



Universidade Federal de Pernambuco  
Centro de Informática

Graduação em Ciência da Computação

Gabriel do Amaral Moreira

**GovPEX3: uma ferramenta para publicação das despesas  
do Governo de Pernambuco em RDF**

Recife,  
abril de 2013

Gabriel do Amaral Moreira

## **GovPEX3: uma ferramenta para publicação das despesas do Governo de Pernambuco em RDF**

Trabalho apresentado ao Programa de Graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco como requisito parcial para obtenção do grau de Bacharel em Ciência da Computação.

**Orientadora:** Bernadette Farias Lóscio (bfl@cin.ufpe.br)

Recife,  
abril de 2013

**Dedico este trabalho aos meus pais, Hélio e Vitória, a Vovô Hélio e a Tia Ana.**

*Valeu a pena? Tudo vale a pena  
Se a alma não é pequena.  
Quem quer passar além do Bojador  
Tem que passar além da dor.  
Deus ao mar o perigo e o abismo deu,  
Mas nele é que espelhou o céu.*

**Fernando Pessoa**

## **Agradecimentos**

Agradeço, primeiramente, a minha mãe e Paião, Maria das Vitórias Negreiros do Amaral e Hélio Moreira da Silva Filho, pelo apoio incondicional em todos os momentos da minha vida. Eles sempre estão presentes e torcendo por mim e contribuíram para que eu tivesse gana para concluir mais este trabalho. Devo tudo o que realizei de bom a eles.

Agradeço também ao meu avô Hélio Moreira da Silva e à tia Ana Crescêncio que sempre me passam mensagens animadoras e de confiança, torcem por mim e mandam ótimas energias que me ajudam a vencer as etapas importantes da carreira e da vida.

Agradeço a Stella, meu amor, que de repente, rapidamente e intensamente entrou na minha vida trazendo muito mais amor a ela. Stella esteve sempre comigo durante o período de desenvolvimento deste trabalho e contribuiu muito para que sua realização tenha sido menos estressante e repleta de amor.

Agradeço aos meus irmãos Hélio Moreira da Silva Neto e Rafael do Amaral Moreira e aos meus amigos que também sempre mandaram energias positivas e participaram dos grandes momentos da minha vida. A Marcello Cysneiros Landim Valença, amigo Cientista da Computação que participou comigo de quase todas as empreitadas na graduação em Ciência da Computação.

Agraço a Bernadette Farias Lóscio, professora que me acolheu orientando-me neste trabalho de forma exemplar: atendendo-me atenciosamente, pressionando-me na medida certa quando necessário e abrindo caminhos imprescindíveis para desenvolvimento do projeto. Creio que não poderia ser mais bem orientado.

Finalmente, agradeço aos diretores da Facilit Tecnologia que contribuíram para este trabalho permitindo que ele fosse implementado sobre portais desenvolvidos pela empresa. Também aos colegas da Facilit que me apoiaram e esclareceram dúvidas durante o desenvolvimento do projeto.

## Resumo

A Web Semântica busca dar significado aos dados publicados na web de forma que computadores possam processá-los e reconhecer os conceitos e objetos do mundo real que estão sendo representados. Para isso, a Web Semântica especifica uma pilha de tecnologias que estende a web tradicional, oferecendo dados semanticamente estruturados com o uso do modelo RDF e de ontologias. Os princípios de Linked Data estabelecem diretrizes para a construção de uma rede de dados interligados, possibilitando navegação automática e descoberta de novos dados na web. Dado Aberto é um conceito que classifica os dados que, entre outras coisas, são úteis e podem ser reutilizados indiscriminadamente e processados por um computador. Todos estes temas são importantes para dar mais transparência à gestão pública quando levados em conta na publicação de informações governamentais.

Este trabalho desenvolveu, então, uma ferramenta chamada de GovPEX3 que oferece a possibilidade dos gestores das instituições do Governo de Pernambuco publicarem os dados das despesas de suas instituições em RDF utilizando esforço mínimo. A ferramenta foi implementada sobre o Portal da Lei de Acesso à Informação (LAI) do Governo de Pernambuco, que não dava suporte a publicação de dados abertos estruturados. Como requisito para o desenvolvimento da GovPEX3 foi criada a Ontologia Despesas GovPE, que define o modelo semântico dos dados das despesas do Governo de Pernambuco. A GovPEX3, realizando um processo de ETL (*Extract, Transform and Load*) que usa tecnologias da Web Semântica e segue princípios de *Linked Data*, recupera os dados através de web services do Portal da Transparência do governo, utiliza a Ontologia Despesas GovPE para converter os dados abertos para RDF e os publica no Portal da LAI.

**Palavras-chave:** Web Semântica, *Liked Data*, Dados Abertos, RDF, ontologia.

# Índice

1.	Introdução .....	9
1.1	Motivação.....	9
1.2	Proposta.....	11
1.3	Organização do Documento.....	12
2.	Fundamentação Teórica.....	14
2.1	Web Semântica.....	14
2.1.1	RDF .....	18
2.1.2	Ontologias.....	19
2.3	Linked Data.....	21
2.3.1	Usar URIs para identificar coisas .....	24
2.3.2	Usar URIs HTTP para que se possa acessar os recursos.....	25
2.3.3	Disponibilizar informações úteis utilizando os padrões.....	26
2.3.4	Interligar recursos com outros URIs para que se descubra mais informações .....	27
2.4	Dados Abertos .....	29
3	GovPEx3.....	32
3.1	Portal da Lei de Acesso à Informação.....	32
3.2	Desenvolvimento da Ontologia Despesas GovPE.....	36
3.2.1	Ontologia da Lei Orçamentaria Anual de 2012.....	37
3.2.2	Especificação da Ontologia Despesas GovPE .....	38
3.3	Implementação.....	42
3.3.1	Desenvolvimento dos clientes para os web services.....	44
3.3.2	Adição do tipo RDF ao portal .....	44
3.3.3	Implementação do Portlet “Publicador RDF” .....	45
3.3.4	Desenvolvimento do mapeamento dos dados.....	46

3.3.5	Implementação dos mecanismos de persistência e disponibilização das triplas RDF .....	50
4	Conclusão .....	52
5	Referências .....	55
	APÊNDICE A - Ontologia Despesas GovPE em OWL.....	58
	APÊNDICE B – Script SQL da Tabela de Triplas RDF .....	65

# 1. Introdução

Neste capítulo será feita uma introdução ao trabalho a partir das seguintes perspectivas: motivação para a realização do projeto, proposta de projeto concebida e a descrição da organização deste documento.

## 1.1 Motivação

Atualmente, é indiscutível a importância dos dados para quem os consome, principalmente para os que fazem uso de técnicas e ferramentas provenientes de áreas como Inteligência Artificial (IA), Business Intelligence (BI), Visualização de Dados e Estatísticas. Consumidores estes, que, com a aplicação dessas técnicas e ferramentas, tornam-se capazes de gerar novos conhecimentos, informações úteis e melhores visualizações dos dados. Em ambientes empresariais é muito comum a utilização do processamento de dados para obter ajuda nas tomadas de decisões estratégicas, principalmente quando se dispõe de grandes quantidades de informações.

Isso nos faz pensar no potencial da plataforma World Wide Web (ou apenas web), que constitui uma enorme base de dados distribuída e colaborativa, onde qualquer um pode publicar dados. Por esse motivo, a quantidade de dados disponíveis nesta plataforma é imensa, e cresceu muito, principalmente com a partir da sua segunda geração, na qual os usuários em geral passaram a ter um papel mais ativo, gerando e compartilhando conteúdos. Este ambiente parece muito propício para integração e tratamento de dados, sendo possível a inferência e geração de informações. O ponto negativo é que são publicados quase sempre em formatos não estruturados ou semiestruturados. O formato mais comum na web é a linguagem de marcação Hypertext Markup Language (HTML), que consiste em texto livre mesclado com um conjunto de etiquetas de marcação utilizada pelos navegadores para fins de exibição.

No âmbito da administração pública não é menor a importância dos dados. As instituições governamentais trabalham rotineiramente analisando índices, números, métricas e informações que podem ajudá-las em suas gestões e no cumprimento de seus objetivos. Os dados analisados são produzidos ou não pelos próprios governos e na realização de suas ações e de seus projetos geram ainda mais dados. Um domínio no qual os dados são sempre visados por todos e devem ser bem analisados para evitar inconsistências é o financeiro. Se tratando de instituições públicas, os cidadãos possuem

papel fundamental na fiscalização do destino dado ao dinheiro público, dentre outras coisas.

Seguindo a tendência da atual era da informação, os governos passaram a utilizar a internet como plataforma para aproximar os cidadãos da gestão pública, disponibilizando informações sobre ações e decisões, prestando serviços online e automatizando processos burocráticos e administrativos. Este fenômeno, da atuação governamental através da internet, recebeu o nome de governo eletrônico, ou e-gov. O e-gov pode ser implantado utilizando diversos tipos de ferramentas, como portais web e aplicativos para dispositivos móveis [12].

As pessoas parecem estar se conscientizando do papel delas, de participar das decisões e fiscalizar o poder público. Isso faz com que, cada vez mais, a sociedade clame por transparência e aumento na quantidade de projetos e serviços no governo eletrônico. Respondendo à necessidade de criação de elos entre governo e sociedade, leis garantindo o acesso dos cidadãos à informação foram criadas, o que tornam obrigatórios o registro e a publicação dos dados governamentais. Em 16 de maio de 2012 entrou em vigor a Lei de Acesso à Informação<sup>1</sup> (LAI) federal, e em 29 de outubro do mesmo ano, a versão estadual da lei<sup>2</sup> foi sancionada no Estado de Pernambuco (PE), entrando em vigor em 1º de janeiro de 2013. O então e atual Secretário da Controladoria Geral do Estado, Djalmo de Oliveira Leão mencionou o que representa esta lei: “A Lei de Acesso à Informação respeita os direitos do cidadão, que passa a participar ativamente da sociedade. A consequência disso é a diminuição da corrupção e a melhoria da gestão pública” [13].

Para facilitar o acesso à informação, em cumprimento destas novas leis, criou-se o Portal da LAI<sup>3</sup>, que reúne páginas na web com informações sobre cada órgão do Governo do Estado de Pernambuco. Apesar disso, a publicação de dados realizada pelo portal ainda é carente em vários aspectos. Os formatos de dados utilizados, por exemplo, não são os mais adequados, são não estruturados ou semiestruturados, trazendo uma série de impedimentos para o acesso e posterior manipulação da informação.

O portal da LAI publica, na maioria dos casos, informações em páginas HTML e algumas vezes direciona o usuário para outros portais, onde estão disponíveis mais páginas HTML e planilhas eletrônicas nos formatos XLS e PDF. Esses formatos servem

---

<sup>1</sup> [http://www.planalto.gov.br/ccivil\\_03/ Ato2011-2014/2011/Lei/L12527.htm](http://www.planalto.gov.br/ccivil_03/ Ato2011-2014/2011/Lei/L12527.htm)

<sup>2</sup> <http://www2.transparencia.pe.gov.br/web/portal-da-transparencia/85>

<sup>3</sup> <http://www.lai.pe.gov.br>

muito bem para a apresentação dos dados para as pessoas, que utilizam programas como navegadores web e ferramentas de escritório para visualiza-los. Porém, esta abordagem dificulta bastante o desenvolvimento de programas para processar os dados, integrar dados de diferentes fontes e extrair informações úteis.

Outra maneira muito comum de disponibilizar dados na web, esta sim destinada à comunicação entre sistemas web, são os web services. Estes podem ser vistos como uma alternativa de comunicação e compartilhamento de dados, pois são serviços web independentes de plataforma ao qual um sistema pode requisitar um recurso ou dado e retornará uma resposta conforme o esquema especificado em sua interface. Entretanto, possuem interfaces de acesso (API's) variadas, e que muitas vezes não seguem um padrão oficial, o que dificulta a recuperação de dados de forma dinâmica e a ligação destes dados com o resto da web.

Com isso, cresceu a necessidade do compartilhamento de dados abertos ligados e estruturados. Tim Berners-Lee propôs, então, a Web Semântica e os princípios de Linked Data. Conceitos que prometem dar início a uma nova geração da web, com dados abertos, mais estruturados e mais interligados do que a web tradicional [2] [7]. Tais atributos fazem com que qualquer programador possa aproveitá-los para desenvolver aplicações que ajudam as pessoas a melhor entenderem ou enxergarem as informações contidas nos dados.

Unindo conceitos e tecnologias relacionadas a governo eletrônico, dados abertos, Web Semântica, Linked Data, IA, entre outros é possível conceber maneiras de melhorar a gestão pública. Ferramentas que facilitam a tarefa de fiscalização e acompanhamento, que tem a população devem ser desenvolvidas. Finalmente, este trabalho busca melhorar a publicação dos dados governamentais, sendo assim, a transparência pública seu maior motivador.

## **1.2 Proposta**

Este trabalho propõe uma solução para a publicação dos dados relativos às despesas do Governo do Estado de Pernambuco (GovPE) seguindo conceitos e padrões provenientes da Web Semântica e de Linked Data. Atualmente, os dados das despesas do governo estão acessíveis por meio de alguns web services do Portal da

Transparência<sup>4</sup> do Governo de Pernambuco. No entanto, o Portal da LAI do GovPE foi o ponto escolhido para a publicação dos dados das despesas do Governo em um formato que permite um melhor reuso, o processamento e a interligação com outras fontes de dados. Os conceitos introduzidos por Tim Berners-Lee, com seus modelos e práticas, aparecem como uma ótima alternativa para a disponibilização dos dados na web, inclusive os governamentais: globais, públicos, formato aberto, reusáveis e estruturados.

De acordo com os princípios de Linked Data, os dados devem ser publicados, preferencialmente em RDF. E é isto que objetiva a ferramenta prevista neste projeto. A ideia é desenvolver uma ferramenta que permite os gestores das instituições do GovPE publicarem, com esforço mínimo, dados sobre os gastos relacionados à sua instituição em formato RDF.

Com base na estrutura dos dados das despesas do governo, foi criada uma ontologia para a definição do modelo semântico dos dados que serão publicados. A ferramenta proposta faz uso de web services para recuperação dos dados das despesas do governo e faz uso da ontologia para realizar a geração dos dados em RDF. Este processo é realizado para cada instituição do governo cadastrada no sistema separadamente, e os dados RDF serão publicados na página relativa à respectiva instituição.

O intuito deste trabalho é trazer mais transparência à gestão pública do Estado de Pernambuco e colocá-lo junto aos governos pioneiros no que diz respeito à abertura dos dados e ao acesso à informação. Reino Unido, Estados Unidos, Irlanda, Austrália, Nova Zelândia e inclusive o próprio Brasil, em esfera federal, são alguns dos que vem reunindo esforços na abertura dos dados e seguindo em maior ou menor grau as recomendações propostas por Berners-Lee.

### **1.3 Organização do Documento**

Esta seção descreve com brevidade a organização deste documento. Este capítulo introduziu o relatório do projeto, expondo o que o motivou e sua proposta. Nos próximos capítulos serão apresentados: a fundamentação teórica, a ferramenta GovPEX3, e, finalmente, a conclusão do trabalho.

O segundo capítulo procura apresentar ao leitor a base teórica que permeia este trabalho. Web Semântica é abordada na primeira parte do capítulo. Em seguida, Linked

---

<sup>4</sup> <http://www.transparencia.pe.gov.br>

Data será tema de discussão terá seus princípios explicados. Terminando o capítulo, serão apresentados conceitos e convenções relacionados a dados abertos e dados governamentais abertos.

O capítulo três expõe detalhes sobre as diferentes etapas do desenvolvimento da ferramenta proposta, a GovPEx3, e dos artefatos necessários. Na primeira seção do capítulo, é feita a descrição do Portal da Lei de Acesso à Informação, que serviu de plataforma para a implementação da ferramenta. A segunda parte trata da criação da Ontologia Despesas GovPE, que se fez necessária para publicação das triplas em formato aberto e estruturado. O capítulo 3 fala, por último, da arquitetura e do desenvolvimento da ferramenta propriamente dita.

Finalizando o relatório, o capítulo 4 apresenta as conclusões que foram feitas com o desenvolvimento do projeto, enumera as contribuições realizadas e faz uma avaliação dos resultados obtidos, propondo melhorias e possíveis trabalhos futuros.

## 2. Fundamentação Teórica

Neste capítulo serão abordados alguns temas relacionados à Web Semântica, Linked Data (dados interligados), e dados abertos. O início do capítulo trata de Web Semântica, onde são descritas as teorias, definições e tecnologias referentes a este conceito. Logo após, Linked Data será o tema abordado, os princípios de Linked Data são citados e, em seguida, explicados de forma mais detalhada. Na seção seguinte os dados abertos e dados abertos governamentais são o tema de discussão. Com esta fundamentação teórica fica mais fácil entender o porquê dos métodos empregados na publicação de dados abertos governamentais.

### 2.1 Web Semântica

Quando a internet se popularizou, na década de noventa, a web já começou a despontar como um grande repositório global de informações. Na sua fase inicial, chamada por alguns de Web 1.0, apenas webmasters e pessoas com conhecimentos em tecnologia conseguiam publicar conteúdos, que eram sempre estáticos. Essa geração teve como alguns dos principais atores grandes portais de notícias, sem muita interação com os usuários, que eram apenas leitores passivos.

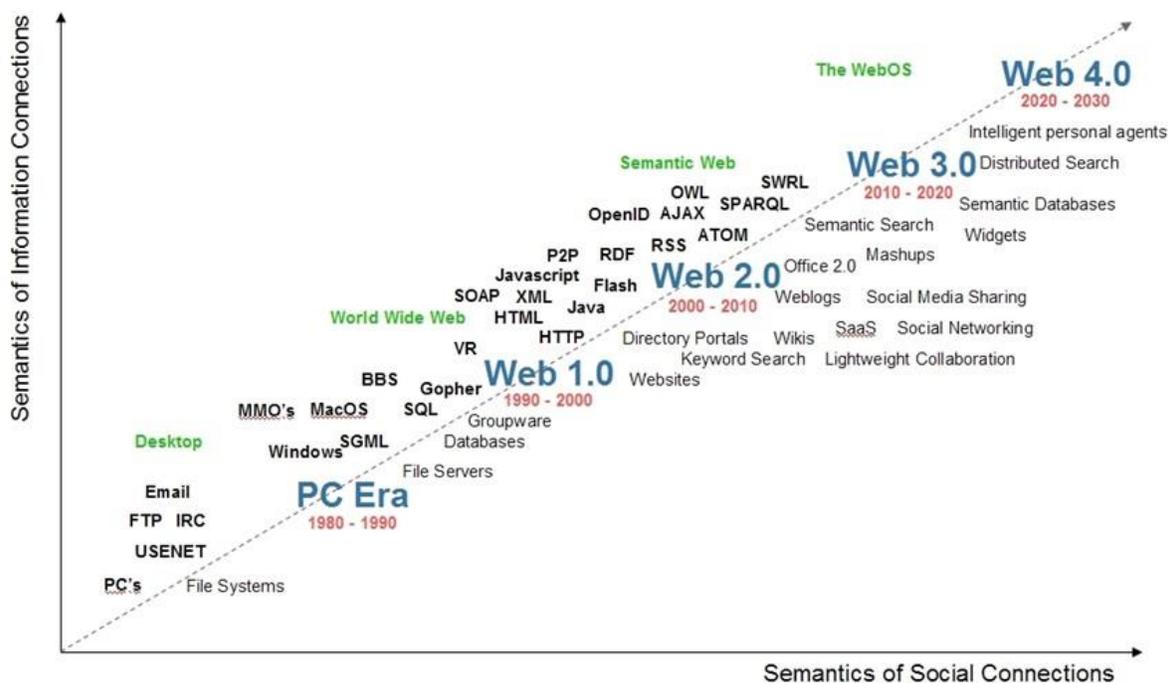
A segunda fase da web, ou Web 2.0 veio com o surgimento de vários novos conceitos e práticas para construção de aplicativos web. O usuário passou a ter um papel mais ativo, plataformas colaborativas viraram tendência, e conceitos como web como plataforma e inteligência coletiva mudaram a maneira de interagir dos usuários e os métodos de desenvolvimento dos engenheiros de software [17] [18].

Agora uma nova mudança já está em curso. A terceira fase da web terá como suporte a chamada Web Semântica. Esta pode ser vista como uma extensão da web tradicional, onde a grande diferença está na preocupação em representar os conteúdos semanticamente. Isto é, os dados passam a ter significados associados a eles, e não apenas uma representação sintática usada na exibição para o usuário. A semântica associada aos dados deve ser formalmente representada, possibilitando o processamento destes pelo computador, fazendo da web algo como uma plataforma de inteligência artificial (IA) distribuída [1].

Na figura 2.1 aparece um gráfico onde é mostrada a linha de evolução da web, linha pontilhada do gráfico, relacionando a semântica das conexões sociais (eixo horizontal), e a semântica das conexões das informações (eixo vertical do gráfico). A

semântica das conexões sociais diz respeito aos ambientes informacionais criados nas diferentes fases da web. Associados a este eixo são listados abaixo da linha pontilhada conceitos e aplicações web, que foram criados a partir da evolução da semântica das conexões das informações, eixo transversal. Este último pode ser visto como a infraestrutura tecnológica que dá suporte a cada fase da web. A Web Semântica aparece como uma dessas infraestruturas, entre a web 2.0 e a 3.0, que busca oferecer suporte para a criação de novos ambientes, mais inteligentes e sensíveis ao contexto, o que configuraria a web 3.0 [15].

Figura 2.1: linha evolutiva da web.



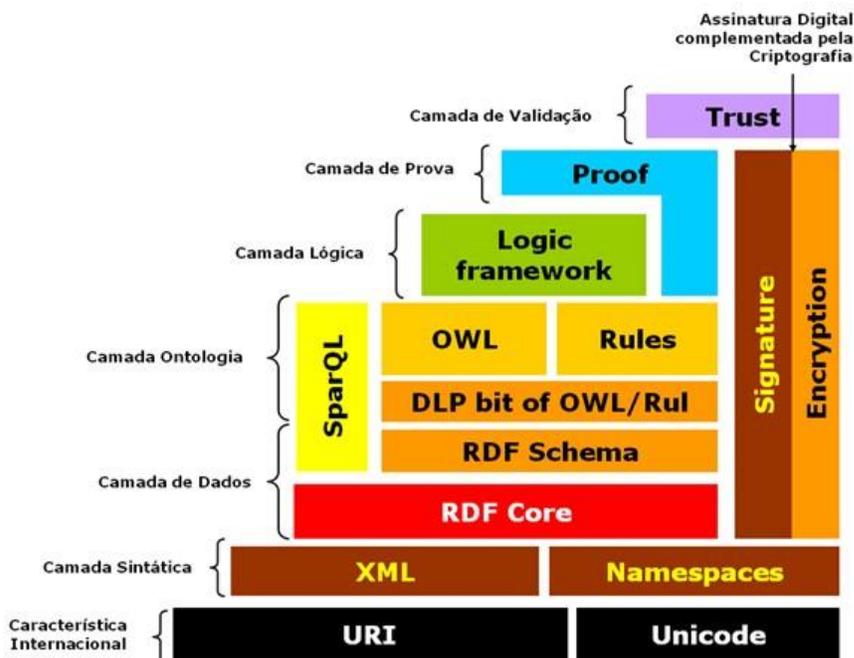
Fonte: SANTOS e ALVES, 2009.

As possibilidades de aplicações na Web Semântica são muitas e animadoras. A busca de informações na web, por exemplo, é tradicionalmente realizada levando-se em consideração apenas o texto da consulta. Já em uma busca na Web Semântica é possível contextualizar muito mais as consultas e fazer requisições muito mais específicas, como por exemplo: “a cidade onde nasce o rio mais longo que está completamente em território brasileiro”. Neste caso, em um sistema semântico hipotético, as instâncias da entidade “rio” seriam varridas e seria retornado o valor da propriedade “temCidadeNascente” daquela instância com maior valor associada à propriedade “temComprimento” e que

tivesse como país associado, através da propriedade “passaPor”, apenas o Brasil. Além disso, a integração de dados provenientes de diferentes bases de dados pode ser feita automaticamente nesta plataforma. Aplicações que integram dados dessa forma são frequentemente chamadas de mashups. É através da semântica atribuída aos dados que um sistema pode ser capaz de distinguir o significado de entidades, às vezes de uma mesma classe, mas que representam propriedades diferentes de um objeto, como por exemplo, os locais da nascente e da foz de um rio [1].

Para que o tratamento semântico dos dados seja possível e as novas aplicações e ambientes possam aproveitar o potencial teórico da Web Semântica, uma série de tecnologias é utilizada. Para solucionar a integração das novas tecnologias que dão suporte ao processamento semântico, uma pilha de protocolos e plataformas deve ser construída em cima da web tradicional. Na figura 2.2 é possível conferir como foi pensada a arquitetura em camadas da Web Semântica. Como acontece na web comum, cada camada foi pensada para trabalhar complementando a camada inferior e de forma independente das superiores [11] [15].

Figura 2.2: arquitetura em camadas da Web Semântica.



Fonte: SANTOS e ALVES, 2009.

A camada mais abaixo está relacionada ao caráter global da web. Unicode é uma codificação padrão de caracteres que permite a manipulação consistente de cadeias de

caracteres provenientes da grande maioria dos sistemas de escrita que existem. URI, *Uniform Resource Identifier*, trata-se de um padrão de identificação de recursos que já é utilizado na web tradicional. Um dos tipos de URI mais conhecidos é o URL, *Uniform Resource Locator*, que é um nome ou identificador que representa um endereço único e global de acesso a um recurso na web. URI utiliza espaços de nomenclatura para identificar o domínio ao qual pertence o recurso, evitando a repetição de nomes. É bastante útil para identificar recursos em ambientes intrinsecamente descentralizados, como é o caso da web [6] [14].

A camada sintática opera com a linguagem eXtensible Markup Language, ou XML, que permite a criação e validação de esquemas de etiquetas de marcação. Isso permite que os dados sejam etiquetados de acordo com o contexto da aplicação, com sintaxe mais elaborada do que a oferecida pela tradicional linguagem de marcação da web, HTML. Namespaces são os espaços de nomenclatura, onde termos e entidades podem ser publicados e identificados de forma única através de um URI. Dois termos ou entidades poderão ter nomes locais iguais desde que estejam em namespaces diferentes, e podem ou não carregar valor semântico equivalente [6].

Na camada de dados, as entidades deverão ser representadas formalmente de uma maneira bem definida, podendo ser identificados seus atributos e relacionamentos com outras entidades. Um exemplo de modelo para representação de dados é o modelo relacional, mas, neste caso, não funcionaria, pois prevê questões como normalização (eliminação de redundância de dados) e verificações de consistência dos dados, que não são compatíveis com o caráter distribuído da web. No contexto da web, RDF é a tecnologia empregada na estruturação dos dados.

Uma questão importante é que, assim como a disposição dos dados, a representação do conhecimento se dá de forma distribuída, diferentemente dos sistemas tradicionais de IA. Nesta formalização descentralizada dos diferentes domínios do conhecimento, a Web Semântica faz uso de ontologias, compondo a sua quarta camada [1].

A camada de lógica é responsável pelo tratamento das informações advindas das camadas inferiores, fazendo as inferências lógicas de acordo com as regras declaradas para o modelo de dados. As camadas de prova e de validação realizam comprovações de coerência lógica e de aspectos semânticos dos dados, bem como avaliação da representação dos dados para prover métricas de confiabilidade, respectivamente [14].

Existem tecnologias e conceitos importantes para funcionamento da Web Semântica, mas que não serão abordados com profundidade por não serem tão relevantes no contexto deste trabalho. Algumas tecnologias que aparecem na pilha da figura 2.2 são alguns exemplos disto. RDF e ontologias, no entanto, são os conceitos mais relevantes das camadas de dados e de ontologia, respectivamente, e são fundamentais para este projeto, por isso, serão detalhados a seguir.

### 2.1.1 RDF

RDF é sigla para *Resource Description Framework*, que em português significa plataforma de descrição de recursos. Como o nome sugere, esta tecnologia especifica um modelo de descrição de recursos. Um recurso pode ser qualquer tipo de entidade, ou conceito: um rio, uma cidade, uma equipe de futebol, uma imagem digital, uma pessoa, os conceitos ou classes “Cidade” e “Pessoa”, por exemplo. Dados em RDF podem ser vistos, também, como metadados, isto é, dados que fornecem informações sobre dados, termo bastante frequente neste campo de estudo.

A maneira como RDF descreve os dados é bem simples, utiliza triplas no formato: <sujeito> <predicado> <objeto>. Este formato é análogo a sentenças simples da linguagem natural. A frase “Madrid é a capital da Espanha” é um exemplo desta analogia, onde ocorre a estruturação da informação em formato de tripla: “Madrid” é um recurso que nesta tripla representa o sujeito, “é a capital de” é o predicado, e “Espanha” é outro recurso, que representa o objeto desta tripla. O predicado contém sempre uma propriedade que tem um domínio e um *range*, conceito análogo a uma função matemática com seu domínio e sua imagem. No caso, “Madrid” integra o domínio da propriedade “é a capital de” e está ligado à “Espanha” através desta propriedade [4] [7].

Para representar dados simples, são utilizados valores literais de tipos primitivos, já para recursos mais complexos, RDF utiliza identificadores. No contexto da web, URIs globais que podem ser reutilizados. Por isso, as triplas RDF podem ser de dois tipos, literais ou RDF link (interligação RDF). A tripla literal é a que tem como objeto um literal de um tipo primitivo, como, por exemplo, uma cadeia de caracteres (*string*), números inteiros (*integer*) ou números decimais (*double*). Já as triplas do tipo RDF link são as que enlaçam identificadores de dois recursos mais complexos, que podem estar em diferentes bases de dados ou não. A criação de um URI para identificar um recurso possibilita sua reutilização em toda web. A decisão de como referenciar o recurso, se de forma literal ou

através de um URI vai depender da aplicação e da possibilidade de reutilizar identificadores de recursos já descritos na web.

RDF estrutura as informações em formato de grafo. O sujeito e o objeto das triplas podem ser vistos como nós de grafos, sejam URIs ou literais, e as arestas dos grafos podem ser representadas pelo predicado das triplas: ligam o sujeito ao predicado. As arestas, assim como as propriedades que compõem os predicados das triplas RDF, são direcionais (saindo do sujeito e indo para objeto) e possuem um tipo: o URI que representa a propriedade. Tal fato tem implicações importantes, pois o grafo é uma estrutura de dados amplamente estudada em Ciência da Computação. Uma característica dos grafos é que permitem que se tratem dados contidos nos nós e se percorram suas arestas, descobrindo caminhos até novos dados relacionados. Isso é fundamental para Linked Data, conceito que será discutido mais a frente neste trabalho.

É bom deixar claro que RDF é um modelo conceitual de descrição de dados por isso não está associado a uma sintaxe específica. Existem várias linguagens de serialização de RDF, ou seja, linguagens que possuem uma sintaxe para definição de triplas RDF. Algumas delas são: RDF/XML, baseado em XML (eXtensible Markup Language), RDFa, linguagem elaborada para inserção de triplas RDF dentro de documentos HTML, Turtle (Terse RDF Triple Language), um formato simplificado, e N-Triples, subconjunto do Turtle bastante simplificado, mas que oferece muita redundância [7].

### **2.1.2 Ontologias**

Na filosofia, ontologia é um campo do conhecimento que estuda a natureza da existência das coisas, os tipos de coisas que existem e suas condições de existência. No entanto, a Ciência da Computação adotou o termo para se referir a uma representação formal de um domínio do conhecimento humano, de modo que um programa de computador possa identificar os conceitos presentes no domínio descrito e fazer inferências, obedecendo às regras de relacionamento entre as entidades estabelecidas pela ontologia. As entidades e regras contidas em uma ontologia são definidas de acordo com o senso comum de um grupo social, sendo possível assim, o compartilhamento desta entre sistemas utilizados por este grupo [4] [6].

Voltando ao exemplo que foi dado anteriormente, onde deseja-se consultar “a cidade onde nasce o rio mais longo que está completamente em território brasileiro”, a ontologia é responsável pela definição da classe “rio”, assim como pela especificação de

que esta classe é domínio das propriedades “temCidadeNascente”, “temComprimento” e “passaPor”. É por meio do uso de ontologias que se diferenciam entidades com mesmo nome, mas com significados diferentes. Por exemplo, a cidade “Barcelona” e o time de futebol com mesmo nome são duas entidades diferentes que podem ser distinguidas por serem instâncias de classes diferentes, a primeira de uma possível classe chamada “Cidade”, e a última de outra hipoteticamente denominada “TimeFutebol”.

Na web muitas ontologias são usadas e qualquer um pode construir e publicar uma, sendo possível e recomendada a reutilização de termos de outras ontologias pré-existentes. Isso faz com que surjam ontologias que definem termos que já são definidos por outras ontologias. Apesar dos termos terem o mesmo significado, possuirão identificadores diferentes, pois os termos, da mesma forma que os dados, também possuem identificadores únicos na web. Assim, para que os programas possam integrar dados que pertencem a uma mesma classe, mas que foram descritos por ontologias diferentes, existem termos de equivalência, que são usados para informar que certo termo de uma ontologia é equivalente a outro termo de outra ontologia [1].

Para definir uma ontologia existem muitas linguagens, mas na Web Semântica, que trabalha com o modelo de dados RDF, são utilizadas RDFS<sup>5</sup> (*RDF Schema*) e OWL<sup>6</sup> (*Web Ontology Language*). Estas linguagens estendem RDF, oferecendo meios para descrever conceitos dos domínios do conhecimento e não apenas instâncias. Ambas são recomendadas pelo W3C (Consórcio World Wide Web), principal órgão de padronização de tecnologias da web, que é liderado pelo próprio criador da web Tim Berners-Lee.

RDFS é a mais básica destas linguagens e permite a definição de esquemas de hierarquias simples de conceitos e propriedades. Já OWL possui muito mais recursos para definição de classes, propriedades e relacionamentos. Com OWL é possível declarar que duas classes são disjuntas e que uma propriedade é transitiva, simétrica, funcional ou se é a inversa de outra propriedade, exemplos de definições que não são suportadas por RDFS.

Ontologias são comumente chamadas de vocabulários RDF neste âmbito de estudos. Existem muitos vocabulários que já estão sendo amplamente utilizados e alguns são recomendados oficialmente pela W3C. *The RDF Data Cube Vocabulary*<sup>7</sup>, ontologia destinada à publicação de dados multidimensionais e estatísticos, *Data Catalog*

---

<sup>5</sup> <http://www.w3.org/TR/rdf-schema>

<sup>6</sup> <http://www.w3.org/TR/owl-features>

<sup>7</sup> <http://www.w3.org/TR/vocab-data-cube>

*Vocabulary*<sup>8</sup> (DCAT), projetada para facilitar a interoperabilidade entre catálogos de dados na web, e *An organization ontology*<sup>9</sup>, que define um vocabulário para descrição de estruturas organizacionais, são exemplos de vocabulários mantidos pelo próprio W3C. O vocabulário FOAF<sup>10</sup> (o *Friend Of A Friend*) pode ser utilizado para descrever dados pessoais e os relacionamentos interpessoais, sendo muito difundido entre os publicadores e pesquisadores de Web Semântica e Linked Data.

Ao longo da camada de ontologia da pilha de tecnologias da Web Semântica aparece SPARQL (*SPARQL Protocol and RDF Query Language*), uma tecnologia fundamental que também é recomendação oficial do W3C. SPARQL é um protocolo e uma linguagem de consulta de dados RDF que tem sintaxe parecida com a de SQL, linguagem de consulta padrão de bases de dados relacionais. Neste contexto, um endpoint SPARQL é um serviço web que está associado a repositório de triplas RDF e ao qual, consultas são submetidas e retornam um conjunto de triplas. Um mecanismo muito interessante desta linguagem é a consulta federada, que pode ser distribuída em vários endpoints associados a conjuntos de dados diferentes. Para cada endpoint a consulta é computada e então os resultados são reunidos em um único conjunto de triplas, que é retornado [4].

## 2.3 Linked Data

Linked Data (dados interligados) é um termo criado por Tim Berners-Lee e se refere a uma série de práticas, consideradas as melhores para a publicação e interligação de dados na web. De forma mais precisa, Linked Data pode ser definido como: o conjunto de princípios, técnicas e padrões que permite a publicação, recuperação e interligação de dados estruturados de forma padronizada e global. Tem como objetivo a criação de um grande, único grafo global de dados, além de oferecer suporte à Web dos Dados (ou Web 3.0). Este grafo, formado por triplas RDF, já é uma realidade e ano a ano novos conjuntos de dados são adicionados (interligados) a ele [2] [7]. As figuras 2.3 e 2.4 revelam este crescimento.

---

<sup>8</sup> <http://www.w3.org/TR/vocab-dcat>

<sup>9</sup> <http://www.w3.org/TR/vocab-org>

<sup>10</sup> <http://xmlns.com/foaf/spec>



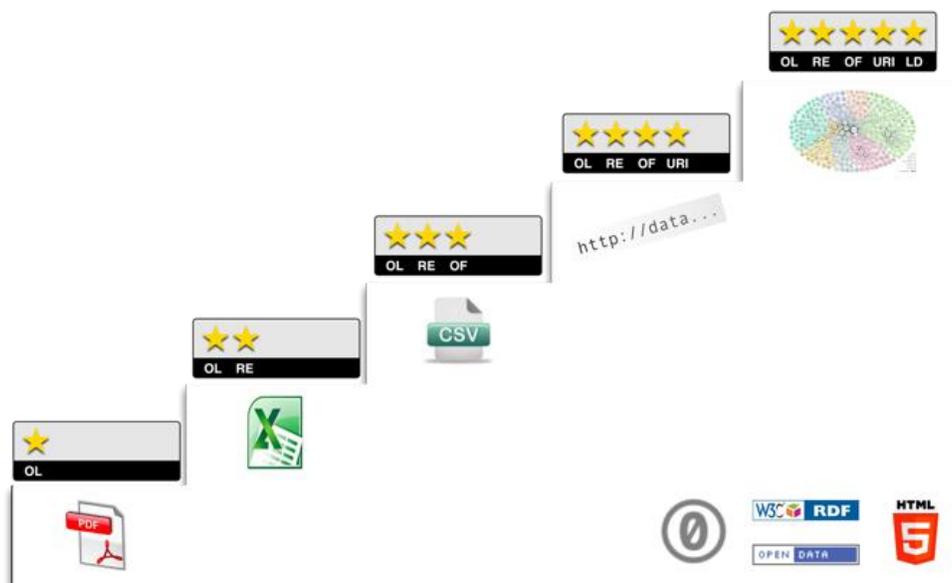
O site *The Linking Open Data cloud diagram* periodicamente publica diagramas que mostram o estado momentâneo do grafo (ou nuvem) gerado por dados publicados e interligados usando os princípios de Linked Data. Os diagramas das figuras 2.3 e 2.4 foram gerados, respectivamente, em março de 2007 e setembro de 2011, em intervalo entre eles de dois anos e meio. Os conjuntos de dados aparecem representados por círculos, que são os nós do grafo. Relacionando os dois grafos é possível perceber um grande crescimento no número de dados publicadas. A cor utilizada em cada base representa o domínio ao qual pertence esta base. No diagrama mais novo aparecem dados de sete domínios diferentes: mídia, geográfico, publicações, conteúdos gerados pelos usuários, governo, domínios variados e ciências. Os dados do domínio governamental, tema deste projeto, aparecem em posição de destaque na parte oeste da nuvem em cor verde escuro.

Como forma de padronizar e facilitar a publicação de dados abertos interligados, as práticas de Linked Data foram especificadas por Berners-Lee em uma nota de projeto que discutia questões sobre Web Semântica. Estas práticas acabaram repercutindo bastante nesta área de pesquisa, ficando conhecidas como *princípios de Linked Data*. São quatro e serão detalhados nas subseções a seguir:

- 1) Usar URIs para identificar coisas;
- 2) Usar URIs HTTP para que se possa acessar os recursos;
- 3) Disponibilizar informações úteis utilizando os padrões (RDF e SPARQL, entre outros);
- 4) Interligar recursos com outros URIs para que se descubra mais informações.

Para classificar o nível de comprometimento das bases de dados com relação aos princípios de Linked Data, Berners-Lee criou um esquema de maturidade de publicação de dados baseado em estrelas. As bases de dados publicadas na web em formatos não estruturados (ou não tratáveis por computador) são classificadas com uma estrela, a pior classificação. Já as bases compostas por dados em RDF referenciando outras fontes de dados, seguindo os padrões da W3C e todos os princípios de Linked Data, são consideradas cinco estrelas, a melhor classificação. A figura 2.5 ilustra este esquema de classificação.

Figura 2.5: esquema para classificar bases de dados publicadas na web. Legendas: OL: Open Licence (livre de licenças); RE: machine-REadable (tratável por computador); OF: Open Format (formato não proprietário); URI: identificado por URI; LD: Linked Data (dados interligados).



Fonte: 5 Stars Open Data (<http://5stardata.info/>), acessado em 22 de janeiro de 2013.

### 2.3.1 Usar URIs para identificar coisas

Este princípio estabelece que se deve utilizar URI para identificar única e globalmente os recursos. URI, como foi mostrado anteriormente, é um padrão de identificação utilizado na web que compõe a camada inferior da Web Semântica. Este método de identificação funciona perfeitamente para web, que é plataforma totalmente descentralizada e possui servidores de nomes hierárquicos. Na web, qualquer pessoa que possui um domínio pode criar um URI para identificar alguma coisa, característica fundamental para uma rede que pretende facilitar a publicação de dados dentro de uma estrutura de dados global [2] [7].

Questão fundamental que diferencia a Web Semântica da web tradicional está no **que** pode ser endereçado, ou seja, identificado através de um URI. Na web tradicional os objetos endereçados são sempre recursos digitais, os quais são requisitados e transferidos pela rede. Comumente são documentos HTML e imagens. Já no contexto da Web Semântica e de Linked Data, um recurso pode ser qualquer objeto do mundo real ou conceito abstrato, incluindo os tradicionais recursos digitais e todos devem receber um URI para representá-lo. A entidade identificada continua podendo ser chamada de recurso, a diferença é que não necessariamente este vai ser um recurso digital.

### 2.3.2 Usar URIs HTTP para que se possa acessar os recursos

O protocolo Hypertext Transfer Protocol é utilizado para a devida comunicação entre os atores da web. URI de qualquer tipo de recurso pode ser requisitado por clientes HTTP e os servidores HTTP devem estar preparados para responder. Isto pode gerar dúvidas, pois na web tradicional as resposta às requisições de recursos digitais normalmente, levam o recurso digital requisitado até o cliente. Mas no caso de URIs que identificam objetos do mundo real ou conceitos, o recurso não pode ser enviado pela rede de computadores e muitas vezes não são sequer tangíveis. Dentro deste novo paradigma, URIs de coisas que não são digitais são utilizados como um meio de acesso a algum artefato que descreve esta entidade [2] [7].

Uma distinção deve ser clarificada, o URI que identifica uma entidade do mundo real e o URI que identifica o documento que a descreve são diferentes. Por exemplo, o URI <http://pt.dbpedia.org/resource/Brasil> identifica o país chamado Brasil na base de dados da dbpedia, e é diferente dos URIs <http://pt.dbpedia.org/page/Brasil> e <http://pt.dbpedia.org:8890/data/Brasil.ntriples>. Os dois últimos identificam documentos que descrevem a entidade Brasil, respectivamente nos formatos HTML e N-Triples, uma das linguagens de serialização RDF. A dbpedia é uma grande base de dados RDF criada com base na enciclopédia colaborativa wikipedia.

Para processar as requisições de recursos que não são digitais os servidores HTTP devem estar preparados para isso. Assim, o W3C especifica duas abordagens para tratar estas requisições: 303 URIs e Hash URIs, as duas são descritas a seguir.

#### 2.3.2.1 303 URIs

Nesta abordagem, quando um URI de recurso não digital é requisitado via HTTP, o servidor retorna URIs que identificam documentos que descrevem a entidade requisitada pelo cliente. O processo acontece em quatro etapas:

- 1) Cliente envia HTTP GET com URI da entidade da qual se deseja obter informações. Nesta requisição o cliente também envia o cabeçalho HTTP *accept* informando os formatos de documento que aceita receber.
- 2) Servidor responde com código 303 que corresponde à mensagem *see other*, e com o URI do documento descritivo da entidade.
- 3) Cliente faz nova requisição, desta vez direcionada para o documento descritivo, utilizando URI recebido do servidor.

4) Servidor responde com código 200 e com o documento descritivo.

Este processo tem a desvantagem de exigir do cliente duas requisições HTTP [7]. Em compensação, o documento que descreve a entidade geralmente é pequeno, já que descreve apenas a entidade requisitada. Veremos que isto ocorre de forma diferente no método Hash URI.

### **2.3.2.2 Hash URIs**

Este método aproveita o fato de o protocolo HTTP ignorar a parte do URI que vem depois do caractere '#'. Assim, constrói-se o URI de um recurso de acordo com a seguinte regra: <namespace>#<nome local>. Então, no momento de fazer uma requisição por este URI, o HTTP ignora a cadeia de caracteres que vem após o '#', no caso o nome local da entidade. Dessa forma, o servidor recebe a requisição e responde com um arquivo que descreve o conjunto de entidades que compartilha o namespace requisitado. Ao receber o arquivo, o cliente seleciona a descrição da entidade de interesse através do seu nome local (parte do URI que foi ignorada na hora da requisição HTTP) [7].

Hash URIs é um método recomendado para recuperação de descrições de vocabulários, quando é interessante para o cliente obter e guardar as descrições de todos os termos e entidades do vocabulário. Esta técnica requer apenas uma requisição HTTP e não duas como em 303 URIs. Por outro lado, o documento, já que pode conter a descrição de muitas entidades, costuma ser maior, podendo fazer com que sua transferência pela rede tome muito mais tempo. Assim, uma das duas técnicas pode ser mais adequada, dependendo do contexto da consulta [7].

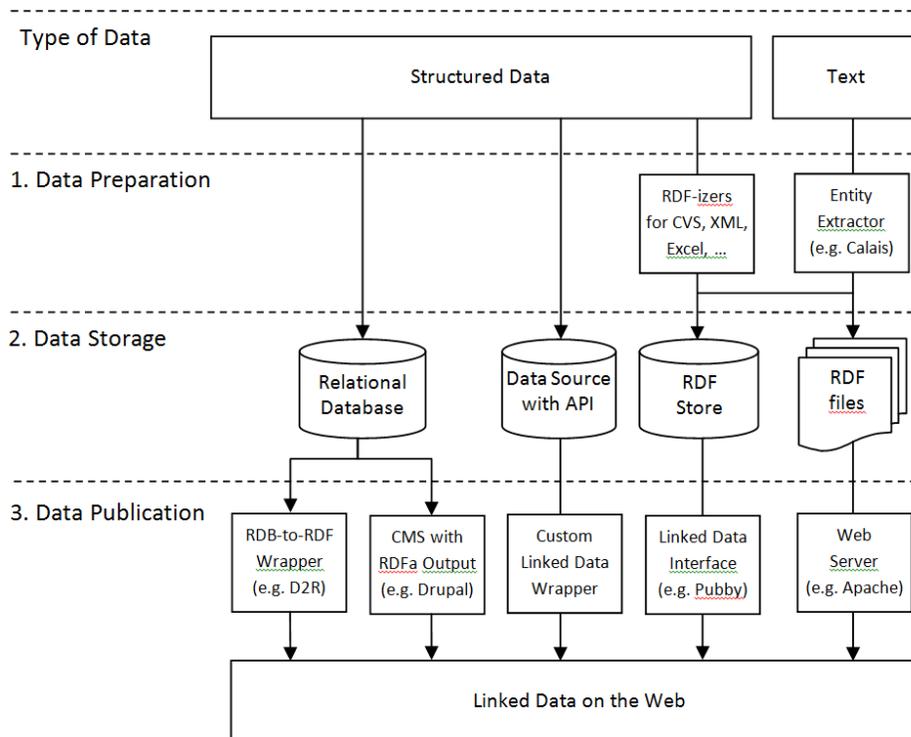
### **2.3.3 Disponibilizar informações úteis utilizando os padrões**

Este princípio recomenda que os projetistas descrevam todos seus dados em forma de triplas RDF, respeitando a semântica original. É fundamental que se represente, sempre que possível, as relações entre os dados, obtendo assim um grafo, ou conjunto de triplas RDF, bastante interligado. Existem várias formas de obter triplas RDF a partir de dados em outros formatos, a estratégia escolhida dependerá do formato de dados de origem [2] [7].

A figura 2.6 mostra o diagrama de transformação de dados em diferentes formatos em triplas RDF. Esta tarefa é algumas vezes chamada de ETL (*Extract, Transform and Load*), ou extrair, transformar e carregar. O processo ETL que converte bases de dados

relacionais em conjuntos de triplas RDF é bastante relevante, uma vez que existe uma enorme quantidade de dados que utiliza este modelo representação. Por isso, o W3C especifica e recomenda RDB2RDF, um conjunto de duas tecnologias para mapeamento de bases de dados relacionais para RDF. Após os mapeamentos necessários é preciso, adotar um formato de serialização, que nada mais é que uma sintaxe para descrever os dados RDF. Idealmente, endpoints SPARQL associados aos repositórios RDF devem ser configurados para fornecer suporte a consultas [7].

**Figura 2.6: diagrama de transformação de dados em RDF (ETL).**



Fonte: HEALTH e BIZER, 2011.

### 2.3.4 Interligar recursos com outros URIs para que se descubra mais informações

Finalmente, o quarto princípio de Linked Data recomenda que enlaces sejam criados entre os URIs, principalmente entre identificadores provenientes de fontes dados diferentes. Os chamados links externos são os que fazem este tipo de interligação, são triplas RDF na qual o sujeito é um URI de uma base de dados e o objeto um URI de outra base de dados mantida por um servidor externo [7].

Este princípio é fundamental para a criação da Web dos Dados, como também é chamada a Web 3.0, pois converte ilhas de dados em um conjunto global de dados, um grande grafo conexo que interliga os dados da web, de maneira que as aplicações podem navegar livremente por este grafo de dados. Dessa forma é possível chegar, a partir de um conjunto inicial de triplas, a URIs de entidades externas. A descoberta automática de novas fontes de dados é um dos principais trunfos deste novo paradigma, sendo possível, entre outras coisas, a criação de *crawlers*, robôs da web que vasculham dados e fazem indexações, e motores de busca diferentes dos da web tradicional, baseados em dados com significado e não em documentos contendo informações desestruturadas. As aplicações que podem ser construídas em cima desta estrutura são inúmeras e são elas que vão determinar a nova era da web, a web 3.0 ou web dos dados.

É interessante distinguir os diferentes tipos de links, ou interligações RDF. Existem três tipos importantes que podem ser frequentemente encontrados no Linked Data, são eles: link de relacionamento, link identidade e link de vocabulário. Serão expostas as definições e a utilidade de cada um deles a seguir.

- **Links de relacionamento**

Estas triplas relacionam entidades, que podem ou não pertencer à mesma fonte de dados. É o tipo de RDF link mais comum. É interessante observar que, diferentemente dos links entre documentos da web comum, os links de relacionamento do Linked Data possuem um tipo. O tipo é indicado justamente pelo predicado da tripla RDF, um URI que identifica um termo de um vocabulário RDF que geralmente se refere a uma propriedade que tem domínio composto por classes definidas por este vocabulário [7].

- **Links identidade**

As triplas que possuem link identidade são as que relacionam dois URIs diferentes que supostamente se referem a mesma entidade do mundo real. Na web, qualquer pessoa pode publicar dados e criar URIs para representar entidades, o que faz com que entidades possam ser referenciadas por vários URIs na web, cada um dentro de um espaço de nomenclatura (namespace) diferente. Portanto, se recomenda que, quando a pessoa que gerencia uma fonte de dados RDF identifique que um URI de sua base se refere ao mesmo objeto ou conceito que outro URI conhecido, adicione triplas RDF descrevendo a equivalência. Isso é realizado utilizando links identidades entre as

entidades [7]. Na construção das triplas que fazem interligações do tipo link identidade, a propriedade mais utilizada no é o “owl:sameAs”, termo definido pelo W3C na linguagem OWL e que possui o seguinte URI: <http://www.w3.org/2002/07/owl#sameAs> [3].

Devem-se usar links identidades sempre que forem identificadas equivalências semânticas. Porém, este tipo de redundância deve ser evitada no momento da criação dos URIs, fazendo-se uma verificação da existência de URIs que identificam o recurso no grafo de dados RDF da web. O fato de qualquer um poder criar um URI que identifique qualquer coisa tem vários benefícios. Primeiramente, faz com que a informação possa ser rastreada, ou seja, saber quem está publicando sobre um determinado assunto, pois o URI vai apontar para o domínio e respectivo namespace do publicador. Em segundo lugar isso permite a emissão de diferentes pontos de vista sobre um determinado assunto, dentro de um meio democrático, que é a web. E finalmente descentralização da responsabilidade sobre os URIs, o que também é saudável por criar uma rede de dados mais tolerante a falha e indisponibilidade.

- **Links de vocabulário**

Os links de vocabulário tem a função de definir e relacionar termos e conceitos para construção dos vocabulários (ou ontologias). Estes podem expressar equivalência entre classes e propriedades, funcionando como um link identidade, e estabelecer outros tipos de relações. Por exemplo, utilizando RDF Schema, para declarar que uma classe é subconjunto de outra se utiliza a propriedade “rdfs:subClassOf”, e para definir que uma propriedade é subpropriedade de outra o termo utilizado é o “rdfs:subPropertyOf”, onde as propriedades podem pertencer a ontologias distintas ou não [7].

Para evitar redundâncias, aumentar o reuso e conseqüente a interligação dos dados, recomenda-se o uso de vocabulários amplamente difundidos e que só se crie um URI novo para um termo quando não se encontre um URI referente a um termo semanticamente equivalente. Inclusive na criação de novas ontologias, é possível e recomendado reaproveitar termos de outras já publicadas.

## **2.4 Dados Abertos**

Nesta seção discutimos o conceito de Dado Aberto, porém antes de apresentar esta definição, é interessante conceituar alguns termos e apresentar uma classificação dos dados quanto a sua estrutura. Primeiramente, dados primários, os mais interessantes

para este projeto: são os que não tiveram seus formatos (ou estruturas) modificados, mantêm formato bruto e o contexto semântico para o qual ele foi definido. Formato bruto é o que foi especificado pelo analista na definição do modelo de dados e utilizado na geração dos dados. Para manter o formato bruto, o dado não pode ter passado por manipulações que alteraram granularidade, nível de abstração valor ou estrutura [10].

Esclarecendo também as definições das classificações de dados quanto a sua estrutura, temos:

- **Estruturados:** possuem organização que obedece a critérios esquemáticos rígidos e pré-definidos, como certa quantidade de atributos ou campos, tipo de dado para cada campo, limite de tamanho para campos determinados, domínio de dados, restrições, entre outras definições.
- **Semiestruturados:** são organizados de uma forma que não se sabe ao certo todos os campos de uma informação, possuindo um conjunto básico invariável de atributos (esquema), mas podendo ter outros campos adicionados ocasionalmente, de acordo com certas circunstância.
- **Não estruturados:** não possuem definições prévias de esquema, conjunto de campos obrigatórios, tipo de dados, nem de limitações de tamanho, apenas um método de exibição. Textos contidos em documentos HTML, PDF, ODF ou DOC são exemplos disso.

O W3C possui um grupo de interesse de governos eletrônicos, o qual define dado aberto da seguinte forma: aquele que foi publicado em formato bruto, estruturado e de tal forma que seja possível sua reutilização total por terceiros. Uma maneira de implementar dados abertos é justamente utilizando as tecnologias da Web Semântica e Linked Data que são recomendações oficiais do W3C, como URI, RDF e SPARQL [10].

O mesmo grupo de interesse discutiu em 2009 a definição de dados abertos governamentais. A partir destas discussões, chegou-se à convenção de uma série de atributos que os dados governamentais devem ter para serem considerados abertos [10]. São eles:

- 1) **Completo:** todo o conjunto de dados públicos, qualquer informação ou gravação digital armazenada, que não possui limitações de privacidade, privilégios e segurança, deve ser disponibilizado abertamente.
- 2) **Primário:** devem ser publicados dados em formato bruto, respeitando semântica original e estrutura original.

- 3) **Em tempo oportuno:** o dado deve ser publicado tão rápido quanto necessário para preservar seu valor.
- 4) **Acessível:** os dados devem ser disponibilizados de forma que o maior número possível de pessoas tenha acesso e para maior número de propósitos possível.
- 5) **Pode ser processado por um computador:** os dados devem ser suficientemente estruturados para permitir processamento automático.
- 6) **Não discriminatório:** os dados devem estar disponíveis para todos sem que o consumidor precise se cadastrar.
- 7) **Não proprietário:** os dados devem estar em um formato sobre o qual nenhuma entidade tenha controle exclusivo.
- 8) **Livre de licença:** os dados devem estar livres de quaisquer questões jurídicas que possam restringir o uso, como patentes ou direitos autorais.

Convencionou-se ainda, as três “leis” dos dados abertos governamentais que, de certa forma, resumem os oito princípios citados acima [10]:

- 1) Se não for possível indexá-los ou chegar navegando até eles, os dados não existem.
- 2) Se não estão disponíveis em um formato aberto e que pode ser processado por um computador, os dados não se encaixam.
- 3) Se alguma questão jurídica impede que eles sejam reaproveitados, os dados não agregam valor.

Formatos fechados ou proprietários de arquivos possuem amarras legais, pois suas especificações são de propriedade exclusiva de alguma organização, que tem o direito de controlar e limitar o uso e a manipulação de dados em tais formatos por programas de terceiros. Assim, o acesso completo e correto dos dados é realizado apenas por quem tem a especificação completa do formato, que é liberado ou não pela organização proprietária. Por esse motivo, formatos fechados de dados não se encaixam com a filosofia da Web Semântica, nem de Linked Data e de dados governamentais abertos.

## 3 GovPEx3

Neste capítulo, serão apresentados detalhes que concernem o desenvolvimento de uma ferramenta para compor o Portal da Lei de Acesso à Informação do Governo do Estado de Pernambuco. O objetivo desta ferramenta é a publicação dos dados das despesas de cada órgão do Governo de Pernambuco (GovPE) em formato aberto e estruturado, utilizando tecnologias da Web Semântica e seguindo os princípios de Linked Data.

Na primeira seção do capítulo, o Portal da LAI é descrito, uma vez que este serviu de plataforma para o desenvolvimento da ferramenta e para a disponibilização dos dados abertos. A Ontologia Despesas GovPE foi o artefato desenvolvido na primeira etapa de implementação da ferramenta e é tema da segunda seção deste capítulo. Na última parte, detalhes de arquitetura e implementação da ferramenta são expostos.

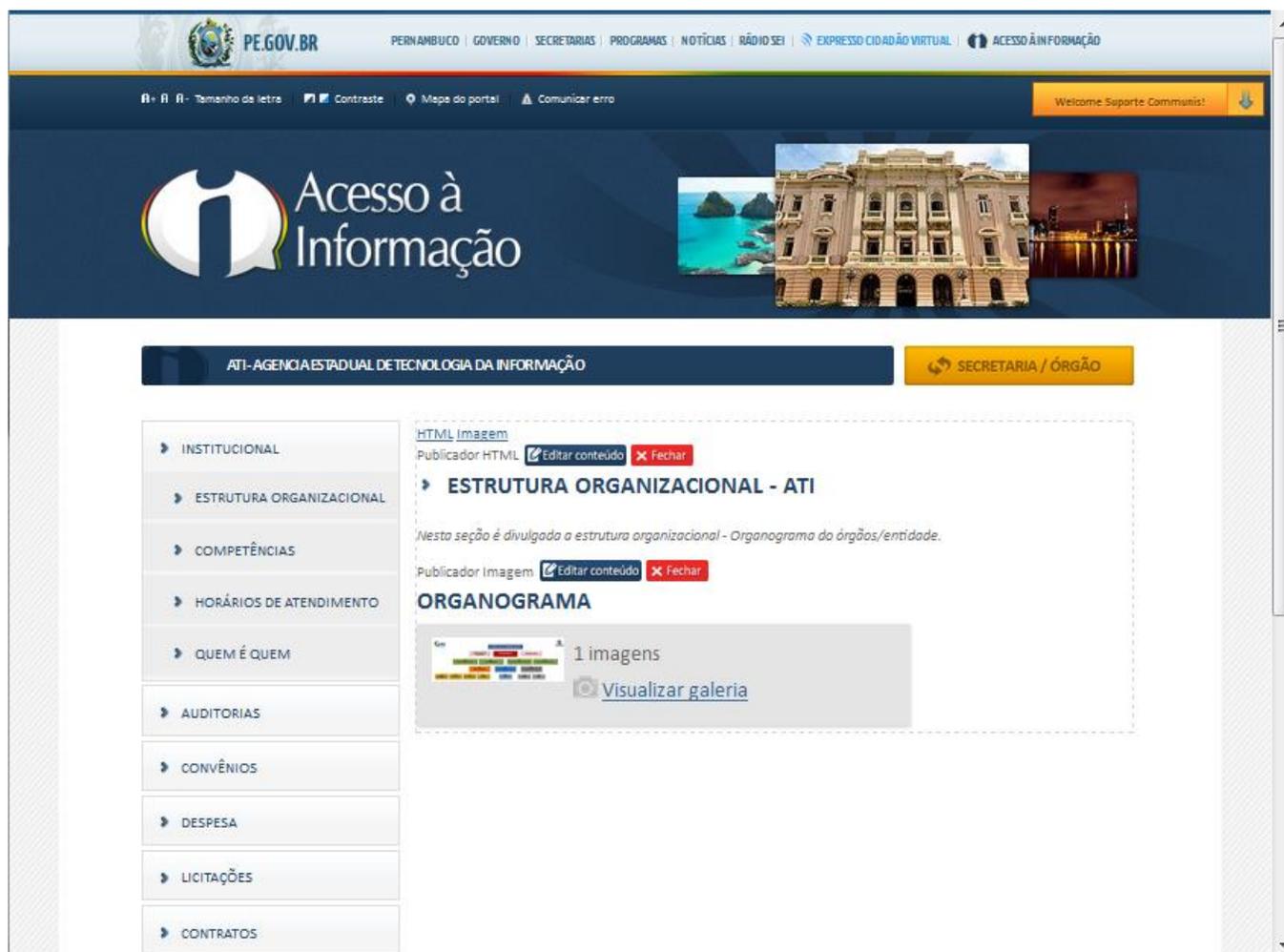
### 3.1 Portal da Lei de Acesso à Informação

O Portal da LAI é uma ferramenta de governo eletrônico desenvolvido utilizando uma versão customizada do Liferay, uma plataforma de construção de portais. Esta plataforma foi desenvolvida na linguagem de programação Java e seu principal tipo de estrutura é o Portlet, um módulo visual independente utilizado para disponibilizar informações dentro de uma página web. O portal entrou no ar no início deste ano, em cumprimento da Lei de Acesso à Informação do Estado de Pernambuco, que entrou em vigor em 1º de janeiro de 2013. Ele visa reunir em um único portal informações sobre todos os órgãos do governo, pois a nova lei obriga o governo a manter canais que garantem o acesso dos cidadãos à informação. Apesar disso, os métodos empregados na disponibilização de dados do portal não obedecem aos padrões da Web Semântica, nem aos princípios de Linked Data e de dados governamentais abertos.

Para ficar claro como ocorre o processo de publicação de dados no portal, é interessante saber como ele está estruturado. O portal é utilizado por três tipos de usuários diferentes: os administradores do portal, os gestores das instituições do GovPE e os usuários comuns. Por isso, o portal implementa perspectivas de visões e permissões diferentes para cada tipo de usuário. Os administradores são os responsáveis por montar a estrutura de *menus* e páginas do portal (exemplificada na figura 3.1), os gestores por publicar os conteúdos personalizados de suas instituições e os usuários comuns são os

cidadãos que acessam o portal para consultar informações sobre as instituições do governo.

**Figura 3.1: página de estrutura organizacional da Agência de Tecnologia da Informação (ATI) no Portal da LAI vista por um usuário logado no sistema (gestor desta instituição ou administrador do portal).**



Fonte: captura de tela do Portal da LAI

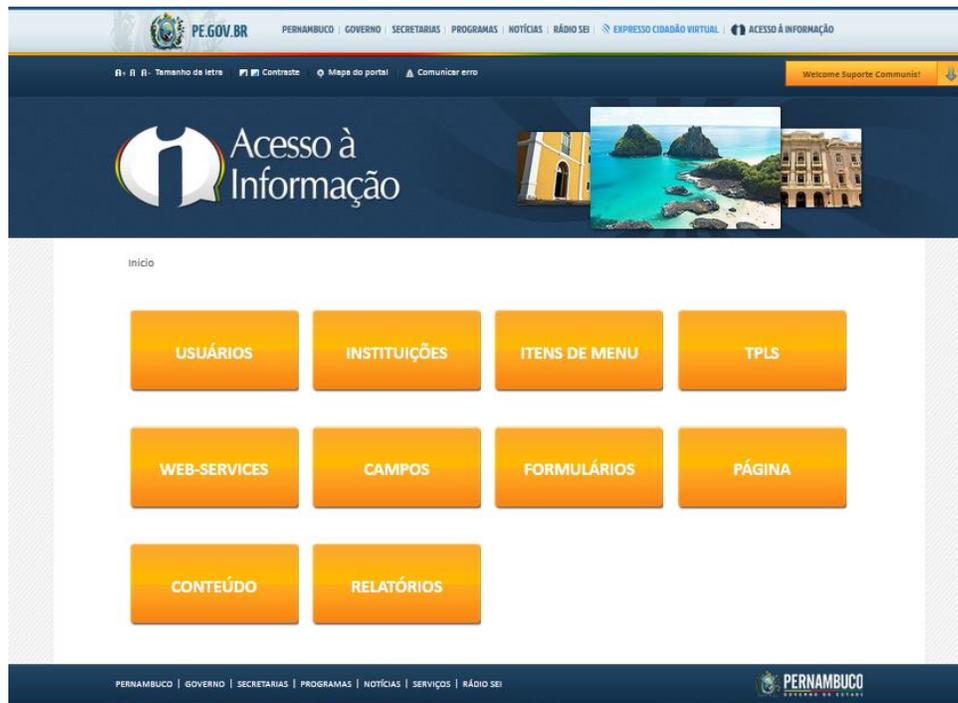
Os administradores são os funcionários da Agência de Tecnologia da Informação (ATI), instituição do GovPE que é responsável pela manutenção do portal. Estes usuários têm a função de montar as páginas destinadas à apresentação de informações das diferentes instituições do governo. Para fazer esta montagem, o administrador utiliza a interface da figura 3.2 e segue os seguintes passos:

- 1) Cadastra as instituições.

- 2) Cadastra os TPLs (TemPLate, significa modelo). A entidade chamada TPL é uma estrutura que organiza o layout da página em áreas retangulares de exibição de conteúdos. A estrutura do TPL é definida escolhendo-se o número de linhas, o número de colunas para cada linha, e a porcentagem de largura que terá cada coluna de cada linha (a soma das porcentagens das colunas de uma linha deve ser 100%). No exemplo da figura 3.1, o TPL escolhido para página tem apenas uma linha, uma coluna com largura 100% e, portanto, apenas uma área de apresentação de informações (delimitada pelas linhas pontilhadas).
- 3) Cadastra páginas, associando a cada uma um TPL.
- 4) Define a estrutura da página, ou seja, escolhe os tipos de informações que poderão ser publicados em cada área da página (o número de áreas da página é definido pelo TPL associado). No caso da página da figura 3.2, os tipos de conteúdos escolhidos para a área única foram “HTML” e “Imagem”. Os tipos possíveis no portal são: “HTML”, “Link”, “Web Service”, “Imagem”, “Documento”, “Conteúdo Padrão” e “Formulário”.
- 5) Cria itens de menus. As páginas e os itens de menus cadastrados podem ser utilizados em uma ou mais instituições.
- 6) Define a estrutura de cada instituição, isto é, monta uma lista de itens de menus associados a páginas do portal ou a páginas externas, formando o menu lateral que pode-se observar do lado esquerdo da página da figura 3.1.

Com a estrutura de *menu* e páginas de cada instituição definida e os tipos de conteúdos escolhidos para cada área de página, tarefas realizadas pelo administrador, os gestores das instituições entram em ação. São eles os responsáveis por alimentar as páginas com as informações dos tipos definidos. O gestor, então, adiciona à página Portlets específicos para cada tipo permitido, ao qual submete, em seguida, informações relativas à sua instituição. A submissão de informações se dá através da parte privada dos Portlets, acessíveis pelo botão “Editar conteúdo” que se pode ver na figura 3.1 (obviamente, o botão aparece apenas para administradores e gestores logados no portal).

**Figura 3.2: página de administração do Portal da LAI.**



**Fonte: captura de tela do Portal da LAI.**

O Portlet chamado “Publicador HTML” está associado ao tipo de conteúdo “HTML” e permite que o gestor publique informações em HTML. Através do Portlet “Publicador Imagem”, associado ao tipo “Imagem”, o gestor publica uma galeria de imagens. Estes dois Portlets estão inseridos na página da figura 3.1. O Portlet “Publicador de Documentos” está associado ao tipo de conteúdo “Documento” e dá suporte à publicação de quaisquer tipos de arquivos, organizados por pastas. O Portlet “Publicador de Links” refere-se ao tipo “Link” e permite a publicação de uma lista de links HTML, cada um com sua URL.

Já o Portlet “Conteúdo Padrão”, que exibe informações do tipo “Conteúdo Padrão”, difere um pouco dos outros porque o gestor tem permissão apenas de inseri-lo na página. Este tipo de informação é um conteúdo em HTML configurado pelo administrador do portal. O Portlet “Publicador Web Service”, que se refere ao tipo “Web Service” do portal, permite que o gestor publique as informações retornadas por um web service escolhido dentre os cadastrados no portal pelo administrador. Finalmente, o Portlet “Publicador Formulário” do portal, associado ao tipo “Formulário”, exibe em HTML informações inseridas pelo gestor em campos de um formulário HTML previamente configurado pelo administrador.

Fica claro que os Portlets que publicam as informações no Portal da LAI, disponibilizam apenas documentos HTML ou arquivos e imagens cadastrados pelos gestores das instituições. Estes formatos, como já foi visto, são não estruturados e não dão suporte semântico nenhum. Portanto, os processos de publicação de dados realizados neste portal não seguem padrões da Web Semântica, nem princípios de Linked Data. Com isso, questões importantes relativas à utilidade dos dados, como processamento automático, reuso e interligação com outras fontes de dados, ficam comprometidas, diminuindo o valor real dos dados.

Neste contexto, foi proposto neste trabalho, a GovPE<sup>3</sup>, uma extensão do Portal da LAI que busca melhorar o processo de publicação de dados das despesas das instituições e, conseqüentemente, aumentar a utilidade dos dados e seu valor agregado para a sociedade.

### **3.2 Desenvolvimento da Ontologia Despesas GovPE**

Na publicação de dados estruturados abertos, para garantir que seja preservada a semântica original das informações, é necessário conhecer a base de dados que se deseja publicar. Conhecer no sentido de saber o que os dados estão representando dentro da área de conhecimento para a qual a base foi criada e seus possíveis formatos de representação. Com isto em mãos, a tarefa seguinte deve ser a busca por ontologias (ou vocabulários) conhecidas que representam as classes e relações das entidades presentes nas informações que se quer publicar. Tal tarefa busca fazer reuso de ontologias, fundamental para interligação dos dados, como prega os princípios de Linked Data. Se não forem encontrados termos de ontologias que definam satisfatoriamente a semântica dos dados a publicar, deve-se modelar uma ontologia que a defina. Posteriormente, as informações devem ser convertidas em triplas RDF através da utilização das classes e propriedades adequadas em um processo de ETL.

No caso da ferramenta proposta por este trabalho, os dados tratam das despesas governamentais do GovPE, que ficam disponíveis no Portal da Transparência do GovPE. Este portal foi implementado em 2007 sobre a mesma plataforma do Portal da LAI (customização do Liferay). Neste portal, os dados das despesas de cada instituição do GovPE estão acessíveis de duas maneiras: através de um Portlet chamado de “Gasto Transparência” e através de web services. O Portlet exibe as informações em HTML, em planilhas XLS, ou PDF, padrão de documentos de propósito geral. Estes formatos são

destinados à leitura de pessoas, HTML e PDF são desestruturados, XLS é proprietário e nenhum dos três dá significado semântico aos dados.

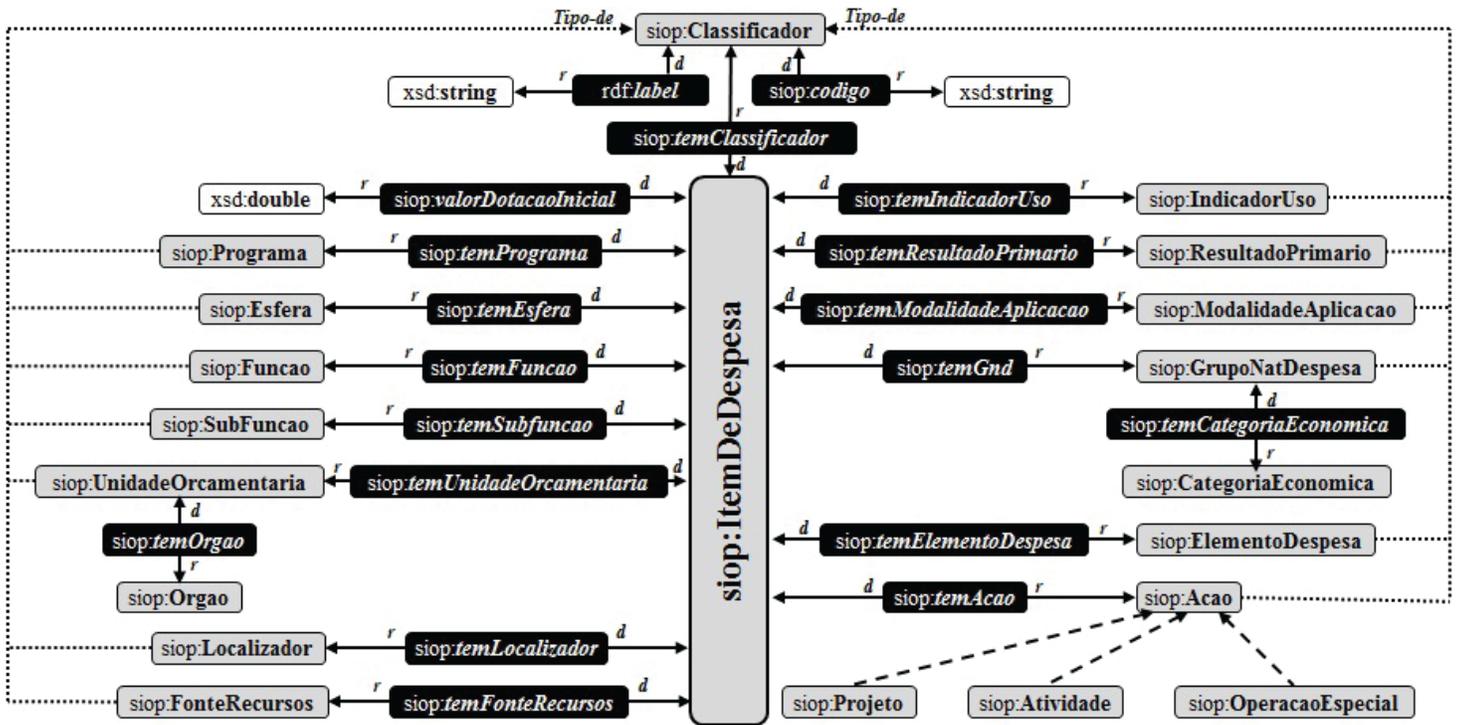
Já os web services, utilizam SOAP (*Simple Object Access Protocol*), um protocolo para transmissão de dados estruturados em ambientes distribuídos que utiliza XML na troca de mensagens, que pode se dá via HTTP. Acompanham estes web services, seus respectivos documentos WSDL (*Web Services Description Language*), artefatos que fornecem uma descrição dos serviços oferecidos pelos web services, funcionando como uma interface de acesso aos objetos remotos. Os web services também possuem uma documentação que apresenta exemplos das mensagens SOAP de requisição e resposta, tipo de retorno e parâmetros de chamadas dos métodos.

Para o melhor entendimento da estrutura dos dados das despesas do GovPE, foram estudados detalhadamente os artefatos relacionados com os web services das despesas e com o Portlet “Gasto Transparência”: documentação e código de implementação dos web services, código de implementação e modelo de classes do Portlet. Além disso, foram consultados desenvolvedores do Portal da Transparência. Após este estudo, foi feita uma pesquisa de ontologias publicadas na web, a fim de encontrar termos que se encaixassem no modelo semântico das despesas do GovPE.

### **3.2.1 Ontologia da Lei Orçamentaria Anual de 2012**

Encontrou-se, então, a ontologia em OWL criada pela Secretaria do Orçamento Federal para representação dos dados orçamentários federais de 2012, a Ontologia da Lei Orçamentaria Anual de 2012 (LOA 2012, que tem seu diagrama de classes e propriedades mostrado na figura 3.3) [4]. Constatou-se que algumas classes e propriedades de LOA 2012, se adequavam ao modelo de dados das despesas das instituições do GovPE, porém vários conceitos relativos a despesas ainda não estavam expressos. Isto motivou a criação da Ontologia Despesas GovPE, uma nova ontologia criada para o contexto das despesas do GovPE que importa LOA 2012. Quando uma ontologia importa outra, ganha acesso aos recursos (classes e propriedades) da ontologia importada e pode utilizá-los para a construção de seus próprios termos.

Figura 3.3: diagrama de classes e propriedades da Ontologia da Lei Orçamentaria Anual de 2012



Fonte: BRASIL, 2012.

As despesas do GovPE estão organizadas de um modo que o empenho é a entidade básica. Um empenho é um documento emitido pela Administração Pública que dá ao credor, entidade que tem direito a receber algum pagamento, o direito de exigir o pagamento devido naquele documento. Os empenhos estão associados a um ou mais itens de despesa e é principalmente nesta relação que está o maior elo entre o modelo de dados de LOA 2012 e o de Despesas GovPE. LOA 2012 modela um item de despesa, entidade central da ontologia, relacionando-o com outras entidades classificadoras, de modo que satisfaz a semântica dos itens de despesa do GovPE.

### 3.2.2 Especificação da Ontologia Despesas GovPE

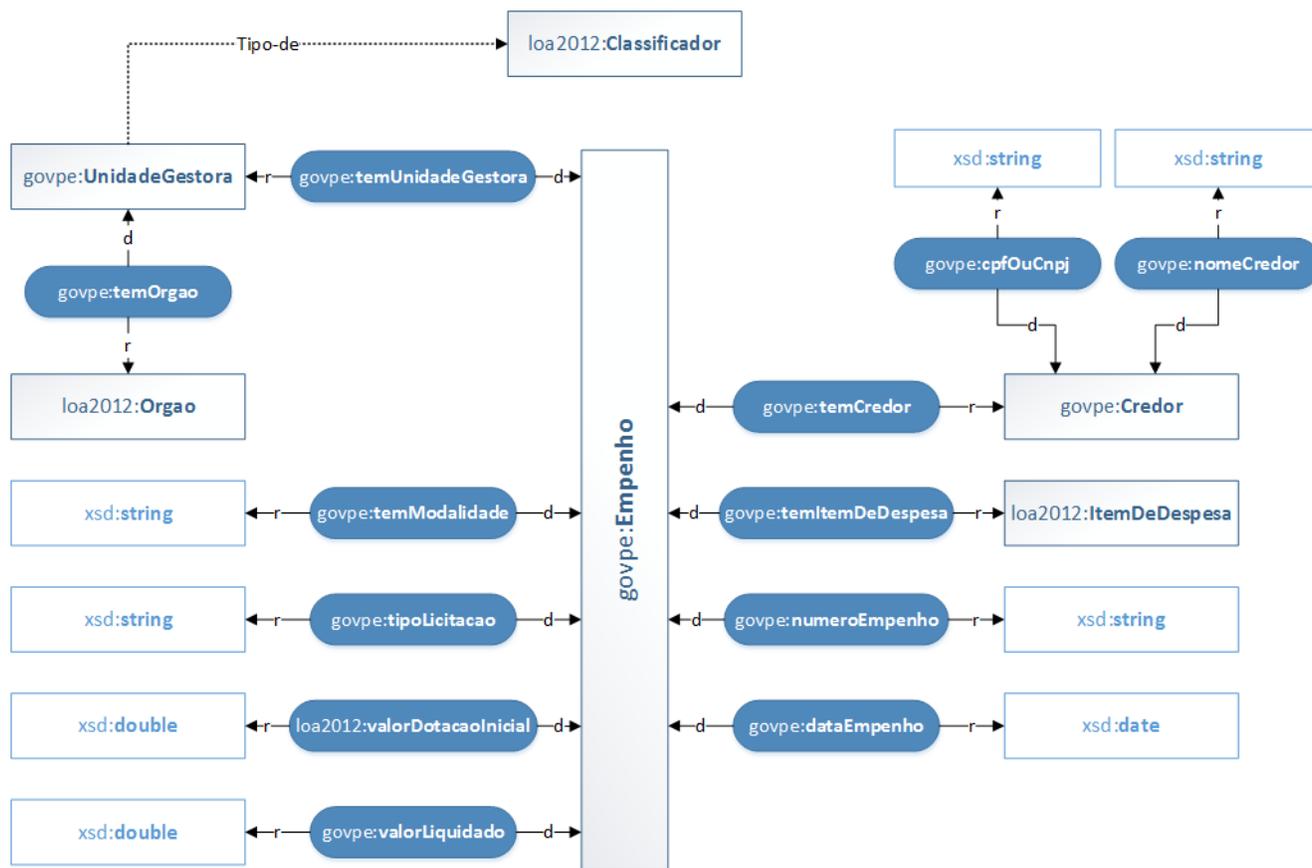
Seguindo o modelo das despesas GovPE, a Ontologia Despesas GovPE foi implementada em OWL com auxílio do Protégé. Este programa de código aberto é um editor de ontologias e bases de conhecimento. Neste projeto foi utilizado mais especificamente seu módulo chamado Protégé-OWL, que trabalha com OWL, RDF e RDFS, entre outras tecnologias. A figura 3.4 mostra o diagrama da ontologia

desenvolvida, já seu código fonte em OWL pode ser consultado no “APÊNDICE A” deste trabalho.

A modelagem da ontologia foi feita baseando-se em um cenário no qual tanto os empenhos quanto os itens de despesa estão relacionados a várias entidades e possuem diversas classificações e atributos que se organizam resumidamente da seguinte forma:

- Um empenho está relacionado a um ou mais itens de despesa, tem um ou mais credores, está associado a uma unidade gestora e possui valor de dotação inicial, valor liquidado, modalidade, tipo de licitação, número e data.
- Uma unidade gestora pode ser vista como uma entidade classificadora, possui etiqueta, código e tem vínculo com um órgão.
- Um credor tem um número de CPF ou CNPJ e um Nome.
- Finalmente, um item de despesa pode ser classificado de acordo com diversas entidades: programa, esfera, função, subfunção, ação, grupo de despesa, categoria econômica e elemento de despesa.

Figura 3.4: diagrama de classes e propriedades da Ontologia Despesas GovPE.



Fonte: elaborada pelo autor no desenvolvimento deste projeto.

Como é possível constatar, LOA 2012 é uma ontologia de orçamento e, por isso, descreve muito bem entidades que modelam orçamento, como é item de despesa. Porém, as entidades que estão relacionadas com despesas governamentais, e não com orçamento, precisaram ser criadas: “Empenho”, “Credor” e “Unidade Gestora”. A tabela 3.1, detalha estas novas classes criadas na Ontologia Despesas GovPE.

**Tabela 3.1: classes da Ontologia Despesas GovPE.**

<b>Classe</b>	<b>Descrição</b>
Empenho	Classe cujos indivíduos representam os empenhos emitidos pelo governo. Cada elemento da classe “Empenho” é caracterizado por se relacionar com um ou mais indivíduos de “ItemDeDespesa” e com um mais indivíduos de “Credor”. Possui também os atributos: “numeroEmpenho”, “valorLiquidado”, “dataEmpenho”, “temModalidade”, “tipoLicitacao” e “temModalidade”.
Credor	Classe que representa os credores dos empenhos. Os elementos deste tipo possuem os atributos “cpfOuCnpj” e “nomeCredor”.
UnidadeGestora	Classe que representa as unidades gestoras vinculadas, cada uma a um elemento de “Orgão” e que classifica indivíduos de “Empenho” e de “ItemDeDespesa”, configurando-se como um tipo (ou classe filha) de “Classificador”, entidade de LOA 2012. Herda, então, os atributos da classe pai: “nome” e “label” (rótulo).

Nas descrições da tabela 3.1, foram explicitadas as relações das classes com outras entidades e também os seus atributos. A partir destas relações se criam na ontologia as propriedades de objeto (object properties), e os atributos são expressos através das propriedades de dado (data properties). As primeiras são aquelas presentes em triplas RDF link e as últimas são as utilizadas em triplas literais. As tabelas 3.2 e 3.3, respectivamente, especificam as propriedades de objetos e de dado que foram criadas ou modificadas pela Ontologia Despesas GovPE.

**Tabela 3.2: propriedades de objeto da Ontologia Despesas GovPE.**

<b>Propriedade de objeto</b>	<b>Descrição</b>
temItemDeDespesa	Propriedade que relaciona a classe “Empenho” (domínio) à classe “ItemDeDespesa” (range).
temCredor	Propriedade que representa a relação entre “Empenho” (domínio) e “Credor” (range).
temUnidadeGestora	Propriedade que representa a relação da classe “Empenho” ou da classe “ItemDeDespesa”, que representam o seu domínio, com a entidade “UnidadeGestora”, seu range.
temOrgao	Propriedade definida pela ontologia do orçamento federal, LOA 2012, que tem como range a classe “Orgao” e tinha em seu domínio apenas a classe “UnidadeOrçamentaria”, mas “UnidadeGestora” foi agregada.

**Tabela 3.3: propriedades de dados da Ontologia Despesas GovPE.**

<b>Propriedade de dado</b>	<b>Descrição</b>
cpfOuCnpj	Propriedade que representa o número de CPF, caso o credor seja uma pessoa física, ou o número de CNPJ, caso seja uma pessoa jurídica. Tem como domínio a classe “Credor” e como range o tipo primitivo xsd:string, que representa cadeia de caracteres.
dataEmpenho	Propriedade que representa a data que um item de despesa foi agregado empenho. A classe “Empenho” é o domínio e o range é o tipo xsd:date, tipo primitivo para datas.
nomeCredor	Propriedade de domínio “Credor” e range xsd:string que representa o nome do credor.
numeroEmpenho	Representa o número de identificação do “Empenho”, classe que é domínio desta propriedade, cujo range é o tipo xsd:string.
temModalidade	Propriedade tem como range o tipo xsd:string, cadeia de caracteres que representa a modalidade do “Empenho”, que é o domínio.

tipoLicitacao	Tipo da licitação, se aplicável, que gerou o “Empenho”, domínio desta propriedade, que range xsd:string.
valorLiquidado	Propriedade que representa o valor pago a um credor, relativo a algum item de despesa do “Empenho”, classe que é seu domínio. Seu range é o tipo xsd:double.
valorDotacaoInicial	Propriedade de LOA 2012 que representa o valor inicial do “ItemDeDespesa”. Seu o domínio, passou a ter também a classe “Empenho”, representando seu valor inicial. Tem como range o tipo xsd:double.

### 3.3 Implementação

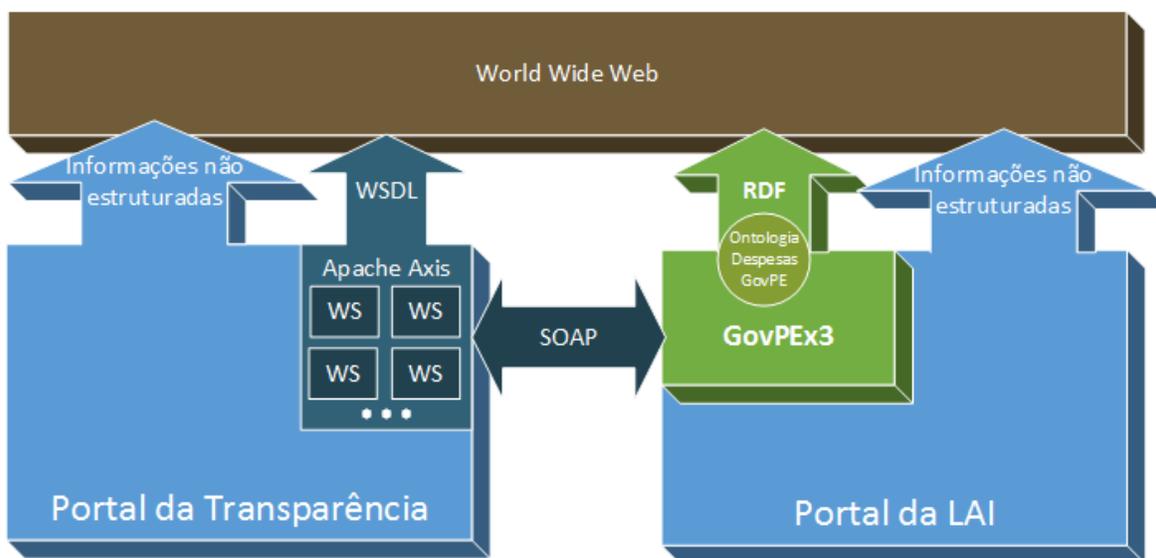
A GovPEX3 consiste em uma extensão para o Portal da LAI, para que este seja utilizado pelos gestores na publicação dos dados das despesas das instituições do GovPE em formato aberto e estruturado, seguindo padrões da Web Semântica e os princípios de Linked Data. A ferramenta realiza um processo ETL, ou seja, extrai dados primários nos formatos não desejados, em seguida os converte para o formato desejado e por último os mantém disponíveis na web.

Como indicado na seção 3.2, o Portal da Transparência disponibiliza serviços de consultas dos dados das despesas das instituições em formato estruturado através de web services SOAP. Como foi visto, este não é um padrão da Web Semântica e não possibilita a interligação dos dados com dados de outras fontes automaticamente. Apesar disso, estes serviços serviram de fontes de dados primários para um processo de ETL que segue os padrões estudados neste trabalho. A figura 3.5 mostra um diagrama representando o contexto no qual está inserida a ferramenta projetada neste trabalho. Em verde, a GovPEX3 aparece como um módulo do Portal da LAI que se comunica com os web services que estão no Portal da Transparência através de mensagens SOAP e disponibiliza na web dados em RDF.

Como o Portal da LAI reside sobre uma plataforma feita em Java, a ferramenta proposta, que está acoplada ao portal, também foi implementada na mesma linguagem de programação. Após a análise do sistema e da hierarquia de chamada dos métodos dos web services, o desenvolvimento da GovPEX3 pôde ser dividido nas seguintes etapas:

- 1) **Desenvolvimento dos clientes para os web services** do Portal da Transparência e criação de uma interface de acesso para os serviços remotos das despesas.
- 2) **Adição do tipo RDF** ao conjunto de tipos de informações do Portal da LAI. Com isto, o tipo fica elegível para o administrador na definição de estrutura de página, podendo ser adicionado a uma área de página do portal.
- 3) **Implementação do Portlet “Publicador RDF”** contendo interface de acesso às triplas e de configuração do órgão (advindo do Portal da Transparência) que será associado à instituição (do Portal da LAI) na geração das triplas.
- 4) **Desenvolvimento do mapeamento dos dados**, isto é, a conversão de objetos Java do modelo de retorno dos web services para triplas RDF.
- 5) **Implementação dos mecanismos de persistência e de disponibilização** dos conjuntos de triplas RDF.

Figura 3.5: diagrama de contexto da GovPEX3.



Fonte: elaborada pelo autor no desenvolvimento deste trabalho.

### 3.3.1 Desenvolvimento dos clientes para os web services

Os clientes dos web services foram gerados através da IDE Eclipse<sup>12</sup> (ambiente integrado de desenvolvimento de código aberto Java nativo). O Eclipse utiliza o Apache Axis<sup>13</sup>, biblioteca Java que implementa o protocolo SOAP, para receber a URL do documento WSDL de um web service e gerar as classes Java do cliente do web service. As classes geradas incluem: conexão com o serviço remoto, chamada dos métodos e modelagem dos objetos. Para cada web service, são geradas cinco classes de serviços de conexão e chamadas de métodos, e quantas classes de modelo forem necessárias para receber os retornos das chamadas. Geralmente uma classe de modelo é criada, pois cada web service costuma retornar objetos de uma entidade. As chamadas aos métodos dos web services foram reunidas em uma classe Java de fachada nomeada “WSClientDespesasGovPE”. Foram utilizados sete web services do Portal da Transparência:

- ElementoDespesaCategoriaEconomicaServiceSoap
- ElementoDespesaGrupoDespesaServiceSoap
- ElementoDespesaServiceSoap
- EmpenhoDespesaServiceSoap
- ItemDespesaServiceSoap
- OrgaoServiceSoap
- UnidadeGestoraServiceSoap

### 3.3.2 Adição do tipo RDF ao portal

O processo de publicação de RDF da GovPEx3 respeitou a sequência de atividades realizadas pelo administrador e pelo gestor na publicação de conteúdos não estruturadas, explicado na seção 3.1. Então, como para qualquer tipo de conteúdo do Portal da LAI, para que as triplas RDF das despesas da instituição sejam publicadas, o administrador do portal deve inserir o respectivo tipo (no caso, “RDF”) na área de uma página e, em seguida, associar esta página a um item de menu na definição de estrutura da instituição. Feito isso, os gestores da instituição devem adicionar um Portlet de publicação para aquele tipo de dado (o “Publicador RDF” foi desenvolvido na próxima etapa da implementação) e configurar as preferências do Portlet, de acordo com sua

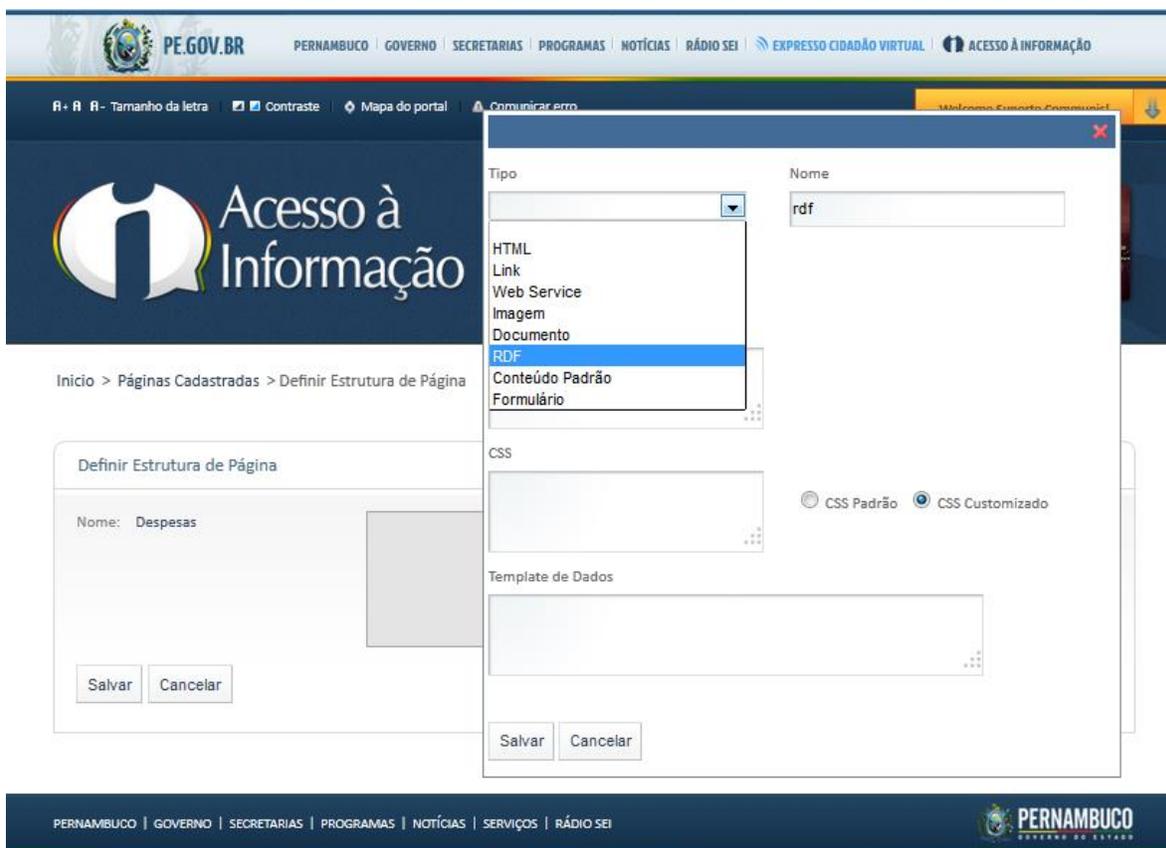
---

<sup>12</sup> <http://www.eclipse.org>

<sup>13</sup> <http://axis.apache.org>

instituição. Esta etapa tratou então, de adicionar um novo tipo “RDF” ao conjunto de tipos de dados do portal, tornando-o elegível pelo administrador para uma área de página. A figura 3.6 mostra a interface de administrador para seleção do tipo de conteúdo de uma área de página com o tipo RDF já configurado.

**Figura 3.6: interface de definição de estrutura de página do Portal da LAI (seleção do tipo de informação para uma área da página).**



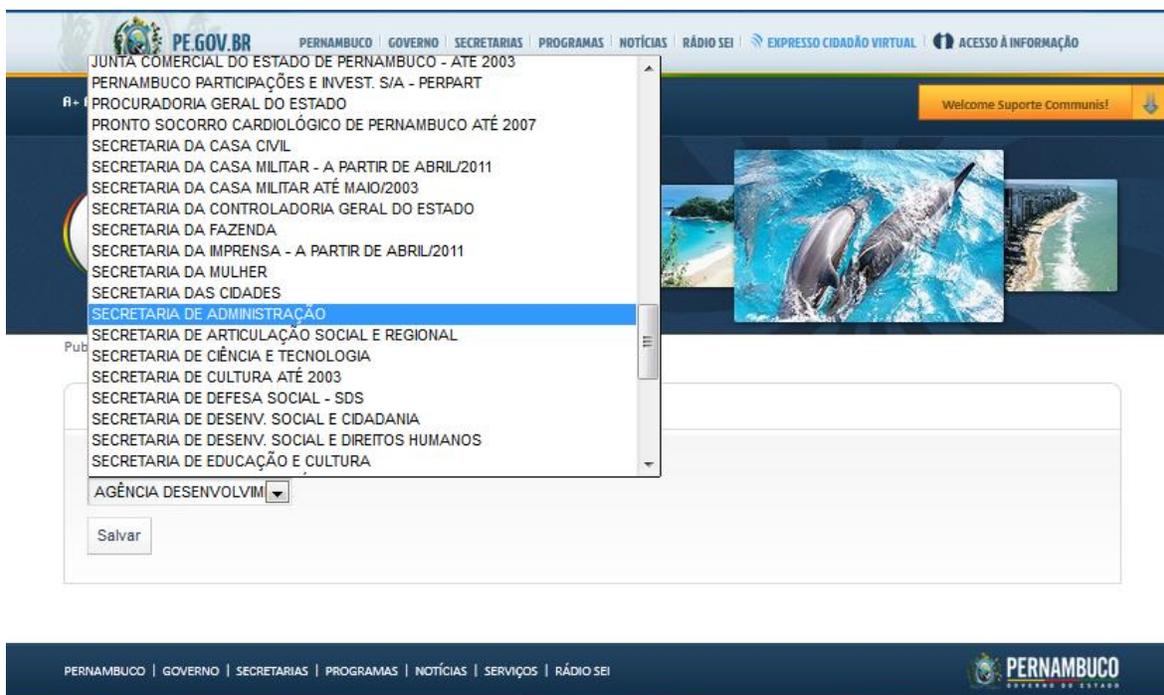
Fonte: captura de tela do Portal da LAI (parte administrativa do portal).

### 3.3.3 Implementação do Portlet “Publicador RDF”

Um tipo de conteúdo, no Portal da LAI, está sempre associado a um Portlet, que trata de exibi-lo ou torna-lo acessível de alguma forma. No caso da GovPEX3 foi desenvolvido o Portlet “Publicador RDF” para ser associado ao tipo RDF. Como a maioria dos Portlets do portal, ele possui duas visões, uma pública e outra privada ou de preferências. Em sua parte pública, o “Publicador RDF” exibe um link para o documento OWL da Ontologia Despesas GovPE e outro link para as triplas RDF geradas em “Turtle”, se estas já tiverem sido geradas para a instituição. A parte de preferências está disponível

para os administradores do portal e para os gestores da instituição. O gestor é incumbido unicamente de escolher o órgão ao qual sua instituição está vinculada e salvá-lo. A tela de preferências é apresentada na figura 3.7. Os órgãos que aparecem como opção na caixa de seleção são os retornados pelo web service do Portal da Transparência que tem seus serviços relacionados a entidade órgão. Retornam objetos do tipo (classe Java) “OrgaoServiceSoap”.

Figura 3.7: interface de preferências do Portlet “Publicador RDF”, seleção do órgão da instituição.



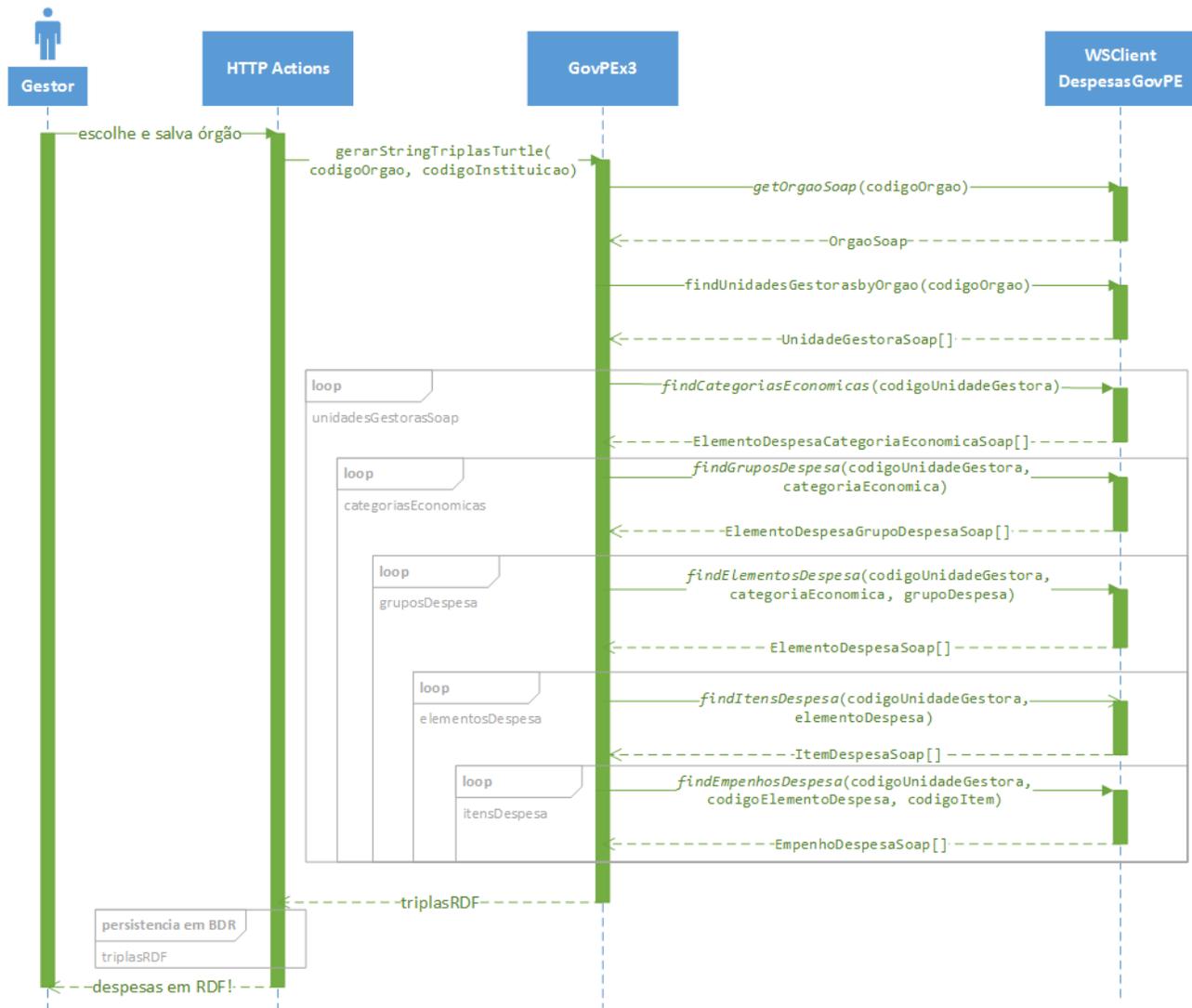
Fonte: captura de tela do Portal da LAI mostrando Portlet desenvolvido neste projeto.

### 3.3.4 Desenvolvimento do mapeamento dos dados

No momento em que o gestor submete um órgão selecionado por ele, a GovPEX3 inicia o mapeamento dos dados: são convertidos os objetos de classes Java retornados pelos web services em triplas RDF. A hierarquia de chamada dos web services das despesas foi desenvolvida seguindo a sequência apresentada no diagrama da figura 3.8. As HTTP Actions, ou ações HTTP, que aparecem no diagrama são os módulos de um sistema web que tratam as requisições HTTP realizadas pelos usuários, executando uma ação e retornando uma resposta para o usuário. Nesta etapa de desenvolvimento foram implementados os procedimentos necessários para a geração de triplas. Estes procedimentos compõem o módulo indicado com o nome da ferramenta na figura 3.8.

Este módulo recebe os objetos remotos retornados pelos web services e cria as triplas RDF relacionadas a cada objeto.

Figura 3.8: diagrama de sequência da GovPEX3.



Fonte: elaborada pelo autor no desenvolvimento deste trabalho.

Para a implementação do mapeamento entre os objetos retornados e as triplas foi usada a plataforma Apache Jena<sup>14</sup>, que implementa em Java: leitura, processamento e escritura de RDF (em XML, N-triples e Turtle), tratamento de ontologias em OWL e RDFS, um motor de inferências baseado em regras, entre outras coisas. O carregamento da

<sup>14</sup> <http://jena.apache.org>

Ontologia Despesas GovPE em memória e conseqüentemente de LOA2012 (como é importada pela primeira, é carregada automaticamente) é feito através de um objeto da classe de Jena chamada de “OntModel”, que representa um grafo RDF (ou um conjunto de triplas). Um novo conjunto de triplas (objeto do tipo OntModel) é criado para serem adicionadas as triplas das instâncias das classes das ontologias mapeadas dos tipos de retorno dos web services. Na medida em que os métodos remotos dos web services, chamados em “WSClientDespesasGovPE”, retornam os objetos remotos, estes são convertidos seguindo o mapeamento desenvolvido, indicado na tabela 3.1. Logo, os indivíduos das classes da ontologia, suas relações com outros indivíduos (RDF links) e seus atributos (triplas literais) são adicionadas ao conjunto de dados de instâncias. A criação do indivíduo se dá pela geração de um URI para o objeto e colocando-o como sujeito de uma tripla que indica que este URI representa um membro da classe mapeada. Os métodos dos web services recebem também um ano como parâmetro e retornam apenas os dados referentes aquele ano, porém, na implementação da GovPEX3, este atributo foi fixado no ano atual e, por isso, este parâmetro foi ignorado no diagrama de seqüência da figura 3.8.

**Tabela 3.1: mapeamento dos dados das despesas realizado pela GovPEX3.**

<b>Classe de retorno dos web services</b>	<b>Classe da Ontologia Despesas GovPE</b>
OrgaoSoap	govpe:Orgao
UnidadeGestoraSoap	govpe:UnidadeGestora
EmpenhoDespesaSoap	govpe:Empenho
EmpenhoDespesaSoap.getCnpjCpf() EmpenhoDespesaSoap.getNomeCredor()	govpe:Empenho
ElementoDespesaCategoriaEconomicaSoap	loa2012:CategoriaEconomica
ElementoDespesaGrupoDespesaSoap	loa2012:GrupoNatDespesa
ElementoDespesaSoap	loa2012:ElementoDespesa
ItemDespesaSoap	loa2012:ItemDeDespesa
OrgaoSoap.getPoder()	loa2012:Esfera

Os prefixos “govpe” e “loa2012”, na tabela 3.1 indicam os prefixos associados aos namespaces das ontologias Despesas GovPE e LOA2012, respectivamente. O namespace do conjunto de dados de instâncias é criado adicionando o sufixo “-i?codigoInstituicao=” ao namespace da Ontologia Despesas GovPE e concatenando ao

final o código da instituição que está tendo suas despesas convertidas em RDF. O prefixo do namespace do conjunto instâncias foi fixado em: “govpe-i”. Os URI’s das instâncias são criados com base nas classes às quais pertencem e no seu código identificador, de acordo com a seguinte regra: <namespace><classe>-<código>. Caso a entidade não possua um código identificador como atributo, a parte <código> é substituída pelo atributo que representa o nome da entidade substituindo os espaços em branco pelo caractere ‘\_’. O formato de serialização RDF escolhido para geração das triplas foi o Turtle (*Terse RDF Triple Language*, que em português significa linguagem concisa de triplas RDF), pois é um formato enxuto (sem muita redundância) e simples, de fácil leitura inclusive para as pessoas.

Para exemplificar as triplas resultantes desta etapa de implementação e os padrões adotados, abaixo estão as dois trechos de descrições RDF, em Turtle, de dois objetos gerados pela GovPEX3. O primeiro é uma instância da classe “govpe:Empenho” e o segundo da classe “govpe:Credor”.

```
govpe-i:Empenho-2013NE000253
  a govpe:Empenho ;
  govpe:dataEmpenho "2013-04-08T03:00:00Z"^^xsd:dateTime ;
  govpe:numeroEmpenho "2013NE000253"^^xsd:string ;
  govpe:temCredor govpe-i:Credor-89021398400 ;
  govpe:temItemDeDespesa
    govpe-i:ItemDeDespesa-2 ;
  govpe:valorLiquidadado
    "403.52"^^xsd:double .

govpe-i:Credor-89021398400
  a govpe:Credor ;
  govpe:cpfOuCnpj "89021398400"^^xsd:string ;
  govpe:nomeCredor "LEONARDO CAVALCANTI CARNEIRO"^^xsd:string .
```

O primeiro trecho denota seis triplas que podem ser lidas por extenso da seguinte forma: o recurso de URI “govpe-i:Empenho-2013NE000253” é do tipo empenho, tem data “8/4/2013”, tem número “2013NE000253”, seu credor tem URI “govpe-i:Credor-89021398400”, está relacionado ao item de despesa de URI “govpe-i:ItemDeDespesa-2” e tem como valor já liquidado a quantia de R\$ 403,52.

O segundo trecho tem três triplas RDF e pode ser lido por extenso assim: o recurso com URI “govpe-i:Credor-89021398400” é do tipo credor, tem CPF ou CNPJ de número “89021398400” e se chama “LEONARDO CAVALCANTI CARNEIRO”.

### 3.3.5 Implementação dos mecanismos de persistência e disponibilização das triplas RDF

Uma vez que foram geradas as triplas RDF, a quinta e última etapa do desenvolvimento está relacionada com a persistência e a disponibilização dos dados. Para o armazenamento das triplas foi criada uma tabela no banco de dados relacional do Portal da LAI. O portal trabalha com o sistema de gerenciamento de banco de dados (SGBD) PostgreSQL (o código SQL de criação da tabela é apresentado no “APÊNDICE B” deste trabalho). A tabela foi criada com os seguintes campos (ou colunas):

- **Código da instituição**, um número inteiro (representado no PostgreSQL pelo tipo “bigint”) que referencia a tabela de instituições (chave estrangeira);
- **Triplas**, uma cadeia de caracteres de tamanho variável (representado no PostgreSQL pelo tipo “text”) que guarda um conjunto de triplas RDF;
- **Serialização**, uma cadeia de caracteres que representa o nome da linguagem RDF em que foram geradas as triplas (tipo “text”);
- **Código da tripla**, identificador do conjunto de triplas, número inteiro (tipo “bigint”) que identifica o registro da tabela (chave primária).
- Alguns campos de auditoria são incluídos em todas as tabelas do Portal da LAI e não foi diferente na tabela das triplas: usuário que criou o registro, data de criação do registro, último usuário a modificar o registro e data da última modificação.

A disponibilização das triplas é feita através de uma HTTP Action. No caso, a action “GetTriplasAction” foi criada para consultar o banco e retornar as triplas requisitadas. Esta action recebe um parâmetro do tipo GET, ou seja, via URL, com o código da instituição que terá suas triplas recuperadas. Esta action possui uma URL associada a ela e a geração do namespaces dos modelos das instancias foi modelado de tal forma que são iguais à URL da “GetTriplasAction” já com o parâmetro que indica o código da instituição a qual se refere modelo de instâncias retornado. Em ambiente localhost, como o que foi desenvolvido a GovPEX3, a URL da “GetTriplasAction”, para retornar as triplas da instituição de código “1”, por exemplo, seria: [http://localhost:8080/c/publicador\\_rdf/despesas\\_govpe-i?codigoInstituicao=1](http://localhost:8080/c/publicador_rdf/despesas_govpe-i?codigoInstituicao=1) . A implicação disso é que, ao se requisitar a URI de um objeto das despesas da instituição de código “1” via HTTP, serão retornadas todas as triplas das despesas desta instituição.

Tal fato é a aplicação do princípio de Linked Data que recomenda disponibilizar informações úteis sobre os recursos, através de seus URI's.

Para completar, foram implementadas as HTTP actions "GetOntoDespesasAction" e "GetOntoLOA2012Action". A primeira responde entregando o arquivo OWL da Ontologia Despesas GovPE (no ambiente de desenvolvimento tem URL: [http://localhost:8080/c/publicador\\_rdf/despesas\\_govpe](http://localhost:8080/c/publicador_rdf/despesas_govpe)). A segunda retorna a ontologia LOA2012 para que esta possa ser recuperada em caso de servidor que a mantem, o repositório de ontologias do Governo Federal<sup>15</sup>, estar indisponível. Com isso, se encerra o desenvolvimento da GovPEX3.

---

<sup>15</sup> <http://vocab.e.gov.br>

## 4 Conclusão

Esta seção conclui o relatório deste projeto fazendo várias considerações acerca das contribuições realizadas e dos resultados obtidos. Primeiramente serão descritas as contribuições e será feita uma reflexão sobre os impactos destas contribuições. Logo, será feita uma avaliação dos resultados logrados nas diferentes etapas do trabalho, sendo apontados aspectos do projeto que podem melhorar. Finalmente, serão conjecturados possíveis trabalhos futuros neste ramo da Ciência da Computação.

Este trabalho fez uso, na criação da GovPEX3, de tecnologias e padrões da Web Semântica: namespaces, URIs, RDF, OWL. O algoritmo da ferramenta segue os três primeiros princípios de Linked Data: cria URI's HTTP para identificar objetos e conceitos do mundo real (empenhos e itens de despesa, entre outros) e disponibiliza informações úteis sobre os objetos, utilizando RDF, com semântica definida por uma ontologia criada em OWL. Os dados publicados pela ferramenta possuem os atributos necessários para serem considerados dados governamentais abertos:

- Relativamente **completos** porque todos os dados das despesas de uma instituição determinada retornados pelos web services do ano atual são convertidos em triplas.
- **Primários** porque os web services retornam dados brutos, que não foram modificados.
- **Em tempo oportuno** porque são dados de despesas realizadas no ano atual, que seguem sendo úteis por muito tempo.
- **Acessíveis** porque são publicados para todas as pessoas com acesso a web, através do Portal da LAI.
- **Podem ser processados por um computador** e são **não proprietários** porque são publicados em RDF, formato livre, sobre o qual nenhuma entidade tem controle exclusivo, e que permite leitura e manipulação automatizadas.
- **Não discriminatórios** porque não exigem cadastramento ou identificação dos seus consumidores.
- E **livres de licenças** porque não existem amarras jurídicas que limitem o seu acesso, nem o seu uso.

Este trabalho realizou, portanto, contribuições relevantes para a consolidação da web dos dados e para o processo de publicação de dados abertos do GovPE. A criação e

publicação da Ontologia Despesas GovPE contribui para que, futuramente, outras bases de dados sobre despesas governamentais possam ser publicadas reusando a ontologia, poupando esforços que seriam destinados a criação de uma nova para modelar este domínio. A ferramenta desenvolvida foi outra contribuição relevante e espera-se que seja aproveitada pela Agência de Tecnologia da Informação do GovPE e seja posta em produção (no ar), permitindo que os gestores do GovPE possam, facilmente, publicar os dados das despesas de suas respectivas instituições. A publicação dos dados em RDF contribui para o crescimento do grafo de dados da web e dá suporte para a criação de aplicações web, que possivelmente integram este conjunto de dados com outros (mashups). Estas aplicações potencializam as informações contidas nos dados (podendo usar, por exemplo, técnicas de Visualização de Dados, BI e Estatísticas), que podem e devem ser usadas para fins de fiscalização e acompanhamento da gestão pública.

Colocando em números a quantidade de dados gerados em RDF para algumas instituições, pode-se estimar a quantidade de dados que podem ser gerados e publicados no Portal da LAI através da GovPEx3, e com isso, e avaliar o potencial da ferramenta. Foram convertidos em RDF os dados das despesas de quatro instituições obtendo-se, para cada uma, um conjunto de triplas RDF. As instituições e quantidades de triplas convertidas estão expostas na tabela 4.1.

**Tabela 4.1: número de triplas RDF geradas a partir dos dados das despesas de instituições do GovPE.**

<b>Instituição do GovPE</b>	<b>Número de triplas RDF</b>
Secretaria da Fazenda (SEFAZ)	2142
Fundação do Patrimônio Histórico e Artístico de Pernambuco (FUNDARPE)	3856
Secretaria Extraordinária da Copa do Mundo da FIFA 2014 (SECOPA)	353
Secretaria de Defesa Social (SDS)	4685

Outro resultado conquistado pela GovPEx3 foi a elevação de classe dos dados publicados no Portal da LAI, dentro do esquema de classificação por estrelas proposto por Berners-Lee, saindo de uma estrela para quatro estrelas. Os dados considerados uma estrela são aqueles publicados em formatos desestruturados, enquanto que quatro estrelas são os dados identificados por URIs e descritos em RDF.

Como trabalho futuro, temos a interligação dos dados das despesas a dados provenientes de outras fontes RDF. Isso instalaria o terceiro princípio de Linked Data ao processo de publicação da GovPE<sup>3</sup> e elevaria seus dados ao patamar cinco estrelas, o mais alto no modelo de classificação de Berners-Lee. A fixação do ano atual na filtragem dos dados recuperados é outro ponto que pode ser modificado em uma próxima versão do projeto para permitir mais flexibilidade na geração de triplas.

Finalizando, são cogitadas algumas propostas de trabalhos na área de dados abertos interligados e Web Semântica, que surgem ao observar os resultados deste projeto. A criação de um endpoint SPARQL para permitir que as aplicações fizessem consultas sobre as triplas do Portal da LAI e porventura de outros portais do GovPE seria uma proposta interessante. Outra proposta seria o desenvolvimento de uma ferramenta de publicação de dados RDF a partir de formatos desestruturados em geral, interligando estes dados a outras bases RDF. Mais uma proposta seria uma aplicação de visualização de dados para o GovPE, interligando dados RDF geográficos e de despesas e permitindo que a população visualize e identifique os locais onde estão sendo feitos os investimentos do governo. Esta aplicação aproveitaria a infraestrutura da Web Semântica e dos dados interligados para apresentar informações úteis para sociedade. São aplicações assim que prometem mudar o rumo da web e construir a Web 3.0, a Web dos Dados.

## 5 Referências

- [1] BERNERS-LEE, Tim; HENDLER, James e LASSILA, Ora. **The Semantic Web: a new form of Web content that is meaningful to computers will unleash a revolution of new possibilities.** Scientific American, 17 de maio de 2001.
- [2] BERNERS-LEE, Tim. **Tim Berners-Lee on the next Web.** (conferência) TED Ideas Worth Spreading 2009, fevereiro de 2009.
- [3] BERNERS-LEE, Tim *et al.* **Linked Open Government Data: lessons from data.gov.uk.** Intelligent Systems, IEEE , vol.27, no.3, pp.16,24, maio-junho 2012.
- [4] BRASIL. Ministério do Planejamento, Orçamento e Gestão. Secretaria de Orçamento Federal. **Lei orçamentária anual LOA 2012 em formato aberto: manual de referência.** Brasília, 2012.
- [5] COLETTA, Remi *et al.* **Public Data Integration with WebSmatch.** First International Workshop On Open Data, Nantes, França, 2012.
- [6] GOMES, Paulo e GAMA, Fernando. **Ontologia: arquitetura da informação digital.** São Paulo, 2009.
- [7] HEALTH, Tom e BIZER, Christian. **Linked Data: evolving the web into a global data space (1ª edição).** Synthesis Lectures on the Semantic Web: Theory and Technology, 1:1, 1-136, Morgan & Claypool Publishers, 2011.
- [8] HENDLER, James *et al.* **US Government Linked Open Data: semantic.data.gov.** Intelligent Systems, IEEE , vol.27, no.3, pp.25,31, maio-junho 2012.
- [9] MAALI, Fadi; CYGANIAK, Richard e PERISTERAS Vassilios. **A Publishing Pipeline for Linked Government Data.** 9th Extended Semantic Web Conference (ESWC2012), Heraklion, Creta, Grécia, 2012.

- [10] MACHADO, Alexandre e OLIVEIRA, José. **DIGO**: an open data architecture for e-government. 15th IEEE International Enterprise Distributed Object Computing Conference Workshops, 2011.
- [11] **Manual dos dados abertos**: governo / traduzido e adaptado de opendatamanual.org; [cooperação técnica científica entre Laboratório Brasileiro de Cultura Digital e o Núcleo de Informação e Coordenação do Ponto BR (NIC.br)]. – São Paulo : Comitê Gestor da Internet no Brasil, 2011.
- [12] OLIVEIRA, Marcelo e SOUZA, Ellen. **Investigação e Desenvolvimento de Soluções de Dados Governamentais Abertos** (projeto de pesquisa e extensão). Serra Talhada: UFRPE, 2013.
- [13] SCGE, Secretaria da Controladoria Geral do Estado. **Lei de Acesso à Informação é implantada em Pernambuco** (notícia publicada no portal). [http://www.portaisgoverno.pe.gov.br/web/scge/exibir\\_noticia?groupId=381648&articleId=2286071&templateId=382363](http://www.portaisgoverno.pe.gov.br/web/scge/exibir_noticia?groupId=381648&articleId=2286071&templateId=382363), acessado em 24 de janeiro de 2013.
- [14] RAMALHO, Rogério; VIDOTTI, Silvana e FUJITA, Mariângela. **Web Semântica**: uma investigação sob o olhar da Ciência da Informação. DataGramaZero, Rio de Janeiro, v. 8, p. 4, 2007.
- [15] SANTOS, Plácida e ALVES, Rachel. **Metadados e Web Semântica para estruturação da Web 2.0 e Web 3.0** (ARTIGO 04). DataGramaZero, Rio de Janeiro, v.10, n.6, dezembro de 2009.
- [16] SILVA, Geiza e LIMA, Tarcísio (orientador). **RDF e RDFS na Infra-estrutura de Suporte à Web Semântica**. Revista Eletrônica de Iniciação Científica, 2002.
- [17] O'REILLY, Tim. **What is Web 2.0**: design patterns and business models for the next generation of software. Communications & Strategies, No. 65, p. 17, 1º trimestre de 2007.

- [18] O'REILLY, Tim e BATTELLE, John. **Web Squared**: Web 2.0 five years on. Web 2.0 Summit, São Francisco, CA, EUA, 20-22 de outubro de 2009.

# APÊNDICE A - Ontologia Despesas GovPE em OWL

```
<?xml version="1.0"?>

<!DOCTYPE rdf:RDF [
  <!ENTITY owl "http://www.w3.org/2002/07/owl#" >
  <!ENTITY swrl "http://www.w3.org/2003/11/swrl#" >
  <!ENTITY swrlb "http://www.w3.org/2003/11/swrlb#" >
  <!ENTITY xsd "http://www.w3.org/2001/XMLSchema#" >
  <!ENTITY rdfs "http://www.w3.org/2000/01/rdf-schema#" >
  <!ENTITY govpe
"http://localhost:8080/c/publicador_rdf/despesas_govpe#" >
  <!ENTITY rdf "http://www.w3.org/1999/02/22-rdf-syntax-ns#" >
  <!ENTITY protege "http://protege.stanford.edu/plugins/owl/protege#" >
  <!ENTITY xsp "http://www.owl-ontologies.com/2005/08/07/xsp.owl#" >
  <!ENTITY loa2012 "http://vocab.e.gov.br/2012/08/loa2012#" >
]>

<rdf:RDF xmlns="http://localhost:8080/c/publicador_rdf/despesas_govpe#"
  xml:base="http://localhost:8080/c/publicador_rdf/despesas_govpe"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:govpe="http://localhost:8080/c/publicador_rdf/despesas_govpe#"
  xmlns:loa2012="http://vocab.e.gov.br/2012/08/loa2012#"
  xmlns:swrl="http://www.w3.org/2003/11/swrl#"
  xmlns:protege="http://protege.stanford.edu/plugins/owl/protege#"
  xmlns:xsp="http://www.owl-ontologies.com/2005/08/07/xsp.owl#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
  xmlns:swrlb="http://www.w3.org/2003/11/swrlb#"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
  <owl:Ontology
rdf:about="http://localhost:8080/c/publicador_rdf/despesas_govpe">
    <rdfs:comment>Ontologia que descreve as despesas do Governo do
Estado de Pernambuco reutilizando a ontologia do orcamento
brasileiro</rdfs:comment>
    <owl:imports
rdf:resource="http://vocab.e.gov.br/2012/08/loa2012"/>
  </owl:Ontology>

  <!--
  ////////////////////////////////////////////////////
  ////////////////////////////////////////////////////
  //
  // Annotation properties
  //
  ////////////////////////////////////////////////////
  ////////////////////////////////////////////////////
  -->

  <!--
```

```

////////////////////////////////////
////////////////////////////////////
//
// Datatypes
//
////////////////////////////////////
////////////////////////////////////
-->

<!-- http://www.w3.org/2001/XMLSchema#date -->

<rdfs:Datatype rdf:about="&xsd:date"/>

<!--
////////////////////////////////////
////////////////////////////////////
//
// Object Properties
//
////////////////////////////////////
////////////////////////////////////
-->

<!-- http://vocab.e.gov.br/2012/08/loa2012#temOrgao -->

<rdf:Description rdf:about="&loa2012;temOrgao">
  <rdfs:domain>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <rdf:Description
rdf:about="&loa2012;UnidadeOrcamentaria"/>
        <rdf:Description rdf:about="&govpe;UnidadeGestora"/>
      </owl:unionOf>
    </owl:Class>
  </rdfs:domain>
</rdf:Description>

<!-- http://localhost:8080/c/publicador_rdf/despesas_govpe#temCredor
-->

<owl:ObjectProperty rdf:about="&govpe;temCredor">
  <rdf:type rdf:resource="&owl;FunctionalProperty"/>
  <rdfs:comment rdf:datatype="&xsd:string">Propriedade (object
property) representa a relacao entre todo um empenho e um
credor</rdfs:comment>
  <rdfs:range rdf:resource="&govpe;Credor"/>
  <rdfs:domain rdf:resource="&govpe;Empenho"/>
</owl:ObjectProperty>

```

```

<!--
http://localhost:8080/c/publicador_rdf/despesas_govpe#temItemDeDespesa --
>

<owl:ObjectProperty rdf:about="&govpe;temItemDeDespesa">
  <rdf:type rdf:resource="&owl;FunctionalProperty"/>
  <rdfs:comment rdf:datatype="&xsd:string">Propriedade (object
property) que relaciona elementos da classe Empenho (domain) aos
elementos da classe siop:ItemDeDespesa (range).</rdfs:comment>
  <rdfs:range rdf:resource="&loa2012;ItemDeDespesa"/>
  <rdfs:domain rdf:resource="&govpe;Empenho"/>
</owl:ObjectProperty>

<!--
http://localhost:8080/c/publicador_rdf/despesas_govpe#temUnidadeGestora -
->

<owl:ObjectProperty rdf:about="&govpe;temUnidadeGestora">
  <rdfs:comment rdf:datatype="&xsd:string">Propriedade (object
property) representa a relacao entre todo empenho com uma unidade
gestora, que esta vinculada a algum orgao</rdfs:comment>
  <rdfs:domain rdf:resource="&govpe;Empenho"/>
  <rdfs:range rdf:resource="&govpe;UnidadeGestora"/>
</owl:ObjectProperty>

<!--
////////////////////////////////////
////////////////////////////////////
////////////////////////////////////
//
// Data properties
//
////////////////////////////////////
////////////////////////////////////
-->

<!-- http://vocab.e.gov.br/2012/08/loa2012#valorDotacaoInicial -->

<rdf:Description rdf:about="&loa2012;valorDotacaoInicial">
  <rdfs:domain>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <rdf:Description rdf:about="&loa2012;ItemDeDespesa"/>
        <rdf:Description rdf:about="&govpe;Empenho"/>
      </owl:unionOf>
    </owl:Class>
  </rdfs:domain>
</rdf:Description>

```

```

<!-- http://localhost:8080/c/publicador_rdf/despesas_govpe#cpfOuCnpj
-->

<owl:DatatypeProperty rdf:about="&govpe;cpfOuCnpj">
  <rdf:type rdf:resource="&owl;FunctionalProperty"/>
  <rdfs:domain rdf:resource="&govpe;Credor"/>
  <rdfs:range rdf:resource="&xsd:string"/>
</owl:DatatypeProperty>

<!--
http://localhost:8080/c/publicador_rdf/despesas_govpe#dataEmpenho -->

<owl:DatatypeProperty rdf:about="&govpe;dataEmpenho">
  <rdf:type rdf:resource="&owl;FunctionalProperty"/>
  <rdfs:domain rdf:resource="&govpe;Empenho"/>
  <rdfs:range rdf:resource="&xsd:date"/>
</owl:DatatypeProperty>

<!-- http://localhost:8080/c/publicador_rdf/despesas_govpe#nomeCredor
-->

<owl:DatatypeProperty rdf:about="&govpe;nomeCredor">
  <rdf:type rdf:resource="&owl;FunctionalProperty"/>
  <rdfs:domain rdf:resource="&govpe;Credor"/>
  <rdfs:range rdf:resource="&xsd:string"/>
</owl:DatatypeProperty>

<!--
http://localhost:8080/c/publicador_rdf/despesas_govpe#numeroEmpenho -->

<owl:DatatypeProperty rdf:about="&govpe;numeroEmpenho">
  <rdf:type rdf:resource="&owl;FunctionalProperty"/>
  <rdfs:domain rdf:resource="&govpe;Empenho"/>
  <rdfs:range rdf:resource="&xsd:string"/>
</owl:DatatypeProperty>

<!--
http://localhost:8080/c/publicador_rdf/despesas_govpe#temModalidade -->

<owl:DatatypeProperty rdf:about="&govpe;temModalidade">
  <rdf:type rdf:resource="&owl;FunctionalProperty"/>
  <rdfs:comment rdf:datatype="&xsd:string">Refere-se a modalidade
do empenho</rdfs:comment>
  <rdfs:domain rdf:resource="&govpe;Empenho"/>
  <rdfs:range rdf:resource="&xsd:string"/>
</owl:DatatypeProperty>

```

```

<!--
http://localhost:8080/c/publicador_rdf/despesas_govpe#tipoLicitacao -->

<owl:DatatypeProperty rdf:about="&govpe;tipoLicitacao">
  <rdf:type rdf:resource="&owl;FunctionalProperty"/>
  <rdfs:comment rdf:datatype="&xsd:string">Descricao do tipo da
licitacao pela qual foi gerado o empenho</rdfs:comment>
  <rdfs:domain rdf:resource="&govpe;Empenho"/>
  <rdfs:range rdf:resource="&xsd:string"/>
</owl:DatatypeProperty>

<!--
http://localhost:8080/c/publicador_rdf/despesas_govpe#valorLiquidado -->

<owl:DatatypeProperty rdf:about="&govpe;valorLiquidado">
  <rdf:type rdf:resource="&owl;FunctionalProperty"/>
  <rdfs:comment rdf:datatype="&xsd:string">Valor referente ao
empenho que foi liquidado</rdfs:comment>
  <rdfs:domain rdf:resource="&govpe;Empenho"/>
  <rdfs:range rdf:resource="&xsd:double"/>
</owl:DatatypeProperty>

<!--
////////////////////////////////////
////////////////////////////////////
//
// Classes
//
////////////////////////////////////
////////////////////////////////////
-->

<!-- http://vocab.e.gov.br/2012/08/loa2012#Classificador -->

<rdf:Description rdf:about="&loa2012;Classificador">
  <rdfs:subClassOf rdf:resource="&owl;Thing"/>
  <owl:disjointWith rdf:resource="&govpe;Credor"/>
  <owl:disjointWith rdf:resource="&govpe;Empenho"/>
</rdf:Description>

<!-- http://localhost:8080/c/publicador_rdf/despesas_govpe#Credor -->

<owl:Class rdf:about="&govpe;Credor">
  <owl:equivalentClass>
    <owl:Class>
      <owl:intersectionOf rdf:parseType="Collection">
        <owl:Restriction>
          <owl:onProperty rdf:resource="&govpe;cpfOuCnpj"/>

```

```

        <owl:someValuesFrom rdf:resource="&xsd:string"/>
    </owl:Restriction>
    <owl:Restriction>
        <owl:onProperty
rdf:resource="&govpe;nomeCredor"/>
        <owl:someValuesFrom rdf:resource="&xsd:string"/>
    </owl:Restriction>
</owl:intersectionOf>
</owl:Class>
</owl:equivalentClass>
<rdfs:comment rdf:datatype="&xsd:string">Classe que representa os
credores de um certo empenho</rdfs:comment>
</owl:Class>

<!-- http://localhost:8080/c/publicador_rdf/despesas_govpe#Empenho --
>

<owl:Class rdf:about="&govpe;Empenho">
    <owl:equivalentClass>
        <owl:Class>
            <owl:intersectionOf rdf:parseType="Collection">
                <owl:Restriction>
                    <owl:onProperty rdf:resource="&govpe;temCredor"/>
                    <owl:someValuesFrom
rdf:resource="&govpe;Credor"/>
                </owl:Restriction>
                <owl:Restriction>
                    <owl:onProperty
rdf:resource="&govpe;temItemDeDespesa"/>
                    <owl:someValuesFrom
rdf:resource="&loa2012;ItemDeDespesa"/>
                </owl:Restriction>
                <owl:Restriction>
                    <owl:onProperty
rdf:resource="&govpe;temUnidadeGestora"/>
                    <owl:someValuesFrom
rdf:resource="&govpe;UnidadeGestora"/>
                </owl:Restriction>
                <owl:Restriction>
                    <owl:onProperty
rdf:resource="&loa2012;valorDotacaoInicial"/>
                    <owl:someValuesFrom rdf:resource="&xsd:double"/>
                </owl:Restriction>
                <owl:Restriction>
                    <owl:onProperty
rdf:resource="&govpe;dataEmpenho"/>
                    <owl:someValuesFrom rdf:resource="&xsd:date"/>
                </owl:Restriction>
                <owl:Restriction>
                    <owl:onProperty
rdf:resource="&govpe;numeroEmpenho"/>
                    <owl:someValuesFrom rdf:resource="&xsd:string"/>
                </owl:Restriction>
            </owl:intersectionOf>
        </owl:Class>
    </owl:equivalentClass>
</owl:Class>

```

```

        <owl:onProperty
rdf:resource="&govpe;temModalidade"/>
        <owl:someValuesFrom rdf:resource="&xsd:string"/>
    </owl:Restriction>
    <owl:Restriction>
        <owl:onProperty
rdf:resource="&govpe;tipoLicitacao"/>
        <owl:someValuesFrom rdf:resource="&xsd:string"/>
    </owl:Restriction>
    <owl:Restriction>
        <owl:onProperty
rdf:resource="&govpe;valorLiquidado"/>
        <owl:someValuesFrom rdf:resource="&xsd:double"/>
    </owl:Restriction>
</owl:intersectionOf>
</owl:Class>
</owl:equivalentClass>
<rdfs:comment rdf:datatype="&xsd:string">Classe definida por
restricoes cujos individuos representam os empenhos realizados pelo
governo, . Cada elemento da classe Empenho é caracterizado por ter object
properties relacionadas a um ou varios itens de despesa e um credor.
Possui tambem as data properties numero, do tipo string, valor liquidado,
do tipo double, e data do tipo date.</rdfs:comment>
    </owl:Class>

<!--
http://localhost:8080/c/publicador_rdf/despesas_govpe#UnidadeGestora -->

    <owl:Class rdf:about="&govpe;UnidadeGestora">
        <owl:equivalentClass>
            <owl:Class>
                <owl:intersectionOf rdf:parseType="Collection">
                    <rdf:Description rdf:about="&loa2012;Classificador"/>
                    <owl:Restriction>
                        <owl:onProperty
rdf:resource="&loa2012;temOrgao"/>
                        <owl:someValuesFrom
rdf:resource="&loa2012;Orgao"/>
                    </owl:Restriction>
                </owl:intersectionOf>
            </owl:Class>
        </owl:equivalentClass>
        <rdfs:comment rdf:datatype="&xsd:string">Classe que representa
uma unidade gestora que está vinculada a um orgao atraves da propriedade
siop:temOrgao.</rdfs:comment>
    </owl:Class>
</rdf:RDF>

<!-- Generated by the OWL API (version 3.2.3.1824)
http://owlapi.sourceforge.net -->

```

## APÊNDICE B – Script SQL da Tabela de Triplas RDF

```
CREATE TABLE LAI_DIR_DESPESAS_INSTITUICAO_RDF (  
    LAI_DIR_ID BIGINT NOT NULL,  
    LAI_INS_ID BIGINT,  
    LAI_DIR_TRIPLAS TEXT NULL,  
    LAI_DIR_SERIALIZACAO TEXT NULL,  
    LAI_DIR_CAD_USER_ID BIGINT,  
    LAI_DIR_DATA_CAD timestamp without time zone,  
    LAI_DIR_MOD_USER_ID BIGINT,  
    LAI_DIR_DATA_MOD timestamp without time zone  
);  
  
ALTER TABLE LAI_DIR_DESPESAS_INSTITUICAO_RDF  
    ADD CONSTRAINT LAI_DIR_DESPESAS_INSTITUICAO_RDF_key PRIMARY KEY  
(LAI_DIR_ID);  
  
ALTER TABLE LAI_DIR_DESPESAS_INSTITUICAO_RDF  
    ADD CONSTRAINT LAI_DIR_DESPESAS_INSTITUICAO_RDF_instituicao_fkey FOREIGN  
KEY (LAI_INS_ID) REFERENCES LAI_INS_INSTITUICAO(LAI_INS_ID) ON UPDATE RESTRICT  
ON DELETE RESTRICT;  
  
COMMENT ON COLUMN LAI_DIR_DESPESAS_INSTITUICAO_RDF.LAI_DIR_ID IS 'Chave primaria  
da entidade';  
  
COMMENT ON COLUMN LAI_DIR_DESPESAS_INSTITUICAO_RDF.LAI_INS_ID IS 'Chave  
estrangeira que referencia a tabela LAI_INS_INSTITUICAO';  
  
COMMENT ON COLUMN LAI_DIR_DESPESAS_INSTITUICAO_RDF.LAI_DIR_TRIPLAS IS 'String  
com triplas RDF';  
  
COMMENT ON COLUMN LAI_DIR_DESPESAS_INSTITUICAO_RDF.LAI_DIR_SERIALIZACAO IS  
'Linguagem de serializacao RDF';  
  
COMMENT ON COLUMN LAI_DIR_DESPESAS_INSTITUICAO_RDF.LAI_DIR_CAD_USER_ID IS 'Usado  
para o log';  
  
COMMENT ON COLUMN LAI_DIR_DESPESAS_INSTITUICAO_RDF.LAI_DIR_DATA_CAD IS 'Usado  
para o log';  
  
COMMENT ON COLUMN LAI_DIR_DESPESAS_INSTITUICAO_RDF.LAI_DIR_MOD_USER_ID IS 'Usado  
para o log';  
  
COMMENT ON COLUMN LAI_DIR_DESPESAS_INSTITUICAO_RDF.LAI_DIR_DATA_MOD IS 'Usado  
para o log';
```