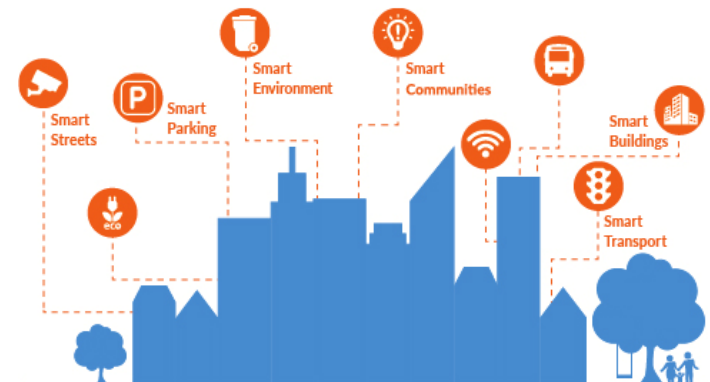


# Delft-FEWS Regionales Anwendertreffen FEWS Web Dienste

Aachen, 28.-29. Juni 2018

Technologiezentrum Aachen, Hydrotec

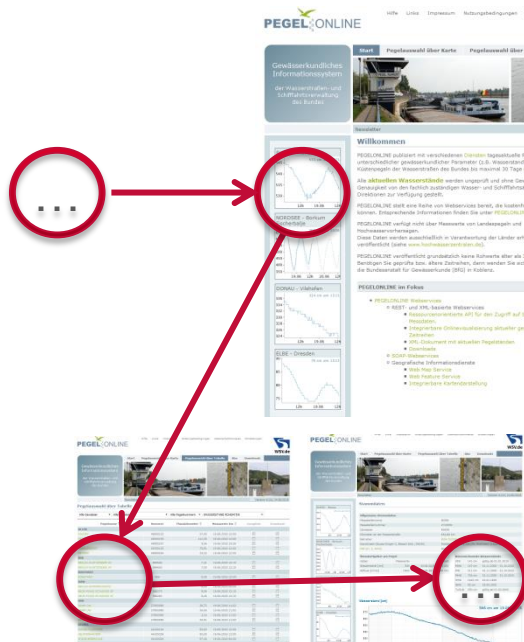


- Warum ein **Webservice**?
- Was kann der FEWS Web REST Service **schon jetzt**?
- Wofür könnten wir das einsetzen?
- **Showcase** - Prototyp Hydrotec Pegelmonitoring mit IoT Integration

- ▶ Ermöglicht die M2M-Kommunikation
- ▶ Dabei werden Daten ausgetauscht und auf entfernten Computern Funktionen aufgerufen.



- Automatisierung der Prozesse
- Vorteile der öffentlich verfügbaren Daten nutzen!
- Integration in die bereits bestehende Applikationen
- Zielgerichtete Abfrage und Weiterverarbeitung



VS

<https://www.pegelonline.wsv.de/webservices/rest-api/v2/stations.json?ids=BONN,2730010>

```

0:
  uuid: "593647aa-9fea-43ec-a7d6-6476a76ae868"
  number: "2710080"
  shortname: "Bonn"
  longname: "BONN"
  km: 654.8
  agency: "WSA DUISBURG-RHEIN"
  longitude: 7.108045442593696
  latitude: 50.7363979694278
  water:
    shortname: "RHEIN"
    longname: "RHEIN"
1:
  uuid: "a6ee8177-107b-47dd-bcfd-30960ccc6e9c"
  number: "2730010"
  shortname: "Köln"
  longname: "Köln"
    
```

Kleiner Webservice in 5 Zeilen:

1. `a=function(){e="",t=prompt("geben Sie eine Stadt an","Koblenz");`
2. `fetch("https://www.pegelonline.wsv.de/webservices/rest-api/v2/stations/"+t+"/W/measurements.json")`
3. `.then(e=>e.json()).then(function(n){n.map(function(t){e=e+"<p>Messwert <b>"+t.value+"</b> um "`
4. `+t.timestamp+"</p><br>"},".main .homepage").html("<h1>Station "+t`
5. `+".</h1><br><h2>letzte Werte für die Station "+t"</h2><br>+e)}});var a,e;`

**Immer weniger Leuten mit mehr Aufgaben  
verarbeiten mehr Informationen.  
Verdichtung schafft Übersichtlichkeit.**

**→ Ressourcen optimal nutzen**



# Was bietet der FEWS Web Service an?

## ▸ Bereits verfügbare Befehle in FEWS PI REST Web Service

### GET



- **locations**
- samples
- filters
- modifiers
- timeseries
- workflows
- timezoneid
- parameters
- processdata
- timeseriesmodifiers
- taskruns
- taskrunstatus

**„locations“**  
Zugriff auf die Stationen

- Koordinaten
- Name der Station
- ...

### POST



- **runtask**
- **modifiers**
- **timeseries**

## ▢ Bereits verfügbare Befehle in FEWS PI REST Web Service

### GET



- **locations**
- samples
- filters
- modifiers
- **timeseries**
- workflows
- timezoneid
- parameters
- processdata
- timeseriesmodifiers
- taskruns
- taskrunstatus

#### „timeseries“

Zugriff auf die Zeitreihen

- Nutzung der Filter
- Einzelne Stationen

### POST



- **runtask**
- **modifiers**
- **timeseries**

## ▸ Bereits verfügbare Befehle in FEWS PI REST Web Service

### GET



- **locations**
- samples
- filters
- modifiers
- **timeseries**
- **workflows**
- timezoneid
- parameters
- **processdata**
- timeseriesmodifiers
- **taskruns**
- **taskrunstatus**

#### „taskruns & taskrunstatus“

Definierte Workflows abfragen

- Voraussichtliches Ende
- Status
- Durch wen gestartet
- ...

### POST



- **runtask**
- **modifiers**
- **timeseries**



## ▸ Bereits verfügbare Befehle in FEWS PI REST Web Service

### GET



- **locations**
- samples
- filters
- modifiers
- **timeseries**
- **workflows**
- timezoneid
- parameters
- **processdata**
- timeseriesmodifiers
- **taskruns**
- **taskrunstatus**

### POST



- **runtask**
- **modifiers**
- **timeseries**

### FEWS Manipulieren

Daten in FEWS schreiben, Workflows starten..

# Wofür könnten wir das nutzen

	Schwellenwerte beobachten / warnen
Messwerte abfragen	Integration in interne Systeme
	Messwerte nach außen kommunizieren
	Szenarien
FEWS Produkte in bestehende Systeme integrieren	Überflutungen berechnen
	Wahrscheinlichkeitsberechnungen aus FEWS nutzen
	offene Quellen mit FEWS Daten ergänzen
Aggregation diverser Daten auf einem Portal	Vorhandene Ressourcen optimal ausschöpfen
	Anbindung weiterer Produkte
	Messwerte / Simulationsergebnisse visualisieren
FEWS Web Client	Workflows (Status, Neustart, Auslösen)
	Stammdaten / Systemstatus abfragen

→ Workshop

# Beispiel

## Prototyp Hydrotec Pegelmonitoring mit IoT Integration



The screenshot shows the IFTTT app interface with a workflow for a smart home system. The workflow starts with a 'Webhook' trigger, followed by a 'Conditional' node. If the condition is met, it triggers 'Device: State' and 'Storage: Get...'. If not, it triggers 'Device: Connect' and 'Device: Disconnect'. The workflow then proceeds through 'Endpoint', 'Data Table', 'Event', 'GCP Pub/Sub', 'Meridian', 'MQTT', 'Particle', 'Timer', 'Virtual Button', and 'Webhook' nodes, ending with 'Authenticate' and 'Generate Token'.

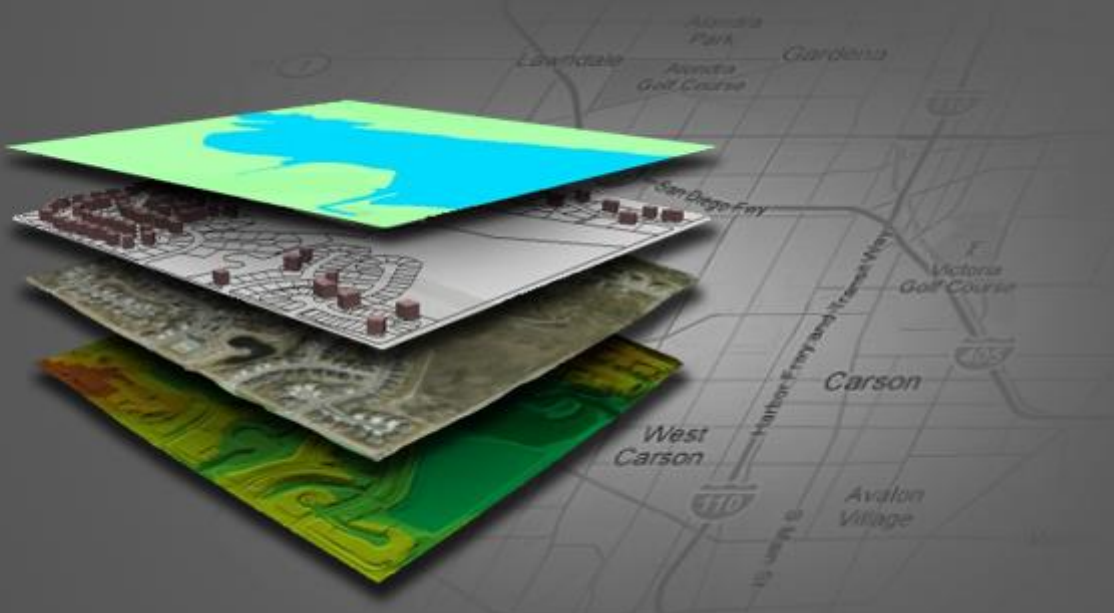
The screenshot displays the ArcGIS GeoEvent Manager interface. The top navigation bar includes 'Monitor', 'Inputs', 'GeoEvent Services', and 'Outputs'. The 'Monitor' tab is active, showing a table with the following data:

Status	In/Out	Count	Rate (over last 5 mins)	Edit Rate	Min Rate	Time Since Last
STARTED	In	144	0 /sec		0 /sec	00:00:01
	Out	92	0 /sec		0 /sec	00:00:02

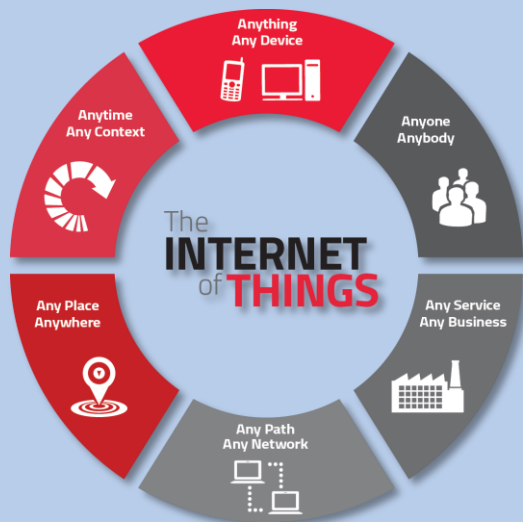
Below the table, a workflow diagram is visible, showing a sequence of processing steps:

- Twitter** (green box) connects to a **Geo-enabled Filter** (yellow diamond).
- The **Geo-enabled Filter** has three outputs:
  - Upwards to **Twitter (stream out)** (blue box).
  - Rightwards to **HQTP-Output-Connector-2** (blue box).
  - Downwards to **Buffer on Tweet (Buffer Creator)** (yellow box).
- The **Buffer on Tweet (Buffer Creator)** connects to **ConnectStreams (JoinTrigger)** (yellow box).
- ConnectStreams (JoinTrigger)** connects to **Mapper1 (Field Mapper)** (yellow box).
- Mapper1 (Field Mapper)** connects to **CountTweets (Field Calculator (Regular Expression))** (yellow box).
- CountTweets (Field Calculator (Regular Expression))** connects to **SummedSpaced (Field Calculator)** (yellow box).
- SummedSpaced (Field Calculator)** connects to **CountTweets (Field Calculator)** (yellow box).
- CountTweets (Field Calculator)** connects to a **MultiJoinCT-roads** (yellow diamond).
- The **MultiJoinCT-roads** has three outputs:
  - Upwards to **HQTP-Output-Connector** (blue box).
  - Rightwards to **CountJoinCT1 (Field Editor)** (yellow box).
  - Downwards to **BufferFinalRoom (Buffer Creator)** (yellow box).
- The **CountJoinCT1 (Field Editor)** connects to **CountJoinCT1 (Field Editor)** (yellow box).
- The **CountJoinCT1 (Field Editor)** connects to **CountJoinCT1 (Field Editor)** (yellow box).
- The **CountJoinCT1 (Field Editor)** connects to **CountJoinCT1 (Field Editor)** (yellow box).

# Geo mit Hardware verzahnen



Leaflet

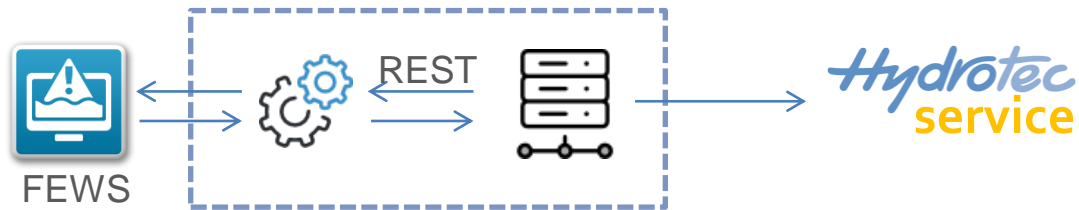


**LOSANT**

ESP8266

MQTT.org

# Was passiert gleich?



# Was passiert gleich?

