

**UNIVERSIDADE DO ESTADO DA BAHIA
DEPARTAMENTO DE CIÊNCIAS EXATAS E DA TERRA
COLEGIADO DE SISTEMAS DE INFORMAÇÃO**

**UMA SOLUÇÃO COMPUTACIONAL PARA
INTEGRAÇÃO ENTRE WEB SEMÂNTICA E REDES
COMPLEXAS**

Shankar Cabus de Teive e Argollo

Salvador
2012

**UNIVERSIDADE DO ESTADO DA BAHIA
DEPARTAMENTO DE CIÊNCIAS EXATAS E DA TERRA
COLEGIADO DE SISTEMAS DE INFORMAÇÃO**

**UMA SOLUÇÃO COMPUTACIONAL PARA
INTEGRAÇÃO ENTRE WEB SEMÂNTICA E REDES
COMPLEXAS**

Shankar Cabus de Teive e Argollo

Monografia apresentada à Banca Examinadora
como exigência parcial para obtenção do título
de bacharel em Sistemas Informação pela Uni-
versidade do Estado da Bahia.

**Orientador:
Eduardo Manuel de Freitas Jorge**

**Salvador
2012**

SHANKAR CABUS DE TEIVE E ARGOLLO

Uma solução computacional para integração entre Web Semântica e Redes Complexas

Monografia apresentada à Banca Examinadora
como exigência parcial para obtenção do título de
bacharel em Sistemas de Informação pela Univer-
sidade do Estado da Bahia.

Orientador. Eduardo Manuel de Freitas Jorge

BANCA EXAMINADORA

Prof. Eduardo Manuel de Freitas Jorge

Doutor em Difusão do Conhecimento

Universidade do Estado da Bahia

Prof. Alexandre Rafael Lenz

Doutorando em Ciência da Computação

Universidade do Estado da Bahia

Prof. Uedson Santos Reis

Mestre em Modelagem Computacional

Serviço Nacional de Aprendizagem Industrial

Agradecimentos

Lista de Figuras

1	Diagrama <i>Linking Open Data</i> em Maio de 2007	p. 9
2	Diagrama <i>Linking Open Data</i> em Setembro de 2011	p. 9
3	Janela do SemanticWebImport com consulta SPARQL	p. 14
4	Gráfico gerado pelo Welkin	p. 15
5	Arquitetura do RDFree	p. 17
6	Representação gráfica da rede semântica gerada pelo Gephi	p. 21

Lista de Algoritmos

3.1	Exemplo de SPARQLWrapper	p. 18
3.2	Estrutura normalizada	p. 18
3.3	SPARQL utilizado para consultar bandas e seus gêneros	p. 19
3.4	Arquivo de configuração	p. 20

Lista de Tabelas

1	Tabela comparativa entre trabalhos relacionados	p. 16
2	Requisitos funcionais e não funcionais	p. 16
3	Exemplo de resultado de uma consulta	p. 18

Sumário

1	Introdução	p. 8
2	?	p. 11
3	Análise e Projeto RDFree	p. 12
3.1	Trabalhos Relacionados	p. 13
3.2	Requisitos e Projeto Arquitetural	p. 16
3.3	Projeto de Baixo Nível	p. 17
3.3.1	Módulo de Extração	p. 18
3.3.2	Módulo de Normalização	p. 18
3.3.3	Módulo de Conversão	p. 19
3.4	Aplicando a solução em um domínio	p. 19
3.5	Análise da solução	p. 21
	Referências	p. 22

1 *Introdução*

Desde o surgimento da primeira rede de computadores em 1960, a internet vem sofrendo transformações no intuito de facilitar a produção, compartilhamento e organização da informação. Novos conceitos de design e usabilidade, novas ferramentas de navegação e publicação, facilidade de acesso, velocidade na transmissão de dados, são características da Web atual, que têm o objetivo de tornar o uso da internet cada vez mais prazeroso e produtivo.

Com a facilidade para compartilhar e produzir conteúdo, o volume de informação na Web cresce diariamente numa escala cada vez maior. Logo, o desafio da Web é facilitar a recuperação dessas informações. Existem ferramentas de busca como Google, Bing e Yahoo! que se propõem a resolver este problema. Seu funcionamento básico consiste em navegar entre links da web, indexar o conteúdo das páginas visitadas e, de acordo com uma busca feita por um usuário, retornar as páginas mais relevantes. Mas a máquina não consegue interpretar a consulta, apenas aplicar regras em cima de um conjunto qualquer de palavras desconexas. Por isso, buscas mais complexas, que envolvem muitos cruzamentos de dados, diminuem consideravelmente a eficácia desse método de recuperação de dados.

Pensando no problema da organização dos dados, Tim Berners-Lee apresenta, em 2001, o termo Web Semântica, que tem como objetivo transferir a responsabilidade de interpretação da informação, do usuário para a máquina (BREITMAN, 2005). Berners-Lee propõe a estruturação das informações de forma semântica, utilizando metadados e modelos semânticos para interligá-las. A essa nova forma de publicar e interligar os dados, Berners-Lee deu o nome de *Linked Data*, partindo da ideia de que o valor e uso dos dados crescem quanto mais eles estiverem interligados (BIZER et al, 2008). Essa nova estrutura possibilita que a máquina interprete os dados, facilitando a organização e consequentemente, a recuperação das informações.

Em Janeiro de 2007 surge o projeto Linking Open Data, com o objetivo de desenvolver *Linked Data* a partir da identificação de dados existentes sob licença livre e convertê-los para o modelo de dados da Web Semântica, o RDF (*Resource Description Framework*). O projeto que começou de forma lenta, vem crescendo a cada ano, passando de 12 conjuntos de dados

em 2007 para 295 em Setembro de 2011. As figuras 1 e 2 mostram a representação gráfica do projeto em 2007 e em 2011, respectivamente.

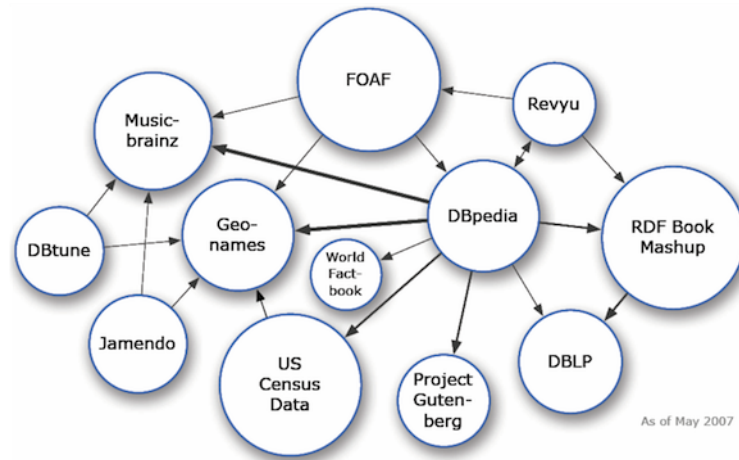


Figura 1: Diagrama *Linking Open Data* em Maio de 2007

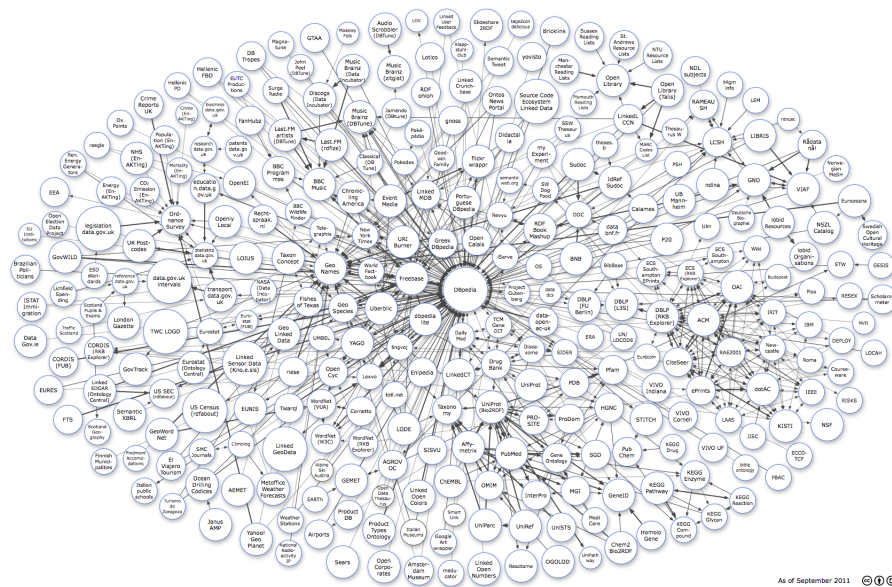


Figura 2: Diagrama *Linking Open Data* em Setembro de 2011

As relações entre os dados do Linked Data podem ser representadas por meio de redes complexas, que consistem em um conjunto de vértices ou nodos ligados por linhas chamadas arestas. Através de ferramentas de análise de redes, é possível fazer uma interpretação gráfica e matemática de propriedades que caracterizam as redes complexas. Contudo, até o presente momento não foram encontradas ferramentas ou protocolos que auxiliem na conversão dos dados no formato RDF para uma entrada legível por estas ferramentas de análise de redes.

Desta forma, este projeto tem por objetivo propor um modelo computacional, chamado de

RDFree, para extrair os dados em RDF de bases de dados Linked Data e convertê-los para um formato possível de ser interpretado por ferramentas de análise de redes complexas, integrando os dois campos de pesquisa supracitados: Web Semântica e Redes complexas.

Espera-se alcançar tal objetivo através das seguintes metas ou objetivos específicos:

- Criar uma arquitetura genérica com os elementos de toda a solução do modelo computacional proposto.
- Criar um componente para leitura de dados no padrão RDF, possibilitando a definição de parâmetros que permitam a filtragem e indicação de elementos que são vértices e arestas numa rede semântica.
- Gerar um arquivo de texto num padrão interpretável por softwares de análise de redes semântica.

O trabalho será validado pelo uso do modelo computacional proposto em um determinado conjunto de dados RDF, convertendo os dados para um arquivo de entrada de softwares de análise de redes e, por fim, gerar métricas que caracterizam e definem a topologia da rede utilizada.

Diferentes campos de estudos científicos e sociais serão beneficiados com este trabalho que se propõe a auxiliar na coleta, processamento e análise de dados espalhados pela Web Semântica, provendo subsídios para que pesquisadores possam interpretar os dados e utilizá-los em duas pesquisas.

2 ?

3 *Análise e Projeto RDFree*

Com novas soluções tecnológicas para facilitar a geração de conteúdo na internet e torná-la mais rápida e dinâmica, surgiram também novos problemas. Um deles é o problema da organização e recuperação das informações, que estão espalhadas pela internet e à mercê de motores de buscadores para serem recuperadas. Mas a depender da complexidade da condição de recuperação, esses buscadores convencionais têm sua eficácia reduzida, pois seu funcionamento se baseia pela frequência e densidade que os termos buscados aparecem numa determinada página da internet.

Focado no problema da organização e recuperação dos dados, a Web Semântica se propõe a organizar e facilitar a recuperação das informações, adicionando uma nova camada de dados sobre os dados, os metadados. Estes metadados criam uma rede de informação, ligando sujeitos a objetos por meio de predicados, tal como a oração "Metallica é uma banda de Rock", onde "Metallica" é o sujeito, "é uma banda de" é o predicado, e "Rock" é o objeto.

Na Web Semântica, as informações estão interligadas e podem ser representada graficamente como uma rede semântica, onde o sujeito e o objeto são os vértices e o predicado a aresta. Existem no mercado softwares de análise de redes que se propõe a gerar o gráfico de forma personalizada, bem como calcular métricas que caracterizam a rede, como grau de complexidade, clusterização, densidade, e outras. Esses *softwares* funcionam a partir de uma entrada de dados num formato conhecido, que varia de acordo com o *software*.

Apesar de existirem muitas ferramentas de análises de rede, este campo de estudo ainda carece de um modelo computacional que faça algumas etapas preliminares: a de consultar e extrair os dados a partir de uma base de dados e depois torná-los legíveis para estas ferramentas.

Portanto, este trabalho tem como objetivo desenvolver uma ferramenta de extração e conversão de dados RDF para um formato interpretável por *softwares* de análise de redes. Esta ferramenta foi denominada RDFree, nome proveniente da junção entre o acrônimo RDF e a palavra em inglês *free* (livre).

Com a utilização do RDFree, pesquisadores da área de redes semânticas podem gerar fa-

cilmente o gráfico e as métricas da rede, a partir dos dados presentes na *Linked Data*, e utilizar esses dados como subsídio para suas pesquisas.

Nas próximas seções será apresentada a ferramenta de conversão desenvolvida, dando uma visão mais aprofundada sobre o seu funcionamento e as etapas que foram necessárias para o alcance do objetivo fulcral deste trabalho. [sumarizar os itens das próximas seções]

Para nortear este projeto, foram analisados trabalhos relacionados que o oferecessem recursos de consulta SPARQL ou geração de gráficos de rede. Na próxima seção, cada projeto é apresentado e comparado com os demais em uma tabela.

3.1 Trabalhos Relacionados

Na busca por soluções semelhantes ao objetivo deste trabalho, que pudessem contribuir na especificação do RDFree, foram encontrados alguns projetos no domínio da web semântica e redes complexas. Esta pesquisa contribuiu para concepção do RDFree e confirmou a deficiência por ferramentas de extração e conversão de dados para um formato reconhecível por softwares de análise de redes.

Uma das mais utilizadas ferramentas de análise de redes é o *software open-source* Gephi, que foi apelidado pela comunidade como o *Photoshop*¹ para gráficos. Segundo a descrição feita pelo site oficial², o Gephi é uma ferramenta de visualização interativa e uma plataforma de exploração de todos os tipos de redes e sistemas complexos, dinâmicos e gráficos hierárquicos, capaz de ser executado nos sistemas operacionais Windows, Linux e Mac OS X.

O Gephi, além de suas ferramentas nativas, possui ainda alguns *plugin*³, como é o caso do SemanticWebImport, um plugin desenvolvido pela Wimmics⁴, que permite a importação de dados semânticos para o Gephi. Os dados importados são obtidos a partir de uma consulta SPARQL em uma base de dados semântica (RDF).

Diferente do RDFree, que tem o propósito de ser uma ferramenta genérica e modularizada, o SemanticWebImport é limitado ao Gephi, sendo incapaz de funcionar de forma independente ou com outras ferramentas. Outro ponto a ser observado na pesquisa é que o resultado obtido com este plugin não foi o esperado. A partir de uma mesma consulta SPARQL a base de dados da DBpedia, os resultados obtidos com o SemanticWebImport não foram compatíveis

¹Software líder do mercado dos editores de imagem

²<https://gephi.org>

³Complementos que podem ser incorporados à ferramenta

⁴<http://wimmics.inria.fr/>

com os resultados obtidos a partir de uma consulta direta à base de dados por meio do Virtuoso SPARQL⁵. Essa inconsistência dos dados não é encontrada no RDFree, que obteve resultados idênticos ao esperado.

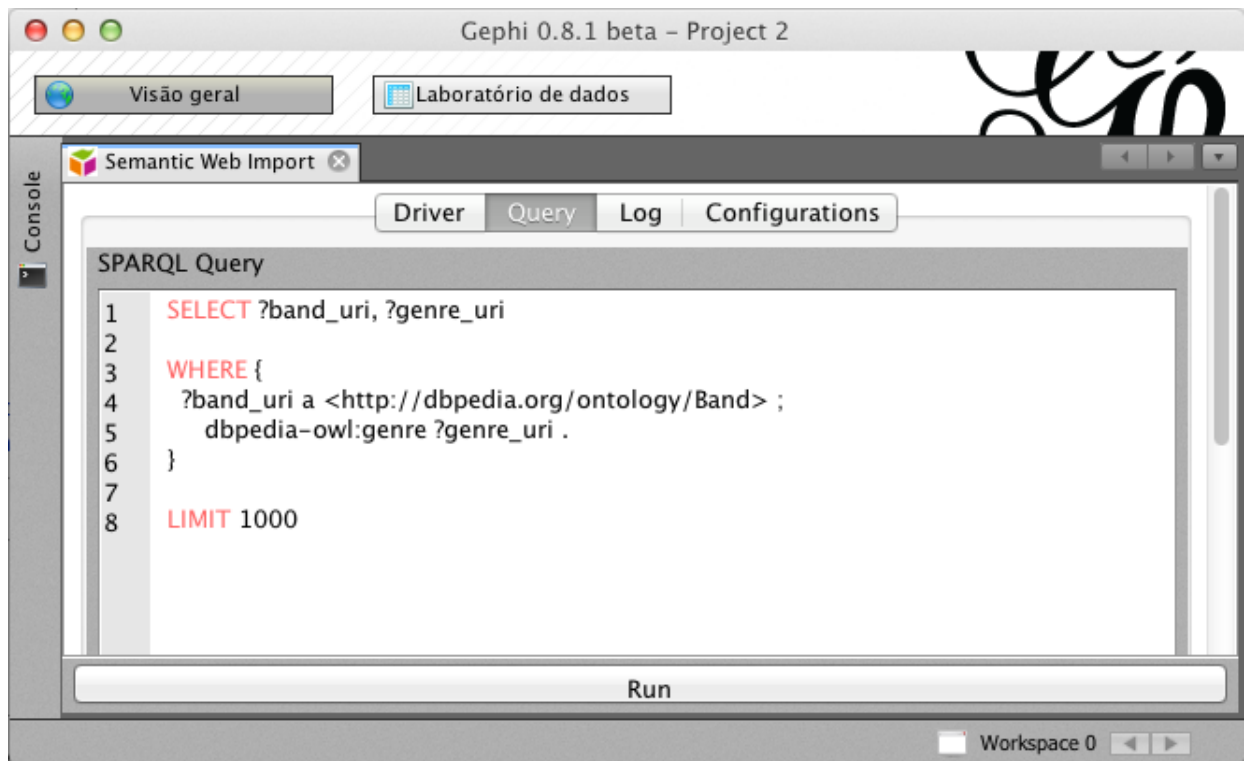


Figura 3: Janela do SemanticWebImport com consulta SPARQL

Uma ferramenta muito completa encontrada neste estudo foi a *TopBraid Composer*. Implementada como um plugin do Eclipse, ela serve como um ambiente de modelagem e edição de ontologias, construção de aplicações semântica, criação de gráficos e ainda, como o site oficial o descreve, "a melhor ferramenta SPARQL no mercado". O *TopBraid Composer* é distribuído em três versões: Free, Standard e Maestro, sendo esta última a mais completa e licenciada por U\$3.450,00. Contudo, este software está mais voltado para criação e modelagem de ontologias, sendo incapaz de fazer uma consulta remota e converter os dados para um formato desejado.

Mais simples que a *TopBraid Composer*, o software Welkin, open-source desenvolvido em 2006 no MIT (*Massachusetts Institute of Technology*), tem o objetivo de ser um visualizador de modelos RDF. Seu *input* está limitado a arquivos do tipo RDF, RDFS, OWL, n3 ou turtle, não sendo capaz de realizar consultas SPARQL em base de dados remotas ou locais. Suas funcionalidades também são reduzidas, limitando-se à criação do gráfico da rede e dos gráficos de grau de entrada, grau de saída e coeficiente de clusterização, como é possível ver na figura 4.

⁵Interface utilizada pela DBpedia para fazer consultas SPARQL

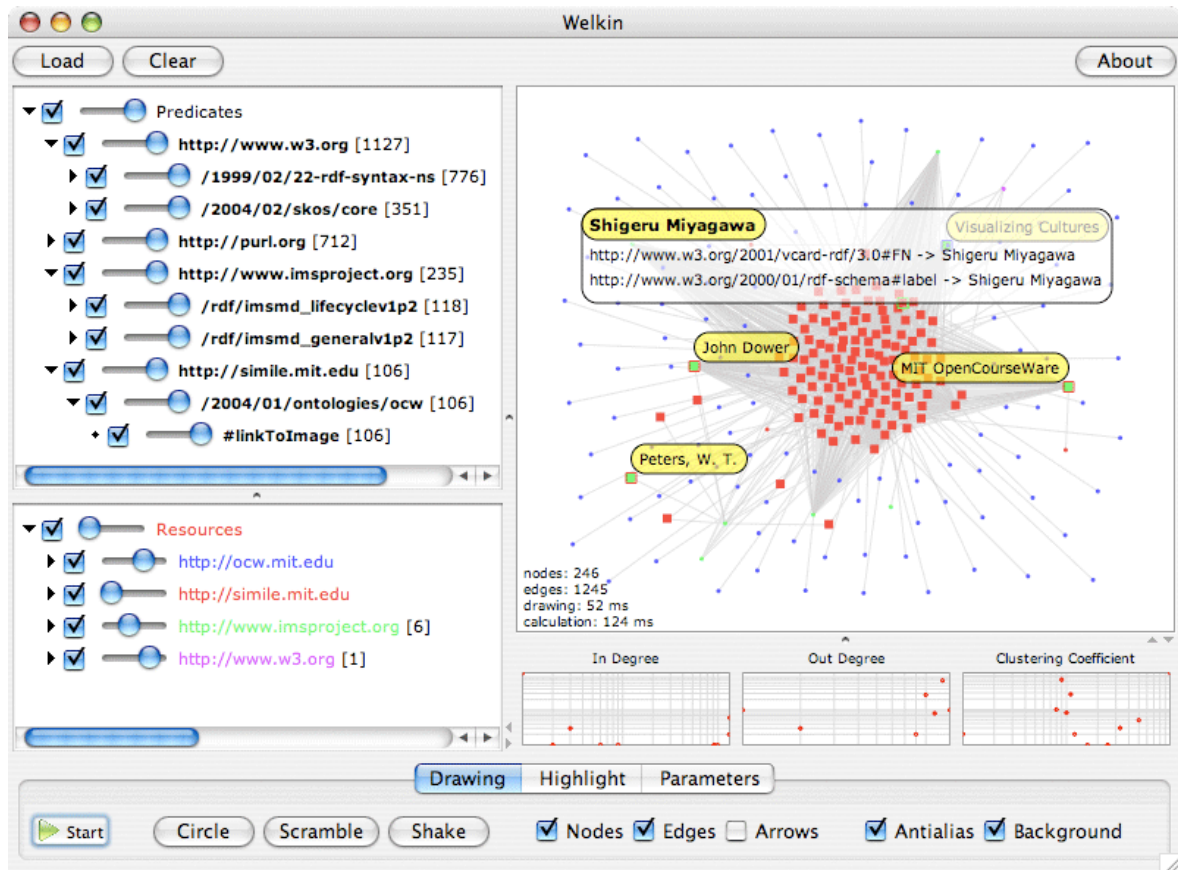


Figura 4: Gráfico gerado pelo Welkin

Assim como o *Welkin*, outra ferramentas se propõe a ser um visualizador de modelo RDF, tais como *visualRDF*, *Visual Browser*, *RDF Gravity* e *IsaViz*. Todas estas encontram-se desatualizadas e com última versão datada de 2004 à 2007.

Vale citar ainda a *Force-Directed Graph*, uma ferramenta online escrita em *JavaScript* e que, entre todas as ferramentas estudadas, é a que se encontra mais atualizada (11/2012). Seu objetivo limita-se à representação gráfica da rede a partir de uma entrada de dados padrão no formato JSON. Apesar do seu propósito estar mais próximo do *Gephi* e mais distante do objetivo deste trabalho, o *Force-Directed Graph* serviu como inspiração para modelagem do *RDFree*, que também utiliza um arquivo JSON como normalizador dos dados.

Concluída a pesquisa por trabalhos correlatos, foi feita uma comparação entre eles e o *RDFree*, como observado na tabela 1. Em seguida, na próxima seção são descritos os requisitos que nortearam a concepção do *RDFree*.

Tabela 1: Tabela comparativa entre trabalhos relacionados

	Open Source	Consulta SPARQL	Gera gráfico	Exporta para outros formatos
Semantic Web Import		x		
TopBraid Composer		x	x	
Welkin	x		x	
Force-Directed Graph	x		x	
RDFree	x	x		x

3.2 Requisitos e Projeto Arquitetural

A análise de trabalhos correlatos foi importante no processo de levantamento de requisitos funcionais (RF) e não funcionais (RNF) do RDFree, como pode ser visto na tabela 2.

Tabela 2: Requisitos funcionais e não funcionais

Requisito	Descrição
RF1	A aplicação deve conseguir importar dados de uma consulta SPARQL a uma base de dados RDF remota ou local.
RF2	Os dados devem ser preparados e normalizados para um formato padrão.
RF3	Os dados normalizados e convertidos devem ser salvos em um arquivo.
RNF1	As definições de configuração do conversor devem estar em um arquivo separado do código.
RNF2	A aplicação deve dar suporte a inclusão de novos módulos de tradução.

O projeto arquitetural deste trabalho é ilustrado pela figura 5 em forma de fluxograma, mostrando as etapas realizadas pela aplicação desde a sua inicialização, até a criação do arquivo de saída.

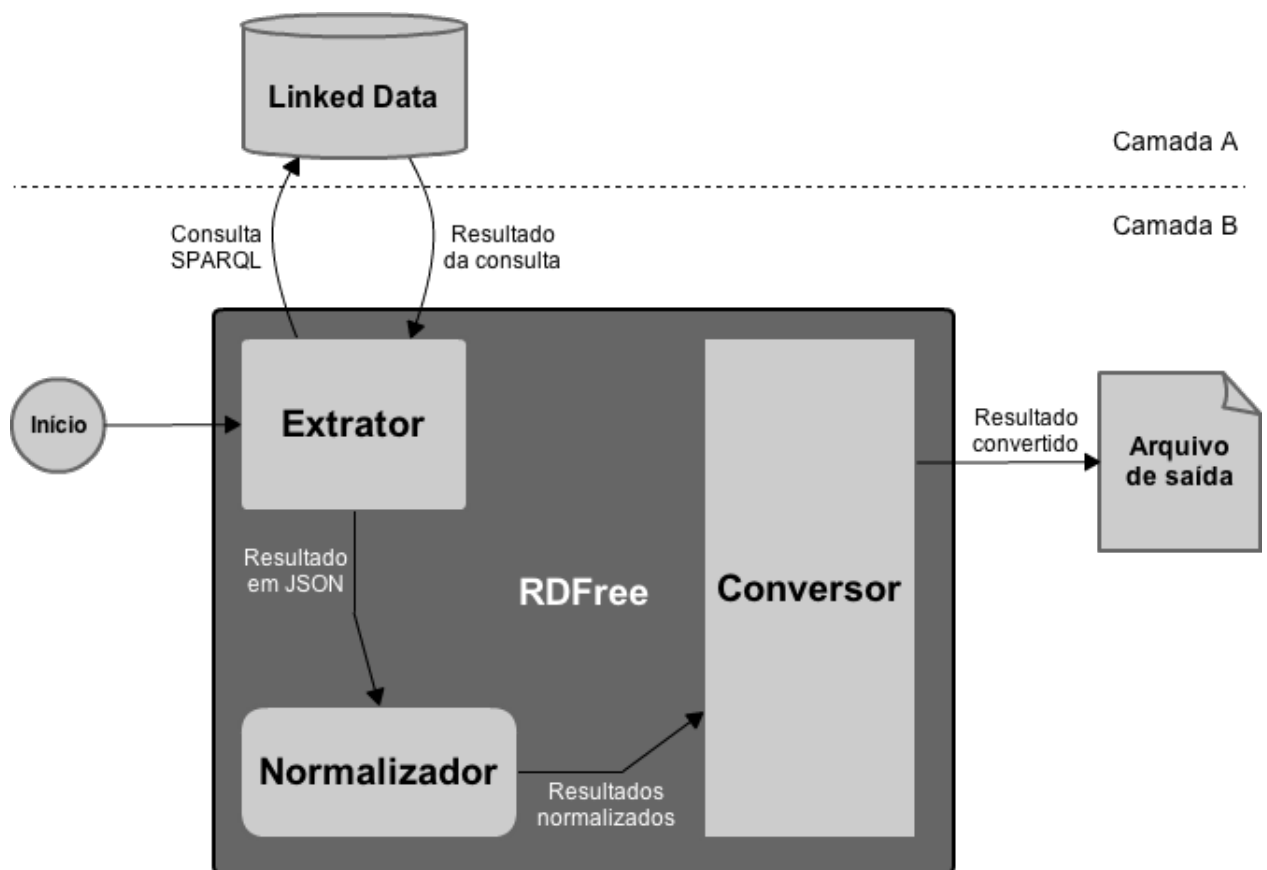


Figura 5: Arquitetura do RDFree

Na camada (A) está a base de dados RDF, que pode ser remota ou local e que tem a função de retornar o resultado de uma consulta para a camada (B), onde estão os três módulos que compõe a ferramenta RDFree:

- **Extrator** - é o módulo responsável por se comunicar com a base de dados, executar a consulta e importar os resultados para a aplicação, contemplando o requisito RF1.
- **Normalizador** - executa passos comuns à todos os conversores, criando uma estrutura de dados já com os nós e arestas da rede, satisfazendo o requisito RF2.
- **Conversor** - este módulo funciona como um tradutor, recebendo como entrada os dados padronizados, gerados pelo normalizador, e retornando um formato de acordo com sua especificação. Atende aos requisitos RF3 e RNF2.

Para compreender melhor o funcionamento e interação desses módulos, na próxima seção é mostrada a solução computacional utilizada na criação de cada módulo, apresentando seu funcionamento de forma mais detalhada, bem como as tecnologias empregadas no projeto.

3.3 Codificação da Solução

O RDFree foi desenvolvido utilizando a linguagem de programação Python, portanto, os trechos de códigos encontrados nesta seção, bem como as bibliotecas utilizadas, estão escritos nesta mesma linguagem.

O módulo de extração inicia a execução do RDFree e é responsável por importar os dados que serão convertidos. A conexão com o endpoint e execução da consulta SPARQL são feitas com o auxílio do framework SPARQLWrapper, que provê uma interface amigável e simples para o programador, como pode ser visto no código 3.1.

Algoritmo 3.1: Exemplo de SPARQLWrapper

```

1 sparql = SPARQLWrapper("http://dbpedia.org/sparql")
2 sparql.setQuery("SELECT * WHERE { ?s ?p ?o } LIMIT 10")
3 sparql.setReturnFormat(JSON)
4 results = sparql.query().convert()

```

No código 3.1, na primeira linha é instanciado um objeto SPARQLWrapper que fará uma consulta à DBpedia. Na linha 2 é determinada a consulta SPARQL que será realizada. Esta consulta retorna o sujeito, predicado e objeto dos dez primeiros resultados. Na linha 3 é determinado o formato dos dados recebidos, neste caso é utilizado o JSON por ser de fácil manipulação no Python. Por fim, na linha 4, a consulta é realizada e os resultados convertidos para JSON.

Ao receber os resultados da consulta, o módulo de extração converte os resultados para o formato JSON, que é um formato de fácil manipulação no Python.

Após a extração, o resultado é normalizado para um formato genérico, que auxiliará na próxima etapa de conversão. Este novo formato, também em JSON, é composto por duas listas: uma de vértices ou nós e outra de arestas. A lista dos vértices é formada por todos os recursos (sujeitos e objetos) únicos presentes na consulta e seguidos de um identificador numérico, que será usado na lista de arestas, formada por pares de relacionamento entre os vértices.

No exemplo de uma consulta pelos filhos de Francisco, o resultado pode ser o mostrado na tabela 3 e o JSON desse resultado representado no código 3.2.

Tabela 3: Exemplo de resultado de uma consulta

Pai	Filho
Francisco	Mirosmar
Francisco	Emival

Algoritmo 3.2: Estrutura normalizada

```

1  # Vertices
2  {
3    "pai" : {
4      "Francisco" : {"id" : 1}
5    },
6    "filho" : {
7      "Mirosmar" : {"id" : 2},
8      "Emival" : {"id" : 3},
9    }
10 }
11
12 # Arestas
13 [
14   [1, 2],
15   [1, 3]
16 ]

```

O módulo de conversão é responsável por transformar os dados normalizados para um padrão interpretável por ferramentas de análise de rede. Até a conclusão deste trabalho, foi criado apenas o módulo de conversão para o formato GEXF (Graph Exchange XML Format), uma linguagem criada em 2007 juntamente com o Gephi, com o objetivo de padronizar a descrição de estruturas complexas de redes, e portanto, o formato de entrada utilizado pelo Gephi. O projeto GEXF possui ainda bibliotecas para Python, Java, C++, JavaScript e Perl que facilitam na criação do arquivo.

3.4 Aplicando a Solução em um Domínio

Para mostrar o funcionamento da ferramenta desenvolvida neste trabalho, será utilizada a base de dados da DBpedia em Português, para buscar por bandas de música e seu respectivo gênero musical. Para fins didáticos, a consulta será limitada a mil resultados, que é equivalente a mil relacionamentos. A consulta SPARQL utilizada para definir o domínio estudado pode ser

visto no código 3.3.

Algoritmo 3.3: SPARQL utilizado para consultar bandas e seus gêneros

```

1 SELECT ?band_uri , ?genre_uri
2 WHERE {
3     ?band_uri a <http://dbpedia.org/ontology/Band> ;
4     dbpedia-owl:genre ?genre_uri .
5 }
6 LIMIT 1000

```

Neste exemplo, será utilizado o Gephi para gerar a representação gráfica dos dados, por isso o arquivo gerado será no formato GEXF.

A consulta SPARQL mostrada no código 3.3, bem como a URI da DBpedia devem estar no arquivo de configuração, como o mostrado no código 3.4. Feito isso, a aplicação já pode ser iniciada executando o arquivo main.py, que salvará o resultado no arquivo output.gexf.

Algoritmo 3.4: Arquivo de configuração

```

1 OUTPUT_FILE = 'output.gexf'
2 ENDPOINT = "http://dbpedia.org/sparql"
3 QUERY = '''
4     SELECT ?band_uri , ?genre_uri
5     WHERE {
6         ?band_uri a <http://dbpedia.org/ontology/Band> ;
7         dbpedia-owl:genre ?genre_uri .
8     }
9     LIMIT 1000
10 '''
11
12 # GEXF
13 CREATOR = 'Shankar Cabus de Teive e Argollo'
14 DESCRIPTION = u'Aplicando a solucao em um dominio'
15 DEFAULTEDGETYPE = 'undirected'
16 MODE = 'static'
17 LABEL = 'Relacionamento entre bandas e genero'

```

Abrindo o arquivo output.gexf no Gephi é possível encontrar o grau médio da rede (1,727), densidade do grafo (0.003), modularidade (0,667), entre outras métricas. Ao abrir o Gephi, já é exibido um gráfico gerado de forma aleatória, mas para efeito didático, na figura 6 é mostrado o gráfico da rede distribuído utilizando o algoritmo de Fruchterman Reingold e classificado por excentricidade, onde os pontos vermelhos são os gêneros musicais e os azuis são as bandas.

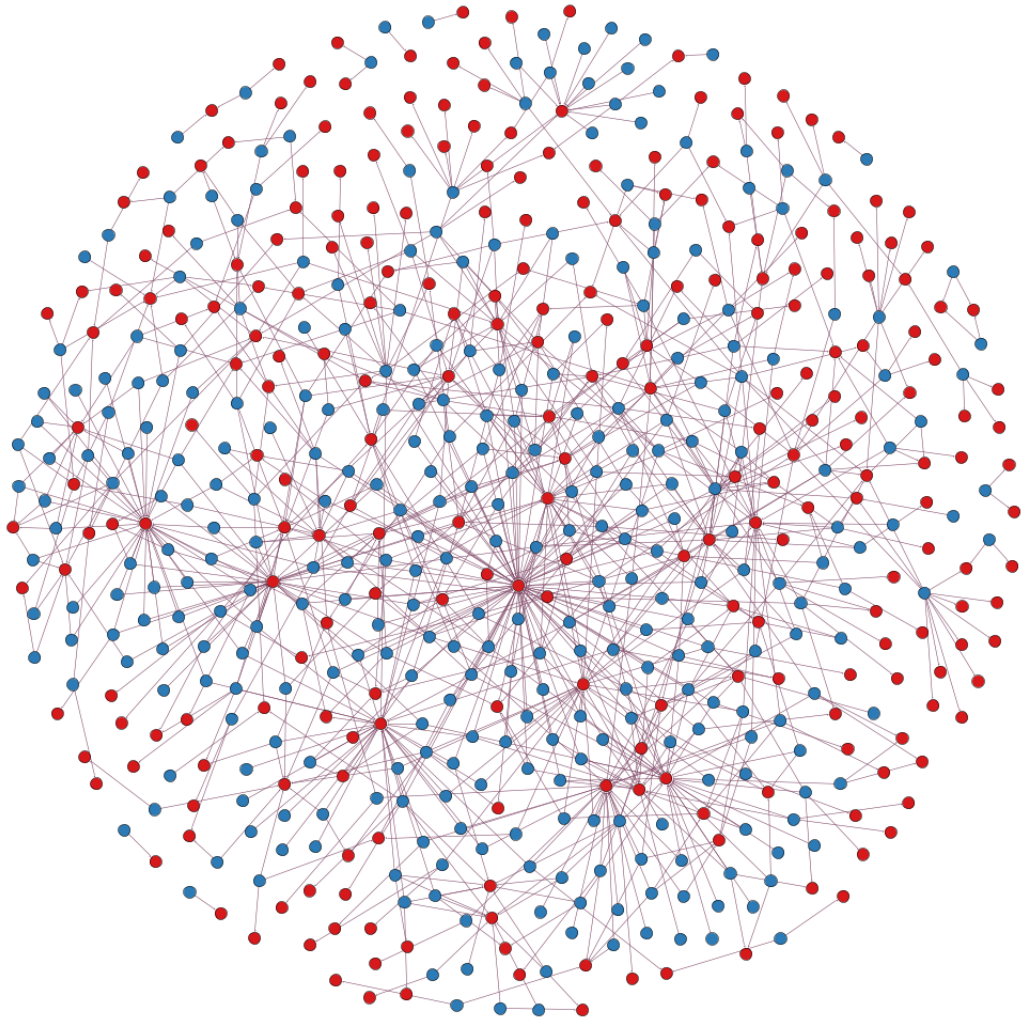


Figura 6: Representação gráfica da rede semântica gerada pelo Gephi

3.5 Análise da solução

Conforme foi mostrado na seção 3.4, o RDFree atendeu à todos os requisitos definidos neste trabalho, executando a consulta a uma base de dados RDF e exportando os dados para um formato aceito por uma ferramenta de análise de redes.

Contudo, o RDFree carece de uma interface amigável para usuários que não tenham conhecimento sobre a linguagem de programação utilizada no desenvolvimento do projeto.

Referências