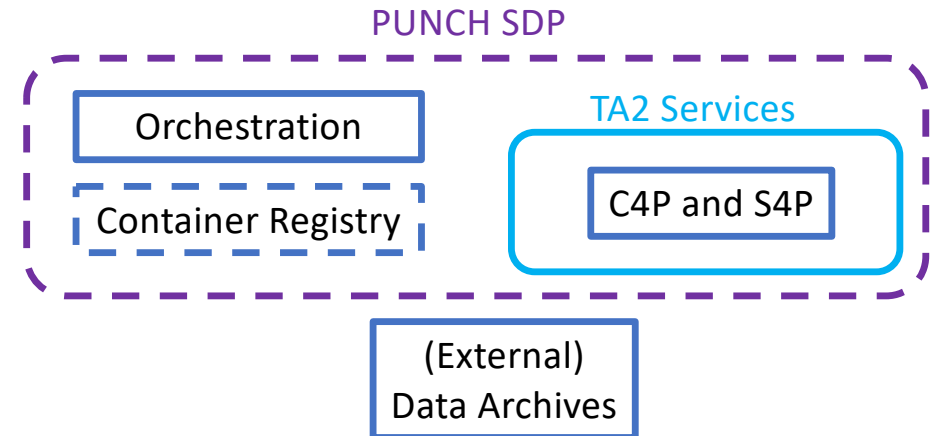


Workflows: a TA3 Perspective

- One focus of PUNCH is to establish a framework to support analysis of data
 - There exist use cases from all 4 communities
- TA3WP4 has task of establishing and testing workflow templates
 - Goal: enable non-PUNCH use of our framework
- TA2 prototype services Compute4PUNCH (C4P) and Storage4PUNCH (S4P) are available
- TA6 AAI services integrated into C4P, S4P, elsewhere
- TA4 concept for Platform/Portal maturing
- TA3 Workflow prototypes now a possibility
- Expect these tests to highlight known issues within TA2 services and perhaps to reveal unexpected problems through „naive“ TA3 testing
- Expect TA4 Platform/Portal to come along later
 - And perhaps be informed from our Workflow prototyping experiences

Workflow concept

- Automated execution of complex tasks on diverse computing resources drawing upon diverse data resources
- Many use cases can be addressed by tackling examples from the five Use Case classes
 - Cross-community use cases ideal
- Start with simple prototypes and build from there

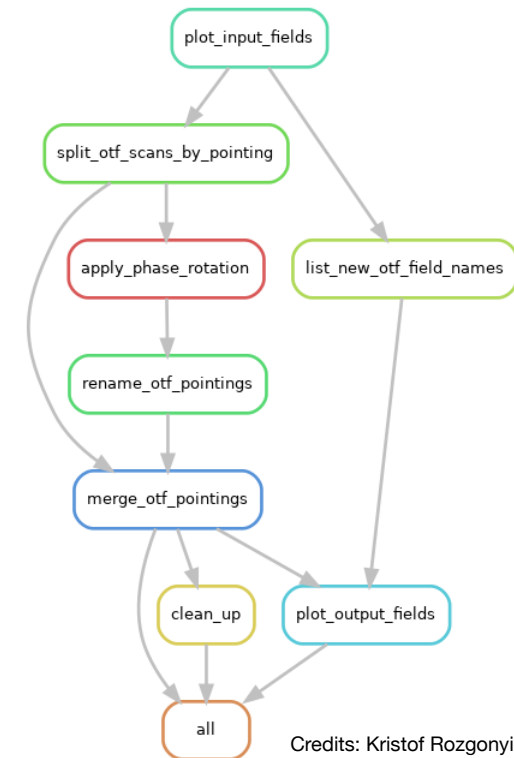


1. **Orchestration**
 - Constructs a plan for the analysis
 - Finds the relevant input data in Data Archives and stages them to S4P resources
 - Constructs the jobs and submits to C4P resources, accounting for dependencies, needed resources, employing appropriate levels of parallelism
2. **(External) Data Archives**
 - Resting place of input and output data
 - (Could be internal/external to PUNCH, could contain public and proprietary data)
3. **C4P and S4P**
 - Compute and storage resources
 - Supporting automated access (AAI token solution instead of GridFTP)
 - Supporting job queue (SLURM)

Initial workflow proposal

- Simple radio imaging workflow that is being developed for the MeerKAT Observatory
 - Kristof Rozgonyi: OTF imaging
 - Nicola Malavasi: PUNCH implementation
- Steps in initial phase
 - (Review the Tutorials available from TA2)
 - Locate needed input data on S4P
 - Build containers and make them available on C4P
 - Submit (simple) jobs to C4P to transform input data
 - Recover output data from S4P
- Later steps:
 - Automate staging of input data/ recovery of output data
 - Execute more complex analysis plan with many steps, different levels of parallelization that produces high quality images
 - Deploy jobs across multiple C4P resources (TA2 Overlay Batch System)
 - Build the test within the Platform/Portal framework
 - Container registry, etc

Initial Data Transformation Workflow



See Nicola Malavasi presentation this morning