Discord METAR Bot

1. General

1.1. Overview

This discord bot provides realtime meteorological aerodrome reports (METAR) and terminal aerodrome forecasts (TAF) of a requested station via a discord channel. METAR and TAF will be provided:

– **primarily in a more readable or even completely decoded way in which all units are stated and converted to SI**. This custom standard strives to be a more modern and better compromise between being able to quickly brief the relevant information and enhancing its readability.
  Important: While the code has been proven reliable over 1 year of daily private use, **it is not certified to be used in real flight operations.**
– secondarily in their original format which is guaranteed to be correct by NOAA, mainly for cross-checking

Once a station is requested, 2 things happen:

1. The current METAR and, if requested, TAF are provided. Only these reports write how long ago they have been published.
2. The station is registered as the currently subscribed station. This means as soon as this station publishes a new METAR and, if requested, a new TAF, it will be automatically provided. This is a convenience functionality and the reason why using this bot is only recommended in a designated bot channel whose name is specified in `discord_bot_channel_names.config`. Channels with the name `bots`, `botspam`, and `metar` will be recognised by default.

1.2. Examples

1.2.1. Example 1: METAR

```
EDDL 311120Z AUTO 23018KT 9999 -SHRA FEW029 FEW///TCU 12/08 Q0995 RESHRA TEMPO 23020G30KT
```

```
EDDL 2023-03-31T11:20 AUTO 230°09m/s 10km+ -SHRA FEW930m|880m FEW///TCU 12°C/08°C Q099,5kPa
RESHRA
TEMPO 230°10G15m/s
```

Note the following differences:

– full datetime
– wind direction separated to wind speed
– wind direction, visibility, cloud altitude, temperature, and QNH all stated with their unit reducing ambiguity
– clouds given with altitude and height if they're different after rounding

1.2.2. Example 2: TAF

```
TAF EDDL 311100Z 3112/0118 22015G25KT 9999 BKN030 PROB30
TEMPO 3112/3113 24020G35KT SHRA BKN030TCU
TEMPO 3113/0103 RA PROB40
TEMPO 3114/3124 22020G30KT
BECMG 0100/0103 21013KT BKN012
TEMPO 0103/0118 4000 RADZ BKN007
BECMG 0106/0109 28010KT
```

```
TAF EDDL 2023-03-31T11:00 31T12/04-01T18 220°08G13m/s 10km+ BKN960m|910m
PROB0,3 TEMPO 31T12/13 240°10G18m/s SHRA BKN960m|910m|TCU
TEMPO 31T13/04-01T03 RA
PROB0,4 TEMPO 31T14/04-01T00 220°10G15m/s
BECMG 04-01T00/03 210°07m/s **BKN410m|370m**
TEMPO 04-01T03/18 **4,0km** RADZ **BKN260m|210m**
BECMG 04-01T06/09 280°05m/s
```

Note the following differences:

- Timespans only contain the minimum amount of necessary information and are [ISO8601](#) compliant. If only the hour changes, only that is stated in the end. If day and hour change, the day is also provided and so on.
- Information violating the customisable weather minimums are encapsulated with double asterisks ** to quickly draw attention at a glance. This is also true for METAR.

1.2.3. Example 3: USA

```
KPHX 051151Z 12006KT 10SM FEW240 08/M08 A3010 RMK AO2 SLP186 T00831078 10133 20083 53005
```

```
KPHX 2023-04-05T11:51 (420s ago) 120°03m/s 16km FEW7.700m|7.300m 08°C/-08°C A101,9kPa
RMK AO2 SLP101,86kPa 08,3°C/-07,8°C TX6h/13,3°C TN6h/08,3°C ΔPRES3h/+50Pa
```

Note the following differences:

- Altimeter settings in $[\mathrm{inHg}]$ are converted to $[\mathrm{Pa}]$, but the letter A remains to let the reader know that this value has been converted. This should usually not be a problem though as $0,01\mathrm{inHg} < 100\mathrm{Pa}$, meaning precision is lost and not falsely added.
- Sea level pressure (SLP) is converted from $[10\mathrm{Pa}\ \text{-}90\mathrm{kPa}\ \text{or -}100\mathrm{kPa}\ \text{wtf}]$ to $[\mathrm{Pa}]$.
- Explicitly requested METAR and TAF state how long ago they have been published.
- Weather station codes in the remark section are changed to be readable and actually provide value.

1.2.4. Example 4: Russia

```
UEEE 051230Z 30005MPS 9999 SCT030CB M15/M22 Q1009 R23L/490139 NOSIG RMK QFE748
```

```
UEEE 2023-04-05T12:30 (1,5ks ago) 300°05m/s 10km+ **SCT1.000m|910m|CB** -15°C/-22°C Q100,9kPa
**R23L/SNOW:DRY/0,51~1/1mm/MEDIUM~GOOD**
NOSIG
RMK QFE099,7kPa
```

Note the following differences:

– **Runway condition codes are decoded**, yet remain relatively compact.
– QFE is converted from $[\mathrm{mmHg}]$ to $[\mathrm{Pa}]$ to actually provide value.

1.3. Core Advantages

– METAR and TAF encoding with consistent use of units and better readability
– better readability makes runway state decoders obsolete and unlocks the additional information of USA weather station codes
– easy access and simple usability via discord
– simple staying up-to-date with station subscription feature, just scroll down

2. How to Install by Inviting My Bot User (recommended)

This method is recommended if you just want to use the bot out of the box and set and forget it. I have it running 24/7, there should be no problems in availability. Updates are also automatically applied as soon as I deploy them on my system.

1. Invite my bot user to your server by clicking [this link](#).
2. Make sure to have a text channel named `bots`, `botspam`, or `metar` on your server. **The bot will only react in these specific channels.**

If you need another channel name the bot reacts to, either contact me or follow the guideline in [chapter 3.](#) to set up your own bot instance.
It is recommended to only have 1 of the beforementioned channels on your server.

3. How to Install by Setting Up Your Own Bot

This method is recommended if you want to have full control about the bot's behaviour and maybe change some things for yourself here and there.

1. Download the source code or download a release `Metric METAR for Discord.exe`.

2. Copy your discord bot token into `discord_bot.token`.

   1. Create a discord application [here](#).
   2. Create your bot.
   3. Add it to your server.
   4. Copy your token into `discord_bot.token`.
   If you don't know how to do these steps, I recommend [this tutorial](#).

3. Copy your discord bot channel name into `discord_bot_channel_names.config`.

4. Execute `main_outer.py` with python or execute the compiled `Metric METAR for Discord.exe`.

4. How to Use

Once set up, using the bot is dead-easy. Write one of the following commands into the designated bot channel:

– current METAR: `{ICAO code}`
  Example: `EDDF`
– current METAR and TAF: `{ICAO code} TAF`
  Example: `EDDF TAF`

5. Possible Future Features

These are the features I am thinking about implementing in the future. Their stages vary widely from "hmm might be nice to have that" to "I have it almost done, but there is this 1 problem I can't solve.". So don't get your hopes up too high, but I'm very much open for feature requests and discussions.

– aerodrome information command
– download VFR charts command
– use of official database instead of open source database