

Анализ движения наземного общественного транспорта Москвы: от PostGIS к MobilityDB

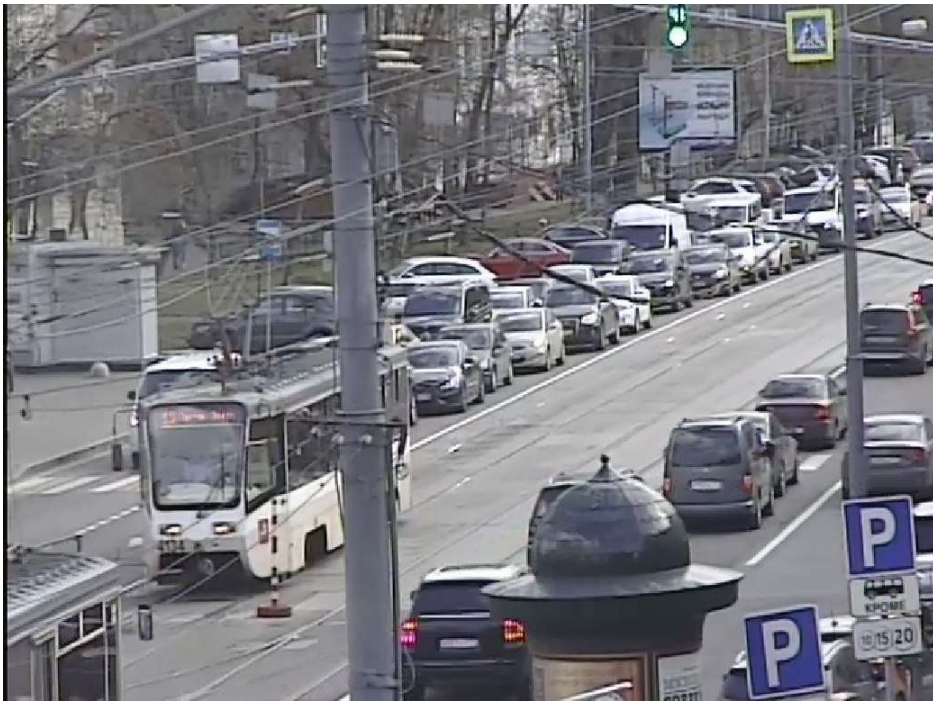
Нина Белявская

5 февраля 2020,
PgConf.ru



Служба движения ГУП «Мосгортранс»

Я работаю в отделе, задача которого – оптимизировать маршрутную сеть и расписания наземного общественного транспорта Москвы.



Мегаполису нужен комфортный общественный транспорт.

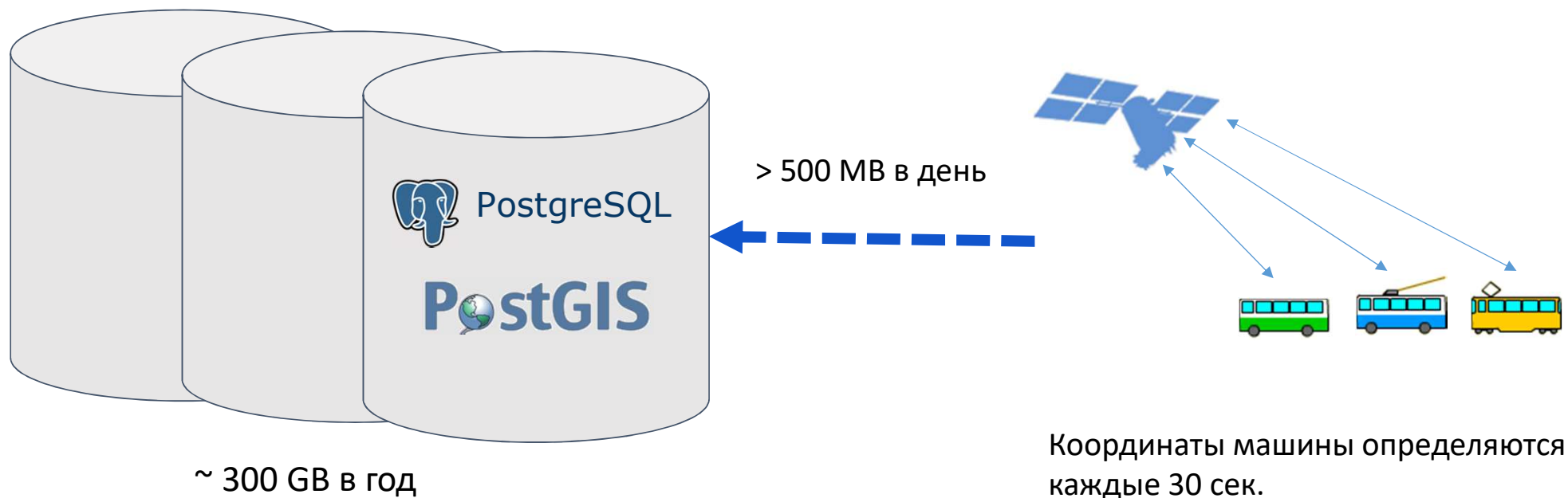
- Не только метро, но и автобусы, троллейбусы, трамваи

Транспорт должен

- ходить регулярно
- соблюдать расписание
- ходить быстро.

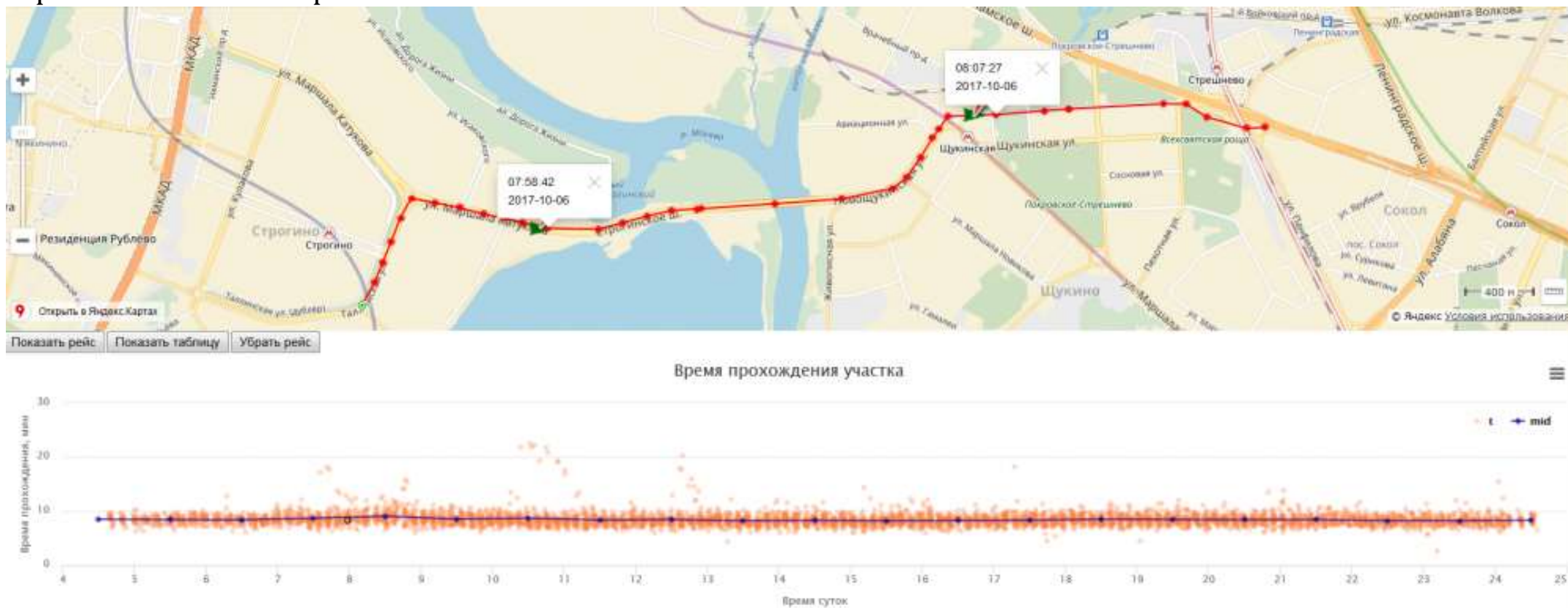
Сбор данных спутниковой навигации

Весь городской общественный транспорт Москвы оборудован навигационными приборами ГЛОНАСС.



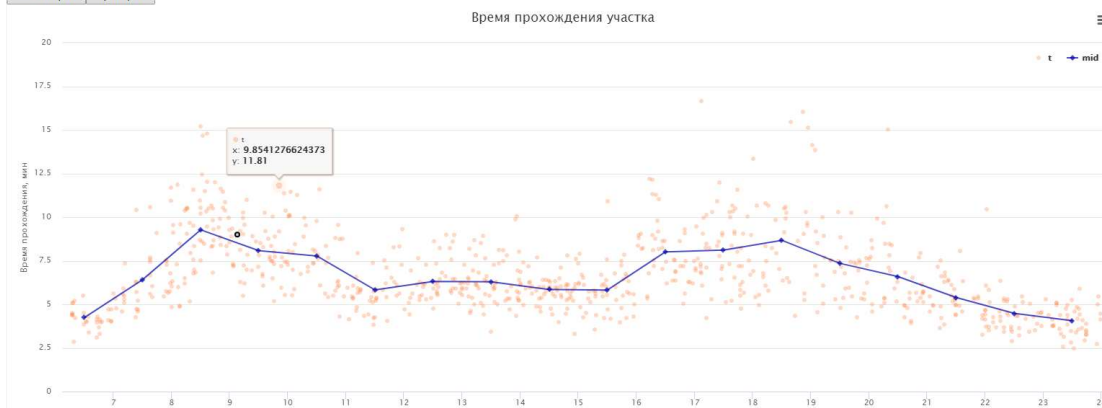
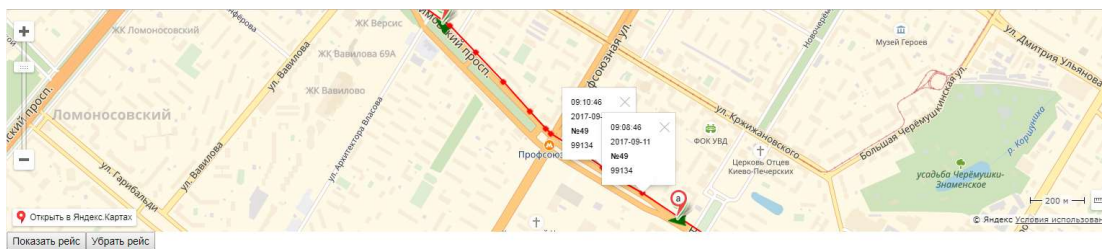
Анализ данных - пробеги

Трамвай №21 Октябрь 2017

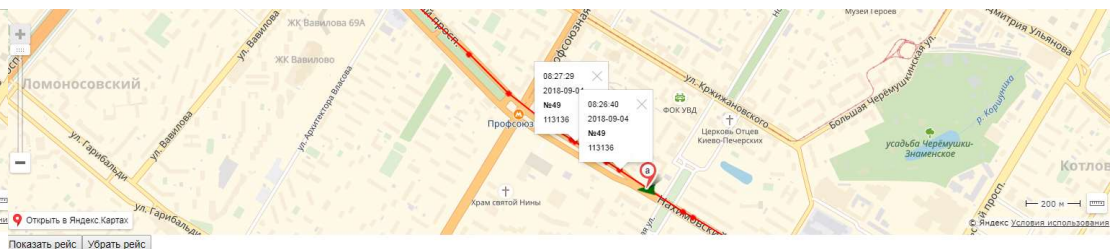


Анализ данных - пробеги

Троллейбус №49



Сентябрь 2017



Сентябрь 2018

Анализ данных – тепловые карты

Выделенные полосы HOT



MobilityDB

<https://github.com/ULB-CoDE-WIT/MobilityDB>



MobilityDB - открытое программное расширение, которое добавляет поддержку темпоральных и темпорально-пространственных объектов к PostgreSQL и PostGIS.

MobilityDB разрабатывается Computer & Decision Engineering Department of the Université Libre de Bruxelles (ULB) под руководством проф. Эстебана Зимани.

Преобразование PostGIS -> MobilityDB

```
SELECT gl_uuid, dtime,  
ST_X(gl_point) as x,  
ST_Y(gl_point) as y  
...  
ORDER BY gl_uuid, dtime
```



```
gl_uuid,  
tgeompoint '[  
    Point(x1 y1)@t1,  
    ...  
    Point(xN yN)@tN  
]
```

Разрывы треков при $t_N - t_{N-1} > 5$ мин

В PostGIS точка и время хранятся в разных колонках, для каждой точки отдельная запись.

В MobilityDB точка и время объединены, одна запись содержит целый трек.

Преобразование PostGIS -> MobilityDB

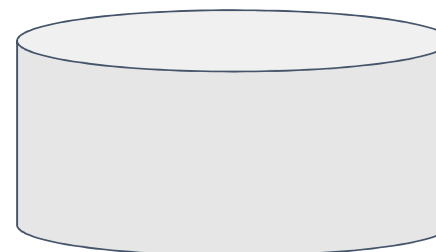
44384	2015-04-06 06:38:00	POINT(37.3826816 55.7937783)
44384	2015-04-06 06:38:30	POINT(37.3826816 55.7937783)
44384	2015-04-06 06:39:00	POINT(37.3826816 55.7937783)
44384	2015-04-06 06:39:30	POINT(37.3826816 55.7937783)
44384	2015-04-06 06:40:00	POINT(37.3826816 55.7937783)
44384	2015-04-06 06:40:30	POINT(37.3826816 55.7937783)
44384	2015-04-06 06:41:00	POINT(37.3826816 55.7937783)
44384	2015-04-06 06:41:30	POINT(37.3826816 55.7937783)
44384	2015-04-06 06:42:00	POINT(37.3826816 55.7937783)
44384	2015-04-06 06:42:30	POINT(37.3826816 55.7937783)

10 млн.записей
в день



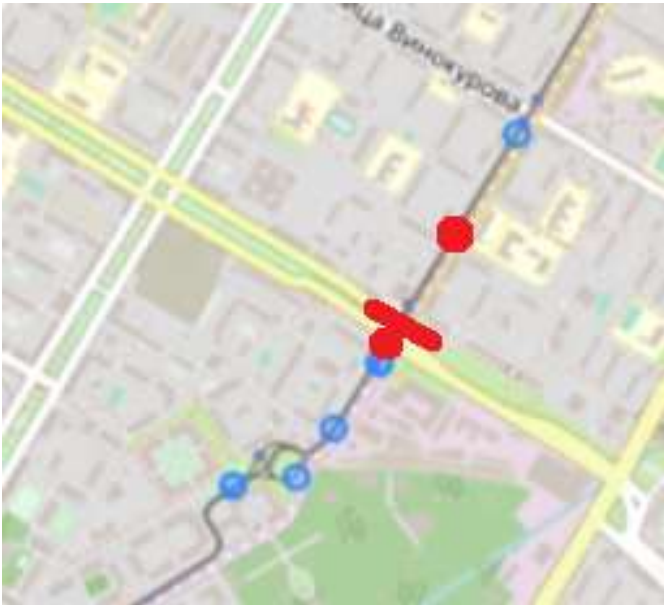
44384	[POINT(37.3826816 55.7937783)@2015-04-06 06:38:00+03, POINT(37.3826816 55.7937783)@2015-04-06 0...
44399	[POINT(37.6126166 55.7274032)@2015-04-06 07:14:29+03, POINT(37.6118683 55.7274732)@2015-04-06 0...
44399	[POINT(37.6127783 55.7265099)@2015-04-06 05:32:14+03, POINT(37.6127783 55.7265099)@2015-04-06 0...
62736	[POINT(37.6078283 55.7158566)@2015-04-06 05:35:17+03, POINT(37.607475 55.71504)@2015-04-06 05:35:...
62771	[POINT(37.6124233 55.7264416)@2015-04-06 05:07:57+03, POINT(37.6124233 55.7264416)@2015-04-06 0...
67756	[POINT(37.608135 55.7163933)@2015-04-06 04:47:23+03, POINT(37.60777 55.7153983)@2015-04-06 04:47:...
67762	[POINT(37.6093483 55.7190449)@2015-04-06 16:58:07+03, POINT(37.6094966 55.7188982)@2015-04-06 1...
67762	[POINT(37.6099266 55.7209099)@2015-04-06 04:41:30+03, POINT(37.60921 55.7190516)@2015-04-06 04:4...

15 тыс.записей
~ 5 MB в день



2GB в год

Пробеги - PostGIS

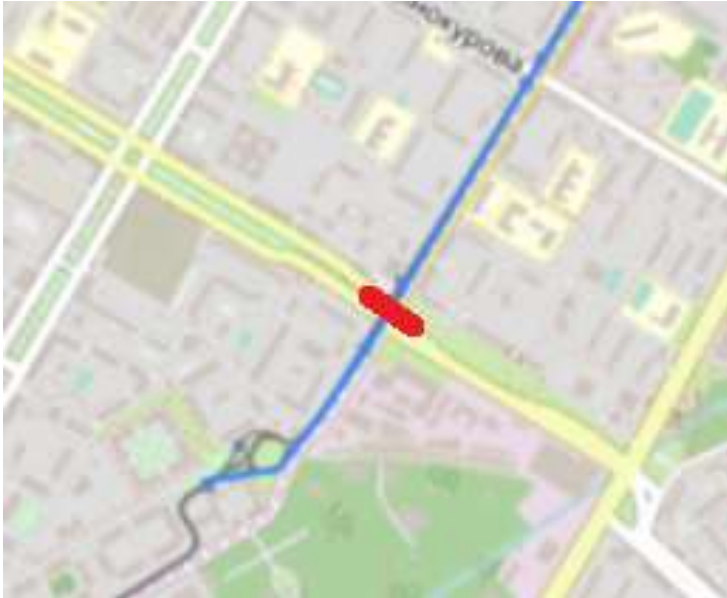


Запрос выдаёт две точки, точное время пересечения отрезка надо аппроксимировать.

```
WITH endpoint AS (SELECT ST_LineFromText(
  'LINESTRING(37.5393505 55.6936058, 37.5365505
  55.6908058)'
) AS l)
SELECT gl_uuid, t, t1
      ST_X(p0) as x0, ST_Y(p0) as y0,
      ST_X(p1) as x1, ST_Y(p1) as y1
FROM (
      SELECT gl_uuid, t, gl_point AS p0,
             lead(t) over w as t1, lead(gl_point)
over w as p1
      FROM tracks
      WINDOW w AS (PARTITION BY gl_uuid ORDER BY t)
) as points, endpoint
WHERE ST_LineCrossingDirection(l,ST_MakeLine(p0,p1))=1
AND abs(ST_X(p1) - ST_X(p0)) < 0.05
AND abs(ST_Y(p1) - ST_Y(p0)) < 0.05
ORDER BY gl_uuid, t1
```

Запрос усложнён из-за необходимости выстроить точки в последовательности с помощью window.

Пробеги - MobilityDB

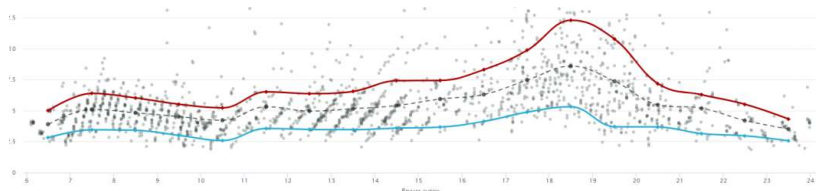


Запрос выдаёт точное время
пересечения отрезка

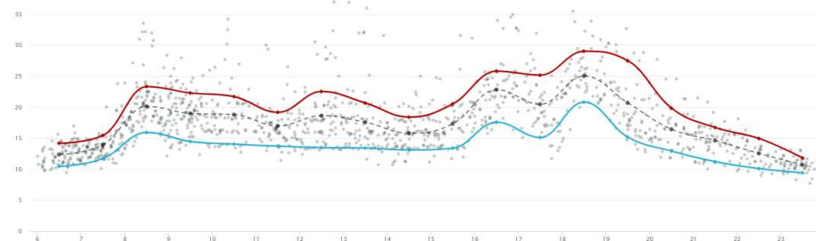
```
WITH
  endpoint as (SELECT ST_LineFromText(
    'LINESTRING(37.5393505 55.6936058, 37.5365505 55.6908058)'
  ) AS l),
  timesets as (
    SELECT gl_uuid, timestamps(atGeometry(tline,l)) as ts
    FROM mtracks, endpoint
    WHERE route IN ('49') AND tline && l),
  times as (SELECT gl_uuid, unnest(ts) as t FROM timesets),
  lines as (
    SELECT a.gl_uuid, route, t,
      getValues(atPeriod(tline,period(t-interval '15
sec',t+interval '15 sec')))) as points
    FROM times a, mtracks b
    WHERE a.gl_uuid=b.gl_uuid AND t <@ tline)
SELECT gl_uuid, route, t::time, 'B' as point
FROM lines, endpoint
WHERE ST_LineCrossingDirection(l,points)=1
ORDER BY gl_uuid, t
```

Пробеги - сравнение

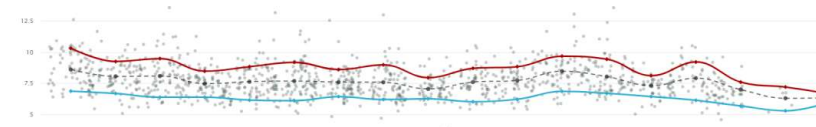
PostGIS



~60 сек



~20 сек



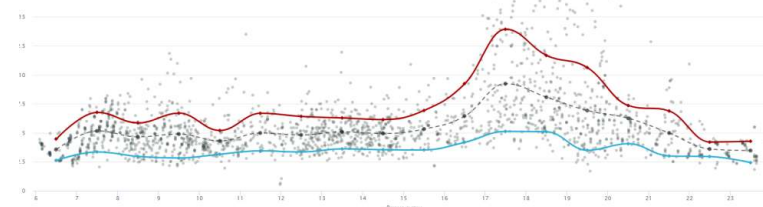
~9 сек

Автобус 224
Пр. Вернадского
- Ленинский пр.
Апрель 2015

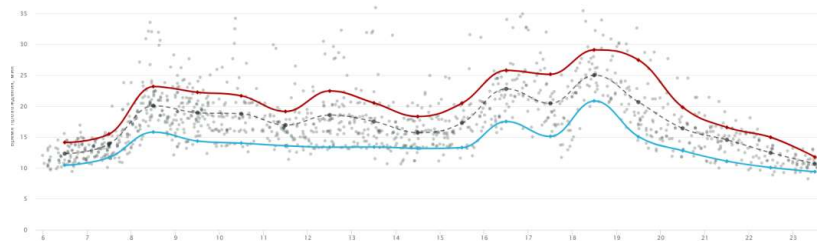
Троллейбус 49
Ул. Цюрупы
- Ленинский пр.
Февраль 2015

Трамвай 21
Строгино -
м.Щукинская
Февраль 2015

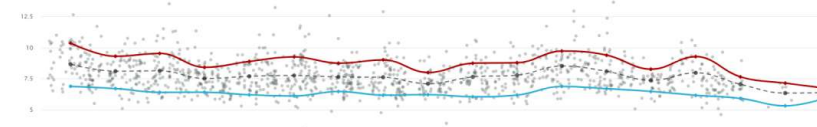
MobilityDB



~20 сек

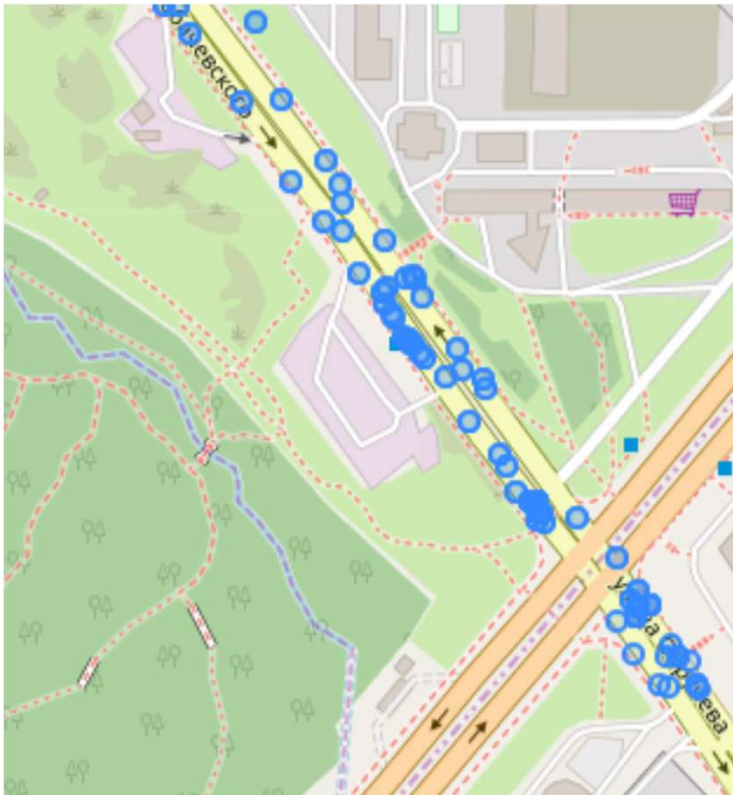


~10 сек



~5.5 сек

Карта скоростей - PostGIS



```
WITH segments AS (  
    SELECT  
    'LINESTRING(37.44654993 55.72910825,...,  
                37.44820638 55.72529983) '::geometry  
    AS line  
)  
SELECT  
    gl_uuid, t, gl_point, ST_X(gl_point) as lon,  
    ST_Y(gl_point) as lat, route,  
    ST_DistanceSphere(gl_point,line) AS r FROM  
    tracks tr, segments s  
    WHERE t>'7:00' AND t< '9:00'  
    AND ST_DistanceSphere(gl_point,line) < 20  
    ORDER BY gl_uuid
```

Получен набор точек, которые следует соединить в треки и выбрать те, что идут в правильном направлении.

Карта скоростей - MobilityDB



Сразу вычисляем скорость на отрезке.

```
WITH segments as (SELECT
  ST_Buffer(
    'LINESTRING(37.44654993 55.72910825,...,37.44820638
55.72529983)'::geography,
    20)::geometry as shape,
  degrees(ST_Azimuth(
    'POINT(37.44654993 55.72910825)'::geometry,
    'POINT(37.44820638 55.72529983)'::geometry)) as azimuth),
pbox as (
  SELECT atPeriod(tline, period '[2015-04-06 07:00,2015-04-06 09:00]') AS
tline
    FROM mtracks),
lines as (SELECT unnest(sequences(atGeometry(tline,shape))) as track
          FROM pbox, segments
          WHERE intersects(shape,tline))
SELECT
  ST_Length(getValues(track)::geography) /
  extract('epoch' from (endTimeStamp(track)-startTimeStamp(track)))
  as v
FROM lines, segments WHERE
ABS(degrees(ST_Azimuth(startValue(track),endValue(track)))-azimuth)<60
```

Карта скоростей - сравнение

PostGIS



~ 4.3 мин.

MobilityDB

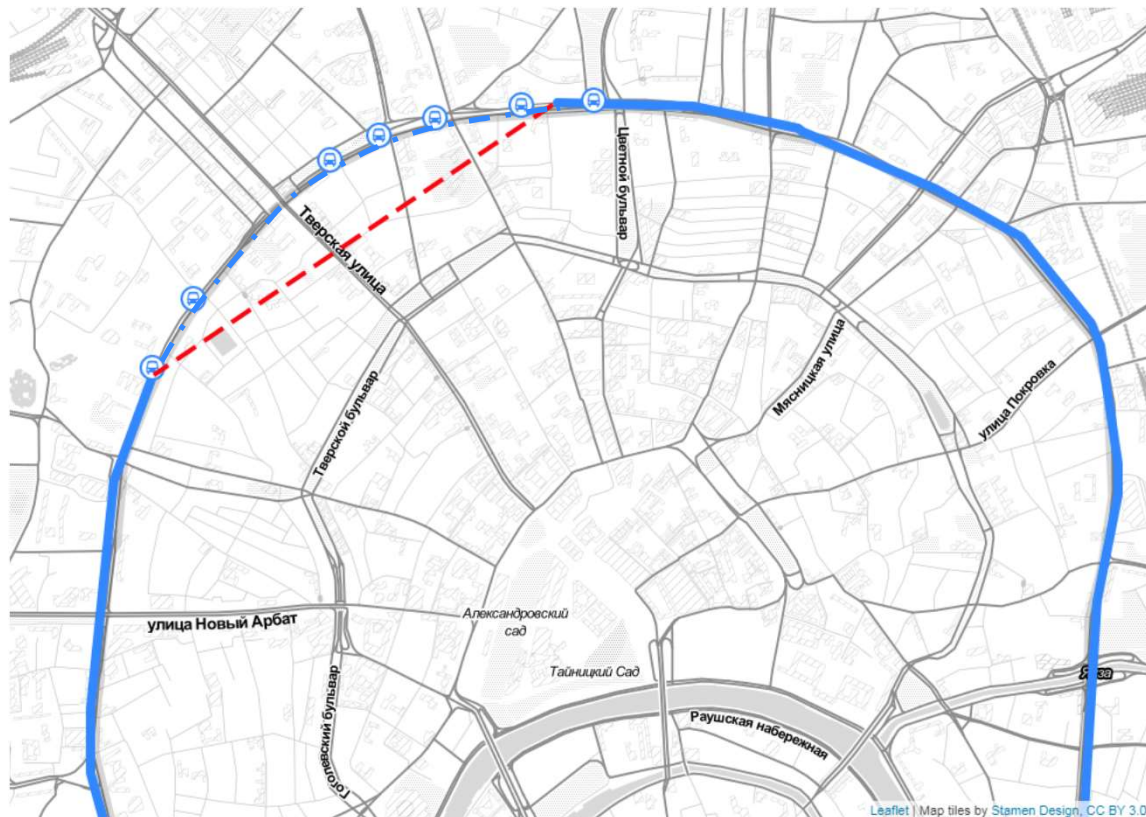


~ 2.5 мин.

Восстановление траектории

Возможны ситуации сбоя навигации, когда в течение нескольких минут данные не поступают.

- Нельзя использовать для измерения скорости, так как скорость не постоянна.
- Нужно для оценки регулярности прибытия транспорта на остановку.



Выводы

Расширение MobilityDB позволяет:

- значительно экономить дисковое пространство при хранении треков транспорта;
- ускорять расчёты транспортной аналитики.

Команда разработчиков MobilityDB готова к сотрудничеству, оперативно отвечает на вопросы и помогает решать проблемы.

**MobilityDB – новый удобный инструмент,
который стоит изучить и использовать в работе.**

<https://github.com/ULB-CoDE-WIT/MobilityDB>

Спасибо за внимание!

belyavskaya_no@mosgortrans.ru
nina@fable.ru

