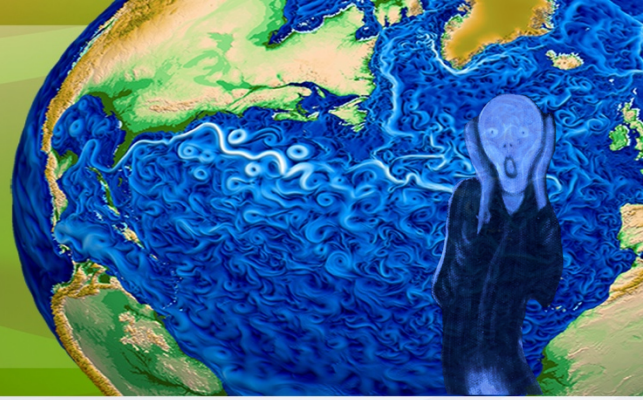
E3SM
Energy Exascale Earth System Model



Nonhydrostatic Atmosphere NGD Update

Contributors: Peter Caldwell, Andy Salinger, Luca Bertagna, Hassan Beydoun, Peter Bogenschutz, Andrew Bradley, Conrad Clevenger, Aaron Donahue, Chris Eldred, Meredith Fossitt, Jim Foucar, Chris Golaz, Oksana Guba, Ben Hillman, Rob Jacob, Jeff Johnson, Noel Keen, Jayesh Krishna, Wuyin Lin, Weiran Liu, Kyle Pressel, Balwinder Singh, Andrew Steyer, Mark Taylor, Chris Terai, Paul Ullrich, Danqing Wu, Xingqiu Yuan

Work from LLNL is performed under the auspices of the US DOE by LLNL under Contract DE-AC52-07NA27344.
UCRL:LLNL-PRES- 816461

U.S. DEPARTMENT OF ENERGY

What is it?

1. New atmosphere model written in C++/Kokkos
   a. Kokkos allows CPUs, GPUs, and future architectures to use the same code
   b. **Focus on simplicity** = less code to write/easier interpretation
   c. Code-name: Simple Cloud Resolving E3SM Atmosphere Model (SCREAM)

2. Target is exascale computers (which require lots of parallel work)
   a. Initial focus is 3 km global **"convection-permitting"** simulations
   b. Will eventually also use for large low-resolution **ensembles**

3. Initial implementation will use prescribed aerosol
   a. EAGLES project is dedicated to creating a prognostic aerosol scheme for SCREAM

Why do This?

1. E3SM needs to run on modern architectures
   a. All new DOE Leadership Class machines use GPUs
   b. Porting the existing E3SM atmosphere model was deemed too hard

Fig: A tropical cloud system overlain by a typically-sized global model grid box

2. Unlocks grand challenges in Earth system modeling:
   a. Explicitly resolving convection breaks the “parameterization deadlock”
   b. Initial condition and perturbed physics ensembles quantify predictive uncertainty, a critical ingredient for actionable prediction

Implementation:

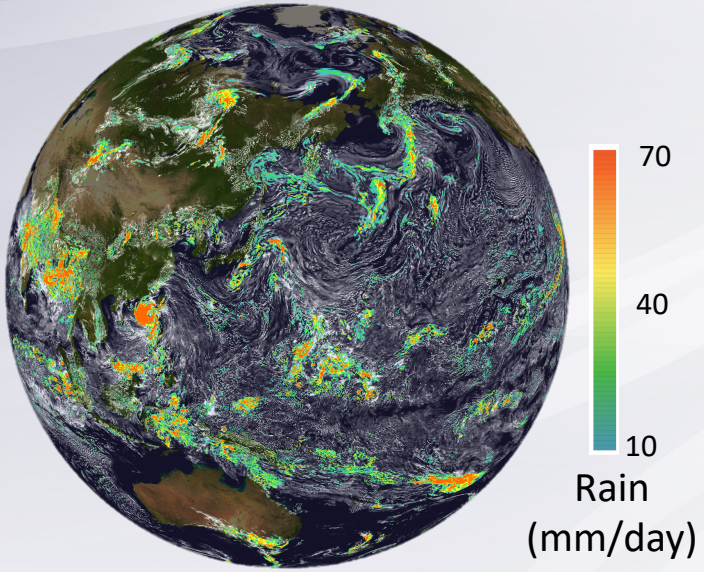


Fig: cloud and precip from SCREAMv0 prototype

Development is occurring in 2 overlapped stages:

1. **SCREAMv0**: F90 implementation using existing EAM infrastructure

   status: ~done

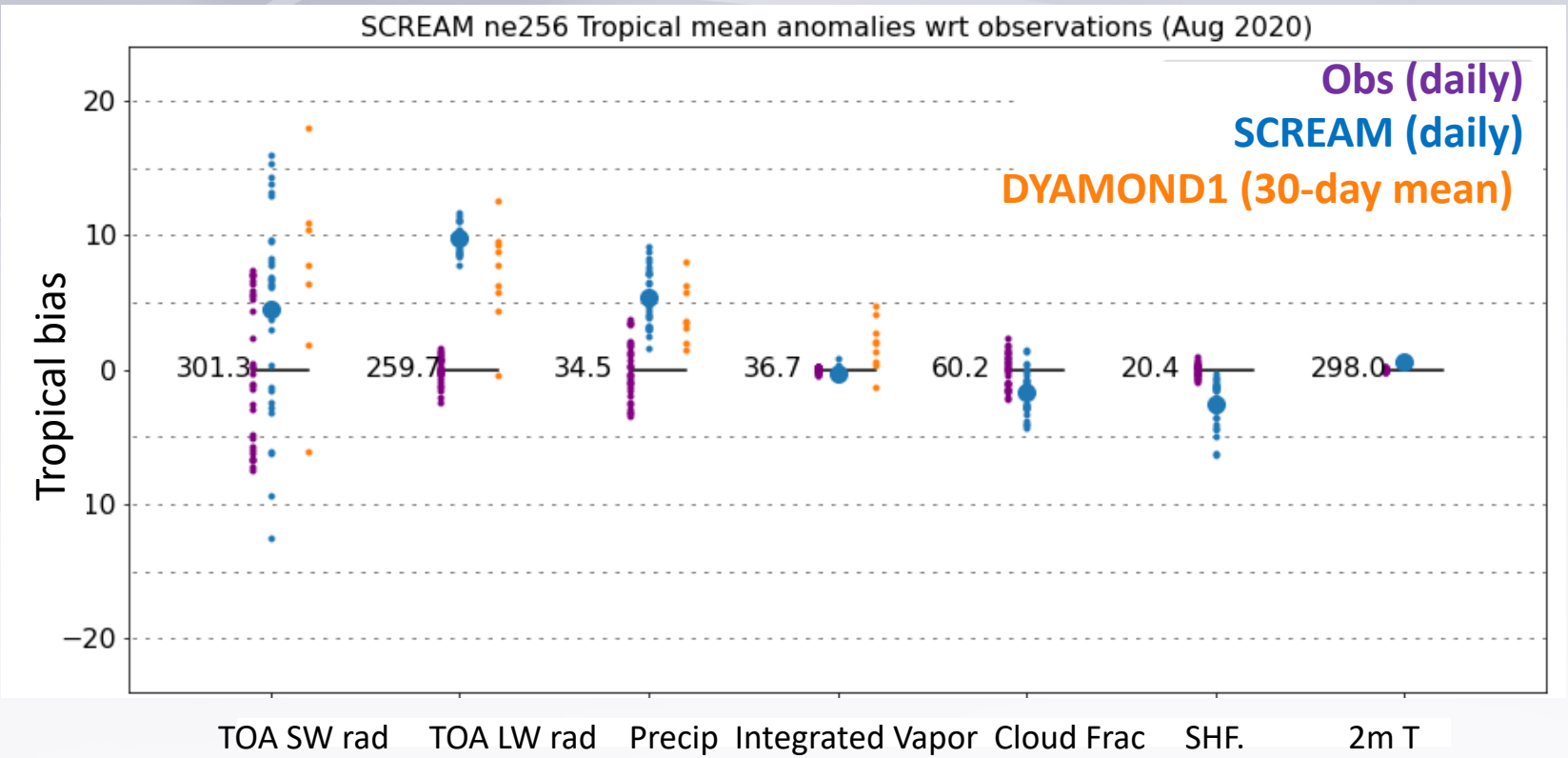
   runs: starting 40 day 3 km fixed-SST simulation for DYAMOND2 intercomparison very soon

2. **SCREAMv1**: C++ version using new infrastructure

   status: Finalizing most pieces now (details later). Still need to stitch them together

   runs: planning to start “Cess” runs on Summit next summer. Just received INCITE allocation

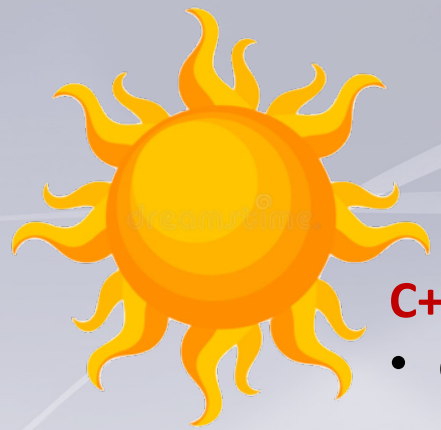
v0 Behavior



- v0 skill is comparable to other DYAMOND models
- 12 km→3 km Δx has small impact except TOA SW increases by 5 W m^{-2} (not shown)
  - Other DYAMOND1 models have similar low cloud resolution sensitivity

Fig: Tropical mean bias for DYAMOND1 period (Aug 1-Sept 10, 2016) from SCREAMv0 at 12 km (ne256) compared to runs from other modeling centers.

v1 Progress



C++ radiation kernels done
- connecting to coupler now

Process coupler is ~done
- Finalizing I/O now

Prescribed Aerosol is ½ done:
- cloud condensation nuclei done
- Prescribed aerosol optics is in-progress

SHOC turbulence/ macrophysics is ½ done

Nonhydrostatic Spectral Element C++ dycore done
- connecting to coupler now

P3 Microphysics done
- Needs better testing + documentation
- connecting to coupler now

- Coupling to **surface** (rest of E3SM) ~done

C++/Kokkos

Writing a global atm model in C++/Kokkos is a grand experiment

- Code *is* more complex but still readable
- Future proofing is working: Kokkos already runs on Frontier and Aurora testbeds
- Performance so far fulfills our hopes: 1 SYPD for dycore on Summit (see fig)!
  - Semi-Lagrangian advection will make even faster, but adding physics will slow model down

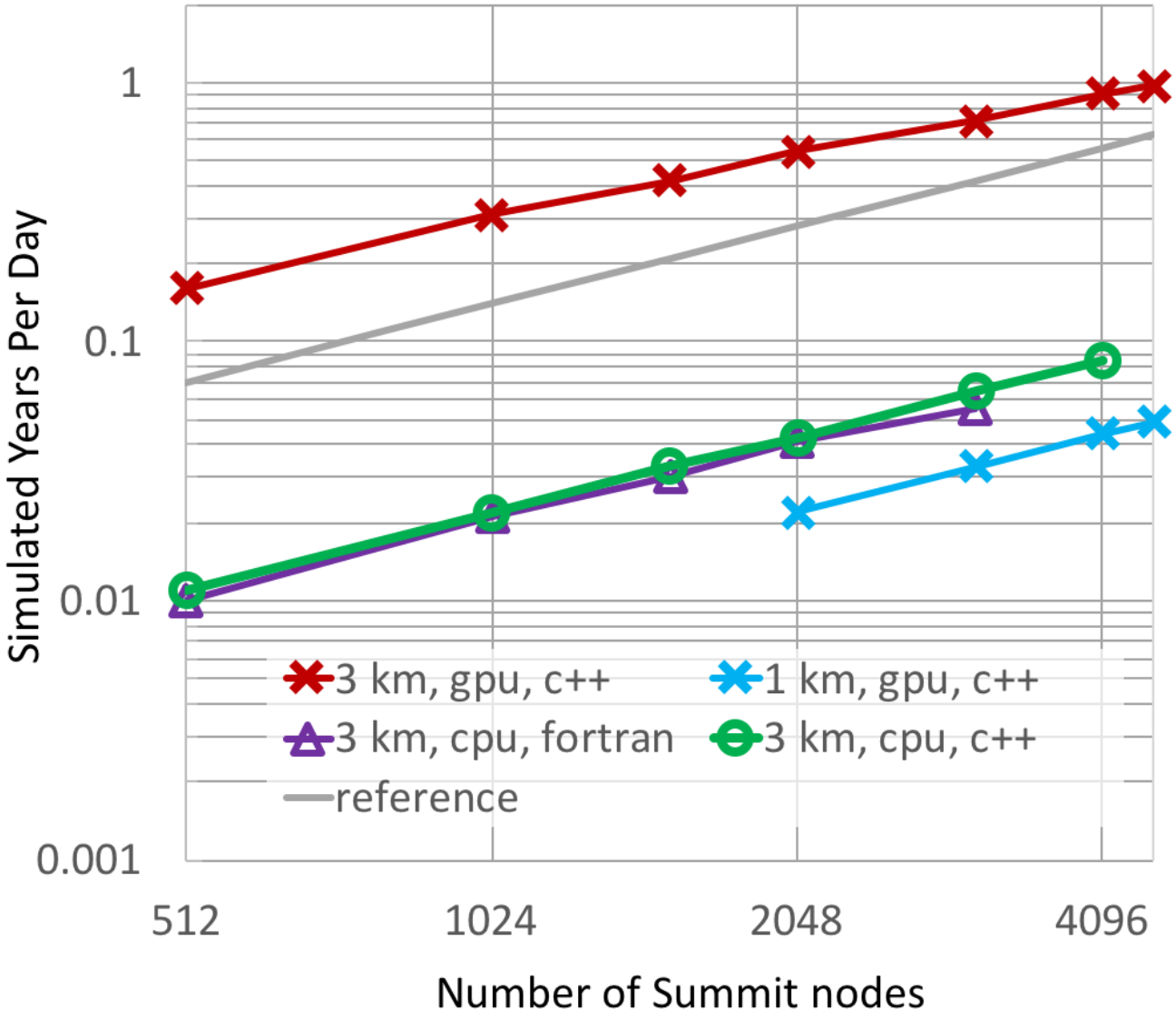
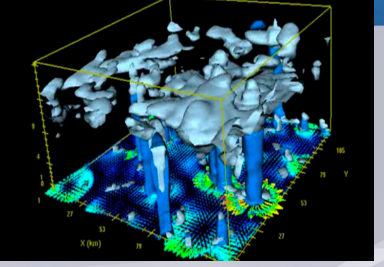
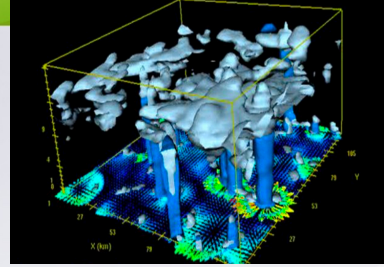


Fig: Nonhydrostatic C++ dycore-only NGGPS timings at ne3072 (blue) and ne1024 (other colors). From Bertagna et al., Supercomputing 2020





Tools

1. **Doubly Periodic SCREAM**: A planar, limited-area implementation should be available soon for v0, eventually for v1
2. **Calling functions from Python**: physics functions can be called from python using F2PY for v0 and (eventually) c bindings for v1
3. **Forecasts**: short forecasts can be performed using CAPT, Betacast, or HICCUP-derived tools
4. **Regionally-Refined Model**: resolution can be focused on a region of interest with other areas nudged to obs
5. **e3smplot**: high-res output can be plotted on its native grid + compared to appropriate observational data

Science Plans

This model unlocks many interesting questions. Some we are focused on are:

1. What weather/climate features are we good/bad at?
   - will be answered by v0 DYAMOND2 runs
2. What is climate sensitivity with resolved deep convection?
   - will be answered by multi-year v1 “Cess” runs: radiative difference between runs with prescribed current-climate and +4 K SST
3. At what resolution do nonhydrostatic effects become important? Postdoc Weiran Liu is answering this
4. What controls tropical anvil cloud extent? Postdoc Hassan Beydoun is answering this (see graphic)

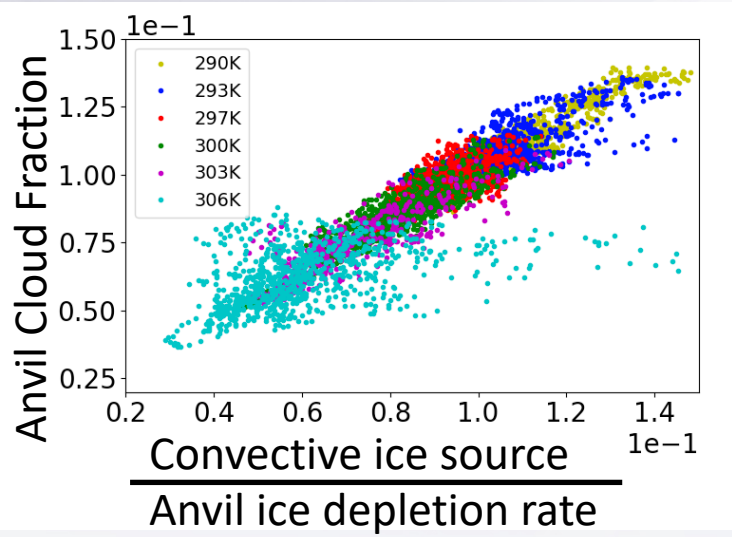


Fig: Anvil cloud fraction in v0 is predicted well by the convective source divided by microphysical depletion rate. Colors indicate SST in radiative-convective equilibrium runs. Each dot is a 3 hr average.

Collaborations

1. within E3SM:
   a. Overhauled and wrote documentation for regional refinement capability for E3SMv2
   b. SCREAMv0 provided bugfixes to/got bugfixes from v2 team
   c. SCREAM supplied P3 and SHOC to atm v3 NGD and received bugfixes
2. with external projects
   a. EAGLES aerosol: Provided code and design plans. Collaborated on some infrastructure
   b. Exascale Computing Project: Received useful tools
   c. LLNL SFA's ASR component is retooling with a focus on SCREAM

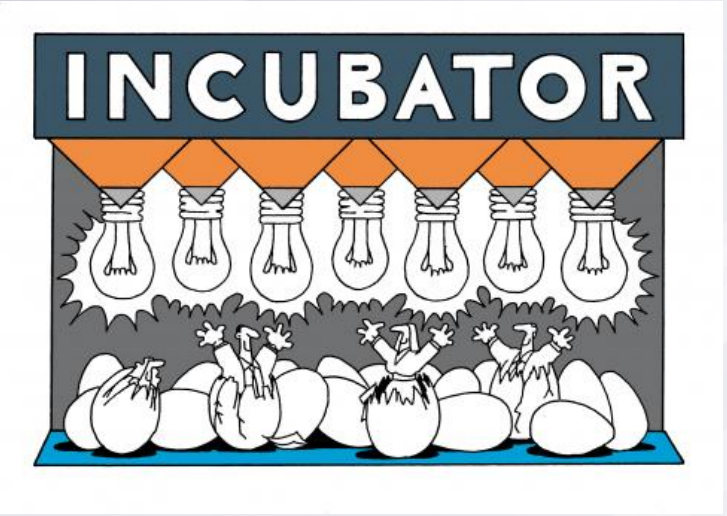


Fig: SCREAM has fewer collaborations because it is meant to be a tech incubator

Building collaborations will be an important theme of E3SM phase 3

Extra Slides

Code/Data Sharing

The SCREAM repo is world-readable so anyone can use it at any time, but…

1. DYAMOND2 data and corresponding SCREAMv0 code snapshot will be available in January
   a. output available NERSC HPSS and the DYAMOND server in Germany

2. Doubly-Periodic/Limited-Area config should be available in Feb(?)

3. We are trying to create good documentation for our tools as we go but have no bandwidth to help external users

Coding

SCREAM will be rewritten from scratch in C++:

- Allows use of Kokkos library, which abstracts on-node parallelism
  - single code runs efficiently on CPUs, GPUs, etc
  - Unlocks more parallelism…
  - But results in more complex code

Original F90

```
kloop_sedi_c2: do k = k_qxtop,k_qxbot,-kdir

   qc_notsmall_c2: if (qc_incld(k)>qsmall) then
      !-- compute Vq, Vn
      call get_cloud_dsd2(qc_incld(k),nc_incld(k),mu_c(k),rho(k),nu,dnu,   &
           lamc(k),tmp1,tmp2,lcldm(k))

      nc(k) = nc_incld(k)*lcldm(k)
      dum = 1._rtype / bfb_pow(lamc(k), bcn)
      V_qc(k) = acn(k)*bfb_gamma(4._rtype+bcn+mu_c(k))*dum/(bfb_gamma(mu_c(k)+4._rtype))
      V_nc(k) = acn(k)*bfb_gamma(1._rtype+bcn+mu_c(k))*dum/(bfb_gamma(mu_c(k)+1._rtype))

   endif qc_notsmall_c2
   Co_max = max(Co_max, V_qc(k)*dt_left*inv_dzq(k))

enddo kloop_sedi_c2
```

Ported to C++/Kokkos

```
Kokkos::parallel_reduce(
  Kokkos::TeamThreadRange(team, kmax-kmin+1), [&] (int pk_, Scalar& lmax) {
    const int pk = kmin + pk_;
    const auto range_pack = scream::pack::range<IntSmallPack>(pk*Spack::n);
    const auto range_mask = range_pack >= kmin_scalar && range_pack <= kmax_scalar;
    const auto qc_gt_small = range_mask && qc_incld(pk) > qsmall;
    if (qc_gt_small.any()) {
      // compute Vq, Vn
      Spack nu, cdist, cdist1, dum;
      get_cloud_dsd2<false>(qc_gt_small, qc_incld(pk), nc_incld(pk), mu_c(pk), rho(pk), nu, dnu, lamc(pk), cdist,
      nc(pk).set(qc_gt_small, nc_incld(pk)*lcldm(pk));
      dum = 1 / (pack::pow(lamc(pk), bcn));
      V_qc(pk).set(qc_gt_small, acn(pk)*pack::tgamma(4 + bcn + mu_c(pk)) * dum / (pack::tgamma(mu_c(pk)+4)));
      if (log_predictNc) {
        V_nc(pk).set(qc_gt_small, acn(pk)*pack::tgamma(1 + bcn + mu_c(pk)) * dum / (pack::tgamma(mu_c(pk)+1)));
      }

      const auto Co_max_local = max(qc_gt_small, -1,
                                    V_qc(pk) * dt_left * inv_dzq(pk));
      if (Co_max_local > lmax)
        lmax = Co_max_local;
    }
}, Kokkos::Max<Scalar>(Co_max));
team.team_barrier();
```

v0 Behavior

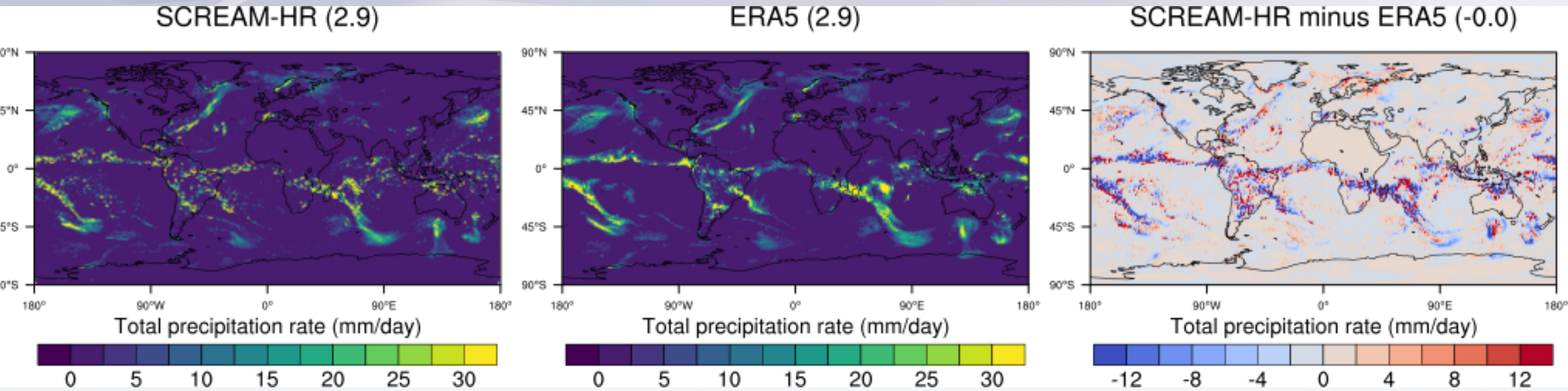


Fig: Precip averaged over the first 2 days of a 3 km (ne1024) DYAMOND2 simulation (Jan 20th-22nd, 2020). Numbers in parentheses in the title are global averages.

- Precipitation matches ERA5 very well in global mean and spatial pattern
- SCREAM slightly underpredicts precip in strong storms
- SCREAM convection has too much small-scale noise