

AIMES: Abstractions and Integrated Middleware for Extreme-Scales

Shantenu Jha, Matteo Turilli - Rutgers U

Jon Weissman - U Minnesota

Daniel S. Katz, Zhao Zhang - U Chicago

DOE NGNS PI Meeting – Mar 18-20 2013, Emeryville, CA

<http://aimes-project.org>

DOE ASCR under grant numbers: DE-FG02-12ER26115,
DE-SC0008617 & DE-SC0008651



RUTGERS



UNIVERSITY OF MINNESOTA



THE UNIVERSITY OF
CHICAGO

Objectives

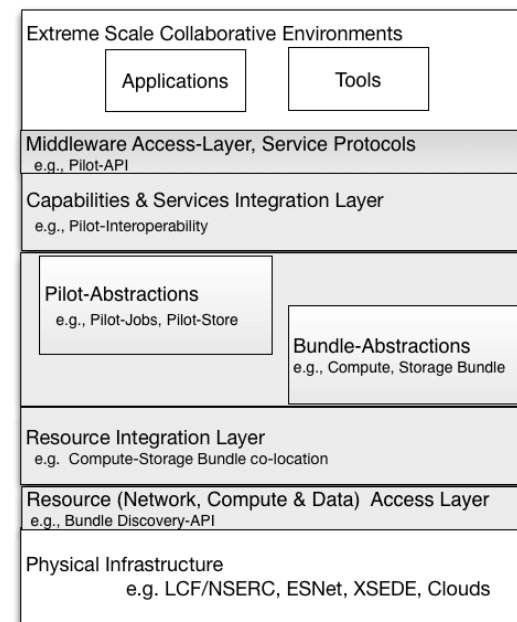
- Distributed computing is an important component in extreme-scale collaboration
- Extreme-Scale Challenges: predictable, flexible, reliable, federation
- “Beyond Glue” => reason about the application and infrastructure
 - Develop new abstractions for resources at all levels

Impact

- Model for next-generation distributed applications and infrastructures
- Predictable performance across diverse infrastructures (flexible execution) for small- and large-scale collaborations
- New integrated resource management paradigms

Progress and Accomplishments

- DoE application analysis to produce initial skeleton drivers for research
- Defined primary abstractions: pilot and bundle interactions, and use cases
- Identified key research elements: signatures
- Identified initial short, medium & long term research & development goals



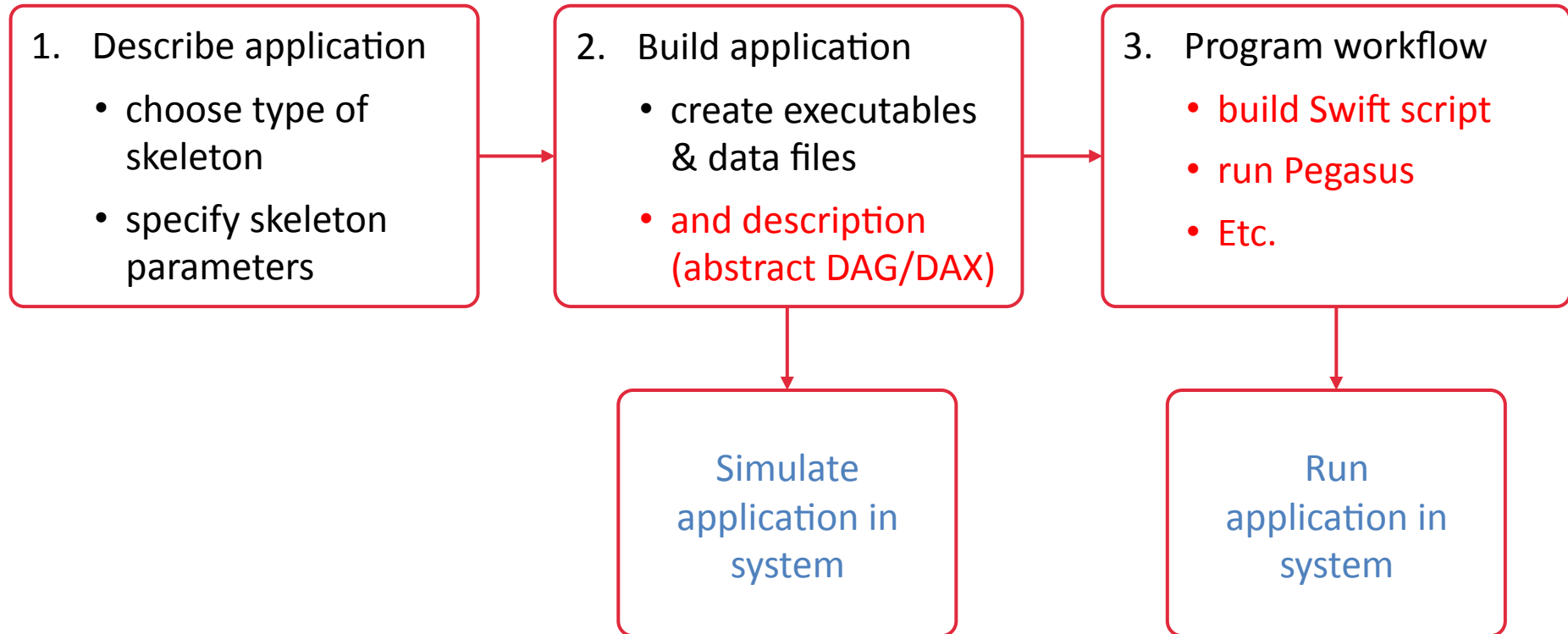
Overview

- Distributed computing is one pathway to extreme-scale and a necessity for collaboration
- **The Problem**
 - Current state of distributed applications and infrastructure is problematic
 - Two situations predominate:
 - Applications are inflexible: optimized and pinned to a specific platform
 - Applications are flexible: run anywhere but performance is unpredictable
- **The Hypothesis**
 - Flexibility and performance are compatible!
 - Extracting Simplicity whilst managing Complexity requires abstractions
 - Abstractions and Models are key to “reason about distribution”
- **The Solution**
 - We advance middleware abstractions that will facilitate reasoning integrated across application and resource level
- **AIMES**: Integrated approaches to Resource Management

The Application Perspective

- AIMES has identified DoE applications in consultation with developers and users and their “patterns”
 - LHC histogramming: Distributed MapReduce
 - MG-RAST and KBASE: Bag of Tasks and Distributed MapReduce
 - PDSF applications: Bag of Tasks
 - TIP: Bag of Tasks
 - Fusion: Concurrent communicating processes
- Other distributed applications for our research:
 - Montage: multi-stage workflow
 - Clustering: iterative MapReduce
 - Kalman filtering Reservoir Modeling: campaign

Role and Use of Application Skeletons

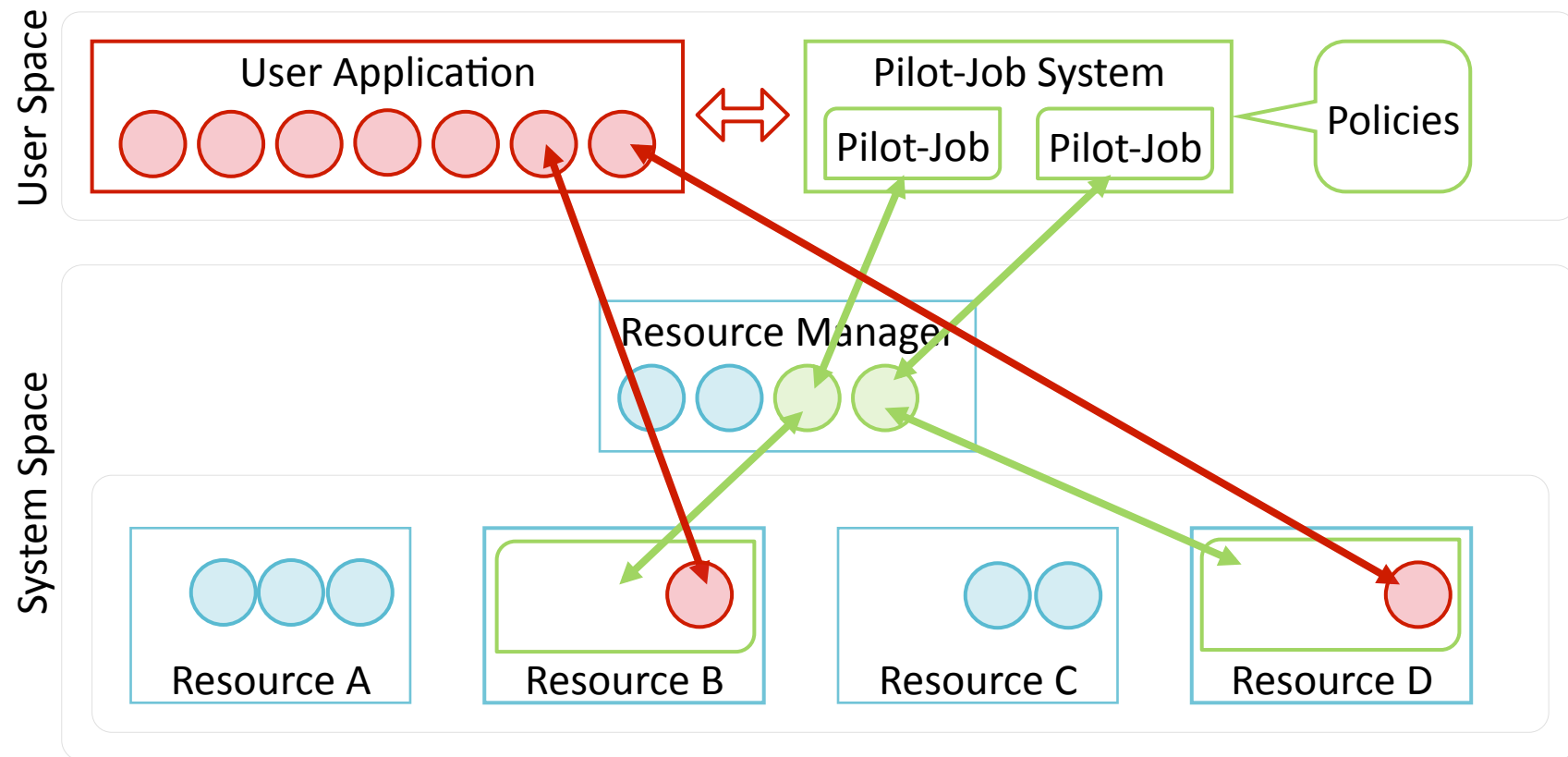


Red = not written yet

Blue = not tested yet

Introduction to Pilot Jobs

- **Working definition:** a system that generalizes a placeholder job to provide multi-level scheduling to allow application-level control over the system scheduler via a scheduling overlay



Introduction to Pilot Jobs

- **Working definition:** a system that generalizes a placeholder job to provide multi-level scheduling to allow application-level control over the system scheduler via a scheduling overlay
- **Advantages** of Pilot Job systems:
 - Abstraction between application and resource layer
 - Avoid bottlenecks of system-level only scheduling
 - Move control upwards and provides flexibility upwards
 - Enable the “slicing and dicing” of resources for applications
- **Limitations** of current Pilot-Jobs implementations:
 - Many Pilot-Job offerings, often semantically distinct
 - Conceptual: P* towards a unified view of pilot-jobs, pilot-abstractions
 - How to “slice and dice” resources? How to map to infrastructure?
 - Role for underlying abstraction (Bundles)?

Introduction to Resource Bundles

- Resource-facing abstraction “below” Pilots
- Aggregated resources: compute, storage, network
- Objective: model of infrastructure resources
 - predictive, flexible, adaptive, active
- **Bundle profile**
 - resource histograms: using probes and workload measurements
- **Bundle signature**
 - compact representation
- **Bundle allocation**
 - late binding of bundle to pilots

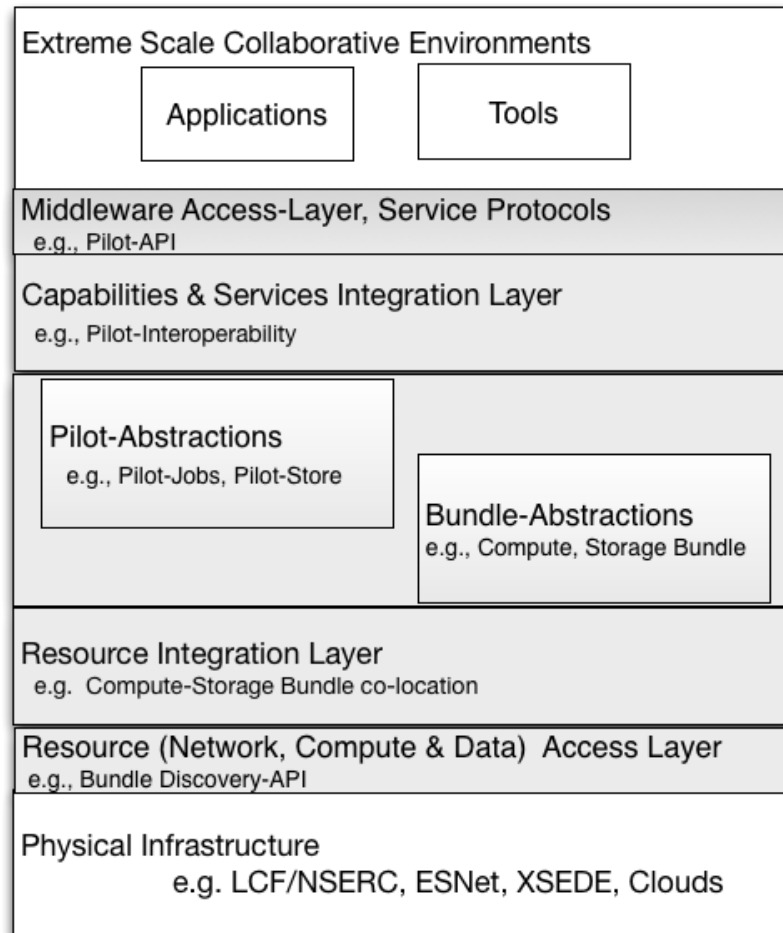
AIMES

Shape of the solution

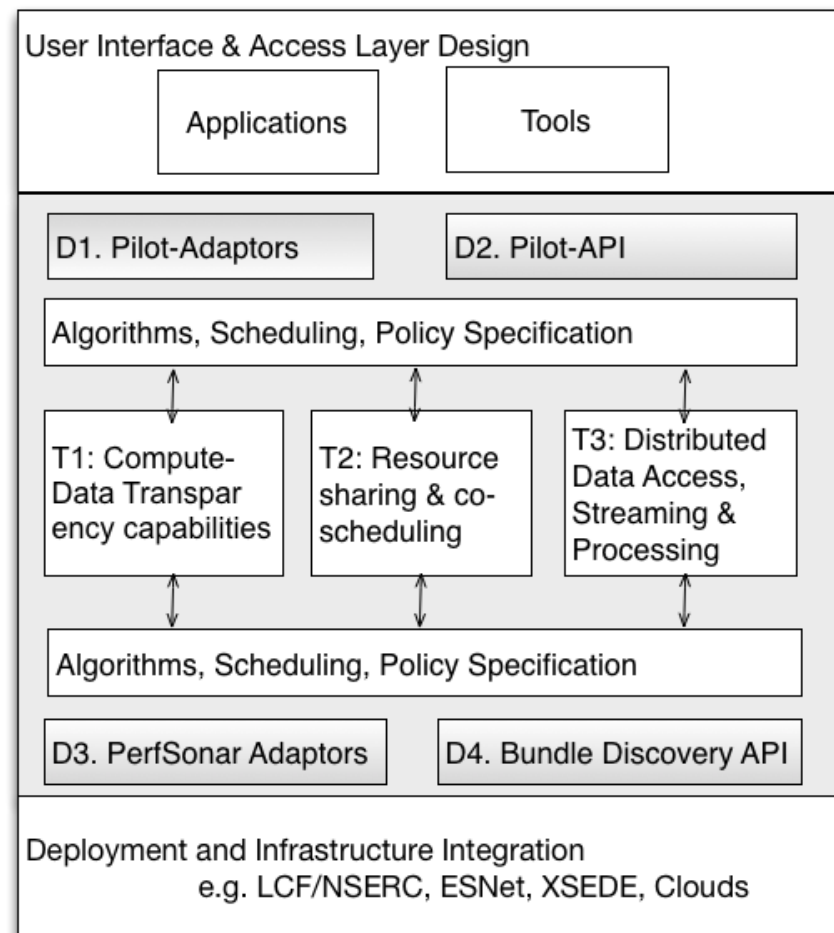
- Resource abstractions at application- and resource-facing levels
- Awareness of application + Awareness of physical resources
- Co-design approach
- Integrative resource management
 - Across compute (C), storage (S), networking (N)
 - Across resource types/scales/federation
 - Vertical: abstractions at application and resource level are aware of each other
 - Horizontal: C, S and N aware

AIMES

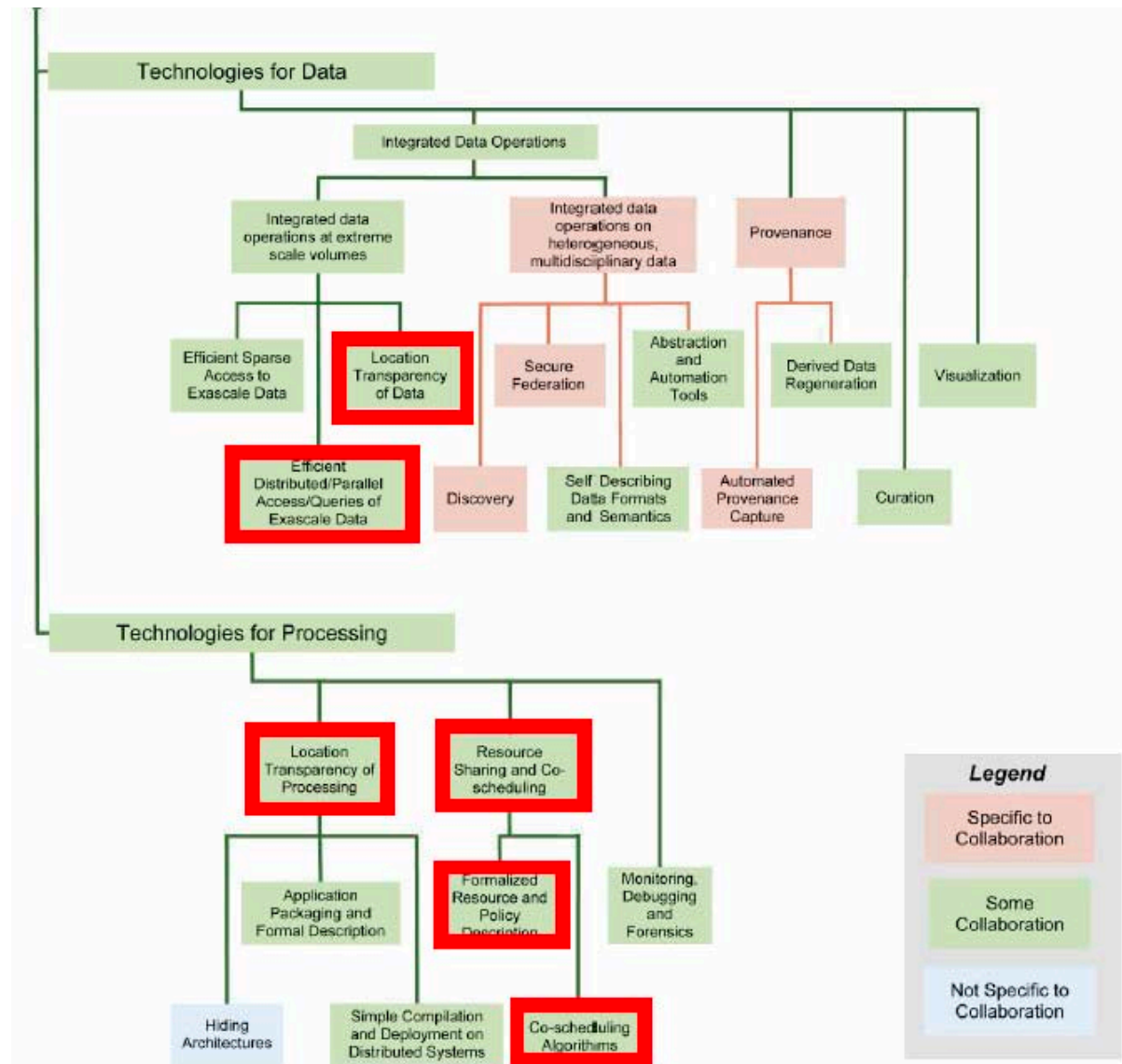
- High-level architecture



- Functional view



AIMES – FOA alignment



Short-term

- Create skeleton that represents Montage
- Define capabilities of bundles
- Define the SoC and API between Pilots and Bundles
- Prototype demonstration:
 - I. Focus on compute resources: Pilots and Bundles
 - II. Integrate Skeleton + Pilot-Abstractions
 - III. Integrate Pilot-Abstractions with Bundles
 - IV. Identify Experimental Infrastructure
 - V. Instantiate montage-skeleton for different parameter range and execute over heterogeneous infrastructure

Middle-term

- **Middle Term:** Develop storage-based Pilot/Bundles; Integrate Pilot/Bundle abstractions with network monitoring tools - e.g. perfsonar
- Understand the role of dynamic network capabilities as an element of resource management in conjunction with pilot/bundle abstractions
 - Obtain skeletons for network and storage-based applications
 - Investigate integrating network scheduling capabilities
 - Explore specific technologies such as NSI, SDN
 - Inform and integrate with groups at OGF

AIMES: Abstractions and Integrated Middleware for Extreme-Scales

Shantenu Jha, Matteo Turilli - Rutgers U

Jon Weissman - U Minnesota

Daniel S. Katz, Zhao Zhang - U Chicago

Thank you



RUTGERS



UNIVERSITY OF MINNESOTA



THE UNIVERSITY OF
CHICAGO

Middle and Long-term

- **Middle Term:** Develop storage-based Pilot/Bundles; Integrate Pilot/Bundle abstractions with network monitoring tools - e.g. perfsonar
- Understand the role of dynamic network capabilities as an element of resource management in conjunction with pilot/bundle abstractions
 - Obtain skeletons for network and storage-based applications
 - Investigate integrating network scheduling capabilities
 - Explore specific technologies such as NSI, SDN
 - Inform and integrate with groups at OGF
- **Long Term:** Provide prototype and deployment of network aware pilot/bundle based resource management
- Reason about performance, trade-offs, configurations and composability
 - What are possible models for distributed applications and (next-generation) DCIs?
 - What properties/requirements/constraints does this impose on middleware?