Requirements

- Rather than assume open source or other define what the requirements are
  - Community developed does not need to be open source

- Might be different at different levels
  - Closer to applications
    - Makes sense to be more towards open source
    - Should be applicable across architectures
  - Closer to hardware
    - Can make sense to be less fully open source
    - Provider may pay for this

- Earlier in timeline focusing on wider community need – program models
  - Allow early investigation
  - Later in timeline then vendor-specific implementation
    - Funding could potentially used for non open source initiative

- Example out in community about mixed development model
  - HPSS – not open source, but community has access
Requirements

- **Community wants**
  - Does not want to be limited to full proprietary solution
  - Flexibility to replace components of stack
  - Open API
  - Leverage investment
  - Applications to have common environment

- **Provider**
  - Not be held responsible for components that do not have control over

- **Facility**
  - Level of quality
  - Best value

- Need 3-way relationship so all parties are involved in delivering end solution
What is suggested Model for software

Hard quadrant for vendor

Community Developed
Provider Supported

Community Developed
Community Supported

Provider Supplied
Provider Supported

Provider Supplied
Community Supported

RFP and acceptance

RFP

*developed implies who implemented
*supplied could be co-developed
Mapping Components to Quadrants

- **Operating systems**
  - Could be co-developed (e.g., Linux)

- **Runtime Systems**

- **I/O systems**

- **Systems Management**
  - Low-level (RAS, power control, boot) vendor developed
  - Higher-level resource management, security, performance co-developed

- **External Environments**

- **Programming Models**
  - Industry Standard (OpenMP, MPI, COF)
  - Other (UPC, ARMCI)

- **Frameworks**

- **Compilers**
  - Different models work

- **Numerical Libraries**
  - Different models work

- **Debugging tools**
  - Different models work

- **Application Element: Algorithms**

- **Application Support: Data Analysis and Visualization**

- **Application Support: Scientific Data Management**

- **Resilience**

- **Power Management**

- **Performance Optimization**

- **Programmability**