

# **TELLIE PCA: Processing automation**

Michal Rigan

February 2023

This document describes what the TELLIE PCA automation does, why, and how.

# 1 Content

- Why automation
- Overview
- Data-taking
  - Validation #1
- Data-processing
  - PCA table generation
    - \* Beamspot fit
    - \* Direction fit
    - \* Angular systematic
    - \* Injection time fit
  - Validation #2
  - PCA constants
  - Benchmarking
  - Monitoring
- Code
- Deploying
- Running
- Todos
- Other documentation

## 2 Why automation

The process of extracting and validating PCA constants from TELLIE data is complex. This piece of software was developed to streamline the process of obtaining the PCA constants, at reasonably high speed. Additionally, it was designed to:

- be independent of the method used for Data-taking
- be modular, easily modifiable and configurable
- require minimal human input
- provide monitoring
- be mostly standalone

## 3 Overview

The TELLIE PCA Automation overview is shown in Figure 1. There are two main parts: TELLIE Data-taking and Data-processing.

Data-taking is done independently of the processing (as the exact method was not yet finalised before developing processing). More information can be found in [TELLIE Data-taking automation document](#). It should be noted that **Validation #1** is taken care of by Data-processing.

Data-processing is everything that is done with TELLIE PCA data once it is stored. This includes performing checks on the data, making fits required for further processing, generating tables (both local and online), extracting PCA constants, benchmarking these constants, and a suite of monitoring for these steps. These will be described below.

## 4 Data-taking

### 4.1 Validation #1

As mentioned above, even though **Validation #1** is logistically part of Data-taking, it is performed by Data-processing, and is also independent of

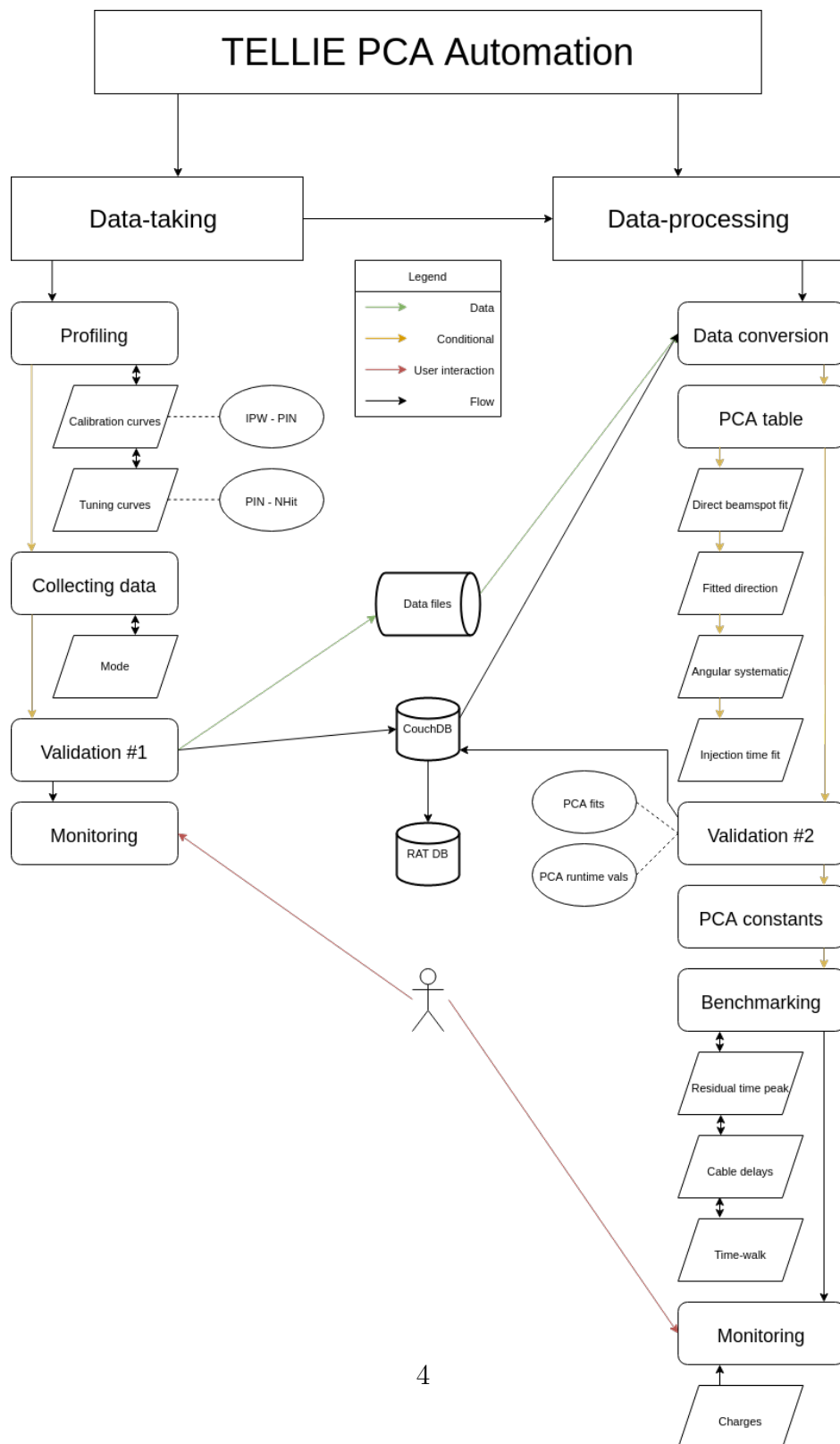


Figure 1: Overview of the TELLIE PCA Automation.

the method the data was taken.

**Goal:** Validate that the data is of required quality for PCA.

## 5 Data-processing

### 5.1 PCA table generation

There are several corrections that need to be fitted for, which are later used for the extraction of PCA constants. The fits need to happen in succession, as the output of one feeds into the next. Between steps, these are stored as text files. After all fits are made, a local table is produced, combining the corrections. This table is loaded by the PCA Processor.

**Goal:** Make fits, obtain corrections required for the extraction of PCA constants. Produce final PCA table.

#### 5.1.1 Fit: beamspot

#### 5.1.2 Fit: direction

#### 5.1.3 Fit: angular systematic

#### 5.1.4 Fit: Injection time

### 5.2 Validation #2

**Goal:** Check and confirm that the fits are sensible.

### 5.3 PCA constants

**Goal:** Run the PCA Processor that extract the pca constants (both timing and charge).

### 5.4 Benchmarking

**Goal:** Compare the set of constants against the closest (previous) set. Useful to see overall stability and for highlighting outliers.

## **5.5 Monitoring**

**Goal:** Provide monitoring of each step of the chain. Also compares fibres between datasets.

## **6 Deploying**

## **7 Running**

## **8 ToDos**

## **9 Other documentation**