High-Performance Computing Hydrologic Model

Fred L. Ogden Craig C. Douglas,
Scott N. Miller, Wencong Lai,
Ye Zhang

UNIVERSITY
OF WYOMING
New Thinking

In cooperation with:

BYU
BRIGHAM YOUNG
UNIVERSITY

UtahStateUniversity
COLLEGE OF ENGINEERING

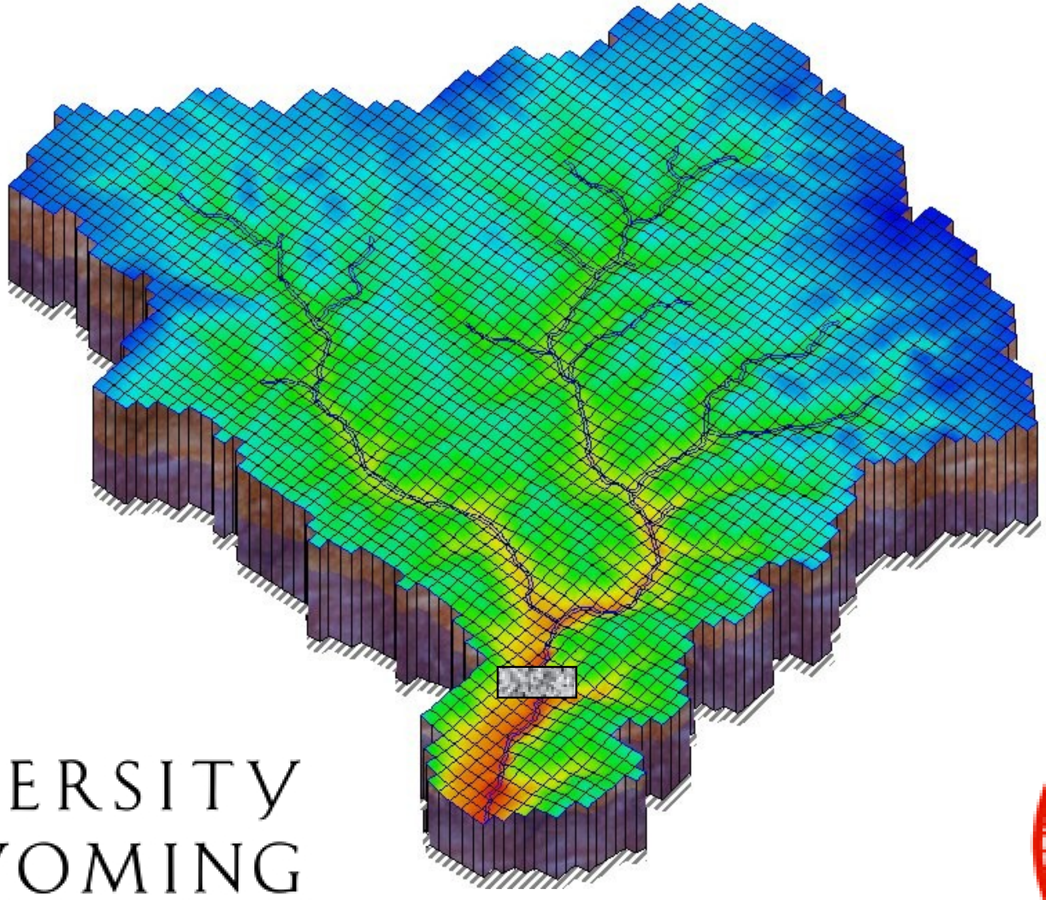
THE
UNIVERSITY
OF UTAH

Our Federal Partners

-U.S. Army Corps of Engineers, Engineer Research and Development Center, Coastal & Hydraulics and Information Technology Laboratories.

- National Center for Atmospheric Research

Gridded Surface/Subsurface Hydrologic Analysis (GSSHA) model



UNIVERSITY OF WYOMING
New Thinking

U.S. ARMY ENGINEER RESEARCH AND DEVELOPMENT CENTER
CERL GSL ITL TEC CRREL EL CHL

GSSHA is a Giga-scale Engineering Hydrologic/Hydraulic Model

- Square Grid
- OpenMP
- Multi-solver (mixed explicit and implicit)
- Multi-physics (different fundamental PDE's)
- 2D overland flow and groundwater flow
- 1D vector channel routing
- Richards eqn. or Green-Ampt with redistribution coupling between overland flow and groundwater
- Sediment/contaminant/nutrient transport
- Hydraulic structures, lakes, reservoirs, detention basins, wetland, storm drainage, tile drains, etc.
- Evapotranspiration/root zone moisture accounting

Petascale??

Discussion starter: who among us uses HPC for hydrologic modeling?

- Many hydrologists do not even do Terascale!
- Single CPU Gigascale modeling common.

High Performance Computing is a new frontier.

To consider the Petascale in hydrology, think BIG.

CI-WATER Project

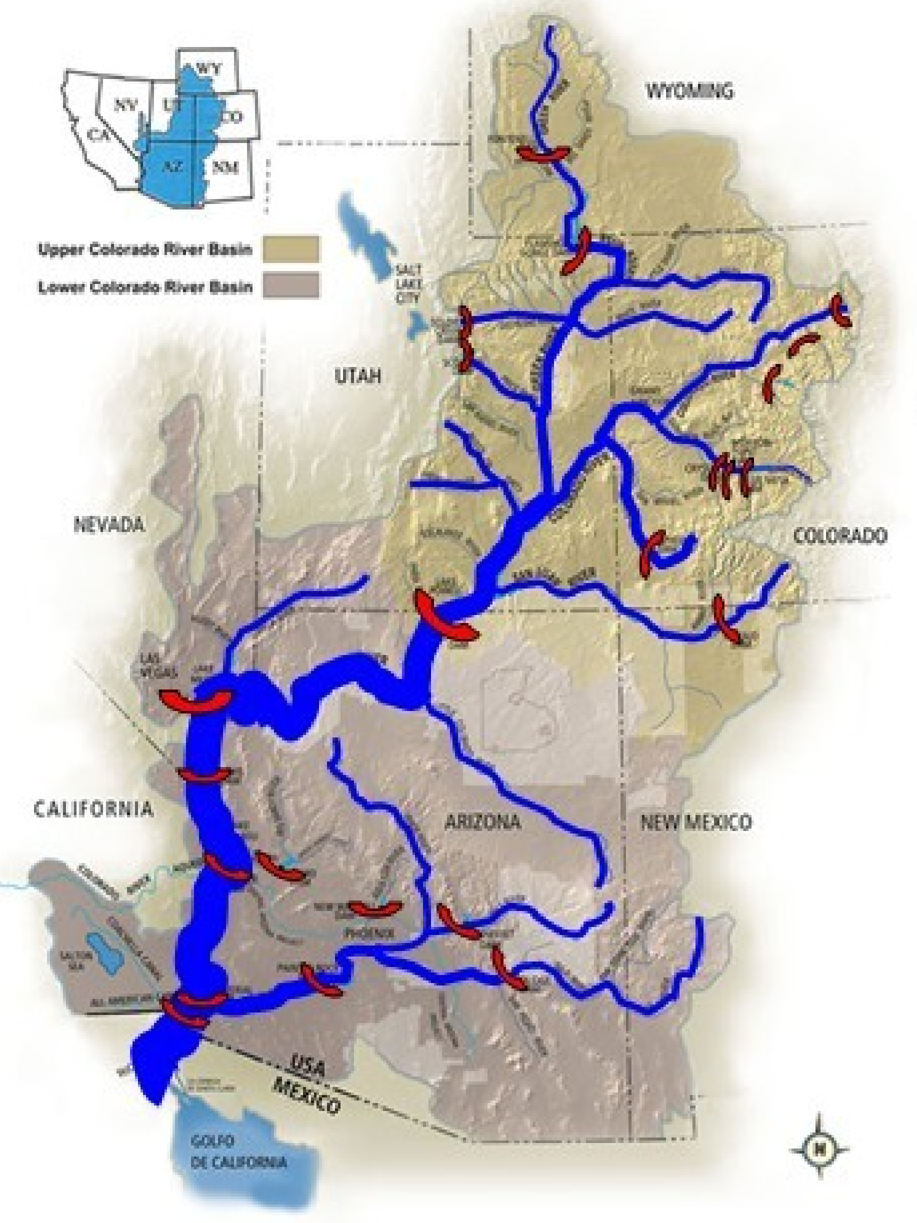
- NSF Cyberinfrastructure Cooperative Agreement joint between Utah and Wyoming EPSCoR jurisdictions.
- Focused on acquisition of hardware, development of software, capacity building, education, and outreach.

A big watershed problem:

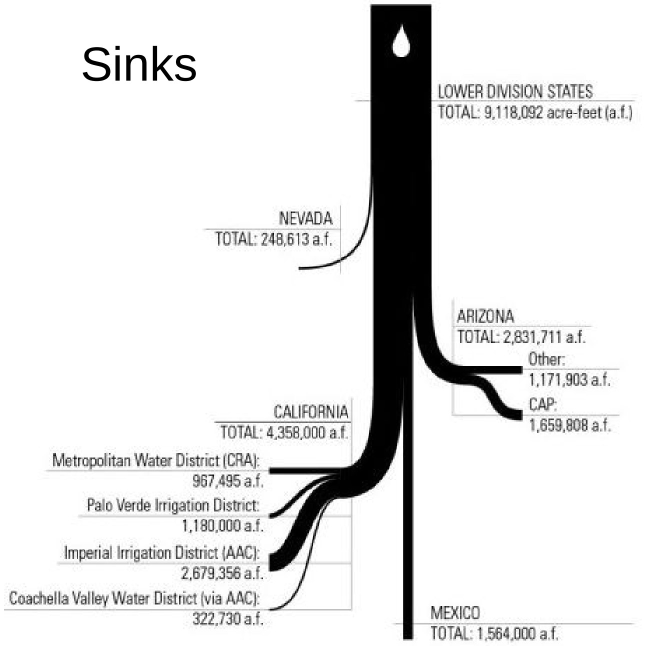
- Upper Colorado River Basin: 280,000 km^2
- High resolution important in mountains, where slope, aspect, vegetation, and wind, drive snow redistribution, sublimation, and melt.
- Low resolution in broad and extensive basins, where hydrologically interesting things seldom happen.
- *Square grid model structure is very inefficient for large watersheds where process scales vary.*

Colorado River Basin

Sources



Sinks



Numbers from the Colorado River Accounting and Water Use Report 2009.

CI-WATER Component 3 Objective

Develop a high-resolution, large-scale hydrologic model to answer three questions:

- What are the potential impacts of climate change on the long-term yield of water from the upper Colorado River basin?
- How will future land-use changes due to development and natural causes such as fire or pine bark beetle affect water supplies?
- What are the effects of trans-basin diversions and increases in water consumptive use on the water storage in Lake Powell in 50 years?

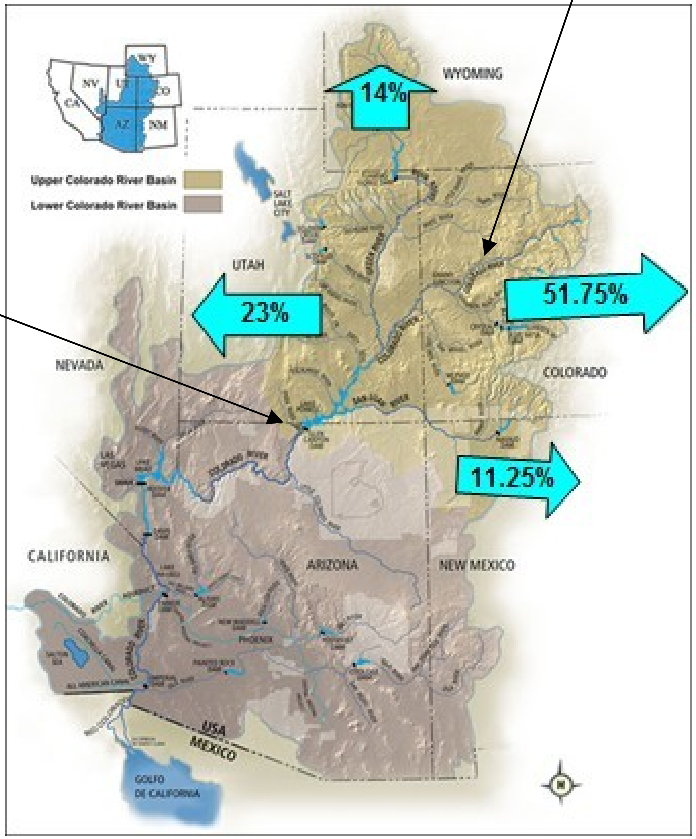
Research Goals

- Increase accessibility of high performance computing to water resources researchers, engineers, and managers.
- Produce a set of modeling tools that allow consideration of future conditions in a modeling and probabilistic framework.
- Engage the wider community by releasing the code developed for research, development, and testing.

Law of the River, Colorado River Compact, 1922

Lees Ferry, AZ, is the legal
dividing point between
Upper and Lower Basin

Upper Basin (CO, UT, WY, NM),



Lower Basin (CA, AZ, NV)
guaranteed 7.5 MAF/y

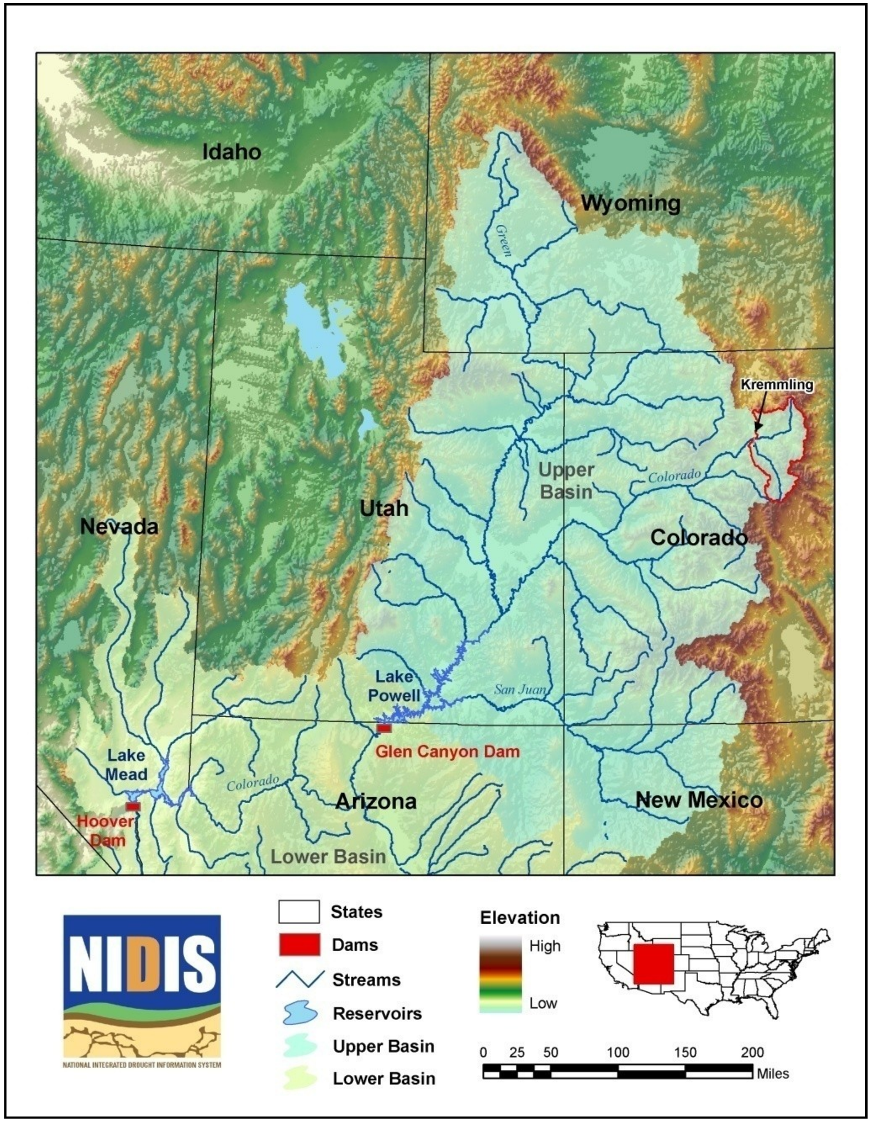
International: Mexico- 1.5 MAF/y

Note: 1 AF = 1.233 Ml

Glen Canyon Dam:
The Upper States' bank account

- Pre-1963 average 12,963,000 AF
- Post-1963 average 10,701,000 AF

Upper Colorado River Basin



Basin Area: 288,000 km2

Streams: 467,000 km

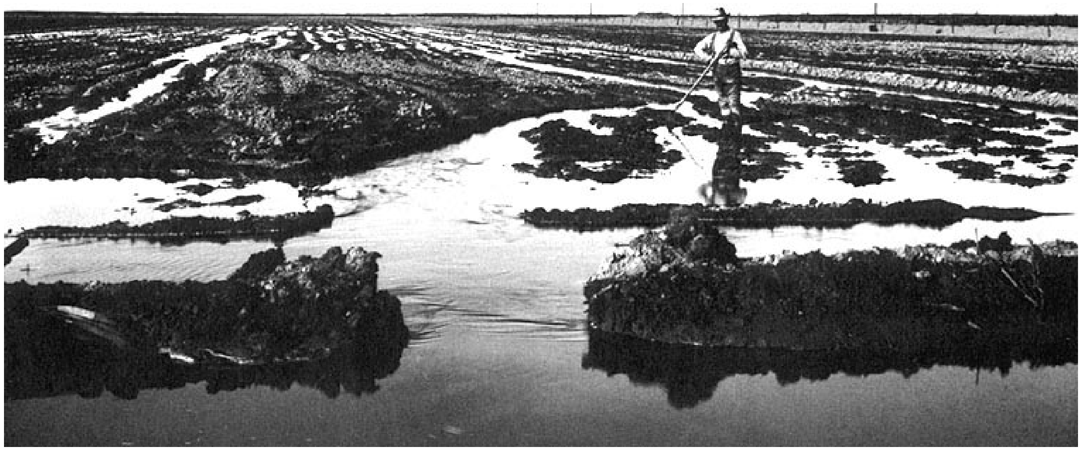
Population: 900,000 (USBR)

Area above 2700 m: 14.5% (9,000 ft)

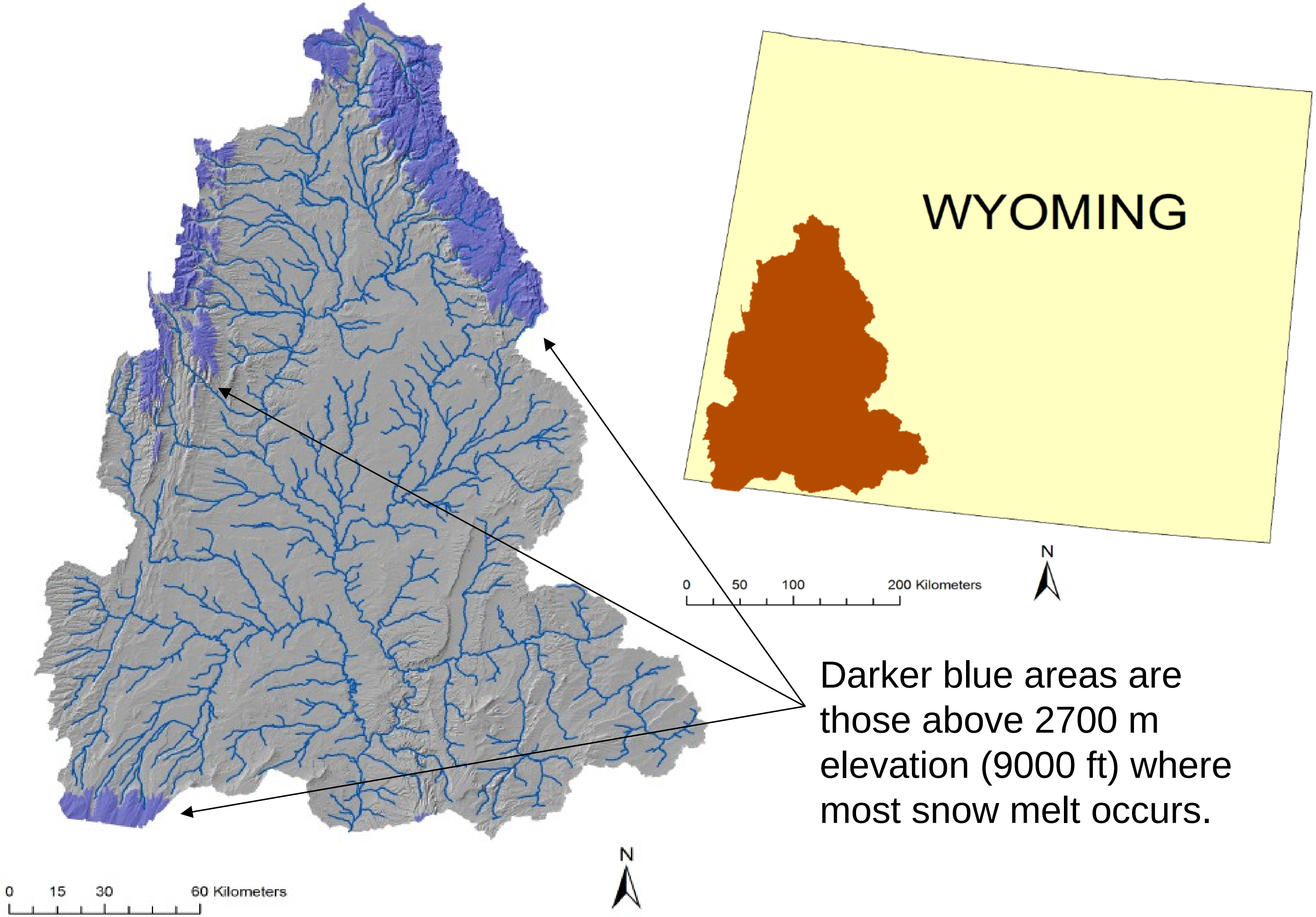
Area above 3050 m: 3.2% (10,000 ft)- this is where most of the water comes from.

High Altitude Complexity

Colorado River Basin is Highly Managed



Test Area: Wyoming Green River Basin



Darker blue areas are those above 2700 m elevation (9000 ft) where most snow melt occurs.

The NCAR-Wyoming Supercomputing Center (NWSC) provides dedicated *Petascale* capabilities for geosciences.


For more information visit, www.nwsc.ucar.edu

NWSC Partners:

University of Wyoming

Wyoming Business Council

NSF

Cheyenne Light Fuel & Power

NCAR
National Center for Atmospheric Research

Cheyenne LEADS
The Cheyenne-Laramie County Corporation for Economic Development

UCAR
University Corporation for Atmospheric Research

Architects, Contractors and Consultants:

H+L Architecture | Saunders Construction, Inc. | California Data Center Design Group | Rumsey Engineers | RMH Group
Martin & Martin Consulting Engineers | Rider Levett Bucknall | Reliable Resources | E Cube, Inc.

Wyoming’s 20% Share of NWSC's 72,300 cores represents a huge increase in EPSCoR HPC capabilities…

- On the latest (6/11) Top500 list of fastest supercomputers, Wyoming’s share on NWSC-1 alone is estimated to be…
  - The 28th fastest computer in the world
  - The 14th largest supercomputer in the US
  - The largest system in an EPSCoR state outside of Department of Energy facilities
  - The largest resource controlled by a university in the US

Reference: http://www.top500.org

HPC Data Issues

- Data assimilation (BIG DATA Problem to solve!)
  - How do we collect enough data to keep a Petascale computer busy? Just inventing data through interpolation is not acceptable.
  - In many sensor based applications today, there is a tsunami of data from each inexpensive sensor.
  - Satellite data comes 1-2 times per day in composite (incomplete) JPEG files. This is not necessarily high enough resolution and cloud cover can be a problem.
  - *We need a massive number of remote, on ground sensors, not just a massive quantity of data from a relatively few sensors*.
  - We need a symbiotic relationship between smart sensors and computational models, e.g., a dynamic data-driven application system, so that we get the right amount of data for the right scales while computing.
  - Finally, how do we afford massive data collection?

HPC Numerical Algorithms

- Multiscale methods
  - We use a base resolution with an average or median mesh size.
  - We can *upscale* to compute on a coarser mesh much quicker than on the base mesh.
  - We can *downscale* to compute on a finer mesh in a subregion of the entire domain to pick up features that are not visible on the base mesh. If the subregion is small enough, this is both computationally feasible and scientifically useful.
  - Dynamic steering of a computation is essential to make this work and can be done as postprocessing.

- Load balancing
  - This is a preprocessing step in the major computations.
  - First generate base meshes of interest and store them.
  - Generate a series of domain decompositions for different representative numbers of cores and store them.
  - Similar to the ocean modeling community meshes.

HPC Time Stepping

- Implicit methods
  - Implicit time stepping allows larger time steps while maintaining stability.
  - With massively parallel computers, an implicit method requires using massively parallel solvers from one time step to the next, while many common algorithms today just do not scale to O(10K) cores due to time consuming communications that eventually spoil scaling.
  - Communications can be modeled simply by the time to move an N byte message by $L+bN$, where L is the *average* latency time to before the first byte from the source gets to the target processor and b is the *average* time to move each byte of the message after the latency period.
  - On a small computer system, estimating L and b is simple and accurate.
  - On a complex, large system with k networks sitting in a (multi-acre) machine room, L and b become 3D variables, e.g., indexed by (pi,pj,nl).

HPC Time Stepping

- Explicit methods
  - Time steps usually limited by CFL stability conditions to $\Delta t < C(\Delta x)2$, where C is a positive real constant.
  - Reduce Δx by a factor of 10 or 100 and Δt is reduced by a factor of 100 or 10,000.
  - A new set of algorithms has recently been developed that are stable on given time steps, but use intermediate time steps (where stability may be violated) so that the stability condition is $\Delta t < C\Delta x$ instead (different C).
  - Hence, vastly larger time steps possible.
  - Mathematical theory incomplete, so they might not always work.
  - Massively parallel computations are straightforward with explicit methods even if they take too long to be useful.

HPC Time Stepping

- Hybrid explicit-implicit methods
  - On the boundaries of the subregions use an explicit method to approximate the solution on the next time step.
  - Use an implicit method in each subregion, where the size of the subregions is small enough so that the algorithm used to get to the next time step scales well.
  - Possibly iterate on the boundary points to improve accuracy.
- Hybrid implicit-explicit methods
  - Downscale the problem to only the boundaries of the subregions and use an implicit method to approximate the solution on the next time step. This can be done in parallel based on subregions.
  - Use an explicit method in each subregion.
  - Possibly iterate on the boundary points to improve accuracy.
- Implications for Petascale computing
  - Both hybrid methods should scale and be fast.
  - Need to analyze which hybrid method works best for CI-WATER.

We are not starting from scratch (thanks to our collaborators)

- USACE-ERDC providing:
  - Finite element computational kernel derived from ADH model
  - Computational model builder (CMB)
  - ezVIZ HPC visualization tools
  - ezHPC user interface toolkit

Conclusions

- The CI-WATER project is joint cooperative agreement between NSF and the Utah and Wyoming NSF EPSCoR jurisdictions

- The project aims to increase accessibility of HPC resources for water resources research and management in the Upper Colorado River Basin.

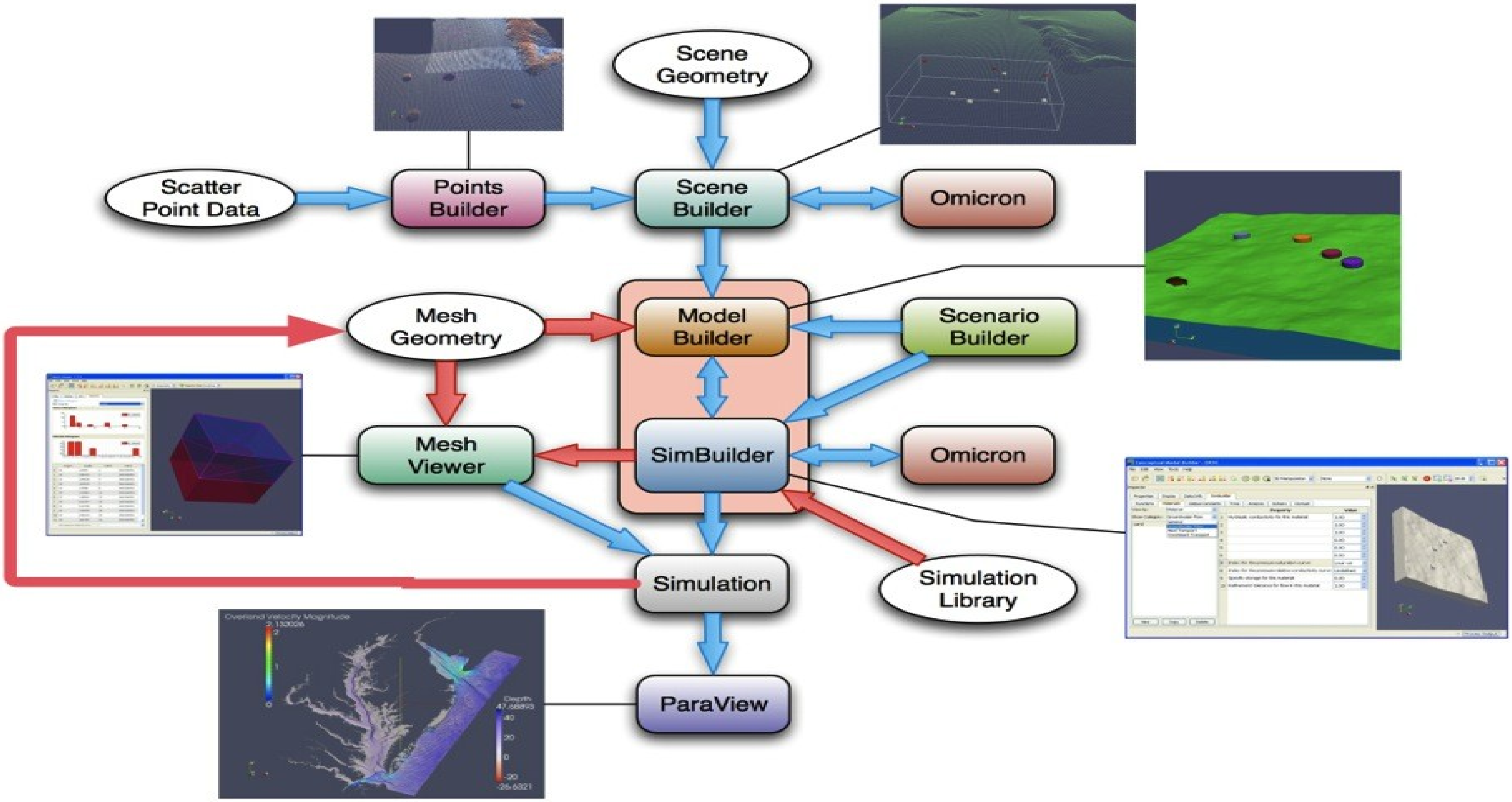
- CI-WATER has components related to data, cloud computing, accessibility, education, outreach, and diversity

Thank you

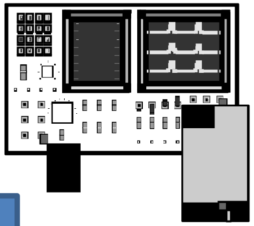
Computational Model Builder

- Designed for large complex domains and HPC
- No licensing fees
- Cross platform
- User-configurable
- Built as several complimentary, independent tools

Computational Model Builder Data Flow



User Interface Toolkit Application Program Interface (API)

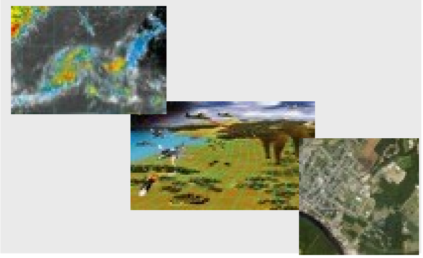


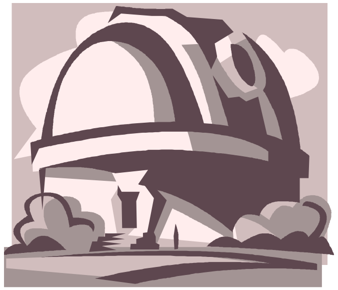
UIT

Making High Performance Computing Easy

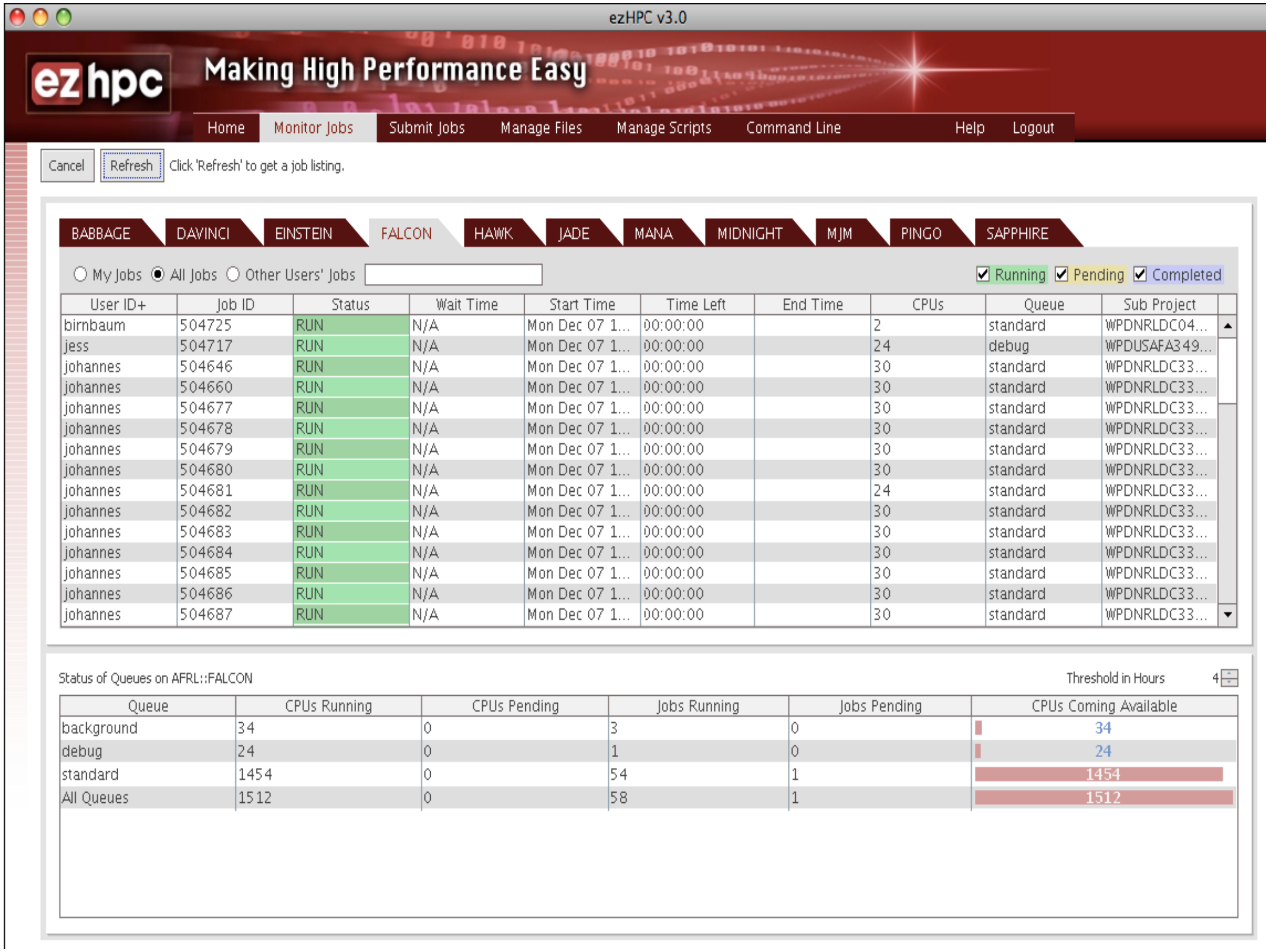
- Supports Novice to Expert Users
- Central Access to HPC Resources
- Custom Productivity Clients
- Complete Job Stream Management
- Fast Large File Transfers
- Secure Authentication

UIT API





User Interface Toolkit – ezHPC



Click 'Refresh' to get a job listing.

○ My Jobs ● All Jobs ○ Other Users' Jobs

☑ Running ☑ Pending ☑ Completed

User ID+	Job ID	Status	Wait Time	Start Time	Time Left	End Time	CPUs	Queue	Sub Project
birnbaum	504725	RUN	N/A	Mon Dec 07 1...	00:00:00		2	standard	WPDNRLDC04...
jess	504717	RUN	N/A	Mon Dec 07 1...	00:00:00		24	debug	WPDUSAFA349...
johannes	504646	RUN	N/A	Mon Dec 07 1...	00:00:00		30	standard	WPDNRLDC33...
johannes	504660	RUN	N/A	Mon Dec 07 1...	00:00:00		30	standard	WPDNRLDC33...
johannes	504677	RUN	N/A	Mon Dec 07 1...	00:00:00		30	standard	WPDNRLDC33...
johannes	504678	RUN	N/A	Mon Dec 07 1...	00:00:00		30	standard	WPDNRLDC33...
johannes	504679	RUN	N/A	Mon Dec 07 1...	00:00:00		30	standard	WPDNRLDC33...
johannes	504680	RUN	N/A	Mon Dec 07 1...	00:00:00		30	standard	WPDNRLDC33...
johannes	504681	RUN	N/A	Mon Dec 07 1...	00:00:00		24	standard	WPDNRLDC33...
johannes	504682	RUN	N/A	Mon Dec 07 1...	00:00:00		30	standard	WPDNRLDC33...
johannes	504683	RUN	N/A	Mon Dec 07 1...	00:00:00		30	standard	WPDNRLDC33...
johannes	504684	RUN	N/A	Mon Dec 07 1...	00:00:00		30	standard	WPDNRLDC33...
johannes	504685	RUN	N/A	Mon Dec 07 1...	00:00:00		30	standard	WPDNRLDC33...
johannes	504686	RUN	N/A	Mon Dec 07 1...	00:00:00		30	standard	WPDNRLDC33...
johannes	504687	RUN	N/A	Mon Dec 07 1...	00:00:00		30	standard	WPDNRLDC33...

Status of Queues on AFRL::FALCON

Threshold in Hours 4

Queue	CPUs Running	CPUs Pending	Jobs Running	Jobs Pending	CPUs Coming Available
background	34	0	3	0	34
debug	24	0	1	0	24
standard	1454	0	54	1	1454
All Queues	1512	0	58	1	1512

Tabbed Functions

- **MOTD and system news @ HOME Tab**
- **Monitor Jobs & Queue Status on all machines**
- 0 Job Management
  - 0 Script generator & editor
  - 0 Allocation and Utilization viewer
  - **Fast large file transfers**
  - **Easy access to custom scripts**

Monitor Kerberos Ticket Session Time

Easy Access to on-line documentation

Establishing a Petascale Collaboratory for the Geosciences: Scientific Frontiers

- "A PCG will enable the simulation of the full spectrum of interactions among physical, chemical, and biological processes in coupled Earth system models.
- Land-atmosphere property fluxes are forced by surface ecosystem heterogeneity on scales of 1 m or less. The forcing is the result of a huge array of interacting biological, chemical, and geological processes
- Understanding the integrated effects of these processes is necessary for predicting ecosystem change and water availability."

A Report to the Geosciences Community.
UCAR/JOSS. 80 pp., 2005

Conclusions

- To get to Petascale, we need the following:
  - New ways of thinking about the hydrology model.
  - More complex and massive data collection through remote, intelligent sensors.
  - Multi-physics hydrologic process models at very fine scales.
  - Much more complex numerical algorithms in both time and space that are stable and new for O(100K) cores.
  - Much more complex software developed for O(1K) to O(10K) cores that is being extended to O(100K) to O(10M) cores.
  - By the time we are done getting to Petascale, Exascale will be threatening us.
  - This is not your daddy’s hydrology or computing model anymore.

- Think BIG??? Maybe EXTREMELY MASSIVE is more appropriate.