



Wetterstation im Amateurfunk



**Bresser Wetterstation
Funk mit Außensensor
Wetter Center
5-in-1, schwarz**

**Aussendung der
Wetterdaten im
ISM-Band 868 MHz**

- **Wetterdaten einer Hausstation weiter verwenden**
Die Idee und das Ziel



- **Der Aufbau und dazu benötigte Komponenten**

- **Software und Integration in das aprs.fi-Netz**



- **Weitere Möglichkeiten der Verwendung von eigenen Wetterdaten**



- **Fazit und Links**



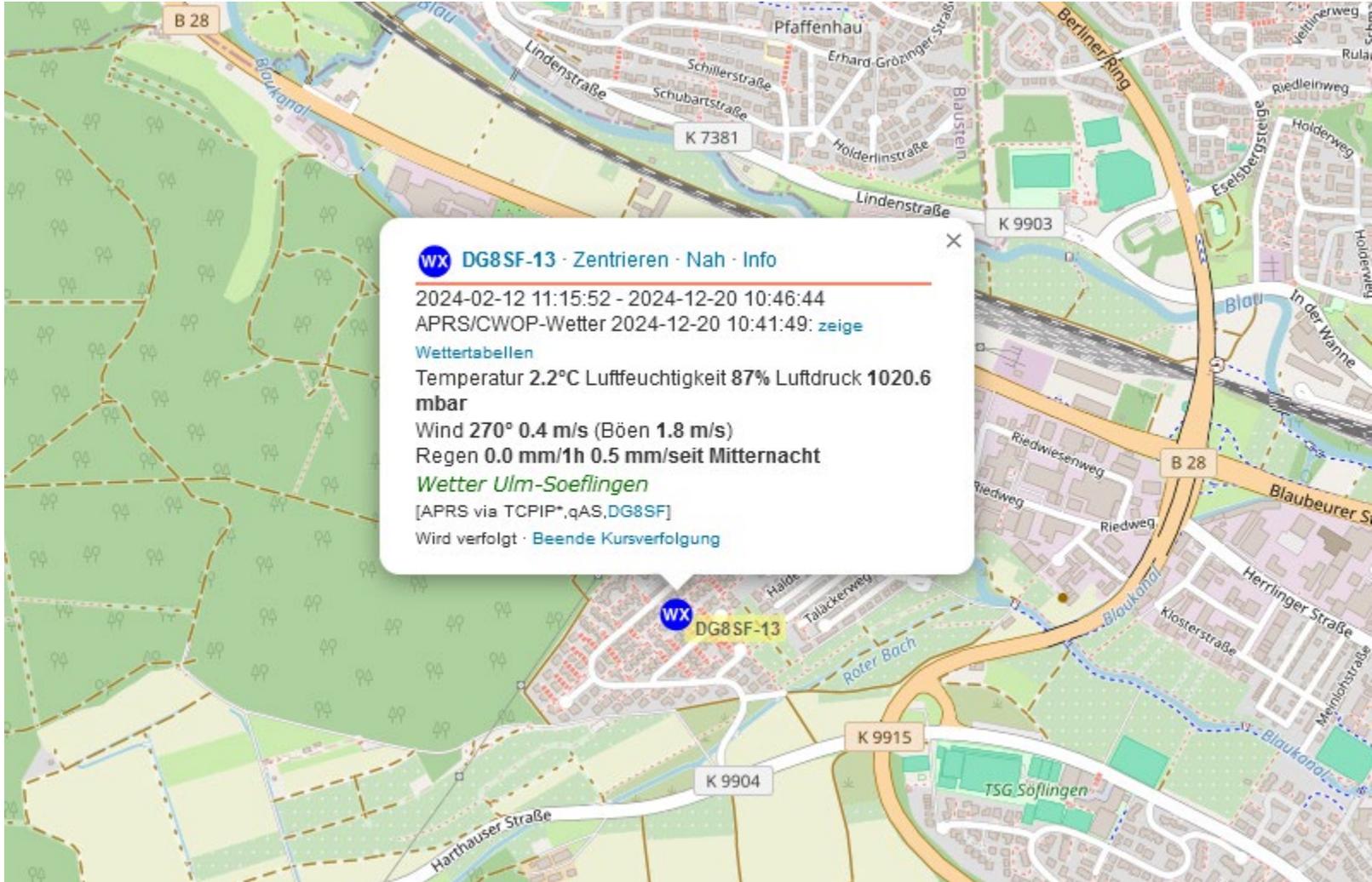
Idee und

Können die via ISM
auf **868 MHz**
übertragenen
Daten nicht noch
anderweitig
verwendet werden?





... das Ziel

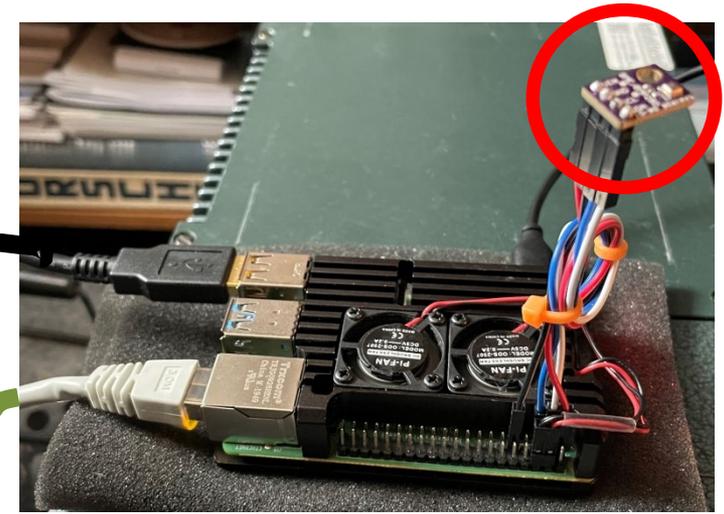
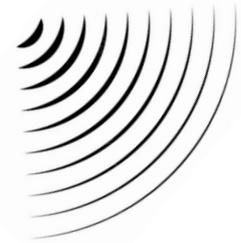




Aufbau und Komponenten



868 MHz



- RTL-SDR-Stick 2832U mit 0.5 ppm TCXO
- RasPi - min. 3B+
- **BME280 Sensor (I2C)**

Internetanbindung



Software und Integration (1)



Das Programm rtl_433 empfängt über den RTL_SDR-Stick die Daten auf 868 MHz von der Wetterstation und wandelt diese in ein passendes – sog. JSON-Format um:

```
{"time" : "2024-12-20 10:11:53", "model" : "Bresser-5in1", "id" : 118,  
"battery_ok" : 1, "temperature_C" : 2.400, "humidity" : 82, "wind_max_m_s" :  
0.800, "wind_avg_m_s" : 0.800, "wind_dir_deg" : 292.500, "rain_mm" :  
4.800, "mic" : "CHECKSUM"}
```

Der Luftdruck wird nicht von der Wetterstation übermittelt. Dieser kommt von einem gesonderten Sensor (BME280), der auf dem RasPi direkt sitzt. Das erforderliche JSON-File wird von einem eigenen Python-Script erzeugt:

```
{"baro_Hpa": 10209, "temp_C": 15.4, "humi_P": 41.5, "Stand": "2024-12-  
20T11:15:28", "baro_raw": 959}
```



Software und Integration (2)



Die Aufbereitung der Rohdaten aus den JSON-Files ist sehr aufwändig und wird in einem eigenen Python-Skript mit insgesamt ca. 400 Zeilen Code erleigt.

Ergebnis ist ein Datenstring, der mittels dem Linux NetCat-Befehl <nc> an den aprs.fi-Server übermittelt wird:

```
# APRS Ausgabe nur alle x timing Min.  
aprs_diff = int(time.time()) - aprs_timer  
if aprs_diff > timing:  
    aprs_timer = int(time.time())  
    datei_schreiben(datei_aprs, aprs_login + msg)  
    cmd = "nc -v -w 5 rotate.aprs2.net 14580 < " + datei_aprs
```

Und so sieht der nach aprs.fi übermittelte Datensatz aus:

DG8SF-

**13>APRS,TCPIP*:@201027h4824.02N/00956.07E_270/002g002t035r
000p000P000h88b10211Wetter Ulm-Soeflingen**



Software und Integration (3)



Folgende Voraussetzungen müssen gegeben sein:

- Zur Anmeldung am aprs.fi-Server ist ein Passcode erforderlich, der jedem Rufzeichen zugeordnet ist.

Hierzu gibt es Passcodegeneratoren im Netz – z.Bsp.:

<https://apps.magicbug.co.uk/passcode/index.php/passcode>

APRS Passcode Generator

Important: Do not use fake callsigns to inject content into the APRS-IS network you will likely get banned.

Callsign

Technical Example of Passcode Generation using PHP

Source code available on [Github](#)

This tool was developed by Peter Goodhall [2M0SQL](#) who also runs [magicbug](#)



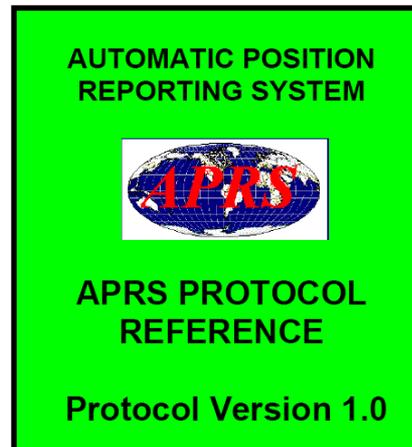
Software und Integration (4)

Folgende Voraussetzungen müssen gegeben sein:

- Der Datenstring, der an den aprs.fi-Server übermittelt wird muss zwingend einem eindeutigen Format entsprechen, um richtig auf der APRS-Webseite dargestellt zu werden.

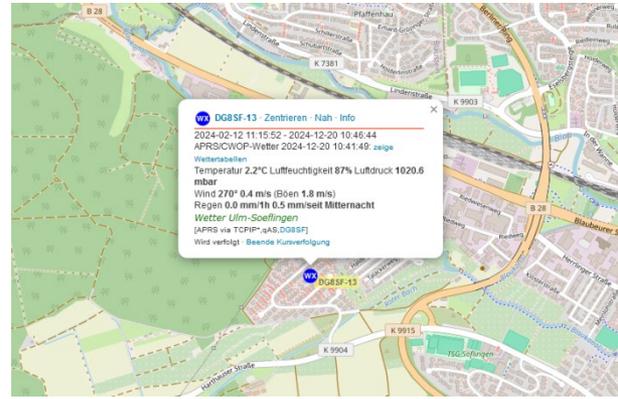
In diesem Manual finden sich alle erforderlichen Informationen:

<http://www.aprs.org/doc/APRS101.PDF>

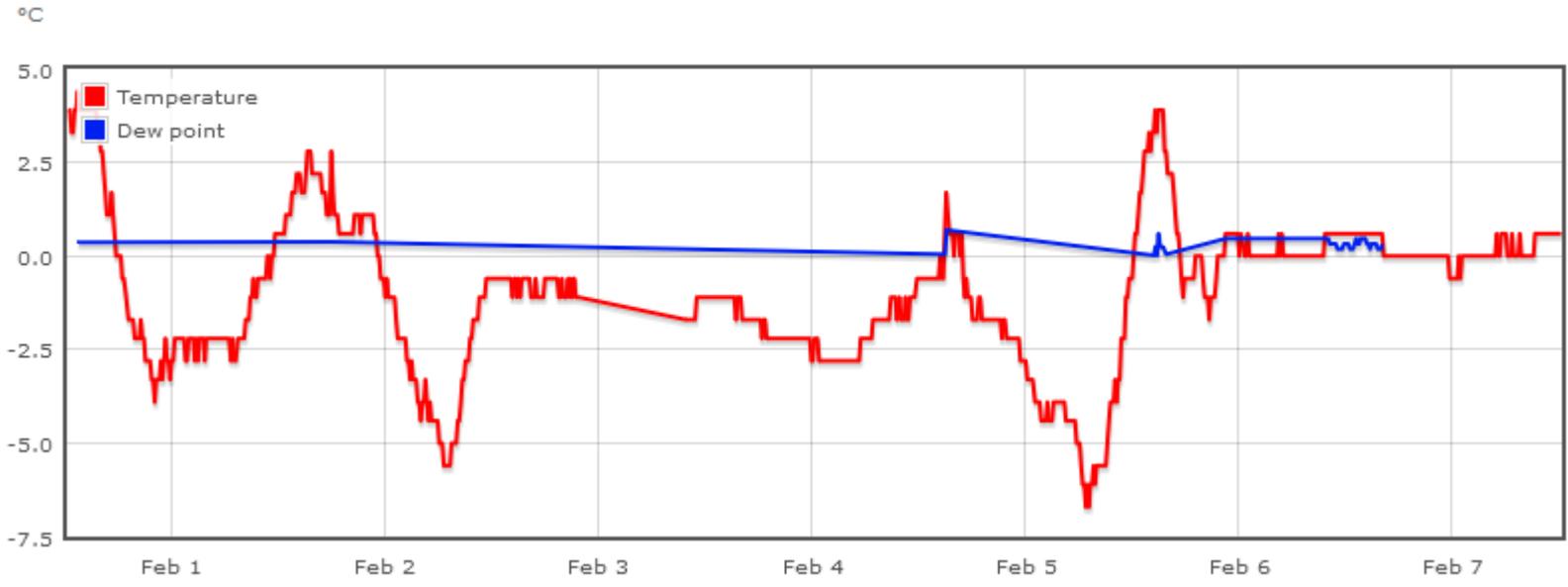




Darstellung in aprs.fi



DG8SF-13 Temperatur 2025-01-31 12:29:40 -> 2025-02-07 12:14:44 CET

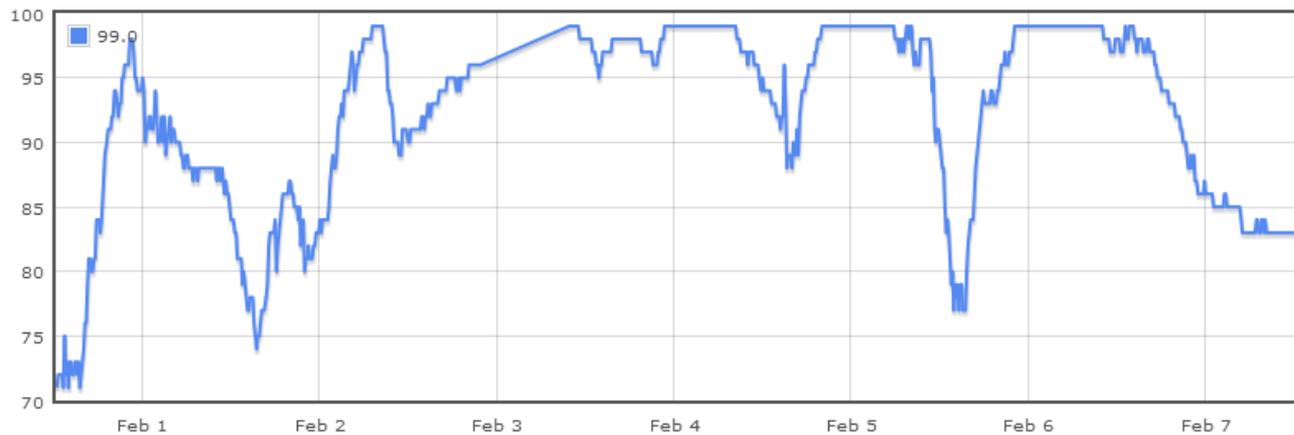




Darstellung in aprs.fi

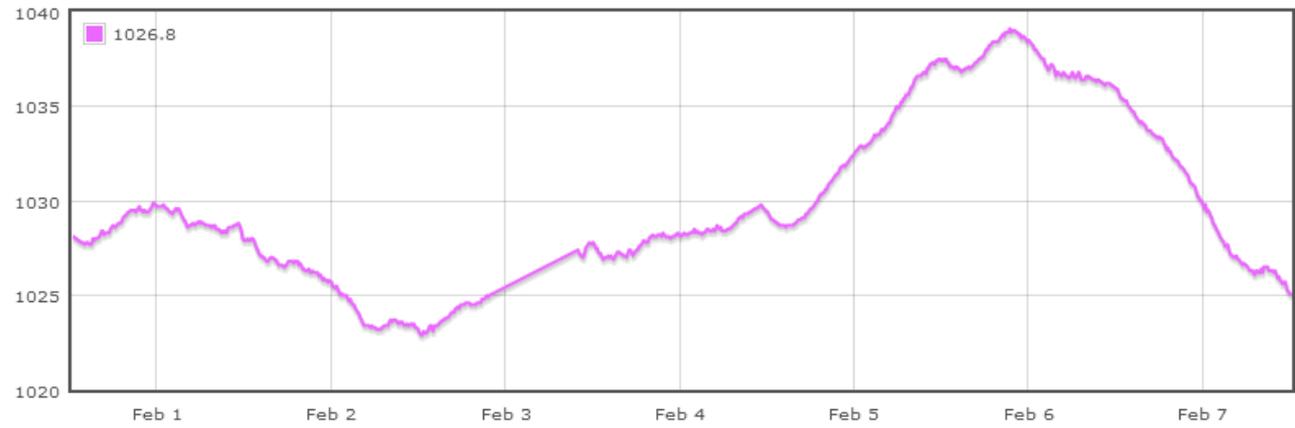
DG8SF-13 Luftfeuchtigkeit 2025-01-31 12:29:40 -> 2025-02-07 12:14:44 CET

%%



DG8SF-13 Luftdruck 2025-01-31 12:29:40 -> 2025-02-07 12:14:44 CET

mbar



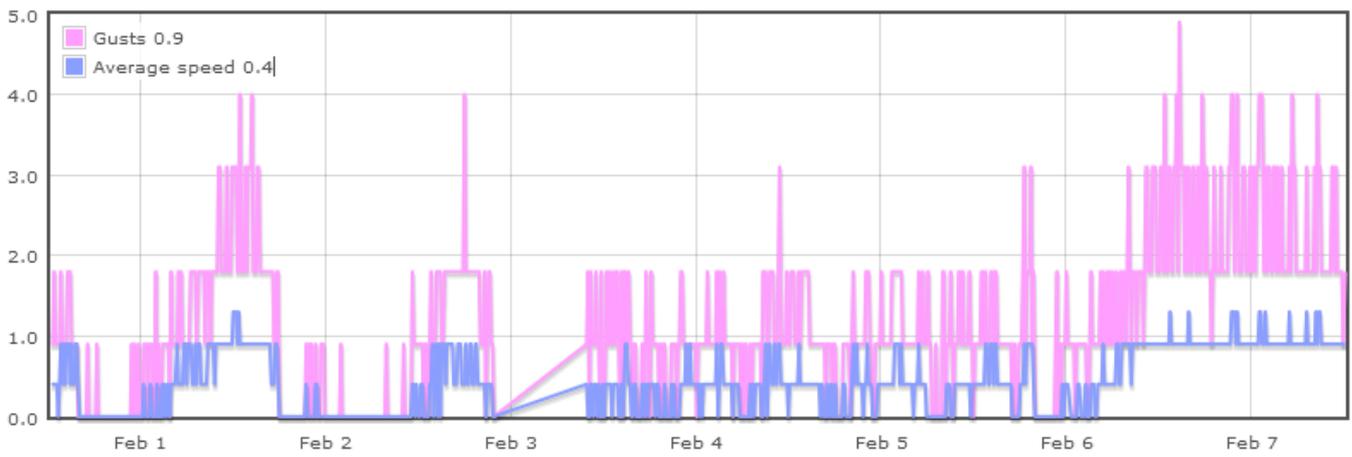


Darstellung in aprs.fi



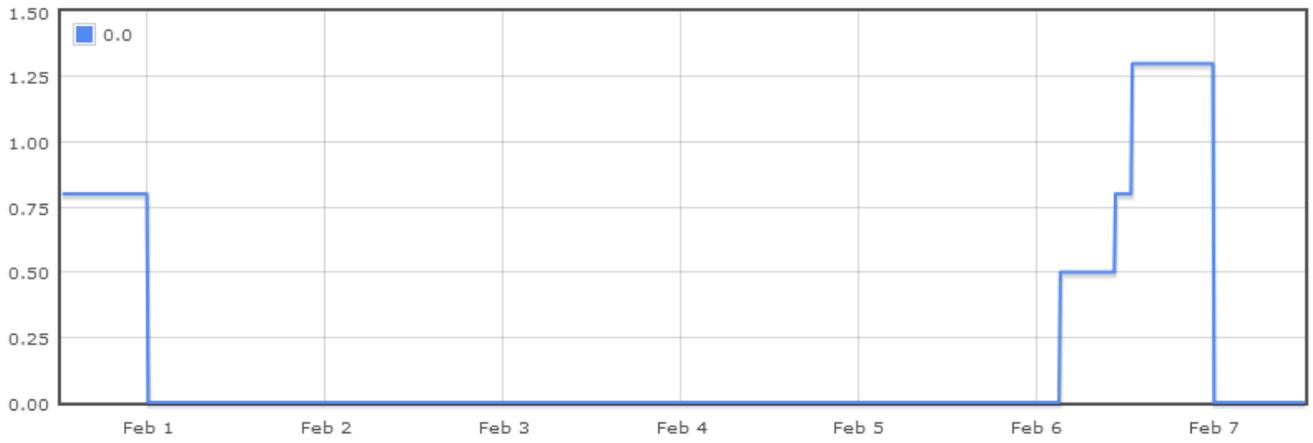
DG8SF-13 Wind 2025-01-31 12:29:40 -> 2025-02-07 12:14:44 CET

m/s



DG8SF-13 Regen seit Mitternacht 2025-01-31 12:29:40 -> 2025-02-07 12:14:44 CET

mm





Aussendung auf 144.800 MHz

Sind die Daten auf dem aprs.fi-Server, sind sie aber noch nicht „in der Luft“.

Hier muss noch ein Digipeater tätig werden:



Hier die Konfiguration: 2 m GM360 – TNC-X und Raspi 3B mit APRX Software



Empfang auf 144.800 MHz



Darstellung Wetter auf einem Kenwood TH-D74 und einem Kenwood TM-D710



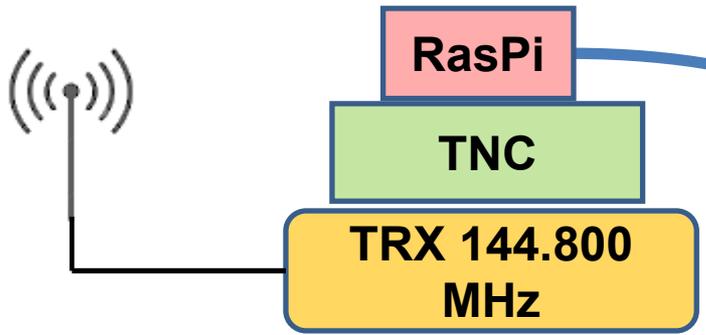
Übersicht

868 MHz

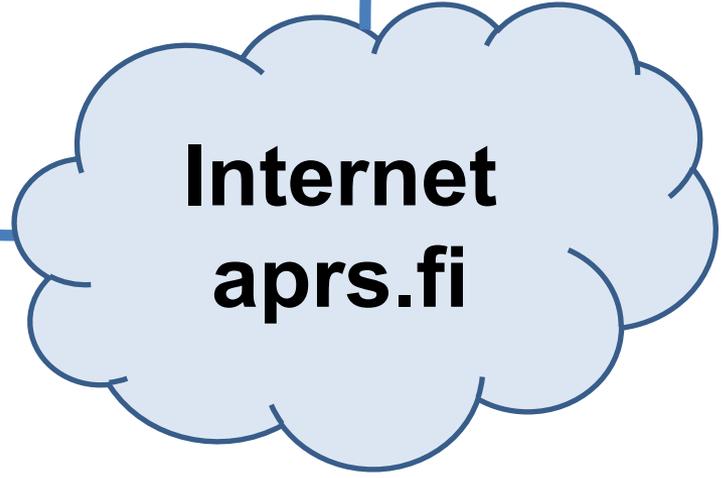


RTL-SDR Stick Empfänger

Aussendung der Wetterdaten



Digipeater-Station





Ausgabe per Audio svxlink



**Wetterbericht
von
DG8SF**

Frequenz: 145.2750



Wetter im Amateurfunk



- ist herausfordernd
- ist abwechslungsreich
- ist sehr lehrreich
- fördert die Kommunikation
- Macht ein wenig stolz wenn es mal läuft ... ;-)



- braucht richtig Zeitressourcen und Linux-Kenntnisse
- nichts für Gutwettertage
- kein plug & play



Wissenswertes Links

- Programm-Sourcen von DL7ATA oder direkt von DG8SF
<https://github.com/dl7ata>
- Wetterstation DG8SF
<https://aprs.fi/#!call=a%2FDG8SF-13&timerange=3600&tail=3600>

Die APRS Landkarte:

<https://aprs.fi>

- Sourcen für APRX:
<https://thelifeofkenneth.com/aprx/>
- APRX Konfiguration:
<http://manpages.ubuntu.com/manpages/bionic/man8/aprx.8.html>
- [Bresser Wetterstation 5in1](#) (beispielhafter Link)
- [RTL SDR Stick](#) (beispielhafter Link)