Planck Lessons Learned: Simulations, Data Analysis & High Performance Computing

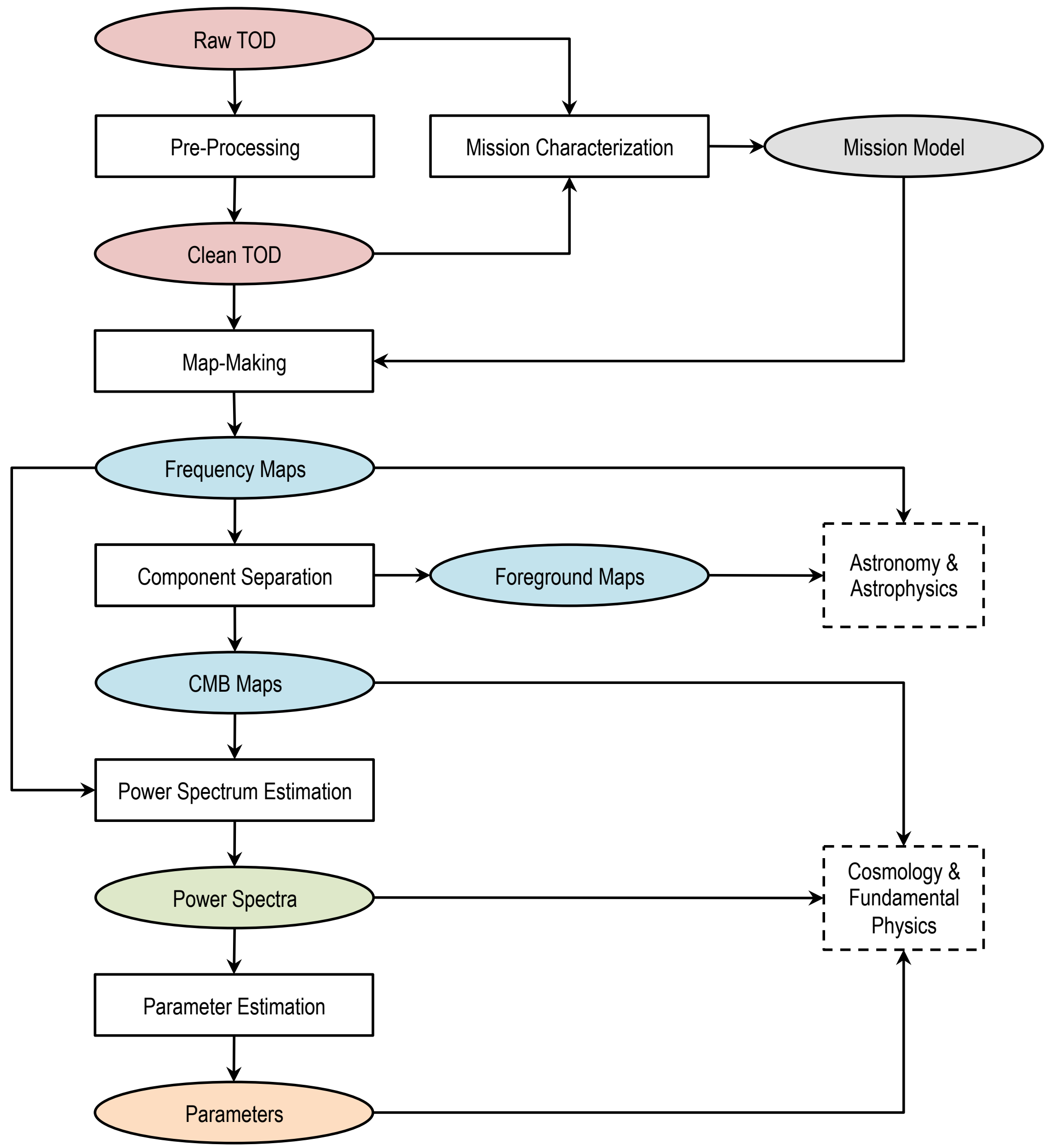
Julian Borrill, Reijo Keskitalo & Ted Kisner
Computational Cosmology Center, Berkeley Lab
& Space Sciences Laboratory, UC Berkeley

The scientific results that we present today are a product of the Planck Collaboration, including individuals from more than 100 scientific institutes in Europe, the USA and Canada.

Planck is a project of the European Space Agency, with instruments provided by two scientific Consortia funded by ESA member states (in particular the lead countries: France and Italy) with contributions from NASA (USA), and telescope reflectors provided in a collaboration between ESA and a scientific Consortium led and funded by Denmark.

Data Analysis

- Sequence of S/N-increasing data compressions via domain transformations:
  - Time
  - Pixels
  - Multipoles
  - Parameters
- Each domain exposes different systematics => iterative looping.
- Must propagate both data *and covariance* for sufficient statistic.



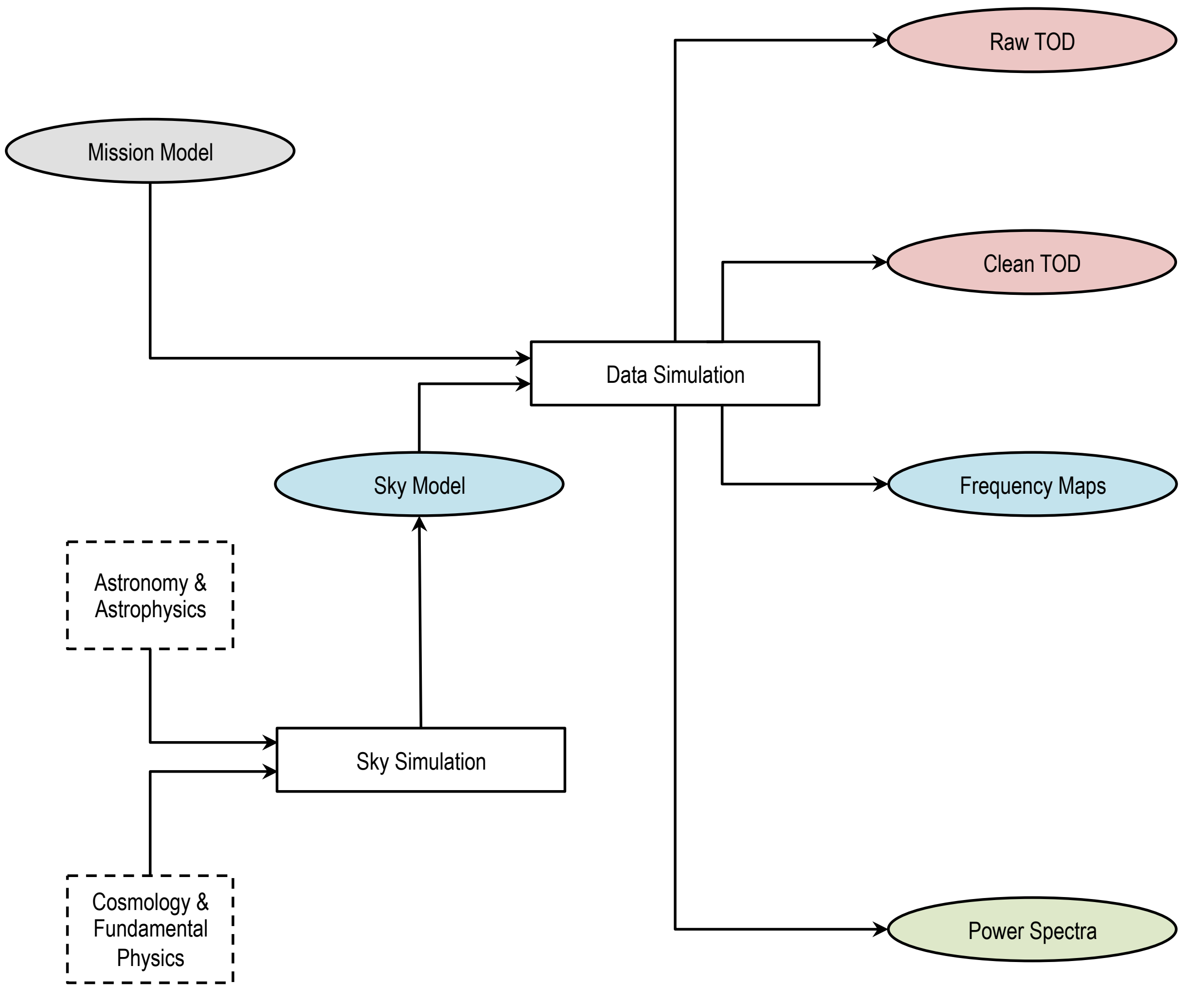
Analysis Methods

- CMB data volumes:
  - Time domain: $\mathcal{N}_t \sim \Sigma_{det}$ Sampling Rate (Hz) x Observation Time (s)
  - Pixel domain: $\mathcal{N}_p \sim \Sigma_{freq,\ pol}\ 10^9$ x Sky Fraction / [Beam (arcmin)]^2
- CMB data analysis scaling dominated by:
  - $\mathcal{N}_p^{\,3}$ for exact methods with explicit covariance matrices.
  - $\mathcal{N}_{mc}\,\mathcal{N}_t$ for approximate methods with MC uncertainty quantification.
- Computational constraints (1% cycles/year on Top 10 system):
  - 2000 : $\mathcal{N}_p < 10^6$ & $\mathcal{N}_t < 10^{12}$
  - 2015 : $\mathcal{N}_p < 10^7$ & $\mathcal{N}_t < 10^{15}$
  - 2030 : $\mathcal{N}_p < 10^8$ & $\mathcal{N}_t < 10^{18}$

Assumes:
- Moore's Law
- 100% & 1% efficiency

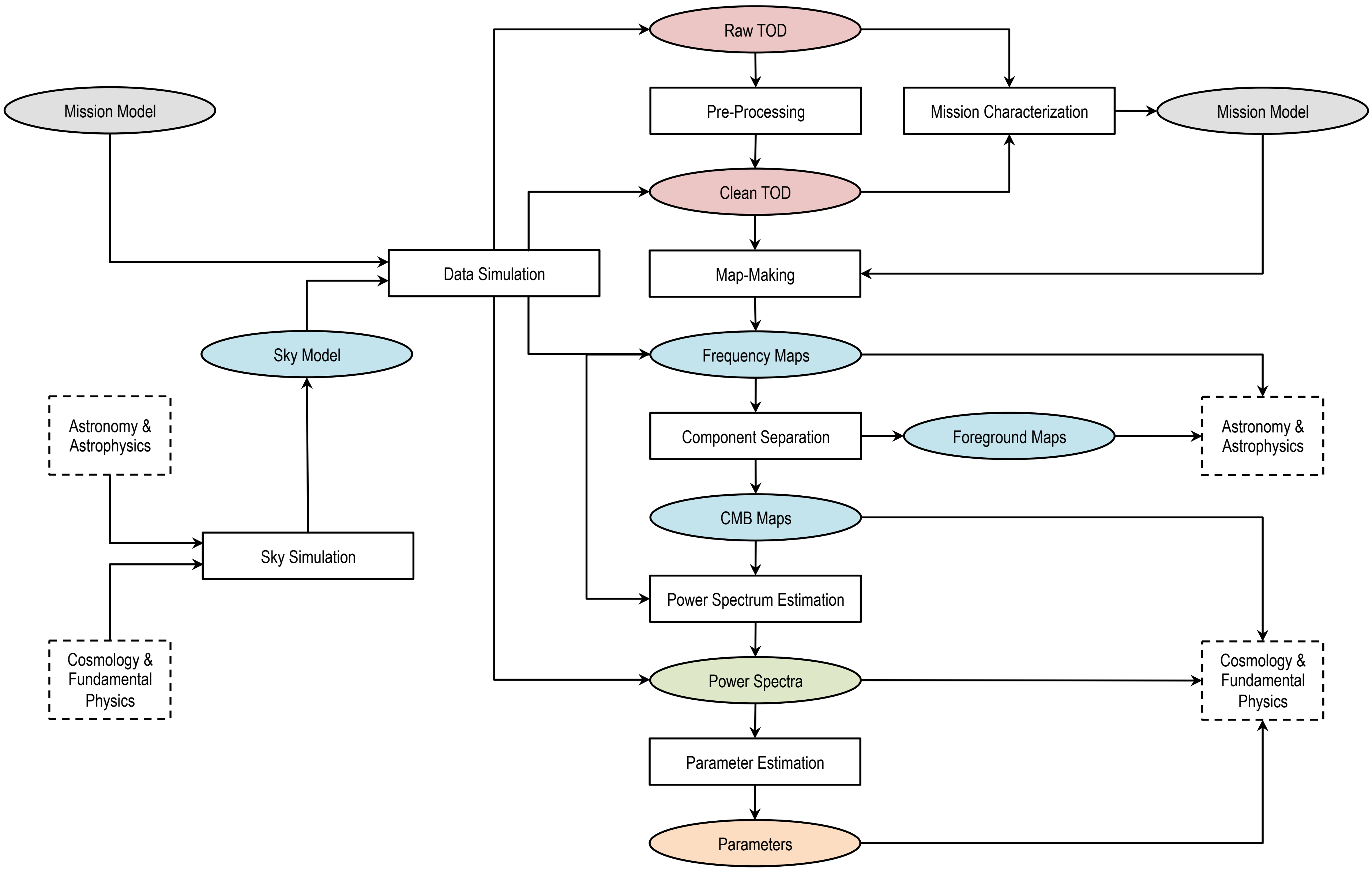
- Except in special cases, exact methods now computationally intractable.

Simulations

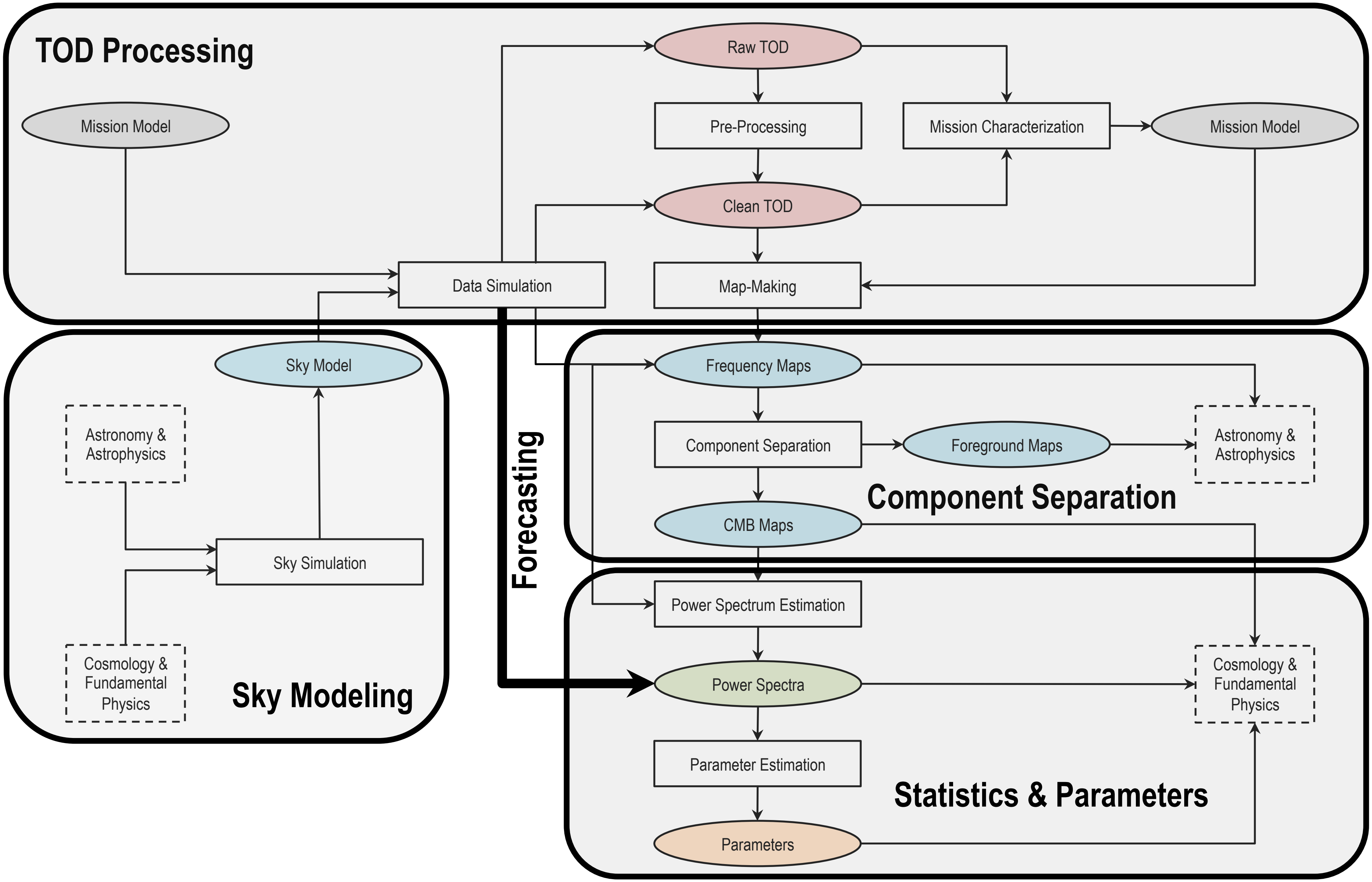


- Needed for:
  - Mission design & development
  - Analysis validation & verification
  - Data uncertainty quantification & debiasing (MC)
- From top to bottom, trade-off between:
  - computational cost
  - realism/reliability

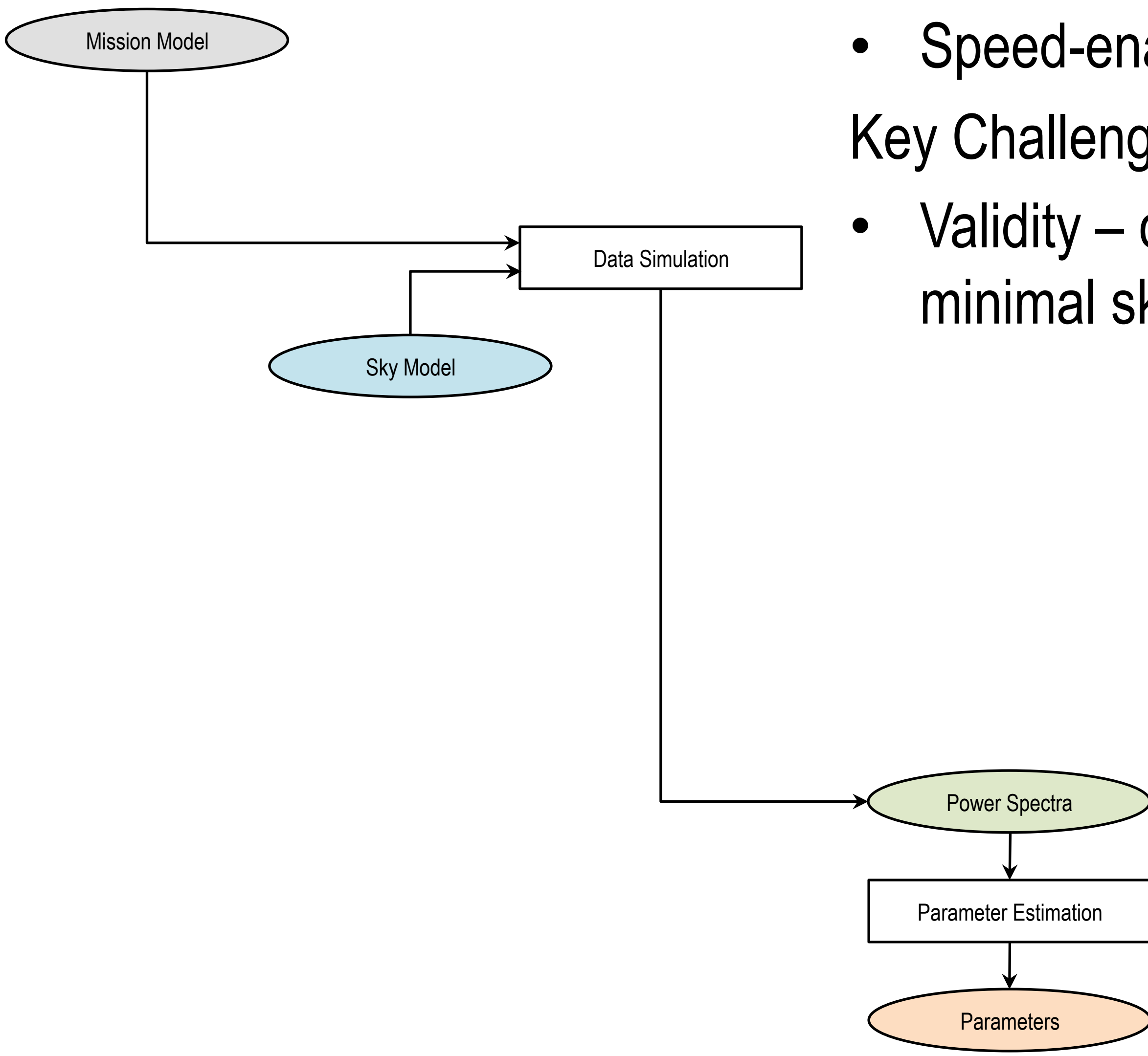
SimDA: Top Down, Wrap Around



SimDA: Sub-Domains



1. Forecasting



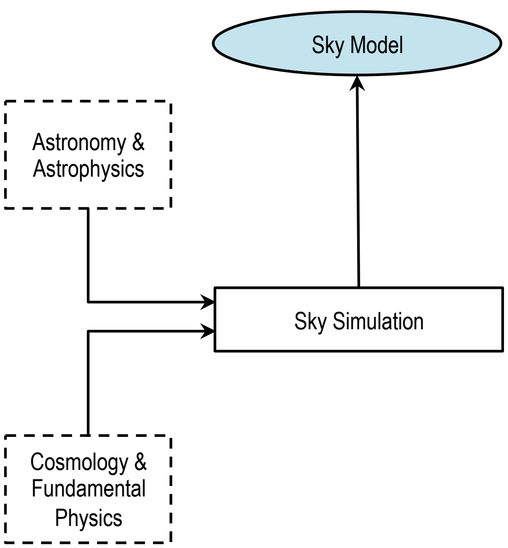
Key Strength:

- Speed-enabled breadth.

Key Challenges:

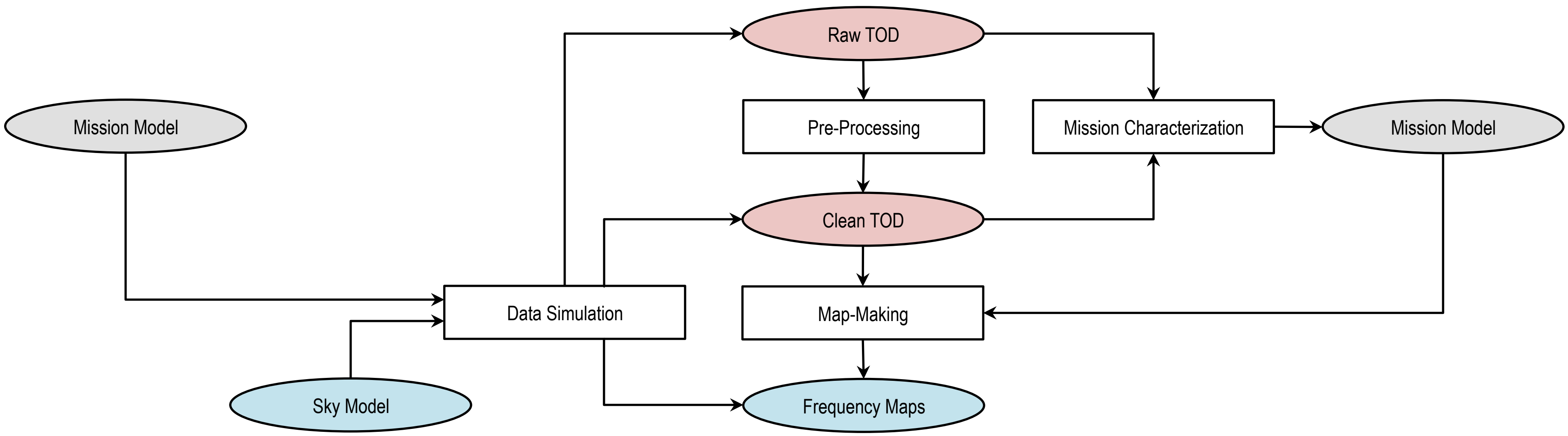
- Validity – capturing complexity in minimal sky & mission models.

2. Sky Modeling



- Key Challenges:
  - Reliability: noisy, confused, band-passed, beam-convolved input data, inc. Planck!
  - Self-consistency: eg. CMB secondaries & extra-Galactic foregrounds
  - Usability: software engineering

3. Time-Ordered Data Processing

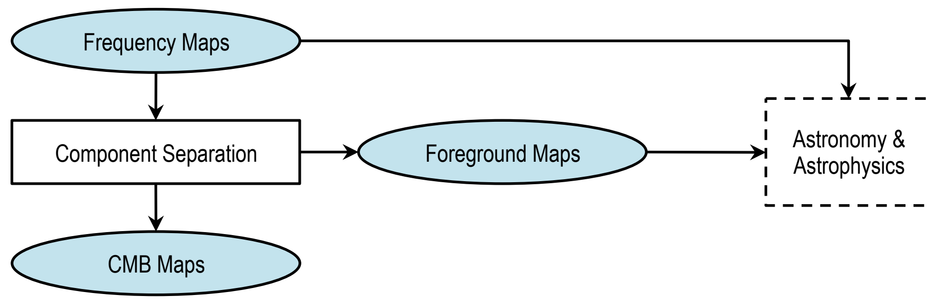


Key Challenges:

- Tractability: computational challenges defined by the TOD volume.
- Systematics: real raw data don't match any *a priori* data model.

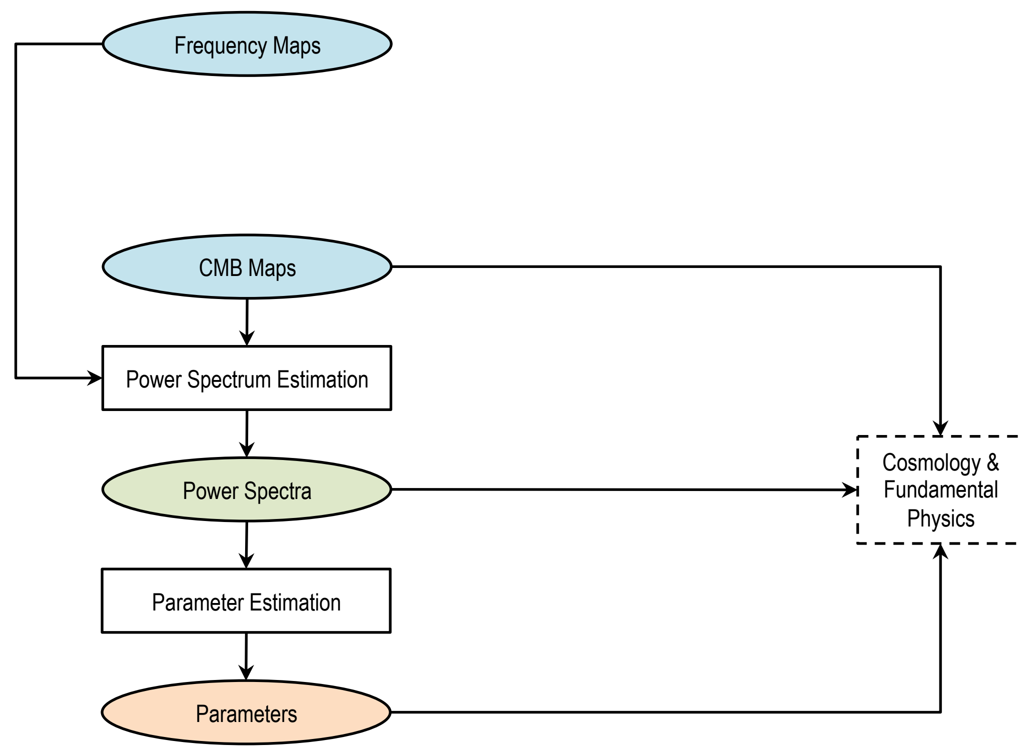
4. Component Separation

- Key Challenges:
  - Validation: are these the right algorithms for the (as yet unknown) real foregrounds?
  - Verification: are these algorithms right given (as yet flawed) simulated foregrounds?

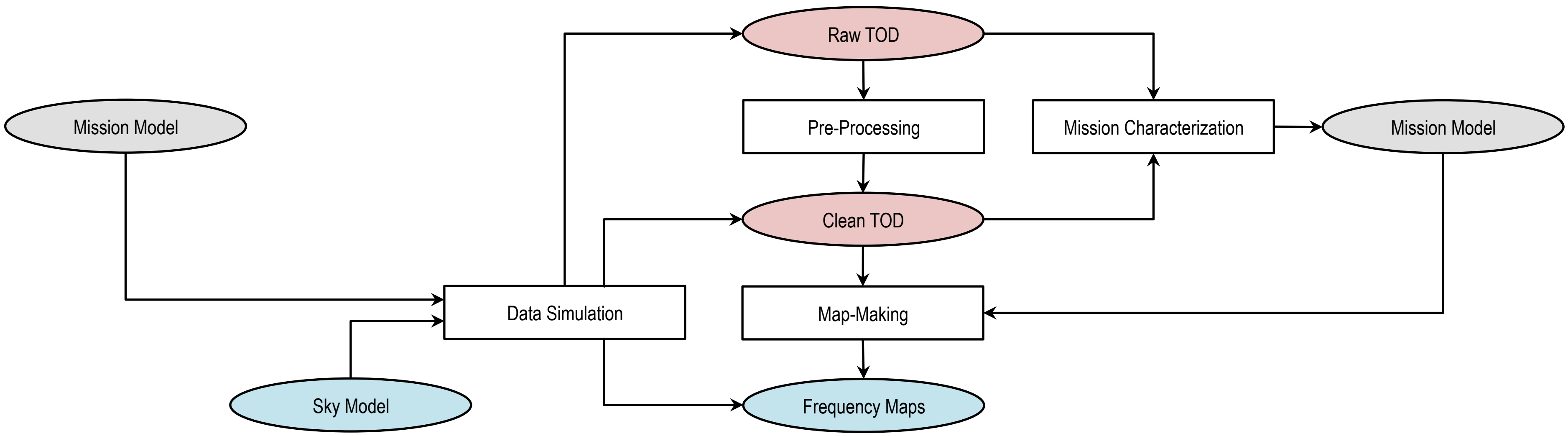


5. Statistics & Parameters

- Key Challenges:
  - Reliability: sufficiency of real data covariance approximations.
  - Tractability: disk space for many millions of MC maps.

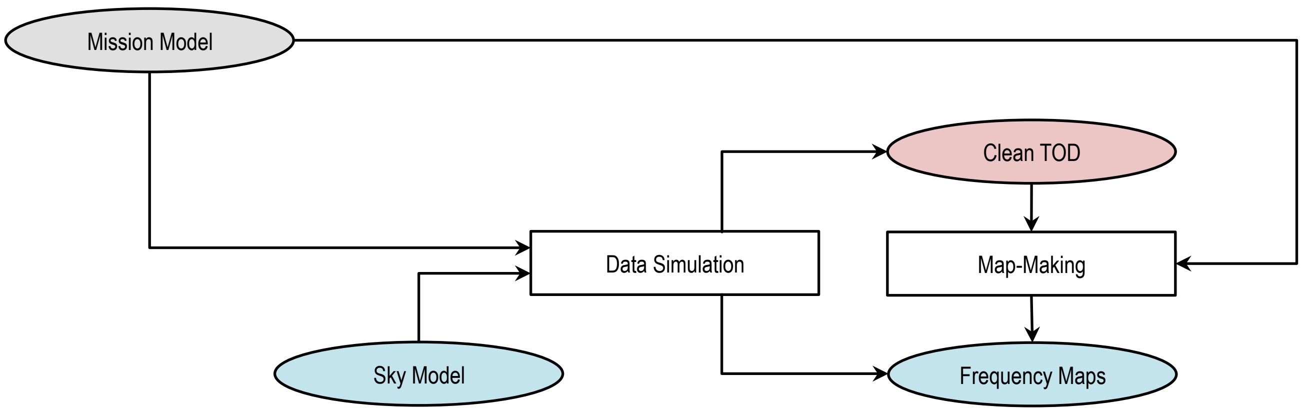


TOD Challenges



- Two bounding challenges:
  - *Tractability* for massive Monte Carlo sets.
  - *Usability* for exploratory pre-processing & mission characterization.

Massive Monte Carlos



- Operation count scales with
  - Number of MC realizations: $\mathcal{N}_{mc} \sim 10^4$
  - Number of map-makings per realization: $\mathcal{N}_{mm} \sim 10$
  - Number of PCG iterations per map-making: $\mathcal{N}_{it} \sim 10$
  - Number of operations per PCG iteration: $\mathcal{N}_{ops} \sim 10 \times \mathcal{N}_t$
- Required FLOP $\sim 10^7 \mathcal{N}_t \sim 10^{19}$ for Planck

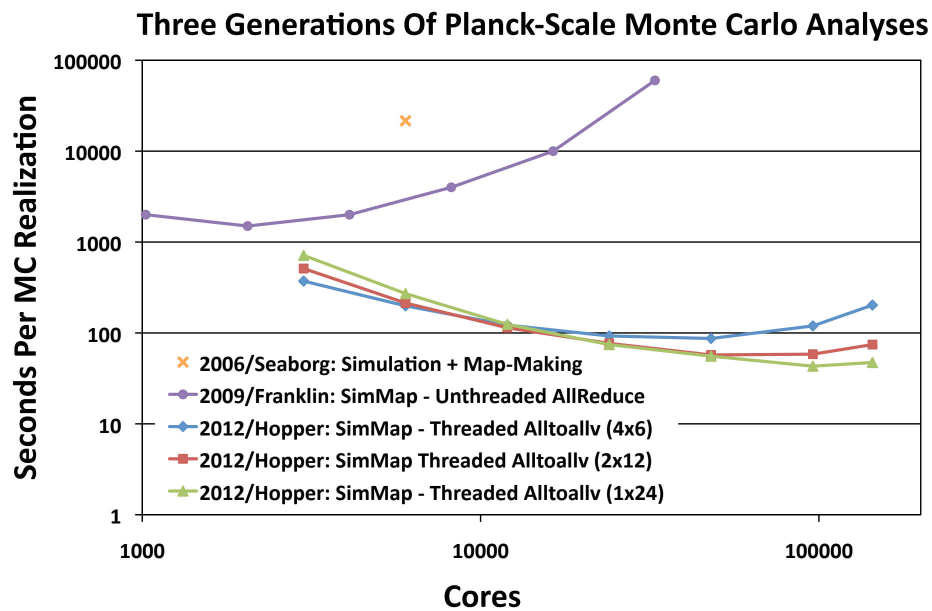
High Performance Computing

- 10^{19} FLOP ~ 10^5 CPU-years at 1% efficiency on 1GHz CPU
  $\Rightarrow$ Massive parallelism + Moore's Law growth
- Whole-data reduction
  $\Rightarrow$ Tightly-coupled cores (not grid/cloud/at-home/etc)
- Planck solution:
  - NERSC: Open-access HPC facility with long-term system upgrade plan.
    - New Top 10 system every 2-3 years
    - 6,000 users from 50 countries
  - NASA/DOE MOU guaranteed minimum annual NERSC allocation for mission lifetime:
    - In practice 1% NERSC cycles/year ~ 10^5 x Peak FLOP/s

Planck MC Efficiency

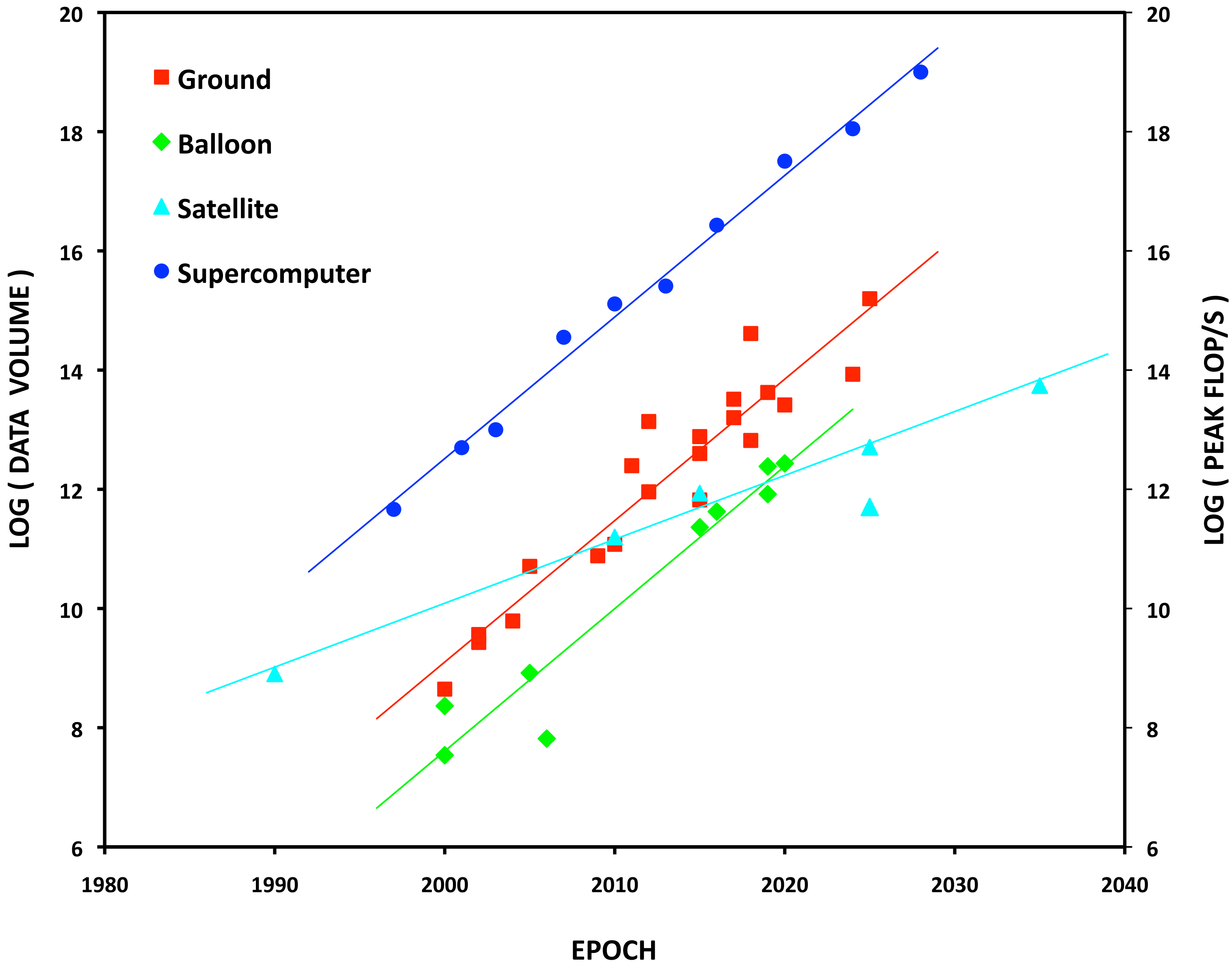
TOAST framework

- TOD input/output avoidance
  - On-the-fly simulation
  - Caching
- Communication optimization
  - MPI/OpenMP hybridization
  - Pairwise map overlap
- System- and run-specific tuning
  - eg. NUMA vs MPI



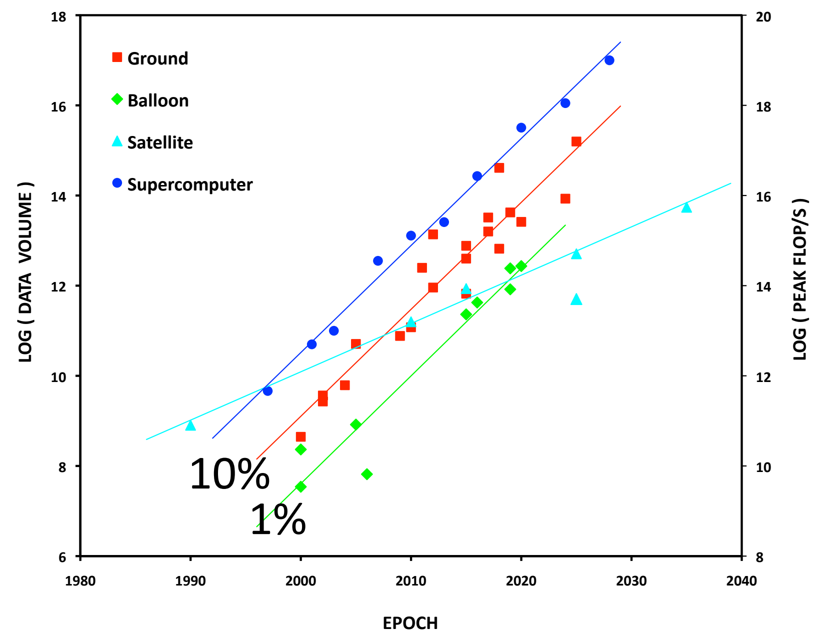
Planck 2015 results.
XII: Full Focal Plane simulations
arXiv:1509.06348

Data & HPC Growth



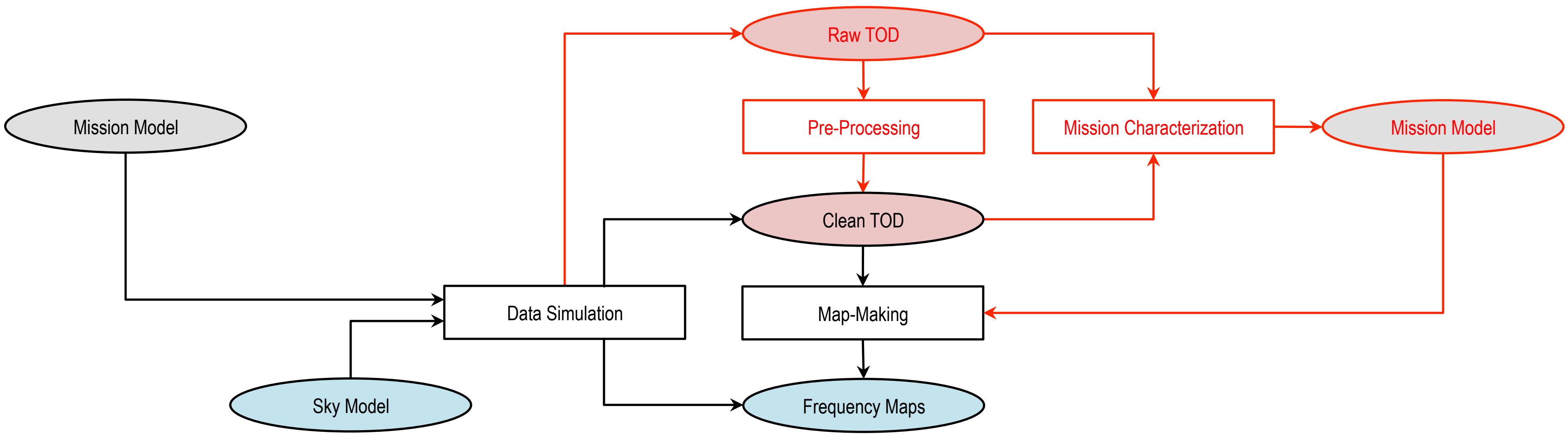
Next Generation Challenges

- Computational Efficiency
  - Required FLOP ~ $10^7 \mathcal{N}_t$
  - Available FLOP ~ 10^5 x Peak
  - Efficiency: $\varepsilon > 10^2 \mathcal{N}_t$ / Peak
    - compare suborbital & space!



- Next-generation HPC challenges
  - Energy constraints limiting Watt/FLOP (Tianhe-2 ~ Belize!)
  - More complex architectures will be harder to program efficiently
    - system heterogeneity, deep memory hierarchies, dark silicon, etc
  - End of Moore's Law

Pre-Processing & Mission Characterization



- A limiting factor for Planck has been our ability to easily and quickly
  - simulate detector-level data in full detail
  - prototype pre-processing/mission characterization algorithms.
- Must expand the framework
  - from MC Sim/Map for HPC geeks
  - to full TOD processing for all

pyTOAST

- Competing requirements:
  - Massively parallel & very efficient even on coming HPC architectures
  - Easy for non-HPC experts to adapt, extend & run
- Re-implement entire TOAST framework as open source python modules
  - Expanded developer base
  - Rapid prototyping
  - Split generic and experiment-specific elements
- Efficiency issues:
  - Start-up cost: pre-bundle libraries (eg. pyinstaller)
  - I/O avoidance: pass data between modules in memory
  - Compute efficiency: link to compiled C(++) code at key points
    - Is cython sufficient to exploit new architecture features?

Conclusions

- TOD data volumes present entirely predictable computational challenges
  - Computational efficiency must be a key design driver, *and*
  - Moore's Law presents a moving architectural target.

  Efficiency is a journey, not a destination.

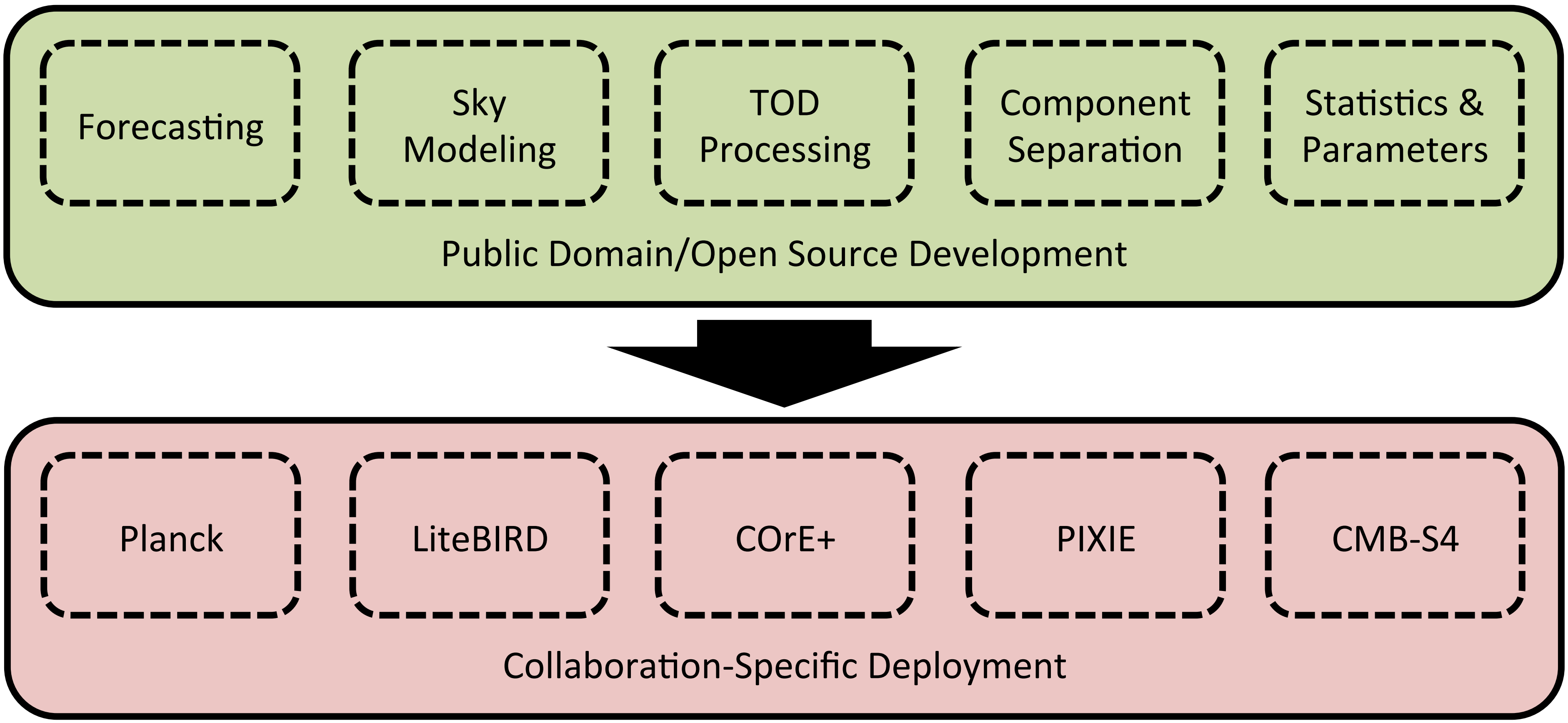
- Pure efficiency is sufficient for massive MCs, but must also be made useable for pre-processing & mission characterization.

- All next-generation CMB experiments are facing common simulation and data analysis challenges, in particular in
  - sky modeling
  - TOD processing

  and the ways in which these impact the downstream analysis domains.

A Modest Proposal

- A two-tier community-wide program:
  - developing common, generic capabilities in the public domain
  - deploying them for specific analyses within our various collaborations



TOD Processing

Birds of a Feather session

Wednesday March 9th 2016

following CMB-S4 Meeting

at Berkeley Lab

(Any interest in other sessions?)