UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
CURSO DE CIÊNCIA DA COMPUTAÇÃO

CASSIANO ROCHA KUPLICH

Desenvolvimento de Data Warehouse e Ferramenta OLAP para a Análise da Produção Acadêmica de Pesquisadores: Estudo de Caso no PPGC

Trabalho de Graduação.

Prof. Dr. Leandro Krug Wives
Orientador

Porto Alegre, dezembro de 2013

“Data is a precious thing and will last longer than the systems themselves.”
— TIM BERNERS-LEE

AGRADECIMENTOS

Agradeço aos meus pais que sempre se esforçaram para me proporcionar uma boa formação e sempre me incentivaram nos estudos. Agradeço a eles também pelo seu amor e carinho incondicionais que me dão força nos momentos difíceis e muita alegria nos momentos felizes.

À Lívia Freire pela sua amizade, apoio e companheirismo ao longo do curso e por ter me apresentado outro grande companheiro, o Inácio, meu gato de estimação.

Aos amigos de infância e aos amigos e colegas que conheci ao longo da faculdade pelo seu apoio em trabalhos e pelas divertidas e interessantes conversas nos intervalos entre as aulas e nos horários de almoço.

Ao meu orientador, Professor Doutor Leandro Krug Wives, que me conduziu no desenvolvimento deste trabalho.

A todos os professores e funcionários do Instituto de Informática e à Universidade Federal do Rio Grande do Sul pela formação de excelência e de qualidade.

SUMÁRIO

LISTA DE ABREVIATURAS E SIGLAS

3FN	Terceira Forma Normal
BI	Business Intelligence
CAPES	Coordenação de Aperfeiçoamento de Pessoal de Nível Superior
DW/BI	Data Warehousing e Business Intelligence
ETL	Extract, Transform, and Load
OLAP	On-Line Analytical Processing
PPGC	Programa de Pós-Graduação em Computação
SCD	Slowly Changing Dimension
SGBD	Sistema de Gerenciamento de Banco de Dados
SNPG	Sistema Nacional de Pós-Graduação

LISTA DE FIGURAS

RESUMO

O objetivo deste trabalho é criar uma solução para auxiliar na análise da produção acadêmica do Programa de Pós-Graduação em Computação (PPGC) do Instituto de Informática da UFRGS. A solução toma como base o conceito de modelagem dimensional no desenvolvimento de um data warehouse para armazenar os dados de produção acadêmica e utiliza uma ferramenta OLAP para a análise dos dados. Esses dados têm origem no Aplicativo Coleta de Dados CAPES, cujo objetivo é coletar informações dos programas de pós-graduação no Brasil. A solução desenvolvida neste trabalho permite visualizar, de forma flexível e dinâmica, o desempenho da produção do PPGC ao longo do tempo, auxiliando nas decisões gerenciais relativas às pesquisas acadêmicas realizadas no programa.

Palavras-chave: Data warehouse, OLAP, modelagem dimensional, PPGC, CAPES.

Development of Data Warehouse and OLAP Tool for the Analysis of Researchers' Academic Production: a Case Study in PPGC

ABSTRACT

The goal of this work is to create a solution to assist in analyzing the academic production of the Graduate Program in Computer Science (PPGC) of the Institute of Informatics, UFRGS. The solution builds on the concept of dimensional modeling in the development of a data warehouse to store academic production data and uses an OLAP tool for data analysis. These data come from the CAPES Data Collection Application, whose goal is to collect information from graduate programs in Brazil. The solution developed in this work allows the visualization, in a flexible and dynamic way, of the production performance of PPGC over time, assisting in management decisions concerning academic research conducted in the program.

Keywords: data warehouse, OLAP, dimensional modeling, PPGC, CAPES.

1 INTRODUÇÃO

O Programa de Pós-Graduação em Computação (PPGC) do Instituto de Informática da UFRGS é um dos mais antigos programas brasileiros de pós-graduação na área. Com atuação em diversas sub-áreas da Computação, consolidou-se ao longo de mais de três décadas, como centro de excelência em ensino e pesquisa. O objetivo do programa é a formação de pesquisadores e profissionais qualificados para desenvolver atividades em empresas de alta tecnologia através dos programas de Mestrado em Ciência da Computação, existente desde 1973, e Doutorado em Ciência da Computação, existente desde 1989 (PPGC, 2013).

Os programas são fortemente integrados com atividades de pesquisa. Os resultados dessas pesquisas se traduzem na publicação de artigos científicos em veículos de grande impacto e no desenvolvimento de produtos e processos de empresas geradoras de tecnologia (PPGC, 2013).

O PPGC prioriza os estudantes com dedicação integral. Todos os estudantes com dedicação integral recebem bolsas de estudo de agências brasileiras. Desde 1973, o PPGC graduou mais de 1100 mestres e 150 doutores. Atualmente é um dos cinco programas brasileiros classificados como de classe internacional pelo Ministério da Educação do Brasil (INSTITUTO DE INFORMÁTICA DA UFRGS, 2013).

O reconhecimento do Ministério da Educação do Brasil é resultado da Avaliação dos Programas de Pós-graduação, que compreende a realização do acompanhamento anual e da avaliação trienal do desempenho de todos os programas e cursos que integram o Sistema Nacional de Pós-graduação, SNPG. A Avaliação dos Programas de Pós-Graduação é um dos processos que compõem o Sistema de Avaliação, cuja implantação foi realizada pela CAPES, Coordenação de Aperfeiçoamento de Pessoal de Nível Superior, em 1976 (FUNDAÇÃO CAPES, 2013a).

A CAPES é uma fundação do Ministério da Educação e desempenha papel fundamental na expansão e consolidação da pós-graduação stricto sensu (mestrado e doutorado) em todos os estados da Federação (FUNDAÇÃO CAPES, 2013b).

A avaliação é feita através do Aplicativo Coleta de Dados CAPES [1], que tem o objetivo de coletar informações dos cursos de mestrado, doutorado e mestrado profissional integrantes do Sistema Nacional de Pós-Graduação. A coleta de dados objetiva ainda prover à CAPES informações necessárias ao planejamento dos seus programas de fomento e delineamento de suas políticas institucionais (FUNDAÇÃO CAPES, 2013c).

O Aplicativo Coleta de Dados CAPES possui uma interface gráfica de usuário para cadastro das informações do Programa de Pós-Graduação da instituição de ensino superior, incluindo a produção acadêmica entre outras informações. As informações cadastradas

[1] Disponível em <http://www.capes.gov.br/avaliacao/coleta-de-dados>.

são armazenadas no banco de dados interno da aplicação. Os dados dos programas são inicialmente transferidos para as suas respectivas Reitorias, pró-reitoria de pós-graduação ou órgão correspondente. Essa transferência é feita por meio físico (pendrive) ou por via eletrônica (rede). A transferência de dados da instituição de ensino superior para a CAPES utiliza a Internet, exclusivamente, por meio do sistema CAPESNET. Mais detalhes do funcionamento do sistema de coleta podem ser encontrados em (FUNDAÇÃO CAPES, 2013c).

A análise da produção acadêmica do PPGC, por parte do Instituto de Informática, é importante para visualizar o desempenho do programa ao longo do tempo e compreender quais os fatores que justificam a quantidade e a qualidade das produções. Essa análise serve de apoio para as decisões gerenciais relativas às pesquisas acadêmicas realizadas no programa, auxiliando nas iniciativas que incentivem o aumento da produção e/ou da qualidade onde há necessidade e na manutenção destas onde o desempenho é satisfatório.

O objetivo deste trabalho é criar uma solução para auxiliar na análise da produção acadêmica do PPGC. A ideia é que essa solução permita a maior flexibilidade possível na análise dos dados, possibilitando a visualização desses dados de diferentes perspectivas e de forma dinâmica, ou seja, sem a necessidade de reestruturação dos dados ou de reprojeto da solução.

Para atingir esse objetivo, foi desenvolvido um data warehouse, um repositório de dados, para armazenar os dados da produção acadêmica do PPGC. Esse repositório também tem como propósito estruturar os dados de forma a facilitar a análise. Uma ferramenta de código-fonte aberto é utilizada para a análise propriamente dita. Ela fornece os instrumentos necessários para a visualização dos dados e criação de relatórios. A origem dos dados do data warehouse é do sistema da CAPES. É necessário que um sistema seja utilizado para popular o data warehouse e mantê-lo com os dados atualizados. O desenvolvimento desse sistema não é abordado neste trabalho, e é sugerido como trabalho futuro.

Este trabalho está organizado da seguinte forma: o segundo capítulo descreve as tecnologias que foram utilizadas, servindo de referencial teórico para a solução desenvolvida. O terceiro capítulo apresenta o projeto da solução desenvolvida, listando os requisitos e mostrando os modelos, diagramas e documentos que auxiliaram no desenvolvimento do data warehouse. O quarto capítulo mostra o desenvolvimento da solução, apresentando a ferramenta utilizada e as configurações necessárias. O quinto capítulo apresenta a solução pronta em funcionamento, mostrando exemplos de uso das ferramentas na análise da produção acadêmica. Por fim, o último capítulo apresenta a conclusão do trabalho, discutindo o que foi efetivamente feito, as limitações e restrições da solução desenvolvida e o que poderá ser realizado futuramente.

2 DATA WAREHOUSE E BUSINESS INTELLIGENCE

Data warehouse é um repositório de dados que possui uma cópia dos dados transacionais, oriundos dos sistemas operacionais, estruturados especificamente para consultas e análises (KIMBALL, 1998). Ele forma a infraestrutura de back-end de uma grande variedade de sistemas de usuário com a função de fornecer compreensão e ação no que diz respeito às decisões de gestão; os data warehouses frequentemente possuem um volume substancial de histórico de dados operacionais de uma organização (EVELSON; NICOLSON, 2008).

O termo Business Intelligence (BI) é um termo popular e genérico que foi promovido por Howard Dresner do Gartner Group em 1989. Ele descreve um conjunto de conceitos e métodos para aprimorar decisões de negócio pelo uso de sistemas de apoio baseados em fatos. BI é por vezes usado como sinônimo de ferramentas de geração de relatórios e consultas e sistemas de informação executivos. Em geral, sistemas de BI são sistemas de apoio à decisão orientados a dados (POWER, 2013).

Thomas Davenport, em (HENSCHEN; DAVENPORT, 2013), dá uma definição um pouco mais precisa das atividades que fazem parte do conceito de BI. Ele argumenta que BI deve ser dividida em consultas, relatórios, OLAP (On-Line Analytical Processing), uma ferramenta de "alertas" e Business Analytics. Nessa definição, Business Analytics é um subconjunto de BI baseado em estatística, previsão e otimização.

Em geral, aplicações de BI utilizam os dados obtidos de um data warehouse. Entretanto, nem todo data warehouse é usado para business intelligence e nem toda aplicação de business intelligence precisa de um data warehouse (WIKIPEDIA, 2013).

2.1 On-Line Analytical Processing (OLAP)

Um data warehouse armazena informações que respondem às perguntas "quem?" e "o quê?" sobre eventos passados. Uma típica consulta submetida a um data warehouse pode ser: "qual a receita total para a região leste no terceiro trimestre?" (OLAP Council, 1997).

Em contraste com um data warehouse, que é geralmente baseado em tecnologia relacional, OLAP usa uma visão multidimensional de agregação de dados para fornecer acesso rápido a informações estratégicas para análise posterior. (OLAP Council, 1997).

OLAP permite compreender os dados através de acesso rápido, consistente e interativo a uma ampla variedade de visões possíveis da informação. OLAP transforma dados brutos para que eles reflitam a dimensionalidade real do empreendimento como é entendido pelo usuário (OLAP Council, 1997).

Enquanto os sistemas OLAP têm a capacidade de responder às questões "quem?" e "o quê?", eles também têm a capacidade de responder às questões "e se?" e "por quê?", o que

os diferencia dos data warehouses. OLAP proporciona a tomada de decisão sobre ações futuras. Um cálculo OLAP típico é mais complexo que simplesmente a soma de dados, por exemplo: “qual seria o efeito sobre os custos do refrigerante para os distribuidores se os preços do xarope subissem 10 centavos por litro e os custos de transporte baixassem 5 centavos por quilômetro?” (OLAP Council, 1997).

OLAP e data warehouses são complementares. Um data warehouse armazena e gerencia dados. OLAP transforma os dados do data warehouse em informação estratégica. OLAP abrange desde navegação básica e consultas (conhecidas como “slice and dice”) até cálculos e análises mais sérias como séries temporais e modelagem complexa. No momento em que os usuários utilizam os recursos mais avançados de OLAP, eles passam do simples acesso a dados em busca de informação para o processo mais avançado de obtenção de conhecimento (OLAP Council, 1997).

2.2 Captura de Dados e Análise de Dados

Em uma organização, a informação é um ativo quase sempre utilizado de duas maneiras diferentes: para manter o registro operacional e para tomada de decisões gerenciais. De forma simplificada, os dados são capturados nos sistemas operacionais (sistemas de informação com registro de operações e transações do negócio) e são obtidos de volta nos sistemas de apoio à decisão, mais conhecidos hoje como sistemas de data warehousing e business intelligence (DW/BI) (KIMBALL; ROSS, 2013).

Os usuários dos sistemas operacionais movem as engrenagens da organização. Eles realizam as mesmas tarefas operacionais todos os dias, executando os processos de negócio da organização. O foco dessa execução é manter os dados atualizados para refletir seu estado mais atual, sem a preocupação de manter um histórico desses dados, pelo menos a longo prazo (KIMBALL; ROSS, 2013).

Os usuários dos sistemas de DW/BI, por outro lado, veem as engrenagens da organização se movendo para avaliar sua performance. Eles estão preocupados se os processos operacionais estão funcionando corretamente. Embora eles precisem de dados detalhados para suportar suas questões que estão em constante mudança, os usuários de sistemas de DW/BI praticamente nunca lidam com uma transação por vez. Esses sistemas são otimizados para consultas de alta performance dado que as questões dos usuários frequentemente requerem centenas ou centenas de milhares de transações sendo consultadas e comprimidas em um conjunto de resposta. Ainda, os usuários dos sistemas de DW/BI demandam que o contexto histórico seja preservado para avaliar de forma precisa a performance da organização ao longo do tempo (KIMBALL; ROSS, 2013).

Em (KIMBALL, 1998) há uma descrição detalhada das diferenças entre os mundos do processamento operacional e do data warehousing. Hoje, já há um amplo reconhecimento de que os sistemas de DW/BI possuem, profundamente, diferentes necessidades, clientes, estruturas e ritmo diferente de uso dos registros em comparação com os sistemas operacionais. Portanto, também se reconhece hoje que esses dois sistemas devem ser desenvolvidos e mantidos separadamente, e com técnicas de desenvolvimento diferentes (KIMBALL; ROSS, 2013).

2.3 Modelagem Dimensional

Modelagem dimensional é uma técnica amplamente aceita como a preferida para apresentar dados analíticos, porque ela aborda dois requisitos simultâneos:

- fornecer dados que são compreensíveis para os usuários de negócio;
- proporcionar rápido desempenho de consulta.

A modelagem dimensional é uma técnica antiga para construção de bancos de dados simples. A simplicidade é crítica pois garante que os usuários possam facilmente compreender os dados, bem como permitir ao software navegar e fornecer resultados rapidamente e com eficiência (KIMBALL; ROSS, 2013).

Na modelagem dimensional os dados podem ser visualizados como se eles estivessem dispostos em um cubo. Um exemplo ajuda na compreensão. Suponha-se que um executivo descreva as atividades da empresa da seguinte forma: "Vendemos produtos em vários mercados e avaliamos nosso desempenho ao longo do tempo". Os projetistas dimensionais ouvem a descrição cuidadosamente e dão uma ênfase especial às palavras produto, mercado e tempo. Eles, então, imaginam o negócio da empresa como um cubo de dados, cujas arestas são rotuladas com as palavras enfatizadas (Figura 2.1). Essas são as dimensões do cubo. Os pontos dentro do cubo são onde as medições, como volume de vendas e lucro, para aquela combinação de produto, mercado e tempo estão armazenadas. Esse é o modelo dimensional (KIMBALL, 1998; KIMBALL; ROSS, 2013).

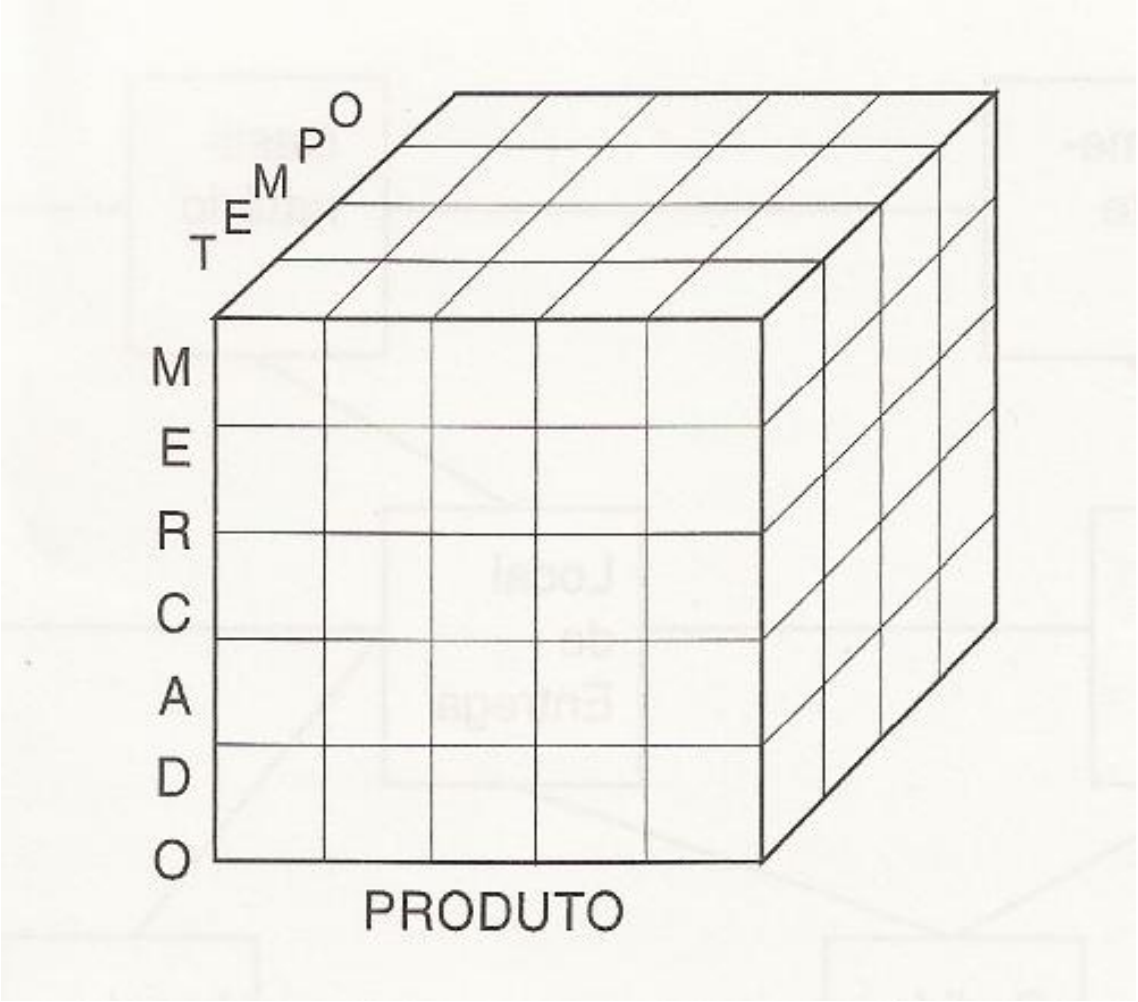


Figura 2.1: Exemplo de modelo dimensional: cada ponto interno ao cubo contém as medições para aquela combinação de Produto, Mercado e Tempo (KIMBALL, 1998)

A capacidade de visualização de algo abstrato como um conjunto de dados em uma maneira concreta e tangível é o segredo para a compreensibilidade. Um modelo de dados simples no princípio tem a chance de permanecer simples até o final do projeto. Um modelo complicado de início certamente será complicado no final, resultando em um lento desempenho de consulta e rejeição por parte dos usuários de negócio (KIMBALL; ROSS, 2013).

Apesar dos modelos dimensionais frequentemente serem instanciados em sistemas de gerenciamento de bancos de dados relacionais, eles são completamente diferentes dos modelos na terceira forma normal (3FN), os quais buscam remover a redundância de dados. Estruturas normalizadas na 3FN dividem os dados em várias entidades discretas, e cada uma se torna uma tabela relacional. Um banco de dados pode começar simples, podendo ter apenas uma tabela que guarda um registro em cada linha. Mas, no desenvolvimento com o modelo 3FN, pode acabar terminando com centenas de tabelas normalizadas (KIMBALL; ROSS, 2013).

A principal diferença entre o modelo 3FN e o modelo dimensional é o grau de normalização dos dados. Estruturas normalizadas na 3FN são extremamente úteis no processamento operacional porque uma transação de alteração ou inserção acessa o banco de dados em um único lugar. Modelos normalizados, por outro lado, são muito complicados para as consultas de BI. Os usuários não conseguem entender, navegar ou relembrar modelos normalizados. Além disso, a maioria dos sistemas de gerenciamento de bancos de dados relacionais não conseguem consultar um modelo normalizado de forma eficiente; a complexidade das consultas imprevisíveis dos usuários supera os otimizadores do banco de dados, resultando em um desastroso desempenho de consulta. Portanto, o uso de modelos normalizados em DW/BI anula a recuperação de dados intuitiva e de alta performance. Felizmente, a modelagem dimensional soluciona o problema de esquemas muito complexos na área de apresentação dos dados (KIMBALL; ROSS, 2013).

2.3.1 Esquema Estrela Versus Cubos OLAP

Os modelos dimensionais que são implementados em bancos de dados relacionais são conhecidos como esquemas estrela pois sua estruturas lembram o formato de uma estrela. Porém, os modelos dimensionais também podem ser implementados em ambientes de banco de dados multidimensionais. Bancos de dados multidimensionais são um tipo de banco de dados que são otimizados para data warehouse e aplicações OLAP. Conceitualmente, um banco de dados multidimensional usa a ideia de um cubo de dados para representar as dimensões de dados disponíveis para um usuário (ROUSE, 2005). Modelos dimensionais implementados nesse tipo de banco de dados são conhecidos como cubos OLAP. A Figura 2.2 ilustra a diferença entre o esquema estrela e o cubo OLAP.

Tanto o esquema estrela quanto o cubo OLAP possuem em comum o projeto lógico com dimensões. Entretanto, a implementação física é diferente. Os dados carregados em um cubo OLAP são armazenados e indexados usando formatos e técnicas que são projetadas para dados dimensionais. O mecanismo de cubo OLAP muitas vezes cria e gerencia agregações de performance e tabelas de resumo pré-calculadas. Consequentemente, os cubos fornecem uma performance de consulta superior, dados os cálculos antecipados, as estratégias de indexação e outras otimizações. Os usuários podem detalhar ou sumarizar as consultas, adicionando ou removendo atributos às suas análises com excelente performance sem precisar criar novas consultas. Outra vantagem dos cubos OLAP é que eles fornecem funções analíticas mais robustas que não estão disponíveis no SQL dos bancos de dados relacionais. O preço a ser pago por essas vantagens é o desempenho mais lento

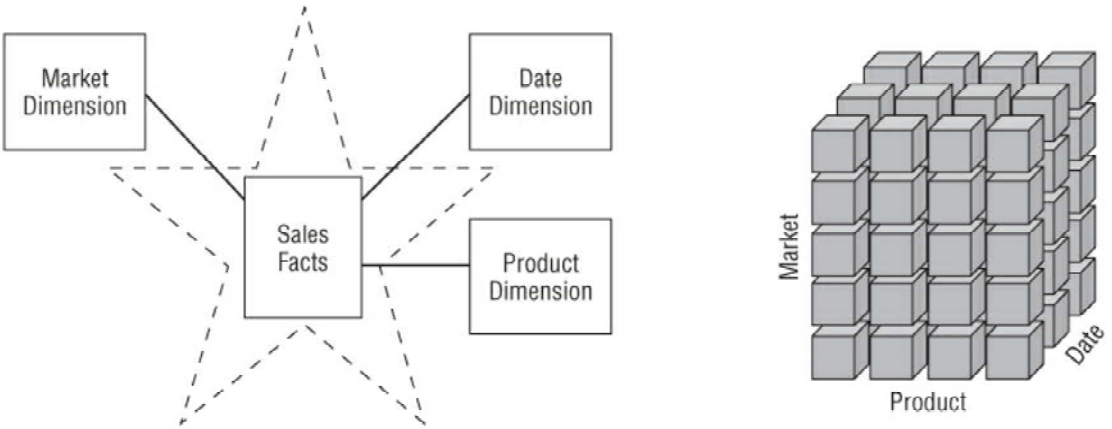


Figura 2.2: Esquema estrela versus cubo OLAP (KIMBALL; ROSS, 2013)

no carregamento, especialmente para conjuntos de dados muito grandes.

Apesar da tecnologia OLAP estar em constante melhoria, Kimball e Ross (2013) recomendam o uso do esquema estrela. Opcionalmente, os cubos OLAP podem ser populados a partir de esquemas em estrela. No desenvolvimento deste trabalho seguiu-se a recomendação feita por (KIMBALL; ROSS, 2013), realizando-se a modelagem dimensional com um esquema estrela. Mais adiante, porém, um cubo OLAP é criado a partir do esquema estrela desenvolvido. Esse processo é mostrado no capítulo sobre o desenvolvimento da solução.

2.3.2 Tabelas de Fatos e de Dimensão

Um modelo dimensional implementado no esquema estrela é composto de tabelas de fatos e tabelas de dimensão. A tabela de fatos em um modelo dimensional armazena as medições de performance decorrentes de eventos de um processo de negócio de uma organização. O termo “fato” representa uma medição de negócio. A quantidade unitária e o valor em dinheiro de cada produto em uma transação de venda de um supermercado são exemplos de medições de negócio. Cada linha em uma tabela de fatos corresponde a um evento de medição. Os dados em cada linha estão em um determinado nível de detalhe, denominado como a “granularidade”, tal como uma linha por produto vendido em uma transação de venda. Todas as linhas de medição em uma tabela de fatos devem ter a mesma granularidade. Isso garante que as medições não sejam contadas mais de uma vez de forma inapropriada (KIMBALL; ROSS, 2013).

A tabelas de fatos também possuem chaves estrangeiras para se conectar com as chaves primárias das tabelas de dimensão. Uma chave estrangeira de produto na tabela de fatos, por exemplo, sempre corresponde a um produto específico na tabela de dimensão de produto (com informações detalhadas de um produto). A tabela de fatos é acessada pelas tabelas de dimensão relacionadas a ela. Em geral, uma tabela de fatos tem como chave primária a composição de um subconjunto de chaves estrangeiras. Muitas vezes ela é chamada de “chave composta” (KIMBALL; ROSS, 2013).

As tabelas de dimensão contém o contexto textual associado com uma medição de evento de processo de negócio. Elas descrevem “quem, o quê, onde, quando, como e o porquê” associado com o evento. As tabelas de dimensão geralmente possuem muitas colunas ou atributos. Elas tendem a ter menos linhas que as tabelas de fatos, mas podem ser mais “largas” com muitas colunas grandes de texto. Cada dimensão é definida por uma

única chave primária, que serve como base para a integridade referencial com qualquer tabela de fatos com a qual ela é relacionada (KIMBALL; ROSS, 2013).

Os atributos de dimensão servem como a fonte primária de restrições de consulta, agrupamentos e rótulos de relatório. Em uma consulta ou requisição de relatório, os atributos são identificados pelas palavras "por". Por exemplo, quando um usuário quer ver o volume de vendas por marca, "marca" deve estar disponível como um atributo de dimensão (KIMBALL; ROSS, 2013).

Os atributos das tabelas de dimensão desempenham um papel vital no sistema de DW/BI. Por serem a origem de virtualmente todas as restrições e rótulos de relatórios, os atributos de dimensão são fundamentais para tornar o sistema de DW/BI utilizável e compreensível. Assim, recomenda-se que os atributos devem consistir em palavras reais em vez de abreviações e códigos. De muitas maneiras, o data warehouse é tão bom quanto os atributos de dimensão. O poder de análise de um ambiente de DW/BI é diretamente proporcional à qualidade e profundidade dos atributos de dimensão. Atributos de dimensão robustos fornecem robustas habilidades de análise (KIMBALL; ROSS, 2013).

As tabelas de dimensão frequentemente representam relacionamentos hierárquicos. Por exemplo, em uma dimensão com informações de produtos, um produto pode ser classificado em marcas que, por sua vez, podem ser classificadas por categorias. Cada linha na dimensão produto deve armazenar a descrição da marca e da categoria do produto. A informação da hierarquia de descrições é armazenada, portanto, de forma redundante no espírito de facilidade de uso e performance de consulta. A normalização das tabelas de dimensão geralmente não causam nenhum impacto substancial no tamanho total do banco de dados, visto que as tabelas de dimensão ocupam muito pouco espaço em relação às tabelas de fatos. Essa normalização recebe o nome de "snowflaking", por causa da estrutura do esquema de tabelas normalizadas ficar parecida com um floco de neve. Recomenda-se, portanto, que se troque o espaço das tabelas de dimensão pela simplicidade e facilidade de acesso proporcionadas pelo modelo não normalizado (KIMBALL; ROSS, 2013).

2.3.3 Uso do Modelo Dimensional

Um exemplo de modelo dimensional completo, com uma tabela de fatos e as tabelas de dimensão relacionadas, pode ser visto na Figura 2.3. Os atributos das dimensões foram omitidos por questões de espaço. Cada processo de negócio é representado por um modelo dimensional que consiste em uma tabela de fatos, contendo as medições numéricas de um evento, cercada de tabelas de dimensão contendo o contexto textual do momento em que o evento ocorreu.

É possível notar a simplicidade e simetria do esquema dimensional. A simplicidade facilita na navegação e compreensão dos dados por parte dos usuários de negócio. O número reduzido de tabelas assim como as descrições textuais que fazem sentido para o negócio possibilitam a facilidade na navegação e restringem a possibilidade de cometer erros. Essa simplicidade também traz benefícios de performance. O número reduzido de relacionamentos faz com que os otimizadores do banco de dados processem o esquema com mais eficiência (KIMBALL; ROSS, 2013).

Os modelos dimensionais são também extensíveis para acomodar mudanças. A estrutura previsível de um modelo dimensional resiste a mudanças inesperadas no comportamento do usuário. Todas as dimensões são equivalentes entre si. Todas as dimensões são pontos de entrada simetricamente iguais para a tabela de fatos. O modelo não tem nenhuma inclinação preconcebida a um padrão de consulta esperado. Seria inviável ajustar os esquemas toda vez que os usuários sugerissem novas formas de analisar o seu negócio

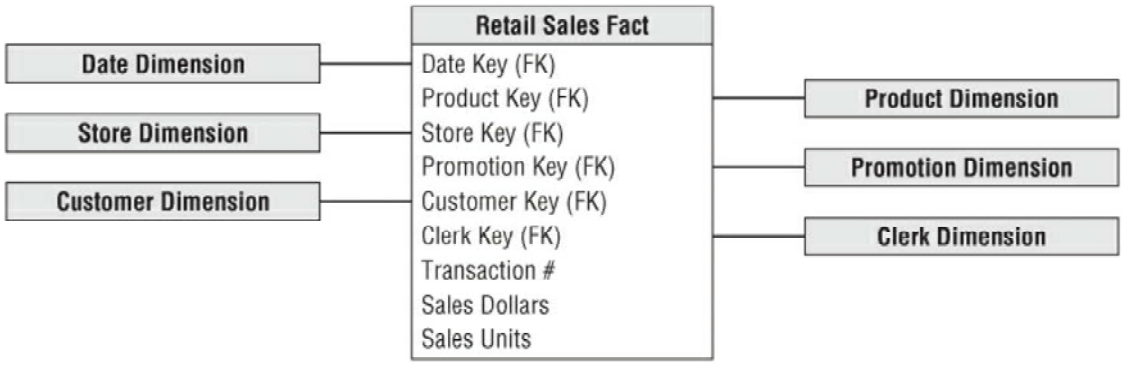


Figura 2.3: Tabela de fatos e tabelas de dimensão em um modelo dimensional (KIMBALL; ROSS, 2013)

(KIMBALL; ROSS, 2013).

A seguir é mostrado um exemplo de uso do modelo dimensional em um relatório (Figura 2.4). Os atributos dimensionais servem como filtros e fornecem os rótulos do relatório, enquanto que a tabela de fatos fornece os valores numéricos.

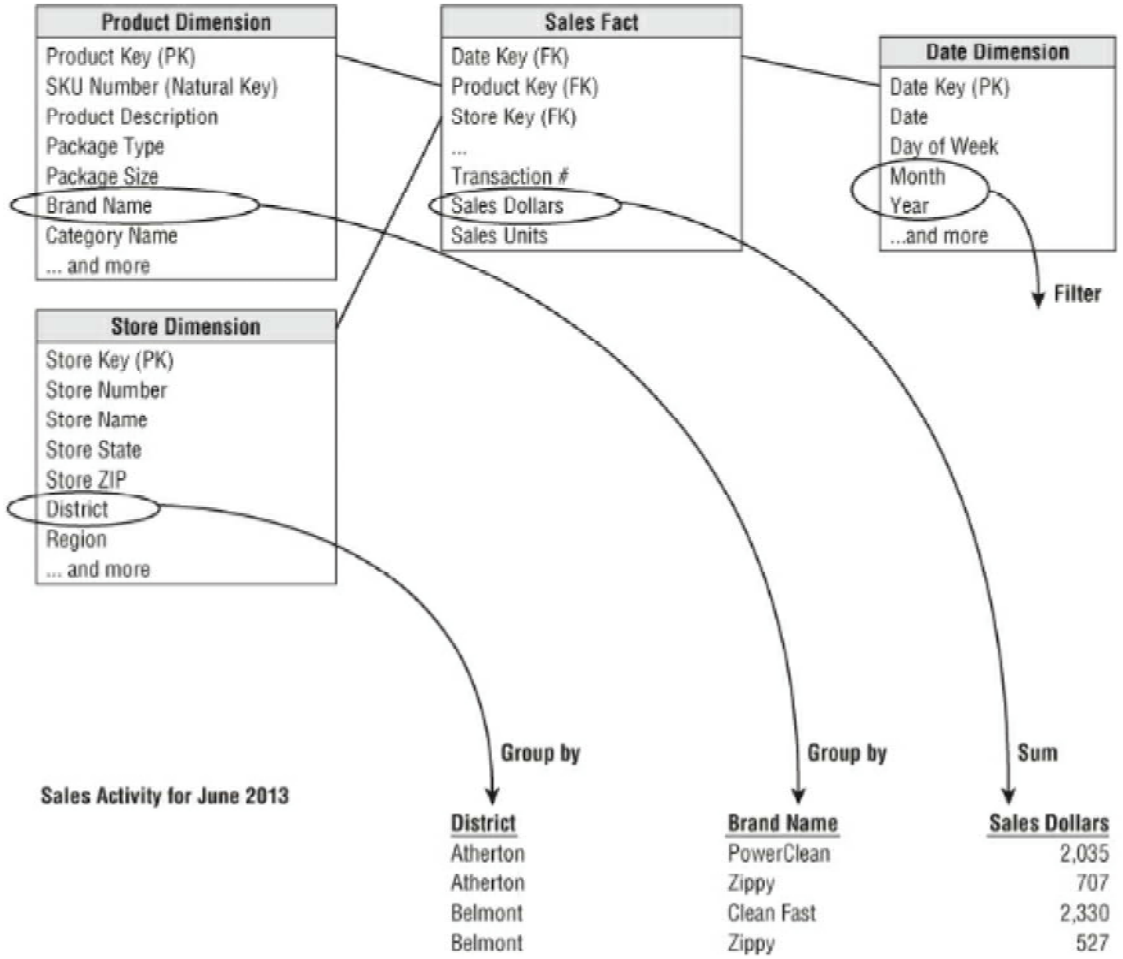


Figura 2.4: Papel dos atributos dimensionais e dos fatos em uma consulta (KIMBALL; ROSS, 2013)

A consulta em SQL que cria esse relatório pode ser facilmente deduzida (ou gerada

por uma ferramenta de BI):

```
SELECT
  store.district_name,
  product.brand,
  sum(sales_facts.sales_dollars) AS "Sales Dollars"
FROM
  store,
  product,
  date,
  sales_facts
WHERE
  date.month_name = "January" AND
  date.year = 2013 AND
  store.store_key = sales_facts.store_key AND
  product.product_key = sales_facts.product_key AND
  date.date_key = sales_facts.date_key
GROUP BY
  store.district_name,
  product.brand
```

2.4 Arquitetura de um Sistema de DW/BI

Nesta seção será vista a arquitetura de um sistema de DW/BI. Essa arquitetura foi proposta por Kimball e Ross (2013) e foi essa visão de arquitetura que foi utilizada neste trabalho.

A Figura 2.5 mostra que um ambiente de DW/BI é composto de quatro componentes: sistemas operacionais de origem, sistema ETL, área de apresentação de dados e aplicações de business intelligence.

2.4.1 Sistemas Operacionais de Origem

Os sistemas operacionais de origem capturam as transações de negócio. Eles podem ser pensados como componentes que não fazem parte do ambiente de DW/BI, pois há pouco ou nenhum controle sobre o conteúdo e formato dos dados nesses sistemas. As principais prioridades dos sistemas de origem são performance de processamento e disponibilidade. Em muitos casos, os sistemas de origem são aplicações de propósito específico sem nenhum compromisso de compartilhar dados comuns com outros sistemas operacionais da organização (KIMBALL; ROSS, 2013).

2.4.2 Sistema ETL

O sistema ETL (Extract, Transform, and Load) consiste em uma área de trabalho, estruturas de dados e um conjunto de processos. O sistema ETL é tudo o que está entre os sistemas operacionais de origem e a área de apresentação de dados (KIMBALL; ROSS, 2013).

O primeiro passo é a extração dos dados. Extração significa leitura e entendimento dos dados de origem e cópia dos dados necessários para dentro do sistema ETL para posterior manipulação. A partir de então, os dados pertencem ao data warehouse (KIMBALL; ROSS, 2013).

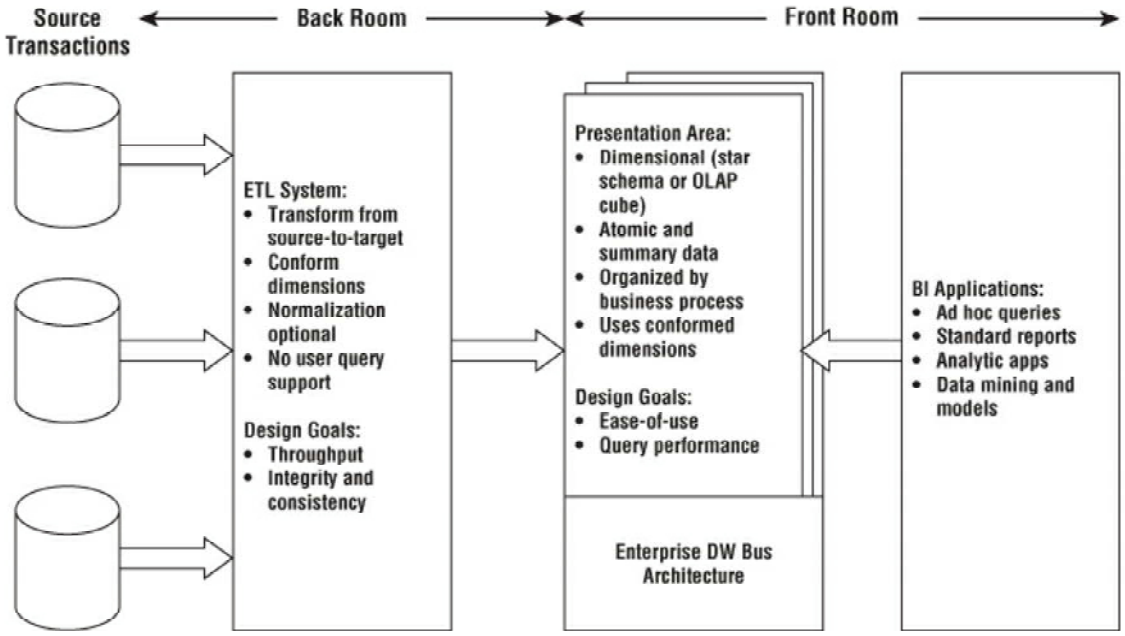


Figura 2.5: Arquitetura de um ambiente de DW/BI (KIMBALL; ROSS, 2013)

Depois que os dados são extraídos para o sistema ETL, existem muitas transformações em potencial, como limpeza de dados (correção de erros ortográficos, resolução de conflitos de domínio, tratamento de elementos ausentes ou conversão para formatos padrão), combinação de dados de múltiplas fontes e de-duplicação de dados. O sistema ETL adiciona valor aos dados com essas tarefas de limpeza e adaptação, alterando-os e os melhorando (KIMBALL; ROSS, 2013).

O passo final do processo de ETL é a estruturação física e carregamento de dados nos modelos dimensionais de destino da área de apresentação. Os subsistemas desse passo são fundamentais, pois a missão principal do sistema ETL é entregar os dados para as tabelas de fatos e de dimensão. Muitos desses subsistemas focam em processamento de tabelas de dimensão, como atribuições de chaves substitutas, pesquisa de códigos para fornecer as descrições apropriadas, divisão ou combinação de colunas para apresentar valores de dados apropriados, ou junção de estruturas de tabelas na terceira forma normal para criar dimensões não normalizadas. Em contraste, tabelas de fatos são tipicamente grandes e consomem muito tempo para serem carregadas, mas prepará-las para a área de apresentação é geralmente simples. Quando todo o processo de ETL é encerrado, os usuários de negócio são notificados de que novos dados foram publicados (KIMBALL; ROSS, 2013).

2.4.3 Área de Apresentação de Dados (Data Warehouse)

A área de apresentação (data warehouse) é onde os dados são organizados, armazenados e disponibilizados para as consultas diretas dos usuários, geradores de relatórios e outras aplicações de BI. A área de apresentação é tudo o que os usuários de negócio podem ver e manusear com suas ferramentas de acesso e aplicações de BI (KIMBALL; ROSS, 2013).

Na área de apresentação, Kimball e Ross (2013) recomendam que os dados devem ser apresentados, armazenados e acessados em modelos dimensionais, seja com esquemas estrela relacionais ou cubos OLAP.

Outra recomendação feita em (KIMBALL; ROSS, 2013) é que a área de apresentação deve conter os dados atômicos, na forma mais detalhada possível. Os dados atômicos são necessários para suportar as imprevisíveis consultas específicas feitas pelos usuários. Embora a área de apresentação também possa conter dados agregados para melhoria de performance, não é o suficiente fornecer esses dados resumidos sem os dados granulares em uma forma dimensional. Os dados com maior granularidade devem estar disponíveis na área de apresentação de forma que os usuários possam fazer as questões mais precisas possíveis.

Os dados da área de apresentação devem estar estruturados em torno de processos de negócio. Os dados não devem ser estruturados de acordo com a interpretação de cada departamento de uma organização. Em outras palavras, uma única tabela de fatos deve ser construída para as métricas atômicas de vendas, por exemplo, em vez de popular bancos de dados separados, com dados similares mas com pequenas diferenças, para o departamento de vendas, para o de marketing, para o de logística e para o departamento de finanças (KIMBALL; ROSS, 2013).

2.4.4 Aplicações de Business Intelligence

As aplicações de Business Intelligence (BI) formam o último componente da arquitetura. O termo aplicação de BI, de acordo com (KIMBALL; ROSS, 2013), se refere à gama de funcionalidades fornecidas aos usuários de negócio na utilização da área de apresentação na tomada de decisão analítica. Por definição, todas as aplicações de BI consultam dados na área de apresentação.

Uma aplicação de BI pode ser simples como uma ferramenta de consultas específicas ou complexa como uma aplicação de mineração de dados ou de modelagem (KIMBALL; ROSS, 2013).

2.5 Resumo

Neste capítulo foram apresentados os conceitos de data warehouse e business intelligence. Como foi descrito, data warehouse é um repositório de dados estruturado especificamente para o processo de análise de dados. Complementando o data warehouse, está o conceito de OLAP, que permite compreender os dados do data warehouse através do acesso rápido, consistente e interativo a uma ampla variedade de visões possíveis da informação. Viu-se que o termo business intelligence abrange um conjunto de conceitos e métodos para aprimorar decisões de negócio por meio do uso de sistemas de apoio baseados em fatos. Esse conjunto de conceitos e métodos inclui os conceitos de data warehouse e OLAP.

A técnica de modelagem dimensional, usada na construção de data warehouses, também foi descrita. Dois tipos de estrutura possíveis na implementação dessa técnica foram apresentadas, o esquema estrela e o cubo OLAP. Discutiu-se as diferenças entre essas estruturas e uma ênfase maior foi dada ao esquema estrela, por este ser o mais recomendado pela literatura de referência. As tabelas de fatos e de dimensão, componentes do esquema estrela, foram descritas e um exemplo de uso da estrutura final foi apresentado, explicando o seu funcionamento.

Por fim, a arquitetura de um sistema de DW/BI foi descrita, apresentando-se seus principais componentes e suas funções. A arquitetura apresentada é baseada também na literatura usada como referência.

3 PROJETO DA SOLUÇÃO

Este capítulo apresenta o projeto da solução de DW/BI para a análise da produção acadêmica. O projeto descreve a concepção do modelo dimensional do data warehouse, de acordo com os requisitos funcionais. O modelo dimensional é projetado seguindo a estrutura do esquema estrela, descrita no capítulo anterior. O esquema estrela resultante do projeto será depois mapeado para um cubo OLAP na configuração da ferramenta de BI utilizada na solução, Pentaho BI Platform. Essa configuração é descrita no capítulo seguinte.

O capítulo começa com a descrição dos requisitos funcionais e não-funcionais. Entre esses requisitos também foram incluídos alguns exemplos de consulta, que facilitam o entendimento de quais informações serão necessárias. Em seguida, são detalhados os passos da modelagem dimensional que foram realizados. A modelagem dimensional começa com um modelo de alto nível, que vai sendo refinado aos poucos até um modelo mais detalhado, que lista todas as dimensões, fatos e atributos. É produzida, ainda, uma documentação descrevendo cada tabela do modelo e seus campos, com tipos de dados, exemplos de valores, origem dos dados, regras de ETL, entre outras informações.

3.1 Requisitos

O objetivo do sistema é analisar a produção acadêmica do Programa de Pós-Graduação em Computação do Instituto de Informática da UFRGS. Os requisitos não-funcionais são os seguintes:

- a solução deve possuir código-fonte aberto e ser gratuita;
- possuir uma interface Web para facilitar o acesso.

Os requisitos funcionais são:

- permitir o gerenciamento dos usuários que podem acessar a ferramenta;
- permitir a inclusão de novos dados anualmente;
- importar os dados do Aplicativo Coleta de Dados CAPES;
- permitir a análise dos dados dinamicamente;
- permitir a geração de relatórios.

3.1.1 Consultas

Complementando os requisitos funcionais acima, também foram elaboradas algumas consultas que o sistema deve ser capaz de responder, tanto na análise quanto na geração de relatórios:

- Qual é a produção por autor?
- Qual é a produção por grupo de autores (docente permanente, docente colaborador, etc.)?
- Qual é a produção por linha de pesquisa?
- Qual é a produção de todo o programa por ano?

A palavra "produção" aqui se refere à soma de publicações acadêmicas (artigos, livros, teses, etc.) de acordo com a restrição pedida pela consulta (por autor, linha de pesquisa, etc.). Essas consultas servem como ponto de partida para se ter uma ideia do tipo de análise que será feita e para entender quais tipos de dados são importantes serem analisados. Com o projeto pronto do data warehouse, outras consultas, com outras restrições, e até mesmo uma composição dessas, poderão ser feitas.

Dados os requisitos, seguiu-se inicialmente com o projeto do data warehouse.

3.2 Modelagem Dimensional

De acordo com (KIMBALL; ROSS, 2013), é recomendado seguir quatro passos para a modelagem dimensional:

1. identificar o processo de negócio;
2. declarar a granularidade do processo de negócio;
3. identificar as dimensões;
4. identificar os fatos.

O processo de negócio é tipicamente determinado na conclusão da coleta de requisitos (KIMBALL; ROSS, 2013). Neste projeto, o processo de negócio pode ser identificado como a "produção acadêmica do Programa de Pós-Graduação em Computação". A seguir, é descrito o processo para os próximos passos.

3.2.1 Modelo de Alto Nível ("Bubble Chart")

A modelagem começa com a confecção de um modelo de alto nível, também conhecido como "bubble chart" (Figura 3.1).

Esse modelo identifica a granularidade da tabela de fatos e as dimensões associadas (KIMBALL; ROSS, 2013). No projeto em questão, a granularidade pode ser definida como uma linha da tabela de fatos para cada produção de um determinado autor. É importante notar que se uma produção possuir vários autores, assim como orientadores e co-orientadores, essa produção aparecerá várias vezes na tabela de fatos para cada autor/orientador/co-orientador associado com a produção.

As dimensões identificadas inicialmente foram: Data, Autor, Produção e Qualis. Qualis é o conjunto de procedimentos utilizados pela CAPES para estratificação da qualidade

Figura 3.1: Modelo de alto nível

da produção intelectual dos programas de pós-graduação. A estratificação da qualidade dessa produção é realizada de forma indireta, a partir da análise da qualidade dos veículos de divulgação, ou seja, periódicos científicos e conferências (FUNDAÇÃO CAPES, 2013d).

Novas dimensões podem surgir ao longo do desenvolvimento do modelo dimensional e do detalhamento dos atributos de cada dimensão, como será visto nas seções seguintes. As dimensões identificadas aqui foram as que surgiram naturalmente após a declaração da granularidade da tabela de fatos.

3.2.2 Detalhamento do Modelo Dimensional

No detalhamento do modelo dimensional foi discutido quais atributos deveriam ser incluídos em cada dimensão e quais métricas deveriam ser observadas na tabela de fatos. Descobriu-se também novas dimensões: Categoria Autor, Papel Autor e Ordem Autor, sendo as duas últimas dimensões degeneradas (*degenerate dimensions*, Kimball e Ross, 2013), dimensões que não precisam de uma tabela associada. Elas são necessárias para saber qual o papel do autor na produção (autor, orientador ou co-orientador) e qual a sua ordem na citação da obra. A dimensão Categoria Autor serve para identificar qual o vínculo do autor (docente, colaborador, discente, etc.) com o programa. Decidiu-se por uma dimensão separada da dimensão Autor para evitar redundância na sua respectiva tabela caso o vínculo do autor mudasse de um ano para o outro. O modelo dimensional detalhado pode ser visto na Figura 3.2.

No detalhamento da tabela de fatos descobriu-se que não há nenhuma métrica a ser medida, fazendo com que a tabela de fatos seja uma "*factless fact table*" (KIMBALL; ROSS, 2013), ou seja, uma tabela de fatos cuja função é simplesmente a de relacionar suas dimensões. Esse relacionamento entre as dimensões e a tabela de fatos permite contar o número de registros retornados, filtrando-os e os agrupando pelos atributos desejados nas tabelas das dimensões. Assim, a única métrica que existe na tabela de fatos é um contador de produção que será sempre "1" para cada linha dessa tabela.

É importante notar na Figura 3.2 que a dimensão Data se transformou na dimensão Ano, pois sabe-se que os dados no sistema de origem são coletados anualmente, não havendo nenhuma outra unidade de tempo. Assim, a dimensão Ano é também uma dimensão degenerada.

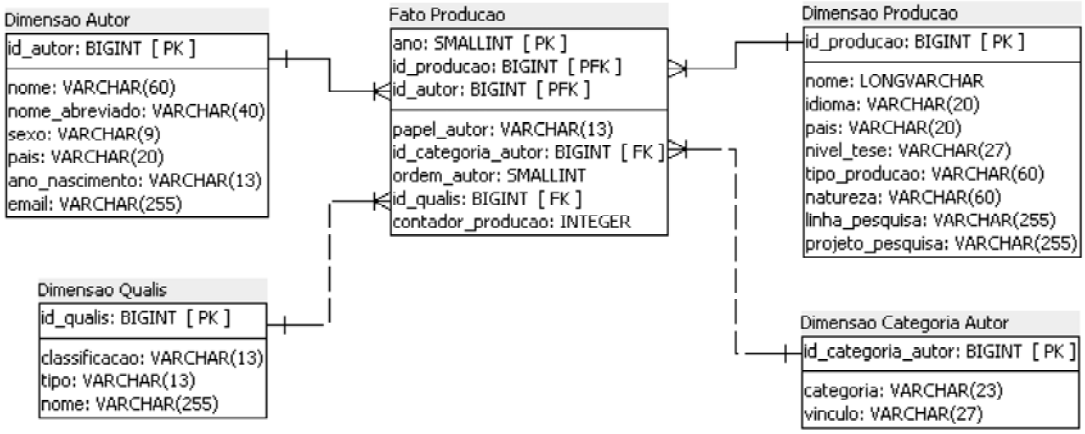


Figura 3.2: Modelo dimensional detalhado

3.2.3 Identificação das Técnicas de "Slowly Changing Dimension"

Técnicas de "*Slowly Changing Dimension*" (SCD) são estratégias bem definidas para lidar com mudanças nos atributos das dimensões ao longo do tempo. Apesar das informações contidas nas dimensões serem relativamente estáticas, elas provavelmente não serão fixas para sempre. Apesar das mudanças serem lentas, elas ainda existem. Por essa razão, Kimball e Ross (2013) desenvolveram e catalogaram diversos tipos de SCD e técnicas para lidar com cada um desses tipos.

Neste projeto foram identificados apenas os tipos 1 e 2 de SCD (KIMBALL; ROSS, 2013). No tipo 1, o valor de um atributo em uma linha da dimensão é simplesmente sobrescrito pelo valor atual. O atributo sempre reflete o valor atribuído mais recente. Não há aqui a preocupação com os diversos valores que um atributo da dimensão tenha recebido ao longo do tempo, ou seja, não há um histórico dos valores daquele atributo. Mais adiante será possível ver que, com exceção de apenas um atributo da dimensão Qualis, todos os outros atributos das dimensões foram identificados com SCD do tipo 1.

O tipo 2 de SCD consiste na inclusão de uma nova linha na tabela de dimensão para refletir o novo valor de um determinado atributo. Isso permite manter um histórico dos valores de um determinado atributo, visto que a linha com o valor antigo é mantida na tabela de dimensão. Para a implementação dessa técnica é necessário incluir campos administrativos de data de efetivação e data de expiração do valor do atributo para identificar sua validade. Assim é possível saber se o valor é atual ou se é um valor antigo.

O único atributo que foi identificado como sendo SCD do tipo 2 é o atributo "classificacao" da dimensão Qualis. Esse atributo informa qual a classificação Qualis de um determinado evento ou periódico. Sabe-se que a definição dessa classificação se altera ao longo dos anos e é interessante manter um histórico das classificações pelas definições antigas no data warehouse. Identificou-se, então, esse atributo como sendo SCD do tipo 2.

3.2.4 Documentação da Modelagem Dimensional

Com a identificação das técnicas de SCD é possível finalizar o projeto do modelo dimensional, produzindo a documentação das tabelas de dimensão e de fatos. Kimball e Ross (2013) sugerem criar uma planilha para cada tabela detalhando seus atributos e o mapeamento desses atributos com os sistemas de origem. A Figura 3.3 mostra a planilha

de uma das tabelas. A documentação completa se encontra no Apêndice B.

Nome da Tabela dim_autor
Nome Lógico Dimensao Autor
Tipo da Tabela Dimensão
Descrição Dimensão com informações de pessoas envolvidas nas produções acadêmicas

	Destino					Origem			
Coluna	**Descrição**	**Tipo de Dados**	**Tamanho**	**Exemplos**	**Tipo SCD**	**Tabela Origem**	**Campo Origem**	**Tipo do Campo Origem**	**Regras ETL**
id_autor	Chave primária	bigint		1, 2, 3, ...					Chave substituta
nome	Nome do autor	varchar	60	João da Silva, Maria da Graça, ...	1	COL_PESSOAL	Nome	varchar(60)	Cópia
nome_abreviado	Nome do autor abreviado para referência bibliográfica	varchar	40	SILVA, J.; GRAÇA, M.; ...	1	COL_PESSOAL	NomeAbreviado	varchar(40)	Cópia
sexo	Sexo do autor	varchar	9	Masculino, Feminino	1	COL_PESSOAL	Sexo	char(1)	M=Masculino, F=Feminino
pais	País de origem do autor	varchar	20	Não Informado, Brasil, Canadá, ...	1	COL_PESSOAL	Pais	char(2)	Mapeamento de uma tabela de códigos de países
ano_nascimento	Ano de nascimento do autor	varchar	13	Não Informado,1980, 1984, ...	1	COL_PESSOAL	AnoNascimento	smallint	NULL=Não informado, Não nulo=conversão entre tipos
email	E-mail do autor	varchar	255	joaodasilva@example.com, ...	1	COL_PESSOAL	Email	varchar(255)	Cópia

Figura 3.3: Planilha da tabela Dimensao Autor

Para cada atributo das tabelas há uma descrição, tipo de dados, exemplos de valores e o tipo SCD. É feito, também, um mapeamento dos atributos das tabelas do sistema de origem para o modelo dimensional, assim como são indicadas algumas regras para o sistema ETL.

3.3 Resumo

Neste capítulo foi descrito o projeto necessário para o desenvolvimento da solução. A partir dos requisitos e das consultas já elaboradas se construiu um modelo de alto nível do data warehouse. A partir deste, seguiu-se para o detalhamento do modelo dimensional, descobrindo-se quais são as dimensões e fatos relevantes, assim como os atributos que são importantes nas análises de dados na solução final. O resultado do projeto é a documentação detalhada de todas as tabelas que implementam o modelo dimensional. Essa documentação descreve os atributos das tabelas e seu mapeamento com o banco de dados do sistema de origem, no caso, o Aplicativo Coleta de Dados CAPES.

4 DESENVOLVIMENTO DA SOLUÇÃO

Neste capítulo é apresentado o desenvolvimento do data warehouse, assim como sua integração com uma ferramenta de BI.

Optou-se pelo uso de ferramentas de código-fonte aberto. O data warehouse foi implementado no SGBD PostgreSQL, versão 9.2 [1]. E para a análise dos dados, utilizou-se o sistema Pentaho BI Platform, versão 4.8.0 [2].

4.1 Implementação do Data Warehouse

Com base no modelo dimensional desenvolvido no capítulo anterior, foi implementado o data warehouse. Como dito anteriormente, a implementação foi feita em um banco de dados relacional, com o SGBD PostgreSQL. Esse SGBD foi escolhido por ele ser de código-fonte aberto, ser robusto e de fácil integração com a plataforma de BI que foi usada (Pentaho BI Platform).

O data warehouse foi construído da seguinte forma: a modelagem dimensional foi executada em uma ferramenta de modelagem de código-fonte aberto, SQL Power Architect, versão 1.0.6 [3]. Com o modelo pronto, essa ferramenta permite a geração automática do script de criação do banco de dados relacional para ser executado no PostgreSQL. O script gerado pode ser visto no Apêndice A.

Importante notar que praticamente todos os campos que compõem as tabelas das dimensões são do tipo VARCHAR, com exceção das chaves primárias e estrangeiras. Isso acontece porque os campos das tabelas de dimensão deverão aparecer como rótulos dos relatórios e análises que serão feitas no banco de dados. Kimball e Ross (2013) recomendam que as informações contidas nas dimensões devem ser textuais e "verbosas", evitando o uso de códigos e abreviações. Isso ajuda na clareza dos resultados das análises e relatórios. Importante notar também que o tipo VARCHAR tem tamanho variável de acordo com o número de caracteres efetivamente usado na tabela, ou seja, esse tipo não desperdiça espaço de armazenamento sem que este seja realmente usado para armazenar informação. Visto que as tabelas do banco de dados terão um volume razoavelmente grande de registros, armazenando, inclusive, dados de histórico, o uso do tipo VARCHAR pode trazer uma grande economia de espaço de armazenamento.

[1] Disponível para download em <http://www.postgresql.org>.
[2] Disponível para download em <http://community.pentaho.com>.
[3] Disponível para download em <http://www.sqlpower.ca/page/architect_download_os>.

4.2 Plataforma de BI

Implementado o banco de dados, segue-se agora para a plataforma de BI. Como mencionado anteriormente, optou-se pelo sistema Pentaho BI Platform. Essa ferramenta vem com toda a estrutura necessária para a implementação da solução de BI: contém um console para gerenciamento de acesso dos usuários, permite a análise dinâmica dos dados do data warehouse e pode gerar relatórios, tudo isso via uma interface de usuário baseada na Web.

4.2.1 Conexão com o Data Warehouse

É preciso configurar a plataforma de BI para indicar que se quer usar o data warehouse implementado no SGBD PostgreSQL como fonte de dados (*data source*). Isso é feito na tela inicial do *Pentaho User Console*. O Pentaho User Console é acessado, na máquina em que a plataforma de BI estiver instalada e iniciada, pelo endereço <http://localhost:8080> em um navegador Web. É necessário que o usuário já tenha efetuado o login como Administrador. Na tela inicial há um botão para criação de um novo data source, "Create New". Um assistente de criação de data source aparece ("Data Source Wizard"). Na primeira etapa cria-se um nome para o data source, seleciona-se o tipo de origem ("Source Type") como "Database Table(s)" e informa-se uma conexão com o banco de dados, que também precisa ser criada, informando o host, a porta, o nome do banco, usuário e senha do PostgreSQL. Em "Create data source for", seleciona-se "Reporting and Analysis (Requires Star Schema)", para que se possa fazer tanto a análise dos dados quanto criar relatórios. A tela da primeira etapa é mostrada na Figura 4.1 com as primeiras configurações preenchidas.

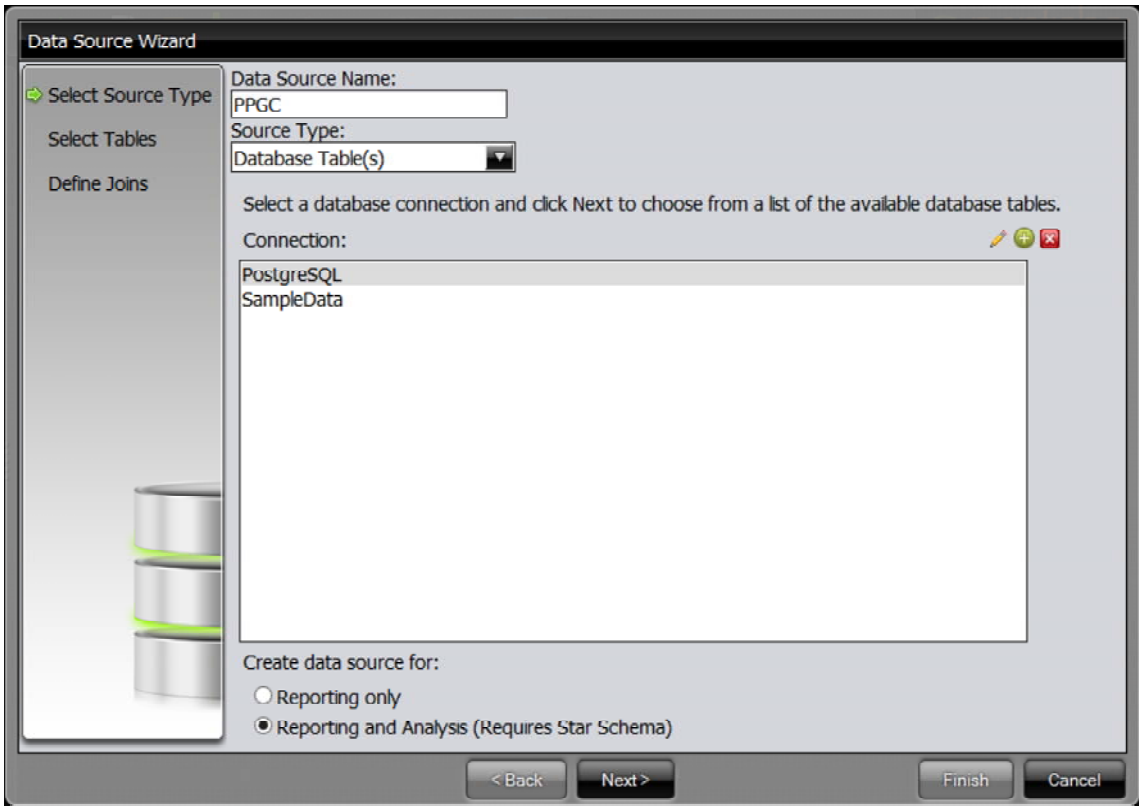


Figura 4.1: Primeira etapa do assistente de criação de data source

Na próxima etapa, é preciso informar quais as tabelas que serão usadas. Selecionou-se todas elas e indicou-se qual delas é a tabela de fatos no campo “Fact Table” (Figura 4.2).

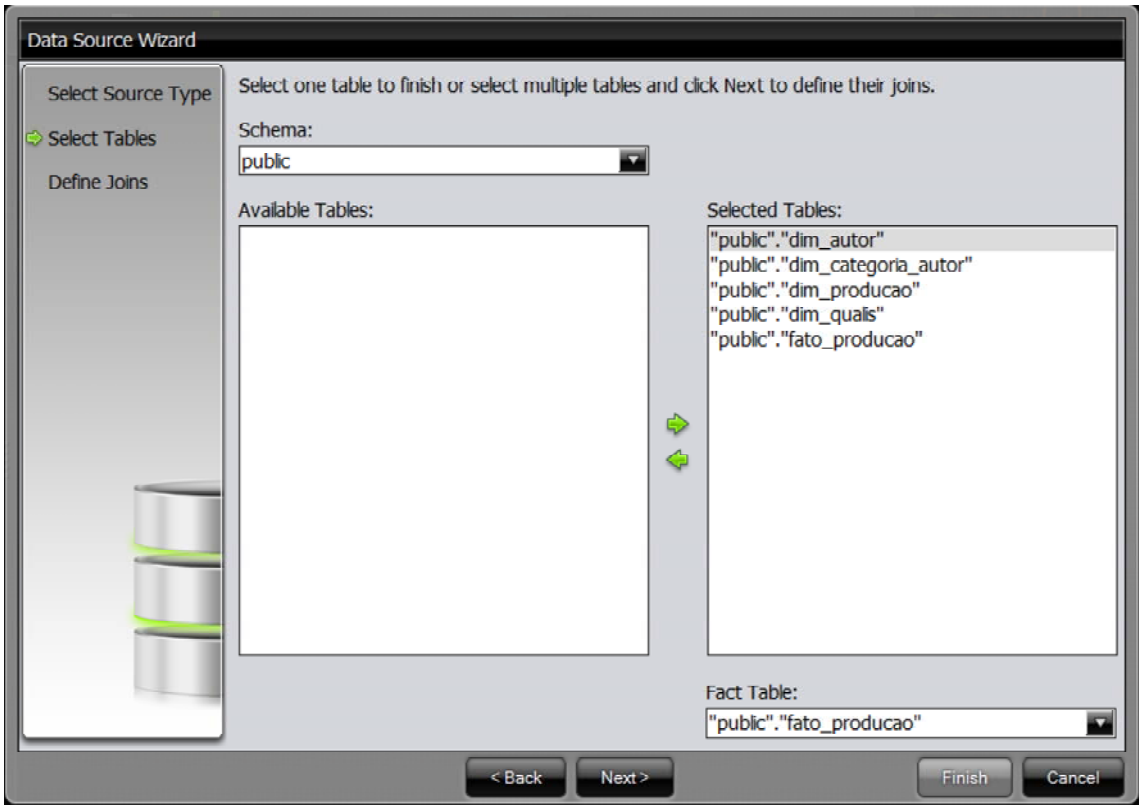


Figura 4.2: Segunda etapa do assistente de criação de data source

Na última etapa configurou-se como as tabelas se relacionam entre si. Para cada chave estrangeira da tabela de fatos é preciso indicar a respectiva chave primária da tabela de dimensão (Figura 4.3).

Após a confirmação da última etapa, a ferramenta cria um modelo padrão de metadados. Porém, ela sugere a personalização desse modelo se for necessário. É o que será visto a seguir.

4.2.2 Configuração do Modelo de Metadados

O modelo de metadados padrão gerado pela ferramenta não reflete a forma como o modelo foi projetado. É preciso alterar esse modelo padrão para que fique consistente com a visão do modelo dimensional que foi concebida. Isso é feito na configuração que será descrita a seguir.

Na realidade a configuração do modelo de metadados é um mapeamento do modelo estrela, que foi implementado no banco de dados relacional, para um modelo em cubo OLAP que a plataforma de BI usa internamente. Alguns conceitos e terminologias do contexto de modelos em cubo OLAP serão introduzidos ao longo da configuração. A documentação do Mondrian [4] foi usada como referência a esses conceitos.

A plataforma de BI possui uma configuração para a ferramenta de análise e uma para a geração de relatórios. Inicialmente será feita a configuração da primeira. Apesar da ferramenta não deixar isso explícito para o usuário, está sendo criado aqui um cubo OLAP.

[4]Disponível em http://mondrian.pentaho.com/documentation>.

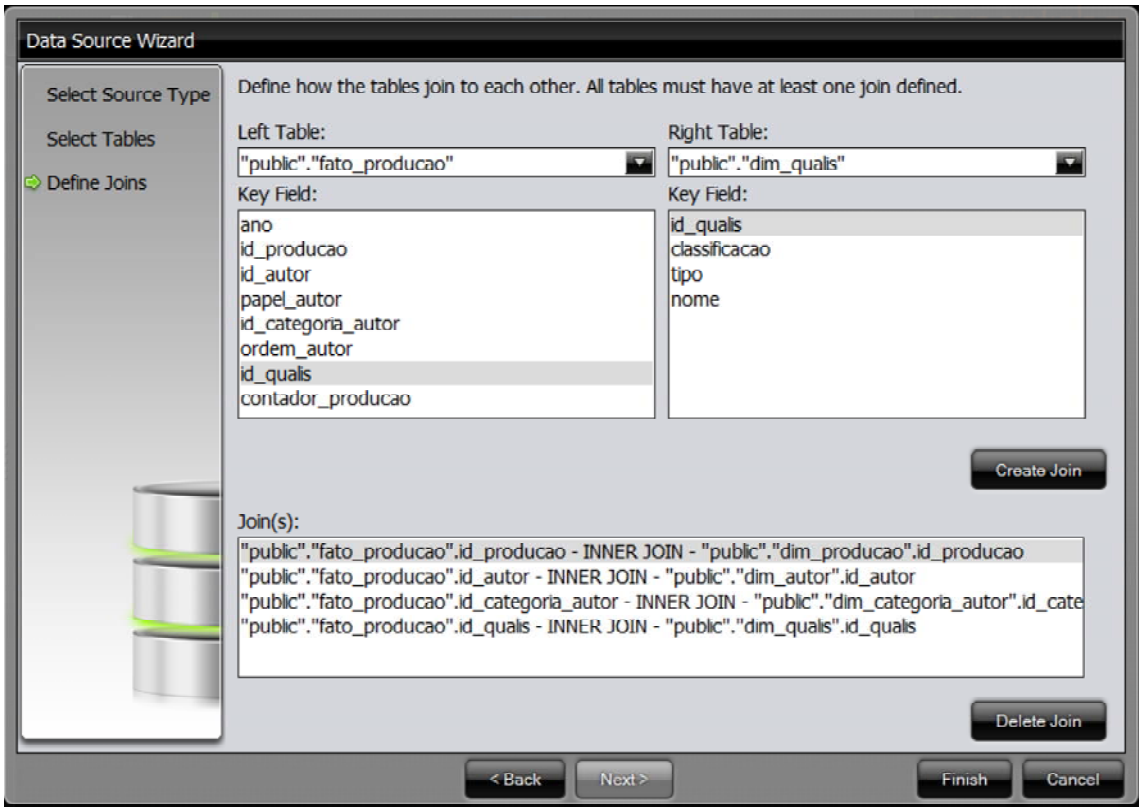


Figura 4.3: Última etapa do assistente de criação de data source

Um cubo é uma coleção de medições e dimensões. É exatamente essa divisão que se encontra na aba “Analysis” da tela de configuração do modelo, que pode ser vista na Figura 4.4.

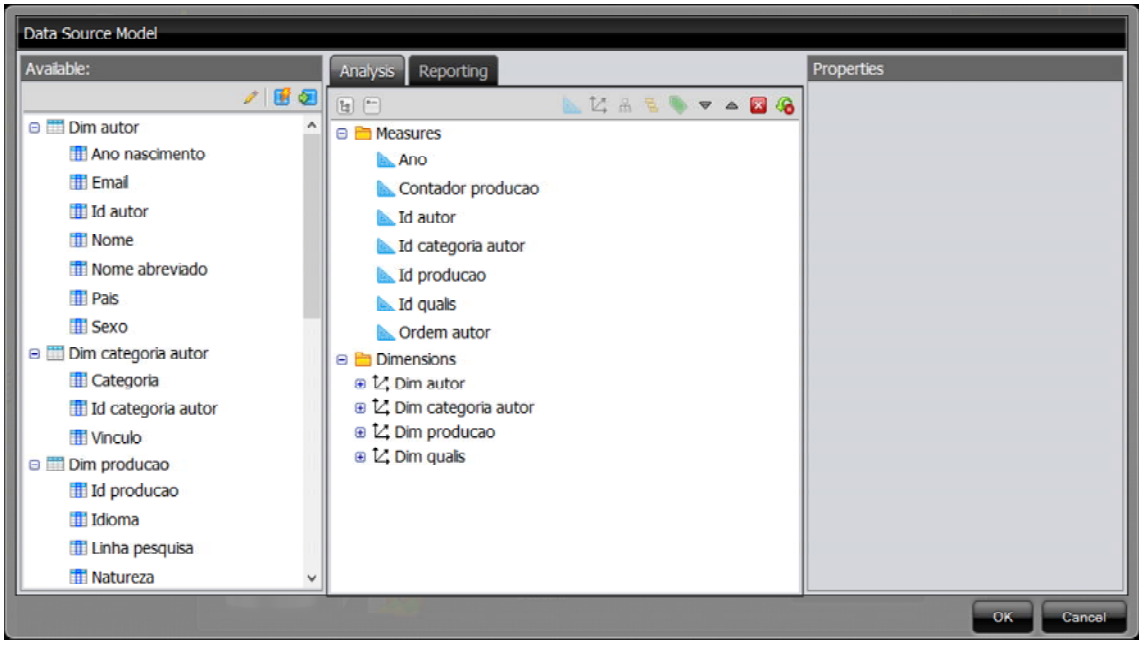


Figura 4.4: Tela de configuração do modelo de metadados

Em “Measures” (medições) é preciso colocar os campos de medição que estão na

tabela de fatos. No caso em questão, precisa-se apenas do campo “Contador producao” da tabela “Fato producao”. O campo foi renomeado para “Producao”, e a função SUM (soma) foi selecionada no campo “Default Aggregation” (Figura 4.5). Essa função é a operação padrão que é realizada sobre o campo de medição durante as análises dos dados. Visto que a função SUM faz uma soma dos valores dos campos, quando o campo “Producao” na ferramenta de análise for selecionado, ela somará os valores contidos nesse campo de acordo com o agrupamento e a seleção das dimensões de uma determinada pesquisa. Maiores detalhes sobre a análise de dados serão vistos adiante.

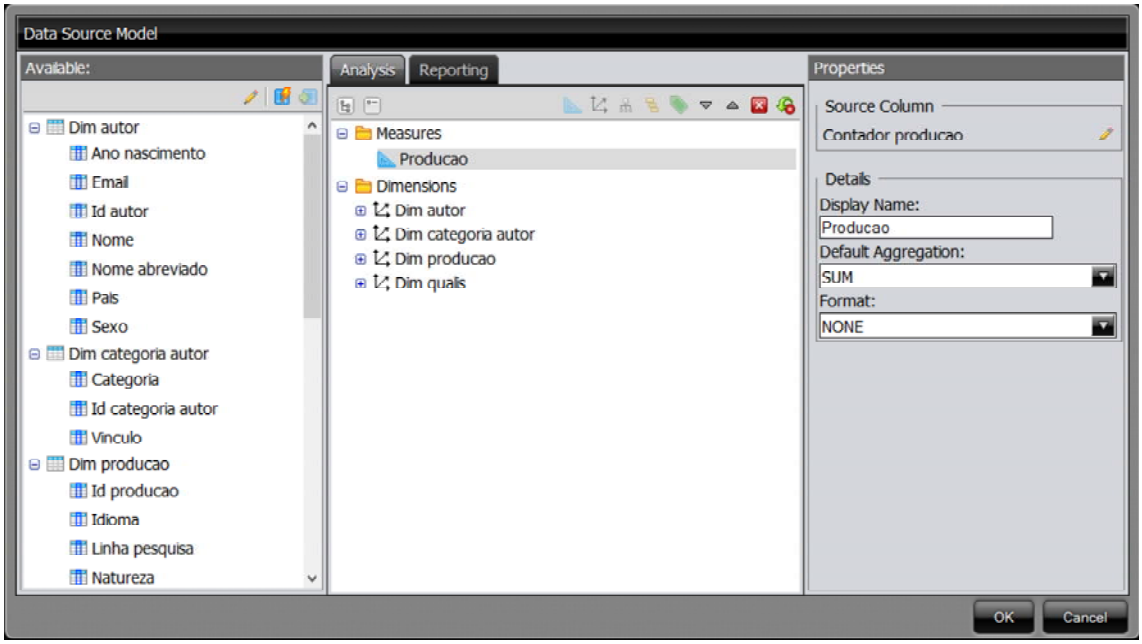


Figura 4.5: Configuração dos campos de medição

Em “Dimensions” (dimensões) são dispostos os campos das dimensões. As dimensões, porém, são mais complexas que as medições. Cada dimensão pode conter várias hierarquias (hierarchies), que são divididas por vários níveis (levels). Ainda, os níveis podem conter propriedades (member properties). A dimensão segue o mesmo conceito da dimensão do modelo dimensional em estrela. Na prática, uma dimensão será mapeada para uma tabela ou um campo da tabela de fatos que representa uma dimensão degenerada. Esse é o caso das dimensões Ano, Ordem Autor e Papel Autor. Essas três dimensões são criadas, arrastando-se os seus respectivos campos da tabela “Fato producao” para o painel central da tela, na área das dimensões. A ferramenta cria as hierarquias e níveis automaticamente para as novas dimensões, como mostra a Figura 4.6.

A hierarquia é um conjunto de membros (valores de atributos das dimensões) organizados em uma estrutura conveniente para análise. Essa estrutura é formada por níveis, onde cada nível é uma coleção de membros que possuem a mesma distância da raiz da hierarquia. A hierarquia permite, com isso, formar subtotais intermediários. Por exemplo, na dimensão Categoria Autor tem-se uma hierarquia composta de categoria e vínculo do autor com o programa. A categoria é o nível mais alto da hierarquia, seguida do vínculo. Assim, pode-se ter o subtotal da produção da categoria de docentes, que é a soma dos subtotais dos tipos de vínculos possíveis para a categoria docente (permanente, visitante, colaborador). A Figura 4.7 mostra a configuração da hierarquia para a dimensão Categoria Autor.

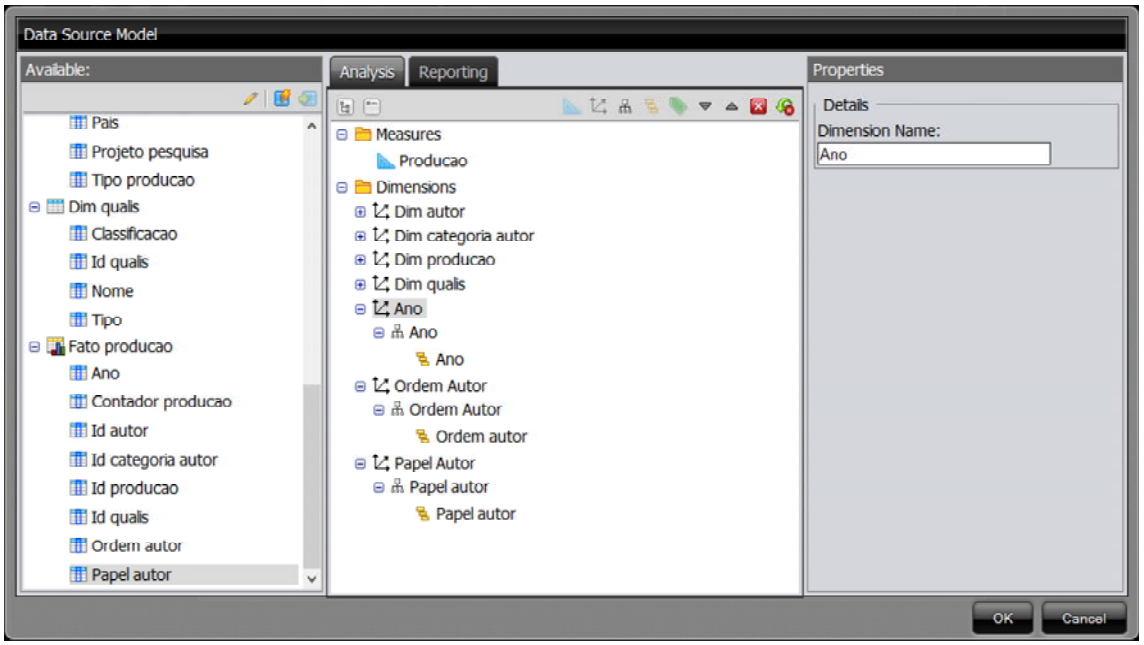


Figura 4.6: Configuração das dimensões degeneradas

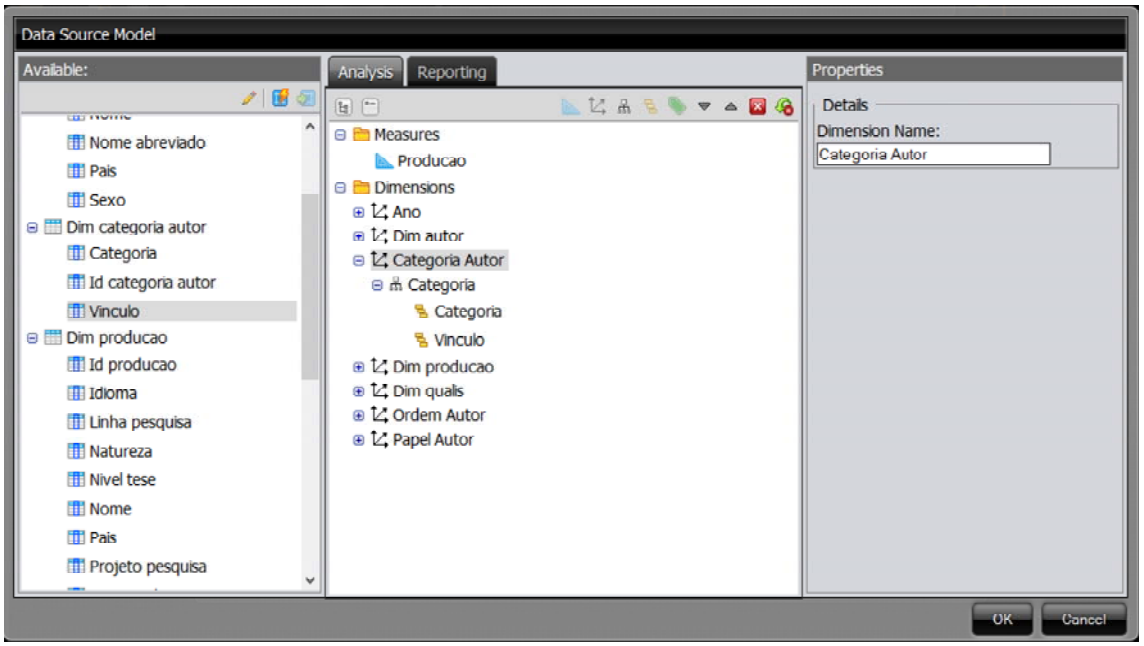


Figura 4.7: Configuração da dimensão Categoria Autor

A ordem em que estão dispostos os níveis é importante. “Categoria” deve estar acima de “Vinculo” para que “Categoria” seja o nível pai de “Vinculo” na hierarquia.

A Figura 4.8 mostra a configuração da dimensão Autor. Foram criadas quatro hierarquias: “Nome”, “Ano Nascimento”, “Pais” (país de origem do autor), e “Sexo”. É importante notar a inclusão das propriedades de membro na hierarquia “Nome” (indicados pelos ícones em forma de etiquetas verdes na figura). Elas servem como informação complementar do membro “Nome” do autor, mas elas podem também ser usadas como condições de filtro nas consultas sobre o banco de dados.

A configuração da dimensão Producao contém muitas hierarquias. A Figura 4.9 des-

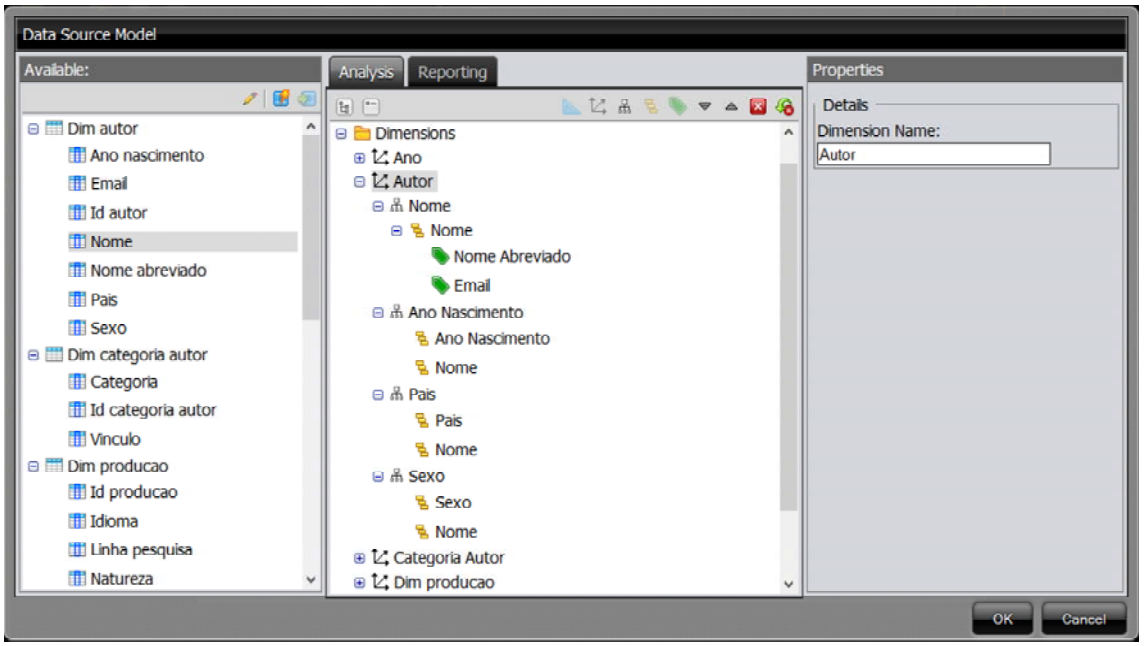


Figura 4.8: Configuração da dimensão Autor

taca a hierarquia “Linha Pesquisa”. Ela informa que uma linha de pesquisa pode ter vários projetos de pesquisa e que um projeto de pesquisa, por sua vez, pode ter várias produções (nível “Nome”). Seria possível incluir aqui outros níveis, como por exemplo, “Tipo Producao”, mas decidiu-se por deixar essa hierarquia mais simples.

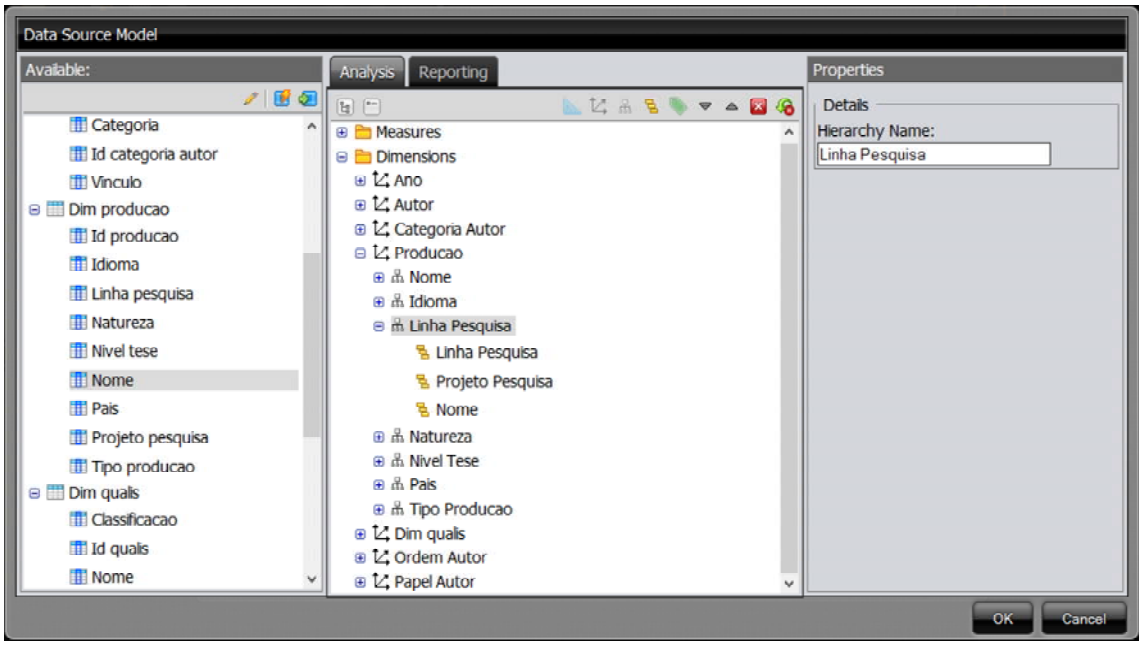


Figura 4.9: Configuração da dimensão Producao

Por fim a dimensão Qualis foi configurada. Decidiu-se por estruturar essa dimensão com apenas uma hierarquia, como mostra a Figura 4.10. Ela permite agrupar a dimensão por tipo (periódico ou evento), depois pela classificação e, por último, listar os nomes dos eventos/periódicos.

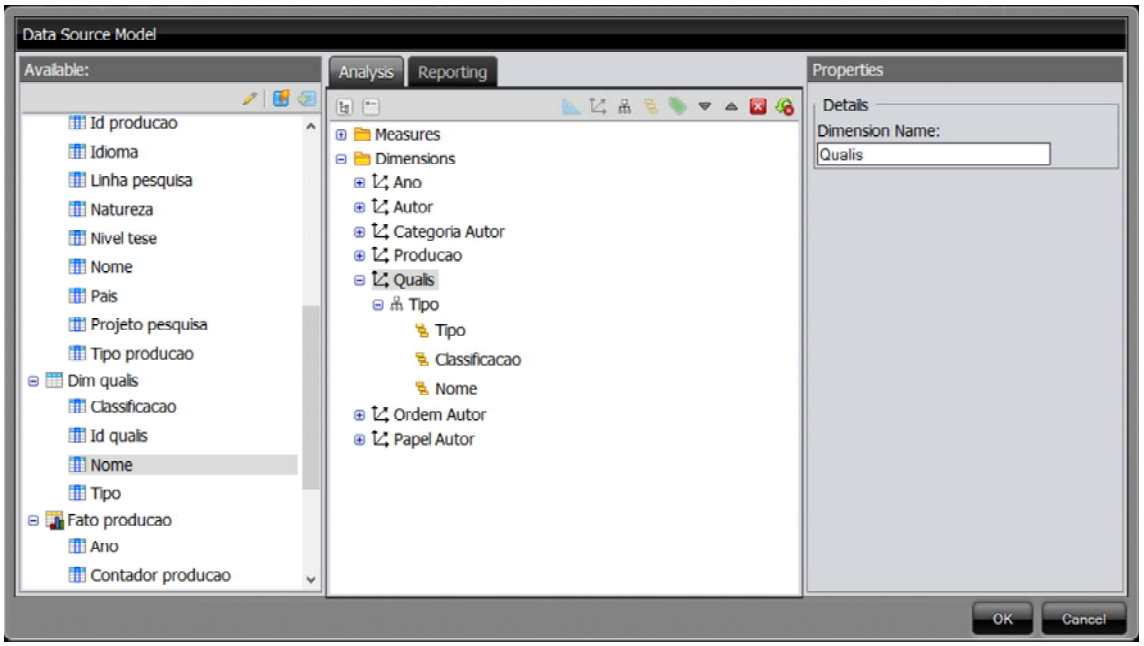


Figura 4.10: Configuração da dimensão Qualis

Feita a configuração, o data warehouse está pronto para ser utilizado pela ferramenta de análise do Pentaho BI Platform.

A configuração para a geração de relatórios é feita a seguir. Na aba “Reporting” se encontra o modelo padrão gerado pela ferramenta. Novamente aqui é preciso que alguns ajustes sejam feitos para refletir a estrutura do modelo projetado.

A configuração se divide em categorias de campos (Categories) e campos (Fields). Uma categoria possui vários campos. Na prática, cada categoria irá se referir a uma tabela de fatos ou uma dimensão do modelo no final da configuração.

Começando pela categoria “Fato producao”, foram excluídos os campos que se referem às chaves estrangeiras (“Id autor”, “Id categoria autor”, “Id producao e Id qualis”), assim como os campos que são dimensões degeneradas (“Ano”, “Ordem autor”, “Papel autor”), mantendo-se apenas o campo “Contador producao”, como mostra a Figura 4.11.

A Figura 4.11 também mostra a configuração do campo “Contador producao” no painel à direita. É possível ver a escolha da função SUM como função de agregação padrão.

O restante das outras categorias foram deixadas com os campos e configurações padrão. Foi preciso, porém, incluir categorias para as dimensões degeneradas Ano, Ordem Autor e Papel Autor. A Figura 4.12 ilustra as categorias criadas.

Esse último passo encerra a configuração referente à geração de relatórios.

4.3 Resumo

Neste capítulo foram descritas a implementação do data warehouse e a configuração da ferramenta de BI para utilizar o data warehouse como fonte de dados. Inicialmente, com o auxílio de uma ferramenta de modelagem, foi gerado um script em SQL para a criação do banco de dados no SGBD PostgreSQL. Em seguida, configurou-se a ferramenta de BI Pentaho BI Platform para usar o banco de dados criado como fonte de dados. Finalmente, o modelo padrão gerado pela ferramenta foi alterado para refletir a estrutura real do modelo dimensional desenvolvido.

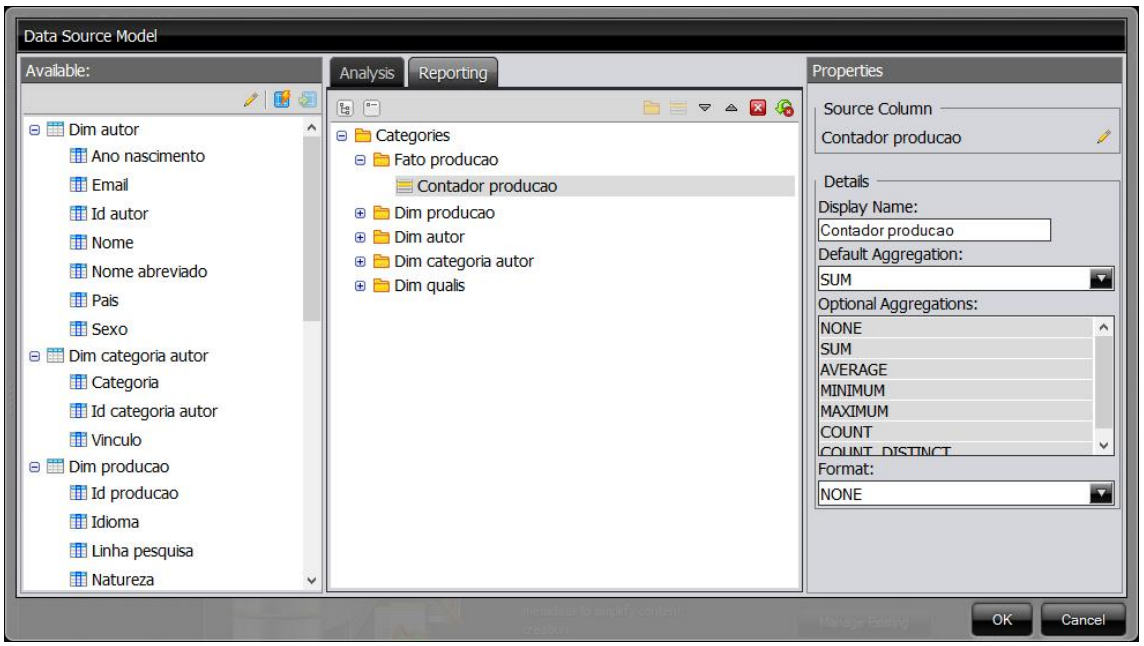


Figura 4.11: Configuração da categoria Fato Produção

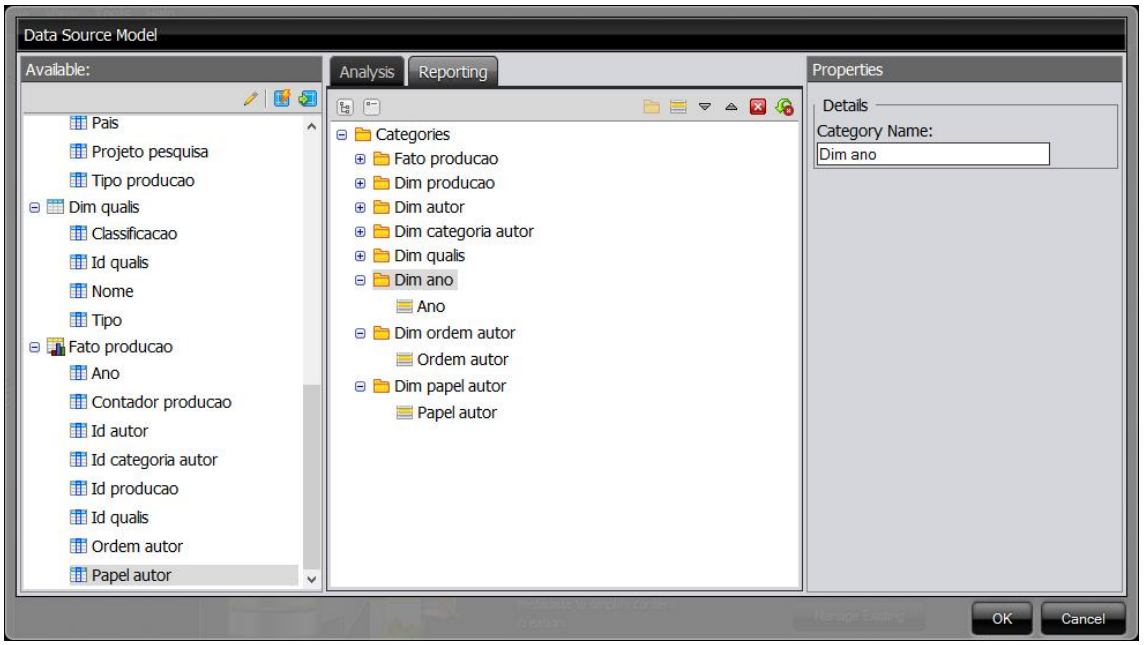


Figura 4.12: Inclusão das categorias referentes às dimensões degeneradas

5 USO DA SOLUÇÃO DESENVOLVIDA

Neste capítulo será apresentado como a ferramenta de BI, Pentaho BI Platform, é utilizada para a análise da produção acadêmica do PPGC. Inicialmente, exemplos de consulta são descritos para ilustrar o funcionamento da ferramenta de análise. Em seguida, um dos exemplos de consulta é reproduzido na demonstração de uso da ferramenta de geração de relatórios.

5.1 Ferramenta de Análise

Na tela principal do Pentaho BI Platform é possível ter acesso à ferramenta de análise clicando-se no botão "New Analysis" e selecionando-se o esquema "PPGC", nome dado ao data source de acordo com a configuração feita no capítulo anterior. Inicialmente a ferramenta mostra uma tabela com todos os dados da primeira hierarquia de cada dimensão.

Para ilustrar como é feita a análise dos dados na ferramenta serão apresentados, a seguir, exemplos de consultas, tomando como base as consultas listadas nos requisitos do terceiro capítulo.

A primeira consulta de exemplo se refere à análise da produção por autor, ou seja, qual a quantidade de produções acadêmicas cada autor possui. A análise também será agrupada por ano, tanto nessa consulta quanto nas seguintes. A análise pode ser feita usando o OLAP Navigator, clicando-se no botão com um ícone no formato de um cubo na barra de ferramentas, como mostra a Figura 5.1.

A Figura 5.1 também mostra a configuração final da consulta. Em "Columns" (colunas) foi incluída a hierarquia "Ano" e em "Rows" (linhas) a hierarquia "Nome" da dimensão Autor. Essas são as informações pelas quais serão agrupados os resultados da consulta. Em "Filter" (filtro) foi selecionada a "Measure" (medição) "Producao" e o "Papel Autor" com o valor "Autor". Este último é muito importante para o resultado, visto que se quer apenas a contagem da produção onde os pesquisadores (pessoas vinculadas ao PPGC) participaram como autores das produções. Outros valores possíveis seriam "Orientador" e "Co-Orientador", que não são pertinentes a essa análise. Importante notar, também, que no resultado dessa consulta podem aparecer produções repetidas, pois uma mesma produção pode ter dois ou mais autores. Isso é perfeitamente aceitável, visto que o que interessa aqui é a produção por autor.

Finalmente, a Figura 5.1 também mostra o resultado da consulta de forma resumida, ou seja, apenas os totais de cada ano. É possível ver que a produção total (incluindo produções repetidas) é de 64, sendo 36 autorias em 2011 e 28 em 2012. Esse resultado, porém, pode ser facilmente detalhado expandindo-se a linha de nomes (onde aparece "All Autor.Nomes"). A Figura 5.2 ilustra a produção por autor. Isso mostra um pouco da flexibilidade da ferramenta de análise. Não é necessário criar uma nova consulta para

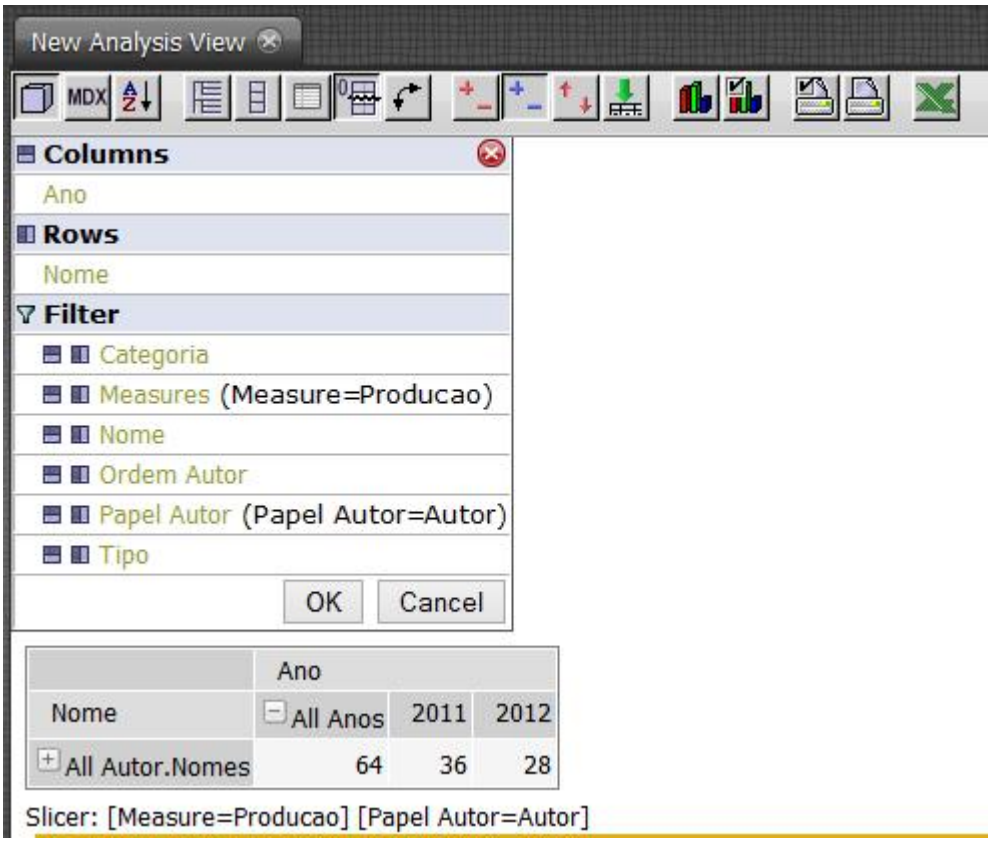


Figura 5.1: Análise de produção por autor

detalhar as informações. A análise dos dados pode ser feita de forma dinâmica, resumindo e detalhando os dados de forma interativa.

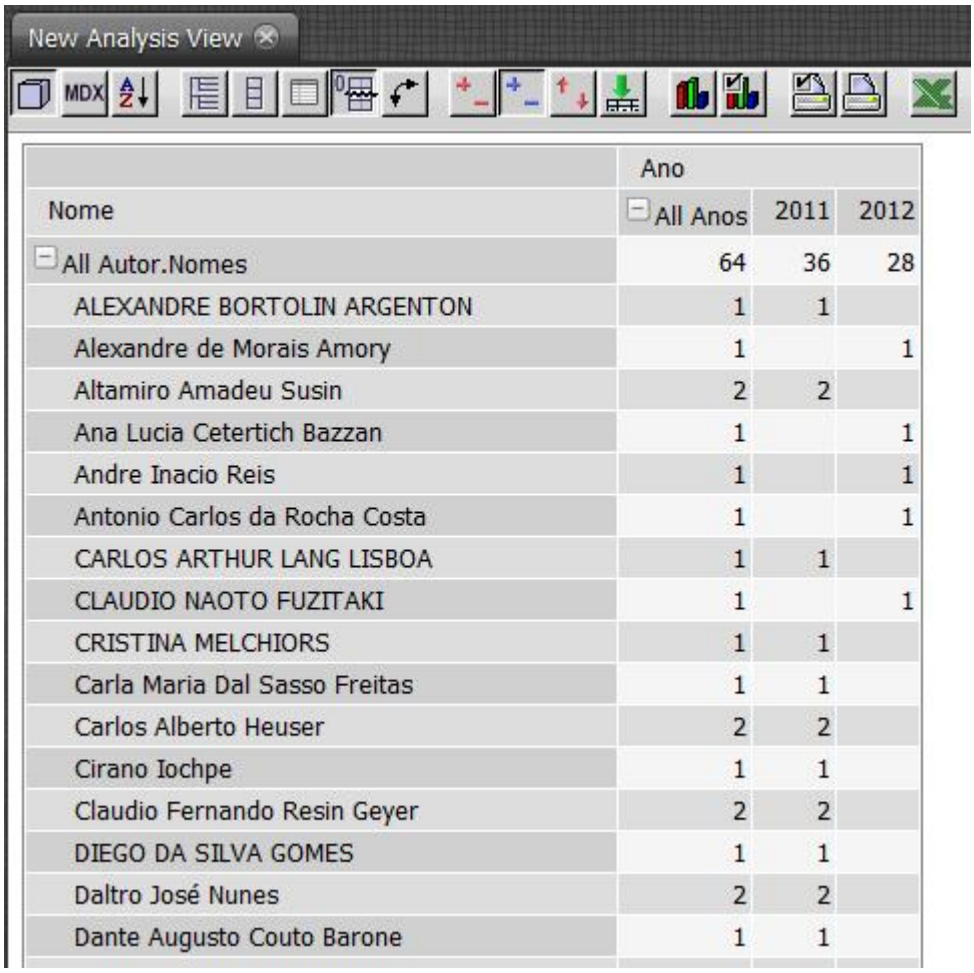
A consulta a seguir se refere à análise da produção por categoria de autor (docentes permanentes, discentes de mestrado, etc.). A Figura 5.3 mostra a configuração do OLAP Navigator e o resultado da consulta com as linhas já expandidas. Nesse resultado a hierarquia de categorias é mais profunda: as categorias se dividem em "Discente" e "Docente", onde cada uma delas também é subdividida. Essa hierarquia foi definida na configuração feita no capítulo anterior. Importante notar, novamente, os filtros definidos no OLAP Navigator. São obtidas apenas as produções cujos pesquisadores atuaram como autores.

A próxima consulta analisa a produção por linha de pesquisa, como mostra a Figura 5.4.

Importante notar que foi incluído mais um filtro, junto com o "Papel Autor": "Ordem Autor" 1. Isso garante que sejam retornados apenas um registro de cada produção, visto que toda produção precisa ter pelo menos um autor e que esse autor sempre terá a ordem 1. Não é possível assim que uma produção apareça duas vezes no resultado, como aconteceu nas consultas anteriores.

Essa análise também pode ser detalhada, listando os projetos de pesquisa para cada linha de pesquisa, como mostra a Figura 5.5. Novamente, isso é possível devido à configuração das hierarquias feita no capítulo anterior.

A última consulta da lista de requisitos analisa a produção total do programa por ano, como mostra a Figura 5.6. Os filtros são os mesmos da consulta anterior, pois se quer saber apenas a produção efetiva, sem considerar as relações de autoria/orientação. A hierarquia "Ano", porém, foi colocada nas linhas e a medição nas colunas. Isso gerou o



Nome	Ano		
	All Anos	2011	2012
All Autor.Nomes	64	36	28
ALEXANDRE BORTOLIN ARGENTON	1	1	
Alexandre de Morais Amory	1		1
Altamiro Amadeu Susin	2	2	
Ana Lucia Cetertich Bazzan	1		1
Andre Inacio Reis	1		1
Antonio Carlos da Rocha Costa	1		1
CARLOS ARTHUR LANG LISBOA	1	1	
CLAUDIO NAOTO FUZITAKI	1		1
CRISTINA MELCHIORS	1	1	
Carla Maria Dal Sasso Freitas	1	1	
Carlos Alberto Heuser	2	2	
Cirano Iochpe	1	1	
Claudio Fernando Resin Geyer	2	2	
DIEGO DA SILVA GOMES	1	1	
Daltro José Nunes	2	2	
Dante Augusto Couto Barone	1	1	

Figura 5.2: Resultados detalhados de produção por autor

resultado que pode ser visto na parte inferior da Figura 5.6.

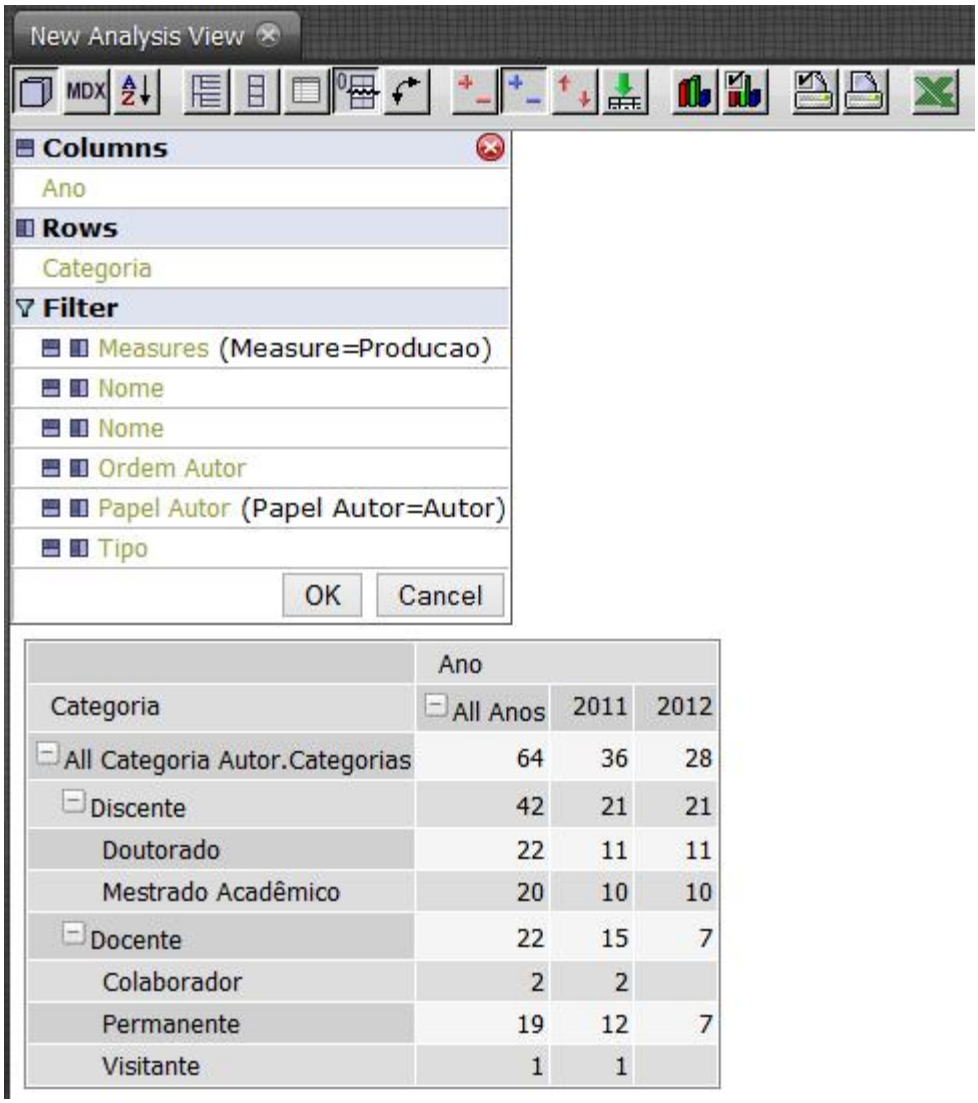
Com os exemplos apresentados é possível verificar que as consultas listadas nos requisitos foram satisfeitas. Além disso, é possível criar outras consultas com o uso do OLAP Navigation, incluindo diferentes dimensões e filtros.

5.2 Geração de Relatórios

A seguir será descrito um exemplo de geração de relatório para a consulta de produção acadêmica por linha de pesquisa, a terceira entre as consultas vistas anteriormente nos exemplos da ferramenta de análise.

Na tela inicial do Pentaho BI Platform, a ferramenta de geração de relatórios pode ser acessada clicando-se no botão “New Report”. A geração de relatório é divida em quatro passos. O primeiro passo consiste em selecionar um data source e um template para o relatório, como mostra a Figura 5.7. É possível ver na figura a seleção do data source PPGC, criado e configurado no capítulo anterior, e a seleção de um dos templates.

O segundo passo consiste na seleção dos itens que serão mostrados no relatório. No painel “Available Items” (itens disponíveis) estão as categorias e campos configurados no capítulo anterior. A parte mais à direita mostra os campos selecionados. Essa parte é dividida em grupos (“Groups”), detalhes (“Details”) e filtros (“Filters”). Os grupos servem para agrupar os dados do relatório. Um ou mais campos podem ser selecionados para criar

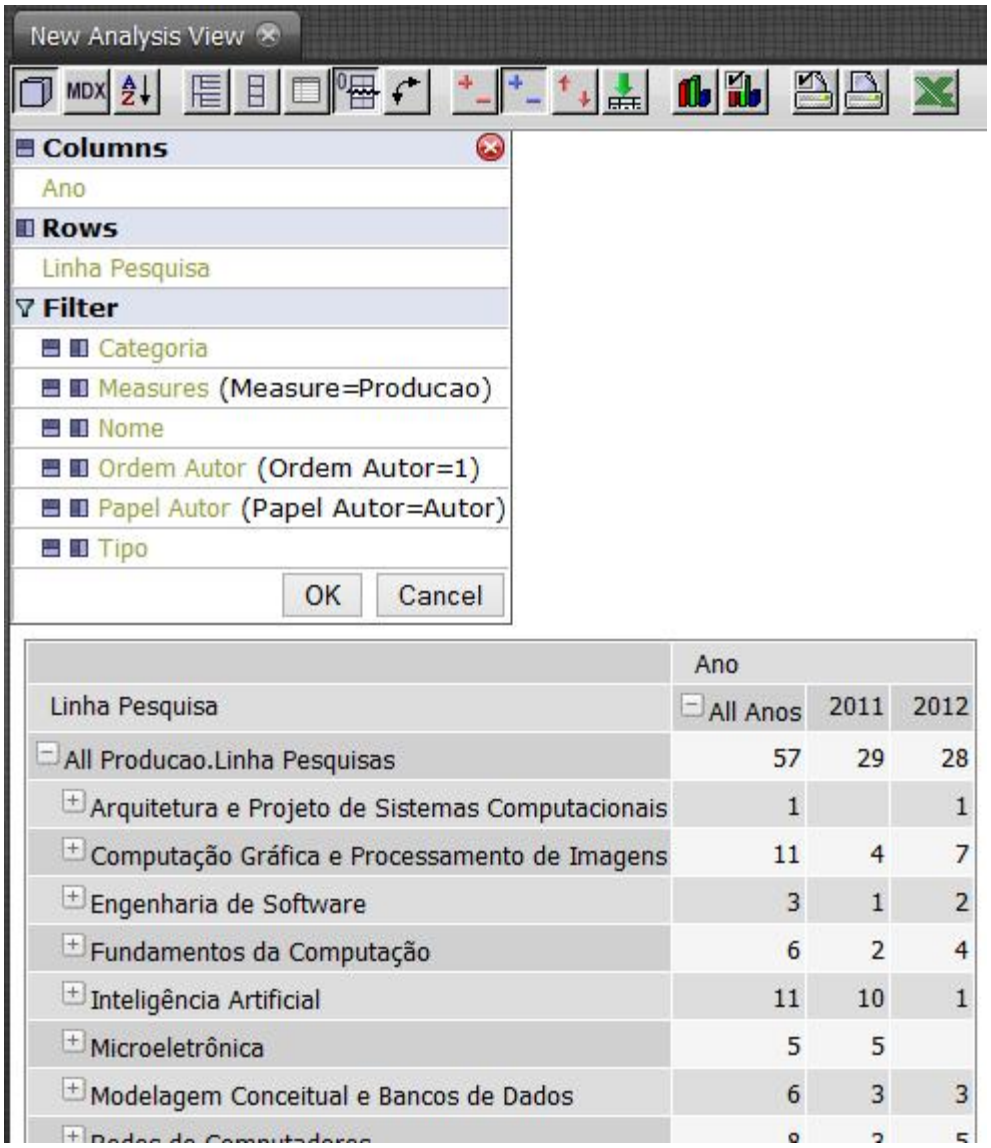


	Ano		
Categoria	All Anos	2011	2012
All Categoria Autor.Categorias	64	36	28
Discente	42	21	21
Doutorado	22	11	11
Mestrado Acadêmico	20	10	10
Docente	22	15	7
Colaborador	2	2	
Permanente	19	12	7
Visitante	1	1	

Figura 5.3: Análise de produção por categoria de autor

o agrupamento. Os detalhes abrigam os campos que mostram os dados de interesse do relatório. E os filtros são campos usados para filtrar as informações que serão mostradas no relatório. A Figura 5.8 ilustra a seleção dos campos para o exemplo discutido. O campo "Ano" foi selecionado para agrupar os dados. Os campos "Linha pesquisa" e "Contador producao" estão nos detalhes para apresentar os dados. E os campos "Ordem autor" e "Papel autor" estão nos filtros para restringir o resultado. Essa restrição é configurada no próximo passo.

No terceiro passo são feitos alguns ajustes nas seleções feitas no passo anterior. Os campos dentro dos grupos e dos detalhes podem ser formatados, selecionando-se o formato numérico e o alinhamento. Eles também podem ser filtrados, adicionando-se restrições no painel "Constraints" (restrições), e também podem ser ordenados de forma crescente ou decrescente no painel "Sort Columns" (ordenar colunas). No exemplo, os campos "Ano" e "Linha pesquisa" foram deixados com as configurações padrão. Nos campos de detalhes há ainda a opção de "Calculation" (cálculo) com funções para realizar cálculos sobre os resultados do relatório no final de cada grupo. Voltando ao exemplo, no campo "Contador producao" foi selecionada a função Sum. Isso irá mostrar uma soma

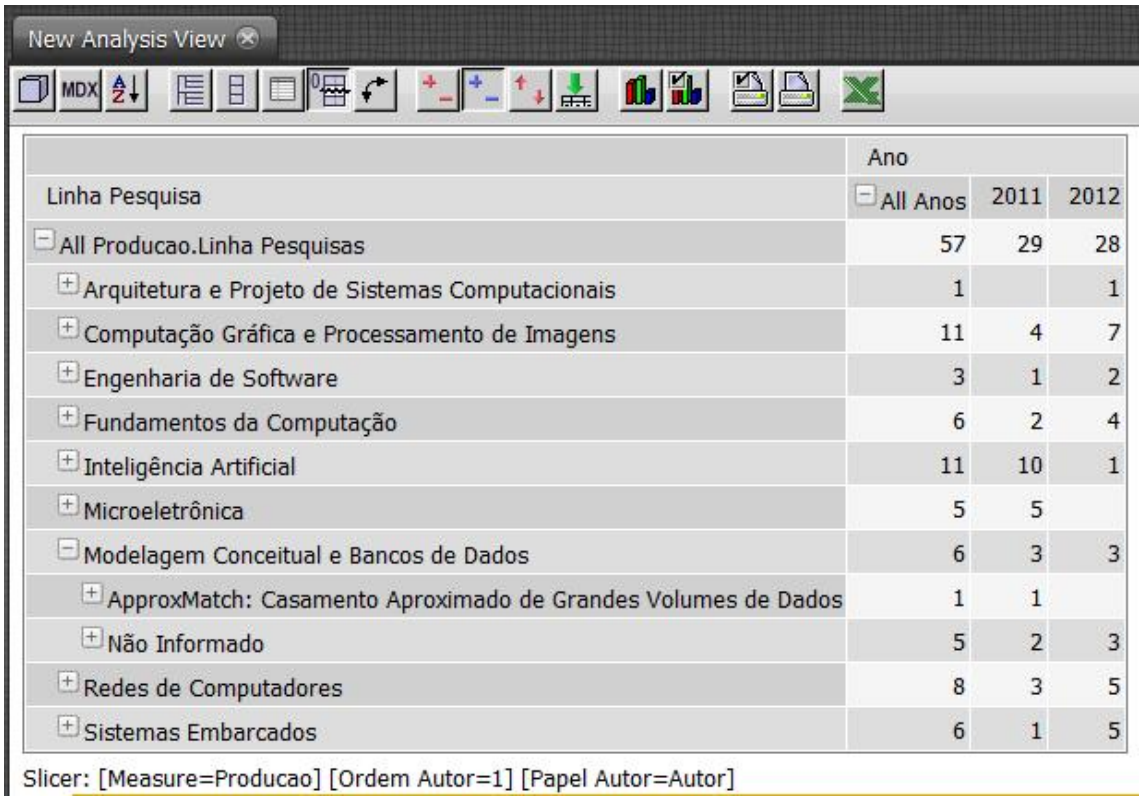


	Ano		
Linha Pesquisa	All Anos	2011	2012
All Producao.Linha Pesquisas	57	29	28
Arquitetura e Projeto de Sistemas Computacionais	1		1
Computação Gráfica e Processamento de Imagens	11	4	7
Engenharia de Software	3	1	2
Fundamentos da Computação	6	2	4
Inteligência Artificial	11	10	1
Microeletrônica	5	5	
Modelagem Conceitual e Bancos de Dados	6	3	3

Figura 5.4: Análise de produção por linha de pesquisa

total da produção no final de cada grupo, definido pelo ano. Finalmente, foram incluídas uma restrição para cada campo dos filtros, “Ordem autor” igual a 1 e “Papel autor” com o valor “Autor”. Esses filtros seguem a mesma lógica da consulta feita na ferramenta de análise para as linhas de pesquisa. A Figura 5.9 mostra as principais configurações feitas no terceiro passo.

No último passo são feitas as configurações de página do relatório. É possível criar cabeçalhos e rodapés, escrever uma descrição do relatório, a orientação da página e o tamanho do papel. No exemplo a orientação retrato (“portrait”) foi escolhida em um papel tamanho A4. Feito isso, o relatório foi gerado usando a opção “Preview As” na parte inferior da tela. O formato HTML foi escolhido e o botão “Go” foi pressionado, gerando o relatório mostrado na Figura 5.10.



	Ano		
Linha Pesquisa	All Anos	2011	2012
All Producao.Linha Pesquisas	57	29	28
Arquitetura e Projeto de Sistemas Computacionais	1		1
Computação Gráfica e Processamento de Imagens	11	4	7
Engenharia de Software	3	1	2
Fundamentos da Computação	6	2	4
Inteligência Artificial	11	10	1
Microeletrônica	5	5	
Modelagem Conceitual e Bancos de Dados	6	3	3
ApproxMatch: Casamento Aproximado de Grandes Volumes de Dados	1	1	
Não Informado	5	2	3
Redes de Computadores	8	3	5
Sistemas Embarcados	6	1	5

Slicer: [Measure=Producao] [Ordem Autor=1] [Papel Autor=Autor]

Figura 5.5: Análise detalhada de produção por linha de pesquisa

5.3 Resumo

O uso da solução desenvolvida foi apresentado neste capítulo. As ferramentas de análise e de geração de relatórios do Pentaho BI Platform foram descritas por meio de exemplos baseados nas consultas listadas nos requisitos do sistema.

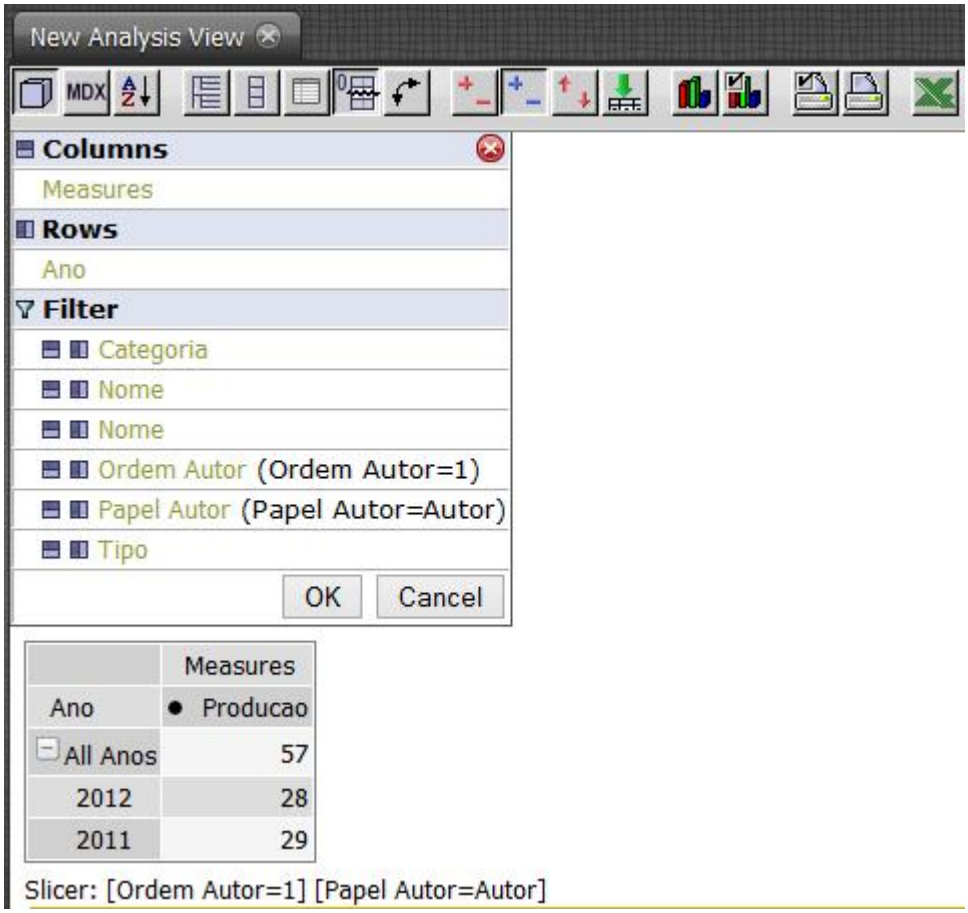


Figura 5.6: Análise de produção total por ano

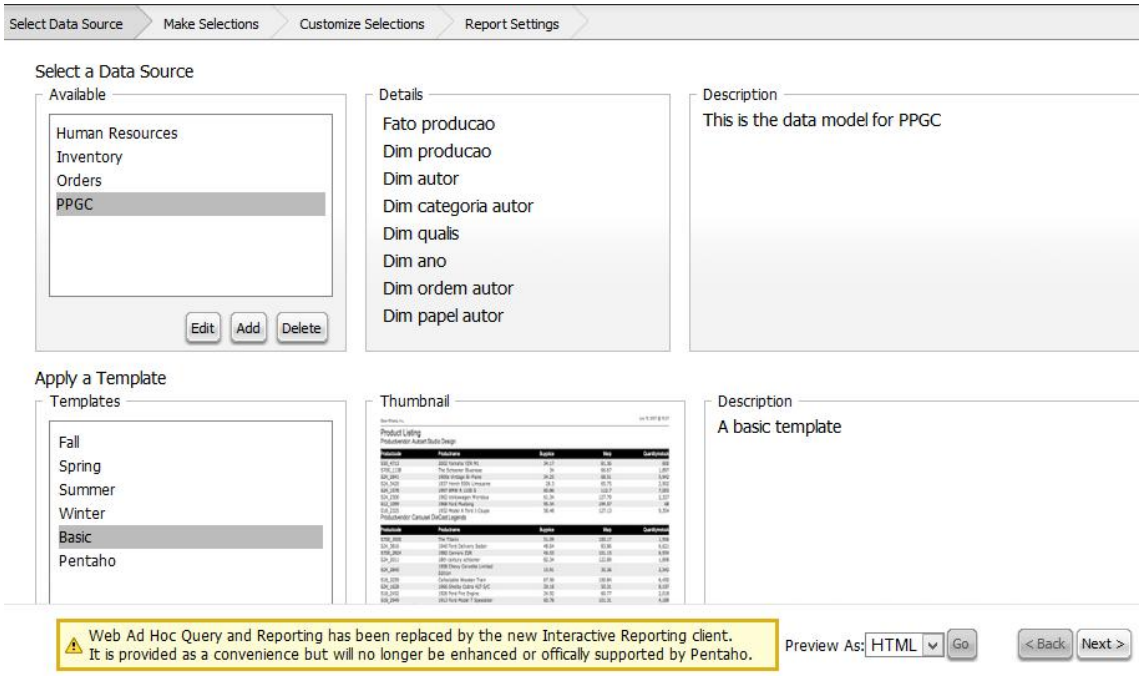


Figura 5.7: Primeiro passo da geração de relatório: seleção de data source

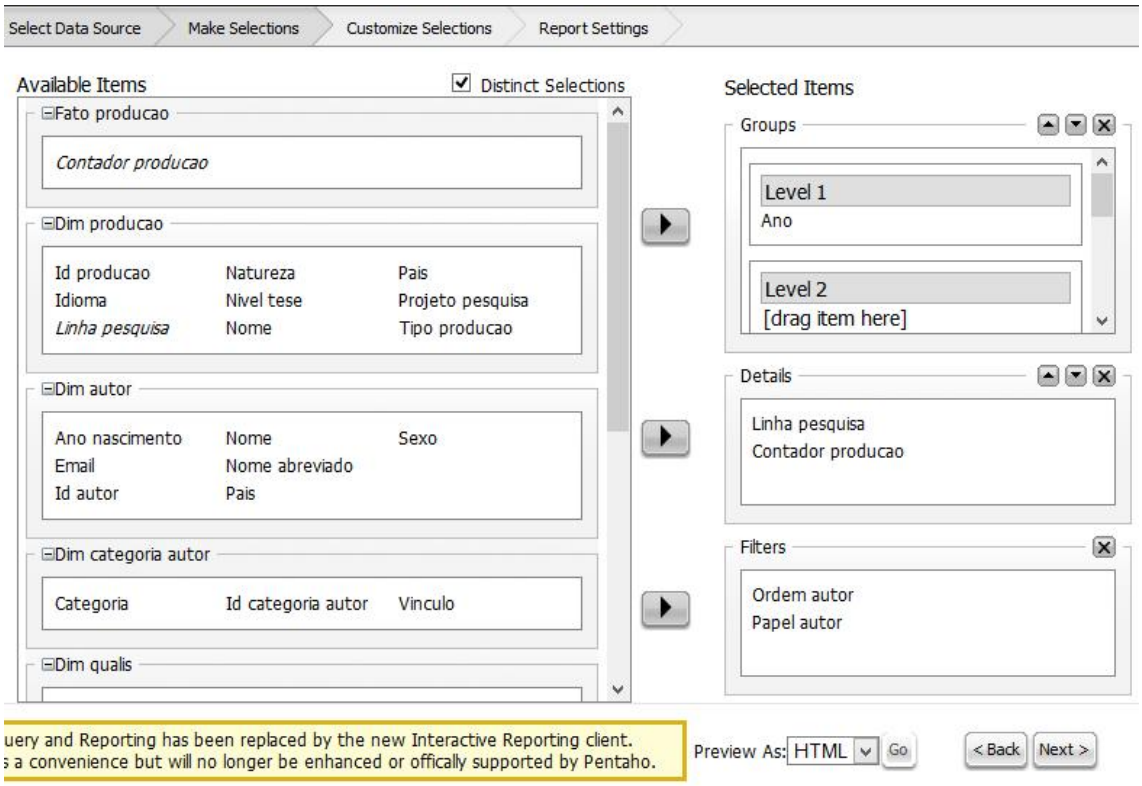


Figura 5.8: Segundo passo da geração de relatório: seleção dos campos

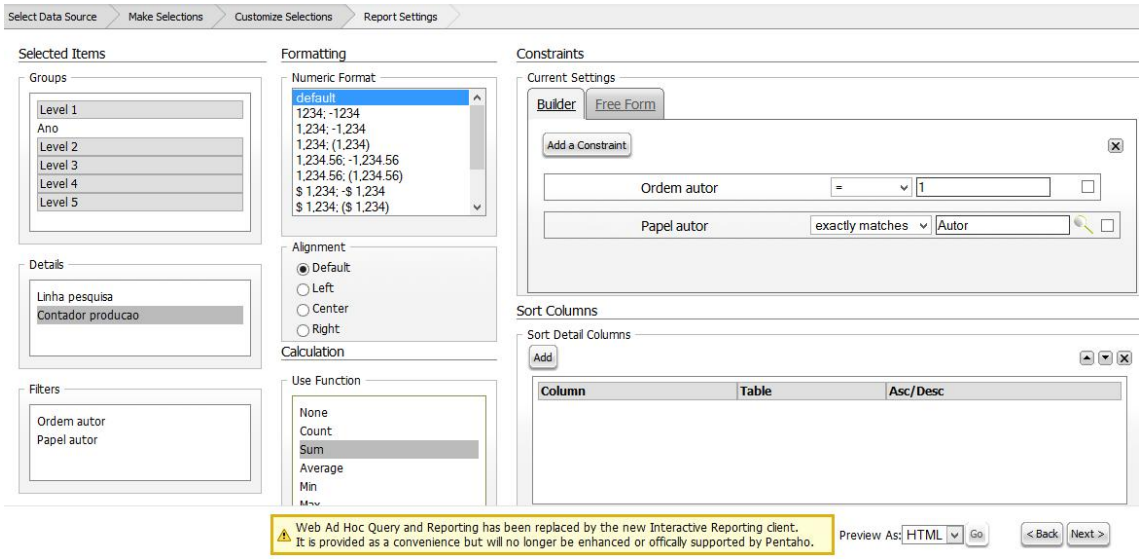


Figura 5.9: Terceiro passo da geração de relatório: ajustes das seleções

Untitled Preview

Ano: 2011

Linha pesquisa	Contador producao
Engenharia de Software	1
Sistemas Embarcados	1
Modelagem Conceitual e Bancos de Dados	3
Redes de Computadores	3
Computação Gráfica e Processamento de Imagens	4
Inteligência Artificial	10
Microeletrônica	5
Fundamentos da Computação	2
Total 2011	**29**

Ano: 2012

Linha pesquisa	Contador producao
Engenharia de Software	2
Inteligência Artificial	1
Arquitetura e Projeto de Sistemas Computacionais	1
Modelagem Conceitual e Bancos de Dados	3
Sistemas Embarcados	5
Redes de Computadores	5
Fundamentos da Computação	4
Computação Gráfica e Processamento de Imagens	7
Total 2012	**28**

Figura 5.10: Relatório de produção por linha de pesquisa

6 CONCLUSÃO

Este trabalho apresentou o projeto e desenvolvimento de uma solução de data warehouse e business intelligence para a análise da produção acadêmica do Programa de Pós-Graduação em Computação. Com base nos requisitos elicitados, desenvolveu-se um data warehouse para armazenar e estruturar os dados de produção acadêmica, utilizando a técnica de modelagem dimensional encontrada no referencial teórico deste trabalho e apresentada no segundo capítulo.

Para a análise dos dados, foi utilizada uma ferramenta de código-fonte aberto, o sistema Pentaho BI Platform. Esse sistema permite a análise dinâmica dos dados e a geração de relatórios, satisfazendo dois itens da lista de requisitos. Para utilizar o data warehouse desenvolvido, foram necessários alguns passos de configuração da ferramenta, que foram apresentados no quarto capítulo.

A utilização da ferramenta foi apresentada no quinto capítulo. Foram apresentados exemplos ilustrando as consultas que foram listadas nos requisitos. Isso mostrou que os requisitos de análise, incluindo a geração de relatórios, foram satisfeitos pela solução.

Este trabalho, porém, não apresentou uma solução de importação dos dados do sistema de origem, o Aplicativo Coleta de Dados CAPES. A solução seria o desenvolvimento de um sistema ETL, cuja função é extrair os dados do sistema de origem, transformar e adaptar os dados de acordo com o data warehouse desenvolvido, além de verificar a qualidade dos dados e, por fim, carregar os dados no data warehouse. Apesar deste trabalho não apresentar o desenvolvimento desse sistema, ele o considera na descrição do projeto do data warehouse, visto que a documentação detalhada apresenta um mapeamento dos dados do sistema de origem para o data warehouse, assim como uma breve descrição das regras que o sistema ETL deverá seguir para cada mapeamento. O projeto do data warehouse pode, portanto, ser utilizado em um futuro desenvolvimento do sistema ETL para essa solução.

REFERÊNCIAS

EVELSON, B.; NICOLSON, N. **Topic Overview**: business intelligence. Cambridge, US: Forrester Research, Inc., 2008.

FUNDAÇÃO CAPES. **Avaliação da Pós-Graduação**. Disponível em: <http://www.capes.gov.br/avaliacao/avaliacao-da-pos-graduacao>. Acesso em: dezembro 2013.

FUNDAÇÃO CAPES. **História e Missão da CAPES**. Disponível em: <http://www.capes.gov.br/sobre-a-capes/historia-e-missao>. Acesso em: dezembro 2013.

FUNDAÇÃO CAPES. **Coleta de Dados 12.0 — Manual do Usuário**. Disponível em: <http://www.capes.gov.br/images/stories/download/coletadados/Manual-do-Usuario_Coleta12_2013.pdf>. Acesso em: dezembro 2013.

FUNDAÇÃO CAPES. **Qualis Periódicos**. Disponível em: <http://www.capes.gov.br/avaliacao/qualis>. Acesso em: dezembro 2013.

HENSCHEN, D.; DAVENPORT, T. **Analytics at Work**: Q&A with Tom Davenport. Disponível em: <http://www.informationweek.com/software/information-management/analytics-at-work-qanda-with-tom-davenport/d/d-id/1085869?>. Acesso em: dezembro 2013.

INSTITUTO DE INFORMÁTICA DA UFRGS. **Página do Instituto de Informática da UFRGS**. Disponível em: <http://www.inf.ufrgs.br/index.php?option=com_content&view=section&layout=blog&id=6&Itemid=59>. Acesso em: dezembro 2013.

KIMBALL, R. **Data Warehouse Toolkit**. 1.ed. São Paulo: Makron Books, 1998.

KIMBALL, R.; ROSS, M. **The Data Warehouse Toolkit**: the definitive guide to dimensional modeling. 3rd.ed. Indianapolis, US: John Wiley & Sons, Inc., 2013.

OLAP Council. **OLAP Council White Paper**. Disponível em: <http://www.symcorp.com/downloads/OLAP_CouncilWhitePaper.pdf>. Acesso em: dezembro 2013.

POWER, D. **A Brief History of Decision Support Systems**. Disponível em: <http://dssresources.com/history/dsshistory.html>. Acesso em: dezembro 2013.

PPGC. **Página do Programa de Pós-Graduação em Computação do Instituto de Informática da UFRGS**. Disponível em: <http://ppgc.inf.ufrgs.br/index.php?option=com_content&view=article&id=49&Itemid=100>. Acesso em: dezembro 2013.

ROUSE, M. **Multidimensional Database (MDB)**. Disponível em: <http://searchoracle.techtarget.com/definition/multidimensional-database>. Acesso em: dezembro 2013.

WIKIPEDIA. **Business Intelligence — Wikipedia, The Free Encyclopedia**. Disponível em: <http://en.wikipedia.org/wiki/Business_intelligence>. Acesso em: dezembro 2013.

APÊNDICE A SCRIPT SQL DE CRIAÇÃO DO DATA WAREHOUSE

```
CREATE SEQUENCE
public.dim_categoria_autor_id_categoria_autor_seq;

CREATE TABLE public.dim_categoria_autor (
id_categoria_autor BIGINT NOT NULL DEFAULT
    nextval('public.dim_categoria_autor_id_categoria_autor_seq'),
categoria VARCHAR(23) NOT NULL,
vinculo VARCHAR(27) NOT NULL,
CONSTRAINT dim_categoria_autor_pk
PRIMARY KEY (id_categoria_autor)
);


ALTER SEQUENCE public.dim_categoria_autor_id_categoria_autor_seq
OWNED BY public.dim_categoria_autor.id_categoria_autor;

CREATE SEQUENCE public.dim_qualis_id_qualis_seq;

CREATE TABLE public.dim_qualis (
id_qualis BIGINT NOT NULL DEFAULT
    nextval('public.dim_qualis_id_qualis_seq'),
classificacao VARCHAR(13) NOT NULL,
tipo VARCHAR(13) NOT NULL,
nome VARCHAR(255) NOT NULL,
CONSTRAINT dim_qualis_pk PRIMARY KEY (id_qualis)
);


ALTER SEQUENCE public.dim_qualis_id_qualis_seq
OWNED BY public.dim_qualis.id_qualis;

CREATE SEQUENCE public.dim_producao_id_producao_seq;

CREATE TABLE public.dim_producao (
id_producao BIGINT NOT NULL DEFAULT
```

```
    nextval('public.dim_producao_id_producao_seq'),
nome TEXT NOT NULL,
idioma VARCHAR(20) NOT NULL,
pais VARCHAR(20) NOT NULL,
nivel_tese VARCHAR(27) NOT NULL,
tipo_producao VARCHAR(60) NOT NULL,
natureza VARCHAR(60) NOT NULL,
linha_pesquisa VARCHAR(255) NOT NULL,
projeto_pesquisa VARCHAR(255) NOT NULL,
CONSTRAINT dim_producao_pk PRIMARY KEY (id_producao)
);


ALTER SEQUENCE public.dim_producao_id_producao_seq
OWNED BY public.dim_producao.id_producao;

CREATE SEQUENCE public.dim_autor_id_autor_seq;

CREATE TABLE public.dim_autor (
id_autor BIGINT NOT NULL DEFAULT
    nextval('public.dim_autor_id_autor_seq'),
nome VARCHAR(60) NOT NULL,
nome_abreviado VARCHAR(40) NOT NULL,
sexo VARCHAR(9) NOT NULL,
pais VARCHAR(20) NOT NULL,
ano_nascimento VARCHAR(13) NOT NULL,
email VARCHAR(255) NOT NULL,
CONSTRAINT dim_autor_pk PRIMARY KEY (id_autor)
);


ALTER SEQUENCE public.dim_autor_id_autor_seq
OWNED BY public.dim_autor.id_autor;

CREATE TABLE public.fato_producao (
ano SMALLINT NOT NULL,
id_producao BIGINT NOT NULL,
id_autor BIGINT NOT NULL,
papel_autor VARCHAR(13) NOT NULL,
id_categoria_autor BIGINT NOT NULL,
ordem_autor SMALLINT NOT NULL,
id_qualis BIGINT NOT NULL,
contador_producao INTEGER DEFAULT 1 NOT NULL,
CONSTRAINT fato_producao_pk
PRIMARY KEY (ano, id_producao, id_autor)
);
```

```
ALTER TABLE public.fato_producao
ADD CONSTRAINT dim_vinculo_fato_producao_fk
FOREIGN KEY (id_categoria_autor)
REFERENCES public.dim_categoria_autor (id_categoria_autor)
ON DELETE NO ACTION
ON UPDATE NO ACTION
NOT DEFERRABLE;

ALTER TABLE public.fato_producao
ADD CONSTRAINT dim_qualis_fato_producao_fk
FOREIGN KEY (id_qualis)
REFERENCES public.dim_qualis (id_qualis)
ON DELETE NO ACTION
ON UPDATE NO ACTION
NOT DEFERRABLE;

ALTER TABLE public.fato_producao
ADD CONSTRAINT dim_producao_fato_producao_fk
FOREIGN KEY (id_producao)
REFERENCES public.dim_producao (id_producao)
ON DELETE NO ACTION
ON UPDATE NO ACTION
NOT DEFERRABLE;

ALTER TABLE public.fato_producao
ADD CONSTRAINT dim_autor_fato_producao_fk
FOREIGN KEY (id_autor)
REFERENCES public.dim_autor (id_autor)
ON DELETE NO ACTION
ON UPDATE NO ACTION
NOT DEFERRABLE;
```

APÊNDICE B PLANILHAS DAS TABELAS DO DATA WAREHOUSE

Nome da Tabela dim_autor
Nome Lógico Dimensao Autor
Tipo da Tabela Dimensão
Descrição Dimensão com informações de pessoas envolvidas nas produções acadêmicas

	Destino					Origem			
Coluna	**Descrição**	**Tipo de Dados**	**Tamanho**	**Exemplos**	**Tipo SCD**	**Tabela Origem**	**Campo Origem**	**Tipo do Campo Origem**	**Regras ETL**
id_autor	Chave primária	bigint		1, 2, 3, ...					Chave substituta
nome	Nome do autor	varchar	60	João da Silva, Maria da Graça, ...	1	COL_PESSOAL	Nome	varchar(60)	Cópia
nome_abreviado	Nome do autor abreviado para referência bibliográfica	varchar	40	SILVA, J.; GRAÇA, M.; ...	1	COL_PESSOAL	NomeAbreviado	varchar(40)	Cópia
sexo	Sexo do autor	varchar	9	Masculino, Feminino	1	COL_PESSOAL	Sexo	char(1)	M=Masculino, F=Feminino
pais	País de origem do autor	varchar	20	Não Informado, Brasil, Canadá, ...	1	COL_PESSOAL	Pais	char(2)	Mapeamento de uma tabela de códigos de países
ano_nascimento	Ano de nascimento do autor	varchar	13	Não Informado,1980, 1984, ...	1	COL_PESSOAL	AnoNascimento	smallint	NULL=Não informado, Não nulo=conversão entre tipos
email	E-mail do autor	varchar	255	joaodasilva@example.com, ...	1	COL_PESSOAL	Email	varchar(255)	Cópia

Figura B.1: Planilha da tabela Dimensao Autor

Nome da Tabela dim_categoria_autor
Nome Lógico Dimensao Categoria Autor
Tipo da Tabela Dimensão
Descrição Dimensão com informações da categoria do autor em relação ao programa de pós-graduação

	Destino					Origem			
Coluna	**Descrição**	**Tipo de Dados**	**Tamanho**	**Exemplos**	**Tipo SCD**	**Tabela Origem**	**Campo Origem**	**Tipo do Campo Origem**	**Regras ETL**
id_categoria_autor	Chave primária	bigint		1, 2, 3, ...					Chave substituta
categoria	Categoria do autor	varchar	23	Docente, Discente, Outro	1	COL_PESSOAL	Categoria	char(1)	D=Docente, I=Discente, O=Outro
vinculo	Vínculo do autor com o programa de pós-graduação	varchar	27	Permanente, Visitante, Doutorado, Outro, ...	1	COL_DOCENTES_IES/ COL_DISCENTES_AUTORES	TipoDocente/ Nivel	char(1)/char(1)	P=Permanente, V=Visitante, C=Colaborador/G=Graduação, M=Mestrado Acadêmico, F=Mestrado Profissionalizante, D=Doutorado/Outro se categoria=O

Figura B.2: Planilha da tabela Dimensao Categoria Autor

Nome da Tabela dim_producao
Nome Lógico Dimensao Producao
Tipo da Tabela Dimensão
Descrição Dimensão com informações de uma produção acadêmica

	Destino					Origem			
Coluna	**Descrição**	**Tipo de Dados**	**Tamanho**	**Exemplos**	**Tipo SCD**	**Tabela Origem**	**Campo Origem**	**Tipo do Campo Origem**	**Regras ETL**
id_producao	Chave primária	bigint							Chave substituta
nome	Nome da produção (título do artigo, tese, etc.)	longvarchar		Design, Automation and Test Conference, ...	1	COL_PRODUCAO	NomeProducao	longvarchar	Cópia
idioma	Idioma em que foi escrita a produção	varchar	20	Português, Inglês, ...	1	G_IDIOMAS	Descricao	varchar(20)	relacionamento entre tabelas COL_PRODUCAO e G_IDIOMAS via COL_PRODUCAO.IdIdioma
pais	País da produção	varchar	20	Brasil, Argentina, Alemanha, ...	1	COL_PRODUCAO	IdPais	char(2)	Mapeamento de uma tabela de códigos de países
nivel_tese	Nível de titulação, caso a produção seja uma tese	varchar	27	Não Aplicável, Mestrado Acadêmico, Mestrado Profissionalizante, Doutorado	1	COL_PRODUCAO	NivelTese	char(1)	NULL=Não Aplicável, M=Mestrado Acadêmico, F=Mestrado Profissionalizante, D=Doutorado
tipo_producao	Tipo da produção, como trabalho em anais, livro, etc.	varchar	60	Organização de Evento, Artigo em Periódico, ...	1	COL_TIPO_PRODUCAO	Descricao	varchar(60)	relacionamento entre tabelas COL_PRODUCAO e COL_TIPO_PRODUCAO via COL_PRODUCAO.IdTipoProducao
natureza	Natureza da produção	varchar	60	Curadoria, Trabalho Completo, ...	1	COL_PRODUCAO_DETALHAMENTO	Conteudo onde Titulo='Natureza'	varchar(60)	relacionamento entre tabelas COL_PRODUCAO e COL_PRODUCAO_DETALHAMENTO via COL_PRODUCAO.IdProducao
linha_pesquisa	Linha de pesquisa onde se encontra a produção	varchar	255	Microeletrônica, Inteligência Artificial, ...	1	COL_LINHAS_PESQUISA	Nome	varchar(255)	relacionamento entre tabelas COL_PRODUCAO e COL_LINHAS_PESQUISA via COL_PRODUCAO.IdLinhaPesquisa
projeto_pesquisa	Projeto de pesquisa em que foi feita a produção	varchar	255	ApproxMatch: Casamento Aproximado de Grandes Volumes de Dados, ...	1	COL_PROJETOS_PESQUISA	Nome	varchar(255)	relacionamento entre tabelas COL_PRODUCAO e COL_PROJETOS_PESQUISA via COL_PRODUCAO.IdProjetoPesquisa

Figura B.3: Planilha da tabela Dimensao Producao

Nome da Tabela dim_qualis
Nome Lógico Dimensao Qualis
Tipo da Tabela Dimensão
Descrição Dimensão com informações da classificação Qualis de um evento ou periódico

	Destino					Origem			
Coluna	**Descrição**	**Tipo de Dados**	**Tamanho**	**Exemplos**	**Tipo SCD**	**Tabela Origem**	**Campo Origem**	**Tipo do Campo Origem**	**Regras ETL**
id_qualis	Chave primária	bigint							Chave substituta
classificacao	Classificação Qualis	varchar	13	A1, A2, B1, B2, ...	2	Arquivo de entrada			Cópia
tipo	Define se a classificação é para evento ou periódico	varchar	13	Periódico, Evento	1				De G_QUALIS_PERIODICOS=Periódico, de G_QUALIS_EVENTOS=Evento
nome	Nome do evento ou periódico	varchar	255	IEEE, Communications of the ACM, ...	1	G_QUALIS_PERIODICOS/ G_QUALIS_EVENTOS	Titulo/NomeEvento	varchar(255)/varchar_ignorecase(255)	Cópia

Figura B.4: Planilha da tabela Dimensao Qualis

Nome da Tabela fato_producao
Nome Lógico Fato Producao
Tipo da Tabela Fato
Descrição Tabela de fatos com medidas para análise da produção do programa de pós-graduação

	Destino					Origem			
Coluna	**Descrição**	**Tipo de Dados**	**Tamanho**	**Exemplos**	**Tipo SCD**	**Tabela Origem**	**Campo Origem**	**Tipo do Campo Origem**	**Regras ETL**
ano	Chave primária e dimensão degenerada indicando o ano da produção	smallint		2010, 2011, 2012, ...	1	COL_PRODUCAO	AnoBase	smallint	Cópia
id_producao	Chave primária e chave estrangeira da dimensão Producao	bigint							
id_autor	Chave primária e chave estrangeira da dimensão Autor	bigint							
papel_autor	Dimensão degenerada indicando a atuação do autor na produção	varchar	13	Autor, Orientador, Co-Orientador	1	COL_R_PRODUCAO_AUTOR/COL_R_ORIENTACOES	Derivado/Tipo Orientacao	Derivado/char(1)	Autor se tabela COL_R_PRODUCAO_AUTOR/O=Orientador, C=Co-Orientador se tabela COL_R_ORIENTACOES
id_categoria_autor	Chave estrangeira da dimensão Categoria Autor	bigint							
ordem_autor	Dimensão degenerada indicando a ordem do autor na citação da produção	smallint		1, 2, 3, ...	1	COL_R_PRODUCAO_AUTOR	OrdemAutor	smallint	Cópia
id_qualis	Chave estrangeira da dimensão Qualis	bigint							
contador_producao	Fato aditivo de contagem de produção	integer		Sempre o valor 1	1				Sempre 1

Figura B.5: Planilha da tabela Fato Producao