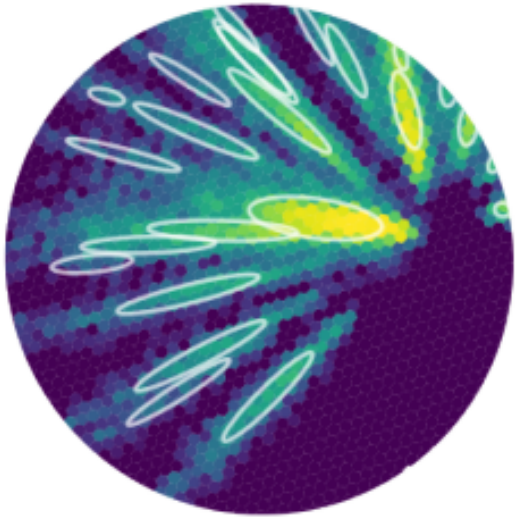


ctapipe

Package available on github

https://github.com/cta-observatory/ctapipe

cta cherenkov telescope array

ctapipe - status and prospects

Francesco Longo — material from CTA consortium talk 1 and talk 2 by K.Kosack and from CTAO computing call slides by S.Schlenstedt

What is ctapipe?

Framework for processing data from Cherenkov Telescopes

- **EventSources** for reading data from many data inputs (plug-ins easy to write for any you need!)
- common **Instrument data model** supporting all CTA telescopes and cameras + MAGIC, HESS, VERITAS, etc
- Container-based internal **Event data model**
- event-wise **Table-oriented output** in HDF5 format with automatic schema generation, output readable by pytables, pandas, etc.
- **Application framework** of Components and Tools providing common configuration (command-line and config file) and logging system
- **Visualization** of Cherenkov Cameras and Arrays
- Library of **common event-level algorithms** for image processing, reconstruction, etc.

Recent changes (v0.8)

Main points:

- ctapipe evolving to be more functional and streamlined, fixing architectural problems by refactoring
- final goal of version 1.0 to provide stable, maintainable, configurable framework and tools for doing each stage of data processing for CTA DPPS DataPipe, CalibPipe, QualPipe
- In the meanwhile, supporting prototyping efforts like:
  - **science pipelines**: *protopipe*, *pyIRF*, *cta-benchmarks*
  - **telescope testbenches** (*lstchain*, *nectarchain*, *target*, hopefully more...)
  - **algorithm exploration testbenches** (protopipe again, Deep Learning, etc)

Recent changes (v0.8)

1 Algorithms

1.1 Calibration

- Update documentation for CameraCalibrator and ImageExtractor (#1283)
- Fixes for CameraCalibrator and `extract_around_peak` (#1172) @watsonjj
- Add application of DL1 charge calibration to CameraCalibrator (#1160) @watsonjj
- Move gain selection from CameraCalibrator to EventSource (#1167) @watsonjj
- ensure gain selection returns uint8, not 64-bit int (#1125) @kosack

1.2 Image

- Cleaning for biggest cluster (#1131) @HealthyPear
- Added tailcuts data volume reducer (#1121) @Hckjs (#1280) @MaxNoe
- Fix bug in ImageExtractor and WaveformModel concerning waveform sampling rate (#1276) @watsonjj
- Waveform toy model (#1244) @watsonjj
- configurable image cleaners (#1201) @kosack
- Fix unit error in SkewedGaussian toymodel (#1120) @MaxNoe
- Fix dtype of island labels (#1134) @MaxNoe
- `timing_parameters` cleaning mask (#1119) @vuillaut

1.2.1 ImageExtractors

- Make ImageExtractor TelescopeComponent, remove duplicated code
- 2-Pass waveform integration à-la-CTA/MARS (#1215) @HealthyPear
- Improvements to ImageExtractor wrt integration$_{\text{correction}}$ and sampling rate (#1242) @watsonjj
- Fix hillas width 0, fixes #772 (#1240) @MaxNoe
- Improvement of `integration_correction` testing, implementation, and docstring (#1233) @watsonjj
- Remove `extract_pulse_time_around_peak` (#1234) @watsonjj
- Update ImageExtractors to utilise TelescopeParameters (#1155) @watsonjj
- Combine sample loops (#1146) @watsonjj
- Improve `extract_pulse_time_around_peak` to ignore negative samples (#1143) @watsonjj
- Remove 3-dimension extractors (#1144) @watsonjj
- Make subarray a required argument to CameraCalibrator and ImageExtractor (#1228) @watsonjj

1.2.2 Muons

- Muon refactoring (#1253) @MaxNoe
- Remove unused and untested muon fitting code (#1254) @MaxNoe
- Calculate ring containment analytically (#1256) @MaxNoe
- Calculate muon image parameters (#1316) @MaxNoe
- Muon intensity improvements (#1261) @MaxNoe
- Various bug fixes and improvements in muon ring analysis (#1245) @moralejo @watsonjj
- add Taubin fit (#1154) @momorning

1.3 Coordinates

- Rename coordinates of NominalFrame and TelescopeFrame, fixes #1260 (#1299) @MaxNoe
- Add directions to GroundFrame docstring as described in #1183 (#1185) @nbiederbeck

2 Framework

2.1 Core

- Replace more usages of Unicode with Path (#1319) @MaxNoe
- Path improvements (#1284) @MaxNoe
- Add a Selector class for keeping track of quality cuts (#1207) @kosack
- ensure command-line args have higher precedence than config options (#1168) @kosack
- Convert config file argument to path type, fixes #1196 (#1236) @MaxNoe
- remove deprecation warning: replace log.warn –> log.warning (#1204) @dneise
- Fix/core tool (#1124) @kosack
- add helper function, might be nice for parametrized tests (#1203) @dneise

2.1.1 TelescopeParameter Trait

- Fix TelescopeParameter config, bug #1216 (#1218) @kosack
- Extra TelescopeParameter use cases (#1175) @watsonjj
- Return global value from TelescopeParameterLookup when passed None (#1173) @watsonjj
- Scalar default values in TelescopeParameter (#1171) @watsonjj
- Add resolving to the TelescopeParameter (#1158) @watsonjj
- Remove inheritance on list for TelescopeParameterList (#1162) @watsonjj
- fixed bug in TelescopeParameterResolver (#1156) @kosack
- TelescopeParameter Trait (#1129) @kosack
- Add BoolTelescopeParameter, expose all options of TailCutsCleaner (#1281) @MaxNoe

2.2 Input EventSources

- remove HESSIOEventSource (#1304) @kosack
- Add more properties to EventSource, implement for SimTelEventSource, fixes #1286 (#1313) @MaxNoe
- Toy source (#1298) @MaxNoe
- Drop dead code in simtel event source (#1288) @MaxNoe
- add option to use effective focal length, fixes #1223 (#1262) @kosack
- Enable access to subarray information from file before event loop (#1157) @watsonjj

2.3 Data Model (containers)

- Container Improvements and Cleanup for DL1 (#1301) @kosack
- Fix writing of default image parameters container (#1311) @MaxNoe
- Add missing config, parent to hdfwriter (#1312) @MaxNoe
- Use pointing container in examples and docs (#1296) @MaxNoe
- Remove sst1m specific containers (#1292) @MaxNoe
- CTA Reference Metadata handling (#1221) @kosack
- Move containers from io to base (#1267) @MaxNoe
- Replace muon container defaults with nan, remove redundant fields (#1248) @MaxNoe
- Add Monitoring container to MC data (#1115) @FrancaCassol
- Pointing container (#1141) @vuillaut
- Improvements to Container classes (#1123) @kosack

2.4 Output (TableWriter)

- Hdf5 column order (#1308) @MaxNoe
- Add array pointing to pointing, use .tel map (#1303) @MaxNoe
- Fix parent propagation in hdf writer (#1320) @MaxNoe
- Store filters and use for new datasets in HDFWriter (#1285) @MaxNoe
- add support of float16 in PYTABLES$_{\text{TYPEMAP}}$ (#1273) @vuillaut

2.5 Instrument

- Remove deprecated instrument from event container (#1294) @MaxNoe
- Fix calculation of pixel area guessing (#1290) @MaxNoe
- Created CameraDescription refactor CameraGeometry (#1241) @watsonjj
- Move pulse shape attributes from mc container to CameraGeometry (#1227) @watsonjj
- Add guessing MAGIC telescope (#1206) @adonini
- Add `sampling_rate` to instruments description (#1142) @watsonjj

2.6 Visualization

- Do not mutate geometry in camera display, fixes #1190 (#1237) @MaxNoe
- Fix bokeh viewer for new bokeh version (#1232) @MaxNoe
- Fix array display for mpl 3.2, fixes #1229 (#1231) @MaxNoe
- colorbar default axes as CameraDisplay.axes (#1193) @vuillaut

2.7 Tools

- Fix input url handling in display$_{\text{dl1}}$ (#1251) @MaxNoe

2.8 Cleanup

- Remove empty or misleading files (#1314) @MaxNoe
- Make arguments to `enum_trait` consistent, rename to `create_class_enum_trait`, fixes #1282 (#1306) @MaxNoe
- Remove old, untested plotting code (#1297) @MaxNoe
- Update travis badge to new travis link (#1291) @MaxNoe
- Ignore unmentioned references (#1269) @MaxNoe
- Replace np.power with power operator, fixes #1028 (#1239) @MaxNoe
- fix #1217 by removing unused _hyperbinning() function (#1219) @kosack
- Fixes for astropy 4.x (#1212) @kosack
- Use `count_nonzero` instead of sum for bool arrays (#1133) @MaxNoe
- Use astropys own version of `quantity_approx` (#1128) @MaxNoe
- a few small random cleanups (#1126) @kosack

3 Testing / Packaging / Docs:

- Update tests with new ctapipe-extra, remove hessio remnants (#1318) @MaxNoe
- Fix tests for Tools, fixes #1214 (#1027) @dneise
- Test with 3.8 (#1174) @MaxNoe
- fix codacy badge (use cta-observatory org) (#1250) @kosack
- Add pyproject toml (#1246) @MaxNoe
- bump astropy version requirement to allow v4.0 (#1213) @kosack
- Pypi upload (#1208) @MaxNoe
- Create CODEOWNERS (#1151) @watsonjj
- fix typo in contributors for release drafter (#1189) @vuillaut
- Update eventio version in setup.py (#1169) @watsonjj
- Add instructions for Slack (#1152) @watsonjj
- increase nbsphinx cell timeout (#1153) @kosack
- added DOI for release v0.7.0 (#1150) @kosack
- Add provenance.log to .gitignore (#1145) @watsonjj
- Also test ctapipe with a pure pip installation (no conda) (#1137) @MaxNoe
- Ci warnings (#1136) @MaxNoe

A simple loop

```
from ctapipe.calib import CameraCalibrator
from ctapipe.io import event_source, HDF5TableWriter

source = event_source("gammas.simtel.gz")
calibrate = CameraCalibrator(subarray=source.subarray)

with HDF5TableWriter("output.h5") as writer:

    for event in source:
        calibrate(event)

        for tel_id in event.dl1.tel:
            writer.write(f"tel_{tel_id}", event.dl1.tel[tel_id])

        # e.g. calibrated image is event.dl1.tel[x].image
```

ctapipe stage1

Standard tool to generate DL1 data from R0, R1, or DL0 input, with many options

```
ctapipe-stage1-process --help

...

Examples
--------

    To process data with all default values:

    > ctapipe-stage1-process --input events.simtel.gz
        --output events.dl1.h5 --progress

       Or use an external configuration file, where you can specify
       all options:

    > ctapipe-stage1-process --config stage1_config.json --progress

The config file should be in JSON or python format (see traitlets docs).
```

Configuration in JSON or Python formats:

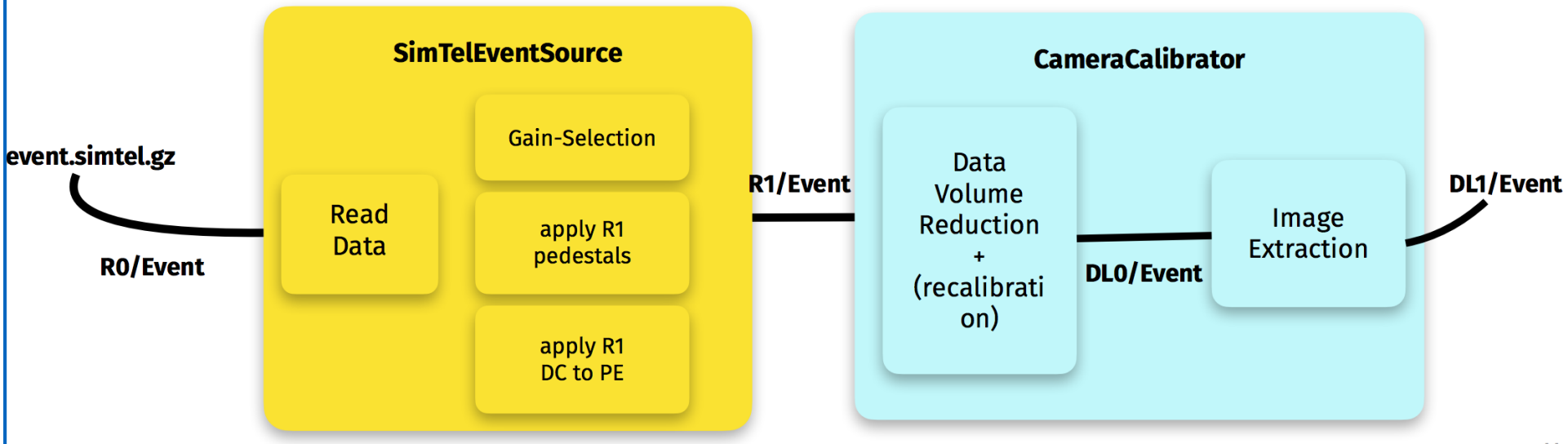
```
{
    "Stage1ProcessorTool": {
        "config_file": "",
        "output_path": "events.dl1.h5",
        "overwrite": true,
        "write_images": true,
        "image_extractor_type": "NeighborPeakWindowSum",
        "image_cleaner_type": "TailcutsImageCleaner"
    },
    "TailcutsImageCleaner": {
        "boundary_threshold_pe": [
            ["type", "*", 5.0],
            ["type", "LST*", 3.0],
            ["type", "MST*", 4.0]
        ],
        "min_picture_neighbors":[
            ["type", "*", 2]
        ],
        "picture_threshold_pe":  [
              ["type", "*", 10.0],
              ["type", "LST_LST_LSTCam", 6.0],
              ["type", "MST_MST_NectarCam", 6.0]
        ]
    },
    "ImageQualityQuery": {
        "quality_criteria": [
            ["enough_pixels", "lambda im: np.count_nonzero(im) > 2"
            ["enough_charge", "lambda im: im.sum() > 50"]
        ]
    }
}
```

Algorithms

Camera Calibration

Large refactoring (but only some small changes for users)

The scheme now (more similar to real data):



11

Algorithms

Image: Image Extraction

waveforms → images of Charge + Peak-time

Refactoring:

- less complex hierarchy
- algorithms now in simple functions
- assume waveforms are already gain-selected
- proper handling of sampling rates and pulse corrections
- speed improvements

New Methods:

- TwoPassWindowSum
  - originally from VERITAS , variant from CTA-MARS
  - uses time gradient to predict integration windows

Algorithms

Image: Muons

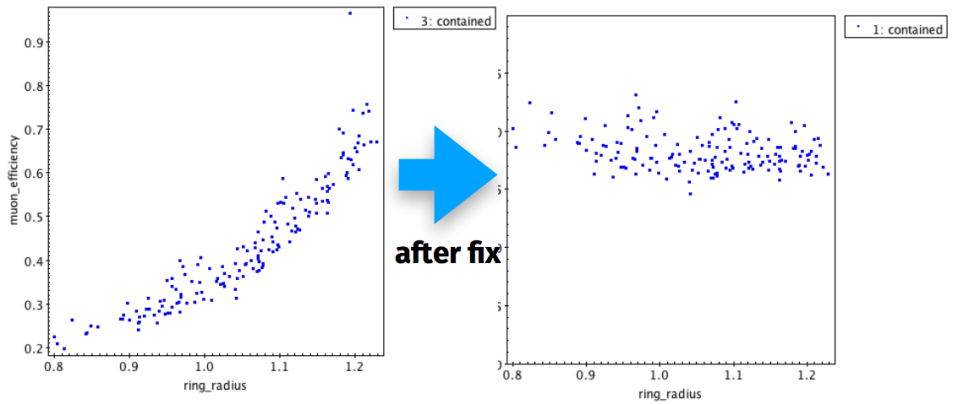
Massive refactoring/cleanup of the Muon parameters computations and ring detection

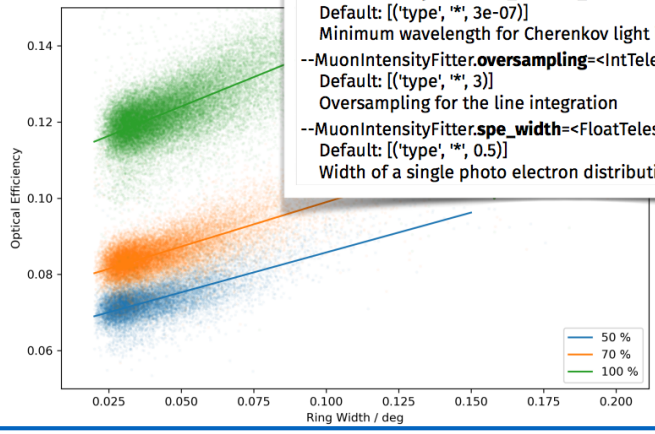
- **MuonRingFitter** and **MuonIntensityFitter** Components
- **ctapipe-reconstruct-muons** tool, with all config options and HDF5 output

Added Taubin fit method

Better naming and cleanup of muon parameter outputs

Fixed/improved calculation of muon efficiency





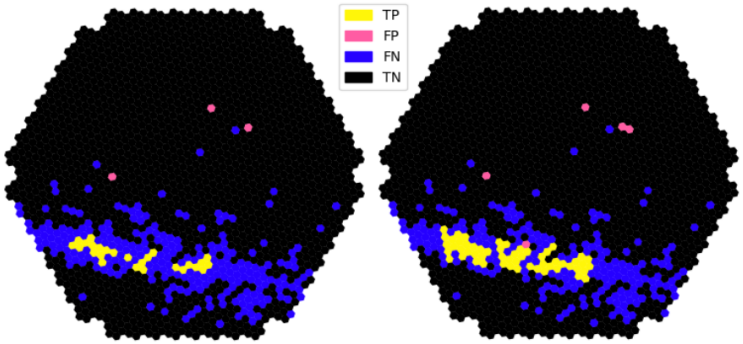

```
MuonRingFitter options
----------------------
--MuonRingFitter.fit_method=<CaselessStrEnum>
    Default: 'kundu_chaudhuri'
    Choices: ['kundu_chaudhuri', 'taubin']

MuonIntensityFitter options
---------------------------
--MuonIntensityFitter.hole_radius_m=<FloatTelescopeParameter>
    Default: [('type', 'LST_*', 0.308), ('type', 'MST_*', 0.244), ('type',...
    Hole radius of the reflector in m
--MuonIntensityFitter.max_lambda_m=<FloatTelescopeParameter>
    Default: [('type', '*', 6e-07)]
    Minimum wavelength for Cherenkov light in m
--MuonIntensityFitter.min_lambda_m=<FloatTelescopeParameter>
    Default: [('type', '*', 3e-07)]
    Minimum wavelength for Cherenkov light in m
--MuonIntensityFitter.oversampling=<IntTelescopeParameter>
    Default: [('type', '*', 3)]
    Oversampling for the line integration
--MuonIntensityFitter.spe_width=<FloatTelescopeParameter>
    Default: [('type', '*', 0.5)]
    Width of a single photo electron distribution
```

Algorithms

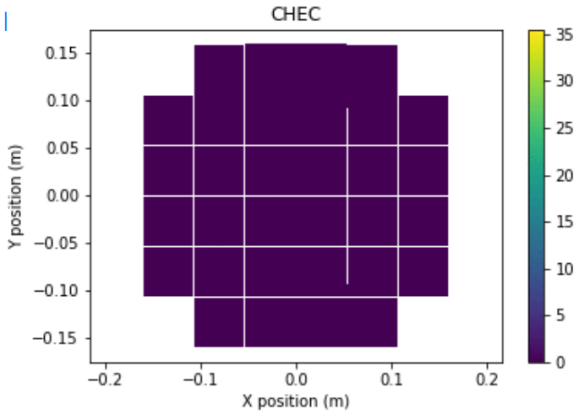
Image: General

New Features:

- Configurable **ImageCleaner** components. (e.g. TailCutsImageCleaner)
- Waveform **Toy Model** (uses reference pulse + toy shower model)
- **TailCutsDataVolumeReducer**
- **Cluster detection** and **Biggest-Cluster cleaning**
- improved **timing parameters** computatio



Tailcuts DVR



Toy model waveform generator

Algorithms

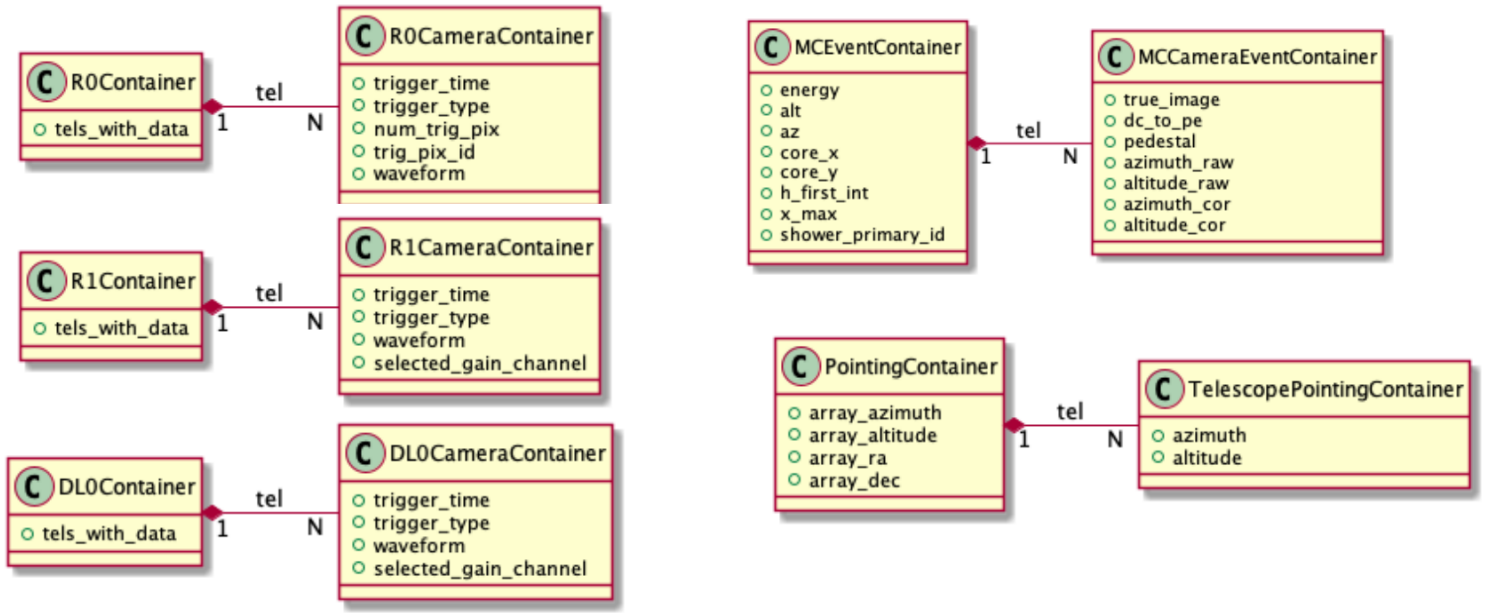
Coordinates

Renaming:

- TelescopeFrame and NominalFrame coordinates are now more correctly named, and closer to what is used in e.g. GammaPy
  - delta_az → fov_lon
  - delta_alt → fov_lat
- Note these are angular coordinates defined on a sphere where 0,0 is the center of the Telescope or Array FOV

Framework

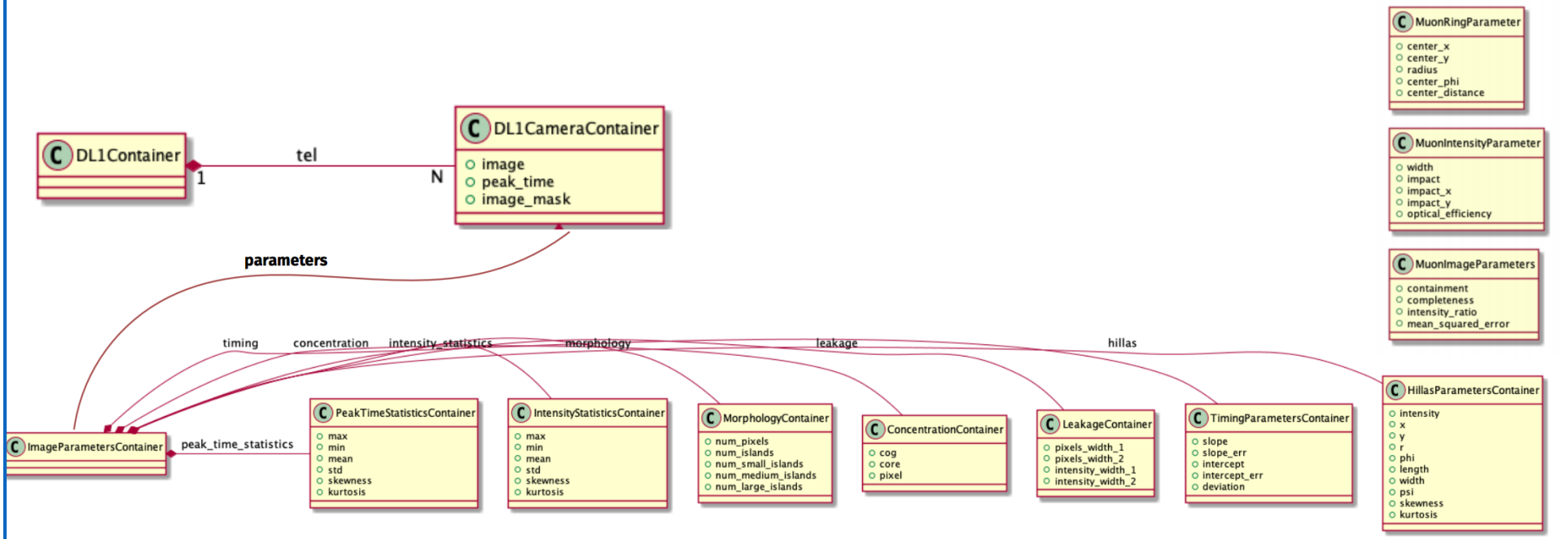
Data Model (containers)



- Lots of **refactoring** of the event structure and field names (requires changes to scripts!)
- Addition of **PointingContainer**
- Addition of CTA **Reference Metadata Header handling**
- **module move**: ctapipe.io.containers → ctapipe.containers (to break circular deps)
- Now can have **n_dims** and **dtype** attributes and **verify**() method to Fields (to enforce data model, optionally)

Framework

Addition of DL1 parameters

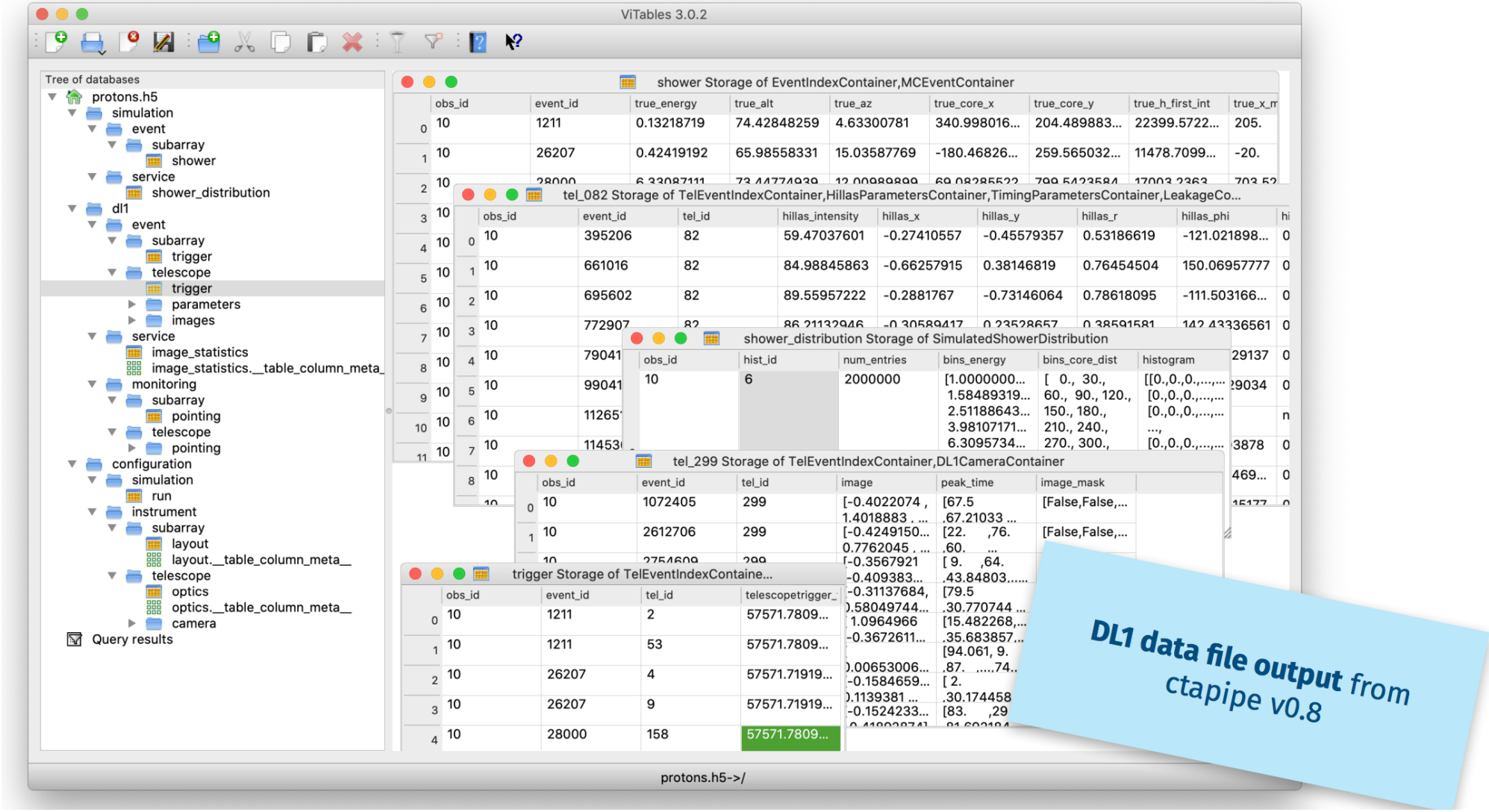


Framework

Instrument Description

- **`event.inst`** removed from event container (not event-wise info!)
  - **accessed instead** from EventSource: `source.subarray`
  - **no need to load an event** to see the instrumental information!
  - Fix calculation of pixel area guessing (#1290) @MaxNoe
  - external EventSources might need update to support this (not difficult), most already updated
- new **CameraDescription** class that contains:
  - **CameraGeometry** (as before)
  - **CameraReadout** (defines camera-hardware specific info, like pulse shape and sampling rate)
  - API change:
    `subarray.tel[i].camera` → **`subarray.tel[i].camera.geometry`**
- Better support for **real MAGIC telescopes** (vs Simulated)

Framework


ViTables 3.0.2
Tree of databases
protons.h5
simulation
event
subarray
shower
service
shower_distribution
dl1
event
subarray
trigger
telescope
trigger
parameters
images
service
image_statistics
image_statistics.__table_column_meta_
monitoring
subarray
pointing
telescope
pointing
configuration
simulation
run
instrument
subarray
layout
layout.__table_column_meta__
telescope
optics
optics.__table_column_meta__
camera
Query results
shower Storage of EventIndexContainer,MCEventContainer
obs_id event_id true_energy true_alt true_az true_core_x true_core_y true_h_first_int
10 1211 0.13218719 74.42848259 4.63300781 340.998016... 204.489883... 22399.5722...
10 26207 0.42419192 65.98558331 15.03587769 -180.46826... 259.565032... 11478.7099...
tel_082 Storage of TelEventIndexContainer,HillasParametersContainer,TimingParametersContainer,LeakageCo...
obs_id event_id tel_id hillas_intensity hillas_x hillas_y hillas_r hillas_phi
10 395206 82 59.47037601 -0.27410557 -0.45579357 0.53186619 -121.021898...
10 661016 82 84.98845863 -0.66257915 0.38146819 0.76454504 150.06957777
10 695602 82 89.55957222 -0.2881767 -0.73146064 0.78618095 -111.503166...
shower_distribution Storage of SimulatedShowerDistribution
obs_id hist_id num_entries bins_energy bins_core_dist histogram
10 6 2000000
tel_299 Storage of TelEventIndexContainer,DL1CameraContainer
obs_id event_id tel_id image peak_time image_mask
10 1072405 299
10 2612706 299
trigger Storage of TelEventIndexContaine...
obs_id event_id tel_id telescopetrigger_
10 1211 2 57571.7809...
10 1211 53 57571.7809...
10 26207 4 57571.71919...
10 26207 9 57571.71919...
10 28000 158 57571.7809...
protons.h5->/
DL1 data file output from ctapipe v0.8

Summary

ctapipe: changes from v0.7 → v0.8

KK - presentation and recording

Over ~~115~~ 120 Pull-Requests merged!

- **Many Thanks to** (in alphabetic order): Noah Biederbeck, Franca Cassol, Alice Donini, Momoka Goto, Jonas Hackfeld, KK, Dominik Niese, Max Nöthe, Abelardo Moralejo, Michele Perresano, Yves Renier, Thomas Vuillaume and Jason Watson
- **A very collaborative effort**! Not to mention: reviews, coding and design help, issue discussions... (many more people involved!)
- Also thanks to **collaborative tools**: GitHub, Slack, Travis, Codacy, Coverage.io, YouTube, etc.

Future:

- releases will be **more frequent** (common complaint, but usually due to manpower)
- provide **Component to write DL1 data**, so no need to re-implement data model in every chain

Main Development:
full *reference implementation* for first stage of pipeline (DL0 → DL1) & DL1 data model.

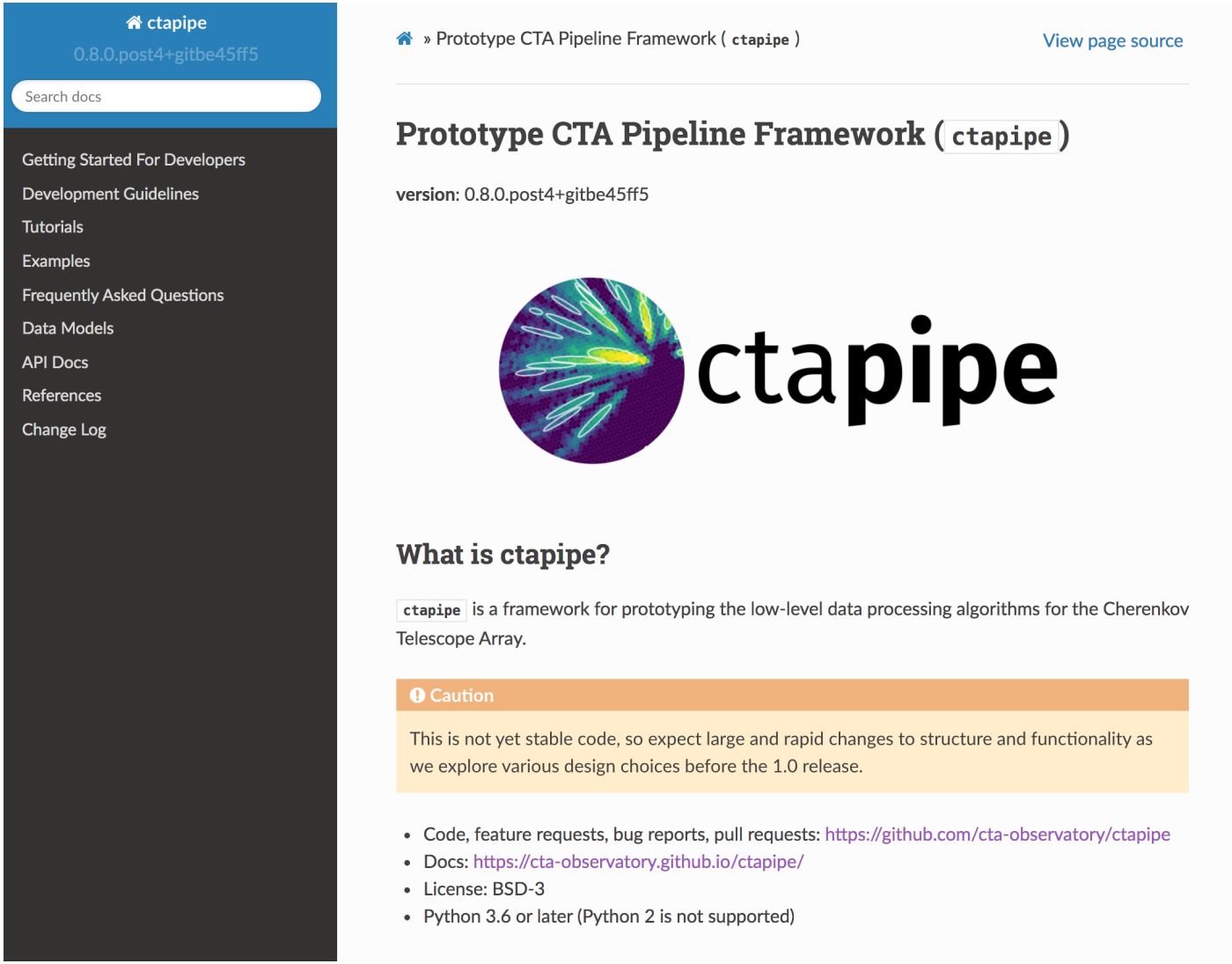
+ many new algorithms, bug-fixes, improvements

See full changelog in parallel talk linked above.

Expect release as conda package this week (final review underway)

pipelines plenary - May 2020 Karl Kosack 4

Documentation



https://cta-observatory.github.io/ctapipe/

Documentation

- Getting Started with ctapipe
  - Part 1: load and loop over data
  - Part 2: Explore the instrument description
  - Part 3: Apply some calibration and trace integration
  - Part 4: Let's put it all together:
    - We can now load in the file we created and plot it
- Coordinates usage in ctapipe
  - Open test dataset
    - Choose event with LST
  - AltAz
  - CameraFrame
  - TelescopeFrame
  - NominalFrame
  - GroundFrame
  - TiltedGroundFrame

- Exploring Raw Data
  - Explore the contents of an event
  - Look for signal pixels in a camera
  - Look at the time trace from a Camera Pixel
  - Do a very simplisitic trace analysis
  - Camera Displays
  - some signal processing...
- Explore Calibrated Data
  - Perform basic calibration:
  - Some image processing:
  - More complex image processing:
    - first, look at a summed image from multiple telescopes
- Make a theta-square plot
  - Plot the results

https://cta-observatory.github.io/ctapipe/tutorials/index.html

Documentation

Examples

Some lower-level examples of features of the ctapipe API (see the Tutorials section for more complete examples)

Algorithms

- Basic Image Cleaning and Dilation
- Use N-dimensional Histogram functionality and Interpolation
- Convert hexagonal cameras into square grid and back

Core functionality

- Working with Instrumental Descriptions
- Example of a Camera Display
- Using Container classes
- Creating command-line Tools
- Using the ctapipe Provenance service
- Writing Containers to a tabular format
- containers_with_enums_and_table_writer

What's next?

Converging on Data Models and Formats

- driven by CTAO + ASWG
- even with incompatible pipelines, a good step toward ***reusability***

Convergence toward a standard pipeline for Simulation analysis… (protopipe, ctaplot, pyrf, cta-benchmarks)

Knowledge Transfer (down to lowest levels)

Soon: Convergence towards a standard pipeline for real data as well

- New **DL1/DL2 data model + formats** will help!
- learn from each prototype, take the best parts and combine…

What's next?

Ideal Situation: A division of Responsibility

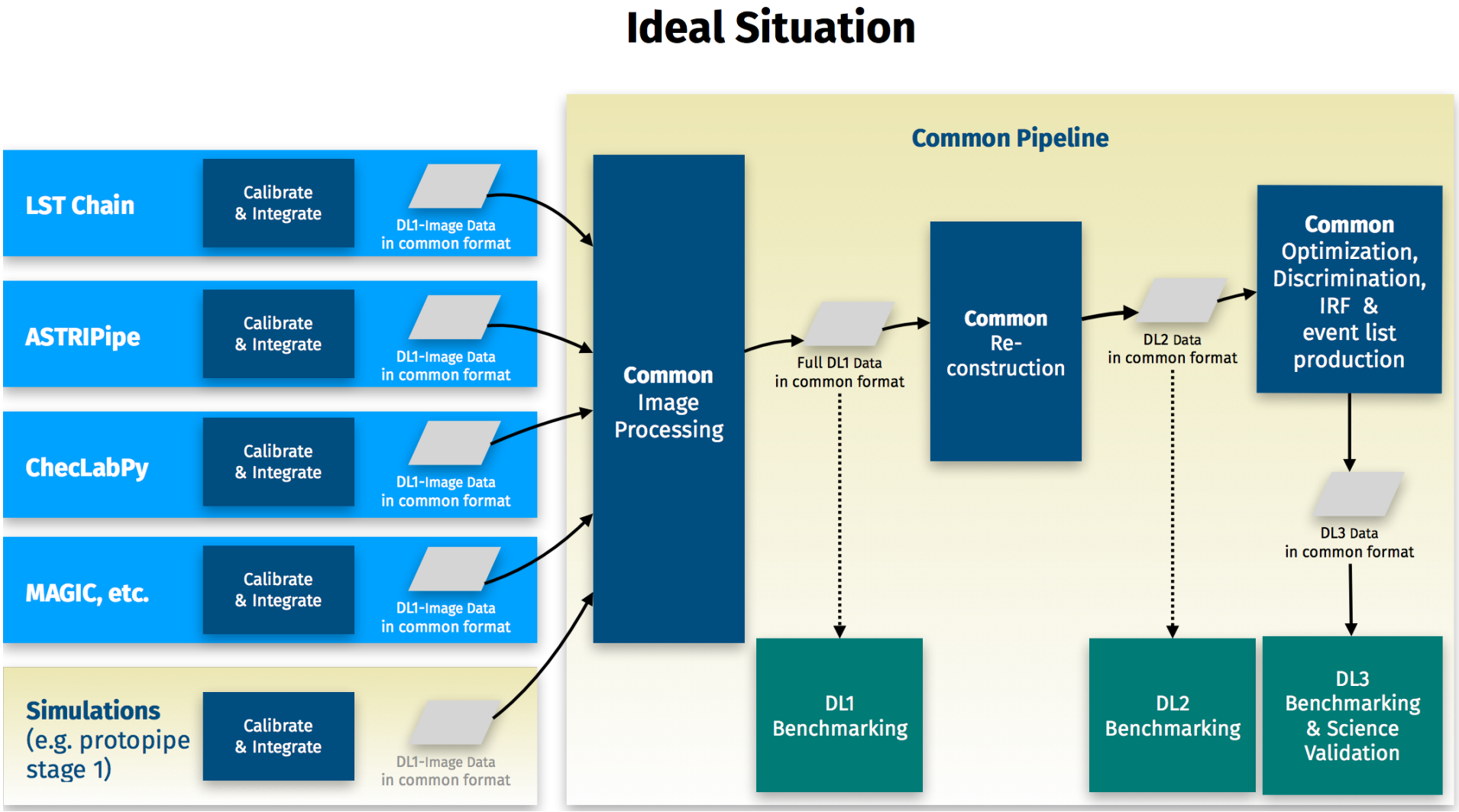
Telescope Benchmark Pipelines (and Simulation Processing Pipeline):

- should **end** at the production of *DL1-Image* data in **common format**
  - includes calibration and waveform integration
  - may or may not use common software for this (ctapipe, astripipe, MAGIC MARS)

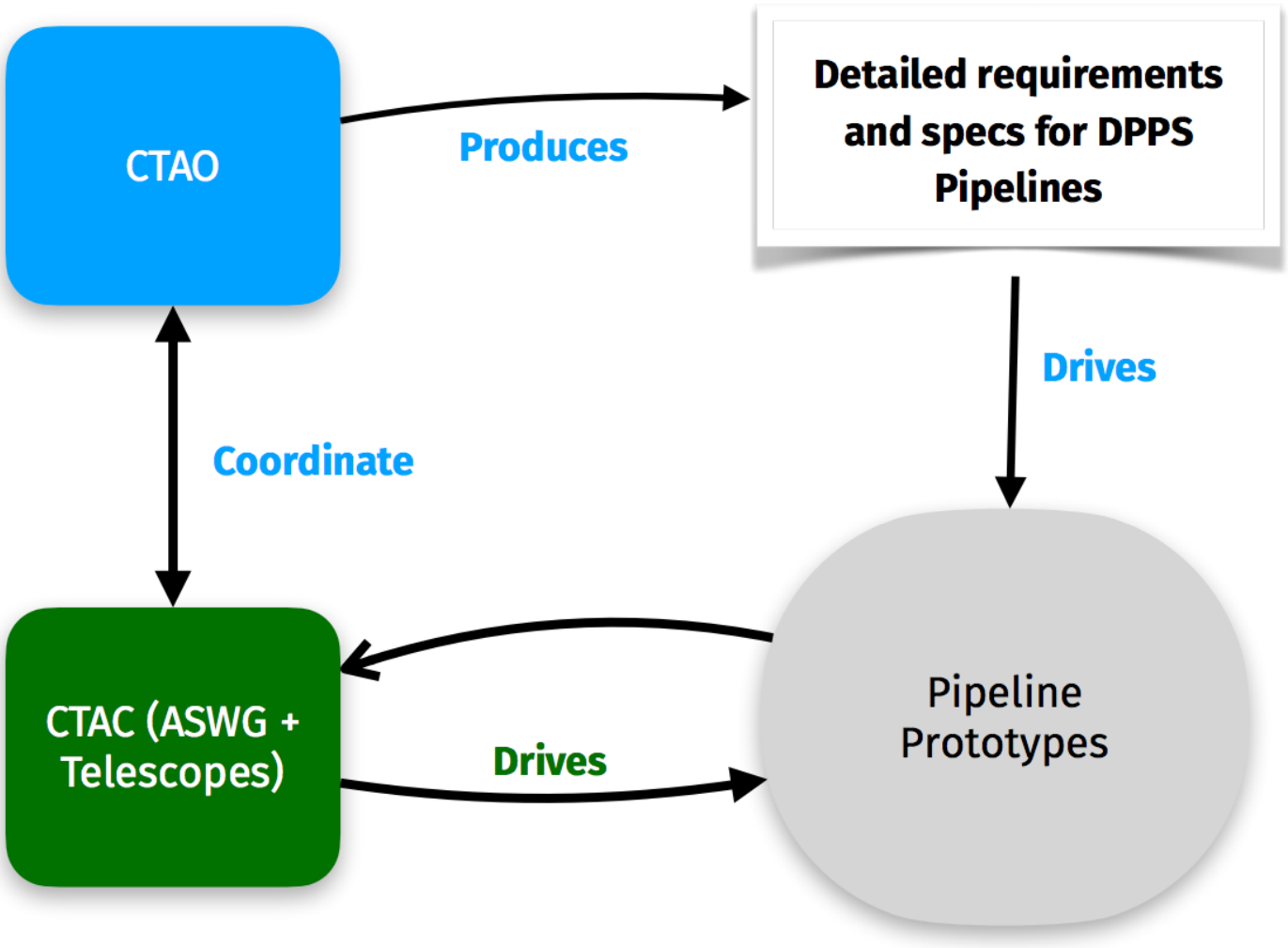
Common Pipeline:

- **Starts** with *DL1-Image* data from telescopes (or *Sim-R0* data from SimTelArray)
- Does **All event processing**: parameterization, reconstruction, optimization, discrimination
- **Benchmarking** of each data level
- Generation of **Science Validation** products

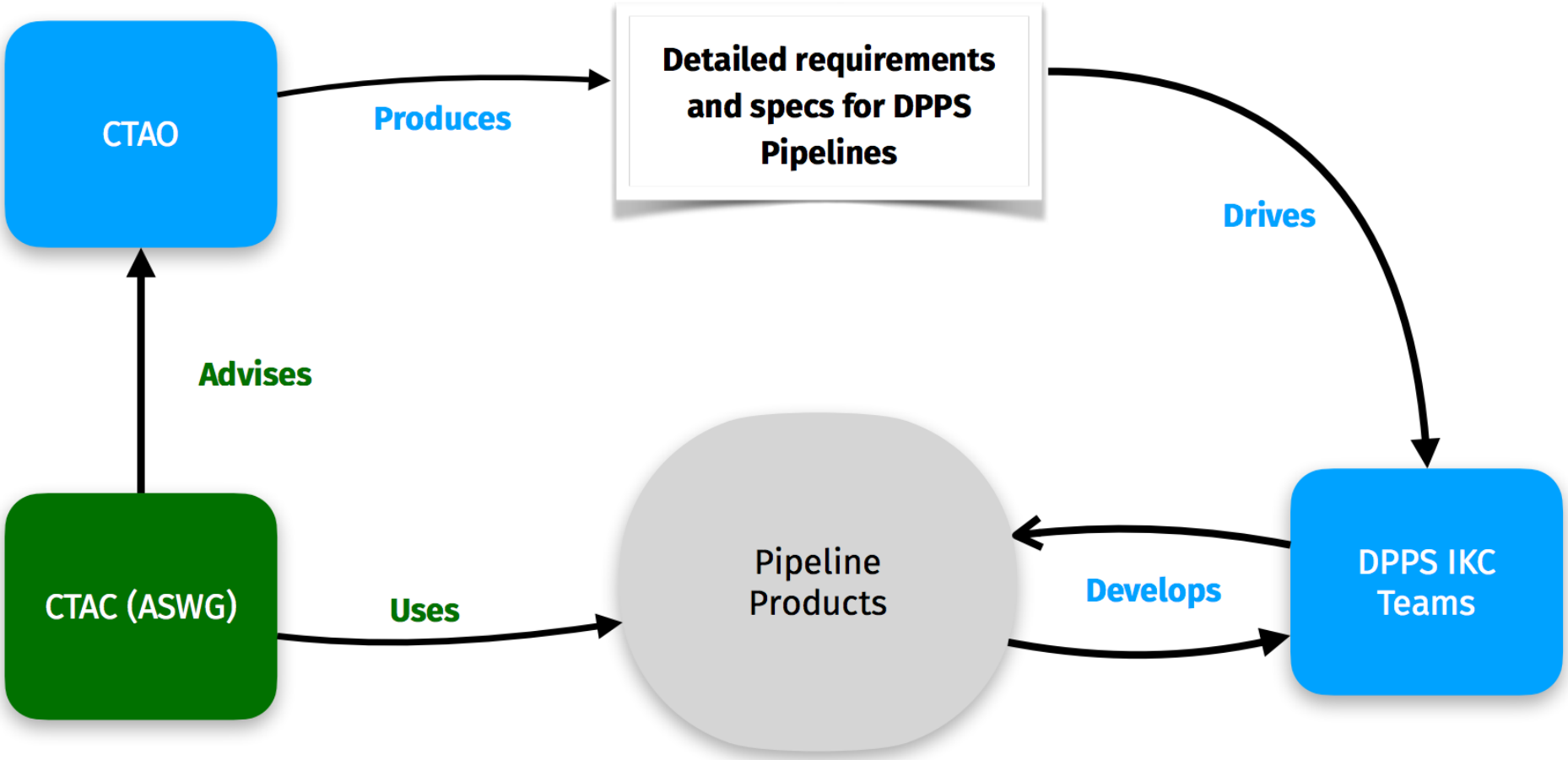
What's next?


Ideal Situation
Common Pipeline
LST Chain
Calibrate & Integrate
DL1-Image Data in common format
ASTRIPipe
Calibrate & Integrate
DL1-Image Data in common format
ChecLabPy
Calibrate & Integrate
DL1-Image Data in common format
MAGIC, etc.
Calibrate & Integrate
DL1-Image Data in common format
Simulations (e.g. protopipe stage 1)
Calibrate & Integrate
DL1-Image Data in common format
Common Image Processing
Full DL1 Data in common format
Common Re-construction
DL2 Data in common format
Common Optimization, Discrimination, IRF & event list production
DL3 Data in common format
DL1 Benchmarking
DL2 Benchmarking
DL3 Benchmarking & Science Validation

What's next?


CTAO
Produces
Detailed requirements and specs for DPPS Pipelines
Drives
Coordinate
CTAC (ASWG + Telescopes)
Drives
Pipeline Prototypes

What's next?

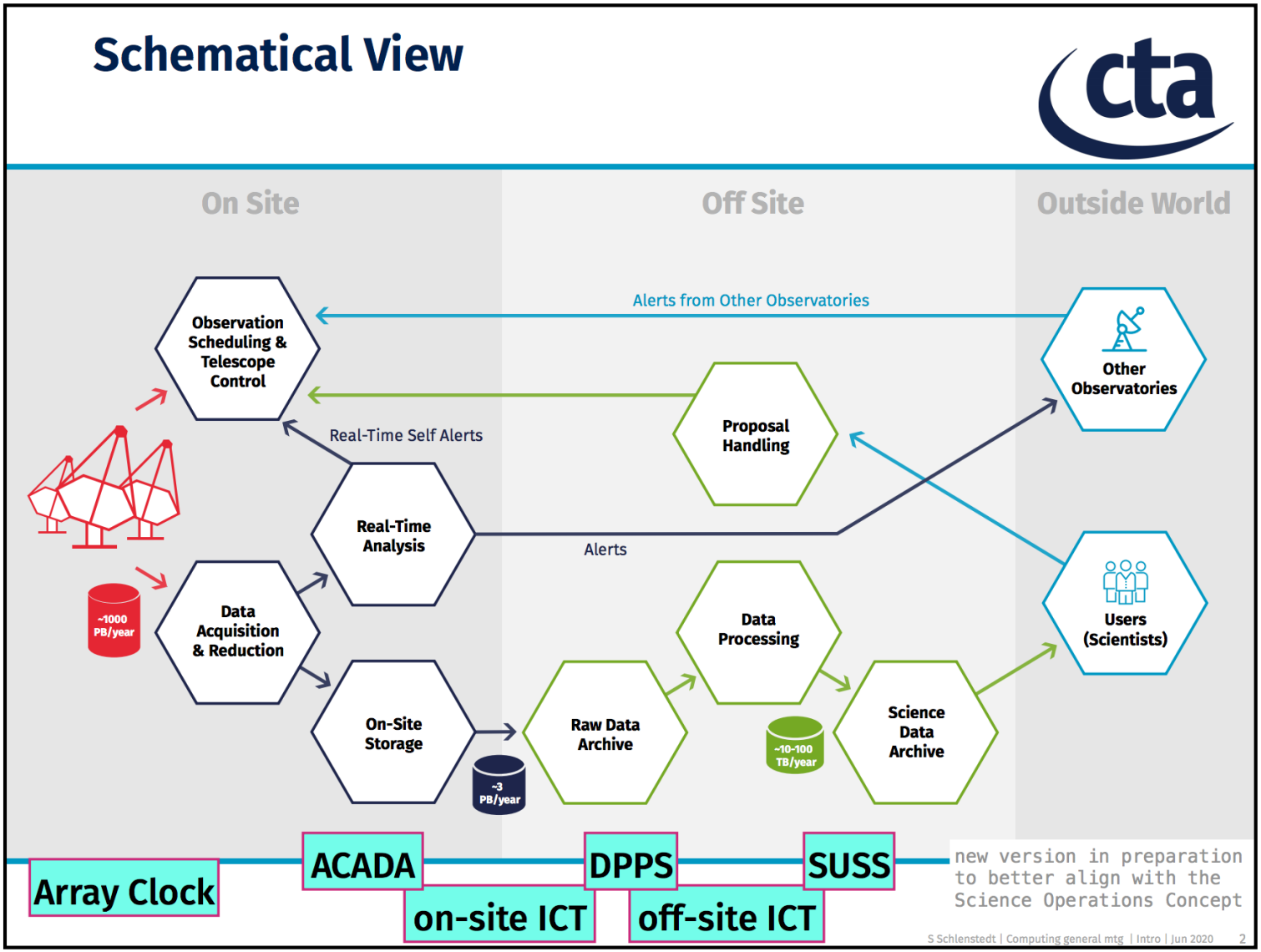

CTAO
Produces
Detailed requirements and specs for DPPS Pipelines
Drives
Advises
CTAC (ASWG)
Uses
Pipeline Products
Develops
DPPS IKC Teams

What's next?

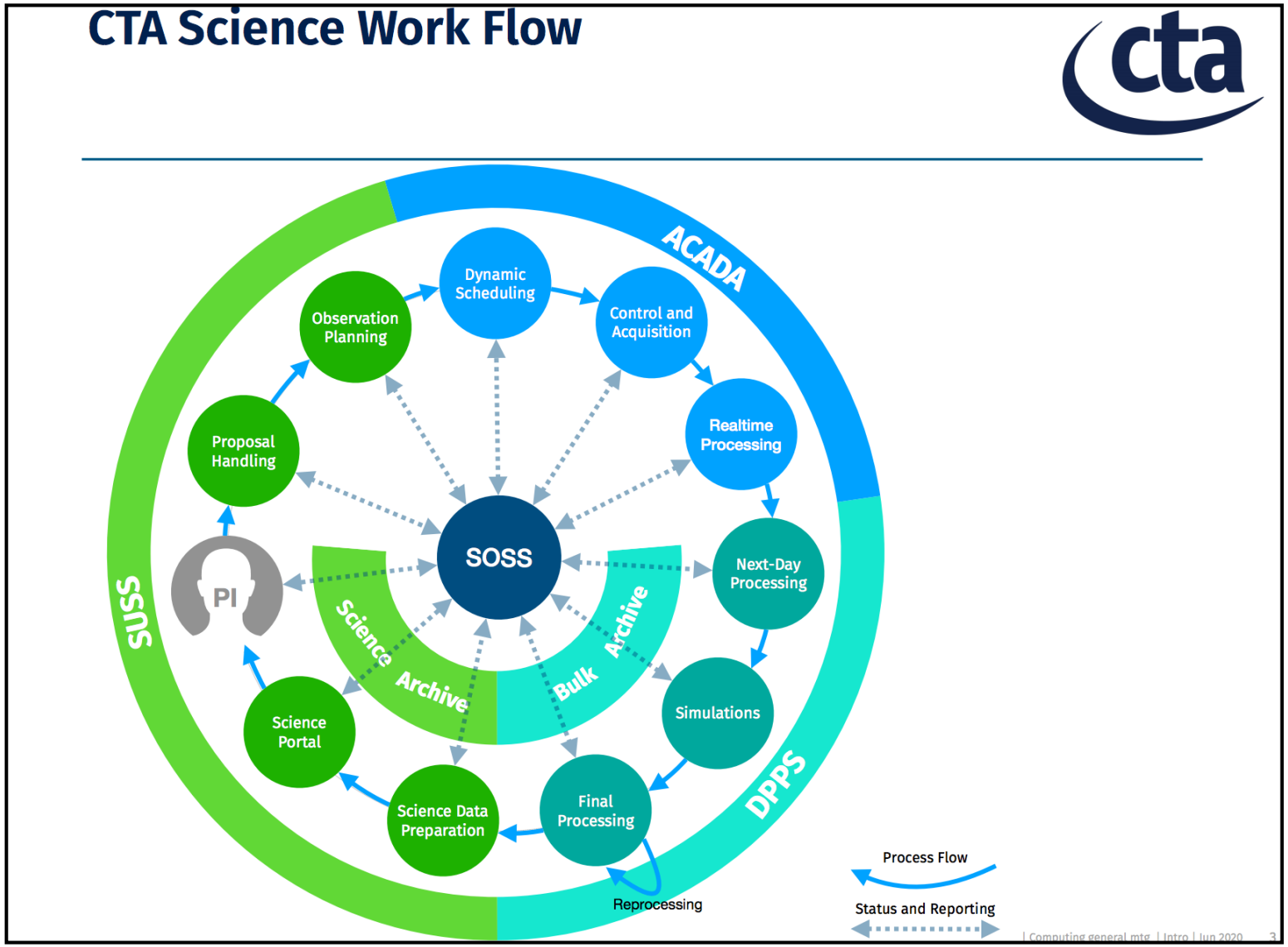
Computing WBS

1. Computing Project Management
2. Computing System Engineering and Infrastructure
3. Software Architecture and Applications
4. Array Control and Data Acquisition System
5. Data Processing and Preservation System
6. Science User Support System
7. Science Operations support system
9. On-site ICT Implementation
10. Array Clock System
11. Off-site ICT Implementation

What's next?


Schematical View
cta
On Site
Off Site
Outside World
Observation Scheduling & Telescope Control
Alerts from Other Observatories
Other Observatories
Proposal Handling
Real-Time Self Alerts
Real-Time Analysis
Alerts
~1000 PB/year
Data Acquisition & Reduction
Data Processing
Users (Scientists)
On-Site Storage
Raw Data Archive
~10-100 TB/year
Science Data Archive
~3 PB/year
Array Clock
ACADA
DPPS
SUSS
on-site ICT
off-site ICT
new version in preparation to better align with the Science Operations Concept
S Schlenstedt | Computing general mtg | Intro | Jun 2020 2

What's next?


CTA Science Work Flow
cta
ACADA
DPPS
SUSS
Dynamic Scheduling
Control and Acquisition
Realtime Processing
Next-Day Processing
Simulations
Final Processing
Science Data Preparation
Science Portal
Proposal Handling
Observation Planning
SOSS
PI
Science Archive
Bulk Archive
Reprocessing
Process Flow
Status and Reporting
| Computing general mtg | Intro | Jun 2020 3

What's next?

In-Kind Contributions and Expression of Interest

- ACADA: IKC teams are working – see Igor's talk
- Plan on-site ICT on CTA-N as IKC with U Tokyo (till move)
- The IKC Framework
  - "CTAO will issue a call for Expressions of Interest for potential IKCs and will provide information to Shareholders about the work package specifications of the IKCs."
  - "In the event of more than one Shareholder being interested in supplying, independently, the same IKC, CTAO [...] shall act as intermediary to foster communication and collaboration spirit among said Shareholders with the aim of arriving at a joint solution. The final decision will be taken by Council [...] with a recommendation by the IKRC. "
- DPPS Expression of Interest (EoI) issued before the summer
  - EoI Template Form; Description of EoI Process for DPPS
  - DPPS WP Management Plan
  - The Concept for DPPS
  - DPPS Requirement Specifications
  - The DPPS Architecture

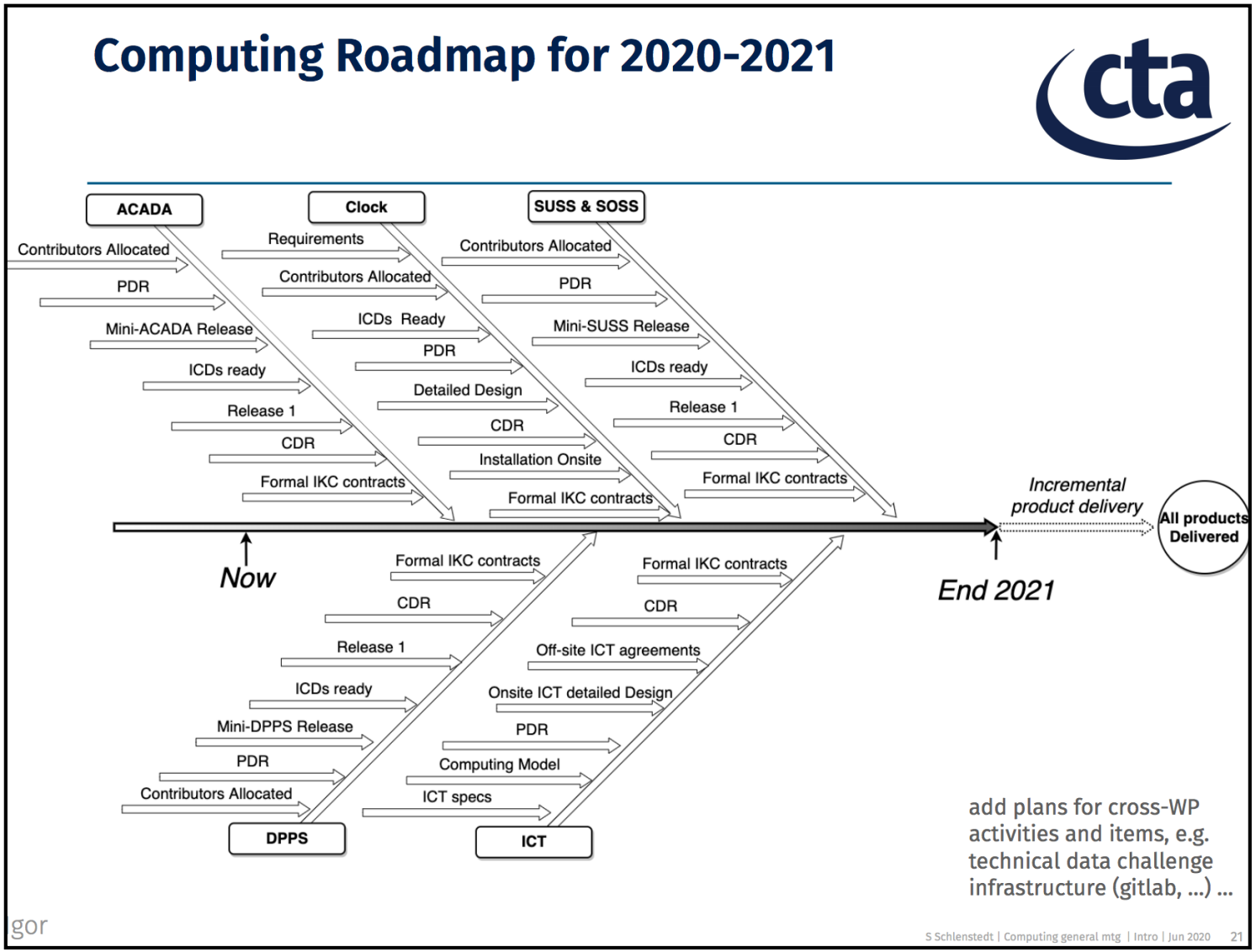
see Karl's talk

- SUSS will follow the same path as DPPS

see Nadine's talk

- off-site ICT will follow: Computing Model implementation

What's next?


Computing Roadmap for 2020-2021
ACADA
Contributors Allocated
PDR
Mini-ACADA Release
ICDs ready
Release 1
CDR
Formal IKC contracts
Clock
Requirements
Contributors Allocated
ICDs Ready
PDR
Detailed Design
CDR
Installation Onsite
Formal IKC contracts
SUSS & SOSS
Contributors Allocated
PDR
Mini-SUSS Release
ICDs ready
Release 1
CDR
Formal IKC contracts
Incremental product delivery
All products Delivered
Now
End 2021
DPPS
Formal IKC contracts
CDR
Release 1
ICDs ready
Mini-DPPS Release
PDR
Contributors Allocated
ICT
Formal IKC contracts
CDR
Off-site ICT agreements
Onsite ICT detailed Design
PDR
Computing Model
ICT specs
add plans for cross-WP activities and items, e.g. technical data challenge infrastructure (gitlab, ...) ...
gor
S Schlenstedt | Computing general mtg | Intro | Jun 2020 21