

# Web OLAP, Exploração de Sistemas OLAP via Web

Afonso Arriaga e Pedro Marques

Departamento de Informática, Universidade do Minho  
<http://di.uminho.pt/>

[afonso.arriaga@gmail.com](mailto:afonso.arriaga@gmail.com), [pckmarkes@gmail.com](mailto:pckmarkes@gmail.com)

**Resumo.** De há algum tempo a esta parte, os sistemas OLAP são utilizados de forma bastante generalizada pelas grandes organizações pois conferem uma mais-valia a nível de poder de decisão. Muitas são as abordagens possíveis como forma de acesso aos dados com uma componente comum: a necessidade de uma instalação física de software na máquina do cliente. Com o decorrer dos tempos e o aparecimento de um novo paradigma, o da Web, grande foi a pressão das organizações para que os seus sistemas de suporte à decisão beneficiassem desta nova visão. A solução encontrada, relativamente aos sistemas OLAP, foi a de criar interfaces de utilização independentes da plataforma e sem necessidade de instalação de software no cliente. Surgiu assim o Web-based OLAP. Neste trabalho serão analisadas algumas ferramentas que permitem este tipo de acesso aos dados sendo apenas necessário um Web *browser* convencional e uma ligação à Internet.

**Keywords:** OLAP, Web-based OLAP, MDX, XMLA, Business Intelligence, Openi, Pentaho BI Suite, SpagoBI, Palo Web Client.

## 1 Introdução

Desde muito cedo, foi perceptível no seio das grandes organizações, a necessidade de uma ferramenta que possibilitasse uma análise de grandes volumes de dados. Devido à complexidade das *queries* necessárias para recolher informação em base de dados tradicionais de grandes dimensões e ao seu tempo de execução, surgiu uma nova arquitectura de armazenamento de dados, denominada Data Warehouse. Desta visão faz parte um modelo de dados multidimensional que permite uma análise dos dados segundo as mais variadas perspectivas e que confere aos decisores um maior poder. Para responder rapidamente a *queries* analíticas em dados assentes neste modelo, surgiu o OLAP (OnLine Analytical Processing); termo que peca por não ser suficientemente explícito. Segundo o “The OLAP Report” [1] uma ferramenta OLAP terá de obedecer a cinco critérios conhecidos como FASMI (F-Fast, A-Analysis, S-Shared, M-Multidimensional, I-Information) que definem as capacidades exigidas à ferramenta relativamente à sua velocidade de resposta, capacidade de análise segundo as mais variadas perspectivas, segurança dos dados, suporte à multidimensionalidade e informação acrescida para o sistema.

Este tipo de sistema permitiu, desde logo, aos analistas uma nova abordagem aos seus problemas do dia-a-dia, uma vez que permite efectuar uma exploração dos dados

segundo as mais vastas perspectivas de análise do negócio. Uma das principais vantagens desta abordagem consiste no facto destes sistemas possuírem a informação da instituição de uma forma centralizada e consolidada segundo as regras do negócio.

Actualmente, com o desenvolvimento das tecnologias Web, nomeadamente do Java e do Ajax que permitem uma maior e mais dinâmica interacção com o utilizador, a maioria das empresas de software que desenvolvem ferramentas para OLAP integram nas suas *suites* uma plataforma Web para os clientes acederem aos seus dados através de um Web browser. Estas ferramentas são conhecidas por Web-based OLAP (WOLAP).

A utilização de ferramentas Web-based OLAP permite que o acesso aos dados e o seu processamento analítico possa ser feito através de qualquer computador com acesso de rede ao servidor OLAP, sem a necessidade de instalação dum *software* cliente. Para além da flexibilidade geográfica proporcionada por esta arquitectura, o custo de manutenção tende a ser inferior a um sistema tradicional de aplicação-servidor, uma vez que só é necessário configurar e actualizar o servidor que fornece este serviço. Esta mais-valia pode parecer irrelevante quando o número de clientes que recorrem às ferramentas OLAP é reduzido. No entanto, para um número elevado de clientes, a mais-valia torna-se notável.

## 2 Arquitectura de um Sistema WOLAP

Tal como um sistema OLAP convencional, a abordagem WOLAP visa permitir a análise de grandes volumes de dados de uma instituição de forma simples e célere a quem dela necessitar. A principal diferença que demarca estes sistemas WOLAP prende-se como facto de esta análise não necessitar de um cliente OLAP instalado na máquina do cliente, sendo apenas necessário um Web *browser* que lhe permita aceder aos dados através da Internet. Todos os cálculos e manipulação de dados são portanto realizados pelo servidor, sendo apenas devolvidas ao cliente páginas Web com informação processada. É por isso necessário compreender de que forma é possível comunicar com o servidor OLAP.

A linguagem de *query* utilizada pela maioria dos servidores OLAP é a MDX[2] (Multidimensional Expressions); da mesma forma que a SQL é utilizada para base de dados relacionais.

Em 2000, a Microsoft propôs a adopção de um standard para o acesso a dados em sistemas analíticos como o OLAP. Esse standard, denominado XMLA[3], é baseado em standards já existentes na indústria como o XML, o SOAP e o HTTP. A primeira versão das especificações do XMLA foi aprovada em 2001 e actualmente é mantida pelo XMLA Council, ao qual pertencem agora mais de 25 empresas. A versão actual das especificações do XMLA (versão 1.1), inclui a mdXML como linguagem de *query*. A mdXML é de forma sumária a MDX envolvida por uma tag <Statement>.

Uma vez que não é possível executar uma *query* MDX directamente a partir duma página estática num browser, nem o browser seria capaz de representar de forma eficaz informação escrita em XMLA, torna-se essencial a utilização de um software que funcione como *middleware* entre o servidor Web e o servidor OLAP. Como é

descrito mais à frente neste relatório, todas as ferramentas estudadas utilizam a tecnologia Java Server Pages (JSP)[4]. Esta tecnologia recorre a um *servlet container* encarregue de executar o código e gerar dinamicamente o HTML. É de salientar que as ferramentas estudadas não foram seleccionadas com base na tecnologia utilizada, tratando-se apenas de uma exemplificação.

Para além de *servlets*, outros tipos de *middleware* podem ser utilizados (como por exemplo *scripts CGI*). De forma genérica e simplificada, a arquitectura dum sistema WOLAP pode ser representada de acordo com a representação da Figura 1.

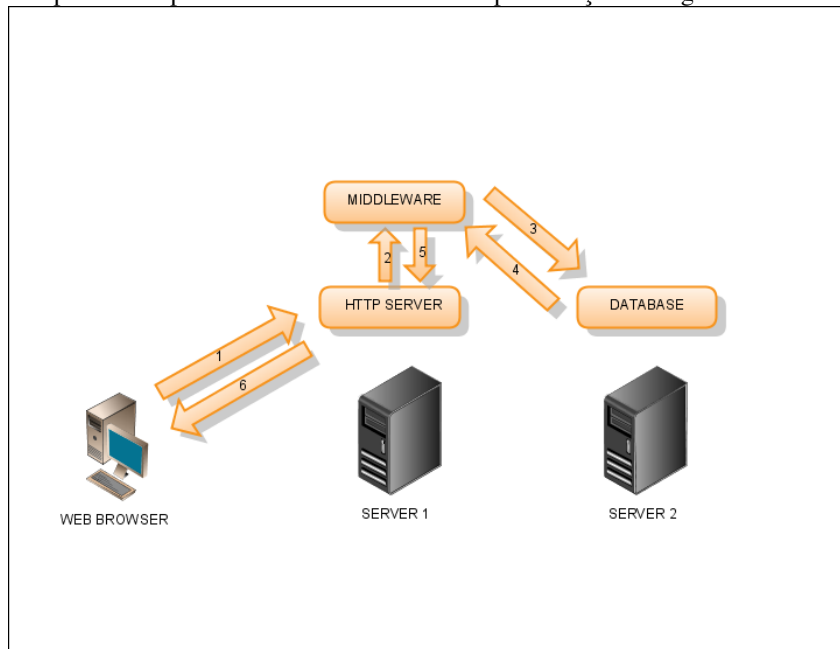


Figura 1 - Fluxo de informação num sistema WOLAP

### 3 Caching em WOLAP

A um sistema OLAP comum encontra-se normalmente associado um sistema de *caching* que é utilizado com o objectivo de diminuir o número de acessos desnecessários ao servidor de dados, aumentando assim o desempenho global do sistema. Neste tipo de abordagem os mecanismos são variados, desde simples *caches* a nível do *browser* (que apenas beneficia o cliente se este repetir *queries* que ele próprio executou), a mecanismos complexos que envolvem vários servidores que actuam como *proxies* e que realizam *caching* das *queries* efectuadas. Estes servidores

comunicam entre si de forma a melhorar o desempenho e distribuir a carga, minimizando, sempre que possível, o acesso ao Servidor de Gestão de Base de Dados (SGBD). Estes servidores têm o nome de OLAP Cache Servers (OCS) e possuem, também eles, capacidade de derivar dados através dos resultados guardados previamente. Isto é, têm a capacidade de manipulação dos dados tal como um servidor OLAP convencional[5].

Nos sistemas WOLAP em particular, com o desenvolvimento de soluções de *middleware* mais sofisticadas, foi possível implementar mecanismos de *caching* bastante sofisticados. É neste ponto que se encontram, na maioria dos sistemas, as funcionalidades responsáveis por devolver os resultados de *queries* já realizadas anteriormente. Estes sistemas de *caching* podem ser distribuídos por diversas máquinas. A fim de manter a informação de forma coerente nos diversos servidores, é necessária a utilização de um algoritmo de gestão de OCS (tanto para arquitecturas centralizadas, semi-centralizadas como autónomas)[5].

#### 4 Análise de Ferramentas WOLAP

Neste momento existe um grande número de empresas de software que desenvolvem soluções Web-based OLAP. Como não é razoável analisar todas as ferramentas disponíveis no mercado, fez-se uma selecção com base em critérios descritos mais à frente neste capítulo.

Para começar, foi escolhido o servidor OLAP com o qual se iria efectuar a análise das ferramentas. Procurou-se um servidor OLAP que executasse *queries* MDX ou mdXML e que funcionasse como *XMLA provider*, obedecendo assim às especificações do XMLA Council. De acordo com este critério, apareceram vários servidores OLAP como o Microsoft Analysis Services, o Mondrian e o SAP BW. Optou-se pelo Mondrian por ser *open-source* e gratuito. A fim de obter um sistema completamente funcional, o servidor OLAP tem de recolher os dados num Data Warehouse. Para isso, fez-se o load do Data Warehouse Foodmart no SGBD PostgreSQL.

Um dos desafios por nós imposto era o de verificar se seria possível construir um sistema OLAP completamente assente em software *open-source* (desde o sistema operativo, SGBD e servidor OLAP até às ferramentas Web disponíveis para os clientes) e se as ferramentas disponíveis teriam a mesma qualidade de soluções comerciais conhecidas como as ferramentas Web disponíveis pela Dundas. Com base neste critério e no tamanho das comunidades que suportam os *softwares*, seleccionaram-se, para análise, as seguintes ferramentas WOLAP: OpenI, Palo Web Client, SpagoBI e Pentaho BI Suite. Esta última ferramenta é desenvolvida pela mesma empresa que desenvolveu o Mondrian (servidor OLAP escolhido para a análise) e apresenta igualmente uma versão comercial com maior suporte por parte da Pentaho.

As ferramentas OpenI e Palo Web Client são apenas peças de software que permitem a análise dos dados através de um *browser* Web, enquanto as restantes ferramentas SpagoBI e Pentaho BI Suite são *suites* completas de Business

Intelligence. Estas suites englobam desde módulos de geração de relatórios a ferramentas de Data Mining.

#### 4.1 OpenI

O OpenI[6] é uma aplicação J2EE que, por omissão, corre sobre Apache Tomcat ou JBoss. Actua como middleware entre o servidor Web e o servidor OLAP e é constituído maioritariamente por 4 componentes:

- Conectores – servem de ponte de comunicação entre o OpenI e as diversas fontes de dados onde este se pode ligar. O protocolo utilizado para efectuar a ligação a bases de dados relacionais é o JDBC. Já para fontes de dados OLAP é utilizado XMLA existindo ainda a possibilidade de integrar com o “the R Project”[7] através de uma API de nome RServe.
- Definição de relatórios – este componente fornece a capacidade de definir, editar e executar relatórios sobre os dados existentes e são utilizados, sempre que possível, standards como JRXML sendo apenas implementado um protocolo de comunicação para relatórios sobre OLAP e Data Mining mas que é público e de código aberto.
- Comunicação com o utilizador – nesta camada são amplamente utilizados componentes de JPivot[8] e JFreeChart[9], unificados numa mesma *framework*. São ainda aproveitadas algumas das características do JPivot de forma a facilitar a integração desta *framework* em portais já existentes nas organizações.
- Segurança – Nesta *framework* são utilizados mecanismos de autenticação integrados na estrutura de segurança do J2EE. É ainda possível definir padrões de segurança a nível das fontes de dados para uma mais fina gestão das questões de segurança.

Todos estes componentes estão interligados e têm fins específicos como é demonstrado no esquema da arquitectura da ferramenta (Figura 2).

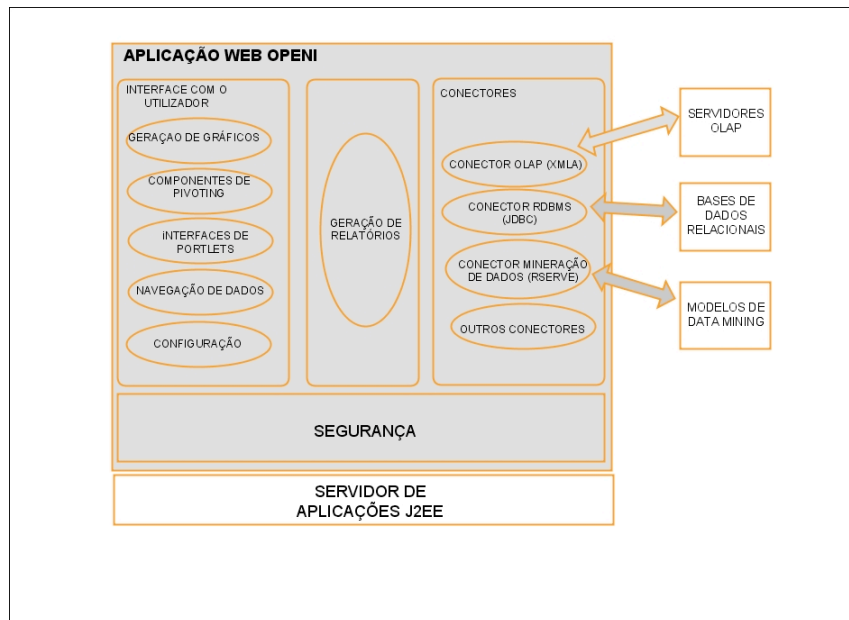


Figura 2 - Arquitectura da ferramenta OpenI

Esta ferramenta dispõe de uma demonstração online que permite que o cliente tenha uma noção de qual será a forma de interagir com os dados. No caso da ferramenta OpenI, esta demonstração encontra-se disponível em <http://demo.openi.org/openi/>.

#### 4.2 Palo Client

A Palo Web Client[10], ferramenta da Tensegrity Software, permite o acesso aos dados sem necessitar de instalação de software cliente, sendo apenas necessário o recurso a um Web browser. São permitidas na interface Web todas as operações de navegação disponibilizadas pelo *thick-client* da mesma empresa, o Palo Client. O modo de visualização de dados é em tudo semelhante, não sendo apenas possível na parte Web realizar operações de modelação dimensional. Estas ferramentas são disponibilizadas segundo a licença GNU General Public License [11] o que faz com que seja gratuita, aberta ao público e de alteração livre.

A ferramenta Web Client é baseada em AJAX, melhorando assim o desempenho desta, necessita apenas de um Tomcat Applications Server do lado do servidor e permite a conexão a um servidor Palo Server e ainda, a bases de dados como Microsoft Analysis Services, SAP BW ou Mondrian, através de uma ligação XMLA.

A ferramenta apresenta um User Interface bastante intuitivo e de fácil utilização de forma a retirar o melhor das funcionalidades que vão desde a simples navegação em

hierarquias dentro de uma dimensão, à mudança de eixo de análise com apenas um *drag-and-drop* passando pela edição de vistas criadas previamente no Palo Client ou Palo Web Client, permitindo ao analista observar vários cubos ou vistas do mesmo cubo em simultâneo através da utilização de *tabs*[12].

Também nesta ferramenta é possível executar uma demonstração online ainda que muito básica, de forma a que o cliente possa ter um primeiro contacto com a forma de interacção desta através da sua interface Web. Esta demonstração está disponível em [http://www.jpalo.com/en/products/palo\\_web\\_client.html#Demo](http://www.jpalo.com/en/products/palo_web_client.html#Demo).

### 4.3 SpagoBI

A ferramenta SpagoBI[13] é uma de muitas soluções na área do Business Intelligence da família Spago. Estas ferramentas são de código livre segundo as normas da GNU Lesser General Public License e recorrem a outras peças de software, que seguem o mesmo espírito, como uma série de módulos que são assim integrados numa única framework. As camadas da aplicação SpagoBI e respectivas peças de software externo utilizadas são:

- Framework de aplicação.
- Repositório de conteúdos – é utilizado Apache Jackrabbit[14] que é um repositório de conteúdos para Java Technology API.
- Pesquisa – nesta camada da aplicação, é utilizado o motor de pesquisa Apache Lucene[15] que é, também este, de código aberto e inteiramente escrito em java.
- Geração de relatórios – o gerador de dados escolhido é de código aberto e tem o nome de JasperReport[16], da empresa JasperSoft, e tem como principais funcionalidades a geração de relatórios e possível exportação para diversos formatos (.pdf, .html, .xml, etc...), a possibilidade de fazer convergir informações de várias fontes e a possibilidade de definição de *scriplets* (pequenas peças de código escritas em Java que realizam algum tipo de processamento adicional necessário).
- Cliente OLAP – o cliente OLAP utilizado é o JPivot[8] que é responsável pela ligação ao servidor OLAP e pelas operações básicas de roll-up/drill-down, etc.
- Data Mining – o módulo de data mining é o Weka[17], um software escrito em Java, de integração livre em outras aplicações e que disponibiliza toda uma gama de algoritmos nesta área.
- Ferramenta de ETL – é utilizada a ferramenta Octopus[18], escrita em Java, que permite definir o funcionamento da fase de extracção, transformação e carregamento dos dados.
- Ferramenta de administração – o componente responsável pela gestão e calendarização de tarefas é o Quartz[19].

- Interface Web – o portal utilizado é o eXo[20] que disponibiliza a interface Web de acesso aos serviços disponibilizados permitindo algumas configurações a nível de *skins* e a adição de *gadgets* escritas segundo as normas da Google (Google Gadgets Specification).

Na página oficial deste produto, como em outros casos, é possível aceder a uma demonstração online das capacidades deste sistema. O endereço onde podemos aceder a esta demonstração é:

<http://spagobi.eng.it/ecm/faces/public/guest/home/solutions/spagobi>

#### 4.4 Pentaho BI Suite

O projecto da Pentaho para Business Intelligence, de nome Pentaho BI Suite[21], é uma ferramenta que, por si só, disponibiliza todas as funcionalidades que se podem exigir a uma ferramenta de análise de dados multidimensional. Da simples geração de relatórios à possibilidade de operar sobre os dados utilizando algoritmos de mineração de dados, passando pela possibilidade de análise ad-hoc dos dados. Tudo isto é permitido nesta ferramenta que engloba vários módulos *open source*, muitos deles, desenvolvidos pela própria Pentaho, como é exemplo o servidor de dados OLAP utilizado, o Mondrian[22] ou o módulo responsável pela geração de relatórios, o Pentaho Reporting[23].

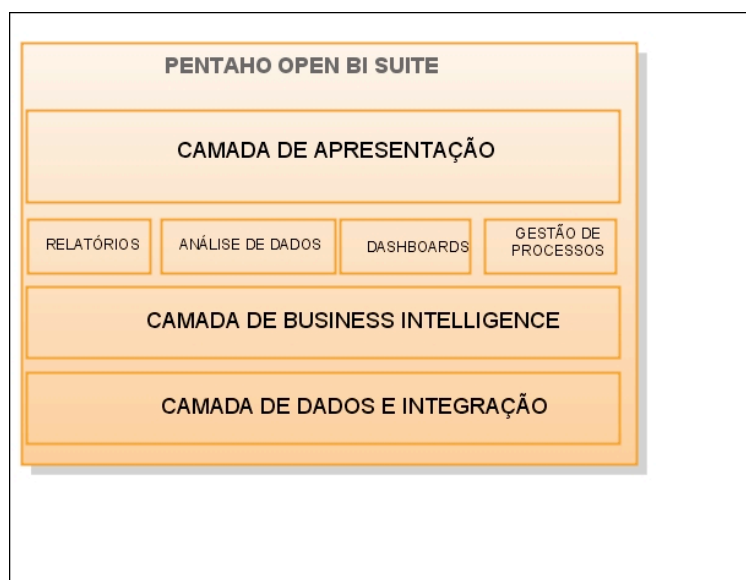


Figura 3 - Arquitectura Pentaho Open BI Suite



A arquitectura modular seguida por este projecto, representada na Figura 3, como é apanágio dos projectos open source de alguma dimensão, permite o reaproveitamento do código de muitos destes módulos para outras aplicações. Os módulos que servem de base a esta suite são os seguintes:

- Servidor OLAP – o servidor OLAP utilizado é o Mondrian[22], desenvolvido também pela Pentaho e que permite as operações de navegação dos dados.
- Automação e gestão do processo de ETL – este módulo dá pelo nome de Kettle Pentaho Data Integration[24] e, como o próprio nome indica, é um software desenvolvido pela Pentaho que gere o processo de extração, transformação e integração dos dados no Data Warehouse.
- Geração de relatórios – o módulo responsável pela geração de relatórios é o Pentaho Reporting[23], anteriormente conhecido como JFreeReport.
- Mineração de dados – é utilizada a ferramenta Weka[17], totalmente escrita em Java, que disponibiliza um grande número de algoritmos de mineração de dados e é mantido pela Universidade de Waikato.

A camada de interface de comunicação com o utilizador é desenvolvida pela Pentaho e pode ser configurada e parametrizada à medida das necessidades de cada organização ou mesmo de cada utilizador. Nesta aplicação podem ser definidas vistas sobre os dados que podem ser guardadas para posterior visualização. A utilização de *dashboards* é possível e de simples criação associando a isto a possibilidade de medição de KPI (Key Performance Indicators) adaptados à lógica de negócio de cada caso. Existe uma demonstração online das potencialidades deste sistema em <http://demo.pentaho.com/pentaho/> onde é possível ter uma ideia de como tudo isto funcionará uma vez instalado na organização do cliente.

## 5 Análise comparativa entre as diversas soluções

Após analisar as várias ferramentas e o seu modo de funcionamento fica a sensação que a evolução destas se dá num mesmo sentido, o de tornar a análise dos dados cada vez mais intuitiva e mais agradável à vista. Nota-se, da parte das empresas que desenvolvem estas soluções, uma preocupação crescente em cativar os seus clientes não só através dos bons desempenhos da sua solução como também de uma boa apresentação visual dos dados, de forma intuitiva e muito gráfica. Esta preocupação resultou numa grande evolução na forma como o utilizador interage com as informações visuais que lhe são transmitidas, desde operações de *drill-down* através de um simples *click* num gráfico, de forma a analisar algum pormenor mais relevante,

à alteração da escala num gráfico de linhas ou barras apenas por arrastamento desta, etc. A utilização de *dashboards* também se encontra cada vez mais em destaque pois esta é uma forma muito agradável e simples de visualizar e monitorizar os valores de algumas variáveis relacionadas com o desempenho da organização (frequentemente designadas como KPI ou Key Performance Indicators), numa mesma localização, embora muitas vezes esta informação provenha de vários locais distintos. Os *dashboards* são frequentemente um conjunto de gráficos que ilustram o comportamento ao longo do tempo de uma determinada característica, por exemplo, a evolução das vendas de um determinado produto em função do que seria expectável.

De entre estas duas soluções completas para BI a escolha é necessariamente difícil até pelo facto de alguns dos seus módulos integrantes serem, quando não os mesmos, muito semelhantes. Sendo esta uma característica fundamental de um sistema deste género, esta opção posiciona as duas ferramentas num patamar muito semelhante. No que diz respeito à camada de análise de dados, tanto na suite SpagoBI, como na Pentaho BI é usado o componente JPivot (biblioteca escrita em Java, muito popular neste tipo de aplicações) o que, mais uma vez, revela a semelhança entre estas duas soluções. Já no que diz respeito à geração de relatórios as opções variam, sendo utilizados os módulos JasperReport na ferramenta SpagoBI e Pentaho Reporting na Pentaho BI Project. No entanto é notável o esforço que a SpagoBI faz para incluir na sua suite as mais recentes ferramentas OLAP. Um exemplo é a inclusão do próprio Palo Web Client na última versão da suite.

As ferramentas OpenI e Palo Web Client são, ao contrário das suites completas apresentadas anteriormente, apenas intermediários entre a camada de apresentação e a camada de dados, estas permitem a ligação a um servidor de dados multidimensional através do protocolo XMLA ou a um servidor de dados relacional através de JDBC bem como a muitas outras fontes de dados. A ferramenta OpenI é ainda capaz de gerar relatórios, recorrendo ao formado JRXML e posteriormente exportar para os formatos PDF, XLS, HTML, entre outros. A camada de apresentação da ferramenta OpenI é baseada em componentes como JPivot ou JFreeChart amplamente utilizados e testados por serem bibliotecas Java, de código aberto e integradas em muitas outras ferramentas desta área. Já a camada de comunicação com o utilizador na solução Palo Web Client revela uma grande simplicidade de utilização, bem como um grande cuidado na apresentação dos resultados para que estes sejam facilmente legíveis, proporcionando assim uma interacção de excelência com o utilizador.

As características mais relevantes podem ser consultadas na Tabela 1.

Tabela 1 - Comparação de características das ferramentas

Ferramenta		OpenI	Palo Web Client	Spago BI	Pentaho BI Project
VERSÃO ESTUDADA		2.0 alpha	2.0	2.0	2.0
TECNOLOGIA		JSP	JSP	JSP	JSP
FUNCIONALIDADES BÁSICAS OLAP	DRILL-DOWN	✓	✓	✓	✓
	ROLL-UP	✓	✓	✓	✓
	PIVOT	✓	✓	✓	✓
	SLICE AND DICE	✓	✓	✓	✓
FUNCIONALIDADES COMPLEMENTARES	VISUALIZAÇÃO EM TABELA	✓	✓	✓	✓
	VISUALIZAÇÃO DE GRÁFICOS	✓	✗	✓	✓
	VISUALIZAÇÃO GEOGRÁFICA	✗	✗	✓	✓
	GERAÇÃO DE RELATÓRIOS	✓	✗	✓	✓
	DASHBOARDS /KPI	✓	✗	✓	✓
FORMAS DE LIGAÇÃO		JDBC, XMLA, RServe	XMLA	XMLA	XMLA, JDBC
LICENÇA		openI Public License 1.0 (baseado em Mozilla Public License 1.1)	GPL	GNU LGPL	GPL (v2.0)
INTEGRAÇÃO TESTADA COM SERVIDORES OLAP		MONDRIAN, MS ANALYSIS SERVICES	SAP BW SERVER, MS ANALYSIS SERVICES, PALO SERVER, MONDRIAN	MS ANALYSIS SERVICES, PALO SERVER, MONDRIAN	MONDRIAN

## 6 Futuro do WOLAP

Sendo a análise multidimensional de dados uma área em franca expansão, é previsível que num curto espaço de tempo, todas as ferramentas desta área disponibilizem uma interface Web que permita aos seus utilizadores acederem aos dados independentemente da sua localização. Assim, a disseminação da tecnologia WOLAP deverá ser grande e será, certamente, o resultado do esforço das grandes empresas de Business Intelligence em satisfazerem os seus clientes, dando-lhes assim uma maior liberdade na utilização destes sistemas. Até ao momento, a evolução tem-se dado em duas direcções, a primeira no sentido de melhorar a arquitectura das soluções existentes tornando-as mais consistentes e melhorando o seu desempenho, a segunda com o objectivo de integrar todas as peças de software necessárias numa única suite de forma a fornecer o serviço como um todo facilitando assim a instalação e manutenção destes sistemas. A adopção, ou não, deste tipo de soluções pelas mais diversas organizações vai decidir-se, quanto a nós, em dois pontos fulcrais, a diminuição dos custos de implementação e manutenção e a fiabilidade fornecida pelas ferramentas adoptadas. Em ambos os pontos a evolução tem sido muito positiva e prevê bons resultados a médio prazo, tornando as soluções *open-source*, soluções cada vez mais competitivas.

## 7 Conclusão

Terminada a análise das ferramentas anteriormente apresentadas, cabe-nos apontar como ponto negativo, a dificuldade com que nos deparamos na instalação dos referidos produtos. Os manuais disponibilizados pelos criadores das ferramentas nem sempre são adequados, estando mesmo errados em alguns pontos, fruto do lançamento de novas versões das ferramentas cujos manuais não foram revistos. Alguns ficheiros foram movidos de uma localização para outra sem que os manuais reflectissem estas alterações o que agrava em muito a dificuldade em instalar este tipo de soluções. Para além da dificuldade de instalação e configuração das ferramentas, a instalação e configuração das dependências nem sempre é simples. Muitas das ferramentas exigem versões do Java e do Tomcat diferentes. Algumas, como o OpenI, exigem ainda a definição de *roles* no Tomcat.

Como aspectos positivos, foi demonstrada neste estudo a possibilidade de montagem de um sistema apenas baseado em software open-source, o que contraria o preconceito que “OLAP é caro”, tornando assim esta tecnologia mais acessível a PME que não pretendem gastar muitos recursos na informatização dum sistema de decisão para o seu negócio.

Se tivéssemos de recomendar uma das ferramentas estudadas, recomendaríamos a suite SpagoBI por estar numa fase de desenvolvimento muito activa (foram lançadas várias versões no decorrer deste estudo). disso, a comunidade que a desenvolve parece estar muito atenta ao estado da arte, uma vez que incluí rapidamente na sua suite novas ferramentas e bibliotecas relevantes para o WOLAP.

## 8 Referências

1. **Pendse, Nigel.** OLAP Report. [Online] Acedido em 10 Dez. 2008. <http://olapreport.com/fasmi.htm>.
2. **Spofford, George.** *Mdx Solutions: With Microsoft SQL Server Analysis Services*. s.l. : John Wiley & Sons Inc, 2001. 9780471400462.
3. XMLA, XML for Analysis. [Online] Acedido em 15 de Dez. de 2008. <http://www.xmlforanalysis.com/>.
4. JavaServer Pages Technology. [Online] Acedido em 11 de Dez. de 2008. <http://java.sun.com/products/jsp/>.
5. **Kalnis, Panos e Papadias, Dimitris.** *Proxy-Server Architectures for OLAP*. California, USA : s.n., 2001.
6. **OpenI.org.** Project OpenI. [Online] Acedido em 21 de 11 de 2008. [http://www.openi.org/openi\\_product.html](http://www.openi.org/openi_product.html).
7. The R Project for Statistical Computing. [Online] Acedido em 15 de Dez. de 2008. <http://www.r-project.org/>.
8. JPivot home. [Online] Acedido em 17 de Dez. de 2008. <http://jpivot.sourceforge.net/>.
9. JFreeChart. [Online] Acedido em 16 de Dez. de 2008. <http://www.jfree.org/jfreechart/>.
10. JPalo - Palo Java World. [Online] Acedido em 10 de Dez. de 2008. [http://www.jpalo.com/en/products/start\\_products.html](http://www.jpalo.com/en/products/start_products.html).
11. Palo Java World - FAQ. [Online] Acedido em 10 de Dez. de 2008. <http://www.jpalo.com/en/faq.html>.
- Dez.. Palo Web Client Features. [Online] Acedido em 10 de Dez. de 2008. [http://www.jpalo.com/en/products/palo\\_web\\_client.html#Features](http://www.jpalo.com/en/products/palo_web_client.html#Features).
13. SpagoBI. [Online] Acedido em 15 de Dez. de 2008. <http://www.spagobi.com/ecm/faces/public/guest/home/solutions/spagobi>.
14. Apache Jackrabbit. [Online] Acedido em 16 de Dez. de 2008. <http://jackrabbit.apache.org/>.
15. Apache Lucene Overview. [Online] Acedido em 16 de Dez. de 2008. <http://lucene.apache.org/java/docs/>.
16. JasperReports home. [Online] Acedido em 17 de Dez. de 2008. [http://jasperforge.org/plugins/project/project\\_home.php?group\\_id=102](http://jasperforge.org/plugins/project/project_home.php?group_id=102).
17. Weka - Data Mining Software in Java. [Online] Acedido em 18 de Dez. de 2008. <http://www.cs.waikato.ac.nz/ml/weka/>.
18. Enhydra Octopus. [Online] Acedido em 17 de Dez. de 2008. <http://www.enhydra.org/tech/octopus/>.
19. Quartz Overview. [Online] Acedido em 18 de Dez. de 2008. <http://www.opensymphony.com/quartz/>.

20. eXo Portal. [Online] Acedido em 18 de Dez. de 2008.  
<http://www.exoplatform.com/portal/public/en/product/portal/overview>.
21. Pentaho BI Suite. [Online] Acedido em 10 de Dez. de 2008.  
<http://www.pentaho.com/>.
22. Pentaho Analysis Services:Mondrian Project. [Online] Acedido em 17 de Dez. de 2008. <http://mondrian.pentaho.org/>.
23. Pentaho Reporting. [Online] Acedido em 19 de Dez. de 2008.  
<http://reporting.pentaho.org/>.
24. Kettle - Pentaho Data Integration. [Online] Acedido em 19 de Dez. de 2008.  
<http://kettle.pentaho.org/>.