



UNIVERSIDADE FEDERAL DO ESTADO DO RIO DE JANEIRO  
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA  
ESCOLA DE INFORMÁTICA APLICADA

ENRIQUECIMENTO DE UM *DATA MART* ATRAVÉS DE DADOS  
ESTRUTURADOS RDF DA *WEB*

Rafael Ferreira e Paulo Gabriel Castro

**Orientador**  
Sean Siqueira

RIO DE JANEIRO, RJ – BRASIL

JULHO DE 2013

ENRIQUECIMENTO DE UM *DATA MART* ATRAVÉS DE DADOS  
ESTRUTURADOS RDF DA *WEB*

Rafael Ferreira e Paulo Gabriel Castro

Projeto de Graduação apresentado à Escola de  
Informática Aplicada da Universidade Federal do  
Estado do Rio de Janeiro (UNIRIO) para obtenção do  
título de Bacharel em Sistemas de Informação.

Aprovada por:

---

Prof. Sean Wolfgang Matsui Siqueira, Orientador (UNIRIO)

---

Prof. Luiz Carlos Montez Monte (UNIRIO)

---

Prof. Kate Cerqueira Revoredo (UNIRIO)

RIO DE JANEIRO, RJ – BRASIL.

AGOSTO DE 2013

## **Agradecimentos**

A todas as pessoas que nos incentivaram e nos apoiaram durante essa jornada, em especial às nossas famílias e amigos, e ao orientador Sean Siqueira, por sua dedicação, ideias e orientação crítica, nos ajudando nos mínimos detalhes para a conclusão do estudo.

## RESUMO

---

O ambiente de dados para suporte ao negócio e tomada de decisão é fundamentalmente diferente do ambiente convencional de processamento de transações. A base deste ambiente é o do *Data Warehouse* (DW), integrando e consolidando dados disponíveis em diferentes acervos para fins de exploração e análise, ampliando o conteúdo informacional destes acervos para atender às expectativas e necessidades de nível estratégico na empresa. Esta monografia tem por objetivo apresentar o *Data Warehouse*, introduzindo os principais conceitos na área, discutir rapidamente as diferenças deste ambiente para o ambiente transacional e mostrar como enriquecer um *Data Mart*, um subconjunto de dados de uma determinada área negócio do *Data Warehouse*, através de dados estruturados RDF oriundos da *web*, a fim de complementar a informação já existente e possibilitar análises mais profundas e completas.

**Palavras-chave:** *data warehouse*, *data mart*, dados estruturados, rdf, sparql, *web*.

## ***ABSTRACT***

---

*The data environment for business support and decision making is fundamentally different from the conventional transaction processing environment. The core component of this environment is the Data Warehouse (DW), integrating and consolidating the data available from different collections for exploration and analysis, increasing the information content of these collections to meet the expectations and needs at a strategic level in the company. This work presents the Data Warehouse, introducing the main concepts in the field, briefly discussing the differences between the data warehouse environment and the transactional environment and showing how to enrich the Data Mart, a subset of data from a particular business area of Data Warehouse, environment through structured RDF data from the web in order to complement the existent information and to make possible deeper and more complete analysis.*

**Keywords:** *data warehouse, data mart, structured data, rdf, sparql, web.*

# Sumário

1	Introdução .....	9
1.1	Motivação.....	9
1.2	Objetivos .....	10
1.3	Organização do texto .....	11
2	Data Warehouse .....	12
2.1	Origem .....	12
2.2	Características.....	14
2.2.1	Orientado por assunto .....	14
2.2.2	Integrado.....	14
2.2.3	Não-Volátil.....	15
2.2.4	Variante no Tempo .....	16
2.2.5	Granularidade .....	16
2.2.6	Normalização / Desnormalização .....	17
2.2.7	Metadados .....	18
2.3	Arquitetura e Componentes.....	18
2.3.1	Sistema Transacional ( <i>Operational Source Systems</i> ) .....	19
2.3.2	Área de <i>Staging</i> ( <i>Data Staging Area</i> ) .....	19
2.3.3	Área de Apresentação de Dados ( <i>Data Presentation Area</i> ).....	20
2.3.4	Área de Acesso aos Dados ( <i>Data Access Tools</i> ) .....	20
2.4	Modelagem de dados em um <i>Data Warehouse</i> .....	21
2.4.1	Modelagem Multidimensional.....	21
2.4.2	Fatos .....	21
2.4.3	Dimensões .....	22
2.4.4	Medidas .....	23
2.4.5	Esquema Estrela.....	23

2.4.6 Esquema Floco de Neve.....	24
2.5 Considerações Finais .....	25
3 Web Semântica e Dados Ligados.....	26
3.1 <i>Web</i> Semântica .....	26
3.2 RDF.....	27
3.3 Dados ligados abertos .....	30
3.4 DBpedia.....	31
3.5 SPARQL .....	34
3.6 Considerações Finais .....	35
4 Abordagens de Enriquecimento do DW.....	36
4.1 Formas de extração de dados RDF .....	36
4.1.1 <i>Websites</i> tradutores de páginas HTML para RDF .....	36
4.1.2 Consultas SPARQL .....	37
4.1.3 Aplicações Pré-definidas.....	39
4.2 Alternativas de implementação dos dados extraídos em RDF no DW.....	40
4.2.1 Nova dimensão .....	41
4.2.2 Incremento de dimensão já existente .....	41
4.2.3 Incremento através de Metadados.....	42
4.3 Considerações Finais .....	42
5 Estudo de Caso: <i>Data Mart</i> de Telecom .....	43
5.1 Introdução.....	43
5.2 Modelo de Dados.....	44
5.3 Tabelas .....	46
5.2.1 Dimensão Canal Venda .....	47
5.2.2 Dimensão Tipo Recarga .....	47
5.2.3 Dimensão Faixa Horária .....	48
5.2.4 Dimensão Atendente Call Center.....	48

5.2.5 Dimensão Estados .....	49
5.2.6 Fato Recarga .....	50
5.4 Considerações Finais .....	51
6 Enriquecendo o DM de Telecom .....	52
6.1 Tipo de Extração Escolhida.....	52
6.2 Por que as outras formas de extração não foram escolhidas? .....	53
6.3 Ferramenta ETL Usada .....	55
6.4 Técnica de implementação dos dados na FATO. ....	56
6.5 Novo Esquema.....	57
6.6 Considerações Finais .....	59
7 Conclusão.....	60
7.1 Contribuições.....	60
7.2 Trabalhos Futuros .....	60
7.3 Limitações do Estudo.....	61

## Lista de Figuras

Figura 1 - Integração no <i>Data Warehouse</i> . Traduzido (INMON, 2005). .....	15
Figura 2 - Determinar o nível de granularidade é a questão mais importante do projeto no ambiente de data warehouse. Traduzido (INMON, 2005). .....	17
Figura 3 - Componentes do <i>Data Warehouse</i> . Traduzido (KIMBALL, 2002). .....	19
Figura 4 – Exemplo de Tabela Fato (KIMBALL, 2002). .....	22
Figura 5 – Representação esquemática de um modelo <i>Star Schema</i> (INMON, 2005)...	24
Figura 6 - Representação esquemática de um modelo <i>Snowflake</i> (INMON, 2005). .....	25
Figura 7 - Um grafo RDF que descreve Eric Miller (W3C, 2004). .....	29
Figura 8 - Código RDF/XML que descreve Eric Miller (W3C, 2004). .....	30
Figura 9 - Nuvem do “Linking Open Data” (LOD), com a <i>DBpedia</i> ao centro, por Richard Cyganiak e Anja Jentzsch ( <a href="http://lod-cloud.net/">http://lod-cloud.net/</a> ). .....	32
Figura 10 - Exemplo de um infobox da Wikipédia (WIKIPÉDIA, 2013 - <a href="http://pt.wikipedia.org/wiki/IBM">http://pt.wikipedia.org/wiki/IBM</a> ). .....	33
Figura 11 - Instância da <i>DBpedia</i> denominada “IBM” (DBPEDIA, 2013). .....	33
Figura 12 - Consulta SPARQL que retorna todas as descrições da instância <i>DBpedia</i> “IBM” (Um exemplo de consulta, criada pelos autores desse trabalho) .....	34
Figura 13 – Extrato do resultado da lista de todas as descrições da instância <i>DBpedia</i> “IBM” (Resultado da consulta da Figura 12, realizada no SPARQL <i>Endpoint</i> da <i>Dbpedia</i> ). .....	35
Figura 14 - Resultado em formato de triplas do <i>website</i> da <i>globo.com</i> passado como parâmetro em um <i>website</i> de tradução de páginas HTML para RDF ( <a href="http://inspector.sindice.com">http://inspector.sindice.com</a> , 2013) .....	37
Figura 15 - Consulta genérica SPARQL que retorna o nome dos melhores amigos de uma determinada pessoa (Um exemplo). .....	38
Figura 16 - Grafo de um <i>VCard</i> (cartão de negócios) do John Smith (JENA , 2010)...	39
Figura 17 - Representação do grafo RDF acima com o auxílio do <i>framework</i> Jena (JENA , 2010). .....	40
Figura 18 - Esquema estrela do DM de <i>telecom</i> . .....	45
Figura 19 - Consulta com a quantidade de ligações de São Paulo agrupadas por faixa horária. ....	46

Figura 20 – Extrato do resultado da consulta com a quantidade de ligações de São Paulo agrupadas por faixa horária. ....	46
Figura 21 - Conteúdo da dimensão Canal de Venda. ....	47
Figura 22 - Conteúdo da dimensão Tipo Recarga. ....	48
Figura 23 - Conteúdo da Dimensão Faixa Horária. ....	48
Figura 24 - Conteúdo da Dimensão Call Center.....	49
Figura 25 - Conteúdo da Dimensão Estados. ....	50
Figura 26 - Conteúdo da Fato Recarga. ....	51
Figura 27 - Consulta SPARQL que resgata alguns dados dos Estados Brasileiros. Consulta realizada em maio de 2013. ....	53
Figura 28 - Arquivo de origem passados como parâmetro. ....	55
Figura 29 - <i>Job</i> que carrega as informações do arquivo de origem para uma tabela temporária ( <i>stage</i> ), sem nenhum tratamento .....	56
Figura 30 - <i>Job</i> que carrega as informações da tabela temporária ( <i>stage</i> ) para a dimensão do DM, com o controle de versão e alguns tratamentos. ....	56
Figura 31 - Consultas usadas para atualização dos dados na fato de recargas.....	57
Figura 32 - Novo modelo de dados do DM de <i>telecom</i> após o enriquecimento .....	58
Figura 33 - Conteúdo da dimensão DIM_DBPEDIA_ESTADOS.....	59

# 1 Introdução

Nesse capítulo serão discutidas as motivações e objetivos desta monografia, além também de uma breve explicação quanto a organização do texto ao longo do documento.

## 1.1 Motivação

No mundo globalizado e competitivo atual, as organizações precisam utilizar toda informação disponível para criar e manter uma vantagem competitiva e sair na frente de seus concorrentes. Porém, tomar decisões corretas e dentro de um curto espaço de tempo não é tão simples se a empresa não tiver um ambiente que proporcione o tipo certo de análise.

O ambiente transacional, antes o único utilizado pelas empresas, não fornece esse tipo de informação, pois não permite uma integração maior (*ad hoc*) de dados de diferentes fontes e, em geral, não guarda o histórico das transações ocorridas.

Surgiu então a ideia de *Data Warehouse* (DW) (INMON, 2005), que se preocupa em extrair os dados de diferentes fontes, sejam sistemas legados ou fontes externas à empresa, e aplicar transformações para criar uma estrutura integradora que permita uma melhor utilização dos dados pelos analistas, gerentes e executivos de modo a apoiar a análise.

Uma vez realizada a integração dos dados nesta nova estrutura, ferramentas próprias para análises, como OLAP (*On-Line Analytical Processing*) (OLAP COUNCIL, 1997) e *Data Mining* (TAN *et al.*, 2011), fornecem mecanismos sofisticados para descobrir informações relevantes. A partir dessas análises é que os analistas conseguem desenvolver novas estratégias, decisões, produtos e serviços específicos para cada tipo de cliente, se colocando assim na frente de seus concorrentes.

Hoje em dia há um grande volume de dados na internet, o que permite que, a partir de sua extração, consigamos obter informações que auxiliem na tomada de decisões empresariais, visando uma melhor atuação no mercado.

Sendo assim, por que não usar essas informações para melhorar a atuação de uma empresa em determinado setor? Com o aumento de dados na internet, em especial dados estruturados como por exemplo o Dbpedia<sup>1</sup>, podemos obter informações mais organizadas e íntegras.

A gama de informações disponibilizadas na internet é muito extensa e permite completar informações para análise praticamente de qualquer tipo de nicho no mercado atual.

## 1.2 Objetivos

Este trabalho tem como objetivo fazer um estudo dos principais conceitos necessários para o desenvolvimento de um ambiente de *Data Warehouse*, e como integrá-lo e enriquecê-lo com dados estruturados no formato RDF.

Nele é apresentado um projeto de *Data Mart* (equivalente a uma visão de DW para um processo de negócio específico) para uma empresa de telecomunicação e a integração de seus dados com dados da *web* semântica em formato RDF, linguagem para descrever informações na *web* (W3C, 2004), extraídos da *DBPedia*.

A *Web Semântica* não é uma *web* separada, mas uma extensão da atual. Nela, a informação é dada com um significado bem definido, permitindo melhor interação entre os computadores e as pessoas (BERNERS-LEE, 2001). Dessa forma, o uso desses dados com significado bem definido pode ajudar as tomadas de decisão ao serem combinadas com os dados já existentes no *DW*.

---

<sup>1</sup> <http://dbpedia.org>

### 1.3 Organização do texto

O presente trabalho está estruturado em capítulos e, além desta introdução, se organiza da seguinte forma:

Capítulo II - **Data Warehouse**: são apresentados os conceitos do *Data Warehouse*, seus componentes, arquitetura e suas principais formas de modelagem.

Capítulo III - **Web Semântica e Dados Ligados**: explica o conceito de *Web Semântica*, dados ligados, RDF, ontologias e o projeto DBPedia.

Capítulo IV - **Abordagens de Enriquecimento do DW**: apresenta a abordagem do enriquecimento de um DW a partir de dados ligados.

Capítulo V - **Estudo de Caso: Data Mart de Telecom**: é apresentado o estudo de caso do projeto, que é um Data Mart de uma empresa de Telecomunicação, e sua estrutura e modelagem.

Capítulo VI - **Enriquecendo o DM de Telecom**: é apresentado a forma que os dados RDF são extraídos da *web* e como são integrados ao nosso ambiente de DM.

Capítulo VII - **Conclusão**: Reúne as considerações finais, assinala as contribuições da pesquisa e sugere possibilidades de aprofundamento posterior.

## 2 Data Warehouse

Nesse capítulo será apresentada a estrutura geral de um *Data Warehouse*, suas principais características, sua arquitetura tradicional, principais componentes, e os tipos de modelagem de dados mais comuns existentes para este tipo de abordagem.

### 2.1 Origem

As origens de armazenamento de dados e processos de sistemas de apoio à decisão remontam aos primeiros dias de computadores e sistemas de informação. É interessante notar que o processos dos sistemas de apoio à decisão foram desenvolvidos a partir de uma longa e complexa evolução da tecnologia da informação. Sua evolução continua até hoje (INMON, 2005).

No início de 1960, o mundo da computação consistia em criar aplicações individuais que executavam arquivos mestres. As aplicações consistiam em relatórios e programas, geralmente construídos em uma linguagem nova para a época, como Fortran ou COBOL (INMON, 2005).

Em meados dos anos 1960, o crescimento dos arquivos mestres e fita magnética explodiu. E com esse crescimento veio uma enorme quantidade de dados redundantes. A proliferação de arquivos mestres e dados redundantes apresentou alguns problemas complexos:

- A necessidade de sincronizar os dados sobre atualização
- A complexidade da manutenção de programas
- A complexidade do desenvolvimento de novos programas
- A necessidade de grandes quantidades de hardware para suportar todos os arquivos mestres

Em pouco tempo, os problemas dos arquivos mestres - problemas inerentes ao próprio meio - tornaram-se sufocantes.

A década de 1970 viu o advento do armazenamento em disco. Com o armazenamento em disco surgiu um novo tipo de software conhecido como sistema de gerenciamento de banco de dados (SGBD). O objetivo do SGBD era tornar mais fácil para o programador o armazenamento e acesso aos dados (INMON, 2005).

Em meados da década de 1970, o processamento de transações online (OLTP) tornaram o acesso aos dados mais rápido, abrindo novas perspectivas para toda a empresa e processamento. O computador podia ser utilizado para tarefas que não eram possíveis anteriormente, incluindo os sistemas de condução, sistemas de caixas de banco (máquinas ATM) e sistemas de controle de fábrica (INMON, 2005).

Logo após a chegada massiva dos sistemas OLTP, uma grande quantidade de programas de extração começaram a aparecer. Uma "teia de aranha" de processamento de extratores começou a formar-se. Primeiro, foram os extratores; então havia extratores de extratores e, depois, extratores de extratores de extratores, e assim por diante. Não era incomum para uma grande empresa atuar com 45.000 extratores por dia. A arquitetura naturalmente evoluída apresentava muitos desafios, tais como credibilidade dos dados, produtividade e incapacidade de transformar dados em informações (INMON, 2005).

O *status* dessa arquitetura a qual a maioria das empresas adotou, simplesmente não era robusta o suficiente para atender às necessidades futuras. Era necessário algo muito maior e para isso, uma alteração da arquitetura, dando origem a arquitetura do *Data Warehouse* (INMON, 2005).

O *Data Warehouse*, ou Armazém de Dados, surgiu da necessidade de se armazenar a informação contida nos sistemas transacionais numa base de dados central, para que houvesse integração total dos dados da empresa. Além disso, era necessário manter o histórico das informações e fazer com que os dados fossem dispostos dimensionalmente, ou seja, o analista de negócios poderia visualizar um mesmo fato através de diversas dimensões diferentes.

Com o surgimento do DW foram necessários também o desenvolvimento de uma nova estruturação dos dados, tanto em termos de modelagem quanto em termos de armazenamento e acesso a eles. Essas técnicas necessárias para arquitetar o DW são bem diferentes às implementadas em sistemas transacionais.

Entender essa nova arquitetura e estrutura dos dados ajuda os analistas e gerentes a identificarem as necessidades e oportunidade de seus negócios e tomarem ações para suprir essas questões.

## **2.2 Características**

Essa seção apresenta as características e comportamento de um *Data Warehouse*.

### **2.2.1 Orientado por assunto**

O *Data Warehouse* é orientado para as principais áreas da corporação que foram definidos no modelo de dados de alto nível (INMON, 2005). Refere-se ao fato do DW guardar informações sobre assuntos específicos e importantes para o negócio da empresa.

Alguns exemplos de assuntos são: produtos, clientes, contratos e políticas. Cada tipo de empresa tem seu próprio conjunto de assuntos (INMON, 2005).

### **2.2.2 Integrado**

De todos os aspectos de um *Data Warehouse*, a integração é a mais importante. Os dados do *Data Warehouse* são alimentados a partir de diferentes e múltiplas fontes. Conforme os dados são alimentados, eles são convertidos, reformatados, resequenciados, sumarizados e assim por diante. O resultado é que os dados - uma vez que residem no *Data Warehouse* - tem uma única imagem corporativa física (INMON, 2005).

Conforme os dados são trazidos para o *Data Warehouse*, eles são convertidos para um estado uniforme, ou seja, sexo é codificado apenas de uma forma. Da mesma maneira, se um elemento de dado é medido em segundos em uma aplicação e em horas em outra, ele será convertido para uma mesma representação ao ser colocado no *Data Warehouse*, como mostra a Figura 1.



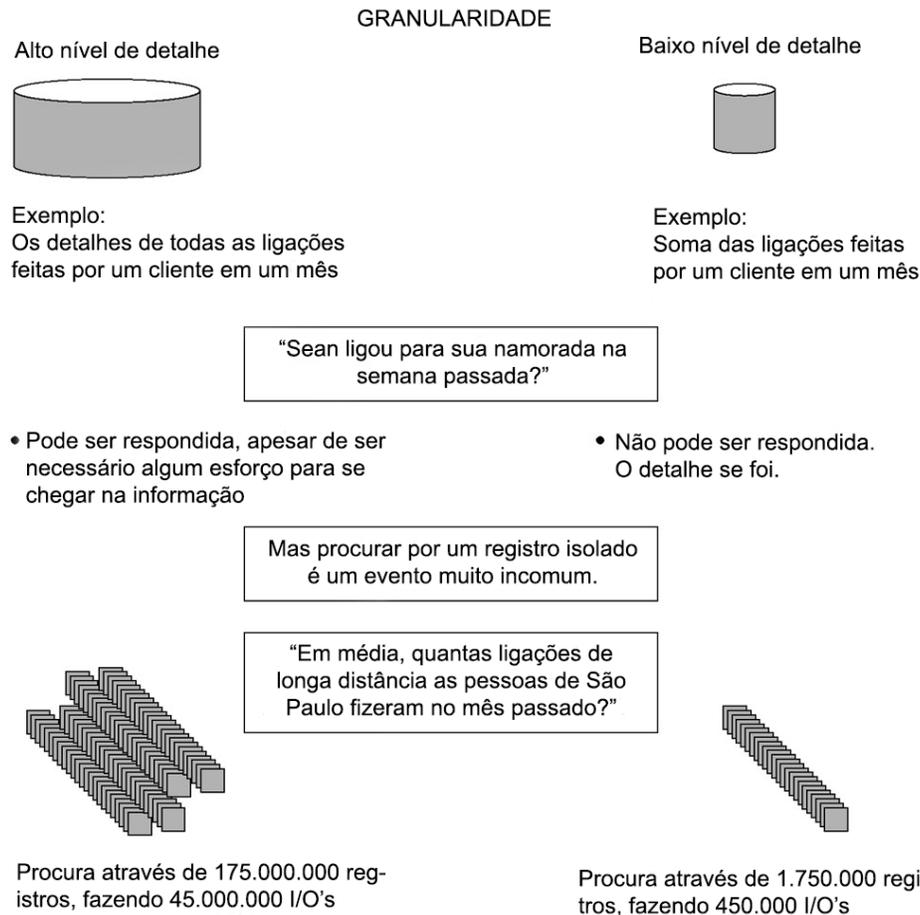
momento, como se fosse uma foto. Quando ocorrem alterações posteriores, um novo registro instantâneo é criado. Ao seguir-se essa estratégia, um registro histórico do dado é mantido no *Data Warehouse* (INMON, 2005).

#### **2.2.4 Variante no Tempo**

Variante no tempo implica que cada unidade de dado no *Data Warehouse* é exato em algum momento no tempo. Em alguns casos, o registro tem controle de hora, com *timestamp*. Em outros casos, um registro tem a data da transação. Mas em todos os casos, existe alguma forma de marcação de tempo para exibir o momento no tempo que aquele dado pertence (INMON, 2005).

#### **2.2.5 Granularidade**

A granularidade é a característica mais importante na construção do *Data Warehouse*. Granularidade refere-se ao nível de detalhe ou sumarização contido nas unidades de dados existentes no *Data Warehouse* (INMON, 2005). Quanto maior o nível de detalhes, menor o nível de granularidade e quanto menor o nível de detalhe, maior o nível de granularidade. O nível de granularidade afeta diretamente o volume de dados armazenado no *Data Warehouse* e ao mesmo tempo o tipo de consulta que pode ser respondida (INMON, 2005). A Figura 2 mostra exemplos de granularidades diferentes.



**Figura 2 - Determinar o nível de granularidade é a questão mais importante do projeto no ambiente de data warehouse. Traduzido (INMON, 2005).**

### 2.2.6 Normalização / Desnormalização

Na construção do modelo dimensional, há uma preocupação com os assuntos a serem modelados, se baseando nos fatos a serem analisados e nas dimensões de análise (com seus atributos), que correspondem às perspectivas de análise. Ao se implementar um modelo dimensional (ou multi-dimensional, por se considerar várias dimensões para cada assunto) em um sistema de gerência de banco de dados (SGBD) relacional (gerando o que se chama de esquema estrela), a desnormalização é muito usada, diferentemente de esquemas relacionais tradicionais utilizados em sistemas transacionais, em que se enfoca a normalização. Quando tem-se tabelas normalizadas, evita-se a redundância, mas aumenta-se o consumo I/O. O motivo de se introduzir dados

redundantes em implementações do modelo dimensional em SGBD relacionais é justamente diminuir a quantidade de junções (*joins*) e consumo *I/O*.

### 2.2.7 Metadados

Os metadados, também conhecidos como "dados sobre dados", são de grande importância para o processo de controle das operações em um *Data Warehouse*.

Segundo INMON (2005), os metadados tem um novo nível de importância no mundo do *Data Warehouse*, já que funcionam como índices do conteúdo do *Data Warehouse* permitindo o uso mais efetivo.

Durante todas as fases do projeto de um *Data Warehouse*, e também após o início de sua operacionalização, metadados devem ser armazenados.

Os aspectos sobre os quais os metadados mantêm informações são:

- A estrutura dos dados segundo a visão do programador;
- A estrutura dos dados segundo a visão dos analistas de sistemas de apoio à decisão;
- As fontes de dados que alimentam o *Data Warehouse*;
- As transformações sofridas pelos dados no momento de sua migração para o *Data Warehouse*;
- O modelo de dados;
- O relacionamento entre o modelo de dados e o *Data Warehouse*;
- O histórico das extrações de dados.

## 2.3 Arquitetura e Componentes

Um dos maiores impedimentos ao sucesso do *Data Warehouse* é confundir as funções e papéis de seus componentes e arquitetura.

A Figura 3 apresenta quatro (4) componentes distintos a serem considerados quando nos referimos à construção de um *Data Warehouse*.

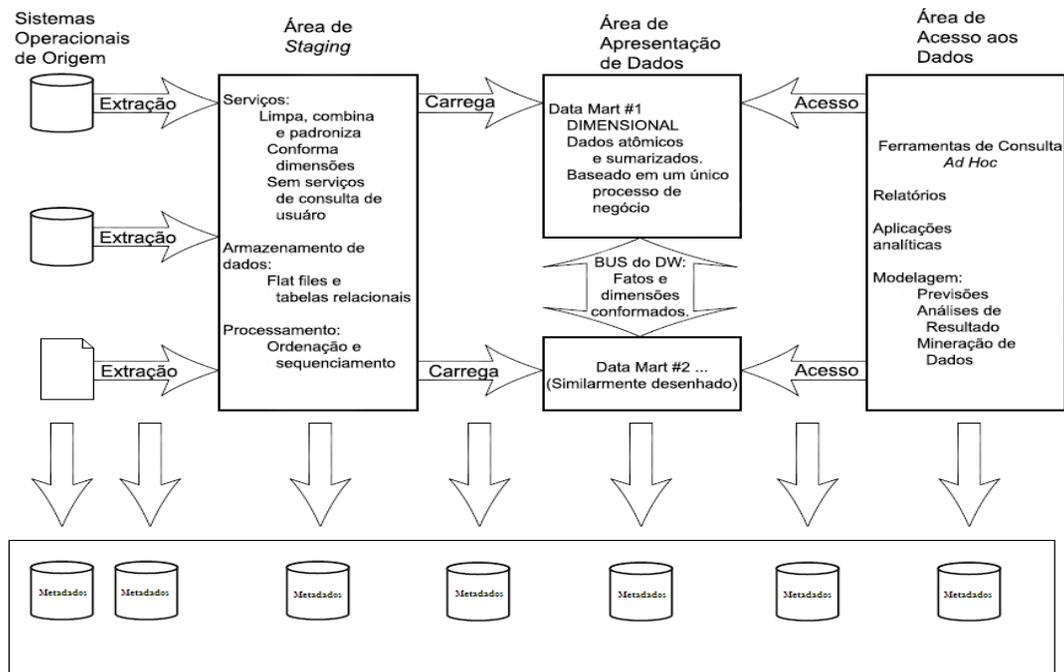


Figura 3 - Componentes do *Data Warehouse*. Traduzido (KIMBALL, 2002).

### 2.3.1 Sistema Transacional (*Operational Source Systems*)

Também conhecido como Sistema de Processamento de Transação ou Sistema Operacional, os dados são representados no maior nível de detalhe, ou seja, de transação. As maiores prioridades do Sistema Transacional é manter uma boa performance, devido ao grande volume de dados, e sempre mantê-los acessíveis (KIMBALL, 2002).

Os Sistemas Transacionais mantêm um histórico dos dados muito curto, mantendo apenas o necessário, em especial o dado corrente (KIMBALL, 2002).

### 2.3.2 Área de *Staging* (*Data Staging Area*)

A *Data Staging Area*, ou Área de preparação dos Dados, desempenha duas funções principais, sendo simultaneamente uma área de armazenamento e uma área onde uma série de processamentos e transformações ocorrem, referida como ETL.

Segundo KIMBALL (2002), a Área de Staging é tudo entre o Sistema Transacional e a Área de Apresentação de Dados.

O primeiro passo para se disponibilizar os dados na área de Staging é a extração dos dados dos Sistemas Transacionais, ou seja, copiar os dados necessários desses sistemas ao *Data Warehouse* para que sejam tratados e transformados.

Uma vez disponibilizados na Área de *Staging*, exigem uma série de transformações potenciais a serem executadas, como a limpeza dos dados (correção de pontos, vírgulas e caracteres especiais, padronização de medidas e etc), combinação dos dados de diferentes fontes e a criação de uma chave interna para controle no *Data Warehouse*.

Todas essas transformações preparam os dados para serem carregados na Área de Apresentação de Dados.

### **2.3.3 Área de Apresentação de Dados (*Data Presentation Area*)**

A Área de Apresentação de Dados é onde os dados são organizados, armazenados e disponibilizados para que os usuários finais realizem consultas e analistas de negócios e gerentes criem relatórios através de aplicações analíticas.

KIMBALL (2002) se refere à Área de Apresentação de Dados como uma série de *Data Marts* integrados e um *Data Warehouse* tem em média aproximadamente 20 ou mais *Data Marts* similares. Todos os *Data Marts* devem ser construídos usando dimensões e fatos comuns para não virarem funis isolados de dados. *Data Marts* que não se *comunicam* são um grande problema do desenvolvimento do *Data Warehouse*.

### **2.3.4 Área de Acesso aos Dados (*Data Access Tools*)**

A Área de Acesso aos Dados é, por definição, todas as aplicações que acessam os dados da Área de Apresentação de Dados. As aplicações da Área de Acesso aos Dados podem ser simples consultas *ad-hoc* ou complexos e sofisticados softwares *OLAP* ou *data mining*.

## 2.4 Modelagem de dados em um *Data Warehouse*

Essa seção apresenta as características e os componentes da modelagem de dados de um *Data Warehouse*.

### 2.4.1 Modelagem Multidimensional

A modelagem multidimensional é, em vários aspectos, mais simples, mais expressiva e mais fácil de entender que a modelagem ER<sup>2</sup>. Segundo MACHADO (2008), a modelagem multidimensional é uma técnica de concepção e visualização de um modelo de dados de um conjunto de medidas que descrevem aspectos comuns do negócio.

Em outras palavras, essa modelagem é ideal para sumarizar e estruturar os dados de uma maneira que suporte a visualização e análise dos dados e suas regras de negócio.

Um modelo multidimensional é formado por três elementos básicos (MACHADO, 2008):

- *Fatos*;
- *Dimensões*;
- *Medidas (variáveis)*.

### 2.4.2 Fatos

A tabela Fato é a principal tabela do modelo multidimensional e é nela que as medidas numéricas e indicadores de performance são armazenados, como por exemplo, a quantidade de itens vendidos de um determinado produto e o valor total dessas vendas. A Figura 4 mostra um exemplo de tabela Fato.

---

<sup>2</sup> O Modelo de Entidade e Relacionamento (MER) é um representação da realidade e pode ser representado por entidades, relacionamentos e atributos (LONDEIX, 1995)

Sales Fact Table
time_key (FK)
product_key (FK)
store_key (FK)
customer_key (FK)
clerk_key (FK)
promotion_key (FK)
dollars_sold
units_sold
dollars_cost

**Figura 4 – Exemplo de Tabela Fato (KIMBALL, 2002).**

Segundo KIMBALL (2002), uma medida se refere a várias *dimensões*, sendo portanto a interseção destas *dimensões*, e essa lista de dimensões é que define o grão da tabela Fato e nos diz o escopo desse fato. Uma linha da tabela Fato corresponde a um fato. Todas as medidas em uma tabela Fato devem, idealmente, estar no mesmo grão.

A maioria das tabelas Fatos de um modelo são *numéricas* e *aditivas*, ou seja, contém medidas numéricas que tornam possíveis as operações de soma. Isso não ocorre por acaso, pois as aplicações de *Data Warehouse* quase nunca buscam apenas uma linha para análise, mas trabalham sempre com um volume alto (centenas, milhares e às vezes milhões) de dados. Com esse volume todo, nada mais interessante que efetuar operações de soma, subtração e média para uma melhor análise de negócio.

Existem também medidas semi-aditivas, que podem ser somadas com o auxílio da dimensão correspondente e não-aditivas, que de maneira alguma podem ser somadas. A solução para analisar dados de medidas não-aditivas é utilizar funções de *count* e *sum* para sumarizar as linhas do resultado.

Geralmente a maioria das tabelas Fato contém uma *chave primária* composta de duas ou mais *chaves estrangeiras* correspondente às dimensões. Assim, toda tabela correspondente a um modelo dimensional que tem uma chave primária composta de duas ou mais chaves estrangeiras é uma tabela Fato. Em outras palavras, toda tabela que tem uma relação de "N pra N" deve ser uma tabela Fato.

### 2.4.3 Dimensões

As tabelas de dimensão contém as descrições de negócio (ou perspectivas de análise) do *Data Warehouse*. Em um *Data Warehouse* bem desenhado, as Dimensões

geralmente contém muitas colunas, às vezes de 50 a 100, que descrevem bem o negócio em questão. Esse tipo de tabela, apesar da grande quantidade de colunas, geralmente contém poucos registros em termos de volumes dentro de um *Data Warehouse*, dificilmente ultrapassando 1 milhão de linhas (na maioria das vezes as Dimensões apresentam muito menos que isso) (KIMBALL, 2002).

Os atributos das dimensões são os principais atributos usados nos filtros das consultas, agrupamentos e relatórios. O poder de um *Data Warehouse* está diretamente ligado com a qualidade dos dados de suas dimensões e quanto mais tempo se gasta no levantamento e definição desses atributos, melhor é o *Data Warehouse* (KIMBALL, 2002).

Os melhores atributos de uma dimensão são os atributos textuais. Eles devem conter palavras completas e não devem ser usadas abreviações, pois estas dificultam o entendimento. Em alguns casos, alguns códigos operacionais ou identificadores são tão importantes quanto descrições e são indispensáveis para a análise do negócio. Nesses casos esse tipo de atributo deve ser adicionado às dimensões junto com as descrições textuais (KIMBALL, 2002).

#### **2.4.4 Medidas**

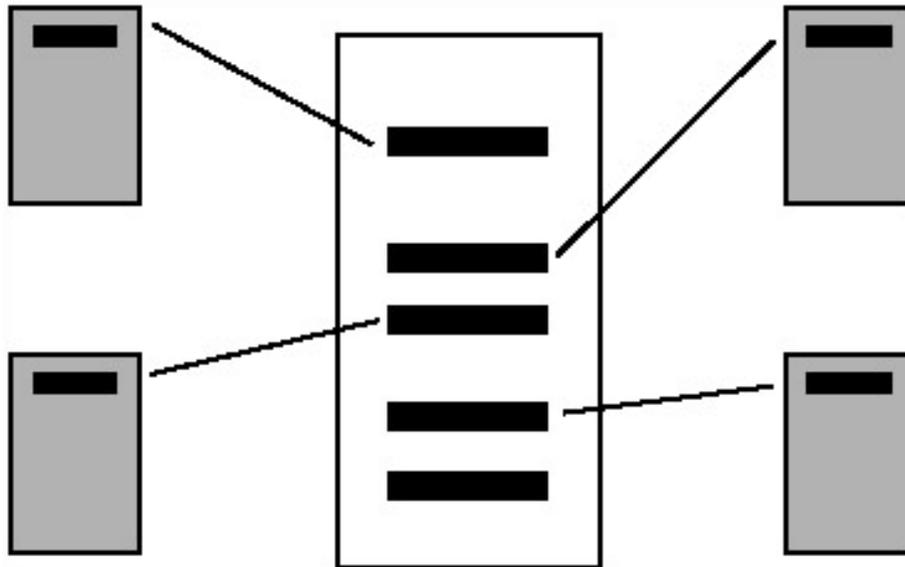
Medidas são os atributos numéricos que representam um fato, a performance de um indicador de negócios relativo às dimensões que participam desse fato (MACHADO, 2008).

Em outras palavras, medidas são os valores em reais das vendas, o número de unidade de produtos vendidos, a quantidade de ligações feitas por um determinado cliente, entre outros.

#### **2.4.5 Esquema Estrela**

O Esquema Estrela (*Star Schema*) é o mais usado na construção de *Data Warehouses* e, segundo MACHADO (2008), sua composição típica é uma grande entidade central denominada fato e um conjunto de entidades menores denominadas

dimensões arranjadas ao redor dessa entidade central, formando uma estrela, como mostra a Figura 5.

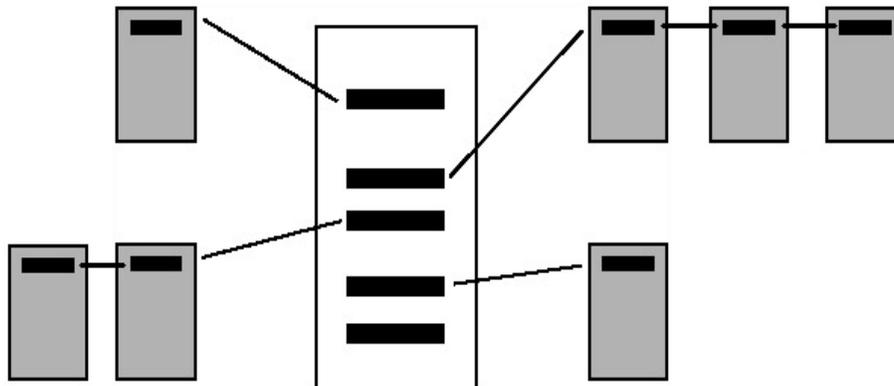


**Figura 5 – Representação esquemática de um modelo *Star Schema* (INMON, 2005).**

Uma grande vantagem do modelo *star schema* é a simplicidade de estruturação dos dados, permitindo que o usuário final entenda e navegue de forma intuitiva pelos dados.

#### 2.4.6 Esquema Floco de Neve

O Esquema Floco de Neve (*snowflake*) é a extensão do Esquema Estrela, mantendo a mesma idéia de ter uma tabela fato central cercada por dimensões. A diferença é que o *snowflake* é o resultado da normalização dessas dimensões, resultando em uma hierarquia de dimensões. Segundo INMON (2005), enquanto o Esquema Estrela é formado por uma tabela de fato ligada a algumas tabelas de dimensão, o Esquema Floco de Neve é quando uma tabela fato está combinada com pelo menos uma tabela de dimensão normalizada, como pode ser visto na Figura 6.



**Figura 6 - Representação esquemática de um modelo *Snowflake* (INMON, 2005).**

As vantagens do *snowflake* são a economia de espaço, já que na tabela fato se guardaria apenas o código e não a descrição, e a maior facilidade de manutenção.

KIMBALL (2002) não recomenda o uso do *snowflake* pois, segundo ele, o espaço em disco economizado pela normalização é insignificante e as consultas se tornam mais complexas, aumentando a quantidade de junções, e assim, diminuindo sua performance em bancos.

## 2.5 Considerações Finais

Nesse capítulo foram mostrados os principais conceitos e fundamentos do *Data Warehouse*. No próximo capítulo serão mostrados os conceitos de *Web Semântica*.

## 3 Web Semântica e Dados Ligados

Nesse capítulo serão apresentados os conceitos de *web* semântica, dados ligados, RDF (W3C, 2004) e SPARQL (W3C, 2004), bem como a explicação do que é a *DBpedia*, fonte principal da informação extraída nesse projeto.

### 3.1 Web Semântica

Para definirmos *Web Semântica*, precisamos começar com a definição de semântica. Semântica é o estudo do significado, se ocupando do que algo significa e, portanto, permite uma utilização mais eficaz dos dados subjacentes (HEBELER *et. al*, 2009). O significado é muitas vezes ausente na maioria das fontes de informação, exigindo que os usuários ou instruções de programação complexas o forneçam.

Por exemplo, páginas da *web* são preenchidas com informações e etiquetas associadas. A maioria das *tags* representam instruções de formatação, como <H1> para indicar um título importante. Semanticamente, sabemos que palavras rodeadas por *tags* <H1> são mais importantes para o leitor do que outro texto por causa do significado de <H1>. Algumas páginas da *web* adicionam semânticas básicas para os motores de busca usando a *tag* <META>, no entanto são apenas palavras-chave isoladas e ligações pobres para fornecerem um contexto mais significativo (HEBELER *et. al*, 2009).

Estas semânticas são fracas e limitam a busca para retornarem o resultado exato. Da mesma forma, os bancos de dados contêm dados e dicas de semânticas limitadas, se as tabelas e colunas que cercam os dados estiverem bem-nomeadas (HEBELER *et. al*, 2009).

Semântica dá um sentido útil à palavra através do estabelecimento de relações entre palavras a fim de estabelecer diferentes entendimentos. Por exemplo, uma palavra-chave *construção* existe em uma página *web* dedicada à ontologias. A *tag* <META>

marca a palavra *construção* para indicar a sua importância. No entanto, a palavra *construção*, dado o contexto da página *web*, pode significar a construção de uma ontologia ou ontologias que se concentram na construção de obras (HEBELER *et. al*, 2009).

A dúvida exposta no parágrafo anterior expõe a dificuldade em simplesmente se expressar semântica. Semântica foi criada para pessoas interpretarem. No entanto, se a palavra-chave relaciona-se com outras palavras-chave nas relações definidas, uma rede de dados ou de outras formas de contexto revelam semântica. Então se a palavra-chave *construção* se relaciona com várias palavras-chave, tais como a *arquiteto*, *pedreiro*, *canteiro de obras*, e assim por diante, as relações expõem semântica.

A *Web Semântica* é uma extensão da *web* atual. Ela liga significados de palavras, e tem como objetivo atribuir um sentido a conteúdos publicados na internet, para que tanto homem e máquina possam interpretá-los. Uma aplicação pode adicionar semântica através de instruções de programação, porém não existe um padrão formal para tais semânticas programadas. Isto faz com que seja difícil tirar proveito desta informação ou mesmo de a reconhecer. A semântica está perdida em um labirinto de instruções de programação *if / else*, pesquisas de banco de dados, e muitas outras técnicas de programação. Isto faz com que seja difícil tirar proveito desta rica informação ou mesmo reconhecer isso tudo. Estando sozinho, o significado de vários termos, tais como *construção*, é simplesmente perdido.

A *Web Semântica* trata semântica através de conexões padronizadas para informações relacionadas. Assim, uma aplicação pode facilmente dizer se um edifício é o mesmo que uma outra referência edifício. Cada elemento de dados único então se conecta a um contexto mais amplo, ou *web*. A *web* oferece potenciais caminhos para a sua definição, relações com uma hierarquia conceitual, relações com a informação associada, e relações com instâncias específicas de construção (HEBELER *et. al*, 2009).

### 3.2 RDF

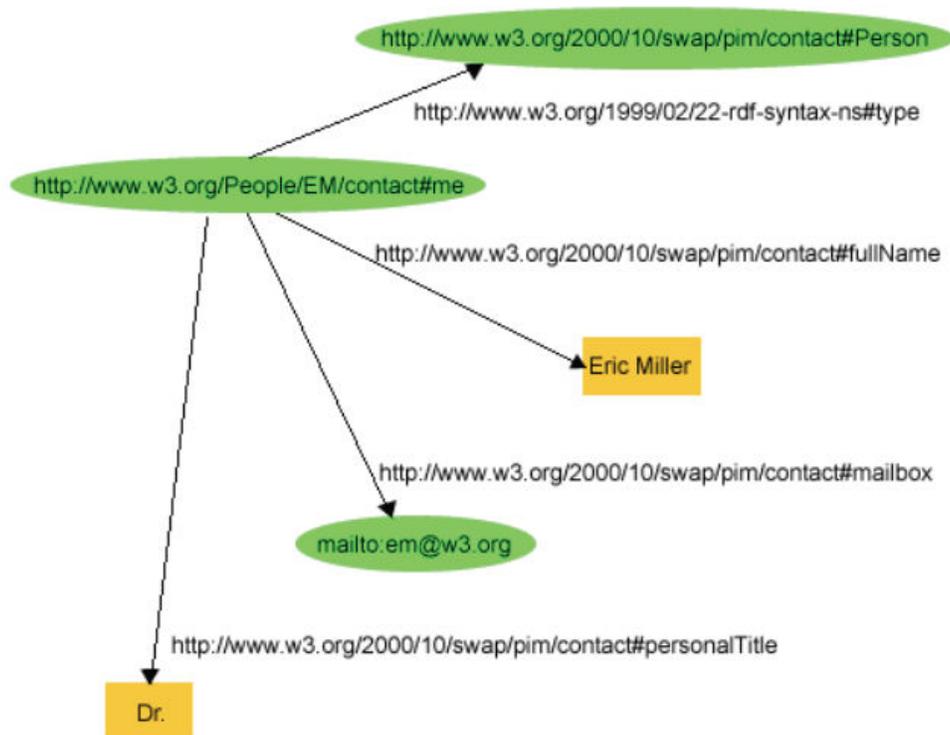
O RDF (*Resource Description Framework*) é uma linguagem para descrever informações na *web* (W3C, 2004). A padronização da descrição destes recursos permite a sua leitura e interpretação por computadores. RDF (HEBELER *et. al*, 2009) é um

modelo abstrato de informações que pode ser serializado de várias maneiras e que representa informações de declarações. Uma declaração é feita por três elementos, um assunto, um predicado e um objeto.

O RDF tem uma abordagem geral baseada em grafo para representar a informação de modo que possa ser facilmente compartilhada. Neste grafo (HEBELER *et. al*, 2009), uma declaração representa uma aresta direcionada entre dois nós, com o predicado (também referida como uma propriedade) representando a borda, e o sujeito e o objeto representados por nós, ou nodos. RDF define dois tipos básicos de nós: literais e recursos. Um literal é um valor concreto, como uma *string* ou um número, e literais não podem ser os sujeitos, apenas os objetos. Recursos representam, em geral, qualquer coisa que possa ser atribuído um *Internacional Resource Identifier* (IRI). Exemplos de recursos incluem pessoas, coisas, lugares e conceitos.

O RDF fornece uma sintaxe baseada em XML, chamada de RDF/XML (W3C, 2004), porém outras representações sintáticas são possíveis, como, por exemplo, N3, N-Triples, Turtle e JSON.

Para exemplificar a forma como o RDF funciona, o grupo de declarações "existe uma Pessoa identificada por <http://www.w3.org/People/EM/contact#me>, cujo nome é Eric Miller, seu endereço de e-mail é em@w3.org e seu título é Dr." poderia ser representado pelo grafo da Figura 7.



**Figura 7 - Um grafo RDF que descreve Eric Miller (W3C, 2004).**

A Figura 7 ilustra que o RDF usa URI's para identificar:

- Indivíduos, por exemplo, Eric Miller é identificado por `http://www.w3.org/People/EM/contact#me`;
- Tipos de recursos, por exemplo, uma pessoa é identificada por `http://www.w3.org/2000/10/swap/pim/contact#Person`;
- Propriedades desses recursos, por exemplo, caixa de correio é identificado por `http://www.w3.org/2000/10/swap/pim/contact#mailbox`;
- Valores dessas propriedades, por exemplo, `mailto:em@w3.org` como o valor da propriedade de caixa (RDF também utiliza cadeias de caracteres como "Eric Miller", e os valores de outros tipos de dados, como números inteiros e datas, como os valores de unidades).

A Figura 8 é um trecho de RDF em notação RDF/XML do grafo da Figura 7, que corresponde ao Eric Miller.

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

  <contact:Person rdf:about="http://www.w3.org/People/EM/contact#me">
    <contact:fullName>Eric Miller</contact:fullName>
    <contact:mailbox rdf:resource="mailto:em@w3.org"/>
    <contact:personalTitle>Dr.</contact:personalTitle>
  </contact:Person>
</rdf:RDF>
```

**Figura 8 - Código RDF/XML que descreve Eric Miller (W3C, 2004).**

Assim como o HTML, o RDF/XML é processável na máquina e, usando URIs, pode interligar pedaços de informações na *web*. No entanto, ao contrário do hipertexto convencional, RDF URIs pode se referir a qualquer coisa identificável, incluindo recursos que não podem ser diretamente recuperáveis na *web* (como a pessoa Eric Miller). O resultado é que, além de descrever páginas *web*, RDF também pode descrever casas, empresas, pessoas, animais, eventos, notícias etc. As propriedades RDF possuem URIs, que identificam com as relações que existem entre os itens relacionados.

### 3.3 Dados ligados abertos

Dados ligados abertos (Linked Open Data) (BERNERS-LEE, 2006) referem-se a criação de conexões entre dados na *web* e a disponibilização destes dados de maneira que possam ser lidos por máquinas. Seus significados estão ligados a outros conjuntos de dados e podem ser ligados a partir de conjuntos de dados externos (BIZER et al., 2008).

A idéia básica de Dados Ligados foi desenvolvida por BERNERS-LEE (2006). Ele definiu os quatro princípios para caracterizar dados ligados que devem ser aplicados para fazer a *web* crescer:

- Usar URIs para nomear "Coisas";
- Usar HTTP URIs de modo que as pessoas possam procurar esses nomes;
- Proporcionar informações úteis quando alguém procura um URI;

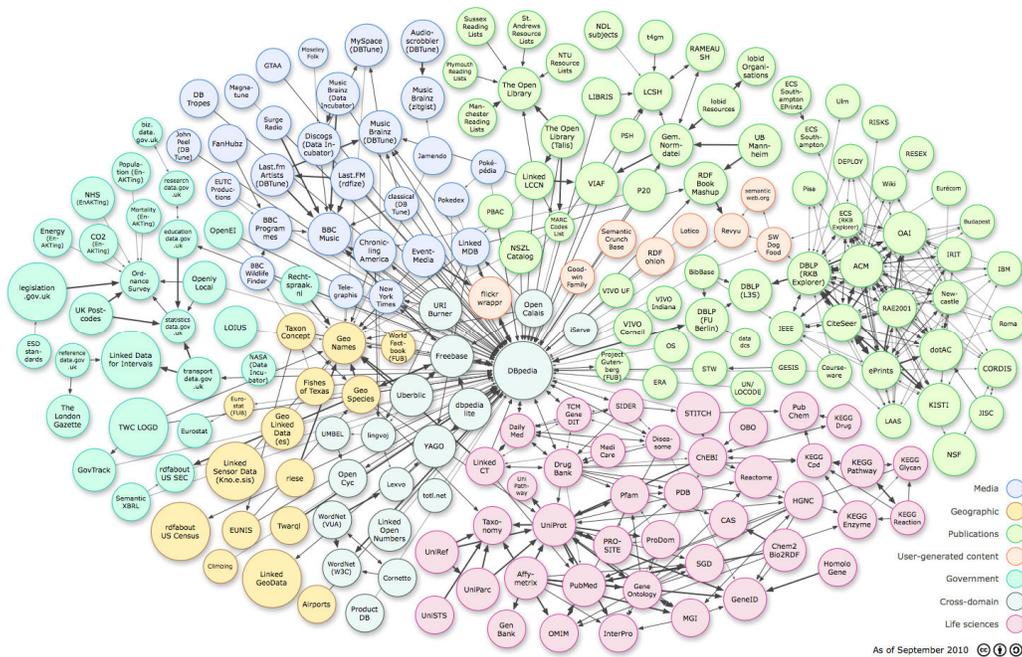
- Incluir links para outros URIs de modo que as pessoas possam encontrar mais informações relacionados com "coisas".

De acordo com BERNERS-LEE (2009), os benefícios globais de dados ligados abertos são: eles são acessíveis por uma variedade ilimitada de aplicações porque eles são disponibilizados em formatos abertos, eles podem ser combinados com qualquer outro conjunto de dados ligados, e nenhum planejamento prévio é necessário para integrar estas fontes de dados, desde que os dois estejam no padrão dos dados ligados. Além disso, é fácil adicionar um conjunto de novos dados ligados aos já existentes, mesmo quando os termos e definições mudam ao longo do tempo.

URIs, HTTP e RDF são as principais tecnologias que suportam dados ligados (BIZER et al., 2008). Além dessas, outras tecnologias da *Web Semântica* são usadas para fornecer diferentes tipos de apoio, tais como a linguagem SPARQL (W3C, 2004) para consultar dados RDF (W3C, 2004), o RDFS (*Resource Description Framework Schema*) (W3C, 2004) e OWL (*Web Ontology Language*) (W3C, 2004), linguagens para definir vocabulários, e RDFa (RDF - em - atributos) (W3C, 2004) linguagem para publicação de páginas HTML com significado.

### 3.4 DBpedia

O projeto *DBpedia* é um esforço da comunidade para extrair dados estruturados da *Wikipedia*, organizá-los através de uma licença aberta e interligá-los com outras fontes de dados abertos na *web*, que pode ser vista na Figura 9 abaixo.



**Figura 9 - Nuvem do “Linking Open Data” (LOD), com a *DBpedia* ao centro, por Richard Cyganiak e Anja Jentzsch (<http://lod-cloud.net/>).**

O funcionamento da *DBpedia* consiste na extração (*dump*) dos *infoboxes* (ou “caixas de informação”) que apresentam dados estruturados encontrados na Wikipédia, e seu posterior mapeamento em ontologias. A ligação dos dados da *DBpedia* é feita através do modelo RDF e, com isso, permite a criação de consultas elaboradas baseadas em SPARQL (AUER, 2007). A Figura 10 mostra um *infobox* da Wikipédia.

International Business Machines (IBM)	
<b>Slogan</b>	<i>Um Planeta Mais Inteligente</i>
<b>Tipo</b>	Pública (NYSE: <a href="#">IBM</a> )
<b>Fundação</b>	1888, incorporada em 1911
<b>Sede</b>	Armonk, NY,  Estados Unidos
<b>Pessoas-chave</b>	 Virginia Rometty (CEO, Presidente e Chairman)  Rodrigo Kede Lima (Diretor Geral - Brasil)
<b>Empregados</b>	399,409 (2009)
<b>Indústria</b>	Hardware Software Serviços de Tecnologia da Informação
<b>Lucro</b>	<span style="color: green;">▲</span> 14,8 bilhões USD (2010)
<b>LAJIR</b>	<span style="color: green;">▲</span> 20,1 bilhões USD (2010)
<b>Faturamento</b>	<span style="color: green;">▲</span> 99,9 bilhões USD (2010)
<b>Página oficial</b>	<a href="http://www.ibm.com/br">www.ibm.com/br</a>

**Figura 10 - Exemplo de um infobox da Wikipédia (WIKIPÉDIA, 2013 - <http://pt.wikipedia.org/wiki/IBM>).**

A Figura 11 apresenta uma instância da *DBpedia* denominada “IBM”, obtida a partir do processamento dos dados da *Wikipédia*.

About: <b>IBM</b>	
An Entity of Type : <a href="#">Public company</a> , from Named Graph : <a href="http://dbpedia.org">http://dbpedia.org</a> , within Data Space : <a href="#">dbpedia.org</a>	
International Business Machines (IBM) é uma empresa dos Estados Unidos voltada para a área de informática. A empresa é uma das poucas da área de Tecnologia da Informação. A IBM fabrica e vende Hardware e Software, oferece serviços de infra-estrutura, serviços de hospedagem e serviços de consultoria nas áreas que vão desde computadores	
Property	Value
<a href="#">dbpedia:owl:abstract</a>	<ul style="list-style-type: none"> <li>IBM és l'acrònim d'International Business Machines també coneguda col·loquialment com el gegant blau. És una empresa d'informàtica des dels seus diferents vessants. Exemples de programari desenvolupat per IBM per al PC són DisplayWrite, Drawii el 1888, tot i que no va constituir-se formalment fins el 15 de juny de 1911. Té la seva seu central a Armonk i és la major empresa 160 països del món. Amb més de 300.000 empleats a tot el món i ingressos per valor de 95 mil milions de dòlars (xifres de 2001), al món, i una de les poques fundada al segle XIX. A Espanya hi té 6.900 empleats i opera des de 1926. Té enginyers i assessors, situats a tot el món, en tots els camps de la informàtica i de la tecnologia de la informació. L'empresa es pionera en alguns d'ells, aquests darrers anys, els serveis i els ingressos de consultoria han estat més grans que els derivats de la fabricació i venda d'ordis. CEO el 29 de gener de 2002 després d'haver dirigit els Serveis Globals d'IBM i ajudar l'empresa a convertir-se en un negoci amb un l'empresa reforçava les seves capacitats consultives de negoci adquirint el braç de consultoria de l'empresa de serveis professional vegada més en la consultoria, els serveis i el programari, posant l'accent també en xips i tecnologies de maquinari d'alt valor afegit professionals tècnics. L'àrea d'investigació d'IBM té vuit laboratoris; cinc d'aquests són a fora dels Estats Units, però tots a Themis quarryadors del premi Nobel. Als EUA, han guanyat quatre Turing Awards, cinc Medalles Nacionals de Tecnologia i cinc Medalla equivalents.</li> <li>International Business Machines Corporation (IBM)– přezdívaná Big Blue neboli Velká modrá, fungující od 1888. akciová společnost informačních technologií. Mezi hlavní činnosti společnosti patří v současnosti výroba a prodej počítačového software a hardware a c</li> <li>Die International Business Machines Corporation (IBM) ist ein US-amerikanisches IT- und Beratungsunternehmen mit Sitz in Armo eines der weltweit führenden Unternehmen für Hardware, Software und Dienstleistungen im IT-Bereich sowie eines der größten Ber Unternehmen der weltweit zweitgrößte Softwarehersteller. Aktuell beschäftigt IBM weltweit 426.751, in Deutschland schätzte die / Anzahl 2009 auf 21.100 Mitarbeiter.</li> <li>International Business Machines Corporation or IBM is an American multinational technology and consulting corporation headquar manufacture and sells computer hardware and software, and it offers infrastructure, hosting and consulting services in areas rangi company was founded in 1911 as the Computing Tabulating Recording Corporation through a merger of three companies: the Tabu Company, and the Computing Scale Corporation. CTR adopted the name International Business Machines in 1924, using a name</li> </ul>

**Figura 11 - Instância da DBpedia denominada “IBM” (DBPEDIA, 2013).**

### 3.5 SPARQL

O SPARQL é uma linguagem de consulta de dados em RDF (PRUD'HOMMEAUX e SEABORNE, 2008) muito semelhante à linguagem SQL usadas nos bancos de dados relacionais.

Geralmente os dados ligados fornecem um *Sparql Endpoint* para consulta aos dados. Um *Sparql Endpoint* é um serviço que implementa o protocolo SPARQL e permite que o usuário, humano ou máquina, faça uma consulta a uma base de dados ligados usando a linguagem SPARQL. O resultado é retornado em algum formato processável por máquina, como por exemplo um arquivo JSON, XML ou RDF/XML. Dessa forma, podemos fornecer para um navegador RDF um URI, que referencia um arquivo RDF diretamente, ou o URI de um *Sparql Endpoint*, que demanda uma consulta SPARQL e retorna um conjunto de triplas RDF.

A Figura 12 apresenta uma consulta em linguagem SPARQL que deve retornar todas as descrições da instância da DBpedia denominada “IBM”

<http://dbpedia.org/resource/IBM>.

```
Query Text
-----
SELECT ?abstract
WHERE
{
    { <http://dbpedia.org/resource/IBM> <http://dbpedia.org/ontology/abstract> ?abstract }
}
```

**Figura 12 - Consulta SPARQL que retorna todas as descrições da instância DBpedia “IBM” (Um exemplo de consulta, criada pelos autores desse trabalho)**

O resultado da consulta apresentada na Figura 12 pode ser visto na Figura 13.

<p>"IBM és l'acrònim d'International Business Machines també coneguda col·loquialment com el gegant dels ordinadors. Exemples de programari desenvolupat per IBM per al PC són DisplayWrite, Drawing Assistant i Lotus 1-2-3. Té la seva seu central a Armonk i és la major empresa informàtica del món, amb fàbriques i oficines a tots els continents. Té unes vendes de 100.000 milions de dòlars (xifres de 2004), IBM és l'empresa de tecnologia de la informació més gran al món, i una de les més valuoses. Té laboratoris de desenvolupament situats a tot el món, en tots els camps de la informàtica i de la física. Aquests darrers anys, els serveis i els ingressos de consultoria han estat més grans que els derivats de la venda d'ordinadors. IBM ha dirigit els Serveis Globals d'IBM i ajudar l'empresa a convertir-se en un negoci amb un gran braç de consultoria de l'empresa de serveis professionals PricewaterhouseCoopers. L'empresa se centra en el valor afegit; al 2005 hi tenia contractats aproximadament 195.000 professionals tècnics. L'àrea de desenvolupament d'IBM hi ha diversos guanyadors del premi Nobel. Als EUA, han guanyat quatre Turing Awards."&gt;@ca</p>
<p>"International Business Machines Corporation (IBM) – přezdívaná Big Blue neboli Velká modrá, fun... hlavní činnosti společnosti patří v současnosti výroba a prodej počítačového software a hardware a d... Die International Business Machines Corporation (IBM) ist ein US-amerikanisches IT- und Beratung... Unternehmen für Hardware, Software und Dienstleistungen im IT-Bereich sowie eines der größten I... beschäftigt IBM weltweit 426.751, in Deutschland schätzte die Amerikanische Handelskammer in E... "International Business Machines Corporation or IBM is an American multinational technology and... software, and it offers infrastructure, hosting and consulting services in areas ranging from mainfr... through a merger of three companies: the Tabulating Machine Company, the International Time Rec... a name previously designated to CTR's subsidiary in Canada and later South America. Its distinctive... of membership information (422, 262). The //4 format is taken from the following: the //0 format is...</p>

**Figura 13 – Extrato do resultado da lista de todas as descrições da instância DBpedia “IBM” (Resultado da consulta da Figura 12, realizada no SPARQL Endpoint da Dbpedia).**

### 3.6 Considerações Finais

Nesse capítulo foram mostrados os principais conceitos de *web* semântica e dados ligados. No próximo capítulo será apresentada uma abordagem para enriquecimento dos dados do DW a partir de dados ligados.

## 4 Abordagens de Enriquecimento do DW

Nesse capítulo serão discutidas algumas formas de extração de dados RDF da *web*, bem como as maneiras que esses dados podem ser inseridos em um DW.

Na seção 4.1 serão apresentadas as diferentes abordagens para extração de dados estruturados em forma de RDF. Na seção 4.2 serão apresentadas as diferentes alternativas de se implementar e enriquecer o DW com os dados extraídos em RDF.

### 4.1 Formas de extração de dados RDF

Nessa seção serão apresentadas as diferentes abordagens para extração de dados estruturados em forma de RDF.

#### 4.1.1 *Websites* tradutores de páginas HTML para RDF

Uma das abordagens possíveis e possivelmente a mais simples de se extrair dados RDF da internet é através de *websites* que fazem a tradução automática de uma página HTML para RDF. A grande vantagem desse tipo de extração é que não requer nenhum conhecimento de SPARQL do usuário, que apenas insere a página HTML que deseja traduzir para dados RDF e o *website* cuida do resto. Além disso, a maioria deles permite que você escolha a forma que quer que os dados sejam apresentados. Pode-se escolher que os dados sejam apresentados em RDF/XML, N-Triples, JSON e outros formatos.

A grande desvantagem desse tipo de extração é que o usuário não consegue definir que tipos de dados ele quer de determinada página. Simplesmente ele recebe toda a informação contida no site em forma de RDF.

Alguns exemplos de *websites* que fazem essa tradução de HTML para RDF são o

*graphite*<sup>3</sup> e o *inspector*<sup>4</sup>. Utilizando o *inspector* e passando como parâmetro o site da *globo.com*, obtemos o resultado mostrado na Figura 14. Observa-se que o resultado é simples, capturando dados de formatação e metadados. Há também captura de dados semânticos, como título e subtítulo, porém em alguns testes realizados, dados do corpo do texto não foram retornados.

subject	predicate	object
<http://www.globo.com/>	dcterms:title	"globo.com - Absolutamente tudo sobre notícias, esportes e entretenimento"
<http://www.globo.com/>	xhv:icon	<http://s.qlbimg.com/en/ho/static/globocom2012/img/favicon.png>
<http://www.globo.com/>	xhv:stylesheet	<http://s.qlbimg.com/en/ho/static/CACHE/css/e6b541604490.css>
<http://www.globo.com/>	xhv:stylesheet	<http://s.qlbimg.com/en/ho/o/home/desktop/ajustes27.css>
<http://www.globo.com/>	xhv:bookmark	<http://revistaqloborural.globo.com/Revista/Common/0,EMI335795-18071,00-CAES+ALIVIAM+DURA+REALIDADE+DE+MORADORES+DE+RUA.html>
<http://www.globo.com/>	xhv:bookmark	<http://oqlobo.globo.com/educacao/juramento-inusitado-suspende-formatura-na-puc-8159113>
<http://www.globo.com/>	xhv:bookmark	<http://extra.globo.com/casos-de-policia/pm-do-choque-encontrado-morto-na-pavuna-fazia-surpresa-para-esposa-antes-de-ser-executado-8157767.html>

**Figura 14 - Resultado em formato de triplas do *website* da *globo.com* passado como parâmetro em um *website* de tradução de páginas HTML para RDF (http://inspector.sindice.com, 2013)**

#### 4.1.2 Consultas SPARQL

Outra abordagem possível usada na extração de dados RDF são as consultas através da linguagem SPARQL. A linguagem SPARQL tem uma sintaxe específica, baseada na sintaxe do SQL, mas que requer o conhecimento da ontologia em que os dados estão representados, e explora as características semânticas desta representação (EISENBERG *et al*, 2004).

Outra necessidade é de um software que permita a execução de consultas SPARQL na base semântica desejada. Uma opção é executar as consultas SPARQL em um *SPARQL endpoint*.

Utilizando uma base que possui uma ontologia simples de relações interpessoais como exemplo, para realizar uma consulta que retorne o nome dos melhores amigos de

<sup>3</sup> <http://graphite.ecs.soton.ac.uk/stuff2rdf>

<sup>4</sup> <http://inspector.sindice.com>

uma determinada pessoa (Paulo), podemos utilizar a consulta abaixo, mostrada na Figura 15:

```

1 PREFIX relacoes: <http://relacoes-interpessoais/predicados/>
2
3 PREFIX pessoas: <http://base-pessoas.com/pessoas/>
4
5 PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
6
7 SELECT ?nome
8 WHERE {
9     pessoas:paulo relacoes:melhoramigode ?melhoramigo .
10    ?melhoramigo rdfs:label ?name .
11 }
12

```

**Figura 15 - Consulta genérica SPARQL que retorna o nome dos melhores amigos de uma determinada pessoa (Um exemplo).**

As cinco primeiras linhas são utilizadas para definir os prefixos, de forma a tornar a consulta mais compacta e menos propensas a erros, de forma análoga ao *alias* definido no SQL. Dessa forma, “relacoes:” se torna uma abreviação de “<http://relacoes-interpessoais/predicados/>”, “pessoas:” uma abreviação de “<http://base-pessoas.com/pessoas/>” e “rdf:” uma abreviação de “<http://www.w3.org/2000/01/rdf-schema#>”. Na sétima linha é definido que a consulta deve retornar o resultado da variável “nome” (o ponto de interrogação antes de uma palavra indica que se trata de uma variável). Na oitava linha está a declaração da cláusula “WHERE”, bloco no qual se encontram as restrições. A primeira restrição define que existe uma variável “melhoramigo” que possui todos os recursos que estão na posição de objeto de uma tripla (sujeito, predicado, objeto) que possua “pessoas:paulo” como sujeito, “relacoes:melhoramigode” como predicado. A segunda restrição, apresentada na décima linha, define a variável “?melhoramigo” como sujeito, o predicado como “rdfs:label” e define o valor da variável “nome”, que aparece no select como retorno da consulta.

Dessa forma, ao guardar os resultados coletados através de consultas SPARQL, o usuário poderia usar ferramentas auxiliares para utilizar esses dados do DW, como ferramentas de Mineração de Dados para descobrir novos padrões e/ou ETL para tratamento e carga desses dados em algum *Data Mart*.

O ponto positivo desta abordagem é a forma como os dados são retornados, já que o usuário, ao montar suas consultas, escolhe que dados quer retornar de um determinado recurso. Já o principal ponto negativo é a complexidade que algumas consultas podem ter devido a necessidade de obtenção dos dados.

#### 4.1.3 Aplicações Pré-definidas

Uma maneira mais tradicional, e preferida por usuários com perfil de programador, é o acesso ao conteúdo semântico de bases RDF através do desenvolvimento de aplicações específicas. Assim, uma aplicação é implementada em alguma linguagem de programação para uma determinada base ou ontologia.

Existem alguns *frameworks* e API's que auxiliam no desenvolvimento dessas aplicações com o intuito de minimizar o trabalho do desenvolvedor no acesso ao dado.

Os *frameworks* mais conhecidos são o Jena (CARROL *et. al.*, 2004) e o R2R Framework (BIZER, 2010), ambos desenvolvidos para a linguagem Java. Jena é uma API Java que pode ser usada para criar e manipular grafos RDF como o apresentado na Figura 17. Jena possui classes para representar grafos, recursos, propriedades e literais, além de também possuir métodos para ler e escrever RDF como XML.

Um exemplo de como o Jena auxilia os desenvolvedores pode ser visto nas Figuras 16 e 17.

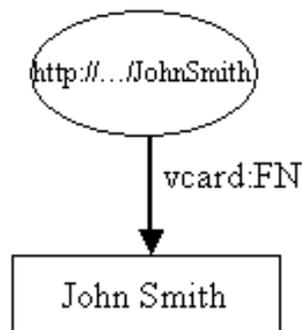


Figura 16 - Grafo de um *VCard* (cartão de negócios) do John Smith (JENA , 2010).

```

// some definitions
static String personURI    = "http://somewhere/JohnSmith";
static String fullName     = "John Smith";

// create an empty Model
Model model = ModelFactory.createDefaultModel();

// create the resource
Resource johnSmith = model.createResource(personURI);

// add the property
johnSmith.addProperty(VCARD.FN, fullName);

```

**Figura 17 - Representação do grafo RDF acima com o auxílio do *framework* Jena (JENA , 2010).**

Ele começa com algumas definições de constantes e então cria um Model vazio, usando o método `createDefaultModel()` de `ModelFactory` para criar um modelo na memória. O recurso John Smith é então criado e uma propriedade é adicionada a ele. A propriedade é fornecida pela classe "constante" `VCARD`, que mantém os objetos que representam todas as definições no esquema de `VCARD`.

O *framework* Jena, além de oferecer facilidades para a manipulação de dados em RDF como visto nos exemplos anteriores, também permite a realização de consultas a bases RDF utilizando o SPARQL e também a escrita e leitura de triplas RDF em diversos formatos, como RDF/XML e N-Triples.

Além dos *frameworks* para Java citados acima, existem também *frameworks* e API's para outras linguagens, como o ActiveRDF (OREN *et. al*, 2007), que provê uma camada de persistência para bases RDF para a linguagem Ruby (FLANAGAN *et. al*, 2008), e o RDFLib, desenvolvido para linguagem Python (SANNER, 1999).

## 4.2 Alternativas de implementação dos dados extraídos em RDF no DW

Nessa seção serão apresentadas algumas formas de implementação das informações extraídas em forma de RDF.

### 4.2.1 Nova dimensão

Existem algumas formas diferentes de implementar as informações extraídas em forma de RDF no *Data Warehouse*. Uma das principais maneiras é criar uma ou mais dimensões com as informações desejadas a fim de complementar e enriquecer o DW. Essa forma depende de como o modelo de dados está implementado. Caso o DW seja implementado como *star schema*, as dimensões mantêm relacionamentos com a tabela fato. Caso o DW seja implementado como *snowflake*, deve-se considerar a normalização dos dados.

### 4.2.2 Incremento de dimensão já existente

Outra maneira de enriquecer o DW é adicionando as informações desejadas em alguma dimensão já existente no modelo. Ao se falar nesse tipo de estratégia, temos duas possíveis soluções. A primeira, é a implementação das novas informações em forma de novas colunas; a segunda, mais simples, seria implementar o endereço html em questão como uma coluna adicional.

No primeiro caso, a dimensão já existente receberia as colunas da nova informação extraída, com os devidos controles para que essa nova informação complemente a informação correspondente já existente. Esse controle pode ser feito de diversas maneiras e cabe ao desenvolvedor, em tempo de programação e/ou desenvolvimento, escolher a melhor maneira.

No segundo caso, a dimensão já existente receberia apenas uma coluna nova, referente à URI que contém a informação que a complementaria. Similarmente, o mesmo tipo de controle deve ser feito a fim de manter a integridade dos dados e complementá-los da forma correta.

Exemplificando para o caso do DM de saúde, caso fossem resgatadas na *web* informações adicionais sobre os pacientes, contidas na URI `<http://base-pacientes.com/paciente_XPTO/>`, essas informações seriam adicionadas, através de novas colunas na dimensão de pacientes, caso a primeira solução apresentada fosse a escolhida. Caso a segunda solução fosse preferida, a dimensão existente seria enriquecida apenas com uma coluna contendo a URI: `<http://base-pacientes.com/paciente_XPTO/>`.

### 4.2.3 Incremento através de Metadados

Existe ainda a possibilidade de enriquecer o DW através de seus metadados, caso a intenção seja não implementar esses dados no esquema do DW, mas sim deixá-los armazenados de maneira mais discreta para eventuais consultas.

Dessa forma, usando como exemplo genérico a tabela de metadados que contém informações sobre todas as tabelas do banco de dados, a equivalente a “user\_tables” do Oracle, e com o intuito de enriquecer o modelo com dados de estados, poderia ser incluída uma coluna chamada “table\_description”, ou algo parecido, e, no registro referente a tabela de estado, ser incluída uma consulta SPARQL para resgatar dados semânticos de estados. Assim, os dados da tabela de estados estariam sendo enriquecidos, porém não no modelo, e sim nos metadados dessa tabela.

## 4.3 Considerações Finais

Nesse capítulo foram mostradas abordagens para o enriquecimento de um DW com dados em formato RDF. Foram mostradas algumas formas de extração dos dados, através de consultas SPARQL e/ou programas em linguagens como Java, Ruby e Python, além de serem apresentadas algumas alternativas de implementação desses dados nas tabelas do DW.

## 5 Estudo de Caso: *Data Mart* de Telecom

### 5.1 Introdução

Neste capítulo, será feita a apresentação do DW utilizado como base deste trabalho. Trata-se de um *Data Mart* com dados levemente alterados, por questões de privacidade e para uso nesse trabalho, de uma empresa de telecomunicações.

Para a criação das tabelas e armazenamento dos dados deste DM, foi utilizada uma versão gratuita do Oracle 11g: o Oracle XE. Apesar de ser limitado em tamanho máximo da base de dados (4 giga bytes no total), este limite acaba sendo suficiente para a criação de nossas tabelas e para a inserção de dados nas mesmas. A escolha do Oracle se deve à experiência anterior de uso pelos autores deste trabalho, além do fato dele ser facilmente configurável.

Para a realização da inserção de dados nas tabelas utilizadas neste trabalho, foram feitas cargas de dados no Oracle a partir de arquivos .txt criados pelos autores deste trabalho. Para tal fim, a aplicação de ETL Ascential Data Stage foi escolhida. Tal escolha foi feita pelo fato dos autores deste trabalho já possuírem experiência de desenvolvimento para esta aplicação.

Em nosso DM, que representa uma empresa fictícia de telecom, optamos por utilizar como assunto principal a recarga. A recarga de celular é feita quando um usuário de telefonia móvel pré-pago não possui mais créditos, por exemplo, para executar uma ligação, uma consulta na internet ou uma mensagem de texto. Para que este possa executar novas operações telefônicas, é necessário realizar antes uma recarga de créditos, daí o nome de “pré-pago”.

Tal assunto foi abrangido, pois a quantidade de usuários pré-pago supera em muito o de usuários pós-pago. Para se ter uma idéia, existem empresas de telefonia que possuem mais de 50 milhões de usuários pré-pago, enquanto os usuários pós-pago não

passam de 10 milhões.

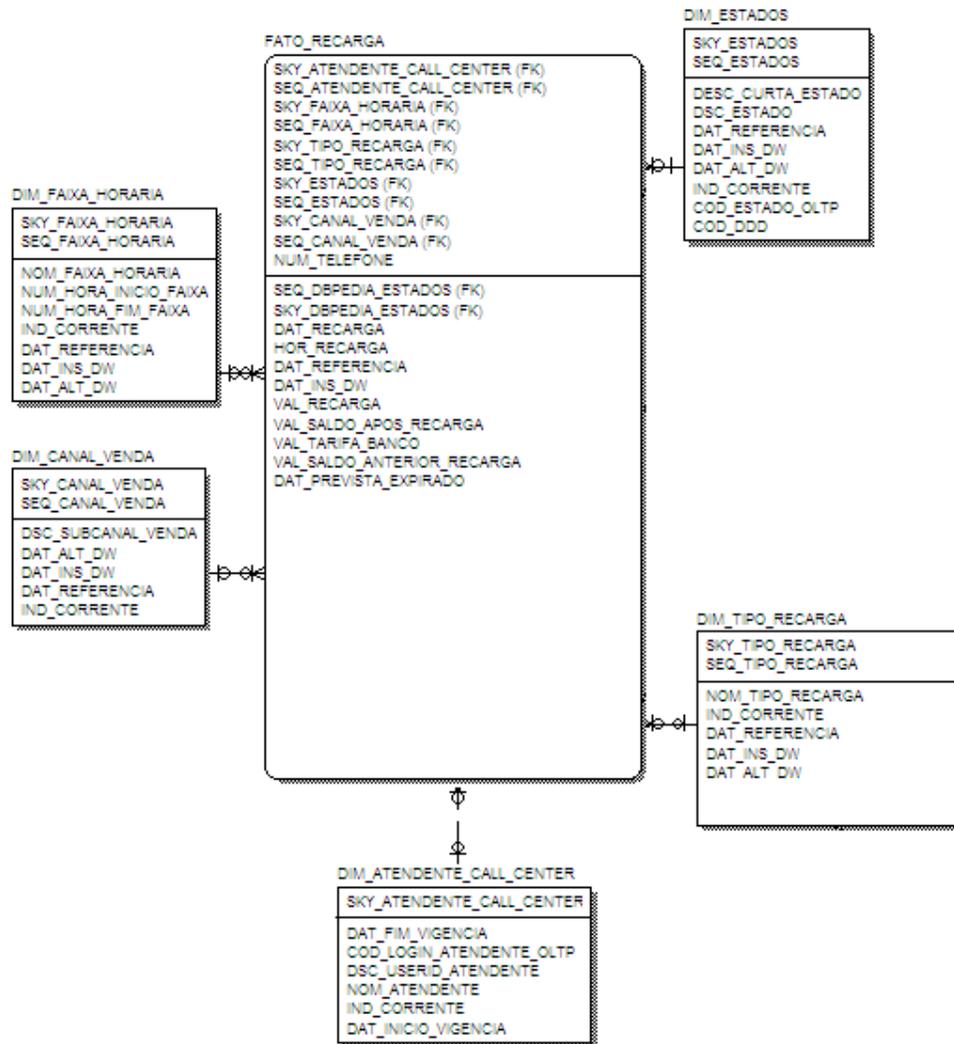
Tendo em vista o potencial de consumo que estes usuários proporcionam para as empresas de telefonia, optamos por abordar este assunto a fim de demonstrar como o cruzamento de dados de um DM de telecom, juntamente com dados extraídos de fontes RDF da *web*, podem ser utilizados para identificarmos potenciais usuários de celular ao redor do Brasil.

A seguir, uma explicação mais detalhada do modelo de dados e das tabelas do nosso DM.

## 5.2 Modelo de Dados

O modelo multi-dimensional foi utilizado, sendo implementado no formato *star schema*, como apresentado na Figura 18. No centro da estrela encontra-se a tabela de fatos e, ao seu redor, as dimensões.

No nosso DM de telecom, que abrange as recargas feitas pelos usuários, a tabela fato contém as recargas realizadas pelos consumidores (pré-pagos), e as dimensões ao redor dessa tabela de fato são referentes ao atendente, faixa horária, canal de venda, tipo de recarga e estados. Este esquema não abrange as tabelas extraídas da *DBpedia*. O modelo enriquecido, com as novas tabelas da *DBpedia*, será mostrado no próximo capítulo.



**Figura 18 - Esquema estrela do DM de telecom**

Em nosso DM de telecom, podemos analisar, por exemplo, que a maioria da população de São Paulo, registro identificado na dimensão “Estados”, costuma efetuar recargas entre 11:00 e 14:00 (Dimensão “Faixa Horária”), como podemos ver na consulta mostrada na Figura 19 e seu resultado na Figura 20.

```

select ft.sky_estados,
       est.desc_curta_estado,
       hor.nom_faixa_horaria,
       count(*) as qtd
from ft_recarga ft,
     system.dim_estados est,
     system.dim_faixa_horaria hor
where ft.sky_estados = est.sky_estados
and   ft.seq_estados = est.seq_estados
and   ft.sky_faixa_horaria = hor.sky_faixa_horaria
and   ft.seq_faixa_horaria = hor.seq_faixa_horaria
and   est.desc_curta_estado = 'SP'
group by ft.sky_estados, est.desc_curta_estado, hor.nom_faixa_horaria
order by qtd desc

```

**Figura 19 - Consulta com a quantidade de ligações de São Paulo agrupadas por faixa horária.**

	SKY_ESTADOS	DESC_CURTA_ESTADO	NOM_FAIXA_HORARIA	QTD
1	1	SP	Faixa entre 12:00:00 e 12:59:59 ...	907
2	1	SP	Faixa entre 13:00:00 e 13:59:59 ...	885
3	1	SP	Faixa entre 11:00:00 e 11:59:59 ...	838



**Figura 20 – Extrato do resultado da consulta com a quantidade de ligações de São Paulo agrupadas por faixa horária.**

### 5.3 Tabelas

Nesta seção serão apresentadas as tabelas do DM de telecom. Apesar de ser uma boa prática de desenvolvimento a implementação de aproximadamente 50 a 100 atributos em uma dimensão, segundo KIMBALL (2002), não achamos necessário implementar toda essa quantidade de atributos por se tratar de um exemplo, não uma implementação real, e também pelo fato das dimensões nesse trabalho serem usadas por apenas um *Data Mart*.

### 5.2.1 Dimensão Canal Venda

Esta dimensão é responsável por armazenar informações de Canais de venda de Recarga. Um canal de venda é o meio utilizado pelo usuário final para efetuar uma recarga de créditos em seu celular. A tabela contém campos de controle, comuns a quase todas as tabelas do DM, que são a *surrogate key* (sky), o sequencial do registro (seq), a data de inserção no DM, a data de referência do registro e o índice corrente, indicando o status, e tem também campos de informação, como a descrição do canal de venda. A Figura 21 mostra o conteúdo dessa tabela, mas optamos por omitir os primeiros campos de controle, *surrogate key* e sequencial, para que pudéssemos mostrar parte do conteúdo de maneira mais legível.

DAT_INS_DW	DAT_REFERENCIA	IND_CORRENTE	DSC_CANAL_VENDA_RECARGA
12/12/2008	18/7/2008	1	LOJAS PROPRIAS
12/12/2008	18/7/2008	1	BANCOS
12/12/2008	18/7/2008	1	LOTERICAS
12/12/2008	18/7/2008	1	LOJAS REGIONAIS
12/12/2008	18/7/2008	1	WEB

**Figura 21 - Conteúdo da dimensão Canal de Venda.**

### 5.2.2 Dimensão Tipo Recarga

Tabela de domínio responsável por armazenar os tipos de recargas possíveis dos clientes da nossa empresa de telecom. Atualmente, as empresas de telecom possuem inclusive pacotes de recargas que atendem apenas empresas. A tabela contém campos de controle, comuns às outras tabelas do DM e já explicados na dimensão de canal de venda, e também contém o campo com o nome do tipo de recarga. A Figura 22 mostra alguns exemplos de recarga implementados.

	SKY_TIPO_RECARGA	SEQ_TIPO_RECARGA	NOM_TIPO_RECARGA	IND_CORRENTE
1	-1	1	NÃO INFORMADO	1
2	0	1	NÃO SE APLICA	1
3	1	1	CARTAO FISICO	1
4	2	1	URA	1
5	3	1	RECARGA PADRAO	1
6	5	1	RECARGA COPA DO MUNDO	1
7	6	1	RECARGA PROGRAMADA	1
8	7	1	RECARGA GOL	1

**Figura 22 - Conteúdo da dimensão Tipo Recarga.**

### 5.2.3 Dimensão Faixa Horária

Esta dimensão é responsável por armazenar o horário em que o cliente efetua uma recarga em seu aparelho. Além dos campos de controle, também contém os campos de informação de nome da faixa horária, o início da faixa horária e o final da faixa horária. A Figura 23 mostra mais detalhes dessa dimensão.

	SKY_FAIXA_HORARIA	SEQ_FAIXA_HORARIA	NOM_FAIXA_HORARIA	NUM_HORA_INICIO_FAIXA	NUM_HORA_FIM_FAIXA
1	-1	1	NÃO INFORMADO	-1	-1
2	-2	1	NÃO SE APLICA	-2	-2
3	0	1	Faixa entre 00:00:00 e 00:59:59	00:00:00	00:59:59
4	1	1	Faixa entre 01:00:00 e 01:59:59	01:00:00	01:59:59
5	2	1	Faixa entre 02:00:00 e 02:59:59	02:00:00	02:59:59
6	3	1	Faixa entre 03:00:00 e 03:59:59	03:00:00	03:59:59
7	4	1	Faixa entre 04:00:00 e 04:59:59	04:00:00	04:59:59
8	5	1	Faixa entre 05:00:00 e 05:59:59	05:00:00	05:59:59
9	6	1	Faixa entre 06:00:00 e 06:59:59	06:00:00	06:59:59
10	7	1	Faixa entre 07:00:00 e 07:59:59	07:00:00	07:59:59
11	8	1	Faixa entre 08:00:00 e 08:59:59	08:00:00	08:59:59
12	9	1	Faixa entre 09:00:00 e 09:59:59	09:00:00	09:59:59
13	10	1	Faixa entre 10:00:00 e 10:59:59	10:00:00	10:59:59
14	11	1	Faixa entre 11:00:00 e 11:59:59	11:00:00	11:59:59
15	12	1	Faixa entre 12:00:00 e 12:59:59	12:00:00	12:59:59
16	13	1	Faixa entre 13:00:00 e 13:59:59	13:00:00	13:59:59
17	14	1	Faixa entre 14:00:00 e 14:59:59	14:00:00	14:59:59

**Figura 23 - Conteúdo da Dimensão Faixa Horária.**

### 5.2.4 Dimensão Atendente Call Center

Esta dimensão é responsável por armazenar o nome dos operadores de *call center* responsáveis por atender os clientes na hora de efetuar uma recarga por telefone. Além dos campos de controle, também contém os campos de código do login do atendente, a

descrição do atendente e o nome do atendente. A Figura 24 mostra mais detalhes dessa dimensão.

	COD_LOGIN_ATENDENTE_OLTP	DSC_USERID_ATENDENTE	NOM_ATENDENTE
1	Yuly12	YULY11	YULY12
2	ADMIN01	ADMIN02	OPERADOR ADMINISTRADOR
3	ADMINENG	ADMINENG	OPERADOR ADMINISTRADOR INGLES
4	ALLBRA	ALLBRA	OPERADOR HABILITADO PARA TODAS AS FUNÇÕES BRASI
5	ALLENG	ALLENG	OPERADOR HABILITADO PARA TODAS AS FUNÇÕES INGLE
6	TESTVB	VBITTENCOURT	OPERADOR TEST VB
7	Viviane	VIVIANE	VIVIANE
8	ADMINBR	ADMINBR	OPERADOR ADMINISTRADOR BRASILEIRO
9	mribeiro	MRIBEIRO	MARIA LUCIA RIBEIRO
10	maolivei	MAOLIVEI	MARIANA RAMOS DE OLIVEIRA
11	aoliveir	AOLIVEIR	ADRIANA APARECIDA DIAS DE OLIVEIRA
12	pcardoso	PCARDOSO	PATRICIA FABIOLA CARDOSO DA SILVA
13	aciriaco	ACIRIACO	ADRIANO REIS CIRIACO
14	prodrigu	PRODRIGU	PAULO RENE FRAGA RODRIGUES
15	prsouza	PRSOUZA	PRISCILLA GONÇALVES DE SOUZA
16	alfernan	ALFERNAN	ALEXANDRE CASTRO FERNANDES
17	rasilva	RASILVA	RADAMES ALVES DA SILVA
18	asouza	ASOUZA	ALEXANDRE SILVA DE SOUZA
19	raneres	RANERES	RAIMUNDO NEVES
20	acinta	ACINTA	ALINE BRAZ COSTA
21	refreita	REFREITA	REGIANE KROLL MARQUES DE FREITAS
22	PRESS	PRESS	OPERAT ORE PRESS
23	avieira	AVIEIRA	AMANDA DE FATIMA VIEIRA

**Figura 24 - Conteúdo da Dimensão Call Center.**

### 5.2.5 Dimensão Estados

Dimensão responsável por Armazenar informações básicas, como sigla e nome, dos estados Brasileiros. Além dos campos de controle, ela também contém campos como sigla do Estado, denominado como “desc\_curta\_estado”, o nome do estado por extenso, denominado por “desc\_estado”, o código OLTP do estado, em ordem numérica em que foi cadastrado, e o código de DDD relacionado a este estado. Em nosso DW optamos por utilizar apenas um código de DDD por estado, visto que nossa futura intenção em trabalhos futuros é apresentar indicadores de negócio divididos por estados do Brasil, não tornando necessário o uso de vários códigos para isso. A Figura 25 mostra mais detalhes dessa dimensão.

DESC_CURTA_ESTADO	DSC_ESTADO	DAT_REFERENCIA	DAT_INS_DW	DAT_ALT_DW	COD_ESTADO_OLTP	COD_DDD
AC	Acre	5/2/2013	5/2/2013	5/2/2013	1	68
AL	Alagoas	5/2/2013	5/2/2013	5/2/2013	2	82
AP	Amapá	5/2/2013	5/2/2013	5/2/2013	3	96
AM	Amazonas	5/2/2013	5/2/2013	5/2/2013	4	92
BA	Bahia	5/2/2013	5/2/2013	5/2/2013	5	71
CE	Ceará	5/2/2013	5/2/2013	5/2/2013	6	88
DF	Distrito Federal	5/2/2013	5/2/2013	5/2/2013	7	61
ES	Espírito Santo	5/2/2013	5/2/2013	5/2/2013	8	27
GO	Goiás	5/2/2013	5/2/2013	5/2/2013	9	62
MA	Maranhão	5/2/2013	5/2/2013	5/2/2013	10	98
MT	Mato Grosso	5/2/2013	5/2/2013	5/2/2013	11	65
MS	Mato Grosso do Sul	5/2/2013	5/2/2013	5/2/2013	12	67
MG	Minas Gerais	5/2/2013	5/2/2013	5/2/2013	13	31
PA	Pará	5/2/2013	5/2/2013	5/2/2013	14	91
PB	Paraíba	5/2/2013	5/2/2013	5/2/2013	15	83
PR	Paraná	5/2/2013	5/2/2013	5/2/2013	16	41
PE	Pernambuco	5/2/2013	5/2/2013	5/2/2013	17	81
PI	Piauí	5/2/2013	5/2/2013	5/2/2013	18	86
RJ	Rio de Janeiro	5/2/2013	5/2/2013	5/2/2013	19	21
RN	Rio Grande do Norte	5/2/2013	5/2/2013	5/2/2013	20	84
RS	Rio Grande do Sul	5/2/2013	5/2/2013	5/2/2013	21	53
RO	Rondônia	5/2/2013	5/2/2013	5/2/2013	22	69
RR	Roraima	5/2/2013	5/2/2013	5/2/2013	23	95
SC	Santa Catarina	5/2/2013	5/2/2013	5/2/2013	24	48
SP	São Paulo	5/2/2013	5/2/2013	5/2/2013	25	11
SE	Sergipe	5/2/2013	5/2/2013	5/2/2013	26	79
TO	Tocantins	5/2/2013	5/2/2013	5/2/2013	27	63

**Figura 25 - Conteúdo da Dimensão Estados.**

### 5.2.6 Fato Recarga

Em nossa tabela de fato, guardamos os dados históricos provenientes de nossas dimensões. Ela contém todos os dados relacionados às recargas realizadas, ou seja, ela contém o fato que nosso DM trata: a recarga. Contém informações de todas as dimensões, como a hora da recarga, o atendente que realizou a recarga, em que estado ela foi feita, o tipo de recarga realizado e o canal de venda utilizado pelo cliente. Dessa forma, podemos identificar, por exemplo, qual o horário mais frequente de recargas feitas em um dia, no estado do Rio de Janeiro. Podemos identificar também, qual estado teve um grande crescimento em termos de recarga em determinado mês, entre outros indicadores de negócio. A Figura 26 mostra mais detalhes da tabela fato.

NUM_TELEFONE	SKY_FAIXA_HORARIA	SEQ_FAIXA_HORARIA	SKY_TIPO_RECARGA	SEQ_TIPO_RECARGA	SKY_ESTADOS	SEQ_ESTADOS	VAL_RECARGA	HOR_RECARGA
1189129167	8	1	2	1	1	1	12,00	81918
1189193700	9	1	8	1	1	1	32,00	93705
1189195007	13	1	2	1	1	1	12,00	134606
1189198369	0	1	8	1	1	1	32,00	1824
1189223999	11	1	2	1	1	1	12,00	112658
1189246603	13	1	2	1	1	1	12,00	131500
1189435555	17	1	2	1	1	1	17,00	175202
1189539000	1	1	8	1	1	1	32,00	12418
1189661412	15	1	2	1	1	1	35,00	152426
1189674195	14	1	2	1	1	1	17,00	142452
1189709502	18	1	8	1	1	1	32,00	183429
1189729106	14	1	2	1	1	1	26,00	142020
1189731477	19	1	2	1	1	1	12,00	190322
1189750058	20	1	2	1	1	1	17,00	203938
1189898836	10	1	2	1	1	1	12,00	104429
2180100467	18	1	2	1	43	1	6,00	182059
2180100941	21	1	2	1	43	1	26,00	210811
2180101981	13	1	12	1	43	1	6,00	134206
2180102158	12	1	2	1	43	1	6,00	121819
2180102380	11	1	12	1	43	1	6,00	113545
2180102734	19	1	2	1	43	1	6,00	195246
2180103641	17	1	12	1	43	1	6,00	173100
2180104733	18	1	12	1	43	1	6,00	182041
2180105727	13	1	2	1	43	1	6,00	131936
2180106687	22	1	13	1	43	1	12,00	222930

**Figura 26 - Conteúdo da Fato Recarga.**

## 5.4 Considerações Finais

Neste capítulo, foram mostradas as tabelas do nosso DW, assim como uma breve descrição do negócio das mesmas e seus objetivos. Foram apresentados também, alguns exemplos de dados pertencentes a estas tabelas. Por fim, foi demonstrado um exemplo de indicador de negócio de nosso DW, o que foi possível demonstrar através do relacionamento entre nossas tabelas de dimensões, com a nossa tabela de fato.

## 6 Enriquecendo o DM de Telecom

Nesse capítulo serão mostradas as etapas de enriquecimento do DM construído com informações da *Wikipedia*.

Na seção 6.1 será apresentada a abordagem para extração dos dados estruturados em forma de RDF da *DBpedia*. Já na seção 6.2, será apresentada a forma como essas informações foram implementadas, a ferramenta usada e como os dados ficaram após a implementação.

### 6.1 Tipo de Extração Escolhida

A forma de extração dos dados RDF escolhida para esse trabalho foi a criação de consultas SPARQL para resgatar essas informações da *DBpedia*. Essa forma de extração se mostrou a mais eficiente, pois permite uma escolha e um controle maior dos dados que se deseja resgatar.

Os dados escolhidos para enriquecer o DM de Telecom foram dados geográficos, pois acreditamos que esse tipo de dados seja um dos mais importantes para análises mercadológicas para esse nicho. A partir desse tipo de dados pode-se fazer análises correlacionando as vendas de aparelhos mais caros, como *smartphones* de última geração, com as cidades e/ou estados que se tem uma maior população, ou ainda, com as cidades e/ou estados que tem uma maior renda *per capita*.

Outro tipo de análise que o enriquecimento com dados geográficos permite são ações relacionadas também com venda de planos e ações de marketing. Por exemplo, por que não aumentar o marketing em locais que se tem uma grande população, e assim, mais chances de conquistar novos clientes e manter os clientes já estabelecidos?

Através das possibilidades que os dados geográficos permitem, escolhemos por

trazer esses tipos de dados e foram montadas 2 tipos de consulta SPARQL. A primeira, mostrada na Figura 27, seleciona todos os estados que sejam recursos do tipo “States\_Of\_Brazil”, ou seja, todos os recursos que tenha uma tripla de predicado *http://dbpedia.org/ontology/type* e objeto *http://dbpedia.org/resource/States\_of\_Brazil*.

```

PREFIX dcterms: <http://purl.org/dc/terms/>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX dbo: <http://dbpedia.org/ontology/>
PREFIX dbp: <http://dbpedia.org/property/>
PREFIX dbrec: <http://dbpedia.org/resource/>

SELECT ?url_dbp_estado
       ?nom_estado
       ?areaTotal
       ?country
       ?coordinatesRegion
       ?website
       ?comment
WHERE{
  ?url_dbp_estado dbo:type dbrec:States_of_Brazil;
    rdfs:label ?nom_estado ;
    dbo:areaTotal ?areaTotal ;
    dbo:country ?country ;
    dbp:coordinatesRegion ?coordinatesRegion ;
    dbp:website ?website ;
    rdfs:comment ?comment
  FILTER(LANG(?nom_estado) = "pt")
  FILTER(LANG(?comment) = "pt")
}

```

**Figura 27 - Consulta SPARQL que resgata alguns dados dos Estados Brasileiros. Consulta realizada em maio de 2013.**

## 6.2 Por que as outras formas de extração não foram escolhidas?

Outros tipos de extração foram testados. A primeira tentativa foi tentar extrair essas informações através de sites feitos exatamente com esse intuito: transformar uma URL qualquer para dados RDF. Essa alternativa foi descartada, pois, além de não termos o controle dos dados a serem trazidos, os dois sites testados apresentavam diversas instabilidades, como problemas na execução da transformação apresentando lentidão e, por algumas tentativas, o site não executava a solitação de transformação.

A segunda tentativa foi com um programa fornecido pelo Eduardo Fritzen, autor da tese de mestrado “Recuperação Contextual de Informação na *Web* a partir da Análise de Mensagens e Enriquecimento em Dados Abertos: Explorando o contexto de ensino/aprendizagem”. O programa original é feito em Java e trabalha com o *framework* Jena para resgatar as informações do *DBpedia*, apresentadas na tela em tempo de execução. Alteramos esse programa para que, ao invés de mostrar as informações na tela, criasse um arquivo em formato “.txt” que serviria de entrada no processo ETL. Era passado o recurso desejado (ex: [http://dbpedia.org/page/Rio\\_de\\_Janeiro\\_\(state\)](http://dbpedia.org/page/Rio_de_Janeiro_(state))), e ele criava um arquivo com todas as triplas desse recurso. Porém a dificuldade de se entender o *framework* pré-definido aliada ao engessamento das informações, pois programar o *framework* para resgatar determinadas triplas utilizando filtros de linguagem e a não obrigatoriedade de algum campo (equivalente ao *is not null* do SQL), fizeram com que desistíssemos dessa maneira.

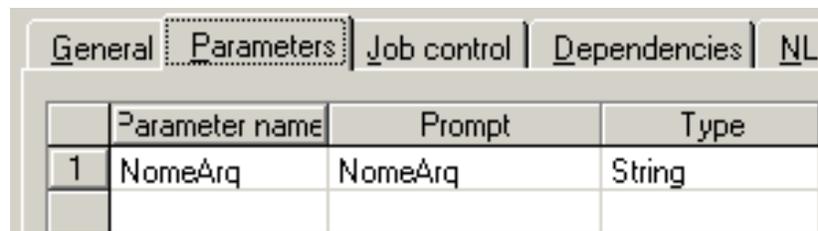
A terceira alternativa, e uma das mais interessantes, foi na verdade uma tentativa intermediária entre as extrações com consultas SPARQL e o ETL. Foi usado um software de manipulação de grafos RDF, chamado *AllegroGraph*, para que o trabalho fosse complementado com algumas análises que nos ajudasse a identificar alguns padrões e características similares entre os recursos. Porém, preparar esse ambiente exigiu certo trabalho e algumas frustrações pelo meio do caminho. A começar pelo *software*, mais estável e parrudo em Linux, necessitando a criação de uma máquina virtual, já que os dois integrantes desse trabalho possuem apenas *Windows* e *OS X*. Passada a dificuldade inicial, o maior problema foi em adequar o arquivo com os dados da *DBpedia* que serviria de entrada no *AllegroGraph*, possibilitando a criação da biblioteca de triplas (*triple store*). Tanto o XML/RDF quanto o arquivo em N-Triplas (extensão *.ntriples*) são gerados pelo SPARQL *Endpoint* da *DBpedia* de uma forma muito peculiar. Os nós recebem uma numeração, o que torna inviável a análise direta pelo *AllegroGraph*, já que o programa passa a lidar com os nós pela sua numeração, e não pelo seu verdadeiro nome. Outra tentativa de inserir os dados dentro do *AllegroGraph* foi criando um arquivo de N-Triplas manualmente, porém o grande esforço e a não garantia de sucesso tornaram essa opção inviável. Além dos arquivos terem de estar em formato *ntriples*, ainda assim muitos apresentaram diversos erros, mesmo com todo o suporte dos analistas da Franz, empresa que detém o *AllegroGraph*.

### 6.3 Ferramenta ETL Usada

Após os dados terem sido extraídos, eles necessitavam um tratamento e uma maneira de serem inseridos no DM, que geralmente são feitos por ferramentas ETL.

A ferramenta ETL escolhida por nós é uma ferramenta da IBM, empresa que trabalhamos, e que estamos bastante familiarizados. A ferramenta se chama *DataStage* e faz parte do pacote *Ascential Infosphere*, que tem outras soluções. Cada processo desenvolvido nessa ferramenta é chamado de *job*. Uma boa prática de desenvolvimento é não sobrecarregar os *jobs*, ou seja, separar em etapas os diferentes processos a fim de otimizar a execução do processo e facilitar uma eventual manutenção.

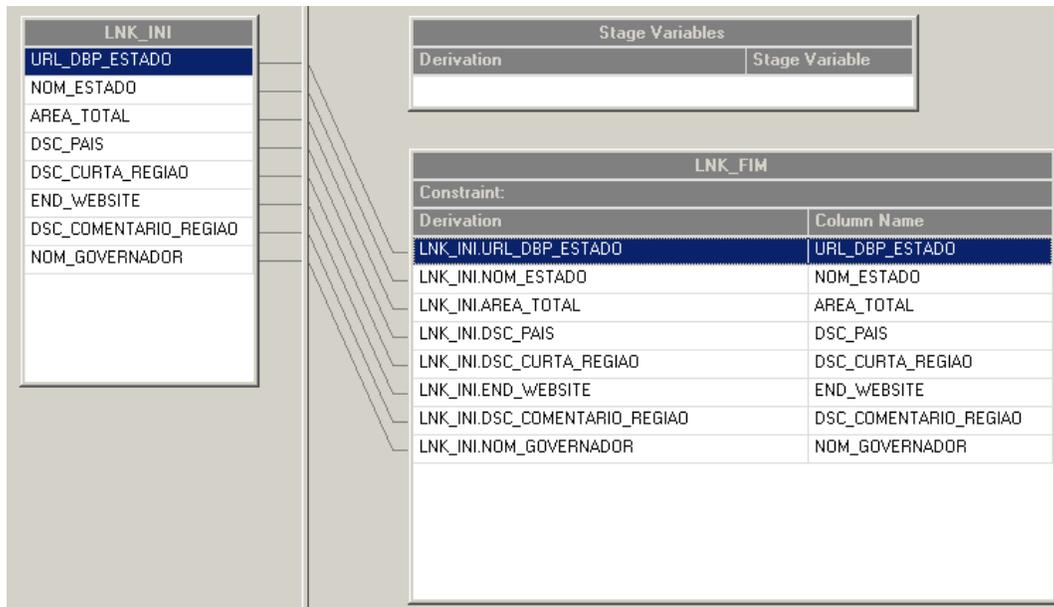
Foram criados dois *jobs*: um *job* de *stage* e outro *job* de inserção de fato no DM. O *job* de *stage* simplesmente lê e carrega os dados dos arquivos de entrada, que no nosso caso são os arquivos extraídos da *DBpedia*. Dessa forma, foi criado um *job* de *stage* que carrega os dados referentes aos estados brasileiros em uma tabela de *stage*, sem nenhum tipo de tratamento, que faz o papel de espelho das informações, e outro *job* genérico similar para as cidades. Como são vários arquivos diferentes de estados mas todos com a mesma estrutura (mesmo formato, quantidade de colunas e etc), o *job* genérico recebe o nome do arquivo de origem por parâmetro. Assim, o mesmo *job* consegue lidar com todos os arquivos de estados, como mostra a Figura 28.



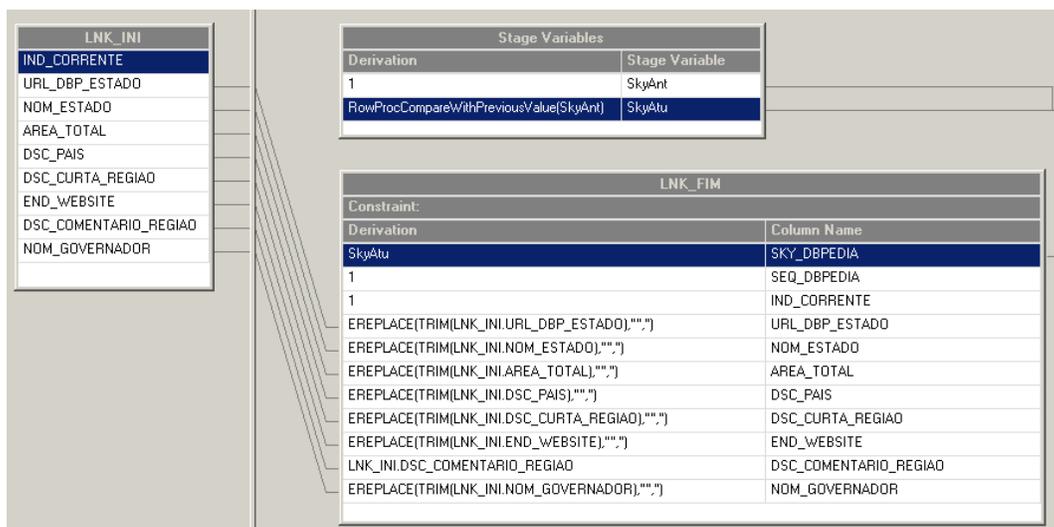
	Parameter name	Prompt	Type
1	NomeArq	NomeArq	String

**Figura 28 - Arquivo de origem passados como parâmetro.**

O outro *job*, que insere os dados de fato no DM, lê os dados da tabela de *stage*, aplica algumas transformações e carrega os dados na respectiva dimensão Estado. As Figuras 29 e 30 mostram um exemplo de cada *job*.



**Figura 29 - Job que carrega as informações do arquivo de origem para uma tabela temporária (stage), sem nenhum tratamento**



**Figura 30 - Job que carrega as informações da tabela temporária (stage) para a dimensão do DM, com o controle de versão e alguns tratamentos.**

#### 6.4 Técnica de implementação dos dados na FATO.

Para atualizarmos a chave estrangeira da tabela de Fato de nosso DM, foi utilizado o campo COD\_DDD\_ESTADOS, vindo da dimensão DIM\_ESTADOS. Dessa maneira,

foi possível bater os dados extraídos da *web* com os dados certos na fato, a fim de evitar uma atualização errada, como por exemplo, incrementar um registro referente a SP com dados do RJ.

Para atualização desses dados na tabela de fatos, foram utilizadas as consultas apresentadas na Figura 31:

```

--PASSO1
INSERT INTO TEMP1
SELECT EST.DSC_ESTADO,
       EST.COD_DDD
FROM   SYSTEM.DIM_ESTADOS EST,
       FT_RECARGA           FT
WHERE  EST.SKY_ESTADOS = FT.SKY_ESTADOS
       AND EST.SEQ_ESTADOS = FT.SEQ_ESTADOS
       AND EST.IND_CORRENTE = 1

--PASSO2
INSERT INTO TEMP2
SELECT DBP.SKY_DBPEDIA, DBP.SEQ_DBPEDIA, TEMP1.COD_DDD
FROM   TEMP1 TEMP1,
       SYSTEM.DIM_DBPEDIA DBP
WHERE  TEMP1.DSC_ESTADO = DBP.NOM_ESTADO
       AND DBP.IND_CORRENTE = 1

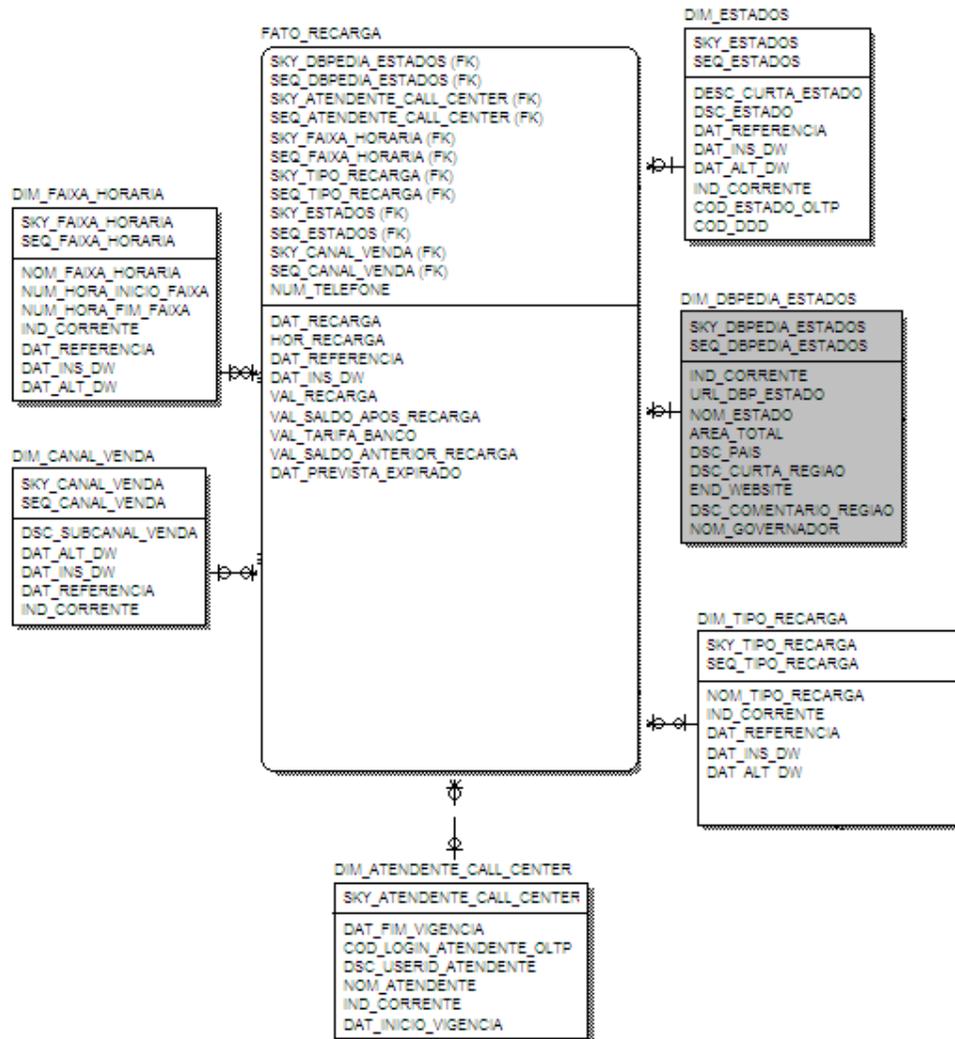
--UPDATE
UPDATE FT_RECARGA FATO
SET   ( FATO.SKY_DBPEDIA = TEMP2.SKY_DBPEDIA ), ( FATO.SEQ_DBPEDIA = TEMP2.SEQ_DBPEDIA )
WHERE TEMP2.COD_DDD = SUBSTR(FATO.NUM_TELEFONE,1,2)

```

**Figura 31 - Consultas usadas para atualização dos dados na fato de recargas**

## 6.5 Novo Esquema

Neste novo esquema enriquecido, apresentado na Figura 33, podemos notar a adição de uma nova tabela: a tabela de dimensão DIM\_DBPEDIA\_ESTADOS (na qual existem dados de estados brasileiros)



**Figura 32 - Novo modelo de dados do DM de *telecom* após o enriquecimento**

A Figura 33 mostra em detalhes a nova tabela do modelo. A tabela DIM\_DBPEDIA\_ESTADOS, além dos dados de controle do DM, contém campos como a *url* da *DBpedia* usada para extrair a informação, o nome do estado, área total dele, o país pertencente, o *website* do estado, e um campo descritivo.

SKY_DBPEDIA	SEQ_DBPEDIA	IND_CORRENTE	URL_DBP_ESTADO	NOM_ESTADO	AREA_TOTAL	DSC_PAIS
1	2	1	http://dbpedia.org/resource/Rio_de_Janeiro_(state)	Rio de Janeiro	43.696.100.000	http://dbpedia.org/resource/Brazil
2	1	0	http://dbpedia.org/resource/Rio_de_Janeiro_(state)	Rio de Janeiro	43.696.100.000	http://dbpedia.org/resource/Brazil
3	2	1	http://dbpedia.org/resource/Cear%C3%A1	Ceará	146.348.300.000	http://dbpedia.org/resource/Brazil
4	1	0	http://dbpedia.org/resource/Cear%C3%A1	Ceará	146.348.300.000	http://dbpedia.org/resource/Brazil
5	2	1	http://dbpedia.org/resource/Goi%C3%A1s	Goiás	340.086.000.000	http://dbpedia.org/resource/Brazil
6	1	0	http://dbpedia.org/resource/Goi%C3%A1s	Goiás	340.086.000.000	http://dbpedia.org/resource/Brazil
7	2	1	http://dbpedia.org/resource/Par%C3%A1	Pará	1.247.689.500.000	http://dbpedia.org/resource/Brazil
8	1	0	http://dbpedia.org/resource/Par%C3%A1	Pará	1.247.689.500.000	http://dbpedia.org/resource/Brazil
9	1	1	http://dbpedia.org/resource/Piau%C3%AD	Piauí	251.529.186.000	http://dbpedia.org/resource/Brazil
10	1	1	http://dbpedia.org/resource/Tocantins	Tocantins	277.620.910.000	http://dbpedia.org/resource/Brazil
11	1	0	http://dbpedia.org/resource/Esp%C3%ADrito_Santo	Espírito Santo (estado)	46.077.519.000	http://dbpedia.org/resource/Brazil
12	2	1	http://dbpedia.org/resource/Esp%C3%ADrito_Santo	Espírito Santo (estado)	46.077.519.000	http://dbpedia.org/resource/Brazil
13	2	1	http://dbpedia.org/resource/Rio_Grande_do_Sul	Rio Grande do Sul	281.748.000.000	http://dbpedia.org/resource/Brazil
14	1	0	http://dbpedia.org/resource/Rio_Grande_do_Sul	Rio Grande do Sul	281.748.000.000	http://dbpedia.org/resource/Brazil
15	1	0	http://dbpedia.org/resource/Santa_Catarina_(state)	Santa Catarina	95.346.181.000	http://dbpedia.org/resource/Brazil
16	2	1	http://dbpedia.org/resource/Santa_Catarina_(state)	Santa Catarina	95.346.181.000	http://dbpedia.org/resource/Brazil
17	3	1	http://dbpedia.org/resource/Sergipe	Sergipe	21.910.348.000	http://dbpedia.org/resource/Brazil
18	2	0	http://dbpedia.org/resource/Sergipe	Sergipe	21.910.348.000	http://dbpedia.org/resource/Brazil
19	1	0	http://dbpedia.org/resource/Sergipe	Sergipe	21.910.348.000	http://dbpedia.org/resource/Brazil
20	2	1	http://dbpedia.org/resource/Amazonas_(Brazilian_state)	Amazonas	1.570.745.700.000	http://dbpedia.org/resource/Brazil
21	1	0	http://dbpedia.org/resource/Amazonas_(Brazilian_state)	Amazonas	1.570.745.700.000	http://dbpedia.org/resource/Brazil
22	2	1	http://dbpedia.org/resource/Bahia	Bahia	567.295.000.000	http://dbpedia.org/resource/Brazil
23	1	0	http://dbpedia.org/resource/Bahia	Bahia	567.295.000.000	http://dbpedia.org/resource/Brazil
24	1	0	http://dbpedia.org/resource/S%C3%A3o_Paulo_(state)	Sao Paulo	248.209.400.000	http://dbpedia.org/resource/Brazil
25	2	1	http://dbpedia.org/resource/S%C3%A3o_Paulo_(state)	Sao Paulo	248.209.400.000	http://dbpedia.org/resource/Brazil
26	2	1	http://dbpedia.org/resource/Para%C3%ADba	Paraíba	56.584.600.000	http://dbpedia.org/resource/Brazil
27	1	0	http://dbpedia.org/resource/Para%C3%ADba	Paraíba	56.584.600.000	http://dbpedia.org/resource/Brazil
28	2	1	http://dbpedia.org/resource/Paran%C3%A1_(state)	Paraná	199.314.900.000	http://dbpedia.org/resource/Brazil
29	1	0	http://dbpedia.org/resource/Paran%C3%A1_(state)	Paraná	199.314.900.000	http://dbpedia.org/resource/Brazil

Figura 33 - Conteúdo da dimensão DIM\_DBPEDIA\_ESTADOS

## 6.6 Considerações Finais

Nesse capítulo foi apresentada a forma escolhida para extração das informações na em RDF da *web* e a maneira como implementamos essa informação no nosso DW de *telecom*, desde as consultas usadas até a forma como os dados foram associados aos da fato existente no ambiente.

## 7 Conclusão

Nesse capítulo serão mostrados as contribuições e limitações desse estudo para a comunidade. Também será comentado o desejo de trabalho futuro a partir do que foi construído neste projeto

### 7.1 Contribuições

A maior contribuição desse trabalho é a disponibilização de uma gama maior de dados para análises mais completas e profundas. Muito se fala hoje em dia de *Analytics* e *Big Data* pelas grandes empresas. Esse tipo de enriquecimento é muito útil para as empresas conhecerem mais sobre seus clientes e poderem ajustar, adaptar e precificar cada vez melhor seus produtos e serviços.

A apresentação das diversas técnicas de implementação auxilia e encoraja outras pessoas e empresas a tentarem fazer o mesmo com o intuito de aprenderem mais sobre seus negócios, produtos, serviços e clientes.

### 7.2 Trabalhos Futuros

A interpretação e análise dos dados externos em formato RDF implementados ao DW darão um novo conhecimento às empresas sobre seus clientes e produtos. Como trabalho futuro, deseja-se moldar melhor as informações colhidas para tentar extrair

informações possivelmente escondidas dentro do DW. O escopo deste trabalho não incluiu as análises e cruzamentos dos dados colhidos com os dados já existentes, porém esse é o desejo futuro.

Empresas de *telecom* podem cruzar dados geográficos e dados demográficos para saber que tipo de promoções e planos podem oferecer a seus clientes de acordo com sua renda, bem como empresas de seguro podem cruzar dados internos de seus clientes com dados geográficos e dados estatísticos para precificar melhor seus produtos de acordo com a área e índices de sinistros do cliente.

Há um vasto leque de possibilidades com o que se pode fazer com os dados disponibilizados em RDF e cada vez mais teremos novos dados disponibilizados nesse formato, cabendo a nós (consultores, analistas e profissionais da área analítica) saber fazer uso deles.

### 7.3 Limitações do Estudo

A maior limitação nesse estudo foi a qualidade dos dados apresentados em formato RDF na *web*. Apesar dos esforços da comunidade de TI serem de imensa importância e ter uma certa padronização, algumas informações ainda se encontram de forma desorganizada.

Muitos dos recursos escolhidos, como dados de estados, cidades e bairros, não possuem padrão. Se for comparado dois recursos do mesmo tipo (dois estados por exemplo), percebe-se que um deles tem informações, como propriedades e formatos, completamente diferentes do outro. Isso se deve ao fato da *DBpedia* ser um *dump* das informações da *Wikipedia*, que por sua vez é editada por usuários da *web*, consistindo em um esforço coletivo e assim apresentando uma dificuldade de padronização da informação.

Outro problema encontrado foi a pouca variedade de dados RDF na *web*. Apesar desse esforço estar sendo disseminado cada vez mais, ainda existem muitas pessoas da própria área de TI que desconhecem o significado de RDF e,

consequentemente não têm idéia do poder que a utilização dos dados disponibilizados dessa maneira pode oferecer.

## Referências

AUER, S. et. al. DBpedia: A Nucleus for a Web of Open Data. In Proceedings of ISWC 2007, Busan, Korea., 2007

BERNERS-LEE, T., HENDLER, J. and LISSILA, O. 2001. The Semantic Web. [Online]. Disponível em: <http://sciam.com/article.cfm?articleID=00048144-10D2-1C70-84A9809EC588EF21>. Acesso em maio de 2013.

BERNERS-LEE, T. Linked data. Design Issues. Julho, 2006. Disponível em: <http://www.w3.org/DesignIssues/LinkedData.html>. Acesso em: junho de 2013. Acesso em junho de 2013.

BERNERS-LEE, T. Putting government data online. Design Issues. Junho, 2009. Disponível em: <http://www.w3.org/DesignIssues/GovData.html>. Acesso em: junho de 2013.

BIZER, C.; HEATH, T.; IDEHEN, K.; BERNERS-LEE, T. Linked data on the web. In: LDOW 2008 - International World Wide Web Conference. 17., Beijing, Proceedings... Abril, 2008. Disponível em: <http://events.linked data.org/ldow2008/papers/00-bizer-heath-ldow2008-intro.pdf>. Acesso em: junho de 2013.

BIZER, C., SCHULTZ, A.: The R2R Framework: Publishing and Discovering Mappings on the Web. 1st International Workshop on Consuming Linked Data (COLD 2010), Shanghai, November 2010.

CARROLL, J. J., DICKINSON, I., DOLLIN, C., REYNOLDS, D., SEABRNE, A., e WILKINSON, K. (2004). Jena: implementing the semantic web recommendations. In International World Wide Web Conference, 2004, New York, USA.

DBpedia. <<http://pt.dbpedia.org/>>. Acesso em maio de 2013.

Diagrama de Dados ligados abertos, por Richard Cyganiak e Anja Jentzsch. <<http://lod-cloud.net/>>. Acesso em junho de 2013.

FLANAGAN, D., MATSUMOTO, Y. The Ruby Programming Language. 1 ed. O'Reilly Media, 2008

HEBELER, John et al. Semantic Web Programming. 3 ed. Wiley: New York, 2009.

INMON, W. Building the Data Warehouse. 4 ed. Wiley: , 2005.

JENA , 2010. <[http://jena.sourceforge.net/tutorial/RDF\\_API/#glos-Resource](http://jena.sourceforge.net/tutorial/RDF_API/#glos-Resource)>. Acesso em junho de 2013.

KIMBALL, R. The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling. 2 ed. Wiley, 2002.

LONDEIX, B. Evaluating the quality of entity relationship models, Information and Software Technology, V.37, 1995.

MACHADO, F. N. R. Tecnologia e Projeto de Data Warehouse: uma visão multidimensional. 4 ed. São Paulo: Ética, 2008.

OLAP Council. < [http://www.symcorp.com/downloads/OLAP\\_CouncilWhitePaper.pdf](http://www.symcorp.com/downloads/OLAP_CouncilWhitePaper.pdf) >. Acesso em agosto de 2013.

OREN, E. et. al. ActiveRDF: object-oriented semantic web programming. Eyal Oren, Renaud Delbru, Sebastian Gerke, Armin Haller, and Stefan Decker. ACM, (2007)

PRUD'HOMMEAUX, E., SEABORNE, A. SPARQL query language for RDF. W3C Recommendation, 2008. Disponível em: <<http://www.w3.org/TR/rdf-sparql-query>>. Acesso em maio de 2013.

SANNER, M. Python: a programming language for software integration and development. *J. Mol. Graph Model*, 17, 57–61.

TAN, Pang-Ning; STEINBACH, Michael; KUMAR, Vipin. *Introdução ao Data Mining Mineração de Dados*. Rio de Janeiro: Editora Ciência Moderna. 2009. 900 p.

THE NEW YORK TIMES. <http://www.nytimes.com/>>. Acesso em: mar. 2010.

W3C, World Wide Web Consortium, recomendações de 2004.

<<http://www.w3.org/TR/rdf-primer/>>. Acesso em maio de 2013.

Wikipedia. <<http://www.wikipedia.org/>>. Acesso em maio de 2013.