



Universidade do Minho
Escola de Engenharia

Carlos Filipe Machado da Silva Costa

BASIS: Uma Arquitetura de
Big Data para Smart Cities

Carlos Filipe Machado da Silva Costa
BASIS: Uma Arquitetura de
Big Data para Smart Cities



Universidade do Minho
Escola de Engenharia

Carlos Filipe Machado da Silva Costa

BASIS: Uma Arquitetura de Big Data para Smart Cities

Dissertação de Mestrado
Ciclo de Estudos Integrados Conducentes ao
Grau de Mestre em Gestão de Sistemas de Informação

Trabalho efetuado sob a orientação da
Professora Doutora Maribel Yasmina Santos

DECLARAÇÃO

Nome: Carlos Filipe Machado da Silva Costa

Endereço electrónico: a61555@alunos.uminho.pt Telefone: 914026349 / 253561296

Número do Bilhete de Identidade: 14150726

Título dissertação: BASIS: Uma Arquitetura de *Big Data* para *Smart Cities*

Orientadora: Professora Doutora Maribel Yasmina Santos

Ano de conclusão: 2015

Designação do Mestrado: Mestrado Integrado em Engenharia e Gestão de Sistemas de Informação

É AUTORIZADA A REPRODUÇÃO INTEGRAL DESTA TESE/TRABALHO
APENAS PARA EFEITOS DE INVESTIGAÇÃO, MEDIANTE
DECLARAÇÃO ESCRITA DO INTERESSADO, QUE A TAL SE
COMPROMETE.

Universidade do Minho, 31/07/2015

Assinatura: Carlos Costa

"Here's to the crazy ones. The misfits. The rebels. The troublemakers. The round pegs in the square holes. The ones who see things differently. They're not fond of rules. And they have no respect for the status quo. You can quote them, disagree with them, glorify or vilify them. About the only thing you can't do is ignore them. Because they change things. They push the human race forward. And while some may see them as the crazy ones, we see genius. Because the people who are crazy enough to think they can change the world, are the ones who do."

(Apple, 1997)

Agradecimentos

Como qualquer longa jornada, esta não foi realizada sozinho. Por conseguinte, aproveito a oportunidade para agradecer às pessoas que me marcaram, moldaram e motivaram, cada uma com a sua forma peculiar de pensar e agir.

Aos meus pais, sobretudo à minha mãe, pela razão óbvia da existência, pela educação e pelo investimento de significativos recursos financeiros na minha formação académica. Aos meus avós, por cada carinho e por cada momento de preocupação, apoiando sempre os meus pais na minha educação. Aos meus tios e primos, por trazerem alegria à família, por cada gargalhada... cada traquinice.

Aos meus amigos, “simplesmente” por estarem sempre presentes, no bom e no mau, com as palavras certas nos momentos de adversidade. Amigos de curso, uma pedra basilar de qualquer percurso académico, a malta dos Sistemas de Informação. Elos criados e jamais esquecidos, relações cujo significado transcende as paredes de qualquer instituição, uma família... “um amor de mar e margem”.

À Universidade do Minho, Escola de Engenharia, Departamento de Sistemas de Informação e Centro ALGORITMI, porque me acolheram nesta “mui nobre academia minhota”, colocando ao meu dispor a melhor qualidade de ensino e investigação do país, assegurado por docentes de notável conhecimento e excelência. Foi nesta instituição que me encontrei, é nela que termino um percurso... é nela que pretendo começar outro.

Sinto o dever de agradecer duplamente à Universidade do Minho, pois permitiu que conhecesse a pessoa que mais amo na vida... Carina, obrigado por seres quem és, pelo teu apoio incondicional nos dias em que o fardo de ser bem sucedido pesava na minha boa disposição. Quero agradecer-te por cada gesto, cada sorriso, cada palavra, por todos os atos simbólicos que me permitem olhar para trás e perguntar o que aconteceria se nunca te tivesse conhecido.

Reservo um agradecimento especial para a professora Maribel Santos, por inúmeras razões cuja transposição para texto implicaria um acréscimo surreal no número de páginas deste documento. A verdade é que não tenho como agradecer-lhe o suficiente. Docente de inigualável conhecimento, profissionalismo, entrega e simpatia. Devo-lhe cada erro corrigido, cada experiência partilhada, cada depósito de confiança, cada nota ou comentário positivo. Posso afirmar, com toda a certeza, que sei o que sou hoje e o que quero ser devido a si. Espero um dia poder retribuir toda a dedicação que colocou nesta dissertação, garantindo que me esforçarei para representar com orgulho tudo o que me ensinou.

Resumo

No mundo atual, os centros urbanos são a principal escolha para habitação. Cidadãos comportam-se como consumidores de serviços, esperando o desempenho adequado por parte dos órgãos gestores das cidades, independentemente das restrições vividas. Com este fenómeno advêm desafios de sustentabilidade económica, social e ambiental. É expectável que as cidades virem o seu foco para o cidadão e o enquadrem num governo participativo. Emerge assim o conceito de *Smart City*, onde a dinâmica humana atual assenta no uso intensivo da *Internet of Things (IoT)*, em que vários dispositivos se encontram permanentemente conectados, gerando vastas quantidades de dados, nos mais variados formatos. As *Smart Cities* passam a ser vistas como uma fonte incessável de *Big Data* e um potencial campo de aplicação para as tecnologias de armazenamento e processamento deste emergente tipo de dados, definido sobretudo pelo volume, variedade e velocidade. Nesta dissertação propõe-se uma arquitetura de *Big Data* para *Smart Cities* (BASIS), cujo contributo foi suportado pela realização de um adequado enquadramento conceptual e tecnológico, onde são estudados os conceitos envolvidos nesta dissertação, as principais abordagens existentes entre o conjunto de publicações científicas e as tecnologias de *Big Data* que podem integrar o leque de componentes tecnológicos da arquitetura. Após este enquadramento, realizou-se a especificação da arquitetura, dando particular atenção à criação de várias camadas de abstração, desde a mais conceptual à mais tecnológica, colmatando assim uma das principais lacunas identificadas na literatura, a falta de detalhe tecnológico. Posteriormente, prestou-se particular atenção à disponibilização pública dos dados, outra questão habitualmente pouco detalhada em outras abordagens, permitindo aqui o foco no Portal *Open (Big) Data* e na forma de integração com outras plataformas de acesso aos dados. Por fim, validou-se a arquitetura BASIS através da criação de dois serviços inteligentes no âmbito de uma *Smart City*: o primeiro serviço consiste no armazenamento e processamento do consumo de eletricidade e gás de 237 casas, de modo a criar perfis de consumo e a prever futuros consumos de eletricidade, comparando cada casa com o perfil em que é inserida; o segundo serviço baseia-se na identificação de perfis de atraso nos voos em certas companhias, aeroportos ou regiões, usando *Data Mining* distribuído para encontrar *clusters* em cerca de 18.000.000 de voos. Os resultados produzidos pelo desenvolvimento destes serviços revelam uma arquitetura capaz de armazenar, processar e disponibilizar *Big Data* no contexto de *Smart Cities*, incluindo modelos de *Data Mining (clustering e time series forecasting)* com desempenhos adequados e reduzidas margens de erro.

Palavras-Chave: *Big Data*, *Smart City*, Arquiteturas de *Big Data*, *NoSQL*, *Hadoop*, *Analytics*.

Abstract

In today's world, urban centers are the main choice for living. Citizens behave like service consumers, expecting adequate performance from cities' government, despite current restrictions. With this phenomenon come economical, social and environmental sustainability challenges. It is expected that cities turn their attention to the citizen, involving him in a participatory government. Thus emerges the Smart City concept, where the actual human dynamic is based on the intensive use of the Internet of Things (IoT), in which multiple devices are permanently connected, generating vast amounts of data, in various formats. Smart Cities are seen as a continuous Big Data source and a potential application field for the storage and processing technologies used within this emerging type of data, mainly defined by the volume, variety and velocity. In this dissertation it is proposed a Big Data architecture in the context of Smart Cities (BASIS), whose contribution was supported by the realization of an adequate conceptual and technological framework, in order to study the concepts present in this dissertation, the main approaches discussed in the scientific publications and the Big Data technologies that can integrate the set of technological components present in the architecture. After this, the architecture was specified, paying particular attention to the creation of multiple abstraction layers, from the most conceptual to the most technological, eliminating one of the main gaps identified in the literature, the lack of appropriate technological detail. Afterwards, particular attention was given to the public availability of data, another issue that was not as detailed as needed in other approaches, focusing specifically on the Open (Big) Data Portal and on the integration with other data access platforms. Lastly, the architecture was validated through the development of two intelligent services in the context of a Smart City: the first service consists in the storage and processing of electricity and gas consumption of 237 homes, in order to create consumption profiles and to forecast electricity consumptions, comparing each home with the profile in which it is inserted; the second one is based on the identification of delay profiles on flights in certain companies, airports or regions, making use of distributed Data Mining in order to find clusters in about 18.000.000 flights. The results produced by the development of these services reveal an architecture capable of storing, processing and make available Big Data in the context of Smart Cities, including advanced analytics as Data Mining models (clustering and time series forecasting) with adequate performances and a small margin of error.

Keywords: Big Data, Smart City, Big Data Architectures, NoSQL, Hadoop, Analytics.

Índice

AGRADECIMENTOS	IV
RESUMO	V
ABSTRACT	VI
ÍNDICE	VII
LISTA DE TABELAS	IX
LISTA DE FIGURAS	X
LISTA DE ABREVIATURAS E SIGLAS	XII
CAPÍTULO 1. INTRODUÇÃO	1
1.1 ENQUADRAMENTO E PROBLEMA DE INVESTIGAÇÃO IDENTIFICADO	1
1.2 ABORDAGEM METODOLÓGICA	2
1.2.1 Fases da Abordagem Metodológica	3
1.2.2 Processo de Estruturação do Enquadramento Conceptual	4
1.3 FINALIDADE E OBJETIVOS DA PRESENTE INVESTIGAÇÃO	5
1.4 ESTRUTURAÇÃO DO DOCUMENTO	6
CAPÍTULO 2. ENQUADRAMENTO CONCEPTUAL	8
2.1 UM OLHAR SOBRE O <i>BIG DATA</i>	8
2.1.1 O Conceito	8
2.1.2 Porquê Toda a Azáfama? – Motivação e Importância do <i>Big Data</i>	13
2.1.3 Com Grandes Dados Advêm Grandes Desafios	15
2.2 PARADIGMAS E TERMOS ASSOCIADOS AO <i>BIG DATA</i>	22
2.2.1 Processamento Tradicional de Dados vs. Processamento de <i>Big Data</i>	22
2.2.2 Requisitos do Processamento de <i>Big Data</i>	24
2.2.3 Os Fundamentos do <i>MapReduce</i>	26
2.2.4 <i>NoSQL</i>	28
2.3 <i>SMART CITIES</i>	31
2.3.1 O que é uma <i>Smart City</i> ?	31
2.3.2 Desafios Associados a Iniciativas de <i>Smart Cities</i>	33
2.4 ARQUITETURAS DE <i>BIG DATA</i> NO CONTEXTO DE <i>SMART CITIES</i>	34
2.4.1 “ <i>Smart City Dashboard for Integrating Various Data of Sensor Networks</i> ”	35
2.4.2 “ <i>Big Data in Large Scale Intelligent Smart City Installations</i> ”	36
2.4.3 “ <i>Cloud based Big Data Analytics for Smart Future Cities</i> ”	38
2.4.4 “ <i>Bootstrapping Smart Cities through a Self-Sustainable Model Based on Big Data Flows</i> ”	39
2.5 MODELO DE CONCEITOS E OBSERVAÇÕES AO ENQUADRAMENTO CONCEPTUAL	42
CAPÍTULO 3. ENQUADRAMENTO TECNOLÓGICO	44
3.1 HADOOP	44
3.2 BASES DE DADOS <i>NoSQL</i> AO ABRIGO DE LICENÇAS <i>OPEN SOURCE</i>	48
3.3 O PAPEL DAS TECNOLOGIAS DE <i>ETL</i> NO PROCESSAMENTO DE <i>BIG DATA</i>	51
3.4 <i>BIG DATA ANALYTICS</i>	52
3.4.1 Técnicas e Ferramentas de <i>Data Mining</i>	52
3.4.2 <i>Data Visualization</i>	54

CAPÍTULO 4. BASIS: UMA ARQUITETURA DE <i>BIG DATA</i> PARA <i>SMART CITIES</i>	55
4.1 CAMADA CONCEPTUAL.....	57
4.2 CAMADA TECNOLÓGICA.....	62
4.3 CAMADA INFRAESTRUTURAL	66
4.4 INTERVENIENTES NA COCRIAÇÃO DE SERVIÇOS.....	69
4.4.1 Camada Tecnológica vs. Intervenientes na Cocriação de Serviços	71
4.4.2 Camada Infraestrutural vs. Intervenientes na Cocriação de Serviços	71
CAPÍTULO 5. PORTAL <i>OPEN (BIG) DATA</i> DA ARQUITETURA BASIS	73
5.1 FUNCIONALIDADES.....	74
5.2 INTEGRAÇÃO COM OUTROS COMPONENTES E APLICAÇÕES EXTERNAS	77
5.3 PROTÓTIPO DE INTERFACE GRÁFICA	78
CAPÍTULO 6. VALIDAÇÃO DA ARQUITETURA BASIS: SERVIÇOS INTELIGENTES EM <i>SMART CITIES</i>	81
6.1 SERVIÇO PARA O CONSUMO INTELIGENTE DE ENERGIA.....	81
6.1.1 Análise e Previsão de Consumos Energéticos	82
6.1.2 Visão Geral da Camada Tecnológica e dos Dados	83
6.1.3 Preparação dos Dados e <i>Data Mining</i>	85
6.1.4 Análise e Visualização de Dados: Reinventando a Fatura de Energia para um Consumo mais Inteligente	91
6.1.5 Síntese.....	96
6.2 SERVIÇO DE AEROPORTOS INTELIGENTES	97
6.2.1 Identificação de Perfis de Atraso nos Voos	98
6.2.2 Análise dos Dados e Respetiva Qualidade	98
6.2.3 Instanciação da Camada Tecnológica	100
6.2.4 Instanciação da Camada infraestrutural.....	102
6.2.5 <i>Clustering</i>	104
6.2.6 Visualização e Disponibilização dos Resultados	106
6.2.7 Síntese.....	108
CAPÍTULO 7. CONCLUSÕES E TRABALHO FUTURO	110
7.1 PORQUÊ <i>BIG DATA</i> ? IMPLICAÇÕES A LONGO PRAZO.....	110
7.2 RESULTADOS OBTIDOS	112
7.3 INVESTIGAÇÃO FUTURA	114
REFERÊNCIAS BIBLIOGRÁFICAS.....	116

Lista de Tabelas

TABELA 1. DESAFIOS DO DESENVOLVIMENTO DE APLICAÇÕES <i>BIG DATA</i> . ADAPTADO DE (M. CHEN ET AL., 2014).....	18
TABELA 2. DIFERENÇAS ENTRE UMA CIDADE TRADICIONAL E UMA <i>SMART CITY</i> . ADAPTADO DE (HEDLUND, 2013).....	33
TABELA 3. COMPONENTES DE DESTAQUE DO HADOOP. BASEADO EM (APACHE, 2015B; HORTONWORKS, 2014).	47
TABELA 4. DESCRIÇÃO DE ALGUMAS FERRAMENTAS DE <i>DATA MINING</i>	53
TABELA 5. FORMAS DE EXTRAÇÃO E CARREGAMENTO DOS DADOS.	58
TABELA 6. TIPOS DE DADOS ASSOCIADOS ÀS TECNOLOGIAS DE ARMAZENAMENTO DA ARQUITETURA BASIS.	63
TABELA 7. CRUZAMENTO DA CAMADA TECNOLÓGICA COM OS INTERVENIENTES NA COCRIAÇÃO DE SERVIÇOS.	71
TABELA 8. CRUZAMENTO DA CAMADA INFRAESTRUTURAL COM OS INTERVENIENTES NA COCRIAÇÃO DE SERVIÇOS.	72
TABELA 9. DESCRIÇÃO DOS ATRIBUTOS USADOS NA ANÁLISE DOS ATRASOS NOS VOOS.	99
TABELA 10. CENTRÓIDES DOS <i>CLUSTERS</i> DE ATRASOS NOS VOOS.	106

Lista de Figuras

FIGURA 1. <i>DSRM FOR INFORMATION SYSTEMS</i> . RETIRADO DE (PEFFERS ET AL., 2008).....	2
FIGURA 2. PROCESSO DE SELEÇÃO DA LITERATURA A INCLUIR NO ENQUADRAMENTO CONCEPTUAL.	5
FIGURA 3. O FENÓMENO DO <i>BIG DATA</i> . EXTRAÍDO DE (M. CHEN ET AL., 2014).....	10
FIGURA 4. REPRESENTAÇÃO CONCEPTUAL DO MODELO DOS 3Vs. RETIRADO DE (ZIKOPOULOS & EATON, 2011).	11
FIGURA 5. MODELO DOS 3Vs AUMENTANDO COM CARACTERÍSTICAS ORIGINÁRIAS DO CRUZAMENTO ENTRE VOLUME, VARIEDADE E VELOCIDADE. RETIRADO DE (KRISHNAN, 2013).....	12
FIGURA 6. MODELO DOS 5Vs. RETIRADO DE (CHANDARANA & VIJAYALAKSHMI, 2014).....	13
FIGURA 7. POPULARIDADE DO TERMO <i>BIG DATA</i> , SEGUNDO O GOOGLE TRENDS. RETIRADO DE (TRENDS, 2015).	13
FIGURA 8. VÁRIAS ÁREAS DE APLICAÇÃO EM QUE O <i>BIG DATA</i> PODE DESEMPENHAR UM PAPEL FUNDAMENTAL. BASEADO EM (CHANDARANA & VIJAYALAKSHMI, 2014; SAGIROGLU & SINANC, 2013).....	15
FIGURA 9. REQUISITOS DE PRIVACIDADE E TÉCNICAS EXISTENTES. INFORMAÇÃO RETIRADA DE (LU ET AL., 2014).	21
FIGURA 10. CICLO DE PROCESSAMENTO TRADICIONAL DE DADOS. RETIRADO DE (KRISHNAN, 2013).....	22
FIGURA 11. CICLO DE PROCESSAMENTO DE <i>BIG DATA</i> . RETIRADO DE (KRISHNAN, 2013).	23
FIGURA 12. PROCESSO DE EXTRAÇÃO DE CONHECIMENTO. ADAPTADO DE (PHILIP CHEN & ZHANG, 2014).....	23
FIGURA 13. FLUXO DE PROCESSAMENTO DE <i>BIG DATA</i> . RETIRADO DE (KRISHNAN, 2013).	24
FIGURA 14. REQUISITOS DE UMA INFRAESTRUTURA <i>BIG DATA</i> . BASEADO EM (KRISHNAN, 2013).	25
FIGURA 15. REQUISITOS DO PROCESSAMENTO DE <i>BIG DATA</i> . BASEADO EM (KRISHNAN, 2013).....	26
FIGURA 16. BREVE EXPLICAÇÃO DOS FUNDAMENTOS DO <i>MAPREDUCE</i> . BASEADO EM (SCHNEIDER, 2013).	27
FIGURA 17. POSSÍVEIS DOMÍNIOS DE ATUAÇÃO NUMA <i>SMART CITY</i> . ADAPTADO DE (SCHAFFERS ET AL., 2011).	32
FIGURA 18. ARQUITETURA PROPOSTA POR SUAKANTO ET AL. (2013).	36
FIGURA 19. ARQUITETURA PROPOSTA POR GIRTELSCHMID ET AL. (2013).....	37
FIGURA 20. ARQUITETURA PROPOSTA POR KHAN ET AL. (2013). ADAPTADO.	38
FIGURA 21. ARQUITETURA PROPOSTA POR VILAJOSANA ET AL. (2013).....	40
FIGURA 22. CADEIA DE VALOR PROPOSTA POR VILAJOSANA ET AL. (2013).	41
FIGURA 23. MAPA DE CONCEITOS DO ENQUADRAMENTO CONCEPTUAL.	42
FIGURA 24. ARQUITETURA DO HDFS. RETIRADO DE (KRISHNAN, 2013).	45
FIGURA 25. COMPONENTES DE DESTAQUE DO HADOOP. BASEADO EM (APACHE, 2015B; HORTONWORKS, 2014).	46
FIGURA 26. CAMADA CONCEPTUAL DA ARQUITETURA BASIS.	57
FIGURA 27. CAMADA TECNOLÓGICA DA ARQUITETURA BASIS.....	62
FIGURA 28. CAMADA INFRAESTRUTURAL DA ARQUITETURA BASIS.....	66

FIGURA 29. INTERVENIENTES NA COCRIAÇÃO DE SERVIÇOS.	69
FIGURA 30. FUNCIONALIDADES DO PORTAL <i>OPEN (BIG) DATA</i> PARA A DISPONIBILIZAÇÃO DE ACESSO E DE DADOS.	74
FIGURA 31. FUNCIONALIDADES "AS A SERVICE" DO PORTAL <i>OPEN (BIG) DATA</i>	75
FIGURA 32. FLUXO DE DADOS ENTRE O PORTAL <i>OPEN (BIG) DATA</i> E OUTROS COMPONENTES OU APLICAÇÕES EXTERNAS.	77
FIGURA 33. PROTÓTIPO DO PORTAL <i>OPEN (BIG) DATA</i> - PÁGINA INICIAL.	78
FIGURA 34. PROTÓTIPO DO PORTAL <i>OPEN (BIG) DATA</i> - PÁGINA DE DISPONIBILIZAÇÃO DE CONJUNTOS DE DADOS.	79
FIGURA 35. PROTÓTIPO DO PORTAL <i>OPEN (BIG) DATA</i> - PÁGINA DO <i>IaaS</i>	79
FIGURA 36. CAMADA TECNOLÓGICA UTILIZADA NO SERVIÇO PARA O CONSUMO INTELIGENTE DE ENERGIA. ADAPTADO DE (COSTA & SANTOS, 2015).	84
FIGURA 37. MODELO DE DADOS DO CONJUNTO DE DADOS ORIGINAL. ADAPTADO DE (COSTA & SANTOS, 2015).	85
FIGURA 38. SCRIPT PIG DE PREPARAÇÃO DOS DADOS PARA <i>CLUSTERING</i> . RETIRADO DE (COSTA & SANTOS, 2015).	86
FIGURA 39. TRANSFORMAÇÃO DOS DADOS PARA PROPÓSITOS DE <i>CLUSTERING</i>	87
FIGURA 40. SIMILARIDADE <i>INTRA-CLUSTER</i> PARA CADA NÚMERO DE <i>CLUSTERS</i> (VALORES MAIS BAIXOS SIGNIFICAM MELHOR SIMILARIDADE <i>INTRA-CLUSTER</i>). RETIRADO DE (COSTA & SANTOS, 2015).	88
FIGURA 41. SÍNTESE DO MODELO DE <i>CLUSTERING</i> (VALORES DE ELETRICIDADE E GÁS ENCONTRAM-SE EM <i>KW/H</i> ; O NÚMERO DE CIDADES EM CADA <i>CLUSTER</i> ENCONTRA-SE SOB O NÚMERO DO <i>CLUSTER</i>). RETIRADO DE (COSTA & SANTOS, 2015).	88
FIGURA 42. AVALIAÇÃO DO <i>FORECASTING</i> DE ELETRICIDADE. ADAPTADO DE (COSTA & SANTOS, 2015).	90
FIGURA 43. DEMONSTRAÇÃO DOS ERROS DO MODELO M5P TREE, SELECIONADO PARA O <i>CLUSTER 5</i> , CUJAS PREVISÕES PERMITEM OBTER UM <i>MAE</i> SIGNIFICATIVAMENTE REDUZIDO. ADAPTADO DE (COSTA & SANTOS, 2015).	90
FIGURA 44. MODELO M5P TREE, SELECIONADO PARA O <i>CLUSTER 5</i> . ADAPTADO DE (COSTA & SANTOS, 2015).	91
FIGURA 45. AMOSTRA DO CONJUNTO DE DADOS FINAL PARA ANÁLISE VISUAL. RETIRADO DE (COSTA & SANTOS, 2015).	92
FIGURA 46. FATURA DE ENERGIA REINVENTADA. RETIRADO DE (COSTA & SANTOS, 2015).	93
FIGURA 47. DEMONSTRAÇÃO DA EXPERIÊNCIA DE UTILIZAÇÃO ASSOCIADA À FATURA DE ENERGIA REINVENTADA.	94
FIGURA 48. MONITORIZAÇÃO E SEGMENTAÇÃO DO CONSUMO DE ENERGIA. RETIRADO DE (COSTA & SANTOS, 2015). ...	95
FIGURA 49. CAMADA TECNOLÓGICA INSTANCIADA PARA O SERVIÇO DE AEROPORTOS INTELIGENTES.	100
FIGURA 50. EXTRAÇÃO E CARREGAMENTO DOS DADOS DOS VOOS.	101
FIGURA 51. INFRAESTRUTURA UTILIZADA NO SERVIÇO DE AEROPORTOS INTELIGENTES.	103
FIGURA 52. PÁGINA DE MONITORIZAÇÃO DO <i>CLUSTER SPARK</i>	104
FIGURA 53. SIMILARIDADE <i>INTRA-CLUSTER</i> NOS ATRASOS DOS VOOS.	105
FIGURA 54. VISUALIZAÇÃO DOS PERFIS DE ATRASO NOS VOOS.	107
FIGURA 55. VISUALIZAÇÃO, NUM DISPOSITIVO MÓVEL, DOS PERFIS DE ATRASO NOS VOOS.	108

Lista de Abreviaturas e Siglas

Neste documento são utilizadas diversas abreviaturas e siglas, sistematizadas de seguida:

ACID - *Atomicity, Consistency, Isolation, Durability*;

APaaS - *Analytical Processing as a Service*;

API - *Application Programming Interface*;

BASE - *Basic Availability, Soft State, Eventual Consistency*;

CAP - *Consistency, Availability, Partition tolerance*;

CRAN - *Comprehensive R Archive Network*;

DSRM - *Design Science Research Methodology*;

DSS - *Decision Support Systems*;

ELT - *Extração, Carregamento e Transformação (Extraction, Loading, Transformation)*;

ETL - *Extração, Transformação e Carregamento (Extraction, Transformation and Loading)*;

HDFS - *Hadoop Distributed File System*;

HTTP - *Hypertext Transfer Protocol*;

IaaS - *Intelligence as a Service*;

IoT - *Internet of Things*;

JDBC - *Java Database Connectivity*;

JSON - *JavaScript Object Notation*;

KDD - *Knowledge Discovery from Databases*;

NoSQL - *Not Only Structured Query Language*;

OCG - *Open Geospatial Consortium*;

ODBC - *Open Database Connectivity*;

RAM - *Random Access Memory*;

RDF - *Resource Description Framework*;

REST - *Representational State Transfer*;

SIG - *Sistema de Informação Geográfica*;

SQL - *Structured Query Language*;

TI - *Tecnologias da Informação*;

TIC - *Tecnologias da Informação e Comunicação*;

VaaS - *Visualization as a Service*.

Capítulo 1. Introdução

Neste capítulo é apresentado o enquadramento e motivação para a realização desta dissertação, assim como a abordagem metodológica, a finalidade e os objetivos e a estruturação do documento.

1.1 Enquadramento e Problema de Investigação Identificado

Devido à enorme evolução liderada pelo avanço nas Tecnologias da Informação e Comunicação (TIC), as pessoas e coisas que as rodeiam estão cada vez mais interconectadas (Zikopoulos & Eaton, 2011). Dispondo de novos dispositivos como uma extensão das capacidades cognitivas humanas, verifica-se a tendência a armazenar, constantemente, um maior volume e variedade de dados. O termo *Big Data* surge para caracterizar esses mesmos dados, tornando inevitável o aparecimento de novas técnicas e tecnologias para lidar com a sua natureza volumosa, variada e veloz.

Várias organizações detêm o acesso a dados, cuja potencial informação a retirar pode alterar radicalmente o seu comportamento, funcionando como uma alavanca impulsionadora da dinâmica organizacional. No entanto, as mesmas não conseguem retirar valor desses dados, pois estes encontram-se armazenados nas mais variadas formas e com as mais distintas características. Uma *Smart City* pode ser vista como uma dessas organizações, onde os conceitos-chave passam pela sustentabilidade económica, social e ambiental, cujo consumidor é o cidadão e a oferta são os serviços. A dinâmica destas cidades é focada no cidadão, envolvendo-o no processo governativo das mesmas.

Smart Cities oferecem uma nova abordagem para a otimização de serviços, redução de custos, sustentabilidade a longo prazo, e desenvolvimento de novos serviços para os cidadãos (IBM, 2014). Neste contexto, são recolhidos dados que provêm de diversas fontes, em formatos estruturados ou não, e a disponibilização de novos serviços de informação é largamente influenciada pela capacidade de integrar, processar e analisar estes dados (M. Chen, Mao, & Liu, 2014).

De modo a que uma cidade seja capaz de providenciar um espaço inteligente e sustentável, adaptando-se à dinâmica dos seus residentes, esta tem que processar vastos conjuntos de dados relevantes, podendo usufruir para isso das oportunidades proporcionadas pela imensa rede de dispositivos conectados, aquilo que conhecemos atualmente como a *Internet of Things (IoT)* (Jara, Bocchi, & Genoud, 2013).

Toda esta necessidade de consumir dados para retirar valor e produzir novos serviços traz consigo desafios significativos, que podem ser ultrapassados pela adoção de uma arquitetura de processamento de *Big Data* adequada, dando origem a novas aplicações (Schneider, 2013) e, consequentemente, a novos serviços que caracterizem uma cidade inteligente. Para tal, torna-se essencial o enquadramento conceptual e tecnológico da temática aqui investigada, incluindo a análise de iniciativas arquiteturais já existentes, de modo a identificar as suas características e limitações, permitindo assim evoluir para a especificação de uma nova arquitetura de *Big Data* no contexto de *Smart Cities*.

1.2 Abordagem Metodológica

Estando esta dissertação enquadrada na área de Sistemas de Informação, a adoção da *Design Science Research Methodology (DSRM) for Information Systems* (Vaishnavi & Kuechler, 2013) revela-se adequada para a realização da mesma. Tendo em conta o enquadramento e objetivos da investigação que irá ser conduzida, optar-se-á pelo modelo desenvolvido por Peffers, Tuunanen, Rothenberger, & Chatterjee (2008), ilustrado na Figura 1.

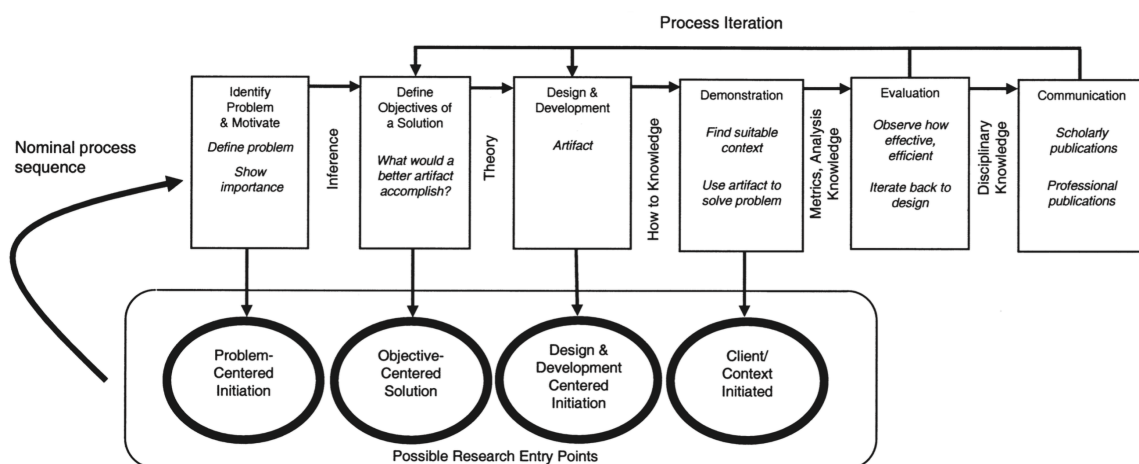


Figura 1. *DSRM for Information Systems*. Retirado de (Peffers et al., 2008).

A principal razão para a adoção deste modelo em particular deve-se à explicitação das principais fases da *DSRM*, bem como a definição clara dos resultados e fluxo do processo.

Este modelo é bastante flexível, permitindo que um projeto de investigação se inicie num de vários pontos de partida. Neste caso, iniciou-se uma investigação centrada no problema, percorrendo assim as 6 fases ilustradas na Figura 1.

1.2.1 Fases da Abordagem Metodológica

A dissertação teve início com a elaboração do plano de trabalhos, seguido do enquadramento conceptual em *Big Data*, *Smart Cities* e Arquiteturas de *Big Data* no contexto de *Smart Cities*. Em paralelo com esta última tarefa, decorreu a realização de um enquadramento tecnológico. Concluídas estas tarefas, iniciou-se o processo de investigação baseado na *Design Science Research Methodology for Information Systems* (Peppers et al., 2008). Por conseguinte, as fases associadas a esta dissertação são as seguintes:

1. Elaboração do plano de trabalhos – escrita de um documento inicial, onde consta a motivação/enquadramento, finalidade e objetivos iniciais, abordagem metodológica, descrição e calendarização das tarefas;
2. Enquadramento conceptual – identificar e rever literatura científica e técnica relevante, de modo a discriminar e relacionar conceitos, problemas e avanços relevantes para a realização da dissertação;
3. Enquadramento tecnológico – a sua realização é relevante para a familiarização com as tecnologias que estão envolvidas na dissertação;
4. Identificar o problema e motivar – definir o problema de investigação e justificar o valor da solução. Como a definição do problema é usada para providenciar uma solução, convém descrever o problema conceptualmente, para que a solução consiga capturar a complexidade do mesmo. Justificar o valor da solução consiste não só em motivar o investigador e partes interessadas a encontrar uma solução, mas também em ajudar a entender o raciocínio associado à percepção do problema por parte do investigador. O *input* principal desta tarefa é o conhecimento do estado do problema e a importância de encontrar uma solução;
5. Definir objetivos da solução – inferir os objetivos da solução através da definição do problema e do conhecimento acerca do que é possível e praticável. Estes podem ser quantitativos, quantificando os termos em que a solução pode ser considerada ideal, ou qualitativos, descrevendo como é que a solução pode resolver problemas ainda por endereçar. Tal como a tarefa anterior, esta tem como *input* o conhecimento do estado do problema, acrescido das soluções atuais e sua eficácia, caso existam algumas;
6. Concepção e desenvolvimento – criar o artefacto resultante da investigação, podendo conter modelos, métodos e instanciações desses mesmos modelos e métodos. Esta

atividade inclui a determinação das funcionalidades desejadas e a arquitetura da solução. O *input* desta tarefa consiste no conhecimento resultante de fundamentos teóricos e práticos para a construção da solução;

7. Demonstração – demonstrar o uso do artefacto produzido para resolver o problema, passando pela experimentação ou simulação. Esta tarefa tem como *input* o conhecimento acerca de como usar o artefacto para resolver o problema;
8. Avaliação – observar e medir quão bem o artefacto suporta a solução do problema, comparando os objetivos da solução com os resultados provenientes da demonstração. Requer conhecimento de métricas relevantes e técnicas de análise adequadas, como métricas de performance da solução, questionários de satisfação, feedback de partes interessadas, entre outros. No final desta atividade o investigador pode retomar a tarefa 6, de modo a otimizar o artefacto, ou continuar para a próxima tarefa e deixar a devida otimização para projetos futuros;
9. Comunicação – Comunicar todo o processo a investigadores e outras partes interessadas como profissionais da área, desde o problema e sua importância até ao artefacto, destacando a sua utilidade, rigor e eficiência. Requer conhecimento de oportunidades de comunicação científica ou técnica, bem como a noção de como comunicar um estudo desta natureza. Todo o processo de redação do documento de dissertação é também contemplado nesta tarefa.

1.2.2 Processo de Estruturação do Enquadramento Conceptual

No enquadramento conceptual existiram limitações contextuais e temporais. É de salientar que a literatura revista corresponde a um subconjunto da interseção entre o conjunto de toda a literatura passível de ser analisada e a literatura disponível para a comunidade académica da Universidade do Minho. Para além disto, como referido, este enquadramento encontra-se limitado temporalmente, nomeadamente a artigos com data igual ou superior a 2010, salvo raras exceções referentes a autores ou artigos relevantes na área.

No que respeita aos serviços de indexação utilizados, identificam-se os seguintes: Google Scholar; Scopus; Web of Science; IEEE Xplore; ACM Digital Library; Springer Open; Elsevier's Science Direct; Emerald Insight; RepositóriUM; Google (uso exclusivo para identificação de literatura de natureza técnica e/ou relacionada com conferências não indexadas nos serviços acima mencionados).

A identificação e seleção da literatura foram efetuadas entre 1 de novembro de 2014 e 15 de janeiro de 2015, utilizando as palavras-chave: *Big Data*, *NoSQL* e *Smart City*. Após este período, a procura de novas referências bibliográficas relevantes foi restrita a conceitos, técnicas ou tecnologias muito particulares, com o objectivo de complementar aspectos concretos no desenvolvimento que estava a ser realizado. É o caso da bibliografia dedicada aos serviços que foram desenvolvidos para validar a arquitetura proposta. Quanto à seleção da literatura a incluir no enquadramento conceptual, o fluxograma na Figura 2 ilustra o processo de tomada de decisão adotado.

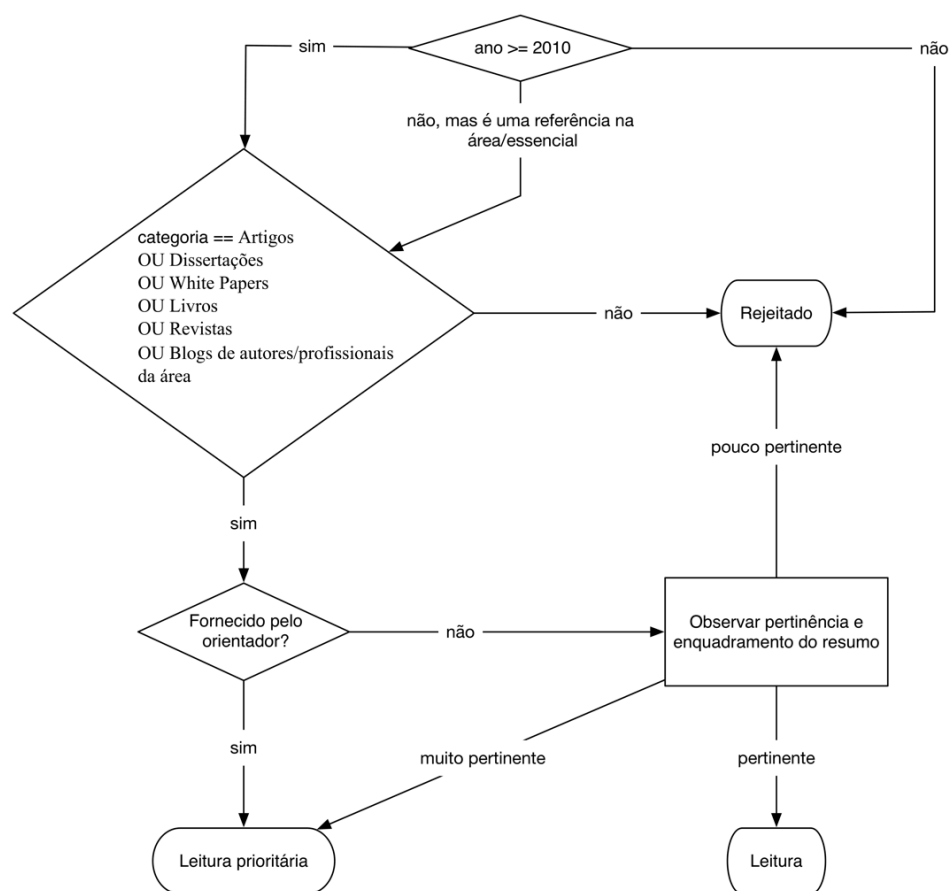


Figura 2. Processo de seleção da literatura a incluir no enquadramento conceptual.

1.3 Finalidade e Objetivos da Presente Investigação

Esta dissertação tem como finalidade estudar o papel do *Big Data* no contexto de *Smart Cities* e verificar as arquiteturas de suporte a estes sistemas, analisando o potencial e limitações de cada uma. Nesta análise deve ser dada particular importância à necessidade de integrar e disponibilizar dados de uma forma aberta (Lessig, 2007), facilitando o desenvolvimento de novos serviços de informação por parte da sociedade em geral (organizações ou cidadãos) (Schaffers et al., 2011). A análise das

arquiteturas já existentes tem como principal objetivo identificar as suas principais limitações de forma a ser possível propor uma nova arquitetura que contemple os seguintes princípios de concepção:

1. Disponibilizar dados de forma aberta, possibilitando o desenvolvimento de novos serviços, não só por iniciativa dos órgãos gestores de uma *Smart City*, mas também pelos cidadãos e organizações;
2. Contemplar várias camadas de abstração, desde a mais conceptual à mais tecnológica;
3. Garantir o armazenamento e processamento distribuído dos dados;
4. Incluir a noção de segurança, privacidade e confiança dos dados;
5. Incorporar formas de gerir o ciclo de vida dos dados, usando para isso conceitos inerentes à partição dos mesmos;
6. Estabelecer modos de cooperação entre as pessoas intervenientes no desenvolvimento de serviços de uma *Smart City*;
7. Ser *service-oriented* e independente do dispositivo ou plataforma que acede aos dados;
8. Fazer uso de tecnologias *open source*, exceto em casos onde a relação custo-benefício justifique outra escolha.

Para além do foco na arquitetura, espera-se ainda como objetivo deste trabalho, o desenvolvimento de serviços inovadores no contexto de uma *Smart City*, tirando partido da arquitetura proposta.

1.4 Estruturação do Documento

Este documento está estruturado em 7 capítulos, de seguida brevemente caracterizados. O capítulo 1 apresenta a introdução ao trabalho, identificando o problema de investigação, a abordagem metodológica e a finalidade e objetivos associados a esta dissertação. O capítulo 2 enquadra conceptualmente os termos *Big Data* e *Smart City*, terminando com a análise de arquiteturas, já existentes, de *Big Data* no contexto de *Smart Cities*. O capítulo 3 apresenta várias tecnologias e técnicas relevantes para o desenvolvimento desta investigação. O capítulo 4 apresenta a arquitetura proposta, através da especificação de várias camadas de abstração e dos constituintes de cada uma, quer em termos conceptuais, quer em termos tecnológicos. O capítulo 5 descreve um componente significativo da arquitetura, o Portal *Open (Big) Data*, identificando as suas principais funcionalidades e questões de integração com outros componentes. Devido à relevância da disponibilização pública de dados e da integração entre componentes da arquitetura proposta, dedicou-se um capítulo ao Portal *Open (Big) Data*,

que inclui a descrição adequada destas duas problemáticas. O capítulo 6 valida a arquitetura proposta, apresentando o desenvolvimento de dois serviços inteligentes no âmbito de uma *Smart City*, que tiram partido de armazenamento e processamento distribuído, onde se destacam as tecnologias de *Big Data* utilizadas, como por exemplo o HDFS, Pig, Hive, HBase, Weka, Spark, Cassandra, entre outros. O capítulo 7 conclui este documento, sistematizando o trabalho realizado e apresentando propostas de trabalho futuro.

Capítulo 2. Enquadramento Conceptual

Neste capítulo apresenta-se, primeiramente, um enquadramento quanto aos temas *Big Data* e *Smart Cities*, de modo sistematizar os conceitos associados a esta dissertação. Posteriormente, ainda neste capítulo, detalha-se a relação entre estes dois conceitos no que diz respeito às arquiteturas de *Big Data* no contexto de *Smart Cities*. Para tal, é utilizada literatura maioritariamente proveniente da comunidade científica, conjuntamente com publicações de cariz técnico, relacionadas com esta temática, com vista a estabelecer um enquadramento conceptual assente em diversas perspetivas partilhadas pelas duas partes, o que incondicionalmente contribui para a qualidade deste enquadramento.

O conteúdo presente neste capítulo visa contribuir para a sintetização dos conceitos associados ao *Big Data* e às *Smart Cities*, dissipando assim algumas dificuldades que possam surgir no estudo, planeamento ou implementação de um projeto neste contexto, por parte da comunidade científica ou técnica ligada a esta temática.

2.1 Um Olhar Sobre o *Big Data*

Quando se investiga uma temática tão abrangente quanto a de arquiteturas de *Big Data* no contexto de *Smart Cities*, é-se envolvido num conjunto interminável de conteúdos, começando logo à partida com *Big Data* como conceito isolado. Por conseguinte, esta secção visa explicar o conceito e sua importância, os desafios esperados, bem como os paradigmas que emergiram com o *Big Data*, destacando sobretudo o *MapReduce* e o *Not Only Structured Query Language (NoSQL)*.

2.1.1 O Conceito

Se é praticamente inédito integrar um mundo onde, de alguma forma, *Big Data* não esteja a ser gerado ou consumido algures à nossa volta, é também quase que impensável começar a descrever o conceito de *Big Data* sem salientar que o mesmo é uma abstração, pois não se trata apenas de um vasto volume de dados, mas sim de todo um conjunto de características que o definem (M. Chen et al., 2014).

Na atualidade, a importância do *Big Data* é amplamente reconhecida, no entanto existe ainda bastante ambiguidade no que diz respeito à sua definição. Segundo M. Chen et al. (2014), a generalidade das definições assenta em “*datasets* que não conseguem ser reconhecidos, adquiridos, geridos ou processados pelas tradicionais Tecnologias da Informação (TI) e *software/hardware* em tempo tolerável”, definição esta assente nos moldes daquilo que a equipa do Apache Hadoop, um dos principais

projetos *open source*, entende ser o *Big Data*. Aliás, a definição de *Big Data* remonta-nos a tempos tão distantes quanto o ano de 2001, quando Doug Laney, analista da presente Gartner apresentou o modelo dos 3Vs (volume, variedade e velocidade), modelo esse usado pela IBM e Microsoft durante, pelo menos, os 10 anos seguintes (M. Chen et al., 2014).

Segundo Krishnan (2013), *Big Data* pode ser definido como quantidades de dados disponíveis em vários graus de complexidade, gerados a diferentes velocidades e com vários graus de ambiguidade, que não se adequam a tecnologias, métodos, algoritmos tradicionais ou quaisquer outras soluções pré-configuradas. Estes dados podem ser provenientes das mais variadas áreas de negócio e existem, segundo o autor, vários departamentos de uma organização que são propícios a produzir dados desta natureza como, por exemplo, vendas e marketing.

Como *Big Data* não possui uma definição comum, ainda antes de explicar o modelo dos 3Vs, convinha salientar as tendências relacionadas que contribuíram para o aparecimento deste conceito (Schneider, 2013):

1. Capturar e gerir uma extensa quantidade de dados, estando o volume dos mesmos a duplicar a cada ano e as organizações a armazenarem três ou mais anos de dados históricos, como demonstram vários estudos;
2. Lidar com novos tipos de dados: estudos mostram que 80% dos dados são não estruturados, como mensagens de texto, publicações nas redes sociais, imagens, entre outros;
3. Explorar as duas tendências anteriores com novas tecnologias, pois grande parte do portefólio aplicacional não consegue acompanhar as necessidades apresentadas pelo volume e variedade dos dados.

De forma a compreender-se a magnitude deste fenómeno, observa-se a Figura 3, que nos ilustra um conjunto de valores agregados acerca da evolução dos dados gerados pelo mundo que atualmente habitamos, acompanhados de duas frases que descrevem aquilo que se pode esperar dessa mesma evolução.

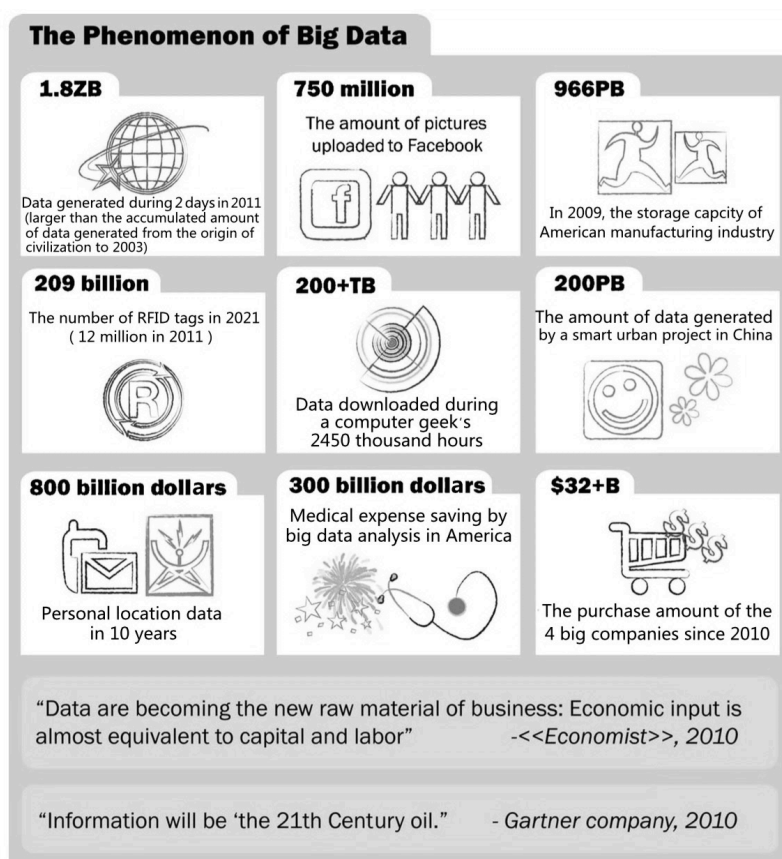


Figura 3. O fenómeno do *Big Data*. Extraído de (M. Chen et al., 2014).

Como podemos observar na Figura 3, para além da natureza volumosa, variada e veloz dos dados, temos que nos aperceber do potencial da informação que pode ser extraída e analisada para revelar novo conhecimento e otimizar o processo de decisão (Sagiroglu & Sinanc, 2013).

Uma perspetiva realista e prática acerca do modelo dos 3Vs, mais precisamente o volume, variedade e velocidade dos dados, é apresentada por Krishnan (2013), afirmando que o volume dos dados caracteriza a quantidade de dados gerados continuamente e diferentes tipos de dados possuem diferentes tamanhos. Isto remete para a variedade dos dados, que denota os múltiplos formatos possíveis: estruturados, semiestruturados ou não estruturados, provenientes de páginas *Web*, *log files*, pesquisas, redes sociais, *e-mail*, documentos, dados de sensores, entre outros (Zikopoulos & Eaton, 2011). Hoje em dia não há controlo sobre o formato de entrada dos dados e a estrutura dos mesmos, tal é a variedade existente. A principal complexidade associada à existência de uma variedade de formatos assenta na disponibilidade de metadados que identifiquem o que os dados verdadeiramente contêm. Por fim, Krishnan (2013) indica que a velocidade consiste no facto de os dados fluírem de forma continua, com o objetivo de serem adquiridos e processados num espaço de tempo muito curto, para retornarem um conjunto de resultados esperados. A Figura 4 conceptualiza aquilo que é o modelo dos 3Vs.

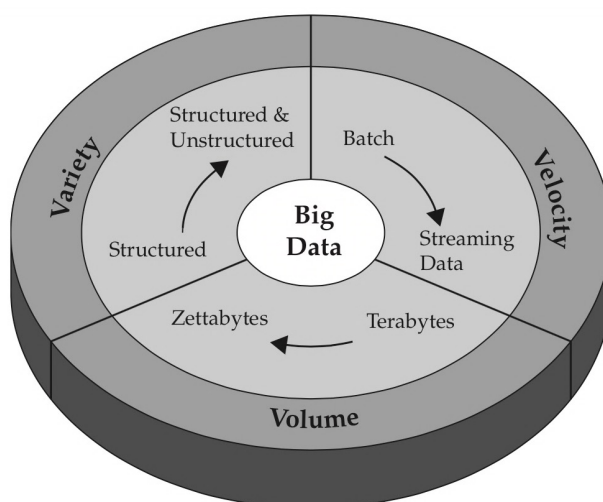


Figura 4. Representação conceptual do modelo dos 3Vs. Retirado de (Zikopoulos & Eaton, 2011).

A IBM também caracteriza o *Big Data* pelo seu volume, variedade e velocidade, como se pode observar na Figura 4 (Zikopoulos & Eaton, 2011). Os autores indicam que a principal causa para o aumento progressivo do volume de dados é o facto de nós, hoje em dia, armazenarmos todos os eventos que ocorrem nas nossas interações com a maioria dos serviços existentes no nosso mundo. O aumento dos sensores, *smart devices* e o uso das Tecnologias da Informação e Comunicação (TIC) contribui para a variedade dos formatos de dados, cujo armazenamento, muitas das vezes, não é adequado em tecnologias de bases de dados relacionais. Por sua vez, se a quantidade e variedade de dados que armazenamos evoluiu, o mesmo aconteceu com a velocidade a que os geramos ou processamos. Segundo Zikopoulos & Eaton (2011), corroborando Krishnan (2013), a ideia de velocidade deve ser aplicada aos dados no momento em que os mesmos fluem, ou seja, a velocidade a que os dados circulam pelas diversas fontes.

Este modelo parece ser uma forma comum de caracterizar *Big Data*, sendo partilhada também por vários outros autores (Chandarana & Vijayalakshmi, 2014; M. Chen et al., 2014; Liu, Yang, & Zhang, 2013; Sagioglu & Sinanc, 2013).

Quando se reflete acerca destas três características, não se deve assumi-las como independentes. Tal facto é referido por Krishnan (2013), quando ilustra um modelo dos 3Vs aumentado com novas características originárias do cruzamento entre volume, variedade e velocidade, como demonstra a Figura 5.

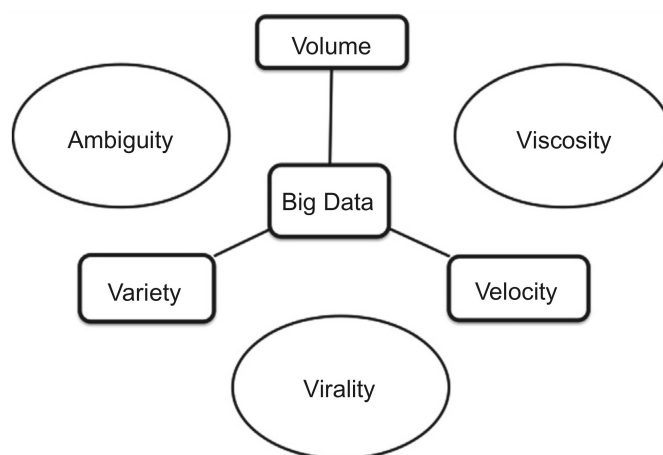


Figura 5. Modelo dos 3Vs aumentando com características originárias do cruzamento entre volume, variedade e velocidade. Retirado de (Krishnan, 2013).

Começa-se assim a reparar na complexidade associada ao processamento de *Big Data* (Krishnan, 2013):

1. Ambiguidade, criada pelo volume e variedade dos dados associados à indisponibilidade de metadados apropriados;
2. Viscosidade, que significa a resistência presente no fluxo do vasto volume de dados a elevada velocidade. Essa resistência pode ter origem, por exemplo, nas regras de negócio da organização ou em limitações tecnológicas;
3. Viralidade, que mede o tempo com que os dados se propagam entre nós de uma rede. Esta característica é fruto do cruzamento entre a variedade dos dados e a velocidade a que os mesmos fluem.

Com o aumento do interesse no conceito de *Big Data* por parte da comunidade científica e técnica, alguns autores foram estudando novas características a acrescentar ao modelo dos 3Vs. Para muitos destes, faltava uma característica relevante para além das três já existentes. Uma característica, aqui implicitamente mencionada, que revela a importância de armazenar, processar e analisar *Big Data*: o valor dos dados.

O valor dos dados pode ser entendido como a importância da informação escondida no *Big Data*, de modo a que justifique a identificação, transformação e extração dos respetivos dados para análise e suporte do processo de tomada de decisão (Liu et al., 2013). Segundo Chandarana & Vijayalakshmi (2014), integrar diferentes tipos de dados, com vista a extrair informação para o negócio e provocar vantagens competitivas, representa o valor do *Big Data*.

Para além disso, muitos autores defendem que um quinto *V* é crucial quando caracterizámos *Big Data*: a veracidade (Chandarana & Vijayalakshmi, 2014). Como os autores explicam, quando lidamos com dados que fluem a grande velocidade, por vezes não existem recursos para a limpeza dos mesmos antes do seu consumo. Esta noção de confiabilidade, autenticidade, reputação, no fundo veracidade, deve ser tida em conta quando lidamos com *Big Data*, como demonstra a Figura 6, que ilustra o modelo dos 5Vs.

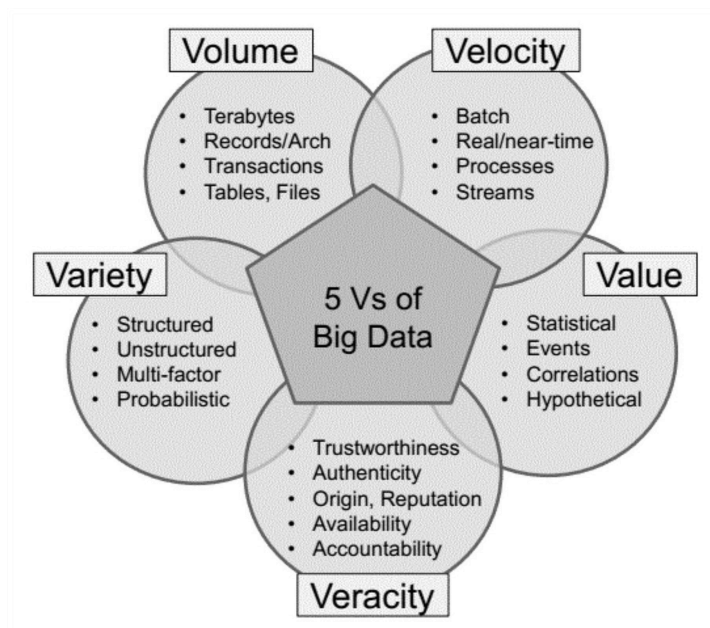


Figura 6. Modelo dos 5Vs. Retirado de (Chandarana & Vijayalakshmi, 2014).

2.1.2 Porquê Toda a Azáfama? – Motivação e Importância do *Big Data*

A área das Tecnologias da Informação e Comunicação foi agitada pelo aumento crescente da popularidade do termo *Big Data* (Figura 7).

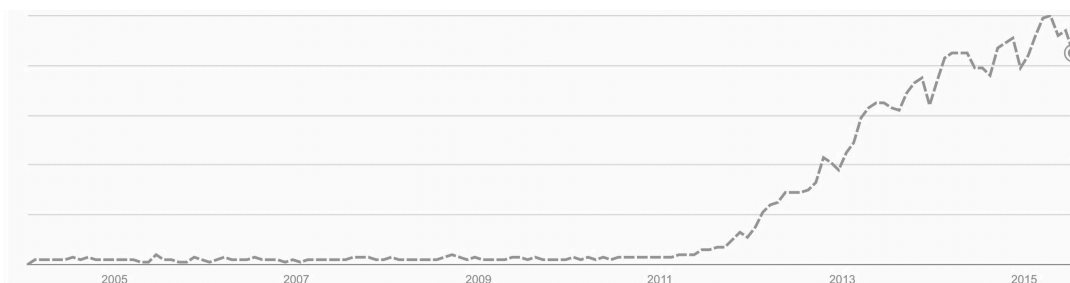


Figura 7. Popularidade do termo *Big Data*, segundo o Google Trends. Retirado de (Trends, 2015).

Muitas outras estatísticas demonstram que a percentagem de dados gerados está a crescer exponencialmente (Chandarana & Vijayalakshmi, 2014). No entanto, ainda hoje, muitos profissionais da

indústria se perguntam: “Porquê *Big Data*? E porquê agora?” (Krishnan, 2013). Segundo Krishnan (2013), a promessa do *Big Data* é a de permitir o acesso a vastos volumes de dados que podem ajudar a retirar grande valor dos mesmos, isto com a mínima intervenção humana e com novas tecnologias de processamento de dados. Após esta afirmação, o autor refere que *Big Data* sempre esteve presente, mas era trabalhado com muita intervenção humana e interações manuais. O que mudou e originou a *buzzword* foi o aparecimento do processo automatizado, veloz, escalável e flexível, resultando muitas das vezes em custos reduzidos.

Convém salientar que as soluções de *Big Data* não são um substituto para as tecnologias relacionais existentes, como o *Data Warehouse*. Segundo Zikopoulos & Eaton (2011), qualquer pessoa que sugira o contrário não tem experiência suficiente ou não compreende a dimensão do investimento feito nestas tecnologias tradicionais de suporte à gestão.

Por conseguinte, deve existir motivação para utilizar soluções de *Big Data* quando (Zikopoulos & Eaton, 2011):

1. Se analisam não só dados estruturados, mas também semiestruturados e não estruturados, provenientes de uma variedade de fontes;
2. A análise de uma determinada amostra dos dados não se revela tão eficaz quanto a análise de todos os dados ou de uma amostra de maior dimensão;
3. A análise assenta num processo iterativo e exploratório, ao contrário da análise tradicional onde as métricas já se encontram pré-estabelecidas.

Dito isto, *Big Data* “tem o potencial para revolucionar a forma como lidamos com o negócio” (Schneider, 2013), incluindo o modo como os clientes interagem com a organização, o desenvolvimento de produtos e serviços, o aumento de vantagens competitivas e a criação de novas estratégias para aumentar a rentabilidade (M. Chen et al., 2014).

As atuais tecnologias não estão a par da exigência associada ao armazenamento e processamento de um vasto volume de dados. Mais relevante ainda, muitas das vezes estas não são capazes de extrair valor desses mesmos dados. Tal como referido por Zikopoulos & Eaton (2011), quando analisar o conjunto completo dos dados é mais eficaz que analisar uma amostra desse conjunto, as organizações detêm uma vantagem competitiva enorme sobre os seu concorrentes (Chandarana & Vijayalakshmi, 2014).

Big Data tem potencial para desempenhar um papel de extrema relevância em várias áreas de aplicação. Na Figura 8 são identificadas algumas delas.

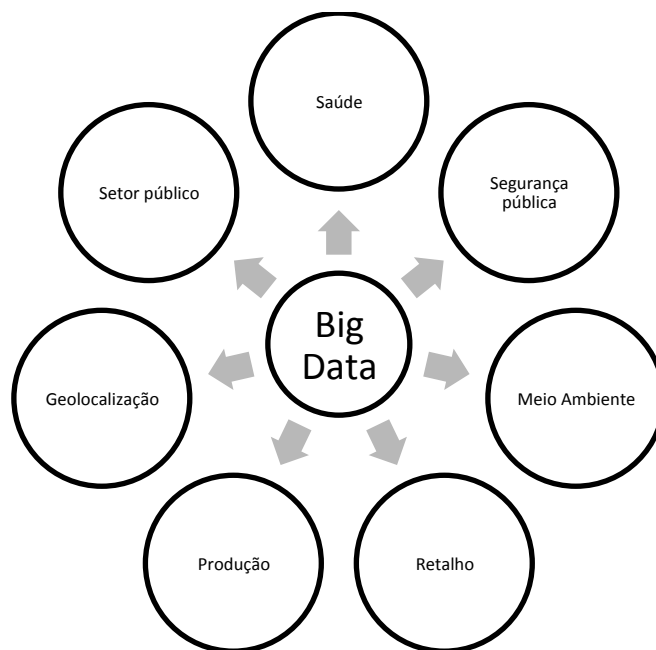


Figura 8. Várias áreas de aplicação em que o *Big Data* pode desempenhar um papel fundamental. Baseado em (Chandarana & Vijayalakshmi, 2014; Sagioglu & Sinanc, 2013).

Com a evolução tecnológica, as organizações incidentes nestas áreas de aplicação irão recolher mais dados, o que levará à continuação do crescimento exponencial do volume de dados. Por conseguinte, o *Big Data* irá ter um enorme impacto na criação de valor para os negócios e consumidores (M. Chen et al., 2014).

2.1.3 Com Grandes Dados Advêm Grandes Desafios

Muitas das organizações da atualidade reconhecem a importância da vasta quantidade de dados que circula pelos seus processos, no entanto as mesmas reconhecem que não dispõem do *know-how* necessário para lidar com esses mesmos dados (Liu et al., 2013).

Liu et al. (2013) adotam uma abordagem que permite relacionar as características associadas ao *Big Data* (volume, variedade, velocidade e valor) com os principais desafios a ele associados. Quando determinada companhia se pretende envolver no armazenamento, processamento e análise de *Big Data* deve esperar desafios em três frentes: desafio de expressão, devido à análise de dados de natureza não estruturada, que segundo o autor, citando o *The US Forrester*, representarão 70 a 80 por cento do aumento do volume de dados entre 2010 e 2015; desafio de processamento, pois um dos principais problemas é o processamento eficiente de *Big Data*, exigindo a combinação entre computação de alta

performance e novas técnicas para extrair informação destes dados; por fim, o desafio da confiança associado à natureza muitas vezes incerta do *Big Data*, podendo levar a informações incorretas, imprecisas ou incompletas. Por outro lado, se os dados forem verdadeiros e valiosos, trazem consigo potenciais riscos de segurança e privacidade. Estes dois aspetos constituem um dos requisitos essenciais para o uso confiável de *Big Data*.

Segundo Chandarana & Vijayalakshmi (2014) a tecnologia, arquiteturas, gestão e análise de *Big Data* lidam com os desafios abaixo apresentados:

1. Privacidade, segurança e confiança - a organização deve garantir a segurança e privacidade da informação, de modo a assegurar a confiança dos utilizadores que vão gerando cada vez mais dados. Changqing Ji et al. (2012) corroboram esta afirmação e dão também particular atenção para o correto estudo dos termos legais e regulamentos quanto à proteção da privacidade dos utilizadores. Segundo Zhang, Chen, & Li (2013) a privacidade, segurança e confiança desempenham um papel relevante no planeamento do sistema de armazenamento dos dados;
2. Gestão e partilha de dados - as organizações começaram-se a aperceber que para quaisquer dados deterem valor, deve ser possível identificá-los, acedê-los e usá-los, quer dentro da organização quer fora da mesma, quando aplicável, como é o exemplo do conceito de *Open Data*;
3. Competências tecnológicas e analíticas - muitas das tecnologias atuais não são capazes de lidar com um vasto volume e variedade de dados. Por conseguinte, as partes interessadas colocam expectativas elevadas nos fornecedores de TIC;
4. Armazenamento e processamento de dados - existem tecnologias capazes de lidar com a entrada de vastas quantidades de dados num determinado sistema. No entanto, quando falamos no processamento transacional, operações de *insert*, *update* e *read* não conseguem, muitas das vezes, ser eficientemente asseguradas pela maioria das tecnologias existentes. Changqing Ji et al. (2012) concordam que as tecnologias atuais não satisfazem as necessidades do *Big Data* e a evolução dos sistemas de armazenamento é inferior à evolução dos dados. Os autores também referem que a maior parte dos algoritmos atuais não lida eficientemente com *Big Data*, devido à sua heterogeneidade;

5. Propriedade dos dados - quando uma organização lida com dados provenientes do seu ambiente, não deve descurar o facto de que por vezes estes pertencem a utilizadores ou a outras entidades. Numa era em que as organizações tendem a armazenar cada vez mais dados, a sua propriedade deve ser cuidadosamente considerada. Um dos exemplos mais óbvios são as redes sociais, onde o dono dos dados é o utilizador que cria a sua página e não a organização que os armazena. Este facto relevante é também explicado por Wigan & Clarke (2013), referindo que as organizações tendem a assumir que detêm os direitos dos dados e que muitas das vezes a legislação atual favorece estas organizações, permitindo até que elas estejam no direito de não serem obrigadas a proceder à eliminação definitiva dos dados, mesmo após solicitação de um utilizador;
6. Outros desafios apresentados por Chandarana & Vijayalakshmi (2014) vão de encontro à perspectiva de Liu et al. (2013), acrescentando o facto de que à medida que a quantidade de dados aumenta, torna-se mais difícil retirar percepções dos dados em tempo útil, destacando assim o desafio da velocidade.

Até este ponto já se percebem alguns dos desafios que a aquisição, armazenamento, gestão e análise de *Big Data* pode trazer, bem como o facto de que os sistemas tradicionais de gestão de bases de dados demonstram-se desadequados para lidar com *Big Data* (M. Chen et al., 2014).

Se os sistemas tradicionais não conseguem lidar eficientemente com *Big Data* e a utilização de novas técnicas de computação e armazenamento de dados se revela mais adequada, deve-se focar também a atenção para desafios específicos no desenvolvimento de aplicações de *Big Data*, pois a concepção destas aplicações difere das tradicionais, devido ao uso de tecnologias inovadoras. M. Chen et al. (2014) identificam vários desafios, destacados na Tabela 1.

Conforme é possível observar, grande parte dos autores referidos nesta secção fazem transparecer a segurança como uma das principais preocupações associadas ao *Big Data*. Por conseguinte, o restante conteúdo desta secção focar-se-á na discussão de questões especificamente relacionadas com a segurança.

De acordo com Paryasto, Alamsyah, Rahardjo, & Kuspriyanto (2014), algumas iniciativas de *Big Data* falham devido à ausência de uma clara especificação de mecanismos de segurança. Contudo, os autores deixam claro que o volume e variedade dos dados dificulta o processo de controlar o acesso a quem pode usar os dados ou analisá-los.

Tabela 1. Desafios do desenvolvimento de aplicações *Big Data*. Adaptado de (M. Chen et al., 2014).

Representação dos dados	Representações inapropriadas podem reduzir o valor dos dados originais e até criar barreiras na análise dos mesmos. Uma representação eficiente deve providenciar uma semântica correta para interpretação computacional e/ou humana.
Compressão de dados e redução da redundância	Muitas das vezes <i>Big Data</i> é redundante e a redução desta redundância e compressão dos dados pode produzir resultados satisfatórios, sem alterar o valor dos mesmos.
Gestão do ciclo de vida dos dados	Com vários mecanismos a gerar uma vasta quantidade de dados, nem sempre é praticável armazenar todos eles. Por vezes, o valor dos dados depende da atualidade dos mesmos. Nesses casos deve ser estabelecida uma estratégia quanto aos dados que devem ser armazenados e os que devem ser descartados. Caso seja necessário armazenar um vasto volume de dados, durante um período de tempo alargado, devem ser consideradas técnicas para garantir consistência e disponibilidade desses dados, a longo prazo (Zhang et al., 2013).
Mecanismos analíticos	Estes mecanismos devem ser capazes de processar um vasto conjunto de dados heterogêneos, dentro de um período limitado de tempo. Novas tecnologias de bases de dados, tais como bases de dados não relacionais, já mostraram as suas vantagens no processamento de <i>Big Data</i> , quando comparadas com as bases de dados relacionais. Contudo as bases de dados não relacionais também detêm alguns problemas, levando por vezes à adoção de arquiteturas híbridas.
Confidencialidade dos dados	Devido à falta de capacidade para lidar com <i>Big Data</i> , por vezes as organizações tendem a delegar este processo a uma outra companhia especializada, levando a possíveis fugas de informação sensível. Devem ser tomadas medidas preventivas para garantir a segurança dos dados. Um exemplo comum de falta de segurança quando se lida com <i>Big Data</i> é retratado por Wigan & Clarke (2013), descrevendo que, muitas das vezes, quando uma determinada organização tenta manter o anonimato das entidades envolvidas num determinado <i>dataset</i> , acaba por disponibilizar atributos que certos analistas, através de inferência, utilizam para re-identificar as entidades em questão.
Consumo de energia	Lidar com <i>Big Data</i> implica, como já referido, novas técnicas de computação, muitas vezes asseguradas por várias máquinas em simultâneo, aumentando inevitavelmente o consumo de energia elétrica. Por conseguinte, não podemos considerar apenas a escalabilidade dos sistemas de <i>Big Data</i> , sem considerar um controlo adequado do consumo energético. Deve existir uma proporcionalidade entre a carga de trabalho e o consumo de energia, usando componentes energeticamente eficientes ou usando apenas os recursos necessários para a computação exigida (Kambatla, Kollias, Kumar, & Grama, 2014).
Escalabilidade	Quando se desenvolve uma aplicação de <i>Big Data</i> , os mecanismos utilizados devem ser capazes não só de analisar os dados que se encontram atualmente ao dispor da organização, mas também estar devidamente preparados para necessidades futuras. Quando se fala em estender a capacidade de armazenamento, esse processo deve ser razoavelmente simples, garantindo assim a escalabilidade da aplicação (Zhang et al., 2013).
Cooperação	A análise de <i>Big Data</i> é uma atividade interdisciplinar, devendo envolver peritos em várias áreas para retirar o máximo valor dos dados. Deste modo, uma arquitetura de rede deve ser estabelecida, para garantir que os vários intervenientes conseguem cooperar e atingir os objetivos esperados.

Existem algumas vulnerabilidades encontradas em projetos de *Big Data*, das quais se destacam:

1. Computação não segura, quando uma aplicação não segura acede a dados sensíveis e pode corrompê-los, levando a resultados incorretos ou até mesmo à interrupção do sistema como um todo;
2. Validação e filtragem dos dados, que devido à quantidade de dados recolhidos, resulta num dos maiores desafios no processo de recolha de dados, levando por vezes à falta de confiança nos mesmos;
3. Controlo de acessos granular, normalmente descurado pelas soluções de *Big Data*, cuja principal preocupação assenta na *performance* e escalabilidade;
4. Armazenamento e partilha não seguros, devido à distribuição do armazenamento por vários nós, bem como à comunicação entre esses mesmos nós, *middlewares* e utilizadores finais;
5. Privacidade nas análises de dados, pois ocasionalmente pode ocorrer violação de termos de privacidade ou divulgação não intencional de informação sensível.

Além das vulnerabilidades acima apresentadas, Paryasto et al. (2014) referem também que nem sempre a encriptação dos dados é solução para todos os problemas de segurança identificados. Diferentes tipos de dados requerem diferentes tipos de proteção e aplicar o mesmo tipo de encriptação para resolver problemas diferentes pode resultar num aumento de custos e complexidade. Por isso, os autores identificam algumas técnicas de proteção de dados que podem ser consideradas quando se lida com *Big Data*:

1. Anonimização dos dados, removendo ou adulterando dados que possam identificar uma entidade individual;
2. *Tokenization*, protegendo dados sensíveis via substituição por um *token* que não tem qualquer significado para entidades não autorizadas a aceder aos dados;
3. Controlos de acesso à base de dados, configurando permissões de acesso, de modo a proteger a base de dados.

Um assunto que levanta muitas questões e preocupações, é o da privacidade individual. Cada fase do ciclo de vida do *Big Data* pode representar sérios riscos à privacidade (Cumbley & Church, 2013). Com as tecnologias atuais, as pessoas podem ser localizadas e monitorizadas com significativa precisão, como por exemplo via partilha de mensagens na *internet*, *cookies*, *RFID*, *smartphones*, videovigilância,

pagamentos electrónicos, entre outros, onde cada vez mais os dados são armazenados. “Estas mudanças são uma consequência irreversível do mundo electrónico... nós vivemos num mundo de *Big Data* e parte desses dados são sobre nós” (Cumbley & Church, 2013).

Quando se transcende para a fase de combinação dos dados provenientes de múltiplas fontes, o problema da privacidade torna-se ainda mais preocupante. Um exemplo elucidativo, foi a revisão da política de privacidade da Google, quando a partir de março de 2012 a gigante americana poderia passar a cruzar dados do seu sistema de pesquisa, Youtube e Maps, de modo a conseguir um perfil mais detalhado do utilizador. Ou seja, a Google dispunha de um sistema capaz de saber o que um utilizador queria saber, o que queria ver e que locais frequentava (Cumbley & Church, 2013).

No entanto, estes não são os únicos problemas. Uma das principais preocupações está na fase onde se retiram percepções acerca dos dados, a análise. Um exemplo bastante comum prende-se com os sistemas de *cross-selling*, cujo objetivo visa recomendar produtos, com base em interações entre o cliente e a loja em questão. À primeira vista a ideia é bastante apelativa para ambas as partes, mas tal como nos exemplificam Cumbley & Church (2013), num dos casos de uma loja, uma gravidez foi revelada aos familiares da respetiva cliente, através de publicidade relacionada com produtos para grávidas, o que parece bastante intrusivo no que à privacidade individual diz respeito. Outra preocupação prende-se com a análise de dados não estruturados, que apesar do significativo valor que pode juntar a percepções acerca de um indivíduo, acrescenta também uma grande probabilidade dessas percepções estarem erradas, devido à complexidade das técnicas associadas a essas mesmas análises. Tal como Paryasto et al. (2014), Cumbley & Church (2013) apontam a técnica de garantir o anonimato dos dados, no entanto ressalvam também o facto de existir alguma facilidade em identificar os dados novamente. Sedayao, Bhardwaj, & Gorade (2014) estudam a anonimização dos dados em técnicas de *Big Data* e concluem que é necessária uma análise cuidada dos *datasets* para garantir que os mesmos não serão alvo de tentativas de correlação.

Por fim, tudo isto afeta diretamente os indivíduos no uso da informação analisada. O marketing é uma área onde é óbvio o uso de *Big Data* (Cumbley & Church, 2013). Com o avanço das técnicas de *Big Data*, outros usos óbvios irão emergir, alguns deles com poucas implicações a nível individual, outros com implicações mais significativas. Para isto, é necessário garantir a veracidade dos resultados produzidos pela análise de *Big Data*.

Lu, Zhu, Liu, Liu, & Shao (2014) referem que a característica veracidade do *Big Data* foi introduzida para endereçar os desafios de segurança. No entanto, os autores ressaltam também que a questão da privacidade do *Big Data* ainda é pouco madura. Por isso, chama-se a atenção para os requisitos de privacidade de uma arquitetura de *Big Data*, como ilustra a Figura 9.

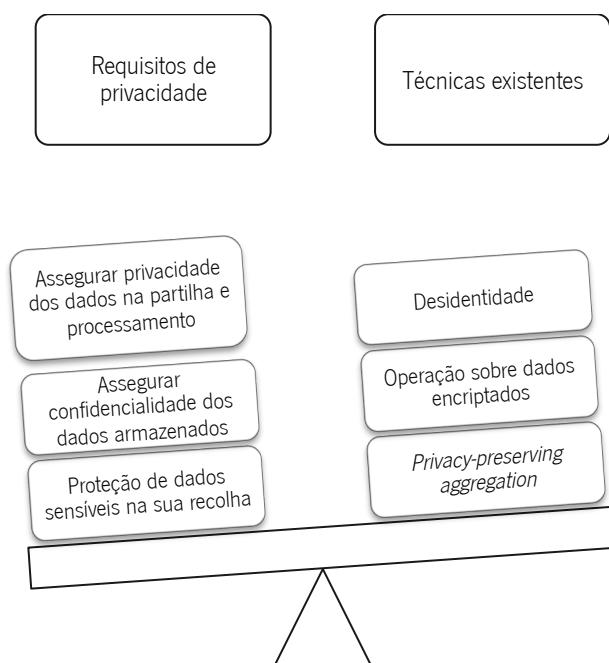


Figura 9. Requisitos de privacidade e técnicas existentes. Informação retirada de (Lu et al., 2014).

Lu et al. (2014) apontam que as técnicas existentes nem sempre se revelam adequadas para *Big Data*, como é o exemplo da *privacy-preserving aggregation*, que apesar de conseguir garantir a privacidade individual nas fases de recolha e processamento de dados, serve um propósito muito específico e essa agregação pode revelar-se inflexível na altura de ser usada em novas análises de dados para extrair conhecimento. Outra técnica que se revela desadequada, devido à elevada complexidade e consumo de tempo, é a operação sobre dados encriptados. Contudo, nem todas as técnicas adequadas estão desalinhadas com os requisitos de privacidade, sendo que a desidentificação assente na randomização dos dados consegue garantir a eficiência e flexibilidade da análise de *Big Data*, apesar de, por vezes, ser possível re-identificar os dados, caso as devidas precauções não sejam tomadas, como é referido por Wigan & Clarke (2013).

2.2 Paradigmas e Termos Associados ao *Big Data*

Esta secção visa explicar alguns paradigmas e termos, frequentemente relacionados com *Big Data*, na medida em que alteram a forma como se lida com o armazenamento e processamento dos dados.

2.2.1 Processamento Tradicional de Dados vs. Processamento de *Big Data*

Krishnan (2013) define processamento tradicional de dados como a recolha, processamento e gestão de dados, resultando na extração de informação para o utilizador final. Esses mesmos dados são de natureza estruturada e com um volume moderado, sendo que tarefas como *data quality* e *data cleansing* não se revelam particularmente problemáticas. O ciclo deste processamento é descrito na Figura 10.

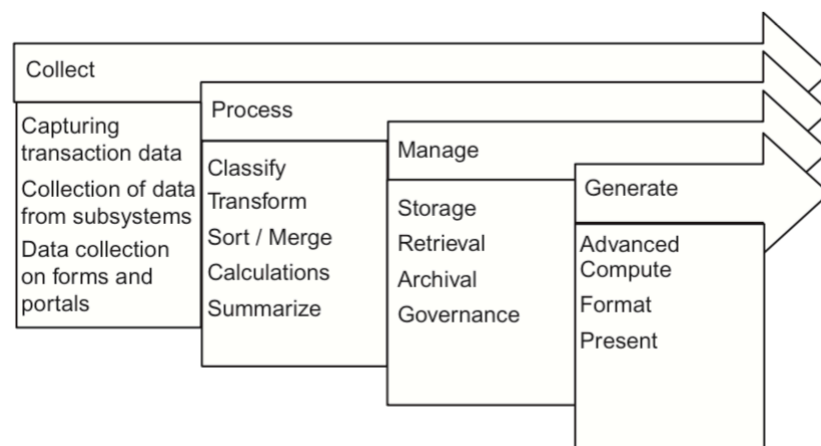


Figura 10. Ciclo de processamento tradicional de dados. Retirado de (Krishnan, 2013).

O processamento de *Big Data* (Figura 11) difere do tradicional processamento de dados (Figura 10), pois neste último começa-se por analisar os dados e estabelecer um conjunto de requisitos que, eventualmente, leva a *data discovery* e à criação de um modelo de dados, seguidos da estruturação da base de dados, carregando o modelo de dados no seu estado final (Krishnan, 2013), enquanto que quando se processa *Big Data*, os dados são recolhidos e carregados numa determinada plataforma, depois os metadados são aplicados e, em seguida, cria-se a estrutura. Assim que a estrutura é aplicada, os dados são transformados e analisados.

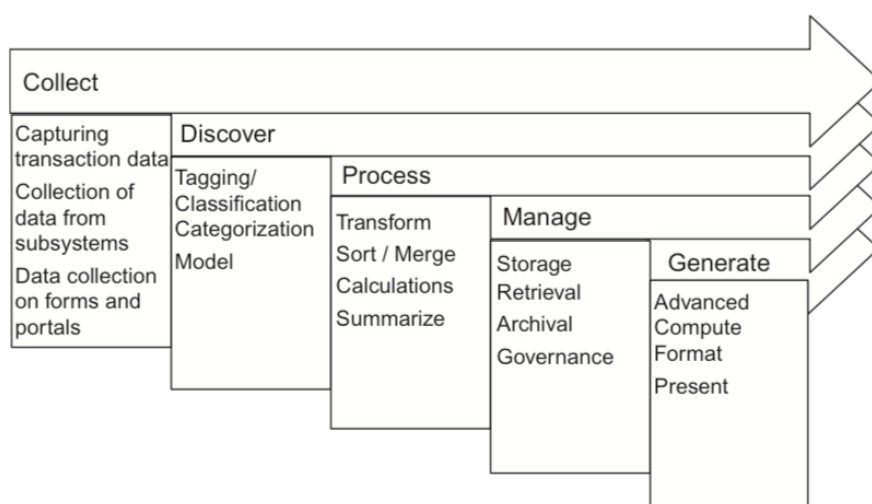


Figura 11. Ciclo de processamento de *Big Data*. Retirado de (Krishnan, 2013).

Como é imaginável, uma arquitetura orientada a bases de dados relacionais não se revelaria adequada. Segundo Krishnan (2013), para processar conteúdos com tal volume e complexidade, é mais adequada uma arquitetura orientada a ficheiros, ou não relacional, com uma determinada linguagem de programação associada.

Se considerarmos que o objetivo principal de qualquer análise de dados é a extração de informação e consequente extração de conhecimento, então podemos considerar que o fluxo de processamento de *Big Data* pode ser genericamente representado da mesma forma que o processo de extração de conhecimento (Figura 12).



Figura 12. Processo de extração de conhecimento. Adaptado de (Philip Chen & Zhang, 2014).

Pode-se afirmar que Krishnan (2013) instancia o processo de extração de conhecimento, retratado na Figura 12, pois este parece servir de alicerce ao fluxo de processamento de *Big Data* descrito pelo autor, tal como se observa na Figura 13, sendo possível identificar 4 principais fases.

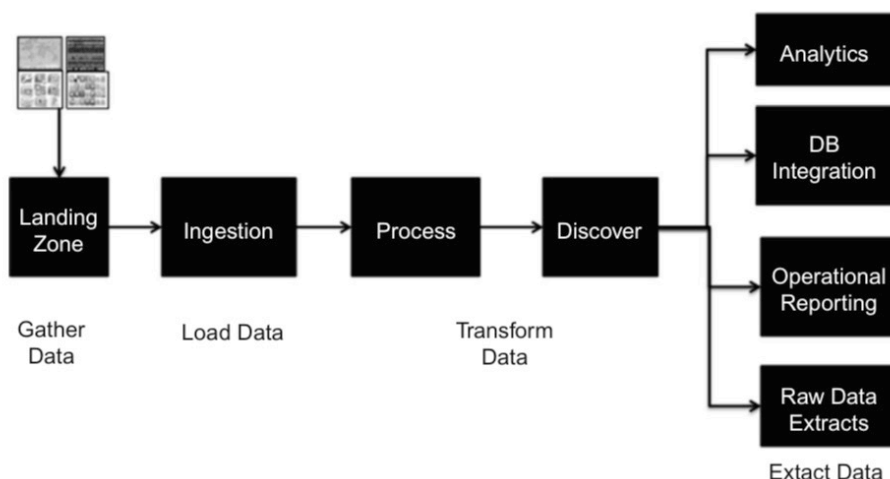


Figura 13. Fluxo de processamento de *Big Data*. Retirado de (Krishnan, 2013).

O objetivo e sequência do fluxo de processamento de *Big Data* são muito semelhantes ao processo de extração de conhecimento acima mencionado:

1. Recolha - os dados são recebidos de diferentes fontes e armazenados numa área designada de *landing zone*.
2. Carregamento - os dados são carregados, sendo depois aplicados metadados sobre os mesmos, criando assim a estrutura dos dados. Nesta fase, o volume total dos dados dá origem a pequenas partições, que podem ser horizontais (particionar os dados por conteúdo) ou verticais (particionar os dados por estrutura).
3. Transformação - os dados são transformados, aplicando regras do negócio e processando o seu conteúdo. Tipicamente, esta fase envolve vários passos e torna-se habitualmente complexa.
4. Extração - o conjunto de dados resultante das fases anteriores é utilizado para *Analytics*, *Reporting*, integração no *Data Warehouse*, ou para servir outros propósitos de visualização.

2.2.2 Requisitos do Processamento de *Big Data*

Baseado naquilo que é entendido como o processamento de *Big Data*, Krishnan (2013) estabelece duas classes de requisitos: requisitos de infraestrutura e requisitos de processamento de dados. A Figura 14 providencia uma visão geral acerca dos requisitos de uma infraestrutura de *Big Data*.

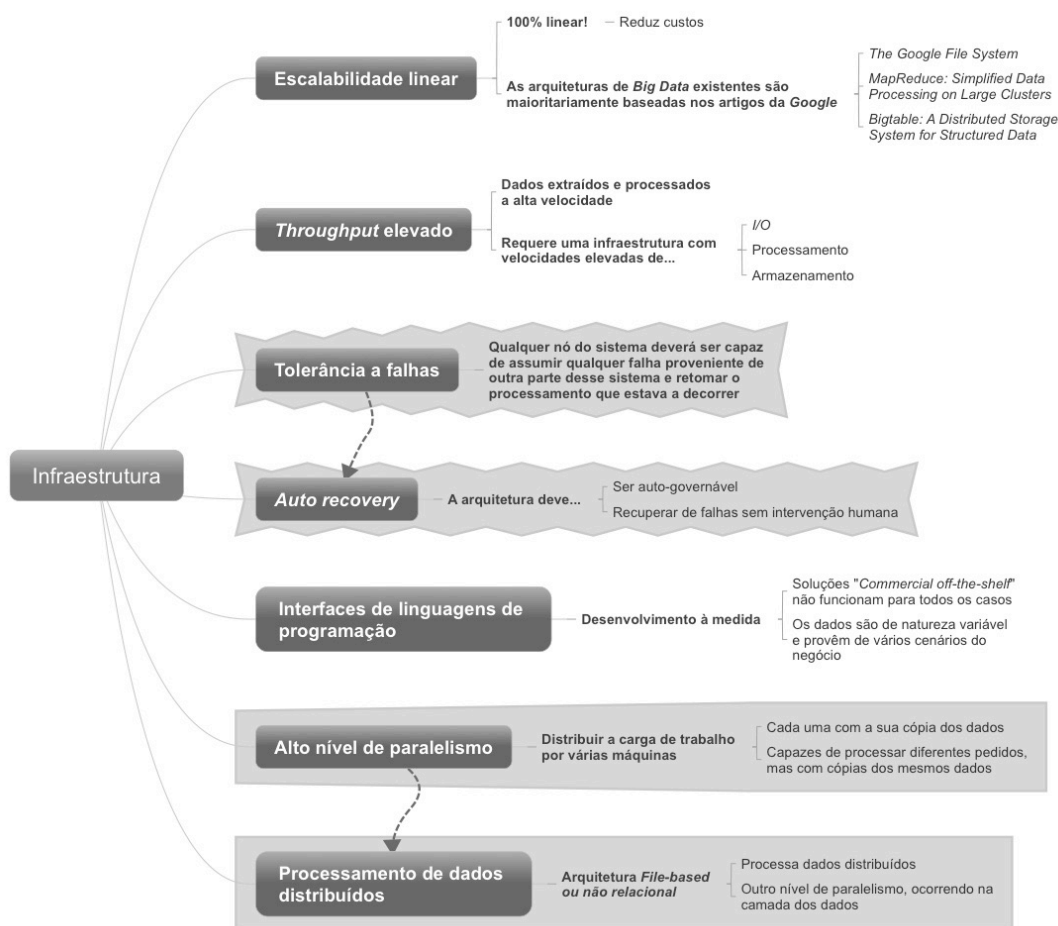


Figura 14. Requisitos de uma infraestrutura *Big Data*. Baseado em (Krishnan, 2013).

Uma infraestrutura capaz de processar *Big Data* pode ser resumida como: 100% escalável, ou seja, quando um recurso é adicionado deve ser aproveitado na sua totalidade, ao contrário do que acontece quando, a título de exemplo, se adiciona um disco a um sistema de armazenamento e só metade do espaço é utilizável (Krishnan, 2013); de alto desempenho, capaz de processar dados a elevada velocidade; assente num ambiente robusto, sendo capaz de tolerar falhas e recuperar das mesmas, com a mínima intervenção humana; uma infraestrutura que incute paralelismo ao nível da carga de trabalho e dos dados.

Contudo, os requisitos do processamento de *Big Data* prendem-se não só com a infraestrutura, mas também com o processamento dos dados propriamente dito. Por conseguinte, a Figura 15 descreve esses mesmos requisitos, sugerindo uma arquitetura de dados desprendida de modelos ou *schemas* fixos e um processo veloz de recolha de dados, com transformações mínimas aos mesmos. Por fim, os dados deverão poder ser particionados, replicados ou partilhados, preferencialmente, num sistema de ficheiros ou não relacional.



Figura 15. Requisitos do processamento de *Big Data*. Baseado em (Krishnan, 2013).

2.2.3 Os Fundamentos do *MapReduce*

Como já pôde ser observado, o paralelismo é um dos requisitos-chave de uma arquitetura de *Big Data*, melhorando a *performance* do sistema, através da comunicação entre vários nós envolvidos na computação (Kambatla et al., 2014).

O *MapReduce*, originalmente criado pela Google (Schneider, 2013), revelou-se uma técnica inovadora para lidar com um vasto volume de dados, que até à data era uma tarefa destinada a servidores com excelentes processadores e uma vasta quantidade de memória, que contavam também com uma extensa capacidade de armazenamento. Acrescido a isto, surgiam os sistemas operativos com elevados custos de aquisição, os Sistemas de Gestão de Bases de Dados Relacionais, as tecnologias de *Business Intelligence*, entre outros, que normalmente comportavam elevados gastos em recursos qualificados e consumiam bastante tempo.

O cenário atual remete para necessidades mais alargadas, entre as quais se destacam o processamento de um maior volume de dados e, muitas vezes, não estruturados. Ciente destas necessidades, a indústria promoveu o *commodity* hardware, *sistemas* de ficheiros distribuídos,

tecnologias *open source*, armazenamento mais barato e arquiteturas *service-oriented* (Schneider, 2013). Contudo, pode-se afirmar que não existia uma *framework* apropriada, lacuna colmatada pela Google quando introduziu o *MapReduce*.

Muito resumidamente, o *MapReduce* divide uma tarefa em pequenas sub-tarefas, passíveis de serem geridas e distribuídas por múltiplos nós, que trabalham em conjunto para obterem um determinado resultado. Pode-se afirmar que a *framework* usa a estratégia *divide-and-conquer*.

A Figura 16 ilustra os fundamentos do *MapReduce*, conseguindo descrever, genericamente, o funcionamento inerente ao paradigma, que traz consigo a capacidade de simplificar computações de larga escala, abstraindo grande parte da complexidade normalmente associada a esse processo (Schneider, 2013).

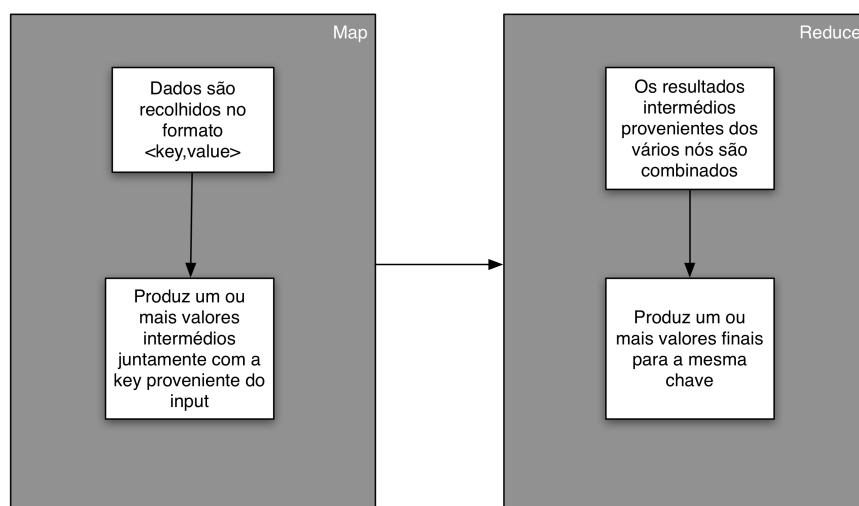


Figura 16. Breve explicação dos fundamentos do *MapReduce*. Baseado em (Schneider, 2013).

Quando se fala de *Big Data* surge, quase incessantemente, associado o Hadoop. O Hadoop é uma instanciação do *MapReduce*, implementada pela Apache (M. Chen et al., 2014) e amplamente adotada por companhias como a Yahoo ou o Facebook. Existem várias distribuições assentes no Hadoop, das quais se podem destacar (Apache, 2014): Cloudera, Hortonworks, IBM InfoSphere BigInsights (Zikopoulos & Eaton, 2011), MAPR e Amazon Elastic MapReduce. Ao longo desta dissertação surgirá a oportunidade de estudar mais aprofundadamente o Hadoop, nomeadamente na fase de enquadramento tecnológico.

2.2.4 *NoSQL*

Lidar com *Big Data* exige novas técnicas de armazenamento. Tiwari (2011) começa por referir que *NoSQL* é a junção de duas palavras, "*No*" e "*SQL*", definindo-o como um paradigma que se opõe à tão conhecida *Structured Query Language (SQL)*. No entanto, atualmente a indústria descreve o termo como *Not Only SQL* (Edlich, 2014). *NoSQL* abrange todas as bases de dados que não seguem os princípios das bases de dados relacionais, e que são comumente associadas a vastos conjuntos de dados provenientes da *Web*. O autor refere que o *NoSQL* não é apenas uma tecnologia, mas sim um conjunto de tecnologias e diversos conceitos de armazenamento e manipulação de dados.

Outra opinião que vai de encontro à anterior, no que diz respeito à dissimilaridade para com as bases de dados relacionais, é a de Coelho (2013) que refere o facto de as bases de dados *NoSQL* diferirem das relacionais devido à não inclusão de relações explícitas nos dados. O autor refere também que estas bases de dados tendem a estar alocadas em ambientes distribuídos.

Uma das grandes diferenças entre as bases de dados *NoSQL* e as bases de dados *SQL* é a forma como se executam as *queries*. Pokorny (2013) refere que o facto de existirem vários tipos de bases de dados *NoSQL* impede (para já) a existência de uma linguagem unificada como o *SQL*. Desta forma, cada tipo de base de dados *NoSQL* tem a sua própria forma de aceder aos dados. No entanto, como podemos observar através de Pokorny (2013), existem variantes do *SQL*, como a *Google Query Language*. Uma outra forma de acesso aos dados é através da *Application Programming Interface (API)* das respetivas bases de dados.

Leavitt (2010) aborda a temática numa perspetiva de tentar prever se o *NoSQL* irá conseguir "cumprir as suas promessas" (Leavitt, 2010) e é um pouco mais sucinto na definição de *NoSQL*, afirmando, talvez num tom irónico, que o que as bases de dados *NoSQL* têm em comum, entre elas, é o facto de não serem relacionais. Curiosamente, Leavitt (2010) defende que estas bases de dados são mais acessíveis para desenvolvedores que não estão ambientados com a linguagem *SQL*. O autor afirma também que as mesmas tendem a poder ser distribuídas por várias máquinas. Ainda no presente artigo, o autor relança-nos para um passado onde identifica a causa do aparecimento de bases de dados *NoSQL*: a necessidade de lidar com dados não estruturados. Particularmente, é referido o facto da maioria deste tipo de bases de dados ser *open source*, juntando baixo custo de aquisição à vantagem na velocidade de processamento e ao não suporte do conceito *ACID (Atomicity, Consistency, Isolation, Durability)*. No entanto, nem tudo são vantagens, destaca Leavitt (2010):

1. Complexidade, devido à não utilização de *SQL* e possível dificuldade no uso de outras linguagens para interrogação de dados, por parte de programadores experientes em *SQL*;
2. Não fiabilidade, pois, a exclusão de *ACID* retira uma camada extra de fiabilidade;
3. Não consistência, também assegurada pelo conceito *ACID*. Contudo esta pode ser assegurada por programação adicional.

Tudorica & Bucur (2011) referem que para as bases de dados *NoSQL* oferecerem alta *performance* e disponibilidade tiveram que sacrificar o *ACID* em troca pelo *BASE* (*Basic Availability, Soft state, Eventual consistency*).

Convém salientar que associados aos dois conceitos acima mencionados surge, frequentemente, o teorema *CAP* (*Consistency, Availability, Partition tolerance*). Este teorema afirma que qualquer sistema de partilha de dados em rede apenas pode deter, simultaneamente, duas de três propriedades (Brewer, 2012):

1. Consistência equivalente a ter todos os dados armazenados num único nó;
2. Disponibilidade dos dados;
3. Partição dos dados.

O *ACID* representa uma filosofia de *design*, tradicionalmente associada a bases de dados relacionais. *BASE* é também uma filosofia de *design*, concebida por Brewer (2012), também autor do teorema *CAP*, curiosamente.

De modo a concluir, Leavitt (2010) refere que as bases de dados *NoSQL* enfrentam um desafio: infamiliaridade com a tecnologia, o que por vezes leva as organizações a não ponderarem sobre este tipo de soluções. Contudo, Leavitt (2010) conclui que este tipo de bases de dados não virá substituir as bases de dados relacionais, mas irá revelar-se uma solução mais adequada para certos tipos de projetos.

2.2.4.1 Tipos de bases de dados *NoSQL*

As bases de dados *NoSQL* podem ser categorizadas em vários tipos, que se destacam atualmente. Leavitt (2010) aponta 3 grandes tipos de bases de dados *NoSQL*:

1. Bases de dados *key-value*, que indexa valores identificados por chaves. Um exemplo é o Amazon's SimpleDB, que providencia acesso e armazenamento de informação na *cloud*;

2. Bases de dados *column-oriented*, contendo uma coluna expansível com dados relacionados. O autor refere o facto de o Facebook ter utilizado uma base de dados *column-oriented* de alto desempenho, designada de Cassandra, para suportar a sua rede social. O facto de a Apache ter desenvolvido o HBase, que é um emulador da reconhecida Google's Big Table, é outra curiosidade mencionada;
3. Como último tipo, surge o armazenamento orientado a documentos, que como o nome sugere, armazena a informação em documentos, permitindo adicionar tantos campos quanto o utilizador desejar, com qualquer dimensão pretendida. Mais uma vez, é apontada a Apache Software Foundation, como desenvolvedora do CouchDB, acompanhada pela 10Gen, como provedora de suporte para a tecnologia MongoDB.

Tudorica & Bucur (2011) identificam, sucintamente, os tipos de bases de dados *NoSQL*, dividindo-os em classes *core* e *soft*:

1. Bases de dados *core NoSQL*, criadas especialmente para suportar serviços *Web 2.0*:
 - a. Bases de dados *column-oriented* (HBase, Cassandra, Hypertable, Cloudata, Amazon SimpleDB);
 - b. Bases de dados orientadas a documentos (CouchDB, MongoDB...);
 - c. Bases de dados *Key-Value/Tuple* (Azure Table Storage, MEMBASE, Riak, Amazon Dynamo, Voldemort...);
 - d. Bases de dados de grafos (Neo4J, Infinite Graph, Sones...).
2. Bases de dados *soft NoSQL*, não relacionadas com serviços *Web 2.0* e muitas das vezes fazendo uso do paradigma *ACID*:
 - a. Bases de dados de objetos (db4o, Versant, Objectivity...);
 - b. Bases de dados orientadas a *Grid & Cloud* (GigaSpaces, Queplix, Hazelcast...);
 - c. Bases de dados *XML* (Mark Logic Server, EMC Documentum xDB...);
 - d. Bases de dados multi-valor (U2, OpenInsight, OpenQM, Globals);
 - e. Outras bases de dados relacionadas com *NoSQL* (IBM Lotus/Domino, Intersystems Cache, eXtremeDB...).

2.3 *Smart Cities*

Se nos debruçarmos sobre a realidade atual, pode-se observar a crescente diversidade de dispositivos inteligentes que integram o nosso quotidiano. Estes dispositivos são considerados inteligentes pois são capazes de exibir comportamento autónomo e inteligente quando inseridos em determinados cenários (Jara et al., 2013). Segundo Jara et al. (2013), são exemplo: *smartphones*, *tablets*, *smart watches*, televisões, carros, sensores, sistemas de controlo de acesso, entre muitos outros. Os autores referem que em 2020, entre 50 e 100 biliões de dispositivos irão conectar todas as coisas em redor do utilizador, alcançando a tão proclamada *Internet of Things (IoT)*.

Quando dispositivos como estes começam a integrar um papel fundamental na nossa sociedade, torna-se essencial compreender a forma como o fazem. Por conseguinte, “as interações e influência nas nossas vidas são questões por explorar, devendo ser compreendido como é que a *IoT* irá desempenhar um papel fundamental nas nossas *Smart Cities*...” (Jara et al., 2013).

Cada vez mais a humanidade vê nos centros urbanos o seu ambiente preferencial de habitação, sendo que, atualmente, mais de metade da população vive em ambientes urbanos (Vilajosana et al., 2013). Os efeitos causados pela crescente urbanização provocam graves alterações nos custos e recursos, como emprego, transportes, infraestrutura de comunicações, energia e água (Hedlund, 2013). Segundo Vilajosana et al. (2013), é urgente o aparecimento de soluções e o avanço tecnológico pode ser a resposta adequada. Para além disso, os autores referem a estimativa de que o mercado das *Smart Cities* alcance as centenas de biliões de dólares em 2020.

2.3.1 O que é uma *Smart City*?

O conceito de *Smart City* tem vindo a atrair alguma atenção no que respeita a iniciativas de desenvolvimento urbano. O avanço nas tecnologias e meios de comunicação e o consequente impulsionamento dos serviços electrónicos revelam-se relevantes na inovação levada a cabo pelas cidades nas áreas da saúde, inclusão social, ambiente e negócio (Schaffers et al., 2011).

Vilajosana et al. (2013), citando Pike Research (2011), definem *Smart City* como “a integração da tecnologia numa abordagem estratégica de sustentabilidade, comportamento cívico e desenvolvimento económico”. Os autores destacam ainda a importância da interação entre os múltiplos domínios de atuação numa cidade, ressaltando que qualquer modelo que tente definir a natureza dinâmica destas cidades deve contemplar a diversidade dos seus elementos.

Uma cidade pode ser designada como inteligente, quando o “investimento em capital humano e social, bem como em transportes e Tecnologias da Informação e Comunicação, alimentar um crescimento económico sustentável e uma confortável qualidade de vida, gerindo de forma inteligente os recursos naturais, através de um governo participativo” (Caragliu, Del Bo, & Nijkamp, 2009). A presente definição contempla aspetos sociais e económicos, destacando simultaneamente o papel fundamental das novas Tecnologias de Informação e Comunicação como capacitadores da participação social, de forma a melhorar a qualidade de vida dos cidadãos (Schaffers et al., 2011). A Figura 17 apresenta os possíveis domínios de atuação numa *Smart City*. Entre eles destacam-se alguns aspetos sociais e económicos fundamentais num ambiente citadino inteligente.



Figura 17. Possíveis domínios de atuação numa *Smart City*. Adaptado de (Schaffers et al., 2011).

Uma cidade inteligente pode ser distinguida de uma cidade tradicional através do investimento e gestão ponderada das Tecnologias da Informação e Comunicação, fazendo com que, de alguma forma, os custos de operação sejam menores e o aproveitamento dos recursos seja maior. Além disto, outra característica que distingue uma *Smart City* das demais tradicionais é o foco no cidadão como peça central do sistema, transformando a forma como a cidade o serve, pois cada vez mais este se comporta como um consumidor exigente de serviços públicos, apesar das restrições económicas vividas atualmente (Hedlund, 2013). A Tabela 2 enumera as principais diferenças entre uma *Smart City* e uma cidade tradicional, destacando aspetos tecnológicos e humanos. Salienta-se, sobretudo, o investimento inteligente em TI, a natureza ágil dos serviços prestados, a qualidade dos dados, o foco no cidadão e a gestão eficiente dos recursos humanos.

Tabela 2. Diferenças entre uma cidade tradicional e uma *Smart City*. Adaptado de (Hedlund, 2013).

Cidade tradicional	<i>Smart City</i>
Compra ou desenvolve aplicações à medida.	Investe e reutiliza as Tecnologias da Informação e capacidades do negócio; Adapta em vez de comprar ou desenvolver soluções à medida.
Resistentes à mudança; Serviços pouco flexíveis.	Serviços ágeis, adaptáveis à mudança, associados a menores custos.
Serviços fáceis de ser usados por especialistas, mas pouco amigáveis para o cidadão.	Cidadão é o utilizador principal; Serviço simples de ser usado, com o objetivo primário de servir o cidadão; Serviços electrónicos retiram sobrecarga aos recursos humanos da cidade.
A integração e reutilização dos dados é dispendiosa; Elevada presença de sistemas legados.	O uso e integração dos dados é facilitada pelas arquiteturas e <i>guidelines</i> utilizadas; Maior qualidade dos dados.
Custos de infraestruturas baixos. No entanto, os custos das Tecnologias da Informação são elevados.	Elevado custo infraestrutural. Baixo custo de TI.
Cidadãos excluídos dos processos.	Cidadãos dispõem de transparência, participação e poder de decisão.
Desperdício de recursos humanos; Pouca automação.	Aproveitamento eficiente dos recursos humanos; Serviços centralizados de apoio ao cidadão.
Especialistas realizam trabalho não especializado.	O trabalho dos especialistas é centrado em tarefas especializadas.

2.3.2 Desafios Associados a Iniciativas de *Smart Cities*

Apesar da necessidade de inovação no mercado imobiliário, mobilidade, energia, recursos naturais, ambiente, entre outros, as iniciativas de *Smart Cities* não estão a corresponder às expectativas criadas. As necessidades supramencionadas deviam, por si só, justificar uma mais abrangente introdução e aceitação das *Smart Cities* (Vilajosana et al., 2013). De acordo com Vilajosana et al. (2013), a necessidade de alterações políticas, o capital limitado e a fragmentação das estruturas de capitais, aliados às dúvidas que surgem em investimentos como estes, retraem as oportunidades de aceitação e desenvolvimento deste tipo de iniciativas.

As iniciativas são necessárias e procuradas, no entanto existe um *gap*, em termos sustentabilidade e viabilidade, entre uma aplicação e um serviço electrónico (Schaffers et al., 2011), isto é, nem todas as aplicações chegam ao patamar de serem designadas como um serviço electrónico,

transformando corretamente as capacidades tecnológicas em inovação, de modo a assegurar o fluxo contínuo de dados e informação. As TI podem ser um elemento-chave na melhoria dos serviços prestados ao cidadão e na eficiência dos governos, mas vários obstáculos terão de ser endereçados. A indústria não pode continuar a acrescentar complexidade às soluções de TI, sem considerar o foco na usabilidade e na correta integração, não só entre soluções, mas também entre o utilizador e a solução propriamente dita (Hedlund, 2013). Uma cidade inteligente deve adaptar os seus serviços às diferentes experiências de utilização e dispositivos. Comunidades *open source* poderão contribuir substancialmente para o aparecimento de boas práticas e *open solutions* (Schaffers et al., 2011), bem como o aparecimento de iniciativas *Open Data*, que podem contribuir para o desenvolvimento de serviços electrónicos por parte de terceiros (Hedlund, 2013).

Na perspetiva de Schaffers et al. (2011), as cidades necessitam de *standards* de aplicações informáticas, modelos de negócio e serviços, podendo o custo destas iniciativas ser reduzido, devido essencialmente à cooperação e partilha de recursos entre *Smart Cities*. Contudo, a necessidade de *standards* não se aplica apenas nas aplicações informáticas, sendo necessário colmatar também a falta de padrões nas políticas ambientais internacionais, nacionais e regionais e a pouca sistematização das metodologias para avaliar o retorno financeiro proporcionado pela aplicação de iniciativas de *Smart Cities* (Vilajosana et al., 2013). Por fim, Schaffers et al. (2011) destacam a relevância dos *Living Labs*, envolvendo o cidadão no processo de inovação, com vista a enriquecer a pesquisa científica baseada na experimentação.

2.4 Arquiteturas de *Big Data* no Contexto de *Smart Cities*

De modo a entender-se o papel do *Big Data* numa *Smart City*, deve-se primeiramente perceber a forma como este é originado e como o seu processamento poderá afetar a dinâmica dos cidadãos. A *Internet of Things*, frequentemente mencionada quando se descreve uma *Smart City*, surge como um ecossistema capaz de fornecer dados capturados pelos *Smart Objects*, que quando processados, podem providenciar a inteligência necessária para analisar e compreender comportamentos (Jara et al., 2013). Segundo Jara et al. (2013), a análise de *Big Data*, gerado pelos *Smart Objects* e enquadrados nas *Smart Cities*, pode fornecer um contributo significativo para a alteração do comportamento citadino, alterando decisões, interesses, ações e prioridades.

A coordenação e integração revelam-se dois factores críticos no desenvolvimento de uma *Smart City*, onde constam diferentes tipos de bases de dados, novos métodos de *Data Mining*, novas

tecnologias de integração de dados e, finalmente, novas formas de gestão governamental (Batty et al., 2012).

A sociedade atual está a redefinir novos papéis para os cidadãos. A interação constante com *Smart Objects*, o uso intensivo da *internet* e das redes sociais, levam à integração do cidadão numa sociedade de informação, em que o mesmo se encontra permanentemente conectado, informado e participativo na produção de opiniões, experiências, sentimentos e informação. Normalmente a sua motivação para a produção da informação é fazer parte da cocriação de valor (Jara et al., 2013).

De acordo com Kitchin (2013), há vários anos que as cidades e seus governos produzem conjuntos de dados de dimensões consideráveis, mas apenas uma parte desses dados podiam ser processados ou disponibilizados publicamente, essencialmente devido a limitações de cariz tecnológico. Por conseguinte, a associação entre *Big Data* e *Smart Cities* surge na esperança de estruturar formas de conseguir lidar com os dados produzidos por uma cidade, de modo a, através do uso das Tecnologias da Informação e Comunicação, disponibilizar novos serviços que alterem significativamente a dinâmica dos seus habitantes, governo, e outras partes interessadas.

De seguida são apresentadas diversas arquiteturas de *Big Data* para *Smart Cities*, relevantes para o adequado enquadramento deste trabalho. Posteriormente, ainda neste capítulo, nomeadamente na secção de apresentação do modelo de conceitos e das observações ao enquadramento conceptual, são destacadas algumas das limitações destas arquiteturas. Além disso, no início do capítulo 4, comentam-se novamente as limitações destas arquiteturas, de forma mais detalhada, com vista a justificar o aparecimento da arquitetura BASIS.

2.4.1 “Smart City Dashboard for Integrating Various Data of Sensor Networks”

Suakanto, Supangkat, Suhardi, & Saragih (2013) apresentam uma arquitetura para monitorizar dados provenientes de vários sensores situados numa *Smart City*, composta por uma rede de sensores, um conjunto de servidores e o *TCP/IP* como protocolo de comunicação, como é ilustrado na Figura 18.

Nesta arquitetura, os sensores (temperatura, poluição do ar/água, tráfego) atuam como uma fonte de dados, que serão posteriormente armazenados num conjunto de servidores que implementam uma base de dados ou um Sistema de Informação Geográfica (SIG). Para além disso, a informação acerca do estado da cidade, em várias vertentes, é analisada através de uma *Dashboard* disponibilizada aos utilizadores.

Esta arquitetura foi validada através de implementação em contexto real, nomeadamente na cidade de Bandung, na Indonésia, provando que o sistema suporta a implementação de qualquer tipo de sensores, segundo Suakanto et al. (2013).

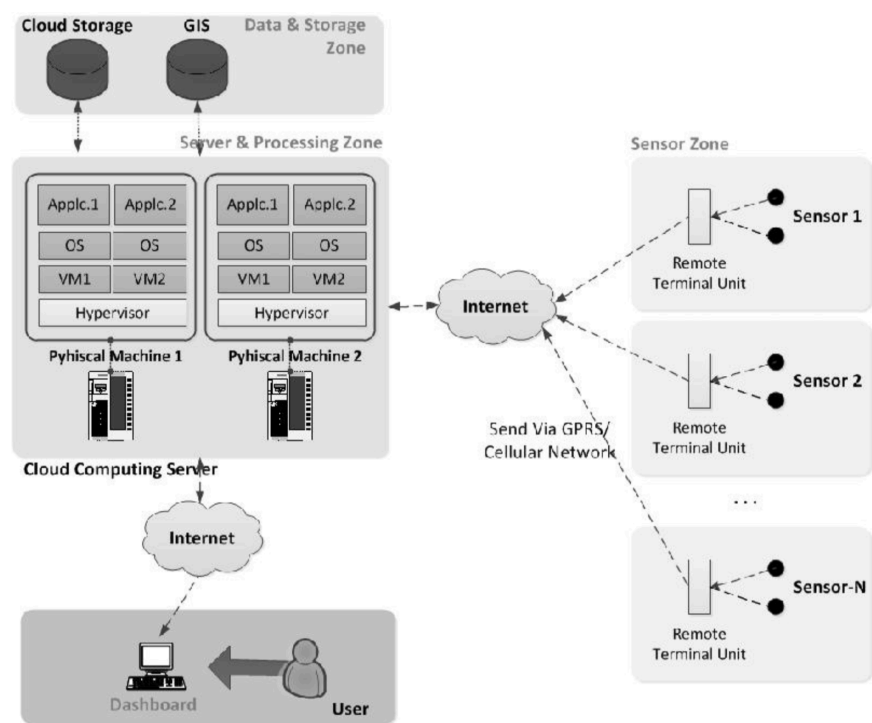


Figura 18. Arquitetura proposta por Suakanto et al. (2013).

Contudo, os autores referem que foi utilizado um número reduzido de sensores, não referindo o volume de dados gerado. Como investigação futura, sugerem a possibilidade de refinar a componente analítica, de modo a que a arquitetura possa incorporar previsões ou outras técnicas associadas aos *Decision Support Systems (DSS)*.

2.4.2 “Big Data in Large Scale Intelligent Smart City Installations”

Girtelschmid, Steinbauer, Kumar, Fensel, & Kotsis (2013) sugerem uma arquitetura que contempla componentes destinados a processamento em tempo real. A Figura 19 apresenta uma visão geral da arquitetura proposta pelos autores, onde constam os seguintes componentes: plataforma de *streaming*, que consiste num *cluster* composto por vários nós, responsáveis pela recepção de dados provenientes de sensores e detecção de alterações e falhas nos mesmos. Esta plataforma é também responsável pela acumulação de leituras dos sensores, quando aplicável, e pela limpeza dos dados; repositório da ontologia, responsável pelo armazenamento das leituras mais recentes e de informação agregada. Alterações na ontologia fazem o motor de regras reavaliar essas mesmas regras e tomar as

devidas ações; o motor de regras contém um conjunto de regras executadas numa base periódica, podendo enviar alertas, adaptar a plataforma de *streaming* para novas situações, ou enviar o resultado da execução da regra para uma aplicação cliente; as aplicações de clientes podem registar-se para receber os resultados da execução das regras, enviar *queries* para o motor de regras e armazenar novas regras nesse mesmo motor.

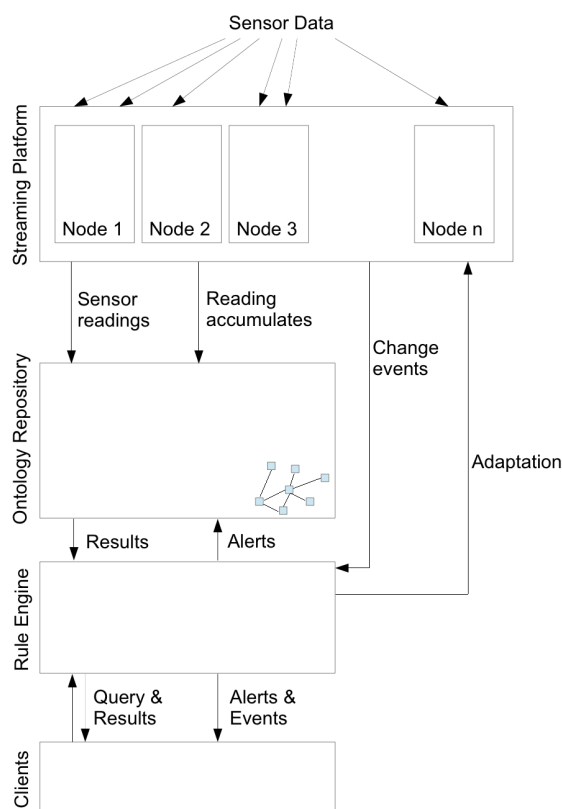


Figura 19. Arquitetura proposta por Girtelschmid et al. (2013).

Girtelschmid et al. (2013) apontam como resultados satisfatórios, o facto de em cenários compostos por um número reduzido de sensores o motor de regras comportar-se satisfatoriamente, elaborando simples decisões, como desligar um eletrodoméstico se a temperatura atingir um determinado limite.

No entanto, os autores ressaltam como limitações consideráveis e investigação futura:

1. O decréscimo significativo da *performance*, quando são processados conjuntos de dados de maior dimensão;
2. Os dados usados para validar a arquitetura não estão diretamente relacionados com dados processados por uma *Smart City*;

3. A não presença de restrições de segurança, tanto na arquitetura como no protótipo desenvolvido para a validar;
4. A falta de métricas de avaliação em instalações de larga escala.

2.4.3 “Cloud based Big Data Analytics for Smart Future Cities”

A arquitetura concebida por Khan, Anjum, & Kiani (2013) pretende suportar um serviço de análise de dados num modelo *Analytics as a Service*. A Figura 20 ilustra as três camadas da arquitetura, seguindo uma abordagem *bottom-up*. Em cada uma das camadas, os autores apresentam as funcionalidades que possibilitarão o cumprimento dos objetivos dessa mesma camada.

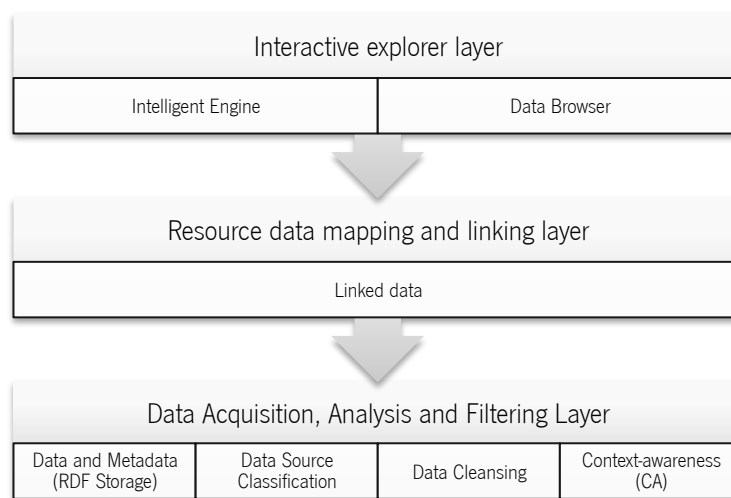


Figura 20. Arquitetura proposta por Khan et al. (2013). Adaptado.

O primeiro requisito é a aquisição de dados, lidando com a possível heterogeneidade das fontes, através de identificação e processamento dos respetivos metadados. Para responder a este requisito, a primeira camada da arquitetura consiste num conjunto de repositórios heterogêneos e distribuídos, que podem ser acedidos através de uma *API*, *Web services* ou conectores especificamente desenvolvidos, quando necessário. De modo a lidar com a complexidade associada à recolha de dados de vários sensores que compõem uma *Smart City*, os autores referem, citando Mitton, Papavassiliou, Puliafito, & Trivedi (2012), um conjunto de *standards* de semântica e do *Open Geospatial Consortium (OGC)* para a integração de dados provenientes de um conjunto heterogêneo de sensores.

A segunda camada da arquitetura é responsável pelo mapeamento dos dados e criação de relações, tornando os dados semanticamente relevantes e disponíveis para exploração futura. Os autores destacam o uso da *Resource Description Framework (RDF)* para o referido mapeamento e criação de

relações, bem como a importância desta camada para a descoberta de novos cenários, cuja identificação não seria possível em repositórios de dados isolados.

Por fim, a última camada é capaz de visualizar e processar os dados provenientes das *Open APIs* fornecidas pela segunda camada, ajudando o utilizador a explorar os dados, submeter *queries*, algoritmos e *workflows*, com o objetivo de extrair informação dos repositórios.

Os autores descrevem alguns princípios de concepção seguidos pela arquitetura, dos quais se destacam:

1. Escalabilidade, fazendo uso do *MapReduce* para realizar as operações de relacionamento, exploração e análise dos dados;
2. Baixa latência, através de armazenamento e análise de dados *in-memory*;
3. Uso de *open standards* para facilidade de integração futura;

Contudo, convém salientar que a arquitetura apresentada não foi validada, impossibilitando a identificação de limitações técnicas e respetivas soluções viáveis.

2.4.4 “Bootstrapping Smart Cities through a Self-Sustainable Model Based on Big Data Flows”

A visão descrita por Vilajosana et al. (2013) assenta no facto de que uma *Smart City* deve ser capaz de fornecer um conjunto distinto de serviços, de modo a satisfazer a necessidade de vários domínios de aplicação. Por conseguinte, os autores para além de apresentarem uma arquitetura, como pode ser visto na Figura 21, apresentam também uma cadeia de valores, descrevendo os vários *stakeholders* fundamentais nas diversas camadas da arquitetura. A referida cadeia de valores é ilustrada na Figura 22.

A arquitetura é constituída por:

1. Uma rede de sensores, actuadores, ou outros componentes de *software*, responsáveis por recolher dados, ou atuar nas infraestruturas da cidade ou nos cidadãos;
2. Um *middleware* responsável pelo controlo e gestão dos dados. A componente de gestão de dados providencia *datasets* históricos ou em tempo real, para aplicações que fazem uso contínuo de *streams*. Paralelamente a este componente, surge também a parte da arquitetura responsável não só pelo controlo da rede de sensores/actuadores, mas

também pelo controlo dos componentes do respetivo *middleware* e devidas configurações;

3. Uma camada de filtragem e pós-processamento dos dados, sobreposta por uma camada que oferece serviços para complementar as funcionalidades disponibilizadas pelo *middleware*, anteriormente descrito. Os serviços de gestão de *Big Data* são integrados nessa mesma camada;
4. Por fim, a camada superior da arquitetura inclui serviços analíticos, *interfaces* para recolha de dados em tempo real e *APIs* de *Open Data*, disponibilizadas através de *Web services*.

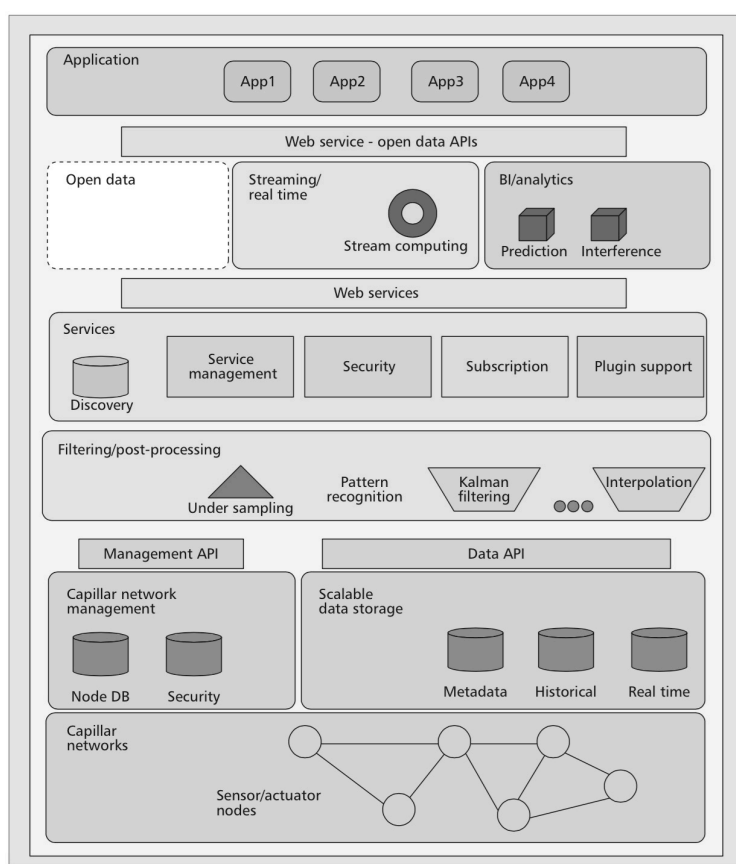


Figura 21. Arquitetura proposta por Vilajosana et al. (2013).

Na Figura 22, pode-se observar a cadeia de valor e os diferentes *stakeholders*. Uma das dificuldades em iniciativas de *Smart City* passa por avaliar o retorno financeiro (Vilajosana et al., 2013). Consequentemente, os autores formularam uma cadeia de valor, onde se pode extrair um modelo de negócio, capaz de garantir retorno do investimento feito numa iniciativa deste género. Os fabricantes de dispositivos que irão incorporar a *IoT* atuam na infraestrutura/rede de sensores e a indústria de *software* atua no *middleware*, serviços e componentes responsáveis pelo controlo, gestão e disponibilização dos

dados. Por sua vez, os desenvolvedores de aplicações de *Third Parties* acrescentam valor ao portefólio de serviços da plataforma, disponibilizando novas aplicações, que podem ser utilizadas pelos cidadãos.

Segundo os autores, a receita proveniente das aplicações pode ser um ponto central no modelo de negócio, funcionando nos moldes das atuais *App Stores*. Acrescido a isto, o consumo dos dados pode deter custos conforme a frequência de acesso aos mesmos, caso aplicável, como o acesso a dados de elevada granularidade (exemplo: informações de trânsito). Contudo, muitas das vezes as cidades podem estabelecer a disponibilização dos dados segundo a visão *Open Data*, sem quaisquer custos associados.

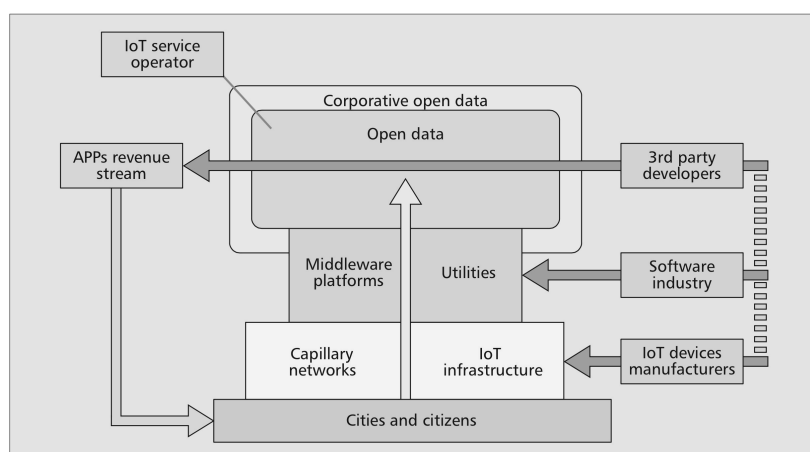


Figura 22. Cadeia de valor proposta por Vilajosana et al. (2013).

Para além de uma arquitetura, os autores apresentam a proposta de um modelo de negócio, com o intuito de impulsionar as iniciativas de *Smart Cities* e ultrapassar algumas barreiras associadas às mesmas. Várias indústrias e entidades poderão beneficiar de um modelo como este, pois as cidades irão usufruir dos serviços disponibilizados por *freelancers*, pequenas e médias empresas, ou outras organizações, que por sua vez também retirarão proveito, através do modelo de receitas provenientes de aplicações. Por outro lado, este modelo de negócio permitirá que a indústria envolvida no investimento inicial garanta retorno desse mesmo investimento feito quer em infraestruturas, quer em *middleware* ou outros componentes.

Uma contrapartida associada a este modelo prende-se com a necessidade constante de atualização de infraestruturas, *APIs* e serviços por parte dos *stakeholders*, o que irá obrigar a uma interação constante, devendo ser regulada pelos devidos termos, de modo a assegurar a qualidade, suporte e persistência da plataforma.

2.5 Modelo de Conceitos e Observações ao Enquadramento Conceptual

Após o enquadramento conceptual de *Big Data* e *Smart Cities*, observa-se que a ambiguidade frequentemente associada aos mesmos pôde ser dissipada com um estudo de literatura científica e técnica direccionada não só para a definição, âmbito, desafios e termos relacionados com *Big Data*, mas também focada na explicação do que pode ser entendido como sendo uma *Smart City*, relacionando-a com as demais tecnologias envolventes (*IoT* e *TIC*).

A Figura 23 sumariza os conceitos estudados, de forma a que se consiga concluir, inequivocamente, acerca do modo como *Big Data* e *Smart City* estão relacionados, tanto entre si como com os demais conceitos.

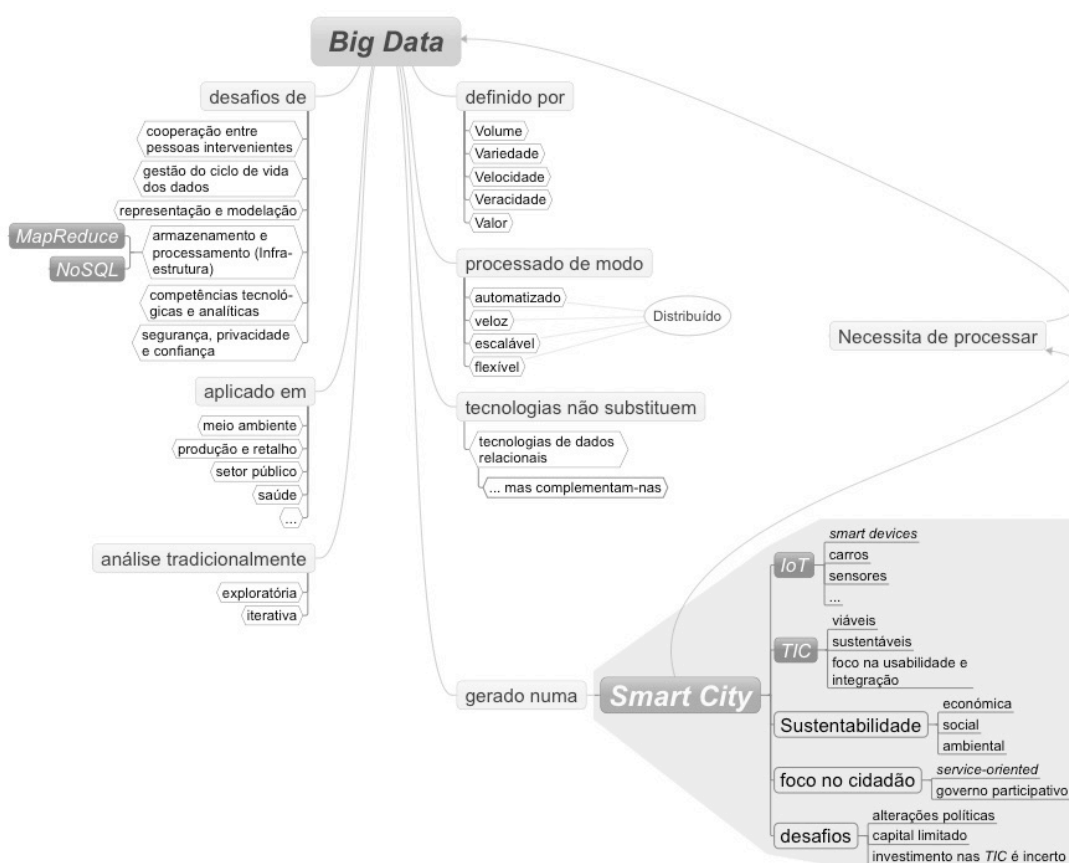


Figura 23. Mapa de conceitos do enquadramento conceptual.

Big Data é geralmente definido pelo modelo dos Vs, o que não significa que todos os autores estudados contemplem na sua definição todas as 5 características: volume, variedade, velocidade, valor e veracidade. Frequentemente, associa-se o aparecimento da *buzzword* “*Big Data*” ao facto de hoje em dia ser necessário processar um vasto volume de dados. Esta associação não está incorreta, no entanto sucede-se que este custoso processamento já há muitos anos se encontra no seio de organizações de

maior dimensão. A *buzzword* começou a ganhar notoriedade devido ao aparecimento de formas de processamento automatizado, veloz, escalável e flexível, no fundo, processamento distribuído. Consequentemente, as tecnologias de *Big Data* não vêm substituir as tecnologias de dados relacionais, como o *Data Warehouse*, mas sim complementar as mesmas, colmatando alguma desadequação no processamento de dados desta nova natureza.

Big Data é gerado e pode ser processado em vários domínios de aplicação, desde a saúde ao meio ambiente, contudo enfrenta desafios de significativa relevância, que podem depender do domínio de aplicação (cooperação entre pessoas intervenientes no processo de implementação de tecnologias *Big Data*; segurança, privacidade e confiança...), ou podem ser transversais a todos os domínios, como é exemplo a gestão do ciclo de vida dos dados, a infraestrutura e as competências tecnológicas e analíticas exigidas.

As *Smart Cities* são uma fonte incessante de *Big Data*, devido à constante presença da *Internet of Things* e à dinâmica dos cidadãos e atividades da cidade. Uma *Smart City* deve ser sustentável nas vertentes económica, social e ambiental e focar a sua atividade no cidadão, disponibilizando-lhe serviços inovadores e enquadrando-o na gestão da cidade. É nestes dois requisitos que a exploração sustentável e viável das TIC desempenha um papel fundamental nas iniciativas de implementação de *Smart Cities*. Ainda assim, estas iniciativas vivem múltiplos desafios, geralmente de cariz político ou económico.

No que respeita às arquiteturas analisadas, verifica-se atualmente alguma preocupação na procura por soluções capazes de processar dados provenientes de uma *Smart City* e entregar um conjunto de serviços inteligentes aos cidadãos, governo e outros intervenientes. Estas arquiteturas tendem a abstrair demasiado a componente tecnológica, focando-se sobretudo nas camadas conceptuais, o que possibilita e motiva a investigação de novas soluções, que possam contemplar diferentes níveis de abstração conceptual e tecnológica. Consegue-se verificar que a falta de *standards* arquiteturais e processuais aliada à ambiguidade dos conceitos de *Big Data* e *Smart City*, dificultam o detalhe das iniciativas existentes e o aparecimento de novas. Para além disso, excetuando a última arquitetura analisada (Vilajosana et al., 2013), nenhuma das arquiteturas ilustra, explicitamente, a importância de disponibilizar dados de forma aberta aos cidadãos ou organizações.

O contributo fornecido pelo estudo destes dois conceitos, juntamente com a análise das arquiteturas apresentadas neste enquadramento conceptual, revelaram-se o ponto de partida para a definição dos objetivos de investigação associados a esta dissertação.

Capítulo 3. Enquadramento Tecnológico

O estudo das tecnologias de recolha, armazenamento, processamento e análise de *Big Data* revela-se um ponto de partida fulcral na investigação de uma arquitetura adequada às necessidades de uma *Smart City*. Estando conceptualmente enquadrados os dois conceitos, esta secção visa identificar e descrever técnicas e tecnologias que podem ser introduzidas ao longo das várias fases do fluxo de processamento de *Big Data*.

3.1 Hadoop

O Hadoop é um ambiente de computação a cargo da Apache Software Foundation, que atua num sistema de ficheiros distribuído, com vista a operar sobre dados de larga escala. O projeto é implementado em Java e inspirado no trabalho desenvolvido pela Google, no *Google File System* e no paradigma de programação *MapReduce*. O principal intuito do Hadoop prende-se com a disponibilização do *MapReduce* a um conjunto mais variado de casos de utilização do que aqueles que o trabalho original da Google poderia alcançar. A título de curiosidade, o nome do projeto surge após o seu criador estabelecer que o mesmo teria de ser fácil de memorizar e não deveria denotar nenhum significado em particular. Por conseguinte, Doug Cutting's designou o projeto de Hadoop, o nome dado pelo seu filho a um elefante de peluche (Zikopoulos & Eaton, 2011). Este não é o único nome invulgar presente no Hadoop.

Através de um sistema de processamento distribuído, escalável e baseado em *batch*, o Hadoop é capaz de produzir resultados provenientes da leitura de vastos conjuntos de dados. O conceito de *batch processing* é relevante na compreensão dos objetivos do Hadoop, pois estes não passam por tempos de resposta reduzidos, acessos real-time ou alto desempenho no processamento de transações, mas sim pela descoberta e análise de resultados em *datasets* outrora quase impossíveis de processar. Contudo, com o decorrer do tempo, vários componentes foram sendo introduzidos no ambiente Hadoop, de modo a oferecer baixa latência no acesso aos dados e processamento de *streams*, tornando-o ainda mais versátil (Apache, 2015b; Hortonworks, 2014).

De acordo com Zikopoulos & Eaton (2011), a replicação é outro conceito-chave quando se descreve o Hadoop, pois são armazenadas réplicas dos dados em vários nós de um *cluster*. Para além disso, o paradigma de programação permite resolver falhas automaticamente, através da execução de partes do programa em múltiplos nós desse mesmo *cluster*. Esta redundância presente no ambiente

Hadoop permite a utilização de *commodity hardware* para implementação de um *cluster* de considerável dimensão, de modo a distribuir os dados e a respetiva programação associada ao processamento dos mesmos, sem temer falhas que possam por em causa o funcionamento do sistema. Apesar do *commodity hardware* ser mais propício a falhas, o mecanismo de replicação garante tolerância às mesmas e a capacidade de auto recuperação.

A camada responsável pelo armazenamento dos dados é denominada de Hadoop Distributed File System (HDFS) (Schneider, 2013), cuja concepção contempla as já mencionadas replicação e tolerância a falhas, acrescidas da escalabilidade linear a 100%, garantindo assim a eficiente utilização da camada de armazenamento e o consequente paralelismo de processamento otimizado ao mais alto nível (Krishnan, 2013).

O HDFS é constituído, principalmente, por um *NameNode* e vários *DataNodes*, tal como ilustra a Figura 24. Num *cluster* Hadoop, os dados são distribuídos em blocos, de modo a que as funções *map* e *reduce* possam ser executadas sobre estes, em oposição à totalidade do conjunto de dados. O *MapReduce* é responsável pela distribuição das tarefas entre os *DataNodes*, em que os blocos estão armazenados, e cópias desses mesmos blocos são armazenadas em outros *DataNodes*, resultado da referida redundância presente no *HDFS*, garantindo assim alta disponibilidade. Por si só, existe um significativo *overhead* associado à coordenação deste processamento distribuído. Contudo, como a execução ocorre localmente em cada um dos nós envolvidos, estes não têm que enviar dados para outros nós, ajudando a melhorar a performance associada à execução de uma tarefa (Zikopoulos & Eaton, 2011).

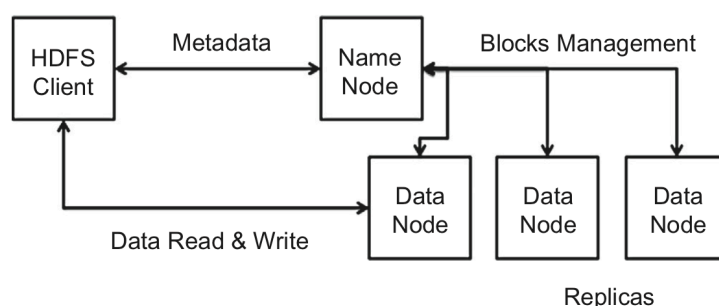


Figura 24. Arquitetura do HDFS. Retirado de (Krishnan, 2013).

Observando a Figura 24 tende-se a supor que o *NameNode*, que gere o acesso aos ficheiros e o mapeamento entre blocos e *DataNodes*, origine um ponto de falha único, devido à utilização de um único *NameNode* em *clusters* Hadoop. No entanto, nas versões mais recentes do Hadoop já é possível a

utilização de redundância de *NameNodes*, permitindo que outro nó atue caso o nó principal falhe (Apache, 2015b).

O ambiente Hadoop engloba um conjunto de componentes que possibilitam o processamento distribuído de um vasto volume de dados, através de modelos de programação relativamente simples. Dentro destes componentes destacam-se o HDFS e o *MapReduce*, supra descritos e explicados, que se enquadram no conjunto de componentes *core* do Hadoop, tal como exemplifica a Figura 25.

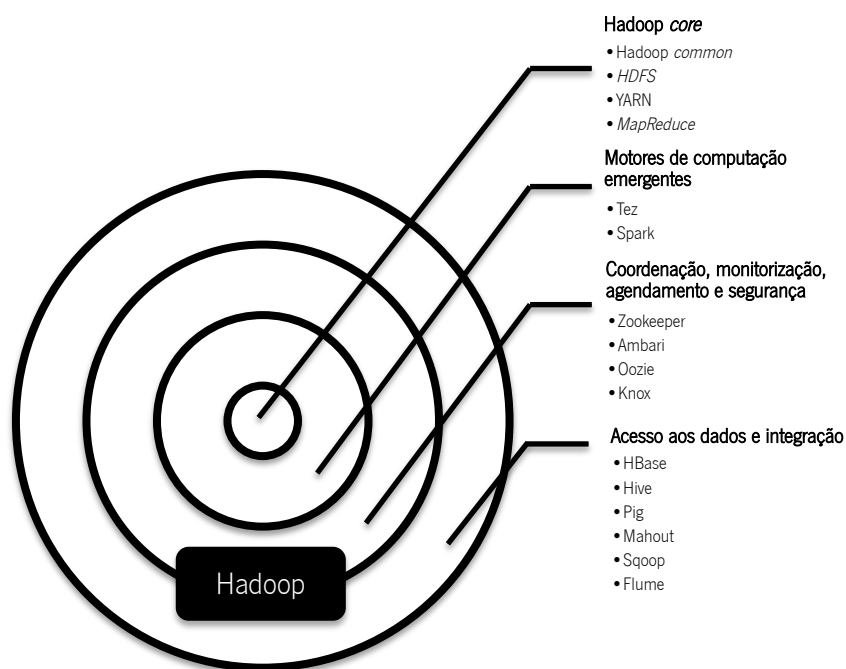


Figura 25. Componentes de destaque do Hadoop. Baseado em (Apache, 2015b; Hortonworks, 2014).

No entanto, convém mencionar que o Hadoop é constituído por múltiplos componentes, sendo a identificação de todos eles uma tarefa de significativa dificuldade. Por conseguinte, a Figura 25 ilustra apenas alguns dos componentes com maior destaque (Apache, 2015b; Hortonworks, 2014). Com vista à superficial compreensão dos componentes identificados, a Tabela 3 apresenta a função de cada um deles.

As versões oficiais do Hadoop provêm da própria Apache. No entanto, outras organizações disponibilizam as suas próprias distribuições, incluindo componentes oficiais da Apache, ou versões modificadas dos mesmos. Estas distribuições fornecidas por organizações como a Cloudera, Hortonworks, IBM, Amazon, MapR, entre outras, são normalmente focadas na utilização comercial, providenciando o suporte adequado às necessidades de um determinado negócio (Apache, 2014).

Tabela 3. Componentes de destaque do Hadoop. Baseado em (Apache, 2015b; Hortonworks, 2014).

Hadoop <i>core</i>	Hadoop <i>common</i>	Representa todas as funcionalidades de suporte aos componentes do Hadoop.
	HDFS	Sistema de ficheiros distribuído do Hadoop.
	YARN	Componente responsável pela gestão da execução das tarefas e dos recursos do <i>cluster</i> .
	<i>MapReduce</i>	Assegura o paralelismo de processamento em <i>datasets</i> de vasta dimensão. O <i>MapReduce</i> é agora um sistema desenvolvido sobre o YARN, ao contrário do que acontecia nas primeiras versões do Hadoop.
Motores de computação emergentes	Tez	Também desenvolvido sobre o YARN, o Tez promete oferecer desempenho e flexibilidade no processamento de dados em <i>batch</i> ou interactivamente. O facto de oferecer melhor desempenho que o <i>MapReduce</i> , levou a que componentes como o Hive e o Pig o adotassem como motor de execução, em substituição do <i>MapReduce</i> .
	Spark	Um motor de computação de elevado desempenho, com múltiplos domínios de aplicação, providenciando um modelo de programação simples, com vista ao desenvolvimento de aplicações de Extração, Transformação e Carregamento de dados (<i>ETL</i>), <i>Data Mining</i> , processamento de <i>streams</i> , computação de grafos, entre outros. O Spark pode também ser utilizado em outros ambientes, ou isoladamente.
Coordenação, monitorização, agendamento e segurança	Zookeeper	Responsável pela coordenação das aplicações distribuídas.
	Ambari	É uma ferramenta <i>Web</i> para a monitorização do <i>cluster</i> Hadoop, desde o estado de funcionamento do mesmo até à análise de desempenho dos seus componentes.
	Oozie	Organiza e agenda tarefas criadas pelos utilizadores de um <i>cluster</i> Hadoop.
	Knox	Providencia autenticação segura no acesso ao <i>cluster</i> Hadoop.
	Sentry	Ao contrário do Knox, que assegura um mecanismo fiável de autenticação, o Sentry disponibiliza mecanismos de atribuição de diferentes permissões/autorizações no acesso granular aos dados.
	Ranger	O projeto Ranger pretende disponibilizar uma ferramenta centralizada de gestão de segurança do <i>cluster</i> Hadoop. Por conseguinte, acaba por englobar mecanismos que já são disponibilizados por outros componentes, como o Sentry.
Acesso aos dados e integração	HBase	Uma base de dados destinada ao armazenamento escalável e distribuído de dados.
	Hive	Disponibiliza capacidades de sumarização de dados e <i>ad hoc querying</i> , de modo muito similar à <i>Structured Query Language (SQL)</i> . É frequentemente associado a uma infraestrutura de <i>Data Warehousing</i> construída sobre o HDFS.
	Pig	É uma linguagem de alto nível destinada à agregação, integração e ordenação dos dados.
	Mahout	Uma biblioteca de <i>Data Mining</i> .
	Sqoop	Transfere, eficientemente, dados de vasta dimensão entre o Hadoop e bases de dados relacionais.
	Flume	Recolhe dados de múltiplas fontes, como <i>logs</i> de servidores <i>Web</i> .

3.2 Bases de Dados *NoSQL* ao Abrigo de Licenças *Open Source*

Como se pôde observar no enquadramento conceptual, lidar com *Big Data* não se centra apenas em novos modelos de programação, como o *MapReduce*, mas sim em toda uma alteração dos requisitos de processamento de dados, devido à sua natureza volumosa, variável e veloz. Consequentemente, através desses requisitos, enumerados por Krishnan (2013), podem-se identificar três mudanças significativas nas estruturas de armazenamento de *Big Data*: o suporte a uma arquitetura de dados *model-less*, a capacidade de partição e replicação e a garantia de uma leitura de dados eficiente.

As bases de dados *NoSQL* surgem frequentemente associadas a soluções de *Big Data*, sendo que a sua crescente adoção não implica o desaproveitamento das bases de dados relacionais (Leavitt, 2010). Cada tecnologia serve propósitos definidos e a notoriedade ganha por estas bases de dados está associada, não exclusivamente, à satisfação dos requisitos do processamento de *Big Data*. No enquadramento conceptual do *NoSQL*, já foi mencionado um conjunto significativo de bases de dados, contudo, este enquadramento tecnológico apenas caracterizará algumas das que estejam ao abrigo de licenças *open source*, devido a permitir uma compreensão algo mais detalhada daquelas que poderão atuar na validação da arquitetura proposta. Por conseguinte, destacam-se as seguintes bases de dados (Baru, Bhandarkar, Nambiar, Poess, & Rabl, 2013; Cattell, 2011; Pokorny, 2013):

1. CouchDB

- a. Desenvolvedor: Apache;
- b. Organização dos dados: Documentos *JavaScript Object Notation (JSON)*;
- c. Principais características:
 - i. Os dados são armazenados em *collections*, que por sua vez são constituídas por documentos. Estes últimos armazenam dados do tipo simples, lista de valores ou outros documentos;
 - ii. Usa Javascript para a formulação de *queries*, cujo processamento pode ser distribuído usando o *MapReduce*. A complexidade associada a este mecanismo de interrogação de dados é suavizada pela disponibilização de uma interface *REST (Representational State Transfer)*, que pode ser utilizada por aplicações programadas em várias linguagens;
 - iii. A escalabilidade é alcançada através da replicação, tendo em consideração que é possível, às aplicações, ler valores desatualizados enquanto as atualizações não são propagadas por todos os servidores;

- iv. Implementa um mecanismo de controlo de versões ao nível dos documentos, detetando atualizações simultâneas aos mesmos, permitindo optar por ações resolventes;
- v. O facto de todas as alterações serem guardadas em disco, escrevendo no final do documento, juntamente com o mecanismo de controlo de versões, proporciona semântica *ACID* ao nível do documento.

2. Redis

- a. Criador: Salvatore Sanfilippo;
- b. Organização dos dados: pares (chave, valor), onde o valor pode ser do tipo simples, lista de valores, entre outros;
- c. Principais características:
 - i. Os dados são armazenados em memória primária, podendo ser copiados para o disco para backup ou manutenção dos servidores;
 - ii. Disponibiliza operações de inserção, remoção e pesquisa, assegurando um elevado desempenho. É reportado que o Redis é capaz de ler ou modificar 100000 registos por segundo, numa implementação suportada por um servidor de 8 núcleos (Cattell, 2011).

3. Cassandra

- a. Desenvolvedor: Apache;
- b. Organização dos dados: grupos de colunas correspondentes a uma chave;
- c. Principais características:
 - i. As alterações aos dados são guardadas em memória e só, posteriormente, armazenadas em disco;
 - ii. Detém mecanismos de partição e replicação. A replicação pode ser assíncrona ou síncrona (Apache, 2015a), dependendo do contexto de aplicação;
 - iii. A detecção e recuperação de falhas é feita de forma automática;
 - iv. Não existência de um ponto de falha único (cada nó do *cluster* tem a mesma função);
 - v. É organizada por *keyspaces*, *column families*, e *columns*.

4. MongoDB

- a. Prestador de suporte: 10Gen;

- b. Organização dos dados: documentos estruturados em objetos e armazenados em coleções;
- c. Principais características:
 - i. Os dados são armazenados de forma similar ao CouchDB;
 - ii. Providencia indexação em qualquer atributo (leitura e escrita), ações contra o *locking* de leitura/escrita e uma linguagem de interrogação de documentos;
 - iii. Disponibiliza *Sharding*, para distribuição de documentos por vários nós;
 - iv. A replicação é sobretudo usada para recuperação a falhas e não para propósitos de escalabilidade;
 - v. A atomicidade é garantida ao nível dos atributos e não ao nível do documento, como no CouchDB.

5. HBase

- a. Desenvolvedor: Apache;
- b. Organização dos dados: Grupos de colunas correspondentes a uma chave;
- c. Principais características:
 - i. Usa o HDFS e, tal como o Cassandra, coloca as alterações aos dados em memória e, periodicamente, armazena-os em disco;
 - ii. Quando surge uma operação de alteração aos dados, essas alterações são armazenadas no final do ficheiro, que é periodicamente compactado. Para além disso, essas mesmas alterações são registadas num ficheiro de *log*, de modo a prevenir futuras falhas;
 - iii. Por defeito, garante atomicidade, através de *locking* e transações ao nível da linha. Contudo as transações podem ocorrer em *scopes* de maior dimensão;
 - iv. Garante partição e distribuição transparente, sem intervenção das aplicações cliente, bem como o suporte de múltiplos *MasterNodes*, de modo a evitar o ponto de falha único. A utilização do *MapReduce* permite a distribuição eficiente das operações;
 - v. O acesso à base de dados pode ser estabelecido não só através de *APIs* Java ou *REST*, mas também através de *Java Database Connectivity (JDBC)* ou *Open Database Connectivity (ODBC)*.

3.3 O Papel das Tecnologias de *ETL* no Processamento de *Big Data*

Uma outra mudança significativa, introduzida pelos requisitos de processamento de *Big Data*, é a transformação mínima dos dados (Krishnan, 2013), nomeadamente no processo de Extração, Transformação e Carregamento (*ETL*). Uma prática que parece ser recorrente no processamento de *Big Data*, é a antecipação da fase de carregamento em relação à fase de transformação (Serra, 2012), assegurando que os dados são manipulados e transformados numa fase estacionária e por tecnologias capazes de lidar com o volume e natureza dos mesmos, como por exemplo o Hadoop. Esta mudança originou o aparecimento do processo de Extração, Carregamento e Transformação (*ELT*).

Quer a fase de transformação ocorra antes ou depois da fase de carregamento, existem algumas tecnologias que merecem ser destacadas pelos esforços realizados na disponibilização de uma forma simples e intuitiva de conduzir um processo *ETL/ELT* de *Big Data*, provando que as mesmas ainda têm um papel relevante dentro do diverso conjunto de tecnologias existentes:

1. Talend Big Data Integration é uma plataforma de integração de dados que se destaca pela significativa cumplicidade com o Hadoop e bases de dados *NoSQL*. A promessa desta tecnologia assenta na disponibilização de um conjunto de instrumentos que simplificam a manipulação de dados em ambientes distribuídos, sem a obrigatoriedade de manter extensos e complicados trechos de código. Por estas razões, o Talend Big Data Integration abstrai os conceitos inerentes ao *MapReduce* ou outras formas de processamento de *Big Data*, num conjunto de cerca de 800 componentes, disponibilizados numa interface gráfica. Esta plataforma mostra uma preocupação constante com a incorporação de vários componentes do Hadoop à medida que vão sendo lançados, tendo rapidamente incorporado o *MapReduce*, *YARN* e *Spark* no seu leque de componentes. Existem vários planos de aquisição desta tecnologia, sendo um deles completamente gratuito, o Talend Open Studio for Big Data (Talend, 2015);
2. Pentaho Big Data Analytics providencia uma plataforma onde as fases do fluxo de processamento de *Big Data* (Krishnan, 2013) são todas contempladas, ou seja, não só o processo de *ETL*, mas também a extração de informação através de *Reporting*, *Data Visualization* e *Data Mining*. Além disso apresenta uma estratégia similar ao Talend Big Data Integration, com um vasto conjunto de componentes para comunicação com o Hadoop e bases de dados *NoSQL*, com técnicas de abstração da complexidade do *MapReduce* e a capacidade de executar tarefas de forma eficiente, tirando partido do

processamento distribuído. Existe também uma versão comunitária, de livre exploração, o Pentaho Big Data Community (Pentaho, 2015).

3.4 **Big Data Analytics**

Em finais dos anos 2000, o termo *Business Analytics* foi introduzido de modo a salientar a importância da componente analítica do *Business Intelligence* (Davenport, 2006). Sendo esta dissertação focada no processamento de *Big Data*, não poderia deixar de ser mencionado o termo *Big Data Analytics*, usado para descrever as técnicas de análise de dados de natureza volumosa e complexa (H. Chen, Chiang, & Storey, 2012).

A presente secção visa a apresentação de dois conceitos frequentemente incorporados no fluxo de processamento de *Big Data*: *Data Mining* e *Data Visualization*. Pretende-se destacar, sumariamente, várias técnicas e tecnologias de *Data Mining* que podem ter lugar entre os componentes de uma arquitetura de *Big Data*, bem como apresentar algumas ferramentas destinadas à visualização de dados, capazes de se conectarem a tecnologias de armazenamento de *Big Data* e apresentar os resultados de forma adequada.

3.4.1 **Técnicas e Ferramentas de Data Mining**

Data Mining pode ser definido de várias formas, devido à sua natureza interdisciplinar, mas pode ser sobretudo entendido como a atividade de extrair conhecimento dos dados. Outros termos são frequentemente usados para referir *Data Mining*, como por exemplo: *knowledge mining from data*, *knowledge extraction*, *data/pattern analysis*, *data archaeology* e *data dredging*. Não fossem estes demasiados termos, alguns autores referem que *Data Mining* é um sinónimo do popular *Knowledge Discovery from Databases (KDD)*, enquanto outros afirmam que *Data Mining* é uma fase do *KDD* (Han, 2011).

Dentro do conjunto de técnicas de *Data Mining* podem-se destacar:

1. Padrões frequentes, como por exemplo regras de associação, onde se analisa padrões frequentes nos dados, tais como “se um consumidor compra determinado produto também compra outros” (Han, 2011);
2. Classificação e Regressão, destinados à previsão de valores nominais ou numéricos, respetivamente, com base num modelo treinado num conjunto de dados, designados de dados de treino (Han, 2011);

3. *Clustering*, destinado a atribuir um grupo a cada instância dos dados, tentando segmentá-los de forma a que a similaridade entre instâncias de um grupo seja máxima e a similaridade entre grupos seja mínima (Han, 2011);
4. *Time series forecasting*, capaz de prever valores futuros tendo em consideração a natureza temporal dos dados, ou seja, pressupondo que os valores aparecem ordenados no tempo (Gama, 2010).

Existem várias ferramentas de *Data Mining* que podem ser usadas livremente. A Tabela 4 apresenta algumas dessas ferramentas, descrevendo sumariamente cada uma delas.

Tabela 4. Descrição de algumas ferramentas de *Data Mining*.

WEKA	Desenvolvido na universidade de Waikato, tornou-se uma ferramenta amplamente usada e reconhecida, disponibilizando um vasto conjunto de algoritmos de classificação, regressão, <i>clustering</i> e regras de associação (Wahbeh, Al-Radaideh, Al-Kabi, & Al-Shawakfa, 2011). Nas últimas versões do <i>software</i> foi também adicionada a técnica de <i>time series forecasting</i> e atualmente abrange formas de distribuir processamento num <i>cluster</i> Hadoop (WEKA, 2015).
Orange	Inclui um conjunto diversificado de algoritmos, acrescido de componentes para <i>input</i> e transformação de dados. O seu modo de utilização assenta numa forte componente visual (Wahbeh et al., 2011).
RapidMiner	Tem como objetivo oferecer uma experiência de utilização sem a necessidade de escrever qualquer trecho de código. Entre as fontes de dados que podem ser utilizadas já se encontram alguns componentes do Hadoop, como o Hive e o HDFS, ou bases de dados <i>NoSQL</i> , como o Cassandra e o MongoDB. No entanto, a versão livre do <i>software</i> não detém estas funcionalidades (RapidMiner, 2015).
R	É um ambiente gratuito de estatística e análise visual de dados. Um dos pontos fortes do R é o <i>Comprehensive R Archive Network (CRAN)</i> , incluindo cerca de 6000 <i>packages</i> desenvolvidos pela comunidade científica e técnica. Este relevante contributo permite a utilização das mais variadas técnicas de <i>Data Mining</i> (RDataMining, 2015).
Knime	Detém componentes de integração, processamento e análise de dados, que podem ser utilizados através de uma interface gráfica para a criação de fluxos de dados (Wahbeh et al., 2011).
Spark e Mahout	São ferramentas integradas no Hadoop (embora o Spark também possa ser executado isoladamente), como já foi mencionado anteriormente, e cuja utilização é adequada em contextos de extração de conhecimento em conjuntos de dados de vasto volume, nos quais as ferramentas acima identificadas não conseguem atuar devido às limitações do processamento <i>single-node</i> . Note-se que nem sempre o Spark e Mahout são necessários quando se tenta extrair conhecimento de <i>Big Data</i> , pois muitas das vezes uma agregação adequada, ajustando o nível de granularidade, usando, por exemplo, o Pig, reduz o volume dos dados para um nível adequado às ferramentas acima identificadas. Contudo, em situações nas quais a agregação não se aplica, apenas ferramentas de <i>Data Mining</i> distribuído asseguram a extração de conhecimento em conjuntos de dados de elevado volume.

3.4.2 *Data Visualization*

Como o nome indica, este é o processo de expressar, através de uma análise visual, os resultados de um fluxo de processamento de dados. Atualmente, uma das particularidades das tecnologias de *Data Visualization* em comparação com as tecnologias tradicionais de *Business Intelligence* é a capacidade de “integrar dados provenientes de várias fontes em apenas uma imagem, interligando dados comuns entre diferentes conjuntos” (Krishnan, 2013). Este facto deve-se à natureza do *Big Data*, cuja manipulação (*drilldowns*, *rollups*, entre outros) não é apropriada na fase de análise visual dos dados. Diferentes tecnologias demonstram particular atenção no *Big Data Visualization*, destacando-se o Datameer, Tableau e Spotfire, como soluções comerciais, e o R como solução *open source*, embora o seu uso não seja destinado explicitamente para este fim (Krishnan, 2013). Como outras soluções *open source*, existem também bibliotecas destinadas à criação de gráficos, como por exemplo o D3.js e o Google Charts.

Capítulo 4. BASIS: uma Arquitetura de *Big Data* para *Smart Cities*

Antes de ser apresentada a arquitetura BASIS¹, convém destacar os fatores que levaram ao desenvolvimento da mesma. As quatro arquiteturas analisadas no enquadramento conceptual, apresentado no capítulo 2, demonstraram diferentes abordagens para lidar com a emergente preocupação associada à recolha, processamento e análise de um vasto volume e variedade de dados, gerados a elevadas velocidades no contexto de uma *Smart City*.

Entre o conjunto das arquiteturas analisadas, dois dos autores adotam uma abordagem centrada nos dados gerados por sensores (*Smart Meters*) (Girtelschmid et al., 2013; Saragih with Suakanto & Supangkat, Suhardi, 2013). Apesar de ser possível conduzir observações e retirar conclusões acerca dos aspetos conceptuais das respetivas soluções, é particularmente difícil transportar os resultados obtidos para o contexto de uma *Smart City*, onde os dados provêm de diversas fontes. No âmbito desta dissertação espera-se a recolha, armazenamento e processamento eficiente dos dados, bem como a sua disponibilização de uma forma aberta, conduzindo à criação de serviços inteligentes. Embora tenha sido relevante analisar as decisões tomadas nas duas arquiteturas acima mencionadas, e se reconheça, segundo os autores, alguns resultados satisfatórios das respetivas implementações, as mesmas servem propósitos muito específicos, que apenas cobrem parte dos objetivos de uma arquitetura capaz de lidar com *Big Data* no contexto de uma *Smart City*. Outra lacuna de significativa relevância é a demasiada abstração da componente tecnológica, levando à incerteza quando se tenta observar a preocupação com o armazenamento e processamento distribuído dos dados.

Contudo, Khan et al. (2013) apresentam um estudo significativamente relevante no que respeita à disponibilização dos dados num modelo “*Analytics as a Service*”, prestando particular atenção à descoberta de relações e extração de informação nos dados. Os autores demonstram alguma preocupação com a escalabilidade e latência da solução apresentada, bem como a necessidade de utilizar *open standards*, embora não sejam muito explícitos quanto a estes aspetos, nem os façam refletir de forma clara na arquitetura. Outras características apresentadas pelos autores consistem em: preocupação com o correto armazenamento dos metadados, eliminando assim a ambiguidade associada ao frequente volume e variedade do *Big Data*; a heterogeneidade das fontes, destacando a importância de recolher dados provenientes de *APIs*, *Web services* ou conectores especialmente desenvolvidos. No

¹ A designação BASIS deriva de *Big Data architecture for Smart Cities*, usando principalmente as iniciais de cada palavra, acabando por formar a palavra inglesa *basis*, que vai de encontro à ambição de fornecer um contributo significativo na procura de uma arquitetura padrão nestes contextos.

entanto, é discutível a referida importância de encontrar relações nos dados armazenados, pois o termo *Big Data* tende a caracterizar dados de natureza variada, cuja relação não costuma ser encontrada diretamente, cruzando de um modo relacional os *datasets*, mas sim usando um conjunto de técnicas de *Big Data Analytics*. Por conseguinte uma camada dedicada ao mapeamento e criação de relações parece algo desadequada, já que não é possível relacionar, de forma direta, o consumo de eletricidade de cada casa com o plano de tarifas mais adequado, apenas para dar um exemplo. No entanto, o nível de importância desta camada varia com a natureza dos dados que são recolhidos e armazenados.

Por fim, Vilajosana et al. (2013) apresentam uma arquitetura cujo objetivo assenta adequadamente num dos principais objetivos desta dissertação: a disponibilização de serviços inteligentes, recorrendo ao armazenamento e processamento de *Big Data* gerado numa *Smart City*. Os autores são bastante explícitos no que respeita à importância do *Open Data* e à cooperação entre intervenientes no desenvolvimento das iniciativas de *Smart Cities*. Observa-se novamente a importância dos *Web services*, desta vez numa ótica de disponibilização dos dados, acompanhada de um conceito significativamente relevante no âmbito do *Big Data*, o conceito de *streaming*. Além disso a arquitetura espelha particular preocupação, embora reduzida, com a segurança e armazenamento escalável. No âmbito desta dissertação, considera-se que a solução apresentada por Vilajosana et al. (2013) revela-se algo adequada ao cumprimento dos objetivos propostos, embora a falta de detalhe tecnológico seja um entrave à sua plena adoção/personalização. Tal como referido pelos autores, a solução deve ser vista como um modelo conceptual que sustenta o desenvolvimento de iniciativas de *Smart Cities*.

Devido a estas conclusões, retiradas da análise das quatro arquiteturas, pode-se afirmar que a arquitetura proposta de seguida não é fruto de alterações a nenhuma das soluções existentes, mas sim uma arquitetura criada de raiz, que adota contudo, algumas linhas de pensamento aqui discutidas, fruto do contributo dos autores supra mencionados. A arquitetura BASIS visa eliminar a ambiguidade tecnológica presente nas quatro arquiteturas analisadas, ao mesmo tempo que cumpre os princípios estabelecidos, destacando a relevância das tecnologias de armazenamento, processamento e análise de *Big Data*. Para tal, adotou-se uma abordagem por camadas de abstração: camada conceptual, camada tecnológica, camada infraestrutural e intervenientes na cocriação de serviços.

4.1 Camada Conceptual

A primeira camada da arquitetura, designada de conceptual, visa encapsular os vários componentes que desempenham um papel crucial, desde a recolha do *Big Data* até à sua análise e disponibilização. A Figura 26 ilustra a camada conceptual da arquitetura BASIS.

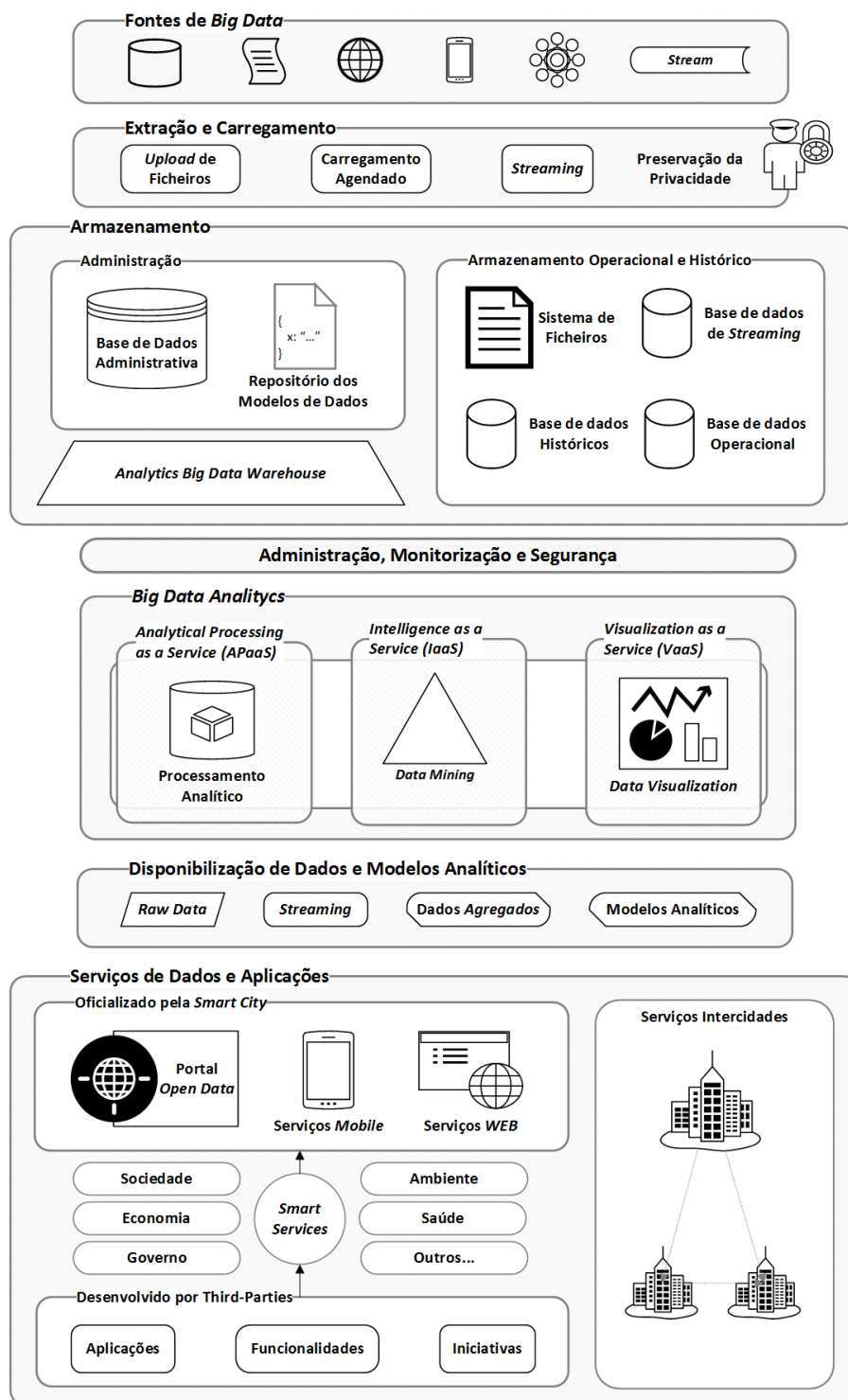


Figura 26. Camada conceptual da arquitetura BASIS.

Numa *Smart City* existem inúmeras **fontes de dados**, mas é possível identificar algumas que se destacam: bases de dados, ficheiros, *APIs Web* (redes sociais, *feeds* de notícias, entre outros...), *smartphones* (exemplo: localização), sensores (exemplo: *Smart Meters*) e *streams*. Embora nenhuma solução deva ser concebida de modo a ser dependente de qualquer fonte de dados ou dispositivo, estas devem ser fontes à qual deve ser dada particular atenção.

De modo a conseguir extrair os dados, que naturalmente serão o “combustível” desta solução, designaram-se três formas de **extração e carregamento**, tal como é representado na Tabela 5.

Tabela 5. Formas de extração e carregamento dos dados.

<i>Upload</i> de ficheiros	<p>Esta forma de recolha de dados desempenhará um papel de elevada importância na dinâmica da <i>Smart City</i>. É através do <i>upload</i> de ficheiros que os vários cidadãos e organizações poderão contribuir para o aumento do volume e variedade dos dados disponíveis. Na submissão do ficheiro deverá ser fornecida informação relevante, tais como os metadados apropriados, a possibilidade de substituição de um ficheiro mais antigo, a garantia de privacidade de certos dados, entre outros. A sua aprovação será, obviamente, assegurada pelas entidades gestoras responsáveis.</p> <p>Um exemplo comum poderia ser a submissão dos pontos de atratividade turística, ou então, a submissão dos crimes registados na cidade. Estes são pequenos exemplos das possibilidades que se abrem ao permitir este nível de interação.</p>
Carregamento agendado	<p>Uma recolha de dados assente numa base periódica, ou seja, os dados serão iterativamente recolhidos e o intervalo entre as recolhas será apropriadamente definido. Esta forma de recolha adequa-se a dados em constante alteração, mas que não necessitam de uma extração em tempo real. Tipicamente assenta naquilo que conhecemos como sendo o <i>ETL/ELT</i>.</p>
<i>Streaming</i>	<p>É o mecanismo de extração mais peculiar e com mais destaque nos dias de hoje. Assenta num fluxo contínuo de dados, ou seja, numa conexão permanente entre a origem e o destino. Atualmente, com a crescente popularidade do <i>Big Data</i>, pode-se recolher continuamente o que as pessoas publicam nas redes sociais, dados de <i>Smart Meters</i>, entre muitos outros. Os dados das redes sociais e dos sensores podem ser considerados, ou não, como <i>streams</i>, dependendo da forma como a solução é implementada. No entanto a <i>stream</i> surge como uma fonte de dados separada, devido às particularidades técnicas que esta acarreta.</p>

Uma das principais preocupações a ter em conta no momento da **extração e carregamento dos dados** é assegurar a privacidade dos mesmos, quando aplicável, garantindo que informação sensível não é armazenada. A desidentificação dos dados via randomização ou encriptação parece ser a solução mais viável, pois na maioria das vezes (senão sempre) o processamento ou disponibilização dos atributos que levantam preocupações com a privacidade dos dados não são necessários no contexto de uma *Smart City*.

Na componente de **armazenamento** estão designados os principais repositórios para o correto funcionamento da solução. Com vista a uma adequada gestão do sistema, será necessário armazenar dados de carácter administrativo, isto é, ciclo de vida dos dados, controlos de acesso, permissões, mapeamento dos nós da rede, *logs*, entre outros. O facto de ser designada uma base de dados administrativa não significa que os dados tenham de estar centralizados numa única base de dados, mas apenas remete para a importância de aceder, recolher e armazenar todos os dados administrativos para o correto funcionamento dos vários componentes do sistema. Acrescentada a esta preocupação deve surgir a correta especificação e armazenamento dos modelos de dados, estejam eles armazenados em ficheiros ou bases de dados, pois só assim se garante a sustentabilidade da disponibilização pública dos dados. Ainda na componente de armazenamento, destacam-se:

1. Armazenamento operacional
 - a. Sistema de Ficheiros - responsável pelo armazenamento dos ficheiros recolhidos por *upload*;
 - b. Base de dados operacional - armazena os dados disponibilizados nos serviços de dados e aplicações;
 - c. Base de dados históricos - A sua utilização depende do volume de dados presente na base de dados operacional, isto é, caso este comece a afetar seriamente o desempenho dos serviços de dados e aplicações, deve-se considerar a migração dos dados mais antigos para um conjunto de nós exclusivamente responsáveis pelo armazenamento dos mesmos. Isto assegura uma gestão eficiente do ciclo de vida dos dados, que pode ser parametrizável, alterando o intervalo de tempo para o qual ocorre o particionamento dos dados;
 - d. Base de dados de *streaming* - é uma base de dados responsável pelo armazenamento dos dados provenientes de *streams*, cujos requisitos-chave passam pela latência de *I/O* e *throughput* notoriamente reduzidos.
2. *Analytics Big Data Warehouse* - deve ser utilizado para fins de análise de dados por parte da gestão da *Smart City*. É necessário como fonte de dados das várias ferramentas de *Data Visualization*, pois o armazenamento operacional nem sempre se revela o mais adequado para análises desta natureza.

No que respeita a **administração, monitorização e segurança**, esta é uma componente particularmente subjetiva, na medida em que estará quase totalmente dependente da tecnologia usada

para instanciar esta arquitetura. Esta camada inclui a administração e monitorização, não só dos *clusters* de armazenamento e de processamento, mas também dos controlos de acesso, permissões, mecanismos de desidentificação, *logs* dos servidores, entre outros aspetos de carácter infraestrutural.

Uma das componentes de maior destaque nesta arquitetura, a par com o armazenamento, é o ***Big Data Analytics***, na medida em que o mesmo é responsável pela manipulação e processamento dos dados, incluindo técnicas como *Data Mining* e *Data Visualization*, de forma a criar serviços que denotem inteligência providenciada pela capacidade de computação atualmente disponível. Nesta componente são designados alguns subcomponentes que conduzem o processo de disponibilização de serviços inteligentes aos cidadãos, desde o pré-processamento dos dados (*Data Cleansing*, transformações, agregações...) até à visualização dos resultados finais, que podem decorrer, ou não, da utilização de avançadas técnicas e algoritmos de *Data Mining*. Um dos princípios de *design* fundamentais, que poderá fomentar a criação de serviços inteligentes por parte da comunidade, é a disponibilização de cada um destes subcomponentes numa ótica “*as a Service*”, eliminando assim a elevada curva de aprendizagem frequentemente associada a estas tecnologias. Cada um dos componentes de *Big Data Analytics* poderá ser disponibilizado como um serviço no portal *Open Data*, seja através de interação direta, isto é manipulando o conjunto de dados diretamente no portal, ou através de *APIs Web*, com vista a serem utilizadas por aplicações externas, desenvolvidas por terceiros.

Todos os dados recolhidos pela solução, ou pelo menos aqueles que sejam entendidos como adequados ao paradigma *Open Data*, devem ser adequadamente disponibilizados através do uso de *APIs Web*, para um acesso independente da plataforma que requiere os respetivos dados. Esta preocupação reflete mais um dos princípios de *design* fundamentais, pois caso não se criem formas de disponibilização de dados independentes das plataformas que pretendem aceder aos mesmos, incorre-se num elevado custo (recursos humanos, tecnológicos e monetários) na altura de acrescentar funcionalidades ou substituir componentes do sistema. Uma das formas de garantir uma arquitetura *service-oriented* é a correta utilização de *Web services* para recolha e disponibilização de dados provenientes da *Smart City*, quando aplicável, o que nem sempre é o caso do carregamento de dados agendado. Os tipos de conjuntos de dados a serem disponibilizados são identificados na componente de **disponibilização de dados e modelos analíticos**:

1. *Raw data* - dados extraídos diretamente do sistema de ficheiros, base de dados operacional ou base de dados históricos;

2. *Streaming* - disponibilização de uma *API* de acesso à recolha em tempo real, ou seja, diretamente ligado à conexão responsável pela recolha dos dados, ou então disponibilização dos dados armazenados na base de dados de *streaming*. A decisão de disponibilizar este tipo de conjuntos de dados de uma ou de outra forma, ou das duas em simultâneo, parte dos compromissos que a cidade queira tomar em termos infraestruturais, pois a disponibilização dos dados em tempo real acarreta algum desenvolvimento adicional e uma adequada infraestrutura de suporte, como será demonstrado na secção “Camada Infraestrutural”;
3. Dados agregados - resultantes de processos que ocorreram na subcomponente de processamento analítico e cujas percepções são relevantes o suficiente para serem disponibilizados como um conjunto de dados *Open Data*;
4. Modelos analíticos - são o resultado da aplicação de técnicas de *Data Mining* a qualquer conjunto de dados armazenados pelas bases de dados ou sistema de ficheiros. À semelhança dos dados agregados, entende-se que estes resultados são relevantes para a criação de conhecimento na sociedade.

Por fim, a correta integração dos vários componentes é espelhada na camada de **serviços de dados e aplicações**. Criando uma arquitetura orientada para os serviços, consegue-se promover o desenvolvimento de serviços internos e externos, sem que seja necessário proceder a alterações significativas na componente de recolha, armazenamento e processamento de dados. No que respeita aos serviços internos, estão presentes: o portal *Open Data* onde a comunidade poderá fazer *download* dos dados, em vários formatos, e aceder a *APIs Web*, que possibilitam não só a incorporação dos dados em aplicações de terceiros, mas também a utilização da componente de *Big Data Analytics as a Service*; o serviço *Web*, que disponibiliza um conjunto de visualizações de dados, provenientes de várias fontes e resultantes dos vários subcomponentes do *Big Data Analytics*, de modo a ser acedido pelos cidadãos para um aumento do conhecimento e envolvimento cidadão; o serviço *Mobile*, como uma oferta similar ao serviço *Web*, mas com a particularidade de ser adequadamente direcionado para os dispositivos móveis. Contudo, a arquitetura contempla, como já referido, a importância do desenvolvimento de soluções por parte da comunidade (aplicações, funcionalidades, iniciativas), pelo que a disponibilização dos dados e funcionalidades de uma forma aberta é orientada para o aparecimento destas soluções, que poderão posteriormente ser integradas nos serviços suportados oficialmente pela *Smart City*. Em último lugar, destaca-se a interação entre diferentes *Smart Cities*, cuja parceria pode resultar em serviços a nível

nacional ou regional, potenciados sobretudo pelo elevado grau de interoperabilidade assegurado pela arquitetura aqui proposta.

4.2 Camada Tecnológica

Não obstante ao coeso contributo presente nos vários componentes da camada conceptual da arquitetura, é significativamente relevante, que se faça a sua interligação com as possíveis tecnologias que poderão desempenhar um papel relevante na implementação da arquitetura BASIS. Um dos principais aspetos de criticismo face às arquiteturas propostas por outros autores é a falta de detalhe tecnológico, daí que se tenha concebido esta camada com ênfase predominante na tecnologia de armazenamento, processamento e análise de *Big Data*. Tal como demonstra a Figura 27, a camada tecnológica contempla a recolha de dados provenientes de várias fontes, cada uma delas com diferentes necessidades de armazenamento e processamento em termos de volume e velocidade.

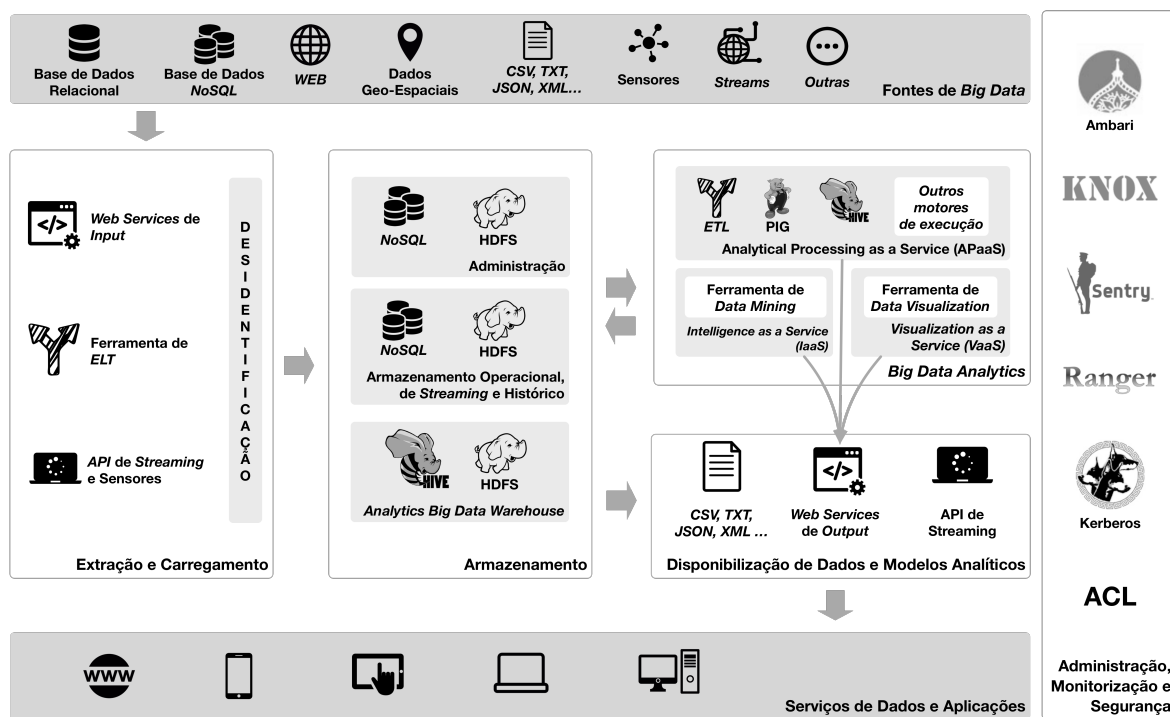


Figura 27. Camada tecnológica da arquitetura BASIS.

Deve ser tido em conta que as tecnologias aqui ilustradas não representam uma “única versão da verdade”, isto é, a arquitetura está concebida de forma a que a sua implementação possa ser conduzida usando as tecnologias que se revelem mais adequadas a cada caso, pois enquanto numa certa iniciativa um dos requisitos pode ser o pleno uso de tecnologias *open source*, noutra iniciativa esse mesmo requisito pode não estar presente. Por conseguinte, esta camada ilustra várias tecnologias que

poderão satisfazer as necessidades da camada conceptual, deixando em aberto algumas possibilidades, como, por exemplo, as ferramentas de *Data Mining*, de *Data Visualization* e a base de dados *NoSQL*.

De acordo com as especificidades de cada fonte de dados, deve ser utilizada a forma de extração e carregamento mais adequada, ou seja: a submissão de um ficheiro por parte da comunidade pode ser efetuada via *Web service* disponibilizado no portal *Open Data*; a leitura de bases de dados ou determinadas *APIs Web* pode ser assegurada periodicamente através de uma ferramenta de *ETL/ELT* (foco no carregamento sem transformações de elevado custo computacional, tal como Krishnan (2013) menciona), cuja orientação seja focada para o *Big Data*, como é o caso das duas ferramentas descritas no enquadramento tecnológico; finalmente, na recolha de dados de sensores ou por *streaming* é adequada a utilização de conexões permanentes que recebem dados à medida que os mesmos são gerados, quaisquer que seja a respetiva granularidade temporal (segundo, minuto, hora...). Na fase de extração e carregamento dos dados, qualquer das três formas, acima descritas, deve assegurar a desidentificação dos dados quando necessário, fazendo uso de mecanismos de encriptação tais como um *hash MD5*, ou randomizando os dados sem qualquer lógica associada, evitando assim a reidentificação de atributos que violem a privacidade pessoal ou organizacional.

No que respeita ao armazenamento, visto que são propostas diferentes tecnologias, identifica-se, na Tabela 6, o tipo de dados associados a cada uma delas. Estas tecnologias foram designadas para atuar na componente de armazenamento dos dados, de modo a assegurar uma total e eficiente utilização das vantagens do *Big Data* e, consequentemente, garantir a sustentabilidade e escalabilidade da solução, a longo prazo.

Tabela 6. Tipos de dados associados às tecnologias de armazenamento da arquitetura BASIS.

HDFS	NoSQL	Hive
<ul style="list-style-type: none"> Dados administrativos (ex: <i>logs</i> de servidores...); Modelos de dados (metadados); Ficheiros submetidos pelos cidadãos ou organizações; Resultados provenientes do <i>Big Data Analytics</i>, cuja consulta seja esporádica ou cujo tempo de vida seja reduzido. 	<ul style="list-style-type: none"> Dados administrativos (ex: chaves de acesso às <i>APIs</i>, caso aplicável, parâmetros associados ao ciclo de vida dos dados...); Dados provenientes de <i>streams</i>; Dados operacionais, utilizados pelos serviços <i>Web</i> e <i>Mobile</i>. Estes dados incluem também resultados do <i>Big Data Analytics</i>, mas de uma forma mais orientada para a integração com as aplicações que suportarão os serviços <i>Web</i> e <i>Mobile</i>; Dados históricos. 	<ul style="list-style-type: none"> Resultados provenientes do <i>Big Data Analytics</i>, que se destinam a ser consultados, recorrentemente, pela gestão da cidade (ex: previsões de consumos energéticos, número de acessos diários à instalação da câmara municipal...).

Embora à primeira vista a adoção de papéis definidos para cada uma das tecnologias de armazenamento possa parecer algo desnecessária, deve ser tido em consideração a variedade dos dados, que rapidamente pode incutir dificuldades na percepção do mapeamento dos recursos informacionais, levando a que as entidades internas responsáveis pelo *Big Data Analytics* desconheçam, por vezes, o local de armazenamento para cada tipo de dados. Assim, pretende-se eliminar esta dificuldade.

Um significativo contributo para o aparecimento de novos serviços inteligentes por parte da comunidade passaria pela disponibilização das técnicas de processamento analítico, *Data Mining* e *Data Visualization*, num paradigma “*as a Service*”, o que permitiria à comunidade, não só utilizar o portal *Open Data* para analisar e processar os dados, utilizando as técnicas mencionadas, mas também incorporar as referidas técnicas no desenvolvimento de aplicações, através de *Web services*. Tal como identificado por Khan et al. (2013), esta componente analítica disponibilizada à comunidade constituiria uma possibilidade com um enorme potencial de criação e dissipação de conhecimento público, fomentando a cocriação de serviços. Por conseguinte, já se consegue entender o papel crucial do *Big Data Analytics* nesta arquitetura, pois para além das técnicas e ferramentas envolvidas nesta componente possibilitarem a exploração e processamento dos dados por parte dos *Data Scientists* e *Analysts* que integram a equipa de TI da *Smart City*, permitem também que as suas funcionalidades possam ser abstraídas a um determinado nível que seja entendido e utilizável por qualquer cidadão ou organização. Esta disponibilização das funcionalidades do *Big Data Analytics* poderá ser feita no âmbito do Portal *Open Data*, juntamente com a simples exploração e *download* dos vários conjuntos de dados nele disponíveis.

Tal como referido anteriormente, embora na Figura 27 sejam instanciados alguns componentes com tecnologias específicas, nem sempre essa instanciação é adequada, como é o exemplo da tecnologia de base de dados *NoSQL* a usar. Cada base de dados *NoSQL* serve propósitos específicos, cada uma com características particulares, tal como identificado no enquadramento tecnológico, mas também com objetivos bastante similares, sobretudo a flexibilidade dos modelos de dados e a elevada *performance*. Assumindo que o *benchmarking* de bases de dados *NoSQL* não se adequa ao âmbito desta dissertação, nem a sua comparação direta, opta-se por deixar a tecnologia *NoSQL* como uma variável da arquitetura. A adoção de uma determinada base de dados poderá ter como fatores influenciadores, a experiência da equipa de TI, a constituição do *cluster* de armazenamento, isto é, questões que variam com cada iniciativa em concreto. No entanto, observando as diferenças e particularidades de cada uma

das bases de dados estudadas no enquadramento tecnológico, consegue-se justificar a adoção de uma, ou várias, num determinado contexto, conforme os requisitos em questão.

Ainda no que diz respeito à componente de *Big Data Analytics*, nomeadamente ao processamento analítico, entre os outros motores de execução que podem ser utilizados, pode-se destacar o Spark, cuja popularidade tem vindo a aumentar consideravelmente, ou qualquer outra linguagem de programação, ou motor de transformação de dados, desde que sejam adequados à natureza dos dados que estejam a ser processados.

Tal como foi apresentado no enquadramento tecnológico, existem várias ferramentas de *Data Mining* e *Data Visualization*, que podem ser utilizadas para instanciar os respetivos componentes da arquitetura, de forma a que se revelem adequadas para a iniciativa que esteja a ser desenvolvida. Novamente, verifica-se a flexibilidade na seleção da tecnologia em específico, de modo a não condicionar o apropriado desenvolvimento das iniciativas que adotem a arquitetura aqui proposta.

A arquitetura permite a disponibilização dos dados através de três formas:

1. *Download* de ficheiros de vários formatos (*JSON, CSV, TXT...*), que pode ser efetuado no Portal *Open Data*;
2. Acesso por *Web service*, permitindo que qualquer serviço ou aplicação, desenvolvida internamente ou pela comunidade, aceda aos dados armazenados. A disponibilização do endereço de acesso ao *Web service* deverá ser também realizada através do Portal *Open Data*;
3. *API* de leitura de *streams*, cujo acesso é também assegurado sobre primitivas *Hypertext Transfer Protocol (HTTP)* e garantindo consequentemente a filosofia *service-oriented*, tal como acontece no acesso por *Web service*.

Fica assim assegurado o acesso independente e a facilidade de integração dos vários serviços de dados e aplicações, incluindo o Portal *Open Data*, o serviço *Web*, o serviço *Mobile*, os serviços intercidades e os serviços desenvolvidos por *Third-Parties*.

Por último, a componente de administração, monitorização e segurança supervisiona todas as operações dos vários componentes do sistema, numa ótica de garantir o correto funcionamento da tecnologia e a utilização segura da mesma. Grande parte das tecnologias associadas são intrínsecas ao Hadoop (Ambari, Knox, Sentry, Ranger), mas outras como o Kerberos e *Access Control Lists (ACL)*

podem ser utilizadas quer no âmbito do Hadoop, quer no âmbito de outras tecnologias. Para além disso, deve ser também considerada a adequada utilização das funcionalidades de monitorização da base de dados *NoSQL* a usar, pois embora não esteja particularmente identificada nesta componente, pressupõe-se a sua correta utilização, para uma maior eficiência na gestão da infraestrutura. As tecnologias indicadas nesta componente de administração, monitorização e segurança devem ser vistas como um ponto de referência e não como um dado adquirido. Dois factores de significativa relevância na altura de atribuir responsabilidades nesta vertente é a necessidade de elevada experiência em administração infraestrutural de *Big Data* e a compreensão e atuação em segurança de sistemas.

4.3 Camada Infraestrutural

Entre as várias camadas da arquitetura, a camada infraestrutural, apresentada na Figura 28, consiste na representação mais aproximada ao nível físico da solução, isto é, a forma como os vários componentes de *hardware* atuarão em conjunto para garantir a satisfação dos pedidos externos.

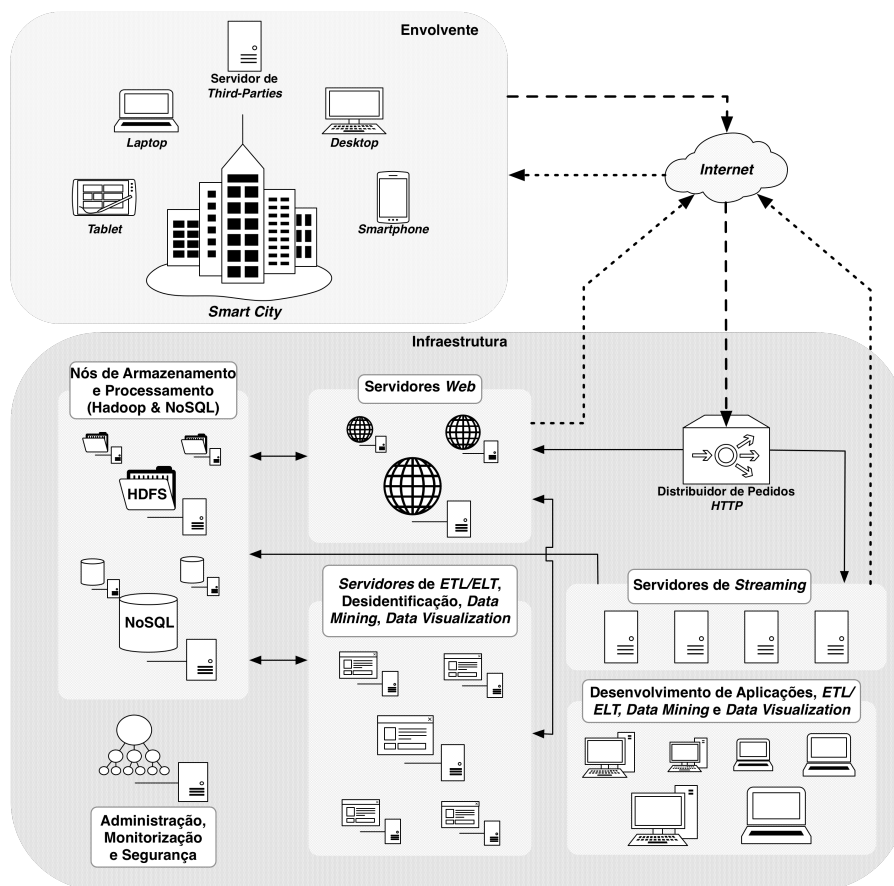


Figura 28. Camada infraestrutural da arquitetura BASIS.

Esta conceptualização da infraestrutura inclui nos seus princípios o conceito de *commodity hardware*, herdado da adoção de tecnologias como o Hadoop e *NoSQL*. Isto significa que apesar da infraestrutura representar um dos maiores (senão o maior) custo na altura de adotar uma solução assente nesta arquitetura, consegue-se ainda assim diminuir significativamente os gastos frequentemente associados à implementação da solução, construindo *clusters* de processamento e armazenamento que utilizam *hardware* de reduzido custo. Em jeito de término quanto às observações que concernem o custo da camada infraestrutural, pode-se ainda referir que a elevada escalabilidade assegurada pelo Hadoop e *NoSQL* cria oportunidades para que as infraestruturas possam ser partilhadas entre várias cidades, pois muito certamente as iniciativas de *Smart Cities* demoram algum tempo até que se atinja o ponto de maturidade e, consequentemente, elevado volume e variedade de dados. Assim sendo, uma implementação semelhante em duas cidades diferentes, pode ser algo a evitar. Estas são decisões que cabem à gestão da cidade tomar, tendo em conta o investimento que esta está disposta a fazer, as vantagens e desvantagens de partilhar infraestrutura, entre outras decisões. No entanto, como o princípio da escalabilidade é um dado adquirido nesta arquitetura, deve-se destacar esta possibilidade, reduzindo assim mais uma vez os custos infraestruturais, no caso de uma infraestrutura partilhada por duas ou mais cidades.

Apesar da conceptualização de uma infraestrutura de *Big Data* no contexto de uma *Smart City* poder parecer uma tarefa de elevada complexidade, a verdade é que a adoção das tecnologias adequadas, como o Hadoop e o *NoSQL*, retiram uma significativa carga de trabalho na hora de conceber a camada infraestrutural, muito devido ao facto de abstraírem a complexidade frequentemente associada à computação distribuída. Por isso, de modo a tornar a camada infraestrutural ainda mais simples de entender, a mesma foi concebida segundo um modelo pedido-resposta. A camada infraestrutural encontra-se assim dividida nos seguintes componentes, que interagem após um pedido efetuado pela envolvente, com vista a retornar uma resposta:

1. Distribuidor de pedidos *HTTP* - este componente é responsável pelo balanceamento dos pedidos provenientes da envolvente, que podem consistir em pedidos aos servidores *Web* ou aos servidores de *streaming*. A sua representação arquitetural, tal como está, não significa que deva ser implementado como um componente único, até porque o distribuidor de pedidos estará dependente da tecnologia usada para constituir o conjunto de servidores acima identificados;

2. Servidores *Web* - a sua tarefa é disponibilizar à envolvente o portal *Open Data*, o serviço *Web* e todo o conjunto de *Web services* que serão utilizados para *input* e *output* de dados ou funcionalidades (*"Analytics as a Service"*). Não se revela necessário a predeterminação de qualquer tecnologia de implementação, desde que as opções tomadas não condicionem a facilidade de interação com qualquer dispositivo da envolvente. Contudo recomenda-se a utilização de *Web services REST*, devido à utilização exclusiva de primitivas *HTTP*, facilitando a sua integração em aplicações;
3. Servidores de *streaming* - estes servidores dedicam-se à leitura de dados provenientes de *streams* e à escrita desses mesmos dados na componente de armazenamento. Além disso, caso seja adequada a disponibilização desses dados publicamente, os mesmos poderão ser imediatamente disponibilizados a todos os clientes conectados aos servidores através de uma conexão *HTTP* permanente. No entanto, como já foi mencionado, a complexidade associada a este tipo de disponibilização de dados em tempo real é significativamente maior do que a leitura periódica da base de dados de *streaming* e consecutiva disponibilização pública. Esta é uma decisão a ter em conta na fase de implementação da solução. Contudo, caso a disponibilização pública dos dados provenientes de *streams* seja feita em tempo real, ou não, estes servidores estarão encarregues dessa tarefa;
4. Nós de armazenamento e processamento (Hadoop e *NoSQL*) - são o ponto fulcral desta infraestrutura. A escalabilidade linear e a eficiência no processamento e armazenamento de *Big Data* asseguram uma *performance* adequada para os serviços de dados e aplicações. Constituem o repositório de dados do sistema, disponibilizando o armazenamento e também o processamento distribuído desses mesmos dados, de forma a garantir elevado desempenho e tolerância a falhas;
5. Servidores de *ETL/ELT*, desidentificação, *Data Mining* e *Data Visualization* - garantem o adequado fluxo dos dados entre as tecnologias integrantes da infraestrutura, para além de executarem as técnicas de desidentificação (randomização, encriptação) dos dados, quando necessário. As suas tarefas incluem também a execução do *Data Mining* e do *Data Visualization*, quer numa ótica de soluções desenvolvidas internamente, quer para entidades externas, segundo o modelo *"as a Service"*;
6. Desenvolvimento de aplicações, *ETL/ELT*, *Data Mining* e *Data Visualization* - a sua presença na infraestrutura assume um papel de experimentação e desenvolvimento, que

normalmente antecipam a implementação de qualquer solução. Estes são recursos internos, usados apenas por *Data Scientists/Analysts* ou desenvolvedores designados pela gestão da *Smart City*;

7. Administração, Monitorização e Segurança - acompanha o funcionamento de todas as componentes do sistema e garante o adequado desempenho do mesmo, assumindo preocupações com a segurança do sistema e dados nele contido. São ferramentas muitas vezes associadas diretamente à adoção de certas tecnologias, como o Hadoop (Ambari, Knox...), ou então desenvolvidas ou personalizadas, fazendo com que não tenham um espaço de execução claramente definido dentro da camada infraestrutural proposta.

4.4 Intervenientes na Cocriação de Serviços

Esta secção, embora esteja enquadrada no capítulo de apresentação da arquitetura BASIS, não constitui propriamente uma camada da mesma. Em vez disso, visa identificar e descrever os principais intervenientes numa iniciativa de *Big Data* no contexto de *Smart Cities*, de forma a enumerar os modos de cooperação entre os mesmos. Por conseguinte, a Figura 29 ilustra esses mesmos intervenientes e as suas principais atividades, resultantes da sua cooperação.

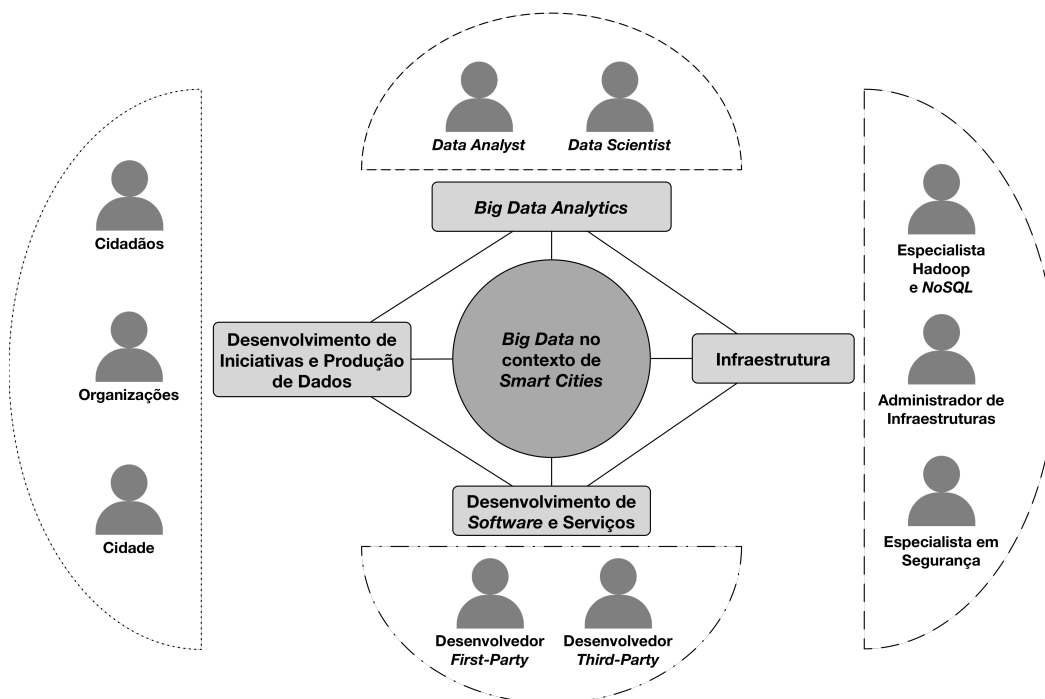


Figura 29. Intervenientes na cocriação de serviços.

Por muito eficiente e inovadora que seja uma solução de recolha, armazenamento, processamento e disponibilização de *Big Data*, a mesma não faz sentido sem o “combustível” que a alimenta, neste caso os dados. De modo a que uma *Smart City* tire proveito da sua solução de *Big Data*, os dados devem ser produzidos e consumidos em considerável volume, variedade e/ou velocidade. É indiscutível o facto de que estas cidades revolucionárias são uma infindável fonte de dados, mas a identificação de casos de aplicação da solução deve ser proveniente dos cidadãos, das organizações e da própria gestão da *Smart City*, tal como identificado na Figura 29. Sem este incentivo para o desenvolvimento de iniciativas e a consequente produção de dados, a solução de *Big Data* estará sempre em subaproveitamento.

A correta administração, monitorização e segurança da infraestrutura fica a cargo de três relevantes intervenientes: o especialista Hadoop e *NoSQL*, que detém conhecimentos na implementação e gestão deste tipo de soluções consideravelmente atuais; o administrador de infraestruturas, como responsável pela gestão do *hardware* propriamente dito, garantindo a correta operacionalização do mesmo; o especialista em segurança, que implementa técnicas que garantem a confiança, privacidade e segurança dos dados e da plataforma, certificando-se que o Hadoop, bases de dados *NoSQL* ou qualquer restante componente da infraestrutura não comprometem a integridade do sistema. Deve-se ter em atenção que todos os intervenientes apresentados não representam um único profissional, podendo as funções de um determinado interveniente ser repartidas por vários colaboradores.

Contudo, nem a suficiente produção de iniciativas, nem a mais adequada infraestrutura, fariam sentido se os dados apenas se limitassem a estar armazenados. O potencial do *Big Data* gerado por uma *Smart City* está sobretudo presente na extração de valor do mesmo, valor esse que pode ser extraído usando o *Big Data Analytics*. Os intervenientes responsáveis por esta tarefa podem ser designados de *Data Scientists* e *Data Analysts*, que usam os seus conhecimentos em *Data Mining* e *Data Visualization* para obterem percepções acerca dos dados, nunca antes descobertas. Estes resultados podem depois ser integrados em *software* e serviços disponibilizados à comunidade.

Finalmente, os intervenientes designados para o desenvolvimento de *software* e serviços que utilizam as percepções provenientes do *Big Data Analytics*, podem ser identificados como desenvolvedores *First-Party* (colaboradores da *Smart City*) ou *Third-Party* (colaboradores de organizações externas ou *freelancers*). Estes intervenientes são responsáveis pela criação de aplicações que estarão disponíveis na camada de serviços de dados e aplicações, não só oficializadas pela *Smart City*, mas também desenvolvidas pela comunidade ou em parceria com outras *Smart Cities*.

A Figura 29 estabelece assim um modo de cooperação entre os vários intervenientes, com vista à disponibilização de serviços inteligentes no contexto de uma *Smart City*, potenciados pelo processamento de *Big Data*. Esta visão colaborativa dos vários intervenientes, permite a identificação das principais responsabilidades e alerta para a importância de cada um deles numa iniciativa deste género.

Para concluir a apresentação da arquitetura BASIS, esta secção visa cruzar as responsabilidades entre as camadas tecnológica e infraestrutural e os vários intervenientes na cocriação de serviços, apresentando as matrizes de responsabilidades dos intervenientes. Nas duas matrizes que são apresentadas de seguida, a responsabilidade é assinalada quando um determinado interveniente desempenha um papel de relativa significância num determinado componente da camada tecnológica ou infraestrutural.

4.4.1 Camada Tecnológica vs. Intervenientes na Cocriação de Serviços

De modo a compreender-se em que componentes da camada tecnológica os vários intervenientes desempenham funções e assumem responsabilidades, observe-se a Tabela 7, que representa o cruzamento entre a camada tecnológica e os intervenientes na cocriação de serviços.

Tabela 7. Cruzamento da camada tecnológica com os intervenientes na cocriação de serviços.

CT\ICS	Cs	O	C	DS/DA	EHN	AI	ES	DFP	DTP
FBD	X	X	X						
EC				X			X	X	
ARM					X	X	X		
BDA				X					
DDMA								X	
SDA	X	X	X					X	X
AMS					X	X	X		

Camada Tecnológica (CT):

Fontes de *Big Data* (FBD); Extração e Carregamento (EC); Armazenamento (ARM); *Big Data Analytics* (BDA); Disponibilização de Dados e Modelos Analíticos (DDMA); Serviços de Dados e Aplicações (SDA); Administração, Monitorização e Segurança (AMS).

Intervenientes na Cocriação de Serviços (ICS):

Cidadãos (Cs); Organizações (O); Cidade (C); *Data Scientist (DS)/Data Analyst (DA)*; Especialista Hadoop e *NoSQL* (EHN); Administrador de Infraestruturas (AI); Especialista em Segurança (ES); Desenvolvedor *First-Party* (DFP); Desenvolvedor *Third-Party* (DTP).

4.4.2 Camada Infraestrutural vs. Intervenientes na Cocriação de Serviços

O mesmo cruzamento pode ser feito entre a camada infraestrutural e os intervenientes na cocriação de serviços. Esse cruzamento é representado na Tabela 8.

Tabela 8. Cruzamento da camada infraestrutural com os intervenientes na cocriação de serviços.

CT\ICS	Cs	O	C	DS/DA	EHN	AI	ES	DFP	DTP
E	X	X	X						X
DPH						X		X	
SW						X	X	X	
SS						X	X	X	
NAP					X	X	X		
SEDMV				X		X	X	X	
DAEMV				X				X	
AMS					X	X	X		

Camada Infraestrutural (CI):

Envolvente (E); Distribuidor de Pedidos *HTTP* (DPH); Servidores *Web* (SW); Servidores de *Streaming* (SS); Nós de Armazenamento e Processamento (NAP); Servidores de *ETL/ELT*, Desidentificação, *Data Mining* e *Data Visualization* (SEDMV); Desenvolvimento de Aplicações, *ETL/ELT*, *Data Mining* e *Data Visualization* (DAEMV); Administração, Monitorização e Segurança (AMS).

Intervenientes na Cocriação de Serviços (ICS):

Cidadãos (Cs); Organizações (O); Cidade (C); *Data Scientist (DS)/Data Analyst (DA)*; Especialista Hadoop e *Nosql* (EHN); Administrador de Infraestruturas (AI); Especialista em Segurança (ES); Desenvolvedor *First-Party* (DFP); Desenvolvedor *Third-Party* (DTP).

Capítulo 5. Portal *Open (Big) Data* da Arquitetura BASIS

Este capítulo dedica-se essencialmente a detalhar o portal *Open (Big) Data*, esclarecendo, numa das secções, as questões de integração com outros componentes, cujo ponto central (*Web services REST*) também se aplica ao serviço *Web* e ao serviço *Mobile*. Esse mesmo detalhe irá consistir na enumeração das principais funcionalidades de cada um dos serviços, ao mesmo tempo que se demonstra a integração com outros componentes da arquitetura.

O conceito de *Open Data* não está diretamente relacionado com *Big Data*, sendo que o primeiro consiste em tornar os dados acessíveis de uma forma aberta (Chignard, 2013), objetivo que não é mencionado na definição relativamente consensual de *Big Data*, segundo a literatura estudada no enquadramento conceptual. De acordo com Chignard (2013), o termo *Open Data* surgiu pela primeira vez em 1995 e tem ganho significativa notoriedade devido à necessidade de transparência organizacional, que hoje é valorizada como medida de combate à desconfiança perante as organizações que atuam em diversas áreas.

Sendo que *Big Data* e *Open Data* são dois conceitos com emergente popularidade, o que aconteceria se eventualmente os dois se cruzassem? A especificação deste portal pretende contribuir para a procura de uma resposta adequada a esta questão. Alguns componentes da arquitetura, tais como o *Big Data Analytics*, quando disponibilizados como um serviço aberto, revelam-se iniciativas essenciais na difusão pública do conhecimento no âmbito de uma *Smart City*, permitindo que vários cidadãos e organizações com experiência na área, ou não, obtenham percepções e padrões nos dados. Contudo, esta iniciativa não é pioneira, sendo que já existem estudos que envolvem a questão de disponibilizar o *Big Data Analytics* como um serviço, focando-se no processo de disponibilização de técnicas de *Data Mining* (Espinosa, Garriga, Zubcoff, & Mazon, 2014; Ye, Wang, Zhou, Wang, & Zhou, 2013), ou, como já se pôde observar nas arquiteturas analisadas, especificando o seu papel numa arquitetura de *Big Data* no contexto de *Smart Cities* (Khan et al., 2013; Vilajosana et al., 2013). No entanto, a especificação de um portal *Open Data*, orientado para a disponibilização pública de *Big Data* e a inclusão de técnicas de análise numa ótica “*as a Service*”, pode revelar-se um contributo de significativa relevância, providenciando uma linha orientadora para futuras iniciativas de *Smart Cities*, sendo que, segundo a pesquisa efetuada, não parecem existir contributos científicos com o mesmo objetivo.

5.1 Funcionalidades

Apesar do portal *Open Data* já ter sido brevemente descrito na apresentação da arquitetura, convém detalhar o leque de funcionalidades que estarão presentes no mesmo, de modo a providenciar, à comunidade, vastos conjuntos de dados de natureza volumosa, variada e veloz, de forma aberta. Além disso, é neste portal que os utilizadores poderão aceder ao *Big Data Analytics* na ótica de um serviço aberto, de modo a poder analisar os conjuntos de dados disponíveis publicamente.

A Figura 30 e Figura 31 ilustram as funcionalidades que serão oferecidas pelo portal *Open (Big) Data* aos vários utilizadores da comunidade. Como se pode observar, a comunidade deve requisitar uma chave de acesso para a incorporação de dados e utilização, em aplicações externas, do *Analytical Processing as a Service (APaaS)*, *Intelligence as a Service (IaaS)* e *Visualization as a Service (VaaS)*. Note-se que apenas a incorporação em aplicações externas requisita uma chave de acesso, pois a exigência da mesma a utilizadores cuja necessidade é a exploração das funcionalidades no próprio portal levantaria entraves na desejada disseminação de inteligência e conhecimento pela comunidade. Assim sendo, o portal assume a responsabilidade de não só disponibilizar à comunidade funcionalidades no próprio portal, mas também garantir que as mesmas são acessíveis por aplicações externas, via *Web services*.

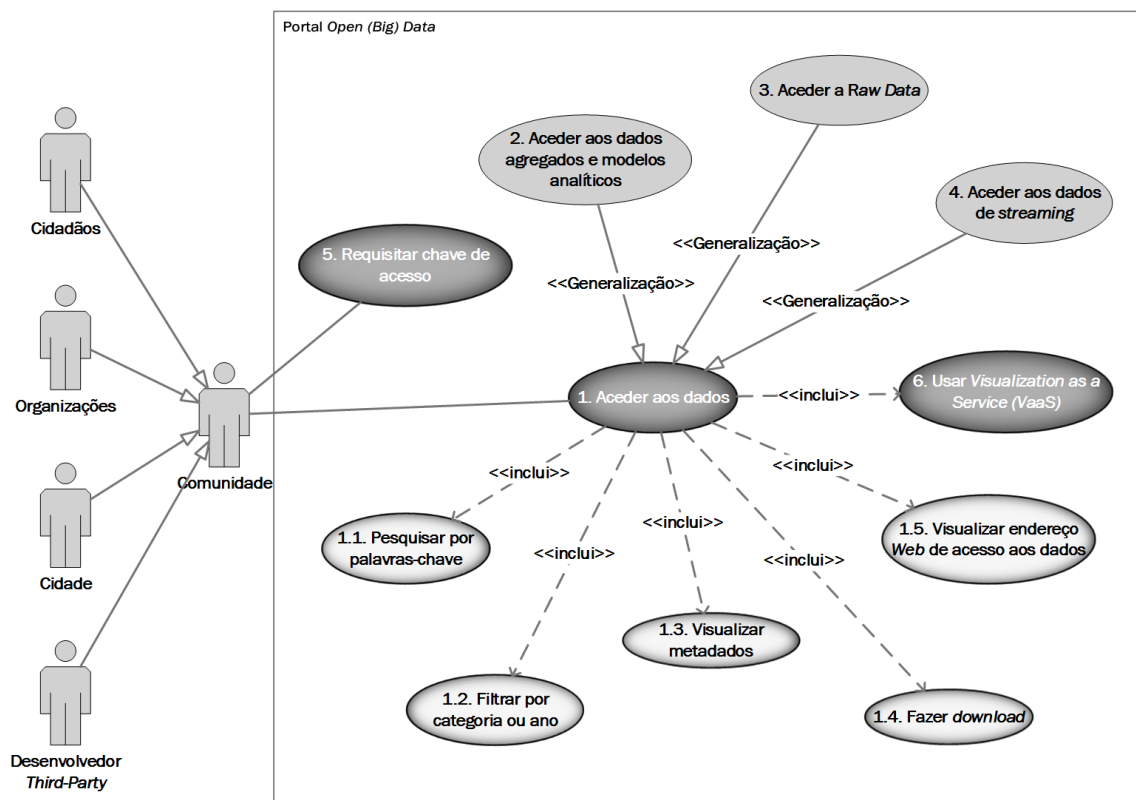


Figura 30. Funcionalidades do portal *Open (Big) Data* para a disponibilização de acesso e de dados.

A pesquisa por conjuntos de dados no portal deve ser intuitiva e coerente, pelo que desde o *raw data* até aos dados de *streaming* deve ser possível pesquisar por palavras-chave, filtrar por categoria ou ano, observar os devidos metadados, identificar os modos de incorporação em aplicações externas ou fazer *download* dos próprios dados, em formatos *JSON*, *CSV*, *TXT* ou *XML*, por exemplo. O *download* de qualquer ficheiro deve ser acompanhado pelos respetivos metadados, de preferência em formato *RDF*. Ainda na Figura 30, pode-se observar a presença do *VaaS* no momento de disponibilização dos dados e não como uma funcionalidade isolada no portal. Esta decisão prende-se ao facto de ser intuitivo para o utilizador analisar visualmente os dados no momento em que está perante o resultado pretendido. Contudo, tal como se observa na Figura 31, a capacidade de visualizar dados também está presente na execução de outras funcionalidades, como o *IaaS* ou o *APaaS*. Assim, em vez de existir uma área isolada para a análise visual de dados, recorre-se à mesma sempre que os resultados provenientes das ações do utilizador possam despoletar resultados visuais.

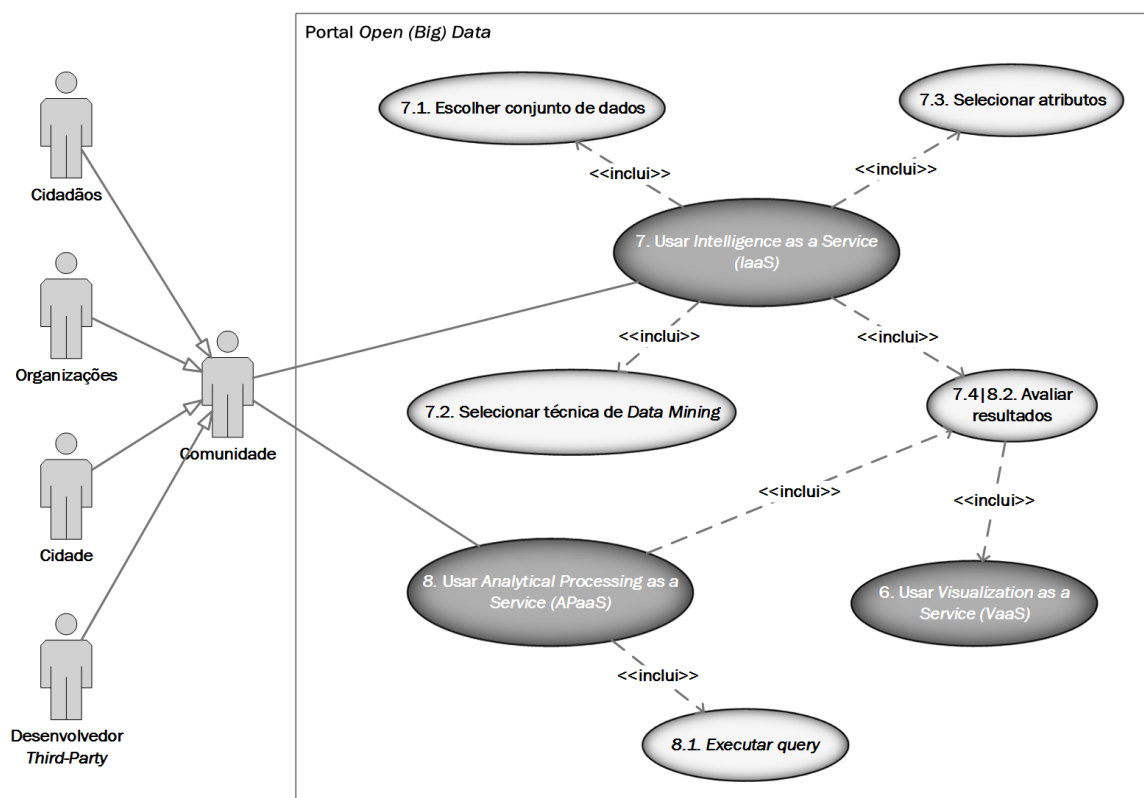


Figura 31. Funcionalidades "as a Service" do portal *Open (Big) Data*.

Até este ponto, esclareceram-se as funcionalidades que poderão ser suportadas por este portal, mas restam 3 incógnitas antes de encerrar esta secção, que passam pela adoção das ferramentas de *Data Visualization*, *Data Mining* e de processamento analítico, de modo a que as mesmas possam ser utilizadas tanto pelos utilizadores do portal como por aplicações externas. Este é um dos principais

requisitos para estas ferramentas, pois a capacidade de integração com outras tecnologias é um fator preponderante nesta arquitetura.

No que diz respeito ao *Data Visualization*, a adoção de uma biblioteca Javascript parece ser a solução mais adequada, de modo a facilitar a sua integração com o portal *Open Data* e serviços *Web* e *Mobile*. Existem algumas soluções, de utilização livre, que se revelam adequadas, das quais se podem destacar o D3.js (www.d3js.org) e o Google Charts (www.developers.google.com/chart). Pode ser ponderada a utilização de ferramentas comerciais, como o Tableau, embora a integração das análises com aplicações *Web* ou *Mobile* seja algo condicionada pela estratégia usada por estas tecnologias, embora seja possível embeber as análises em páginas *Web*. Ainda assim, não parece existir forma de criar as análises através de aplicações externas, pois as mesmas necessitam de ser criadas usando as ferramentas proprietárias, que comportam, normalmente, custos bastante significativos. Por conseguinte, a adoção de uma biblioteca de criação de gráficos permitirá aos utilizadores do portal criar as suas próprias análises, utilizando os dados disponíveis publicamente.

Quanto ao processamento analítico, o Pig ou o Spark poderão ser duas escolhas adequadas, devido à interoperabilidade assegurada com várias fontes de dados do Hadoop e bases de dados *NoSQL*, permitindo várias operações sobre os dados, como por exemplo a limpeza e a agregação dos mesmos. O Pig e o Spark podem ser utilizados em múltiplas linguagens de programação e ficará a cargo do desenvolvedor do portal abstrair a sua complexidade de modo a que o utilizador possa escrever a sua própria *query* para processamento dos dados.

Por fim, para disponibilizar o *Data Mining* poderá ser utilizado novamente o Spark, devido à biblioteca MLlib, que disponibiliza um conjunto considerável de técnicas e algoritmos. Contudo, em casos em que o volume do conjunto de dados não necessite de processamento distribuído poderá ser utilizado o WEKA, por exemplo, cuja utilização externa é bastante facilitada por uma biblioteca em Java significativamente completa, disponibilizando um vasto conjunto de técnicas e algoritmos de *Data Mining*, revelando-se assim ainda mais completo que o Spark neste aspeto. Apesar do WEKA ser desenvolvido em Java e a sua biblioteca oficial ser implementada nessa mesma linguagem, é possível a sua integração com outras linguagens de programação, e como a arquitetura BASIS é *service-oriented*, as funcionalidades da ferramenta podem ser encapsuladas em *Web services*. O WEKA poderá também ser utilizado em ambientes distribuídos delegando ao Hadoop a execução dos algoritmos, funcionalidade disponibilizada em versões mais recentes da ferramenta.

A utilização das ferramentas aqui mencionadas são meras decisões de implementação, sendo que a adoção de uma tecnologia em particular não deverá condicionar os princípios de concepção da arquitetura BASIS. A escolha de uma particular tecnologia não deve criar *bottlenecks* no desempenho do sistema, nem condicionar a facilidade de integração com qualquer plataforma que pretenda aceder aos dados ou funcionalidades.

5.2 Integração com Outros Componentes e Aplicações Externas

O portal *Open Data* encontra-se inserido na componente de Serviços de Dados e Aplicações da arquitetura BASIS e, consequentemente, os seus processos de entrada e saída de dados são estabelecidos por *Web services*. Com vista a entender-se como se processa o fluxo de dados entre o portal e os outros componentes e aplicações externas observe-se a Figura 32.

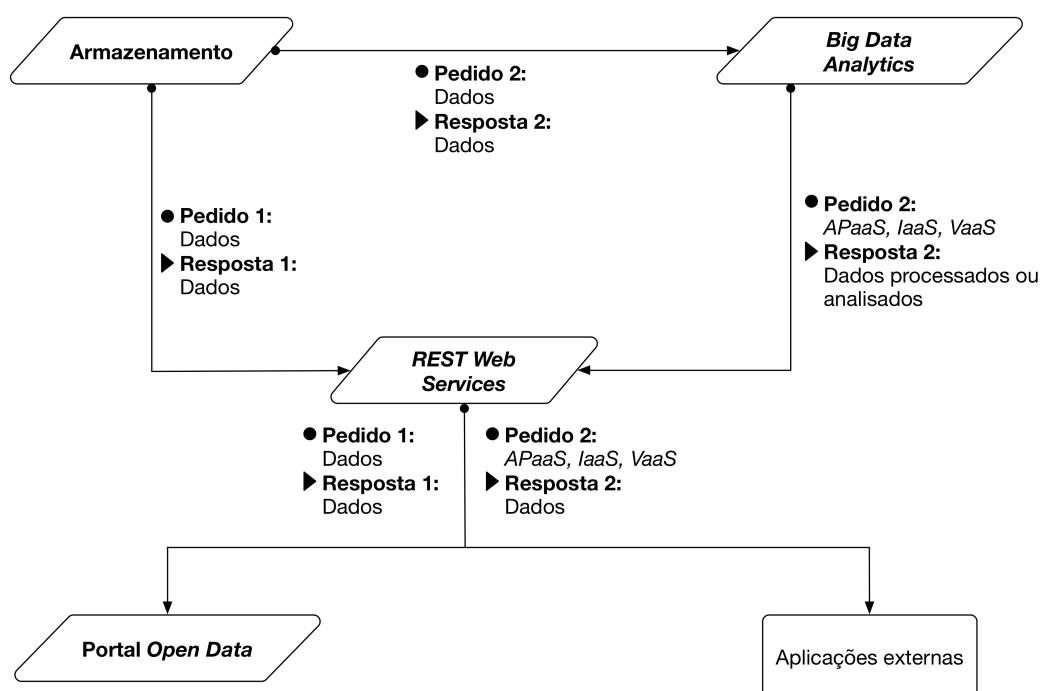


Figura 32. Fluxo de dados entre o portal *Open (Big) Data* e outros componentes ou aplicações externas.

Os *Web services* são o ponto central de distribuição de dados, garantindo que não existe dependência com a plataforma que acede aos mesmos, sendo esta o portal ou qualquer outra aplicação externa. Segundo o que se pode observar na Figura 32, tanto o portal *Open Data* como as aplicações externas podem utilizar os mesmos recursos *REST*, sendo que o portal pode ser consultado pelos desenvolvedores para visualizar o endereço *HTTP* associado ao *Web service* pretendido. Conclui-se, assim, tal como mencionado em diversas ocasiões, que a arquitetura está concebida de forma a que até os serviços desenvolvidos pela própria cidade sejam tratados como aplicações externas, na medida em

que apenas consomem *Web services*. Tal fator garante facilidade na alteração das tecnologias adotadas por outros componentes sem alterar o código associado ao desenvolvimento dos serviços e aplicações.

5.3 Protótipo de Interface Gráfica

Embora o desenvolvimento do portal *Open Data* não se enquadre no âmbito desta dissertação, a especificação do mesmo é relevante para a compreensão do potencial do mesmo e facilitação do aparecimento de iniciativas futuras. Consequentemente, optou-se pelo desenvolvimento de um protótipo de interface gráfica, onde é ilustrada a disposição das várias funcionalidades acima mencionadas.

A Figura 33 apresenta a página inicial do portal, onde os utilizadores são transportados diretamente para a descrição dos vários tipos de disponibilização de dados. Nesta página os utilizadores podem também aceder às funcionalidades “*as a Service*” e ao serviço *Web* e *Mobile* desenvolvidos pela cidade. Além disso, informações gerais como o objetivo do portal e contactos são também fornecidos.



Figura 33. Protótipo do portal *Open (Big) Data* - página inicial.

À semelhança das funcionalidades que foram definidas, é possível pesquisar os conjuntos de dados disponíveis por determinadas palavras-chave, categorias ou ano de publicação, de forma a encontrar o *dataset* apropriado ao caso de utilização pretendido pelo utilizador. A funcionalidade de pesquisa por conjuntos de dados é ilustrada na Figura 34. Esta funcionalidade é transversal à disponibilização de *raw data*, dados agregados, modelos analíticos e dados de *streaming*, sendo que o modo de visualização e pesquisa no portal é idêntico. Destaca-se apenas o facto de na disponibilização de dados de *streaming* dever ser indicado ao utilizador como usar a *API* e estabelecer a conexão,

salientando limitações da mesma, caso existam (exemplo: número de conexões por minuto, tempo da conexão, entre outros).

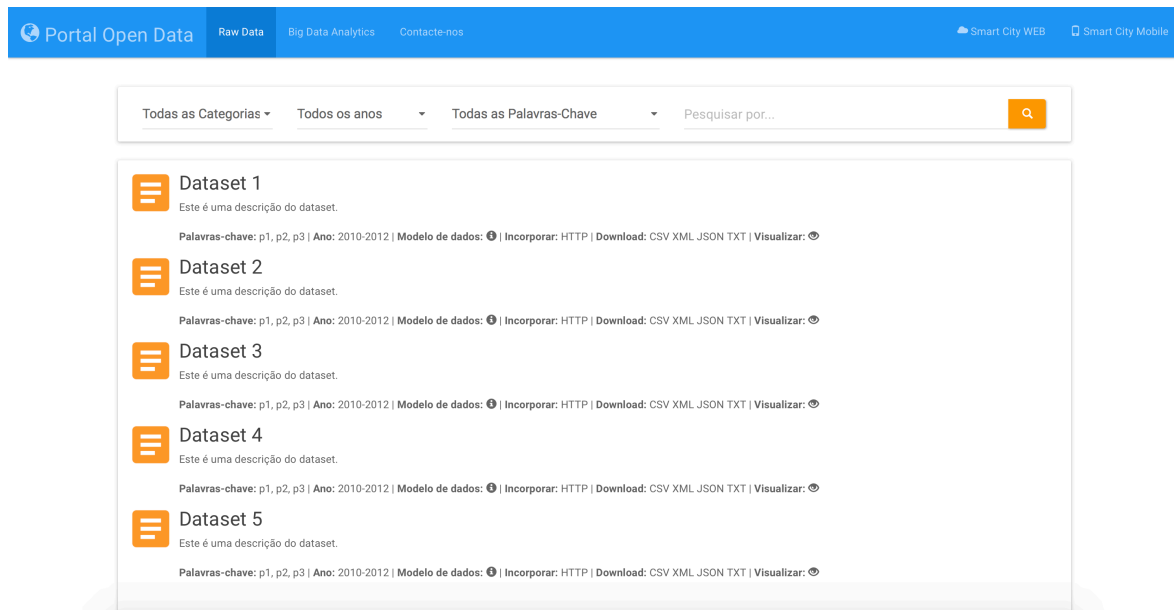


Figura 34. Protótipo do portal *Open (Big) Data* - página de disponibilização de conjuntos de dados.

De modo a finalizar a apresentação do protótipo desenvolvido, é apresentada, na Figura 35, a página usada para aceder às funcionalidades de *Data Mining*, permitindo que os utilizadores do portal usem qualquer conjunto de dados disponibilizado publicamente, à exceção dos dados de *streaming*, devido à natureza dos mesmos.

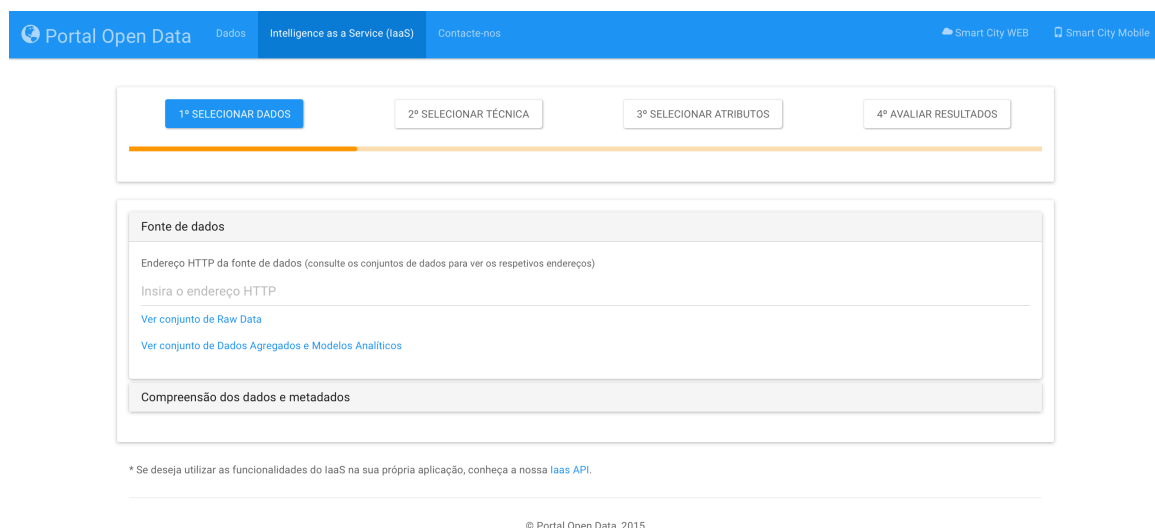


Figura 35. Protótipo do portal *Open (Big) Data* - página do *IaaS*.

No entanto, os dados de *streaming* acabam por ser disponibilizados publicamente como *raw data*, pois os mesmos após armazenados perdem a denotação de fluxo em tempo real, e passam a ser também considerados como dados de carácter histórico.

A interface gráfica para o *VaaS* e o *APaaS* não é apresentada neste documento, pois entende-se que a anterior especificação das funcionalidades e a forma como se enquadram no portal são claras o suficiente para o desenvolvimento de futuras iniciativas. Conclui-se, assim, a especificação do portal *Open (Big) Data*, cujo desenvolvimento não se enquadra no âmbito desta dissertação.

Capítulo 6. Validação da Arquitetura BASIS: Serviços Inteligentes em *Smart Cities*

Este capítulo consolida a validação da arquitetura BASIS, utilizando um diverso conjunto de tecnologias adequadas ao desenvolvimento de dois serviços inteligentes no contexto de *Smart Cities*. Estes serviços representam um ponto crucial nesta investigação, na medida em que permitem retirar conclusões acerca da especificação de vários componentes da arquitetura e das tecnologias de *Big Data* propostas, não só a nível qualitativo, em termos de contributo para o quotidiano dos cidadãos, mas também ao nível quantitativo, no que diz respeito ao desempenho das tecnologias quando lidamos com *Big Data*.

6.1 Serviço para o Consumo Inteligente de Energia

Atualmente, as cidades consomem cada vez mais energia para sustentar as suas atividades do quotidiano. Com a crescente utilização de dispositivos elétricos enfrentamos desafios associados à distribuição e controlo de energia. Além disso, tendemos a despendar bastante energia para aquecer ou arrefecer as nossas habitações.

Este trabalho instancia certos componentes da camada tecnológica da arquitetura BASIS, referidos posteriormente, com o objetivo de garantir a extração, carregamento, transformação, *clustering* e *forecasting* de *Big Data* associado a consumos de energia, cujo resultado encontra-se já publicado cientificamente em (Costa & Santos, 2015).

O serviço aqui proposto faz uso de um conjunto de dados que contém o consumo de eletricidade e de gás em casas distribuídas por múltiplas cidades dos Estados Unidos da América (Commercial and Residential Reference Building Models, 2013). Este conjunto de dados é utilizado devido à disponibilização do mesmo de uma forma aberta. O objetivo principal deste serviço consiste em disponibilizar, aos cidadãos, uma nova forma de monitorizarem o seu consumo de eletricidade e gás, permitindo que comparem o consumo da sua habitação com outras pertencentes ao mesmo *cluster* ou estado/cidade, tendo também a possibilidade de prever consumos de eletricidade futuros. Para além disso, o serviço também providencia, à gestão das cidades e aos fornecedores de energia, uma forma mais inteligente de monitorizarem o panorama do consumo de energia.

O serviço faz uso de dados simulados, provenientes dos Estados Unidos da América, em conjunto com tecnologias como o Hadoop, Talend Open Studio for Big Data, WEKA e Tableau, para armazenar e processar *Big Data*, produzir *clusters* e *time series forecasts* e visualizar informação, respetivamente. Os resultados alcançados mostram que usando esta abordagem é possível identificar *clusters* de habitações a partir do seu consumo energético. É também possível prever futuros consumos de eletricidade com uma reduzida margem de erro.

As principais vantagens do serviço proposto são o facto de qualquer cidadão poder monitorizar periodicamente o seu consumo de energia, comparar-se com outras casas do mesmo *cluster* ou cidade e prever quanta energia irá gastar se mantiver o seu padrão de consumo. Isto conduz os cidadãos a um processo de tomada de decisão mais rico, informando-os sobre possíveis divergências em relação a consumos de energia usuais, tornando-os mais conscientes em termos ambientais, ou pelo menos, conscientes em termos económicos. Os governos das cidades e os fornecedores de energia podem também adotar o ideal proposto neste serviço, com vista a gerir de forma mais eficiente o consumo energético das cidades, usar os dados processados em iniciativas de marketing ambiental e prever necessidades energéticas.

6.1.1 Análise e Previsão de Consumos Energéticos

Entre a comunidade científica já se encontram disponíveis algumas abordagens de previsão de consumos energéticos. Alguns dos trabalhos estão sobretudo relacionados com o preço da energia (Azevedo & Vale, 2006; Xie, Zheng, & Zhang, 2007; Zhou, Wu, & Li, 2011), enquanto que outros endereçam a investigação em torno do consumo energético (Alzate & Sinn, 2013; Gu, Cheng, & Wang, 2011). A prática comum nestes trabalhos relacionados parece ser o *Data Mining* para a identificação de *clusters* antes da aplicação de modelos de *forecasting*. Nesta matéria, o trabalho de Alzate & Sinn (2013) apresenta uma melhoria de 20% na precisão das previsões, usando *clustering* antes de aplicar um modelo de *forecasting*.

Independentemente das técnicas de *clustering* usadas, bem como as técnicas de *forecasting*, parece haver, geralmente, uma abordagem comum: usar *clustering* com vista a melhorar a eficiência do modelo de *forecasting*, quer através da adaptação do modelo para cada *cluster*, quer através do uso de *clustering* como técnica de extração de novos atributos no conjunto de dados. Estes trabalhos relacionados focam os seus resultados na melhoria da eficiência do *Data Mining*, recorrendo a técnicas inovadoras, e, em geral, os resultados mostram uma conclusão satisfatória. Contudo, estes são

primariamente focados no processo de *Data Mining* e nos resultados, descartando não só a natureza dos dados do mundo real, que requerem novas tecnologias de armazenamento e processamento, mas também a importância da possível implementação tecnológica, com vista à disponibilização de novos serviços ao cidadão.

Outros trabalhos relacionados descrevem os dados dos *Smart Meters* como sendo *Big Data*, apresentando alguns métodos para a visualização da informação e extração de conhecimento (Alahakoon & Yu, 2013; Simmhan et al., 2013). Além disso, existem alguns trabalhos que estão a ser desenvolvidos com o objetivo de estudar a importância da infraestrutura de armazenamento e processamento (Arenas-Martínez et al., 2010; Mayilvaganan & Sabitha, 2013; Simmhan et al., 2013), destacando as bases de dados *NoSQL* e o Hadoop.

No serviço proposto nesta dissertação pretende-se demonstrar como podemos processar os dados de energia, através da arquitetura de *Big Data* proposta, usando técnicas de *clustering* e *time series forecasting*, não só para selecionar os modelos de *forecasting* adequados para cada *cluster*, mas também com vista a enriquecer a análise visual e criar um serviço inteligente, disponibilizando aos cidadãos uma fatura de energia reinventada e providenciando aos governos e fornecedores de energia uma nova forma de monitorizarem e segmentarem o consumo energético. Consequentemente, os resultados apresentados são focados não só no sucesso do *Data Mining*, mas também na forma de mudar a experiência de consumidor e fornecedor de energia, reinventando formas de apresentação do consumo energético. Alcançar um erro reduzido nas técnicas de *clustering* e *forecasting* é também um objetivo crucial neste serviço. A abordagem seguida faz uso de tecnologias de *Big Data* de modo a validar a possível implementação num cenário de aplicação real e apresenta os resultados finais numa análise visual cuidada, de modo a tentar ultrapassar a barreira entre uma aplicação de *Data Mining* bem sucedida e um serviço de uma *Smart City*.

6.1.2 Visão Geral da Camada Tecnológica e dos Dados

Todos os passos e componentes tecnológicos utilizados neste serviço são ilustrados na Figura 36, que representa uma instanciação da camada tecnológica da arquitetura BASIS, de modo a compreender o que pode ser um ponto de partida para futuras implementações de outros serviços.

Neste trabalho, faz-se uso de múltiplos componentes do Hadoop, tais como: HDFS (www.hadoop.apache.org) para armazenar os ficheiros não estruturados; Pig (www.pig.apache.org) para processar *scripts* de agregação de dados; HBase (www.hbase.apache.org) para armazenar

temporariamente os resultados do Pig; Hive (www.hive.apache.org) para atuar como um *Data Warehouse*, contendo o conjunto de dados final, de modo a originar a análise visual. O Talend Open Studio for Big Data (www.talend.com/products/talend-open-studio) é responsável por todos os fluxos de dados, direcionando os mesmos do HDFS e HBase para o sistema de ficheiros local e vice-versa. Posteriormente, o WEKA (www.cs.waikato.ac.nz/ml/weka) é usado para a construção de modelos de *clustering* e *forecasting*. O Talend Open Studio for Big Data usa a biblioteca Java do WEKA, de modo a integrar os modelos em tempo real e armazenar os resultados no Hive. Finalmente, foi desenvolvida uma análise visual de dados usando o Tableau (www.tableau.com).

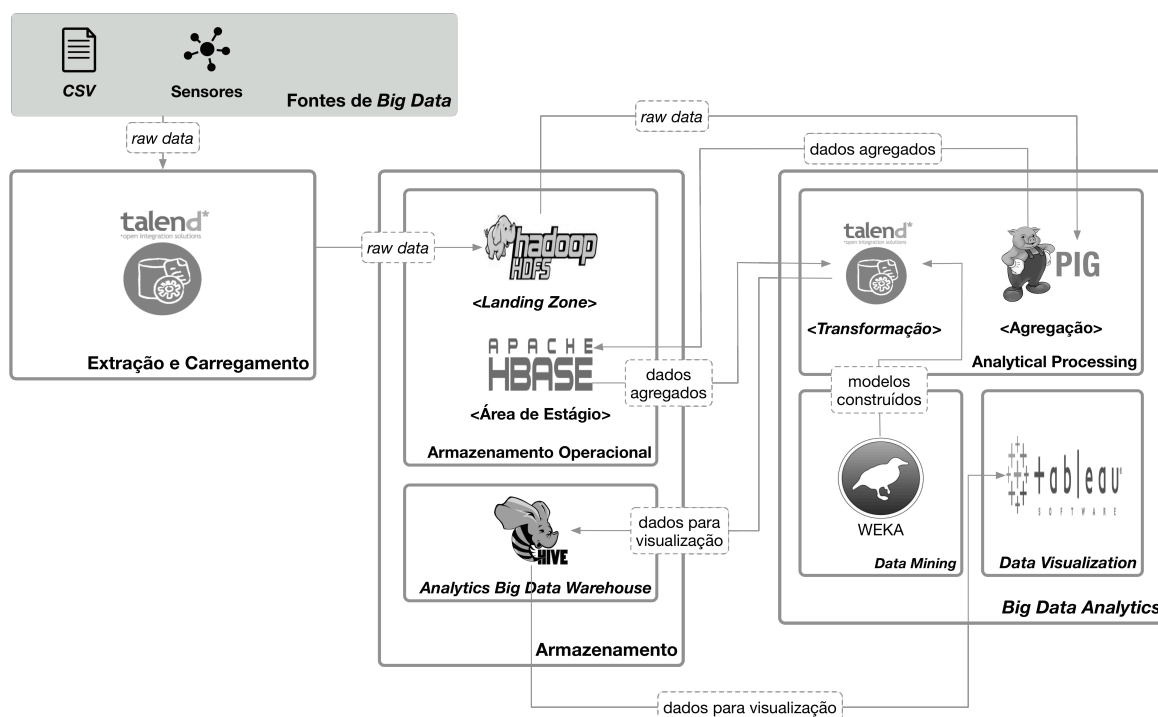


Figura 36. Camada tecnológica utilizada no serviço para o consumo inteligente de energia. Adaptado de (Costa & Santos, 2015).

De modo a perceber o conjunto de dados utilizado neste serviço, convém apresentar o modelo de dados, além da forma como todos os 237 ficheiros foram verificados, extraídos e armazenados no Hadoop. Cada ficheiro neste conjunto contém dados do consumo horário de eletricidade e gás, ao longo de 1 ano, numa casa simulada com características comuns, como 3 quartos e 1 ou 2 casas de banho, tendo em consideração o ambiente e clima em que a casa está inserida. Existe um ficheiro por cada cidade dos Estados Unidos da América, representando o consumo de uma casa enquadrada na respetiva cidade. A Figura 37 ajuda na compreensão do modelo e conteúdo dos ficheiros, além das classes de atributos. *HVAC* significa “aquecimento, ventilação e ar condicionado”, *APPL* significa eletrodomésticos e *MISC* significa variados.

Como pode ser visto na Figura 37, a informação do estado e cidade encontra-se no nome do ficheiro, o que requiere um esforço de processamento adicional para extraí-los e armazená-los dentro do ficheiro, com vista a uma análise futura.

Lidar com *Big Data* implica a verificação de certas características que os dados possam ter: volume, variedade, velocidade, veracidade e valor (Chandarana & Vijayalakshmi, 2014). No caso deste serviço, e tendo em consideração que é uma prova de conceito a ser executada em apenas uma máquina, os dados apresentam um volume considerável (237 ficheiros, como mencionado, contendo mais de 8760 linhas cada um, totalizando mais de 550 *megabytes*). Num cenário real, tendo em conta que numa cidade existem inúmeras casas, o conjunto de dados a ser analisado será de um volume bastante superior. Além disso, estes dados virão de vários sensores e serão atualizados numa base horária. Este conjunto de dados foi usado com vista a testar a arquitetura e o serviço inteligente proposto, porque traz consigo a veracidade do processo de simulação (Commercial and Residential Reference Building Models, 2013) e a informação valiosa que pode ser extraída usando *Data Mining* e *Data Visualization*.

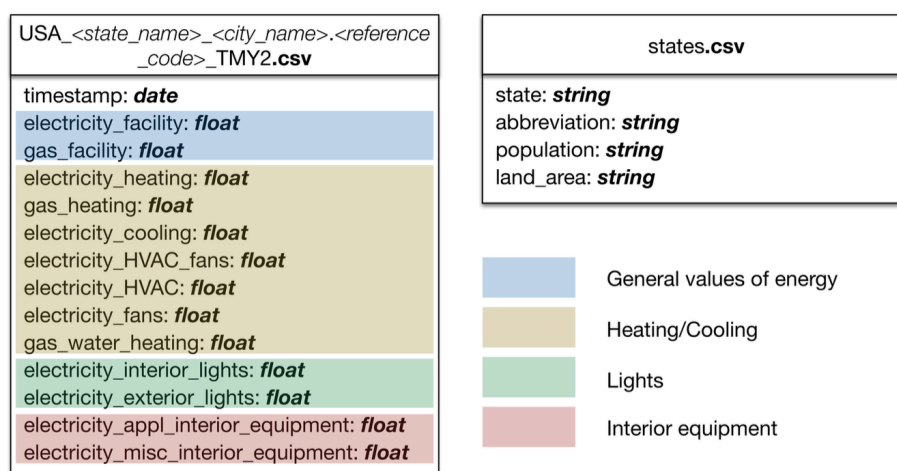


Figura 37. Modelo de dados do conjunto de dados original. Adaptado de (Costa & Santos, 2015).

6.1.3 Preparação dos Dados e *Data Mining*

A análise da qualidade dos dados em contexto de *Big Data* é uma tarefa desafiante, devido sobretudo ao seu volume (Liu et al., 2013). Neste caso de demonstração, todos os 237 ficheiros foram armazenados no Hadoop, e foi verificada a qualidade dos dados usando o Talend Open Studio for Data Quality (www.talend.com/products/talend-open-studio). Os dados não apresentaram falhas a considerar em futuras fases de transformação, muito devido ao facto de serem simulados e não influenciados por problemas de inserção manual. Numa aplicação em contexto real é também expectável que os dados

não apresentem falhas significativas, devido a serem extraídos de sensores, usando métodos autónomos. Neste caso, alguns ficheiros apresentaram o valor zero no consumo de gás. Posteriormente, é explicado como este facto afetará o processo de desenvolvimento.

A tarefa de *Data Mining* integrou um exercício de *clustering*, de modo a segmentar as casas pelo seu consumo de eletricidade e gás, e um exercício de *time series forecasting* para prever futuros consumos de eletricidade. Combinando estas duas técnicas, foi disponibilizado um serviço que permite não só comparar o consumo histórico e previsto de uma casa com outras casas no mesmo *cluster*, mas também comparar consumos entre *clusters*.

6.1.3.1 Clustering

Com vista a identificar *clusters* a partir do conjunto de dados original, era necessário um conjunto de dados menos detalhado, nomeadamente, dados agrupados por estado e cidade. De modo a obter esse resultado, foi realizada uma operação de agregação em Pig, uma linguagem de alto nível incorporada no Hadoop e designada para realizar análise de dados. A *script* Pig é demonstrada na Figura 38. Os resultados consistem no consumo de energia anual por estado e cidade e foram armazenados no HBase, que atuou como a área de estágio dos dados.

```

1 loads = LOAD 'base_load' USING PigStorage(';') AS (timestamp:chararray,...);
2
3 usa_states = LOAD 'usa_states/states.csv' USING PigStorage(';') AS(state_name:chararr
4
5 loads_join = JOIN loads by state LEFT OUTER, usa_states BY state_abbr;
6
7 grouped_loads = GROUP loads_join BY (state_name, city);
8
9 average_loads = FOREACH grouped_loads GENERATE group as row_key, group.state_name AS
10 group.city AS city,
11 SUM(loads_join.electricity_facility) AS electricity_sum,...;
12
13 STORE average_loads INTO 'hbase://loads_by_state_and_city'
14 USING org.apache.pig.backend.hadoop.hbase.HBaseStorage('geo:state geo:city electricit

```

Figura 38. Script Pig de preparação dos dados para *clustering*. Retirado de (Costa & Santos, 2015).

Este conjunto de dados resultante foi posteriormente transformado, usando o Talend Open Studio for Big Data, tal como é apresentado na Figura 39. Neste processo de transformação, começou-se por ler os dados armazenados previamente no HBase e, de forma a construir novos dados, os atributos mais detalhados de consumo de energia foram agregados em três categorias: aquecimento ou arrefecimento, luzes e equipamento de interior. Todos os outros atributos permaneceram inalterados. Seguidamente, os valores com casas decimais foram arredondados para o inteiro mais próximo, devido a

propósitos de apresentação visual. Usando o último componente, guarda-se o ficheiro para ser usado no processo *clustering*.

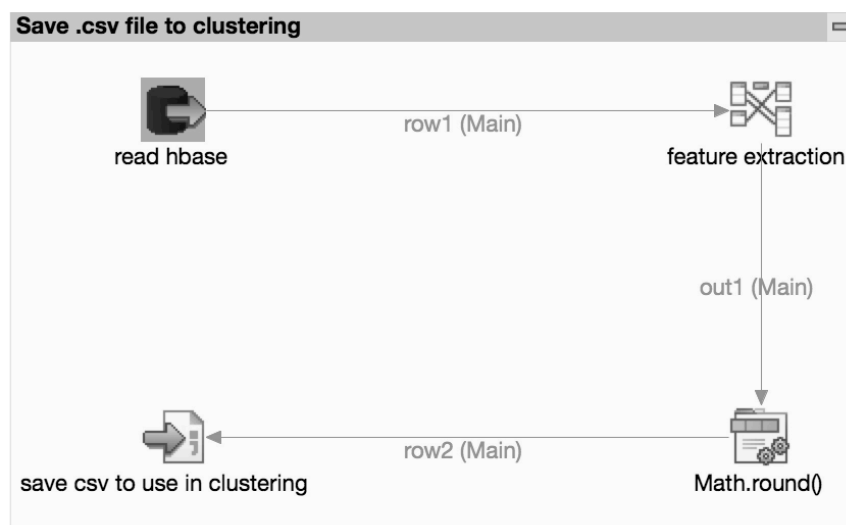


Figura 39. Transformação dos dados para propósitos de *clustering*.

Contudo, após analisar a dispersão e correlação de todos os atributos, optou-se pela remoção destes atributos detalhados do processo de *clustering*, pois detinham baixa dispersão e alta correlação com os atributos gerais de eletricidade e gás. Atributos altamente correlacionados tendem a influenciar algumas técnicas de *clustering* (Mooi & Sarstedt, 2011).

Neste serviço, o processo de *clustering* é conduzido usando o algoritmo K-means, que requiere a especificação de um parâmetro de entrada k , isto é, o número de *clusters*. Uma vez que não existe indicação do número adequado de *clusters* para este *dataset*, todos os dados foram iterativamente usados para produzir *clusters*, incrementando o k de modo a registar cada valor de similaridade *intra-cluster*. O K-means é um algoritmo de *clustering* bastante reconhecido, que particiona um conjunto de dados em k grupos, selecionando o centro dos *clusters* e refinando-os iterativamente (Wagstaff, Cardie, Rogers, & Schroedl, 2001).

Este foi o único algoritmo escolhido devido à simplicidade em avaliar os resultados, usando a similaridade *intra-cluster*. A Figura 40 mostra a similaridade *intra-cluster* para cada k (número de *clusters* utilizado). Usando o método L, em que o “joelho é encontrado num gráfico com o número de *clusters* vs. métrica de avaliação de *clustering*” (Salvador & Chan, 2004), foi possível identificar que 5 aparenta ser o número adequado de clusters para este conjunto de dados.

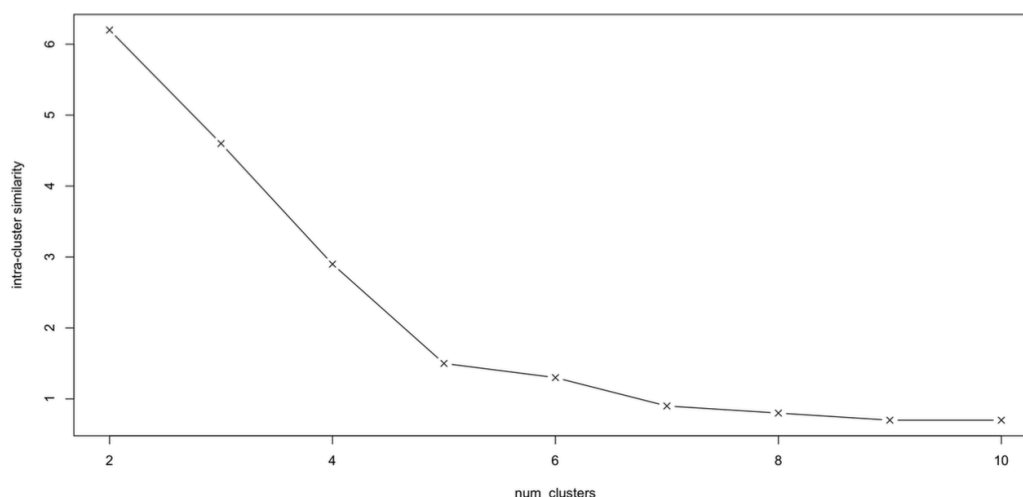


Figura 40. Similaridade *intra-cluster* para cada número de *clusters* (valores mais baixos significam melhor similaridade *intra-cluster*). Retirado de (Costa & Santos, 2015).

A síntese do modelo construído é mostrada na Figura 41, ilustrando os centróides (em *kw/h*) dos *clusters* identificados e o número de casas em cada um deles (neste conjunto de dados, tal como anteriormente explicado, apenas existe uma casa por cidade). O modelo de *clustering* é construído usando o consumo de eletricidade e gás, agrupado por estado e cidade, como previamente descrito.

Final cluster centroids:

Attribute	Full Data (237.0)	Cluster# 0 (12.0)	1 (30.0)	2 (52.0)	3 (114.0)	4 (29.0)
electricity_sum	10584.2658	9807	14734.6	12653.5962	9116.2281	8672.8276
gas_sum	29001.827	73002.25	0	15134.3846	41387.5351	16973.7586

Figura 41. Síntese do modelo de *clustering* (valores de eletricidade e gás encontram-se em *kw/h*; o número de cidades em cada *cluster* encontra-se sob o número do *cluster*). Retirado de (Costa & Santos, 2015).

6.1.3.2 Time Series Forecasting

No processo de *time series forecasting*, os resultados do *clustering* são usados como *input* para um ficheiro contendo o número do *cluster* a que cada casa pertence. Este ficheiro foi integrado com o conjunto de dados original (Figura 37), usando outra *script* Pig, também responsável por agrupar o resultado por *cluster* e *timestamp* (dia e hora), disponibilizando o consumo horário de cada *cluster* ao longo de todo o ano.

No que concerne à granularidade temporal, como o conjunto de dados continha apenas um ano de consumo energético, não permitindo qualquer análise sazonal, a granularidade temporal selecionada foi “por semana”, permitindo prever as próximas semanas de consumo, oferecendo elevado valor de monitorização e planeamento.

Tomadas estas decisões, o conjunto de dados a usar no processo de *forecasting* inclui assim os seguintes atributos:

1. Número de *cluster*;
2. Semana;
3. Consumo de eletricidade;
4. Consumo de gás.

Antes de iniciar o processo de *forecasting*, como a primeira e última semanas do conjunto de dados não ofereciam todos os 7 dias, as mesmas não foram consideradas, melhorando assim a variância da série temporal. Além disso, devido a estarmos agora a lidar com dados de natureza temporal, deve-se lembrar que alguns valores do consumo de gás são zero, fazendo com que um dos *clusters* obtivesse zero como centróide. Além desta questão, observou-se que outro *cluster* apresentava sérios declives no consumo de gás. Devido a estes dois factores, este serviço apenas efetuará previsões de consumo de eletricidade.

Testar *time series forecasting*, usando o WEKA, é muito semelhante a testar outras técnicas de *Data Mining*. De modo a avaliar os modelos, foi usada uma abordagem *holdout*, reservando 80% dos dados para treino e os restantes 20% para propósitos de teste. Foram utilizadas 3 métricas diferentes, descritas, respetivamente, na equação 1, 2 e 3: *Mean Absolute Error (MAE)*; *Root Mean Squared Error (RMSE)*; *Direction Accuracy* (x representa o valor real, $\hat{x}$ o valor previsto e n o número de previsões).

$$MAE = \frac{1}{n} \sum_{i=1}^n (|\hat{x} - x|) \quad (1)$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (\hat{x} - x)^2}{n}} \quad (2)$$

$$Direction Accuracy = \frac{count(\sin(x_{current} - x_{previous}))}{n} = \frac{\sin(\hat{x}_{current} - \hat{x}_{previous})}{n} \quad (3)$$

Quatro modelos foram construídos e avaliados, para cada um dos cinco *clusters*. Estes modelos observam principalmente as 12 últimas semanas de modo a prever as 8 seguintes. A Figura 42 demonstra os resultados obtidos para cada modelo e *cluster*, utilizando os algoritmos WEKA de *Linear Regression*, *Neural Network*, *Support Vector Machines* e *Decision Tree*.

electricity forecasting evaluation						
Model	Measure	cluster1	cluster2	cluster3	cluster4	cluster5
Linear Regression	Mean absolute error	29,8	46,8	22,9	27,1	23,2
	Root mean squared error	31,2	53,8	27,7	28,2	25,0
	Direction accuracy	55,7	39,7	55,5	77,1	59,0
Neural Network (Multilayer Perceptron)	Mean absolute error	9,7	111,2	20,9	14,1	12,3
	Root mean squared error	12,3	127,4	28,9	18,2	14,2
	Direction accuracy	54,3	29,8	49,2	22,3	51,7
Support Vector Machines (SMOreg)	Mean absolute error	89,0	172,7	16,8	18,1	12,4
	Root mean squared error	92,3	189,9	25,3	22,3	14,4
	Direction accuracy	49,5	36,5	68,4	77,1	62,6
Decision tree (M5P Tree)	Mean absolute error	37,0	45,4	17,2	9,2	6,7
	Root mean squared error	38,1	55,6	20,7	12,5	6,9
	Direction accuracy	54,3	59,3	68,9	77,1	58,0

Figura 42. Avaliação do *forecasting* de eletricidade. Adaptado de (Costa & Santos, 2015).

A escolha destes algoritmos deve-se à necessidade de testar diversos *forecasters*, de forma a verificar os que melhor se adaptam aos dados e que portanto garantem melhores resultados no processo preditivo.

Os resultados obtidos indicam que os algoritmos selecionados identificam modelos que apresentam bons resultados, tendo em consideração que os erros são medidos em *kilowatts/hora* (*kw/h*). Isto significa que para cada *cluster*, existe pelo menos um modelo que consegue prever o consumo elétrico para as próximas 8 semanas com um *MAE* inferior ou igual a 16.8 *kw/h*, excepto para o *cluster 2*, cujo menor *MAE* observado é cerca de 46 *kw/h*. Além disso, existem modelos capazes de prever com um *MAE* de 6.7 *kw/h* (Figura 43). Tendo em consideração que os valores médios de consumo semanal variam entre 120 e 412 *kw/h*, estes testes revelam resultados satisfatórios.

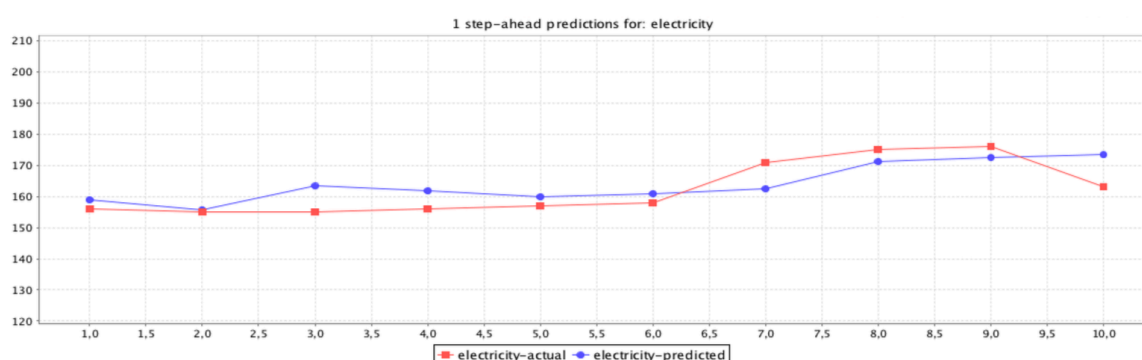


Figura 43. Demonstração dos erros do modelo M5P Tree, selecionado para o *cluster 5*, cujas previsões permitem obter um *MAE* significativamente reduzido. Adaptado de (Costa & Santos, 2015).

O conjunto de dados foi testado com todos os 4 modelos, com o objetivo de encontrar o melhor *forecaster* para cada *cluster* e as suas correspondentes casas. O modelo subjacente aos *clusters 2, 3, 4 e*

5 é a *Decision Tree*, enquanto que para o *cluster* 1 é a *Neural Network*, como pode ser visto na Figura 42. Uma demonstração do modelo construído para o *cluster* 5 é apresentada na Figura 44.

```

electricity:
M5 pruned model tree:
(using smoothed linear models)

week*Lag_electricity-8 <= 3372 :
| week*Lag_electricity-12 <= 3404.5 : LM1 (8/7.45%)
| week*Lag_electricity-12 > 3404.5 : LM2 (6/6.259%)
week*Lag_electricity-8 > 3372 :
| Lag_electricity-12 <= 149.5 : LM3 (8/22.257%)
| Lag_electricity-12 > 149.5 :
| | Lag_electricity-5 <= 190 : LM4 (15/20.651%)
| | Lag_electricity-5 > 190 : LM5 (4/3.014%)

LM num: 1
electricity =
0.1235 * week
+ 0.3486 * Lag_electricity-1
- 0.2334 * Lag_electricity-12
+ 0.0014 * week*Lag_electricity-8
+ 0.0016 * week*Lag_electricity-12
+ 124.3329

```

Figura 44. Modelo M5P Tree, selecionado para o *cluster* 5. Adaptado de (Costa & Santos, 2015).

Na abordagem proposta, cada casa herdará o *forecaster* do seu *cluster*. Por exemplo, se uma casa de New York pertence ao *cluster* 3, o *forecaster* que será usado para prever o consumo de eletricidade será a *Decision Tree*, isto é, o modelo será novamente treinado para o consumo da respetiva casa, mas está pré-estabelecido que o *forecaster* usado para esse fim é a *Decision Tree* do *cluster* 3.

A próxima subsecção apresenta a análise e visualização dos modelos de previsão num contexto de uma *Smart City*, salientando a relevância do serviço de consumo inteligente de energia proposto nesta dissertação, quer pelo seu papel de prova de conceito para a validação da arquitetura proposta, quer pela sua relevância no contexto das *Smart Cities*.

6.1.4 Análise e Visualização de Dados: Reinventando a Fatura de Energia para um Consumo mais Inteligente

Esta subsecção demonstra como os modelos de *time series forecasting* de cada *cluster* são usados para prever consumos das casas pertencentes a um dado *cluster*. Isto é alcançado através da integração da biblioteca Java do WEKA com o Talend Open Studio for Big Data, permitindo que cada casa contenha não só dados históricos, mas também dados sobre previsões de consumo energético.

Em termos de infraestrutura, todos os dados históricos, segmentados e previstos encontravam-se armazenados no HDFS, sendo necessário devolver alguma estrutura aos mesmos, formando um conjunto de dados adequado para análise e visualização. Desta forma, os dados foram reestruturados, passando a integrar:

1. *State_abbreviation* - abreviação do estado a que a casa pertence;

2. *State_name* - nome do estado a que a casa pertence;
3. *City* - nome da cidade a que a casa pertence;
4. *Cluster* - número do *cluster* a que a casa pertence;
5. *Model* - modelo usado para prever os consumos de eletricidade;
6. *Electricity* - consumo de eletricidade da casa na respetiva semana;
7. *Electricity_cluster_avg* - consumo médio de eletricidade do *cluster* na respetiva semana;
8. *Gas* - consumo de gás da casa na respetiva semana;
9. *Gas_cluster_avg* - consumo médio de gás do *cluster* na respetiva semana;
10. *Week* - número da semana. Inicia na semana 2 e termina na semana 60. Nesta demonstração, as semanas 53 a 60 contêm valores previstos;
11. *Predicted* - uma *flag* que indica se o consumo de eletricidade da respetiva semana foi previsto, ou se é histórico.

Todos estes atributos foram integrados numa tabela Hive (Figura 45), destinada a armazenar os dados de uma forma estruturada e apropriada para a análise visual.

state_abbreviation	state_name	city	cluster	model	electricity
AK	Alaska	Anchorage	cluster4	M5P	235
AK	Alaska	Anchorage	cluster4	M5P	216
electricity_cluster_avg	gas	gas_cluster_avg	week	predicted	
225	2177	1767	2	False	
220	1481	1588	3	False	

Figura 45. Amostra do conjunto de dados final para análise visual. Retirado de (Costa & Santos, 2015).

Uma vez que o serviço proposto inclui funcionalidades que permitem processar dados de consumo energético, assim como realizar a sua segmentação e efetuar previsões, novas perspetivas de como fornecer *feedback* às casas podem emergir. A Figura 46 mostra como é possível reinventar a fatura de energia para uma casa em New York, por exemplo. A análise começa por comparar a casa em New York com o *cluster* em que é inserida, apresentando os consumos médios de energia. Posteriormente, o gráfico de linhas sobrepõe o consumo de eletricidade da casa (linha azul escuro) com o consumo do *cluster* (linha azul claro), contendo não só dados históricos mas também valores previstos (linha laranja escuro para a casa e linha laranja claro para o *cluster*), resultantes da aplicação do modelo de *time series forecasting*.

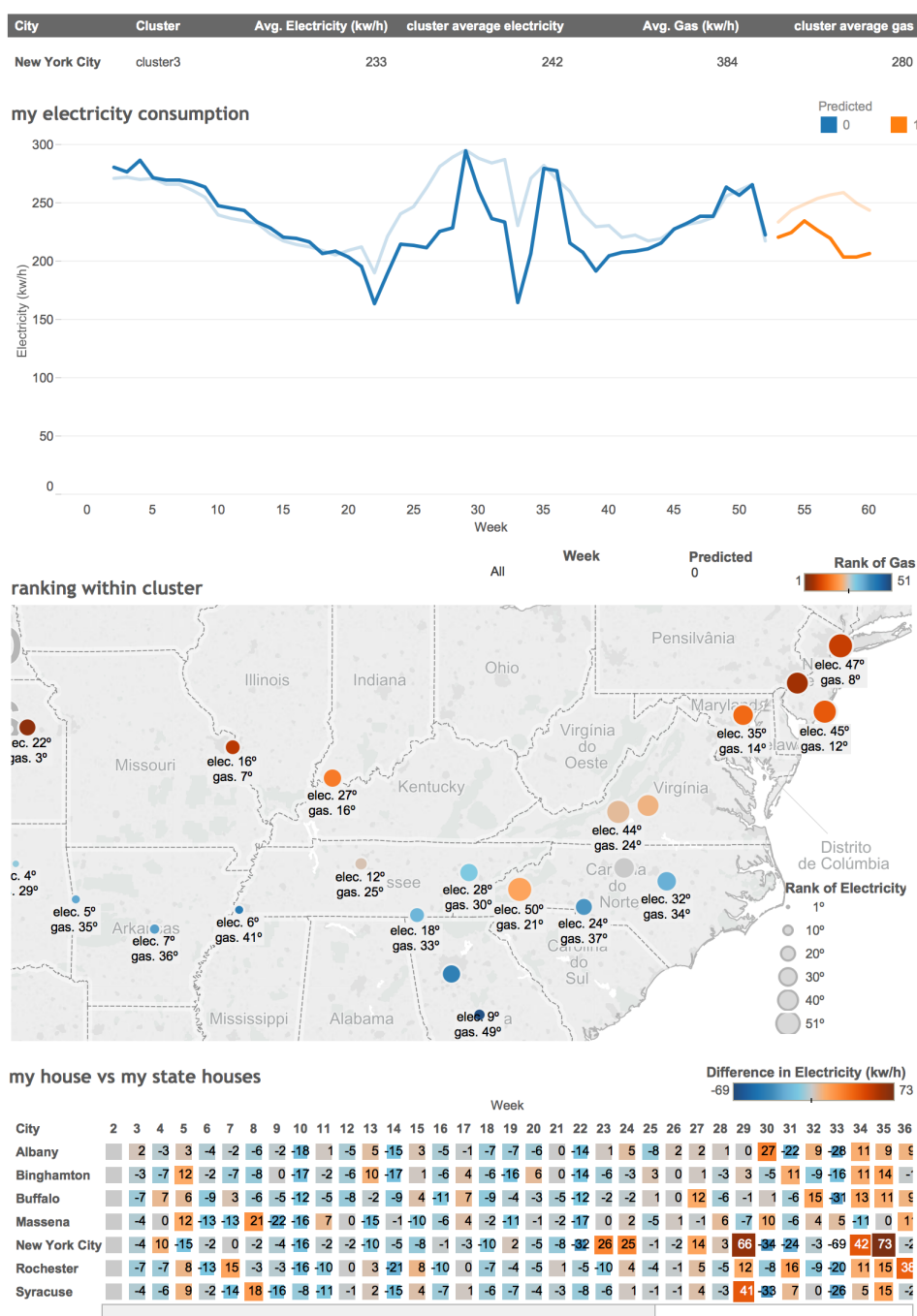


Figura 46. Fatura de energia reinventada. Retirado de (Costa & Santos, 2015).

Além disso, é também possível fazer o *ranking* das casas pelo seu consumo de energia, comparando uma certa casa com outras dentro do seu *cluster*. Esse *ranking* é ilustrado no mapa geográfico. O último gráfico na Figura 46 mostra um mapa de calor, com o objetivo de comparar a casa em New York com outras casas dentro do mesmo estado.

A análise, acima descrita, exemplifica o quão inovadores e inteligentes podem ser os serviços de uma *Smart City* e, neste caso particular, como o *clustering* e *time series forecasting* podem ser

combinados de modo a formar uma análise visual que contribui significativamente para que os cidadãos tenham uma experiência de consumo mais controlada, monitorizando dados históricos ou previstos e comparando a sua casa com outras dentro do seu *cluster* ou estado (ou distrito, região...). A Figura 47 descreve um exemplo da experiência associada ao serviço de consumo inteligente de energia proposto, baseado nos dados demonstrativos da Figura 46.

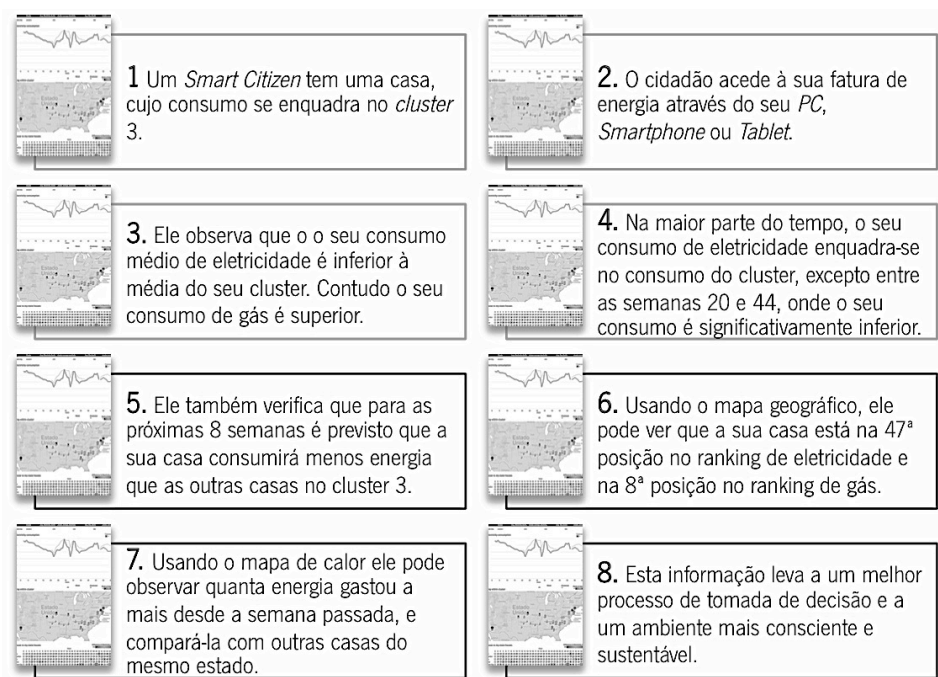


Figura 47. Demonstração da experiência de utilização associada à fatura de energia reinventada.

As tradicionais faturas de energia não fornecem uma visão clara do nosso consumo energético. Estas, obviamente, mostram-nos quanta energia gastamos em determinado período, mas se quisermos tornar as nossas cidades mais inteligentes e colocar arquiteturas como esta em aplicações de contexto real, a fatura de energia poderá: mostrar como a nossa casa se compara com a média no nosso *cluster* e como se comparará no futuro se continuarmos a gastar energia da forma como gastamos; ilustrar num mapa geográfico qual o *ranking intra-cluster* da nossa casa, usando dados históricos e previstos; mostrar como a nossa casa se compara com outras do mesmo estado ou região, numa base semanal. Estes são alguns dos possíveis exemplos.

Outra perspetiva acerca dos dados que este estudo providencia revela-se útil para fornecedores de energia e para o governo. Atualmente, os governos enfrentam dificuldades na gestão de recursos, e a energia é um deles. A Figura 46 apresentou uma fatura de energia reinventada, enquanto que na Figura 48 pode-se observar uma nova forma de monitorizar e segmentar energia.

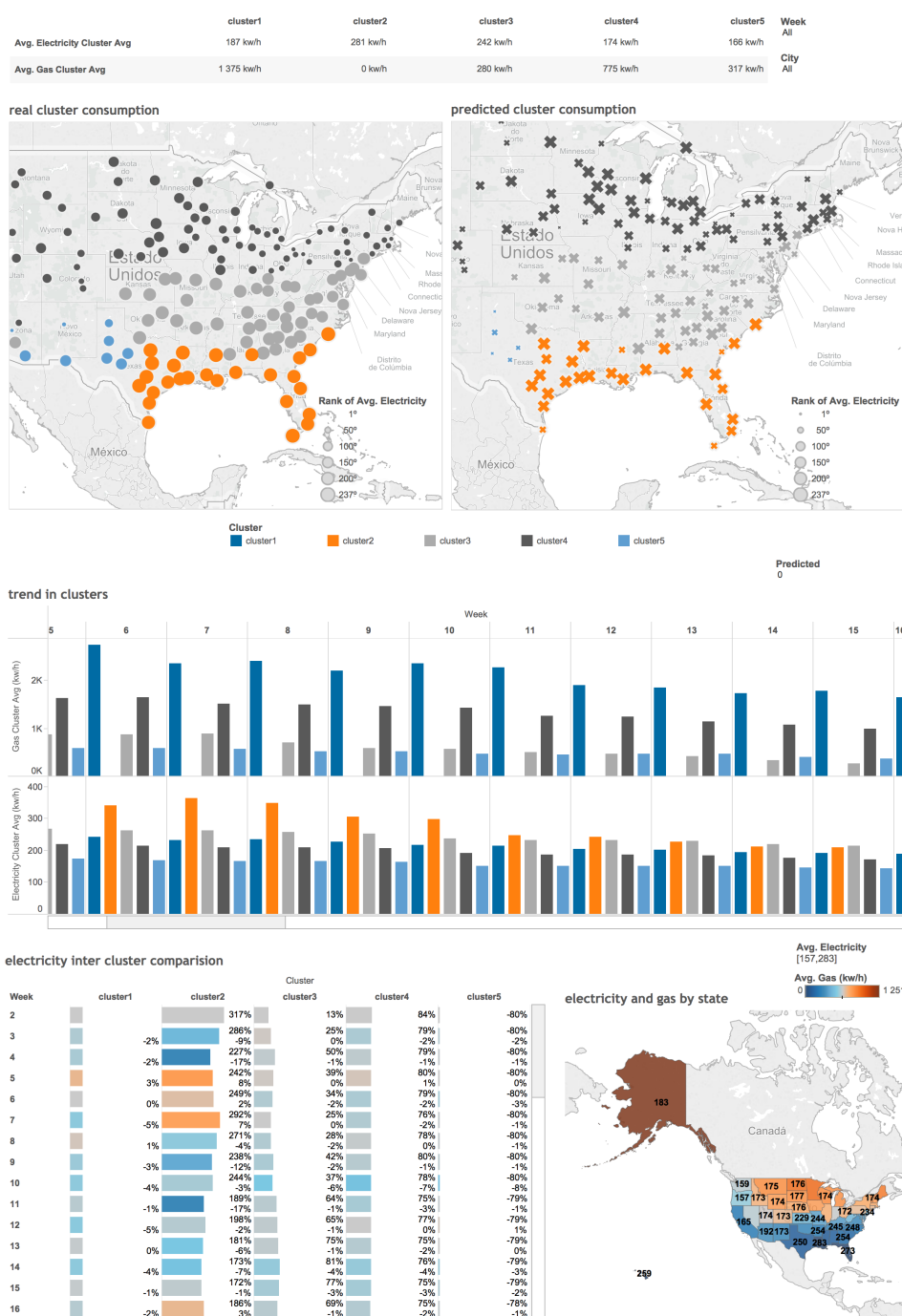


Figura 48. Monitorização e segmentação do consumo de energia. Retirado de (Costa & Santos, 2015).

Nesta visualização temos a capacidade de comparar os valores médios dos *clusters* e observar mudanças previstas no *ranking* de consumo dos mesmos. Além disso, a tendência semanal de consumo de eletricidade e gás pode também ser analisada e pode ser estabelecida uma comparação *inter-cluster*, comparando o consumo de eletricidade de cada um dos *clusters* e a respetiva evolução em relação às semanas anteriores. O governo das *Smart Cities* pode gerir os seus recursos muito mais facilmente, e os fornecedores de energia podem segmentar e distribuir a mesma, baseando-se numa visão mais

abrangente, como a agregação por *cluster*. Além disso, poderão prever quanta energia será necessária durante os próximos dois meses e como o *ranking* dos *clusters* se irá alterar. Todas estas perspetivas na análise de dados conduzem a um processo de tomada de decisão mais enriquecido.

6.1.5 Síntese

Este trabalho apresentou uma instanciação da camada tecnológica da arquitetura BASIS, contemplando o armazenamento, processamento, *clustering*, *forecasting* e visualização de dados. A abordagem apresentada neste trabalho é adequada para suportar o armazenamento e processamento de todos os dados. Contudo, algumas questões ainda não se encontram desenvolvidas, como a execução do Hadoop em vários nós, as formas de disponibilização de dados para outras plataformas como aplicações *mobile*, *Web* ou portais *Open Data*, e questões de segurança.

Os componentes de *Data Mining* mostraram resultados interessantes, dado que foi alcançada uma significativa variedade de *clusters*, como os centróides demonstram. O algoritmo K-means foi também capaz de alcançar uma similaridade *intra-cluster* reduzida. Além disso, uma taxa de erro tão reduzida em quase todos os *forecasters* testados revelou-se um resultado bastante satisfatório. Finalmente, a análise visual de dados fundiu todos os resultados numa experiência de utilização refinada, com vista a validar, com sucesso, a arquitetura e o seu subjacente serviço de consumo inteligente de energia.

Se iniciativas de *Smart Cities* começarem a ter lugar em Portugal, será possível recolher dados dos *Smart Meters* e disponibilizar aos cidadãos uma nova forma de auto-monitorizarem o seu consumo de eletricidade e gás, comparando-se com outras casas dentro do seu *cluster* (ou cidade, distrito, região...) e prevendo consumos de energia futuros. Além disso, os fornecedores de energia e as cidades terão uma melhor visão do panorama da energia, comparando *clusters* e observando previsões.

Convém também salientar que o grau de implementação já alcançado serve como prova de conceito para a arquitetura BASIS, que providencia serviços inovadores aos cidadãos e cidades. Que tenhamos conhecimento, nenhum outro trabalho providencia tal ambiente integrado, capaz de disponibilizar informação útil que conduza ao comportamento pró-ativo na população.

Para investigação futura é de salientar as questões que ainda não se encontram desenvolvidas neste serviço, como a execução do Hadoop em vários nós, a integração com aplicações *mobile*, *Web* ou portais *Open Data*, e questões de segurança. A sazonalidade do consumo poderá ser também integrada

em trabalhos futuros. Para isso, deve ser utilizado um conjunto de dados com múltiplos anos de consumo energético.

6.2 Serviço de Aeroportos Inteligentes

Um dos meios de transporte com elevada significância na mobilidade presente numa grande cidade é o transporte aéreo. A chegada e partida de pessoas através de voos representa um dos principais meios de entrada e saída de cidadãos nas grandes áreas metropolitanas. Num mundo em que o recurso tempo é cada vez mais valioso para um vasto conjunto de cidadãos, torna-se relevante antever situações em que esse recurso possa ser desperdiçado, o que por si só acarreta não só implicações a nível pessoal, mas também a nível das companhias aéreas e aeroportos. Este serviço demonstra como podemos usar a arquitetura proposta nesta dissertação para encontrar perfis de atraso nos voos ao nível da região, companhia aérea e aeroporto.

Para a demonstração deste serviço é utilizado um conjunto de dados de performance dos voos nos vários aeroportos dos Estados Unidos da América (RITA-BTS, 2015). O objetivo principal deste serviço consiste, por um lado, em disponibilizar aos cidadãos, cuja mobilidade seja significativamente dependente do transporte aéreo, uma análise visual de perfis de atraso nos voos, permitindo assim que o mesmo planeie as suas deslocações evitando viagens em determinadas companhias, ou salvaguardando atrasos frequentes em voos para determinadas localizações. Por outro lado, os aeroportos e companhias aéreas podem compreender de forma mais adequada a frequência com que ocorrem determinados atrasos, que poderão ser consequência de falhas existentes nos processos de atuação interna.

Este serviço faz uso de tecnologias como o Hadoop, Cassandra, Talend Open Studio for Big Data, Spark e várias linguagens de programação (Java, Javascript, HTML, CSS) para armazenar e processar *Big Data*, produzir *clusters* e visualizar/disponibilizar informação.

Os resultados alcançados revelam que usando este serviço é possível identificar diversos perfis de atraso nos voos, tendo em conta atributos como o atraso devido à companhia, meteorologia, sistema nacional de aviação, segurança ou atraso devido ao próprio avião. Demonstra-se também que, usando esta instanciação da camada tecnológica e infraestrutural da arquitetura BASIS, é possível criar modelos de *clustering* utilizando fontes de dados de elevado volume, como é exemplo o conjunto de 3 anos de voos usado neste serviço (aproximadamente 4 *gigabytes* e 18 milhões de linhas). A abordagem proposta apresenta-se não só como uma forma eficiente de encontrar perfis de atraso nos voos, mas também como uma forma de ultrapassar as limitações do *Data Mining* em ferramentas de processamento não

distribuído, assegurando que em ambientes de aplicação em contexto real poder-se-á produzir um modelo de *clustering* sobre conjuntos de dados de elevada dimensão, independentemente do contexto.

6.2.1 Identificação de Perfis de Atraso nos Voos

Entre o conjunto de literatura, não foram encontradas abordagens diretamente relacionadas com a proposta neste serviço, isto é, tentar encontrar perfis de atraso nos voos, utilizando um conjunto de dados de vasta dimensão. Contudo, destacam-se algumas abordagens de *Data Visualization*, com o objetivo principal de encontrar padrões nos dados, usando técnicas de visualização apropriadas:

1. Wicklin & Allison (2009) propõem uma análise visual do congestionamento presente no tráfego aéreo, utilizando 120 milhões de observações, com o principal objetivo de encontrar acontecimentos causados pelo parâmetro tempo, espaço ou companhia aérea;
2. Hofmann et al. (2009) integram um diverso conjunto de análises que se debruçam sobre vários indicadores, entre eles os cancelamentos, atrasos, eficiência de combustível, entre outros;
3. Por fim, Wickham (2009) adota uma abordagem de comparação direta entre dois aeroportos, no que concerne ao tráfego aéreo.

Surgem também alguns trabalhos relacionados com a previsão de atrasos através do uso de *Data Mining* (Klein, Craun, & Lee, 2010; Rebollo & Balakrishnan, 2014). No entanto, nenhum dos autores aborda a problemática de encontrar perfis de atraso nos voos, através de *clustering*. Além disso, não é abordada a problemática do volume dos dados. Tendo em conta o elevado número de registos presentes neste conjunto, é de salientar a importância de tal abordagem. A identificação destas lacunas constituem a base motivacional do trabalho aqui proposto, conjuntamente com a necessidade de assegurar técnicas de *Data Mining* distribuído no contexto de uma *Smart City*.

6.2.2 Análise dos Dados e Respetiva Qualidade

O conjunto de dados utilizado consiste em 36 ficheiros *CSV*, em que cada um deles contém informação acerca dos voos dos Estados Unidos da América, desde as durações planeadas aos atrasos e cancelamentos verificados. A Tabela 9 apresenta os atributos que foram utilizados neste serviço, assim como o significado dos mesmos.

Os atributos referentes ao nome da companhia aérea e do aeroporto foram extraídos através da junção entre os dados dos voos e os dados suplementares fornecidos publicamente pela American Statistical Association (ASA, 2009).

Tabela 9. Descrição dos atributos usados na análise dos atrasos nos voos.

<month><year>.csv	
<i>year</i>	Ano em que o voo ocorreu.
<i>month</i>	Mês.
<i>day_of_month</i>	Dia do mês.
<i>day_of_week</i>	Dia da semana.
<i>quarter</i>	Trimestre.
<i>crs_dep_time</i>	Tempo planeado de partida.
<i>tail_num</i>	Número do avião.
<i>arr_delay_15</i>	Flag que indica se o voo chegou atrasado mais do que 15 minutos.
<i>arr_delay_min</i>	Atraso da chegada em minutos.
<i>carrier_name</i>	Nome da companhia aérea.
<i>crs_arr_time</i>	Tempo planeado de chegada.
<i>dest_airport_name</i>	Nome do aeroporto de chegada.
<i>dest_city_name</i>	Nome da cidade do aeroporto de chegada.
<i>dest_state_code</i>	Código do estado do aeroporto de chegada.
<i>dest_state_name</i>	Nome do estado do aeroporto de chegada.
<i>origin_airport_name</i>	Nome do aeroporto de partida.
<i>origin_city_name</i>	Nome da cidade do aeroporto de partida.
<i>origin_state_code</i>	Código do estado do aeroporto de partida.
<i>origin_state_name</i>	Nome do estado do aeroporto de partida.
<i>carrier_delay</i>	Atraso, em minutos, devido a questões internas à companhia aérea.
<i>weather_delay</i>	Atraso, em minutos, devido a questões meteorológicas.
<i>nas_delay</i>	Atraso, em minutos, devido a questões do sistema nacional de aviação.
<i>security_delay</i>	Atraso, em minutos, devido a questões de segurança.
<i>late_aircraft_delay</i>	Atraso, em minutos, devido ao atraso do avião em questão.

No que respeita a problemas identificados na qualidade dos dados, é de salientar a ausência de valores nos atributos “*crs_arr_time*” e “*crs_dep_time*”, em certos registos. Por conseguinte, optou-se por remover estes mesmos registos, já que representavam uma fração pouco significativa do conjunto total. Além do problema acima descrito, não foram identificados quaisquer outros problemas em relação à qualidade dos dados.

Já na fase de *Analytical Processing* começa-se por corrigir as falhas na qualidade dos dados, supra mencionadas, e complementar os dados dos voos com os nomes dos aeroportos e companhias aéreas. Após o tratamento dos dados, os respetivos ficheiros voltam a ser armazenados no HDFS, para posterior transformação no Spark (www.spark.apache.org).

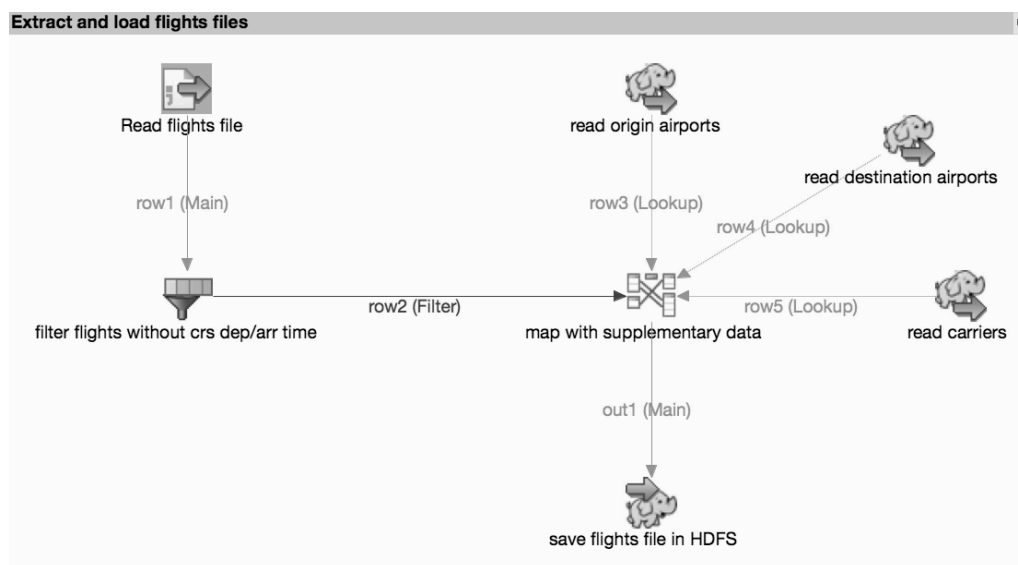


Figura 50. Extração e Carregamento dos dados dos voos.

No que respeita às transformações aplicadas no Spark, estas consistem em duas grandes tarefas: selecionar apenas os registos que correspondem a atrasos e agrupar os vários tipos de atraso em classes, onde cada classe corresponde a um intervalo de 15 minutos. Esta última transformação permite criar uma abstração, em que a cada 15 minutos de atraso acresce um nível de gravidade do mesmo. O resultado final desta transformação, cujo volume corresponde a cerca de 3.500.000 registos, é utilizado como *input* do algoritmo de *clustering* utilizado no Spark, o K-means. Após a execução do mesmo, os dados resultantes são armazenados numa *column family* no Cassandra, incluindo o *cluster* a que pertence cada registo.

No que diz respeito à utilização das estratégias de visualização, este serviço diferencia-se do anteriormente elaborado, na medida em que foi explorada a utilização de *Web services* para a disponibilização de dados em aplicações *Web*, em vez da utilização de uma ferramenta de *Data Visualization*, como o Tableau. Por conseguinte, implementou-se um conjunto de *Web services*, desenvolvidos em Java, que retornam os dados armazenados no Cassandra, permitindo a utilização do Google Charts para criação de gráficos numa página *Web*. O Cassandra foi a base de dados *NoSQL* designada para este serviço, de modo a conseguirmos estabelecer algum ponto de comparação com o HBase, no que respeita à usabilidade destas tecnologias.

A seguinte subsecção apresenta a instanciação da camada infraestrutural utilizada no desenvolvimento deste serviço, destacando o uso de armazenamento e processamento distribuído.

6.2.4 Instanciação da Camada infraestrutural

Na camada infraestrutural de suporte ao serviço de consumo inteligente de energia foram utilizados componentes de armazenamento e processamento distribuído (HDFS, HBASE, Pig, Hive) num ambiente suportado por uma única máquina virtual. No entanto, esta decisão foi devidamente ponderada, pois os recursos utilizados revelavam-se suficientes para a prova de conceito a que nos propusemos, cujas tarefas de *Data Mining* podiam ser adequadamente executadas numa única máquina, uma vez que quase toda a carga computacional assentava no processo de agregação dos dados. Uma vez que nesse serviço era necessário agregar cerca de 2.000.000 de registos, o uso do Hadoop em máquina virtual demonstrou ser adequado. No entanto, é expectável a necessidade de utilização do Hadoop distribuído por vários nós, de modo a assegurar as funcionalidades do serviço no contexto de uma iniciativa real, com dados reais.

Contudo, o volume apresentado pelo conjunto de dados utilizado no serviço de aeroportos inteligentes remete-nos para a utilização de um ambiente de *Data Mining* distribuído, pois a criação de perfis de atraso nos voos não engloba uma forte componente de agregação como pré-processamento ao respetivo *Data Mining*. Assim sendo, foi definido como meta encontrar *clusters* num *dataset* com o mesmo nível de granularidade do conjunto de dados original, causando assim um significativo *overhead* na fase de *Data Mining*, ao contrário do sucedido no serviço anterior, em que o processamento intensivo encontrava-se na fase de agregação dos dados. Por conseguinte, adota-se assim uma infraestrutura em que a carga computacional exigida pelo *clustering* é distribuída por vários nós que trabalham em conjunto, neste caso utilizando o Spark.

A Figura 51 demonstra a infraestrutura utilizada para desenvolver o serviço aqui apresentado, de modo a garantir a extração de *clusters* em 36 ficheiros, contendo cerca de 18.000.000 de voos, em que mais de 3.500.000 apresentam atrasos, caso que corresponde ao objeto de análise em questão.

Na instalação distribuída da infraestrutura foram utilizadas 3 máquinas. A infraestrutura representada na Figura 51 deve ser entendida como uma solução ajustada aos recursos disponíveis, isto é, verifica-se a sobreposição de várias responsabilidades no que diz respeito à máquina 192.168.1.100, devido ao facto de oferecer melhor desempenho. Pode-se observar que esta máquina é responsável não só pelo desenvolvimento de *ETL/ELT* e *Data Mining*, mas também pela execução do HDFS, do servidor

Web e pelo significativo contributo no *cluster* Spark. Em ambientes de desenvolvimento, tal como o que aqui se descreve, os recursos podem ser repartidos, existindo algumas sobreposições de responsabilidades, desde que sejam assegurados os requisitos mínimos das várias ferramentas utilizadas. No entanto, em ambiente de produção, não é aconselhável a acumulação de demasiadas responsabilidades por uma só máquina.

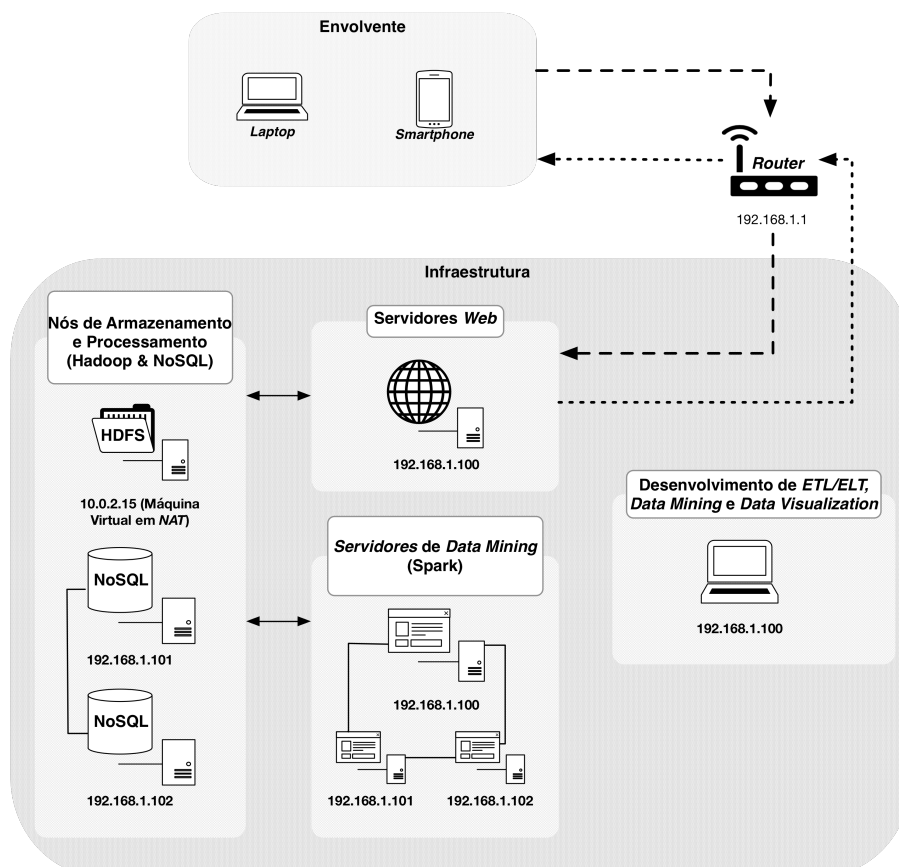


Figura 51. Infraestrutura utilizada no serviço de aeroportos inteligentes.

Na infraestrutura deste serviço, os acontecimentos processam-se da seguinte forma:

1. Um objeto da envoltório (*laptop* ou *smartphone*) requer a página *Web*, que corresponde à visualização dos dados;
2. O servidor *Web* lê os dados armazenados no *cluster NoSQL* (Cassandra), utilizando *Web services REST*, de modo a garantir que a disponibilização dos dados é independente da plataforma que acede aos mesmos.
3. O servidor *Web* devolve a página que foi requerida pelo objeto da envoltório, disponibilizando, assim, a análise visual referente aos perfis de atraso nos voos.

6.2.5 Clustering

O Spark foi a tecnologia designada para assegurar o treino e teste do modelo de *clustering* K-means, cuja carga de trabalho foi distribuída por três máquinas, um Spark *Master* e três *Spark Workers*, sendo que uma das máquinas assume, simultaneamente, o papel de *Master* e *Worker*. O *cluster* utilizado dispõe de 8 núcleos de processamento e 6 *gigabytes* de *RAM*, o que totaliza a contribuição de *hardware* definida para cada um dos nós, de acordo com a Figura 52, onde se pode observar a página de monitorização do *cluster* Spark. Utilizando esta configuração, os tempos de execução para a resolução deste problema rondam os 4,5 minutos.

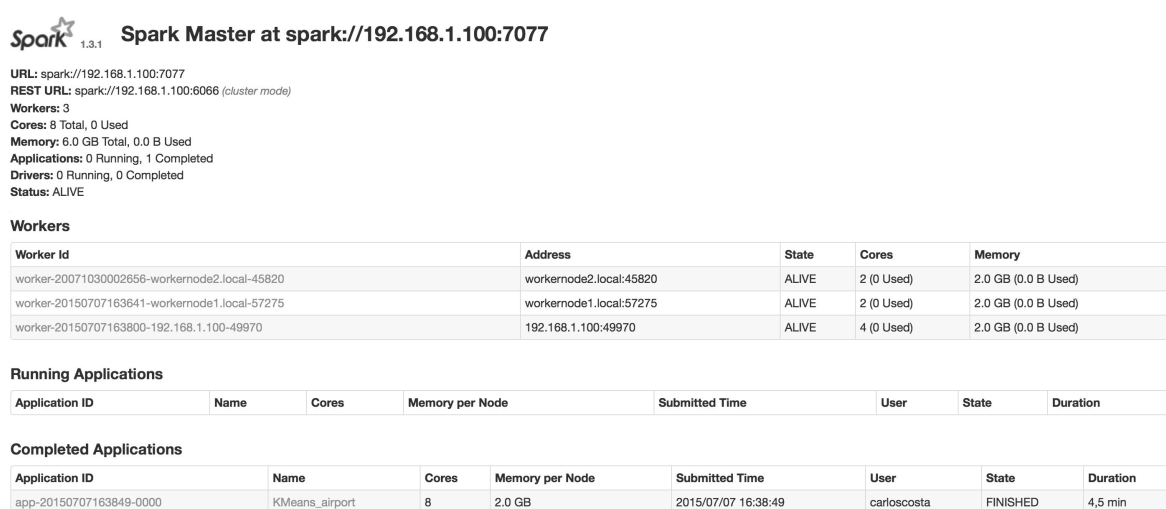


Figura 52. Página de monitorização do *cluster* Spark.

Uma vez que os dados já se encontravam armazenados no HDFS, que atuaria como a fonte de dados para o algoritmo de *clustering*, programou-se a seguinte *script* em Python², com o objetivo de treinar e testar o modelo:

```
from pyspark import SparkContext
from pyspark.mllib.clustering import KMeans
from numpy import array
from math import sqrt, ceil
from pyspark.sql import SQLContext

sc = SparkContext(appName="KMeans_airport")
sqlContext = SQLContext(sc)

# Ler todos os ficheiros dos voos guardados no HDFS
df = sqlContext.load(source="com.databricks.spark.csv",
    header="true",
```

² Python em Spark demonstra ser uma linguagem bastante acessível para efetuar transformações aos dados ou executar algoritmos de *Data Mining*, tirando partido das possibilidades oferecidas pela computação distribuída.

```

path = "hdfs://quickstart.cloudera:8020/user/cloudera/smart_airport/flights/flights_monthly/*.csv",
delimiter=";")

# Para cada registo, extrair apenas os atrasos e agrupá-los em classes de atrasos por cada 15 minutos
parsedData = df.filter(df.ARR_DEL15 >= 1).map(lambda row: array([divideBy15(row.CARRIER_DELAY),
    divideBy15(row.WEATHER_DELAY),
    divideBy15(row.NAS_DELAY),
    divideBy15(row.SECURITY_DELAY),
    divideBy15(row.LATE_AIRCRAFT_DELAY)]))

# Construir o modelo de clustering K-means. Nota: exemplo com 14 clusters.
clusters = KMeans.train(parsedData, 14, maxIterations=100, runs=10, initializationMode="k-means | |")
# Calcular a similaridade intra-cluster
WSSSE = parsedData.map(lambda point: error(point)).reduce(lambda x, y: x + y)
print("Within Set Sum of Squared Error = " + str(WSSSE))

sc.stop()

##### Funções #####

# Função utilizada para agrupar os valores dos atrasos em classes de 15 minutos (exemplo: 0-15, 16-30,...)
def divideBy15(value):
    return ceil(float(value) / 15)

# Função para avaliar os clusters produzidos através do cálculo da similaridade intra-cluster
def error(point):
    center = clusters.centers[clusters.predict(point)]
    return sqrt(sum([x**2 for x in (point - center)]))

```

O teste para encontrar o número adequado de *clusters* consistiu em solicitar ao Spark a execução da *script* acima mencionada, num total de 13 vezes, em incrementos de 2, desde 2 até 26 *clusters*. A Figura 53 representa o valor do erro (medida de similaridade *intra-cluster*) encontrado para cada incremento, numa tentativa de encontrar o “joelho” da curva (Salvador & Chan, 2004).

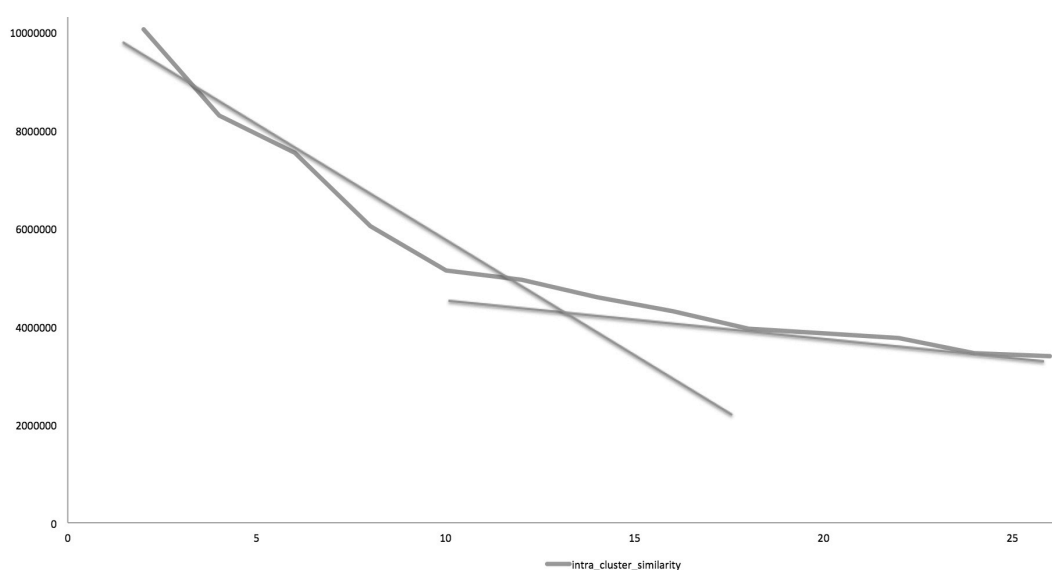


Figura 53. Similaridade *intra-cluster* nos atrasos dos voos.

Observando o respetivo gráfico, pode-se concluir que o número adequado de *clusters* situar-se-á próximo de 14, sendo este o valor assumido neste trabalho. Encontrado o número adequado de *clusters*, o algoritmo foi executado pela última vez, de modo a extrair o valor dos centróides e classificar cada registo com o *cluster* a que o mesmo pertence. Posteriormente, os resultados foram armazenados no Cassandra. A Tabela 10 apresenta os centróides de cada um dos 14 *clusters* produzidos.

Tabela 10. Centróides dos *clusters* de atrasos nos voos.

<i>CLUSTER</i>	<i>CARRIER_DELAY</i>	<i>WEATHER_DELAY</i>	<i>NAS_DELAY</i>	<i>SECURITY_DELAY</i>	<i>LATE_AIRCRAFT_DELAY</i>
<i>cluster0</i>	0,2496	0,1679	1,8281	0,0164	0,1824
<i>cluster1</i>	0,1692	0,0439	12,6223	0,0005	0,6016
<i>cluster2</i>	0,7632	0,1759	0,5922	0,0036	15,4051
<i>cluster3</i>	0,5337	0,0806	0,5236	0,0027	4,7549
<i>cluster4</i>	0,1067	8,0601	1,1420	0,0003	0,7265
<i>cluster5</i>	0,2416	0,0968	5,1984	0,0019	0,4396
<i>cluster6</i>	48,0886	0,0256	0,5286	0,0000	1,3590
<i>cluster7</i>	7,9434	0,0129	0,4807	0,0008	0,4883
<i>cluster8</i>	0,5874	0,0624	0,4184	0,0048	2,0312
<i>cluster9</i>	0,5687	0,1212	0,5865	0,0027	8,6790
<i>cluster10</i>	2,7613	0,0078	0,4183	0,0011	0,2630
<i>cluster11</i>	1,1182	0,1736	0,5163	0,0014	29,0314
<i>cluster12</i>	0,1441	26,2569	1,2802	0,0000	0,8658
<i>cluster13</i>	17,7202	0,0167	0,5003	0,0002	0,4437

Pode-se observar que 14 *clusters* garantem uma relação adequada entre a similaridade *intra-cluster* e a variedade dos mesmos, pois segundo a Tabela 10 são identificados vários perfis de atraso nos voos, desde o “*cluster 0*” que apresenta um atraso moderado devido a questões relacionadas com o sistema nacional de aviação, ao “*cluster 13*” cujos elevados atrasos são marcados por questões internas à companhia aérea. Convém salientar novamente que os valores presentes na tabela são classes de atraso por cada 15 minutos, portanto, para encontrar uma aproximação ao valor do atraso em minutos basta multiplicar o mesmo por 15.

6.2.6 Visualização e Disponibilização dos Resultados

Uma vez que os resultados encontram-se armazenados no Cassandra, a última tarefa assenta na disponibilização de uma análise visual dos perfis de atraso encontrados. Neste serviço, optou-se por complementar a abordagem seguida no serviço anterior, implementando um conjunto de *Web services REST* em vez de optar por uma ferramenta de *Data Visualization*, como o Tableau, de modo a demonstrar o que se entende por independência na integração com qualquer plataforma. Isto permite

concluir, por exemplo, que a companhia presente na Figura 54 apresenta sobretudo os *clusters* 8 e 12 como perfis predominantes, evidenciando atrasos na ordem dos 30 minutos, devido a atrasos do próprio avião, e também uma certa predominância em atrasos de séria gravidade (cerca de 400 minutos), causados por questões meteorológicas. De realçar que outras visualizações podem ser desenvolvidas a partir dos resultados já produzidos.

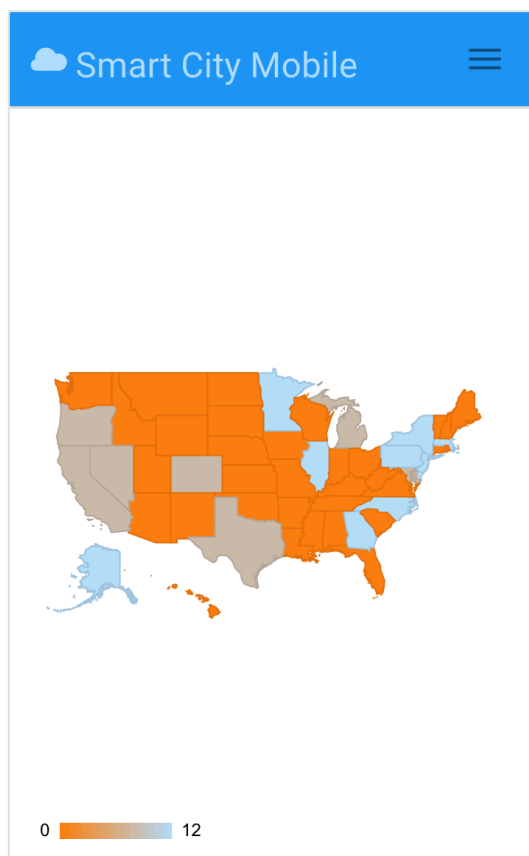


Figura 55. Visualização, num dispositivo móvel, dos perfis de atraso nos voos.

6.2.7 Síntese

A abordagem aqui proposta, além de disponibilizar aos cidadãos uma forma de complementar a sua decisão quanto ao planeamento de uma determinada viagem (hora de chegada, companhia selecionada, entre outros), valida a arquitetura BASIS, no que respeita à disponibilização pública de dados, garantindo a independência perante a plataforma que acede aos mesmos, através do uso de padrões arquiteturais, como o *REST*. Além disso, e talvez o contributo mais significativo, este trabalho demonstra que em iniciativas futuras, onde o volume de dados apresentado possa levantar dificuldades no processamento do mesmo, será possível executar *Data Mining* nesses conjuntos de dados, em ambientes de computação distribuída, não só usando a técnica de *clustering*, como aqui validado, mas

também outras técnicas, como por exemplo classificação, regressão e regras de associação. Tais técnicas podem ser desenvolvidas de raiz no Spark, ou usando a biblioteca de *Data Mining* do mesmo (MLlib).

O comportamento do *cluster* Spark revelou-se adequado, sendo que foram processados cerca de 18.000.000 de registos e após selecionados apenas os atrasos, construiu-se um modelo de *clustering* com cerca de 3.500.000 registos. A execução de todo este processo ocorreu num tempo significativamente reduzido de 4 minutos e 30 segundos, tendo em conta que o *cluster* faz uso de recursos significativamente limitados.

Com o desenvolvimento deste serviço foram colmatadas algumas questões de investigação pendentes após a concretização do serviço anterior, tais como o número ideal de nós presentes no *cluster* de armazenamento e processamento e a questão de disponibilização dos dados em aplicações *mobile* e *Web*. Neste sentido, conclui-se que a estratégia mais adequada em iniciativas de *Smart Cities* é planear o *cluster* de armazenamento e processamento (Hadoop e *NoSQL*) com recursos adequados para o tempo inicial do projeto e escalar à medida que as necessidades surjam, ou seja, não é necessário estabelecer um número ideal de máquinas, mas sim uma relação custo-benefício específica para cada iniciativa, sendo que a escalabilidade linear está garantida pela adoção das tecnologias abordadas nesta dissertação. Além disso, no que diz respeito à disponibilização dos dados para plataformas *mobile* ou *Web*, este serviço valida o facto de o padrão arquitetural *REST* se revelar adequado para este fim, garantindo independência perante a plataforma que acede aos dados.

Como investigação futura, resta ainda a exploração e validação das questões de segurança, sendo que embora tenham sido identificadas várias tecnologias cruciais na manutenção, administração e segurança deste tipo de sistemas, a sua exploração não foi contemplada em nenhum dos serviços desenvolvidos. Prevê-se que numa investigação futura esta preocupação possa ocupar o conjunto de objetivos prioritários.

Capítulo 7. Conclusões e Trabalho Futuro

Este documento começa por expor o enquadramento conceptual de *Big Data* no contexto de *Smart Cities*, explicando o que pode ser entendido como *Big Data*, a motivação que provocou o aparecimento do conceito, os desafios esperados e os principais paradigmas e termos associados ao mesmo, tais como o *MapReduce* e o *NoSQL*. Seguidamente, foram apresentados o conceito de *Smart City* e os desafios associados a estas iniciativas. O enquadramento conceptual terminou com a análise de arquiteturas, já existentes, de *Big Data* no contexto de *Smart Cities*, descrevendo-as, ilustrando-as e referenciando algumas limitações e características particulares, que motivam o aparecimento da arquitetura BASIS. Após o enquadramento conceptual, foram identificadas as principais técnicas e tecnologias envolvidas no processamento de *Big Data*, prestando particular atenção ao Hadoop, o *standard* atual para computação distribuída de dados, às bases de dados *NoSQL*, destinadas ao armazenamento de dados de natureza volumosa, variada e veloz e, por fim, ao *ETL/ELT* e *Big Data Analytics*.

Posteriormente, criou-se a especificação da arquitetura BASIS, detalhando cada característica e componente através da concepção de uma arquitetura por camadas: conceptual, tecnológica e infraestrutural. Também foram identificados os principais intervenientes na cocriação de serviços de uma *Smart City* e quais as suas principais responsabilidades. Tendo especificado a arquitetura, identificou-se a necessidade de descrever como atuaria o Portal *Open (Big) Data* e quais as principais questões de integração com outros componentes da arquitetura, prestando particular atenção à disponibilização pública dos dados.

Por último, criaram-se dois serviços inteligentes no âmbito de uma *Smart City*, com vista a validar a arquitetura BASIS: o primeiro serviço consiste em promover o consumo inteligente de eletricidade e gás, de modo a criar perfis de consumo e prever futuros consumos de eletricidade, comparando cada casa com o perfil em que é inserida; o segundo serviço baseia-se na identificação de perfis de atraso nos voos em certas companhias, aeroportos ou regiões, executando técnicas de *Data Mining* distribuído para encontrar *clusters* num conjunto de dados com cerca de 18.000.000 de voos.

7.1 Porquê *Big Data*? Implicações a Longo Prazo

À primeira vista, quando se começa a desenvolver uma iniciativa de *Smart City*, onde os dados assumem um dos papéis principais, pode parecer tentador considerar o uso exclusivo de tecnologias

tradicionais de armazenamento e processamento de dados, tais como as bases de dados relacionais e as ferramentas de *Data Mining* ou processamento analítico não distribuídos, devido à sua ampla adoção ao longo dos anos. Contudo, assumindo que desenvolvemos este tipo de iniciativas para uma sustentabilidade a longo prazo, principal motivação para criar uma *Smart City*, devemos preparar a cidade para o volume, variedade e velocidade dos dados que serão constantemente produzidos e que, consequentemente, terão de ser processados apropriadamente.

Por isso, nesta dissertação, pretendeu-se demonstrar a adequação das tecnologias de *Big Data* no contexto de uma *Smart City*, validando o desempenho da arquitetura BASIS através do desenvolvimento de dois serviços inteligentes. Apesar de se ter lidado com um volume considerável de dados na validação da arquitetura, convém esclarecer as possíveis implicações em contexto real, usando como analogia os serviços desenvolvidos.

Assuma-se como exemplo o serviço de consumo inteligente de energia. Neste serviço utilizou-se um conjunto de dados com cerca de 2.000.000 de registos, representando o consumo horário de eletricidade e gás em 237 casas. Contudo, se se transpuser o caso de estudo para, por exemplo, Lisboa, que conta com mais de 500.000 habitantes, já se consegue imaginar a dimensão do problema causado pelo volume de dados que poderiam ser recolhidos. De modo a formular a dimensão do problema, observe-se a equação 4.

$$Vol_{Lisboa} = num_{casas} * num_{anos} * 8760 \text{ registos} \quad (4)$$

Com vista a verificar alguns valores em causa, assuma-se que se recolhem os consumos de 50.000 casas. Apenas no período de um ano, a infraestrutura tem de ser capaz de armazenar e processar 438.000.000 de registos, mais de 100 *gigabytes* de dados (tendo em conta que 2.000.000 representavam 550 *megabytes*), tarefa impensável para qualquer ferramenta de processamento analítico não distribuído. Um problema semelhante ocorre refletido no segundo serviço. Caso se pretenda encontrar perfis de atraso nos transportes públicos em Lisboa, ou analisar zonas de elevado congestionamento no trânsito automóvel, por exemplo, a carga computacional colocada sobre o *Data Mining* seria totalmente desadequada para uma única unidade de processamento.

No que diz respeito à variedade, a adoção de tecnologias como o HDFS e as bases de dados *NoSQL* assegura que se consegue recolher novos dados, nos mais variados formatos, sem necessitar de reformular os modelos de dados já existentes, dado que não existe a noção de modelo rígido ou relacionamentos explícitos. Se, hipoteticamente, se pretendesse complementar a análise dos perfis e

previsão de consumo de energia, poder-se-ia, por exemplo, recolher um conjunto de dados com as várias tarifas de eletricidade e gás e recomendar a mais adequada para cada casa, sem afetar todo o trabalho desenvolvido até agora, tal é a flexibilidade destas tecnologias.

Por fim, a velocidade a que os dados são produzidos, recolhidos e disponibilizados representa um desafio relevante no desenvolvimento de serviços inteligentes no contexto de uma *Smart City*. Num contexto real, temos de ser capazes de desenvolver mecanismos de recolha eficazes, mas também garantir que a base de dados assegura tempos de leitura e escrita adequados ao serviço em questão. Por exemplo, no caso do serviço de consumo inteligente de energia, a cada hora poderão ser recolhidos dados de um vasto conjunto de sensores, solicitando um elevado número de operações de escrita. As bases de dados *NoSQL*, em particular o Cassandra, garantem tempos de escrita bastante reduzidos. Por outro lado, quando determinado utilizador consulta os seus consumos, é expectável que a base de dados responda com eficácia e eficiência. Tendo em consideração o volume de dados que pode ser alcançado num serviço deste género em contexto real, as bases de dados *NoSQL* são a tecnologia adequada.

7.2 Resultados Obtidos

Esta dissertação teve como finalidade estudar o papel do *Big Data* no contexto de *Smart Cities* e verificar as arquiteturas de suporte a estes sistemas, nomeadamente o seu potencial e principais limitações. Tal estudo permitiu dar cumprimento a um dos principais objetivos desta dissertação, associado à proposta de uma nova arquitetura que contemple vários princípios de concepção estabelecidos, com vista a ser validada pela criação e disponibilização de serviços inteligentes para *Smart Cities*. Tudo isto foi concretizado e descrito ao longo deste documento.

Os resultados obtidos nesta dissertação revelaram-se satisfatórios, uma vez que se conseguiu propor uma arquitetura capaz de cumprir todos os princípios de concepção estabelecidos:

1. Disponibilizar dados de forma aberta e possibilitar a criação de novos serviços por parte de toda a comunidade, tal como identificado na componente de Disponibilização de Dados e Modelos Analíticos e na componente de Serviços de Dados e Aplicações;
2. Contemplar várias camadas de abstração, através da abordagem adotada, que consistiu numa especificação por camadas;
3. Assegurar, com o Hadoop e as bases de dados *NoSQL*, o armazenamento e processamento distribuído;

4. Incluir preocupações de segurança, privacidade e confiança dos dados, alcançado através da especificação da componente de Administração, Monitorização e Segurança, demonstrando particular atenção à inclusão das tecnologias adequadas e das técnicas de preservação da privacidade na recolha de dados;
5. Incorporar preocupação com o ciclo de vida dos dados, assegurado pela inclusão de uma base de dados históricos na área de armazenamento da arquitetura, retratando uma forma de gerir o ciclo de vida dos dados. Além disso, técnicas inerentes a algumas bases de dados *NoSQL*, como o Cassandra, permitem lidar eficientemente com elevado volume de dados de natureza temporal, melhorando a leitura e escrita, conforme a distância temporal dos mesmos;
6. Definir as responsabilidades dos vários intervenientes na cocriação de serviços de uma *Smart City*, demonstrando os principais resultados provenientes da cooperação entre os mesmos, de modo a definir-se claramente os vários papéis relevantes numa iniciativa deste género;
7. Propor uma arquitetura *service-oriented* e independente da plataforma que acede aos dados, uma vez que qualquer disponibilização de dados é realizada através de *Web services*, tal como pôde ser validado no serviço de aeroportos inteligentes, através do uso do padrão arquitetural *REST*.
8. Fazer uso de tecnologias *open source*, exceto em casos onde a relação custo-benefício justifique outra escolha, como foi provado, por exemplo, pela utilização de vários componentes do Hadoop, do Cassandra, do Talend Open Studio for Big Data e do Weka. A única exceção apresentada foi o Tableau, devido maioritariamente às suas vastas capacidades analíticas, que acabaram por se revelar adequadas à criação do serviço de consumo inteligente de energia. Contudo, a transposição do mesmo para uma ferramenta de *Data Visualization open source* é perfeitamente admissível.

Quanto aos serviços desenvolvidos, demonstrou-se que os resultados produzidos pelos mesmos revelam uma arquitetura capaz de armazenar, processar e disponibilizar *Big Data* no contexto de *Smart Cities*, incluindo modelos de *Data Mining (clustering e time series forecasting)* com desempenhos adequados e reduzidas margens de erro. O número de tecnologias usadas é de considerável dimensão, permitindo retirar algumas conclusões acerca das que se alteraram do primeiro serviço para o segundo. Por exemplo, no que diz respeito ao Cassandra e ao HBase, embora não se tenha estabelecido uma comparação direta entre os dois, conseguiu-se concluir que a simplicidade na instalação e gestão de um

cluster Cassandra pode revelar-se uma mais valia na altura de seleccionar a base de dados adequada. No segundo serviço optou-se também pela substituição do WEKA pelo Spark, como ferramenta de *Data Mining*, devido ao volume de dados que foi processado. A principal diferença entre estas duas ferramentas assenta no facto de o Spark assegurar a execução dos algoritmos num ambiente distribuído, onde os limites de processamento vão muito além do que consegue ser alcançado com qualquer outra ferramenta tradicional de *Data Mining*.

Pode-se, assim, afirmar que para além do resultado final desta dissertação oferecer dois serviços inovadores cujas técnicas e tecnologias podem ser adotadas noutros contextos, disponibiliza sobretudo um artefacto que ambiciona ser um contributo significativo na representação padrão de uma arquitetura de *Big Data* adequada ao contexto de *Smart Cities*.

Para concluir esta seção que sistematiza os resultados obtidos com a realização deste trabalho, importa referir que uma componente deste trabalho foi já publicada numa conferência internacional, nomeadamente o serviço de consumo inteligente de energia (Costa & Santos, 2015), estando agora prevista a escrita de um artigo científico com a proposta de arquitetura e a concretização do serviço de aeroportos inteligentes.

7.3 Investigação Futura

Após a conclusão desta dissertação, restam alguns pontos da arquitetura que necessitam ser validados, de modo a garantir uma apropriada implementação em contexto real. O primeiro ponto proposto para investigação futura é a exploração das técnicas e tecnologias que garantirão a segurança do sistema e o funcionamento adequado do mesmo. Tanto o Hadoop como as bases de dados *NoSQL* requerem um significativo esforço de parametrização e configuração, que devem ser assegurados pelo correto estudo das diretrizes de administração, monitorização e segurança destas tecnologias.

Seguidamente, seria relevante validar o papel do *Big Data Analytics* disponibilizado num modelo “as a Service”, tal como proposto nesta arquitetura. A disponibilização pública de ferramentas de processamento analítico, *Data Mining* e *Data Visualization*, permite fomentar a criação de serviços por parte da comunidade, pois toda a complexidade associada ao uso das mesmas pode ser abstraída num conjunto de *Web services*, cuja invocação em aplicações externas providenciaria uma capacidade analítica que, provavelmente, seria difícil de alcançar por parte das aplicações desenvolvidas pela comunidade.

Por fim, e apesar de se terem alcançado resultados satisfatórios nos serviços desenvolvidos, quer a nível técnico quer a nível do contributo para os cidadãos, existe margem para melhorar os mesmos, nomeadamente incluir a análise da sazonalidade nas previsões do consumo de eletricidade.

Referências Bibliográficas

- Alahakoon, D., & Yu, X. (2013). Advanced analytics for harnessing the power of smart meter big data. In *2013 IEEE International Workshop on Intelligent Energy Systems (IWIES)* (pp. 40–45). <http://doi.org/10.1109/IWIES.2013.6698559>
- Alzate, C., & Sinn, M. (2013). Improved Electricity Load Forecasting via Kernel Spectral Clustering of Smart Meters. In *2013 IEEE 13th International Conference on Data Mining (ICDM)* (pp. 943–948). <http://doi.org/10.1109/ICDM.2013.144>
- Apache. (2014). Hadoop Distributions and Commercial Support. Obtido de <http://wiki.apache.org/hadoop/Distributions%20and%20Commercial%20Support> em Nov/2014.
- Apache. (2015a). Welcome to Apache Cassandra. Obtido de <http://cassandra.apache.org> em Jan/2015.
- Apache. (2015b). Welcome to Apache Hadoop. Obtido de <http://hadoop.apache.org> em Jan/2015.
- Arenas-Martínez, M., Herrero-Lopez, S., Sanchez, A., Williams, J. R., Roth, P., Hofmann, P., & Zeier, A. (2010). A Comparative Study of Data Storage and Processing Architectures for the Smart Grid. In *2010 First IEEE International Conference on Smart Grid Communications (SmartGridComm)* (pp. 285–290). <http://doi.org/10.1109/SMARTGRID.2010.5622058>
- ASA. (2009). ASA Data Expo 2009 Supplemental data. Obtido de <http://stat-computing.org/dataexpo/2009/supplemental-data.html> em Jun/2015
- Azevedo, F., & Vale, Z. A. (2006). Forecasting Electricity Prices with Historical Statistical Information using Neural Networks and Clustering Techniques. In *Power Systems Conference and Exposition, 2006. PSCE '06. 2006 IEEE PES* (pp. 44–50). <http://doi.org/10.1109/PSCE.2006.296247>
- Baru, C., Bhandarkar, M., Nambiar, R., Poess, M., & Rabl, T. (2013). Benchmarking Big Data Systems and the BigData Top100 List. *Big Data*, 1(1), 60–64. <http://doi.org/10.1089/big.2013.1509>
- Batty, M., Axhausen, K. W., Giannotti, F., Pozdnoukhov, A., Bazzani, A., Wachowicz, M., ... Portugali, Y. (2012). Smart cities of the future. *European Physical Journal-Special Topics*, 214(1), 481.

- Brewer, E. (2012). CAP twelve years later: How the “rules” have changed. *Computer*, 45(2), 23–29.
<http://doi.org/10.1109/MC.2012.37>
- Caragliu, A., Del Bo, C., & Nijkamp, P. (2009). *Smart cities in Europe* (Serie Research Memoranda No. 0048). VU University Amsterdam, Faculty of Economics, Business Administration and Econometrics.
 Retrieved from <https://ideas.repec.org/p/vua/wpaper/2009-48.html>
- Cattell, R. (2011). Scalable SQL and NoSQL Data Stores. *SIGMOD Rec.*, 39(4), 12–27.
<http://doi.org/10.1145/1978915.1978919>
- Chandarana, P., & Vijayalakshmi, M. (2014). Big Data analytics frameworks. In *2014 International Conference on Circuits, Systems, Communication and Information Technology Applications (CSCITA)* (pp. 430–434). <http://doi.org/10.1109/CSCITA.2014.6839299>
- Changqing Ji, Yu Li, Wenming Qiu, Yingwei Jin, Yujie Xu, Uchechukwu Awada, ... Wenyu Qu. (2012). Big Data Processing: Big Challenges and Opportunities. *Journal of Interconnection Networks*, 13(3/4), 1–19. <http://doi.org/10.1142/S0219265912500090>
- Chen, H., Chiang, R. H., & Storey, V. C. (2012). Business Intelligence and Analytics: From Big Data to Big Impact. *MIS Quarterly*, 36(4), 1165–1188.
- Chen, M., Mao, S., & Liu, Y. (2014). Big Data: A Survey. *Mobile Networks and Applications*, 19(2), 171–209. <http://doi.org/10.1007/s11036-013-0489-0>
- Chignard, S. (2013). A brief history of Open Data. Obtido de <http://www.paristechreview.com/2013/03/29/brief-history-open-data/> em Maio/2015.
- Coelho, F. A. C. L. (2013). Implementation and test of transactional primitives over Cassandra. Retrieved from <http://repositorium.sdum.uminho.pt/handle/1822/27870>
- Commercial and Residential Reference Building Models. (2013). Commercial and Residential Hourly Load Profiles for all TMY3 Locations in the United States [Catalog]. Retrieved from

- <http://en.openei.org/doe-opendata/dataset/commercial-and-residential-hourly-load-profiles-for-all-tmy3-locations-in-the-united-states>
- Costa, C., & Santos, M. Y. (2015). Improving Cities Sustainability through the Use of Data Mining in a Context of Big City Data. In *The World Congress on Engineering* (Vol. 1). The 2015 International Conference of Data Mining and Knowledge Engineering: IAENG.
- Cumbley, R., & Church, P. (2013). Is “Big Data” creepy? *Computer Law & Security Review*, 29(5), 601–609. <http://doi.org/10.1016/j.clsr.2013.07.007>
- Davenport, T. H. (2006). Competing on analytics. *Harvard Business Review*, 84(1), 98–107, 134.
- Edlich, S. (2014). NoSQL - Your Ultimate Guide to the Non - Relational Universe [Blogue]. Obtido de <http://nosql-database.org> em Dez/2014.
- Espinosa, R., Garriga, L., Zubcoff, J. J., & Mazon, J.-N. (2014). Linked Open Data mining for democratization of big data. In *2014 IEEE International Conference on Big Data (Big Data)* (pp. 17–19). <http://doi.org/10.1109/BigData.2014.7004479>
- Gama, J. (2010). *Knowledge Discovery from Data Streams*. Retrieved from <http://www.crcpress.com/product/isbn/9781439826119>
- Girtelschmid, S., Steinbauer, M., Kumar, V., Fensel, A., & Kotsis, G. (2013). Big Data in Large Scale Intelligent Smart City Installations. In *Proceedings of International Conference on Information Integration and Web-based Applications; Services* (pp. 428:428–428:432). New York, NY, USA: ACM. <http://doi.org/10.1145/2539150.2539224>
- Gu, Y.-D., Cheng, J.-Z., & Wang, Z.-Y. (2011). An fuzzy forecasting algorithm for short term electricity loads based on partial clustering. In *2011 International Conference on Machine Learning and Cybernetics (ICMLC)* (Vol. 4, pp. 1560–1565). <http://doi.org/10.1109/ICMLC.2011.6017031>
- Han, J. (2011). *Data Mining: Concepts and Techniques* (3rd Revised edition edition). Waltham, Mass.: Morgan Kaufmann.

- Hedlund, J. (2013, December). The Smart City: Using IT to Make Cities More Livable. Retrieved from <http://smartcitiescouncil.com/resources/smart-city-using-it-make-cities-more-livable-0>
- Hofmann, H., Cook, D., Kielion, C., Schloerke, B., Hobbs, J., Loy, A., ... Yin, T. (2009). *Delayed, Cancelled, On-Time, Boarding ... Flying in the USA*. Data Expo 09. Retrieved from <http://stat-computing.org/dataexpo/2009/posters/>
- Hortonworks. (2014). Hortonworks Data Platform Guide. Obtido de <http://info.hortonworks.com/download-hdp-guide.html> em Fev/2015
- IBM. (2014, October 30). IBM - Smarter Cities - building and carrying out ways for a city to realize its full potential. Retrieved November 10, 2014, from http://www.ibm.com/smarterplanet/us/en/smarter_cities/overview/index.html
- Jara, A. J., Bocchi, Y., & Genoud, D. (2013). Determining Human Dynamics through the Internet of Things. In *2013 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)* (Vol. 3, pp. 109–113). <http://doi.org/10.1109/WI-IAT.2013.161>
- Kambatla, K., Kollias, G., Kumar, V., & Grama, A. (2014). Trends in big data analytics. *Journal of Parallel and Distributed Computing*, 74(7), 2561–2573. <http://doi.org/10.1016/j.jpdc.2014.01.003>
- Khan, Z., Anjum, A., & Kiani, S. L. (2013). Cloud Based Big Data Analytics for Smart Future Cities. In *2013 IEEE/ACM 6th International Conference on Utility and Cloud Computing (UCC)* (pp. 381–386). <http://doi.org/10.1109/UCC.2013.77>
- Kitchin, R. (2013). The real-time city? Big data and smart urbanism - Springer. <http://doi.org/10.1007/s10708-013-9516-8>
- Klein, A., Craun, C., & Lee, R. S. (2010). Airport delay prediction using weather-impacted traffic index (WITI) model. In *Digital Avionics Systems Conference (DASC), 2010 IEEE/AIAA 29th* (pp. 2.B.1–1–2.B.1–13). <http://doi.org/10.1109/DASC.2010.5655493>

- Krishnan, K. (2013). *Data Warehousing in the Age of Big Data* (1st ed.). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.
- Leavitt, N. (2010). Will NoSQL Databases Live Up to Their Promise? *Computer*, 43(2), 12–14.
<http://doi.org/10.1109/MC.2010.58>
- Lessig, L. (2007, Dezembro). The Annotated 8 Principles of Open Government Data. Obtido de <http://opengovdata.org> em Jan/2015.
- Liu, Z., Yang, P., & Zhang, L. (2013). A Sketch of Big Data Technologies. In *2013 Seventh International Conference on Internet Computing for Engineering and Science (ICICSE)* (pp. 26–29).
<http://doi.org/10.1109/ICICSE.2013.13>
- Lu, R., Zhu, H., Liu, X., Liu, J. K., & Shao, J. (2014). Toward efficient and privacy-preserving computing in big data era. *IEEE Network*, 28(4), 46–50. <http://doi.org/10.1109/MNET.2014.6863131>
- Mayilvaganan, M., & Sabitha, M. (2013). A cloud-based architecture for Big-Data analytics in smart grid: A proposal. In *2013 IEEE International Conference on Computational Intelligence and Computing Research (ICICIR)* (pp. 1–4). <http://doi.org/10.1109/ICICIR.2013.6724168>
- Mitton, N., Papavassiliou, S., Puliafito, A., & Trivedi, K. S. (2012). Combining Cloud and sensors in a smart city environment. *EURASIP Journal on Wireless Communications and Networking*, 2012(1), 1–10. <http://doi.org/10.1186/1687-1499-2012-247>
- Mooi, E., & Sarstedt, M. (2011). Cluster Analysis. In *A Concise Guide to Market Research* (pp. 237–284). Springer Berlin Heidelberg. Retrieved from http://link.springer.com/chapter/10.1007/978-3-642-12541-6_9
- Paryasto, M., Alamsyah, A., & Rahardjo, B., Kuspriyanto. (2014). Big-data security management issues. In *2014 2nd International Conference on Information and Communication Technology (ICoICT)* (pp. 59–63). <http://doi.org/10.1109/ICoICT.2014.6914040>

- Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2008). A Design Science Research Methodology for Information Systems Research. Retrieved from <http://sites.cgu.edu/chatterjees/files/2013/07/jmis-article.pdf>
- Pentaho. (2015). Pentaho Big Data Analytics. Obtido de <http://www.pentaho.com/product/big-data-analytics#discovery-and-visualizations> em Jan/2015.
- Philip Chen, C. L., & Zhang, C.-Y. (2014). Data-intensive applications, challenges, techniques and technologies: A survey on Big Data. *Information Sciences*, 275, 314–347. <http://doi.org/10.1016/j.ins.2014.01.015>
- Pike Research. (2011). *Smart Cities. Intelligent Information and Communications Technology Infrastructure in the Government, Buildings, Transport, and Utility Domains*. Retrieved from <http://www.navigantresearch.com/research/smart-cities>
- Pokorny, J. (2013). NoSQL databases: a step to database scalability in web environment. *International Journal of Web Information Systems*, 9(1), 69–82.
- RapidMiner. (2015). Predictive Analytics, Data Mining, Self-service, Open source. Retrieved February 7, 2015, from <https://rapidminer.com/>
- RDataMining. (2015). Introduction to Data Mining with R. Retrieved from <http://www.rdatamining.com/docs/introduction-to-data-mining-with-r>
- Rebollo, J. J., & Balakrishnan, H. (2014). Characterization and prediction of air traffic delays. *Transportation Research Part C: Emerging Technologies*, 44, 231–241. <http://doi.org/10.1016/j.trc.2014.04.007>
- RITA-BTS. (2015). Airline On-Time Performance Data. Obtido de http://www.transtats.bts.gov/Fields.asp?Table_ID=236 em Jun/2015
- Sagiroglu, S., & Sinanc, D. (2013). Big data: A review. In *2013 International Conference on Collaboration Technologies and Systems (CTS)* (pp. 42–47). <http://doi.org/10.1109/CTS.2013.6567202>

- Salvador, S., & Chan, P. (2004). Determining the number of clusters/segments in hierarchical clustering/segmentation algorithms. In *16th IEEE International Conference on Tools with Artificial Intelligence, 2004. ICTAI 2004* (pp. 576–584). <http://doi.org/10.1109/ICTAI.2004.50>
- Saragih, R. with Suakanto, S., & Supangkat, S. H., Suhardi. (2013). Smart city dashboard for integrating various data of sensor networks. In *2013 International Conference on ICT for Smart Society (ICISS)* (pp. 1–5). <http://doi.org/10.1109/ICTSS.2013.6588063>
- Schaffers, H., Komninos, N., Pallot, M., Trousse, B., Nilsson, M., & Oliveira, A. (2011). Smart Cities and the Future Internet: Towards Cooperation Frameworks for Open Innovation. In J. Domingue, A. Galis, A. Gavras, T. Zahariadis, D. Lambert, F. Cleary, ... M. Nilsson (Eds.), *The Future Internet* (pp. 431–446). Springer Berlin Heidelberg. Retrieved from http://link.springer.com/chapter/10.1007/978-3-642-20898-0_31
- Schneider, R. (2013, January 16). Hadoop for Dummies. Retrieved November 6, 2014, from <http://www-01.ibm.com/common/ssi/cgi-bin/ssialias?infotype=PM&subtype=BK&htmlfid=DCM03002USEN>
- Sedayao, J., Bhardwaj, R., & Gorade, N. (2014). Making Big Data, Privacy, and Anonymization Work Together in the Enterprise: Experiences and Issues. In *2014 IEEE International Congress on Big Data (BigData Congress)* (pp. 601–607). <http://doi.org/10.1109/BigData.Congress.2014.92>
- Serra, J. (2012, Janeiro 4). Difference between ETL and ELT. Obtido de <http://www.jamesserra.com/archive/2012/01/difference-between-etl-and-elt/> em Jan/2015.
- Simmhan, Y., Aman, S., Kumbhare, A., Liu, R., Stevens, S., Zhou, Q., & Prasanna, V. (2013). Cloud-Based Software Platform for Big Data Analytics in Smart Grids. *Computing in Science Engineering*, 15(4), 38–47. <http://doi.org/10.1109/MCSE.2013.39>
- Talend. (2015). Talend Products: Big Data Integration. Obtido de <http://www.talend.com/products/big-data> em Jan/2015.

- Tiwari, S. (2011). *Professional NoSQL*. John Wiley & Sons. Obtido de <https://books.google.pt/books?id=tv5iO9MnObUC> em Nov/2014
- Trends, G. (2014). Google Trends - Pesquisas do termo Big Data. Obtido de <https://www.google.pt/trends/explore> em Nov/2014
- Tudorica, B. G., & Bucur, C. (2011). A comparison between several NoSQL databases with comments and notes. In *Roedunet International Conference (RoEduNet), 2011 10th* (pp. 1–5). IEEE. Retrieved from http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5993686
- Vaishnavi, V., & Kuechler, B. (2013, October 23). Design Science Research in Information Systems. Retrieved from <http://desrist.org/desrist/content/design-science-research-in-information-systems.pdf>
- Vilajosana, I., Llosa, J., Martinez, B., Domingo-Prieto, M., Angles, A., & Vilajosana, X. (2013). Bootstrapping smart cities through a self-sustainable model based on big data flows. *IEEE Communications Magazine*, 51(6), 128–134. <http://doi.org/10.1109/MCOM.2013.6525605>
- Wagstaff, K., Cardie, C., Rogers, S., & Schroedl, S. (2001). Constrained K-means Clustering with Background Knowledge. In *In ICML* (pp. 577–584). Morgan Kaufmann.
- Wahbeh, A. H., Al-Radaideh, Q. A., Al-Kabi, M. N., & Al-Shawakfa, E. M. (2011). A comparison study between data mining tools over some classification methods. *IJACSA) International Journal of Advanced Computer Science and Applications*, 18–26.
- WEKA. (2015). Weka: Data Mining Software in Java. Retrieved from <http://www.cs.waikato.ac.nz/ml/weka/index.html>
- Wickham, C. (2009). *A tale of two airports: An exploration of flight traffic at SFO and OAK*. Data Expo 09. Retrieved from <http://stat-computing.org/dataexpo/2009/posters/>
- Wicklin, R., & Allison, R. (2009). *Congestion in the sky: Visualising domestic airline traffic with SAS*. Data Expo 09. Retrieved from <http://stat-computing.org/dataexpo/2009/posters/>

- Wigan, M. R., & Clarke, R. (2013). Big Data's Big Unintended Consequences. *Computer*, 46(6), 46-53.
<http://doi.org/10.1109/MC.2013.195>
- Xie, Y., Zheng, H., & Zhang, L.-Z. (2007). Electricity price forecasting by clustering-LSSVM. In *Power Engineering Conference, 2007. IPEC 2007. International* (pp. 697-702).
- Ye, F., Wang, Z., Zhou, F., Wang, Y., & Zhou, Y. (2013). Cloud-Based Big Data Mining and Analyzing Services Platform Integrating R. In *2013 International Conference on Advanced Cloud and Big Data (CBD)* (pp. 147-151). <http://doi.org/10.1109/CBD.2013.13>
- Zhang, J., Chen, Y., & Li, T. (2013). Opportunities of innovation under challenges of big data. In *2013 10th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)* (pp. 669-673).
<http://doi.org/10.1109/FSKD.2013.6816280>
- Zhou, H., Wu, X. H., & Li, X. G. (2011). An ANFIS model of electricity price forecasting based on subtractive clustering. In *2011 IEEE Power and Energy Society General Meeting* (pp. 1-5).
<http://doi.org/10.1109/PES.2011.6039228>
- Zikopoulos, P., & Eaton, C. (2011). *Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data* (1st ed.). McGraw-Hill Osborne Media.