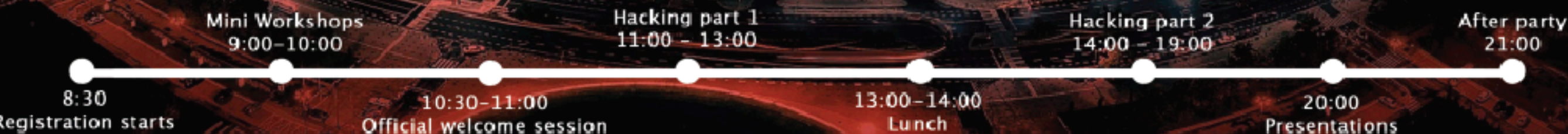


<http://bit.ly/2xwYhbS>

1. Urban Sensors is a unique hackathon designed
2. to make use of the platform,
3. architecture and data produced within
4. the VaVeL project.
- 5.
- 6.# The objective of the hackathon is to promote the use of
- 7.# VaVeL framework and Open Data to build innovative urban
- 8.# services, initiate new startups and extend services
- 9.# offered by cities and already existing enterprises.
- 10.
11. The hackathon is opened for
12. data scientists,
13. computer scientists, statisticians, data artists
14. and others obsessed with data.

URBAN SENSORS THE DATA SCIENCE HACKATHON

26.09.2017



PARTNERS


















The goal of the VaVeL project is to radically advance our ability to use urban data in applications that can **identify and address citizen needs and improve urban life**. This project will develop a general purpose framework for managing and mining multiple heterogeneous urban data streams for cities become more efficient, productive and resilient.



- A. Online data for trams and buses
- B. Offline data for trams and buses
- C. Offline data from cellphones

Offline data for trams and buses

Offline data for trams and buses

 192.168.137.209/hackathon/buses/2017-09-03/			
Index of /hackathon/buses/2017-09-03			
<u>Name</u>	<u>Last modified</u>	<u>Size</u>	<u>Description</u>
<hr/>			
 Parent Directory		-	
 part-0-0	2017-09-25 16:06	120M	
 part-0-1	2017-09-25 16:06	120M	
 part-0-2	2017-09-25 16:06	120M	
 part-0-3	2017-09-25 16:06	120M	
 part-0-4	2017-09-25 16:06	120M	
 part-0-5	2017-09-25 16:06	120M	
 part-0-6	2017-09-25 16:06	120M	
 part-0-7	2017-09-25 16:06	120M	
 part-0-8	2017-09-25 16:07	120M	
 part-0-9	2017-09-25 16:07	120M	
 part-0-10	2017-09-25 16:06	33M	
<hr/>			

Access data from

<http://192.168.137.209/hackathon>

1 20170628;702;1;2017-08-31 23:59:58;21.258522;52.108589;21.258522;52.108589;MOVING;0.0;;;0.0;0.0;0.0;;0.0;0.0;0.0;
2 20170628;213;3;2017-08-31 23:59:55;21.201462;52.171064;21.201462;52.171064;MOVING;0.0;;;0.0;0.0;0.0;;0.0;0.0;0.0;
3 20170628;179;011;2017-08-31 23:59:56;20.992993;52.185578;20.992993;52.185578;MOVING;0.0;;;0.0;0.0;0.0;;0.0;0.0;0.0;
4 20170628;179;9;2017-08-31 23:59:54;21.002383;52.181693;21.002383;52.181693;MOVING;0.0;;;0.0;0.0;0.0;;0.0;0.0;0.0;
5 20170628;194;2;2017-08-31 23:59:52;20.863243;52.207873;20.863243;52.207873;MOVING;32.0;4149-PKP Gołabki;2017-09-01
6 20170628;N46;1;2017-08-31 23:59:53;20.992987;52.242245;20.992987;52.242245;MOVING;-58.0;7082-Nowolipki;2017-08-31
7 20170628;108;3;2017-08-31 23:59:54;21.087724;52.194533;21.087724;52.194533;MOVING;34.0;3079-EC Siekierki;2017-08-31
8 20170628;715;2;2017-08-31 23:59:49;21.017815;52.150628;21.017815;52.150628;MOVING;-46.0;3013-Poleczki;2017-09-01
9 20170628;165;3;2017-08-31 23:59:58;21.00699;52.177951;21.00699;52.177951;MOVING;18.0;3249-Instytut Fizyki;2017-08-31
10 20170628;N01;3;2017-08-31 23:59:51;21.002301;52.187967;21.002301;52.187967;MOVING;378.0;3240-Woronicza;2017-08-31
11 20170628;N01;1;2017-08-31 23:59:55;20.921234;52.258379;20.921234;52.258379;MOVING;99.0;5161-Nowe Bemowo;2017-08-31
12 20170628;179;2;2017-08-31 23:59:54;21.068656;52.128946;21.068644;52.128964;STOPPED;0.0;;;0.0;0.0;0.0;;0.0;0.0;0.0;
13 20170628;N36;314;2017-08-31 23:59:59;21.007467;52.209486;21.007467;52.209486;MOVING;95.0;7094-Biblioteka Narodowa;2017-08-31
14 20170628;L11;5;2017-08-31 23:59:59;20.941372;52.40086;20.94137;52.400864;STOPPED;0.0;1800-PKP Legionowo;2017-09-01
15 20170628;N01;2;2017-08-31 23:59:52;20.914551;52.20055;20.914551;52.20055;MOVING;-28.0;4214-PKP Włochy;2017-09-01
16 20170628;N25;193;2017-08-31 23:59:56;21.063978;52.228753;21.063978;52.228753;MOVING_SLOWLY;96.0;2097-Saska;2017-08-31
17 20170628;N88;212;2017-08-31 23:59:59;20.922544;52.19624;20.922555;52.196231;STOPPED;0.0;;;0.0;0.0;0.0;;0.0;0.0;0.0;
18 20170628;739;4;2017-08-31 23:59:55;21.018841;52.098314;21.018841;52.098314;MOVING;75.0;3160-Mysiadło;2017-08-31
19 20170628;739;2;2017-08-31 23:59:56;21.036694;52.09304;21.036694;52.09304;MOVING;-43.0;3344-Agatowa;2017-09-01 00:00:00
20 20170628;179;4;2017-08-31 23:59:55;21.058528;52.150584;21.058528;52.150584;MOVING;-95.0;3208-Płaskowickiej;2017-08-31
21 20170628;N95;191;2017-08-31 23:59:58;20.898772;52.238502;20.898772;52.238502;MOVING;317.0;5050-Os.Górczewska;2017-08-31
22 20170628;154;3;2017-08-31 23:59:57;20.897951;52.255548;20.897951;52.255548;MOVING;47.0;5054-Archimedes;2017-08-31
23 20170628;194;4;2017-08-31 23:59:58;20.930747;52.225145;20.930747;52.225145;MOVING_SLOWLY;0.0;5011-Cm.Wolski;2017-08-31
24 20170628;197;8;2017-09-01 00:00:00;20.953239;52.276417;20.953239;52.276417;MOVING;-20.0;6021-Bielany-Ratusz;2017-08-31
25 20170628;N03;202;2017-08-31 23:59:59;21.042981;52.249875;21.042981;52.249875;MOVING_SLOWLY;129.0;1001-Kijowska;2017-08-31
26 20170628;N02;4;2017-08-31 23:59:53;20.928017;52.290842;20.928017;52.290842;MOVING_SLOWLY;0.0;6059-Metro Młociny;2017-08-31
27 20170628;134;1;2017-08-31 23:59:57;21.051963;52.312345;21.051963;52.312345;MOVING;-53.0;1283-Współczesna;2017-09-01
28 20170628;N02;3;2017-08-31 23:59:54;20.939054;52.277032;20.939054;52.277032;MOVING;24.0;6022-Przybyszewskiego;2017-08-31
29 20170628;178;2;2017-08-31 23:59:53;20.900988;52.189309;20.900988;52.189309;MOVING;3578.0;4336-Hassa;2017-08-31 23:59:59
30 20170628;N88;176;2017-08-31 23:59:54;20.970283;52.203953;20.970283;52.203953;MOVING;133.0;4007-Korotyńskiego;2017-08-31
31 20170628;197;2;2017-08-31 23:59:57;20.923268;52.196529;20.923268;52.196529;MOVING_SLOWLY;0.0;;;0.0;0.0;0.0;;0.0;0.0;0.0;
32 20170628;178;6;2017-08-31 23:59:57;20.983437;52.230015;20.983437;52.230015;MOVING;3547.0;5041-Karolkowa;2017-08-31
33 20170628;146;4;2017-08-31 23:59:54;21.175461;52.157865;21.175461;52.157865;MOVING;144.0;2055-Bysławska;2017-08-31
34 20170628;115;4;2017-08-31 23:59:59;21.117832;52.234177;21.117832;52.234177;STOPPED;0.0;;;0.0;0.0;0.0;;0.0;0.0;0.0;
35 20170628;146;1;2017-08-31 23:59:56;21.113505;52.236478;21.113505;52.236478;MOVING;-67.0;2255-Fieldorfa;2017-08-31
36 20170628;N16;199;2017-08-31 23:59:58;21.002093;52.227854;21.002093;52.227854;MOVING;0.0;;;0.0;0.0;0.0;;0.0;0.0;0.0;
37 20170628;147;2;2017-08-31 23:59:53;21.052018;52.253198;21.052018;52.253198;STOPPED;0.0;;;0.0;0.0;0.0;;0.0;0.0;0.0;
38 20170628;147;5;2017-08-31 23:59:55;21.243267;52.221362;21.243267;52.221362;MOVING;0.0;;;0.0;0.0;0.0;;0.0;0.0;0.0;
39 20170628;104;3;2017-08-31 23:59:56;21.033426;52.334909;21.033427;52.334918;STOPPED;0.0;1176-Brzeziny;2017-09-01
40 20170628;132;1;2017-08-31 23:59:58;21.044976;52.279027;21.044976;52.279027;MOVING;4618.0;1281-Ostródzka;2017-08-31
41 20170628;N61;121;2017-08-31 23:59:57;21.005163;52.22883;21.005163;52.22883;MOVING_SLOWLY;0.0;7002-Dw.Centralny;2017-08-31
42 20170628;N11;201;2017-08-31 23:59:54;21.063738;52.325253;21.063738;52.325253;MOVING;-6.0;1183-Kąty Grodziskie;2017-08-31
43 20170628;N62;128;2017-08-31 23:59:57;20.990843;52.257051;20.990843;52.257051;MOVING;196.0;7019-Dw.Gdański;2017-08-31
44 20170628;N03;3;2017-08-31 23:59:57;20.940995;52.329279;20.940995;52.329279;MOVING;27.0;1298-Stefanika;2017-08-31

"versionID", #String
"line", #String,
"brigade", #String
"time", #String,
"lon", #Double,
"lat", #Double,
"rawLon", #Double,
"rawLat", #Double,
"status", #String,
"delay", #String,
"delayAtStop", #String,
"plannedLeaveTime", #String,
"nearestStop", #String,
"nearestStopDistance", #Double,
"nearestStopLon", #Double,
"nearestStopLat", #Double,
"previousStop", #String,
"previousStopLon", #Double,
"previousStopLat", #Double,

"previousStopDistance", #Double,
"previousStopArrivalTime", #String,
"previousStopLeaveTime", #String,
"nextStop", #String,
"nextStopLon", #Double,
"nextStopLat", #Double,
"nextStopDistance", #Double,
"nextStopTimetableVisitTime", #String,
"courseIdentifier", #String,
"courseDirection", #String,
"timetableIdentifier", #String,
"timetableStatus", #String,
"receivedTime", #String,
"processingFinishedTime", #String
"onWayToDepot", #String
"overlapsWithNextBrigade", #String
"atStop", #String
"overlapsWithNextBrigadeStopLineBrigade", #String
"speed"

```

library("dplyr")
library("lubridate")
library("ggplot2")

# read all files
files <- list.files(pattern = "part")
parts <- lapply(files, read_csv2, col_names = FALSE)
one_day <- do.call(rbind, parts)

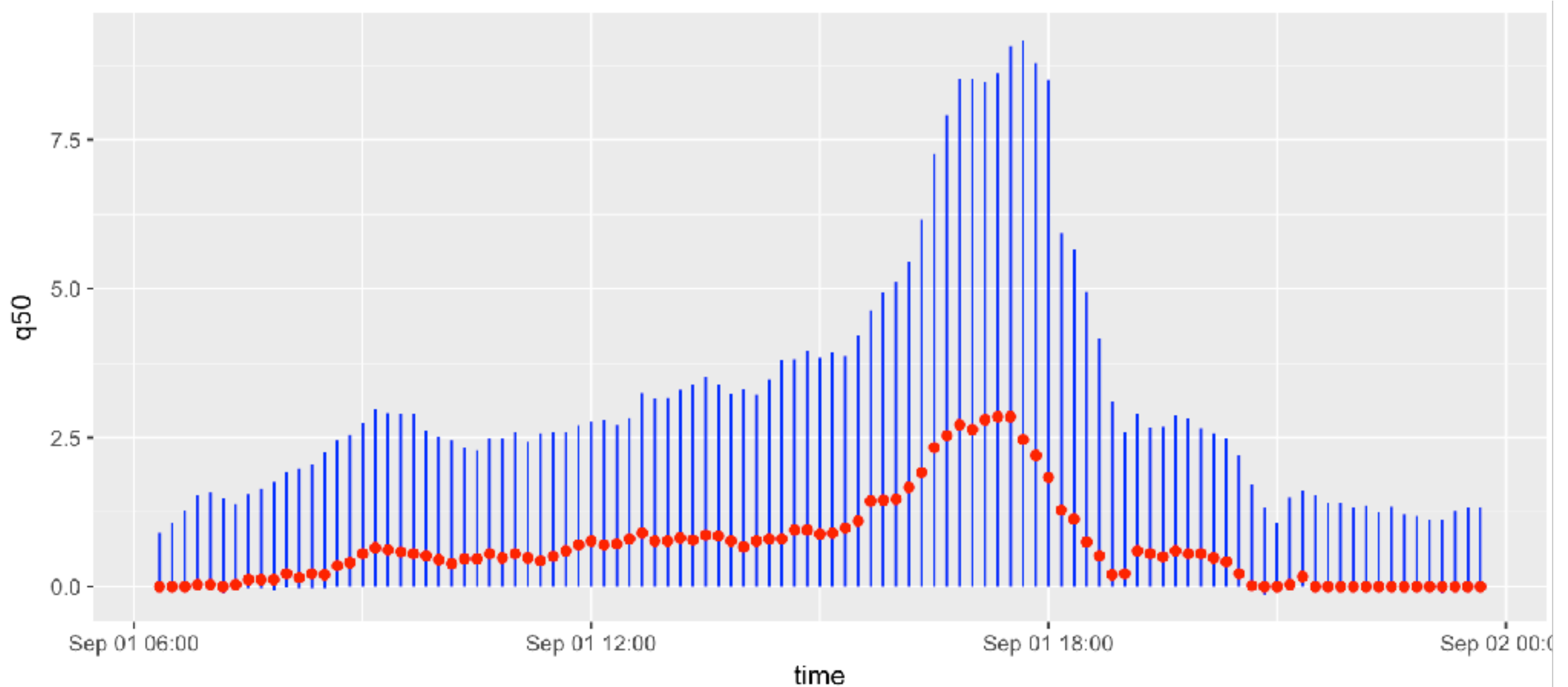
# enrich raw data
colnames(one_day) <- c( "versionID", ..... )
one_day$time16 <- substr(one_day$time, 1, 15)
one_day$delay <- as.numeric(as.character(one_day$delay))

# calculate aggregates
quant <- one_day[,-39] %>%
  filter(time16 > "2017-09-01 06:1",
         time16 < "2017-09-01 23:5") %>%
  mutate(delay=delay/60) %>%
  group_by(time16) %>%
  summarise(q05 = quantile(delay, 0.5, na.rm=TRUE),
            q25 = quantile(delay, 0.25, na.rm=TRUE),
            q50 = quantile(delay, 0.50, na.rm=TRUE),
            q75 = quantile(delay, 0.75, na.rm=TRUE),
            q95 = quantile(delay, 0.95, na.rm=TRUE)) %>%
  mutate(time = ymd_hm(paste0(time16,"0")))

```


Offline data for trams and buses

```
ggplot(quant, aes(time, q50)) +  
  geom_linerange(aes(ymin=q25, ymax=q75), color="blue") +  
  geom_point(color="red")
```



Online data for trams and buses

Online data for trams and buses

Read current positions and characteristics of trams and buses with our REST service.

You will need a token (should have one already).

Use the short/full hooks for smaller / longer list of characteristics.

```
curl -i \  
  -H 'Accept:application/json' \  
  -H 'Authorization:Token 35dbb2ebd27b23cfbec359dbd560adf2d4a5b27b' \  
  "https://vavel.mini.pw.edu.pl/api/vehicles/v1/short/?line=10,17,33"
```

```
curl -i \  
  -H 'Accept:application/json' \  
  -H 'Authorization:Token 35dbb2ebd27b23cfbec359dbd560adf2d4a5b27b' \  
  "https://vavel.mini.pw.edu.pl/api/vehicles/v1/full/?line=10,17,33"
```

Online data for trams and buses

```
library("rvest")  
library("jsonlite")
```

```
# set token and list of lines
```

```
linie <- "10,17,33"
```

```
token2 <- "35dbb2ebd27b23cfbec359dbd560adf2....."
```

```
# get data
```

```
res <- GET(url = paste0("https://vavel.mini.pw.edu.pl/api/vehicles/v1/short/?line=", linie),  
          add_headers(Authorization = paste("Token", token2)))
```

```
jsonlite::fromJSON(as.character(res))
```

```
# brigade line delay      status      time      lon      lat  
#2118877 025 10 0 STOPPED 2017-09-25T23:47:39Z 21.00311  
#2224342 8 17 90 MOVING 2017-09-26T00:04:51Z 20.99225  
#2318883 17 17 0 STOPPED 2017-09-25T21:43:11Z 21.00261  
#2418885 9 10 0 MOVING 2017-09-25T21:19:04Z 21.00261  
#2523154 11 10 0 MOVING 2017-09-25T21:22:37Z 20.96954  
#2629925 036 17 0 MOVING 2017-09-25T20:52:11Z 20.93416
```

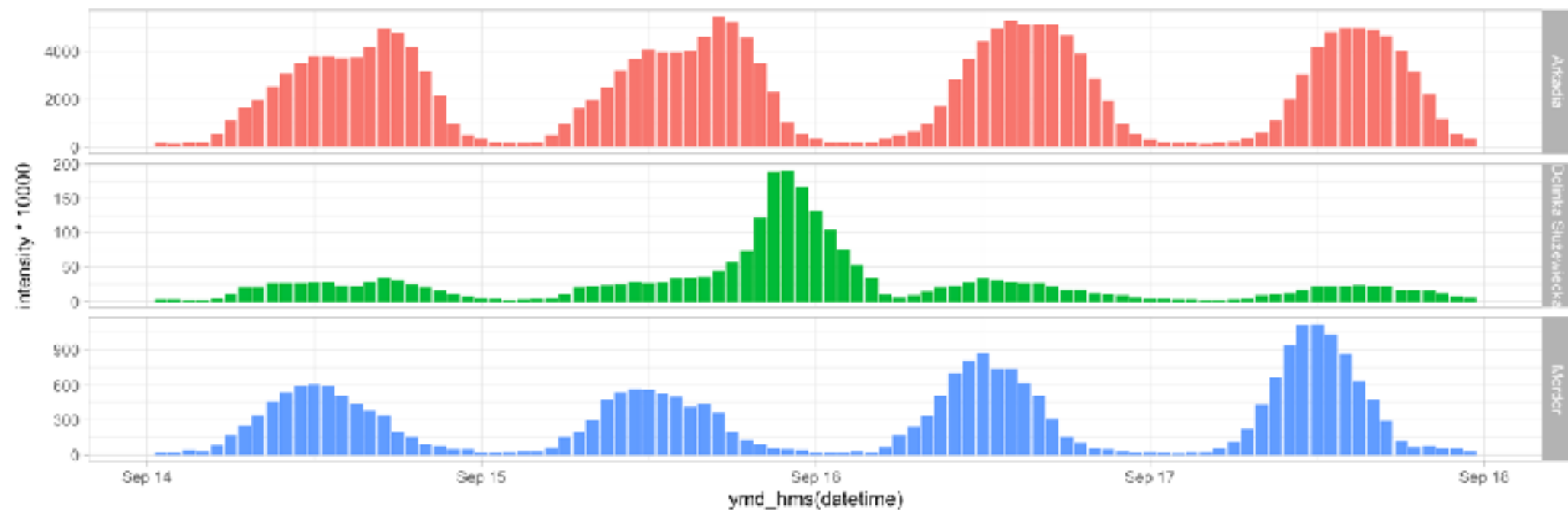

Offline data from cellphones

Offline data from cellphones

measurements.csv									
1	datetime,zoneid,intensity,geom4326,geom3857								
2	2017-09-04	00:00:00	1	0.02072913149322	"MULTIPOLYGON(((20.9832771275726	52.259601516439	20.9876244827842	52.259569327
3	2017-09-04	01:00:00	1	0.0135282957291873	"MULTIPOLYGON(((20.9832771275726	52.259601516439	20.9876244827842	52.259569327
4	2017-09-04	02:00:00	1	0.011225094014391	"MULTIPOLYGON(((20.9832771275726	52.259601516439	20.9876244827842	52.259569327
5	2017-09-04	03:00:00	1	0.0143463041177372	"MULTIPOLYGON(((20.9832771275726	52.259601516439	20.9876244827842	52.259569327
6	2017-09-04	04:00:00	1	0.0168584621028028	"MULTIPOLYGON(((20.9832771275726	52.259601516439	20.9876244827842	52.259569327
7	2017-09-04	05:00:00	1	0.0429707118381188	"MULTIPOLYGON(((20.9832771275726	52.259601516439	20.9876244827842	52.259569327
8	2017-09-04	06:00:00	1	0.0940139802748665	"MULTIPOLYGON(((20.9832771275726	52.259601516439	20.9876244827842	52.259569327
9	2017-09-04	07:00:00	1	0.148338362150815	"MULTIPOLYGON(((20.9832771275726	52.259601516439	20.9876244827842	52.259569327
10	2017-09-04	08:00:00	1	0.201964742592399	"MULTIPOLYGON(((20.9832771275726	52.259601516439	20.9876244827842	52.259569327
11	2017-09-04	09:00:00	1	0.254855955120922	"MULTIPOLYGON(((20.9832771275726	52.259601516439	20.9876244827842	52.259569327
12	2017-09-04	10:00:00	1	0.346754862486224	"MULTIPOLYGON(((20.9832771275726	52.259601516439	20.9876244827842	52.259569327
13	2017-09-04	11:00:00	1	0.417827595238002	"MULTIPOLYGON(((20.9832771275726	52.259601516439	20.9876244827842	52.259569327
14	2017-09-04	12:00:00	1	0.492589829858719	"MULTIPOLYGON(((20.9832771275726	52.259601516439	20.9876244827842	52.259569327
15	2017-09-04	13:00:00	1	0.502275529140201	"MULTIPOLYGON(((20.9832771275726	52.259601516439	20.9876244827842	52.259569327
16	2017-09-04	14:00:00	1	0.478188737270444	"MULTIPOLYGON(((20.9832771275726	52.259601516439	20.9876244827842	52.259569327
17	2017-09-04	15:00:00	1	0.49299114316749	"MULTIPOLYGON(((20.9832771275726	52.259601516439	20.9876244827842	52.259569327
18	2017-09-04	16:00:00	1	0.544936393550535	"MULTIPOLYGON(((20.9832771275726	52.259601516439	20.9876244827842	52.259569327
19	2017-09-04	17:00:00	1	0.540932366728436	"MULTIPOLYGON(((20.9832771275726	52.259601516439	20.9876244827842	52.259569327
20	2017-09-04	18:00:00	1	0.503936181964235	"MULTIPOLYGON(((20.9832771275726	52.259601516439	20.9876244827842	52.259569327
21	2017-09-04	19:00:00	1	0.432903558681025	"MULTIPOLYGON(((20.9832771275726	52.259601516439	20.9876244827842	52.259569327
22	2017-09-04	20:00:00	1	0.289685766296383	"MULTIPOLYGON(((20.9832771275726	52.259601516439	20.9876244827842	52.259569327
23	2017-09-04	21:00:00	1	0.165402704985864	"MULTIPOLYGON(((20.9832771275726	52.259601516439	20.9876244827842	52.259569327
24	2017-09-04	22:00:00	1	0.0777710017591791	"MULTIPOLYGON(((20.9832771275726	52.259601516439	20.9876244827842	52.259569327
25	2017-09-04	23:00:00	1	0.0495021697311057	"MULTIPOLYGON(((20.9832771275726	52.259601516439	20.9876244827842	52.259569327
26	2017-09-05	00:00:00	1	0.0258236214342157	"MULTIPOLYGON(((20.9832771275726	52.259601516439	20.9876244827842	52.259569327
27	2017-09-05	01:00:00	1	0.0167922516247921	"MULTIPOLYGON(((20.9832771275726	52.259601516439	20.9876244827842	52.259569327
28	2017-09-05	02:00:00	1	0.013877501604484	"MULTIPOLYGON(((20.9832771275726	52.259601516439	20.9876244827842	52.259569327
29	2017-09-05	03:00:00	1	0.0144412562288536	"MULTIPOLYGON(((20.9832771275726	52.259601516439	20.9876244827842	52.259569327
30	2017-09-05	04:00:00	1	0.01860694859375	"MULTIPOLYGON(((20.9832771275726	52.259601516439	20.9876244827842	52.259569327
31	2017-09-05	05:00:00	1	0.0479311206877238	"MULTIPOLYGON(((20.9832771275726	52.259601516439	20.9876244827842	52.259569327
32	2017-09-05	06:00:00	1	0.108734642045238	"MULTIPOLYGON(((20.9832771275726	52.259601516439	20.9876244827842	52.259569327
33	2017-09-05	07:00:00	1	0.159912834998777	"MULTIPOLYGON(((20.9832771275726	52.259601516439	20.9876244827842	52.259569327
34	2017-09-05	08:00:00	1	0.19979875801994	"MULTIPOLYGON(((20.9832771275726	52.259601516439	20.9876244827842	52.259569327
35	2017-09-05	09:00:00	1	0.237895084988897	"MULTIPOLYGON(((20.9832771275726	52.259601516439	20.9876244827842	52.259569327
36	2017-09-05	10:00:00	1	0.297474563366448	"MULTIPOLYGON(((20.9832771275726	52.259601516439	20.9876244827842	52.259569327
37	2017-09-05	11:00:00	1	0.365783357280594	"MULTIPOLYGON(((20.9832771275726	52.259601516439	20.9876244827842	52.259569327
38	2017-09-05	12:00:00	1	0.397153438597114	"MULTIPOLYGON(((20.9832771275726	52.259601516439	20.9876244827842	52.259569327
39	2017-09-05	13:00:00	1	0.435550782100287	"MULTIPOLYGON(((20.9832771275726	52.259601516439	20.9876244827842	52.259569327

Offline data from cellphones

```
library("curl")
library("httr")
library("dplyr")
library("lubridate")
library("tidyr")
library("ggplot2")
```



```
# read measurements from phones
```

```
dane <- read.csv("measurements.csv", stringsAsFactors = FALSE)
```

```
selected <- filter(dane,
  zoneid %in% c(1, 553, 751),
  datetime > "2017-09-14 00:00:00")
```

```
selected$zoneid <- factor(selected$zoneid,
  labels=c("Arkadia", "Dolinka Służewska", "Mordor"))
```

```
# plot data for 3 zones
```

```
ggplot(selected, aes(ymd_hms(datetime), intensity*10000, fill=factor(zoneid))) +
  geom_col() + facet_grid(zoneid~., scales = "free_y") +
  theme_light()
```

Offline data from cellphones

```
library("sf")
library("animation")

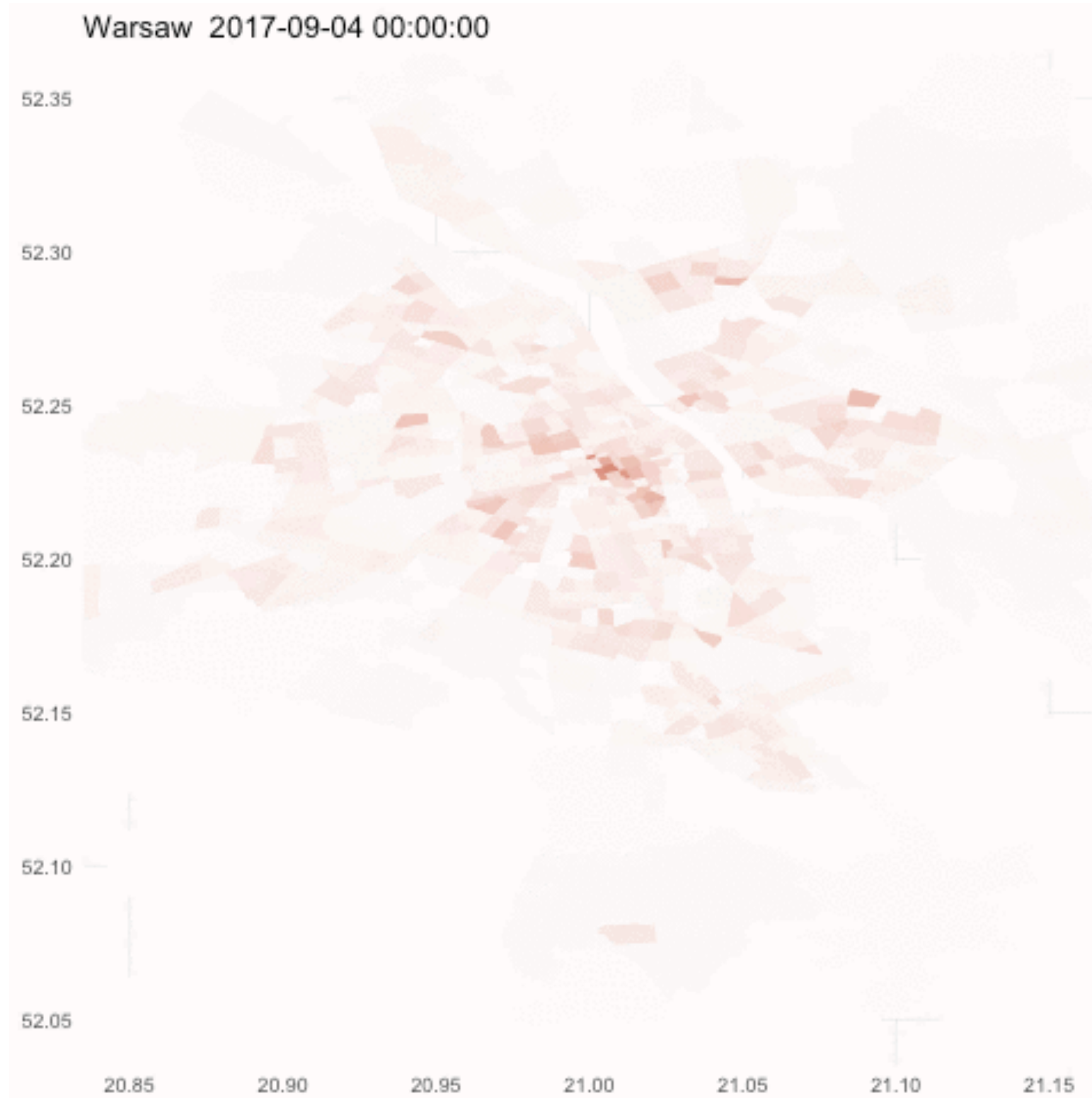
saveGIF({
  for (czas in czasy[1:24]) {
    daneGodzina <- filter(dane, datetime == czas)

    tmp <- gsub(daneGodzina[,4], pattern = "[^0-9\\.]", replacement = "")
    tmp2 <- strsplit(tmp, split=",")
    tmp3 <- lapply(tmp2, strsplit, split = " ")
    tmp4 <- lapply(tmp3, function(x) {
      tt <- sapply(x, as.numeric)
      tt <- cbind(tt, tt[,1])
      st_polygon(list(t(tt)))
    })

    geometry <- do.call(st_sfc, tmp4)
    areas <- 100/(st_area(geometry) + 5*10^-6) # scaling factor
    ss <- st_sf(intensity = pmin(daneGodzina$intensity * 10000 * areas/2649104, 2000), geometry )

    pl <- ggplot(ss) +
      geom_sf(aes(fill = intensity, color=intensity)) +
      theme_minimal() +
      scale_fill_gradient(limits = c(0, 2000), low = "#ffffff", high = "#990000") +
      scale_color_gradient(limits = c(0, 2000), low = "#ffffff", high = "#990000") +
      ggtitle(paste("Warsaw ", czas)) +
      theme(legend.position = "none") +
      coord_sf(xlim=c(20.85,21.15), ylim = c(52.05,52.35))
    print(pl)
  }
}, clean = FALSE, interval=0.2)
```

Offline data from cellphones



Your turn

How to start? Propositions

Programming

Mobile app that will show closest trams / buses

Webapp that will compare two alternative routes

App that will estimate number of people during an event

Analysis

Detect or predict future delays

Report current level of traffic
'korkometr'

Compare a single event to a baseline

Data vis

Show distribution of delays per time / line / place

Show where are closes trams / buses

Show how single event (marathon) affected data