RealTraffic Developer Documentation for RT v10.0

In version 10, RealTraffic (RT) has evolved into both a traffic and weather source. This is very important because flying with real air traffic in the vicinity is only realistic if the weather experienced by the traffic around you is mirrored in the simulator you are flying. While this has always been supported by RT for landing and departing traffic by providing the correct air pressures and nearest METARs, RT now provides global weather coverage from the crowded skies of Europe to the remotest corners in the polar and oceanic regions. With satellite derived air traffic now available in RT v10, you no longer are flying on your own during a long-haul flight, and it is even more important that your aircraft experiences the same head- and tail winds as the other traffic, as well as the same air temperatures, so that the Mach number you're flying on the track system matches that of traffic ahead and behind you – so you don't catch up to them, and they don't catch up to you.

RealTraffic obtains traffic and weather information for the location desired, either by tracking the flight simulator's location (information which you have to provide), or by using the spotter location. If you use the spotter location, you only need to listen for UDP packets. If you want to integrate it into a simulator application however, you need to provide your location, and ideally attitude, in regular intervals to RealTraffic.

In both cases, RealTraffic will transmit weather and traffic as UDP packets in approximately 2 second intervals, traffic is limited to approximately 100 NM of range around the center of the position provided, and weather is given for the 7 nearest airports providing METARs as well as actual conditions at your present altitude, to include winds and temperatures aloft, turbulence, and cloud information.

Providing your simulator position to RealTraffic

Your simulator plugin needs to provide a TCP socket server connection. By default, RealTraffic expects port 10747 (but this can be changed and is configurable in the Settings panel of the RealTraffic User Interface). Once your plugin detects that RealTraffic has connected to it, you should transmit the following parameters ideally at 5 Hz (i.e. 5 times per second):

- Pitch in radians * 100000
- Bank in radians * 100000
- Heading (or Track) in radians
- Altitude in feet * 1000
- True air speed in meters per second
- Latitude in radians
- Longitude in radians
- All of that preceded by the characters "Qs121="

An example string would look like this:

"Qs121=6747;289;5.449771266137578;37988724;501908;0.6564195830703577;-2.1443275933742236"

If your simulator is moving slowly, you need to be careful to inject something useful such as true heading instead of track, because track calculations can get a bit iffy with limited precision latitude/longitude points.

For a correct formatting example, look at the following Java code, which translates almost as is to C, just use %0.15f for the float formatting rather than %.15f:

Example code for a correctly formatted string in Java:

```
message = String.format(Locale.US, "Qs121=%d;%d;%.15f;%d;%d;%.15f;%.15f",
          (int)(pitch_in_degs * 100000 * d2r), (int)(bank_in_degs * 100000*d2r)*-1,
          track_in_degs * d2r, (int)(altitude_in_m * 3028), TAS_in_mps, latitude * d2r,
          longitude * d2r);
```

Once you are injecting this data correctly, RealTraffic will spring to life and indicate your position on the map display, as well as show any traffic around you.

If you want the Foreflight and/or Garmin Pilot apps to be supported correctly, you must inject position and attitude updates at 5 Hz.

Using the Destination Traffic/Weather feature

To receive destination traffic and weather on ports 49005/49006, send the following message to RT:

Qs376=YSSY;KLAX

This would set the destination to KLAX and start broadcasting destination traffic and weather. This feature is only available for professional license holders of RT.

How to find out license and version information

Send the string

Qs999=version

to RT and you'll receive a JSON string back with the following information:

{ "version":"10.0.240","level": 2 }

Where version is the current version being run in format major.minor.build, and the subscription level with 0 = pre-v9-release standard license, 1 = post-v9-release standard license, 2 = professional license.

Controlling the time offset from the simulator

To create a more coherent simulation experience, the time offset for RealTraffic can be controlled from the simulator. Simulator time offset control can be enabled/disabled both in the RT GUI and from the simulator itself.

send Qs999=timeoffsetcontrol=? to inquire who's in control. you'll either receive

{ "timeoffsetcontrol": "Simulator" }

or

{ "timeoffsetcontrol": "RealTraffic" }

and you can send this to control it:

Qs999=timeoffsetcontrol=Simulator

or

Qs999=timeoffsetcontrol=RealTraffic

To set the time to be used in RT, send the UTC time formatted as epoch milliseconds with a Qs123 message as follows:

Qs123=1674956315329

You can verify in the RT GUI that the correct time offset is running. Note that this only works for professional licenses, you can capture that in your plugin by first checking what type of license the user has, see “How to find out license and version information” above for that. As of this writing, time offsets are limited to 2 weeks into the past.

Reading weather and traffic information

RealTraffic broadcasts all traffic and weather information via UDP on the local network, but can provide much more detailed weather via the TCP connection you already are connected to. The broadcast weather information pertains to the nearest 7 airports. ADS-B altitudes for traffic are already pressure corrected. You can use the altitudes as they are reported by RT and inject the traffic into your simulator using those altitudes.

UDP Weather Broadcasts

The weather messages are broadcast as UDP packets once every 10 seconds on port 49004 containing a JSON string with the nearest airport data along with a lot of additional data collections of the following format:

{
"ICAO":"KMCE",
"QNH":1019.7,
"METAR": "KMCE 130453Z AUTO 14005KT 6SM -RA BR OVC043 13/12 A3011 RMK AO2 RAB0354E08B11E21B37 SLP197 P0003 T01330117",
"TA":18000,

"locWX": {
"Info": "2023-03-13_0558Z",
"ST":11.4,
"SWSPD":9.59,
"SWDIR":146.29,
"SVis":18224,
"LLCC":17,
"MLCC":49.7,
"HLCC":0,
"DZDT":-0.0302,
"PRR":0.55,
"T":-52.57,
"WDIR":268.4
"WSPD":137.27,
"TPP":10084.52,
"SLP":1019.73,
"Profiles":"RTFX1 ^N3737.2 ^W12034.8 ^FL381 265/071 -51 ^FL361 267/075 -52 ^FL341 269/078 -53 ^FL321 269/082 -51 ^FL301 268/086 -48 ^^",
},

"AM": [
"KMER 130535Z AUTO 17007KT 10SM SCT028 OVC035 13/13 A3012 RMK AO1",
"KCVH 130535Z AUTO 00000KT 10SM OVC012 13/12 A3013 RMK A01",
"KNLC 130456Z AUTO 13007KT 10SM FEW090 13/11 A3011 RMK AO2 SLP196 T01330111",
"KFAT 130453Z 10008KT 6SM RA BKN060 OVC070 13/10 A3012 RMK AO2 SLP196 P0009 T01330100",
"KSNS 130453Z AUTO 01008KT 10SM SCT045 14/12 A3010 RMK AO2 SLP202 T01390117",
"KE16 130535Z AUTO 17003KT 2 1/2SM BR FEW008 OVC015 13/12 A3013 RMK AO2 PWINO PNO"]
}

The fields are:

- **ICAO:** the ICAO code of the nearest airport,
- **QNH:** the reported pressure in hPa, often to within 0.1 hPa precision,
- **METAR:** contains the full METAR received,
- **TA:** the transition altitude in ft
- **locWX:** is the location weather at the present position, and contains the following information:

- **Info** contains the timestamp if data is present, if no data is present it contains the reason for no data. Valid reasons are:
    - TinyDelta: Means that less than one minute has elapsed since the last query, or the lateral/vertical distance to the last query is less than 10NM / 2000ft.
    - error: Means there was an error on the server, the description after error contains more information.
    - If no data is present, all fields are set to -1
- **ST**: surface temperature in C
- **SWSPD**: surface wind speed in km/h
- **SWDIR**: surface wind direction in km/h
- **SVis**: surface visibility in meters
- **PRR**: precipitation rate on ground in mm/h.
    - < 0.5: none - or drizzle if not zero
    - < 2.5: light
    - < 7.5: moderate
    - > 7.5: heavy
- **LLCC**: low level cloud cover in percent (lowest third of troposphere)
- **MLCC**: medium level cloud cover in percent (medium third of troposphere)
- **HLCC**: high level cloud cover in percent (top third of troposphere)
- **DZDT**: vorticity of atmospheric layer. Larger values are indicative of turbulence. As a rule of thumb:
    - < 0.05: still air
    - < 0.5: light
    - < 1: medium (spills coffee)
    - > 1: strong (unattached objects and people go flying)
    - > 2: severe (unable to retain positive aircraft control at all times – block altitude needed)
- **T**: OAT/SAT in C
- **WDIR**: wind direction in degrees
- **WSPD**: wind speed in km/h
- **TPP**: tropopause height in meters
- **SLP**: sea level pressure in hPa
- **Profiles**: contains a vertical cross section / profile at the current location, formatted as "Aerowinx Format D". This is a common output format used in flight planning and dispatch software. The format contains a line with caret symbols as newline placeholders. In the example above, the line reads as follows:

```
RTFX1
N3737.2
W12034.8
FL381  265/071 -51
FL361  267/075 -52
FL341  269/078 -53
FL321  269/082 -51
```

FL301 268/086 -48

The first line contains the waypoint name (RT Fix 1)
The second and third lines contain the coordinates of the fix
Lines 4 – 8 contain the altitude (or FL) in 2000ft increments above and below the current FL, and the Winddirection/Windspeed and OAT in C for each of those altitudes.

- **AM**: additional metars. Contains a list of an additional 6 nearest METARs in a forward looking direction (if available). You can use these METARs to gain additional understanding of the weather surrounding the aircraft.

The traffic data is broadcast as UDP packets, and these come in several formats:

- XTRAFFICPSX (Foreflight format) which is broadcast on port 49002,
- RTTFC (RT Traffic format), broadcast on port 49005
- RTDEST (destination Traffic), broadcast on port 49006

Further to this, the simulator position is re-distributed (for the benefit of other third party apps) as XGPS and XATT messages on the same port as Foreflight messages, 49002.

XTRAFFICPSX format

In foreflight format, each traffic in your area will be broadcast in a string with the following format:

“XTRAFFICPSX,hexid,lat,lon,alt,vs,airborne,hdg,spd,cs,type”

where the parameters are:

- Hexid: the hexadecimal ID of the transponder of the aircraft. This is a unique ID, and you can use this ID to track individual aircraft.
- Lat: latitude in degrees
- Lon: longitude in degrees
- Alt: altitude in feet
- Vs: vertical speed in ft/min
- Airborne: 1 or 0
- Hdg: The heading of the aircraft (it’s actually the true track, strictly speaking.)
- Spd: The speed of the aircraft in knots
- Cs: the ICAO callsign (Emirates 413 = UAE413 in ICAO speak, = EK413 in IATA speak)
- Type: the ICAO type of the aircraft, e.g. A388 for Airbus 380-800. B789 for Boeing 787-9 etc.

The GPS and Attitude messages are as follows:

“XGPSPSX,lon,lat,alt,track,gsp”

Where the parameters are:

- Lon: Longitude in degrees (remember west = negative)
- Lat: Latitude in degrees (north = positive)
- Alt: Altitude in meters
- Track: True track
- Gsp: Ground speed in meters per second

"XATTPSX,hdg,pitch,roll"

Where the parameters are:

- Hdg: The heading of the aircraft (true heading)
- Pitch: The pitch in degrees, positive = pitch up
- Roll: The roll angle in degrees, positive = right roll

RTTFC/RTDEST format

This format contains almost all information we have available from our data sources. The format is:

"RTTFC,hexid, lat, lon, baro_alt, baro_rate, gnd, track, gsp, cs_icao, ac_type, ac_tailno, from_iata, to_iata, timestamp, source, cs_iata, msg_type, alt_geom, IAS, TAS, Mach, track_rate, roll, mag_heading, true_heading, geom_rate, emergency, category, nav_qnh, nav_altitude_mcp, nav_altitude_fms, nav_heading, nav_modes, seen, rssi, winddir, windspd, OAT, TAT, isICAOhex,augmentation_status,authentication"

- hexid = the transponder's unique hexadecimal ID
- lat = latitude
- lon = longitude
- baro_alt = barometric altitude
- baro_rate = barometric vertical rate
- gnd = ground flag
- track = track
- gsp = ground speed
- cs_icao = ICAO call sign, the actual callsign an flight is known as to air traffic services
- ac_type = aircraft type
- ac_tailno = aircraft registration
- from_iata = origin IATA code
- to_iata = destination IATA code
- timestamp = unix epoch timestamp when data was last updated
- source = data source
- cs_iata = IATA call sign (the flight number you would see on an airport arrival/departure announcement board
- msg_type = type of message

- alt_geom = geometric altitude (WGS84 GPS altitude)
- IAS = indicated air speed
- TAS = true air speed
- Mach = Mach number
- track_rate = rate of change for track
- roll = roll in degrees, negative = left
- mag_heading = magnetic heading
- true_heading = true heading
- geom_rate = geometric vertical rate
- emergency = emergency status
- category = category of the aircraft
- nav_qnh = QNH setting navigation is based on
- nav_altitude_mcp = altitude dialled into the MCP in the flight deck
- nav_altitude_fms = altitude set by the flight management system (FMS)
- nav_heading = heading set by the MCP
- nav_modes = which modes the autopilot is currently in
- seen = seconds since any message updated this aircraft state vector
- rssi = signal strength of the receiver
- winddir = wind direction in degrees true north
- windspd = wind speed in kts
- OAT = outside air temperature / static air temperature
- TAT = total air temperature
- isICAOhex = is this hexid an ICAO assigned ID.
- Augmentation_status = has this record been augmented from multiple sources
- Authentication = authentication status of the license, safe to ignore.

The “source” field can contain the following values:

- adsb_icao: messages from a Mode S or ADS-B transponder.
- adsb_icao_nt: messages from an ADS-B equipped "non-transponder" emitter e.g. a ground vehicle.
- adsr_icao: rebroadcast of an ADS-B messages originally sent via another data link
- tisb_icao: traffic information about a non-ADS-B target identified by a 24-bit ICAO address, e.g. a Mode S target tracked by SSR.
- adsc: ADS-C (received by satellite downlink) – usually old positions, check tstamp.
- mlat: MLAT, position calculated by multilateration. Usually somewhat inaccurate.
- other: quality/source unknown. Use caution.
- mode_s: ModeS data only, no position.
- adsb_other: using an anonymised ICAO address. Rare.
- adsr_other: rebroadcast of ‘adsb_other’ ADS-B messages.
- tisb_other: traffic information about a non-ADS-B target using a non-ICAO address
- tisb_trackfile: traffic information about a non-ADS-B target using a track/file identifier, typically from primary or Mode A/C radar

A couple of example data sets:

```
RTTFC,11234042,-33.9107,152.9902,26400,1248,0,90.12,490.00,AAL72,B789,
N835AN,SYD,LAX,1645144774.2,X2,AA72,adsb_icao,27575,320,474,0.780,
```

```
0.0,0.0,78.93,92.27,1280,none,A5,1012.8,35008,-1,71.02,
autopilot|vnav|lnav|tcas,0.0,-21.9,223,24,-30,0,1,170124
```

```
RTTFC,10750303,-33.7964,152.3938,20375,1376,0,66.77,484.30,UAL842,B789,
N35953,SYD,LAX,1645144889.8,X2,UA842,adsb_icao,21350,343,466,0.744,-0.0,
0.5,54.49,67.59,1280,none,A5,1012.8,35008,-1,54.84,
autopilot|vnav|lnav|tcas,0.0,-20.8,227,19,-15,14,1,268697
```

Using Satellite Imagery

As of version 8, global real-time satellite coverage is available. Of particular interest for developers are the RDW (Radar raw) data and the true color image.

The geostationary satellites used to create the imagery are

- GOES17 at 137.2 degrees West
- GOES16 at 75.2 degrees West
- Meteosat 10 at 0 degrees
- Meteosat 8 at 45 degrees East
- Himawari 8 at 140.7 degrees East

Meteosat updates every 15 minutes, the others every 10 minutes. Best resolution for Meteosat is 1km per pixel, for the others 500m per pixel.

Processing time for each satellite is approximately 8 minutes, therefore by the time the data is available, the lag behind real-time is between 8 and 18 minutes depending on the scan time. GOES16 and 17 as well as the Himawari 8 satellites don't take a single picture like a camera, but rather scan the earth in 10 horizontal scan lines, each taking about 45 seconds to complete. The Meteosat satellites are a fair bit older and use a different method of scanning the earth. The entire satellite is rotating and multiple scan lines are taking data as the earth zips by the apertures.

The format of the images is in high quality jpg encoding. Resolution is 1600 x 1600 pixel for each tile (equivalent to the native resolution available at the sub-satellite point), each spanning 10 x 10 degrees in a Mercator projection. This makes it relatively easy to derive the position of each pixel in simulator space.

If you would like any products not already available, please inquire and I can make them available as all data is derived from the satellite's raw data feeds.

True color image

This is a color enhanced rendition of the earth with all clouds. Native resolution is 500m per pixel at the sub-satellite point (less the further away one goes). This image is available from RT if it is selected as the currently visible image.

False colour infrared

This is the 10.4 micrometer channel in the infrared. It is coloured such that cloud top temperatures of -40C and less are rendered in colour, making it particularly useful to identify regions of convective activity, e.g. thunderstorms.

PWV – Precipitable Water Vapour

This image is a false colour composite comprised of three water vapour channels: Blue represents water vapour at about 9km altitude, green at 5km altitude, and red at 3km altitude. This image is useful to identify Jetstream locations, as well as areas of clear air turbulence (where Jetstream collide or make sharp turns).

RDR – Radar image

This is a product created to estimate radar returns as they would be shown on an on-board radar system.

RDW – Raw radar image

This is the image the radar image is based on, but is in grayscale, with full white indicating strongest returns. This allows a plugin developer to implement gain control in the on-board radar. This image is not available for display in the GUI, but it is available from the RT web interface regardless of which satellite overlay is selected by the user.

How to access the data

RealTraffic has a web interface at port 60888. Connect a web-browser to this port and you can download the data products (or any app you might write is able to talk to RT on that port):

- Launch a web browser and navigate to the host where RT is running on, usually that's localhost: http://localhost:60888
- This will show you a JSON string with the UTC timestamp (in epoch seconds) of the currently shown image, it also shows which image is being shown, as well as the lower left corner latitude and longitude of the image. Each image is 10 x 10 degrees (mercator projected). This allows you to calculate each pixel's latitude and longitude easily.
- navigate to http://localhost:60888/data and the currently used image in the GUI is shown in the browser.
- navigate to http://localhost:60888/rdw and a satellite derived radar reflectivity map is shown. This could be used to create a realistic radar display in the cockpit. It's a grayscale image that can be used to simulate a radar return based on the whiteness of the pixels shown.

If you have any questions or would like to see other satellite image products made available, don't hesitate to contact me via balt@inside.net.