Networking Academy

it bildungsnetz

Akademie für Lehrerfortbildung und Personalführung

Cisco Virtual Partner Technical eXchange

„Vom Sensor in die Cloud“ – Eine mögliche Umsetzung der neuen Lernfelder (LF 7/8/9) der IT-Berufe

Michael Feike, Berufsschule Fürth

20. Nationaler Akademietag der Bildungsinitiative Networking
22./23. April 2021

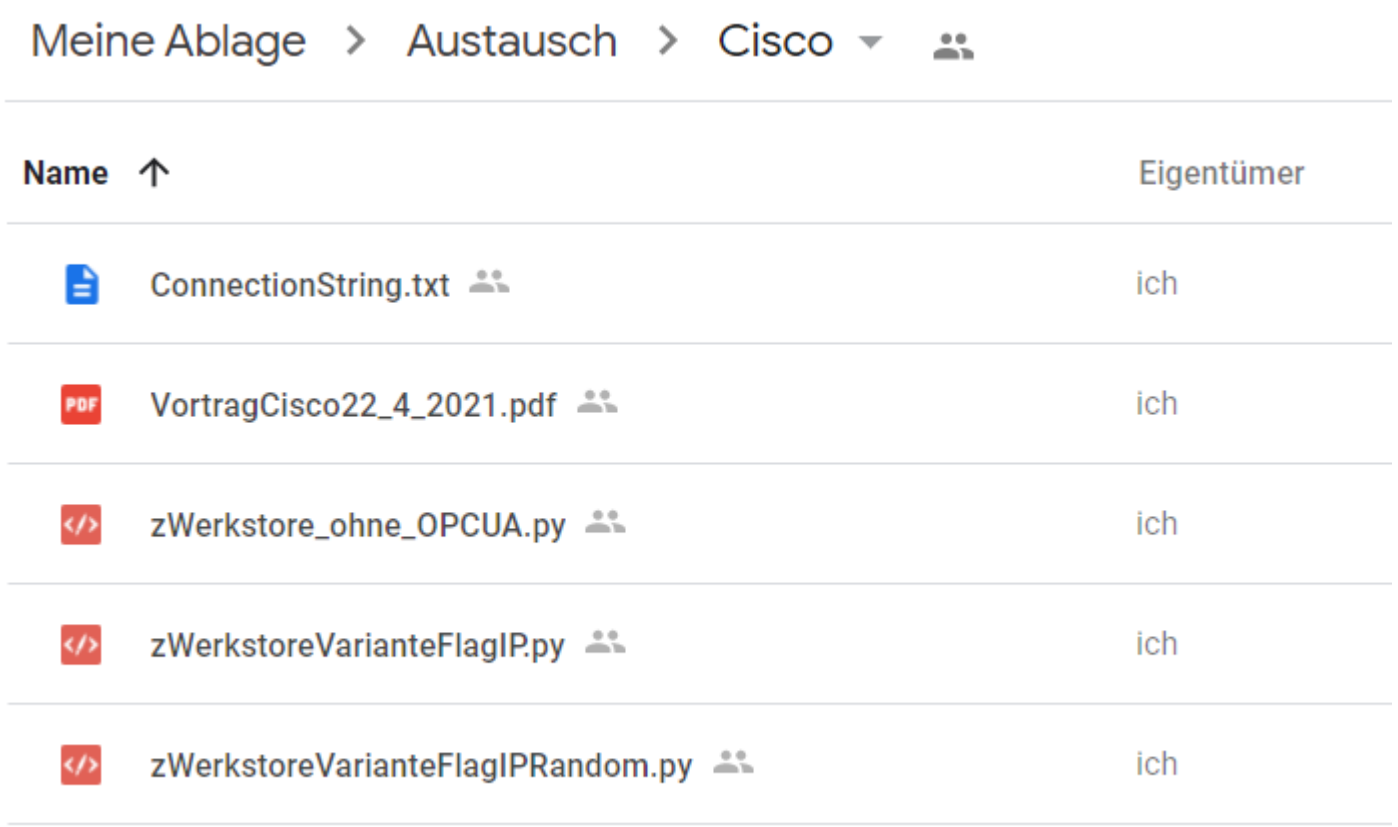
#CiscoPTX20

Michael Feike

mfeike@b3-fuerth.de

bis 2001 Berufsfachschule für Informatik Roth
bis 2017 Seminarlehrer für Informatik/IT-Technik Fürth
seit 2016 IT – Abteilungsleiter (**Ausbildung Fachinformatiker**)
seit Ende 2016 Mitglied und Dozent der Fachgruppe „Datenkommunikation“ der Akademie in Dillingen (verantwortlich **Digitale Transformation**/Wirtschaft 4.0 **M. Lotter**)

Unterlagen: ibit.ly/UTeJ



Lernfeld 7 **72 Std.**

Cyber-physische Systeme ergänzen

Zielformulierung

Die Schülerinnen und Schüler verfügen über die Kompetenz, die physische Welt und IT-Systeme funktional zu einem cyber-physischen System zusammenzuführen.

Die Schülerinnen und Schüler **analysieren** ein cyber-physisches System bezüglich eines Kundenauftrags zur Ergänzung und Inbetriebnahme weiterer Komponenten.

Sie **informieren** sich über den Datenfluss an der Schnittstelle zwischen physischer Welt und IT-System sowie über die Kommunikation in einem bestehenden Netzwerk. Sie verschaffen sich einen Überblick über die Energie-, Stoff- und Informationsflüsse aller am System beteiligten Geräte und Betriebsmittel.

Die Schülerinnen und Schüler **planen** die Umsetzung des Kundenwunsches, indem sie Kriterien für die Auswahl von Energieversorgung, Hardware und Software (*Bibliotheken, Protokolle*) aufstellen. Dazu nutzen sie Unterlagen der technischen Kommunikation und passen diese an.

Sie **führen** Komponenten mit dem cyber-physischen System funktional **zusammen**.

Sie **prüfen** systematisch die Funktion, messen physikalische Betriebswerte, validieren den Energiebedarf und protokollieren die Ergebnisse.

Die Schülerinnen und Schüler **reflektieren** den Arbeitsprozess hinsichtlich möglicher Optimierungen und diskutieren das Ergebnis in Bezug auf Betriebssicherheit und Datensicherheit.

→**Verbund von** mechatronische Komponenten (z.B. eine Produktionsstation einer Fertigung) oder **Teilsystemen (Raspberry PI)** über Netzwerke. Dieser ermöglicht die Steuerung und die Kontrolle von Systemen und Infrastrukturen.

Lernfeld 8
Daten systemübergreifend bereitstellen
(→ Richtung Datenbanken)

→ Datenquelle … auswählen
→ Ortsunabhängige Implementierung
→ …

→lokale oder Cloud-basierte DBs
→MySQL (lokal z.B. xampp)
→MongoDB (lokal oder DBaaS)

Lernfeld 9
Netzwerke und Dienste bereitstellen

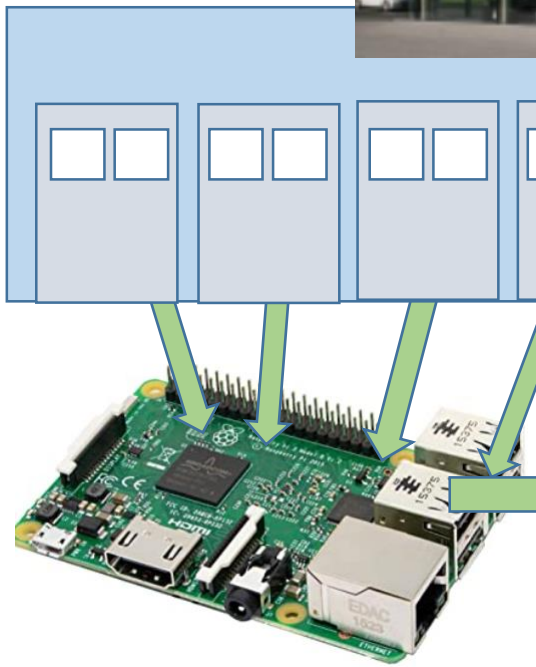
→Eigenschaften, Funktion, Leistungsmerkmale … von Netzwerkkomponenten
→NW-Dienste und Protokolle

→Netzwerkdienste, …
→realisiert über **Standards/ Protokolle z.B. MQTT, OPCUA, …**

„Vom Sensor in die Cloud→Cyberphysische Systeme“

Autohaus „Nettmann“

Kunde möchte Werkstatttore „überwachen“.



OPCUA, MQTT, …

OPC Router

TCP/IP-Socket
MongoDB Wire-Protokoll

DBaaS
NoSQL-Datenbank

mongo DB

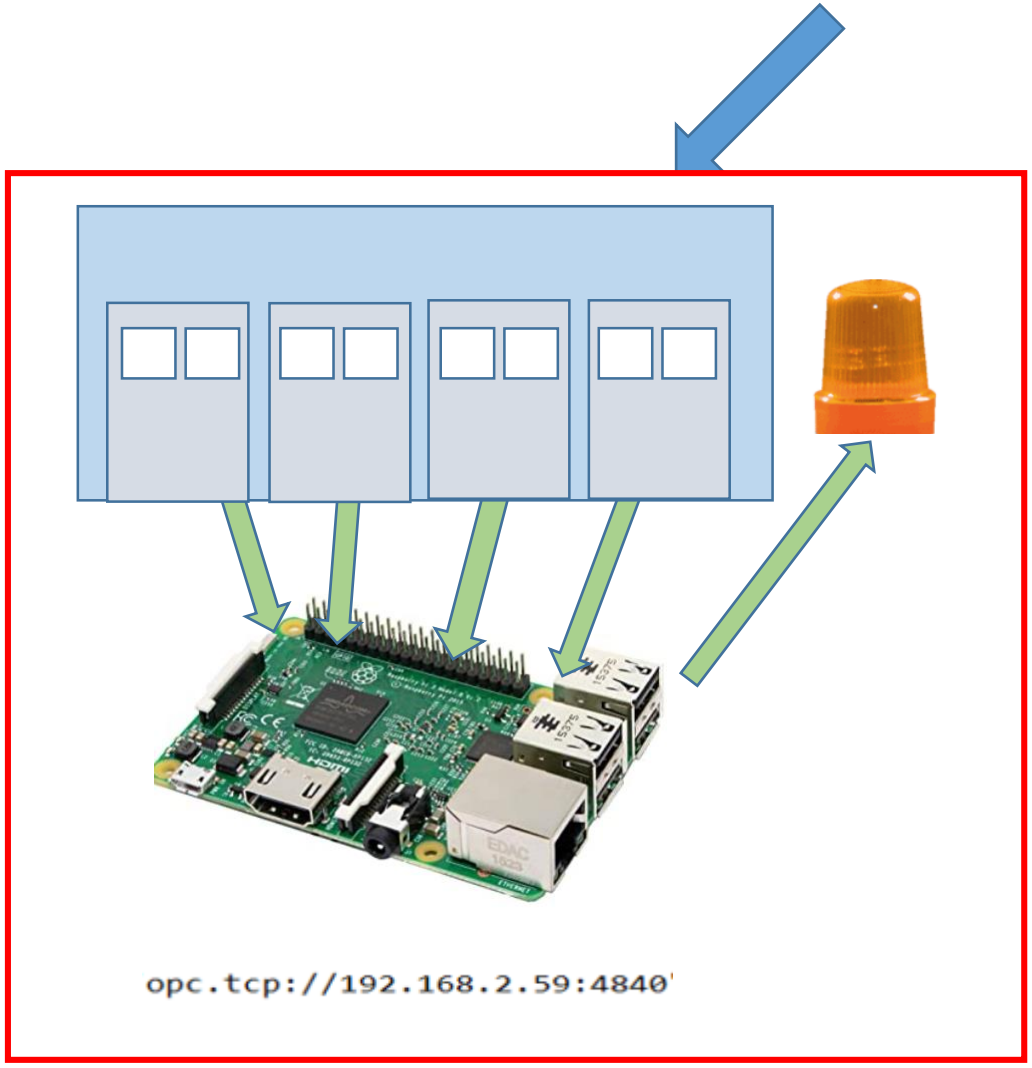
SQL Datenbank

Teilsysteme mit Sensoren oder **mechatronische Komponenten**

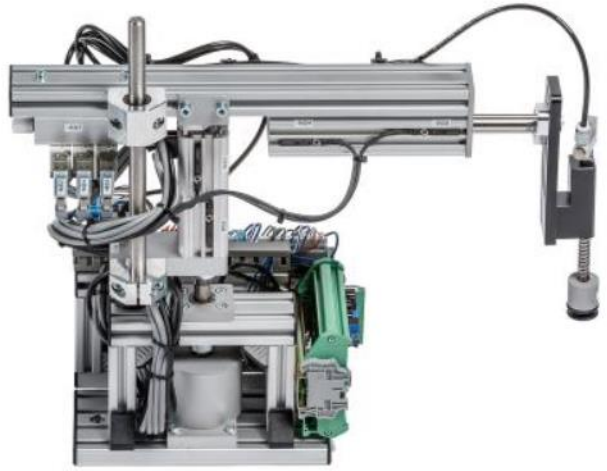
Middleware-Software

Datenbanken
(Zielanwendung in der vertikalen Kommunikation)

1. Teilsystem mit Sensoren/Aktoren oder **mechatronische Komponente**

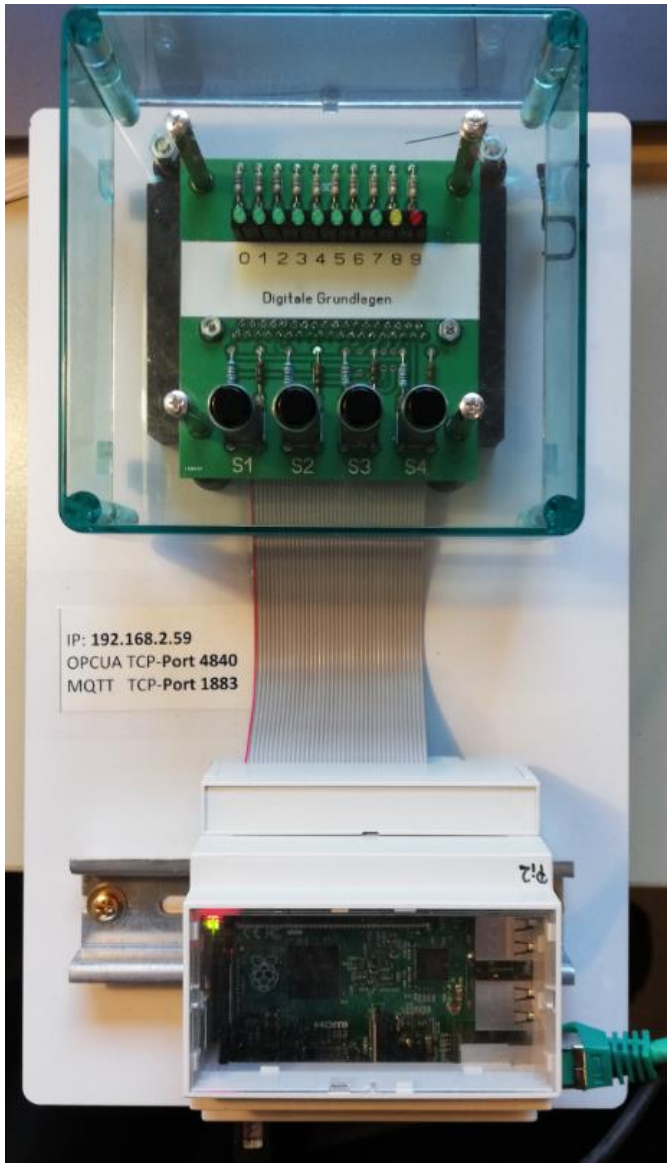


Pick and Place …

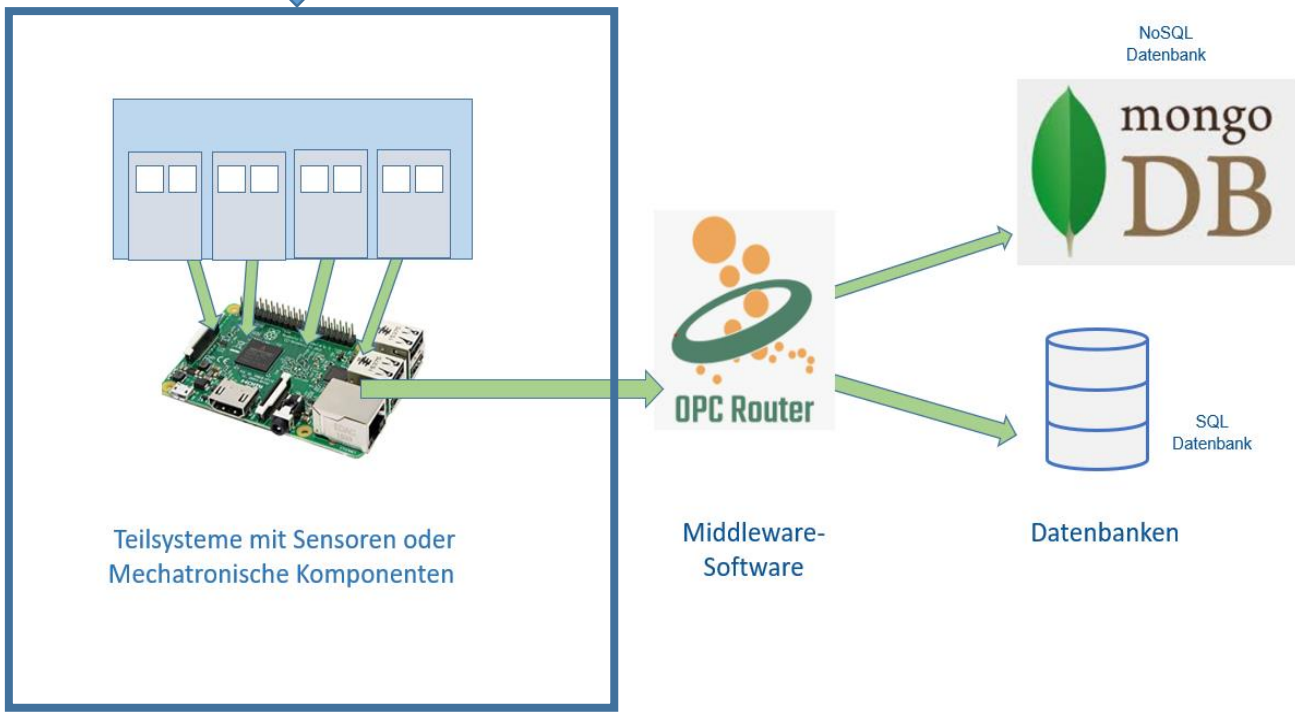


Förderband …

1. **Teilsystem** mit Sensoren/Aktoren oder **mechatronische Komponente**



Rasperry Pi mit 4 Tastern, über die der Zustand der Tore (offen/geschlossen) erfasst werden kann



1. Teilsystem mit Sensoren/Aktoren oder **mechatronische Komponente**

1. Schritt: Ausgabe nur LEDs (Python)

Modul „time“ importieren, GPIOs konfigurieren , Zählweise BCM der Pins festlegen
GPIOs Pin 5,6,13,19 als Eingänge festlegen (Sensor, Schließer o. Taster)
GPIOs Pin 16,12,7,8 als Ausgänge festlegen (Aktor, LED)
Solange (true)

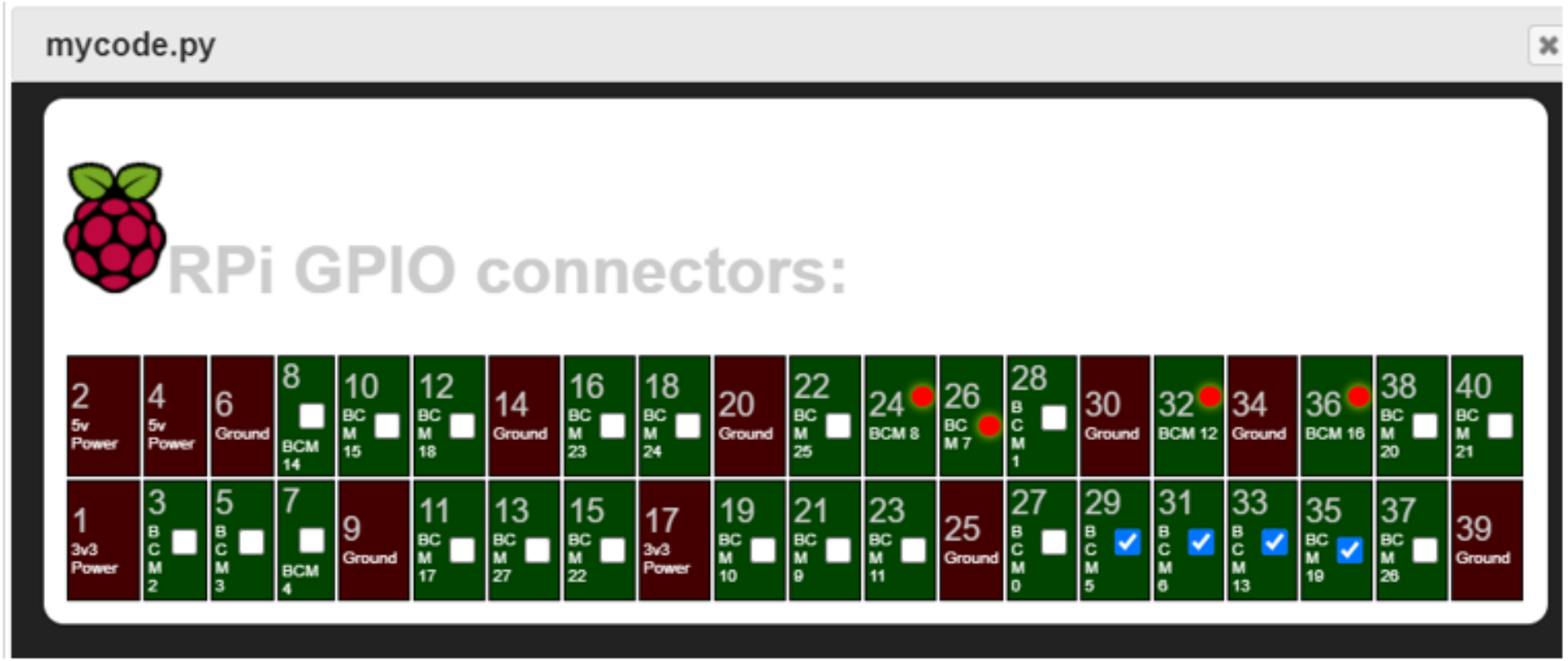
Taster an GPIO-Eingang 5 betätigt?	
LED GPIO Pin 16→ High (LED an)	LED GPIO Pin 16→ Low (LED aus)
Taster an GPIO-Eingang 6 betätigt?	
LED GPIO Pin 12→ High (LED an)	LED GPIO Pin 12→ Low (LED aus)
Taster an GPIO-Eingang 13 betätigt?	
LED GPIO Pin 7→ High (LED an)	LED GPIO Pin 7→ Low (LED aus)
Taster an GPIO-Eingang 19 betätigt?	
LED GPIO Pin 8→ Low (LED aus)	LED GPIO Pin 8→ Low (LED aus)
1 Sekunde warten	

Rasperry Pi mit 4 Tastern, über die der Zustand der Tore (offen/geschlossen) erfasst werden kann

Online-Editor für Raspberry PI

→Ausgabe der GPIOs

https://create.withcode.uk/python/A3



1. **Teilsystem** mit Sensoren (Tastern) und OPCUA – Standard zum Datenaustauch

Rasperry Pi mit 4 Tastern, über die der Zustand der Tore (offen/geschlossen) erfasst werden kann

2. Schritt: Ausgabe über einen OPCUA-Server (Python)

```
#Server Objekt anlegen und IP/Ports angeben
server=Server()
url="opc.tcp://192.168.2.59:4840"
server.set_endpoint(url)

#OPCUA Namensraum festleben
name="OPCUA_Serverraum"
addspace=server.register_namespace(name)
node= server.get_objects_node();

#Objekt Raspi im Namensraum festlegen
Raspi=node.add_object(addspace,"Raspi")


#Ordner Schalter für das Objekt Raspi anlegen
myfolder = Raspi.add_folder(addspace, "Schalter")

#OPUA Datenpunkte "Schalter1-4" festlegen und schreibar setzen
Schalter1 = myfolder.add_variable(addspace,"Schalter1",6.1, ua.VariantType.Boolean)
Schalter2 = myfolder.add_variable(addspace,"Schalter2",6.2, ua.VariantType.Boolean)
Schalter3 = myfolder.add_variable(addspace,"Schalter3",6.3, ua.VariantType.Boolean)
Schalter4 = myfolder.add_variable(addspace,"Schalter4",6.4, ua.VariantType.Boolean)
Time = node.add_variable(addspace,"Time",0)

Schalter1.set_writable()
Schalter2.set_writable()
Schalter3.set_writable()
Schalter4.set_writable()
Time.set_writable()

#OPCUA-Server starten
server.start()
```

Endpunkt

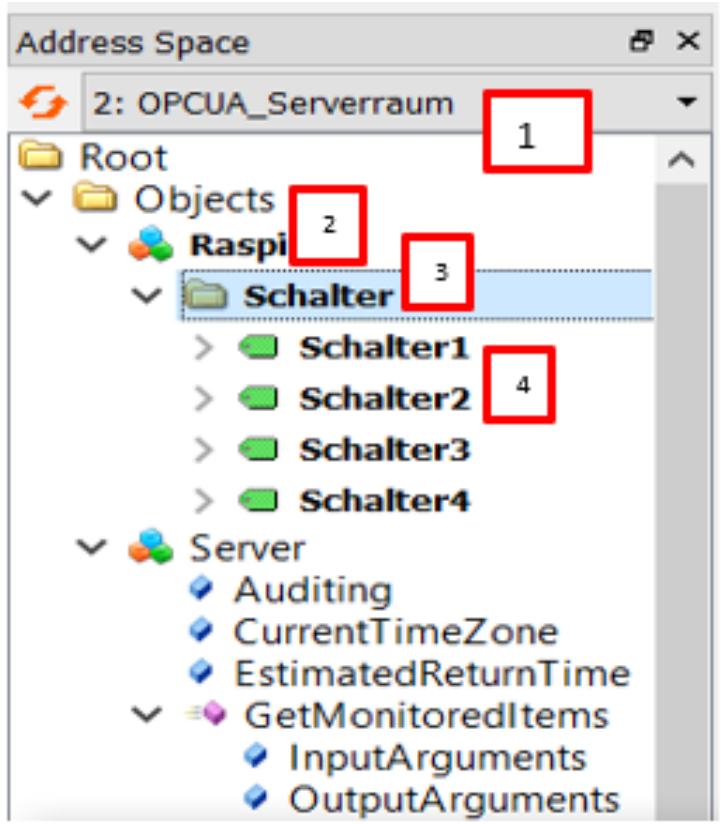
1

2

3

4

5



Ansicht im UA-Expert

Hinweis: Der UA-Expert ist ein Explorer für OPCUA-Objekte und OPCUA-Datenpunkte

1. **Teilsystem** mit Sensoren/Aktoren oder **mechatronische Komponente**

Rasperry Pi mit 4 Tastern, über die der Zustand der Tore (offen/geschlossen) erfasst werden kann

2. Schritt: Ausgabe über einen OPCUA-Server (Python)

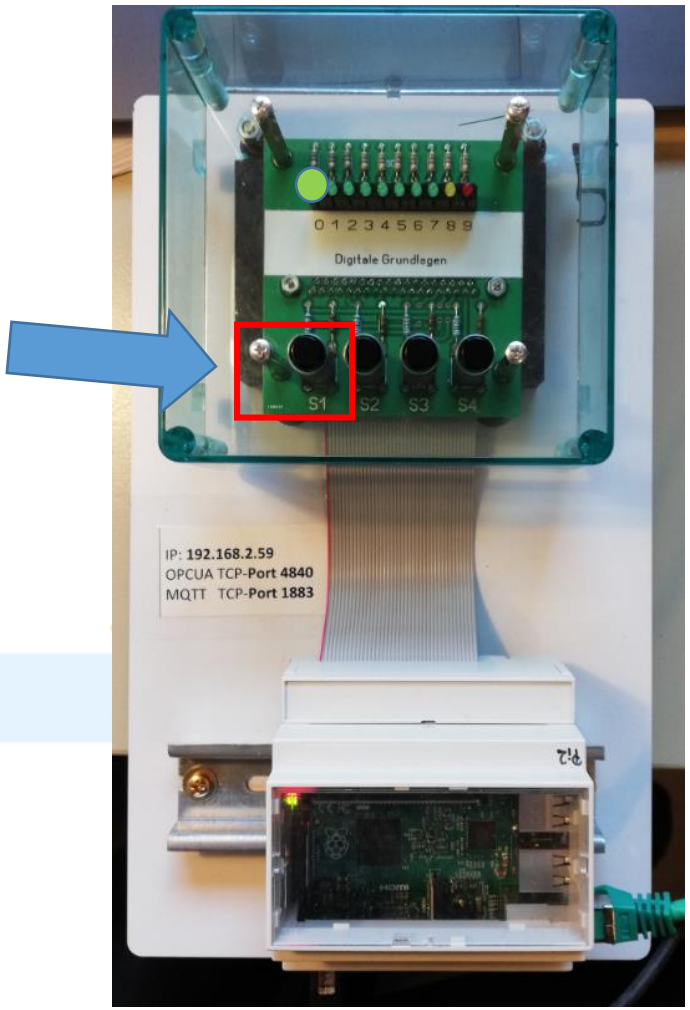
Durch Drücken des Schalters **S1=1** leuchtet die erste LED

Gleichzeitig:
Python OPCUA-Datenpunkt Schalter1 auf true setzen!

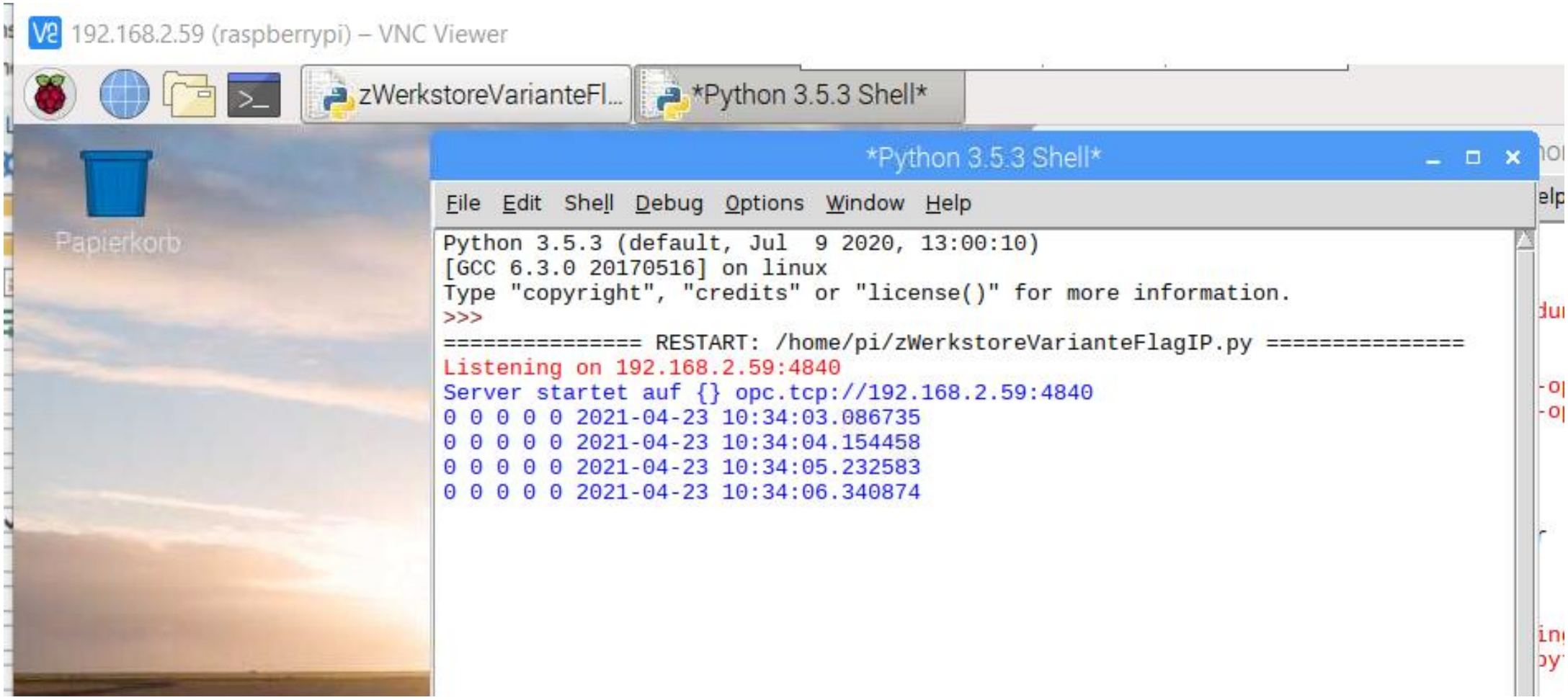
Schalter1.set_value(S1)

Python-Programme:

zWerkstore_ohne_OPCUA	12.11.2020 18:06	Python File
zWerkstoreVarianteFlagIP	10.02.2021 15:50	Python File
zWerkstoreVarianteFlagIPRandom	10.02.2021 15:50	Python File

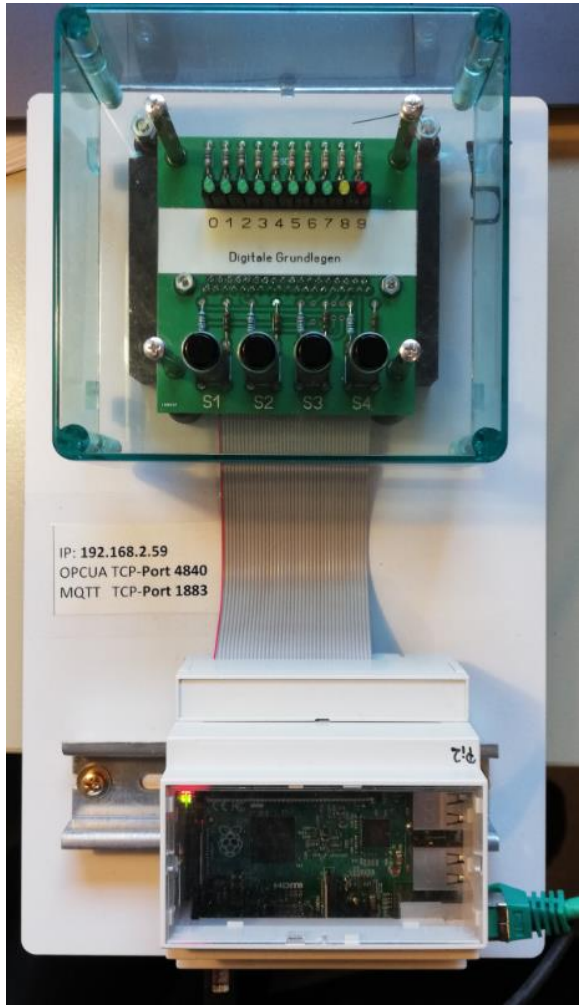


Hinweis: OPCUA-Datenpunkt „Flag“ zeigt jegliche Änderung (Changed) an. Die Version „Random“ liefert zufällige Schalterzustände und ist damit für die häusliche Schülerarbeit geeignet, da keine Schalter benötigt werden.


192.168.2.59 (raspberrypi) – VNC Viewer
zWerkstoreVarianteFl...
Python 3.5.3 Shell
Papierkorb
Python 3.5.3 Shell
File Edit Shell Debug Options Window Help
Python 3.5.3 (default, Jul 9 2020, 13:00:10)
[GCC 6.3.0 20170516] on linux
Type "copyright", "credits" or "license()" for more information.
>>>
========== RESTART: /home/pi/zWerkstoreVarianteFlagIP.py ==========
Listening on 192.168.2.59:4840
Server startet auf {} opc.tcp://192.168.2.59:4840
0 0 0 0 0 2021-04-23 10:34:03.086735
0 0 0 0 0 2021-04-23 10:34:04.154458
0 0 0 0 0 2021-04-23 10:34:05.232583
0 0 0 0 0 2021-04-23 10:34:06.340874

1. **Teilsystem** mit Sensoren/Aktoren oder **mechatronische Komponente**

Rasperry Pi mit 4 Tastern, über die der Zustand der Tore (offen/geschlossen) erfasst werden kann



3. Möglicher Schritt: Security hinzufügen (noch nicht realisiert)

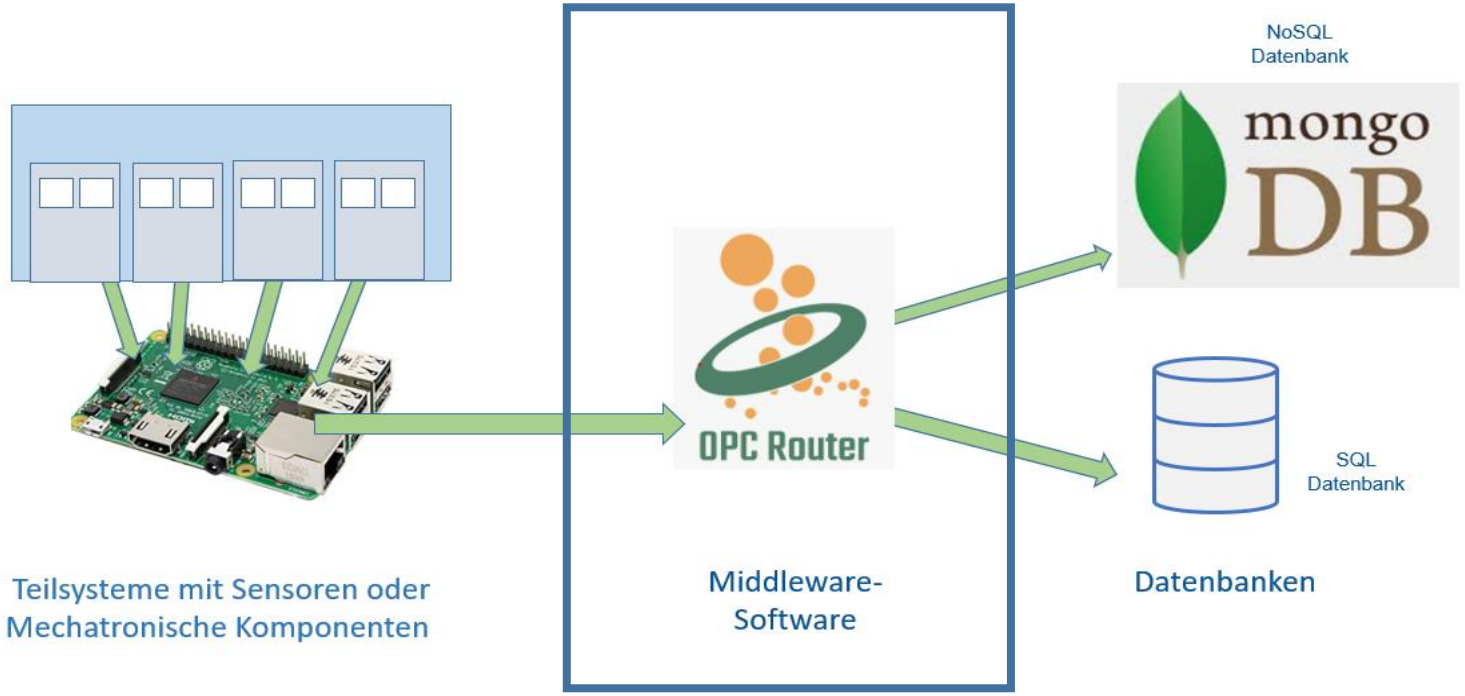
→Authentisierung
→verschlüsselte Datenübertragung
→Zertifikate
→VLAN
→…

2. Middleware – Software

z.B. OPC Router von Inray

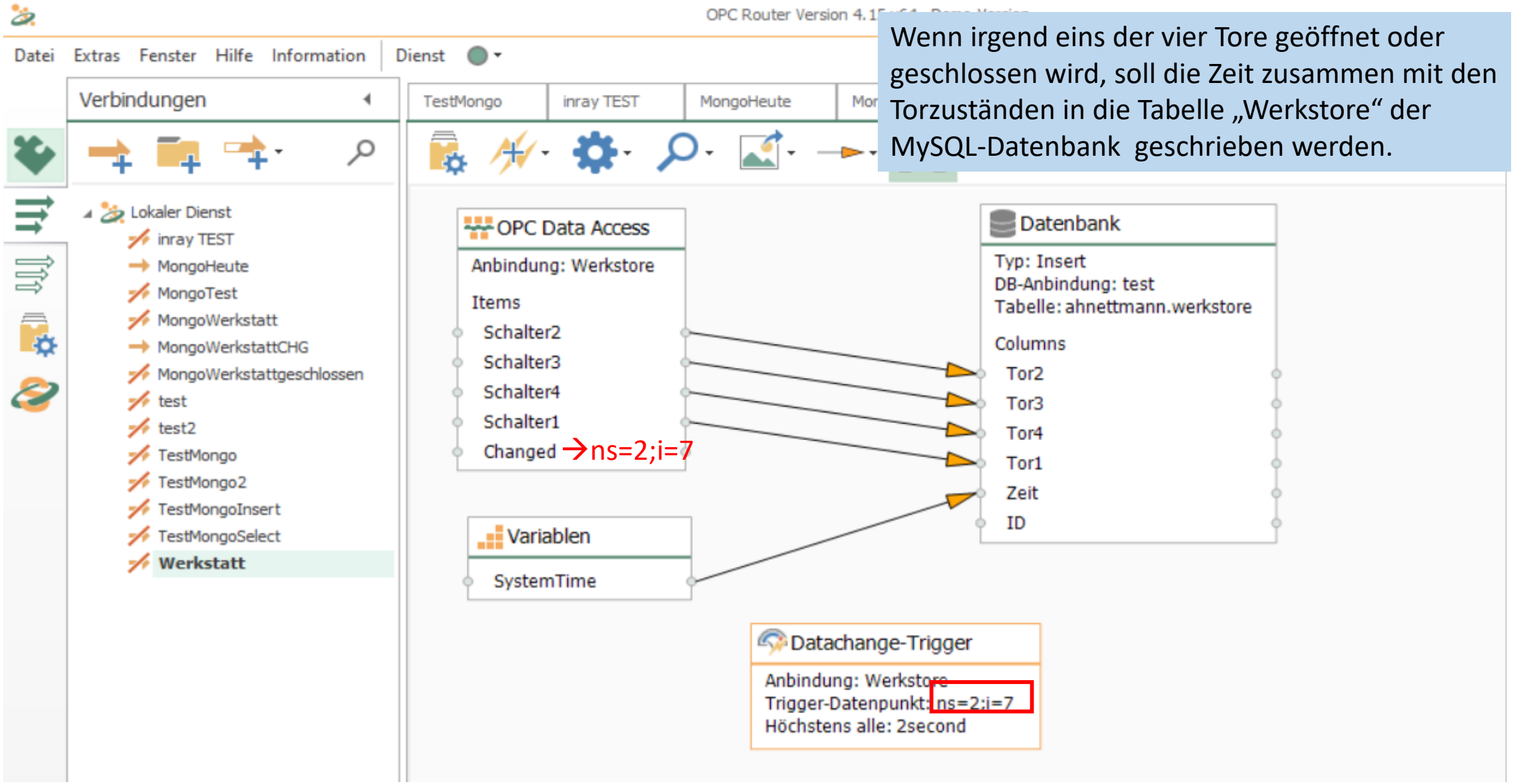
Anwendungsneutrale Programme, die zwischen Prozessen vermitteln. Hier z.B. zwischen einem OPCUA-Server und zwei Datenbanken

Hauptvorteil für die Nutzer von Middleware ist der Wegfall der komplexen Programmierung. Der Fokus kann mehr auf Planungs- und Konfigurationsaufgaben bei Schülern gelegt werden. Eine nicht zu komplexe Middleware-Software ist der **OPC-Router von Inray**.

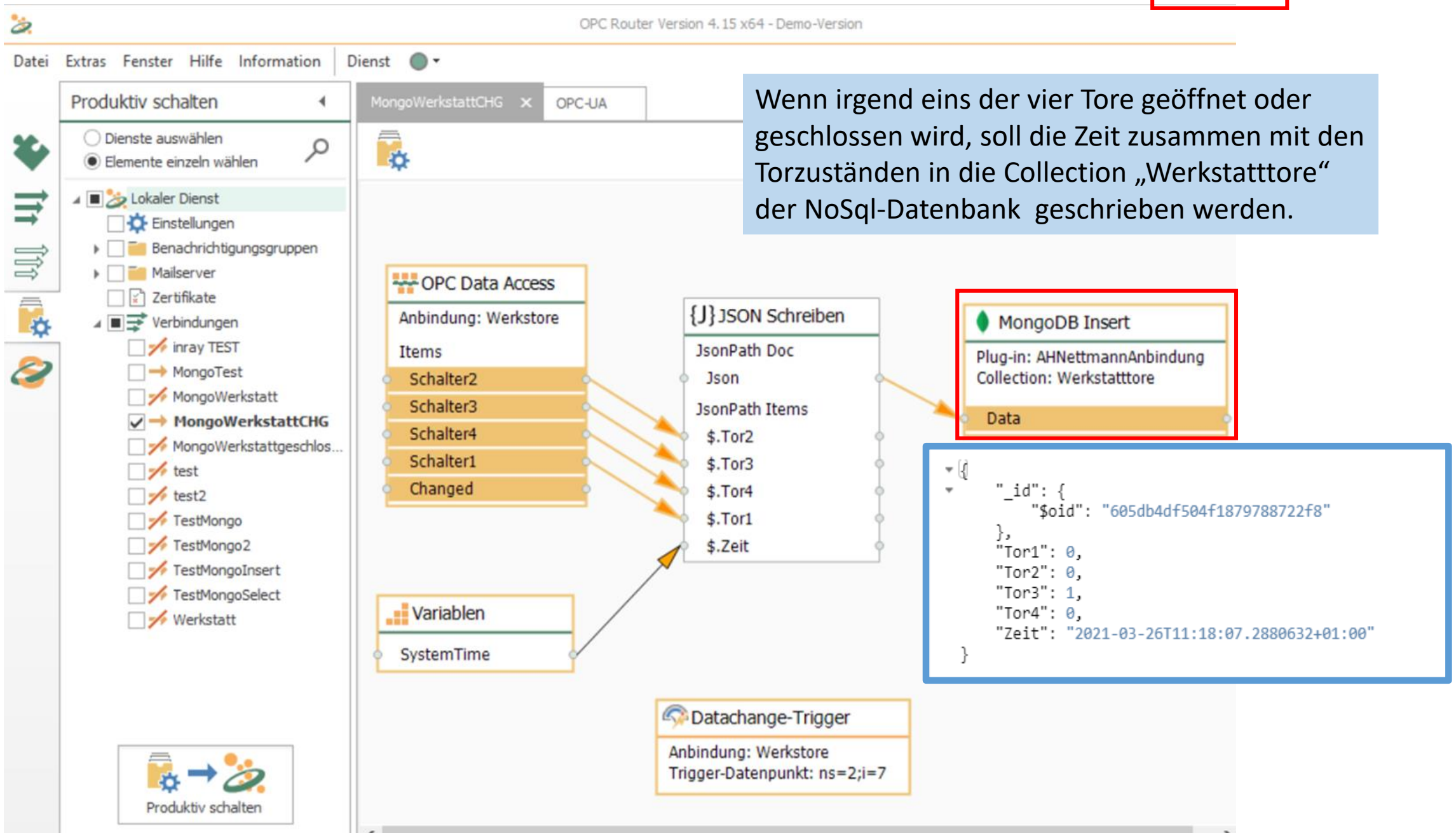


2.1 Middleware →OPCUA-Daten in eine MySQL-Datenbank schreiben (Insert)

Wenn irgend eins der vier Tore geöffnet oder geschlossen wird, soll die Zeit zusammen mit den Torzuständen in die Tabelle „Werkstore“ der MySQL-Datenbank geschrieben werden.



2.2 Middleware →OPCUA-Daten in eine NoSQL-Datenbank schreiben (Insert)

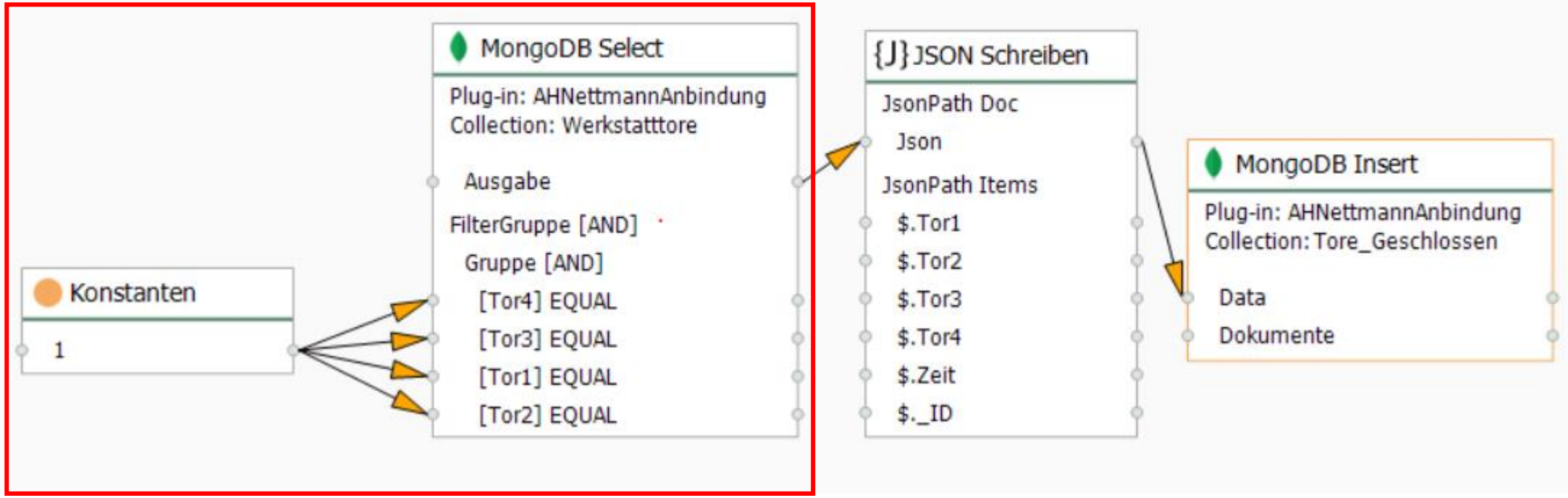


Wenn irgend eins der vier Tore geöffnet oder geschlossen wird, soll die Zeit zusammen mit den Torzuständen in die Collection „Werkstatttore“ der NoSql-Datenbank geschrieben werden.

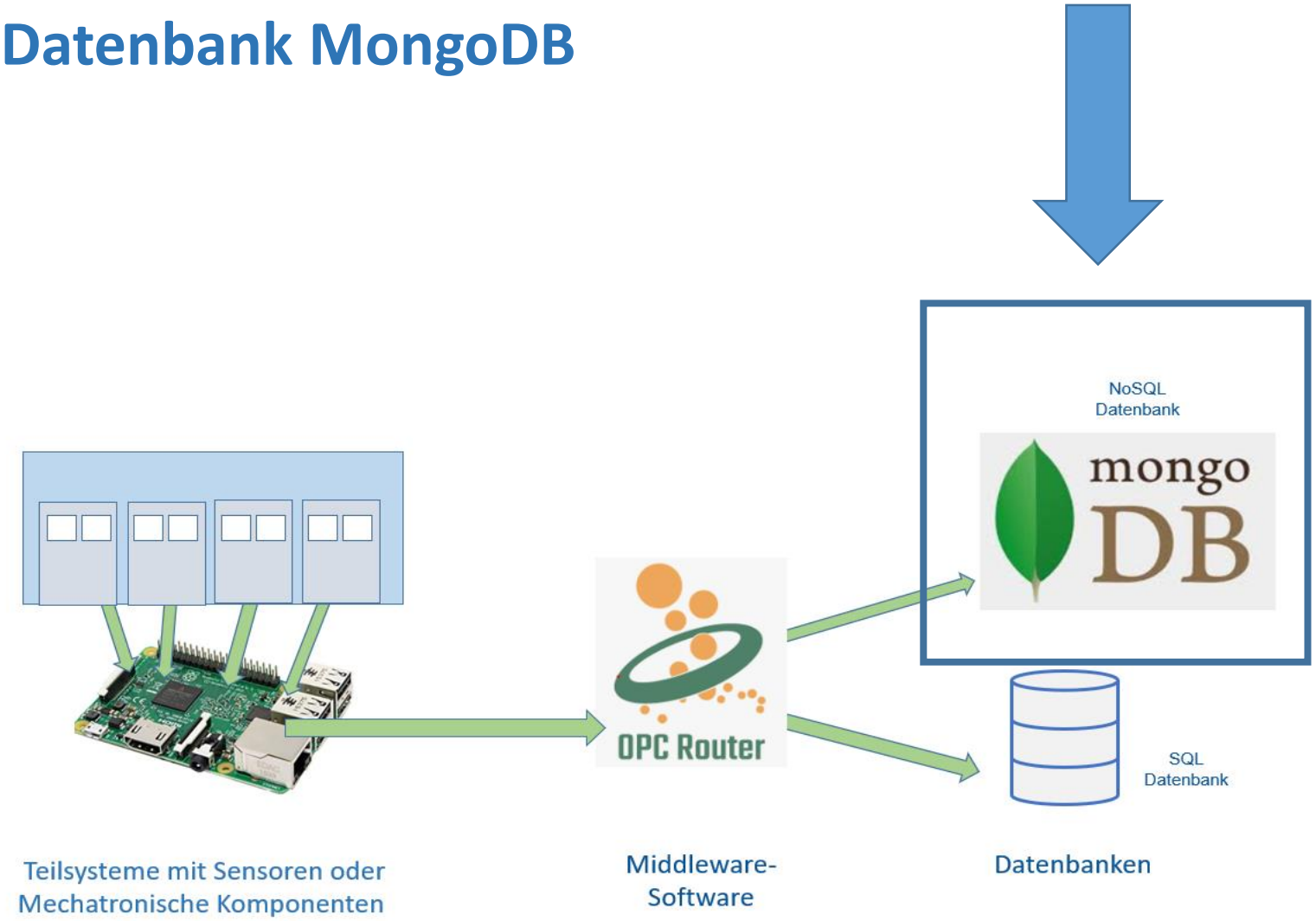
2.2 Middleware →OPCUA-Daten in eine NoSQL-Datenbank (Mongo) schreiben (Select+Insert)

Alle Dokumente (Datensätze), bei denen alle Tore geschlossen sind, sollen in die Collection (Tabelle) „Tore_geschlossen“ geschrieben werden.


```
{
    "_id": {
        "$oid": "5fd63e8f99ef6b087411b320"
    },
    "Tor1": 1,
    "Tor2": 1,
    "Tor3": 1,
    "Tor4": 1,
    "Zeit": "2020-12-13T17:17:19.8001567+01:00"
}
```

3. Zielanwendung: NoSQL-Datenbank MongoDB



3. Zielanwendung: NoSQL-Datenbank MongoDB

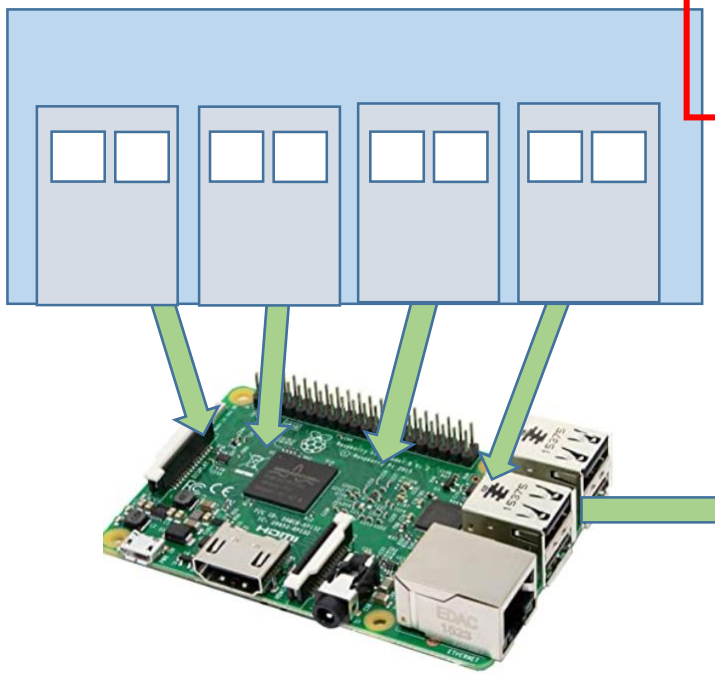
mongoDB Compass

→Zugriff über Programm/App

Connection-String

```
mongodb+srv://gast:Cisco_123@cluster0.arkrd.mongodb.net/test?authSource=admin&replicaSet=
atlas-l7lavu-shard0&readPreference=primary&appname=MongoDB%20Compass&ssl=true
```

DBaaS

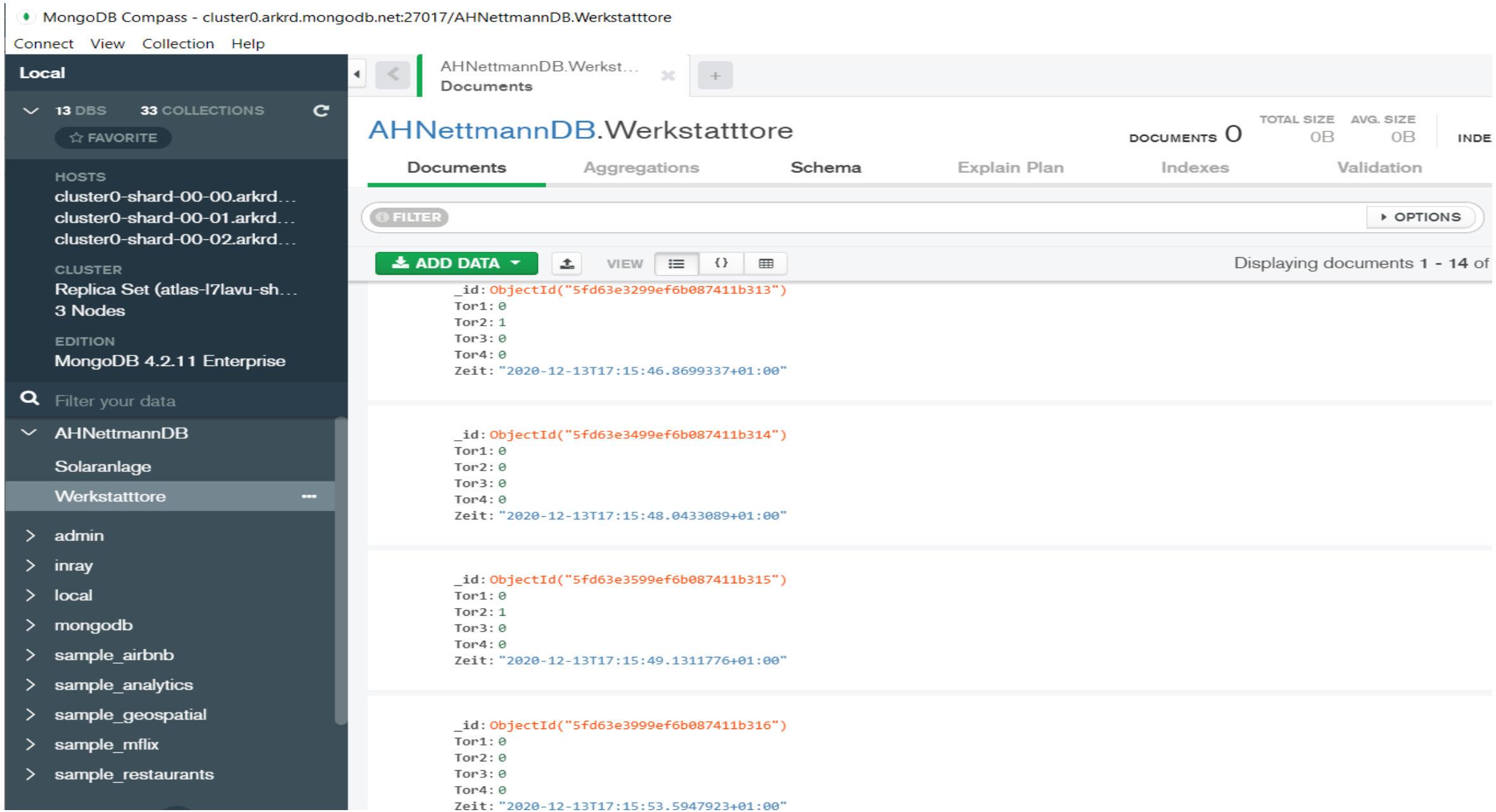


Teilsystem mit OPCUA-Server

Middleware-Software

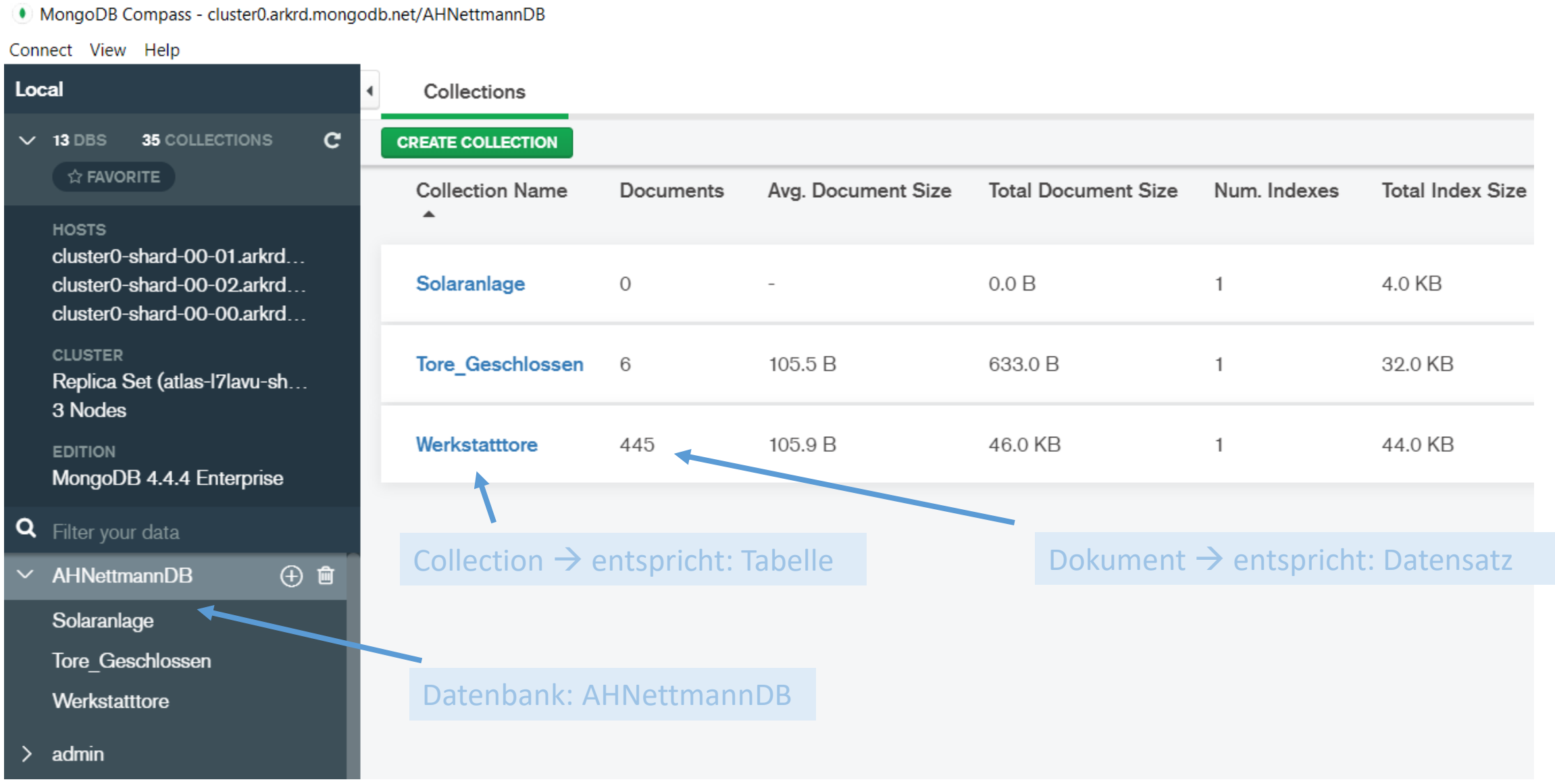
NoSQL-Datenbank

Hinweis: DBaaS, SaaS, weitere: IaaS Infrastructure, Paas Plattform, StaaS Storage, BaaS Backup


MongoDB Compass - cluster0.arkrd.mongodb.net:27017/AHNettmannDB.Werkstatttore
Connect View Collection Help
Local
13 DBS 33 COLLECTIONS
FAVORITE
HOSTS
cluster0-shard-00-00.arkrd...
cluster0-shard-00-01.arkrd...
cluster0-shard-00-02.arkrd...
CLUSTER
Replica Set (atlas-l7lavu-sh...
3 Nodes
EDITION
MongoDB 4.2.11 Enterprise
Filter your data
AHNettmannDB
Solaranlage
Werkstatttore
admin
inray
local
mongodb
sample_airbnb
sample_analytics
sample_geospatial
sample_mflix
sample_restaurants
AHNettmannDB.Werkst...
Documents
AHNettmannDB.Werkstatttore
DOCUMENTS 0
TOTAL SIZE 0B
AVG. SIZE 0B
INDE
Documents
Aggregations
Schema
Explain Plan
Indexes
Validation
FILTER
OPTIONS
ADD DATA
VIEW
Displaying documents 1 - 14 of
_id: ObjectId("5fd63e3299ef6b087411b313")
Tor1: 0
Tor2: 1
Tor3: 0
Tor4: 0
Zeit: "2020-12-13T17:15:46.8699337+01:00"
_id: ObjectId("5fd63e3499ef6b087411b314")
Tor1: 0
Tor2: 0
Tor3: 0
Tor4: 0
Zeit: "2020-12-13T17:15:48.0433089+01:00"
_id: ObjectId("5fd63e3599ef6b087411b315")
Tor1: 0
Tor2: 1
Tor3: 0
Tor4: 0
Zeit: "2020-12-13T17:15:49.1311776+01:00"
_id: ObjectId("5fd63e3999ef6b087411b316")
Tor1: 0
Tor2: 0
Tor3: 0
Tor4: 0
Zeit: "2020-12-13T17:15:53.5947923+01:00"

mongoDB Compass

„Explorer für Datenbanken“
→Zugriff über Programm/App



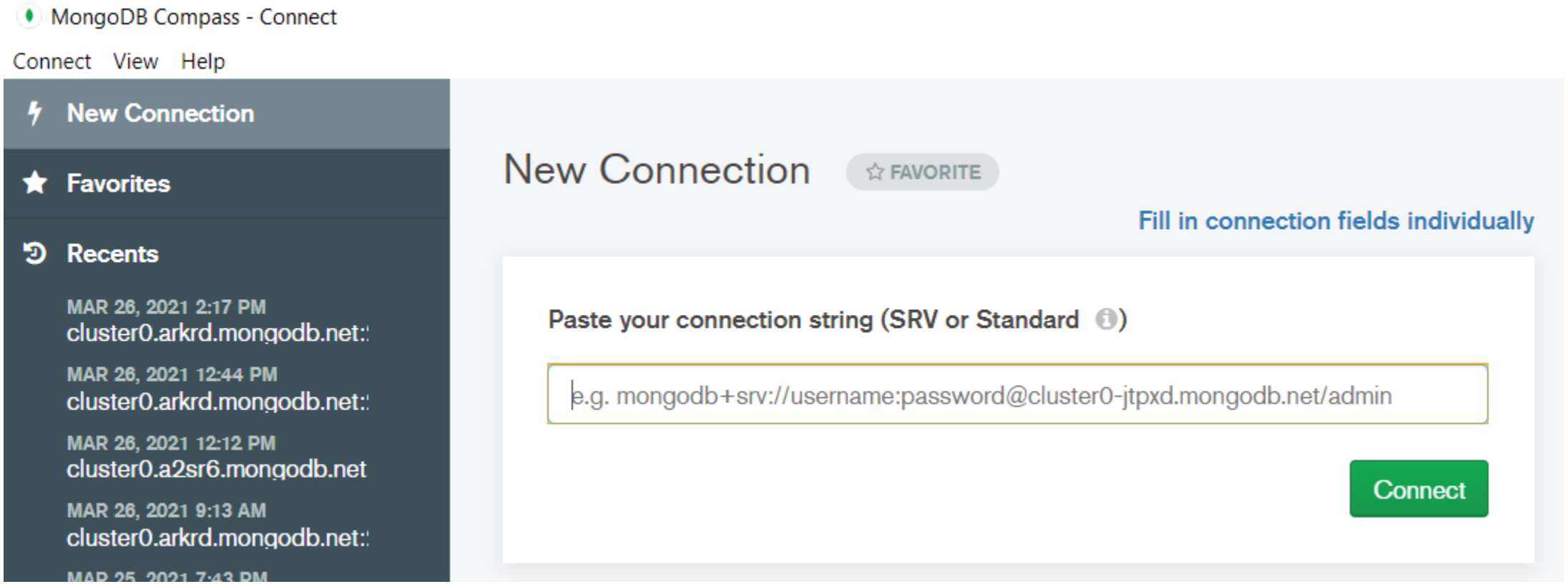
Stellen Sie nun eine Verbindung zur MongoDB (DBaaS) über den folgenden Connection-String her!

mongodb+srv://gast:Cisco_123@cluster0. arkrd.mongodb.net/test?authSource=admin&replicaSet=atlas-l7lavu-shard0&readPreference=primary&appname=MongoDB%20Compass&ssl=true

Vorsicht! Leerezeichen etc. entfernen!!!

Starten Sie MongoDBCompass und fügen Sie den Connection-String ein!

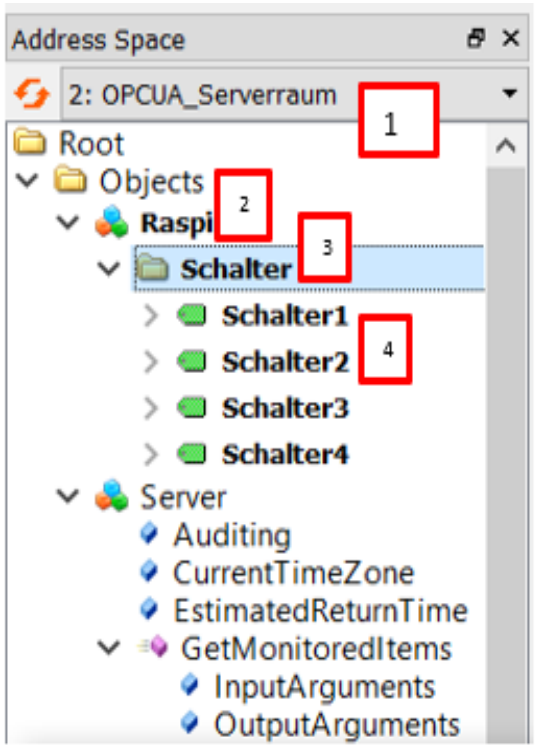
mongodb+srv://[redacted]@cluster0.[redacted].mongodb.net/test?authSource=admin&replicaSet=atlas-l7lavu-shard0&readPreference=primary&appname=MongoDB%20Compass&ssl=true



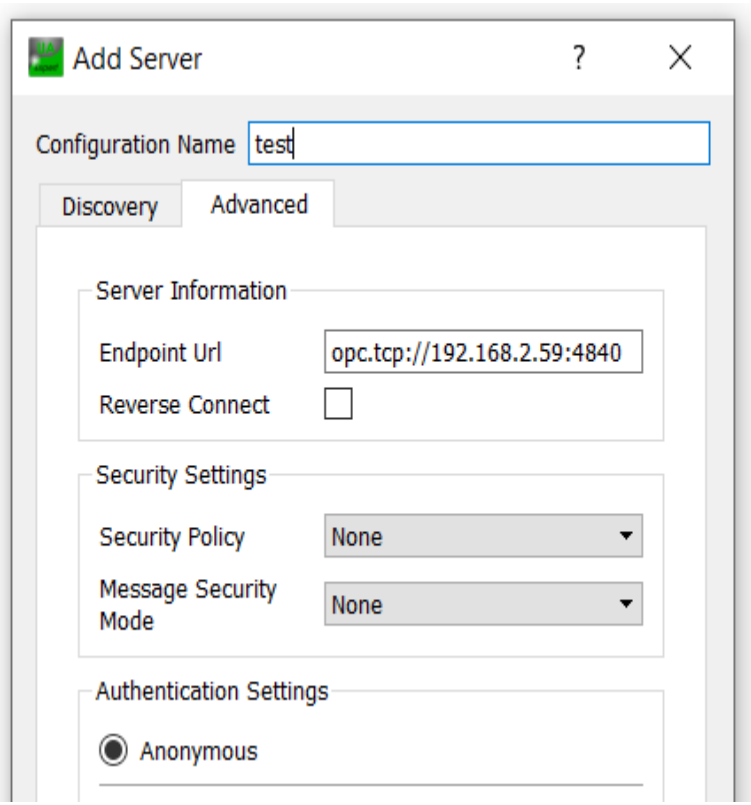
„Hausaufgabe für Interessierte:“

→Laden Sie auf Ihren Raspberry Pi das Programm „zWerkstoreVarianteFlagIPRandom.py“ und starten Sie es!

→Greifen Sie nun auf die OPCUA-Datenpunkte des Raspi OPCUA-Servers von Ihrem Laptop aus mit Hilfe des „OPCUA-Explorers“ UA Expert (Siehe Seite/Folie 7) der Fa. Unified Automation GmbH zu und lassen Sie sich alle OPCUA-Datenpunkte ausgeben!



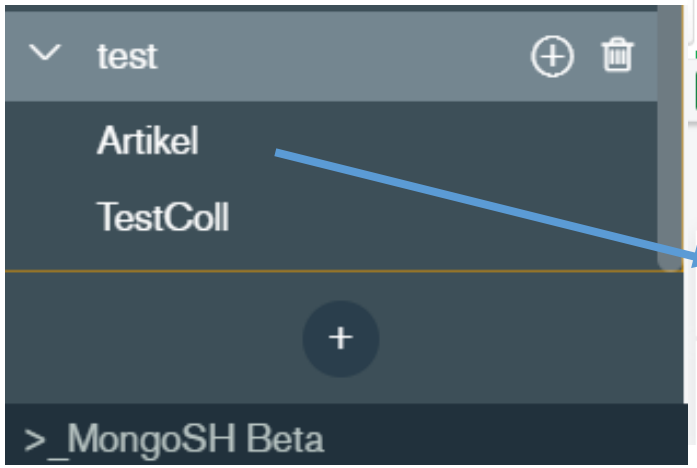
Ansicht im UA-Expert



Eine Collection „Artikel“ und „TestColl“ in der Datenbank „test“ erstellen

```
_id: ObjectId("604f517f504f181828805efb")
Tor1: 1
Tor2: 1
```

```
>_MongoSH Beta
> db.createCollection("Artikel")
< { ok: 1 }
> use test
< 'switched to db test'
> db.createCollection("TestColl")
< { ok: 1 }
>
```



Collections

CREATE COLLECTION

Collection Name	Documents	Avg. Document Size	Total Document Size	Num. Indexes	Total Index Size	Properties
Artikel	0	-	0.0 B	1	4.0 KB	
TestColl	0	-	0.0 B	1	4.0 KB	

MongoShell Befehle im Zusammenhang mit der vorhandenen Datenbank AHNettmann:

Hinweis: Ausführung der meisten Befehle funktioniert nicht als „gast“-User

use AHNettmannDB // Datenbank auswählen
db.Werkstatttore.find() // Alle Dokumente der Collection Werkstatttore ausgeben

//Alle Dokumente, bei denen das Tor geschlossen ist

```
> db.Werkstatttore.find({'Tor1':1})
```

//Alle Dokumente, bei denen alle Tore geschlossen sind

```
> db.Werkstatttore.find({'Tor1':1, 'Tor2':1, 'Tor3':1, 'Tor4':1})
```