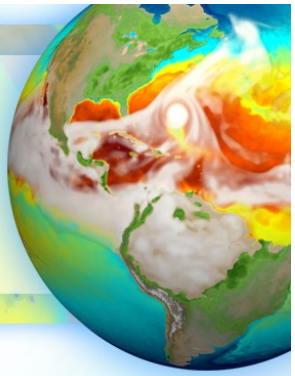
Performers Performing Performance

E3SM 2021 Summer All-Hands



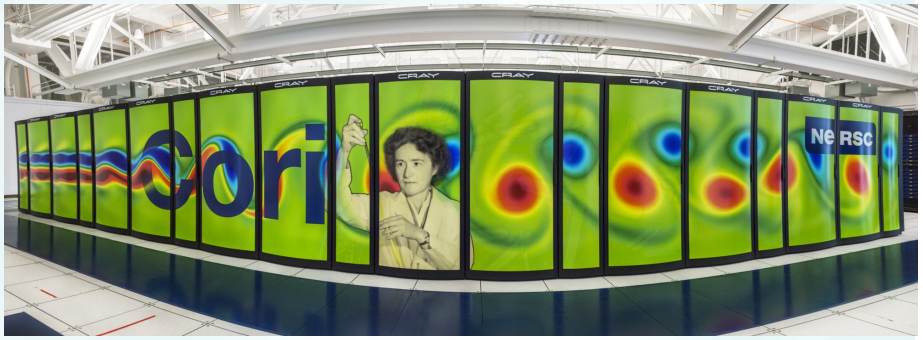
Phil Jones, Sarat Sreepathi on behalf of Performance Group:

Oksana Guba, Noel Keen, Youngsung Kim, Jayesh Krishna, Azamat Mametjanov, Mark Taylor, Matt Turner, Pat Worley, Min Xu

Also: Nichols Romero, Xingqiu Yuan

Goals

- Optimizing throughput for campaigns
  - Chrysalis, CompyMcNodeFace, Anvil
  - Debugging, load-balancing
- Preparing for future architectures
  - Frontier
  - Perlmutter
  - Aurora
- Creating infrastructure for performance
  - Metrics: Standard benchmarks
  - Monitoring and Analysis Tools



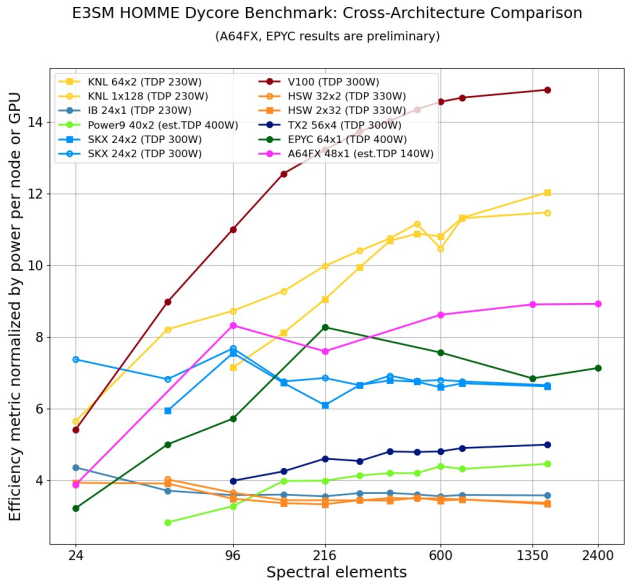

"They're here."

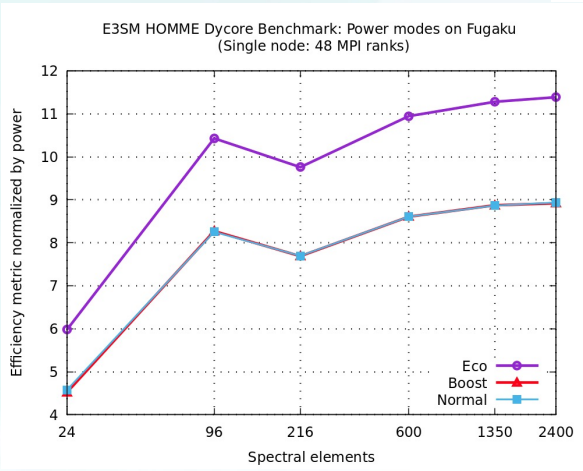
New machines/early access now available

- Perlmutter Phase I (installed, NESAP)
  - Epyc Milan CPU/4 Nvidia A100 GPU
  - Slingshot/Dragonfly
  - SS (Flash) – no disk, Lustre
  - CUDA, OpenACC/OpenMP
  - Phase II: more CPU-only nodes, more NICs
- Frontier (delivery this year, early sys avail)
  - 1 AMD Epyc CPU/ 4 AMD Radeon GPU
  - Slingshot/Dragonfly
  - Flash/Disk hybrid/tiered Lustre
  - HIP, OpenMP Offload
- Aurora (2022, systems in JLSE already avail)
  - 2 Intel Xeon CPU/ 6 Intel Xe GPUs
  - Slingshot/Dragonfly
  - DAOS (Distrb Async Obj Store), Lustre
  - oneAPI, SYCL, OpenMP
- Now is the opportune moment

Fugaku

- A64FX
  - Evaluation of “custom” approach
  - Modified ARM architecture
  - Environment still immature
- HOMME Dycore Efficiency
  - 1k elements*nsteps/s/node or GPU
  - Good, but modest
- Other custom efforts
  - Project 38
  - Grace (Nvidia)
  - Most focused on memory
- Hybrids still most likely





Strategery remains…

- Kokkos/YAKL for atmosphere
  - C++ performance-portable
  - See Peter's presentation
- Fortran+Directives for ocean/ice
  - Merging these from ECP version
  - Transition to C++, API/PM TBD in NGD
  - Issues with performance portability
- Land
  - Fortran+directives
  - Useful primarily for N. American high-res
- Optimal use cases
  - MMF, highest res
- Just about there…

In the meantime: Recent Issues in Production

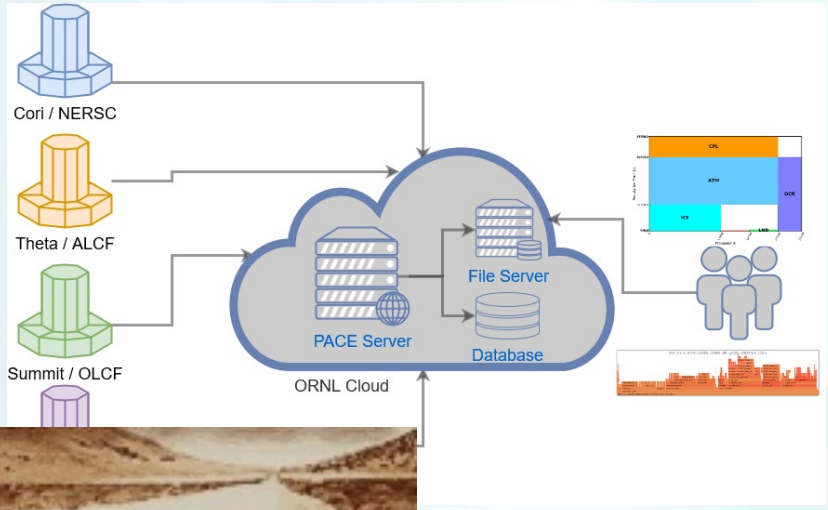
- Chrysalis
  - Hardware: bad switches, cables, network configuration (also Anvil)
  - Software: kernel settings (CPU power), MPI bugs, hyperthreading
  - Now all fixed – 40 SYPD! , thanks to Az, Jayesh, et al.
- Debugging/Optimization
  - Usual PE layout tuning
  - I/O tuning
  - Debugging non bfb issues in several cases
  - Fixing OpenMP issues
  - See Code Review…
- MPAS-seaice in F cases to replace CICE
  - Eliminated excess I/O, remaining costs are mesh/partition related
- Others…

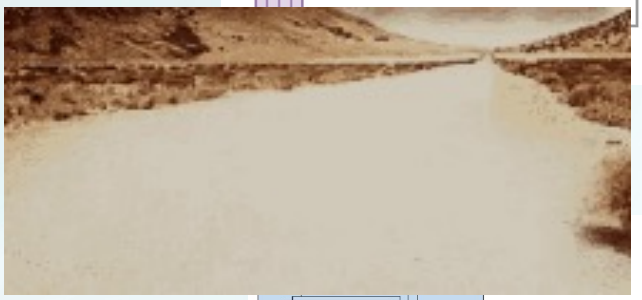
Coder Integrity / Coder Suppression

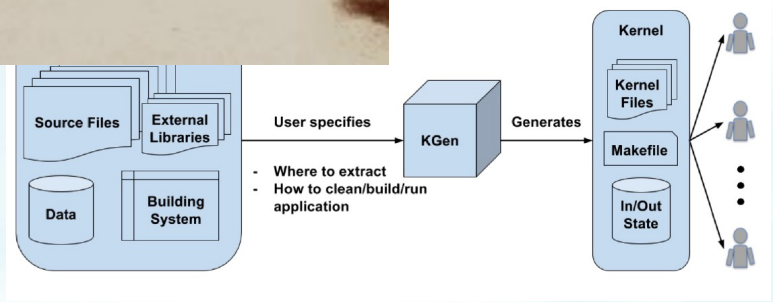
- Coder fraud, Coding errors
  - Registering lots of new coders
  - Audits, observers (testing), esp. coupled mode
  - Absentee PRs (b4b when turned off)
  - Gaps in process
  - Contributed to delay in calling v2
- Code Review Deep Dive
  - Improve process for coder integrity
  - Not so heavy that it contributes to coder suppression
- From Performance perspective
  - Let us help early on w/ design
  - Need to evaluate performance
  - GPU-ready code, OpenMP, etc

The Infrastructure Plan

- Metrics gathering
  - Standard E3SM benchmarks (v2)
  - CPMIP, AGU
  - New: Memory diagnostics (rootPE, per PE)
  - New: More detail in I/O timers, per file, per component, r/w
- I/O
  - ADIOS 2, read capability, CIME mods
  - Also more disk, filesystem policies on Compy
- PACE
  - Integrate, summarize I/O metrics
  - Simulation context to detect outliers
  - Performance recommendations
  - Provenance (with IG)
  - Naming standards help with searching, context
- Machine usage monitoring
  - Anvil underutilized! New dashboard image?
  - How well are we using, update queue/usage policies
- Evaluating other tools
  - Byfl, Nsight, HPC toolkit, etc.
- New – kernel extraction







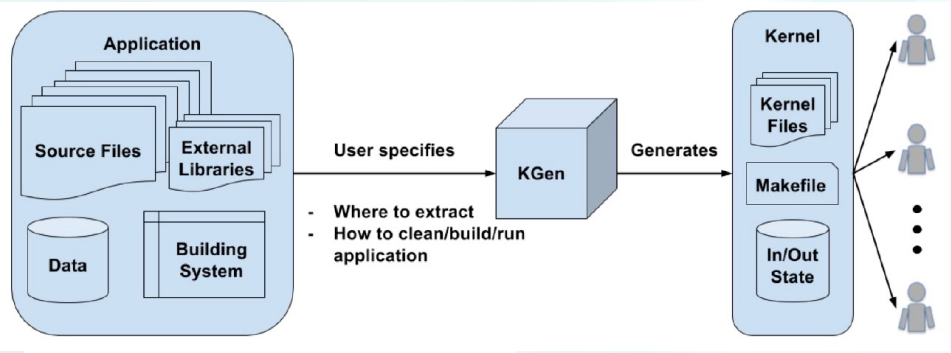
Kernel Extraction

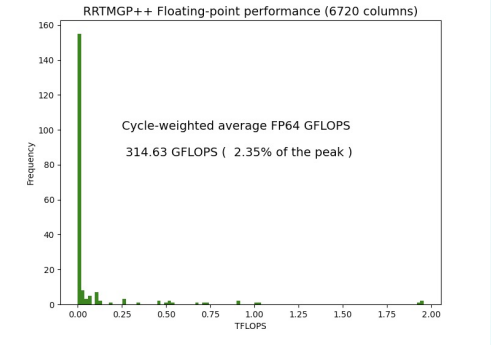
Colonel Extraction

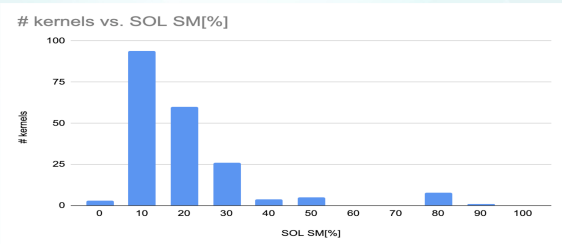
Kernel Extraction

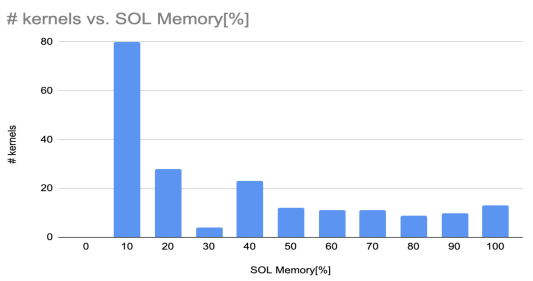
Kernel Extraction and Characterization

- ekea (tool formerly known as EKgen)
  - Extracts kernel and relevant data
  - Used for detailed characterization with realistic data
  - Vendor interactions
  - GitHub: E3SM-Project/ekea
- Detailed GPU characterization
  - Example: RRTMG++
  - Many kernels, diff behavior
  - Integer calculation (indexing)
  - Memory bandwidth
- Kernel classification









MPAS Loop Kernel

- Many MPAS-Ocean loop forms have incompatible optimizations
  - Optimal GPU forms do not vectorize on CPU, more work (1.5-2x slower on CPU)
  - Optimal CPU/vector forms do not provide enough parallel threads for GPU (GPU form 10x faster on GPU)
- Created standalone loop kernel code
  - Github: E3SM-Project/codesign-kernels
- Loop re-ordering complex
  - Not amenable to common abstraction
  - Will need to retain two versions

```
!$acc parallel loop collapse(2) &
!$acc   present(various arrays) &
!$acc   private(private vars)
do iEdge = 1, nEdges
do k = 1, nVertLevels

   Compute intermediate factors

   do i = 1, nCellsForEdge(iEdge)
      compute edgeFlx(i)
   end do

   do i = 1, nCellsForEdge(iEdge)
      highOrderFlx(k,iEdge) =
      highOrderFlx(k,iEdge) +
           edgeFlx(i)
   end do

enddo ! Vert (k) loop
enddo ! iEdge loop
```

GPU-friendly form:
Edge,k loops collapsed for more parallelism
But inside loop is not vectorizable
More memory accesses (arrays can't be reduced to scalars), masks for k-index

```
!$acc parallel loop
!$acc    present(various arrays)
&
!$acc    private(private vars)
do iEdge = 1, nEdges

  ! compute common factors as
  !k-vectors
  do k = 1, nVertLevels
     temps(k) = stuff
  end do

  do i = 1, nCellsForEdge(iEdge)

    Compute several (i,iEdge)
    factors as scalars
    do k = 1, maxLevelCell(iCell)
      flxTmp(k) = stuff
    end do ! k loop

  end do ! i loop

  do k=1,nVertLevels
     highOrderFlx(k,iEdge) =
          flxTmp(k)
  end do
enddo
```

CPU-friendly form:
Only outside loop is parallelized
Inside k-loop is vectorized
More scalar, reduced array temps for better memory use
Loop over only active layers

Summary

- Continuing to improve performance and keep campaigns going
- Building out a $2T Infrastructure plan
- But new machines hitting the floor so…

For you Letterkenny fans…