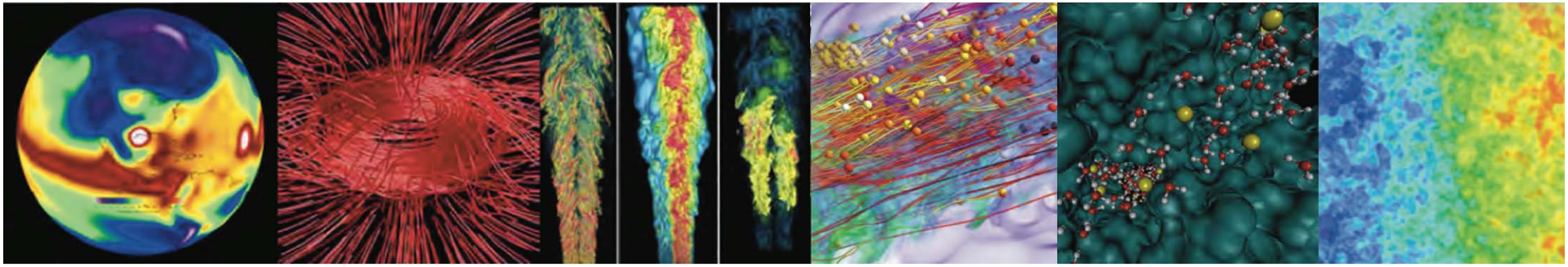
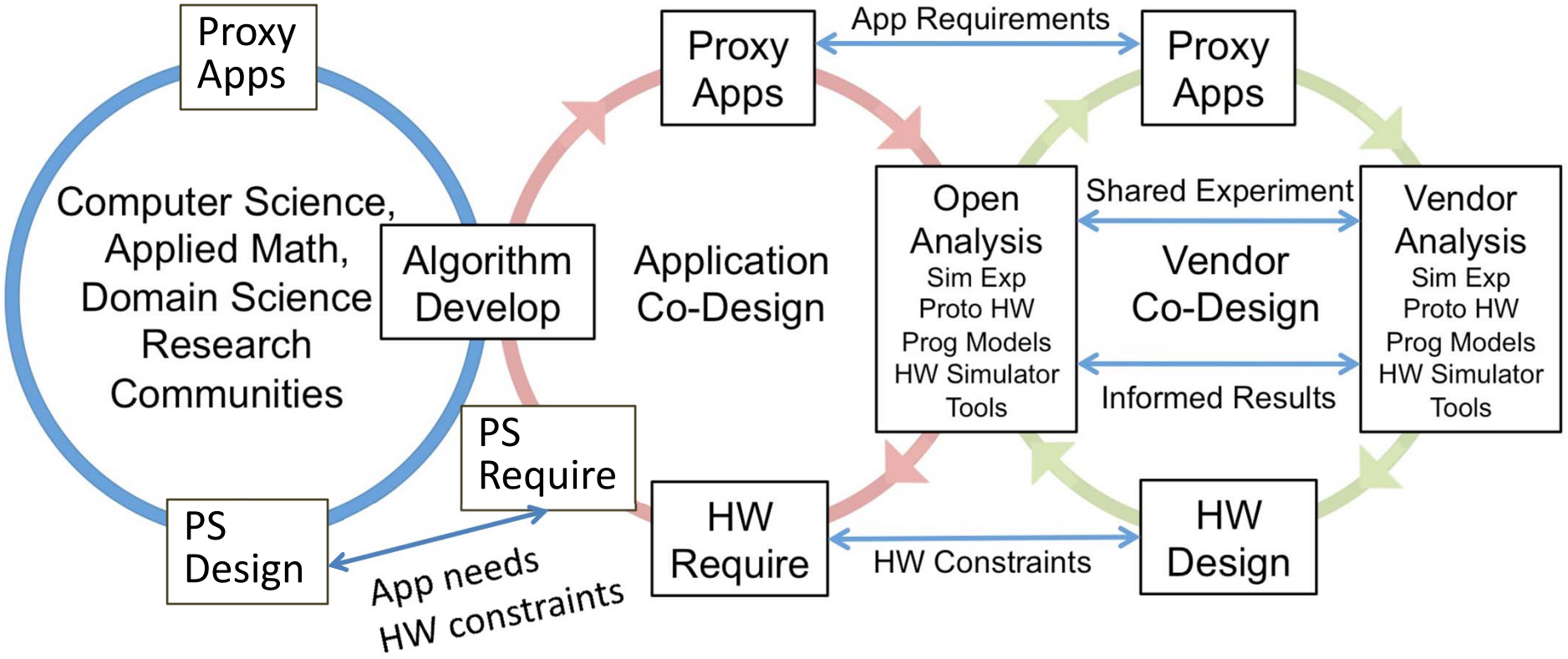


Co-Design and X-Stack Coordination

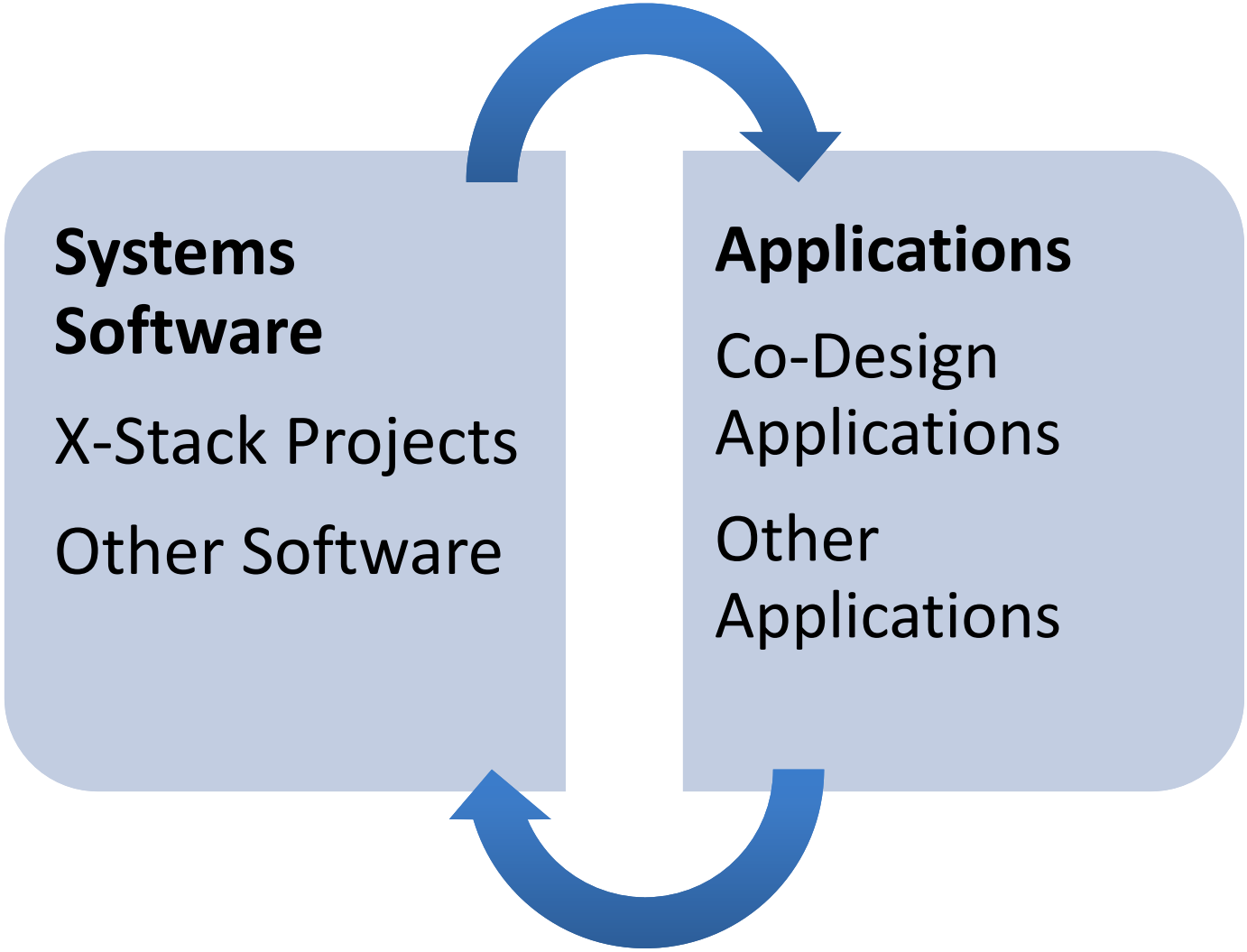
Kathy Yelick,
with input from John Bell, Jackie Chen,
Pat McCormick, Jim Belak,
Ron Brightwell, Thomas Sterling, Dan Quinlan,
Wilf Pinfold, Shekhar Borkar

Application Co-Design Center Interactions

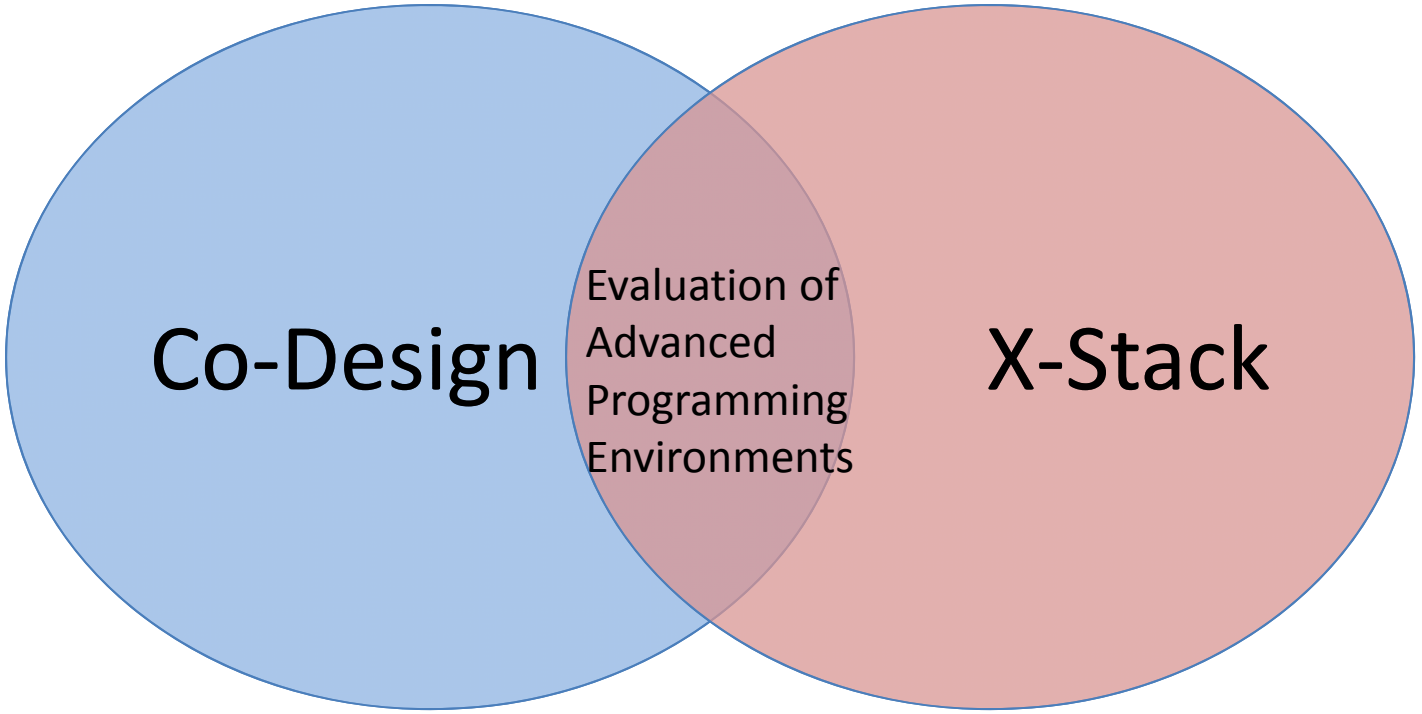


PS = Programming Systems (languages and their implementations, programming libraries, programming environment tools)

This is not just about X-Stack and Co-Design Centers



Beyond the History



- **X-Stack needs to do evaluation. Anything else is bad science.**
- **Co-Design needs to evaluate advanced programming systems; this will improve software development costs *in the long term***
- ***Not about specific individuals (aka postdocs) but full team efforts***

The Real Exascale Communication Challenge

```
apply[fn;x;a] =
    [atom[fn] → [eq[fn;CAR] → caar[x];
                 eq[fn;CDR] → cdar[x];
                 eq[fn;CONS] → cons[car[x];cadr[x]];
                 eq[fn;ATOM] → atom[car[x]];
                 eq[fn;EQ] → eq[car[x];cadr[x]];
                 T → apply[eval[fn;a];x;a]];
    eq[car[fn];LAMBDA] → eval[caddr[fn];pairlis[cadr[fn];x;a]];
    eq[car[fn];LABEL] → apply[caddr[fn];x;cons[cons[cadr[fn];
                                                caddr[fn]]
```

$$\frac{\partial \rho}{\partial t} + \frac{\partial (\rho u)}{\partial x} + \frac{\partial (\rho v)}{\partial y} + \frac{\partial (\rho w)}{\partial z} = 0$$

$$\frac{\partial (\rho u)}{\partial t} + \frac{\partial (\rho u^2)}{\partial x} + \frac{\partial (\rho uv)}{\partial y} + \frac{\partial (\rho uw)}{\partial z} = -\frac{\partial p}{\partial x} + \frac{1}{Re_r}\left[\frac{\partial \tau_{xx}}{\partial x} + \frac{\partial \tau_{xy}}{\partial y} + \frac{\partial \tau_{xz}}{\partial z}\right]$$

$$\frac{\partial (\rho v)}{\partial t} + \frac{\partial (\rho uv)}{\partial x} + \frac{\partial (\rho v^2)}{\partial y} + \frac{\partial (\rho vw)}{\partial z} = -\frac{\partial p}{\partial y} + \frac{1}{Re_r}\left[\frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \tau_{yy}}{\partial y} + \frac{\partial \tau_{yz}}{\partial z}\right]$$

$$\frac{\partial (\rho w)}{\partial t} + \frac{\partial (\rho uw)}{\partial x} + \frac{\partial (\rho vw)}{\partial y} + \frac{\partial (\rho w^2)}{\partial z} = -\frac{\partial p}{\partial z} + \frac{1}{Re_r}\left[\frac{\partial \tau_{xz}}{\partial x} + \frac{\partial \tau_{yz}}{\partial y} + \frac{\partial \tau_{zz}}{\partial z}\right]$$

Interaction Models

Require people at all levels of seniority

- **Proxy App Model**
  - Leverages work already done by Co-Design Centers
  - Good to get X-Stack started, but limited (easy to misinterpret)
- **Deep Collaboration Model**
  - 1-on-1s, ongoing to discuss Application challenges and PS design
  - A lot of work, but best alignment; having people in common helps
- **Hackathon**
  - Apps folks bring code examples; modified to run on simulators, etc.
  - Requires mature systems from X-Stack side
- **Attend each others meetings**
  - All-hands meetings are part of (at least) the large X-Stack projects and Co-Design project; Cross-invitations are key

X-Stack Portfolio

D-TEC (Dan Quinlan)

Complete software stack solution, from DSLs to optimized runtime systems code.

X-Tune (Mary Hall)

Unified autotuning framework that integrates programmer-directed and compiler-directed autotuning.

DEGAS (Kathy Yelick)

Hierarchical and resilient programming models, compilers and runtime support.

CORVETTE (Koushik Sen)

Automated bug finding methods to eliminate non- determinism in program execution and to make concurrency bugs and floating point behavior reproducible.

XPRESS (Ron Brightwell)

Software architecture and interfaces that exploit the ParalleX execution model, prototyping several of its key components.

GVR (Andrew Chien)

Global view data model for architecture support for resilience.

Traleika (Shekhar Borkar)

Exascale programming system, execution model and runtime, applications, and architecture explorations, with open and shared simulation infrastructure.

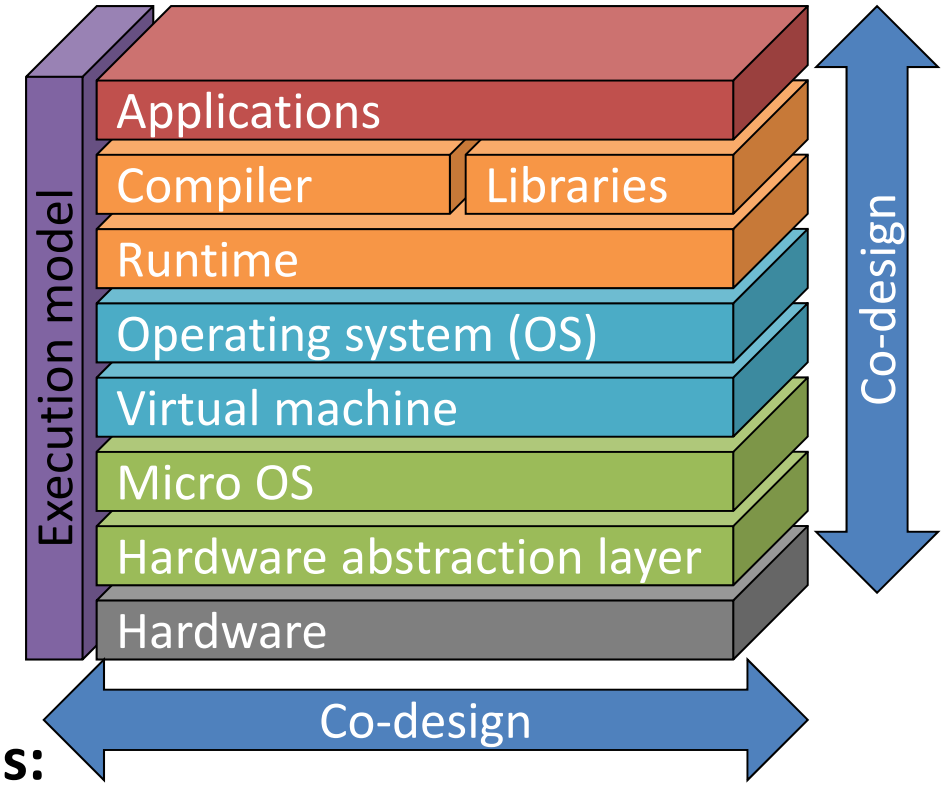
SLEEC (Milind Kulkarni)

Semantics-aware, extensible optimizing compiler that treats compilation as an optimization problem.

DynAX (Rishi Khan)

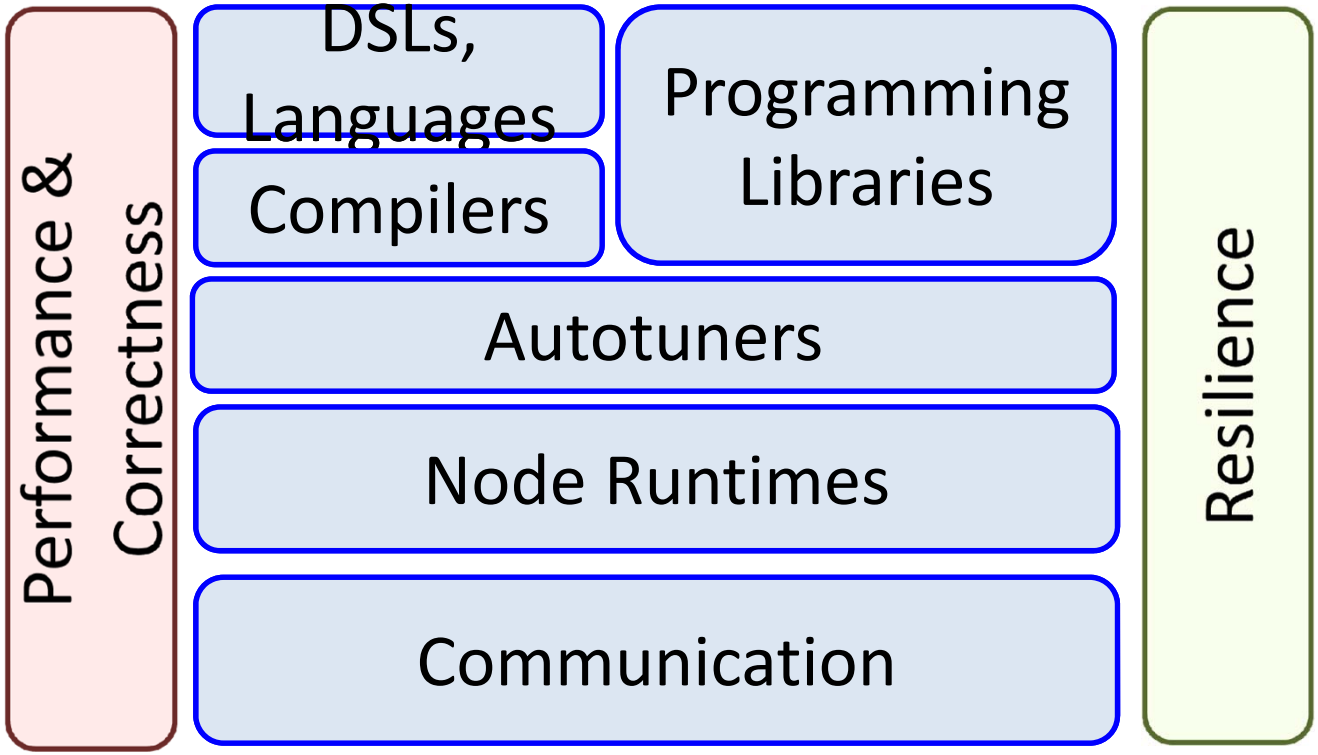
Novel programming models, dynamic adaptive execution models and runtimes.

The Software Stack (source: Bill Harrod)



- **X-Stack Goals:**
  - Enable DOE applications to take maximum use of future architectures (involves MPI)
  - Broaden the base of future applications by improving programmability (may not involve MPI)

X-Stack Stack



- **Languages include GPLs and DSLs implemented with compilers and libraries:** X10, Swift, UPC Habanero-C (HC). XPI, CAF 2.0. SLEEC, SWARM/SCALE, Sketch, Rosette, Legion

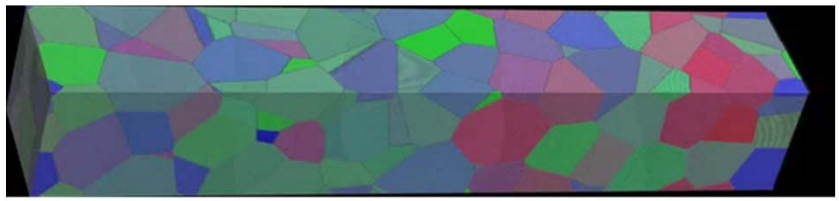
- **May after the 10th at MIT – check ipdps, ics, hpdc, pldi**

X-Stack Goals and Objectives

- **Scalability:**
  - Billion-way concurrency, thousand-way on chip with new architectures
- **Programmability:**
  - Convenient programming
- **Performance Portability:**
  - Ensure applications can be moved across diverse machines
- **Resilience:**
  - Support for capturing state and recovering from faults
- **Energy Efficiency:**
  - Take advantage of low-power designs (may change programming)
- **Interoperability:**
  - Provide transition path for applications

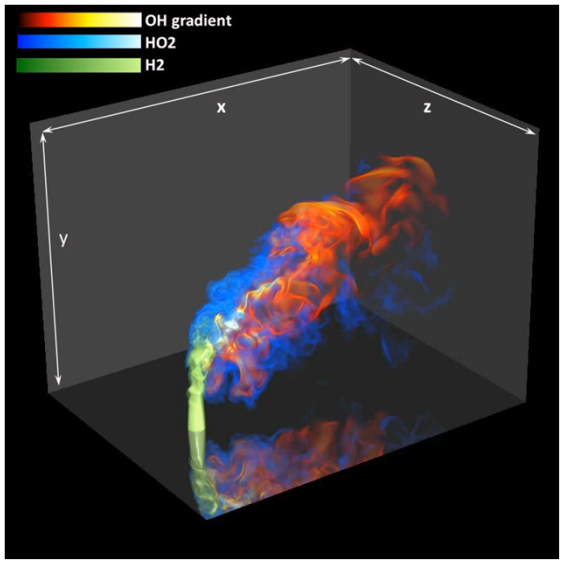
ExMatEx: Collaborations with X-Stack

- **Proxy Apps**
  - PGAS app to be sent (→ DEGAS,…)
- **Deep collaborations**
  - D-TEC collaboration (via Colella)
- **Hackathons**
  - Planned with Intel / ETI
- **Meetings**
  - Several X-Stack reps attended all-hands meeting



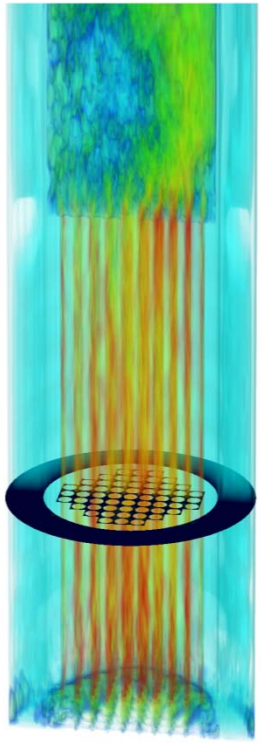
ExaCT: Collaborations with X-Stack

- **Proxy Apps**
  - PGAS app to be sent (→ DEGAS,…)
- **Deep collaborations**
  - D-TEC and X-Tune (via Williams)
  - Legion and DEGAS (via GASnet)
  - Node scheduling / locality (via Shalf)
- **Hackathons**
  - Planned with Intel / ETI
- **Meetings**
  - X-Stack reps attended all-hands meeting



Cesar: Collaborations with X-Stack

- **Proxy Apps**
  - Several uses of NEK-Bone (X-Tune,…)
- **Deep collaborations**
  - Postdoc hired specifically for this interaction
- **Hackathons**
  - TBD
- **Meetings**
  - TBD

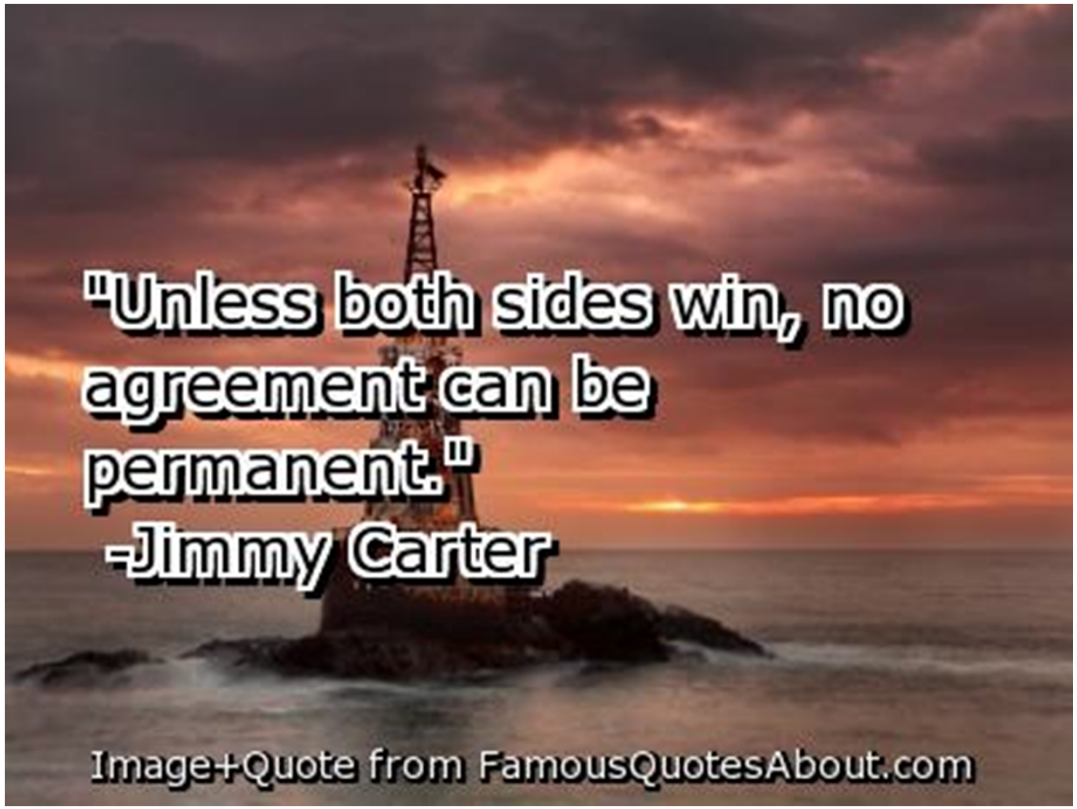


Cross-Cutting Issues

- **ExMatX has a postdoc working on Resilience**
  - Probably best done by interacting with resilience efforts
  - DEGAS (CDs/BLCR), GVR, D-TEC HL/LL, Intel,…
- **Correctness:**
  - Corvette sensitivity to numerical precision: ExMatEx and ExaCT (in discussions so far)
- **Performance:**
  - Pervasive, so may not be separate activity
  - Simulators are closely related

		X10	Swift	UPC	Habanero-C (HC)	CAF 2.0	SLEEC	SWARM/SCALE	Sketch	XPI	Rosette
6	Energy, resilience,...	Resilience (in progress)	Automatic retry	Resilience (in progress)	Resilience (in progress)	Resilience (in progress)	Heterogeneity	Resilience (in progress)		Resilience (in progress)	

Both Sides Win



- **Collaborative model will help X-Stack centers increase impact**
- **Collaborative model will help Co-Design centers (and apps in general) adapt to future systems and reduce software costs**

Metrics

- **Joint publications**
- **Talks on joint work: at Exascale PI in particular**

Comments

- **Evolutionary vs. revolutionary paths and stack (does it evolve)**
- **Scoping the collaboration**

The X-Stack Programming Model Puzzle Pieces

		X10	Swift	UPC	Habanero-C (HC)	CAF 2.0	SLEEC	SWARM/SCALE	Sketch	XPI	Ros-ette
1	DSL?	No	No	No	No	No	Yes	No	Yes	No	Yes
2	Inter/Intra-Node?	Same	Inter	Both	Both	Same	Neither	Both	Intra	Same	Intra
3	Types of parallelism	Task, Map-Reduce	Task	SPMD	Task → SPMD	SPMD	N/A	Task	Implicit	Tasks	Implicit
4	Synch	Conditional atomics	Implicit dataflow	Barriers locks → Atomics	Finish, Asynch, Phasers → Futures, atomics, Actors	Events, locks, cofence, barriers, finish. → atomics.	N/A	Depen-dences, codelet chaining, latches, barriers, locks, etc.		Futures Dataflow	
5	Comm	Shared memory		PGAS & collect-ives	Shared mem + {MPI, GASNet}	PGAS & collect-ives.	N/A	Futures, remote codelets		Active GAS	
6	Energy, resilience,...	Resilience (in progress)	Automatic retry	Resilience (in progress)	Resilience (in progress)	Resilience (in progress)	Heterogeneity	Resilience (in progress)		Resilience (in progress)	

Project Title	Lead PI(s)	Lead Institution	Description
Traleika Glacier	**Shekhar Borkar**	**Intel**	Simulation infrastructure. Compiler optimization, execution models, and runtime environments. https://sites.google.com/site/traleikaglacierxstack/
DEGAS	**Kathy Yelick**	**LBNL**	Hierarchical programming models, language design, compilers, communications layer, adaptive runtime, resilience. http://crd.lbl.gov/organization/computer-and-data-sciences/ftg/projects/DEGAS/
D-TEC: DSL Technology for Exascale	**Dan Quinlan**	**LLNL/MIT**	Complete solution for X-Stack: DSL, compilers, abstract machine model, refinement and transformation framework, adaptive runtime systems. http://www.dtec-xtack.org
XPRESS	**Ron Brightwell**	**Sandia/IU**	Runtime system implementing Parallex, co-designed with an OS. Framework to translate MPI and OpenMP legacy codes. http://xpress.sandia.gov
DynAX: Adaptive X-Stack	**Rishi Khan**	**ETI**	Programming Models, Compilers and Runtime Systems for Dynamic Adaptive Event-Driven Execution Models. http://www.etinternational.com/xstack
Autotuning for Exascale	**Mary Hall**	**Unv. Utah**	A unified autotuning framework that seamlessly integrates programmer-directed and compiler-directed autotuning. http://ctop.cs.utah.edu/x-tune/
Global View Resilience	**Andrew Chien**	**UC/ANL**	Global view data model to enable application-informed, portable resilience. End-to-end approach for incremental adoption and evaluation of architecture support for resilience. http://gvr.cs.uchicago.edu
CORVETTE	**Koushik Sen**	**LBNL**	Automated bug finding methods to eliminate non- determinism in program execution and to make concurrency bugs and floating point behavior reproducible. http://crd.lbl.gov/organization/computer-and-data-sciences/ftg/projects/corvette
SLEEC	**Milind Kulkarni**	**Purdue**	Annotation language. Function semantics exposed in DSL libraries. Design and development of a semantics-aware, extensible optimizing compiler that treats compilation as an optimization problem. https://engineering.purdue.edu/~milind/sleec/