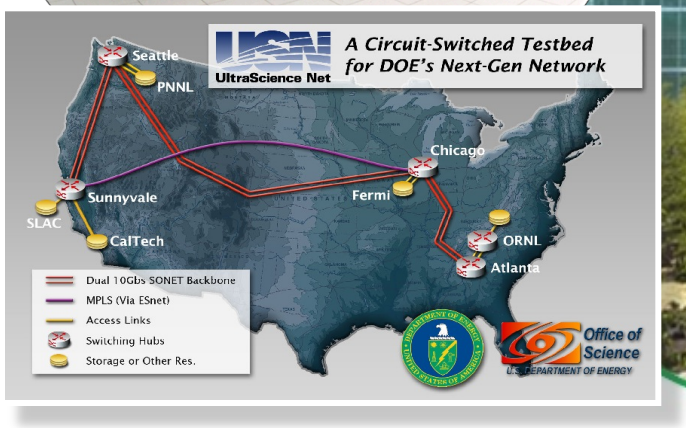
UCCS

Universal Common Communication Substrate



Presented by:
Pavel Shamis

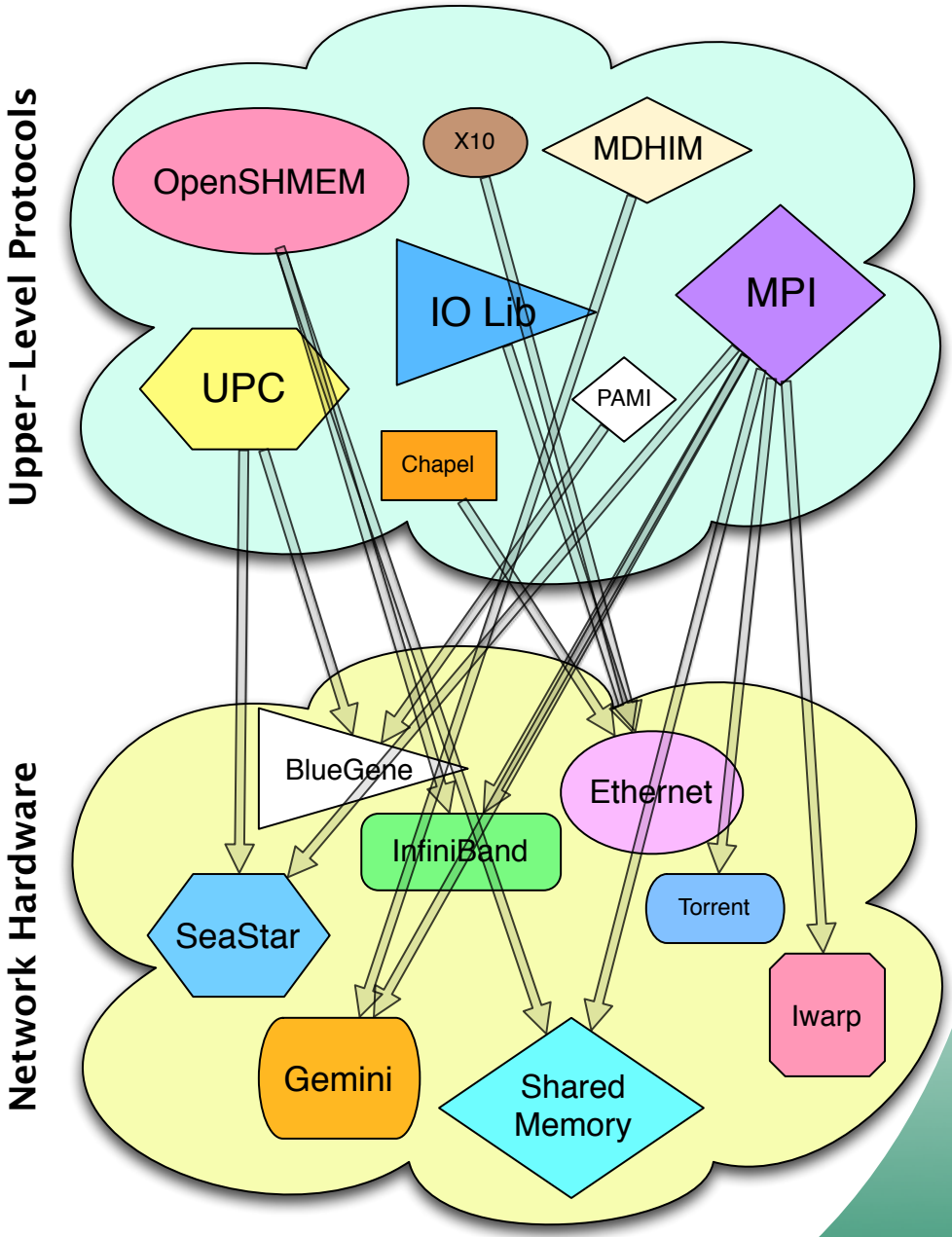
November, 2012

Outline

- **Motivation**
- **UCCS**
- **Goals & Requirements**
- **Status**

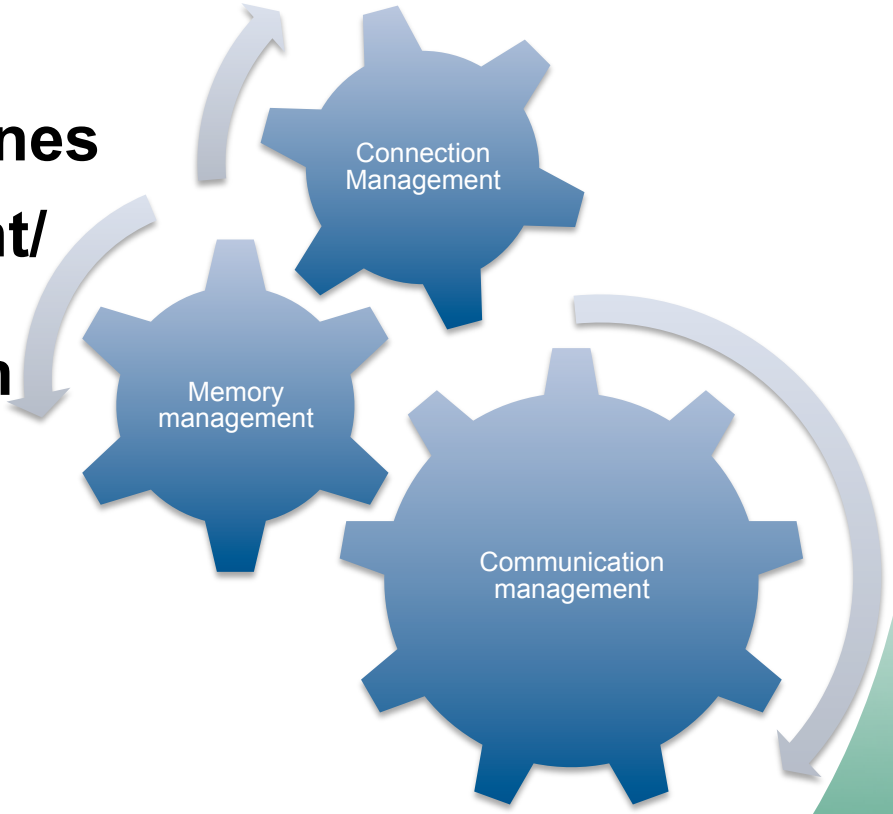
Motivation

- **Upper-Level Protocols (ULP) provides a wide degree of variation in communication**
- **Network Hardware exposes a range of different capabilities and interfaces**



Motivation – Cont'd

- **Low-level Network Interfaces are complicated**
  - **Tens of thousands code lines**
  - **Several years to implement/ debug/optimize full communication stack from scratch**
  - **High performance implementation requires hardware vendor level of expertise**

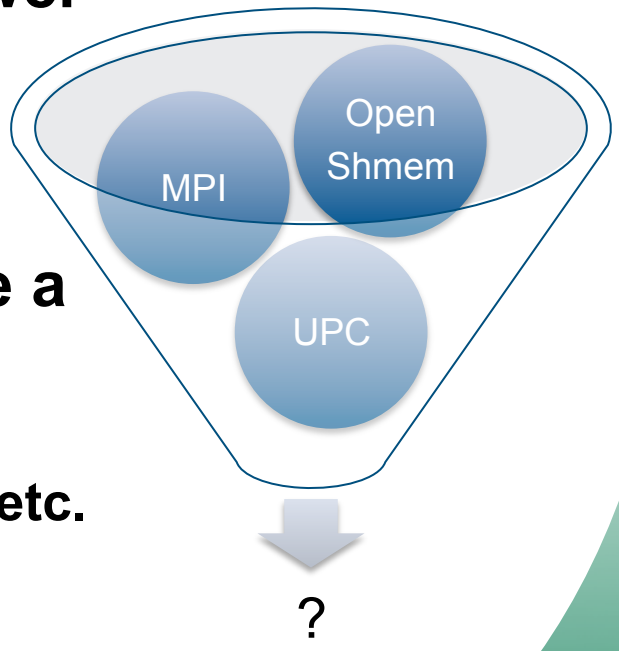


Motivation – Cont'd

- Multiple ULPs work hard to Re-implement low-level communication layer
- High performance communication support is required over a range of network hardware !
- "Implement it on top of MPI"
  - Good for prototypes
  - Performance penalty

Is there any hope ?

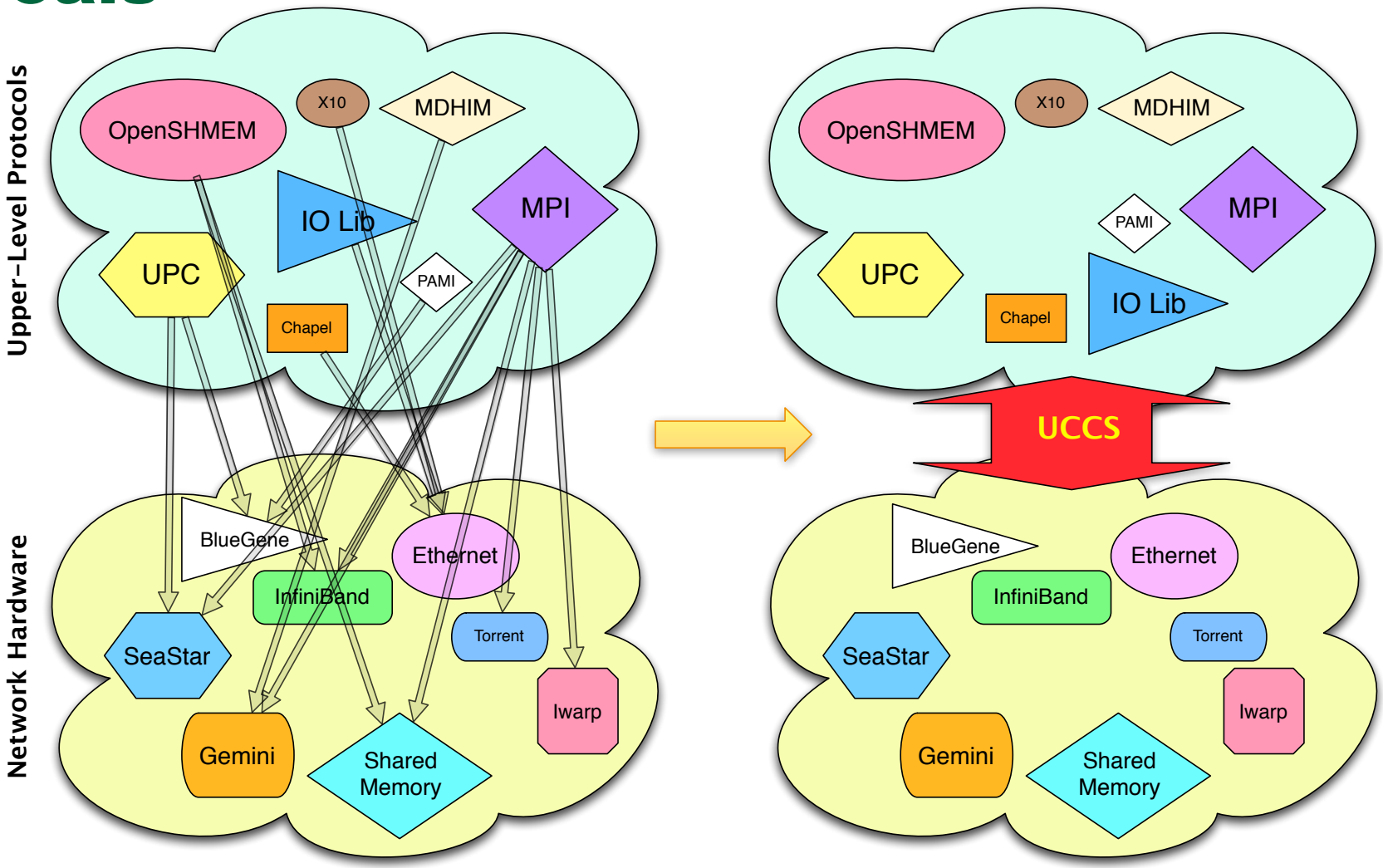
- ULPs
  - For a carefully chosen division of the communication stack, ULPs can have a high degree of overlap in the requirements they place on the lower level layers
- Low-level Network Interfaces
  - Communication interface can have a high degree of overlap in communication semantics
    - Send/Recv, RDMA, AMO, Collectives, etc.



UCCS

- **Universal Common Communication Substrate (UCCS)**
  - **High performance communication middleware for parallel programing models, File I/O, and BigData**

Goals



- **Provide scalable high-performance communication capabilities while supporting multiple programming models and network hardware technologies**

Goals – Cont'd

- **Reduce the development cycle barriers for new ULPs and programming models by providing a broader, more flexible network abstraction**
- **Reduce the application/programming barriers for new networks, by providing a stable p-model/user layer which can use any UCCS-supporting network**

Goals – Cont'd

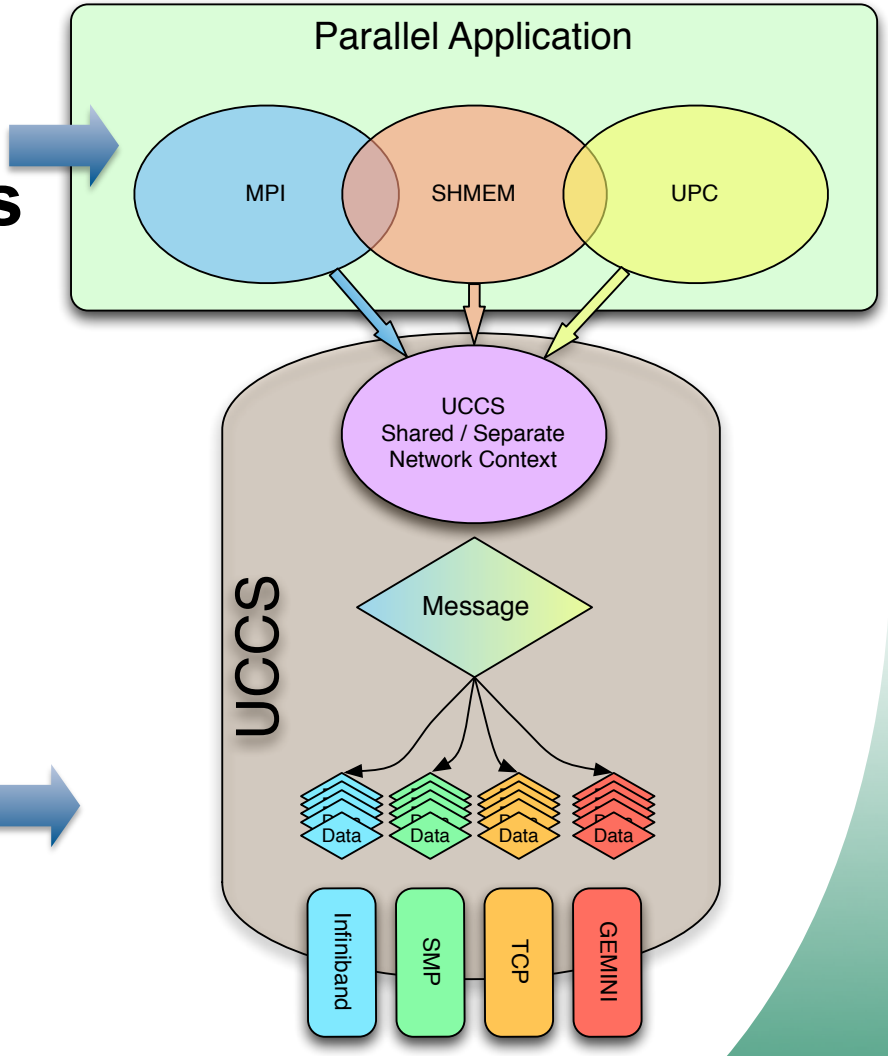
- Support a range of programming models
  - PGAS (OpenSHMEM, UPC, Chapel, X10, etc.)
  - MPI
  - I/O (SPIL)
  - Multi Dimensional Hashed Indexed Metadata (MDHIM)
  - Language extensions
  - BigData
  - Business Analytics

Goals – Cont'd

- If possible, leverage existing community project (s)
- Allow for long term support
- Scale to ten's of thousands of nodes
- Assume >= 10 year lifespan
- Allow for I/O, Libraries, Language enhancements

Low-level Communication Library Support Requirements

- Capable of simultaneous support for multiple ULP's



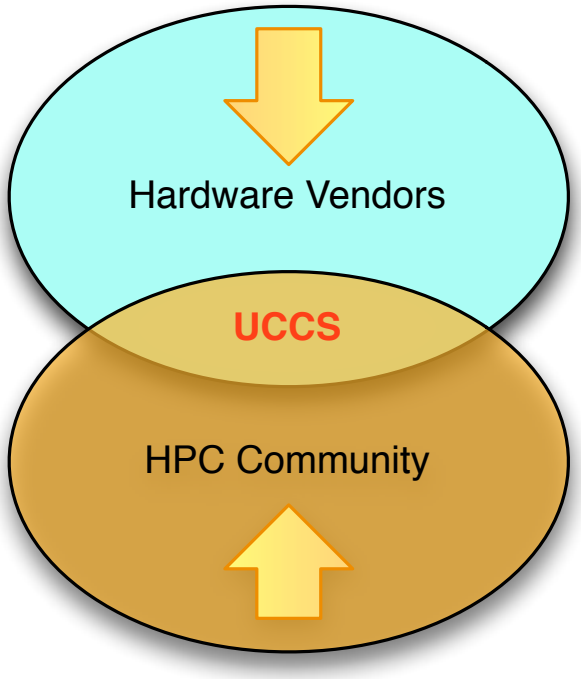
- Simultaneous use of different hardware communication stacks (enabling technology)

Low-Level Communication Library Support Requirements

- Low S/W overheads in “critical path”
  - RMA, AMO, collectives
  - Modern network devices demonstrate sub-micro latencies, making the software overhead more dominant.
- Flexible and extendable interface
  - Hardware “friendly” requirements

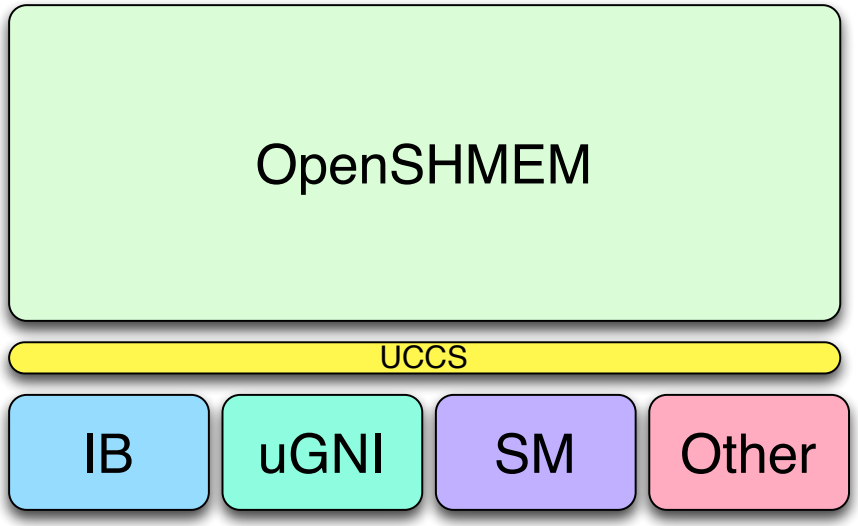
Long Term Goals

- Direct network hardware support
- Co-design
  - Hardware
  - Compilers
- Community support



OpenSHMEM & UCCS

- **Strong support for PGAS models like OpenSHMEM (but not only!)**
- **Very short critical path**
  - **Tight integration with hardware**
- **Maximum hardware utilization**



Status

- UCCS Specification v0.1
- Implementation
  - Based on the Module Component Architecture (MCA) and Open MPI network layer (Not MPI!)
  - Extended for PGAS/IO/…
- We are open for collaboration !

Early Results

- Infiniband Connext-X rev1 / Perftest
- PUT:
  - Typical ULP overheads: ~150-800 nsec (above VERBS)
  - UCCS : ~32 nsec Faster than native VERBS!
- GET:
  - Typical ULP overheads: ~250-800 nsec (above VERBS)
  - UCCS: ~10 nsec (above VERBS)

Acknowledgements

This work was supported by the United States Department of Defense & used resources of the Extreme Scale Systems Center at Oak Ridge National Laboratory.