

Um Sistema para Consultar Eventos de Trânsito Históricos do Waze

Marcos Vinicius Pontarolo¹, Murilo Ferreira¹, Rebeca Schroeder², Carmem S. Hara¹

¹ Universidade Federal do Paraná, Curitiba – PR – Brazil

² Universidade do Estado de Santa Catarina, Joinville – SC, Brazil

{marcospontarolo, muriloferreira, carmemhara}@ufpr.br,
rebeca.schroeder@udesc.br

Abstract. *This paper presents MIDET-IU (Method for Indexing Traffic Events - User Interface), a system that supports queries on traffic events collected by the Waze application. The implementation of the system uses the querying mechanism and storage model from MIDET, which has been proposed in SBBD 2021. The model is based on a grid spatial partitioning over an area of interest, generating geographic cells (GCs) of fixed size. Traffic events that occur in the same GC are stored together aiming at obtaining good performance in spatial queries. In this demo, Waze data from the city of Joinville is used. A graphical interface allows the user to select a region of interest, a time interval, as well as the types of events to be presented in the query result.*

Resumo. *Este artigo apresenta o MIDET-IU (Method for Indexing Traffic Events - Interface de Usuário), um sistema que permite realizar consultas sobre eventos de trânsito coletados pelo aplicativo Waze. A implementação do sistema utiliza o modelo de armazenamento e consultas do MIDET, proposto no SBBD 2021. O modelo é baseado no particionamento espacial em forma de grade sobre uma área de interesse, gerando células geográficas (CGs) de tamanho fixo. Dessa forma, eventos de trânsito que ocorrem em uma mesma CG são armazenados em conjunto com o objetivo de obter um bom desempenho em consultas espaciais. Nesta demonstração, são utilizados dados do Waze da cidade de Joinville. Uma interface gráfica permite selecionar uma região de interesse, um intervalo de tempo, bem como os tipos de eventos a serem apresentados no resultado da consulta.*

1. Introdução

A utilização de aplicativos com mapas digitais, como o Waze e Google maps, é bastante difundida, principalmente pelos motoristas em grandes cidades. Além de auxiliar na criação de rotas, estes aplicativos coletam e apresentam informações sobre eventos de trânsito, tais como engarrafamentos e alertas. Estes eventos constituem uma fonte de informação importante para o planejamento das cidades, especialmente para a melhoria da mobilidade urbana. De acordo com o Instituto de Pesquisa Econômica Aplicada (IPEA), o modo rodoviário é responsável por 92% do transporte de passageiros do país e a taxa de motorização saltou de 87 veículos/1000 habitantes em 1990 para 303 veículos/1000 habitantes em 2016 [Goes 2020]. Este crescimento da frota, principalmente em grandes centros, causa um grande impacto na mobilidade urbana, com o aumento de congestionamentos e consequente aumento no tempo de trajeto.

Este artigo apresenta um sistema para a consulta de eventos de trânsito históricos coletados pelo Waze, chamado de MIDET-IU. O MIDET-IU foi implementado utilizando o modelo de armazenamento de eventos MIDET [Duarte et al. 2021], apresentado no SBBB 2021. Este modelo particiona a área de interesse em células geográficas (CGs) retangulares de tamanho fixo e utiliza indexação baseada em *Bitmaps*. Através do MIDET, eventos de trânsito que ocorrem em uma mesma CG são armazenados em conjunto com o objetivo de obter um bom desempenho em consultas espaciais.

O MIDET-IU oferece uma interface gráfica que permite consultas espaço-temporais sobre a base histórica de eventos do Waze. Atualmente, a base é constituída de eventos na cidade de Joinville¹. Ela permite selecionar uma área de interesse sobre o mapa da cidade, bem como um período de tempo e tipos de eventos de trânsito a serem exibidos sobre o mapa, e que podem envolver alertas, irregularidades ou engarrafamentos.

O restante do artigo é organizado da seguinte forma. A Seção 2 apresenta trabalhos relacionados. A Seção 3 descreve as funcionalidades e o desenvolvimento do sistema MIDET-IU. O artigo conclui na Seção 4, apresentando possíveis trabalhos futuros.

2. Trabalhos Relacionados

A maioria dos sistemas que atuam na área de mobilidade urbana oferece dados em tempo real sobre a existência de engarrafamentos através de aplicativos, como o Waze e Google Maps, bem como por meio de páginas Web, como a CETSP². Outros sistemas incluem alertas, como a existência de obras e vias bloqueadas. Dentre eles podem ser citados o aplicativo Alertas Trânsito e a página de incidentes da cidade de Maryland³.

No entanto, sistemas para a consulta sobre dados de eventos de trânsito históricos são menos comuns. É possível argumentar que consultas espaço-temporais sobre eventos de trânsito podem ser tratadas como consultas em outros contextos. Estudos sobre diferentes formas de otimizar consultas que envolvem relacionamentos espaciais foram apresentados em [Chaudhry et al. 2020] e [Mahmood et al. 2019]. O diferencial do presente trabalho está no uso do modelo de armazenamento MIDET [Duarte et al. 2021], que, diferente de outros trabalhos, foi projetado para consultar dados históricos. MIDET pode ser facilmente estendido para considerar duas características importantes de eventos de trânsito: a inserção de dados em fluxo e o processamento paralelo e distribuído de consultas.

O modelo MIDET é baseado no particionamento da área geográfica de interesse em células geográficas (CGs) de tamanho fixo. Eventos que ocorrem dentro de uma mesma CG são armazenados em um mesmo bloco de um arquivo. O particionamento espacial de dados é também adotado em diversos outros sistemas [Patel and DeWitt 1996, Tsitsigkos et al. 2019]. Além disso, o MIDET adota um índice baseado em *bitmaps*, que otimiza a junção entre diferentes tipos de eventos e um armazenamento misto, que utiliza tanto o sistema de arquivos como uma base relacional.

¹Dados concedidos ao projeto *Smart Mobility*, através de uma parceria entre o Waze e a UDESC.

²<http://www.cetsp.com.br/transito-agora/>

³<https://chart.maryland.gov/incidents/index.php>

3. O Sistema MIDET-IU

O MIDET-IU considera três tipos de eventos de trânsito: alertas, engarrafamentos e irregularidades. O modelo de armazenamento adotado pelo MIDET envolve arquivos e uma base de dados relacional. São criados arquivos para cada tipo de evento. Além disso, os arquivos são compostos por blocos. Cada bloco contém eventos, ordenados cronologicamente, que ocorreram em uma mesma célula geográfica. Esta organização tem como objetivo o processamento eficiente de consultas espaço-temporais. O armazenamento dos eventos em arquivos possibilita que futuramente o sistema possa ser estendido para armazenar eventos reportados em tempo real, com um baixo tempo de inserção.

A Figura 1 apresenta o esquema da base relacional do MIDET. Ele é composto pela tabela `metadata`, que contém metadados do sistema, tais como a geometria da área de interesse e o número de linhas e colunas da grade criada sobre a área e que delimitam as células geográficas (CGs). Além disso, a tabela `tabBitmaps` contém índices *bitmap* para cada tipo de evento. Por exemplo, o atributo `alerta` é um vetor *bitmap*, indexado pelas CGs, que contém o valor 1 se há eventos de alerta na CG, e 0, caso contrário. Este índice é utilizado para otimizar consultas espaciais e operações de junção espacial, através de operações bit a bit. Quando uma determinada CG contém eventos de interesse para uma consulta, a tabela `vetor_bloco` é acessada para determinar o arquivo no qual os blocos estão armazenados, bem como seus endereços.

metadata		vetor bloco {evento}		tabBitmaps	
id	integer	arquivo	integer	periodoDiaIni	integer
areaInteresse	geometry	cglD	integer	periodoDiaFim	integer
periodo	bigint	bloco	bigint	alerta	roaringbitmap
linhas	integer			engarrafamento	roaringbitmap
colunas	integer			irregularidade	roaringbitmap
passoX	numeric			arquivoAlerta	text
passoY	numeric			arquivoEngarrafamento	text
qtdEventos	integer			arquivoIrregularidade	text

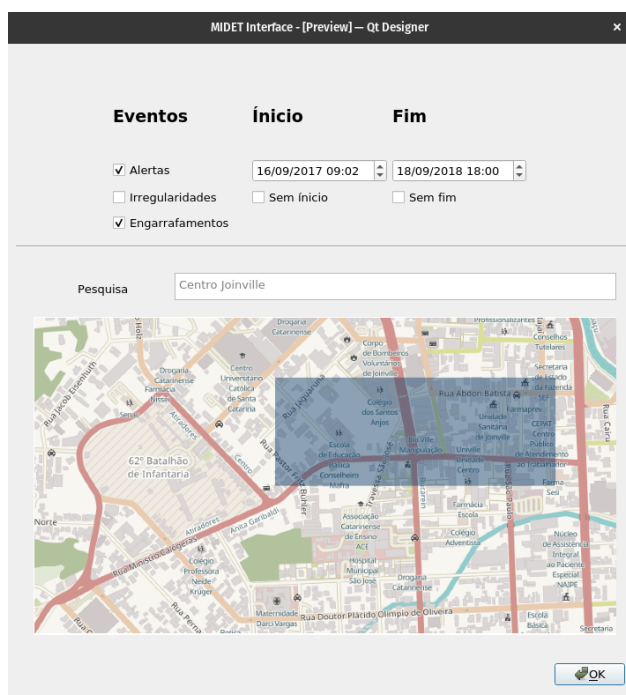
Figura 1. Esquema da Base Relacional do MIDET

3.1. Interface de Consulta

Para esta demonstração, foi utilizada uma base de dados de eventos do Waze da cidade de Joinville no período de Setembro de 2017 a Setembro de 2018. A consulta implementada consiste em obter os filtros desejados pelo usuário, ou seja, os tipos de eventos desejados, a data de início e fim dos eventos, e ainda um polígono, obrigatoriamente fechado, que define a área de interesse para a filtragem dos eventos.

A configuração inicial do mapa centraliza o mapa no centro de Joinville, cidade alvo da base de dados, mas também permite a navegação “arrastando” o mapa e também através de um campo de `Pesquisa`, que localiza bairros, ruas e até mesmo locais pelo nome.

Apesar da possibilidade de ser operado totalmente por linha de comando pelo usuário, no desenvolvimento do MIDET-IU optou-se pela construção de uma interface gráfica para facilitar o seu uso, bem como permitir a visualização imediata dos resultados. Contudo, como a interface gráfica não está acoplada ao motor de consultas do MIDET, seu uso é opcional e apenas uma camada de interação. A Figura 2a apresenta um exemplo da interface desenvolvida. A consulta que ela processa corresponde à consulta em SQL da Figura 2b.



(a) Com área selecionada na interface

```
SELECT e.geom, a.geom
FROM engarrafamento e, alerta a
WHERE e.pub_utc_date >= '...'
and e.pub_utc_date <= '...'
and a.pub_utc_date >= '...'
and a.pub_utc_date <= '...'
and ST_Intersects(e.geom, a.geom)
and ST_Intersects(e.geom, POLYGON(...))
and ST_Intersects(a.geom, POLYGON(...))
```

(b) Em SQL

Figura 2. Exemplo de consulta

3.2. Implementação

A arquitetura do sistema é apresentada na Figura 3. Detalhes de sua implementação e processamento da consulta são apresentados a seguir.

Priorizando a utilização de ferramentas de código aberto, a interface do MIDET-IU foi codificada utilizando o *QT*, um framework para construção de interfaces gráficas com a filosofia *write once run anywhere*, e com auxílio do *PyQT*, o empacotador do framework para a linguagem *Python*. Para a exibição do mapa e montagem do polígono foi utilizado o *Folium*, que por sua vez utiliza o *basemap* do *OpenStreetMap*. Para a centralização do mapa em um ponto de interesse, que pode ser digitado no campo de Pesquisa, é utilizado o motor de busca do *OpenStreetMap*.

Assim que todos os parâmetros são recebidos pelo aplicativo, este por sua vez monta a linha de comando necessária para a execução correta da consulta, e então cria um subprocesso que de fato executa o motor de consultas do MIDET. Após a finalização da

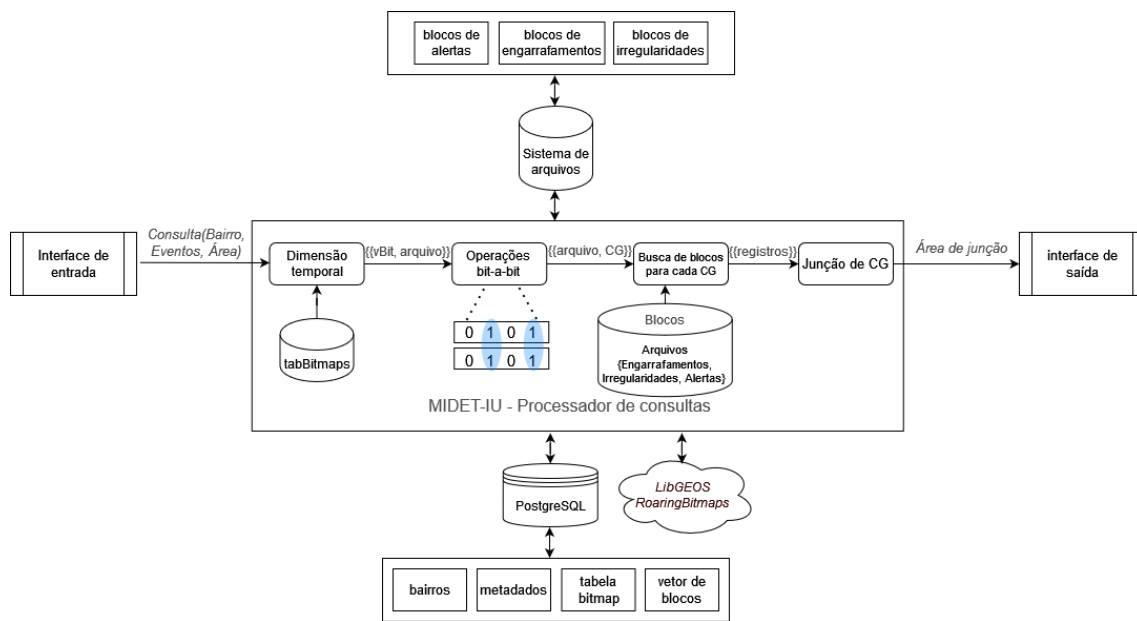


Figura 3. Arquitetura do sistema

execução da aplicação principal, a interface gráfica utiliza sua saída para plotar os eventos que foram retornados, permitindo a visualização dos pontos no mapa.

O processador de consultas foi implementado utilizando a linguagem C, juntamente com as APIs do *RoaringBitmaps* para a manipulação das estruturas de *Bitmaps* e do *LibGEOS* com as funções para dados geo-espaciais. O SGBD utilizado foi o *PostgreSQL*, que também conta com a extensão do *RoaringBitmaps* e a extensão *PostGIS* para a manipulação dos dados geo-espaciais dentro do SGBD.

3.3. Processamento da consulta

Após receber a consulta proveniente da interface do usuário, é realizada uma busca na tabela de *Bitmaps* (*tabBitmaps* - Figura 1) utilizando um período de tempo específico. Após receber os vetores de *Bitmaps*, o processador de consultas realiza operações bit-a-bit para selecionar as CGs, nas quais ocorrem as junções dos eventos desejados. Após selecionadas as CGs e utilizando os arquivos da tabela de *Bitmaps*, os blocos referentes às CGs são selecionados (tabela vetor bloco - Figura 1) e lidos para a memória, realizando enfim a junção dos registros contidos nos blocos para a consolidação do resultado final. O resultado da consulta lista os eventos obtidos e seus detalhes, cujas geometrias são retornadas no formato WKT (*well-known binary*) referente às respectivas regiões afetadas. A interface utiliza tais informações para plotar os eventos resultantes, permitindo a visualização das geometrias no mapa.

4. Conclusão

Este artigo apresentou o desenvolvimento do MIDET-IU, um sistema para executar consultas espaço-temporais sobre dados históricos do aplicativo Waze. Trabalhos futuros incluem tanto extensões nas funcionalidade do modelo MIDET para inserir dados recebidos na forma de um fluxo de dados, bem como o processamento paralelo de consultas. Novas funcionalidades do sistema MIDET-IU incluem extensões na interface com novos

tipos de filtros baseados em dados geográficos da cidade, tais como bairros e escolas, bem como a possibilidade de executar consultas envolvendo outros relacionamentos espaciais além da sobreposição geométrica. Também há interesse em expandir as possibilidades de pós-utilização do usuário, habilitando a exportação dos resultados para formatos abertos e amplamente utilizados na área de dados geoespaciais, como o *shapefile*.

Referências

- Chaudhry, N., Yousaf, M. M., and Khan, M. T. (2020). Indexing of real time geospatial data by IoT enabled devices: Opportunities, challenges and design considerations. *Journal of Ambient Intelligence and Smart Environments*, 12:281–312.
- Duarte, M. G., Pontarolo, M. V., Schroeder, R., and Hara, C. S. (2021). MIDET: A method for indexing traffic events. In *Anais do XXXVI Simpósio Brasileiro de Banco de Dados*, pages 1–12.
- Goes, G. V. (2020). Entrada de meios alternativos de transporte urbano: Histórico e perspectivas futuras. Technical Report 66, Instituto de Pesquisa Econômica Aplicada.
- Mahmood, A. R., Punni, S., and Aref, W. G. (2019). Spatio-temporal access methods: a survey (2010 - 2017). *Geoinformatica*, 23:1–36.
- Patel, J. M. and DeWitt, D. J. (1996). Partition based spatial–merge join. In *Proc. of the 1996 ACM SIGMOD international conference on Management of data*, pages 259–270.
- Tsitsigkos, D., Bouros, P., Mamoulis, N., and Terrovitis, M. (2019). Parallel in-memory evaluation of spatial joins. In *Proc. of the 27th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 516–519.