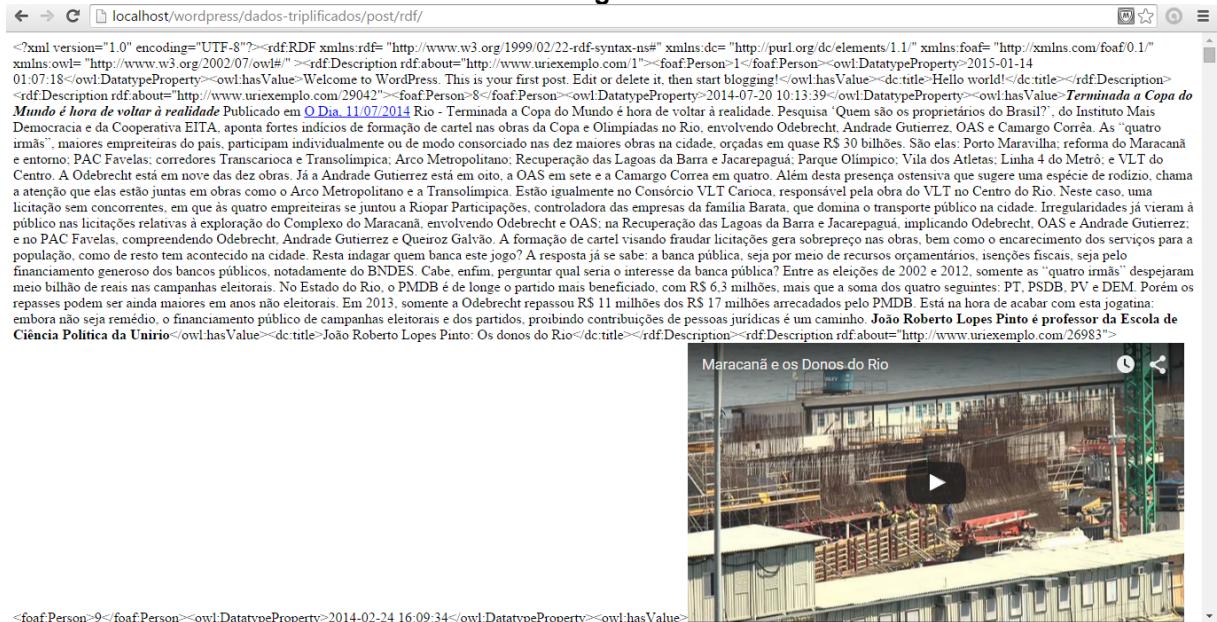
### 5.1.3.3 Resultado em RDF

O resultado em RDF é muito semelhante ao formato XML, exceto por duas diferenças: a tag definindo sobre quem, ou que é determinada triplificação, é a <rdf:Description> ... </rdf:Description>. Outra diferença é que o contexto, assim como no JSON-LD, é único para todos os dados (Figura 30). No exemplo do *post* de

ID = 1, o resultado seria este:

```
<rdf:Description>
rdf:about="http://www.uri exemplo.com/1"><foaf:Person>1</foaf:Person><npg:dateTi
me>2015-01-14 01:07:18</npg:dateTime><owl:hasValue>Welcome to WordPress.
This is your first post. Edit or delete it, then start
blogging!</owl:hasValue><dc:title>Hello world!</dc:title></rdf:Description>.
```

**Figura**



## 5.2 Tipo “whoowns\_owner”

Este tipo é o que possui os dados relacionados ao projeto Proprietários do Brasil. Importante mencionar que a ontologia desenvolvida para o projeto não abrange todos os conceitos contidos no site. Destacou-se o mapeamento para as relações entre as empresas e pessoas. A ontologia é dada por uma composição de conceitos novos e de outros vocabulários já existentes, como apresentado na subseção 2.2.2.2.

### 5.2.1 URL base e post-type

Assim como no exemplo da subseção 5.1, começa-se preenchendo a URL base, caso ainda não tenha sido preenchida, ou caso deseja-se trocá-la. Escolhemos por não a alterar, deixando-a como “<http://localhost/wordpress/dados-triplificados>”. Neste exemplo, exibido na Figura 31, desejamos triplificar o tipo

“whoowns\_owner” e então, o selecionamos para iniciar a definição de correspondências.

**Figura**

**Enter the triplified data's URL basis:**

`http://localhost/wordpress/` `dados-triplificados`

**Already triplified Post-types:**

`post` `whoowns_owner`

**Search a Post-type and manually triplify it:**

- attachment
- nav\_menu\_item
- page
- portfolio
- post
- revision
- slide
- whoowns\_bndes\_proj
- whoowns\_ed\_candidate
- whoowns\_ed\_donation
- whoowns\_owner

**Browse**

### 5.2.2 URI e suas correspondências

Como no exemplo da subseção 5.1, o próximo passo é definir a URI do tipo que está sendo triplificado, neste tipo optamos por escolher a URI “<http://proprietariosdobrasil.org.br/owner>”. Conforme a ontologia proposta, deve-se definir as correspondências. Neste exemplo em específico, há uma pequena discordância entre a ontologia e os valores definidos nas correspondências por diversos motivos, por exemplo, valores disponíveis no banco de dados.

Neste tipo, o nome do agente está na coluna “post\_title”, por isso a equivalência dela é “foaf:name”. O registro do agente está na coluna whoowns\_legal\_registration, que foi definida como “rov:registration” na

correspondência, bem como sua nacionalidade, que estava na coluna `whoowns_nationality` foi traduzida para “`pext:nationality`”.

Nas outras duas colunas, o valor do objeto na tripla não é um simples valor, é uma chave estrangeira, nesse caso específico então, é um outro “`whoowns_owner`”. Por causa disto, o checkbox à esquerda do nome da coluna foi marcado e definida a URI a ser concatenada no conteúdo dessa coluna na tabela. A coluna “`whoowns_controlled_by`” é o agente que controla a organização e a coluna “`whoowns_controlled_by_final`” é controlador final da organização em questão, ambos são mapeadas com o mesmo predicado: “`org:transitiveSuborganizationOf`”. Podemos observar estas informações na Figura 32.

**Figura**

You are searching for `whoowns_owner`

Type the URI basis of the posts:

http://proprietariosdobr

Enter the correspondences and check the checkbox if the shown result of this column is an URI:

Check the option “not important” if it’s correspondence isn’t important, so it should not be shown in the final result  
Select “not important” or leave the correspondences with default values has the same results

6-  `post_title => foaf:name`  not important

32-  `whoowns_legal_registration => rov:registration`  not important

40-  `whoowns_nationality => pext:nationality`  not important

43-  `http://proprietariosdobr whoowns_controlled_by => org:transitiveSuborganiz`  not important

44-  `http://proprietariosdobr whoowns_controlled_by_final => org:transitiveSuborganiz`  not important

**Save options**

### 5.2.3 Resultados

A exemplo da subseção 5.1.3, esta subseção contém os resultados da triplicação. Os comentários em cada subseção são exatamente os mesmos feitos na subseção 5.1.3, exceto para o fato de que as colunas assinaladas como URI no caso, foram adicionadas como as URI's inseridas pelo usuário.

### 5.2.3.1 Resultado em JSON-LD

Acessando a URL definida e visualizando o resultado em JSON-LD na Figura 33:

**Figura**

The screenshot shows a browser window with the URL `localhost/wordpress/dados-triplificados/whoowns_owner/`. The page content is a large JSON-LD document, likely a dump of a triple store. It contains numerous triples with subjects, predicates, and objects, primarily in the foaf namespace. Some recognizable names include "INDUSTRIAS GESSY LEVER LTDA", "JOSEPH JEAN FRANCOIS BOURQUARD", "HANS PETER WAHL", "VOLKSWAGEN COMERCIO E PARTICIPACOES LTDA", "ALEXANDRE MAHLER", "JACQUES DEFFOREY", "TOYOTA DO BRASIL LTDA", "REPREA COMERCIO E PARTICIPACOES LTDA", "TONIPART PARTICIPACOES S C LTDA", "DAIMLERCHRYSLER DO BRASIL LTDA", "SIDERURGICA MENDES JUNIOR S/A", "NESTLE INDUSTRIAL E COMERCIAL LTDA", "RICARDO Luiz dos Santos Carvalho", "CARRIFOUR COMERCIO E INDUSTRIA LTDA", "EDUARDO DE QUEIROZ GALVAO", "ARCS DOURADOS COMERCIO DE ALIMENTOS LTDA", "FIC DISTRIBUIDORA DE DERIVADOS DE PETROLEO LTDA", "ARCOS DOURADOS COMERCIO DE ALIMENTOS LTDA", "WYNDHAM COMERCIO DE PRODUTOS ALIMENTICIOS LTDA", "HONDA SOUTH AMERICA LTDA", "VOLKSWAGEN DO BRASIL LTDA", "LABORATORIOS PFIZER LTDA", "LBR - Lácteos Brasil S/A", "BRIDGESTONE do Brasil Industrias e Comérico Ltda", "Atento Brasil S/A", "SIEMENS LTDA", "ALVARO MALVEIRO", "EVERALDO NIGRO DOS SANTOS", "JOAO CAIO GOULART PENTEADO", and "ALCAN EMPREENDIMENTOS LTDA". The JSON-LD structure includes context blocks and various predicates like `foaf:name`, `foaf:knows`, and `foaf:workFor`.

### 5.2.3.2 Resultado em XML

Acessando a URL definida e visualizando o resultado em XML na Figura 34:

## Figura

<?xml version="1.0"?><posts><post xmlns:foaf="http://xmlns.com/foaf/0.1/" xmlns:pext="http://www.ontotext.com/proton/protonext" xmlns:rov="http://www.w3.org/ns/regorg"><URI>http://proprietariosdobrasil.org.br/owner26887</URI><rov:registration>01340937000179</rov:registration><foaf:name>Galvão Engenharia S/A</foaf:name><post><post xmlns:foaf="http://xmlns.com/foaf/0.1/" xmlns:pext="http://www.ontotext.com/proton/protonext" xmlns:rov="http://www.w3.org/ns/regorg"><URI>http://proprietariosdobrasil.org.br/owner26872</URI><rov:registration>9081076000101</rov:registration><foaf:name>FERTIPAR FERTILIZANTES DO PARANA LTDA</foaf:name><post><post xmlns:foaf="http://xmlns.com/foaf/0.1/" xmlns:pext="http://www.ontotext.com/proton/protonext" xmlns:rov="http://www.w3.org/ns/regorg"><URI>http://proprietariosdobrasil.org.br/owner26871</URI><rov:registration>6106827600104</rov:registration><foaf:name>INDUSTRIAS GESSY LEVERLTDA</foaf:name><post><post xmlns:foaf="http://xmlns.com/foaf/0.1/" xmlns:pext="http://www.ontotext.com/proton/protonext" xmlns:rov="http://www.w3.org/ns/regorg"><URI>http://proprietariosdobrasil.org.br/owner26853</URI><foaf:name>NESTLE S. A.</foaf:name><post><post xmlns:foaf="http://xmlns.com/foaf/0.1/" xmlns:pext="http://www.ontotext.com/proton/protonext" xmlns:rov="http://www.w3.org/ns/regorg"><URI>http://proprietariosdobrasil.org.br/owner26851</URI><rov:registration>01417663871</rov:registration><foaf:name>MARIO LUIZ OTAROLA</foaf:name><post><post xmlns:foaf="http://xmlns.com/foaf/0.1/" xmlns:pext="http://www.ontotext.com/proton/protonext" xmlns:rov="http://www.w3.org/ns/regorg"><URI>http://proprietariosdobrasil.org.br/owner26849</URI><rov:registration>07770184894</rov:registration><foaf:name>JOSEPH JEAN FRANCOIS BOURQUARD</foaf:name><post><post xmlns:foaf="http://xmlns.com/foaf/0.1/" xmlns:pext="http://www.ontotext.com/proton/protonext" xmlns:rov="http://www.w3.org/ns/regorg"><URI>http://proprietariosdobrasil.org.br/owner26847</URI><rov:registration>03695972815</rov:registration><foaf:name>HANS PETER WAHLI</foaf:name><post><post xmlns:foaf="http://xmlns.com/foaf/0.1/" xmlns:pext="http://www.ontotext.com/proton/protonext" xmlns:rov="http://www.w3.org/ns/regorg"><URI>http://proprietariosdobrasil.org.br/owner26843</URI><rov:registration>01440806802</rov:registration><foaf:name>ANTONIO SALGADO PERES FILHO</foaf:name><post><post xmlns:foaf="http://xmlns.com/foaf/0.1/" xmlns:pext="http://www.ontotext.com/proton/protonext" xmlns:rov="http://www.w3.org/ns/regorg"><URI>http://proprietariosdobrasil.org.br/owner26841</URI><rov:registration>0146827876</rov:registration><foaf:name>ALEXANDRE MAHLER</foaf:name><post><post xmlns:foaf="http://xmlns.com/foaf/0.1/" xmlns:pext="http://www.ontotext.com/proton/protonext" xmlns:rov="http://www.w3.org/ns/regorg"><URI>http://proprietariosdobrasil.org.br/owner26839</URI><foaf:name>VOLKSWAGEN COMERCIO E PARTICIPACOES LTDA</foaf:name><post><post xmlns:foaf="http://xmlns.com/foaf/0.1/" xmlns:pext="http://www.ontotext.com/proton/protonext" xmlns:rov="http://www.w3.org/ns/regorg"><URI>http://proprietariosdobrasil.org.br/owner26837</URI><rov:registration>06353230877</rov:registration><foaf:name>PETER FRAUENDORF</foaf:name><post><post xmlns:foaf="http://xmlns.com/foaf/0.1/" xmlns:pext="http://www.ontotext.com/proton/protonext" xmlns:rov="http://www.w3.org/ns/regorg"><URI>http://proprietariosdobrasil.org.br/owner26833</URI><foaf:name>TONIPART PARTICIPACOES S/C LTDA</foaf:name><post><post xmlns:foaf="http://xmlns.com/foaf/0.1/" xmlns:pext="http://www.ontotext.com/proton/protonext" xmlns:rov="http://www.w3.org/ns/regorg"><URI>http://proprietariosdobrasil.org.br/owner26830</URI><rov:registration>20542333821</rov:registration><foaf:name>JACQUES DEFFOREY</foaf:name><post><post xmlns:foaf="http://xmlns.com/foaf/0.1/" xmlns:pext="http://www.ontotext.com/proton/protonext" xmlns:rov="http://www.w3.org/ns/regorg"><URI>http://proprietariosdobrasil.org.br/owner26828</URI><rov:registration>4974005800190</rov:registration><foaf:name>BREPA COMERCIO E PARTICIPACOES LTDA</foaf:name><post><post xmlns:foaf="http://xmlns.com/foaf/0.1/" xmlns:pext="http://www.ontotext.com/proton/protonext" xmlns:rov="http://www.w3.org/ns/regorg"><URI>http://proprietariosdobrasil.org.br/owner26822</URI><rov:registration>59104761000191</rov:registration><foaf:name>TOYOTA DO BRASIL LTDA</foaf:name><post><post xmlns:foaf="http://xmlns.com/foaf/0.1/" xmlns:pext="http://www.ontotext.com/proton/protonext" xmlns:rov="http://www.w3.org/ns/regorg"><URI>http://proprietariosdobrasil.org.br/owner26928</URI><rov:registration>59104273000129</rov:registration><foaf:name>DAIMLERCHRYSLER DO BRASIL LTDA</foaf:name><post><post xmlns:foaf="http://xmlns.com/foaf/0.1/" xmlns:pext="http://www.ontotext.com/proton/protonext" xmlns:rov="http://www.w3.org/ns/regorg"><URI>http://proprietariosdobrasil.org.br/owner26924</URI><rov:registration>1746970100177</rov:registration><foaf:name>SIDERURGICA MENDES JUNIOR S/A</foaf:name><post><post xmlns:foaf="http://xmlns.com/foaf/0.1/" xmlns:pext="http://www.ontotext.com/proton/protonext" xmlns:rov="http://www.w3.org/ns/regorg"><URI>http://proprietariosdobrasil.org.br/owner26921</URI><rov:registration>60409075000152</rov:registration><foaf:name>NESTLE INDUSTRIAL E COMERCIAL LTDA</foaf:name><post><post

### **5.2.3.3 Resultado em RDF**

Acessando a URL definida e visualizando o resultado em RDF na Figura 35:

## Figura

<?xml version="1.0" encoding="UTF-8"?><rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" xmlns:foaf="http://xmlns.com/foaf/0.1/" xmlns:org="http://www.w3.org/ns/org#" xmlns:pext="http://www.tontotext.com/proton/protnext" xmlns:rov="http://www.w3.org/ns/rovgong"><rdf:Description rdf:about="http://proprietariosdobrasil.org.br/owner26887"><rov:registration>01349037000179</rov:registration><foaf:name>Galvão Engenharia S/A</foaf:name><rdf:Description rdf:about="http://proprietariosdobrasil.org.br/owner26872"><rov:registration>9081076000101</rov:registration><foaf:name>FERTIPAR FERTILIZANTES DO PARANA LTDA</foaf:name><rdf:Description rdf:about="http://proprietariosdobrasil.org.br/owner26871"><rov:registration>6106827600104</rov:registration><foaf:name>INDUSTRIAS GESSY LEVER LTDA</foaf:name><rdf:Description rdf:about="http://proprietariosdobrasil.org.br/owner26870"><fov:registration>014081076000101</fov:registration><foaf:name>NESTLE S A</foaf:name><rdf:Description rdf:about="http://proprietariosdobrasil.org.br/owner26861"><rov:registration>0141766371</rov:registration><foaf:name>MARIO LUIZ YON TAROLA</foaf:name><rdf:Description rdf:about="http://proprietariosdobrasil.org.br/owner26849"><rov:registration>07707184894</rov:registration><foaf:name>JOSEPH JEAN FRANCOIS BOURQUARD</foaf:name><rdf:Description rdf:about="http://proprietariosdobrasil.org.br/owner26847"><rov:registration>03695972815</rov:registration><foaf:name>HANS PETER WAHLI</foaf:name><rdf:Description rdf:about="http://proprietariosdobrasil.org.br/owner26843"><rov:registration>01440806802</rov:registration><foaf:name>ANTONIO SALGADO PERES FILHO</foaf:name><rdf:Description rdf:about="http://proprietariosdobrasil.org.br/owner26841"><rov:registration>01468267876</rov:registration><foaf:name>ALEXANDRE MAHLER</foaf:name><rdf:Description rdf:about="http://proprietariosdobrasil.org.br/owner26839"><fov:registration>014081076000101</fov:registration><foaf:name>VOLKSWAGEN COMERCIO E PARTICIPACOES LTDA</foaf:name><rdf:Description rdf:about="http://proprietariosdobrasil.org.br/owner26837"><rov:registration>06532320877</rov:registration><foaf:name>PETER FRAUENDORF</foaf:name><rdf:Description rdf:about="http://proprietariosdobrasil.org.br/owner26835"><fov:registration>014081076000101</fov:registration><foaf:name>TONIPART PARTICIPACOES S/C LTDA</foaf:name><rdf:Description rdf:about="http://proprietariosdobrasil.org.br/owner26830"><rov:registration>2054233821</rov:registration><foaf:name>JACQUES DEFFOREY</foaf:name><rdf:Description rdf:about="http://proprietariosdobrasil.org.br/owner26828"><rov:registration>4974003000190</rov:registration><foaf:name>BREPA COMERCIO E PARTICIPACOES LTDA</foaf:name><rdf:Description rdf:about="http://proprietariosdobrasil.org.br/owner26822"><rov:registration>5910476100191</rov:registration><foaf:name>TOYOTA DO BRASIL LTDA</foaf:name><rdf:Description rdf:about="http://proprietariosdobrasil.org.br/owner26928"><rov:registration>59104273000129</rov:registration><foaf:name>DAIMLERCHRYSLER DO BRASIL LTDA</foaf:name><rdf:Description rdf:about="http://proprietariosdobrasil.org.br/owner26924"><rov:registration>17469701000177</rov:registration><foaf:name>SIDERQUICIA MENDES JUNIOR S/A</foaf:name><rdf:Description rdf:about="http://proprietariosdobrasil.org.br/owner26921"><rov:registration>6040907000152</rov:registration><foaf:name>NESTLE INDUSTRIAL E COMERCIAL LTDA</foaf:name><rdf:Description rdf:about="http://proprietariosdobrasil.org.br/owner26919"><rov:registration>58460896820</rov:registration><foaf:name>Ricardo Luiz dos Santos Carvalho</foaf:name><rdf:Description rdf:about="http://proprietariosdobrasil.org.br/owner26908"><rov:registration>45543915000181</rov:registration><foaf:name>CARREFOUR COMERCIO E INDUSTRIA LTDA</foaf:name><rdf:Description rdf:about="http://proprietariosdobrasil.org.br/owner26906"><rov:registration>30996945334</rov:registration><foaf:name>EDUARDO DE QUEIROZ GALVÃO</foaf:name><rdf:Description rdf:about="http://proprietariosdobrasil.org.br/owner26883"><rov:registration>01349764000150</rov:registration><foaf:name>FIC DISTRIBUIDORA DE DERIVADOS DE PETROLEO LTDA</foaf:name><rdf:Description rdf:about="http://proprietariosdobrasil.org.br/owner26879"><rov:registration>42591651000143</rov:registration><org:transitiveSubOrganizationOf><proprietariosdobrasil.org.br/owner26742><org:transitiveSubOrganizationOf><foaf:name>ARCOS DOURADOS COMERCIO DE ALIMENTOS LTDA</foaf:name><rdf:Description rdf:about="http://proprietariosdobrasil.org.br/owner26877"><rov:registration>5876003000136</rov:registration><org:transitiveSubOrganizationOf><proprietariosdobrasil.org.br/owner26733><org:transitiveSubOrganizationOf><foaf:name>WYNNDHAM COMERCIO DE PRODUTOS ALIMENTICIOS LTDA</foaf:name><rdf:Description rdf:about="http://proprietariosdobrasil.org.br/owner26784"><rov:registration>6074463000190</rov:registration><foaf:name>ZENECA SEMENTES LTDA</foaf:name><rdf:Description rdf:about="http://proprietariosdobrasil.org.br/owner26869"><rov:registration>43149806000159</rov:registration><foaf:name>HONDA SOUTH AMERICA LTDA</foaf:name><rdf:Description rdf:about="http://proprietariosdobrasil.org.br/owner26819"><rov:registration>59104422000150</rov:registration><foaf:name>VOLKSWAGEN DO BRASIL LTDA</foaf:name><rdf:Description rdf:about="http://proprietariosdobrasil.org.br/owner26881"><rov:registration>46070685000169</rov:registration><org:transitiveSubOrganizationOf><proprietariosdobrasil.org.br/owner26744><org:transitiveSubOrganizationOf><foaf:name>LABORATORIOS

## 6 Conclusão

Para fechar o projeto, disponibilizamos o endereço onde nosso *plug-in* pode ser baixado: <https://br.wordpress.org/plugins/data-triplify/>. Fora isto, temos esta seção com as dificuldades encontradas, as possibilidades de uso e os trabalhos futuros, onde deixamos algumas ideias que poderão ser acrescentadas para a melhora do *plug-in*.

Foi desenvolvido um *plug-in* para WordPress onde o objetivo é que o usuário possa divulgar seus dados, unir com outras bases, etc. O *plug-in* é uma ferramenta capaz de triplificar os dados contidos no banco do WordPress seja eles quais forem e de que natureza forem.

O WordPress, apesar de ter uma documentação extensa e farta, não tinha sido utilizado previamente pelos autores deste trabalho, assim como a linguagem PHP. Apesar de ser uma linguagem bastante comum, cada linguagem tem suas nuances, e para alguém que nunca trabalhou com ela antes, foram encontrados diversos problemas que, em outras linguagens, talvez fossem triviais de resolver.

As possibilidades de uso incluem qualquer usuário que tenha permissão de administrador e utilize o WordPress. O “Proprietários do Brasil” foi um exemplo que poderia servir para o *plug-in*. Basta que a base de dados onde se queira triplificar, esteja armazenada em algum lugar e seja acessível a partir da instalação do WordPress que foi realizada pelo usuário administrador.

Como trabalhos futuros, pensamos que o usuário poderia adicionar algum formato ainda não implementado, fazendo *upload* de um arquivo contendo as regras (ou uma classe) de como esse novo formato dispõe os dados, além de salvar em algum lugar do banco o nome desse novo formato. Fora isto, o usuário também poderia fazer o *upload* de um prefixo novo, e o mesmo aparecer no *autocomplete*. Outro detalhe, é a adição de uma chave estrangeira, por exemplo. Infelizmente é um processo manual um pouco mais complicado e que talvez possa ser melhorado.

Para finalizar, o *plug-in* também teria a possibilidade de oferecer uma área onde os dados seriam interligados ou armazenados como trabalho futuro. O próprio DBpedia, por exemplo, o que eliminaria a própria necessidade do usuário de fazê-lo.

## **Referências**

- Alonso, J.M. et al. Melhorando o acesso ao governo com o melhor uso da web. Disponível em: <<http://www.w3c.br/pub/Materiais/PublicacoesW3C/gov-web.pdf>>. Acesso em 11 de fevereiro de 2015.
- Antoniou, G.; and van Harmelen, F. A semantic web primer. MIT press, 2004.
- Berners-Lee, T.; Hendler, J.; and Lassila, O. The semantic web. *Scientific American*, p. 29-37, mai. 2001.
- Boff, L. Corrupção: crime contra a sociedade. Disponível em: <<http://www.jb.com.br/leonardo-boff/noticias/2012/04/15/corrupcao-crime-contra-a-sociedade/>>. Acesso em 11 de fevereiro de 2015.
- Cass, S. Top 10 Programming Languages. Disponível em: <<http://spectrum.ieee.org/computing/software/top-10-programming-languages>>. Acesso em 3 de fevereiro de 2015.
- European Union Open Data Portal. Data. Disponível em: <<http://open-data.europa.eu/en/data/>>. Acesso em 3 de fevereiro de 2015.
- Europeana Labs. Europeana Linked Open Data. Disponível em: <<http://labs.europeana.eu/api/linked-open-data-introduction>>. Acesso em 18 de fevereiro de 2015.
- Freire, P. Pedagogia da indignação: cartas pedagógicas e outros escritos. Editora UNESP, 2000.
- Mo'mix Solutions. Benefits of Open Government Transparency for Better America. Disponível em: <<http://momixpc.com/momix-infographic-benefits-of-government-transparency>>. Acesso em 1 de março de 2015.
- Patel, S.K.; Rathod, V.R.; and Parikh, S. Joomla, Drupal and WordPress - A

Statistical Comparison of Open Sources CMS. *IEE Trends in Information Sciences and Computing (TISC)*, p. 182-187, 2011.

Patel, S.K.; Rathod, V.R.; and Prajapati, J.B. Performance Analysis of Content Management Systems-Joomla, Drupal and Wordpress. *International Journal of Computer Applications*, v. 21, n. 4, p. 39-43, 2011.

Paulheim, H.; and Fürnkranz, J. Unsupervised Feature Generation from Linked Open Data. *International Conference on Web Intelligence, Mining, and Semantics (WIMS'12)*, 2012.

Powers, S. Practical RDF. O'Reilly Media, 2003.

Sabin-Wilson, L. Wordpress for Dummies. John Wiley & Sons, 2011.

Shadbolt, N.; Hall W.; and Berners-Lee T. The semantic web revisited, *IEEE Intelligent Systems*, v. 21, n. 3, p. 96-101, 2006.

Simons, R. UK open government data: the results of the official audit. Disponível em: <<http://www.theguardian.com/news/datablog/2012/apr/18/uk-open-government-data-national-audit-office>>. Acesso em 18 de fevereiro de 2015.

Singer, P. Introdução à Economia Solidária. Fundação Perseu Abramo, 2002.

Tygel, A. F.; Tygel, D.; and Campos, M.L.M. Who owns Brazil? The complex network of power and ownership in the Brazilian market. COMPLEX NETWORKS 2013 - Second International Workshop on Complex Networks and their Applications, Kioto, Japan, 2013.

W3C. W3C Semantic Web FAQ. Disponível em: <<http://www.w3.org/2001/sw/SW-FAQ>>. Acesso em 3 de fevereiro de 2015.