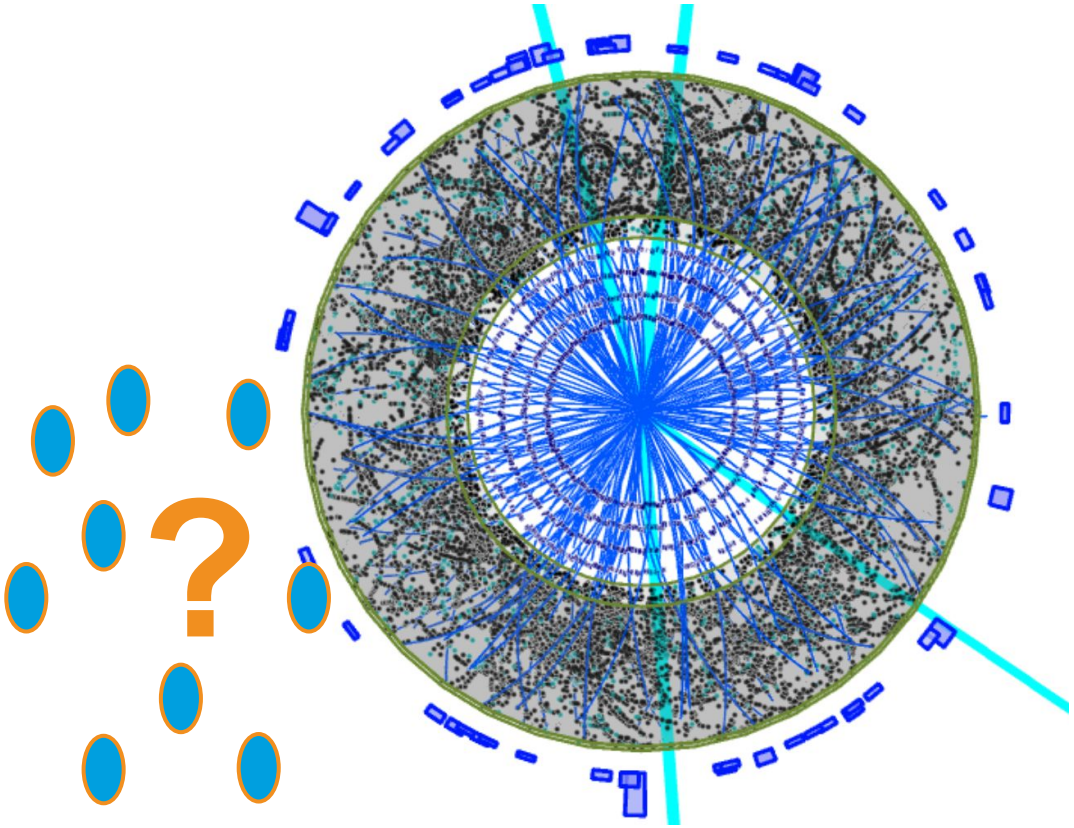
Requirements from ATLAS

Needs from the experiment

Nick Styles

ACTS Workshop

26.09.2022



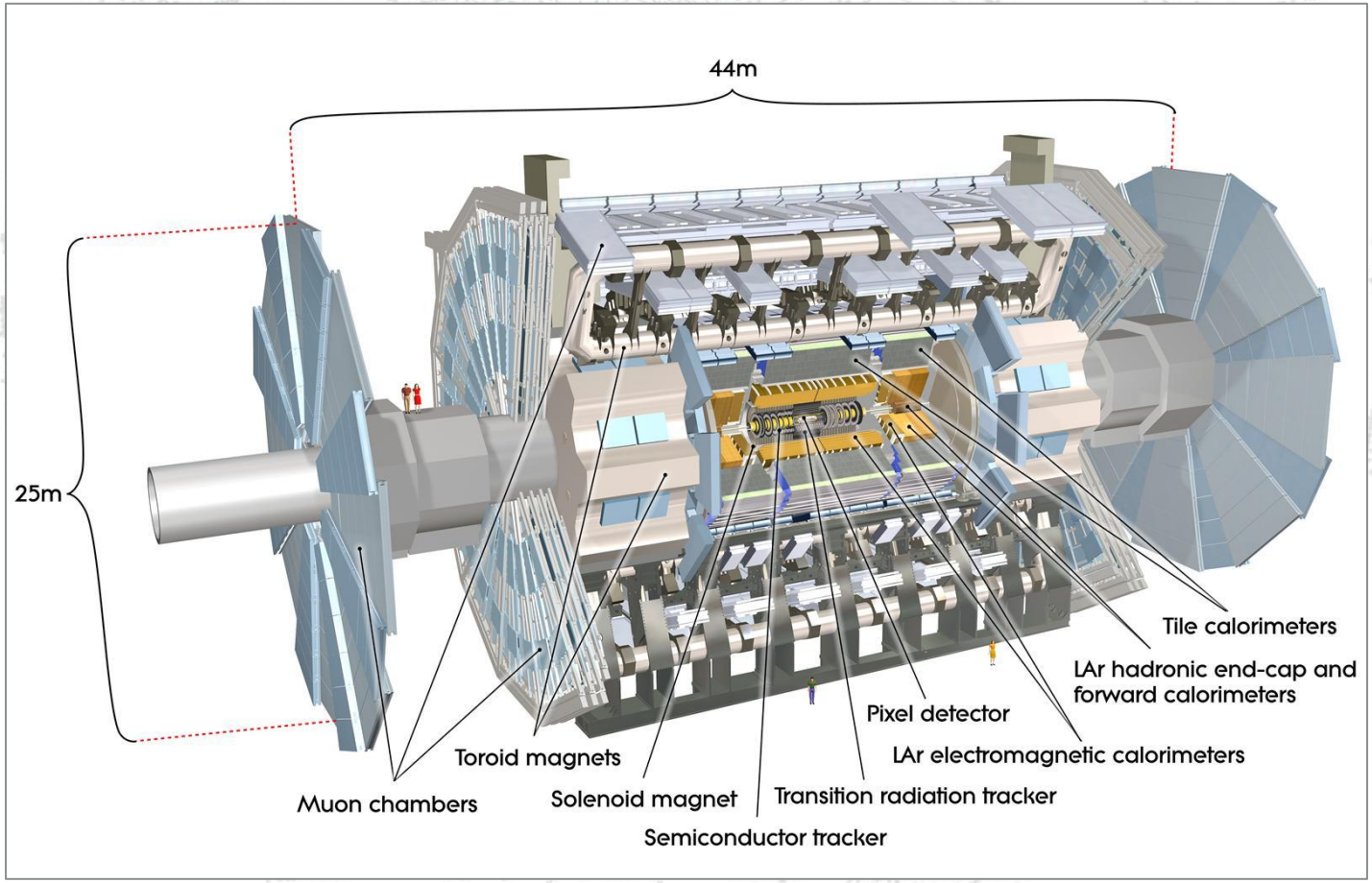
HELMHOLTZ RESEARCH FOR GRAND CHALLENGES

DESY.

ATLAS Experiment

ATLAS in a nutshell

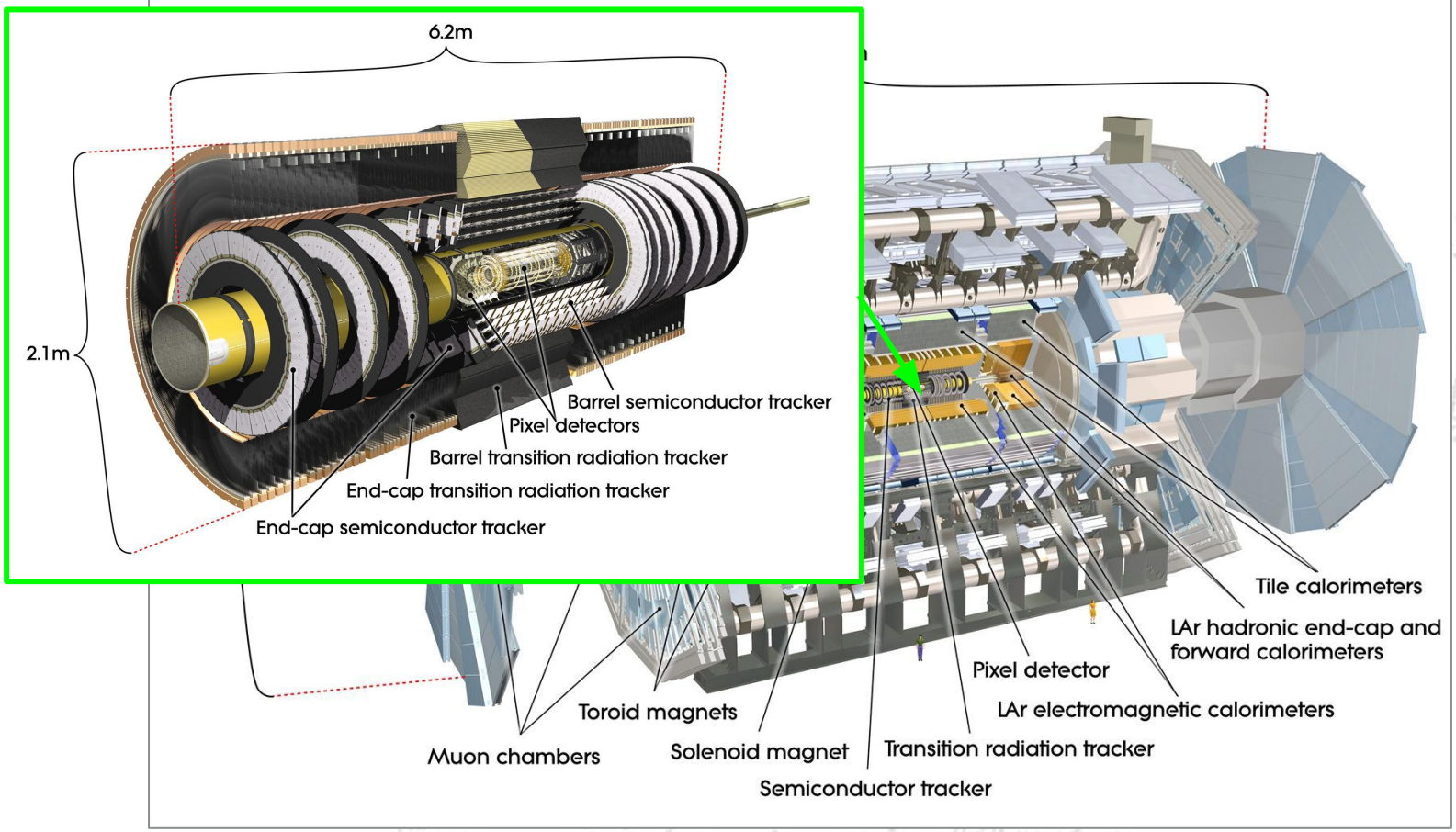
- General-purpose detector at the LHC



ATLAS Experiment

ATLAS in a nutshell

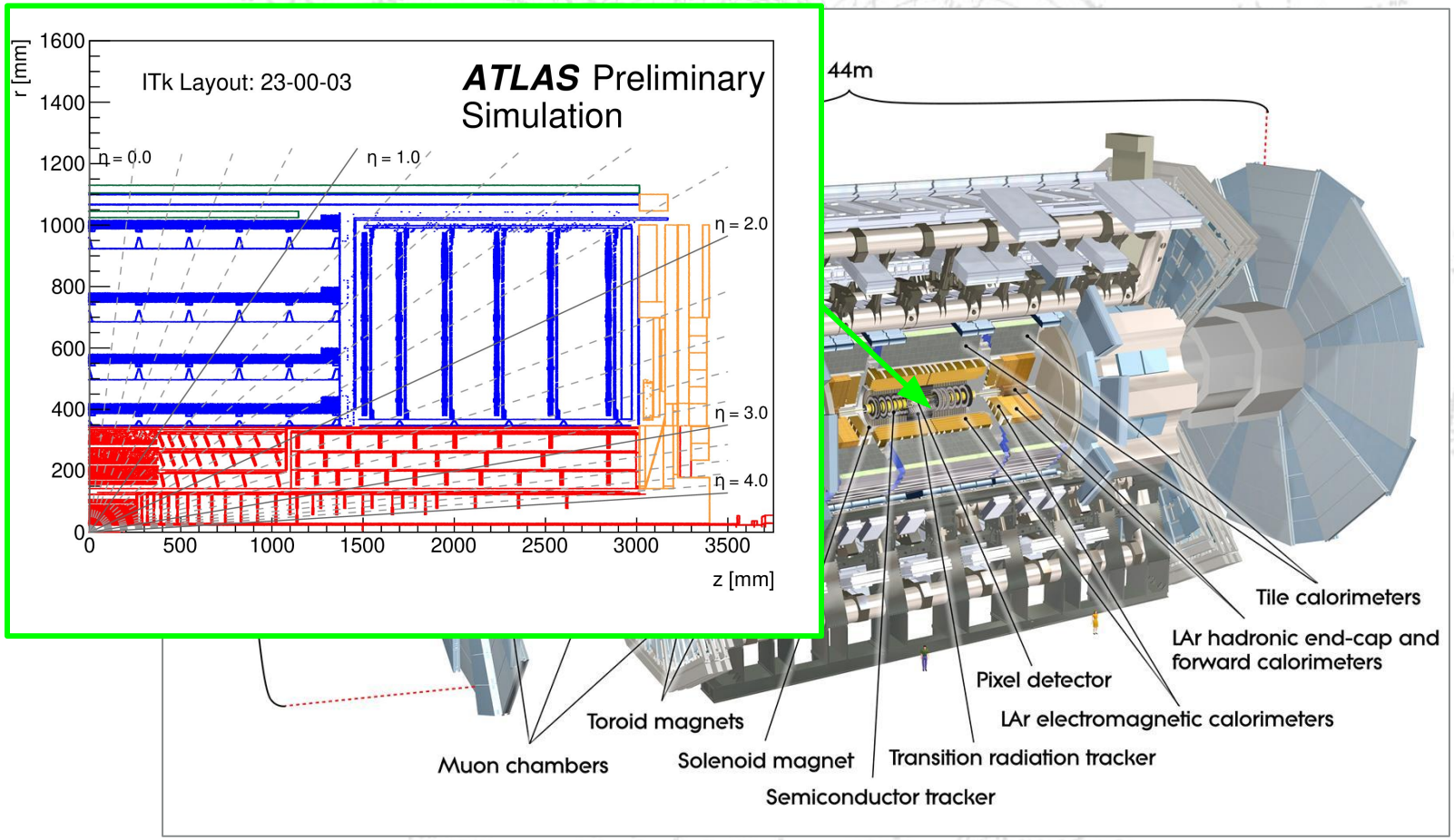
- LHC Runs 1/2/3: Inner Detector comprising silicon pixel & microstrip (SCT) detectors, plus gaseous straw tube Transition Radiation Tracker (TRT)



ATLAS Experiment

ATLAS in a nutshell

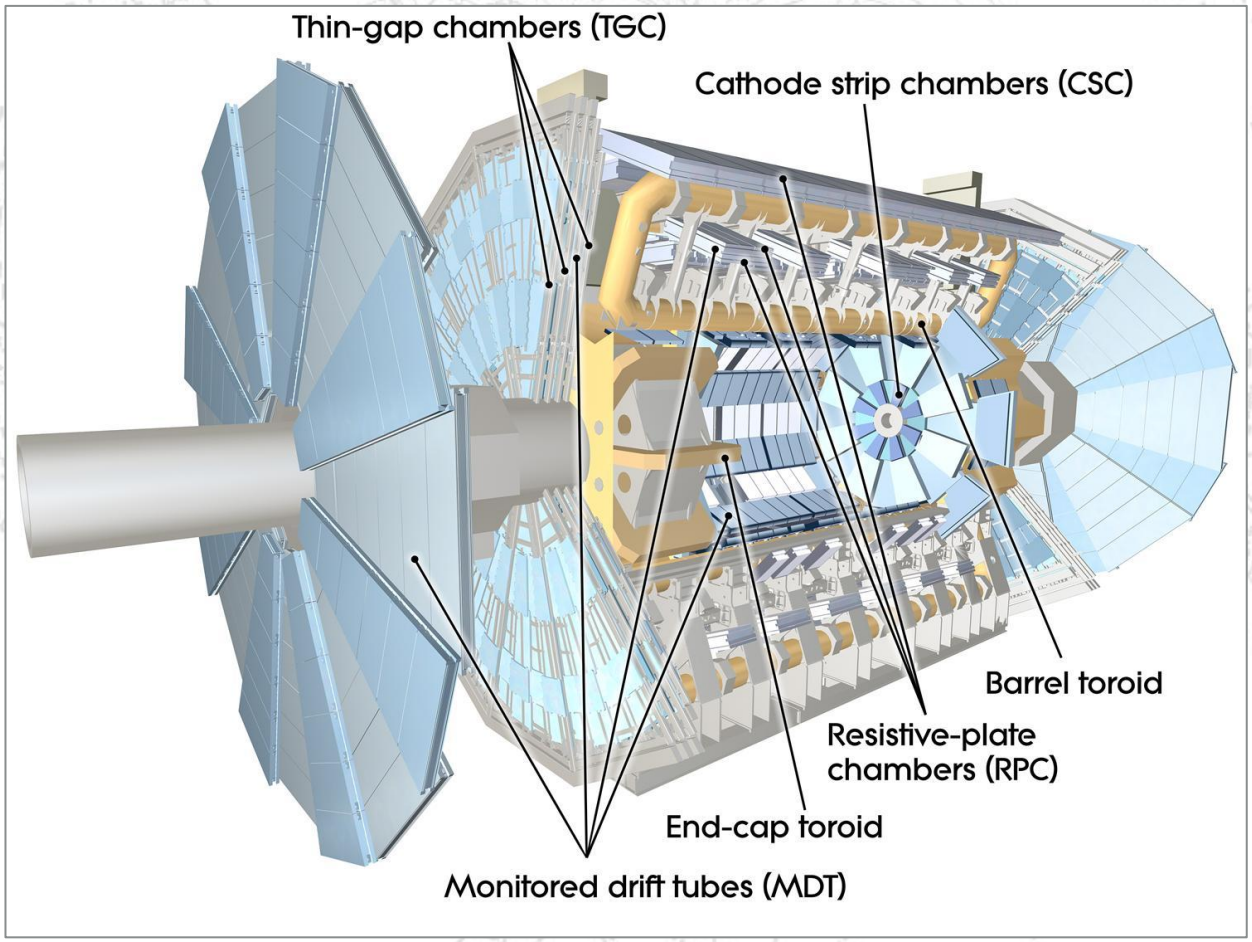
- HL-LHC: New all-silicon tracking detector (ITk) comprising pixel and microstrip detectors, optimized for performance at significantly higher instantaneous luminosities (<mu>~200)



ATLAS Experiment

ATLAS in a nutshell

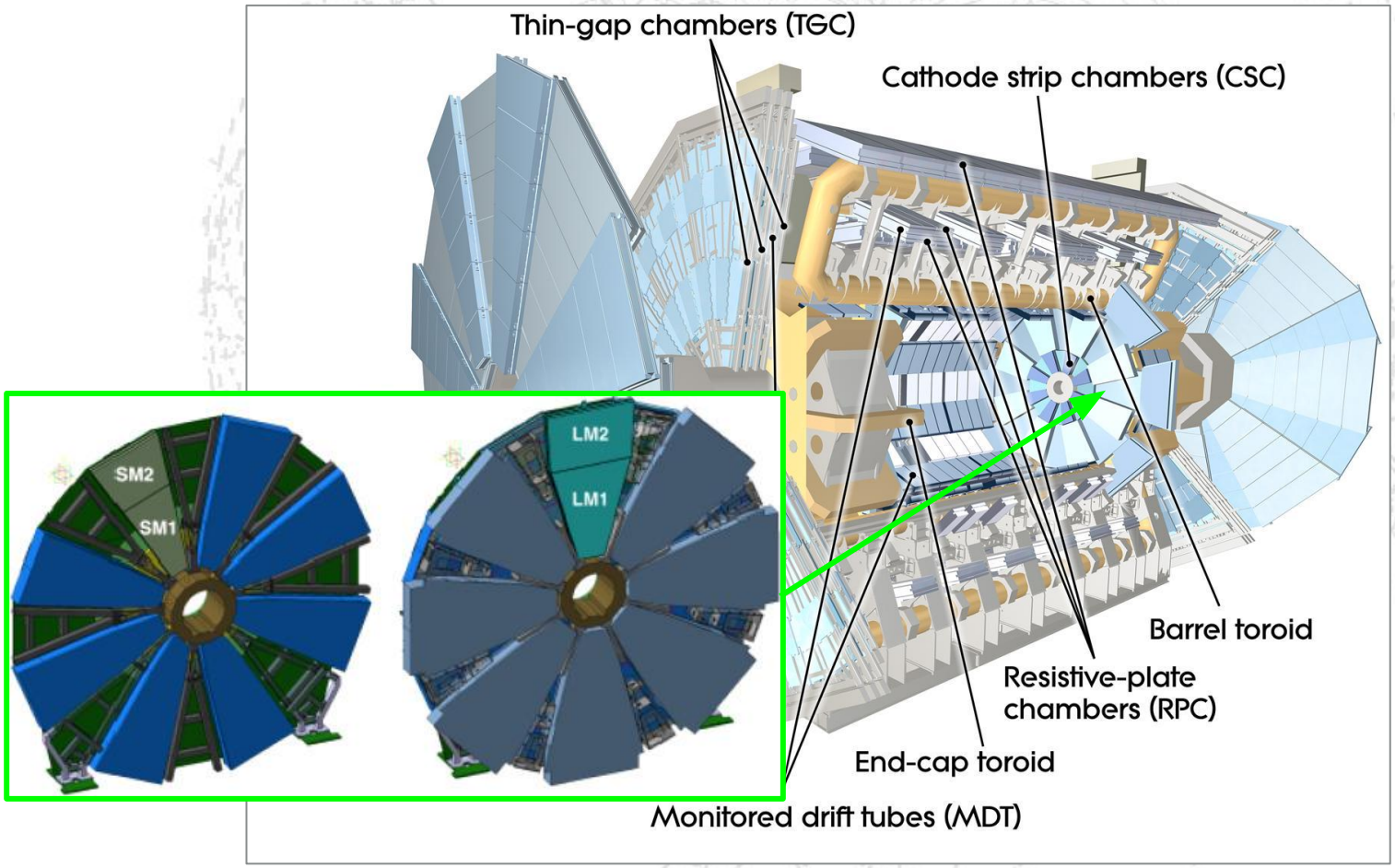
- Muon Spectrometer: 4 different technologies used: TGC, RPC, CSC, MDT



ATLAS Experiment

ATLAS in a nutshell

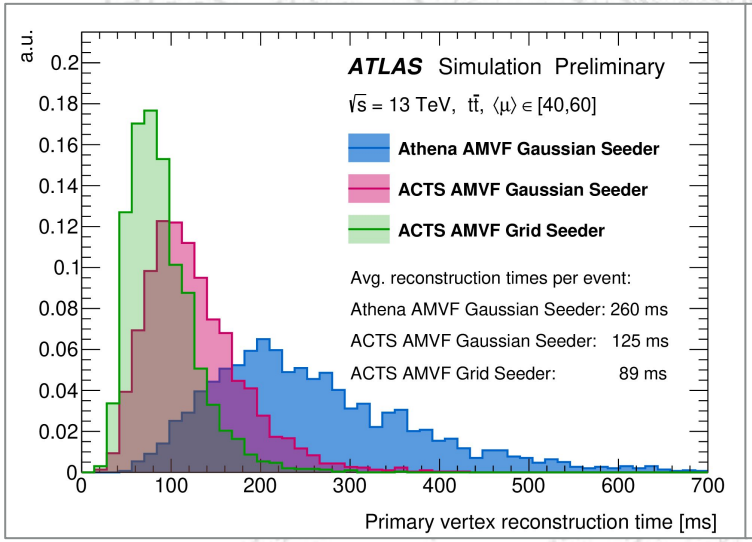
- Muon Spectrometer: Since Run3, New Small Wheels (NSW) introduce Micromegas (MM) and Small Thin Gap Chambers (sTGC)

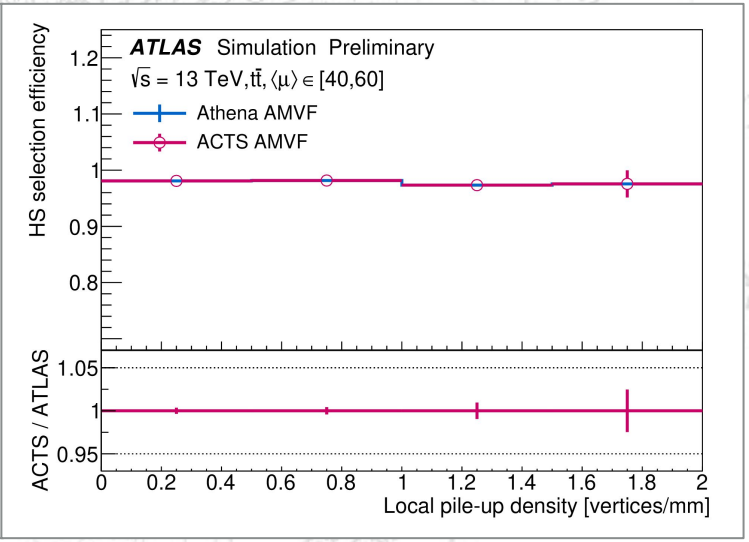


Plans for ACTS Usage

Integration with Athena framework

- Ideally hope to use ACTS tools for entire track reconstruction chain
  - Starting from measurement formation, through to final track fit
  - Replacing legacy code that was in parts written prior to Run 1 (~15 years ago) and incrementally updated - large technical debt accrued
  - Already using ACTS tools successfully for vertex reconstruction





- To really benefit, should migrate to ACTS for both InDet/ITk and Muon Spectrometer **plus** both Online (i.e. trigger) and Offline reconstruction
  - Otherwise legacy tools may still require support, splitting effort of limited developer base - this is the scenario we want to avoid at all costs!

Implications

What does this mean we need from ACTS

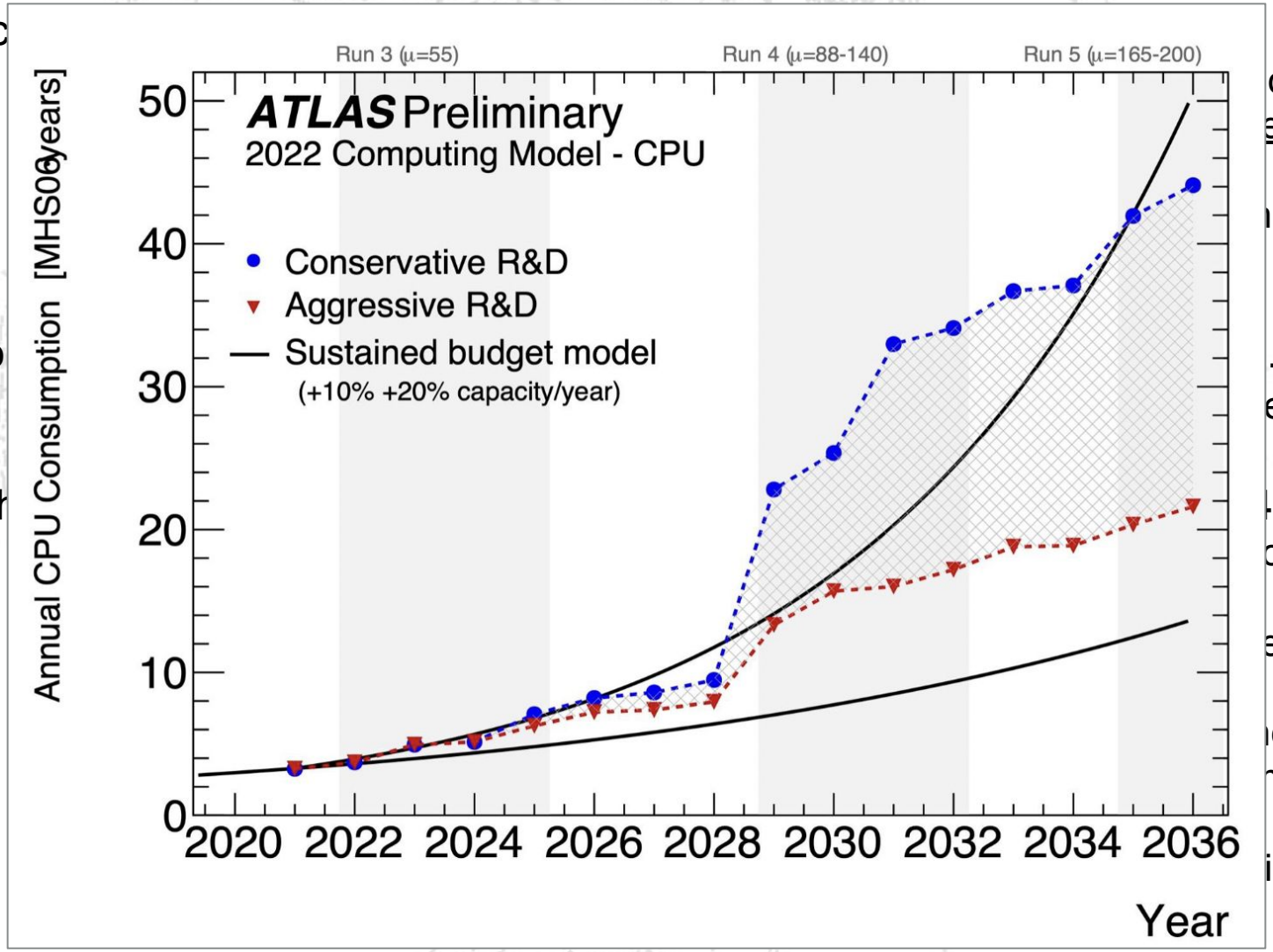
- All current and future detector technologies should be supported
  - Run4 software not supporting at least Run3 data/MC reconstruction risks the inability to expose our Run4 SW to data prior to start of HL-LHC
  - Would mean that ideally e.g. TRT drift circle formation and track fits allowing sign ambiguities would need to be available (ATLAS policy not yet final)
  - Also that combined fits between InDet/ITk and MS must be supported
- Should ease pending technology decisions (e.g. x86 vs GPU vs FPGA …)
  - Since most current ATLAS work on tracking in heterogeneous systems comes via traccc R&D line this is already on a good footing
- Technical and physics performance should meet requirements for Run4/Run5
  - For ATLAS track reconstruction at HL-LHC, overall require significant reductions to CPU and memory overheads compared to Run3, achieved through appropriate choice of algorithms, configuration, phase space, etc...
  - From ACTS we require comparable efficiencies, fake rates, parameter resolutions, etc, to current code, for similar computing performance*
  - Small physics performance penalties may be tolerable if they lead to large technical benefits
  - (see for example work on fast tracking chain in current software - anticipates further improvement to be acceptable performance for offline)

*Precise metrics to judge this need to be decided and agreed

Implications

What does this mean we need from ACTS

- All c
  - ould
  - e our
  -
  -
- Sho
  - …)
  - es via
- Tech
  -
  - ode, with
  -
  - echnical
  -
  - ıce
  - nance
  -
  - ion



Practicalities I

A few organizational points

- ATLAS applies a policy known as “Frozen Tier-0” to production branches of Athena
  - Changes to reconstruction output only permitted for solving serious bugs
  - (e.g. issues causing unacceptable job failure rates, or making output unusable for physics)
  - In case an important ACTS update came only in a new major version, likely not possible to apply this in a production branch of Athena
  - Seems to be in line with ACTS “semantic versioning” policies
- ACTS release policy presumably includes plans for patch versions
  - Other clients may have similar policies
  - Will there be potential for experiment-specific patch branches?
  - What would be threshold for these?
- Related issue of EDM and API changes
  - May need some lead-in time to adapt Athena code base for incoming API changes, in particular for very commonly-used tools (extrapolator, etc)
  - Presumably these be limited to major release versions?
  - (or at least avoid deprecation of old interfaces outside of major version updates?)
  - EDM changes may require *even more* lead-in time
  - How will these be advertised ahead of time? Will there be a formal evolution strategy for supporting older EDM versions?

Practicalities II

A few organizational points

- ATLAS benefits in communication with ACTS community due to many ACTS developers being ATLAS collaboration members
  - For future, do we need to formalize how communication and bug reporting will be done?
  - For example: Recent fixes for some vertexing issues have relied on developers being able to view ATLAS (internal) JIRA tickets
  - Presumably github issues can be opened, and Athena repository is public, but may not be always possible to provide a reproducer if the issue requires ATLAS data or MC to demonstrate
- Also communication in the other direction
  - Through what channels will ATLAS hear about bugs found in ACTS?
  - These may have to be factored in to our Software release planning
  - (we hope/expect this will be a relatively infrequent occurrence)

Common Definitions

Understanding what to expect from updates

- As mentioned, both maintaining performance and ensuring reproducibility are critical
  - In order to ensure this, need to make sure that the metrics for these are consistent
- ACTS has its own test suite to check its consistency, but possible that this does not cover full phase space?
  - Blind spots also in Athena Continuous Integration tests which can “conspire together” to hide issues
  - Risk only finding differences when large statistics are processed - how best to mitigate this should be considered
- Similarly for reporting performance changes - need common basis for understanding
  - E.g. ACTS reports efficiency improvements from moving to new version after improvements to implementation - great, but…
  - This can depend on e.g. phase space like min p_T threshold, which can be very different between client applications
  - Similarly for technical improvements: different measurement density (i.e. <mu>) can make the optimal strategy different
- Some common understanding and reference points must be established

Summary

- ATLAS hopes to use ACTS extensively for its track reconstruction in future
  - As part of a highly performant, highly maintainable code base
  - Already replaced some legacy components with ACTS in our most recent production releases
  - Dedicated efforts ongoing on to plan and implement integration of other components
- To make process as simple as possible, several important factors to consider
  - Release and branching strategy - understanding what to expect from moving versions
  - Related: communication of upcoming EDM and API changes
  - Reporting of bugs/performance issues
  - Common basis for performance comparisons
- Looking forward to a very fruitful collaboration with ACTS team!