



THE DELAY DOPPLER PROCESSOR

Mark Pattle

mark.pattle@isardsat.co.uk



- The DeDop Processor
 - Introduction
 - Software Architecture
 - The Processing Chain
 - Standard Workflow
- DeDop Studio
 - The DeDop GUI
- DeDop CLI
 - Creating a Workspace
 - Processing a File
 - Analysing the Results



- DeDop
 - Dedop is a Fully Adaptable and Configurable Delay-Doppler Processor
 - L1A-to-L1B processor
 - Reads Sentinel-3 L1A files
 - Produces L1B and L1B-S output
 - Open Source
 - MIT Licence
 - <http://www.github.com/DeDop/>
 - Developed in Python 3



```
Anaconda Prompt

ESA DeDop Shell, version 0.5.3

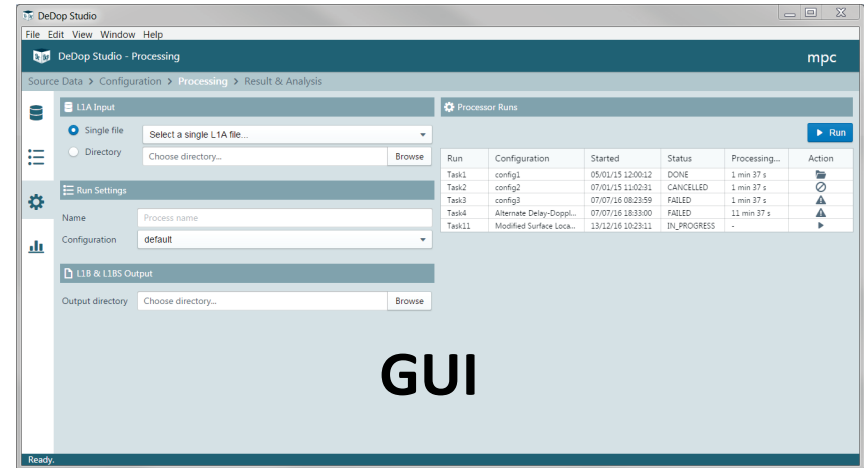
positional arguments:
  COMMAND      One of the following commands. Type "COMMAND -h" to get
                command-specific help.

workspace (w)
  Manage DeDop workspaces.
config (c)     Manage DeDop DDP configurations.
input (i)     Manage L1A inputs.
output (o)    Manage and analyse L1B outputs.
run (r)       Run the Delay Doppler Processor (DDP).
status (s)    Print DeDop status information.
notebook      Open a Jupyter Notebook for DeDop.
man           Open DeDop user manual in browser window.
copyright     Print DeDop copyright information.
license       Print DeDop license information.

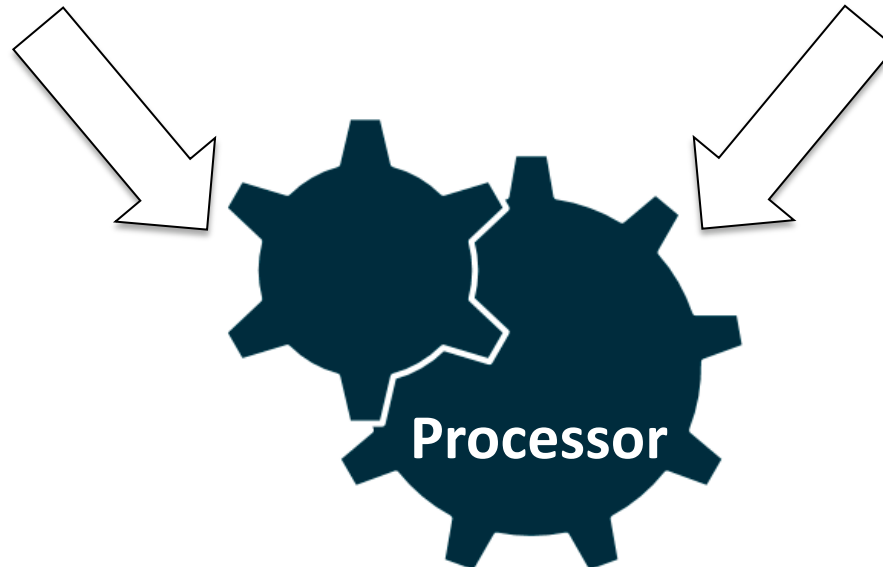
optional arguments:
  -h, --help            show this help message and exit
  --version             show program's version number and exit
  -e, --errors          on error, print full Python stack trace
  --new-conf            write a new DeDop tools configuration file and exit

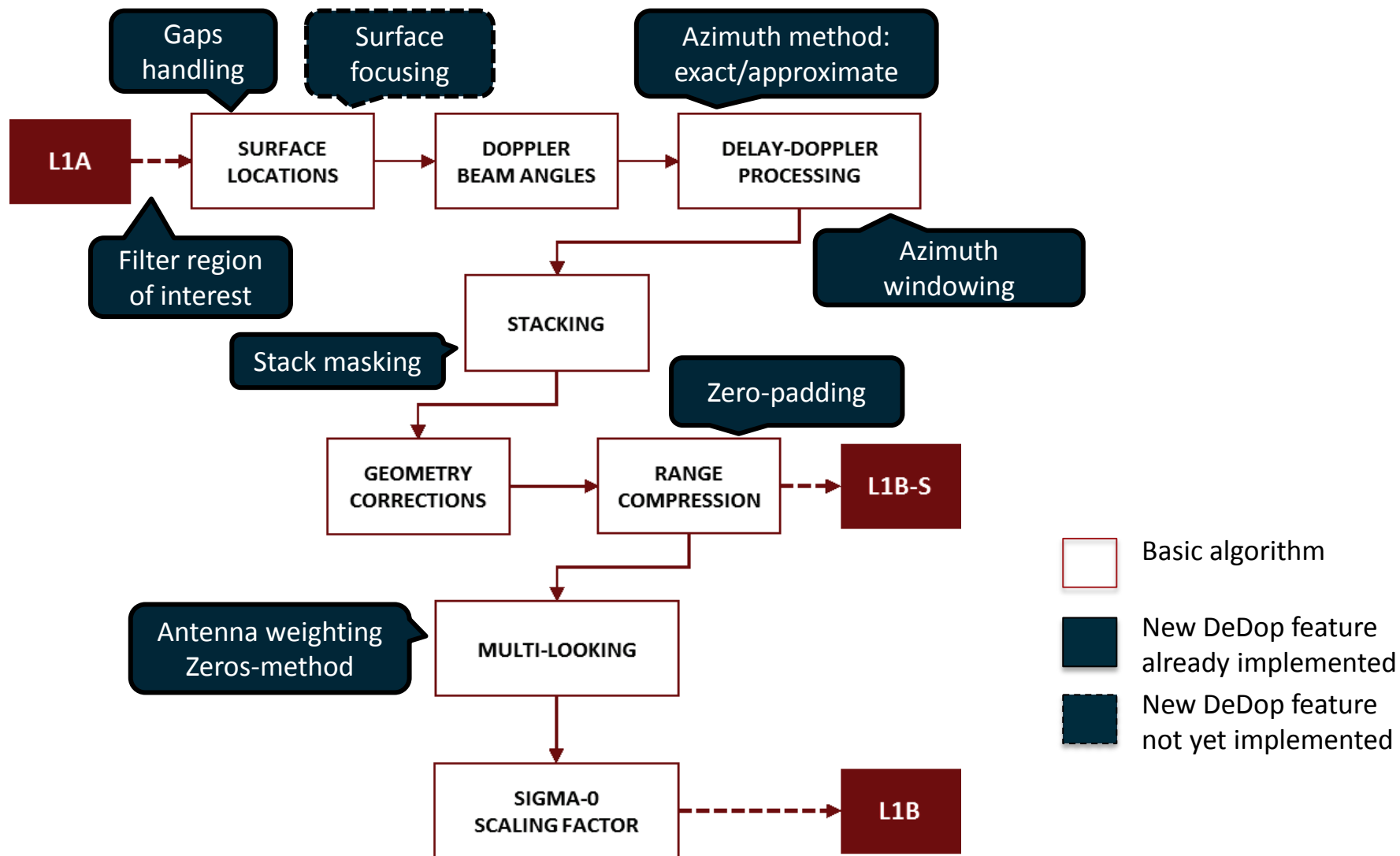
(C:\Users\Mark\Miniconda3) C:\Users\Mark>
```

CLI



GUI







Set Up

- Select Input files
- Set Configuration

Process Data

- Produce L1B and L1B-S Files

Analyse Results

- Launch Jupyter Notebook