Philipps Universität Marburg

# Datenintegration – 3rd. Phase

Johannes Gesk

Niklas Standop

# Adjustments on the showcase

**Originally:** With the given data from the periods of covid, the Ukraine invasion and the “9€” ticket, is it possible to see correlations between delays and to give a forecast for the “9€” ticket time in Germany.

Philipps Universität Marburg

**New:** With the given data which agency does produce the most delays and needs the most attention to avoid the delays.

# Data cleaning

3 rows × 8 columns pd.Dataframe

| | Id | IsDeleted | TripUpdate.T... | TripUpdate.T... | TripUpdate.T... | TripUpdate.T... | TripUpdate.T... | TripUpdate.StopTimeUpdate |
|---|---|---|---|---|---|---|---|---|
| 0 | 292595792 | False | 292595792 | 43235_2 | 21:07:00 | 20230702 | Scheduled | [{'StopId': '000009013927', 'Departure': {'Del... |
| 1 | 284276697 | False | 284276697 | 51652_3 | 13:10:00 | 20230702 | Scheduled | [{'StopId': '000009054797', 'Departure': {'Del... |
| 2 | 291600955 | False | 291600955 | 14205_3 | 13:38:00 | 20230702 | Scheduled | [{'StopId': '000000990042', 'Departure': {'Del... |

3 rows × 7 columns pd.Dataframe

| TripId | StopId | StopSequence | ArrivalDelay | ArrivalTime | DepartureDelay | DepartureTime | ScheduleRelation |
|---|---|---|---|---|---|---|---|
| 292595792 | 9013927 | NaN | NaN | NaN | 0.0 | NaN | Scheduled |
| 284276697 | 9054797 | NaN | NaN | NaN | 0.0 | NaN | Scheduled |
| 291600955 | 990042 | NaN | NaN | NaN | 0.0 | NaN | Scheduled |

3 rows × 5 columns pd.Dataframe

| | EntityId | TripId | RouteId | StartTime | StartDate |
|---|---|---|---|---|---|
| 0 | 292595792 | 292595792 | 43235 | 21:07:00 | 20230702 |
| 1 | 284276697 | 284276697 | 51652 | 13:10:00 | 20230702 |
| 2 | 291600955 | 291600955 | 14205 | 13:38:00 | 20230702 |

Out[14]:

| | EntityId | TripId | RouteId | StartTime | StartDate |
|---|---|---|---|---|---|
| 0 | 292595792 | 292595792 | 43235 | 21:07:00 | 20230702 |
| 1 | 284276697 | 284276697 | 51652 | 13:10:00 | 20230702 |
| 2 | 291600955 | 291600955 | 14205 | 13:38:00 | 20230702 |

Further cleaning

Out[15]:

| TripId | StopId | StopSequence | ArrivalDelay | ArrivalTime | DepartureDelay | DepartureTime | ScheduleRelationship |
|---|---|---|---|---|---|---|---|
| 292595792 | 9013927 | NaN | NaN | NaN | 0.0 | NaN | Scheduled |
| 284276697 | 9054797 | NaN | NaN | NaN | 0.0 | NaN | Scheduled |
| 291600955 | 990042 | NaN | NaN | NaN | 0.0 | NaN | Scheduled |

# Data cleaning - joins

remove redundant column EntityId

```
# actual data
data_fetch_dates = ['0905', '0706', '0207']
actual_data = pd.concat([pd.merge(
    pd.read_csv(f'actualdata_trips_{date}.csv'), pd.read_csv(f'actualdata_stop_times_{date}.csv'),
    how='left', left_on=['TripId'], right_on='TripId'
) for date in data_fetch_dates], axis=0)
actual_data['Month'] = [int(str(y)[4:6]) for y in actual_data['StartDate']]

# data cleaning: remove redundant column EntityId because it holds sim(EntityId, TripId) = 1
actual_data.drop('EntityId', axis=1, inplace=True)

actual_data.set_index(['TripId', 'StartDate'], inplace=True)
actual_data.head()
```

| TripId | StartDate | RouteId | StartTime | StopId | StopSequence | ArrivalDelay | ArrivalTime | DepartureDelay | DepartureTime |
|---|---|---|---|---|---|---|---|---|---|
| 254163638 | 20230508 | 43975 | 19:44:00 | 9057862 | NaN | NaN | NaN | 0.0 | NaN |
| 282351984 | 20230509 | 43245 | 17:12:00 | 9013478 | NaN | NaN | NaN | 0.0 | NaN |
| 282385362 | 20230508 | 43195 | 20:03:00 | 9058008 | 5.0 | -60.0 | NaN | 0.0 | NaN |
| 264505093 | 20230508 | 59935 | 19:12:00 | 9050640 | 11.0 | 30.0 | NaN | 30.0 | NaN |
| 282384857 | 20230509 | 43216 | 04:37:00 | 9049320 | NaN | NaN | NaN | 0.0 | NaN |

In [179…

```
# overall delay
arrival_delays = [x for x in actual_data['ArrivalDelay'] if not (pd.isna(x) or x == 0)]
departure_delays = [x for x in actual_data['DepartureDelay'] if not (pd.isna(x) or x == 0)]
ad = len(arrival_delays) / actual_data.shape[0] * 100
dd = len(departure_delays) / actual_data.shape[0] * 100
print('arrival delay in {x} % of rides'.format(x=round(ad, 2)))
print('departure delay in {x} % of rides'.format(x=round(dd, 2)))
print('arrival delay: {x}'.format(x=round(np.mean(arrival_delays), 2)))
print('departure delay: {x}'.format(x=round(np.mean(departure_delays), 2)))
```

```
arrival delay in 15.6 % of rides
departure delay in 17.99 % of rides
arrival delay: 110.68
departure delay: 134.61
```

join on targetData

```
# target data
dir_target = 'C:\\Users\\nstan\\OneDrive\\Winfo\\SoSe_23\\Data Integration\\target_data_vbn_modified'
rnames = ['trips', 'stops', 'stop_times', 'agency', 'routes', 'transfers']
target_data = {rname: pd.read_csv(f'{dir_target}\\{rname}.csv', low_memory=False) for rname in rnames}
```

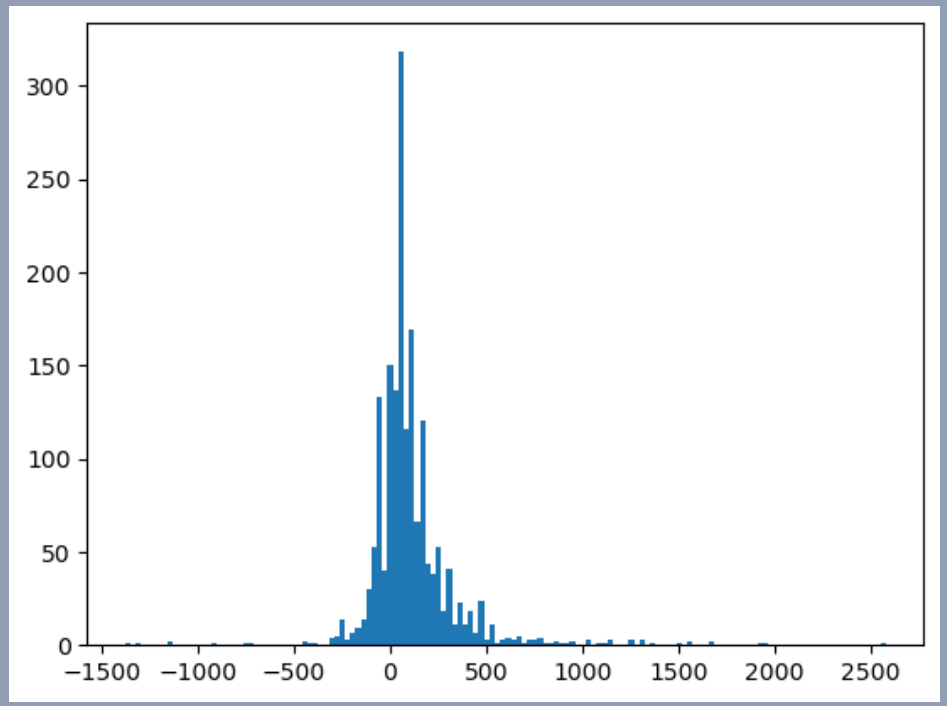
```
# perform joins on target data relations
# step 1: join routes and agency (= r1)
r1 = pd.merge(target_data['routes'], target_data['agency'], how='left', left_on=['agency_id'], right_on=['agency_
r1.head(3)
```

| | route_id | agency_id | route_short_name | route_type | agency_name |
|---|---|---|---|---|---|
| 0 | 71026 | 1060 | SEV24 | 1 | S-Bahn Hamburg |
| 1 | 71025 | 1060 | SEV10 | 1 | S-Bahn Hamburg |
| 2 | 70978 | 1060 | SEV21 | 2 | S-Bahn Hamburg |

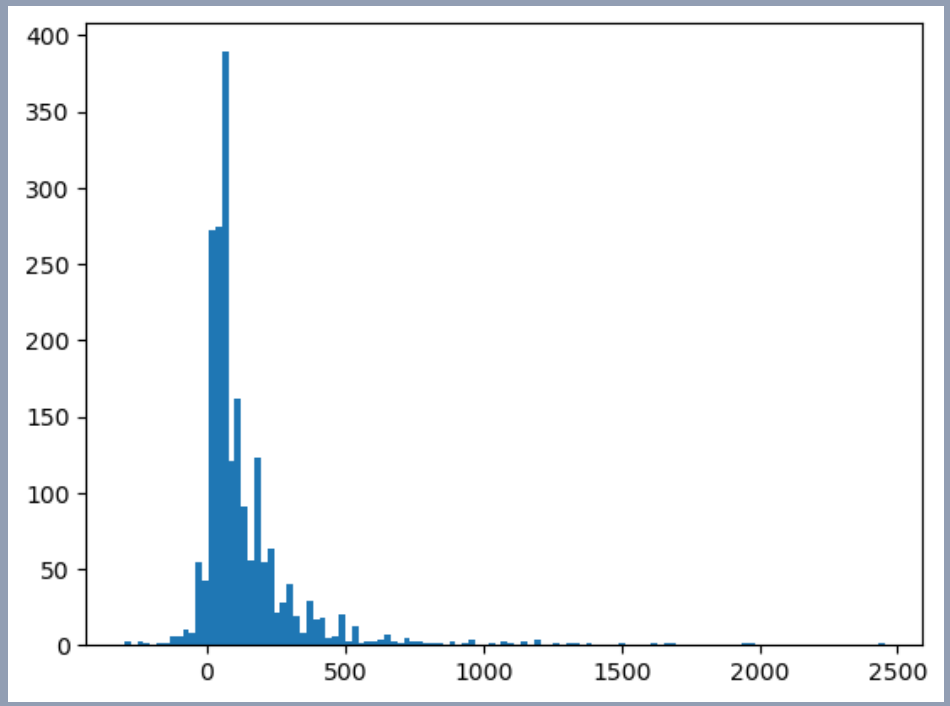
# Delays

## Arrival Delays

| Month | ArrivalDelay | DepartureDelay |
|---|---|---|
| 5 | 55.197930 | 20.106013 |
| 6 | 83.177808 | 34.854118 |
| 7 | 36.043728 | 12.414034 |

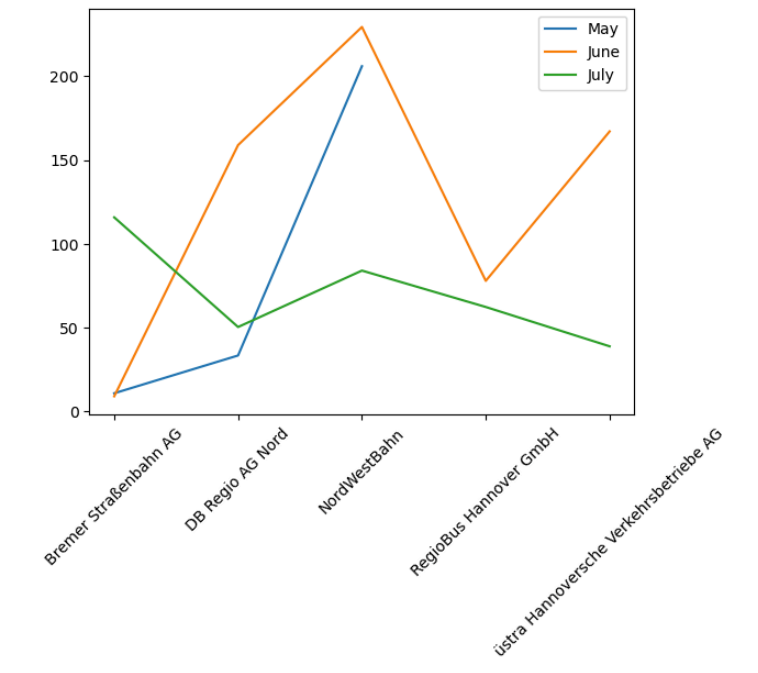



## Departure Delays



# Delays

## Arrival Delays

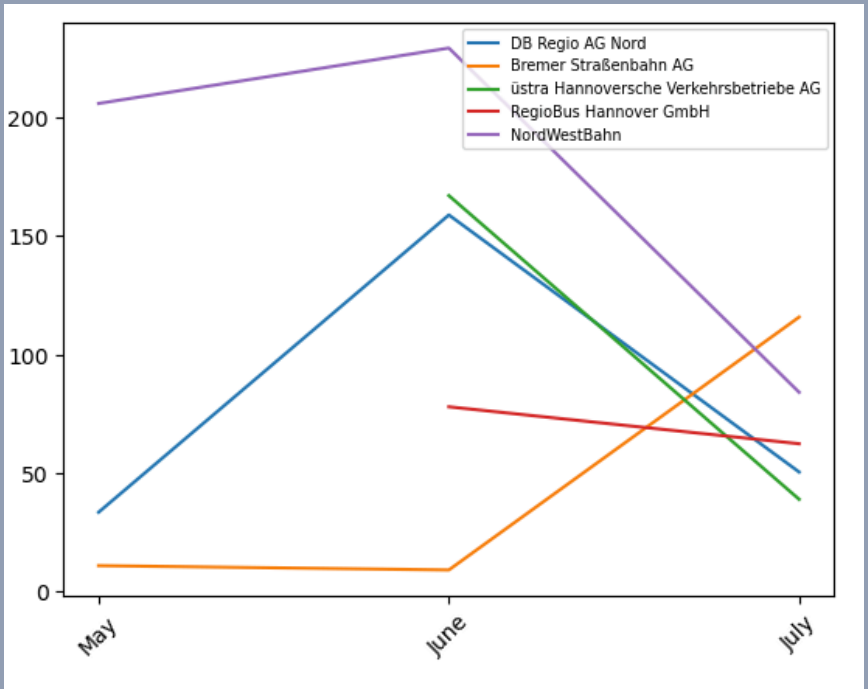



Out[188...

| agency_name | Month | ArrivalDelay | DepartureDelay |
|---|---|---|---|
| Bremer Straßenbahn AG | 5.0 | 10.772727 | 21.051724 |
| | 6.0 | 8.956989 | 25.369565 |
| | 7.0 | 115.780000 | 100.407407 |
| DB Regio AG Nord | 5.0 | 33.333333 | 100.800000 |
| | 6.0 | 158.918919 | 236.842105 |
| | 7.0 | 50.322581 | 124.285714 |
| NordWestBahn | 5.0 | 206.000000 | 592.500000 |
| | 6.0 | 229.411765 | 378.000000 |
| | 7.0 | 84.000000 | 250.000000 |
| RegioBus Hannover GmbH | 6.0 | 77.875000 | 146.378378 |
| | 7.0 | 62.272727 | 114.172414 |
| üstra Hannoversche Verkehrsbetriebe AG | 6.0 | 167.090909 | 117.255639 |
| | 7.0 | 38.800000 | 84.648148 |

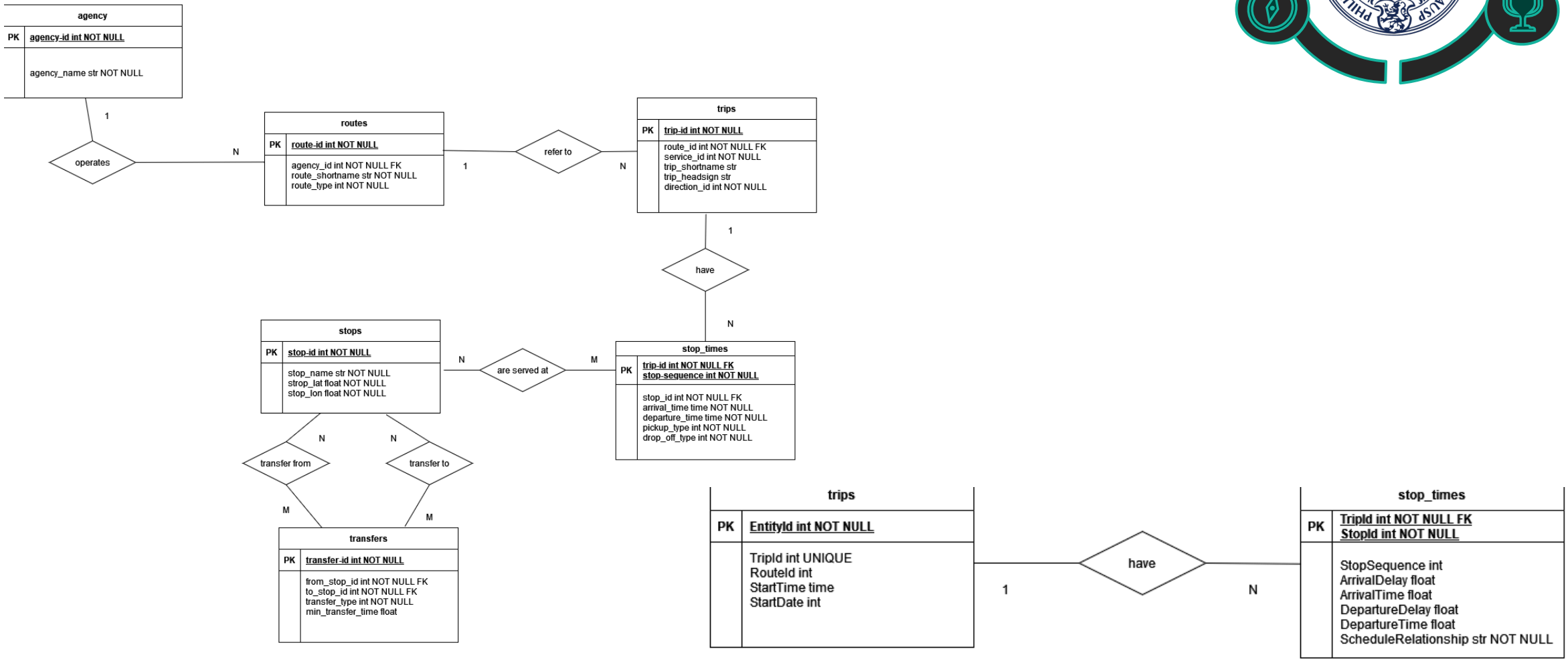
# Delays

## Arrival Delays




| agency_name | Month | ArrivalDelay | DepartureDelay |
|---|---|---|---|
| Bremer Straßenbahn AG | 5.0 | 10.772727 | 21.051724 |
| | 6.0 | 8.956989 | 25.369565 |
| | 7.0 | 115.780000 | 100.407407 |
| DB Regio AG Nord | 5.0 | 33.333333 | 100.800000 |
| | 6.0 | 158.918919 | 236.842105 |
| | 7.0 | 50.322581 | 124.285714 |
| NordWestBahn | 5.0 | 206.000000 | 592.500000 |
| | 6.0 | 229.411765 | 378.000000 |
| | 7.0 | 84.000000 | 250.000000 |
| RegioBus Hannover GmbH | 6.0 | 77.875000 | 146.378378 |
| | 7.0 | 62.272727 | 114.172414 |
| üstra Hannoversche Verkehrsbetriebe AG | 6.0 | 167.090909 | 117.255639 |
| | 7.0 | 38.800000 | 84.648148 |

# Final status on ER model



# Wrap up:


**Agency that needs the most attention is the NordWestBahn.**

**In June the NordWestBahn had their peak of delays.**