



PÓS-GRADUAÇÃO

SENSORIAMENTO REMOTO

SER-300 – Introdução ao Geoprocessamento Arquitetura SIG & Bancos de Dados Geográficos

Vitor Conrado Faria Gomes
Gilberto Ribeiro de Queiroz

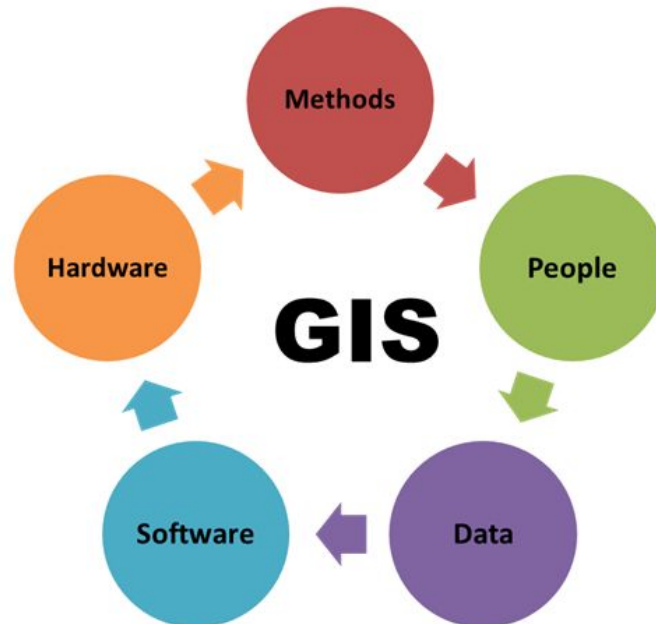
18.03.2022

“The best race car drivers understand how their cars work. The best architects know how carpenters, bricklayers, and electricians do their jobs. And the best programmers know how the hardware they are programming does computation”

Mark L. Chang “[Reconfigurable Computing](#)”, 2008

Sistema de Informação Geográfica (SIG)

- Geografic Information System (GIS)
- Sistema Computacional capaz de armazenar, processar e manipular dados geográficos (Worboys and Duckham, 2004)



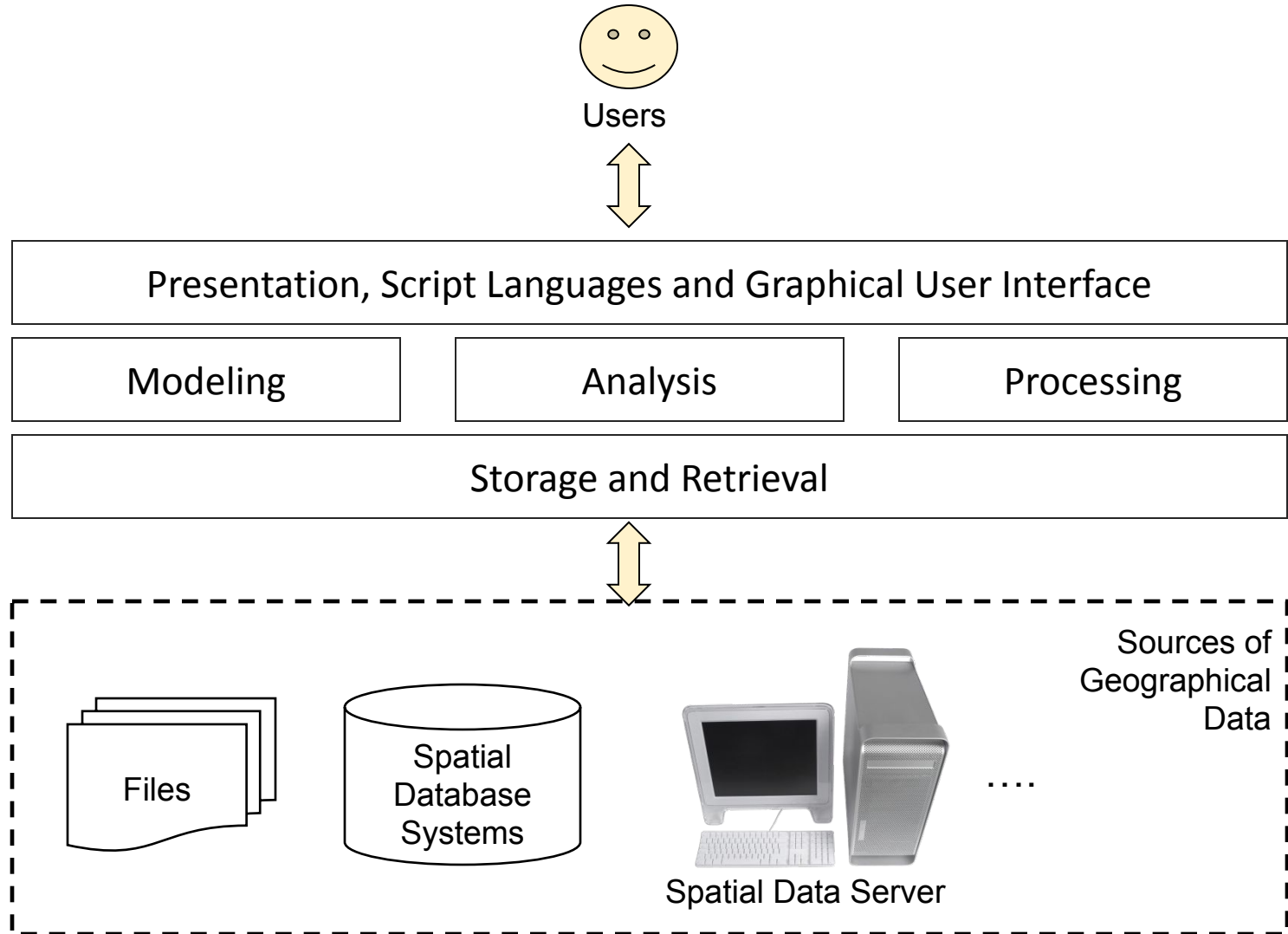
Sistema de Informação Geográfica (SIG)

Permite que os usuários visualizem, compreendam, questionem, interpretem e visualizem dados de várias maneiras que revelam relacionamentos, padrões e tendências na forma de mapas, globos, relatórios e gráficos [L. Kanickaraj \(2018\)](#).

Sistema de Informação Geográfica (SIG)

- 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 para criar novas representações do espaço geográfico; e
 - Ferramenta para análise espacial.

Sistema de Informação Geográfica (SIG)



Sistemas Desktop

QGIS

TerraView

gvSIG

Google Trends

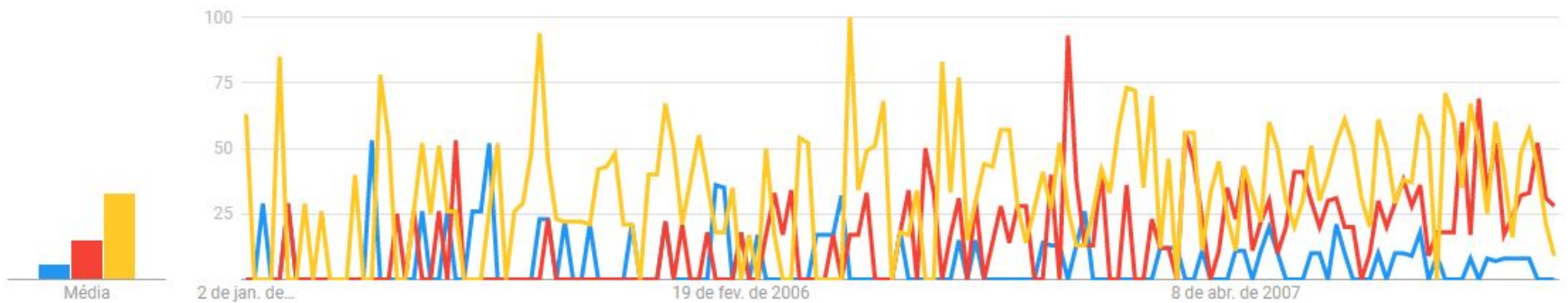
Todo o mundo ▾

01/01/2005 – 31/12/2007 ▾

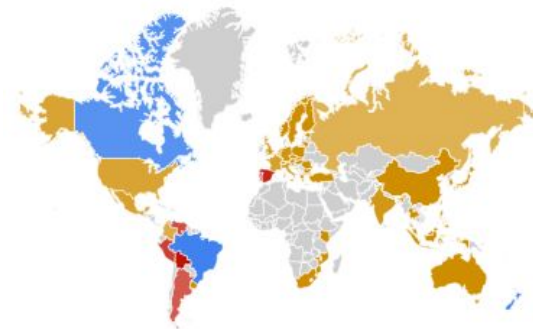
Todas as categorias ▾

Pesquisa na Web ▾

Interesse ao longo do tempo ⓘ



● terraview ● gvSIG ● qgis



Google Trends

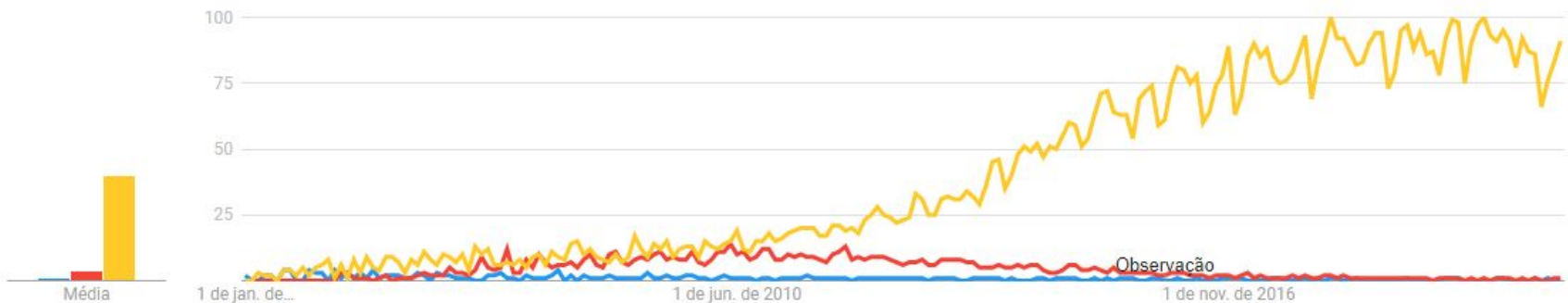
Todo o mundo ▾

2004 - presente ▾

Todas as categorias ▾

Pesquisa na Web ▾

Interesse ao longo do tempo ⓘ

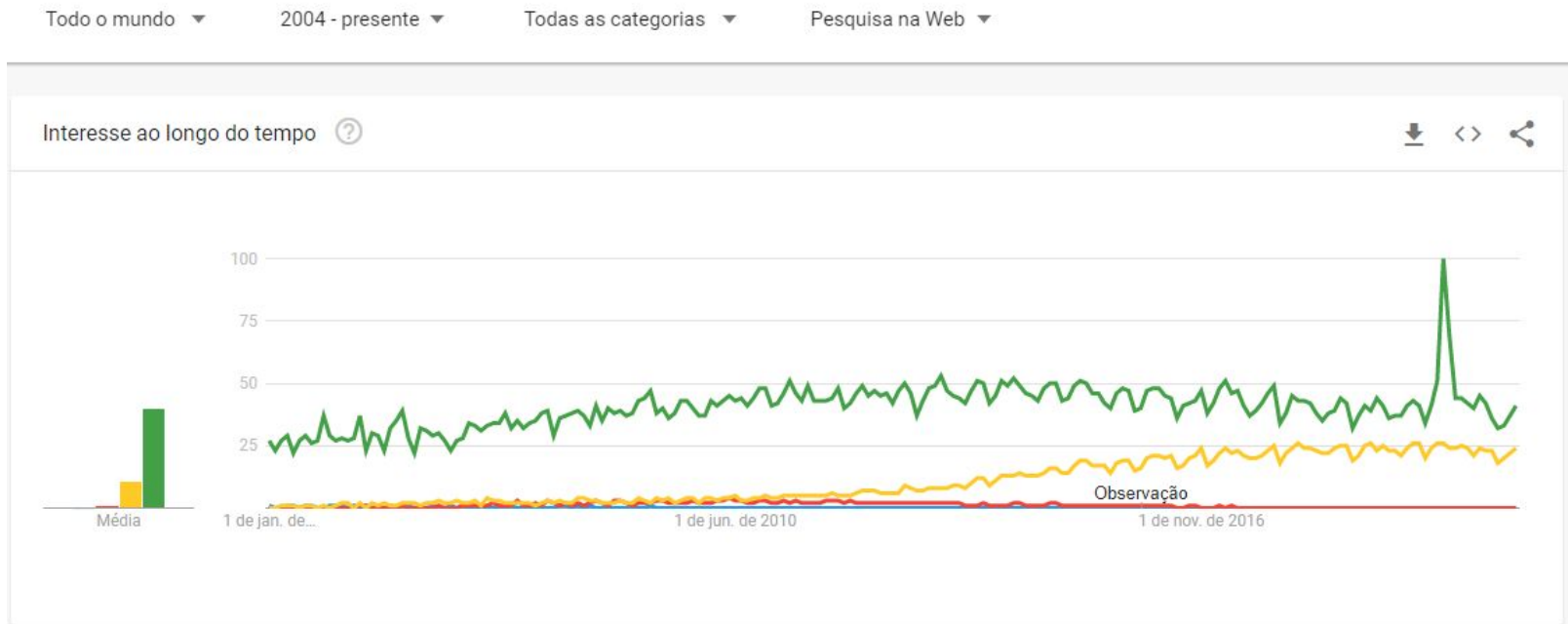


● terraview ● gvSIG ● qgis



Fonte: Google Trends: 18/03/2021

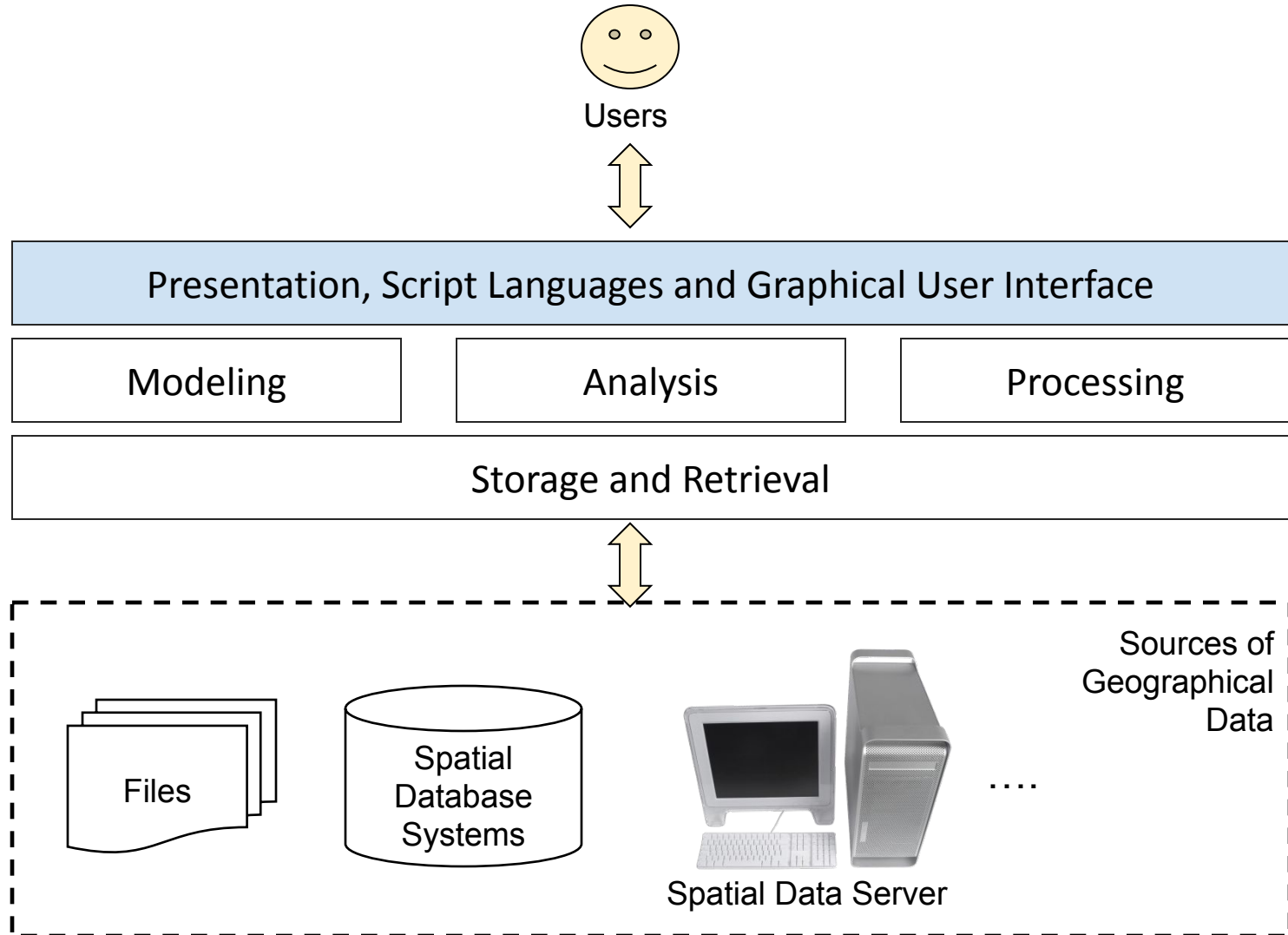
Google Trends



● terraview ● gvSIG ● qgis ● arcgis



Sistema de Informação Geográfica (SIG)



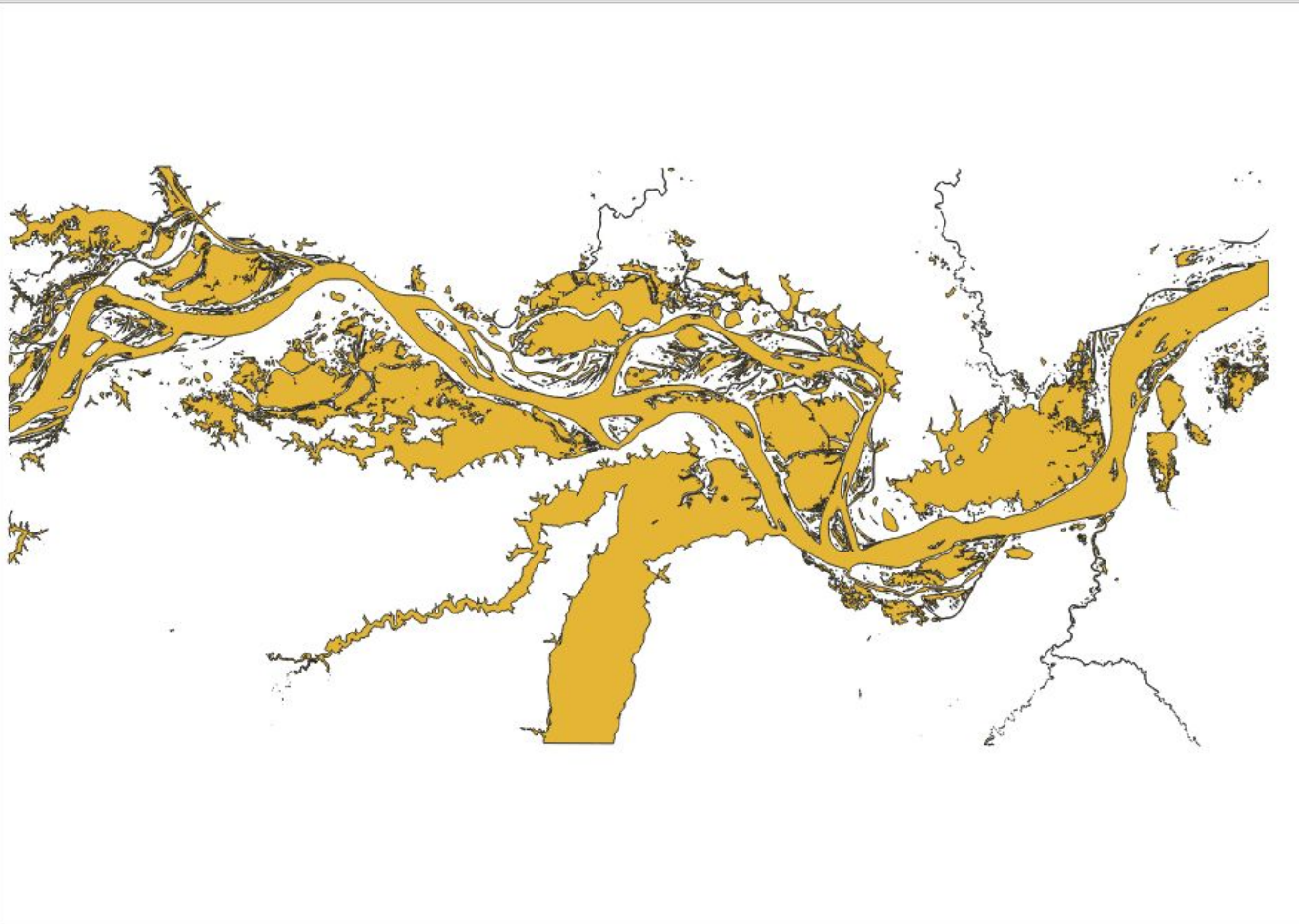


Browser

- Home
- /
- GeoPackage
- Spatialite
- PostGIS
- SAP HANA
- MSSQL
- WMS/WMTS
- Vector Tiles
- XYZ Tiles

Layers

- polygon

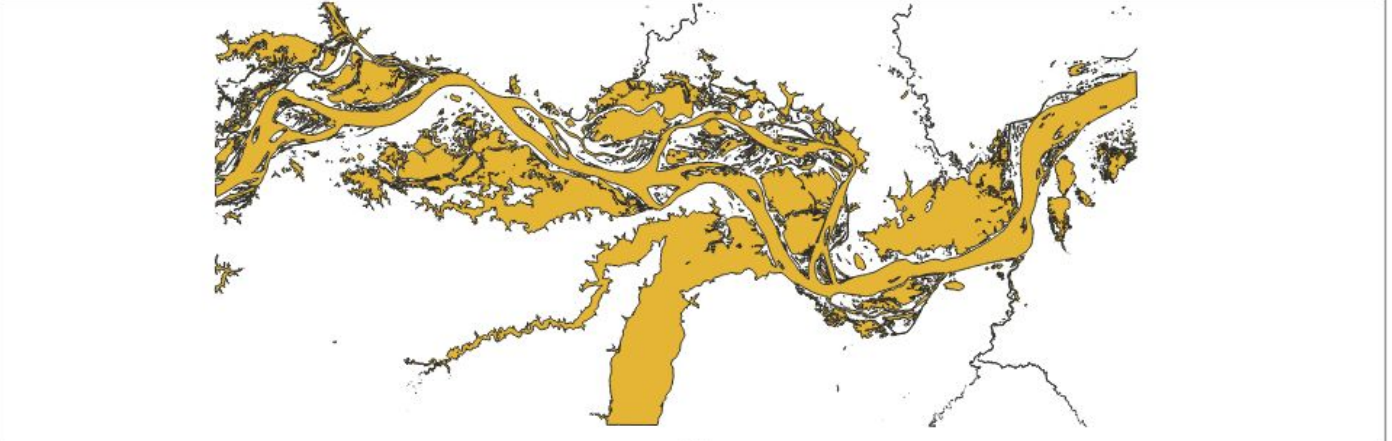


Type to locate (Ctrl+K)



Browser

- ⌵ Favorites
- ⌵ Spatial Bookmarks
- ⌵ Home
- /
- GeoPackage
- Spatialite
- PostGIS
- SAP HANA
- MSSQL
- ⌵ WMS/WMTS
- ⌵ Vector Tiles
- ⌵ XYZ Tiles



Layers

- ✓ polygon

Python Console

```
1 # Python Console
2 # Use iface to access QGIS API interface or type help(iface) for more info
3 # Security warning: typing commands from an untrusted source can harm your computer
4 >>> mc = iface.mapCanvas()
5 >>> layer = mc.currentLayer()
6 >>> layer.name()
7 'polygon'
8
```

```
>>>
```


TerraView - Project: New Project - Brasil_project.xml*

File View Project Layer Map Tools Plugins Vector Processing Slider Raster Processing Help

Layer Explorer Map Display Style Explorer

public.br_focos2003
public.br_munic_2001
public.br_uf_2010

Style Explorer

Style

- Rule
 - Polygon Symbol 0

Basic Symbology

Property	Value
Basic Fill	
Color	[0, 170, 0] (45)
Opacity	18
Basic Stroke	
Color	[85, 255, 0] (255)

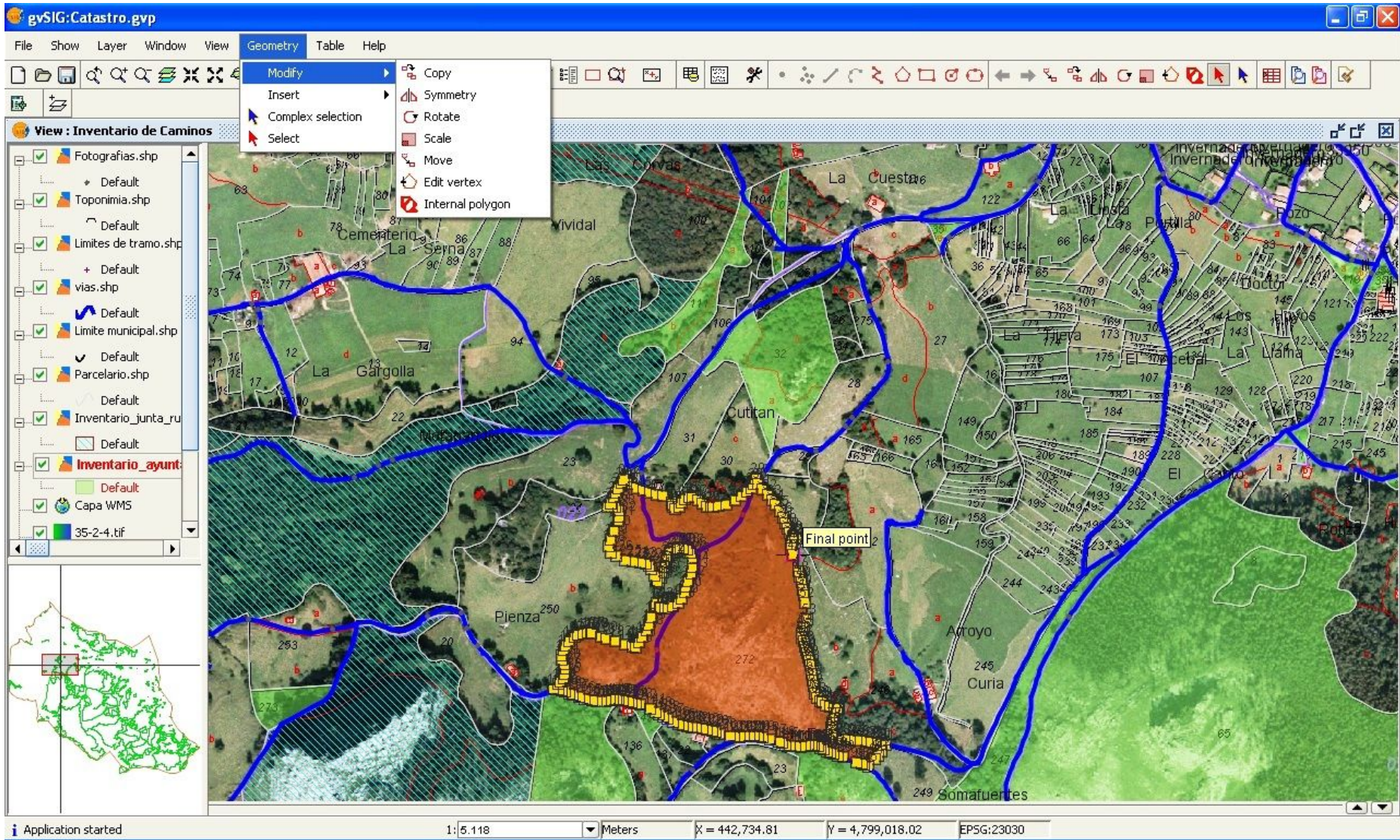
Graphic Symbology

Polygons Lines Points Raster

public.br_uf_2010

	gid	geocodigo	nome	area_tot_g	teste1
1	1	11	RONDÔNIA	237576.167	00:00:00
2	2	12	ACRE	152581.388	00:00:01
3	3	13	AMAZONAS	157745.68	00:00:02
4	4	14	RODRIGUES	231288.00	00:00:03

Selected rows: 0 EPSG:4291 (-72.50841, 18.72686)



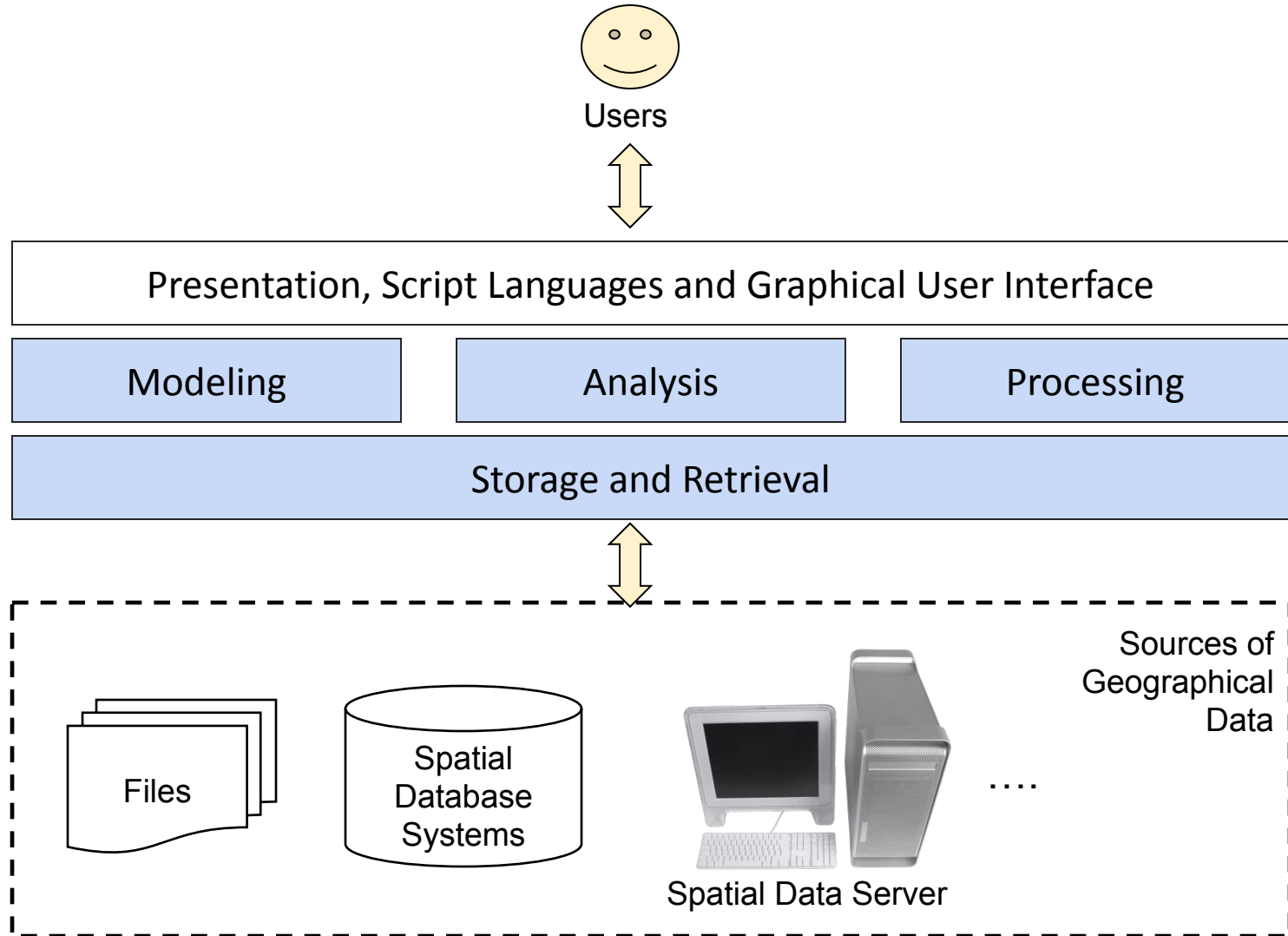
Fonte: wikipedia.de

WebGIS

<https://www.openstreetmap.org>

<http://www.inpe.br/queimadas/bdqueimadas/>

Sistema de Informação Geográfica (SIG)



Plataformas e Bibliotecas para Construção de GIS

<http://www.terralib.org>

<http://www.gdal.org>

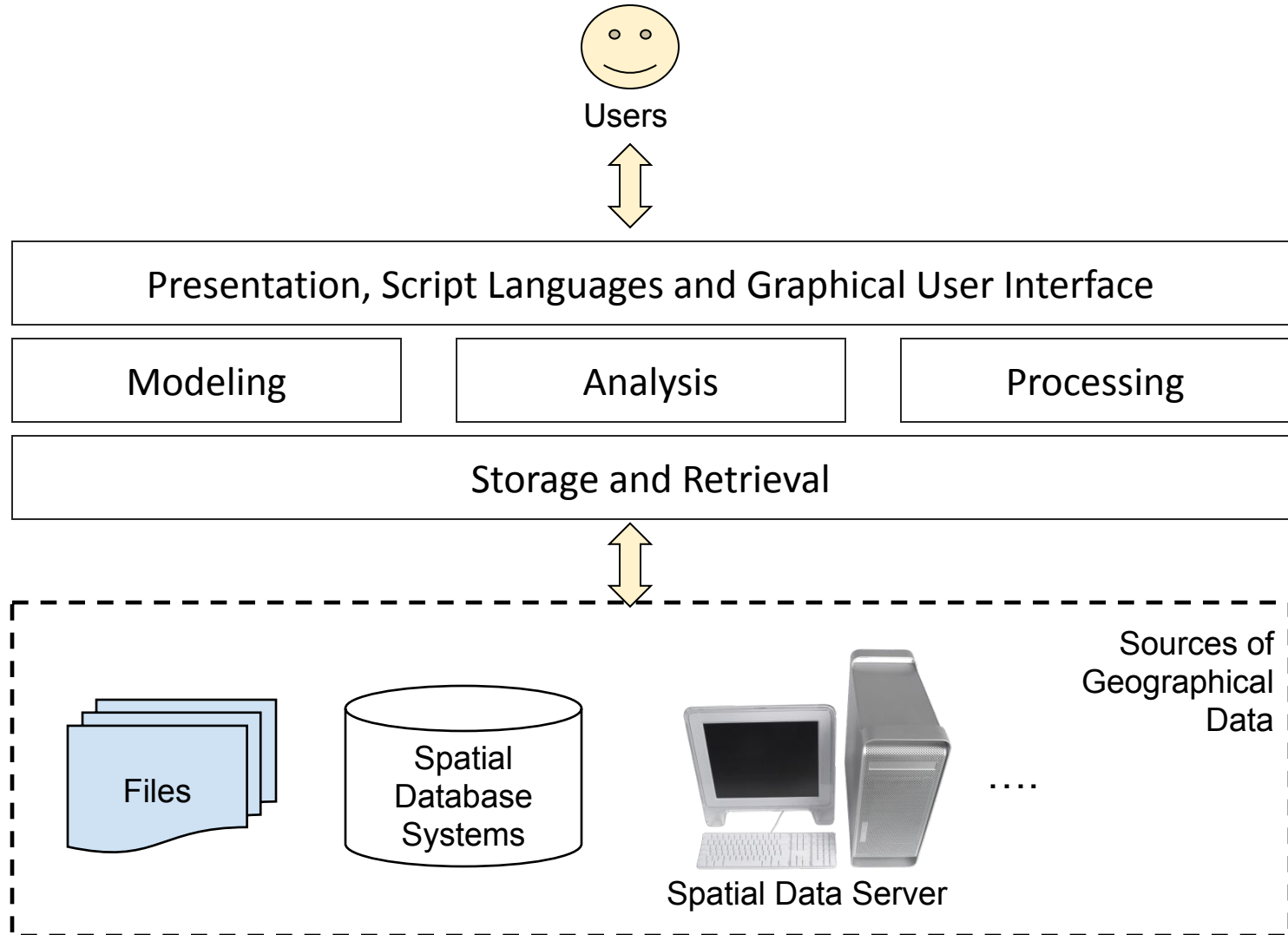
<http://www.geotools.org>

<https://geoext.org>

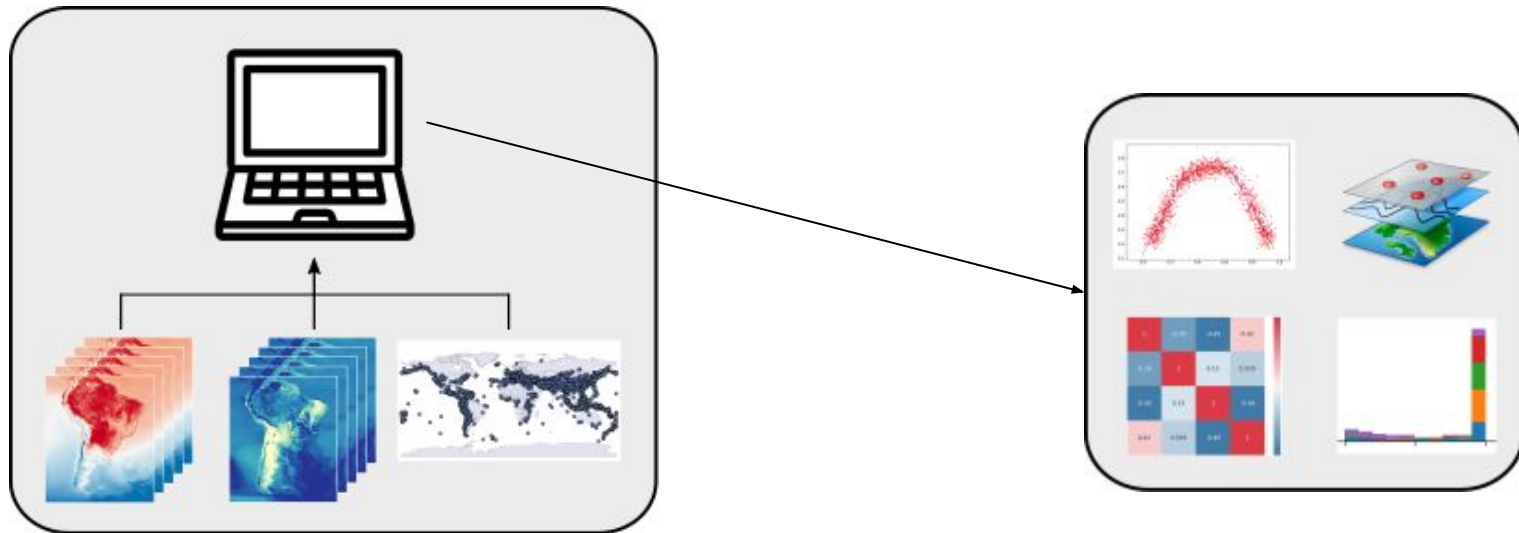
<https://leafletjs.com>

<https://openlayers.org>

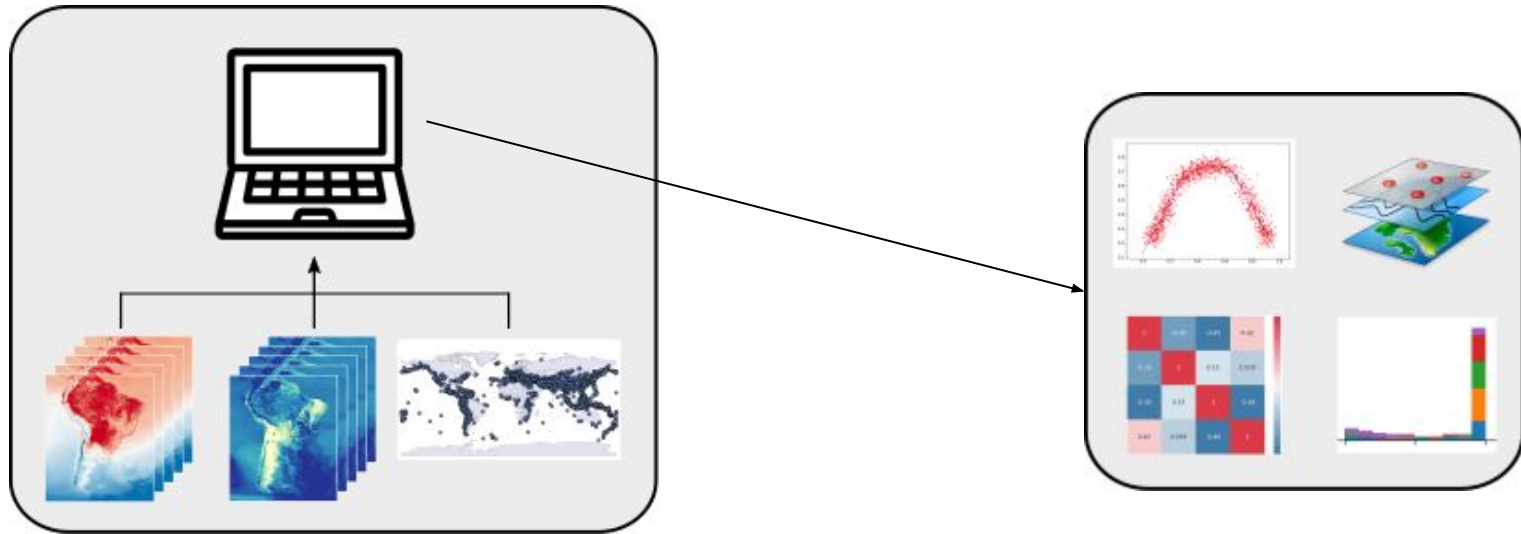
Sistema de Informação Geográfica (SIG)



Análise baseada em arquivos

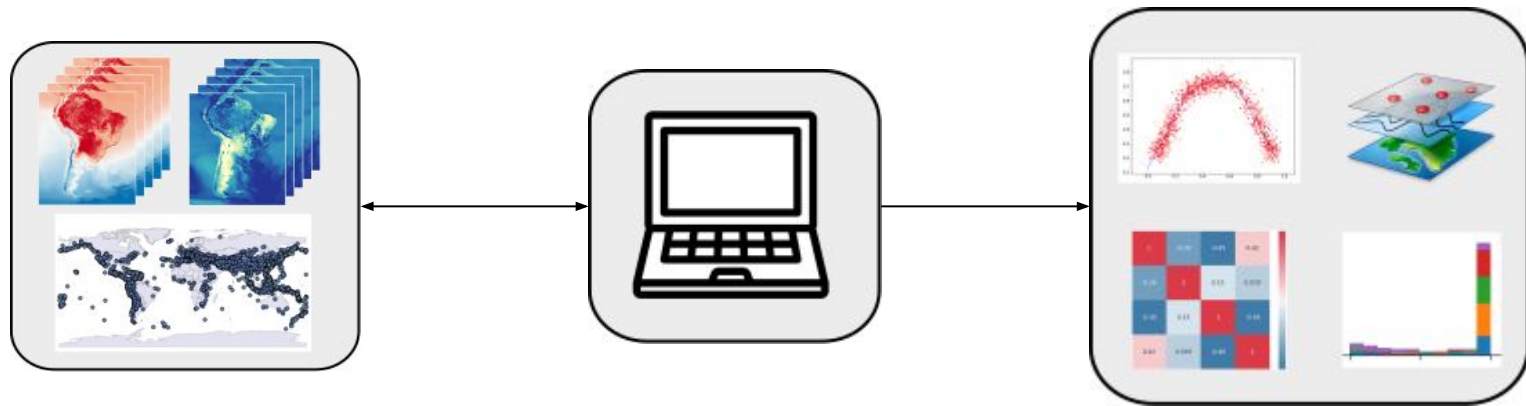


Análise baseada em arquivos

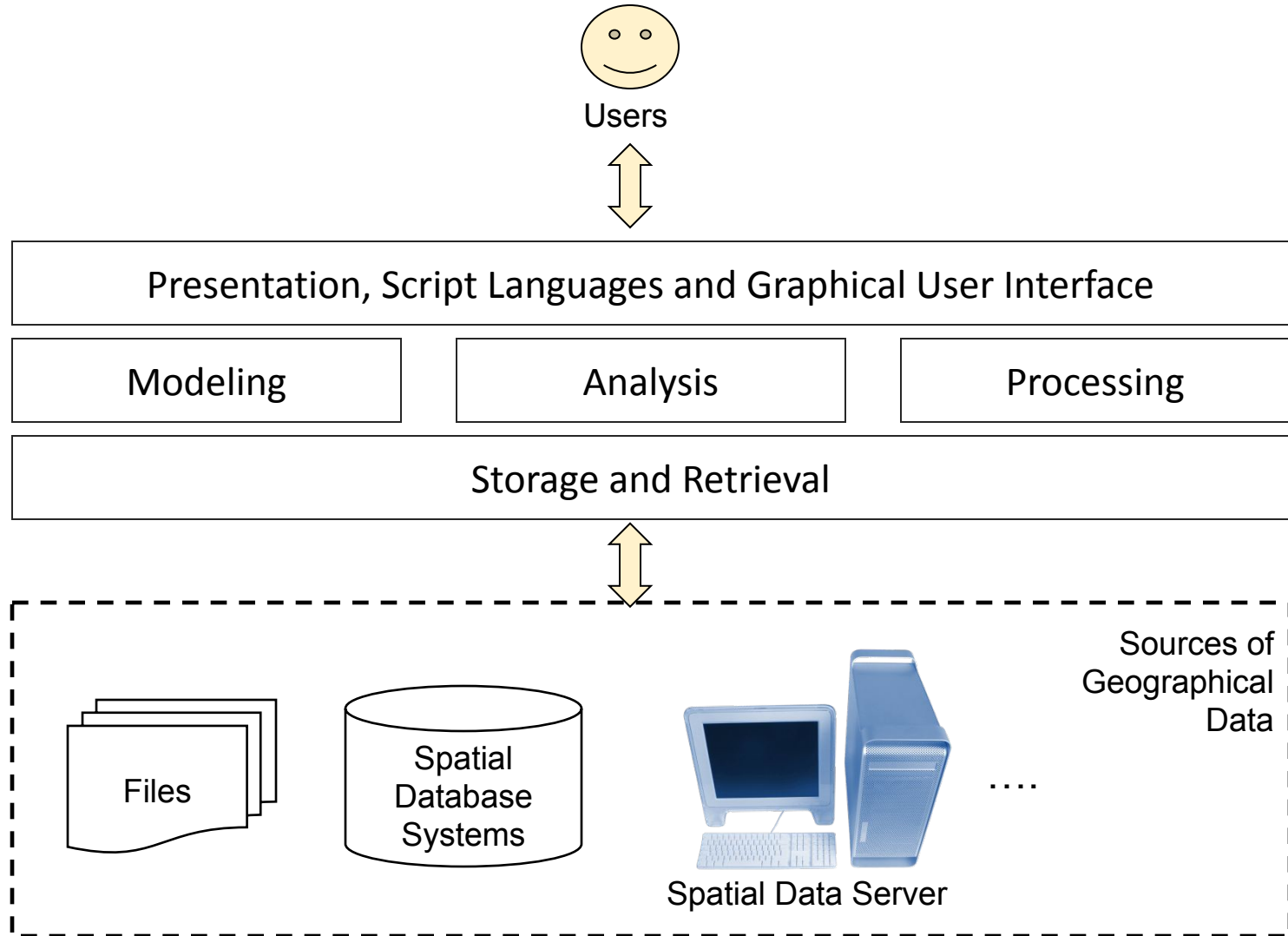


- Excedem **capacidade** computacional
- **Poucos** dados efetivamente **utilizados**

Análise baseada em serviços



Sistema de Informação Geográfica (SIG)

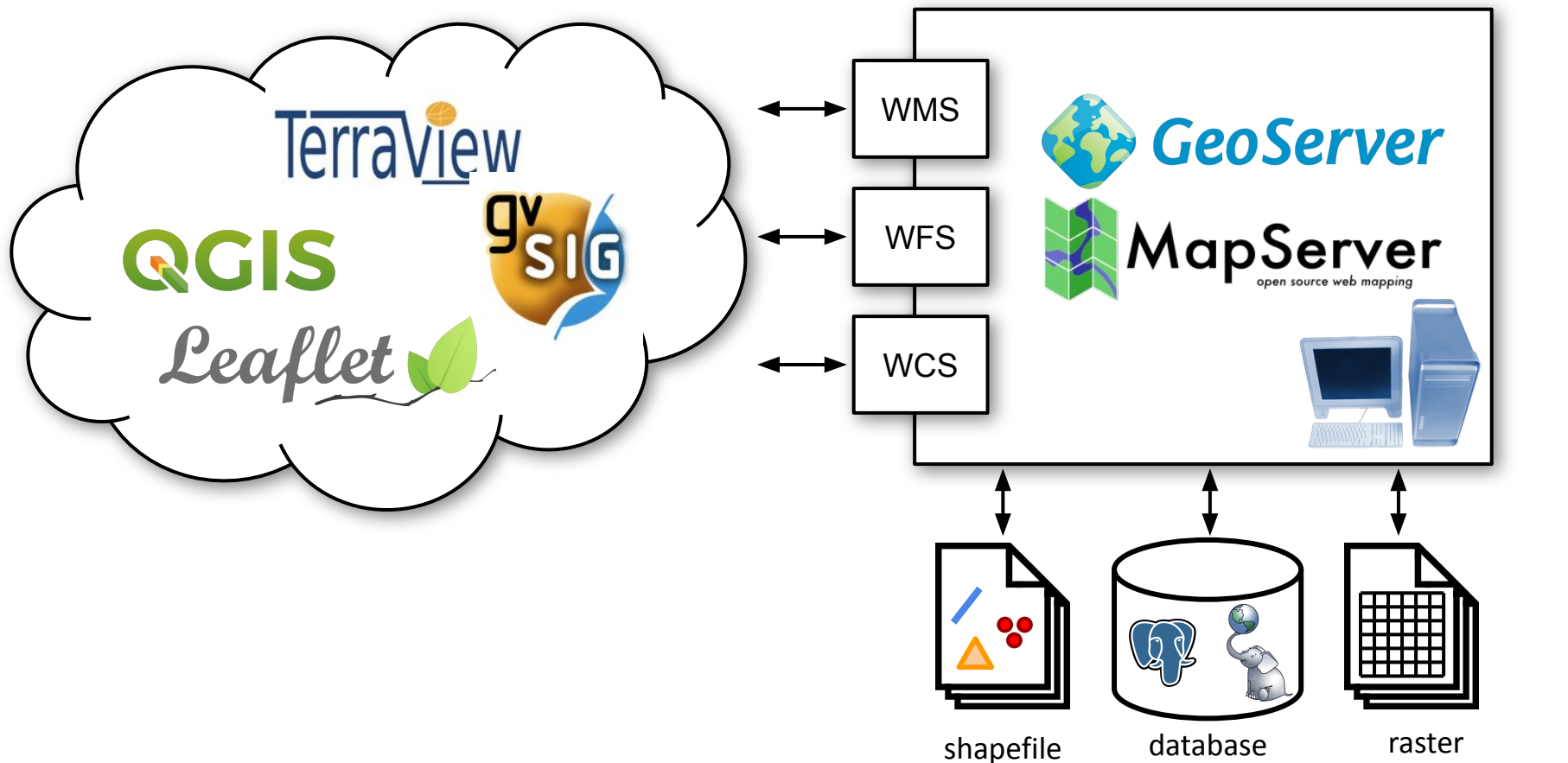


Servidores de Dados Geoespaciais

<http://geoserver.org/>

<http://mapserver.org/>

Servidores de Dados Geoespaciais

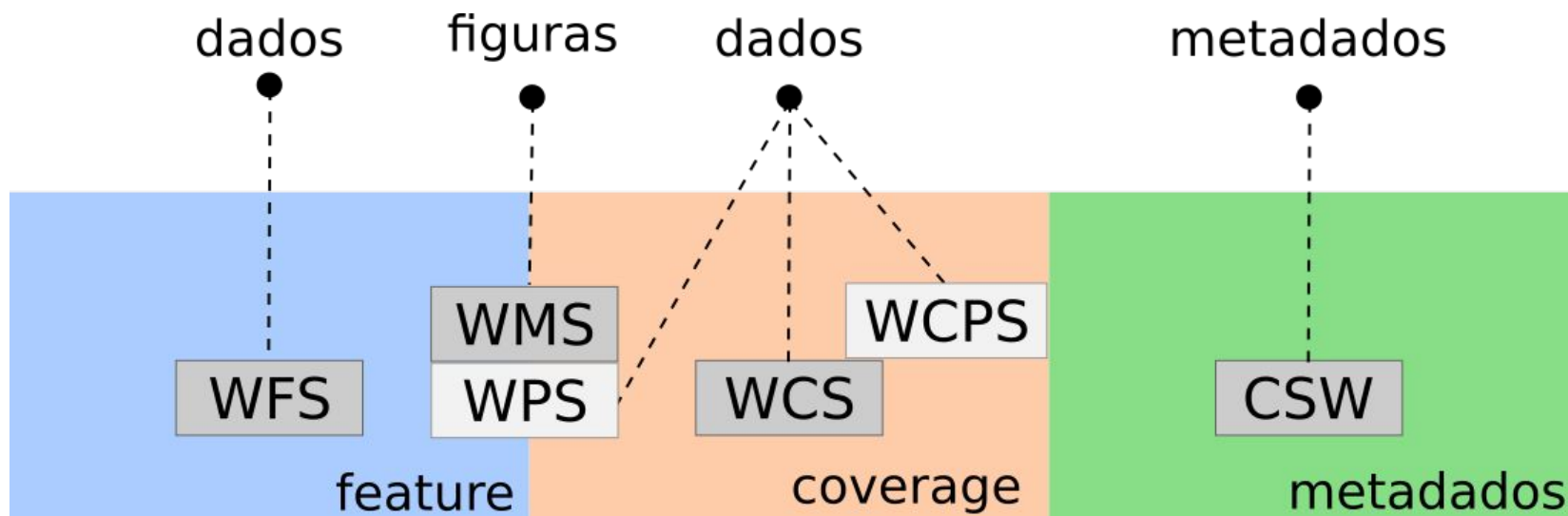


Open Geospatial Consortium (OGC)

- O OGC é um consórcio formado por empresas, universidades e agências governamentais de diversos países.
- Um de seus objetivos é promover o desenvolvimento de padrões que facilitem a interoperabilidade entre sistemas de informação geoespaciais.
- Parte do trabalho do OGC é apresentado sob a forma de especificações abertas de interfaces e padrões de intercâmbio.
- Site: <http://www.opengeospatial.org/>

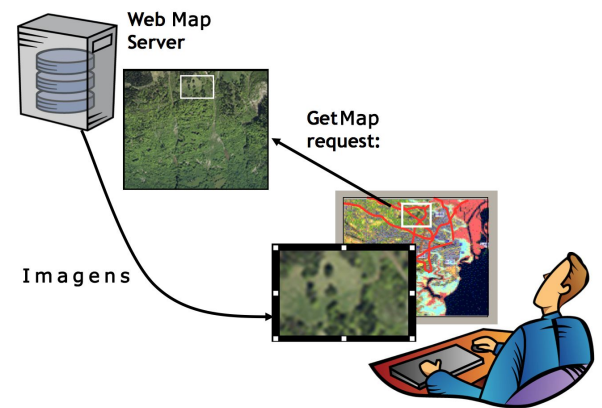
Open Geospatial Consortium (OGC)

- Principais protocolos:
 - WMS, WFS, WCS, CSW, WPS e WCPS



OGC: Web Map Service (WMS)

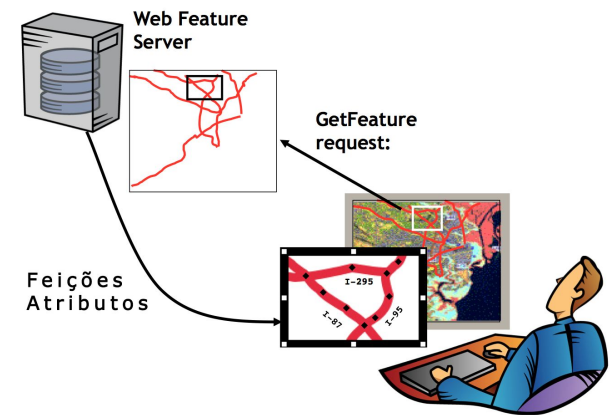
- Coverages e Features → Figuras
- Disponibiliza imagens para visualização
- Requisições:
 - GetCapabilities
 - **GetMap**
 - DescribeLayer
 - GetLegendGraphic



```
url?request=GetMap&layers=map&bbox=x0,y0,xf,yf&width=780&height=330&format=image:png
```

OGC: Web Feature Service (WFS)

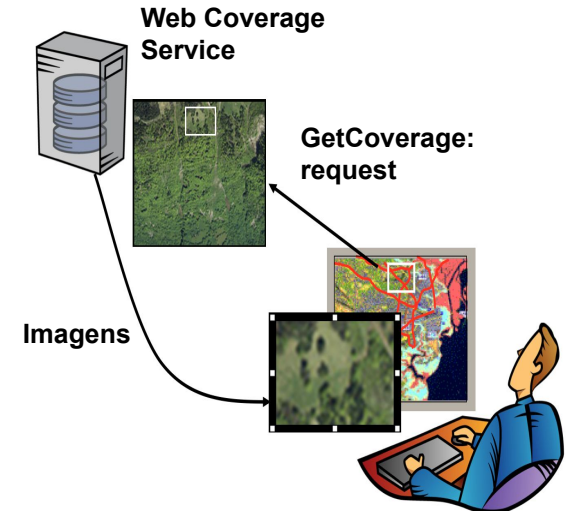
- Feature → Dados (GML)
- Disponibiliza dados vetoriais para proc./análise
- Requisições:
 - GetCapabilities
 - DescribeFeatureType
 - **GetFeature**
 - GetPropertyValue



`url?request=GetFeature&typeName=feature&count=10&sortBy=name`

OGC: Web Coverage Service (WCS)

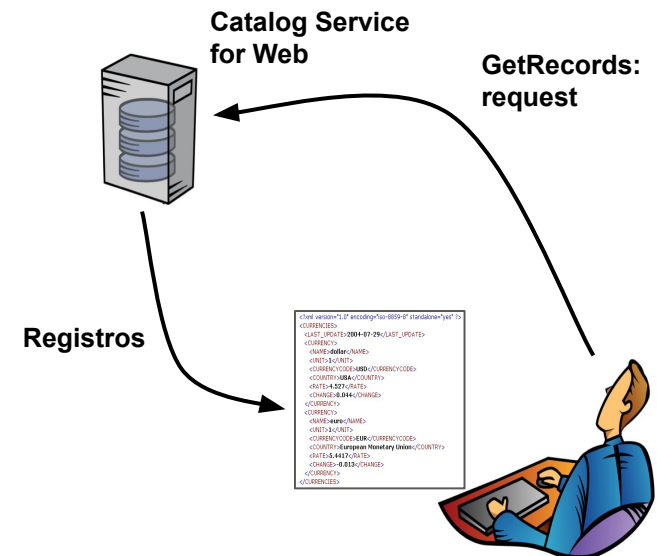
- Coverage → Dados (Tiff, ArcGrid, etc)
- Disponibiliza dados raster para proc./análise
- Requisições:
 - GetCapabilities
 - DescribeCoverage
 - **GetCoverage**



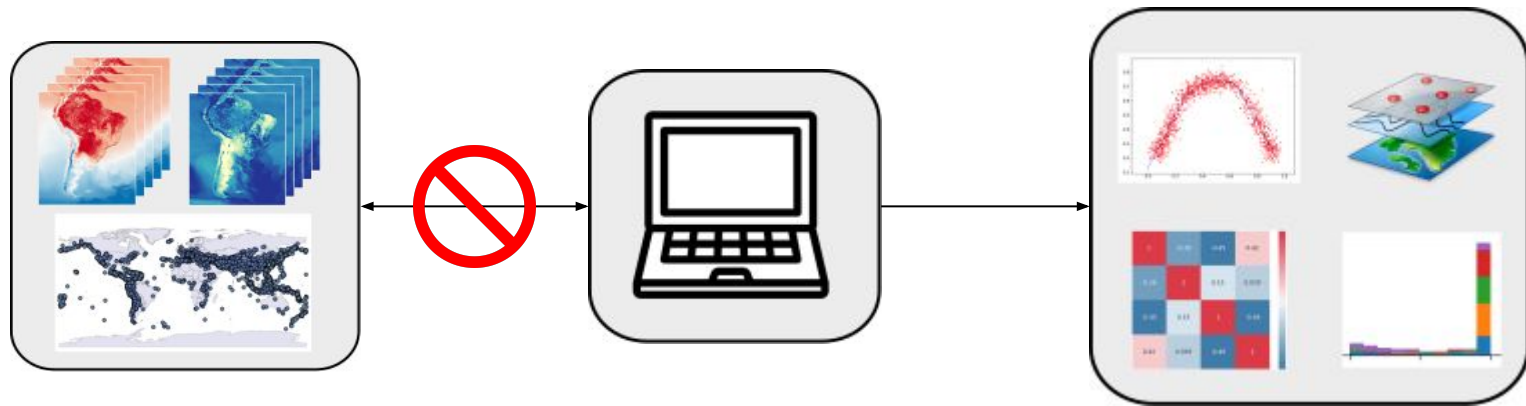
```
url?request=GetCoverage&coverage=cov&bbox=x0,y0,xf,yf&width=780&height=330&format=GTiff
```

OGC: Catalog Service for Web (CSW)

- Registros → Metadados (XML)
- Descoberta de recursos
- Requisições:
 - GetCapabilities
 - **GetRecords**
 - DescribeRecords
 - GetRecordById
 - ...

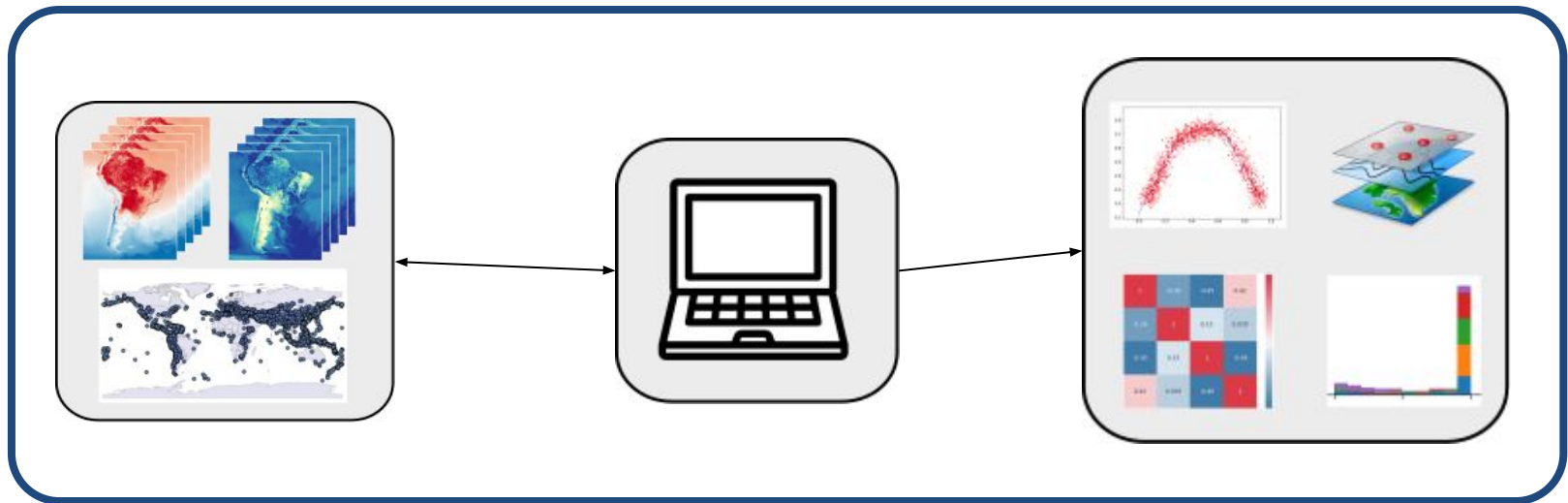


Análise baseada em serviços



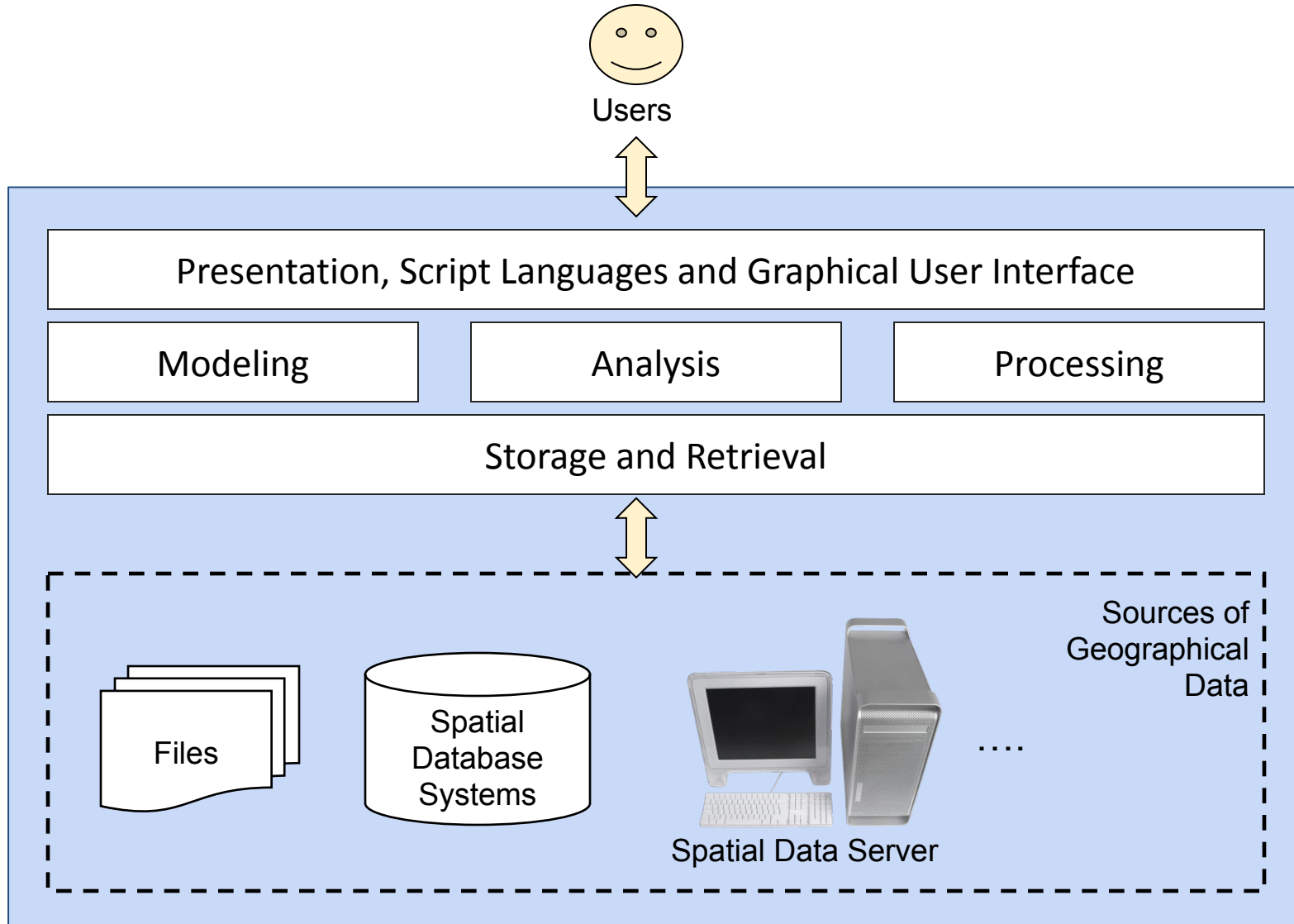
- Movimentação de dados pela rede
- Excedem **capacidade** computacional

Análise em grandes volumes de dados



23/05: Plataformas de Big Data para Dados de
Observação da Terra

Sistema de Informação Geográfica (SIG)



Sistemas nas Nuvens

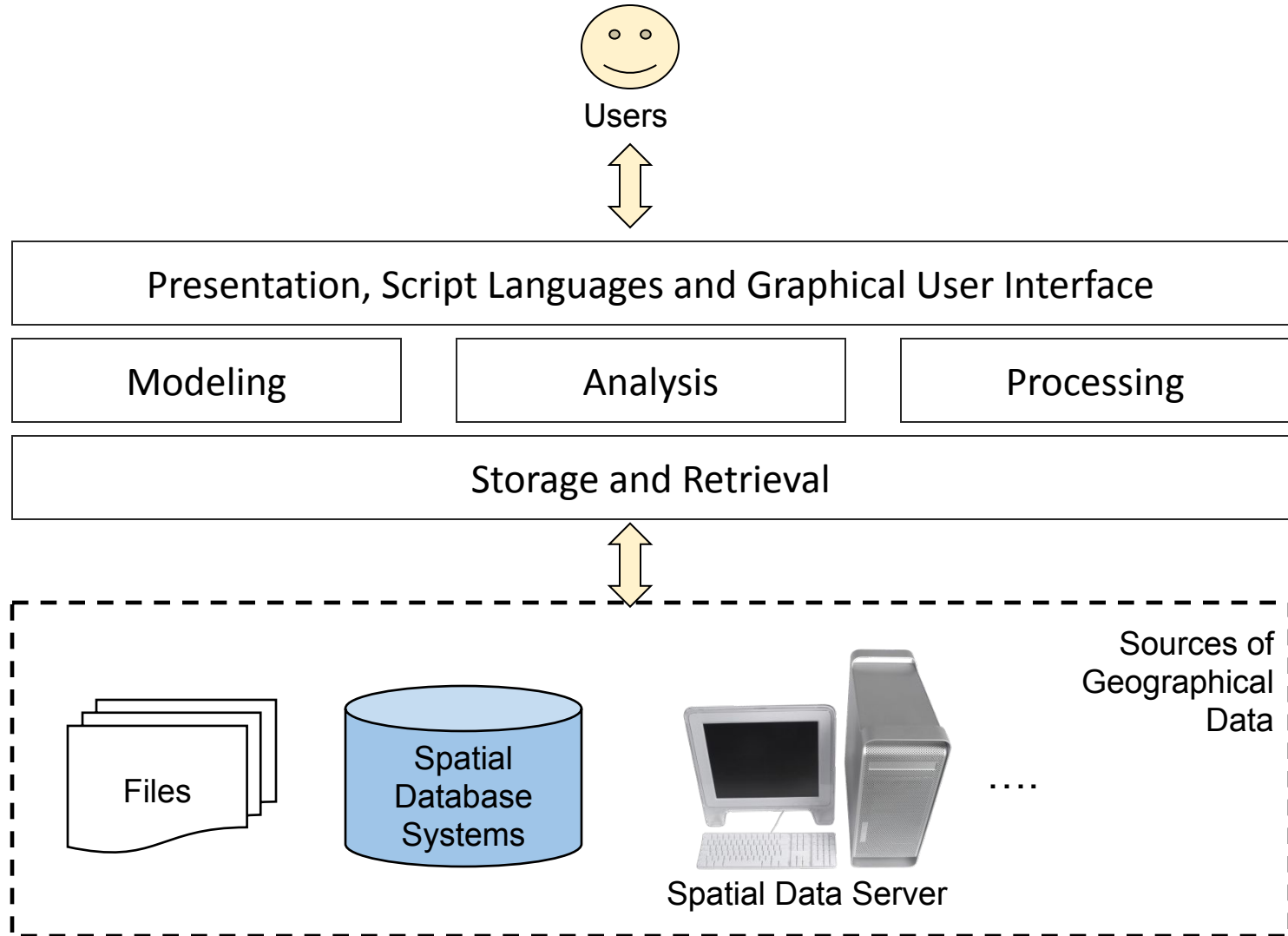
<https://earthengine.google.com>

<https://sepal.io>

<https://www.opendatacube.org>

<http://brazildatacube.dpi.inpe.br/>

Sistema de Informação Geográfica (SIG)

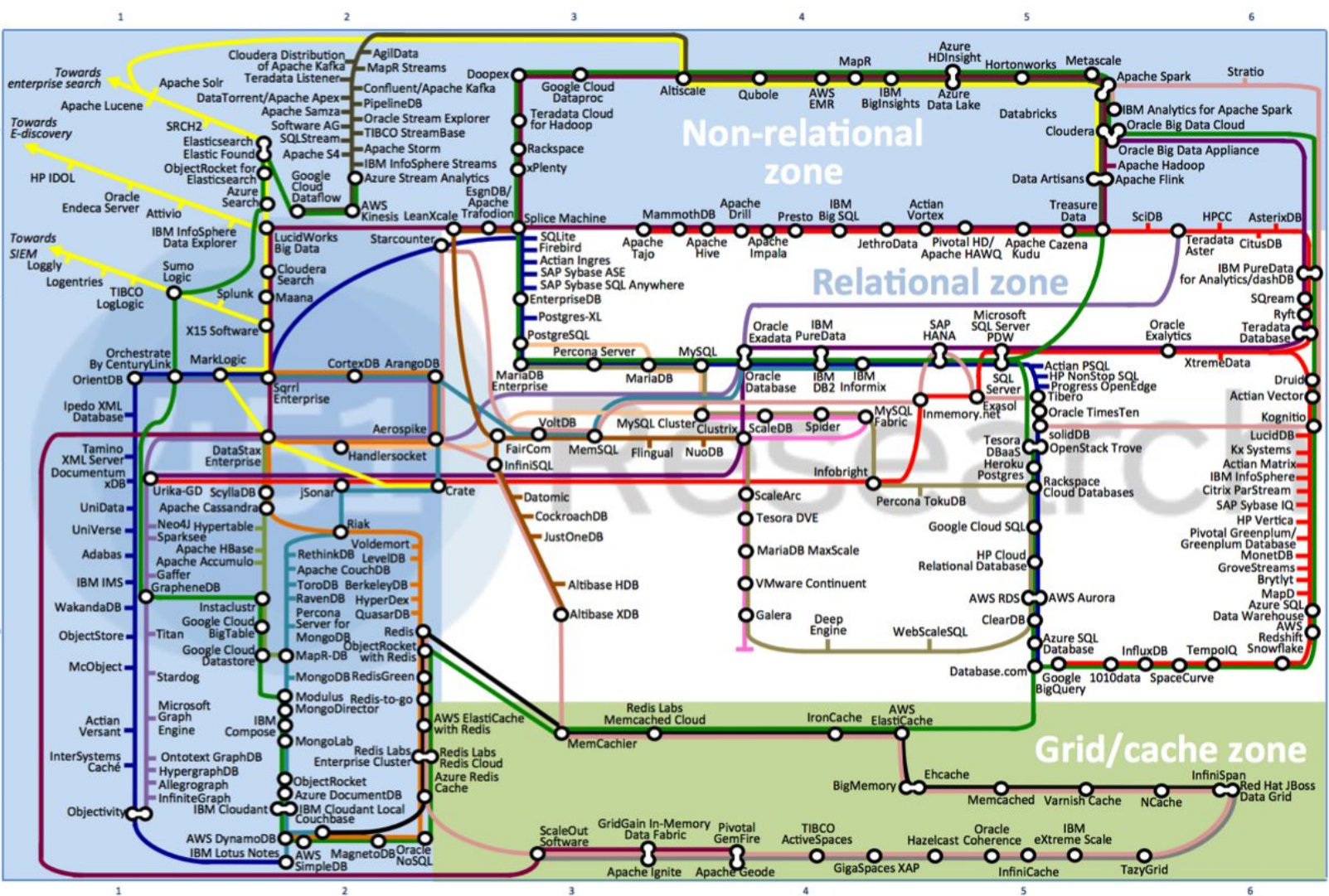


Bancos de Dados

SGBD: uma tecnologia amplamente difundida

- A tecnologia de bancos de dados tem sido um componente fundamental em quase todos os tipos de aplicações:
 - Conta bancária: depósitos e saques
 - Reservas de passagens aéreas
 - Reservas em hotéis
 - Compras de livros, CDs, DVDs e outros bens
 - Busca por artigos em uma revista eletrônica (Transactions of GIS ou ACM digital library)
- Tecnologia que vem expandindo suas fronteiras...

Tecnologias de Bancos de Dados



451 Research

Data Platforms Map
January 2016

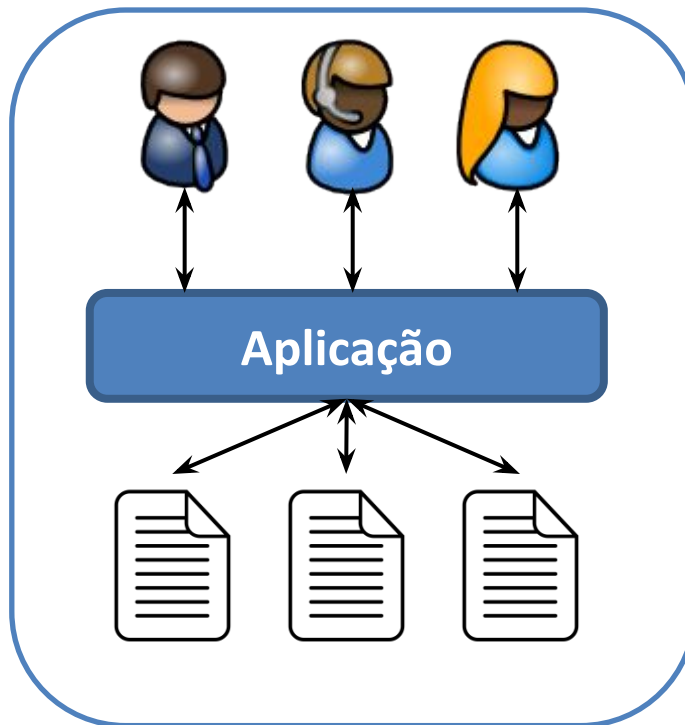
- Key:**
- General purpose
 - Specialist analytic
 - as-a-Service
 - BigTables
 - Graph
 - Document
 - Key value stores
 - Key value direct access
 - Hadoop
 - MySQL ecosystem
 - Advanced clustering/sharding
 - New SQL databases
 - Data caching
 - Data grid
 - Search
 - Appliances
 - In-memory
 - Stream processing

<https://451research.com/state-of-the-database-landscape>

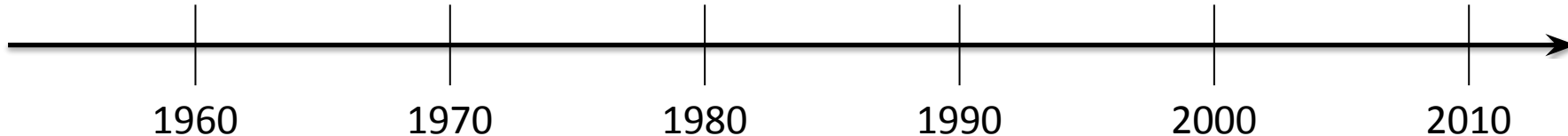
© 2016 by 451 Research LLC. All rights reserved

Tecnologias de Bancos de Dados

Evolução das Tecnologias de Bancos Dados

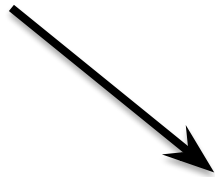


Programas dependentes arquivos

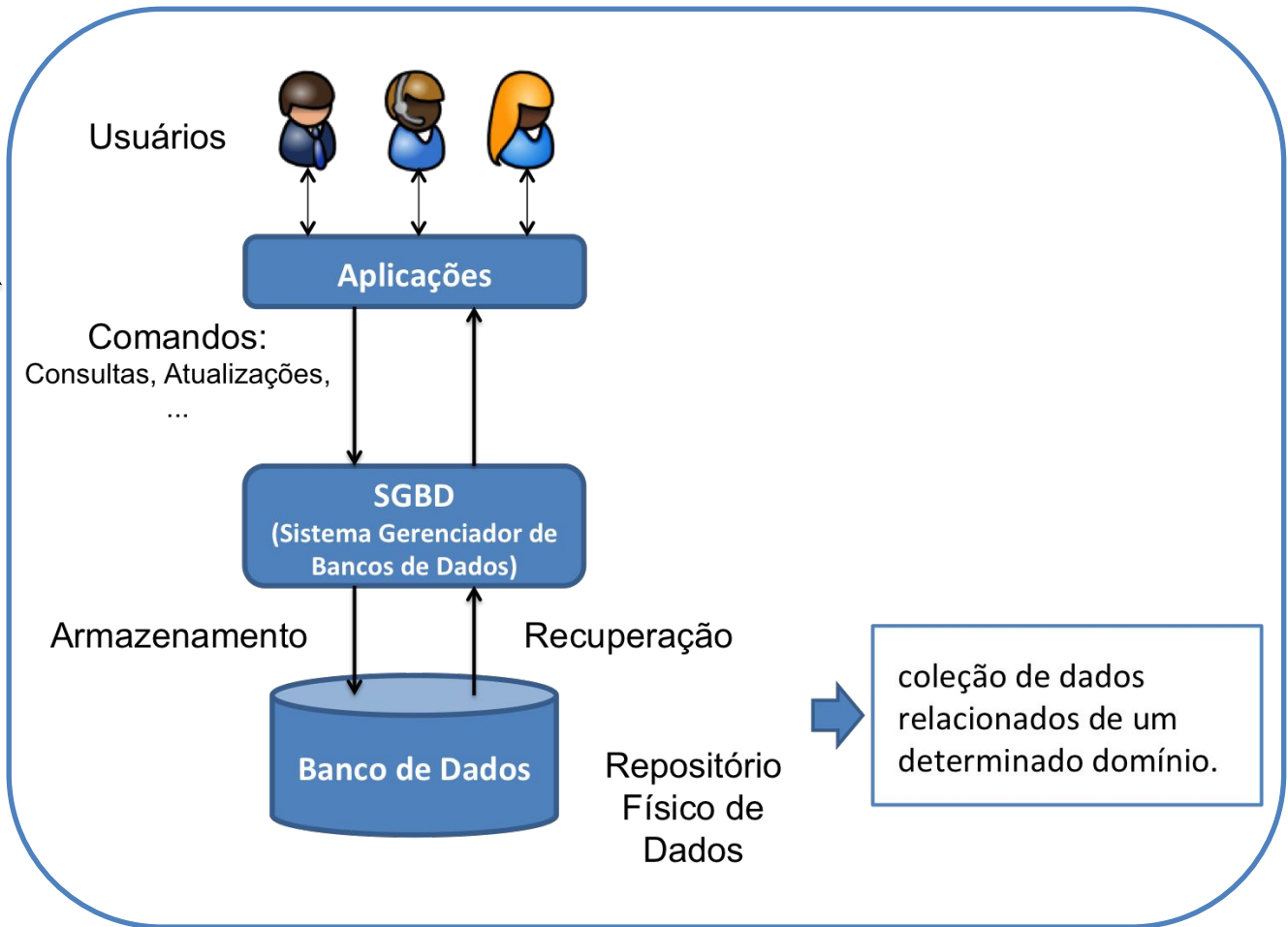


Evolução das Tecnologias de Bancos Dados

Final Anos 60



Programação Generalizada



1960

1970

1980

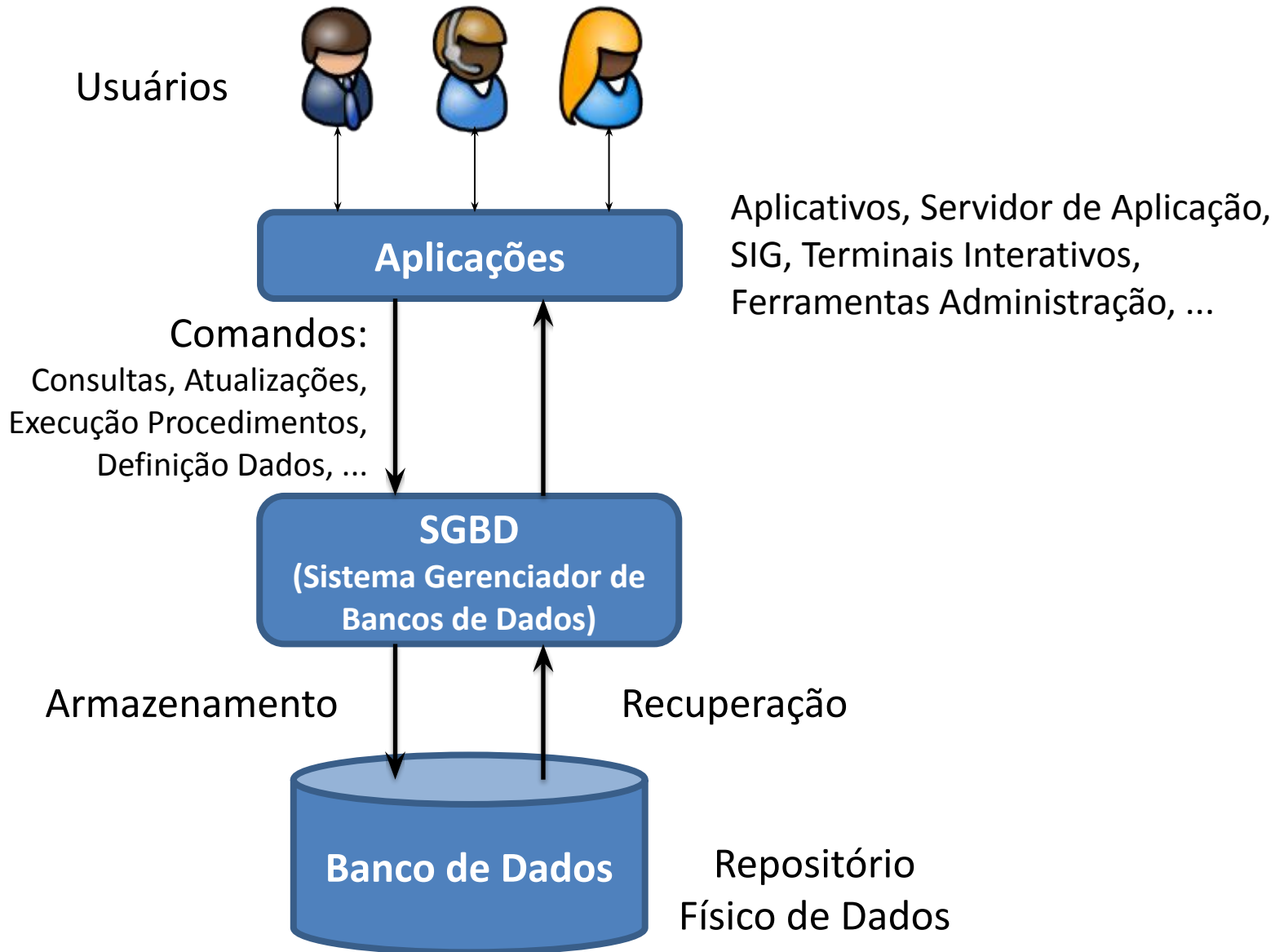
1990

2000

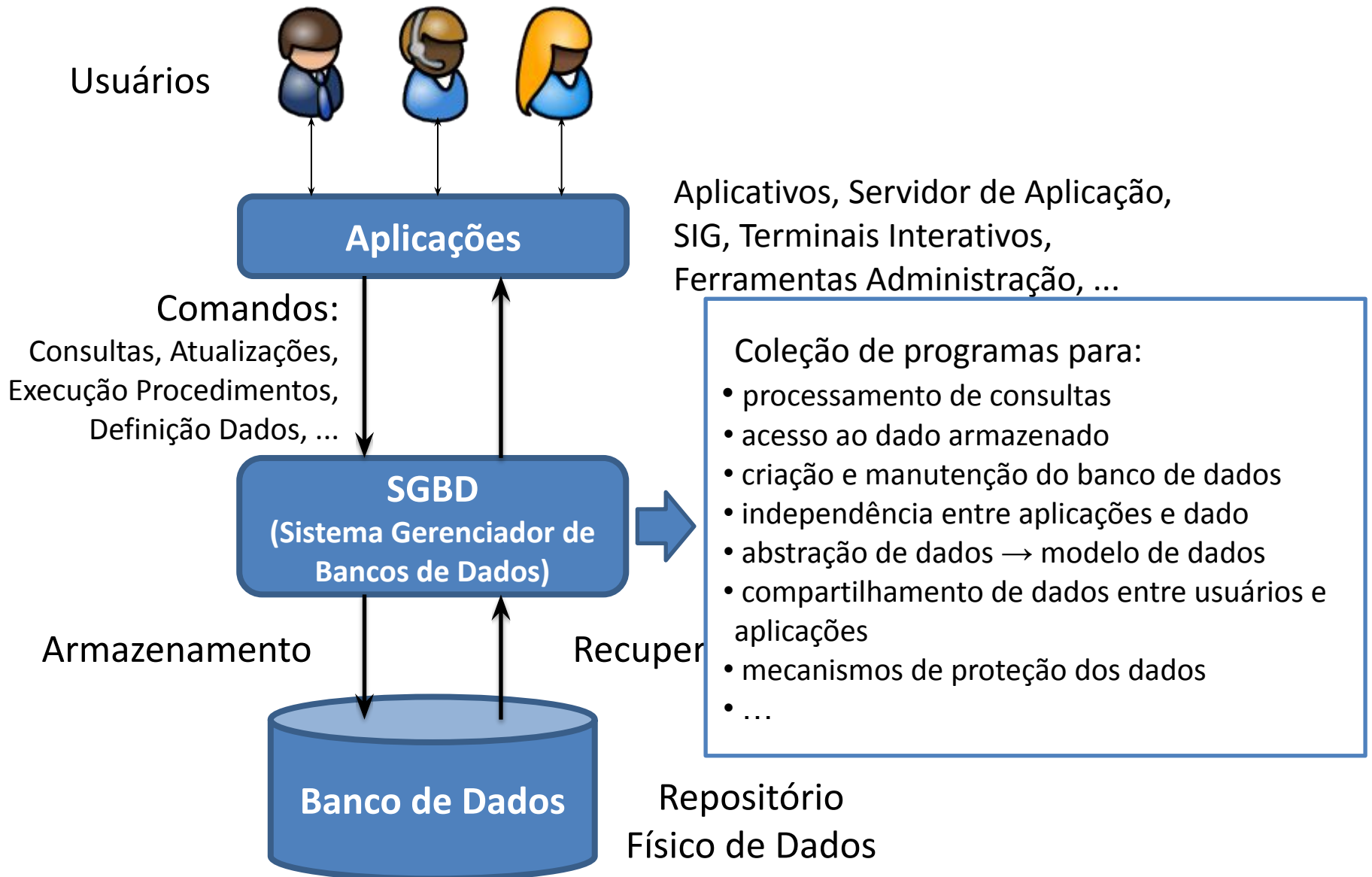
2010

2020

Sistemas de Bancos de Dados

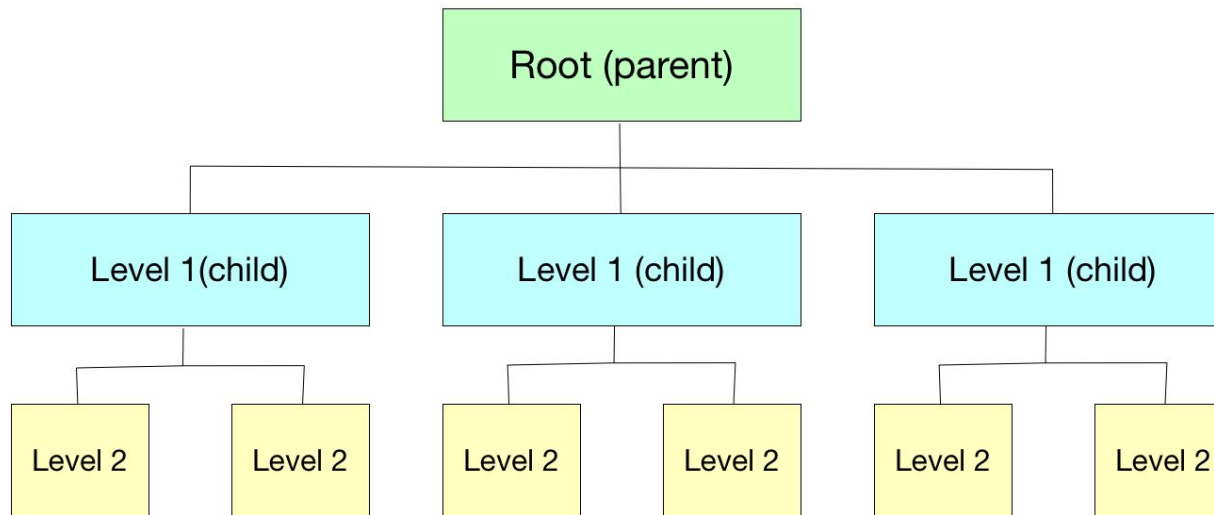


Sistemas de Bancos de Dados



Evolução das Tecnologias de Bancos Dados

The Hierarchical Database Model

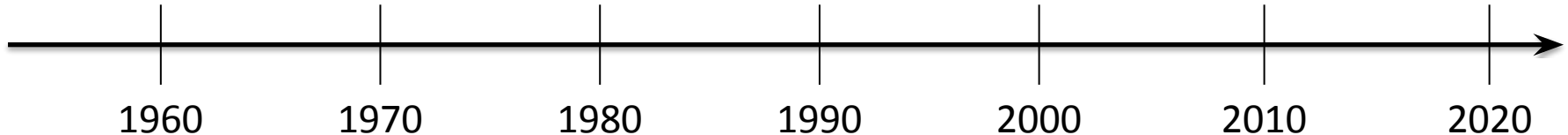


Fonte: [Robert Polding](#)

Modelo Bancos Dados Rede

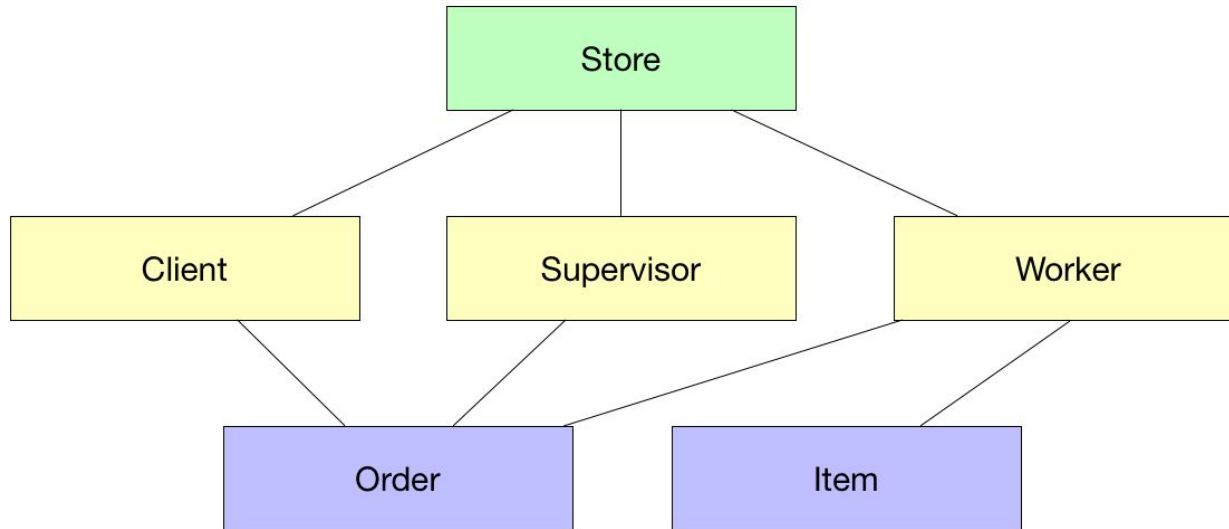


Modelo Bancos Dados Hierárquico



Evolução das Tecnologias de Bancos Dados

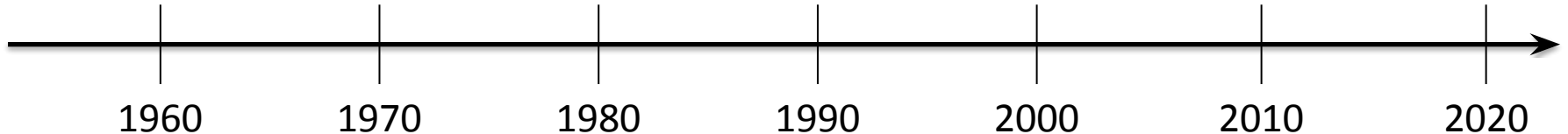
The Network Database Model



Fonte: [Robert Polding](#)

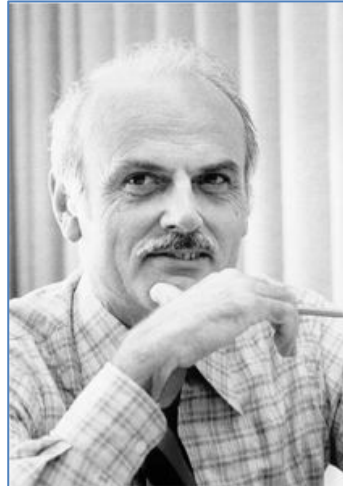
Modelo Bancos Dados Rede

Modelo Bancos Dados Hierárquico



Evolução das Tecnologias de Bancos Dados

ACM Turing Award (1981)



E. F. Codd. 1970. *A relational model of data for large shared data banks*. Communications of the ACM, v. 13, n. 6, June 1970, pp. 377-387.

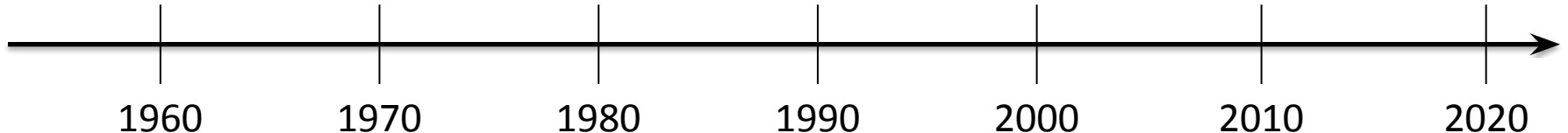
Modelo Relacional

Edgar Frank Codd
Fonte: [Wikipedia](#)

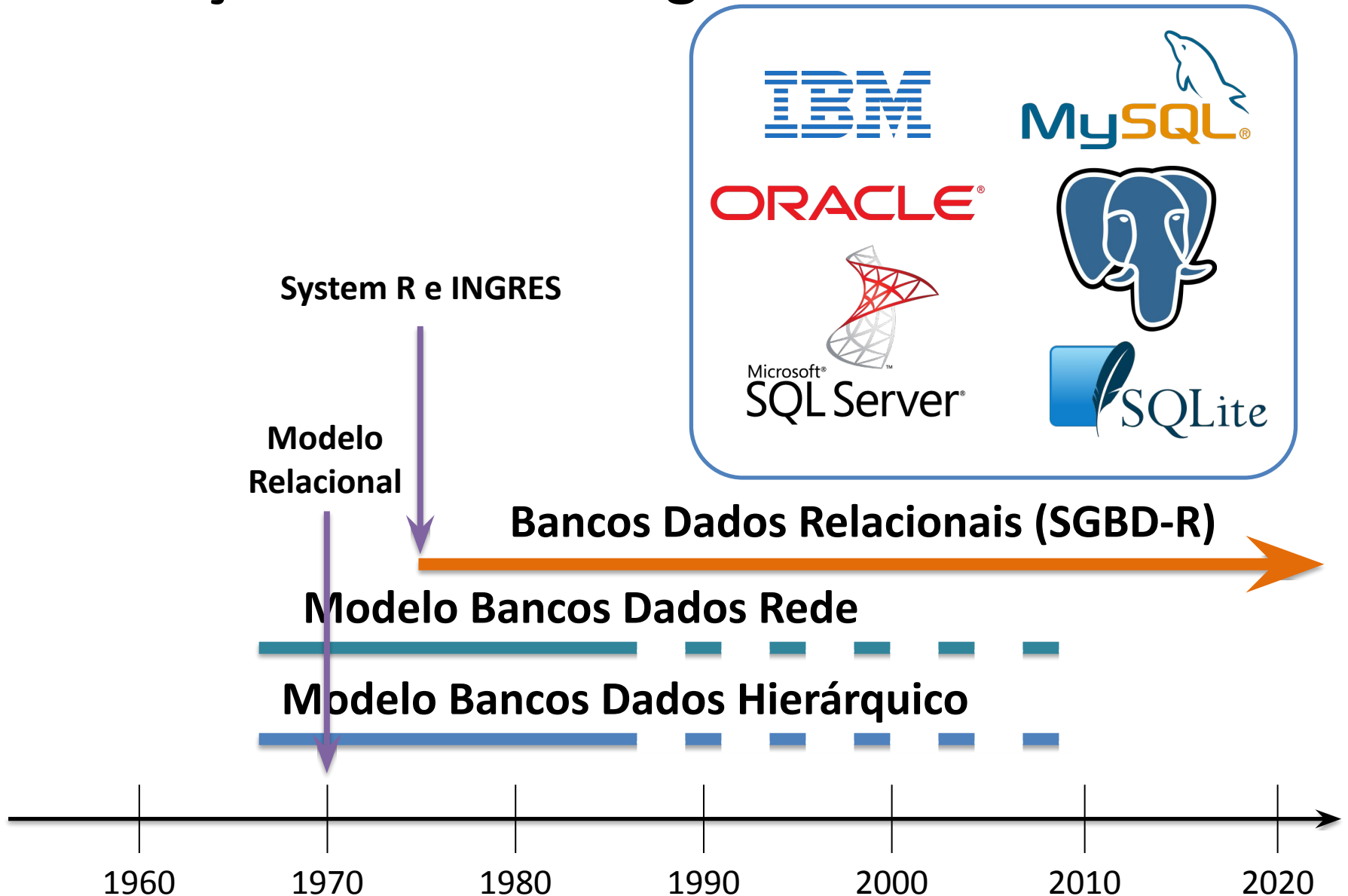
Bancos Dados Relacionais (SGBD-R)

Modelo Bancos Dados Rede

Modelo Bancos Dados Hierárquico



Evolução das Tecnologias de Bancos Dados



Relação (ou Tabela)

- Um banco de dados relacional é organizado em uma coleção de relações (ou tabelas) possivelmente relacionadas entre si.

The diagram illustrates a table structure. On the left, three labels with arrows point to the table: 'Tabela' points to the entire table, 'Colunas' points to the header row, and 'Linhas' points to the data rows. On the right, two labels with brackets indicate parts of the table: 'Esquema Tabela' covers the header row, and 'Instância' covers the data rows.

idades		
cid	nome	populacao
3146107	Ouro Preto	70.227
3140001	Mariana	58.233
3131901	Itabirito	45.449
...

Modelo Relacional

- Toda tabela (ou relação) possui um nome:
 - Em geral, esse nome é único dentro de um mesmo banco de dados.
- As colunas de uma tabela são também chamadas de:
 - campos, domínios ou atributos.
- Cada coluna possui um nome e deve ter um tipo de dado associado:
 - Numérico, Cadeia de Caracteres, Data e Hora, Geométrico, etc.
- As linhas também são conhecidas por:
 - tuplas ou registros.

Chave Primária (Primary Key)

- Campo ou o conjunto de campos cujos valores que identificam unicamente cada linha de uma tabela.

Chave Primária →

países		
id	nome	populacao
1	Alemanha	82.000.000
2	Brasil	190.000.000
...

Chave Primária →

cidades			
cid	nome	populacao	pais_id
191	Correntina	...	2
181	Ouro Preto	...	2
987	Munster	...	1
192	Belmonte	...	2
...

Chave Primária
Composta

cliente_telefone		
ncid	fone	tipo
1	555-7654	residencial
1	345-9876	comercial
2	888-7777	residencial

Relacionamentos entre tabelas

países	
pid	nome
1	Alemanha
2	Brasil
...	...

cidades			
cid	nome	populacao	pais_id
3146107	Ouro Preto	70.227	2
3140001	Mariana	58.233	2
9879999	Munster	291.754	1
3131901	Itabirito	45.449	2
...

países_x_cidades				
pid	p_nome	cid	c_nome	c_populacao
2	Brasil	3146107	Ouro Preto	70.227
2	Brasil	3140001	Mariana	58.233
1	Alemanha	987	Munster	291.754
2	Brasil	3131901	Itabirito	45.449
...

Chave Estrangeira (Foreign Key)

- Coluna ou combinação de colunas, cujos valores aparecem necessariamente na chave primária de uma outra tabela*.

Chave Primária

países		
id	nome	populacao
1	Alemanha	82.000.000
2	Brasil	190.000.000
...

chidades

cid	nome	populacao	pais_id
191	Correntina	...	2
181	Ouro Preto	...	2
987	Munster	...	1
192	Belmonte	...	2
...

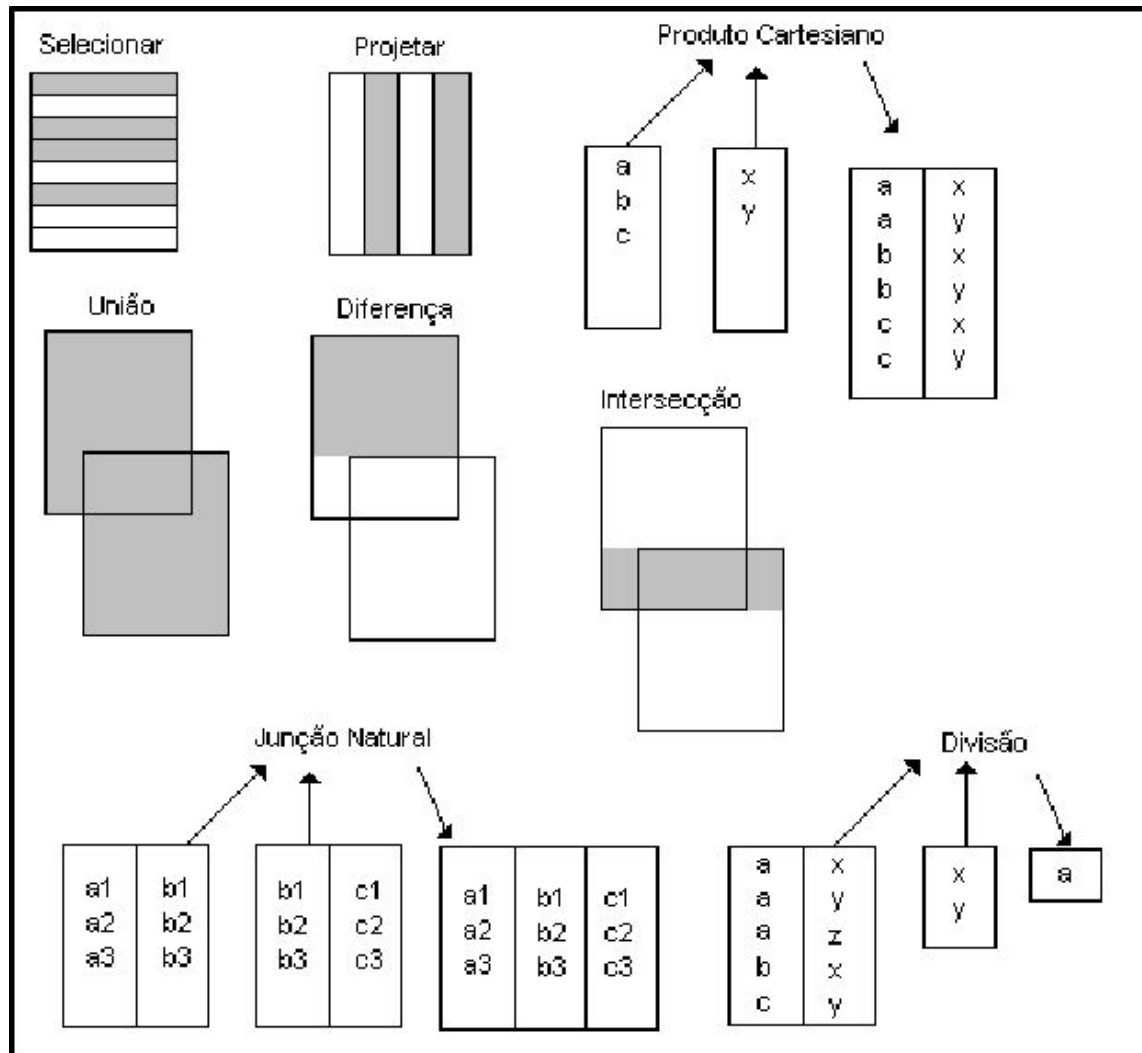
Chave Estrangeira

*uma chave estrangeira não precisa ter o mesmo nome do que a chave primária correspondente na outra tabela (apenas o mesmo domínio)

Álgebra Relacional

- Linguagem formal de consulta.
- Conjunto de operações que usam uma ou mais relações como entrada e geram uma nova relação de saída:
 - operação $(R_1) \rightarrow R_n$
 - operação $(R_1, R_2) \rightarrow R_n$
- Operações básicas:
 - Operações unárias: seleção (σ), projeção (π).
 - Operações binárias: produto cartesiano (\times), junção (θ), interseção, união e diferença.
- Os operadores podem ser combinados de forma a realizar operações mais complexas.

Álgebra Relacional: Operadores



Fonte: C. J. Date (1993)

Álgebra Relacional: Seleção

- **Exemplo:**

- **Operador de Seleção:** seleciona tuplas de uma relação que satisfazem um certo predicado (ou condição).
- **Consulta:** selecionar as cidades com população acima de 60.000 habitantes.

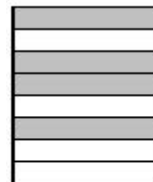
cidades			
cid	nome	populacao	pais_id
...	Ouro Preto	70.227	2
...	Mariana	58.233	2
...	Munster	291.754	1
...	Itabirito	45.449	2

Tabela/Relação de Entrada

$\sigma_{populacao > 60000}(cidades)$



Selecionar



nova_relacao			
cid	nome	populacao	pais_id
...	Ouro Preto	70.227	2
...	Munster	291.754	1

Tabela/Relação de Saída

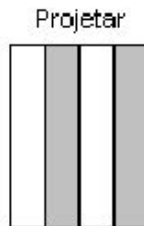
Álgebra Relacional: Projeção

- Este operador gera uma nova relação contendo apenas as colunas desejadas de uma relação de entrada.
- Exemplo: projetar o atributo nome sobre a relação “países”.

países		
id	nome	populacao
1	Alemanha	82.000.000
2	Brasil	190.000.000
...

Tabela de Entrada

$\pi_{nome}(países)$



nova_relacao
nome
Alemanha
Brasil
...

Tabela de Saída

Álgebra Relacional: Produto Cartesiano

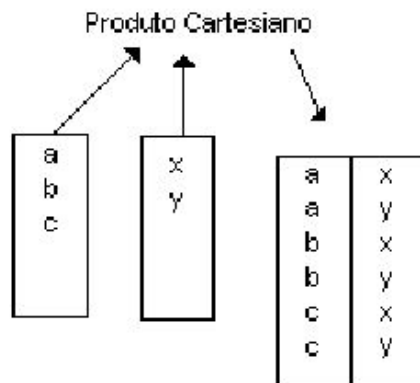
- Este operador gera uma nova relação formada pela combinação de todas as tuplas de duas relações de entrada.

$$(países)\theta(cidades)$$

países		
id	nome	populacao
1	Alemanha	82.000.000
2	Brasil	190.000.000
...

×

cidades			
cid	nome	populacao	pais_id
191	Correntina	...	2
181	Ouro Preto	...	2
987	Munster	...	1
192	Belmonte	...	2
...

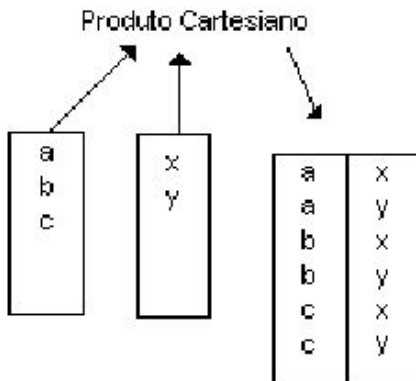


Álgebra Relacional: Produto Cartesiano

- Este operador gera uma nova relação formada pela combinação de todas as tuplas de duas relações de entrada.

$(países)\theta(cidades)$

nova_relacao						
id	nome	populacao	cid	nome	populacao	pais_id
1	Alemanha	82000000	191	Correntina	...	2
1	Alemanha	82000000	181	Ouro Preto	...	2
1	Alemanha	82000000	987	Munster	...	1
1	Alemanha	82000000	192	Belmonte	...	2
2	Brasil	190.000.000	191	Correntina	...	2
2	Brasil	190.000.000	181	Ouro Preto	...	2
2	Brasil	190.000.000	987	Munster	...	1
2	Brasil	190.000.000	192	Belmonte	...	2
...

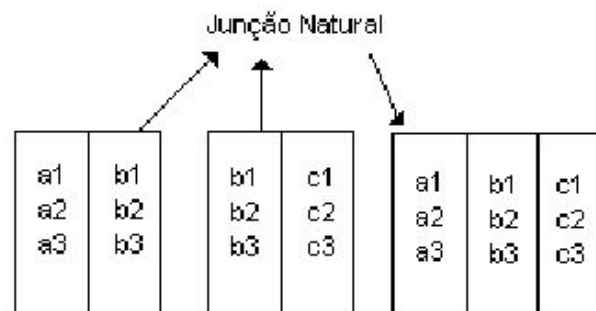


Álgebra Relacional: Junção (Join)

- Produto cartesiano seguido de uma seleção.

$$(paises)\theta(cidades) \Leftrightarrow \sigma_{paises.id=cidades.pais_id} (paises \times cidades)$$

nova_relacao						
id	nome	populacao	cid	nome	populacao	pais_id
1	Alemanha	82000000	987	Munster	...	1
2	Brasil	190.000.000	191	Correntina	...	2
2	Brasil	190.000.000	181	Ouro Preto	...	2
2	Brasil	190.000.000	192	Belmonte	...	2
...



Linguagem de Consulta: SQL

- O modelo relacional é a base para linguagens de alto nível:
 - Álgebra/Cálculo Relacional → Linguagem Declarativa → ISO/SQL (Structured Query Language)

```
CREATE TABLE paises
(
  pid INT4 PRIMARY KEY,
  nome VARCHAR(50)
);
```

DDL: Definição
Dados

paises	
pid	nome

paises	
pid	nome
1	Alemanha
2	Brasil
...	...

DML:
Manipulação
Dados

```
INSERT INTO paises
VALUES (1, 'Alemanha');

INSERT INTO paises
VALUES (2, 'Brasil');
```

Linguagem de Consulta: SQL

- O modelo relacional é a base para linguagens de alto nível:
 - Álgebra/Cálculo Relacional → Linguagem Declarativa → ISO/SQL (Structured Query Language)

cidades			
cid	nome	populacao	pais_id
...	Ouro Preto	70.227	2
...	Mariana	58.233	2
...	Munster	291.754	1
...	Itabirito	45.449	2

```
SELECT nome  
FROM cidades  
WHERE populacao > 60000
```

Consulta

resultado
nome
Ouro Preto
Munster

Evolução das Tecnologias de Bancos Dados

Aplicações emergentes e novas demandas:
CAD, SIG, Multimedia,
OLAP, Real-time,
Científicas

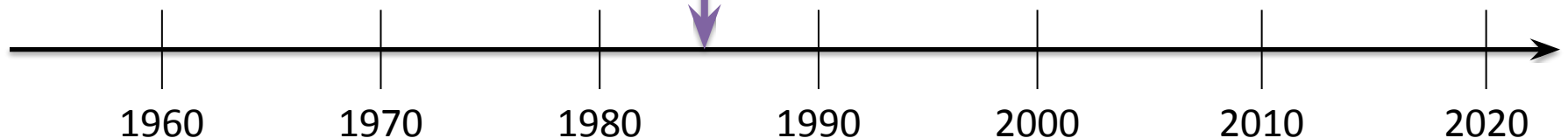
Período de muita pesquisa sobre extensibilidade dos SGBDs

Protótipos de pesquisa no final dos anos 80 voltados a SIG:
Probe, DASDBS GEO-Kernel, Gral, SIRO-DBMS, Starburst, Geo++, GéoSabrina, GODOT, GeoO₂, Paradise

Bancos Dados Relacionais (SGBD-R)

Modelo Bancos Dados Rede

Modelo Bancos Dados Hierárquico



Gral: An Extensible Relational Database System for Geometric Applications

Ralf Hartmut Güting

Fachbereich Informatik, Universität Dortmund
D-4600 Dortmund 50, West Germany

Abstract: We describe the architecture of a relational database system that is extensible by user-defined data types and operations, including relation operations. The central concept is to use languages based on many-sorted algebra to represent queries as well as query execution plans. This leads to a simple and clean extensible system architecture, eases the task of an application developer by providing a uniform framework, and also simplifies rule-based optimization. As a case study the extensions needed for a geometric database system are considered.

1. Introduction

Much of the database research of recent years was aimed at providing a better support for non-standard applications such as office information systems, geographic information systems, CAD databases, etc. A common need of these applications is the representation and manipulation of more complex objects than those representable by a tuple of a relation in the traditional relational model, for example, an office form, a complete map or a river, say, in a geographic information system, or the design of a VLSI circuit.

A fundamental choice for the representation of a complex

A lot of work has been done to support the modeling of visible object structures. Enhancements to the relational model have been proposed by linking together tuples to represent an object either explicitly [Co79, HaL82] or implicitly, through the use of nested relations [ScS86]. Most of the more recent data models and system proposals do also support structural object orientation, for example [MaD86, CaDV88, PiT86].

The idea of allowing application-specific abstract data types as base types, or attribute domains, of a database system was perhaps first put forward in [StRG83]. Since base types need to be implemented in a programming language and because they are application-specific, a user must be able to implement such a type and to add it to a database system. This observation has led to efforts by several groups to construct *extensible* database systems. Two directions can be distinguished. One is to select a data model and to implement for this data model a system with well-defined interfaces for user extensions. This is the approach chosen by the POSTGRES [StR86] and Starburst [Schw86] projects, based on the relational model, and within the PROBE project [Daya87] for an extended functional data model. A different view is taken in the EXODUS [Care86] and GENESIS [Bato86] projects where a collection of powerful tools for

R. H. Güting. 1989. Gral: an extensible relational database system for geometric applications. In Proceedings of the 15th international conference on Very large data bases (VLDB '89). Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 33-44.

THE **GEO++** SYSTEM: AN EXTENSIBLE GIS

Tom Vijlbrief

TNO Institute for Perception,

P.O. Box 23, 3769 ZG Soesterberg, The Netherlands.

Email: tom@izf.tno.nl

and

Peter van Oosterom

TNO Physics and Electronics Laboratory,

P.O. Box 96864, 2509 JG The Hague, The Netherlands.

Email: oosterom@fel.tno.nl

In this paper we present a classification of the architectures of Geographic Information Systems (GIS). Most commercial GISs are closed. This means that if certain functionality is not available, it is impossible for the users to extend or modify the system for their own purpose. We then present a solution based on the extensible database management system “Postgres”, in which new data types and operators may be defined. The resulting extensible GIS is called *GEO++*. We illustrate this powerful capability with two examples.

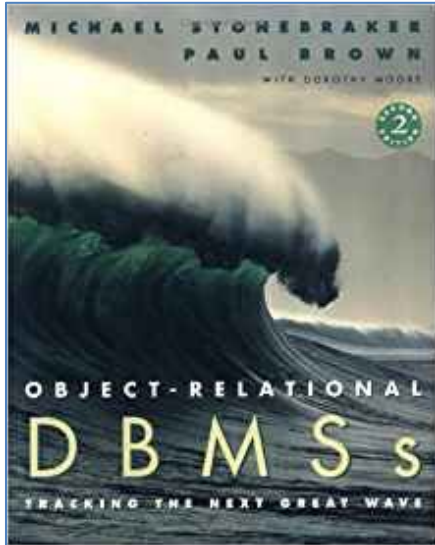
1 Introduction

Most commercial Geographic Information Systems (GISs) are based on a relational DataBase Management System (DBMS), such as Oracle or Ingres. One obvious drawback of the standard DBMSs is that they cannot manipulate geographic data. That is, there are no ge-

Rowe, & Hirohama, 1990) that we implemented. Section 4 enumerates the basic capabilities of our Postgres GIS front-end *GEO++*. The implementation of the system has been described in an earlier paper (van Oosterom & Vijlbrief, 1991). The real power of *GEO++* is demonstrated in Section 5, in which the system is extended with user-defined types.

Vijlbrief, T. and van Oosterom, P. The *GEO++* system: An extensible GIS. Proceedings of the Fifth **International Symposium on Spatial Data Handling**, Charleston, SC, 1992.

Evolução das Tecnologias de Bancos Dados



Stonebraker et al. (1996)

Difusão dos SGBD-OR

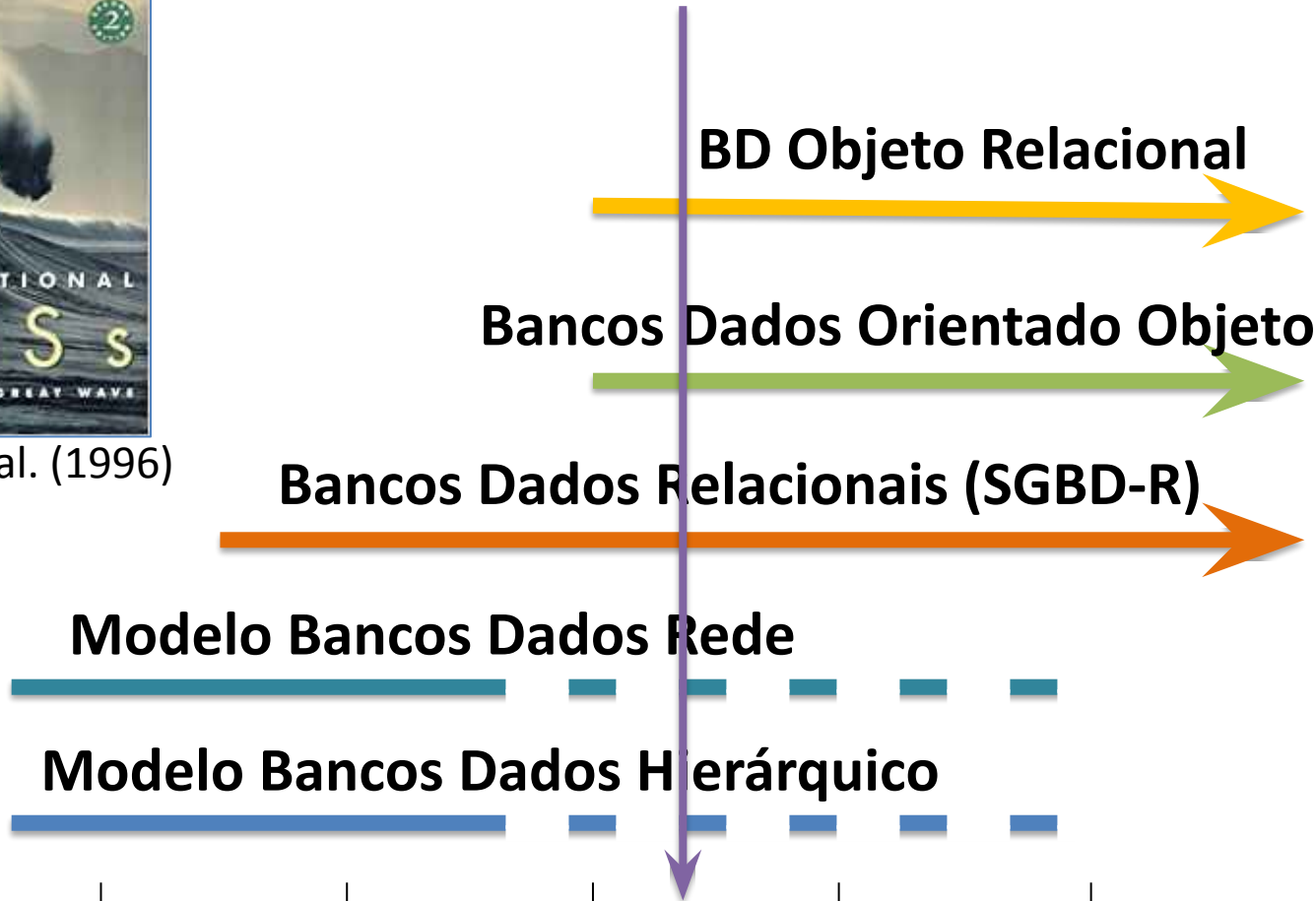
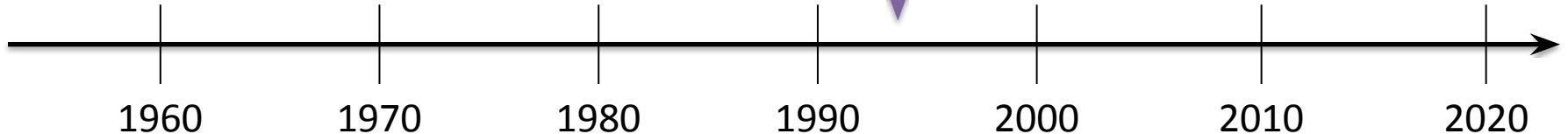
BD Objeto Relacional

Bancos Dados Orientado Objeto

Bancos Dados Relacionais (SGBD-R)

Modelo Bancos Dados Rede

Modelo Bancos Dados Hierárquico



Michael Stonebraker*



Fonte: [Wikipedia](#) (2016)

- Pesquisador na Universidade da Califórnia, Berkeley (70-2000).
- Atualmente no MIT
- Liderou diversos times de estudantes que criaram implementações reais de SGBDs:
 - Ingres (1974-1984)
 - Postgres (1985-1993)
 - Mariposa (Cohera) (1997-2001)
 - Aurora (StreamBase)
 - C-Store (Vertica)
 - H-Store (VoltDB)
 - SciDB

*ACM Turing Award (2014)

Sistemas Gerenciadores de Bancos de Dados Objeto-Relacional (SGBD-OR)

- Em um SGBD-OR, um **tipo de dado** (*data type*) é definido por uma **representação de armazenamento** juntamente com **operadores e funções** sobre este tipo.
- Esta flexibilidade do sistema de tipos dos SGBD-OR os tornam uma ferramenta extremamente poderosa para modelar aplicações mais complexas, tais como GIS.

SGBD-OR: User Defined Types

```
CREATE TYPE GEO_POINT AS  
(  
  x REAL,  
  y REAL,  
  srid INTEGER  
);
```

O comando **CREATE TYPE** permite definir a **representação de armazenamento** para o **tipo geo_point**.

SGBD-OR: User Defined Types

```
CREATE TABLE sedes_municipais  
(  
  id INTEGER PRIMARY KEY,  
  location GEO_POINT  
);
```



```
INSERT INTO sedes_municipais  
VALUES (1, '(1, 2, 4326)::GEO_POINT);
```

SGBD-OR: User Defined Functions

```
CREATE OR REPLACE FUNCTION distance(first GEO_POINT, second GEO_POINT)
RETURNS REAL
AS $$
DECLARE
    dx REAL;
    dy REAL;
BEGIN
    dx = (first.x - second.x) * (first.x - second.x);
    dy = (first.y - second.y) * (first.y - second.y);
    RETURN sqrt(dx + dy);
END;
$$
LANGUAGE plpgsql;
```

O comando **CREATE FUNCTION** permite **criar** ou **estender** a álgebra de um determinado tipo de dado, neste caso o tipo **GEO_POINT**.

SGBD-OR: User Defined Functions

```
CREATE OR REPLACE FUNCTION less_than(first GEO_POINT, second GEO_POINT)
RETURNS REAL
AS $$
BEGIN
    IF(first.x < second.x)
    THEN RETURN TRUE;
    END IF;

    IF(first.x > second.x)
    THEN RETURN FALSE;
    END IF;

    ...
    RETURN FALSE;
END;
$$
LANGUAGE plpgsql;
```

O comando **CREATE FUNCTION** permite **criar** ou **estender** a álgebra de um determinado tipo de dado, neste caso o tipo **GEO_POINT**.

SGBD-OR: User Defined Functions

```
SELECT distance('(1, 2, 4326)::GEO_POINT, '(10, 20, 4326)::GEO_POINT);
```

```
SELECT less_than('(1, 2, 4326)::GEO_POINT, '(10, 20, 4326)::GEO_POINT);
```

```
SELECT less_than('(1, 2, 4326)::GEO_POINT, '(-1, 2, 4326)::GEO_POINT);
```

As funções definidas pelo usuário passam a fazer, automaticamente, parte da **linguagem de consulta do SGBD**.

SGBD-OR: Sobrecarga de Operadores

```
CREATE OPERATOR <  
(  
  leftarg = GEO_POINT,  
  rightarg = GEO_POINT,  
  procedure = less_than,  
  commutator = >,  
  negator = >=  
);
```



```
SELECT '(1, 2, 4326)'::GEO_POINT < '(10, 2, 4326)'::GEO_POINT;
```

SGBD-OR: UDTs mais Complexos

```
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);
```

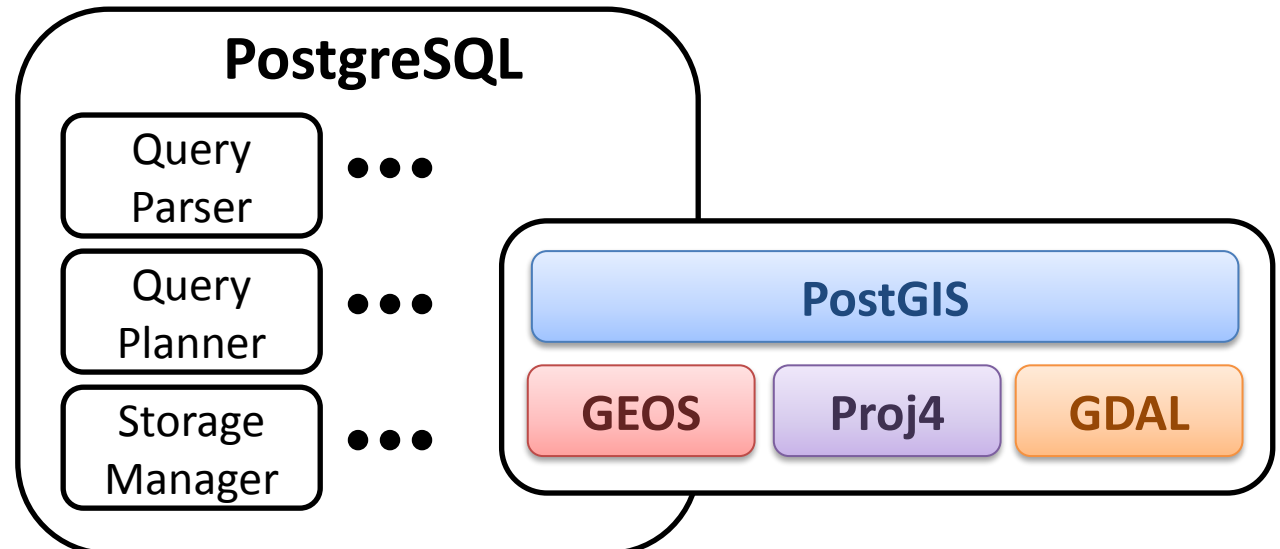
```
CREATE OR REPLACE FUNCTION _ST_Touches(geom1 geometry, geom2 geometry)
  RETURNS boolean
  AS '$libdir/postgis-2.1','touches'
  LANGUAGE 'c' IMMUTABLE STRICT
  COST 100;
```

...

PostGIS

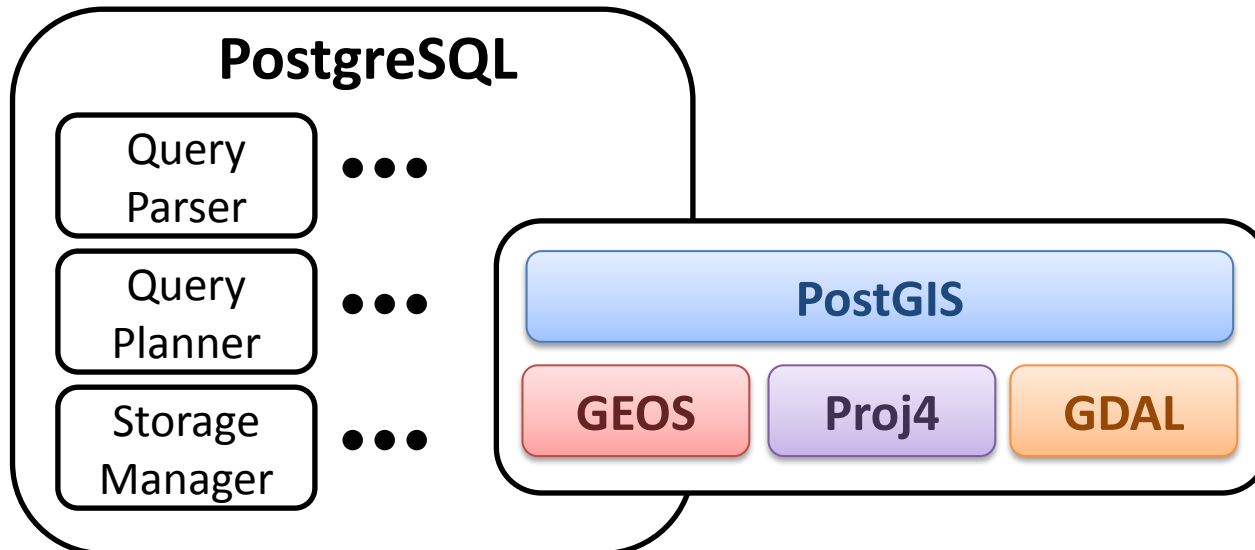


- Projeto de software livre (GPLv2) desenvolvido inicialmente pela empresa Canadense Refrations Research: <http://postgis.refrations.net>
- Extensão geográfica para o SGBD-OR PostgreSQL:
 - Inicialmente:
 - Tipos geométricos e operadores espaciais OGC SFS
 - Índice espacial: árvore-R sobre GiST.
 - Atualmente:
 - Tipos circulares e compostos, 3D, Tipo geográfico, Raster, Topologia, Redes, Geocodificação de endereços



PostGIS

- Grande evolução nos últimos anos:
 - Tipos circulares e compostos, 3D
 - Tipo geográfico
 - Raster
 - Topologia
 - Redes
 - Geocodificação de endereços



Evolução das Tecnologias de Bancos Dados

PostgreSQL → PostGIS

MySQL → Spatial and Geodetic
Geography Types

SQLite → SpatiaLite and RasterLite

Oracle → Oracle Spatial, GeoRaster,
Topology and Network
Models

IBM DB2 → Spatial Extender

SQL Server (2008) → Spatial Types

Geoespacial

BD Objeto Relacional

Bancos Dados Orientado Objeto

Bancos Dados Relacionais (SGBD-R)

Modelo Bancos Dados Rede

Modelo Bancos Dados Hierárquico

1960

1970

1980

1990

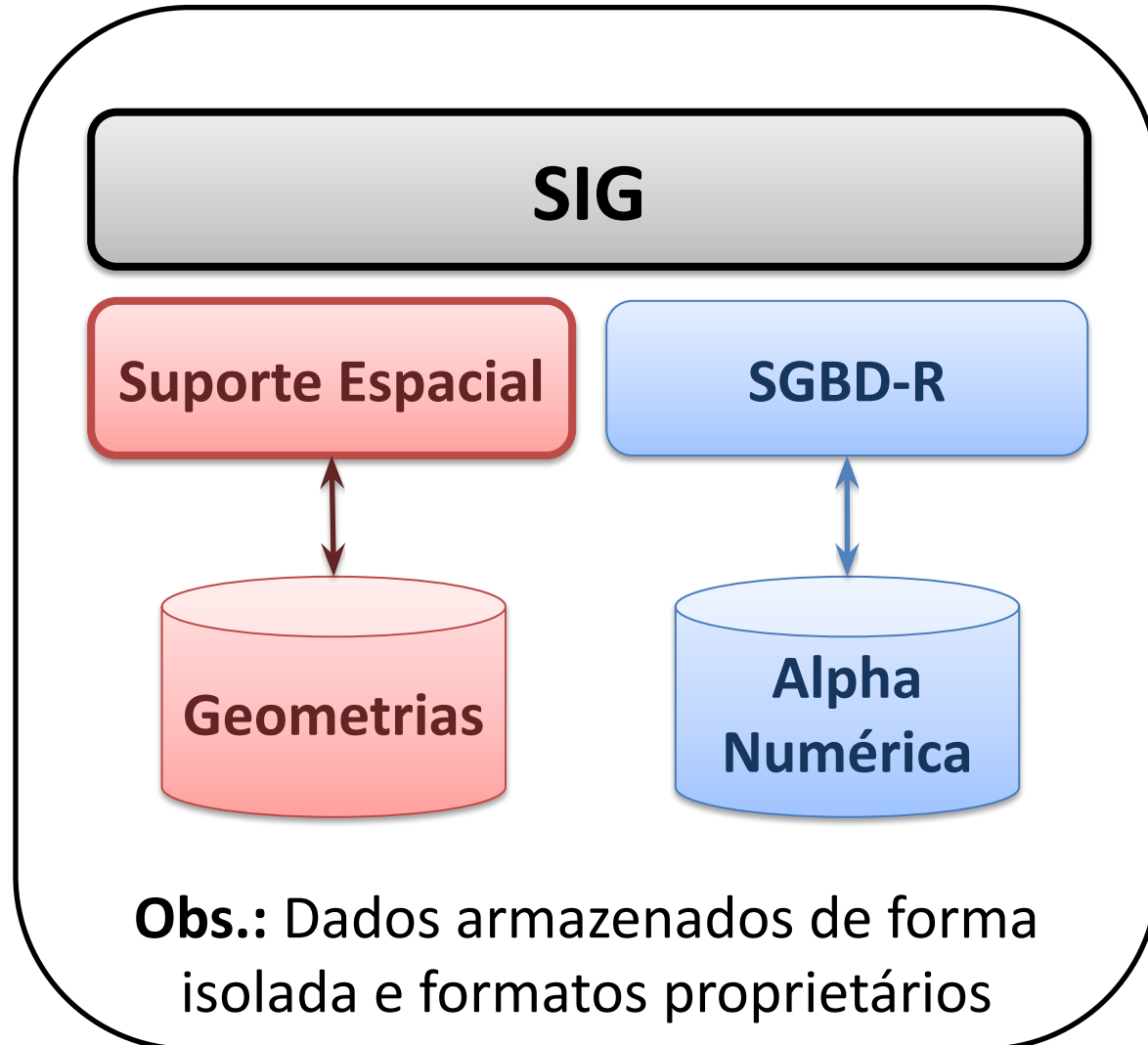
2000

2010

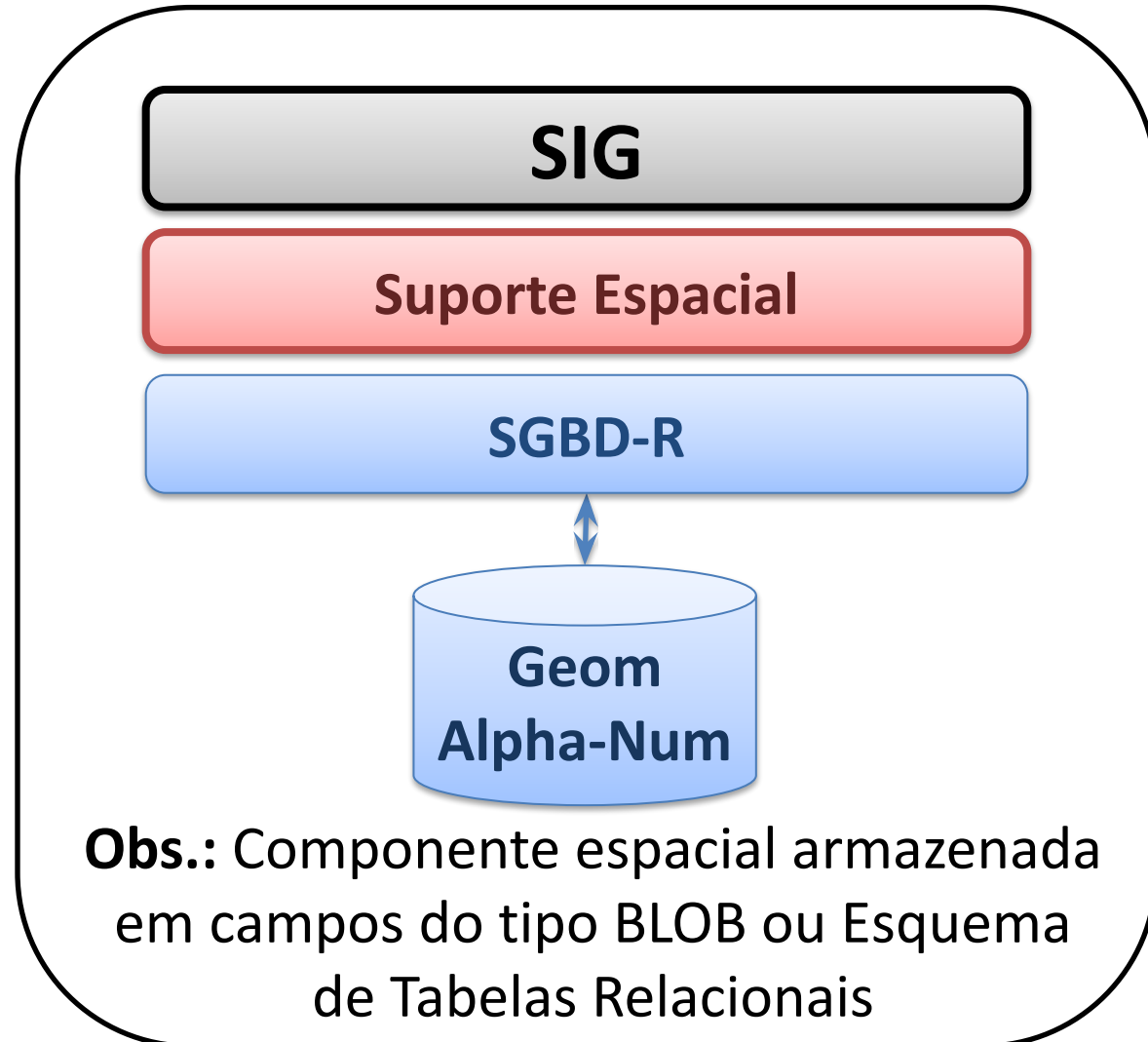
2020

Antes dos anos 2000, como era a
integração SIG e SGBD?

Arquitetura Dual





Arquitetura em Camadas



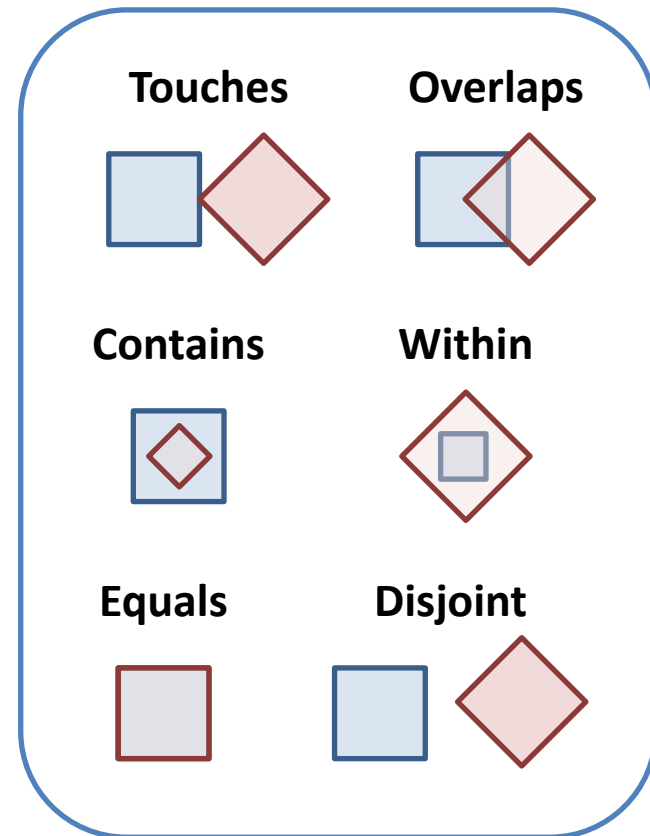
Com a integração do suporte espacial nos SGBD-R como passou a ser a integração SIG \leftrightarrow SGBD-OR?

Arquitetura Integrada: Tipos de Dados Geoespaciais

Tabela com Feições

unidades_federativas				
ufid	nome	populacao	e_vida	fronteira
31	Minas Gerais	20.997.560	77	
35	São Paulo	44.396.484	77,8	
...

Operações Espaciais



Obs.: Padrões OGC Simple Feature e ISO/SQL-MM Spatial

Sistemas de Bancos de Dados Espaciais

Um Sistema de Bancos de Dados Espacial é...

**Sistema
Bancos Dados**

as informações geométricas podem estar conectadas a dados não-espaciais:
“o proprietário de um dado lote”

+

**Tipos de Dados
Espaciais**

O Modelo de dados e a linguagem de consulta devem suportar tipos espaciais tais como pontos, linhas, ou polígonos e suas operações (interseção, área, ...)

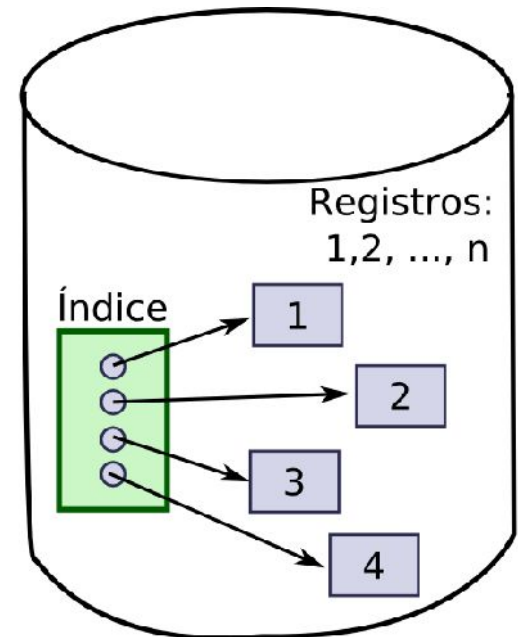
+

Indexação Espacial

deve-se evitar varrer todo o conjunto de objetos quando se executa uma consulta ou quando se realiza uma junção espacial

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

R-tree: Visão Geral

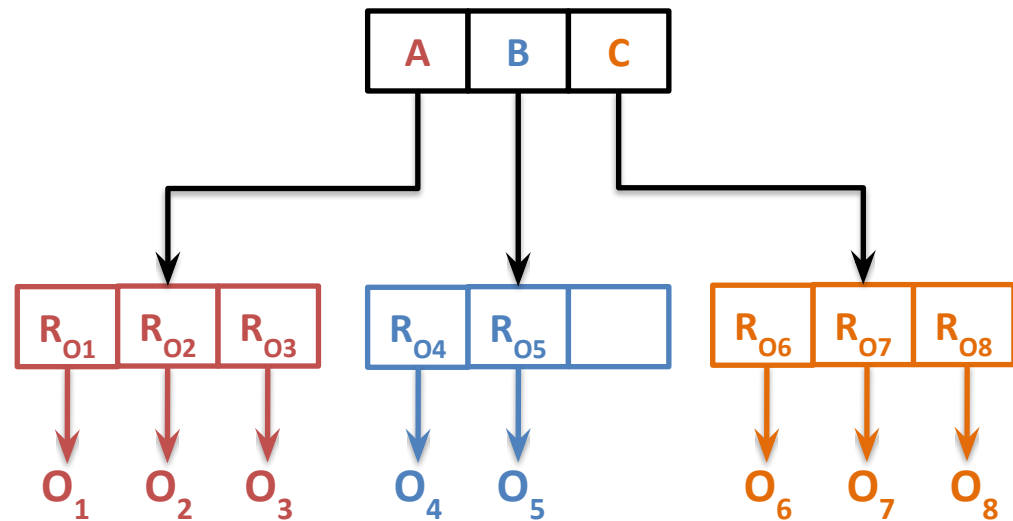
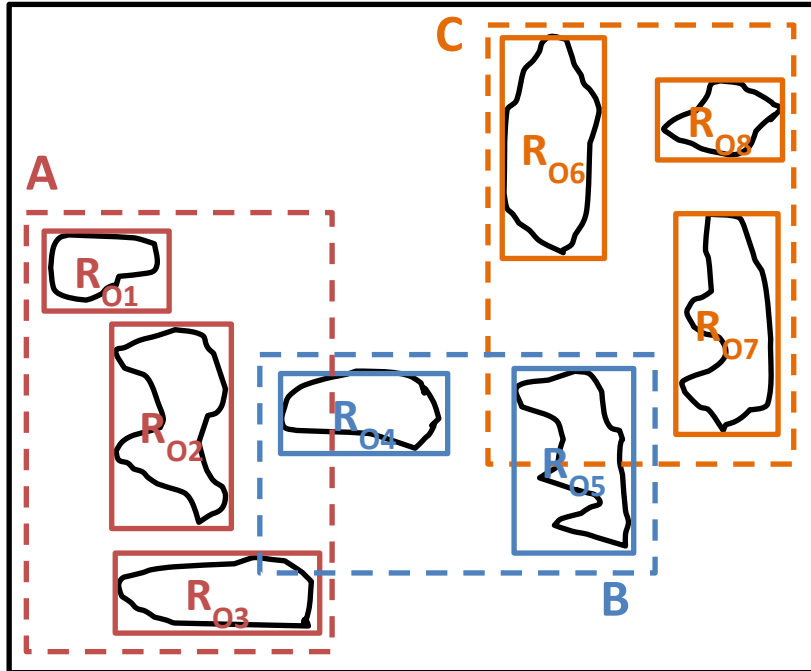
- Intervalos são organizados em uma estrutura de árvore:

- Leaf nodes: $[(I, \textit{tuple-id})]$

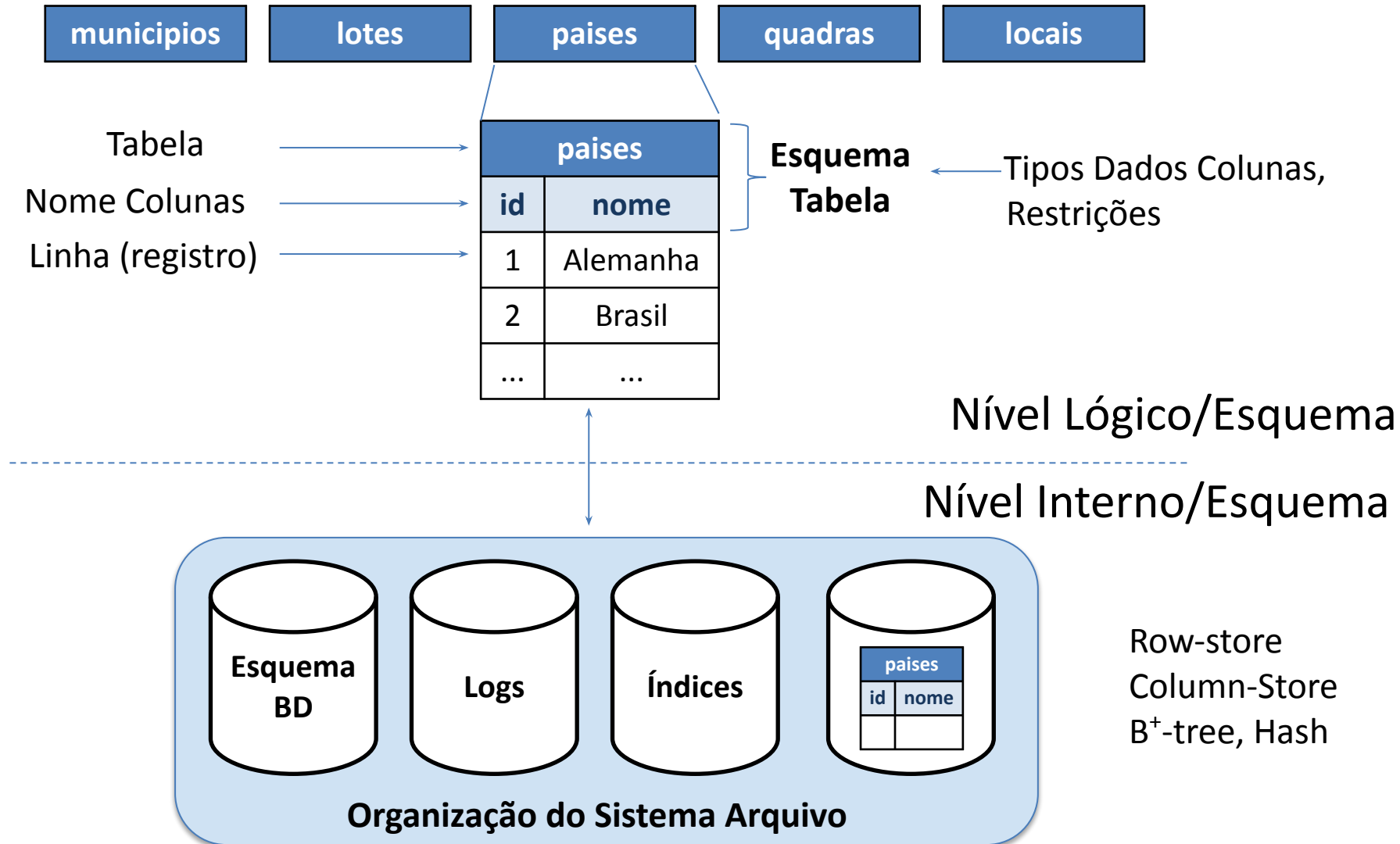
$$I = (I_0, I_1, \dots, I_{k-1})$$

- Child nodes: $[(I, \textit{child-ptr})]$

onde $I_i = [a, b]$



Independência Física dos Dados



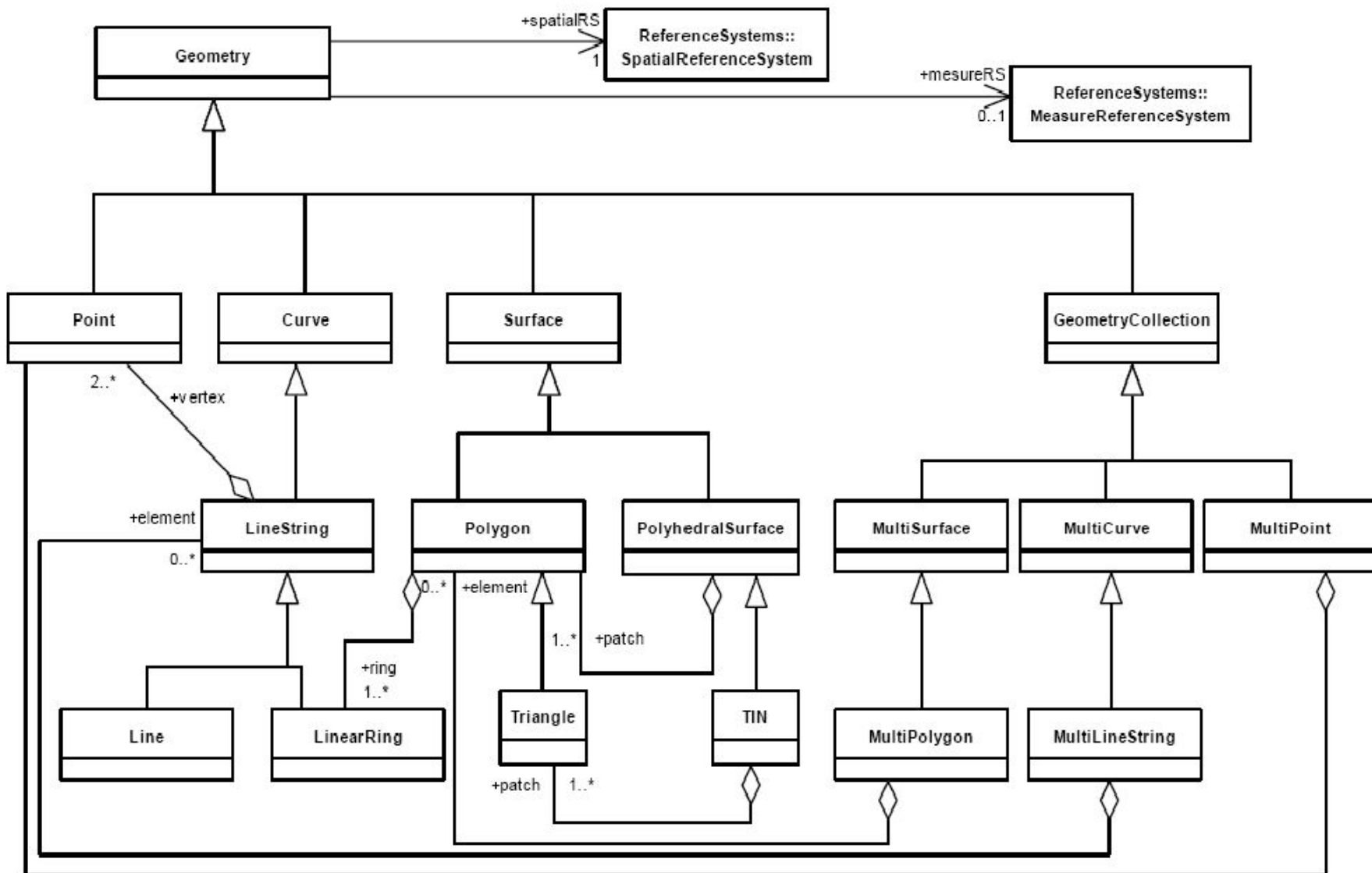
Tipos de Dados Espaciais

OGC Simple Feature

O que é a especificação *Simple Feature* (SFS)?

- Especificação criada pelo consórcio OGC que trata das questões de representação da componente espacial vetorial de dados geográficos:
 - Basicamente, os aspectos relativos à representação de pontos, linhas e polígonos.
- A SFS é dividida em duas partes:
 - OpenGIS Implementation Specification for Geographic information - Simple feature access - Part 1: Common architecture.
 - OpenGIS Implementation Specification for Geographic information - Simple feature access - Part 2: SQL option.

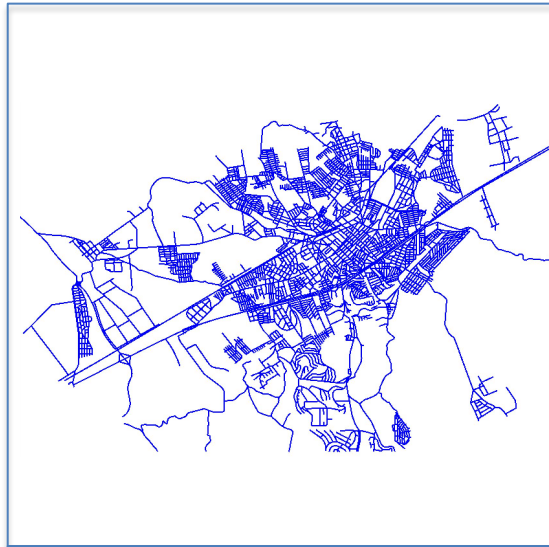
OGC SFS: Modelo Geométrico



Fonte: OGC (2012a)



Hidrelétricas e
Termelétricas



Trechos Rua



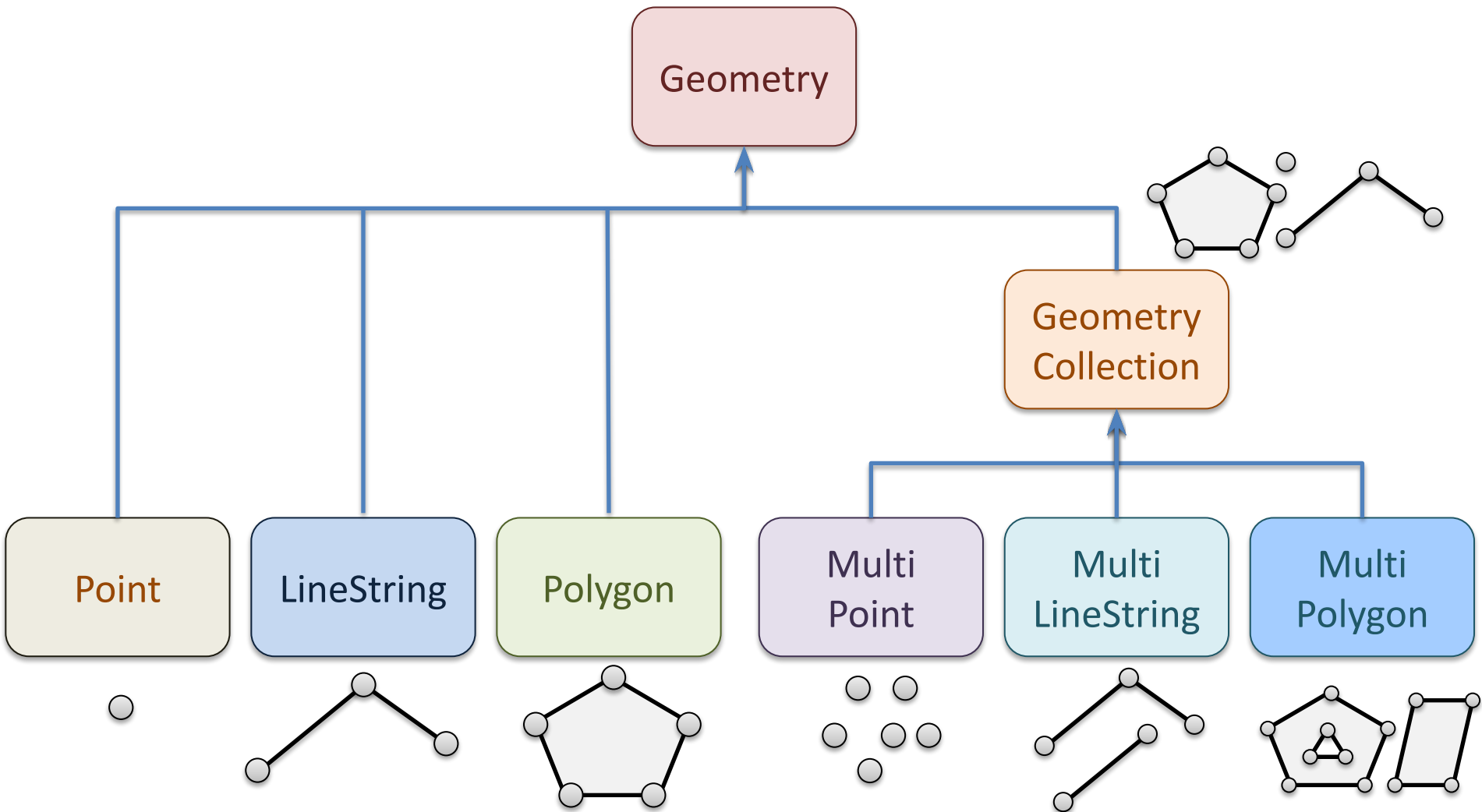
Municípios

Modelos Baseados em Entidades:

- ✓ Objeto 0-dimensional: Ponto.
- ✓ Objeto 1-dimensional: Linha Poligonal (linha).
- ✓ Objeto 2-dimensional ou Superfície: Polígonos.

Tipos Geométricos

(Modelo Simplificado OGC SFS)



Tipo Geométrico: Ponto

- Objeto 0-dimensional:
 - não possui comprimento, área ou volume.
- Pontos podem ser representados no R^n :
 - R^2 : `Point(x y)` ou `Point(x y m)`
 - R^3 : `Point(x y z)` ou `Point(x y z m)`

Tipo Geométrico: Ponto

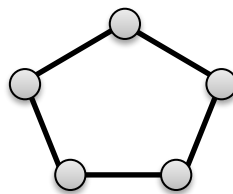
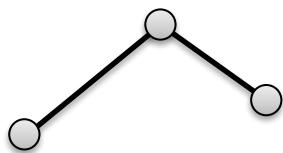
- Em geral, os SBDE utilizam números em ponto flutuante para representação das coordenadas de um ponto.
- Ex:

```
typedef struct  
{  
    double x;  
    double y;  
} point;
```




Tipo Geométrico: Linha Poligonal

- Linha Poligonal ou Linha ou LineString.
- Objeto 1-dimensional:
 - possui comprimento.
 - não possui área ou volume.
- Distinção:
 - Linha Aberta x Linha Fechada (ou Anel)





Tipo Geométrico: Linha Poligonal

- Linhas podem ser representados no R^n :

R^2 : `LineString(x1 y1, ..., xn yn)`

R^3 : `LineString(x1 y1 z1, ..., xn yn zn)`

ou `LineString(x1 y1 z1 m1, ..., xn yn zn mn)`

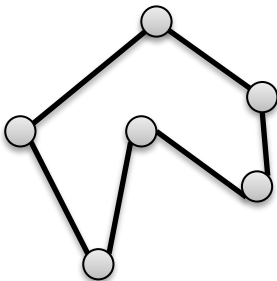
- Em geral, os SBDE utilizam números em ponto flutuante para representação das coordenadas dos vértices de uma linha. Ex:

```
typedef struct
{
    int npts;
    coord2d* coords;
} linestring;
```

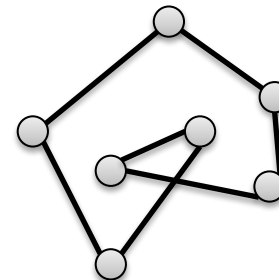


Tipo Geométrico: Polígono

- Polígono
- Objeto 2-dimensional:
 - possui área.
 - não possui volume.
- Polígono Simples:



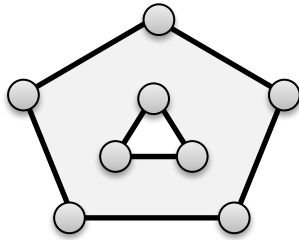
Polígono não-simples:



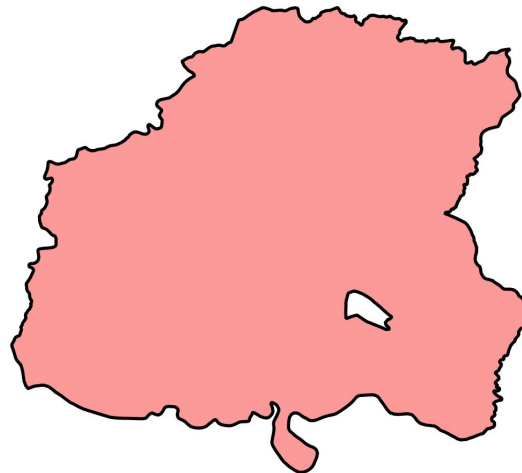


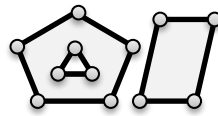
Tipo Geométrico: Polígono

- Polígono com buraco:
 - Exterior é desconectado.



Exemplo: São Pedro (SP)

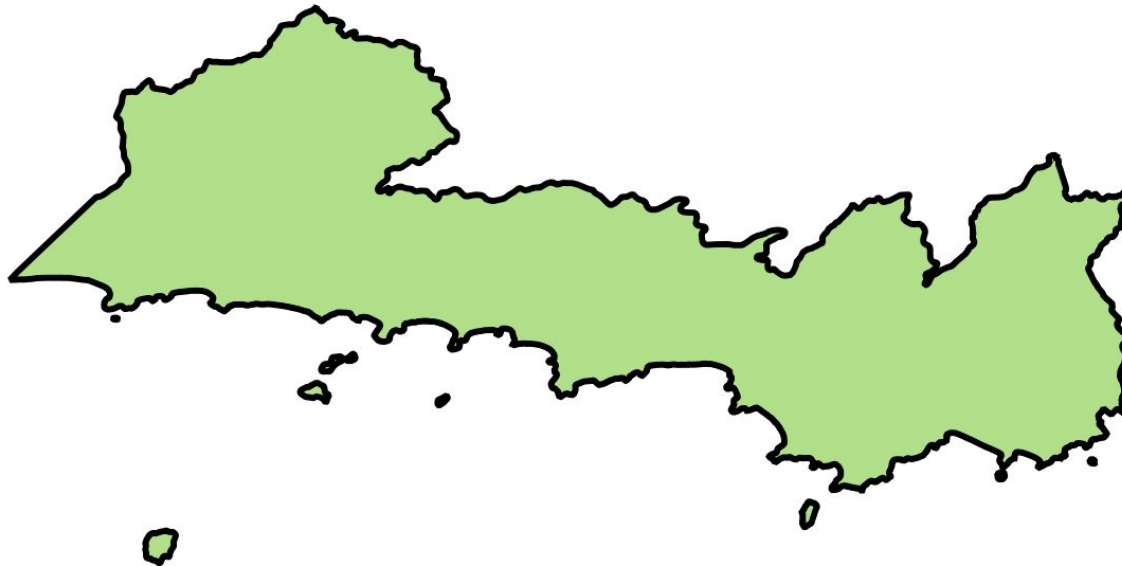




Tipo Geométrico: Conjunto de Polígonos

- Conjunto de Polígonos ou MultiPolygon.
- Representação de objetos com componentes não conectados:

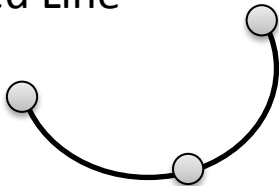
Exemplo: São Sebastião (SP)



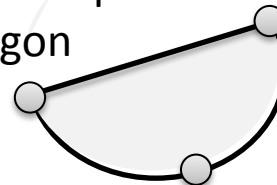
Tipos Geométricos

- Sobre a escolha do tipo geométrico na representação de um objeto geográfico:
 - Arbitrária.
 - Geralmente é dependente do uso e da escala.
- Os objetos espaciais LineString e Polygon pressupõe a representação através de segmentos de reta, isto é, uma interpolação linear entre os vértices.
- Os SBDE podem fornecer tipos geométricos mais complexos:
 - Tipos Curvos (arcos),
 - Geometrias Compostas de segmentos e arcos.

Curved Line




Curved and Compound Polygon




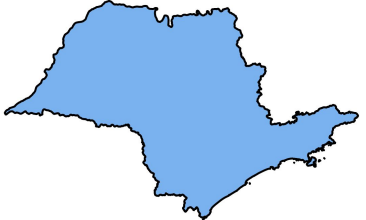
Representação de Coleções de Objetos Geográficos

Obs.: Objetos Geográficos também
são chamados de *Feature* (Feições).

Feição (Objeto Geográfico)

unidade_federativa				
ufid	nome	populacao	e_vida	fronteira
31	Minas Gerais	20.997.560	77	

Coleção de Feições: Modelo OGC *Simple Feature*

unidades_federativas				
ufid	nome	populacao	e_vida	fronteira
31	Minas Gerais	20.997.560	77	
35	São Paulo	44.396.484	77,8	
...

Coleção de Feições: Modelo OGC *Simple Feature*

- A geometria de cada objeto geográfico é descrita de forma independente dos demais objetos da coleção.
- Não há informação de topologia armazenada. Qualquer relacionamento espacial precisa ser computado *on-the-fly*.
- Logo teremos uma certa redundância na representação das fronteiras de objetos adjacentes.
- Vantagem: Simplicidade!

Coleção de Feições: Modelo Topológico

$$A_1 = \{L_1, L_2, L_3\}$$

$$A_2 = \{L_3, L_4, L_5\}$$

$$A_3 = \{L_2, L_5, L_6\}$$

$$L_1 = \{A_1\}$$

$$L_2 = \{A_1, A_3\}$$

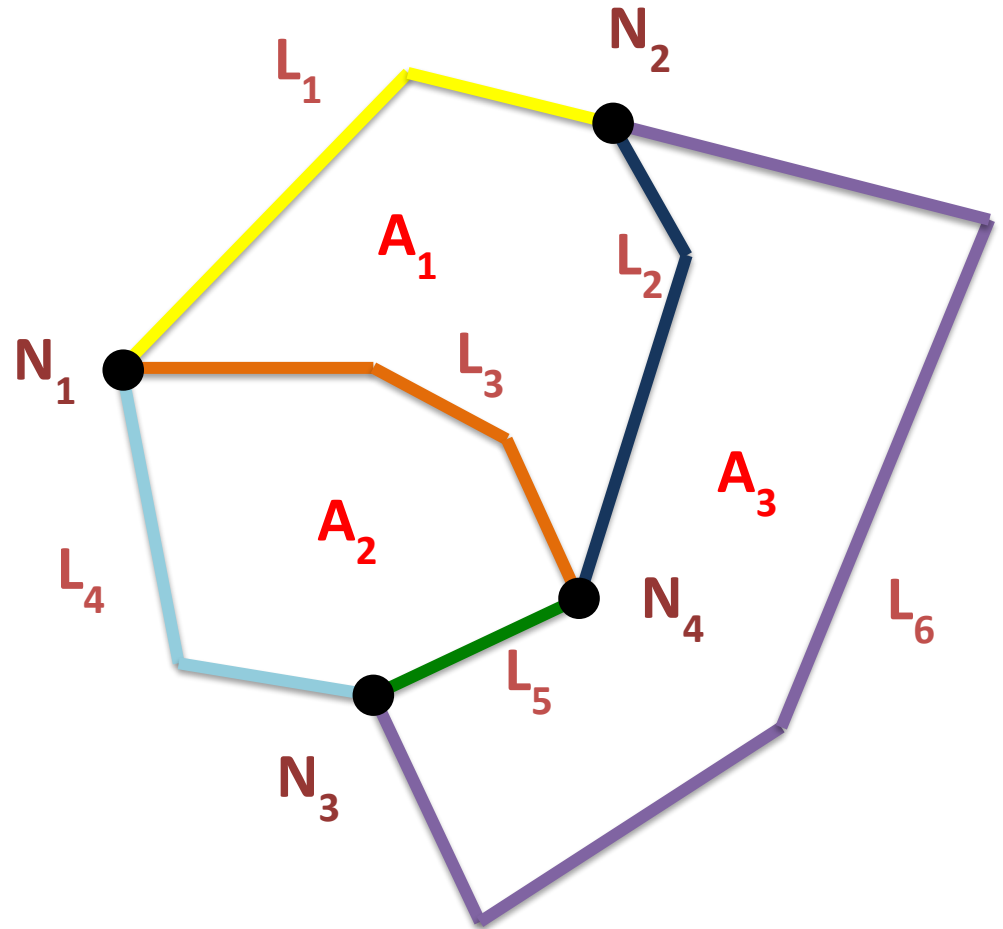
$$L_3 = \{A_1, A_2\}$$

...

$$N_1 = \{L_1, L_3, L_4\}$$

$$N_2 = \{L_1, L_6, L_2\}$$

...



Coleção de Feições: Modelo Topológico

- Maior velocidade para responder consultas topológicas.
 - Ex: Quais os polígonos adjacentes ao polígono A_1 ?
- Maior facilidade para manter a consistência espacial uma vez que as fronteiras são compartilhadas.
- Visualização é mais lenta!

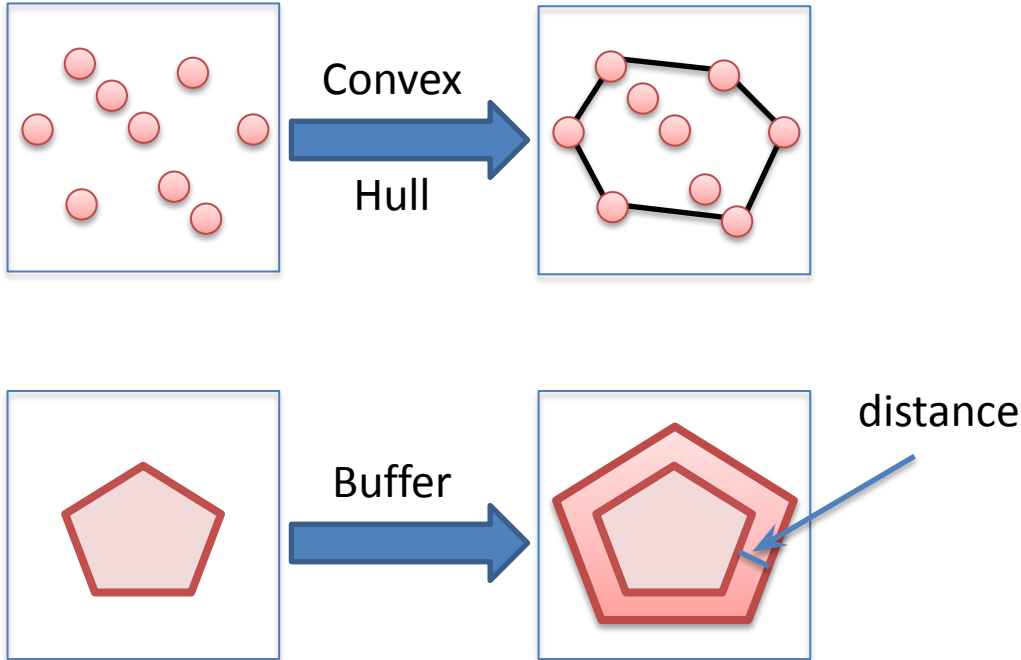
Operações Espaciais

Operações Geométricas

Operações Métricas

Relacionamentos Espaciais

Operadores Geométricos



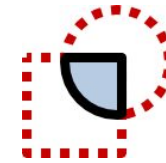
Topological Transforms: rotation, translation, scale change, symmetry.

Dimensional Transforms: boundary.

Extraction: MBR, centroid.

Overlay/Set operations

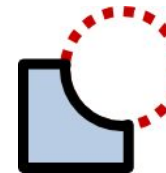
Intersection



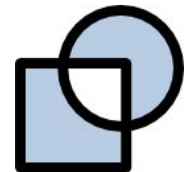
Union



Difference



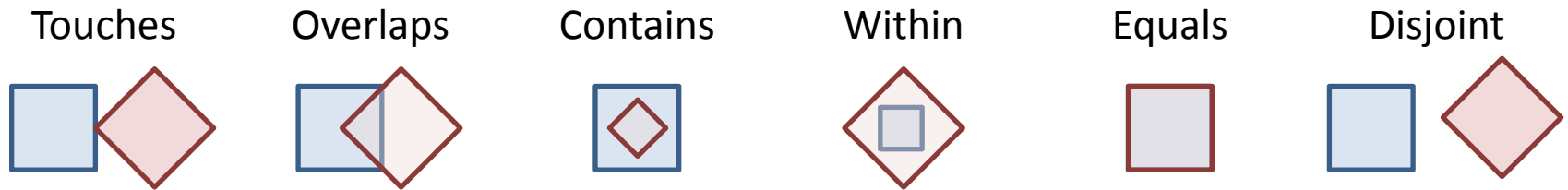
Symetric Difference



Object Properties: is_convex, is_connected, is_simple.

Relacionamentos Espaciais (Spatial Relationships)

- Topological relationships:



- Direction relationships:

- Above, below, north_of, ...

- Metric relationships:

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

Relacionamentos Espaciais (Spatial Relationships)

Fonte: 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

Operadores Topológicos

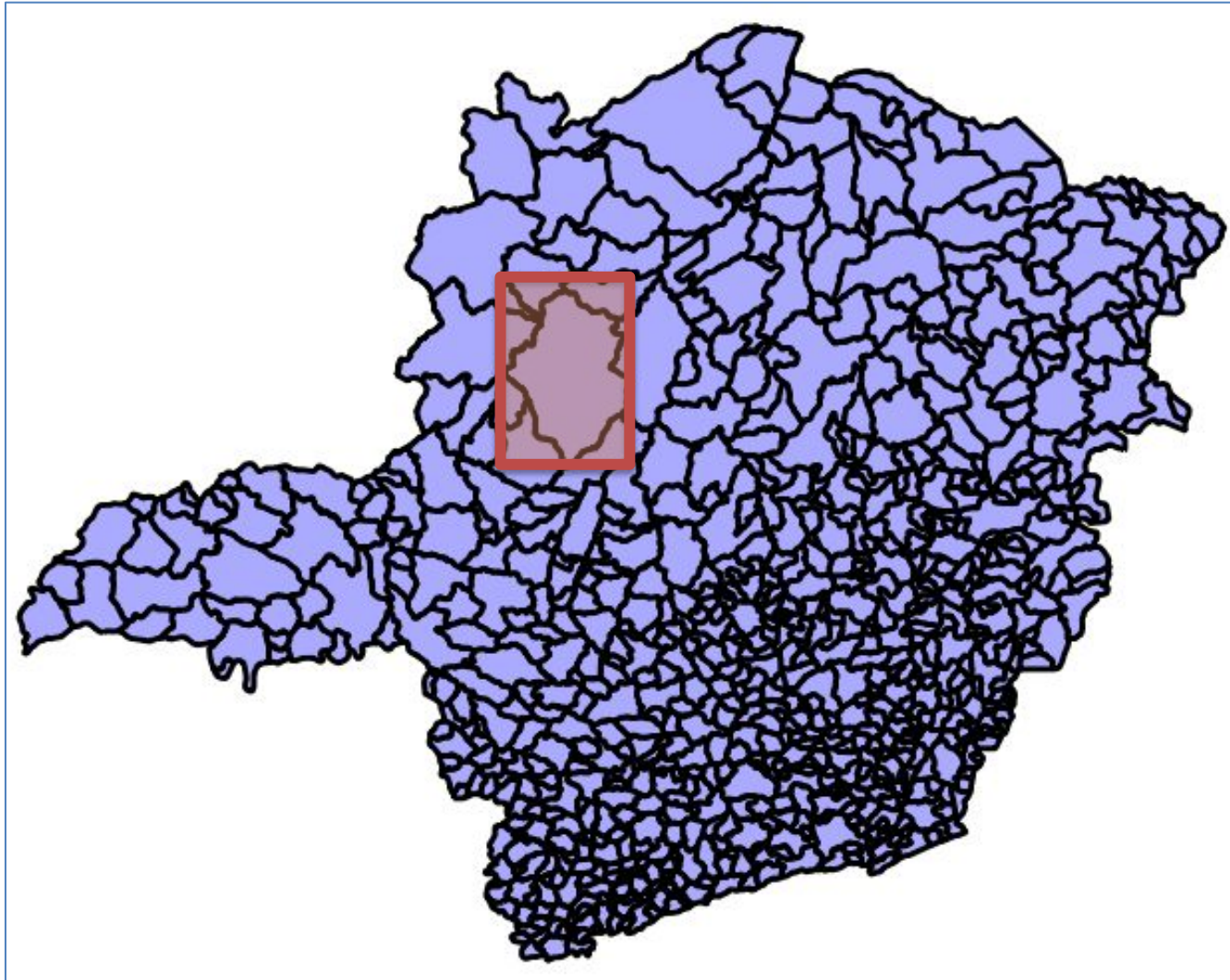
- `ST_Contains(geom1, geom2)`
- `ST_Within(geom1, geom2)`
- `ST_Covers (geom1, geom2)`
- `ST_CoveredBy(geom1, geom2)`
- `ST_Touches(geom1, geom2)`
- `ST_Crosses(geom1, geom2)`
- `ST_Overlaps(geom1, geom2)`
- `ST_Equals(geom1, geom2)`
- `ST_Intersects(geom1, geom2)`
- `ST_Disjoint(geom1, geom2)`
- `ST_Relate(geom1, geom2)`

Operações Típicas de um GIS

Operações Típicas em um GIS

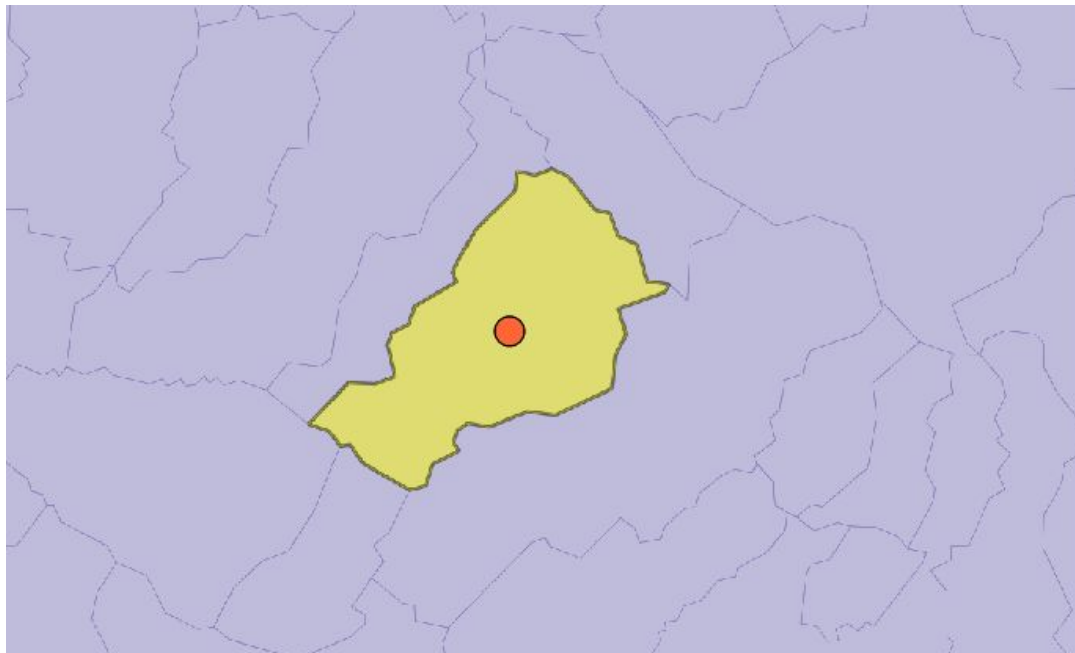
- Existe um conjunto de operações comuns de um GIS que devem ser suportadas por um SGBDE:
 - Projeção
 - Seleção
 - União
 - Overlay
 - Seleção Espacial
 - Merger

Consulta de Janela (ou Box)



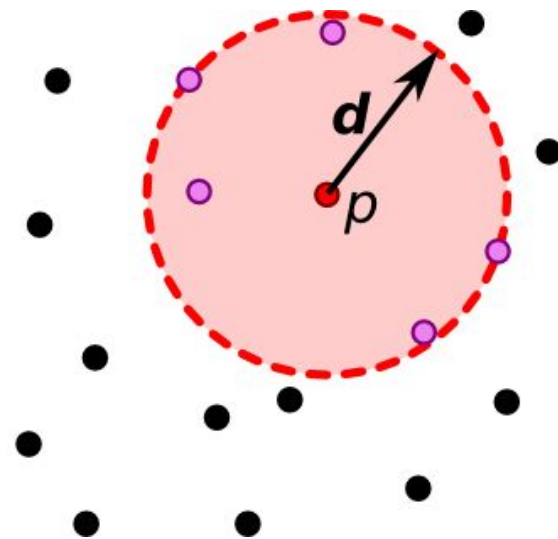
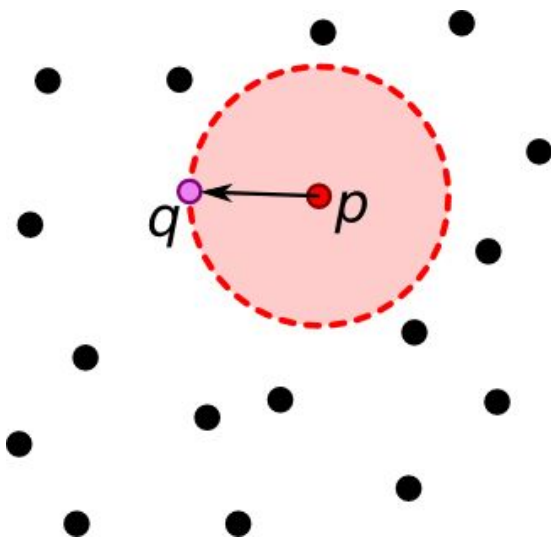
Consultas Espaciais: Apontamento

- Caso especial da consulta por intervalo, também conhecida como “point query” ou “stabbing query”:
 - O retângulo de consulta é degenerado a um ponto
 - **Definição:** dado um ponto, localizar os objetos que contém este ponto;
- Exemplo:



Consultas Espaciais: Vizinhaça

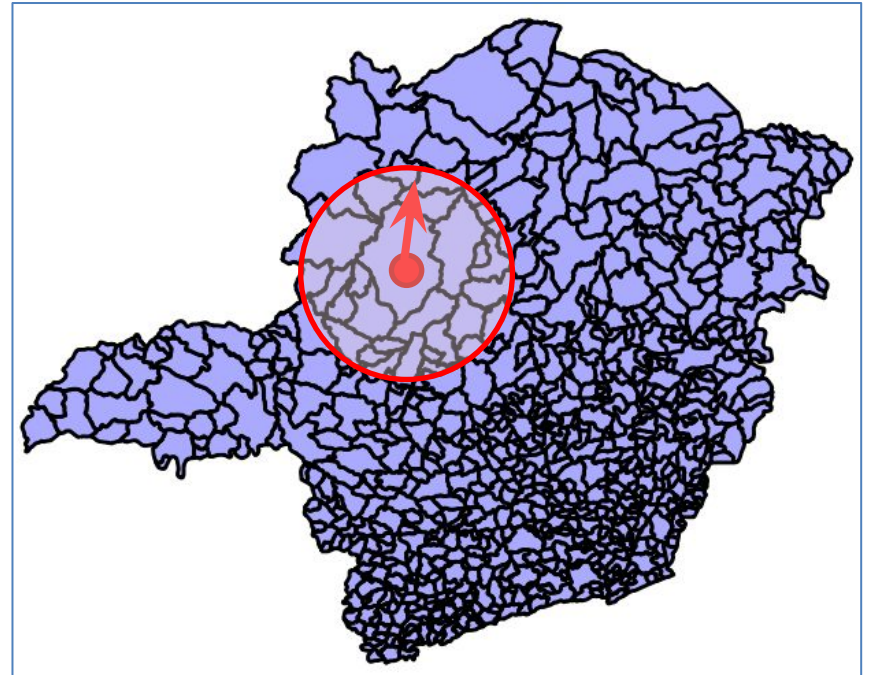
- Vizinho mais próximo:
Definição: Localizar o(s) objeto(s) q mais próximo(s) de um dado objeto p
- Vizinho(s) mais próximo(s) a uma certa distância:
Definição: Localizar o(s) objeto(s) q mais próximo(s) de um dado objeto p , a uma distância máxima de d unidades



Recuperando objetos a uma certa distância

- Fazer uma consulta considerando:
 - Ponto: -45.9 -17.5
 - Raio: 1.0
 - Operador: `ST_DWithin(geom1, geom2, dist)`

```
SELECT nome_muni,  
       ST_AsText(geom) AS geom  
FROM mg_municipios  
WHERE ST_DWithin(geom,  
ST_GeomFromText(  
    'POINT(-45.9 -17.5)', 4618),  
1.0);
```



Considerações Finais

Considerações Finais

- O dado geoespacial é de suma importância:
 - Gestão de recursos naturais.
 - Monitoramento ambiental.
 - Controle do território.
 - Planejamento e gestão urbana.
 - Construção de aplicativos que exploram o espaço geográfico.
- ...

Considerações Finais

- Atualmente, quase todos os SGBD-OR possuem suporte espacial:
 - Os padrões OGC Simple Feature e SQL ISO/MM Spatial são responsáveis pela interoperabilidade entre este suporte.
 - No entanto, para o dado matricial ainda não há um padrão estabelecido, o que pode ser percebido pela diferença em cada implementação.
- Os sistemas não-relacionais (ou NoSQL) também começam a incluir algum tipo de suporte dada a importância desse tipo de dado.

Considerações Finais

- SQL Espacial é uma forma fácil e poderosa de realizar consultas em bancos de dados.
- Os usuários de GIS devem considerar aprender SQL pois muita coisa que não se encontra no *toolbox* de um GIS pode ser realizado através desta linguagem e sua curva de aprendizado é pequena, principalmente, quando comparada a linguagens **não declarativas** como C e C++.

Considerações Finais

- A importância do componente **tempo** em GIS tem sido um tópico de pesquisa de grande interesse nas últimas décadas.

[Postgis-T](#)

Referências Bibliográficas

Artigos

- E. F. Codd. 1970. ***A relational model of data for large shared data banks.*** *Communications of the ACM*, v. 13, n. 6, June 1970, pp. 377-387.
- Chen, P. ***The Entity-Relationship Model-Toward a Unified View of Data.*** *ACM Transactions on Database Systems*, v. 1, n. 1, 1976, pp. 9-36.
- GRAY, J. ***Evolution of Data Management.*** *IEEE Computer*, v. 29, n. 10, 1996, pp. 38-46.

Artigos

- R. H. Güting. ***Gral: an extensible relational database system for geometric applications***. In Proceedings of the 15th international conference on Very large data bases (VLDB '89). Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1989, pp. 33-44.
- Vijlbrief, T., and P. van Oosterom. ***The GEO++ System: An Extensible GIS***. Proc. 5th Intl. Symposium on Spatial Data Handling, Charleston, South Carolina, 1992, pp. 40-50.
- Dangermond, J. ***A Classification of Software Components Commonly Used in Geographic Information Systems***. In Proceedings of the U.S.-Australia Workshop on the Design and Implementation of Computer-Based Geographic Information Systems, Honolulu, HI, 1982, pp. 70–91.

Artigos

- Eliseo Clementini, Paolino Di Felice, and Peter van Oosterom. **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, 1993, pp. 277-295.
- Eliseo Clementini and Paolino Di Felice. **A comparison of methods for representing topological relationships.** Inf. Sci. Appl. v. 3, n. 3 (May 1995), 1995, pp. 149-178.

Especificações e Padrões

- OGC. ***OpenGIS Implementation Specification for Geographic information - Simple feature access - Part 1: Common architecture***. Disponível em: <http://www.opengeospatial.org>. Acesso: Outubro de 2012.
- OGC. ***OpenGIS Implementation Specification for Geographic information - Simple feature access - Part 2: SQL option***. Disponível em: <http://www.opengeospatial.org>. Acesso: Outubro de 2012.
- <http://www.opengeospatial.org/standards/sfs>. 01 de Julho de 2016
- ISO. ***SQL Multimedia and Application Packages – Part 3: Spatial***.

Obrigado!