

Redis



Plucking Data from Thin Air with Software Defined Radio



Guy Royse

Developer Advocate

Redis

 @guy.dev

 @guyroyse

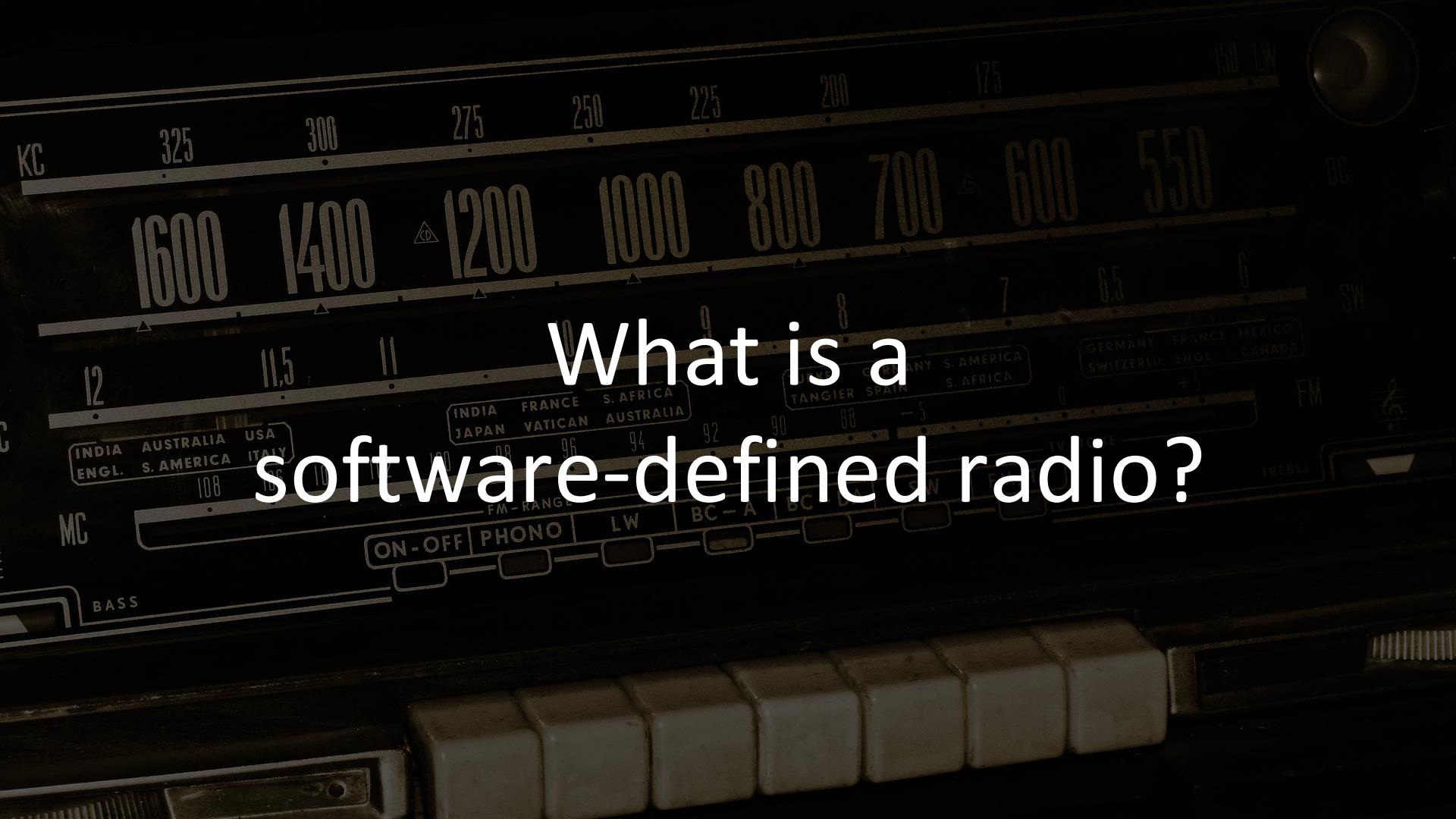
 github.com/guyroyse

 guy.dev

 W8GUY



What is a software-defined radio?



A radio you plug into your computer

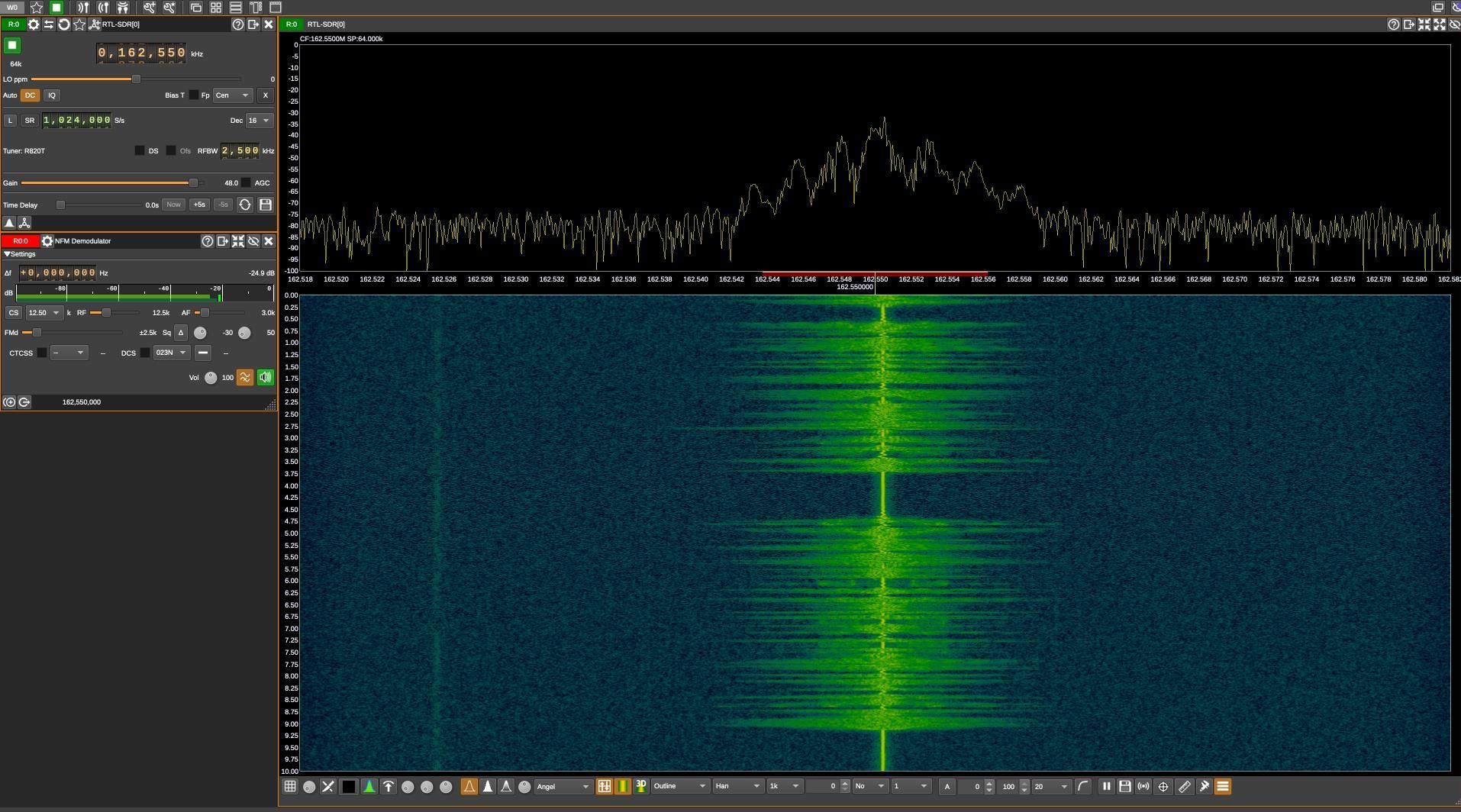
Antenna

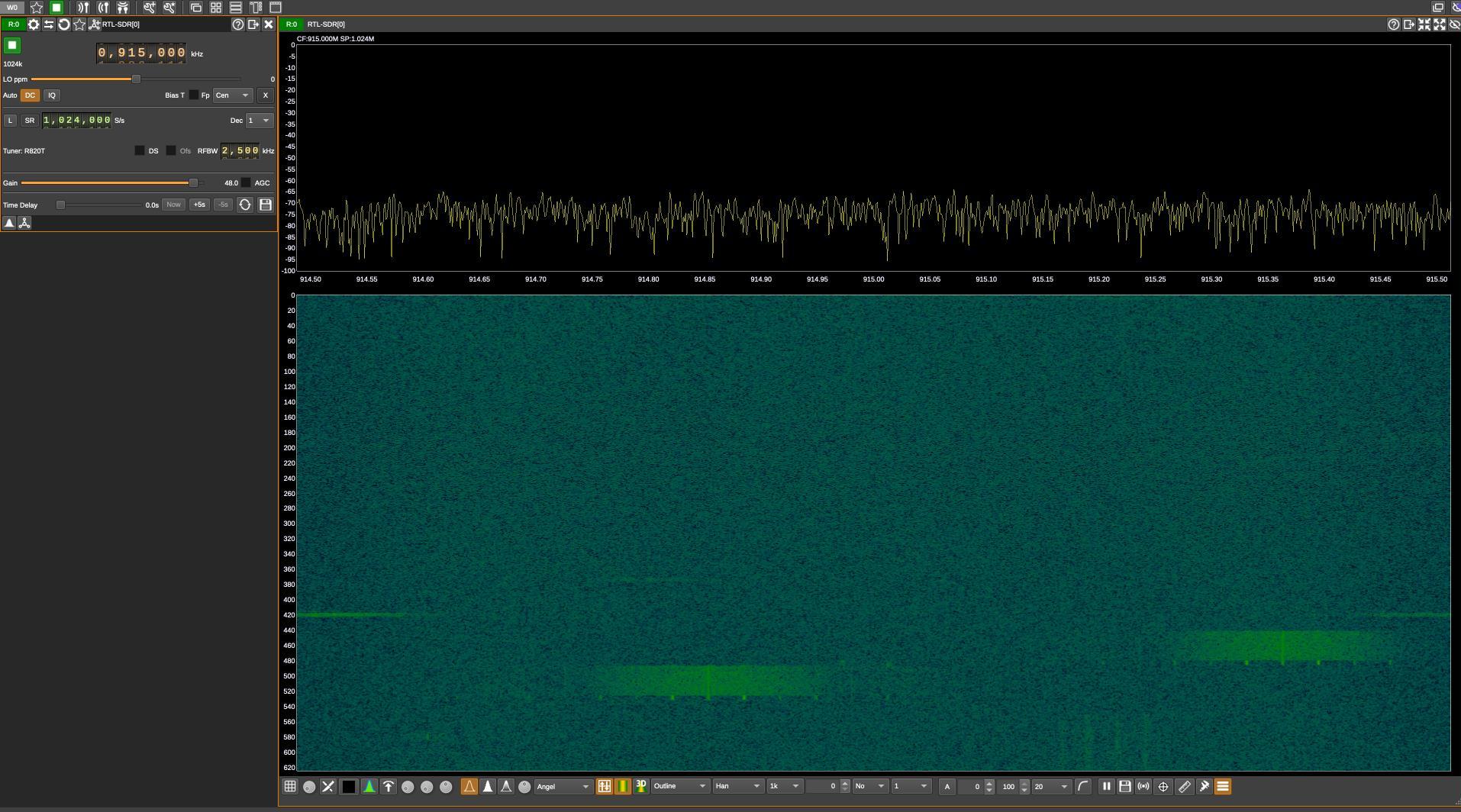


USB

A regular radio with less stuff







Where we usually get our data



Users



Phones &
Tablets



APIs



Databases



Cloud
Services



ISPs



Devices emit data over radio



Vehicles



Security Systems



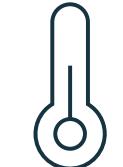
Satellites



Aircraft



Radio Services



Weather Stations



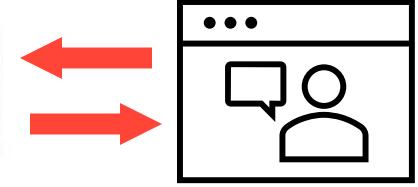
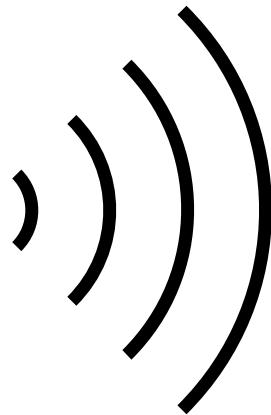
Utilities



Home Automation

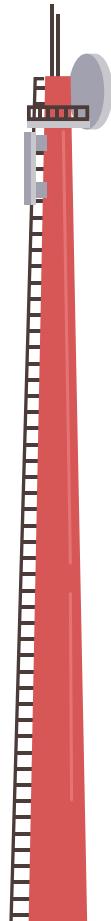
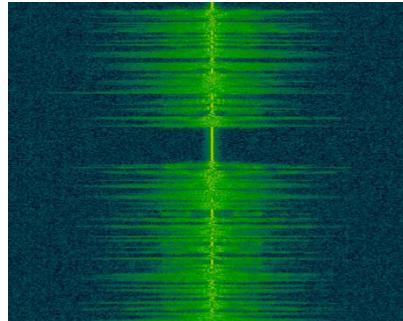


Software-defined radio can decode it all



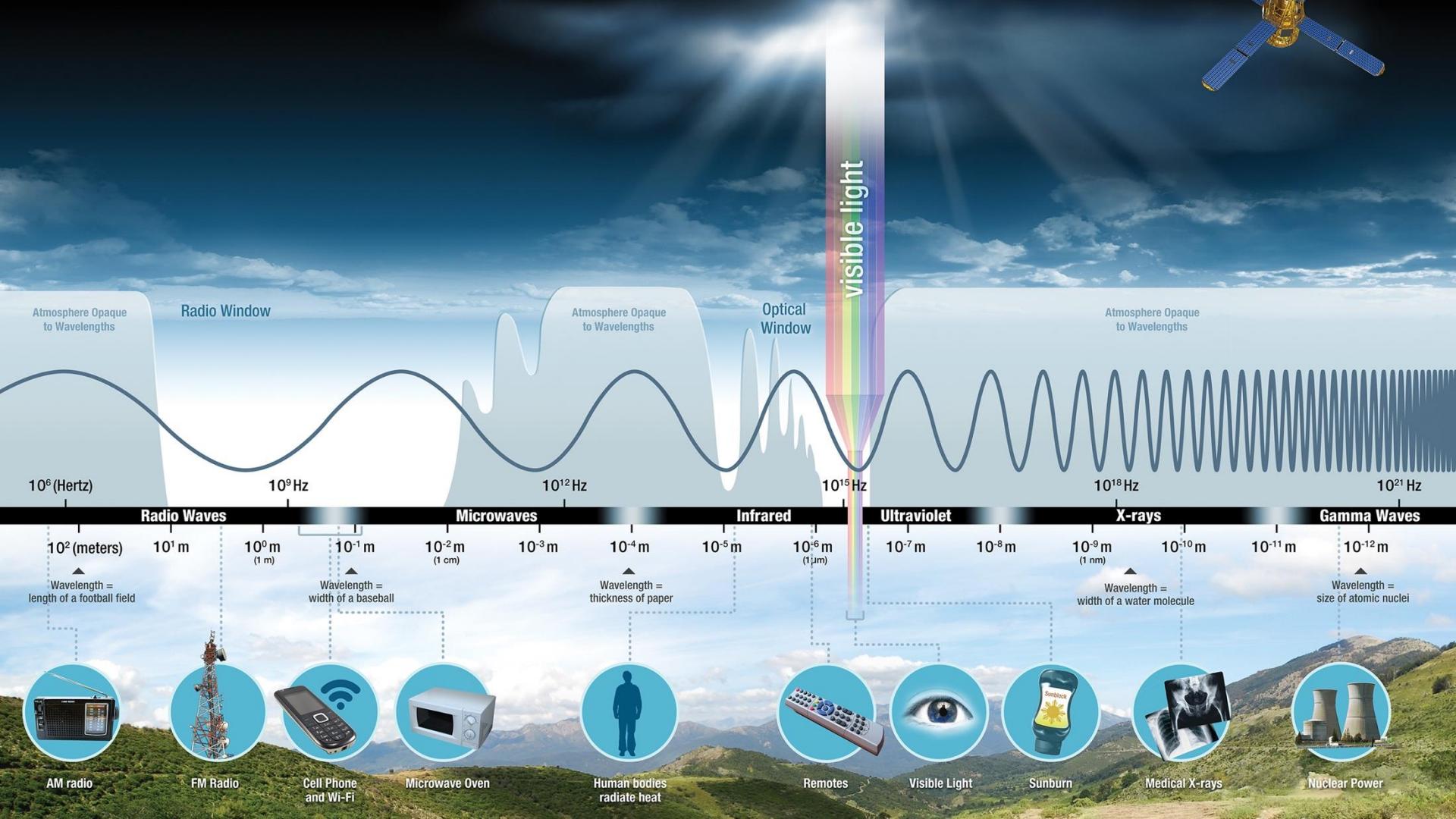
Let's get hands on!

01-INSTALLATION-AND-TESTING.md



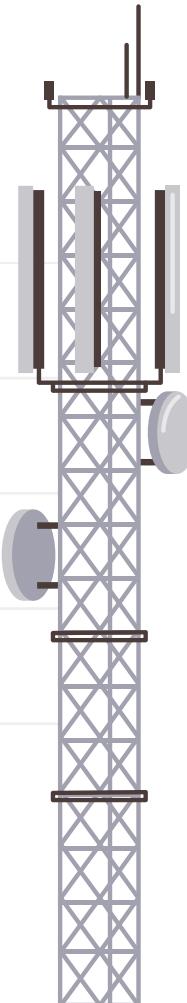
● A PRIMER

Radio for noobs



A bit of the radio spectrum

	Frequency	Wavelength	Propagation	Uses
Low frequency LF	30 kHz – 300 kHz	10 km – 1 km	Groundwave	Submarine communication
Medium frequency MF	300 kHz – 3 MHz	1 km – 100 m	Groundwave Skywave	AM radio Maritime communication
High frequency HF	3 MHz – 30 MHz	100 m – 10 m	Skywave	Shortwave radio CB radio
Very-high frequency VHF	30 MHz - 300 MHz	10 m – 1 m	Line-of-sight	FM radio Television
Ultra-high frequency UHF	300 MHz – 3 GHz	1 m – 100 cm	Line-of-sight	Walkie-talkies Television



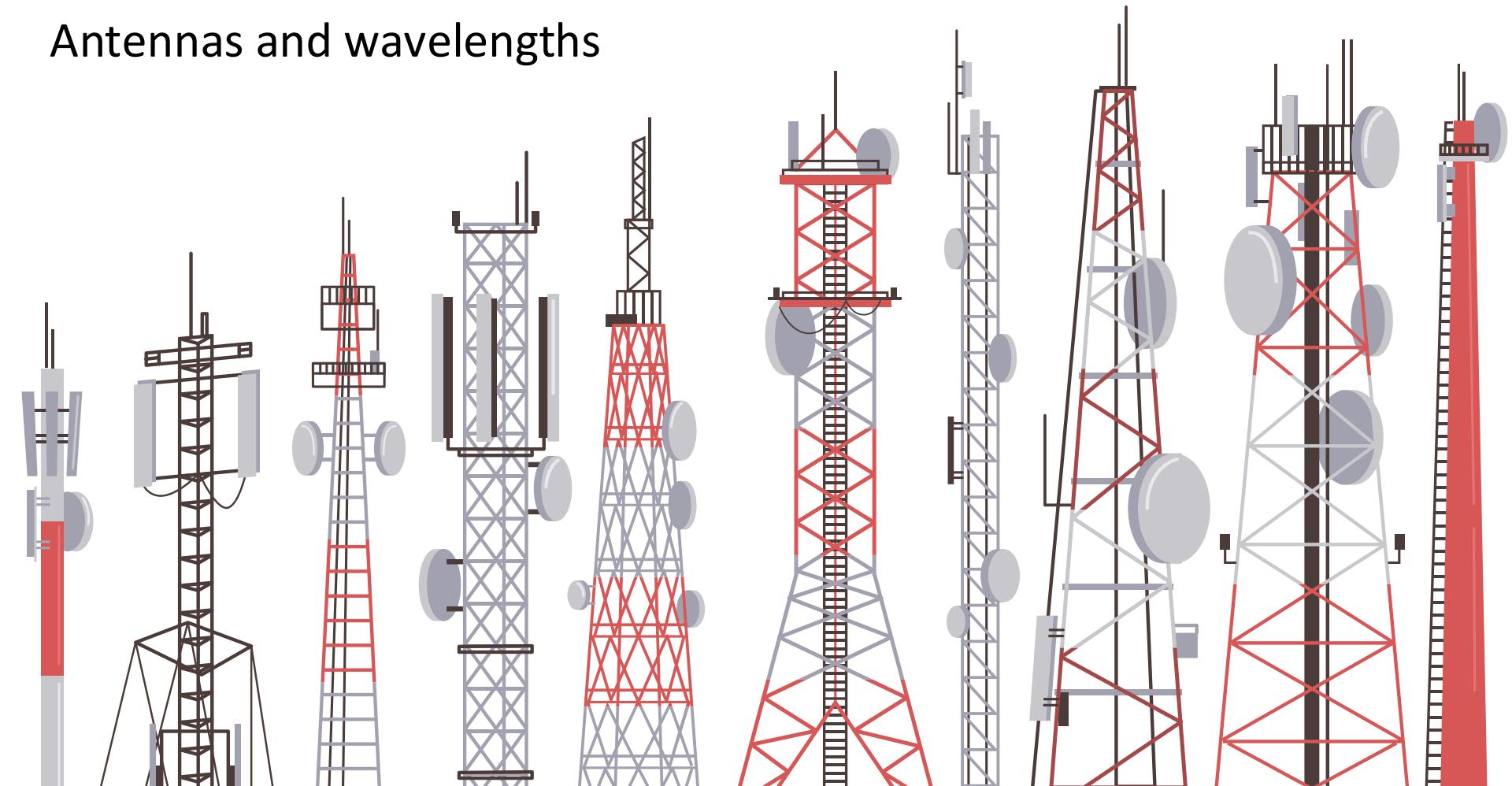


$$\text{speed of light} \div \text{frequency} = \sim 300 \text{ Mm/s} \div 150 \text{ MHz} = \sim 2 \text{ m}$$

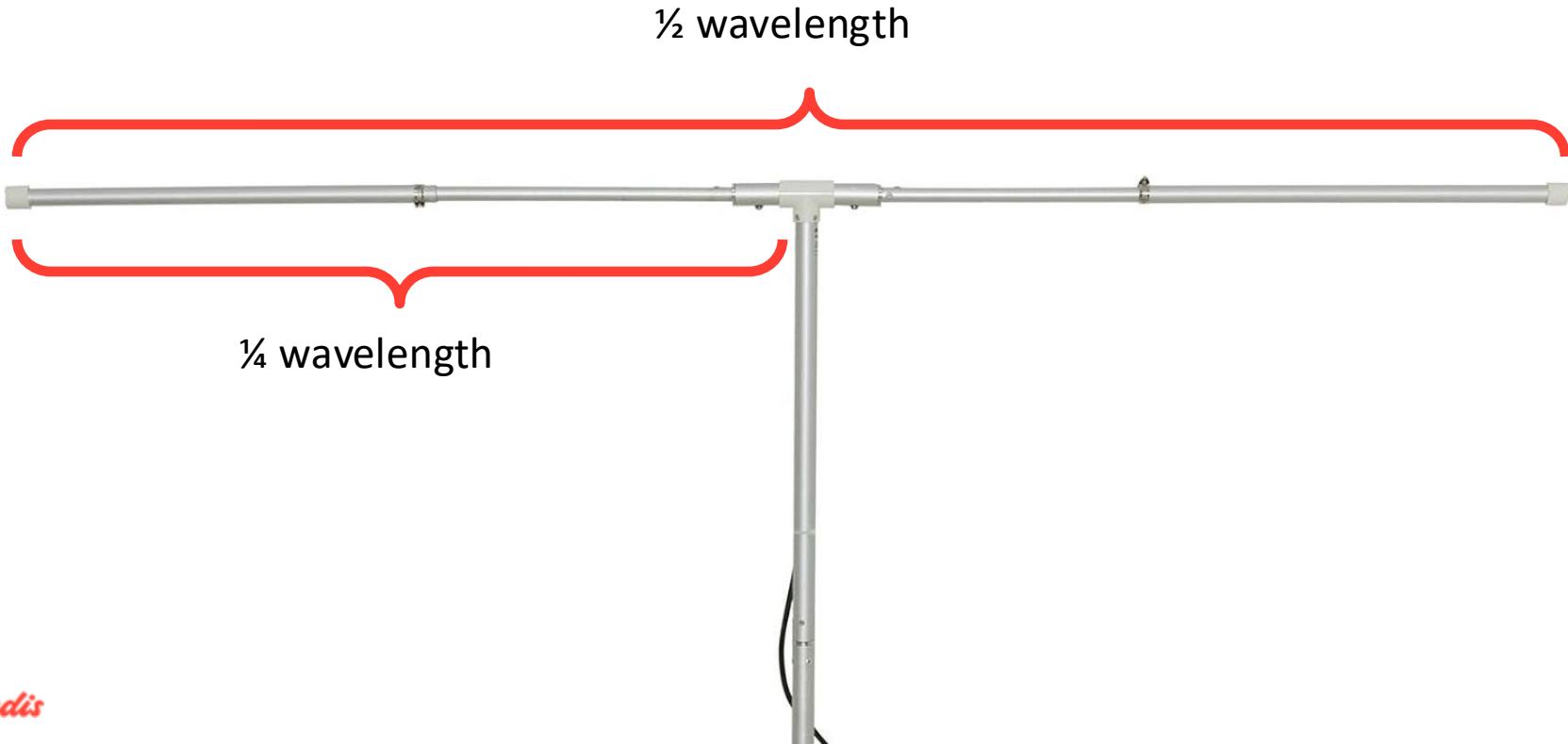
0.0dBV rms/div 250.0Hz /div Sq=5.000 kHz

Radio waves have a physical length

Antennas and wavelengths



A simple half-wave dipole antenna

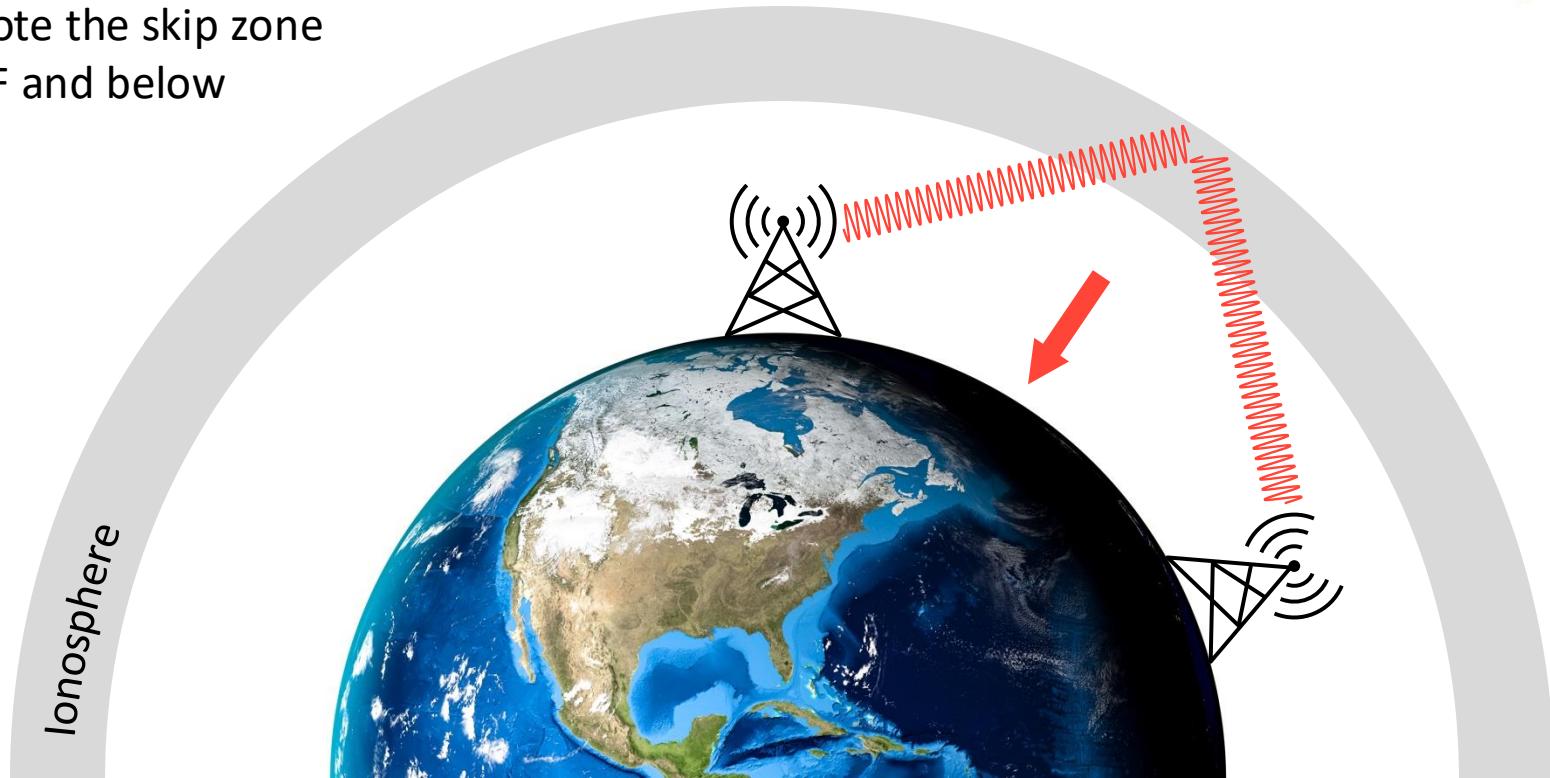


Radio is light



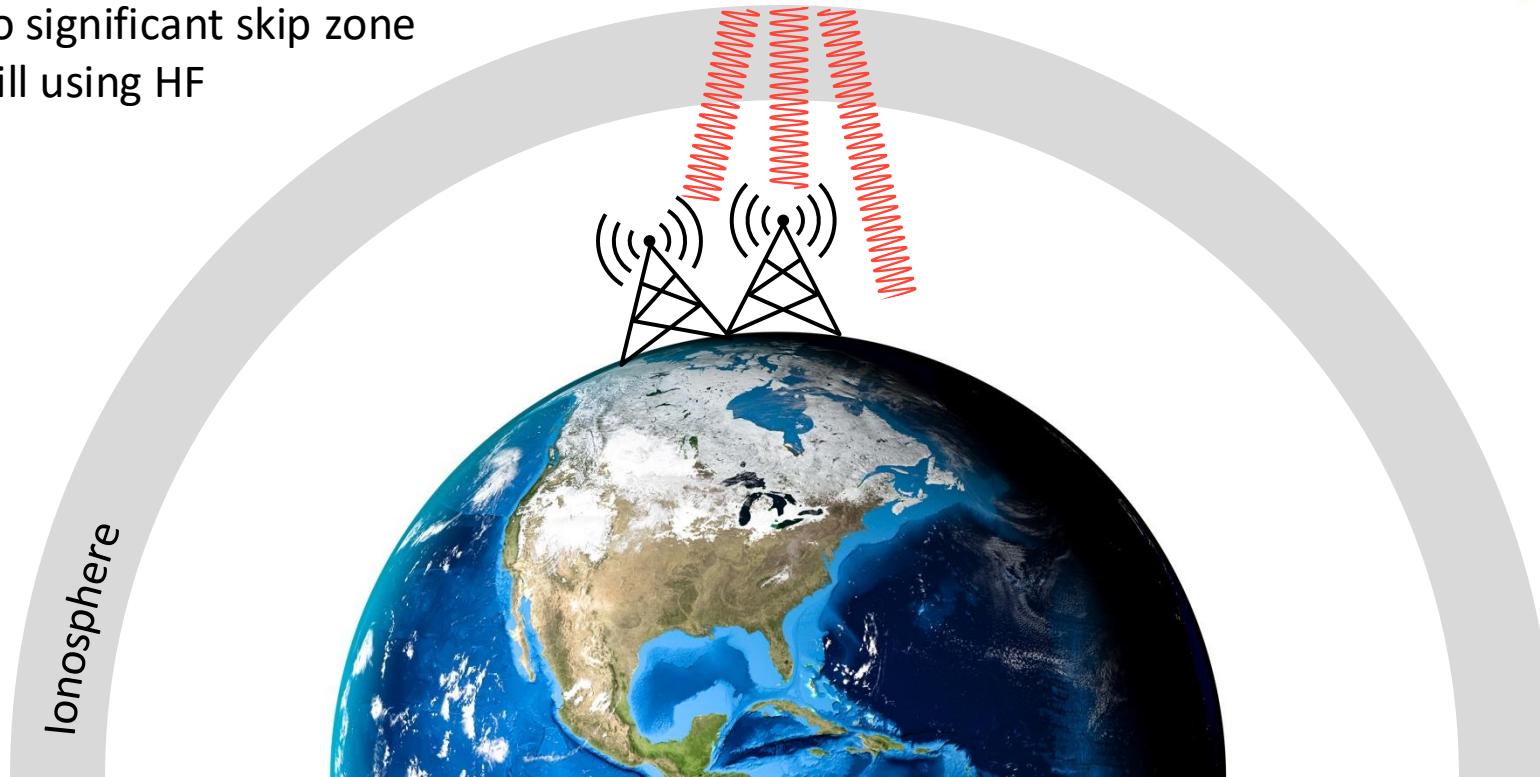
Skywave propagation

- Conditions in the ionosphere matter
- Note the skip zone
- HF and below



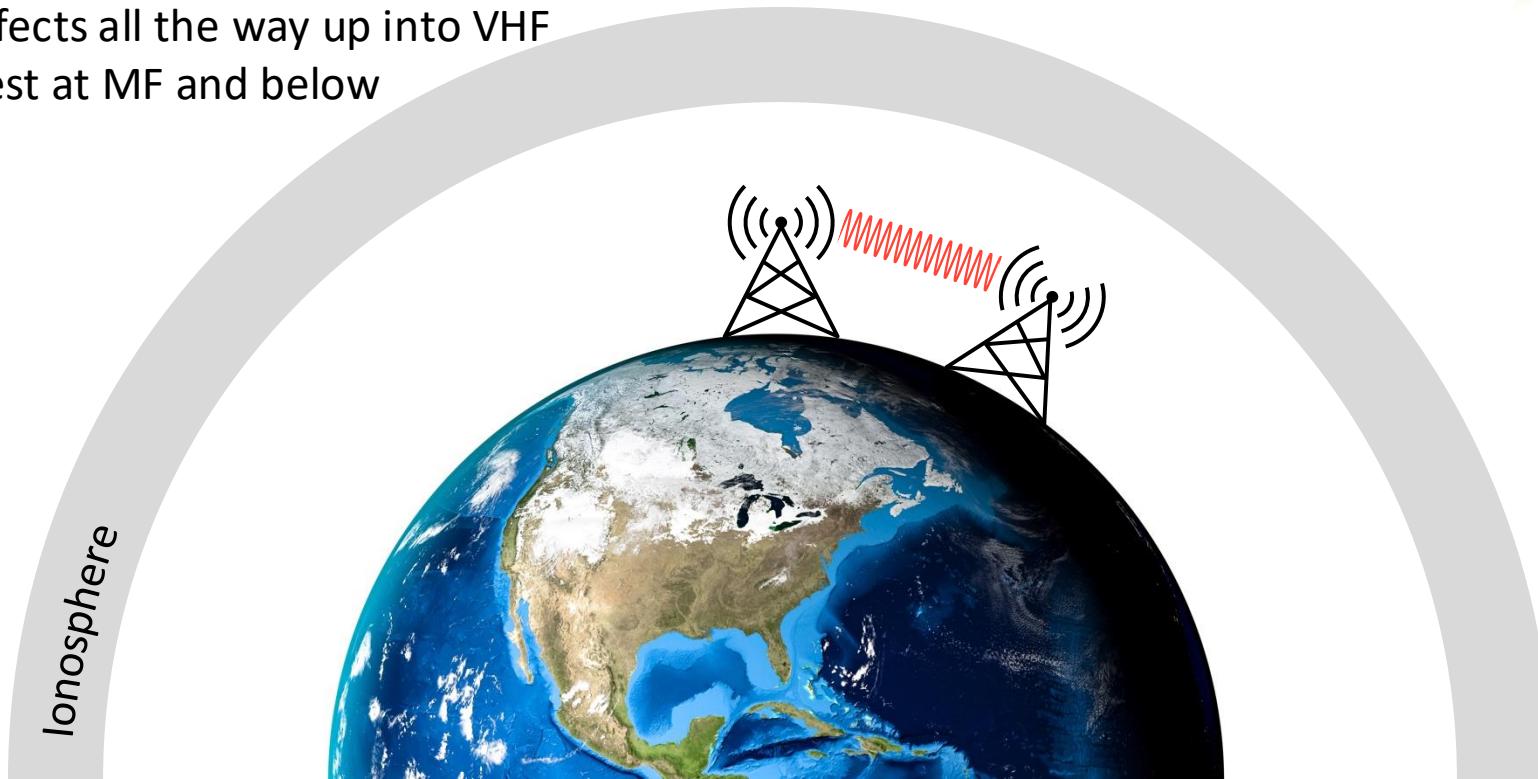
Near-vertical incidence skywave propagation

- Great for regional communication
- No significant skip zone
- Still using HF



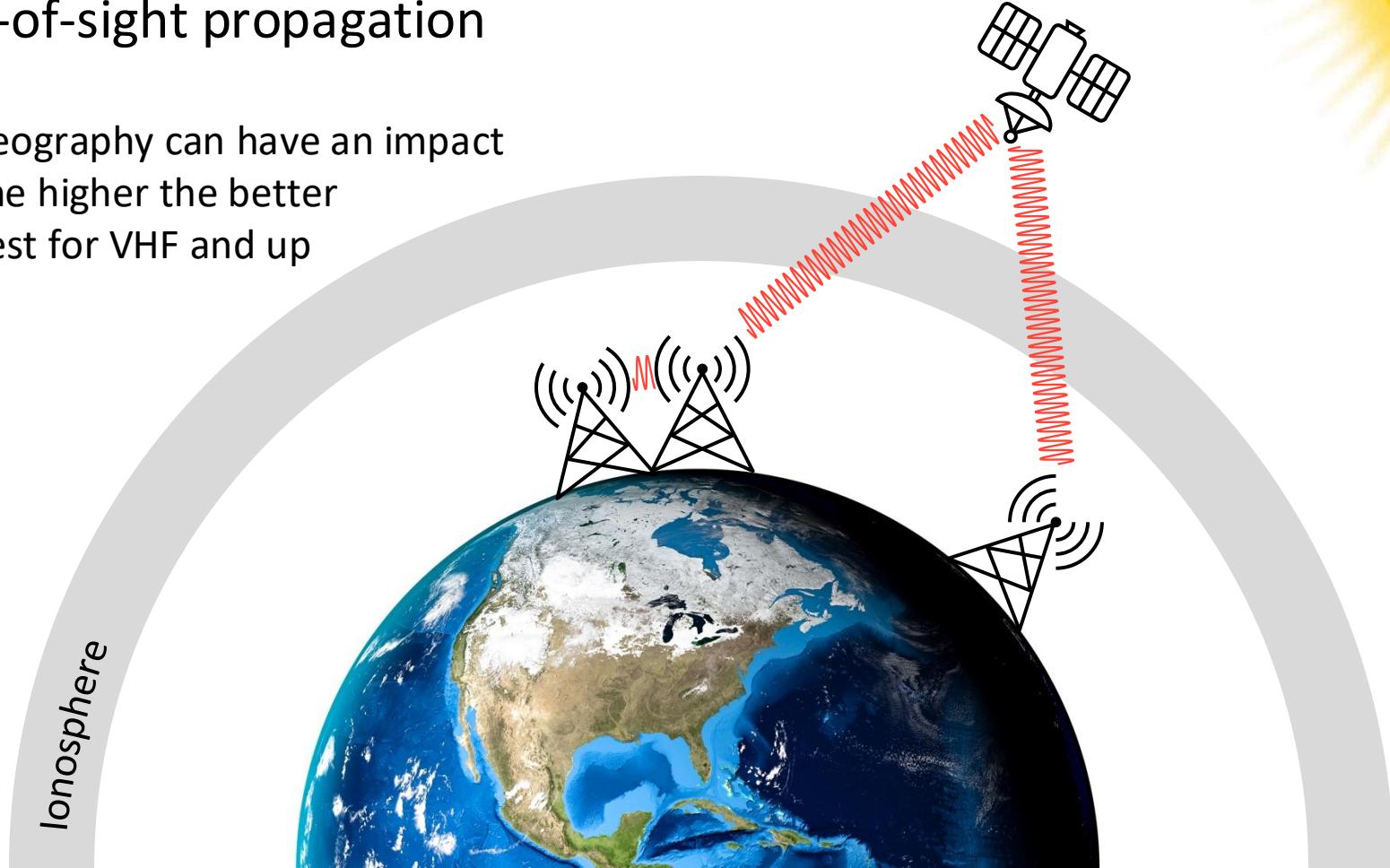
Groundwave propagation

- Conductivity of the surface matters
- Effects all the way up into VHF
- Best at MF and below



Line-of-sight propagation

- Geography can have an impact
- The higher the better
- Best for VHF and up



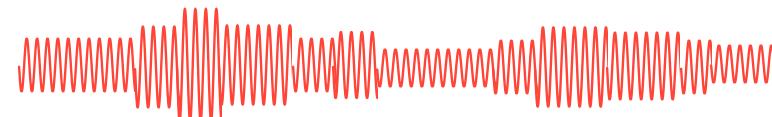
Carrier waves and signal modulation



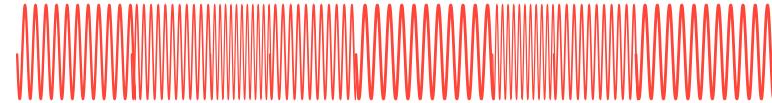
Continuous wave
CW



Amplitude modulation
AM



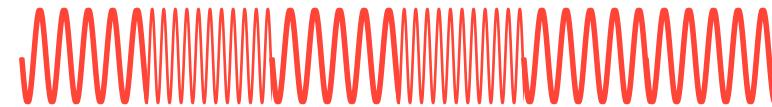
Frequency modulation
FM



Phase-shift keying
PSK



Frequency-shift keying
FSK





- THE DASHBOARD

Running the software

The Signals Dashboard

Welcome to the dashboard! Make sure you have your client running for a particular item and then select it.

Power Meter

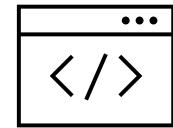
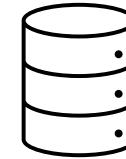
Plane Finder

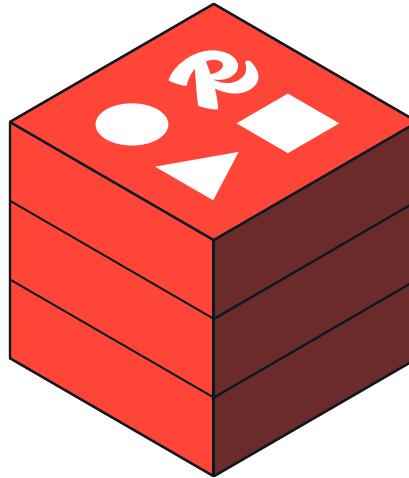
Packet Watcher

Weather



1010
1010





Memory First

Everything is in memory so
it's wicked fast.

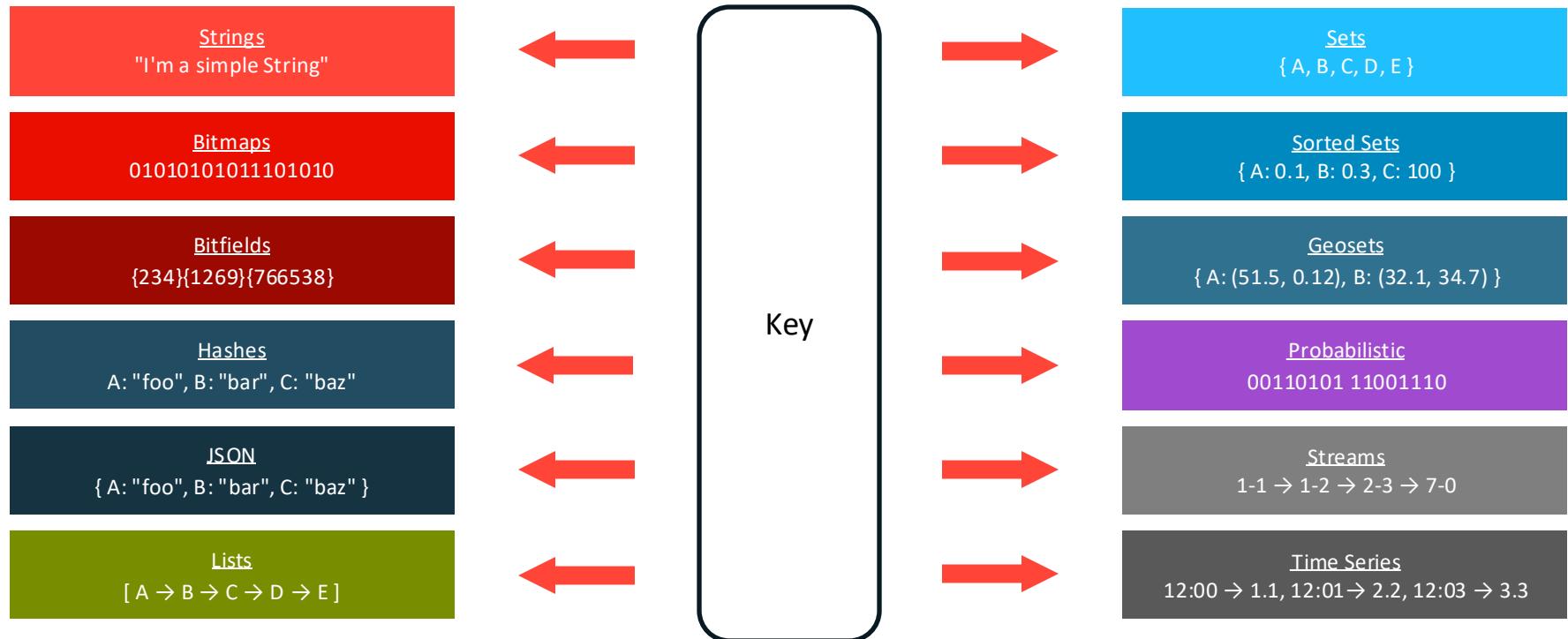
NoSQL

Serves up data structures
that we know and love.

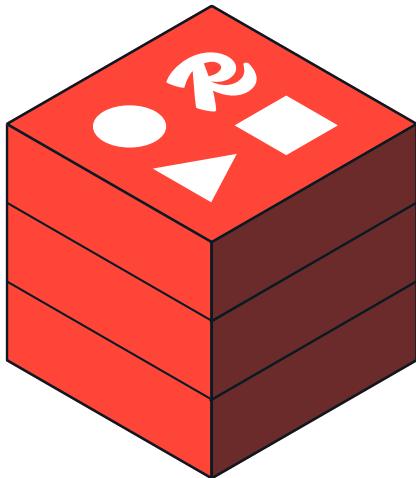
Database

Persistable
Clusterable
Replicatable

A giant hash table



Talking to Redis



SET foo bar

OK

GET foo

"bar"

PING

PONG



All Key Types

Filter by Key Name or Pattern



+ Key

Total: 8

10 min



STREAM	aprs:packets	No limit	50 KB
STREAM	rtl_433:Acurite-606TX	No limit	5 KB
STREAM	rtl_433:Acurite-5n1	No limit	69 KB
STREAM	rtl_433:Acurite-6045M	No limit	5 KB
STRING	foo	No limit	56 B
HASH	s1	No limit	80 B
HASH	s2	No limit	88 B
SET	rtl_433:models	No limit	112 B

SET

rtl_433:models

112 B Length: 3 TTL: No limit

9 min

Member

Search

Acurite-5n1



Acurite-606TX



Acurite-6045M



> CLI

Connecting...

Pinging Redis server on localhost:6379

Connected.

Ready to execute commands.

> SET foo bar [NX | XX] [GET] [EX seconds | PX milliseconds | EXAT unix-time-seconds | PXAT unix-time-milliseconds | KEPTTL]

> CLI

Command Helper

Profiler



Let us know what you think

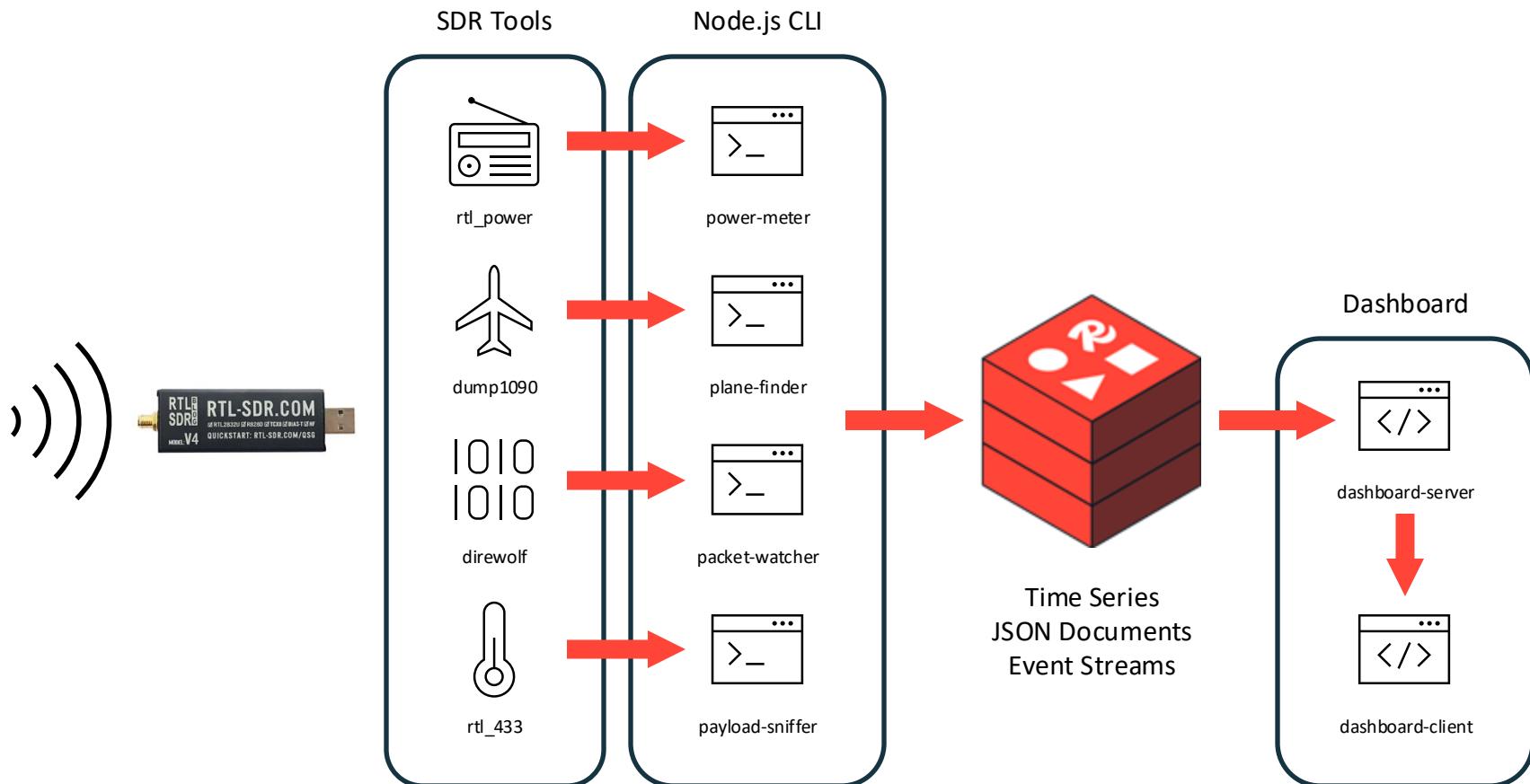
Accessing Redis from code

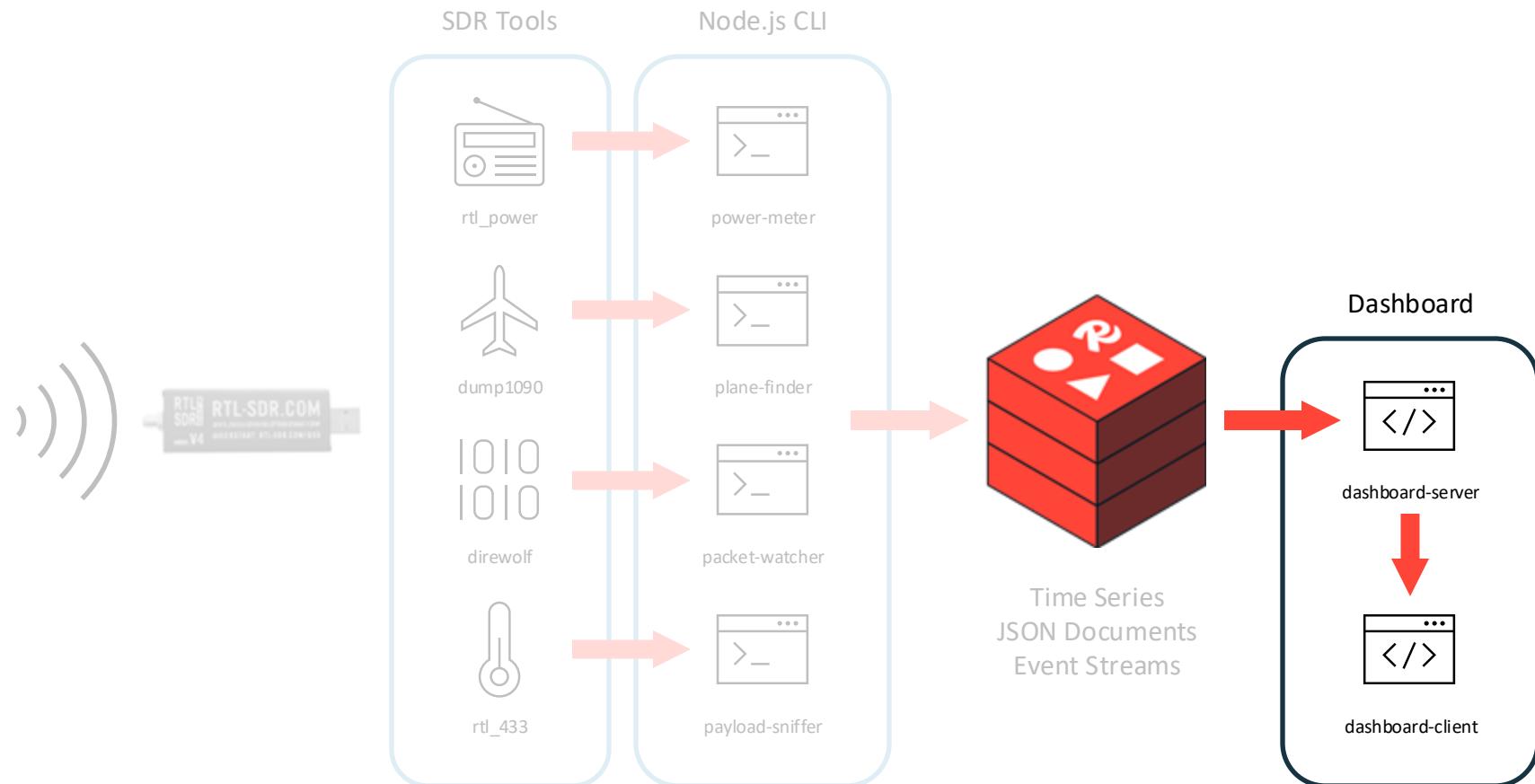
```
import { createClient } from 'redis'

const redis = await createClient({
  url: 'redis://alice:foobared@my.redis.server:6379'
}).connect()

await redis.set('foo', 'bar')

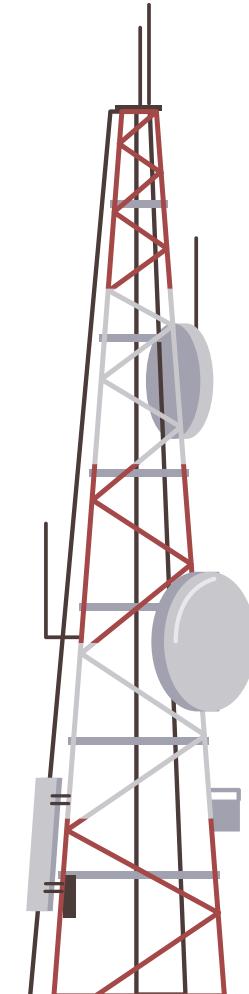
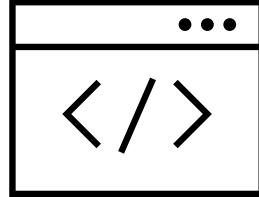
const value = await redis.get('foo')
```





Let's get hands on!

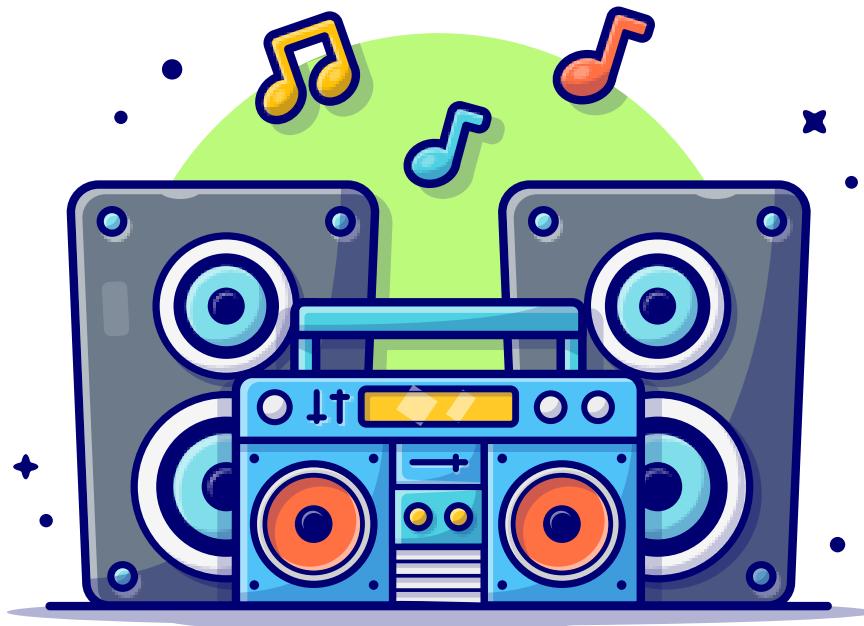
02-RUNNING-THE-DASHBOARD.md



- THE POWER METER

Monitoring broadcast radio signal strength

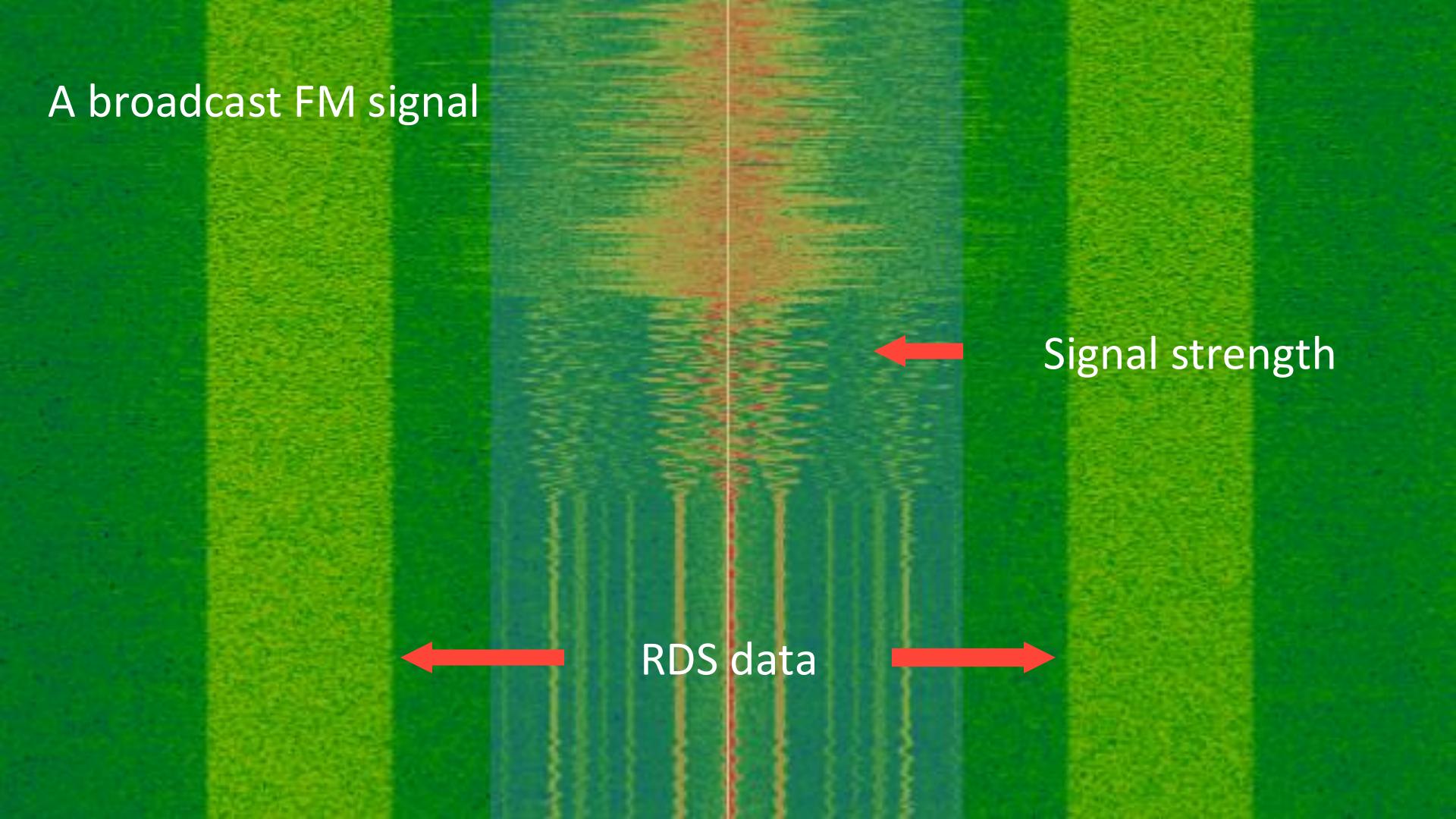
Broadcast FM radio



88.0 MHz to 108.0 MHz



A broadcast FM signal

A spectrogram visualization of an FM broadcast signal. The vertical axis represents frequency, and the horizontal axis represents time. The main signal is a dense band of green and yellow, centered around a red vertical line. To the left of this central band, there is a distinct vertical strip of blue, which contains several vertical lines of data, labeled as RDS data. To the right of the central band, there is another vertical strip of blue, labeled as Signal strength, with a red arrow pointing towards it.

Signal strength



RDS data



Signal strength is great for timeseries

Each Redis key contains a series of values over time

power:100.7MHz

20:55:05.007
5.1 dB

20:55:06.389
5.2 dB

20:55:08.420
3.7 dB

20:55:09.012
3.8 dB

Can query distinct values for a specified time-range

20:55:05.007
5.1 dB

20:55:06.389
5.2 dB

20:55:08.420
3.7 dB

20:55:09.012
3.8 dB

Or get the latest value

And aggregate to a desired resolution

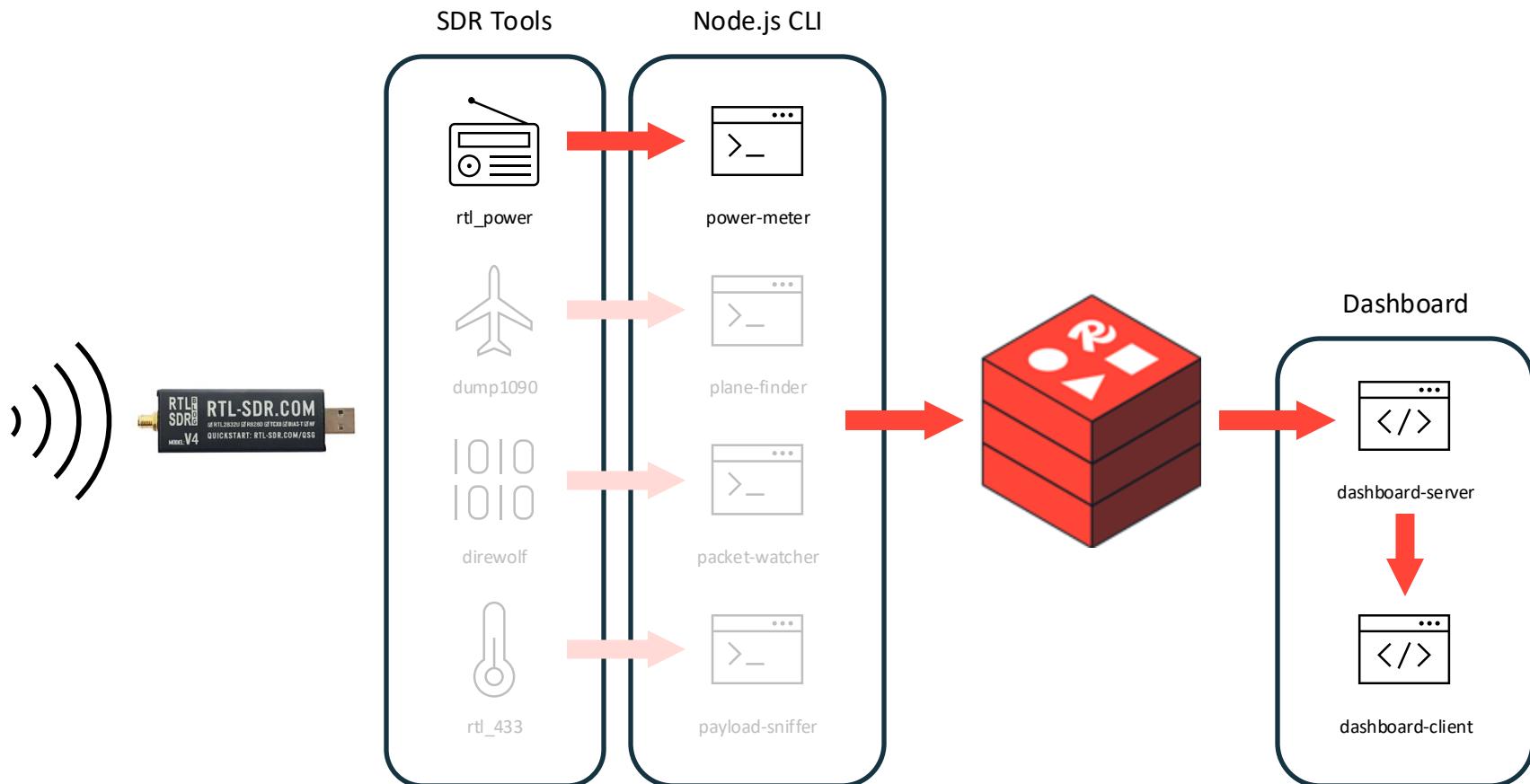
20:55:05.000
5.152 dB

20:55:06.000
4.922 dB

20:55:07.000
3.854 dB

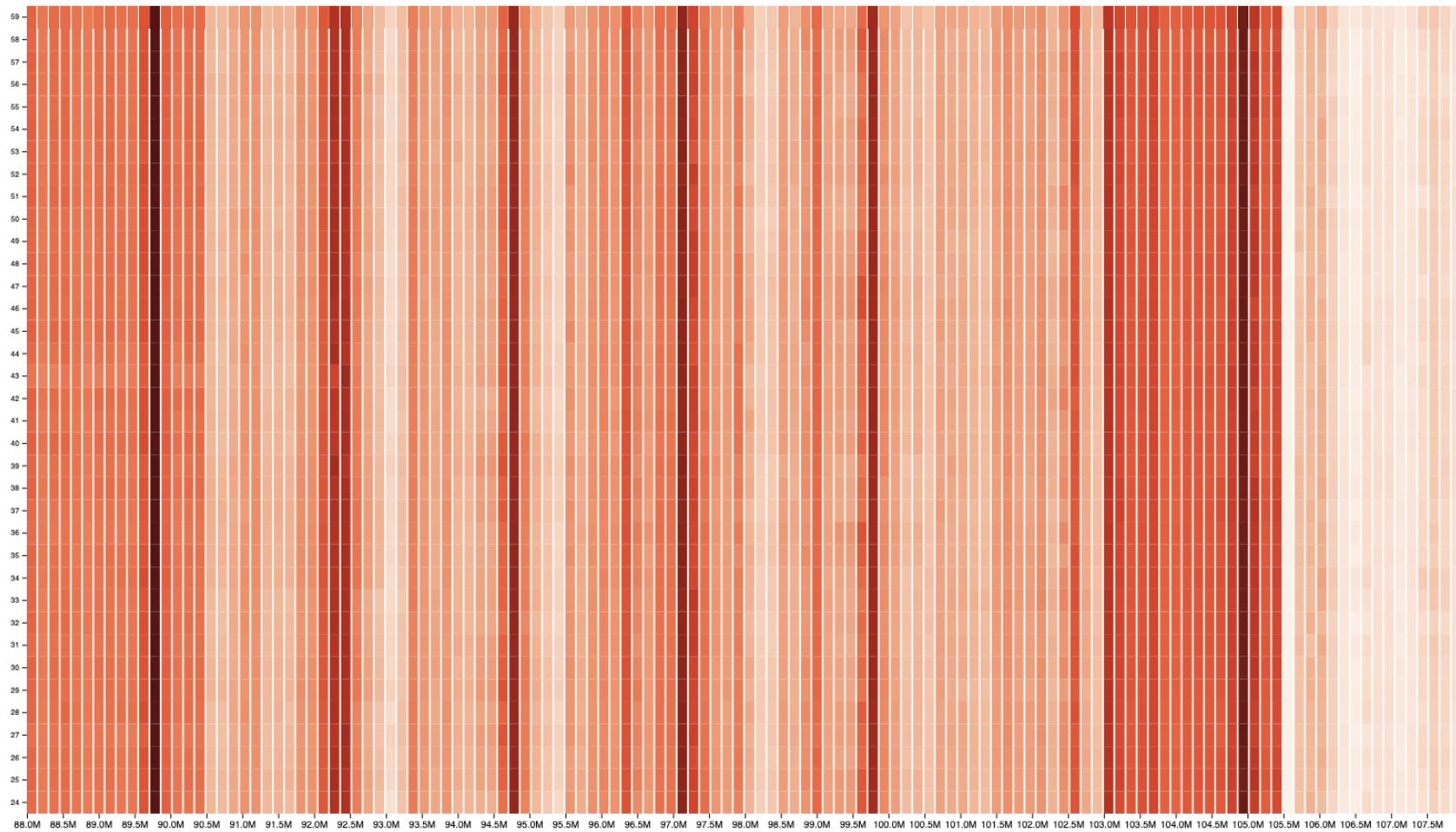
20:55:08.000
3.644 dB

20:55:08.000
3.8 dB



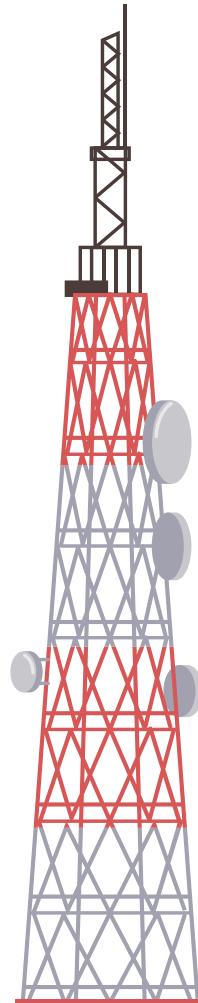
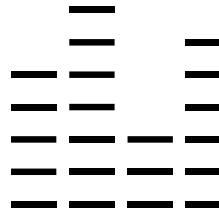
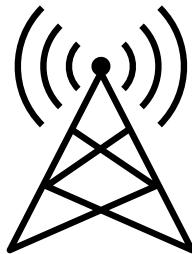
Redis

Signal Strength



Let's get hands on!

03-RADIO-STATION-POWER.md



- THE PLANE FINDER

Receiving aircraft flight information



ADS-B



ICAO ID

A835AF



CALLSIGN

N628TS



ALTITUDE

22,100 ft.



SPEED

395 kn.



HEADING

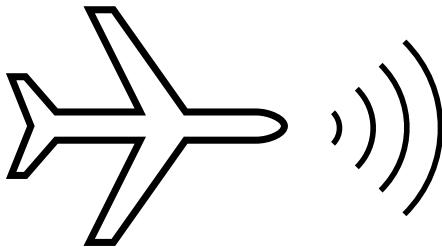
344°



LOCATION

39.963°N
83.205°W

JSON is an easy way to store aircraft status



aircraft:A835AF

aircraft:A835AF

aircraft:A835AF

```
{  
  "icaold": "A835AF",  
  "latitude": 39.963,  
  "longitude": -83.205  
}
```

```
{  
  "icaold": "A835AF",  
  "altitude": 22100,  
  "latitude": 39.963,  
  "longitude": -83.205  
}
```

```
{  
  "icaold": "A835AF",  
  "altitude": 22000,  
  "latitude": 39.817,  
  "longitude": -83.251,  
  "callsign": "N628TS"  
}
```

Searching for aircraft using Redis Query Engine

aircraft:A835AF

```
{  
  "icaoid": "A835AF",  
  "altitude": 22100,  
  "callsign": "N628TS"  
}
```

aircraft:ABDD1C

```
{  
  "icaoid": "ABDD1C",  
  "altitude": 32000,  
  "callsign": "DAL2828"  
}
```

aircraft:A742DD

```
{  
  "icaoid": "A742DD",  
  "altitude": 22000,  
  "callsign": "UAL1492"  
}
```

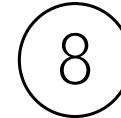
Creates indices for assorted data types



TEXT



TAG



NUMERIC



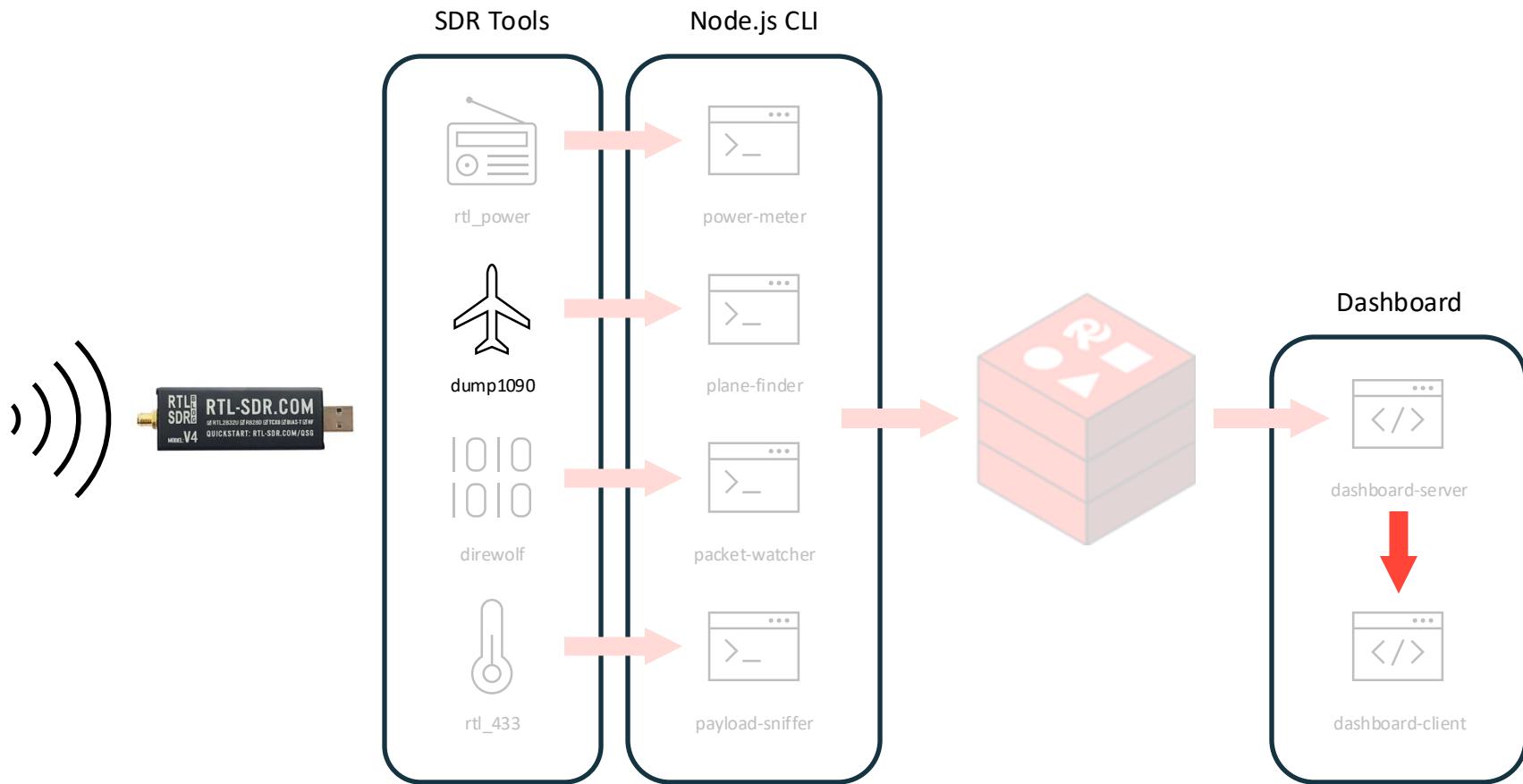
GEO



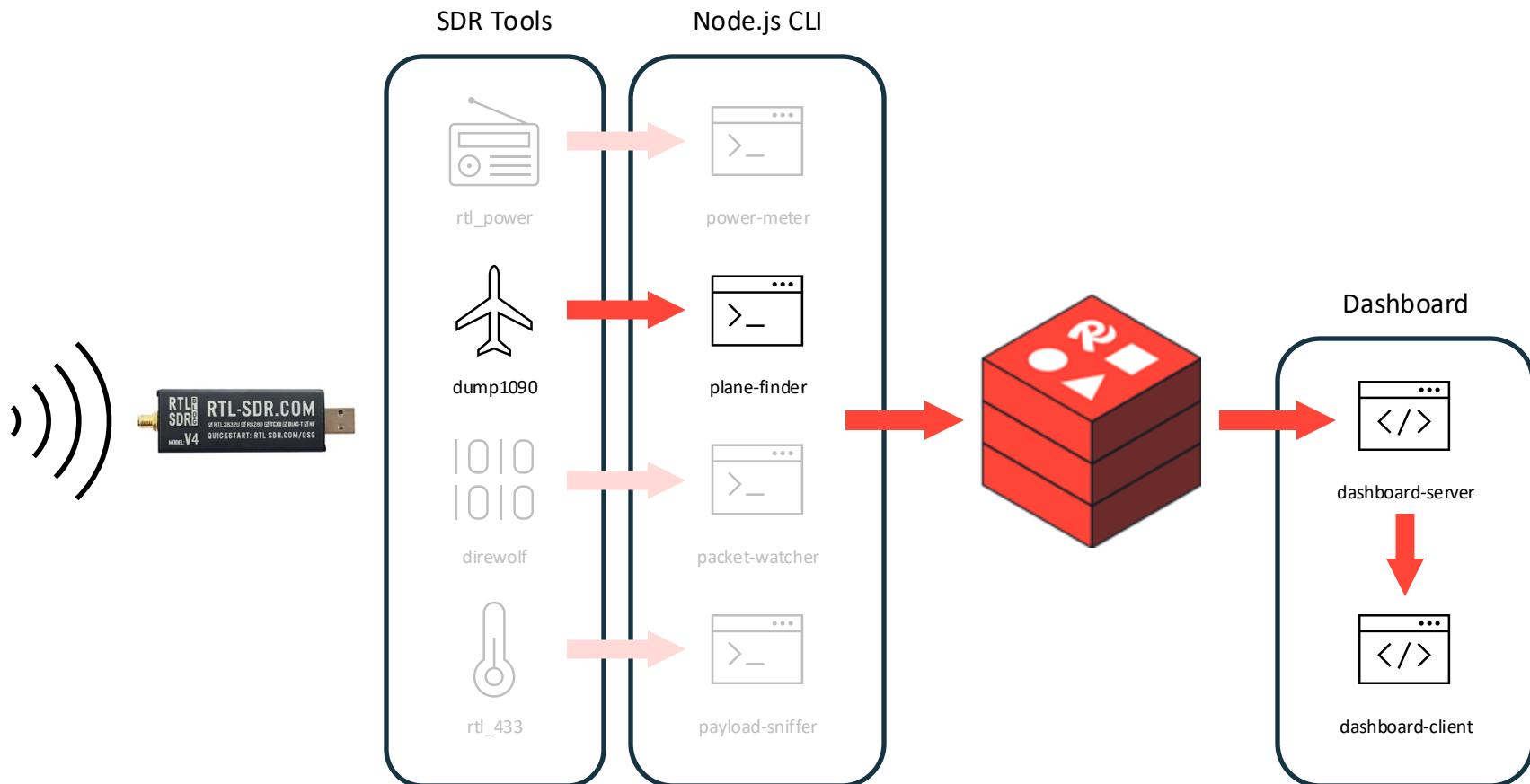
VECTOR

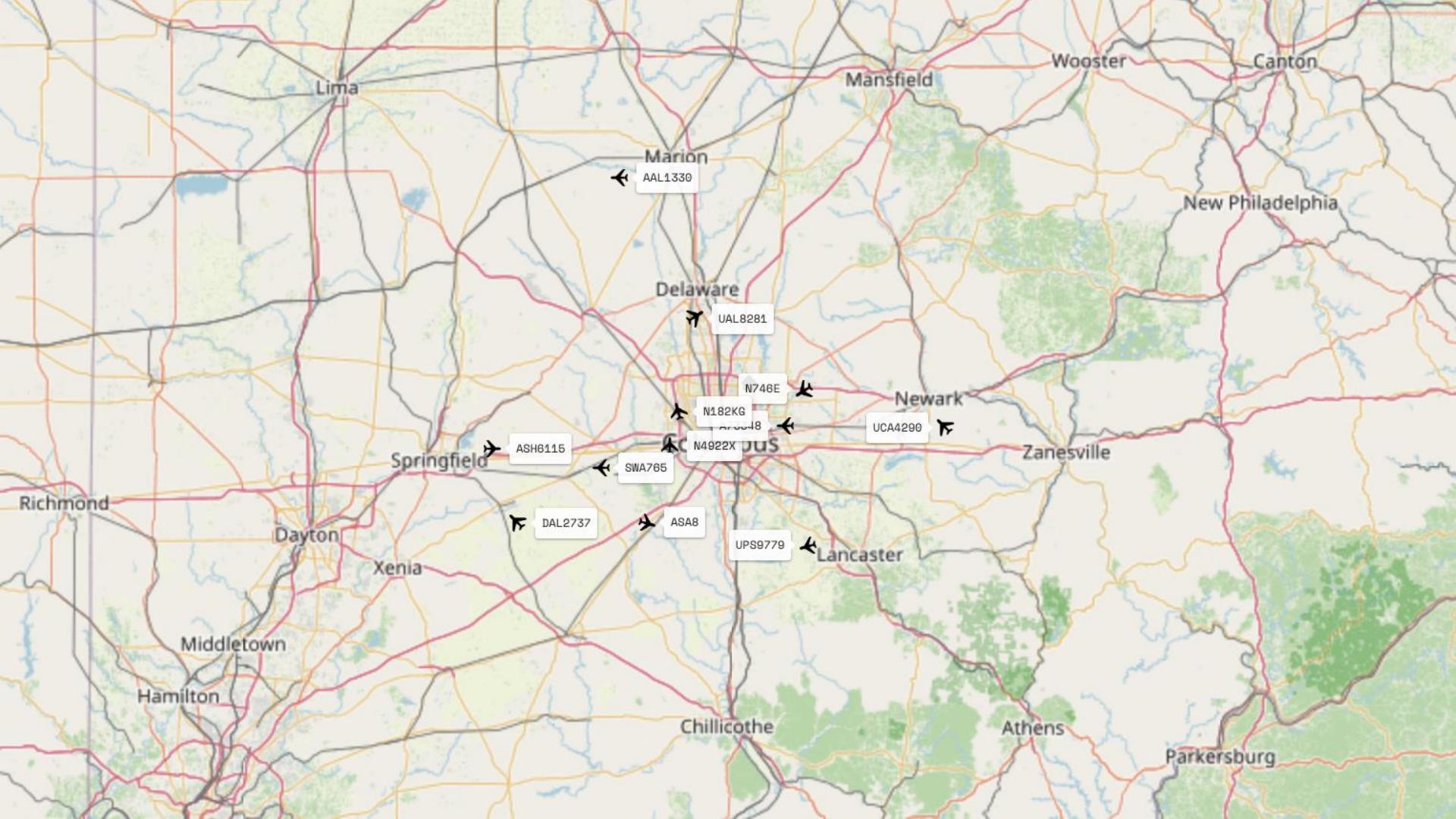
Queried using a simple query language

```
@callsign:{ DAL2828 }  
@altitude:[ 20000 30000 ]  
*
```



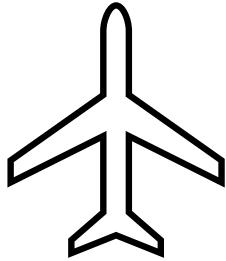
Tot: 15 Vis: 15 RSSI: Max -25.3+ Mean -32.4 Min -34.7- MaxD: 0.0nm+ /												
Hex	Mode	Sqwk	Flight	Alt	Spd	Hdg	Lat	Long	RSSI	Msgs	Ti	
A446F1	A2	1640	LXJ375	22100	395	344	39.963	-83.205	-31.4	347	04	
A296F4	A2	6746	N266TD	5575	189	273	39.994	-82.956	-30.5	123	30	
A87108	A2	6702	PAC369	30650	523	051	40.384	-83.412	-33.3	286	00	
A04E60	A2		RPA4747	30000	394	271	40.285	-82.991	-32.4	265	29	
A655F1	A2	5612	RVJ55160	6075	296	098	40.097	-82.960	-25.3+	454	08	
AAA244	A2	1040	UAL296	31000	494	101	40.369	-83.046	-31.5	1026	12	
AC0417	A2	4063	ASH6330	9725	268	272	40.009	-83.261	-34.0	410	22	
A2587B	A2	1535	XSR250	39000	534	059	40.560	-82.938	-33.6	688	52	
AB2AE1	A2	2231		35025	575	059	40.676	-83.134	-34.7-	346	20	
A8A70F	A0		DABJET55	36025	347	264	39.756	-83.036	-32.9	51	31	
AA1A1F	A2		ABX3185	31000					-33.5	571	43	
ABBBC7	A2		ASA1108	31025	517	098	40.238	-82.689	-32.0	885	44	
ADA94D	A2	4164	N98FE	40000	348	239	39.852	-83.509	-34.1	2350	00	
A575D3	A2		JRT51	43000	399	209	39.947	-83.783	-33.1	852	15	
ACBCFC	A2		N92MZ	20000	210	281	40.257	-83.377	-33.7	1632	26	





Let's get hands on!

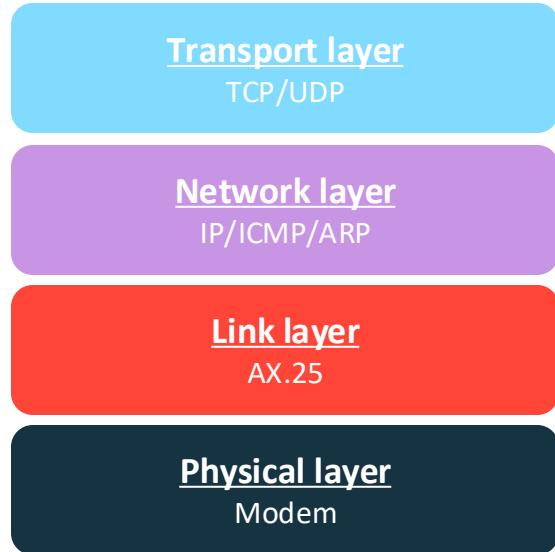
04-AIRCRAFT-FLIGHT-DATA.md



- THE PACKET WATCHER

Decoding APRS data packets

Packet data networks over amateur radio



AMPRNet

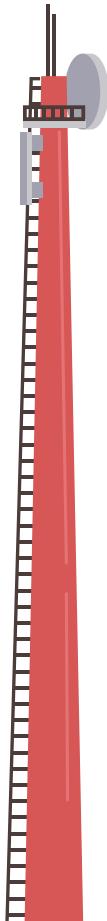
Implements point-to-point comms
Winlink, packet BBSs, chat

VHF/UHF
1200/9600 baud

HF
300 baud

AFSK
Bell tones

Connects to hardware or software modem
using USB or serial ports



APRS is a simplified version of AX.25

Link layer

APRS

Doesn't implement point-to-point comms

Just broadcasts packets

Physical layer

Modem

VHF/UHF

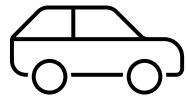
1200/9600 baud

Connects to hardware or software modem
using USB or serial ports

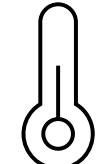


APRS is used to send messages and share status

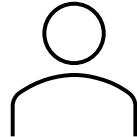
Personal vehicles



Weather stations



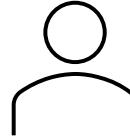
Individual operators



Radio infrastructure



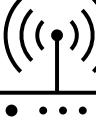
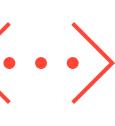
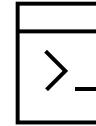
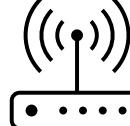
Individual operators



APRS repeaters



Internet gateways and applications



Track callsign: Clear

Search ?

Address, city or locator: Clear

Search ?

Show last:

Track tail length:

Wx: 25°F 78% 1020 mbar 4.9 MPH W

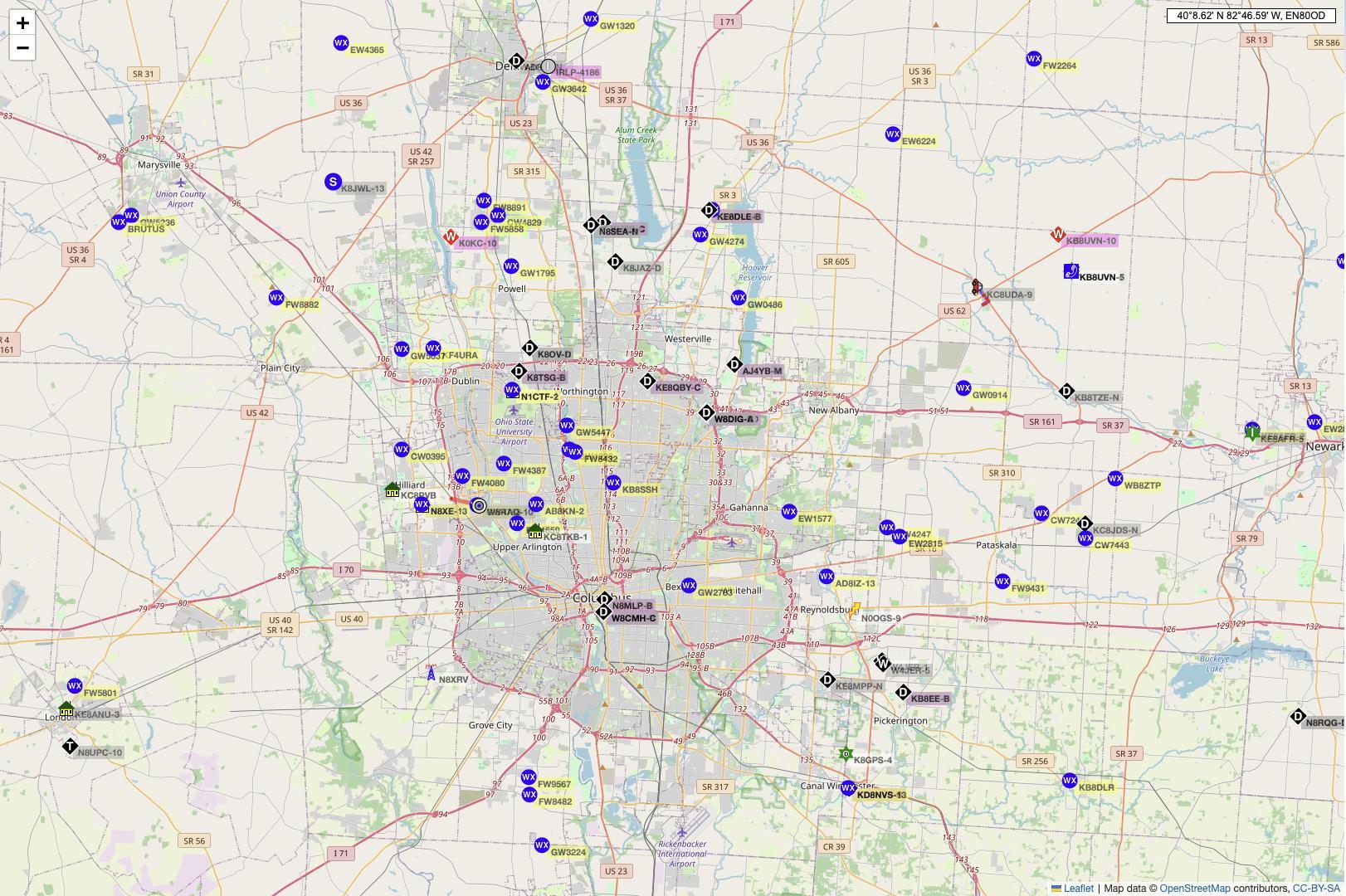
Other views:

- Station info
- Raw packets
- Status packets - Beacon packets
- APRS/CWOP weather - Telemetry
- Messages - Bulletin board
- Prefix browsing
- Google Earth KML ?
- Data export tool
- Preferences - My account

Information:

Stations currently moving · User guide · FAQ ·
[Blog](#) · [Discussion group](#) · [Linking to aprs.fi](#) ·
 AIS sites · Service status · Database statistics ·
[Advertising on aprs.fi](#) · Technical details · API ·
[Change log](#) · Planned changes · Credits and
 thanks · Terms of Service · iPhone/iPad APRS

idle: points updated 6, deleted 0



Saving APRS packets in an event stream



1655410429872-1



1655410429872-0



1655410429867-0



1655410429863-1



1655410429863-0



1655410427912-0



1655410427590-0

timestamp in
milliseconds



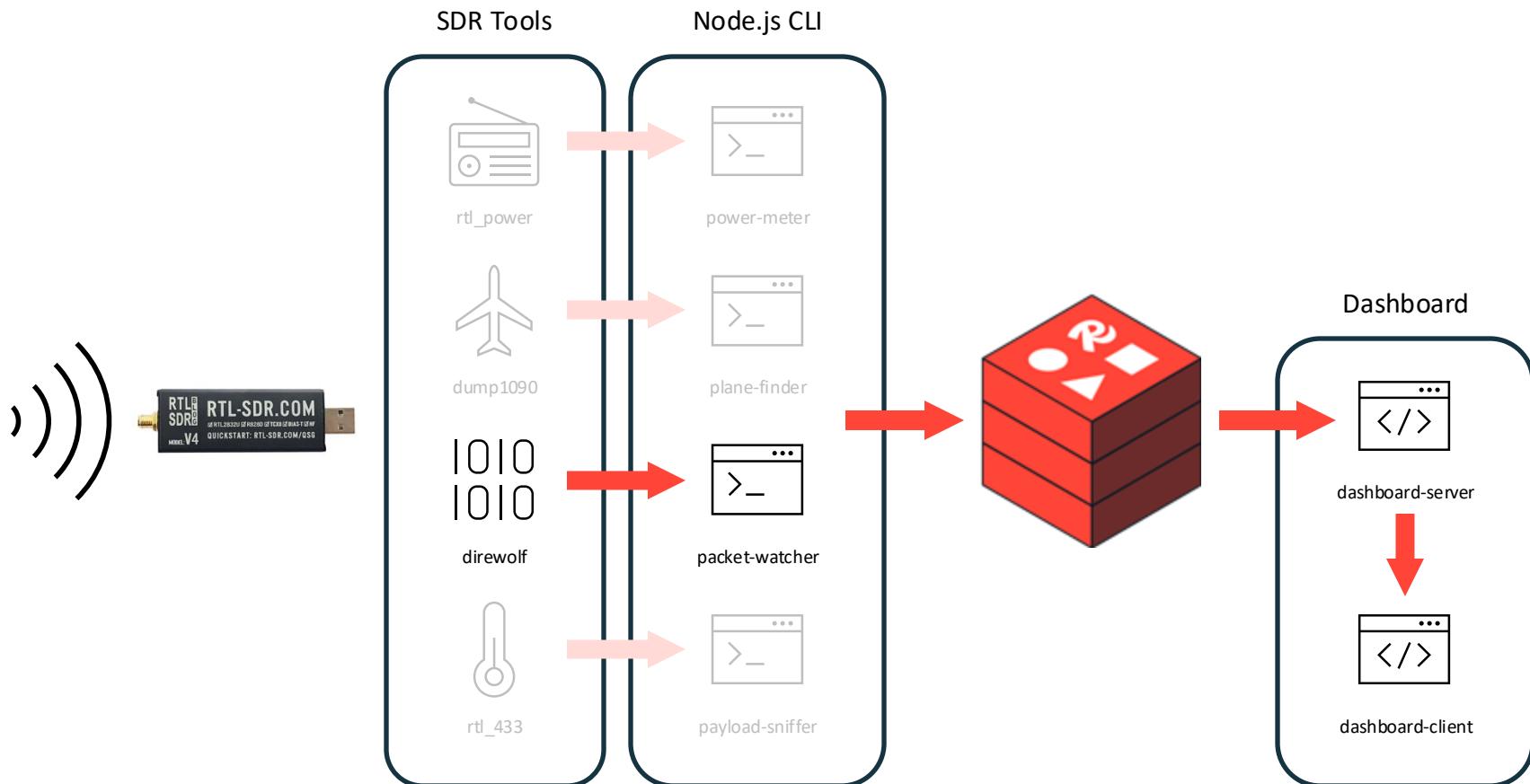
sequence
number



16554109863-0

Timestamp	2024-12-20T19:57:43.763Z
Destination	APNU19
Source	K8LGN-5
Path	K8GPS-4,WIDE2-1
Payload	>111111z K8LGN Logan Ohio HVARC



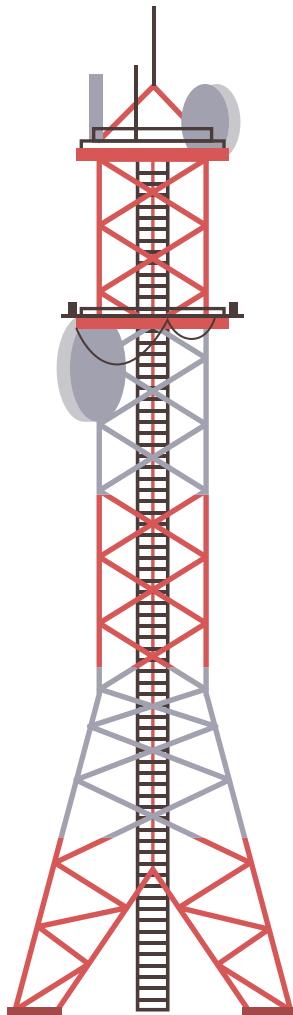
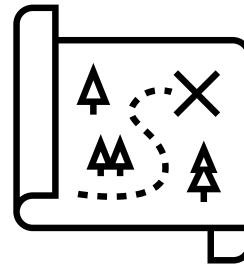
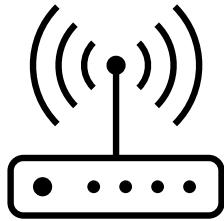


Recent APRS Packets

Timestamp	Dest	Source	Path	Information
2024-12-20T20:02:53.722Z	APN391	K8JWL-13	KB8UVN-7,K8GPS-4,WIDE2	!!002C0008018A2C0127A302B7038E01A2016203980005005E
2024-12-20T20:01:52.794Z	APN000	N0OGS-9	K8GPS-4,WIDE1,WIDE2-1	=3957.21N\08247.57WJ347/001VGC-N76
2024-12-20T20:01:37.921Z	TPPV0Y	K04MUL-9	KB8UVN-7,WIDE1,K8GPS-4,WIDE2	`o&Eps\$>/`"6z}_2
2024-12-20T20:01:20.906Z	APRS	KE8WSC-8	W8BLV,WIDE1,WB8VSU-10,K8GPS-4,WIDE2	!3933.68N/08421.48Wf348/000/A=000554KE8WSC Dale 878
2024-12-20T19:59:44.054Z	APMI06	N8HKO-3	KB8UVN-7,K8GPS-4,WIDE2	La Rue, OH, 60watts, HAAT 57', 6db gain, 1/2" hardline
2024-12-20T19:59:21.111Z	TPPU9T	K04MUL-9	K8GPS-4,WIDE1,WIDE2-1	`o`{p6_>/`"6q}_2
2024-12-20T19:58:42.866Z	TPPV0S	K04MUL-9	K8GPS-4,WIDE1,WIDE2-1	`o_zn4g>/`"6k}_2
2024-12-20T19:57:51.532Z	APN391	K8JWL-13	KB8UVN-7,K8GPS-4,WIDE2	!!004100F9018B2C0127A202B6038E01A20162039300050034
2024-12-20T19:57:43.763Z	APNU19	K8LGN-5	K8GPS-4,WIDE2-1	>111111z K8LGN Logan Ohio HVARC
2024-12-20T19:57:33.392Z	SYUR1Z	KD6XH-2	K8GPS-4,WIDE1	`o]N1!:>/`"6r}_1
2024-12-20T19:57:15.471Z	APMI06	K8GPS-4	WIDE2-2	T#077,167,154,061,013,000,00000000
2024-12-20T19:56:52.598Z	APN000	N0OGS-9	K8GPS-4,WIDE1,WIDE2-1	=3957.21N\08247.57WJ347/001VGC-N76
2024-12-20T19:56:19.263Z	APMI06	K8GPS-4	WIDE2-2	@202008z3951.81N/08247.94WWX3in1Plus2 U=12.5V, T=55.5F http://www.k8gps.com
2024-12-20T19:56:16.420Z	APNU19	K8LGN-5	K8GPS-4,WIDE2-1	!3930.40N/08224.59WPHG4420 W2 OHn-N/Logan UI-Digi
2024-12-20T19:56:05.726Z	SY5X0R	KF8CON-9	K8GPS-4,WIDE1	`nV)1!}>/`"6U}_4
2024-12-20T19:54:28.702Z	APTT4	W8ALC	N8HKO-3,WIDE1,KB8UVN-7,K8GPS-4,WIDE2	T#539,255,094,071,069,069,10111011
2024-12-20T19:53:28.854Z	APTT4	N1CTF-2	KB8UVN-7,WIDE1,K8GPS-4,WIDE2	/195310z4005.62N/08304.46W_022/003g003t038r135P304h90b101301274X181jSIG
2024-12-20T19:53:21.620Z	APTT4	N1CTF-2	WIDE1-1,WIDE2-1	/195310z4005.62N/08304.46W_022/003g003t038r135P304h90b101301274X181jSIG
2024-12-20T19:52:54.008Z	APN391	K8JWL-13	KB8UVN-7,K8GPS-4,WIDE2	!!003F00E8018C2C0127A102B6038E01A20162038E00050043
2024-12-20T19:52:45.484Z	APMI06	K8EMA-4	WB8VSU-10,K8GPS-4,WIDE2	@201952z4016.61N/08409.99W-WX3in1Plus2.0

Let's get hands on!

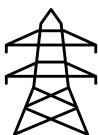
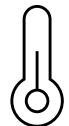
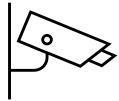
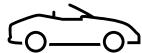
05-APRS-DATA-PACKETS.md



- THE PAYLOAD SNIFFER

Observing IoT data payloads

Lots of devices are emitting data



315 MHz

Automotive applications

Key fobs, tire pressure sensors

345 MHz

Security systems

Alarm systems, locks

433 MHz

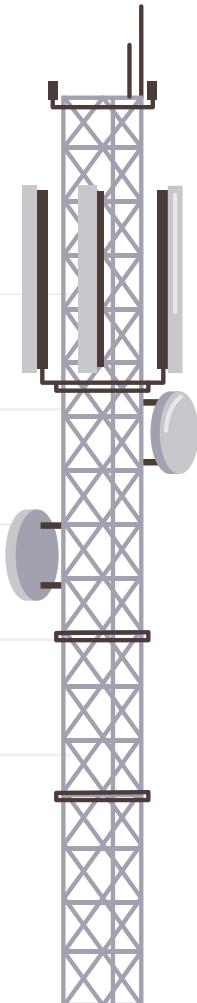
IoT devices

Home automation, doorbells, weather stations, etc.

915 MHz

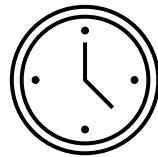
Industrial, commercial, and LoRa

Utility meters, various industrial sensors, Meshtastic, Reticulum

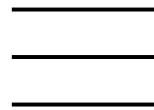


How to store this stuff

Timeseries



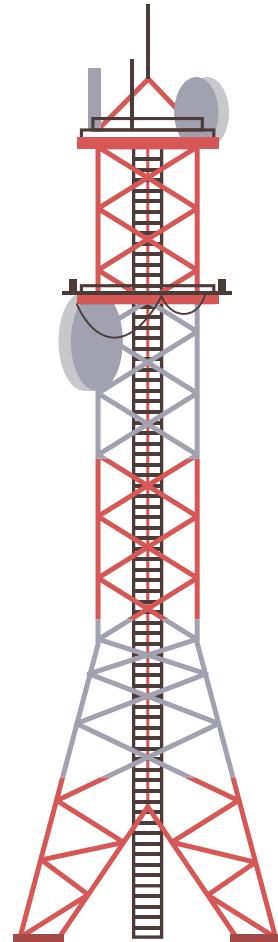
Event streams

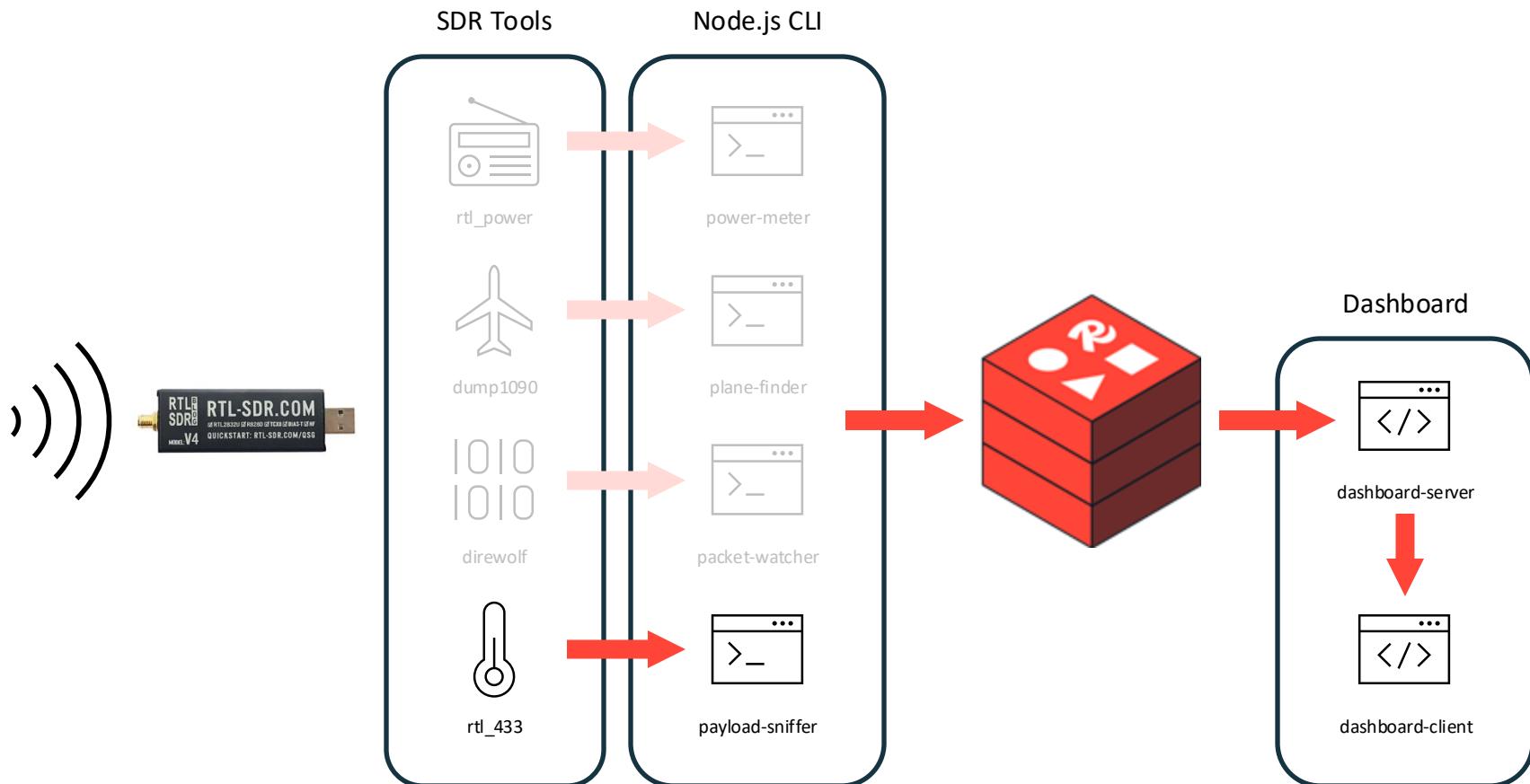


One event stream per device



Devices stored in a set





Current Weather Conditions

Select a device

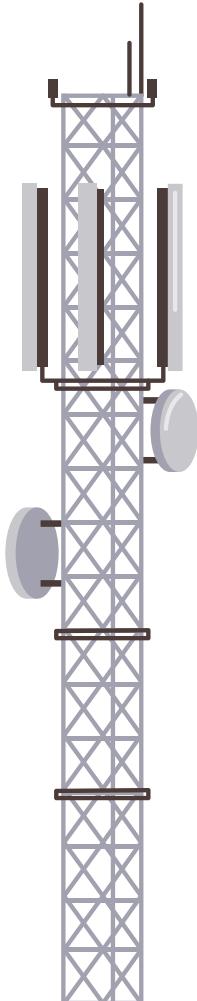
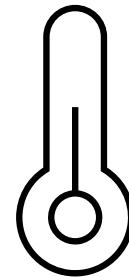
Acurite-5n1

	Temperature	69 °F
	Humidity	45%
	Wind	0 mph ESE
	Rainfall	0.09 in

Let's get hands on!

05-IOT-DATA-PACKETS.md

1010
1010



- CLOSING

That's it!

Would you like to know more?



ARRL
The National Association
for Amateur Radio
arrl.org



Redis Discord Server
discord.gg/redis

Redis
redis.io
university.redis.io

THANKS



Guy Royse

Developer Advocate

Redis

-  @guy.dev
-  @guyroyse
-  github.com/guyroyse
-  guy.dev
-  W8GUY

Redis



Plucking Data from Thin Air with Software Defined Radio

