



3. Zwischenpräsentation PG INFRASENSE

Michael Birke, Florian Dyck, Alexander Tesch, Malte Kuhlmann, Mukhran Kamashidze, Malte Schott, Richard Schulte

30.06.2023

Agenda

1 Projektplan & Ziele

2 Arbeitsgruppe „Meldeplattform“

3 Arbeitsgruppe „Data“

4 Arbeitsgruppe „Dashboard“

5 Diskussion



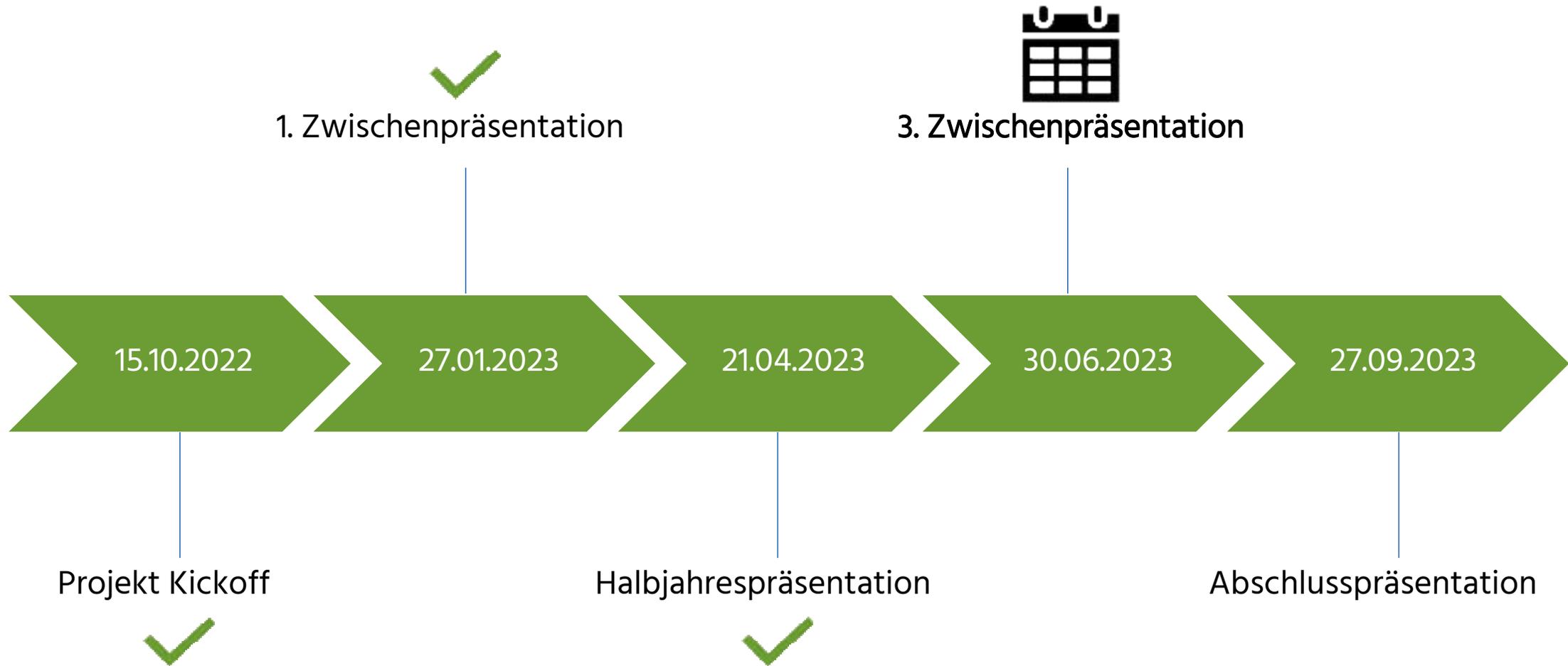
PG INFRAsense

Projektziele

- Ziel 1: Dashboard zur Unterstützung der Radverkehrsplanung
- Ziel 2: Recherche und Implementierung geeigneter KI-Methoden zur Auswertung von Bilddaten zur Fahrradinfrastruktur
 - Meldeplattform für Radwegschäden



Projektverlauf



Meldeplattform

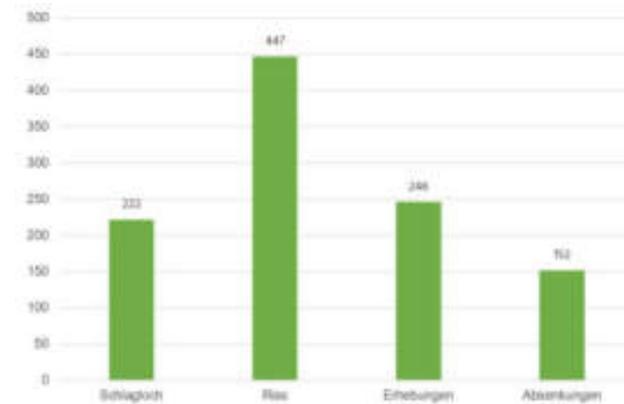
Malte Kuhlmann, Malte Schott

Rückblick



- Klassifikation von Riss, Schlaglöcher, Absenkungen und Erhebungen

- Eigener Datensatz mit über 1500 Schäden



Feedback

Probleme mit Absenkung und Erhebung



- Problem: Absenkung und Erhebung schwer zu erkennen
- Andere Projekte sind besser geeignet
 - Z.B. aZur Projekt
- Resultat: Fokus auf Riss und Schlagloch

Feedback

Segment Anything

- Segment Anything (Image Segmentation)
 - Rechenpower intensiv
 - Nicht zielführend



<https://segment-anything.com/demo>

Feedback

Data Augmentation

- Erweiterung des Datensatzes durch Drehen / Spiegeln



- ist bereits in den bisherigen Modellen integriert

Modelling

Algorithmus

Semantische Segmentierung



Straße, Schlagloch, Schild

Keine Objekte, nur Pixel

Klassifikation



Schlagloch

1 Objekt

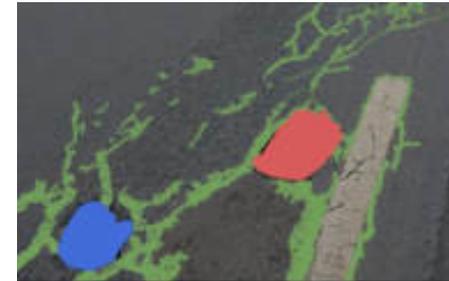
Objekterkennung



Schlagloch, Schlagloch, Riss

mehrere Objekte

Instanz Segmentierung



Schlagloch, Schlagloch, Riss

Ergebnis:

1. Objekterkennung mit YOLOv8
2. Klassifikation mit efficientnet v2



Modelling

...angepasst und optimiert

1.



Object Detection
mit YoloV8
als „damage“

2.



Erkannter „damage“ wird
ausgeschnitten

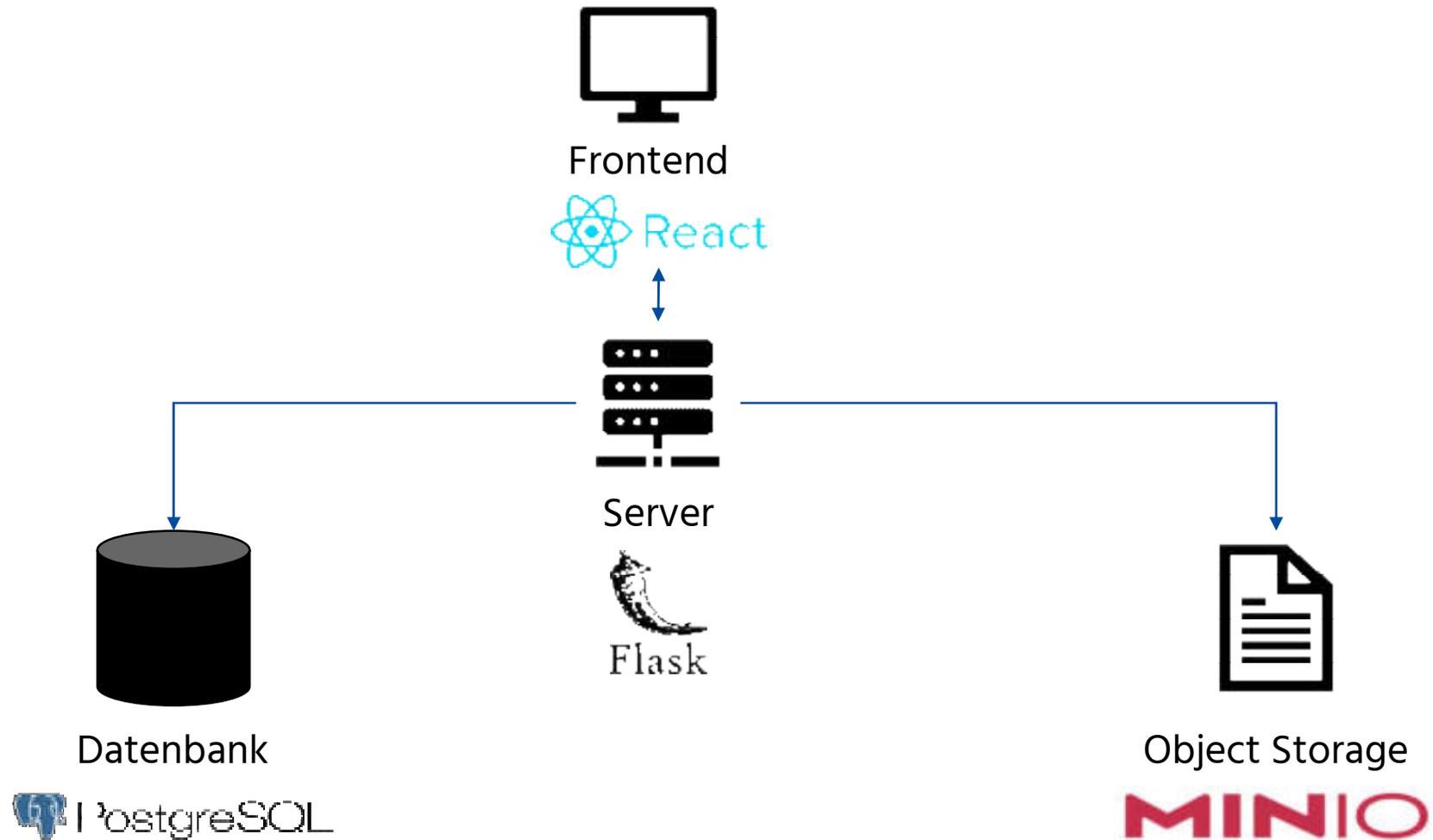
3.



Classification mit
efficientnetv2 auf „damage“

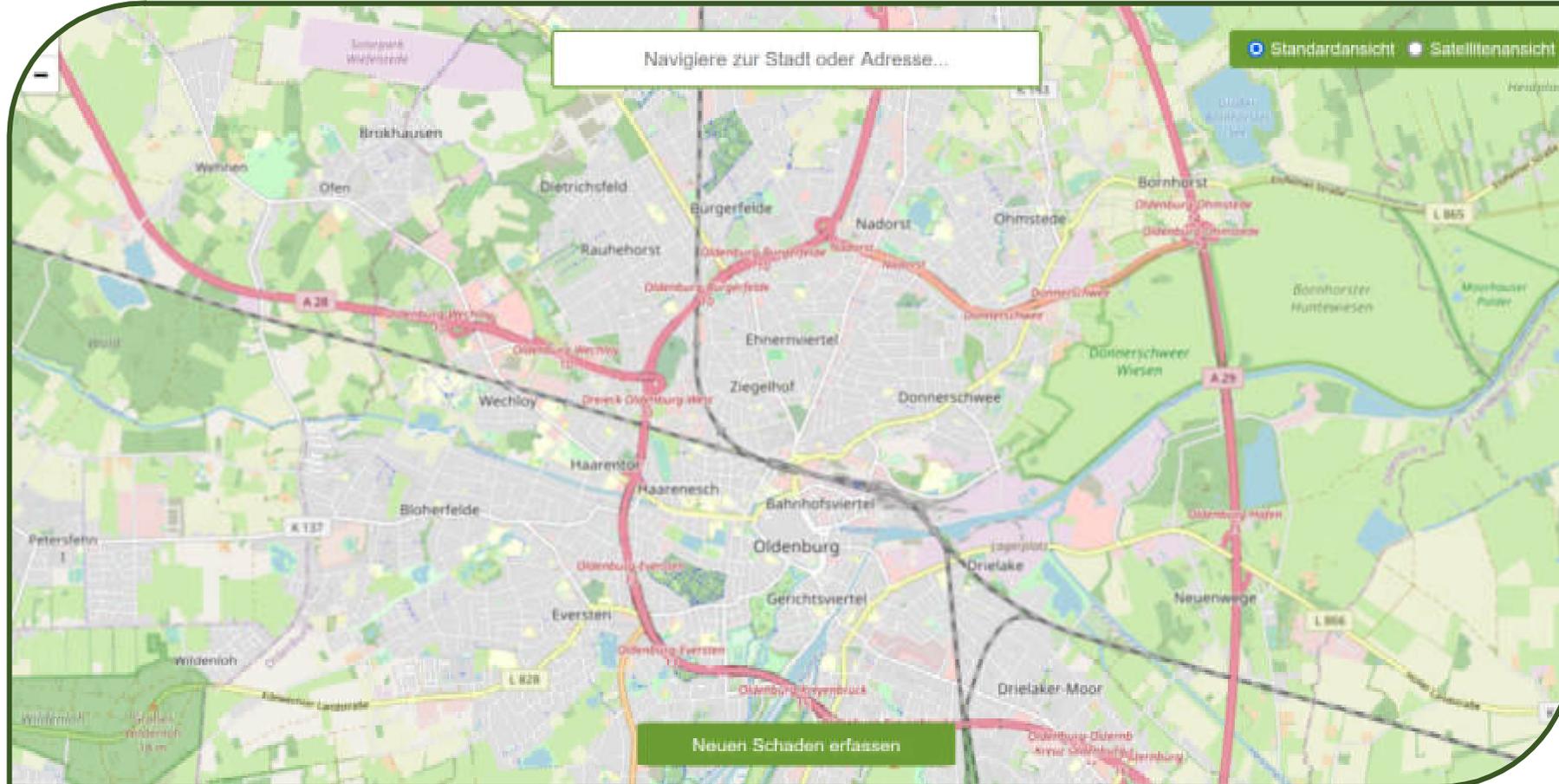
- Unterteilung in
Riss und Schlagloch

Systemarchitektur



Meldeplattform

Live Demo



Ausblick

Frontend

- Visualisierung und Einbindung der gemeldeten Schäden auf der Karte
- Optimierung der User Experience
- Automatische Standortverarbeitung

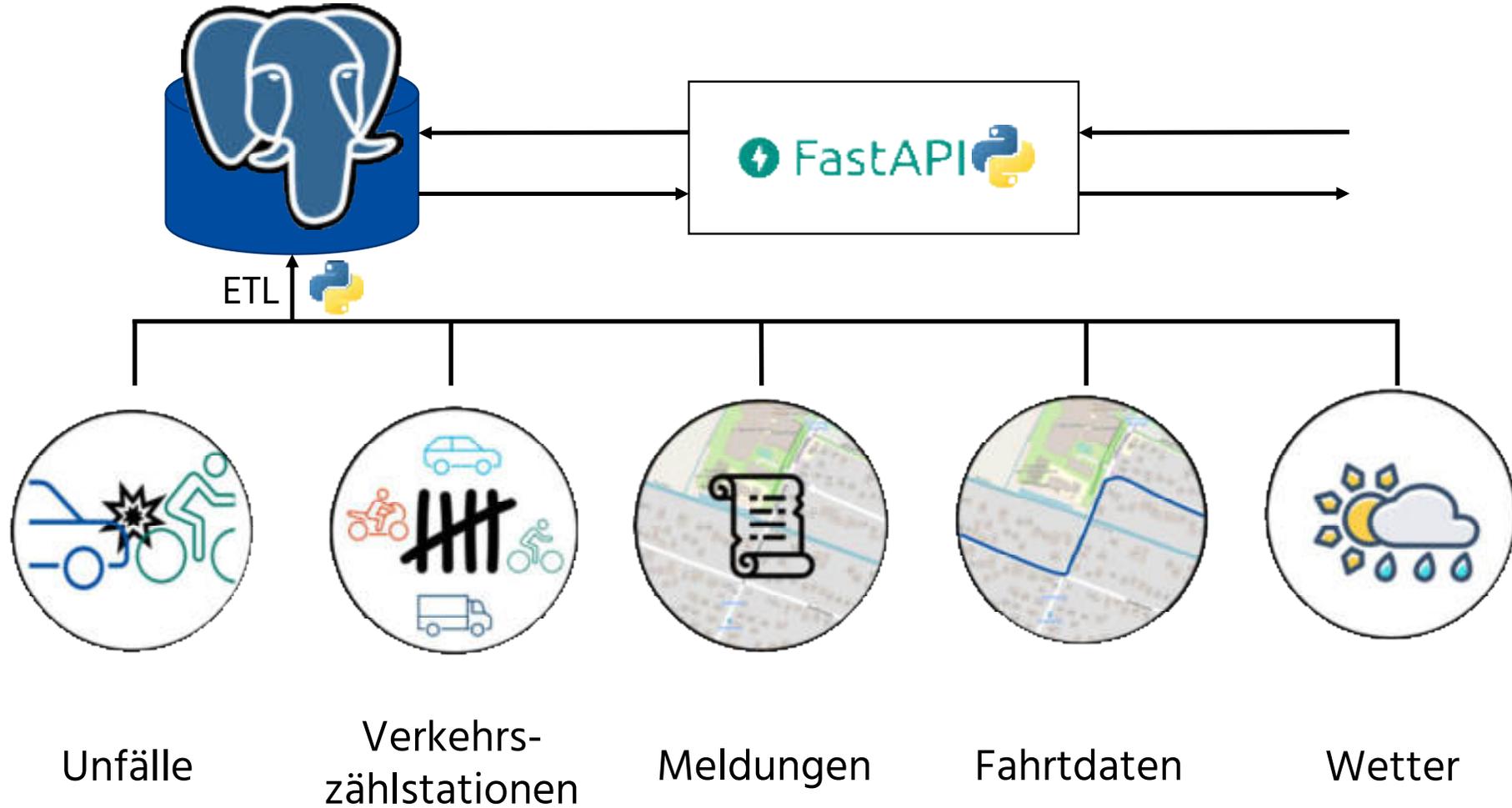
Backend

- Deployment – Verbesserungen
- Verbesserung des Datensatzes / 4. Iteration
- Mail - Info – Service

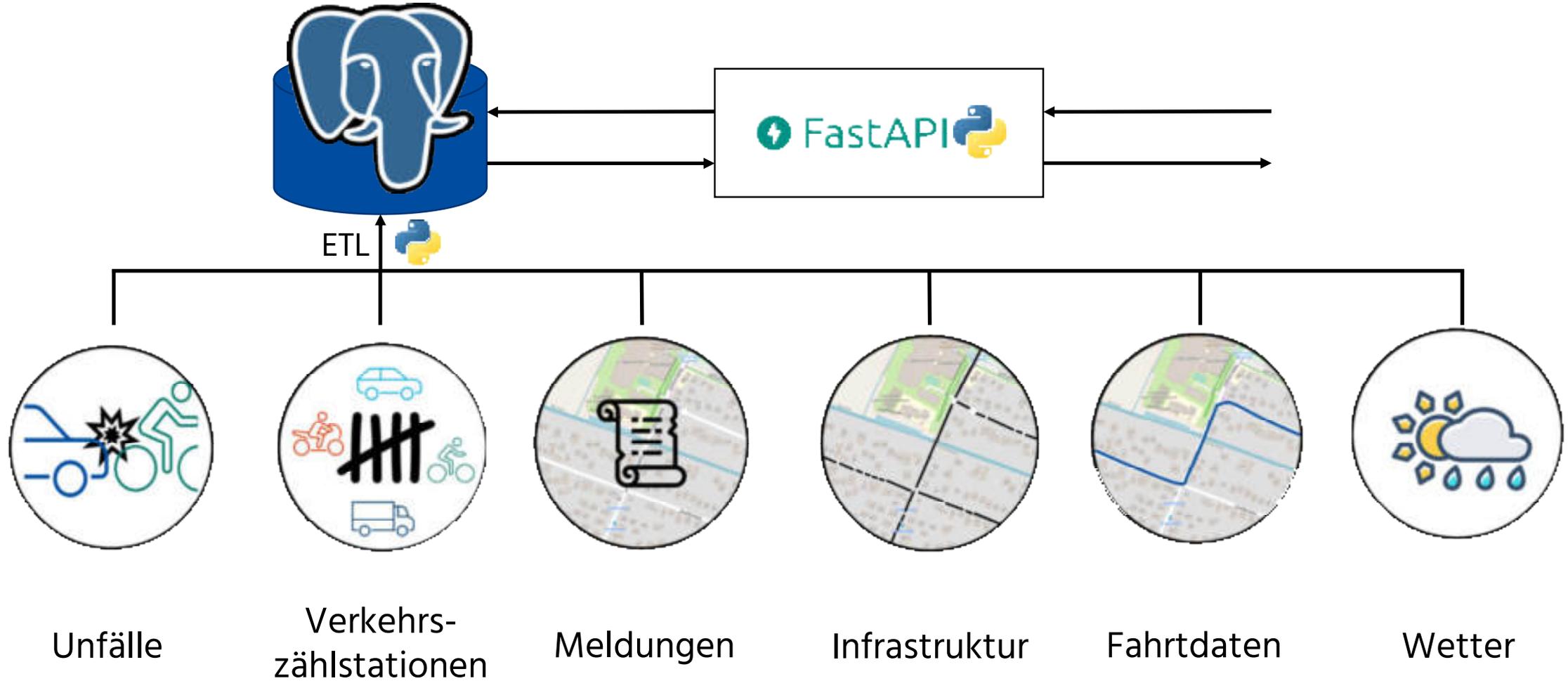
Arbeitsgruppe Data

Florian Dyck, Richard Schulte

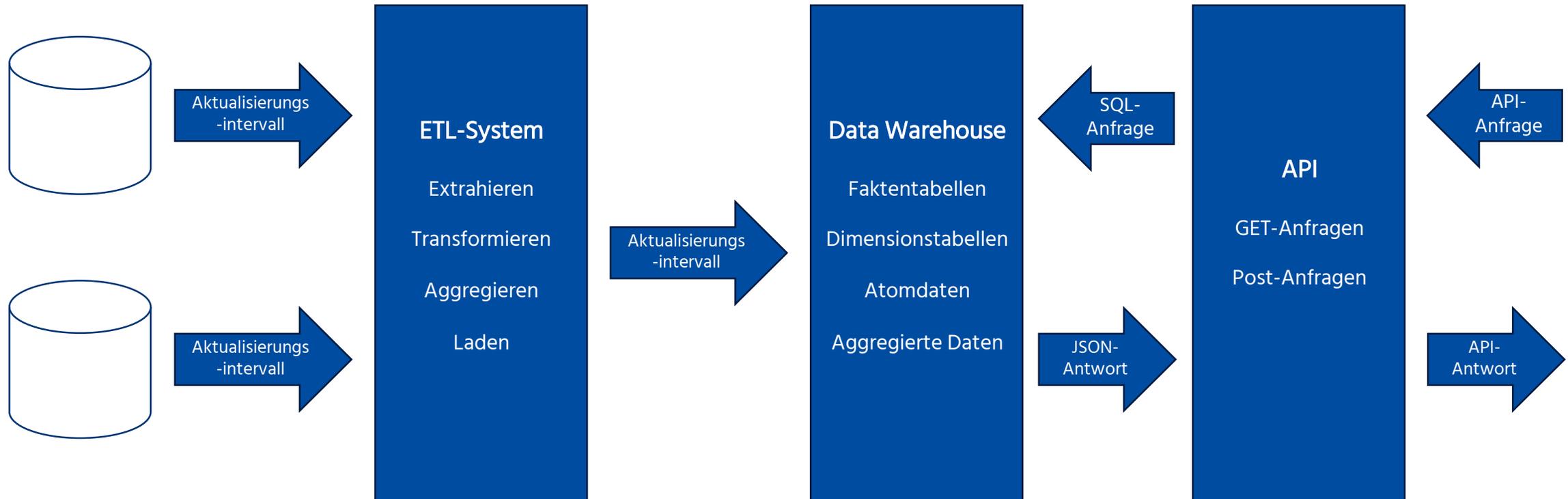
Datenanbindung - Rückblick



Datenanbindung



Data Warehouse Konzept



Anpassung der API

Dynamischer Join

Unfall #1	Fahrrad
	Auto

Dynamischer Join

Unfall #1

Unfall #1	Fahrrad
Unfall #1	Auto

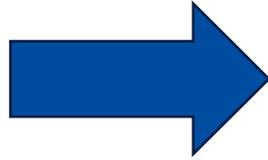


Vollständige Bezeichnungen



Unfalltypen

„AB“

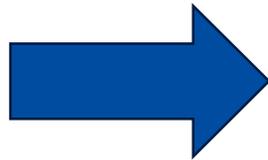


„Abbiege-Unfall“



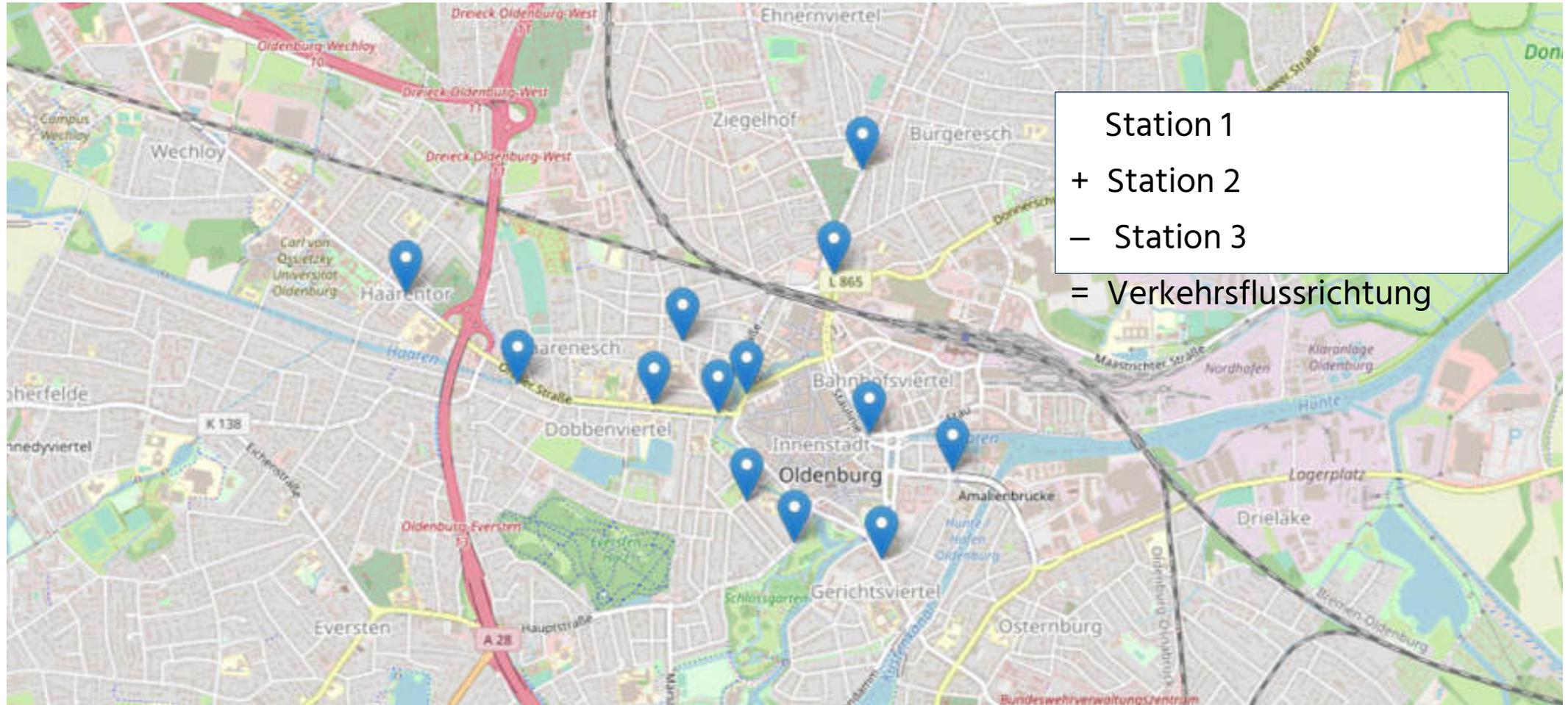
Zeitschlüssel

0

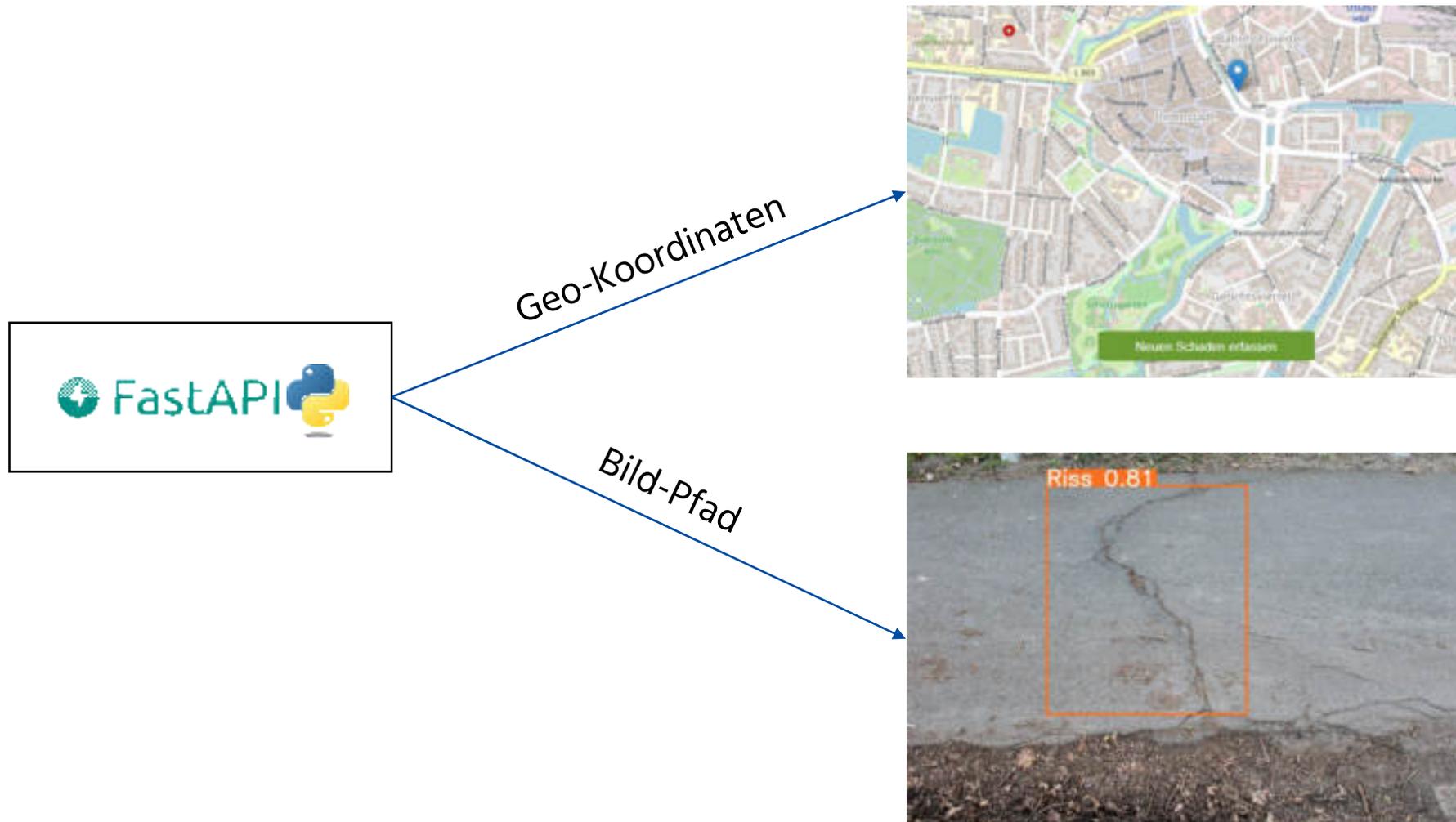


„Sonntag“

Addition und Subtraktion von Kennzahlen



Endpunkt Schadensbilder

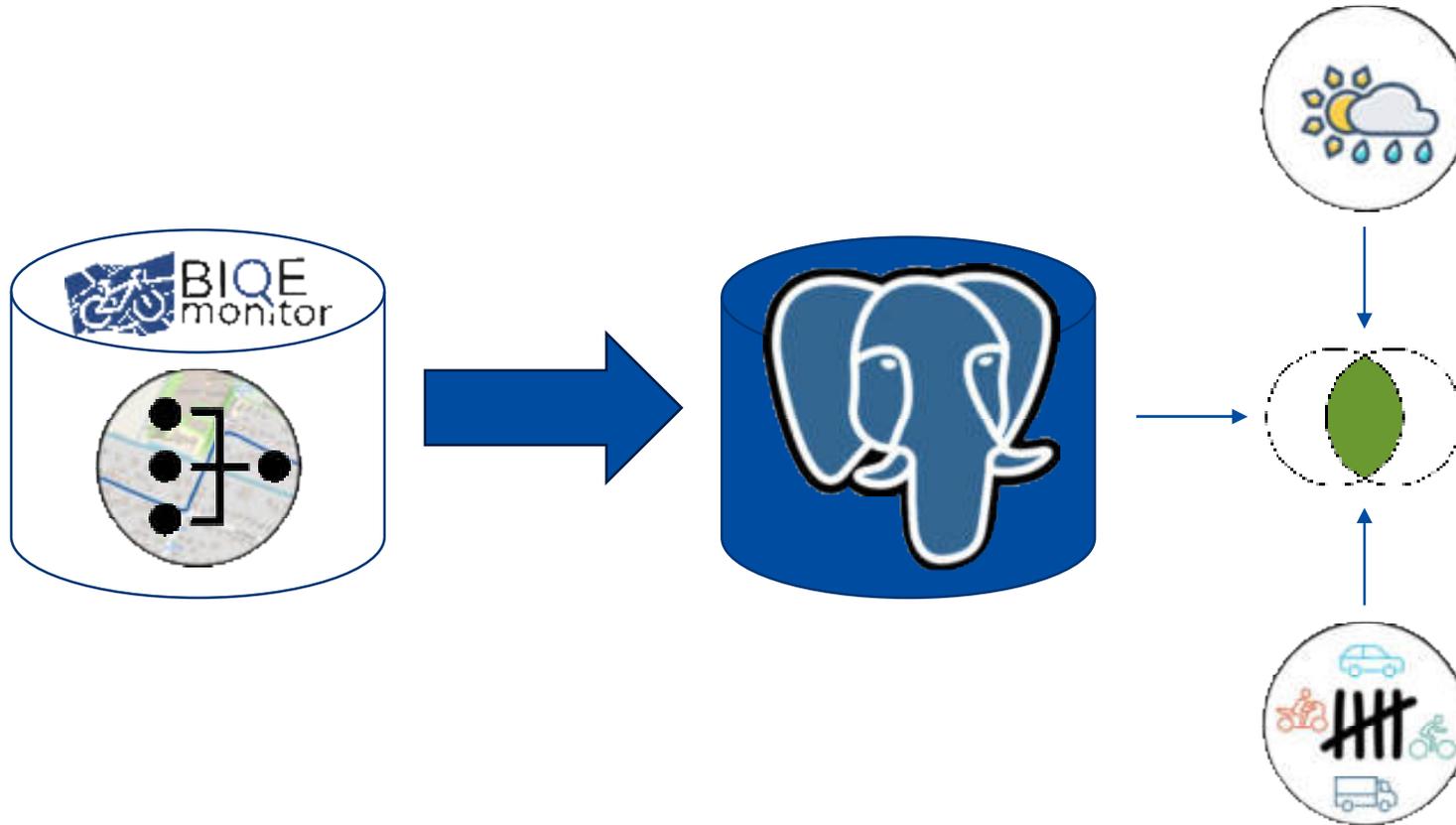


BIQEmonitor KPIs

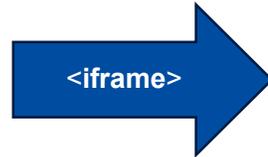
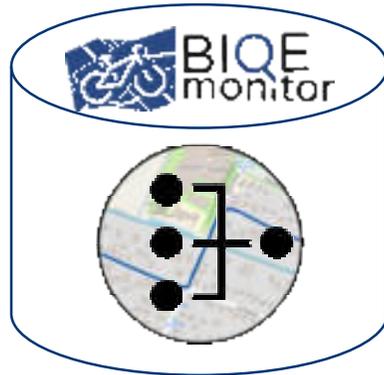
www.biqemonitor.de



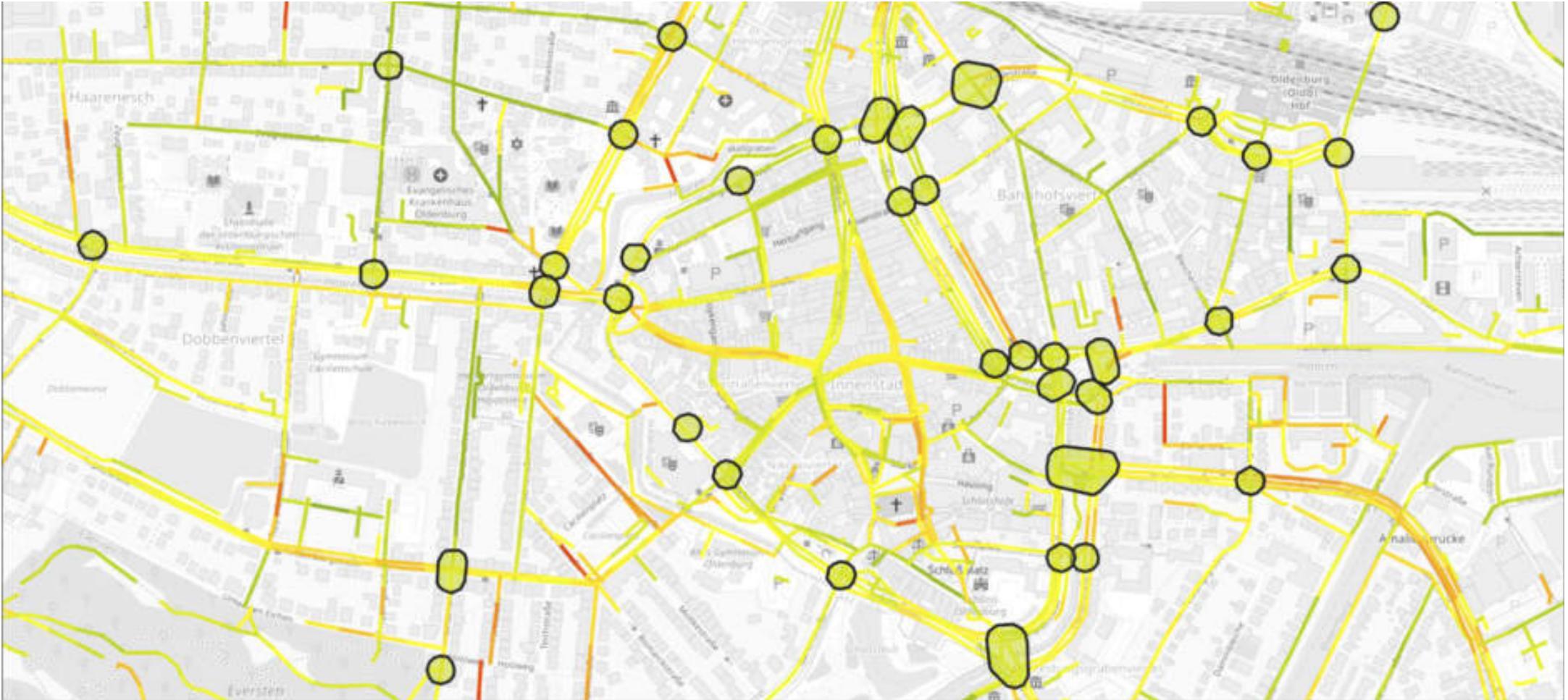
Fahrt-/Sensordaten



Fahrt-/Sensordaten



BIQEmonitor - Straßenabschnitte



BIQEmonitor - Heatmap



BIQEmonitor - Ansichten

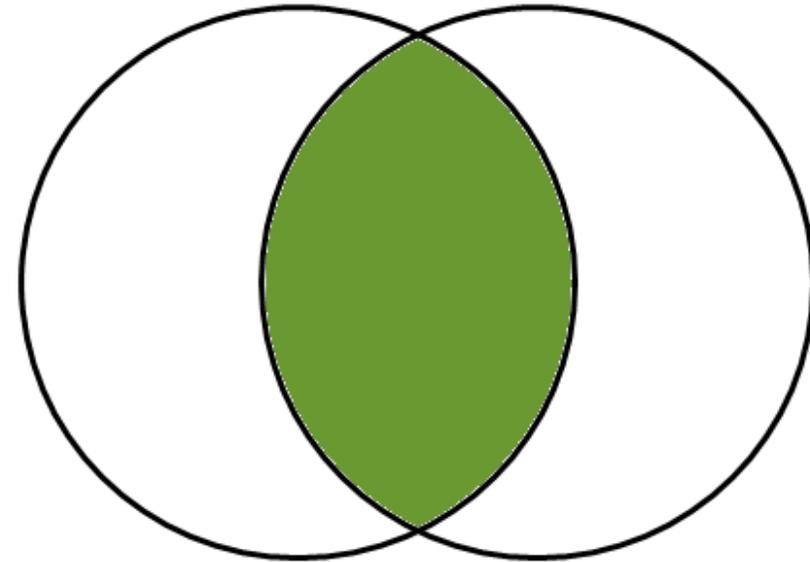
- Knotenpunkte
- Anomaliegrad
- Anomaliegrad (Heatmap)
- Anomaliegrad v2 (Heatmap)
- Anzahl Durchfahrten
- Erschütterungsgrad
- Erschütterungsgrad (Heatmap)
- Erschütterungsgrad v2
- Erschütterungsgrad v2 (Heatmap)
- Geschwindigkeit (absolut)
- Geschwindigkeit (normiert)
- Standzeit
- Vollbremsungen
- Zeitverlust



Ausblick



BIQEmonitor-Integration



KPI-Verknüpfung

Dashboard

Michael Birke, Alexander Tesch, Mukhran Kamashidze

KPI-Realisierung

Oberkategorien

- Fahrrad-Verkehrsaufkommen
- Unfälle
- Bürgermeldungen



KPI-Realisierung

Fahrrad-Verkehrsaufkommen

- KPI 1: Auswahl einzelner Zählstellen

- KPI2: Zählstellenvergleich

KPI-Realisierung

Bürgermeldungen

- KPI 3: Zeitverteilung der Meldungen
- KPI 4: Verteilung Meldungstypen
- KPI 5: Erkannte Probleme
- KPI 6: Dauer bis Meldung behoben ist



KPI-Realisierung

Unfälle

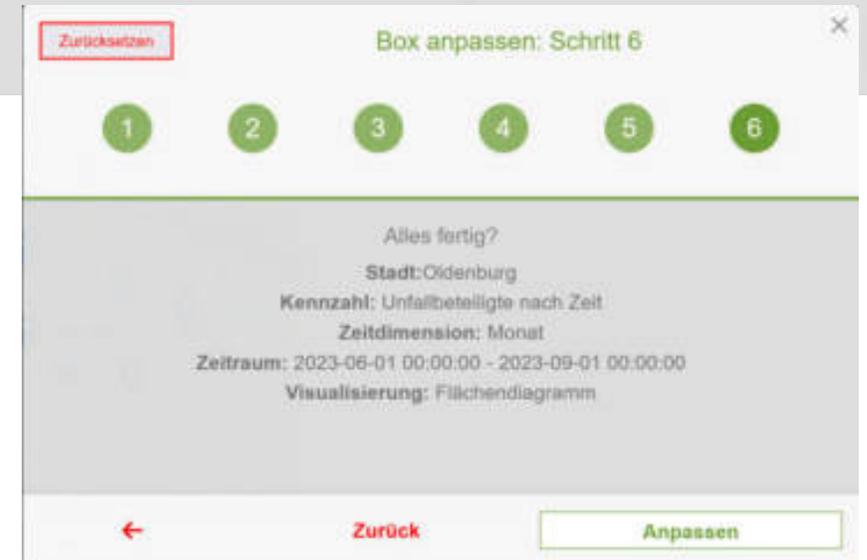
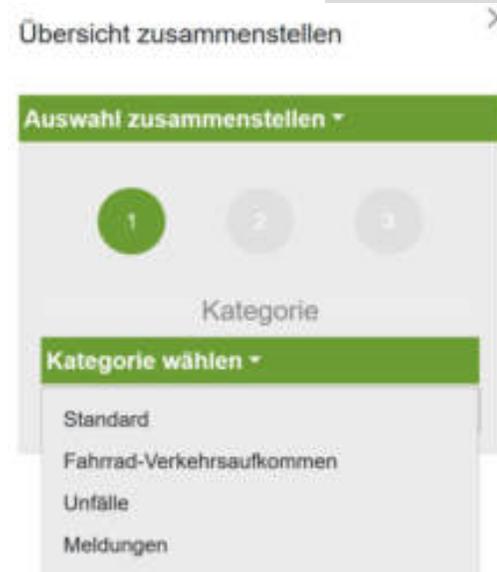
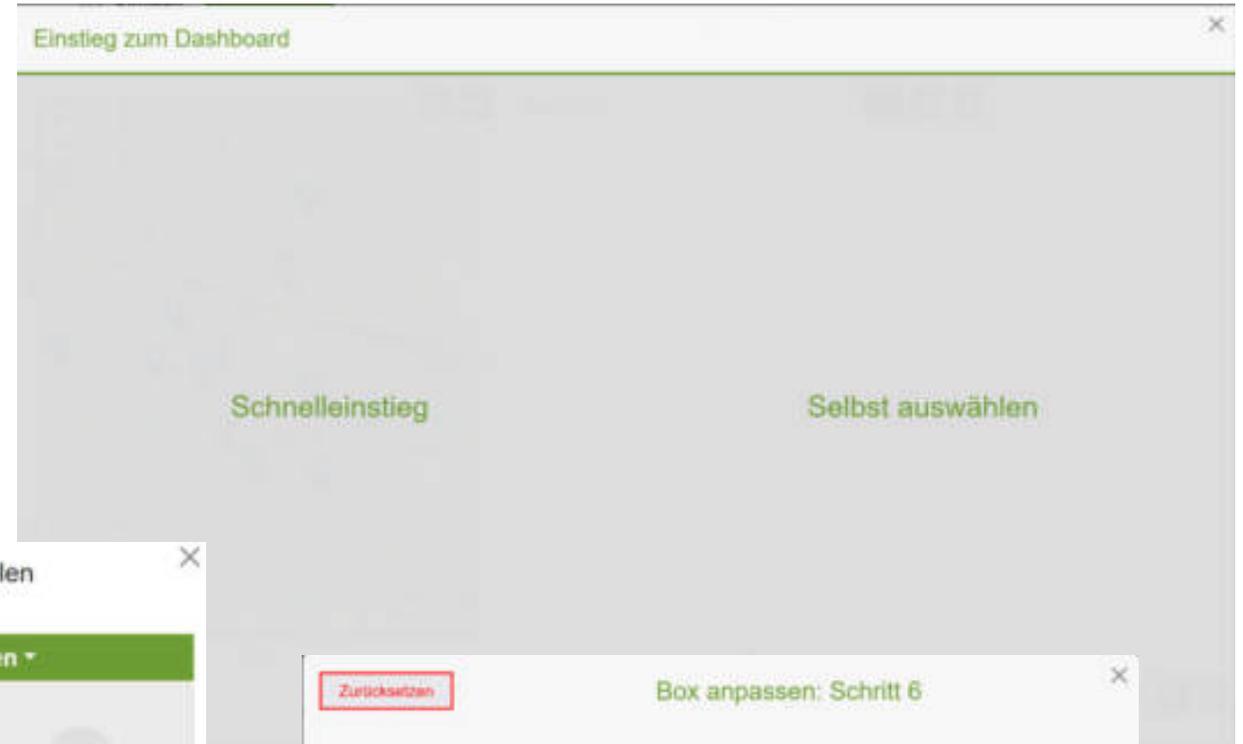
- KPI 7: Zeitverteilung der Unfälle
- KPI 8: Verteilung Unfalltypen
- KPI 9: Unfälle nach Verursacher
- KPI 10: Beteiligte nach Zeit
- KPI 11: Unfallbeteiligte nach Unfalltyp

Feedback

Optimierte Nutzerführung

Verschiedene Optionen zur Erstellung eines Dashboards geschaffen

- Schnelleinstieg
- Selbst auswählen



Login-Konzept

Prototyp

- ExpertInnen sollen sich einloggen können
- Möglichkeit Dashboards / Views speichern zu können

Anmelden

Email

Kennwort

ANMELDEN



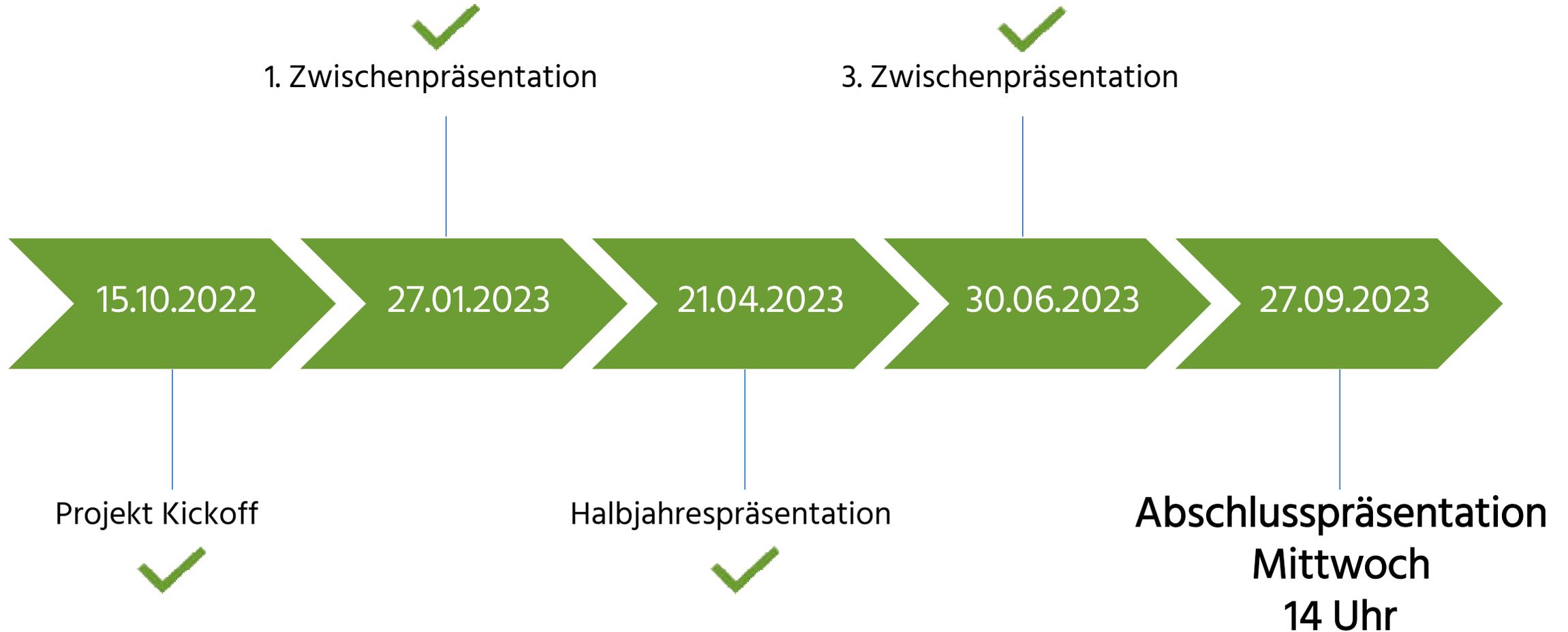


Ausblick

- Weitere KPIs mittels API-Anbindung realisieren
 - Zum Beispiel Länger der Fahrten je nach Wetter (insb. Regen, Wind)
 - Straßenranking/-score
 - Weitere
- Einbindung von Karten-Inhalten
- Weiteres Optimieren der Nutzer-Führung
- Ausbau des Rollenkonzepts und der Login-Funktion



SAVE THE DATE!



Vielen Dank für Ihre Aufmerksamkeit!

Gibt es Fragen?

Kontakt

pg@radweg-radar.de



Quellenverzeichnis

Literatur

- GE, Zheng, et al. Yolox: Exceeding yolo series in 2021. *arXiv preprint arXiv:2107.08430*, 2021.
- Jocher, G. (2020). YOLOv5 by Ultralytics (Version 7.0) [Computer software]. <https://doi.org/10.5281/zenodo.3908559>
- Tan, Mingxing, and Quoc Le. "Efficientnet: Rethinking model scaling for convolutional neural networks." *International conference on machine learning*. PMLR, 2019.
- Wirth, Rüdiger, and Jochen Hipp. "CRISP-DM: Towards a standard process model for data mining." *Proceedings of the 4th international conference on the practical applications of knowledge discovery and data mining*. Vol. 1. 2000.

Quellenverzeichnis

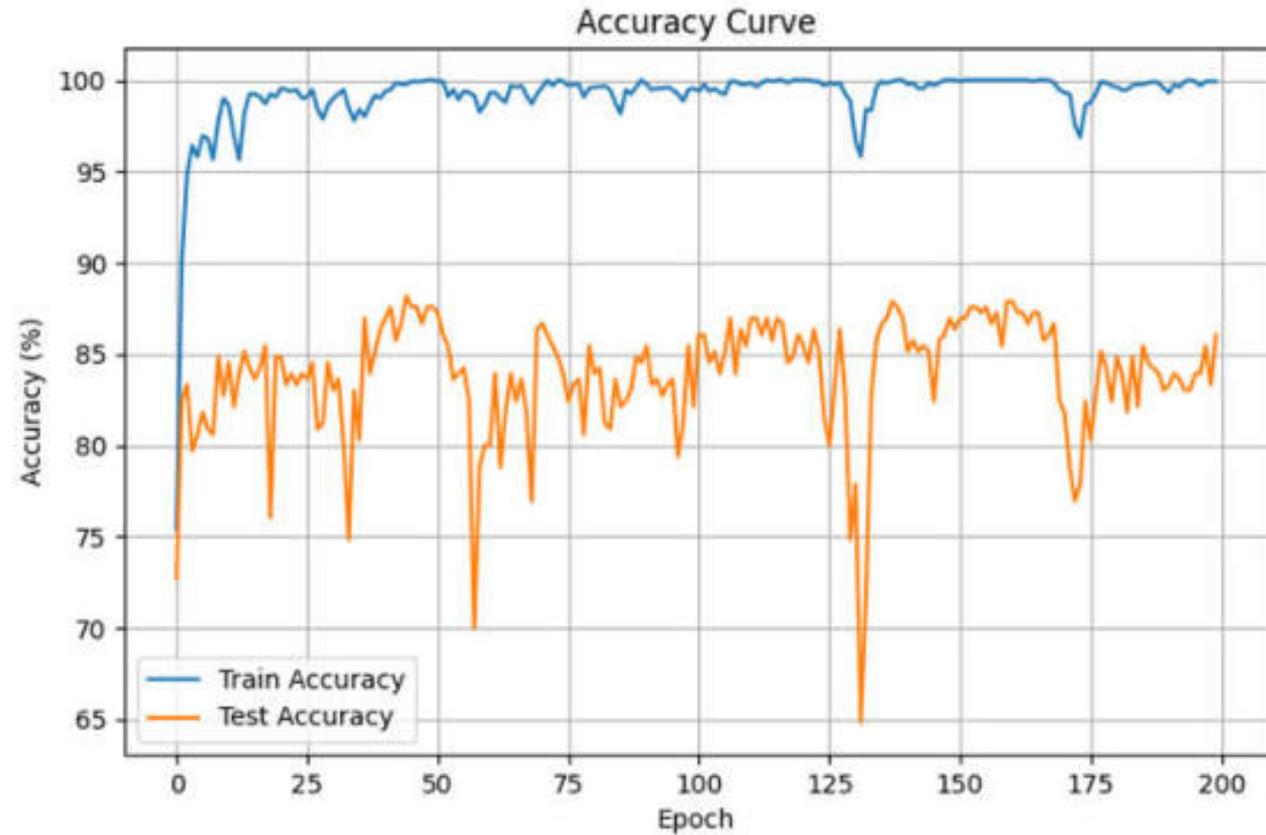
Bilder

- https://de.wikipedia.org/wiki/PostgreSQL#/media/Datei:Postgresql_elephant.svg
- <https://www.openstreetmap.org/#map=17/53.14593/8.18101>
- <https://pixabay.com/vectors/sun-cloud-rain-icon-summer-sky-3000986/>
- <https://fastapi.tiangolo.com>
- <https://www.python.org/community/logos/>
- https://de.wikipedia.org/wiki/JavaScript_Object_Notation
- http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf
- <https://github.com/heartexlabs/label-studio>



Metriken

Accuracy of Yolo and EfficientNet



Review & Feedback

- Absenkung & Erhebung schwierig mit Computer Vision – Modelle lösbar
- Andere Ansätze aus ähnlichem Projekt vielversprechender
 - Auf dieser Basis haben wir uns fortan konzentriert auf Schlagloch & Riss
- Weiterer Ansatz binäre Klassifikation vorgestellt und umgesetzt
 - D.h. Rauswurf der nicht mehr berücksichtigten Klassen und Training mit einer „damage“-Klasse
 - Im Anschluss versuch mittels Efficinet automatisiert die Aufteilung in Schlagloch & Riss zu machen
- Object Augmentation automatisch in YOLO inkludiert
- Segment Anything geprüft und verworfen, da keine Rechenpower und nicht hilfreich für unser Problem