AvXML Converter

1. Introduction

Meteo-France, has evaluated tools related to translating TAC to AvXML [1] in the frame of ICAO amendments 76, 77 and 78. The studied tools should have the capability to convert METAR, TAF and SIGMET messages to AvXML (see annex I).

Several existing open source tools and libraries were studied in order to convert from METAR and TAF into XML. Between them, we highlight metaf2xml tool [2] already used in Meteo-France in the frame of TOPMET project[1] and Java Weather Library (jweather) [3]. Metaf2xml is a tool written in Perl which converts METAR and TAF into XML whereas jweather is a Java library for parsing raw weather data which provides an API to access to the METAR data. In both cases, it would be necessary to modify the source code to add SIGMET and yield the output in AvXML format. Additionally, jweather would need support for TAF messages as well.

ECMWF [4] converter which already converts METAR messages into AvXML-RC1 (Release Candidate) was also evaluated. This converter uses ecCodes[2] –a general purpose decoder able to decode binary and text messages– and Python as a back-end to produce AvXML messages. An interesting feature of ecCodes is that it allows the possibility of using different programming languages, like Java or C++ instead of Python, for writing the back-end of the converter.

Following evaluation of the different options, we considered ECMWF converter as the best candidate, as it gives us the possibility to write the back-end in different programming languages (Python, C++, Java ...) in function of the needed performance. Furthermore, ECMWF offers support this tool.

2. Converter architecture overview

The conversion process is composed of two different steps as shown in Figure [1]. The first step uses the ecCodes tool which takes as input messages (METAR, TAF ...) to be decoded and uses the rules to decode the messages.

EcCodes rules (see ecCodes rules example in annex I) are programming sentences, written in the own ecCodes language according to the specification of each kind of messages. These rules allow the ecCodes engine to interpret and decode METAR and TAF messages to represent them in a key/value format. At this stage, ecCodes provides an interface to make accessible the key/values output in several programming languages.

[1] *TOPMET is a validation project of SESAR. The purpose is to make available some new meteorological data product for Air Traffic Control, Flight dispatcher and pilot during flight trials and see the impact on the actors.*

[2] *ecCodes is a project under development at ECMWF. Meteo-France has had early access on a development pre-release version because there is the intention to start a collaboration (to be finalized) between Meteo- France and ECMWF on the development of a tool to convert METAR and TAF messages to AvXML format .*

Therefore, for each kind of messages (METAR, TAF, SIGMET...), a new rule set must be written according to the regulations that should be applied to the message. The process of writing rules is not straightforward due to the fact that regulations are not written from a computer science point of view. That means that a good understanding of the regulations is needed to write good rules to successfully decode the messages.

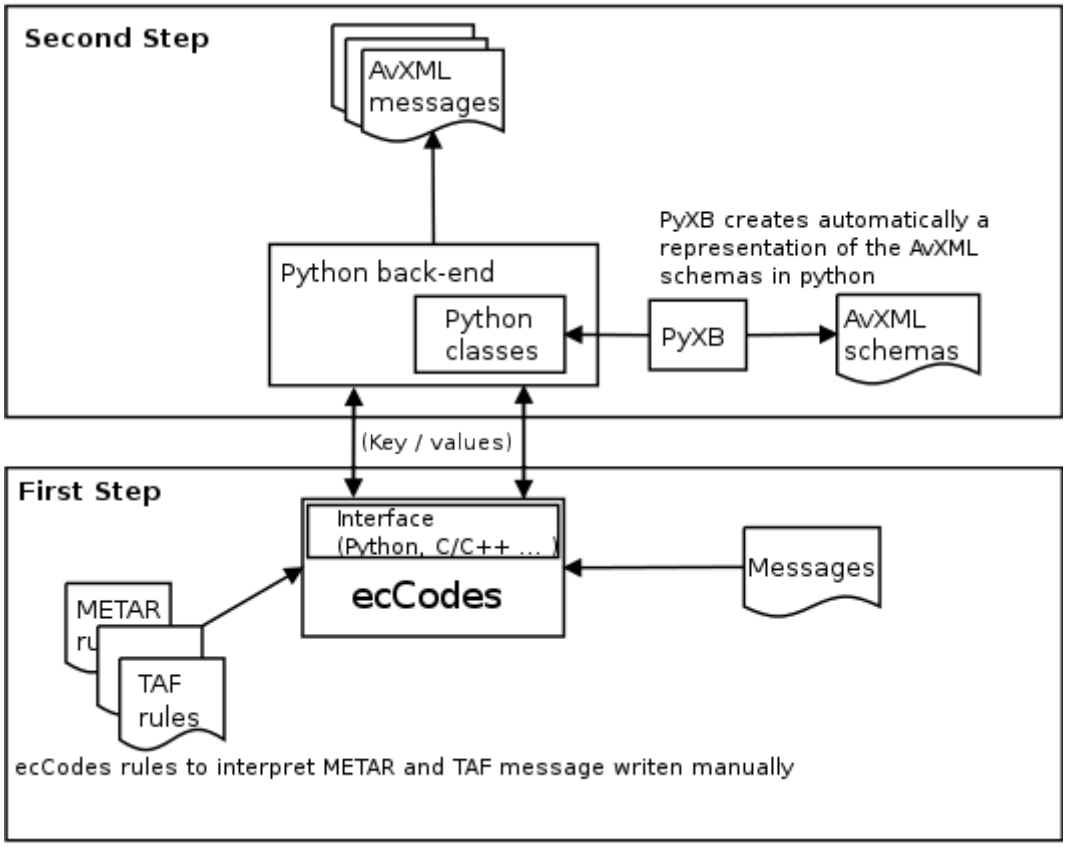

Figure 1: AvXML Converter Architecture

The second step deals with the ecCodes output and the AvXML model which can be seen as the back-end of the converter. A first back-end is written in Python making use of the Python XML Schema Binding (PyXB) [5] library. This library allows to create automatically a Python classes representation of the AvXML model. Therefore, using the ecCodes output (keys/values) and the AvXML Python classes, messages in AvXML format are yielded (see AvXML messages in Annex I).

An additional step is required to validate the messages against schematron [6] to verify that the messages are correctly built.

3. Work achieved at Meteo-France

A first version to convert METAR messages using the AvXML-RC1 was provided by ECMWF. This version does not fully work with AvXML-RC2 due to the changes performed between RC1 and RC2. Meteo-France modified the back-end to support AvXML-RC2.

In parallel, new versions of ecCodes rule set and back-end to decode TAF were developed. Although some work is still needed to improve the quality of the rules for both METAR and TAF messages.

4. Results

Only European METAR and TAF messages were used to check the performance, the quality of the converter and the pertinence of the AvXML-RC2 model. we have only focused on European METAR and TAF messages. Two samples for METAR and TAF messages were processed by the converter. The results are shown in the table below.

	Number of messages	Successfully converted messages
METAR	33362	18522
TAF	2848	1093

Table 1: Number of converted messages

As one can see, 55% of METAR messages are successfully converted whereas TAF messages conversion reaches only 40%.

Messages that are not successfully converted can be classified as:

- Conversion failure related to the messages themselves:
  - Not well-formed messages because of missing groups.
- Conversion failure related to the tool:
  - Unable to determine the airport name from the ICAO location indicator: which link to AIXM ?
  - Not decoded messages because ecCodes rules not yet enough well defined.
- Problems related to the model/schematron:
  - Constraints in AvXML Model.

After manual check and classification of the messages not successfully converted, a problem with the AvXML model was detected. When a METAR message contains a fully automated observation without human intervention, no mean in Runway visual Range must be specified. However the current AvXML model forces to introduce a mean. A cardinality for meanRVR should be specified in the AvXML model to fix the problem (this is already taken in consideration to the V1 version). Another problem was found on differences between ICAO and WMO regulation on missing groups for auto METAR.

	Number of messages	ecCodes runtime	Back-end runtime
METAR	33362	21 seconds	44 minutes
TAF	2848	11 second	2 minutes

Table 2: Performance

As shown in Table 2 ecCodes performs well. Nevertheless, PyXB seems not to be efficient. This may be due to the complexity of the AvXML schema.
If this performance issue is a show stopper, other solutions can be adopted for writing the back-end, like for instance, CodeSynthesis XSD [7] which is an open source written in C++ that generates classes that represents the AvXML schemas.

5. Conclusions

In this report we have studied different existing tools to convert from METAR and TAF into AvXML. We considered ECMWF converter as the best option because it provides a quick way to implement a METAR and TAF into AvXML converter and it gives us the possibility to write the back-end in different programming languages in function of the needed performance.

Meteo-France modified the converter to support AvXML-RC2 and convert TAF messages.

The results showed that a big portion of the messages tested cannot be converted because of missing information or because the converter is unable to determine the airport name from the ICAO location indicator. Some work is still need in METAR and TAF rules.

METAR and TAF decoder can only be written interpreting the regulations what it means that a long implementation time is need although ecCodes rules language makes the process quicker.

6. References

1. http://www.wmo.int/pages/prog/www/WIS/wiswiki/tiki-index.php?page=AvXML1.0RC2
2. http://metaf2xml.sourceforge.net/
3. http://sourceforge.net/projects/jweather/
4. http://www.ecmwf.int/
5. http://pyxb.sourceforge.net/
6. http://www.schematron.com/