Write-up for an ADS-B Installation

20180824/BM

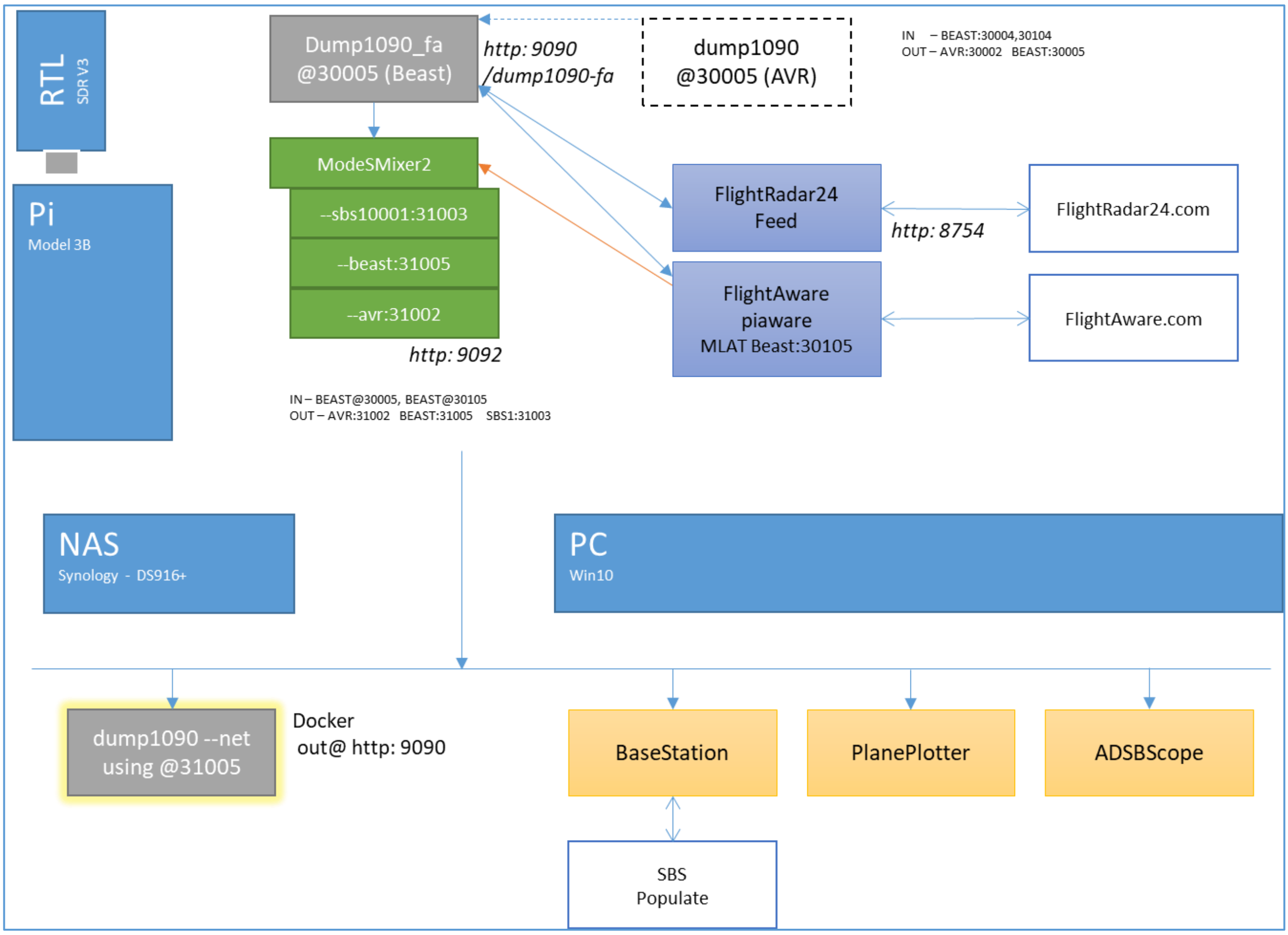
Contents

Intro

To build and run a Raspberry Pi as ADS-B receiver station is rather easy once you figured out all the bits and pieces. However there is no comprehensive write-up available (at least I did not found one).

The final setup should look like this. Whether or not the various Radar Programs are used is not decided however the base architecture should allow to do so. Hence we provide feeds of all kind.

The receiver is a single RTL-SDR V3 dongle for the moment but using the mixer allows to integrate more sources without re-designing the base architecture.



One experiment is to find out if the Synology NAS allows to seamless provide a web service using a Docker version of dump1090 (Ted Sluis).

The base feed is setup on a Raspi V3 – which may be changed to the 3 B+ in order to have more than 600mA available on the USB ports. (Need to enable this in the boot image)

The base image for the Raspi is provided and available on the FlightRadar24 website. Other SW used and sources are listed below.

For the initial phase all http servers of the components are enabled mainly to debug and validate.

Connectable feeds for outside are based on port 3100X in order to avoid mismatch with the dump program.

NOTE: copy and paste of the code bits below may cause issues with the dash (-) which you need to replace in the Pi console with the real dash character (WinWord sometimes thinks it has nicer dashes but they only look like)

Raspberry Pi ADS-B Multi Feeder Station

Multifeeder Station

Ref: https://forum.flightradar24.com/threads/11137-How-to-Feed-Data-to-Multiple-Sites-A-Brief-Guide

In order to make this mostly seamless and working…

I start with the FR24 Pi image, then using the dump1090-fa and serve the feeders via modeSmixer2

From ref above. OPTION-2:

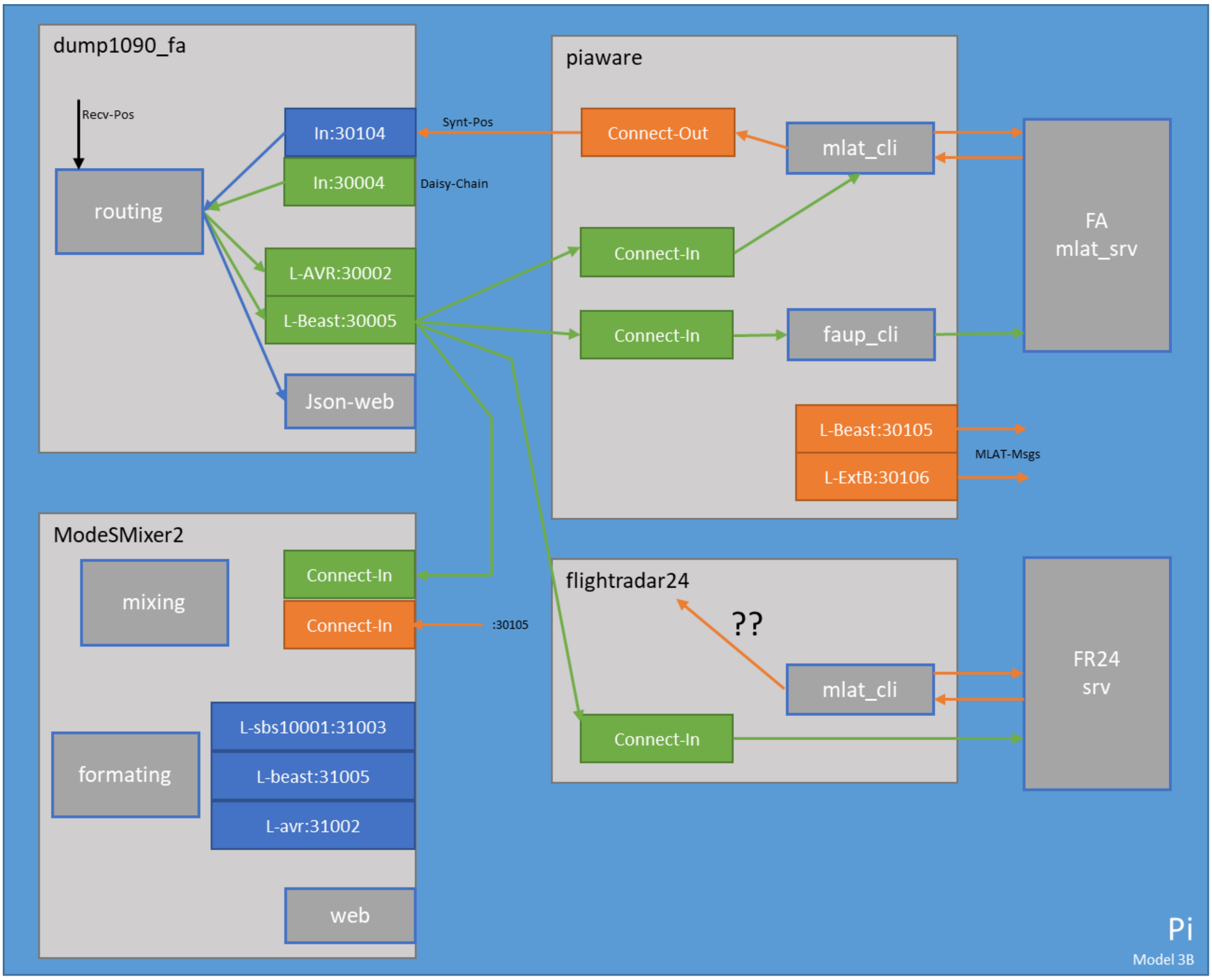
FR24 Feeder:
If an external decoder (like dump1090 MR, dump1090-mutability, dump1090-fa, or ModeSDeco2) is installed, then select:

"Receiver: ModeS Beast (TCP)" and "HOST/IP: 127.0.0.1:30005"

OR

"Receiver: AVR(TCP)" and "HOST/IP: 127.0.0.1:30002".

Let's start here and setup the RasPi …



Get the current Pi image from FlightRadar24.com (just my path, others will work as well)

Create a bootable uSD card and run it the first time in order to connect with SSH.

Additional files to be added can be put into the boot section and removed later (assuming you prepare with a Win10 desktop and mount the SD card) or using WinSCP to transfer files seamlessly from Win10 to your Pi.

While FR24 would normally start the delivered instance of dump1090 this behavior is disabled when setting the FR24 feeder to receive from an alternate source (which is set in the fr24feeder config file below)

Installing PIAware support will also create the needed startup files to use dump1090-fa without having two instances competing with each other (and creating a mess..)

We also need to add the modeSmixer2 add-on to be started and added to boot (see below).

As this setup uses an LNA sourced by the cable (Bias-T) we need a driver that supports this and to enable it while init. See Appendix: **RTL-SDR Driver and Bias-T**

We have FR24 items from the image

Use the raspi config to set preferences

```
        $ sudo raspi-config
```

Set your password, hostname, locales and other feeder independent stuff

piaware install

Ref: https://flightaware.com/adsb/piaware/install

This will download and install the PiAware repository package, which tells your Pi's package manager (apt) how to find FlightAware's software packages in addition to the packages provided by Raspbian.

```
        $ wget http://de.flightaware.com/adsb/piaware/files/packages/pool/piaware/p/piaware-
        support/piaware-repository_3.6.2_all.deb
        $ sudo dpkg -i piaware-repository_3.6.2_all.deb
```

This will download and install PiAware and required dependencies on your Raspberry Pi.

```
        $ sudo apt-get update
        $ sudo apt-get install piaware
```

Download and Install dump1090 - If you don't already have ADS-B receiver software such as dump1090 installed, then you can install FlightAware's version of dump1090 by executing the following command.

```
        $ sudo apt-get install dump1090-fa
        $ sudo piaware-config
```

modeSmixer2

Ref and thanks: http://sonicgoose.com

Add modesmixer2 supplied as tar file through the boot partition
Link to binaries – http://xdeco.org/?page_id=48

Get it into the pi home dir, copy to location and adjust owner and access

```
        $ tar –x –f modesmixer2_rpi2-3_deb9_20180616.tgz   (or what your name was)
        $ sudo cp modesmixer2 /usr/bin
        $ sudo chown root:root /usr/bin/modesmixer2
        # check
        $ ls –l /usr/bin/modesmixer2

        -rwxr-xr-x 1 root root 3185480 Aug 16 19:50 /usr/bin/modesmixer2
```

You may remove the extra files from tar –x now

Add sqlite3

```
        $ sudo apt-get install sqlite3
```

Create documents folders in our ‘pi’ account for database and graphics the mixer can show

```
        $ mkdir documents
        $ mkdir documents/OperatorFlags
        $ mkdir documents/Outlines
```

get populated db files into the documents folder – see below where and what (think about updates)

Ref: *https://data.flightairmap.com/*
https://radarspotting.com/forum/index.php?topic=3182.120

Ref: *https://forum.virtualradarserver.co.uk/viewtopic.php?t=236*
You can download a BaseStation.sqb from here: *http://www.virtualradarserver.co.uk/Fil ... tation.zip*

Getting a populated flightpath.sqb was not possible without registering with the PlanePlotter Yahoo group, decide on your own. I was creating a flightpath.sqb from an available route file. See appendix for more information.

Settings will be in the startupfile:

```
$ touch modesmixer2.sh
$ nano modesmixer2.sh
```

Type/Copy Paste

```
#!/bin/sh -
# connect to dump1090  BEAST output and piaware MLAT results
# provide data for SBS1, BEAST, AVR on 3100x ports
# create a http server on port 9092
# use some db files manually provided in the pi home dir

/usr/bin/modesmixer2 --inConnect 127.0.0.1:30005 --inConnect 127.0.0.1:30105 \
--outServer beast:31005 --outServer sbs10001:31003 \
--outServer avr:31002 \
--web 9092 \
--location 47.:8. \
--dbro /home/pi/documents/basestation.sqb \
--frdb /home/pi/documents/flightroute.sqb \
--silhouettes /home/pi/documents/Outlines
```

Ctrl-X to exit and save

Move and set exec rights

```
$ sudo cp modesmixer2.sh /usr/bin
$ sudo chown root:root /usr/bin/modesmixer2.sh
$ sudo chmod +x /usr/bin/modesmixer2.sh
# check
$ ls -l /usr/bin/modesmixer2.sh

-rwxr-xr-x 1 root root 499 Aug 16 19:50 /usr/bin/modesmixer2.sh
```

Add to startup – we create a service and link the stuff

```
$ touch modesmixer2.service
$ nano modesmixer2.service
```

Type/Copy Paste

```
# modesmixer2 service for systemd
# install in /lib/systemd/system and link to default.target

[Unit]
Description=ModeSMixer2
Documentation=http://xdeco.org
Wants=network-online.target
After=dump1090-fa.service network-online.target time-sync.target

[Service]
User=piaware
RuntimeDirectory=piaware
ExecStart=/usr/bin/modesmixer2.sh
ExecReload=/bin/kill -HUP $MAINPID
Type=simple
Restart=on-failure
RestartSec=30

[Install]
WantedBy=default.target
```

Ctrl-X to exit and save

```
$ sudo cp modesmixer2.service /lib/systemd/system
$ sudo chown root:root /lib/systemd/system/modesmixer2.service
# check
$ sudo ls –l /lib/systemd/system/modesmixer2.service

-rw-r--r-- 1 root root 449 Aug 17 00:26 /lib/systemd/system/modesmixer2.service

$ sudo ln -s  /lib/systemd/system/modesmixer2.service \
/etc/systemd/system/default.target.wants

# check
$ sudo ls –l /etc/systemd/system/default.target.wants/modesmixer2.service

lrwxrwxrwx 1 root root 39 Aug 17 00:30
/etc/systemd/system/default.target.wants/modesmixer2.service ->
/lib/systemd/system/modesmixer2.service
```

dump1090 Init

Ref: see appendix

Setup the Pi local instance to provide the needed feeds.

Also I found for the RTL-SDR V3 dongle with the matching LNA that the AutoGain set by default (--gain -10) causes the receiver to congest and no longer decode nearby aircraft signals properly. Setting the gain to 24 (or 29) manually seems more reasonable if you have an antenna with some gain. A plain λ/4 pole may work with the autogain. You will see a decrease in the very long range but this is a tradeoff you want to decide on your own – either near range is OK or as long a range as possible.
Also the setting in the piaware init did not cause the gain setting for the dump1090-fa to change; hence I changed it here.

We start with the default provided through piaware installation and modify/extend:

```
$ sudo nano /etc/default/dump1090-fa     when using the fa dump version
```

Type/Copy Paste/Edit

```
# dump1090-fa configuration
# This is read by the systemd service file as an environment file,
# and evaluated by some scripts as a POSIX shell fragment.

# If you are using a PiAware sdcard image, this config file is regenerated
# on boot based on the contents of piaware-config.txt; any changes made to this
# file will be lost.


RECEIVER_OPTIONS="--device-index 0 --gain 24 --ppm 0 --net-bo-port 30005"
DECODER_OPTIONS="--max-range 360"
NET_OPTIONS="--net --net-heartbeat 60 --net-ro-size 1000 --net-ro-interval 1 --net-ri-
port 0 --net-ro-port 30002 --net-sbs-port 30003 --net-bi-port 30004,30104 --net-bo-port
30005"
JSON_OPTIONS="--json-location-accuracy 1"
```

Ctrl-X to exit and save

```
$ sudo nano /etc/default/dump1090-mutability  only when using the FR24 version
```

Type/Copy Paste/Edit

```
LAT="la.nnnnn"
LON="lo.nnnnn"
# Port to listen on for FATSV-format output connections. 0 disables.
FATSV_OUTPUT_PORT="0"
# Additional options that are passed to the Daemon.
EXTRA_ARGS="="--net --net-http-port 9090"
```

Ctrl-X to exit and save

This is to set the location until it is updated through a registered account in the config web page

```
$ sudo nano /var/cache/piaware/location.env
```

Add a line that is evaluated by dump1090-fa startup

```
#
PIAWARE_DUMP1090_LOCATION_OPTIONS=”--lat 47.mmmmm --lon 8.nnnnn”
```

Ctrl-X to exit and save

Note: this will be replaced by the feedback through piaware once changing it there with

```
PIAWARE_LAT="47."
PIAWARE_LON="8."
PIAWARE_DUMP1090_LOCATION_OPTIONS="--lat 47. --lon 8."
```

RTL-SDR BiasT modification

See Appendix (This applies for the RTL-SDR V3 dongle using a BiasT feed LNA), add the red line after building the utility (see appendix) -b 0 would disable BiasT, this is to be run before the dongle is used and locked by the dump1090 process – as we use the fa version it is added to the appropriate startup

```
$ sudo nano /etc/systemd/system/default.target.wants/dump1090-fa.service
```

dump1090-fa startup: `/etc/systemd/system/default.target.wants/dump1090-fa.service`

```
# dump1090-fa service for systemd

[Unit]
Description=dump1090 ADS-B receiver (FlightAware customization)
Documentation=https://flightaware.com/adsb/piaware/
Wants=network.target
After=network.target

[Service]
EnvironmentFile=/etc/default/dump1090-fa
EnvironmentFile=-/var/cache/piaware/location.env
User=dump1090
RuntimeDirectory=dump1090-fa
RuntimeDirectoryMode=0755
ExecStartPre=/home/pi/rtl_biast/build/src/rtl_biast -b 1
ExecStart=/usr/bin/dump1090-fa \
 $RECEIVER_OPTIONS $DECODER_OPTIONS $NET_OPTIONS $JSON_OPTIONS $PIAWARE_DUMP1090_LOCATION_OPTIONS
\
 --write-json /run/dump1090-fa --quiet
Type=simple
Restart=on-failure
RestartSec=30
Nice=-5

[Install]
WantedBy=default.target
```

Ctrl-X to exit and save

Note: this will fail to start if the red line fails (else use ExecStartPre=-/home… then it passes without msg)

FR24 Init

Ref: FR24 manual (as of date of writing)

Editing the fr24feeder config file

Add the key supplied when registering the feeder.
Then use MLAT calc and choose the option for the independent feed from the dump1090-fa decoder.
Disable outputs to not conflict with other outputs on the same ports.

```
$ sudo nano /etc/fr24feed.ini
```

Type/Copy Paste/Edit

```
fr24key=""   (insert yours)
mlat="yes"
mlat-without-gps="yes"
logmode="2"
# The output of the modeSMixer2 instance is used
receiver="beast-tcp"
host="127.0.0.1:30005"
# disable FR24 outputs
bs="no"
raw="no"
```

Ctrl-X to exit and save

fr24feeder startup

Change the startup to follow the dump1090-fa and modesmixer2

```
$ sudo nano /etc/systemd/system/multi-user.target.wants/fr24feed.service
```

Update the After line and disable the FR24 dump1090 start (even if it would bypass itself)

```
[Unit]
Description=Flightradar24 Decoder & Feeder
After=dump1090-fa.service modesmixer2.service network-online.target time-sync.target

[Service]
Type=simple
Restart=always
LimitCORE=infinity
# ExecStartPre=-/usr/lib/fr24/install_dump1090.sh
ExecStartPre=-/usr/lib/fr24/unregister_kernel_modules.sh
ExecStartPre=-/usr/lib/fr24/create_missing_directories.sh
ExecStart=/usr/bin/fr24feed
User=fr24
Group=fr24
PermissionsStartOnly=true
StandardOutput=null

[Install]
WantedBy=multi-user.target
```

Ctrl-X to exit and save

piaware Init

Editing the piaware config file

```
$ sudo nano /etc/piaware.conf
```

modify/extend:

```
# Should piaware use the wired ethernet port for network access?
wired-network yes
# Should piaware use a wifi adaptor for network access?
wireless-network no
# Should automatic system updates (initiated by FlightAware) be allowed?
allow-auto-updates no
# Should manual system updates (initiated by you from the website control
# panel) be allowed?
allow-manual-updates yes
# Should PiAware enable multilateration where possible?
# You may need to disable this if multilateration overloads your receiver.
allow-mlat yes
# Multilateration results are returned to PiAware from FlightAware
# Default feedback:
# beast,connect,localhost:30104
# this feeds multilateration results back to the local dump1090 for display on the map
# beast,listen,30105 - this provides a Beast binary format feed of multilateration
results that can be used by external systems, e.g.,
mlat-results yes
# Should PiAware enable reception of Mode A/C messages when requested?
# You may need to disable this if processing Mode A/C overloads your receiver.
allow-modeac no
# Additional settings can be added below.
# a unique identifier /var/log/piaware.log on your existing install.
feeder-id xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
# Sets rfkill ("airplane mode") on the Pi 3
rfkill no
# configures how PiAware attempts to talk to the ADS-B receiver (see below)
# this also kills PiAware feeds (according to spec)
receiver-type other
receiver-host 127.0.0.1
receiver-port 30005
#
rtlsdr-gain 24
```

Ctrl-X to exit and save

# check

```
$ sudo piaware-config
```

```
allow-auto-updates   no                                   # value set at /etc/piaware.conf:13
allow-manual-updates yes                                  # value set at /etc/piaware.conf:15
allow-mlat           yes                                  # value set at /etc/piaware.conf:18
allow-modeac         no                                   # value set at /etc/piaware.conf:27
feeder-id            xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx # value set at /etc/piaware.conf:30
mlat-results         yes                                  # value set at /etc/piaware.conf:24
receiver-host        127.0.0.1                            # value set at /etc/piaware.conf:36
receiver-port        30005                                # value set at /etc/piaware.conf:37
receiver-type        other                                # value set at /etc/piaware.conf:35
rfkill               no                                   # value set at /etc/piaware.conf:32
wired-network        yes                                  # value set at /etc/piaware.conf:9
wireless-network     no                                   # value set at /etc/piaware.conf:11
```

Piaware startup:

Change the startup to follow the modesmixer2

```
$ sudo nano /etc/systemd/system/default.target.wants/piaware.service
```

Update the After line

```
# piaware uploader service for systemd
# install in /etc/systemd/system

[Unit]
Description=FlightAware ADS-B uploader
Documentation=https://flightaware.com/adsb/piaware/
Wants=network-online.target
After=dump1090-fa.service modesmixer2.service network-online.target time-sync.target

[Service]
User=piaware
RuntimeDirectory=piaware
ExecStart=/usr/bin/piaware -p %t/piaware/piaware.pid -plainlog -statusfile %t/piaware/status.json
ExecReload=/bin/kill -HUP $MAINPID
Type=simple
Restart=on-failure
RestartSec=30
# exit code 4 means login failed
# exit code 6 means startup failed (bad args or missing MAC)
RestartPreventExitStatus=4 6

[Install]
WantedBy=default.target
```

Ctrl-X to exit and save

Set the piaware (dump1090-fa) webserver to port 9090

Change the port in the lighthttp script for dump.

```
$ sudo nano /etc/lighttpd/conf-enabled/89-dump1090-fa.conf
```

Type/Copy Paste/Edit

```
# Allows access to the static files that provide the dump1090 map view,
# and also to the dynamically-generated json parts that contain aircraft
# data and are periodically written by the dump1090 daemon.

alias.url += (
  "/dump1090-fa/data/" => "/run/dump1090-fa/",
  "/dump1090-fa/" => "/usr/share/dump1090-fa/html/"
)

# redirect the slash-less URL
url.redirect += (
  "^/dump1090-fa$" => "/dump1090-fa/"
)

# Listen on port 8080 and serve the map there, too.
$SERVER["socket"] == ":9090" {
  alias.url += (
    "/data/" => "/run/dump1090-fa/",
    "/" => "/usr/share/dump1090-fa/html/"
  )
}

# Add CORS header
server.modules += ( "mod_setenv" )
$HTTP["url"] =~ "^/dump1090-fa/data/.*\.json$" {
  setenv.add-response-header = ( "Access-Control-Allow-Origin" => "*" )
```

Ctrl-X to exit and save

You may access the server via `http://yourPi[.local]/dump1090-fa` OR `http://yourPi[.local]:9090`

Blacklist another DVB-T user of the dongle

```
$ sudo nano /etc/modprobe.d/blacklist-rtl28xxu.conf
```

Type/Copy Paste/Edit

```
blacklist blacklist dvb_usb_rtl28xxu
```

Ctrl-X to exit and save

Fix a readonly mount problem (with the FR24 Image build I use)

Issued daily schedule to update the fr24feed_updater (in /etc/cron.d) mounts the filesystem as readonly – well everything fails pretty much afterwards…

```
$ sudo nano  /usr/lib/fr24/fr24feed_updater.sh
```

Update at the very end to mount as read-write rather than ro (read-only)

```
if [ "${NEEDS_REMOUNT}" = true ]; then
    mount -o remount,rw /
fi
```

Ctrl-X to exit and save

Add data traffic monitor

Ref: https://en.wikipedia.org/wiki/Iftop

Install iftop – similar to top for the processor it reports the actual traffic on the interface

```
$ sudo apt-get install iftop
```

Run iftop for the Wifi interface reporting Bytes/sec (or kBytes/sec)

```
$ sudo iftop -B -i wlan0   (or eth0)
```

Reboot daily

```
$ sudo crontab –e
```

First time you may need to choose your editor e.g. 2 nano

Type/Copy Paste/Edit

```
1 3 * * * /sbin/shutdown -r
```

Ctrl-X to exit and save

Writes a daily job to reboot at 03:01 (low air traffic time) into the scheduler (cron daemon)

Dump all changes

Use this script to dump all changes for documentation and review

```
$ touch multifeed.sh
$ nano multifeed.sh
```

Type/Copy Paste

```
# Report all files touched here
#
echo "DUMP of the multifeed extensions"
date

echo "*** HOSTNAME"
cat /etc/hostname

echo "All users defined on this system"
cat /etc/passwd

echo "+++++++ Multifeed modifications ++++++"
echo "*** ModeSMixer2 - Binary"
ls -l /usr/bin/modesmixer2

echo "*** BiasT addition - Binary"
ls -l /home/pi/rtl_biast/build/src/rtl_biast

echo "*** Dump1090-fa - Service"
cat /etc/systemd/system/default.target.wants/dump1090-fa.service

echo "*** Dump1090-fa - Config"
cat /etc/default/dump1090-fa

echo "*** ModeSMixer2 - Script"
ls -l /usr/bin/modesmixer2.sh
cat /usr/bin/modesmixer2.sh

echo "*** ModeSMixer2 - Service"
ls -l /lib/systemd/system/modesmixer2.service
cat /lib/systemd/system/modesmixer2.service
ls -l /etc/systemd/system/default.target.wants/modesmixer2.service

echo "*** FR24 Feeder - Service"
cat /etc/systemd/system/multi-user.target.wants/fr24feed.service

echo "*** FR24 Feeder - Config"
cat /etc/fr24feed.ini

echo "*** PiAware Feeder - Service"
cat /etc/systemd/system/default.target.wants/piaware.service

echo "*** PiAware Feeder - Config"
cat /etc/piaware.conf

echo "*** PiAware Web Service - Config"
cat /etc/lighttpd/conf-enabled/89-dump1090-fa.conf

echo "END OF DUMP of the multifeed extension"
```

Ctrl-X to exit and save

Make it exec

```
$ chmod +x multifeed.sh
```

Run it with sudo as some files are not accessibly by non root users

```
$ sudo ./multifeed.sh > mfReport.txt
$ more mfReport.txt
```

Reboot and hope it works…

You may check

```
/var/log/daemon.log
/var/log/piaware.log
/var/log/fr24feed/fr24feed.log
/var/log/syslog
/var/log/messages
```

Docker solution on Synology NAS

Todo…

Ref: https://github.com/mutability/dump1090

This is the recommended configuration; a dedicated webserver is almost always going to be better and more secure than the collection of hacks that is the dump1090 webserver. It works by having dump1090 write json files to a path under /run once a second (this is on tmpfs and will not write to the sdcard). Then an external webserver is used to serve both the static html/javascript files making up the map view, and the json files that provide the dynamic data.

This uses a configuration file installed by the package at /etc/lighttpd/conf-available/89-dump1090.conf. It makes the map view available at http:///dump1090/

This should also work fine with other webservers, you will need to write a similar config to the lighttpd one (it's basically just a couple of aliases). If you do set up a config for something else, please send me a copy so I can integrate it into the package!

Appendix A

FR24 Image - Raspberry Pi reference 2017-11-29

Base build: Raspbian Jessie Lite

```
        Username: pi
        Password: raspberry

        /etc/hostname
        raspberrypi
```

General startup and files

FR24 is started via: `/etc/systemd/system/multi-user.target.wants/fr24feed.service`

PIA is started via:

```
 /etc/systemd/system/default.target.wants/dump1090-fa.service
-> /lib/systemd/system/dump1090-fa.service

 /etc/systemd/system/default.target.wants/lib/systemd/system/piaware.service
```

Default configs

```
/etc/fr24feed.ini
        bs=yes
        raw=yes
        mlat="yes"
        mlat-without-gps="yes"
        logmode="2"
        receiver="dvbt"

/etc/default/dump1090-mutability
        # dump1090-mutability configuration file
        # "dpkg-reconfigure dump1090-mutability"
        # Set to "yes" to start dump1090 on boot.
        START_DUMP1090="no"
        # User to run dump1090 as.
        DUMP1090_USER="dump1090"
        # Logfile to log to
        LOGFILE="/var/log/dump1090-mutability.log"

        # ***Receiver options
        # RTLSDR device index or serial number to use
        # If set to "none", dump1090 will be started in --net-only mode
        DEVICE=""
        # RTLSDR gain in dB.
        # If set to "max" (the default) the maximum supported gain is used.
        # If set to "agc", the tuner AGC is used to set the gain.
        GAIN="max"
        # RTLSDR frequency correction in PPM
        PPM="0"
        # If yes, enable sampling at 2.4MHz. Otherwise, 2.0MHz is used.
        OVERSAMPLE="yes"
        # If yes, enables phase-enhancement of messages
        PHASE_ENHANCE="yes"

        # ***Decoding options
        # If yes, fixes messages with correctable CRC errors.
        FIX_CRC="yes"
        # If yes, enables aggressive fixes to damaged messages.
        # Use with caution - it can increase the rate of undetected errors.
        AGGRESSIVE="no"
        # If set, supplies a reference location for local position decoding.
        LAT=""
        LON=""
        # If set, provides the absolute maximum receiver range used to
        # filter bad position reports, and to determine when local position
        # decoding is safe to use. Specify this in nautical miles (NM).
        MAX_RANGE="300"

        # ***Networking options
        # Port to listen on for HTTP connections. 0 disables.
        # HTTP defaults to being disabled unless you specify something here. I
        # that you do not enable this, and instead serve the contents of
        # /usr/share/dump1090-mutability and JSON_DIR (below) using a proper
        # webserver. See /etc/lighttpd/conf-available/90-dump1090.conf
        # for an example configuration ("sudo lighty-enable-mod dump1090" to enable)
        HTTP_PORT="0"
        # Port to listen on for raw (AVR-format) input connections. 0 disables.
```

```
RAW_INPUT_PORT="30001"
# Port to listen on for raw (AVR-format) output connections. 0 disables.
RAW_OUTPUT_PORT="30002"
# Port to listen on for SBS-format output connections. 0 disables.
SBS_OUTPUT_PORT="30003"
# Port to listen on for Beast-format input connections. 0 disables.
BEAST_INPUT_PORT="30004"
# Port to listen on for Beast-format output connections. 0 disables.
BEAST_OUTPUT_PORT="30005"
# Port to listen on for FATSV-format output connections. 0 disables.
FATSV_OUTPUT_PORT="10001"
# TCP heartbeat interval in seconds. 0 disables.
NET_HEARTBEAT="60"
# Minimum output buffer size per write, in bytes.
NET_OUTPUT_SIZE="500"
# Maximum buffering time before writing, in seconds.
NET_OUTPUT_INTERVAL="1"
# TCP buffer size, in bytes
NET_BUFFER="262144"
# Bind ports on a particular address. If unset, binds to all interfaces.
# This defaults to binding to localhost. If you need to allow remote
# connections, change this.
NET_BIND_ADDRESS="127.0.0.1"

# ***Misc options
# Interval (in seconds) between logging stats to the logfile. 0 disables.
STATS_INTERVAL="3600"
# Path to write json state to (for use with an external webserver). Blank disables.
JSON_DIR="/run/dump1090-mutability"
# Interval between writing json state (in seconds). 0 disables.
JSON_INTERVAL="1"
# Accuracy of receiver location to write to json state, one of "exact" / "approximate" / "none"
JSON_LOCATION_ACCURACY="approximate"
# Set to yes to log all decoded messages
# This can get large fast!
LOG_DECODED_MESSAGES="no"
# Additional options that are passed to the Daemon.
EXTRA_ARGS=""
```

FR24 Feed parameters

/etc/fr24feed.ini – use `--Option` from the command line

`fr24key=0123456789ABCDEF`

- Your 16 character long sharing key that can be used only on one machine simultaneously. If you have multiple receivers please requests more sharing keys as otherwise it might get blocked due to one session kicking the other out.

`bs=yes* | no`

- Defaults to "yes" and decides whether or not the Basestation compatible data stream should be generated on port 30003. When using dump1090 with "--net" please disable as that feed is already provided and otherwise it will cause conflicts.

`raw=yes | no*`

- Defaults to "no" and decides whether or not the RAW data stream should be available on port 30334 for all receivers but DVBT which uses port 30002 instead. When using dump1090 with "--net" please disable as that feed is already provided and otherwise it will cause conflicts.

`mpx=yes | no*`

- Defaults to "no" and decides whether or not the SBS1/3 compatible data stream should be enabled on port 10001. This option is only available when Kinetic Avionics receiver is used to enable Basestation connection. For all other receivers this options is ignored.

`mlat="yes" | no`

- Not in the manual; may be only as ini file param. Toggles the MLAT calculation participation – it seems…

`mlat-without-gps="yes" | no`

- Not in the manual; may be only as ini file param. Toggles the mode and feed of the MLAT calculation participation – it seems…

`logmode="2"`

- Changes log file mode "0" - no log file, "1" - rotate at midnight keep for 48 hours, "2" - rotate at midnight keep for 72 hours. Defaults to "0" that is no logfile stored.

`logpath="/path/to/logfile/directory"`

- On Linux and OSX this parameter lets you change the location of log files. Defaults to "/var/log" on Linux and "/var/tmp" on OSX. Windows users have their log file located in the application directory itself.

`receiver="dvbt" | sbs1tcp | sbs1usb | beast | beast-tcp | dvbt-mr`

- Defines the receiver FR24 Decoder should connect to. There is no default value as you need to know what receiver you have. Supported types are:

sbs1tcp - SBS1, SBS1er, SBS3 connected via Ethernet/TCP, please specify its IP and port by using "--host", see below. When using a Kinetic Avionics receiver an additional option "--mpx" is available so you can connect Basestation as well.
The default IP and port is 192.168.1.170:10001.
1. Configure Basestation to connect to TCP rather than USB using the following details:
Settings-> Hardware settings-> Network-> 127.0.0.1, port 20072
2. Edit the fr24feed.ini and add the following line:
mpx="yes"
3. Start FR24 software first so it connects to the SBS-1e over USB
4. Start Basestation that will connect to FR24 software via TCP
5. Make sure to have DATA SOURCE changed to Use Network.

sbs1usb - SBS1, SBS1er, SBS3 connected directly via USB.
For Linux and MacOSX users the device path of the receiver needs to be specified using "--path" (see below). Windows users may omit this parameter when only one receiver is connected as it will be detected automatically. When using a Kinetic Avionics receiver an additional option "--mpx" is available so you can connect Basestation as well.
beast - Mode-S Beast connected directly via USB.
For Linux and MacOSX the device path of the receiver needs to be specified using "--path" (see below). Windows users should provide serial number or leave it empty when only one Mode-S Beast is connected to the system. Only the binary output mode at 3Mbit is currently supported.
beast-tcp - Mode-S Beast connected via TCP (serial to network, etc).
Please specify its IP and port by using "--host", see below.
dvbt - DVBT USB dongle connected directly via USB.
Only one DVBT dongle can be used at a time. When using with stand alone dump1090 instance or another software defined radio demodulator please use the "avr-tcp" receiver type instead.
dvbt-mr - Same as above, but using Malcolm Robb's dump1090 version that has some additional features like built in HTTP interface etc.
avr-tcp – Only noted above – connects to RAW out of the source (--host)

`procargs="flags and arguments"`

- Extra arguments that should be passed to dump1090 application on startup. To enable HTTP server/web interface of Malcolm Robb's dump1090 please set to "--net --net-http-port 8888" and make sure both "bs" and "raw" feeds are switched off to avoid conflicts.

`windowmode="MODE"`

- Decides whether the console window should be visible "0" or hidden "1" on startup. This parameter is only used in the MS Windows build. Defaults to "0" that is show console window.

`bind-interface="IP_ADDRESS"`

- Changes the network interface that all FR24 Feeder services listen on. By default it listens on all available interfaces (0.0.0.0). This affects BS/RAW/MPX feeds as well as the WWW interface.

gt=TIMEOUT

- Defines global timeout, in seconds, for receiver data/mode-s frames. When no frame is received within that time it will reconnect to the receiver or restart the process in case of DVBT stick used directly. Defaults to 600 seconds.

--config-file=”/path/to/fr24feed.ini”

- Command line option:

Lets you override the default configuration file location. Useful only for standalone version users, lets you have multiple configs for different receivers etc.

piaware Configuration

Ref: https://flightaware.com/adsb/piaware/about

Ref: https://de.flightaware.com/adsb/piaware/advanced_configuration

- Use the "piaware-config" utility to view and change settings.
  Run "`piaware-config <setting> <value>`" to make changes.
  Run "`piaware-config`" to show the current configuration settings.
- Edit `/boot/piaware-config.txt` to make configuration changes, using an editor of your choice (e.g. "`sudo nano /boot/piaware-config.txt`").
- For package-based installs, use `piaware-config` or edit `/etc/piaware.conf`.

Status of piaware:

```
$ sudo piaware-status
```

You can restart PiAware by typing:

```
$ sudo systemctl restart piaware
```

dump1090-fa startup: `/etc/systemd/system/default.target.wants/dump1090-fa.service`

```
# dump1090-fa service for systemd

[Unit]
Description=dump1090 ADS-B receiver (FlightAware customization)
Documentation=https://flightaware.com/adsb/piaware/
Wants=network.target
After=network.target

[Service]
EnvironmentFile=/etc/default/dump1090-fa
EnvironmentFile=-/var/cache/piaware/location.env
User=dump1090
RuntimeDirectory=dump1090-fa
RuntimeDirectoryMode=0755
ExecStart=/usr/bin/dump1090-fa \
 $RECEIVER_OPTIONS $DECODER_OPTIONS $NET_OPTIONS $JSON_OPTIONS $PIAWARE_DUMP1090_LOCATION_OPTIONS
\
 --write-json /run/dump1090-fa --quiet
Type=simple
Restart=on-failure
RestartSec=30
Nice=-5

[Install]
WantedBy=default.target
```

Piaware startup: `/etc/systemd/system/default.target.wants/lib/systemd/system/piaware.service`

```
# piaware uploader service for systemd
# install in /etc/systemd/system

[Unit]
Description=FlightAware ADS-B uploader
Documentation=https://flightaware.com/adsb/piaware/
Wants=network-online.target
After=dump1090-fa.service network-online.target time-sync.target

[Service]
User=piaware
RuntimeDirectory=piaware
ExecStart=/usr/bin/piaware -p %t/piaware/piaware.pid -plainlog -statusfile %t/piaware/status.json
ExecReload=/bin/kill -HUP $MAINPID
Type=simple
Restart=on-failure
RestartSec=30
# exit code 4 means login failed
# exit code 6 means startup failed (bad args or missing MAC)
RestartPreventExitStatus=4 6

[Install]
WantedBy=default.target
```

/etc/piaware.conf - Default config file:

```
# This file configures the Piaware sdcard image configuration. Whenever the system
boots, this file is read and its contents are used to configure the system.
# For simplicity, this file only shows the most common options.
# For a complete list of options, see
# https://flightaware.com/adsb/piaware/advanced_configuration

# ***NETWORK CONFIGURATION
# Should piaware use the wired ethernet port for network access?
wired-network yes

# Should piaware use a wifi adaptor for network access?
wireless-network yes
# Wifi SSID and password.
wireless-ssid MyWifiNetwork
wireless-password s3cr3t99

# *** RECEIVER CONFIGURATION
# For a complete list of receiver types and their associated settings,
# see the Advanced Configuration page linked at the top of this file.
# For a receiver type of 'rtlsdr', this setting controls the dongle gain.
# -10 means AGC / maximum gain; other values mean a gain value in dB.
# RTL-SDR gains -1.0 1.5 4.0 6.5 9.0 11.5 14.0 16.5 19.0 21.5 24.0 29.0 34.0 42.0
rtlsdr-gain -10

# ***OTHER CONFIGURATION
# Should automatic system updates (initiated by FlightAware) be allowed?
allow-auto-updates yes
# Should manual system updates (initiated by you from the website control
# panel) be allowed?
allow-manual-updates yes
# Should PiAware enable multilateration where possible?
# You may need to disable this if multilateration overloads your receiver.
allow-mlat yes
# Should PiAware enable reception of Mode A/C messages when requested?
# You may need to disable this if processing Mode A/C overloads your receiver.
allow-modeac yes

# Additional settings can be added below.
mlat-results yes* or no
# if "yes", multilateration results are returned to PiAware from FlightAware
feeder-id 12345678-1234-1234-1234-123456789abc
# a unique identifier (for example: 12345678-1234-1234-1234-123456789abc)configures the
unique identifier of a site; this should only be set if attempting to preserve an
existing site after a hardware upgrade or reflashing the SD card image; the existing
feeder-id is labeled "Site Identifier" on the My ADS-B page or may be found in the
PiAware logs in /var/log/piaware.log on your existing install.
rfkill        yes or no*
#if "yes", sets rfkill ("airplane mode") on the Pi 3 to reduce interference if
wifi/bluetooth are not required
receiver-type       rtlsdr, beast, radarcape, relay, other rtlsdr
# configures how PiAware attempts to talk to the ADS-B receiver (see below)
```

Receiver Types

RTL-SDR

This is the traditional setup with a RTL-SDR dongle directly connected to the Pi. PiAware will connect to localhost:30005 for ADS-B data. Set "receiver-type" to "**rtlsdr**" and (on SD card image installs only) configure the other rtlsdr-* values if needed.

Beast (SD card image installs only)

This configures PiAware to talk to a Mode S Beast connected by USB directly to the Pi. Set "receiver-type" to "beast". No further configuration is needed.

Radarcape (SD card image installs only)

This configures PiAware to talk to a separate Radarcape over the network. Set "receiver-type" to "radarcape" and "radarcape-host" to the hostname or IP address of the Radarcape.

External receiver - Relayed connection (SD card image installs only)

This configures PiAware to talk to an external receiver or other ADS-B source over the network. The receiver needs to provide data in the Beast binary format over TCP. Set "receiver-type" to "relay" and "receiver-host" / "receiver-port" to the host/port to connect to. PiAware will establish a single TCP connection to the receiver and internally relay data to the local map display, faup1090 and mlat-client as needed.

External receiver - Direct connection
This configures PiAware to talk to an external receiver or other ADS-B source over the network. The receiver needs to provide data in the Beast binary format over TCP. Set "receiver-type" to "other" and "receiver-host" / "receiver-port" to the host/port to connect to.
PiAware will establish multiple TCP connections to the receiver. It will not forward data to the local map display or the standard ports. It does nothing extra beyond consuming the data from an existing source and forwarding it to FlightAware. This option requires more network bandwidth than the "relayed connection" mode and doesn't arrange for the local map to be fed, but it will work on package installs. It can also be used to connect to a receiver running locally on the Pi that's not covered by the other options.

Multilateration Settings
The "mlat-results-format" configuration setting controls how multilateration results are provided. It is a space-separated list, where each item is something that can be passed to the "--results" option of mlat-client:
<protocol>,connect,host:port - makes an outgoing connection to host:port
<protocol>,listen,port - listens for incoming connections on the given port

Supported protocols are:
basestation: the "port 30003" BaseStation format
ext_basestation: an extended version of the BaseStation format that includes mlat-specific data
beast: results encoded as DF18 ADS-B messages, Beast binary format

The default settings set up three items:
beast,connect,localhost:30104 - this feeds multilateration results back to the local dump1090 for display on the map
beast,listen,30105 - this provides a Beast binary format feed of multilateration results that can be used by external systems, e.g., VRS
ext_basestation,listen,30106 - like 30105, but in a different format
It should not be necessary to change the multilateration settings in most cases.

/etc/default/dump1090-fa - Default config file:

```
# dump1090-fa configuration
# This is read by the systemd service file as an environment file,
# and evaluated by some scripts as a POSIX shell fragment.

# If you are using a PiAware sdcard image, this config file is regenerated
# on boot based on the contents of piaware-config.txt; any changes made to this
# file will be lost.


RECEIVER_OPTIONS="--device-index 0 --gain -10 --ppm 0 --net-bo-port 30005"
DECODER_OPTIONS="--max-range 360"
NET_OPTIONS="--net --net-heartbeat 60 --net-ro-size 1000 --net-ro-interval 1 --net-ri-port 0 --
net-ro-port 30002 --net-sbs-port 30003 --net-bi-port 30004,30104 --net-bo-port 30005"
JSON_OPTIONS="--json-location-accuracy 1"
```

Dump1090

Ref: MalcolmRobb fork - https://github.com/MalcolmRobb/dump1090 (dump1090_mr)

Ref: Ted Sluis - fork: https://github.com/tedsluis/dump1090

Ref: Oliver Jowett - fork: https://github.com/mutability/dump1090

Command line of dump1090 ModeS Receiver

```
--device-index <index>   Select RTL device (default: 0)
--gain <db>              Set gain (default: max gain. Use -10 for auto-gain)
--enable-agc             Enable the Automatic Gain Control (default: off)
--freq <hz>              Set frequency (default: 1090 Mhz)
--ifile <filename>       Read data from file (use '-' for stdin)
--interactive            Interactive mode refreshing data on screen
--interactive-rows <num> Max number of rows in interactive mode (default: 15)
--interactive-ttl <sec>  Remove from list if idle for <sec> (default: 60)
--interactive-rtl1090    Display flight table in RTL1090 format
--raw                    Show only messages hex values
--net                    Enable networking
--modeac                 Enable decoding of SSR Modes 3/A & 3/C
--net-beast              TCP raw output in Beast binary format
--net-only               Enable just networking, no RTL device or file used
--net-bind-address <ip>  IP address to bind to (default: Any; Use 127.0.0.1 for
private)
--net-http-port <port>   HTTP server port (default: 8080)
--net-ri-port <port>     TCP raw input listen port  (default: 30001)
--net-ro-port <port>     TCP raw output listen port (default: 30002)
--net-sbs-port <port>    TCP BaseStation output listen port (default: 30003)
--net-bi-port <port>     TCP Beast input listen port  (default: 30004)
--net-bo-port <port>     TCP Beast output listen port (default: 30005)
--net-ro-size <size>     TCP raw output minimum size (default: 0)
--net-ro-rate <rate>     TCP raw output memory flush rate (default: 0)
--net-heartbeat <rate>   TCP heartbeat rate in seconds (default: 60 sec; 0 to disable)
--net-buffer <n>         TCP buffer size 64Kb * (2^n) (default: n=0, 64Kb)
--lat <latitude>         Reference/receiver latitude for surface posn (opt)
--lon <longitude>        Reference/receiver longitude for surface posn (opt)
--fix                    Enable single-bits error correction using CRC
--no-fix                 Disable single-bits error correction using CRC
--no-crc-check           Disable messages with broken CRC (discouraged)
--phase-enhance          Enable phase enhancement
--aggressive             More CPU for more messages (two bits fixes, ...)
--mlat                   display raw messages in Beast ascii mode
--stats                  With --ifile print stats at exit. No other output
--stats-every <seconds>  Show and reset stats every <seconds> seconds
--onlyaddr               Show only ICAO addresses (testing purposes)
--metric                 Use metric units (meters, km/h, ...)
--snip <level>           Strip IQ file removing samples < level
--debug <flags>          Debug mode (verbose), see README for details
--quiet                  Disable output to stdout. Use for daemon applications
--ppm <error>            Set receiver error in parts per million (default 0)
--help                   Show this help

Debug mode flags: d = Log frames decoded with errors
                  D = Log frames decoded with zero errors
                  c = Log frames with bad CRC
                  C = Log frames with good CRC
                  p = Log frames with bad preamble
                  n = Log network debugging info
                  j = Log frames to frames.js, loadable by debug.html
```

Connect external dump1090 and serve the local instance (e.g. to use with a `--net-only` instance)

Create an input server for a remote source and feed the local listener

```
$ nc remote-dump1090.example.net 30002 | nc localhost 30001
```

ModeSMixer2

Ref: http://xdeco.org/?page_id=48

```
modesmixer2 -h

Program options:
-h [ --help ] This help message
--inConnect arg Input connector.
Format: host:port
--inServer arg Input server.
Format: port
--inServerUdp arg Input udp server.
Format: port
--inSerial arg Input serial.
Format: port[:speed[:flow_control]]
FlowControl: none,software,hardware
--outConnect arg Output connector.
Format: type:host:port
Types: beast,avr,avrmlat,msg,sbs30006
--outConnectId arg Output connector with Id.
Format: host:port[:name:lat:lon:TEXT:freq]
--outConnectUdp arg Output udp connector.
Format: type:host:port
Types: beast,avr,avrmlat,msg,sbs30006
--outServer arg Output server.
Format: type:port
Types: beast,avr,avrmlat,msg,sbs30006,sbs10001,fatsv
--globes arg Enable Globe-S server.
Format: port:tablename:stationid
--web arg Enable Web Interface on port
--silhouettes arg Set Silhouettes directory
--pictures arg Set Aircraft Pictures directory
--db arg Set DB Filename
--dbro arg Set DB Filename with RO access only
--frdb arg Set FlightRoute DB Filename
--localtime Local Time in MSG format output (default: UTC)
--filter-expire arg Filter expire time, sec (default: 20)
--filter-count arg Filter min count (default: 6)
--filter-time arg Filter min time, sec (default: 60)
--filter-nocountry Disable ICAO Country filter (default: on)
--flight-expire-time arg Flight expire time, sec (default: 3600)
--log-noconsole Disable logging to console (default: on)
--log-file arg Enable logging to file (default: off)
--log-level arg Set logging level 0..5 (default: 4)
0 disabled, 1 error, 2 warning,
3 notice, 4 info, 5 debug
```

RTL-SDR Driver and Bias-T

Ref: https://www.rtl-sdr.com/getting-the-v3-bias-tee-to-activate-on-piaware-ads-b-images

Ref: https://forum.flightradar24.com/threads/11826-RTl_SDR-v3-bias-tee-power-on-off?p=104563&viewfull=1#post104563

This needs attention for dump1090_mr delivered with FR24

STEP-1:
Build package rtl_biast

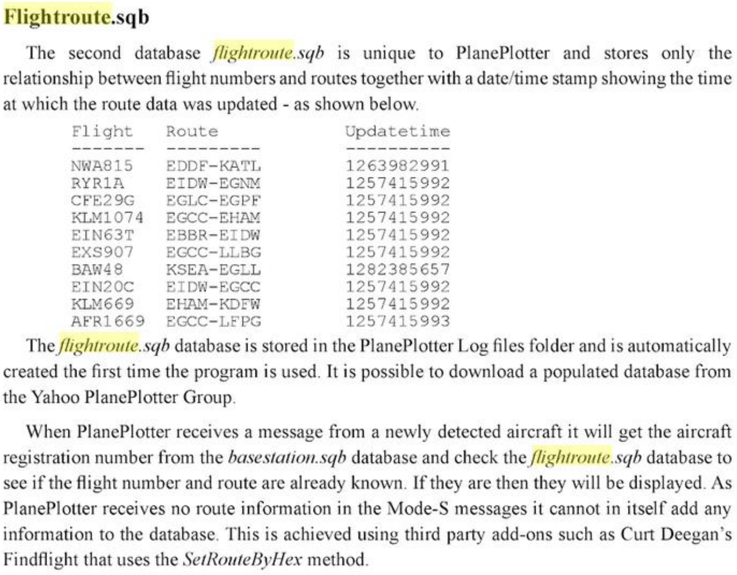
```
#First installed Packages needed to build and to fulfill dependencies
sudo apt-get install git
sudo apt-get install cmake
sudo apt-get install libusb-1.0

#Now build rtl_biast
git clone https://github.com/rtlsdrblog/rtl_biast
cd rtl_biast
mkdir build
cd build
cmake ..
make

#TEST-1
cd src
./rtl_biast -b 1

usb_claim_interface error -6

#TEST-2
sudo systemctl stop fr24feed
sudo systemctl stop dump1090-mutability

./rtl_biast -b 1
Found Rafael Micro R820T tuner
```

STEP-2:
Configure fr24feeder to switch-on bias-t of RTL-SDR V3, BEFORE starting fr24feed (which will start dump1090-mutability).
(a) Edit file fr24feed.service and ADD line shown in red, above the line ExecStart=/usr/bin/fr24feed
(b) Reboot Pi.
(c) After reboot, DC Voltage will be available at RTL-SDR V3 for LNA

```
$ sudo nano /etc/systemd/system/multi-user.target.wants/fr24feed.service

[Unit]
Description=Flightradar24 Decoder & Feeder
After=network-online.target

[Service]
Type=simple
Restart=always
LimitCORE=infinity
ExecStartPre=-/usr/lib/fr24/install_dump1090.sh
ExecStartPre=-/usr/lib/fr24/unregister_kernel_modules.sh
ExecStartPre=-/usr/lib/fr24/create_missing_directories.sh
ExecStartPre=/home/pi/rtl_biast/build/src/rtl_biast -b 1
ExecStart=/usr/bin/fr24feed
User=fr24
Group=fr24
PermissionsStartOnly=true
StandardOutput=null

[Install]
WantedBy=multi-user.target
```

Note: dump1090.sh will only start the FR24 dump1090-mutability if the device is set as `dvbt` in the ini file

Flightroute format



Flightroute.sqb

The second database *flightroute.sqb* is unique to PlanePlotter and stores only the relationship between flight numbers and routes together with a date/time stamp showing the time at which the route data was updated - as shown below.

```
Flight   Route       Updatetime
-------  ---------   ----------
NWA815   EDDF-KATL   1263982991
RYR1A    EIDW-EGNM   1257415992
CFE29G   EGLC-EGPF   1257415992
KLM1074  EGCC-EHAM   1257415992
EIN63T   EBBR-EIDW   1257415992
EXS907   EGCC-LLBG   1257415992
BAW48    KSEA-EGLL   1282385657
EIN20C   EIDW-EGCC   1257415992
KLM669   EHAM-KDFW   1257415992
AFR1669  EGCC-LFPG   1257415993
```

The *flightroute.sqb* database is stored in the PlanePlotter Log files folder and is automatically created the first time the program is used. It is possible to download a populated database from the Yahoo PlanePlotter Group.

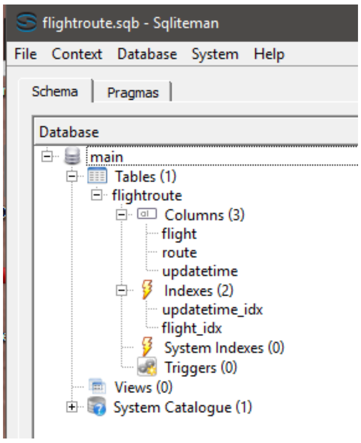
When PlanePlotter receives a message from a newly detected aircraft it will get the aircraft registration number from the *basestation.sqb* database and check the *flightroute.sqb* database to see if the flight number and route are already known. If they are then they will be displayed. As PlanePlotter receives no route information in the Mode-S messages it cannot in itself add any information to the database. This is achieved using third party add-ons such as Curt Deegan's Findflight that uses the *SetRouteByHex* method.

Ref: https://data.flightairmap.com/

Download Routes (tsv format) (2018/08/19 - 8.93M)

Using Sqliteman (PortableApps.com)

Create a new database `flightroute.sqb`:



```
CREATE TABLE "FlightRoute" (
    "flight" TEXT,
    "route" TEXT,
    "updatetime" INTEGER
);
CREATE INDEX "flight_idx" on flightroute (flight ASC);
CREATE INDEX "updatetime_idx" on flightroute (updatetime ASC);
```

Using Excel to patch this from the route.tsv file (the update time is set the same for all entries)

```
flight;route;updatetime
TAY41G;LFLL-N460;1263982981
TAY58F;EBLG-ELLX;1263982981
…
```

Save as fr.csv (CSV format)
Then import the data via Sqliteman – Database – Import Table Data …
Seems to work …

Feed Experience after the first 24/48 hours

Tue 21-08-2018 – 22:20 – site ID T-LSME4

The 3 element 5/8λ collinear Antenna is in the attic under a tiled roof about 9 m above ground. Connected with 1m (3.3ft) coax to the LNA and directly into the RTL SDR dongle, then a short USB into the Pi 3 B. Data connection through Wifi into the home network.

Antenna Ref: https://radarspotting.com/forum/index.php?topic=3943.0

Dongle RTL SDR V3, ADS-B Triple Filtered LNA (Bias Tee Powered)

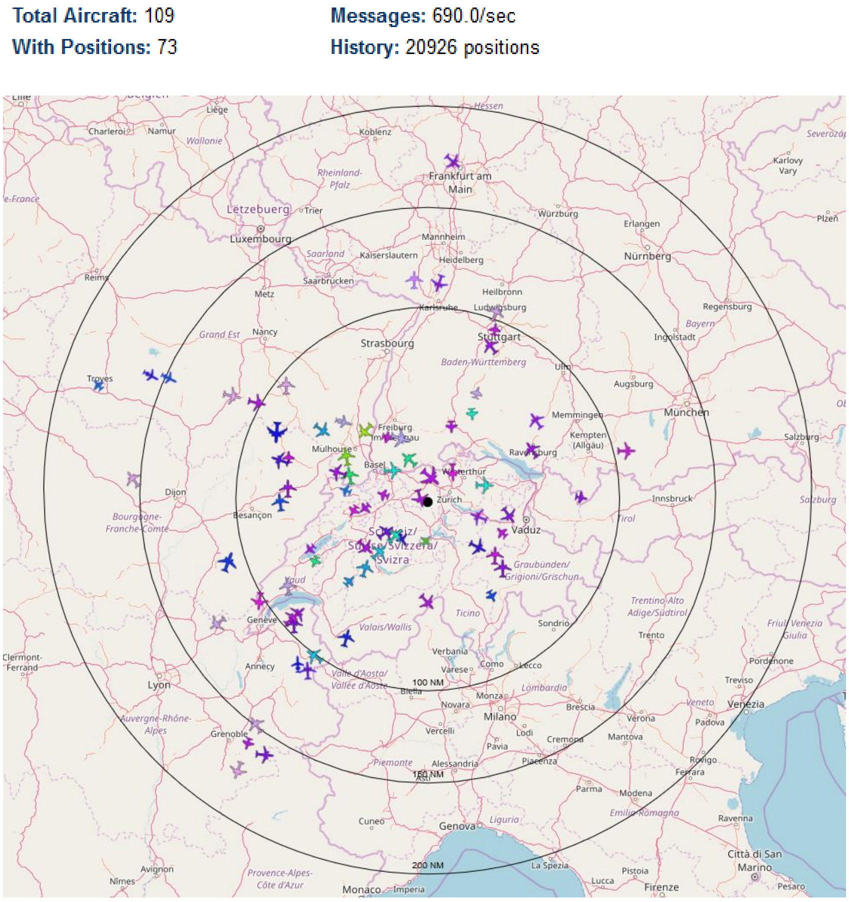
Dongle Ref: https://www.rtl-sdr.com/buy-rtl-sdr-dvb-t-dongles/

A word on data traffic.. piaware is running with about 3.2 kByte/sec outbound to flightaware servers (grung.hou.flightaware.com) and low inbound traffic as long as no program requests MLAT feedback. With a consumer on the mlat client port 30105 the inbound traffic also raises to about 3.5kByte/sec. The FR24, also with MLAT enabled, connections seem to be negligible compared to FA.

Expect about 270 Mbytes/day outbound with MLAT enabled and the observed traffic shown here.

Dump1090-fa (piaware) Website

Typical pattern



Total Aircraft: 109 **Messages:** 690.0/sec

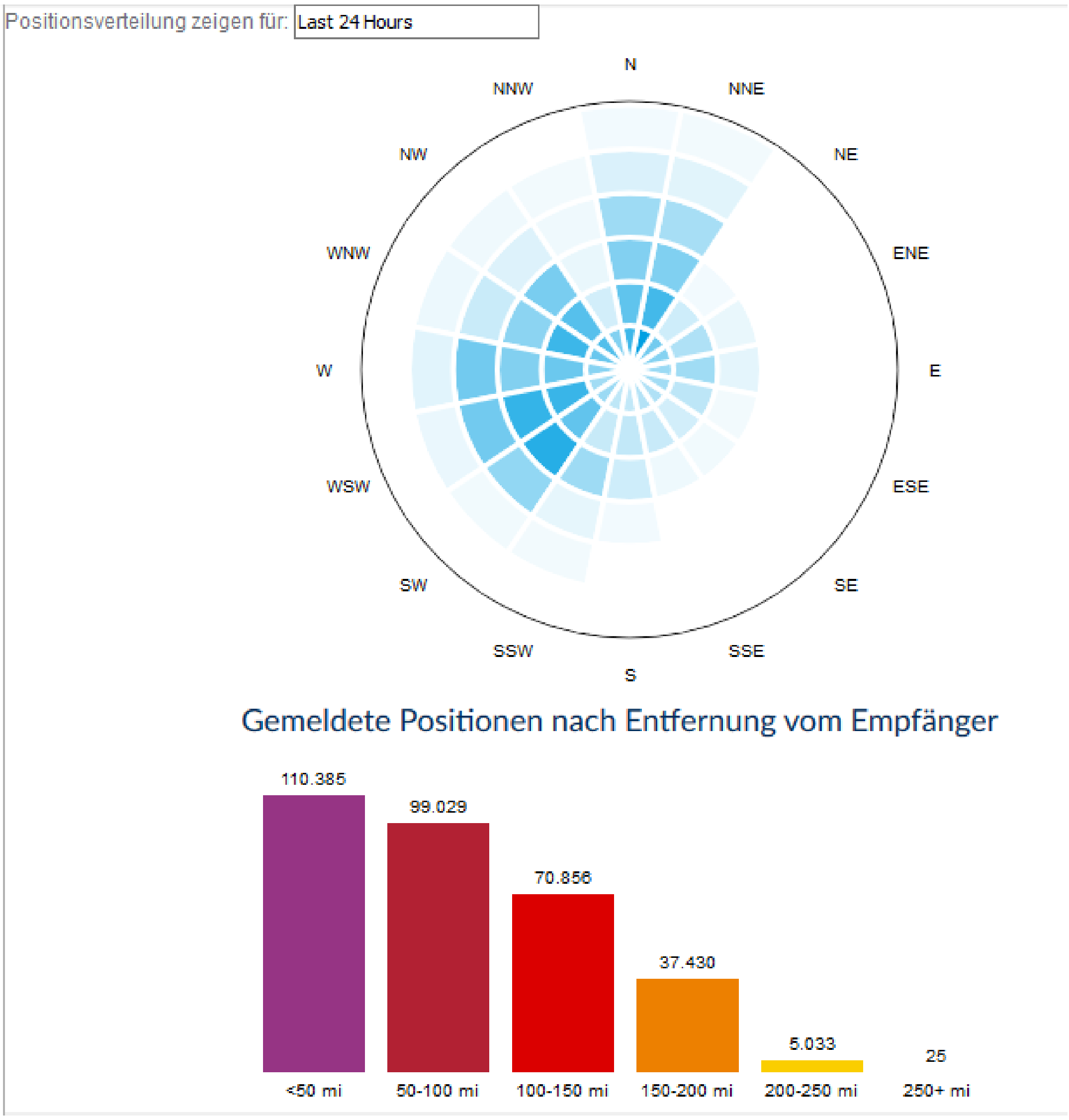
With Positions: 73 **History:** 20926 positions

Flightaware:

Stats 23.Aug. 2018 23:00:

ADS-B Aircraft	ADS-B Positions	MLAT Aircraft	MLAT Positions	Other Aircraft	Other Positions
11.612	1.163.622	2.157	634.553	2.698	821.731

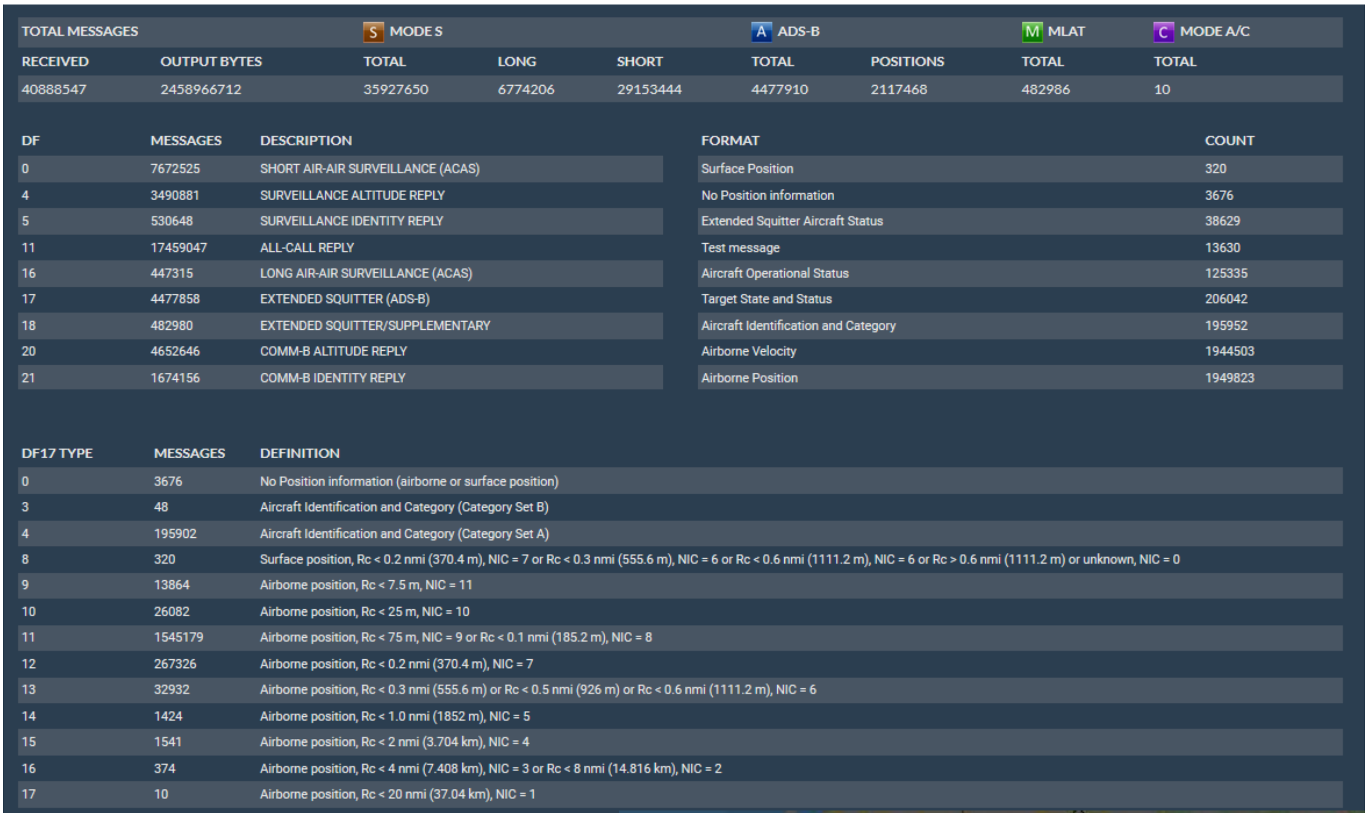
24 hours range



Positionsverteilung zeigen für: Last 24 Hours

Gemeldete Positionen nach Entfernung vom Empfänger

Modesmixer2 - 21h Stats



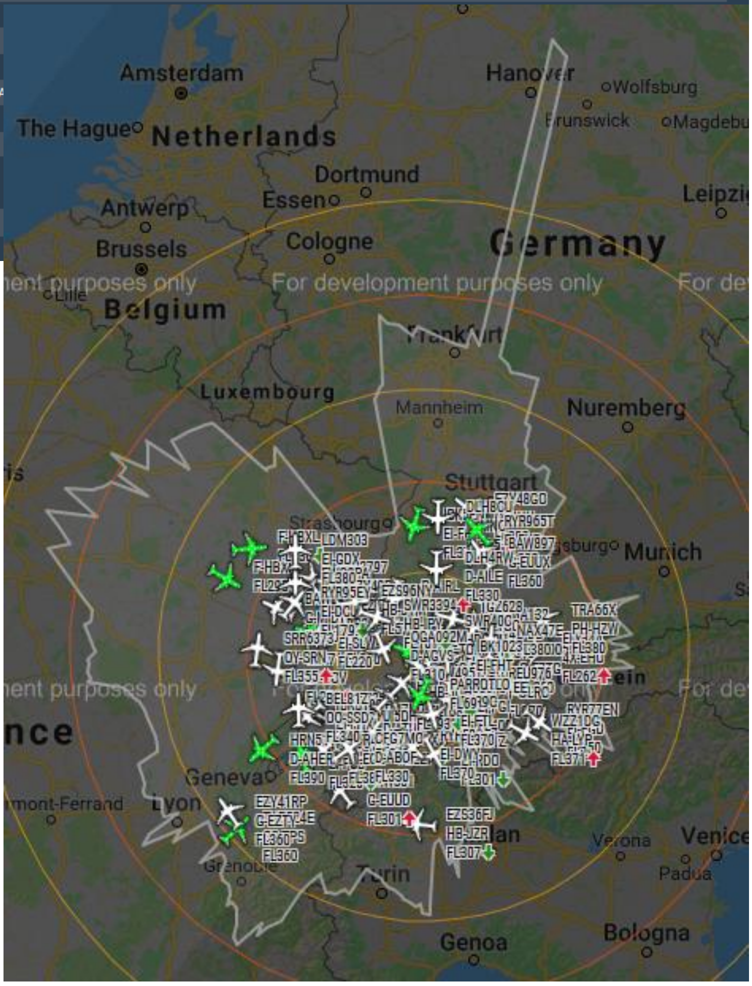
TOTAL MESSAGES		S MODE S			A ADS-B		M MLAT	C MODE A/C
RECEIVED	OUTPUT BYTES	TOTAL	LONG	SHORT	TOTAL	POSITIONS	TOTAL	TOTAL
40888547	2458966712	35927650	6774206	29153444	4477910	2117468	482986	10

DF	MESSAGES	DESCRIPTION
0	7672525	SHORT AIR-AIR SURVEILLANCE (ACAS)
4	3490881	SURVEILLANCE ALTITUDE REPLY
5	530648	SURVEILLANCE IDENTITY REPLY
11	17459047	ALL-CALL REPLY
16	447315	LONG AIR-AIR SURVEILLANCE (ACAS)
17	4477858	EXTENDED SQUITTER (ADS-B)
18	482980	EXTENDED SQUITTER/SUPPLEMENTARY
20	4652646	COMM-B ALTITUDE REPLY
21	1674156	COMM-B IDENTITY REPLY

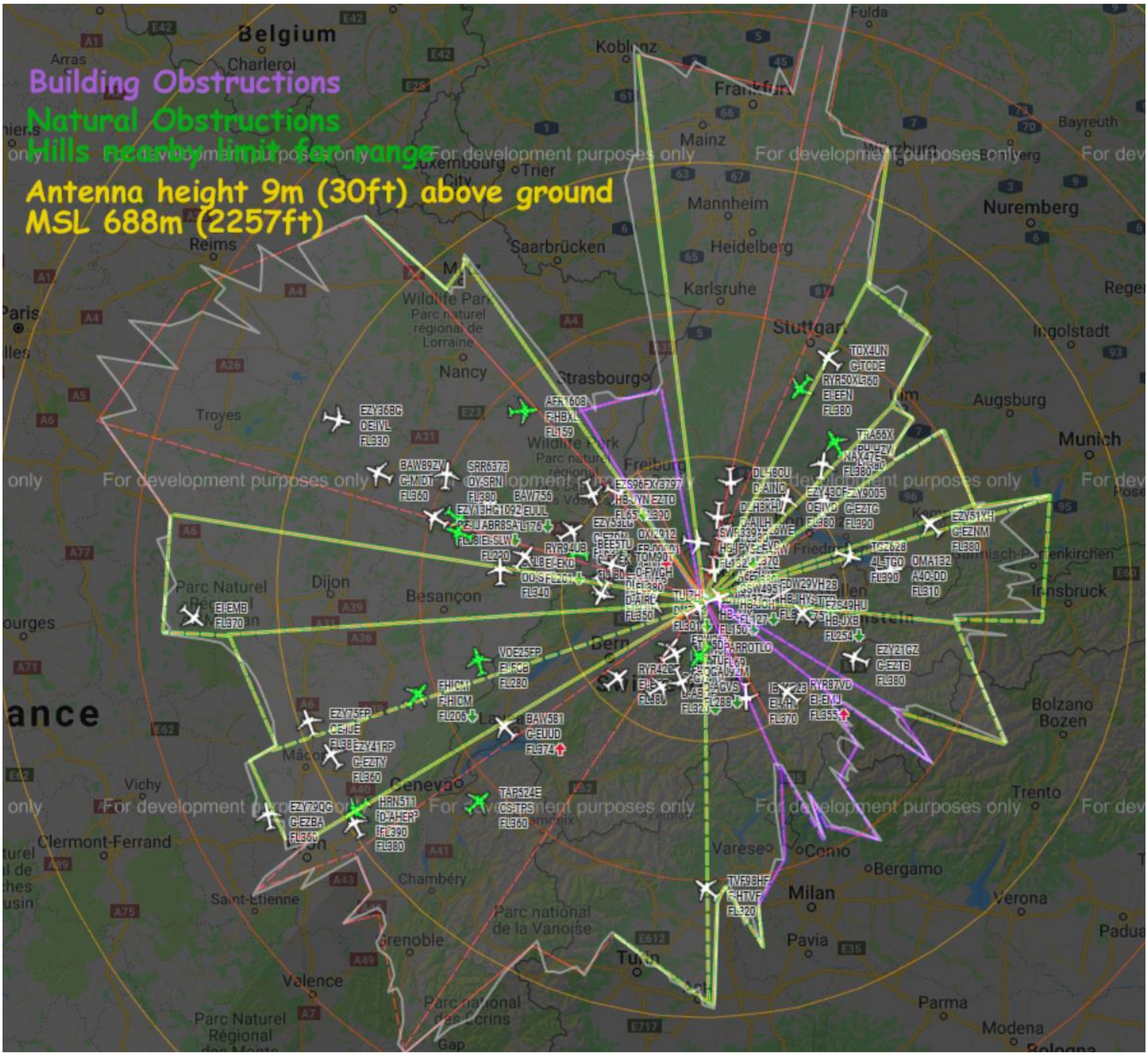
FORMAT	COUNT
Surface Position	320
No Position information	3676
Extended Squitter Aircraft Status	38629
Test message	13630
Aircraft Operational Status	125335
Target State and Status	206042
Aircraft Identification and Category	195952
Airborne Velocity	1944503
Airborne Position	1949823

DF17 TYPE	MESSAGES	DEFINITION
0	3676	No Position information (airborne or surface position)
3	48	Aircraft Identification and Category (Category Set B)
4	195902	Aircraft Identification and Category (Category Set A)
8	320	Surface position, Rc < 0.2 nmi (370.4 m), NIC = 7 or Rc < 0.3 nmi (555.6 m), NIC = 6 or Rc < 0.6 nmi (1111.2 m), NIC = 6 or Rc > 0.6 nmi (1111.2 m) or unknown, NIC = 0
9	13864	Airborne position, Rc < 7.5 m, NIC = 11
10	26082	Airborne position, Rc < 25 m, NIC = 10
11	1545179	Airborne position, Rc < 75 m, NIC = 9 or Rc < 0.1 nmi (185.2 m), NIC = 8
12	267326	Airborne position, Rc < 0.2 nmi (370.4 m), NIC = 7
13	32932	Airborne position, Rc < 0.3 nmi (555.6 m) or Rc < 0.5 nmi (926 m) or Rc < 0.6 nmi (1111.2 m), NIC = 6
14	1424	Airborne position, Rc < 1.0 nmi (1852 m), NIC = 5
15	1541	Airborne position, Rc < 2 nmi (3.704 km), NIC = 4
16	374	Airborne position, Rc < 4 nmi (7.408 km), NIC = 3 or Rc < 8 nmi (14.816 km), NIC = 2
17	10	Airborne position, Rc < 20 nmi (37.04 km), NIC = 1
18	61071	Airborne position, Rc > 20 nmi (37.04 km) or unknown, NIC = 0
19	1944477	Airborne Velocity Message
23	13630	Test message
28	38629	Emergency/Priority Status and Mode A Code or 1090ES TCAS RA
29	206039	Target State and Status
31	125334	Aircraft Operational Status

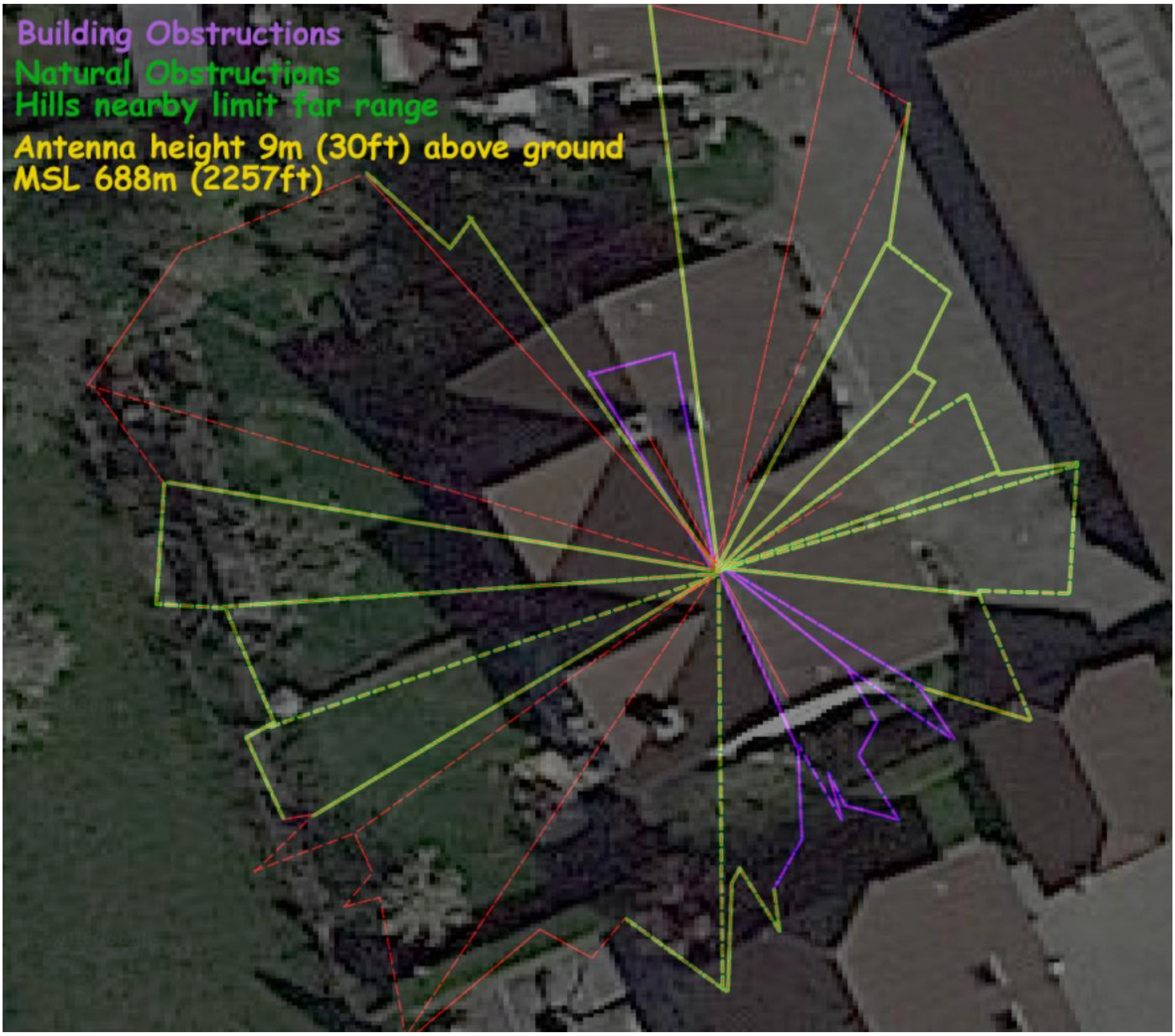
UPTIME
START TIME
DURATION



Observed Range Assessment


Building Obstructions
Natural Obstructions
Hills nearby limit for range
Antenna height 9m (30ft) above ground
MSL 688m (2257ft)
For development purposes only

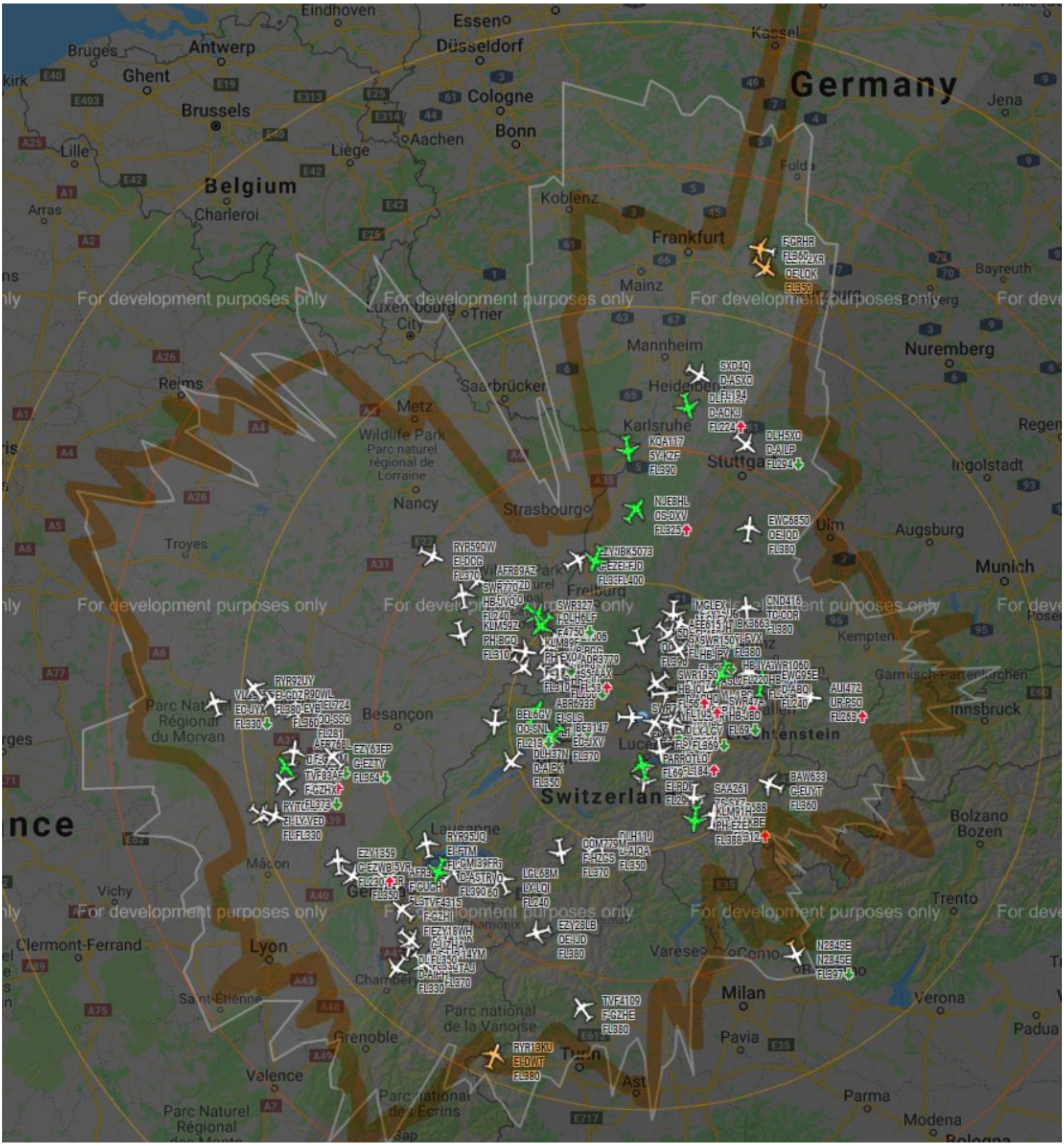
Map multiple Satellite / Terrain Images with different scales with the observed range and assess what may limit the detected signals.


Building Obstructions
Natural Obstructions
Hills nearby limit far range
Antenna height 9m (30ft) above ground
MSL 688m (2257ft)

Moving the Antenna

After identifying the obstructions the antenna was moved about 1m (3.3ft) higher and 4m (13ft) SW

New range plot where the orange outline is the 24h range before the move.


Eindhoven
Essen
Kassel
Bruges
Antwerp
Düsseldorf
Germany
Ghent
Jena
Brussels
Cologne
Aachen
Bonn
Lille
Liège
Fulda
Belgium
Charleroi
Koblenz
Arras
Frankfurt
Bayreuth
Mainz
Luxembourg City
Trier
Mannheim
Nuremberg
Reims
Heidelberg
Saarbrücken
Metz
Karlsruhe
Regen
Wildlife Park
Parc naturel régional de Lorraine
Stuttgart
Ingolstadt
Nancy
Strasbourg
Ulm
Augsburg
Troyes
Munich
Freiburg
Kempten
Garmisch-Partenkirchen
Parc Naturel Régional du Morvan
Besançon
Innsbruck
Liechtenstein
Lucerne
Switzerland
Bolzano
Bozen
Lausanne
Mâcon
Vichy
Geneva
Trento
Clermont-Ferrand
Lyon
Chamonix
Varese
Como
Chambéry
Milan
Verona
Saint-Étienne
Padua
Parc national de la Vanoise
Grenoble
Pavia
Turin
Asti
Valence
Parma
Parc national des Écrins
Modena
Parc Naturel Régional
Bologna
For development purposes only
F-GRHR
FL350
OE-LQK
SXD4Q
D-ASXQ
KQA117
5Y-KZF
FL390
DLH5XO
D-AILP
FL294
NJE8HL
CS-DXV
FL325
EWG6850
OE-IQD
FL380
RYR59DW
EI-DCG
FL370
RYR92UY
CND416
TC-DOR
AUI472
UR-PSQ
FL263
BAW633
G-EUYT
FL360
SAA261
EZY63EP
G-EZTY
FL364
EZY1359
RYR95JQ
LGL68M
LX-LQI
FL240
DLH11U
EZY28LB
OE-IJD
N234SE
FL397
TVF4109
F-GZHE
RYR13KU
EI-DWT
FL380