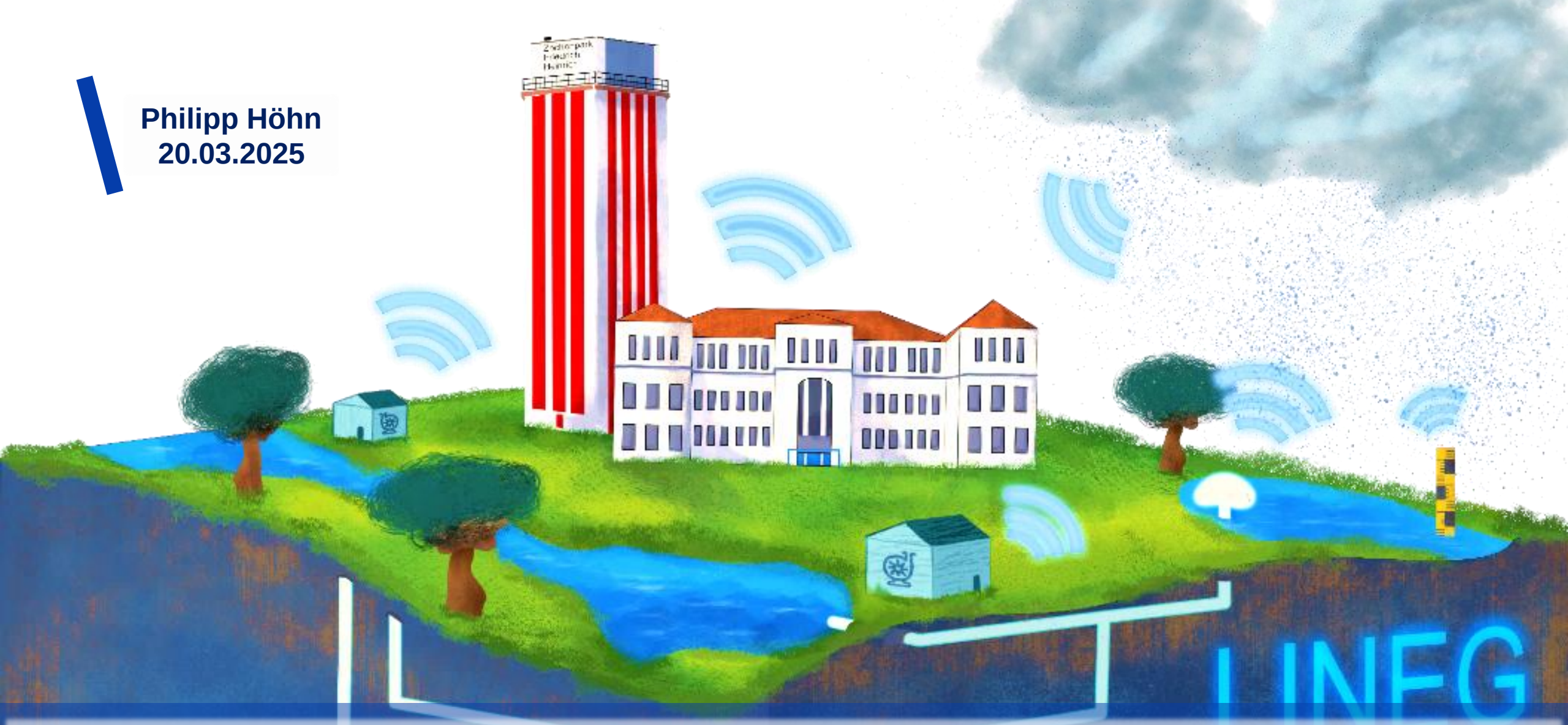
Philipp Höhn
20.03.2025

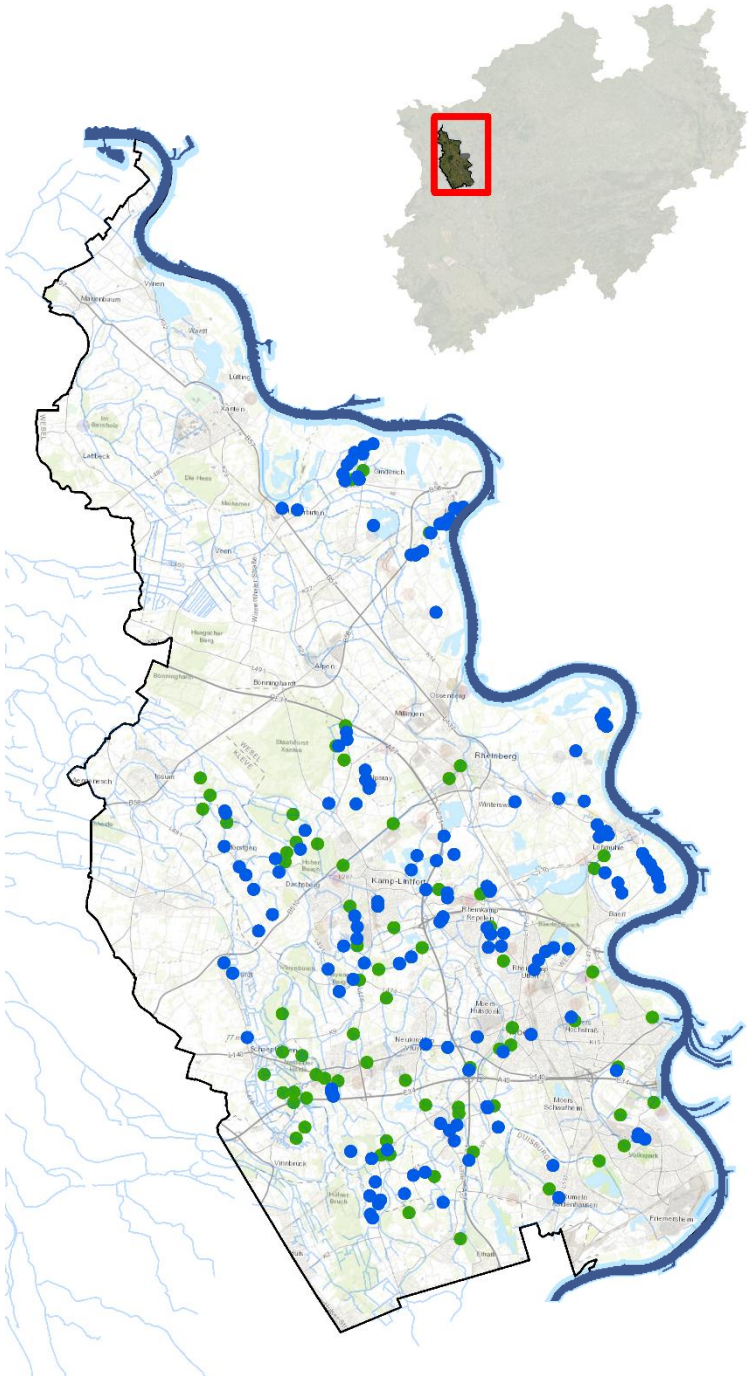


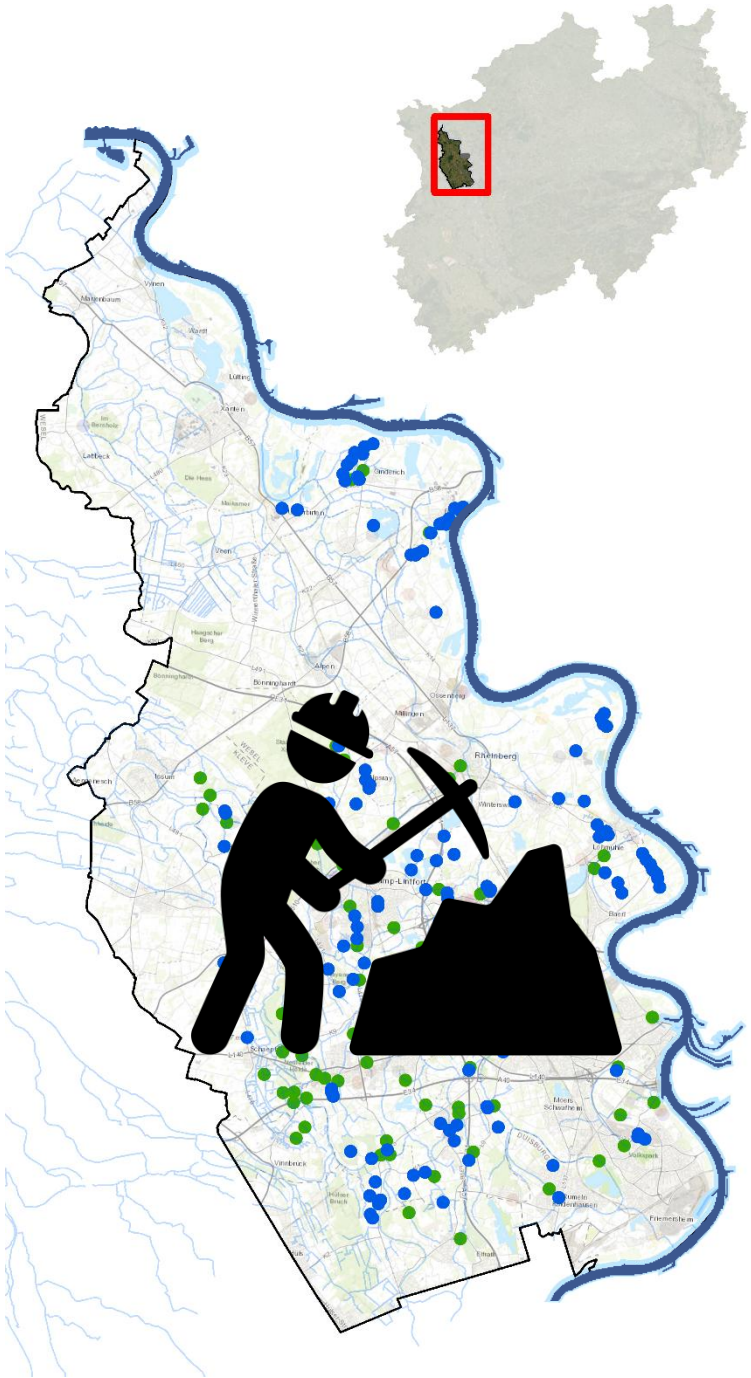
Digitale Gewässerwirtschaft im Bergsenkungsgebiet:

von Betriebsunterstützung und Potentialen vorausschauender Bewirtschaftung

Aufgaben

Wir verbinden Gewässer und Naturschutz.

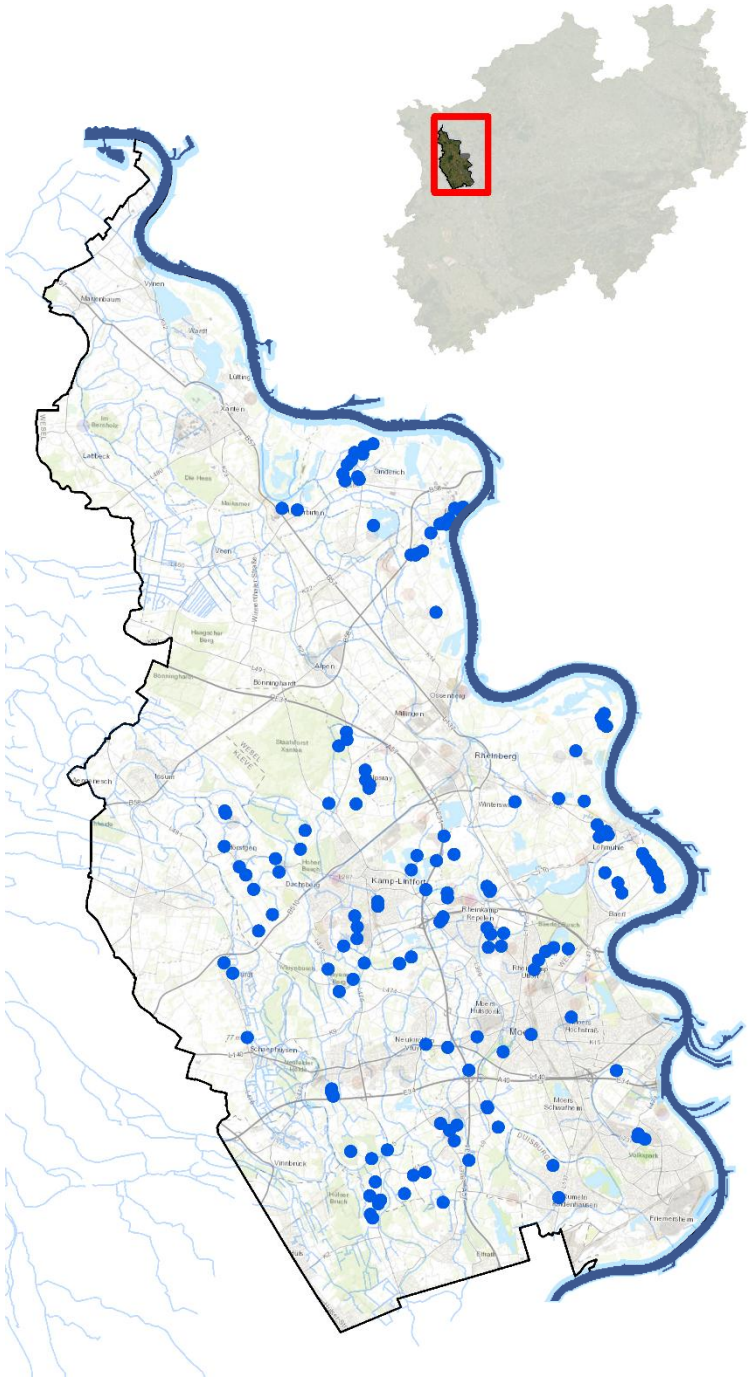




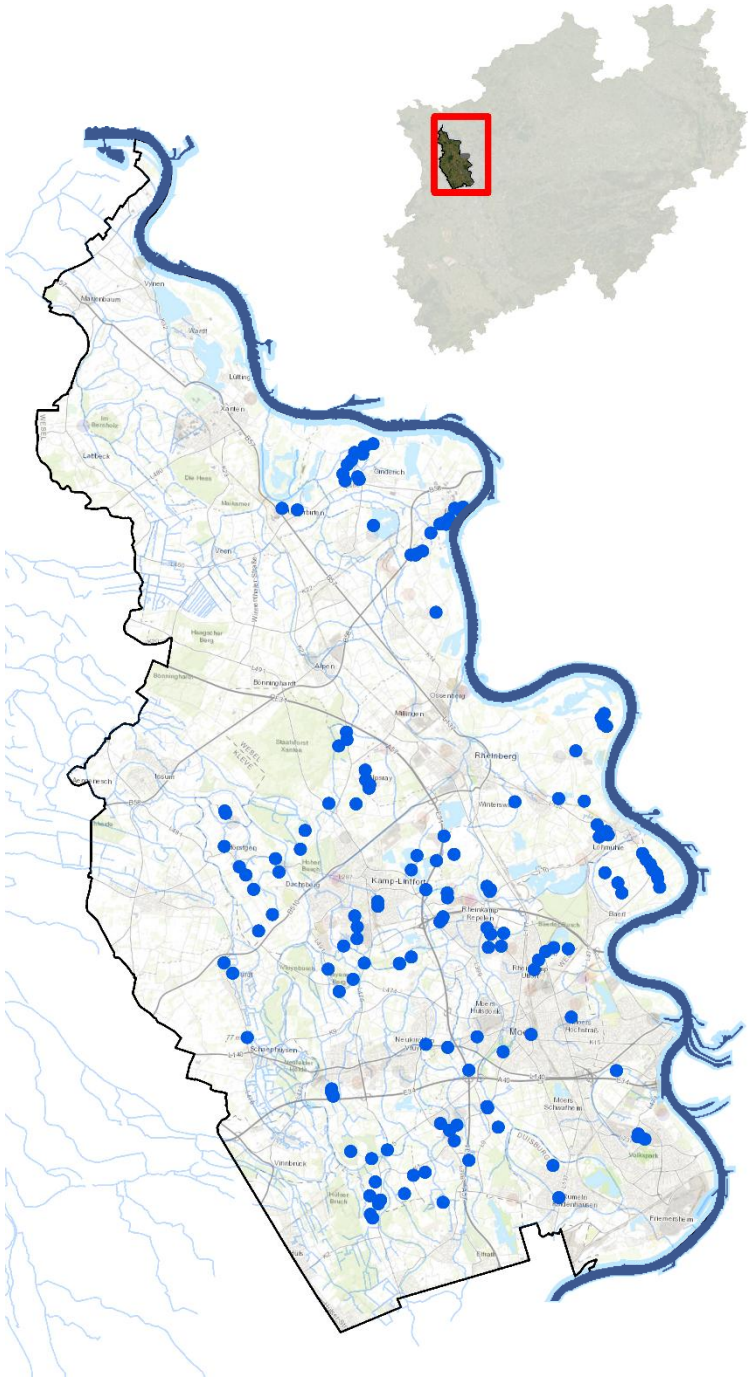
Piktogramm nach thenounproject.com

gasanzeiger.de

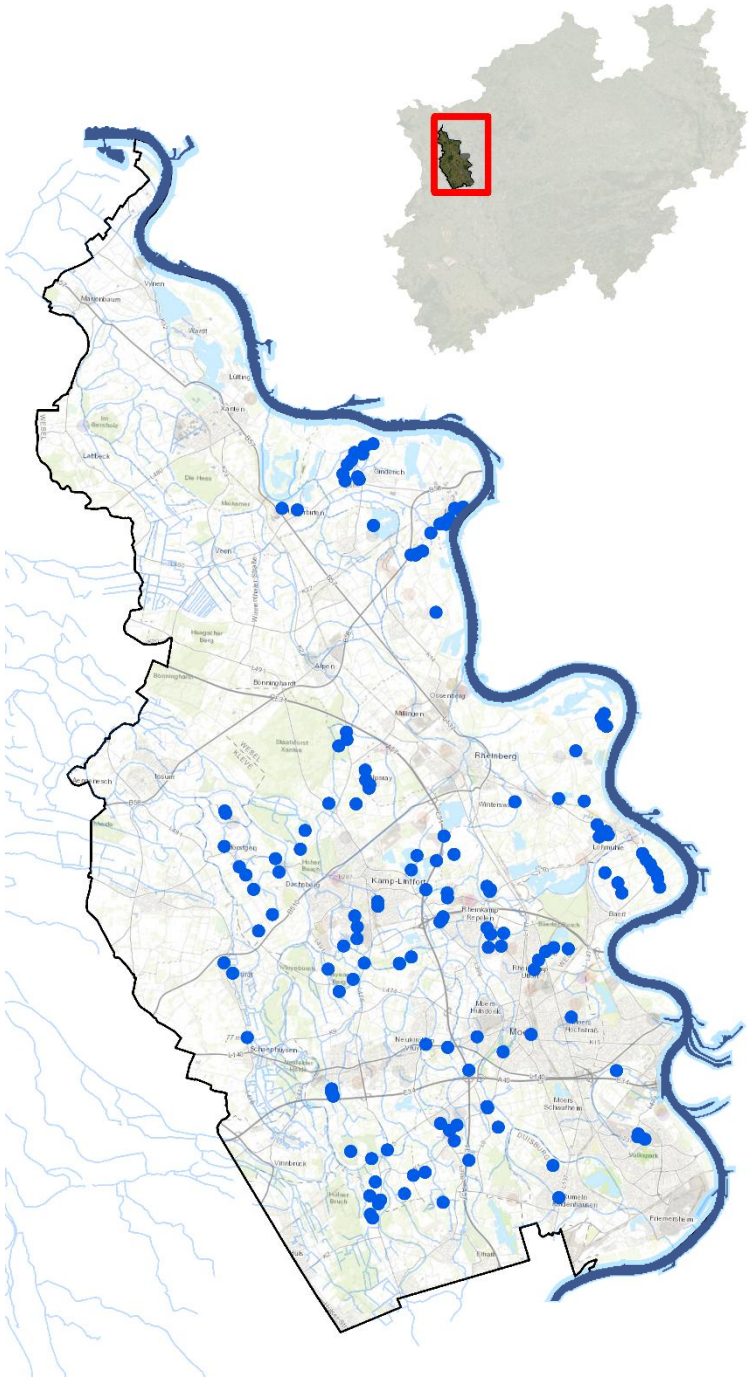
Daseinsvorsorge

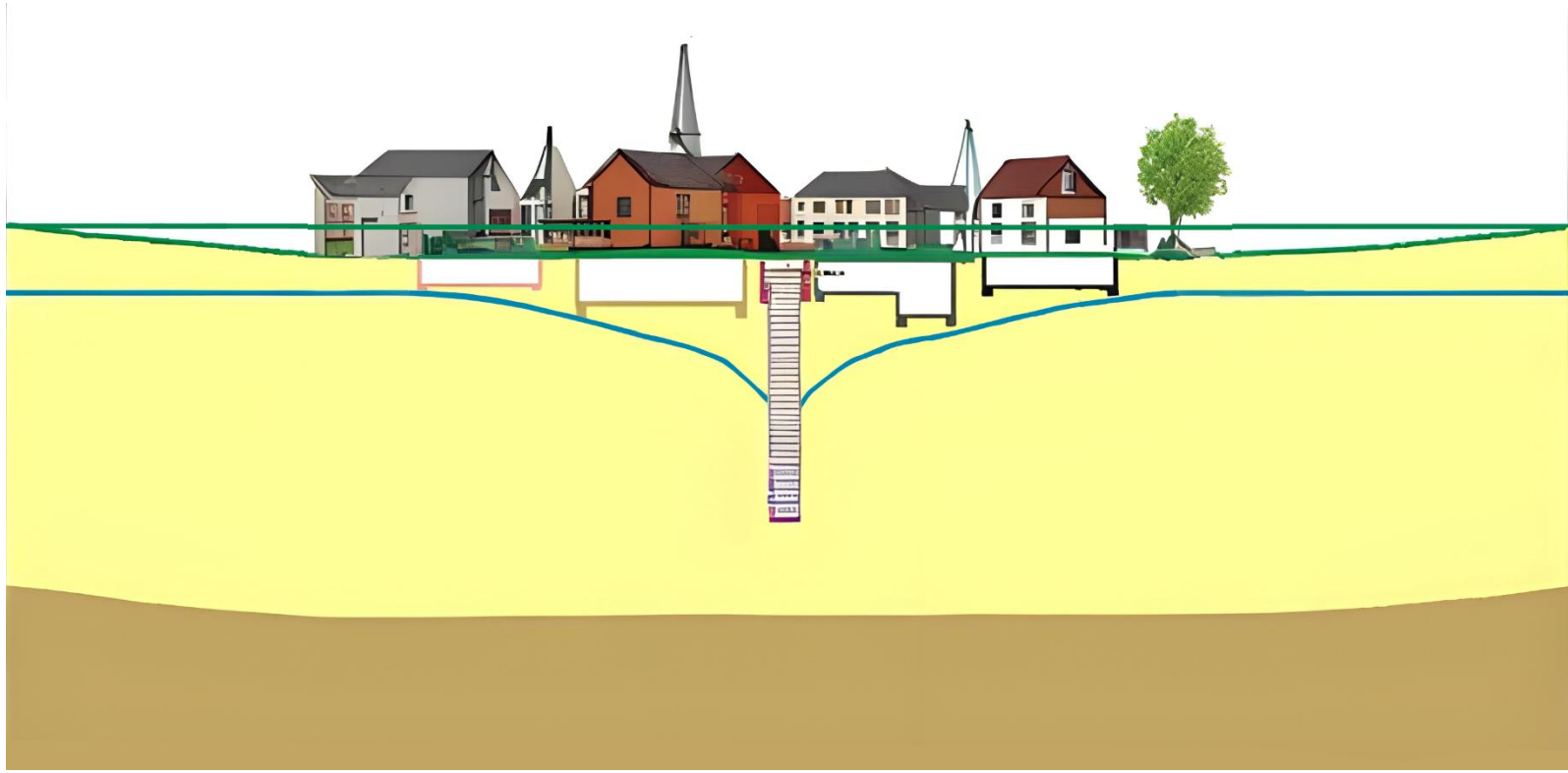


Daseinsvorsorge



Daseinsvorsorge





Grundwasser: ≈ 90 $Mm^3\ a^{-1}$ ~ 10 $GWh\ a^{-1}$

Vorflut: ≈ 168 $Mm^3\ a^{-1}$ ~ 18,5 $GWh\ a^{-1}$

Angaben aus 2022

Anlagenbetrieb heute

Anlagenbetrieb heute

Pumpe 6: 20,21 mNHN

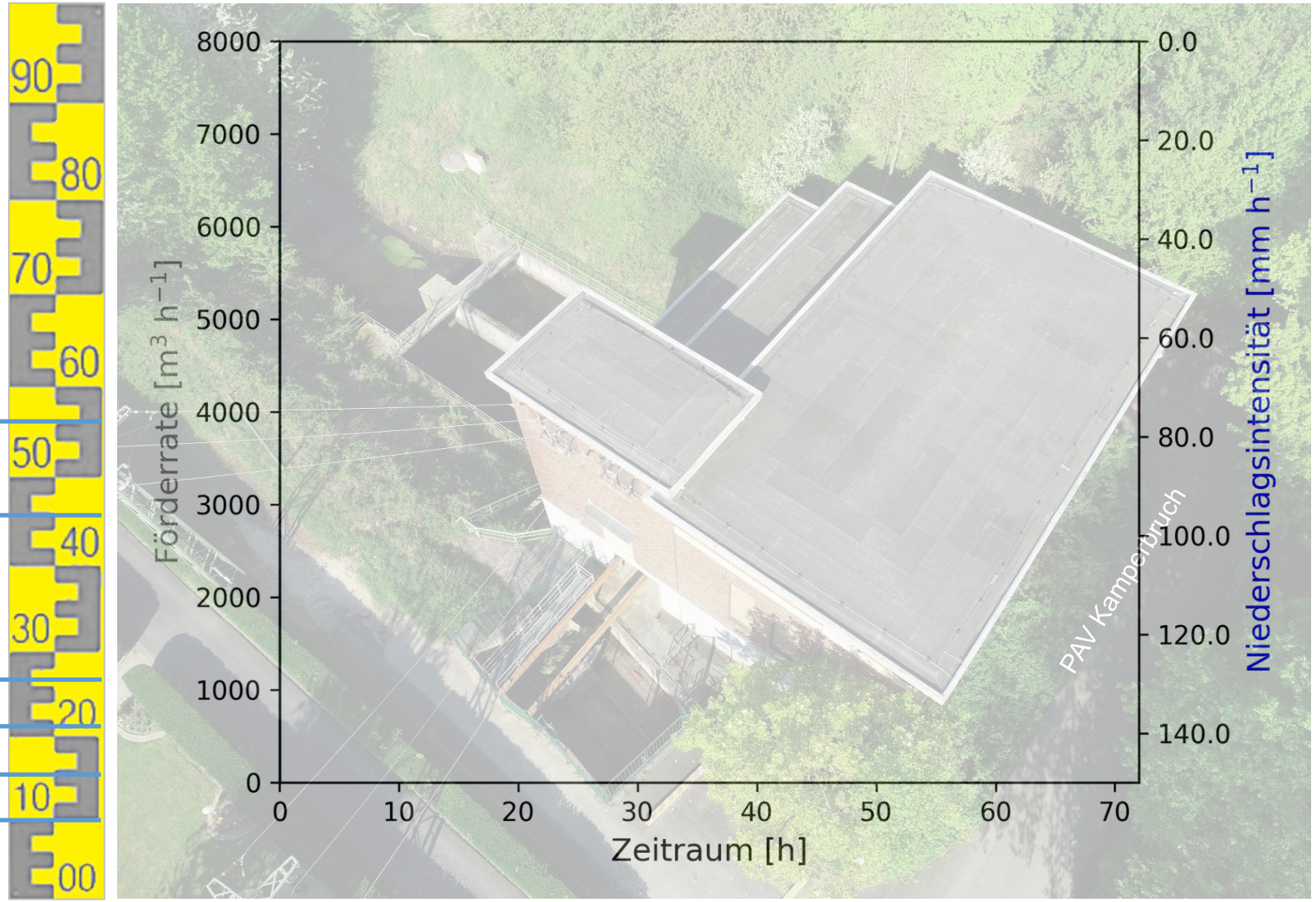
Pumpe 5: 20,11 mNHN

Pumpe 4: 19,91 mNHN

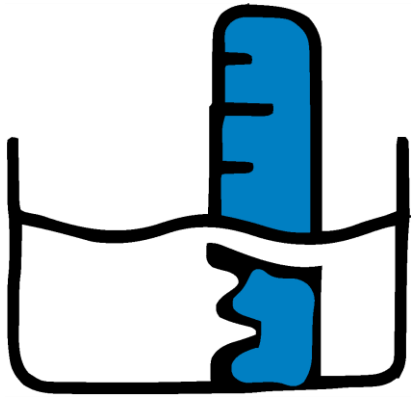
Pumpe 3: 19,86 mNHN

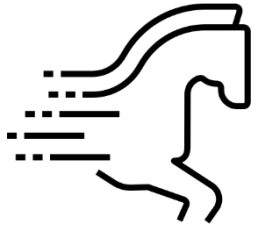
Pumpe 2: 19,81 mNHN

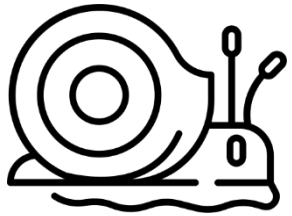
Pumpe 1: 19,76 mNHN



Reaktionszeiten



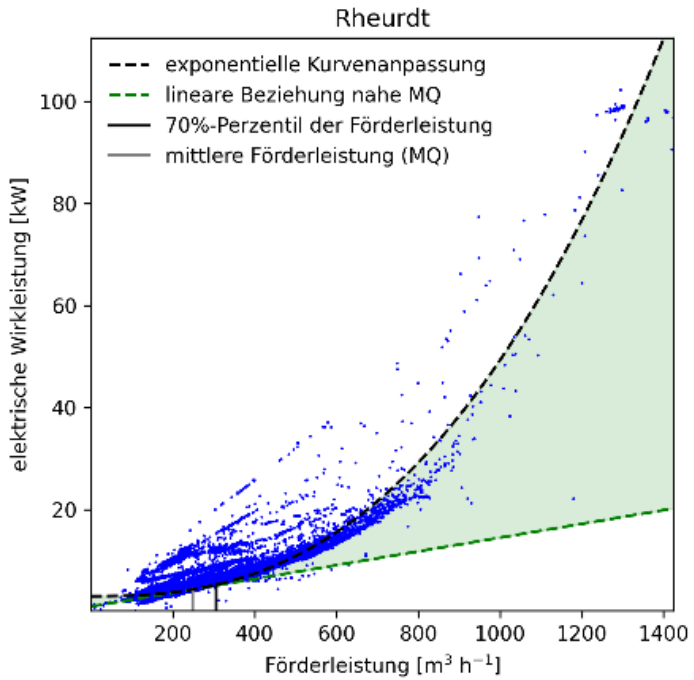


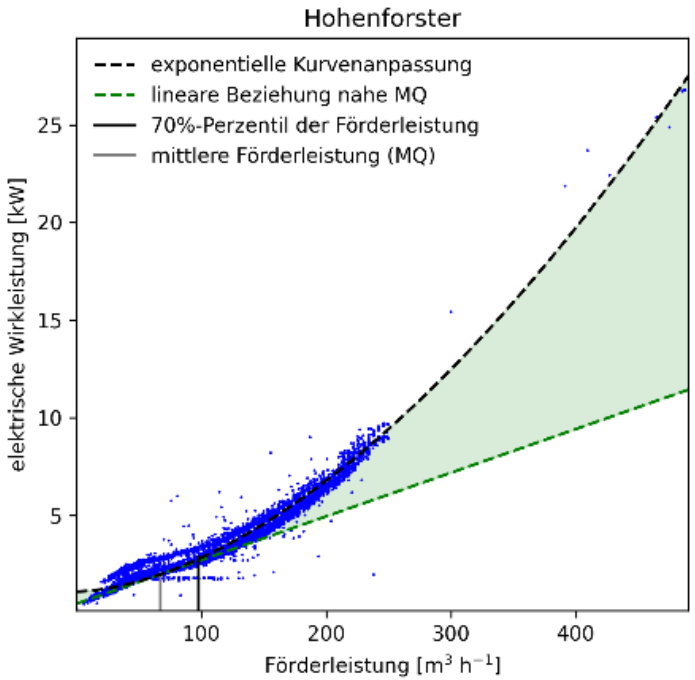


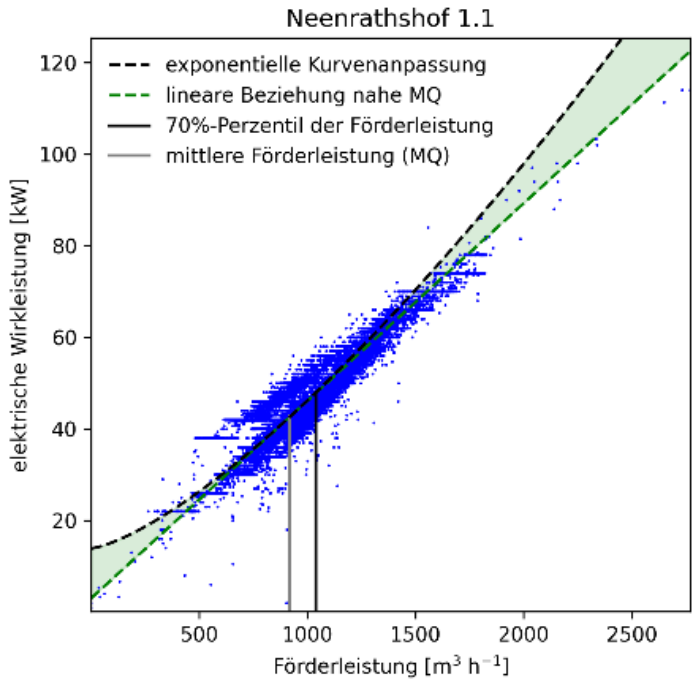
zeitliche Entkopplung
nützlich durch Trägheit

Piktogramme nach thenounproject.com

Potentialbewertung





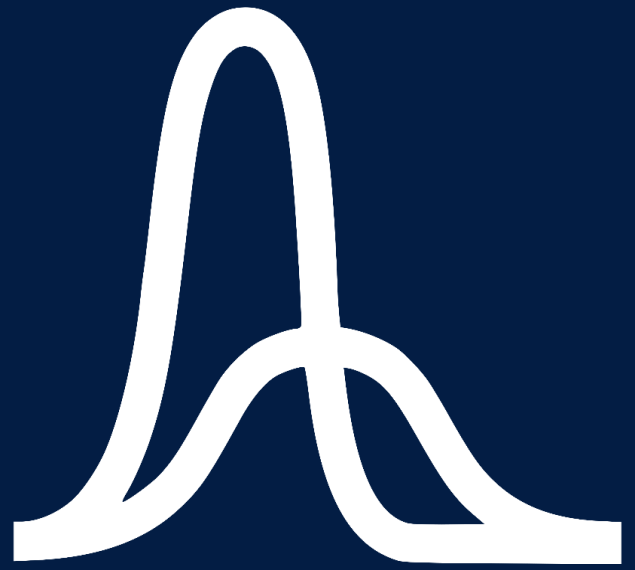


Spezifischer Verbrauch steigt in Lastfällen: gepumpter m³ wird teurer.

Motivation

Einsparung

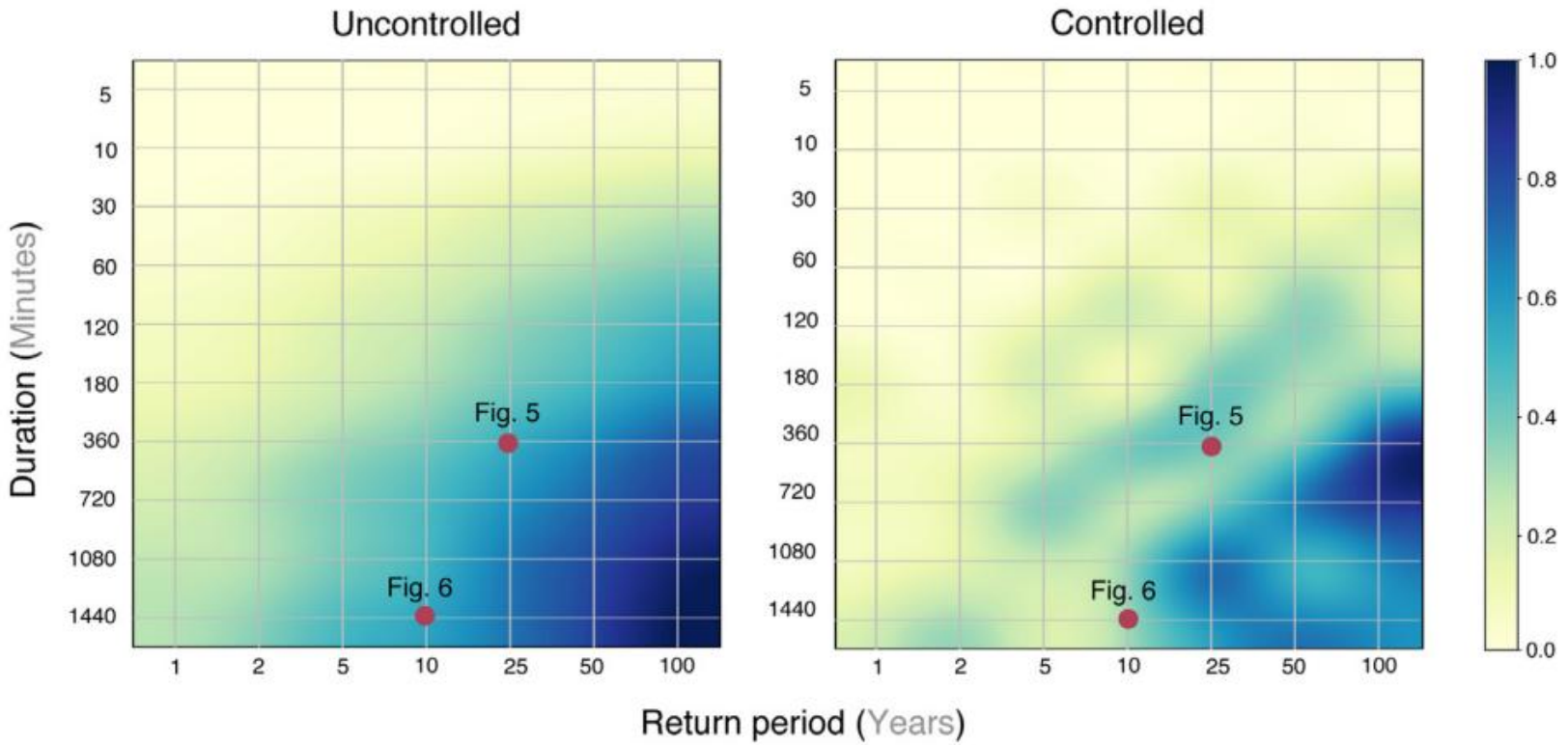
Resilienz



Vergleichmäßigung

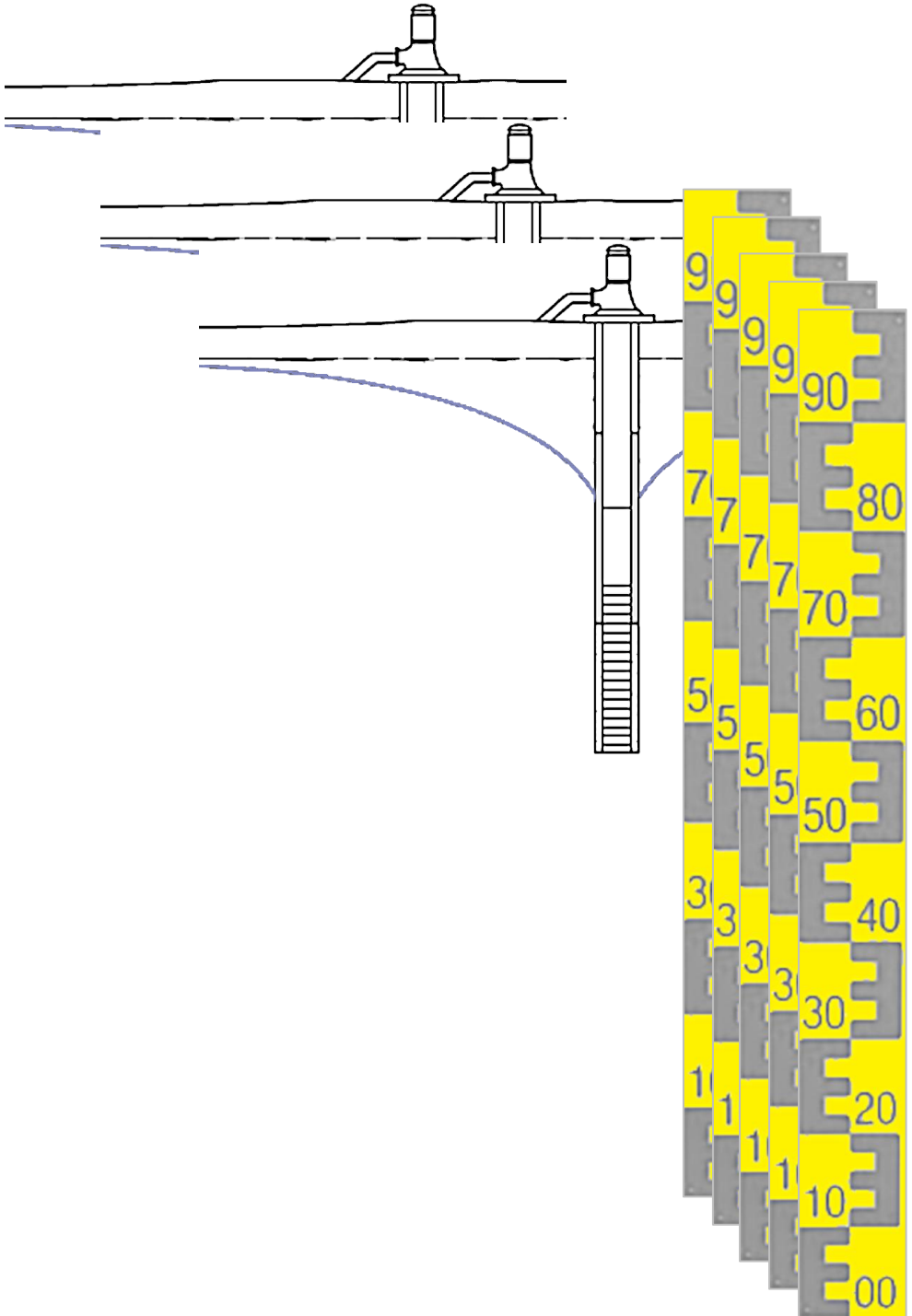
Piktogramme nach thenounproject.com

Motivation

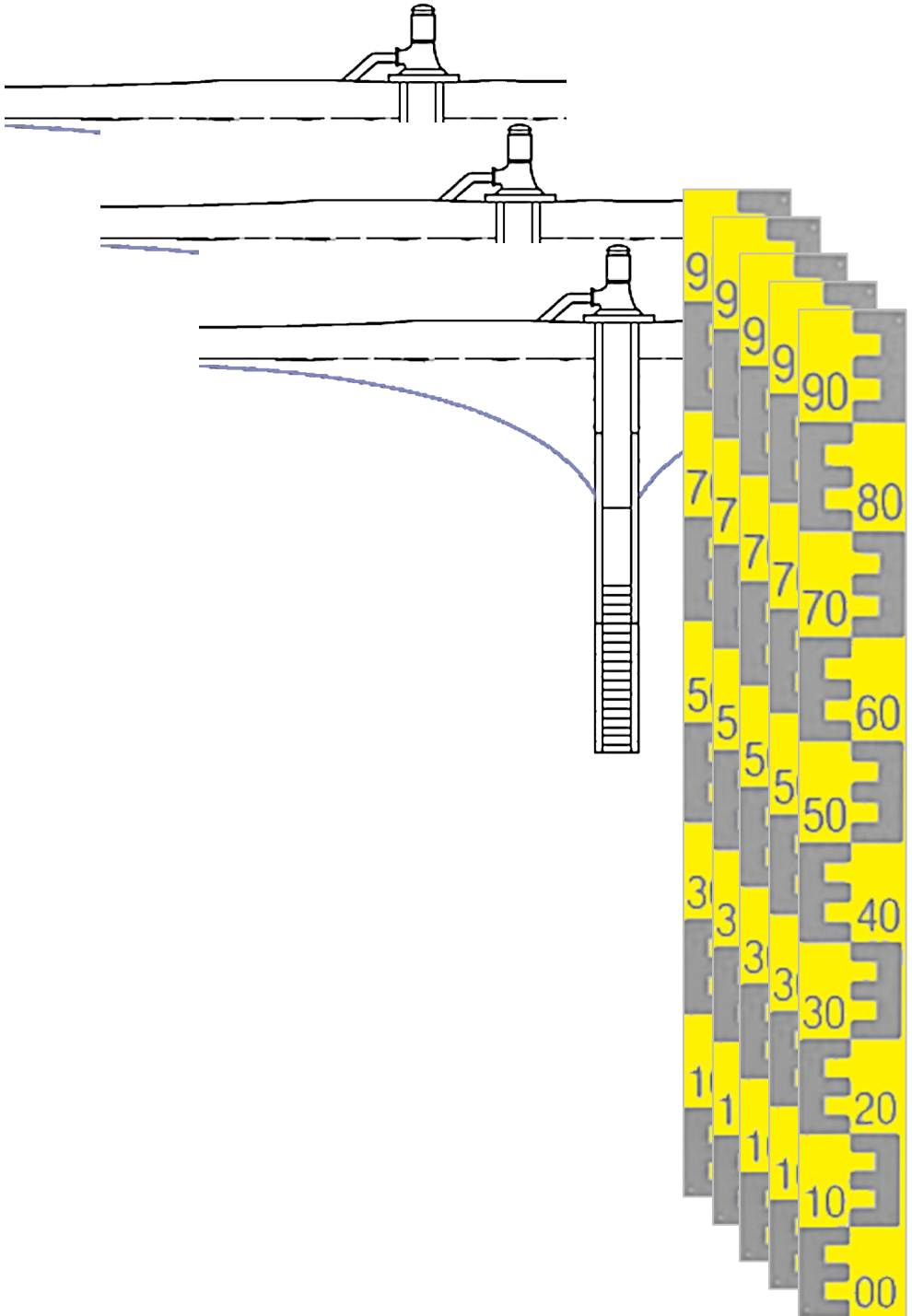


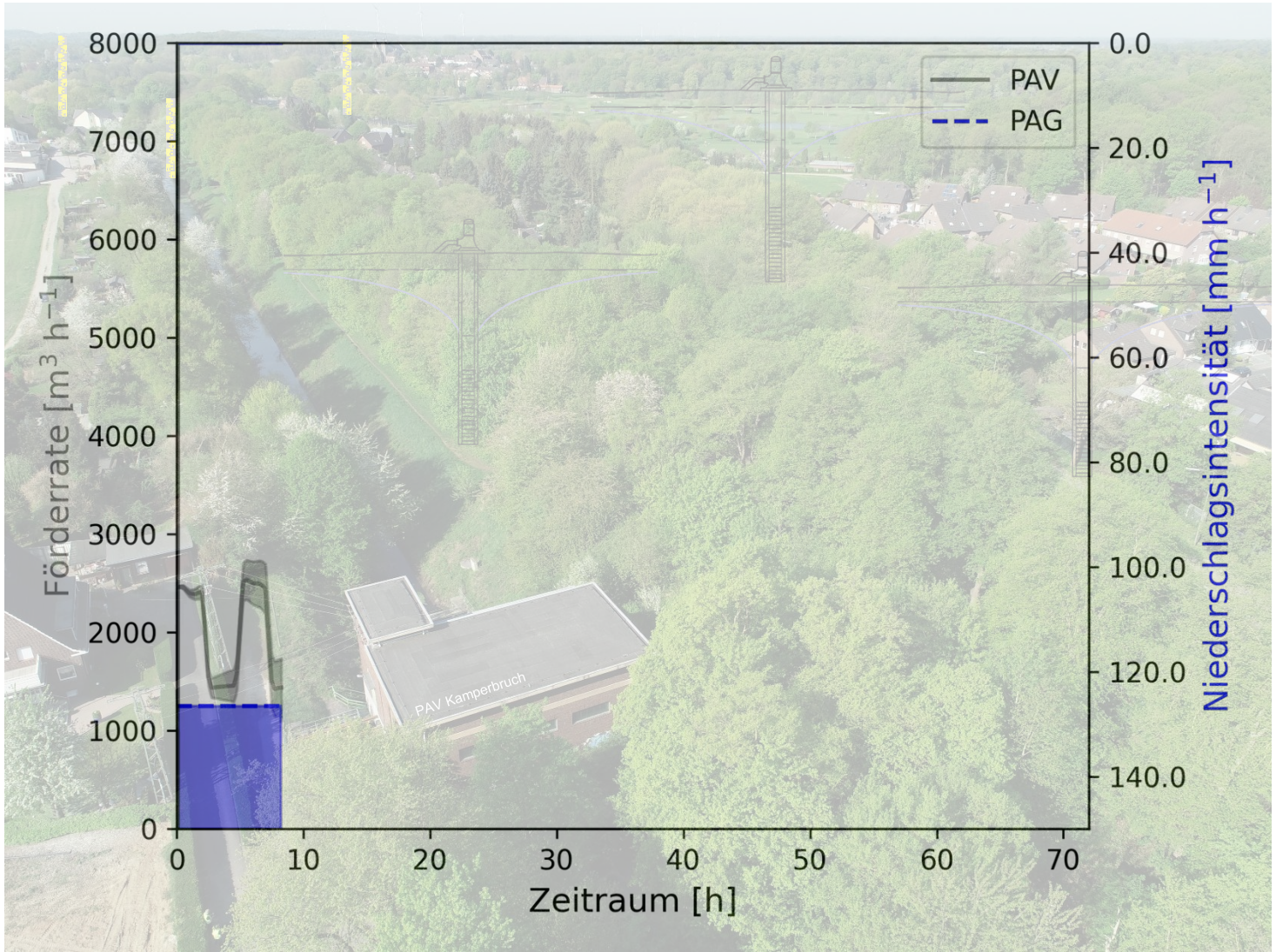
Mullapudi et al. (2020); open-storm.org

Anlagenbetrieb im Verbund

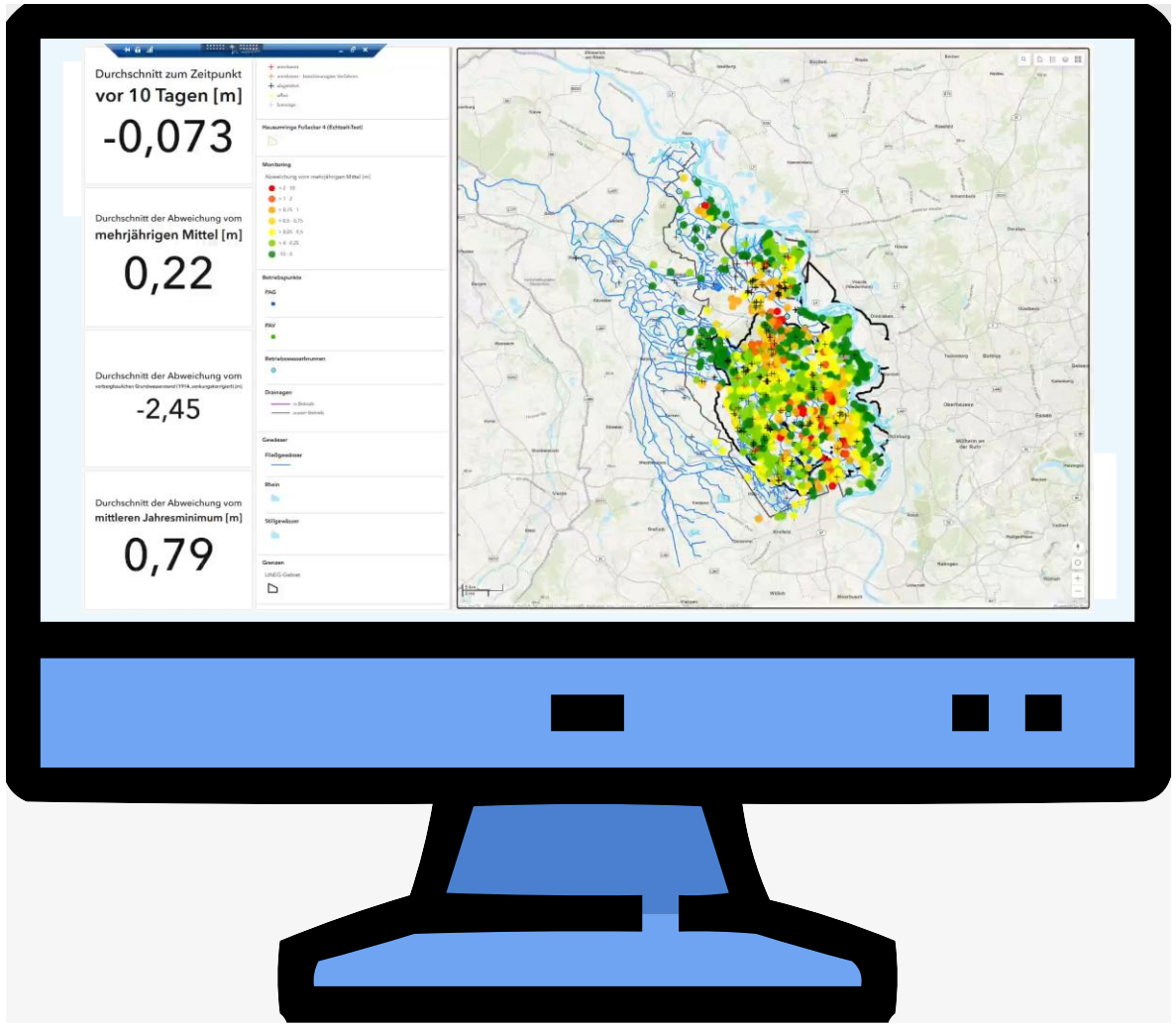


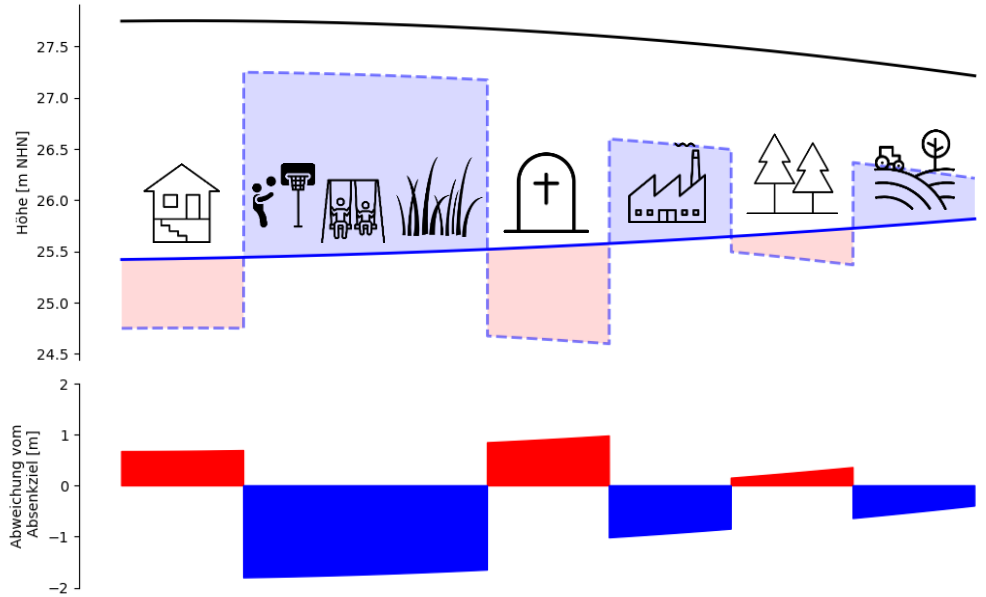
Anlagenbetrieb im Verbund

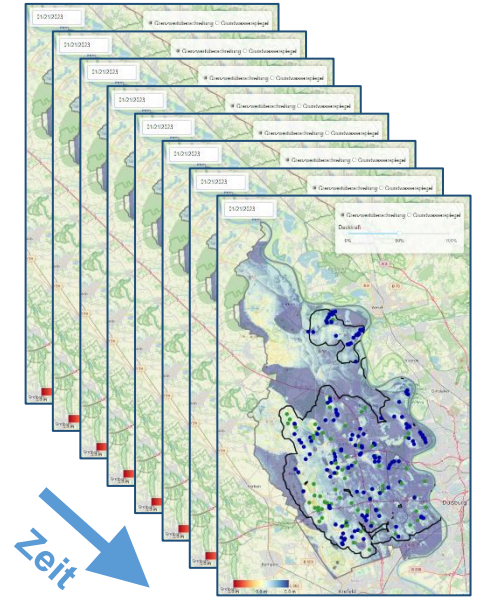




Digitales Schutzziel-Kataster







Datensilos

- multiple Quellen
- heterogene Konnektivität

Messnetze

Piktogramme nach thenounproject.com

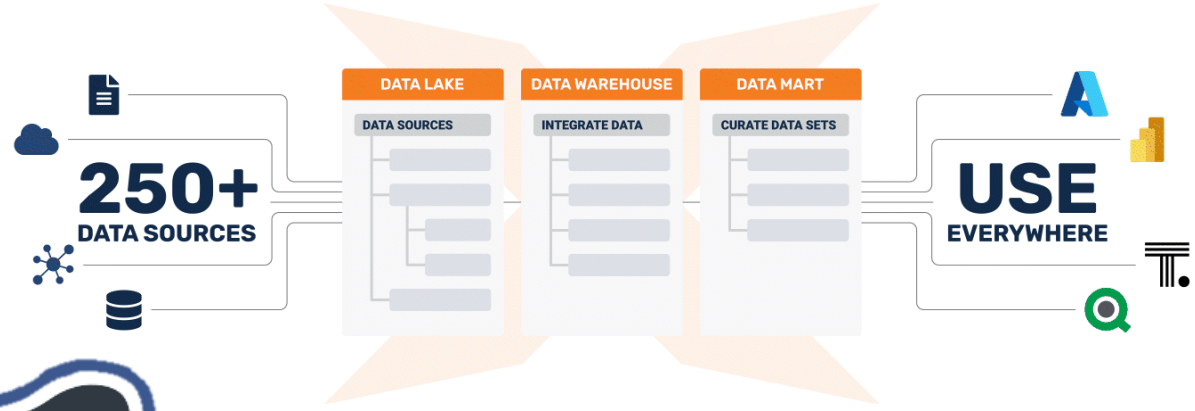
Messnetze

Datensilos


Prozessleitsystem (PLS)
Microsoft SQL Server
on-premises
DWD
FTP
off-premises
GIS
SQL Server
on-premises
Messdaten-
Managementsystem
dateibasiert / REST API
on-premises
TIMEXTENDER
Data Integration
Master Data
Management
Data Quality
Ochestration
DATAWAREHOUSE
SQL
Microsoft SQL
On-Premises
DataRobot
Model
Development
Timeseries
Forecasting
Realtime
Prediction
Deployment
Monitoring
ThoughtSpot.
Searchdriven
Analytics
Liveboard
Analytics

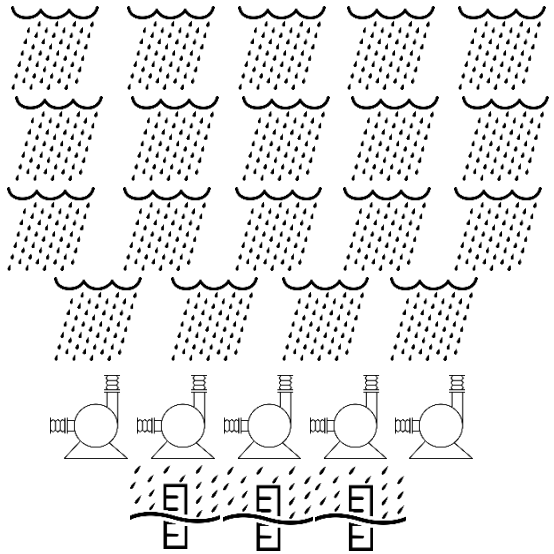
Datenflüsse

```python
def MNQ_berechnen(self, key, freq='1D'):
    df = self.createPandasDataframe(key='val')
    df = df.groupby(pd.Grouper(freq=freq)).mean()
    df = df.assign(Season=((df.index + pd.offsets.MonthBegin(2))).day + 1)
    MNQ = df.groupby('Season').min()
    MNQ = np.mean(MNQ[key])

    return MNQ

def MQ_berechnen(self, key, freq='1D'):
    df = self.createPandasDataframe(key='val')
    df = df.groupby(pd.Grouper(freq=freq)).mean()
    df = df.assign(Season=((df.index + pd.offsets.MonthBegin(2))).day + 1)
    MQ = df.groupby('Season').mean()
    MQ = np.mean(MQ[key])

    return MQ

def WWJ_ergaenzen(self, df):
    df = df.assign(WWJ=(df.index + pd.offsets.MonthBegin(2)).year)

    return df

def TagimWWJ_ergaenzen(self, df):
    df['start_of_WWJ'] = [pd.Timestamp(df.WWJ[i]-1, 11, 1, 0, 0, 0) for i in range(len(df.WWJ))]
    df['days_since_start_of_WWJ'] = [v.total_seconds()/(60.*60.*24.) for v  in (df.index - df['start_of_WWJ'])]

    return df

def Intervallmittelwerte(self, df, freq):
    df = df.groupby(pd.Grouper(freq=freq)).mean()

    return df
```



Vorhersage-Pipeline

Input



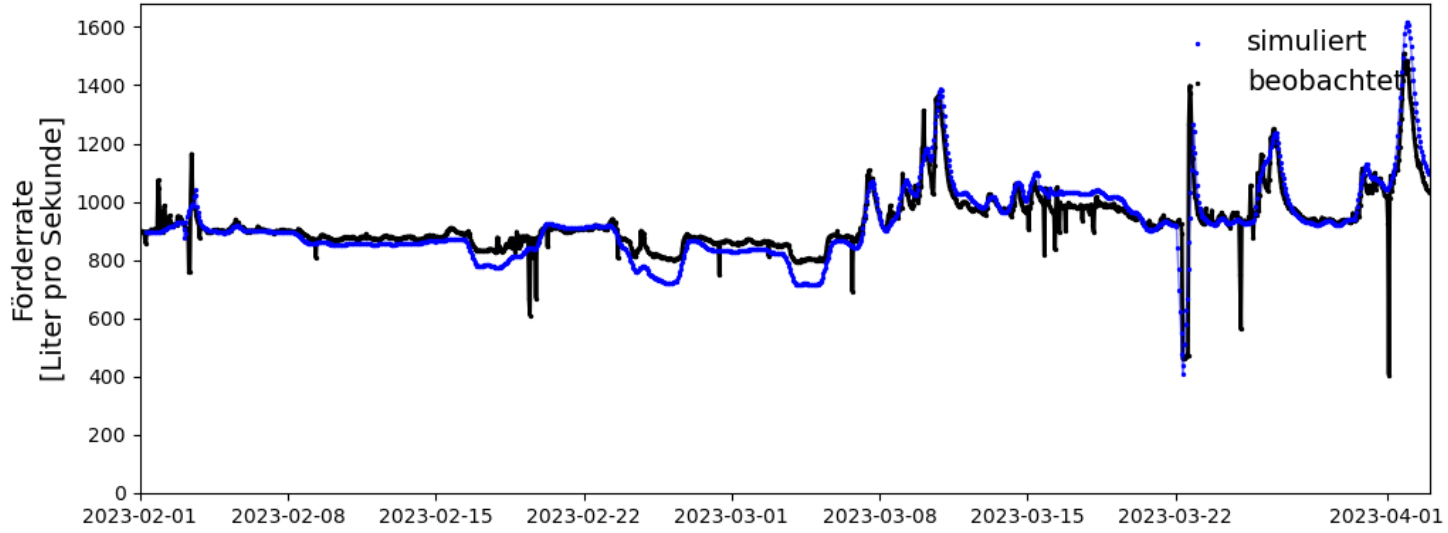
Output

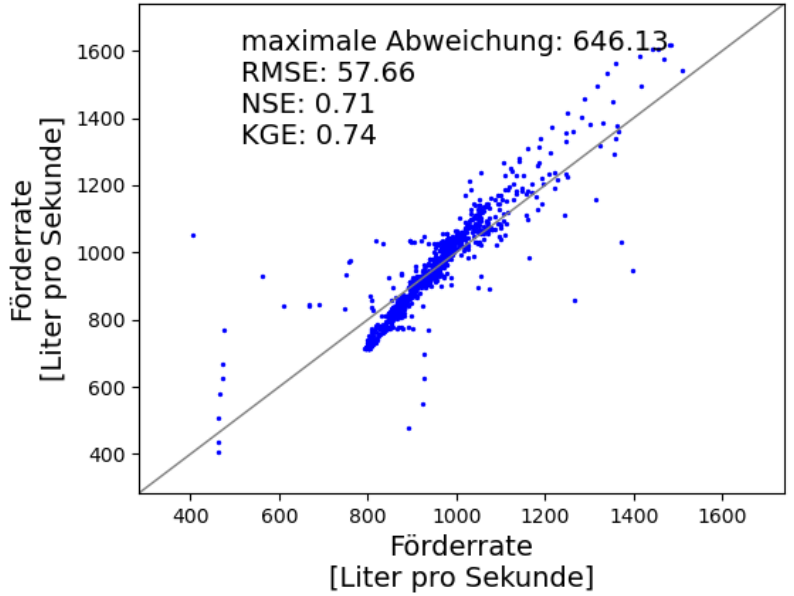
Ø
Förderrate
nächste 12h

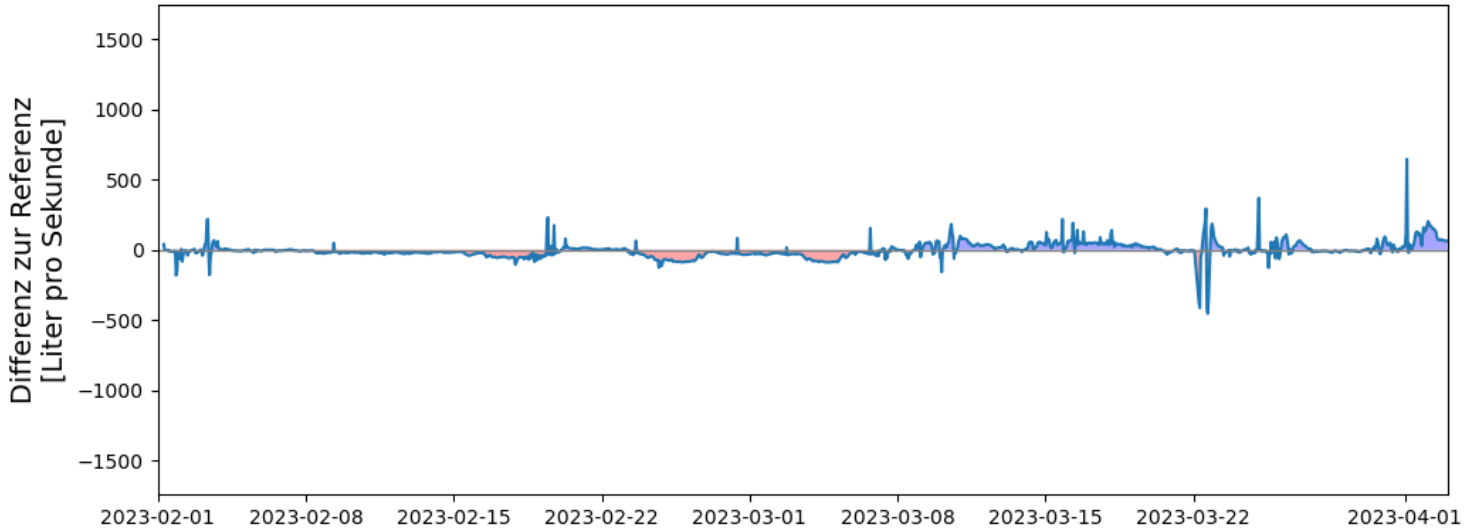
Ø
Wasserstand
nächste 12h

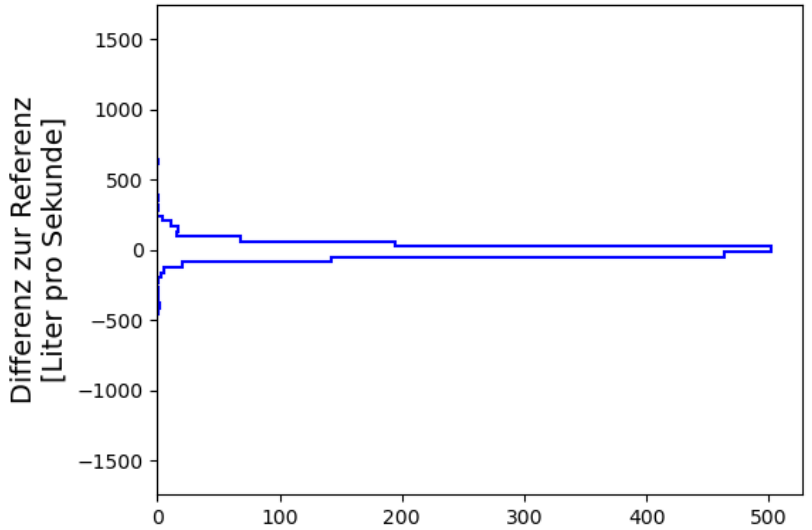
max.
Wasserstand
nächste 12h

12h-Prognosen









Betriebsalternativen

Ergänzend
zu Bewährtem:

Rückfallebene mit normalem
Schaltpunkt-Betrieb, jederzeit

weitere denkbare Betriebsalternativen, z.B.:

1) temporärer PAV-Grundbetrieb trotz Lastfall
   (bei schadfreier Retention)
2) tageszeitenabhängiger Betrieb

Vielen Dank.

LINEG

Dr. Philipp Höhn
hoehn.p@lineg.de