Pós-Graduação em Ciência da Computação

GIOVANI FELIPE JAHN

UMA PROPOSTA DE ARQUITETURA PARA TRATAMENTO DE DADOS NÃO ESTRUTURADOS NO ÂMBITO DOS INSTITUTOS FEDERAIS DE EDUCAÇÃO

UNIVERSIDADE FEDERAL DE PERNAMBUCO

posgraduacao@cin.ufpe.br

www.cin.ufpe.br/~posgraduacao

RECIFE

2017

GIOVANI FELIPE JAHN

UMA PROPOSTA DE ARQUITETURA PARA TRATAMENTO DE DADOS NÃO ESTRUTURADOS NO ÂMBITO DOS INSTITUTOS FEDERAIS DE EDUCAÇÃO

Dissertação apresentada como requisito parcial à obtenção do grau de Mestre em Ciência da Computação, do Programa de Pós-graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco.

ORIENTADOR: Vinicius Cardoso Garcia

RECIFE

2017

Catalogação na fonte
Bibliotecária Monick Raquel Silvestre da S. Portes, CRB4-1217

J25p Jahn, Giovani Felipe
Uma proposta de arquitetura para tratamento de dados não estruturados no âmbito dos institutos federais de educação / Giovani Felipe Jahn. – 2017.
130 f.: il., fig., tab.

Orientador: Vinícius Cardoso Garcia.
Dissertação (Mestrado) – Universidade Federal de Pernambuco. CIn, Ciência da Computação, Recife, 2017.
Inclui referências e apêndices.

1. Engenharia de software. 2. Arquitetura de software. I. Garcia, Vinícius Cardoso (orientador). II. Título.

005.1 CDD (23. ed.) UFPE- MEI 2017-257

Giovani Felipe Jahn

Uma proposta de arquitetura para tratamento de dados não estruturados no âmbito dos Institutos Federais de Educação

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Mestre Profissional em 14 de junho de 2017.

Aprovado em: 14 / 07 / 2017.

BANCA EXAMINADORA

_____

Prof. Fernando da Fonseca de Souza

Centro de Informática / UFPE

_____

Prof. Edson Luiz Padoin

UNIJUI

_____

Prof. Vinícius Cardoso Garcia

Centro de Informática / UFPE

(Orientador)

A meus filhos.

Dedico.

AGRADECIMENTOS

A lista merecedora de meus créditos se mensurada individualmente, talvez preencha todo este espaço. Contudo cabem alguns destaques, que os farei sucintamente, não menos importantes que os demais eventualmente, nominalmente, não citados.

Agradeço em primeiríssimo lugar aos meus filhos, Cassiano, Daniel e Rafael, meus companheiros e componentes de uma família bem peculiar há mais de dez anos. Agraço pelo entendimento deles quando da minha ausência e também da minha impaciência por vezes, fomentada pela preocupação em dar mais atenção aos estudos do que a eles.

Agradeço a “Re” por simplesmente me ajudar em tudo.

Agradeço ao “Vinicius” por simplesmente me orientar em tudo, direta e objetivamente, e com aquele peculiar vocabulário.

Agradeço aos meus colegas de mestrado, todos amigos, parceiros e que ajudaram a fazer destes anos um ótimo tempo.

Agradeço a meu pai, minha mãe e demais familiares pelo apoio em todos os aspectos.

Por fim, mas muito significativamente, agradeço aos “Originais” (Anderson Bitoca, Eric Google, Jadson Caveirão, Janderson Peixe-Boi, Marcelo Empresário) pelo convívio, pelos estudos, pelas aventuras e incontáveis histórias, mas principalmente, por me darem a oportunidade de formar uma nova família de irmãos.

RESUMO

Uma das tendências para resolver os diversos problemas e desafios gerados pelo contexto do *Big Data* é o movimento denominado *NoSQL (Not only SQL*), o qual promove diversas soluções inovadoras de armazenamento e processamento de grande volume de dados. Os trabalhos disponíveis na literatura relacionados a *NoSQL* explicam, além do seu surgimento, sistemas disponíveis para a manipulação de dados que necessitam de um poder de processamento eficiente, escalável e amplo. O que também impulsiona a criação de sistemas de tratamento de dados *NoSQL* é a sua inferência a dados complexos, semiestruturados ou não estruturados, presentes hoje em redes sociais, sensores, *logs* de Internet, entre outros. Em face dos desafios sobre a manipulação e processamento de dados neste contexto, um novo conjunto de plataformas de ferramentas voltadas para *Big Data* tem sido proposto. Muitas delas na forma *open source* ou de licenças livres revelando-se excelentes veículos para o desenvolvimento de soluções para tratamento de dados desta natureza. Este trabalho, proposto no Programa de Mestrado Profissional em Ciência da Computação, na linha de pesquisa Redes de Computadores, objetiva apresentar uma arquitetura de referência para tratamento de dados não estruturados, inerentes à realidade dos institutos federais de educação, para que possam prover análise sobre dados oriundos de redes sociais. Inicialmente, a revisão bibliográfica expõe os conceitos, linguagens e ferramentas das principais tecnologias a respeito de *NoSQL*. Produtos como Hadoop, Hive, HBase e outros denotam a grande quantidade de soluções *NoSQL* disponíveis no mercado para uma escolha futura na implementação de aplicações e são consideradas neste trabalho. A seguir, um levantamento de dados institucionais mostra que o tratamento de dados não estruturados ainda é considerado um ineditismo para estas instituições. Foi utilizada uma abordagem metodológica teórico-conceitual, agregando-se paradigmas do método DSR (*Design Science Research*) para dar à pesquisa um conhecimento sólido e potencialmente relevante quando da elaboração de uma proposta de arquitetura de referência para tratamento de dados não estruturados no âmbito dos institutos federais de educação. A realização de um adequado enquadramento conceitual e tecnológico sobre as ferramentas *open source* fomentaram esta proposta, a qual por sua vez, passa por avaliação e crivo de especialistas.

Palavras-chave: *Big Data*. *NoSQL*. Ferramentas *open source.*

ABSTRACT

One of the trends to solve the various problems and challenges generated by the *Big Data* context is the movement called *NoSQL* (Not only SQL), which promotes several innovative solutions for storing and processing large volumes of data. The work available in the literature related to *NoSQL* explains its emergence in the context of large amount of data generated and consequently, systems available for the manipulation of this data that need an efficient, scalable and broad processing power. What also drives the creation of *NoSQL* data processing systems is their inference to complex, semi-structured or unstructured data, present in social networks, sensors, Internet logs, among others. Given the challenges of manipulating and processing immense data in this context, a new set of tool platforms geared to *Big Data* have been proposed. Many of them in the open source form, proving to be excellent vehicles for the development of data processing solutions of this nature. This work, proposed in the Master Program in Computer Science, in the research line Computer Networks, aims to present a reference architecture for the treatment of unstructured data, inherent to the reality of the federal institutes of education, so that they can provide analysis on these data come from social networks, as well as content from various sources on the Internet. Initially, the bibliographic review exposes the concepts, languages and tools of the main technologies regarding *NoSQL*. Products such as Hadoop, Hive, HBase and others denote the large number of *NoSQL* solutions available in the market for a future choice in the implementation of applications and are considered in this work. Next, a survey of institutional data shows that the treatment of unstructured data is still considered an unpublished data for these institutions. A conceptual theoretical methodological approach was used, adding paradigms of the DSR (Design Science Research) method to give the research a solid and potentially relevant knowledge when elaborating a proposal of reference architecture for the treatment of unstructured data within such institutes. The realization of an adequate conceptual and technological framework on the open source tools fomented this proposal, which in turn, passes through evaluation and selection of specialists.

Keywords: *Big Data*. *NoSQL*. Open source tools

LISTA DE FIGURAS

LISTA DE QUADROS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

ACID - Atomicity, Consistency, Isolation, Durability

API- Application Programming Interface

ATAM – Software Architecture Analysis Method

AVA – Ambientes Virtuais de Aprendizagem

BASE – Basic Availability, Soft State, Eventual Consistency

BI – Business Inteligence

BSON – Binary JSON

CAP – Consistency, Availability and Partition tolerance

DGA- Dados Governamentais Abertos

DHT – Distributed Hash Table

DMBOK – Data Management Body of Knowledge

DSR – Design Science Research

E-Gov- Eletronic Government

ETL – Extraction, Transformation and Load

EXT_n – Extended (1, 2 ou 3)

G2B – Government to Business

G2C – Government to Citizen

G2G – Government to Government

GD – Governança de Dados

GPL – General Public License

HDFS – Hadoop File System

HTML- Hiper Text Marcup Language

HTTP – Hiper Text Transfer Protocol

IBM - International Business Machines

IFFAR – Instituto Federal de Educação, Ciência e Tecnologia Farroupilha

IoT - Internet of Things

JSON- Javascript Object Notation

NoSQL – Not Only SQL

NTFS – Network File System

OLAP - Online Analytical Processing

PDI – Plano de Desenvolvimento Institucional

PPC – Projeto Pedagógico do Curso

REDIS – Remote DIctionary Server

REST – Representational State Transfer

RFID – Radio-Frequency IDentification

SAAM – Software Architecture Analysis Method

SGBD- Sistema Gerenciador de Banco de Dados

SPARQL – SPARQL Protocol and RDF Query Language

SQL - Structured Query Language

TCO - Total Cost of Ownership

TCP – Transmission Control Protocol

TI – tecnologia da Informação

TIC - Tecnologia da Informação e Comunicação

URI - Uniform Resource Identifier

W3C – World Wide Web Consortium

Weka - Waikato Environment for Knowledge Analysis

XML- Extensible Markup Language

SUMÁRIO

1 INTRODUÇÃO

Neste capítulo são apresentadas as motivações, enquadramento e justificativas para a realização deste trabalho. Também são descritos os objetivos, geral e específicos, a abordagem metodológica e as etapas desta, que norteiam a pesquisa, seguidos da estratégia de revisão da literatura. Por fim a estrutura do documento é apresentada.

1.1 Problema

O crescimento exponencial de dados não estruturados oriundos de fontes diversas como redes sociais, conteúdo *web*, redes locais, dispositivos de IoT, entre outros, é iminente. Estes dados são úteis às organizações dos mais diversos segmentos, inclusive aos Institutos Federais de Educação. Baseado nesta premissa, uma questão primordial surge: como tratar da melhor maneira esses dados, visando tirar proveito para o que se propõe um Instituto Federal de Educação?

1.2 Motivação e Enquadramento

Os dados produzidos pelas redes ligadas à Internet chegam a seus usuários sob os mais diversos formatos e em volumes cada vez maiores, humanamente imensuráveis. O tratamento destes dados, com finalidades analíticas, bem como a complexidade deste tipo de trabalho só aumenta e onde guardá-los é uma preocupação igualmente frequente. A melhor forma de se ter acesso a estes dados, sem que eles signifiquem apenas um amontoado usurpador de espaço e tempo, são os projetos envolvendo *Big Data* no ambiente *Cloud*. O uso de nuvens híbridas pelas Instituições Federais de Educação, Ciência e Tecnologia já se faz realidade há algum tempo. Torna-se então conveniente agregar a estas, as novas tendências de tratamento de dados não estruturados, oriundos dos diversos dispositivos interligados às redes, Internet, IoT, redes sociais, entre outras.

De forma geral, estes dados podem ser minerados[1] visando fornecer *insights* e conhecimentos valiosos para as instituições de ensino, o que significa que utilizar *Big Data* vai ao encontro da missão destas instituições. A mineração de dados agrega veracidade na visualização de desempenho de alunos, por exemplo, fomentando recomendações e ações por parte docente. Igualmente, métodos analíticos podem sugerir desde como tratar certos

[1] Mineração de dados é o processo de descoberta de informações acionáveis em grandes conjuntos de dados. Fonte Microsoft.com.

comportamentos de alunos até a indicação de cursos ou atividades educacionais direcionadas a conteúdos específicos. Ainda, segundo Wassanr (2015), instituições de ensino estão usando dados analíticos para melhorar os serviços prestados, em face do crescimento de alunos e cursos. O uso de plataformas de *Big Data* e modelos de programação como *MapReduce*, podem acelerar a análise de dados educacionais. E, bem mais do que apenas capturar dados, cabe à instituição dentro desta realidade, transformá-los em informação útil para a sua gestão educacional, bem como adaptar suas ações.

Dentro do fenômeno *Big Data*, as fontes *Open Data* (dados abertos), especialmente no que diz respeito a dados governamentais, incorporam benefícios como transparência, controle social e participação. Tais benefícios, segundo Giffinger et al. (2007) são inerentes a *Smart Governance* (transparência), *Smart People* (participação na vida pública) e *Smart Economy* (empreendedorismo). Segundo esta premissa, pode-se afirmar que *open data* provê incentivo para os quesitos em questão, nas instituições de ensino. Salienta-se ainda que, por se tratar de órgão público que faz parte efetiva de um governo, o uso de *open data* vai ao encontro da Lei de Acesso à Informação (Lei nº 12.527, de 18 de novembro de 2011). O uso de dados abertos governamentais pelo governo melhora a eficiência organizacional dos setores públicos, a criação de novos produtos e serviços, além de benefícios sociais (MANYKA et al., 2013).

Recomenda-se a utilização de Tecnologias de Informação e Comunicação (TIC) como forma de qualificar serviços prestados nas áreas da educação. Inerente a isso é importante conhecer quais as tecnologias de bancos de dados disponíveis, como funcionam e quais os benefícios que elas poderiam trazer num âmbito educacional, acerca do tratamento de dados, para que se façam futuramente escolhas sobre tais quando da implementação de aplicativos a serem utilizados pelas instituições federais de ensino. Para isso, uma análise exploratória sobre estas tecnologias é cabível e muito útil.

Premiando os argumentos supracitados, associados a um contexto envolvendo dados não estruturados, este estudo visa também mostrar que determinadas informações, se coletadas e analisadas, podem servir de suporte para os processos de aprendizagem e formação do indivíduo, além de incentivar o fomento à pesquisa.

1.3 Objetivos

1.3.1 Geral

Propor uma arquitetura de referência para tratamentos de dados não estruturados.

1.3.2 Específicos

a) Apresentar o estado da arte nos temas *Big Data*, *Analytics*, *NoSQL*;
b) Caracterizar o papel das bases de dados *NoSQL*;
c) Comparar as formas de tratamento de dados;
d) Identificar as técnicas e ferramentas empregadas para soluções de tratamento de dados não estruturados;
e) Propor uma arquitetura de tratamento de dados não estruturados (*NoSQL*), identificada com a realidade do Instituto Federal de Educação Ciência e Tecnologia Farroupilha; e
f) Construir conhecimento no escopo dos Institutos Federais de Educação a partir do estudo de dados não estruturados.

Desta forma, espera-se inicialmente caracterizar *Big Data* e Dados *NoSQL*, bem como o papel deste último em relação ao primeiro. Num segundo momento, efetuar um estudo e análise de ferramentas e métodos que possam compor uma arquitetura, direcionada à realidade dos Institutos Federais de Educação, em particular ao IFFar (Instituto Federal de Educação Ciência e Tecnologia Farroupilha), para o tratamento de dados não estruturados.

1.4 Abordagem Metodológica

A abordagem metodológica norteadora desta dissertação é do tipo teórico-conceitual fundamentada em revisão da literatura (exploratória). Objetiva, segundo Vergara (2005), explorar uma área que apresenta escassez de conhecimento sistematizado. Considerando que a área que engloba as tecnologias *NoSQL* não produziu até o momento vasto ou completo acervo sobre o tema, torna-se pertinente explorar mais intrinsicamente o termo em si, bibliografias, produtos ofertados no mercado, dentre outros, com o intuito de fazer uma análise quanto às vantagens que se pode obter em contraste com a realidade do IFFar. O modelo DSR é usado para corroborar com a eficácia da pesquisa.

A adoção de metodologia significa aqui, escolher um determinado percurso transcorrido em etapas, sob determinadas regras, mas não obstruindo a criatividade, auxiliando o autor a pensar criticamente, ter disciplina, desenvolver e escrever este trabalho, com crivos de padrões metodológicos e acadêmicos. Identifica como se processam as operações mentais no processo de pesquisa científica e o sistematiza em passos distintos, mostrando os procedimentos adotados em cada um deles.

1.4.1 Caracterização da pesquisa

Pesquisa é um procedimento racional e sistemático que tem como objetivo buscar respostas a problemas previamente propostos (GIL, 2002). Ainda com base neste autor, considera-se este trabalho como pesquisa tecnológica, visto que preocupa-se em produzir um constructo de caráter funcional. Quanto aos procedimentos adotados, a pesquisa intitula-se bibliográfica-exploratória (GIL, 2002), pois utiliza-se de fontes bibliográficas inúmeras, artigos, periódicos, livros, teses e outras pesquisas.

Cabe ressaltar, segundo Cervo e Bervian (2002), que a pesquisa é uma atividade voltada para solução de problemas teóricos ou práticos utilizando processos científicos quando se tem um problema e não há informações para solucioná-lo.

Não obstante a isso, Daft e Lewin (1990) justificam a necessidade de modernização quanto aos métodos de pesquisa, bem como sua adequação ao problema investigado. Sugere às organizações, a adoção de métodos de *Design Science*, pois seriam estes veículos preponderantes para atender as exigências rigorosas da pesquisa. Romme (2003), sob o mesmo prisma, enfatiza que os estudos que envolvem organizações carecem de maior afinco e afirma que tais estudos devem incluir Design Science nas suas pesquisas, por ser esta a forma mais concisa na produção de conhecimento. Contudo, para que se dê operacionalização a estes conceitos, garantindo o devido rigor à pesquisa, se faz necessária a adoção de um método compatível e inerente ao proposto. Este método de pesquisa é denominado *Design Science Research* (DSR).

Unir uma abordagem metodológica teórico-conceitual fundamentada em ampla revisão da literatura, juntamente com Design Science e, por conseguinte, norteada pela DSR, corrobora para o sucesso da pesquisa. Baseada neste paradigma, esta é a proposta adotada para se construir um artefato do tipo constructo, permitindo um trabalho inerente ao ambiente organizacional do IFFar.

1.4.2 Pesquisa bibliográfico-exploratória

A presente pesquisa é definida quanto aos fins como exploratória, o que a caracteriza por possuir uma maior familiaridade do pesquisador com o tema, podendo ser alicerçada em hipóteses ou intuições. Sugere um levantamento bibliográfico de maior magnitude, no qual as citações e exemplos visam facilitar o entendimento da matéria. Na opinião de Babbie (1986), a pesquisa exploratória, onde se permite controlar efeitos que desvirtuem a percepção do pesquisador, possibilita que a realidade seja percebida tal como ela é, e não como o pesquisador pensa que seja.

Quanto aos meios, por sua vez, a pesquisa condiciona-se a ser bibliográfica, pois para a fundamentação teórica e metodológica se fez necessário investigar sobre os seguintes assuntos: *NoSQL*, *Big Data*, arquiteturas de dados, ferramentas para tratamento de dados, entre outros assuntos e termos correlatos. Averiguações assim caracterizadas são abundantemente utilizadas nas pesquisas exploratórias e contam muito com a intuição do pesquisador.

Para tornar fáticas as teorias transcritas nos parágrafos anteriores, foram usados mecanismos de buscas acadêmicas como *Web Science*[2], IEEE[3], *Scopus*[4], *Google Scholar*[5], *Digital Library*[6], além de outros veículos que trouxessem fomento e elucidação a assuntos relevantes ao tema desta pesquisa.

1.4.2.1 Estratégia de Revisão da Literatura

A revisão da literatura objetiva a aquisição de conhecimento para absorver o contexto atualizado em que se inserem os dados não estruturados correlatos aos objetivos propostos e relacionados com o problema anteriormente apresentado.

A revisão da literatura se coloca como um dos principais pilares deste projeto de dissertação. Além de perceber o estado atual em que se encontram os termos inerentes à pesquisa, é necessário uma abordagem conceitual sobre as ferramentas e arquiteturas disponíveis. A revisão foi inicialmente delineada para ser realizada de forma gradativa, iniciando pelos termos mais genéricos e posteriormente mais intrínsecos e específicos, como arquiteturas relacionadas a *NoSQL*, ferramentas *open source* e de livre licença para *Big Data* e

[2] webofknowledge.com
[3] https://www.ieee.org
[4] https://www.scopus.com
[5] scholar.google.com.br
[6] http://dl.acm.org

tratamento de dados não estruturados, além de inserções pertinentes às ramificações de assuntos dentro de cenários e circunstâncias diversas.

Sendo assim, se inserem os mesmos em repositórios acadêmicos e científicos, tais como: "*Google Scholar", "Scopus", "Web Of Science", "IEEE"* e *"ACM Digital Library*", levando em consideração o ano de publicação dos artigos, a relevância dos mesmos de acordo com o seu número de citações, além de associações entre os autores na área. Os conceitos-chave utilizados na pesquisa foram os constantes na Tabela 1, dando primazia ao período 2013-2017, não se abdicando obviamente de produções de conteúdo julgado significativo ao contexto da pesquisa, que estivesse fora desta cronologia.

No processo de revisão bibliográfica, as buscas transcorreram em mais de uma plataforma, sendo que inúmeros documentos inicialmente foram encontrados, contendo os termos pressupostos. Posteriormente, uma breve visualização se fez necessária sobre a maioria destes, para então filtrar o conteúdo e a consequente aquisição do material.

Tabela 1: Relação de termos pesquisados em repositórios e quantidade retornada em número de artigos/dissertações/teses

	Google Scholar	Scopus	Web Of Science	IEEE	ACM Digital Library
BigData	12.300	387	235	281	158
NoSQL	15.900	1.331	810	602	182
NoSQL Database	7.920	1.201	623	538	80
NoSQL Clusters	71	122	83	74	1
NoSQL Engine	57	135	56	65	2
NoSQL Architecture	65	272	159	147	1
NoSQL Education	1	24	9	9	0
Hadoop	+ 28.300	4.964	3.183	2.961	510
MapReduce	+ 16.600	5.105	3.575	3.123	657
Arquiteturas *BigData*	0	0	0	282	0
BigData Analytics	322	78	53	61	11

Fonte: Elaborada pelo autor (2017)

A partir desse ponto, implementaram-se refinamentos na pesquisa, para então serem mais objetivamente retomadas as práticas de leitura de resumos, de artigos na íntegra, de material citado nestes e de outras fontes.

Dentre os documentos lidos, aqueles que mais significativamente se associavam à proposta inicial deste trabalho de pesquisa, ou que produziam subsídios relevantes, estão mesurados no apêndice A.

1.4.3 DSR

Ao se estudar os fenômenos artificiais, que diferentemente dos naturais (descrevem as interações e comportamentos da natureza), são os criados pela humanidade, produz-se um intento de satisfação de necessidades, desejos e objetivos (VAISHNAVI e KUECHLER, 2009). Tais autores defendem que o *Design Science* é um método apropriado para o estudo de fenômenos desta origem, pois possui técnicas para a construção de artefatos que satisfaçam seus pressupostos, utilizando design, análise, reflexão e abstração.

Vakkari (1994) complementa, afirmando que a ciência da informação é sinônimo de ciência aplicada, e, por conseguinte, uma "ciência de projeto", logo, *Design Science* é uma ciência desenvolvedora de artefatos tecnológicos para atender às necessidades práticas de organizações, instituições ou indivíduos.

A missão principal da *Design Science* é desenvolver conhecimento para a concepção e desenvolvimento de artefatos (VAN AKEN, 2004), sendo assim é correto afirmar que a *Design Science Research* (DSR) é uma meta-teoria que propicia ao pesquisador criar artefatos por meio de processos, processos estes que por suas vez criam conhecimento, justificando assim o caráter cientifico da pesquisa. Por conseguinte ao se produzir um artefato, está se gerando conhecimento científico para melhorar ou aperfeiçoar os processos inerentes a uma organização ou instituição.

Van Aken (2004) nos traz a citação que expressa com muito oportunismo o cerne do que se propõe uma metodologia embasada em DSR: "A *Design Science* não se preocupa com a ação em si mesma, mas com o conhecimento que pode ser utilizado para projetar as soluções" (VAN AKEN, 2004, p. 228). De acordo com Hevner et al. (2004), a DSR procura identificar e tratar problemas reais de um mundo real, propondo soluções práticas e apropriadas para a solução destes, por meio da construção e aplicação de um artefato. Estes artefatos podem ser constructos, modelos, métodos ou instâncias de um sistema, construídos e avaliados sob o prisma do problema que se propõe a tratar.

Lacerda (2013) é um pouco mais enfático quando sugere que a DSR é um método cerne para propostas de pesquisa de feição tecnológica, resultando efetivamente em produção cientifica rigorosa, quando bem conduzida.

O processo de DSR inicia quando o pesquisador toma consciência do “problema”, que é condição inicial para iniciar a pesquisa. Identificar o problema e seu contexto é primordial para a compreensão e delimitação do ambiente estudado, e partindo desse contexto, motivar e justificar a importância da pesquisa a ser realizada. Esta consciência pode vir de múltiplas fontes, e resulta em uma proposta formal de um novo esforço de pesquisa (VAISHNAVI e KUECHLER, 2009).

A fase seguinte é de “sugestão”, onde o pesquisador, a partir do problema, define os objetivos a serem atingidos por sua solução. Elencados os objetivos e, apoiando-se no estado da arte de sua área, inicia-se esta criativa etapa na qual são sugeridas proposições que podem englobar uma reconfiguração de elementos já existentes ou novos (VAISHNAVI e KUECHLER, 2009).

A partir das sugestões, inicia-se a etapa do “desenvolvimento”, que trata efetivamente da construção do artefato para a solução do problema. Esse desenvolvimento deve obedecer às métricas do DSR no tocante à construção de um artefato.

Para DSR, um artefato é a organização dos componentes de um ambiente interno, como meio para obtenção de êxito em atingir objetivos em um determinado ambiente externo (SIMON, 1996). Uma vez definidos os artefatos, pode-se tipificá-los. Artefatos podem ser definidos como: Constructos, Modelos, Métodos e Instanciações (MARCH; SMITH, 1995).

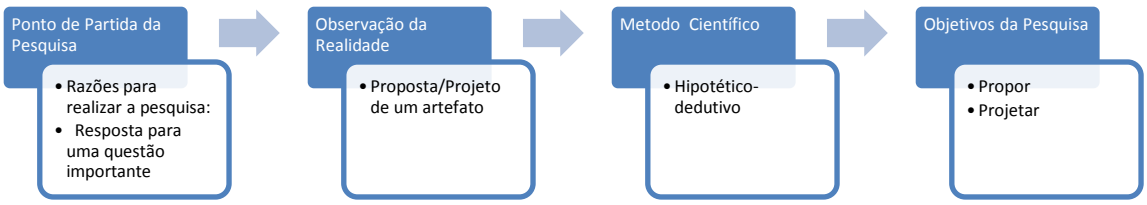
É importante frisar que, quando se trata de desenvolvimento de uma solução, não se está invariavelmente, ou exclusivamente, direcionando-se ao desenvolvimento de produtos. A DSR atende perfeitamente tal propósito, contudo, possui objetivos de maior amplitude: gerar conhecimento que seja aplicável e útil para a solução de problemas, melhorias em algo já existente ou ainda sim, a criação de novas soluções (artefatos) (VENABLE, 2006).

A “avaliação”, etapa final da DSR, é definida como o processo rigoroso de verificação do comportamento do artefato no ambiente para o qual foi projetado, em relação às soluções que se propôs alcançar.

Em vista destes argumentos, é possível resumir que a DSR permite sanar um problema de pesquisa, sem desconsiderar o rigor científico necessário quando se produz novo conhecimento, tampouco se desconsidera a aplicabilidade desta solução em uma situação real. Desta feita, estes conceitos vêm ao encontro à proposta desta pesquisa sobre tratamento de dados não estruturados em âmbito institucional especifico.

No que tange efetivamente ao trabalho aqui proposto, sugere-se chegar a um artefato, tecnicamente embasado, para solucionar o problema do tratamento de dados não estruturados no âmbito institucional. As etapas da pesquisa estão sistematizadas na Figura 1.

Figura 1: Demonstrativo sintético da pesquisa, segundo o DSR



Fonte: Elaborada pelo autor (2017)

1.4.4 Fases da abordagem metodológica

Exposta a definição do problema, dar-se-á início a pesquisa. Conforme Marconi e Lakatos (1990), na área científica, um problema se refere a qualquer circunstância não resolvida e que é instrumento de discussão, em qualquer área de conhecimento, despertando assim, o interesse do pesquisador. O objetivo do estudo proposto é então a solução para este problema. Contudo, é primordial verificar inicialmente se o problema sugerido se enquadra como "científico". Ainda, segundo Marconi e Lakatos (1990), um problema, de ordem prática ou intelectual, é de natureza científica quando envolve variáveis que podem ser testadas, observadas, manipuladas por meio de metodologias. Esta proposta de pesquisa, e dissertação subsequente, se enquadra nos quesitos impostos.

Após a definição ou identificação do problema e com objetivos delineados, Dresch (2013) recomenda cientificar-se das repercussões deste para a organização.

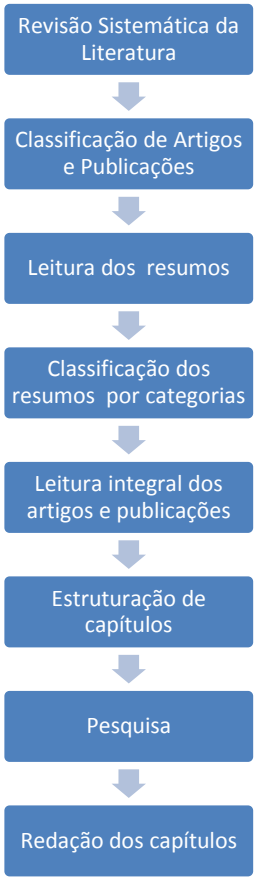
Os resultados que esta pesquisa busca alcançar estão então delineados pela proposição dos objetivos geral e específicos, e apontam para um cunho mais direcionado ao exploratório e explicativo, sem, contudo, desproverem-se do descritivo.

Um plano de trabalho é então elaborado para nortear com eficácia as etapas e tarefas inerentes à pesquisa. A partir disso, realiza-se uma revisão sistemática da literatura, buscando estabelecer um quadro contendo as soluções disponíveis até o momento.

1.4.4.1 Construção de um Plano de Trabalho

De acordo com Marconi e Lakatos (2010), o plano de trabalho se constitui no encadeamento das etapas lógicas, adotadas pelo pesquisador, com vistas a lograr êxito no seu objetivo, bem como gerar um conhecimento verdadeiramente reconhecido. Fazer uma ampla e atenciosa revisão exploratória da literatura visa objetivamente encontrar técnicas que podem ser utilizadas para projetar a arquitetura de referencia para tratamento de dados não estruturados. De igual seriedade, conceber conceitos acerca das áreas, ferramentas e aplicações sobre dados não estruturados de grande volume, se torna essencial e primária atitude para sucesso da referida proposta. Sinalizam-se os conceitos antepostos pela Figura 2:

Figura 2: Plano de trabalho

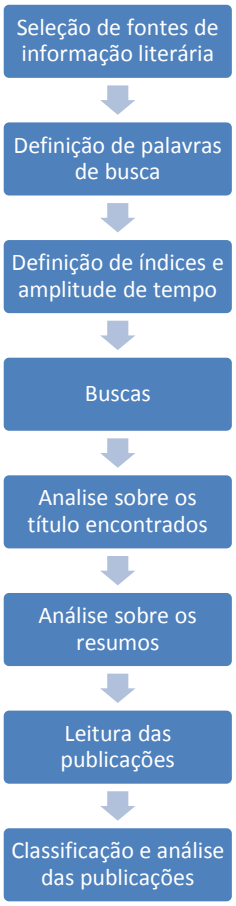


Fonte: Elaborada pelo autor (2017)

1.4.4.2 Coleta de Dados

Para dar seguimento à pesquisa fez-se necessária a coleta de dados, subdividida em duas esferas. Uma em repositórios literários, com a finalidade de prover recursos técnicos e teóricos, sintetizada pela Figura 3:

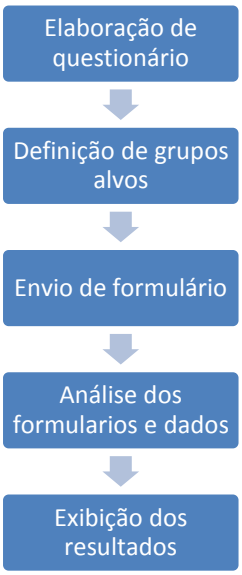
Figura 3: Síntese da coleta de dados em repositórios literários



Fonte: Elaborada pelo autor (2017)

Outra coleta de dados foi realizada na instituição de ensino, através de pesquisa direcionada aos servidores, que serviu para trazer à tona a realidade do IFFar, no que se refere ao nível de conhecimento por parte dos analistas e técnicos de TI sobre o tema "dados não estruturados". Foi aplicada no tipo *survey*, para ancorar os objetivos da pesquisa. O diagrama da Figura 4 sintetiza o trabalho realizado quanto a estas inferências.

Figura 4: Síntese da coleta de dados em instituições federais de ensino – *survey*



Fonte: Elaborada pelo autor (2017)

1.4.4.3. Proposição da Solução

Com as técnicas encontradas na revisão, fomentadas com os conceitos e formas sobre tratamento de dados e também sobre a situação institucional, é possível propor uma arquitetura de referência para atender os requisitos e atributos de qualidade, quanto ao tratamento de dados não estruturados.

1.4.4.4 Avaliação

A avaliação de uma proposição de pesquisa é certamente uma das mais marcantes características da atividade científica, pode ser, e na maioria das vezes é, aplicada em todas as etapas. Contudo, concerne que ao final do processo, uma avaliação mais criteriosa deve ser imposta sobre o produto final, aquele que busca responder ao problema e que contempla, ou não, os objetivos propostos pelo pesquisador. Tal avaliação abordará questões gerais, metodológicas e éticas e estará fundamentada em métodos concisos e confiáveis.

Para que se atinja, ou que se tente atingir, na plenitude estes paradigmas avaliativos, este trabalho submete, a cargo de especialistas, a arquitetura proposta para que, ao crivo dos mesmos, através de um método descritivo, seja avaliada, aceita, refutada ou analisada e comentada.

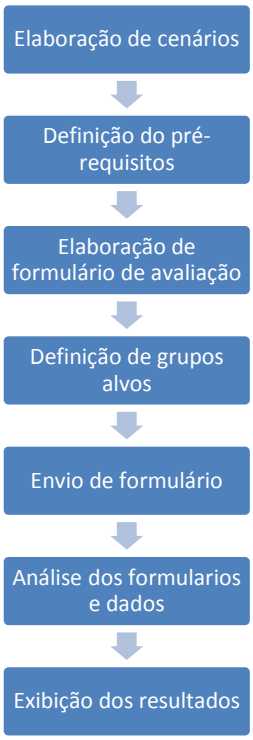
Desta forma e conforme Hevner et al. (2004), a forma de avaliação utilizada é descritiva, pois utiliza as pesquisas relevantes sobre o assunto para corroborar com os argumentos que ressaltam a utilidade do constructo. Tal avaliação busca demonstrar a utilidade do artefato desenvolvido.

Ainda, frisa-se que esta proposta é uma arquitetura de referência apenas. Arquiteturas de referência são genéricas e visam atender considerações dos *stakeholders*[7] acerca de um domínio específico. Nesse prospecto, não há um método específico para a avaliação desse tipo de arquitetura, ao contrário do que poderíamos encontrar em avalições de arquiteturas de software específicas.

1.4.4.5 Coleta de Dados Acerca da Avaliação da Solução Proposta

Para que um nível elevado de incontestabilidade da avalição fosse atingido, alguns passos precaveram o método, que segue exposto na Figura 5:

Figura 5: Etapas do processo de avaliação



Fonte: Elaborada pelo autor (2017)

Assim sendo, a abordagem metodológica, durante este trabalho sugere:

[7] Termo usado em diversas áreas como, como gestão de projetos ou de software, entre tantas, referente às partes interessadas.

a) Definir a questão de investigação: “De que forma uma arquitetura para tratamento de dados não estruturados pode ser útil aos institutos federais de ensino e, por conseguinte, ao IFFar?”;

b) Propor objetivos a serem alcançados;

c) Construir um plano de trabalho, descrevendo etapas;

d) Realizar o trabalho e a escrita da dissertação, com composição tecnológica; e

e) Obedecer à metodologia da pesquisa através de revisão da literatura norteada por DSR, como suporte a este trabalho.

1.5 Estrutura do Documento

O presente documento está dividido da seguinte forma:

O capítulo inicial fornece ao leitor uma breve introdução acerca das razões e motivos que impulsionaram o surgimento desta pesquisa, bem como a estratégia de pesquisa utilizada para a revisão de literatura e os objetivos propostos. O capítulo descreve a abordagem metodológica da pesquisa, além de uma contextualização sobre o método DSR. Ao final do capítulo, é apresentada a estrutura do documento.

No capítulo dois, destinado ao referencial teórico e enquadramento conceitual é elaborada a revisão da literatura. Em primeiro lugar é abordado o aspecto do *Big Data* dissecando suas adjacências para que o leitor tome posse do conhecimento sobre este tema, cujas definições também se enquadram no contexto de dados não estruturados, haja vista a estreita relação entre os dois termos. Faz-se também um apanhado técnico sobre bases de dados *NoSQL* e ferramentas para tratamento de dados.

No capítulo três mostram-se os detalhamentos da pesquisa para se chegar a uma proposta de arquitetura para tratamento de dados não estruturados no ambiente dos institutos federais de educação e a apresentação de uma proposta de arquitetura de referência, baseada em soluções *open source* e de livre licença.

No capítulo quatro é apresentada a arquitetura de referência que o autor considera pertinente aos objetivos da pesquisa, voltada à utilização pelos Institutos Federais de Educação.

O quinto capítulo visa à avaliação da arquitetura proposta, apresentando o desenvolvimento da arquitetura através de cenários propostos a mesma, que tiram partido de

dados não estruturados, armazenamento e processamento distribuído, destacando tecnologias, como por exemplo, Apache Hadoop[8], HBase[9], HDFS[10], Kafka[11], Tableau[12], entre outros.

No derradeiro capítulo há conclusões e futuras proposições. Nele são apresentadas as conclusões do trabalho e são levantadas possibilidades para trabalhos futuros.

[8] http://hadoop.apache.org/
[9] https://hbase.apache.org
[10] https://hadoop.apache.org/docs/r1.2.1/hdfs_design.html
[11] https://kafka.apache.org/
[12] https://www.tableau.com/

2 REFERENCIAL TEÓRICO

Com o propósito de apresentar alguns conceitos fundamentais acerca dos temas abordados por esta pesquisa, este capítulo propõe introduzir os principais conceitos que a norteiam. Dentre eles, impreterivelmente estão inclusas as características de *Big Data*, *Analytics*, *NoSQL*, entre outros, para que, através de uma conceituação mais detalhada, se façam abordagens mais concisas sobre tratamento de dados, com uma oportuna ênfase no que tange a dados não estruturados, além dos aspectos existentes em torno de modelos e ferramentas.

Cabe inicialmente recordar que o modelo relacional de banco de dados[13] esteve (e ainda está) presente em muitos sistemas de tratamento de dados. Contudo, o crescimento do intenso volume de dados nas organizações, associado à ansiedade por sorver informações úteis ao mercado ou à gestão, tem proporcionado o surgimento de modelos alternativos ao relacional, impulsionados também por uma necessidade de escalabilidade destes sistemas. Estes modelos são atualmente intitulados *NoSQL*, e sua amplitude vem crescendo e ganhando força, tanto no ambiente acadêmico quanto no meio comercial.

Segundo Navathe e Elmasri (2010), diferentemente do modelo relacional, os sistemas *NoSQL* não estão "amarrados" às estruturas tabulares, apresentando maior flexibilidade e não dependência de esquemas.

Com o aumento expressivo no número de usuários ligados às mídias sociais e à Internet como um todo, dispositivos de IoT e outros tantos segmentos geradores de dados diversos, é primordial que haja um meio capaz de armazenar e proceder a uma análise confiável e ágil destes dados.

É importante frisar que conceitos de adoção de tecnologias *NoSQL* estão presentes em domínios de *Big Data* (VIEIRA *et. al.*, 2012). Falar sobre dados não estruturados recai em aprimorar paradigmas de *Big Data*. Assim sendo, cabe iniciar este referencial com uma dissecação deste tema, identificar ferramentas inerentes ao seu tratamento e em seguida agregar, a este contexto, especificidades do *NoSQL*.

[13] http://www.dsc.ufcg.edu.br/~baptista/cursos/BDadosI/Capitulo22.pdf

2.1 *Big Data*

A premissa do crescimento acelerado em escala quase imensurável de dados gerados pela sociedade informatizada é o que concebe o termo *Big Data*. Apoiado nas mudanças tecnológicas, o aspecto outrora fictício de uma escala na casa do zettabytes é hoje um adjetivo para uso até mesmo doméstico. A critério de exemplo, White (2012) traz à tona a afirmação anterior mensurando e citando algumas empresas como o Facebook[14], que armazena mais de dez bilhões de fotos, compreendendo mais de um petabyte de informação armazenada.

Além da acelerada escala em que volumes cada vez maiores de dados são criados, caracteriza-se *Big Data* com a heterogeneidade e quantidade de dispositivos e de usuários conectados e ao fato de o uso das tecnologias de informação e comunicação serem usadas para as mais cotidianas rotinas realizadas por um usuário (MAYER-SCHÖNBERGER; CUKIER, 2013). Não apenas pessoas são responsáveis por produzirem dados, equipamentos eletrônicos das mais diversas categorias e características, também se tornaram grandes geradores de registros de dados.

Big Data surge enraizado no propósito de extrair dados que excedem a capacidade de processamento convencional dos sistemas de banco de dados, transformando-os em informações úteis, inerentes e pertinentes, no mesmo momento em que são processados. Tais dados, além de excessivamente grandes, movem-se rapidamente pelos meios e não se encaixam nas restritas arquiteturas de banco de dados convencionais. Por estes motivos, obriga-se o uso de ferramentas especificas capazes de tratar grandes volumes, de forma que toda e qualquer informação nestes meios possa ser encontrada, analisada e utilizada oportunamente e em tempo hábil (FAN e BIFET, 2012).

> *Big Data is a new term used to identify datasets that we can not manage with current methodologies or Data Mining software tools due to their large size and complexity*. (FAN e BIFET, 2012).

Sinteticamente, Kim, Trimi e Ji-Hyoung (2014) definem o termo *Big Data* como uma quantidade enorme de dados digitais, coletados de inúmeras fontes. Marth e Scharkow (2013) completam a afirmação acima acrescentando que estes conjuntos de dados são grandes demais para serem manipulados por infraestruturas de armazenamento e processamento regulares.

[14] Rede social fundada em 2004 por Mark Zuckerberg, Eduardo Saverin, Andrew McCollum, Dustin Moskovitz e Chris Hughes, estudantes da Universidade Harvard.

Aprofundando um pouco mais o termo *Big Data* dentro dos segmentos literários, verificam-se, e obriga-se a agregar a este, outras importantes características além do já mensurado volume expressivo. A proveniência de fontes diversas destes dados condiz com variedade, bem como o tempo real em que se tratam estes dados para gerar informação subjuga velocidade (MANYIKA et al., 2011; MCAFEE & BRYNJOLFSSON, 2012).

A variedade de fontes de dados está diretamente relacionada aos mais diversos sistemas, aplicativos, ferramentas, equipamentos, conexões e usuários envolvidos na captação dos dados, sendo que, atualmente em sua maioria, estão em forma não estruturada. São, sim, imagens, *streamings* de áudio e vídeo, textos, logs de Internet, dentre tantos outros (DAVENPORT, 2014).

De igual relevância aos aspectos anteriores, há a característica da velocidade. Em resumo, os dados gerados ficam disponíveis em tempo real, simultaneamente e imediatamente à sua criação, análise e processamento, subsidiando tomadas de decisões instantâneas (MCAFFE; BRYNJOLFSSON, 2012).

Está iminente aqui o aspecto do volume exagerado. São gerados petabytes de dados a cada dia, estimando-se ainda que este número dobre em poucos meses. Também é variedade, visto que, conforme já mencionado, estes dados vêm de sistemas estruturados e não estruturados (a imensa maioria), gerados por e-mails, mídias sociais (Facebook, Twitter, YouTube e outros), documentos eletrônicos, apresentações, mensagens instantâneas, sensores, etiquetas RFID[15] (*Radio-Frequency IDentification*), câmeras de vídeo, e muito mais. Velocidade, porque muitas vezes precisamos agir praticamente em tempo real sobre este imenso volume de dados. Além destes três "Vs", é perfeitamente plausível agregar ao conceito de *Big Data* outros dois: veracidade e valor (DEMCHENKO et al., 2013). Segundo o autor, a confiabilidade dos dados precisa ser comprovada quanto à sua origem e autenticidade, além do que estes dados devem perfazer um sentido. Quanto ao valor, Demchenko (2013) implica em afirmar que este significa trabalhar com dados que de fato tenham sentido e que sejam fontes de valor agregado para a tomada de decisões, principio este básico para o retorno dos investimentos em projetos de *Big Data*.

2.1.1 Paradigmas do *Big Data*

Afirma-se que as tecnologias, ainda atuais para a maioria, mas efetivamente já de outrora, no que se refere a tratamento de dados não são mais adequadas. Que fatores

[15] Identificação por radiofrequência

sustentam tal prerrogativa? Cabe aqui contextualizar os ditos paradigmas do *Big Data*, tendo como finalidade principal sanar esta incógnita pressuposta e ressaltar as propriedades que então devem possuir os novos sistemas para tratamento de dados.

Já foi mencionado por diversas vezes neste compêndio, que as enormes quantidades de dados na Internet criaram a necessidade de uma tecnologia que permita às empresas, instituições, órgãos, extrairem valor a partir desses dados aleatórios e de diversos formatos, de forma eficiente, ágil e que resulte em informação que gere conhecimento ou rentabilidade. *Big Data* é esta tecnologia.

Sobre os dados de grande volume é necessário que se consiga tratar toda a abrangente variedade que lhes caracteriza, assim definida quanto aos tipos de dados: Estruturados - são os dados que detêm formatos bem definidos, como os extraídos de planilhas ou bancos de dados relacionais no formato SQL[16]; Semiestruturados – Semelhantes aos dados estruturados, mas não obedientes na totalidade quanto à forma. Nesta linha estão os registros de linguagens baseadas em HTML e XML; Não estruturados ou *NoSQL* - não possuem um formato específico, são os dados coletados na sua forma original, como um texto, um vídeo, um fragmento de email, um log de sistema ou ainda uma mera foto.

Quando Edgar F. Codd [17], em 1969, propôs o modelo relacional para bancos de dados[18], a demanda recaía unicamente sobre dados de estrutura definida, de origem interna nas empresas que se propunham a usá-lo. Tal modelo, obviamente, não fora concebido para tratar dados não estruturados, pelo simples fato de que isso era inimaginável naquele tempo, tampouco para volumes na ordem dos terabytes/petabytes/exabytes/zettabytes, vistos na contemporaneidade. Sendo assim, o modelo criado por Codd era plenamente satisfatório, além de simples, para o contexto cronológico em que se encontrava.

Não custa repetir, a necessidade de novos modelos para tratar dados na escala de volume, variedade e velocidade do *Big Data* e o surgimento de sistemas de banco de dados *NoSQL* se materializam.

O aspecto que envolve análise dos dados captados também é fator preponderante quando mencionamos *Big Data*. Na concepção de Davenport (2007), componentes de

[16] *Structured Query Language*

[17] Edgar (Ted) Codd, matemático da IBM, conhecido por criar o modelo "relacional" para representar dados.

[18] IBM, 2016 https://www-03.ibm.com/ibm/history/exhibits/builders/builders_codd.html

Analytics[19] são importantíssimos no que se refere a converter dados em valor de negócio, coexistindo com os tradicionais sistemas de *Business Inteligence*[20] hoje existentes.

Ainda, de acordo com Marz e Warren (2015), a complexidade, escalabilidade, robustez e facilidade são as principais propriedades que adjetivam um sistema *Big Data.* Cabe aqui uma revisão formal sobre tais propriedades:

a) Tolerância a falhas: o próprio conceito de sistemas distribuídos[21], por si só, torna esta caraterística um tanto difícil de ser atingida. Contudo, é plausível e necessário fazer com que os sistemas se portem robustamente, apesar da aleatoriedade dos nós que compõem os *clusters*, da semântica complexa, da duplicidade dos dados, da concorrência e das demais situções inerentes a sitemas distribuidos. Este contexto sugere que, para sistemas de *Big Data* possuirem robustez e tolerância a falhas, o caminho mais curto é evitar complexidades desnecessárias, concentrando-se na razão de existência deste sistema, a que ele se propõe. Isso dificulta a incidência de erros humanos, além de dar robustez ao mesmo.

b) Escalabilidade: segundo Cattell (2011), um sistema de *Big Data* escalável é o que detém a capacidade de tratar dados em volumes crescentes, sem perder desempenho e sem deixar de satisfazer as demandas crescentes em mesma proporção. Em resumo, é a capacidade de manter o desempenho do sistema quando do aumento de dados ou carga, e quando recursos de *hardware* e *software* são solicitados ao sistema. A Arquitetura Lambda[22], de característica escalável horizontalmente, torna-se um exemplo ideal para explicar esta propriedade, pois adiciona mais máquinas ao conjunto de *clusters* quando da necessidade de maior desempenho.

c) Baixa latência: requisitos de latência para leitura e atualização variam muito de um sistema para outro. Porém, para sistemas *Big Data*, é primordial conseguir trabalhar com baixa latência, no que se refere principalmente a atualizações, sem que haja comprometimento da robustez.

d) Extensibilidade: Mudanças, programadas ou não, em sistemas de *Big Data*, não podem comprometer nehuma das qualidades que este sistema traz. Um sistema com extensibilidade permite ao desenvolvedor adicionar um novo recurso ou uma alteração,

[19] Uma abordagem centrada em dados que combina a ciência de análise preditiva com capacidades avançadas de inteligência de negócios.
[20] BI - uma técnica para auxiliar o gestor no planejamento estratégico
[21] "Um sistema distribuído é um conjunto de computadores independentes entre si que se apresenta a seus usuários como um sistema único e coerente" – Tanenbaum e Van Steen (2008)
[22] http://lambda-architecture.net/

sem comprometer possiveis migrações de dados ou ainda dispender custos elevados para tal. Tornar um sistema extensível significa facilitar os processos de migrações, em pequena ou larga escala, de forma rápida e fácil.

e) Depuração ágil e manutenção mínima: mesmo quando um sistema já está adjetivado como estável e robusto, não significa a improbabilidade de erros futuros ou ao longo de sua existência. Seria considerado utópico prever a perfeição durante todo o processo existencial de um sistema. Em sistemas *Big Data*, servem os preditos citados e, desta forma, estes devem oferecer condições favoráveis e transparência de informações quanto da necessidade de depuração quando algo sair errado. Rastreabilidade sobre valores dos dados também é primordial, para se chegar às possíveis ações equivocadas que o sistema venha a cometer. Em nível de igual importância está a manutenção do referido sistema. Manutenções são ações previamente estipuladas pelo desenvolvedor de um sistema. Em *Big Data* isso implica em antecipar quando adicionar máquinas aos *clusters*, manter os processos funcionando e depurar tudo o que tenha dado errado. Um fator minimizador de manutenção é escolher componentes que tenham a menor complexidade de implementação possível. Quanto mais complexo um sistema, mais provável algo vai dar errado. A complexidade de um sistema *Big Data* é combatida pela utilização de algoritmos simples com componentes simples.

2.1.2 Classificação de *Big Data* Segundo suas Categorias

Pode-se ainda caracterizar um pouco mais o *Big Data*, através de uma sintética classificação em categorias, com o propósito de trazer ao leitor um pouco mais de detalhes sobre como os dados são tratados e os processos envolvidos, além de formalizar alguns padrões para tal:

a) Quanto ao tipo de análise dos dados - análise em tempo real; análise posterior (por lotes de dados);

b) Quanto ao método de processamento - compreende a técnica aplicada no processamento dos dados. Pode ser *ad hoc*, analítica, preditiva ou relatórios. Também é pertinente combinar mais de uma destas técnicas, elegidas de acordo com a finalidade e propósito do sistema;

c) Quanto aos tipos de dado – significa a classificação dos tipos de dados a serem processados de acordo com as diretivas de classificação do sistema como, por exemplo, transacionais, históricos, implícitos, explícitos e outros;

d) Quanto ao formato dos dados - quanto à forma os dados podem ser estruturados (bancos de dados relacionais), semiestruturados (arquivos XML) ou não estruturados (imagens, áudio, vídeo, texto, logs);

e) Quanto ao tamanho e frequência dos dados - compreende o volume estimado e com que frequência estes dados passam pelo processamento. Determinar previamente estes dois quesitos ajuda a definir como será feito o armazenamento, quanto ao mecanismo utilizado, formato e as ferramentas necessárias para que o processamento seja o mais otimizado possível. As fontes que determinam frequência e tamanho, são: sob demanda, dados de mídias sociais, por exemplo; *feed* em tempo real, que são os dados transacionais (climáticos), por exemplo; série temporal, que são dados com base em intervalos de tempo;

e) Quanto à fonte dos dados – compreendem a origem da fonte de dados, se foram gerados por um dispositivo qualquer, um computador conectado gerando logs, páginas web, redes sociais, entre outros. Identificar todas as fontes de dados ajuda a determinar o escopo de uma perspectiva quanto aos objetivos do sistema de *Big Data* (BI, negócios, análise comportamental);

f) Quanto aos consumidores de dados – enumera-se uma lista de todos os possíveis consumidores dos dados processados:

- processos de negócios;
- usuários corporativos;
- aplicativos corporativos
- pessoas individuais em várias funções de negócios;
- parte dos fluxos do processo;
- outros repositórios de dados ou aplicativos corporativos.

g) Quanto ao *Hardware* – toda solução *Big Data* terá como base um tipo de equipamento físico (*Hardware*), que será responsável por “rodar” o sistema. A escolha do *Hardware* é diretamente proporcional ao desempenho da solução de *Big Data*.

2.1.3 Importância do *Big Data*

O crescimento maciço do uso de *smartphones*, sensores e outros dispositivos produtores de dados, associados ao uso da computação em nuvem e às melhorias da Web, como rotina para indivíduos e instituições, exemplifica claramente o avanço tecnológico vivenciado em dias atuais. Tais avanços contribuem para um maior volume, velocidade e variedade de dados produzidos. Isso é um fato gerador de inúmeros exemplos da importância, tanto das fontes de *Big Data* como do conceito em si, inerente a sistemas computacionais para tomada de decisões, conforme os transcritos que se seguem.

O serviço Google Trends[23] corrobora em provar que o termo *Big Data* desperta importância tal qual seu efetivo benefício. Nos últimos anos cresce exponencialmente a procura acerca do assunto, bem como surgem ferramentas e dispositivos computacionais de hardware e software para tratar e aplicar, velozmente, dados de grande volume e variedade.

Mais do que apenas um amontoado gigantesco de dados, o *Big Data* representa o fomento para a produção de informações de natureza estruturada e não estruturada, dando-lhe uma característica de gerador de dados sobre dados, uma metainformação acerca de indivíduos e usuários, com influência direta e indireta sobres estes. Inerente a todas as profissões, o *Big Data* influencia na relação de uso dos poderes que o conhecimento produz. A utilização desta tecnologia associada a algoritmos de *Machine Learning*[24] permite evidenciar cenários de tendências de usuários, pressupor decisões administrativas, inferir decisões governamentais em determinadas áreas, ações inimagináveis sem o uso de uma abordagem baseada nestes paradigmas. Com isso fica lúcido o conceito que o termo *Big Data* não está relacionado apenas à coleta e ao armazenamento em grandes proporções, mas obviamente implica também em processamento destes dados para agregar valor ao contexto em que for aplicado. Estas prerrogativas sugerem aplicabilidade preferencialmente em sistemas de *Business Intelligence*, contudo, significam equidistância de aplicabilidade em ambientes ou instituições educacionais.

Incontestável é a relação entre dado, informação e conhecimento, no sentido de que os dois primeiros agregam o terceiro (CHOO, 2006). A coleta de informações adequadas tem como principal finalidade ser útil, do ponto de vista estratégico e funcional, às organizações. Ainda, o conhecimento tem sido cada vez mais reconhecido como o bem mais valioso nas

[23] https://trends.google.com.br/

[24] Tipo de Inteligência Artificial que facilita a capacidade de um computador em aprender e essencialmente ensinar-se a evoluir à medida que ele é exposto a novos dados e em constante mudança. Fonte: http://computerworld.com.br

instituições. Desta forma, para que as organizações sejam bem sucedidas elas devem estar capacitadas a capturar, integrar, criar e utilizar o conhecimento de maneira disciplinada, sistêmica e estratégica (SAMBAMURTHY; SUBRAMANI, 2005).

É igualmente correto afirmar que diariamente produz-se uma expressiva quantidade de dados sobre cada usuário, espelhando nossas atividades comerciais, políticas, sociais, pessoais e privadas. Deixamos rastros digitais a respeito do que gostamos de comer, lugares a visitar, amores declarados, desgostos e generalidades.

O uso de redes sociais, compras, cadastros diversos, transações, dentre tantas outras atividades realizadas por intermédio da Internet, geram uma quantidade significativamente grande de dados sobre cada sujeito. Empresas, órgãos governamentais, entidades diversas, estão aptas cada vez mais a se apoderarem de informações referentes ao indivíduo, como localização, preferências pessoais, afinidades e até mesmo fase sentimental, de maneira precisa e perspicaz. Neste sentido, torna-se óbvio o potencial controle que o *Big Data* pode aferir no que tange ao controle estratégico, social, politico e de negócios.

Pode-se considerar que atualmente vive-se a era da explosão de dados, onde a quantidade de dados criados e armazenados possui volume quase imensurável (ZIKOPOULOS et al., 2012). APESAR DE SER UM TERMO RELATIVAMENTE NOVO, O *BIG DATA* ESTÁ PRESENTE E CRESCE EM TODOS OS SETORES DA sociedade. Empresas como Google ou Netflix fazem uso de um fluxo constante de dados complexos para milhões de requisições em tempo real. Outro exemplo são as empresas de negócios, que usufruem da imensidão de dados dispostos por usuários em redes sociais, considerados úteis para discernir sobre o gosto e as preferências destes em relação a um produto ou serviço oferecido, no claro objetivo de estreitar o relacionamento entre fornecedor e consumidor.

Cabe frisar que os dados precisam ter qualidade, para que se tornem significativos de uso, produzindo benefícios às instituições. Segundo Salvador et al. (2006), um dado possui qualidade se está dentro dos seguintes preceitos:

> - Preciso: é a medida de quão correto, quão livre de erros, quão próximo está este dado do fato verdadeiro. É medida fundamental da qualidade de dados; se um dado não é correto, as outras dimensões são menos importantes. Para ser correto, um valor deve ser certo e deve ser representado de uma forma consistente e sem ambiguidade.
> - Disponível em tempo: os dados estão suficientemente atualizados para as tarefas que os necessitam;
> - Relevante: importante para o tomador de decisões em um contexto; é útil e aplicável à tarefa em questão;
> - Governança de Dados e Qualidade de Dados segundo DMBOK
> - Completo: deve conter todos os fatos importantes, na amplitude e profundidade adequadas às suas necessidades;

- Simples/Compreendido: evitando a chamada “sobrecarga de informação”;
- Confiável: depende da fonte ou método de coleta. (SALVADOR et al.,. 2006).

Surge iminentemente neste cenário, um grande desafio para com o assunto *Big Data*: como filtrar, entender e usar os dados gerados? Sobrepor este desafio requer primeiramente obter acesso a esse extenso conjunto de dados nada homogêneos, gerados e propagados em escala imensurável. De igual forma outro desafio, também oportunidade, é o fato dos gestores possuírem a capacidade então de usarem este grande volume de variados dados não estruturados oriundos de fontes inúmeras, como smartphones, redes sociais, blogs, dentre tantos outros (DI MARTINO et al., 2014) em processos decisórios, visto que as instituições estão lidando com uma expansão de dados que são incompatíveis com os tradicionais métodos de gerenciamento e análise, o que leva à necessidade de se pensar em novas formas para tal, a fim de gerar informações pertinentes e oportunas (DAVENPORT, 2012).

No que tange aos órgãos governamentais - instituições de ensino caracterizam-se como tal – cabe ressaltar aqui alguns preâmbulos sobre dados abertos (*open data*). É pertinente no sentido de que a instituição de ensino pode valer-se de consultas de naturezas diversas para compor informações úteis e disponíveis à sociedade.

É comum encontramos associação entre *open data* e *open government*, porém, além de conceituar ambos, é necessário frisar que não são sinônimos. Enquanto que a referência ao termo dados abertos (*open data*) é relevante para dados que podem ser usados livremente, reutilizados e redistribuídos por qualquer pessoa, desde que se sujeitem a atribuição da fonte e as regras de compartilhamento (OPEN DATA HANDBOOK, 2014) e não restrinja o seu uso e compartilhamento por terceiros, indiferentemente de qual seja a finalidade. O termo governo aberto (*open government*), traduz-se num conjunto de iniciativas que buscam dar ao Estado maior transparência e responsabilidade, além de, segundo Barros et al. (2010), propor padronização técnica, semântica e organizacional na produção e divulgação de dados e informações públicas, que indiretamente implicaria maior responsabilidade ao Estado quanto a gestão de recursos e melhoria na prestação dos serviços públicos.

Adjetivar um dado de “aberto” implica em algumas exigências: permitir a redistribuição, a reutilização, a disponibilidade – preferencialmente na Internet - e o acesso modificável. Não pode também haver discriminação acerca de qualquer que seja a área de atuação, bem como grupos ou indivíduos - um titulo de somente educativo, por exemplo, não mais caracteriza o dado como aberto (OPEN DATA HANDBOOK, 2014).

Em consonância com os pressupostos mencionados sobre dados abertos, pode-se afirmar que, para a existência de uma característica de governo aberto, este deve obrigatoriamente fazer uso dos conceitos do *open data*. Disponibilizar dados governamentais, sob o título de abertos, tem por objetivo quebrar barreiras existentes entre o cidadão usuário de informações do serviço público e o Estado. Por conveniência, ao cidadão deve ser assegurado o acesso, bem como o entendimento, aos dados públicos, segundo seus interesses.

Desta forma, a ideia primordial associada ao *Open Government Data*[25] é de disponibilizar e permitir compartilhamento, à maior possível quantidade de dados oficiais de um "governo".

A partir destas premissas surge o "jargão" e-gov (Governo Eletrônico), entendido como uma das principais formas de modernização do Estado. Um novo paradigma de inclusão digital, que tem como sujeito principal o cidadão, ao passo que propicia ao Estado oferecer serviços públicos com maior qualidade, reduzir custos, simplificar processos, aprimorar a gestão e ser transparente (CHAHIN et al., 2004).

Segundo Matheus, Vaz e Ribeiro (2014), o Brasil vem evoluindo neste sentido de enquadra-se como governo aberto. Inicialmente utilizando apenas de páginas estáticas com relatórios financeiros, passando pela adoção dos portais de transparência e por fim, *datasets*[26] em formato aberto (Dados Governamentais Abertos - DGA). Porém, mesmo apresentando evolução neste aspecto, é importante salientar que apenas disponibilizar informações em portais governamentais não assegura a necessária transparência. Segundo Paiva e Revoredo (2016), em relacionado artigo, o grande volume de dados e a não padronização são as causas que ainda inviabilizam esta prática e o efetivo acompanhamento sistemático desses dados. Relatam ainda:

> A solução para esse tipo de problema se dá através da aplicação de técnicas de tratamentos de dados que permitam a estruturação da informação de forma mais clara e elucidativa para a população. Para isso, é necessário que se desenvolvam ferramentas capazes de processar esse grande volume de dados e que permitam uma visualização consolidada dessas informações [...] (PAIVA, REVOREDO, 2016)

Por outro lado, só existe sentido em se dispor dados abertos, se o cidadão manifesta iniciativa, ou pelo menos curiosidade, em reutilizá-los conforme seu interesse. Contudo, ainda encontram-se empecilhos para tal. A maioria das informações do poder público ainda se apresenta em formas proprietárias ou incompatíveis à totalidade de indivíduos (pessoas com

[25] https://opengovernmentdata.org/
[26] Conjunto de dados tabulados

deficiência), dispositivos e equipamentos. Desta feita, é necessário impor que instituições propensas a publicar dados governamentais, o façam segundo alguns princípios:

a) selecionar que dados serão disponibilizados;

b) identificar e responsabilizar os controladores dos dados;

c) apresentar estes dados sob forma reutilizável e amigável; e

d) publicar e divulgar os dados.

Além destes, segundo OPEN GOV DATA (2007), os dados devem caracterizar-se por:

a) completos - Todos os dados públicos estão disponíveis, sem limitações de privacidade, segurança ou controle de acesso;

b) primários - Apresentados tais como os coletados na fonte, sem modificações;

c) atuais - Disponibilizados tão logo quando criados, para preservação do seu valor;

d) acessíveis - Maior alcance possível de usuários e para o maior conjunto possível de finalidades;

e) compatíveis ao *Hardware* - Os dados devem possibilitar processamento automatizado;

f) não discriminatórios - Disponíveis para todos, nativamente;

g) não proprietários – Disponíveis sob qualquer formato, sem exclusividade de sistema, aplicativo ou software; e

h) livres de licenças - Os dados não estão sujeitos a nenhuma restrição de direito autoral, patente, propriedade intelectual ou segredo industrial.

Ainda correlacionado ao tema recém citado, é pertinente dissertar sobre a governança de dados (GD). Entende-se por governança de dados o conjunto de práticas, processos, padrões, tecnologias e politicas de acessibilidade, disponibilidade, qualidade, consistência e segurança sobre os dados de uma organização (PANIAN, 2010), usada como uma ferramenta de criação de diretrizes de controle destes dados.

As instituições, das mais diversas áreas, podem dar caminhos distintos quanto à forma de atuar sobre a governança de dados. Existem por exemplo metodologias diferentes para mesmos paradigmas de GD, como os programas de Governança de Dados da Oracle[27], da IBM[28], e do DMBOK - *Data Management Body of Knowledge* (DAMA, 2015). Contudo, independente do foco principal que a instituição direcionar, alguns objetivos são considerados comuns em qualquer tipo de política de tratamento de dados pela governança. São eles, segundo Fernandes e Abreu (2012):

[27] http://www.devmedia.com.br/governanca-de-dados-implementando-a-gestao-e-governanca-de-dados/30915
[28] https://www-01.ibm.com/software/br/data/info/information-governance/dm.html

a) permitir uma melhor tomada de decisões;

b) proteger as necessidades dos *stakeholders*;

c) institucionalizar uma gerência comum no tratamento de problemas de dados;

d) construir padrões, processos e metodologias que possam ser disseminadas pela organização;

e) reduzir custos e aumentar a eficácia através da coordenação de esforços conjuntos; e

f) garantir a transparência dos processos.

Cabe por fim ressaltar, que o IFFar, uma instituição pública com governabilidade própria, não se abstém das responsabilidades e inferências, direta ou indiretamente, impostas pelos preceitos da governança de dados. Cabe sim a ela, quando da proposta de governar através do tratamento e/ou análise de dados, dar estreita observância ao imposto por estes conceitos, anteriormente citados.

2.1.3 *Big Data* Analytics

Traduz-se *Big Data Analytics*, didaticamente, como uma análise sobre grandes quantidades de dados. E de fato é isso que vem a ser. Consiste em fazer uso de ferramentas de análise avançada sobre um volume significativamente grande de dados, com o objetivo de prover informação inteligente, através do uso de software de altíssimo desempenho.

Agrega um conjunto de ações que envolvem a busca de dados, consequente processamento destes e com o objetivo de gerar *insights* para tomadas de decisões estratégicas, por meio da análise profunda. Tudo isso realizado no menor espaço de tempo previsível, conforme conceitua Russom (2016).

> [..] Organizações de usuários estão implementando formas específicas de análise, o que às vezes é chamado de análise avançada. Esta é uma coleção de técnicas relacionadas e tipos de ferramentas, geralmente incluindo análise preditiva, mineração de dados, análise estatística complexa e SQL. Podemos também estender a lista para cobrir a visualização de dados, inteligência artificial, processamento de linguagem natural e capacidades de banco de dados que suportam análises (como *MapReduce*, análise de banco de dados, bancos de dados em memória, armazenamento de dados em colunas). Em vez de "análise avançada", um termo melhor seria "análise de descoberta", porque é isso que os usuários estão tentando realizar. Em outras palavras, com grande análise de dados, o usuário é tipicamente um analista de negócios que está tentando descobrir novos fatos de negócios que ninguém na empresa sabia antes. Para isso, o analista precisa de grandes volumes de dados detalhados. (RUSSOM, 2016)

O objetivo maior do *Big Data Analytics* pode ser traduzido em: otimizar processos de trabalho, adquirir *insights* valiosos acerca das tendências de mercado, comportamento dos consumidores, identificar o perfil de um determinado público e previsões.

Para ser bem sucedido quanto aos propósitos informados, um *software* de análise sobre *Big Data*, busca subsídios nas seguintes fontes:

a) conteúdo de mídias sociais;

b) estatísticas de sensores conectados a IoT;

c) pesquisas de satisfação;

d) textos de e-mails;

e) arquivos de log;

f) dados de ferramentas de *Business Intelligence*;

g) relatórios empresariais; e

h) indicadores econômicos.

Há bilhões de usuários ativos nas redes sociais que compartilham milhões de informações, fotos, vídeos, *tweets*[29] em um dia e que conectados, pressupõem dados para a coleta, armazenamento e análise de ferramentas de *Big Data Analytics*, por meio de *Big Social Data* [30]. Tais informações propiciam as corporações, instituições e governos traçarem perfis mais específicos e segmentados sobre estes usuários, deixando as tomadas de decisões mais concernentes aos seus propósitos.

2.1.4 *Big Social Data*

As redes sociais *online*, no presente, são repositórios gigantescos de informações pessoais, interações entre usuários e entre instituições, opiniões acerca de assuntos de natureza quase que infinita, identificadores de cenários de tendências e muito mais. Em face deste ambiente, surge, explicita e perceptível, a importância da análise de redes sociais para diversos segmentos, inclusive para entidades educacionais. Cabe aqui, por conseguinte, em conexão ao propósito desta pesquisa, explanar sobre alguns tópicos inerentes às redes sociais e os benefícios de sua análise.

Conforme Wasserman (1994), uma rede social se define pela existência de relacionamentos entre um conjunto de atores (usuários/perfis). Ainda, França et al. (2014), em oportuno artigo, compara uma rede social a um organismo vivo, dando sinônimo de usuário, às células deste. Sugestiona que os usuários de uma rede social (como num organismo), podem ter seu tempo de vida longo dentro da rede, ao passo que outros, estarão ali momentaneamente, com propósito curto e definido. Conteúdo dissertado neste citado artigo,

[29] Nome utilizado para designar as publicações feitas na rede social do *Twitter*.

[30] http://www.inf.ufpr.br/sbbd-sbsc2014/sbbd/proceedings/artigos/pdfs/127.pdf

ainda nos revela que, os participantes, agregados pelas características e/ou objetivos de um individuo de uma rede social, tendem a se relacionar com os pares de outro individuo, montando uma estreita ligação entres estes chamamos nós, concebendo então uma "teia" de usuários de grandeza imensurável.

O autor ainda oportunamente enfatiza:

> A tendência de pessoas se unirem e formarem grupos é uma característica de qualquer sociedade [Castells, 2000]. Esse comportamento é retratado, nos dias atuais, através do avanço das mídias sociais e comunidades online que evidenciam o poder de unir usuários ao redor do mundo. Conteúdos gerados por seus usuários atingiram alto grau de alcance através de seus comentários, relatos de acontecimentos quase em tempo real, experiências, opiniões, críticas e recomendações que são lidos, compartilhados e discutidos, de forma quase instantânea, em diversas plataformas disponíveis na Web.(FRANÇA et al.,2014)

Assim sendo, a tamanha grandeza de informações destes usuários, propicia farto material para análise, com objetivo de filtrar informação, que servirá de subsidio para uma infinidade de ações, tais como: táticas de propaganda e *marketing*, análise de tendências de mercado, formação política, promoção de eventos, divulgação de pesquisas, entretenimento, combate a malefícios – terrorismo, entre tantos outros.

Em vista disso é correto induzir ao raciocínio de quão importante pode vir a ser a análise de dos dados de redes sociais *online*, que dada a sua grandeza em volume e variedade, adiciona a si o adjetivo de *Big Social Data*, ou *Big Data* em redes sociais. Este volume de conteúdo produzido e compartilhado nas redes sociais *online*, pelo grande número de usuários geograficamente dispersos, é fonte de novas e contínuas informações, que podem ser agregadas as já existentes de diversas áreas. Esse desafio de analisar essa grande variedade de dados deve ser visualizado sob o prisma das ferramentas disponíveis de tratamento de dados não estruturados.

Identificar a relevância da análise de dados de redes sociais, bem como a sua correlação com entidades educacionais e não obstante a isso, a forma como estes dados poderiam ser coletados analisados e usados efetivamente para produzir subsídios diversos a uma gestão institucional, também fará, mesmo que indiretamente, parte deste enunciado de pesquisa.

2.1.5 O Contexto da Utilização de *Big Data* e *Cloud Computing*

Já é uma realidade que ambientes de computação em nuvem vêm sendo utilizados para o gerenciamento de dados em forma de *Big Data*. Tecnologias de nuvem associadas aos conceitos do *Big Data Analytics* tendem a oferecer um modelo de distribuição de baixo custo

para análise de dados, através das Bases de Dados Como Serviço (DaaS)[31] e Infraestrutura Como Serviço (IaaS)[32]. Segundo Lome (2009), a infraestrutura fornecida como serviços, fornece elasticidade, backup automático e agilidade na implantação a custos específicos da demanda necessária. Por outro lado, DaaS é capaz de propiciar o gerenciamento remoto de servidores alocados em uma infraestrutura externa, e com custos reduzidos.

De igual forma é importante mencionar que, à medida que a computação em nuvem cresce, empresas provedoras de serviços na nuvem intensificam a oferta de serviços e consumidores desses serviços aprimoram a construção de ambientes mais ágeis e eficientes para tirar proveito dessa oferta. É prudente levar a gestão das organizações de TI para ações que olhem para a computação em nuvem como a estrutura para oferecer suporte ao tratamento de dados não estruturados. Ambientes desta natureza requerem *clusters* de servidores que deem suporte a ferramentas de processamento de grandes e variados volumes de dados. Inerente ao fato, nuvens são constituídas em conjuntos de servidores, o que propicia tanto o aumento como a diminuição dos recursos ofertados ao usuário, conforme seja a sua real necessidade. É enfim, uma maneira econômica para prover suporte a tecnologias de *Big Data* e de análises mais avançadas. Estes citados recursos, e tantos outros, não estão mais sob responsabilidade de quem os usará, mas sim, estão sendo substituídos por plataformas terceirizadas com a finalidade de permitir o uso de serviços sob demanda, independente de localização física ou geográfica, associando um alto grau de transparência e custos adequados ao uso.

BRANTNER et al. (2008) adjetiva a computação em nuvem de *Utility Computing*, uma evolução dos serviços de TI, cujo principal objetivo é fornecer armazenamento, processamento e banda[33] escaláveis, na forma de mercadoria para consumo, através de provedores específicos, a um custo equivalente a reivindicação do usuário. *Backups*, disponibilidade de infraestrutura, acesso a leitura e gravação, são transparentes ao usuário. O provedor é o único responsável por tornar os dados disponíveis em tempo hábil, através de métodos replicantes de servidores.

Ainda segundo BRANTNER et al. (2008), o uso de serviços baseados em *Utility Computing* é igualmente significativo para os provedores, no sentido de que se houver a necessidade de uso de serviços de terceiros, estes serão obrigados a pagar apenas o

[31] *Database as a Service* (DaaS)
[32] *Infrastructure as a Service* (IaaS)
[33] Medida da capacidade de transmissão de um determinado meio, conexão ou rede, determinando a velocidade que os dados passam através desta rede específica.

equivalente a seu uso do que efetivamente recebem, dispensando investimentos iniciais em infraestrutura de TI. Contudo cabe ressaltar que de um modo geral, ambientes em nuvem demandam capacidade de suporte a uma alta taxa de processos de leitura e escrita de dados, além de atualizações e análises.

Estas premissas levam a concluir que os usuários estão movendo seus dados para a nuvem podendo assim interagir com eles de forma simples e independente de local de acesso. Cabe, dentro deste contexto, apresentar mais alguns detalhes acerca do assunto *Cloud Computing*.

2.1.5.1 Modelos de Serviço da Computação em Nuvem

Mell e Grance (2011) identificam três padrões de arquitetura para modelos de serviço na computação em nuvem. São eles:

a) Infraestrutura como Serviço (IaaS) - Significa o provisionamento de servidores virtuais e outros dispositivos, pretendido pelo usuário e tarifado por fatores, como o número de servidores virtuais e quantidade de dados armazenados ou trafegados, além de outros componentes. Assim sendo cabe ao usuário apenas contratar os recursos computacionais fundamentais para a implantação das suas atividades, sistemas operacionais e aplicações.

b) Software como Serviço (SaaS) - Sistema onde o provedor da nuvem disponibiliza ao usuário uma aplicação qualquer, um sistema ou software, que está por sua vez implantado na de nuvem. É um modelo onde a aquisição e ou utilização de um software não está relacionada a compra de licenças.

c) Plataforma como Serviço (PaaS) - Refere-se a um modelo de serviços, fornecido pelo provedor da nuvem, que fica entre o SaaS e IaaS, visando flexibilizar a utilização de recursos. Nele é facultado a utilização de software e o desenvolvimento de aplicações próprias (usuário). O usuário não controla a infraestrutura de servidores (rede, armazenamento e sistemas), mas consegue controlar as aplicações implantadas e configurações do ambiente da nuvem.

2.1.5.2 Modelos de Implantação da Computação em Nuvem

Os modelos de implantação, ainda conforme Mell e Grance (2011), simbolizam a disposição dos ambientes da computação em nuvem e os classificam da seguinte forma:

a) Nuvem privada- Neste modelo toda infraestrutura da nuvem pertence e é controlada por uma organização ou entidade exclusiva, tendo sua estrutura situada na própria empresa ou ainda terceirizada em outro local;

b) Nuvem pública- Significa quando a infraestrutura de nuvem é disponibilizada ao público em geral, recaindo seu controle sobre uma organização responsável também pela sua comercialização. Neste formato, qualquer usuário pode se beneficiar dos serviços prestados pela nuvem, bastando para isso ter conhecimento sobre como acessá-la;

c) Nuvem comunitária- Propõe o formato de infraestrutura compartilhada. A infraestrutura da nuvem é compartilhada por mais de uma organização com interesses comuns; e

d) Nuvem híbrida- este modelo ocorre quando a infraestrutura da nuvem é uma composição de dois ou mais modelos anteriores, conectadas por meio de uma tecnologia padronizada.

Nesse sentido, Vieira et al. (2012) destacam que a uso da computação em nuvem para a utilização de bancos de dados *NoSQL* já é uma realidade para empresas como Google Facebook, IBM, Twitter entre outras, para dar implementação ao processamento analítico de *logs*, *posts*, trânsito web e demais tarefas que fazem uso da alta escalabilidade, alta disponibilidade, escrita e leitura com baixa latência, armazenamento eficiente e acesso rápido em tempo real, além de ser considerada uma forma eficiente e de baixo custo.

Os modelos de computação em nuvem podem ajudar a acelerar o potencial para soluções de análise escaláveis. As nuvens oferecem flexibilidade e eficiência para acessar dados, oferecer *insights* e impulsionar valor. Nesse contexto, cabe mensurar a estreita relação entre as caraterísticas da computação em nuvem e a parametrização dos dados de *Big Data* e, por conseguinte, também sobre dados não estruturados.

2.2 Dados não estruturados

A quantidade de dados gerada diariamente em vários domínios digitais na ordem de terabytes/petabytes/zettabytes implica, como já outrora mencionado, em novos e complexos desafios na forma de manipulação, armazenamento, trato e utilização de consultas nas mais diversas áreas da computação, especialmente no que tange à recuperação de informações (CUZZOCREA, 2011).

Sob este prisma, e com a necessidade de gerir dados cujos formatos são dificilmente acomodáveis em sistemas relacionais, considera-se inadequado o uso dos tradicionais sistemas de gerenciamento de banco de dados (SGBD) para com o trato destes, agora atendendo pelo pseudônimo "*Big Data*", impondo necessidades como: execução de consultas com baixa latência, tratamento de grandes volumes de dados, escalabilidade, suporte a modelos flexíveis de armazenamento de dados, e suporte simples à replicação e à distribuição dos dados.

Para solucionar os diversos problemas e desafios gerados pelo *Big Data*, emerge como tendência o movimento denominado *NoSQL* (*Not only SQL*), predestinado a prover soluções inovadoras de armazenamento e processamento de grande volume de dados, que na sua maioria necessitam ter uma arquitetura escalar, fácil, horizontal, onde novos dados possam ser inseridos de forma eficiente, coexistindo harmoniosamente com ambientes em nuvem (AGRANAL, 2011). Empresas, como Google e Facebook, adjetivadas como grande geradoras de dados, já se utilizam de tal para o processamento analítico de logs, consultas, inserções convencionais, entre inúmeras outras tarefas sobre seus dados. De igual forma encontram-se exemplos de instituições financeiras, agências governamentais, *e-commerce*. Isto nos prova a existência de grandes demandas para soluções que tenham desempenho em diferentes modelos de dados complexos, com flexibilidade, escalabilidade e suporte.

Atualmente referência para bases de dados não relacionais, o termo *NoSQL* é originário cronologicamente de 1998. Nesta ocasião, ao não utilizar linguagem SQL em programas de consultas de dados, Carlo Strozzi concebe o termo ao que primeiramente chamara de "Strozzi SQL". O termo voltou a ser utilizado mais tarde em 2009 numa conferência organizada por Johan Oskarsson acerca de bases de dados do tipo *open source* com o nome "*NoSQL meetup*" (ABRAMOVA et al., 2014).

Oportunamente, Sadalage e Fowler (2013) afirmam que a tecnologia *NoSQL* surgiu por força da necessidade de se tratar com eficácia grandes volumes de dados, tendo em vista o seu crescimento exponencial ao longo dos últimos anos e por esta ser uma tendência igualmente futura. Os autores supracitados, ainda defendem que, o uso desta tecnologia em novos sistemas da a estes mais flexibilidade que os tradicionais modelos de bancos de dados relacionais, por não serem amarrados em esquemas rígidos, consistirem-se distribuídos, escaláveis e de melhor desempenho, além de não necessitarem de uma infraestrutura robusta para armazenar seus dados.

Por sua vez, o Google, através de artigo publicado em 2006 sob o titulo "*BigTable: A Distributed Storage System for Structured Data*", emerge novamente o termo *NoSQL* como proposta de tratamento de dados, com a promessa de ser um banco de dados escalável e tolerante a falhas, de consulta muito rápida, pois já indexava os dados assim que estes eram inseridos no banco.

Dissemina-se, assim, um novo conceito para tratamento de dados, conceito que imediatamente cai nas graças das comunidades *open source*, que passaram a desenvolver inúmeras soluções de banco de dados não relacionais.

Associando estes fatores à necessidade de prover informações por meio de dados cujos formatos não se enquadravam em sistemas relacionais tradicionais, espalhados por múltiplos servidores e em quantidade significativamente grande e exponencial, corroboraram em termos fáticos para o surgimento efetivos das bases de dados *NoSQL*.

Sua maior premissa era permitir processamento de dados de forma rápida e eficiente, priorizando o desempenho, e deixando de lado de vez os padrões relacionais dos modelos antigos. Segundo Leavitt (2010), estas prerrogativas possibilitam armazenar e recuperar os dados de forma rápida e eficiente, independentemente da sua estrutura e conteúdo.

O autor ainda enuncia que as bases de dados *NoSQL* possuem uma arquitetura distribuída tolerante a falhas, que se baseia na distribuição dos dados por vários servidores. Caso um servidor deixe de funcionar, o sistema manter-se-á em funcionamento garantindo assim elevados índices de disponibilidade.

Outra importante caraterística *NoSQL* é a escalabilidade horizontal, que consiste em aumentar o número de máquinas do sistema dividindo o processamento dos dados conforme as necessidades do mesmo, de forma a obter sempre elevados níveis de desempenho. Cabe ressaltar que, quando um sistema "escala" horizontalmente, ele adiciona mais nós ao sistema, um novo computador ao *cluster*. Por outro lado, escalar verticalmente um sistema implica em agregar recursos de processamento, memória ou disco, por exemplo, a um único nó.

A flexibilidade e a manipulação simplória também caracterizam os bancos de dados não relacionais. Denotam simplicidade na sua manipulação e configuração, ao passo que não possuem esquema determinístico, como nos SGBD em SQL, o que facilita a distribuição dos dados entre vários servidores, cada um responsabilizando-se por uma parte dos dados a serem trabalhados.

2.2.1 Características Inerentes as Bases de Dados *NoSQL*

Para tentar descrever as cartacterísticas da tecnologia *NoSQL* para tratamento de dados, necessita-se inicilamente lembrar da herança das mesmas no que tange ao termo *Big Data*. Invariavelmente a associação entre *Big Data* e *NoSQL*, acerca de seus adjetivos funcionais quanto ao tratamento dos dados, lhes é singular. Contudo, um detalhamento mais peculiar, a seguir, se faz necessário.

Cabe inicialmente ponderar que o Teorema CAP (*Consistency, Availability, and Partition Tolerance*), conforme expõe Souza (2010), retrata os três requisitos que afetam os sistemas distribuídos:

a) consistência - o sistema fica pronto para ser usado logo após a inserção de um registo;

b) disponibilidade - o sistema permanece ativo durante um período de tempo; e

c) tolerância a falhas - o sistema é capaz de funcionar em caso de falha dos seus componentes.

Ainda segundo o autor, e com base no citado teorema, não é possível, num sistema de dados distribuído, coexistirem estes três requisitos. É possível ter dois deles concomitantemente, sendo que a escolha de quais destes estará diretamente relacionado aos requisitos impostos. Estes paradigmas formam a infraestrutura conceitual para o modelo de consistência eventual (*Basically Available, Eventual, Consistency*) que é utilizado pelas bases de dados *NoSQL*. Por este novo prisma sugere-se que durante um determinado tempo não existirão atualizações e, nesse interim, todas as atualizações pendentes serão propagadas por todos os nós do sistema, tornando o sistema consistente.

Além disso, são numerosas as características que podem ser aferidas, na totalidade, quando se tenta trazer à tona o comportamento dos bancos de dados *NoSQL*. Contudo, algumas delas se sobressaem e transperecm quanto aos adjetivos destes. De acordo com Näsholm (2012), os bancos de dados não estruturados em sua maioria comparticipam de uma coleção de características, podendo apresentar algumas exceções devido ao amplo contexto em que se enquadram. Sob um prisma generalizado, seriam eles:

a) Escalabilidade (horizontal) – Escalabilidade implica em ser possível a expansão de um sistema de armazenamento de dados, sem aferir a esta, custos adicinais, ou então, ao menor custo possível. Expressa, nesse sentido, ser capaz de manobrar uma ascendente carga de trabalho uniformemente, ou seja, estar apto a crescer, expandir-se. Um sistema de escalada horizontal é autônomo para agregar mais nós ao sistema de *cluster*, como por exemplo, um novo computador ou dispositivo de processamento, ao passo que um

sistema que tem características escalares verticais, adiciona recursos a um único nó, através do incremento de memória ou espaço fisico, por exemplo. Teoricamente não se impõe limites quanto aos números de nós em um *cluster*, o que lhe são atribuidos sim, é a uma noção de realidade e investimento. Segundo Pritchett (2008):

> Quando um banco de dados relacional cresce além da capacidade de um único nó é preciso se optar por escalabilidade vertical ou horizontal. Escalabilidade vertical não é uma opção para sistemas que lidam com grandes volumes de dados. Assim, a opção é escalar horizontalmente. (PRITCHETT, 2008)

Os bancos de dados *NoSQL* possuem uma peculiaridade comum: a consistencia eventual. Esta característica que lhes é também atribuida é primordial para o êxito em atingir niveis elevados de escalabilidade;

b) Grandes Volumes – a principal proposta dos sistemas *NoSQL* é atender aos requisitos de gerenciamento de grandes volumes de dados em escala de improvável sucesso, se adotados os bancos de dados tradicionais;

c) Distribuídos - Os sistemas distribuídos compõem a base do processamento e do armazenamento dos bancos de dados *NoSQL*. Replicar dados em inúmeros nós e servidores, é fator preponderante para o sucesso de um sistema que visa redundância e alta disponibilidade;

d) Simplicidade - Os bancos de dados *NoSQL*, em sua maioria, apresentam-se "simples" quanto ao uso e configuração. Significa que o conjunto das suas funções são facilmente acessiveis sob o prisma do uso comum por outro sistema (PALMER, 2010). O sistema de tratamento de dados MongoDB[34], que utiliza o formato JSON[35], é um exemplo de tal afirmação. Seguindo-se a política aqui descrita, não se faz necessária a contratação de especialistas em bancos de dados para o gerenciamento em bancos de dados *NoSQL*, ficando esta tarefa a cargo dos desenvolvedores;

e) Flexibilidade – Bancos de dados *NoSQL* são livres de imposição de um esquema. Não denotam estrutura fixa ou normatização para com o armazenamento dos dados não estruturados, mantendo um desempenho considerado aceitável para as exigências de sistemas de *Big Data*. Contrariamente ao que denota os bancos SQL, tal abordagem facilita a distribuição dos dados entre vários nós, onde cada um destes nós é encarregado de "cuidar" de apenas uma fatia da totalidade dos dados. Desta forma, os

[34] https://www.mongodb.com/
[35] *JavaScript Object Notation* - Notação de Objetos JavaScript - www.json.org

clientes estão aptos a armazenar os dados na forma que escolherem, sem a obediência a estrutura pré-definida, como fariam nos bancos de dados relacionais;

f) BASE x ACID - Segundo Pritchett (2008), apesar de as propriedades de transações ACID (Atômica, Consistente, Isolada e Durável) propiciarem um desenvolvimento de sistemas mais simples, esta deve ser preterida em relação à abordadgem BASE (*Basically Available, Soft state, Eventualy consistent*). Enquando ACID cria dificuldades quanto ao desenvolvimento de bancos de dados no modelo distribuido, BASE detem características de disponível, leve e consistente, além de ser tolerante a inconsistências temporárias quando se prioriza disponibilidade; e

g) Diponibilidade - Um sistema de alta disponibilidade é um sistema capaz de resistir à falhas e manter os serviços ativos o máximo de tempo possível. Implica também em alto gerenciamento de memória e capacidade de processamento inerente às respostas exigidas pelas requisições do sistema, com tempo de resposta otimizado.

2.2.2 Tipos de bases de dados *NoSQL*

Diferentes abordagens são utilizadas para delinear classificação aos bancos de dados *NoSQL*. Contudo, dentro dessa nova classe de tecnologias, é correto inferir que os bancos de dados não relacionais são classificados em Bancos de Esquema Chave-Valor (*Key-Value-based*), Bancos de Dados orientados a Documentos (*Document-based*), Bancos de dados orientados a Colunas (*Column-based*) e Bancos de Dados orientados a Grafos (*Graph-based*).

2.2.2.1 Bancos de Dados Chave-Valor

Caracterizam-se como as mais simples e de onde se tira o melhor desempenho dentre os demais tipos. Utiliza uma chave para cada campo e um valor e na forma de tabelas de *hash* distribuídas (DHT). Seu conteúdo pode ser armazenado em qualquer formato de dados. Utiliza-se na prática de um conjunto de algoritmos ou matrizes que efetuam buscas em todos os dados dos arquivos compartilhados onde o acesso a esses dados é sempre efetuado através de sua chave – primária e única (ABRAMOVA e. al., 2014).

Por possuir como principal vantagem a boa escalabilidade horizontal e o alto desempenho, o modelo Chave-Valor é usado principalmente em aplicações onde as mudanças nos dados ocorrem com alta frequência, não sendo indicado para aplicações inerentes a consultas de alta complexidade, devido a sua baixa capacidade de indexação. Os nós são

programados para encontrar assuntos específicos em arquivos e trazê-los como resultado da busca.

As bases de dados do tipo Chave-Valor mais conhecidas, por assim dizer, no ambiente comercial dos dados *NoSQL* são Riak[36], Dynamite, Redis[37], Amazon DynamoDB[38], Azure Table Storage, Berkeley DB[39], e até mesmo o Cassandra (mesmo que este tenha propriedades orientadas para a coluna).

2.2.2.2 Bancos de Dados orientados a Documentos

Estes tipos de bancos de dados perfazem armazenamento sob a forma de conjuntos de documentos e são baseados em pares de chave-valor, tendo um esquema altamente flexível. Por ter um esquema diferenciado das demais bases de dados *NoSQL*, no sentido de que cada documento é constituído por um conjunto de campos armazenados de forma não-estruturada, identificados e associados por uma chave única, o modelo de banco de dados orientado a documentos é superior no que tange à forma de consultas, pois permite consultas na forma de conjuntos de documentos, com ordem ou restrições sobre os resultados. Por ser considerada a base de dados mais versátil, são opções ideais para tratamento de dados semiestruturados, armazenados em um formato padrão, como XML[40], JSON[41] ou BSON[42]. De acordo com Kuznetsov & Poskonin (2014), são apropriadas para problemas onde não se tem certeza do tipo de dado a ser investigado e trabalhado, onde o desempenho é irrelevante e o foco principal está sim na premissa de um bom desempenho na consulta destes dados e no armazenamento de grandes volumes.

Para Kaur et al. (2013), a forma de trabalho em detrimento da complexidade associada à alta escalabilidade, tanto para dados estruturados, semiestruturados ou não-estruturados, é uma das suas principais vantagens sobre os demais modelos. Sob o olhar de Abramova et al. (2014), sistemas como CouchDB e o MongoDB, representam assim, num contexto prático de aplicação, uma ótima escolha para se trabalhar com grandes quantidades de documentos que

[36] basho.com
[37] https://redis.io/
[38] https://aws.amazon.com/pt/dynamodb/
[39] www.oracle.com
[40] **XML** (*e**X**tensible **M**arkup **L**anguage*) é uma recomendação da W3C para gerar linguagens de marcação para necessidades especiais. Cria uma infraestrutura única para diversas linguagens.
[41] **JSON** (*JavaScript Object Notation* - Notação de Objetos JavaScript) é uma formatação leve de troca de dados, fácil de interpretar e gerar. Está baseado em um subconjunto da linguagem de programação JavaScript.
[42] **BSON (**Binário JSON) é uma forma binária para representar estruturas de dados simples, matrizes associativas (objetos chamados ou documentos em MongoDB) e vários tipos de dados de interesse específico para MongoDB.

são armazenados em arquivos tais como XML, documentos de texto, emails, sistemas de comércio eletronico na Internet, dentre outros de mesmas características.

2.2.2.3 Bancos de dados orientados a colunas

É um modelo considerado apto a suportar grande quantidade de dados. Em relação ao modelo Chave-Valor, apresenta maior complexidade, pelo fato da orientação passar dos registros para a forma de colunas onde nem todas as linhas têm a mesma quantidade de colunas e, sim, uma linha corresponde a um conjunto de colunas associadas à mesma chave primária, o que onera um pouco a escrita de um novo registro. Isso se traduz num menor esforço computacional aumentando o desempenho do sistema.

Segundo Abramova et al. (2014), o modelo de banco de dados orientado a colunas é o ideal para sistemas que convergem para o tratamento de estruturas de dados complexas e de grande volume, afirmação amparada na característica das colunas poderem ser armazenadas por famílias de colunas, com a finalidade de facilitar a organização e a distribuição dos dados. Na prática, são ideais em ocasiões em que o número de operações de escrita (*write*) é superior ao numero de operações de leitura (*read*), caso de sistemas com elevado número de requisições.

Cassandra e Hbase formam a dupla principal para exemplificar este modelo de base de dados, seguidos ainda por outros como BigTable (Google)[43], Hypertable[44], Infobright[45] e até mesmo Riak.

2.2.2.4 Bancos de Dados orientados a Grafos

Os bancos de dados de grafos são considerados os de maior complexidade, primordialmente pelo fato de guardarem objetos e não registros como os demais, onde a busca pelos dados é feita através da navegação por estes objetos, armazenando arestas e vétices como representação da associação que estes mesmo dados possuem entre si (ALVARES, *et.al*, 2016).

Considerado ideal para o trato de dados de redes sociais e do mesmo modo útil como ferramenta para extração de dados para construção de conhecimento competitivo entre empresas, as bases de dados orientadas a grafos detêm capacidade de armazenar informação

[43] https://cloud.google.com/bigtable/?hl=pt-br
[44] hypertable.org
[45] http://www.ignitetech.com/solutions/information-technology/infobrightdb

obtida através da relação entre nós, além de não possuirem uma estrutura previamente definida.

Sua arquitetura funcional baseia-se no uso de nós e arestas para representar dados armazenados, sendo que os nós contêm propriedades na forma de pares Chave-Valor e os relacionamentos têm sempre um nome associado e uma direção contendo um nó remetente e um nó destinatário, muito úteis para armazenar informações em modelos com muitos relacionamentos (ROBINSON et al., 2013).

Neo4j[46], HyperGraphDB[47], AllegroGraph[48] e VertexDB[49], são exemplos de bancos de dados orientados a grafos.

Um enquadramento ou classificação quanto à forma da base de dados, com a finalidade de expor o grande número de opções existentes, pode ainda ser encontrada e detalhada em outras fontes literárias, por hora não pertinente.

2.2.3 Haddop

No contexto *Big Data*, que engloba um volume muito grande de dados, a utilização de um mecanismo para armazenar informações ao longo de várias máquinas é indispensável. O projeto Apache Hadoop, criado por Doug Cutting[50], é uma das soluções mais conhecidas para *Big Data* atualmente. É um sistema de alto desempenho e tolerante a falhas para armazenamento e processamento de dados, na forma de uma solução *open source* licenciada pela Apache, para computação distribuída, escalável e segura. Sua finalidade é oferecer uma infraestrutura para armazenamento e processamento de grande volume de dados, provendo escalabilidade linear e tolerância a falhas, permitindo assim armazenar e processar dados grandes em um ambiente distribuído entre *cluster*s de computadores pelo uso de métodos simplificados de programação. Segundo White (2012), Hadoop vem quebrando recordes tanto na agregação de máquinas no *cluster* como em velocidades de processamento.

No Hadoop, a alta disponibilidade não fica sob a responsabilidade do hardware. Nele a detecção de falhas que eventualmente venham a ocorrer, até mesmo porque as máquinas do *cluster* estão sujeitas a errar, é feita na camada de aplicação. Seus serviços estão respaldados em dois agentes principais: o Sistema de Arquivos Distribuídos Hadoop (HDFS), responsável

[46] https://neo4j.com
[47] hypergraphdb.org
[48] https://franz.com/agraph/allegrograph
[49] www.dekorte.com/projects/opensource/vertexdb
[50] Arquiteto Chefe da Cloudera, Inc. Conselheiro da Proximal Labs, Inc. Fundou os projetos Apache Hadoop, Nutch e Lucene.

pelo armazenamento de dados, com notória confiabilidade, e o Hadoop *MapReduce*, responsável pelo processamento em alto desempenho destes dados. Imediatamente aqui, cabe maior detalhamento sobre estes artefatos.

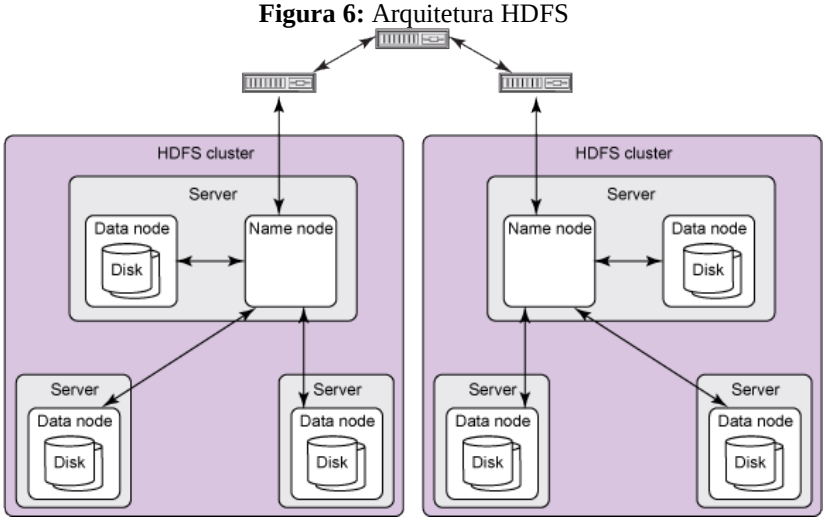
2.2.4 HDFS

No formato de um subprojeto do projeto da Apache Hadoop (*Apache Software Foundation*), podemos sinteticamente conceituar o *Hadoop Distributed File System* (HDFS), como sendo um sistema de arquivos usado no tratamento de dados do ecossistema Hadoop. Como já previamente mensurado, o Hadoop é ideal para armazenar grandes quantidades de dados, do porte de terabytes e petabytes, permite armazenar e processar dados grandes em um ambiente distribuído entre *clusters* de computadores usando simples modelos de programação e, por conseguinte, é o HDFS que o Hadoop utiliza como sistema de armazenamento.

O HDFS outorga a conectividade entre computadores (nós do *cluster*), através da qual os arquivos de dados são distribuídos. É permissível acessar e armazenar os arquivos de dados em um formato contínuo e as partes executáveis do sistema, bem como comandos são executados seguindo os paradigmas do *MapReduce* (descrito mais adiante). Este sistema armazena os chamados metadados em locais diferentes dos dados da aplicação. Estes metadados são armazenados em um servidor especificamente designado para esta função e lhe é dado o nome de *NameNode*. Por outro viés, os dados da aplicação, são gravados em outros servidores alheios ao primeiro e são batizados de *DataNodes*. Ambos os servidores se comunicam seguindo as regras baseadas em protocolo TCP. Quam (2000) orienta sobre que metadados são definidos como atributos que descrevem dados, em um nível de abstração superior aos dados, utilizados para descrever a origem, ou proveniência de certo conjunto de dados.

O HDFS é altamente tolerante a falhas e é plausível de uso em computadores “simples”, de baixo custo. Ele consegue isso através da replicação dos blocos de arquivos, especificada pelo aplicativo em quantidade que julgar pertinente. Quem agrega a responsabilidade pela replicação é *NameNode*. A integridade dos dados é garantida pela validação de soma de verificação nos conteúdos dos arquivos do HDFS armazenando somas de verificação calculadas em arquivos ocultos, separados, no mesmo *NameSpace* que os dados reais. O *NameSpace* do HDFS é armazenado usando um log de transação mantido por cada *NameNode*. A Figura 6 mostra, arquiteturalmente, a maneira como age o HDFS.

Figura 6: Arquitetura HDFS



Fonte IBM.com

2.2.5 *MapReduce*

A grande escalada de volume e variedade de dados do *Big Data* vem acarretando significativas dificuldades no que tange à capacidade de processamento de dados em informações úteis desta natureza. Para solucionar este gargalo de processamento, Lin e Dyer (2010) sugerem que a única alternativa plausível para solucionar este revés advém da aplicação do paradigma de dividir e conquistar. Estrategicamente significa particionar um grande imbróglio em fatias de problemas menores. Assim inicialmente caracteriza-se o surgimento do conceito aplicável do *MapReduce*.

Ainda citando White (2012), *MapReduce* pode ser definido como um paradigma de programação voltado para processamento em lotes (*batch*) de grande volume de dados ao longo de várias máquinas, obtendo resultados em tempo razoável. Utiliza-se o conceito de programação distribuída para resolver problemas, adotando a estratégia de dividi-los em problemas menores e independentes.

MapReduce é disposto em conjunto com um sistema de arquivos distribuídos, especialmente projetado para dar suporte a aplicações que necessitam tratar grandes volumes de dados. Conceituando um pouco mais o *MapReduce*, Paiva e Revoredo (2016), fazendo-se valer da obra de Dean e Ghemawat, salientam que:

> *O MapReduce é um modelo de programação paralela para grandes volumes de dados. Ele é inspirado na estratégia dividir e conquistar, mas abstrai do programador a complexidade dos problemas típicos do gerenciamento de aplicações distribuídas, permitindo que o desenvolvedor possa se dedicar apenas à*

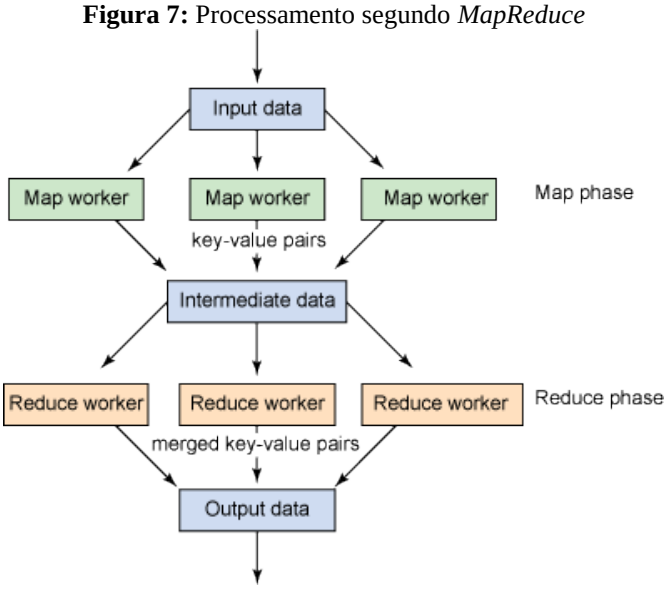
> *solução do problema a ser tratado, deixando que a aplicação execute a distribuição e o paralelismo.* (PAIVA, REVOREDO, 2016)

O modelo de trabalho do *MapReduce*, faz uso de *clusters* de computadores para o processamento das tarefas, distribuindo-as entre os nós. Cada nó é efetivamente uma máquina do *cluster*, lhe sendo atribuído o adjetivo de "mestre" ou "escravo". Os nós mestres recebem a tarefa de gerenciar o processamento efetivamente executado pelos nós escravos.

Numa referência prática, segundo Dean e Ghemawat (2004), a função do operador *Map* é dividir o problema inicial em grupos menores, que serão entregues aos outros nós do *cluster*. Em contrapartida, a operação *Reduce* retém para si a tarefa de tratar a informação oriunda inicialmente dos grupos de dados entregues pela função *Map*, dando assim uma resposta ao problema original. Ao contrário do SQL, os processos de *MapReduce* armazenam os dados de forma persistente para posterior consulta. White (2012) complementa enfatizando que o processamento das tarefas é diluído em três etapas dissemelhantes: a etapa "*map*", a etapa "*reduce*", estas acessíveis ao programador, e uma etapa intermediária denominada "*shuffle*", criada pelo sistema em tempo de execução.

Segundo Dean e Ghemawat (2010), o *MapReduce* não requer programadores renomados em construção de sistemas distribuídos, ao contrário, permitem a programadores sem experiência nesta específica área, a criação de sistemas capazes de tratar grandes quantidades de dados. As tarefas relacionadas ao escalonamento, interação entre servidores e tolerâncias a falhas, por exemplo, ficam sob a responsabilidade do *MapReduce*, mantendo maior foco durante a construção do sistema, voltado para a parte referente ao tratamento dos dados em si. A Figura 7 e Figura 8 sintetizam o processamento segundo o paradigma de mapear e reduzir.

Figura 7: Processamento segundo *MapReduce*



Fonte IBM.com (2017)

Figura 8: Exemplo de *MapReduce*

Fonte IBM.com (2017)

2.2.6 Arquiteturas *NoSQL*

Padrões de armazenamento de dados estruturados são amplamente difundidos no meio tecnológico por décadas, estão presentes e são de corriqueiro trato por instituições de variados segmentos. Contudo, o atual momento tecnológico permite que, além dos tradicionais dados estruturados, exemplificados por tabelas em bases de dados SQL, os dados semiestruturados e principalmente os dados não estruturados, sob o pseudônimo *NoSQL*, estejam em evidência. Armazenar, tratar e aproveitar resultados de dados não estruturados exige uma forma diferenciada do que costumava implementar até então. Arquivos de texto, *logs*, vídeos, imagens, *streaming* de áudio, e tantos outros passam a integrar o contexto da análise de dados.

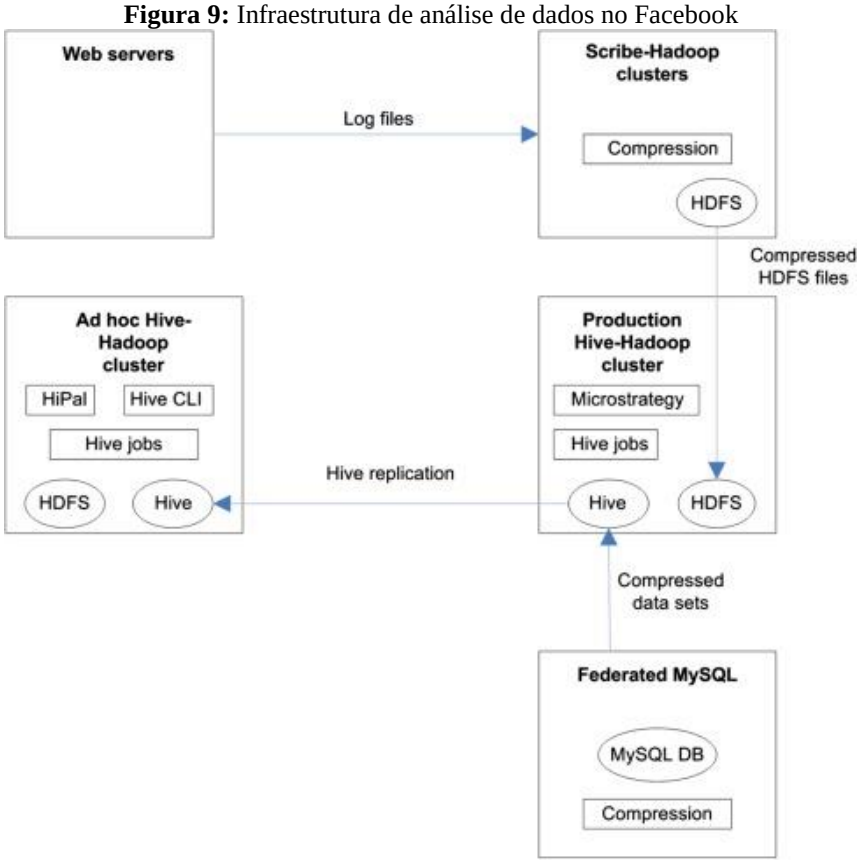
A manipulação de dados não estruturados, para além do que faz o próprio sistema de arquivos, exige novas arquiteturas. Os mais significativos exemplos para estas arquiteturas encontram-se no ecossistema Hadoop e no algoritmo *MapReduce* (ambos têm atenção especial e a parte nesta dissertação).

Ao se utilizar arquiteturas compostas de, por exemplo, Hadoop, *MapReduce*, bases de dados *NoSQL*, *Data Warehouse*[51], conjuntamente e não isoladamente um ou outro, está caracterizado o trabalho com *Big Data*.

MEIER (2013) possibilita mostrar funcionalidades de arquiteturas apresentadas na implementação de empresas como Facebook, LinkedIn e Oracle. O Facebook coleta dados estruturados e baseados em fluxo dos usuários, que é aplicado para análise de dados baseada em lote, conforme mostra a Figura 9.

[51] Depósito de dados digitais que serve para armazenar informações detalhadas relativamente a uma empresa.

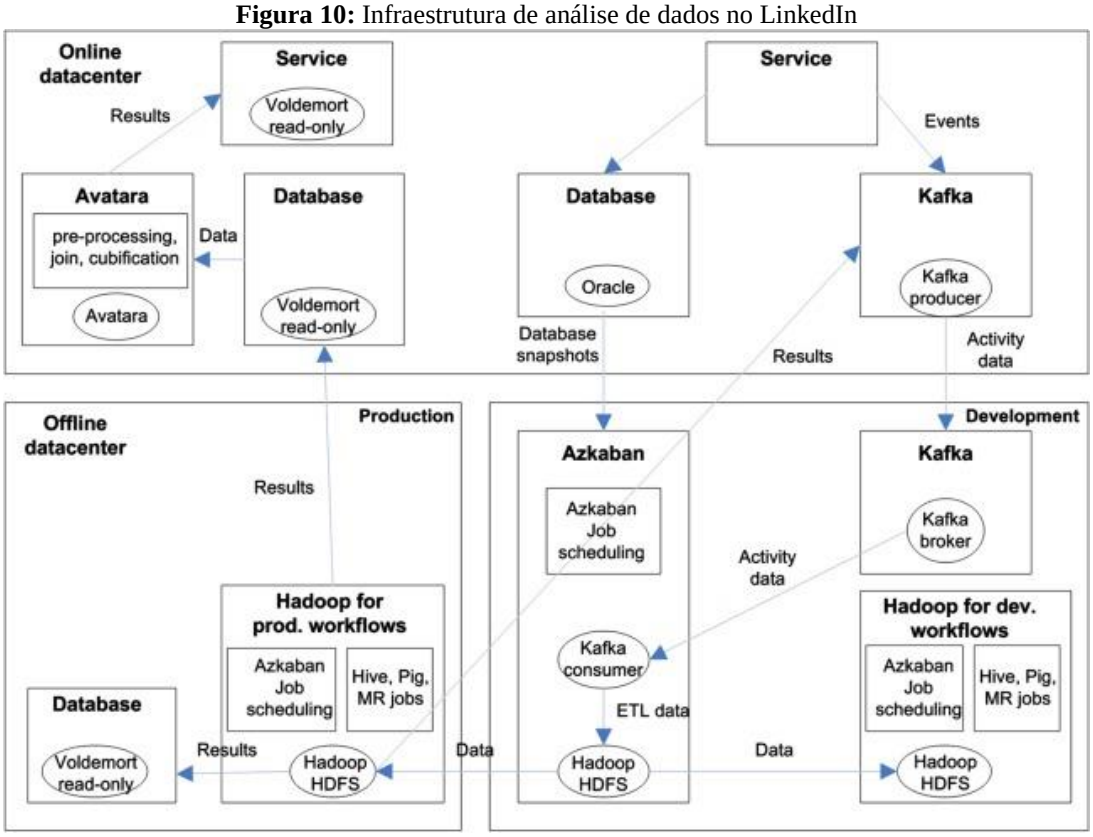
Figura 9: Infraestrutura de análise de dados no Facebook



Fonte: Pääkkönen e Daniel Pakkala (2016)

O mesmo autor, através da Figura 10, mostra que o LinkedIn também coleta dados estruturados e não estruturados, que são analisados em ambientes de desenvolvimento e produção e fornece serviços aos usuários finais com base na análise de dados.

Figura 10: Infraestrutura de análise de dados no LinkedIn

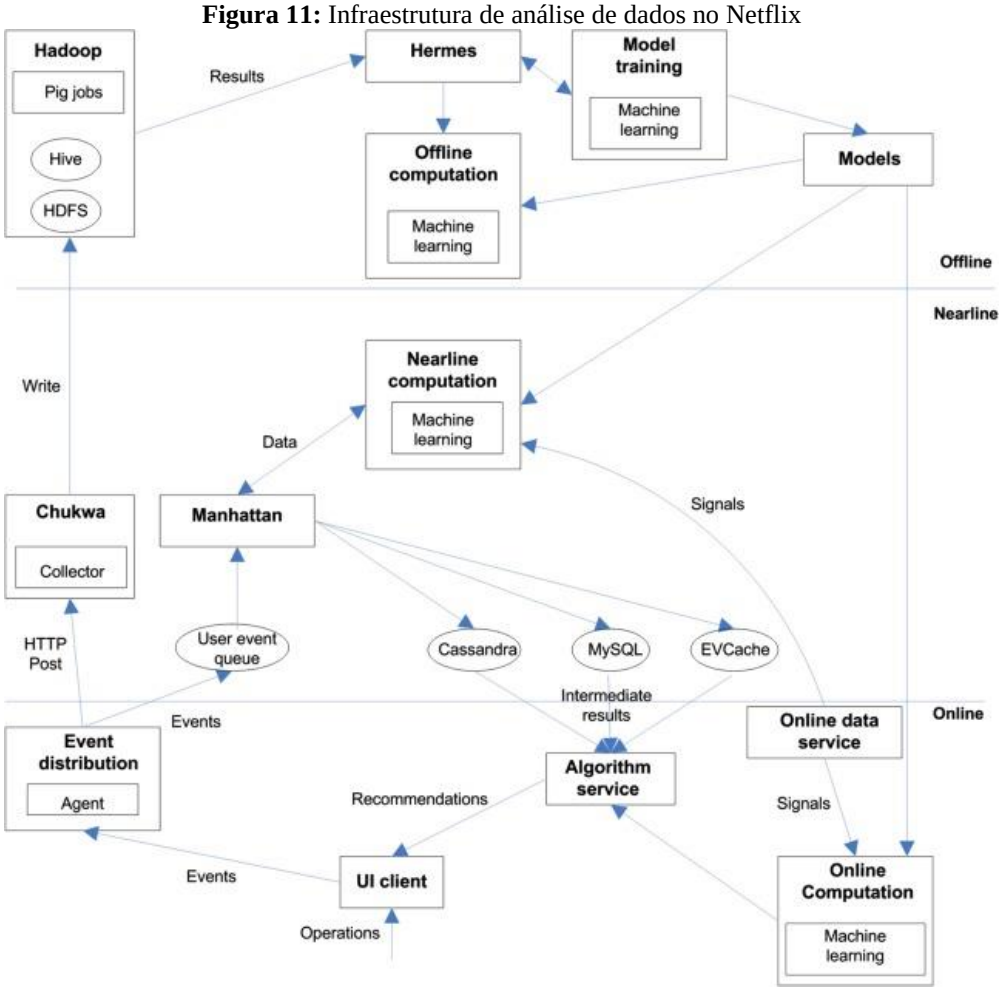


Fonte: Pääkkönen e Daniel Pakkala (2016)

O Twitter[52] lida principalmente com *tweets*, que possuem requisitos de processamento em tempo real. O Netflix[53] é um serviço comercial de streaming de vídeo para usuários finais que coleta processa e analisa eventos de usuários em ambientes *online*, *offline*, além de análise de dados em tempo real, visto por meio da Figura 11.

[52] https://twitter.com
[53] https://www.netflix.com/br

Figura 11: Infraestrutura de análise de dados no Netflix



Fonte: Pääkkönen e Daniel Pakkala (2016)

2.2.7 Arquitetura Lambda

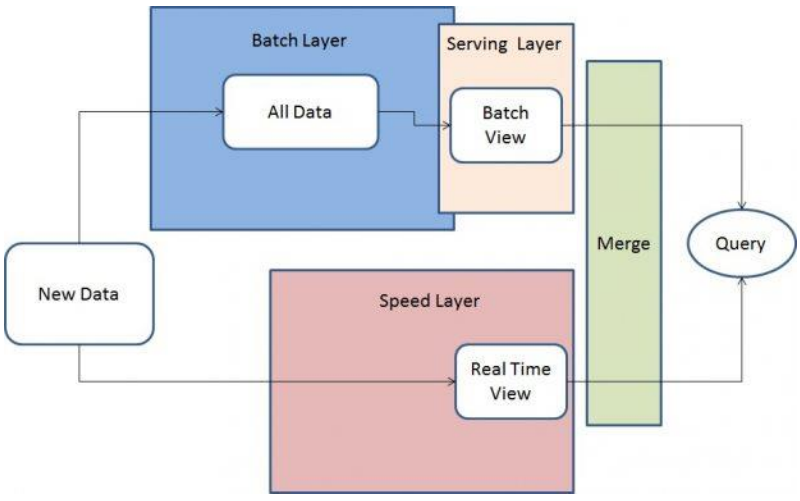
Já fora aqui citado, em tópico pertinente, que os sistemas convencionais de tratamento de dados, quando se trata do processamento de dados de grande volume, com atualizações frequentes, acabam por se tornar complexos, quase impossíveis de prover escalabilidade além de estarem mais suscetíveis a erros humanos. De igual forma também já se discorreu sobre os requisitos de variedade, volume e velocidade, de que são inerentes à concepção de sistemas de *Big Data*.

Tendo como premissa este cenário é que surge a arquitetura Lambda que, segundo Marz e Warren (2015), é uma arquitetura que forma a base para a construção de sistemas de *Big Data*, sob a forma de uma série de camadas. A cada camada é destinada a função de satisfazer um subconjunto de propriedades baseadas na funcionalidade da camada inferior. Os sistemas devem então ser capazes de lidar, em velocidades próximas a tempo real, com um enorme volume de dados, provenientes de origens variadas e distintas.

Mais especificamente, a arquitetura Lambda é dividida em três camadas, vistas na Figura 12: *batch layer*, *speed layer* e *serving layer* e diversas tecnologias podem ser dadas como exemplo da utilização dessa arquitetura na implementação de sistema de *Big Data*. Por exemplo: Kafka para a distribuição dos dados, Hadoop e Spark na camada *batch*, Spark ou Storm para processamento em tempo real dos dados e bancos *NoSQL* como HBase ou Cassandra na *speed layer* e para integração na *serving layer*.

Remonta a Nathan Marz, enquanto trabalhava no Twitter, a criação de uma arquitetura genérica propondo que uma mesma massa de dados dará origem a fluxos independentes de análise, sendo o primeiro – denominado "*batch layer*", responsável por persistir os dados (um banco de dados *NoSQL* ou em um sistema de arquivos distribuídos) e o segundo – denominado "*serving layer*", responsável por realizar análises sobre esses dados, disponibilizando-os sob mais de um prisma. Em contrapartida, coexiste a este procedimento a camada chamada "*speed layer*", que cria análises em tempo real e ambas as camadas podem fornecer consultas simultâneas. A caraterística mor da arquitetura lamba está na ambiguidade e importância de suas camadas. Elas se completam.

Figura 12: Arquitetura Lambda



Fonte: Bar (2016)

2.2.8 Tecnologias e Ferramentas de *Big Data* e *NoSQL* Disponíveis

Visualizam-se as tecnologias pilares do *Big Data* e *NoSQL* sob duas perspectivas: de infraestrutura e "*Analytics"*, destacando como exemplo os bancos de dados *NoSQL* para esta última e o Hadoop associado ao *MapReduce*, para a primeira citada.

Um ecossistema de *Big Data* é composto por tecnologias de: ingestão, armazenamento, processamento, mensagens, bibliotecas de referência e biblioteca de aprendizado de máquina.

Cabe a seguir, mesmo que resumidamente, citar algumas tecnologias – na forma de ferramentas de *softwares*/sistemas, consideradas de maior importância no atual contexto do *Big Data*, bem como no tratamento de dados não estruturados. Salientando que, no nível existencial no mercado, inúmeras outras ferramentas e aplicações estão aptas a tratar dados não relacionais. A intenção aqui, por ora, é proporcionar uma visão de amplitude sobre o tema, e não apenas de escolha desta ou daquela solução para implementação de qualquer projeto.

a) **Flume -** é um sistema distribuído de forma eficiente para coletar grandes volumes de serviço de dados de logs, de forma eficiente e confiável. Usa um modelo simples que permite a aplicação analítica *online* (Apache, 2017). Segue uma arquitetura flexível, tolerante a falhas e, segundo Hoffman (2013), seus três principais componentes são: fontes, canal de comunicação e *sink*, Após passar por estes, os arquivos podem então ser distribuídos em formato não relacional, em um sistema de arquivos distribuídos;

b) **Sqoop -** é uma ferramenta que permite a transferência de dados entre bancos relacionais e a plataforma Hadoop (Apache, 2017). Foi criada com o propósito de transferir eficientemente grandes pacotes de dados entre Hadoop HDFS e sistemas de bancos de dados relacionais como MySQL[54], Oracle ou PostgreSQL[55];

c) **Kafka -** é um software de mensagens de logs distribuido, adequado para uso em modos *offline* e *online*. Foi projetado para permitir que um único *cluster* sirva como a espinha dorsal de dados, podendo ser expandido elasticamente;

d) **RabbitMQ -** é um sistema *open source* servidore de mensagens que, suporta múltiplas opções de configuração, clusterização e alta disponibilidade (Rabbitmq, 2017);

e) **Cassandra -** Lakshman e Malik (2010) definem Cassandra como um sistema de armazenamento distribuído para o gerenciamento de grandes volumes de dados por meio de inúmeros *clusters*, com alta disponibilidade, consistência e escalabilidade, além de baixo custo de implementação e administração. Sua tolerância a falhas – por não

[54] https://www.mysql.com
[55] https://www.postgresql.org

vincular-se a um único ponto de falha possível, aumenta a confiabilidade do sistema, pois os dados são replicados em diversos nós do *cluster*;

f) **Hadoop HDFS -** É um sistema de arquivos distribuído e o objeto mais importante do Hadoop. Não é uma ferramenta ou biblioteca, mas é o cerne da plataforma Hadoop. Oferece alto desempenho e suporta arquivos grandes. É tolerante a falhas;

g) **MongoDB -** MongoDB é um sistema de tratamento de dados orientado a documentos, para consultas e agregações complexas de dados. Suporta replicação e sharding e tem se tornado muito popular dentre os de suas características;

h) **HBase -** É um banco de dados orientado a colunas e foi construído para fornecer pedidos com baixa latência sob Hadoop HDFS;

i) **Elastic -** Ferramenta para consultas de texto de modo distribuído. Possui tempo de busca rápido por trabalhar com índices;

j) **Yarn –** *Yet Another Resource Negotiator* é um *framework* que representa a próxima geração do Hadoop. É responável pelo controle dos recursos do *cluster*;

k) **Tableau -** É uma plataforma de visualização e análise de dados proprietária (possui versões gratuitas);

l) **Mesos -** Mesos é um sistema distribuído para gestão dos recursos de um *cluster* desenvolvido pelo sistema Universidade de Berkeley. Pode referenciar até dez mil nós;

m) **Hadoop *MapReduce* -** É a implementação do Hadoop *MapReduce*. Projetado para trabalhar em HDFS e com processamento paralelo sobre o paradigma de mapear e reduzir. Como faz uso intenso do disco, decrementa o desempenho do sistema, ainda assim é uma das estruturas para tratamento de dados mais importantes já surgidas;

n) **Spark -** Um sistema *open source*, com uma estrutura de processamento paralelo para computação em *cluster* objetivando a análise de dados na forma mais rápida possível. Desenvolvido com foco em velocidade e facilidade de uso, visando sempre uma análise sofisticada. Ele trabalha intensamente na memória tornando-se até cem vezes mais rápido do que o Hadoop *MapReduce*;

o) **Storm** - O Apache Storm é um *framework* para desenvolvimento de aplicações de processamento de fluxos de dados, de forma distribuída. Impulsionado pelo Twitter, possui *design* voltado para processar eventos de forma extremamente rápida (mais de um milhão de registros por segundo/nó);

p) **Flink -** É um *framework* recente, para o processamento de *streaming* com alto desempenho e baixa latência;

q) **Spark Mllib -** Spark MLlib é uma estrutura que inclui algoritmos de aprendizado de máquina aproveitando os benefícios da computação distribuída e trabalho intensivo de memória. Ele inclui algoritmos de classificação, regressão e de agrupamento;

r) **Hive -** O Hive[56] começou como um subprojeto do projeto Hadoop. Fornece um conjunto de ferramentas para ler, escrever e gerenciar dados do Hadoop através de uma sintaxe SQL-like;

s) **Pig -** Pig[57] é uma plataforma para análise de dados que consiste de uma linguagem de alto nível para expressar uma análise de dados e a infraestrutura para execução desta linguagem;

t) **Spark SQL -** É um módulo incluído no Spark para trabalhar com dados estruturados usando a sintaxe SQL, mas aproveitando execução no núcleo do Spark;

u) **R -** Ambiente para análise estatística;

v) **D3 -** É uma biblioteca JavaScript para visualização de dados;

x) **Mahout -** É um projeto da *Apache Software Foundation* para produzir implementações livres de algoritmos de aprendizado de máquina escaláveis;

y) **Talend -** Sob os padrões abertos da *General Public License* (GPL) se apresenta como uma ferramenta *Open source* para ETL (Extração, Transformação e Carga) e integração de dados estável, sólida e inovadora, com uma interface gráfica baseada em componentes;

z) **Knime -** É uma boa ferramenta de mineração de dados. Plataforma de código aberto de nível empresarial, de fácil implementação;

aa) **Pentaho -** É uma plataforma *open source* com uma arquitetura poderosa para criação de soluções para as necessidades de BI (*Business Intelligence* - Inteligência de Negócios). A ferramenta foi desenvolvida em Java, já fora considerado um dos melhores *softwares* para inteligência empresarial. Suporta ETL (*Extraction, Transformation and Load*), relatórios, *workflow*, OLAP (*Online Analytical Processing*) e mineração de dados com *Data-mining* e *Big Data*;

ab) **MySQL -** Produzido pela Oracle®[58], é um sistema de banco de dados de código aberto com comprovado desempenho, confiabilidade e facilidade de uso, principalmente quando opção de banco de dados para aplicativos baseados na Web;

[56] http://hive.apache.org
[57] http://pig.apache.org
[58] https://www.oracle.com/br/

ac) **Redis -** Redis é uma ferramenta de estrutura de dados *open source* para armazenamento em memória. Usado como banco de dados, cache e corretor de mensagens. Ele suporta estruturas de dados como seqüências de caracteres, *hashes*, listas, conjuntos, *logs*, entre outros;

ad) **Qlik View Sense Desktop** - Ferramenta *open source* destinada à visualização de dados, que permite criar relatórios e *dashboards* interativos com tabelas e gráficos;

ae) **Weka** - O software Weka (*Waikato Environment for Knowledge Analysis*), licenciado ao abrigo da *General Public License,* é uma ferramenta que aplica conceitos de *Machine Learning* (aprendizado de máquina) para análise computacional e estatística de dados através da mineração de dados, buscando gerar soluções, baseadas nas induções que estes mesmos dados geram; e

af) **Neo4j -** Ferramenta *open source*, é um banco de dados *NoSQL* orientado a grafos indicado para trabalhar com uma grande quantidade de dados (*Big Data*). Possui suporte ACID e para *Clustering*, além de agregar velocidade por trabalhar com os dados em memória. Considerada ótima ferramenta para se trabalhar com dados de redes sociais.

Subsequente a estas descrições (coletadas dos sítios oficiais de cada um dos citados), a ilustração da Figura 13 sugere através de uma rápida concepção visual, uma maior identificação acerca das ferramentas e solução, quanto ao campo de atuação. A escolha de algumas ferramentas, agregados, podem vir a formar uma solução prática para o tratamento de dados *NoSQL*, pelo qual foram descritos aqui.

Figura 13: Ferramentas de *Big Data* em conformidade com a área de atuação

Fonte: Elaborada pelo autor (2017)

Estima-se que o mercado global de *Big Data* estará valendo 88 bilhões de dólares até 2021[59]. Com esse prospecto, não basta para as instituições apenas ter interesse em usar *Big Data*, analisar dados em grande quantidade ou de estruturas variadas, é preciso ser capaz de traduzir os dados. O que fazer com o volume e a variedade de dados? O que eles significam? Como analisá-los em tempo real? Que, conhecimento ou melhorias isso pode trazer para uma gestão institucional? Estas e muitas outras incógnitas buscam respostas no cientista de dados.
O que requer para si, conhecimento e domínio apropriado sobre tratamento de dados, visão de negócios e habilidades de programação (banco de dados), além, obviamente, entender e dominar as variadas plataformas de *Big Data*. Deve ser capaz de traduzir dados em informação gerencial e possuir para isso capacidade analítica para identificar informações de valor com base nas ferramentas de *Big Data*. Sendo assim, conhecer ferramentas de hardware e software para coleta e análises de dado, como o Hadoop (de código aberto) é igualmente uma atividade inerente.

[59] http://www.cienciaedados.com/big-data-como-servico/

2.2.9 *NoSQL* no Contexto O*pen Source* e Licenças Livres

Para utilizarmos todos esses dados gerados no contexto *NoSQL*, e extrair deles informações, é necessário o uso ferramentas especiais para armazenamento, extração, análise, formatação e visualização. No nível de usuário o mercado oferece soluções pagas que prometem satisfazer a contento (na visão do fornecedor) estas premissas. Contudo a comunidade *open source* também esta inserida nessa proposição, através de projetos tecnicamente eficazes, como por exemplo, as fornecidas pela Apache Foundation[60].

A maior vantagem das aplicações *free*[61]*/open-source* é a possibilidade de minimizar, ou mesmo eliminar, o TCO (*Total Cost of Ownership* – Custo Total de Propriedade) (YANG, 2016). O movimento *Open Source* oferece soluções robustas e profissionais na área de banco de dados para quase todos os tipos de aplicações e problemas.

Considerando ainda, que quando o assunto é cortar custos, os responsáveis pelas soluções de TI nas instituições podem recorrer à adoção de um modelo *open source*, sem taxa de licenciamento. Contudo cabe lembrar que isso requer procedimentos e incentivos em treinamentos e manutenção do produto, responsabilidade pelas instalações e atualizações necessárias.

Partindo desse cenário, cabe visualizar no Quadro 1, uma breve relação de algumas ferramentas *free/open source* passíveis de uso no embasamento funcional de uma arquitetura de referência, objetivo deste trabalho:

Quadro 1: Ferramentas *free/open source* para tratamento de dados

Bases de dados	**Entrada de dados**	**Processamento Distribuído**	**Análise de dados**	**Visualização de dados**
MongoDB Cassandra Neo4j Hbase MySQL Spark SQL	Kafka Sqoop Talend Flume	Apache Hadoop/HDFS Hive Impala Pig Redis Pentaho	R Pentaho Elastic Mahout Knime Weka	Qlik View Tableau

[60] www.apache.org/foundation/
[61] Sem custos de licenciamento ou uso.

		Knime Yarn Spark Storm Redis		

Fonte: Elaborado pelo autor (2017)

2.3 Trabalhos relacionados

A demonstração em etapas de arquiteturas de referência, bem como o uso de software livre para coleta e tratamento de dados de *Big Data* e dados não estruturados, estão presentes em arquiteturas de outras instituições ou empresas.

A pesquisa intitulada "*Towards a Big Data Reference Architecture*" (MAIER, 2013), consiste numa abordagem sobre tecnologias atuais, que podem ser usadas para implementar uma arquitetura de referência, construída a partir de empresas tradicionais. Mostra que a implementação de componentes com tecnologias como o ecossistema Apache Hadoop e os chamados bancos de dados "*NoSQL*" podem compor arquitetura de referência. A arquitetura de referência proposta e uma pesquisa do estado da arte pode segundo a pesquisa, orientar os designers na criação de sistemas de tratamento de dados não estruturados para tomadas de decisões.

A citada pesquisa de Maier (2013) corrobora positivamente para que a proposta da pesquisa que dá origem a esta dissertação, receba subsídios que possibilite ao autor contruir ordenadamente e faticamente uma arquitetura de referência, que seja compatível com a atualidade no que se refere a uma possível implementação, visto que aborda pontualmente os dados *NoSQL*. Contudo cabe ressalvar que esta, mesmo plausível de aceitação genericamente, não faz menção a realidade proposta pela pesquisa deste autor, quando não se refere à gestão de instituições educacionais. Uma realidade notadamente diferenciada de empresas de grande porte como as citadas por Maier (2013).

Já a pesquisa de Pääkkönen e Pakkala (2016), "*Reference Architecture and Classification of Technologies, Products and Services for Big Data Systems*" aborda casos de uso em *Big Data* por grandes empresas como Facebook, LinkedIn ou Netflix, mostrando uma arquitetura de referência independente de tecnologia para sistemas de dados, que se baseia na análise de arquiteturas de implementação publicadas.

Com relação à pesquisa de Pääkkönen e Pakkala (2016), salienta-se que a mesma possui um foco mais direcionado ao tratamento de grandes volumes de dados, o que em relação à proposta desta dissertação, não dá a devida atenção aos dados estruturados, tampouco ao aspecto “instituição pública”.

Ainda, a dissertação de mestrado de Costa (2015) ”BASIS: Uma Arquitetura de *Big Data* para *Smart Cities*” se faz relevante quando este propõe uma arquitetura de *Big Data* para *Smart Cities*, realizando um enquadramento conceitual e tecnológico, voltados ao estudo das principais abordagens existentes entre o conjunto de publicações científicas e as tecnologias de *Big Data* que podem integrar componentes tecnológicos de uma arquitetura em várias camadas de abstração, desde a mais conceitual até a mais tecnológica.

Por fim, mencionando também a pesquisa de Costa (2015), frisa-se, embora abordando ferramentas *open source* para tratamento de dados e também dados *NoSQL*, além de muito útil como fomento para o aprofundamento acerca das ferramentas disponíveis no mercado, a preocupação é voltada a IoT. O termo IoT é coadjuvante a pesquisa proposta neste compendio, porém não o foco principal.

Assim sendo, sugere-se que os trabalhos oportunamente aqui citados, merecem atenção do autor no sentido de instigar a partir destes, complementos através de pesquisa adicional, no que tange ao tratamento de dados não estruturados, a construção de arquitetura específica, com voltas a realidade das instituições de ensino, o que ao seu fim trará para a comunidade acadêmica e ao público que se vincula a estas instituições, um novo material de referência para uso na efetivação de sistemas de tratamento de dados não estruturados.

3 ARQUITETURA DE REFERÊNCIA PARA TRATAMENTO DE DADOS NÃO ESTRUTURADOS

Nos capítulos anteriores foram apresentadas algumas tecnologias importantes para a implementação de aplicações que se encaixam no contexto de dados não estruturados. O propósito do capítulo dois consistiu em trazer à tona o conhecimento necessário para que seja possível analisar e projetar soluções baseadas em diferentes cenários que envolvem o processamento de dados não estruturados e em grandes volumes. Também, foram analisadas funcionalidades, fluxos de dados e armazenamento de dados de arquiteturas dispostos na literatura. O próximo passo deste trabalho é propor um modelo de arquitetura de referência. Ante ao exposto, segue uma proposição de solução para o problema que esta dissertação expõe na sua etapa inicial. Tal procedimento se dará por um processo que consiste na decisão de um tipo para a arquitetura de referência, seleção de uma estratégia de projeto, aquisição empírica de dados, construção da arquitetura de referência e avaliação.

As redes sociais deixaram de ser utilizadas apenas para lazer e atualmente são usadas pelos cidadãos para expressar opiniões dos mais variados assuntos, principalmente quando se referem às questões sociais ou de comportamento. Instituições de ensino podem fazer uso destes dados, analisá-los e aplica-los na qualificação da gestão educacional, proposição de cursos, métodos de ensino, acompanhamento de egressos ou ainda obter um panorama de alocação no mercado de trabalho, entre tantas outras ações. Além das redes sociais, outros veículos geradores de dados na Internet, podem corroborar com este papel. Processar estas informações não é uma tarefa trivial, pois os dados, além de diferentes quanto a sua estrutura, são gerados em ordem de elevada quantidade.

Será apresentado um modelo de arquitetura com a intenção de contemplar a situação descrita anteriormente. A quantidade de dados gerados pelos veículos citados exige também uma visão voltada para o contexto *Big Data*, pois como visto em capítulos anteriores, bancos de dados relacionais e outras tecnologias tradicionais não são adequadas para este tipo de problema. Portanto este estudo objetiva servir de veículo para futuras implementações, que possam tratar dados não estruturados, nos Institutos Federais de Educação Ciência e Tecnologia.

A construção desta arquitetura de referência adota um método empírico e se baseia na análise de casos de uso de arquiteturas para tratamento de dados publicados em trabalhos

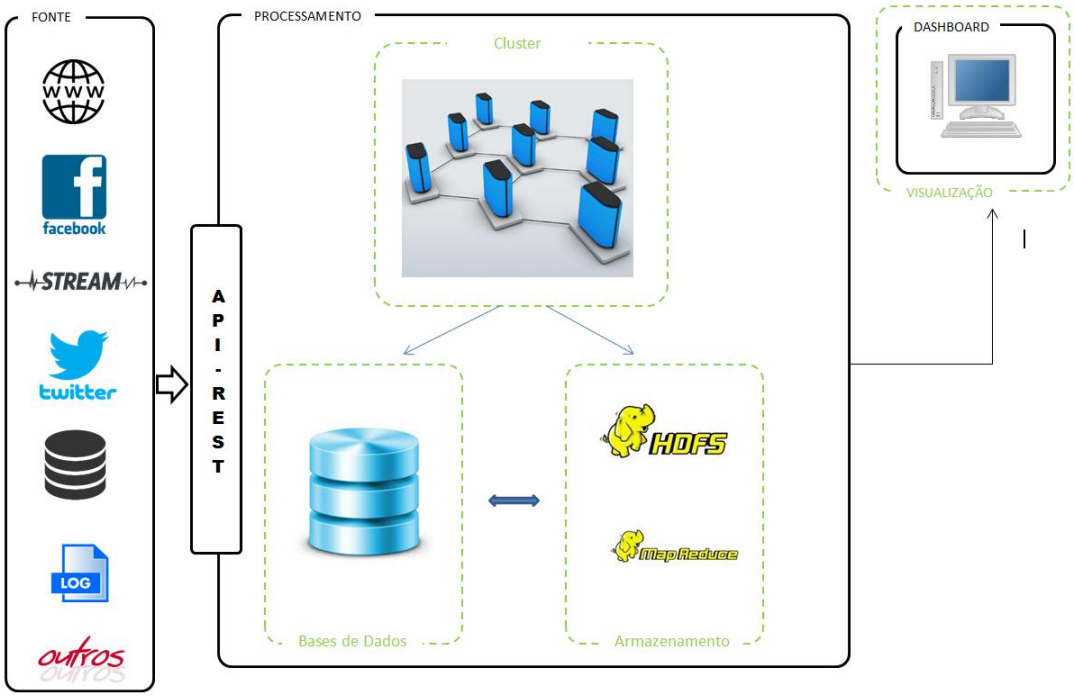
sobre o uso de tecnologias e arquiteturas heterogêneas, que se concentraram principalmente na descrição de arquiteturas de contribuições individuais como Facebook (MEIER, 2013) ou LinkedIn (SUMBALY, 2013), já citados no capítulo dois. Agrega-se a isso a condução de um estudo sobre ferramentas e técnicas de tratamento de dados. Este método abstrato de concepção e a consequente arquitetura resultante destinam-se a facilitar a criação de uma arquitetura de *design* mais elaborada e a seleção de tecnologias ou soluções comerciais, para se construir um sistema para o tratamento de dados considerados complexos. Assim sendo, a concepção da arquitetura de referência para tratamento de dados não estruturados é apresentada, construída indutivamente com base nos casos da literatura.

A arquitetura de referência se propõe a ser útil das seguintes formas: deve facilitar a criação de outra(s) arquitetura(s) concreta(s); aumentar a compreensão sobre o tema; prover uma imagem genérica sobre o tratamento de dados de diferentes formatos; conter as funcionalidades típicas e fluxos de dados do sistema que se propõe futuramente construir. A referida arquitetura, em resumo, como se mostra na Figura 14, deve prover a execução das seguintes tarefas:

a) Extrair dados relacionados ao contexto educacional (mas não somente estes) postados por usuários de redes sociais;
b) Extrair dados relacionados ao contexto educacional (mas não somente estes) encontrados na Internet;
c) Extrair dados de sistemas utilizados pela instituição;
d) Prover escalabilidade linear para a quantidade de registros que são armazenados;
e) Analisar e classificar dados de conteúdo (função *analytic* – de acordo com os objetivos específicos da instituição); e
f) Exibir para o usuário final um *Dashboard*[62] ou aplicativos web para visualização dos resultados obtidos após a análise.

[62] Apresentação visual das informações importantes, consolidadas e ajustadas em uma tela para fácil acompanhamento.

Figura 14: Resumo do aspecto conceitual da arquitetura de referência proposta



Fonte: Elaborada pelo autor (2017)

O desenvolvimento da arquitetura de referência será explicado recorrendo a dois modelos:

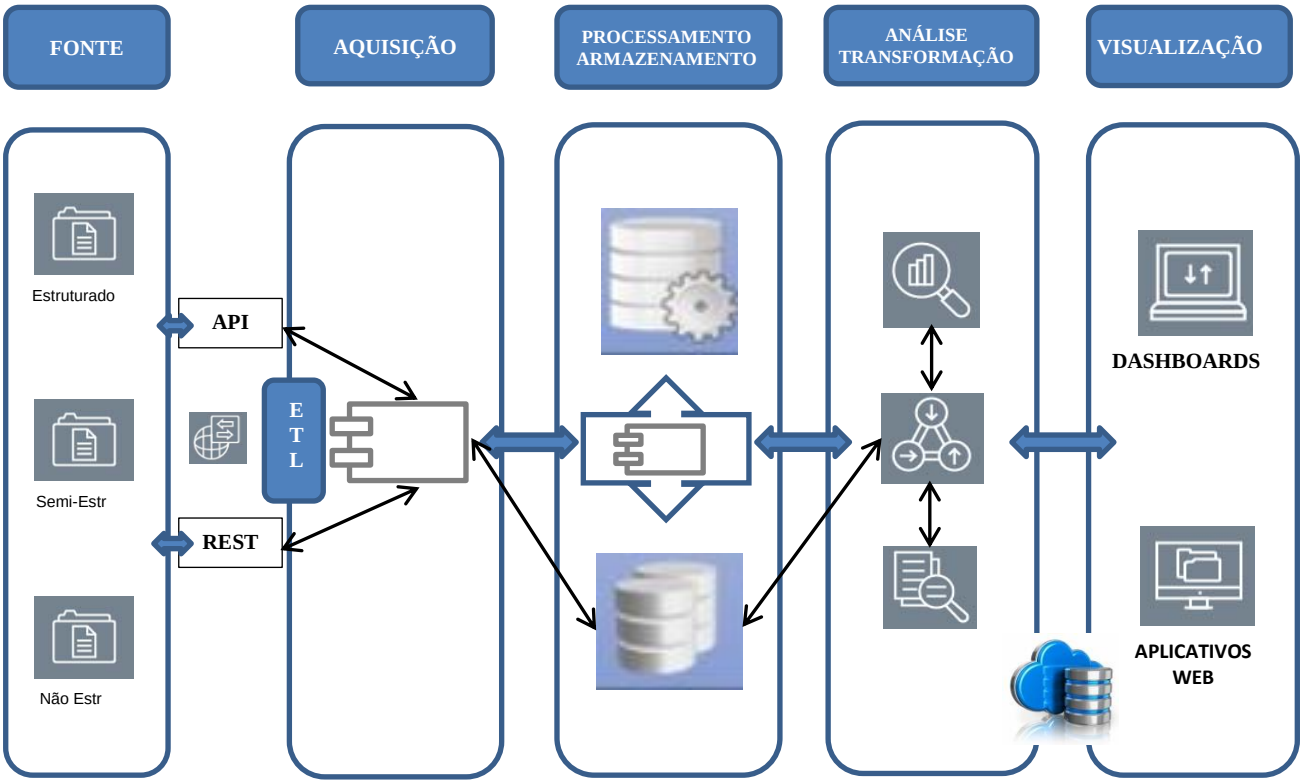
a) **Arquitetura Conceitual** - descreve os níveis que constituem a arquitetura e a explicação das atividades que são realizadas em cada um dos níveis. Esta arquitetura é apresentada na Seção 3.1.

b) **Arquitetura Funcional** - descreve uma solução tecnológica, por meio da instanciação de tecnologias para cada um dos níveis identificados na arquitetura conceitual. Realça-se que poderão existir outras soluções tecnológicas distintas da apresentada. Esta arquitetura é apresentada no capítulo quatro.

3.1 Arquitetura Conceitual

A arquitetura proposta é composta por cinco níveis. Cada nível dá suporte a um conjunto de atividades a ele associadas. Estas vão desde a coleta dos dados até à disponibilização de informações ao usuário final, obtidos pelas análises realizadas nos dados. Os níveis que subdividem e constituem a arquitetura, conforme visualizado na Figura 15, são: Fonte, Aquisição, Processamento/Armazenamento, Análise/Transformação e Visualização.

Figura 15: Arquitetura de referência - alto grau de abstração



Fonte: Elaborado pelo autor (2017)

3.1.1 Fonte

As fontes de dados, representadas em ambas as arquiteturas procuram identificar as diferentes origens e tipos de dados que podem ser utilizados, nomeadamente dados provenientes de redes sociais, arquivos de texto, vídeos, entre outros, além de dados estruturados oriundos de sistemas locais, devido ao fato de, nas organizações de pequena e média dimensão, existirem uma enorme quantidade de dados com origem nestas fontes.

Essas fontes podem conter o conteúdo de um banco de dados relacional, que é estruturado com base em um banco de dados qualquer, informações armazenadas de sistemas proprietários da instituição, dados não estruturados, desassociados a um modelo de dados, como por exemplo, um conteúdo de página Web ou imagens e dados semiestruturados, ditos irregulares ou de estrutura parcial como documentos XML e JSON.

3.1.2 Aquisição

A aquisição corresponde à entrada de dados no sistema. A extração destes se dará através de APIs disponibilizadas. Por exemplo: a extração dados e publicações de redes sociais deverá ocorrer através das APIs disponibilizadas pelas redes sociais. A mesma premissa indicará a aquisição de conteúdo web. Estas ferramentas possibilitam capturar postagens públicas através de serviços REST.

REST (*Representational State Transfer*) pode ser descrita como um estilo arquitetural com restrições aplicadas a componentes e elementos de dados dentro de um sistema de web hipermídia distribuído. Definido oficialmente pela W3C[63], o REST não se preocupa com detalhes do protocolo ou componente, mas sim com sua interação com outros componentes. É um conjunto de princípios que definem como *Web Standards* (HTTP[64] e URIs[65]) devem ser usados. Ao se aderir a princípios REST durante o processo de composição da aplicação, compor-se-á um sistema que explora a arquitetura da Web em benefício do próprio sistema.

Este nível da arquitetura de referência está responsável também pelo processo denominado *Extract Transform and Load* (ETL) que compreende as ações relativas à extração dos dados, sejam eles estruturados, semiestruturados ou não estruturados, de fontes variadas e distintas, transformação e limpeza (correções) dos mesmos, assegurando assim que posteriormente estes dados possam ser levados para o processo (área) de armazenamento (CHAUDHURI et al., 2011). As tarefas que esta etapa deve executar são: limpeza dos dados; detecção de erros; extração de dados para posterior análise; armazenamento temporário em uma base de dados. Ainda, o armazenamento dos dados que foram tratados com a utilização do processo ETL e que depois serão utilizados para o tratamento analítico deve se constituir como um repositório capaz de armazenar diversos tipos e origens de dados.

É importante complementar que um fluxo ETL pode ser visto como um pipeline de dados. Os dados entram numa extremidade do processo numa forma saindo numa outra diferente e desejada. A complexidade dos requisitos de coleta e transformação irá depender dos objetivos do sistema e poderá ter até inúmeros estágios, conectando uma ou várias fontes de dados ou ainda executando em um ou vários servidores.

Quando os dados são extraídos, eles podem ser armazenados temporariamente (base de dados TEMP) ou transferidos e carregados em outra base de armazenamento que pode ser

[63] The World Wide Web Consortium
[64] Hiper Text Transfer Protocol.
[65] Uniform Resource Identifier.

chamada de “Dados Crus”, exclusiva para dados não processados, conforme sugerem Pääkkönen e Pakkala (2014). O mesmo procedimento pode, a critério das funcionalidades pretendidas pelo sistema, ser atribuido aos dados de transmissão em fluxo contínuo. Um processo de compressão dos dados extraídos, pode também melhorar a eficiência dos processos de tranferência e carga. Os dados armazenandos em “Dados Crus” podem ser limpos ou combinados e salvos em um novo armazenamento ou enviados diretamente para a etapa de análise. A limpeza e combinação referem-se à melhoria da qualidade dos dados brutos não processados. “Dados Prontos” podem ser replicados entre os armazenamentos de dados. A extração de informações refere-se ao armazenamento de dados brutos em um formato estruturado. O repositório denominado “Dados Prontos” é destinado ao armazenamento de dados processados e limpos.

3.1.3 Processamento/Armazenamento

Seguindo as premissas de Klein et al. (2016), em se tratando de uma arquitetura de referência, em linhas gerais, o módulo de “processamento” deve focar sua responsabilidade sobre a execução eficiente, escalável e confiável das etapas da arquitetura. Suas funções são: disseminar os dados por toda arquitetura, implantar e gerenciar os mecanismos para dar condições de atender os requisitos que o sistema se propõe satisfazer, implantar e gerenciar a infraestrutura de distribuição dos dados entre os *clusters*, prover escalabilidade, obtida através da criação de novos canais de dados quando necessário, efetuando isso na forma de um processamento distribuído e paralelo. Deve também configurar e combinar os outros módulos de ações sobre os dados, integrando atividades em uma aplicação coesa.

Em relação ao processamento distribuído, deve permitir que os canais de dados sejam distribuídos aos diferentes hosts e manipular o armazenamento de dados entre todas as máquinas do *cluster*, utilizando um sistema de arquivos distribuídos e com múltiplas réplicas. Sistemas de arquivos distribuídos são necessários, uma vez que os dados se tornem grandes demais para serem armazenados em apenas uma máquina. O sistema deve prover a leitura de dados do *cluster*, realizar as operações pertinentes e escrever os resultados delas em um ambiente temporário, realizar a operação subsequente e reescrever no *cluster* os resultados.

Possui a responsabilidade de armazenar os dados oriundos das diversas etapas e diferentes níveis da arquitetura. As informações, após serem armazenadas e analisadas em um formato de *cluster*, terão seus resultados gerados através da etapa de análise e serão a partir desta, disponibilizados através de *dashboards* (aplicativos web).

Considera-se que o módulo processamento possui escopo amplo, atuando também sobre a etapa que de ETL.

3.1.4 Análise/Transformação

A etapa “análise” está preocupada com a obtenção eficiente do conhecimento a partir dos dados, normalmente trabalhando com múltiplos conjuntos de dados com diferentes características.

Esta etapa ocorre desde a extração, podendo ser feitas análises mais profundas de acordo com solicitações do usuário (com uso de Haddop, por exemplo). Este nível da arquitetura é responsável por realizar as análises aos dados e disponibilizar os resultados para um nível de usuário final. O uso de *cloud computing* nesse processo é indicado para permitir que os dados sejam guardados, acessados e utilizados em qualquer local. Um conjunto de procedimentos analíticos que podem ser realizados na forma de análises que utilizam algoritmos de *Data Mining* ou de *Predictive Analysis*[66], ou ainda análises realizadas através de *querys adhoc*[67], ou seja, não estruturadas na base de dados.

Os resultados da análise podem ser armazenados novamente nos chamados dados prontos ou ainda em um armazenamento de resultados de análise separado. Estes por sua vez podem ser divididos em resultados de tempo real ou de armazenamento. A análise de tempo real pode ser um sinônimo para análise de fluxo. A análise de fluxo refere-se à análise de dados de transmissão em fluxo contínuo. Os resultados da análise de dados também podem ser denominados como uma base de dados que serve de interface e aplicações de visualização, como por exemplo, um atendimento às consultas OLAP (*Online Analytical Processing*).

3.1.5 Visualização

A etapa de “visualização” está preocupada com a apresentação dos dados processados em um formato que expresse conhecimento. Ela fornece uma "interface humana" para estas informações em relação ao usuário final. A visualização dos dados envolve o uso e a prática de técnicas estatísticas adequadas para responder às solicitações que a instituição terá por opção solicitar.

---

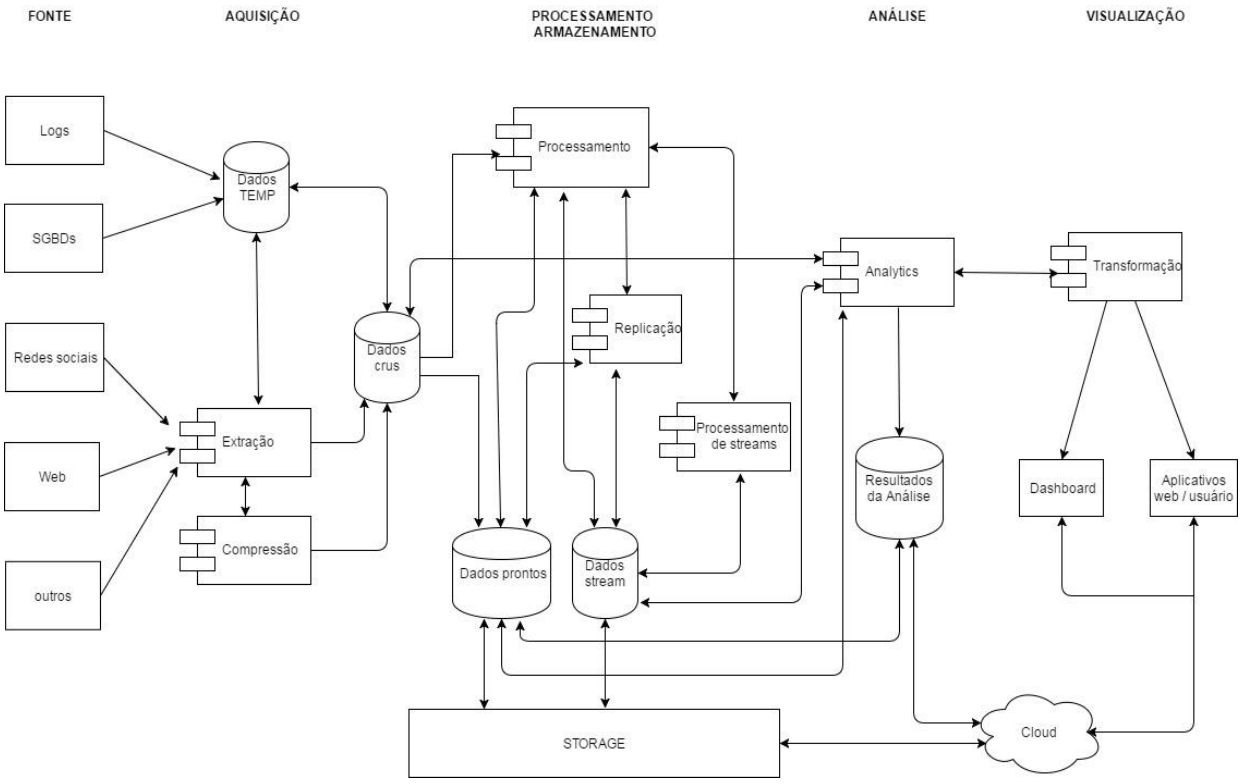
[66] Analisar um cenário específico e traçar possíveis tendências ou mudanças.
[67] Uma consulta que não pode ser determinada antes do momento em que a consulta é emitida.

Algumas técnicas de visualização podem gerar informações em cache para acesso posterior, como um relatório ou um gráfico, ou ainda incluir geração sob demanda, por meio de uma interface interativa, como resultados de uma pesquisa, por exemplo. Segundo Klein et al. (2016) podem incluir a capacidade de criar, confirmar ou corrigir, atualizando os dados. O usuário final deve poder especificar tarefas ou consultas interativamente na interface do usuário, que por sua vez são então mapeadas e levadas para o respectivo processamento.

As ferramentas de visualização a serem implementadas devem também permitir ao usuário publicar relatórios acessíveis de plataformas, como computadores ou smartphones. Os resultados das análises são frequentemente fornecidos a outras aplicações. Isso pode incluir interfaces técnicas e APIs para acessar os dados e os resultados, sendo que esta também deve ser uma ação plausível para a etapa de visualização.

Dadas estas considerações sobre como deve se portar a arquitetura de referência proposta, é conveniente expressar tais funções no formato de um diagrama mais detalhado. Esse pressuposto é represntado pela Figura 16:

Figura 16: Arquitetura de referência



Fonte: Elaborada pelo autor (2017)

3.2 Considerações finais sobre este capítulo

O conjunto dos processos das etapas que compõem esta arquitetura de referência, constitu-se exemplo de utilização para uma proposta genérica de tratamento de dados. Esta arquitetura mesmo genérica, possibilita que o tratamento de dados não estruturados seja abordado para com futuras proposições de implementação de um sistema, ou ainda a composição de uma arquitetura mais explicitamente refinada para este fim. Pode-se imaginar por meio dela, a utilização de dados de redes sociais para composição de decições sobre a área da educação, aprimorando a escolha de segmentos educacionais ou ainda compondo *insigths* para melhoria da gestão pública no âmbito dos institutos federais de educação.

A partir destas concepções criadas, pode-se também propor um nível menor de abstração, incluindo sobre esta arquitetura, exemplos de ferramentas disponíves, a critério de dar aceitação para a mesma, ou entao para vislumbrar um aspecto mais funcional. Mesmo que ainda carecendo de maiores detalhes intrínsecos a uma implementação prática, propõe-se no capítulo a seguir uma alternativa de melhoria, através da agregação de ferramentas disponíveis sobre a forma de licenças livres e/ou *open source*, para com inerência aos Institutos Federais de Educação, tomando por base dados colhidos junto ao Intituto Federal Farroupilha, para com este propósito específico do tratamento de dados. Por fim a partir deste passo, sugere-se uma forma de avaliação da arquitetura de referência, por meio da opinião de especialistas na área.

4 UMA PROPOSTA DE UTILIZAÇÃO DE ARQUITETURA PARA TRATAMENTO DE DADOS NÃO ESTRUTURADOS NO AMBIENTE DOS INSTITUTOS FEDERAIS DE EDUCAÇÃO

Neste capítulo é apresentada e detalhada uma proposta na tentativa de prover uma solução para o problema inicial. Visando tal êxito, faz-se necessário previamente inserir o IFFar como representativo dos demais Institutos Federais de Educação, Ciência e Tecnologia. São descritos os requisitos relacionados aos Institutos Federais de Educação, as prerrogativas acerca da importância de se utilizar preferencialmente tecnologias embasadas no formato de licenças *open source* e por fim a apresentação da arquitetura elaborada, bem como detalhamento dos seus componentes. Perfazendo assim a proposta do autor de construir um modelo de referência para aplicações que se enquadrem no trato de dados não estruturados, e em concordância aos objetivos aqui propostos inicialmente e que nortearam os passos para se obter tal resultado.

O Instituto Federal de Educação, Ciência e Tecnologia Farroupilha - IF Farroupilha[68] é uma instituição cujo foco é oferta de educação profissional e tecnológica nas modalidades presencial e à distância no que tange ao nível médio, além de cursos técnicos integrados e subsequentes. No que se refere ao ensino superior, o IF Farroupilha oferece cursos de tecnologia, licenciatura e bacharelado, buscando a verticalização do ensino na instituição, observando as demandas regionais de cada campus. Quanto à pós-graduação *lato sensu*, são oferecidos anualmente cursos que vinculam a educação com as áreas de formação de cada campus. No nível de *stricto sensu* está sendo ofertado o Mestrado Profissional em Educação Profissional e Tecnológica (ProfEPT).

O IFFar tem como características institucionais a natureza jurídica de autarquia, atribuindo-lhe autonomia patrimonial e financeira, administrativa, didático-pedagógica e disciplinar (PDI[69], 2014).

Antes de adentrar especificamente no que tange a escolha de ferramentas e arquiteturas para tratamento de dados, cabe elucidar um pouco mais a caracterização dos dados quanto às instituições de ensino. Uma breve leitura sobre dados educacionais, correlacionados com dados abertos se faz pertinente, visto que o tratamento de dados não

[68] Criado pela Lei n° 11.892, de 29 de dezembro de 2008, por meio da integração do Centro Federal de Educação Tecnológica de São Vicente do Sul, de sua Unidade Descentralizada de Júlio de Castilhos, da Escola Agrotécnica Federal de Alegrete, e do acréscimo da Unidade Descentralizada de Ensino de Santo Augusto que anteriormente pertencia ao Centro Federal de Educação Tecnológica de Bento Gonçalves

[69] Plano de Desenvolvimento Institucional

estruturados pode vir a corroborar para o fomento da utilização destes em maior escala social, além do que esta proposta de dissertação visa à aplicabilidade no âmbito de uma instituição federal de ensino.

4.1 Dados Educacionais

No contexto educacional, dados de todos os tipos e formatos é matéria prima para o desenvolvimento de pesquisas e estudos de diversos níveis de relevância no melhoramento e inovação tecnológica e inerente ao cidadão. Bases de dados (conteúdos) importantes para a educação, como censos escolares, são publicados e dispostos na Internet em formatos como pdf[70], xls[71], csv[72], doc[73] e muitos outros. Dados educacionais são cruciais ao governo e ao cidadão, visto que espelham a realidade atual da educação. Torna-se óbvio, se dada à devida atenção a estes preditos, que o acesso, interpretação e manipulação destes dados podem servir de alavanca na tomada de decisão dos gestores escolares.

Os dados abertos do *E-government*, podem ser utilizados no auxílio à gestão escolar por meio de sistemas tecnológicos de tomadas de decisão, na concepção de novos artefatos tecnológicos, apenas citando alguns exemplos, ou ainda no aprimoramento de recursos educacionais.

Contudo, o desenvolvimento de soluções tecnológicas que venham a contemplar tal realidade, ainda se faz muito oneroso, até mesmo porque os dados educacionais estão predominantemente em formato não estruturado, o que praticamente impede ou inviabiliza a reutilização.

Ainda, em razão dos Institutos Federais de Educação, se constituírem como parte de um governo no sentido "gestão", cabe mencionar que a Lei nº 12.527 de 18 de novembro de 2011[74], garante ao cidadão brasileiro o acesso às informações públicas dos poderes Executivo, Legislativo e Judiciário, com o propósito de incentivar maior participação pública, maior fiscalização contra irregularidades e através de contrapartidas, melhorias na gestão.

Logo, face ao exposto, se as instituições detiverem conhecimento para propor meios capazes de tratar dados não estruturados, estarão corroborando para que as soluções

[70] *Portable Document Format*: formato de arquivo criado pela empresa Adobe Systems para que qualquer documento seja visualizado, independente de qual tenha sido o programa que o originou.
[71] *Extensible Style Language* (Linguagem de Estilo Extensível).
[72] *Comma-separated values*.
[73] Arquivo de tratamento de texto
[74] BRASIL. Lei n° 12.537 de 18 de novembro de 2011. 2014. Disponível em: <http://www.planalto.gov.br/ccivil_03/_Ato2011-2014/2011/Lei/L12527.htm>. Acesso em: 12 Abril. 2017.

tecnológicas, acima citadas, possam ser implementadas com maior eficácia e com dispêndio menor de recursos já escassos, seja por meio próprio de desenvolvimento na sua totalidade, ou ainda se fazendo usar de soluções terceirizadas em parte ou compondo o todo da solução.

4.2 Dados não estruturados de redes sociais e a importância para a educação

Conforme mencionam Moraes e Gomes (2014) em seu artigo, as redes sociais tornaram-se um fenômeno de adesão e popularidade, mostrando que, inúmeras pesquisas realizadas com base em redes sociais existentes, apontam para a revelação de que 78% de pessoas de todas as idades que acessam a Internet (no Brasil), são usuários, possuem perfis ou acessam algum tipo de rede social. Tempestivamente diz ainda o autor:

> De acordo com Lorenzo (2011) as redes sociais podem gerar novas sinergias entre os membros de uma comunidade educativa, como por exemplo, facilitar o compartilhando de informações envolvendo temas estudados em sala de aula, o estudo em grupo, a divulgação dos mais diversos conteúdos informativos, o compartilhamento de recursos (documentos, apresentações, links, vídeos) e, sobretudo, de projetos, além de fortalecer o envolvimento dos alunos e professores e criar um canal de comunicação entre eles e outras instituições de ensino.
> [...] são o habitat dos estudantes, o Facebook, por exemplo, em pesquisa realizada pela Tyntec (2013) mostrou que os brasileiros usam essa rede social em seu celular, pelo menos uma vez por dia [...] (MORAES e GOMES, 2014).

Fica evidenciada, num primeiro momento, a importância da análise de dados de redes sociais, no que tange a fortalecer os aspectos relacionados ao ensino e a aprendizagem, visto que, os dados provenientes desta análise, podem compelir à adoção de práticas pedagógicas que incluam a utilização das redes sociais, pois se utilizadas de forma adequada, pertinente e salutar ao ambiente escolar, favorecem a uma aprendizagem colaborativa entre docentes e alunos.

Ainda no que dizem respeito à gestão escolar, os dados coletados de redes sociais podem vir a compor aporte para tomadas de decisão dos gestores quanto à utilização ou não de Ambientes Virtuais de Aprendizagem (AVA), vagas a serem ofertadas, elaboração dos planos de políticas pedagógicas, incentivos de custeio para pesquisas em áreas pré-determinadas, e tantas outras ações, haja vista que as informações das redes sociais hoje forjam importante ferramenta de interação, comunicação, troca de experiências e conhecimentos, no que se refere à socialização do individuo e sua pretensão acadêmica e profissional. Werhmuller e Silveira (2012) trazem à tona uma coerente discussão sobre redes sociais e o meio acadêmico, relatando que, no momento em que a rede social usada pelo aluno, serve para compartilhar suas emoções, anseios pessoais, prospecções futuras, estas

muitas vezes não são percebidas pelo corpo docente em sala de aula. Ainda colocam os autores que:

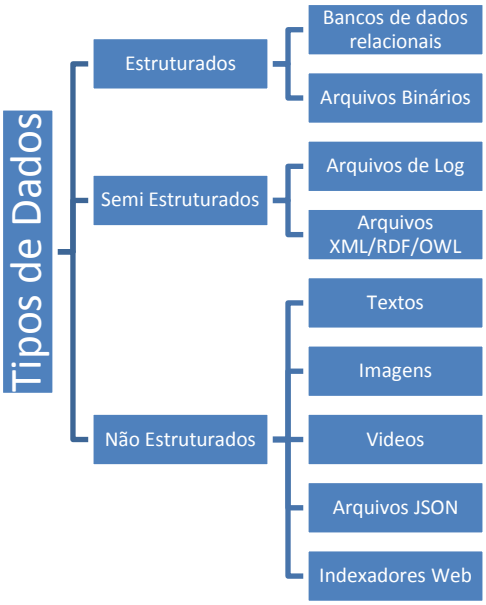
> [...] as redes sociais como ferramentas de apoio à educação centralizam em um ambiente online todas as atividades de ensino em conjunto com a troca de informações dos usuários da rede e alimentadas pelos professores e seus alunos [...] Werhmuller e Silveira (2012).

Estas colocações sugerem provas contundentes para comprovar a importância gestora da análise de dados de redes sociais.

4.3 A situação atual do IFFar quanto a dados não estruturados

A instituição de ensino, como qualquer outra, está em seu cotidiano, realizando constantes tarefas que envolvam o tratamento e absorção de dados, sejam em grandes ou pequenas quantidades, ou então em formatos distintos, sinteticamente expressos pela Figura 17.

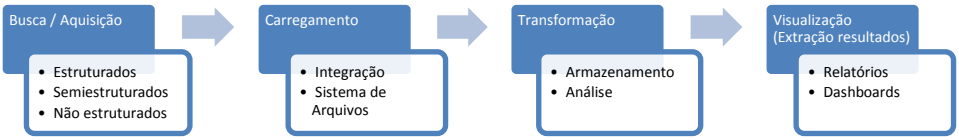
Figura 17: Tipos de dados relacionados ao cotidiano da instituição



Fonte: Elaborada pelo autor (2017)

Não cabe à instituição se abster das exigências implícitas no tratamento destes dados, tampouco desobedecer as etapas de tratamento, estas visualizadas na Figura 18. Segundo Krishnan (2013) um sistema qualquer para tratamento de dados, segue quatro etapas no que se refere a sua concepção: Busca ou Aquisição dos dados, Carregamento, Transformação e por fim Extração dos resultados. Fica assim embasado o prosseguimento de um raciocínio mais transparente sobre a realidade institucional.

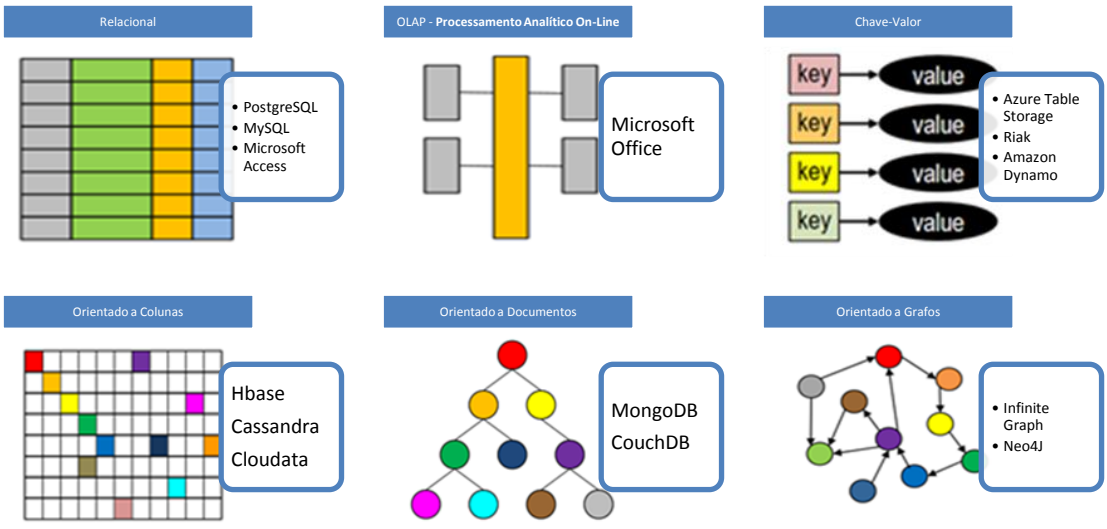
Figura 18: Etapas de um sistema de tratamento de dados



Fonte: Adaptada de Krishnan (2013)

A interação das instituições com dados de natureza e origens diversas, em especial os dados não estruturados (*NoSQL*), bem como com as ferramentas disponíveis no mercado, algumas delas vistas na Figura 19, em especial pelo fato da percepção que as empresas, órgão governamentais, entre outros, tem em relação à importância destes dados nas tomadas de decisão. A análise destes dados, e a consequente formação de informação útil, é prerrogativa de sucesso em áreas de *E-Comerce*, *Business Intelligence*, *E-Government*, e outras tantas áreas correlatas.

Figura 19: Tipos de tratamento de dados e exemplos de ferramentas



Fonte: Elaborada pelo autor (2017)

É notório que o pensamento acerca de dados *NoSQL*, como citado anteriormente, deve fazer parte também do cotidiano das instituições ligadas à educação. Além do grande número de pessoas envolvidas com a instituição, sejam alunos, professores, técnicos, empresas conveniadas, que necessitam de informação para otimizar suas tarefas e prover conhecimento científico, há também o aspecto gerencial de processos que norteiam o fator organizacional e interativo das partes.

Contudo, o tratamento de dados de grande volume e sob o formato não estruturado, ainda é sinônimo de ineditismo para a maioria das instituições de ensino. Conforme dados coletados através do método *survey*, no qual analistas e técnicos de TI foram inquiridos na forma de questões abertas e particulares, é possível mensurar o abismo que se forma entre a realidade empresarial e a realidade de uma instituição de ensino, por exemplo, no que se refere ao tema. Além disso, verifica-se claramente o despreparo, ou pra ser menos incisivo, o desconhecimento quanto ao próprio assunto *NoSQL*, e aos seus correlatos geradores destes e a ações passíveis de seu uso. Corrobora com este preocupante quadro, a inexistência de profissional qualificado, ou ainda, profissional meramente conhecedor das ferramentas que tratam dados desta natureza, na concepção de software, aplicações ou ainda projetos relacionados a ações institucionais, que poderiam usar da informação oriunda destes dados para propiciar otimização ou excelência em ações cotidianas, de pesquisa, de expansão ou de gestão.

Ao ser aplicado um simplificado questionário, acerca das incumbências dos profissionais de TI, sua formação e sua consequente relação com o tema desta pesquisa, restou explícita a necessidade em aprofundar o tema na instituição e presumidamente nas demais. O panorama do *survey* foi sistematizado através de tabelas.

Tabela 2: Nível de conhecimento acerca dos assuntos da pesquisa - Profissionais de TI/IFFar – 2016

		"Nunca ouvi falar" / Desconheço	Li a respeito	Conheço o assunto	Meu trabalho na Instituição é inerente ao assunto
Dados não Estruturados e/ou Dados *NoSQL*	n	8	7	4	1
	%	40,0%	35,0%	20,0%	5,0%
IoT	n	7	7	5	1
	%	35,0%	35,0%	25,0%	5,0%
***Big Data* Analytics**	n	7	12	1	0
	%	35,0%	60,0%	5,0%	0,0%
Open Data	n	9	10	1	0
	%	45,0%	50,0%	5,0%	0,0%
Smart Governance	n	15	5	0	0
	%	75,0%	25,0%	0,0%	0,0%
Social *Big Data*	n	10	10	0	0
	%	50,0%	50,0%	0,0%	0,0%
Data Science"	n	13	6	1	0
	%	65,0%	30,0%	5,0%	0,0%

Fonte: Elaborada pelo autor (2017)

A Tabela 2 comprova que a maioria (40%) dos profissionais de TI da instituição pesquisada, sequer teve algum contato literário com o assunto “dados não estruturados”, mesmo sendo estes profissionais em sua maioria detentores de titulação de pós-graduação, conforme demonstra a Tabela 3. Este número é similar ou maior com relação a outros termos que são ou poderiam ser associados a dados não estruturados.

Tabela 3: Cargo que ocupa x formação acadêmica - Profissionais de TI/IFFar - 2016

		Técnico de Nível Médio em T.I.	Superior	Superior em T.I.	Pós Graduado	Pós Graduado em T.I.	Total
Analista de T.I.	N	0	0	0	0	3	3
	%	0,0%	0,0%	0,0%	0,0%	100,0%	100,0%
Técnico de T.I.	N	2	2	2		4	10
	%	20,0%	20,0%	20,0%	0,0%	40,0%	100,0%
Gestor Administrativo	N	0	1	0	4	0	5
	%	0,0%	20,0%	0,0%	80,0%	0,0%	100,0%
Professor	N	0	0	0	2	0	2
	%	0,0%	0,0%	0,0%	100,0%	0,0%	100,0%

Fonte: Elaborada pelo autor (2017)

A situação exposta pela Tabela 2 e pela Tabela3 sugere certo antagonismo se associado ao que se constata pelos dados da Tabela 4. Quando questionados sobre a

importância dos dados não estruturados, principalmente se oriundos de redes sociais, os responsáveis pela tecnologia da informação consideram importantes ou relevantes para a instituição. Já para os gestores, o uso de dados não estruturados oriundos de redes sociais é extremamente importante para a gestão institucional. Percebe-se ainda que os profissionais de TI da Instituição, ainda estão alheios aos assuntos concernentes a dados não estruturados.

Tabela 4: Nível de importância dos dados não estruturados de redes sociais x Cargo que ocupa - Profissionais de TI/IFFar - 2016

	Analista de T.I.		Técnico de T.I.		Gestor Administrativo		Professor	
	N	%	n	%	N	%	N	%
Extremamente importante	0	0%	1	10%	3	60%	0	0%
Poderia ser útil de alguma forma	1	33%	2	20%	0	0%	1	50%
Irrelevante	0	0%	2	20%	0	0%	0	0%
Relevante	2	67%	5	50%	2	40%	1	50%
Total	3	100%	10	100%	5	100%	2	100%

Fonte: Elaborada pelo autor (2017)

É significativo também o fato de que a totalidade das unidades da instituição ainda não possui nenhum tipo de contato em nível de software ou aplicativo, quanto ao uso na instituição, para o tratamento de dados não estruturados. Situação quantificada pela Tabela 5.

Tabela 5: Utilização de softwares no tratamento de dados não estruturados - Profissionais de TI/IFFar - 2016

Dados não estruturados	Analista de T.I.		Técnico de T.I.		Gestor Administrativo		Professor	
	n	%	n	%	n	%	n	%
Até o momento não obtive contato com nenhum	2	67%	7	70%	4	100%	1	50%
Já utilizei (como usuário)	1	33%	2	20%	0	0%	1	50%
Já participei no processo de construção de software	0	0%	1	10%	0	0%	0	0%
Total	3	100%	10	100%	4	100%	2	100%

Fonte: Elaborada pelo autor (2017)

Ainda, a perspectiva atual remete à permanência desse quadro, visto que boa parte dos responsáveis pela tecnologia da informação na instituição, ainda desconhecem, mesmo que literariamente, ferramentas como Cassandra, HBase, Hadoop, entre outras, conforme explícito na Tabela 6.

Tabela 6: Conhecimento quanto a ferramentas específicas - Profissionais de TI/IFFar - 2016

	Leu a respeito		Utilizou na construção de um sistema	
	n	%	n	%
Hadoop	3	15%	0	0%
MapReduce	1	5%	0	0%
Cassandra	1	5%	0	0%
MongoDb	2	10%	0	0%
Kafka	1	5%	1	5%
Hbase	1	5%	1	5%

Fonte: Elaborada pelo autor (2017)

Torna-se evidente e iminente, a necessidade de transpor estas barreiras do desconhecimento sobre os dados não estruturados, e, de maneira imediata, prover condições para que ainda no presente, ou num futuro muito próximo, este quadro se reverta. É prerrogativa intrínseca e urgente, fomentar o estudo acerca do tema, direcionado e focado aos profissionais de TI. Esta pesquisa objetiva e concerne para isso de maneira objetiva e funcional. Ainda, cabe trazer à superfície técnica e funcional dos profissionais, os ferramentais disponíveis no mercado, quando se discorre sobre dados não estruturados, a forma como estes podem servir de mecanismos para a construção de sistemas e aplicativos para uso da instituição. Por fim cabe formalizar conceitos bem fundamentados, mostrando a possibilidade de se fazer realidade ações como análise de dados não estruturados, inferindo que o investimento necessário se justifica pela futura produção de conhecimento, pela tomada de decisões corretas embasadas por informações colhidas e tratadas por este intermédio, e principalmente, por propiciar progresso técnico e contínuo de pesquisa e extensão científica, pilares que denotam a realização deste trabalho.

4.4 A situação atual do IFFar quanto Aos *Data Center*s e serviços

A utilização de *clusters* é uma solução providencial quando se objetiva tratar dados de grande volume e variedade. O investimento em grandes *Data Centers* de propriedade única é uma realidade aplicada a poucas instituições. Contudo, a possibilidade de utilizar excedentes em *Data Center*s existentes, fragmentados pelas unidades institucionais, faz corroborar para que a realidade de se implantar métodos para tratamento de dados não estruturados, com as características acima mencionadas, seja um ato plausível.

Inerente a estas afirmações, com a intenção de conferir veracidade ao exposto, agregou-se alguns dados sobre o contingente de *hardware* dos *Data Center*s da instituição alvo desta pesquisa, através de visitação *in loco*, com a finalidade de servir de exemplo e sugerir que esta realidade possa estar também presente nas demais instituições federais de ensino de mesmo porte.

Apurou-se que, em praticamente todas as dependências em que habita um *Data Center*, há, senão algum equipamento excedente, pelo menos uma certa "folga" na carga de trabalho. Esse número aumentaria se fosse mensurado também o fator da carga de trabalho em relação a horários específicos. Porém a intenção não é explicitar a carga de trabalho atual dos *Data Center*s da instituição, e sim, apenas visualizar a probabilidade física de se implantar ou não uma sistema de tratamento de dados em *clusters*.

4.5 As vantagens de se usar uma solução *open source* para Os Institutos federais

Inúmeras inovações surgiram nos últimos anos no que se refere a tratamento de dados, principalmente dados em modelos não relacionais de crescimento exponencial em volume, impulsionadas pelo uso massivo da *web* por todos os segmentos sociais.

A comunidade de *software* livre e de código aberto em geral, não se absteve de participar desta evolução e diversas soluções de bancos de dados não relacionais foram assim criadas. Empresas criaram suas próprias soluções, ainda que sem conceber o termo *NoSQL*, que somente surgiria em 2009 (CHANG et al., 2006), quando a comunidade de software livre passa a desenvolver novas opções de bancos de dados, inspiradas nas ideias publicadas em artigos da época.

Já fora evidenciado que lidar com dados não estruturados não significa apenas encorajar esforços em tarefas de programação baseadas em novos paradigmas de busca, tratamento e análise, como o *MapReduce*. Outrossim, segundo Krishnan (2013), há de se fazer toda uma mudança nos requisitos de processamento de dados.

No que tange às bases de dados *NoSQL*, Leavitt (2010) informa, que sua crescente adoção não implica no desaproveitamento das bases de dados relacionais e que cada tecnologia serve a propósitos definidos.

O conjunto significativo de bases de dados, já aqui algumas vezes citados, bem como a ampla gama de ferramentas para integração com estas bases de dados, já oportunamente descritas nos itens 2.2.8 e 2.2.9 deste trabalho, enfatizam a importância delas estarem sobre título de *open source*. Contudo, ainda cabe um maior detalhamento de algumas delas (Padhy

et al., 2011), enquadradas sobre esta forma de licenças, a título de permitir uma compreensão mais detalhada daquelas que poderão atuar na aceitação da arquitetura proposta.

1 - HBase

a) Tipos de tratamento de dados: Orientado a colunas;

b) Usa HDFS;

c) Utiliza *MapReduce*;

d) Faz alterações dos dados em memória, para posterior armazenamento em disco, em intervalos periódicos;

e) Suporte de múltiplos *MasterNodes*, evitando ponto de falha único;

f) Partição e distribuição transparente;

g) As alterações dos dados são armazenadas inicialmente no final do arquivo, compactado periodicamente;

h) Para prevenir falhas, as alterações nos dados são também registradas em um log;

i) Permite acesso através de *Java Database Connectivity* (JDBC) ou *Open Database Connectivity* (ODBC), a base de dados; e

j) Desenvolvido pela Apache (Apache, 2016).

2 – Cassandra

a) Tipos de tratamento de dados: Orientado a colunas;

b) Usa HDFS;

c) Utiliza *MapReduce*;

d) As alterações aos dados são guardadas em memória para após serem armazenadas em disco;

e) Replicação assíncrona ou síncrona, dependendo do contexto;

f) Organização em Colunas, Super Colunas, Família de Colunas e keyspaces;

g) Automática detecção e recuperação de falhas;

h) Uma mesma função é desempenhada por cada nó pertencente ao *cluster*; e

i) Desenvolvedor: Apache (IBM.com, 2016; Apache, 2016; Datastax, 2013).

3 – CouchDB

a) Tipos de tratamento de dados: Orientado a documentos - *JavaScript Object Notation* (JSON);

b) Usa HDFS;

c) Utiliza *MapReduce*;

d) Dados simples, listas de valor ou outros documentos perfazem um "documento". Vários documentos constituem um "*collection*", sendo esta a forma de armazenamento do CouchDB;

e) Disponibiliza interface *Representational State Transfer* (REST)[75] com suporte a aplicações oriundas de linguagens de programação diversas;

f) Todas as alterações feitas nos documentos são guardadas em disco, gravadas no final do documento;

g) Detecta atualizações simultâneas nos documentos;

h) Alta escalabilidade por meio de replicação;

i) proporciona semântica ACID (atomicidade, consistência, isolamento e durabilidade) no nível do documento; e

j) Desenvolvedor: Apache.

4 – MongoDB[76]

a) Tipos de tratamento de dados: Orientado a documentos – Binário JSON;

b) Utiliza *MapReduce*;

c) Alto desempenho;

d) Documentos estruturados em objetos e armazenados em coleções (forma similar ao CouchDB);

e) Indexa atributos de leitura e escrita;

f) Disponibiliza *Sharding*, na distribuição de documentos nos vários nós do *cluster*;

g) Possuem índices e consultas dinâmicas;

h) Usa replicação para garantir a recuperação de falhas;

i) Atomicidade no nível de atributo e não no nível de documento; e

j) Aplicação de código, escrito na linguagem C++.

5 - Redis (*REmote DIctionary Server*)

a) Tipos de tratamento de dados: Chave-valor;

b) O valor do dado pode assumir um tipo simples, uma lista de valores e outros;

---

[75] Estilo arquitetural que consiste de um conjunto coordenado de restrições arquiteturais aplicadas a componentes, conectores e elementos de dados dentro de um sistema de hipermídia distribuído.

[76] mongoacademico.blogspot.com

c) Os dados são armazenados em memória primária, para posterior cópia em disco, quando for o caso;

d) Realiza operações de inserção, remoção e pesquisa com desempenho elevado;

e) É um servidor TCP com seu funcionamento baseado em um modelo cliente-servidor bem simplificado; e

f) Criador: Salvatore Sanfilippo[77].

6 – Neo4J[78]

a) Tipos de tratamento de dados: Grafo;

b) Seu ponto forte é a rapidez na execução de *queries* (consultas);

c) Dá suporte a transações distribuídas que envolvem mais de uma base de dados;

d) Não há segurança no nível dos dados;

e) Método de consulta SPARQL; e

f) Dá suporte a API java e REST.

O sistema de bases de dados Neo4J, assim como a MongoDB, possuem um segmento de sua aplicação *freeware*, e outra versão com licenciamento específico. É igualmente pertinente, expor sucintamente mais algumas características, quanto APIs (REST e Java), suporte, base de dados e adesão ao GPL, desta forma expressa no Quadro 2:

Quadro 2: Características resumidas das bases de dados

Base de Dados	Tipo	GPL	*MapReduce*	API REST	API Java	Método de Distribuição
Cassandra	Família de Colunas	sim	Sim	não	sim	Hashing
CouchDB	Documento	sim	Sim	sim	sim	Hashing
HBase	Família de Colunas	sim	Sim	sim	sim	Range
MongoDB	Documento	sim	Sim	sim	sim	range
Neo4J	Grafo	parcial	Não	sim	sim	não se aplica
Redis	Chave/valor	sim	Não	não	sim	Hashing

Fonte: Elaborado pelo autor (2017)

[77] http://*NoSQL*-database.org
[78] http://*NoSQL*-database.org/, https://neo4j.com

4.6 Uma proposta de arquitetura baseada em ferramentas *open source* e de livre licença

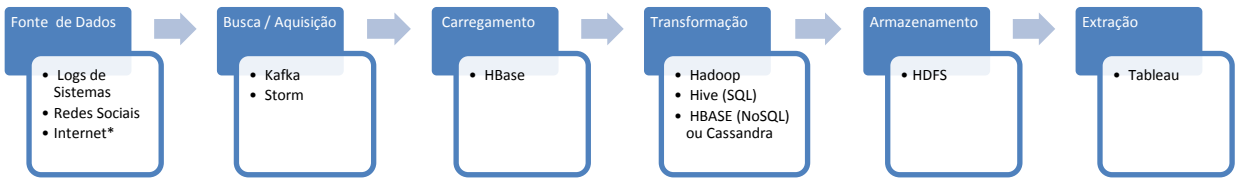
O conteúdo desta seção objetiva propor uma arquitetura de referência que melhore ou possibilite implementações de processos de tratamento de dados não estruturados de grande volume. Este procedimento, quando associado à intenção de implementação de uma proposta arquitetural tende a ser bastante dispendioso em tempo e em custos, dada à dificuldade de se perceber o que realmente se pretende buscar de informação com estes dados. Contudo, este trabalho pretende propiciar não a apresentação de uma exata solução, mas a agregação de conhecimento acerca das múltiplas soluções que podem ser montadas com as ferramentas à disposição no mercado.

As soluções *open source* e de licença livre possuem importância significativa neste contexto, dada à redução de custos que permeia seu uso, se comparadas às opções de aplicativos licenciados. Sendo assim, a escolha das possibilidades que compõem a arquitetura pressuposta, recaem obrigatoriamente, no parecer do autor, sobre estas formas de licença.

Para responder aos objetivos da pesquisa, após a análise das diversas tecnologias de *Big Data* e *NoSQL*, a escolha de Hadoop e HBase como cerne da arquitetura, pareceu ser a mais plausível, primeiramente por permitirem dados não estruturados, em seguida por serem escaláveis, por estarem em constante evolução, por utilizarem *MapReduce* na agregação de dados e por serem bastante utilizadas por empresas de renome que promovem a inserção no mercado de softwares para tratar e analisar grandes volumes de dados.

Resumidamente, a arquitetura apresentada propõe etapas, listadas na Figura 20, e para cada uma delas (que interagem entre si obviamente) é sugerida uma ferramentalização dos acontecimentos sobre os dados, desde a sua origem até a produção de informação.

Figura 20: Proposta de arquitetura de referência com ferramentas open source para o IFFar – baseada em análise de literatura



Fonte: Elaborada pelo autor (2017)

Inicialmente, na busca, aquisição e carregamento os dados são importados a partir de fontes como redes sociais, pesquisas na Internet, registros de *logs*, entre outros, utilizando ferramentas como Storm associado ao Kafka, ambas de licença livre, dentro do que anteriormente se propunha. A coordenação do *cluster* para execução das tarefas de processamento pode ficar a cargo do Storm.

Findada a etapa de busca e consequente carga dos dados, é iniciada a etapa de processamento (transformação) destes dados através das técnicas e funcionalidades que a implementação Apache Hadoop proporciona. A plataforma Hadoop com *MapReduce* fornece uma infraestrutura altamente escalável e tolerante a falhas, inferindo um baixo custo ao processamento e ao armazenamento. O Uso do Hadoop para processar serviços relativos a ETL libera recursos para o processamento analítico posterior.

O processo de armazenamento deste contingente gerado fará uso de um sistema de banco de dados *NoSQL*, que servirá de fonte de consumo das tarefas de análise e apresentação de informações ao usuário. Uma vez que os dados brutos estão no *cluster*, esse banco de dados será então modificado e transformado com base nas necessidades e requisitos do sistema.

Usando o Hadoop para processar trabalhos ETL, não só se fornece eficiência de custo para esses processos de baixo valor, mas também libera recursos valiosos para processar o processamento analítico. Em tempo, o uso do Hive para processamento de dados estruturados bem como Pig para processamento em lote, também pode ser considerado.

A composição da base de dados é de responsabilidade do Hbase, comportando dados não estruturados e parcialmente estruturados, organizados em famílias de colunas. O Hbase por sua vez é um projeto igualmente *open source* escrito em Java, otimizado para consultas em tempo real com alto desempenho, para grandes quantidades de dados distribuídos em *clusters*. Um *cluster* HBase é, na verdade, dois *clusters* distintos trabalhando em conjunto não necessariamente, no mesmo nó. O *cluster* HDFS é composto por um *namenode* (nó de nome), atuando como o ponto de entrada do *cluster* e sabendo quais são os *datanodes* (nós de dados) que armazenam qualquer informação desejada. Ele fornece um banco de dados em tempo real, distribuído, estruturado em cima do sistema de arquivos do Hadoop e seguem, segundo Padhy et al. (2011), os seguintes conceitos:

a) as Tabelas: se originam de linhas e colunas;

b) cada coluna pertence a uma dada família de colunas;

c) cada linha é identificada por sua chave; e

d) uma célula de tabela é a interseção de uma linha e uma coluna.

Ainda, segundo Dimiduk and Khuarana (2013), Hadoop é uma plataforma para armazenar e recuperar dados com acesso aleatório. Com esse sistema de banco de dados é possível a construção de um modelo de dados dinâmico e flexível, pois ele não restringe os tipos de dados nele inseridos. Como o HBase é parte do projeto Hadoop, há uma característica de forte integração, além de permitir que se execute facilmente trabalhos de *MapReduce*, usando o HBase para um *background* de armazenamento de dados.

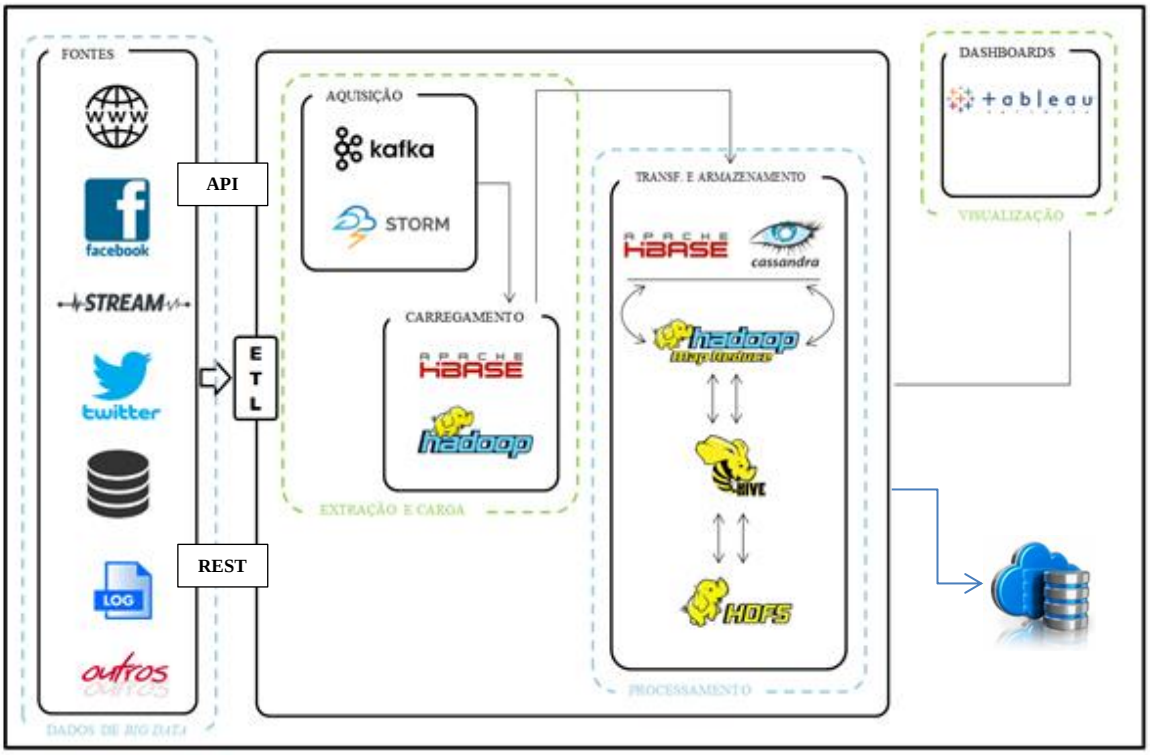
Posteriormente, há um procedimento de agregação, e reserva de resultados pelo processo de *MapReduce* que guarda os resultados indexados num formato próprio para elaboração de procedimentos de consulta sobre esses dados.

Em tempo, cabe salientar que o sistema de banco de dados Cassandra, fora também sugerido com alternativa ao HBase, ou ainda em alguns casos, como coadjuvante, caso o administrador do sistema a ser implementado queira usar "também" esta ferramenta. Com algumas pouquíssimas diferenças, no caso de se optar por uma base de dados Cassandra, os atributos acima descritos para o HBase, também lhe cabem (GREHAN, 2014).

Adicionalmente, cabe dizer que um sistema baseado em HDFS comporta tanto interface para o armazenamento de dados quanto para o armazenamento de metadados e dá suporte a sistemas operacionais Linux, Mac OS e Windows.

A Figura 21 traduz graficamente o exposto nas considerações acima.

Figura 21: Arquitetura proposta para tratamento de dados não estruturados para IFFAR



Fonte: Elaborada pelo autor (2017)

5 AVALIAÇÃO DA PROPOSTA

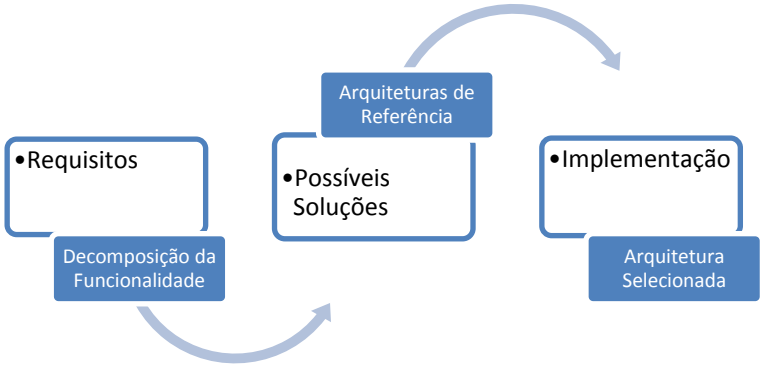
Para melhor elucidar a avaliação da proposta de arquitetura apresentada, associada ao intuito de entender o papel da análise arquitetural dentro do processo de desenvolvimento de uma arquitetura de referência para o tratamento de dados não estruturados, cabe aqui, antes, uma breve contextualização acerca das concepções da engenharia de software, bem como, sobre o método denominado SAAM (*Software Architecture Analysis Method*), pois a identificação de requisitos arquiteturais a um sistema e construção de cenários para definição da arquitetura, é inerente ao que se propões este trabalho.

5.1 Análise de arquitetura de *software*

À medida que os sistemas se tornam maiores e mais complexos, fatores como desempenho, robustez e qualidade, passam a ser considerados também, além das tradicionais técnicas de programação. Tais fatores estão intrínseca e intimamente relacionados à organização arquitetural do sistema (*software*) e contribuíram para o surgimento da Engenharia de *Software* (Naur, 1969), que tem como ideia fulcral utilizar conceitos de engenharia na produção de sistemas de software, face a necessidade de se lidar com o crescimento exponencial e complexidade deste sistemas, primando pela sua confiabilidade. Dentro desta perspectiva surge uma nova metodologia de trabalho: a arquitetura de software, cujos passos principais podem ser vistos na Figura 22.

O êxito em projetos de sistemas de software de grande porte está diretamente associado às premissas da arquitetura de software. O contexto cerne da arquitetura de software é de que um sistema de software com nível de abstração alto pode ser entendido e descrito na forma de subsistemas, perfazendo distintas partes correlatas, relacionadas entre si e interconectadas de alguma forma.

Figura 22: Descrição - arquiteturas de um sistema

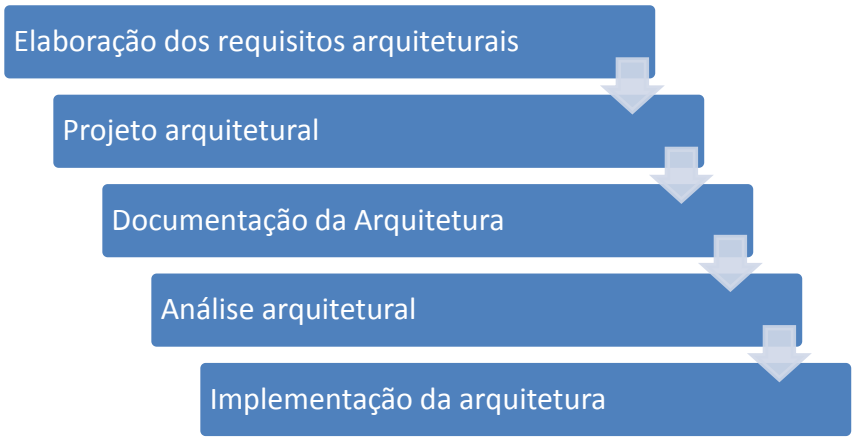


Fonte: Adaptado de Silva Filho, A.M. (2006)

Não se esquecendo dos preditos acima, há de se considerar que a implementação de uma arquitetura é particionada a fim de identificar possíveis subsistemas ou módulos funcionais. Isto permite uma melhor análise sobre componentes do sistema em relação aos seus requisitos previamente impostos, visando encontrar ou apresentar uma arquitetura que satisfaça as necessidades do sistema em si (SILVA FILHO, 2006).

Sendo assim, as arquiteturas apresentadas serão vislumbradas como referências que simbolizam e descrevam as funcionalidades. Tal processo resulta em dar ao projetista, alternativas de arquiteturas disponíveis, descritas e classificadas, compondo assim as arquiteturas de referência. Considerando as arquiteturas de referência, pode-se então definir e formular regras indicativas de boas (melhores) opções ao projeto de um sistema. Etapas estas, mensuradas na Figura 23:

Figura 23: Etapas no projeto da arquitetura de *software*



Fonte: Adaptado de Silva Filho, A.M. (2006)

5.1.1 Método SAAM

Aqui convém uma breve consideração cobre o método de análise de arquitetura se software SAAM, mesmo este não sendo ideal para a tarefa adiante, mas com o objetivo de elucidar algumas características posteriormente incorporadas no processo de análise da arquitetura proposta para o tratamento de dados *NoSQL*.

O método de análise de arquitetura de software SAAM (*Software Architecture Analysis Method*) tem com principal objetivo auxiliar arquitetos de software na

escolha/comparação de proposições de soluções arquiteturais de sistema. Compreende os seguintes objetivos (Kazman et al., 1994):

a) Definir um conjunto de cenários representativos do uso do sistema em relação ao contexto proposto;
b) Utilizar cenários para dar uma visão funcional do domínio a que se propõe o sistema, associando cenários a funções existentes; e
c) Realizar a análise das arquiteturas propostas, através do uso dos cenários e das partes funcionais, concomitantemente.

Para cada cenário apresentado é fornecido, através da análise, uma pontuação inerente às arquiteturas, objetos de avaliação. O avaliador determina os pesos dos cenários considerados, bem como a pontuação das arquiteturas proponentes.

Fazendo uso de cenários, propostos pelo método SAMM, o analista pode usar uma descrição de arquitetura para mensurar o potencial do sistema que propõe ser construído. É importante, contudo, dar uma atenção especial ao contexto no qual o sistema encontra-se inserido, bem como considerar as circunstâncias específicas àquele contexto.

SAAM compreende um conjunto de cinco passos interdependentes:

a) Desenvolvimento de cenários – visa ilustrar os tipos de atividades que o sistema dará suporte;
b) Descrição de arquitetura – cada análise deve possuir uma representação arquitetural do sistema;
c) Avaliação de cada cenário – para cada cenário determina-se uma arquitetura candidata e se esta atende aos requisitos do cenário (suporte direto) ou se alguma modificação é necessária (suporte indireto);
d) Determinação da interação de cenários – uma interação entre cenários ocorre quando dois ou mais cenários exigem modificações de alguma natureza; e
e) Avaliação de cenários e da interação de cenários - realiza-se uma avaliação global atribuindo um escore a cada cenário

O resultado previsto com o método SAAM é ter na forma de resultados, a produção dos cenários inerentes ao sistema, primando pela qualidade e proporcionando o mapeamento entres estes cenários e os componentes da arquitetura proposta. O processo de análise para se chegar a este resultado é expresso nas seguintes etapas: Especificação dos requisitos do sistema, descrição da arquitetura, extração de cenários, priorização de cenários, avaliação da

arquitetura em relação aos cenários, interpretação e apresentação dos resultados (KAZMAN et al., 1994).

Segundo Babar et al. (2006), os modelos arquiteturais detém importante característica de servirem de elo entre os requisitos de um sistema e sua implementação fática. Coloca ainda o referido autor, que estes modelos são considerados o primeiro conjunto de decisões de um projeto com correlação ao atendimento dos requisitos previamente propostos.

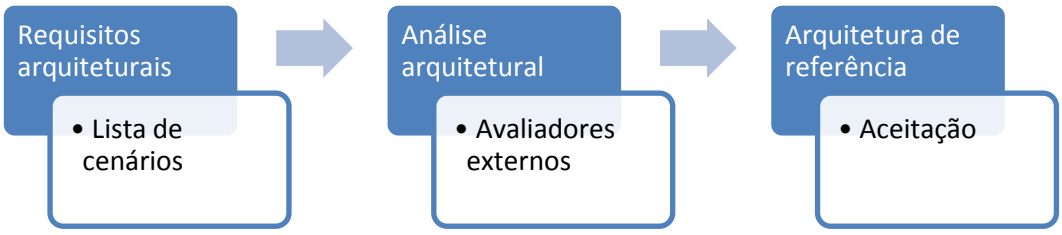
Krutchen et al. (2006) afirmam oportunamente, que do ponto de vista prático, é possível controlar o desenvolvimento de sistemas por meio da arquitetura de software. Seguindo neste raciocínio, arquiteturas de referência se designam como arquiteturas especiais que canalizam ações para se chegar à especificação de uma arquitetura mais concisa e específica. Porém, é necessário que se diga, que métodos de avaliação de arquiteturas como SAAM, por exemplo, não podem ser diretamente aplicados a esse tipo de arquitetura. Isso se deve, segundo Bass et al. (2003), ao fato de existirem significativas diferenças entre arquiteturas concretas e arquiteturas de referência. A principal delas consiste em que arquiteturas de referência são de natureza genérica e são projetadas para atender funcionalidades de interesse de todos os *stakeholders* de um domínio específico. Em decorrência destas caracterizações, necessita-se compor metodologias específicas, ou adaptadas, para avaliação de arquiteturas de referência.

Pelas prerrogativas apresentadas acerca de arquiteturas de referência, bem como sobre a aplicabilidade do método SAAM, para avaliação da proposta adaptou-se o método SAAM, em acordo com a finalidade da pesquisa.

5.2 Metodologia de avaliação da arquitetura proposta

Com a intenção de colher uma opinião critica sobre a arquitetura proposta, foram convidados a formar conceitos sob um olhar técnico, um grupo de analistas, docentes e profissionais da área de TI, Banco de Dados, Arquitetura de Software e correlatas, para que fosse viável mostrar aqui um conceito relacionado à avaliação, mesmo que num nível de abstração mais elevado, sinteticamente expresso pela Figura 24. A criação de cenários e descrição do método de avaliação é descrito nos itens próximos.

Figura 24: Análise arquitetural para uma arquitetura de referência



Fonte: Elaborada pelo autor (2017)

5.3 Cenários

Para efetivar o processo de avaliação, adaptado do método SAAM, torna-se necessário a criação de cenários. Um cenário é uma situação na qual a arquitetura deve suportar e podem ser de casos de uso ou exploratórios.

Os seguintes cenários foram criados em decorrência da proposição:

Cenário 1 – Permitir a coleta de dados de diferentes fontes (sistemas, rede social, BDs, etc);

Cenário 2 – Prover o carregamento de dados de formatos diversos (texto, imagem, *logs*, arquivos, *streaming*, etc);

Cenário 3 – Permitir armazenamento em larga escala;

Cenário 4 – Permitir a análise e transformação de dados de diferentes formatos;

Cenário 5 – Permitir escalabilidade;

Cenário 6 – Fornecer mecanismo para tolerância a falhas;

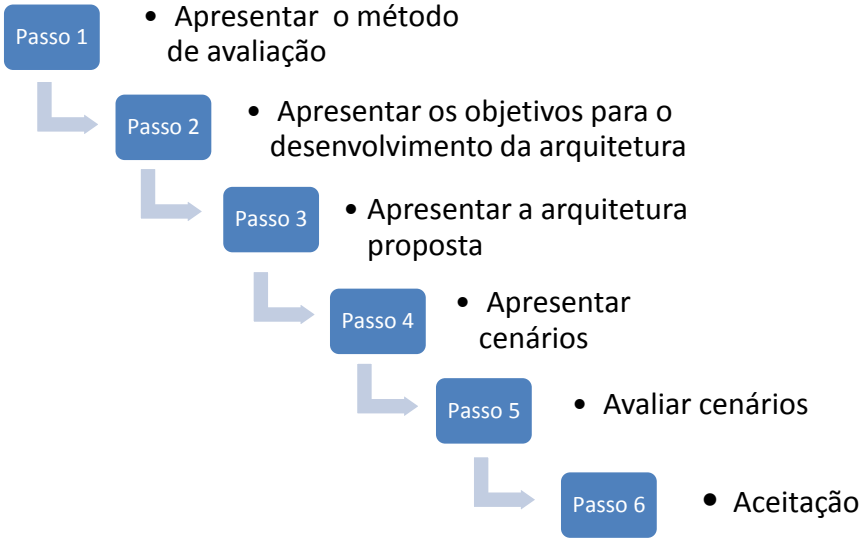
Cenário 7 – Fornecer suporte para serviços de *Cloud*; e

Cenário 8 – Permitir a visualização dos dados transformados.

Para com o objetivo deste trabalho e enfatizando a proposição de uma arquitetura de referência a ser analisada pelo método adaptado do SAMM, acredita o autor que os cenários acima apresentados sejam pertinentes e suficientes ao contexto, bem como ao objetivo da avaliação.

O método SAAM foi readaptado para o contexto da pesquisa, compondo seis etapas descritas na Figura 25:

Figura 25: Etapas do Método SAAM adaptado



Fonte: Elaborada pelo autor (2017)

5.4 Descrição do Processo de Avaliação

Como já mencionado, a metodologia utilizada foi baseada na metodologia *Software Architecture Analisys Method* (SAAM), adaptando-se às necessidades da proposição deste estudo, montando-se uma equipe remotamente distribuída para aferir acerca dos cenários e sobre a arquitetura presumida pelo autor. A seguir, como próximo segmento do processo de avaliação, é feita a apresentação da arquitetura propriamente dita, de forma que todos os avaliadores a entendam.

Em seguida, deve-se contextualizar a equipe de avaliação acerca do conceito de "cenário". Finalmente, os cenários são analisados quanto sua interação entre si e quanto à viabilidade da arquitetura em relação à contemplação dos mesmos, para que se tenham então os efetivos resultados e pareceres da avaliação.

Já se faz evidente que o processo de avaliação da arquitetura é tão importante quanto sua definição. Sendo assim, os participantes do referido processo, na condição de avaliadores, devem preencher o quadro enviado (Apêndice B), tomando por base a arquitetura de referência para tratamento de dados não estruturados, sob uma perspectiva de realidade do IFFar (Instituto Federal de Ciência Tecnologia e Educação Farroupilha). Ainda, cada avaliador deve possuir conhecimento (literário ao menos) acerca dos pré-requisitos, não

necessariamente todos os citados, mas os inerentes ao que se propõe o cenário, visto que alguns são ambíguos. A partir destas iniciais condições, pode então o avaliador, segundo sua análise, afirmar se a arquitetura de referência atende ou não, o cenário proposto.

O avaliador poderá tecer parecer ou opiniões acerca de cada cenário em relação à arquitetura. Exemplificando ferramentas adjacentes ou concorrentes, por exemplo, ou ainda questionando ou sugerindo qualquer ação, prática ou situação.

O avaliador será, antes de iniciar o processo de avaliação, devidamente orientado através de um resumo acerca do assunto, devendo este ser previamente lido, para um melhor entendimento do método de avaliação e dos propósitos desta.

O quadro com cenários, pré-requisitos, avaliação e considerações está disposto no Apêndice B.

5.5 Equipe de Avaliação

Para dar viabilidade ao processo de avaliação desta arquitetura de referência, propõe-se colher a apreciação técnica de especialistas externos, julgados aptos, por titulação ou cargo, sendo estes especialistas, mestres ou doutores, para aferir com imparcialidade suas observações sobre todos os quesitos inerentes ao propósito. A composição da equipe de avaliação é diretamente proporcional à condição de veracidade do processo avaliativo e não obstante dizer, são raros os profissionais que detém conhecimento necessário para executar tal procedimento.

A equipe de avaliação será enquadrada nos aspectos elencados conforme mostra o Quadro 3.

Quadro 3: Quanto à formação/cargo que exerce/ área em que efetivamente atua

Doutor Docente em BD/ *Data Mining*	2
Mestre Docente em BD/*Data Mining*	
Doutor Docente em Desenvolvimento de Sistemas	
Mestre Docente em Desenvolvimento de Sistemas	3
Doutor Docente em Áreas Correlatas	
Mestre Docente em Áreas Correlatas	2
Doutor Analista de TI em BD/ *Data Mining*	
Mestre Analista de TI em BD/*Data Mining*	
Doutor Analista de TI em Desenvolvimento de Sistemas	
Mestre Analista de TI em Desenvolvimento de Sistemas	
Doutor Analista de TI em Áreas Correlatas	
Mestre Analista de TI em Áreas Correlatas	

Fonte: Elaborado pelo autor (2017)

Quanto aos avaliadores que receberam convite para participar do processo, apenas um se julgou inapto para tal. Em relação aos demais, todos efetivamente participaram de todas as etapas (contato inicial, reuniões virtuais e resposta via documento) do processo. Destes, 02 (dois) mestres em Tecnologias Educacionais em Rede e professor de Desenvolvimento de Sistemas, 01 (um) mestre em Tecnologias Educacionais em Rede e professor de Desenvolvimento de Sistemas, 01 (um) mestre em Computação Aplicada e professor de Áreas Correlatas, 01 (um) mestre em Ciência da Computação e professor em Áreas Correlatas e 02 (dois) doutores em *Data Mining* e professor de Banco de Dados, totalizando a participação de 07 (sete) profissionais.

5.6 Resultados do processo de Avaliação

Cada participante do processo de avaliação respondeu ao formulário (Apêndice B) de forma independente. Os participantes foram incentivados a tecerem comentários na forma de pareceres acerca de cada cenário, sugerindo, concordando, ou discordando acerca das ações ou ferramentas pertinentes ao processo exposto no cenário.

A partir disso, as respostas foram sistematizadas e analisadas pelo autor. Foram considerados os aspectos inerentes à relevância dos cenários e à satisfação destes pela arquitetura de referência baseada em ferramentas *open source*.

Dos cenários propostos, nenhum destes recebeu conceito de rejeição, conforme visto no Quadro 4. Alguns foram plenamente aceitos e a outros se sugeriram algumas verificações ou advertências, sem, contudo, rebaixá-los a ineficientes. Segue uma descrição em relação a cada um dos cenários e, por conseguinte, da arquitetura proposta.

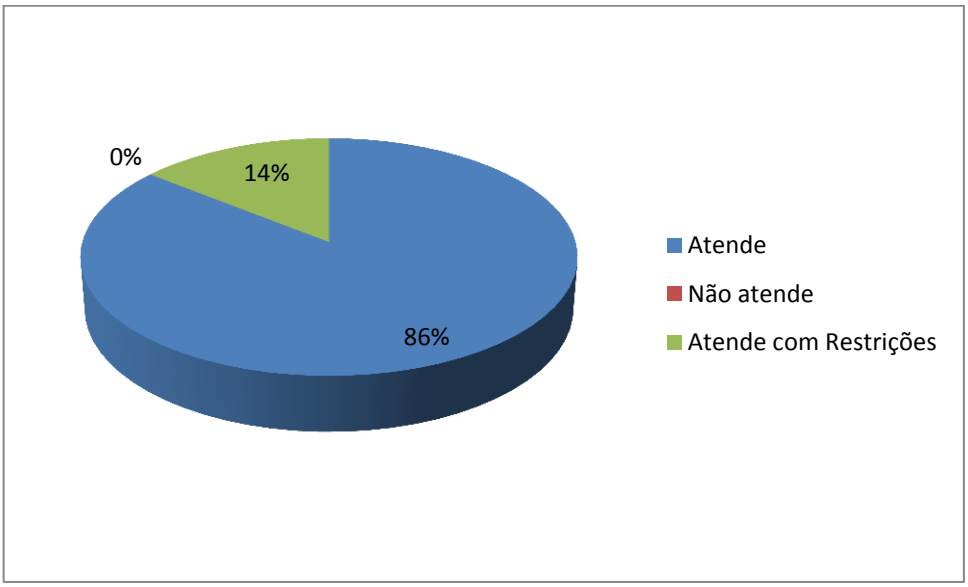
Quadro 4: Avaliação dos cenários

Cenário	1 - Atende	2 - Não Atende	3 - Atende com ressalvas
C1 – Permitir a coleta de dados de diferentes fontes (sistemas, rede social, BDs, etc)	6		1
C2 – Prover o carregamento de dados de formatos diversos (texto, imagem, logs, arquivos, streaming, etc);	6		1
C3 – Permitir armazenamento em larga escala (*Big Data*)	6		1
C4 – Permitir a analise e transformação de dados de diferentes formatos	6		1
C5 – Permitir escalabilidade	6		1
C6 – Fornecer mecanismo para tolerância a falhas	6		1
C7 – Fornecer suporte para serviços de Cloud	6		1
C8 – Permitir a visualização dos dados transformados	6		1

Fonte: Elaborado pelo autor (2017)

Como é possível notar pela visualização da Figura 26, na opinião dos participantes, através das manifestações dos avaliadores acerca dos cenários, a arquitetura proposta pelo autor obteve o conceito 1 – "atende" em 86% dos avaliadores, conceito 3 atende parcialmente ou com ressalvas na opinião de 14% dos avaliadores, enquanto que nenhum avaliador questionou a viabilidade da arquitetura. Sendo assim pode-se considerar que a arquitetura respondeu positivamente à avaliação, com ressalvas que não ofuscam nem inviabilizam sua aplicabilidade. Atributos de armazenamento em larga escala, diversidade, escalabilidade, robustez e utilização de recursos sugerem-se contemplados.

Figura 26: Avaliação dos especialistas quanto ao atendimento dos cenários pela arquitetura de referência



Fonte: Elaborada pelo autor (2017)

Especificamente, quanto ao cenário C1, apenas um dos avaliadores observou que é adequada à proposta de arquitetura, contudo cabe uma atenção especial a obrigatoriamente usar um filtro (*key words*[79]) para orientar a coleta. Para o cenário C2, o mesmo avaliador mensura que deve ser observado e quantificado o esforço computacional para acionar motores de busca, assim como para o cenário C3. Ainda em relação ao cenário C6, há observância no sentido de considerar uma taxa de *Main Time to Failure*[80] para o ambiente dito tolerante a falhas.

[79] Palavra ou identificador que tenha um significado particular para a linguagem de programação.
[80] "Período médio entre falhas": é um valor atribuído a um determinado dispositivo ou aparelho para descrever a sua confiabilidade.

Quanto aos demais cenários, foram considerados todos adequados pela totalidade dos avaliadores e presumidos de sucesso quanto à aplicabilidade pela arquitetura proposta.

5.7 Ameaças ao processo de Avaliação

O processo de avaliação desta arquitetura de referência, previamente apresentada e descrita, procurou surtir efeito de verificação e aceitação inicial, ou seja, não visa consolidar de imediato a proposta, mas trazer à tona a sua possibilidade ou não de utilização futura, seja como base de implementação ou então apenas como uma atividade precursora de aperfeiçoamento de arquitetura definitiva para tratamento de dados não estruturados e de volume elevado.

Contudo, é necessário que se expresse aqui algumas ameaças à avaliação procedida. Tais ameaças não afetam diretamente o resultado final da avaliação, porém é necessário que se faça uma análise delas, visando aprimoramentos futuros.

Um primeiro indício de ameaça à confiabilidade da avaliação está relacionado ao considerado pequeno número de avaliadores consultados. Sugere-se então uma amplitude maior quanto aos participantes numa etapa futura.

Outro aspecto, mesmo considerando e impondo como pré-requisitos, habilidades e conhecimentos, mesmo que apenas teóricos aos avaliadores, não se pode pressupor que de fato isso ocorra com primazia, visto que não fora feito nenhum teste verificador sobre os avaliadores acerca deste pressuposto conhecimento prévio. Há de se ressaltar aqui também, a dificuldade em prover disponibilidade dos avaliadores para com as solicitações de reuniões para discutir o processo e a avaliação em si.

Outra ameaça se dá pelo fato de não ter havido uma implementação efetiva e casos de testes, com base nos cenários propostos. Houve apenas uma previsão acerca do estudo das ferramentas e a capacidade destas de atender os requisitos, com base em análise de literatura apenas.

6 CONSIDERAÇÕES FINAIS

Neste capítulo, são apresentadas as conclusões que emergiram ao longo e após esse trabalho. Uma vez que se pôde de fato sugerir uma arquitetura de referência, tanto num nível maior de abstração quanto num aspecto de implementação plausível com uso de ferramentas, cabe aqui um apontamento sobre o trabalho de pesquisa como um todo.

6.1 Trabalho realizado

O problema de pesquisa inicialmente proposto questiona como tratar dados não estruturados num ambiente inerente aos Institutos Federais de Educação. Dados estes de origens diversas como Internet e redes sociais, além de constituírem volume considerável.

Preocupou-se também nesta pesquisa com questionamentos relacionados à quais os dados a serem tratados, a origem dos dados, a transformação dos dados, quais informações devem ser produzidas, quais as principais ferramentas que poder-se-ia utilizar neste processo, além é claro de contextualizar tudo isso com referência aos Institutos Federais de Educação. Desta feita, preocupou-se igualmente em se referenciar a dados abertos, dados governamentais, por ser uma instituição governamental.

Para solucionar estes problemas, foi adotado um método de pesquisa, norteado por DSR, para se propor um mecanismo de tratamento de dados *NoSQL*, na forma de uma arquitetura de referência para tratamento de dados não estruturados.

Nos procedimentos de revisão bibliográfica, ao longo deste trabalho foi realizada uma ampla revisão teórica sobre temas variados inerentes ao problema, tornando possível estudar e produzir conhecimento quanto às tecnologias envolvidas com *Big Data*, *NoSQL*, ferramentas para tratamento de dados de *Big Data* e/ou *NoSQL* além contextualizar a importância dos dados para educação.

Para dar propósito à execução dessa pesquisa, preocupou-se em mostrar através de *survey*, o grau de ineditismo do assunto "*NoSQL*" e seus adjacentes no IF (IFFar). Tal pesquisa atingiu a grande maioria dos analistas de técnicos de TI de uma instituição federal de ensino e corroborou em fomentar a necessidade de abordagem do assunto, na forma de elaboração deste trabalho. Detalhes sobre esta pesquisa em particular estão mostrados na Seção 3.4.

A partir deste prévio procedimento de pesquisa bibliográfica, formaram-se alicerces para transcorrer um prospecto de arquitetura, que se propõe a resolver o problema inicial. Esta

arquitetura foi guiada por ferramentas *open source*, que tiveram seu uso justificado através de conceitos técnicos sobre seu funcionamento, serviram como base para o projeto arquitetural.

Finalmente, foi então procedida uma temática de avaliação e executada (conforme descrito no capítulo quatro) por meio de entrevista a especialistas aptos a dar aceitação a esta arquitetura de referência. Apesar de terem sido apontadas algumas ressalvas, pode-se afirmar que a arquitetura respondeu positivamente à avaliação.

6.2 Análise dos resultados

Os resultados gerais foram obtidos a partir do processo de avaliação executado junto a especialistas diversos, conforme ilustra o Quadro 6. De acordo com os participantes, no referido processo, esta amostra arquitetural contempla os padrões necessários para fomentar sua utilização em um contexto real. Logo, como trabalho futuro, a proposta aqui delineada poderá ser implantada em um ambiente real pelas instituições entusiastas dos objetivos do tratamento e uso de dados não estruturados.

Foi possível presumir que a análise da literatura propiciou a aquisição de conhecimento para projetar uma arquitetura de referência para tratamento de dados não estruturados, utilizando ferramentas disponíveis no mercado na forma *open source* e/ou livres de licença. Foi possível, em um primeiro momento, entender as dimensões que envolveriam uma ação desse porte, a classificação dos dados quanto às diversas formas, além de estruturas variadas. De mesmo modo, a análise da literatura trouxe à tona exemplos de arquiteturas de para tratamento de dados usadas por outras entidades (como é o caso do Facebook ou Linkedin), bem como as características principais sobre as ferramentas candidatas de escolha, como Hbase, Cassandra, MogoDB, Kafka, Tableau, entre tantas, para elencar um leque de tecnologias, instrumentalizando o pesquisador na definição das que comporiam a arquitetura proposta.

Entre as mencionadas, o Kafka foi utilizado, pois objetiva habilitar o processamento em tempo real dos fluxos de dados, além de possuir interação nativa com o Storm e HDFS, podendo ser executado como um *cluster* em um ou mais servidores. Tal processamento produzirá os tópicos de dados a partir de fluxos contínuos de entrada.

Já o Apache Storm pode ser adotado em conjunto com o Kafka, visto que é um sistema *open source*, simples, em tempo real, distribuído, que facilita o processamento confiável de fluxos de dados ilimitados, compatível com qualquer linguagem de programação.

Além disso, é escalável, tolerante a falhas, garante que seus dados serão processados, fácil de configurar e operar.

Ainda, Cassandra e Hbase são bases de dados escaláveis horizontalmente e que trabalham em *cluster*, com configurações simples até mesmo quando da inserção de máquinas novas no *cluster* ou replicação automática do *cluster*. Tem em sua raiz mecanismos de *Map/Reduce* e modelos de orientação a colunas e estão aptas a lidarem com grandes volumes de dados. O HBase oferece uma consistência forte no nível de registro, enquanto que a documentação do Cassandra é mais robusta e didática que a do HBase. Pode-se perceber igualmente, que ambos são gratuitos, *open source* e sob a licença Apache 2.0. O trabalho de pesquisa aponta que tanto uma quanto outra pode compor uma solução com finalidades de prover o tratamento de dados não estruturados.

Quanto ao objetivo principal deste trabalho, que remete invariavelmente à avaliação da arquitetura proposta, conclui-se que a mesma o contempla, uma vez que os cenários propostos, bem como as ferramentas apresentadas para referenciar uma futura implementação, receberam afirmação e concordância com a proposta do autor pelos especialistas, o que já ficara evidenciado na revisão da literatura, onde os benefícios citados na teoria sobre o tratamento de dados, tais como composição de dados, foram confirmados.

Ainda em relação às contribuições acadêmicas da pesquisa exploratória desta dissertação, foi propiciado a novos pesquisadores uma visão sobre tratamento de dados *NoSQL*, na forma de uma proposta arquitetural que pode servir de exemplo para estudos que venham a propor implementações de soluções. Também nesse contexto, a contribuição maior da proposta da arquitetura foi em auxiliar o mapeamento das ferramentas *open source* disponíveis no mercado, como mecanismos de tratamento de dados de grande volume e não estruturados. Além disso, o conteúdo desta dissertação, com dados já compilados acerca de temas como *Big Data* e *NoSQL*, além de dados educacionais, servem de base bibliográfica a leitores de áreas diversas.

A instituição em questão IFFar é igualmente beneficiada pela pesquisa, visto que a maioria dos seus analistas e técnicos em TI, responsáveis pela elaboração das propostas computacionais a serem usadas no IFFar, desconhecem em detalhes a cena que envolve o tratamento de dados não estruturados (conforme demonstrado na sistematização dos dados na Seção 3.4). Logo, para esta, possuir uma análise sobre si mesma a respeito do tema, possibilita acréscimo de subsídios. A instituição também é beneficiada na proposição de um produto, mesmo que de implementação futura. Por fim os gestores, enquanto responsáveis por

um órgão governamental e de educação, pesquisa e extensão têm neste trabalho um veiculo de reflexão e ação em relação à análise de dados de redes sociais, para compor matéria prima nas tomadas de decisões.

6.3 Dificuldades e Limitações

Em relação às dificuldades encontradas, cita-se a questão de adentrar num campo relativamente novo, com autores e trabalhos de quantidade um pouco mais restrita quando se trata de *NoSQL*, implicando em o autor estar em constante atenção sobre publicações mais recentes e pertinentes para complementar a fundamentação teórica e dar base à construção da arquitetura proposta.

As limitações durante a execução deste trabalho ocorreram na forma da pouca interação dos demais profissionais de TI da Instituição e pela dificuldade em se testar efetivamente, via implementação de um *cluster*, a arquitetura sugerida.

6.4 Trabalhos Futuros

Sugere-se implantar em um ambiente real a atual proposta delineada neste trabalho aplicando esta arquitetura na construção de um projeto de implantação de um sistema que trate dados não estruturados, com o objetivo de prover informação útil acerca destes para serem aplicadas em gestão educacional, bem como na disponibilização de informações para a população no que diz respeito às diversas atividades referenciadas e enumeradas a uma Instituição Federal de Ensino. É fundamental o investimento governamental, visto se tratar de instituição deste tipo o objeto deste estudo, em estratégias para melhorar a vida comprendida pela sociedade educacional e o cidadão.

Ainda, quanto à investigação futura, seria conveniente a exploração de técnicas e tecnologias voltadas para a segurança deste presumido sistema a ser implementado com base em um arquitetura de referência bem como testes para adequar seu funcionamento à realidade intitucional. Mesmo considerando que os equipamentos detidos pelas unidades de uma instituição de ensino, possam prover um sistema distribuído a contento, cabe frisar que tanto o Hadoop como as bases de dados *NoSQL* requerem um significativo esforço de implantação estrutural e configuração.

O papel analítico a ser devenvolvido pela aplicação deve ser validado com eficácia, para dar proveito útil ao dado trabalhado. Um modelo "*as a service*" para processamento analítico, *Data Mining* e visulização, na opinião prematura do autor, seria pertinente.

REFERÊNCIAS

ABRAMOVA, V., BERNARDINO, J. *NoSQL* Databases. In Proceedings of the International C* Conference on Computer Science and Software Engineering -C3S2E ’13. pp. 14–22. 2013. Available at: http://dl.acm.org/citation.cfm?id=2494444.2494447.

ABRAMOVA, V., BERNARDINO, J., FURTADO, P. Experimental Evaluation of *NoSQL* Databases. International Journal of Database Management Systems, 6(3), pp.1–16. 2014.

AGRAWAL, Divyakant; DAS, Sudipto; EL ABBADI,Amr. *Big Data* and Cloud Computing: Current State and Future Opportunities. In: Proceedings of the 14th International Conference on Extending Database Technology. ACM, 2011. Disponível em: <http://delivery.acm.org/10.1145/1960000/1951432/p530-agrawal.pdf?ip=200.132.175.100&id=1951432&acc=ACTIVE%20SERVICE&key=344E943C9DC262BB%2E7471D66E3620B64D%2E4D4702B0C3E38B35%2E4D4702B0C3E38B35&CFID=766147081&CFTOKEN=26026165&__acm__=1495560681_5a1aaa6b153b9d6bc0ee090c0157a0a3>. Acesso em Mar 2017.

ALVAREZ, Guilherme M.; CECI, Flávio; GONÇALVES, Alexandre L. Análise Comparativa dos Bancos Orientados a Grafos de Primeira e Segunda Geração–Uma Aplicação na Análise Social. III Encontro de Inovação em SI. Florianópolis. 2016. Disponível em:<http://www.lbd.dcc.ufmg.br/colecoes/eise/2016/003.pdf>. Acesso em Mar 2017

APACHE. Apache Flume: documentação. Disponível em: <https://flume.apache.org//>. Acesso em 28 mai. 2016.

APACHE. Apache Kafka: documentação. http://kafka.apache.org/. Accesso em: 28 de Mai de 2016.

BABBIE, E. The practice of social research. 4th ed. Belmont, Wadsworth Publ., 1986.

BAPTISTA, Cláudio. Banco de Dados Capítulo 2: Modelo Relacional. Disponível em <http://www.dsc.ufcg.edu.br/~baptista/cursos/BDadosI/Capitulo22.pdf>. Acesso em 10 de Março de 2017.

BAR, Jeff. Arquitetura Lambda para processamento Batch e Tempo Real na AWS com Spark Streaming e Spark SQL. 2016. Disponível em:< https://imasters.com.br/infra/aws/arquitetura-lambda-para-processamento-batch-e-tempo-real-na-aws-com-spark-streaming-e-spark-sql/?trace=1519021197&source=single> Acesso em: 10 de mar 2017.

BARROS, A.; CANABARRO, D. R.; CEPIK, M. A. C. Para além da e-Ping: o desenvolvimento de uma plataforma de interoperabilidade para e-Serviços noBrasil. In: BRETAS, N. L.; MESQUITA, C. (Ed.). Panorama da Interoperabilidade. Brasília, DF: Ministério do Planejamento, Orçamento e Gestão, 2010. P. 137-157.

BASS, L., CLEMENTS, P., KAZMAN, R. (2003). Software Architecture in Practice. Addison-Wesley.

BIJNENS, M. Lambda Architecture » λ lambda-architecture.net. Disponível em: <http://lambda-architecture.net/>. Acesso em: 12 dez. 2016.

CALDAS, Max Silva; SILVA, Emanoel Costa Claudino. Fundamentos e aplicação do *Big Data*: como tratar informações em uma sociedade de yottabytes. Bibliotecas Universitárias: pesquisas, experiências e perpectivas, v. 3, n. 1, 2016. Disponível em: < https://seer.ufmg.br/index.php/revistarbu/article/view/1995>

CATTELL, R. Scalable sql and *NoSQL* data stores. SIGMOD Rec., ACM, New York, NY,USA, v. 39, n. 4, p. 12–27, may 2011. ISSN 0163-5808. Disponível em: <http://doi.acm.org/10.1145/1978915.1978919>. Citado na página 51. Acesso em 15 de fev de 2017.

CERVO, A. L. BERVIAN, P. A. Metodologia científica. 5.ed. São Paulo: Prentice Hall, 2002.

CHAUDHURI, S., DAYAL, U., NARASAYYA, V. An overview of business intelligence technology. Commun. ACM, 54 (2011), pp. 88–98

Cuzzocrea, A.,SONG, I.-y., DAVIS, K. C. Analytics over large-scale multidimensional data: the*Big Data* revolution! In Int'l Workshop on Data Warehousing and OLAP (DOLAP), 2011.

DATASTAX; Comparing the Hadoop Distributed File System (HDFS) with the Cassandra File System (CFS), 2013, http://www.datastax.com/wp-content/uploads/2012/09/WP-DataStax-HDFSvsCFS.pdf

DAVENPORT, T. H. Competing on Analytics: the new science of winning, 2007.

DAVENPORT, T. H. Enterprise analytics: Optimize performance, process, and decisions through *Big Data*. Upper Saddle River, New Jersey: FT Press OperationsManagement, 2012.

DAVENPORT, T. H. (2014). How strategists use "*Big Data*" to support internal business decisions, discovery and production. Strategy and Leadership, 42(4), 45–50.

DEAN J., S. GHEMAWAT, 2004, *MapReduce*: Simplified Data Processing on Large Clusters, OSDI'04 Proceedings of the 6th conference on Symposium on Operating Systems Design & Implementation, volume 6, pp. 10-10.

DEAN, J.; GHEMAWAT, S. *MapReduce*: A flexible data processing tool. 2010. Disponível em: <http://doi.acm.org/10.1145/1629175.1629198>. Acesso em: 10 Ago. 2016.

DEMCHENKO, Y., Grosso, P., De Laat, C. & Membrey, P. (2013). Addressing *Big Data* issues in scientific data infrastructure. *Colaboration Technologies ans Systems (CTS)*.

DEAN J, GHEMAWAT S. *MapReduce*: Simplified data processing on large clusters, osdi'04: Sixth symposium on operating system design and implementation, san francisco, ca, december, 2004. S Dill, R Kumar, K McCurley, S Rajagopalan, D Sivakumar, ad A Tomkins, Self-similarity in the Web, Proc VLDB. 2001;

DUMBILL, Edd. The Data Lake Dream. **Forbes: Data Driven,** Usa, 14 jan. 2014. Disponível em: <http://www.forbes.com/sites/edddumbill/2014/01/14/the-data-lake-dream/>. Acesso em: 02 set. 2015.

DUMBILL, Edd. **Planning for *Big Data*.** Boston-USA: O'really, 2012.

FAN, Wei; BIFET, Albert. Mining *Big Data*: Current Status, and Forecast to the Future. SIGKDD Explorations, China, v. 2, n. 14, p.1-5, mar. 2012. Disponível em: http://www.kdd.org/sites/default/files/issues/14-2-2012-12/V14-02-01- Fan.pdf Acesso em: 15 dez. 2014.

FERNANDES, A. G. **E-governo: o que já fazem estados e municípios**. 2000.

FERNANDES, Agnaldo Aragon; ABREU, Vladimir Ferraz de. Implantando a Governança de TI: da estratégia à gestão de processos e serviços. 3. ed. Rio de Janeiro: Brasport, 2012.

GIFFINGER, Rudolf et al., Smart cities Ranking of European medium-sized cities. Vienna: Vienna University Of Technology, 2007. 28 p. Disponível em: <http://www.smartcities.eu/download/smart_cities_final_report.pdf>. Acesso em: 06 jan. 2015.

GIL, A. C. Como elaborar projetos de pesquisa. 4a. ed. São Paulo: Atlas, 2002.

GREHAN, R. *Big Data* showdown: Cassandra vs. HBase. Revista InfoWorld. [on-line]. San Francisco: USA. 2 Abr. 2014. Disponível em:< http://www.infoworld.com/article/2610656/database/big-data-showdown--cassandra-vs--hbase.html?page=4>. Acesso em:<04 abr. 2017>

HEVNER, A.R.; MARCH, S.T.; PARK, J.; RAM, S. Design science in information systems research. MIS Quarterly, v. 28, n. 1, p. 75-105, 2004.

HADOOP WIKI. Disponível em: <https://wiki.apache.org/hadoop/>. Acesso em 22 fev. 2017.

KAUR, J., Kaur, H. & Kaur, K., 2013. A Review on Document Oriented and Column Oriented Databases. International Journal of Computer Trends and Technology, 4, pp.338–344. Disponível em: http://www.ijcttjournal.org/Volume4/issue-3/IJCTT-V4I3P128.pdf. Acesso em : 15 de fev. de 2017

KAZMAN, R., Bass, L., Abowd, G., and Webb, M. (1994). SAAM: A method for analyzing the properties of software architectures. In Proc. of the 16th Int. Conf. on

KIM, G-H; TRIMI, S.A.;JI-HYONG, C. *Big Data* Applications in the Government Sector. communications of the ACM, vol. 57, no. 3, 2014.

KLEIN, J.; BUGLAK ,R.; BLOCKOW, D.; WUTTKE, T.; COOPER, B. A Reference Architecture for *Big Data* Systems in the National Security Domain. 2nd International Workshop on *Big Data* Software Engineering (2016).

KRUCHTEN, P. B.; OBBINK,H.; SATANFORD, J. The past, present, and future for software architecture. Software, IEEE, v. 23, n.2, p. 22-30, 2006.

Kuznetsov, S.D. & Poskonin, a. V., 2014. *NoSQL* data management systems. Programming and Computer Software, 40(6), pp.323–332. Disponível em: http://link.springer.com/article/10.1134/S0361768814060152. Acesso em 15 fev. 17.

LACERDA, D. P.; DRESCH, A.; PROENÇA, A; JÚNIOR, J. Design Science Research: método de pesquisa para a engenharia de produção. Gest. Prod., São Carlos, v. 20, n. 4, p. 741-761, 2013.

LAKSHMAN, A. and Malik, P. (2010). Cassandra: a decentralized structured storage system. SIGOPS Oper. Syst. Rev. Software Engineering, pages 81–90, Sorrento, Italy.

LIN J, DYER C. Data-intensive text processing with *MapReduce*. Synthesis Lectures on Human Language Technologies. 2010;3(1):1–177.

MAIER, M. Towards a *Big Data* reference architecture. Master's thesis Eindhoven University of Technology (October 2013).

MANYIKA, James et al., *Big Data*: The next frontier for innovation, competition, and productivity. Nova York: Mckinsey Global Institute, 2011. 20 p. Disponível em: http://www.mckinsey.com/business-functions/digital-mckinsey/our-insights/big-data-the-next-frontier-for-innovation. Acesso em: 21 dez. 2016.

MARZ, N.Warren, J. *Big Data*. Tradução . 1. ed. Shelter Island, NY: Manning Publ, 2015.

MAYER-SCHÖNBERGER, Viktor; CUKIER, Kenneth. *Big Data*: A revolution that will transform how we live, work, and think. Boston: Houghton Mifflin Harcourt, 2013.

MORAES, Ana Carolina de; GOMES, Kelly Aparecida, 2015. Redes Sociais na Educação: a importância da capacitação docente. VIII Simpósio Nacional da ABCiber COMUNICAÇÃO E CULTURA NA ERA DE TECNOLOGIAS MIDIÁTICAS ONIPRESENTES E ONISCIENTES ESPM-SP – 3 a 5 de dezembro de 2014

MCAFEE, Andrew; BRYNJOLFSSON, Erik. *Big Data*: the management revo-lution. Harvard Business Review, Brighton, v. 90, n. 10, p. 61-67, oct. 2012. Disponível em: <https://hbr.org/2012/10/big-data-the-management-revolution#>. Acesso em: 12 set. 2017.

MAHRT, M.; SCHARKOW, M. The Value of *Big Data* in Digital Media Research. Journal of Broadcasting & Electronic Media, 57(1), 20-33, 2013.

NAUR, P., Randell, B. and Buxton, J. (Eds.), "Software Engineering: A Report on a Conference Sponsored by NATO Science Committee, NATO, 1969.

NAVATHE, S. B; ELMASRI, R. Sistemas de Banco de Dados. 6. ed. São Paulo: Pearson Addison Wesley, 2010.

OPEN KNOWLEDGE FOUNDATION. Open data handbook. [2014]. Disponível em: <http://opendatahandbook.org/guide/en/>. Acesso em: 14 set. 2016.

OPEN GOV DATA. Eight principles of open government data. Disponível em: http://resource.org/8_principles.html. Acesso em: 05 jan. 2017.

PADHY, Rabi Prasad; PATRA, Manas Ranjan; SATAPATHY, Suresh Chandra. RDBMS to *NoSQL*: Reviewing Some Next-Generation Non-Relational Database's. International Journal Of Advanced Engineering Sciences And Technologies, Vol No. 11, Issue No. 1, 015 – 030, 2011.

PALMER, B., 2010. Why not try an API? Small software innovations can be powerful branding tools. Brandweek, 1 February, Volume 51, p. 12.

PRITCHETT, Dan. Base: An acid alternative. Queue, v. 6, n. 3, p. 48-55, 2008.

QUAN, Eilen. MINNESOTA METADATA GUIDELINES FOR DUBLIN CORE METADATA. Minnesota Department of Natural Resources. 2000.

RABBITMQ. Disponível em: <http://www.rabbitmq.com/>. Acesso em: 20 fev. 2017

ROBINSON, I., WEBBER, J. & EIFREM, E., 2013. Graph Databases First Edit. M. Loukides & N. Jepson, eds., O'Reilly Media, Inc.

ROMME, A. G. L. Making a difference: Organization as Design. Organization Science, v. 14, n. 5, p. 558-573, 2003. http://dx.doi.org/10.1287/orsc.14.5.558.16769. Acesso 07 de Mar de 2017.

RUSSOM, Philip. *Big Data* Analytics, TDWI Best Practices Report. 2011.

RUSSOM, Philip. TDWI Best Practices Report - *Big Data* Analytics. 2011 by TDWI (The Data Warehousing InstituteTM).

SADALAGE, Pramod J., FOWLER, Martin. *NoSQL* distilled: a brief guide to the emerging world of polyglot persistence.Pearson Education, Inc., 2013.

SALVADOR, Valéria Farinazzo Martins et al., Qualidade de dados para gestão de conhecimento na área de saúde. In: CONGRESSO BRASILEIRO DE INFORMÁTICA EM SAÚDE, 10., 2006, Florianópolis. Anais... . Florianópolis: S.i., 2006. p. 32 - 38. Disponível em: <www.researchgate.net/publication/255631635_Qualidade_de_Dados_para_Gestao_de_Conhecimento_na_Area_de_Saude> . Acesso em: 21 dez. 2016.

SAMBAMURTHY, V.; SUBRAMANI, M. Special issue on information technologies and knowledge management. MIS Quarterly, v.29, pp. 193-195, 2005.

SOUSA, Paulo, 2010. O teorema CAP. Disponível em: http://unrealps.wordpress.com/2010/12/28/o-teorema-cap/Acesso em: 15 dez. 2016.

SUMBALY, R., KREPS, J., SHAH S. The "*Big Data*" Ecosystem at LinkedIn 2013 ACM SIGMOD International Conference on Management of Data, New York, New York, USA (22–27 June, 2013).

TAURION, Cezar. Como se preparar para o hype do data lake? **Computerworld,** São Paulo, p.1-2, 11 dez. 2014. Disponível em: <http://computerworld.com.br/tecnologia/2014/12/11/como-se-preparar-para-o-hype-do-data-lake>. Acesso em: 10 set. 2015.

Tiago Cruz França, Fabrício Firmino de Faria, Fabio Medeiros Rangel, Claudio Miceli de Farias e Jonice Oliveira; **Big Social Data: Princípios sobre Coleta, Tratamento e Análise de Dados Sociais,** 2014, SBC, 1ªed. ISBN 978-85-7669-290-4

VAKKARI, P. Library and Information Science: Its Content and Scope. In: GODDEN IRENE, P. (Org.). Advances in librarianship. San Diego, 1994.

VAN AKEN, J. E. Management Research Based on the Paradigm of the Design Sciences: The Quest for Field-Tested and Grounded Technological Rules. Journal of Management Studies, v. 41, n. 2, p. 219-246, 2004. http://dx.doi.org/10.1111/j.1467-6486.2004.00430.x

WASSAN, Jyotsna Talreja. Discovering *Big Data* Modelling for Educational World. 2014. Procedia - Social and Behavioral Sciences 176 (2015) 642 – 649.

WASSERMAN, S., Faust, K. (1994), "Social Network Analysis: Methods and Applications", Cambridge University Press.

WHITE, T. Hadoop: The Definitive Guide. Third edition. Beijing: O'Reilly, 2012.

WERHMULLER, Claudia Miyuki; SILVEIRA Ismar Frango. Redes Sociais Como Ferramentas de Apoio à Educação, Anais do II Seminário Hispano Brasileiro - CTS, p. 594-605, 2012.

YANG, Heechun. Total Cost of Ownership for Application Replatform by Open-source SW. Procedia Computer Science, v. 91, p. 677-682, 2016. Disponível em:<http://www.sciencedirect.com/science/article/pii/S1877050916313631>. Acesso em: 12 Jan de 2017.

ZIKOPOULOS, P.C. et al.,Understanding *Big Data*: Analytics for Enterprise-Class Hadoop and Streaming Data. McGraw-Hill, New York, 2012.

APÊNDICE A - MATRIZ DE ARTIGOS UTILIZADOS

Quadro 5: Matriz de artigos utilizados

Nome do artigo	Ano	Autor	Publicação
Composable architecture for rack scale *Big Data* computing	2017	Li, Chung-Sheng, et al.	Future Generation Computer Systems 67
Persisting big-data: The *NoSQL* landscape	2017	Corbellini, Alejandro, et al.	Information Systems 63
Big Data e Transparência: Utilizando Funções de *MapReduce* para incrementar a transparência dos Gast os Públicos	2016	Eduardo de Paiva Kate Revoredo	XII Brazilian Symposium on Information Systems, Florianópolis, SC, May 17-20, 2016
Reference Architecture for *Big Data* Systems in the National Security Domain	2016	John Klein Ross Buglak, David Blockow, Troy Wuttke, Brenton Cooper	2nd International Workshop on *Big Data* Software Engineering
An Effective *NoSQL*-Based Vector Map Tile Management Approach	2016	Wan, Lin, Zhou Huang, and Xia Peng	ISPRS International Journal of Geo-Information
Análise Comparativa dos Bancos Orientados a Grafos de Primeira e Segunda Geração–Uma Aplicação na Análise Social	2016	Alvarez, Guilherme M., Flávio Ceci, and Alexandre L. Gonçalves	III Encontro de Inovação em SI, Florianópolis, SC
Use a análise de *Big Data* e de dados rápidos para usufruir da análise como serviço (AaaS)	2016	Chelliah Pethuru Raj, Skylab Vanga	IBM Developer Works
Forensic investigation framework for the document store *NoSQL* DBMS: MongoDB as a case study	2016	Yoon, Jongseong, et al.	Digital Investigation 17
NoSQL Injection: Data Security on Web Vulnerability	2016	Abdalla, Hemn B., et al.	International Journal of Security and Its Applications 10.9
RDBMS, *NoSQL*, Hadoop: A Performance-Based Empirical Analysis	2016	Yassien, Amal W., and Amr F. Desouky	Proceedings of the 2nd Africa and Middle East Conference on Software Engineering
A flexible and scalable architecture for real-time ANT+ sensor data acquisition and *NoSQL* storage	2016	Mehmood, Nadeem Qaisar, Rosario Culmone, and Leonardo Mostarda	International Journal of Distributed Sensor Networks 12.5
Design Assistant for *NoSQL* Technology Selection	2015	John Klein and Ian Gorton	Proceedings of the 1st International Workshop on Future of Software Architecture Design Assistants
Big Data Design	2015	Alberto Abelló	In: Proceedings of the ACM Eighteenth International Workshop on Data Warehousing and OLAP
Análise de desempenho e otimização do Apache HBase	2015	Neves, Francisco Nuno Teixeira	Dissertação de Mestrado

para dados relacionais			
BASIS: uma Arquitetura de *Big Data* para Smart Cities	2015	Costa, Carlos Filipe Machado da Silva	Diss. Universidade do Minho. Escola de Engenharia
Big Data design	2015	Abelló, Alberto	Proceedings of the ACM Eighteenth International Workshop on Data Warehousing and OLAP. ACM
Column-based databases: estudo exploratório no âmbito das bases de dados *NoSQL*	2015	Cunha, José Pedro	Diss. Universidade do Minho. Escola de Engenharia
Dados abertos conectados para a Educação	2015	Bandeira, Judson, et al.	Jornada de Atualização em Informática na Educação 4.1
Data analytics: abordagem para visualização da informação	2015	Ribeiro, Luís Rafael Araújo	Diss. Universidade do Minho. Escola de Engenharia
Discovering *Big Data* Modelling for Educational World	2015	Jyotsna Talreja Wassan	
Maturing, Consolidation and Performance of *NoSQL* Databases-Comparative Study	2015	e Souza, Vanessa Cristina Oliveira, and Marcus Vinícius Carli dos Santos	Proceedings of the annual conference on Brazilian Symposium on Information Systems: Information Systems: A Computer Socio-Technical Perspective-Volume 1. Brazilian Computer Society
Reference Architecture and Classification of Technologies, Products and Services for *Big Data* Systems	2015	Pääkkönen, Pekka, and Daniel Pakkala	*Big Data* Research 2.4
Scalable Database Management in Cloud Computing	2015	Kaur, Pankaj Deep, and Gitanjali Sharma	rocedia Computer Science 70
Enhancing the management of unstructured data in e-learning systems using MongoDB	2015	Stevic, Milorad Pantelija, Branko Milosavljevic, and Branko Rade Perisic	Emerald Insigth Program 49.1
Design Science: Filosofia da Pesquisa em Ciência da Informação e Tecnologia	2014	Marcelo Peixoto Bax	XV ENANCIB (Belo Horizonte)
Governo Aberto: A Tecnologia Contribuindo para Maior Aproximação entre Estado e a Sociedade	2014	Cyntia Barberian Patricia Mello Renata Miranda	Revista do TCU 131
Redes Sociais na Educação: a importância da capacitação docente	2014	Ana Carolina de Moraes Kelly Aparecida Gomes	VIII Simpósio Nacional da ABCiber COMUNICAÇÃO E CULTURA NA ERA DE TECNOLOGIAS MIDIÁTICAS ONIPRESENTES E ONISCIENTES ESPM-SP – 3 a 5 de dezembro de 2014
Discovering *Big Data* Modelling for Educational World	2014	Jyotsna Talreja Wassan	Procedia - Social and Behavioral Sciences Volume 176, 20 February 2015, Pages 642-649

Tossing*NoSQL*–Databases out to Public Clouds	2014	Antoniadis, et al.	2014 IEEE/ACM 7th International Conference on Utility and Cloud Computing
Analise de estratégias de acesso a grandes volumes de dados	2014	de Oliveira, Douglas Ericson M., Cristina Boeres, and Fábio Porto	29 th SBBD { SBBD Proceedings { ISSN 2316-5170 October 6-9, 2014 { Curitiba, PR, Brazil
Armazéns de dados em bases de dados *NoSQL*	2014	Pereira, Daniel José Pinto	Tese de Doutorado
Big Social Data: Princípios sobre Coleta, Tratamento e Análise de Dados Sociais	2014	França, Tiago Cruz, et al.	XXIX Simpósio Brasileiro de Banco de Dados–SBBD 14
Critérios para Seleção de SGBD *NoSQL*: o Ponto de Vista de Especialistas com base na Literatura	2014	de Souza, Alexandre Morais, et al.	Anais doX Simpósio Brasileiro de Sistemas de Informação (Londrina–PR, Brasil. 27 a 30/05/2014
BASIS - Uma Arquitetura de *Big Data* para Smart Cities	2014	Carlos Filipe Machado da Silva Costa	Dissertação de Mestrado (Universidade do Minho – Portugal)
Big Social Data: Princípios sobre Coleta, Tratamento e Análise de Dados Sociais	2014	França, Tiago Cruz, et al.	XXIX Simpósio Brasileiro de Banco de Dados–SBBD 14
Plataformas de *Big Data* : Spark, Storm e Flink	2014	Jean Luca Bez	Diss. Mestrado
Enhancing the management of unstructured data in e-learning systems using MongoDB	2014	Stevic, Milorad Pantelija, Branko Milosavljevic, and Branko Rade Perisic	Program 49.1
Implementação de uma solução open source de Business Intelligence com metodologia Ágil	2014	Batista, Luís Pedro Lopes	Dissertação de Mestrado Universidade Nova de Lisboa
Uma arquitetura para cidades inteligentes baseada na Internet das coisas	2014	Tomas, Gustavo Henrique Rodrigues Pinto	Diss. Mestrado UFPE
DSR: Método de Pesquisa para Engenharia de Produção	2013	Daniel Lacerda Aline Dresch Adriano Proença José A. V. Jr	Revista Gestão da Produção
NoSQL no Suporte à Análise de Grande Volume de Dados	2013	Joel Alexandre Luís Cavique	Revista de Ciências da Computação, 2013, nº8
NoSQL Databases: a step to database scalability in Web environment	2013	Jaroslav Pokorny	International Journal of Web Information Systems, v. 9, n. 1, p. 69-82, 2013.
Performance evaluation of a mongodb and hadoop platform for scientific data analysis	2013	Dede, Elif, et al.	Proceedings of the 4th ACM workshop on Scientific cloud computing.
On the necessity of model checking *NoSQL* database schemas when building saas applications	2013	Scherzinger, Stefanie, et al.	Proceedings of the 2013 International Workshop on Testing the Cloud.
Comparing the Hadoop Distributed File System (HDFS) with the Cassandra File System (CFS)	2013	datasax	White Paper BY DATASTAX CORPORATION
Addressing *Big Data* issues in scientific data	2013	Demchenko, Yuri, et al.	Collaboration Technologies and Systems (CTS), 2013

infrastructure			International Conference on. IEEE
Design Sciencee Design Science Researchcomo Artefatos Metodológicos para Engenharia de Produção	2013	ALINE DRESCH	Diss. De Mestrado
Inclusão de funcionalidades *MapReduce* em sistemas de data warehousing	2013	Silva, Dário Almeno Matos da	Dissertação de Mestrado (Universidade do Minho – Portugal
NoSQL no Suporte à Análise de Grande Volume de Dados	2013	Alexandre, Joel, and Luís Cavique	Revista de Ciências da Computação, 2013, nº8
REDES SOCIAIS COMO FERRAMENTAS DE APOIO À EDUCAÇÃO	2013	Miyuki Werhmuller, Claudia, and Ismar Frango Silveira	Revista de Ensino de Ciências e Matemática 3.3
Towards a *Big Data* Reference Architecture	2013	Maier, Markus, A. Serebrenik, and I. T. P. Vanderfeesten	Master's Thesis University of Eindhoven
Redes Sociais Como Ferramentas De Apoio À Educação	2012	Claudia Miyuki Werhmuller Ismar Frango Silveira	Anais do II Seminário Hispano Brasileiro - CTS, p. 594-605, 2012
I/O Characteristics of *NoSQL* Databases	2012	Jiri Schindler	. Proceedings of the VLDB Endowment
Bancos de Dados *NoSQL*: conceitos, ferramentas, linguagens e estudos de casos no contexto de *Big Data*	2012	Vieira, Marcos Rodrigues, et al.	Simpósio Brasileiro de Bancos de Dados
Bases de Dados *NoSQL*	2012	Cardoso, Ricardo Manuel Fonseca	Diss. Instituto Superior de Engenharia do Porto
RDBMS to *NoSQL*: Reviewing Some Next-Generation Non-Relational Database's	2011	Rabi Prasad Padhy Manas Ranjan Patra Suresh Chandra Satapathy	International Journal Of Advanced Engineering Sciences And Technologies, Vol No. 11, Issue No. 1, 015 - 030
Abordagem *NoSQL* – uma real alternativa	2011	Renato Molina Toth	Sorocaba, São Paulo, Brasil: Abril, v. 13, 2011.
On the elasticity of *NoSQL* databases over cloud management platforms.	2011	KONSTANTINOU, Ioannis et al.	Proceedings of the 20th ACM international conference on Information and knowledge management.
Big Data and cloud computing: current state and future opportunities	2011	Agrawal, Divyakant, Sudipto Das, and Amr El Abbadi	Proceedings of the 14th International Conference on Extending Database Technology. ACM
RDBMS to *NoSQL*_Reviewing Some Next-Generation Non-Relational Database's	2011	Padhy, Rabi Prasad, Manas Ranjan Patra, and Suresh Chandra Satapathy	International Journal of Advanced Engineering Science and Technologies
NoSQL na web 2.0: Um estudo comparativo de bancos não-relacionais para armazenamento de dados na web 2.0	2010	De Diana, Mauricio, and Marco Aurélio Gerosa	IX Workshop de Teses e Dissertações em Banco de Dados
Governo Eletrônico: Uma Visão sobre a Importância do Tema	2009	João Batista Ferri de Oliveira	Revista Eletônica Informática Pública Ano 11
Um Levantamento de Métodos de Avaliação de Arquiteturas de Software Específicas	2009	Lucas Bueno Ruas de Oliveira Elisa Yumi Nakagawa	3º Simpósio Brasileiro de Componentes, Arquiteturas e Reutilização de Software
O Governo Eletônico no Brasil: Perspectiva Histórica a Partir de um Modelo Estruturado de Análise	2008	Diniz, et. Al	Revista de Administração Pública

Sobre a Importância da Arquitetura de Software no Desenvolvimento de Sistemas de Software	2006	Antonio Mendes da Silva Filho	União dos Institutos Brasileiros de Tecnologia 52050-002 – Recife – PE – Brasil

APÊNDICE B - CENÁRIOS QUE A ARQUITETURA DEVE ATENDER, PRÉ-REQUISITOS DO AVALIADOR E PARECERES

Cenário	Pré-requisitos que o avaliador deve possuir para avaliar a arquitetura em relação ao cenário	Avaliação 1 - Atende 2 - Não atende 3 - Atende Parcialmente	Parecer Observações
C1 – Permitir a coleta de dados de diferentes fontes (sistemas, rede social, BDs, etc)	Formação acadêmica com especialização em bando de Dados/*Data Mining*/Análise de Sistemas Deter conhecimento acerca de dados não estruturados/BI (*Business Intelligence*)/ETL (extração, transformação e carga)/*Data Warehouse* Conhecer a Ferramenta Kafka/Storm/Talend/outra		
C2 – Prover o carregamento de dados de formatos diversos (texto, imagem, *logs*, arquivos, *streaming*, etc);	Formação acadêmica com especialização em bando de Dados/*Data Mining*/Análise de Sistemas/Desenvolvimento de Sistemas Deter conhecimento acerca de *NoSQL*/Hadoop/*MapReduce*/ETL (extração, transformação e carga)/*Clusters* Conhecer a Ferramenta Haddop HDFS/Apache Hbase/Hive/Cassandra/Spark/outras		
C3 – Permitir armazenamento em larga	Formação acadêmica com especialização em bando de Dados/*Data Mining*/Análise de Sistemas/Desenvolvimento de Sistemas		

escala (*Big Data*)	Deter conhecimento acerca de *Big Data*/*NoSQL*/Hadoop/*MapReduce*/ETL (extração, transformação e carga)/*Clusters* Conhecer a Ferramenta Haddop HDFS/Apache Hbase/Hive/Cassandra/MongoDB/CouchDB/Spark/outras		
C4 – Permitir a analise e transformação de dados de diferentes formatos	Formação acadêmica com especialização em bando de Dados/*Data Mining*/Análise de Sistemas/Desenvolvimento de Sistemas Deter conhecimento acerca de *Big Data*/*NoSQL*/Hadoop/*MapReduce*/ETL (extração, transformação e carga)/ Data Center/Virtualização/*Cloud Computing* Conhecer a Ferramenta PIG/Haddop HDFS/Apache Hbase/Hive/Cassandra/Spark/outras		
C5 – Permitir escalabilidade	Formação acadêmica com especialização em bando de Dados/*Data Mining*/Análise de Sistemas/Desenvolvimento de Sistemas Deter conhecimento acerca de *Big Data*/*NoSQL*/Hadoop/*MapReduce*/ETL (extração, transformação e carga)/*Clusters*/*Data Center*/Virtualização/*Cloud Computing* Conhecer a Ferramenta PIG/Haddop HDFS/Apache Hbase/Hive/Cassandra/outras		
C6 – Fornecer mecanismo para tolerância a falhas	Formação acadêmica com especialização em bando de Dados/*Data Mining*/Análise de Sistemas/Desenvolvimento de Sistemas Deter conhecimento acerca de *Big Data*/*NoSQL*/Hadoop/*MapReduce*/ETL (extração, transformação e carga)/*Clusters*/*Data Center*/Virtualização/Cloud Computing Conhecer a Ferramenta PIG/Haddop		

	HDFS/Apache Hbase/Hive/Cassandra/Spark/outras		
C7 – Fornecer suporte para serviços de *Cloud*	Formação acadêmica com especialização em bando de Dados/*Data Mining*/Análise de Sistemas/Desenvolvimento de Sistemas Deter conhecimento acerca de *Big Data*/*NoSQL*/Hadoop/*MapReduce*/ETL (extração, transformação e carga)/*Clusters*/*Data Center*/Virtualização/*Cloud Computing* Conhecer a Ferramenta PIG/Haddop HDFS/Apache Hbase/Hive/Cassandra/outras		
C8 – Permitir a visualização dos dados transformados	Formação acadêmica com especialização em bando de Dados/*Data Mining*/Análise de Sistemas/Desenvolvimento de Sistemas Deter conhecimento acerca de *Big Data*/*NoSQL*/Hadoop/*MapReduce*/ETL (extração, transformação e carga)/*Clusters*/*Data Center*/Virtualização/*Cloud Computing* Conhecer a Ferramenta PIG/Haddop HDFS/Apache Hbase/Hive/Cassandra/Tableau/Talend/Microsoft Office/outras		

Orientações:

O avaliador deve preencher o quadro acima, tomando por base a arquitetura de referência para tratamento de dados não estruturados, sob uma perspectiva de realidade do IFFar (Instituto Federal de Ciência Tecnologia e Educação Farroupilha).

O avaliador deve possuir conhecimento (literário ao menos) acerca dos pré-requisitos, não necessariamente todos os citados, mas os inerentes ao que se propõe o cenário, visto que alguns são ambíguos.

O avaliador, segundo sua análise, irá afirmar se a arquitetura de referência atende (1,2,3) o cenário proposto.

O avaliador poderá tecer parecer ou opiniões acerca de cada cenário em relação à arquitetura. Exemplificando ferramentas adjacentes ou concorrentes, por exemplo, ou ainda questionando ou sugerindo qualquer ação, prática ou situação.

O avaliador deve ler previamente o resumo a seguir exposto, para um melhor entendimento do método de avaliação e dos propósitos desta.