MINISTÉRIO DA CIÊNCIA E TECNOLOGIA
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

Bancos de Dados Geográficos – 2014

Lúbia Vinhas

Introdução

- Essa disciplina visa dar aos alunos dos cursos CAP e SERE a compreensão sobre conceitos de Sistemas Gerenciadores de Bancos de Dados e sua aplicação ao domínio da Geoinformação
- Proponho um enfoque híbrido: não é um curso forte de SGBD para alunos da Computação, nem um curso de somente de aplicação para alunos do Sensoriamento Remoto
- Espera-se que os alunos do Sensoriamento Remoto aprendam a modelar corretamente seus problemas, possam avaliar diferentes sistemas e fazer o melhor uso deles
- Espera-se que os alunos do Computação aprendam um uso não convencional de SGBDs relacionais e objeto-relacionais, de modo a serem capazes de produzir ferramentas mais adequadas ao domínio da geoinformação

Dinâmica do curso

- Docente Lúbia Vinhas (lubia@dpi.inpe.br), Karine Reis Ferreira (karine@dpi.inpe.br) e Gilberto Ribeiro de Queiroz (gribeiro@dpi.inpe.br)
- As aulas são as segundas e quartas, das 08:15 as 10:00 no auditório do LabGeo. Se houver necessidade de mudança de horário ou local, avisarei com antecedência por e-mail até o dia anterior a aula.
- Página da disciplina: http://wiki.dpi.inpe.br/doku.php?id=cap349
- Alunos isolados: tem até o dia 28/junho para decidir se ficam ou saem. Quem ficar depois disso deve acompanhar as aulas e fazer as tarefas e avaliações propostas e terá nota.

Avaliação

- A nota final do curso será dada pela seguinte fórmula:

$$NF = NL*0.2 + NP*0.3 + NT*0.5$$

Onde:

NF = Notal final

NL = Nota de listas (individual)

NP = Nota da prova (individual)

NT = Nota do trabalho <u>proposto pelo ALUNO</u> (individual)

Referências

1. Shekkar, S. and S. Chawla (2003). Spatial databases - a tour. Upper Saddle River, NJ, USA, Prentice-Hall
2. Rigaux, P., M. Scholl, et al. (2002). Spatial Databases with Application to GIS. San Francisco, Morgan Kaufman
3. Casanova, M., G. Câmara, et al. (2005). Bancos de Dados Geográficos. Curitiba
4. Yeung and Brent Hall (2011). Spatial DatabaseSystems - Design, Implementation and Project Management, The GeoJournal Library.

- **Notas de Aulas**: serão disponibilizadas na wiki conforme forem sendo criadas e/ou revisadas.

MINISTÉRIO DA CIÊNCIA E TECNOLOGIA
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

Motivação

Dados Espaciais

- *"GIScience is defined here as the study of geographic information, how to collect, analyse, manage and visualise it."* – Taylor & Reitsma, IJGIS, 2013
- Dados Espaciais
  - Dados que contém a uma localização espacial em algum sistema de referência. Informação sobre a localização e forma (representadas por coordenadas) e as relações (topológicas) entre feições geográficas
- Sistemas de Referência
  - São basicamente as diferentes perspectivas de um observador a descrição de medidas (ex. posição) e sistemas de coordenadas são as diferentes formas de descrever medidas sob essas perspectivas
- Dados Geográficos ou Geoespaciais
  - São aqueles onde o sistema de referência é a superfície da terra

Dados Espaciais

A nível conceitual, um objeto geográfico corresponde a um entidade do mundo real que possui duas componentes:

1. um conjunto de atributos alfa-numéricos, ou **atributos descritivos**
2. uma **componente espacial**, que pode conter a geometria (localização e forma) e a topologia (relacionamentos espaciais com outros objetos)

Objetos geográficos podem ser atômicos ou complexos

Como são obtidos os dados espaciais:

- através de levantamentos de campo (ex. GPS) e através de sensores remotos
- digitalizados a partir de mapas já existentes

Dados espaciais

- De acordo com algumas fontes, 80% dos dados existentes possuem uma componente espacial (pense em endereços por exemplo)
- Enormes quantidades de dados espaciais são gerados por sistemas de observação da terra: ex: dados do CBERS, LANDSAT, SRTM, etc.
- Vários serviços como MapQuest, Yahoo! Maps, Google Maps, Google
- Os serviços geográficos estão crescendo:
  - Planejamento de rotas
  - Observação e Mitigação de desastres naturais
  - Observação de crimes
  - Simulação de efeitos ambientais
  - Serviços Baseados em Localização em geral
- Em geral, é necessário integrar informações espaciais em diferentes contextos.

Exemplos de Dados Espaciais

Imagens SR

Dados Temáticos

Fotos aéreas

Coisas móveis

Dados de Topografia

Aplicações sobre dados espaciais

Cartografia

Uso da Terra

Monitoramento ambiental

Saúde

Outras...

- Outras aplicações incluem:
  - Cadastro urbano
  - Desenvolvimento imobiliário
  - Desenvolvimento econômico
  - Marketing
  - Navegação
  - Climatologia
  - Gerenciamento de aparelhos públicos
  - Redes de transporte
  - Biodiversidade
  - Monitoramento de desastres ambientais
  - ...
- Aplicativos geográficos tratam fundamentalmente da solução de problemas. São usados em vários níveis, desde indústrias especializadas até governos e academia

Aplicativos geográficos

- Softwares que permitem **armazenar**, **gerenciar** e visualizar dados espaciais e fornece ferramentas de **análise**. Exemplos de tarefas:

1. visualização e exploração de dados;
2. criação de dados;
3. edição de dados;
4. armazenamento de dados;
5. integração de dados de diferentes fontes;
6. consultas para selecionar parte dos dados;
7. análise de dados: extração de novas informações a partir dos dados existentes;
8. transformação de dados
9. criação de mapas

Categorias de aplicativos geográficos

- SIGs Desktop: software instalado em um computador pessoal. Não permite acesso remoto por outros computadores clientes
- Sistemas Gerenciadores de Banco de Dados Espacial: fornece uma alternativa a manipulação de dados geográficos armazenados em arquivos
- Servidors de Mapas na Web: oferece um serviço de produção de mapas que dá acesso a dados espaciais
- Servidores SIG: expõe processamento típicos de aplicativos geográficos
- Clientes SIG
- SIG móvel: roda em plataforma móveis como tablets ou smartphones;
- Bibliotecas e extensões: oferecem funcionalidades que podem ser usadas para a construção de outras aplicações.

Categorias de aplicativos geográficos

- SIGs Desktop (GIS)
- Sistemas Gerenciadores de Banco de Dados Espaciais
- Servidor de Mapas na Web
- Servidores SIG
- Cliente SIG para Web
- SIG móvel
- Bibliotecas e extensões

Banco de Dados

Um **Banco de Dados** (BD) é uma coleção volumosa de dados relacionados entre si, em um ambiente computacional

Esses dados são persistentes, ou seja, sobrevivem a falhas de hardware ou software

Um **Sistema Gerenciador de Banco de Dados** (SGBD) é um sistema computacional que gerencia a estrutura do banco de dados e controla o acesso aos dados armazenados

Banco de Dados

Um **Banco de Dados** (BD) é uma coleção volumosa de dados relacionados entre si, em um ambiente computacional

Aplicações geográficas atualmente são intensivas tanto em termos de dados quanto de computação. Requerem o armazenamento de grandes volumes de dados dados tradicionais (alfanuméricos) e não tradicionais (objetos geométricos, imagens, séries temporais).

Nenhuma aplicação geográfica irá rodar de maneira eficiente sem **um Banco de dados Espacial** acoplado.

Visão conceitual dos sistemas de informação espacial: 4 componentes centrais que servem a um ou mais domínios de aplicação. Banco de Dados Geográfico é a componente principal.

[Yeung and Hall, 2011]

Sistema Gerenciador de Banco de Dados

- Um **SGBD** facilita as seguintes tarefas:
  - **Definição** do BD: tipos de dados, estruturas e restrições que devem ser consideradas
  - **Construção** do BD: inserção propriamente dos dados no meio persistente;
  - **Manipulação** do BD
  - **Consulta** ao BD: recuperar uma informação específica
  - **Manutenção** ao BD: alterar uma informação armazenada
- São requisitos de um SGBD:
  - Facilidade de uso
  - Correção
  - Facilidade de manutenção
  - Confiabilidade
  - Segurança
  - Desempenho

Sistema Gerenciador de Banco de Dados Espacial

- Até o início dos anos 90 os SIGs seguina uma abordagem baseada em arquivos de dados
- A partir daí começa uma aproximação entre os produtores de SIG e os produtores de Sistemas Gerenciadores de Bancos de Dados
- Um SGBD Espacial típico hoje em dia é um SGBD comum com capacidades adicionais para manipular dados espaciais:
  - Tipos espaciais
  - Indexação espacial
  - Operadores espaciais
  - Rotinas e aplicações espaciais (ex. carga de dados)

SIG x Banco de Dados Espacial

[Yeung and Hall, 2011]

Interfaces para SGBDs

GUI

Prompt

API

```
int main()
{
…
}
```

SGBB

Interfaces para bancos de dados

Interfaces para bancos de dados

Interfaces para bancos de dados

Formulário

Sistemas de Informação Geográfica

- Um SIG é um software que permite **armazenar**, **gerenciar** e visualizar dados espaciais e fornece ferramentas de **análise** como:
  - **Consulta**: por região, por coordenadas, classificação
  - **MNT**: declividade, rede de drenagem, bacias
  - **Rede**: conectividade, caminho mínimo
  - **Distribuição**: detecção de mudanças, proximidade
  - **Análise/Estatística espacial**: padrões, auto-correlação
  - **Medidas**: distância, forma, adjacência, direção, perímetro
- Exemplos de SIG
  - Livres de licença: TerraView, SPRING, QuantunGIS, gvSIG, etc.
  - Proprietários: ArcGIS, MapInfo, etc.

Visão Geral de um SIG

BANCO DE DADOS GEOGRÁFICO

Interfaces para SGBDs

SIG

Evolução das arquiteturas de SIG

Daqui pra diante…

- Vamos estudar como os SGBD's funcionam
- Como a arquitetura integrada funciona
- Exercitar o uso do SGBD e do SIG
- O que ganhamos em termos de funcionalidade
- Questões de interoperabilidade
- Novas possibilidades
- O caso da internet
- etc…

- ESSE CURSO É UMA FERRAMENTA.

Qual software iremos usar?

Fonte: Steiniger and Hunter, 2013

MINISTÉRIO DA CIÊNCIA E TECNOLOGIA
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

Representações para dados geográficos

Lúbia Vinhas

"…GIScientists study the representations of the real world rather than the real world itself."

Taylor & Reitsma, 2013

Geoprocessamento

Representa a área do conhecimento que utiliza técnicas matemáticas e computacionais para tratar dados espaciais, produzindo informações relevantes para tomada de decisão

Geoprocessamento

Exemplo pioneiro

Epidemia de cólera em Londres em 1854:

Mapa criado pelo Dr. John Snow mostrando um cluster de ocorrências ao redor do poço de fornecimento de água

Fechamento do poço coincide com o retrocesso da epidemia, corroborando hipótese da transmissão pela água

Histórico

- **Anos 50**: redução de custos de produção e manutenção de mapas. Aplicações específicas foram desenvolvidas na Inglaterra (botânica) e nos EUA (tráfego)
- **Anos 60**: programa governamental no Canadá para inventário de recursos naturais
- **Anos 70**: criação da expressão Geographic Information System - GIS (SIG – Sistema de Informação Geográfica). Empresas começam a comercializar SIGs, alto custo e para computadores de grande porte
- **Anos 80**: computadores pessoais e estações de trabalho popularizam acesso aos SIG's. Criação de centros específicos como NCGIA marcam o estabelecimento do Geoprocessamento
- **Anos 90**: interesse em software livre de licença chega, aumentando uso dos SIG's
- **Anos 2000**: ênfase em acesso e publicação de dados no ambiente da Internet

Dados Espaciais

- Dados que contém uma **localização** espacial em algum **sistema de referência**. Contém informação sobre a localização, forma e as relações topológicas entre feições de interesse.

- Sistemas de Referência
  - São basicamente as diferentes perspectivas de um observador quanto a descrição de medidas (ex. posição) . Sistemas de coordenadas são as diferentes formas de descrever medidas sob essas perspectivas

- Dados Geográficos ou Geoespaciais
  - São aqueles onde o sistema de referência é a superfície da terra

Dados espaciais ou geográficos

Caracterizam-se por: **localização geográfica** (onde o fenômeno ocorre) e **atributos** (descrição do fenômeno)

Foto mostra massa de ar frio cobrindo o Sudeste brasileiro. Foto: Nasa

Paradigma dos 4 universos

- Permite traduzir o mundo real para o ambiente computacional :
  - Mundo Real: fenômenos a serem representados (cadastro urbano, vegetação, solos)
  - Universo Conceitual: distinção entre classes formais de fenômenos contínuos e objetos individualizáveis (campos e objetos)
  - Universo de Representação: diferentes representações geométricas (matrizes e vetores)
  - Universo de implementação: estruturas de dados e linguagens de programação (*R-tree* e *Quad-tree*)

Universo do Mundo Real

- Geoprocessamento manipula dados de naturezas e fontes distintas
- São organizados na forma de **camadas** (ou mapas). Um mapa é um modelo simplificado da realidade. Uma representação, em escala, de uma seleção de entidades abstratas relacionadas com a superfície da Terra
- Modelo que se interpõe entre a realidade e a base de dados de uma aplicação geográfica

Exemplo camadas

"*The Potteries*" ("As cidades ceramistas") conjunto de 6 cidades da Inglaterra com diversas indústrias relacionadas a cerâmica

A região se desenvolveu durante a revolução industrial inglesa

As comunidades locais produziam produtos de alta qualidade, mesmo em condições menos favoráveis

Baseado em Worboys & Duckhan 2004

Inventário de recursos

Dado o patrimônio cultural da área, a indústria local de turismo é significante

Camada com pontos de interesse cultural, equipamentos de recreação presentes na região e combiná-los com detalhes de infra-estrutura de transporte e hospedagem

Baseado em Worboys & Duckhan 2004

Análises de redes

Deseja-se encontrar uma rota, usando as rodovias principais, para visitar cada cidade (e o Museu da Cidade) uma vez, minimizando o tempo de viagem

É necessário uma rede que diga o tempo de viagem entre as cidades.

Gerado a partir do tempo médio de viagem nas vias principais mostradas na **camada de rede viária**

Baseado em Worboys & Duckhan 2004

Análises de terreno

Comunidades locais estão interessadas no impacto visual causado pela possível abertura de novas minas de carvão

Análise de visibilidade: medida o tamanho da população local dentro de um ângulo de visada (um mapa de todos os pontos visíveis a partir de uma dada localização)

Análise de terreno é geralmente baseada em uma **camada com dados de elevação** em localizações pontuais

Isolinhas de altimetria

Projeção perspectiva a partir de um ponto de visada. Regiões mais escuras marcam as áreas que não estarão visíveis.

Baseado em Worboys & Duckhan 2004

Sobreposição de camadas

Determinar o potencial de diferentes localizações para a extração de areia e cascalho

Apresentar e analisar camadas de:

- Geologia
- Estrutura urbana
- Lençol freático
- Zoneamento

Localização dos depósitos de areia e cascalho

Baseado em Worboys & Duckhan 2004

Sobreposição de camadas

Consulta: encontre as localizações que estão a 0.5 km de uma rodovia principal, em uma área não construída, com depósitos de areia/cascalho

0.5 km zona de influência das rodovias principais

Depósitos conhecidos de areia e cascalho

Áreas escuras indicam áreas não construídas

Interseção das 3 camadas resultando nas áreas que atendem a consulta

Baseado em Worboys & Duckhan 2004

Análises de localização

Alocação de hospitais na área das *Potteries*

- Construa a vizinhança dos hospitais existentes, baseado nas suas posições e os tempos de viagem até elas
- É então possível suportar melhor decisões sobre o fechamento, realocação ou a criação de um hospital

a. Isolinhas de mesma distância

b. Diagrama de Voronoi

Baseado em Worboys & Duckhan 2004

Tipos de mapas

- Mapas **Temáticos**: informações **qualitativas** sobre o espaço. Ex: Mapa de uso do solo ou Mapa de vegetação
- Mapas **Numéricos**: informações **quantitativas** sobre o espaço. Ex: Grade com valores de altimetria
- Mapas **Cadastrais**: informações sobre objetos discretos do mundo. Ex: Lotes urbanos com sua localização e seus atributos.
- Mapa de **Redes**: informações sobre objetos discretos que forma redes. Ex. Rede elétrica (postes e linhas de transmissão)
- **Imagens**: informações numéricas obtidas por sensores remotos. Ex: Fotografias aéreas, imagens de satélites e radares

IMAGENS

Aprisionar a luz

- Câmera de orifício (*pinhole camera*)
  - Século 4 AC – gregos
  - Século 10 DC – Ibn al-Haytham
- Câmera escura (*dark chamber*)
  - Giovanni Baptista Della Porta, 1558

(*G&W-07*)

(*Wiki-08*)

Fotografia

- Joseph Nicéphore Niépce (1765-1833)
- Heliografia (gravar com o sol)

View from the Window at Le Gras (1826)

Primeira imagem digital (?)

- Russel Kirsch - 1957
- Standards Eastern Automatic Computer (SEAC),
- National Bureau of Standards (NBS)
  (Hoje: National Institute of Standards and Technology – NIST).
- 176×176 píxeis e 5cm × 5cm

http://www.nist.gov/public_affairs/techbeat/tb2007_0524.htm

O início do processamento digital de imagens

- Anos 60
- Computadores suficientemente poderosos
- Programa espacial americano
- JPL – Jet Propulsion Laboratory
- Imagens da Lua pelo Ranger-7

Ranger-7 (jul,1969)

Satélites

- LANDSAT 80, 60 / 30 / 15 m
- SPOT-5 20 / 10 m , 5 / 2.5 m
- Ikonos 4 / 1 m
- Quickbird 2.4 / 0.6 m
- WorldView-1 0.5 m
- WorldView-2 1.84 / 0.46 m
- GeoEye-1 1.65 / 0.41 m (ex-Orb view)
- CBERS 2B / 20m, 80 m

landsat.gsfc.nasa.gov/

http://www.spot.com/home

Spot-5 2.5 m Trípoli

Aquisição de Imagens

Níveis de quantização típicos: 1, 8, 11 bits / píxel

Espectro Eletromagnético

-

Espectro Eletromagnético

(*Wikipedia-nov09*)

Curvas Espectrais

Curvas espectrais

Exemplos: ASTER Spectral Library
http://speclib.jpl.nasa.gov/

Escala dos Tons de Cinza

- LANDSAT-5 Washington-DC band 4

Infravermelho próximo

0.76-0.90 μm

(8 bits/píxel)

(G&W-08)

Colorida (True Color)

- LANDSAT-5 Washington-DC (R,G,B) => (3,2,1)

R Vermelho

G Verde

B Azul

Colorida (Falsa Cor)

- LANDSAT-5 Washington-DC (R,G,B) =>(5,4,3)

R Infravermelho médio

G Infravermelho próximo

B Vermelho

Colorida (Falsa Cor)

- LANDSAT-5 Washington-DC (R,G,B) => (4,3,2)

R Infravermelho próximo

G Vermelho

B Verde

Multispectral

- Imagem LANDSAT-TM

Multiespectral

- Bandas LANDSAT-TM

Band No.	Name	Wavelength (μm)	Characteristics and Uses
1	Visible blue	0.45–0.52	Maximum water penetration
2	Visible green	0.52–0.60	Good for measuring plant vigor
3	Visible red	0.63–0.69	Vegetation discrimination
4	Near infrared	0.76–0.90	Biomass and shoreline mapping
5	Middle infrared	1.55–1.75	Moisture content of soil and vegetation
6	Thermal infrared	10.4–12.5	Soil moisture; thermal mapping
7	Middle infrared	2.08–2.35	Mineral mapping

Temática (pseudo-cor)

- Imagem em pseudo-cor (mapeamento de tom de cinza para cor)

Quickbird pan+multiespectral
(0.6/2.4m)

Imagem temática
(classificação maxver)

4 dimensões de resolução

- Espacial
- Espectral
- Radiométrica
- Temporal

CBERS-2B

Imagem da Câmera CCD de alta resolução de Manaus

Imagem do WFI (Represa de Itaipu)

A cabeça óptica do WFI

Características da Câmera Imageadora de Alta Resolução CCD	
Bandas espectrais	0,51 - 0,73 µm (pan) 0,45 - 0,52 µm (azul) 0,52 - 0,59 µm (verde) 0,63 - 0,69 µm (vermelho) 0,77 - 0,89 µm (infravermelho próximo)
Campo de Visada	8,3º
Resolução espacial	20 x 20 m
Largura da faixa imageada	113 km
Capacidade de apontamento do espelho	±32º
Resolução temporal	26 dias com visada vertical (3 dias com visada lateral)
Frequência da portadora de RF	8103 MHz e 8321 MHz
Taxa de dados da imagem	2 x 53 Mbit/s
Potência Efetiva Isotrópica Irradiada	43 dBm

Características do Imageador de Amplo Campo de Visada WFI	
Bandas espectrais	0,63 - 0,69 µm (vermelho) 0,77 - 0,89 µm (infra-vermelho)
Campo de Visada	60º
Resolução espacial	260 x 260 m
Largura da faixa imageada	890 km
Resolução temporal	5 dias
Frequência da portadora de RF	8203,35 MHz
Taxa de dados da imagem	1,1 Mbit/s
Potência Efetiva Isotrópica Irradiada	31,8 dBm

Mapas temáticos

- Distribuição espacial qualitativa da grandeza ou atributo em estudo
- Os valores da grandeza podem ser:
  - nominal: lista de valores. Ex. possíveis classes de vegetação em um mapa de vegetação {floresta, cerrado, desmatamento}
  - ordinal: escala de medida. Ex. fatias de declividade {0-5%, 5-10%, 10-15%, 15-30%, >30%}

Mapas Numéricos

- Distribuição espacial quantitativa da grandeza em estudo
- Os valores da grandeza podem ser:
  - intervalo dentro de uma referência arbitrária. Ex. Altimetria, batimetria, temperatura em graus Celsius
  - razão: referência natural. Ex. Peso
- Localização espacial pode ser representada por grades regulares ou triangulares

Mapas cadastrais

- Contém informações sobre **objetos discretos** do mundo
- Exemplos: cadastros de países, de lotes, de propriedades rurais

País	PIB (US$bn)	Pop (milhões)
Brasil	350	159
Argentina	295	34
Chile	45	14

Mapas de rede

- Contêm objetos cuja referência geográfica está associada a nós ou arcos que formam uma **topologia de rede**

Paradigma de 4 universos

- Permite traduzir o mundo real para o ambiente computacional :
  - Mundo Real: fenômenos a serem representados (cadastro urbano, vegetação, solos)
  - Universo Conceitual: distinção entre classes formais de fenômenos contínuos e objetos individualizáveis (campos e objetos)
  - Universo de Representação: diferentes representações geométricas (matrizes e vetores)
  - Universo de implementação: estruturas de dados e linguagens de programação (*R-tree* e *Quad-tree*)

Universo conceitual: campo ou geo-campo

- <u>Geo-Campo</u>: $f = [R, A, \lambda]$, onde: $R \subset \Re^2$, $A = \{ A_1, A_2, ..., A_n\}$ é um conjunto atributos e λ: $R \rightarrow A$ mapeia pontos de R para valores em A
- Representa a distribuição espacial de uma variável que possui valores em todos os pontos pertencentes a R, num dado tempo *t*
- Se o contra-domínio A de *f* é um conjunto enumerável temos um dado temático. Ex. Mapa de cobertura vegetal
- Se o contra-domínio A de *f* é um conjunto de valores contínuos ($-\infty$ a $+\infty$) temos um dado numérico. Ex. mapa de aeromagnetometria
- Imagens são uma especialização de dados numérico

Definições auxiliares

- Região Geográfica (R) - uma superfície qualquer pertencente ao espaço geográfico, que pode ser representada num plano vetorial ou reticulado, dependente de uma projeção cartográfica
- Plano de informação - suporte para a representação geográfica de diferentes tipos de dados geográficos
  - É o lugar geométrico de um conjunto de dados geográficos - um geo-campo ou um mapa de geo-objetos
- Banco de Dados Geográficos - composto por conjuntos de planos de informação, um conjunto de geo-objetos e um conjunto de objetos não-espaciais

Universo conceitual: objeto ou geo-objetos

- Dadas as regiões geográficas $R_1, R_2, .. R_n$ ($R_i \subset \Re^2$) e o conjunto de atributos convencionais $\{A_1, ..., A_n\}$
- Um *Geo-Objeto* *go* = $[r_1, r_2, ..., r_n, a_1, a_2, ..., a_n]$, onde $r_i \subseteq R_i$ é uma parte de uma região geográfica R_i e $a_i \in A_i$ é um valor particular do atributo A_i
- É um elemento único que possui atributos não-espaciais e está associado a múltiplas localizações geográficas

País	PIB (US$bn)	Pop (milhões)
Brasil	350	159
Argentina	295	34
Chile	45	14

Geo-objetos

- Um geo-objeto pode possuir múltiplas representações dependendo da projeção cartográfica, escala ou instante de tempo

id	nome
	Amazonas
	Xingu

id	nome
1	Yanomami
2	Waimiri
3	Kayapó

Universo conceitual: objetos não-espaciais

- Um objeto não-espacial é um objeto que não possui localizações espaciais associadas
- Informações não georeferenciada agregada a uma aplicação geográfica

Resumo do universo conceitual

- Campos ou geo-campos (variáveis geográficas continuas)
  - Temático
  - Numérico
  - Imagem

- Objetos ou geo-objetos (variáveis geográficas discretas)
  - Cadastral
  - Redes

Mapa Lotes

Lotes

Geoid	Dono	Ender	Cadastro
22	Guimaraes	P. Chuí, 768	250186

Paradigma dos 4 universos

- Permite traduzir o mundo real para o ambiente computacional :
  - Mundo Real: fenômenos a serem representados (cadastro urbano, vegetação, solos)
  - Universo Conceitual: distinção entre classes formais de fenômenos contínuos e objetos individualizáveis (campos e objetos)
  - Universo de Representação: diferentes representações geométricas (matrizes e vetores)
  - Universo de implementação: estruturas de dados e linguagens de programação (*R-tree* e *Quad-tree*)

Universo de Representação

- Define as possíveis formas de representar no mundo digital os modelos do universo conceitual. Podem ser de dois tipos:
  - Vetoriais
  - Matriciais

(a) matricial (b) vetorial

Representação vetorial

- A localização e a aparência gráfica dos objetos são representadas por um ou mais pares de coordenadas que formam elementos gráficos

- Elementos da representação vetorial podem ser compostos

Representação vetorial

- Componentes de uma representação vetorial: ponto, linha, região
  - Ex: Região M={1,2,3,4,5}, formada pelas linhas 1, 2, 3, 4 e 5

- **Topologia**: descreve relações espaciais entre objetos, que são invariantes a rotação e translação
  - Ex. Toulose fica **Dentro** da região M

Representação vetorial

- Topologia Arco-Nó
  - Representa uma rede linear conectada
  - Nó: representa a intersecção entre linhas, são os pontos iniciais e finais
  - Linhas são conectadas

- Topologia Arco-Nó-Polígono
  - Representa elementos gráficos do tipo área ou partições do espaço

Representação matricial

- Divide a área sendo representada em uma matriz de células. A cada célula atribui-se um valor que o relaciona com o fênomeno sendo representado

10	2	1	5
6	3	4	10
3	10	94	3
11	2	7	0

10.5	2.3	1.9	5.4
6.1	3.7	4.6	10.1
3.1	10.0	94	3.4
11.6	2.3	7.9	0.5

linhas
colunas
tamanho das células
resolução dos valores

Comparação entre representações

- Vetorial
  - preserva relacionamentos topológicos
  - associa atributos a elementos gráficos
  - melhor exatidão e eficiência de armazenamento
- Matricial
  - fenômenos variantes no espaço
  - adequado para simulação e modelagem
  - processamento mais rápido e simples
  - maior gasto em armazenamento

MINISTÉRIO DA CIÊNCIA E TECNOLOGIA
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

Representações para dados geográficos

Lúbia Vinhas

"…GIScientists study the representations of the real world rather than the real world itself."

Taylor & Reitsma, 2013

Geoprocessamento

Representa a área do conhecimento que utiliza técnicas matemáticas e computacionais para tratar dados espaciais, produzindo informações relevantes para tomada de decisão

Geoprocessamento

Exemplo pioneiro

Epidemia de cólera em Londres em 1854:

Mapa criado pelo Dr. John Snow mostrando um cluster de ocorrências ao redor do poço de fornecimento de água

Fechamento do poço coincide com o retrocesso da epidemia, corroborando hipótese da transmissão pela água

Histórico

- **Anos 50**: redução de custos de produção e manutenção de mapas. Aplicações específicas foram desenvolvidas na Inglaterra (botânica) e nos EUA (tráfego)
- **Anos 60**: programa governamental no Canadá para inventário de recursos naturais
- **Anos 70**: criação da expressão Geographic Information System - GIS (SIG – Sistema de Informação Geográfica). Empresas começam a comercializar SIGs, alto custo e para computadores de grande porte
- **Anos 80**: computadores pessoais e estações de trabalho popularizam acesso aos SIG's. Criação de centros específicos como NCGIA marcam o estabelecimento do Geoprocessamento
- **Anos 90**: interesse em software livre de licença chega, aumentando uso dos SIG's
- **Anos 2000**: ênfase em acesso e publicação de dados no ambiente da Internet

Dados Espaciais

- Dados que contém uma **localização** espacial em algum **sistema de referência**. Contém informação sobre a localização, forma e as relações topológicas entre feições de interesse.

- Sistemas de Referência
  - São basicamente as diferentes perspectivas de um observador quanto a descrição de medidas (ex. posição) . Sistemas de coordenadas são as diferentes formas de descrever medidas sob essas perspectivas

- Dados Geográficos ou Geoespaciais
  - São aqueles onde o sistema de referência é a superfície da terra

Dados espaciais ou geográficos

Caracterizam-se por: **localização geográfica** (onde o fenômeno ocorre) e **atributos** (descrição do fenômeno)

Foto mostra massa de ar frio cobrindo o Sudeste brasileiro. Foto: Nasa

Paradigma dos 4 universos

- Permite traduzir o mundo real para o ambiente computacional :
  - Mundo Real: fenômenos a serem representados (cadastro urbano, vegetação, solos)
  - Universo Conceitual: distinção entre classes formais de fenômenos contínuos e objetos individualizáveis (campos e objetos)
  - Universo de Representação: diferentes representações geométricas (matrizes e vetores)
  - Universo de implementação: estruturas de dados e linguagens de programação (*R-tree* e *Quad-tree*)

Universo do Mundo Real

- Geoprocessamento manipula dados de naturezas e fontes distintas
- São organizados na forma de **camadas** (ou mapas). Um mapa é um modelo simplificado da realidade. Uma representação, em escala, de uma seleção de entidades abstratas relacionadas com a superfície da Terra
- Modelo que se interpõe entre a realidade e a base de dados de uma aplicação geográfica

Exemplo camadas

"*The Potteries*" ("As cidades ceramistas") conjunto de 6 cidades da Inglaterra com diversas indústrias relacionadas a cerâmica

A região se desenvolveu durante a revolução industrial inglesa

As comunidades locais produziam produtos de alta qualidade, mesmo em condições menos favoráveis

Baseado em Worboys & Duckhan 2004

Inventário de recursos

Dado o patrimônio cultural da área, a indústria local de turismo é significante

Camada com pontos de interesse cultural, equipamentos de recreação presentes na região e combiná-los com detalhes de infra-estrutura de transporte e hospedagem

Baseado em Worboys & Duckhan 2004

Análises de redes

Deseja-se encontrar uma rota, usando as rodovias principais, para visitar cada cidade (e o Museu da Cidade) uma vez, minimizando o tempo de viagem

É necessário uma rede que diga o tempo de viagem entre as cidades.

Gerado a partir do tempo médio de viagem nas vias principais mostradas na **camada de rede viária**

Baseado em Worboys & Duckhan 2004

Análises de terreno

Comunidades locais estão interessadas no impacto visual causado pela possível abertura de novas minas de carvão

Análise de visibilidade: medida o tamanho da população local dentro de um ângulo de visada (um mapa de todos os pontos visíveis a partir de uma dada localização)

Análise de terreno é geralmente baseada em uma **camada com dados de elevação** em localizações pontuais

Isolinhas de altimetria

Projeção perspectiva a partir de um ponto de visada. Regiões mais escuras marcam as áreas que não estarão visíveis.

Baseado em Worboys & Duckhan 2004

Sobreposição de camadas

Determinar o potencial de diferentes localizações para a extração de areia e cascalho

Apresentar e analisar camadas de:

- Geologia
- Estrutura urbana
- Lençol freático
- Zoneamento

Localização dos depósitos de areia e cascalho

Baseado em Worboys & Duckhan 2004

Sobreposição de camadas

Consulta: encontre as localizações que estão a 0.5 km de uma rodovia principal, em uma área não construída, com depósitos de areia/cascalho

0.5 km zona de influência das rodovias principais

Depósitos conhecidos de areia e cascalho

Áreas escuras indicam áreas não construídas

Interseção das 3 camadas resultando nas áreas que atendem a consulta

Baseado em Worboys & Duckhan 2004

Análises de localização

Alocação de hospitais na área das *Potteries*

- Construa a vizinhança dos hospitais existentes, baseado nas suas posições e os tempos de viagem até elas
- É então possível suportar melhor decisões sobre o fechamento, realocação ou a criação de um hospital

a. Isolinhas de mesma distância

b. Diagrama de Voronoi

Baseado em Worboys & Duckhan 2004

Tipos de mapas

- Mapas **Temáticos**: informações **qualitativas** sobre o espaço. Ex: Mapa de uso do solo ou Mapa de vegetação
- Mapas **Numéricos**: informações **quantitativas** sobre o espaço. Ex: Grade com valores de altimetria
- Mapas **Cadastrais**: informações sobre objetos discretos do mundo. Ex: Lotes urbanos com sua localização e seus atributos.
- Mapa de **Redes**: informações sobre objetos discretos que forma redes. Ex. Rede elétrica (postes e linhas de transmissão)
- **Imagens**: informações numéricas obtidas por sensores remotos. Ex: Fotografias aéreas, imagens de satélites e radares

IMAGENS

Aprisionar a luz

- Câmera de orifício (*pinhole camera*)
  - Século 4 AC – gregos
  - Século 10 DC – Ibn al-Haytham
- Câmera escura (*dark chamber*)
  - Giovanni Baptista Della Porta, 1558

(*G&W-07*)

(*Wiki-08*)

Fotografia

- Joseph Nicéphore Niépce (1765-1833)
- Heliografia (gravar com o sol)

View from the Window at Le Gras (1826)

Primeira imagem digital (?)

- Russel Kirsch - 1957
- Standards Eastern Automatic Computer (SEAC),
- National Bureau of Standards (NBS)
  (Hoje: National Institute of Standards and Technology – NIST).
- 176×176 píxeis e 5cm × 5cm

http://www.nist.gov/public_affairs/techbeat/tb2007_0524.htm

O início do processamento digital de imagens

- Anos 60
- Computadores suficientemente poderosos
- Programa espacial americano
- JPL – Jet Propulsion Laboratory
- Imagens da Lua pelo Ranger-7

Ranger-7 (jul,1969)

Satélites

- LANDSAT 80, 60 / 30 / 15 m
- SPOT-5 20 / 10 m , 5 / 2.5 m
- Ikonos 4 / 1 m
- Quickbird 2.4 / 0.6 m
- WorldView-1 0.5 m
- WorldView-2 1.84 / 0.46 m
- GeoEye-1 1.65 / 0.41 m (ex-Orb view)
- CBERS 2B / 20m, 80 m

landsat.gsfc.nasa.gov/

http://www.spot.com/home

Spot-5 2.5 m Trípoli

Aquisição de Imagens

Níveis de quantização típicos: 1, 8, 11 bits / píxel

Espectro Eletromagnético

-

Espectro Eletromagnético

(*Wikipedia-nov09*)

Curvas Espectrais

Curvas espectrais

Exemplos: ASTER Spectral Library
http://speclib.jpl.nasa.gov/

Escala dos Tons de Cinza

- LANDSAT-5 Washington-DC band 4

Infravermelho próximo

0.76-0.90 μm

(8 bits/píxel)

(G&W-08)

Colorida (True Color)

- LANDSAT-5 Washington-DC (R,G,B) => (3,2,1)

R Vermelho

G Verde

B Azul

Colorida (Falsa Cor)

- LANDSAT-5 Washington-DC (R,G,B) =>(5,4,3)

R Infravermelho médio

G Infravermelho próximo

B Vermelho

Colorida (Falsa Cor)

- LANDSAT-5 Washington-DC (R,G,B) => (4,3,2)

R Infravermelho próximo

G Vermelho

B Verde

Multispectral

- Imagem LANDSAT-TM

Multiespectral

- Bandas LANDSAT-TM

Band No.	Name	Wavelength (μm)	Characteristics and Uses
1	Visible blue	0.45–0.52	Maximum water penetration
2	Visible green	0.52–0.60	Good for measuring plant vigor
3	Visible red	0.63–0.69	Vegetation discrimination
4	Near infrared	0.76–0.90	Biomass and shoreline mapping
5	Middle infrared	1.55–1.75	Moisture content of soil and vegetation
6	Thermal infrared	10.4–12.5	Soil moisture; thermal mapping
7	Middle infrared	2.08–2.35	Mineral mapping

Temática (pseudo-cor)

- Imagem em pseudo-cor (mapeamento de tom de cinza para cor)

Quickbird pan+multiespectral
(0.6/2.4m)

Imagem temática
(classificação maxver)

4 dimensões de resolução

- Espacial
- Espectral
- Radiométrica
- Temporal

CBERS-2B

Imagem da Câmera CCD de alta resolução de Manaus

Imagem do WFI (Represa de Itaipu)

A cabeça óptica do WFI

Características da Câmera Imageadora de Alta Resolução CCD	
Bandas espectrais	0,51 - 0,73 µm (pan) 0,45 - 0,52 µm (azul) 0,52 - 0,59 µm (verde) 0,63 - 0,69 µm (vermelho) 0,77 - 0,89 µm (infravermelho próximo)
Campo de Visada	8,3º
Resolução espacial	20 x 20 m
Largura da faixa imageada	113 km
Capacidade de apontamento do espelho	±32º
Resolução temporal	26 dias com visada vertical (3 dias com visada lateral)
Frequência da portadora de RF	8103 MHz e 8321 MHz
Taxa de dados da imagem	2 x 53 Mbit/s
Potência Efetiva Isotrópica Irradiada	43 dBm

Características do Imageador de Amplo Campo de Visada WFI	
Bandas espectrais	0,63 - 0,69 µm (vermelho) 0,77 - 0,89 µm (infra-vermelho)
Campo de Visada	60º
Resolução espacial	260 x 260 m
Largura da faixa imageada	890 km
Resolução temporal	5 dias
Frequência da portadora de RF	8203,35 MHz
Taxa de dados da imagem	1,1 Mbit/s
Potência Efetiva Isotrópica Irradiada	31,8 dBm

Mapas temáticos

- Distribuição espacial qualitativa da grandeza ou atributo em estudo
- Os valores da grandeza podem ser:
  - nominal: lista de valores. Ex. possíveis classes de vegetação em um mapa de vegetação {floresta, cerrado, desmatamento}
  - ordinal: escala de medida. Ex. fatias de declividade {0-5%, 5-10%, 10-15%, 15-30%, >30%}

Mapas Numéricos

- Distribuição espacial quantitativa da grandeza em estudo
- Os valores da grandeza podem ser:
  - intervalo dentro de uma referência arbitrária. Ex. Altimetria, batimetria, temperatura em graus Celsius
  - razão: referência natural. Ex. Peso
- Localização espacial pode ser representada por grades regulares ou triangulares

Mapas cadastrais

- Contém informações sobre **objetos discretos** do mundo
- Exemplos: cadastros de países, de lotes, de propriedades rurais

País	PIB (US$bn)	Pop (milhões)
Brasil	350	159
Argentina	295	34
Chile	45	14

Mapas de rede

- Contêm objetos cuja referência geográfica está associada a nós ou arcos que formam uma **topologia de rede**

Paradigma de 4 universos

- Permite traduzir o mundo real para o ambiente computacional :
  - Mundo Real: fenômenos a serem representados (cadastro urbano, vegetação, solos)
  - Universo Conceitual: distinção entre classes formais de fenômenos contínuos e objetos individualizáveis (campos e objetos)
  - Universo de Representação: diferentes representações geométricas (matrizes e vetores)
  - Universo de implementação: estruturas de dados e linguagens de programação (*R-tree* e *Quad-tree*)

Universo conceitual: campo ou geo-campo

- <u>Geo-Campo</u>: $f = [R, A, \lambda]$, onde: $R \subset \Re^2$, $A = \{ A_1, A_2, ..., A_n\}$ é um conjunto atributos e λ: $R \rightarrow A$ mapeia pontos de R para valores em A
- Representa a distribuição espacial de uma variável que possui valores em todos os pontos pertencentes a R, num dado tempo *t*
- Se o contra-domínio A de *f* é um conjunto enumerável temos um dado temático. Ex. Mapa de cobertura vegetal
- Se o contra-domínio A de *f* é um conjunto de valores contínuos ($-\infty$ a $+\infty$) temos um dado numérico. Ex. mapa de aeromagnetometria
- Imagens são uma especialização de dados numérico

Definições auxiliares

- Região Geográfica (R) - uma superfície qualquer pertencente ao espaço geográfico, que pode ser representada num plano vetorial ou reticulado, dependente de uma projeção cartográfica
- Plano de informação - suporte para a representação geográfica de diferentes tipos de dados geográficos
  - É o lugar geométrico de um conjunto de dados geográficos - um geo-campo ou um mapa de geo-objetos
- Banco de Dados Geográficos - composto por conjuntos de planos de informação, um conjunto de geo-objetos e um conjunto de objetos não-espaciais

Universo conceitual: objeto ou geo-objetos

- Dadas as regiões geográficas $R_1, R_2, .. R_n$ ($R_i \subset \Re^2$) e o conjunto de atributos convencionais $\{A_1, ..., A_n\}$
- Um *Geo-Objeto* *go* = $[r_1, r_2, ..., r_n, a_1, a_2, ..., a_n]$, onde $r_i \subseteq R_i$ é uma parte de uma região geográfica R_i e $a_i \in A_i$ é um valor particular do atributo A_i
- É um elemento único que possui atributos não-espaciais e está associado a múltiplas localizações geográficas

País	PIB (US$bn)	Pop (milhões)
Brasil	350	159
Argentina	295	34
Chile	45	14

Geo-objetos

- Um geo-objeto pode possuir múltiplas representações dependendo da projeção cartográfica, escala ou instante de tempo

id	nome
	Amazonas
	Xingu

id	nome
1	Yanomami
2	Waimiri
3	Kayapó

Universo conceitual: objetos não-espaciais

- Um objeto não-espacial é um objeto que não possui localizações espaciais associadas
- Informações não georeferenciada agregada a uma aplicação geográfica

Resumo do universo conceitual

- Campos ou geo-campos (variáveis geográficas continuas)
  - Temático
  - Numérico
  - Imagem

- Objetos ou geo-objetos (variáveis geográficas discretas)
  - Cadastral
  - Redes

Mapa Lotes

Lotes

Geoid	Dono	Ender	Cadastro
22	Guimaraes	P. Chuí, 768	250186

Paradigma dos 4 universos

- Permite traduzir o mundo real para o ambiente computacional :
  - Mundo Real: fenômenos a serem representados (cadastro urbano, vegetação, solos)
  - Universo Conceitual: distinção entre classes formais de fenômenos contínuos e objetos individualizáveis (campos e objetos)
  - Universo de Representação: diferentes representações geométricas (matrizes e vetores)
  - Universo de implementação: estruturas de dados e linguagens de programação (*R-tree* e *Quad-tree*)

Universo de Representação

- Define as possíveis formas de representar no mundo digital os modelos do universo conceitual. Podem ser de dois tipos:
  - Vetoriais
  - Matriciais

(a) matricial (b) vetorial

Representação vetorial

- A localização e a aparência gráfica dos objetos são representadas por um ou mais pares de coordenadas que formam elementos gráficos

- Elementos da representação vetorial podem ser compostos

Representação vetorial

- Componentes de uma representação vetorial: ponto, linha, região
  - Ex: Região M={1,2,3,4,5}, formada pelas linhas 1, 2, 3, 4 e 5

- **Topologia**: descreve relações espaciais entre objetos, que são invariantes a rotação e translação
  - Ex. Toulose fica **Dentro** da região M

Representação vetorial

- Topologia Arco-Nó
  - Representa uma rede linear conectada
  - Nó: representa a intersecção entre linhas, são os pontos iniciais e finais
  - Linhas são conectadas

- Topologia Arco-Nó-Polígono
  - Representa elementos gráficos do tipo área ou partições do espaço

Representação matricial

- Divide a área sendo representada em uma matriz de células. A cada célula atribui-se um valor que o relaciona com o fênomeno sendo representado

10	2	1	5
6	3	4	10
3	10	94	3
11	2	7	0

10.5	2.3	1.9	5.4
6.1	3.7	4.6	10.1
3.1	10.0	94	3.4
11.6	2.3	7.9	0.5

linhas
colunas
tamanho das células
resolução dos valores

Comparação entre representações

- Vetorial
  - preserva relacionamentos topológicos
  - associa atributos a elementos gráficos
  - melhor exatidão e eficiência de armazenamento
- Matricial
  - fenômenos variantes no espaço
  - adequado para simulação e modelagem
  - processamento mais rápido e simples
  - maior gasto em armazenamento

MINISTÉRIO DA CIÊNCIA E TECNOLOGIA
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

Sistemas de Referência Espacial

Lubia Vinhas

Universo de Representação

- Vetorial

- Matricial

10	2	1	5
6	3	4	10
3	10	94	3
11	2	7	0

Representação

Dado matricial

Como esses dados são tratados digitalmente?
Onde está representado:
o quê e onde?

Dado vetorial

Dados geográficos digitais

Dado matricial

0	**0**	**0**
10	10	10
30	30	30

altimetria, radiância, etc...

O quê?

Dado vetorial

rio, cidade, unidade da paisagem, etc.

ID	Atr1	Atr2
o1	10.4	xxxx
o2	21.7	yyyy
o3	23.8	zzzz

Localização

Três pessoas foram ao campo, com diferentes equipamentos GPS, para localizar 2 pontos de interesse e voltaram com as seguintes medições:

	PESSOA 1		PESSOA 2		PESSOA 3	
	X	Y	X	Y	X	Y
P2	-42.3591666667	-23.3194444444	770061.694961	7418652.21437	1189337.72907	-2627767.87227
P3	-43.0166666667	-29.5347222222	692193.800396	6731129.17863	1063065.76762	-3318803.38603

- Qual delas fez as medições certas?
- Quais desses pontos caem dentro do perímetro de uma dada fazenda?
- Qual a cena CBERS-2B que recobre essa área?

Dados geográficos digitais

Dado matricial

Onde?

Dado vetorial

Mas em que sistema de referência estão essas coordenadas?

MINISTÉRIO DA CIÊNCIA E TECNOLOGIA
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

Sistemas de Referência Espaciais

Natureza dos dados espaciais

- Dados espaciais caracterizam-se especificamente pelo atributo da localização geográfica estabelecida quando:
  - possível descrevê-lo em relação a outro objeto cuja posição seja conhecida
  - possível descrevê-lo em um certo sistema de coordenadas

Minha casa

Long: 45° 53‘ 24.00“ O

Lat: 23° 11‘ 74.01“ S

Moro abaixo e a esquerda da Torre Eiffel

Torre Eiffel

Long: 2° 17‘ 54.01“ L

Lat: 48° 53‘ 33.24“ N

Sistema de coordenadas geográficas

- É o sistema de coordenadas mais antigo. Nele, cada ponto da superfície terrestre é localizado na interseção de um meridiano com um paralelo, definidos sobre uma superfície de referência

Mas qual é essa superfície de referência?

É uma esfera?

É uma elipse?

Quais suas dimensões?

Qual a forma da Terra?

Esse sistema é absoluto?

Conceitos de Geodésia

A Geodésia é a ciência que se encarrega da determinação da forma e das dimensões da Terra

Antigamente acreditava-se que a Terra era uma esfera.

Evolução da Física e Gravimetria chegou-se a conclusão de que a terra era achatada, ou um elipsóide (achatamento definido por gravimetria)

Século XIX – Legendre e Gauss provaram que estava havendo um erro quanto a forma da Terra. Concluíram que a Terra não era uma elipsóide mudando novamente o conceito da figura da Terra, mais tarde este novo conceito foi chamado de Geóide

Conceitos de Geodésia

Geodésia trata da determinação das dimensões e da forma da Terra

Geóide é aceito como figura matemática da Terra

Superfície equipotencial do campo gravitacional da Terra que mais se aproxima do nível médio dos mares

Conceitos de Geodésia

Na prática o geóide não é conhecido globalmente: faltam estações gravimétricas em todo planeta e equações complexas

Surge uma superfície de referência mais adequada à Terra real, ou seja, tratável matematicamente: Elipsóide de Referência ou Terra Cartográfica

Um elipsóide é caracterizado por seus semi-eixos maior (raio Equatorial) e menor (achatamento dos polos)

Datum planimétrico

É composto por uma superfície de referência posicionada em relação à Terra real;

O procedimento prático de estabelecer uma referência geodésica começa com a seleção arbitrária de um ponto conveniente para o Datum e de sua representação na superfície de um elipsóide escolhido

Datum planimétrico

Seleciona-se o elipsóide de referência mais adequado à região

Posiciona-se o elipsóide em relação à Terra real – preservando o paralelismo entre o eixo de rotação da Terra e do elipsóide

Escolhe-se um ponto central (origem) no país ou região e faz-se a anulação do desvio da vertical

Datum planimétrico

Pode ser global (o centro do elipsóide coincide com o centro de massa da Terra) ou local (o centro do elipsóide está deslocado do centro da Terra)

Mensagem importante: as Coordenadas Geográficas, dependem de um Datum planimétrico, pois ele define a referência para os meridianos e paralelos.

Sistema de coordenadas geográficas

Latitude geodésica ou geográfica

ângulo entre a normal à superfície de referência (elipsóide ou esfera), no ponto em questão, e o plano do equador. Varia de 0º a 90º (norte ou sul)

Longitude geodésica ou geográfica

ângulo entre o meridiano que passa pelo ponto e o meridiano origem (Greenwich, por convenção). Varia 0º a 180º (leste e oeste)

φ – latitude geodésica (graus)

λ – longitude geodésica (graus)

h – altitude elipsoidal (metros)

Datum usados no Brasil

- Legalmente:
  - **SAD69** - South American Datum 1969
  - **SIRGAS2000** - Sistema de Referência Geocêntrico para as Américas
- Também é comum encontrar cartas topográficas que referem-se à **Córrego Alegre**, o antigo Datum brasileiro
- 25 de fevereiro de 2005: SIRGAS2000 foi oficialmente adotado como o novo sistema de referência geodésico para o Sistema Geodésico Brasileiro (SGB) e para o Sistema Cartográfico Nacional (SCN)
- Foi também definido um período de transição, não superior a 10 anos, onde o sistema novo (SIRGAS2000) e os antigos (SAD 69, Córrego Alegre) poderão ser utilizados concomitantemente.
- Depois de passado o período de transição, o SIRGAS2000 será o único sistema geodésico de referência legalizado no país.

Datum usados no Brasil

- Diferenças entre o SAD69 e o SIRGAS2000:
  - SAD69 é um sistema de referência topocêntrico que tem como referência um ponto na superfície da Terra
  - SIRGASS2000 é geocêntrico que tem como referência um ponto no centro de massa da Terra
  - SIRGASS2000 atende a uma necessidade de compatibilização com o sistema de posicionamento GPS, que também é geocêntrico

Outros Datum

- Locais
  - SAD69, Córrego Alegre, NAD27, Indian...
- Globais
  - WGS84, SIRGAS, NAD83...

- WGS84 e SIRGAS200 são praticamente idênticos, pois utilizam o mesmo elipsóide de referência (GRS80), com alguns centímetros de diferença no valor do achatamento.

Erros de Posicionamento

- Dados em coordenadas geodésicas, em diferentes Datum, podem gerar erros de posicionamento
- Por exemplo, um mapeamento realizado em SAD69 e outro em SIRGAS2000 não podem ser mostrados no mesmo mapa sem que seja feito algum tratamento

Fonte: http://www.pign.org/PIGN3/Portugues/cadastral.htm

(relatório do Projeto demonstração 2)

Erros de Posicionamento

- De SIRGAS200 para SAD69 : ~65 metros no território brasileiro
- De SIRGAS2000 para WGS84: nenhum
- De Córrego Alegre para SAD69: <= 60 metros

SOLUÇÃO:

- lembre que a variação das coordenadas geográficas afeta a exatidão de sua base de dados
- use um SIG que saiba levar em conta essa variação de coordenadas
- saiba o que está medindo com um receptor GPS
- tenha cuidado com dados compartilhados (importação e exportação)

No mapa a Terra é plana

Processo de criação de um mapa

Projeções Cartográficas

- Impossível representar uma superfície curva num plano sem deformação, por isso apareceu o conceito de Superfície de Projeção

- Superfície de Projeção é uma superfície desenvolvível no plano, capaz de representar um sistema plano de meridianos e paralelos sobre o qual pode ser desenhada uma representação cartográfica (carta, mapa, planta)

Projeções Cartográficas

- Uma projeção cartográfica determina a correspondência matemática biunívoca entre os pontos da esfera (ou elipsóide) e sua transformação num plano

- Sistemas de projeção resolvem as equações:

  (x e y – coordenadas planas, ϕ,λ −coordenadas geográficas)

$$x = f_1(\phi,\lambda) \qquad y = f_2(\phi,\lambda)$$

$$\lambda = g_1(x,y) \qquad \phi = g_2(x,y)$$

Projeções cartográficas

- Impossível representar uma superfície curva num plano sem deformação. Por isso, existem diferentes classes de projeção, que causam diferentes distorções e por isso tem diferentes aplicações
- Uma mesma área sob diferentes projeções geram mapas diferentes

Cilíndrica Plana Cônica

Classes de projeção

Equidistantes: preservam distâncias

Equivalentes: preservam áreas

Conformes: preservam ângulos

Quanto as propriedades

Quanto a superfície de projeção

Projeções Cartográficas

- Superfície ou figura de referência
  - esfera, elipsóide
- Superfície de projeção
  - plano, cone, cilindro, poliedro
- Posição da superfície de projeção
  - normal ou equatorial, oblíqua, transversa
- Método de construção
  - projetivo, analítico

Projeções Cartográficas

- Projeções planas ou azimutais
  - plano tangente ou secante
    - estereográfica polar, azimutal de Lambert
- Projeções cônicas
  - cone tangente ou secante
    - cônica de Lambert, cônica de Albers
- Projeções cilíndricas
  - cilindro tangente ou secante
    - UTM, Mercator, Miller

Projeções Cartográficas

- Projeções conformes ou isogonais
  - preservam ângulos
    - UTM, Mercator, cônica conforme de Lambert
- Projeções equivalentes ou isométricas
  - preservam áreas
    - cônica equivalente de Albers
- Projeções equidistantes
  - representam distâncias em verdadeira grandeza ao longo de certas direções
    - cilíndrica equidistante

Projeções Cartográficas

- Parâmetros das projeções
  - figura de referência (elipsóide ou esfera)
    - datum planimétrico
  - paralelo padrão (latitude reduzida)
    - deformações nulas, escala verdadeira … verdadeira grandeza
    - Um ou dois paralelos se a superfície é tangente ou secante
  - longitude de origem (meridiano central)
    - posição do eixo Y das coordenadas planas
    - para a UTM é o meridiano central de um fuso
  - latitude origem
    - posição do eixo X das coordenadas planas
    - Equador para a maioria das projeções

Projeções Cartográficas

- Sistema UTM – Universal Transversa de Mercator

Projeções Cartográficas

- O sistema UTM é Universal, pois é aplicável em toda a extensão do globo terrestre

Zonas do sistema UTM
(Fonte: Serviço Geodésico Interamericano, s/d)

Projeções Cartográficas Especificações UTM

- Adota 60 cilindros de eixo transverso, de maneira que cada um cobre a longitude de 6º (3º para cada lado do meridiano central)

- Em latitude os fusos são limitados ao paralelo 80º N e S pois, acima deste valor as deformações se acentuam muito

- Para evitar coordenadas negativas a partir da origem das coordenadas (cruzamento Equador com meridiano central) será acrescida em cada fuso das constantes 10.000.000 metros no eixo das ordenadas (NS) e de + 500.000 metros no eixo das abcissas (EW)

Especificações UTM

Universal Transversa de Mercator

- Adota 60 cilindros de eixo transverso, de maneira que cada um cobre a longitude de 6º (3º para cada lado do meridiano central)
- Em latitude os fusos são limitados ao paralelo 80º N e S pois, acima deste valor as deformações se acentuam muito

Universal Transversa de Mercator

- Adota 60 cilindros de eixo transverso, de maneira que cada um cobre a longitude de 6º (3º para cada lado do meridiano central)
- Em latitude os fusos são limitados ao paralelo 80º N e S pois, acima deste valor as deformações se acentuam muito

Universal Transversa de Mercator

- Para evitar coordenadas negativas a partir da origem das coordenadas (cruzamento Equador com meridiano central) será acrescida em cada fuso das constantes 10.000.000 metros no eixo das ordenadas (NS) e de + 500.000 metros no eixo das abscissas (EW)

Projeções Cartográficas

Índice de Nomenclatura das Folhas

Nomenclaturas Folhas

Nomenclaturas Folhas

1: 10.000 (SF 23 - Z – D – VI – 3- SO- F)

1: 25.000 (SF 23 - Z – D – VI – 3- SO)

1: 50.000 (SF 23-Z – D – VI - 3)

Carta Topográfica

CARTA DO BRASIL ESC. 1:50 000

LUTÉCIA

MI-2361/3
FOLHA SF-22-Z-A-III-3

PROJEÇÃO UNIVERSAL TRANSVERSA DE MERCATOR

ARTICULAÇÃO DA FOLHA

QUATÁ	HERCULÂNDIA	POMPÉIA
PARAGUAÇU PAULISTA	LUTÉCIA	EXAPORÃ
MARACAÍ	ASSIS	CAMPOS NOVOS PAULISTA

LOCALIZAÇÃO DA FOLHA NO ESTADO

ESCALA 1: 50 000

1 000 m 0 1 000 2 000 3 000 m

Eqüidistância das curvas de nível: 20 metros

Origem da quilometragem: Equador e Meridiano 51° W. Gr., acrescidas as constantes 10 000 km e 500 km respectivamente

Datum vertical: marégrafo Imbituba, SC
Datum horizontal: Córrego Alegre, MG

Levantamento estereofotogramétrico topográfico regular Aerofotografias - 1965 ; apoio suplementar e reambulação executados em 1973 pelo Departamento de Geodésia e Topografia; restituição, aerotriangulação e preparo para a impressão realizados pelo Departamento de Cartografia

Esta folha foi preparada e impressa em decorrência do Convênio entre o IBGE e o Departamento de Águas e Energia Elétrica do Estado de São Paulo

SUPERINTENDÊNCIA DE CARTOGRAFIA

PRIMEIRA EDIÇÃO — 1974

DIREITOS DE REPRODUÇÃO RESERVADOS

Impressa no Serviço Gráfico do IBGE

A SUPERINTENDÊNCIA DE CARTOGRAFIA agradece a gentileza da comunicação de falhas ou omissões verificadas nesta Folha

Projeções Cartográficas

- Principais projeções no Brasil
  - UTM (Universal Transverse Mercator)
    - cartas topográficas
  - Mercator
    - cartas náuticas
  - Cônica conforme de Lambert
    - cartas ao milionésimo
    - cartas aeronáuticas
  - Policônica
    - mapas temáticos
    - mapas políticos

Projeções Cartográficas

- Outras projeções importantes
  - Cilíndrica equidistante
    - apresentação de dados em SIG
    - mapas mundi
  - Estereográfica polar
    - substitui a UTM nas regiões polares
  - Cônica conforme bipolar oblíqua
    - mapa político das Américas
  - Cônica equivalente de Albers
    - cálculo de área em SIG

Conceitos de Geodésia

- Datum altimétrico ou vertical
  - superfície de referência para a contagem das altitudes (geóide)
  - rede de marégrafos faz medições contínuas para a determinação do nível médio dos mares
  - adota-se um dos marégrafos como ponto de referência do datum vertical
  - no Brasil usa-se o marégrafo de Imbituba, em Santa Catarina

Conceitos de Geodésia

- Relevo
  - Saber se dois ou mais pontos estão no mesmo nível (altitude) – subir ou descer
  - **Nível base** – origem padrão de medidas = nível médio dos mares
  - **Altitude** é a medida do desnível que existe entre qualquer ponto da superfície e o nível do mar.

  - Altitude é DIFERENTE de Altura

Conceitos de Geodésia

- Datum altimétrico ou vertical
  - superfície de referência para a contagem das altitudes (geóide).
  - rede de marégrafos faz medições contínuas para a determinação do nível médio dos mares
  - adota-se um dos marégrafos como ponto de referência do datum vertical
  - no Brasil usa-se o marégrafo de Imbituba, em Santa Catarina

Conceitos de Geodésia

- Altitudes
  - Positivas – elevações, ou seja estão acima do nível base
  - Negativas – depressões, ou seja, estão abaixo do nível base
  - Mapas – representação em **curvas de nível**
  - **Curva de nível são linhas que unem pontos de mesma altitude**

Escala

- É a relação entre a medida de um objeto ou lugar geográfico representado no papel e sua verdadeira dimensão.

- Normalmente é expressa das seguintes formas:
  - Fração representativa ou numérica
  - Gráfica ou escala de barras

Escala

- A escala de um mapa deve representar os detalhes
  - Naturais (rios, mares, montanhas)
  - Artificiais (estradas, pontes, edificações)

- Problemas
  - Necessidade de reduzir as proporções dos acidentes a representar.
  - Determinados acidentes, dependendo da escala, não permitem redução acentuada pois se tornam imperceptíveis.

A solução é se utilizar símbolos cartográficos.

Escala numérica

$$E = d / D$$

d: distância medida na carta

D: distância real

As escalas numéricas mais comuns são da forma:

$E = 1 / 10x$ ou $E = 1:10x$

Escala gráfica

- É a representação gráfica de várias distâncias do terreno sobre uma linha reta graduada
- É constituida de um segmento à direita da referência zero, conhecida como escala primária
- Consiste também de um segmento à esquerda da origem denominada de Talão ou Escala de Fracionamento, que é dividido em sub-múltiplos da unidade escolhida graduada da direita para a esquerda

Escalas especiais

- As fotografias aéreas e grande parte das projeções cartográficas não possuem escalas constantes, elas são variáveis dependendo de uma série de fatores inerentes ao processo de elaboração da projeção.
- As fotografias aéreas, por serem uma projeção central, a escala é variável do centro da foto para a periferia, sendo tanto menor quanto mais próximo das bordas.
- Quando a escala for grande, não ocorrerão muitos problemas pois os erros serão desprezíveis, o que já não ocorrerá em escalas pequenas, podendo ser constante ao longo dos paralelos e variável ao longo dos meridianos, ou vice-versa. Depende do tipo de projeção e da sua estrutura projetiva.

Escalas especiais

- Na projeção de Mercator por exemplo, a escala é variável, constante ao longo dos paralelos e variável ao longo dos meridianos, variando com a latitude, quanto maior a latitude, maior a escala.

PROJEÇÃO DE MERCATOR

Escala em Diferentes Latitudes

1/50 000 000 no Equador - 1/9 132 500 na Latitude de 24

Escala

- Precisão gráfica
  - É a menor grandeza medida no terreno, capaz de ser representada em desenho na mencionada escala.
  - Menor comprimento: 0,2 mm

Seja E = 1 / M

Erro tolerável: 0,0002 metro X M

E = 1/20000 ----- 0.2mm = 4000 mm = 4 m

E = 1/10000 ----- 0,2mm = 2000 mm = 2 m

E = 1/40000 ----- 0,2mm = 8000 mm = 8 m

E = 1/100000 ---- 0,2mm = 20000 mm = 20 m

Escala

- Escolha de escala

Considerando uma região que se queira mapear e que possua muitos acidentes de 10m de extensão, a menor escala que se deve adotar será:

Erro tolerável = 0,0002 metro X M

M = Erro tolerável / 0,0002 metro

M = 10m / 0,0002m = 50.000 ou seja

E = 1:50.000

Escala

- As condicionantes básicas para a escolha de uma escala de representação são:
  - dimensões da área do terreno que será mapeado;
  - tamanho do papel que será traçado o mapa;
  - a orientação da área;
  - erro gráfico;
  - precisão do levantamento e/ou das informações a serem plotadas no mapa.

MINISTÉRIO DA CIÊNCIA E TECNOLOGIA
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

Resumo

Cartografia para geoprocessamento

- Cartografia: preocupa-se em apresentar um modelo de representação de dados para os processos que ocorrem no espaço geográfico
- Geoprocessamento representa a área do conhecimento que utiliza técnicas matemáticas e computacionais, fornecidas pelos Sistemas de Informação Geográfica (SIG), para tratar os processos que ocorrem no espaço geográfico

(fonte: Maguire, Goodchild, Rhind, 1991)

Dados geográficos

Duas **componentes**: o quê e onde

O quê → semântico: altimetria, município, reflectância, estação de coleta, etc.

Onde → localização em um sistema que representa a superfície terrestre, ou seja, Sistema de Referência Espacial

Representação: estrutura de dados usada para armazenar o dado geográfico

Matricial: matriz regular de valores

Vetorial: primitivas geométricas ponto, linha e polígono para representar a localização, associada a um conjunto de valores

Geoprocessamento

Geoprocessamento: disciplina que trata da manipulação de dados geográficos.

SIG: sistema computacional usado para materializar as técnicas de geoprocessamento

Como descrever um SRS?

1. Qual seu nome?
2. Qual é o Datum?
3. É geográfico?
    a. Qual a unidade angular?
    b. Qual o meridiano padrão?
    c. Qual o paralelo de latitude 0?
4. É projetado?
    1. Qual a unidade linear?
    2. Qual a projeção?
    3. Quais os parâmetros da projeção?

Como descrever um SRS?

```
GEOGCS["WGS 84",
   DATUM["WGS_1984",
      SPHEROID["WGS 84",6378137,298.257223563]],
   PRIMEM["Greenwich",0],
   UNIT["degree",0.01745329251994328]]
```

```
PROJCS["WGS 84 / UTM zone 23S",
   GEOGCS["WGS 84",
      DATUM["WGS_1984",
         SPHEROID["WGS 84",6378137,298.257223563],
      PRIMEM["Greenwich",0],
      UNIT["degree",0.01745329251994328]],
   UNIT["metre",1],
   PROJECTION["Transverse_Mercator"],
   PARAMETER["latitude_of_origin",0],
   PARAMETER["central_meridian",-45],
   PARAMETER["scale_factor",0.9996],
   PARAMETER["false_easting",500000],
   PARAMETER["false_northing",10000000],
   AXIS["Easting",EAST],
   AXIS["Northing",NORTH]]
```

WKT – Well Known Text for SRS
OGC – Open Geospatial Consortium

Como descrever um SRS?

```
GEOGCS["WGS 84",
  DATUM["WGS_1984",
    SPHEROID["WGS 84",6378137,298.257223563,
      AUTHORITY["EPSG","7030"]],
    AUTHORITY["EPSG","6326"]],
  PRIMEM["Greenwich",0,
    AUTHORITY["EPSG","8901"]],
  UNIT["degree",0.01745329251994328,
    AUTHORITY["EPSG","9122"]],
  AUTHORITY["EPSG","4326"]]
```

```
PROJCS["WGS 84 / UTM zone 23S",
  GEOGCS["WGS 84",
    DATUM["WGS_1984",
      SPHEROID["WGS 84",6378137,298.257223563,
        AUTHORITY["EPSG","7030"]],
      AUTHORITY["EPSG","6326"]],
    PRIMEM["Greenwich",0,
      AUTHORITY["EPSG","8901"]],
    UNIT["degree",0.01745329251994328,
      AUTHORITY["EPSG","9122"]],
    AUTHORITY["EPSG","4326"]],
  UNIT["metre",1,
    AUTHORITY["EPSG","9001"]],
  PROJECTION["Transverse_Mercator"],
  PARAMETER["latitude_of_origin",0],
  PARAMETER["central_meridian",-45],
  PARAMETER["scale_factor",0.9996],
  PARAMETER["false_easting",500000],
  PARAMETER["false_northing",10000000],
  AUTHORITY["EPSG","32723"],
  AXIS["Easting",EAST],
  AXIS["Northing",NORTH]]
```

SRS ID: identificador dado por uma autoridade
EPSG – European Petroleum Survey Group

Voltando ao exemplo

Tres pessoas foram ao campo, com diferentes equipamentos, para localizar 2 pontos de interesse e voltaram com as seguintes medições:

	PESSOA 1		PESSOA 2		PESSOA 3	
	X	Y	X	Y	X	Y
P2	-42.3591666667	-23.3194444444	770061.694961	7418652.21437	1189337.72907	-2627767.87227
P3	-43.0166666667	-29.5347222222	692193.800396	6731129.17863	1063065.76762	-3318803.38603

Qual delas fez as medições certas?

Resposta:

não posso avaliar sem saber em qual Sistema de Referência Espacial estão esses números.

Voltando ao exemplo

	PESSOA 1		PESSOA 2		PESSOA 3	
	X	**Y**	**X**	**Y**	**X**	**Y**
P2	-42.3591666667	-23.3194444444	770061.694961	7418652.21437	1189337.72907	-2627767.87227
P3	-43.0166666667	-29.5347222222	692193.800396	6731129.17863	1063065.76762	-3318803.38603
	SRID = 4623		SRID = 32723		SRID = 29101	

Qual delas fez as medições certas?

Neste caso, todas estão corretas. Todas se referem a mesma localização, mas medidas em sistemas de referência espacial diferentes.

Conversão de coordenadas

	PESSOA 1		PESSOA 2		PESSOA 3	
	X	Y	X	Y	X	Y
P2	-42.3591666667	-23.3194444444	770061.694961	7418652.21437	1189337.72907	-2627767.87227
P3	-43.0166666667	-29.5347222222	692193.800396	6731129.17863	1063065.76762	-3318803.38603
	SRID = 4623		SRID = 32723		SRID = 29101	

Conversão de coordenadas

Conversão de coordenadas

PROJ.4

PROJ.4 - Cartographic Projections Library

SRS EM DADOS VETORIAIS

Open Geospatial Consortium

- Consórcio entre companhias, universidade e agências governamentais
- Objetivo: promover o desenvolvimento de tecnologias que facilitem a interoperabilidade entre sistemas envolvendo informação geo-espacial
- Os produtos do trabalho do OGC são apresentados sob forma de especificações de interfaces e padrões de intercâmbio

Open Geospatial Consortium

- Algumas especificações OGC:
  - GML (Geography Markup Language): intercâmbio de dados
  - OWS (OGC Web Service): especificações de serviços WEB
    - WFS: Web Feature Service
    - WMS: Web Map Server
  - SFSQL (Simple Feature Specification For SQL): especificações sobre o armazenamento e recuperação de dados espaciais em SGBD's OR

OGC Simple Feature

WKT – Well Known Text for geometry

- Gramática que define como representar textualmente todos as diferentes instâncias de geometrias. Exemplos:

Geometry Type	Text Literal Representation	Comment
Point	Point (10 10)	a Point
LineString	LineString (10 10, 20 20, 30 40)	a LineString with 3 points
Polygon	Polygon ((10 10, 10 20, 20 20, 20 15, 10 10))	a Polygon with 1 exteriorRing and 0 interiorRings

WKT

Multipoint	MultiPoint ((10 10), (20 20))	a MultiPoint with 2 points
MultiLineString	MultiLineString ((10 10, 20 20), (15 15, 30 15))	a MultiLineString with 2 linestrings
MultiPolygon	MultiPolygon (((10 10, 10 20, 20 20, 20 15, 10 10)), ((60 60, 70 70, 80 60, 60 60)))	a MultiPolygon with 2 polygons
GeomCollection	GeometryCollection (POINT (10 10), POINT (30 30), LINESTRING (15 15, 20 20))	a GeometryCollection consisting of 2 Point values and a LineString value
PolyhedralSurface	PolyhedralSurface Z (((0 0 0, 0 0 1, 0 1 1, 0 1 0, 0 0 0)), ((0 0 0, 0 1 0, 1 1 0, 1 0 0, 0 0 0)), ((0 0 0, 1 0 0, 1 0 1, 0 0 1, 0 0 0)), ((1 1 0, 1 1 1, 1 0 1, 1 0 0, 1 1 0)), ((0 1 0, 0 1 1, 1 1 1, 1 1 0, 0 1 0)), ((0 0 1, 1 0 1, 1 1 1, 0 1 1. 0 0 1)))	A polyhedron cube, corner at the origin and opposite corner at (1, 1, 1).

WKT

Tin	Tin Z (((0 0 0, 0 0 1, 0 1 0, 0 0 0)), ((0 0 0, 0 1 0, 1 0 0, 0 0 0)), ((0 0 0, 1 0 0, 0 0 1, 0 0 0)), ((1 0 0, 0 1 0, 0 0 1, 1 0 0)),)	A tetrahedron (4 triangular faces), corner at the origin and each unit coordinate digit.
Point	Point Z (10 10 5)	a 3D Point
Point	Point ZM (10 10 5 40)	the same 3D Point with M value of 40
Point	Point M (10 10 40)	a 2D Point with M value of 40

WKB – Well Known Binary for geometry

- Versão binária para representar as geometrias. Pode ser usada na especificação do armazenamento

Key

1 WKB Polygon
2 ring 1
3 ring 2

Figure 25: Well-known Binary Representation for a geometric object
in NDR format (B = 1)
of type Polygon (T = 3)
with 2 LinearRings (NR = 2)
each LinearRing having 3 points (NP = 3)

Códigos para as geometrias

Type	Code
Geometry	0
Point	1
LineString	2
Polygon	3
MultiPoint	4
MultiLineString	5
MultiPolygon	6
GeometryCollection	7
CircularString	8
CompoundCurve	9
CurvePolygon	10
MultiCurve	11
MultiSurface	12
Curve	13
Surface	14
PolyhedralSurface	15
TIN	16

Type	Code
Geometry Z	1000
Point Z	1001
LineString Z	1002
Polygon Z	1003
MultiPoint Z	1004
MultiLineString Z	1005
MultiPolygon Z	1006
GeometryCollection Z	1007
CircularString Z	1008
CompoundCurve Z	1009
CurvePolygon Z	1010
MultiCurve Z	1011
MultiSurface Z	1012
Curve Z	1013
Surface Z	1014
PolyhedralSurface Z	1015
TIN Z	1016

Type	Code
Geometry M	2000
Point M	2001
LineString M	2002
Polygon M	2003
MultiPoint M	2004
MultiLineString M	2005
MultiPolygon M	2006
GeometryCollection M	2007
CircularString M	2008
CompoundCurve M	2009
CurvePolygon M	2010
MultiCurve M	2011
MultiSurface M	2012
Curve M	2013
Surface M	2014
PolyhedralSurface M	2015
TIN M	2016

Type	Code
Geometry ZM	3000
Point ZM	3001
LineString ZM	3002
Polygon ZM	3003
MultiPoint ZM	3004
MultiLineString ZM	3005
MultiPolygon ZM	3006
GeometryCollection ZM	3007
CircularString ZM	3008
CompoundCurve ZM	3009
CurvePolygon ZM	3010
MultiCurve ZM	3011
MultiSurface ZM	3012
Curve ZM	3013
Surface ZM	3014
PolyhedralSurface ZM	3015
TIN ZM	3016

OGC Well-Known Text for SRS

```
GEOGCS["WGS 84",
   DATUM["WGS_1984",
      SPHEROID["WGS 84",6378137,298.257223563]],
   PRIMEM["Greenwich",0],
   UNIT["degree",0.01745329251994328]]
```

```
PROJCS["WGS 84 / UTM zone 23S",
   GEOGCS["WGS 84",
      DATUM["WGS_1984",
         SPHEROID["WGS 84",6378137,298.257223563],
      PRIMEM["Greenwich",0],
      UNIT["degree",0.01745329251994328]],
   UNIT["metre",1],
   PROJECTION["Transverse_Mercator"],
   PARAMETER["latitude_of_origin",0],
   PARAMETER["central_meridian",-45],
   PARAMETER["scale_factor",0.9996],
   PARAMETER["false_easting",500000],
   PARAMETER["false_northing",10000000],
   AXIS["Easting",EAST],
   AXIS["Northing",NORTH]]
```

WKT – Well Known Text for SRS
OGC – Open Geospatial Consortium

SRS EM DADOS MATRICIAIS

Georreferenciamento

- Trata da aplicação de transformações geométricas para corrigir as distorções inerentes ao processo de aquisição da imagem
- Fontes de distorções geométricas (sensores MSS, TM, HRV, AVHRR, WFI)
  - rotação da Terra (skew)
  - distorções panorâmicas (compressão)
  - curvatura da Terra (compressão)
  - arrastamento da imagem durante uma varredura
  - variações de altitude, atitude e velocidade do satélite

Georeferenciamento

- Para que possam ser integradas a base geográfica as imagens devem passar por vários níveis de processamento para corrigir as distorções radiométricas e geométricas.. Ex:
  1. Com correção radiométrica
  2. Com correção de sistema (radiométrica + boresishts + atitude + efemérides)
  3. Georreferenciada (sistema + pontos de controle 2D)
  4. Ortorretificada (sistema + pontos de controle 2D + DTM)

Georeferenciamento

- Exemplo de como diferentes produtores de imagens chamam seus níveis de processamento

	Correção			
Satélite	Radiométrica	Sistema	Georrefenciada	Ortorretificada
SPOT	1A	2A	2B	3
Landsat	0	1R	1G	1P (precision)
CBERS	1	2	3	4

- Atualmente as imagens são distribuídas com correção > 2, porém a necessidade de integrar imagens antigas, ou devido a problemas nos parâmetros do satélite, muitas vezes os próprios usuários tem que fazer algumas correções geométricas.

Como representar o georeferenciamento?

$$x = x_0 + i*r_x$$
$$y = y_0 + j*r_y$$

$$x = x_0 + i*r_x + i*r_y$$
$$y = y_0 + j*r_y + j*r_x$$

* r_y deve ser considerada negativo

Como representar o georeferenciamento?

Brasilia.tif

+

Brasilia.tfw

30	→ R_X
0.0	→ Rotação em relação ao eixo X
0.0	→ Rotação em relação ao eixo Y
-30	→ $-1 * R_Y$
183557.00	→ X_0
8.258940.00	→ Y_0

Como representar o gereferenciamento?

brasilia.tif

- SRS=…
- Res X=…
- Res Y=…
- PCs = [….]

Geo TAGs

Imagem

OGC para representações matriciais?

- Não existe um padrão para representações matriciais nem em modo texto, nem em modo binário

- No nível de outras especificações (a serem vistas adiante) GeoTiff é considerado um formato padrão.

MINISTÉRIO DA CIÊNCIA E TECNOLOGIA
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

Sistemas de Referência Espacial

Lubia Vinhas

Universo de Representação

- Vetorial

- Matricial

10	2	1	5
6	3	4	10
3	10	94	3
11	2	7	0

Representação

Dado matricial

Como esses dados são tratados digitalmente?
Onde está representado:
o quê e onde?

Dado vetorial

Dados geográficos digitais

Dado matricial

0	**0**	**0**
10	10	10
30	30	30

altimetria, radiância, etc...

O quê?

Dado vetorial

rio, cidade, unidade da paisagem, etc.

ID	Atr1	Atr2
o1	10.4	xxxx
o2	21.7	yyyy
o3	23.8	zzzz

Localização

Três pessoas foram ao campo, com diferentes equipamentos GPS, para localizar 2 pontos de interesse e voltaram com as seguintes medições:

	PESSOA 1		PESSOA 2		PESSOA 3	
	X	Y	X	Y	X	Y
P2	-42.3591666667	-23.3194444444	770061.694961	7418652.21437	1189337.72907	-2627767.87227
P3	-43.0166666667	-29.5347222222	692193.800396	6731129.17863	1063065.76762	-3318803.38603

- Qual delas fez as medições certas?
- Quais desses pontos caem dentro do perímetro de uma dada fazenda?
- Qual a cena CBERS-2B que recobre essa área?

Dados geográficos digitais

Dado matricial

Onde?

Dado vetorial

Mas em que sistema de referência estão essas coordenadas?

MINISTÉRIO DA CIÊNCIA E TECNOLOGIA
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

Sistemas de Referência Espaciais

Natureza dos dados espaciais

- Dados espaciais caracterizam-se especificamente pelo atributo da localização geográfica estabelecida quando:
  - possível descrevê-lo em relação a outro objeto cuja posição seja conhecida
  - possível descrevê-lo em um certo sistema de coordenadas

Minha casa

Long: 45° 53‘ 24.00“ O

Lat: 23° 11‘ 74.01“ S

Moro abaixo e a esquerda da Torre Eiffel

Torre Eiffel

Long: 2° 17‘ 54.01“ L

Lat: 48° 53‘ 33.24“ N

Sistema de coordenadas geográficas

- É o sistema de coordenadas mais antigo. Nele, cada ponto da superfície terrestre é localizado na interseção de um meridiano com um paralelo, definidos sobre uma superfície de referência

Mas qual é essa superfície de referência?

É uma esfera?

É uma elipse?

Quais suas dimensões?

Qual a forma da Terra?

Esse sistema é absoluto?

Conceitos de Geodésia

A Geodésia é a ciência que se encarrega da determinação da forma e das dimensões da Terra

Antigamente acreditava-se que a Terra era uma esfera.

Evolução da Física e Gravimetria chegou-se a conclusão de que a terra era achatada, ou um elipsóide (achatamento definido por gravimetria)

Século XIX – Legendre e Gauss provaram que estava havendo um erro quanto a forma da Terra. Concluíram que a Terra não era uma elipsóide mudando novamente o conceito da figura da Terra, mais tarde este novo conceito foi chamado de Geóide

Conceitos de Geodésia

Geodésia trata da determinação das dimensões e da forma da Terra

Geóide é aceito como figura matemática da Terra

Superfície equipotencial do campo gravitacional da Terra que mais se aproxima do nível médio dos mares

Conceitos de Geodésia

Na prática o geóide não é conhecido globalmente: faltam estações gravimétricas em todo planeta e equações complexas

Surge uma superfície de referência mais adequada à Terra real, ou seja, tratável matematicamente: Elipsóide de Referência ou Terra Cartográfica

Um elipsóide é caracterizado por seus semi-eixos maior (raio Equatorial) e menor (achatamento dos polos)

Datum planimétrico

É composto por uma superfície de referência posicionada em relação à Terra real;

O procedimento prático de estabelecer uma referência geodésica começa com a seleção arbitrária de um ponto conveniente para o Datum e de sua representação na superfície de um elipsóide escolhido

Datum planimétrico

Seleciona-se o elipsóide de referência mais adequado à região

Posiciona-se o elipsóide em relação à Terra real – preservando o paralelismo entre o eixo de rotação da Terra e do elipsóide

Escolhe-se um ponto central (origem) no país ou região e faz-se a anulação do desvio da vertical

Datum planimétrico

Pode ser global (o centro do elipsóide coincide com o centro de massa da Terra) ou local (o centro do elipsóide está deslocado do centro da Terra)

Mensagem importante: as Coordenadas Geográficas, dependem de um Datum planimétrico, pois ele define a referência para os meridianos e paralelos.

Sistema de coordenadas geográficas

Latitude geodésica ou geográfica

ângulo entre a normal à superfície de referência (elipsóide ou esfera), no ponto em questão, e o plano do equador. Varia de 0º a 90º (norte ou sul)

Longitude geodésica ou geográfica

ângulo entre o meridiano que passa pelo ponto e o meridiano origem (Greenwich, por convenção). Varia 0º a 180º (leste e oeste)

φ – latitude geodésica (graus)

λ – longitude geodésica (graus)

h – altitude elipsoidal (metros)

Datum usados no Brasil

- Legalmente:
  - **SAD69** - South American Datum 1969
  - **SIRGAS2000** - Sistema de Referência Geocêntrico para as Américas
- Também é comum encontrar cartas topográficas que referem-se à **Córrego Alegre**, o antigo Datum brasileiro
- 25 de fevereiro de 2005: SIRGAS2000 foi oficialmente adotado como o novo sistema de referência geodésico para o Sistema Geodésico Brasileiro (SGB) e para o Sistema Cartográfico Nacional (SCN)
- Foi também definido um período de transição, não superior a 10 anos, onde o sistema novo (SIRGAS2000) e os antigos (SAD 69, Córrego Alegre) poderão ser utilizados concomitantemente.
- Depois de passado o período de transição, o SIRGAS2000 será o único sistema geodésico de referência legalizado no país.

Datum usados no Brasil

- Diferenças entre o SAD69 e o SIRGAS2000:
  - SAD69 é um sistema de referência topocêntrico que tem como referência um ponto na superfície da Terra
  - SIRGASS2000 é geocêntrico que tem como referência um ponto no centro de massa da Terra
  - SIRGASS2000 atende a uma necessidade de compatibilização com o sistema de posicionamento GPS, que também é geocêntrico

Outros Datum

- Locais
  - SAD69, Córrego Alegre, NAD27, Indian...
- Globais
  - WGS84, SIRGAS, NAD83...

- WGS84 e SIRGAS200 são praticamente idênticos, pois utilizam o mesmo elipsóide de referência (GRS80), com alguns centímetros de diferença no valor do achatamento.

Erros de Posicionamento

- Dados em coordenadas geodésicas, em diferentes Datum, podem gerar erros de posicionamento
- Por exemplo, um mapeamento realizado em SAD69 e outro em SIRGAS2000 não podem ser mostrados no mesmo mapa sem que seja feito algum tratamento

Fonte: http://www.pign.org/PIGN3/Portugues/cadastral.htm

(relatório do Projeto demonstração 2)

Erros de Posicionamento

- De SIRGAS200 para SAD69 : ~65 metros no território brasileiro
- De SIRGAS2000 para WGS84: nenhum
- De Córrego Alegre para SAD69: <= 60 metros

SOLUÇÃO:

- lembre que a variação das coordenadas geográficas afeta a exatidão de sua base de dados
- use um SIG que saiba levar em conta essa variação de coordenadas
- saiba o que está medindo com um receptor GPS
- tenha cuidado com dados compartilhados (importação e exportação)

No mapa a Terra é plana

Processo de criação de um mapa

Projeções Cartográficas

- Impossível representar uma superfície curva num plano sem deformação, por isso apareceu o conceito de Superfície de Projeção

- Superfície de Projeção é uma superfície desenvolvível no plano, capaz de representar um sistema plano de meridianos e paralelos sobre o qual pode ser desenhada uma representação cartográfica (carta, mapa, planta)

Projeções Cartográficas

- Uma projeção cartográfica determina a correspondência matemática biunívoca entre os pontos da esfera (ou elipsóide) e sua transformação num plano

- Sistemas de projeção resolvem as equações:

  (x e y – coordenadas planas, ϕ,λ −coordenadas geográficas)

$$x = f_1(\phi,\lambda) \qquad y = f_2(\phi,\lambda)$$

$$\lambda = g_1(x,y) \qquad \phi = g_2(x,y)$$

Projeções cartográficas

- Impossível representar uma superfície curva num plano sem deformação. Por isso, existem diferentes classes de projeção, que causam diferentes distorções e por isso tem diferentes aplicações
- Uma mesma área sob diferentes projeções geram mapas diferentes

Cilíndrica Plana Cônica

Classes de projeção

Equidistantes: preservam distâncias

Equivalentes: preservam áreas

Conformes: preservam ângulos

Quanto as propriedades

Quanto a superfície de projeção

Projeções Cartográficas

- Superfície ou figura de referência
  - esfera, elipsóide
- Superfície de projeção
  - plano, cone, cilindro, poliedro
- Posição da superfície de projeção
  - normal ou equatorial, oblíqua, transversa
- Método de construção
  - projetivo, analítico

Projeções Cartográficas

- Projeções planas ou azimutais
  - plano tangente ou secante
    - estereográfica polar, azimutal de Lambert
- Projeções cônicas
  - cone tangente ou secante
    - cônica de Lambert, cônica de Albers
- Projeções cilíndricas
  - cilindro tangente ou secante
    - UTM, Mercator, Miller

Projeções Cartográficas

- Projeções conformes ou isogonais
  - preservam ângulos
    - UTM, Mercator, cônica conforme de Lambert
- Projeções equivalentes ou isométricas
  - preservam áreas
    - cônica equivalente de Albers
- Projeções equidistantes
  - representam distâncias em verdadeira grandeza ao longo de certas direções
    - cilíndrica equidistante

Projeções Cartográficas

- Parâmetros das projeções
  - figura de referência (elipsóide ou esfera)
    - datum planimétrico
  - paralelo padrão (latitude reduzida)
    - deformações nulas, escala verdadeira … verdadeira grandeza
    - Um ou dois paralelos se a superfície é tangente ou secante
  - longitude de origem (meridiano central)
    - posição do eixo Y das coordenadas planas
    - para a UTM é o meridiano central de um fuso
  - latitude origem
    - posição do eixo X das coordenadas planas
    - Equador para a maioria das projeções

Projeções Cartográficas

- Sistema UTM – Universal Transversa de Mercator

Projeções Cartográficas

- O sistema UTM é Universal, pois é aplicável em toda a extensão do globo terrestre

Zonas do sistema UTM
(Fonte: Serviço Geodésico Interamericano, s/d)

Projeções Cartográficas Especificações UTM

- Adota 60 cilindros de eixo transverso, de maneira que cada um cobre a longitude de 6º (3º para cada lado do meridiano central)

- Em latitude os fusos são limitados ao paralelo 80º N e S pois, acima deste valor as deformações se acentuam muito

- Para evitar coordenadas negativas a partir da origem das coordenadas (cruzamento Equador com meridiano central) será acrescida em cada fuso das constantes 10.000.000 metros no eixo das ordenadas (NS) e de + 500.000 metros no eixo das abcissas (EW)

Especificações UTM

Universal Transversa de Mercator

- Adota 60 cilindros de eixo transverso, de maneira que cada um cobre a longitude de 6º (3º para cada lado do meridiano central)
- Em latitude os fusos são limitados ao paralelo 80º N e S pois, acima deste valor as deformações se acentuam muito

Universal Transversa de Mercator

- Adota 60 cilindros de eixo transverso, de maneira que cada um cobre a longitude de 6º (3º para cada lado do meridiano central)
- Em latitude os fusos são limitados ao paralelo 80º N e S pois, acima deste valor as deformações se acentuam muito

Universal Transversa de Mercator

- Para evitar coordenadas negativas a partir da origem das coordenadas (cruzamento Equador com meridiano central) será acrescida em cada fuso das constantes 10.000.000 metros no eixo das ordenadas (NS) e de + 500.000 metros no eixo das abscissas (EW)

Projeções Cartográficas

Índice de Nomenclatura das Folhas

Nomenclaturas Folhas

Nomenclaturas Folhas

1: 10.000 (SF 23 - Z – D – VI – 3- SO- F)

1: 25.000 (SF 23 - Z – D – VI – 3- SO)

1: 50.000 (SF 23-Z – D – VI - 3)

Carta Topográfica

CARTA DO BRASIL ESC. 1:50 000

LUTÉCIA

MI-2361/3
FOLHA SF-22-Z-A-III-3

PROJEÇÃO UNIVERSAL TRANSVERSA DE MERCATOR

ARTICULAÇÃO DA FOLHA

QUATÁ	HERCULÂNDIA	POMPÉIA
PARAGUAÇU PAULISTA	LUTÉCIA	EXAPORÃ
MARACAÍ	ASSIS	CAMPOS NOVOS PAULISTA

LOCALIZAÇÃO DA FOLHA NO ESTADO

ESCALA 1: 50 000

1 000 m 0 1 000 2 000 3 000 m

Eqüidistância das curvas de nível: 20 metros

Origem da quilometragem: Equador e Meridiano 51° W. Gr., acrescidas as constantes 10 000 km e 500 km respectivamente

Datum vertical: marégrafo Imbituba, SC
Datum horizontal: Córrego Alegre, MG

Levantamento estereofotogramétrico topográfico regular Aerofotografias - 1965 ; apoio suplementar e reambulação executados em 1973 pelo Departamento de Geodésia e Topografia; restituição, aerotriangulação e preparo para a impressão realizados pelo Departamento de Cartografia

Esta folha foi preparada e impressa em decorrência do Convênio entre o IBGE e o Departamento de Águas e Energia Elétrica do Estado de São Paulo

SUPERINTENDÊNCIA DE CARTOGRAFIA

PRIMEIRA EDIÇÃO — 1974

DIREITOS DE REPRODUÇÃO RESERVADOS

Impressa no Serviço Gráfico do IBGE

A SUPERINTENDÊNCIA DE CARTOGRAFIA agradece a gentileza da comunicação de falhas ou omissões verificadas nesta Folha

Projeções Cartográficas

- Principais projeções no Brasil
  - UTM (Universal Transverse Mercator)
    - cartas topográficas
  - Mercator
    - cartas náuticas
  - Cônica conforme de Lambert
    - cartas ao milionésimo
    - cartas aeronáuticas
  - Policônica
    - mapas temáticos
    - mapas políticos

Projeções Cartográficas

- Outras projeções importantes
  - Cilíndrica equidistante
    - apresentação de dados em SIG
    - mapas mundi
  - Estereográfica polar
    - substitui a UTM nas regiões polares
  - Cônica conforme bipolar oblíqua
    - mapa político das Américas
  - Cônica equivalente de Albers
    - cálculo de área em SIG

Conceitos de Geodésia

- Datum altimétrico ou vertical
  - superfície de referência para a contagem das altitudes (geóide)
  - rede de marégrafos faz medições contínuas para a determinação do nível médio dos mares
  - adota-se um dos marégrafos como ponto de referência do datum vertical
  - no Brasil usa-se o marégrafo de Imbituba, em Santa Catarina

Conceitos de Geodésia

- Relevo
  - Saber se dois ou mais pontos estão no mesmo nível (altitude) – subir ou descer
  - **Nível base** – origem padrão de medidas = nível médio dos mares
  - **Altitude** é a medida do desnível que existe entre qualquer ponto da superfície e o nível do mar.

  - Altitude é DIFERENTE de Altura

Conceitos de Geodésia

- Datum altimétrico ou vertical
  - superfície de referência para a contagem das altitudes (geóide).
  - rede de marégrafos faz medições contínuas para a determinação do nível médio dos mares
  - adota-se um dos marégrafos como ponto de referência do datum vertical
  - no Brasil usa-se o marégrafo de Imbituba, em Santa Catarina

Conceitos de Geodésia

- Altitudes
  - Positivas – elevações, ou seja estão acima do nível base
  - Negativas – depressões, ou seja, estão abaixo do nível base
  - Mapas – representação em **curvas de nível**
  - **Curva de nível são linhas que unem pontos de mesma altitude**

Escala

- É a relação entre a medida de um objeto ou lugar geográfico representado no papel e sua verdadeira dimensão.

- Normalmente é expressa das seguintes formas:
  - Fração representativa ou numérica
  - Gráfica ou escala de barras

Escala

- A escala de um mapa deve representar os detalhes
  - Naturais (rios, mares, montanhas)
  - Artificiais (estradas, pontes, edificações)

- Problemas
  - Necessidade de reduzir as proporções dos acidentes a representar.
  - Determinados acidentes, dependendo da escala, não permitem redução acentuada pois se tornam imperceptíveis.

A solução é se utilizar símbolos cartográficos.

Escala numérica

$$E = d / D$$

d: distância medida na carta

D: distância real

As escalas numéricas mais comuns são da forma:

$E = 1 / 10x$ ou $E = 1:10x$

Escala gráfica

- É a representação gráfica de várias distâncias do terreno sobre uma linha reta graduada
- É constituida de um segmento à direita da referência zero, conhecida como escala primária
- Consiste também de um segmento à esquerda da origem denominada de Talão ou Escala de Fracionamento, que é dividido em sub-múltiplos da unidade escolhida graduada da direita para a esquerda

Escalas especiais

- As fotografias aéreas e grande parte das projeções cartográficas não possuem escalas constantes, elas são variáveis dependendo de uma série de fatores inerentes ao processo de elaboração da projeção.
- As fotografias aéreas, por serem uma projeção central, a escala é variável do centro da foto para a periferia, sendo tanto menor quanto mais próximo das bordas.
- Quando a escala for grande, não ocorrerão muitos problemas pois os erros serão desprezíveis, o que já não ocorrerá em escalas pequenas, podendo ser constante ao longo dos paralelos e variável ao longo dos meridianos, ou vice-versa. Depende do tipo de projeção e da sua estrutura projetiva.

Escalas especiais

- Na projeção de Mercator por exemplo, a escala é variável, constante ao longo dos paralelos e variável ao longo dos meridianos, variando com a latitude, quanto maior a latitude, maior a escala.

PROJEÇÃO DE MERCATOR

Escala em Diferentes Latitudes

1/50 000 000 no Equador - 1/9 132 500 na Latitude de 24

Escala

- Precisão gráfica
  - É a menor grandeza medida no terreno, capaz de ser representada em desenho na mencionada escala.
  - Menor comprimento: 0,2 mm

Seja E = 1 / M

Erro tolerável: 0,0002 metro X M

E = 1/20000 ----- 0.2mm = 4000 mm = 4 m

E = 1/10000 ----- 0,2mm = 2000 mm = 2 m

E = 1/40000 ----- 0,2mm = 8000 mm = 8 m

E = 1/100000 ---- 0,2mm = 20000 mm = 20 m

Escala

- Escolha de escala

Considerando uma região que se queira mapear e que possua muitos acidentes de 10m de extensão, a menor escala que se deve adotar será:

Erro tolerável = 0,0002 metro X M

M = Erro tolerável / 0,0002 metro

M = 10m / 0,0002m = 50.000 ou seja

E = 1:50.000

Escala

- As condicionantes básicas para a escolha de uma escala de representação são:
  - dimensões da área do terreno que será mapeado;
  - tamanho do papel que será traçado o mapa;
  - a orientação da área;
  - erro gráfico;
  - precisão do levantamento e/ou das informações a serem plotadas no mapa.

MINISTÉRIO DA CIÊNCIA E TECNOLOGIA
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

Resumo

Cartografia para geoprocessamento

- Cartografia: preocupa-se em apresentar um modelo de representação de dados para os processos que ocorrem no espaço geográfico
- Geoprocessamento representa a área do conhecimento que utiliza técnicas matemáticas e computacionais, fornecidas pelos Sistemas de Informação Geográfica (SIG), para tratar os processos que ocorrem no espaço geográfico

(fonte: Maguire, Goodchild, Rhind, 1991)

Dados geográficos

Duas **componentes**: o quê e onde

O quê → semântico: altimetria, município, reflectância, estação de coleta, etc.

Onde → localização em um sistema que representa a superfície terrestre, ou seja, Sistema de Referência Espacial

Representação: estrutura de dados usada para armazenar o dado geográfico

Matricial: matriz regular de valores

Vetorial: primitivas geométricas ponto, linha e polígono para representar a localização, associada a um conjunto de valores

Geoprocessamento

Geoprocessamento: disciplina que trata da manipulação de dados geográficos.

SIG: sistema computacional usado para materializar as técnicas de geoprocessamento

Como descrever um SRS?

1. Qual seu nome?
2. Qual é o Datum?
3. É geográfico?
    a. Qual a unidade angular?
    b. Qual o meridiano padrão?
    c. Qual o paralelo de latitude 0?
4. É projetado?
    1. Qual a unidade linear?
    2. Qual a projeção?
    3. Quais os parâmetros da projeção?

Como descrever um SRS?

```
GEOGCS["WGS 84",
   DATUM["WGS_1984",
      SPHEROID["WGS 84",6378137,298.257223563]],
   PRIMEM["Greenwich",0],
   UNIT["degree",0.01745329251994328]]
```

```
PROJCS["WGS 84 / UTM zone 23S",
   GEOGCS["WGS 84",
      DATUM["WGS_1984",
         SPHEROID["WGS 84",6378137,298.257223563],
      PRIMEM["Greenwich",0],
      UNIT["degree",0.01745329251994328]],
   UNIT["metre",1],
   PROJECTION["Transverse_Mercator"],
   PARAMETER["latitude_of_origin",0],
   PARAMETER["central_meridian",-45],
   PARAMETER["scale_factor",0.9996],
   PARAMETER["false_easting",500000],
   PARAMETER["false_northing",10000000],
   AXIS["Easting",EAST],
   AXIS["Northing",NORTH]]
```

WKT – Well Known Text for SRS
OGC – Open Geospatial Consortium

Como descrever um SRS?

```
GEOGCS["WGS 84",
  DATUM["WGS_1984",
    SPHEROID["WGS 84",6378137,298.257223563,
      AUTHORITY["EPSG","7030"]],
    AUTHORITY["EPSG","6326"]],
  PRIMEM["Greenwich",0,
    AUTHORITY["EPSG","8901"]],
  UNIT["degree",0.01745329251994328,
    AUTHORITY["EPSG","9122"]],
  AUTHORITY["EPSG","4326"]]
```

```
PROJCS["WGS 84 / UTM zone 23S",
  GEOGCS["WGS 84",
    DATUM["WGS_1984",
      SPHEROID["WGS 84",6378137,298.257223563,
        AUTHORITY["EPSG","7030"]],
      AUTHORITY["EPSG","6326"]],
    PRIMEM["Greenwich",0,
      AUTHORITY["EPSG","8901"]],
    UNIT["degree",0.01745329251994328,
      AUTHORITY["EPSG","9122"]],
    AUTHORITY["EPSG","4326"]],
  UNIT["metre",1,
    AUTHORITY["EPSG","9001"]],
  PROJECTION["Transverse_Mercator"],
  PARAMETER["latitude_of_origin",0],
  PARAMETER["central_meridian",-45],
  PARAMETER["scale_factor",0.9996],
  PARAMETER["false_easting",500000],
  PARAMETER["false_northing",10000000],
  AUTHORITY["EPSG","32723"],
  AXIS["Easting",EAST],
  AXIS["Northing",NORTH]]
```

SRS ID: identificador dado por uma autoridade
EPSG – European Petroleum Survey Group

Voltando ao exemplo

Tres pessoas foram ao campo, com diferentes equipamentos, para localizar 2 pontos de interesse e voltaram com as seguintes medições:

	PESSOA 1		PESSOA 2		PESSOA 3	
	X	Y	X	Y	X	Y
P2	-42.3591666667	-23.3194444444	770061.694961	7418652.21437	1189337.72907	-2627767.87227
P3	-43.0166666667	-29.5347222222	692193.800396	6731129.17863	1063065.76762	-3318803.38603

Qual delas fez as medições certas?

Resposta:

não posso avaliar sem saber em qual Sistema de Referência Espacial estão esses números.

Voltando ao exemplo

	PESSOA 1		PESSOA 2		PESSOA 3	
	X	**Y**	**X**	**Y**	**X**	**Y**
P2	-42.3591666667	-23.3194444444	770061.694961	7418652.21437	1189337.72907	-2627767.87227
P3	-43.0166666667	-29.5347222222	692193.800396	6731129.17863	1063065.76762	-3318803.38603
	SRID = 4623		SRID = 32723		SRID = 29101	

Qual delas fez as medições certas?

Neste caso, todas estão corretas. Todas se referem a mesma localização, mas medidas em sistemas de referência espacial diferentes.

Conversão de coordenadas

	PESSOA 1		PESSOA 2		PESSOA 3	
	X	Y	X	Y	X	Y
P2	-42.3591666667	-23.3194444444	770061.694961	7418652.21437	1189337.72907	-2627767.87227
P3	-43.0166666667	-29.5347222222	692193.800396	6731129.17863	1063065.76762	-3318803.38603
	SRID = 4623		SRID = 32723		SRID = 29101	

Conversão de coordenadas

Conversão de coordenadas

PROJ.4

PROJ.4 - Cartographic Projections Library

SRS EM DADOS VETORIAIS

Open Geospatial Consortium

- Consórcio entre companhias, universidade e agências governamentais
- Objetivo: promover o desenvolvimento de tecnologias que facilitem a interoperabilidade entre sistemas envolvendo informação geo-espacial
- Os produtos do trabalho do OGC são apresentados sob forma de especificações de interfaces e padrões de intercâmbio

Open Geospatial Consortium

- Algumas especificações OGC:
  - GML (Geography Markup Language): intercâmbio de dados
  - OWS (OGC Web Service): especificações de serviços WEB
    - WFS: Web Feature Service
    - WMS: Web Map Server
  - SFSQL (Simple Feature Specification For SQL): especificações sobre o armazenamento e recuperação de dados espaciais em SGBD's OR

OGC Simple Feature

WKT – Well Known Text for geometry

- Gramática que define como representar textualmente todos as diferentes instâncias de geometrias. Exemplos:

Geometry Type	Text Literal Representation	Comment
Point	Point (10 10)	a Point
LineString	LineString (10 10, 20 20, 30 40)	a LineString with 3 points
Polygon	Polygon ((10 10, 10 20, 20 20, 20 15, 10 10))	a Polygon with 1 exteriorRing and 0 interiorRings

WKT

Multipoint	MultiPoint ((10 10), (20 20))	a MultiPoint with 2 points
MultiLineString	MultiLineString ((10 10, 20 20), (15 15, 30 15))	a MultiLineString with 2 linestrings
MultiPolygon	MultiPolygon (((10 10, 10 20, 20 20, 20 15, 10 10)), ((60 60, 70 70, 80 60, 60 60)))	a MultiPolygon with 2 polygons
GeomCollection	GeometryCollection (POINT (10 10), POINT (30 30), LINESTRING (15 15, 20 20))	a GeometryCollection consisting of 2 Point values and a LineString value
PolyhedralSurface	PolyhedralSurface Z (((0 0 0, 0 0 1, 0 1 1, 0 1 0, 0 0 0)), ((0 0 0, 0 1 0, 1 1 0, 1 0 0, 0 0 0)), ((0 0 0, 1 0 0, 1 0 1, 0 0 1, 0 0 0)), ((1 1 0, 1 1 1, 1 0 1, 1 0 0, 1 1 0)), ((0 1 0, 0 1 1, 1 1 1, 1 1 0, 0 1 0)), ((0 0 1, 1 0 1, 1 1 1, 0 1 1. 0 0 1)))	A polyhedron cube, corner at the origin and opposite corner at (1, 1, 1).

WKT

Tin	Tin Z (((0 0 0, 0 0 1, 0 1 0, 0 0 0)), ((0 0 0, 0 1 0, 1 0 0, 0 0 0)), ((0 0 0, 1 0 0, 0 0 1, 0 0 0)), ((1 0 0, 0 1 0, 0 0 1, 1 0 0)),)	A tetrahedron (4 triangular faces), corner at the origin and each unit coordinate digit.
Point	Point Z (10 10 5)	a 3D Point
Point	Point ZM (10 10 5 40)	the same 3D Point with M value of 40
Point	Point M (10 10 40)	a 2D Point with M value of 40

WKB – Well Known Binary for geometry

- Versão binária para representar as geometrias. Pode ser usada na especificação do armazenamento

Key

1 WKB Polygon
2 ring 1
3 ring 2

Figure 25: Well-known Binary Representation for a geometric object
in NDR format (B = 1)
of type Polygon (T = 3)
with 2 LinearRings (NR = 2)
each LinearRing having 3 points (NP = 3)

Códigos para as geometrias

Type	Code
Geometry	0
Point	1
LineString	2
Polygon	3
MultiPoint	4
MultiLineString	5
MultiPolygon	6
GeometryCollection	7
CircularString	8
CompoundCurve	9
CurvePolygon	10
MultiCurve	11
MultiSurface	12
Curve	13
Surface	14
PolyhedralSurface	15
TIN	16

Type	Code
Geometry Z	1000
Point Z	1001
LineString Z	1002
Polygon Z	1003
MultiPoint Z	1004
MultiLineString Z	1005
MultiPolygon Z	1006
GeometryCollection Z	1007
CircularString Z	1008
CompoundCurve Z	1009
CurvePolygon Z	1010
MultiCurve Z	1011
MultiSurface Z	1012
Curve Z	1013
Surface Z	1014
PolyhedralSurface Z	1015
TIN Z	1016

Type	Code
Geometry M	2000
Point M	2001
LineString M	2002
Polygon M	2003
MultiPoint M	2004
MultiLineString M	2005
MultiPolygon M	2006
GeometryCollection M	2007
CircularString M	2008
CompoundCurve M	2009
CurvePolygon M	2010
MultiCurve M	2011
MultiSurface M	2012
Curve M	2013
Surface M	2014
PolyhedralSurface M	2015
TIN M	2016

Type	Code
Geometry ZM	3000
Point ZM	3001
LineString ZM	3002
Polygon ZM	3003
MultiPoint ZM	3004
MultiLineString ZM	3005
MultiPolygon ZM	3006
GeometryCollection ZM	3007
CircularString ZM	3008
CompoundCurve ZM	3009
CurvePolygon ZM	3010
MultiCurve ZM	3011
MultiSurface ZM	3012
Curve ZM	3013
Surface ZM	3014
PolyhedralSurface ZM	3015
TIN ZM	3016

OGC Well-Known Text for SRS

```
GEOGCS["WGS 84",
   DATUM["WGS_1984",
      SPHEROID["WGS 84",6378137,298.257223563]],
   PRIMEM["Greenwich",0],
   UNIT["degree",0.01745329251994328]]
```

```
PROJCS["WGS 84 / UTM zone 23S",
   GEOGCS["WGS 84",
      DATUM["WGS_1984",
         SPHEROID["WGS 84",6378137,298.257223563],
      PRIMEM["Greenwich",0],
      UNIT["degree",0.01745329251994328]],
   UNIT["metre",1],
   PROJECTION["Transverse_Mercator"],
   PARAMETER["latitude_of_origin",0],
   PARAMETER["central_meridian",-45],
   PARAMETER["scale_factor",0.9996],
   PARAMETER["false_easting",500000],
   PARAMETER["false_northing",10000000],
   AXIS["Easting",EAST],
   AXIS["Northing",NORTH]]
```

WKT – Well Known Text for SRS
OGC – Open Geospatial Consortium

SRS EM DADOS MATRICIAIS

Georreferenciamento

- Trata da aplicação de transformações geométricas para corrigir as distorções inerentes ao processo de aquisição da imagem
- Fontes de distorções geométricas (sensores MSS, TM, HRV, AVHRR, WFI)
  - rotação da Terra (skew)
  - distorções panorâmicas (compressão)
  - curvatura da Terra (compressão)
  - arrastamento da imagem durante uma varredura
  - variações de altitude, atitude e velocidade do satélite

Georeferenciamento

- Para que possam ser integradas a base geográfica as imagens devem passar por vários níveis de processamento para corrigir as distorções radiométricas e geométricas.. Ex:
  1. Com correção radiométrica
  2. Com correção de sistema (radiométrica + boresishts + atitude + efemérides)
  3. Georreferenciada (sistema + pontos de controle 2D)
  4. Ortorretificada (sistema + pontos de controle 2D + DTM)

Georeferenciamento

- Exemplo de como diferentes produtores de imagens chamam seus níveis de processamento

	Correção			
Satélite	Radiométrica	Sistema	Georrefenciada	Ortorretificada
SPOT	1A	2A	2B	3
Landsat	0	1R	1G	1P (precision)
CBERS	1	2	3	4

- Atualmente as imagens são distribuídas com correção > 2, porém a necessidade de integrar imagens antigas, ou devido a problemas nos parâmetros do satélite, muitas vezes os próprios usuários tem que fazer algumas correções geométricas.

Como representar o georeferenciamento?

$$x = x_0 + i*r_x$$
$$y = y_0 + j*r_y$$

$$x = x_0 + i*r_x + i*r_y$$
$$y = y_0 + j*r_y + j*r_x$$

* r_y deve ser considerada negativo

Como representar o georeferenciamento?

Brasilia.tif

+

Brasilia.tfw

30	→ R_X
0.0	→ Rotação em relação ao eixo X
0.0	→ Rotação em relação ao eixo Y
-30	→ $-1 * R_Y$
183557.00	→ X_0
8.258940.00	→ Y_0

Como representar o gereferenciamento?

brasilia.tif

- SRS=…
- Res X=…
- Res Y=…
- PCs = [….]

Geo TAGs

Imagem

OGC para representações matriciais?

- Não existe um padrão para representações matriciais nem em modo texto, nem em modo binário

- No nível de outras especificações (a serem vistas adiante) GeoTiff é considerado um formato padrão.

MINISTÉRIO DA CIÊNCIA E TECNOLOGIA
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

SQL - Structured Query Language

Karine Reis Ferreira

karine@dpi.inpe.br

SQL - Structured Query Language

- Linguagem padrão (ISO) para sistemas de bancos de dados
- É uma linguagem declarativa de alto nível que permite:
  - Consultar dados
  - Definir e alterar dados
  - Definir visões
  - Especificar autorização e regras de segurança
  - Definir restrições de integridade
  - Criar índices
  - Controlar transações
  - ...

SQL - Structured Query Language

- Linguagem de consulta usada pelos SGBD-R e SGBD-OR
- Baseada na álgebra relacional
- É divida em:
  - Linguagem de manipulação de dados (SQL DML)
  - Linguagem de definição de dados (SQL DDL)
  - Definição de visões (SQL DDL)
  - Especificação de autorização (SQL DDL)
  - Especificação de integridade (SQL DDL)
  - Controle de transação (SQL DDL)

SQL - Structured Query Language

SQL-DDL

```
CREATE DATABSE Teste
```

```
CREATE TABLE Estados (
NOME      VARCHAR(100)
SIGLA     VARCHAR(2)
POP       NUMBER(10,10))
```

SQL-DML

```
INSERT INTO Estados
VALUES ("Minas
Gerais", "MG", 9999)
```

```
SELECT *
FROM   Estados
WHERE  SIGLA = "MG"
```

SQL - Structured Query Language

Alguns comandos SQL:

Comandos	Usado para	Tipo
select	Consultar dados	DML
insert, *update*, *delete*	Incluir, alterar e remover dados	DML
commit, *rollback*	Controlar transações	DDL
create, *alter*, *drop*	Definir, alterar e remover esquemas (tabelas)	DDL

SQL - Structured Query Language

DDL– Data Definition Language

CREATE DATABASE – cria um novo banco de dados

ALTER DATABASE – modifica um banco de dados

CREATE TABLE – cria uma nova tabela

ALTER TABLE – altera uma tabela

DROP TABLE – remove uma tabela

CREATE INDEX – cria um índice

DROP INDEX – remove um índice

SQL - Structured Query Language

DML – Data Manipulation Language

SELECT – extrai dados de um banco de dados

UPDATE – altera os dados de um banco de dados

DELETE – apaga dados de um banco de dados

INSERT INTO – insere dados no banco de dados

SQL – Create database – Example

```
CREATE DATABASE lab_bdgeo
WITH OWNER = postgres
ENCODING = 'UTF8'
TABLESPACE = pg_defaultt;
```

SQL – Create table

```
CREATE TABLE <table name>
(  <column name> <column type> [<attribute constraint>]
{, <column name> <column type> [<attribute constraint>] }
[<table constraint> {, <table constraint> } ] )
```

[…]: opcional

{…}: repetições -> 0 or *n* vezes

| : mutualmente exclusivos

SQL – Column type

Numeric	Integer numbers	`INT` ou `INTEGER`, `SMALLINT`
	Floating-point numbers	`FLOAT` ou `REAL`, `DOUBLE PRECISION`
	Formated numbers: *i* (precision): number of decimal digits and *j* (scale): number of digits after decimal point	`DECIMAL (i,j)` ou `DEC(i,j)` ou `NUMERIC(i,j)`
Character-string	Fixed length with *n* characters	`CHAR(n)` ou `CHARACTER(n)`
	Varying length with maximum *n* characters	`VARCHAR(n)` ou `CHAR VARYING(n)` ou `CHARACTER VARYING(n)`
	Large text values (ex. documents)	`CHARACTER LARGE OBJECT (CLOB)`
Bit-string	Fixed length with *n* bits	`BIT(n)`
	Varying length with maximum *n* bits	`BIT VARYING(n)`
	Large binary values (ex. images)	`BIT LARGE OBJECT (BLOB)`

SQL – Column type

Boolean	Values of TRUE or FALSE or UNKNOWN	`BOOLEAN`
Date	YEAR, MONTH, and DAY (YYYY-MM-DD)	`DATE`
Time	HOUR, MINUTE, and SECOND (HH:MM:SS) with or without time zone (HOURS:MINUTES)	`TIME` e `TIME WITH TIME ZONE`
Timestamp	Both date and time, with or without time zone	`TIMESTAMP` e `TIMESTAMP WITH TIME ZONE`
Time interval	A relative value that can be used to increment or decrement an absolute value of a date, time, or timestamp.	`INTERVAL`

SQL – Constraints

Restringir que um atributo não tenha valores nulos	`NOT NULL`
Restringir valores e domínios de atributos	`CHECK (<expression>)`
Restringir que um ou mais atributos tenham valores únicos	`UNIQUE (<column_name> {,<column_name>})`
Definir chave primária	`PRIMARY KEY (<column_name> {,<column_name>})`

Definir restrições de integridade referencial (chave estrangeira)	`FOREIGN KEY (<column_name> {,<column_name>})`
	`REFERECES <table_name>`
	`(<column_name> {,<column_name>})`
	`ON DELETE (SET DEFAULT | SET NULL | CASCADE)`
	`ON UPDATE (SET DEFAULT | SET NULL | CASCADE)`

SQL – Create table – Examples

```
CREATE TABLE Cliente (
    ClienteID   INT            NOT NULL,
    Empresa     VARCHAR( 40 )  NOT NULL,
    Endereco    VARCHAR( 60 ),
    Cidade      VARCHAR( 15 ),
    Estado      VARCHAR( 15 ),
    CEP         VARCHAR( 10 ),
    Pais        VARCHAR( 15 ),
    Fone        VARCHAR( 24 ),
    CONSTRAINT ClientePK PRIMARY KEY (ClienteID)
);


CREATE TABLE Fornecedor (
    FornecedorID    INT             NOT NULL,
    Nome            VARCHAR( 40 )   NOT NULL,
    Endereco        VARCHAR( 60 ),
    CONSTRAINT FornecedorPK PRIMARY KEY (FornecedorID)
);

CREATE TABLE Produto (
    ProdutoID       INT             NOT NULL,
    FornecedorID    INT             NOT NULL,
    Descricao       VARCHAR( 40 )   NOT NULL,
    PrecoUnitario   REAL            NOT NULL  CHECK(PrecoUnitario>=0),

    CONSTRAINT ProdutoPK        PRIMARY KEY (ProdutoID),

    CONSTRAINT FornecProdutoFK  FOREIGN KEY (FornecedorID) REFERENCES Fornecedor(FornecedorID)
                                ON DELETE CASCADE
                                ON UPDATE CASCADE
);
```

SQL – Create table – Examples

```
CREATE TABLE Vendedor (
    VendedorID  INT             NOT NULL,
    Nome        VARCHAR( 40 )   NOT NULL,

    PRIMARY KEY ( VendedorID )
);

CREATE TABLE Pedido (
    PedidoID    INT             NOT NULL,
    VendedorID  INT             NOT NULL,
    ClienteID   INT             NOT NULL,
    ProdutoID   INT             NOT NULL,
    Quantidade  INT             NOT NULL    DEFAULT 1  CHECK(Quantidade>0),

    PRIMARY KEY ( PedidoID ),

    CONSTRAINT PedidoVendedorFK FOREIGN KEY (VendedorID) REFERENCES Vendedor(VendedorID)
                                ON DELETE CASCADE
                                ON UPDATE CASCADE,

    CONSTRAINT PedidoClienteFK FOREIGN KEY (ClienteID) REFERENCES Cliente(ClienteID)
                                ON DELETE CASCADE
                                ON UPDATE CASCADE,

    CONSTRAINT PedidoProdutoFK FOREIGN KEY (ProdutoID) REFERENCES Produto(ProdutoID)
                                ON DELETE CASCADE
                                ON UPDATE CASCADE
);
```

SQL – Create table – Examples

Diagrama criado com o aplicativo DBDesigner 4.

DBDesigner 4

- ✓ DBDesigner 4 is a visual database design system that integrates database design, modeling, creation and maintenance into a single, seamless environment.
- ✓ Open Source (GPL)
- ✓ http://www.fabforce.net/dbdesigner4/
- ✓ Developed and optimized for the open source MySQL-Database, but it can create standard SQL scripts from its diagrams

DBDesigner 4 - [DB Model | projeto_bancoCliente_DBDesigner4]

File Edit Display Database Plugins Options Windows Help

Fornecedor
- FornecedorID: INTEGER
- Nome: VARCHAR(40)
- Endereco: VARCHAR(60)

FornecProdutoFK

Produto
- ProdutoID: INTEGER
- FornecedorID: INTEGER (FK)
- Descricao: VARCHAR(40)
- PrecoUnitario: REAL

PedidoProdutoFK

Cliente
- ClienteID: INTEGER
- Empresa: VARCHAR(40)
- Endereco: VARCHAR(60)
- Cidade: VARCHAR(15)
- Estado : VARCHAR(15)
- CEP: VARCHAR(10)
- Pais: VARCHAR(15)
- Fone: VARCHAR(24)

PedidoClienteFK

Pedido
- PedidoID: INTEGER
- ProdutoID: INTEGER (FK)
- ClienteID: INTEGER (FK)
- VendedorID: INTEGER (FK)
- Quantidade: INTEGER

PedidoVendedorFK

Vendedor
- VendedorID: INTEGER
- Nome: VARCHAR(40)

Navigator & Info

Navigator | Info

113%

Datatypes

Common | All types

- INTEGER
- FLOAT
- VARCHAR
- DATETIME
- BOOL
- TEXT
- LONGBLOB
- Varchar(20)
- Varchar(45)
- Varchar(255)

DB Model

Model

All Tables
- Cliente
- Fornecedor
- Produto
- Vendedor
- Pedido

113 % | The model was successfully saved to C:\Users\karine\Dropbox\Apresentações\2014_aulas_disciplina_BDGeo\projeto_bancoCliente_DBDesigner4.xml. | Not connected to a Database

```sql
CREATE TABLE Vendedor (
    VendedorID  INT             NOT NULL,
    Nome        VARCHAR( 40 )   NOT NULL,

    PRIMARY KEY ( VendedorID )
);

CREATE TABLE Pedido (
    PedidoID    INT             NOT NULL,
    VendedorID  INT             NOT NULL,
    ClienteID   INT             NOT NULL,
    ProdutoID   INT             NOT NULL,
    Quantidade  INT             NOT NULL    DEFAULT 1  CHECK(Quantidade>0),

    PRIMARY KEY ( PedidoID ),

    CONSTRAINT PedidoVendedorFK FOREIGN KEY (VendedorID) REFERENCES Vendedor(VendedorID)
                                ON DELETE CASCADE
                                ON UPDATE CASCADE,

    CONSTRAINT PedidoClienteFK FOREIGN KEY (ClienteID) REFERENCES Cliente(ClienteID)
                                ON DELETE CASCADE
                                ON UPDATE CASCADE,

    CONSTRAINT PedidoProdutoFK FOREIGN KEY (ProdutoID) REFERENCES Produto(ProdutoID)
                                ON DELETE CASCADE
                                ON UPDATE CASCADE
);
```

SQL scripts from diagrams

SQL – Insert table

```
INSERT INTO <table name>
[ ( <column name> {, <column name> } ) ]
( VALUES ( <constant value>, { <constant value> } )
{, ( <constant value> {, <constant value> } ) }
| <select statement> )
```

[…]: opcional

{…}: repetições -> 0 or *n* vezes

| : mutualmente exclusivos

SQL – Insert table – Examples

```sql
INSERT INTO Cliente Values ( 1, 'ACM', 'Rua das Flores, 10', 'Sao Paulo', 'SP', '1222000', 'Brasil', '112233445566');
INSERT INTO Cliente Values ( 2, 'VW', 'Rua do Comercio, 47', 'Sao Paulo', 'SP', '1222010', 'Brasil', '11298735566');
INSERT INTO Cliente Values ( 3, 'GM', 'Via Dutra, 1000', 'Sao Jose dos Campos', 'SP', '1222560', 'Brasil', '122239876566');
INSERT INTO Cliente Values ( 4, 'TEX', 'AV Brasil, 1210', 'Rio de Janeiro', 'RJ', '348890', 'Brasil', '212134567');

INSERT INTO Vendedor Values ( 1, 'Jose Marcio');
INSERT INTO Vendedor Values ( 2, 'Luis Claudio');
INSERT INTO Vendedor Values ( 3, 'Andre Carlos');

INSERT INTO Fornecedor Values ( 1, 'Ferragens Santa Lucia', 'Rua Catalao, 20, Goiania, GO');
INSERT INTO Fornecedor Values ( 2, 'Borracharia Campos', 'Rua dos Ipes 1235, Presidente Prudente, SP');
INSERT INTO Fornecedor Values ( 3, 'Tintas Brasil', 'Avenida dos Guararapes 44, Paulinia, SP');

INSERT INTO Produto Values ( 1, 2, 'Roda', 500.00);
INSERT INTO Produto Values ( 2, 1, 'Mola', 234.00);
INSERT INTO Produto Values ( 3, 1, 'Porca', 11.00);
INSERT INTO Produto Values ( 4, 1, 'Parafuso', 5.30);
INSERT INTO Produto Values ( 5, 2, 'Prego', 1.20);
INSERT INTO Produto Values ( 6, 3, 'Tinta', 234.00);

INSERT INTO Pedido Values ( 1, 2, 4, 2, 450);
INSERT INTO Pedido Values ( 2, 1, 2, 1, 123);
INSERT INTO Pedido Values ( 3, 2, 1, 2, 60);
INSERT INTO Pedido Values ( 4, 3, 2, 2, 121);
INSERT INTO Pedido Values ( 5, 3, 3, 6, 65);
INSERT INTO Pedido Values ( 6, 1, 3, 5, 36);
INSERT INTO Pedido Values ( 7, 2, 1, 5, 140);
INSERT INTO Pedido Values ( 8, 3, 4, 1, 200);
INSERT INTO Pedido Values ( 9, 3, 2, 3, 67);
INSERT INTO Pedido Values ( 10, 1, 2, 3, 89);
```

SQL – Insert table – Examples

```sql
CREATE TABLE Cliente2 (
    ClienteID   INT             NOT NULL,
    Empresa     VARCHAR( 40 )   NOT NULL,
    Endereco    VARCHAR( 60 ),
    Cidade      VARCHAR( 50 ),
    Estado      VARCHAR( 15 ),
    CEP         VARCHAR( 10 ),
    Pais        VARCHAR( 15 ),
    Fone        VARCHAR( 24 ),
    CONSTRAINT Cliente2PK PRIMARY KEY (ClienteID)
);

INSERT INTO Cliente2 (SELECT * FROM Cliente);
```

SQL – Catalog

- SQL2 uses the concept of a catalog: a named collection of schemas in an SQL environment.

- Schema INFORMATION_SCHEMA: information on all the schemas in the catalog and all the element descriptors in these schemas.

- Tables:

CHECK_CONSTRAINTS	TABLES
COLUMNS	TABLE_CONSTRAINTS
COLUMN_PRIVILEGES	TABLE_PRIVILEGES
CONTRAINT_COLUMN_USAGE	USER_DEFINED_TYPES
CONSTRAINT_TABLE_USAGE	VIEWS
REFERENTIAL_CONSTRAINT	…

SQL – Catalog – Examples

```
SELECT * FROM information_schema.tables

SELECT * FROM information_schema.columns

SELECT * FROM information_schema.views

SELECT * FROM information_schema.triggers

...
```

SQL – Select-From-Where

```
SELECT [ DISTINCT ] <attribute list>
FROM   (<table name> {<alias>} | <joined table> )
       {(<table name> {<alias>} | <joined table>)}
[ WHERE <condition> ]
[ GROUP BY    <grouping attributes>
              [HAVING <group selection condition>] ]
[ ORDER BY    <column name> [<order>]
              {, <column name> [<order>] }]
```

[…]: opcional

{…}: repetições -> 0 or *n* vezes

| : mutualmente exclusivos

SQL – Select-From-Where

```
SELECT [ DISTINCT ] <attribute list>

 <attribute list> := …
```

Define quais colunas farão parte do resultado da consulta

Equivale ao operador *projeção* da álgebra relacional

Opções	Descrição
DISTINCT	Indica que as linhas duplicadas devem ser eliminadas do resultado
*	Indica que todas as colunas de todas as tabelas da cláusula FROM devem ser incluídas no resultado
<column_name>	Nome de uma coluna de uma tabela da cláusula FROM que será incluída no resultado.
<function>	Funções definidas em SQL como, por exemplo, funções de agregação (ex.: *avg*, *min*, *max*, *count*, etc)

SQL – Select-From-Where

```
FROM   (<table name> {<alias>}
                | <joined table> )
       {(<table name> {<alias>}
                | <joined table>)}
```

Define quais tabelas serão consultadas

Equivale ao operador *produto cartesiano* da álgebra relacional

Opções	Descrição
`<alias>`	Nome alternativo para uma coluna, expressão ou tabela
`<table_name>`	Nome de uma tabela envolvida na consulta
`<jointed_table>`	Junção de tabelas envolvidas na consulta

SQL – Jointed Table

```
SELECT *
FROM table1 INNER JOIN table2 ON table1.id = table2.id
```

```
SELECT *
FROM table1 LEFT JOIN table2 ON table1.id = table2.id
```

```
SELECT *
FROM table1 RIGHT JOIN table2 ON table1.id = table2.id
```

SQL – Select-From-Where

```
[ WHERE <condition> ]
```

Define quais as restrições que as linhas das tabelas da cláusula FROM devem satisfazer para entrarem no resultado

Equivale ao operador *seleção* da álgebra relacional

Opções	Descrição
`<condition>`	Uma condição à qual as linhas das tabelas da cláusula FROM devem satisfazer para entrarem no resultado

SQL – Select-From-Where

```
[ GROUP BY <grouping attributes>
        [HAVING
        <group selection condition>] ]
```

GROUP BY: Indica que o resultado deve ser agrupado

HAVING: Indica quais os grupos gerados pela cláusula GROUP BY entrarão no resultado

Opções	Descrição
<column_name>	Uma ou mais colunas cujos valores serão usados para agrupar o resultado.

Opções	Descrição
<group_selection_condition>	Uma condição à qual os grupos gerados pela cláusula GROUP BY devem satisfazer para entrarem no resultado.

SQL – Select-From-Where

```
[ ORDER BY    <column name> [<order>]
              {, <column name>
                     [<order>] }]
```

Indica como o resultado deve ser ordenado

Opções	Descrição
`<column_name> [(ASC | DESC)]`	Uma ou mais colunas cujos valores serão usados para ordenar o resultado. A ordenação pode ser de forma ASCENDENTE ou DESCENDENTE.

SQL – Operadores de Agregação

- AVG(…): média dos valores da coluna
- SUM(…): soma dos valores da coluna
- COUNT(…): número de valores na coluna
- MAX(…): maior valor na coluna
- MIN(…): menor valor na coluna
- ...

Podem ser aplicados pra todos os registros de uma coluna ou para grupos de registros (usando a cláusula GROUP BY)

SQL – Select-From-Where – Examples

- ✓ Selecione todos os clientes ordenados pela empresa
- ✓ Selecione as empresas, enderecos e telefones de todos os clientes, ordenado pelo endereco
- ✓ Quantos clientes existem?

```
SELECT * FROM cliente ORDER BY empresa;
```

```
SELECT empresa, endereco, fone FROM cliente
ORDER BY endereco;
```

```
SELECT COUNT (*) FROM cliente;
```

SQL – Select-From-Where – Examples

- ✓ Selecione todos os pedidos do cliente "ACM"
- ✓ Quantos itens o cliente "ACM" comprou?

```
SELECT *
FROM cliente INNER JOIN pedido
     ON cliente.clienteid = pedido.clienteid
WHERE cliente.empresa = 'ACM';


SELECT SUM (pedido.quantidade)
FROM cliente INNER JOIN pedido
     ON cliente.clienteid = pedido.clienteid
WHERE cliente.empresa = 'ACM';
```

SQL – Select-From-Where – Examples

- ✓ Quantos itens cada cliente comprou?
- ✓ Quais clientes compraram mais que 200 itens?

```sql
SELECT cliente.empresa, SUM (pedido.quantidade)
FROM cliente INNER JOIN pedido
     ON cliente.clienteid = pedido.clienteid
GROUP BY cliente.empresa;
```

```sql
SELECT cliente.empresa, SUM (pedido.quantidade)
FROM cliente INNER JOIN pedido
     ON cliente.clienteid = pedido.clienteid
GROUP BY cliente.empresa
HAVING SUM (pedido.quantidade) > 200
```

SQL – Select-From-Where – Examples

- ✓ Selecione todas as informações dos pedidos: identificador do pedido, nome do vendedor, descricao do produto, nome do fornecedor e quantidade comprada.

```
SELECT  pedido.pedidoid as pedidoid, vendedor.nome as vendedor,
        cliente.empresa as cliente, produto.descricao as produto,
        fornecedor.nome as fornecedor, pedido.quantidade
FROM (((pedido INNER JOIN vendedor ON pedido.vendedorid = vendedor.vendedorid)
     INNER JOIN cliente ON pedido.clienteid = cliente.clienteid)
     INNER JOIN produto ON pedido.produtoid = produto.produtoid)
     INNER JOIN fornecedor ON produto.fornecedorid = fornecedor.fornecedorid
ORDER BY cliente
```

SQL – Create view

Tabela Virtual ou *View* é uma tabela que é derivada de outras tabelas e não existe fisicamente armazenada no banco de dados.

```
CREATE VIEW <view_name>
[(<column_name> {, <column_name> })]
AS <select statement>
```

SQL – Create view – Examples

```
CREATE VIEW pedido_descricao (pedidoid, vendedor, empresa, produto, fornecedor, quantidade)
AS
SELECT  pedido.pedidoid, vendedor.nome, cliente.empresa, produto.descricao,
        fornecedor, pedido.quantidade
FROM    (((pedido INNER JOIN vendedor ON pedido.vendedorid = vendedor.vendedorid)
        INNER JOIN cliente ON pedido.clienteid = cliente.clienteid)
        INNER JOIN produto ON pedido.produtoid = produto.produtoid)
        INNER JOIN fornecedor ON produto.fornecedorid = fornecedor.fornecedorid
```

```
SELECT * FROM pedido_descricao
```

SQL – Update

- ✓ Altera valores dos registros das tabelas

```
UPDATE <table name>
SET <column name> = <new value>
{, <column name> = <new value>}
[ WHERE <condition> ]
```

SQL – Update – Examples

```sql
UPDATE cliente
SET endereco = 'Rua das Flores, 505'
WHERE empresa = 'ACM'



UPDATE pedido
SET quantidade = quantidade * 2
```

SQL – Delete

- ✓ Remove registros das tabelas

```
DELETE <table name>
[ WHERE <condition> ]
```

SQL – Delete – Examples

```sql
DELETE FROM vendedor
WHERE nome = 'Andre Carlos'

DELETE FROM vendedor
```

OBS 1: Note que após executar o primeiro comando, todos os pedidos associados ao vendedor “Andre Carlos” são removidos da tabela “Pedido”. Isso acontece porque a restrição entre as tabelas “Vendedor” e “Pedido” foi criada com a ação “ON DELETE CASCADE”!

OBS 2: Note que após executar o segundo comando, todos os vendedores e pedidos são removidos do banco. Isso acontece porque a restrição entre as tabelas “Vendedor” e “Pedido” foi criada com a ação “ON DELETE CASCADE”!

SQL – Alter table

```
ALTER TABLE <table name> ADD <column definition>
ALTER TABLE <table name> ADD COLUMN <column definition>

ALTER TABLE <table name> DROP COLUMN <column name> <action>

ALTER TABLE <table name>   ALTER COLUMN <column name>
                           <new column definition>
ALTER TABLE <table name>   ALTER COLUMN <column name>
                           TYPE <new column type>
```

SQL – Alter table

```
ALTER TABLE <table name> DROP CONSTRAINT <constraint name>
                                         <action>

ALTER TABLE <table name> RENAME COLUMN  <column name> TO
                                         <new column name>

ALTER TABLE <table name> RENAME TO <new table name>
```

SQL – Alter table – Examples

```
ALTER TABLE Cliente ADD CPF VARCHAR( 14 ) NOT NULL DEFAULT 0000000

ALTER TABLE Cliente ADD COLUMN CPF2 VARCHAR( 14 ) NOT NULL DEFAULT 0000000

ALTER TABLE Cliente DROP COLUMN CPF2 CASCADE

ALTER TABLE Cliente ALTER COLUMN CPF DROP NOT NULL

ALTER TABLE Cliente ALTER COLUMN CPF TYPE VARCHAR( 150 )
```

SQL – Drop table

```
DROP TABLE <table name> [(CASCADE | RESTRICT)]
```

- ✓ CASCADE: exclui também todos os objetos relacionados ao objeto excluído
- ✓ RESTRICT: o objeto só é excluído se não há nenhum outro objeto relacionado a ele. (opção default)

SQL – Drop table – Examples

```sql
DROP TABLE Vendedor

DROP TABLE Vendedor CASCADE
```

OBS: Note que após executar o segundo comando, todas as restrições (*constraints*) relacionadas a essa tabela são removidas.

SQL – Drop table – Examples

```sql
DROP TABLE vendedor CASCADE;
DROP TABLE fornecedor CASCADE;
DROP TABLE prodduto CASCADE;
DROP TABLE cliente CASCADE;
DROP TABLE pedido CASCADE;
```

OBS: Os comandos acima removem todas as tabelas do banco de dados

MINISTÉRIO DA CIÊNCIA E TECNOLOGIA
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

Introdução a Sistemas de Banco de Dados

Karine Reis Ferreira

karine@dpi.inpe.br

Sistemas de Banco de Dados

- Características:
  - Coleção de arquivos inter-relacionados + Conjunto de programas para armazenar, gerenciar e acessar esses arquivos.

- Funcionalidades:
  - Consistência e integridade dos dados
  - Segurança
  - Controle de acesso concorrente (multi-usuário)
  - Backup e recuperação de falhas

Sistemas de Banco de Dados

Bancos de Dados:
é uma coleção de dados
relacionados de um determinado domínio.

Sistemas de Banco de Dados

SGBD – Sistemas Gerenciadores de Bancos de Dados ou

DBMS – Database Management Systems

Sistema Gerenciador de Bancos de Dados (SGBD ou DBMS):
é uma coleção de programas para criar, gerenciar e manipular um banco de dados.

Exemplos: MySQL, Oracle, PostgreSQL, ...

Sistemas de Banco de Dados

Sistemas de Banco de Dados

GUI

Prompt

API

```
int main()
{
…
}
```

SGBD

Banco de Dados

Sistemas de Banco de Dados

Sistema de Bancos de Dados:

Banco de Dados
+
SGBD
+
Programas de Aplicação

SGBD

- Armazenar, consultar e atualizar o banco de dados de maneira eficiente
  - Linguagem de definição de dados (SQL-DDL)
  - Linguagem de consulta e manipulação de dados (SQL-DML)
- Manter a consistência e integridade dos dados
  - Evitar redundância
- Controlar acessos concorrentes (multiusuários)
- Manter a segurança dos dados
  - Recuperar falhas e fazer cópias de reserva (backup)
  - Restringir e controlar os acessos dos usuários
  - Restrições de segurança

Alguns SGBDs bem conhecidos

IBM

DB2. Universal Database

ORACLE®

MySQL® | Sun microsystems

Microsoft® SQL Server® 2008

IBM Informix SOFTWARE

The database company™

PostgreSQL

Exemplo de aplicação

TerraView

Sistemas de Informações Geográficas (SIG): Sistema computacional capaz de armazenar, processar e manipular dados geográficos (Worboys and Duckham, 2004)

Sistemas de Informações Geográficas (SIG)

- *Geographic Information System* (*GIS*)
- Funcionalidades de um SIG (Rigaux et al, 2002):
  - Entrada e validação de dados espaciais;
  - Armazenamento e gerenciamento;
  - Saída e apresentação visual;
  - Transformação de dados espaciais;
  - Interação com o usuário;
  - Combinação de dados espaciais - novas representações do espaço
  - Ferramentas para análise espacial.

Evolução dos SIGs

Armazenamento de dados geográficos

Arquitetura Dual

Arquitetura Integrada

MINISTÉRIO DA CIÊNCIA E TECNOLOGIA
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

Modelo Conceitual de Entidade-Relacionamento (ER)

Projeto de Bancos de Dados

Figure 3.1
A simplified diagram to illustrate the main phases of database design.

Fonte: Elmasri et. al. 2007

Projeto Conceitual

- Abstração do mundo real
- Descrição detalhada das entidades, relacionamentos e restrições.
- Esquema conceitual de BD independente do SGBD
- Modelo Entidade-Relacionamento (ER)

Projeto de Bancos de Dados

Figure 3.1
A simplified diagram to illustrate the main phases of database design.

Fonte: Elmasri et. al. 2007

Projeto Lógico

- Mapeamento do modelo conceitual para o modelo do SGBD (relacional)
- Dependente de SGBD
- Modelo Relacional

Projeto de Bancos de Dados

Figure 3.1
A simplified diagram to illustrate the main phases of database design.

Fonte: Elmasri et. al. 2007

Projeto Físico

- Definição dos métodos de acesso
- Particionamento físico de tabelas
- Controle do Buffer de memória, tamanho de página no disco...

Modelo de dados

Conjunto de conceitos usados para representar os dados, seus relacionamentos e as regras que garantem a sua consistência.

No processo de modelagem é necessário construir uma abstração dos objetos e fenômenos do mundo real.

[Yeung and Hall, 2011]

Modelo Entidade-Relacionamento (E-R)

- Introduzido em 1976 por Peter Chen, é a abordagem mais adotada para modelagem conceitual de dado. Tem por objetivo facilitar o projeto do banco de dados representando sua estrutura lógica
- Definição: modelo baseado na percepção do mundo real como um conjunto de objetos chamados entidades e pelo conjunto de relacionamentos entre esses objetos.
- Ao longo do tempo diferentes versões de foram sendo propostas para representar diferentes conceitos.
- Um modelo entidade-relacionamento é um modelo conceitual independente de hardware ou software

Modelo Entidade-Relacionamento (E-R)

- Conceitos básicos:
  - <u>Entidades</u>
    - Objetos básicos do mundo real
    - Um conjunto de entidades agrupa entidades do mesmo tipo
  - <u>Relacionamentos</u>
    - Associação entre conjuntos de entidades
  - <u>Atributos</u>
    - Associados a entidades e a relacionamentos
    - Uma entidade é representada por um conjunto de atributos
    - Cada atributo possui um domínio

Modelo Entidade-Relacionamento (E-R)

- Representação:
  - Retângulos: conjunto de entidades
  - Elipses: atributos
  - Losangos: relacionamentos

Modelo Entidade-Relacionamento (E-R)

- Atributo identificador
  - Atributo que vai identificar unicamente cada instância da entidade (chave primária)
  - Devem ser sublinhados

Modelo Entidade-Relacionamento (E-R)

- Cardinalidade:
  - expressa o número de entidades que uma entidade pode estar associada com
  - 1:1, 1:n, n:1, n:n

Modelo Entidade-Relacionamento (E-R)

- Exemplos de cardinalidade: 1:1

Relação correta

Relação Incorreta

Modelo Entidade-Relacionamento (E-R)

- Exemplos de cardinalidade: 1:N

Relação correta

Relação incorreta

Modelo Entidade-Relacionamento (E-R)

- Exemplos de cardinalidade: N:N

Relação correta

Modelo Entidade-Relacionamento (E-R)

- Relacionamento parcial:
  - Quando não existe obrigatoriedade de todas as entidades de um conjunto participarem do relacionamento
    - Ex. Podem existir proprietários sem cadastro
    - A participação da entidade “proprietário” no relacionamento “possui” é parcial.

Modelo Entidade-Relacionamento (E-R)

- Relacionamento total (dependência de existência):
  - Quando todas as entidades de um conjunto participam obrigatoriamente do relacionamento
    - Ex. Todo cadastro tem que estar a associado a um proprietário.
    - A participação da entidade “cadastro” no relacionamento “possui” é total.

Modelo Entidade-Relacionamento (E-R)

- Entidade fraca:
  - Quando a existência dessa entidade depende da existência de uma outra entidade (dependência de existência).

Modelo Entidade-Relacionamento (E-R)

- Auto relacionamento:
  - Associa entidades de um conjunto a entidades desse mesmo conjunto
    - Ex. Um lote pode ser composto por outros lotes

Modelo Entidade-Relacionamento (E-R)

- Relacionamento múltiplo:
  - Associa mais de dois conjuntos de entidades
    - Ex. Um proprietário pode possuir vários lotes, cada um com um contrato de compra e venda diferente.

Modelo Entidade-Relacionamento (E-R)

- Agregação:
  - Uma limitação do ER: expressar relacionamentos entre relacionamentos
  - Agregação é uma abstração na qual os relacionamentos são tratados como conjunto de entidades de nível superior, permitindo o relacionamento entre relacionamentos.

Modelo Entidade-Relacionamento (E-R)

- Ex.: um proprietário pode estar associado a vários bancos. Pra cada banco associado, o proprietário pode ter um ou mais empréstimos.

Modelo Entidade-Relacionamento (E-R)

- Ex.: um proprietário pode estar associado a vários bancos. Pra cada banco associado, o proprietário pode ter um ou mais empréstimos.

Redundância: todo par proprietário-banco faz parte dos relacionamentos "associado" e "financiado"

Modelo Entidade-Relacionamento (E-R)

- Agregação:
  - Ex.: um proprietário pode estar associado a vários bancos. Pra cada banco associado, o proprietário pode ter um ou mais empréstimos.

Modelo Entidade-Relacionamento (E-R)

- Generalização
  - Enfatiza a semelhança entre diferentes tipos de entidades e abstrai suas diferenças.
- Especialização:
  - Subdivide entidades semelhantes em conjuntos de entidades mais específicas e enfatiza suas diferenças.

Modelo Entidade-Relacionamento (E-R)

- Generalização e Especialização
  - Ex.: Uma quadra pode ser do tipo comercial ou residencial

Modelo Entidade-Relacionamento (E-R)

O que é um bom modelo?

- Facilita a descrição de consultas corretas e compreensíveis
- Os elementos do modelo tem um significado intuitivo
- O modelo é o mais simples possível, mas não mais simples que isso
- É suficientemente abstrato para não sofrer alterações face a alterações menores no domínio do problema
- Se o domínio do problema se altera significantemente é fácil modificar o modelo de dados (suficientemente flexível)
- Mais tarde é necessário considerar o impacto do modelo de dados na eficiência das operações no banco de dados

MINISTÉRIO DA CIÊNCIA E TECNOLOGIA
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

Modelo Relacional, Álgebra Relacional e SQL

Modelo Relacional

- O modelo relacional foi inventado por E.F. (Ted) Codd como um modelo geral de dados (~1970)
- Baseia-se no princípio de que todos os dados são representados matematicamente como relações *n-árias*, onde uma *relação n-ária* é um subconjunto do produto cartesiano de *n* domínios
- Dados são manipulados através de um *cálculo* ou *álgebra relacional*
- Diferentemente do modelo E-R, um modelo relacional implica uma implementação para um SGBD relacional em particular:
  - IBM, Informix, Microsoft, Oracle, Sybase, etc.

Modelo Relacional

- A maioria dos SGBDs seguem um modelo relacional ou objeto-relacional
- **Modelo de Dados Relacional**: Banco de dados é organizado em uma coleção de relações ou tabelas relacionadas entre si.

Aluno

MATRICULA	NOME	CURSOID
98765	João	MAT
67765	José	BIO
84562	Maria	ENG
34256	Luis	INFO
3452672	Ana	MAT
34529	Luana	MAT

Curso

CURSOID	TITULO	DURAÇÃO
INFO	Informática Indust.	4
BIO	Biologia	4
ENG	Engenharia Civil	5
MAT	Licenciatura Mat.	4

Modelo Objeto-Relacional

- É uma extensão do Modelo Relacional com conceitos da modelagem Orientada por Objeto.
- Combina os benefícios dos dois modelos.
- Fornecem suporte para:
  - Criar objetos complexos
  - Executar consultas avançadas sobre dados complexos
- A linguagem de consulta OR é uma extensão da linguagem SQL para suportar o modelo de objetos

Modelo Objeto-Relacional

Exemplo: Oracle

```
CREATE TYPE SDO_GEOMETRY AS OBJECT (
SDO_GTYPE       NUMBER,
SDO_SRID        NUMBER,
SDO_POINT       SDO_POINT_TYPE,
SDO_ELEM_INFO   SDO_ELEM_INFO_ARRAY,
SDO_ORDINATES   SDO_ORDINATE_ARRAY);
```

```
CREATE TABLE Estados (
SIGLA    VARCHAR(2)
POP      NUMBER(10,10)
GEOM     SDO_GEOMETRY)
```

Estados

SIGLA	POP	GEOM
MG	222222	
RJ	333333	
SP	444444	

Modelo Relacional: Relação

Conceito de *relação*:

- Dados os domínios D_1, D_2, ..., D_n não necessariamente distintos, uma relação é definida como:

$$R = \{ (d_1, d_2,..., d_n) \mid d_1 \in D_1, d_2 \in D_2,..., d_n \in D_n \}$$

- O conjunto (d_1, d_2,..., d_n) de valores ordenados define uma *tupla*
- Uma relação é o conjunto de *n-tuplas* ordenadas, onde *n* define o grau da relação

Modelo Relacional: Relação

- No dia a dia, usamos o termo tabela ao invés de relação
- Cada tabela possui um nome
- Cada linha representa uma tupla da Relação*:*
  - Algumas pessoas usam o termo registro ao invés de linhas
- A ordem das linhas não é significante:
  - Mas todas as linhas são distintas
- A ordem das colunas é significante:
  - Corresponde à ordem D_1, D_2, ..., D_n dos domínios sobre os quais a Relação é definida
- Cada coluna (também chamada de atributo ou campo) possui um nome

Modelo Relacional: Relação

Domínio:
{MAT, BIO, ENG, INFO }
ou
cadeia de caracteres com 4 dígitos

Coluna

Nome da Tabela ou Relação

Aluno

matricula	nome	curso_id
98765	João	MAT
67765	José	BIO
84562	Maria	ENG
34256	Luis	INFO
3452672	Ana	MAT
34529	Luana	MAT

Linhas ou Tuplas

ou Registro

Domínio:
{Números Inteiros }

Domínio:
{Caracteres de A..Z }

Instâncias x Esquemas

Esquema da relação Aluno

Instâncias da relação Aluno

Aluno

matricula	nome	curso_id
98765	João	MAT
67765	José	BIO
84562	Maria	ENG
34256	Luis	INFO
3452672	Ana	MAT
34529	Luana	MAT

Modelo Relacional: Operações

- Os operadores existentes realizam operações de recuperação que geram novas tabelas a partir das existentes
- Permitem, por exemplo, extrair um subconjunto das linhas ou colunas
- Ex: informações aluno 34256

Aluno

matricula	nome	curso_id
98765	João	MAT
67765	José	BIO
84562	Maria	ENG
34256	Luis	INFO
3452672	Ana	MAT
34529	Luana	MAT

Operação Seleção

Resultado:

Aluno

matricula	nome	curso_id
34256	Luis	INFO

Modelo Relacional: Chaves

- Chave Primária (*Primary Key* ou *PK*):
  - Campo ou conjunto de campos cujos valores identificam unicamente cada linha de uma tabela

chave primária

Aluno

matricula	nome	curso_id
98765	João	MAT
67765	José	BIO
84562	Maria	ENG
34256	Luis	INFO
3452672	Ana	MAT
34529	Luana	MAT

chave primária composta

TelefoneAluno

matricula	fone	tipo
98765	9876-0001	CEL
98765	551-4243	RES
67765	551-4243	RES
84562	8123-0179	CEL
34256	8888-8888	CEL
34529	9999-9999	CEL

chave primária

Curso

curso_id	titulo	duracao
INFO	Infomática Indust.	4
BIO	Biologia	4
ENG	Engenharia Civil	5
MAT	Licenciatura Mat.	4

Modelo Relacional: Chaves

- Chave Estrangeira (*Foreign Key* ou *FK*):
  - Coluna ou combinação de colunas, cujos valores aparecem necessariamente na chave primária de uma outra tabela
  - Obs.: uma chave estrangeira não precisa ter o mesmo nome do que a chave primária correspondente na outra tabela

chave estrangeira

Aluno

matricula	nome	curso_id
98765	João	MAT
67765	José	BIO
84562	Maria	ENG
34256	Luis	INFO
3452672	Ana	MAT
34529	Luana	MAT

Relacionamento entre as tabelas

chave primária

Curso

curso_id	titulo	duracao
INFO	Infomática Indust.	4
BIO	Biologia	4
ENG	Engenharia Civil	5
MAT	Licenciatura Mat.	4

Restrições de Integridade

- Também conhecidas como *constraints*
- Especificam critérios para manter a consistência dos dados
- Tipos de restrições de integridade:
  - Restrições de domínio: especifica qual o tipo de cada atributo
  - Restrições de chave: chave primária e candidata (valores únicos)
  - Restrições de nulidade: especifica se o valor de um atributo pode ou não ser nulo
  - Restrição de integridade valor (semântica ou física):
    - Exemplo: o preço de um produto não pode ser zero (preço >=0)
  - Restrições de integridade referencial ou chave estrangeira:
    - Especifica uma restrição de integridade referencial entre duas relações R_1 e R_2

Transação

Conjunto de procedimentos executados por um SGBD e que para o usuário é visto como uma única ação.

Ex.: transferência de R$ 500,00 do cliente A para o cliente B.

tempo	Cliente A	Cliente B
T1	Verificar saldo	
T2	Debitar R$500	
T3		Depositar R$500

Erro no sistema

Transação

A integridade da transação depende de 4 propriedades (ACID):

(1) **Atomicidade**: Uma transação nunca é executada pela metade.

(2) **Corretude ou Consistência**: A execução de uma transação isolada (sem execução concorrente de outra transação) preserva a consistência do banco.

(3) **Isolamento**: Transações concorrentes não podem ser intercaladas de forma a gerar inconsistência.

(4) **Durabilidade**: Se a transação completar-se com sucesso, todas as mudanças feitas no banco de dados persistem, até mesmo se houver falas no sistema (Mecanismo para recuperar falhas - Rollback).

Trigger

***Trigger* (gatilho)**: um gatilho é um comando executado automaticamente pelo sistema como um efeito colateral de uma modificação no banco de dados. Exemplo: sempre que alguém alterar a nota de um aluno, recalcule sua média.

Notas

Aluno	Nota	Disciplina
001	6.0	Banco de Dados
001	5.0	Estatística
001	8.5	Fisica

Media

Aluno	Media
001	6.5
002	7.25

Álgebra Relacional

- Linguagem de consultas procedural
- Conjunto de operações que usam uma ou duas relações como entrada e geram uma relação de saída
  - *operação* (REL_1) → REL_2
  - *operação* (REL_1,REL_2) → REL_3
- Operações básicas:
  - Operações unárias:
    - seleção, projeção, renomeação
- Operações binárias:
    - produto cartesiano, união e diferença

Álgebra Relacional - Resumo

Operadores da Álgebra Relacional

- Seleção:
  - seleciona tuplas que satisfazem um certo predicado ou condição

Clientes

Nome	***Registro***
João	1
Maria	2
José	3

a) selecionar tuplas cujo nome = João

$\sigma_{nome=\text{“João”}}$ (Clientes)

Nome	***Registro***
João	1

Operadores da Álgebra Relacional

b) selecionar as tuplas de Clientes cujo registro > 1

$\sigma_{registro>1}$ (Clientes)

Nome	***Registro***
Maria	2
José	3

c) selecionar as tuplas de Clientes com registro > 1 e registro < 3

$\sigma_{registro>1 \wedge registro < 3}$ (Clientes)

Nome	***Registro***
Maria	2

Operadores da Álgebra Relacional

- Projeção:
  - gera novas relações excluindo alguns atributos
  - exemplo: projete o atributo nome sobre a relação Clientes

$$\Pi_{nome}\,(\text{Clientes})$$

Clientes

Nome	***Registro***
João	1
Maria	2
José	3

→

Nome
João
Maria
José

Operadores da Álgebra Relacional

- União:
  - união de atributos do mesmo domínio que estão em relações diferentes
  - as relações devem possuir o mesmo número de atributos
  - exemplo: encontre todos os clientes da agência que possuem conta corrente ou empréstimo.
    - Relações existentes na agência:
      - ContaCorrente e Emprestimo

Operadores da Álgebra Relacional

- União: Π_{nome} (ContaCorrente) U Π_{nome} (Emprestimo)

ContaCorrente

Nome	***Conta***
João	1
Maria	2
José	3

Emprestimo

Nome	***Empréstimo***
Paulo	100
Maria	200
Carlos	300

=

Resultado da união

Nome
João
Maria
José
Paulo
Carlos

Operadores da Álgebra Relacional

- Diferença:
  - tuplas que se encontram em uma relação, mas não em outra
  - exemplo: encontre todos clientes sem empréstimo

$$\Pi_{nome}(\text{ContaCorrente}) - \Pi_{nome}(\text{Emprestimo})$$

ContaCorrente

Nome	***Conta***
João	1
Maria	2
José	3

-

Emprestimo

Nome	***Empréstimo***
Paulo	100
Maria	200
Carlos	300

=

Resultado da diferença

Nome
João
José

Operadores da Álgebra Relacional

- Produto Cartesiano
  - Faz todas as combinações entre as tuplas de duas relações
  - Gera uma nova relação formada pela união dessas combinações
  - Exemplo: produto cartesiano entre os clientes e os empréstimos de Maria

$\sigma_{\text{emprestimo.nome = “Maria”}}$ (ContaCorrente X Emprestimo)

Nome$_{cc}$	***Conta***	***Nome$_{emp}$***	***Empréstimo***
João	1	Maria	200
Maria	2	Maria	200
José	3	Maria	200

Operadores da Álgebra Relacional

- Operadores derivados:
  - Intersecção
    - Seleciona tudo que está em ambas relações
    - Exemplo: todos os clientes que possuem empréstimo

$$\Pi_{nome}(\text{ContaCorrente}) \cap \Pi_{nome}(\text{Emprestimo})$$

ContaCorrente

Nome	***Conta***
João	1
Maria	2
José	3

Emprestimo

Nome	***Empréstimo***
Paulo	100
Maria	200
Carlos	300

=

Resultado da intersecção

Nome
Maria

Operadores da Álgebra Relacional

- Operadores derivados
  - Junção
    - Inclui um produto cartesiano, seguido de uma seleção (pode ter projeção ao final)
    - Exemplo: nomes dos clientes com conta corrente e número de empréstimo:

$$\Pi_{\text{contacorrente.nome, emprestimo.emprestimo}}$$

$$(\sigma_{\text{contacorrente.nome = emprestimo.nome}} (\text{ContaCorrente X Emprestimo}))$$

$$\downarrow$$

$$\Pi_{\text{contacorrente.nome, emprestimo.emprestimo}}$$

$$(\text{ContaCorrente} \bowtie \text{Emprestimo}))$$

SQL - Structured Query Language

- Linguagem padrão (ISO) para sistemas de bancos de dados
- É uma linguagem declarativa de alto nível que permite:
  - Consultar dados
  - Definir e alterar dados
  - Definir visões
  - Especificar autorização e regras de segurança
  - Definir restrições de integridade
  - Criar índices
  - Controlar transações
  - ...

SQL - Structured Query Language

- Linguagem de consulta usada pelos SGBD-R e SGBD-OR
- Baseada na álgebra relacional
- É divida em:
  - Linguagem de manipulação de dados (SQL DML)
  - Linguagem de definição de dados (SQL DDL)
  - Definição de visões (SQL DDL)
  - Especificação de autorização (SQL DDL)
  - Especificação de integridade (SQL DDL)
  - Controle de transação (SQL DDL)

SQL - Structured Query Language

SQL-DDL

```
CREATE DATABSE Teste
```

```
CREATE TABLE Estados (
NOME      VARCHAR(100)
SIGLA     VARCHAR(2)
POP       NUMBER(10,10))
```

SQL-DML

```
INSERT INTO Estados
VALUES ("Minas
Gerais", "MG",
9999)
```

```
SELECT *
FROM   Estados
WHERE  SIGLA = "MG"
```

SQL - Structured Query Language

- Alguns comandos em SQL

Comandos	Usado para	Tipo
select	Consultar dados	DML
insert, *update*, *delete*	Incluir, alterar e remover dados	DML
commit, *rollback*	Controlar transações	DDL
create, *alter*, *drop*	Definir, alterar e remover esquemas (tabelas)	DDL

SQL - Structured Query Language

```
CREATE TABLE cliente
(nome            CHAR(20) NOT NULL,
 endereço        CHAR(30),
 cidade          CHAR(30),
 PRIMARY KEY     (nome))
```

```
ALTER TABLE cliente ADD RG CHAR(10)
```

```
SELECT nome, endereco
FROM cliente
WHERE cidade = 'São José dos Campos'
```

SQL - Structured Query Language

DDL– Data Definition Language

CREATE DATABASE – cria um novo banco de dados

ALTER DATABASE – modifica um banco de dados

CREATE TABLE – cria uma nova tabela

ALTER TABLE – altera uma tabela

DROP TABLE – remove uma tabela

CREATE INDEX – cria um índice

DROP INDEX – remove um índice

SQL - Structured Query Language

DML – Data Manipulation Language

SELECT – extrai dados de um banco de dados

UPDATE – altera os dados de um banco de dados

DELETE – apaga dados de um banco de dados

INSERT INTO – insere dados no banco de dados

SQL - Structured Query Language

Próxima aula - laboratório: SQL na prática!!!!

Modelo Entidade-Relacionamento (E-R)

Conversão E-R→ Modelo Relacional

- Entidades com atributos chaves bem definidos geram uma relação.
- Relacionamentos podem gerar uma relação adicionando-se os atributos chaves das entidades relacionadas e os atributos do relacionamento.
- Entidades com atributos chaves não bem definidos geram uma relação adicionando-se a chave da relação que dependem.

Conversão E-R→ Modelo Relacional - Exemplo

DER

Relações

Lote (id, area_total, area_const)

Lote_proprietario (id_lote, cpf, data)

Proprietario (cpf, nome)

Conversão E-R→ Modelo Relacional

- Cada entidade é traduzida para uma tabela.
- Cada atributo (simples) da entidade define uma coluna da tabela.
- A coluna correspondente ao atributo identificador é chave primária

Ex:

Lote(id, area_total, area_const)

Conversão E-R→ Modelo Relacional

- Relacionamento
  - A tradução do relacionamento depende da cardinalidade das entidades que participam do relacionamento.
  - Formas básicas de tradução:
    - Tabela própria
    - Colunas adicionais dentro da tabela de entidade

Conversão E-R→ Modelo Relacional

- Relacionamento 1:N ou N:1

Lote(id, area_total, area_const, **num_quadra**)

Quadra(num_quasea, area)

Conversão E-R→ Modelo Relacional

- Relacionamento N:N

Lote(id_lote, area_total, area_const)

Fronteira(**id_lote**, **id_rua**, num_inicial, num_final)

Rua(id_rua, nome)

Conversão E-R→ Modelo Relacional

- Auto relacionamento

Lote(<u>id_lote</u>, area_total, area_const)

Composição(**<u>id_lote</u>**, **<u>id_lote_comp</u>**, quantidade)

Conversão E-R→ Modelo Relacional

- Relacionamento múltiplo

Lote(<u>id_lote</u>, area_total, area_const)

Proprietario(<u>cpf</u>, nome)

Contrato(<u>id_contrato</u>, documento)

Lote_Prop_Contr(**<u>id_lote</u>**, **<u>cpf</u>**, **<u>id_contrato</u>**, data)

Conversão E-R→ Modelo Relacional

- Especialização

Solução 1

Quadra_comercial (num_quadra, area, imposto_servico)

Quadra_residencial (num_quadra, area, num_residencias)

Conversão E-R→ Modelo Relacional

- Especialização

Solução 2

Quadra(num_quadra, area)

Quadra comercial (num_quadra, imposto_servico)

Quadra_residencial (num_quadra, num_residencias)

Fases de projeto de um Banco de Dados

- Projeto Conceitual
  - Abstração do mundo real
  - Gera um esquema conceitual de BD independente do SGBD
- Projeto Lógico
  - O esquema conceitual é mapeado para o modelo de implementação de dados do SGBD
- Projeto Físico
  - Especificação das necessidades de recursos do SGBD como estruturas de dados, métodos de acesso e segurança

Projeto Lógico de BD

- Normalização
  - Processo pelo qual um esquema de tabelas (relações) insatisfatório é quebrado de forma que seus atributos formem relações menores que sejam mais adequadas:
    - Sem redundância de informações
    - Maior facilidade de manutenção
  - Baseado em varias regras de normalização:
    - 1ª forma normal
    - 2ª forma normal
    - 3ª forma normal
- Regra de ouro: entre com o mínimo de data necessário, evite duplicar informação com menor risco para a integridade dos dados

Normalização

- Exemplo:

solicitacao_compra

num_solic	data_solic	cod_func	nome_func	contato	cod_prod	desc_prod	quant_prod
001	12/06/09	func01	Joao Americo	39436523 39456444	2345	papel	3
001	12/06/09	func01	Joao Americo	39436523 39456444	2398	tinta	2
001	12/06/09	func01	Joao Americo	39436523 39456444	4300	impressora	1
002	6/04/03	func02	Luis Souza	39436518	2345	papel	1
002	6/04/03	func02	Luis Souza	39436518	1200	CD	1

Problemas da Tabela Não-Normalizada

- Redundância dos dados
  - Possui vários grupos repetidos
- Anomalias de inserção
  - Inserir um novo funcionário
  - Inserir um novo produto
- Anomalias de atualização
  - Alterar o nome de um funcionário
- Anomalias de remoção
  - Remover um produto

num_solic	data_solic	cod_func	nome_func	contato	cod_prod	desc_prod	quant_prod
001	12/06/09	func01	Joao Americo	39436523 39456444	2345	papel	3
001	12/06/09	func01	Joao Americo	39436523 39456444	2398	tinta	2
001	12/06/09	func01	Joao Americo	39436523 39456444	4300	impressora	1
002	6/04/03	func02	Luis Souza	39436518	2345	papel	1
002	6/04/03	func02	Luis Souza	39436518	1200	CD	1

Problemas da Tabela Não-Normalizada

- Redundância dos dados
  - Possui vários grupos repetidos
- Anomalias de inserção
  - Inserir um novo funcionário
  - Inserir um novo produto
- Anomalias de atualização
  - Alterar o nome de um funcionário
- Anomalias de remoção
  - Remover um produto

num_solic	data_solic	cod_func	nome_func	contato	cod_prod	desc_prod	quant_prod
001	12/06/09	func01	Joao Americo	39436523 39456444	2345	papel	3
001	12/06/09	func01	Joao Americo	39436523 39456444	2398	tinta	2
001	12/06/09	func01	Joao Americo	39436523 39456444	4300	impressora	1
002	6/04/03	func02	Luis Souza	39436518	2345	papel	1
002	6/04/03	func02	Luis Souza	39436518	1200	CD	1

Problemas da Tabela Não-Normalizada

- Redundância dos dados
  - Possui vários grupos repetidos
- Anomalias de inserção
  - Inserir um novo funcionário
  - Inserir um novo produto

Gera linhas incompletas!

- Anomalias de atualização
  - Alterar o nome de um funcionário
- Anomalias de remoção
  - Remover um produto

num_solic	data_solic	cod_func	nome_func	contato	cod_prod	desc_prod	quant_prod
001	12/06/09	func01	Joao Americo	39436523 39456444	2345	papel	3
001	12/06/09	func01	Joao Americo	39436523 39456444	2398	tinta	2
001	12/06/09	func01	Joao Americo	39436523 39456444	4300	impressora	1
002	6/04/03	func02	Luis Souza	39436518	2345	papel	1
002	6/04/03	func02	Luis Souza	39436518	1200	CD	1

Problemas da Tabela Não-Normalizada

- Redundância dos dados
  - Possui vários grupos repetidos
- Anomalias de inserção
  - Inserir um novo funcionário
  - Inserir um novo produto
- Anomalias de atualização
  - Alterar o nome de um funcionário
- Anomalias de remoção
  - Remover um produto

Tem que alterar várias linhas!

num_solic	data_solic	cod_func	nome_func	contato	cod_prod	desc_prod	quant_prod
001	12/06/09	func01	Joao Americo	39436523 39456444	2345	papel	3
001	12/06/09	func01	Joao Americo	39436523 39456444	2398	tinta	2
001	12/06/09	func01	Joao Americo	39436523 39456444	4300	impressora	1
002	6/04/03	func02	Luis Souza	39436518	2345	papel	1
002	6/04/03	func02	Luis Souza	39436518	1200	CD	1

Problemas da Tabela Não-Normalizada

- Redundância dos dados
  - Possui vários grupos repetidos
- Anomalias de inserção
  - Inserir um novo funcionário
  - Inserir um novo produto
- Anomalias de atualização
  - Alterar o nome de um funcionário
- Anomalias de remoção
  - Remover um produto

Remove informação sobre os funcionários

num_solic	data_solic	cod_func	nome_func	contato	cod_prod	desc_prod	quant_prod
001	12/06/09	func01	Joao Americo	39436523 39456444	2345	papel	3
001	12/06/09	func01	Joao Americo	39436523 39456444	2398	tinta	2
001	12/06/09	func01	Joao Americo	39436523 39456444	4300	impressora	1
002	6/04/03	func02	Luis Souza	39436518	2345	papel	1
002	6/04/03	func02	Luis Souza	39436518	1200	CD	1

Normalização

- 1ª Forma normal:
  - Uma relação esta na 1FN se, e somente se, todos os domínios contiverem apenas valores atômicos.
  - Uma relação está na 1FN quando seus atributos não contém grupos de repetição
  - Uma maneira de trazer uma tabela para a 1FN é separar as entidades claramente identificadas em tabelas separadas

Normalização

- 1ª Forma normal:

solicitacao_compra

num_solic	data_solic	cod_func	nome_func	cod_prod	desc_prod	quant_prod
001	12/06/03	func01	Joao Silva	2345	papel	3
001	12/06/03	func01	Joao Silva	2398	tinta	2
001	12/06/03	func01	Joao Silva	4300	impressora	1

1FN

solicitacao_compra

num_solic	data_solic	cod_func	nome_func
001	12/06/03	func01	Joao Silva

solicitacao_produtos

num_solic	cod_prod	desc_prod	quant_prod
001	2345	papel	3
001	2398	tinta	2
001	4300	impressora	1

Normalização

- Dependência funcional
  - Dada uma relação R, o atributo Y de R é funcionalmente dependente do atributo X de R

$$(R.X \rightarrow R.Y)$$

se, e somente se, sempre que duas tuplas de R têm o mesmo valor para X elas tem também o mesmo valor para Y.

  - Ex.:

cod_func → nome_func

cod_prod → desc_prod

num_solic, cod_prod → quant_prod

Normalização

- 2ª Forma normal:
  - Uma relação está na segunda forma normal se, e apenas se, estiver na 1FN, e cada atributo não-chave for totalmente dependente funcional da chave primária.
  - Ocorre quando a chave primária é composta por mais de um campo.
    - verificar se todos os campos que não fazem parte da chave dependem de todos os campos que compõem a chave. Se algum campo depender somente de parte da chave composta, então este campo deve pertencer a outra tabela.

Normalização

- 2ª Forma normal:

solicitacao_produtos

num_solic	cod_prod	desc_prod	quant_prod
001	2345	papel	3
001	2398	tinta	2
001	4300	impressora	1

2FN

produtos

cod_prod	desc_prod
2345	papel
2398	tinta
4300	impressora

solicitacao_produtos

num_solic	cod_prod	quant_prod
001	2345	3
001	2398	2
001	4300	1

Normalização

- 2ª Forma normal - resultado:

solicitacao_compra

num_solic	data_solic	cod_func	nome_func
001	12/06/03	func01	Joao Silva

produtos

cod_prod	desc_prod
2345	papel
2398	tinta
4300	impressora

solicitacao_produtos

num_solic	cod_prod	quant_prod
001	2345	3
001	2398	2
001	4300	1

Normalização

- 3ª Forma normal:
  - Um relação está na terceira forma normal se e apenas se, estiver na 2FN, e não tiver dependências transitivas
  - *Dependência transitiva*: ocorre quando um atributo não-chave, além de depender da chave primária da tabela, depende funcionalmente de outro atributo ou combinação de atributos não-chave.
  - Em uma tabela na 3FN não existem atributos não-chave que tenham dependência de outros atributos não chave.

Normalização

- 3ª Forma normal:

solicitacao_compra

num_solic	data_solic	cod_func	nome_func
001	12/06/03	func01	Joao Silva

3FN

funcionarios

cod_func	nome_func
func01	Joao Silva

solicitacao_compra

num_solic	data_solic	cod_func
001	12/06/03	func01

Normalização

- 3ª Forma normal - resultado:

funcionarios

cod_func	nome_func
func01	Joao Silva

solicitacao_compra

num_solic	data_solic	cod_func
001	12/06/03	func01

produtos

cod_prod	desc_prod
2345	papel
2398	tinta
4300	impressora

solicitacao_produtos

num_solic	cod_prod	quant_prod
001	2345	3
001	2398	2
001	4300	1

MINISTÉRIO DA CIÊNCIA E TECNOLOGIA
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

Exemplos de SGBDs relacionais e objeto-relacionais

- Auto-contido: necessita do mínimo de suporte de outras bibliotecas ou sistemas operacionais
- Não é um servidor: é responsável por ler e escrever os arquivos da base, sem comunicação entre processos
- Não requer configuração
- Transacional
  - Transação: uma instrução simples e lógica sobre o banco de dados. Ex: transferência de fundos de uma conta para outra.
  - **A**tomic: tudo ou nada
  - **C**onsistent: de um estado consistente para outro
  - **I**solated: uma transação não altera os dados manipulados por outra
  - **D**urable: transações que são executadas com sucesso, não são perdidas

SQLite Studio

- SQLiteStudio is a SQLite database manager with the following features:
- Single executable file - no need to install or uninstall. Binary distribution is just the single, ready to use file.
- Intuitive interface,
- All SQLite3 and SQLite2 features wrapped within simple GUI,
- Cross-platform - runs on Windows 9x/2k/XP/2003/Vista/7, Linux, MacOS X, Solaris, FreeBSD and should work on other Unixes (not tested yet).
- Localizations, currently translated to: English, Polish, Spanish, German, Russian, Japanese, Italian, Dutch, Chinese,

SQLite Studio

- Exporting to various formats (SQL statements, CSV, HTML, XML),
- Numerous small additions, like formatting code, history of queries executed in editor windows, on-the-fly syntax checking, and more,
- UTF-8 support,
- skinnable (interface can look native for Windows 9x/XP, KDE, GTK, Mac OS X, or draw widgets to fit for other environments, WindowMaker, etc),
- Configurable colors, fonts and shortcuts.
- Open source and free - Released under GPLv2 licence.

PostgreSQL

http://www.postgresql.org

PostgreSQL

- Site oficial: http://www.postgresql.org
- Sistema Gerenciador de Bancos de Dados Objeto-Relacional
- Um dos mais avançados SGBD open source: licença BSD
- Multi-plataforma: AIX, Solaris, Linux, OpenBSD, Windows e outros
- Características:
  - Funcões:
  - PL/PgSQL, PL/Perl, PL/Python, PL/Tcl, C, C++, ...
  - Triggers (gatilhos)
  - Índices: B+-tree, hash, GiST
  - Transação e Concorrência
  - Chaves estrangeiras (foreign keys)
  - Tipos de dados e funções definidos pelo usuário

Utilização do PostgreSQL

- Através de interfaces de programação:
  - C, PHP, Java (JDBC), ODBC, Python, Perl

- Através de terminais interativos:
  - ferramentas que permitem estabelecer comunicação com o servidor de bancos de dados e enviar comandos SQL para que o servidor faça o processamento.
  - Os dois terminais mais conhecidos são:
    - psql (terminal – sem interface gráfica)
    - **PgAdmin III** (ferramenta gráfica)

PgAdmin III

pgAdmin III
Arquivo
Editar
Visualizar
Ferramentas
Ajuda
Navegador de objetos
Servidores (1)
PostgreSQL Database Server 8.3 (localhost:5432)
Bancos de Dados (3)
dbgeo
postgres
template_postgis
Tablespaces (2)
Roles do Grupo (0)
Login Roles (1)
Propriedades
Estatísticas
Dependências
Dependentes
Propriedade
Valor
Descrição
PostgreSQL Database Server 8.3
Nome da Máquina
localhost
Porta
5432
Serviço
pgsql-8.3
Manutenção do ban...
postgres
Nome de usuário
postgres
Armazenar senha?
Sim
Restaurar ambiente?
Não
Painel SQL
Estabelecendo conexão...Concluído
0.25 seg.

Cliente/Servidor

- Gerencia os arquivos dos bancos de dados
- Aceita conexões com as aplicações clientes
- Realiza operações nos bd em nome dos clientes

Spatial Database Systems

Gilberto Ribeiro de Queiroz
(gribeiro@dpi.inpe.br)

Roadmap

- What is a Spatial Database System (SDS)?
- Spatial Data Types and Operators
- What is a Multi-dimensional Access Method (Spatial Indexing)?
- GIS and DBMS integration
- Standards for SDS
- Historical Development of Spatial Database Systems

What is a Spatial Database System?

Which applications have to manage data related to space?

Geographic Information Systems (GIS)

Source: Wikipedia

Electronic Design Automation (EDA or ECAD)

Source: Wikipedia

3D Macromolecular Structures

Source: MMDB

3D Magnetic Resonance Imaging (MRI)

Source: 3DSlicer

Database Technology Evolution

Database Technology Evolution

Emergent and demanding applications: CAD, **GIS**, Multimedia, OLAP, Real-time, Scientific

Several Research Prototypes in the late 80's: Probe, DASDBS GEO-Kernel, Gral, SIRO-DBMS, Starburst, Geo++, GéoSabrina, GODOT, $GeoO_2$, Paradise

Is there a unified term for SDS?

- Common terms in the literature:
  - Pictorial
  - Image
  - Geometric
  - Geographic
  - Spatial database system

raster images: such as remote sensing satellite data or computer tomography

set of objects in space

- Most distinguished:
  - Image database systems
  - Spatial database systems

We will have specific classes about this topic

This will be our focus for the next classes

A Spatial Database System is…

Database System

geometric information may be connected to non-spatial data:
"who is the owner of a given parcel?"

+

Spatial Data Types

The model and query language must support spatial types such as points, lines, or polygons and their operations (intersection, area, ...)

+

Spatial Indexing

The system should avoid scanning the whole set of objects when executing a given query or performing a spatial join

A Spatial Database System is not a full GIS

A SDS should provide the underlying database technology for a GIS

GIS is the driven technology of today SDS

Spatial Data Types

Spatial Data Types

Single Objects

Point

- A point represents an object for which only its location in space is relevant:
  - its extent is not important for modeling.
  - Zero dimensional objects.

- Points may be embedded in:
  - $\Re^2$ (2D space):
    - (x, y)
  - $\Re^3$ (3D Space)
    - (x, y, z)
    - (x, y, z, m) obs.: m = measure

Line

- Used for representing a curve in space:
  - A sequence of coordinates
  - Linear objects or 1-dimensional
  - Roads, rivers, cables, electricity lines

- Lines may be embedded in:
  - $\Re^2$ (2D space):
    - $\{ (x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$
  - $\Re^3$ (3D Space)
  - $\Re^3$ (3D Space) + Measure

- Segment interpolation:
  - Linear interpolation between nodes
  - Curved segments

Polygon

- A polygon (or region) is used to represent something having an extent and area:
  - One **outer ring** and zero or more **inner rings**.
  - Surface objects or 2-dimensional.
  - A polygon is a region bounded by a closed line (called ring).

Simple polygon
(no self intersection)

Polygon without holes
(non-disconnected exterior)

Polygon with holes
(disconnected exterior)

A region with several disjoint pieces

Curved and Compound Polygon

Spatial Data Types

How to represent a collection of spatial objects?

Partitions of the plane

MG Cities

MG Crop Lands

No overlap of regions.
Specially interset: adjacency relationship.
Useful for thematic maps.

Networks

A graph embedded in the plane.
Set of points forming the nodes and a set of lines forming the edges.

Internal Representation

- Single objects (points, lines, polygons):
  - Data structures for sequences: vector or lists;
  - Orientation: clockwise or counter-clockwise;
  - Ordering: x or y.

- Representating a collection of objects:

Spaghetti Model

Topological Model

Network Model

Spatial Data Types

Spatial Algebras

Spatial Algebras

- When introducing spatial types into a DBMS it is necessary to define a spatial algebra:

$\forall p$ *in polygon*

p	$\times$	p	$\rightarrow$	*bool*	intersects, disjoint, touches, overlaps
p	$\times$	p	$\rightarrow$	*geometry*	intersection
p			$\rightarrow$	[*lines*]	boundary
p	$\times$	p	$\rightarrow$	*real*	distance
p			$\rightarrow$	*real*	area, perimeter

Abstract Data Types (ADT):

- ✓ combination of data types and their operations;
- ✓ Accessible only through operations defined on them (encapsulation).

Spatial Operations

Geometric Operators

Metric Operators

Spatial Relationships

Geometric Operators

Topological Transforms: rotation, translation, scale change, symmetry.

Dimensional Transforms: boundary.

Extraction: MBR, centroid.

Object Properties: is_convex, is_connected, is_simple.

Metric Operators

- Length
- Perimeter
- Area
- Distance

Spatial Relationships

- Topological relationships:

- Direction relationships:
  - Above, below, north_of, …

- Metric relationships:
  - Distance between two objects are less than a given number of units.

Spatial Relationships

Source: Egenhofer and Herring (1990)

Spatial Relationships in Experimental Query Languages	
Query Language	**Spatial Relationships**
Freeman (1975)	left of, right of, beside, above, below, near, far, touching, between, inside, outside
ATLAS (Tsurutani et al., 1980)	area adjacency, line adjacency, boundary relation- ship, containment, distance, direction
MAPQUERY (Frank, 1982)	on, adjacent, within
KBGIS (Smith and Pazner, 1984)	containment, subset, neighborhood, near, far, north, south, east, west
KGIS (Ingram and Phillips, 1987)	distance, overlay, adjacent, overlap
PSQL (Roussopoulos et al., 1988)	covering, coveredBy, overlapping, disjoint, near- est, furthest, within, outside, on perimeter
SQL extension (Herring et al., 1988)	adjacent, contains, contains point, enclosed by, intersect, near, self intersect
Geo-Relational Algebra (Güting, 1988)	equal, not equal, inside, outside, intersect
Spatial SQL (Egenhofer, 1989)	disjoint, equal, meet, overlap, concur, commonBounds

A Framework for the Description of Topological Spatial Relations

$$
\begin{array}{c c}
 & \begin{array}{cc} \mathbf{B^o} & \boldsymbol{\delta B} \end{array} \\
\begin{array}{c} \mathbf{A^o} \\ \\ \boldsymbol{\delta A} \end{array} & \begin{bmatrix} A^o \cap B^o & A^o \cap \delta B \\ \\ \delta A \cap B^o & \delta A \cap \delta B \end{bmatrix}
\end{array}
$$

4-intersection Matrix
(Egenhofer and Franzosa, 1991)

4-Intersection Matrix

$$\begin{array}{c|cc} & \partial B & B^\circ \\ \hline \partial A & \varnothing & \varnothing \\ A^\circ & \varnothing & \varnothing \end{array}$$

disjoint

$$\begin{array}{c|cc} & \partial B & B^\circ \\ \hline \partial A & \neg\varnothing & \varnothing \\ A^\circ & \varnothing & \varnothing \end{array}$$

meet

$$\begin{array}{c|cc} & \partial B & B^\circ \\ \hline \partial A & \varnothing & \varnothing \\ A^\circ & \neg\varnothing & \neg\varnothing \end{array}$$

contains

$$\begin{array}{c|cc} & \partial B & B^\circ \\ \hline \partial A & \neg\varnothing & \varnothing \\ A^\circ & \neg\varnothing & \neg\varnothing \end{array}$$

covers

$$\begin{array}{c|cc} & \partial B & B^\circ \\ \hline \partial A & \neg\varnothing & \varnothing \\ A^\circ & \varnothing & \neg\varnothing \end{array}$$

equal

$$\begin{array}{c|cc} & \partial B & B^\circ \\ \hline \partial A & \neg\varnothing & \neg\varnothing \\ A^\circ & \neg\varnothing & \neg\varnothing \end{array}$$

overlap

$$\begin{array}{c|cc} & \partial B & B^\circ \\ \hline \partial A & \varnothing & \neg\varnothing \\ A^\circ & \varnothing & \neg\varnothing \end{array}$$

inside

$$\begin{array}{c|cc} & \partial B & B^\circ \\ \hline \partial A & \neg\varnothing & \neg\varnothing \\ A^\circ & \varnothing & \neg\varnothing \end{array}$$

covered by

Refinements on Intersection Matrix

$$
\begin{array}{c}
 \\ A^o \\ \delta A \\ A^-
\end{array}
\begin{array}{c}
\begin{array}{ccc} B^o & \delta B & B^- \end{array} \\
\begin{bmatrix}
A^o \cap B^o & A^o \cap \delta B & A^o \cap B^- \\
\delta A \cap B^o & \delta A \cap \delta B & \delta A \cap B^- \\
A^- \cap B^o & A^- \cap \delta B & A^- \cap B^-
\end{bmatrix}
\end{array}
$$

9-intersection Matrix
(Egenhofer, 1991)

$$
\begin{array}{c}
 \\ A^o \\ \delta A \\ A^-
\end{array}
\begin{array}{c}
\begin{array}{ccc} B^o & \delta B & B^- \end{array} \\
\begin{bmatrix}
\dim(A^o \cap B^o) & \dim(A^o \cap \delta B) & \dim(A^o \cap B^-) \\
\dim(\delta A \cap B^o) & \dim(\delta A \cap \delta B) & \dim(\delta A \cap B^-) \\
\dim(A^- \cap B^o) & \dim(A^- \cap \delta B) & \dim(A^- \cap B^-)
\end{bmatrix}
\end{array}
$$

Dimension Extended 9-intersection Matrix
(Clementini et al. , 1993)

Operações Topológicas

- Qual é o relacionamento espacial entre os objetos A e B?

$$\begin{array}{c|ccc} & B^\circ & \partial B & B^- \\ \hline A^\circ & 2 & 1 & 2 \\ \partial A & 1 & 0 & 1 \\ A^- & 2 & 1 & 2 \end{array}$$

$$\begin{array}{c|ccc} & B^\circ & \partial B & B^- \\ \hline A^\circ & F & F & 2 \\ \partial A & F & 1 & 1 \\ A^- & 2 & 1 & 2 \end{array}$$

```
ST_Relate(A, B)
```

Operations on Collections of Objects

- Voronoi diagram
- Overlay:
  - Generates a new set based on two input sets.
- Aggregate:
  - fusion/merging of adjacent areas if an attributes are equal.
- Merge:
  - Union of two or more sets with the same schema.
- Closest:
  - Find the objects closest to a query object.

Implementation Issues

Geometric Algorithms

How to efficiently compute the intersection matrix?

- Segment intersection?
- Point in polygon tests?
- Temporary data structures used to represent the objects boundary?

$$
\begin{array}{c|ccc}
 & B^\circ & \partial B & B^- \\
\hline
A^\circ & 2 & 1 & 2 \\
\partial A & 1 & 0 & 1 \\
A^- & 2 & 1 & 2
\end{array}
$$

Spatial Access Methods

Let’s see some typical search operations in a GIS data relies on spatial indexing…

Point Queries

- **Definition:** Given a point in the space, find all objects that contain it.
- Also known as “stabbing query”.

Region Queries

- **Definition:** Given a region (or query window), find all objects that intersect it.
- Also known as "range queries"

Nearest Neighbors

- **Definition:** Given a point or object, find its nearest neighbor(s).
- **Variations:** fixed distance, fixed number of neighbors, closest object

Spatial Joins

- **Definition:** Given two sets of spatial objects S_1 and S_2, and a spatial relationship θ, find the pairs $(s_1, s_2) \in S_1 \times S_2$ which geometry matches θ.
- Example:

S_1 = focos
S_2 = trechos rodoviários
θ = focos contidos num raio de 1000 metros de um dado trecho

Spatial Access Methods

kd-tree, Quad-tree, Grid, R-Tree

k-d-tree

(Bentley, 1975)

Quad-trees

(Finkel and Bentley, 1974; Samet, 1990)

Fixed-Grid

Associações:
Célula(0,0): O4
Célula(0,1): O4
Célula(0,2): O1
Célula(0,3): O1
Célula(0,4): O2
Célula(1,0): O4
Célula(1,1): O4
Célula(1,2): O1
Célula(1,3): O1
Célula(2,1): O5
Célula(2,2): O5
Célula(3,1): O5
Célula(3,2): O5
Célula(3,3): O3
Célula(3,4): O3
...
Célula(9,5): O7

R-tree

(Guttman, 1984)

Spatial Query Processing

Spatial Selections

Spatial Joins

Spatial Query
(Kriegel et al., 1993)

Spatial Joins

(Brinkhoff et al., 1994)

Spatial Query Languages

Spatial Databases Standards

OGC SFS

ISO SQL-MM Spatial

GIS and DBMS

GIS and DBMS
(Before the advent of SDS)

GIS and DBMS
(Advent of SDS)

References

Introductory

- Ralf Hartmut Güting. 1994. ***An introduction to spatial database systems***. The VLDB Journal, v. 3, n. 4, 357-399, October 1994.

- Philippe Rigaux, Michel Scholl and Agnès Voisard. 2001. ***Spatial Databases with Application to GIS***. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Surveys

- Volker Gaede and Oliver Günther. 1998. **Multidimensional access methods**. *ACM Computing Surveys, v.* 30, n. 2, June 1998, pp. 170-231.

- Hanan Samet. 1990. **The Design and Analysis of Spatial Data Structures**. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.

Topological Relations

- M. Egenhofer and J. Herring. ***Categorizing Binary Topological Relations Between Regions, Lines, and Points in Geographic Databases***. Technical Report, Department of Surveying Engineering, University of Maine, 1990.

- M. Egenhofer and R. Franzosa. ***Point-Set Topological Spatial Relations***. International Journal of Geographical Information Systems 5 (2): 161-174, 1991.

- M. Egenhofer. ***Reasoning about Binary Topological Relations***. Second Symposium on Large Spatial Databases, Zurich, Switzerland. O. Gunther and H.-J. Schek (eds.), Lecture Notes in Computer Science, Vol. 525, Springer-Verlag, pp. 143-160, August 1991.

- Eliseo Clementini, Paolino Di Felice, and Peter van Oosterom. 1993. ***A Small Set of Formal Topological Relationships Suitable for End-User Interaction***. In *Proceedings of the Third International Symposium on Advances in Spatial Databases* (SSD '93), David J. Abel and Beng Chin Ooi (Eds.). Springer-Verlag, London, UK, UK, 277-295.

Spatial Access Methods

- Jon Louis Bentley. 1975. ***Multidimensional binary search trees used for associative searching***. *Communications of the ACM,* v. 18, n. 9, September 1975, pp. 509-517.

- R. A. Finkel and J. L. Bentley. ***Quad trees: A data structure for retrieval on composite keys***. *Acta Informatica*, 4(1):1-9, 1974.

- Antonin Guttman. 1984. ***R-trees: a dynamic index structure for spatial searching***. In *Proceedings of the 1984 ACM SIGMOD international conference on Management of data* (SIGMOD '84). ACM, New York, NY, USA, 47-57.

Spatial Query Processing

- Kriegel H.-P., Brinkhoff T., Schneider R. ***Efficient Spatial Query Processing in Geographic Database Systems***. IEEE Data Engineering Bulletin, Vol. 16, No. 3, 1993, pp. 10-15.

- Thomas Brinkhoff, Hans-Peter Kriegel, Ralf Schneider, and Bernhard Seeger. 1994. ***Multi-step processing of spatial joins***. *SIGMOD Rec.* 23, 2 (May 1994), 197-208.

Spatial Query Processing

- Oliver Günther. 1993. ***Efficient Computation of Spatial Joins***. In *Proceedings of the Ninth International Conference on Data Engineering*. IEEE Computer Society, Washington, DC, USA, pp. 50-59.

MINISTÉRIO DA CIÊNCIA E TECNOLOGIA
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

SGBDs Espaciais

Lubia Vinhas

Arquiteturas para aplicações geográficas: dual

- SGBD relacional: armazena dados alfanuméricos
- Arquivos: armazenam dados espaciais

Arquitetura Dual

TABELA ARMAZENADA NO SGBD

CODPAR	CODFAZ	CODPROJ	TALHAO	AREATOT	DTPLAN
5	205017	105	*147*	9,91	31/10/89
5	205017	105	*148*	25,66	18/12/89
5	205017	068	*152*	26,34	7/10/93
5	205017	068	*153*	21,65	14/10/93
5	205017	068	*154*	27,90	21/10/93
5	205017	068	*155*	23,52	23/11/93
5	205017	109	*162*	26,29	5/11/89
5	205017	109	*163*	27,57	9/11/89

CODPAR = código do parque florestal ; CODFAZ = código da fazenda

CODPROJ = código do projeto; TALHÃO = número do talhão

AREATOT = Área total plantada; DTPLAN = Data do plantio

Arquitetura dual: exemplos

- SPRING
  - Dados alfanuméricos: SGBD relacional (DBase, Access, MySQL, Oracle, PostgreSQL)
  - Dados espaciais: arquivos com formato específico
- ArcView
  - Dados alfanuméricos: DBF relacional
  - Dados espaciais: “shapefiles”
- IDRISI
  - Dados alfanuméricos: SGBD relacional
  - Dados espaciais: matrizes

Arquitetura dual: ex. SPRING

Dados de controle que garantem o modelo conceitual

Arquitetura dual: ex. SPRING

Arquitetura dual: ex. ArcView

Shapefile: Estados.shp, Estados.shx, Estados.dbf

Shape	Id	Name1	Name2	Parts	Points	Length
Polygon	35503031	DI31	Guaianazes	1.00000	193.00000	
Polygon	35503093	DI93	Vila Matilde	1.00000	170.00000	
Polygon	35503088	DI88	Vila Guilherme	1.00000	54.00000	
Polygon	35503008	DI08	Belem	1.00000	62.00000	
Polygon	35503010	DI10	Bras	1.00000	86.00000	
Polygon	35503057	DI57	Pari	1.00000	56.00000	
Polygon	35503082	DI82	Tatuape	1.00000	101.00000	
Polygon	35503091	DI91	Vila Maria	1.00000	97.00000	
Polygon	35503060	DI60	Penha	1.00000	115.00000	
Polygon	35503038	DI38	Jacana	1.00000	139.00000	
Polygon	35503094	DI94	Vila Medeiros	1.00000	91.00000	

Como consultar a base? Ex. apontamento

	ID	NOME	ROTULO	AREA	PERIMETRO	NOMEST	UF	REGIAO
19	19	33	33	42120876032.000000	1581134.375000	Rio de J...	RJ	SE
20	20	35	35	247546937344.000000	2912832.500000	São Paulo	SP	SE
21	21	41	41	196600905728.000000	2331258.500000	Paraná	PR	SU
22	22	42	42	95387869184.000000	2012606.500000	Santa C...	SC	SU
23	23	43	43	271939272704.000000	3233131.500000	Rio Gra...	RS	SU
24	24	50	50	361756459008.000000	2783440.250000	Mato Gr...	MS	CO
25	25	51	51	898282422272.000000	4755015.000000	Mato Gr...	MT	CO
26	26	52	52	338731171840.000000	3100251.000000	Goiás	GO	CO
27	27	53	53	6123834368.000000	312854.906250	Distrito ...	DF	CO

1. Capture o ponto da tela
2. Transforme em coordenadas do mundo
3. <u>Busque nos arquivos qual geometria contém esse ponto</u>
4. Recupere o ID do objeto associado a geometria
5. Envie consulta ao SGDB:

   SELECT * FROM estados

   where ID = xxx;
6. Acenda a linha da grade correspondente ao resultado da consulta

Feito pelo SIG

Feito pelo SGBD

Arquitetura Dual

- Vantagens
  - Acesso aos atributos via SQL “padrão”
  - Acesso a ferramentas do SGBD (ex. formulários e relatórios)
  - Não exige nenhuma capacidade a mais do SGBD
- Desvantagens
  - Falta de interoperabilidade
  - Dificuldade em manter a integridade entre os dados espaciais e atributos
  - Dificulta a manutenção de um ambiente multiusuário
  - Métodos de acesso e otimizados de consulta devem ser implementados pelo SIG

Arquitetura Dual

- Características:
  - Apropriada para SIG “desktop” em um ambiente monousuário
    - Cada usuário gerencia seus dados
  - Compartilhamento de dados
    - Exige duplicação dos arquivos de dados
  - Atualização da informação
    - Requer nova cópia para todos os usuários

Arquitetura Integrada

- Usa o SGBD: para armazenar todo o dado geográfico
- Duas alternativas:
  - SGBD relacional
  - SGBD objeto-relacional

Arquitetura integrada: SGBD relacional

- **Modelo relacional**
  - Banco de dados é organizado em uma coleção de relações ou tabelas relacionadas entre si

Aluno

MATRICULA	NOME	CURSOID
98765	João	MAT
67765	José	BIO
84562	Maria	ENG
34256	Luis	INFO
3452672	Ana	MAT
34529	Luana	MAT

Curso

CURSOID	TITULO	DURAÇÃO
INFO	Informática Indust.	4
BIO	Biologia	4
ENG	Engenharia Civil	5
MAT	Licenciatura Mat.	4

Como armazenar um polígono em um SGBD-R?

- Alternativa 1: tabelas de pontos (x,y)

Pol1

Pol2

Poligonos

id	num_coords	num_holes
pol1	4	0
pol2	12	1

Pts_pol

id	pt
pol1	1
pol1	2
pol1	3
pol1	4
pol1	5

Pontos

id	x	y
1	1	5
2	5	5
3	5	1
4	1	1
5	1	5

Como armazenar um polígono em um SGBD-R?

- Alternativa 2: campo longo binário (BLOB), uma sequência de bits sem significado para o banco

Pol1

Pol2

Tipo BLOB

Poligonos

id	num_coords	num_holes	poligono
pol1	4	0	(xy, xy, xy...)
pol2	12	1	(xy, xy, xy...)

Arquitetura Integrada - SGBD Relacional

- Vantagens
  - Facilidade na manutenção de consistência entre a componente espacial e alfanumérica
  - Uso dos recursos do SGBD também para a parte espacial:
    - transação, recuperação de falhas, controle de acesso concorrente, etc.
- Desvantagens
  - Falta de semântica dos dados espaciais
  - Sujeita as limitações da SQL para manipular BLOBs
  - Métodos de acesso e otimizados de consulta devem ser implementados pelo SIG

Processamento de consultas espaciais

- Por quais municípios brasileiros passa o trecho da BR-040 destacado?

Aproximação pelo ***Retângulo Envolvente Mínimo***
(REM ou Bounding Box ou MBR)

Processamento de consultas espaciais

Tipos de SGBD

Rede

Hierárquico

Objeto

P-Id	Nome	Sobrenome	Cidade
1	Lais	Costa	SJC
2	Maria	Silva	SP

Cidade	População	Renda
SJC	1000000	32244
...	...	...

Relacional

Tipos de SGBD

P-Id	Nome	Sobrenome	Cidade
1	Lais	Costa	SJC
...	...	...	...

Cidade	População	Renda
SJC	1000000	32244
...	...	...

Relacional

Objeto

ID	XY	DF	ER
56		XXX	
45		YYY	
...	...	...	...

Objeto-Relacional

SGBD Espacial

- A visão de Gutting (1994):
  1. Um SGBD espacial é um SGBD
  2. Oferece tipos de dados espaciais em seu modelo de dados e em sua linguagem de consulta
  3. Suporta tipos de dados espaciais em sua implementação, fornecendo algoritmos de indexação espacial e junção espacial eficientes
- Extensões espaciais implementam um SGBD espacial utilizando o modelo objeto relacional para estender suas capacidades para tratar dados espaciais (complexos)
  - PostGIS, Oracle Spatial, Spatialite, MySQL, IBM DB2 Spatial...
- É desejável que haja interoperabiliade entre eles

Tipos Espaciais - OGC

Operações sobre geometrias

- Olhando para a representação OO do tipo geometry é possível ter uma ideia das operações espaciais possíveis para um tipo espacial

Geometry

```
+ dimension() : Integer
+ coordinateDimension() : Integer
+ spatialDimension() : Integer
+ geometryType() : String
+ SRID() : Integer
+ envelope() : Geometry
+ asText() : String
+ asBinary() : Binary
+ isEmpty() : Boolean
+ isSimple() : Boolean
+ is3D() : Boolean
+ isMeasured()() : Boolean
+ boundary() : Geometry
query
+ equals(another :Geometry) : Boolean
+ disjoint(another :Geometry) : Boolean
+ intersects(another :Geometry) : Boolean
+ touches(another :Geometry) : Boolean
+ crosses(another :Geometry) : Boolean
+ within(another :Geometry) : Boolean
+ contains(another :Geometry) : Boolean
+ overlaps(another :Geometry) : Boolean
+ relate(another :Geometry, matrix :String) : Boolean
+ locateAlong(mValue :Double) : Geometry
+ locateBetween(mStart :Double, mEnd :Double) : Geometry
analysis
+ distance(another :Geometry) : Distance
+ buffer(distance :Distance) : Geometry
+ convexHull() : Geometry
+ intersection(another :Geometry) : Geometry
+ union(another :Geometry) : Geometry
+ difference(another :Geometry) : Geometry
+ symDifference(another :Geometry) : Geometry
```

Métodos de acesso a estrutura e a definição do tipo. WKB, WKT e código para o tipo devem seguir o padrão OGC.

Geometry

- + dimension() : Integer
- + coordinateDimension() : Integer
- + spatialDimension() : Integer
- + geometryType() : String
- + SRID() : Integer
- + envelope() : Geometry
- + asText() : String
- + asBinary() : Binary
- + isEmpty() : Boolean
- + isSimple() : Boolean
- + is3D() : Boolean
- + isMeasured()() : Boolean
- + boundary() : Geometry

query

- + equals(another :Geometry) : Boolean
- + disjoint(another :Geometry) : Boolean
- + intersects(another :Geometry) : Boolean
- + touches(another :Geometry) : Boolean
- + crosses(another :Geometry) : Boolean
- + within(another :Geometry) : Boolean
- + contains(another :Geometry) : Boolean
- + overlaps(another :Geometry) : Boolean
- + relate(another :Geometry, matrix :String) : Boolean
- + locateAlong(mValue :Double) : Geometry
- + locateBetween(mStart :Double, mEnd :Double) : Geometry

analysis

- + distance(another :Geometry) : Distance
- + buffer(distance :Distance) : Geometry
- + convexHull() : Geometry
- + intersection(another :Geometry) : Geometry
- + union(another :Geometry) : Geometry
- + difference(another :Geometry) : Geometry
- + symDifference(another :Geometry) : Geometry

Operações sobre geometrias

Operações sobre geometrias

Predicados espaciais que são usados na construção de junções espaciais. OGC propõe que eles sejam baseados na matriz de 9 interseções extendida de Egenhofer

Geometry

- + dimension() : Integer
- + coordinateDimension() : Integer
- + spatialDimension() : Integer
- + geometryType() : String
- + SRID() : Integer
- + envelope() : Geometry
- + asText() : String
- + asBinary() : Binary
- + isEmpty() : Boolean
- + isSimple() : Boolean
- + is3D() : Boolean
- + isMeasured()() : Boolean
- + boundary() : Geometry

query

- + equals(another :Geometry) : Boolean
- + disjoint(another :Geometry) : Boolean
- + intersects(another :Geometry) : Boolean
- + touches(another :Geometry) : Boolean
- + crosses(another :Geometry) : Boolean
- + within(another :Geometry) : Boolean
- + contains(another :Geometry) : Boolean
- + overlaps(another :Geometry) : Boolean
- + relate(another :Geometry, matrix :String) : Boolean
- + locateAlong(mValue :Double) : Geometry
- + locateBetween(mStart :Double, mEnd :Double) : Geometry

analysis

- + distance(another :Geometry) : Distance
- + buffer(distance :Distance) : Geometry
- + convexHull() : Geometry
- + intersection(another :Geometry) : Geometry
- + union(another :Geometry) : Geometry
- + difference(another :Geometry) : Geometry
- + symDifference(another :Geometry) : Geometry

Operadores baseados na DE-9IM

- Dados um objeto *a*, sejam *I(a)*, *B(a)* e *E(a)* o seu interior, borda e exterior respectivamente.
  - Seja *dim(x)* a dimensão máxima (-1, 0, 1 ou 2) dos objetos geométricos em *x*, com o valor -1 correspondendo a *dim(∅)*.
  - A intersecção de quaisquer dois *I(a)*, *B(a)* e *E(a)* pode resultar em um conjunto de objetos geométricos de dimensão mista.
  - Por exemplo, a intersecção de das bordas de dois polígonos pode resultar em um ponto e uma linha.

Table 1: The DE-9IM

	Interior	Boundary	Exterior
Interior	$dim(I(a) \cap I(b))$	$dim(I(a) \cap B(b))$	$dim(I(a) \cap E(b))$
Boundary	$dim(B(a) \cap I(b))$	$dim(B(a) \cap B(b))$	$dim(B(a) \cap E(b))$
Exterior	$dim(E(a) \cap I(b))$	$dim(E(a) \cap B(b))$	$dim(E(a) \cap E(b))$

Operadores baseados na DE-9IM

- Exemplo de uma instância da DE-9IM para o caso de dois polígonos que se sobrepõe

	Interior	Boundary	Exterior
Interior	2	1	2
Boundary	1	0	1
Exterior	2	1	2

Operadores baseados na DE-9IM

- São nomeados cinco predicados baseados na DE-9IM
  1. Disjoint
  2. Touches
  3. Crosses
  4. Within
  5. Overlaps
- Detalhe do <u>Disjoint</u>

```
a.Disjoint(b) ⇔ [ (I(a) ∩ I(b) = ∅) ∧
                  (I(a) ∩ B(b) = ∅) ∧
                  (B(a) ∩ I(b) = ∅) ∧
                  (B(a) ∩ B(b) = ∅) ]
              ⇔  a.Relate(b, "FF*FF****")
```

Operações sobre geometrias

Geometry

- + dimension() : Integer
- + coordinateDimension() : Integer
- + spatialDimension() : Integer
- + geometryType() : String
- + SRID() : Integer
- + envelope() : Geometry
- + asText() : String
- + asBinary() : Binary
- + isEmpty() : Boolean
- + isSimple() : Boolean
- + is3D() : Boolean
- + isMeasured()() : Boolean
- + boundary() : Geometry

query

- + equals(another :Geometry) : Boolean
- + disjoint(another :Geometry) : Boolean
- + intersects(another :Geometry) : Boolean
- + touches(another :Geometry) : Boolean
- + crosses(another :Geometry) : Boolean
- + within(another :Geometry) : Boolean
- + contains(another :Geometry) : Boolean
- + overlaps(another :Geometry) : Boolean
- + relate(another :Geometry, matrix :String) : Boolean
- + locateAlong(mValue :Double) : Geometry
- + locateBetween(mStart :Double, mEnd :Double) : Geometry

analysis

- + distance(another :Geometry) : Distance
- + buffer(distance :Distance) : Geometry
- + convexHull() : Geometry
- + intersection(another :Geometry) : Geometry
- + union(another :Geometry) : Geometry
- + difference(another :Geometry) : Geometry
- + symDifference(another :Geometry) : Geometry

Usam predicados espaciais e medidas canônicas sobre geometrias para gerar outras geometrias

Exemplo de organização e consultas

- Camadas: tabelas do modelo relacional

  Municipios[id:text, nome:text, populacao:int, geometria:geo, uf:text]

  Estados[id:text, nome:text, geometria: geo]

  Ferrovias[nome:text, administracao:text, geometria:geo]

Exemplo de organização e consultas

- Camadas: tabelas do modelo relacional

  Municipios[id:text, nome:text, populacao:int, geometria:geo, uf:text]

  Estados[id:text, nome:text]

  Ferrovias[nome:text, administracao:text, geometria:geo]

- Consulta não espacial: número de habitantes em SJC

```
SELECT  populacao
FROM    Municipios
WHERE  nome = 'SJC'
```

Exemplo de organização e consultas

- Camadas: tabelas do modelo relacional

  Municipios[id:text, nome:text, populacao:int, geometria:geo, uf:text]

  Estados[id:text, nome:text]

  Ferrovias[nome:text, administracao:text, geometria:geo]

- Consulta não espacial: número de habitantes no estado de SP

```
SELECT  SUM(populacao)
FROM    Municipios, Estados
WHERE   Estado.nome = 'SP'
        AND Municipio.uf = Estado.id
```

Exemplo de organização e consultas

- Camadas: tabelas do modelo relacional

  Municipios[id:text, nome:text, populacao:int, geometria:geo, uf:text]

  Estados[id:text, nome:text]

  Ferrovias[nome:text, administracao:text, geometria:geo]

- Consulta espacial: desenhar o estado de SP

```
SELECT  Municipios.geometria
FROM    Municipios, Estados
WHERE   Estado.nome = 'SP'
        AND Municipio.uf = Estado.id
```

Exemplo de organização e consultas

- Camadas: tabelas do modelo relacional

  Municipios[id:text, nome:text, populacao:int, geometria:geo, uf:text]

  Estados[id:text, nome:text, geometria: geo]

  Ferrovias[nome:text, administracao:text, geometria:geo]

- Consulta espacial: municípios adjacentes a SJC

```
SELECT  Municipios.nome
FROM    Municipios m1, Municipios m2
WHERE   m1.nome = 'SJC'
AND     TOUCHES(m1.geometria,m2.geometria)
```

Exemplo de organização e consultas

- Camadas: tabelas do modelo relacional

  Municipios[id:text, nome:text, populacao:int, geometria:geo, uf:text]

  Estados[id:text, nome:text, geometria: geo]

  Ferrovias[nome:text, administracao:text, geometria:geo]

- Consulta espacial: municipios e suas áreas

```
SELECT  Municipios.nome, AREA(Municipios.geometria)
FROM    Municipios
```

- Consulta espacial: área do estado de SP

```
SELECT  Municipios.nome, SUM(AREA(Municipios.geometria))
FROM    Municipios, Estados
WHERE Municipos.uf = Estados.id AND
```

Exemplo de organização e consultas

- Camadas: tabelas do modelo relacional

  Municipios[id:text, nome:text, populacao:int, geometria:geo, uf:text]

  Estados[id:text, nome:text, geometria: geo]

  Ferrovias[nome:text, administracao:text, geometria:geo]

- Consulta espacial: trechos de rodovias que passam por SJC

```
SELECT  INTERSECTION(Ferrovias.geo,Municipios.geo)
FROM    Municipios, Ferrovias
WHERE  Municipios.nome = 'SJC'
```

A IMPLEMENTAÇÃO DE UM BANCO DE DADOS GEOGRÁFICO BASEADO EM SQL

Standards relevantes

OGC – Simple Feature Access SFA

- Também chamada ISO 19125, define um esquema SQL padrão que suporta o armazenamento, recuperação, consulta e atualização de coleções de feições (objetos) geográficas via SQL
- Coleção de feições ➔ *Feature Table*
- Cada feição ➔ uma linha de uma *feature table*
- Atributos não espaciais ➔ colunas de tipos básicos
- Atributos espaciais ➔ colunas de tipos geométricos

Esquema para feature tables com tipos geométricos

[OGC, 2006]

Tipos Espaciais

OGC SFA

ISO SQL/MM

Categorias de métodos

1. Conversão entre geometrias e formatos externos
2. Recuperação de propriedades e medidas a partir de uma geometria
3. Comparação de duas geometrias com respeito a seus relacionamentos espaciais
4. Criação de novos geometrias a partir de uma existente

Categorias de métodos

1. Conversão entre geometrias e formatos externos
   a. WKT
   b. WKB
   c. Geographic markup language – GML

- Cada tipo implementa métodos que permite a sua criação, dado um WKT ou WKB, opcionalmente fornecendo o SRID
- ST_AsText, ST_AsBinary e ST_AsGML : para conversão de geometrias e seus respectivos formatos externos

Categorias de métodos

2. Recuperação de propriedades. Exemplos
   - ST_Boundary
   - ST_IsValid
   - ST_IsEmpty
   - ST_X
   - ST_IsRing
   - ST_Length

Categorias de métodos

3. Comparação de duas geometrias. Exemplos
   - ST_Equals
   - ST_Disjoint
   - ST_Intersects, ST_Crosses, ST_Overlaps
   - ST_Touches
   - ST_Within, ST_Contains
   - ST_Length

Categorias de métodos

4. Geração de novas geometrias. Exemplos
   - ST_Buffer
   - ST_ConvexHull
   - ST_Difference, ST_Intersection, ST_Union

Como obter eficiência em banco de dados?

INDEXAÇÃO

Métodos de acesso

- **Métodos de acesso** são os procedimentos empregados pelo SGBD com o objetivo de acelerar a localização e a recuperação de algum dado

Ex: Busca Sequencial

- Segue cada registro uma a um comparando com o critério de seleção

  Quais são as informações referentes ao aluno de matrícula **100203**?

	Tabela: Alunos			
Nº Reg	Matricula	Nome	Telefone	Data Matrícula
1	9765421	Gilberto	11-2345	1969
2	8763524	Antônio	12-3456	2000
	1000203	Maria	10-0987	2004
	6827265	José	08-9865	2005
5	2039457	João	23-8754	2002
6	1000204	Eduardo	23-5677	1998
7	8888823	Ana	55-5676	1997
8	5678999	Carolina	58-5676	1943
...	...	...	...	
100000	1243547	Cássia	56-5676	2000

Busca sequencial: 3 comparações

Busca sequencial

Quais são as informações referentes ao aluno de matrícula 1243547?

	Tabela: Alunos			
Nº Reg	Matricula	Nome	Telefone	Data Matrícula
1	9765421	Gilberto	11-2345	1969
2	8763524	Antônio	12-3456	2000
3	1000203	Maria	10-0987	2004
4	6827265	José	08-9865	2005
5	2039457	João	23-8754	2002
6	1000204	Eduardo	23-5677	1998
7	8888823	Ana	55-5676	1997
8	5678999	Carolina	58-5676	1943
...	...	...	...	
100000	1243547	Cássia	56-5676	2000

Busca sequencial: 100000 de comparações

Métodos de acesso

- Em geral, uma consulta envolve apenas uma pequena parcela do BD
- Percorrer todo o BD procurando pelos dados relevantes para a consulta é em geral muito ineficiente
- Um método de acesso estabelece um plano de execução para a consulta podendo usar estruturas auxiliares
- **Índices** são estruturas de dados que ajudam a melhorar a velocidade de execução de operações de consulta sobre um conjunto de dados
  - Existem diferentes tipos de índices, cada qual adequado a um tipo de atributo ou situação. Muitos são baseados em árvores.
  - Em geral a criação de índices é opcional, ficando a cargo do administrador do banco de dados
  - Existe um custo associado a um índice, uma vez que ele é uma estrutura adicional aos dados em si

Árvores

- Estruturas de indexação para dados convencionais, em geral, baseiam-se em estruturas de dados do tipo *árvore*

Grau: número máximo de filhos que um nó pode ter

Árvores Balanceadas

Árvores Binárias

Árvores binárias

Quais os dados do aluno cuja matrícula é 2039457?

Árvores binárias de pesquisa (BST)

- Grau máximo de um nó: 2
- Árvores balanceadas mais comuns:
  - AVL (Adelson-Velskii e Landis)
  - Red Black Tree
  - Splay Tree
- Operações em $O(\log_2 n)$
  - Localizar um nó, dada um valor de chave;
  - Inserir um nó, dado um valor de chave;
  - Remover um nó, dado um valor de chave.

Árvores B (B-Tree)

- Uma B-tree de ordem m é tal que*:*
  - Todas as folhas estão no mesmo nível
  - Todos os nós internos, exceto a raiz, podem ter no máximo m e no mínimo m/2 filhos
  - A raiz tem no mínimo 2 e no máximo m *filhos*

Árvores B

- Todas as chaves são mantidas em folhas, e algumas chaves são repetidas em nós não-folha para definir caminhos para localizar registros individuais
- As folhas são ligadas através de uma lista duplamente encadeada, de modo a oferecer um caminho seqüencial para percorrer as

Árvore B

[Yeung and Hall, 2011]

Árvores B

- Como os número de filhos dos nós internos pode variar, e esses não necessitam estar cheios, a árvore não necessita ser totalmente rebalanceada sempre.
- Cada nó é armazena do em uma página de disco.

1. Se o nó contém menos que o número máximo permitido, insira o elemento mantendo a ordem.
2. Senão, divida o nó em dois:
   a. O valor médio é escolhido entre os elementos do nó e o novo elemento.
   b. Valores menores que o valor médio são colocados a esquerda e maiores a direita
   c. Insira o valor médio no nó acima (caso ele tenha pai) seguindo o mesmo procedimento. Caso não tenha crie um novo nó.

Exemplo

Caçapava, Igaratá, Jacareí, Jambeiro, Monteiro Lobato, Paraibuna, Redenção, Sbranca, Taubaté e Tremembé

Responde bem a consulta: “me dê os dados de Paraibuna”

Exemplo

Caçapava, Igaratá, Jacareí, Jambeiro, Monteiro Lobato, Paraibuna, Redenção, Sbranca, Taubaté e Tremembé

Não responde a consulta: “me dê os dados das cidades dentro de tal área de interesse”. Por que?

Métodos de acesso espaciais

- Em BD espaciais, o SGBD precisa contar com métodos de acesso especificamente voltados para componente geográfica ou espacial do dado
- Os métodos tradicionais são também usados, mas apenas sobre os dados alfanuméricos
- A componente geográfica é multidimesional, portanto é preciso pensar em índices e métodos de acesso multidimensionais

Índices multidimensionais

- As árvores mostradas anteriormente, são estruturas unidimensionais, ou seja, pressupõem que a chave de pesquisa seja formada por apenas um atributo ou pela concatenação de vários atributos
- Facilitam o processamento das consultas por intervalos quando estes são unidimensionais
- Muitas vezes a busca que se deseja fazer é multidimensional

Quais os empregados com idade entre 50 e 55 anos com salário entre 3000 e 5000?

Métodos de acesso espaciais

- Métodos de acesso espaciais são estruturas de dados auxiliares, porém essenciais para o processamento de consultas e para a execução de procedimentos de análise espacial com eficiência

- Também são chamados de índices espaciais

  Ao contrário dos índices convencionais, os espaciais são de uso obrigatório, para que o desempenho seja minimamente aceitável em BDs de tamanho razoável

Métodos de acesso espaciais

- O plano de execução realiza uma *filtragem*, para determinar um subconjunto dos objetos do BD que *podem atender às* especificações da consulta
- Essa filtragem precisa ser executada com muita rapidez, e portanto é realizada sobre uma *aproximação da forma* geométrica de cada objeto

Índices espaciais

- Busca atender a consultas onde objetos são selecionados com respeito a sua localização espacial
- Índices tradicionais dependem de uma ordem total em uma chave
- Índices espaciais buscam preservar proximidade espacial

Uso de índices espaciais

- Seleção de objetos para visualização
  - Objetos contidos no retângulo do zoom
- Localização de objetos selecionados por
  - apontamento
  - Objetos cujas fronteiras contêm ou se aproximam do ponto indicado na tela
- Consultas topológicas
  - Encontrar objetos relacionados topologicamente a um objeto espacial dado (contido em, contém, adjacente a, cruzando, etc.)

Métodos de indexação espacial

- Determinados pelo espaço: baseiam-se em partições do espaço independente da distribuição dos dados (pontos ou retângulo envolvente) no plano 2D.
- Determinado pelos dados: particionam um conjunto de objetos e não o espaço

K-d Tree

- Indexa chaves formadas por *k atributos* (geometricamente, dimensões)
- Cada nível da árvore corresponde a uma das dimensões
- As dimensões ocorrem ciclicamente pelos níveis da árvore

K-d Tree

- Generalização árvore binária de pesquisa para o espaço multidimensional

K-d Tree

Monteiro Lobato
Tremembe
SJC
Taubate
Caçapava
Igaratá
Redencao
Jambeiro
Jacareí
SBranca
Paraibuna

Monteiro Lobato
SJC
Tremembé
Igarata
Cacapava
Taubaté
Jacareí
Paraibuna
SBranca
Jambeiro
Redencao

y
x

K-d Tree

Para para localizar o ponto (4,9):

raiz, N1, N4, N8 até a página P2.

Para localizar os pontos no retângulo ((1,4),(4,9)):

raiz, N1, N4, N7 e N8 até as páginas P1 e P2.

K-d Tree: Busca

- Seja um retângulo R=[(x1, y1): (x2, y2)] contendo o intervalo de pesquisa:
  - Começamos pela raiz (nível 0 → par)
  - Se o ponto deste nó ∈ R, ele é reportado
  - Se ele for um nó em um nível par:
    - Se R.x1 < p.x, aplicamos recursivamente o passo 2 à sub-árvore esquerda
    - Se R.x2 > p.x, aplicamos recursivamente o passo 2 à sub-árvore direita
  - Caso contrário, se ele for um nó em um nível ímpar:
    - Se R.y1 < p.y, aplicamos recursivamente o passo 2 à sub-árvore esquerda
    - Se R.y2 > p.y, aplicamos recursivamente o passo 2 à sub-árvore direita
- Pesquisa por apontamento: $O(log_2 n)$
- Pesquisa por janela: $O(N)$

K-d Tree

- Estrutura de dados *d*-dimensional que representa a subdivisão recursiva do espaço em subespaços por meio de *d*-1 hiperplanos. "*k*" representa a dimensão do espaço.
- Uma K-d tree é uma árvore de busca binária
- Os hiperplanos são orientados alternadamente entre as *d* possibilidades
- Cada particionamento do plano contém pelo menos um ponto, que é usado para ser representado na árvore.
- Inserções e buscas são simples, mas remoções não.
- Existem muitas variações e extensões da K-d tree: k-d tree adaptativa, hB-Tree, QuadrTree, k-b-B Tree

Quad-Tree

- Espécie de árvore em que cada nó possui sempre quatro “folhas”
- A cada nó corresponde uma região quadrada do espaço
- Os objetos são relacionados ao menor quadrado que contém seu retângulo envolvente

QuadTrees

- Baseia-se no conceito de divisão por quadrantes
- O espaço é particionado até que a capacidade da página seja atingida

QuadTree de pontos

Hashing

- Transformação de chave: consiste em criar uma série de *pacotes*, numerados seqüencialmente que receberão os identificadores
- Cada identificador que chega, seja para ser inserido, seja para ser pesquisado, é transformado em um número de 1 a n, identificando o *pacotes* correspondente a ele.

Grades fixas

- O espaço é dividido em uma grade de $n_x \times n_y$ células, igualmente espaçadas. Cada célula corresponde a uma página de disco
- Um ponto *P* é associado a célula *c* se ela o contém
- O índice requer uma matriz $[1{:}n_x\ ,\ 1{:}n_y]$ como um diretório. O elemento *DIR[i, j]* contém o endereço de página *ID* que armazena os pontos associados a célula $c_{i,j}$

Grades Fixas

- Se $[S_x \times S_y]$ é o tamanho do espaço de busca 2D, cada célula tem o tamanho $[S_x/n_x \times S_y/n_y]$.
- Operações:
  - Inserindo P(a,b):
    - Calcule $i = (a-x_0)/(S_x/n_x) + 1$ e $j = (b-y_0)/(S_y/n_y) + 1$
    - Leia a página DIR[i,j].Id e insira P
  - Consulta por apontamento: dado um ponto P(a,b) busque a página para inserção, leia a página, passe pelas entradas e verifique qual delas é igual a P.
  - Consulta por janela: calcule o conjunto de S de células *c* tais que c.MBR intercepte a janela de consulta *W*; para cada célula $c_{i,j}$ em *S*, leia a página DIR[i,j].Id e retorne os pontos na pagina que estão contidos em *W*
- Consultas por ponto requerem uma operação de E/S
- Número de operações de E/S para executar uma consulta por janela depende do número de intersecções com a janela *W*.

Grades fixas

- A resolução da grade depende do número de pontos sendo indexado; dada uma capacidade de células M, é possível criar grades fixas com pelo menos N/M células.
- Grades fixas são muito sensíveis a distribuição dos pontos

Grid-files

- Partição se adapta a distribuição dos pontos
- No diretório duas células podem apontar para a mesma página
- Duas estruturas que representam a escala de particionamento em cada eixo: S_x e S_y.

Grid-Files

- Objetos são associados as células que seus MBR interceptam

Grid Files

- Capacidade 4 nas páginas

Grid-Files - consultas

- Dado um ponto P(a,b), determine a célula que o contém. Acesse a página correspondente e obtenha a coleção de objetos E tais que P ∈ e.mbr.
- Teste exatamente quais geometrias contém o ponto
- Dado uma janela determine todas as células que a interceptam. Acesse a coleção de objetos, remova as duplicações e teste os objetos.

Consulta por ponto

Space filling curves

- Uma space-filling curves define uma ordem total nas céluas de uma frade 2-dimensional. A a cada célula é associado um número de forma que números próximos estão associados a células próximas no espaço

z-ordem

Hilbert

R-Tree

- Indexa objetos pelo seu retângulo envolvente mínimo
- Pontos também podem ser indexados (retângulo envolvente nulo)
- Cada bloco de armazenamento pode conter um número variável de retângulos
- O aumento do número de objetos causa subdivisões nos blocos, e a redução provoca fusões de blocos

R-Tree

R1
R4
R11
R3
R9
R5
R13
R10
R14
R8
R12
R2
R7
R18
R17
R6
R16
R19
R15

R1 R2
R3 R4 R5
R6 R7
R8 R9 R10
R11 R12
R13 R14
R15 R16
R17 R18 R19

R-Tree

- Existem diversas variações na literatura
  - R+-Tree
  - R*-Tree
  - Hilbert R-Tree
  - X-Tree
  - ...
- O mais usual é encontrar implementações da R-tree original nos produtos

SAM nas extensões espaciais

- Métodos de acesso espacial são implementados em extensões espaciais e são usados para tornar eficiente a consulta ao dado espacial:
- Oracle spatial: R-Tree, Quad-Tree
- PostGIS: R-Tree-over-GiST
- MySQL: R-Tree

Próximos passo…

- Exercitar tudo isso na prática com o PostGIS

- Próxima aula: trazer computador com o PostGIS instalado. De preferência instalar uma GUI como o PGAdmin.

SGBD e SIG

- Devido a evolução em termos de arquitetura (mostrado anteriormente) os SIGs podem usar diferentes SGBDS
- Podem inclusive deixar ao usuário a escolha de a qual SGBD se associar

- Qual escolher? Diferentes critérios podem ser considerados

Principais características dos SGBDs

- Licença:
  - Livres de licença (grátis)
  - Proprietários (uso sujeito a compra)
- Compartilhamento
  - Permite acesso por vários clientes em um ambiente de rede (cliente-servidor). Permite que vários usuários acessem a mesma base de dados.
  - Acesso restrito a um usuário por vez, no computador onde está instalado

Principais características dos SGBDs

- Possui extensão espacial:
  - Não. Suporta apenas tipos convencionais: texto, números, datas.
  - Sim. Também é capaz de criar e gerenciar nativamente tipos geográficos: linhas, polígonos, pontos, imagens.
- Além dos critérios mencionados acima deve-se observar outras restrições particulares de cada SGBD que possam ser de seu interesse. Exemplos:
  - Tem restrições quanto ao volume de dados armazenado?
  - É multiplataforma (Windows, Linux, MacOS)?

Exemplos

SGBD	Licença	Cliente/ Servidor	Extensão Espacial	Multiplataforma
DBase	Livre	Não	Não	Windows
ACCESS	Proprietário	Não	Não	Windows
MySQL	Livre	Sim	Parcial	Windows/Linux/Mac
PostgreSQL	Livre	Sim	Não	Windows/Linux/Mac
PostGIS	Livre	Sim	Sim	Windows/Linux/Mac
Oracle	Proprietário	Sim	Não	Windows/Linux/Mac
OracleSpatial	Proprietário	Sim	Sim	Windows/Linux/Mac
SQLite	Livre	Não	Não	Windows/Linux/Mac
SQL Server	Proprietário	Sim	Não	Windows
SQL Server Spatial	Proprietário	Sim	Sim	Windows

OBS: Todos os SGBDs mostrados acima são do tipo relacional ou objeto-relacional

Spatial Database Systems

R-trees

Gilberto Ribeiro de Queiroz
(gribeiro@dpi.inpe.br)

Roadmap

- Geometric Approximations
- R-tree
- R^*-tree
- Packed R-trees
- R^+-tree

Geometric Approximations

Well Known Approximations

MBR

RMBR

Convex Hull

MBC

MBE

Minimum Bounding Rectangle (MBR)

- MBR is the most used approximation:
  - Also known as: Minimum Bounding Box (MBB) or Bounding Box (BBOX);
  - A common approximation for objects with extension (lines and polygons);
  - In 2D space only a pair of coordinates are needed to represent it;
  - Fast filter for geometry relationship test.

MBR in Practice

Case 1) If the rectangles don't intersect: then the geometries will not intersect too.

Case 2) If the rectangles intersect: then the geometries ***may*** intersect.

R-tree

(Guttman, 1984)

R-tree: Overview

- Given a set o spatial objects in a space R^k ($k > 1$), the R-tree index organize the underlying space into a hierarchy of possible overlapping k-dimensional intervals:

8 Objects in R^2

R-tree: Overview

- Given a set o spatial objects in a space R^k ($k > 1$), the R-tree index organize the underlying space into a hierarchy of possible overlapping k-dimensional intervals:

8 Objects in R^2

Use MBR for object approximation

R-tree: Overview

- Given a set o spatial objects in a space R^k ($k > 1$), the R-tree index organize the underlying space into a hierarchy of possible overlapping k-dimensional intervals:

8 Objects in R^2

Use MBR for object approximation

Group rectangles in a hierarchy

R-tree: Overview

- Intervals are organized in a tree data structure:
  - Leaf nodes: $[(I, tuple\text{-}id)]$
  - Child nodes: $[(I, child\text{-}ptr)]$ where $I = (I_0, I_1, \ldots, I_{k-1})$

    $I_i = [a, b]$

An R-tree must satisfy the following properties

- The maximum number of entries in a node: M
- The minimum number of entries in a node: $m \geq \frac{M}{2}$
- Every node contains between *m* and *M* valid entries, unless it is the root node.
- If the tree is more than one level high, the root has at least two child nodes.
- For each entry of the form $(I, \textit{child-ptr})$, I is the smallest interval containing the intervals in the child node.
- For each entry of the form $(I, \textit{tuple-id})$, I is the smallest interval containing the spatial object.
- All leaves appear on the same level.

R-tree: Properties

- The height of an R-tree for indexing *N* spatial objects is: $\left(\log_m N\right) - 1$

- The maximum number of nodes:

$$\left\lceil \frac{N}{m} \right\rceil + \left\lceil \frac{N}{m^2} \right\rceil + \ldots + 1$$

- Worst case space utilization of nodes (except the root): $\frac{m}{M}$

R-tree: Searching

- Query algorithms traverse the tree in a top-down fashion:

```
Description:
  n -> search node
  I -> search interval

search(n, I)
  if(is_not_leaf(n))
    for_all entry in nodo do
      if(intersects(entry, I))
        search(entry.child-ptr, I)
  else /* it is a leaf node */
    for_all entry in nodo do
      if(intersects(entry, I))
        emit_found_candidate(entry.tuple-id)
```

R-tree: Insertions and Removal

Key idea: apply heuristics to minimize the number of paths traversed for improving query performance

R-tree: Insertion Operations

- Remarks:
  - When the number of valid entries in a node *n* gets larger than *M*, a new sibling node *n'* must be created and filled with part of the entries from node *n*.
  - If *n*'s parent node doesn't have room for *n'*, it has to split too.

R-tree: Insertion Algorithm

```
Description:
  I -> interval to be inserted
  n -> node

 1.    insert(I, n)
 2.      leaf_node <- choose_leaf(I, n)
 3.      if(has_room(leaf_node))
 4.        insert(I, leaf_node)
 5.        adjust_tree(leaf_node)
 6.      else /* leaf node doesn't have one more room */
 7.        leaf_node' <- split(leaf_node)
 8.        adjust_tree(leaf_node, leaf_node')
```

R-tree: Insertion Algorithm

- choose_leaf:
  - Choose the leaf that requires the least interval (or MBR) enlargement.

- adjust_tree:
  - Propagate changes upward.
  - splits may be propagate upward
  - There are several strategies for splitting nodes

R-tree: Node Splitting

- Guttman (1984) proposed 3 strategies for the partitioning of *M + 1* entries into two groups:
  - Exhaustive algorithm:
    - Evaluate all the 2^{M-1} possible grouping choices => Computational Complexity: $O(2^M)$
  - Quadratic algorithm: Computational Complexity: $O(M^2)$
  - Linear algorithm: Computational Complexity: O(M)

Source: Guttman (1984)

R-tree: Removal Operations

- When the number of entries drops below *m*, *m - 1* entries should be moved to other sibling nodes.

R-tree: Deletion Algorithm

```
Description:
  I -> interval of the object to be removed from the index
  n -> node

 1.    delete(I, n)
 2.      leaf_node <- find_leaf(I, n)
 3.      remove_entry(I, leaf_node)
 4.      condense_tree(leaf_node)

 /* has the root node n only one child? */
 5.      if(num_children(root) == 1)
 6.        n <- first_child(n).child-ptr
```

R-tree: Deletion Algorithm

- condense_tree:
  - It will eliminate the node if it has few entries and will perform the relocation of these entries (calling insert for them).
  - Propagate node elimination upward (updating intervals on the path to the root).

R^*-trees

(Beckmann et al., 1990)

R^*-tree

- Optimization is based on a combination of area, margin and overlap of each enclosing rectangle:
  - Not just area.

- Outperforms Guttman's original linear and quadratic variants.

R^+-trees

(Sellis et al., 1987)

R^+-trees

- Variant of Guttman's R-tree that adopts non-overlapping intervals:
  - Attempt to improve search operations;
  - Drawback: need some extra space.

Packed R-trees

(Roussopoulos and Leifker, 1985)

(Kamel and Faloutsos, 1993)

Packed R-trees

- Methods for bottom-up construction of an R-tree.
- Pack:
  - attempt to maximize the fill factor for nodes
- Well known variations:
  - Lower-x packed R-tree
  - Distance Sorted R-tree
  - Hilbert R-tree

Last Question…

How to process efficiently the nearest neighbor query?

References

References

- Antonin Guttman. 1984. ***R-trees: a dynamic index structure for spatial searching***. In *Proceedings of the 1984 ACM SIGMOD international conference on Management of data* (SIGMOD '84). ACM, New York, NY, USA, pp. 47-57.

- Timos K. Sellis, Nick Roussopoulos, and Christos Faloutsos. 1987. ***The R+-Tree: A Dynamic Index for Multi-Dimensional Objects***. In *Proceedings of the 13th International Conference on Very Large Data Bases* (VLDB '87), Peter M. Stocker, William Kent, and Peter Hammersley (Eds.). Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp. 507-518.

- Norbert Beckmann, Hans-Peter Kriegel, Ralf Schneider, and Bernhard Seeger. 1990. ***The R*-tree: an efficient and robust access method for points and rectangles***. *SIGMOD Rec.* 19, 2 (May 1990), pp. 322-331.

Packed R-trees

- Nick Roussopoulos and Daniel Leifker. 1985. ***Direct spatial search on pictorial databases using packed R-trees***. In *Proceedings of the 1985 ACM SIGMOD international conference on Management of data* (SIGMOD '85). ACM, New York, NY, USA, 17-31.

- Ibrahim Kamel and Christos Faloutsos. 1993. ***On packing R-trees***. In *Proceedings of the second international conference on Information and knowledge management* (CIKM '93), Bharat Bhargava, Tim Finin, and Yelena Yesha (Eds.). ACM, New York, NY, USA, 490-499.

Cost Models

- Yannis Theodoridis and Timos Sellis. 1996. ***A model for the prediction of R-tree performance***. In *Proceedings of the fifteenth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems* (PODS '96). ACM, New York, NY, USA, 161-171.

Bancos de Dados de Imagens

Lubia Vinhas

O que é um banco de dados de imagens?

- Coleções de imagens, tipicamente associadas com as atividades de uma ou mais organizações relacionadas
- Foco: organização das imagens e seus metadados de maneira eficiente
  - Algumas vezes o foco está no conteúdo das imagens
- Existem softwares disponíveis para esse fim

Características de um banco de dados de imagens

- Armazenamento eficiente
- Modelagem de dados que trate dos aspectos necessários para representar imagens no banco de dados
- Suportar a manipulação das imagens no nível do banco de dados:
  - recorte, reamostragem, manipulação, etc
- Permitir o gerenciamento dos metadados da imagem

O que seria desejável em um IM - DB

- Capacidade de manipulação
- Segurança
- Backup e recuperação
- Extensibilidade
- Flexibilidade

Metadados e Dados

Informações	
Satélite	CB2B
Sensor	HRC
Órbita	157_B
Ponto	118_1
Data de Passagem	2009-08-21
SceneId	CB2BHRC157_B118_120090821
Revolução	10081
Latitude Norte	-15.62840
Longitude Oeste	-48.04920
Latitude Sul	-15.89330
Longitude Leste	-47.85140
Tempo Central(GMT)	13:37:13
Orientação da Imagem	8.50435
Ângulo Nadir	0
Azimuth Sol	41.6492
Elevação do Sol	53.4308

Cobertura de Nuvens			
Q1	10	Q2	10
Q3	10	Q4	10

Arquitetura “Dual”

Imagens

ID	BOX	Res X	Data	Sensor	Satelite	etc	Arquivo
1	[]	20					Img1.tif
2	[]	20					Img2.tif
3	[]	30					Img3.tif

Arquitetura Integrada 1

Imagens

ID	BOX	Res X	Data	Sensor	Satelite	etc	Arquivo
1	[]	20					
2	[]	20					
3	[]	30					…

BLOB

Arquitetura Integrada 2

Imagens

ID	BOX	Res X	Data	Sensor	Satelite	etc	Arquivo
1	[]	20					RASTER
2	[]	20					RASTER
3	[]	30					…

Tipo Espacial

Resoluções

Espacial

1 x 1 m

5 x 5 m

20 x 20 m

30 x 30 m

Radiométrica

2 níveis

8 níveis

16 níveis

64 níveis

Resoluções

Existe um compromisso entre nível de detalhe e extensão

Partições

181/93 180/93 179/93 178/93 177/93 176/93 175/93 174/93 173/93 172/93 171/93 170/93
181/94 180/94 179/94 178/94 177/94 176/94 175/94 174/94 173/94 172/94 171/94 170/94
181/95 180/95 179/95 178/95 177/95 176/95 175/95 174/95 173/95 172/95 171/95 170/95
181/96 180/96 179/96 178/96 177/96 176/96 175/96 174/96 173/96 172/96 171/96 170/96
181/97 180/97 179/97 178/97 177/97 176/97 175/97 174/97 173/97 172/97 171/97 170/97
180/98 179/98 178/98 177/98 176/98 175/98 174/98 173/98 172/98 171/98 170/98 169/98
180/99 179/99 178/99 177/99 176/99 175/99 174/99 173/99 172/99 171/99 170/99 169/99
180/100 179/100 178/100 177/100 176/100 175/100 174/100 173/100 172/100 171/100 170/100 169/100
180/101 179/101 178/101 177/101 176/101 175/101 174/101 173/101 172/101 171/101 170/101 169/101
179/102 178/102 177/102 176/102 175/102 174/102 173/102 172/102 171/102 170/102 169/102
179/103 178/103 177/103 176/103 175/103 174/103 173/103 172/103 171/103 170/103 169/103
179/104 178/104 177/104 176/104 175/104 174/104 173/104 172/104 171/104 170/104 169/104 168/104
179/105 178/105 177/105 176/105 175/105 174/105 173/105 172/105 171/105 170/105 169/105 168/105

Cobertura ou *Coverage*

No Data

Pirâmide Multiresolução + Particionamento

Mecanismos de indexação usando as características de particionamento

te_layer
layer_id: INTEGER
layer_representation
te_representation
repres_id: INTEGER
layer_id: INTEGER (FK)
geom_type: INTEGER
geom_table: VARCHAR(255)
description: VARCHAR(255)
lower_x: DECIMAL(24,15)
lower_y: DECIMAL(24,15)
upper_x: DECIMAL(24,15)
upper_y: DECIMAL(24,15)
res_x: DECIMAL(24,15)
res_y: DECIMAL(24,15)
num_cols: INTEGER
num_rows: INTEGER
initial_time: DATETIME
final_time: DATETIME
te_representation_FKIndex1
layer_id
Raster_(layer_id)
geom_id: INTEGER
object_id: VARCHAR(255)
raster_table: VARCHAR(255)
lut_table: VARCHAR(255)
res_x: DECIMAL(24,15)
res_y: DECIMAL(24,15)
num_bands: INTEGER
num_cols: INTEGER
num_rows: INTEGER
block_height: INTEGER
block_width: INTEGER
lower_x: DECIMAL(24,15)
lower_y: DECIMAL(24,15)
upper_x: DECIMAL(24,15)
upper_y: DECIMAL(24,15)
tiling_type: INTEGER
raster_layer_metadata
Raster_(layer_id)_metadata
band_id: INTEGER
geom_id: INTEGER (FK)
min_value: DECIMAL(24,15)
max_value: DECIMAL(24,15)
num_bits: INTEGER
data_type: INTEGER
compression_type: INTEGER
photometric_type: INTEGER
dummy: DECIMAL(24,15)
Raster_layer_id_metadata_FKIndex1
geom_id
raster_table
block_id: VARCHAR(50)
lower_x: DECIMAL(24,15)
lower_y: DECIMAL(24,15)
upper_x: DECIMAL(24,15)
upper_y: DECIMAL(24,15)
band_id: INTEGER
resolution_factor: INTEGER
subband: INTEGER
spatial_data: BLOB
block_size: INTEGER
te_lut_table
index_id: INTEGER
r_val: INTEGER
g_val: INTEGER
b_val: INTEGER

Santos Digital

Aplicação

Camada TeraLib

PostGIS

Dados Convencionais

Tipos

`string, float, date`

Índices

`b-tree, hash`

Funções

`strlen(string), pow(float, float), now()`

Dados Geográficos

Tipos Espaciais

`geometry, geography`

Índices Espaciais

`r-tree, quad-tree, kd-tree`

Funções Espaciais

`ST_Length(geometry), ST_Area(geometry)`

A manipulação de dados convencionais quanto de dados geográficos é feita através da SQL – Structured Query Language

E as representações matricias…

como são tratadas pela extensão espacial?

Grades

Imagens

A manipulação de dados convencionais quanto de dados geográficos é feita através da SQL – Structured Query Language

PostGIS Raster

É um projeto em andamento que visa acrescentar ao PostGIS a capacidade de tratar dados com representação matricial (*raster*)

Faz parte do PostGIS a partir da versão 2.0 (~ abril/2012)

Características:

- um tipo nativo: `raster` equivalente ao tipo `geometry`
- com implementação similar e tão fácil de usar quanto o tipo `geometry`
- ser o mais integrado possível ao tipo `geometry`

PostGIS Raster e Imagens de SR

Avaliar o estado do PostGIS raster para uso em aplicações típicas de uso de imagens de SR

12 cenas CBERS adquiridas do CDSR/INPE
3 bands por cena
20 x 20 m, 8 bits
UTM WGS84 (32721)

Aplicação

1. Organização das cenas que recobrem a área de interesse e sua organização no banco de dados;
2. Obtenção de estatísticas dos valores da cobertura formada;
3. Realce de contraste das imagens que formam a cobertura;
4. Classificação;
5. Recorte das cenas pela área de interesse;
6. Mosaico dos recortes formando uma cobertura da área de interesse;
7. Cálculo de área de classes.

Banco de Dados

Flexível para qualquer arranjo desejado, pois o conteúdo de uma célula raster é auto-contido. Cabe ao usuário definir o melhor arranjo para seu caso.

Cada bada em uma tabela, cada cena dividida em blocos de 512 x 512

Banco de dados

Imagens for a do banco

raster2pgsql.py -r c:/imagesets/landsat/image.tif -t landsat -R

Importação de Dados

```
/Dados/CBERS/Banda2/$ raster2pgsql -t 512x512 -F -N 0*.tif public.band2 > band2.sql
/Dados/CBERS/Banda2/$ psql -h localhost -U postgres -w -d cbers -f band2.sql
```

Até o momento da escrita desse trabalho, não existiam SIGs com a funcionalidade de definição e injestão de images em um banco PostGIS/ Raster

```
SELECT r_table_name as table, srid, scale_x, scale_y, nodata_values, pixel_types FROM raster_columns;

table        |  srid   | scale_x | scale_y | nodata_values | pixel_types
-------------+---------+----------+----------+--------------------+------------
band2        | 32721 |       20 |      -20 | {0}                | {8BUI}
band3        | 32721 |       20 |      -20 | {0}                | {8BUI}
band4        | 32721 |       20 |      -20 | {0}                | {8BUI}
(3 registros)
```

Consulta a valores das imagens

A extensão possui, em sua SQL, funções para consulta aos valores dos pixels das imagens armazenadas, que podem ser agrupadas de diferentes maneiras

```
SELECT rid, ST_Count(rast, true) AS exclui_nodata
            ST_Count(rast, false) As inclui_nodata
FROM band2 WHERE rid=3;

rid | exclui_nodata | inclui_nodata
----+---------------------+--------------------
1   |          167086|  262144
(1 registro)
```

```
SELECT rid, (stats).*
FROM
( SELECT rid, ST_SummaryStats(rast) AS stats
  FROM band2 WHERE rid=3)
AS aux;

rid |   count  |    sum     |   mean   |  stddev  | min | max
-----+-----------+-------------+------------+------------+------+-----
  3 | 167086 | 6722299 | 40.2325 | 26.1698 |   30 | 255
(1 registro)
```

Consulta a valores das imagens

A extensão possui funções bastante específicas, por exemplo Histograma

```
SELECT (stats).*
FROM (SELECT rid, ST_Histogram(rast,1,10) AS stats
          FROM novoprogresso_b3 where rid=1) AS aux;

  min    |   max   |    count     |  percent
---------+----------+----------- ---+--------------
       1 |    26.4 |           90 |      0.000280
    26.4 |    51.8 |  29232950 |     91.300890
    51.8 |    77.2 |    2752271 |      8.595940
    77.2 |   102.6 |         5393 |      0.016840
   102.6 |     128 |              4 |      1.249280
     128 |   153.4 |              0 |      0.000000
   153.4 |   178.8 |              0 |      0.000000
   178.8 |   204.2 |              1 |      0.000003
   204.2 |   229.6 |              0 |      0. 000000
   229.6 |     255 |        27542 |      0.086019
(10 registros)
```

Realce

A extensão possui funções para alterar valores das imagens de acordo com alguma regra ou álgebra. Ex: realce de contraste.

```
UPDATE novoprogresso_b3
SET enh_rast = ST_MapAlgebraExpr(rast,NULL,'[rast]*5.9 - 128.95')
WHERE rid = 1;
```

Classificação

A extensão possui funções para fazer classificações baseadas em faixas de valores

```
ALTER TABLE band2 ADD COLUMN rast_class raster;
UPDATE band2
SET clas_rast= ST_Reclass(enh_rast,1,'0-80:1, 81-150:128, 151-255:254', '8BUI',0)
WHERE rid=128;
```

Vetorização

É possível vetorizar um resultado transformando os conjuntos de pixels com mesmo valor no polígono que os contém

```
INSERT INTO segs (class, geom)
  SELECT val, ST_GeomFromText(ST_AsText(geom) ,32721)
  FROM
    ( SELECT (ST_DumpAsPolygons(clas_rast)).*
     FROM band2 WHERE rid=128 ) AS aux ORDER val;
```

`ST_Area` sobre polígonos

Cálculo de área

Podem ser contados os números de pixels com um certo valor para saber a área

```
SELECT (pvc).value AS classe, (pvc).count AS pixels, (pvc).count*20 AS area_m2
FROM (SELECT ST_ValueCount(clas_rast) As pvc
   FROM band2 WHERE rid=128) As aux
   ORDER BY (pvc).value;

 classe |   pixels |  area_m2
---------+-----------+--------------
       1 |     1856 |     37120
   128 | 250940 | 5018800
   254 |     9348 |    186960
(3 registros)
```

Integração com o `geometry`

As extensão oferece algumas funções que combinam o tipo raster com o tipo geometry. Ex: calcular o valor médio dentro de uma área.

```
SELECT bid,  (gv).geom AS the_geom,
             (gv).val
FROM (SELECT bid,
               ST_Intersection(rast, geom2) AS gv
        FROM band2_notile, buffer_am
        WHERE ST_Intersects(rast, geom2) )  aux;
```

Integração com o tipo `geometry`

```
CREATE TABLE buffer_img_inter AS
SELECT bid, (gv).geom AS the_geom,
                (gv).val
FROM (SELECT bid, ST_Intersection(rast, geom) AS gv
          FROM band2_notile,  buffer_am
          WHERE ST_Intersects(rast, geom) ) foo;
```

```
SELECT bid, sum(ST_Area(the_geom)) * val)  /
                sum(ST_Area(the_geom) AS meanval
FROM buffer_img_inter
GROUP BY id
ORDER BY id;
```

Integração com o tipo `geometry`

Outro exemplo: recorte da imagem por um polígono.

```
ALTER TABLE band2 ADD COLUMN clip_rast raster;
UPDATE band2
SET clip_rast = ST_Clip (enh_rast, 1, munic_2001.geom)
FROM munic_2001
WHERE band2.rid=86 AND munic_2001.nome = 'Novo Progresso';
```

Overviews

raster2pgsql.py -r *.jpg -t raster_table -s 26986 -l 4 -k 100x100 -I -o raster_overview_4.sql

psql -d postgisdb -f raster_overview_4.sql

QGis

Quantum GIS 1.8.0-Lisboa
Camadas
public.cbers
Load Postgis Raster layer
Postgis' connection:
pgis_lubia
Reading mode:
Read one table as a raster
Schema
Table
SRID
Pixel T
4 public mosaic 32721 ['8BUI']
5 public cbers 32721 ['8BUI']
6 public band4_notile 32721 ['8BUI']
7 public band4_file 32721 ['8BUI']
8 public band4_512 32721 ['8BUI']
Table name:
public.cbers
Cancel
OK
Ordem de controle de renderização
Coordenada:
-54.8268,-6.2687
Escala
1:67384
Renderizar EPSG:4326

Discussão

Em termos de funcionalidades, o PostGIS / Raster mostra-se como uma alternativa bastante promissora:

- flexibilidade no modelo do banco
- simplicidade da API SQL disponível
- similaridade com o tipo geometry que é bastante maduro
- repertório de funções específicas já razoável e em expansão
- repertório de funções que misturam os dois tipos espaciais
- possibilidade de construção de sistemas interoperáveis

Discussão

Aspectos negativos:

- eficiência ainda deixa a desejar
- ainda em processo de estabilização
- falta de ferramentas amigáveis para construção e visualização da base de dados

Conclusão

Para as aplicações típicas que usam imagens de sensoriamento remoto ainda vai levar algum tempo para que existam aplicações que utilizem estensivamente o PostGIS Raster

Mas é uma alternativa livre de licença que tende a se espandir e melhorar muito no futuro próximo

Ainda há muito que se explorar...

MINISTÉRIO DA CIÊNCIA E TECNOLOGIA
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

Oracle Spatial – Geometry and GeoRaster

Karine Reis Ferreira – karine@dpi.inpe.br

CAP 349 – Bancos de Dados Geográficos (09/08/2014)

Disponível em: http://wiki.dpi.inpe.br/doku.php?id=cap349

Oracle Spatial

- Oracle Spatial is an integrated set of functions and procedures that enables spatial data (vector and raster) to be stored, accessed, and analyzed in an Oracle database.
- Comercial system
- Current version: 11g
- http://www.oracle.com/index.html

Oracle Spatial provides

- A schema (MDSYS) that prescribes the storage, syntax, and semantics of geometric and raster data types
- A spatial indexing mechanism
- Operators, functions, and procedures for performing area-of-interest queries, spatial join queries, and other spatial analysis operations
- Topology data model for working with data about nodes, edges, and faces in a topology
- Network data model for representing capabilities or objects that are modeled as nodes and links in a network
- GeoRaster, a feature that lets you store, index, query, analyze, and deliver GeoRaster data, that is, raster image and gridded data and its associated metadata

Oracle Spatial – Vector Data

Geometry data types

Oracle Spatial – Vector Data

- Tipos de dados geométricos.
- Operadores e funções espaciais.
- Métodos de Acesso Espacial:
  - R-Tree e QuadTree

```
CREATE TYPE SDO_GEOMETRY AS OBJECT (
SDO_GTYPE          NUMBER,
SDO_SRID           NUMBER,
SDO_POINT          SDO_POINT_TYPE,
SDO_ELEM_INFO      SDO_ELEM_INFO_ARRAY,
SDO_ORDINATES      SDO_ORDINATE_ARRAY);
```

Oracle Spatial – SDO_GEOMETRY

- Criação de tabelas com tipos de dados espaciais:

```
CREATE TABLE distritossp
( cod              NUMBER(32),
  sigla            VARCHAR(10),
  denominacao      VARCHAR(50),
  spatial_data     MDSYS.SDO_GEOMETRY
  PRIMARY KEY (cod)
);
```

Oracle Spatial – Metadata Tables

MDSYS.CS_SRS

SC_NAME	VARCHAR2(68)
SRID	NUMBER(38)
AUTH_SRID	NUMBER(38)
AUTH_NAME	VARCHAR2(256)
WKTEXT	VARCHAR2(2046)
SC_BOUDS	SDO_GEOMETRY

USER_SDO_GEOM_METADATA

TABLE_NAME	VARCHAR2(32)
COLUMN_NAME	VARCHAR2(32)
DIMINFO	SDO_DIM_ARRAY
SRID	NUMBER

USER_SDO_INDEX_INFO

SDO_INDEX_OWNER	VARCHAR2(32)
INDEX_NAME	VARCHAR2(32)
TABLE_NAME	VARCHAR2(32)
COLUMN_NAME	VARCHAR2(32)
SDO_INDEX_TYPE	VARCHAR2(32)
SDO_INDEX_TABLE	VARCHAR2(32)
SDO_INDEX_STATUS	VARCHAR2(32)

Oracle Spatial – Examples

- Inserindo dados em tabelas com tipos de dados espaciais:

```
INSERT INTO distritossp (cod, sigla,
denominacao, spatial_data)
VALUES (1, 'VMR', 'VILA MARIA'
MDSYS.SDO_GEOMETRY(2003, NULL, NULL,
MDSYS.SDO_ELEM_INFO_ARRAY( 1, 1003, 1 ),
MDSYS.SDO_ORDINATE_ARRAY(6,10, 10,1, 14,10,
10,14, 6,10)))
```

Oracle Spatial – Examples

- Indexando uma coluna espacial (R-Tree):

```
CREATE INDEX distritossp_IDX
ON distritossp(SPATIAL_DATA)
INDEXTYPE IS MDSYS.SPATIAL_INDEX
```

- Funções para trabalhar com os índices:

```
SDO_TUNE.QUALITY_DEGRADATION
ALTER INDEX REBUILD
```

Oracle Spatial – Spatial Query

- Operadores:
  - Usados na cláusula WHERE de uma consulta SQL
  - Utilizam indexação espacial

Operadores	Descrição
SDO_FILTER	Implementa o primeiro filtro do modelo de consulta (baseado nos MBR)
SDO_RELATE (SDO_TOUCH, SDO_ON, SDO_INSIDE)	Avalia se as geometrias possuem uma determinada relação topológica
SDO_WITHIN_DISTANCE	Verifica se duas geometrias estão dentro de uma determinada distância.
SDO_NN	Identifica os n vizinhos mais próximos de uma geometria

Oracle Spatial – Spatial Query

- Funções:
  - Definidas como subprogramas PL/SQL
  - Usados na cláusula WHERE ou em SUBCONSULTAS
  - Podem ser utilizadas sobre colunas espaciais não indexadas

Funções	Descrição
SDO_INTERSECTION, SDO_UNION SDO_DIFFERENCE, SDO_XOR	Operações de conjunto
SDO_BUFFER, SDO_CENTROID, SDO_CONVEXHULL	Operações que geram novas geometrias
SDO_AREA, SDO_ LENGTH, SDO_DISTANCE	Operações métricas

Oracle Spatial – Spatial Query

Buffer

Tolerance

Oracle Spatial – Spatial Query

Topological relations

Oracle Spatial – Examples

- “Recuperar o nome de todos os municípios da grande São Paulo que são vizinhos ao município de São Paulo”.

```
SELECT d2.nomemunicp
FROM  grande_sp d1,
       grande_sp d2
WHERE SDO_TOUCH (d1.spatial_data,
                d2.spatial_data) = 'TRUE'
AND (d2.nomemunicp <> 'SAO PAULO')
AND (d1.nomemunicp = 'SAO PAULO')
```

Oracle Spatial – Examples

- “Recuperar todos os distritos que estão num raio de 3Km de um determinado rio”

```
SELECT di.deno
FROM   sp_distritos di,
       sp_drenagem dr,
       user_sdo_geom_metadata m,
WHERE
   SDO_RELATE (di.spatial_data,
   SDO_BUFFER (dr.spatial_data, m.diminfo, 3000),
   'mask=INSIDE+TOUCH+OVERLAPBDYINTERSECT') = 'TRUE'
AND m.table_name = 'sp_drenagem'
AND m.column_name = 'spatial_data'
AND dr.object_id = '59';
```

Oracle Spatial – Raster Data

- GeoRaster is a feature of Oracle Spatial that lets you store, index, query, analyze, and deliver raster data and its associated metadata.

Oracle GeoRaster – Architecture

Five components to support the storage and use of raster data in Oracle Database:

Oracle GeoRaster – Architecture

Five components to support the storage and use of raster data in Oracle Database:

GeoRaster Engine: provides the native GeoRaster object type and GeoRaster functionality including raster data and metadata indexing, update, query and manipulations.

Oracle GeoRaster – Architecture

Five components to support the storage and use of raster data in Oracle Database:

SQL API : standard SQL access to the raster and grid-based data in GeoRaster databases.

Oracle GeoRaster – Architecture

Five components to support the storage and use of raster data in Oracle Database:

SQL API : standard SQL access to the raster and grid-based data in GeoRaster databases.

Three PL/SQL packages:

(1) MDSYS.SDO_GEOR: for creating, modifying, and retrieving GeoRaster objects

(2) MDSYS.SDO_GEOR_UTL: for utility operations related to GeoRaster

(3) MDSYS.SDO_GEOR_ADMIN: for administrative operations related to GeoRaster

Oracle GeoRaster – Architecture

Five components to support the storage and use of raster data in Oracle Database:

C/C++/Java – Java, OCI, and OCCI : access to the raster and grid based data in GeoRaster with or without calling the GeoRaster SQL API.

Oracle GeoRaster – Architecture

Five components to support the storage and use of raster data in Oracle Database:

Viewing Tools: A variety of third party visualization and analysis tools:

(1) Oracle Fusion Middleware MapViewer;

(2) GeoRaster Viewer: a standalone viewer comes with the Oracle GeoRaster installation and can be used as a development or DBA tool.

Oracle GeoRaster – Architecture

Five components to support the storage and use of raster data in Oracle Database:

Input and Output [data] adapters: Facilitate loading and unloading raster data between well-known image file formats and GeoRaster. A variety of third party ETL tools now support loading and unloading GeoRaster data. GeoRaster also provides limited importing and exporting capability on six standard image file formats through both the server-side SQL API and the client-side Java tool.

GeoRaster – Logical Data Model

Oracle defines *georaster* object as a multidimensional matrix of cells (raster) and a set of metadata. It is logically layered.

GeoRaster – Logical Data Model

Raster: a **multidimensional** matrix of raster cells. Each cell is one element of the matrix, and its value is called the cell value. The matrix has a number of dimensions, a cell depth, and a size for each dimension.

It can be **blocked** for optimal storage, retrieval and processing.

Pyramids (generalized, lower-resolution versions of the image – useful for fast retrieval in web applications) of the core raster data can be generated, stored and processed the same way.

GeoRaster – Logical Data Model

A georaster object is logically **layered**. The core data is called the **object layer** or layer 0, and consists of one or more logical layers (or sublayers).

GeoRaster – Logical Data Model

GeoRaster metadata:

(1) Object information (cell depth, blocking size, compression, info about pyramids, …)

(2) Raster information

(3) Spatial reference system information

(4) Date and time (temporal reference system) information

(5) Spectral (band reference system) information

(6) Layer information for each layer (RGB colormap, grayscale lookup table, statistics, NODATA values, value ranges, …)

GeoRaster Engine

Physically, the GeoRaster data model is embodied as:

(1) two native data types: SDO_GEORASTER and SDO_RASTER

(2) an object-relational schema inside Oracle ORDBMS.

GeoRaster – Database Schema

Schema designed to store and manage raster data inside the database.

GeoRaster – Database Schema

Schema designed to store and manage raster data inside the database.

GeoRaster table: A GeoRaster table is any user-defined table, which has at least one data column of type SDO_GEORASTER.

GeoRaster – Database Schema

Schema designed to store and manage raster data inside the database.

SDO_GEORASTER Object: include metadata and information about *how* to retrieve the raster data stored in another user-defined table called a *Raster Data Table*.

GeoRaster – Database Schema

Schema designed to store and manage raster data inside the database.

Raster data table: user-defined table which is an object table of type SDO_RASTER.

SDO_RASTER Object: includes a BLOB column called RASTERBLOCK, which stores the real raster blocks.

GeoRaster – Database Schema

Schema designed to store and manage raster data inside the database.

Other information associated with the GeoRaster objects can be stored in separate columns or tables, such as a Value Attribute Table (VAT).

SDO_GEORASTER Object

Native data type: each image or raster grid is stored as a single object of this native type.

```
CREATE TYPE sdo_georaster AS OBJECT (
        rasterType NUMBER,
        spatialExtent SDO_GEOMETRY,
        rasterDataTable VARCHAR2(32),
        rasterID NUMBER,
        metadata XMLType);
```

SDO_GEORASTER Object

Native data type: each image or raster grid is stored as a single object of this native type.

```
SQL> describe mdsys.sdo_georaster
 Name                                      Null?    Type
 ----------------------------------------- -------- ---------------------
 RASTERTYPE                                         NUMBER
 SPATIALEXTENT                                      MDSYS.SDO_GEOMETRY
 RASTERDATATABLE                                    VARCHAR2(32)
 RASTERID                                           NUMBER
 METADATA                                           XMLTYPE
```

SDO_GEORASTER Object

Native data type: each image or raster grid is stored as a single object of this native type.

```
SQL> describe mdsys.sdo_georaster
 Name                                      Null?    Type
 ----------------------------------------- -------- ---------------------
 RASTERTYPE                                         NUMBER
 SPATIALEXTENT                                      MDSYS.SDO_GEOMETRY
 RASTERDATATABLE                                    VARCHAR2(32)
 RASTERID                                           NUMBER
 METADATA                                           XMLTYPE
```

RasterType: contains dimensionality information and the data type that can be extended

SpatialExtent: spatial extent of the raster. GeoRaster uses R-Tree to index them.

RasterDataTable: the table name where the raster is physically stored.

RasterId: the index of the raster in the *Raster Data Table*.

Metadata: XML document (Oracle XML Type data type) according to the GeoRaster metadata XML schema defined by GeoRaster

SDO_GEORASTER Object: RasterType

5-digit number in the format [**d**][**b**][**t**][**gt**], where:

[**d**] identifies the number of spatial dimensions. Must be 2 for the current release.

[**b**] indicates band or layer information: 0 means one band or layer; 1 means one or more than one band or layer.

[**t**] is reserved for future use and should be specified as 0 (zero).

[**gt**] identifies the 2-digit GeoRaster type:

00	Reserved for Oracle use.
01	Any GeoRaster type. This is the only value supported for the current release.
02-50	Reserved for Oracle use.
51-99	Reserved for customer use in future releases.

SDO_GEORASTER Object: RasterType

For example, a RasterType value of 20001 means:

Two-dimensional data

One band (layer)

Any GeoRaster type

SDO_RASTER Object

Native type: each block of the image or raster grid (of a SDO_GEORASTER object) is stored as a single object of this type.

```
CREATE TYPE sdo_raster AS OBJECT (
        rasterID NUMBER,
        pyramidLevel NUMBER,
        bandBlockNumber NUMBER,
        rowBlockNumber NUMBER,
        columnBlockNumber NUMBER,
        blockMBR SDO_GEOMETRY,
        rasterBlock BLOB);
```

SDO_RASTER Object

Native type: each block of the image or raster grid (of a SDO_GEORASTER object) is stored as a single object of this type.

```
SQL> describe mdsys.sdo_raster
 Name                                      Null?    Type
 ----------------------------------------- -------- --------------------
 RASTERID                                           NUMBER
 PYRAMIDLEVEL                                       NUMBER
 BANDBLOCKNUMBER                                    NUMBER
 ROWBLOCKNUMBER                                     NUMBER
 COLUMNBLOCKNUMBER                                  NUMBER
 BLOCKMBR                                           MDSYS.SDO_GEOMETRY
 RASTERBLOCK                                        BLOB
```

SDO_RASTER Object

Native type: each block of the image or raster grid (of a SDO_GEORASTER object) is stored as a single object of this type.

```
SQL> describe mdsys.sdo_raster
 Name                                      Null?    Type
 ----------------------------------------- -------- ---------------------
 RASTERID                                           NUMBER
 PYRAMIDLEVEL                                       NUMBER
 BANDBLOCKNUMBER                                    NUMBER
 ROWBLOCKNUMBER                                     NUMBER
 COLUMNBLOCKNUMBER                                  NUMBER
 BLOCKMBR                                           MDSYS.SDO_GEOMETRY
 RASTERBLOCK                                        BLOB
```

RasterId: the raster id

PyramidLevel: the pyramid level of this block

BandBlockNumber: the band block number

RowBlockNumber: the row block number

ColumnBlockNumber: the column block number

BlockMBR: the precise extent of the block

RasterBlock: the raster block as a binary large object (BLOB)

Example

CITY_IMAGES table
(one row per city)
For each row
(each image):
(Various user-defined columns...)
SDO_GEORASTER object
(for example, for Boston)
SDO_GEORASTER object
GeoRaster type
Spatial extent for this image (SDO_GEOMETRY)
Raster data table name (table of SDO_RASTER)
Raster ID
Metadata (SYS.XMLType)
Raster data table
(one row of SDO_RASTER object type for each block)
For each row
(each block
of the image):
(Raster ID,
pyramid level,
...)
MBR for this block
(SDO_GEOMETRY)
Image data for
this block (BLOB)

GeoRaster – Layer, Bands and Interleaving

In GeoRaster, *band* and *layer* are different concepts.

Band is a physical dimension of the multidimensional raster data set. Bands are numbered from *0* to *n-1*, where n is the highest layer number.

Layer is a logical concept in the GeoRaster data model. Layers are mapped to bands. Typically, one layer corresponds to one band. Layers are numbered from *1* to *n*; that is, *layerNumber* = *bandNumber* + 1.

A **GeoRaster object** can contain multiple bands, which can also be called multiple layers.

Interleaving: Must be one of the following values: BSQ (band sequential), BIL (band interleaved by line), or BIP (band interleaved by pixel). Example: interleaving=BSQ

GeoRaster – Layer and Band

In GeoRaster, each layer is a two-dimensional matrix of cells that consists of the row dimension and the column dimension.

Example: for multichannel remote sensing imagery, the sublayers are used to model the channels or bands of the imagery.

This figure shows an image with multiple layers and a single raster data table. Each layer contains multiple blocks, each of which typically contains many cells. Each block has an entry in the raster data table. Note that GeoRaster starts layer numbering at 1 and band numbering at 0 (zero).

GeoRaster – Interleaving

Interleaving: Must be one of the following values: **BSQ (band sequential)**, BIL (band interleaved by line), or BIP (band interleaved by pixel). Example: interleaving=BSQ

Source: Esri ArcGIS's home page

GeoRaster – Interleaving

Interleaving: Must be one of the following values: BSQ (band sequential), **BIL (band interleaved by line)**, or BIP (band interleaved by pixel). Example: interleaving=BSQ

Source: Esri ArcGIS's home page

GeoRaster – Interleaving

Interleaving: Must be one of the following values: BSQ (band sequential), BIL (band interleaved by line), or **BIP (band interleaved by pixel)**. Example: interleaving=BSQ

Source: Esri ArcGIS's home page

GeoRaster – Interleaving

IMPORTANT NOTES:

(1) SDO_GEOR.importFrom: This procedure does not support source multiband raster data with BIL and BSQ interleaving types. Only BIP interleaving.

Case Study

Region of interest: Novo Progresso, Pará

Images: 12 CBERS-2B scenes (CCD sensor)

Case Study - Images

Spatial reference system: UTM / WGS-84 Datum.

Spatial resolution: 20 meters

Radiometric resolution: 8 bits unsigned. It means that each image element, or pixel, has an integer value in the range of 0 to 255. The value 0 indicates pixels with "*no data*", or with no valid information.

Each scene has 3 bands (2, 3 and 4)

Each scene is a GeoTIFF file.

Total: 36 files.

Case Study – Create Tables

```
CREATE TABLE para_georaster
(       r_georid NUMBER,
        r_scene VARCHAR(10),
        r_band NUMBER,
        r_satellite VARCHAR(20),
        r_date VARCHAR(20),
        r_image MDSYS.SDO_GEORASTER);


CREATE TABLE para_raster OF MDSYS.SDO_RASTER
(PRIMARY KEY ( rasterID, pyramidLevel,
               bandBlockNumber, rowBlockNumber,
               columnBlockNumber));
```

Case Study – Import GeoTIFF files

One scene with its three bands is represented as a `georaster` object and stored in a row of the `para_georaster` table.

Tiles: 512 x 512 pixels.

Case Study – Import GeoTIFF files

We have 2 strategies to insert the GeoTIFF files (each file is one band) into the database using the package `sdo_geor`:

1. Import each file as it is, using the `sdo_geor.importFrom` function and, afterwards, use the `sdo_geor.mergeLayers` function to merge all bands of the same image into a single georaster object.

2. Create a new GeoTiff file combining all bands of a scene and, afterwards, use the `sdo_geor.importFrom` function to import the new file to a georaster object. The function `sdo_geor.importFrom` supports only multiple BIP (band interleaved by pixel) GeoTiff files.

Case Study – Import GeoTIFF files

```
DECLARE
        geor10 SDO_GEORASTER;
BEGIN
        INSERT INTO para_georaster VALUES(10, '167_108', 234,
        'CBERS2B_CCD1XS', '20090820',
        sdo_geor.init('para_raster') );

        SELECT r_image INTO geor10 FROM para_georaster
        WHERE r_georid = 10 FOR UPDATE;

        sdo_geor.importFrom(geor10,'blocking=TRUE
        blocksize=(512,512) spatialExtent=TRUE srid=32721',
        'GeoTIFF', 'file', '/home/../file1.tif');

        UPDATE para_georaster SET r_image = geor10
        WHERE r_georid = 10;
COMMIT;
END;
```

Using strategy 2

Case Study – Import GeoTIFF files

`sdo_geor.init` function: register automatically the new raster objects and their related raster data tables in the two metadata tables `user_sdo_geor_sysdata` and `all_sdo_geor_sysdata`.

`sdo_geor.importFrom` function: storage parameters (block size, compression type, pyramid generation using different resampling methods, …).

Important Note: to use the function `mdsys.sdo_geor.init`, your Oracle user must have permition to insert and update tables in the MDSYS schema!!!! Because of the metadata tables!

Case Study – Image Access

Return the raster value in a position given by a spatial coordinate (672512.103, 9214134.635):

```
SELECT sdo_geor.getCellValue(r_image, 0,
               sdo_geometry(2001, 32721,
               sdo_point_type(672512.103, 9214134.635, null),
               null, null), 1)
FROM para_georaster
WHERE r_georid = 10;
```

Case Study – Image Access

Extract statistics and histogram:

```
DECLARE
        geor10   SDO_GEORASTER;
        window   SDO_NUMBER_ARRAY := NULL;
BEGIN
        SELECT r_image INTO geor10 FROM para_georaster
        WHERE r_georid = 10 FOR UPDATE;

        sdo_geor.setBinFunction(geor10, 1,
                                sdo_number_array(0,10,1,0,255));

        sdo_geor.generateStatistics(  geor10, 'samplingFactor=1',
                                      window,'TRUE’,'1','TRUE');

        UPDATE para_georaster SET r_image=geor10 WHERE r_georid=10;
COMMIT;
END;
```

Case Study – Image Access

`sdo_geor.generateStatistics` function: extracts summarizing values, such as minimum, maximum and mean values. It can extract a histogram parameterized with a function to control the number of bins. It can also retrieve the statistics of a specific window within the image and to disregard *nodata* values.

This function stores the results in the georaster object metadata.

…

Case Study – Image Access

Get the store statistics and histogram from the georaster objects:

```
SELECT sdo_geor.getStatistics(r_image, 1)
FROM para_georaster WHERE r_georid = 10;

SELECT sdo_geor.getHistogram(r_image, 1)
FROM para_georaster WHERE r_georid = 10;
```

Case Study – Visualization

GeoRaster Viewer:

Case Study – Visualization

Georaster objects stores metadata about their visualization:

(1) Color map or *pallets*: mechanism to transform a range of input values into a range of colors (functions `sdo_geor.setColorMapTable` and `sdo_geor.setColorMap`)

Case Study – Visualization

Georaster objects stores metadata about their visualization:

(2) Association of image bands to Red-Green-Blue components of a display using the functions:

```
DECLARE
        geor SDO_GEORASTER;
BEGIN
        SELECT r_image INTO geor FROM para_georaster
        WHERE r_georid = 86 FOR UPDATE;

        sdo_geor.setDefaultRed(geor, 1);
        sdo_geor.setDefaultGreen(geor, 3);
        sdo_geor.setDefaultBlue(geor, 2);

        UPDATE para_georaster SET r_image = geor
        WHERE r_georid = 86;
COMMIT; END;
```

Case Study – Clip

We can clip a georaster object using a geometry (SDO_GEOMETRY) as a mask through the function `sdo_geor.subset`:

Geoprocessamento e Internet

Lúbia Vinhas

Divisão de Processamento de Imagens

INPE

Internet

- A **Internet** é um sistema de global que liga bilhões de redes de computadores administradas, mantidas e sustentadas em separado por diferentes instituições e organizações
- **World Wide Web** (WWW) é uma das aplicações construídas sobre a internet (outro exemplo: correio eletrônico)
  - Rede de hipertextos que podem ser vistos em navegadores, em uma arquitetura cliente-servidor
- Protocolos padrão e abertos
  - Ex: Internet Protocol Suite (TCP/IP), HTTP, POP, WWW, XML...

Arquitetura cliente-servidor

- Estrutura de computação distribuída que divide as tarefas entre os fornecedores de um recurso, servidores, e seus consumidores chamados clientes

Exemplo de arquitetura Cliente-Servidor
Servidor Web

WWW - World Wide Web

- Hipertexto: texto mostrado em um computador ou outro dispositivo eletrônico com referências (ou hiperlinks) a outros textos que podem ser acessados facilmente através de um clique de mouse. Além de textos, podem conter tabelas, imagens vídeos, etc.
- A WWW – World Wide Web, ou simplesmente Web, é formada por bilhões de páginas de hipertexto
- A WWW é regulada pelo W3C Consortium

HTML

- Páginas Web são escritas usando a linguagem HTML – Hypertext Markup Language

w3c.htm

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">

<html>
<head>
<title>WWW</title>
</head>
<body>
<h1>HTML - Hypertext Markup Language</h1>

<p>É a linguagem predominates na criação de páginas Web.</p>

<p>É regulada pelo Consórcio <a href="http://www.w3.org/">W3C</a>.</p>

<p><center><img src="w3c.jpg" alt="W3C" /></center></p>

</body>
</html>
```

HTML Dinâmico

- É possível criar páginas web que são dinâmicas e suportam interação, ou seja, podem dar um resultado diferente para cada acesso. Para isso são usadas outras linguagens além de HTML, como PHP, JavaScript, CSS, etc.

dhtml2.htm *

```
1  <html>
2  <body>
3
4  <p>Hoje é:</p>
5  <script type="text/javascript">
6  document.write(Date());
7  </script>
8
9  </body>
10 </html>
11
12
```

HTML Dinâmico

- Torna possível fazer algo mais interessante como mostrar o conteúdo de uma base de dados. Exemplo: home banking

Entro com **minha conta e senha** e o sistema mostra a **minha página**

Web Mapping

Termo genérico significando o uso da web para disseminar dados com conteúdo geográfico. Ou aplicações geográficas no ambiente web. Faz uso de tecnologias e ferramentas para construir páginas dinâmicas.

Cliente é um navegador Web

Base de dados com conteúdo geográfico

Mapas estáticos

- Programas rodando no servidor produzem imagens e as enviam para o navegador do usuário. Nenhuma (ou pouca) interação é permitida
- Exemplo: Mosaico do Brasil (http://www.dpi.inpe.br/mosaico/)
  - Conjunto de camadas ou mapas com dados de sensoriamento remoto ou cartografia
  - Os mapas são divididos em blocos em diferentes nível de zoom, e já estão pré-calculados

Mosaico do Brasil - Mozilla Firefox
Arquivo Editar Exibir Histórico Favoritos Ferramentas Ajuda
http://www.dpi.inpe.br/mosaico/
Divisão de Processamento de Imagens ...
Mosaico do Brasil
Mosaico do Brasil
Landsat 1990/Político
MENU Mosaico do Brasil
São Luís
Belém
Manaus
Fortaleza
Porto Velho
Palmas
Aracaju
Salvador
Cuiabá
Brasília
Campo Grande
Belo Horizonte
São Paulo
Rio de Janeiro
Curitiba
Florianópolis
Porto Alegre
S45:00:00 O75:00:00

Mosaico do Brasil - Mozilla Firefox
Arquivo Editar Exibir Histórico Favoritos Ferramentas Ajuda
http://www.dpi.inpe.br/mosaico/
Divisão de Processamento de Imagens ...
Mosaico do Brasil
10 coisas que aconteceriam caso o cas
Mosaico do Brasil
Apresentação | Navegar | Intere
Landsat 1990/Político
S23:12:00 O45:36:00
MENU Mosaico do Brasil
Jambeiro
Jacareí
Santa Branca
Paraibuna
10 km
S:23:29:52 O:45:41:43

Outros exemplos

- Mosaico do Brasil foi uma das primeiras aplicações web mapping do INPE
- Outras aplicações mais recentes:
  - Site de disseminação dos dados do PRODES e DETER
  - Site de disseminação dos focos de calor detectados por satélite (QUEIMADAS)

Arquitetura de aplicações WEB

[Yeung and Hall, 2011]

PRODES

BDQUEIMADAS – Banco de
Prodes Digital
www.dpi.inpe.br/prodesdigital/prodes.php
(5 não lidos) – lubia
New Tab
Technology giants a
Wiki Table Editor
ISO 19115 and 1911
Ministério da Ciência e Tecnologia
Destaques do
PRODES
OBT
DPI
Recompor
Imagens Satélite
Cartografia
Consulta Cenas Individuais
Selecione Ano
2000 a 2012
e/ou selecione Orbita/Ponto (*)
Estado/Regiao
AM
Município (opcional)
Consultar
(*) Segundo grade Landsat TM
Consulta Mosaicos Estaduais
Selecione Ano
2012
Estado/Regiao
Toda Amazonia Legal
Download
Desmatamento nos Municípios
Desmatamento nas Unidades de Conservação
Donwload dos dados (sem interface gráfica)
Acessórios
Ajuda...
Descrição das Classes
Classes Sisprodes x Spring
Home PRODES
Dúvidas, comentários e sugestões:
prodes@dpi.inpe.br
TerraLib
php
MySQL
Mosaico IRS/DMC 2012 (AMZ)/Nenhuma
N00:00:00 O52:00:00
500 km
S:10:31:60 O:66:35:12
S12:00:00 O68:00:00
13 cena(s) nesta tela...
Gerar Tabela para Download

BDQUEIMADAS – Banco de
Prodes Digital
www.dpi.inpe.br/proarco/bdqueimadas/
(5 não lidos) – lubia
New Tab
Technology giants
Wiki Table Editor
ISO 19115 and 1911
Ciência e Tecnologia e Meio Ambiente
Ministério da Ciência e Tecnologia e Inovação e Ministério do Meio Ambiente
BRASIL
Queimadas
Monitoramento de Focos
CPTEC
OBT
DPI
Recompor
Meteorologia
Imagens Satélite
Cartografia
Mapas Temát
Parâmetros Básicos
Data Inicial (aaaa-mm-dd) 2013-05-19 00:00:00 GMT
Data Final (aaaa-mm-dd) 2013-05-20 23:59:59 GMT
Pais BRASIL
Estado/Regiao (um ou mais) TODOS AC AL AM
Municipio (opcional)
Satélite (um ou mais) TODOS Satelite Referencia NOAA 15 Noite NOAA 15 Manha
Bioma Brasileiro Todos
Região (opcional)
Norte 55.0
Oeste -120.0 Leste 65.0
Sul -55.0
Coordenada Especifica (opcional)
Latitude Longitude
Consultar
Gráficos
Tipo Politico
Histograma
Focos nas Unidades de Conservação...
Acessórios
Coordenadas dos focos na projeção UTM, Policônica, Mercator, Albers...
Focos NOAA Antigos: 1992 A 1998...
Ajuda...
Modis Terra/RapidResponse 2013-05-19/Divisão Política/
Bogota
Sao Luis
Palmas
Brasilia
Campo Grande

Queimadas

Interoperabilidade

- Os exemplos mostrados anteriormente são soluções ad-hoc que atendem a usuários humanos
  - ele usa o navegador para ir até o site
  - ele aprende a interagir com os diferentes formulários
  - ele faz download dos dados para a sua máquina local, para poder usá-los em seus trabalhos
- A partir dos anos 2000 surge um grande investimento em tecnologias para facilitar o compartilhamento de dados, considerando que:
  - existem diferentes produtores de dados, com organização e tecnologias diferentes
  - é desejável que se possa visualizar/acessar o dado em outro cliente além do navegador. O SIG por exemplo

Interoperabilidade

- Capacidade de trocar e usar informações numa rede distribuída. Envolve questões sintáticas, semânticas e de infra-estrutura, resolvidas através de acordos e/ou padrões reconhecidos e aceitos pelos diferentes atores envolvidos no assunto
- Exemplos de organizações envolvidas com interoperabilidade:

ISO International Organization for Standardization

W3C Brasil

INMETRO

Padrões USA

International Organisation for Standardisation (ISO)
Is a committee of
Is member of
ISO Technical Committee (TC) 211
Inter-organisational liaison
Accredits U.S. TAG to
American National Standards Institute (ANSI)
Office of Management and Budget
Accredits
International Committee on Information Technology Standards (INCITS)
Sets policies for
States
Software Vendors
Is a Sub-committee of
Members of
Participate in standards development
Federal Geographic Data Committee (FGDC)
Is a voting member of
INCITS Technical Committee L1-GIS
Is an advisory member of
Open Geospatial Consortium (OGC)
Member of
Academic & research community
Members of
Counties / Cities
Is a member of
Other Consortia
Federal Spatial Data Standards
Standards Development Organisations
Industry Consortia

OGC®
Open Geospatial Consortium, Inc.

- O Open Geospatial Consortium, Inc.® (OGC) é uma organização sem fins lucrativos, internacional, formada pela academia, indústria e comunidade, que lidera as questões de interoperabilidade no contexto de aplicações geográficas
- Desenvolve especificações para produtos, formatos de dados e serviços geográficos.
- Essas especificações visam resolver questões de interoperabilidade, de forma que dois sistemas possam se comunicar.

OGC®
Open Geospatial Consortium, Inc.

- O OGC tem dezenas de padrões relativos a diferentes aspectos da construção de operações geográficas tais como:
  - a estrutura de geometrias
  - a representação de geometrias em formato texto e binário
  - como implementar operadores geográficos (ex. toca, cruza, etc.)
  - como armazenar geometrias em bancos de dados
  - serviços web para a troca de dados e processamentos

Web Services

- **Web Service** é uma solução utilizada na integração de sistemas e na comunicação entre aplicações diferentes. Uma aplicação pode invocar outra para efetuar tarefas simples ou complexas mesmo que estejam em diferentes sistemas e escritas em linguagens diferentes
  - São trocadas requisições e respostas
- Web Services proveem uma funcionalidade bem especificada e atômica
- Web Services baseiam-se em protocolos e padrões abertos

O framework baseado em serviços para compartilhar dados espaciais

Aplicações Web

Web Services

Web Service

Web Services Geográficos

- Quais são as funcionalidades básicas em aplicações geográficas?
  - Visualizar mapas
  - Acessar dados
  - Processar dados

- Como devem ser as requisições e as repostas?

Web Services Geográficos

- Quais são as funcionalidades básicas em aplicações geográficas?
  - Visualizar mapas, acessar dados e processar dados

Service Framework

- Define um arcabouço para a criação de serviços web no domínio geo-espacial (OGC OWS) e padroniza requisições e respostas para alguns serviços

WMS – Web Map Service

- O serviço Web Map Service (WMS) da OGC é um serviço para a produção de mapas na Internet. O mapa é uma representação visual dos dados geográficos e não os dados de fato. São representações geradas em formatos de imagem, como PNG, GIF e JPEG.

WMS – Exemplo de implementação

- Cliente: Plugin WMS do TerraView
- Servidor: Mapas do MMA

1) Enviar ao servidor a requisição das suas capacidades

WMS – Exemplo de implementação

2) Receber a descrição das camadas

WMS – Exemplo de implementação

3) Solicitar uma ou mais camadas

WMS – Exemplo de implementação

4) Desenhar as camadas

5) Obter propriedades de uma feição

WMS

- Ao construir meu cliente, eu não sei, nem preciso saber:
  - Como os dados estão organizados internamente no servidor
  - Qual a plataforma de software que está sendo usada no servidor
- Ao construir meu servidor, eu não sei, nem preciso saber:
  - Quais clientes irão acessá-lo
  - Qual a plataforma de software que está sendo usada no cliente
- Servidores e clientes seguem:
  - Os protocolos de comunicação acordados
  - São fiéis as interfaces acordadas
  - Protocolos e interfaces são **abertos**
- Servidores devem atender as requisições mínimas definidas na especificação definida pelo OGC

WFS – Web Feature Service

- A especificação OpenGIS Web Feature Service (WFS) define um serviço para que clientes possam recuperar objetos (*features*) espaciais. WFS devolve o dado em formato GML.

Cliente

1: GetCapabilites →

← 2: *Layers (XML)*

3: DescribeFeatureType(Feição) →

← *4: Esquema GML da feição*

5: GetFeature(Feição) →

← *6: Feição* (GML)

Servidor

GML – Geography Markup Language:
especificação OGC para codificar
informação geográfica em XML

```
<distritos>
  <TeGeometry>
    <gml:Polygon srsName="EPSG:29193">
      <gml:outerBoundaryIs>
        <gml:LinearRing>
          <gml:coordinates>
            330221.3,7396108.7 ...
          </gml:coordinates>
        </gml:LinearRing>
      </gml:outerBoundaryIs>
    </gml:Polygon>
  </TeGeometry>
  <sprarea>3842344.0313</sprarea>
  <sprperimet>8576.6837</sprperimet>
  <sprrotulo>54</sprrotulo>
  <sprnome>54</sprnome>
  <id2>413</id2>
  <area>3852</area>
  <cod>70</cod>
  <sigla>SCE</sigla>
  <deno>SANTA CECILIA</deno>
  <object_id_7>53</object_id_7>
</distritos>
```

Figura: Gilberto Ribeiro

WFS – Exemplo de implementação

- Cliente: gvSIG
- Servidor: Servidor de dados ambientais DAS/CPTEC

1) Enviar ao servidor a requisição das suas capacidades

WFS – Exemplo de implementação

2) Receber a descrição das camadas

WFS – Exemplo de implementação

3) Descrever as feições

WFS – Exemplo de implementação

4) Receber e plotar os dados

Outros W*

- Existem muitos outros serviços padronizados pelo OGC. Exemplos:
  - WCS – Web Coverage Service: para disseminar dados com representação matricial (imagens, grades, etc.)
  - WPS – Web Processing Service: para executar processamentos sobre dados geográficos
  - CWS – Catalogue Web Service: para disseminar catálogos de metadados de repositórios de dados geográficos
  - WCTS – Web Coordinate Transformation Service: para transformação de coordenadas

OWS

- Existe uma longa lista de serviços e especificações OGC.O consórcio tem uma influência forte da indústria. Especificações são difíceis de entender. Existem problemas de versionamento, as especificações estão em constante mudança.
- Especificações tratam de generalidades. Existem casos onde o seu problema é específico e não existe uma especificação para esse problema. Ex. modelagem dinâmica LUCC.
- Tamanho do dado pode tornar o seu tráfego mais lento do que o cliente está disposto a esperar.
- Processamentos podem ser longos demais.
- Se o servidor sai do ar... o serviço sai do ar.

O que é preciso para entrar no mundo OWS

CONSUMIDOR

Usar SIG's que possam ser usados como clientes:

http://www.opengeospatial.org/resource/products/compliant

Ex: plugin WMS para o TerraView, para o UDIG, etc.

WMS, WFS: cada vez mais as ferramentas irão suportá-los, pois de fato, são serviços mais alto contidos.

WCS: ainda existem poucos produtos.

WPS: muito gerais, difícil ter um cliente para qualquer processamento. Clientes para domínios específicos são mais prováveis.

Outros serviços ainda estão em amadurecimento.

PROVEDOR

Ferramentas para construção de servidores:

http://www.opengeospatial.org/resource/products/compliant

Ex: TerraOGC para disponibilizar uma base TerraLib através de WMS e WFS; MapServer, etc.

WCS: envolve muito pré-processamento de dados. Existem ferramentas que facilitam isso, mas é uma longa receita de bolo para fazer um servidor funcionar corretamente (ex. GeoServer, Deegree).

WPS: dada sua característica geral, você vai precisar de programadores! Não existem ferramentas prontas...

Encadeamento de serviços

Figura: OGC

PACOTES DE SOFTWARE QUE FACILITAM O USO DE PADRÕES

MapServer
open source web mapping

- Plataforma para a publicação de dados espaciais e construção de aplicações baseadas em mapas no ambiente Web

MapServer
open source web mapping

```
MAP
    NAME "sample"
    STATUS ON
    SIZE 600 400
    SYMBOLSET "../etc/symbols.txt"
    EXTENT -180 -90 180 90
    UNITS DD
    SHAPEPATH "../data"
    IMAGECOLOR 255 255 255
    FONTSET "../etc/fonts.txt"

    #
    # Start of web interface definition
    #
    WEB
        IMAGEPATH "/ms4w/tmp/ms_tmp/"
        IMAGEURL "/ms_tmp/"
    END

    #
    # Start of layer definitions
    #
    LAYER
        NAME 'global-raster
        TYPE RASTER
        STATUS DEFAULT
        DATA bluemarble.gif
    END
END
```

É instalado em um servidor web e através de arquivos map, como o mostrado ao lado, são definidos quais dados são mostrados e como mostrá-los

MapServer
open source web mapping

```
LAYER
    NAME    "world_poly"
    DATA          'shapefile/countries_area.shp'
    STATUS        ON
    TYPE          POLYGON
    CLASS
        NAME        'The World'
        STYLE
            OUTLINECOLOR     0 0 0
        END
    END
END # layer
```

```
LAYER
  CONNECTIONTYPE postgis
  NAME "roads"
  CONNECTION "user=theuser password=thepass dbname=thedb host=theserver"
  DATA "the_geom from roads"
  STATUS ON
  TYPE LINE
  CLASS
    STYLE
      COLOR 0 0 0
    END
  END
END
```

Ex: Canasat usa MapServer

MapServer demo

- Para outros exemplos e demonstrações de MapServer:
  - http://demo.mapserver.org/

OpenLayers

Você já viu esse leiout?

OpenLayers

É uma biblioteca de software para a construção de aplicações web map baseada em JavaScripts no lado do cliente
Suporta vários padrões OGC e outros padrões abertos

Oferece:
Tiling
Zoom in/out
Panning
Zoom Box
…

OpenLayer: alguns exemplos

OpenLayers: WMS

```html
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0, maximum-scale=1.0, user-scalable=0">
    <meta name="apple-mobile-web-app-capable" content="yes">
    <title>OpenLayers Basic Single WMS Example</title>
    <link rel="stylesheet" href="../theme/default/style.css" type="text/css">
    <link rel="stylesheet" href="style.css" type="text/css">
    <script src="../OpenLayers.js"></script>
    <script type="text/javascript">
        var map, layer;
        function init(){
            map = new OpenLayers.Map( 'map' );
            layer = new OpenLayers.Layer.WMS( "OpenLayers WMS",
                    "http://vmap0.tiles.osgeo.org/wms/vmap0",
                    {layers: 'basic'} );
            map.addLayer(layer);
            map.zoomToMaxExtent();
        }
    </script>
  </head>
  <body onload="init()">
    <h1 id="title">Basic Single WMS Example</h1>

    <div id="tags">
        basic, simple, minimal, cleanup
    </div>

    <div id="shortdesc">Show a Simple Map</div>

    <div id="map" class="smallmap"></div>

    <div id="docs">
        <p>This example shows a very simple layout with minimal controls.
        This example uses a single WMS base layer.</p>
    </div>
  </body>
</html>
```

OpenLayers: WMS

OpenLayers: outros exemplos

- Para outros exemplos de uso do OpenLayers consulte o site:
  - http://openlayers.org/dev/examples

Globos Virtuais

- Representação 3D da Terra
- Permite que o usuário navegue livremente no ambiente virtual, mudando seu ângulo de visada e sua posição
- Podem apresentar diferentes visões da superfície
- Podem mostrar feições geográficas naturais e/ou feitas pelo homem

Globos Virtuais

Mapa 2D Tradicional

Globo Virtual 3D

KML e KMZ

- KML - Keyhole Markup Language é uma extensão XML para para descrever feições geográficas a serem visualizadas em globos virtuais. KMZ é a versão comprimida de um KML

```
<?xml version="1.0" encoding="UTF-8"?>
<kml xmlns="http://www.opengis.net/kml/2.2">
<Document>
<Placemark>
  <name>New York City</name>
  <description>New York City</description>
  <Point>
    <coordinates>-74.006393,40.714172,0</coordinates>
  </Point>
</Placemark>
</Document>
</kml>
```

Exemplo

- No site abaixo podem ser encontrados diversos exemplos de dados geográficos ambientais em formato KML:
  - http://maine.gov/dep/gis/datamaps/

Outros exemplos interessantes

- TerraBrasilis: http://www.terrabrasilis.info/
- VINDE: http://www.visualizador.inde.gov.br/

Sensor Web

- **Sensor Observation Service (SOS)**: a web service interface for requesting, filtering and retrieving sensor system information and observations.
- **Sensor Planning Service (SPS)**: a web service interface for requesting user-driven observations or data acquisitions.
- **Sensor Alert Service (SAS)**: a web service interface for advertising, publishing and subscribing to alerts from sensors.
- **Web Notification Service (WNS)**: a web service interface for registering single users or groups of users and sending notification messages to registered users via different types of communication protocols.
- **Sensor Model Language (SensorML)**: a metadata encoding for describing a functional model of a sensor system and related processes. The current version of SensorML also specifies the SWE Common Data Model.
- **Observations and Measurements Schema (O&M)**: for encoding sensor data objects.

Sensor Web - Example

SOS - Examples

http://smart.uah.edu/casestudy/
http://sensorweb.demo.52north.org/
http://matsu.opencloudconsortium.org/namibiaflood

Conclusões

- Desde o início dos anos 2000 tem surgido muita pesquisa e desenvolvimentos tecnológicos com o objetivo de aproveitar a internet no domínio geo-espacial
- O OGC tem um tido um papel importante no fomento do conceito de interoperabilidade nesse domínio
- As tecnologias (SIGs, bibliotecas, SGBD's) vem implementado e popularizando o uso da Internet nesse domínio

Programando SGBD’s

Lubia Vinhas

Interfaces SGBD

GUI

Prompt

API

```
int main()
{
...
}
```

SGBB

Interfaces SGBD

GUI

Prompt

API

```
int main()
{
...
}
```

SGBB

API para SGBD

- Uma API é uma biblioteca de código em alguma linguagem de programação de forma que você possa criar um programa que converse com o banco
- Pra isso você precisa de um ambiente de edição, compilação e linking para aquela linguagem

*Open Database Commun*ity (ODBC)

Java Database Connectivity (JDBC)

Exemplos

SQLite

Small. Fast. Reliable.
Choose any three.

About Sitemap Documentation Download License News Support

Search SQLite Docs... Go

An Introduction To The SQLite C/C++ Interface

This article provides an overview and roadmap to the C/C++ interface to SQLite.

PostgreSQL

Search Docu

Home → Documentation → Manuals → PostgreSQL 8.2

PostgreSQL 8.2.21 Documentation

Prev Fast Backward

Chapter 29. libpq - C Library

MySQL

The world's most popular open source database

Developer Zone Downloads Documentation

MySQL Server MySQL Enterprise MySQL Workbench MySQL Cluster Topic Guid
Archives About

MySQL 5.0 Reference Manual :: 19 Connectors and APIs

Documentation Library

Table of Contents

MySQL 5.6 Manual

MySQL 5.5 Manual

MySQL 5.1 Manual

MySQL 5.0 Manual

MySQL 3.23/4.0/4.1 Manual

Search manual:

Go

Chapter 19. Connectors and APIs

Table of Contents [+/-]

19.1. MySQL Connector/ODBC [+/-]
19.2. MySQL Connector/Net [+/-]
19.3. MySQL Connector/J [+/-]
19.4. MySQL Connector/MXJ [+/-]
19.5. MySQL Connector/C [+/-]
19.6. MySQL Connector/OpenOffice.org [+/-]
19.7. libmysqld, the Embedded MySQL Server Library
19.8. MySQL C API [+/-]
19.9. MySQL PHP API [+/-]
19.10. MySQL Perl API
19.11. MySQL Python API
19.12. MySQL Ruby APIs [+/-]
19.13. MySQL Tcl API
19.14. MySQL Eiffel Wrapper

Ambiente de programação

Ambiente:
editor,
compilador,
linker.
...

APIs para SGBD

- Objetos: quais são os objetos que representam os diferentes componentes do SBGD

- Funções: quais as funções sobre esses objetos

Exemplo: SQLite

- Objetos:

  `sqlite3` : representa uma conexão ao SGBD

  `sqlite3_stmt`: representa um comando a ser submetido ao SGBD

- Funções:

  `sqlite3_open()`
  `sqlite3_prepare()`
  `sqlite3_step()`
  `sqlite3_column()`
  `sqlite3_finalize()`
  `sqlite3_close()`

Exemplo: SQLite e C++

```c
#include <stdio.h>
#include <sqlite3.h>

int main(int argc, char* argv[])
{
   sqlite3 *db;
   char *zErrMsg = 0;
   int rc;

   rc = sqlite3_open("test.db", &db);

   if( rc ){
      fprintf(stderr, "Can't open database: %s\n", sqlite3_errmsg(db));
      exit(0);
   }else{
      fprintf(stderr, "Opened database successfully\n");
   }
   sqlite3_close(db);
}
```

```
$gcc test.c -l sqlite3
$./a.out
Opened database successfully
```

Exemplo: SQLite e JAVA

```
import java.sql.*;

public class SQLiteJDBC
{
  public static void main( String args[] )
  {
    Connection c = null;
    try {
      Class.forName("org.sqlite.JDBC");
      c = DriverManager.getConnection("jdbc:sqlite:test.db");
    } catch ( Exception e ) {
      System.err.println( e.getClass().getName() + ": " + e.getMessage() );
      System.exit(0);
    }
    System.out.println("Opened database successfully");
  }
}
```

```
$javac SQLiteJDBC.java
$java -classpath ".:sqlite-jdbc-3.7.2.jar" SQLiteJDBC
Open database successfully
```

Exemplo: SQLite e PHP

```php
<?php
   class MyDB extends SQLite3
   {
      function __construct()
      {
         $this->open('test.db');
      }
   }
   $db = new MyDB();
   if(!$db){
      echo $db->lastErrorMsg();
   } else {
      echo "Opened database successfully\n";
   }
?>
```

Exemplo: SQLite e PHP

```php
<?php
   class MyDB extends SQLite3
   {
      function __construct()
      {
         $this->open('test.db');
      }
   }
   $db = new MyDB();
   if(!$db){
      echo $db->lastErrorMsg();
   } else {
      echo "Opened database successfully\n";
   }
?>
```

Exemplo: SQLite e Python

```
#!/usr/bin/python

import sqlite3

conn = sqlite3.connect('test.db')

print "Opened database successfully";
```

```
$chmod +x sqlite.py
$./sqlite.py
Open database successfully
```

```
  try {
    /*
    * Load the JDBC driver and establish a connection.
    */
    Class.forName("org.postgresql.Driver");
    String url = "jdbc:postgresql://localhost:5432/database";
    conn = DriverManager.getConnection(url, "postgres", "");
    /*
    * Add the geometry types to the connection. Note that you
    * must cast the connection to the pgsql-specific connection
    * implementation before calling the addDataType() method.
    */
    ((org.postgresql.PGConnection)conn).addDataType("geometry",Class.forName("org.postgis.PGgeometry"));
    ((org.postgresql.PGConnection)conn).addDataType("box3d",Class.forName("org.postgis.PGbox3d"));

    /*
    * Create a statement and execute a select query.
    */
    Statement s = conn.createStatement();
    ResultSet r = s.executeQuery("select geom,id from geomtable");
    while( r.next() ) {
      /*
      * Retrieve the geometry as an object then cast it to the geometry type.
      * Print things out.
      */
      PGgeometry geom = (PGgeometry)r.getObject(1);
      int id = r.getInt(2);
      System.out.println("Row " + id + ":");
      System.out.println(geom.toString());
    }
    s.close();
    conn.close();
  }
catch( Exception e ) {
  e.printStackTrace();
```

Database Software

[Yeung and Hall, 2011]

Database Software

[Yeung and Hall, 2011]

Digital Terrain Modeling

Laércio Namikawa

INPE - DPI

Topics

Definitions

Sources

Models

Products

What are Elevation Models

Computational Representation of the distribution of a geographic phenomenon

Phenomenon: Terrain Elevation

Z
Y
X

Naming
DTM, DSM, DEM
TIN

M: Model, Modeling

D: Digital

T: Terrain

E: Elevation

S: Surface

Most correct - DEM

Modeling
Digital Elevation Model

Elevation in the Real World

Ayers, H. B. (2000). Local Geoid Models for 3-D Helmert Transformations, Leica Geosystems Ltd

Metadata

SRTM

ARIA

DESCRIPTION "Elevation from SRTM, reprojected to UTM, 90-meter resolution"
MEASURE_DATETIME "2000/02/1 0:0:0"
PRODUCTION_DATETIME "2006/01/26 0:0:0"
PRODUCTION_METHOD "Linear Interpolation"
PRODUCTION_SOURCE “SRTM 3arc-sec”
SOURCE_SCALE "25000"
MEASURE_UNIT "Meters"
MEASURE_NAME "Elevation"
HORIZONTAL_ACCURACY "5"
VERTICAL_ACCURACY "16"

DESCRIPTION "Elevation from Arizona Image Archive - ARIA. Source data is probable one, from INEGI topographic maps"
MEASURE_DATETIME "1995/12/01 0:0:0"
PRODUCTION_DATETIME "2006/01/26 0:0:0"
PRODUCTION_METHOD " Interpolation"
PRODUCTION_SOURCE “Contour Lines”
SOURCE_SCALE "50000"
MEASURE_UNIT "Meters"
MEASURE_NAME "Elevation"
HORIZONTAL_ACCURACY "20"
VERTICAL_ACCURACY "15"

Data Sources

Field Surveys

Photogrammetry – Stereoscopic analysis of images

Cartographic sources – Contour lines and profiles

Radar Interferometry - Synthetic Aperture Radar Interferometry (InSAR) – Aerial, Satellite, Space Shuttle (SRTM)

LIDAR - Light detecting and ranging – Aerial, Satelite

GPS - Global Positioning System

Photogrammetry

Geometria de Aquisição

Deslocamento Devido ao Relevo

$\Delta h = (d\ H)/r$

d = image displacement
r = radial distance from the principal point to the image point
H = flying height above ground

Parallax Equations

$p_a = x_a - x'_a$

$h_A = H - B.f/p_a$

$X_A = B.x_a/p_a$

$Y_A = B.y_a/p_a$

where

p_a : parallax of point A

h_A : elevation of point A above vertical datum

H : flying height above vertical datum

B : distance between the exposure stations

f : focal length of the camera

X_A, Y_A : ground coordinates of point A in the XY coordinate system with origin at point P on vertical datum of the left photo. X axis is in same vertical plane as x and x' flight axes and Y axis passes through P and is perpendicular to the X axis

x_a, y_a : photo coordinates of point a measured with respect to the flight line axes on the left photo

FIGURE 8-11
Geometry of an overlapping pair of vertical photographs.

InSAR
SRTM

Figure 4: Differential Distance Gives Topography

A_1 *Antenna 1*
A_2 *Antenna 2*
B *Baseline*

Equações básicas

Diferença de fase: $$\Delta\phi = \frac{4\pi\Delta r}{\lambda}$$

Relação trigonométrica: $$sin(\alpha-\theta) = \frac{(r+\Delta_r)-r^2-B^2}{2rB}$$

Cálculo de altura: $$z = H - r\cos\theta$$

LIDAR

Aircraft
Platform
Sweep
angle
two-way
slant range
Laser
Pulse
Ground Surface

500m x 500m ~ 1,000,000 points

GPS

INPE/CPTEC G-8 IR 2003/03/25 1500Z

1. All satellites have clocks set to exactly the same time
2. All satellites know their exact position from data sent to them from the system controllers
3. Each satellites transmits its position and a time signal
time and orbit position
4. The signals travel to the receiver delayed by distance traveled
5. The differences in distance traveled make each satellite appear to have a different time
6. The receiver calculates the distance to each satellite and can then calculate its own position

Data Collection

- Precision
  - Measurement instrument
- Accuracy
  - Instrument calibration
- Validity
  - Correctly executed measurement
- Reliability
  - Measurement is repeatable

Digital Model Constraints

Finite:

specification of the expected bounds for the data.
amount of data: storage capacity and processing power.

Discrete (integer):

the smallest difference between values that can be discerned

Define

minimum distance between different locations,

minimum difference in value,

maximum size of region under study,

maximum amount of data

Data Quality

Quality is a generic term that depends on the context

- In geographic representation context, more difficult to define given that physical characteristics of geographical reality can not be directly assessed

Related to accuracy, precision, consistency, and completeness of the representation.

Elevation representation:

- accuracy is defined as the measure of its quality.

Accuracy

Measure of how different the representation is in relation to the real world entity

Metrics

- Root Mean Square Error (RMSE) and the vertical accuracy at 95% confidence

DEM accuracy is difficult to be assessed since there is no independent model of the real world to test our digital model against

"True value" of elevation is just a representation that is considered to have higher accuracy than the one having its accuracy defined.

Accuracy of DEMs is dependent on the Earth's surface characteristics and the measurement techniques

- Errors are not randomly distributed over the entire DEM

Quality Factors

Measurement Instrument

Data Processing

Digital Representation

Algorithms

Error Sources

Ground Surveys – Depend on instruments and methods - highly accurate, surveyor is at the field.

Photogrammetric Data Capture – Paralax principle – Stereocorrelation. Accuracy dependent on photograph scale, image resolution, discernibility of image features. Correlation window (10x10 pixels)

Error Sources

Photogrammetric Data Capture

Error Sources

Cartographic Sources – Dependent on original data, usually photogrammetric, equipment/operator lag, analog recording precision, digitalization or scanning resolution, map resolution →0.4 mm of scale.

InSAR – Ground Range Resolution, layover/shadowing effects, speckle noise, vegetation depending on band.

Error Sources

InSAR

Error Sources

LIDAR

Pulse width, atmospheric effects, vegetation influence, footprint size (typical laser beam projects to 24–60 cm diameter at a distance of 1219 m)

Accuracy:

USGS DEM

Level-1 DEM reserved for ones created by scanning National High Altitude Photography (NHAP)/NAPP photography.

- Vertical RMSE of 7 meters is the desired standard. A RMSE of 15 meters is the maximum permitted.

Level-2 DEM data sets have been processed or smoothed for consistency and edited to remove identifiable systematic errors and were derived from hyposographic and hydrographic data digitizing.

- RMSE of one-half contour interval is the maximum permitted.

Level-3 DEMs are derived from DLG data by incorporating selected elements from both hypsography (contours, spot elevations) and hydrography (lakes, shorelines, drainage).

- RMSE of one-third of the contour interval is the maximum permitted.

RMSE error is calculated on 27 sample points

- any 27 points distributed on the area.

Accuracy:
ASTER DEM

Advanced Spaceborne Thermal Emission and Reflection Radiometer

Generated from along-track stereo

Grid 30 meter

Estimated accuracy:

- Relative vertical accuracy between ±12 and 25 meters

Accuracy:
SRTM C-Band DEM

Absolute

- 16 meters vertical 90% linear error (LE90)
- 20 meters horizontal 90% circular error (CE90)

Relative

- 10 meters vertical LE90
- 15 meters horizontal CE90

Accuracy:
LIDAR

ICESat

- 15 cm
- footprint: 60 m diameter

Saab TopEye system for of bare soil and low grass

- Between 10 and 16 cm RMSE

DTM Representations

Contour Lines

Regular Grid

TIN – Triangulated Irregular Network

Contour Lines

Define the surface only along the lines

Regions between two lines are inferred to be between the values of the lines

Contour Lines

Regular Grid

Triangulated Irregular Network

DTM Products

Generated from the models

Derivatives

Contour Lines

Profile

3D Visualization

Drainage Analysis

DTM Visualization

Slope Angle and Aspect

$$S = \mathrm{arctg}\,\{[(\delta Z/\delta X)^2+(\delta Z/\delta Y)^2]^{1/2}\}$$

$$A = \mathrm{arctg}\,[-(\delta Z/\delta Y)/(\delta Z/\delta X)]\ (-\Pi < A < \Pi)$$

$\delta Z/\delta X$ and $\delta Z/\delta Y$ are the partial derivatives in X and Y directions

Derivatives Estimation

The partial derivatives are estimated using:

$$[\delta Z/\delta X]_{i,j} = [(Z_{i+1,j+1} + 2*Z_{i+1,j} + Z_{i+1,j-1}) - (Z_{i-1,j+1} + 2*Z_{i-1,j} + Z_{i-1,j-1})]/8*\delta X$$

$$[\delta Z/\delta Y]_{i,j} = [(Z_{i+1,j+1} + 2*Z_{i,j+1} + Z_{i-1,j+1}) - (Z_{i+1,j+1} + 2*Z_{i,j-1} + Z_{i-1,j-1})]/8*\delta Y$$

where z elements are on the grid, indicated by:

$Z_{i-1,j+1}$	$Z_{i,j+1}$	$Z_{i+1,j+1}$
$Z_{i-1,j}$	$Z_{i,j}$	$Z_{i+1,j}$
$Z_{i-1,j-1}$	$Z_{i,j-1}$	$Z_{i+1,j-1}$

Elevation Derivatives

Márcio de Morisson Valeriano

Third-Order Geomorphometric Derivatives

Jozef Minár , Marián Jenčo , Ian S. Evans , Jozef Minár Jr. , Martin Kadlec , Jozef Krcho , Jan Pacina , Libor Burian & Alexandra Benová (2013): Third-order geomorphometric variables (derivatives): definition, computation and utilization of changes of curvatures, International Journal of Geographical Information Science, DOI:10.1080/13658816.2013.792113

Slope Angle

SPRING-5.2.3 [IntroRS][Botucatu]
File Edit View Image Thematic DTM Cadastral Network Analysis SCarta Apply Tools TerraLib Plugins Help
Auto
1/36.726562
Disable
Control Panel
Active Screen : Assistant
Available Infolayers
Selecteds Infolayers
Category / Infolayer
( ) CAT_DTM
( ) SRTM90
( ) Drainage
( ) LandUse
( ) Landsat
( ) ParcelsMap
(V) Slope
(Im) SlopePerc
( ) Aspect
( ) Transportation
Samples
Isolines
Grid
Text
TIN
Image
Main
Assistant
Screen 2
Screen 3
Screen 4
IL: SlopePerc

Aspect

SPRING-5.2.3 [IntroRS][Botucatu]
File Edit View Image Thematic DTM Cadastral Network Analysis SCarta Apply Tools TerraLib Plugins Help
Auto
1/36.726562
Disable
Control Panel
Active Screen : Assistant
Available Infolayers
Selecteds Infolayers
Category / Infolayer
() CAT_DTM
() SRTM90
() Drainage
() LandUse
() Landsat
() ParcelsMap
(V) Slope
() SlopePerc
(Im) Aspect
() Transportation
Samples
Isolines
Grid
Text
TIN
Image
Main
Assistant
Screen 2
Screen 3
Screen 4

Horizontal Curvature

Contour Lines

3D Visualization

Profile

Edit View Image Thematic DTM Cadastral Network Analysis SCarta Apply Tools TerraLib Plugins Help
Auto
1/36.726562
Disable
trol Panel
Active Screen : Assistant
Available Infolayers
Selecteds Infolayers
Category / Infolayer
(V) CAT_DTM
(I) SRTM90
() Drainage
() LandUse
() Landsat
() ParcelsMap
() Slope
() SlopePerc
() Aspect
() Transportation
Profile
Input: Grid TIN
Trajectory: Editor PI
Lines: Create Delete
Points: Add Move Delete
Graphic Title: Profile
Y Axis: Quota
Unit: m
Apply
Close
Help
Gráfico
Profile
Quota m
850
800
700
600
500
440
0
2500
5000
7500
10000
12500
15000
Distância

Drainage Analysis

Based on D8 Method

120	150	170
80	100	150
150	160	90

DTM Hydrologically Corrected

60.7	62.4	60.6	58.6	54.5	51.2	54.3
62.4	60.2	57.1	58.3	57.7	56.0	56.7
64.1	61.0	57.5	55.5	55.9	59.5	59.1
66.7	63.6	58.4	53.1	57.8	62.5	63.9
69.3	64.2	58.2	63.1	60.3	65.1	69.8
67.3	65.3	66.1	65.0	68.6	70.8	73.7
70.9	71.3	74.5	72.1	76.0	74.2	75.0

“sink”

SRTM
“sinks”

Hydrologically Correct DTM

60.7	62.4	60.6	58.6	54.5	51.2	54.3
62.4	60.2	57.1	58.3	57.7	56.0	56.7
64.1	61.0	57.5	55.5	55.9	59.5	59.1
66.7	63.6	58.4	53.1	57.8	62.5	63.9
69.3	64.2	58.2	63.1	60.3	65.1	69.8
67.3	65.3	66.1	65.0	68.6	70.8	73.7
70.9	71.3	74.5	72.1	76.0	74.2	75.0

60.7	62.4	60.6	58.6	54.5	51.2	54.3
62.4	60.2	57.1	58.3	57.7	56.0	56.7
64.1	61.0	57.5	55.5	55.9	59.5	59.1
66.7	63.6	58.4	53.1	57.8	62.5	63.9
69.3	64.2	58.2	63.1	60.3	65.1	69.8
67.3	65.3	66.1	65.0	68.6	70.8	73.7
70.9	71.3	74.5	72.1	76.0	74.2	75.0

...

60.7	62.4	60.6	58.6	54.5	51.2	54.3
62.4	60.2	57.1	58.3	57.7	56.0	56.7
64.1	61.0	57.5	55.5	55.9	59.5	59.1
66.7	63.6	58.4	53.1	57.8	62.5	63.9
69.3	64.2	58.2	63.1	60.3	65.1	69.8
67.3	65.3	66.1	65.0	68.6	70.8	73.7
70.9	71.3	74.5	72.1	76.0	74.2	75.0

60.7	62.4	60.6	58.6	54.5	51.2	54.3
62.4	60.2	57.1	58.3	57.7	56.0	56.7
64.1	61.0	57.5	55.5	55.9	59.5	59.1
66.7	63.6	58.4	53.1	57.8	62.5	63.9
69.3	64.2	58.2	63.1	60.3	65.1	69.8
67.3	65.3	66.1	65.0	68.6	70.8	73.7
70.9	71.3	74.5	72.1	76.0	74.2	75.0

60.7	62.4	60.6	58.6	54.5	51.2	54.3
62.4	60.2	57.1	58.3	57.7	51.7	56.7
64.1	61.0	57.5	55.5	52.4	59.5	59.1
66.7	63.6	58.4	53.1	57.8	62.5	63.9
69.3	64.2	58.2	63.1	60.3	65.1	69.8
67.3	65.3	66.1	65.0	68.6	70.8	73.7
70.9	71.3	74.5	72.1	76.0	74.2	75.0

CAMILO DALELES RENNÓ

Drainage Network

CAMILO DALELES RENNÓ

Drainage From Curvature

Comparison: SRTM HAND and Flood Extents from Landsat 8 Imagery

HAND using 500K accumulated Cells to define drainage network

Landsat8: Acquired 2014/July/05

Flood Level 13 meters

Maximum level: 15 m on July 2nd

(R) LC82240802014186LGN00_B4
(G) LC82240802014186LGN00_B5
(B) LC82240802014186LGN00_B6

Comparison: SRTM HAND and Flood Extents from Landsat 8 Imagery

HAND using 100K accumulated Cells to define drainage network

Landsat8: Acquired 2014/July/05

Flood Level 13 meters

Maximum level: 15 m on July 2nd

Comparison: SRTM HAND and Flood Extents from Landsat 8 Imagery

HAND using 1000 accumulated Cells to define drainage network

Landsat8: Acquired 2014/July/05

Flood Level 13 meters

Maximum level: 15 m on July 2nd

Ortorretificação

Referências

Livros

Maune, D. F., ed. (2001). Digital Elevation Model Technologies and Applications: The DEM Users Manual. Bethesda: American Society for Photogrammetry and Remote Sensing

El-Sheimy, N., C. Valeo and A. Habib (2005). Digital Terrain Modeling: Acquisition, Manipulation And Applications. Norwood: Artech House Publishers

Li, Z., Q. Zhu and C. Gold (2005). Digital Terrain Modeling: Principles and Methodology. Boca Raton: CRC Press.

Preparata, F.P.; Shamos M.I. Computational geometry. New York, Springer-Verlag, 1985. 398 p.

http://wiki.dpi.inpe.br/doku.php?id=elevacao

Sistemas de Bancos de Dados Não-Relacionais

Sistemas NoSQL, NewSQL e MapReduce

Gilberto Ribeiro de Queiroz
(gribeiro@dpi.inpe.br)

Stonebraker: *"One Size Fits All": An Idea Whose Time Has Come and Gone* (2005)

Evolução das Tecnologias de Bancos Dados

Evolução das Tecnologias de Bancos Dados

Fonte: Wikipedia

Programação Generalizada

Programas dependentes arquivos

Modelo Banco Dados Hierárquico (IBM IMS)

1960 1970 1980 1990 2000 2010

Evolução das Tecnologias de Bancos Dados

Evolução das Tecnologias de Bancos Dados

Quais são os principais conceitos em bancos de dados relacionais?

Relação (ou Tabela)

- Um banco de dados relacional é organizado em uma coleção de relações (ou tabelas) possivelmente relacionadas entre si.

Tabela

Colunas

Linha

paises			
id	**nome**	**populacao**	**fronteira**
1	Alemanha	82.000.000	
2	Brasil	190.000.000	
...	...	...	...

Esquema Tabela

Instância

Álgebra Relacional: Operadores

Fonte: C. J. Date (1993)

Linguagem de Consulta: SQL

- O modelo relacional (Codd, 1970) é a base para linguagens de alto nível:
  - Álgebra Relacional → Linguagem Declarativa → ISO/SQL (Structured Query Language)

```
CREATE TABLE paises
(
  id          INT4 PRIMARY KEY,
  nome        VARCHAR(50),
  populacao   INT4
);
```

Definição Dados →

paises		
id	nome	populacao

↓ **Manipulação Dados**

```
INSERT INTO paises
      VALUES (1, 'Alemanha', 82000000)

INSERT INTO paises
      VALUES (2, 'Brasil', 190000000)
```

← **Manipulação Dados**

paises		
id	nome	populacao
1	Alemanha	82.000.000
2	Brasil	190.000.000
...	...	...

Linguagem de Consulta: SQL

- O modelo relacional (Codd, 1970) é a base para linguagens de alto nível:
  - Álgebra Relacional → Linguagem Declarativa → ISO/SQL (Structured Query Language)

paises		
id	nome	populacao
1	Alemanha	82.000.000
2	Brasil	190.000.000
...	...	...

Consulta (Não-Procedural) →

```
SELECT nome
 FROM paises
WHERE populacao > 80000000
```

Nota: stored procedures ou procedural languages: PL/SQL, T-SQL, PL/pgSQL

Métodos de Acesso (Indexação)

- Problema: Como processar de forma eficiente as consultas?
  - Através do uso de estruturas de dados conhecidas como Índices ou Métodos de Acesso;

- Os índices reduzem o conjunto de objetos a serem verificados durante o processamento das consultas:
  - Normalmente, uma consulta envolve apenas uma pequena parcela do banco de dados;
  - Neste caso, percorrer todo o banco pode ser bastante ineficiente;
  - Portanto, um plano de execução eficiente para a consulta tipicamente considera a existência de índices.

Registros de um arquivo e o índice associado a este arquivo

Independência Física dos Dados

Fonte: Adaptado de Gray (1996)

Visões (Views)

- Muitas vezes pode ser necessário fornecer diferentes perspectivas do banco de dados dependendo do usuário. Uma visão (ou view) pode ser:
  - um subconjunto dos dados do banco de dados
  - pode conter dados derivados do banco de dados

Visões (Dado Virtual)

Tabelas (Dado Bruto)

Arquiteturas de SGBD-R

- Cliente/Servidor ou Embutido (ou embarcado)
- Em memória (In-memory)
- Paralelos ou Distribuídos
- Armazenamento Linha x Coluna:
  - linha: atributos do registro são colocados contiguamente no meio de armazenamento. (bom para aplicações OLTP)
  - coluna: todos os valores de uma coluna são armazenados contiguamente. (bom para aplicações otimizadas para leitura tais como data warehouses e OLAP)

Evolução das Tecnologias de Bancos Dados

SGBD Objeto-Relacionais

```
CREATE TYPE Address AS
(
  street    VARCHAR(50),
  city      VARCHAR(50),
  zip       CHAR(8)
);
```

```
CREATE TABLE Student
(
  name    VARCHAR(50),
  address  Address
);
```

```
CREATE TYPE Geometry
(
internallength = variable,
input = geometry_in,
output = geometry_out,
send = geometry_send,
receive = geometry_recv,
typmod_in = geometry_typmod_in,
typmod_out = geometry_typmod_out,
delimiter = ':',
alignment = double,
analyze = geometry_analyze,
storage = main);
```

```
CREATE FUNCTION ValidAddress RETURNS BOOLEAN
 EXTERNAL NAME '/address-module.so' LANGUAGE 'C';
```

```
INSERT INTO student VALUES('eduardo', ('albino sartori', 'ouro preto', '35400'));
SELECT (address).city FROM student;
```

Evolução das Tecnologias de Bancos Dados

PostgreSQL → PostGIS
MySQL → Spatial Types
SQLite → SpatiaLite and RasterLite
Oracle → Oracle Spatial, GeoRaster, Topology and Network Models
IBM DB2 → Spatial Extender
SQL Server (2008) → Spatial Types

Stonebraker: *"One Size Fits All": An Idea Whose Time Has Come and Gone* (2005)

Evolução das Tecnologias de Bancos Dados

Interessante: o número de tecnologias de bancos de dados com características diferentes dos SGBD-R tem aumentado nos últimos 8 anos!

O “cardápio” de opções aumentou?

- *Sistemas Não-Relacionais* ou *Not Only SQL* ou *Pós-relacionais*:
  - http://nosql-database.org/
  - https://en.wikipedia.org/wiki/NoSQL

- Diferentes modelos de dados:
  - Document Oriented: MongoDB, CouchDB;
  - Column Stores: Cassandra;
  - Graph Databases: OrientDB, Neo4J;
  - Array Databases: SciDB, Rasdaman.

- Nem todos são baseados no paradigma de transações ACID.

- Escalabilidade: Horizontal x Vertical

“Novas” tecnologias de bancos de dados

Document Stores

“JSON Landscape”

Document Oriented

SGBD-R

todas as linhas compartilham o mesmo schema

Tabela: estudantes

#id	nome	sexo	idade
0	Eduardo	M	33
1	Emiliano	M	35
2	Julia	F	18

Document Store: documentos JSON

schemaless

```
{
    "nome": "Eduardo Queiroz",
    "sexo": "masculino",
    "idade": 33 ,
    "local-nascimento": "Ouro Preto"
}
```

```
{
    "nome": "Emiliano Castejon",
    "sexo": "male",
    "idade": 35,
    "endereço": { … }
}
```

Document Oriented x Relational: Schemaless

```
{
 "name": "Gilberto",
 "addresses":
    [{
      "type": "residential"
      "street": "Cidade Jardim",
      "city": "S. J. Campos",
      "state": "SP"
     },
     {
      "type": "professional"
      "street": "Astronautas",
      "city": "S. J. Campos",
      "state": "SP"
     }
    ]
}
```

Table: customers

#id	name	...
0	Gilberto	..
...	...	...

Table: adresses

#id	type	street	...
0	residential	Cidade Jardim	...
0	professional	Astronautas	...
...	...	...	...

more natural?

avoid data repetition?

Document Oriented

- The basic unit of work is the document
  - in general they work with some kind of JSON notation
- Schemaless:
  - the documents may not share a global schema
  - this is one of the greatest benefits of a document database:
    - the ability to change the format of documents stored in the database at any time without requiring a costly schema update
- Ability to replicate data between nodes
- Ideal for applications that need eventual or relaxed consistency
- Most widely used systems: CouchDB and MongoDB

CouchDB

- Based on a REST API:

curl -X GET http://localhost:5984/geodb/_all_docs


```
{
 "total_rows":3,
 "rows":[{"id":"0","key":"0","value":{"rev":"2-0000099af25ecf6100000000000000000"}},
          {"id":"1","key":"1","value":{"rev":"2-000009a5c3ff15e400000000000000000"}},
          {"id":"2","key":"2","value":{"rev":"2-000009b5e30f20c400000000000000000"}}
         ]
}
```


curl -X GET http://localhost:5984/geodb/0


```
{
 "name":"Eduardo Queiroz",
 "gender":"male",
 "age":33
}
```


* for Couchbase you should use port 8092

CouchDB

- At its core there is a B-tree and then all data retrieval is key based:

  curl -X GET http://localhost:5984/geodb/_all_docs

  curl -X GET http://localhost:5984/geodb/0

  curl -X GET http://localhost:5984/geodb/_all_docs?descending=true

  curl -X GET http://localhost:5984/geodb/_all_docs?key=\"2\"

  curl -X GET http://localhost:5984/geodb/_all_docs?startkey=\“1\"

  curl -X GET http://localhost:5984/geodb/_all_docs?startkey=\“1\“;endkey=\”2\”

CouchDB

- Views:

  Enables to index and query data based on Map/Reduce functions written in JavaScript

CouchDB

- Views:
  - Enables to index and query data based on Map/Reduce functions written in JavaScript
  - Generate a list of documents
  - Updated when a document changes

Map

```
function(doc) {
  if(doc.gender == "male") {
    emit(doc.age, doc.age);
  }
}
```

Index → Key → age
Output data → doc

Reduce

```
function(keys, values, rereduce) {
  return sum(values);
}
```

Allows to generate aggregates

CouchDB

- How Map/Reduce functions work?
  - The map function outputs (or emits) : (key, id) and doc-value
  - The reduce function receives:
    - ([[key1, id1], [key2, id2], [key3, id3]], [value1, value2, value3], false) or,
    - (null, [reduceResult, reduceResult, reduceResult], true)

Map

```
function(doc) {
  if(doc.gender == "male") {
    emit(doc.name, doc.age);
  }
}
```

Index → Key → age
Output data → doc

Reduce

```
function(keys, values, rereduce) {
  return sum(values);
}
```

Allows to generate aggregates

CouchDB

Querying through the REST API:

*GET http://localhost:5984/dbname/**_design**/designdocname/**_view**/viewname*

Some supported arguments for querying a view

Parameter	Value	Description
descending	-	return the docs in descendent key order
key	key-value	return only docs that match the specified key
keys	array of key-values	return docs that match the specified array of keys
limit	number	limit the number of returned docs
skip	number	
startkey	key-value	return docs with key greater or equal to the value
endkey	key-value	stop returning docs with key greater than the value

CouchDB

- Querying a view that sums all of the ages:

```
curl -X GET http://127.0.0.1:5984/geodb/_design/myviews/_view/sum_ages
```

GeoCouch

- It is an spatial extension for CouchDB
- Site: https://github.com/couchbase/geocouch
- The spatial functionality is base on GeoJSON
- It uses an R-tree to index spatial data
- Spatial views are limited to map functions
- GDAL/OGR has a driver for it

GeoCouch

Map

```
function(doc) {
 if(doc.geometry) {
  emit(doc.geometry, { "image" : doc.image_url});
 }
}
```

Key → geometry
value → an image URL

Querying through the REST API:

http://localhost:5984/geodb/_design/fotos/_spatial/fotos?bbox=-180,-90,180,90

query result


```
{"total_rows":0,
 "rows":[
    {"id":"8758094944",
     "bbox":[16.95973,51.106573,16.95973,51.106573],
     "geometry":{"type":"Point", "coordinates":[16.95973,51.106573]},
     "value":{"image":"http://farm3.static.flickr.com/2875/8758094944_a4e7a01fa9_s.jpg"}},
     ...
    ]}
```

CouchDB Implementations

- Apache CouchDB:
  - Site: http://couchdb.apache.org
  - Web admin tool (Futon): http:// 127.0.0.1:5984/_utils/

- Couchbase
  - Merge of CouchDB and Membase
  - Site: http://www.couchbase.com
  - Community Edition x Enterprise Edition
  - Version used: 2.0.1
  - Installing a client library: *pip install couchbase*
  - Admin: http://localhost:8091/index.html
  - Default port: 8092
  - GeoCouch is already present

CouchDB Hosting

- Cloudant (http://cloudant.com):
  - limited size datasets and limited number of requests can be hosted for free
  - Big Couch: http://cloudant.com/solutions/bigcouch

- Iris Couch:
  - Couchbase

MongoDB

- It is an open source document oriented database available for several platforms: Linux, Mac OS X and Microsoft Windows.
- Designed from the ground to be scalable and to provide a dynamic schema.
- Site: http://www.mongodb.org
- Used version: 2.4.3
- The source code is written in C++:
  https://github.com/mongodb/mongo

Document Oriented?

- Let's think in MongoDB as a JSON database:
  - as it fundamentally works with JSON objects;
  - although internally it uses a representation called BSON (Binary JSON);

- **Row → Document**:
  - A set of keys with associated values;
  - A single record can be used to represent a complex hierarchical relationship (*nested objects and arrays*) → remember the impedance mismatch?
  - Key and Values don't have fixed types or sizes;
  - No fixed Schema → adding or removing fields become simpler.

- **Collections → Tables**
  - Dynamic schemas.

Embedded Objects and Arrays

- In relational database systems we are used to normalizing the data, that is, create a table schema to accommodate lists of values or to avoid data duplication.

- In a document-oriented database like MongoDB, it is natural to create nested objects and arrays.

```
{
 "name" : "Gilberto",
 "email" : "gribeiro@dpi.inpe.br",
 "addresses" : [{"type" : "professional",
                  "street" : "Av. Astronautas"},
                 {"type" : "residential",
                  "street" : "Av. Cidade Jardim" }],
 "following" : ["Camara", "Miguel"]
};
```

Each Key Has a Data Type

- Common JSON data types:

Number (integer or float/real):

```
{ "int-key" : 10 }
```

or

```
{ "real-key" : 15.2 }
```

internally it is used 64-bit float numbers for both cases.

String (UTF-8):

```
{ "string-key" : "any text value" }
```

Each Key Has a Data Type

- Common JSON data types:

Boolean:

```
{ "boolean-key" : true }
```

or

```
{ "boolean-key" : false }
```

Null:

can be used to represent null values and also non-existing fields.

```
{ "null-key" : null }
```

Each Key Has a Data Type

- Common JSON data types:

Objects:

A document is an object and as in JSON it can have nested objects.

```
{
 "name" : "Gilberto",
 "age" : 36,
 "address" : {
              "street" : "Cidade Jardim",
              "zip-code" : "12233-002"
             }
}
```

Arrays:

although not usual, it can have mix data types.

```
{ "phones" : ["8123-0179", "3933-7794"] }
```

can be used to represent sets/lists of objects, numbers, strings, boolean values and any other allowed data type

Each Key Has a Data Type

- MongoDB specific data types:

stored as milliseconds since the epoch.

Date:

```
{ "last-login" : new Date() }
```

or

```
{ "birthday" : ISODate("1976-12-30"); }
```

Object Identifier:

```
{ "user-id" : ObjectId() }
```

or

```
{ "user-id" : ObjectId("51ad29342d86c7052dca383a") }
```

Each Key Has a Data Type

- MongoDB specific data types:

4-byte signed integer:

```
{ "int4" : NumberInt("2147483648") }
```

8-byte signed integer:

```
{ "int8" : NumberLong("9223372036854775808") }
```

Using Query Operators

- Retrieving the documents of an author list:

```
> db.authors.find( { "name":
                { $in : ["gribeiro", "eduardo"] } } );
```

- Retrieving all authors born between 1978 and 1998:

```
> db.authors.find(
  { "birthday": { $gte : new Date(1978, 0, 1),
                  $lte : new Date(1998, 0, 1)} } );
```

Or

```
> db.authors.find( { $and: [
     { "birthday": { $gte : new Date(1978, 0, 1) } },
     { "birthday": { $lte : new Date(1998, 0, 1) } }
                                  ] } );
```

Map/Reduce?

GO TO: http://webmapreduce.sourceforge.net/education.php

Defining a Map Function

```
> var map = function() {
                emit(this.key, this.value);
            };
```

Let's create a map function that signals the author of a post and the tags used in the post

```
> var map = function() {
                author = this["author"];

                tags = this["tags"];

                emit( author, { "tags": tags } );
            };
```

Defining a Reduce Function

```
> var reduce = function(key, emits) {
                 return same-type-as-emits;
               };
```

Let's create a reduce function that merges the tags used by a given author for all its posts

```
> var reduce = function(key, emits) {
               var alltags = [];

               for(var i in emits) {
                   var tags = emits[i]["tags"];

                   for(var j in tags) {
                        tag = tags[j];

                       if(alltags.indexOf(tag) == -1) {
                           alltags.push(tag);
                       }
                   }
               }

               return {"tags" : alltags};
           };
```

Running Map-Reduce

```
> db.posts.mapReduce(map, reduce, {"out": "authors_tags" } );

{
  "result" : "authors_tags",
  "timeMillis" : 13,
  "counts" : {
              "input" : 4,
              "emit" : 4,
              "reduce" : 1,
              "output" : 3
  },
  "ok" : 1,
}

> db.authors_tags.find();
{ "_id": "cassia", "value": { "tags": [ "meals", "meat",
"chicken" ] } }
{ "_id": "gribeiro", "value": { "tags": [ "geo", "database",
"nosql", "scidb", "mongodb", "couchdb" ] } }
...
```

The name of the temporary collection

Notes on Map-Reduce

- We can:
  - add a query clause before passing documents to the map function;
  - limit the number of documents to sent to the map function;
  - sort documents before sending to map function;
  - use a finalize function called after the last reduce output;
  - use a scoped value to pass to map, reduce and finalize functions.

- Store the result in a temporary collection:
  - By default these collections are dropped after the connection being closed;
  - We can control this behaviour → "keeptemp ": true

- Can be performed in multi-thread.

- mongos dispatch the jobs across all shards.

Replication

Replica Set

Replication

- Database replication is a well known technique for keeping copies of our data on multiple servers:
  - we need to be prepared for situations where one instance crashes or it becomes unavailable.

- Replication ensures:
  - Data redundancy and backup;
  - Automatic failover.

Replication

- In MongoDB, replication occurs through groups of servers known as *replica set*:
  - One is designated as the primary and the others as secondaries;
  - All writes from the clients are directed to the primary;
  - While the secondaries replicate asynchronously from the primary;
  - The automatic failover will elect a new primary instance from the secondaries.

- The algorithm used in MongoDB for replication is based on majorities:
  - you need a majority of members to elect a primary;
  - a primary can only stay primary so long as it can reach a majority;
  - and a write is safe when it's been replicated to a majority;
  - You can have an arbiter.

Setting up a replica set with three instances in our single machine

Replica Set and Clients

Working with Sharded Environments

Sharding

What is Shard?

- Sometimes we need to split a large database across several instances:
  - resources can be to big to fit in a single machine;
  - each individual partition is referred to as a shard or database shard.

- Do not get confused between sharding and replication, as they are very distinct.

“Novas” tecnologias de bancos de dados

Graph Databases

“Links and Properties”

Graph Databases

- Instead of tables, rows and columns we have:
  - Nodes, Edges and Properties
- Nodes and Edges are first class citizens
- This model fits well for
  - Web 2.0 Social Networking Applications
  - Semantic Web

Neo4J

- Site: http://www.neo4j.org

API Java

```
firstNode = graphDb.createNode();
firstNode.setProperty( "message", "Hello, " );
secondNode = graphDb.createNode();
secondNode.setProperty( "message", "World!" );

relationship = firstNode.createRelationshipTo( secondNode, RelTypes.KNOWS );
relationship.setProperty( "message", "brave Neo4j " );
```

Source: neo4j manual

Neo4J Spatial

- It is a library that adds spatial functionalities to Neo4j:
  - Support for geometry types
  - Topology operations
  - R-tree
  - Layer: shapefile importer, Open Street Map file importer

- Work with:
  - GeoServer, Geotools and uDig

- Site: https://github.com/neo4j/spatial

Graph Databases

- Other Implementations:
  - OrientDB:
    - License: Apache 2.0
    - Language: Java
    - Site: http://www.orientechnologies.com/orient-db.htm

- Gremlin:
  - a graph traversal language
  - Site: http://wiki.github.com/tinkerpop/gremlin

“Novas” tecnologias de bancos de dados

Array Databases

“Arrays as first class citizens”

Referências

- ELMASRI, R.; NAVATHE, S. B. ***Fundamentals of database systems***. Addison Wesley, 2006. 1139p.

- GRAY, J. ***Evolution of Data Management***. IEEE Computer 29(10): 38-46, 1996.

- NAUGHTON, J. F. ***DBMS Research: First 50 Years, Next 50 Years***. Kynote speaker' slides at ICDE 2010. Available at: http://pages.cs.wisc.edu/~naughton/naughtonicde.pptx. Access: April, 2013.

MINISTÉRIO DA CIÊNCIA, TECNOLOGIA E INOVAÇÃO
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

Mineração de Dados

CAP 349: Bancos de Dados Geográficos 2014

Adeline Marinho
Doutorado – CAP – INPE
adelsud6@gmail.com

8 setembro 2014

Introdução

- Crescimento explosivo das bases de dados
  - Governo, corporações, institutos científicos
  - *Internet*
- Demanda por informações estratégicas
  - Meios convencionais para examinar dados
  - Análise automática e inteligente de grandes BDs
- Disponibilidade de dados
  - Armazenamento eletrônico

Cenário

Hoje, a maioria das organizações produz mais informações em uma semana do que muitas pessoas poderiam ler em toda vida.

“MAIS DADOS PODE SIGNIFICAR MENOS INFORMAÇÕES”

Cenário

- Dificuldade para analisar dados
  - métodos manuais tradicionais
- A informação não está explicitada
- Necessidade
  - técnicas que facilitem a extração da informação
  - dado operacional não oferece grande valor quando estudado isoladamente.

O que fazer?

Dados ricos, porém pobres em informação

Fonte: (HAN e KAMBER, 2006)

Informação e Conhecimento

- Segundo Han e Kamber (2006)
  - A abundância de dados
  - A necessidade de ferramentas de análise de dados

- É descrita como uma situação de dados ricos, mas pobres em informação.
  - Os dados tornam-se "dados túmulos" arquivos de dados que raramente são visitados.

Processo de KDD

- Descoberta de Conhecimento em BDs
  - *Knowledge Discovery in Databases* (KDD) (FAYYAD et al., 1996)
  - Ferramentas e técnicas empregadas para análise automática e inteligente de imensos repositórios
- Processo não trivial de identificar em dados padrões que sejam
  - Válidos
  - Novos (previamente desconhecidos)
  - Potencialmente úteis
  - Compreensíveis
- Visando melhorar o entendimento de um problema ou um procedimento de tomada de decisão.

Etapas do KDD

Etapas do processo KDD

Etapas do KDD

- Fases/Etapas do KDD
  - Seleção.
  - Pré-processamento.
  - Transformação.
  - Mineração de dados
  - Interpretação e Avaliação.

Etapas do KDD

- **Seleção**
  - Selecionar ou segmentar dados de acordo com critérios definidos:
    - Ex.: Todas as pessoas que são proprietárias de carros é um subconjunto de dados determinado.

Etapas do KDD

- **Pré-processamento**
  - Estágio de limpeza dos dados, onde informações julgadas desnecessárias são removidas.
    - Ex. : O sexo de um paciente, gestante
  - Reconfiguração dos dados para assegurar formatos consistentes (identificação)
    - Ex. : sexo = “F” ou “M”

      sexo = “M” ou “H”
  - Agregar dados externos

Etapas do KDD

- **Transformação**
  - Transforma-se os dados em formatos utilizáveis.
  - Depende da técnica *data mining* usada.
    - converter valor literal em valor numérico
  - Normalização de valores
  - Agregar semântica ao dado
  - Disponibilizar os dados de maneira usável e navegável.

Etapas do KDD

- **Mineração de Dados**
  - É a verdadeira extração dos padrões de comportamento dos dados
  - Utilizando a definição de fatos, medidas de padrões, estados e o relacionamento entre eles.

Etapas do KDD

- **Interpretação e Avaliação**
  - Identificado os padrões pelo sistema, estes são interpretados em conhecimentos, os quais darão suporte a tomada de decisões humanas
  - Ex.: Tarefas de previsões e classificações

Processo de KDD

- Alguns possíveis resultados são:
  - Confirmação do óbvio
  - Conhecimento novo
  - Nenhum relacionamento encontrado (dados aleatórios)
- Problemas:
  - Identificação dos dados relevantes
  - Representação dos dados
  - Busca por modelos ou padrões válidos

Mineração de Dados

- É uma das etapas no KDD
- A mineração de dados preocupa-se em buscar conhecimento compreensível em grandes conjuntos de dados.

Mineração de Dados

- Utiliza técnicas e algoritmos de diferentes áreas do conhecimento
  - inteligência artificial
    - aprendizagem de máquina
  - banco de dados
    - recursos para manipular grandes bases de dados
  - Estatística
    - avaliação e validação de resultados

Técnicas de Mineração de Dados

Aprendizagem Supervisionada

- Aprendizagem do modelo é supervisionada
  - É fornecida uma classe à qual cada amostra no treinamento pertence

- Algoritmos preditivos
  - suas tarefas de mineração desempenham inferências nos dados;
  - fornecem previsões ou tendências, obtendo informações não disponíveis a partir dos dados disponíveis.

Aprendizagem Supervisionada

- **Classificação**
  - determina o valor de um atributo *(classe)* através dos valores de um subconjunto dos demais atributos.

- Ex. Qual o perfil dos clientes que consomem cosméticos importados?
  - Inferir (Prever) - "clientes do sexo feminino, com renda superior a R$ 1.500,00 e com idade acima de 30 anos compram cosméticos importados "
  - Atributo compra cosmético importado = classe
    - Atributo alvo (Sim ou Não)

Aprendizagem Supervisionada

- **Classificação**
  - Formas mais comuns de representação: regras e árvores.

Árvore de decisão

Aprendizagem Supervisionada

- **Seleção de atributos**
  - Alguns atributos têm um peso maior ou até determinante nas tarefas de mineração de dados
    - ex.: o atributo renda é determinante nos hábitos de consumo do cliente
  - Com os algoritmos é possível determinar os atributos relevantes para a mineração separando-os dos atributos irrelevantes
    - ex.: nome do cliente caso não influenciem nos hábitos de consumo

Aprendizagem Não-Supervisionada

- O rótulo da classe de cada amostra do treinamento não é conhecida
  - Número ou conjunto de classes a ser treinado pode não ser conhecido a priori

- Descritivos
  - Descreve de forma concisa os dados disponíveis
  - Fornece características das propriedades gerais dos dados minerados.

Aprendizagem Não-Supervisionada

- **Associação**
  - A classe da tarefa de mineração não é determinada
  - Gera regras do tipo:
    - clientes do sexo masculino, casados, com renda superior a R$ 1.800,00 têm o seguinte hábito de consumo: *roupas de grife, perfumes nacionais, relógios importados*
  - Revela associações entre valores dos atributos

Aprendizagem Não-Supervisionada

- Exemplo de Regras de associação

```
 1. classe=tripulação 885 ==> idade=adulto 885    conf:(1)
 2. classe=tripulação sexo=masculino 862 ==> idade=adulto 862    conf:(1)
 3. sexo=masculino sobreviveram=não 1364 ==> idade=adulto 1329    conf:(0.97)
 4. classe=tripulação 885 ==> sexo=masculino 862    conf:(0.97)
 5. classe=tripulação idade=adulto 885 ==> sexo=masculino 862    conf:(0.97)
 6. classe=tripulação 885 ==> idade=adulto sexo=masculino 862    conf:(0.97)
 7. sobreviveram=não 1490 ==> idade=adulto 1438    conf:(0.97)
 8. sexo=masculino 1731 ==> idade=adulto 1667    conf:(0.96)
 9. idade=adulto sobreviveram=não 1438 ==> sexo=masculino 1329    conf:(0.92)
10. sobreviveram=não 1490 ==> sexo=masculino 1364    conf:(0.92)
```

Aprendizagem Não-Supervisionada

- *Clustering*
  - Verifica como as instâncias de uma determinada base de dados se agrupam
    - considera características intrínsecas de seus atributos, sem que seja definida uma classe para a tarefa
  - A partir de uma métrica de similaridade
    - objetos são agrupados com base no princípio da maximização da similaridade intraclasse e da minimização da similaridade interclasse
  - ex.: identificar subgrupos homogêneos de clientes

Clustering

Clustering

Validação de Resultados

- É importante que os resultados e modelos possam ser avaliados e comparados
- Teste e validação
  - fornecem parâmetros de validade e confiabilidade nos modelos gerados
    - *cross validation*, *suplied test set*, *training set*, *percentage split*
  - indicadores estatísticos para auxiliar a análise dos resultados
    - matriz de confusão, estatística kappa, erro médio absoluto, precisão...

Medidas de interesse

Artigo (item)	número que o representa
Pão	1
Leite	2
Açúcar	3
Papel Higiênico	4
Manteiga	5
Fralda	6
Cerveja	7
Refrigerante	8
Iogurte	9
Suco	10

Representação numérica de cada artigo do supermercado

Um banco de dados de transações de clientes

TID	Itens comprados
101	{1,3,5}
102	{2,1,3,7,5}
103	{4,9,2,1}
104	{5,2,1,3,9}
105	{1,8,6,4,3,5}
106	{9,2,8}

Medidas de Interesse - Confidência

- Grau de confiança
  - conf(A=>B)
  - A porcentagem das transações que suportam B dentre todas as transações que suportam A

$$conf(A \rightarrow B) = \frac{\text{número de transações que suportam } (A \cup B)}{\text{número de transações que suportam } A}$$

  - Ex. O grau de confiança da regra
  - {cerveja}=>{manteiga}, isto é {7}=>{5}, com relação ao banco de dados é 1{100%}

Confidência

Artigo (item)	número que o representa
Pão	1
Leite	2
Açúcar	3
Papel Higiênico	4
Manteiga	5
Fralda	6
Cerveja	7
Refrigerante	8
Iogurte	9
Suco	10

Representação numérica de cada artigo do supermercado

Um banco de dados de transações de clientes

TID	Itens comprados
101	{1,3,5}
102	{2,1,3,7,5}
103	{4,9,2,1}
104	{5,2,1,3,9}
105	{1,8,6,4,3,5}
106	{9,2,8}

Conf {7}->{5} = 1/1 = 1

Confidência

- Seja uma regra "A => B".
- A confidência da regra é dada por:

$$\text{Confidência } (A \Rightarrow B) = \frac{\#_\text{Tuplas_Contendo_Tanto_A_Como_B}}{\#_\text{Tuplas_Contendo_A}}$$

- Exemplo:
  - Uma confidência de 85% (0,85) da regra:

    compra(mulher, computadores) =>compra(mulher, software)
  - *S*ignifica que 85% das mulheres que compram computadores também compram software.

Medidas de Interesse - Suporte

- Função do suporte
  - determinar a frequência que ocorre um *itemset* dentre todas as transações da base de dados
    - é a porcentagem de transações onde este *itemset* aparece
  - Um *itemset* será frequente se seu suporte for maior ou igual a um suporte mínimo estabelecido previamente.
- Forma:
  - X =>Y (lê-se X implica em Y)
    - onde X é o antecessor e Y o consequente
    - X e Y são dois *itemsets* distintos na Base de Dados

Suporte

- O suporte da regra "A => B" é dado por:

$$\text{Suporte (A => B)} = \frac{\text{\#_Tuplas_Contendo_Tanto_A_Como_B}}{\text{\#_Total_De_Tuplas}}$$

$$\text{Sup}\ (X\ U\ Y) = \frac{\text{Nº de registros com } (X\ U\ Y)}{\text{Nº total de transações do BD}}$$

  - Um suporte de 5% significa que de todas as transações comerciais realizadas, 5% são efetuadas por *mulheres que comprando computador também compram softwares.*

Suporte

Um banco de dados de transações de clientes

TID	Itens comprados
101	{1,3,5}
102	{2,1,3,7,5}
103	{4,9,2,1}
104	{5,2,1,3,9}
105	{1,8,6,4,3,5}
106	{9,2,8}

Artigo (item)	número que o representa
Pão	1
Leite	2
Açúcar	3
Papel Higiênico	4
Manteiga	5
Fralda	6
Cerveja	7
Refrigerante	8
Iogurte	9
Suco	10

Representação numérica de cada artigo do supermercado

Itemset	Suporte
{1,3}	0,6666
{2,3}	0,3333
{1,2,7}	0,16666
{2,9}	0,5

Suporte de alguns *itemsets*

Sup **{7}->{5} = 1/6 = 0.16...**

Critérios de Comparação

- Critérios para comparar métodos e resultados de mineração de dados permitem avaliar e optar pelo melhor custo/benefício
  - Precisão avaliativa ou preditiva
    - habilidade do modelo para avaliar ou prever corretamente classes, agrupamentos, regras.
  - Velocidade
    - custo computacional da geração e utilização do modelo

Critérios de Comparação

- Robustez
  - habilidade do modelo para avaliar ou prever corretamente utilizando dados ruidosos ou com valores ausentes
- Escalabilidade
  - capacidade de construir modelos eficientemente a partir de grandes volumes de dados
- Interpretabilidade
  - nível de compreensão fornecido pelo modelo

Softwares

KDnuggets Polls

May 20 - Jun 2, 2014

What Analytics, Big Data, Data mining, Data Science software you used in the past 12 months for a real project? [3285 voters]	
Legend: Red: Free/Open Source tools Green: Commercial tools	% users in 2014 % users in 2013
RapidMiner (1453), 35.1% alone	44.2% 39.2%
R (1264), 2.1% alone	38.5% 37.4%
Excel (847), 0.1% alone	25.8% 28.0%
SQL (832), 0.1% alone	25.3% na
Python (639), 0.9% alone	19.5% 13.3%
Weka (558), 0.4% alone	17.0% 14.3%
KNIME (492), 10.6% alone	15.0% 5.9%
Hadoop (416), 0% alone	12.7% 9.3%
SAS base (357), 0% alone	10.9% 10.7%
Microsoft SQL Server (344), 0% alone	10.5% 7.0%
Revolution Analytics R (300), 13.3% alone	9.1% 4.5%
Tableau (298), 1.3% alone	9.1% 6.3%
MATLAB (277), 0% alone	8.4% 9.9%

Fonte: http://www.kdnuggets.com/polls/2014/analytics-data-mining-data-science-software-used.html

MINISTÉRIO DA CIÊNCIA, TECNOLOGIA E INOVAÇÃO
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

Mineração de Dados Espaciais

Exemplo Histórico

- Cólera Asiática em Londres (1855): um poço identificado como a fonte do problema

Exemplos Modernos

- *Clusters* de pessoas com cancer para investigar a influência do ambiente
- Locais de concentração de crimes para planejar as rotas de patrulha da polícia
- Identificação de características onde certos tipos de águias fazem os ninhos

Exemplos Modernos

- Aquecimento anormal de região do oceano Pacífico (El Niño) afeta o clima

Average Monthly Temperature
(Courtsey: NASA, Prof. V. Kumar)

Global Influence of El Nino during the Northern Hemisphere Winter
(D: Dry, W: Warm, R: Rainfall)

O que é um padrão espacial?

- O que não é um padrão?
  - Aleatório, ao acaso, acidental
  - Casual

- O que é um padrão (*pattern*)?
  - Um arranjo frequente, configuração, regularidade
  - Uma regra, lei, método
  - Uma direção ou tendência importante
  - Uma irregularidade espacial importante

O que é descoberta de conhecimento em dados espaciais?

- Definindo *Data Mining* Espacial
  - Procura por padrões espaciais
  - Procura não-trivial tão automática quanto possível
    - reduzindo o esforço humano
  - Padrões espaciais interessantes, úteis e inesperados
    - Desconhecidos
  - Auxílio especialista do domínio

O que é *data mining* espacial?

- Busca não trivial por padrões espaciais **interessantes** e **desconhecidos**
- Busca não trivial
  - Grande (ex. exponencial) espaço de busca de hipóteses plausíveis
  - Ex. Cólera Asiática - causas plausíveis: água, alimento, ar, insetos, …;
- Interessante
  - Útil em algum domínio de aplicação
  - Ex. Desativando o poço identificado => salvar vidas humanas
- Inesperado
  - O padrão não é conhecimento comum
  - Pode levar a um novo entendimento do mundo
  - Ex. A conexão Poço - Colera levou a teoria do "germe"

O que NÃO é *data mining* espacial?

- Consultas simples a dados espaciais
  - Encontre os vizinhos de Porto Alegre dados os nomes e limites de todas as cidades
  - Encontre o menor caminho do RS a SP na malha de rodovias
  - O espaço de busca não é grande (não é exponencial)
- Testar uma hipótese através de uma análise simples de dados
  - Ex. O território das chimpanzes femeas é menor do que o dos machos
  - O espaço de busca não é grande!

O que NÃO é *data mining* espacial?

- Padrões espaciais não interessantes ou óbvios
  - Muita chuva em Porto Alegre está correlacionada com muita chuva em Canoas, dado que as duas cidades são vizinhas.
  - Conhecimento comum: lugares próximos tem precipitações similares
- Mineração de dados não espaciais
  - As vendas de fraldas e cervejas são correlacionadas nas sextas-feiras

Porque estudar mineração de dados espaciais?

- Novo conhecimento dos processos geográficos para questões críticas
  - Ex. Como está a saúde do planeta Terra?
  - Ex. Caracterizar os efeitos da atividade humana para o ambiente e a ecologia
  - Ex. Predizer o efeito do El Niño no clima e na economia

Porque estudar mineração de dados espaciais?

- Abordagem tradicional: gerar e testar hipóteses manualmente
  - Mas os dados espaciais estão crescendo rápido demais para uma análise manual
    - Imagens de satélite, trajetórias geradas por GPS, sensores em rodovias, …
  - Número de hipóteses geográficas possíveis é grande demais para uma análise manual
    - Grande número de objetos geográficos
    - O número de relacionamentos entre os objetos cresce exponencialmente
    - Ex. Encontre correlação entre eventos climáticos oceânicos e em terra firme
- *Data Mining* Espacial pode reduzir o conjunto de hipóteses plausíveis

Escolha dos métodos

- Duas abordagens:
  - Uso de técnicas específicas para mineração de dados espaciais
  - Obtenção dos dados ou relacionamentos espaciais de interesse para uso com métodos de DM clássicos

- Abordagem possível:
  - Defina o problema: obtenha as necessidades particulares
  - Analise os dados usando mapas e outras técnicas de visualização
  - Tentar usar métodos clássicos de *data mining*
    - Se não obtiver resultados satisfatórios, tente novos métodos
  - Avalie os métodos escolhidos rigorosamente

Famílias de padrões espaciais

- Famílias comuns de padrões espaciais
  - Classificação
  - *Clustering*
  - Detecção de *Outliers*
  - Co-location
  - Trajetórias
  - ….
- Outras famílias podem ser definidas

Classificação

- Dado um conjunto de instâncias, a função da classificação é descobrir as classes das instâncias

- Objetos podem ser caracterizados, classificados, por diferentes tipos de informações
  - Atributos não espaciais (população)
  - Atributos espacialmente relacionados com valores não-espaciais (população total que vive a menos de 100 metros das antenas celulares);

Remote Sensing Data Mining (Silva *et al.* 2008)

Figure 2. Examples of patterns of tropical deforestation proposed by Mertens and Lambin (1997) in the Brazilian Amazonia: corridor, diffuse, fishbone, and geometric.

Método

Figure 1. Proposed method for remote sensing image mining.

Métricas de Ecologia da Paisagem

- Perimeter (m):

$$\mathrm{PERIM} = p_{ij}. \tag{1}$$

- Area (ha):

$$\mathrm{AREA} = \left(a_{ij}/10\,000\right). \tag{2}$$

- PARA, perimeter–area ratio, a measure of shape complexity:

$$\mathrm{PARA} = \frac{p_{ij}}{a_{ij}}. \tag{3}$$

- Shape, shape compactness index, calculated by the patch perimeter p_{ij} divided by $p_{ij\ \min}$, which is the minimum perimeter possible for a maximally compact patch of the matching patch area. It is equal to 1 when the region is a square and grows according to the region's irregularity.

$$\mathrm{SHAPE} = \frac{p_{ij}}{p_{ij\ \min}}. \tag{4}$$

Árvore de Decisão

Figure 11. Decision tree for Vale do Anari spatial patterns. The metrics are: perimeter/area ratio (PARA) and circumscribing circle (CIRCLE).

Resultados

Geométrico
Irregular
Linear

Área desflorestada - Vale do Anari
ÁREA (ha)
10000
8000
6000
4000
2000
0
85
85_88
88_91
91_94
94_97
97_00
ANO
IRR
LIN
GEO

Clustering (cluster analysis)

- *Clustering* é um processo de particionamento de um conjunto de dados em um conjunto de grupos chamados *clusters*

- Um cluster é um conjunto de dados (objetos) com características similares
  - que podem ser tratadas coletivamente como um grupo

- *Clustering* é um método não supervisionado
  - não há classes pré-definidas

Clustering Analysis (Kumar 2005)

- Diferentes formas de agrupamento para o mesmo conjunto de pontos

Principais categorias de *clustering*

- Métodos de Particionamento
  - A divisão de dados em subconjuntos (*clusters*)
  - Dados n objetos, o método constrói k partições, onde cada partição representa um grupo, e $k \leq n$.
- Métodos Hierárquicos
  - Um conjunto de clusters aninhados organizado como uma árvore hierárquica
- Métodos Baseados em Densidade
  - Encontra grupos com base na densidade das regiões
- Baseado em Grades
  - Encontrar grupos com base no número de pontos em cada célula

K-Means

- Métodos de Particionamento
- Cada conjunto está associado com um centroide
- Cada ponto é atribuído ao grupo com o centroide mais próximo
- A desvantagem das k-means é que o número de *clusters* K é um parâmetro de entrada

K-Means

Hierarchical Clustering

- Existem dois tipos principais: Divisivo e Aglomerativo
- Aglomerativo
  - Começa com todos os objetos como aglomerados individuais
  - A cada passo, uni os dois *clusters* mais semelhantes
  - Até restar um *cluster* (ou k *clusters*)

DBSCAN (Ester 1996)

- DBSCAN é um algoritmo baseado em densidade
- Densidade = número de pontos dentro de um raio específico (EPS)
- Um ponto é um ponto central
  - se tem mais do que um determinado número de pontos (MinPts) dentro Eps
- Um ponto de fronteira
  - tem menos que MinPts dentro Eps, mas está na vizinhança de um ponto central
- Um ponto de ruído é qualquer ponto que não é um ponto central ou um ponto de fronteira.

DBScan: Core and border points

DBSCAN example

Identifying core, border and noise points

Computing distance

Final Clusters

Detecção de *Outliers*

- O que é um *outlier*?
  - Observações inconsistentes com o resto do conjunto de dados
- O que é um *outlier* espacial?
  - Observações inconsistentes com sua vizinhança
  - Uma instabilidade ou descontinuidade locais

Outliers – Examples *(Shekhar 2003)*

- Map Production
  - Error identification
  - E.g., spatial object violation

Ferramentas

- GeoMiner (Han 1997)
- INGENS (Malerba 2001)
- Weka-GDPM (Bogorny 2006)
  - http://www.inf.ufsc.br/~vania/software.html
- GeoDMA (Korting 2013)
  - http://www.dpi.inpe.br/menu/Projetos/geodma.php
- Spatial Data Mining and Visual Analytics Lab
  - http://www.spatialdatamining.org
- GeoImageRMP - Rapid Miner extension (Guyet 2013)
  - http://geoimagermp.gforge.inria.fr
- Lista com outros *softwares*
  - http://www.spatial.cs.umn.edu/sdm_software.htm
  - http://www.rdatamining.com/resources/tools

Geographic Data Mining Analyst (GeoDMA)

- Sistema que permite a aplicação de técnicas de mineração de dados espaciais sobre unidades de paisagem definidas por processos de segmentação

Fonte: Korting et al. (2013)

GeoDMA

Fonte: Korting et al. (2013)

VIS-STAMP: A Visualization System for Space-Time and Multivariate Patterns

Spatial Data Mining and Visual Analytics Lab

Fonte: Guo *et al*. (2006)

SOMVIS: Multivariate Mapping and Visualization

Spatial Data Mining and Visual Analytics Lab

Fonte: Guo *et al*. (2006)

Referências

- Adaptado de material elaborado pelo Prof. Shashi Shekhar, University of Minnesota www.cs.umn.edu/~shekhar
- SHEKHAR, S., CHAWLA, S. **Spatial databases: a tour**. Upper Saddle River, NJ: Prentice Hall, 2003.
- Bogorny, V. and Shekhar S. **Tutorial on Spatial and Spatio-Temporal Data Mining** In: SBBD 2008. (http://www.inf.ufsc.br/~vania/tutorial.zip).
- FAYYAD, U. M.; PIATETSKY-SHAPIRO, G.; SMYTH, P. From data mining to knowledge discovery in databases. **AI Magazine**, AAAI Press, v. 17, p. 37–54, 1996. Disponível em: <http://www.aaai.org/aitopics/assets/PDF/AIMag17-03-2-article.pdf>.
- HAN, J.; KAMBER, M.; PEI, J. **Data Mining: Concepts and Techniques**. 3rd ed.. ed. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2011.