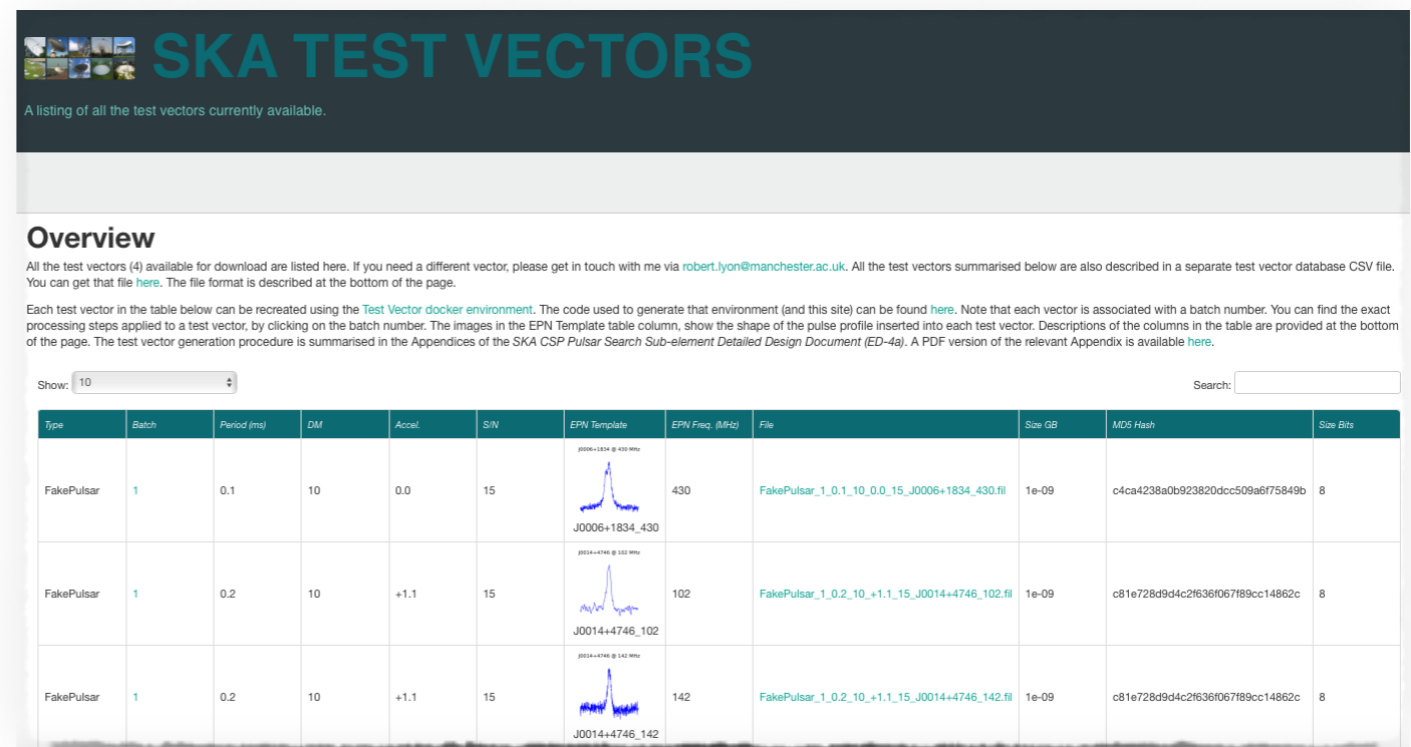


# Project: Gold Standard Test vectors

Important for,

- Testing our pulsar search algorithms.
- Studying our metadata/provenance requirements.
- Useful for networking tests (data isn't stored by SDP).
- A gold standard to evaluate against?
- Crucial for the SKA.
- Basic creation pipeline exists - consist of a docker container, python and web interfaces.
- Room for improvement!

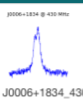



**SKA TEST VECTORS**  
A listing of all the test vectors currently available.

**Overview**  
All the test vectors (4) available for download are listed here. If you need a different vector, please get in touch with me via [robert.lyon@manchester.ac.uk](mailto:robert.lyon@manchester.ac.uk). All the test vectors summarised below are also described in a separate test vector database CSV file. You can get that file [here](#). The file format is described at the bottom of the page.

Each test vector in the table below can be recreated using the [Test Vector docker environment](#). The code used to generate that environment (and this site) can be found [here](#). Note that each vector is associated with a batch number. You can find the exact processing steps applied to a test vector, by clicking on the batch number. The images in the EPN Template table column, show the shape of the pulse profile inserted into each test vector. Descriptions of the columns in the table are provided at the bottom of the page. The test vector generation procedure is summarised in the Appendices of the *SKA CSP Pulsar Search Sub-element Detailed Design Document (ED-4a)*. A PDF version of the relevant Appendix is available [here](#).

Show: 10 Search:

Type	Batch	Period (ms)	DM	Accel	S/N	EPN Template	EPN Freq. (MHz)	File	Size GB	MD5 Hash	Size Bits
FakePulsar	1	0.1	10	0.0	15	 J0006+1834_430	430	FakePulsar_1_0.1_10_0.0_15_J0006+1834_430.fl	1e-09	c4ca4238a0b923820ccc509a6f75849b	8
FakePulsar	1	0.2	10	+1.1	15	 J0014+4746_102	102	FakePulsar_1_0.2_10_+1.1_15_J0014+4746_102.fl	1e-09	c81e728d9d4c2f636f067f89cc14862c	8
FakePulsar	1	0.2	10	+1.1	15	 J0014+4746_142	142	FakePulsar_1_0.2_10_+1.1_15_J0014+4746_142.fl	1e-09	c81e728d9d4c2f636f067f89cc14862c	8

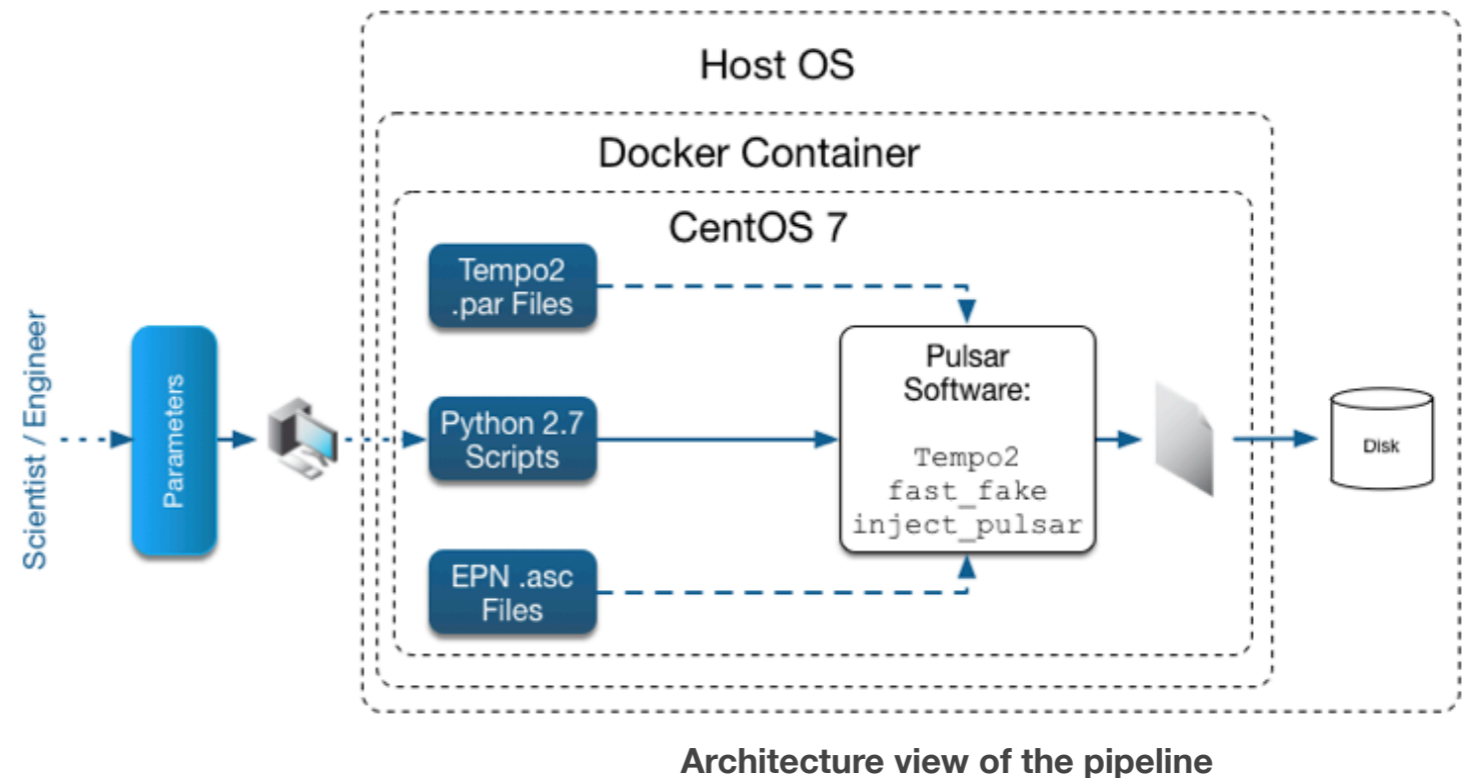
Current web interface

Pipeline code: DOI:10.5281/zenodo.1165435

Current interface: <https://github.com/scienceguyrob/SKA-TestVectorGenerationPipelineInterface>

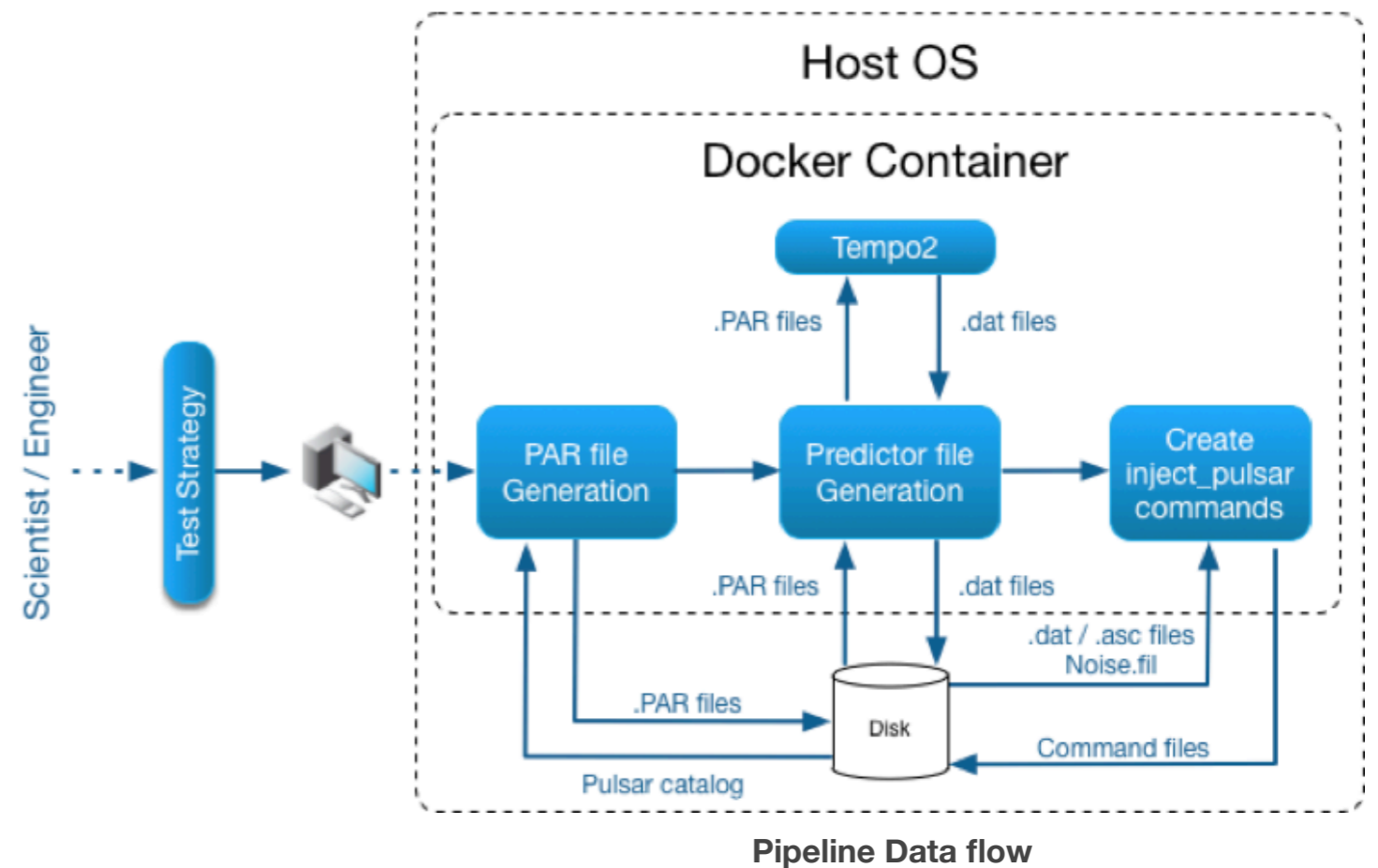
# Pipeline Overview

- Pipeline contained within a docker container.
- It holds multiple astronomy software packages.
- But such software is complex, and requires many input parameters.
- The pipeline simplifies their execution, and makes the creation of test vectors reproducible.
- Achieve via simple Python code that bolts all the software together.
- Can be executed on any machine that can install docker.



# Basic steps

1. Create a filterbank noise file using `fast_fake`. Specify the “observation” length (s), sampling interval (microsec), channel bandwidth (KHz), overall bandwidth (MHz), channels etc.
2. Choose the signal to be injected:
  - Pulse shape (from EPN database).
  - Pulsar details (DM, pulse period etc) from the ATNF catalog.
3. Inject the signal using the `inject_pulsar` tool.
4. Save the resulting filter bank file to the host for analysis.



# Goals

**Modest expectations for now** - the pipeline works but can be improved.

- Improve the containerisation of the pipeline.
- Improve the pipeline interface.
- Determine what use cases people have so it can be useful for others (not just for pulsar search).

**Future goals** - cool functionality, e.g.

- Request a test vector via an online interface.
- Add single-pulse injection code.
- Support the addition of new modules (e.g. RFI injection, non-stationary noise, pulsar timing support).
- Lots more!

**Lets build some test vectors for reproducible science, software analysis, and software testing!**

 @scienceguyrob

 robert.lyon@manchester.ac.uk