Coupling Approaches for Next Generation Architectures (CANGA)

CANGA Team Members:

P Jones, P Bochev, R Jacob, S Painter, P Ullrich, X Jaio, Z Liu,

E Constantinescu, E Coon, I Demeshko, J Guerra, H Guo, P Kuberry, V Mahadevan,

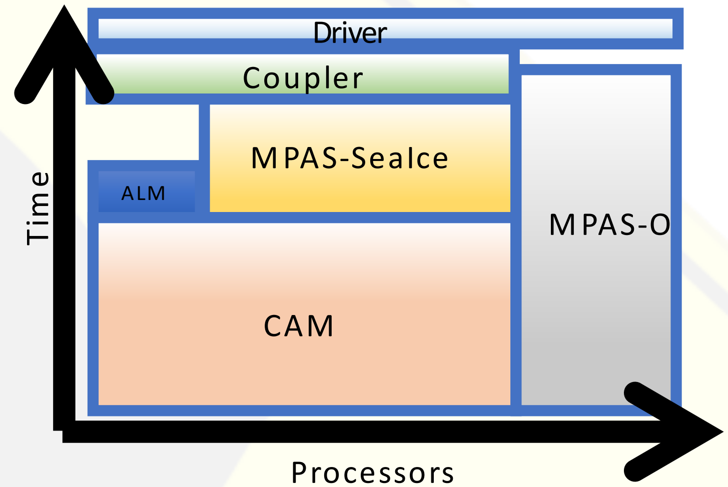
T Peterka, K Peterson, J Pietarila-Graham, M Raj, D Ridzal, D Ringo, R Ross, N Trask, H Zhang

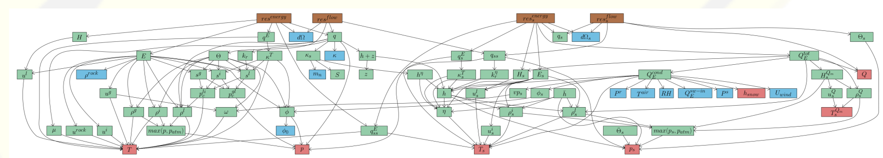
U.S. DEPARTMENT OF ENERGY | Office of Science

SciDAC Scientific Discovery through Advanced Computing

CANGA Goals

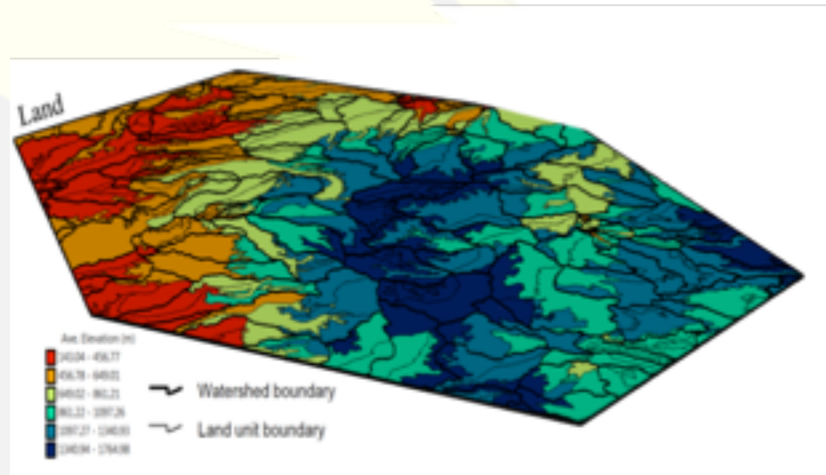
- Highly performant, accurate and robust coupling strategies for a new E3SM
- PIGLET: Prototype Integration of Global models using Legion Execution of Tasks
  - Replace hub/spoke, monolithic components
  - Asynchronous Many-Task Model
  - Exposes more parallelism
  - Better load balancing
  - Fault tolerance
  - Manage complexity
  - Enable process coupling at proper time, spatial scales

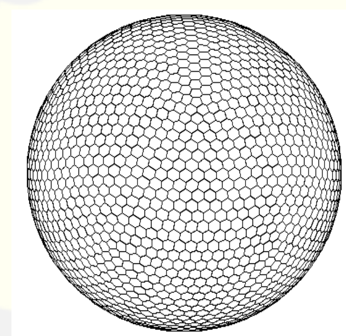


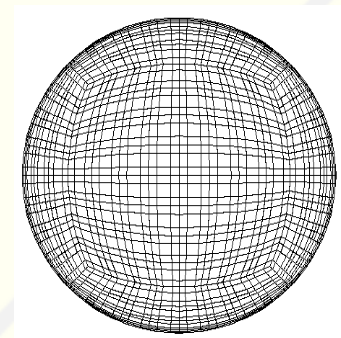


CANGA Goals

- Upgrade coupling algorithms
  - Remapping Online-Offline (ROO)
    - Non-convex cells
    - On-line adaptive remapping
    - Vector and property-preserving
    - Meshfree (agnostic to staggering location)
    - Remap test suite
  - Time InteGration for Greater E3SM Robustness (TIGGER)
    - Replace ad-hoc time-lagging and instability
    - Address multiple space, timescales
  - Integrate into task-based coupler
- Applications, mini-apps
  - Simpler coupled systems to analyze, evaluate

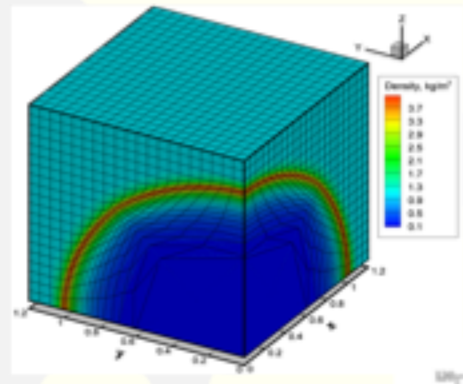




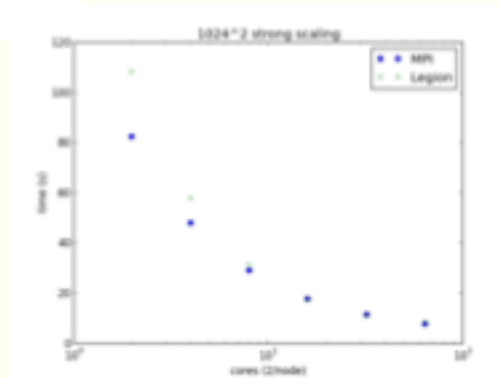


Task Parallelism with Legion and FleCSI

- Legion
  - Stanford/LANL/Nvidia
  - Logical regions intersect field/index space
  - Fast runtime analyzes dependencies (DAG) and schedules tasks
- FleCSI: C++ framework designed for multi-physics apps
  - Run-time abstraction layer (Legion, Charm++, HPX, MPI)
  - Control, execution and data models



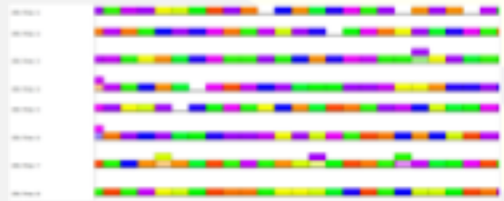
FleCSI: Pure 3D Lagrangian Sedov



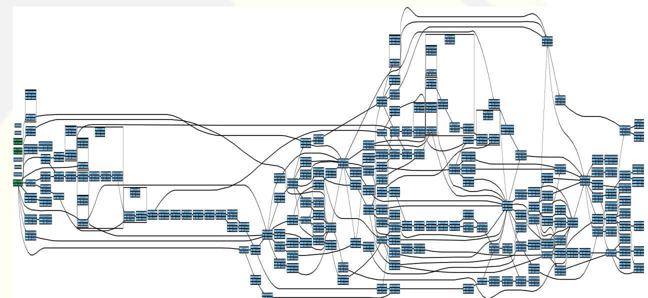
FleCSALE : Strong Scaling

Task-parallel implementation approach

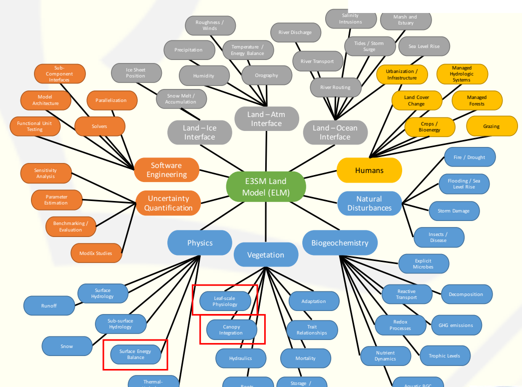
- Top-down
  - Reproduce current coupling
  - Currently have Regent prototype
    - Converting to FleCSI
  - Not enough parallelism
- Bottom-up
  - Begin to create task-based components
  - Ocean/ice (MPAS)
    - Large component – prototype interfaces
  - Land Model
    - Ideal component w/ many tasks that have high space and time heterogeneity
  - Extract in-situ analysis (RAPIDS)
    - Lagrangian particle tracking



Parallel regent prototype of coupler layer



Task graph for current ocean interfaces

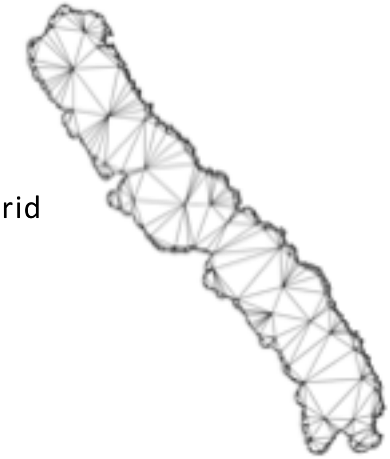


High-level diagram of land model processes

Field remapping

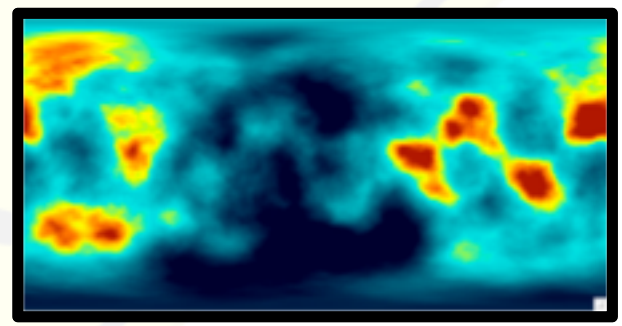
- Current coupler
  - Conservative, bilinear, others
  - Static, linear, convex cells
- CANGA-ROO
  - Irregular meshes (offline, static)
  - Adaptive on-line
    - Changing mesh/boundary, adjust order
  - Meshless
    - Support field staggering
  - Vector fields
  - Property-preserving
    - Constrained optimization
  - Test framework
    - Analytic fields derived from observation
    - Spherical harmonic truncated expansion

Triangulation of a concave grid cell generated within TempestRemap.



Optimization+meshfree remap: treating DoF as scattered data allows to circumvent complications of non-coincident interfaces

Example artificial data field constructed from spherical harmonics to mimic topography on the sphere.

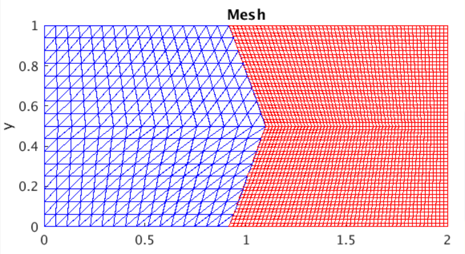


Time integration of coupled systems

- Current coupling
  - Explicit integration based on flux/field exchange
  - Time-lagged or time-averaged fields to enable concurrent execution and ease of interface
  - Effectively a sub-cycling approach: no reason why it shouldn't [not] introduce instabilities
- New approaches to advance integrated system
  - Consistent schemes with no/minimal iteration
  - Example: Lagrangian multiplier/dual Schur complement approaches
    - Still solve independently in each subdomain, estimates of flux/BC info
- Shift to multiple process timescales in task-parallel system

Simpler model hierarchy

- Capture enough complexity but no more
  - Perform stability analysis of integrated coupled system
  - Explore impact/feasibility of new time integration algorithms
- Hierarchy
  - Coupled transport (advection-diffusion) on two domains
  - Linearized Navier-Stokes
  - Q-GCM: quasi-geostrophic
  - All include field remapping
- Simple python coupler
  - Replaced by task-based coupler



Model advection-diffusion problem: convergence studies

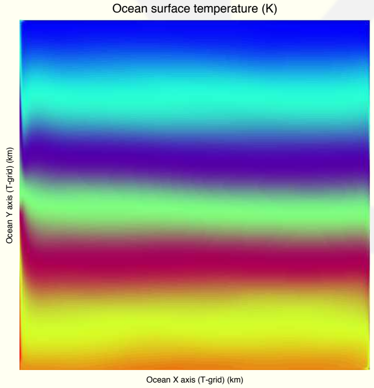
Strong advection, $\epsilon = 0.0001$

h(Ω_1)	h(Ω_2)	Δt	IVR(L1)	IVR(L2)	IVR(LC)
0.25	0.0714	1.042e-02	1.121e-01	9.114e-02	9.114e-02
0.125	0.0357	5.102e-03	3.482e-02	3.426e-02	3.783e-02
0.0625	0.0179	2.551e-03	8.282e-03	8.279e-03	8.747e-03
0.03125	0.00893	1.272e-03	1.620e-03	1.613e-03	1.663e-03
Rate	-	-	2.04	1.95	1.94

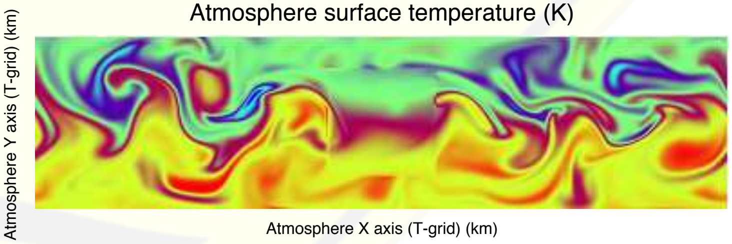
h(Ω_1)	h(Ω_2)	Δt	IVR(L1)	IVR(L2)	IVR(LC)
0.25	0.0714	1.042e-02	3.215	2.321	2.321
0.125	0.0357	5.102e-03	1.412	1.297	1.448
0.0625	0.0179	2.551e-03	0.6381	0.6261	0.6502
0.03125	0.00893	1.272e-03	0.3012	0.2985	0.3020
Rate	-	-	1.14	0.993	0.998

$$\begin{cases} \dot{\varphi}_i - \nabla \cdot F_i(\varphi_i) = f_i & \text{in} \quad \Omega_i \times [0,T] \\ \varphi_i = g_i & \text{in} \quad \Gamma_i \times [0,T] \end{cases}$$

Ocean surface temperature (K)

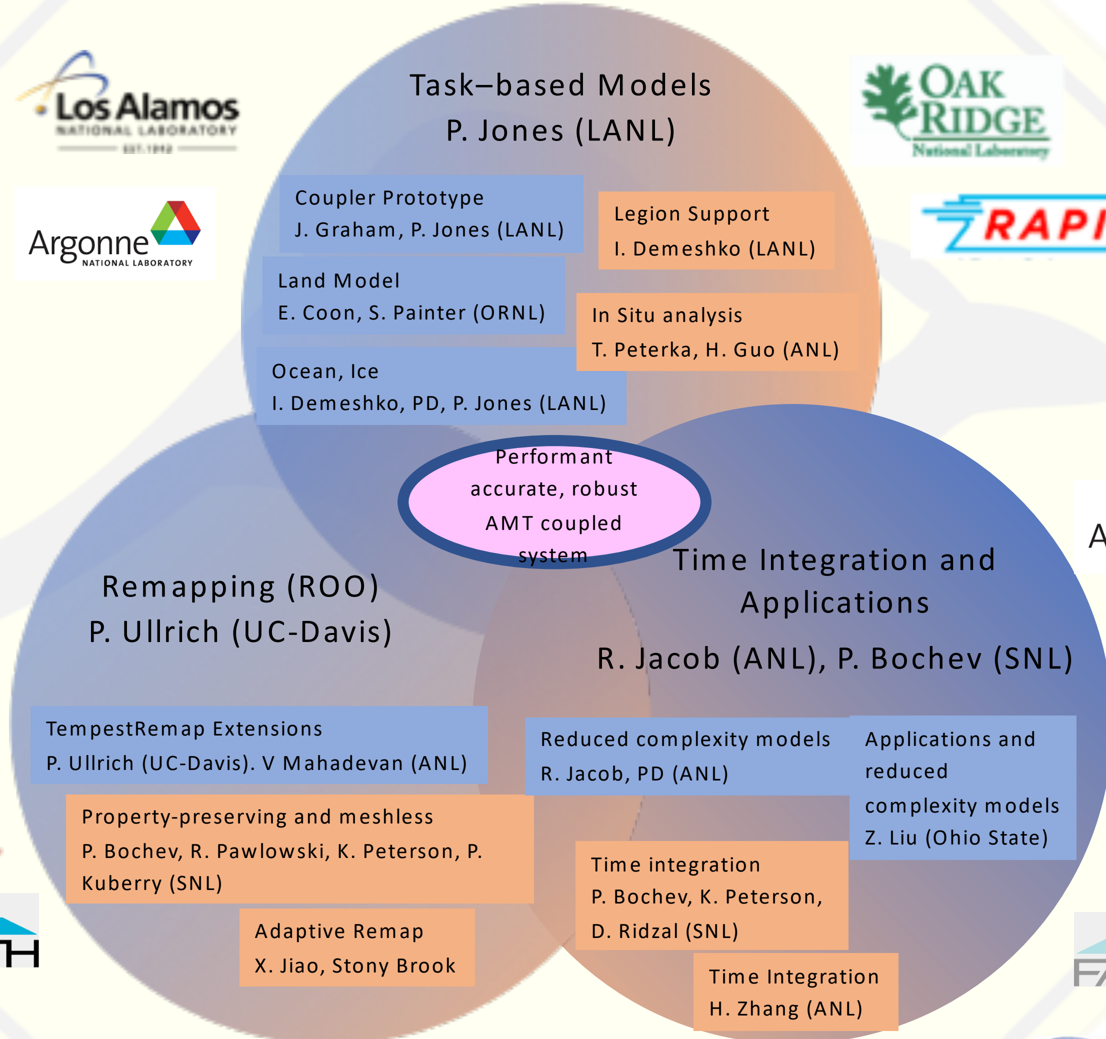


Atmosphere surface temperature (K)



ASCR funded

BER funded



Challenges, potential SciDAC links

- Task-parallel strategies
  - Many packages focused more on standard domain partitioning
  - Everyone wants control of data layout
  - Still need to optimize for node-level memory, core features (platform ready)
  - In-situ analysis, I/O as additional task parallelism
- Remapping
  - Optimized mesh and meshless tools: searching, geometry, partitioning/ordering
  - Property preservation, higher-order
- Time integration
  - Algorithms, approaches for less tightly coupled systems and with minimal iteration