US Contribution:
High Performance Computing

Julian Borrill

Computational Cosmology Center, Berkeley Lab

Space Sciences Laboratory, UC Berkeley

Hardware Requirements

- Sufficiently tight control of statistical and systematic uncertainties requires large numbers of realistic mission simulations.
  - *Realism* requires these be generated in the time-domain and reduced to maps with the same processing as the real data.
  - *Multiplicity* then requires **huge numbers of compute cycles**.
- Reasonable turn-around time requires **massive parallelism**.
- Reducing time-ordered data to maps requires **tightly-coupled cores**.
- Simulation products need to be **stored/archived** and then be **made available** to the collaboration – and eventually the community – along with sufficient **disk & cycles** for post-processing.
- Critical resources must be available for the **lifetime of the mission**.

⇨ **HIGH PERFORMANCE COMPUTING CENTER**

Planck @ NERSC

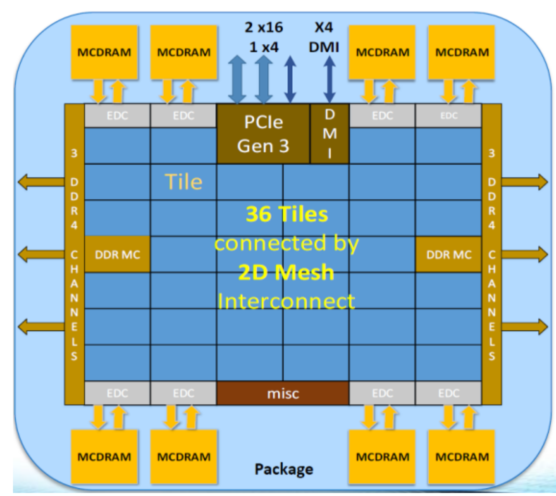
- **Cycles**
  - 1% of total available cycles each year for 15 years:
    - O(100K) CPU-hrs in 2000 => O(100M) CPU-hrs in 2015.
- **Storage**
  - Temporary & permanent disk, tape archive.
  - Project-specific disk, file-group & user for data management.
- **Access**
  - Unlimited* user accounts.
  - High-bandwidth connectivity (ESNet).
- **Longevity**:
  - New Top 10 supercomputer every 3 years.
  - NASA/DOE MoU guarantees access through mission lifetime.

Symbiosis

- NERSC benefits from Planck:
  - Early adopters of new technology & processes
  - Science code for procurement tests
  - Inspiration for many project services
- Planck benefits from NERSC:
  - Elevated queue priority during face-to-face hackathons
  - Early access to new systems
  - Buy-in to exceptional levels of service
    - Planck cluster, Carver cabinet; Project disk space
- CMB community benefits from Planck@NERSC
  - Community allocation for post-Planck experiments
  - Legacy Project disk

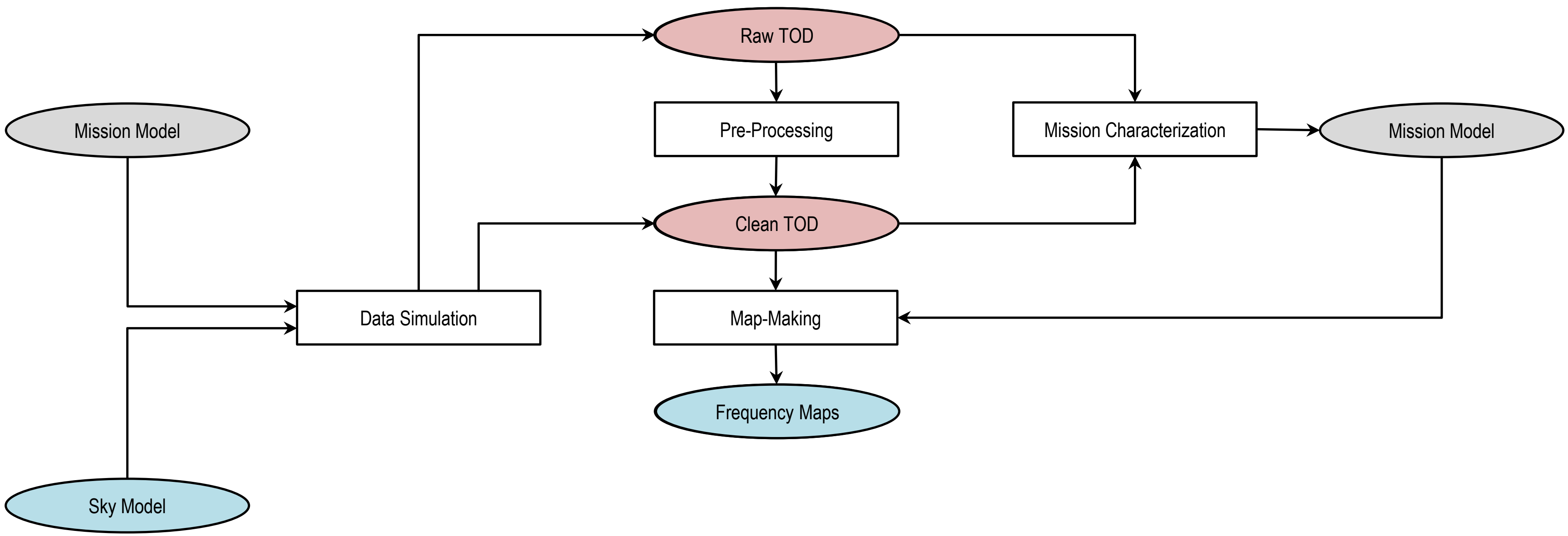
HPC Hardware Resources for COrE

- *Cori* : KNL-based Cray XC system arrives in summer 2016
  - Also *Theta/Aurora* at ALCF, likely future PRACE system(s).
- 10x increase in available cycles, but much harder to use efficiently.
- Many low energy cores/node
  => 10-100x increase in threads per node
- Deeper memory hierarchy, including per-cabinet “burst buffer”
  => complex, hierarchical, cache management.



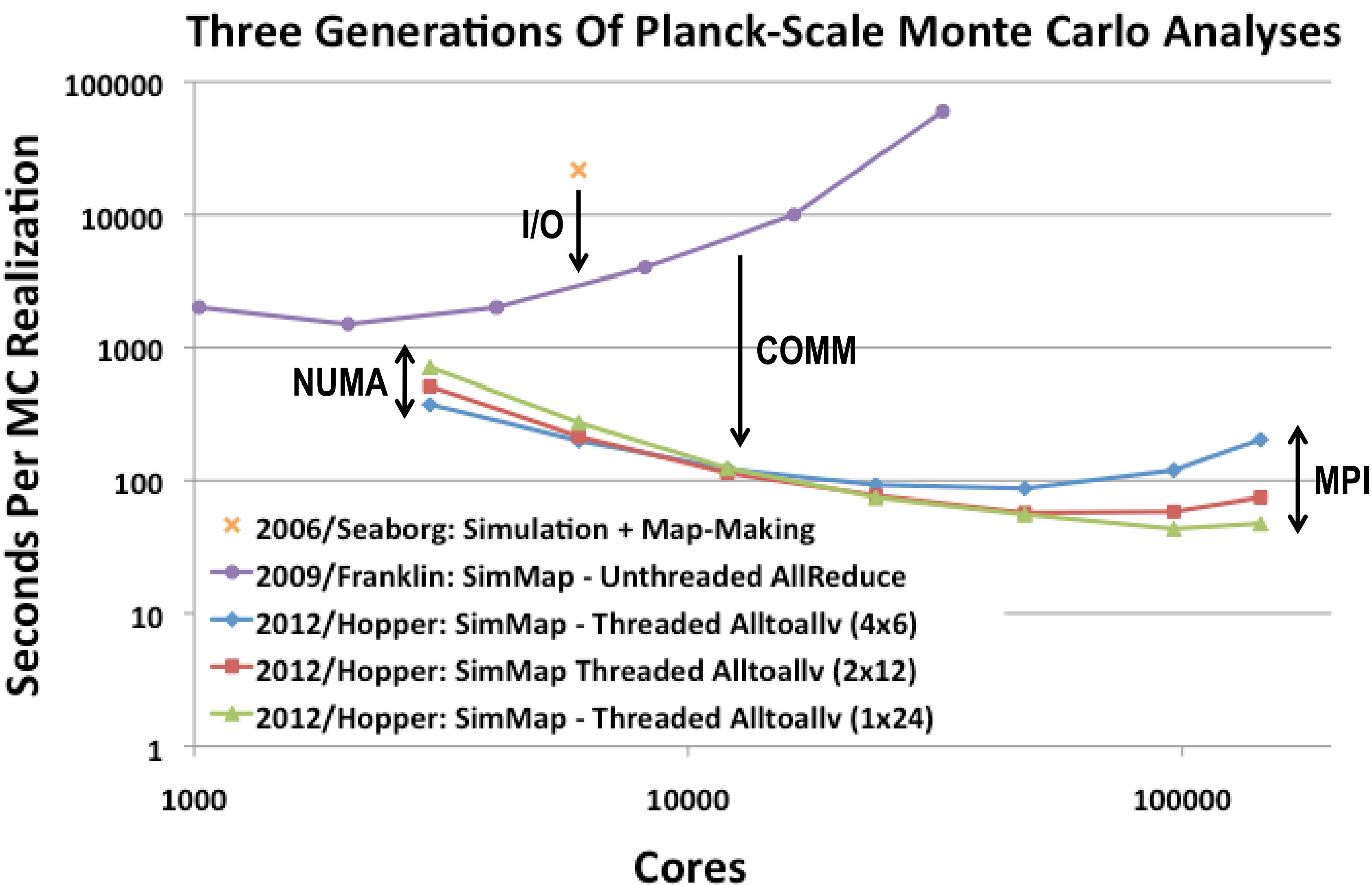
Software Requirements

- Computational cost is dominated by time-domain processing
  – Simulation, Pre-Processing, Characterization, Map-Making.



- These steps must run efficiently on state-of-the-art HPC architecture
  – Multiple architecture instances during mission lifetime.

Optimization

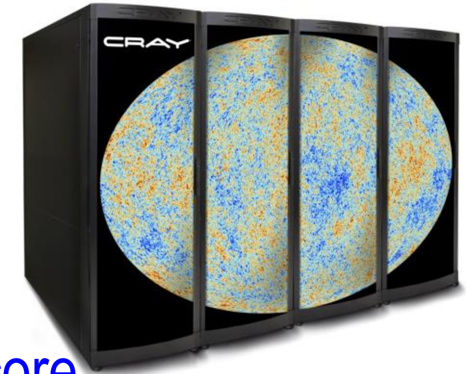


TOAST @ C^3

- Time-Ordered Astrophysics Scalable Tools
  - Massively parallel data distribution
  - On-the-fly simulation capability
    - Auto- and cross-correlated noise
    - 4π beam convolved sky signal
  - Feeds data to generic or experiment-specific TOD processing

Version 1	Version 2
Library of TOAST functions called by compiled executables: - iscalm2tod - MADAM-TOAST	Python wrapping, calling python or compiled library functions: - libCONVIQT - libMADAM

HPC Software Resources for COrE



- Base framework & generic tools/scripts
  - Public git repo https://github.com/hpc4cmb/toast
- Experiment-specific extensions & scripts:
  - Private git repo https://github.com/hpc4cmb/toast-core
  - Also toast-planck, toast-litebird, toast-cmbs4, etc
- Planned additions/extensions:
  - **Xeon Phi KNL port/optimization**
  - On-the-fly band-pass integration
  - HWP-varying beam, bandpass
  - Multichroic/multiplexed cross-talk
  - Planet/variable source observations
  - (Atmosphere & ground-pickup)

Help Wanted!

Example: Hit & Condition Maps

Single boresight detector, 1 year survey

FAST SCAN | SLOW SCAN | LiteBIRD

Note factor of 10 smaller scale.