



# Estudo e Avaliação de Plataformas de Business Intelligence em contexto empresarial

**JOAQUIM MANUEL OLIVEIRA LAPA**

Outubro de 2015

# **Estudo e Avaliação de Plataformas de *Business Intelligence* em contexto empresarial**

**Joaquim Manuel Oliveira Lapa**

**Dissertação para obtenção do Grau de Mestre em  
Engenharia Informática, Área de Especialização em  
Tecnologias da Decisão e Conhecimento**

**Orientador: Prof.<sup>a</sup> Ana Maria Neves de Almeida Baptista Figueiredo**

**Co-orientador: Prof. Jorge Fernandes Rodrigues Bernardino**

**Júri:**

Presidente:

[Nome do Presidente, Categoria, Escola]

Vogais:

[Nome do Vogal1, Categoria, Escola]

[Nome do Vogal2, Categoria, Escola] (até 4 vogais)

Porto, Outubro de 2015



# Dedicatória

À Sílvia, pela compreensão e apoio.

Às minhas filhas, Carolina e Sofia, que a tudo dão sentido.





# Resumo

O crescente interesse pela área de *Business Intelligence* (BI) tem origem no reconhecimento da sua importância pelas organizações, como poderoso aliado dos processos de tomada de decisão. O BI é um conceito dinâmico, que se amplia à medida que são integradas novas ferramentas, em resposta a necessidades emergentes dos mercados. O BI não constitui, ainda, uma realidade nas pequenas e médias empresas, sendo, até, desconhecido para muitas. São, essencialmente, as empresas de maior dimensão, com presença em diferentes mercados e/ou áreas de negócio mais abrangentes, que recorrem a estas soluções.

A implementação de ferramentas BI nas organizações depende, pois, das especificidades destas, sendo fundamental que a informação sobre as plataformas disponíveis e suas funcionalidades seja objetiva e inequívoca. Só uma escolha correta, que responda às necessidades da área de negócio desenvolvida, permitirá obter dados que resultem em ganhos, potenciando a vantagem competitiva empresarial. Com este propósito, efectua-se, na presente dissertação, uma análise comparativa das funcionalidades existentes em diversas ferramentas BI, que se pretende que venha auxiliar os processos de seleção da plataforma BI mais adaptada a cada organização e/ou negócio.

As plataformas BI enquadram-se em duas grandes vertentes, as que implicam custos de aquisição, de índole comercial, e as disponibilizadas de forma livre, ou em código aberto, designadas *open source*. Neste sentido, equaciona-se se estas últimas podem constituir uma opção válida para as empresas com recursos mais escassos.

Num primeiro momento, procede-se à implementação de tecnologias BI numa organização concreta, a operar na indústria de componentes automóveis, a Yazaki Saltano de Ovar Produtos Eléctricos, Ltd., implantada em Portugal há mais de 25 anos. Para esta empresa, o desenvolvimento de soluções com recurso a ferramentas BI afigura-se como um meio adequado de melhorar o acompanhamento aos seus indicadores de performance. Este processo concretizou-se a partir da *stack* tecnológica pré-existente na organização, a plataforma BI comercial da Microsoft.

Com o objetivo de, por um lado, reunir contributos que possibilitem elucidar as organizações na escolha da plataforma BI mais adequada e, por outro, compreender se as plataformas *open source* podem constituir uma alternativa credível às plataformas comerciais, procedeu-se a uma pesquisa comparativa das funcionalidades das várias plataformas BI *open source*. Em resultado desta análise, foram seleccionadas duas plataformas, a SpagoBI e a PentahoBI, utilizadas na verificação do potencial alternativo das *open source* face às plataformas comerciais. Com base nessas plataformas, reproduziu-se os processos e procedimentos desenvolvidos no âmbito do projeto de implementação BI realizado na empresa Yazaki Saltano.

**Palavras-chave:** *Business Intelligence*, Plataformas BI Comerciais, Plataformas BI *Open Source*, Ferramentas de BI

# Abstract

The growing interest in Business Intelligence (BI) is based on the recognition of its importance by organizations as a powerful supporter in the decision making process. BI is a dynamic concept that expands as you integrate new tools in response to markets needs. BI isn't yet a reality in small and medium enterprises and it's even unknown in many of them. It's mostly the larger companies, operating in different markets and/or business that use these solutions.

BI tools implementation depends on organizations specificities, and the information available about platforms and their functionalities must be objective and clear. Only a correct choice, in response to business needs, allows data that result in gains and competitive advantage. For this purpose, a comparative analysis of several BI tools was made, to help select the most suitable BI platform to an organization and/or business.

BI Platforms fall into two main areas, the ones involving acquisition costs, the commercial BI platforms, and those available in open code, the so called open source platforms. The latest may be a relevant option for small and medium enterprises, especially for companies with limited resources.

In a first moment, it occurs the implementation of BI technology in an automotive components industry, Yazaki Saltano de Ovar Produtos Eléctricos, Ltd., located in Portugal for over 25 years. For this company, the development of BI solutions is an appropriate way to improve monitoring of performance indicators. This process used the already existing technology stack at Yazaki, Microsoft commercial BI platform.

Then, after a comparative survey of several open source BI platforms, two of them were chosen to test their potential: SpagoBI and PentahoBI. With these platforms, some procedures from the BI implementation held at Yazaki Saltano have been reproduced. The purpose was to gather some contributions in order to help organizations to choose the most appropriate BI platform and verify whether the open source platforms can be a real alternative to commercial ones

**Keywords:** Business Intelligence, Commercial BI Platforms, Open Source BI Platforms, BI tools



# Agradecimentos

A concretização da presente dissertação foi possível em resultado do contributo e apoio de várias pessoas. A todas dirijo o meu agradecimento e sincero reconhecimento.

Endereço um especial agradecimento à Yazaki Saltano de Ovar, em particular ao Eng. Miguel Miranda, Dr. Rúben Pinto e a Eng.<sup>a</sup> Rosa Rocha pela valiosa oportunidade concedida, por toda a informação disponibilizada e pela total recetividade, cooperação e interesse manifestados.

Agradeço aos meus orientadores, Professora Doutora Ana Almeida e Professor Doutor Jorge Bernardino, o conhecimento, a partilha e disponibilidade.

Uma palavra de apreço e estima à minha família, por todo o incentivo.



# Índice

<b>Dedicatória</b>	<b>iii</b>
<b>Resumo</b>	<b>v</b>
<b>Abstract</b>	<b>vii</b>
<b>Agradecimentos</b>	<b>ix</b>
<b>Índice</b>	<b>xi</b>
<b>Lista de Figuras</b>	<b>xv</b>
<b>Lista de Tabelas</b>	<b>xix</b>
<b>Acrónimos</b>	<b>xxi</b>
<b>1 INTRODUÇÃO</b>	<b>1</b>
1.1 Motivação	1
1.2 Objetivos	2
1.3 Principais Contributos	2
1.4 Estrutura da dissertação	2
<b>2 BUSINESS INTELLIGENCE</b>	<b>5</b>
2.1 Conceitos	6
2.1.1 Extract, Transform and Load	6
2.1.2 Data Warehouse	7
2.1.3 On-Line Analytical Processing	12
2.1.4 Data mining	14
2.2 <i>Self-Service</i> BI	15
2.3 Aplicabilidade	15
<b>3 PLATAFORMAS DE BUSINESS INTELLIGENCE</b>	<b>17</b>
3.1 Plataformas de BI <i>Open Source</i>	18
3.1.1 Actuate	18
3.1.2 JaspersoftBI	19
3.1.3 OpenI	20
3.1.4 PALO BI Suite	21



3.1.5	PentahoBI	22
3.1.6	SpagoBI	23
3.1.7	Vanilla	24
<b>3.2</b>	<b>Plataformas de BI Comerciais</b>	<b>25</b>
3.2.1	beMemo	25
3.2.2	IBM Cognos	26
3.2.3	Microsoft BI	27
3.2.4	MicroStrategy	28
3.2.5	Oracle BI	29
3.2.6	SAP BI	30
3.2.7	SAS Business Intelligence & Analytics	31
3.2.8	SOL7 BIMachine	32
<b>3.3</b>	<b>Comparação das Plataformas BI</b>	<b>32</b>
3.3.1	Critérios de Comparação	32
3.3.2	Análise comparativa	36
3.3.3	Resumo	38
<b>4</b>	<b>CASO DE ESTUDO</b>	<b>39</b>
<b>4.1</b>	<b>Caraterização da Empresa</b>	<b>39</b>
4.1.1	Missão e Visão	40
4.1.2	Valores	40
<b>4.2</b>	<b>Descrição do Problema</b>	<b>40</b>
<b>4.3</b>	<b>Identificação do Processo de Negócio, Granularidade, Dimensões e Factos</b>	<b>45</b>
4.3.1	Processo de Negócio	45
4.3.2	Granularidade	45
4.3.3	Dimensões	45
4.3.4	Factos	48
<b>4.4</b>	<b>Modelo Dimensional</b>	<b>49</b>
<b>4.5</b>	<b>Dados Adicionais</b>	<b>49</b>
<b>4.6</b>	<b>Descrição Técnica</b>	<b>50</b>
4.6.1	Integration Services Project	53
4.6.2	Analysis Services Project	65
4.6.3	Report Server Project	69
4.6.4	PowerBI – Excel 2013	72
<b>4.7</b>	<b>Síntese Conclusiva</b>	<b>87</b>
<b>5</b>	<b>AVALIAÇÃO EXPERIMENTAL DO SPAGOB I E PENTAHOB I</b>	<b>89</b>
<b>5.1</b>	<b>Instalação e configuração do servidor</b>	<b>90</b>

5.1.1	SpagoBI	91
5.1.2	PentahoBI	92
5.1.3	Análise	93
<b>5.2</b>	<b>Processo de ETL</b>	<b>93</b>
5.2.1	Talend Open Studio	94
5.2.2	Pentaho Data Integration - KETTLE	95
5.2.3	Análise	97
<b>5.3</b>	<b>Análises OLAP</b>	<b>97</b>
5.3.1	SpagoBI - Mondrian	97
5.3.2	PentahoBI – Saiku Analytics	99
5.3.3	Análise	100
<b>5.4</b>	<b>Análise comparativa entre PentahoBI e SpagoBI</b>	<b>101</b>
<b>6</b>	<b>CONCLUSÃO E TRABALHO FUTURO</b>	<b>103</b>
6.1	Limitações	105
6.2	Publicações	105
6.3	Trabalho futuro	106
	<b>REFERÊNCIAS</b>	<b>107</b>
	<b>ANEXOS</b>	<b>111</b>
	Anexo I - Store Procedure	111
	Anexo II - Script SQL	115



# Lista de Figuras

Figura 1 – Modelo de Kimball .....	7
Figura 2 – Esquema em Estrela .....	8
Figura 3 – Esquema em Floco de Neve .....	8
Figura 4 – Esquema em Constelação .....	9
Figura 5 – Modelo de Inmon .....	10
Figura 6 – Modelo Híbrido .....	11
Figura 7 – Cubo Dimensional .....	13
Figura 8 – BIRT <i>Product Line</i> (fonte: <a href="http://infodecisionnel.com/">http://infodecisionnel.com/</a> ) .....	19
Figura 9 – Arquitetura da Plataforma JasperSoft BI (fonte: <a href="http://www.dewcis.com">http://www.dewcis.com</a> ) .....	20
Figura 10 – Arquitetura OpenI (fonte: <a href="http://openi.sourceforge.net/openi_product.html">http://openi.sourceforge.net/openi_product.html</a> ) .....	21
Figura 11 – Arquitetura da Jedox Palo BI Suite (fonte: <a href="http://www.legrandbi.com">http://www.legrandbi.com</a> ) .....	22
Figura 12 – Arquitetura da Pentaho <i>Open BI Suite</i> (fonte: <a href="http://helicaltech.com">http://helicaltech.com</a> ) .....	23
Figura 13 – SpagoBI - Plataforma de Integração (fonte: <a href="http://www.stratebi.com/pt/spagobi">http://www.stratebi.com/pt/spagobi</a> ) .....	24
Figura 14 – <i>Vanilla Functional Schema</i> (fonte: <a href="http://www.bpm-conseil.com">http://www.bpm-conseil.com</a> ) .....	25
Figura 15 – Visão geral do beMemo (fonte: <a href="https://www.bememo.com">https://www.bememo.com</a> ) .....	26
Figura 16 – IBM Cognos 10.2 (fonte: <a href="http://pmsquare.com">http://pmsquare.com</a> ) .....	27
Figura 17 – Microsoft BI <i>Stack</i> (fonte: <a href="http://www.microsoft.com">http://www.microsoft.com</a> ) .....	28
Figura 18 – Arquitetura do MicroStrategy 9 (fonte: <a href="http://www.microstrategy.pt">http://www.microstrategy.pt</a> ) .....	28
Figura 19 – <i>Business Intelligence Foundation Suite</i> (fonte: <a href="https://obibb.wordpress.com">https://obibb.wordpress.com</a> ) .....	30
Figura 20 – Arquitetura SAP BW (fonte: <a href="http://help.sap.com">http://help.sap.com</a> ) .....	31
Figura 21 – SAS <i>Business Solutions framework</i> (fonte: <a href="http://sas.com">http://sas.com</a> ) .....	32
Figura 22 – Plataforma de <i>Business Intelligence</i> SOL7-BI (fonte: <a href="http://www.sol7.com.br">http://www.sol7.com.br</a> ) .....	32
Figura 23 – <i>Building on Early Successes in UBM's Social Business Strategy</i> – Sep. 25, 2013 .....	33
Figura 24 – Diagrama Entidade-Relação simplificado da base de dados MHT .....	41
Figura 25 – ER expandido da base de dados <i>WheTool</i> .....	42
Figura 26 – Exemplo de relatório para <i>Manufacturing Design</i> .....	44
Figura 27 – Esquema do modelo dimensional .....	49
Figura 28 – Arquitetura Microsoft <i>Self-Service BI</i> .....	51
Figura 29 – <i>Master Execution Package</i> .....	54
Figura 30 – <i>Control Flow</i> da Base de <i>Dados</i> MHT .....	54
Figura 31 – Detalhe do Componente <i>Data Flow Task</i> .....	55
Figura 32 – Visão Geral do <i>Package CreateInsertUpdate_Datamart</i> .....	56
Figura 33 – <i>Scripts SQL</i> do DM .....	57
Figura 34 – <i>Precedence Constraint Editor</i> .....	58
Figura 35 – Comando SQL na dimensão <i>CarMaker</i> .....	59
Figura 36 – Dimensões <i>Person, Manager e TeamLeader</i> .....	59
Figura 37 – Transformações no fluxo da Dimensão <i>Person</i> .....	60
Figura 38 – Componente <i>Load TeamLeader</i> .....	61
Figura 39 – Carregamento da Tabela <i>DimTeamLeader</i> .....	62

Figura 40 – Cálculos, transformações e inicialização das métricas.....	62
Figura 41 – Detalhes do componente QtyCircuits .....	63
Figura 42 – Visão geral do fluxo da tabela de Factos .....	64
Figura 43 – Variáveis utilizadas no projeto .....	65
Figura 44 – Campos do cubo de dados .....	66
Figura 45 – SSAS – Análise 1 .....	66
Figura 46 – SSAS – Análise 2 .....	67
Figura 47 – SSAS – Análise 3 .....	67
Figura 48 – SSAS – Análise 4 .....	67
Figura 49 – SSAS – Análise 5 .....	68
Figura 50 – SSAS – Análise 6 .....	68
Figura 51 – SSAS – Análise 7 .....	69
Figura 52 – SSRS – Tabela <i>Matrix</i> .....	70
Figura 53 – SSRS – Gráfico .....	70
Figura 54 – SSRS – Propriedades do projeto .....	71
Figura 55 – SSRS – Relatório <i>on-line</i> .....	72
Figura 56 – Modelo de dados no <i>Power Pivot</i> .....	73
Figura 57 – <i>Pivot Tables and Pivot Chart</i> – Relatório 1 .....	75
Figura 58 – Tabela de apoio ao relatório 1 .....	76
Figura 59 – <i>Pivot Tables and Pivot Chart</i> – Relatório 2 .....	76
Figura 60 – <i>Pivot Tables and Pivot Chart</i> – Relatório 3 .....	77
Figura 61 – Detalhe Relatório 3 .....	78
Figura 62 – <i>Pivot Tables and Pivot Chart</i> – Relatório 4 .....	78
Figura 63 – Regras do relatório 4 .....	79
Figura 64 – <i>Pivot Tables and Pivot Chart</i> - Relatório 5 .....	79
Figura 65 – <i>Target's</i> do relatório 5 .....	80
Figura 66 – <i>Pivot Tables and Pivot Chart</i> – Relatório 6 .....	80
Figura 67 – <i>Power View</i> – <i>Dashboard</i> Total de Tarefas .....	81
Figura 68 – Detalhe do <i>Dashboard</i> Total de Tarefas .....	81
Figura 69 – <i>Slice and Dice</i> – <i>Dashboard</i> Total de Tarefas .....	82
Figura 70 – Detalhes do <i>Dashboard</i> Total de Tarefas .....	83
Figura 71 – <i>Power View</i> – <i>Dashboard</i> Geral .....	84
Figura 72 – Detalhe 1 – <i>Dashboard</i> Geral .....	84
Figura 73 – Detalhe 2 – <i>Dashboard</i> Geral .....	84
Figura 75 – Detalhe 4 – <i>Dashboard</i> Geral .....	85
Figura 76 – <i>Power View</i> – <i>Dashboard</i> Dispersão das tarefas .....	86
Figura 77 – <i>Power View</i> – <i>Dashboard</i> 4 .....	86
Figura 78 – <i>Power View</i> – Detalhe <i>Dashboard</i> 4 .....	87
Figura 79 – SpagoBI – Ecrã de <i>login</i> e ambiente de trabalho inicial .....	91
Figura 80 – Gráficos, <i>Dashboards</i> , KPI's e OLAP .....	91
Figura 81 – PentahoBI – Ecrã de <i>login</i> e ambiente de trabalho inicial .....	92
Figura 82 – PentahoBI – Navegação, OLAP, Graficos e KPI's .....	93
Figura 83 – Talend – Carregamento das dimensões temporais .....	94

Figura 84 – Talend – Tempo de execução do carregamento das dimensões temporais.....	94
Figura 85 – Talend – Carregamento de dimensões .....	95
Figura 86 – KETTLE – Carregamento das dimensões temporais.....	96
Figura 87 – Kettle – Tempo de execução do carregamento das dimensões temporais .....	96
Figura 88 – Kettle – Fluxo de ETL .....	96
Figura 89 – <i>SpagoBIStudio – Deploy OLAP</i> .....	97
Figura 90 – Mondrian – Análise 1 .....	98
Figura 91 – Mondrian – Análise 2 .....	98
Figura 92 – Mondrian – Análise 3 .....	98
Figura 93 – Mondrian – Análise 4 .....	99
Figura 94 – <i>Saiku Analytics – Análise 1</i> .....	99
Figura 95 – <i>Saiku Analytics – Análise 2</i> .....	100



# Lista de Tabelas

Tabela 1 – <i>Data Warehouse</i> vs Base de Dados Operacional.....	11
Tabela 2 – Registo inicial na tabela de dimensão .....	12
Tabela 3 – Alteração SCD do Tipo 1 .....	12
Tabela 4 – Alteração SCD do Tipo 2 .....	12
Tabela 5 – Alteração SCD do Tipo 3 .....	12
Tabela 6 – Aplicações funcionais de BI .....	16
Tabela 7 – Comparação de funcionalidades .....	37
Tabela 8 – Detalhes das tabelas principais .....	42
Tabela 9 – Dimensão <i>Activity</i> .....	45
Tabela 10 – Dimensão <i>CarMaker</i> .....	46
Tabela 11 – Dimensão <i>Complexity</i> .....	46
Tabela 12 – Dimensão <i>Date</i> .....	46
Tabela 13 – Dimensão <i>Manager</i> .....	46
Tabela 14 – Dimensão <i>Person</i> .....	47
Tabela 15 – Dimensão <i>Phase</i> .....	47
Tabela 16 – Dimensão <i>Report</i> .....	47
Tabela 17 – Dimensão <i>TeamLeader</i> .....	48
Tabela 18 – Dimensão <i>TypeTask</i> .....	48
Tabela 19 – Campos da tabela de Factos .....	48
Tabela 20 – Detalhes da hierarquia .....	50
Tabela 21 – SSRS – Fórmulas utilizadas no relatório.....	70
Tabela 22 – Cálculos, Mediadas/Métricas e <i>Time Intelligence</i> .....	74
Tabela 23 – Tabela comparativa PentahoBI vs SpagoBI .....	102





# Acrónimos

<b>API</b>	<i>Application Programming Interface</i>
<b>BD</b>	Base de Dados
<b>BI</b>	<i>Business Intelligence</i>
<b>BIRT</b>	<i>Business Intelligence and Reporting Tools</i>
<b>CIF</b>	<i>Corporate Information Factory</i>
<b>DAX</b>	<i>Data Analysis Expression</i>
<b>DCBD</b>	Descoberta de Conhecimento em Base de Dados
<b>DM</b>	<i>Data Mart</i>
<b>DQP</b>	<i>Data Quality Poor</i>
<b>DSA</b>	<i>Data Staging Area</i>
<b>DW</b>	<i>Data Warehouse</i>
<b>ETL</b>	<i>Extract, Transform and Load</i>
<b>GNU</b>	<i>General Public License</i>
<b>GWT</b>	<i>Google Web Toolkit</i>
<b>JDK</b>	<i>Java Development Kit</i>
<b>JRE</b>	<i>Java Runtime Environment</i>
<b>JVM</b>	<i>Java Virtual Machine</i>
<b>KETTLE</b>	<i>Kettle Extraction, Transformation, Transportation, and Load Environment</i>
<b>KPI's</b>	<i>Key Performance Indicators</i>
<b>LGPL</b>	<i>Lesser General Public Licence</i>
<b>MDX</b>	<i>MultiDimensional Expressions</i>
<b>MPL</b>	<i>Mozilla Public License</i>
<b>OLAP</b>	<i>On-Line Analytic Processing</i>
<b>OLTP</b>	<i>On-Line Transaction Processing</i>

<b>PMEs</b>	Pequenas e Médias Empresas
<b>SaaS</b>	<i>Software as a Service</i>
<b>SCD</b>	<i>Slowly Changing Dimensions</i>
<b>SQL</b>	<i>Structured Query Language</i>
<b>SSAS</b>	<i>SQL Server Analysis Services</i>
<b>SSBI</b>	<i>Self-Service Business Intelligence</i>
<b>SSIS</b>	<i>SQL Server Integration Services</i>
<b>SSRS</b>	<i>SQL Server Reporting Services</i>
<b>TI</b>	Tecnologias de Informação
<b>YTD</b>	<i>Year To Date</i>

# 1 Introdução

Todo o contexto organizacional necessita de um processo de planeamento rigoroso, que uniformize procedimentos e optimize a globalidade dos recursos existentes. Estes pressupostos exigem aceder, de forma constante e regular, a dados relevantes e sempre atualizados, que possibilitem decidir em ordem à sustentabilidade e crescimento das organizações.

O conceito *Business Intelligence* (BI) aponta para um conjunto de metodologias de gestão, implementado através de ferramentas de *software*, cuja função é conferir rentabilidade e vantagem aos processos de decisão e administração das organizações. Desta forma, os decisores poderão dispor, num dado local e momento, de ferramentas com capacidade analítica, que possibilitam, de forma agrupada, importantes informações e dados.

O processo de seleção das ferramentas BI mais adequadas a uma determinada organização vai depender de diversos fatores, desde logo da capacidade financeira, dos recursos humanos e materiais existentes, mas essencialmente da visão de crescimento dos seus líderes.

## 1.1 Motivação

O universo das Pequenas e Médias Empresas (PME's) revela, ainda, algum desconhecimento relativamente ao BI e, conseqüentemente, do potencial e mais-valia que pode conferir ao seu desenvolvimento. Entende-se ser fundamental promover uma adequada divulgação/informação sobre as suas ferramentas, com especial enfoque nos benefícios associados. Neste âmbito, o contexto académico, através dos seus vários agentes, pode desempenhar junto das PME's um papel ativo.

Muitas destas empresas não dispõem de estratégias concretas na área das Tecnologias de Informação (TI), apresentando uma realidade incipiente, pouco profissionalizada e de reduzida rentabilidade. Vários são os motivos: ausência de informação, indisponibilidade financeira e/ou processos de gestão pouco inovadores. Contrariar o desconhecimento do potencial do BI e das

suas vantagens competitivas para o posicionamento das empresas no mercado constitui a principal motivação da presente dissertação.

## 1.2 Objetivos

Como finalidade principal, pretende-se reunir informação sobre as plataformas BI que contribua para uma escolha adequada das organizações face às diversas opções existentes. Entende-se que constitui uma vantagem crítica encontrar todas as ferramentas BI num único ambiente gráfico e que tal pode reduzir o tempo de formação ao utilizador final. Este é um aspeto importante na implementação destes projetos, sobretudo quando podem implicar custos.

É fundamental que as empresas, sobretudo as PME's, percebam o investimento em BI como sinónimo de novas oportunidades de negócio. Assim, é de grande relevância para estas empresas a possibilidade de recurso às ferramentas *open source*. Neste sentido, esta dissertação visa determinar se as plataformas *open source* podem constituir uma alternativa fiável às plataformas comerciais; pelo que se propõe efetuar uma pesquisa comparativa das funcionalidades das várias plataformas BI *open source* e testar as suas potencialidades por referência aos resultados obtidos com a operacionalização de uma plataforma BI comercial em contexto organizacional real.

## 1.3 Principais Contributos

Constituem os principais contributos desta dissertação:

- Reunir informação essencial sobre as plataformas e ferramentas BI, que faculte às organizações maior conhecimento sobre as mesmas, potenciando uma melhor seleção em situação de implementação;
- Redigir artigos científicos de divulgação da informação acima reunida;
- Implementar, de raiz, tecnologias BI em ambiente real, numa unidade industrial;
- Avaliar o desempenho de plataformas BI *open source*, enquanto alternativas credíveis às comerciais.

## 1.4 Estrutura da dissertação

Esta dissertação compreende seis capítulos. No capítulo 2 descreve-se e analisa-se, nas suas várias componentes, o conceito *Business Intelligence*, bem como a sua base teórica, evolução e tendências.

No capítulo 3 apresenta-se as plataformas de BI, relacionando-se o conceito *Business Intelligence* com as plataformas comerciais e *open source*. Descreve-se as diferentes plataformas consideradas, sua origem, arquitectura e funcionalidades. Este capítulo

compreende, ainda, a análise comparativa efetuada a essas plataformas e respetivas conclusões.

O capítulo 4 é relativo ao caso de estudo. Apresenta e caracteriza o contexto de enquadramento, a empresa Yazaki Saltano de Ovar Produtos Eléctricos, Ltd., descreve o problema identificado nessa organização, detalha toda a intervenção técnica desenvolvida com recurso a ferramentas BI e apresenta os resultados obtidos.

O capítulo 5 refere a avaliação experimental de duas plataformas *open source*, apresentando-se os critérios inerentes à análise comparativa efetuada às plataformas e as conclusões sugeridas pela operacionalização das mesmas, em contraponto a uma plataforma BI comercial.

O capítulo 6 enuncia as principais considerações e conclusões finais, por referência aos objetivos previamente definidos, bem como as propostas de trabalho futuro.

Relativamente aos anexos que integram a dissertação, estes compreendem os dois *scripts* em Transact-SQL, relativos à unidade industrial em que se implementou o projecto BI.



## 2 Business Intelligence

*Business Intelligence* pode ser descrito como um sistema que conjuga, com recurso a ferramentas analíticas, recolha de dados, seu armazenamento e gestão de conhecimento, proporcionando à tomada de decisão informação complexa e competitiva [1]. Trata-se de um sistema que fornece aos decisores, no momento e local certos, informação operacional e pertinente [2].

A génese do conceito BI é imputada a Howard Dresner, quando, por volta de 1989, integrava o Instituto Gartner. Inicialmente, o conceito surgiu mais associado à gestão organizacional do que à área tecnológica e remetia para o conjunto de modelos e métodos que promovia a tomada de decisão, com recurso a sistemas de apoio sustentados em dados [3, 4].

*Business Intelligence* engloba um conjunto de metodologias de gestão, implementado através de ferramentas de software, cuja função é proporcionar eficácia às organizações, potenciando a sua competitividade e liderança nos mercados. BI é um contributo importante para a produção e gestão do conhecimento e, conseqüentemente, promove a melhoria da performance organizacional. De acordo com William R. King, a gestão do conhecimento concentra-se nos processos de criação de conhecimento, aquisição, refinamento, armazenamento, transferência, partilha e utilização. Estes, apoiam processos organizacionais envolvendo inovação, aprendizagem individual, aprendizagem coletiva e tomada de decisão colaborativa. Os resultados intermédios da gestão do conhecimento melhoram comportamentos organizacionais, decisões, produtos, serviços, processos e relações que permitem à organização melhorar o seu desempenho global [5].

Atualmente, a globalidade das organizações, sobretudo as de maior dimensão, usufrui de sistemas de Tecnologias da Informação (TI) que possibilitam recolher, processar, transmitir e difundir dados. Os sistemas de TI são desenhados a fim de facultarem o registo de amplas quantidades de dados gerados no processo de negócio.



Como resultado da operacionalização das ferramentas BI, todos estes dados adquirem estrutura e organização, oferecendo informação essencial e fundamental ao processo de tomada de decisão. Este encontra suporte nas evidências geradas pelas referidas ferramentas, designadamente, tabelas, gráficos, *dashboards*, indicadores de performance, processamento analítico de dados, exploração de dados, entre outros. Das leituras interpretativas destes elementos, os decisores podem perspetivar e depreender tendências, desvios ou padrões na respetiva área de negócio. Tal promoverá o fortalecimento do planeamento estratégico das organizações, bem como uma adequada gestão das relações contratuais com os diferentes parceiros, resultando num crescimento efetivo do negócio. Este crescimento não pode alhear-se da dinâmica das redes sociais, fonte de informação sobre os comportamentos dos mercados e seus agentes [6, 7, 8]. A conversão destas informações em conhecimento organizacional, através da integração das tecnologias colaborativas no BI, revela-se, pois, em nosso entendimento, imprescindível à sustentabilidade das organizações.

A escolha correta da plataforma BI é um aspeto fundamental para garantir o sucesso da sua aplicabilidade. A taxa de insucesso de projetos BI situa-se entre 70 e 80% [9] e deve-se ao facto da implementação de um sistema BI não ser idêntico à de um projeto de TI [10]. Foram identificadas três dimensões como fatores-chave para projetos BI bem-sucedidos: a dimensão organizacional, a dimensão do processo e a dimensão tecnológica. Para este trabalho, e numa perspectiva orientada ao negócio elege-se a análise da dimensão tecnológica, que contempla a escalabilidade e flexibilidade da *framework* técnica, sustentabilidade, qualidade e integridade de dados [11]. Desta forma, procede-se à avaliação e análise comparativa das características das plataformas BI, assim como das funcionalidades existentes nas plataformas comerciais e *open source*.

## 2.1 Conceitos

O recurso a sistemas BI impõe um conhecimento dos principais conceitos associados: *Extract, Transform and Load* (ETL), *Data Warehouse* (DW), *On-Line Analytical Processing* (OLAP) e *Data Mining* [12].

### 2.1.1 Extract, Transform and Load

A recolha de dados é um processo crítico nos projetos BI. Sem uma base de dados fiável, não será possível usufruir do potencial das ferramentas BI. O referido processo é realizado com o recurso a ferramentas de extração, transformação, limpeza e carregamento de dados - ETL. Este permite extrair elementos de diversas fontes: bases de dados operacionais (*On-line Transaction Processing* - OLTP), ficheiros XML, folhas de cálculo, redes sociais, entre outras. Uma vez modificados, os dados são inseridos numa base usualmente designada por DW. Em momento anterior, a arquitetura dos projectos de ETL contempla uma área de armazenamento de dados temporária, onde estes sofrem as alterações necessárias para posterior carregamento na DW. Trata-se da *Data Staging Area* (DSA).

## 2.1.2 Data Warehouse

O termo *Data Warehouse* surgiu em 1992 [13], entendido como um grande armazém de dados, um arquivo único de dados consistente e normalizado. É considerado a base de trabalho de todas as ferramentas BI. A sua arquitetura não é unânime, existindo diferentes abordagens à sua criação e manutenção.

Bill Inmon [14] define DW como sendo um conjunto de dados não volátil, orientado por assunto, integrado e temporalmente catalogado, que dá suporte ao processo de tomada de decisão. No seguimento desta definição, surge o conceito de *Data Mart* (DM), como um subconjunto de dados delimitado a um determinado assunto ou área, extraído da DW.

Kimball [13] discorda desta definição e defende um movimento inverso, que batizou de *Data Warehouse Bus Architecture*. Neste âmbito, os DM's devem ser orientados a dados ou a fontes de dados. O mais viável para as empresas é desenvolver vários DM's consolidados, sendo posteriormente integrados e que partilham dimensões comuns. Contrariamente a um DW centralizado, complexo e previamente definido, Kimball preconiza um conjunto sequencial de DM's independentes e consistentes, que pode gerar valor para o negócio desde o primeiro DM [15]. Este autor define os seguintes objetivos e características do processo de criação das DW [16]:

- Tornar a informação da organização rapidamente acessível, compreensível e navegável;
- Tornar a informação da organização consistente e comparável entre diferentes unidades organizacionais;
- Constituir uma fonte de informação adaptável e fiável;
- Garantir segurança no acesso à informação;
- Constituir a fundação dos processos de tomada de decisão.

Na Figura 1 é possível visualizar o fluxo de dados do modelo de Kimball.

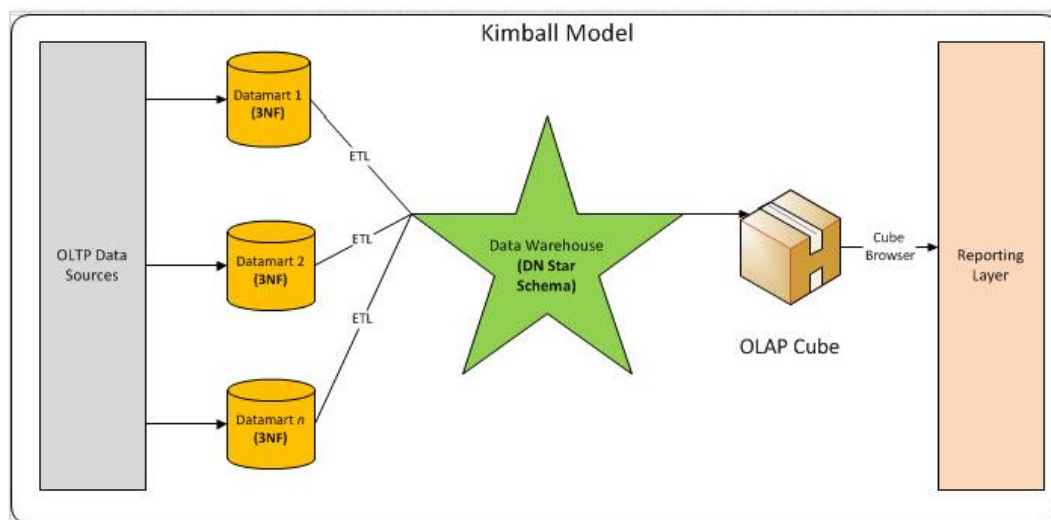


Figura 1 – Modelo de Kimball

(fonte: <https://bennyaustin.wordpress.com/2010/05/02/kimball-and-inmon-dw-models>,  
acedido em 2015)

O Modelo de kimball é também designado por Esquema em Estrela, que constitui a forma mais comum, pensada para possibilitar a maior eficiência possível nas consultas. Por norma, a sua configuração contempla uma tabela de factos no centro, rodeada por diversas dimensões. A ligação entre essa tabela e as dimensões é assegurada por chaves primárias (figura 2).

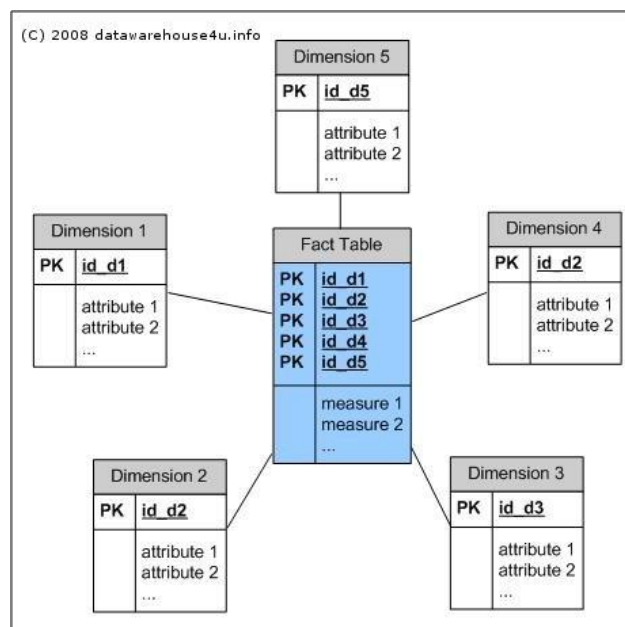


Figura 2 – Esquema em Estrela

Deste Esquema em Estrela, podem suceder duas derivações, o Floco de Neve e a Constelação. O primeiro compreende uma normalização das dimensões, tornando o esquema mais complexo e evitando a redundância excessiva de dados. Uma desvantagem deste esquema, decorrente da referida normalização, é o menor desempenho na resposta a questões (figura 3).

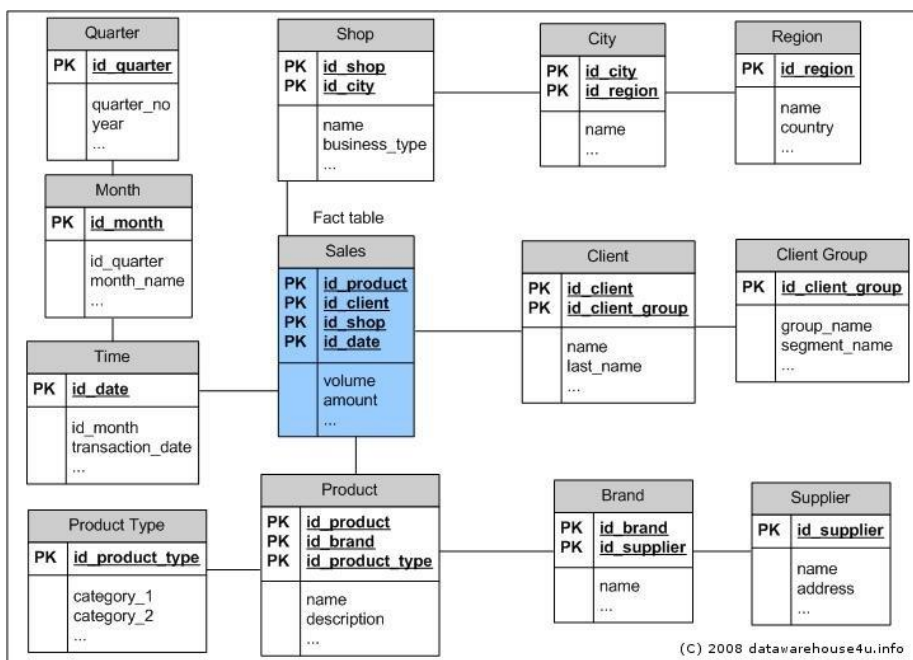


Figura 3 – Esquema em Floco de Neve

A Constelação apresenta diversas tabelas de factos que partilham as mesmas dimensões. Pode assumir, simultaneamente, a forma de um ou mais esquemas de estrela e floco de neve, tornando-se numa constelação (figura 4).

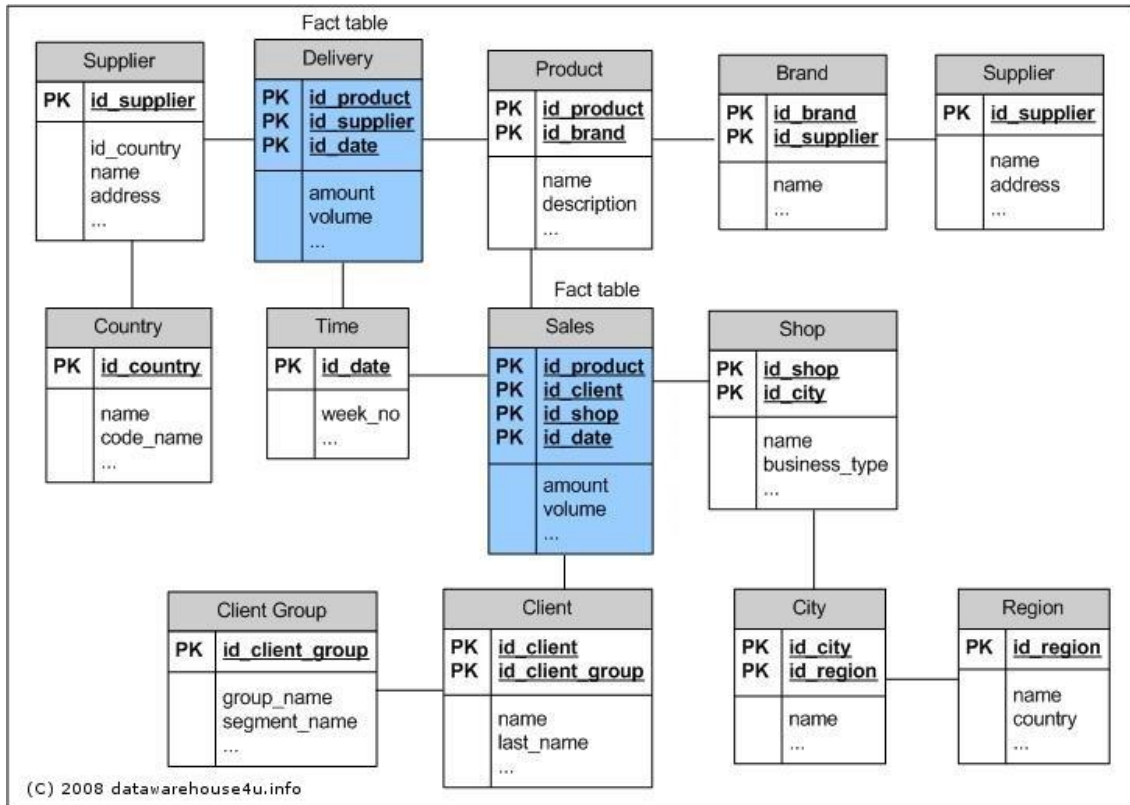


Figura 4 – Esquema em Constelação

Por oposição à otimização do processamento de atualizações, como se verifica no modelo relacional de base de dados, o desenho da DW/DM baseia-se na modelação multidimensional cujos objetivos são criar uma base de dados de fácil compreensão e utilização e otimizar o desempenho no processamento de questões [17].

No modelo multidimensional estão presentes dois tipos de tabelas, a de factos e a de dimensões. A tabela de factos integra chaves estrangeiras que a ligam às dimensões com que se relaciona. Estão também presentes atributos numéricos (métricas ou medidas), correspondentes a factos. Estes permitem registar acontecimentos que posteriormente podem ser analisados com recurso a ferramentas de exploração de cubos de dados. Os factos podem ser aditivos, quando permitem a agregação por todas as dimensões; semi-aditivos, sempre que permitem essa agregação, havendo, contudo, pelo menos uma dimensão onde não faz sentido a agregação; e não-aditivos, que não permitem qualquer agregação de dados.

As tabelas de dimensões apresentam dados não normalizados, frequentemente com grande número de atributos, e um número reduzido de registos quando comparadas com a tabela de factos. As tabelas de dimensões permitem responder a questões colocados pelo utilizador final, sendo as mais comuns: quem, quando, onde, porquê e como (quem comprou, quando, onde e como).

A implementação real de um projeto BI apresentada no capítulo 4 utiliza o enquadramento teórico de Kimball, cujo modelo compreende quatro passos:

- Identificação dos processos de negócio: Selecionar e conhecer os requisitos dos processos de negócio, bem como os dados disponíveis;
- Granularidade: Escolher o detalhe dos dados contidos na DW. O nível mais baixo de granularidade é designado por atómico, que não pode ser alvo de mais decomposições. Este nível de granularidade permite ao utilizador agregar os dados como pretendido. Um nível intermédio de granularidade implica o risco de não ser possível satisfazer todas as pesquisas solicitadas pelos utilizadores. Por granularidade de um facto entende-se a unidade de medida de um indicador de desempenho;
- Dimensões: Determinar todas as entidades que servem de perspetivas de análise;
- Factos: Definir as métricas a serem incluídas na tabela de factos.

Contrariamente à abordagem *Bottom Up* de Kimball, Inmon propõe uma abordagem *Top Down*. Nesta, primeiro dever-se-á construir uma DW, modelando-se toda a empresa para se chegar a um único modelo corporativo. Posteriormente, procede-se à construção, por assuntos ou departamentos, dos DM's. Inmon defende que o ponto de partida seria os *Corporate Information Factory* - CIF, uma infra-estrutura ideal para incorporar os dados da empresa. A figura 5 exhibe este modelo.

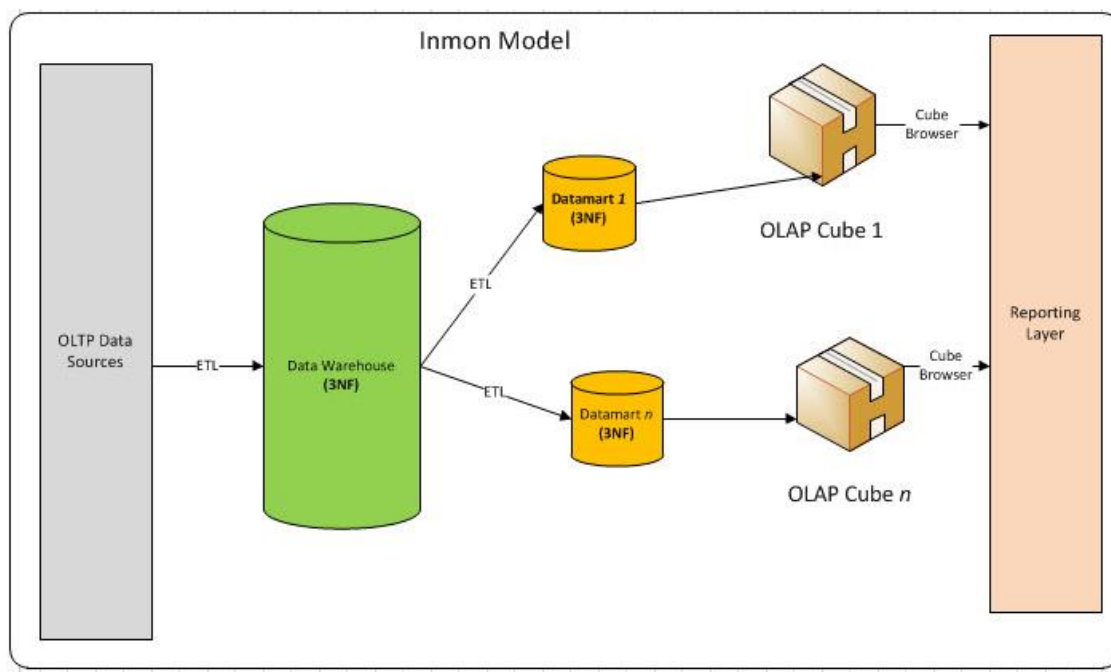


Figura 5 – Modelo de Inmon

(fonte: <https://bennyaustrin.wordpress.com/2010/05/02/kimball-and-inmon-dw-models>,  
acedido em 2015)

Em contextos reais, verificou-se o surgimento de uma terceira abordagem, uma solução híbrida, que junta os dois modelos. A criação de uma DW normalizada a partir do método CIF, seguindo-se a criação de *data marts* como preconizado por Kimball (figura 6).

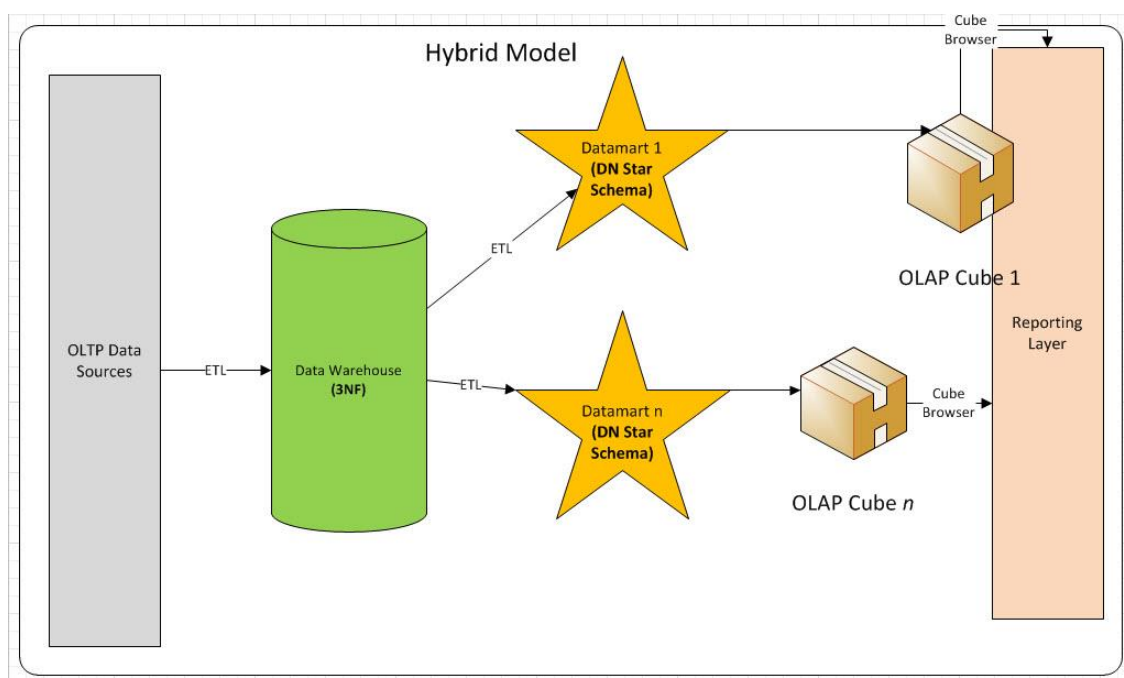


Figura 6 – Modelo Híbrido

(fonte: <https://bennaustin.wordpress.com/2010/05/02/kimball-and-inmon-dw-models>,  
acedido em 2015)

As DW são mantidas de forma independente da base de dados operacional. Estes dois tipos de bases têm objectivos diferentes. A DW preserva os dados de múltiplos anos, rotulados temporalmente, de forma a permitir ao utilizador final a realização de análises e questões complexas. Por outro lado, a base de dados operacional tem como principal objetivo registar os dados diários da organização.

Na tabela 1 são apresentadas algumas diferenças entre as bases de dados operacionais (OLTP) e as DW [18, 19].

Tabela 1 – Data Warehouse vs Base de Dados Operacional

Data Warehouse	Base de Dados Operacional
Registo histórico	Objetivo operacional
Acessos só de leitura	Acessos de leitura e escrita
Acesso por questões <i>ad-hoc</i> e relatórios periódicos	Acesso por transações
Acesso a muitos registos de cada vez	Acesso a poucos registos de cada vez
Carregamentos periódicos de mais dados	Dados atualizados em tempo real
Estrutura otimizada para processamento de questões	Estrutura otimizada para atualizações

Como referido anteriormente, as DW/DM são apenas de leitura. O carregamento das alterações aos dados é efetuado através de um mecanismo designado por *Slowly Changing Dimensions* (SCD). Este mecanismo compreende diferentes tipos, sendo os principais: Escrever por cima

(SCD do Tipo 1), Inserir novo registo (SCD do Tipo 2) e Adicionar novo campo (SCD do Tipo 3) [16].

A tabela 2 permite visualizar um primeiro carregamento de dados, a que se segue a Tabela 3, que exhibe a inserção de um novo dado, substituindo o anterior (SCD do Tipo 1).

Tabela 2 – Registo inicial na tabela de dimensão

Chave	Chave da BD operacional	Departamento	Nome
001	034	Produção	António Soares

Tabela 3 – Alteração SCD do Tipo 1

Chave	Chave da BD operacional	Departamento	Nome
001	034	Produção	Carlos Almeida

A tabela 4 mostra uma SCD do Tipo 2, que constitui a forma mais utilizada na atualização de dimensões. Neste tipo de SCD é adicionado um novo registo, preservando-se sempre os anteriores.

Tabela 4 – Alteração SCD do Tipo 2

Chave	Chave da BD operacional	Departamento	Nome
001	034	Produção	António Soares
002	034	Produção	Carlos Almeida

A tabela 5 apresenta uma alteração do Tipo 3, que acresce uma nova coluna, em resultado da inserção de um novo responsável pelo departamento de produção.

Tabela 5 – Alteração SCD do Tipo 3

Chave	Chave da BD operacional	Departamento	Nome	Nome 2
001	034	Produção	António Soares	Carlos Almeida

### 2.1.3 On-Line Analytical Processing

As OLAP constituem as ferramentas mais comuns para explorar, extrair e analisar dados sob diversas perspetivas (dimensões), possibilitando efetuar consultas *ad-hoc* e mostrar esses dados sob a forma de relatórios e *dashboards*.

A correcta exploração e utilização das ferramentas OLAP impõem compreender alguns conceitos associados [20]:

- Cubo – Estrutura de dados que agrupa as medidas, por níveis e hierarquias, de cada dimensão em análise. Pode conjugar diversas dimensões (exemplos: data, territórios, produtos), resumindo os dados que lhes são referentes (figura 7);

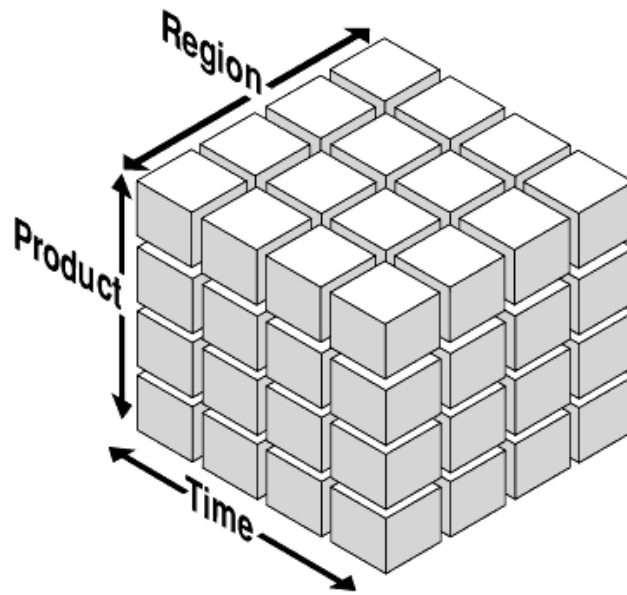


Figura 7 – Cubo Dimensional

(fonte: [http://docs.oracle.com/cd/A87860\\_01/doc/server.817/a76994/dimensio.htm](http://docs.oracle.com/cd/A87860_01/doc/server.817/a76994/dimensio.htm), acessado em 2015)

- Métricas – Conjunto de valores, comumente numérico, presente no cubo e que se baseia numa coluna. As métricas constituem os valores centrais do cubo, processados, agregados e analisados;
- Membro – Item de uma hierarquia, que exprime uma ou mais ocorrências de dados. Pode ser exclusivo ou não-exclusivo. Por exemplo, dois anos civis configuram membros exclusivos, mas os doze meses representam membros não-exclusivos, uma vez que é possível haver mais do que um mesmo mês nos dois anos;
- Dimensão – Entidade que serve de perspectiva de análise, podendo compreender diferentes níveis hierarquizados;
- Hierarquia – Grupo de dados entre os quais existe uma relação. Exemplos possíveis são hierarquias de secções ou geográficas. Nestas últimas, o país está no ponto mais alto da hierarquia, seguido das regiões nele existentes e, no ponto mais baixo, surgem as cidades dessas regiões;
- Nível – Grau de detalhe de uma dada hierarquia.

As principais operações permitidas nos cubos de dados são:

- *Drill-down* – Forma mais comum de exploração de dados por parte dos utilizadores [16]. Permite analisar diferentes níveis de detalhe na informação apresentada. Por exemplo, com o *Drill-down* da dimensão Tempo, é possível observar o resultado anual das métricas ou descer até ao dia, aumentando o grau de profundidade;
- *Roll-up* – É a operação oposta do *Drill-down*. Usando o exemplo apresentado anteriormente, poder-se-á iniciar com os detalhes diários e fazer sucessivas agregações por semana, mês, trimestre, ano etc;



- *Slice and Dice* – É uma das características mais relevante de uma ferramenta OLAP. O *Slice* permite selecionar um subconjunto de dados do cubo, através da restrição de uma determinada dimensão. O *Dice* permite definir um subconjunto através da restrição de mais do que uma dimensão [17].

Existem diversos tipos de servidores OLAP, dos quais se destacam três [21]:

- *Relational OLAP (ROLAP)* – Esta metodologia baseia-se na manipulação dos dados armazenados na base de dados relacional. Desta forma, é possível simular as funcionalidades tradicionais dos servidores OLAP. Em essência, cada acção de *Slice and Dice* é equivalente a adicionar uma cláusula *where* numa instrução SQL;
- *Multidimensional OLAP (MOLAP)* – Esta é a forma mais tradicional de análise OLAP. Em MOLAP, os dados são armazenados num cubo multidimensional. O armazenamento não está na base de dados relacional, mas em formatos proprietários;
- *Hybrid OLAP (HOLAP)* – A tecnologia HOLAP tenta combinar as vantagens de MOLAP e ROLAP. Para obter informações agregadas, HOLAP aproveita a tecnologia de cubo para um desempenho mais rápido. Quando as informações de detalhe são necessárias, HOLAP pode detalhar a partir do cubo presente na base de dados relacional.

#### 2.1.4 Data mining

*Data Mining* é uma das fases do processo de Descoberta de Conhecimento em Base de Dados (DCBD) e remete para a identificação de relações, padrões ou modelos presentes nas bases de dados [17]. De acordo com o que é pretendido, o *Data Mining* pode ter a função descrição, ou seja, possibilitar um acréscimo de conhecimento sobre os dados analisados, ou a função previsão, de detecção de modelos que prevejam, com um elevado nível de precisão, o valor de uma variável.

Em função do objectivo pretendido, são várias as técnicas de *Data Mining* que podem ser executadas - árvores de decisão, regras de associação, regressão linear, redes neuronais, algoritmos genéticos e vizinhos mais próximos.

O *Data Mining* compreende quatro tarefas [17, 22]:

- Classificação - integração de itens em classes previamente definidas. Trata-se de classificar novos dados por referência à classificação de itens passados.
- Segmentação ou *clustering* - contrariamente à tarefa anterior, nesta tarefa as classes não se encontram predefinidas, sendo identificadas a partir das agregações de dados com métricas idênticas, diferenciando-os de outros;
- Sumariação - descrição sumária de determinados dados. Esta tarefa é frequente nas análises exploratórias dos dados, podendo constituir a própria finalidade do *Data Mining* ou integrar as restantes tarefas;
- Modelação de dependências - identificação de um modelo que revela as associações, combinações e correlações mais relevantes entre itens, sendo verificado, a posteriori, a relação existente com a globalidade dos dados.

## **2.2 Self-Service BI**

Uma outra forma de explorar os dados presentes numa DW consiste no recurso a ferramentas *Self-Service Business Intelligence* (SSBI). Estas apresentam-se como uma recente tendência no universo das tecnologias BI, com acentuado crescimento nas organizações, por proporcionarem ao utilizador final uma autonomia técnica na interpretação e significação dos dados. Este pode conceber e gerar análises e relatórios, sem um domínio prévio das ferramentas e estruturas tecnológicas [23]. Estas atividades não requerem a intervenção direta dos profissionais das TI, contudo, compete-lhes a configuração da DW, bem como a implementação da própria tecnologia SSBI, sendo facilitadores da sua utilização. É fundamental que exista, à partida, um alinhamento das ferramentas SSBI ao negócio da organização, devendo ser assegurada uma adequada formação nesta área, que permita aos utilizadores explorarem dados pertinentes à tomada de decisão. A facilidade de gerar inúmeros relatórios, com elevado grau de personalização, pode conduzir a situações de informação desnecessária, reiterada ou excedente nas organizações. Os profissionais das TI desempenham um papel fundamental no evitamento destes cenários ao definirem os conteúdos críticos ao negócio e todos os parâmetros necessários à utilização do SSBI [24].

## **2.3 Aplicabilidade**

O BI capacita as empresas de práticas de medição e avaliação da performance das diferentes áreas do negócio: clientes, aprovisionamento, produção, comercial, qualidade, segurança etc. Identificar as áreas críticas, que acrescentam significativo valor ao negócio, é determinante para definir as fontes de conhecimento credível e fundamental à tomada de decisão.

A implementação de plataformas BI nas empresas surge como uma necessidade premente, decorrente das dinâmicas concorrenciais e competitivas praticadas pelas organizações. A tabela 6 apresenta algumas funcionalidades possíveis, por área organizacional, da utilização das tecnologias BI numa empresa [25]:

Tabela 6 – Aplicações funcionais de BI

<b>Área</b>	<b>Aplicação funcional na empresa</b>
Comercial	Análise do comportamento do consumidor
	Análise da rentabilidade de consumidores / segmentos
	Análise de <i>cross-selling</i>
	Análise da força de vendas
	Análise dos canais de distribuição
Marketing	Penetração no mercado/segmentos
	Eficácia das campanhas de marketing
	Análise do ciclo de vida do produto/serviço
Finanças	Previsão, planeamento e orçamentação
	Análise de performance
	Consolidação financeira
	<i>Reporting</i> Financeiro
Operações/Logística	Eficiência operacional
	Planeamento da produção
	Controlo de qualidade
	Análise da cadeia logística
Recursos Humanos	Planeamento da afetação de recursos
	Avaliação de <i>performance</i>
	Análise da compensação
	Avaliação de competências

## 3 Plataformas de Business Intelligence

Neste capítulo apresentam-se as principais plataformas BI *open source* e comerciais, após o que se efectua uma comparação entre as diversas opções.

As licenças *open source*, também designadas código livre, possibilitam um acesso, sem restrições e custos, a *software* previamente desenvolvido [26, 27]. As ferramentas de código livre permitem uma dinâmica em comunidade, podendo ser adaptadas, reconfiguradas, melhoradas e devolvidas à livre utilização. Muitos autores têm elencado as vantagens da utilização de *software* de código livre [28, 29, 30]. A principal mais valia é a redução, e até isenção, de custos de aquisição. No entanto, o que realmente distingue o *software* de código livre é a combinação de quatro liberdades essenciais [31]:

- Liberdade de executar a aplicação direccionada a qualquer objetivo (liberdade 0);
- Liberdade de analisar o funcionamento da aplicação e alterá-la para que execute o que se pretende (liberdade 1);
- Liberdade de redistribuir cópias, de modo cooperante, à comunidade (liberdade 2);
- Liberdade de partilhar cópias das versões alteradas. Desta forma, toda a comunidade beneficiará das modificações efetuadas (liberdade 3).

As ferramentas de código livre são distribuídas sob a forma de licenças. Cada licença, para além de confirmar a *openness* das ferramentas, define, entre outras, as regras pelas quais o código fonte pode ser alterado, modificado e distribuído.

A implementação e utilização das plataformas comerciais pressupõem o pagamento de licenças aos seus detentores. Este facto pode limitar um acesso generalizado, considerando os custos associados. Desta forma, apenas as organizações com maior capacidade financeira poderão dispor dos recursos necessários à sua aquisição.

## 3.1 Plataformas de BI *Open Source*

Das plataformas *open source* existentes, proceder-se-á à análise das seguintes: Actuate, JaspersoftBI, OpenI, PALO, PentahoBI, SpagoBI e Vanilla. Da globalidade das leituras efetuadas, depende-se serem estas as plataformas de maior utilização no mercado.

### 3.1.1 Actuate

A Actuate [32] fornece *software* para programadores e fabricantes, permitindo o desenvolvimento de soluções seguras e escaláveis, de forma a economizar tempo e melhorar o desempenho. Fundou e gere, em parceria com a Fundação *Eclipse*<sup>1</sup>, o projeto *Eclipse BIRT (Business Intelligence and Reporting Tools)*, cuja finalidade consiste permitir a visualização de dados estruturados, direcionados ao BI. No início de 2015, a OpenText adquiriu a Actuate Corporation, com o objectivo de integrar a plataforma da Actuate nos seus produtos.

Esta plataforma está disponível em duas versões, a versão *BIRT open source* e a versão comercial Actuate ONE. A figura 8 ilustra o conjunto de recursos para a conceção e implementação de projetos BI, que o *BIRT open source* contempla:

- *Birt Interactive Viewer* – permite ao utilizador modificar e personalizar relatórios;
- *Birt Studio* – ferramenta de desenvolvimento baseada em *Web* implementado através da plataforma Actuate. O acesso é efetuado através de um *browser* sem necessidade de instalação de *software*;
- *Birt Spreadsheet* – possibilita o acesso direto a dados presentes em folhas de cálculo;
- *Birt Page Level Security* e *Birt Smartsheet Security* – disponibiliza as ferramentas de segurança da plataforma;
- *Eclipse Birt Designers And Engine* – motor de desenvolvimento de relatórios;
- *Birt iServer and Birt iServer Express* – disponibiliza, gere, faz agendamentos, disponibiliza proteção, executa e partilha relatórios e *dashboards* previamente criados;
- *Birt Mobile* – fornece recursos para implementação de soluções para dispositivos móveis;
- *Birt Performance Scorecard* – permite efectuar a gestão de desempenho em tempo real;
- *Birt OnPerformance* – disponibiliza ferramentas de gestão de desempenho;
- *e.Reports* – ferramenta que permite criar relatórios.

---

<sup>1</sup> A *Eclipse Foundation* é uma fundação sem fins lucrativos, que apoia, desenvolve e promove *software's* em código aberto.

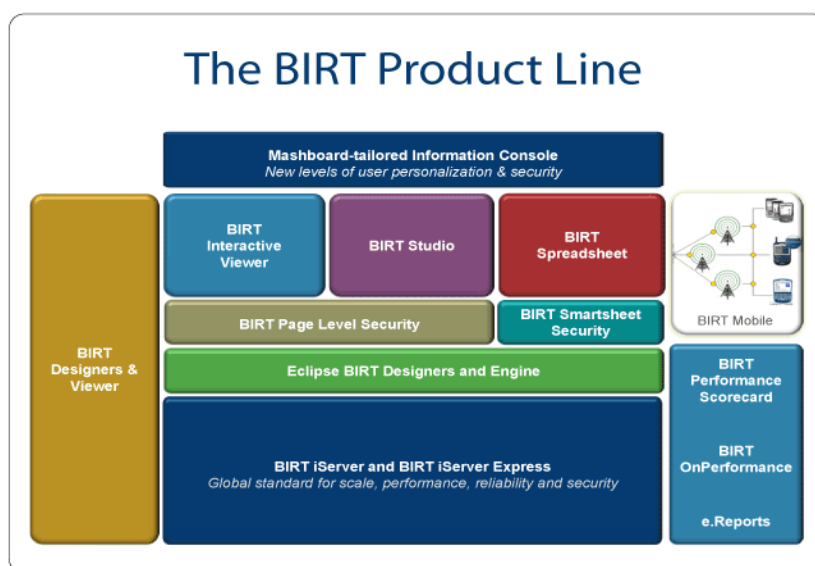


Figura 8 – BIRT Product Line (fonte: <http://infodecisionnel.com/>)

### 3.1.2 JaspersoftBI

A plataforma JaspersoftBI [33] faculta serviços centralizados, necessários à gestão de todo o conjunto BI, permitindo o processamento, organização, proteção e acesso a relatórios, *dashboards*, análises, integração de dados, OLAP e exploração de dados *in-Memory*.

Possibilita transformar, de uma forma rápida, dados, relatórios e gráficos estáticos em elementos interativos, com potencial de partilha imediata e automática pela totalidade da estrutura da empresa.

O utilizador final poderá recorrer aos serviços disponibilizados, sem a intervenção de profissionais das TI. Permite, também, a integração em dispositivos móveis iOS e *Android*.

Esta plataforma está disponível sob dois tipos de licenças: a AGPL<sup>2</sup> (Jaspersoft BI *Community*), gratuita, e a comercial que compreende quatro versões: *Reporting*, *AWS*, *Professional* e *Enterprise*.

Na figura 9 pode-se observar a arquitetura da plataforma BI da Jaspersoft. Ao centro, os principais componentes da plataforma:

- *Jasper Reports Library* – repositório de relatórios em Java para programadores;
- *iReport Designer* – plataforma gráfica de relatórios para programadores e utilizadores avançados;
- *Jasper Reports Server* – servidor de relatórios interativos para utilizadores avançados, que podem ser incorporadas em aplicações *Web* ou móveis. Pode, ainda, funcionar como um repositório central de dados da empresa, através da apresentação de dados críticos em tempo real ou de forma programada na *Web*, dispositivos móveis,

<sup>2</sup> *Affero General Public License*, é uma licença de software livre publicada pela *Free Software Foundation*.

impressora ou endereços de correio electrónico. Os dados podem ser fornecidos sob diversos formatos: HTML, PDF, Excel, OpenOffice, Word, entre outros;

- *Jaspersoft OLAP* – análise de dados OLAP para utilizadores avançados.

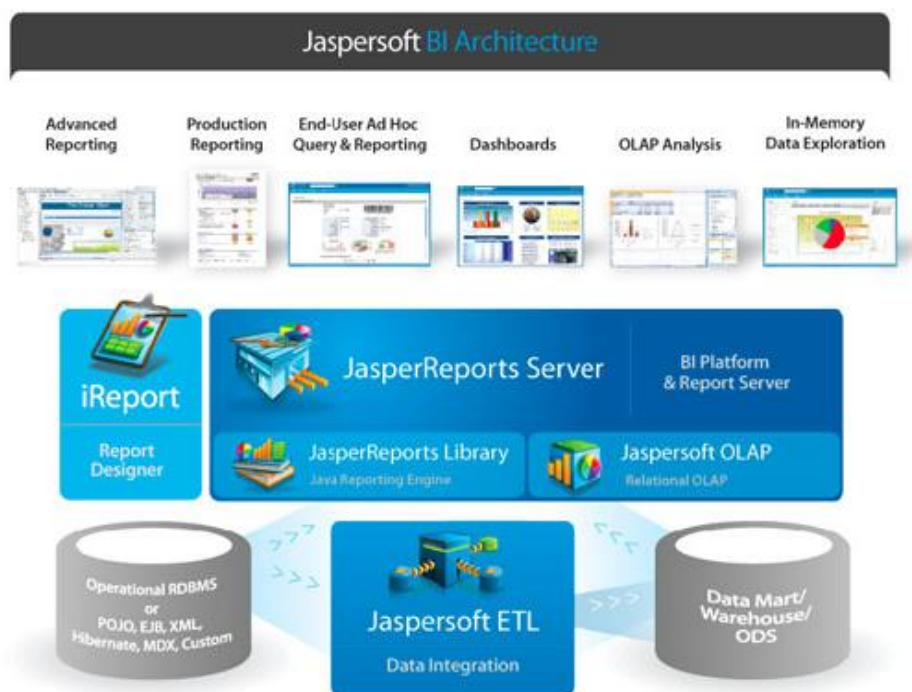


Figura 9 – Arquitetura da Plataforma JaspersoftBI (fonte: <http://www.dewcis.com>)

A JaspersoftBI utiliza o Talend para o processo de ETL e o WEKA<sup>3</sup> para *data mining*.

### 3.1.3 OpenI

A OpenI [34] é uma plataforma BI desenvolvida, desde 2005, sob uma licença *open source*. Ao contrário de outras plataformas, todo o desenvolvimento é efetuado com recurso ao *SourceForge* (repositório de código fonte baseado em *Web*). O modelo seguido pelo projeto OpenI é o *Free/Libre Open Source Software*, que assegura a existência de apenas uma edição do *software*.

Na figura 10 observa-se um servidor de aplicação desenvolvido em *Java*, que disponibiliza os restantes componentes da plataforma. Existe uma camada que garante a segurança dos dados e o acesso aos mesmos. Num nível superior, existem três componentes: interface de utilização, definição de relatórios e conectores. Estes últimos proporcionam a ligação a servidores OLAP, motores de base de dados e ferramentas de *data mining*.

Os componentes permitem aceder a um leque de aplicativos BI: elaboração de gráficos, *Pivot tables*, relatórios, OLAP e processos de *Data Mining*.

<sup>3</sup> WEKA (*Waikato Environment for Knowledge Analysis*) é um *software* disponibilizado sob licença *General Public Licence* (GNU) de *data mining* desenvolvido em *Java* pela universidade de Waikato, Nova Zelandia.

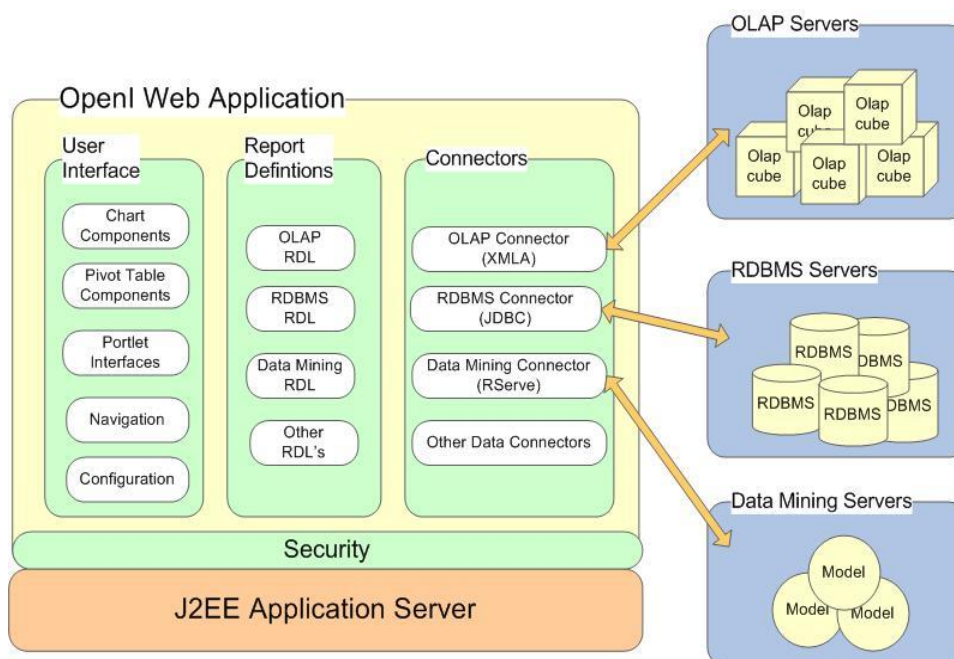


Figura 10 – Arquitetura OpenI (fonte: [http://openi.sourceforge.net/openi\\_product.html](http://openi.sourceforge.net/openi_product.html))

Programados em *Java* e utilizando a tecnologia JavaServer Pages, os componentes utilizados pela plataforma incluem o servidor OLAP *Mondrian* e o motor de relatórios *JasperReports*. O OpenI não inclui ferramenta de ETL. O desenvolvimento desta plataforma esteve focado no conceito *Software as a Service* (SaaS<sup>4</sup>), em detrimento da tradicional plataforma BI centrada num servidor, podendo os seus utilizadores aceder a partir de um *browser*, independentemente do local em que se encontrem.

Após a versão 3.0, o produto foi descontinuado, tendo sido sugerida a utilização de um *add-on*<sup>5</sup> para o PentahoBI e JaspersoftBI.

### 3.1.4 PALO BI Suite

A Palo BI Suite [35] é uma ferramenta desenvolvida pela empresa Jedox AG, distribuída sob uma licença GNU<sup>6</sup>. Esta oferta de BI é totalmente livre de quaisquer taxas. A Palo BI combina todas as aplicações principais – *Palo OLAP Server*, *Palo Web*, *Palo ETL Server* e *Palo para Excel* - numa plataforma BI personalizável. A plataforma é totalmente baseada em produtos de código aberto.

A Palo está concebida para suportar Microsoft Office Excel e Open Office Calc. A Palo para Excel compreende um servidor de base de dados OLAP e o componente Microsoft Excel *Add-On*/Open Office Calc *Add-On*, que permite acrescentar um separador de funcionalidades BI no ambiente da aplicação (Excel ou Calc). Este separador possibilita trabalhar os dados registados

<sup>4</sup> Modelo que permite a aquisição de *software* enquanto um serviço, não havendo a necessidade de proceder à sua instalação, nem existência de *hardware* específico.

<sup>5</sup> *Add-On* refere-se a módulos de *hardware* ou *software* que acrescentam novas funcionalidades ao produto original.

<sup>6</sup> GNU é um acrónimo recursivo para "*GNU is Not Unix*", um projeto lançado em 1983 por Richard Stallman, o objetivo principal deste projeto foi a criação de um sistema operativo baseado em *software* livre.



nas folhas de cálculo, sendo os gráficos e relatórios gerados com recurso às ferramentas existentes na aplicação de suporte (Excel ou Calc). Na ferramenta Palo BI, a inclusão de dados (ETL) e as consultas OLAP são as funcionalidades que adquirem maior expressão.

A figura 11 revela um núcleo central onde interagem, através da ferramenta *Palo ETL Server*, diversas fontes de dados. Existe o módulo *Palo Supervision Server*, que permite a monitorização de toda a plataforma, e o módulo *Palo OLAP Server* para a criação de cubos de dados e posteriores análises, através do Excel e do Calc. Surge, ainda, o *Palo Worksheet Server* que possibilita, com recurso a um *browser*, o acesso a dados.

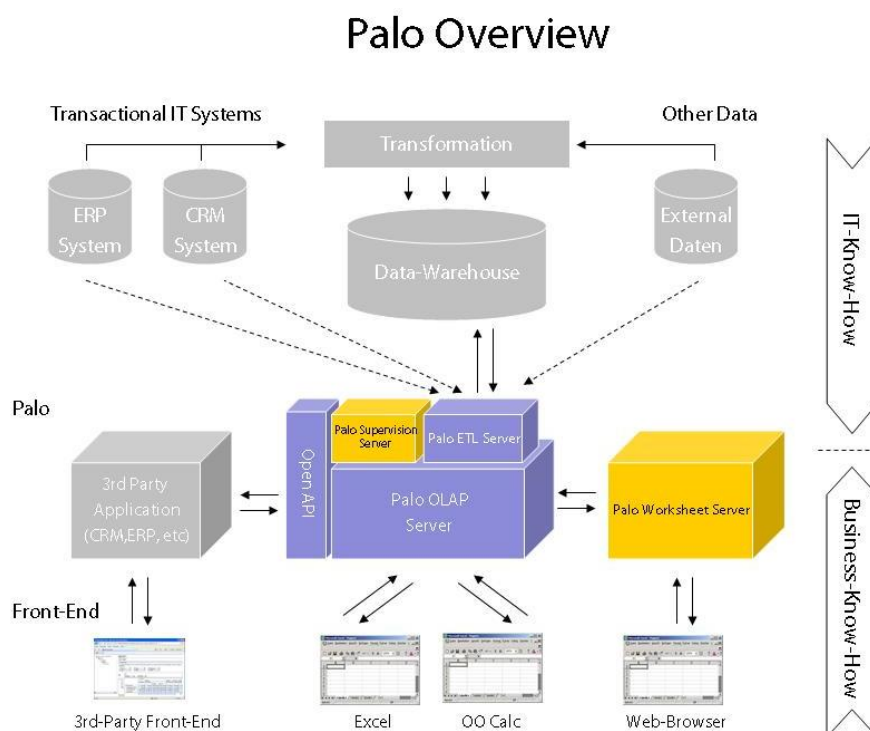


Figura 11 – Arquitetura da *Jedox Palo BI Suite*(fonte: <http://www.legrandbi.com>)

A *Palo OLAP Server* suporta a modelação em tempo real. Isso significa que os utilizadores podem modificar hierarquias e até criar novos cubos em tempo real, no ambiente familiar do Excel. A base de dados devolve novos resultados, imediatamente agregados. A modelação em tempo real é um poderoso recurso de simulação, que permite aos decisores criarem cenários hipotéticos e, desta forma, melhorarem o desempenho da empresa. Esta funcionalidade constitui uma aproximação à recente tendência do *Self-Service BI*.

### 3.1.5 PentahoBI

A PentahoBI [36] é desenvolvida pela *Pentaho Corporation* e é distribuída em duas edições distintas: a versão *Open Source Community Edition* e a *Enterprise Edition*, produto comercial que inclui todas as características da primeira, acrescida de suporte técnico e de algumas ferramentas mais avançadas de ajuda à criação de modelos e relatórios BI.

A PentahoBI é uma plataforma em código aberto, desenvolvida em linguagem *Java*, que inclui as funcionalidades: ETL, OLAP, modelagem preditiva e *data mining*, relatórios e *dashboards*.

A figura 12 apresenta os componentes de constituição desta plataforma: *Pentaho Data Integration* (ferramenta Kettle), *Pentaho Analysis Services* (Mondrian OLAP server), *Pentaho Reporting*, *Pentaho Data Mining* (software Weka) e *Pentaho DashBoard*.

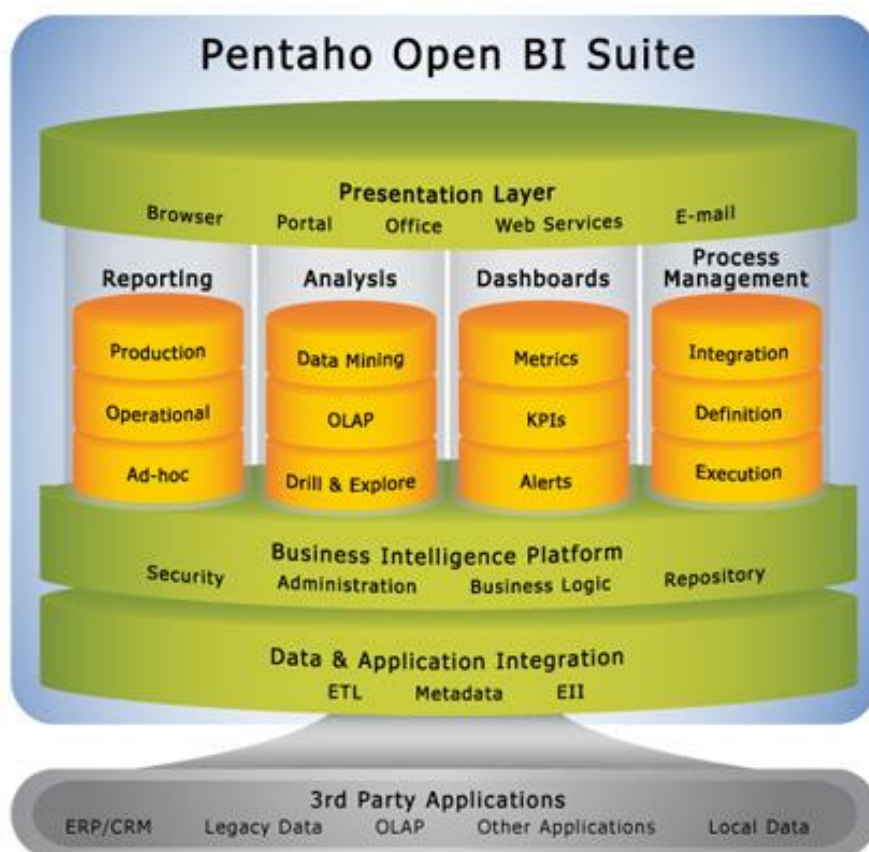


Figura 12 – Arquitetura da Pentaho *Open BI Suite*(fonte: <http://helicaltech.com>)

A plataforma PentahoBI fornece a arquitetura e infra-estrutura necessárias à construção de soluções BI. A *framework* disponibiliza serviços essenciais, incluindo autenticação, registo, auditoria, serviços *Web* e regras de utilização.

### 3.1.6 SpagoBI

A SpagoBI [37] é uma aplicação desenvolvida pela *SpagoWorld*, organização fundada em 2006, com o suporte de uma comunidade *open source*.

Esta ferramenta possui uma licença GNU LGPL<sup>7</sup>, que permite aos utilizadores usufruir da totalidade das funcionalidades. A SpagoBI é inteiramente *open source*, desenvolvida em

<sup>7</sup> GNU LGPL (*Lesser General Public License*) é uma licença de *software* livre aprovada pela *Free Software Foundation*, sendo uma licença de *open source* permite a associação de *software* não livre, incluindo *software* proprietário.

linguagem *Java* e adota o modelo puro de código aberto. Trata-se de um pacote de soluções BI completo disponibilizado, unicamente, na versão *Community*.

Das funcionalidades da SpagoBI, evidencia-se a criação e exportação de relatórios, a criação de gráficos, ETL, OLAP, *data mining*, KPIs, servidor de aplicação, serviço de *login*, *dashboards*, integração com *Office*, GEO, agenda de tarefas e servidor *Web* (figura 13).

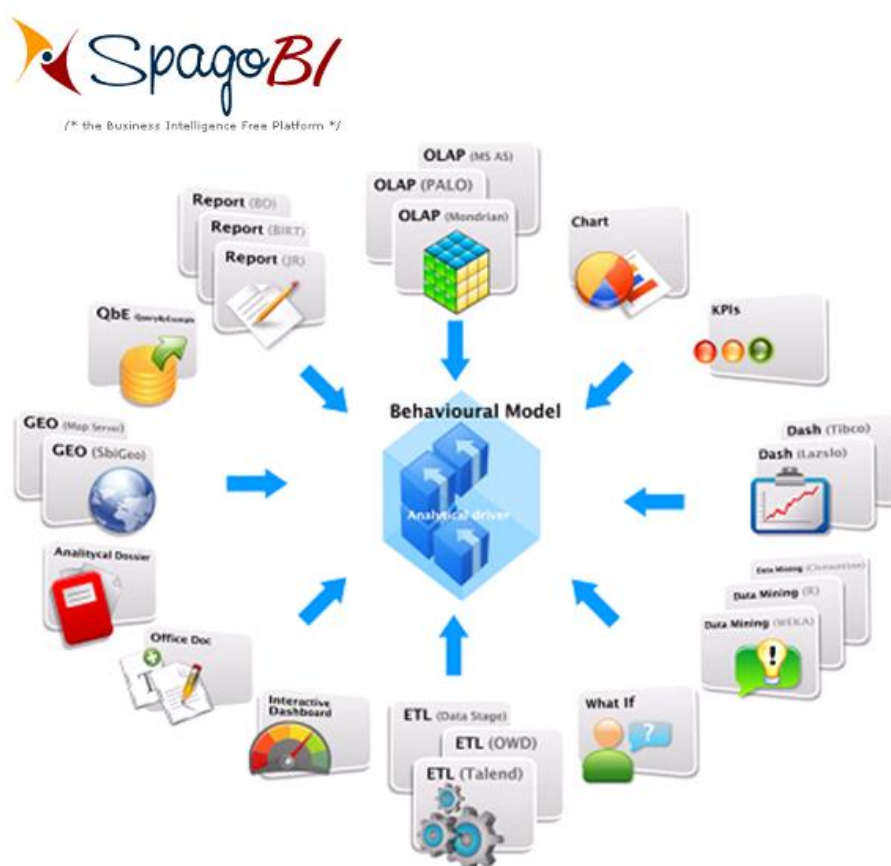


Figura 13 – SpagoBI - Plataforma de Integração (fonte: <http://www.stratebi.com/pt/spagobi>)

Trata-se de uma plataforma BI completa, estável e escalável. A SpagoBI tem soluções diferenciadas como: GEO, consulta por exemplo (QbE), KPIs, *dashboards* interativos e BI em tempo real.

### 3.1.7 Vanilla

A empresa francesa BPM-Conseil, fundada em 2004, é líder no mercado francês de *open source* BI e autora da plataforma Vanilla [38]. Esta é totalmente *open source*, distribuída através de uma licença MPL (*Mozilla Public License*). É desenvolvida em *Java* e apresentada num portal *Web* com recurso à tecnologia GWT<sup>8</sup>.

<sup>8</sup> *Google Web Toolkit* é um conjunto de ferramentas que permite aos programadores criar aplicações com tecnologia *Ajax* em *Java*.

A Vanilla contempla todas as ferramentas necessárias à gestão de dados e visualização destes pelo utilizador final. A última versão, a Vanilla 5, contempla os componentes: *Vanilla Gateway*, *Vanilla Workflow*, *Vanilla Metadata*, *Vanilla plugins for Birt*, *Vanilla Dashboard*, *Vanilla Analysis Designer*, *Vanilla Portal*, *Vanilla KPI*, *Vanilla Analysis*, *Vanilla WebReport*, *Enterprise Services and Hypervision*.

A plataforma fornece suporte a diversos tipos de base de dados, tais como: MySQL (*version 5.6*), MariaDb (10), Oracle (*version 12*), PostgreSQL (*version 9.3*), Microsoft SqlServer 2008 / SqlServer 2012, entre outras.

Na figura 14 pode-se observar os diferentes recursos e características da plataforma, como integração de dados através da ferramenta de ETL (*BiGateway*) na DW, criação de cubos de dados, *dashboards*, relatórios, KPIs e aplicação de técnicas de *data mining*.

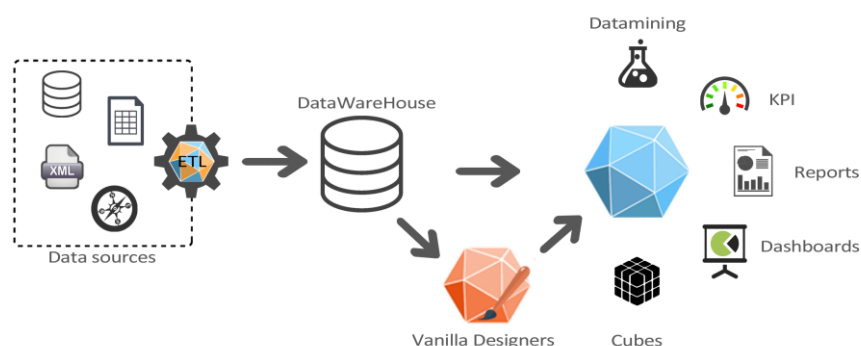


Figura 14 – *Vanilla Functional Schema* (fonte: <http://www.bpm-conseil.com>)

## 3.2 Plataformas de BI Comerciais

De entre as plataformas comerciais, apresenta-se a beMemo, IBM Cognos, Microsoft BI, MicroStrategy, Oracle BI, SAP BI, SAS BI & Analytics e SOL7-BI Machine. Infere-se das leituras realizadas, serem as plataformas que registam maior implementação no mercado.

### 3.2.1 beMemo

A beMemo [39] centra-se no desenvolvimento de aplicações de *Cloud BI* ou *Cloud OLAP* para utilizadores finais ou profissionais de TI. Abrange pequenas e médias empresas, bem como organizações de maior dimensão. Evidencia uma fácil utilização, com a possibilidade de aplicação de diversos modelos já existentes ou de criação de novos modelos, exclusivos de uma dada organização.

A beMemo é disponibilizada como ferramenta SaaS. Possibilita o acompanhamento de KPI's, a realização de análises temporais, comparações e, até mesmo, projeções e simulações. O ETL à base de dados é efectuado a partir de folhas de cálculo ou ficheiros CSV. A plataforma possui versões para *iPad* e *iPhone*, que permitem aos utilizadores aceder e visualizar relatórios e

*dashboards* em tempo real. A sincronização dos dados possibilita o acesso aos mesmos em modo *off-line*.

A figura 15 apresenta uma visão geral da plataforma, desde a integração de dados até à sua disponibilização sob a forma de *dashboards* e relatórios.



Figura 15 – Visão geral do beMemo (fonte: <https://www.bememo.com>)

### 3.2.2 IBM Cognos

A IBM adquiriu, em 2007, a empresa Cognos e, inerentemente, a sua plataforma BI. Atualmente, esta apresenta a versão 10.2.22.

A família IBM Cognos [40] inclui três versões: *Cognos Insight*, para utilização individual e por pequenas empresas; *Cognos Express*, para empresas de média dimensão, departamentos ou grupos de trabalho; *Cognos Enterprise*, para grandes organizações.

As soluções BI incluem relatórios, análise (cenários *what-if*), consultas *ad-hoc*, *scorecards* (KPIs), *dashboards*, OLAP, acesso através de dispositivos móveis, monitorização em tempo real, colaboração, visualização de dados e *Self-Service* BI.

Esta plataforma permite, como já referido, proceder, em tempo real, à importação e exportação de dados, efetuar monitorizações e obter estatísticas, projeções e orçamentos, alargando o BI e tornando-o mais colaborativo, o que potencia o trabalho em equipa e a tomada de decisão.

A IBM Cognos contempla uma versão para dispositivos móveis, que pode ser utilizada *on-line* ou *off-line*. Esta versão tem suporte para: *Apple iPhone* e *iPad*; *tablets Android* e *smartphones*.



Para o processo de ETL é disponibilizada a ferramenta *InfoSphere DataStage* v9.1.

Na figura 16 é possível visualizar, num nível inicial, diversas fontes de dados, seguindo-se um nível intermédio, de gestão do acesso aos dados e, por fim, um nível de disponibilização dos mesmos, sob a forma de análises, relatórios, *dashboards*, *scorecards*, consultas *ad-hoc*, deteção de tendências e análises estatísticas.

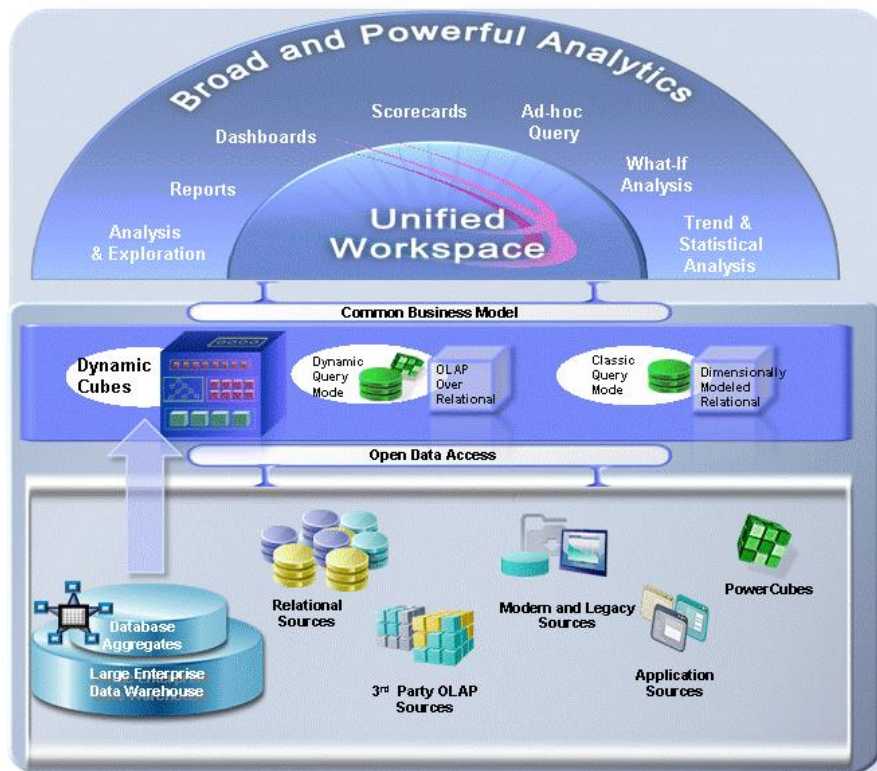


Figura 16 – IBM Cognos 10.2 (fonte: <http://pmsquare.com>)

### 3.2.3 Microsoft BI

Para BI, a Microsoft [41] disponibiliza três ferramentas: SQL Server, Office e SharePoint (ver figura 17). Estas permitem realizar consultas, integração de dados, sincronização, pesquisas, armazenamento na *cloud*, importação/exportação de dados, *dashboards* e *scorecards*, assim como, aceder à informação a partir de dispositivos móveis. As tecnologias colaborativas são disponibilizadas pelo SharePoint Server 2010.

A solução Microsoft divide-se em três grandes grupos: Plataforma de BI (RDBMS, ETL, OLAP, Relatórios); Ferramentas do Utilizador Final & Aplicações de Gestão de Desempenho e Entrega.

O processo de ETL é efetuado a partir do SQL *Integration Services*. A análise de dados e a produção de relatórios é possível com recurso ao SQL *Reporting Services* e ao SQL *Analysis Services*. A última versão da solução Microsoft acresce novos recursos, o *Self-Service BI* e a análise preditiva através do PowerBI.

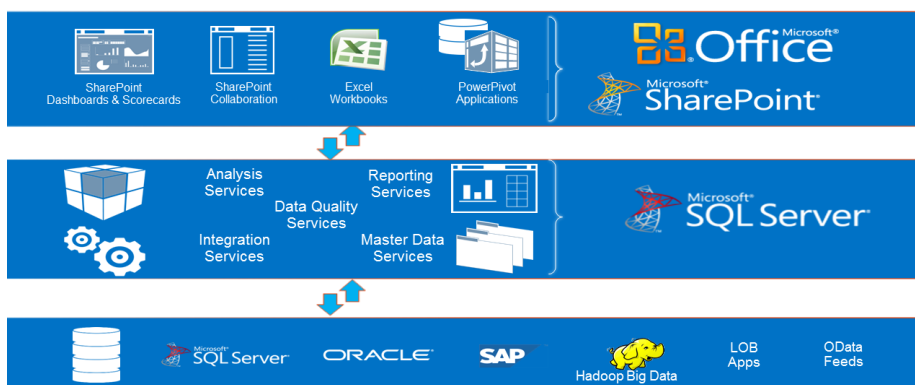


Figura 17 – Microsoft BI Stack (fonte: <http://www.microsoft.com>)

### 3.2.4 MicroStrategy

Fundada em 1989, a MicroStrategy [42] disponibiliza ferramentas integradas BI, que permitem visualizações interativas (*scorecards* e *dashboards*), relatórios, consultas *ad-hoc*, delimitações e alertas, integração com Microsoft Office e dispositivos móveis.

Na figura 18, verifica-se que a versão base da MicroStrategy faculta, na ótica da equipa de TI, as ferramentas de desenvolvimento *MicroStrategy Desktop Analyst & Designer*; *MicroStrategy Architect* e *MicroStrategy Intelligence Server*. As primeiras permitem a criação de relatórios e *dashboards* e a última constitui uma ferramenta de gestão de dados.

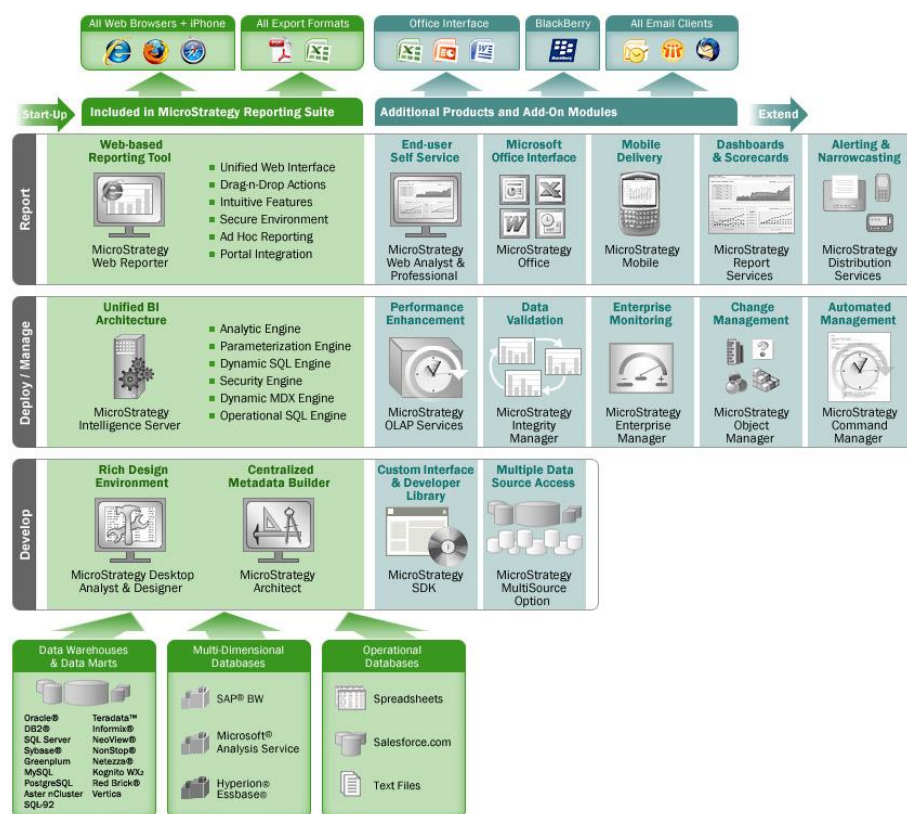


Figura 18 – Arquitetura do MicroStrategy 9 (fonte: <http://www.microstrategy.pt>)

Na ótica do utilizador final, a *MicroStrategy Web Reporter* possibilita a criação e acesso a relatórios e *dashboards*.

Existe, ainda, a disponibilização de outras ferramentas sob a forma de *add-on*. São exemplos destes suplementos, o *Microsoft Office Interface*, *MicroStrategy Mobile*, *MicroStrategy Report Service*, *MicroStrategy Enterprise Manager* etc.

A *MicroStrategy* melhorou a facilidade de utilização, com a introdução de um novo recurso de deteção de dados, o *Visual Insight*, que permite aos utilizadores explorar dados e identificar tendências, minimizando a necessidade de recurso aos profissionais de TI.

Em resposta a necessidades do mercado, a *MicroStrategy* desenvolveu aplicações compatíveis com *software* social, como o *Facebook*. A oferta de *Social Intelligence* da *MicroStrategy* divide-se em duas categorias, a *Enterprise Social Intelligence* e *Consumer Social Intelligence*.

A plataforma *MicroStrategy* não inclui uma ferramenta de ETL; para este processo é sugerida a utilização de uma ferramenta externa.

### **3.2.5 Oracle BI**

O Oracle BI [43] encontra-se disponível no mercado com a designação *Oracle Business Intelligence Foundation Suite*. A plataforma compreende soluções que possibilitam a criação e visualização de relatórios, a partir dos seguintes recursos: *Enterprise BI Platform*, *OLAP Analytics*, *Scorecard and Strategy Management*, *Mobile BI* e *Enterprise Reporting*.

Para as PME's, a Oracle disponibiliza o *Oracle BI Suite Standard Edition One*, solução que integra relatórios, *dashboards*, gráficos, consultas *ad-hoc*, modelação de dados e gestão de base de dados.

Para o segmento das grandes empresas, é disponibilizado o *Enterprise BI (Oracle Business Intelligence Enterprise Edition - OBIEE)*, que acresce às anteriores ferramentas, análises *ad-hoc*, *scorecards*, tecnologias colaborativas, alertas etc.

A *Business Intelligence Foundation Suite* permite gerar relatórios, realizar consultas *ad-hoc*, *dashboards* interativos, *scorecards*, integrar em dispositivos móveis, modelar cenários *what-if*, visualizar geoespacialmente e integrar no Microsoft Office (figura 19).

No que diz respeito ao processo de ETL, este é baseado nas ferramentas *Oracle Warehouse Builder* (OWB) e *Oracle Data Integrator* (ODI).



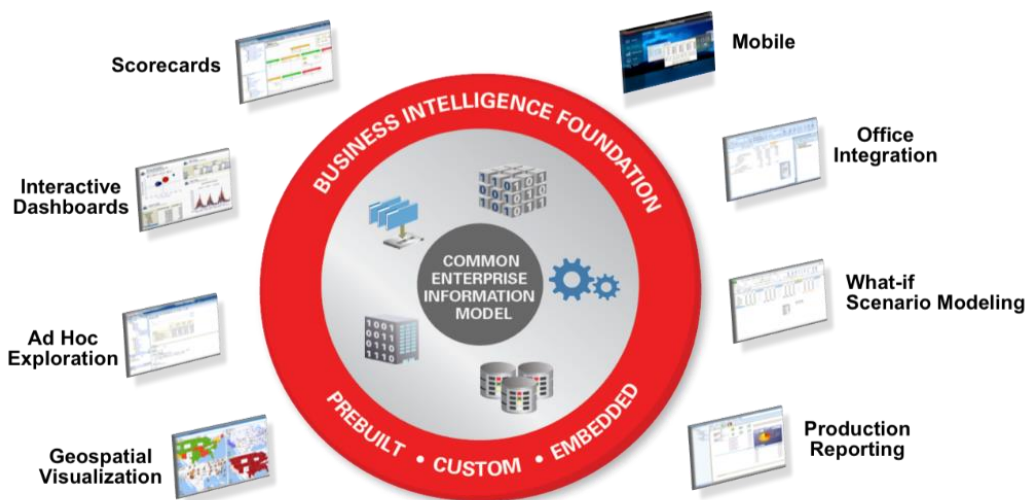


Figura 19 – *Business Intelligence Foundation Suite* (fonte: <https://obibb.wordpress.com>)

### 3.2.6 SAP BI

Surgida em 1972, a SAP [44] (“*Systems, Applications and Products in Data Processing*”) encontra-se fortemente posicionada no mercado do *software* empresarial, assumindo-se como líder. As suas soluções BI pretendem potenciar uma gestão mais eficaz e eficiente, destinando-se a organizações de todas as dimensões e de qualquer área de negócio.

A mensagem institucional da SAP confere relevância à inovação e ao desenvolvimento, associando esse facto à projeção obtida em mais de 50 países e a uma carteira de mais de 250000 clientes.

A SAP permite um vasto leque de funcionalidades: relatórios, *dashboards*, consultas *ad-hoc*, integração com Microsoft Office, integração com dispositivos móveis, OLAP, visualização interativa, modelação preditiva e *data mining*, KPIs e *cloud*. Não integra, ainda, tecnologias colaborativas.

As subáreas e funções que integram a arquitetura SAP BW são: *Data Warehousing*, Plataforma BI, BI Suite (*Business Explorer*) e ferramentas de desenvolvimento (BI Java SDK, *Open Analysis Interfaces* e *Web Design API*). Para ETL, a plataforma utiliza as ferramentas *Business Objects Data Integrator* (BODI) e *Business Objects Data Services* (BODS). Esta última inclui a funcionalidade *Data Quality* (figura 20).

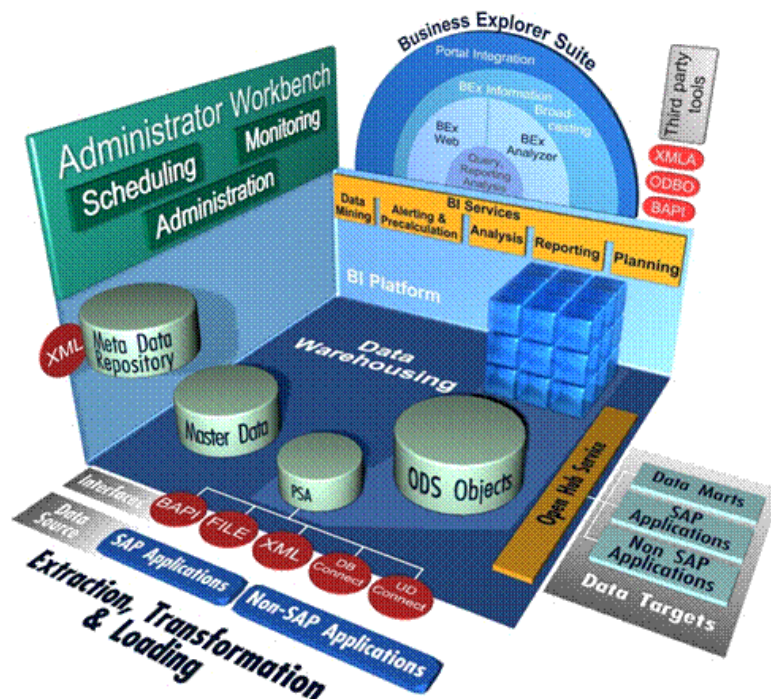


Figura 20 – Arquitetura SAP BW (fonte: <http://help.sap.com>)

### 3.2.7 SAS Business Intelligence & Analytics

A plataforma da SAS [45] fornece *Self-Service* BI e análises, diretas e em tempo real, nos dispositivos móveis, aplicações Microsoft Office e SharePoint. Está, também, disponível a visualização de dados, análise de geolocalização, gráficos, *dashboards*, KPIs, OLAP, *data mining*, tecnologias colaborativas, análise preditiva e relatórios. Estes recursos são fornecidos pelas ferramentas *SAS Enterprise BI Server*, *SAS Analytics Pro*, *SAS Enterprise Guide*, *SAS Enterprise Miner*, *SAS Office Analytics*, *SAS Visual Analytics* e *SAS Visual Statistics*. O *SAS Data Management* proporciona a actividade de ETL.



Figura 21 – SAS Business Solutions framework (fonte: <http://sas.com>)

A figura 21 exibe os principais componentes da *framework SAS: Data Management, High-Performance Analytics, Business intelligence e Analytics*.

### 3.2.8 SOL7 BIMachine

A SOL7 BIMachine [46] é uma plataforma abrangente e completa de ferramentas BI (ver figura 22). Os seus recursos incluem ETL, análise, relatórios, KPI's, portal *Web, Self-Service* BI, tecnologias colaborativas, geolocalização, OLAP, APP Mobile para iOS e *Android*, gráficos e *dashboards*. Estas funcionalidades assentam em dados recolhidos em diversas áreas da empresa e podem ser integrados num interface único, disponibilizando aos utilizadores inúmeras possibilidades de exploração. As análises podem ser exportadas para Excel ou PDF.

O SOL7 BIMachine é uma ferramenta totalmente *Web based*, através de uma licença SaaS.



Figura 22 – Plataforma de *Business Intelligence* SOL7-BI (fonte: <http://www.sol7.com.br>)

## 3.3 Comparação das Plataformas BI

Nesta secção são identificados os critérios de comparação entre as diversas plataformas e procede-se à análise comparativa das respetivas funcionalidades.

### 3.3.1 Critérios de Comparação

Para analisar e comparar as plataformas apresentadas, recorreu-se aos quinze critérios de avaliação das plataformas BI do Instituto Gartner [47]. Estes critérios agrupam-se em três grandes e fundamentais categorias: integração, entrega e análise de informações.

A categoria **Integração** contempla os critérios: Infra-estrutura de BI, Gestão de metadados, Ferramentas de desenvolvimento e Colaboração.

No que respeita à categoria **Entrega de informação**, surgem os critérios Relatórios, *Dashboards*, Consultas *ad-hoc*, Integração no Microsoft Office, Pesquisa baseada em BI e BI móvel.

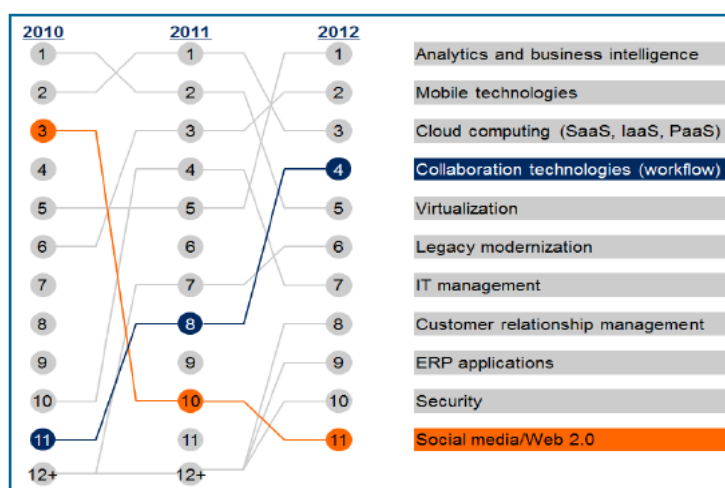
Quanto à categoria **Análise**, esta compreende: Processamento analítico *on-line* (OLAP); Visualização interativa; Modelagem preditiva e *Data Mining*; *Scorecards* (KPIs); Modelagem prescritiva, simulação e otimização.

Na perspectiva do Instituto Gartner, os critérios acima enunciados constituem funcionalidades que, uma vez inerentes às plataformas BI, permitem às organizações dotarem-se de sistemas inequívocos de classificação e medição, no intuito de conferirem suporte à tomada de decisão. Consequentemente, as organizações poderão potenciar os seus desempenhos e níveis de rentabilidade, desenvolvendo uma visão clara das diferentes dimensões que as compõem: clientes, funcionários, fornecedores, serviços e produtos.

Da análise dos resultados de inquéritos realizados a 2300 CIOs [48], sobre as suas prioridades tecnológicas, pode-se observar que as Tecnologias Colaborativas surgem como uma necessidade premente (figura 23). De acordo com o apurado nesses inquéritos, as Tecnologias Colaborativas passaram de 11º lugar em 2010 a 4º lugar em 2012, a seguir às opções: *Analytics and Business Intelligence*, *Mobile Technologies* e *Cloud* (SaaS, IaaS, PaaS). Sublinha-se que, atualmente, estas já integram as plataformas BI.

## the fast rise of collaboration

### Ranking of technologies CIOs selected as one of their top 3 priorities



Decreasing appetite for “social media / Web 2.0” technologies and increasing interest in “collaboration technologies” supports positioning as broader productivity solution



Figura 23 – *Building on Early Successes in UBM’s Social Business Strategy* – Sep. 25, 2013

Esta conquista das Tecnologias Colaborativas nas prioridades dos decisores parece evidenciar ser incontornável virem a integrar permanentemente as plataformas BI. Importa referir que,

em maio de 2014, foi possível constatar já se encontrarem disponíveis na plataforma MicroStrategy [42].

É um facto que as plataformas BI estão em constante e rápida evolução, integrando novos critérios em função das variações e tendências dos mercados. Como exemplo deste desenvolvimento dinâmico e contínuo, surge o acesso a partir de Dispositivos Móveis (Android, OS, Windows *Mobile*). Em 2011, esta funcionalidade não era parte integrante dos critérios BI, nem da maioria das plataformas, mas volvidos 3 anos, está disponível em todas as plataformas consultadas.

Esta tendência é evidenciada pela mudança dos critérios propostos pelo Instituto Gartner em 2015 [49], por referência ao proposto em 2013 [47]. Como exemplos desta constante evolução, pode-se referir o aparecimento de critérios como a implementação na *Cloud* e o *Self-Service* BI; assim como a alteração da definição de Tecnologias Colaborativas, que passou a incluir a integração de dados de diferentes redes sociais. A necessidade de abranger estes critérios foi referida no artigo apresentado na ISDOC 2014 [50].

Quando do início da presente dissertação, os mais recentes critérios da Gartner disponíveis datavam de 2013, tendo sido com base nestes que se desenvolveu esta dissertação. Posteriormente, foram divulgados os critérios de 2014 e 2015. Os primeiros não produzem alterações relevantes aos do ano anterior. Quanto aos referentes a 2015, traduzem uma reorganização dos critérios divulgados na publicação de 2013, que assumem, alguns deles, a forma de subcritérios. A quase totalidade dos critérios de 2013 encontra-se presente em 2015.

Dos mencionados critérios Gartner, seleccionaram-se onze: Relatórios, *Dashboards*, Consultas *ad-hoc*, Integração no Microsoft Office, BI móvel, Processamento analítico *on-line* (OLAP), Visualização Interativa, Modelagem preditiva/*Data Mining*, *Scorecards* (KPI), *Cloud Computing* e Tecnologias Colaborativas. Esta seleção teve por base o facto de ser exequível aferir das respetivas funcionalidades. Os restantes apresentam constrangimentos, desde logo comerciais, de acesso e verificação. A seguir, expõe-se, de forma sucinta, as funcionalidades que cada um dos critérios escolhidos comporta.

O critério Relatórios remete para a capacidade de gerar exposições formatadas e interativas, com ou sem parâmetros, com distribuição a grande escala e possibilidade de agendamento das execuções e distribuições.

Os *Dashboards* consistem na publicação de relatórios na *Web* ou em dispositivos móveis, com painéis intuitivos e interativos que permitem a avaliação e comparação de informações relevantes para atingir um ou mais objetivos de negócio. Permitem uma visualização geral e um acompanhamento mais facilitado dos negócios. É cada vez mais frequente a utilização destes painéis para divulgar dados estratégicos, financeiros e operacionais, em tempo real.

As Consultas *ad-hoc* permitem a utilizadores não especializados obterem dados, em resposta a uma necessidade específica de informação, sem dependerem de relatórios gerados pelas TI. A funcionalidade deve ser concebida de modo a possibilitar um fácil acesso e cruzamento dos dados disponíveis.

O critério Integração no Microsoft Office assenta no facto deste, sobretudo o Excel, ser frequentemente utilizado para a elaboração de relatórios e análises. Neste âmbito, é imprescindível acautelar que as plataformas possibilitem efetivamente essa integração.

O BI móvel permite às organizações fornecerem conteúdos oportunos em dispositivos móveis, através de publicações, que podem assumir formas interativas. Independentemente da localização dos clientes, funcionários ou fornecedores, são disponibilizadas, de forma contínua, informações atualizadas. Por outro lado, e do ponto de vista comercial, o facto das organizações conhecerem a localização dos clientes, através dos dispositivos móveis, pode ser determinante no desenho das estratégias de gestão em prol de uma maior competitividade.

Também relevante para potenciar estratégias competitivas, o Processamento analítico *on-line* (OLAP) permite a análise de informações de natureza comercial, com elevada complexidade e precisão, provenientes de grandes bases de dados organizadas. Esta tecnologia recorre a estruturas multidimensionais para fornecer acesso rápido aos dados. É possível obter, para as diferentes dimensões do negócio (valores de vendas, orçamentos, quantidades vendidas etc.), dados a vários níveis, com maior ou menor pormenor, dependendo das necessidades do utilizador.

O critério Visualização Interativa possibilita disponibilizar, de forma mais eficaz e compreensível, diversos aspetos dos dados ao recorrer a imagens e gráficos interativos.

A Modelação Preditiva/*Data Mining* possibilita às organizações classificarem e categorizarem as diferentes variáveis. A identificação de padrões conduz à definição de modelos de previsão de eventos e cenários, utilizando algoritmos matemáticos.

Os *Scorecards* (KPIs) consistem em indicadores de desempenho mensuráveis, a partir das quais é possível analisar a evolução do negócio, medindo níveis de eficiência e satisfação, no intuito de potenciar resultados. Em função dos objetivos estratégicos da organização, torna-se fundamental selecionar e priorizar indicadores chave de desempenho

As Tecnologias Colaborativas assumem, em nosso entendimento, uma relevante e crescente importância para os decisores das organizações, uma vez que possibilitam a integração de dados provenientes das redes sociais. Estas constituem uma fonte inestimável e inesgotável de informação sobre comportamentos, preferências, tendências, expectativas e necessidades, provenientes dos mais diferentes pontos do planeta. Estes dados, uma vez processados, podem ser determinantes no planeamento e gestão estratégicos dos negócios.

O conceito de *Cloud Computing* (computação em nuvem) reporta-se à utilização da capacidade de processamento, armazenamento e memória existente na Internet. Cloud computing é um modelo que permite a partilha e utilização de recursos através da Internet (por exemplo, redes, servidores, armazenamento, aplicações e serviços) [51]. Desta forma, aplicações, dados e serviços encontram-se disponíveis na Internet, não se confinando aos computadores e servidores das organizações. Muito embora sejam reconhecidas as várias vantagens associadas (redução de custos com equipamento e operações etc.) subsistem algumas questões quanto à garantia de segurança.

Estas funcionalidades permitem às organizações implementarem decisões mais adequadas e melhorarem o seu desempenho. Estas funcionalidades são analisadas nas 15 plataformas BI descritas nas secções anteriores, seguindo-se uma análise comparativa das mesmas.

### 3.3.2 Análise comparativa

Na base das especificações técnicas das plataformas BI, analisadas em função dos critérios selecionados, elaborou-se uma tabela comparativa sobre a disponibilidade das respetivas funcionalidades, tendo em conta as suas características e os sistemas operativos que as suportam.

A tabela 7 pretende constituir um auxiliar à seleção da Plataforma BI mais adequada a uma qualquer organização, considerando as necessidades que esta sinta face à sua área de negócio. O objetivo da análise não é avaliar a performance das funcionalidades das ferramentas, pelo que não se emite qualquer juízo de valor sobre o nível de desempenho das mesmas. Pretende-se expor, de forma comparativa, o que cada uma disponibiliza, contribuindo para um adequado processo de decisão das organizações face às tecnologias BI. Assim, poderá ser implementada, numa lógica de racionalização de custos, a ferramenta BI mais apropriada ao negócio, que forneça dados com impacto no crescimento organizacional.

Da análise comparativa das funcionalidades presentes nas plataformas BI, ressalta o facto das plataformas comerciais abrangerem um maior número dessas funcionalidades. Esta constatação era expectável dado haver forte empenho no seu desenvolvimento. Este sustenta-se nas receitas geradas pela venda da plataforma e o constante investimento tecnológico traduz-se em novos retornos comerciais.

Relativamente às plataformas *open source*, algumas destas são disponibilizadas em versão comercial, com um maior número de funcionalidades, o que determina que as respetivas congéneres em código aberto apresentem uma quantidade reduzida de funcionalidades. São disto exemplo a OpenI e a JaspersoftBI. Esta última valida 7 dos 11 critérios na versão GPL e na versão comercial - *Enterprise* - valida a quase totalidade dos critérios.

Analisando apenas as versões em código aberto, verifica-se que as plataformas da PentahoBI e Vanilla apresentam uma validação de 91%, ou seja, reunindo 10 dos 11 critérios analisados. A SpagoBI valida 100%. São, pois, plataformas muito competitivas quando comparadas com as comerciais, podendo constituir uma efetiva alternativa a estas.

Todas as plataformas compreendem os critérios: relatórios, *dashboards*, integração no Microsoft Office e OLAP. A presença das funcionalidades relatórios e *dashboards* encontra justificação no facto de constituir a ferramenta de gestão mais utilizada nas organizações. Da mesma forma, a universalização do Microsoft Office torna obrigatória a sua integração nas plataformas BI. As vantagens proporcionadas pelos cubos de dados OLAP são uma mais-valia das tecnologias associadas ao conceito BI; por este motivo, são uma presença obrigatória nestas plataformas.

Tabela 7 – Comparação de funcionalidades

Funcionalidades / Critérios	Plataformas Business Intelligence														
	Open Source							Comerciais							
	Actuate	JaspersoftBI	OpenI	PALO BI Suite	PentahoBI	SpagoBI	Vanilla	beMemo	IBM Cognos	Microsoft BI	MicroStrategy	Oracle BI	SAP BI	SAS BI & Analytics	Sol7-BI Machine
Relatórios	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Dashboards	✓		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Consultas <i>ad-hoc</i>			✓		✓	✓	✓		✓	✓	✓	✓	✓	✓	✓
Integração com Microsoft Office	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
BI móvel	✓	✓			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
OLAP	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Visualização Interativa			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Modelagem preditiva e <i>Data Mining</i>		✓	✓		✓	✓	✓		✓	✓	✓	✓	✓	✓	
KPIs	✓				✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Tecnologias Colaborativas						✓			✓	✓	✓	✓		✓	✓
Cloud Computing		✓		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Contrariamente a estas funcionalidades, as tecnologias colaborativas apenas se encontram presentes numa plataforma *open source*, a SpagoBI e em 6 plataformas comerciais. Tal fica a dever-se ao facto de ser uma exigência relativamente recente dos decisores e gestores das organizações. Reitera-se a perspetiva de que todas as plataformas BI virão, com alguma brevidade, a integrar esta funcionalidade, em resposta aos mercados.



Sublinha-se que o presente trabalho não incide na análise comparativa da eficácia das funcionalidades das plataformas, pelo que não se tece considerações sobre o seu desempenho quanto a rapidez, usabilidade, fiabilidade, robustez, relação investimento/retorno etc.

### **3.3.3 Resumo**

Da comparação realizada, conclui-se que as plataformas BI mais abrangentes são a SpagoBI, IBM Cognos, Microsoft BI, MicroStrategy, Oracle BI and SAS BI & Analytics. Estas validam a totalidade dos critérios em análise, seguindo-se as plataformas PentahoBI, Vanilla, SAP BI e SOL7-BI Machine, que validam 10 dos 11 critérios.

Embora não tenha integrado os pressupostos iniciais deste trabalho, no decurso deste, a recente tendência do *Self-Service BI* foi adquirindo visibilidade, constatando-se que se encontra presente na grande maioria das plataformas avaliadas.

Contrariamente ao expetável, a MicroStrategy é a única plataforma comercial que não inclui uma ferramenta de ETL, funcionalidade importante no universo BI.

Analisadas as funcionalidades das plataformas BI, prossegue-se com a implementação destas em ambiente organizacional real. À semelhança de qualquer projeto, a seleção/implementação do BI tem que partir de um correto diagnóstico das condições, recursos, fragilidades e potencialidades da organização. Seguindo-se o planeamento e programação de todas as fases a desenvolver, a implementação das ferramentas e a respetiva operacionalização. Impõe-se, também, a monitorização dos resultados para, se necessário, proceder a medidas retificativas.

## 4 Caso de estudo

No presente capítulo apresenta-se o contexto organizacional que serviu de suporte à implementação de tecnologias associadas ao conceito BI. Trata-se de uma indústria de componentes automóveis, de origem nipónica, presente no nosso país há mais de 25 anos. A fundamentação para o recurso ao BI radicou na necessidade da empresa melhorar o acompanhamento aos indicadores de performance já desenvolvidos e em desenvolvimento. Para tal, procedeu-se ao conhecimento prévio da atividade produtiva, à definição da profundidade e detalhe dos dados a analisar e à identificação das dimensões e factos.

Seguidamente, refere-se o modelo dimensional selecionado - o modelo em estrela - e elenca-se a informação não presente nas bases de dados operacionais, mas imprescindível à implementação do projeto. Finaliza-se com a explicação de todo o processo de implementação das ferramentas BI.

### 4.1 Caracterização da Empresa

A YAZAKI [52] foi criada no Japão há mais de 70 anos com a abertura, em 1938, da primeira unidade industrial. Multinacional de referência, é líder mundial na produção de fios e cabos elétricos e electrónicos, para o setor de componentes automóvel. Surgiu na Europa em 1986, tendo localizado a sua primeira fábrica em Portugal. Atualmente, contabiliza cerca de 134 fábricas filiais, que empregam mais de 200.000 trabalhadores, disseminados por cerca de 38 países.

A Yazaki Saltano de Ovar Produtos Eléctricos, Ltd. (CAE - 23720) localiza-se, desde novembro de 1991, na Zona Industrial de Ovar. De acordo com dados de outubro de 2013, compreende um efetivo de 1668 trabalhadores, distribuídos pelas diferentes áreas que a compõem. Apresenta uma laboração contínua, estruturada em turnos fixos e rotativos.

A empresa possui infraestruturas repartidas por uma área total de 190.000 m<sup>2</sup>, com 54.892 m<sup>2</sup> de área coberta. As áreas cobertas incluem dois pavilhões:

- Pavilhão A, no qual se desenvolvem atividades de produção, garantia de qualidade, armazenamento e expedição;
- Pavilhão B, onde se encontra, entre outros, o PTC, *Porto Technical Center*, responsável pelo desenvolvimento de novos produtos e serviços.

#### 4.1.1 Missão e Visão

A empresa apresenta como missão a oferta de serviços de superior valor aos seus clientes. Coloca o foco nestes, a nível mundial, sendo a sua satisfação a sustentação para o sucesso do negócio. Desta forma, a visão da Yazaki alicerça-se na manutenção de uma liderança pautada pelas práticas tradicionais da casa mãe, embora adaptadas às especificidades culturais das subsidiárias. Pretende afirmar-se como uma organização internacional comprometida e socialmente responsável, capaz de responder de forma célere e flexível ao setor em que opera [52].

#### 4.1.2 Valores

Sendo uma organização global, a Yazaki combina um confiável sistema de valores tradicionais com a modernidade, perspetivando sempre o futuro e a sua posição de *player* global no mercado automóvel.

Respeito pelo meio ambiente; promoção do diálogo e respeito entre todos os intervenientes; compromisso com o sucesso corporativo; garantia de segurança do emprego através de um crescimento rentável e empenho na qualidade, inovação e segurança, constituem os principais valores da empresa [52]. Estes geram uma rede de múltiplas interações e comprometimentos com todos os *stakeholders*.

## 4.2 Descrição do Problema

Com o objetivo de minimizar eventuais desvios ao planeamento da produção, em 2013, a YAZAKY Saltano iniciou a implementação de um conjunto de Indicadores de Performance (*Key Performance Indicators* - KPI's), através de consultas SQL. Para o efeito, recorreu aos *outputs* de desempenho referentes aos últimos 12 meses, que possibilitavam aferir do cumprimento, antecipação ou ultrapassagem das datas previsionais.

Os referidos *outputs* são provenientes de duas bases de dados: a *WheTool*, que armazena elementos inerentes ao planeamento das tarefas executadas, e a MHT, que agrega os dados relativos ao número de horas decorrente dessa execução de tarefas.

No que respeita à base de dados MHT, esta compreende 75 tabelas. De entre estas, selecionaram-se, para a presente análise, as tabelas *bookingDay*, *bookingHours* e *person*, que contêm os dados necessários à implementação dos referidos KPI's. Na figura 24 mostra-se, o diagrama Entidade-Relação (ER), com essas três tabelas e relações.

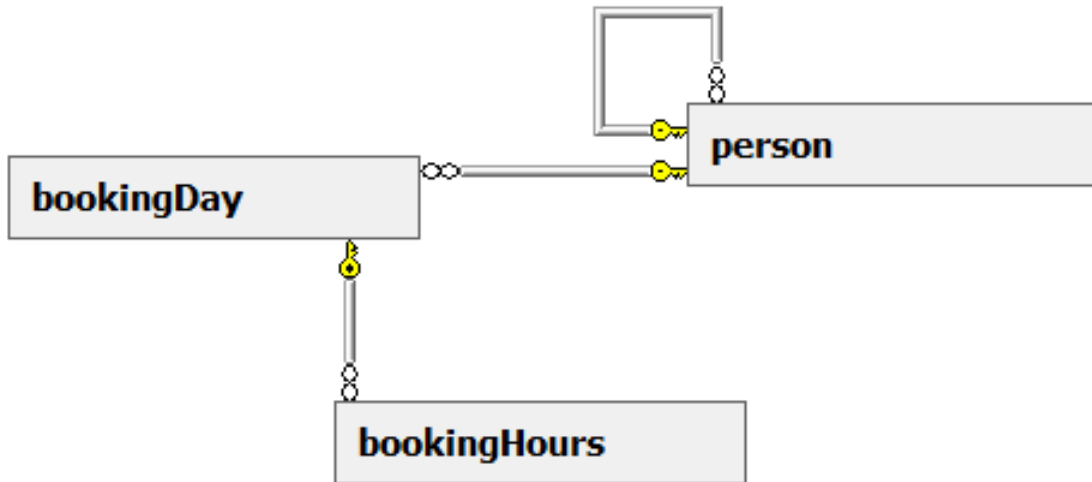


Figura 24 – Diagrama Entidade-Relação simplificado da base de dados MHT

A tabela *bookingDay* contém 157.137 registos e 10 campos. A *bookingHours* dispõe de 192.888 registos e 13 campos e a *person*, 1.367 registos e 39 campos. Embora a base MHT abranja a globalidade das horas executada em todo o grupo YAZAKI, tendo, pois, uma expressão internacional, os dados enunciados apenas dizem respeito à unidade produtiva localizada em Ovar, Portugal.

O *software* MHT encontra-se totalmente desenvolvido e implementado, proporcionando os dados necessários à elaboração dos indicadores previamente definidos. O *software WheTool* está em fase de desenvolvimento, sendo constituído pelas seguintes quatro atividades:

1. “*Manufacturing Design*” – CAE data creation
2. “*ECR’s Incremental*”
3. “*Design Engineer*” – 2D Customer drawing creation
4. “*Design Engineer*” – ECR’s Detail for Quotation

As atividades 1. e 3. estão concluídas e implementadas, as remanescentes encontram-se em desenvolvimento. Presentemente, esta base de dados contempla 118 tabelas, sendo possível verificar, na figura 25, uma amostra representativa do respetivo diagrama ER.

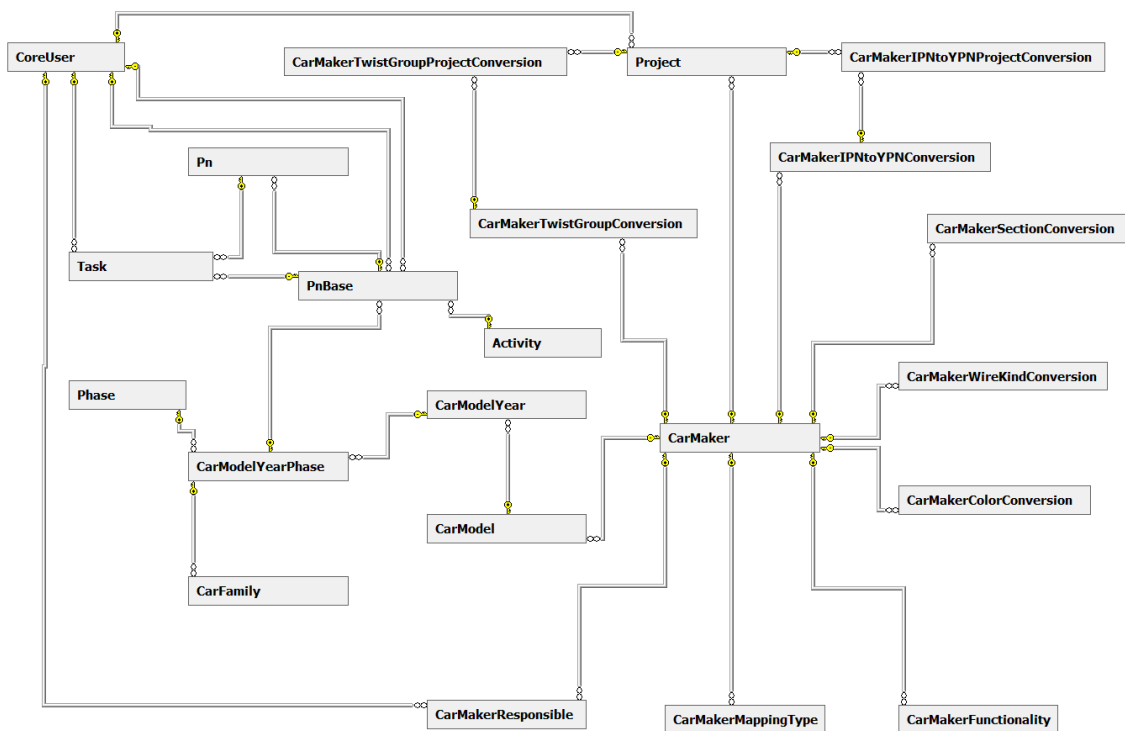


Figura 25 – ER expandido da base de dados *WheTool*

As principais tabelas da base de dados *WheTool* são a *PnBase*, *Pn* e *Task*. Segue-se a tabela 8, que fornece o número exato desses campos e o número de registos associado a cada uma das tabelas.

Tabela 8 – Detalhes das tabelas principais

Nome da Tabela	Nº Campos	Nº Registos
PnBase	106	31.225
Pn	44	125.129
Tasks	25	244.379

No âmbito da base de dados *Whetool*, a empresa definiu, para a atividade *Manufacturing Design*, os seguintes indicadores de performance:

**Part numbers creation**

- **Time Per circuit** - work hours of the period (related wk [Base wk + PN wk]) / qty circuits of all part numbers create [Flag1 date] on the period
- **Quality/PPM** - DCD's qty / circuits of all part numbers create on the period

**Base Drawing Process**

- **Time Per circuit** - work hours of the period (related wk [Base wk]) / qty circuits of all Base drawings processed on the period

- **Report:** Base drawing / complexity range / circuits / qty PN of the period

**Part number Issue**

- **Time Per circuit / Process IssueTime** - work hours of the period (related wk [PN wk]) / qty PNs issued on the period (wk from Base + PN)
- **Delivery Performance** - flag 1, cutting process, assembly process (request date/agreed date)

Para as restantes atividades, *ECR's Incremental, Design Engineer – 2D customer drawing creation* e *Design Engineer - ECR's Detail for Quotation*, não foram ainda definidos indicadores.

Os 56 relatórios, atualmente gerados em Excel, apresentam uma estrutura estática e hierárquica, permitindo a leitura dos dados divididos pelas três fases do ciclo produtivo: RFQ - *Request For Quotation* (definição de requisitos de fabrico), *Prototype* (protótipo) e MP - *Mass Production* (produção). Estes relatórios são disponibilizados, com diferenciados níveis de informação, para cada um dos responsáveis hierárquicos: *Sénior Manager, Manager e Team Leader*.

No que aos *Team Leader* diz respeito, os relatórios são elaborados de acordo com os vários *OEM (Original Equipment Manufacturer)*, ou seja, fabricantes. Importa referir que o *Sénior Manager* tem permissão de acesso a todos os dados, o *Manager* aos dados de todas as *OEM's* dos seus *Team Leader* e estes aos dados das respetivas *OEM's*. No sistema atual, ao nível das *OEM's*, estão agregados os desempenhos globais dos colaboradores, não sendo possível uma leitura individualizada do contributo de cada um no processo produtivo.

Para analisar os dados relativos a cada responsável hierárquico e ou a cada *OEM*, impõe-se o acesso a um ficheiro específico e isolado dos restantes que lhe estão relacionados. Desta forma, não é possível proceder, a partir de um só ficheiro, a uma análise inter-relacionada e integrada dos dados. Este facto constitui um condicionalismo identificado pela empresa, perspetivando-se que o recurso a ferramentas BI, nomeadamente os cubos OLAP, permita o *drill-down, roll-up* e o *slice and dice* por diversas dimensões. Esta análise multidimensional e dinâmica dos dados constitui uma mais-valia dos cubos, permitindo manusear, analisar e visualizar os dados de diferentes formas.

A implementação dos KPI's está a ser realizada através de *Stored Procedures*<sup>10</sup>, cujos resultados são apresentados em tabelas de Excel. No Anexo I, é possível visualizar o *Store Procedure* usado para extrair dados exibidos no relatório *Manufacturing Design* (figura 26). Este constitui um exemplo da natureza estática, avulsa e desagregada dos dados manuseados na empresa, bem como da lacuna que se pretende ultrapassar com a implementação de ferramentas associadas ao conceito BI. Neste relatório, a alternância entre os dados exige a mudança de folhas no Excel,

---

<sup>10</sup> O *Stored Procedure* [62] é um procedimento executado com recurso a comandos em SQL. Estes comandos podem assumir uma configuração de simples consultas ou processos mais complexos, com a execução de multi-tarefas simultâneas como inserções, atualizações ou seleções.

sendo impossível acrescentar novas dimensões (detalhes por mês, diários, análises por colaborador etc.), dispensando a criação de um novo *Stored Procedure*.

Para todas as fases produtivas, o relatório exibido na figura 26 é gerado com base na diferença entre a data planeada e a data real da tarefa *Cutting Process*. São mostradas as percentagens e a quantidade de tarefas terminadas no dia planeado, bem como o número de tarefas antecipado e o quantitativo concluído depois do prazo previsto. É, também, exibida a medida *YTD Delay (qty)*, que corresponde ao valor acumulado das tarefas executadas depois da data planeada. Apresenta-se, ainda, o *YTD (%)*, ou seja, o valor em percentagem das tarefas executadas na data prevista. Por fim, surge o total mensal das tarefas antecipadas (*Earlier*) e cumpridas no prazo definido (*On Time*). Importa referir que o valor das tarefas antecipadas está incorporado no indicador *On Time*. O gráfico é apresentado em colunas empilhadas 100%, com os valores das percentagens das medidas *Delay*, *On Time* e *Earlier* por mês. Sendo necessário assegurar a confidencialidade dos dados da empresa, os resultados quantitativos das métricas são ocultados, apenas sendo visíveis os valores percentuais. Este procedimento de ocultação de dados é repetido ao longo da dissertação e alargado à identificação dos Fabricantes, *Managers* e *Team Leaders*.

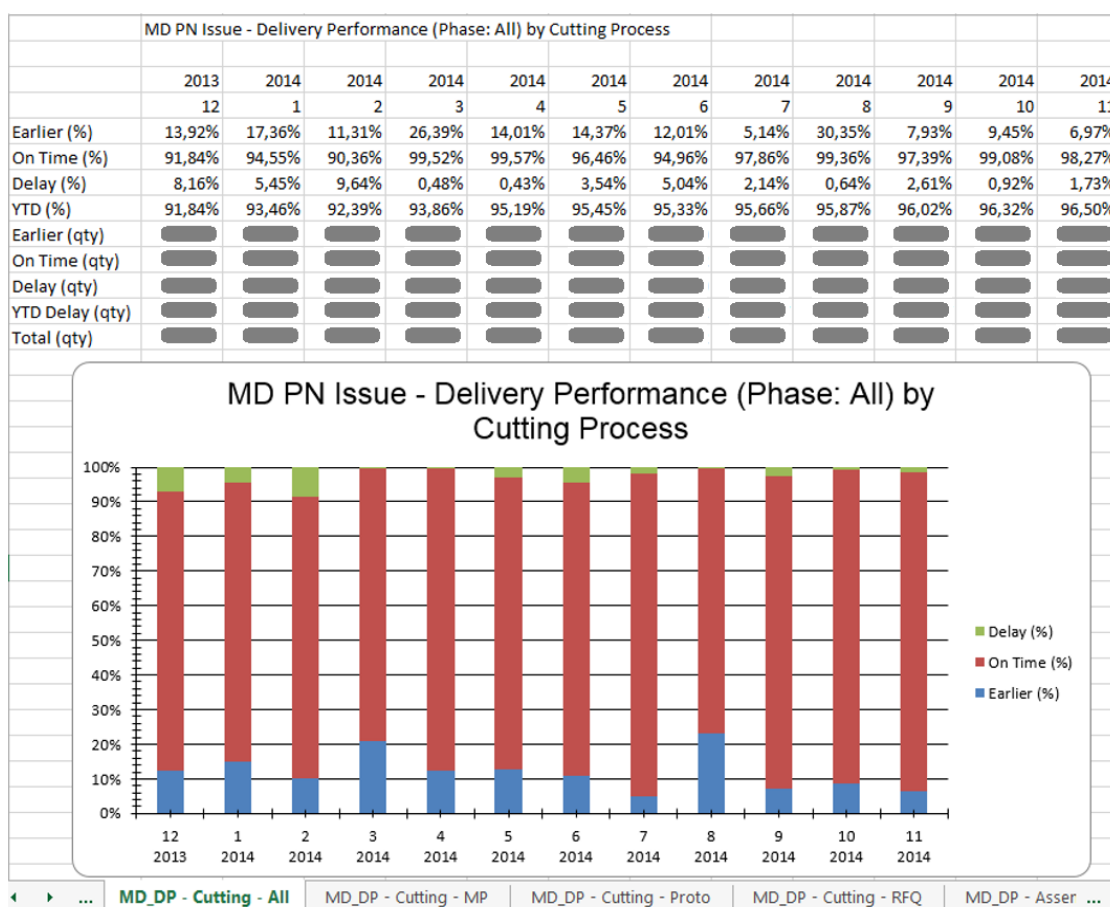


Figura 26 – Exemplo de relatório para *Manufacturing Design*

## 4.3 Identificação do Processo de Negócio, Granularidade, Dimensões e Factos

Nesta secção é apresentada a implementação prática do modelo de Kimball no caso em estudo, identificando-se o processo de negócio, o nível de granularidade, as dimensões e factos.

### 4.3.1 Processo de Negócio

O foco da problemática em análise centra-se no controlo da gestão de tarefas. As métricas analisadas avaliam se as tarefas foram realizadas no tempo previsto, antecipadas ou sofreram atrasos.

### 4.3.2 Granularidade

No presente processo de implementação, o detalhe mais elevado é a verificação do cumprimento, em dias, dos prazos estipulados para cada tarefa.

### 4.3.3 Dimensões

Com base no modelo de dados apresentado na figura 24, e considerando todos os requisitos inerentes ao processo analisado, foram definidas as seguintes dimensões:

- *Activity* – Permite escolher a atividade associada à tarefa;
- *CarMaker* – Descrição dos clientes da empresa;
- *Complexity* – Descrição da complexidade associada à tarefa;
- *Date* – Dimensão temporal, associa as datas de realização das tarefas;
- *Manager* – Identificação do *Manager* associado à tarefa;
- *Person* – Colaborador que executa à tarefa;
- *Phase* – Fase em que se encontra a atividade da tarefa;
- *Report* – Identifica o tipo de relatório mostrado;
- *TeamLeader* – Identificação do *Team Leader* associado à tarefa;
- *TypeTask* – Tipo de tarefas a executar.

Quanto às dimensões acima indicadas, estas vão compreender os seguintes campos e tipos de dados:

Tabela 9 – Dimensão *Activity*

<b>Campos</b>	<b>Tipo</b>
ActivityKey	int
ActivityID	uniqueidentifier
ActivityName	nvarchar(50)
ActivityDescription	nvarchar(MAX)
ActivityKeyName	nvarchar(50)



Tabela 10 – Dimensão *CarMaker*

<b>Campos</b>	<b>Tipo</b>
CarMakerKey	int
CarMakerID	uniqueidentifier
CarMakerName	nvarchar(50)
CarMakerDescription	nvarchar(MAX)
CarMakerCoreUserIdResponsible	uniqueidentifier

Tabela 11 – Dimensão *Complexity*

<b>Campos</b>	<b>Tipo</b>
ComplexityKey	int
ComplexityDescription	nvarchar(250)

Tabela 12 – Dimensão *Date*

<b>Campos</b>	<b>Tipo</b>
DateKey	int
FullDate	datetime
DayNumberOfMonth	tinyint
DayNumberOfWeek	tinyint
DayOfWeek	nvarchar(10)
Month	tinyint
MonthName	nvarchar(10)
Year	int
Semester	tinyint
Quarter	tinyint
FiscalYear*	int

\*Ano Fiscal de Junho a Julho (ano fiscal atual é o 74)

Tabela 13 – Dimensão *Manager*

<b>Campos</b>	<b>Tipo</b>
ManagerKey	int
ManagerName	nvarchar(250)
ManagerPersonKey	int

Tabela 14 – Dimensão *Person*

<b>Campos</b>	<b>Tipo</b>
PersonKey	int
CoreUserId	uniqueidentifier
PersonID	int
PersonUserName	nvarchar(250)
PersonFirtsName	nvarchar(250)
PersonLastName	nvarchar(250)
CoreUserIdAspNet	uniqueidentifier
PersonCountryID	int
PersonEmail	nvarchar(250)
PersonNumber	nvarchar(15)
IsSeniorManager	bit
IsManager	bit
IsTeamLeader	bit
ManagerPersonKey	int
ManagerUserName	nvarchar(250)
EffectiveDate	datetime
ExpireDate	datetime
IsCurrent	nvarchar(3)

Tabela 15 – Dimensão *Phase*

<b>Campos</b>	<b>Tipo</b>
PhaseKey	int
PhaseID	uniqueidentifier
PhaseName	nvarchar(50)
PhaseDescription	nvarchar(MAX)

Tabela 16 – Dimensão *Report*

<b>Campos</b>	<b>Tipo</b>
ReportKey	int
ReportID	int
ReportName	nvarchar(250)
ShortName	nvarchar(250)
FieldStartDate	nvarchar(250)
FieldEndDate	nvarchar(250)

Tabela 17 – Dimensão *TeamLeader*

<b>Campos</b>	<b>Tipo</b>
TeamLeaderKey	Int
TeamLeaderName	nvarchar(250)
TeamLeaderPersonKey	Int
ManagerTeamLeaderName	nvarchar(250)

Tabela 18 – Dimensão *TypeTask*

<b>Campos</b>	<b>Tipo</b>
TypeTaskKey	Int
TypeTaskSortKey	int
TypeTaskName	nvarchar(250)
TypeTaskFriendlyName	nvarchar(250)

#### 4.3.4 Factos

Como os indicadores do processo de negócio são os cumprimentos das datas planeadas, torna-se pertinente o surgimento de campos que apresentam a diferença, em dias, das datas previstas das reais.

A tabela de factos é composta pelos campos exibidos na tabela 19. Destes, os últimos cinco representam as métricas. A chave da tabela de factos compreende os campos *TaskID* e *ReportKey*.

Tabela 19 – Campos da tabela de Factos

<b>Métrica</b>	<b>Explicação</b>
DateKey	Chave da data da realização da tarefa
ActivityKey	Chave da atividade associada à tarefa
PhaseKey	Chave associada à fase da tarefa
CarMakerKey	Chave que identifica o Cliente
PersonKey	Chave identificativa do colaborador executante da tarefa
TeamLeaderKey	Chave identificativa do “Team Leader” associado à tarefa
ManagerKey	Chave identificativa do “Manager” associado à tarefa
TypeTaskKey	Chave identificativa do tipo de tarefa a executar
ReportKey	Chave identificativa do tipo de relatório mostrado
ComplexityKey	Chave identificativa da complexidade associada à tarefa
TaskID	Chave da tarefa na base de dados operacional
DiffDate	Valor da diferença entre a data prevista da real, neste campo podemos ter valores negativos (tarefa realizada antes do previsto), valor igual a zero (tarefa realizada no dia previsto e valores positivos (tarefa realizada com atraso)
EarlierDate	Valor igual a 1 se a tarefa for realizada antes da data prevista.
OntimeDate	Valor igual a 1 se a tarefa for realizada antes ou no dia previsto.
DelayDate	Valor igual a 1 se a tarefa for realizada depois da data prevista.
QtyCircuits	Quantidade de circuitos da tarefa

## 4.4 Modelo Dimensional

O modelo dimensional sugerido consiste no esquema em estrela (*Star Schema*), onde todas as tabelas estão diretamente relacionadas com a tabela de factos. Decorrente da análise efetuada na secção 4.3, conclui-se pela existência de apenas uma tabela de factos com diversas dimensões, não tendo sido necessário a normalização destas últimas. Tal facto legitimou a utilização do esquema em estrela, que se afigurou ser o mais adequado (ver figura 27).

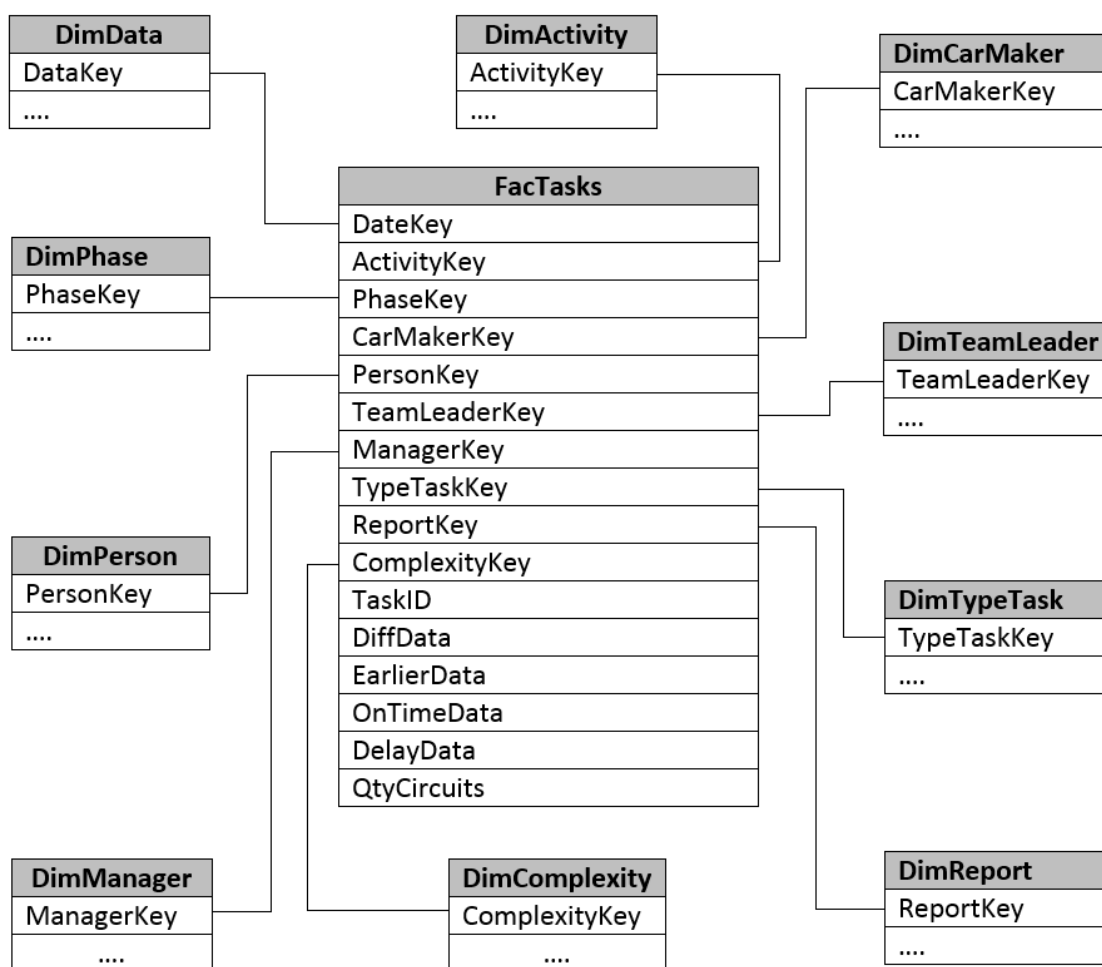


Figura 27 – Esquema do modelo dimensional

## 4.5 Dados Adicionais

Existem dados essenciais de parametrização que não estão presentes nas duas bases de dados referidas (*Whetool* e *MHT*). Estes dados encontram-se num ficheiro autónomo, em formato Excel.

A estruturação interna das tarefas a executar encontra-se definida/tipificada na seguinte hierarquia: *Senior Manager*, *Manager*, *Team Leader* e Colaborador. Na tabela 20 são identificados os *Manager* e *Team Leaders*.

Por questões de confidencialidade organizacional, a identificação nominal dos colaboradores foi ocultada e substituída por uma codificação numérica sequencial.

Tabela 20 – Detalhes da hierarquia

<i>Senior Manager</i>	<i>Manager</i>	<i>Team Leader</i> CarMakerResponsible
<i>SeniorManager.1</i>	<i>Manager.1</i>	<i>TeamLeader.1</i>
		<i>TeamLeader.2</i>
		<i>TeamLeader.3</i>
	<i>Manager.2</i>	<i>TeamLeader.4</i>
		<i>TeamLeader.5</i>
		<i>TeamLeader.6</i>
	<i>Manager.3</i>	<i>TeamLeader.7</i>
		<i>TeamLeader.8</i>
		<i>TeamLeader.9</i>
	<i>Manager.4</i>	<i>TeamLeader.10</i>
		<i>TeamLeader.11</i>
		<i>TeamLeader.12</i>
		<i>TeamLeader.13</i>

## 4.6 Descrição Técnica

O desenvolvimento do projecto foi efetuado com recurso à utilização das seguintes quatro ferramentas:

- *SQL Server Integration Services (SSIS)* – possibilita a implementação do processo de ETL;
- *SQL Server Analysis Services (SSAS)* – permite a criação do cubo de dados;
- *SQL Server Reporting Services (SSRS)* – viabiliza a criação de relatórios;
- *PowerBI 2013* – possibilita a criação de relatórios e *dashboards*.

A utilização destas ferramentas deveu-se ao facto da empresa já dispor da licença de utilização Microsoft SQL Server 2008 R2. Desta forma, recorreu-se à *stack* tecnológica já implementada, que inclui as três primeiras ferramentas. Foi proposto à empresa, no âmbito do projeto a implementar, o recurso a ferramentas *Self-service BI*, dado constituírem a recente tendência no panorama das tecnologias *Business Intelligence*. Considerando a importância de haver uniformidade entre as ferramentas utilizadas, optou-se pelo PowerBI, mantendo-se, assim, uma só plataforma, a da Microsoft.

O SSIS constitui uma ferramenta de ETL, que permite extrair e transformar dados de uma ampla variedade de fontes (arquivos de dados XML, arquivos simples e dados relacionais), transferindo-os para um ou mais destinos. O *Integration Services* possibilita a construção e gestão de *packages*. É uma ferramenta de utilização *drag-and-drop* para criar soluções. A adição de componentes prescinde da escrita de código [53].

O SSAS é uma ferramenta de dados *On-Line Analytic Processing (OLAP)* e *Data Mining* que permite a criação de relatórios. É usado pelas organizações para analisar e efetuar consultas simples ou complexas, relacionando-as com as dimensões presentes no modelo dimensional [54].

O SSRS inclui recursos de programação que, baseados no servidor, permitem alargar e personalizar a funcionalidade dos relatórios criados, para várias fontes de dados. Inclui um conjunto completo de ferramentas para criar, gerir e entregar relatórios e *Application Programming Interface (APIs)*, possibilitando aos programadores integrar ou difundir dados noutras aplicações. As ferramentas do *Reporting Services* funcionam no ambiente *Microsoft Visual Studio* e são totalmente integrados com as ferramentas e componentes do *SQL Server*. Com o *Reporting Services* pode-se criar relatórios interativos, tabulares, *Free-form*<sup>11</sup> e, ainda, sob a forma de gráficos, tabelas ou mapas. É possível publicar relatórios ou agendar processamentos, bem como seleccionar e exportar os relatórios para outras aplicações, como o Microsoft Excel. Os relatórios podem ser exibidos no *SharePoint*, através de uma ligação baseada na *Web* ou como parte de uma aplicação do Microsoft Windows. Podem ser definidos alertas de relatórios publicados no *SharePoint* e receber mensagens de correio electrónico quando das atualizações dos mesmos [55].

O PowerBI 2013 é a ferramenta de *Self-Service BI* da Microsoft. Na figura 28, apresenta-se a arquitetura da sua plataforma cujo ambiente gráfico é o Excel, devido à sua grande disseminação e facilidade de utilização. A ferramenta PowerBI é composta por 4 suplementos: *Power Pivot* (o principal componente), *Power View*, *Power Query*, e *Power Map*.

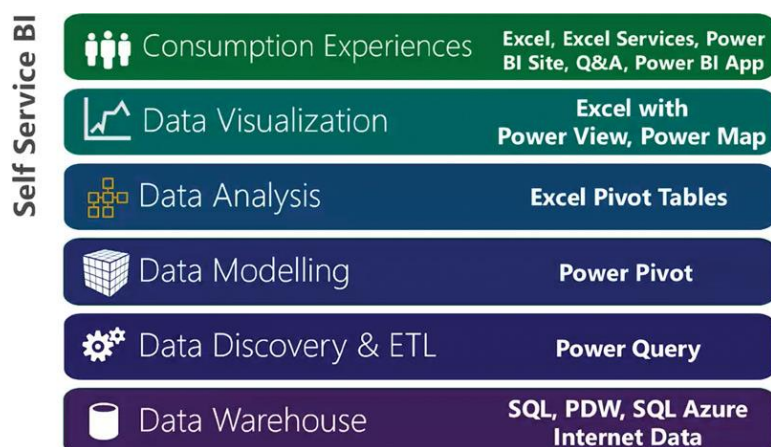


Figura 28 – Arquitetura Microsoft *Self-Service BI*

<sup>11</sup> *Free-form* - Consulta direta a dados através de SQL. Esta não recorre às ferramentas fornecidas pela plataforma, desta forma, inibe as vantagens do cubo de dados, como a possibilidade de fazer o drill down da informação.

O *Power Pivot* facilita a tomada de decisão, podendo ser utilizado para efetuar poderosas tarefas de análise e criar modelos de dados sofisticados. Permite processar grandes volumes de dados, observar rapidamente informações e partilhar facilmente o resultado dessas análises. O *Power Pivot* permite, ainda, a utilização das *Data Analysis Expressions (DAX)*, que efetuam diversos cálculos a partir dos dados.

O *Power View* é uma experiência interativa de exploração, visualização e apresentação de dados através de tabelas, gráficos e mapas. O *Power View* está disponível no Microsoft Excel e no Microsoft SharePoint 2010 e 2013.

O *Power Query* permite construir *datasets*<sup>12</sup> de diversas fontes de dados, inclusive de uma forma colaborativa (ex: *Facebook*, Internet). Disponibiliza um interface de utilização intuitivo para extração, transformação e carregamento de dados.

O *Power Map* exhibe os dados no contexto da sua localização, com recurso a mapas 3D.

A atual plataforma do PowerBI assenta no Office 365 e na Internet, permitindo a partilha e recolha de dados de uma forma colaborativa e de utilização facilitada.

Como referido anteriormente, o centro da plataforma *Self-Service BI* da Microsoft é o *Power Pivot*. As restantes ferramentas trabalham sobre o modelo de dados criado no *Power Pivot*. Habitualmente, os projetos em BI estão centrados na recolha de dados de diversas fontes, tratamento e carregamento numa DW, para posterior análise e criação de relatórios. No caso do *Power Pivot*, este permite utilizar a informação presente numa DW, adicionando a esse modelo outras fontes de informação, o que o enriquece. Neste sentido, permite efectuar, de uma forma simples, análises diferentes, mais complexas, com cruzamento de dados, e completas. Esta ferramenta não substitui, nem invalida a necessidade de criar as DW tradicionais, de forma a manter os dados centralizados.

O desempenho demonstrado pelo PowerBI deve-se à utilização do motor de acesso aos dados *xVelocity In-memory Analytics*. Este proporciona um acesso rápido a grandes quantidades de registos e o seu desempenho advém da utilização da memória RAM. As análises *in-memory* (dados carregados na memória) evitam o efeito “funil”, que acontece quando os dados são gravados no disco, componente muito mais lento. Esta vantagem é mais evidente nos sistemas de 64 bits.

São algumas das vantagens do PowerBI as que a seguir se elencam [56]:

- Funciona como complemento do Excel;
- Facilidade de integração de dados de diversas fontes;
- Possibilidade de utilizar grandes quantidades de dados na ordem dos milhões de registos;
- Utilização das já conhecidas *Pivot Tables* do Excel, presentes desde a versão do Office 97 [57];
- Inclusão da linguagem DAX;

---

<sup>12</sup> Dataset é um conjunto de dados presente na base de dados.

- O modelo de dados criado no PowerBI é apenas de leitura, aumentando o nível de segurança e integridade.

A associação do PowerBI ao SharePoint permite as seguintes vantagens adicionais:

- Partilha e desenvolvimento colaborativo dos projetos trabalhados em PowerBI;
- Agendamentos da atualização dos dados;
- Gestão de versões;
- Criação de perfil de utilização (ex. utilizador só de leitura ou utilizador com acesso total).

#### 4.6.1 Integration Services Project

Nesta secção, é apresentado o processo de ETL (Extração, Transformação e Carregamento de Dados), tendo sido utilizado, para o efeito, o *SQL Server Integration Services*.

No processo de ETL, à extração dos dados das várias fontes para DSA, segue-se um conjunto de transformações dos mesmos. Algumas dessas transformações são a limpeza (por exemplo, correção de erros ortográficos), purgação (que consiste na eliminação dos campos que não vão ser usados), combinação de fontes de dados (agrega dados que fazem sentido estar juntos em vistas unificadas) e criação de identificadores únicos para os objetos que se armazenam nos modelos multidimensionais. Quando a transformação dos dados está completa, estes são carregados no DM.

Este processo implicou a criação de quatro *packages*: dois carregamentos, em separado, das duas bases de dados originárias dos sistemas operacionais (*WheTool* e *MHT*); criação e manutenção do DM e criação de um *package* para gerir os três primeiros.

Os quatro *packages* são:

1. *CreateSA\_From\_WheTool-Full.dtsx*;
2. *AddDataFromMHT\_2\_to\_SA.dtsx*;
3. *CreateInsertUpdate\_Datamart.dtsx*;
4. *MasterExecutionPackage.dtsx*.

Os componentes utilizados no *Package Master Execution Package* compreendem:

1. *Execute SQL Task* – Executa de *scripts* em SQL;
2. *Sequence Container* – Providencia a execução de outros componentes;
3. *Execute Package Task* – Executa um *package* do SSIS.

O *package* “Pai”, que vai despoletar/chamar os *package* “Filhos”, é o *Master Execution Package*, que começa por eliminar e criar a DSA através de um *script*<sup>13</sup> (em SQL). De seguida, são executados/chamados, de uma forma sequencial, os três *packages* dentro do componente *Sequence Container*. A figura 29 mostra a organização do *package MasterExecutionPackage*.

---

<sup>13</sup> Script em SQL é um ficheiro de texto que contém comandos SQL.



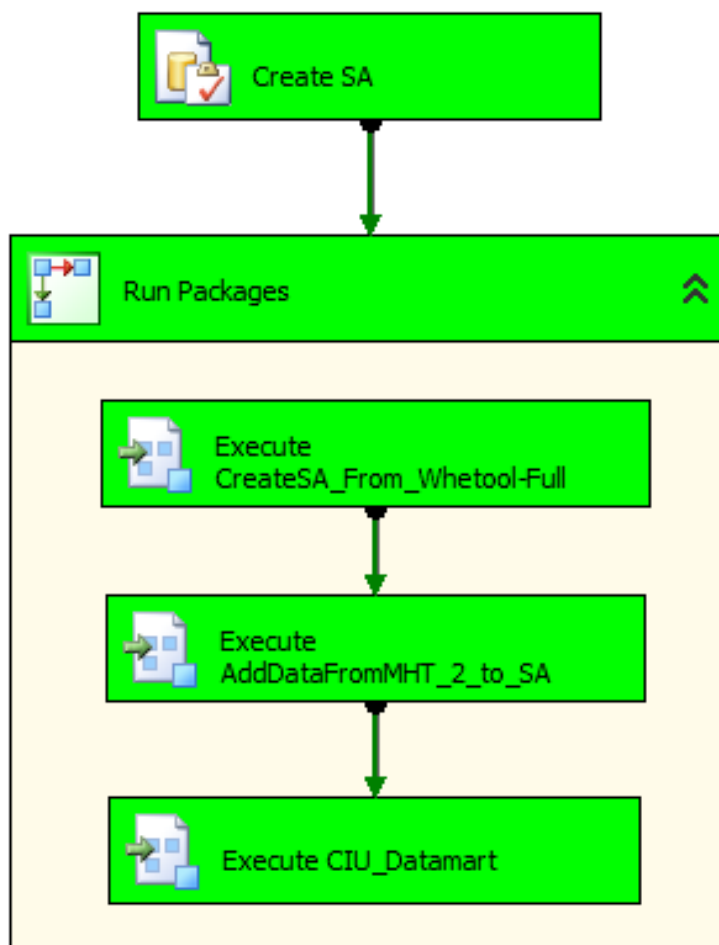


Figura 29 – Master Execution Package

Os packages *CreateSA\_From\_Whetool-Full* e *AddDataFromMHT\_2\_to\_SA* foram criados através da *SSIS Import and Export Wizard...* Este wizard contém 6 passos de fácil execução.

Como resultado do wizard relativo à tabela MHT, é criado o fluxo de dados apresentado na figura 30, que compreende dois componentes: o *Execute SQL Task* e o *Data Flow Task*.

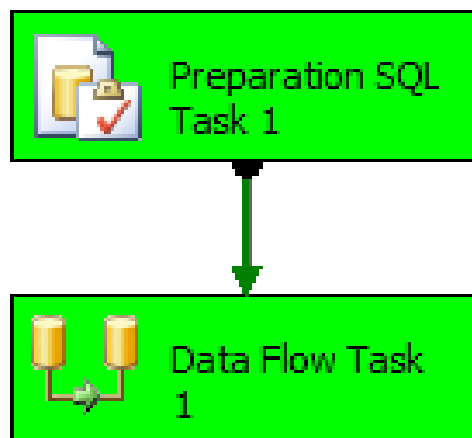


Figura 30 – Control Flow da Base de Dados MHT

O primeiro componente executa um comando em *SQL*, com a finalidade de criar as tabelas, os respetivos campos e tipo de dados, como observável no seguinte código:

```
CREATE TABLE [dbo].[bookingDay] (  
  [bookingDayID] int NOT NULL,  
  .....  
  [notes] nvarchar(4000)  
)  
GO  
CREATE TABLE [dbo].[bookingHours] (  
  [bookingHoursID] int NOT NULL,  
  .....  
  [departmentID] int  
)  
GO  
CREATE TABLE [dbo].[person](  
  [personID] [int] NOT NULL,  
  .....  
  [IsMale] [bit] NULL  
)  
GO
```

O segundo componente - *Data Flows Task* - desdobra-se em três fluxos, um para cada tabela. Cada um dos fluxos, que engloba um *OLE DB Source* e um *OLE DB Destination*, é responsável pela transferência dos dados da tabela de origem para a tabela de destino. A figura 31 mostra a execução, com sucesso, da extração e inserção de dados nas tabelas *bookingDay*, *bookingHours* e *Person*, presentes na DSA. No *Data Flows Task* foram extraídos e carregados 351.392 registos. O mesmo procedimento é verificado no *wizard* utilizado para a tabela *WheTool*.

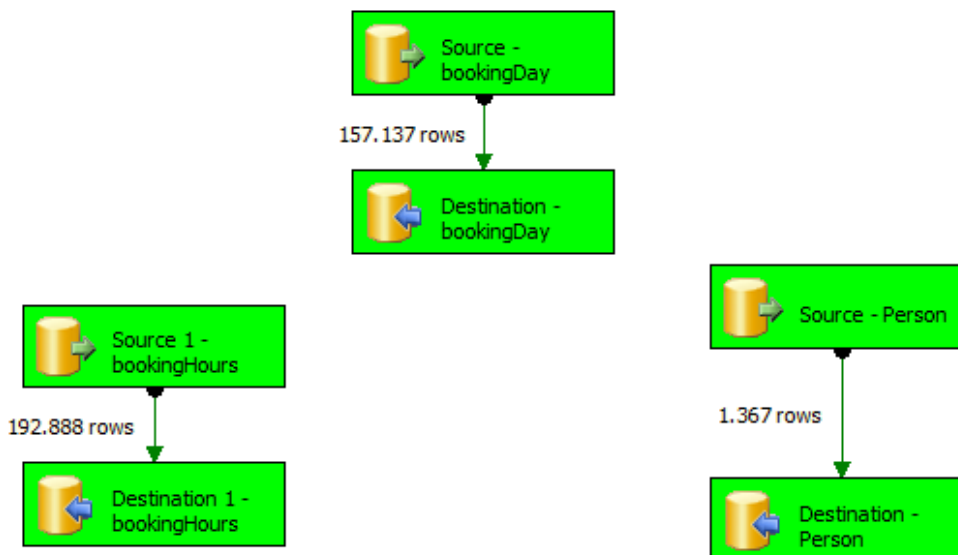


Figura 31 – Detalhe do Componente *Data Flow Task*

O *package CreateInsertUpdate\_Datamart* permite a criação, inserção e atualização dos dados no DM. A figura 32 exibe uma visão geral de todo este processo.

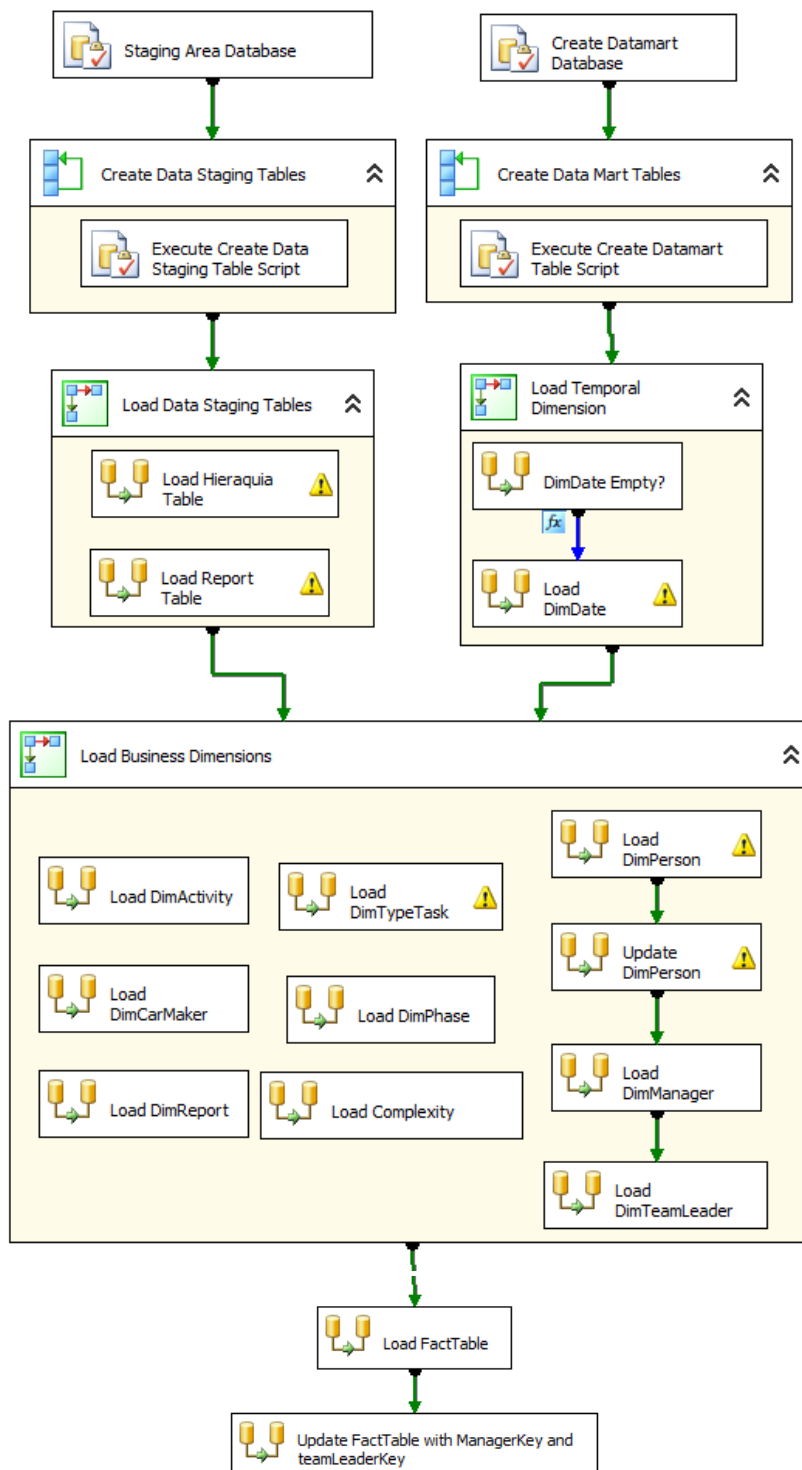


Figura 32 – Visão Geral do *Package CreateInsertUpdate\_Datamart*

Neste *package* utiliza-se os seguintes componentes:

1. *Execute SQL Task* – Executa *scripts* em SQL;
2. *Foreach Loop Container* – Repete a execução de um componente;
3. *Sequence Container* – Providencia a execução de outros componentes;
4. *Data Flow Task* – Possibilita a extração e carregamento de dados;

5. *OLE DB Source* – Extrai dados na origem;
6. *Row Count* – Executa um contagem de registos;
7. *Excel Source* – Extrai dados de um ficheiro em Excel;
8. *OLE DB Destination* – Carrega dados no destino;
9. *Lookup* – Efetua uma correspondência exata de dados;
10. *Data Conversion* – Converte dados para um formato diferente, criando um novo campo;
11. *Slowly Changing Dimension* – Atualiza SCD's;
12. *Derived Column* – Faz atualização de dados;
13. *OLE DB Command* – Permite a atualização de dados no destino;
14. *Conditional Split* – Redireciona o fluxo de dados com base em condições lógicas;
15. *Multicast* – Cria cópias do fluxo de dados;
16. *Union All* – Une diversos fluxo de dados.

Numa primeira fase, é feito um carregamento, em paralelo, da DSA e do DM. São adicionadas na DSA as tabelas Hierarquia, Reports e FactMD\_DQP, necessárias para completar e atualizar os seus dados. Este procedimento é realizado através de um componente que executa *scripts* em SQL, existentes na pasta *ScriptsSA*, associada ao projeto de integração. Sempre que seja necessário criar ou atualizar uma tabela, estes *scripts* revelam-se de grande utilidade por permitirem proceder a alterações mantendo-se a *stream* (fluxo de dados). Através do *Execute Create Data Staging Table Script*, o componente *Create Data Staging Tables* executa todos os *scripts* contidos na pasta *ScriptsSA*. Na fase de carregamento de dados na tabela de Factos, os dados que não tiverem correspondência são encaminhados para a tabela *FactMD\_DQP (Data Quality Poor)*, com a indicação da razão da falha, para posterior análise e correção da mesma.

O componente *Load Data Staging Tables* é composto por dois *Data Flow Task* que fazem a ligação a um ficheiro em Excel, com dados adicionais, de forma a carregar as tabelas Hierarquia e Reports. Os dados são extraídos, mapeados e inseridos nas tabelas de destino.

Como referido anteriormente, é executado, em paralelo, o carregamento inicial do DM. Nesta fase são criadas todas as tabelas do DM e carregada a dimensão *Date*. O procedimento é idêntico ao utilizado na DSA. Na figura 33, são listados os ficheiros executados em SQL.

Nome	Tipo	Tamanho
CreateDimActivity.sql	Microsoft SQL Server Query File	1 KB
CreateDimCarMaker.sql	Microsoft SQL Server Query File	1 KB
CreateDimComplexity.sql	Microsoft SQL Server Query File	1 KB
CreateDimDate.sql	Microsoft SQL Server Query File	1 KB
CreateDimManager.sql	Microsoft SQL Server Query File	1 KB
CreateDimPerson.sql	Microsoft SQL Server Query File	1 KB
CreateDimPhase.sql	Microsoft SQL Server Query File	1 KB
CreateDimReport.sql	Microsoft SQL Server Query File	1 KB
CreateDimTeamLeader.sql	Microsoft SQL Server Query File	1 KB
CreateDimTypeTask.sql	Microsoft SQL Server Query File	1 KB
CreateFactMD.sql	Microsoft SQL Server Query File	4 KB

Figura 33 – *Scripts* SQL do DM

No componente *Load Temporal Dimension* é feito o carregamento da dimensão *Date*. Desse componente fazem parte dois componentes do tipo *Data Flow Task*, estando a execução do segundo condicionada à ocorrência do primeiro. Neste, é estabelecida uma ligação à dimensão *Date*, sendo devolvido o número de linhas existente. Este valor é armazenado numa variável. Na passagem do primeiro para o segundo *Data Flow Task*, é realizado um teste lógico que avalia se o número de linhas é igual a zero (ver figura 34). Confirmando-se essa condição, o componente seguinte é executado, sendo carregada a dimensão *Date*.

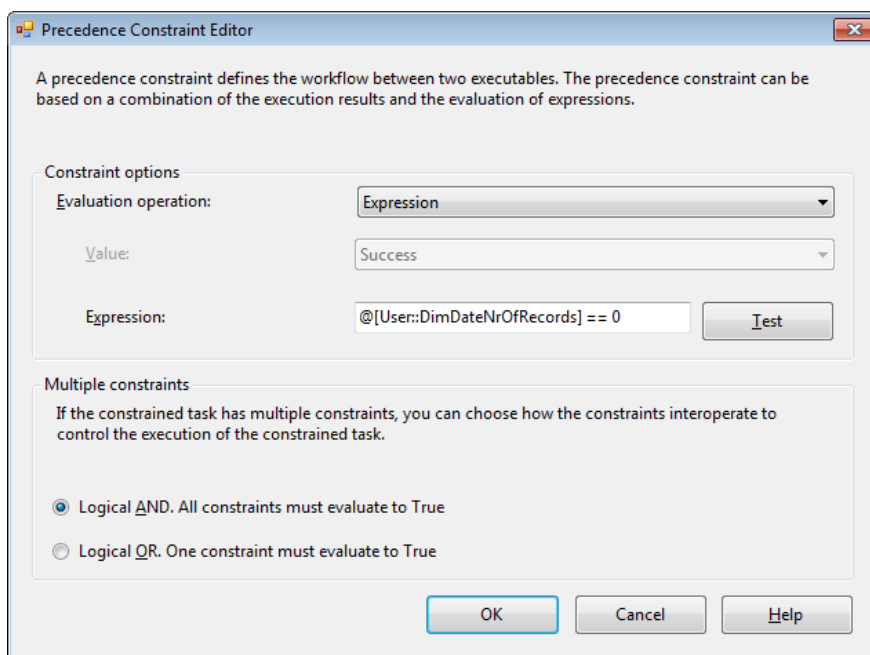


Figura 34 – *Precedence Constraint Editor*

Depois da DSA e da dimensão *Date* carregadas, prossegue-se com o carregamento das restantes dimensões, através das *Data Flow Task*'s. O carregamento das dimensões *Activity*, *CarMaker*, *Report*, *TypeTask* e *Phase* é similar. O processo é executado através de três componentes: *OLE DB Source*, *Lookup* e *OLE DB Destination*. No primeiro componente, se os dados da DSA constarem de uma só tabela, o acesso pode ser efetuado através de uma tabela ou vista. Se os dados estiverem em mais do que uma tabela, o acesso faz-se através de um comando *SQL* (figura 35).

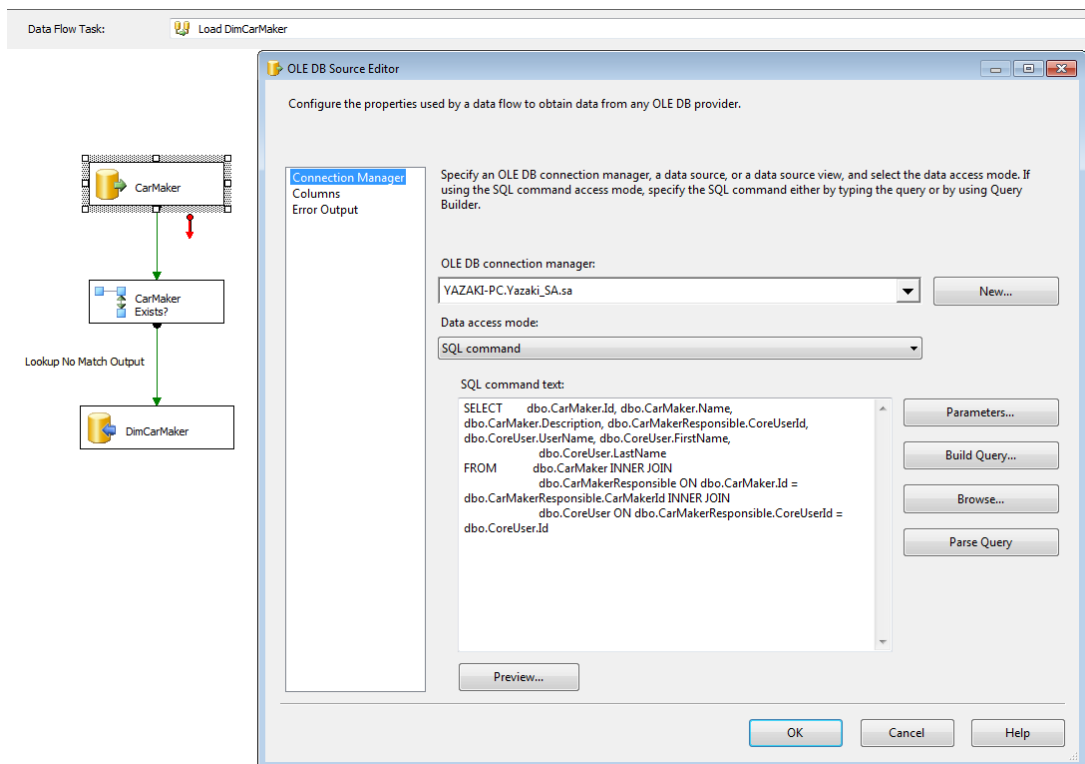


Figura 35 – Comando SQL na dimensão *CarMaker*

Após a extração dos dados, e de modo a evitar a duplicação de registos, ocorre a verificação da sua presença no DM. Este procedimento é realizado com a utilização do componente *Lookup*. A inexistência de *match* resulta na inserção de dados.

A dimensão *Complexity* é idêntica às anteriores, apenas diferenciando-se na utilização de um componente *Data Conversion*. Este permite converter a descrição da *complexity* num formato de *string*, compatível com o existente no DM.

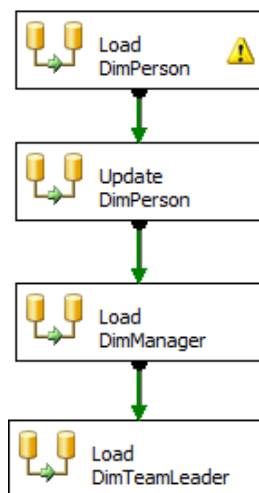


Figura 36 – Dimensões *Person*, *Manager* e *TeamLeader*

O carregamento das dimensões *Person*, *Manager* e *TeamLeader* é efetuado de forma sequencial (figura 36). A dimensão *Person* resulta da junção de três tabelas existentes na DSA: a *Person* originária da MHT, *CoreUser* proveniente da *WheTool* e a Hierarquia. A associação dos dados das três tabelas é realizada através de um componente *OLE DB Source*.

No componente *Derived Column Transformation Editor* (figura 37) é efetuada a transformação para maiúsculas, dos campos alfanuméricos *Person* e *UserName*. São, também, inicializados os campos *IsSeniorManager*, *IsManager*, *IsTeamLeader*, *EffectiveDate* e *IsCurrent*.

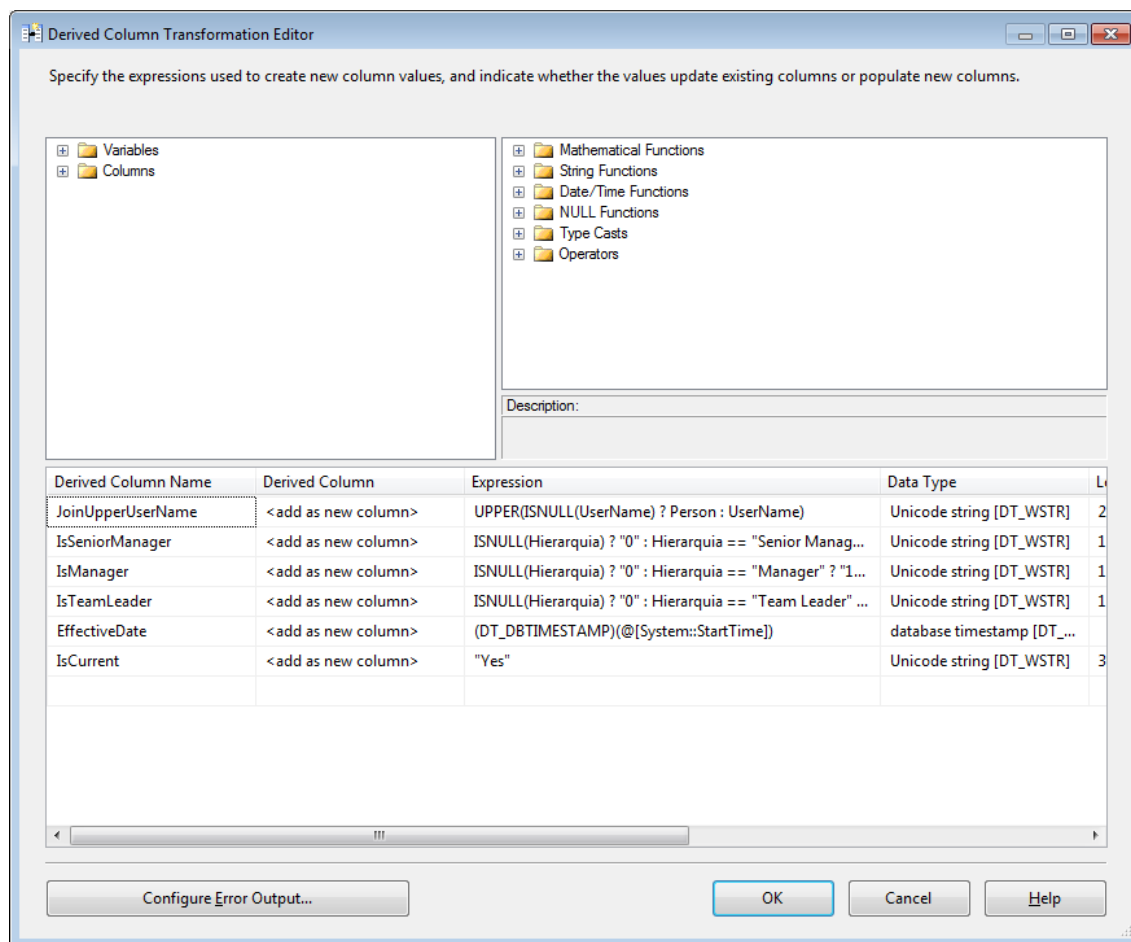


Figura 37 – Transformações no fluxo da Dimensão *Person*

Às transformações ocorridas, segue-se a verificação da duplicação de dados no DM e a inserção, na tabela *DimPerson*, dos que não apresentam *match*.

No componente *Update DimPerson* é feita uma atualização do campo *ManagerKey*, presente na dimensão *Person*.

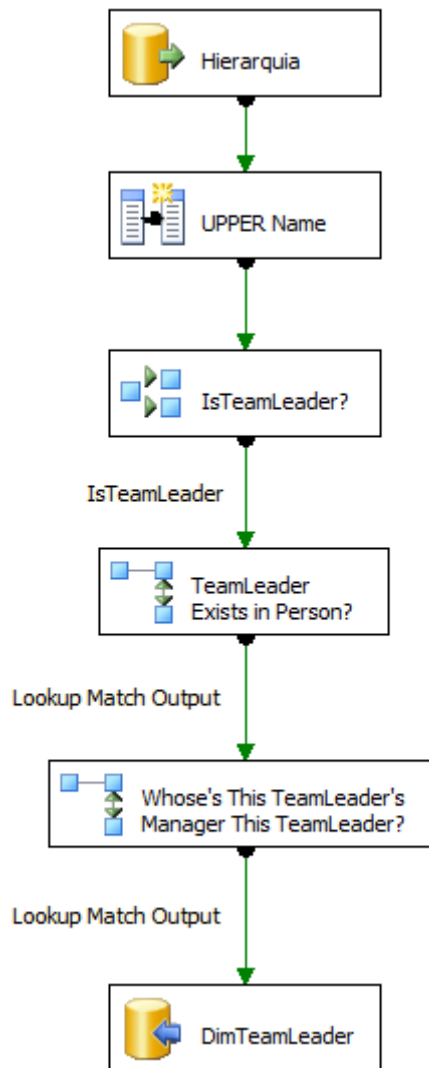


Figura 38 – Componente *Load TeamLeader*

Os componentes *Load DimManager* e *Load TeamLeader* são executados de uma forma semelhante. É possível observar que, no primeiro componente da figura 38, é feita a leitura dos dados na DSA, seguida de uma transformação. O campo *Person* é convertido em texto, no formato maiúsculas. No terceiro componente da figura, os *TeamLeader's* são isolados. No quarto, é procurada na tabela *Person* a *Key* do *TeamLeader*. Segue-se a busca pelo *Manager* correspondente ao *TeamLeader* e o fluxo culmina no mapeamento (figura 39) e carregamento dos dados na tabela *DimTeamLeader*.



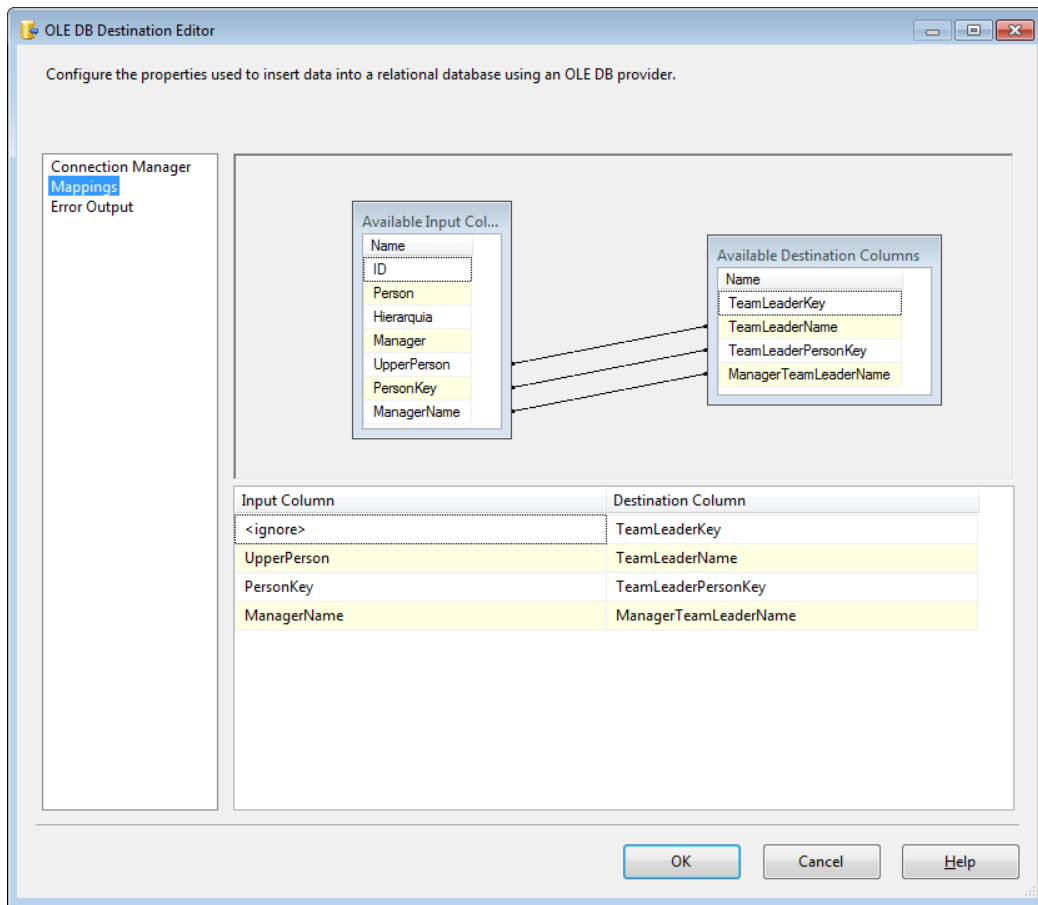


Figura 39 – Carregamento da Tabela *DimTeamLeader*

Terminado o carregamento das dimensões, inicia-se o carregamento da tabela de factos (figura 46). O processo principia com o componente *OLE DB Source*, sendo executada uma consulta *SQL* (Anexo II). Os dados extraídos carecem de transformações, que implicam uma truncatura de campos do tipo *Date* e respetivas conversões. Esta ação permite fazer o *Lookup* da *Date* e obter a correspondente *DateKey*. Executam-se, ainda, seis *Lookup*'s para conseguir as respetivas *Key*'s. No componente *Split by Task*, verifica-se a divisão do fluxo de dados, de acordo com o tipo de relatório pretendido. Em virtude do campo *PnFlag1* dar origem a dois relatórios distintos, é feita uma duplicação do fluxo *OutputFlag1*, através do componente *Multicast*.

Simultaneamente, nos dois componentes seguintes, é calculada, em dias, a diferença entre as datas, sendo, também, criado um novo campo, que identifica o relatório solicitado. Estão, pois, reunidas as condições para a inicialização das métricas *EarlierDate*, *OnTimeDate* e *DelayDate* (figura 40).

Derived Column Name	Derived Column	Expression
DiffDate	<add as new column>	DATEDIFF("day",PnFlag1PlanEndDate,PnFlag1RealEndDate)
ReportName	<add as new column>	"Agree Date"
EarlierDate	<add as new column>	DiffDate < 0 ? "1" : "0"
OnTimeDate	<add as new column>	DiffDate <= 0 ? "1" : "0"
DelayDate	<add as new column>	DiffDate > 0 ? "1" : "0"

Figura 40 – Cálculos, transformações e inicialização das métricas

A inexistência de *match* nos *Lookup's* das datas redireciona os registos, através do fluxo *Lookup Error Output*, para um componente que adiciona a razão da falha, sendo os mesmos carregados na tabela *FactMD\_DQP*, existente na DSA.

Os registos com *Lookup* positivo são direcionados, através do fluxo *Lookup Match Output*, até ao componente *Union All Fact*, obtendo-se um único fluxo de dados. A este fluxo é adicionada a *Key* do tipo de relatório e é inicializado o campo *QtyCircuits*. Neste, são inseridas as quantidades existentes no campo *TaskQty*. Se este campo for *NULL*, é atribuído o valor zero (figura 41).

Derived Column Name	Derived Column	Expression
QtyCircuits	<add as new column>	ISNULL(TaskQty) ? 0 : TaskQty

Figura 41 – Detalhes do componente QtyCircuits

Posteriormente, verifica-se se o facto já existe na tabela de factos. Caso haja correspondência da chave primária, o fluxo é redirecionado para eventual atualização do registo nessa tabela. Na inexistência de *match*, ocorre a inserção do novo facto. O último componente, o *Update FactTable with ManagerKey and TeamLeaderKey*, efetua uma atualização à tabela de factos, com as *Key's* do *TeamLeader* e do *Manager* (figura 32). A tabela de factos passa a conter 186.153 registos (figura 42). Este número está aquém do quantitativo real de registos existente na base de dados da empresa. Para este projeto, e por motivo de proteção de dados, foi disponibilizado um menor número de registos. Por exemplo, numa das bases de dados operacional cedida, a MHT, a tabela *bookingHours* contém 192.888 registos, porém, na base de dados real, essa tabela ultrapassa os 3 milhões.

A figura 42 exhibe uma visão geral do fluxo de dados, desde a extração da DSA até ao carregamento na tabela de factos. Na máquina virtual onde foi desenvolvido, todo o processo tem a duração de cerca de 6 minutos. Este período de tempo deve-se às características da referida máquina, que utiliza 4096 MB de memória RAM, processador Core i7, com quatro núcleos e sistema operativo Windows 7, de 64 bits.

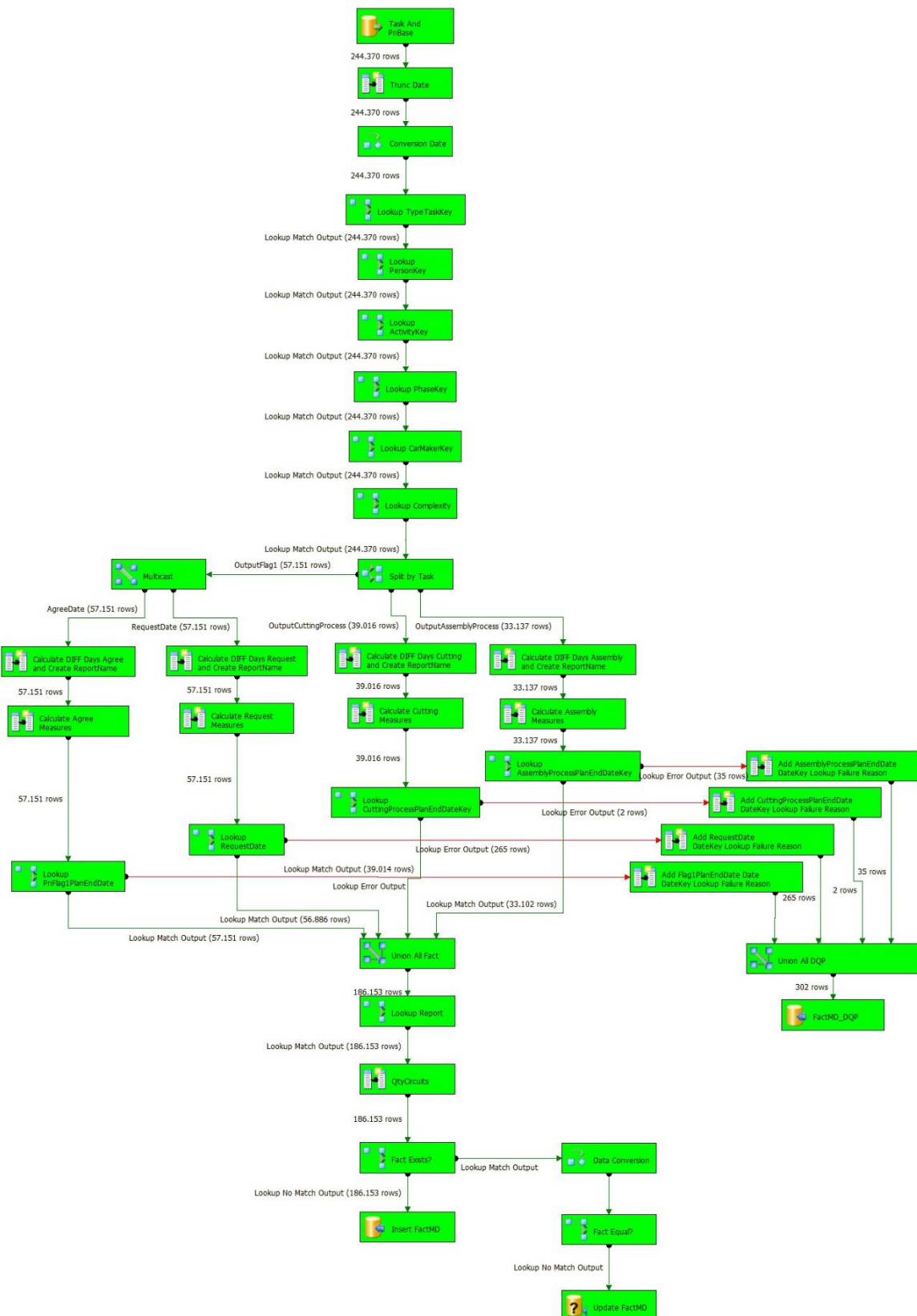


Figura 42 – Visão geral do fluxo da tabela de Factos

Importa referir que são utilizadas diversas variáveis que possibilitam copiar o projeto para outro computador e definir, de forma simples, o nome da DSA, do DM e dos servidores das bases de

dados. Neste projecto, foram criadas nove variáveis (figura 43). Para assegurar a leitura de todos os ficheiros necessários à criação da DSA e do DM, são usadas as variáveis *DMTableScriptFilename* e *SATableScriptFilename* nos componentes *Create DataMart Tables* e *Create Data Staging Tables*, respetivamente. A *RecordsCount* permite validar/controlar a existência de dados na dimensão *Date*. O *PackagePath* serve para indicar a localização do projeto no disco rígido.

Name	Scope	Data Type	Value
DMName	CIU_Datamart	String	Yazaki_DM
DMTableScriptFilename	CIU_Datamart	String	
PackagePath	CIU_Datamart	String	C:\Users\Yazaki\Documents\Visual Studio 2008\Projects\SSIS_Yazaki
RecordsCount	CIU_Datamart	Int32	0
SAName	CIU_Datamart	String	Yazaki_SA
SATableScriptFilename	CIU_Datamart	String	
ServerName_MHT_2	CIU_Datamart	String	YAZAKI-PC
ServerName_SA_and_DM	CIU_Datamart	String	YAZAKI-PC
ServerName_Whetool	CIU_Datamart	String	YAZAKI-PC

Figura 43 – Variáveis utilizadas no projeto

#### 4.6.2 Analysis Services Project

Nesta secção apresenta-se os procedimentos de criação do cubo de dados e algumas das análises feitas ao mesmo.

O SSAS permite gerar cubos de dados, sendo a quase totalidade do processo realizada através da execução de três *wizard's*:

1. *New Data Source*<sup>14</sup>;
2. *New Data Source View*<sup>15</sup>;
3. *New Cube*.

Nesta fase, o cubo contém apenas as chaves das tabelas de dimensões. Torna-se imprescindível adicionar os atributos existentes na *Data Source View*, considerados necessários a cada dimensão presente (ver figura 44).

<sup>14</sup> Data Source contém os dados de ligação à base de dados, nomeadamente *Provider*, *Server Name* e *Database Name*.

<sup>15</sup> Data Source View contém o modelo lógico do esquema usado pelo SSAS.

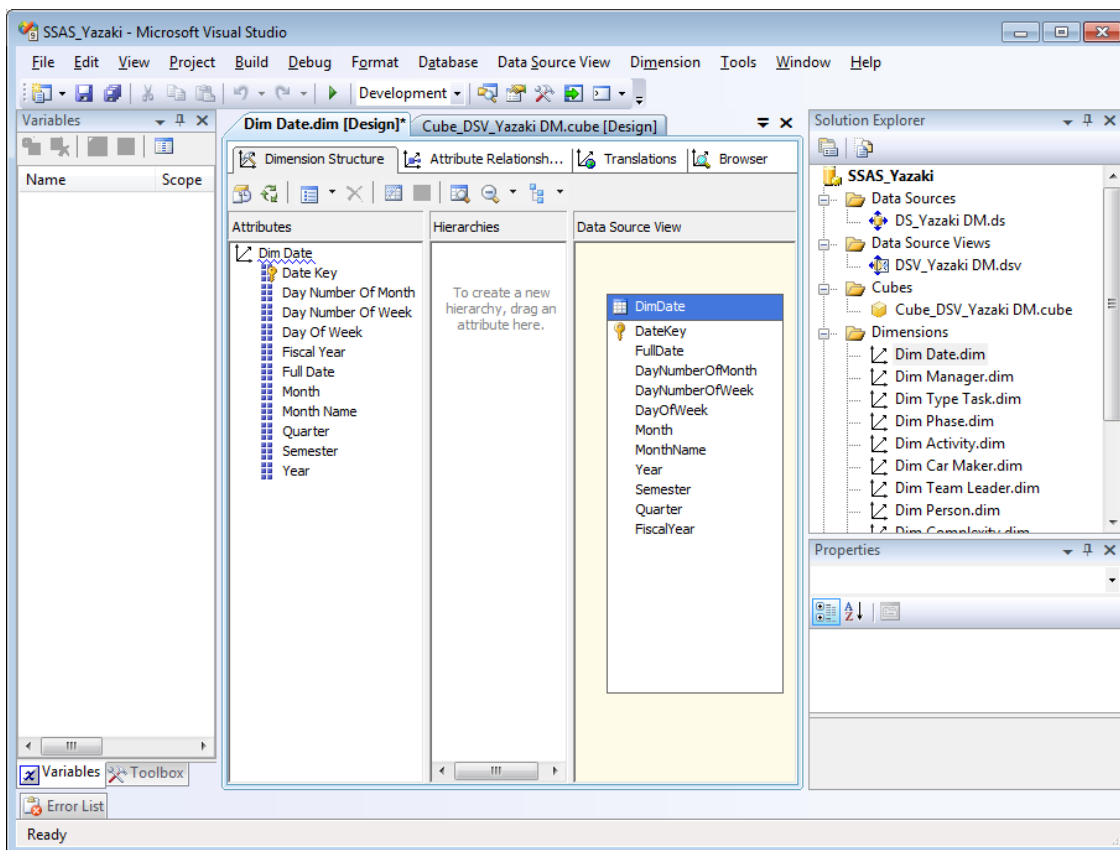


Figura 44 – Campos do cubo de dados

Uma vez criado o cubo, encontram-se reunidas as condições para a realização de diversas análises. Como exemplo destas, apresentam-se sete estudos:

1. A primeira análise refere-se aos valores das métricas na actividade MD, da tarefa *Cutting Process*, nos primeiros nove meses de 2014 (figura 45).

Dimension	Hierarchy	Operator	Filter Expression
Dim Activity	Name	Equal	{ MD }
Dim Report	Name	Equal	{ Cutting Process }
Dim Date	Year	Equal	{ 2014 }
<Select dimension>			

Drop Filter Fields Here											
	Year	Month	Month Name								Total
	2014	1	2	3	4	5	6	7	8	9	
	January	February	March	April	May	June	July	August	September		
Fields Here	Earlier Date										
	On Time Date										
	Delay Date										

Figura 45 – SSAS – Análise 1

2. No sistema existente na empresa, a visualização dos dados relativos às fases *RQF*, *Prototype* ou *MP* do processo produtivo, implica a mudança de folha do Excel. No sistema agora implementado, ao inserir a descrição dessas fases na exploração do cubo de dados, obtém-se os respetivos valores (figura 46).

		Year ▼	Month ▼	Month Name ▼							
		2014									
		1	2	3	4	5	6	7	8	9	Total
		January	February	March	April	May	June	July	August	September	
Name ▼											
RFQ	Earlier Date										
	On Time Date										
	Delay Date										
Prototype	Earlier Date										
	On Time Date										
	Delay Date										
MP	Earlier Date										
	On Time Date										
	Delay Date										

Figura 46 – SSAS – Análise 2

- Esta análise apresenta a evolução dos valores da métrica *OnTime*, nas fases *RQF*, *Prototype* e *MP*, ao longo dos meses (figura 47).

		Year ▼	Month ▼	Month Name ▼							
		2014									
		1	2	3	4	5	6	7	8	9	Total
		January	February	March	April	May	June	July	August	September	
Name ▼		On Time Date	On Time Date	On Time Date	On Time Date	On Time Date	On Time Date	On Time Date	On Time Date	On Time Date	On Time Date
RFQ											
Prototype											
MP											

Figura 47 – SSAS – Análise 3

- Para visualizar os valores de um determinado *Manager* e *Team Leader*, é necessário acrescentar as respetivas dimensões e selecionar os colaboradores pretendidos. Na figura 48, exibe-se uma consulta neste âmbito.

Dimension	Hierarchy	Operator	Filter Expression
Dim Activity	Name	Equal	{ MD }
Dim Report	Name	Equal	{ Cutting Process }
Dim Date	Year	Equal	{ 2013 }
Dim Manager	Manager Name	Equal	
Dim Team Leader	Team Leader Name	Equal	
<Select dimension>			

Drop Filter Fields Here	
Month Name ▼	December
Name ▼	Earlier Date   On Time Date   Delay Date
RFQ	
Prototype	
MP	

Figura 48 – SSAS – Análise 4

- Na figura 49 está patente o detalhe informativo referente ao mês de dezembro 2014, por colaborador. Esta análise possibilita verificar que todos os incumprimentos são relativos ao mesmo colaborador, com X entregas atrasadas, na fase de *Prototype*.

		Month Name ▾		
		December		
Name ▾	Core User Name ▾	Earlier Date	On Time Date	Delay Date
☐ RFQ				
	Total			
☐ Prototype				
		Total		
☐ MP				
		Total		
Grand Total				

Figura 49 – SSAS – Análise 5

6. A figura 50 mostra, em pormenor, as tarefas do colaborador referido na análise anterior. Verifica-se que os atrasos se confirmam unicamente na fase de *Prototype*, tendo sido entregues Y tarefas em cumprimento do prazo, na fase de *MP*.

		Month Name ▾			Grand Total		
		December					
Name ▾	Core User Name ▾	Earlier Date	On Time Date	Delay Date	Earlier Date	On Time Date	Delay Date
☐ Prototype							
	Total						
☐ MP							
	Total						
Grand Total							

Figura 50 – SSAS – Análise 6



7. Na análise da figura 51, adicionou-se a data de realização das tarefas e, também, o grau de complexidade das mesmas. Observa-se que as tarefas com entrega em incumprimento do prazo revelam uma complexidade *High*.

				Month Name ▾			
				December			
Name ▾	Core User Name ▾	Full Date ▾	Description ▾	Earlier Date	On Time Date	Delay Date	
▣ Prototype	▣	▣ 2013-12-02 00:00:00.000	Low	▣	▣	▣	
			Total	▣	▣	▣	
		▣ 2013-12-03 00:00:00.000	High	▣	▣	▣	
			Total	▣	▣	▣	
Total				▣	▣	▣	
▣ MP	▣	▣ 2013-12-09 00:00:00.000	Medium High	▣	▣	▣	
			Total	▣	▣	▣	
		▣ 2013-12-16 00:00:00.000	Medium High	▣	▣	▣	
			Total	▣	▣	▣	
		▣ 2013-12-20 00:00:00.000	Low	▣	▣	▣	
	Total	▣	▣	▣			
Total				▣	▣	▣	
Grand Total				▣	▣	▣	

Figura 51 – SSAS – Análise 7

Os exemplos apresentados nesta secção constituem algumas das análises passíveis de efectuar com recurso ao cubo de dados e serão detalhadas e exploradas na secção 4.6.4, com a utilização da ferramenta *Self-Service* da Microsoft, o PowerBI.

### 4.6.3 Report Server Project

Na presente secção aborda-se o potencial do *SQL Server Reporting Services*, que disponibiliza uma ampla gama de ferramentas e serviços para criar, implementar e gerir relatórios.

Um relatório do tipo *Report Server Project*, no *SQL Server Business Intelligence Development Studio*, inicia-se através de um *wizard* ou de um novo projeto. Na presente implementação, optou-se por este último, com a seguinte sequência procedimental: criação de um *data source* e de um *dataset*, através da opção *New Dataset...*

Ao gerar um *dataset*, a seleção de dados a constar do relatório pode ser executada com recurso a uma consulta utilizando a linguagem *MDX (Multidimensional Expressions)* ou através do *Query Designer*. Apresenta-se um exemplo de uma consulta em *MDX*:

```
SELECT
  NON EMPTY { [Measures].[On Time Date], [Measures].[Delay Date],
  [Measures].[Earlier Date] } ON COLUMNS,
  NON EMPTY { ([Dim Date].[Year].[Year].ALLMEMBERS ) } ON ROWS
FROM [Cube_DSV_Yazaki DM]
WHERE ( [Dim Activity].[Name].&[MD] )
```

Gerados o *datasource* e o *dataset*, segue-se a criação do relatório. Na *Toolbox* existente no lado esquerdo do ecrã, que contempla os *layouts* possíveis dos relatórios, seleciona-se os pretendidos. No caso em análise, e continuando a responder à necessidade da empresa de monitorizar a execução das tarefas do processo produtivo, escolheu-se uma *Matrix* e um *Chart*.



Quanto à primeira, na segunda coluna, foi acrescentado o campo *Year* e *Month\_Name* e, na terceira, o rótulo *Total*. A partir da terceira linha, foram incluídos os rótulos *Earlier (%)*, *On Time (%)*, *Delay (%)*, *Earlier (qty)*, *On Time (qty)*, *Delay (qty)* e *Total* (figura 52). Na tabela 21, apresenta-se as fórmulas associadas a cada uma das linhas da *Matrix* e na figura 53, exibe-se as configurações do gráfico.

	[Year]	Total	%
	[Month_Name]		
Earlier (%)	«Expr»		
On Time (%)	«Expr»		
Delay (%)	«Expr»		
Earlier (qty)	[Sum(Earlier_Date)]	[Sum(Earlier_Date)]	«Expr»
On Time (qty)	[Sum(On_Time_Date)]	[Sum(On_Time_Date)]	«Expr»
Delay (qty)	[Sum(Delay_Date)]	[Sum(Delay_Date)]	«Expr»
Total (qty)	«Expr»	«Expr»	

Figura 52 – SSRS – Tabela *Matrix*

Tabela 21 – SSRS – Fórmulas utilizadas no relatório

Campo	Expressão
Earlier (%)	=Sum(Fields! Earlier_Date.Value) / (Sum(Fields!On_Time_Date.Value) + Sum(Fields!Delay_Date.Value))
On Time (%)	=Sum(Fields!On_Time_Date.Value) / (Sum(Fields!On_Time_Date.Value) + Sum(Fields!Delay_Date.Value))
Delay (%)	=Sum(Fields!Delay_Date.Value) / (Sum(Fields!On_Time_Date.Value) + Sum(Fields!Delay_Date.Value))
Earlier (qty)	=Sum(Fields!Earlier_Date.Value)
On Time (qty)	=Sum(Fields!On_Time_Date.Value)
Delay (qty)	=Sum(Fields!Delay_Date.Value)
Total	=Sum(Fields!On_Time_Date.Value) + Sum(Fields!Delay_Date.Value)

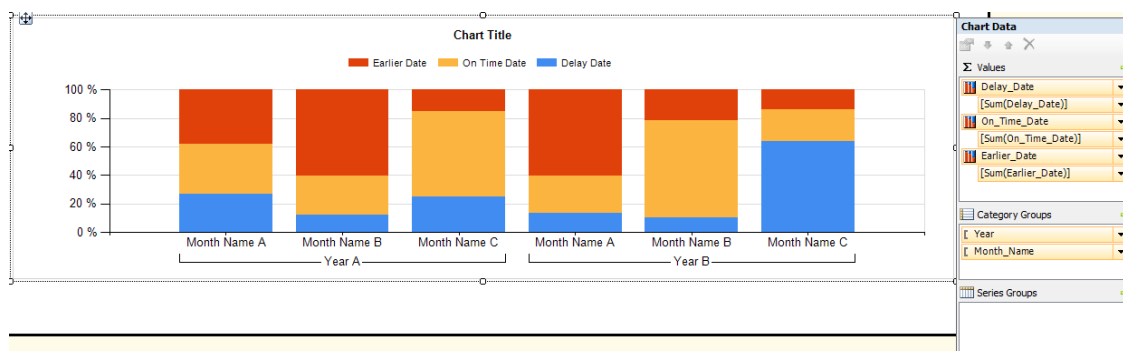


Figura 53 – SSRS – Gráfico

Concluída a construção do relatório, o *Preview* permite visualizar o resultado final e verificar da existência de eventuais erros. A partir do *View Report* é possível aceder, isoladamente, aos dados de cada dimensão.

Confirmada a inexistência de erros nos relatórios, a disponibilização destes aos utilizadores faz-se através do *Reporting Services*. Para esse efeito, operacionaliza-se o *Deploy* do projecto, no menu *Build*. Importa referir que, antes deste procedimento, é necessário verificar, nas propriedades do projeto de Relatórios, se o *TargetServerURL* está correto (figura 54).

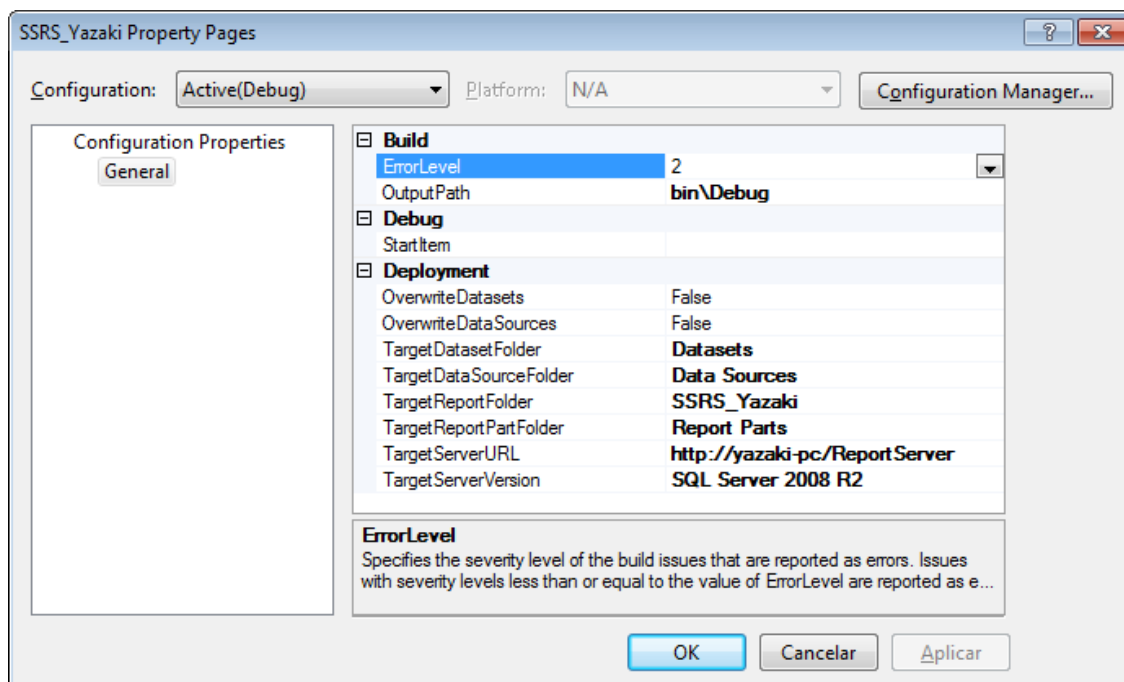


Figura 54 – SSRS – Propriedades do projeto

Executado o *Deploy* com sucesso, é possível aceder ao relatório através de um *browser* no endereço do *Report Server Web Service* - <http://yazaki-pc/ReportServer> ou do *Report Manager Site Identification* - <http://yazaki-pc/Reports>. Estes endereços estão apenas disponíveis na rede local da empresa e a sua identificação encontra-se na ferramenta de configuração *Reporting Services Configuration Connection*, no separador *Web Service URL* e *Report Manager URL*.

Na pasta *SSRS\_Yazaki* está disponível o relatório em questão. A visualização, total ou parcial, dos seus valores requer a escolha prévia das dimensões pretendidas. No exemplo da figura 55, mostra-se os dados relativos à actividade MD, num relatório do tipo *Agree Date*, de um determinado fabricante automóvel. O relatório pode ser exportado para diversos formatos como PDF, Word, Excel, entre outros.

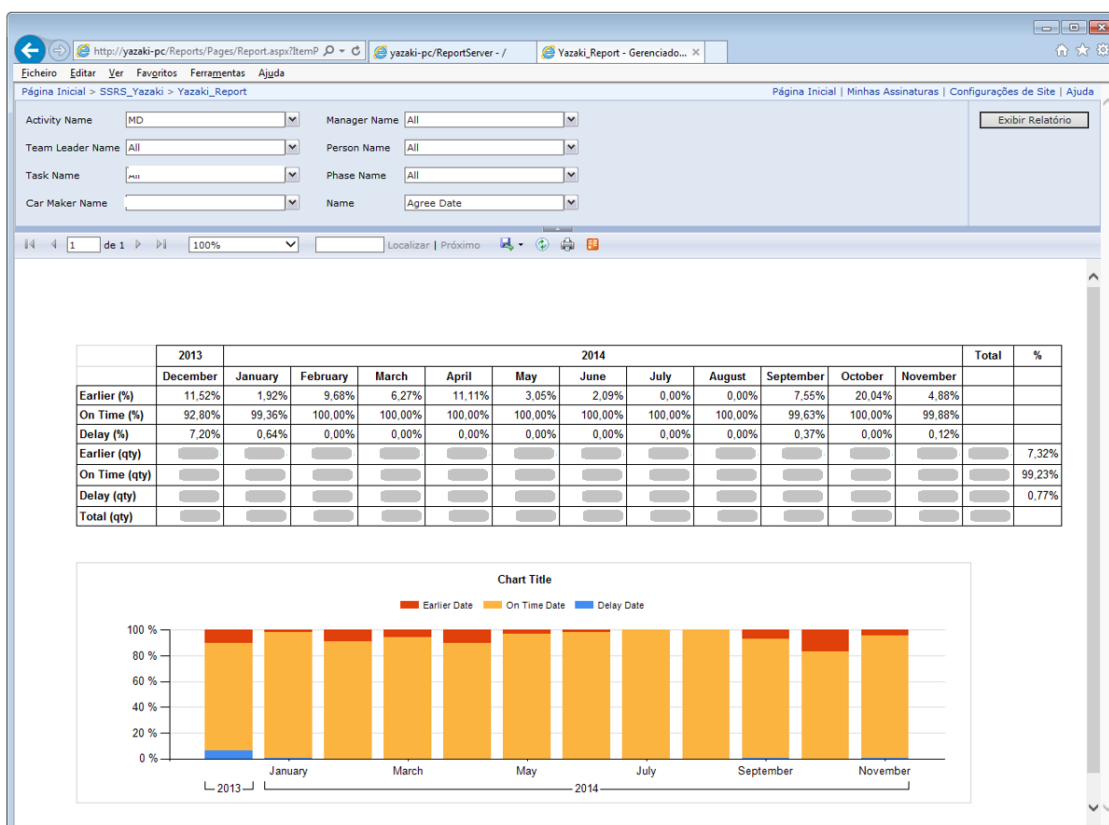


Figura 55 – SSRS – Relatório on-line

#### 4.6.4 PowerBI – Excel 2013

Nesta secção apresenta-se e aplica-se a ferramenta PowerBI, que constitui a resposta da Microsoft a uma nova tendência, o *Self-Service Business Intelligence*. Neste âmbito, foram desenvolvidos seis relatórios e quatro *dashboards*, referentes a tarefas da atividade produtiva da empresa. Estas análises pretenderam colocar em evidência a possibilidade de produzir diferentes abordagens, inter-relações e interpretações dos dados que se encontram referidos na secção 4.3 – Identificação do Processo de Negócio, Granularidade, Dimensões e Factos.

Nesta implementação, foi utilizado o Excel 2013. Para ser possível a utilização do PowerBI, o Excel tem que ser configurado em Gerir Suplementos COM.

##### 4.6.4.1 Importação de Dados

Para a elaboração dos relatórios e *dashboards* pretendidos, procedeu-se à importação da totalidade das tabelas presentes no DM, criado no SSIS (ver secção 4.6.1 – *Integration Services Project*).

##### 4.6.4.2 Modelo de Dados

Na sequência da importação de todo o DM, o *Power Pivot* assume, de imediato, um modelo de dados, exibindo as relações entre as tabelas importadas. São geradas folhas com todas as dimensões e uma folha com a tabela de factos. A visualização pode ser efectuada a partir da

Vista de Dados ou da Vista de Diagrama. Nesta fase, procede-se à confirmação se todas as relações estão criadas da forma correta. A figura 56 mostra o modelo de dados importado.

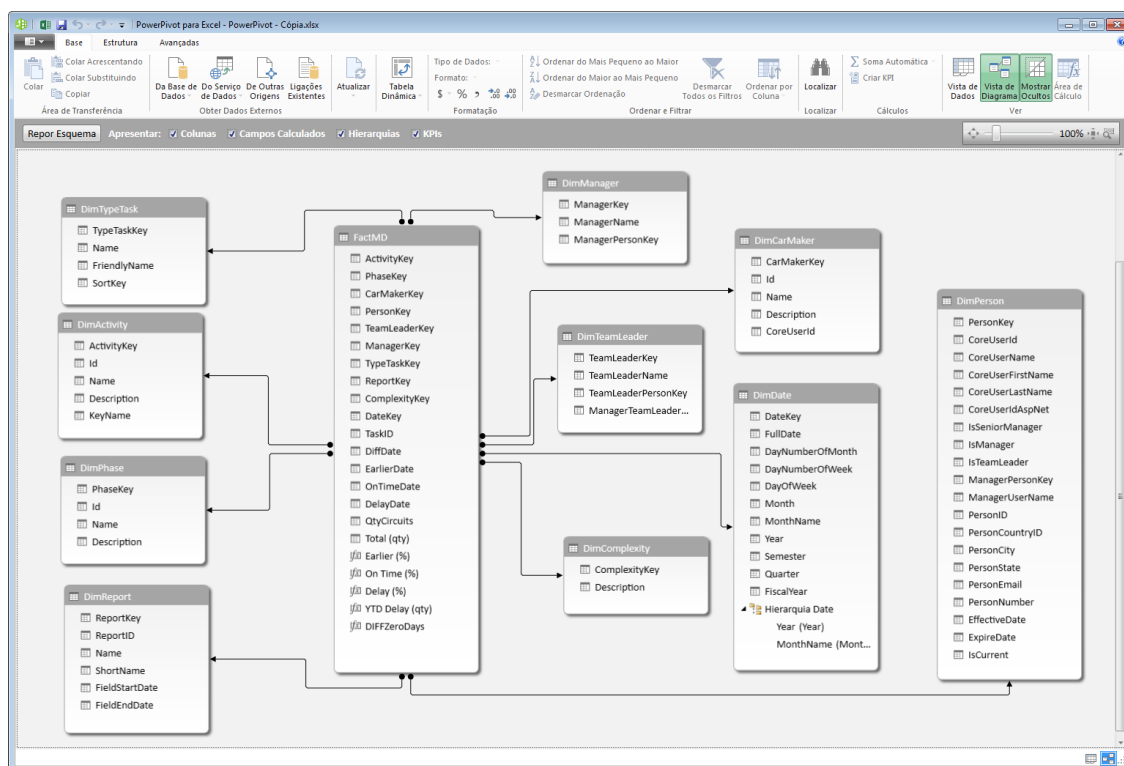


Figura 56 – Modelo de dados no *Power Pivot*

A utilização de hierarquias no *Power Pivot* permite criar relatórios, com execução do *drill-down* e *roll-up* dos dados. No caso em análise, foi gerada uma hierarquia na dimensão *DimDate*, que põe em relação o Ano e o Mês.

#### 4.6.4.3 Cálculos, Mediadas/Métricas e Time Intelligence com DAX

O modelo de dados criado pode ser enriquecido com novos cálculos e métricas, gerando os relatórios e *dashboards* pretendidos. Para o efeito, o *Power Pivot* usa as fórmulas DAX<sup>16</sup>. Desta forma, na tabela de factos acrescentou-se uma nova coluna, designada *Total (qty)*, que soma os valores presentes na coluna *OnTimeDate* e *DelayDate*. De um modo simplista, pode-se afirmar que os cálculos ocorrem quando são criadas novas colunas e as medidas são acrescentadas no final das tabelas, numa zona específica para o efeito, a Área de Cálculo. Na tabela 22, exhibe-se medidas criadas nesta Área.

<sup>16</sup> DAX (Data Analysis Expressions) - permite aos utilizadores realizar cálculos personalizados nas tabelas do PowerPivot (colunas calculadas) e nas Tabelas Dinâmicas do Excel (medidas).

Tabela 22 – Cálculos, Mediadas/Métricas e *Time Intelligence*

Formula	Formato
Earlier (%) := Sum([EarlierDate])/Sum([Total (qty)])	%
Delay (%) := Sum([DelayDate])/Sum([Total (qty)])	%
On Time (%) := Sum([OnTimeDate])/Sum([Total (qty)])	%
YTD_Delay_SamePeriodLastYear := CALCULATE(SUM(FactMD[DelayDate]); SAMEPERIODLASTYEAR(DimDate[FullDate]))	Geral
DIFFZeroDays := [Soma de OnTimeDate] - [Soma de EarlierDate]	Geral
YTD Delay (Geral) := TOTALYTD(SUM(FactMD[DelayDate]); DimDate[FullDate])	Geral
YTD_Delay_For_Report1 := CALCULATE(SUM(FactMD[DelayDate]); DATESYTD (DimDate[FullDate]; "2013-12-01"); ALL (DimDate))	Geral
DiffAnual := SUM([DelayDate]) - [YTD_Delay_SamePeriodLastYear]	Geral

Para facilitar a leitura, compreensão e interpretação do relatório 5, apresentado na figura 64, foram acrescentados indicadores de performance em duas das medidas, na *Delay (%)* e no *DiffAnual*.

Uma das potencialidades das fórmulas DAX é a possibilidade de incorporar funcionalidades de *Time Intelligence* (Inteligência Temporal). Permitem mostrar evoluções num período de tempo chamado *Date-Based Analysis* (Análises Baseadas em Datas). A implementação destas fórmulas implica a existência da dimensão temporal (DimDate). No caso em análise, foram usadas as funções *TOTALYTD*, *DATESYTD* e *SAMEPERIODLASTYEAR* cujo resultado está patente na figura 72. Este modelo de dados é dinâmico e evolutivo, podendo agregar novos cálculos e medidas.

#### 4.6.4.4 Análises com Tabelas Dinâmicas e Gráficos

Nesta secção são apresentados seis relatórios, com recurso à utilização de Tabelas Dinâmicas (*Pivot Tables*) e Gráficos.

A primeira análise, exibida na figura 57, dá uma visão geral das métricas presentes no DM. De forma a escolher-se o período temporal apresentado na tabela e no gráfico dinâmicos, foi adicionada uma linha cronológica<sup>17</sup> para filtrar, de forma interativa, as datas. Foram, também, adicionadas sete segmentações de dados<sup>18</sup> que permitem filtrar dados individualizados. No relatório 1, é apresentada uma tabela com os dados de dezembro de 2013 a novembro de 2014, relativos ao *Report Type - Agree Date*.

Nas colunas da tabela, exibem-se os rótulos ordenadamente por ano e mês; nas linhas são apresentadas as métricas *Earlier (%)*, *On Time (%)*, *Delay (%)*, *YTD Delay (qty)* e *Total (qty)*. É calculado o somatório das métricas presentes no DM: *EarlierDate*, *OnTimeDate* e *DelayDate*. De forma a tornar a visualização mais apelativa e intuitiva, foram introduzidos efeitos visuais

<sup>17</sup> As linhas cronológicas agilizam e facilitam a seleção de períodos de tempo na aplicação de filtros às tabelas dinâmicas, gráficos dinâmicos e funções cúbicas.

<sup>18</sup> As segmentações de dados simplificam e aceleram a filtragem de tabelas, tabelas dinâmicas, gráficos dinâmicos e funções cúbicas.

nas linhas do *EarlierDate* e *DelayDate*. Na métrica *EarlierDate* aparece uma barra verde e a dimensão da barra é tanto maior quanto maior for o valor das tarefas antecipadas. Na métrica *DelayDate* ocorre o mesmo, sendo a barra vermelha.

O gráfico utilizado é de colunas empilhadas a 100%. Este mostra as métricas *DelayDate*, *EarlierDate* e *DIFFZeroDays*. A utilização desta última métrica dispensa a presença da métrica *OnTimeDate*, que incorpora as tarefas antecipadas e as concluídas no dia previsto. Se esta métrica fosse utilizada no gráfico, produziria uma duplicação de valores, uma vez que as tarefas antecipadas surgiriam, simultaneamente, contabilizadas nas métricas *OnTimeDate* e *EarlierDate*.

Os gráficos gerados nas *Pivot Tables* exibem todos os dados presentes na tabela, não sendo possível isolar linhas. Por este motivo, foi criada uma nova folha de cálculo, com os dados necessários à apresentação do gráfico pretendido (ver figura 58). Esta folha está ligada à linha cronológica e às segmentações de dados constantes do relatório 1. Ao alterar os parâmetros cronológicos e/ou dos segmentos de dados, estes refletem-se nas duas tabelas e, consequentemente, no gráfico associado.



Figura 57 – *Pivot Tables and Pivot Chart* – Relatório 1

Rótulos de Linha	EarlierDate	DIFFZeroDays	DelayDate
2013			
December			
2014			
January			
February			
March			
April			
May			
June			
July			
August			
September			
October			
November			
Total Geral			

**FILTROS**

**LINHAS**

- Hierarquia Date
- MonthName

**COLUNAS**

Σ Valores

Σ VALORES

- EarlierDate
- DIFFZeroDays
- DelayDate

Figura 58 – Tabela de apoio ao relatório 1

No relatório 2 é mostrado, com recurso ao efeito visual Escala de Cor Progressiva, os valores acumulados mensalmente das métricas *DelayDate*, *OnTimeDate* e *EarlierDate* (figura 59). Na métrica *DelayDate*, os valores mais altos surgem a cor vermelha e os mais baixos a verde, existindo nos valores intermédios uma passagem progressiva do vermelho para o verde. No *EarlierDate*, o funcionamento das cores está invertido, sendo o verde para os mais altos e o vermelho para os mais reduzidos. A esta tabela foi acrescido, em cada métrica, um gráfico *SparkLine* do tipo Coluna. O primeiro gráfico identifica a vermelho o ponto mais alto e a verde o mais baixo. Nos restantes dois, esta identificação está invertida.

A título exemplificativo, seleccionou-se aleatoriamente um fabricante, sendo interessante observar que todas as segmentações associadas sofreram alterações, ou seja, é mostrado, de forma diferenciada, as segmentações que interagem com este fabricante.

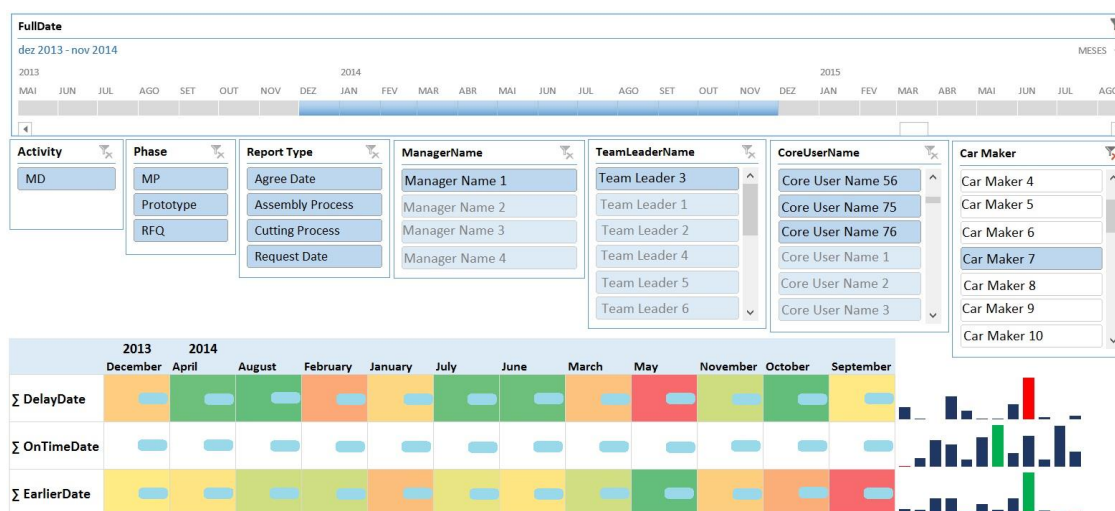


Figura 59 – Pivot Tables and Pivot Chart – Relatório 2

O relatório 3, exibido na figura 60, refere o acumulado diário de tarefas concluído em situação de incumprimento da data prevista.



Foi adicionado um conjunto de ícones visuais para melhor identificar os dias que, devido a um maior número de incumprimentos, devem ser analisados. Foi determinado que, se o valor diário fosse inferior a 10, o ícone seria verde, entre 10 e 25, surgiria a amarelo e a cor vermelha se ultrapassasse o número 25.

Para concretizar uma análise detalhada das tarefas em incumprimento, selecionou-se o dia que contemplava o maior número de atrasos. Ao adicionar um filtro no *Report Type*, do tipo *Agree Date*, o quantitativo total de tarefas em atraso diminuiu. Com um *Drill-down* da informação, obtém-se a identificação dos colaboradores que concluíram tarefas nesse dia. Desta forma, identifica-se qual ou quais os colaboradores que apresentam incumprimentos. É, ainda, possível aprofundar a análise, progressivamente, quanto ao grau de complexidade das tarefas, número de dias de atraso ocorrido e identificação das mesmas. Na figura 61, é mostrado o exemplo de um colaborador com atraso de um dia na entrega de X tarefas de complexidade baixa e Y de complexidade média. A análise ao nível da tarefa permite verificar a razão do atraso: fator externo ao planeamento, constrangimento operativo, ausência do colaborador, tarefa catalogada com uma complexidade desajustada, número de dias previsto para a execução desfasado da realidade etc. De referir que, no decurso da realização desta implementação, a empresa incluiu, na base de dados operacional, um campo descritivo que permite ao colaborador registar a razão do atraso.

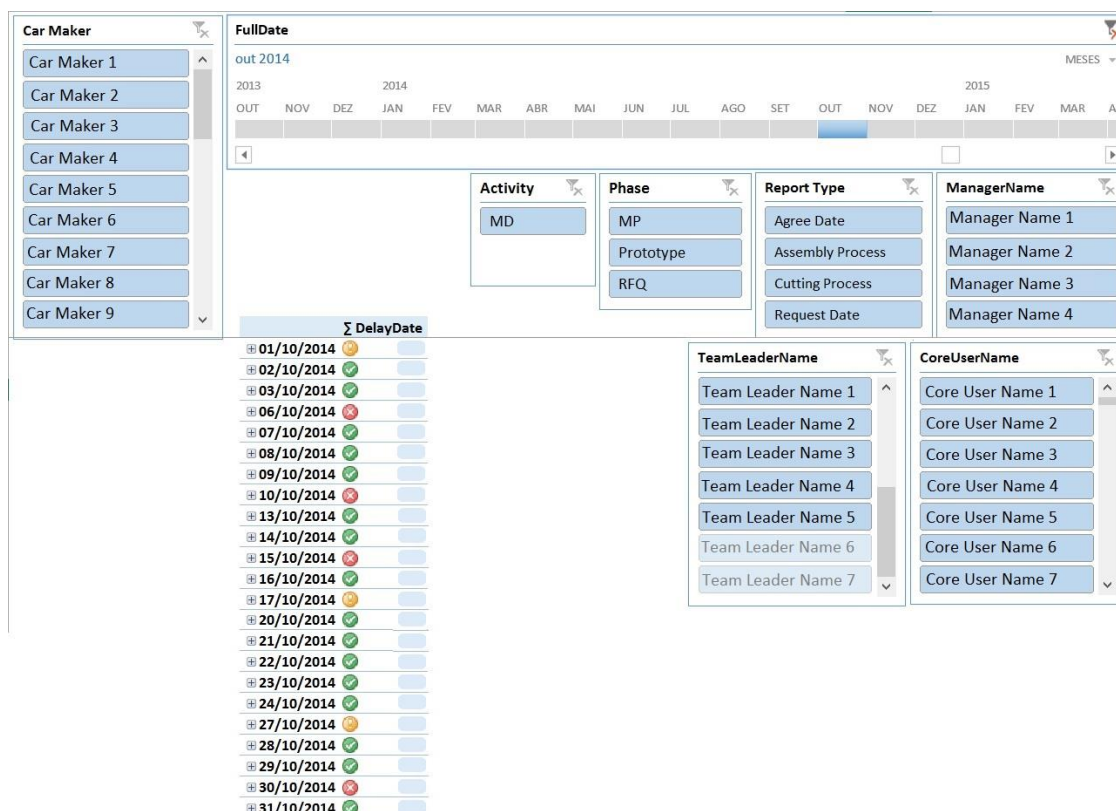


Figura 60 – Pivot Tables and Pivot Chart – Relatório 3



10/10/2014

- Core User Name 1
- Core User Name 2
- Core User Name 5
- Core User Name 8
  - Low
    - 1
      - {028A07F8-C1F8-453C-937D-2BC9763FCAB7}
      - {684A6D5B-82A9-4E47-AFC4-49E78C847976}
      - {750C55D8-AC1B-4AF2-9D88-A9A3EB8D355D}
    - Medium
      - 1
        - {0D7CEA43-A156-47C2-AEA1-8262F760802C}
        - {1B730339-B89F-45B1-A71F-70A468142995}
        - {1E139CDB-891B-48C1-ABD1-CBFFC819C7EF}
        - {36ECE533-C6CB-4CB3-9061-92E0336D4A74}
        - {511970E9-B7CD-4B2C-8D95-9088B708DD06}
        - {A2A8F9B7-69F1-4B0D-9EBC-354A0A19E413}
        - {C6F966E5-69D5-491C-BD18-DA79276691C9}
        - {C96B3A44-3A0C-48D1-8862-C6D1A45CC054}
        - {F25473DB-3581-45F2-932E-63234AED2085}
        - {FB4D9D61-91D2-4BEC-9CA8-B8F91C45D611}
  - Core User Name 10
  - Core User Name 16

Figura 61 – Detalhe Relatório 3

O relatório 4 exibe uma tabela que detalha as tarefas antecipadas por fabricante. Nas linhas, estão indicados todos os fabricantes; nas colunas apresenta-se: o valor acumulado de tarefas, o valor acumulado das tarefas antecipadas e a percentagem das tarefas antecipadas. Foi associado um gráfico em colunas, que mostra os valores percentuais (ver figura 62).

Rótulos de Linha	Task (qty)	EarlierDate	Earlier (%)
Car Maker 1			10,39%
Car Maker 2			5,66%
Car Maker 3			28,49%
Car Maker 4			8,21%
Car Maker 5			3,85%
Car Maker 6			13,67%
Car Maker 7			21,15%
Car Maker 8			9,51%
Car Maker 9			10,38%
Car Maker 10			13,28%
Car Maker 11			0,00%
Car Maker 12			2,13%
Car Maker 13			0,00%
Car Maker 14			14,53%
Car Maker 15			2,00%
Car Maker 16			4,66%
Car Maker 17			9,53%
Car Maker 18			7,24%
Car Maker 19			18,74%
Car Maker 20			11,75%
Car Maker 21			8,02%
Car Maker 22			5,26%
Car Maker 23			16,13%
Car Maker 24			12,03%
Car Maker 25			55,00%
Car Maker 26			10,16%
Car Maker 27			0,00%
Car Maker 28			9,92%
Car Maker 29			6,45%
<b>Total Geral</b>			<b>12,31%</b>

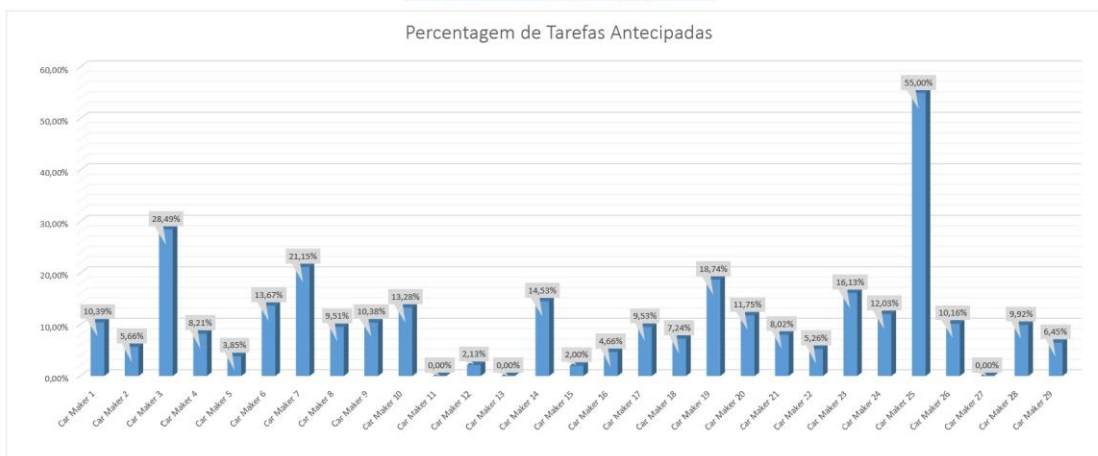


Figura 62 – Pivot Tables and Pivot Chart – Relatório 4

Foi adicionado, ainda, um KPI visual, com barras crescentes, na coluna *Earlier (%)*. Os ícones são mostrados conforme as regras indicadas na figura 63.

Mostrar cada ícone de acordo com estas regras:

Ícone	Regra	Operador	Valor	Tipo
	quando o valor for	>=	0,5	Número
	quando < 0,5 e	>=	0,3	Número
	quando < 0,3 e	>=	0,2	Número
	quando < 0,2 e	>=	0,1	Número
	quando < 0,1			

Figura 63 – Regras do relatório 4

No relatório 5 (figura 64), é efetuada uma comparação do número de tarefas em incumprimento, em períodos homólogos. À data desta implementação, não se encontrava disponível um quantitativo de dados que permitisse uma análise comparativa superior a 3 meses. Nesta, foram utilizadas as funções *DAX Time Intelligence*.

A tabela mostra, na primeira coluna, o ano e, na seguinte, o mês ordenado cronologicamente. Na terceira coluna, através de uma métrica presente na tabela 22, é apresentado o somatório de *Delay's* referentes ao período homólogo do ano anterior. A quarta coluna apresenta os *Delay's* referente ao período em curso. Seguem-se duas colunas alusivas à diferença entre o mês corrente e o homólogo do ano anterior. Sob a forma de KPI, apresenta-se um apontamento visual a verde se o número de *Delay's* diminui, a amarelo se os valores forem iguais e a vermelho se o valor do período em curso for superior ao homólogo do ano anterior. As duas colunas seguintes são referentes à percentagem de *Delay's* em relação ao total de tarefas do período em curso. É igualmente apresentado um KPI, com os seguintes *targets*: a verde os valores inferiores a 5%, a amarelo os situados entre 5% e 20% e a vermelho os superiores a 20% (figura 65). Na última coluna, é exibido o valor acumulado por ano, através de uma métrica também presente na tabela 22.

Report Type	Year	MonthName	Delay's Same Period Last Year	Σ Delay's in Period	Δ Period	Δ Period Performance	Delay (%)	Delay (%) Performance	YTD Delay's
<ul style="list-style-type: none"> <li>Agree Date</li> <li>Assembly Process</li> <li>Cutting Process</li> <li>Request Date</li> </ul>	2013	September					8,14%		
		October					0,00%		
		November					4,36%		
		December					11,53%		
		December					7,92%		
	2014	January					5,72%		
		February					7,43%		
		March					20,90%		
		April					2,77%		
		May					2,82%		
		June					6,21%		
		July					7,66%		
		August					2,43%		
September					6,34%				
October					2,17%				
November					1,59%				
December					1,99%				

Figura 64 – Pivot Tables and Pivot Chart - Relatório 5

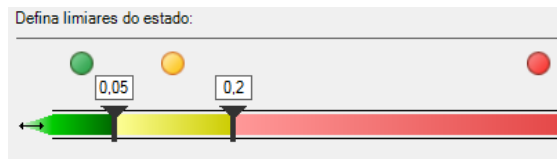


Figura 65 – Target's do relatório 5

O relatório 6 revela, para cada mês, o número médio de dias de incumprimento das tarefas. Esta análise pode ser extensível às tarefas concluídas antecipadamente.

São apresentadas duas colunas, uma com o valor da média do número de dias de incumprimento e outra com o mesmo valor, mas em idêntico período do ano anterior. Nas colunas está realçado a vermelho o valor mais elevado e a verde o mais baixo. É apresentado um gráfico representativo desta variação mensal, com uma combinação de Colunas Agrupadas com Áreas Empilhadas (ver figura 66).

Year	MonthName	Average Delays (Days)	Same Period Last Year
2013	October	2,37	
	November	4,94	
	December	7,60	
2014	January	2,13	
	February	4,21	
	March	1,81	
	April	3,58	
	May	5,13	
	June	3,13	
	July	2,50	
	August	1,01	
	September	2,39	
	October	2,13	2,37
	November	1,28	4,94
	December		7,60

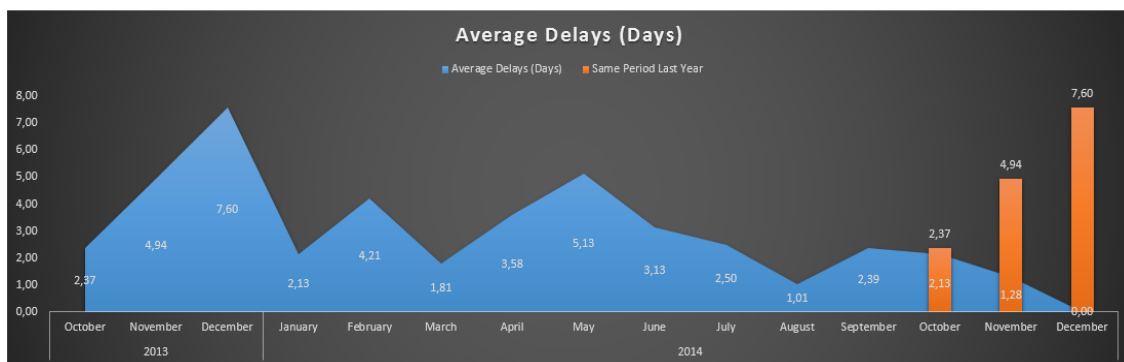


Figura 66 – Pivot Tables and Pivot Chart – Relatório 6

#### 4.6.4.5 Análises com Power View

Nesta secção são apresentados quatro dashboards, com recurso à utilização do Power View.

No dashboard Total de Tarefas (figura 67), mostra-se um gráfico com os totais de tarefas por cada Manager e uma tabela. Nesta, e para cada Manager, é exibido o nome dos TeamLeader's, as métricas EarlierDate, OnTimeDate, DelayDate e respetivas percentagens.

É possível efetuar uma ordenação por qualquer coluna existente. Uma funcionalidade de relevante interesse é a que possibilita isolar os elementos presentes no dashboard. Sejam tabelas ou gráficos, todos dispõem de um botão no canto superior direito que permite expandir os elementos, de forma a ocuparem toda a área, facilitando a sua leitura.

## Total de Tarefas

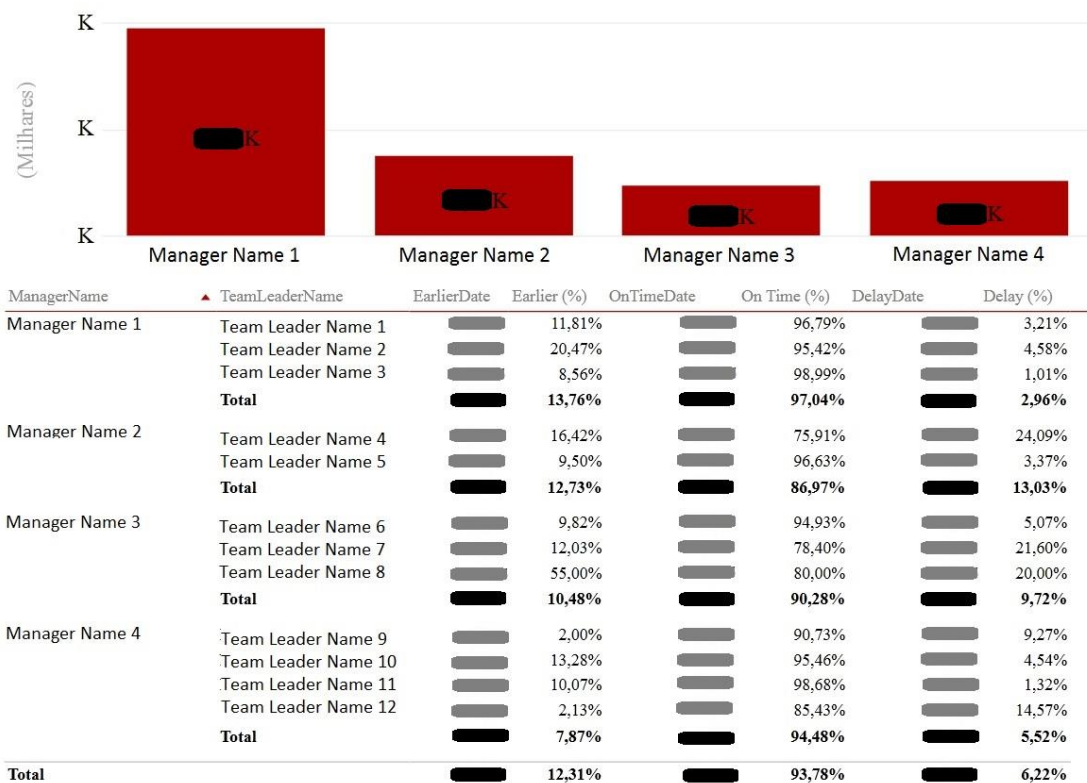


Figura 67 – Power View – Dashboard Total de Tarefas

Ao clicar sobre um dos *Manager*, é automaticamente feito um filtro, sendo só mostrados, na tabela, os dados relativos a esse *Manager* e, no gráfico, surge evidenciado por relação aos restantes *Managers* (ver figura 68).

## Total de Tarefas

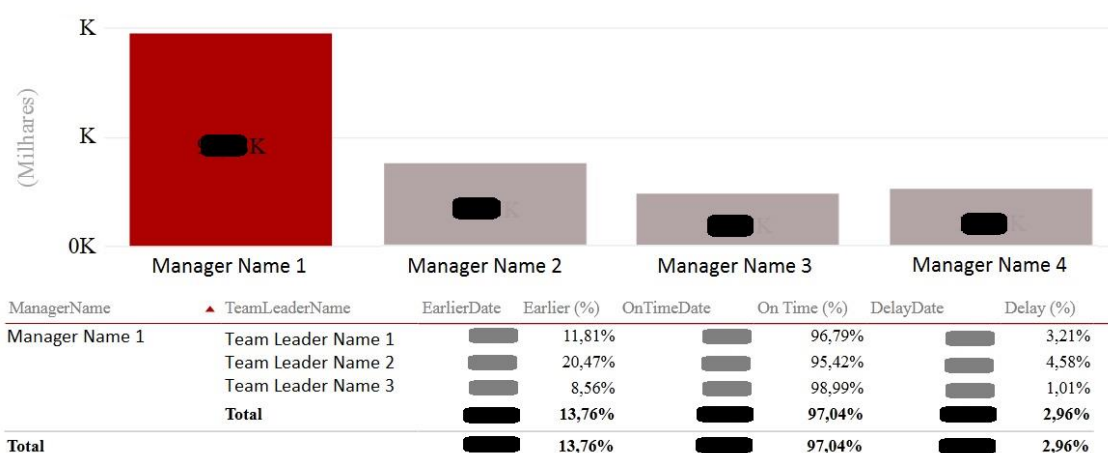


Figura 68 – Detalhe do Dashboard Total de Tarefas

Uma outra possibilidade é a execução de um *Slice and Dice*. Na figura 69, mostra-se o mesmo *dashboard*, mas unicamente com dados relativos ao ano de 2013.

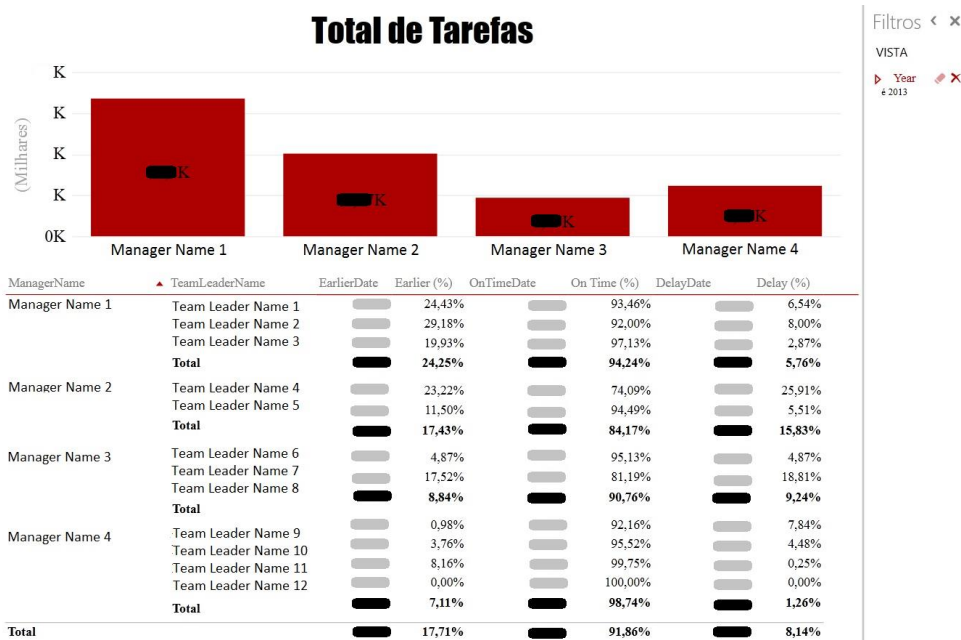
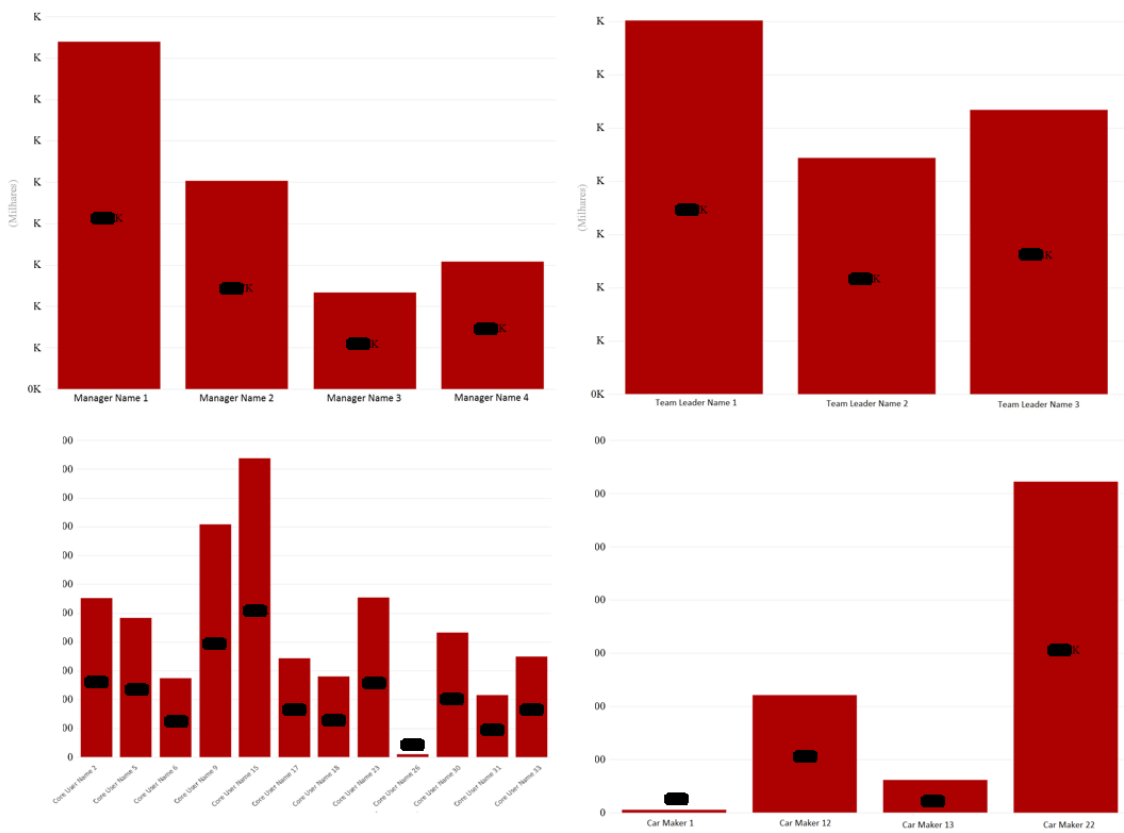


Figura 69 – Slice and Dice – Dashboard Total de Tarefas

Como referido na secção 4.6.4.2, é possível criar hierarquias na vista de diagrama do *Power Pivot*. Também o *Power View* permite criar hierarquias, com a vantagem de incorporar campos de diferentes dimensões. Assim, procedeu-se à criação de uma hierarquia, com o nome dos *Manager*, *TeamLeader*, *CoreUser* e *CarMaker*, exemplificada na figura 70. Esta exhibe, sucessivamente, quatro gráficos de colunas, à medida que se efetua o *drill-down* dos dados. A seleção de cada *Manager* apresenta os respetivos *TeamLeaders* e com a seleção de cada um destes acede-se aos dados de todos os colaboradores agregados, que executam tarefas geridas por cada *TeamLeader*. É, ainda, possível detalhar mais um nível, a partir do colaborador, identificando-se os fabricantes que lhe estão adstritos e o número de tarefas executado.



Drill-down : Manager Name 1 – Team Leader Name 3 – Core User Name 6

Figura 70 – Detalhes do Dashboard Total de Tarefas

O *dashboard* Geral, apresentado na figura 71, refere-se à execução de tarefas. É composto por seis elementos: um título, uma tabela, dois gráficos de colunas, um gráfico circular e outro de colunas agrupadas. Na tabela exibem-se os somatórios das tarefas antecipadas, concluídas no dia previsto e as terminadas fora da data prevista. O primeiro gráfico revela a percentagem de tarefas antecipadas por fabricantes. O segundo mostra o total de tarefas realizado por fabricante. O gráfico circular assinala as tarefas totais antecipadas, as terminadas no prazo previsto e as terminadas fora do prazo. Por fim, surge um gráfico com colunas agrupadas, com as mesmas métricas do gráfico circular, mas com uma agregação de dados por *Manager*.

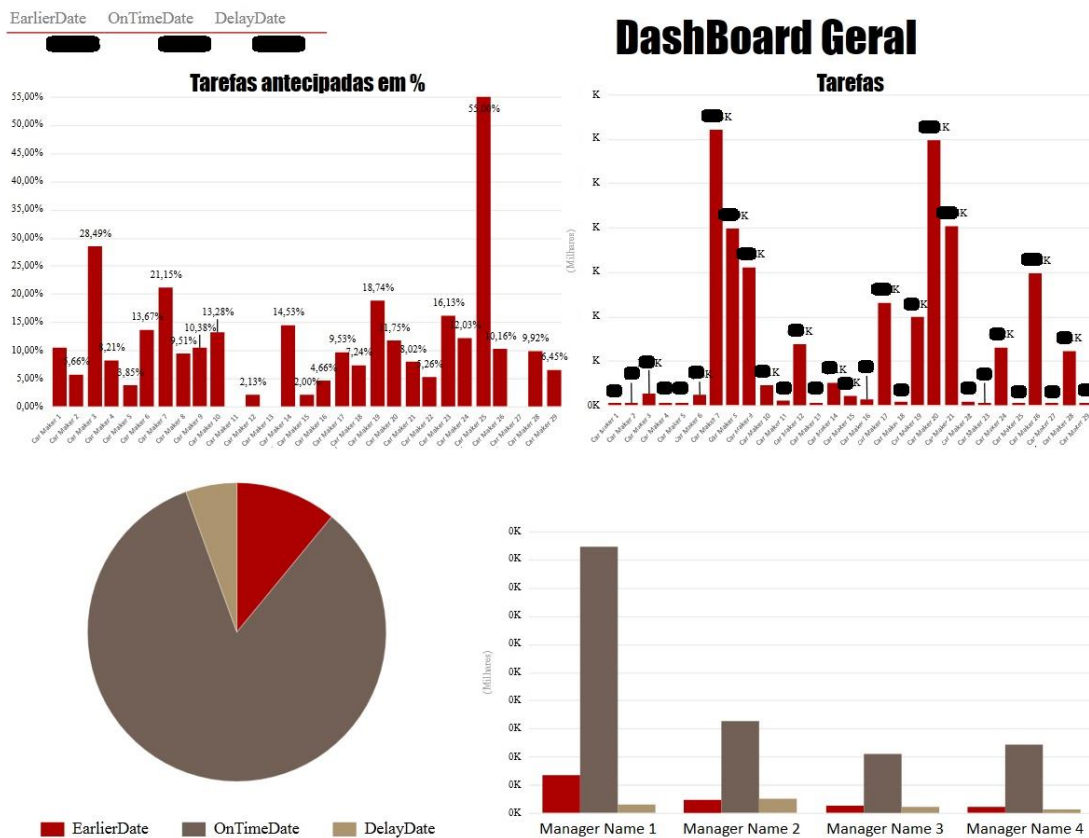


Figura 71 – Power View – Dashboard Geral

É possível obter informação detalhada sobre a execução das tarefas, a partir da seleção isolada de cada dado. O exemplo exibido na figura 72 mostra o fabricante com maior número de tarefas. Ao selecioná-lo, todos os valores da tabela e dos gráficos se alteram, possibilitando verificar qual é o *Manager* associado e a situação das tarefas que lhe dizem respeito. A figura 73 exemplifica uma outra seleção isolada, a partir de um dado *Manager*.

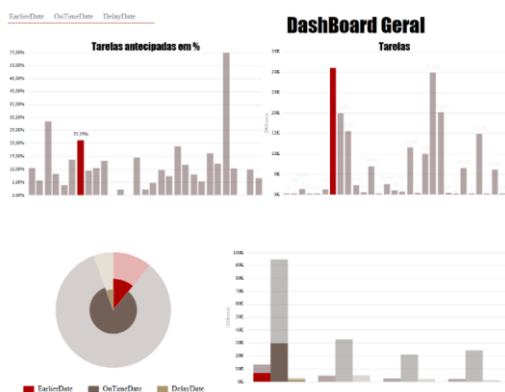


Figura 72 – Detalhe 1 – Dashboard Geral

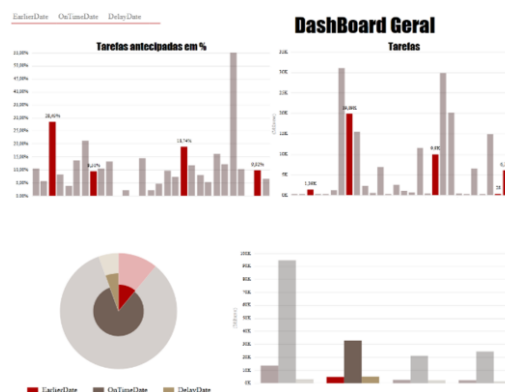


Figura 73 – Detalhe 2 – Dashboard Geral

O *dashboard* permite, também, a seleção simultânea de vários dados. Na figura 74 encontram-se selecionados os cinco fabricantes com mais tarefas.

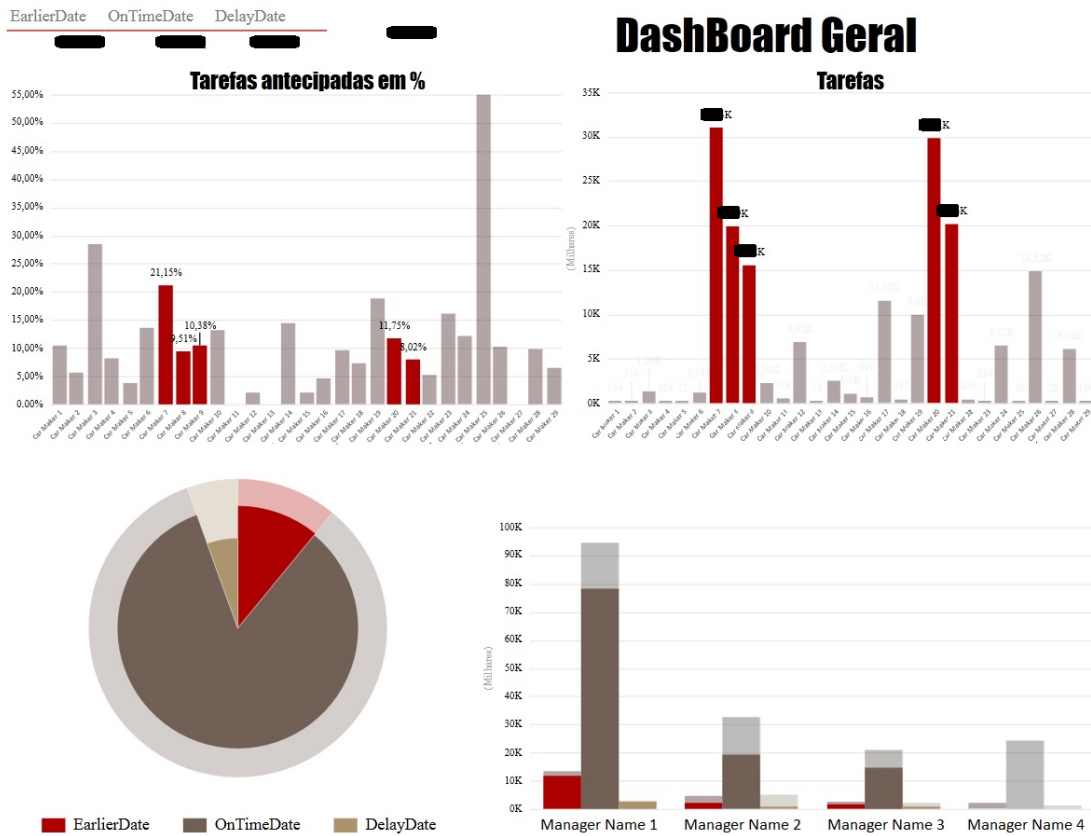


Figura 74 – Detalhe 3 – *Dashboard Geral*

Uma outra funcionalidade consiste em isolar o gráfico circular, adicionando, na secção Múltiplos Verticais, um determinado campo, de qualquer dimensão. A figura 75 mostra a seleção do campo nome do fabricante. De forma automática, é gerado um gráfico circular por cada um.

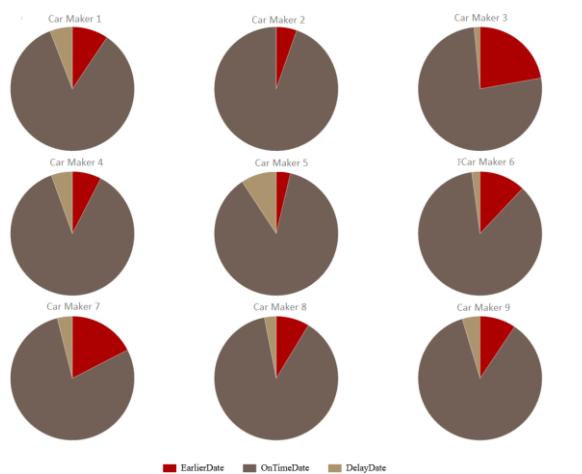


Figura 75 – Detalhe 4 – *Dashboard Geral*



O *dashboard* Dispersão das tarefas apresenta um gráfico de dispersão, com os valores mensais de tarefas executadas por ano. São exibidos círculos coloridos, nos quais quanto maior for o tamanho do círculo, maior é o valor de tarefas desse mês. Foi acrescentado um eixo de reprodução com os meses, que permite ver, de forma animada, a evolução mensal dos valores, ao longo do ano (ver figura 76).

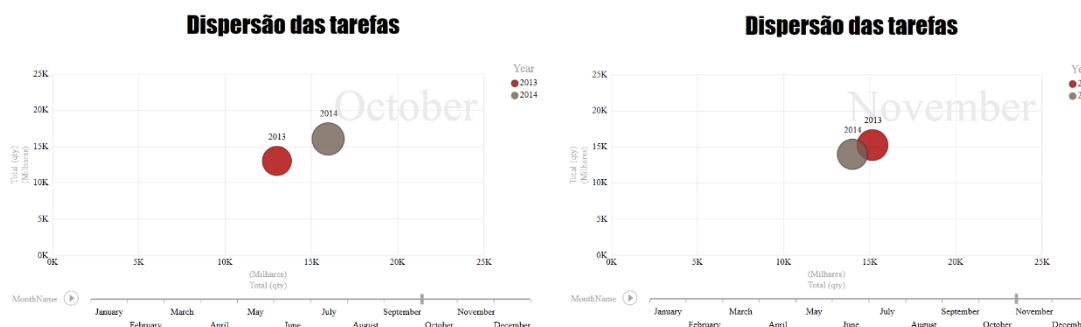


Figura 76 – Power View – Dashboard Dispersão das tarefas

No *dashboard* 4 mostra-se um mapa com a localização geográfica da residência dos colaboradores da empresa. A apresentação deste mapa implicou a seguinte sequência procedimental: crescer, na dimensão *Person* do modelo de dados, o campo *City*; arrastar este campo para a área de trabalho do *dashboard* e selecionar o botão Mapa, presente no separador Estrutura. Tal apenas é exequível com ligação à Internet.

No mapa, observa-se que o local de residência, que compreende o maior número de colaboradores, encontra-se identificado com o círculo de maior dimensão. Para enriquecer o *dashboard*, é possível sobrepor um gráfico de barras que, de forma decrescente, ordene o número de colaboradores pelos respetivos locais de residência; por questões de confidencialidade o gráfico não é apresentado na figura 77.

### Zona de Residência dos Colaboradores

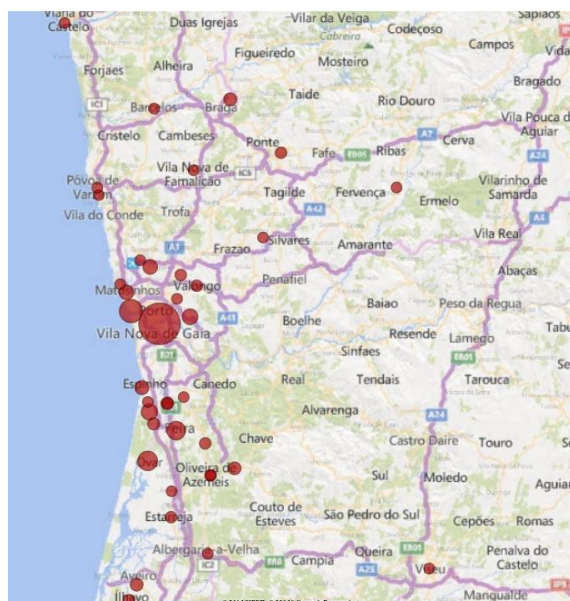


Figura 77 – Power View – Dashboard 4





## 5 Avaliação Experimental do SpagoBI e PentahoBI

No seguimento da análise comparativa das plataformas BI, efetuada no capítulo 3, decidiu-se pela experimentação de duas plataformas BI *open source*, a SpagoBI e PentahoBI. A escolha destas plataformas fundamentou-se no facto de serem duas das três que obtiveram pontuação mais elevada no processo de validação das suas funcionalidades, patente na tabela 6, da secção 3.3.2. A terceira plataforma, a Vanilla, à data da análise efetuada apresentava uma versão que não oferecia total estabilidade, em virtude das recentes alterações à sua arquitetura, reestruturada numa diferente linguagem de programação.

Tendo havido a oportunidade de implementar tecnologias associadas ao conceito BI, de carácter comercial, em contexto real, na Yazaki Saltano de Ovar Produtos Eléctricos, Ltd., afigurou-se ser relevante testar as potencialidades concretas de plataformas *open source*, no sentido de compreender se podem constituir uma efetiva alternativa às plataformas comerciais. Procurou-se reproduzir, utilizando as plataformas *open source* acima identificadas, alguns dos resultados obtidos no projeto implementado na Yazaki Saltano, os quais assentaram na utilização das ferramentas disponibilizadas na plataforma BI da Microsoft, que constituía a *stack* tecnológica da empresa.

Os critérios qualitativos subjacentes à análise comparativa realizada compreendem:

- Grau de acessibilidade e usabilidade das ferramentas - refere-se à capacidade dos utilizadores identificarem, localizarem e executarem as funcionalidades, bem como se apresenta um interface agradável, intuitivo e ergonómico;

- Nível de desempenho da ferramenta de ETL - reporta-se à capacidade de extração, tratamento e carregamento de dados de diferentes proveniências;
- Utilização da ferramenta OLAP - apreciação da simplicidade e desempenho da criação de um cubo de dados OLAP, assim como, da sua exploração.

Este procedimento de avaliação, embora assuma uma dimensão simplificada, tem por referência a norma internacional de avaliação da qualidade de produtos de *software*, a ISO/IEC 14598-4 [58], na ótica do comprador do produto e a ISO/IEC 9126-1 [59].

No que respeita ao método avaliativo, cada critério é classificado em função de uma escala qualitativa que compreende:

- **Bom**, sempre que a funcionalidade se revelar eficiente, excedendo o que era expectável;
- **Suficiente**, quando a funcionalidade manifestar um desempenho adequado aos seus propósitos;
- **Insuficiente** quando a funcionalidade não corresponder ao esperado e/ou não cumprir o mínimo exigido.

Qualquer processo de avaliação não se encontra isento de subjetividade. Não sendo possível garantir a total neutralidade do ato de avaliar, definem-se apenas três níveis de classificação que permitem delimitar o mais possível a leitura do desempenho das ferramentas. Desta forma, pretende-se minimizar o risco de juízos de valor ao focalizar a avaliação em aspetos concretos dos resultados. Sempre que cada um dos critérios avaliados compreenda vários requisitos para validação, será aplicada a escala classificativa de Bom/Suficiente/Insuficiente, resultando a avaliação do critério em si numa apreciação geral dos respetivos requisitos. A cada nível de classificação corresponde a seguinte escala de pontuação: Bom – 3 pontos, Suficiente – 2 pontos e 1 – Insuficiente.

O presente capítulo inicia com o processo de instalação das plataformas SpagoBI e a PentahoBI, seguindo-se o seu teste, comparação e súmula conclusiva. As principais funcionalidades operacionalizadas são o processo de ETL e a análise dos cubos de dados. Esta última tem por referência os resultados apresentados no capítulo 4.6.2, relativos ao projeto de implementação na empresa Yazaki Saltano.

## 5.1 Instalação e configuração do servidor

Procedeu-se ao *download*, instalação, configuração das plataformas, bem como ao teste das bases de dados nelas incorporadas. Este teste incidiu na usabilidade das plataformas e na visualização de relatórios, *dashboards* e análises OLAP. Para tal, recorreu-se a duas máquinas virtuais idênticas, com sistema operativo *Windows 7*; 4 GB de RAM e *SQL Server 2008 R2*.

### 5.1.1 SpagoBI

Uma vez efetuado o *download* do ficheiro *All-In-One-SpagoBI-4.1*, no respetivo site, procedeu-se à descompactação do mesmo. Previamente, foi instalado o *Java Development Kit (JDK)*<sup>19</sup> versão 6, e criadas as variáveis de sistema *CATALINA\_HOME* e *JAVA\_HOME*, na opção “Variáveis do ambiente”, presente nas “Definições Avançadas do Sistema”.

É possível aceder e testar a plataforma através de um *browser*, no endereço <http://localhost:8080/SpagoBI>. No ecrã inicial da plataforma, efetua-se o *login*, através de um *username* e *password* (ver figura 79).

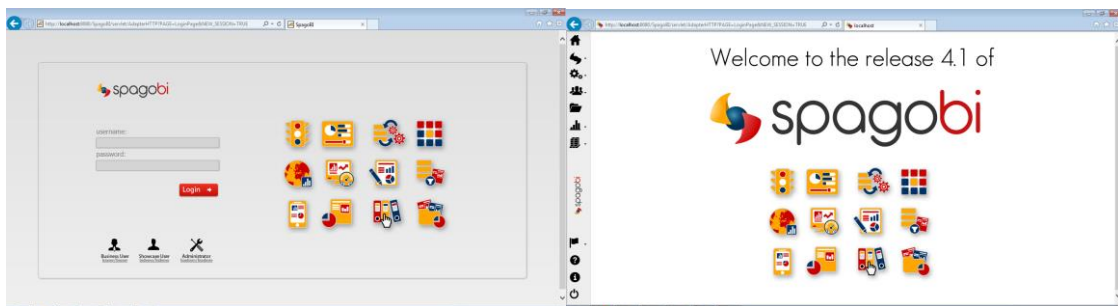


Figura 79 – SpagoBI – Ecrã de *login* e ambiente de trabalho inicial

A figura 80 mostra alguns exemplos de testes às diversas funcionalidades presentes na plataforma (navegabilidade, gráficos, *dashboards*, KPI's, OLAP e tabelas). Afigura-se que é de fácil utilização e acesso a todos os seus recursos.

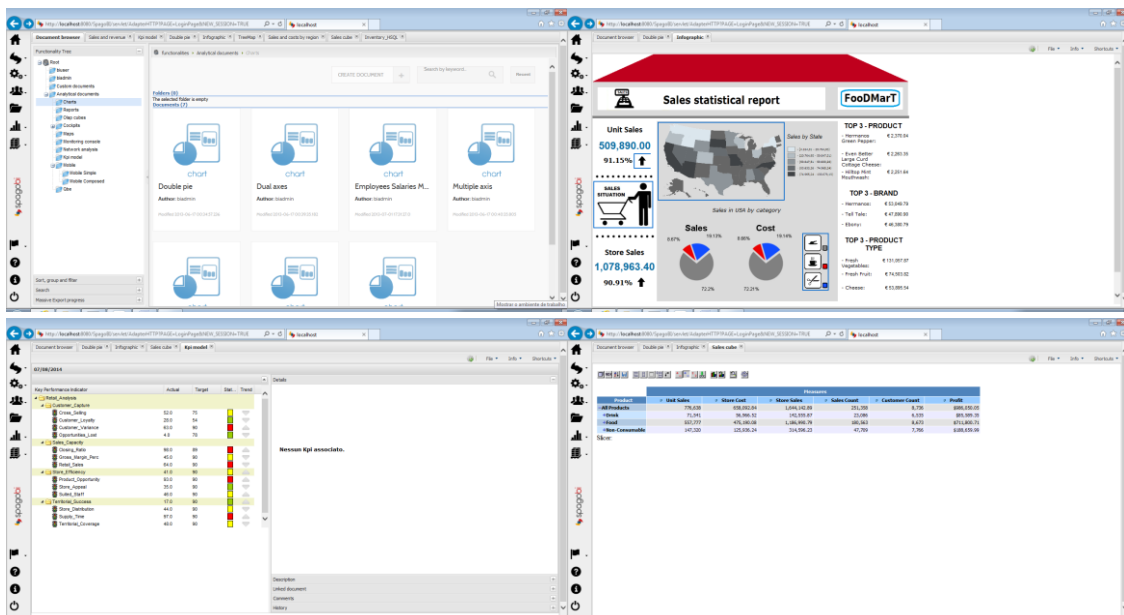


Figura 80 – Gráficos, *Dashboards*, KPI's e OLAP

<sup>19</sup> *Java Development Kit* é um ambiente de desenvolvimento para programar aplicações em *Java*.

### 5.1.2 PentahoBI

No que diz respeito à plataforma da Pentaho, os requisitos mínimos de *hardware* são 2GB de memória RAM, 1 GB de disco rígido e um processador de duplo core. Esta plataforma pode ser instalada em ambiente *Windows* (*Windows XP with Service Pack 2* e posteriores) e as versões mais recentes funcionam também em Linux, Solaris 10 e Mac OS X 10.4. Independentemente do SO, é necessário a instalação *Java Runtime Environment (JRE)*<sup>20</sup> *version 1.5*.

A versão instalada foi a 5.0.1., tendo sido necessário, à semelhança do procedimento desenvolvido na SpagoBI, criar a variável *JAVA\_HOME*. A plataforma pode ser acedida através de um *browser* no endereço <http://localhost:8080/pentaho> (ver figura 81).

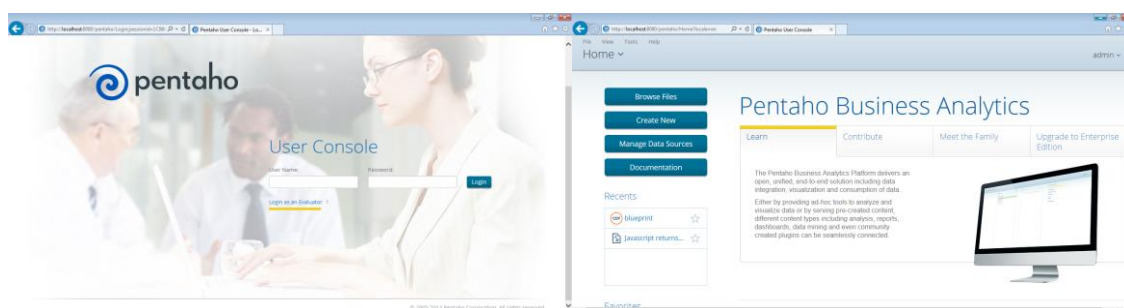


Figura 81 – PentahoBI – Ecrã de *login* e ambiente de trabalho inicial

O ambiente de trabalho da plataforma é de utilização intuitiva, sendo possível, no menu *View*, mudá-lo para outro *template* gráfico. Na funcionalidade *MarketPlace*, é permitida a mudança do idioma dos menus, encontrando-se disponível a opção pelo português.

Através da base de dados existente, visualiza-se algumas das funcionalidades possíveis, tais como: análises, *dashboards*, integração de dados e relatórios.

A navegabilidade da plataforma caracteriza-se por uma prática e acessível disposição dos conteúdos, através de separadores (ver figura 82).

---

<sup>20</sup> *Java Runtime Environment* é um ambiente desenvolvido pela *Oracle*, que permite a execução de *softwares* desenvolvidos em *Java*. Integra bibliotecas de desenvolvimento e a *Java Virtual Machine (JVM)*.

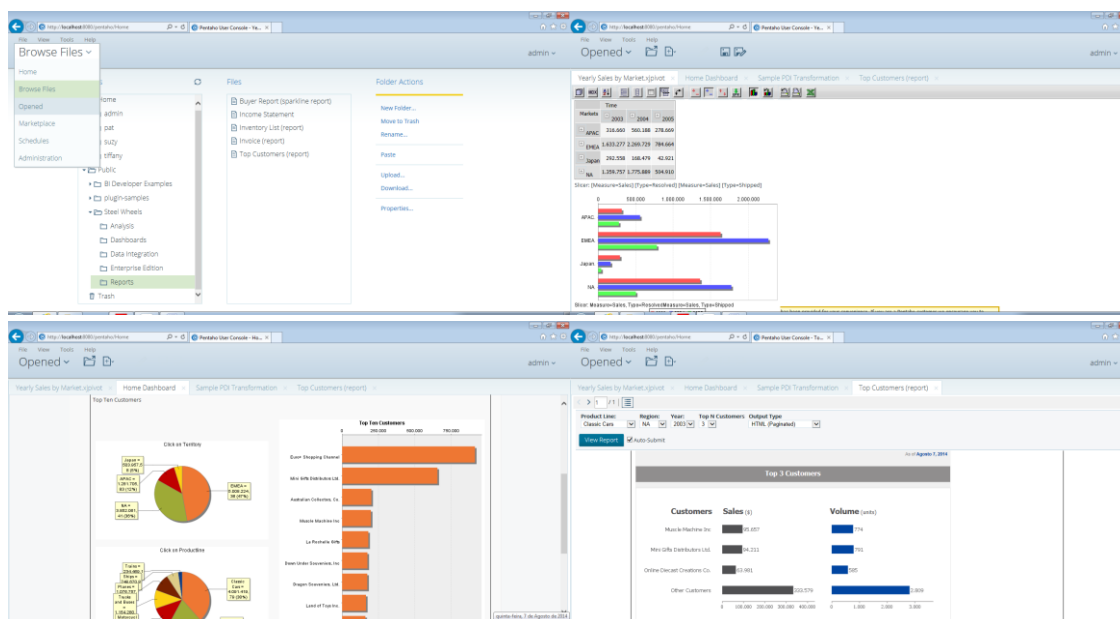


Figura 82 – PentahoBI – Navegação, OLAP, Graficos e KPI's

### 5.1.3 Análise

A instalação e configuração das duas plataformas é similar, sendo sempre necessário criar variáveis de sistema. Ambas são desenvolvidas em *Java* e usam um servidor *Web Tomcat*. O acesso e respetivo controlo é efetuado através da identificação do utilizador e palavra-chave.

No que respeita à informação disponível aos utilizadores para a operacionalização das plataformas, a SpagoBI faculta um manual mais completo que a Pentaho. Porém, esta lacuna é colmatada com a diversa literatura e vídeos existentes na Internet, gerados pela comunidade de utilizadores PentahoBI.

## 5.2 Processo de ETL

Nesta secção apresenta-se o processo de teste às ferramentas de ETL. A Pentaho fornece ferramenta própria, a KETTLE, enquanto a SpagoBI propõe o recurso a uma ferramenta externa, também *open source*, a *Talend Open Studio Data Integration (Talend OS DI)*.

O processo de teste compreendeu três fases: criação da DSA, transformação de dados e carregamento da tabela de factos e dimensões. Com o objetivo de garantir iguais circunstâncias no processo de avaliação do desempenho das ferramentas, em ambas o carregamento das dimensões temporais obedeceu a idêntico procedimento, isto é, leitura de ficheiro de Excel, transformação e carregamento na respetiva dimensão.



### 5.2.1 Talend Open Studio

Como referido, a plataforma da Spago recorre a ferramenta externa de ETL, a *Talend OS DI*. Escalabilidade e flexibilidade são duas das suas principais vantagens. Podem ser criados componentes personalizados ou efetuar, sem custos, *downloads* de componentes da comunidade *Talend Exchange*. Nesta, existem mais de 430 componentes personalizados, desenvolvidos e partilhados por utilizadores Talend. Para o carregamento da DSA foi utilizado um destes componentes, o *tTransferDatabase*, disponível no site <http://www.talendforge.org/exchange/>.

O carregamento das dimensões *DimDate* e *DimTime* é efetuado através de ficheiros Microsoft Excel. No processo afeto a cada uma das dimensões, são utilizados três componentes:

1. *tFileInputExcel* – Permite a leitura de dados de uma folha de Excel;
2. *tMap* – Mapea os dados de origem com os campos presentes no DM;
3. *tMSSqlOutput* – Carrega dados no DW/DM.

A figura 83 mostra a execução do carregamento das dimensões. Na dimensão *DimDate* é carregado um total de 44926 linhas e na *DimTime*, 960 linhas.

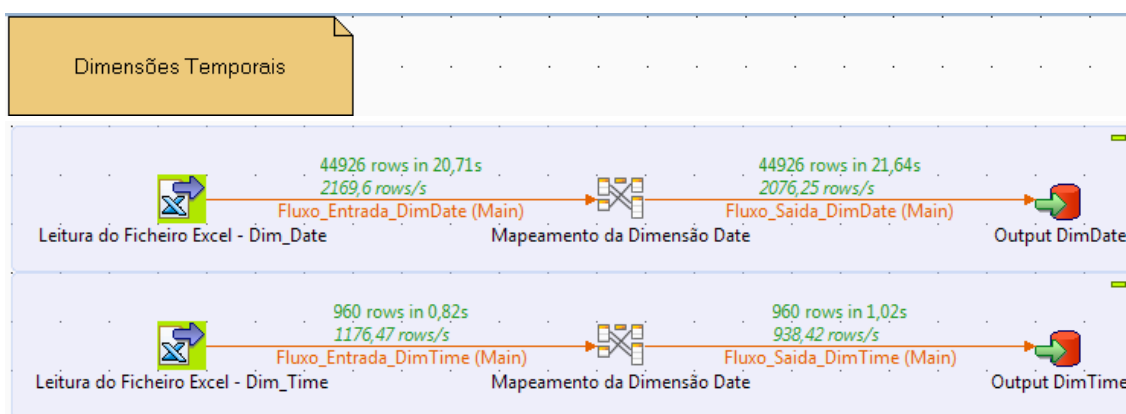


Figura 83 – Talend – Carregamento das dimensões temporais

A execução da totalidade dos componentes, presente na figura 92, decorre em, aproximadamente, 23 segundos (figura 84).

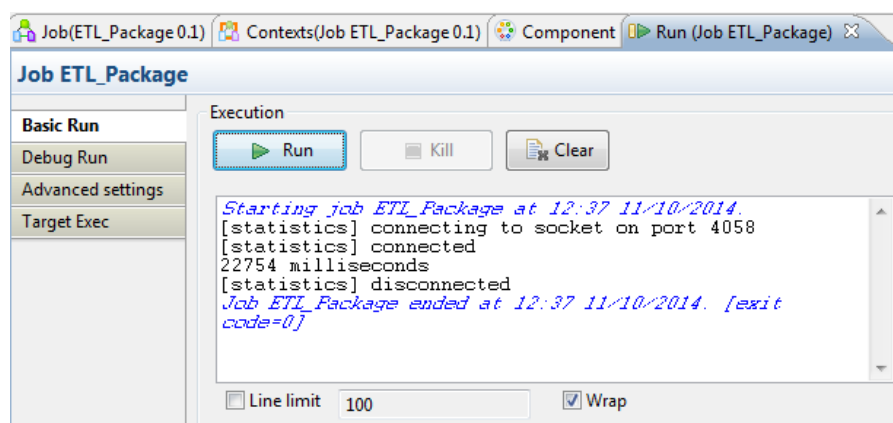


Figura 84 – Talend – Tempo de execução do carregamento das dimensões temporais

O carregamento das dimensões e da tabela de factos recorre à utilização de diversos componentes. Na figura 85 observa-se o fluxo de dados que permite o carregamento das dimensões *DimArmazéns*, *DimCentroCusto* e *DimZona*. Neste processo foram utilizados 5 componentes:

1. *tMSSqlInput* – Permite a leitura de dados provenientes do SQL Server;
2. *tUnit* – Unifica, num único fluxo, dados provenientes de diversos fluxos;
3. *tMap* – Mapea os dados de origem com os campos presentes no DM;
4. *tMSSqlOutput* – Carrega dados no DW/DM;
5. *tMSSqlSCD* – Efetua carregamentos de SCD.

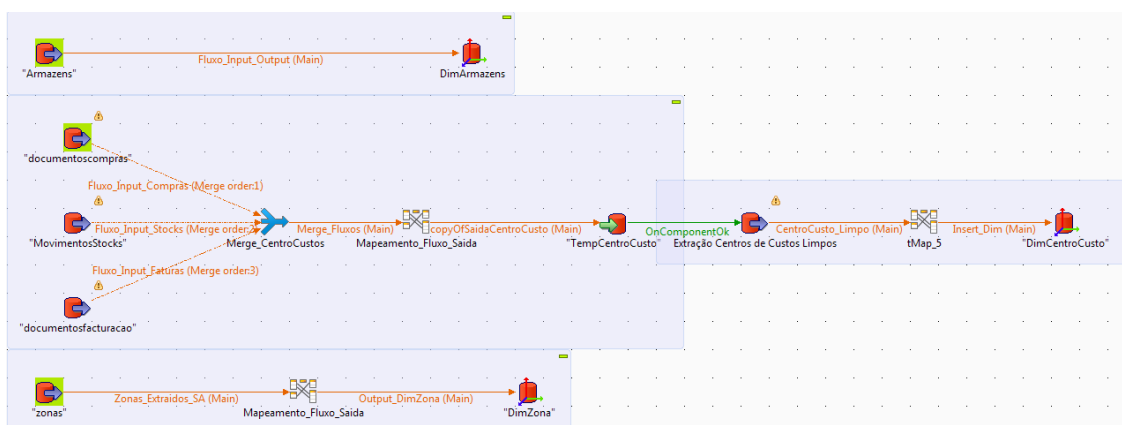


Figura 85 – Talend – Carregamento de dimensões

Um dos principais componentes presentes na ferramenta é o *tMSSqlSCD*, que permite implementar o carregamento das SCD.

### 5.2.2 Pentaho Data Integration - KETTLE

A ferramenta fornecida pela plataforma da Pentaho é a KETTLE. Tal como a *Talend Data Integration*, a KETTLE também possui um template para o carregamento das dimensões temporais. Porém, na presente análise comparativa, e como referido, o processo de carregamento é efetuado tendo por base ficheiros Microsoft Excel.

Neste processo foram utilizados quatro componentes:

1. *Execute SQL Statements* – Executa comandos escritos em SQL;
2. *Microsoft Excel Input* – Permite a leitura de dados de uma folha de Excel;
3. *Select / Rename Values* – Possibilita a seleção e o renomear de campos no fluxo de dados;
4. *Insert / Update* – Efetua carregamento e atualização dos dados nas tabelas de destino.

Na figura 95 observa-se os fluxos que permitem o carregamento das dimensões *DimDate* e *DimTime*.

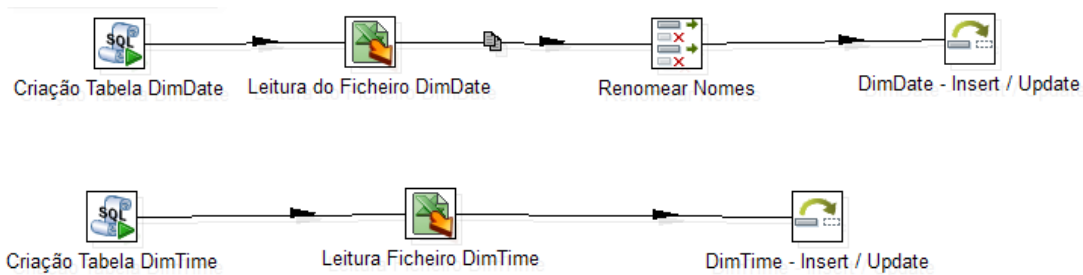


Figura 86 – KETTLE – Carregamento das dimensões temporais

A execução dos sete componentes exibidos na figura 86 tem uma duração de cerca de 1 minuto e 45 segundos, valor observável na figura 87.

#	Nome do step	Copia nr	Lidos	escritos	Entrada	Saída	Atualizados	Rejected	Erros	Ativo	Tempo	Velocidade (r/s)	Pri/ent/sai
1	Criação Tabela DimDate	0	0	1	0	0	0	0	0	Finished	0.0s	-	-
2	Leitura do Ficheiro DimDate	0	0	44926	44926	0	0	0	0	Finished	1mn 5s	683	-
3	Criação Tabela DimTime	0	0	1	0	0	0	0	0	Finished	0.0s	56	-
4	Renomear Nomes	0	44926	44926	0	0	0	0	0	Finished	1mn 26s	518	-
5	Leitura Ficheiro DimTime	0	0	960	960	0	0	0	0	Finished	2.3s	423	-
6	DimTime - Insert / Update	0	960	960	960	960	0	0	0	Finished	4.5s	215	-
7	DimDate - Insert / Update	0	44926	44926	44926	44926	0	0	0	Finished	1mn 45s	427	-

Figura 87 – Kettle – Tempo de execução do carregamento das dimensões temporais

O fluxo de ETL, apresentado na figura 85, utiliza os seguintes componentes:

1. *Table Input* – Permite a leitura de dados provenientes da BD de origem;
2. *DataBase Lookup* – Permite correspondência de dados;
3. *Filter Rows* – Filtra o fluxo de dados de forma a respeitar uma condição lógica;
4. *Add Constants* – Adiciona campos ao fluxo de dados;
5. *Microsoft Excel Output* – Permite a inserção de dados numa folha de Excel;
6. *Table Output* - Efetua carregamento dos dados nas tabelas de destino;
7. *Dimension Lookup / Update* – Permite a atualização de SCD.

Os carregamentos SCD são executados através do componente *Dimension Lookup/Update*. Um outro componente muito utilizado nestes processos é o *DataBase Lookup*, que faz a correspondência entre um valor da base de dados operacional e os dados previamente carregados na DW/DM, devolvendo a respetiva *Key*.

A Figura 88 mostra o fluxo de dados que permite o carregamento de uma dimensão.

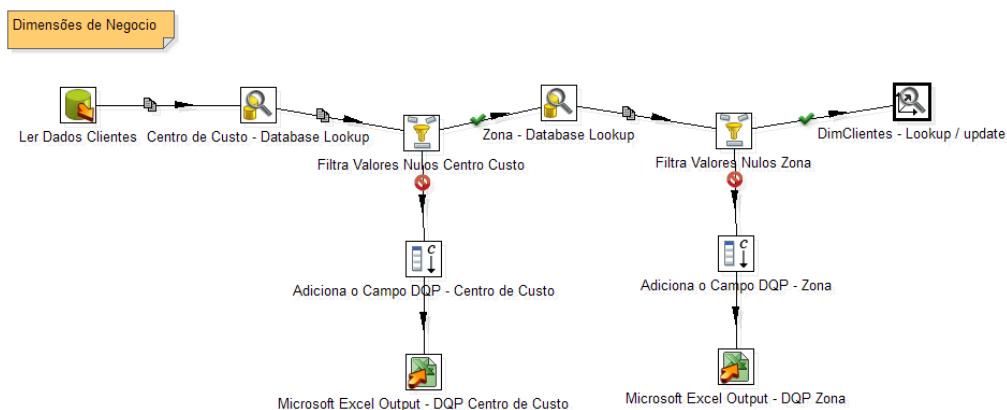


Figura 88 – Kettle – Fluxo de ETL



		DimDate								
Measures	2014									
Measure	1	2	3	4	5	6	7	8	9	
EarlierDate										
OnTimeDate										
DelayDate										

Slicer: [Name=MD] [Name=Cutting Process]

Figura 90 – Mondrian – Análise 1

A figura 91 exibe a análise presente na figura 46, da secção 4.6.2., referente à visualização dos dados relativos às fases *Request For Quotation*, *Prototype* e *Mass Production*, do processo produtivo da empresa, ao longo dos meses.

DimPhase		Measures	DimDate									
(All)	Name	Measure	-2014	2014								
				+1	+2	+3	+4	+5	+6	+7	+8	+9
-All DimPhase		EarlierDate										
		OnTimeDate										
		DelayDate										
All DimPhase	MP	EarlierDate										
		OnTimeDate										
		DelayDate										
	Prototype	EarlierDate										
		OnTimeDate										
		DelayDate										
	RFQ	EarlierDate										
		OnTimeDate										
		DelayDate										

Slicer: [Name=MD] [Name=Cutting Process]

Figura 91 – Mondrian – Análise 2

A figura 92 representa a análise patente na figura 47, da secção 4.6.2. Esta análise é relativa à evolução dos valores da métrica *OnTime*, nas fases *Request For Quotation*, *Prototype* e *Mass Production*.

**Columns** ✖

DimDate

---

**Rows**

DimPhase

Measures

---

**Filter**

DimActivity (Name=MD)

DimCarMaker

DimComplexity

DimManager

DimPerson

DimReport (Name=Cutting Process)

DimTeamLeader

DimTypeTask

OK Cancel

DimPhase		Measures	DimDate									
(All)	Name	Measure	-2014	2014								
				+1	+2	+3	+4	+5	+6	+7	+8	+9
-All DimPhase		OnTimeDate										
All DimPhase	MP	OnTimeDate										
	Prototype	OnTimeDate										
	RFQ	OnTimeDate										

Slicer: [Name=MD] [Name=Cutting Process]

Figura 92 – Mondrian – Análise 3

Uma função, de grande utilidade, é a *Drill Through*, que permite detalhar os factos numa tabela. No exemplo mostrado na figura 93, é detalhado o valor de *Delay's* na fase de produção *Request For Quotation*.

Slicer: [Name=MD] [ManagerName=] [Name=Cutting Process] [TeamLeaderName=]

DimPhase	Measures	DimDate	
		-2014	+1
RFQ	EarlierDate	↓	↓
	OnTimeDate	↓	↓
	DelayDate	↓	↓
Prototype	EarlierDate	↓	↓
	OnTimeDate	↓	↓
	DelayDate	↓	↓
HP	EarlierDate	↓	↓
	OnTimeDate	↓	↓
	DelayDate	↓	↓

Name	Name_0	Description	Year	Month	FullDate	ManagerName	CoreUserName	Name_1	Name_2	Name_3	TeamLeaderName	ManagerTeamLeaderName	DelayDate
MD		Medium High	2014	1.00	01/12/2014			RFQ	Cutting Process	PnCuttingProcess			1.00
MD		Medium High	2014	1.00	01/12/2014			RFQ	Cutting Process	PnCuttingProcess			1.00
MD		Medium High	2014	1.00	01/12/2014			RFQ	Cutting Process	PnCuttingProcess			1.00
MD		Medium High	2014	1.00	01/12/2014			RFQ	Cutting Process	PnCuttingProcess			1.00
MD		Medium High	2014	1.00	01/12/2014			RFQ	Cutting Process	PnCuttingProcess			1.00

Page 1/1 Rows/page 10

Figura 93 – Mondrian – Análise 4

### 5.3.2 PentahoBI – Saiku Analytics

Esta plataforma disponibiliza, de raiz, a possibilidade de efetuar análises através do JPivot, igualmente baseado no servidor OLAP *Mondrian*. Uma vez que, na secção anterior, este servidor já foi testado, no âmbito da plataforma *SpagoBI*, optou-se por analisar outra ferramenta presente no *MarketPlace* da Pentaho, a *Saiku Analytics*.

Na figura 94 exibe-se a análise apresentada na figura 48, da secção 4.6.2., a qual é relativa a um determinado *Manager* e *Team Leader*.

		12		
		December		
Name	Name	EarlierDate	OnTimeDate	DelayDate
Cutting Process	MP			
	Prototype			
	RFQ			

Applied filters:

- Year > Year > 2013
- ManagerName > ManagerName > [ ]
- TeamLeaderName > TeamLeaderName > [ ]
- Name > Name > MD

Figura 94 – Saiku Analytics – Análise 1

A figura 95 mostra a análise da figura 49, da secção 4.6.2., que exibe detalhes informativos referentes ao desempenho de cada colaborador, no mês de dezembro 2014.

			12			
			December			
Name	Name	CoreUserName	EarlierDate	OnTimeDate	DelayDate	
Cutting Process	MP					
	Prototype					
	RFQ					

Applied filters:

Year	»	Year	»	2013	}
ManagerName	»	ManagerName	»	<input type="text"/>	
TeamLeaderName	»	TeamLeaderName	»	<input type="text"/>	
Name	»	Name	»	MD	

Figura 95 – Saiku Analytics – Análise 2

### 5.3.3 Análise

As plataformas SpagoBI e PentahoBI recorrem ao servidor OLAP Mondrian, sendo idêntico o processo de criação e exploração do cubo de dados. Ambas permitem efetuar diversas análises sem recurso à linguagem MDX

O PentahoBI admite, ainda, a utilização de um outro servidor presente no *MarketPlace*, o *Saiku Analytics*.

## 5.4 Análise comparativa entre PentahoBI e SpagoBI

Com o propósito de compreender se as plataformas *open source* podem constituir uma alternativa fiável às plataformas comerciais, procedeu-se a testes às potencialidades das plataformas PentahoBI e SpagoBI, ambas *open source*. Os referidos testes incidiram em processos idênticos aos efetuados com recurso à plataforma BI da Microsoft, de natureza comercial, a partir de dados reais da empresa Yazaki Saltano de Ovar Produtos Eléctricos, Ltd. (capítulo 4).

Com base nos pressupostos avaliativos, enunciados no início do presente capítulo, foi elaborada a tabela 23. Os critérios qualitativos em análise compreendem: Grau de acessibilidade e usabilidade das ferramentas, Nível de desempenho da ferramenta de ETL e Utilização da ferramenta OLAP.

No que respeita ao critério *Grau de acessibilidade e usabilidade das ferramentas*, ambas as plataformas apresentam a classificação máxima. Respondem adequadamente às expectativas e compreendem todas as funcionalidades esperadas e necessárias às análises pretendidas. Cada uma assenta numa plataforma única, permite o controlo de acesso através de login com palavra-chave, possibilita a visualização de relatórios e *dashboards*, a realização de análises a cubo de dados, apresentam KPI's etc. Trata-se de plataformas de utilização intuitiva. Sublinha-se, porém, que se considera o portal da SpagoBI mais organizado e de acesso facilitado aos conteúdos. O portal da Pentaho apresenta a mais-valia de ser convertido para português europeu.

Quanto ao *Nível de desempenho da ferramenta de ETL*, considera-se que, na globalidade, a SpagoBI evidencia uma performance de maior destaque. Neste âmbito, sublinha-se a rapidez de processamento da Talend (SpagoBI), observável na duração do carregamento das dimensões temporais, que foi significamente superior à da KETTLE (PentahoBI). Esta diferença de desempenho pode encontrar justificação no facto da Talend utilizar mais recursos ao nível de *hardware*, principalmente de memória RAM. No entanto, esta potencialidade não se encontra isenta de dificuldades, uma vez que, quando da operacionalização da Talend, a máquina virtual utilizada por vezes bloqueou, não tendo sido suficiente os 4 GB de RAM de que dispunha.

A ferramenta de ETL da Pentaho, a KETTLE, ao integrar a plataforma, patenteia um ambiente de trabalho e um procedimento de configuração das ligações às BDs similares à própria plataforma, o que facilita a sua utilização. Tal não se verifica na Talend, uma vez que se trata de uma ferramenta externa à SpagoBI. Embora a KETTLE seja de uso mais facilitado e requeira menos recursos de *hardware*, não apresenta um nível de eficiência e rapidez tão elevado como a Talend, o que se entende constituir uma vantagem desta última.

Relativamente ao critério *Utilização da ferramenta OLAP*, a PentahoBI parece evidenciar um desempenho mais relevante. Para além do servidor OLAP Mondrian, esta plataforma permite a utilização de um outro servidor presente no *MarketPlace*, o *Saiku Analytics*, cuja instalação se constatou apresentar um baixo grau de dificuldade. Na ótica da utilização, esta ferramenta é do tipo *Drag-and-Drop*. Sendo *User-Friendly*, dispensa a intervenção dos profissionais das TI, o



que vai ao encontro da tendência *Self-Service BI*. A totalidade destes aspectos constitui uma mais-valia da plataforma PentahoBI.

Na SpagoBI, o servidor utilizado é o OLAP Mondrian. As análises efetuadas revelaram alguma rigidez de utilização. A ferramenta poderia ser mais *User-Friendly* e, conseqüentemente, é pouco *Self-Service*, obrigando a intervenção técnica.

Tabela 23 – Tabela comparativa PentahoBI vs SpagoBI

Critérios	PentahoBI	Pontuação	SpagoBI	Pontuação
Grau de acessibilidade e usabilidade das ferramentas	Bom	3	Bom	3
Nível de desempenho da ferramenta de ETL	Suficiente	2	Bom	3
Utilização da ferramenta OLAP	Bom	3	Suficiente	2
Classificação		8		8

Na generalidade, conclui-se por uma equivalência das plataformas. Apresentam diferenciação em dois dos três critérios considerados, contudo resultam num equilíbrio avaliativo. Face a este resultado igualitário, e tendo por base a experimentação das ferramentas, considera-se que, no presente, se se atribuir preponderância ao nível de desempenho da ferramenta de ETL, a plataforma mais indicada é a SpagoBI. Do ponto de vista do interface com o portal, a PentahoBI afigura-se mais adequada pelo ambiente gráfico apelativo e pela facilidade de utilização, de que se destaca a possibilidade de selecionar a língua portuguesa.

Na base dos testes realizados, conclui-se que as plataformas *open source* constituem uma alternativa exequível às plataformas comerciais. Constatou-se ser possível, com recurso às primeiras, replicar de forma fiável, a extração, transformação e carregamento de dados, bem como as análises efetuadas na plataforma Microsoft BI.

## 6 Conclusão e Trabalho Futuro

As soluções que as diferentes plataformas BI disponibilizam têm como objetivo central permitir às organizações ganhos de rentabilidade e uma permanente e sustentável adaptação às constantes alterações dos mercados.

O *Business Intelligence* apresenta duas potencialidades essenciais às organizações. Por um lado, permite a redução de custos e, por outro, o aumento de receitas. Isto ocorre porque as ferramentas BI podem constituir um suporte fundamental às estratégias e decisões de gestão. Disponibiliza dados cuja análise e inter-relação podem permitir criar, melhorar ou redefinir produtos, serviços e/ou processos de comercialização e negociação. Qualquer organização, comercial ou não, independentemente da sua dimensão, área de negócio, posicionamento e tipo de mercado, pode beneficiar da implementação de BI.

As tecnologias BI encontram-se num processo irreversível de inclusão nas organizações. Agregam, de forma progressiva, mais e diferentes elementos, num movimento de grande complexidade e diversidade. A rápida massificação quotidiana de algumas tecnologias, como por exemplo os dispositivos móveis, determinou que estes passassem a ser integrados pela generalidade das plataformas, respondendo, assim, a uma tendência e necessidade dos mercados.

Tradicionalmente, o conceito BI assenta em bases de dados relacionais e analíticas, mas tudo indica que as novas e sucessivas gerações BI venham a contemplar bases de dados não relacionais, cujos dados são extraídos de dispositivos eletrónicos, internet, sensores e redes sociais.

As novas tendências BI, que poderão contribuir para uma rápida disseminação do mesmo, apontam, ainda, para uma consolidação:

- Das análises em tempo real, com forte e direto impacto nos processos de tomada de decisão;

- Do *Self-Service* BI, possibilitando uma personalização de análises, acessível a diferentes utilizadores, de várias áreas e negócios;
- Das análises preditivas que, gerando cenários e padrões para o futuro, são fundamentais aos decisores;
- Das tecnologias colaborativas que, ao integrarem comportamentos, emoções, reações e atitudes manifestados por toda a rede global, geram análises de elevada qualidade informativa, fundamentais às estratégias de gestão.

Selecionar a plataforma BI mais adequada, que vá ao encontro das necessidades específicas da organização, constitui um momento crucial, de que dependerá o sucesso do processo de implementação da mesma. A utilização de ferramentas de diferentes plataformas BI pode acartar dificuldades acrescidas de configuração, utilização e partilha. Iniciou-se esta dissertação pressupondo-se como significativamente vantajoso dispor, num único ambiente gráfico, de todas as ferramentas BI, reduzindo-se, assim, eventuais custos e tempo de formação dos utilizadores. Neste âmbito, e de acordo com Wayne W. Eckerson, entre várias vantagens, o mesmo ambiente gráfico fornece um conjunto básico de funcionalidades e módulos, concebido para ir ao encontro de informações específicas e requisitos de análise de uma ampla gama de utilizadores [60]. Escolher uma única plataforma BI, com um ambiente que integre as ferramentas necessárias, favorece a sua operacionalização. Foi possível confirmar este facto quando do teste comparativo efetuado à ferramenta de ETL, entre as plataformas BI PentahoBI e SpagoBI. Na primeira, a ferramenta, a KETTLE, integra a própria plataforma, apresentando um ambiente de trabalho e um procedimento de configuração das ligações às BDs idênticos à plataforma. A segunda recorre a uma ferramenta externa, a Talend. Desta forma, constatou-se estar facilitada a utilização de diversas ferramentas num mesmo ambiente gráfico. O mesmo foi verificado no decurso da implementação de tecnologias BI na empresa Yazaki Saltano, que recorreu a ferramentas integradas na plataforma BI da Microsoft, a *stack* tecnológica já existente na organização. O processo de integração do DM, extraído do SQL Server, no PowerBI foi isento de quaisquer obstáculos. Tal reforça a importância de recorrer a um ambiente tecnológico único, sendo evidentes os ganhos de tempo e produtividade.

Quanto à implementação das ferramentas BI na empresa Yazaki Saltano, sublinha-se o facto desse processo ter possibilitado o desenvolvimento de diferentes inter-relações, análises e leituras dos dados. Para além do BI por em evidência algumas necessidades na gestão dos dados, que uma vez respondidas permitem melhorar a monitorização e controlo dos diferentes processos da empresa; possibilita, também, disponibilizar, em períodos de tempo mais reduzidos, informação relevante de suporte à decisão.

Perceber o investimento em BI como uma forte possibilidade de crescimento do negócio é fundamental, sobretudo, para as pequenas e médias empresas. Um dos objetivos desta dissertação consiste em verificar se as plataformas *open source* podem constituir uma alternativa real e concreta às plataformas comerciais. Em resultado da análise comparativa efetuada às funcionalidades das várias plataformas BI *open source* e do teste realizado às suas potencialidades, tendo como referência os resultados obtidos na implementação de BI na empresa Yazaki Saltano, conclui-se que essas plataformas podem consistir uma opção de

elevado interesse para as empresas, sobretudo em contexto de retração do investimento. Recorrendo a duas plataformas *open source*, foi possível efetuar, com elevada fiabilidade, o ETL dos dados e as análises que haviam sido realizados na plataforma Microsoft BI da Yazaki Saltano.

Uma das grandes vantagens que se aponta às plataformas *open source*, relativamente às comerciais, é a possibilidade de optar por soluções híbridas, numa combinação eclética de funcionalidades, isenta de encargos e de grande adequação às especificidades das organizações e negócios.

## 6.1 Limitações

Considera-se ter sido possível alcançar os objetivos previamente definidos no âmbito da presente dissertação. Contudo, é de realçar alguns constrangimentos identificados no decurso do trabalho desenvolvido.

Na fase inicial da dissertação, os critérios do Instituto Gartner disponíveis datavam de 2013, na base dos quais se desenvolveu a análise efetuada no capítulo 3. Os critérios divulgados em 2014 não implicaram alterações significativas à versão de 2013, mas os de 2015 reorganizam todos os critérios, tendo sido estabelecidos subcritérios. Uma vez que os critérios de 2013 estão presentes em 2015, ainda que com diferente configuração, manteve-se o proposto inicialmente.

Também a constante evolução das plataformas BI exigiu alguma dinâmica de atualização do referido capítulo, com implicações no processo da análise comparativa das funcionalidades abrangidas.

Decorrente da empresa Yazaki Saltano não autorizar, por questões de confidencialidade de dados, a colocação destes na *Cloud*, não foi possível testar a ferramenta *Predictive Forecasting*, do PowerBI, que se encontra disponível apenas em versão *on-line*.

## 6.2 Publicações

No âmbito do trabalho desenvolvido, foram publicados os seguintes artigos:

J. Lapa, J. Bernardino e A. Almeida, “A Comparative Analysis of Open Source Business Intelligence Platforms”, in International Conference on Information Systems and Design of Communication - ISDOC, 2014. Pages 86-92.

J. Lapa, J. Bernardino e A. Almeida, “Commercial Business Intelligence Suites Comparison”, in New Contributions in Information Systems and Technologies, eds: Alvaro Rocha, Ana Correia, Sandra Costanzo, Luis Paulo Reis, Advances in Intelligent Systems and Computing, 353, pp 237-246, Springer.

### 6.3 Trabalho futuro

A presente dissertação permitiu verificar que, na generalidade, as plataformas BI *open source* constituem uma alternativa credível às plataformas comerciais. Tomando este pressuposto como ponto de partida, seria de desenvolver, na íntegra, todo o processo de implementação de uma plataforma BI em contexto organizacional real, preferencialmente numa PME, avaliando a eficácia do mesmo e do seu impacto enquanto suporte às tomadas de decisão da empresa.

No que respeita ao processo de análise comparativa das funcionalidades presentes nas plataformas *open source*, seria pertinente alargar, de forma mais exaustiva, o leque das funcionalidades analisadas, enriquecendo o conhecimento sobre as mesmas e, desta forma, clarificar e facilitar o processo de seleção por parte das empresas quando da implementação de tecnologias BI.

Quanto ao projeto implementado na empresa Yazaki Saltano, a sua continuidade poderia permitir o acréscimo de novos campos na tabela de factos, redesenhando o processo de ETL e desenvolver a DW, de modo a abranger outros KPI's não contemplados neste desenvolvimento.

## Referências

- [1] S. Negash e P. Gray, “Business Intelligence,” em *Ninth Americas Conference on Information Systems (AMCIS)*, 2003.
- [2] D. Parrek, *Business Intelligence for Telecommunications*, Auerbach Publications, 2007.
- [3] “Q&A: BI Visionary Howard Dresner,” Information Week – Connecting the Business Technology Community, [Online]. Available: <http://www.informationweek.com/software/information-management/qanda-bi-visionary-howard-dresner>. [Acedido em Maio 2015].
- [4] E.-P. Lim, H. Chen e G. Chen, “Business intelligence and analytics: Research directions,” em *ACM Trans. Manage. Inf. Syst.* 3, 4, Article 17, 10 pages., January 2013.
- [5] W. R. King, “Knowledge Management and Organizational Learning,” em *Annals of Information Systems* 4, DOI 10.1007/978-1-4419-0011-1\_1, © Springer Science+Business Media, 2009.
- [6] F. P. S. M. N. I. J. N. L. M. S. Sulaiman Ainin, “Factors influencing the use of social media by SMEs and its performance outcomes,” em *Industrial Management and Data Systems*, 2015.
- [7] F. W. R. M. Yafeng Lu, “Business Intelligence from Social Media: A Study from the VAST Box Office Challenge,” em *IEEE Computer Graphics and Applications*, 2014.
- [8] P. A. O. Rob Meredith, “A Framework for Understanding the Role of Social Media in Business Intelligence Systems. Journal of Decision Systems,” vol. 20(3), pp. 263-282, 2011.
- [9] P. Meehan, Interviewee, *Gartner Business Intelligence Summit*. [Entrevista]. 31 January 2011.
- [10] W. Yeoh e A. Koronios, “Critical Success Factors for Business Intelligence Systems,” em *Journal of Computer Information Systems*, 50 (3), 23-32., 2010.
- [11] S. Adamala e L. Cidrin, “Key Success Factors in Business Intelligence,” em *Journal of Intelligence Studies in Business* 1, 2011.
- [12] P. A. C. P. V. S. Saagari, “Data Warehousing, Data Mining, OLAP and OLTP Technologies Are Essential Elements to Support Decision-Making Process in Industries,” *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, pp. 88-93, 2013.
- [13] “Business Intelligence,” Intel - Diálogo TI - Next Generation Center.
- [14] W. Inmon, *Building the Data Warehouse*, Wiley Publishing, Inc, 2005.
- [15] J. Serra, “Data Warehouse Architecture – Kimball and Inmon methodologies,” [Online]. Available: <http://www.jamesserra.com/archive/2012/03/data-warehouse-architecture-kimball-and-inmon-methodologies/>. [Acedido em Outubro 2015].
- [16] R. Kimball e M. Ross, *The Data Warehouse Toolkit, The Definitive Guide to Dimensional Modeling, Third Edition*, John Wiley & Sons, Inc., 2013.

- [17] M. Y. Santos e I. Ramos, *Business Intelligence*, FCA - Editora de Informática, 2009.
- [18] D. Cardon, “Database vs Data Warehouse: A Comparative Review,” *Health Catalyst*, 2014.
- [19] A. Parekh, “Introduction on Data Warehouse with OLTP and OLAP,” *International Journal Of Engineering And Computer Science*, vol. 2, pp. 2569-2573, 2013.
- [20] “Visão geral do OLAP (processamento analítico online),” [Online]. Available: <https://support.office.com/pt-br/article/Vis%C3%A3o-geral-do-OLAP-processamento-anal%C3%ADtico-online-15d2cdde-f70b-4277-b009-ed732b75fdd6>. [Acedido em Agosto 2015].
- [21] “MOLAP, ROLAP, And HOLAP,” [Online]. Available: <http://www.1keydata.com/datawarehousing/molap-rolap.html>. [Acedido em Agosto 2015].
- [22] A. Rakesh e S. Ramakrishnan, “Fast Algorithms for Mining Association Rules,” em *Proceedings of the 20th International Conference on Very Large Data Bases*, 1994.
- [23] “Self-Service Business Intelligence,” Gartner, Inc., [Online]. Available: <http://www.gartner.com/it-glossary/self-service-business-intelligence>. [Acedido em 2015].
- [24] M. Rouse, “What is Self-Service Business Intelligence?,” TechTarget, [Online]. Available: <http://searchbusinessanalytics.techtarget.com/definition/self-service-business-intelligence-BI>. [Acedido em Outubro 2015].
- [25] C. Sezões, J. Oliveira e M. Baptista, *Business intelligence, SPI – Sociedade Portuguesa de Inovação*, 2006.
- [26] A. H. Ghapanchi e A. Aurum, “The impact of project licence and operating system on the effectiveness of the defect-fixing process in open source software projects,” *International Journal of Business Information Systems - IJBIS*, vol. 8, nº 4, pp. 413-424, 2011.
- [27] K. Wurst, L. Postner e S. Jackson, “Teaching open source (software),” em *In Proceedings of the 45th ACM technical symposium on Computer science education (SIGCSE '14)*, New York, NY, USA, 2014.
- [28] R. Pollock, “Innovation, Imitation and Open Source,” *International Journal of Open Source Software & Processes*, vol. 1, nº 2, pp. 114-127, 2009.
- [29] J. Bessen, “Open Source Software: Free Provision of Complex Public Goods,” In *Jürgen Bitzer & Philipp J. H. Schröder (Eds.), The Economics of Open Source Software Development*, Elsevier B.V., 2006.
- [30] R. M. Sauer, “Why Develop Open-Source Software? The Role of Non-Pecuniary Benefits, Monetary Rewards, and Open-Source Licence Type,” *Oxford Review of Economic Policy*, vol. 23, nº 4, pp. 605-619, 2007.
- [31] R. Stallman, “What is free software: The free software definition,” Official website of Free Software Foundation (FSF), 2013. [Online]. Available: <http://www.gnu.org/philosophy/free-sw.en.html>. [Acedido em 2015].
- [32] Actuate. [Online]. Available: [www.actuate.com](http://www.actuate.com).

- [33] JasperSoft. [Online]. Available: [www.jaspersoft.com](http://www.jaspersoft.com).
- [34] OpenI. [Online]. Available: <http://openi.sourceforge.net/index.html>.
- [35] PALO. [Online]. Available: [www.palo.net](http://www.palo.net).
- [36] Pentaho. [Online]. Available: [www.pentaho.com](http://www.pentaho.com).
- [37] SpagoBI. [Online]. Available: [www.spagoworld.org](http://www.spagoworld.org).
- [38] Vanilla. [Online]. Available: [www.bpm-conseil.com](http://www.bpm-conseil.com).
- [39] beMemo. [Online]. Available: [www.bememo.com](http://www.bememo.com).
- [40] IBM Cognos, [Online]. Available: [www-01.ibm.com/software/analytics/cognos/](http://www-01.ibm.com/software/analytics/cognos/).
- [41] Microsoft. [Online]. Available: [www.microsoft.com/bi](http://www.microsoft.com/bi).
- [42] MicroStrategy. [Online]. Available: [www.microstrategy.com](http://www.microstrategy.com).
- [43] Oracle. [Online]. Available: [www.oracle.com/us/solutions/business-analytics/business-intelligence/overview/index.html](http://www.oracle.com/us/solutions/business-analytics/business-intelligence/overview/index.html).
- [44] SAP. [Online]. Available: <http://www.sap.com>.
- [45] SAS Business Intelligence & Analytics, [Online]. Available: [www.sas.com](http://www.sas.com).
- [46] SOL7-BI. [Online]. Available: [www.sol7.com.br](http://www.sol7.com.br).
- [47] K. Schlegel, R. Sallam, D. Yuen e J. Tapadinhas, “Magic Quadrant for Business Intelligence and Analytics Platforms,” Gartner, 2013.
- [48] W. E2, “Building on Early Successes in UBM’s Social Business Strategy,” 2013. [Online].
- [49] R. Sallam, B. Hostmann, K. Schlegel, J. Tapadinhas, J. Parenteau e T. Oestreich, “Magic Quadrant for Business Intelligence and Analytics Platforms,” Gartner, 2015.
- [50] J. Lapa, J. Bernardino e A. Figueiredo, “A Comparative Analysis of Open Source Business Intelligence Platforms,” em *International Conference on Information Systems and Design of Communication - ISDOC*, 2014.
- [51] P. Mell e T. Grance, “The NIST Definition of Cloud Computing,” National Institute of Standards and Technology Special Publication, 2011.
- [52] [Online]. Available: [www.yazaki-europe.com](http://www.yazaki-europe.com). [Acedido em julho 2015].
- [53] “Integration Services - SSIS,” Microsoft, Março 2015. [Online]. Available: <https://msdn.microsoft.com/en-us/library/ms141026.aspx>.
- [54] “Analysis Services - SSAS,” Microsoft, Março 2015. [Online]. Available: <https://msdn.microsoft.com/en-us/library/bb522607.aspx>.
- [55] “Reporting Services - SSRS,” Microsoft, Março 2015. [Online]. Available: <https://msdn.microsoft.com/en-us/library/ms159106.aspx>.
- [56] D. Clark, *Beginning Power BI with Excel 2013: Self-Service Business Intelligence Using Power Pivot, Power View, Power Query, and Power Map*, Apress Media, 2014.
- [57] S. Sousa e M. J. Sousa, *Microsoft Office 97 Avançado*, FCA - Editora de Informática, 1998.
- [58] “ISO/IEC 14598 - Software Product Evaluation - Part 4: Process for Acquirers,” International Organization for Standardization.



- [59] “ISO/IEC 9126 - Software Product Quality - Part 1: Quality Model,” International Organization for Standardization.
- [60] W. W. Eckerson, “The Keys to Enterprise Business Intelligence: Critical Success Factors,” The Data Warehousing Institute (TDWI), 2005.
- [61] R. Kimball e M. Ross, The Data Warehouse Toolkit, The Complete Guide to Dimensional Modeling, Second Edition, John Wiley and Sons, Inc., 2002.
- [62] H. Bedoya, F. Cruz, D. Lema e S. Singkorapoom, Stored Procedures, Triggers, and User-Defined Functions on DB2 Universal Database for iSeries, IBM Redbooks, 2006.
- [63] R. Sallam, J. Tapadinhas, J. Parenteau, D. Yuen e B. Hostmann, “Magic Quadrant for Business Intelligence and Analytics Platforms,” Gartner, 2014.

# Anexos

## Anexo I – Store Procedure

Report\_MD\_DeliveryPerformance\_PnIssue

```
USE [WheTool-Full]
GO
/***** Object: StoredProcedure
[dbo].[usp_PlanningReport_MD_DeliveryPerformance_PnIssue]    Script Date:
12/05/2014 15:50:25 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
-- =====
-- Author:          Yazaki
-- Create date:    2013-02-28
-- Description:
--CASE @reportTypeID
--     WHEN 1 THEN AgreeDate
--     WHEN 2 THEN RequestDate
--     WHEN 3 THEN CuttingProcess
--     WHEN 4 THEN AssemblyProcess
-- =====
ALTER PROCEDURE [dbo].[usp_PlanningReport_MD_DeliveryPerformance_PnIssue]
-- standards filters that should appear in all reports
@startDate datetime,
@endDate datetime,
@carMakerId uniqueidentifier,
@coreUserId_TeamLeader xml,
@phaseId uniqueidentifier,
@complexityRange nvarchar(50),

@ytdTarget float,
@reportTypeID int
AS
BEGIN
-- SET NOCOUNT ON added to prevent extra result sets from
-- interfering with SELECT statements.
SET NOCOUNT ON;

SET TRANSACTION ISOLATION LEVEL READ UNCOMMITTED

DECLARE @activityId uniqueidentifier
SET @activityId = 'D5045C21-826C-4A30-AC8E-393024F1CDDDB'

SELECT
    Year,
    Month,
    SUM(Total) AS 'Total',
    SUM(Delay) AS 'Delay',
    SUM(Earlier) AS 'Earlier'
FROM
(
    SELECT
        vw.PnBaseSwoId,
        CASE @reportTypeID
            WHEN 1 THEN YEAR(vw.PnFlag1PlanEndDate)
            WHEN 2 THEN YEAR(vw.PnRequestDate)
```

```

        WHEN 3 THEN YEAR(vw.PnCuttingProcessPlanEndDate)
        WHEN 4 THEN YEAR(vw.PnAssemblyProcessPlanEndDate)
    END AS Year,
    CASE @reportTypeID
        WHEN 1 THEN MONTH(vw.PnFlag1PlanEndDate)
        WHEN 2 THEN MONTH(vw.PnRequestDate)
        WHEN 3 THEN MONTH(vw.PnCuttingProcessPlanEndDate)
        WHEN 4 THEN MONTH(vw.PnAssemblyProcessPlanEndDate)
    END AS Month,
    COUNT(*) AS 'Total',
    0 AS 'Delay',
    0 AS 'Earlier'
FROM
    vwBaseAndPns AS vw
WHERE
    @activityId = vw.ActivityId AND
    ( @carMakerId IS NULL OR vw.CarMakerId = @carMakerId ) AND
    ( @phaseId IS NULL OR vw.PhaseId = @phaseId ) AND
    ( @coreUserId_TeamLeader IS NULL OR vw.PnBaseCreator_UserId IN
(
    SELECT
    ParamValues.ID.value('.', 'uniqueidentifier')
    FROM
    @coreUserId_TeamLeader.nodes('/ROOT/ID') AS ParamValues(ID) ) ) AND
    (
        ( @reportTypeID = 1 AND vw.PnFlag1PlanEndDate BETWEEN
        @startDate AND @endDate ) OR
        ( @reportTypeID = 2 AND vw.PnRequestDate BETWEEN
        @startDate AND @endDate ) OR
        ( @reportTypeID = 3 AND vw.PnCuttingProcessPlanEndDate
        BETWEEN @startDate AND @endDate ) OR
        ( @reportTypeID = 4 AND vw.PnAssemblyProcessPlanEndDate
        BETWEEN @startDate AND @endDate )
    )
GROUP BY
    vw.PnBaseSwoId,
    CASE @reportTypeID
        WHEN 1 THEN YEAR(vw.PnFlag1PlanEndDate)
        WHEN 2 THEN YEAR(vw.PnRequestDate)
        WHEN 3 THEN YEAR(vw.PnCuttingProcessPlanEndDate)
        WHEN 4 THEN YEAR(vw.PnAssemblyProcessPlanEndDate)
    END,
    CASE @reportTypeID
        WHEN 1 THEN MONTH(vw.PnFlag1PlanEndDate)
        WHEN 2 THEN MONTH(vw.PnRequestDate)
        WHEN 3 THEN MONTH(vw.PnCuttingProcessPlanEndDate)
        WHEN 4 THEN MONTH(vw.PnAssemblyProcessPlanEndDate)
    END
UNION ALL
SELECT
    vw.PnBaseSwoId,
    CASE @reportTypeID
        WHEN 1 THEN YEAR(vw.PnFlag1PlanEndDate)
        WHEN 2 THEN YEAR(vw.PnRequestDate)
        WHEN 3 THEN YEAR(vw.PnCuttingProcessPlanEndDate)
        WHEN 4 THEN YEAR(vw.PnAssemblyProcessPlanEndDate)
    END AS Year,
    CASE @reportTypeID
        WHEN 1 THEN MONTH(vw.PnFlag1PlanEndDate)
        WHEN 2 THEN MONTH(vw.PnRequestDate)
        WHEN 3 THEN MONTH(vw.PnCuttingProcessPlanEndDate)
        WHEN 4 THEN MONTH(vw.PnAssemblyProcessPlanEndDate)

```

```

        END AS Month,
        0 AS 'Total',
        COUNT(*) AS 'Delay',
        0 AS 'Earlier'
FROM
    vwBaseAndPns AS vw
WHERE
    @activityId = vw.ActivityId AND
    ( @carMakerId IS NULL OR vw.CarMakerId = @carMakerId ) AND
    ( @phaseId IS NULL OR vw.PhaseId = @phaseId ) AND
    ( @coreUserId_TeamLeader IS NULL OR vw.PnBaseCreator_UserId IN
(
    SELECT
ParamValues.ID.value('.', 'uniqueidentifier')
FROM
@coreUserId_TeamLeader.nodes('/ROOT/ID') AS ParamValues(ID) ) ) AND
(
    ( @reportTypeID = 1 AND vw.PnFlag1PlanEndDate BETWEEN
@startDate AND @endDate AND DATEDIFF(d, vw.PnFlag1PlanEndDate,
vw.PnFlag1RealEndDate ) > 0 ) OR
    ( @reportTypeID = 2 AND vw.PnRequestDate BETWEEN
@startDate AND @endDate AND DATEDIFF(d, vw.PnRequestDate,
vw.PnFlag1RealEndDate ) > 0 ) OR
    ( @reportTypeID = 3 AND vw.PnCuttingProcessPlanEndDate
BETWEEN @startDate AND @endDate AND DATEDIFF(d, vw.PnCuttingProcessPlanEndDate,
vw.PnCuttingProcessRealEndDate ) > 0 ) OR
    ( @reportTypeID = 4 AND vw.PnAssemblyProcessPlanEndDate
BETWEEN @startDate AND @endDate AND DATEDIFF(d,
vw.PnAssemblyProcessPlanEndDate, vw.PnAssemblyProcessRealEndDate ) > 0 )
)
GROUP BY
    vw.PnBaseSwoId,
    CASE @reportTypeID
        WHEN 1 THEN YEAR(vw.PnFlag1PlanEndDate)
        WHEN 2 THEN YEAR(vw.PnRequestDate)
        WHEN 3 THEN YEAR(vw.PnCuttingProcessPlanEndDate)
        WHEN 4 THEN YEAR(vw.PnAssemblyProcessPlanEndDate)
    END,
    CASE @reportTypeID
        WHEN 1 THEN MONTH(vw.PnFlag1PlanEndDate)
        WHEN 2 THEN MONTH(vw.PnRequestDate)
        WHEN 3 THEN MONTH(vw.PnCuttingProcessPlanEndDate)
        WHEN 4 THEN MONTH(vw.PnAssemblyProcessPlanEndDate)
    END
END
UNION ALL
SELECT
    vw.PnBaseSwoId,
    CASE @reportTypeID
        WHEN 1 THEN YEAR(vw.PnFlag1PlanEndDate)
        WHEN 2 THEN YEAR(vw.PnRequestDate)
        WHEN 3 THEN YEAR(vw.PnCuttingProcessPlanEndDate)
        WHEN 4 THEN YEAR(vw.PnAssemblyProcessPlanEndDate)
    END AS Year,
    CASE @reportTypeID
        WHEN 1 THEN MONTH(vw.PnFlag1PlanEndDate)
        WHEN 2 THEN MONTH(vw.PnRequestDate)
        WHEN 3 THEN MONTH(vw.PnCuttingProcessPlanEndDate)
        WHEN 4 THEN MONTH(vw.PnAssemblyProcessPlanEndDate)
    END AS Month,
    0 AS 'Total',
    0 AS 'Delay',
    COUNT(*) AS 'Earlier'

```

```

FROM
vwBaseAndPns AS vw
WHERE
@activityId = vw.ActivityId AND
( @carMakerId IS NULL OR vw.CarMakerId = @carMakerId ) AND
( @phaseId IS NULL OR vw.PhaseId = @phaseId ) AND
( @coreUserId_TeamLeader IS NULL OR vw.PnBaseCreator_UserId IN
(
SELECT
ParamValues.ID.value('.', 'uniqueidentifier')
FROM
@coreUserId_TeamLeader.nodes('/ROOT/ID') AS ParamValues(ID) ) ) AND
(
( @reportTypeID = 1 AND vw.PnFlag1PlanEndDate BETWEEN
@startDate AND @endDate AND DATEDIFF(d, vw.PnFlag1PlanEndDate,
vw.PnFlag1RealEndDate ) < 0 ) OR
( @reportTypeID = 2 AND vw.PnRequestDate BETWEEN
@startDate AND @endDate AND DATEDIFF(d, vw.PnRequestDate,
vw.PnFlag1RealEndDate ) < 0 ) OR
( @reportTypeID = 3 AND vw.PnCuttingProcessPlanEndDate
BETWEEN @startDate AND @endDate AND DATEDIFF(d, vw.PnCuttingProcessPlanEndDate,
vw.PnCuttingProcessRealEndDate ) < 0 ) OR
( @reportTypeID = 4 AND vw.PnAssemblyProcessPlanEndDate
BETWEEN @startDate AND @endDate AND DATEDIFF(d,
vw.PnAssemblyProcessPlanEndDate, vw.PnAssemblyProcessRealEndDate ) < 0 )
)
GROUP BY
vw.PnBaseSwoId,
CASE @reportTypeID
WHEN 1 THEN YEAR(vw.PnFlag1PlanEndDate)
WHEN 2 THEN YEAR(vw.PnRequestDate)
WHEN 3 THEN YEAR(vw.PnCuttingProcessPlanEndDate)
WHEN 4 THEN YEAR(vw.PnAssemblyProcessPlanEndDate)
END,
CASE @reportTypeID
WHEN 1 THEN MONTH(vw.PnFlag1PlanEndDate)
WHEN 2 THEN MONTH(vw.PnRequestDate)
WHEN 3 THEN MONTH(vw.PnCuttingProcessPlanEndDate)
WHEN 4 THEN MONTH(vw.PnAssemblyProcessPlanEndDate)
END
) AS TT
GROUP BY
Month,
Year
ORDER BY
1,2

```

END

## Anexo II – Script SQL

SELECT

```
Task.Id AS 'TaskId',
Task.PnBaseId AS 'TaskPnBaseId',
Task.PnId AS 'TaskPnId',
Task.Name AS 'TaskName',
Task.FriendlyName AS 'TaskFriendlyName',
Task.SortKey AS 'TaskSortKey',
Task.PlanStartDate AS 'TaskPlanStartDate',
Task.PlanEndDate AS 'TaskPlanEndDate',
Task.RealStartDate AS 'TaskRealStartDate',
Task.RealEndDate AS 'TaskRealEndDate',
Task.CoreUserId AS 'TaskCoreUserId',
Task.OverdueReason AS 'TaskOverdueReason',
Task.Notes AS 'TaskNotes',
Task.Qty AS 'TaskQty',
PlanTaskStatus.nameSimple AS 'TaskStatusName',
PlanTaskStatus.name AS 'TaskStatusNameDetail',
PlanTaskStatus.Number AS 'TaskStatusNumber',
PlanTaskStatus.NumberSimple AS 'TaskStatusNumberSimple',
PlanTaskStatus.color AS 'TaskStatusColor',
CoreUser.FirstName + ' ' + CoreUser.LastName AS 'TaskCoreUserFullName',
Activity.SearchByClient AS 'ActivitySearchByClient',
vwBaseAndPns.*
```

FROM

```
dbo.Task INNER JOIN
(
```

SELECT

```
vwBase.*,
Pn.Id AS 'PnId',
Pn.YazakiNumber AS 'PnYazakiNumber',
Pn.YazakiLevel AS 'PnYazakiLevel',
Pn.ClientNumber AS 'PnClientNumber',
Pn.ClientLevel AS 'PnClientLevel',
Pn.PnBaseId AS 'PnPnBaseId',
Pn.QtyCircuits AS 'PnQtyCircuits',
Pn.SocleOrModules AS 'PnSocleOrModules',
Pn.Notes AS 'PnNotes',
Pn.ApCrossCheckPlanStartDate AS 'PnApCrossCheckPlanStartDate',
Pn.ApCrossCheckPlanEndDate AS 'PnApCrossCheckPlanEndDate',
Pn.ApCrossCheckRealStartDate AS 'PnApCrossCheckRealStartDate',
Pn.ApCrossCheckRealEndDate AS 'PnApCrossCheckRealEndDate',
Pn.Flag1PlanStartDate AS 'PnFlag1PlanStartDate',
Pn.Flag1PlanEndDate AS 'PnFlag1PlanEndDate',
Pn.Flag1RealStartDate AS 'PnFlag1RealStartDate',
Pn.Flag1RealEndDate AS 'PnFlag1RealEndDate',
Pn.CuttingProcessPlanStartDate AS 'PnCuttingProcessPlanStartDate',
Pn.CuttingProcessPlanEndDate AS 'PnCuttingProcessPlanEndDate',
Pn.CuttingProcessRealStartDate AS 'PnCuttingProcessRealStartDate',
Pn.CuttingProcessRealEndDate AS 'PnCuttingProcessRealEndDate',
Pn.AssemblyProcessPlanStartDate AS 'PnAssemblyProcessPlanStartDate',
Pn.AssemblyProcessPlanEndDate AS 'PnAssemblyProcessPlanEndDate',
Pn.AssemblyProcessRealStartDate AS 'PnAssemblyProcessRealStartDate',
Pn.AssemblyProcessRealEndDate AS 'PnAssemblyProcessRealEndDate',
Pn.De2dBreakDownPlanStartDate AS 'PnDe2dBreakDownPlanStartDate',
Pn.De2dBreakDownPlanEndDate AS 'PnDe2dBreakDownPlanEndDate',
Pn.De2dBreakDownRealStartDate AS 'PnDe2dBreakDownRealStartDate',
Pn.De2dBreakDownRealEndDate AS 'PnDe2dBreakDownRealEndDate',
Pn.RequestDate AS 'PnRequestDate',
Pn.TechnicalChangeNumber AS 'PnTechnicalChangeNumber'
```

FROM

```

SELECT (
  CarMaker.Id AS 'CarMakerId',
  CarMaker.Name AS 'CarMakerName',

  CarModel.Id AS 'CarModelId',
  CarModel.Name AS 'CarModelName',

  CarModelYear.Id AS 'CarModelYearId',
  CarModelYear.Name AS 'CarModelYearName',

  Phase.Id AS 'PhaseId',
  Phase.Name AS 'PhaseName',

  CarModelYearPhase.Id AS 'CarModelYearPhaseId',

  Activity.Id AS 'ActivityId',
  Activity.Name AS 'ActivityName',
  Activity.KeyName AS 'ActivityKeyName',

  -- required to include in our Core
  PnBase.Id,
  PnBase.CreatedBy,
  PnBase.CreatedOn,
  PnBase.ModifiedBy,
  PnBase.ModifiedOn,
  PnBase.RowNotes,
  PnBase.RowVersion,

  PnBase.Name AS 'PnBaseName',
  PnBase.YazakiNumber AS 'PnBaseYazakiNumber',
  PnBase.YazakiLevel AS 'PnBaseYazakiLevel',
  PnBase.ClientNumber AS 'PnBaseClientNumber',
  PnBase.ClientLevel AS 'PnBaseClientLevel',
  PnBase.HoursPlanned AS 'PnBaseHoursPlanned',
  PnBase.RequestNumber AS 'PnBaseRequestNumber',
  PnBase.TechnicalChangeNumber AS 'PnBaseTechnicalChangeNumber',
  PnBase.PhaseInternalName AS 'PnBasePhaseInternalName',

  PnBase.CoreUserIdCreator AS 'PnBaseCreator_UserId',
  (select FirstName + ' ' + LastName from CoreUser Where CoreUser.Id =
PnBase.CoreUserIdCreator) AS 'PnBaseCreator_FullName',

  PnBase.CoreUserIdResponsible AS 'PnBaseResponsible_UserId',
  (select FirstName + ' ' + LastName from CoreUser Where CoreUser.Id =
PnBase.CoreUserIdResponsible) AS 'PnBaseResponsible_FullName',

  PnBase.CoreUserIdChecker AS 'PnBaseChecker_UserId',
  (select FirstName + ' ' + LastName from CoreUser Where CoreUser.Id =
PnBase.CoreUserIdChecker) AS 'PnBaseChecker_FullName',

  PnBase.SwoId AS 'PnBaseSwoId',
  PnBase.SwoInternalNumber AS 'PnBaseSwoInternalNumber',
  PnBase.QtyCircuits AS 'PnBaseQtyCircuits',
  PnBase.ComplexityRange AS 'PnBaseComplexityRange',
  PnBase.DcDsQty AS 'PnBaseQtyDcDs',
  PnBase.DcDsQtyPnsAffected AS 'PnBaseDcDsQtyPnsAffected',
  PnBase.DcDsPnsAffected AS 'PnBaseDcDsPnsAffected',
  PnBase.MdFilename AS 'PnBaseMdFilename',
  PnBase.DrawingReleaseDate AS 'PnBaseDrawingReleaseDate',
  PnBase.SocleOrModules AS 'PnBaseSocleOrModules',
  PnBase.Notes AS 'PnBaseNotes',
  PnBase.LastECRProcessed AS 'PnBaseLastECRProcessed',

```

PnBase.AffectedPns AS 'PnBaseAffectedPns',  
PnBase.LevelForTp AS 'PnBaseLevelForTp',  
PnBase.Obsolete AS 'PnBaseObsolete',  
  
PnBase.DrawingAnalysisPlanStartDate AS 'PnBaseDrawingAnalysisPlanStartDate',  
PnBase.DrawingAnalysisPlanEndDate AS 'PnBaseDrawingAnalysisPlanEndDate',  
PnBase.DrawingAnalysisRealStartDate AS 'PnBaseDrawingAnalysisRealStartDate',  
PnBase.DrawingAnalysisRealEndDate AS 'PnBaseDrawingAnalysisRealEndDate',  
PnBase.PdInputPlanStartDate AS 'PnBasePdInputPlanStartDate',  
PnBase.PdInputPlanEndDate AS 'PnBasePdInputPlanEndDate',  
PnBase.PdInputRealStartDate AS 'PnBasePdInputRealStartDate',  
PnBase.PdInputRealEndDate AS 'PnBasePdInputRealEndDate',  
PnBase.PdCheckerPlanStartDate AS 'PnBasePdCheckerPlanStartDate',  
PnBase.PdCheckerPlanEndDate AS 'PnBasePdCheckerPlanEndDate',  
PnBase.PdCheckerRealStartDate AS 'PnBasePdCheckerRealStartDate',  
PnBase.PdCheckerRealEndDate AS 'PnBasePdCheckerRealEndDate',  
PnBase.DpPlanStartDate AS 'PnBaseDpPlanStartDate',  
PnBase.DpPlanEndDate AS 'PnBaseDpPlanEndDate',  
PnBase.DpRealStartDate AS 'PnBaseDpRealStartDate',  
PnBase.DpRealEndDate AS 'PnBaseDpRealEndDate',  
PnBase.MdPlanStartDate AS 'PnBaseMdPlanStartDate',  
PnBase.MdPlanEndDate AS 'PnBaseMdPlanEndDate',  
PnBase.MdRealStartDate AS 'PnBaseMdRealStartDate',  
PnBase.MdRealEndDate AS 'PnBaseMdRealEndDate',  
  
PnBase.AddDelSheetReleasePlanStartDate AS  
'PnBaseAddDelSheetReleasePlanStartDate',  
PnBase.AddDelSheetReleasePlanEndDate AS 'PnBaseAddDelSheetReleasePlanEndDate',  
PnBase.AddDelSheetReleaseRealStartDate AS  
'PnBaseAddDelSheetReleaseRealStartDate',  
PnBase.AddDelSheetReleaseRealEndDate AS 'PnBaseAddDelSheetReleaseRealEndDate',  
  
PnBase.De2dDrawAnalysisPlanStartDate AS 'PnBaseDe2dDrawAnalysisPlanStartDate',  
PnBase.De2dDrawAnalysisPlanEndDate AS 'PnBaseDe2dDrawAnalysisPlanEndDate',  
PnBase.De2dDrawAnalysisRealStartDate AS 'PnBaseDe2dDrawAnalysisRealStartDate',  
PnBase.De2dDrawAnalysisRealEndDate AS 'PnBaseDe2dDrawAnalysisRealEndDate',  
PnBase.De2dDrawInputDsiPlanStartDate AS 'PnBaseDe2dDrawInputDsiPlanStartDate',  
PnBase.De2dDrawInputDsiPlanEndDate AS 'PnBaseDe2dDrawInputDsiPlanEndDate',  
PnBase.De2dDrawInputDsiRealStartDate AS 'PnBaseDe2dDrawInputDsiRealStartDate',  
PnBase.De2dDrawInputDsiRealEndDate AS 'PnBaseDe2dDrawInputDsiRealEndDate',  
PnBase.De2dDrawingCheckPlanStartDate AS 'PnBaseDe2dDrawingCheckPlanStartDate',  
PnBase.De2dDrawingCheckPlanEndDate AS 'PnBaseDe2dDrawingCheckPlanEndDate',  
PnBase.De2dDrawingCheckRealStartDate AS 'PnBaseDe2dDrawingCheckRealStartDate',  
PnBase.De2dDrawingCheckRealEndDate AS 'PnBaseDe2dDrawingCheckRealEndDate',  
PnBase.De2dDrawProcessingPlanStartDate AS  
'PnBaseDe2dDrawProcessingPlanStartDate',  
PnBase.De2dDrawProcessingPlanEndDate AS 'PnBaseDe2dDrawProcessingPlanEndDate',  
PnBase.De2dDrawProcessingRealStartDate AS  
'PnBaseDe2dDrawProcessingRealStartDate',  
PnBase.De2dDrawProcessingRealEndDate AS 'PnBaseDe2dDrawProcessingRealEndDate',  
PnBase.De2dFinalCheckPlanStartDate AS 'PnBaseDe2dFinalCheckPlanStartDate',  
PnBase.De2dFinalCheckPlanEndDate AS 'PnBaseDe2dFinalCheckPlanEndDate',  
PnBase.De2dFinalCheckRealStartDate AS 'PnBaseDe2dFinalCheckRealStartDate',  
PnBase.De2dFinalCheckRealEndDate AS 'PnBaseDe2dFinalCheckRealEndDate',  
PnBase.De2dProcessIssueRealEndDate AS 'PnBaseDe2dProcessIssueRealEndDate',  
PnBase.De2dBomCreationPlanStartDate AS 'PnBaseDe2dBomCreationPlanStartDate',  
PnBase.De2dBomCreationPlanEndDate AS 'PnBaseDe2dBomCreationPlanEndDate',  
PnBase.De2dBomCreationRealStartDate AS 'PnBaseDe2dBomCreationRealStartDate',  
PnBase.De2dBomCreationRealEndDate AS 'PnBaseDe2dBomCreationRealEndDate',  
PnBase.De2dProcessIssuePlanStartDate AS 'PnBaseDe2dProcessIssuePlanStartDate',  
PnBase.De2dProcessIssuePlanEndDate AS 'PnBaseDe2dProcessIssuePlanEndDate',  
PnBase.De2dProcessIssueRealStartDate AS 'PnBaseDe2dProcessIssueRealStartDate',  
PnBase.De2dRequestDate AS 'PnBaseDe2dRequestDate',



```

PnBase.De2dReceivedDate AS 'PnBaseDe2dReceivedDate',

PnBase.EcrQuotDetailPlanStartDate AS 'PnBaseEcrQuotDetailPlanStartDate',
PnBase.EcrQuotDetailPlanEndDate AS 'PnBaseEcrQuotDetailPlanEndDate',
PnBase.EcrQuotDetailRealStartDate AS 'PnBaseEcrQuotDetailRealStartDate',
PnBase.EcrQuotDetailRealEndDate AS 'PnBaseEcrQuotDetailRealEndDate',
PnBase.EcrQuotAddDelCheckerPlanStartDate AS
'PnBaseEcrQuotAddDelCheckerPlanStartDate',
PnBase.EcrQuotAddDelCheckerPlanEndDate AS
'PnBaseEcrQuotAddDelCheckerPlanEndDate',
PnBase.EcrQuotAddDelCheckerRealStartDate AS
'PnBaseEcrQuotAddDelCheckerRealStartDate',
PnBase.EcrQuotAddDelCheckerRealEndDate AS
'PnBaseEcrQuotAddDelCheckerRealEndDate'

FROM
dbo.CarMaker INNER JOIN
dbo.CarModel ON dbo.CarMaker.Id = dbo.CarModel.CarMakerId INNER JOIN
dbo.CarModelYear ON dbo.CarModel.Id = dbo.CarModelYear.CarModelId INNER JOIN
dbo.CarModelYearPhase ON dbo.CarModelYear.Id =
dbo.CarModelYearPhase.CarModelYearId INNER JOIN
dbo.Phase ON dbo.Phase.Id = dbo.CarModelYearPhase.PhaseId INNER JOIN
dbo.PnBase ON dbo.CarModelYearPhase.Id = dbo.PnBase.CarModelYearPhaseId INNER
JOIN
dbo.Activity ON dbo.Activity.Id = dbo.PnBase.ActivityId
) AS vwBase INNER JOIN
dbo.Pn ON vwBase.Id = dbo.Pn.PnBaseId
) AS vwBaseAndPns ON vwBaseAndPns.PnId = dbo.Task.PnId INNER JOIN
CoreUser ON dbo.Task.CoreUserId = CoreUser.Id INNER JOIN
PlanTaskStatus ON task.planTaskStatusId = PlanTaskStatus.Id INNER JOIN
Activity on vwBaseAndPns.ActivityId = Activity.Id

UNION ALL
SELECT *,
NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,
NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,
NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL,
NULL

FROM (
SELECT DISTINCT
Task.Id AS 'TaskId',
Task.PnBaseId AS 'TaskPnBaseId',
Task.PnId AS 'TaskPnId',
Task.Name AS 'TaskName',
Task.FriendlyName AS 'TaskFriendlyName',
Task.SortKey AS 'TaskSortKey',
Task.PlanStartDate AS 'TaskPlanStartDate',
Task.PlanEndDate AS 'TaskPlanEndDate',
Task.RealStartDate AS 'TaskRealStartDate',
Task.RealEndDate AS 'TaskRealEndDate',
Task.CoreUserId AS 'TaskCoreUserId',
Task.OverdueReason AS 'TaskOverdueReason',
Task.Notes AS 'TaskNotes',
Task.Qty AS 'TaskQty',
PlanTaskStatus.nameSimple AS 'TaskStatusName',
PlanTaskStatus.name AS 'TaskStatusNameDetail',
PlanTaskStatus.number AS 'TaskStatusNumber',
PlanTaskStatus.numberSimple AS 'TaskStatusNumberSimple',
PlanTaskStatus.color AS 'TaskStatusColor',
CoreUser.FirstName + ' ' + CoreUser.LastName AS 'TaskCoreUserFullName',
Activity.SearchByClient AS 'ActivitySearchByClient',
vwBase.*

FROM
dbo.Task INNER JOIN

```

```

SELECT
    (
        CarMaker.Id AS 'CarMakerId',
        CarMaker.Name AS 'CarMakerName',

        CarModel.Id AS 'CarModelId',
        CarModel.Name AS 'CarModelName',

        CarModelYear.Id AS 'CarModelYearId',
        CarModelYear.Name AS 'CarModelYearName',

        Phase.Id AS 'PhaseId',
        Phase.Name AS 'PhaseName',

        CarModelYearPhase.Id AS 'CarModelYearPhaseId',

        Activity.Id AS 'ActivityId',
        Activity.Name AS 'ActivityName',
        Activity.KeyName AS 'ActivityKeyName',

        -- required to include in our Core
        PnBase.Id,
        PnBase.CreatedBy,
        PnBase.CreatedOn,
        PnBase.ModifiedBy,
        PnBase.ModifiedOn,
        PnBase.RowNotes,
        PnBase.RowVersion,

        PnBase.Name AS 'PnBaseName',
        PnBase.YazakiNumber AS 'PnBaseYazakiNumber',
        PnBase.YazakiLevel AS 'PnBaseYazakiLevel',
        PnBase.ClientNumber AS 'PnBaseClientNumber',
        PnBase.ClientLevel AS 'PnBaseClientLevel',
        PnBase.HoursPlanned AS 'PnBaseHoursPlanned',
        PnBase.RequestNumber AS 'PnBaseRequestNumber',
        PnBase.TechnicalChangeNumber AS 'PnBaseTechnicalChangeNumber',
        PnBase.PhaseInternalName AS 'PnBasePhaseInternalName',

        PnBase.CoreUserIdCreator AS 'PnBaseCreator_UserId',
        (select FirstName + ' ' + LastName from CoreUser Where CoreUser.Id =
PnBase.CoreUserIdCreator) AS 'PnBaseCreator_FullName',

        PnBase.CoreUserIdResponsible AS 'PnBaseResponsible_UserId',
        (select FirstName + ' ' + LastName from CoreUser Where CoreUser.Id =
PnBase.CoreUserIdResponsible) AS 'PnBaseResponsible_FullName',

        PnBase.CoreUserIdChecker AS 'PnBaseChecker_UserId',
        (select FirstName + ' ' + LastName from CoreUser Where CoreUser.Id =
PnBase.CoreUserIdChecker) AS 'PnBaseChecker_FullName',

        PnBase.SwoId AS 'PnBaseSwoId',
        PnBase.SwoInternalNumber AS 'PnBaseSwoInternalNumber',
        PnBase.QtyCircuits AS 'PnBaseQtyCircuits',
        PnBase.ComplexityRange AS 'PnBaseComplexityRange',
        PnBase.DcDsQty AS 'PnBaseQtyDcDs',
        PnBase.DcDsQtyPnsAffected AS 'PnBaseDcDsQtyPnsAffected',
        PnBase.DcDsPnsAffected AS 'PnBaseDcDsPnsAffected',
        PnBase.MdFilename AS 'PnBaseMdFilename',
        PnBase.DrawingReleaseDate AS 'PnBaseDrawingReleaseDate',
        PnBase.SocleOrModules AS 'PnBaseSocleOrModules',
        PnBase.Notes AS 'PnBaseNotes',
        PnBase.LastECRProcessed AS 'PnBaseLastECRProcessed',
    )

```

```

PnBase.AffectedPns AS 'PnBaseAffectedPns',
PnBase.LevelForTp AS 'PnBaseLevelForTp',
PnBase.Obsolete AS 'PnBaseObsolete',

PnBase.DrawingAnalysisPlanStartDate AS 'PnBaseDrawingAnalysisPlanStartDate',
PnBase.DrawingAnalysisPlanEndDate AS 'PnBaseDrawingAnalysisPlanEndDate',
PnBase.DrawingAnalysisRealStartDate AS 'PnBaseDrawingAnalysisRealStartDate',
PnBase.DrawingAnalysisRealEndDate AS 'PnBaseDrawingAnalysisRealEndDate',
PnBase.PdInputPlanStartDate AS 'PnBasePdInputPlanStartDate',
PnBase.PdInputPlanEndDate AS 'PnBasePdInputPlanEndDate',
PnBase.PdInputRealStartDate AS 'PnBasePdInputRealStartDate',
PnBase.PdInputRealEndDate AS 'PnBasePdInputRealEndDate',
PnBase.PdCheckerPlanStartDate AS 'PnBasePdCheckerPlanStartDate',
PnBase.PdCheckerPlanEndDate AS 'PnBasePdCheckerPlanEndDate',
PnBase.PdCheckerRealStartDate AS 'PnBasePdCheckerRealStartDate',
PnBase.PdCheckerRealEndDate AS 'PnBasePdCheckerRealEndDate',
PnBase.DpPlanStartDate AS 'PnBaseDpPlanStartDate',
PnBase.DpPlanEndDate AS 'PnBaseDpPlanEndDate',
PnBase.DpRealStartDate AS 'PnBaseDpRealStartDate',
PnBase.DpRealEndDate AS 'PnBaseDpRealEndDate',
PnBase.MdPlanStartDate AS 'PnBaseMdPlanStartDate',
PnBase.MdPlanEndDate AS 'PnBaseMdPlanEndDate',
PnBase.MdRealStartDate AS 'PnBaseMdRealStartDate',
PnBase.MdRealEndDate AS 'PnBaseMdRealEndDate',

PnBase.AddDelSheetReleasePlanStartDate AS
'PnBaseAddDelSheetReleasePlanStartDate',
PnBase.AddDelSheetReleasePlanEndDate AS 'PnBaseAddDelSheetReleasePlanEndDate',
PnBase.AddDelSheetReleaseRealStartDate AS
'PnBaseAddDelSheetReleaseRealStartDate',
PnBase.AddDelSheetReleaseRealEndDate AS 'PnBaseAddDelSheetReleaseRealEndDate',

PnBase.De2dDrawAnalysisPlanStartDate AS 'PnBaseDe2dDrawAnalysisPlanStartDate',
PnBase.De2dDrawAnalysisPlanEndDate AS 'PnBaseDe2dDrawAnalysisPlanEndDate',
PnBase.De2dDrawAnalysisRealStartDate AS 'PnBaseDe2dDrawAnalysisRealStartDate',
PnBase.De2dDrawAnalysisRealEndDate AS 'PnBaseDe2dDrawAnalysisRealEndDate',
PnBase.De2dDrawInputDsiPlanStartDate AS 'PnBaseDe2dDrawInputDsiPlanStartDate',
PnBase.De2dDrawInputDsiPlanEndDate AS 'PnBaseDe2dDrawInputDsiPlanEndDate',
PnBase.De2dDrawInputDsiRealStartDate AS 'PnBaseDe2dDrawInputDsiRealStartDate',
PnBase.De2dDrawInputDsiRealEndDate AS 'PnBaseDe2dDrawInputDsiRealEndDate',
PnBase.De2dDrawingCheckPlanStartDate AS 'PnBaseDe2dDrawingCheckPlanStartDate',
PnBase.De2dDrawingCheckPlanEndDate AS 'PnBaseDe2dDrawingCheckPlanEndDate',
PnBase.De2dDrawingCheckRealStartDate AS 'PnBaseDe2dDrawingCheckRealStartDate',
PnBase.De2dDrawingCheckRealEndDate AS 'PnBaseDe2dDrawingCheckRealEndDate',
PnBase.De2dDrawProcessingPlanStartDate AS
'PnBaseDe2dDrawProcessingPlanStartDate',
PnBase.De2dDrawProcessingPlanEndDate AS 'PnBaseDe2dDrawProcessingPlanEndDate',
PnBase.De2dDrawProcessingRealStartDate AS
'PnBaseDe2dDrawProcessingRealStartDate',
PnBase.De2dDrawProcessingRealEndDate AS 'PnBaseDe2dDrawProcessingRealEndDate',
PnBase.De2dFinalCheckPlanStartDate AS 'PnBaseDe2dFinalCheckPlanStartDate',
PnBase.De2dFinalCheckPlanEndDate AS 'PnBaseDe2dFinalCheckPlanEndDate',
PnBase.De2dFinalCheckRealStartDate AS 'PnBaseDe2dFinalCheckRealStartDate',
PnBase.De2dFinalCheckRealEndDate AS 'PnBaseDe2dFinalCheckRealEndDate',
PnBase.De2dProcessIssueRealEndDate AS 'PnBaseDe2dProcessIssueRealEndDate',
PnBase.De2dBomCreationPlanStartDate AS 'PnBaseDe2dBomCreationPlanStartDate',
PnBase.De2dBomCreationPlanEndDate AS 'PnBaseDe2dBomCreationPlanEndDate',
PnBase.De2dBomCreationRealStartDate AS 'PnBaseDe2dBomCreationRealStartDate',
PnBase.De2dBomCreationRealEndDate AS 'PnBaseDe2dBomCreationRealEndDate',
PnBase.De2dProcessIssuePlanStartDate AS 'PnBaseDe2dProcessIssuePlanStartDate',
PnBase.De2dProcessIssuePlanEndDate AS 'PnBaseDe2dProcessIssuePlanEndDate',
PnBase.De2dProcessIssueRealStartDate AS 'PnBaseDe2dProcessIssueRealStartDate',
PnBase.De2dRequestDate AS 'PnBaseDe2dRequestDate',

```

```

PnBase.De2dReceivedDate AS 'PnBaseDe2dReceivedDate',

PnBase.EcrQuotDetailPlanStartDate AS 'PnBaseEcrQuotDetailPlanStartDate',
PnBase.EcrQuotDetailPlanEndDate AS 'PnBaseEcrQuotDetailPlanEndDate',
PnBase.EcrQuotDetailRealStartDate AS 'PnBaseEcrQuotDetailRealStartDate',
PnBase.EcrQuotDetailRealEndDate AS 'PnBaseEcrQuotDetailRealEndDate',
PnBase.EcrQuotAddDelCheckerPlanStartDate AS
'PnBaseEcrQuotAddDelCheckerPlanStartDate',
PnBase.EcrQuotAddDelCheckerPlanEndDate AS
'PnBaseEcrQuotAddDelCheckerPlanEndDate',
PnBase.EcrQuotAddDelCheckerRealStartDate AS
'PnBaseEcrQuotAddDelCheckerRealStartDate',
PnBase.EcrQuotAddDelCheckerRealEndDate AS
'PnBaseEcrQuotAddDelCheckerRealEndDate'

FROM
dbo.CarMaker INNER JOIN
dbo.CarModel ON dbo.CarMaker.Id = dbo.CarModel.CarMakerId INNER JOIN
dbo.CarModelYear ON dbo.CarModel.Id = dbo.CarModelYear.CarModelId INNER JOIN
dbo.CarModelYearPhase ON dbo.CarModelYear.Id =
dbo.CarModelYearPhase.CarModelYearId INNER JOIN
dbo.Phase ON dbo.Phase.Id = dbo.CarModelYearPhase.PhaseId INNER JOIN
dbo.PnBase ON dbo.CarModelYearPhase.Id = dbo.PnBase.CarModelYearPhaseId INNER
JOIN
dbo.Activity ON dbo.Activity.Id = dbo.PnBase.ActivityId
) AS vwBase ON vwBase.Id = dbo.Task.PnBaseId INNER JOIN
CoreUser ON dbo.Task.CoreUserId = CoreUser.Id INNER JOIN
PlanTaskStatus ON task.planTaskStatusId = PlanTaskStatus.Id INNER JOIN
Activity on vwBase.ActivityId = Activity.Id

WHERE
Task.PnId IS NULL

) AS T

```