

EDGAR MACARI JUNIOR

ANÁLISE DE SUITES DE FERRAMENTAS INTEGRADAS PARA A
CONSTRUÇÃO DE *DATA WAREHOUSES* ESPACIAIS

Florianópolis

Maio de 2010

EDGAR MACARI JUNIOR

ANÁLISE DE SUITES DE FERRAMENTAS INTEGRADAS PARA A
CONSTRUÇÃO DE *DATA WAREHOUSES* ESPACIAIS

Trabalho de Conclusão de Curso
apresentado como requisito parcial para
obtenção do grau de Bacharel em
Ciências da Computação na
Universidade Federal de Santa Catarina.

Orientação:

Prof. Dr. Renato Fileto

UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO
DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA
CIÊNCIAS DA COMPUTAÇÃO

Florianópolis

Maio de 2010

EDGAR MACARI JUNIOR

ANÁLISE DE SUITES DE FERRAMENTAS INTEGRADAS PARA A
CONSTRUÇÃO DE *DATA WAREHOUSES* ESPACIAIS

Proposta de Trabalho de Conclusão de Curso
aprovada pelo professor orientador.

Orientação:
Prof. Dr. Renato Fileto

Florianópolis
Maio de 2010

Resumo

O armazenamento e a manipulação de dados espaciais em *data warehouses* (*data warehouses* espaciais) é um tema de pesquisa e desenvolvimento com vários pontos ainda em aberto, particularmente no que diz respeito a ferramentas para a construção de *data warehouses* espaciais. Com o recente surgimento das primeiras ferramentas que oferecem suporte a dados espaciais em *data warehouses*, um estudo dessas ferramentas faz-se necessário com o intuito de verificar como é feito tal suporte e quais características são suportadas. O objetivo deste trabalho é analisar tais ferramentas baseando-se na sua documentação e na implementação de um *data warehouse* com cada uma delas, finalizando com um quadro comparativo entre elas. Este estudo foi desenvolvido no Laboratório para Integração de Sistemas e Aplicações Avançadas (LISA), em parceria com a Empresa de Pesquisa Agropecuária e Extensão Rural de Santa Catarina (EPAGRI).

Palavras-Chave: *Data warehouse* espacial, dados espaciais, OLAP, *Microsoft Analysis Services*, *PentahoBI*, *SpagoBI*.

Abstract

The storage and the handling of spatial data in data warehouses (spatial data warehouses) is a research and development theme with various open issues, particularly with regard to tools for building spatial data warehouses. With the emergence of the first tools that support spatial data in data warehouses, a study of these tools is necessary in order to determine how this support is made and what features are supported. The aim of this study is to analyze these tools based on their documentation and in the implementation of a data warehouse with each one of them. This comparison ends with a comparative table of the available tools. This study was developed at the Laboratory for Systems Integration and Advanced Applications (LISA), in partnership with the Company for Agricultural Research and Rural Extension of Santa Catarina (EPAGRI).

Keywords: Spatial data warehouse, spatial data, OLAP, Microsoft Analysis Services, PentahoBI, SpagoBI.

Sumário

Resumo.....	4
<i>Abstract</i>	5
Sumário	6
Lista de Figuras.....	8
Lista de Quadros.....	10
1. Introdução.....	11
1.1 Contextualização	11
1.2 Motivação	12
1.3 Objetivos	12
1.3.1 Objetivos Gerais	12
1.3.2 Objetivos Específicos.....	12
1.4 Justificativa.....	13
1.5 Pesquisa Bibliográfica.....	13
1.5.1 Análise da documentação disponível.....	14
1.5.2 Testes de execução	14
1.5.3 Prototipação de um DWE Agrícola nas ferramentas estudadas	14
2. Fundamentos	15
2.1 Sistemas de Informações Geográficas	15
2.1.1 Dados Espaciais	15
2.1.1.1 Dados Geográficos.....	16
2.1.1.2 Especificação de Dados Espaciais.....	17
2.1.2 Operadores Geográficos	22
2.2 <i>Data warehouse</i>	27
2.2.1 Modelo Dimensional	28
2.2.2 Abordagens para a construção de <i>Data Warehouses</i>	32
2.2.3 <i>Online Analytical Processing</i>	32
2.3 <i>Data warehouses</i> espaciais	35
3. <i>Data warehouse</i> Espacial para a EPAGRI	38
4. Análise das ferramentas.....	41
4.1 <i>SQL Server</i> 2008.....	41
4.1.1 Suporte a Dados Espaciais.....	43
4.1.2 Suporte a Operadores Espaciais.....	44

4.1.3	Implementação de Um Data Warehouse Espacial.....	45
4.2	<i>PentahoBI Platform</i>	49
4.2.1	Suporte a Dados Espaciais.....	50
4.2.2	<i>GeoKettle</i>	50
4.2.3	Suporte a Operadores Espaciais.....	51
4.2.4	Implementação de Um Data Warehouse Espacial.....	52
4.3	<i>SpagoBI</i>	56
4.3.1	Suporte a Extensões Espaciais.....	57
4.3.2	Implementação de Um Data Warehouse Espacial.....	58
4.4	Critérios de Análise das Ferramentas e Comparativo.....	58
5.	Trabalhos Relacionados	62
6.	Conclusões e Trabalhos Futuros	64
	REREFÊNCIAS BIBLOGRÁFICAS	65

Lista de Figuras

FIGURA 1: EXEMPLO DE REPRESENTAÇÃO CARTESIANA DAS MESORREGIÕES DE SANTA CATARINA	17
FIGURA 2: EXEMPLO DE PONTO REPRESENTADO EM DUAS DIMENSÕES.....	18
FIGURA 3: UM MULTIPONTO REPRESENTADO EM DUAS DIMENSÕES.....	18
FIGURA 4: (A) LINESTRING SIMPLES; (B) LINESTRING NÃO-SIMPLES; (C) LINEARRING (LINESTRING SIMPLES E FECHADO); (D) LINESTRING NÃO-SIMPLES E FECHADO.....	19
FIGURA 5: (A)MULTILINESTRING SIMPLES; (B) MULTILINESTRING NÃO-SIMPLES COM DOIS ELEMENTOS; (C) MULTILINESTRING NÃO SIMPLES E FECHADO COM DOIS ELEMENTOS.....	20
FIGURA 6: EXEMPLOS DE FIGURAS QUE NÃO SÃO REPRESENTÁVEIS COMO UMA INSTÂNCIA SIMPLES DE UM POLÍGONO.....	21
FIGURA 7: EXEMPLOS DE MULTIPOLÍGONOS (AB), (B), (C) E (D) COM UM, DOIS, TRÊS E QUATRO POLÍGONOS RESPECTIVAMENTE.....	22
FIGURA 8: EXEMPLO DE OBJETOS QUE NÃO SÃO REPRESENTÁVEIS POR UMA SIMPLES INSTÂNCIA DE UM MULTIPOLÍGONO.....	22
FIGURA 9: CARACTERÍSTICAS DE UMA REGIÃO: LIMITE, INTERIOR, EXTERIOR.....	23
FIGURA 10: EXEMPLO DE OPERAÇÃO ENTRE DUAS REGIÕES.....	24
FIGURA 11: EXEMPLO DA RELAÇÃO DE IGUALDADE ENTRE DOIS ELEMENTOS.....	24
FIGURA 12: EXEMPLO DA RELAÇÃO DE DISJUNÇÃO ENTRE DOIS ELEMENTOS.....	25
FIGURA 13: EXEMPLO DA RELAÇÃO DE TOQUE ENTRE DOIS ELEMENTOS.....	25
FIGURA 14: EXEMPLO DA RELAÇÃO DE ATRAVESSAMENTO ENTRE DOIS ELEMENTOS.....	25
FIGURA 15: EXEMPLO DA RELAÇÃO DENTRO ENTRE DOIS ELEMENTOS.....	26
FIGURA 16: EXEMPLO DA RELAÇÃO DE SOBREPOSIÇÃO ENTRE DOIS ELEMENTOS.....	26
FIGURA 17: EXEMPLO DA RELAÇÃO CONTÉM.....	27
FIGURA 18: EXEMPLO DA RELAÇÃO DE INTERSECÇÃO ENTRE DOIS ELEMENTOS.....	27
FIGURA 19: EXEMPLO DE ESQUEMA ESTRELA. UMA ÚNICA TABELA FATO E VÁRIAS DIMENSÕES.....	29
FIGURA 20: EXEMPLO DE ESQUEMA FLOCO-DE-NEVE. MAIS DE UMA TABELA FATO, COM DIMENSÕES COMPARTILHADAS E CERTO NÍVEL DE NORMALIZAÇÃO.....	30
FIGURA 21: EXEMPLO DE CUBO OLAP REPRESENTANDO A PRODUÇÃO DE ALGUMAS CULTURAS EM MUNICÍPIOS AO LONGO DOS ÚLTIMOS QUATRO ANOS, BASEADO DO MODELO DIMENSIONAL DA FIGURA 19.....	31
FIGURA 22: OPERAÇÃO DE <i>DRILL-DOWN</i>	33
FIGURA 23: OPERAÇÃO DE <i>ROLL-UP</i>	34
FIGURA 24: OPERAÇÃO <i>PIVOTING</i>	34
FIGURA 25: OPERAÇÃO <i>SLICE AND DICE</i>	35
FIGURA 26: EXEMPLO DE DIMENSÃO NÃO ESPACIAL.....	36
FIGURA 27: EXEMPLO DE DIMENSÃO ESPACIAL PARA NÃO-ESPACIAL.....	37
FIGURA 28: EXEMPLO DE DIMENSÃO ESPACIAL PARA ESPACIAL.....	37
FIGURA 29: MODELO DIMENSIONAL DO <i>DATA MART</i> “PRODUÇÃO AGRÍCOLA”.....	39
FIGURA 30: ESQUEMA DA ORGANIZAÇÃO DA SUÍTE DE FERRAMENTAS DA <i>MICROSOFT</i>	41

FIGURA 31: ARQUITETURA DOS TIPOS DE DADOS GEOGRÁFICOS DO <i>MICROSOFT SQL SERVER 2008</i> . OS TIPOS DE DADOS QUE PODEM SER INSTANCIADOS ESTÃO INDICADOS EM AZUL.....	44
FIGURA 32: CRIAÇÃO DE UM NOVO PROJETO DO <i>ANALYSIS SERVICES</i> NO <i>MICROSOFT SQL SERVER BUSINESS INTELLIGENCE DEVELOPMENT STUDIO</i>	45
FIGURA 33: CONFIGURAÇÃO DE UMA NOVA CONEXÃO COM BANCO DE DADOS NO <i>MICROSOFT SQL SERVER BUSINESS INTELLIGENCE DEVELOPMENT STUDIO</i>	46
FIGURA 34: CONFIGURAÇÃO DE UMA NOVA FONTE DE DADOS NO <i>MICROSOFT SQL SERVER BUSINESS INTELLIGENCE DEVELOPMENT STUDIO</i>	47
FIGURA 35: DEFINIÇÃO DE UMA <i>VIEW</i> SOBRE A FONTE DE DADOS CRIADA E MENSAGEM DE ERRO INFORMANDO TIPOS INCOMPATÍVEIS DE DADOS ESPACIAIS UTILIZADOS PELO <i>ORACLE</i> E PELA <i>MICROSOFT</i>	48
FIGURA 36: ARQUITETURA DA PLATAFORMA <i>PENTAHO</i> (<i>PENTAHO, 2006</i>).....	49
FIGURA 37: CRIAÇÃO DE UMA NOVA TRANSFORMAÇÃO NO <i>KETTLE</i>	52
FIGURA 38: CONFIGURAÇÃO DE UMA NOVA CONEXÃO COM O BANCO DE DADOS NO <i>KETTLE</i>	53
FIGURA 39: SELECIONANDO OS CAMPOS A SEREM EXTRAÍDOS DA TABELA <i>MUNICIPIO</i>	54
FIGURA 40: EXEMPLO DE TRANSFORMAÇÃO NO <i>KETTLE</i> . DADOS ESPACIAIS DE TRÊS TABELAS SÃO EXTRAÍDOS E AGRUPADOS.	55
FIGURA 41: ARQUIVO XML COM COMPONENTE ESPACIAL EM BRANCO.....	55
FIGURA 42: ARQUITETURA DA PLATAFORMA <i>SPAGOBI</i> (<i>SPAGOBI, 2009</i>).	56
FIGURA 43: ARQUITETURA DO <i>SPAGOBI SERVER</i> (<i>SPAGOBI, 2009</i>).	57

Lista de Quadros

QUADRO 1: COMPARATIVO ENTRE OS MODELOS RELACIONAL E DIMENSIONAL.	31
QUADRO 2: QUADRO COMPARATIVO ENTRE AS FERRAMENTAS ESTUDADAS.	58
QUADRO 3: COMPARATIVO ENTRE OS TIPOS DE DADOS ESPACIAIS SUPOSTADOS.	60
QUADRO 4: COMPARATIVO ENTRE OS OPERADORES ESPACIAIS SUPOSTADOS.	61

1. Introdução

1.1 Contextualização

Um *data warehouse* (DW) é um repositório de dados centralizado, que integra grandes volumes de dados provenientes de diferentes fontes e formatos, mas organizados segundo o modelo dimensional (DAMIANI, 2006). *Data warehouses* permitem aos seus usuários extrair informações consolidadas para avaliar o comportamento de um empreendimento e tomar decisões estratégicas.

Um *data warehouse* possui quatro características básicas (INMON 1997): organizado por assunto, integrado, variável em relação ao tempo e não volátil. Ser organizado por assunto significa dispor as informações de acordo com diferentes dimensões ou categorias de assunto (por exemplo: espaço, tempo, produto) e possuir subconjuntos, chamados *data marts*, voltados para o atendimento às necessidades específicas de análise de dados. Ser integrado significa que os dados de fontes heterogêneas (*e.g.*, sistemas transacionais) são modificados e convertidos para um formato uniforme, de modo a permitir a carga no *data warehouse*. Variável em relação ao tempo significa que somente um subconjunto dos dados das fontes, referentes uma determinada janela de tempo, são carregados no *data warehouse*. Ser não volátil determina que os dados carregados no *data warehouse* ficam disponíveis para o usuário apenas para consulta, não para alteração.

Um *data warehouse* espacial (DWE) possui as características mencionadas acima e tem a particularidade de possuir dados espaciais, além de dados convencionais (não espaciais). Estes dados espaciais podem estar tanto na tabela de fatos, quando nas dimensões, dependendo da modelagem do problema (HAN, 1998). Extensões espaciais permitem apresentar resultados do processamento dos dados de um *data warehouse* na forma de mapas e aplicar operadores geográficos (*e.g.*, interseção de polígonos representando entidades geográficas como áreas de terreno com certas características) para o cálculo desses resultados.

1.2 Motivação

A motivação deste trabalho encontra-se no fato de que até pouco tempo nenhuma *suíte* de aplicativos oferecia suporte a dados espaciais em *data warehouses*. O surgimento das primeiras ferramentas que se propõem a esta tarefa torna fundamental um estudo com o intuito de verificar como esse suporte está sendo efetuado e quais são os recursos oferecidos por tais ferramentas. Este estudo está sendo desenvolvido no Laboratório para Integração de Sistemas e Aplicações Avançadas (LISA), em parceria com a Empresa de Pesquisa Agropecuária e Extensão Rural de Santa Catarina (EPAGRI).

1.3 Objetivos

1.3.1 Objetivos Gerais

O objetivo geral deste trabalho é realizar um estudo de como as novas ferramentas que suportam dados espaciais para a construção de *data warehouses* estão oferecendo este suporte. As suítes analisadas são *Microsoft Analysis Services 2008*, *PentahoBI*, *SpagoBI*, *GeoBI* e *GOLAPA*, sendo dada preferência a *suítes* que permitam acesso ao *software* para a realização de experimentos na área de agricultura.

1.3.2 Objetivos Específicos

Os objetivos específicos deste trabalho são:

- Estudar a documentação disponível dessas ferramentas;
- Comparar as ferramentas selecionadas segundo diversos critérios, incluindo:
 - linguagens utilizadas na sua configuração e programação das aplicações;
 - forma de acoplamento entre os módulos;
 - tecnologia utilizada na interface com o usuário;
 - aderência a padrões e protocolos para o intercâmbio e tratamento de dados espaciais;

- tipos de extensões espaciais suportadas, incluindo tipos de dados e operadores;
- Analisar as ferramentas selecionadas em execução;
- Implementar *data warehouses* espaciais sobre algumas dessas ferramentas, começando pelo *Microsoft Analysis Services 2008*, para demonstrar a utilização de extensões espaciais na área agrícola, utilizando dados e aplicações da EPAGRI.

1.4 Justificativa

O caráter temporal dos *data warehouses* permite análises de dados de grande impacto para o apoio a tomada de decisão. Atualmente, diversas fontes de dados dos mais variados tipos, sejam elas estruturadas ou não, possuem dados espaciais. São endereços de clientes, informações sobre rodovias, dados referentes à agricultura, entre muitos outros. Com a dinâmica cada vez mais acelerada dos processos corporativos e da sociedade em geral, a demanda por tomadas de decisão em tempo cada vez menor apenas aumenta.

O uso de dados espaciais em *data warehouses* pode possuir as mais diversas aplicações, como por exemplo, dar foco a ações de marketing de uma empresa, localizar o público alvo de determinado produto, questões estratégicas e de planejamento relativas à agricultura ou ainda aplicações na área de saúde para mapear a propagação de determinada doença. Com este cenário, a utilização de dados espaciais em *data warehouses* torna-se imprescindível.

1.5 Pesquisa Bibliográfica

Primeiramente foi realizada uma pesquisa bibliográfica com o intuito de levantar informações sobre modelos de *data warehouses* espaciais, operadores geográficos, funções de agregação de dados espaciais e *suítes* de ferramentas integradas disponíveis para a construção de *data warehouses* espaciais. O objetivo desta pesquisa é melhor compreender os conceitos teóricos fundamentais de *data warehouses* espaciais e definir parâmetros para a análise das *suítes* selecionadas para estudo.

1.5.1 Análise da documentação disponível

Foi coletada documentação sobre as *suítes* de ferramentas disponíveis para a construção de *data warehouses* espaciais. A documentação de cada ferramenta foi analisada para efetuar uma primeira análise das ferramentas, no que diz respeito aos requisitos e parâmetros de análise definidos na fase de pesquisa bibliográfica.

1.5.2 Testes de execução

As ferramentas selecionadas, a partir da análise das funcionalidades e características não funcionais descritas em sua documentação, foram instaladas e testadas, inicialmente com os próprios exemplos e estudos de caso que as acompanham.

1.5.3 Prototipação de um DWE Agrícola nas ferramentas estudadas

Para analisar as características do suporte a dados espaciais de cada ferramenta com base em necessidades reais do nosso grupo de pesquisa e da Epagri, foi criado um *data warehouse* espacial para o domínio agrícola, utilizando sistemas da Epagri como fontes de dados. Foram executados testes com extensões espaciais tanto na tabela de fatos quanto em algumas dimensões de análise de dados.

2. Fundamentos

2.1 Sistemas de Informações Geográficas

Sistemas de Informações Geográficas (SIG) permitem a captura, modelagem, manipulação, recuperação, análise e apresentação de dados georreferenciados (WORBOYS, 1995), ou seja, dados referentes a objetos e fenômenos cuja localização geográfica é essencial à informação e indispensável para analisá-la (CÂMARA, 1996). Estes sistemas permitem que usuários criem consultas (*queries*), submetam-nas e analisem os resultados interativamente e editem os mapas resultantes. As aplicações de SIG são muito variadas, incluindo desde a cartografia a estudos de impacto ambiental ou de prospecção de recursos, ações de *marketing*. Tais sistemas que manipulam componentes espaciais constituem o que se pode designar por Sistemas Espaciais de Apoio à Decisão.

2.1.1 Dados Espaciais

Segundo (CÂMARA, 1996), dados espaciais são dados que fazem referência a quaisquer objetos ou fenômenos associados a alguma dimensão espacial. Portanto, dados espaciais podem ser, por exemplo, representação de estruturas moleculares de compostos ou, no escopo deste trabalho, representação de aspectos geográficos de uma região do planeta.

Devido a essa grande abrangência dos dados espaciais, torna-se necessário existir diversas formas de representar e descrever esses dados. Os tipos que serão abordados nesse trabalho são os dados geométricos, que são representações de figuras geométricas complexas como pontos e polígonos; e os dados geográficos, que representam fenômenos e objetos sobre a Terra,

sempre associados a uma localização, num determinado período de tempo (CÂMARA, 1996).

2.1.1.1 Dados Geográficos

A representação de informação em um plano bidimensional é feita através de um tipo de dados chamado “dado geométrico”, utilizando diversos tipos de unidades de medida, como polegadas, milhas, *pixels*, entre outros (CASANOVA, 2005). Este tipo de dado pode ser usado conjuntamente com alguma projeção planar do mundo real, como um mapa ou um diagrama. Por exemplo, um conjunto de pontos utilizados para alimentar um *data warehouse* sobre as localizações das sedes de propriedades agrícolas. Outros objetos espaciais, como as parcelas de terra ocupadas por propriedades agrícolas, municípios, estados e regiões, podem ser cada qual representados por polígono ou múltiplos polígonos (no caso de parcelas de terra descontínuas, como aquelas que incluem ilhas). Com estes elementos definidos, podem-se executar operações como intersecção, união, verificar se um polígono está contido no outro, verificar se um polígono tangencia outro ou ainda se um polígono é interceptado por outro, entre outras.

A principal característica deste tipo de dado é que ele funciona com unidades uniformes de dados. O número de unidades no topo do plano é exatamente o mesmo que na base do plano, e o número de unidades no lado esquerdo do plano é o mesmo que no lado direito, como um plano cartesiano.

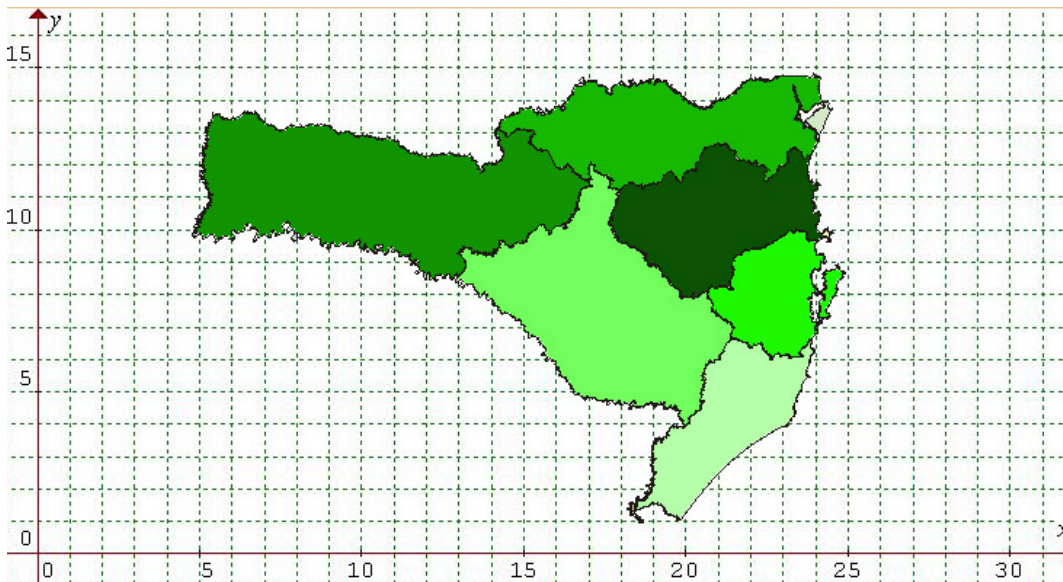


Figura 1: Exemplo de representação cartesiana das mesorregiões de Santa Catarina.

Como foi descrito na seção 2.1.1, dados geográficos estão associados a uma localização na superfície terrestre, num determinado período de tempo. A temporalidade dos dados geográficos possibilita o estudo de diversos estados geográficos históricos, o que permite fazer análises aprofundadas nos aspectos espaço e tempo, como identificar algum padrão de comportamento e fazer projeções futuras (RIBEIRO JUNIOR, 2003).

2.1.1.2 Especificação de Dados Espaciais

O uso e a implementação de elementos espaciais são regulamentados pelo *Open Geospatial Consortium* (OGC), uma organização sem fins lucrativos, internacional, voluntária, responsável por definir padrões de consenso e normas para serviços geoespaciais baseados em localização (OGC 2006). Abaixo seguem algumas definições do OGC, baseadas em definições SFS (*Simple Features Specification*), dos objetos espaciais que serão utilizados nesse trabalho:

GeometryCollection: é um objeto geométrico que é uma coleção de outros objetos geométricos, de quaisquer tipos, desde que pertencentes ao mesmo Sistema de Referência Espacial (*Spatial Reference System*).

Ponto (Point): um ponto é um objeto geométrico adimensional que representa uma localização simples em um sistema de coordenadas. Um ponto possui um valor de coordenada x e um valor de coordenada y . Adicionalmente,

no caso de um espaço com mais de duas dimensões, um ponto pode também apresentar valores de coordenadas z e m . O limite de um ponto é sempre um conjunto vazio.

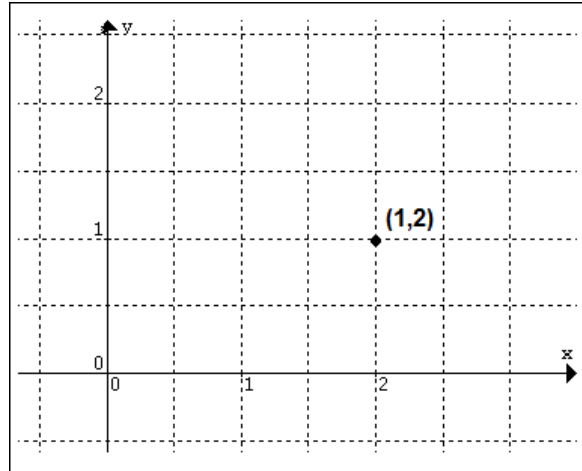


Figura 2: Exemplo de Ponto representado em duas dimensões.

Multiponto (*Multipoint*): um multiponto é um objeto geométrico composto apenas por pontos. Estes pontos não estão ligados entre si e não possuem nenhum tipo de ordenação semanticamente importante. Um Multiponto é considerado simples se nenhum ponto pertencente ao multiponto em questão for igual ao outro, ou seja, possui valores idênticos de X e Y . O limite de um multiponto é sempre um conjunto vazio.

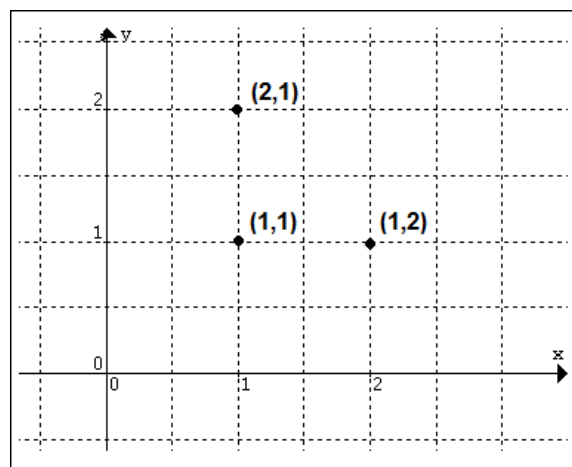


Figura 3: Um Multiponto representado em duas dimensões.

Curva (Curve): uma curva é um objeto geométrico unidimensional que é uma imagem homomórfica do intervalo real e fechado. Uma curva é considerada simples se ela não passa duas vezes pelo mesmo ponto. Uma curva é dita fechada se o seu ponto inicial é igual ao seu ponto final.

Caso esta curva seja simples e fechada, ela é chamada de Anel (*Ring*). O limite de uma curva fechada é sempre um conjunto vazio, mas, o limite de uma curva não fechada são os seus dois pontos extremos (pontos finais).

LineString, Line e LinearRing: estes três conceitos são apresentados ao mesmo tempo, pois são fortemente relacionados. Um *LineString* é uma curva, porém com interpolação linear entre pontos. Cada par de pontos consecutivos define um segmento de linha (*Line*). Uma linha (*Line*) pode ser considerada como um *LineString* com exatamente dois pontos. Um *LinearRing* é um *LineString* que é fechado e simples ao mesmo tempo. A figura 4 ilustra como é cada um destes tipos de objetos geométricos:

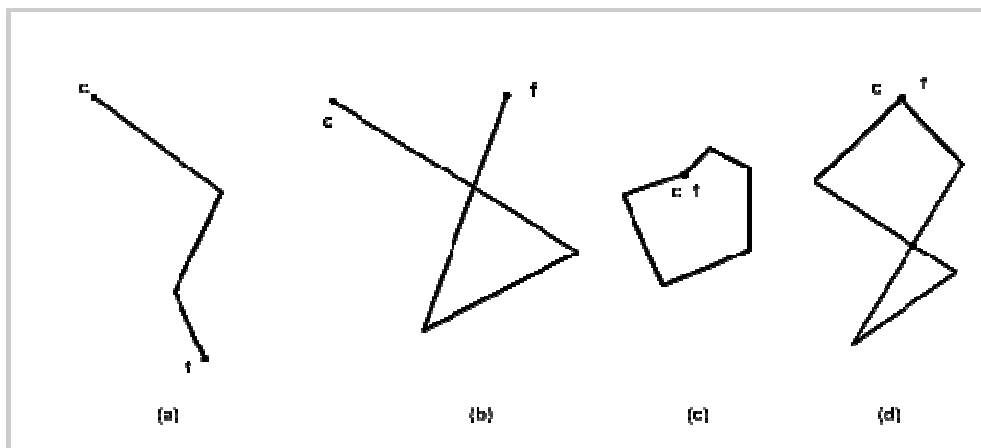


Figura 4: (a) *LineString* simples; (b) *LineString* não-simples; (c) *LinearRing* (*LineString* simples e fechado); (d) *LineString* não-simples e fechado.

MultiCurva (MultiCurve): uma MultiCurva é uma coleção unidimensional cujos elementos são curvas. Um objeto MultiCurva é simples se e somente se todos os seus elementos forem simples e as únicas intersecções entre dois elementos quaisquer ocorrerem entre pontos que estão no limite dos dois elementos. Um ponto está no limite de uma MultiCurva se ele estiver nos limites de um número ímpar de elementos da MultiCurva. O limite de um objeto

MultiCurva é um conjunto vazio, e, um objeto desde grupo só é dito fechado se todos os seus elementos forem fechados.

MultiLineString: um *MultiLineString* é uma MultiCurva cujos elementos são *LineStrings*. O limite de um *MultiLineString* são seus pontos extremos, caso ele não seja fechado. Caso seja fechado, seu limite é um conjunto vazio, conforme ilustra a figura 5:

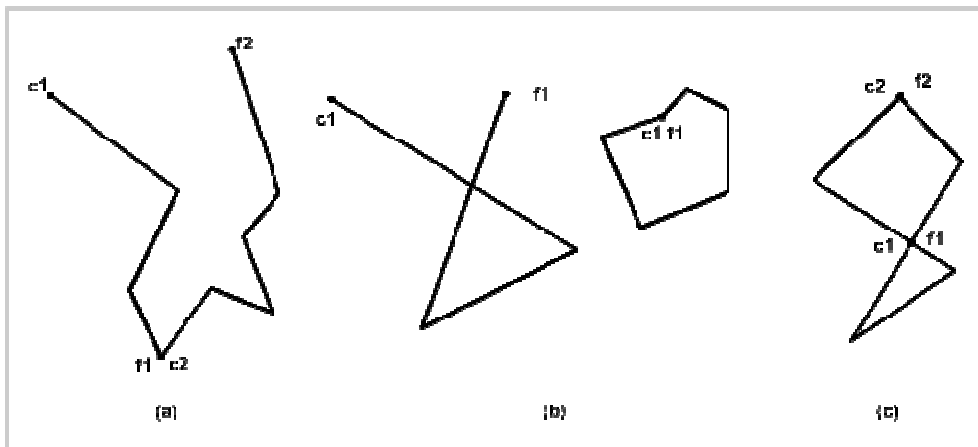


Figura 5: (a) *MultiLineString* simples; (b) *MultiLineString* não-simples com dois elementos; (c) *MultiLineString* não simples e fechado com dois elementos.

Superfície (Surface): uma superfície é um objeto geométrico bidimensional. Uma superfície que é dita simples consiste de um único *patch* (pedaço de superfície único e contínuo) que está associado com um limite exterior de zero ou mais limites interiores. Superfícies poliédricas são formadas pela justaposição de *patches*. Estas superfícies podem não ser inteiramente planares, dependendo da orientação de seu referencial. O limite de uma superfície simples é um conjunto fechado de curvas.

Polígono (Polygon): um Polígono é uma superfície planar definida por um limite exterior e zero ou mais limites interiores, e, cada limite interior define um orifício no Polígono. Todo polígono deve obedecer a certas condições, para ser classificado como tal:

- Polígonos são topologicamente fechados;
- Os limites, externo e internos, de polígonos consistem em um conjuntos de *LinearRings*;

- Não existem dois *LinearRings* que se cruzam no limite do polígono e, podem possuir uma intersecção em um ponto, mas apenas tangenciando;
- Um polígono não pode ter linhas cortadas, pontas ou cavidades;
- O interior de todo polígono é um conjunto de pontos conectados;
- O exterior de um polígono com um ou mais orifícios não é contínuo. Cada orifício define um componente conectado do exterior do polígono.

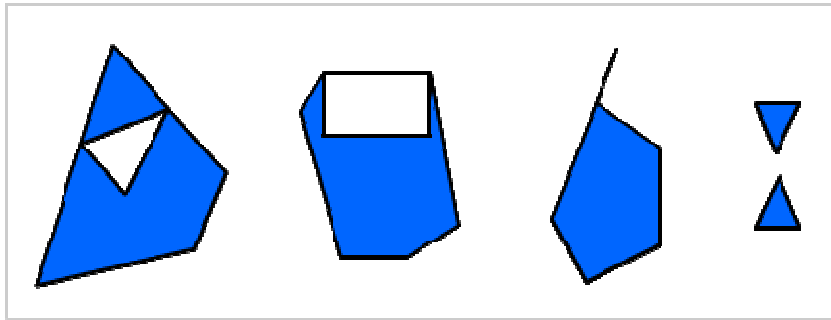


Figura 6: exemplos de figuras que não são representáveis como uma instância simples de um polígono.

MultiSuperfície (*MultiSurface*): uma multisuperfície é um *GeometryCollection* bidimensional cujos elementos são superfícies que usam o mesmo sistema de coordenadas.

MultiPolígono (*MultiPolygon*): um multipolígono é uma multisuperfície cujos elementos são polígonos. Para ser considerado um multipolígono, um objeto espacial deve cumprir com as seguintes condições:

- Os interiores de dois polígonos pertencentes a um multipolígono não devem possuir uma intersecção;
- Os limites de quaisquer dois polígonos pertencentes a um multipolígono não podem se cruzar, podem apenas tocarem-se em um número finito de pontos;
- Um multipolígono é topologicamente fechado;
- Um multipolígono não pode ter linhas cortadas, pontas ou cavidades;
- Um multipolígono é um conjunto fechado e regular de pontos.

O interior de um multipolígono que possui mais de um polígono não é contínuo. O número de componentes contínuos n por interior de um multipolígono é igual ao número de polígonos dentro do multipolígono.

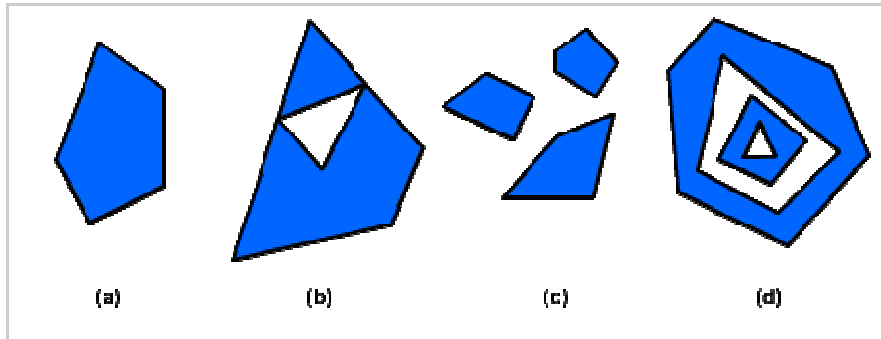


Figura 7: Exemplos de multipolígonos (a), (b), (c) e (d) com um, dois, três e quatro polígonos respectivamente.

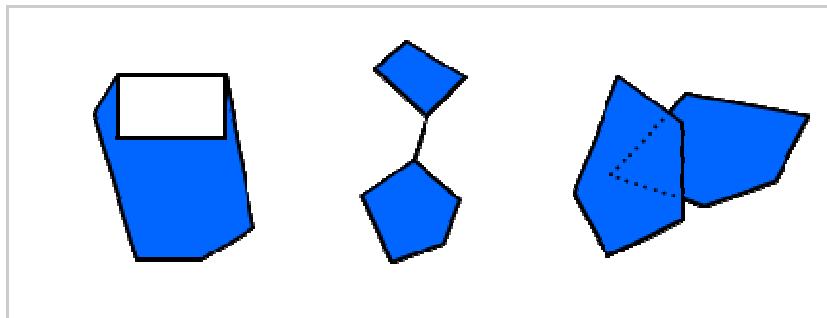


Figura 8: Exemplo de objetos que não são representáveis por uma simples instância de um multipolígono.

2.1.2 Operadores Geográficos

Segundo a OGC, operadores geográficos são métodos booleanos que denotam a existência de alguma relação espacial entre dois elementos. Os dados espaciais devem ser projetados em um sistema de coordenadas bidimensional que representa a superfície terrestre, para então serem comparados.

Uma das principais formalizações de operadores geográficos é a que foi apresentada por Egenhofer (1992). Neste trabalho são apresentadas as operações passíveis de serem aplicadas entre duas regiões bidimensionais A e B, baseadas na comparação de características como interior, limite (borda) e exterior. Estas características são representadas pela seguinte notação:

- Interior de uma região A: A° ;
- Limite de uma região A: ∂A ;
- Exterior de uma região A: A^- .

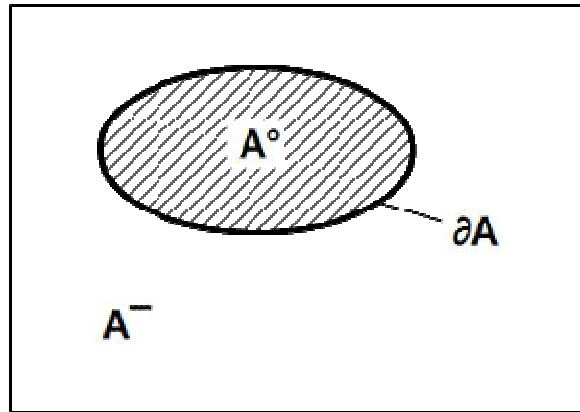


Figura 9: Características de uma região: limite, interior, exterior.

Baseando-se nestas notações, as seguintes definições de operadores geográficos podem ser formalizadas:

- A intersecção do interior de A com o interior de B: $(A^\circ \cap B^\circ)$;
- A intersecção do interior de A com o limite de B: $(A^\circ \cap \partial B)$;
- A intersecção do interior de A com o exterior de B: $(A^\circ \cap B^-)$;
- A intersecção dos limites de A e B: $(\partial A \cap \partial B)$;
- A intersecção do limite de A com o interior de B: $(\partial A \cap B^\circ)$;
- A intersecção do limite de A com o exterior de B: $(\partial A \cap B^-)$;
- A intersecção dos exteriores de A e B: $(A^- \cap B^-)$;
- A intersecção do exterior de A com o limite de B: $(A^- \cap \partial B)$;
- A intersecção do exterior de A com o interior de B: $(A^- \cap B^\circ)$.

Para facilitar a representação, foi proposta (Egenhofer, 1992) uma representação matricial para as relações entre duas regiões A e B:

$$R(A, B) = \begin{pmatrix} (A^\circ \cap B^\circ) & (A^\circ \cap \partial B) & (A^\circ \cap B^-) \\ (\partial A \cap B^\circ) & (\partial A \cap \partial B) & (\partial A \cap B^-) \\ (A^- \cap B^\circ) & (A^- \cap \partial B) & (A^- \cap B^-) \end{pmatrix}$$

Onde cada elemento da matriz pode assumir dois valores:

- \emptyset : vazio. As duas regiões não possuem tal relação.
- $\neg \emptyset$: não-vazio. As duas regiões possuem tal relação.

Na figura 10 segue um exemplo para ilustrar tal notação:

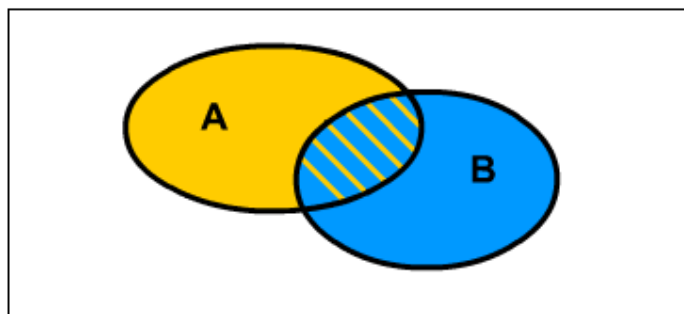


Figura 10: exemplo de operação entre duas regiões.

Nesta relação entre A e B, existe intersecção entre seus interiores, limites e exteriores em todas as combinações possíveis, assim podendo ser representado matricialmente por:

$$R(A, B) = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

Com esta análise do interior, intersecção e exterior de dois objetos, podem-se expressar os operadores especificados pela OGC que são (OGC 2006):

Equals (ou igual): dois elementos “a” e “b” são iguais se “a” está contido em “b” e “b” está contido em “a”, conforme a figura 11:

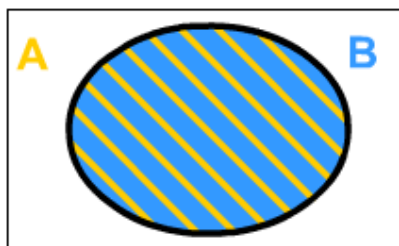


Figura 11: Exemplo da relação de igualdade entre dois elementos.

Disjoint (ou disjunto): dois elementos “a” e “b” são disjuntos se a intersecção entre eles é vazia, conforme a figura 12:

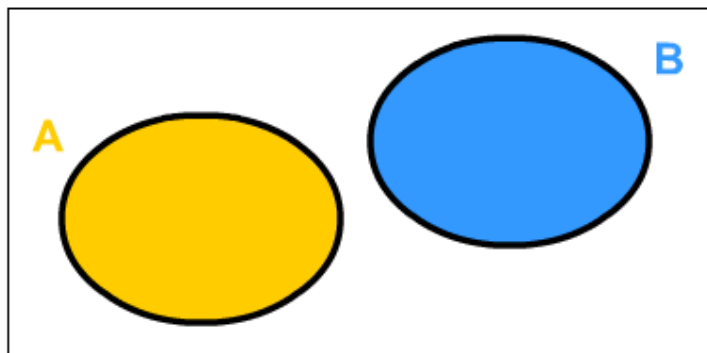


Figura 12: Exemplo da relação de disjunção entre dois elementos.

Touches (ou toca): dois elementos “a” e “b” tocam-se quando a intersecção entre eles é vazia e a intersecção de seus interiores também é vazia, conforme a figura 13:

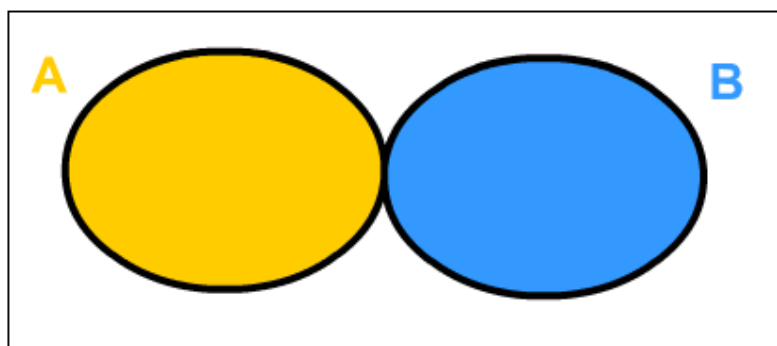


Figura 13: Exemplo da relação de toque entre dois elementos.

Crosses (ou atravessa): dois elementos “a” e “b” cruzam-se quando a intersecção deles resulta em uma geometria cujas dimensões são menores do que a dimensão máxima das instâncias do elemento de origem e o conjunto de intersecções é interior a ambas as instâncias do elemento de origem, conforme a figura 14:

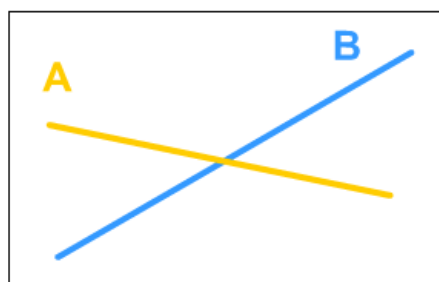


Figura 14: Exemplo da relação de atravessamento entre dois elementos.

Within (ou dentro): um elemento “a” está dentro de um elemento “b” quando a intersecção entre eles é igual a “a” e a intersecção entre o interior de “a” e o exterior de “b” é vazia, conforme a figura 15:

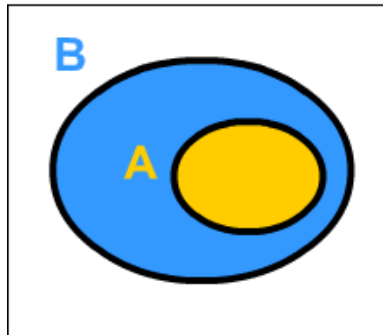


Figura 15: Exemplo da relação dentro entre dois elementos.

Overlaps (ou sobrepõe): elementos “a” e “b” sobrepõem-se se a região que representa a intersecção entre elas tiver a mesma dimensão que as instâncias e as instâncias não forem iguais, conforme a figura 16:

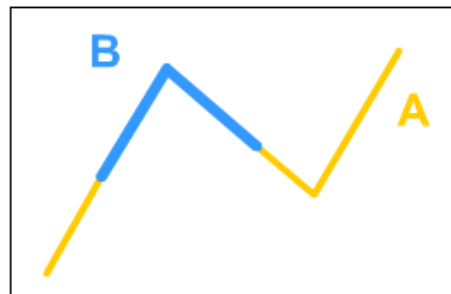


Figura 16: Exemplo da relação de sobreposição entre dois elementos.

Contains (ou contém): um elemento “b” contém um elemento “a” quando “a” está dentro de “b”, conforme mostrado na figura 17;

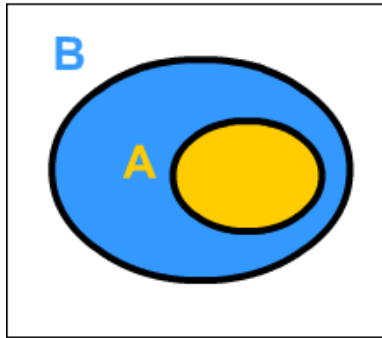


Figura 17: Exemplo da relação contém.

Intersects (ou intercepta): um elemento “a” intercepta um elemento “b” se “a” não é disjuncto de “b”, conforme explicitado anteriormente, conforme a figura 18:

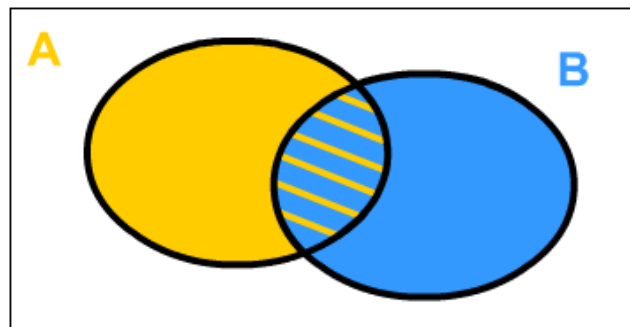


Figura 18: Exemplo da relação de intersecção entre dois elementos.

2.2 Data warehouse

Um *data warehouse* (DW) poder ser definido como um repositório de dados possivelmente coletados de diversas fontes cuja função é prover apoio à tomada de decisão (KIMBALL, 1998). Segundo Inmon (1997), um DW é um repositório de dados organizado por assuntos, integrado, não volátil e variável no tempo. Ser organizado por assunto significa possuir os assuntos separados em estruturas similares a um DW, mas menores, chamadas *Data marts* (DM). É integrado, pois dados de diversas fontes são extraídos, integrados e carregados no DW. A natureza não-volátil de um DW se explica pelo fato de após os dados serem extraídos, integrados e carregados no DW, eles estarão disponíveis para o usuário apenas para consulta. A característica temporal de um DW (variável no tempo) deve-se ao fato de um DW armazenar informações

por grandes períodos de tempo, e é um fator fundamental para suas aplicações.

2.2.1 Modelo Dimensional

Um DW é organizado segundo o modelo dimensional, onde medidas (*e.g.*, área colhida, área plantada) são organizadas segundo dimensões (produto agrícola, espaço e tempo). As dimensões de análise são organizadas em hierarquias. Por exemplo, a dimensão tempo pode ser vista como uma hierarquia incluindo níveis como ano, mês e dia.

Um esquema dimensional é composto de uma ou mais tabelas de fatos relacionadas a tabelas de dimensões. As tabelas de fatos têm um caráter quantitativo, contendo apenas dados numéricos, isto é, medidas além das chaves das tabelas de dimensão. As tabelas de dimensão servem para qualificar os fatos. Elas contêm vários atributos que descrevem em detalhes valores qualitativos a que as mediadas se referem (*e.g.*, área plantada de certo produto, em certa região em um determinado período de tempo).

Os esquemas dimensionais podem ser de dois tipos principais, como proposto por Kimball (1998): esquema estrela ou esquema floco-de-neve. Um esquema estrela é caracterizado por uma única tabela de fatos cercada por dimensões, e as dimensões se relacionam apenas com a tabela de fatos, ou seja, não existem dimensões relacionando-se entre si (desnormalizadas). A figura 19 ilustra um esquema estrela:

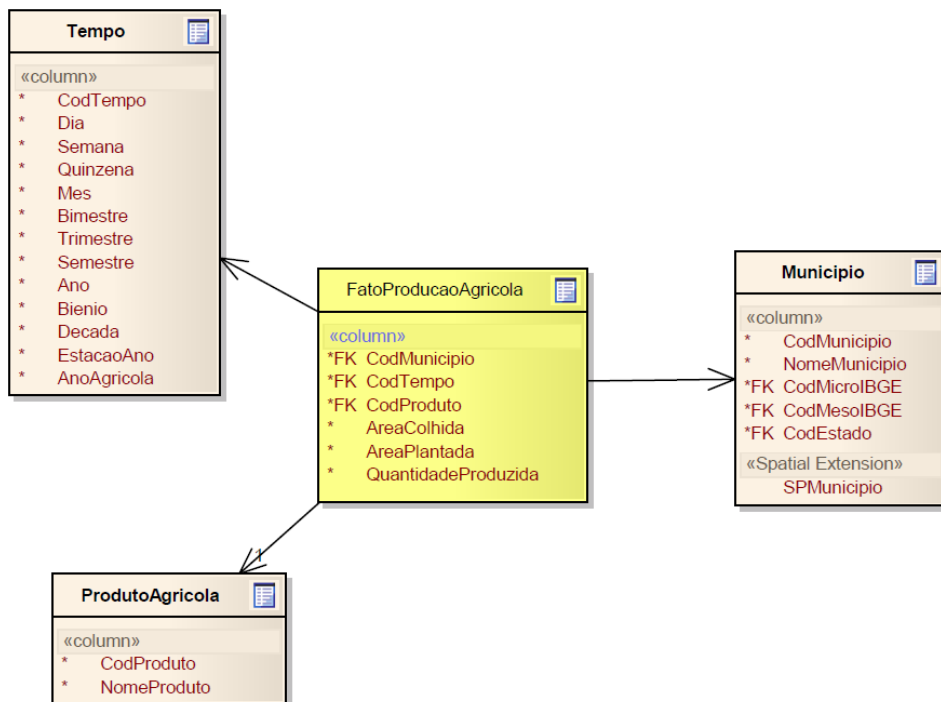


Figura 19: exemplo de esquema estrela. Uma única tabela fato e várias dimensões.

Um esquema floco-de-neve caracteriza-se por possuir diversas tabelas fato com dimensões compartilhadas, e as dimensões são organizadas hierarquicamente por meio da normalização, geralmente até a terceira forma normal (ELMASRI; NAVATHE, 2006), de maneira a permitir um acesso à informação com um nível mais fino de detalhe e uma redução da redundância. A normalização de um *data warehouse* pode impactar em seu desempenho, embora possa ser justificável em diversos casos, mas esta discussão está fora do escopo deste trabalho. Na Figura 20 é apresentado um exemplo de esquema floco-de-neve:

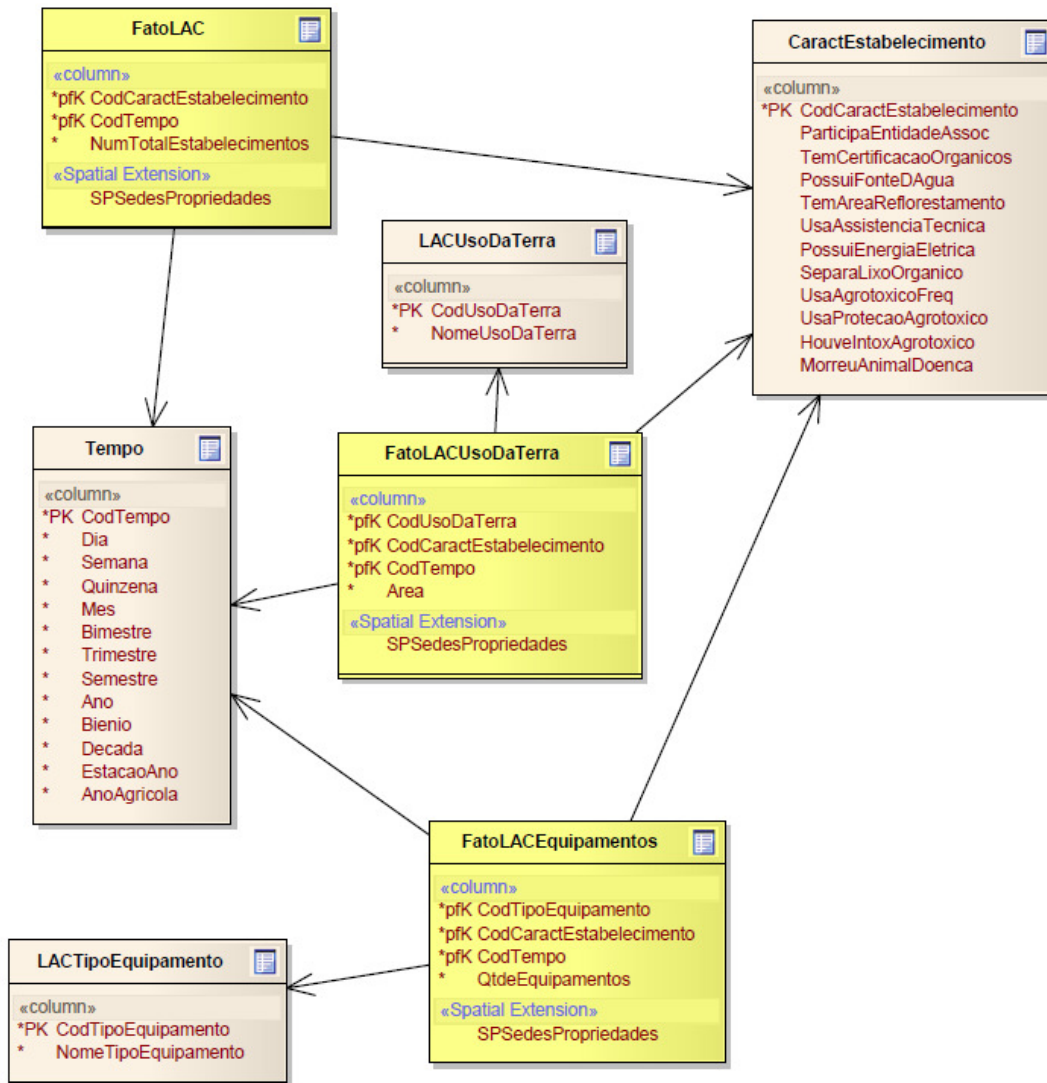


Figura 20: exemplo de esquema floco-de-neve. Mais de uma tabela fato, com dimensões compartilhadas e certo nível de normalização.

Devido a sua natureza, diferentemente de um esquema relacional, nos modelos dimensionais é possível fazer agregações de dados, o que abre uma enorme quantidade de possibilidades de análises a serem feitas. A análise de dados baseada em um modelo dimensional é feita através da construção de um cubo com as dimensões do modelo, assim podendo responder consultas envolvendo diversas variáveis de forma muito mais eficiente do que um modelo relacional. A figura 21 mostra um exemplo de cubo.

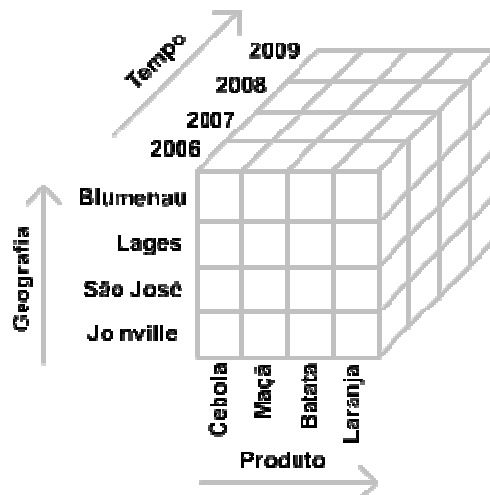


Figura 21: exemplo de cubo OLAP representando a produção de algumas culturas em municípios ao longo dos últimos quatro anos, baseado do modelo dimensional da Figura 19.

Através deste cubo, são analisadas três variáveis ao mesmo tempo: geografia, tempo e produto, e, as medidas da tabela fato vão povoar este cubo. Dessa forma, é possível responder consultas multidimensionais como: “quais cidades apresentaram uma taxa de crescimento na produção de maçã superior a 10% nos últimos dois anos?”. A partir de um modelo dimensional podem ser montados diversos cubos, de acordo com as perguntas a serem respondidas.

As principais diferenças entre o modelo relacional e o modelo dimensional estão elencadas no quadro comparativo abaixo (MELLO, 2002):

Quadro 1: comparativo entre os modelos relacional e dimensional.

	Modelo Relacional	Modelo Dimensional
Função	Automatizar operações cotidianas.	Apoio à decisão.
Usuário Foco	Cliente, atendente.	Executivo, analista, técnico.
Granularidade	Única e atômica.	Múltipla e agregada.
Temporalidade dos Dados	Valor corrente. Atualizada constantemente.	Histórico de valores. Completado periodicamente.
Direção	Leitura e escrita.	Somente leitura (não-

		volátil).
--	--	-----------

2.2.2 Abordagens para a construção de *Data Warehouses*

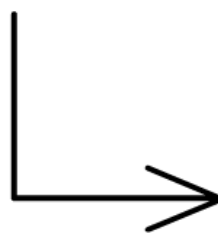
A construção de um *data warehouse* pode ser abordada de duas formas: *top-down* e *bottom-up*, propostas por Inmon (1997) e Kimball (1998), respectivamente. A abordagem *top-down* prevê que se comece o projeto primeiramente do DW como um todo, para depois lidar com os seus DMs. Esta abordagem é pouco utilizada, visto que demanda um esforço inicial muito grande e um conseqüente elevado tempo para que o DW comece de fato a trabalhar. Já a abordagem *bottom-up* propõe o contrário, começa-se o projeto pelas partes menores, os DMs, e, a partir deles o DW irá sendo construído aos poucos. Esta técnica leva uma grande vantagem, pois o sistema pode começar a ser utilizado antes de estar totalmente pronto, impactando diretamente no tempo de retorno do investimento feito para a construção do DW.

2.2.3 *Online Analytical Processing*

O *OnLine Analytical Processing* (OLAP) permite analisar informação de um cubo dimensional em vários níveis de granularidade (SOARES, 2008). Para selecionar e transitar por entre as dimensões e os diversos níveis de agregação faz-se o uso dos operadores básicos OLAP em cubos dimensionais:

Drill-down: esta técnica provê uma visão das informações em um nível de granularidade mais fino (ELMASRI; NAVATHE, 2006), trazendo mais detalhes. Como por exemplo, ao analisar medidas relativas à determinada meso-região e refinar a análise por município daquela mesma meso-região, ou seja, trazendo mais detalhes à análise, assim como ilustra a figura 22:

	Medidas		
	Área Plantada	Área Colhida	Quantidade Produzida
[+] Meso-região: SUL CATARINENSE	7243	7243	12596

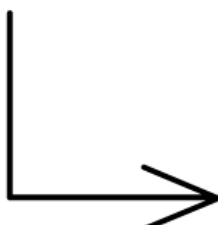


	Medidas		
	Área Plantada	Área Colhida	Quantidade Produzida
[-] Meso-região: SUL CATARINENSE	7243	7243	12596
Praia Grande	167	167	321
Jacinto Machado	400	400	500
Timbé do Sul	10	10	16
São João do Sul	300	300	900
Turvo	100	100	110
Santa Rosa do Sul	150	150	300
Morro Grande	5	5	9
Passo de Torres	70	70	140
Sombrio	200	200	360
Ermo	160	160	240
...

Figura 22: Operação de *drill-down*.

Roll-up: esta operação é o oposto da operação de *drill-down*, ou seja, ela provê uma visão num nível de granularidade mais espesso, agregando informações, conforme a figura 23 demonstra:

	Medidas		
	Área Plantada	Área Colhida	Quantidade Produzida
[-] Meso-região: SUL CATARINENSE	7243	7243	12596
Praia Grande	167	167	321
Jacinto Machado	400	400	500
Timbé do Sul	10	10	16
São João do Sul	300	300	900
Turvo	100	100	110
Santa Rosa do Sul	150	150	300
Morro Grande	5	5	9
Passo de Torres	70	70	140
Sombrio	200	200	360
Ermo	160	160	240
...




	Medidas		
	Área Plantada	Área Colhida	Quantidade Produzida
[+] Meso-região: SUL CATARINENSE	7243	7243	12596

Figura 23: Operação de roll-up.

Pivoting: esta técnica permite analisar os dados de outra forma, trocando de lugar ou adicionando dimensões da análise, com o intuito de mostrar uma orientação diferente dos eixos do cubo. Como ilustra a figura 24, onde a princípio estavam sendo analisadas apenas as medidas “Área Plantada” e “Área Colhida” e, posteriormente, foi adicionada uma terceira coluna representando a medida “Quantidade Produzida”.

	Medidas	
	Área Plantada	Área Colhida
[-] Meso-região: SUL CATARINENSE	7243	7243
Praia Grande	167	167
Jacinto Machado	400	400
Timbé do Sul	10	10
São João do Sul	300	300
Turvo	100	100
Santa Rosa do Sul	150	150
Morro Grande	5	5
Passo de Torres	70	70
Sombrio	200	200
Ermo	160	160
...



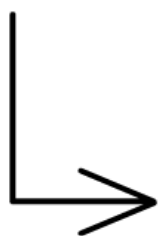
	Medidas		
	Área Plantada	Área Colhida	Quantidade Produzida
[-] Meso-região: SUL CATARINENSE	7243	7243	12596
Praia Grande	167	167	321
Jacinto Machado	400	400	500
Timbé do Sul	10	10	16
São João do Sul	300	300	900
Turvo	100	100	110
Santa Rosa do Sul	150	150	300
Morro Grande	5	5	9
Passo de Torres	70	70	140
Sombrio	200	200	360
Ermo	160	160	240
...

Figura 24: Operação pivoting.

Slice-and-dice: esta operação fixa ou até mesmo remove uma informação de uma dimensão, com o objetivo de simplificar a visualização e análise das informações. A figura 25 ilustra uma situação onde estavam sendo analisadas medidas de todas as cidades de uma meso-região e,

posteriormente, fixaram-se dados de três municípios, eliminando os outros da análise:

	Medidas		
	Área Plantada	Área Colhida	Quantidade Produzida
[-] Meso-região: SUL CATARINENSE	7243	7243	12596
Praia Grande	167	167	321
Jacinto Machado	400	400	500
Timbé do Sul	10	10	16
São João do Sul	300	300	900
Turvo	100	100	110
Santa Rosa do Sul	150	150	300
Morro Grande	5	5	9
Passo de Torres	70	70	140
Sombrio	200	200	360
Ermo	160	160	240
...



	Medidas		
	Área Plantada	Área Colhida	Quantidade Produzida
[-] Meso-região: SUL CATARINENSE	7243	7243	12596
Praia Grande	167	167	321
Jacinto Machado	400	400	500
Timbé do Sul	10	10	16

Figura 25: Operação *slice and dice*.

2.3 *Data warehouses* espaciais

Os benefícios do apoio à decisão provido por um DW e OLAP (*On-Line Analytical Processing*) juntamente á análise espacial provida por um SIG são um excelente motivo para a integração desses dois tipos de ferramentas, de modo a viabilizar consultas analíticas multidimensionais com características espaciais. O resultado dessa integração é um *Data warehouse* Espacial (DWE) (SIQUEIRA, 2008).

Data warehouses Espaciais estendem *Data warehouses* tradicionais, oferecendo suporte ao tratamento de dados espaciais (FIDALGO, 2005). Um DWE integra dados e operadores do modelo dimensional tradicional com representações geométricas, operadores e funções de agregação de dados geográficos (RAO *et al.*, 2003; MALINOWSKY e ZIMÁNYI, 2007). Com a representação de dados geográficos em DWs, é possível realizar análises e operações envolvendo dados espaciais e exibir os resultados em mapas (por

exemplo, colorir dinamicamente regiões de um mapa de acordo com a produção agrícola de determinada época) (DEGGAU, 2009).

Dados espaciais podem aparecer nas dimensões (por exemplo, dimensão região com os níveis mesorregião, microrregião e município) de um esquema de um DWE ou como medidas na tabela de fato (por exemplo, pontos onde houve casos de intoxicação por agrotóxico). Essa modelagem depende de quais questões as consultas do DWE terão que responder.

Uma classificação proposta por Han (1998) determina três tipos de dimensões (não-espacial, espacial para não-espacial e espacial para espacial) e dois tipos de fatos (medida numérica e medida espacial) para *data warehouses* espaciais, como explicadas a seguir.

Dimensão Não-Espacial é aquela que, como o próprio nome já diz, contém apenas dados não-espaciais. Por exemplo, uma dimensão chamada “CaractEstabelecimento” que define característica de estabelecimentos agrícolas, que pode ter apenas generalizações independentes de dados espaciais, como generalizações relativas a seu porte, classificadas como “pequeno”, “médio” ou “grande”, conforme ilustra a figura 26:



Figura 26: Exemplo de dimensão não espacial.

Dimensão Espacial para Não-Espacial é aquela cujo nível mais baixo de granularidade é espacial, mas suas generalizações (níveis mais altos) tornam-se não-espaciais. Por exemplo, uma dimensão que define espacialmente as micro-regiões do estado de Santa Catarina pode ter generalizações representadas por dados não-espaciais, apenas como “produtora de maçã” ou “produtora de cebola” perdendo assim seu caráter espacial, conforme pode ser observado na figura 27:

MicroIBGE	
«column»	
*	CodMicroIBGE
*	NomeMicroIBGE
*	Cultura
«Spatial Extension»	
	SPMicroIBGE

Figura 27: Exemplo de dimensão espacial para não-espacial.

Dimensão Espacial para Espacial possui dados espaciais em seu nível mais baixo de granularidade e em todos os outros níveis mais altos que podem ser generalizados. Um exemplo é a dimensão que define espacialmente as micro-regiões do estado de Santa Catarina cujas generalizações classificam os territórios em “até 1000 habitantes”, “de 1000 a 5000 habitantes” e assim por diante, conforme figura 28:

MicroIBGE	
«column»	
*	CodMicroIBGE
*	NomeMicroIBGE
*	População
«Spatial Extension»	
*	SPMicroIBGE

Figura 28: Exemplo de dimensão espacial para espacial.

Medida Numérica representa única e exclusivamente quantidades. É puramente numérica, retornam apenas um número quando a ela aplicadas funções de agregação (máximo, mínimo, média etc.). Por exemplo, a quantidade de cebola plantada ou a quantidade de milho colhida.

Medida Espacial refere-se a um conjunto de objetos espaciais. Por exemplo, sedes de propriedades agrícolas de Santa Catarina.

3. *Data warehouse* Espacial para a EPAGRI

A EPAGRI foi criada em 1991 com o objetivo de promover a preservação, recuperação e conservação de recursos naturais, buscando a competitividade da agricultura catarinense frente aos mercados globalizados e promover a melhoria da qualidade de vida do meio rural e pesqueiro. A EPAGRI possui uma vasta quantidade de informação, referente a pesquisas e projetos necessários ao atendimento de seus clientes. Esta informação precisa ser adequadamente analisada e transmitida ao público interno e externo, para que possa ser devidamente utilizada nos procedimentos de atendimento aos clientes.

Visando o incremento da elaboração de sistemas com o foco estratégico e a inclusão da empresa no portal da Inovação do Ministério da Ciência e Tecnologia, para compartilhar o conhecimento adquirido, percebeu-se a possibilidade de avançar nos procedimentos de integração das informações através da construção de um *Data warehouse* (DW), com uso de informações OLAP integrado a um Sistema de Informações Geográficas (SIG). Muitos dos sistemas existentes têm um bom potencial para uso de informações georeferenciadas e esta utilização permitiria aos usuários destes sistemas um novo recurso na análise de dados auxiliando no processo de tomada de decisão.

Um *Data warehouse* Espacial foi criado (DEGGAU e FILETO, 2009), agregando dados de diversas fontes da EPAGRI, com o objetivo de servir como fonte de estudo e permitir a realização de diversos experimentos e análises com seu conteúdo.

O *Data warehouse* construído para a EPAGRI se baseou inicialmente nos dados dos sistemas SIAGRO – Sistema de Informações Agropecuárias e nos dados do LAC – Levantamento Agropecuário Catarinense. Estes sistemas possuem grande potencial para uso de informações geográficas, pois abrangem dados de todo o estado de Santa Catarina e no caso específico do SIAGRO, dados de outros estados do Brasil e de países. Estes sistemas possuem séries históricas de informação e possuem um conjunto de dados com interesse desde o nível gerencial da empresa até usuários finais.

Esta segunda abordagem mostrada é a escolhida para desenvolver o *data warehouse* espacial para a EPAGRI. A figura 29 apresenta exemplo de *Data Mart*:

Data Mart Produção Agrícola

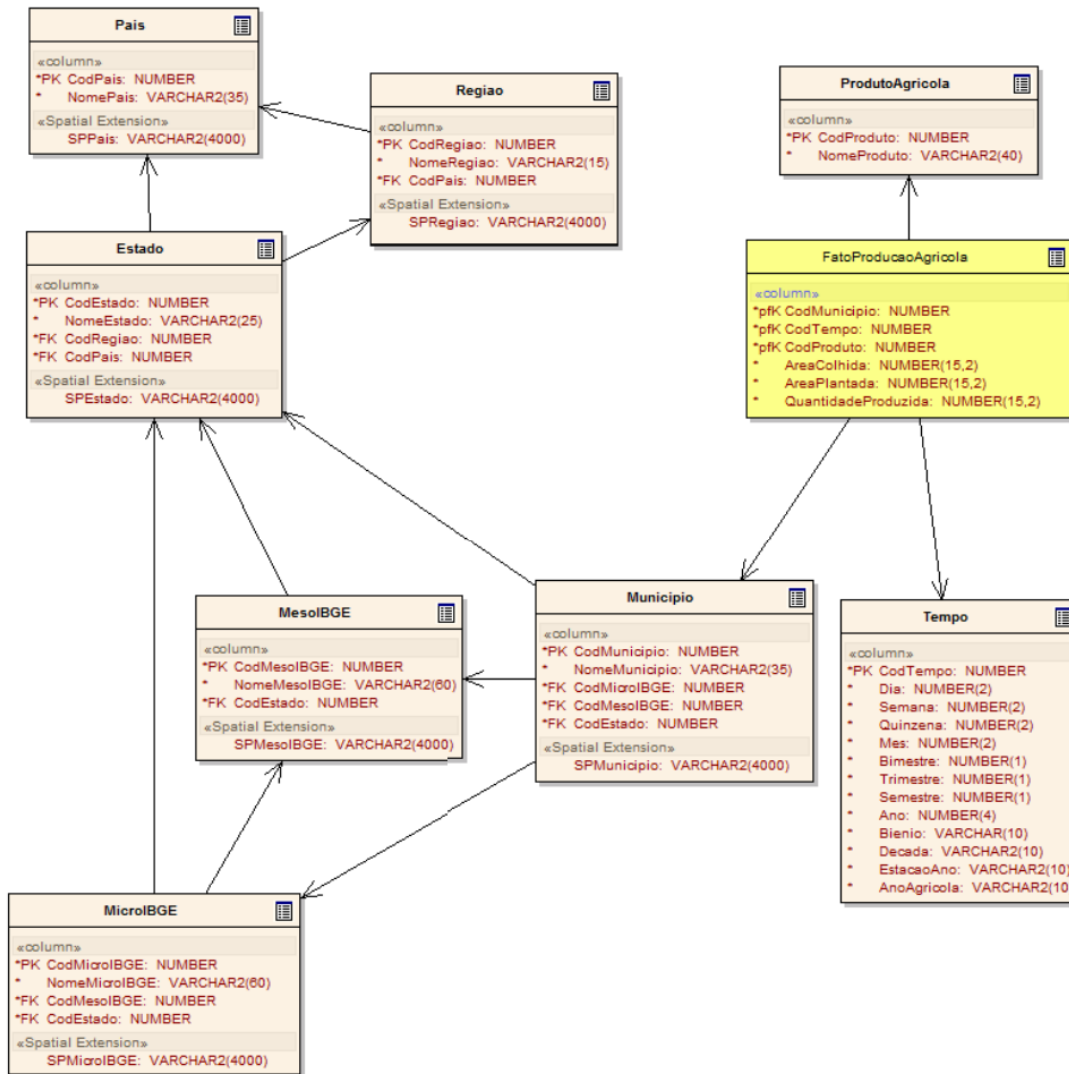


Figura 29: Modelo dimensional do *data mart* “Produção Agrícola”

O *data mart* “Produção Agrícola” é um exemplo de *data mart* que possui dados espaciais apenas em dimensões. As dimensões “Região”, “MesoIBGE”, “MicroIBGE”, “Município” e “País” possuem um atributo denotado “*SpatialExtension*” que é a representação espacial da entidade. Todas estas dimensões podem ser classificadas como dimensão “espacial para não-

espacial”, pois o seu menor nível de granularidade é espacial, mas a partir dele podem ser feitas generalizações não-espaciais. Por exemplo: “O nome das micro-regiões que compõem a Meso Região do Vale do Itajaí” retornará apenas nomes, e não a extensão geográfica das micro-regiões, embora isto também seja possível.

4. Análise das ferramentas

O objetivo desta seção é apresentar as ferramentas escolhidas para análise, descrevendo suas características ao lidar com dados espaciais em *data warehouses*. Depois de apresentadas, as ferramentas são comparadas segundo os seguintes critérios: integração do modelo dimensional com o modelo espacial, suporte a dados e operadores espaciais, funções de agregação espacial, visualização de resultados de consultas OLAP espacial em mapas e acoplamento entre os módulos.

4.1 *SQL Server 2008*

O *Microsoft SQL Server* é uma *suíte* de aplicativos que consolida o gerenciamento de servidores e a criação de objetos de apoio ao negócio por meio de dois ambientes de trabalho: o *SQL Server Management Studio* e o *Business Intelligence Development Studio*. Estes ambientes englobam aplicativos mais específicos como *SQL Server* (mecanismo de banco de dados), *SQL Server Compact 3.5 SP1*, *Analysis Services*, *Integration Services* e *Reporting Services*. A figura 30 ilustra como se dá a arquitetura desta *suíte* (MSDN, [S.I.]):

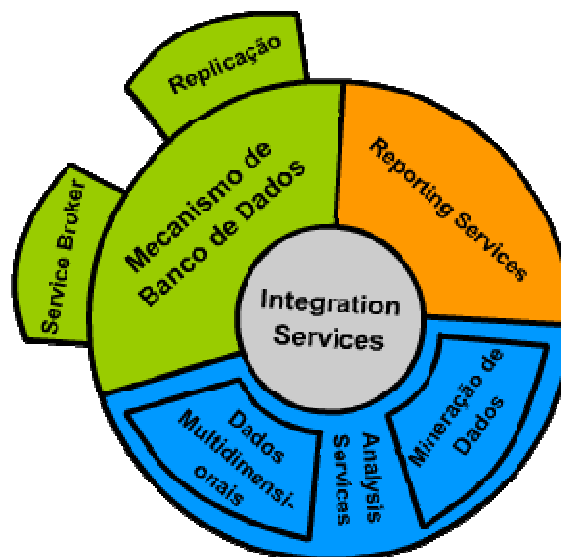


Figura 30: Esquema da organização da suíte de ferramentas da *Microsoft*.

Mecanismo de Banco de Dados: é o serviço principal para armazenamento, processamento e segurança de dados. Sua principal função é criar bancos de dados relacionais de processamento de transações online ou dados provenientes de consultas OLAP. Isso inclui a criação de tabelas para armazenamento de dados e objetos de banco de dados, como índices, exibições e procedimentos armazenados para exibição, gerenciamento e segurança de dados.

Reporting Services: é um conjunto de componentes de processamento, ferramentas e interfaces de programação que oferecem suporte ao desenvolvimento e uso de relatórios em um ambiente gerenciado. O conjunto de ferramentas inclui ferramentas de desenvolvimento, de configuração e de administração, além de ferramentas de visualização de relatórios.

Integration Services: é uma plataforma para criar integração de dados e soluções de transformações de dados. A função do *Integration Services* é solucionar problemas corporativos complexos copiando ou baixando arquivos, enviando mensagens de *e-mail* em resposta a eventos, atualizando *data warehouses*, fazendo a limpeza e mineração de dados e gerenciando objetos e dados do *SQL Server*. Os pacotes podem funcionar sozinhos ou junto com outros pacotes para tratar das necessidades empresariais complexas. O *Integration Services* pode extrair e transformar dados a partir de uma ampla variedade de fontes como arquivos de dados *XML*, arquivos simples e fontes de dados relacionais e transferir dados para um ou mais destinos.

Analysis Services: é uma ferramenta que faz parte da *suíte Microsoft SQL Server 2008*. Este aplicativo tem como foco principal a manipulação de *Data warehouses*, construção de cubos OLAP, além de possuir mecanismos de mineração de dados. Ela permite projetar, criar e gerenciar estruturas multidimensionais que contenham detalhes e dados de agregação de várias fontes de dados, como bancos de dados relacionais, em um único modelo lógico e unificado com suporte para cálculos internos. Esta ferramenta opera com *data warehouses*, *data marts*, bancos de dados de produção e armazenamento de dados operacional, com suporte à análise de dados históricos e em tempo real. O *Analysis Services* também possui um suporte à mineração de dados e, para isso, oferece um conjunto de algoritmos de

mineração de dados, o *designer* de Mineração de Dados, que permite criar, gerenciar e explorar modelos de mineração de dados, e depois criar previsões usando esses modelos; e ainda a linguagem DMX que pode ser usada para gerenciar modelos de mineração de dados e para criar consultas de previsão complexas.

O *Analysis Services* é uma ferramenta integrada ao ambiente *Microsoft Business Intelligence Studio*, onde as fontes e as visões de dados são configuradas, as tabelas de fatos e de dimensões são criadas, extraíndo e transformando dados da fonte de dados pré-configurada; e também é onde os cubos OLAP são criados e analisados.

A base de dados que servirá de base para o DW a ser modelado no *Analysis Services* não necessariamente precisa estar no *SQL Server 2008*, pode ser em qualquer outro sistema gerenciador de banco de dados que possua alguma forma de acesso externo aos seus dados (como uma conexão *ODBC*, por exemplo). No caso deste trabalho, a fonte de dados foi uma base *Oracle 11g* da EPAGRI.

4.1.1 Suporte a Dados Espaciais

Na plataforma *SQL Server 2008*, os dados espaciais são divididos em duas categorias: *geometry* e *geography*. O tipo de dados *geometry*, que segue as especificações da OGC citadas na seção 2.1.1.2, representa dados planares, também conhecidos por euclidianos. Já o tipo de dados *geography* oferece uma representação de dados do globo terrestre (elipsoidais) de latitude e longitude. No escopo deste trabalho, apenas o tipo de dados *geometry* apresenta-se relevante.

A plataforma *Microsoft* oferece suporte a onze objetos espaciais, tanto *geography* quanto *geometry*, embora apenas sete deles sejam instanciáveis, podendo ser trabalhadas em um banco de dados. Estas instâncias herdam algumas características do tipo de dados pai que as distinguem entre si como *Points*, *LineStrings* ou ainda como várias instâncias, sejam elas *geography* ou *geometry*, em uma *GeometryCollection*. A figura 31 ilustra a hierarquia dos objetos espaciais *geometry*, na qual tanto dados *geometry* quanto *geography* se baseiam:

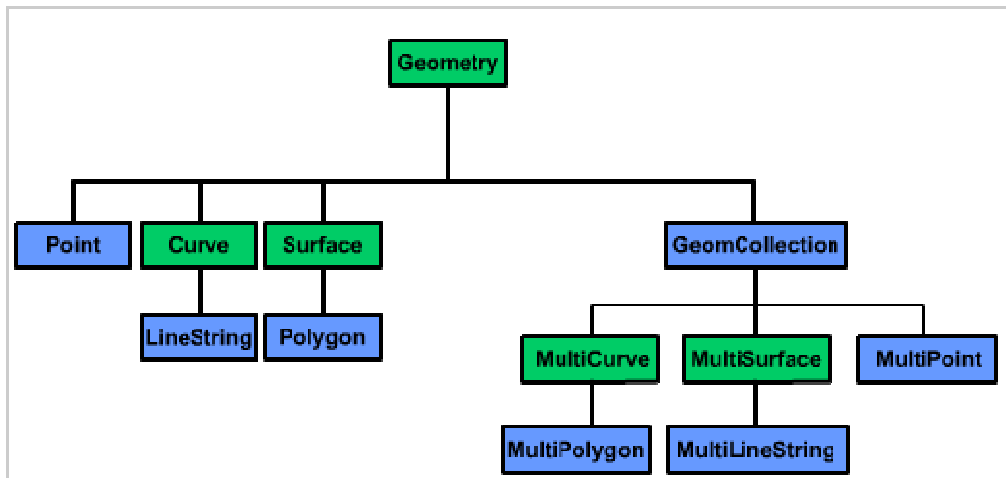


Figura 31: Arquitetura dos tipos de dados geográficos do *Microsoft SQL Server* 2008. Os tipos de dados que podem ser instanciados estão indicados em azul.

4.1.2 Suporte a Operadores Espaciais

Os operadores espaciais suportados pela plataforma *Microsoft* são todos binários, ou seja, expressam a relação entre apenas dois elementos, e retornam sempre um valor *booleano*, com exceção do operador *STDistance*, que retorna a distância entre dois elementos espaciais (MSDN, 2009). Os operadores espaciais suportados estão listados abaixo, juntamente com os métodos que os implementam:

- ***STEquals***: igualdade;
- ***STDisjoint***: disjunção;
- ***STIntersects***: intersecção;
- ***STTouches***: toca;
- ***STOverlaps***: sobreposição;
- ***STCrosses***: cruza;
- ***STWithin***: está contido;
- ***STContains***: contém;
- ***STOverlaps***: sobreposição;
- ***STRelate***: relação especial;
- ***STDistance***: distância mais curta.

4.1.3 Implementação de Um Data Warehouse Espacial

A manipulação de data warehouses bem como a etapa de ETL e a criação de cubos na plataforma da Microsoft é feita pelo Analysis Services, conforme apresentado na seção 4.1. A interação com os projetos do Analysis Services é feita através do *Microsoft SQL Server Business Intelligence Development Studio*, que é um ambiente de desenvolvimento baseado no *Microsoft Visual Studio* (MSDN, 2008).

A interação com o ambiente da Microsoft é bastante intuitiva. Para criar um novo projeto do Analysis Services, deve-se ir a *File>New>Project...* e selecionar Analysis Services Project, conforme ilustra a figura 32.

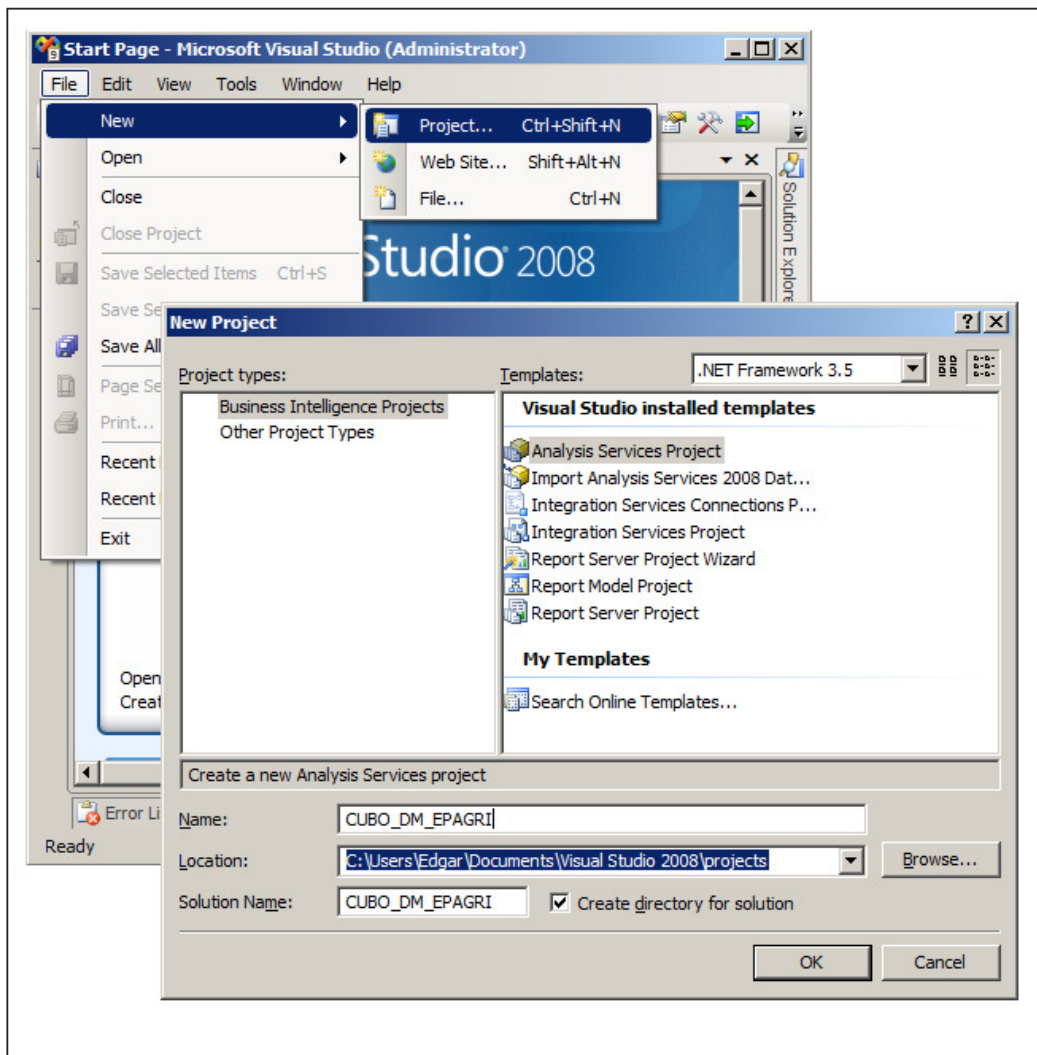


Figura 32: Criação de um novo projeto do *Analysis Services* no *Microsoft SQL Server Business Intelligence Development Studio*.

Depois de criado o projeto, é preciso configurar uma conexão com o banco de dados. Neste exemplo, a base de dados é um *Oracle 11g* que está instalado na mesma máquina que o *Microsoft SQL Server Business Intelligence Development Studio*. Para efetuar a conexão, basta ir a *Data Connections*, localizado no painel *Server Explorer*, clicar com o botão direito no mouse e selecionar a opção *Add Connection*, e depois preencher os campos de acordo com a base de dados a ser utilizada no projeto, conforme mostrado pela figura 33.

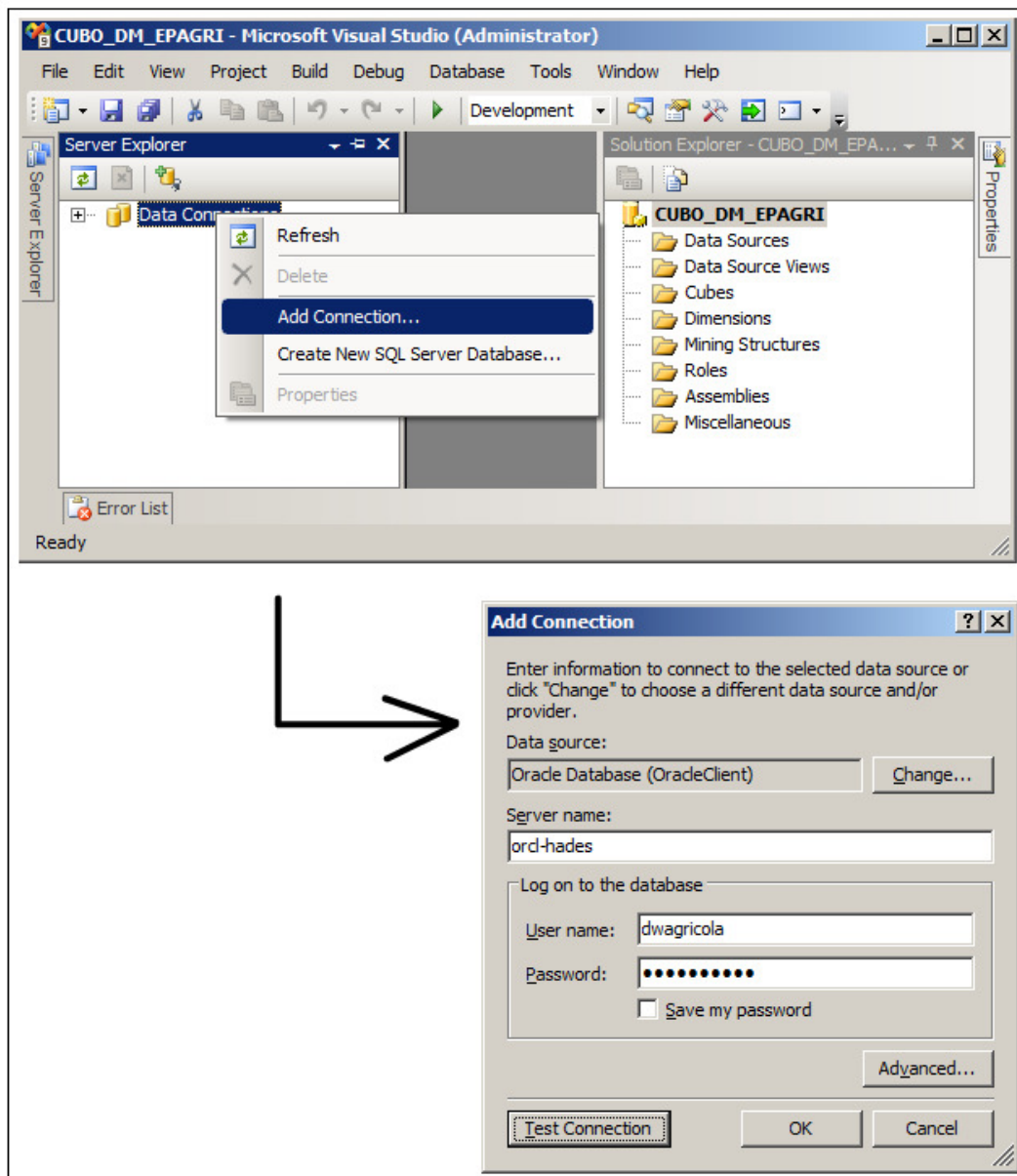


Figura 33: Configuração de uma nova conexão com banco de dados no *Microsoft SQL Server Business Intelligence Development Studio*.

Tendo configurado uma conexão com o banco de dados, é necessário definir uma fonte de dados. Para isso, basta ir ao painel *Solution Explorer*, clicar com o botão direito do mouse e selecionar a opção *New Data Source*. Na tela que abrirá, basta preenchê-la de acordo com os dados da base de dados a ser utilizada, como ilustrado pela figura 34.

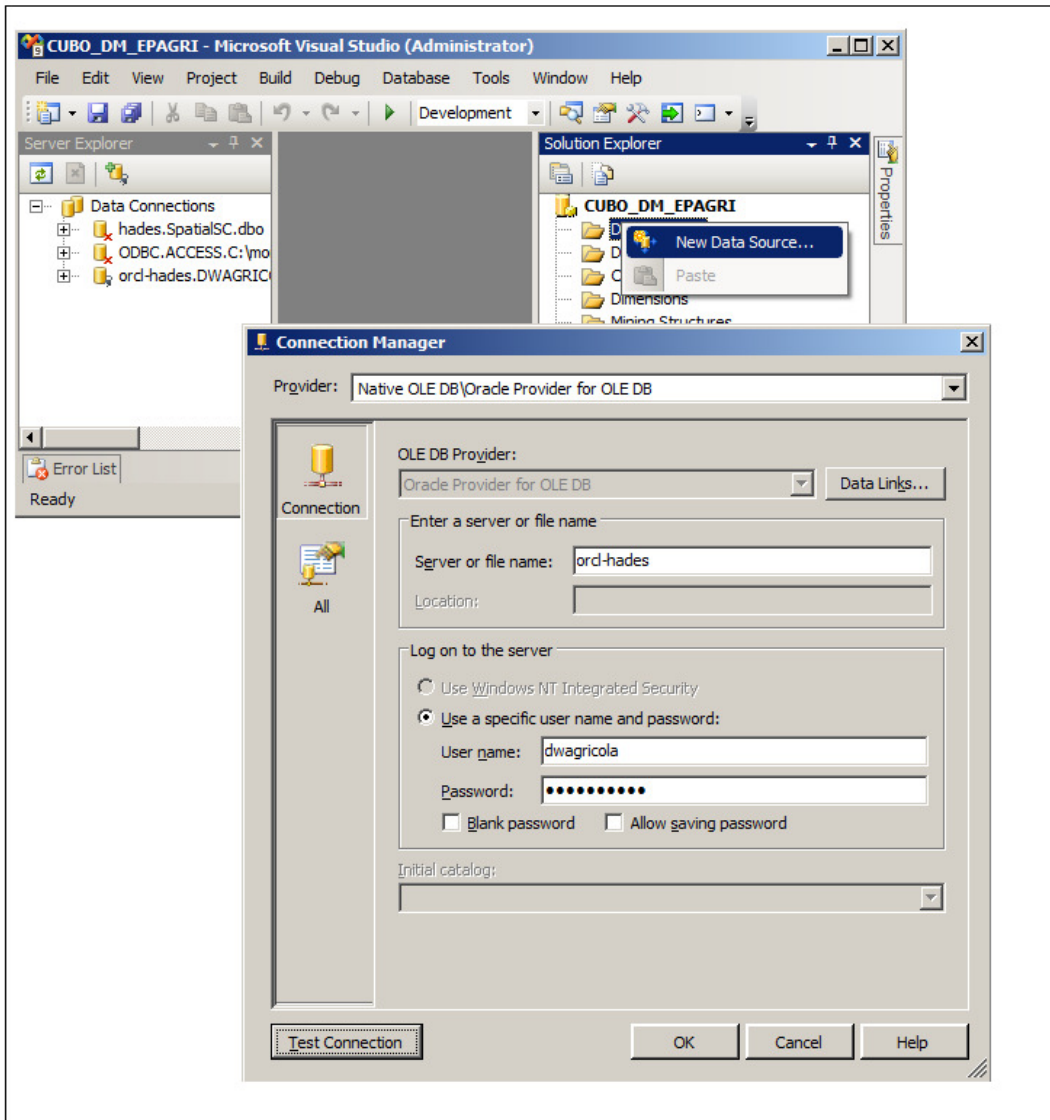


Figura 34: Configuração de uma nova fonte de dados no *Microsoft SQL Server Business Intelligence Development Studio*.

Com a fonte de dados definida, agora é necessário criar uma *view* desta fonte de dados. O procedimento também é bastante simples, bastando apenas

ir novamente ao painel *Solution Explorer*, clicar com o botão direito do *mouse* em *Data Source Views*. Logo em seguida aparecerá uma tela com todas as tabelas de sua base de dados, onde você deve escolher as tabelas que serão utilizadas, assim como mostra a figura 35.

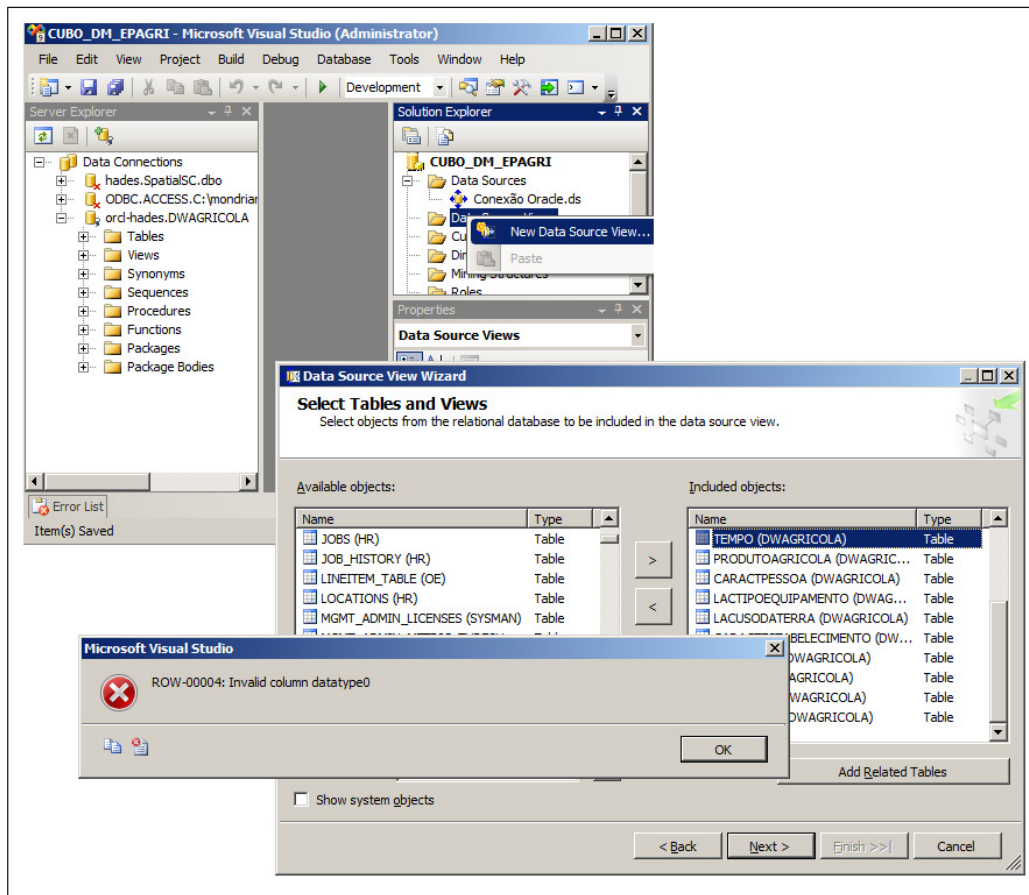


Figura 35: Definição de uma *view* sobre a fonte de dados criada e mensagem de erro informando tipos incompatíveis de dados espaciais utilizados pelo *Oracle* e pela *Microsoft*.

Na figura 35 também é possível observar um erro que aparece ao tentar utilizar tabelas com dados espaciais oriundos do *Oracle*. A mensagem de erro apresentada limita-se a informar que os tipos de dados não são compatíveis. Muito foi pesquisado na documentação, manuais online disponibilizados tanto pela *Oracle* quanto pela *Microsoft*, bem como em fóruns relacionados a este assunto, mas muito pouco se encontra sobre como resolver este problema de incompatibilidade.

4.2 PentahoBI Platform

O *PentahoBI Platform* é uma plataforma *open source* de *business intelligence* (BI) que provê a arquitetura e a infra-estrutura necessária para construir soluções de BI. Esta plataforma é composta por ferramentas de integração de dados e ETL (*extraction, transform, load*), análise (OLAP), geração de relatórios e mineração de dados. Abaixo segue esquema da arquitetura da plataforma. A versão analisada neste trabalho é a *community*, distribuída gratuitamente, sob a licença *GNU LGPL (Lesser General Public License)*.

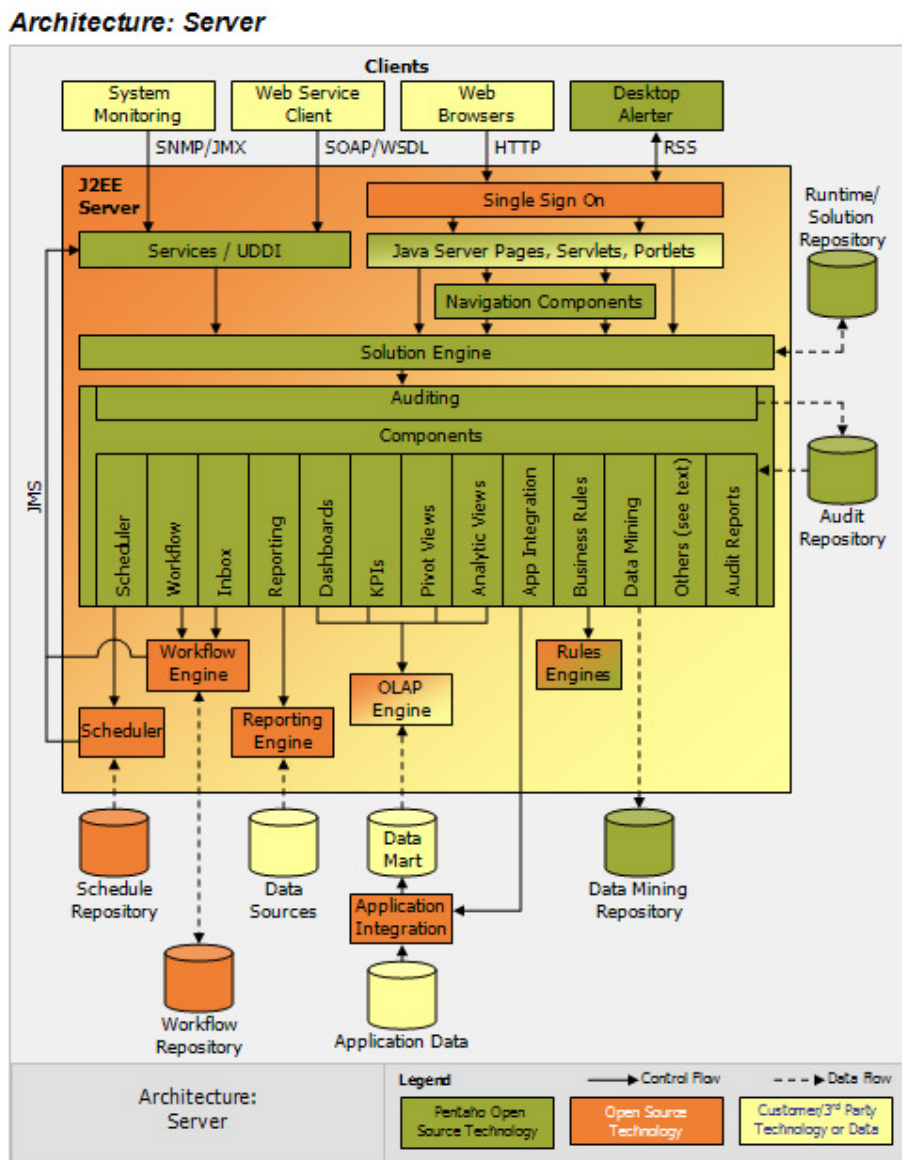


Figura 36: Arquitetura da Plataforma *Pentaho* (PENTAHO, 2006).

Todos os aplicativos da plataforma *Pentaho* são escritos em Java e são apresentadas abaixo:

Mondrian: é o servidor OLAP da plataforma, onde os cubos são construídos e posteriormente analisados. Assim como toda a plataforma, é escrito em *Java* e implementa as linguagens *MDX*, *XML for Analysis* e especificações de *Java OLAP (JOLAP)*;

Kettle: é o aplicativo responsável pelas etapas de integração de dados e de ETL (*extraction, transform and load*). Nele é possível acessar dados de diversas fontes diferentes, fazer operações ou transformações e apresentar os resultados em diversos formatos, como *XML*, *XLS* ou *CSV* por exemplo.

Weka: é uma ferramenta de mineração de dados. O *Weka* possui uma série de algoritmos implementados para a descoberta de conhecimento, que podem ser invocados de dentro de um código em *Java* ou aplicados diretamente sobre um conjunto de dados, que podem ser provenientes de transformações do *Kettle*, por exemplo.

PentahoReporting: ferramenta que acessa o conjunto de dados e auxilia a criação de relatórios personalizados, ajudando a visualizar os resultados obtidos.

4.2.1 Suporte a Dados Espaciais

A suíte de aplicativos *Pentaho* não possui nenhum módulo que ofereça suporte nativo a extensões espaciais. Uma das soluções para o tratamento de dados espaciais pela suíte *PentahoBI*, bem como seus operadores, pode ser feito utilizando o aplicativo *GeoKettle*.

4.2.2 GeoKettle

GeoKettle é uma versão do aplicativo *Kettle*, apresentado na seção anterior, com modificações para suportar extensões espaciais, desenvolvido pela *Université Laval*, e, assim como o *Kettle*, é distribuído também sob a licença *GNU LGPL*. Em adição aos tipos de dados já suportados pelo *Kettle*, o *GeoKettle* oferece suporte ao tipo de dados *geometry*, oferecendo suporte a elementos espaciais como ponto, polígono, linha, entre outros (BADARD, 2008), implementados seguindo a especificação da *JTS Topology Suite*.

A *JTS Topology Suite* é uma *API Java* que implementa as definições dos dados espaciais, bem como as operações entre eles (VIVID Solutions, 2003). Esta *API* segue os padrões descritos pela OGC, com algumas alterações pontuais que serão explicitadas quando necessário.

Os tipos de dados espaciais suportados, definidos pela OGC, e implementados por essa especificação são:

- *GeometryCollection*;
- *Point*;
- *MultiPoint*;
- *LineString*;
- *LinearRing*;
- *MultiLineString*;
- *Polygon*;
- *MultiPolygon*.

4.2.3 Suporte a Operadores Espaciais

Os operadores espaciais suportados pelo *GeoKettle* estão elencados abaixo, representados pelo nome do método que os implementa:

- ***GIS_INTERSECTS***: intersecção;
- ***GIS_EQUALS***: igualdade;
- ***GIS_CONTAINS***: contém;
- ***GIS_CROSSES***: cruza;
- ***GIS_DISJOINT***: disjunção;
- ***GIS_WITHIN***: está contido;
- ***GIS_OVERLAPS***: sobreposição;
- ***GIS_TOUCHES***: toca;
- ***GIS_ISVALID***: é válido.

Todos os operadores são binários, ou seja, comparam um elemento a outro, com exceção do operador “é válido”, que é um operador unário que retorna verdadeiro ou falso sobre a validade de um único elemento.

4.2.4 Implementação de Um Data Warehouse Espacial

As etapas de integração de dados e de ETL são realizadas pelo aplicativo *Kettle*, conforme já descrito na seção 4.2. A interação com o *Kettle* se dá pelo *Spoon*, que é uma ferramenta gráfica onde as transformações de dados são desenhadas. Para iniciar os *Spoon*, basta executar o arquivo *spoon.bat* no Windows ou *spoon.sh* nos sistemas operacionais baseados no Unix.

Para criar uma nova transformação, basta ir ao painel *Main Tree*, clicar com o botão direito do *mouse* e selecionar *New*, conforme apresentado na figura 37.

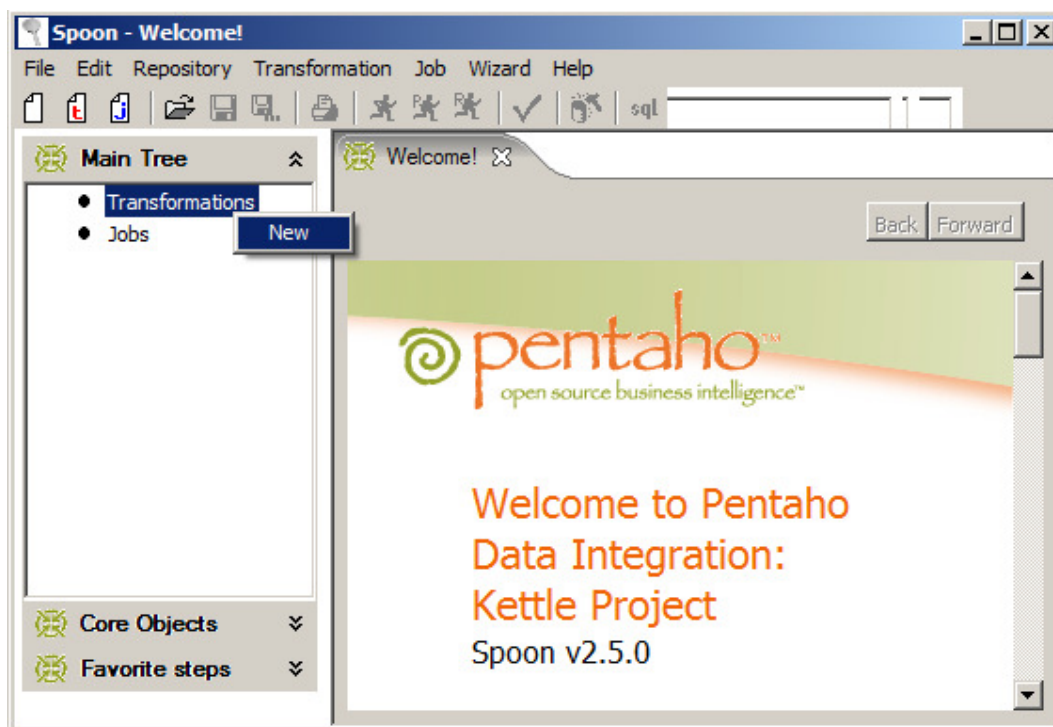


Figura 37: Criação de uma nova transformação no *Kettle*.

Tendo criado a nova transformação, é necessário configurar uma conexão com o banco de dados. Para realizar a configuração basta ir ao painel *Main Tree*, clicar com o botão direito do mouse em *Database Connections*, e selecionar a opção *New* e preencher os dados da base de dados a ser utilizada na tela que aparecerá, conforme mostrado na figura 38.

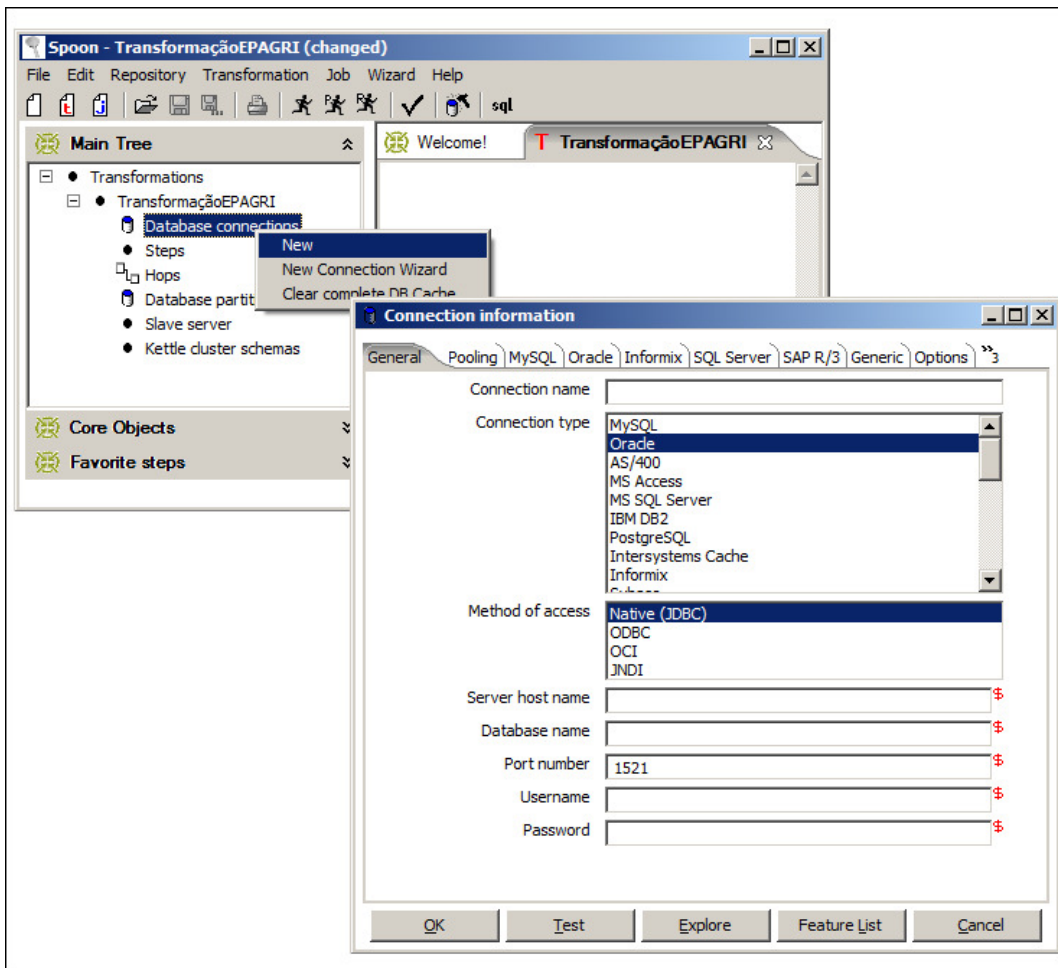


Figura 38: Configuração de uma nova conexão com o banco de dados no *Kettle*.

O uso do *Kettle* não é tão intuitivo quando o uso da ferramenta da *Microsoft*, mas é fácil de encontrar exemplos e explicações no site oficial, em fóruns e outros sites relacionados. Os elementos para construir os passos (no *Kettle*, chamados de *steps*) encontram-se no painel *Core Objects*.

A primeira ação a se fazer em uma transformação é definir uma entrada de dados. O *Kettle* é capaz de ler dados provenientes de diversas fontes, como: bases de dados, planilhas, arquivos XML, arquivos de texto, entre outros. Neste caso, a fonte de dados é uma base de dados Oracle, cuja conexão já foi configurada, então basta adicionar o objeto *Table Input* e em sua configuração selecionar a conexão com a base de dados que foi configurada e escolher os dados a serem extraídos da base de dados neste *step*. Podem existir diversos *steps* para acesso e extração de dados como este em uma mesma transformação. A figura 39 ilustra a configuração de um *step* que extrai

os dados CODMUNICIPIO, NOMEMUNICIPIO e sua componente espacial SPMUNICIPIO da tabela MUNICIPIO:

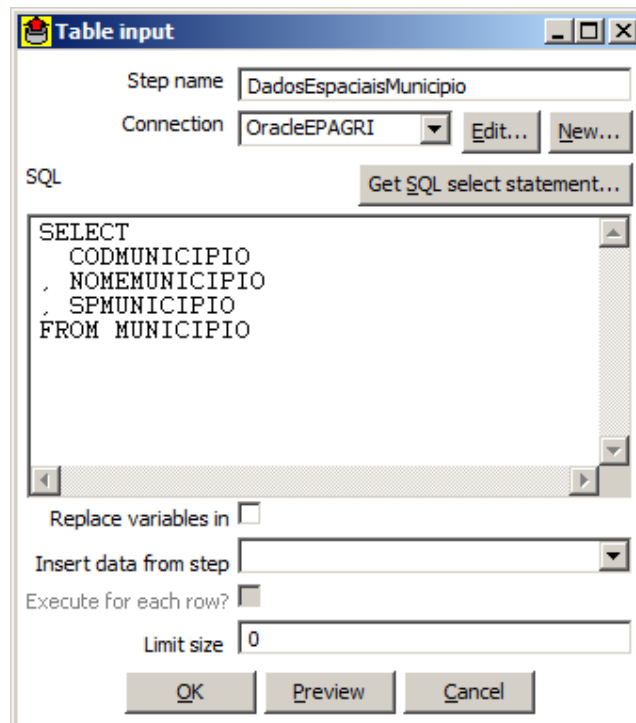


Figura 39: Selecionando os campos a serem extraídos da tabela MUNICIPIO.

Depois de escolhidos e extraídos, existem uma série de *steps* que podem ser realizados para transformar estes dados, bem como escolher a forma como estes dados serão exportados. Para o *output* de dados também há uma série de opções suportadas pelo *Kettle*, como: bases de dados, arquivos XML, planilhas, entre outros.

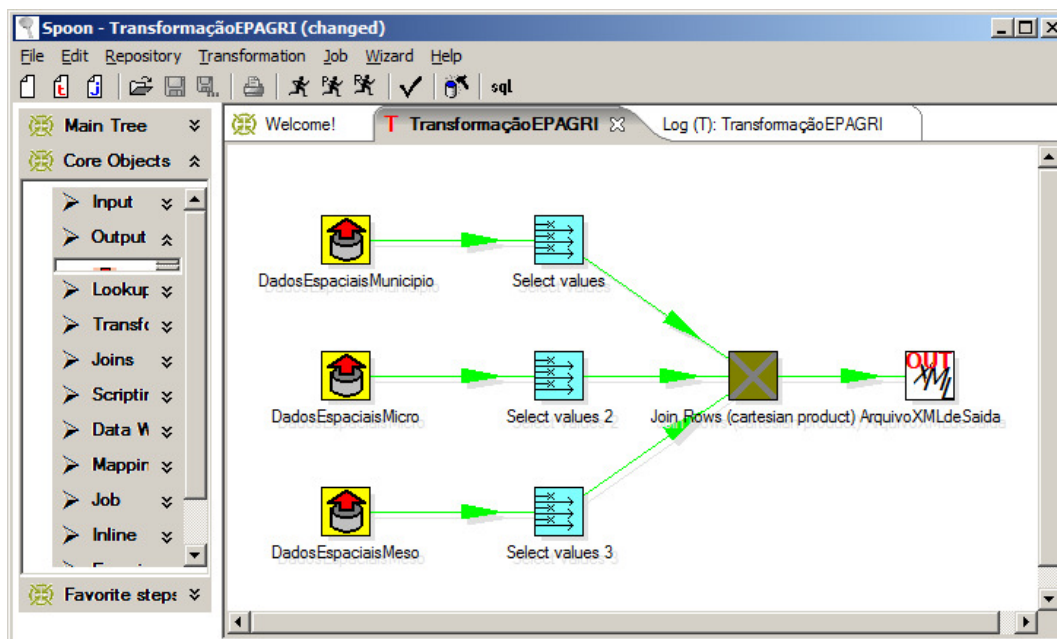


Figura 40: Exemplo de transformação no *Kettle*. Dados espaciais de três tabelas são extraídos e agrupados.

O problema encontrado no *GeoKettle* foi na extração dos componentes espaciais das tabelas. Independentemente do modo escolhido para a saída de dados, o componente espacial retorna sempre em branco. Nenhum erro é apresentado graficamente no *Spoon*, nem em seu arquivo de *log* de erros. A figura 41 ilustra o arquivo XML de saída de uma transformação com que apresenta o componente espacial em branco:

```

1  <Row><CODMESOIBGE>4201</CODMESOIBGE> <NOMEMESOIBGE>Oeste Catarinense
   </NOMEMESOIBGE>
2  <CODESTADO>42</CODESTADO> <SPMESOIBGE/> </Row>
3
4
5  <Row><CODMESOIBGE>4202</CODMESOIBGE> <NOMEMESOIBGE>Norte Catarinense
   </NOMEMESOIBGE>
6  <CODESTADO>42</CODESTADO> <SPMESOIBGE/> </Row>
7
8
9  <Row><CODMESOIBGE>4203</CODMESOIBGE> <NOMEMESOIBGE>Serrana
   </NOMEMESOIBGE>
10 <CODESTADO>42</CODESTADO> <SPMESOIBGE/> </Row>
11
12
13 <Row><CODMESOIBGE>4204</CODMESOIBGE> <NOMEMESOIBGE>Vale do Itajaí
   </NOMEMESOIBGE>
14 <CODESTADO>42</CODESTADO> <SPMESOIBGE/> </Row>
15
16

```

Figura 41: Arquivo XML com componente espacial em branco.

Apesar da grande e ativa comunidade de usuários e desenvolvedores da suíte de aplicativos *Pentaho* existente na internet, a comunidade de usuários de *GeoKettle* ainda é muito incipiente, visto que ainda é um projeto experimental. É importante ressaltar que no site do projeto *GeoKettle* (BADARD, 2008) é mencionado que o *GeoKettle* ainda está em testes, diversas funcionalidades ou compatibilidade com determinados padrões ainda não está implementadas ou devidamente testadas e ainda não recomendam seu uso em um ambiente de produção.

4.3 SpagoBI

SpagoBI é uma plataforma open source de *business intelligence* (BI). Esta plataforma é composta por aplicações que provêm integração de dados, ETL, geração de relatórios, análise de dados (OLAP), entre outros; seguindo a arquitetura apresentada na Figura 42.

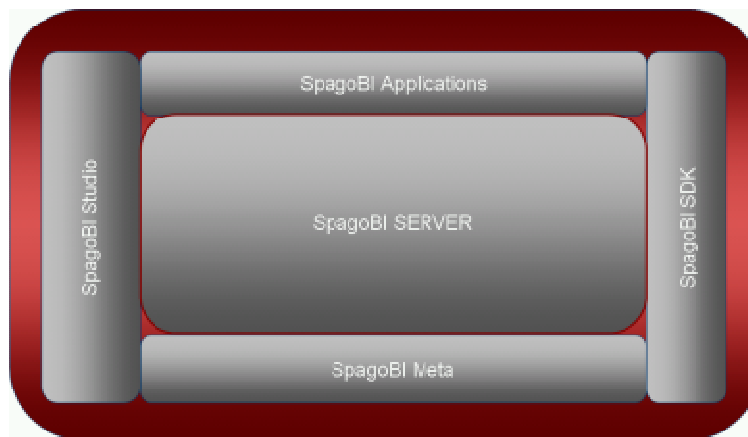


Figura 42: arquitetura da plataforma *SpagoBI* (SPAGOBI, 2009).

SpagoBI Server: é o principal módulo da *suíte*, e provê todas as funcionalidades de análise. Na figura 43 está ilustrada sua arquitetura:

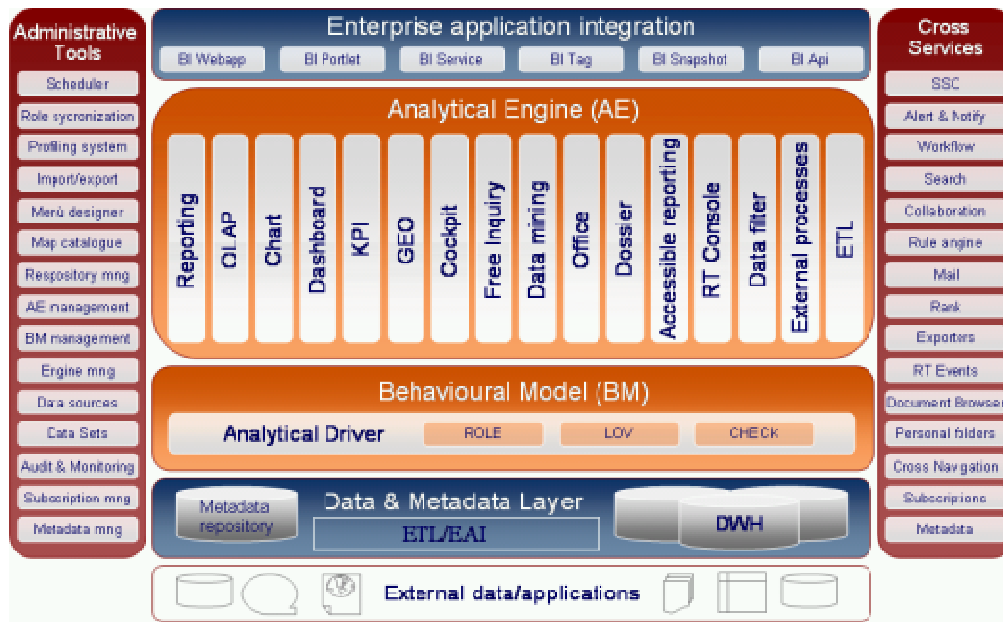


Figura 43: arquitetura do *SpagoBI Server* (SPAGOBI, 2009).

SpagoBI Studio: é um ambiente de desenvolvimento baseado no *Eclipse*. Ele permite a modificação dos documentos de análise (relatórios, OLAP, *dashboards* e mineração de dados).

SpagoBI Meta: é um módulo que gerencia os metadados técnicos e metadados do negócio, assim, permitindo a edição e a importação destes dados de ferramentas externas. Este módulo também oferece uma ferramenta de suporte à engenharia reversa de banco de dados.

SpagoBI SDK: ferramenta desenvolvida para a integração dos serviços prestados pelo servidor, bem como para a publicação destes documentos em um portal externo. O módulo *SpagoBI Studio* utiliza-se desde módulo para permitir o *download* e *upload* dos documentos de análise do servidor.

SpagoBI Applications: aplicação de ERP da plataforma *SpagoBI*.

4.3.1 Suporte a Extensões Espaciais

A documentação encontrada no *site* do *SpagoBI* é muito superficial do ponto de vista técnico sobre quais extensões espaciais são suportadas. O material encontrado no *site* limita-se a um vídeo demonstrando o uso da ferramenta, e os materiais encontrados na internet limitam-se a dizer que seu módulo espacial (*SpagoBI GeoReportEngine*) é capaz de lidar com o tipo de

dados *geometry*, mas em momento algum citam os tipos de dados espaciais, bem como operadores, suportados.

O *SpagoBIGeoReportEngine* é uma aplicação *web* para gerar relatórios integrando dados dimensionais, provenientes de um *data warehouse* e dados espaciais, que podem ser oriundos de um sistema de informações geográficas ou de um *data warehouse* espacial. É baseada o *GeoReport*, que compõe a *suíte GeoBI*.

A *suíte GeoBI* é desenvolvida pela empresa INOVA e visa integrar *business intelligence* com dados espaciais. Essa *suíte* é composta basicamente por três módulos: *GeoOLAP*, *GeoETL* e *GeoReporting*, responsáveis por consultas OLAP e visualização de resultados em mapas, ETL envolvendo dados espaciais e geração de relatórios, respectivamente, e é baseada nas *suítes PentahoBI* e *SpagoBI*, no que diz respeito a *business intelligence*. Em seu *site* (GEOBI, 2010), principal meio de divulgação e documentação da *suíte*, as informações também não são muito completas, e limitam-se a dizer que ele utiliza-se de um *engine* para lidar com dados espaciais, e que pode ser usado o *GeoKettle* como tal *engine*, comunicando-se com o *SpagoBIGeoReportEngine*.

4.3.2 Implementação de Um Data Warehouse Espacial

Conforme explicitado na seção anterior, o *GeoKettle* pode ser incorporado à *suíte SpagoBI*, assim, sua análise já foi apresentada 4.2.4 deste trabalho.

4.4 Critérios de Análise das Ferramentas e Comparativo

Esta seção apresenta o comparativo das *suítes* de ferramentas analisadas, de acordo com os critérios escolhidos. O objetivo é mostrar de forma direta as vantagens, desvantagens e características mais significativas para escolher uma *suíte* de ferramentas para a construção de um *data warehouse* espacial.

Quadro 2: quadro comparativo entre as ferramentas estudadas.

	<i>Microsoft SQL</i>	<i>PentahoBI</i>	<i>SpagoBI</i>
--	----------------------	------------------	----------------

	Server 2008		
Visualização de consultas OLAP em mapas	Feita internamente no SQL <i>Server Management Studio</i> .	Via navegador <i>web</i> .	Via navegador <i>web</i> .
Acoplamento entre os módulos	Alto nível de acoplamento entre os módulos, o que implica em um baixo nível de coesão.	Baixo acoplamento entre os módulos, permitindo maior versatilidade e integração ou substituição de outros módulos, como é o caso da incorporação do <i>GeoKettle</i> .	Baixo acoplamento entre os módulos, permitindo maior versatilidade e integração ou substituição de outros módulos, como é o caso da incorporação do <i>GeoKettle</i> .
Plataforma	<i>Windows</i>	Independente	Independente
Licença	Paga	<i>GNU LGPL</i>	<i>GNU LGPL</i>
Documentação	Fácil acesso e completa.	Pouco completa e imprecisa sistema <i>wiki</i> com poucas informações.	Praticamente inexistente. <i>Site</i> do projeto limita-se em mostrar “o que” a <i>suíte</i> faz e não em “como” faz.
Vantagens	Ferramenta intuitiva, boa documentação e riqueza de informações	Ferramenta gratuita, comunidade muito ativa na internet onde se pode obter muitas informações.	É gratuita.

Desvantagens	Ferramenta paga, mensagens de erro pouco informativas.	A versão gratuita (<i>community</i>) ainda existe, mas a versão paga (<i>enterprise</i>) é o foco da Pentaho no momento, assim, deixando a versão gratuita em segundo plano.	Documentação bastante escassa e ferramenta pouco intuitiva.
Tecnologia Utilizada na Interface com o Usuário	Aplicação <i>stand alone</i> proprietária <i>Microsoft SQL Server Business Intelligence Development Studio</i> .	Aplicação web em Java (J2EE).	Aplicação web em Java (J2EE), baseada na plataforma eXo Portal.

As tabelas 3 e 4 fazem um comparativo entre os dados espaciais e os operadores espaciais suportados pelo *Microsoft SQL Server 2008* e o *GeoKettle*, respectivamente. A comparação é feita somente entre a ferramenta da *Microsoft* e o *GeoKettle* pois tanto o *PentahoBI* quanto o *SpagoBI*, fazem o uso do *GeoKettle* para lidar com a extração, transformação e carga de dados espaciais, conforme explicitado nas seções 4.2.1 e 4.3.1.

Quadro 3: comparativo entre os tipos de dados espaciais suportados.

	<i>Microsoft SQL Server 2008</i>	<i>GeoKettle (utilizado no PentahoBI e no SpagoBI)</i>
Ponto	X	X
Multiponto	X	X
<i>LinearRing</i>		X

LineString	X	X
MultiLineString	X	X
Polígono	X	X
Multipolígono	X	X
GeometryCollection	X	X
Line	É suportado, mas é considerado como um <i>LineString</i> de apenas dois pontos.	É suportado, mas é considerado como um <i>LineString</i> de apenas dois pontos.

Quadro 4: comparativo entre os operadores espaciais suportados.

	Microsoft SQL Server 2008	GeoKettle (utilizado no PentahoBI e no SpagoBI)
Equals	X	X
Disjoint	X	X
Intersects	X	X
Touches	X	X
Crosses	X	X
Within	X	X
Contains	X	X
Overlaps	X	X
Relate	X	
Distance	X	

5. Trabalhos Relacionados

A maioria dos trabalhos relacionados aborda a modelagem de *data warehouses* espaciais e assuntos mais teóricos, não entrando em detalhes mais práticos de como estes modelos são implementados em determinadas ferramentas. Entre os trabalhos pesquisados, os que fazem análise de ferramentas são os trabalhos de MACDONALD e RUBICK (2007) e SOARES (2008).

Em seu trabalho, MACDONALD e RUBICK (2007) analisam, primeiramente, ferramentas de ETL e OLAP para *web* de licença livre isoladamente, e, logo após, fazem uma análise de *suítes* de ferramentas, incluindo *PentahoBI* e *SpagoBI*. Embora citem que em alguns desses aplicativos, ou *suítes*, seja possível incorporar dados espaciais, não entram em detalhes, pois está fora do escopo de seu trabalho. A seção de comparação entre as ferramentas é feita em duas partes: primeiramente, em um grau menor de granularidade, analisando servidores OLAP, clientes OLAP e *suítes*; após isso é apresentada a comparação entre as ferramentas OLAP. Os critérios para comparação entre *suítes* não são definidos claramente, visto que não era o principal foco do trabalho.

Outro trabalho relacionado é o trabalho de SOARES (2008) sobre criação de uma aplicação OLAP envolvendo dados espaciais e sua visualização cartográfica em ambiente *web*. Neste trabalho, o autor compara algumas ferramentas para implementação de protótipos, entrando em um nível de detalhe bastante técnico, como, por exemplo, as linguagens de comunicação de tais ferramentas. A visualização de resultados das consultas OLAP é feita com a integração de alguma ferramenta *GIS*, como *MapPoint*, *MapWindow*, *ArcGIS*, *AvisMap* e *Quantum GIS*.

Embora relacionados, a principal diferença entre este trabalho e os trabalhos aqui citados é que neste trabalho, o foco é apenas nas *suítes* de ferramentas para a implementação de *data warehouses* espaciais (já integrando as tecnologias de DW e SIG), as quais não estavam disponíveis na época de desenvolvimento dos outros trabalhos. Dessa forma, outros aplicativos que podem ser integrados às plataformas para a construção de *data*

warehouses e visualização de consultas OLAP estão fora do escopo deste trabalho.

6. Conclusões e Trabalhos Futuros

O principal objetivo deste trabalho era, a princípio, analisar a documentação de algumas *suítes* de ferramentas, compará-las e testá-las na implementação de um *data warehouse* usando a problemática oriunda da Epagri. Os principais problemas encontrados, que impediram a implementação deste *data warehouse* espacial nas ferramenta analisadas foram a falta de documentação apropriada, mensagens de erro pouco informativas e uma comunidade pouco ativa, particularmente no caso da ferramenta *SpagoBI*.

A contribuição deste trabalho fica centrada análise da documentação, informações coletadas nos *sites* oficiais e na tentativa de implementação de um *data warehouses* nas mesmas. Aqui foi colocada uma visão geral sobre extensões espaciais e, em cima disto, uma comparação entre as características das ferramentas. Como não foi possível implementar os modelos nas ferramentas propostas, não foi possível aprofundar a análise de alguns aspectos, pouco detalhados na documentação, e que seriam observados na ferramenta em execução. Com isso, é apresentado um panorama das principais ferramentas existentes e o que é suportado por cada uma delas.

Partindo deste trabalho, pode ser sugerido como trabalho futuro, que tenha como foco a implementação de um *data warehouse* espacial usando as ferramentas estudadas nesse trabalho. Outra possibilidade de trabalho futuro é um trabalho comparando outras *suítes* de ferramentas para a construção de *data warehouses* espaciais, tais como ferramentas da *Oracle*.

A integração entre *business intelligence* e dados espaciais é um assunto que ainda possui muitos pontos em aberto, principalmente no que diz respeito às ferramentas, tanto na inserção de dados espaciais diretamente em *data warehouses*, quanto na integração de um *data warehouse* com sistemas de informação geográficos.

REREFÊNCIAS BIBLOGRÁFICAS

- BADARD, Thierry. **GEOKETTLE**. Québec, 2008. Disponível em:
<http://geosoa.scg.ulaval.ca/en/index.php?module=pagemaster&PAGE_user_op=view_page&PAGE_id=17>. Acesso em: 07 dez. 2009.
- CASANOVA, Marco et al. **Bancos de Dados Geográficos**. Curitiba: Mundogeo, 2005.
- CÂMARA, Gilberto et al. **Anatomia de Sistemas de Informação Geográfica**. Departamento de Ciência da Computação da Unicamp, 1996.
- DAMIANI, Maria L.; SPACCAPIETRA, Stefano. **Spatial Data warehouse Modelling**, In *Processing And Managing Complex Data For Decision Support*, p. 1-27, IDEA Group.
- DEGGAU, Renato, Fileto, Renato. **Enriquecendo Data warehouses Espaciais com Descrições Semânticas**. In *Workshop de Teses e Dissertações em Bancos de Dados*, 8. Fortaleza, 2008.
- EGENHOFER, Max J., HERRING John R. **Categorizing Binary Topological Relations Between Regions, Lines, and Points in Geographical Databases**. *Technical Report, Department of Surveying Engineering, University of Maine, Orono, ME.*
- ELMASRI, Ramez; NAVATHE, Shamkant. **Fundamentals of Database Systems**. 4. ed. Nova Iorque: Addison Wesley, 2003.

- FIDALGO, Robson. **Uma Infra-estrutura para Integração de Modelos, Esquemas e Serviços Multidimensionais e Geográficos**. 2005. Tese de Doutorado (Ciência da Computação) - Universidade Federal de Pernambuco, Recife.
- GEOBI. Disponível em: <<http://www.geobi.org/>>. Acesso em: 21 abr. 2010.
- HAN, Jiawei; STEFANOVIC, Nebojsa; KOPERSKI, Krzysztof. **Selective materialization: An efficient method for spatial data cube construction**. In: *Proc. Pacific-Asia Conf. on Knowledge Discovery and Data Mining (PAKDD)*. 1998.
- INMON, William H. **Como Construir o Data warehouse**. Tradução Ana Maria Netto Guz. 2 ed. Rio de Janeiro: Campus, 1997.
- IBGE (Instituto Brasileiro de Geografia e Estatística). **Noções Básicas de Cartografia**. Rio de Janeiro, 1998. Disponível em: <http://www.ibge.gov.br/home/geociencias/cartografia/manual_nocoos/index.htm >. Acesso em: 08 ago. 2009.
- KIMBALL, Ralph. **Data warehouse Toolkit: The Complete Guide to Dimensional Modeling**. 2. ed. Nova Iorque: John Wiley & Sons, 1998.
- MELLO, João Alexandre Bonin de. **Uma proposta de modelo de dados para suporte ao processamento transacional e de Data Warehouse simultaneamente**. 2002. 101 f. Dissertação de Mestrado (Engenharia de Produção) – Universidade Federal de Santa Catarina, Florianópolis.
- MACDONALD, Giovani; RUBIK, João. **Pesquisa e Seleção de Ferramentas Livres e baseadas em padrões de sistemas abertos para a elaboração de interfaces OLAP sobre a Web**. 2007. 114 f. Monografia

de Conclusão de Curso (Ciência da Computação) - Universidade Federal de Santa Catarina, Florianópolis.

MSDN - MICROSOFT DEVELOPERS NETWORK. **Tipos de dados espaciais.** [S.l.]. Disponível em: <<http://msdn.microsoft.com/pt-br/library/bb964711.aspx>>. Acesso em: 08 ago. 2009.

MSDN - MICROSOFT DEVELOPERS NETWORK. **Introdução ao SQL Server Management Studio para Business Intelligence.** 2008. Disponível em: <<http://msdn.microsoft.com/pt-br/library/cc646025.aspx>>. Acesso em: 08 ago. 2009.

OGC (Open Geospatial Consortium). **OpenGIS® Implementation Specification for Geographic Information - Simple Feature Access - Part 1: Common Architecture.** Out. 2006. Disponível em: <<http://www.opengeospatial.org/standards/sfa>>. Acesso em 02 dez. 2009.

PENTAHO. Documentação da **Plataforma Pentaho Community.** [2006]. Disponível em: <<http://community.pentaho.com/javadoc/>>. Acesso em 26 nov. 2009.

RAO, F. et al. **Spatial hierarchy and OLAP-favored search in spatial data warehouse.** In: *Proceedings of the 6th ACM international Workshop on Data Warehousing and OLAP (DOLAP)*. Nova Iorque: ACM, 2003. p. 48-55.

RIBEIRO JUNIOR, Luiz Carlos Miyadaira; SARAIVA, Antonio Mauro; MURAKAMI, Edson. **Modelagem De Banco De Dados Espaciais Para Agricultura De Precisão.** In: IV Congresso Brasileiro da Sociedade Brasileira de Informática Aplicada A Agropecuária e Agroindústria, 4., 2003, Porto Seguro: [s.n]. p. 2 - 3.

- SILVA, Joel. **GEOMDQL**: Uma linguagem de consulta geográfica e multidimensional. 2008. 194 f. Tese de Doutorado (Ciência da Computação) – Universidade Federal de Pernambuco, Recife.
- SIQUEIRA, Thiago Luís Lopes ; CIFERRI, Ricardo Rodrigues ; TIMES, Valéria Cesário . **I-DWE**: Uma Estrutura de Indexação para *Data warehouse* Espacial. In: *Workshop* de Teses e Dissertações em Bancos de Dados, 7., Campinas, 2008. p 79-84.
- SOARES, Miguel. Uma **aplicação OLAP com visualização cartográfica via Web**. 2008. 67 f. Monografia de Conclusão de Curso (Ciência da Computação) - Universidade Federal de Santa Catarina, Florianópolis.
- SPAGOBI. **The Suite Architecture**. 2009. Disponível em:
<<http://www.spagoworld.org/xwiki/bin/view/SpagoBI/Architecture>>.
Acesso em: Acesso em 26 nov. 2009.
- VIVID Solutions. **JTS Topology Suite Technical Specifications 1.4**. [S.l.], 2003. Disponível em:
<<http://www.vividsolutions.com/jts/bin/JTS%20Technical%20Specs.pdf>>.
Acesso em: 05 dez. 2009.
- WORBOYS, Michael. **GIS: A Computing Perspective**. 2. ed. Londres: Taylor and Francis, 1995