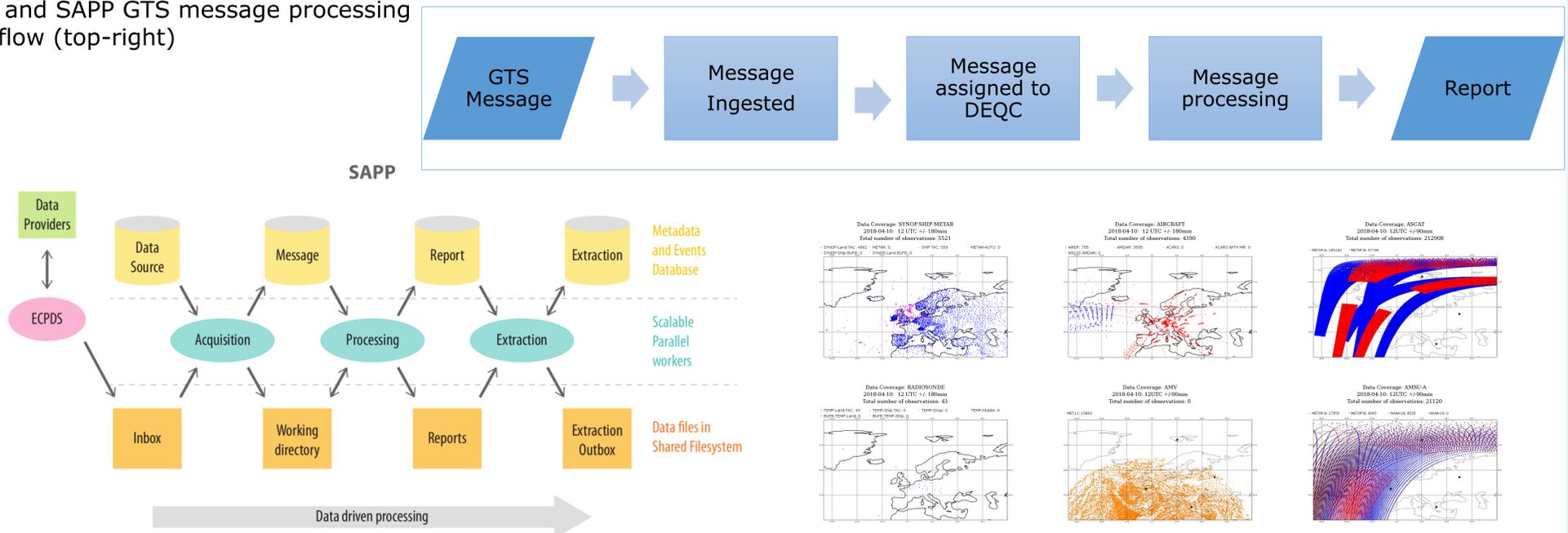


SAPP: Scalable Acquisition & Pre-Processing

Fig. 1: SAPP implementation at ECMWF (left) and SAPP GTS message processing data flow (top-right)



1. Introduction

- Scalable Acquisition and Pre-Processing (SAPP) is software developed by ECMWF to process GTS messages for use in NWP. SAPP was introduced into operations at ECMWF in 2014.
- We have tested SAPP as a possible replacement for our current operational pre-processing system which has been in use to process observational data for data assimilation in our NWP models since the 1990s.

2. Overview

- An overview of the SAPP system is shown in Fig. 1
- SAPP can acquire observations from multiple sources
- SAPP decodes multiple formats (for example, BUFR, GRIB, HDF, netCDF, ASCII)
- SAPP applies quality control algorithms to the process
- All data are converted to a consolidated format (BUFR/ODB2) before use in data assimilation

3. Installation

- ECMWF provided the test system as a virtual machine (SAPP-VM)
- Configuration and installation instructions are provided on ECMWF's SAPP wiki
- A working version of SAPP-VM can conceivably be installed and processing GTS data in a matter of hours.
- ECMWF provided useful information during this process.

4. Components

- Acquisition - The data acquisition is carried out by the Python script, **acq_scanner.py**. Data types are separated by GTS header.
- Processing - The process dispatcher is a Python script, **proc_dispatcher.py**, which reads the lists of data to be processed by SAPP and triggers the required decoders.
- DEQC - decoder and QC processors that produce the reports for NWP. Note: new converters can be tested outside the system and plugged in.
- Metadata - GTS message and NWP report metadata are managed by a **relational database**.

5. Features

- Scalable with **proc_dispatcher.py**, managing load balancing
- Fault tolerant. Unprocessed items are available for re-processing.
- Configuration of database via web interface
- Data on file system, metadata on database for better performance.
- eCodes used for routing and report metadata.
- Developed using Python - quicker prototyping, reduced amount of code, easier maintenance
- Suite of web-based monitoring tools

Web-based monitoring example - percentage of reports completed by observation type in past six hours

data	data	data	data	data	data
data	data	data	data	data	data
ACMR	egpr_gts	423	0	0	
AIRC	egpr_gts	1465	649	44.48	
AMDA	egpr_gts	1125	0	0	
AMDW	egpr_gts	1125	1003	100	
ARPI	egpr_gts	2561	2561	100	
AUTO	egpr_gts	28	0	0	
B002	egpr_gts	60	60	100	
B004	egpr_gts	423	0	0	
BSSH	egpr_gts	2725	2657	97.5	
BTKM	egpr_gts	43	41	95.34	

Fig. 2: Web-based monitoring example - percentage of reports completed by observation type in past six hours

6. Validation

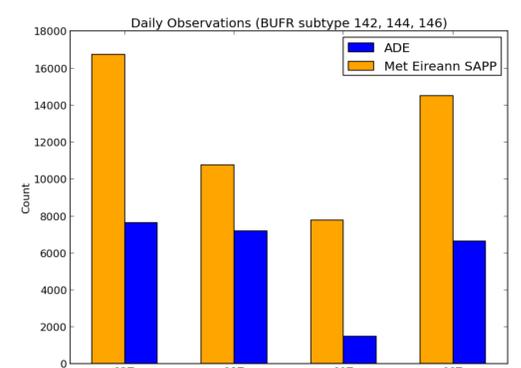


Fig. 3: Total processed observation count of aircraft reports (BUFR type 4) over a 24-hour period in December 2017 for SAPP-VM versus local software for the European Area 30-90N, -50-60W].

- SAPP-VM ran for six months without any system failures
- SAPP-VM produced many more reports for NWP when compared with output from local software (Fig. 3, Fig. 4)

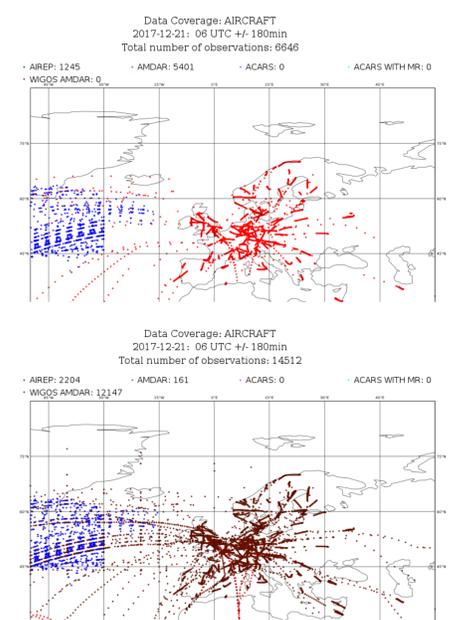


Fig. 4: Aircraft reports as produced locally [6646] (top) and by SAPP-VM [14512] (bottom) for 0600 UTC on December 21st 2017

- SAPP-VM is a viable alternative to local software used to process GTS messages for NWP
- SAPP-VM will be further developed for local use

- An internal NWP Note provides a full report on the evaluation of SAPP-VM in Met Éireann
- Contact: eoin.whelan@met.ie