

Azure Digital Twin

PROTOTYPISCHE UMSETZUNG EINER WINDTURBINE ALS
DIGITAL TWIN

Unser Projekt

- Physische System ist eine Windturbine des Typs [AN Bonus 1000/54](#)
- Keine Anbindung einer existierenden Windturbine
- Daten sind historische gemessene Werte der Windturbine
- Sensordaten zur Umwelt abgefragt über [Openweathermap](#) API
- Durch fehlende Rückkopplung an eine Hardware kann nur ein Digitaler Schatten umgesetzt werden

Use-Case und Ziele des Digitalen Schattens

Windturbinen sind dezentrale Einheiten eines Gesamtsystems. Eine Überwachung der Performance ist wettbewerbskritisch und ermöglicht Innovation.

Betreiber, Maintainer und ggf. Researcher können den Status, Umwelt- und Leistungsfaktoren Remote abrufen.

Maintainer: Echtzeitanalyse gegenüber der zyklischen Wartung

Researcher: Einsicht der Auswirkungen der Pitch-Angle auf Leistung

Rolle der Research Papers

- Paper A: Digital Twin of Wind Turbine Based on Microsoft Azure IoT Platform
 - Erstellter PoC diene als Grundlage zu unserer Umsetzung samt Formeln zur Windkraft Berechnung über Windkraft an sich gekoppelt mit Leistungskoeffizienten einer Windturbine
- Paper B: Digital twin with Machine learning for predictive monitoring of CO2 equivalent from existing buildings
 - Genaue Beschreibung eines Aufbaus deines DTs IoT Hub, Azure Functions, Event Grid etc.

Azure Digital Twins (PaaS)

- IoT Hub bietet die Möglichkeit zur Kommunikation mit dem PS per Http
- 3D-Szene zur grafischen Repräsentation
- Einfaches Anlegen des DTs durch DTDL
- Function Apps erlauben zusätzliches Handeln bei Events
- Speichern der Daten und Zugriff in ADT

Model Information

```
{
  "@id": "dtmi:example:Power;1",
  "@type": "Interface",
  "@context": "dtmi:dtdl:context;2",
  "displayName": "Power",
  "contents": [
    {
      "@type": "Property",
      "name": "Kilowatts",
      "schema": "double"
    },
    {
      "@type": "Property",
      "name": "Cp",
      "schema": "double"
    }
  ]
}
```

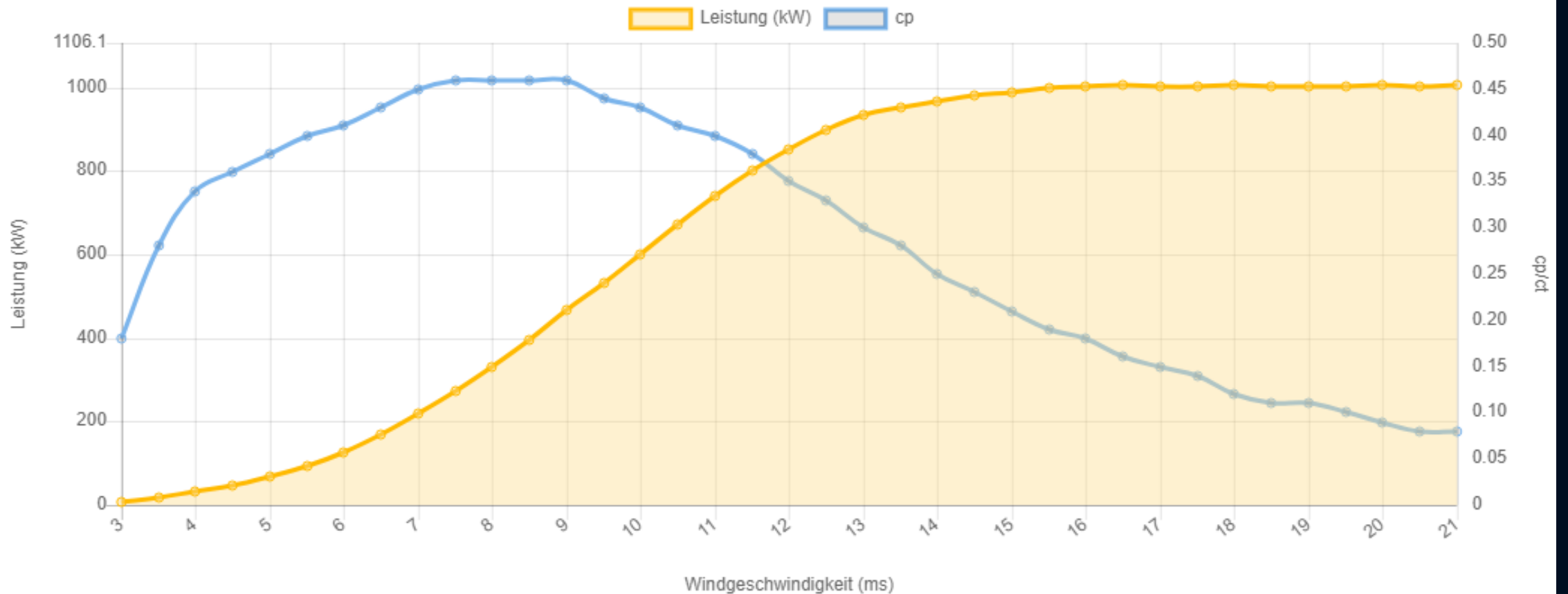
Physikalische System: AN Bonus 1000/54

- Modell- und historische Daten aus der Open Source Quelle www.wind-turbine-models.com bezogen.
- Rotordurchmesser/ -radius: 54.2 / 27.1 m
- Nennleistung: 1000.0 Kilowatt
- Einschaltgeschwindigkeit: 3.0 m/s
- Nenngeschwindigkeit: 15.0 m/s
- Abschaltgeschwindigkeit: 25.0 m/s
- Typ: On-Shore



Physikalisches System: AN Bonus 1000/54

Leistungskurve

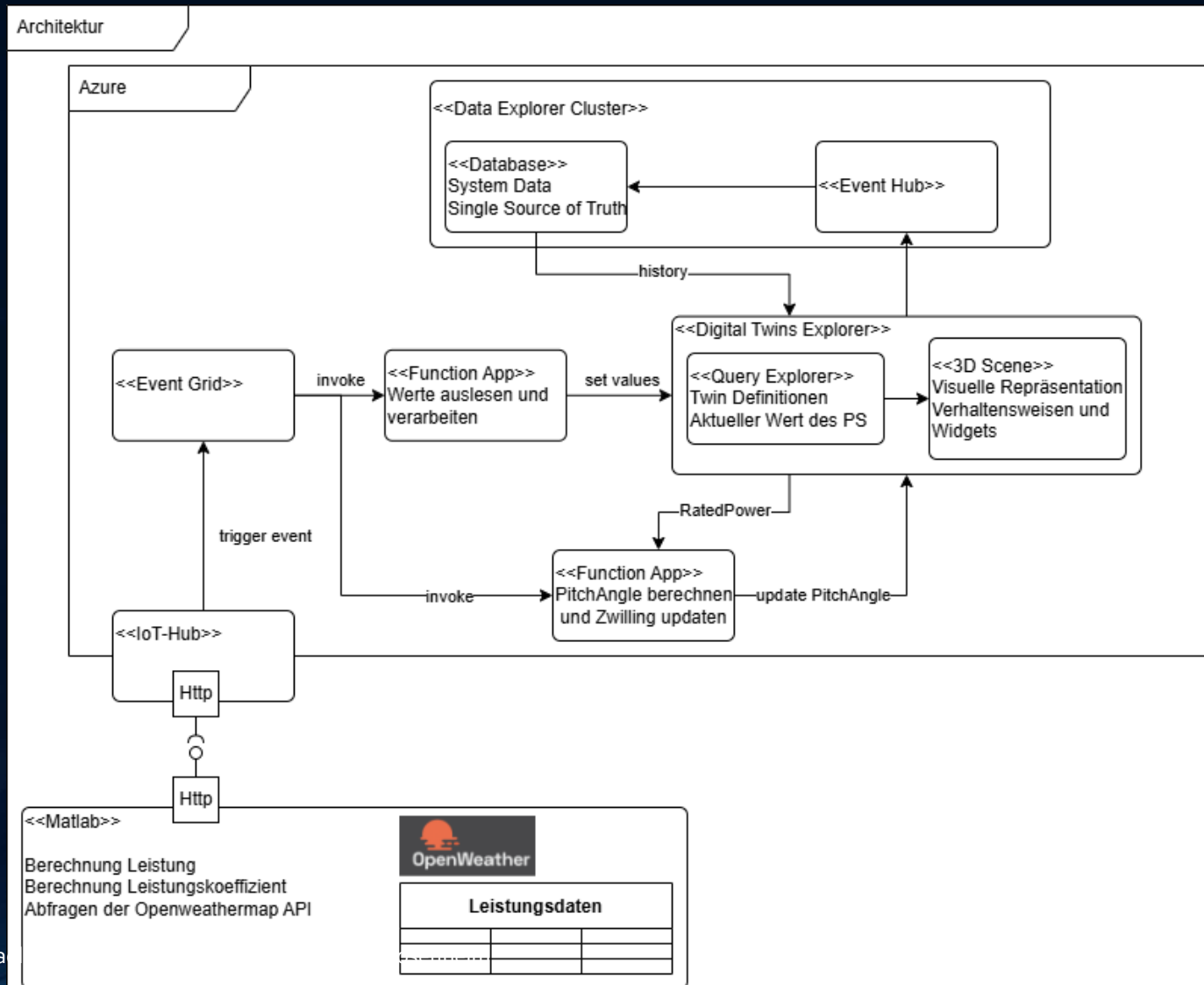


Physikalische System: AN Bonus 1000/54

- Sensordaten der Windturbine am Standort: Dingle, Ireland
- Relative Luftfeuchtigkeit: 0-1
- Luftdruck: in hPa
- Temperatur T: in Grad Celsius
- Windgeschwindigkeit: in m/s
- Luftdichte: Berechnet durch $\rho = \frac{(hPa * 100)}{(R_s * (T + 273.15))}$ mit R_s als spezifische Gaskonstante

Architektur des Digitalen Schattens

AN Bonus 1000/54

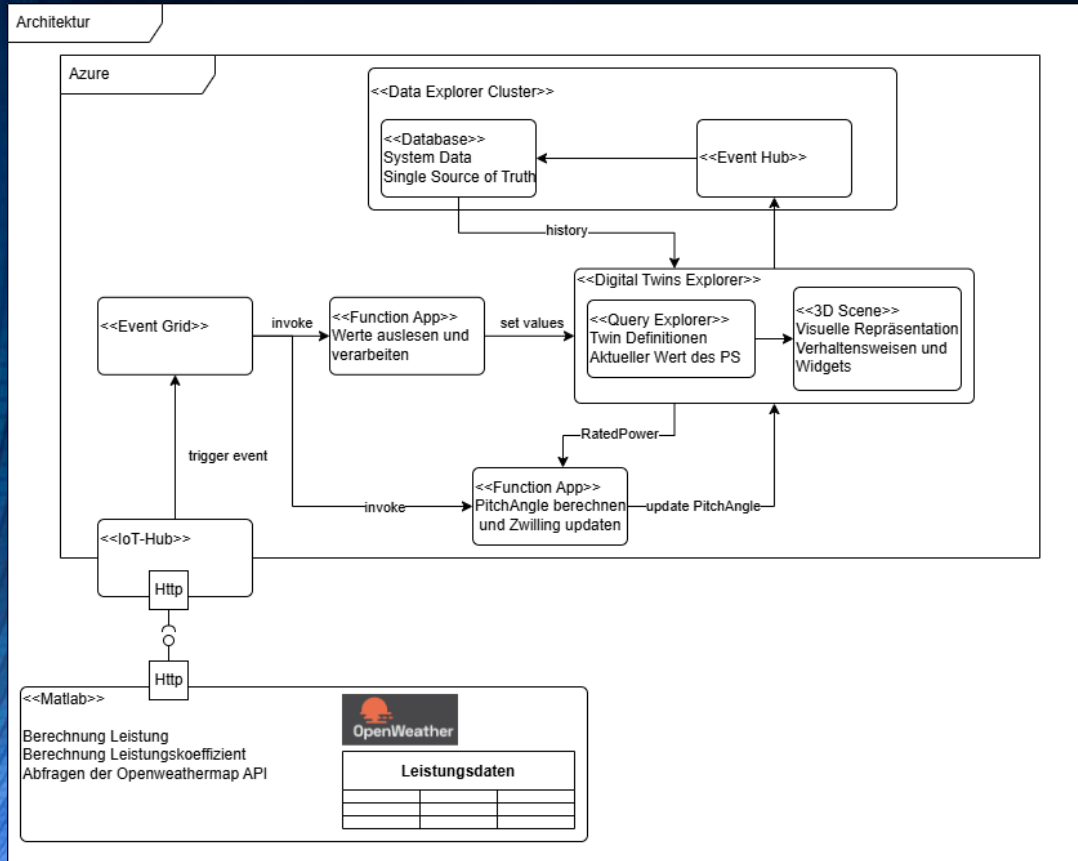


Communication Interface: Webserver

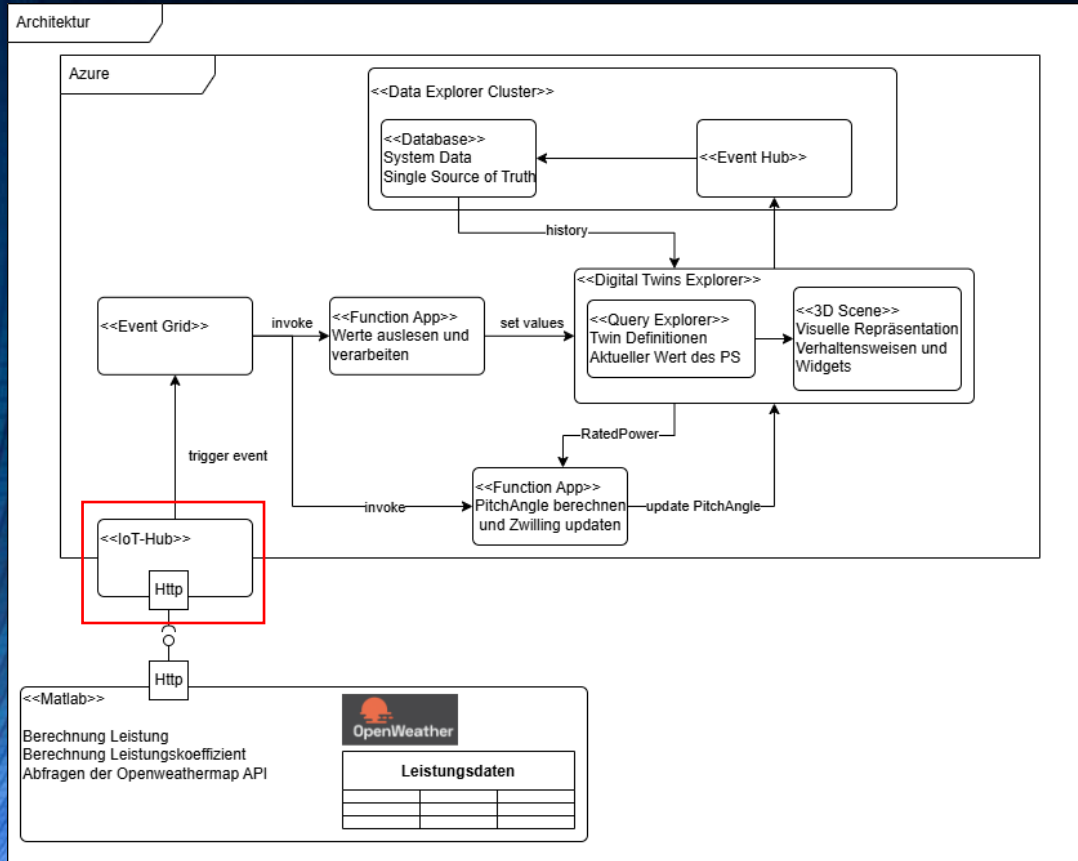
- Ein Webserver holt sich alle drei Minuten* die aktuelle Wetterlage in Dingle, Ireland über die API
- Anhand der Wetterdaten werden die Leistungsdaten interpoliert für präzisere Daten
- Der Luftdruck wird berechnet
- Der Webserver sendet alle drei Minuten Sensor- und interpolierte Leistungsdaten per http-Request an das Azure IoT-Hub

*geschuldet der teuren Komponenten in Azure und Limitationen in der Openweathermap API, sonst wäre Echtzeit möglich.

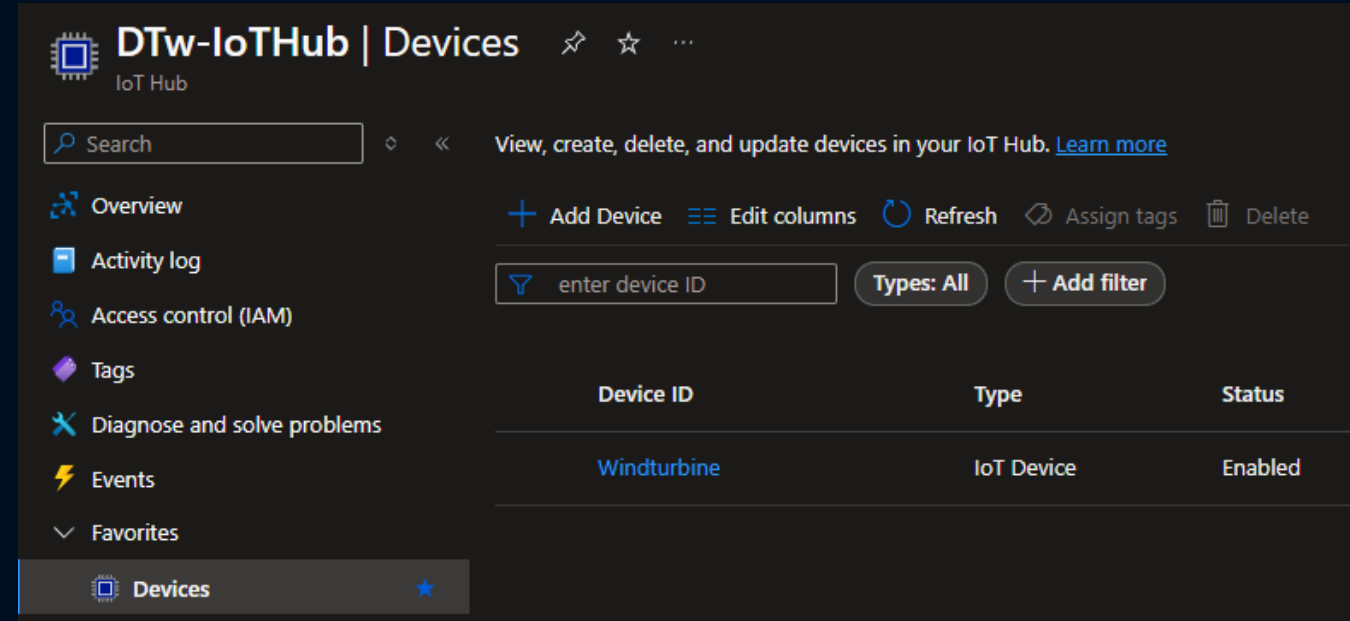
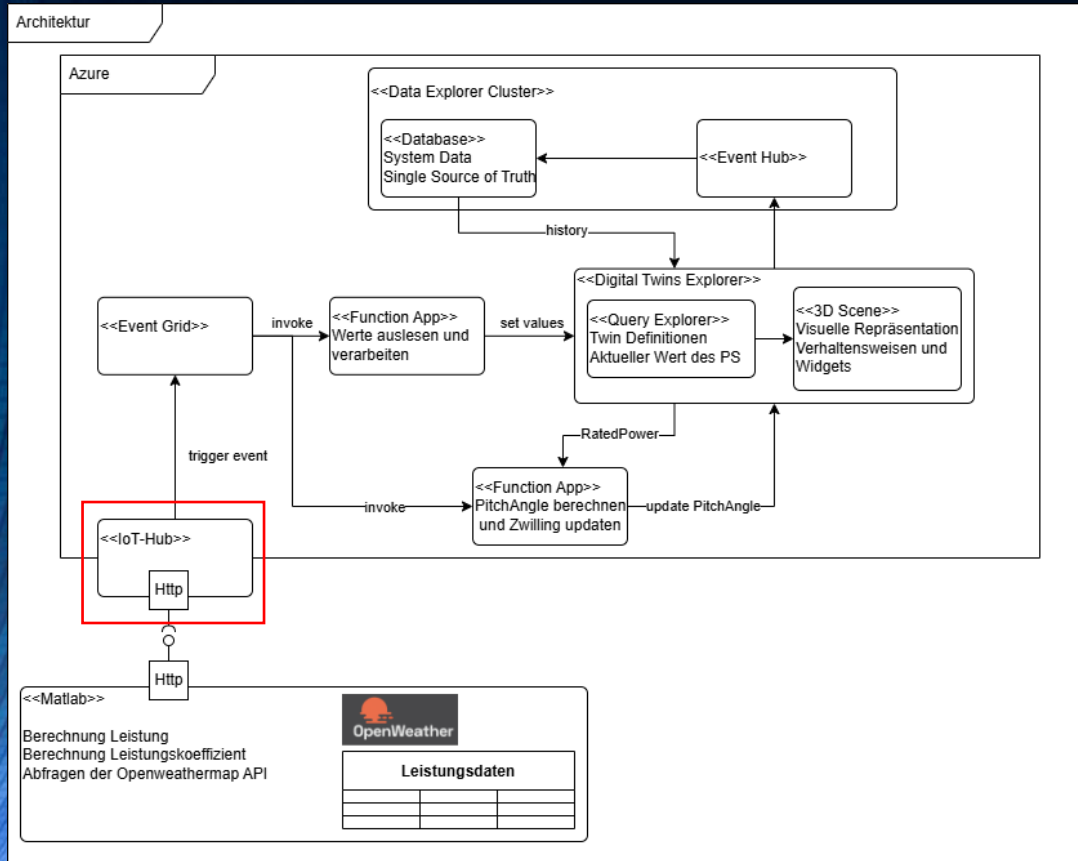
Communication Interface: IoT Hub/ Function App



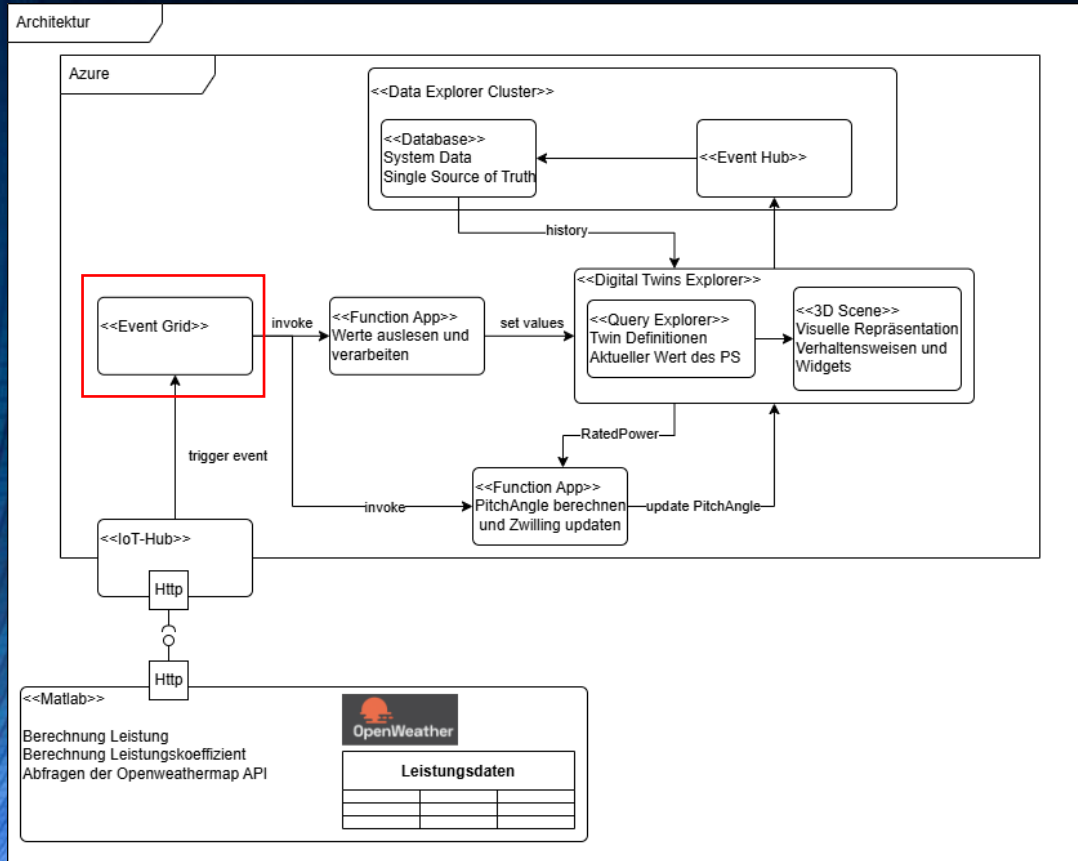
Communication Interface: IoT Hub/ Function App





Communication Interface: IoT Hub/ Function App

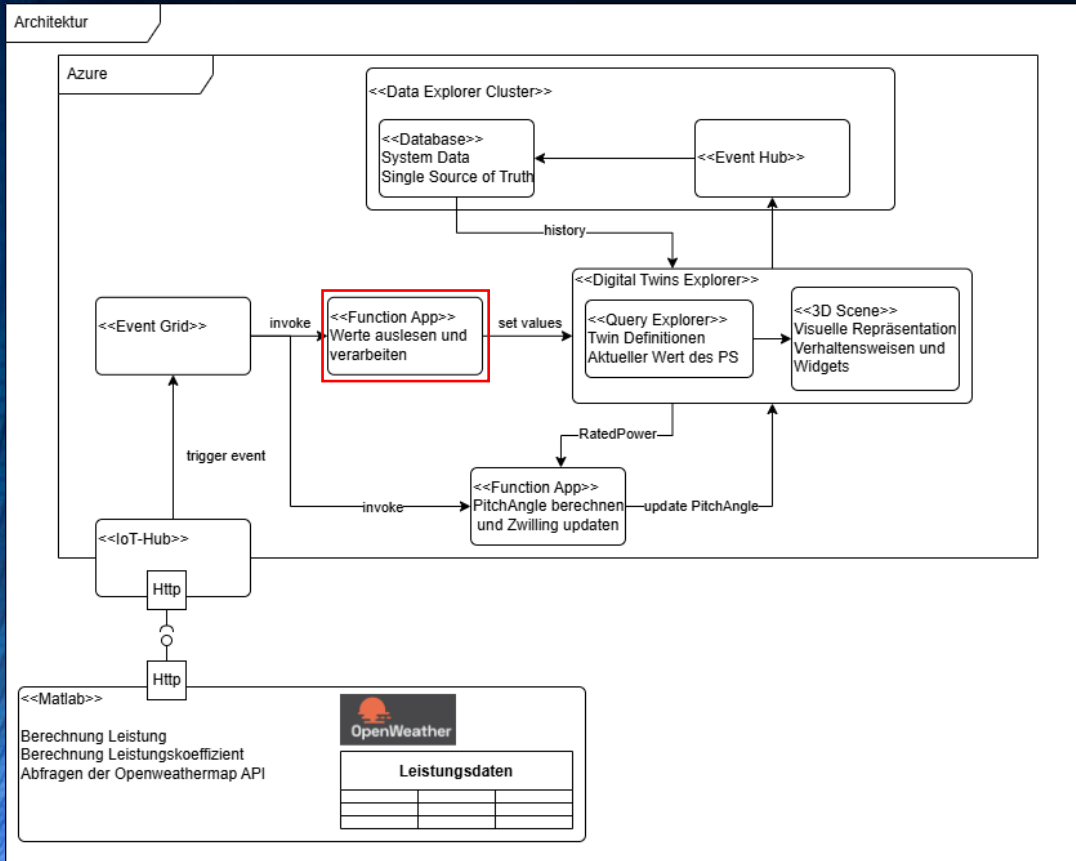


Communication Interface: IoT Hub/ Function App



Name	Endpoint
 adt-event-subscription	AzureFunction
 calculation	AzureFunction

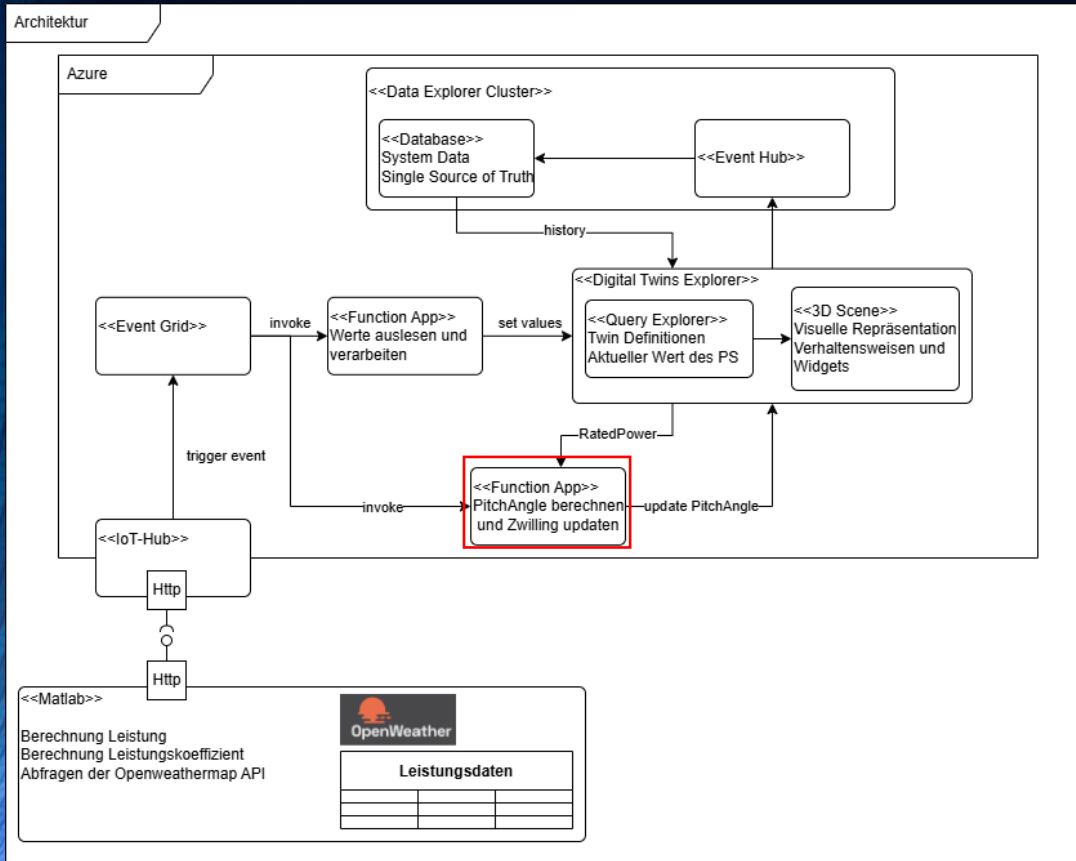
Communication Interface: IoT Hub/ Function App



```

{
  "properties": {
  },
  "systemProperties": {
    "iothub-content-type": "application/json; charset=UTF-8",
    "iothub-connection-device-id": "Windturbine",
    "iothub-connection-auth-
method": "{u0022scope\u0022:\u0022device\u0022,\u0022type\u0022:\u0022sas\u0022,\u0022iss
uer\u0022:\u0022iothub\u0022}",
    "iothub-connection-auth-generation-id": "638676288583917894",
    "iothub-enqueuedtime": "2025-01-06T10:25:47.409Z",
    "iothub-message-source": "Telemetry"
  },
  "body": {
    "Name": "AN_Bonus_1000_54",
    "RotorArea": 2300,
    "MaxRPM": 22,
    "CutInSpeed": 3,
    "RatedSpeed": 15,
    "CutOutSpeed": 25,
    "RatedPower": 1000
  }
}
  
```

Communication Interface: IoT Hub/ Function App

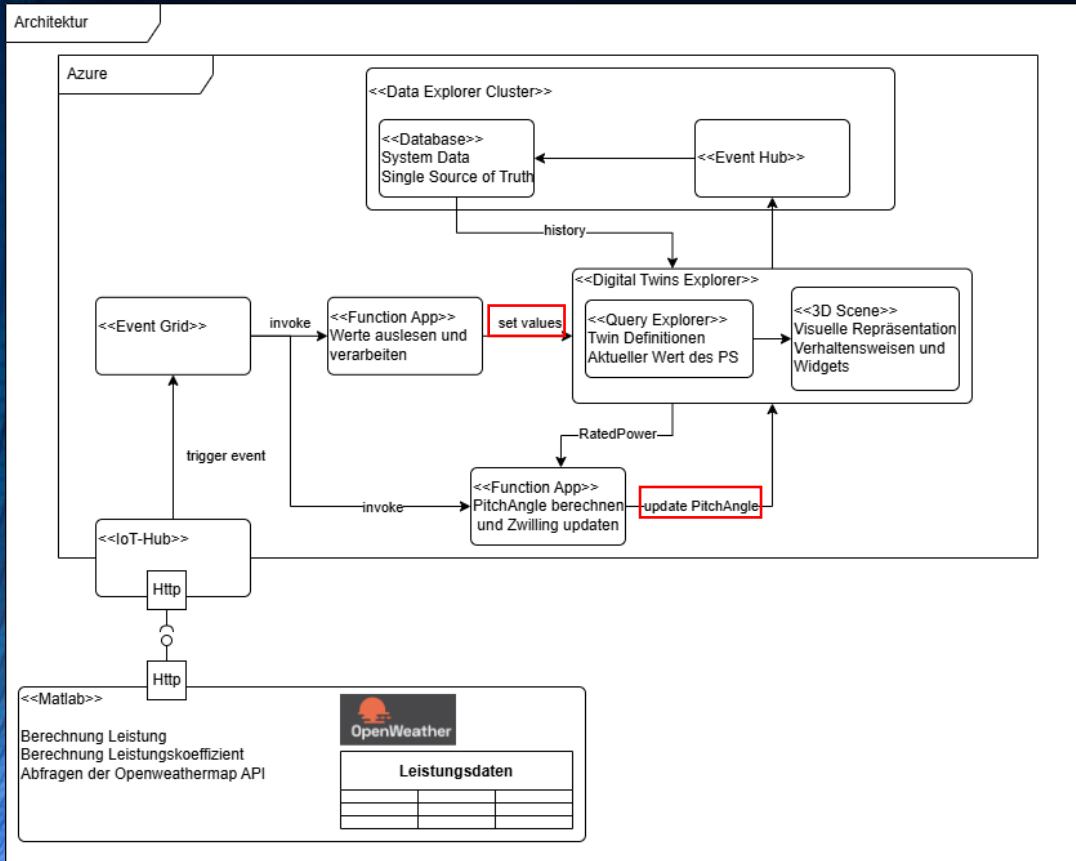


```

{
  "properties":{
  },
  "systemProperties":{
    "iothub-content-type":"application/json; charset=UTF-8",
    "iothub-connection-device-id":"Windturbine",
    "iothub-connection-auth-
method":"{\u0022scope\u0022:\u0022device\u0022,\u0022type\u0022:\u0022sas\u0022,\u0022iss
uer\u0022:\u0022iothub\u0022}",
    "iothub-connection-auth-generation-id":"638676288583917894",
    "iothub-enqueuedtime":"2025-01-06T10:25:47.784Z",
    "iothub-message-source":"Telemetry"
  },
  "body":{
    "Kilowatts":827.06,
    "Cp":0.365
  }
}

```

Communication Interface: IoT Hub/ Function App



Nuget: Azure.DigitalTwins.Core

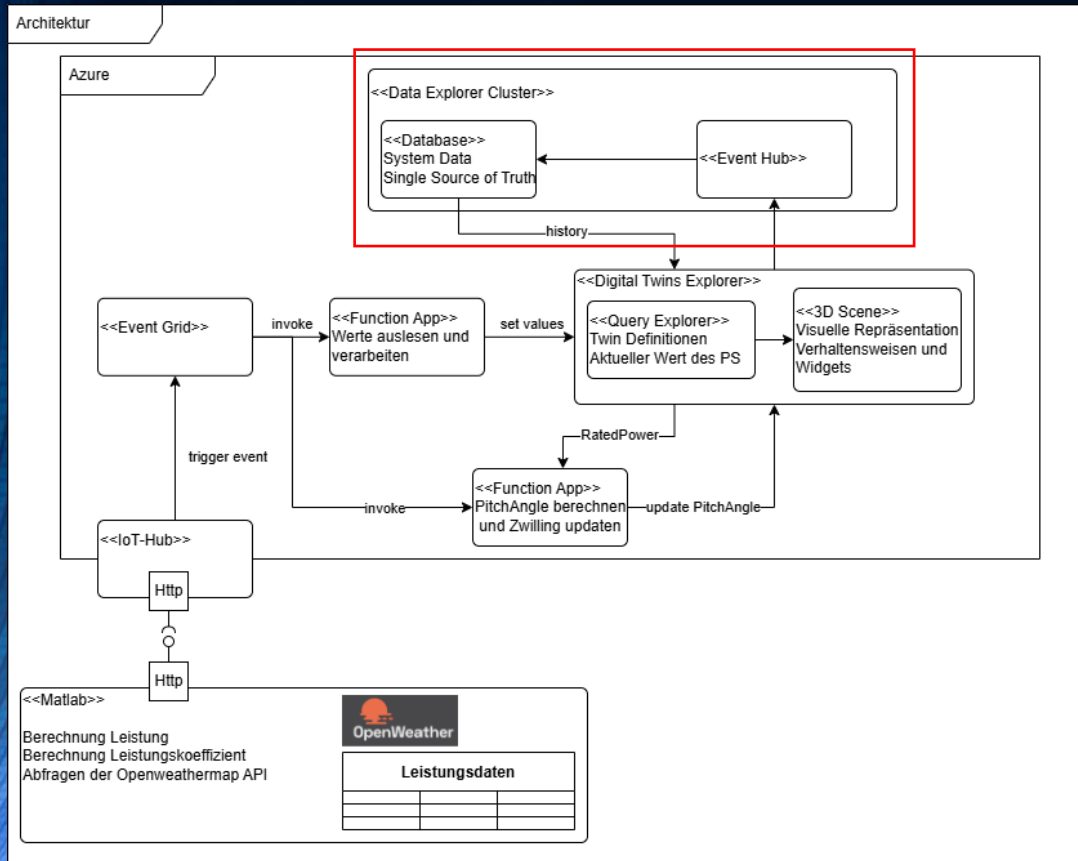
```
var response = await client.UpdateDigitalTwinAsync(deviceId, updateTwinData);
```

	pitchAngle /subscriptions/6332ee0a-68b7-442a-9a49-f6d8b24d5d92/resourc...	App Service or Function App	Azure Digital Twins Data Owner ⓘ
	windturbineingest /subscriptions/6332ee0a-68b7-442a-9a49-f6d8b24d5d92/resourc...	App Service or Function App	Azure Digital Twins Data Owner ⓘ

Lucas Weiss, Raphael Wudy - DTW Technische Hochschule Rosenheim

19.02.2025

System Data: Azure Data Explorer



Connection Details	
Status	Succeeded
Message time stamp	2024-12-15T09:36:53.9676361Z
Authentication	
Managed identity	N/A
Event hub	
Subscription	6332ee0a-68b7-442a-9a49-f6d8b24d5d92
Event hub namespace	DTw-EventHub
Event hub	dtw-hub
Consumer Group	\$Default
Azure Data Explorer	
Subscription	6332ee0a-68b7-442a-9a49-f6d8b24d5d92
Azure Data Explorer cluster	N/A
Database	
Property event table name	AdtPropertyEvents
Record twin / relationship property and item removals	Yes
Twin lifecycle event table name	Lifecycles
Relationship lifecycle event table name	Relationships

Virtuelles System

Live Demo

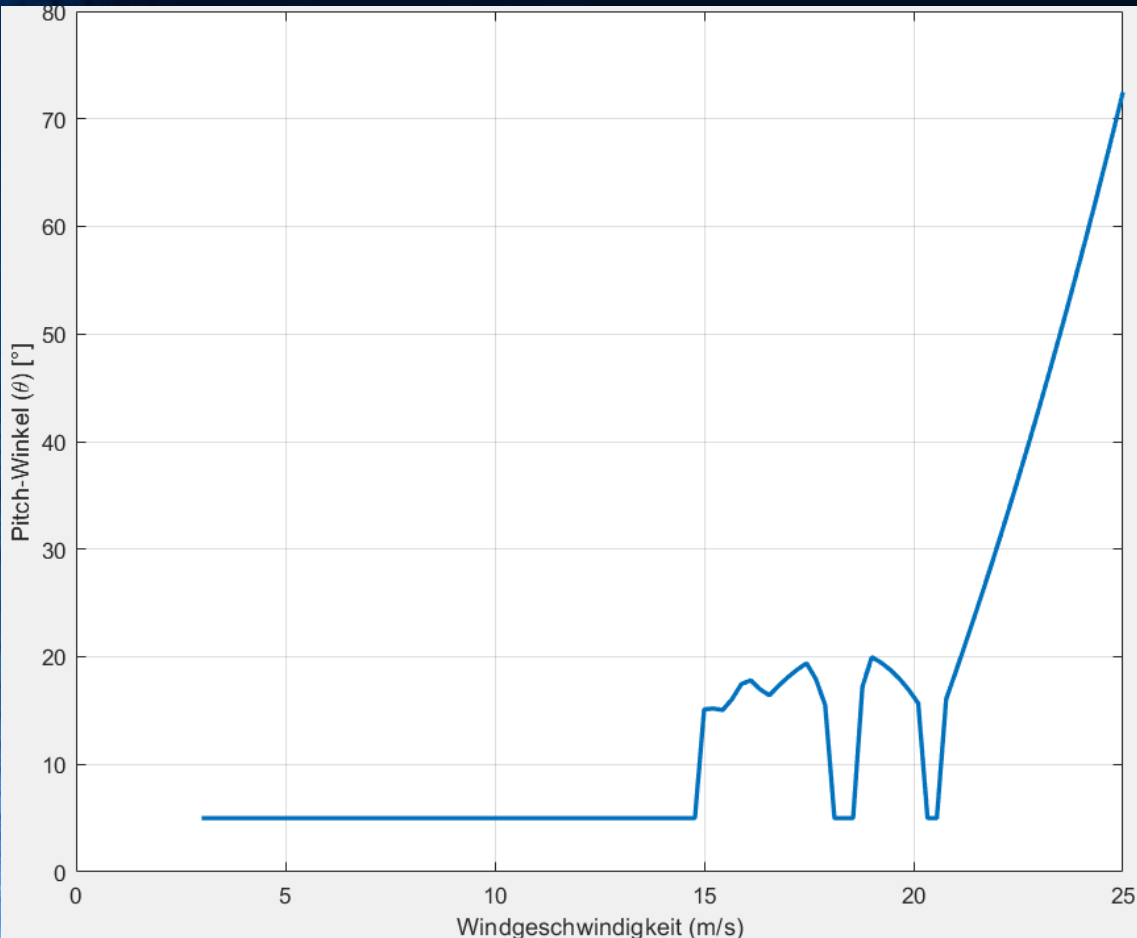
Services: Berechnungen und Simulationen

- Berechnung der Rotorblattstellung (PitchAngle)

- $$\theta(v_{wind}, C_p) = \left\{ \begin{array}{l} \theta_{minimal}, P_{turbine} < P_{rated} \\ \theta_{maximal}, P_{turbine} \geq P_{rated} \\ \text{Linearer Verlauf, für den Übergang} \end{array} \right\}$$

- $P_{turbine}$ = Leistung der Turbine aktueller Wert
- P_{rated} = Nennleistung der Turbine in diesem Wall 1000 Kilowatt

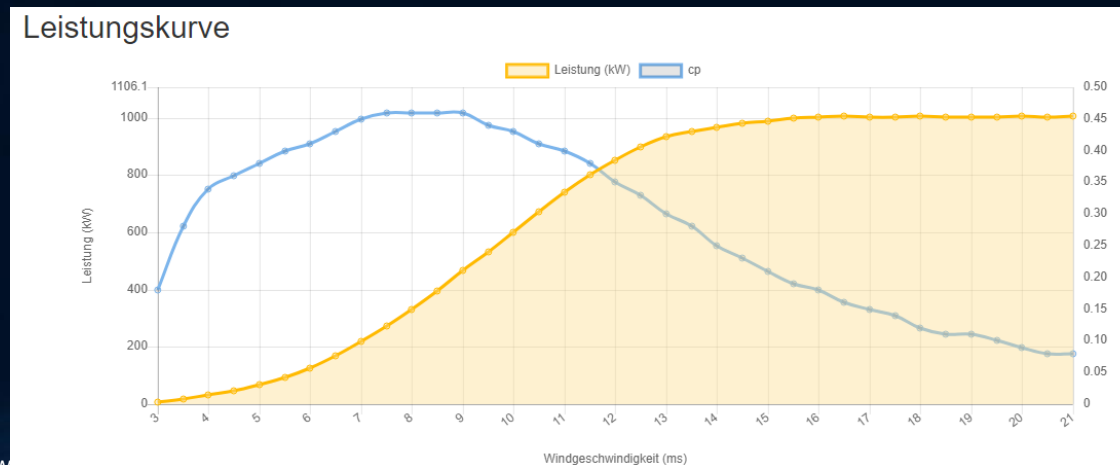
Services: Berechnungen und Simulationen



- Einhalten der Bedingungen:
- $P_{turbine} < P_{rated} : \theta = 5$
- $P_{turbine} \geq P_{rated} : \theta = 15 + \frac{P_{turbine} - P_{rated}}{P_{rated}} * 75$
Limit $\theta = 90$
- Lin. Verlauf : $\theta = 5 + \frac{P_{turbine}}{P_{rated}} * 10$

Services: Berechnungen und Simulationen

- Mit historischen Leistungsdaten ist eine lineare Ausgleichsrechnung möglich für C_p = Leistungskoeffizient
- Daten erzeugen 37x5 Matrix in Bezug auf v = Windgeschwindigkeit
- Modellfunktion Polynom 4. Grades:
$$C_p(v; k_0, k_1, k_2, k_3, k_4) = k_0 v^4 + k_1 v^3 + k_2 v^2 + k_3 v + k_4$$



Services: Berechnungen und Simulationen

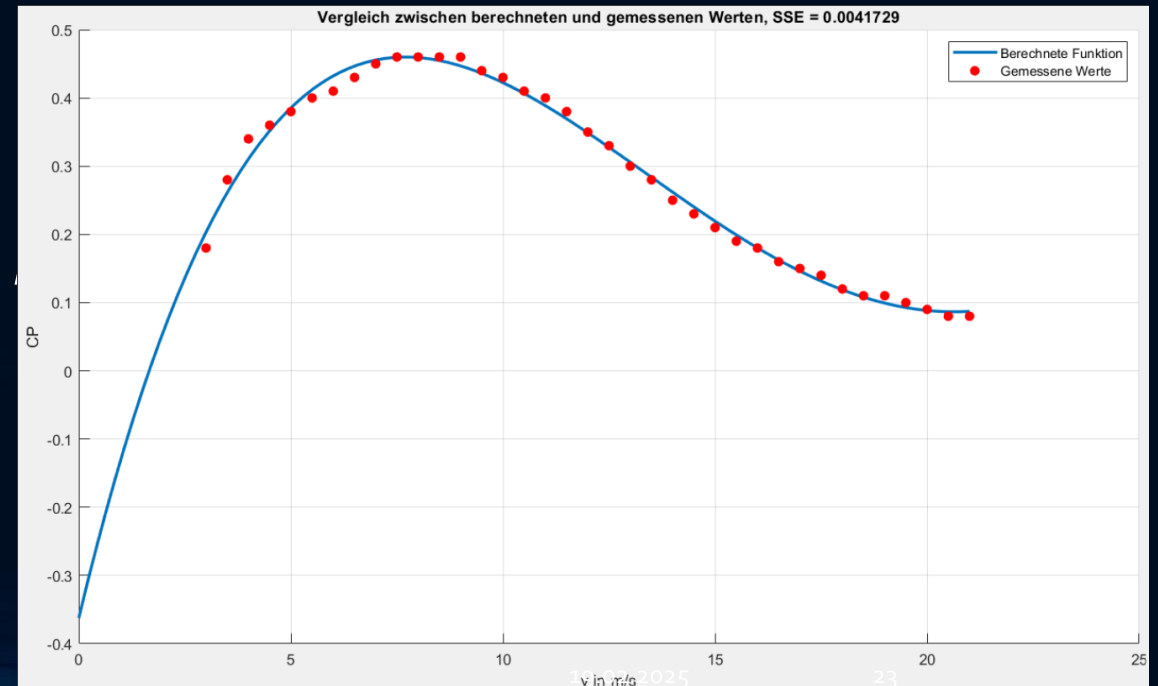
- Durch die polyfit Funktion in Matlab resultiert, unter den angegebenen Daten, die optimale Anpassung:

$$C_p(v) = -1.0588 * 10^{-5} v^4 + 9.4546 * 10^{-4} v^3 - 0.026546 v^2 + 0.25998 v - 0.36238$$

- Fehlersumme = 0.0041729
- Mit der Formel der Windkraft

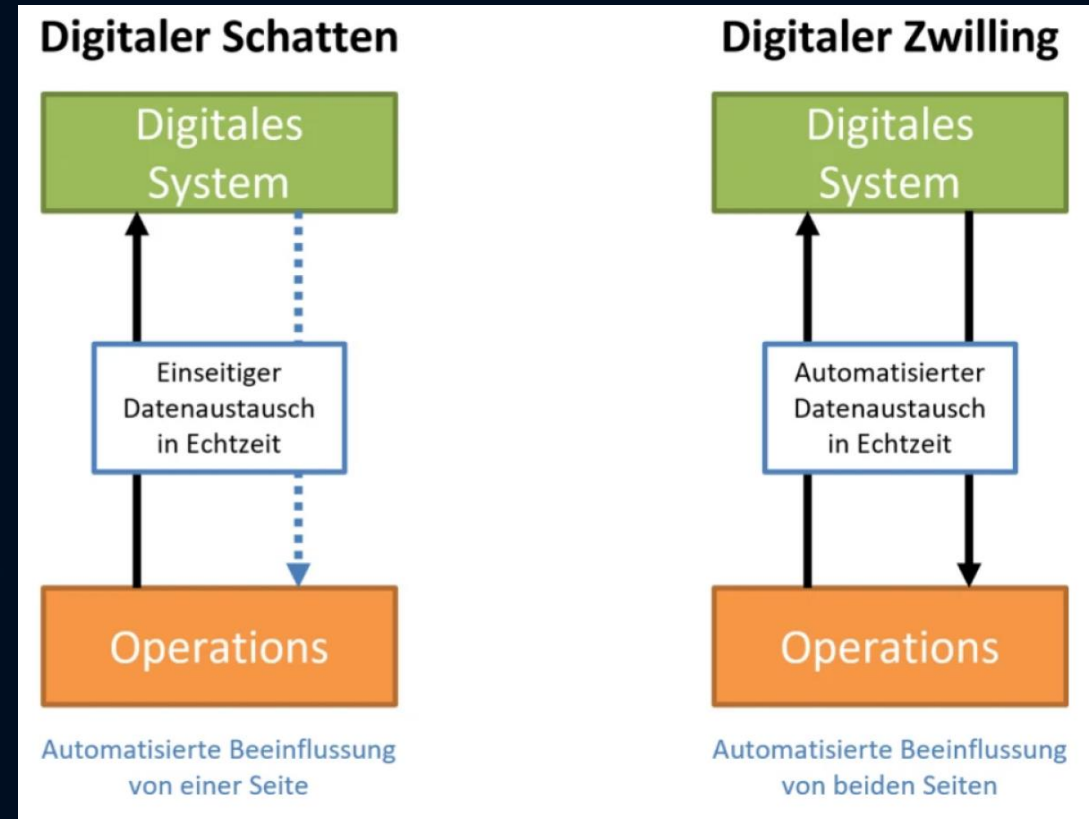
$$P_{wind} = \left(\frac{1}{2}\right) * \rho * A * v^3 * C_p$$

und C_p kann die Leistung der Windturbine in Watt berechnet werden



Vom Digitalen Schatten zum Zwilling

- Daten kommen vom einem simulierten PS basierend auf historischen Daten
- Berechnungen in der digitalen Welt finden statt
- Fehlende Hardware auf PS-Seite (Rotorblätter)
- Kommunikation über http auf MQTT umstellen



Fazit

- ↓ Ohne Vorkenntnisse schwierig in der vorgegebenen Zeit, weil hoher Konfigurationsaufwand in Azure samt Berechtigungen, komplexe Steuerung
- ↓ „Versteckte Kosten“ für volles Potential des DTs
- ↓ Fehlersuche nicht einfach, da Debugging der Function Apps = try and error mit nicht aussagekräftigen Fehlermeldungen
- ↑ Einfaches Erstellen von 3D-Szenen und Anbinden der hinterlegten DT-Modelle
- ↑ Anbinden des PS an das IoT Hub und updaten der Daten im DT
- ↑ Man hat schnell was greifbares erstellt