



embarcadero conference

Georreferenciamento com Delphi

Carlos Henrique Cantu





GIS ?



- GIS - Geographic Information System (Sistema de Informação Geográfica)
- Geolocalização no Firebird
 - O Firebird não possui recursos/funções nativas para suporte a Geolocalização
 - Isso não significa que não possamos implementar nossas próprias soluções 😊



Geolocalização nessa palestra

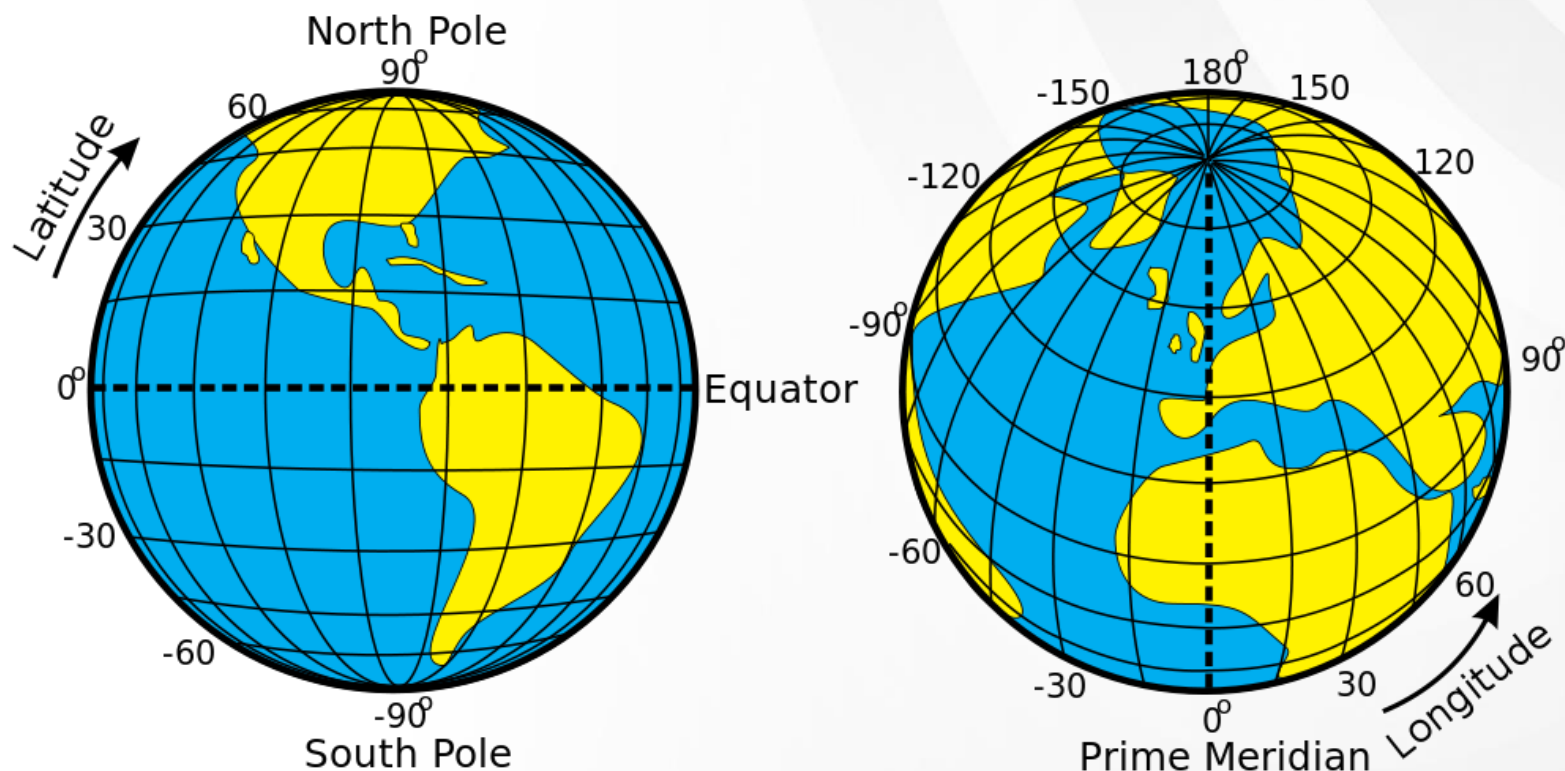


- **Como georreferenciar uma base de dados já contendo endereços cadastrados?**
- **Construir uma consulta (select) que retornará os clientes localizados em um determinado raio, a partir de um ponto central do globo terrestre (latitude/longitude).**



Latitude / Longitude

Sistema de referenciamento usado para localizar um ponto na superfície terrestre.





Notação



- **Graus, minutos e segundos**
 - Ex: 38° 53' 23" **N**, 77° 00' 32" **W**

- **Graus decimais**

- Ex: 38.8897, -77.0089
- Latitudes positivas significam que o ponto está ao norte do equador e negativas ao sul.
- Longitudes positivas significam que o ponto está a leste do meridiano primário e negativas ao oeste.
- Todos os pontos no Brasil estarão tanto com latitude como longitudes negativas.

Mais informações: en.wikipedia.org/wiki/Decimal_degrees



Latitude / Longitude





Latitude/Longitude no BD



- Na prática, uma **boa precisão** (para aplicações comerciais) necessita de **6 casas decimais** (precisão de ~1m, no equador)
- Qual o **melhor tipo de dado no Firebird**, para armazenar latitude/longitude?



Ponto Flutuante no Firebird



- **FLOAT**

- 32 bits: 1 para o sinal, 8 para o expoente e 23 para a mantissa.
- **7 dígitos de precisão**
- Entre 3.4×10^{-38} e 3.4×10^{38}

- **DOUBLE PRECISION**

- 64 bits: 1 para o sinal, 11 para o expoente e 52 para a mantissa
- **15 dígitos de precisão**
- Entre 1.7×10^{-308} e 1.7×10^{308}



Será que algum deles serve para armazenarmos latitudes e longitudes?



Double Precision ?



- Garante 15 dígitos de precisão
- Maior grau = 180 (3 dígitos)
- Casas decimais necessárias para precisão "ok" = 6
- Total de dígitos necessários = $3 + 6 = 9$

Double Precision





Float ?



- Float não é indicado, pois com 2 ou mais dígitos no “grau”, perde-se precisão nas casas decimais.

- Ex:

1.234567890 (6 casas decimais – ok) 

12.34567890 (5 casas decimais – ruim) 

123.4567890 (4 casas decimais – ruim) 

↓
Precisão garantida
de 7 dígitos



Ponto fixo



- NUMERIC (p,s) / DECIMAL (p,s)
- Pode ser armazenado ocupando 16, 32 ou 64 bits
- ***p*** = *precision (total de dígitos)* [$1 \leq p \leq 18$]
s = *scale (número de dígitos após a vírgula)*
- ***s*** deve ser sempre **menor ou igual a p**
- Se não informado *p* e *s*, tipo interno será INTEGER
- Firebird não segue o standard, e tanto *decimal* quanto *numeric* permitem armazenar um total de dígitos maior que o declarado.
- **Valor recuperado é exatamente igual ao armazenado!**



Armazenamento interno de NUMERIC e DECIMAL

PRECISION (p)	TIPO	DIALETO 3	DIALETO 1
1..4	NUMERIC	SMALLINT (*) [16 bits]	SMALLINT [16 bits]
1..4	DECIMAL	INTEGER (*) [32 bits]	INTEGER [32 bits]
5..9	NUMERIC e DECIMAL	INTEGER [32 bits]	INTEGER [32 bits]
10..18	NUMERIC e DECIMAL	BIGINT [64 bits]	DOUBLE PRECISION(!) [64 bits]

No Firebird, DECIMAL e NUMERICs são idênticos, salvo quando $p \leq 4$.

(*) Nesse caso, a faixa de valores suportados será diferente entre NUMERIC e DECIMAL



Limites numeric/decimals



Como determinar o limite de armazenamento?

1. Determinar o tipo de dado usado internamente de acordo com a precisão (p) do campo.
2. Determinar a faixa de valores suportados pelo tipo interno (smallint/integer/bigint).
3. Dividir o menor e o maior valor da faixa por **10^s** para obter a faixa de valores aceitos.



Limites *numeric/decimals*



Exemplo:

1. **NUMERIC (9,2) ou DECIMAL (9,2)**
2. **Armazenados internamente como INTEGER**
3. **Integer = -2.147.483.648 à 2.147.483.647**
4. **Como $s = 2$, dividir por 10^2**
5. **Faixa de um NUMERIC/DECIMAL (9,2) =
-21.474.836,48 à 21.474.836,47**



“Calculadora de ranges”



www.firebase.com.br/artigo.php?id=2777



Qual o melhor tipo para lat/long?

- **DOUBLE PRECISION** ✓

Tamanho interno: INTEGER (**64 bits**)

Valores recuperados podem conter “dígitos extras”

- **NUMERIC/DECIMAL (7,6)** ✓

Tamanho interno: INTEGER (**32 bits**)

Range de valores: -2147,483648 a 2147,483647

Valor recuperado é exatamente o que foi armazenado



Geocodificando endereços



- **BD contendo tabela de clientes**
- **Campos para armazenar lat/long**
- **Usar algum serviço de geocodificação, passando o endereço e obtendo as coordenadas (*Google, Bing, MapQuest, etc*)**
- **API GoogleMaps**
 - **Limite de solicitações por dia**
 - **2.500 chamadas por dia, 50 por segundo (se usar uma chave)**
 - **USD 0,50 por cada 1.000 solicitações extras**

<https://developers.google.com/maps/documentation/geocoding/usage-limits?hl=pt-br>



GoogleMaps Geocoding API



Limites de uso padrão



Usuários da API padrão:

- 2.500 solicitações gratuitas por dia, calculadas como a soma das consultas [do lado do cliente](#) e do lado do servidor.
- 50 solicitações por segundo, calculadas como a soma das consultas [do lado do cliente](#) e do lado do servidor.

Ative a cobrança de pagamento conforme o uso para ter acesso a cotas maiores:

US \$0,50 / 1000 solicitações adicionais, até 100.000 solicitações diárias.

[ATIVAR COBRANÇA](#)

Limites de uso premium



Clientes da [Google Maps APIs Premium Plan](#):

- [Cota diária compartilhada gratuita](#) de 100.000 solicitações a cada 24 horas. Solicitações adicionais mediante compra anual de crédito das Maps APIs.
- 50* solicitações do lado do servidor por segundo.

* Limite padrão. Entre em contato com o gerente de contas de vendas empresariais do Google se precisar de um limite mais alto. Observe que o [serviço do lado do cliente](#) oferece ilimitada solicitações por segundo por projeto.

Benefícios adicionais de um Plano premium:

- Contratos anuais com termos empresariais
- Suporte técnico 24 horas
- Acordo de nível de serviço (ANS)
- Licenças para casos de uso internos, de OEM e de rastreamento de ativos

[Departamento de vendas](#) para saber mais.



Chamando API do GoogleMaps



<http://maps.google.com/maps/api/geocode/json?address=Rua+Governador+Pedro+de+Toledo+1000+Piracicaba+SP>



Exemplo de geocodificação da tabela de clientes, usando Delphi + GoogleMaps



Curiosidades



- A Terra é uma esfera, mas não uma esfera perfeita
- Localizar pontos dentro de uma área, precisa levar em consideração a curvatura da Terra na delimitação da mesma, etc.
- 6.371.008 metros é o raio médio/aproximado da Terra





Delimitação de áreas





SQL “mágico”



-- Raio da Terra = 6371008m

```
select C.CODIGO, C.NOME
from CLIFOR C
where (acos(sin(:LAT) * sin(C.LAT * 3.14159 / 180) +
          cos(:LAT) * cos(C.LAT * 3.14159 / 180) *
          cos((C.LNG * 3.14159 / 180) - :LNG)) * 6371008) < :RAIO
```

:LAT = Latitude do ponto central

:LNG = Longitude do ponto central

:RAIO = Raio, em metros, da área a ser considerada



PROCEDURE SELECIONÁVEL



- **Recurso exclusivo do Firebird/InterBase**
- **Permite que um *select* chame uma *procedure*, e que ela retorne um *resultset*, como se fosse uma tabela.**



PROCEDURE SELECIONÁVEL



```
create or alter procedure CHC_GETCLIENTECOORD V1 (  
    LAT double precision,  
    LNG double precision,  
    RAIO_METROS float)  
  
returns (  
    CODIGO integer,  
    NOME type of column CLIFOR.NOME,  
    LOCAL varchar(150),  
    LONGITUDE D_GPS,  
    LATITUDE D_GPS)  
  
as  
  
begin  
  
    LAT = LAT * 3.14159 / 180; -- Converte graus em radianos  
    LNG = LNG * 3.14159 / 180; -- Converte graus em radianos  
  
  
    for select C.CODIGO, C.NOME, C.LAT, C.LNG,  
        (C.ENDERECO || ', ' || C.NUMERO || ' - ' ||  
        coalesce(C.BAIRRO || ' - ', '')) ||  
        C.CIDADE || '/' || ESTADO) as LOCAL  
    from CLIFOR C  
    where (acos(sin(:LAT) * sin(C.LAT * 3.14159 / 180) + cos(:LAT) *  
        cos(C.LAT * 3.14159 / 180) * cos((C.LNG * 3.14159 / 180) - :LNG)) * 6371008) < :RAIO_METROS  
    into :CODIGO, :NOME, :LATITUDE, :LONGITUDE, :LOCAL  
  
do  
    suspend;  
  
end
```



Resultados de execução



- `SELECT *`
`FROM`
`chc_getclientecoord_v1`
`(-23.574859, -46.621226, 500);`
- 16 Registros encontrados
- 5.095 leituras **não indexadas** (todos os registros da tabela foram percorridos)





Geolocalização - 14º Firebird Developers Day - Carlos Henrique Cantu (c)

Cientes Consulta

Query

```
select *  
from CHC_GETCLIENTECOORD (-23.574859,-46.621226, 500)
```

Fechar

Executar

Código	Nome	Endereço	Longitude	Latitude
285	NAR Suprimentos Médicos Ltda	R: Mesquita, 110 - Cambuci - São Paulo/SP	-46,621226	-23,574859
2318	Best Computer do Brasil Ltda - ME	Rua Teodureto Souto, 970 - Cambuci - São Paulo/SP	-46,621754	-23,573266
2660	Grafite Comunicações Ltda	Rua Mesquita, 121 - Cambuci - São Paulo/SP	-46,621002	-23,574983
2982	Raimundo Nonato das Chagas Som - ME	Rua Mesquita, 456 - Vila Deodoro - São Paulo/SP	-46,621482	-23,577992
2991	A.B. de Sousa Filho Materiais Elétricos - ME	Rua Robertson, 438 - Cambuci - São Paulo/SP	-46,622663	-23,573689
3100	Drogaria e Perfumaria Facil	Av. Lins de Vasconcelos, 932 - - São Paulo/SP	-46,622697	-23,571506
3189	Della Via Pneus Ltda	Av. Lins de Vasconcelos, 1469 - Cambuci - São Paulo/SP	-46,623445	-23,576364
3199	Danielle Anastacia Bornellis ME	AV LINS DE VASCONCELOS, 1281 - Cambuci - São Paulo/SP	-46,623068	-23,574696
3237	C&L Chaveiro Lins Ltda - ME	Rua Robertson, 384 - Cambuci - São Paulo/SP	-46,623179	-23,573614
3290	Empório Monte Verde Ltda EPP	Av. Lins de VAsconcelos, 1496 - Cambuci - São Paulo/SP	-46,623647	-23,576495
3411	Grill 1018 Ltda	Av. Lins de Vasconcelos, 1018 - Cambuci - São Paulo/SP	-46,622942	-23,572280
3463	Rhoma Lidos Ltda ME	rua robertson, 536 - São Paulo/SP	-46,621713	-23,573856
3693	Bar e Cantina Tia Rosa Ltda ME	rua Heitor peixoto, 728 - cambuci - São Paulo/SP	-46,620332	-23,575964
3751	D F Comercio de Bolos Eireli Me	Av. Lins de Vasconcelos, 1136 - Cambuci - São Paulo/SP	-46,623007	-23,573311
3797	Dia Brasil Sociedade Ltda	R Heitor peixoto, 910 - Cambuci - São Paulo/SP	-46,618391	-23,576315
3813	Panificadora e Confeitaria Karol Ltda	Rua Basilio Cunha, 556 - Cambuci - São Paulo/SP	-46,622289	-23,576930





- **Índices**
- **Estruturas baseadas em B-TREE**
- **Chaves armazenadas com compactação “simples”**
- **Chaves numéricas são armazenadas em uma representação modificada de “double precision” (menos BIGINT)**
- **Seletividade do índice – quanto mais próximo de 1, pior.**
- **Firebird tem a capacidade de utilizar vários índices e “cruzar” os resultados, recuperando apenas os registros que “se cruzam”, minimizando a necessidade de chaves compostas.**



Otimizando a pesquisa (índices)



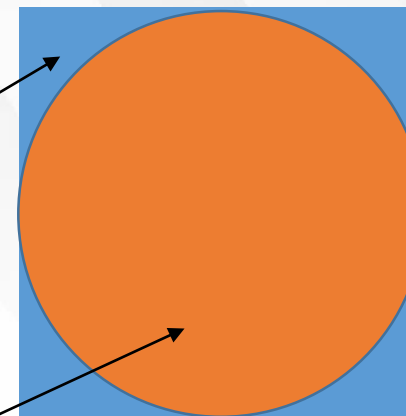
- **Estatísticas não atualizam automaticamente e ficam defasadas com o passar do tempo, sendo assim, precisam ser recalculadas:**
 - **SET STATISTICS**
 - **Desativar / Ativar o índice**
 - **Backup / Restore**
- **Uma busca indexada exige a leitura dos registros encontrados, para determinar se o mesmo “pode ser visto” pela transação ativa.**



Procedure “melhorada”



```
create or alter procedure CHC GETCLIENTECOORD (  
    LAT double precision, LNG double precision, RAOIO_METROS float)  
returns (  
    CODIGO integer, NOME type of column CLIFOR.NOME, LOCAL varchar(150), LONGITUDE D_GPS, LATITUDE D_GPS)  
as  
declare variable MAXLAT D_GPS; declare variable MAXLNG D_GPS; declare variable MINLAT D_GPS;  
declare variable MINLNG D_GPS; declare variable DELTA double precision;  
begin  
    DELTA = (RAIO_METROS / 6371008) * (180 / 3.14159);  
    MAXLAT = LAT + DELTA; MINLAT = LAT - DELTA;  
    DELTA = (RAIO_METROS / 6371008 / cos(LAT * 3.14159 / 180)) * (180 / 3.14159);  
    MAXLNG = LNG + DELTA; MINLNG = LNG - DELTA;  
  
    LAT = LAT * 3.14159 / 180; -- Converte graus em radianos  
    LNG = LNG * 3.14159 / 180; -- Converte graus em radianos  
    for select C.CODIGO, C.NOME, C.LAT, C.LNG,  
        (C.ENDERECO || ', ' || C.NUMERO || ' - ' ||  
        coalesce(C.BAIRRO || ' - ', '')) ||  
        C.CIDADE || '/' || ESTADO) as LOCAL  
    from CLIFOR C  
    where (C.LAT between :MINLAT and :MAXLAT) and  
        (C.LNG between :MINLNG and :MAXLNG) and  
        (acos(sin(:LAT) * sin(C.LAT * 3.14159 / 180) + cos(:LAT) *  
            cos(C.LAT * 3.14159 / 180) * cos((C.LNG * 3.14159 / 180) - :LNG)) * 6371008) < :RAIO_METROS  
    into :CODIGO, :NOME, :LATITUDE, :LONGITUDE, :LOCAL  
do suspend;  
end
```





Otimizando a pesquisa



- **CREATE INDEX** IDX_CLIFOR_LAT
ON CLIFOR (**LAT**);
 - PLAN (C INDEX (IDX_CLIFOR_LAT))
 - 124 leituras (indexadas)
- **CREATE INDEX** IDX_CLIFOR_LNG
ON CLIFOR (**LNG**);
 - PLAN (C INDEX (IDX_CLIFOR_LAT, IDX_CLIFOR_LNG))
 - 17 leituras (indexadas)

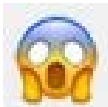


Otimizando a pesquisa



- **CREATE INDEX** IDX_CLIFOR_LATLNG
ON CLIFOR (**LAT, LNG**);
 - PLAN (C INDEX (IDX_CLIFOR_LAT, IDX_CLIFOR_LNG))
 - 17 leituras (indexadas)

- **DROP INDEX** IDX_CLIFOR_LNG;
DROP INDEX IDX_CLIFOR_LAT;
 - PLAN (C INDEX (IDX_CLIFOR_LATLNG))
 - 124 leituras (indexadas)





Já migrou para o Firebird 3?



Ainda não migrou para o Firebird 3?

Economize tempo e evite dores de cabeça no processo de migração, compre agora o *Guia de Migração para o Firebird 3*.

R\$ 50,00!



Dúvidas?

www.firebase.com.br

@chcantu

 YouTube^{BR} FirebirdDevelopersDay

