1

Introdução

Este capítulo contextualiza os principais assuntos abordados neste trabalho de dissertação, apresenta as motivações que levaram à escolha do tema, os objetivos gerais e específicos da proposta desta pesquisa, bem como a justificativa para conduzir uma investigação no assunto debatido e suas principais contribuições.

1.1 Contextualização

Desde a década de 1970, com a criação do modelo relacional por Edgar Frank Codd, a estrutura de armazenamento adotada por muitos desenvolvedores de sistemas da área de tecnologia da informação tem se baseado no conceito de entidade e relação proposto por Codd. A maioria dos sistemas gerenciadores de banco de dados que possui aceitação no mercado fazem uso desse modelo, por exemplo o MySQL, Oracle e Microsoft SQL Server. Porém, os requisitos para o desenvolvimento de ferramentas de software modernas têm mudado significativamente, especialmente com o aumento das aplicações Web [NASHOLM (2012)]. Este segmento de aplicações exige requisitos com alta escalabilidade e vazão, onde sistemas que utilizam um armazenamento com esquema relacional não conseguem atender satisfatoriamente. Em resposta a isso, novas abordagens de armazenamentos de dados utilizando o termo de NoSQL tornaram-se popular [SILVA (2016)].

O termo NoSQL é constantemente interpretado como "*Not Only SQL*", cujo SQL refere-se a linguagem de manipulação de dados dos gerenciadores de armazenamento de dados relacionais (RDBMS - Relational Database Management System) - Structure Query Language [NASHOLM (2012)]. O grande propósito das abordagens NoSQL é oferecer alternativas onde os esquemas relacionais não apresentam um bom desempenho. Esse termo abrange diferentes tipos de sistemas. Em geral, banco de dados NoSQL usam modelo de dados não-relacionais, com poucas definições de esquema, são executados em clusters e aplicados a alguns bancos de dados recentes como o Cassandra, o Mongo, o Neo4J e o Riak [FOWLER; SADALAGE (2013)].

Muitas empresas coletam e armazenam milhares de gigabytes de dados por dia, no qual a análise desses dados torna-se uma vantagem competitiva no mercado. Por isso, há uma grande necessidade de uma nova arquitetura para o gerenciamento de suporte à decisão que possa alcançar melhor escalabilidade e eficiência [LIU; THOMSEN; PEDERSEN (2013)]. Para auxiliar no processo de gerenciamento de suporte à decisão uma das formas mais utilizadas é a criação de um ambiente data warehousing, que é responsável por providenciar informações estratégicas e esquematizadas a respeito do negócio [CHAUDHURI; DAYAL (1997)].

Segundo a definição de KIMBALL; ROSS (2002), data warehouse (DW) é uma coleção de dados para o processo de gerenciamento de suporte à decisão orientado a assunto, integrado, variante no tempo e não volátil. Os dados de diferentes fontes de sistemas são processados em um data warehouse central através da Extração, Transformação e Carga (ETL) de maneira periódica. Os processos de ETL consistem em um conjunto de técnicas e ferramentas para transformar dados de múltiplas fontes de dados para fins de análise de negócio [SILVA (2016)]. Ferramentas de ETL são sistemas de software responsáveis por extrair dados de diversas fontes, transformar e customizar os dados e inseri-los no data warehouse. Comumente, esses processos são executados periodicamente, onde a otimização do seu tempo de execução torna-se importante [VASSILIADIS; SIMITSIS; SKIADOPOU-LOS (2005), SILVA (2016)].

O projeto de ETL consome cerca de 70% dos recursos de implantação de um DW, pois

desenvolver esse projeto é crítico e custoso, tendo em vista que gerar dados incorretos pode acarretar em más decisões. Porém, por algum tempo pouca importância foi dada ao processo de ETL pelo fato de ser visto somente como uma atividade de suporte aos projetos de DW. Apenas a partir do ano 2000, a comunidade acadêmica passou a dar mais importância ao tema[(SILVA (2012)].

Tradicionalmente, o DW é implementado em uma base de dados relacional, onde o dado é armazenado nas tabelas fato e tabelas dimensões, na qual forma um esquema em estrela [KIMBALL; ROSS (2002)]. Por isso, é comum que as ferramentas de ETL utilizadas no mercado atualmente demonstrem mais importância aos esquemas relacionais. Para oferecer suporte aos sistemas que necessitem utilizar um esquema não relacional de BDs NoSQL em DW, a proposta desse trabalho é especificar um framework programável, flexível e integrado para modelagem e execução de processos ETL em BDs NoSQL.

1.2 Motivação

A integração de dados e os processos de ETL são procedimentos cruciais para a criação de data warehouses e sistemas BI (*business intelligence*). Porém, os sistemas para ETL e integração de dados são tradicionalmente desenvolvidos para dados estruturados em modelos relacionais que representam apenas uma pequena parte dos dados mantidos por muitas empresas [DARMONT et al. (2005), Russom 2007, Pedersen 2009]. Dessa forma, existe uma demanda crescente para integrar os dados não estruturados e semi estruturados em um repositório unificado. Devido a complexidade desses dados, novos desafios estão surgindo quando lidamos com dados heterogêneos e distribuídos no ambiente de integração [Salem, 2012].

Além disso, muitas empresas encontram dificuldades para lidar com as ferramentas ETL disponíveis no mercado. Aprender a lidar com essas ferramentas pode ser muito custoso em termos financeiros e de tempo, e por isso, acabam optando desenvolver os seus processos por meio de uma linguagem de programação de propósito geral [Awad et al., 2011; Muñoz et al., 2009].

Portanto, este trabalho propõe um framework programável para desenvolvimento de sistemas de ETL que possibilita a integração de dados estruturados, não estruturados e semi estruturados armazenados em bases relacionais ou NoSQL. O framework possui um ambiente integrado para a importação e mapeamento dos dados, além da modelagem e customização dos processos de ETL. Os processos de importação e mapeamento do framework integram dados estruturados, não estruturados e semi estruturados. Esses processos possibilitam a leitura e manipulação de dados de bases NoSQL, e também o armazenamento desses dados em bases deste tipo, oferecendo uma alternativa não relacional para a construção de DWs.

Uma alternativa para organizar e manipular grandes volumes de dados sem utilizar um modelo relacional e ainda processá-los e armazená-los de maneira distribuída é fazer o uso de BDs NoSQL [CARVALHO SCABORA (2016)]. Com isso, surge a necessidade de se promover

meios para o uso desses BDs em DWs.

As pesquisas presentes na literatura sobre extração de dados em BDs NoSQL mostram que não há uma ferramenta que seja integrada para o uso de BDs NoSQL, as ferramentas existentes no mercado apenas oferecem a possibilidade para alguns SGBDs NoSQL, ficando a cargo da equipe de implantação do projeto de DW todo o trabalho de modelagem e programação ao se utilizar BDs NoSQL [SILVA (2016), CHEVALIER et al. (2015), LIU; THOMSEN; PEDERSEN (2013)].

SILVA (2012) aponta em sua pesquisa que muitas empresas evitam ferramentas de ETL disponíveis no mercado, e adotam o desenvolvimento dos processos a partir de uma linguagem de programação de propósito geral, pelo fato dessas ferramentas terem uma longa curva de aprendizagem e grande complexidade no seu uso.

O aumento do uso de banco de dados com esquemas não relacionais baseados no paradigma NoSQL e a falta de uma ferramenta programável, flexível e integrada, independente de plataforma que dê suporte à extração, transformação e carga em data warehouses para esses esquemas é a grande motivação deste trabalho.

Dessa forma, encontrar uma solução que seja programável, flexível e integrada para extração, transformação e carga dos dados em BDs NoSQL é a proposta deste trabalho.

1.3 Objetivos

O objetivo principal desta pesquisa é especificar um framework programável, flexível e integrado para modelagem e execução de processos ETL de banco de dados estruturados, não estruturados e semi estruturados sob os modelos relacionais e NoSQL. Os objetivos específicos são detalhados a seguir.

1.3.1 Objetivo Específico

Este trabalho de dissertação tem como um dos objetivos específicos apresentar os componentes do framework ETL4NoSQL, bem como suas funcionalidades. Outro objetivo deste trabalho é realizar um estudo experimental de software a fim de caracterizar as principais funcionalidades das ferramentas de ETL na manipulação de dados estruturados, semi estruturados e não estruturados. O estudo experimental poderá comparar o framework proposto, suas vantagens e desvantagens, em relação às ferramentas de ETL encontradas na literatura.

1.4 Contribuições

Uma das contribuições deste trabalho é fornecer um framework programável, flexível e integrado que auxilia na modelagem e execução dos processos de ETL em bases de dados estruturadas, semi estruturadas e não estruturadas, denominado ETL4NoSQL. Assim, é possível extrair, integrar e carregar dados que estão armazenados em diversas estruturas como é o caso

dos bancos de dados NoSQL, ou até mesmo, repositórios de dados textuais e banco de dados relacionais em um único repositório. O ETL4NoSQL é um recurso valioso, principalmente para os desenvolvedores responsáveis pela fase de ETL, onde muitos encontram dificuldades para lidar com as ferramentas ETL disponíveis no mercado.

Outra contribuição desta pesquisa é apresentar, por meio de um estudo experimental, as principais características, de acordo com algumas ferramentas de ETL presentes na literatura, bem como possíveis melhorias, vantagens e desvantagens, em suas funcionalidades.

1.5 Organização do Trabalho

Este trabalho está organizado de acordo com a seguinte estrutura:

- **Fundamentação Teórica:** apresenta uma revisão da literatura dos principais assuntos abordados neste trabalho. Serão tratados temas a respeito de ETL, banco de dados NoSQL, Frameworks e estudo experimental de software.
- **Trabalhos Correlatos:** descreve os trabalhos correlatos encontrados na literatura a respeito de ferramentas de ETL. Este capítulo visa um comparativo das diversas ferramentas existentes com a proposta deste trabalho.
- **O Framework ETL4NoSQL:** descreve os requisitos, arquitetura e componentes do framework exposto neste trabalho.
- **Estudo Experimental de Software:** expõe o roteiro da experimentação de software para ferramentas de ETL. Define o objetivo, planejamento, operação e resultado do estudo.
- **Considerações Finais:** expressa as limitações e ameaças à validade do trabalho, considerações finais e sugere de trabalhos futuros.