

Die Integration des Air-Q in openhab erfolgt über das EXEC-Binding.

Thing-Konfiguration:

```
Thing exec:command:airqdata "AirQ" [command="/usr/bin/python3 absoluterPfadzumScript/airq-data.sh", timeout=15, interval=120, autorun=false]
```

Der Parameter interval definiert die Wiederholrate in Sekunden.

Damit das Script ausgeführt wird, muss es in der EXEC-Whitelist definiert sein (im Ordner misc, Datei exec.whitelist).

```
# For security reasons all commands that are used by the exec binding or transformation need to be whitelisted.  
# Every command needs to be listed on a separate line below.  
/usr/bin/python3 absoluterPfadzumScript/airq-data.sh
```

Items-Definition:

```
String AirQ_data "AirQ-Json-Data [%s]" {channel="exec:command:airqdata:output"}
DateTime AirQ_Timestamp "Zeitstempel: [%1$td.%1$tm.%1$tY - %1$tH:%1$tM]"
String AirQ_Status "Status: [%s]"
Number AirQ_CO "CO [%.1f ppm]"
Number AirQ_CO2 "CO2 [%.1f ppm]"
Number AirQ_O2 "Sauerstoff [%.2f %%]"
Number AirQ_PM1 "Feinstaub PM1 [%.1f µg/m³]"
Number AirQ_PM25 "Feinstaub PM2.5 [%.1f µg/m]"
Number AirQ_PM10 "Feinstaub PM10 [%.1f µg/m]"
Number AirQ_Temp "Temperatur [%.1f C]"
Number AirQ_aktZeit "aktiv seit [%s s]"
Number AirQ_Lautstaerke "Lautstärke [%s db]"
Number AirQ_Luftfeuchtigkeit_abs "Luftfeuchtigkeit (absolut) [%.1f g/m]"
Number AirQ_Luftfeuchtigkeit "Luftfeuchtigkeit [%.1f %%]"
Number AirQ_O3 "Ozon [%.1f µg/m³]"
Number AirQ_SO2 "SO2 [%.1f µg/m³]"
Number AirQ_NO2 "Stickstoff [%.1f µg/m³]"
Number AirQ_Luftdruck "Luftdruck [%.1f hPa]"
Number AirQ_TVOC "VOCs [%.2f ppb]"
Number AirQ_health "Gesundheitsindex [%.1f %%]"
Number AirQ_performance "Leistungsindex [%.1f %%]"
```

Sitemap-Definition:

```
sitemap airq label="Air-Q"  
{  
  Frame label="" {  
    Text item=AirQ_health icon="status"  
    Text item=AirQ_performance icon="heating"  
  }  
  Frame label="Status" {  
    Text item=AirQ_Timestamp icon="clock"  
    Text item=AirQ_aktZeit  
    Text item=AirQ_Status  
  }  
  Frame label="Sensoren" {  
    Text item=AirQ_Lautstaerke icon="soundvolume"  
    Text item=AirQ_Luftdruck  
    Text item=AirQ_Luftfeuchtigkeit_abs icon="rain"  
    Text item=AirQ_Luftfeuchtigkeit icon="rain"  
    Text item=AirQ_CO  
    Text item=AirQ_CO2 icon="carbondioxide"  
    Text item=AirQ_O3  
    Text item=AirQ_PM1  
    Text item=AirQ_PM25  
    Text item=AirQ_PM10  
    Text item=AirQ_O2  
    Text item=AirQ_SO2  
    Text item=AirQ_NO2  
    Text item=AirQ_Temp icon="temperature"  
    Text item=AirQ_TVOC  
  }  
}
```

Rules-Definition:

```
rule "Air-Q meldet neue Werte"
when
  Item AirQ_data changed
then
  AirQ_Timestamp.postUpdate(transform("JSONPATH", "$.timestamp", AirQ_data.state.toString))
  AirQ_Status.postUpdate(transform("JSONPATH", "$.Status", AirQ_data.state.toString))
  AirQ_aktZeit.postUpdate(transform("JSONPATH", "$.uptime", AirQ_data.state.toString))
  AirQ_health.postUpdate(Float::parseFloat(transform("JSONPATH", "$.health", AirQ_data.state.toString)) / 10 )
  AirQ_performance.postUpdate(Float::parseFloat(transform("JSONPATH", "$.performance", AirQ_data.state.toString)) / 10)
  AirQ_CO.postUpdate(transform("JSONPATH", "$.co[0]", AirQ_data.state.toString))
  AirQ_CO2.postUpdate(transform("JSONPATH", "$.co2[0]", AirQ_data.state.toString))
  AirQ_Lautstaerke.postUpdate(transform("JSONPATH", "$.sound[0]", AirQ_data.state.toString))
  AirQ_Luftdruck.postUpdate(transform("JSONPATH", "$.pressure[0]", AirQ_data.state.toString))
  AirQ_Luftfeuchtigkeit_abs.postUpdate(transform("JSONPATH", "$.humidity_abs[0]", AirQ_data.state.toString))
  AirQ_Luftfeuchtigkeit.postUpdate(transform("JSONPATH", "$.humidity[0]", AirQ_data.state.toString))
  AirQ_NO2.postUpdate(transform("JSONPATH", "$.no2[0]", AirQ_data.state.toString))
  AirQ_O2.postUpdate(transform("JSONPATH", "$.oxygen[0]", AirQ_data.state.toString))
  AirQ_O3.postUpdate(transform("JSONPATH", "$.o3[0]", AirQ_data.state.toString))
  AirQ_PM1.postUpdate(transform("JSONPATH", "$.pm1[0]", AirQ_data.state.toString))
  AirQ_PM25.postUpdate(transform("JSONPATH", "$.pm2_5[0]", AirQ_data.state.toString))
  AirQ_PM10.postUpdate(transform("JSONPATH", "$.pm10[0]", AirQ_data.state.toString))
  AirQ_Temp.postUpdate(transform("JSONPATH", "$.temperature[0]", AirQ_data.state.toString))
  AirQ_SO2.postUpdate(transform("JSONPATH", "$.so2[0]", AirQ_data.state.toString))
  AirQ_TVOC.postUpdate(transform("JSONPATH", "$.tvoc[0]", AirQ_data.state.toString))
end
```