

UNIVERSIDADE FEDERAL DE JUIZ DE FORA
INSTITUTO DE CIÊNCIAS EXATAS
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

SISTEMA OLAP PARA MANIPULAÇÃO DOS DADOS DE ACESSO DOS USUÁRIOS DO MOODLE

Túlio César Faria Pinto

JUIZ DE FORA
AGOSTO, 2013

SISTEMA OLAP PARA MANIPULAÇÃO DOS DADOS DE ACESSO DOS USUÁRIOS DO MOODLE

TÚLIO CÉSAR FARIA PINTO

Universidade Federal de Juiz de Fora
Instituto de Ciências Exatas
Departamento de Ciência da Computação
Bacharelado em Ciência da Computação

Orientador: Eduardo Barrére

JUIZ DE FORA
AGOSTO, 2013

SISTEMA OLAP PARA MANIPULAÇÃO DOS DADOS DE ACESSO DOS USUÁRIOS DO MOODLE

Túlio César Faria Pinto

MONOGRAFIA SUBMETIDA AO CORPO DOCENTE DO INSTITUTO DE CIÊNCIAS
EXATAS DA UNIVERSIDADE FEDERAL DE JUIZ DE FORA, COMO PARTE INTE-
GRANTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE
BACHAREL EM CIÊNCIA DA COMPUTAÇÃO.

Aprovada por:

Eduardo Barrére
Prof.Dr.

Liamara Scortegagna
Prof^a. Dr^a.

Crystiam Kelle Pereira e Silva
Analista de TI - CEAD - UFJF

JUIZ DE FORA
DE AGOSTO, 2013

A minha família e amigos.

Aos pais, pelo apoio e incentivo.

A Carol por todo suporte e carinho.

Ao Prof. Barrére por toda paciência e dedicação na orientação desse trabalho.

Resumo

Com a evolução dos Learning Management System (LMS), ou Sistemas de Gerenciamento de Ensino, muitas informações são guardadas nos bancos de dados de plataformas de suporte ao ensino. Um dos LMS mais conhecidos é o Moodle que hoje é utilizado em diversas Universidades. Utilizando técnicas de Data Warehouse e de visualização dinâmica OLAP foi desenvolvida uma arquitetura para aproveitar melhor toda a informação contida nos bancos de dados do Moodle, gerando relatórios dinâmicos e criando oportunidades para a aferição de padrões de dados antes não conhecidos.

Palavras-chave: Moodle, OLAP, Data Warehouse.

Abstract

With the development of Learning Management System (LMS) or Learning Management Systems, a lot of information are stored in databases platforms support teaching. One of the largest knowned LMS it's Moodle, that today is used in various universities. Using techniques of Data Warehouse and OLAP dynamic visualization, was developed an architecture to better leverage all the information contained in these databases, generating dynamic reports and creating opportunities for benchmarking data standards not previously known.

Keywords: Moodle, OLAP, Data Warehouse.

Agradecimentos

A todos os meus familiares e amigos pelo apoio durante toda essa jornada.

Em especial ao Professor Eduardo Barrère pela orientação, amizade e principalmente, pela grande paciência, sem a qual este trabalho não se realizaria.

A todos os professores do Departamento de Ciência da Computação pelas experiências compartilhadas e ensinamentos passados.

Aos amigos que fiz durante o curso, que tiveram uma contribuição fundamental para a conclusão dessa etapa.

“Se fosse fácil achar o caminho das pedras, tantas pedras no caminho não seria ruim”.

*Humberto Gessinger (Outras
Frequências)*

Sumário

1	Introdução	10
2	Sistemas de Apoio à Decisão	12
2.1	DataWarehouse	13
2.2	Data Mart	15
2.3	Ferramentas de Tomada de Decisão	16
2.3.1	OLAP - On-line Analytical Processing	17
3	Motivação e Objetivos	21
4	Arquitetura Proposta	24
4.1	Estrutura de Dados Moodle	25
4.2	Modelagem da Arquitetura	27
4.2.1	Seleção, Tratamento e Extração de Dados	29
4.2.2	Medidas Adotadas	29
4.2.3	Interface OLAP	31
5	Desenvolvimento Plug-in de referência	33
5.1	Requisitos Necessários	33
5.2	Criando um novo bloco no Moodle	34
5.3	Implementação Ferramenta OLAP	36
5.3.1	Layout	37
5.3.2	Carregamento e acesso aos dados	38
5.3.3	Criação das Medidas	39
5.3.4	Criação dos Filtros	40
5.3.5	Geração das Tabelas de Resultados	41
5.3.6	Exportação dos dados	43
6	Análise de Resultados	44
6.1	Trabalhos Futuros	49
7	Conclusão	50
	Referências Bibliográficas	52

Lista de Figuras

2.1	Processo de acesso e geração de relatórios por ferramenta OLAP	18
3.1	Acesso aos dados sem nenhuma ferramenta OLAP vinculada	22
3.2	Acesso aos dados com uma ferramenta OLAP vinculada	23
4.1	Etapas da Arquitetura Proposta	24
4.2	Tabela mdllog	26
5.1	Layout da Ferramenta	37
5.2	Visualização dos Filtros de Dados	41
5.3	Geração da Tabela de Resultados	42
5.4	Botão para a exportação dos Dados	43
6.1	Análise: Módulo Acessado X Ano	47
6.2	Análise: Módulo Acessado X Ação Efetuada	48

Lista de Tabelas

2.1	Comparativo OLPT e OLAP)	20
-----	------------------------------------	----

Lista de Abreviações

BD	Banco de Dados
DM	Data Mart
DOLAP	Desktop On-Line Analytical Processing
DW	Data Warehouse
ETL	Extract, Transform and Load
FAD	Ferramentas de Apoio à Decisão
GPL	General Public License
JSON	JavaScript Object Notation
LMS	Learning Management System
MOLAP	Multidimensional On-Line Analytical Processing
MOODLE	Modular Object-Oriented Dynamic Learning Environment
OLAP	On-line Analytical Processing
OLTP	Online Transaction Processing
PHP	Personal Home Page
ROLAP	Relational On-Line Analytical Processing
SAD	Sistema de Apoio à Decisão
SGBD	Sistema de Gerenciamento de Banco de Dados

1 Introdução

Em ambientes acadêmicos modernos existem vários sistemas de gestão de aprendizagem. Segundo Perkins (1996), estes são sistemas computacionais que estão disponíveis na internet e se destinam a oferecer suporte às atividades mediadas pelas tecnologias de informação e comunicação.

Almeida (2003) afirma que:

Esses sistemas permitem integrar múltiplas mídias, linguagens e recursos, apresentar informações de maneira organizada, desenvolver interações entre pessoas e objetos de conhecimento, elaborar e socializar produções tendo em vista atingir determinados objetivos.

Atualmente existe uma ferramenta muito utilizada em ambientes acadêmicos denominada *Learning Management System* (LMS), que é um sistema de gerenciamento que permite uma significativa interação entre os educadores e alunos para cursos não presenciais, semi-presenciais ou ainda cursos presenciais que utilizam ferramentas tecnológicas para apoio às suas atividades. Um dos LMS mais utilizados é o Moodle¹ (*Modular Object-Oriented Dynamic Learning Environment*), sistema de código aberto para Gerenciamento de Cursos, disponibilizado livremente na forma de software livre - sob a licença de software livre *GNU Public License*, e pode ser instalado em diversas plataformas.

O Moodle é desenvolvido colaborativamente por uma comunidade virtual - reúne programadores e desenvolvedores de software livre e está em constante evolução, visto que sempre conta com a colaboração direta de seus usuários. Esse sistema permite que os usuários através de suas várias funcionalidades armazenem uma infinidade de dados relacionados ao histórico de atividades acadêmicas dos alunos, como conjunto de atividades, notas, tarefas realizadas, chats, fóruns, wikis, dados de frequência de acesso, entre outros. Neste ambiente específico, tais dados são de grande valia, pois através deles pode-se acompanhar de forma sistêmica a trajetória de um aluno em determinada disciplina/atividade, bem como gerenciar o rendimento individual do aluno ou mesmo de toda uma turma.

Em um grande ambiente educacional todas essas informações armazenadas permitem diversas análises que podem gerar melhorias nos processos. Devido à importância do

¹www.moodle.org

tema em questão, este trabalho tem como objetivo criar um ambiente computacional que facilite essas análises de forma rápida e intuitiva, além de auxiliar na gestão do sistema. Para cumprir com essa finalidade, foi adotado como base o conceito de armazém de dados (DataWarehouse), modelo que armazena dados históricos de determinado segmento para serem usados em processos de tomada de decisão. Esse modelo foi desenvolvido especificamente para a análise dos dados gerados pelas diversas áreas de atuação, com o intuito de auxiliar na geração de relatórios pontuais e dinâmicos com embasamento histórico que facilitem a tomada de decisão.

Na área comercial o Datawarehouse (DW) é muito bem explorado dando uma grande agilidade aos gestores e consequentemente trazendo um aumento considerável na competitividade das empresas. Os relatórios gerados trazem tendências, às vezes não explícitas, que permitem uma tomada de decisão consciente.

O intuito, portanto, é desenvolver uma arquitetura que contemple a criação de um DW, resultando em uma ferramenta que permita o acesso a esses dados de forma rápida e também segura, além da manipulação intuitiva desses dados, e ofereça um grande suporte na gestão educacional. Além da configuração dessa coleção de dados, este trabalho também tem como objetivo criar um novo plug-in para o Moodle que gere uma interface OLAP (*On-line Analytical Processing*) com o Data Mart (DM). Como a idéia central é obter relatórios e dados que auxiliem na gestão dos cursos, do desempenho dos alunos, etc, de forma dinâmica, a interface OLAP visa atender as necessidades tanto de administradores experientes, quanto de tutores e professores sem experiência prévia na área de Tecnologia da Informação. Assim, é proposta uma arquitetura com a finalidade de criar um DM focado nos dados de acesso dos usuários do Moodle e desenvolver uma interface que consiga extrair os dados de forma intuitiva e rápida. Na seção 2 são apresentados os conceitos de Sistemas de apoio à decisão que vão nortear o trabalho. Na seção 3 são apresentados os objetivos centrais do trabalho. Na seção 4 a arquitetura proposta no trabalho. Na seção 5 todo o desenvolvimento da ferramenta proposta.

2 Sistemas de Apoio à Decisão

Através da evolução dos meios de armazenamento de dados e de comunicação, as informações, tanto pessoais como profissionais, que antes eram registradas em papel, foram transferidas para os meios digitais. Com essa realidade ficando cada vez mais presente, apenas a análise óbvia dos dados não eram suficientes para abordar todos os pontos e perfis que poderiam ser verificados e analisados.

Com tantas informações sobre perfis econômicos, políticos, científicos e sociais, apenas a análise gerencial dos dados não extraía todo o potencial presente nos volumes de informações. Para que essa deficiência fosse sanada, técnicas de modelagem e distribuição foram adotadas para otimizar o armazenamento e acesso desses dados, que ao passar do tempo tornaram-se mais complexos e volumosos. Com o intuito de gerenciar esses dados surgiram os sistemas SAD (Sistema de Apoio à Decisão). (BORTOLI, 2010)

Os SAD são sistemas computacionais que visam sistematizar e apoiar os processos decisórios empresariais, sendo formados pelos componentes básicos: interface com o usuário, banco de dados e banco de modelos. O objetivo desses componentes é fornecer apoio em decisões semi-estruturadas e desestruturadas, que estão sob o controle do tomador de decisões e disponibiliza um conjunto de ferramentas para estruturar e aumentar a efetividade das decisões (JUNIOR; FERNANDES; PEREIRA, 2006).

Segundo Júnior (2012), um SAD possui algumas características básicas, tais como as que podem ser verificadas a seguir:

- Manipulação de Grandes Volumes de Dados: Bancos de dados avançados permitem que o administrador busque informações em outros bancos de dados;
- Obtenção e Integração de Dados de fontes diferentes: Possui a capacidade de acessar e integrar dados de fontes externas;
- Flexibilidade de relatórios gerenciais: Um SAD possui relatórios de diferentes formatos (textuais, gráficos) a critério de quem opera o sistema;

- Suporte para diversos níveis na tomada de decisão: os SAD's auxiliam gerentes de todos os níveis de uma organização (operacional tático e estratégico).

Em contrapartida, os aplicativos gerenciais originam respostas e relatórios gerenciais padrões, segundo Falsarella e Chaves (2008), e as Ferramentas de Apoio à Decisão (FAD) auxiliam na simulação das situações, como na representação gráfica das informações e na descoberta do conhecimento implícito.

O Banco de Dados (BD) vinculado a uma FAD é formado por dados de diferentes fontes que contém informações relativas ao passado de ações de uma determinada área e que auxiliam na tomada de decisões. Esse banco, que contém informações externas e internas, segue os conceitos de um DataWarehouse , conforme será visto à seguir.

2.1 DataWarehouse

Os bancos de dados transacionais ou operacionais corporativos armazenam as informações diárias de uma empresa, sendo atualizados constantemente e tendo uma quantidade grande de transações. Nesse tipo de banco não ocorrem redundâncias de dados e as informações históricas da empresa não ficam armazenadas por muito tempo. Devido a esse fator, consultas no histórico dos dados da empresa para o auxílio na tomada de decisões não é satisfatório, uma vez que existe um grande volume de operações e de dados, tornando-as lentas e inviáveis.

A solução encontrada para esse problema foi a criação de um banco de dados específico para a análise dos dados históricos da empresa. Surge o conceito de DW, que pode ser entendido como um armazém de dados, sendo um banco orientado para uma estrutura gerencial. Segundo Perkins (2006) , esse armazém provê as necessidades de informações de uma empresa em todos os níveis organizacionais. Não só informações complexas, mas com mecanismos simples para serem obtidas de forma rápida e precisa.

De acordo com Taurion (1998), ao reunir informações dispersas nos diversos bancos de dados operacionais da empresa, que podem estar em plataformas distintas, o DW permite que sejam feitas consultas e análises bastante eficazes, transformando dados esparsos em informações antes inacessíveis ou sub-aproveitadas, que podem ser convertidas

em estratégias para os negócios.

O DW fornece acesso a dados corporativos ou organizacionais que são consistentes e podem ser separados e combinados usando qualquer medição possível. Além disso, é possível ter acesso a um conjunto de ferramentas para consultar, analisar e apresentar informações. É um local onde se publica dados confiáveis, sendo a qualidade desses um impulso à reengenharia de negócios (KIMBALL, 2002).

A definição universalmente utilizada para DW foi citada no livro *Building the Data Warehouse* escrito por Bill Inmon (2005):

Uma coleção de dados orientados por assuntos, integrados, variáveis com o tempo e não voláteis, para dar suporte ao processo de tomada de decisão; trata-se de um processo em andamento que aglutina dados de fontes heterogêneas, incluindo dados históricos.

Como se vê nas definições anteriores à criação de um armazém de dados é a estruturação destes em uma coleção de dados de áreas específicas de uma empresa. É um volume de dados selecionado, em um repositório específico, modelado com hierarquias e com tratamentos específicos para os acessos dinâmicos e para tomadas de decisão.

Ainda de acordo com Inmon (2005), as características básicas de um DW são:

- **Orientação ao Assunto:** um DW é desenvolvido especificamente para uma determinada área, podendo ser desenvolvido um detalhamento específico dentro desta mesma área. Podemos citar como exemplo, o caso de uma empresa que precisa que seu gerenciador saiba mais sobre a área de vendas, neste caso um DW será desenvolvido com o foco em vendas;
- **Dados Integrados:** Um DW necessita que dados de diferentes fontes sejam colocados em um formato consistente. Estes necessitam ser transformados e tratados para que fiquem em um padrão único;
- **Não Volátil:** Significa que uma vez que os dados estão dentro do DW eles não sofrem alterações, apenas inserções;
- **Variável com o Tempo:** Contrariamente a um sistema OLTP (*Online Transaction Processing*), onde o desempenho requerido demanda que os dados históricos sejam movidos para um arquivo, o DW possui o foco em mudanças de acordo com o tempo.

O processo de criação de um Data Warehouse é composto por algumas etapas e a primeira delas é a *ETL* (*Extract Transform Load*), onde ocorre a extração, transformação e carga dos dados para o Data Warehouse. De acordo com Costa (2007), as etapas são definidas em:

- Extração: esta fase consiste em definir as fontes de dados e movê-las para o DW, de forma a manter os dados no mesmo local. Entretanto, serão movidos apenas os dados que estão de acordo com os objetivos do DW;
- Transformação: este processo consiste em limpar e alterar os dados obtidos da fonte, de forma a padronizar os dados. A limpeza dos dados está ligada ao 'lixo' e inconsistências existentes entre e nas bases de dados. Já a alteração servirá para padronizar um dado que tem a mesma representação semântica, só que escrito de forma diferente.
- Carga: este processo consiste em integrar os dados, mantendo sua integridade. Este processo tem enorme complexidade, pois muitos fatores são considerados.

Um Data Warehouse possui várias etapas, mas é na ETL que é gasto mais tempo e mais recurso financeiro. Como é um processo complexo, o DW não precisa ser construído de uma vez, mas área por área, ou seja, através de um Data Mart (DM), cujo conceito pode ser acompanhado no próximo capítulo.

2.2 Data Mart

Para Inmon (2005), um DM geralmente é escrito como um subconjunto dos dados contidos em um DW e extraído para um ambiente separado. Geralmente são construídos e controlados por um departamento dentro de uma organização, como vendas, finanças e marketing, e permite extrair os dados de poucas fontes, que podem ser sistemas de operações internos, um DW central, ou externos.

A principal diferença entre um DW e um DM é o escopo. Um DW engloba coleções de dados vindos de várias fontes diferentes dentro de um ambiente corporativo, sendo utilizado apenas um “ramo” dessa coleção. A variabilidade do DW em relação ao

tempo é grande para que se faça uma análise confiável, enquanto essa variação em um DM é menor, por se tratar de um foco menor de análise. Existe uma divergência conceitual com relação ao DM entre dois grandes nomes em Data warehouse

De acordo com Inmon (2005), o DM seria uma subdivisão de um DW, mas especializado em uma área específica e extraindo dados diretamente de um data warehouse, enquanto para Kimball (1998), a integração de vários DM gera um DW. A criação de um DM para cada área específica de análise e a integração destes DMs gera um DW, sendo o fluxo do DM para o DW.

2.3 Ferramentas de Tomada de Decisão

Existem várias ferramentas de tomadas de decisão. Nesse trabalho vamos focar no conjunto de tecnologias OLAP, pois este tipo de tecnologia faz a abstração dos dados com o cubo de dados e permite a visualização dos mesmos em diversas dimensões sendo o resultado gerado dinamicamente.

Tan, Steinbach e Kumar(2009) afirmam que a mineração de dados é o processo de descoberta automática de informação úteis e recentes que poderiam não ser considerados e permanecerem ignorados. O processo para a descoberta desses padrões não esperados passa pelas seguintes etapas:

- Extração: definição de quais dados serão analisados;
- Seleção de dados: identificação de potenciais tabelas para que se obtenha resultados consistentes e úteis, sendo a escolha dessas tabelas de suma importância para que a análise traga padrões ou resultados extremamente produtivos. A escolha dos atributos certos a serem avaliados implicam em uma diminuição considerável na complexidade dos dados, do volume de dados e de ruídos no final da análise.
- Transformação: nessa fase é garantida a qualidade dos dados a serem processados bem como a categorização dos mesmos.
- Limpeza dos dados (*Data Cleaning*): tem como intuito eliminar as inconsistências de gravação de dados, caracteres especiais, tipos de dados, ou seja, toda a limpeza

das tabelas a serem analisadas é feita nessa etapa. A etapa de transformação dos dados é a fase onde os dados selecionados e limpos são transformados nos padrões para as análises a serem feitas de acordo com o tipo e o algoritmo utilizado.

2.3.1 OLAP - On-line Analytical Processing

Como visto OLAP (*On-line Analytical Processing*) pertence a classe dos SAD e segundo Chaudhuri e Dayal (1998) é formado por um conjunto de tecnologias para a visualização de sistemas multidimensionais que ajudam os analistas e executivos a sintetizarem informações sobre a empresa, através de comparações, visões personalizadas, análise histórica e projeção de dados.

2.3.1.1 Principais Conceitos

Anzanello (2005) afirma que a visualização é realizada em dados agregados, e não em dados operacionais, isto porque a aplicação OLAP tem por finalidade apoiar os usuários finais a tomar decisões estratégicas.

Segundo Forsman (1997, citado por Bispo e Cazarini, 1998) um sistema OLAP aumenta a produtividade dos gerentes da organização inteira, a sua flexibilidade significa que os gerentes podem se tornarem mais auto-suficientes. Permitem aos gerentes modelar problemas que seriam impossíveis usando sistemas menos flexíveis, com um tempo de resposta longo, ou incompatíveis com a estrutura analítica dos negócios.

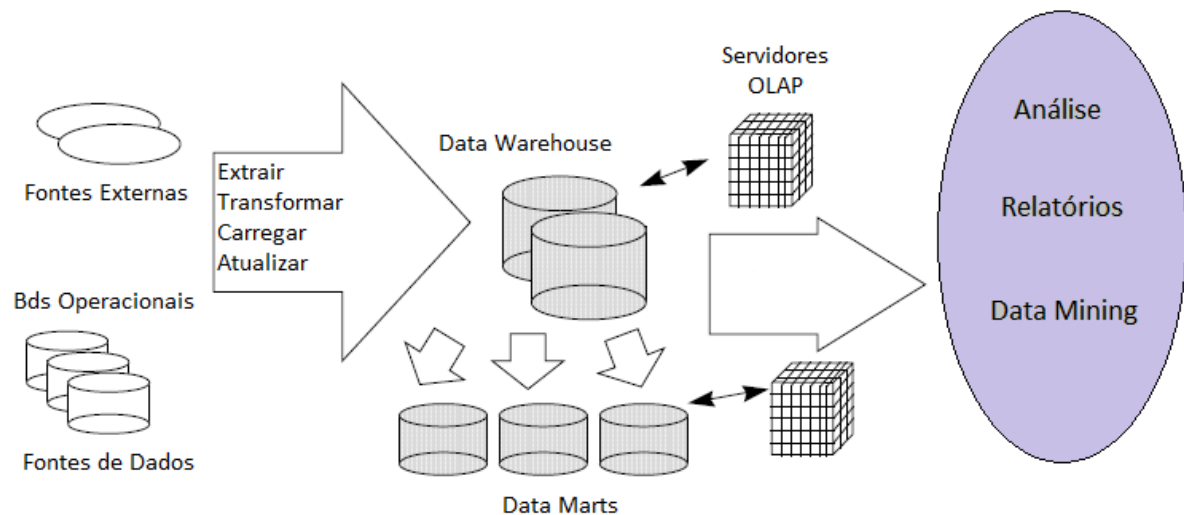


Figura 2.1: Processo de acesso e geração de relatórios por ferramenta OLAP

Na figura acima se pode ver o fluxo completo desde as primeiras etapas da criação de um DW - conforme os conceitos de Kimball e Inmon, com as etapas de ELT (Seleção, extração e transformação) dos dados e depois o acesso de uma ferramenta OLAP e a geração de relatórios e análises.

Em 1993, E.F. Codd escreveu um artigo onde definia as 12 normas básicas de definição de OLAP e são elas: (Codd; Codd; Salley, 1993)

1. Visão Multidimensional: Um sistema OLAP deverá por natureza possuir uma visão multidimensional dos dados.
2. Transparência: O sistema deve ser transparente ao usuário. O sistema pode ser integrado à algum outro sistema ou standalone, mas para o usuário não pode ser exposto de nenhuma maneira ao código, ou informações de baixo nível de código.
3. Acessibilidade: A ferramenta deve ser capaz de aplicar sua estrutura lógica de acesso em fontes heterogêneas de dados e realizar qualquer conversão necessária para apresentar uma visão coerente para o usuário.
4. Performance consistente de geração de relatórios: A performance da ferramenta não deve sofrer significativamente quanto o aumento de dimensões selecionadas.

5. Arquitetura Cliente/Servidor: O servidor da ferramenta deve ser suficientemente inteligente para que vários usuários possam se conectar sem o mínimo esforço e o servidor deve ser capaz de mapear e consolidar os dados entre banco distintos.
6. Dimensionalidade Genérica: Cada dimensão de dados deve ser equivalente em sua própria estrutura e em sua capacidade operacional.
7. Manipulação dinâmica de matriz esparsa: O servidor deve possuir estrutura física para otimizar a manipulação dinâmica de matriz esparsa.
8. Suporte Multi-usuário: A ferramenta deve fornecer recuperação simultânea e acesso de atualização, integridade e segurança.
9. Irrestritas operações interdimensionais: Deve permitir qualquer número de dimensões e não restringir qualquer relacionamento.
10. Manipulação dos Dados intuitiva: As operações devem ser feitas diretamente no modelo analítico, sem a necessidade de realizar vários caminhos.
11. Relatórios Flexíveis: Deve-se gerar relatórios da maneira que o usuário quiser visualizar os dados.
12. Ilimitados níveis de agregações e dimensões: O número de dimensões de dados suportados devem, para todos os efeitos, ser ilimitado.

2.3.1.2 Métodos de Armazenamento

Para Sant'anna(2008), a principal característica de um sistema OLAP é permitir a visão dos armazenados multidimensionais através de cubos.

Os dados a serem visualizados no sistema OLAP são normalmente extraídos de um DW, mais precisamente de uma especialização de área, ou seja, um DM. Normalmente essas ferramentas utilizam um banco multidimensional, pois assim, a exibição dos dados ao usuário final se torna mais intuitiva e simples. Porém, essas ferramentas podem possuir variações de tipo de acordo com o método de armazenamento dos dados.

Em contrapartida, a ferramenta *DOLAP* (*Desktop On-Line Analytical Processing*) é uma variação que existe para fornecer portabilidade dos dados. A vantagem que

oferece esta arquitetura é a redução do tráfego na rede.

As ferramentas que buscam seus dados em um banco de dados relacional utilizando o SQL padrão, *ROLAP (Relational On-Line Analytical Processing)*, suportam um grande volume de dados, mas têm um tempo de resposta muito grande, além de ocupar um grande espaço em disco, conforme Pires (2004). Já as ferramentas *MOLAP (Multidimensional On-Line Analytical Processing)* utilizam bancos de dados multidimensionais e a estrutura de cubos, possuindo um bom tempo de resposta nas consultas, mas apresentando limitações na escalabilidade. Com o aumento de dimensões a carga fica muito alta. São recomendadas em casos em que já se possui as dimensões e o escopo dos dados bem definidos.

A *HOLAP (Hybrid On-Line Analytical Processing)* é uma combinação entre *ROLAP* e *MOLAP*, que consegue um melhor desempenho por integrar bancos de dados relacionais e multidimensionais. Pode-se usar um banco de dados multidimensional para fazer um "cachê" dos dados com um nível maior de agregação, e usar um banco relacional para fazer um acesso dinâmico aos dados detalhados. A seguir uma tabela comparativa entre OLPT e OLAP segundo ANZANELLO(2005).

Tabela 2.1: Comparativo OLPT e OLAP)

CARACTERÍSTICAS	OLPT	OLAP
Operação Típica	Atualização	Análise
Telas	Imutável	Definida pelo Usuário
Nível de Dados	Atomizado	Altamente Sumarizado
Idade dos Dados	Presente	Histórico, Atual e Projetado
Recuperação	Poucos Registros	Muitos Registros
Orientação	Registro	Arrays
Modelagem	Processo	Assunto

3 Motivação e Objetivos

Com a evolução do processo educacional e com o aumento da tecnologia nos meios acadêmicos, a administração dos processos de ensino seguiu o fluxo natural das modernizações e se tornou hoje uma grande ferramenta para o ciclo de aprendizado de universitários e de outros níveis. Seguindo esse contexto, ferramentas que vieram para auxiliar nesse processo foram desenvolvidas com o intuito de integrar, monitorar e acompanhar de forma objetiva o crescimento do conhecimento das partes envolvidas.

Além disso, essas ferramentas possuem um papel muito importante na integração de atividades, permuta de materiais, pesquisas e gerenciamento de cursos à distância. Oferecem a oportunidade para que os cursos fora da extensão física das universidades fossem ministrados e orientados com qualidades. Essa ferramenta pode ser também um apoio de extrema importância no controle das atividades das classes presenciais.

Existem diversos softwares que cumprem muito bem esse papel. Para desenvolver este trabalho será utilizado a ferramenta de código aberto Moodle, por se tratar da ferramenta em uso na Universidade Federal de Juiz de Fora e como mencionado anteriormente é uma das ferramentas mais utilizada para gerenciamento de ensino acadêmico atualmente no mercado e também por isso é foco deste trabalho

A constante interação entre os usuários do sistema (educadores e alunos), gera um grande volume de dados sobre todo o desenvolvimento dos cursos e de aprendizado de um aluno. Tem-se então informações valiosas sobre todo o trajeto de um aluno dentro de um determinado curso, suas interações com outros usuários, tutores, professores, etc. Suas preferências, rotinas de acesso, frequência, localidades, modos de acesso, rede de contatos, nível de interação e demais informações que geradas ao longo do tempo se tornam uma fonte relevante de dados e que se bem trabalhadas podem transformar informações subaproveitadas em fontes interessantes de pesquisa.

Visualizando esse cenário, têm-se uma boa oportunidade de aferir padrões e de aproveitar essas informações, que a princípio seriam informações apenas relevantes para a parte de manutenção do sistema, colocando-as em contato com quem realmente interage

diariamente com os módulos de ensino, professores e tutores.

Assim, o objetivo desse trabalho é criar, através dos conceitos de Sistemas de Tomada à Decisão (SAD), uma arquitetura DW com uma ferramenta OLAP para a visualização dos dados. A intenção principal é fornecer ao administrador do sistema e demais gestores um recurso intuitivo e dinâmico que auxilie na tomada de decisão utilizando os dados armazenados provenientes das bases de dados do Moodle.

Em um ambiente que não possua uma ferramenta OLAP, todas as requisições de relatório feitas precisam passar por uma pessoa com conhecimento técnico na área de TI para saber extrair e interpretar as informações para depois repassá-las aos solicitantes. Conforme a figura a seguir:

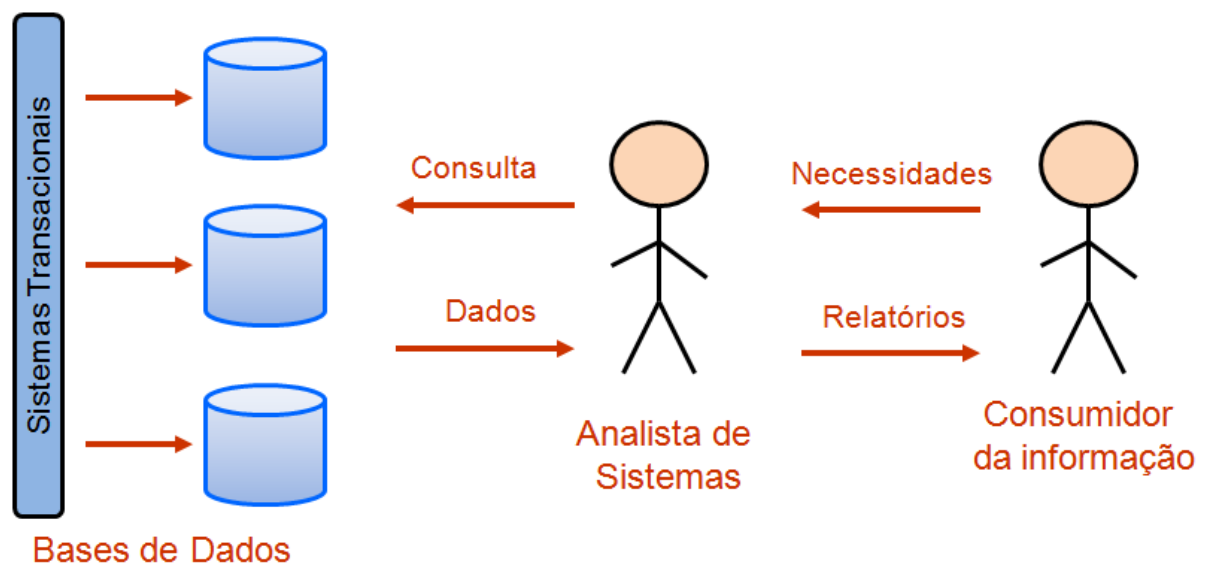


Figura 3.1: Acesso aos dados sem nenhuma ferramenta OLAP vinculada

Utilizando uma ferramenta OLAP, usuários que não possuam o conhecimento técnico na área de Banco de Dados podem ter acesso às informações de forma intuitiva e dinâmica, vide figura 3.2. Um tutor mesmo sem muita experiência com informática poderia através dessa ferramenta acessar dados relevantes dinamicamente e que irão auxiliá-lo em suas demais funções. O acesso fácil e rápido a esses dados desonera e tira carga do setor de TI responsável, e oferece maior autonomia ao usuário do sistema.

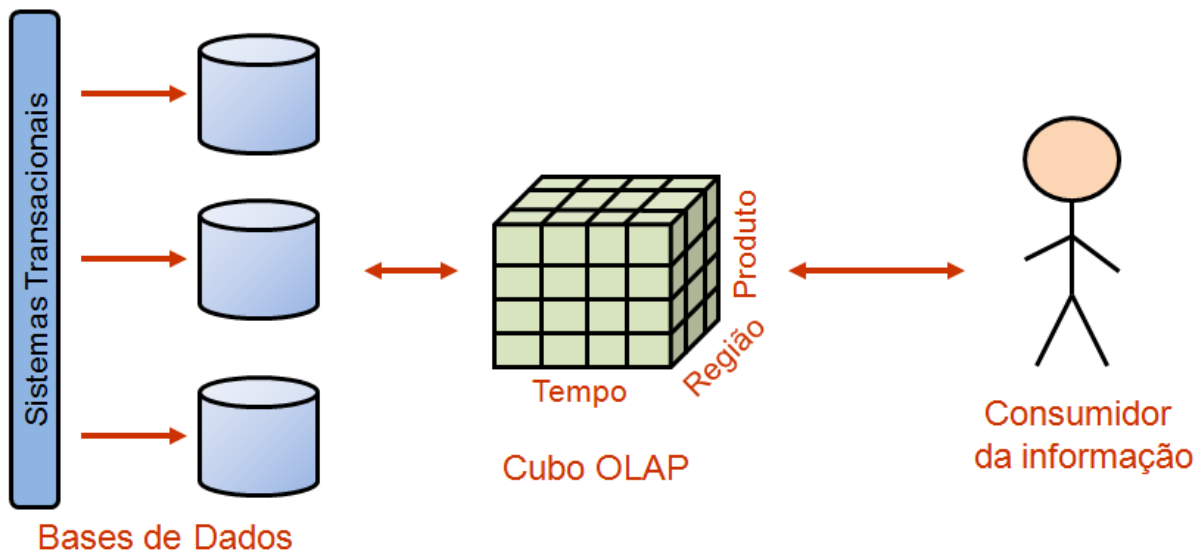


Figura 3.2: Acesso aos dados com uma ferramenta OLAP vinculada

Os dados de acesso foram recolhidos para ilustrar e demonstrar a criação e utilização dessa ferramenta, já que possui dados interessantes sobre o acesso e distribuição dos mesmos ao longo dos cursos e anos. Isso porque a tabela LOG que contém esses dados possui uma característica interessante que será demonstrada no capítulo 4.

Estabelecidos os objetivos e motivações, será apresentado adiante a arquitetura proposta para a criação dessa ferramenta.

4 Arquitetura Proposta

A arquitetura proposta nesse trabalho possui 3 partes:

- Seleção e extração dos Dados - ETL
- Criação de um DM referente aos dados de acesso do Moodle
- Interface OLAP para acesso aos dados selecionados através da implementação de um plugin no Moodle



Figura 4.1: Etapas da Arquitetura Proposta

Todas as etapas são integradas e em cadeia. Na primeira etapa da arquitetura é feita a identificação, seleção, tratamento e extração dos dados que serão objetos de estudo do trabalho. Foi feito um levantamento de todas as tabelas relevantes relacionadas aos dados de acesso, as integrações entre elas, as medidas relevantes e o volume de dados presente. A partir desse momento os dados foram tratados com o objetivo de extrair qualquer “ruído” e não conformidades dos dados correspondentes. Ao final dessa etapa foi feita a extração, o tratamento e o envio desses dados para o novo banco não transacional que foi criado, nosso DW.

Na segunda parte com os dados selecionados e tratados, foi criado e modelado o banco para a criação do DW (tabela de fatos e dimensões). Nessa etapa foram definidas as medidas e dimensões que foram utilizadas.

Na última parte foi feito o desenvolvimento da interface de interação OLAP. Um novo módulo dentro do Moodle foi implementado - uma ferramenta OLAP para a exibição

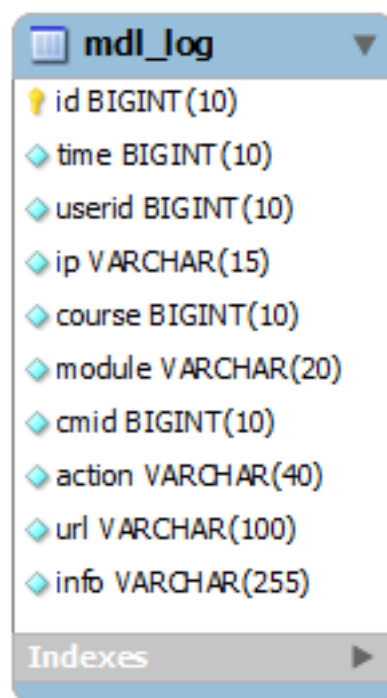
dos dados de forma interativa. Primeiramente foi tratado a criação da estrutura inicial de um novo módulo interno à plataforma Moodle (procedimentos, métodos, framework e tabelas) depois foi apresentado o desenvolvimento da ferramenta interna em si.

Para que o ambiente planejado tenha sucesso, todas as informações que compõem os sistemas devem ser bem conhecidas. Inicialmente é apresentada a estrutura de dados do Moodle.

4.1 Estrutura de Dados Moodle

O Moodle trabalha com uma estrutura de dados bem definida no que tange à nomenclatura e associação entre as tabelas. As tabelas que guardam as informações geradas pela interação com o usuário possuem sempre a seguinte composição: mdl_(nome da área relacionada). Existem outras tabelas que são específicas da manutenção do sistema. A interação entre elas é feita por chaves estrangeiras que possuem ligação clara com os participantes dessa junção.

Como o foco será no estudo dos dados de acesso do Moodle, a primeira tabela e a mais importante para o desenvolvimento do trabalho é a tabela “mdl_log”, cuja estrutura pode ser vista na figura 4.2. O Moodle não trabalha um arquivo de Log como normalmente é utilizado em vários sistemas, essa tabela guarda todas as informações relacionadas ao acesso do usuário a qualquer seção do sistema, desde o momento do login até um post em um fórum ou mesmo um chat.



mdl_log	
id	BIGINT(10)
time	BIGINT(10)
userid	BIGINT(10)
ip	VARCHAR(15)
course	BIGINT(10)
module	VARCHAR(20)
cmid	BIGINT(10)
action	VARCHAR(40)
url	VARCHAR(100)
info	VARCHAR(255)

Indexes

Figura 4.2: Tabela mdllog

Essa tabela possui a chave primária *id* que corresponde a um número incrementado sequencialmente que representa cada ação de algum usuário no sistema. O campo *time*, representa o momento exato da ação, é do tipo BIGINT porque ele é um `TIMESTAMP`¹ do UNIX. O `TIMESTAMP` do UNIX é um valor que representa o total de segundos decorridos a partir de primeiro de Janeiro de 1970. Então o número gerado é o número total de segundos entre essa data de início e a data referida. Dessa maneira a ordenação por data é muito menos custosa. UNIXTIMESTAMP (2011).

O campo *userid* é uma chave estrangeira para a tabela *mdl_user* e corresponde ao usuário que efetuou a ação. O campo *ip* representa o ip utilizado no momento da ação pelo usuário. O campo *course* é a chave estrangeira para a tabela *mdl_course* e especifica qual curso o usuário estava acessando no momento da ação. O campo *module* representa com qual tipo de módulo o usuário estava acessando, ou com qual módulo a ação se relaciona no momento da ação, ou seja, ele guarda o valor “user” se o usuário estiver acessando dados relacionados ao cadastro do usuário, da mesma forma que o valor será “forum” quando a ação for relacionada a algum post ou outro tipo de ação em fóruns.

¹www.unixtimestamp.com

O campo *action* guarda qual ação foi realizada, se foi a visualização de um post, login, logout, etc. O campo *url* é um campo auxiliar na montagem dos resultados dos relatórios nativos do Moodle, ele agrega os dados relacionados a ação e já exporta a url de consulta. Esse campo tem utilidade apenas para a própria plataforma Moodle. O último campo *info* faz a ligação entre o campo *action* e o recurso utilizado para tal ação. Se a ação realizada foi visualizar um post o campo *action* guarda o valor “view” e o campo *info* guarda a chave estrangeira para a tabela *mdl_forumposts* e assim se relaciona com as demais ações. Existem ações que não possuem vínculo direto com chave estrangeira nesses casos o campo guarda a própria informação.Ex.:wiki.

Além da tabela *mdl_log* existem tabelas que também são importantes nessa estrutura. Essas tabelas guardam informações relativas às dimensões presentes na tabela *mdl_log*. Como ainda será visto no próximo capítulo, na modelagem do banco DW essa tabela assim como as outras que representam dimensões,originarão tabelas mais compactas mas com o mesmo objetivo de representar dimensões.

A tabela *mdl_course* possui como campos mais importantes *id* que é a chave primária que faz vínculo com a chave estrangeira *course* presente na tabela *mdl_log*. Ainda pode-se destacar os campos *category* que é o campo que faz vínculo com a tabela *mdl_course_categories*, *fullname* e *shortname* que representam o nome do curso.

Outra tabela utilizada é a tabela “*mdl_course_categories*”, essa apenas faz join com a tabela “*mdl_course*” onde se extrai a categoria do curso em questão.

4.2 Modelagem da Arquitetura

Como descrito nos objetivos desse trabalho, a meta é criar um DM com os dados selecionados utilizando a abstração do Cubo de Dados para uma ferramenta OLAP interna ao sistema Moodle. A abstração de um Cubo de Dados como relatado na conceituação de OLAP é muito importante para que a visualização se torne dinâmica e rápida.

Nessa seção são descritos os passos para a criação do novo banco de dados e como a tabela de fatos é estruturada para o desenvolvimento da arquitetura.

Como o banco de um DW é um banco não transacional, primeiramente um novo banco precisa ser criado para que a velocidade das consultas e do retorno dos dados não

seja afetada. Além do ganho com o desempenho em um modelo de banco de dados não transacional, as tabelas não são normalizadas, justamente para que haja um ganho de desempenho no retorno dos dados. Esse novo banco pode ser criado na mesma instância do banco principal do Moodle ou pode ser criado em outra instância de outro servidor de banco. O controle de acesso aos dados do banco será feito através do próprio Moodle, como os dados serão acessados apenas pela interface do próprio Moodle as autorizações para a visualização do modulo serão as mesmas para o acesso ao banco.

Com o banco criado têm-se início a criação da estrutura da tabela de fatos. Como anteriormente citado, a tabela de log se assemelha a estrutura da tabela de fatos. Nessa “nova” tabela de fatos irão conter além das chaves estrangeiras que conectam essa tabela com todas as dimensões do armazém de dados, temos os dados de medidas que compõem a análise. A tabela é estruturada criando as chaves estrangeiras para as dimensões e os valores de análise. A tabela mdl_action foi criada, para auxiliar na composição Abaixo o script de criação da tabela de fatos.

```
1 CREATE TABLE IF NOT EXISTS MDL_LOG_DW
2 SELECT
3     TIME,
4     COURSE AS CURSO,
5     M.id AS MODULO_DE_ACESSO,
6     A.id AS ACAO,
7     USERID AS QT_USUARIOS,
8     IP AS QT_IP,
9     COUNT(DISTINCT L.ID) AS QT_ACESSO,
10 LEFT(FROM UNIXTIME (time),4) AS ANO,
11 SUBSTRING((FROM UNIXTIME (time)),6,2) AS MES,
12 SUBSTRING((FROM UNIXTIME (time)),9,2) AS DIA,
13 CC.id AS CURSO_CATEGORIA
14 FROM MDL_LOG L
15 LEFT JOIN MDL_COURSE C
16     ON C.id = L.course
17 LEFT JOIN MDL_COURSE_CATEGORIES CC
18     ON C.category = CC.id
19 LEFT JOIN MDL_ACTION A
20     ON L.action = A.action
21 LEFT JOIN MDL_ACESS_MODULE M
22     ON M.MODULE = L.module
23 GROUP BY TIME, COURSE,M.id,A.id,USERID,IP, CC.description
```

4.2.1 Seleção, Tratamento e Extração de Dados

Nessa seção terão início as etapas de *Data Staging Area* (ETL - *Extraction, Transformation and Load*), que é a etapa de extração, transformação e carga. Os dados deverão ser escolhidos com exatidão para que não haja ruído no estudo.

A tabela `mdl_log` será filtrada para que não possua informações não interessantes a análise dos dados, como entrada do usuário no sistema e a saída do usuário do sistema. Esses dados apesar de trazerem informações que dizem respeito ao horário de acesso de um determinado usuário, são informações básicas de qualquer acesso, ou seja, para que qualquer outra ação ocorra no sistema é condição obrigatória que o primeiro acesso seja feito, podendo afirmar que todas as ações presentes na tabela tem esse primeiro acesso, tornando essa informação irrelevante para o estudo. Da mesma forma a saída do sistema. Uma especificidade do Moodle é que por utilizar uma tabela de Log e por todas as ações estarem centralizadas numa mesma tabela isso torna a tabela muito confiável e íntegra. Os dados são alimentados pelo sistema e então essa etapa de “limpeza” (*Data Cleaning*) se torna não muito custosa.

Outra informação que para o estudo não será relevante são os registros de ação dos usuários administradores do sistema. Os usuários que administram o sistema cuidam de tarefas estritamente ligadas à manutenção do sistema, do banco de dados, inserção ou deleção de usuários, grupos de acesso, etc. Como esses usuários trabalham na manutenção do sistema a sua análise não agregará valor relevante ao foco do estudo, então os registros serão descartados.

4.2.2 Medidas Adotadas

A modelagem de um DW que se baseia em uma ferramenta OLAP é a abstração de um cubo de dados. Nessa abstração tem-se fatos e dimensões. Para se definir os fatos, ou seja, o que realmente será tratado no DW deve-se fazer a pergunta, o que realmente será medido?

Primeiramente são classificados os tipos de fatos existentes :aditivos, semi-aditivos e não-aditivos:

- Aditivos: São fatos em que se pode agregar e realizar operações de média e demais

operações com a granularidade alta. São fatos representados pelo tipo numérico. Normalmente são os fatos mais expressivos e que melhor representam a medida do negócio.

- Semi-aditivos: Normalmente são fatos que podem ser somados e agregados por algumas dimensões mas não todas e não possuem granularidade muito alta.
- não-aditivos: São fatos que não podem ser agregados de maneira nenhuma. Geralmente são medidas textuais.

Analizando o contexto do trabalho e a tabela em questão que será estudada, pode-se definir algumas medidas que são intuitivas. Como todas as ações são registradas, o fato aditivo mais relevante será a quantidade de ações. Como esse fato é aditivo pode-se agregar por todas as dimensões existentes na tabela, curso, módulo, tipo de ação e tipo de curso.

Outro fato que se pode aferir é que cada ação na tabela possui um usuário relacionado, então é interessante aferir quantos usuários efetuaram tal ação agregado pelas dimensões da tabela. Um dado interessante que a tabela possui é o ip. Através do ip no momento da consulta pode-se identificar de qual localidade a funcionalidade é acessada. A tabela do Moodle não possui a informação de identificação de localidade, mas pensando em trabalhos futuros se pode aferir quantos ips diferentes acessarão tais funcionalidades e se poderá criar uma nova medida relacionando os acessos à localidade identificada.

Essas medidas são respectivamente aditiva e as duas últimas semi-aditivas. Saber quantos usuários acessaram tais funcionalidades em tal dia, pode ser muito útil para diversas análises: Por exemplo, qual é o dia de pico de acesso no mês? Através de uma análise histórica desses dados. Como é feita a organização dos alunos em relação à atividades? Fóruns e demais práticas durante o curso. Como é distribuído o acesso entre os cursos?, etc. Enfim alguns questionamentos dão o embasamento necessário para que esse fato seja relevante ao trabalho.

O fato IP como mencionado, é um valor relevante para cursos multi-regionais onde se é necessário tirar conclusões à respeito da diversidade de acesso e rendimento dos participantes das diversas regiões.

Analisando a criação da tabela de fatos pode-se constatar que os fatos escolhidos e que sejam semi-aditivos são gerados dinamicamente, ou seja, como um fato semi-aditivo não se soma com todas as dimensões então ele não pode ser um campo calculado na tabela, ele precisa ser gerado dinamicamente.

As dimensões escolhidas são claras no contexto da tabela. Como se possui ações e seus usuários e cursos correspondentes, pode-se aceitar como dimensões os cursos, módulos, os tipos de ações e tipos de cursos. Os tipos de ações são ótimas medidas, pois possuem um alto grau de granularidade o que torna os relatórios gerados com um nível de detalhe bem expressivo.

Como em uma estrutura de tabela de fatos bem definida, as informações das dimensões estão em tabelas externas vinculadas a uma chave estrangeira. Para tal foi-se necessário a criação de duas tabelas como citado no capítulo anterior. A tabela `mdl_action` foi criada para contemplar todos os valores possíveis para os tipos de ação. Essa ação fez-se necessária, pois como os valores desse tipo de fato são do tipo texto/string, precisava-se de uma referência numérica como chave primária, pois com um campo do tipo string as agregações são bem mais lentas. Então por ganho de desempenho e por integridade dos dados na geração dos resultados optou-se por utilizar essa estrutura. Da mesma forma a tabela `mdl_accessmodules`, pois era necessário uma referência dentro da tabela de fatos e houve um ganho significativo em desempenho da consulta.

4.2.3 Interface OLAP

A modelagem da ferramenta OLAP será feita dentro de um novo módulo implementado dentro do sistema Moodle. Para esse novo módulo foi utilizado o conceito OLAP chamado MOLAP já citado anteriormente. Esse padrão é frequentemente implementado em um banco de dados relacional, porém não na terceira forma normal (ANZANELLO, 2005).

A idéia é criar uma interface para o cubo de dados proposto em cima da arquitetura de um DM seguindo o esquema estrela, para que os tutores e gestores do Moodle possam ter acesso e ter uma ferramenta importante na tomada de decisões. Essa interface entre dados e gestores precisa ser muito intuitiva, pois os tutores possuem as mais diver-

sas formações e não necessariamente possuem um conhecimento avançado em sistemas computacionais.

O desenvolvimento é detalhado na próxima seção.

5 Desenvolvimento Plug-in de referência

O desenvolvimento de um módulo interno ao moodle se baseia em entender o funcionamento padrão do mesmo. Várias estruturas de procedimentos e funções internas do sistema, e toda a parte de padronização de código para que o plug-in pudesse ser integrado ao sistema do Moodle padrão.

Esse procedimento consiste na escolha de ferramentas compatíveis com a versão escolhida e no entendimento de toda documentação disponível em <http://docs.moodle.org/19/en/Blocks/Howto>. Esse capítulo é dividido em 4 partes: Requisitos mínimos para a implementação; Como gerar um novo módulo; Desenvolvimento da ferramenta OLAP; Layout.

5.1 Requisitos Necessários

Para a implementação da ferramenta foi utilizado a versão 1.9.14+ do Moodle. É uma versão estável da plataforma, as demais versões estão disponíveis para o download gratuito na página da ferramenta.

O Moodle é um sistema desenvolvido na linguagem PHP. A versão do PHP utilizada para a criação do módulo foi a 5.3.5.

A plataforma suporta vários tipos de Sistemas Gerenciadores de Bancos de Dados(SGBD). Para o novo módulo foi escolhido o MySQL versão 5.5.8. A escolha do MySQL foi feita por ser um sistema que utiliza a licença GPL(General Public License)¹ e que possui uma grande robustez para sistemas web. Outro parâmetro utilizado na escolha do SGBD foi a escalabilidade, o MySQL suporta mais de 20 plataformas de desenvolvimento, o que traz uma grande mobilidade na implementação da plataforma, além de possuir suporte nativo a triggers e stored procedures.

¹www.gnu.org/licenses/gpl.html

5.2 Criando um novo bloco no Moodle

A criação de um novo bloco no Moodle é baseada na linguagem PHP assim como toda a plataforma. Para a criação de um novo módulo além do conhecimento em PHP deve-se entender as funções e procedimentos do framework próprio do Moodle. Na página da plataforma existe uma vasta documentação de todas as funções.

Existem 2 tipos diferentes de funcionalidades, o módulo e o bloco.

- O módulo possui uma característica de interação direta com o usuário. Ele é desenvolvido para se integrar as demais funcionalidades de ação direta do usuário final, no nosso caso o aluno. São exemplos de módulo: quiz, formulário de pesquisa, etc.
- O bloco possui uma característica de ser uma funcionalidade administrativa do sistema. Ele pode ou não ter interação direta com o usuário final, mas normalmente é acessado por administradores e tutores. Ele se assemelha as demais ferramentas de manutenção e administrativas como relatórios, gráficos, backup do sistema, etc. Outra característica importante do bloco é que o mesmo pode ter vínculo direto no layout principal da plataforma, assim podendo ser acessado de forma mais rápida e intuitiva.

Para esse sistema foi escolhido o Bloco justamente por se melhor encaixar na categoria de utilitário do sistema. Tendo maior flexibilidade e segurança para trabalhar com dados de todos os usuários do sistema.

A estrutura básica de um novo Bloco permite que ele seja integrado ao layout de forma a se tornar um utilitário fixo no sistema. Possui diretivas de restrição em relação ao nível de usuários. Pode-se desenvolver um novo bloco apenas acessível aos administradores do sistema, apenas aos administradores dos cursos, etc. Esse controle é muito importante pois os dados tratados podem ser sigilosos para algum nível de usuário no sistema.

O primeiro passo para o desenvolvimento do bloco é a criação de uma pasta com o nome do bloco a ser desenvolvido na hierarquia de pastas do Moodle dentro do diretório “blocks”. Nesse diretório se encontram as pastas de todos os blocos presentes no sistema.

Após a criação da pasta é necessário que se crie o arquivo base para o bloco, `block(nomedobloco).php`. Nesse primeiro arquivo deve-se estender a classe base dos blocos

blockbase e inicializar o método *init()*:

```
1 class block_olap extends block_base {
2     function init() {
3         $this->title    = get_string('olap', 'block_olap');
4         $this->version = 2004111200;
5     }
```

O método *get_content()* é um método importante que trabalha com a interface de chamada da ferramenta. Nesse método é instanciado a maior parte do conteúdo que terá vínculo com o layout do Moodle.

```
1 function getcontent() {
2     global $USER, $CFG, $COURSE;
3     if ($this->content != NULL) {
4         return $this->content;
5     }
6     if (!isloggedin())
7         return $this->content;
8     $course = getrecord('course','id',$COURSE-id);
9     $this->content = new stdClass;
10    $this->content->text = '<a href="'. $CFG-wwwroot.'/blocks/
        olap/analysis.php">' . getstring('analysis', 'blockolap') . '</a>';
11    $this->content->footer = '';
12    return $this->content;
13 }
```

Seguindo a documentação indicada na página http://docs.moodle.org/19/en/Blocks_Howto, o desenvolvimento dos demais métodos é feito de forma sequencial e padrão para todos. Destaca-se outro método importante na construção de um bloco básico é o método *specialization()*. Esse método dentro do arquivo *block(nome_dobloco)* é o responsável pelo título do bloco no layout do Moodle.

```
1 function specialization() {
2     if (!empty($this->config->title)) {
3         $this->title = $this->config->title;
4     } else {
5         $this->config->title = 'OLAP';
6     }
7 }
```

Em um novo arquivo *locallib.php* defini-se toda a estrutura principal de chamada. Até o momento o bloco possui a estrutura pronta mas não possui nenhuma funcionalidade efetiva. Nesse arquivo implementamos a funcionalidade em si. Essa funcionalidade pode

utilizar toda a API do Moodle para criar blocos administrativos semelhantes aos que já se possui, ou pode fazer o desenvolvimento em linguagem PHP nativo e fazer a chamada do método criado recentemente.

Na implementação da ferramenta proposta por esse trabalho, o desenvolvimento foi todo feito em PHP nativo, pois pela utilização dos recursos gráficos e de algumas classes em JavaScript houve a necessidade de não usar a API do Moodle para o conteúdo, apenas foi utilizada na montagem da estrutura.

5.3 Implementação Ferramenta OLAP

Após a criação de um bloco básico sem nenhuma funcionalidade ativa, vamos inicia-se o desenvolvimento da ferramenta OLAP. Como mencionado no capítulo anterior o desenvolvimento da ferramenta também será feito na linguagem PHP. Conforme visto no capítulo 2 a ferramenta OLAP tem como característica a visualização dinâmica de dados multidimensionais, para tal, fez-se necessário a utilização de recursos gráficos para facilitar a manipulação dos mesmos. Estruturas utilizando JavaScript e Json foram implementadas dando mais robustez na dinâmica dos dados.

A linguagem JavaScript é uma linguagem interpretada, ou seja, não existe compilação dos dados por nenhum compilador externo. O browser interpreta o código de forma a interagir com o usuário, sem a necessidade do código passar pelo servidor. O Json é um formato leve para a movimentação de dados e possui a característica de fazer o acesso e retorno dos dados dinamicamente, sem carregamento da página pelo browser, sendo uma solução muito interessante para o tipo de aplicação proposta.

A implementação será dividida em algumas partes:

- Carregamento e acesso aos dados
- Criação das Medidas
- Visualizando as Dimensões do Cubo
- Geração das Tabelas de Resultados
- Exportação dos dados

Para fazer com que a interação com o usuário fosse mais simples possível utilizamos a biblioteca JQuery, pois a mesma possui componentes de interação com o usuário muito eficazes e de boa usabilidade.

O projeto foi desenvolvido para que a cada interação válida do usuário gerasse relatórios de dados de forma dinâmica e rápida. Cada ação gerada pelo sistema deve ser bem clara e simples para o usuário. Para facilitar essa interação o usuário não tem a necessidade de digitar nada, apenas posicionar o cursor no campo desejado, pressionar e arrastar para o campo correspondente. Isso facilita muito a dinâmica do processo tanto quanto minimiza possíveis erros de operação.

5.3.1 Layout

O layout foi desenvolvido com base em modelos clássicos de ferramenta OLAP. As colunas à direita representando as dimensões e medidas disponíveis, separadas entre camada superior e inferior. Na área central e direita possuímos os campos correspondentes à disposição dos dados na tabela de resultados, qual campo irá compor a linha, qual campo irá compor a coluna e qual medida será utilizada. Na área inferior esquerda temos o local onde o resultset será exibido.

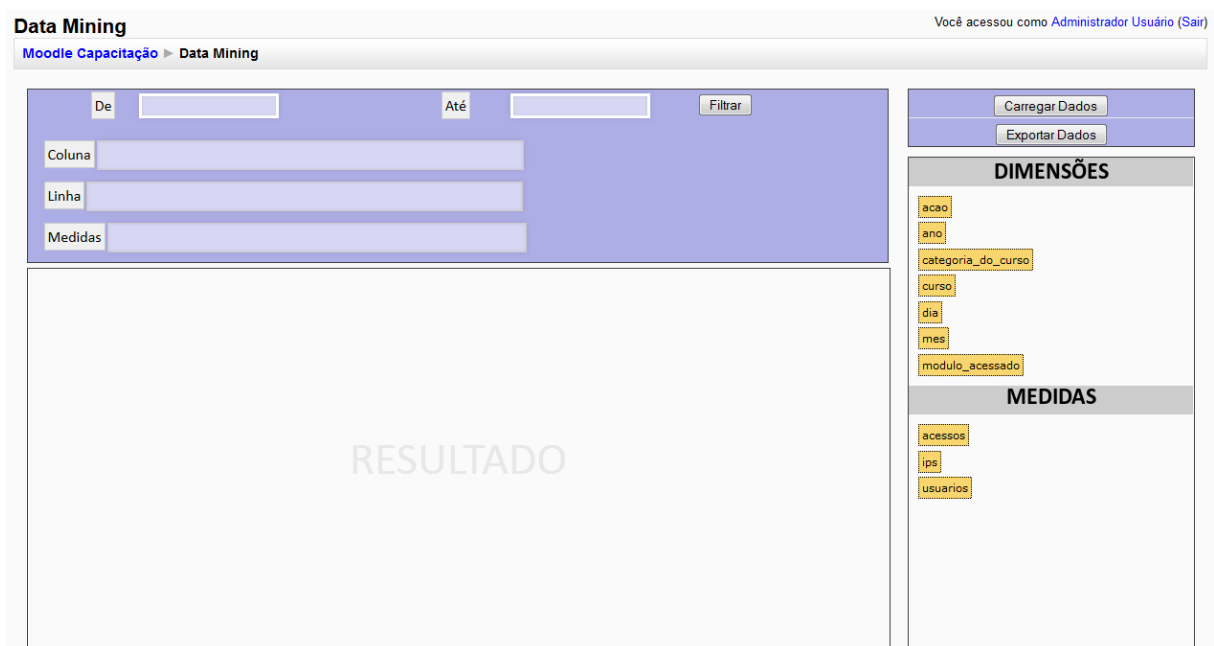


Figura 5.1: Layout da Ferramenta

Os principais métodos utilizados para a interação feita no layout da ferramenta foram as funções *draggable* e *droppable* que se encontram no pacote JQuery. Essas funções são as responsáveis pelos movimentos de arraste e solte das dimensões e medidas para os campos correspondentes. São métodos que possuem uma vasta documentação e são de fácil implementação.

5.3.2 Carregamento e acesso aos dados

O primeiro módulo da implementação desenvolvido é o acesso aos dados e posterior carregamento dos mesmos. O acesso de dados será todo via a linguagem T-SQL, forma mais simples e robusta de acessar os dados, utilizando funções já existentes no próprio PHP.

Foi desenvolvida uma classe com o nome *queryCube*. Essa classe possui todos dos métodos de acesso e inclusão de dados. A utilização da classe específica para o banco de dados, cria uma abstração necessária dos demais módulos, além de deixar o código bem enxuto e conciso.

Os principais métodos que compõem essa classe são:

- *getColumns()*: Esse método busca através da tabela *information_schema*, presente em todos os SGBDs, e que contém informações sobre todas as tabelas e colunas do banco. Esse método então faz a busca pelas colunas através da tabela que utilizamos como nossa fonte de consulta. Seu retorno são os nomes dos campos da tabela do nosso DM.
- *setGroup(value)*: Método para realizar as agregações dos valores que estão nos campos linha e coluna.
- *setPivot(column, row)*: Um dos métodos mais importantes da classe, ele faz a junção e a interposição dos dados entre coluna e linha. Quando o usuário escolhe qual campo vai ser utilizado como coluna ou linha é necessário fazer a junção dessas tabelas de acordo com a medida escolhida. Primeiramente os dados são agrupados com a função *setGroup*, o retorno dessa função juntamente com as respectivas funções que retornam os filtros escolhidos pelo usuário são a base para essa interpolação entre

os dados. O comando T-SQL utilizado para fazer a interpolação é o *SUM*, esse comando juntamente com o agrupamento permite que os dados sejam agrupados e gerados dinamicamente, ou seja, o conceito de granularidade do Cubo de Dados é expresso nessa condição. Dependendo do filtro utilizado e da data escolhida como filtro os dados são expandidos ou não.

- *buildCube()*: Esse método é utilizado para fazer a geração e seleção dos dados para o nosso DM.
- *returnqueryCube()*: Esse método é utilizado para realizar a junção dos dados participantes da agregação linha x coluna e envia os mesmos para a função *setPivot* para fazer a interpolação dos mesmos.

5.3.3 Criação das Medidas

A criação das medidas é um passo muito importante na implementação e no desenvolvimento da arquitetura. A geração de todos os dados históricos para dar suporte as tomadas de decisão só são válidos se as formas de mensuração são de alguma relevância e importância para essa tomada de decisão.

A estrutura da tabela de fatos e dimensões já foi exposta na seção 4.2. A implementação desses dados se torna mais simplista se na concepção da tabela a máxima granularidade for alcançada. Para a medida *qtacao* não precisamos fazer nenhuma operação pois na tabela de fatos já é guardada como soma, por ser o valor mais unitário da tabela.

As medidas *qtusuarios* e *qtip* são calculadas fazendo uma agregação simples de acordo com a dimensão escolhida.

As dimensões são calculadas dinamicamente pelos campos da tabela que começam com a letra D_nomecoluna. Esses dados são cruzados pelo nome da coluna com a sua tabela de origem através da chave estrangeira correspondente. Para dar maior agilidade nas agregações e demais operações todos os campos são do tipo inteiro e depois de feitas as interpolações e agregações as informações são cruzadas com as tabelas correspondentes.

5.3.4 Criação dos Filtros

A utilização dos filtros é muito importante para essa visão histórica dos dados. Logicamente a visão de um todo também é importante, mas identificar padrões em determinados períodos, em determinadas características do banco são de grande valia. Durante a implementação da ferramenta definimos alguns filtros relevantes para a visualização dos dados.

O primeiro dos filtros desenvolvidos foi os filtros por data. Como vimos nas estruturas das tabelas do Moodle a data de cada ação na tabela de Log do Moodle é guardada no formato timestamp Unix. Esse tipo de formato facilita bastante a comparação entre datas pois é apenas uma comparação de inteiros. Na implementação realizada essa comparação é feita através de uma função presente na classe *queryCube* que se chama *setDateFilter(dtini, dtfim)*.

```
1  function setDateFilter($dt_ini, $dt_fim) {  
2      if ($dt_ini AND $dt_fim){  
3  
4          $aux1 = strtotime($dt_ini);  
5          $aux2 = strtotime($dt_fim);  
6          $this->filter_date = 'time between ' . $aux1 . ' and ' . $aux2 . ' ' ;  
7  
8      }else  
9          $this->filter = null;  
10  
11 }
```

Além dos filtros de data temos os filtros dos campos que compõem as colunas e linhas da tabela. Ao final de cada ação do usuário para a geração das tabelas, o campo que foi escolhido, tanto como coluna ou como linha, é exibida uma lista com todos os valores válidos para o respectivo campo, para que se faça o filtro correspondente.

Os filtros tanto de data como os filtros por campo são aplicados simultaneamente para dar maior precisão nos dados solicitados. Na implementação temos algumas funções que geram os filtros. Na função *setFilterRow* temos a geração do filtro do campo que está na linha dos dados. Essa função monta a linha de filtro em comando SQL para a consulta final dos dados. Da mesma forma possuímos a função *setFilterColumn* que é a mesma funcionalidade para os dados de coluna. Quando um usuário seleciona todos os filtros simultaneamente temos que gerar um único comando em SQL que possua todos

The screenshot shows a web-based OLAP tool interface. At the top, there are input fields for 'De' (From) and 'Até' (Until) with a 'Filtrar' (Filter) button. Below these are fields for 'Coluna' (Column) set to 'curso', 'Linha' (Row) set to 'ano', and 'Medidas' (Measures) set to 'acessos'. There are also 'Filtro' (Filter) dropdowns. The main area displays a pivot table with 'curso' as rows and 'ano' as columns. The table data is as follows:

	2011	2012	2013
av_port	2376	791	35
AVA_2011	5257	135	0
MDI	189	4	0
atv_ef	952	3	0
atv_cd	733	1	0
atv_tl	670	1	0
atv_mf	103	0	0
atv_lc	482	0	0
atv_p	377	1	0
atv_f	990	2	0
atv_fr	199	17	0

On the right sidebar, there are buttons for 'Carregar Dados' (Load Data) and 'Exportar Dados' (Export Data). Below these are two sections: 'DIMENSÕES' (Dimensions) with a list of dimension fields (acao, ano, categoria_do_curso, curso, dia, mes, modulo_acessado) and 'MEDIDAS' (Measures) with a list of measure fields (acessos, ips, usuarios).

Figura 5.3: Geração da Tabela de Resultados

A função mais importante desse módulo é a que faz a interpolação dos dados entre linha e coluna, *setPivot*. Essa função conforme relato anterior faz a junção e interpolação dos dados de forma dinâmica. Segue a função:

```

1 function setPivot($column, $row) {
2
3     $sql = $this->mountFilterRow();
4     $query = mysql_query($sql);
5
6     $aux = "SELECT IFNULL(" . $column . ", 'totais') AS teste";
7     $this->headerPivot = "<tr><th>" . str_replace("d_", "",
8         $column) . "</th>";
9
10    if (mysql_num_rows($query)) {
11        while ($dados = mysql_fetch_row($query)) {
12            foreach ($dados as $key => $value) {
13                $aux = $aux . ',tabela.' . $value;
14                $aux2 = $aux2 . " ,SUM(IF(b." . $row . "=" .
15                    $value . ",b.total,0)) AS '" . $value . "'";
16                $this->headerPivot = $this->headerPivot . "<th>"
17                    . $value . "</th>";
18            }
19        }
20    }
21
22    $this->headerPivot = $this->headerPivot . "</tr>";
23    $aux = $aux . " FROM ( SELECT b." . $column;
24    $aux = $aux . $aux2;
25    $aux = $aux . " FROM ( " . $this->returnqueryCube() . " ) AS
26        b GROUP BY " . $column . " WITH ROLLUP ) AS tabela ";

```

```
23     return $aux;  
24 }
```

Essa função exporta uma query em SQL que faz a geração da tabela. Note que independente dos campos de dimensão e coluna que serão colocados sempre será uma tabela agrupado de acordo com a granularidade escolhida pelo usuário através dos filtros disponibilizados.

5.3.6 Exportação dos dados

Depois de geradas todas as consultas, pode-se extrair a tabela e exportar para o formato xls. O dispositivo desenvolvido para essa exportação é bem simples e interativo. O botão no canto superior esquerdo abre uma janela para salvar o arquivo onde o usuário desejar.

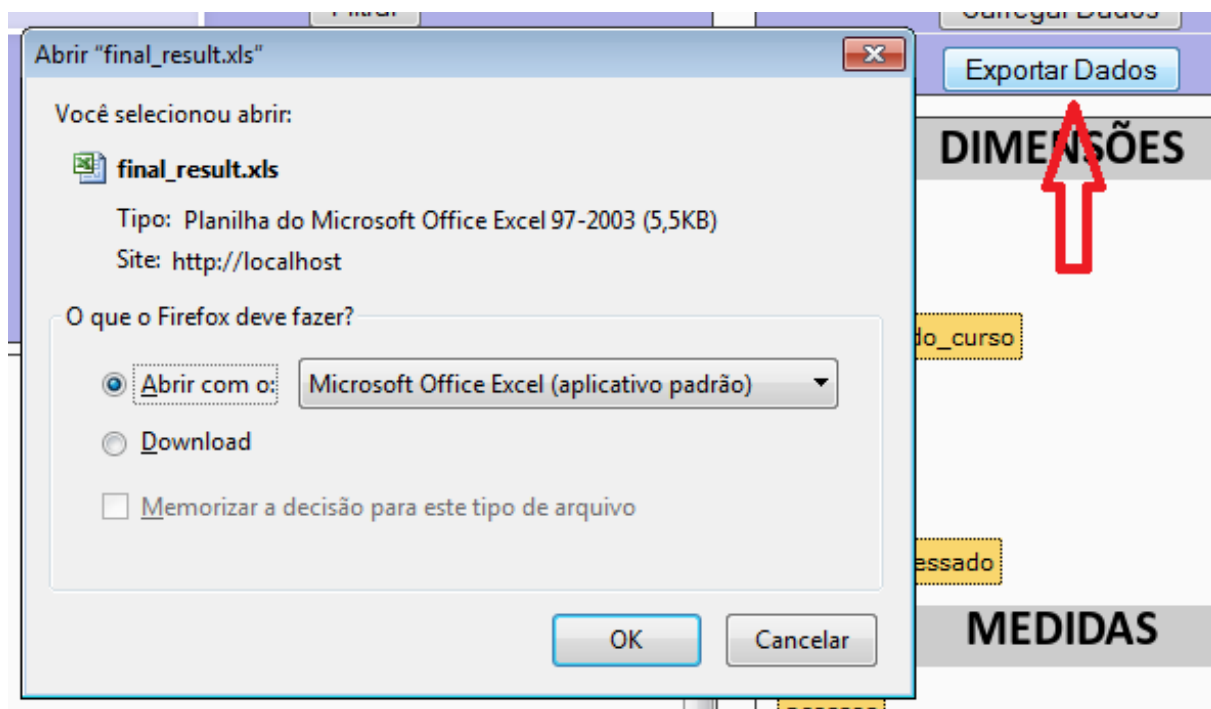


Figura 5.4: Botão para a exportação dos Dados

6 Análise de Resultados

Antes de se analisar a estrutura, efetividade e eficácia da ferramenta uma análise do processo de criação é válido para que se tenha noção exata das etapas que foram percorridas até o fim do trabalho.

Uma etapa de grande importância e que se tornou custosa ao longo do desenvolvimento foi a etapa ETL. Mesmo os dados vindos do Moodle serem de grande integridade e bem “limpos” um processamento teve que ser feito. Em todas as etapas de desenvolvimento ajustes tiveram que ser feitos para que a compreensão do usuário do que estava sendo medido fosse clara. Em alguns campos das tabelas o Moodle utiliza referências sem padrão, ou seja, dentro de um mesmo campo podemos ter endereços url, nomes de colunas, números, etc. Como cada parte da plataforma, em momentos específicos utilizam isso faz sentido junto ao sistema. Estes mesmos campos, mesmo que as vezes despadronizados, possuem informações relevantes a análise dos dados e sempre uma extração e posterior tratamento era feito.

Outra etapa essencial foi o desenvolvimento do bloco propriamente dito junto ao Moodle. A documentação não foi um fator impeditivo na implementação. Por se tratar de um sistema com licenças de código aberto a documentação é vasta e conta com vários foruns dos desenvolvedores colaboradores da ferramenta. O possibilidade da inserção de código PHP puro dentro do núcleo do bloco desenvolvido também destravou a programação de estruturas com tecnologias atuais e com recursos gráficos mais eficazes e com uma adaptação mais fácil ao código.

De acordo com as observações feitas sobre a estrutura da ferramenta, pode-se constatar algumas verificações importantes acerca do trabalho. As observações começam verificando se as características principais de uma aplicação em um padrão OLAP foram atingidas. A primeira característica marcante observada é a simplicidade no uso da ferramenta, qualquer usuário, com um nível básico de conhecimento em sistemas computacionais consegue operar a ferramenta de forma satisfatório e consegue extrair da mesma informações de forma rápida e intuitiva.

Outro conceito muito importante atingido na estrutura da ferramenta é a visualização dos dados de forma multidimensional, utilizando o conceito de cubo de dados. Fazendo um paralelo com os 12 conceitos básicos de Codd, apresentado anteriormente, esse seria o primeiro conceito. A liberdade na escolha de qual face visualizar é requisito básico esse tipo de visualização.

Continuando com o paralelo com os conceitos básicos de Codd, pode-se citar a transparência dos dados em relação aos usuários. Como não existe digitação de medidas ou dimensões o foco do mesmo é direcionado totalmente para a geração do relatório, deixando a utilização e o entendimento mais suave ao usuário. O conceito de arraste dos dados deixa o ambiente de navegação mais transparente e amigável ao usuário.

Além da utilização da ferramenta, um conceito importante é a performance e a qualidade dos relatórios gerados. Na implementação da ferramenta em linguagem PHP voltado para um dispositivo que fosse integrado a um sistema existente, algumas estruturas poderiam ser menos efetivas em função desse ajuste funcional exigido. Verificou-se que a utilização de estruturas genéricas, para que a ferramenta fosse a mais adaptável possível a diferentes conjuntos de bases de dados, criou uma complexidade maior na junção dos dados exportados. Percebendo esse aumento de complexidade que poderia impactar na qualidade da ferramenta, alguns padrões da modelagem inicial do banco de dados tiveram que ser alterados. Basicamente a estrutura inicial da tabela de fatos, ondes os dados não deveriam ser normalizados para que o acesso ao banco fosse menos custoso, teve que ser alterada. Em um banco de dados, com muitos registros, onde são feitas agregações e junções constantes e quase todos os campos dos tipos String participam dessas agregações é inevitável que a complexidade dessas operações e o tempo de retorno dado vão ser grandes.

Para amenizar o tempo de retorno, foi feita uma alteração nessa disposição. A tabela de fatos foi alterada de forma a apenas possuir valores numéricos e depois da montagem dos relatórios dinâmicos(agregações/junções), os valores de referência fossem acrescentados a tabela de resultados. Ou seja, após as operações dinâmicas com os dados numéricos(complexidade simples), os valores de referência que pertecem a esses códigos foram inseridos para que os dados exportados tenham qualidade maior no entendimento

do usuário. Essa alteração fez com que a sensação de transparência dos dados se tornasse mais evidente junto ao usuário, pois a complexidade das operações foi amenizada através desse artifício.

Alguns conceitos interessantes não puderam ser observados de forma isolada nesse trabalho. Em uma ferramenta desse nível de interatividade, é muito importante a análise da disposição dos recursos aos usuários no momento as gerações dos dados. Como utilizamos a mesma estrutura do Moodle, temos os mesmos recursos de sistema e hardware que o sistema Moodle em si. Como é interessante separarmos as estâncias de banco de dados transacionais e do nosso DW, poderemos ter em alguns uma disponibilidade maior de recursos para a utilização da ferramenta, mas que na maioria dos casos não relevante para o estudo da ferramenta. O controle de acesso aos dados de forma externa ao Moodle utiliza a mesma diretiva de segurança implementada na plataforma Moodle.

Outro detalhe pertinente a análise é a capacidade de interpolação dos dados linha X coluna de forma eficiente. Essa conversão de eixos é muito interessante pois tras uma perspectiva nova ao relatório gerado. Na implementação essa interpolação ocorre de forma suave e sem restrições às dimensões disponíveis. Esse cruzamento de dimensões que previamente ainda não tinham sido utilizadas juntas gera um padrão ainda antes não observado e que pode trazer informações importantes.

Os filtros utilizados trazem uma especialização de dados, que é a capacidade da ferramenta filtrar os dados o máximo possível de acordo com a necessidade do usuário. A análise de um momento isolado pode ser decisiva para se aferir o padrão adotado em anos anteriores. Por exemplo: Se em uma determinada faixa do ano percebe-se uma grande dispersão nos acessos, um grande egresso, pode-se trabalhar na melhor disposição das tarefas e demais atividades e até a necessidade de um feedback dos alunos com relação à esse período em especial. O que estaria acontecendo nessa época? O que pode-se trabalhar melhor esse ano? São questões desse tipo que fazem a tomada de decisão importante para um sistema de gestão integrada. Na ferramenta trabalhada pode-se filtrar por qualquer medida e por períodos específicos e tem-se também ano e mês como medida de agregação de dados para que um estudo temporal seja feito.

A idéia do usuário ter o domínio do retorno dos dados a qualquer momento

reforça o conceito de dinamicidade e tomada de decisão real time. Na implementação da ferramenta, teve-se uma preocupação para que ela fosse o mais intuitiva e simples possível, para justamente fornecer a sensação de mobilidade e flexibilidade necessária para o gestor ou tutor que utilizará a ferramenta.

Além desses critérios utilizados podemos adotar um critério que é mais estrural do que operacional, a relevância dos dados. A verificação da objetividade do que será medido depende da modelagem correta do banco. Quando o banco é analisado, verificando a quantidade de usuários acessando o curso X em determinada data, busca-se através de uma análise histórica algum movimento que tenha ocorrido perto do período analisado para verificar se o quantitativo aferido corresponde à uma situação normal ou se existe alguma anormalidade nesse contexto.

assignment	681	203	4
chat	0	16	0
choice	138	2	0
data	1	0	0
forum	2202	735	2
glossary	1	0	0
label	22	2	0
quiz	620	86	0
resource	629	507	12

Figura 6.1: Análise: Módulo Acessado X Ano

Analisando os dados gerados percebe-se que em determinados relatórios observa-se dados que não seriam analisados em relatórios gerenciais convencionais. Na tabela módulo acessado X ano, pode-se perceber a discrepância entre o número de acessos no módulo fórum com os outros módulos, conforme fig. 6.1. Continuando na análise dessa dimensão observa-se que a maioria significativa desses acessos foram apenas para visualização de mensagens do fórum conforme consta na face do cubo módulo acessado X

ação. Quando se analisa os dados em relação aos dias e meses com mais frequência nos acessos se percebe-se uma linearidade maior nos acessos. Essa mudança de perspectiva entre as faces e a velocidade com que elas acontecem trazem para o gestor que utiliza a ferramenta a possibilidade de verificar por entre essas faces informações que poderiam não ser observadas com tanta clareza.

update mod	update post	upload	user report	view	view all	view discussion	view forum	view submission
185	0	0	1	7519	387	0	0	0
0	0	42	0	685	0	0	0	82
0	0	0	0	8	0	0	0	0
0	0	0	0	116	0	0	0	0
0	0	0	0	0	0	0	0	0
0	26	0	104	0	0	1046	1509	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	205	0	0	0	0
0	0	0	0	1004	0	0	0	0
185	26	42	105	9537	387	1046	1509	82

Figura 6.2: Análise: Módulo Acessado X Ação Efetuada

Para esse trabalho utilizou-se um banco amostral de dados retirado do banco de dados dos cursos da Universidade Federal de Juiz de Fora, para tornar essa análise mais real possível. Utilizando a ferramenta tirou-se algumas conclusões sobre essa amostra e o retorno que essa ferramenta pode trazer. Analisando o primeiro item que seria a quantidade de acessos por curso e por ano sobre toda a amostra.

Alguns parâmetros foram alterados no momento da implementação para dar uma maior praticidade e agilidade na ferramenta. O agrupamento dos dados seguindo o modelo teórico proposto, ou seja, seguindo o conceito de que a tabela de fatos deva ser a congruência dos dados que fazem a interlocução dos dados no DW, para que a aplicação funcionasse de forma mais ágil, foi implementado a tabela de fatos apenas com os Ids correspondentes das dimensões dos cenários. Dessa forma os agrupamentos e transições do

cubo de dados na aplicação acontece mais rápido. No momento da exibição dos dados na tela, os ids são utilizados como referências para o join entre as tabelas e então a exibição da descrição dos dados.

Os valores utilizados na aplicação foram extraídos de um banco de teste do Centro de Educação a Distância(CEAD) da Universidade Federal de Juiz de Fora e serviram como base para as análises feitas. Os valores encontrados foram utilizados apenas como referência e comprovação da utilização da ferramenta.

Diversas informações podem-se ser extraídas da ferramenta, mas não era o objetivo maior nesse trabalho. Como informado anteriormente o objetivo era desenvolver a ferramenta e parametrizá-la para que a mesma sirva como base para bases reais do moodle e em bancos transacionais reais. Como estudo e análise de possibilidades para um melhor desenvolvimento de tecnologias que possam auxiliar os gestores da educação. Analisando dados isoladamente, como o de um curso, uma ação específica do sistema, uma categoria de curso, etc, sobre todas as faces disponíveis cria um ambiente favorável a abstrair o cenário padrão dessa variável e fazer a interlocução real desses dados com outras variáveis que também compõe e atuam no cenário proposto.

6.1 Trabalhos Futuros

Como relatado no decorrer deste trabalho, têm-se essa implementação como o marco inicial para a análise de outros nichos de dados dentro do banco do Moodle. Seriam vários DMs formando um grande e compacto DW, fornecendo velocidade e informação para processos gerenciais. Como esse trabalho focou em dados de acesso, pode se tornar objeto de estudo dados educacionais como, pontualidade, efetividade nas atividades e uma gama de outras linhas de trabalho. A base da ferramenta está pronta e facilmente pode ser customizada para outras tabelas. Propõe-se ampliar a ferramenta para que sejam gerados outros tipos de análises gráficas, estatísticas e diversas outras formas de análise.

Através dessa base montada é possível expandir a ferramenta para outros tipos de análises textuais e gráficos e além de algoritmos Data Mining para reconhecimento de padrões.

7 Conclusão

Durante todo o desenvolvimento do estudo e da implementação, concluí-se que existem diversas maneiras de aproveitar o momento de grande evolução da informatização de todas as áreas de estudo. A possibilidade de cruzar dados e aferir resultados antes nunca esperados ou mensurados é de grande valia e aumenta a visão de todos sobre determinados aspectos.

O objetivo desse trabalho foi sempre abrir as possibilidades de análises de dados em fontes possivelmente pouco aproveitadas. A utilização da tabela Log do Moodle é apenas a primeira parte de um trabalho que pode ser expandido para as demais áreas dentro da plataforma. A velocidade na geração de análises específicas de dados é uma saída de extrema importância para a tomada de decisões rápidas e com um embasamento teórico.

O projeto foi feito de forma a desenvolver o primeiro módulo de um tipo de arquitetura que fizesse a extração e análise dos dados em um mesmo ambiente e a visualização em tempo real.

A criação de ferramentas com esse conceito criam uma maior noção de gerência dos processos de todos os envolvidos. O acesso rápido à informação auxilia o acompanhamento histórico de padrões ou medidas importantes. Esse acompanhamento é necessário pois sempre é necessário rever processos e medidas que não surtem efeitos ou são ineficientes. Para quem atua na ponta do processo de aprendizagem essa informação ajuda na análise dos mesmos e ajuda a expandir a percepção dos problemas ou soluções para os demais níveis de hierarquia na gestão de processos educacionais.

O Moodle é uma plataforma que possibilita a implantação de várias estruturas customizadas e encontramos uma grande documentação a respeito das mesmas. O desenvolvimento foi feito sempre levando em conta a incorporação do módulo na plataforma e que fosse um módulo mais genérico possível. Como ainda tem-se muitas áreas de interesse a se expandir, essa implementação servirá de base para as futuras pesquisas e implementações.

A utilização de um plugin integrado a uma ferramenta de ensino, especificamente o Moodle, deixa em aberto para que a expansão seja feita para as outras áreas de interesse do próprio Moodle, mas que também sirva de base para outros sistemas/plataformas e outras áreas de acesso.

Referências Bibliográficas

1. ALMEIDA, M. E.B. Educação a distância na internet: abordagens e contribuições dos ambientes digitais de aprendizagem. *Educação e Pesquisa*, São Paulo, Brasil, v. 29, n. 2, p. 327–340, 2003.
2. ANZANELLO, C. A. OLAP *Conceitos e Utilização* Porto Alegre, Rio Grande do Sul, Instituto de Informática, UFRGS. Disponível em: www.inf.ufrgs.br/~clesio/cmp151/cmp15120021/artigo_cynthia.pdf Acesso em: 17-01-2012.
3. BISPO, C. A. F. ; CAZARINI, E. W. A nova geração de Sistemas de Apoio à Decisão *ENEGEP 2008* Disponível em: http://www.abepro.org.br/biblioteca/enegep1998_art222.pdf Acesso em: Fev/2013.
4. BORTOLI, A. L. Um projeto de data warehouse. *Universidade Mater Dei* Disponível em: <http://www.materdei.edu.br/arquivos/Um%20Projeto%20de%20Data%20Warehouse.pdf> Acesso em: Março/2013.
5. CHAUDHURI, S; DAYAL, U; An Overview of Data Warehousing and OLAP Technology. *ACM Sigmod Record*, 1997. Disponível em: <http://research.microsoft.com/pubs/76058/sigrecord.pdf> Acesso em: 17-01-2013.
6. CODD, E. F. ; CODD, S. B.; SALLEY, C. T. Providing OLAP (On-line Analytical Processing) to User-Analysts: An IT Mandate Disponível em: http://www.minet.uni-jena.de/dbis/lehre/ss2005/sem_dwh/lit/Cod93.pdf Acesso em: Junho/2013.
7. COSTA, C. M. Uma generalização do processo etl em sistemas data warehouse. *Universidade Federal do Ceará*, 2007. Disponível em: <http://www.lia.ufc.br/~claytonmaciel/dw.pdf> Acesso em: 17-08-2012.
8. FALSARELLA, O. M.; CHAVES, E. O. C. Sistema de Informação Gerencial. Disponível em: http://www.cpmestadualcm.seed.pr.gov.br/redeescola/escolas/5/430/12/arquivos/File/apostilas/%20dos%20professores/Apostila_11.pdf Acesso em: 30-09-2012.
9. FORSMAN, S. OLAP *Concil with Paper* Olap Concil Disponível em: www.olapcouncil.org/research/whtpapco.html Acesso em: Fev/1998.
10. IMHOFF, C.; GALEMMO, N.; GEIGER, J. G. Mastering data warehouse design. *Relational and Dimensional Techniques*. Indianapolis, Indiana: Wiley, 2003.
11. INMON, W. H. Building the Data Warehouse Indianapolis, Indiana: Willey, 4 e, 2005. 576p.
12. JUNIOR, R. F. T.; FERNANDES, F. C. F.; PEREIRA, N. A. Sistema de apoio à decisão para programação da produção em fundições de mercado. *Gestão & Produção*, São Paulo, Brasil, v. 13, n. 2, p. 205-221, 2006.

13. JUNIOR, S. A. M. B. Sistemas de apoio à decisão. Disponível em: <http://www.al.urcamp.tche.br/infocamp/edicoes/nov05/Apoio\%20a\%20Decisao.pdf> Alegre, Rio Grande do Sul Acesso em: 08-12-2012.
14. KIMBALL, R. The Data Warehouse Toolkit: *The Complete Guide to Dimensional Modeling*. 2 e, Wiley, 2002.464p.
15. PERKINS, A. Developing a data warehouse: The enterprise engineering approach. Disponível em: <http://www.ies.aust.com/PDF-papers/dw.pdf> Acesso em: 30-11-2012.
16. SANT'ANNA, I. F. B. T. Um Roteiro para Construção de Cubos e Consultas OLAP *Instituto Nacional de Pesquisas Espaciais* Disponível em: <http://mtc-m18.sid.inpe.br/col/lac.inpe.br/worcap/2004/10.05.09.36/doc/WorCapQuatroIris.PDF> Acesso em: Junho/2013.
17. TAN, P; STEINBACH, M; KUMAR, V; Introdução ao Data Mining *Mineração de Dados*. Ciência Moderna;1 e, 2009.928p.
18. TAURION, C. O Data warehouse será útil para sua organização? *Revista Developers*, p. 26-27, fev/1998.