Accelerating Parameter Sweep Workflows by Utilizing Ad-hoc Network Computing Resources: an Ecological Example

Jianwu Wang[1], Ilkay Altintas[1], Parviez R. Hosseini[2],
Derik Barseghian[2], Daniel Crawl[1], Chad Berkley[3], Matthew B. Jones[3]

[1] *San Diego Supercomputer Center, UCSD, U.S.A.*
[2] *The Consortium for Conservation Medicine, Wildlife Trust, U.S.A.*
[3] *National Center for Ecological Analysis and Synthesis, UCSB, U.S.A.*

Outline

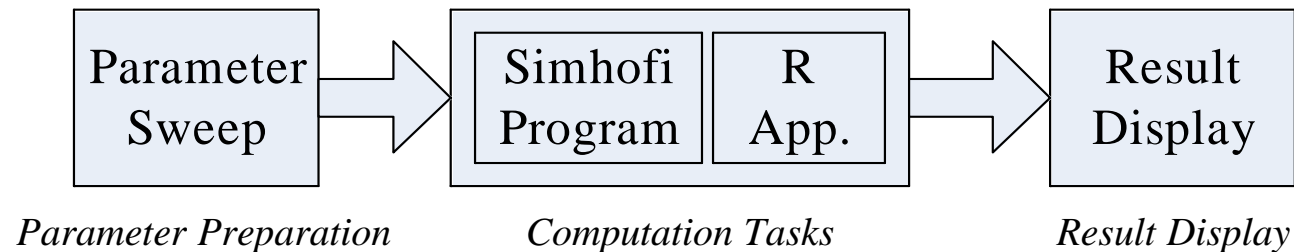- **Introduction**

- **Theoretical Ecology Use Case**

- **Background**
    - Kepler
    - Master-Slave Architecture

- **Our Approach**
    - Distributed Composite Actor
    - Provenance Collection

- **Results**

- **Conclusion and Future Work**

Introduction

- Many scientific computing problems have linear or greater time complexity based on parameter configuration ranges

- Domain scientists should be able to easily leverage distributed computing resources with little knowledge of the underlying techniques

- We will discuss a distributed execution framework, called Master-Slave Distribution, to distribute sub-workflows to ad-hoc network computing resources

Theoretical Ecology Use Case

| Parameter Sweep | Simhofi Program | R App. | Result Display |

Parameter Preparation　　　　*Computation Tasks*　　　　*Result Display*

- **Characteristics of the use case**
  - Parameter Sweep: independent multiple execution, i.e., "embarrassingly parallel problems"
  - Smooth Transition of Computation Environments
  - Partial Workflow Distribution
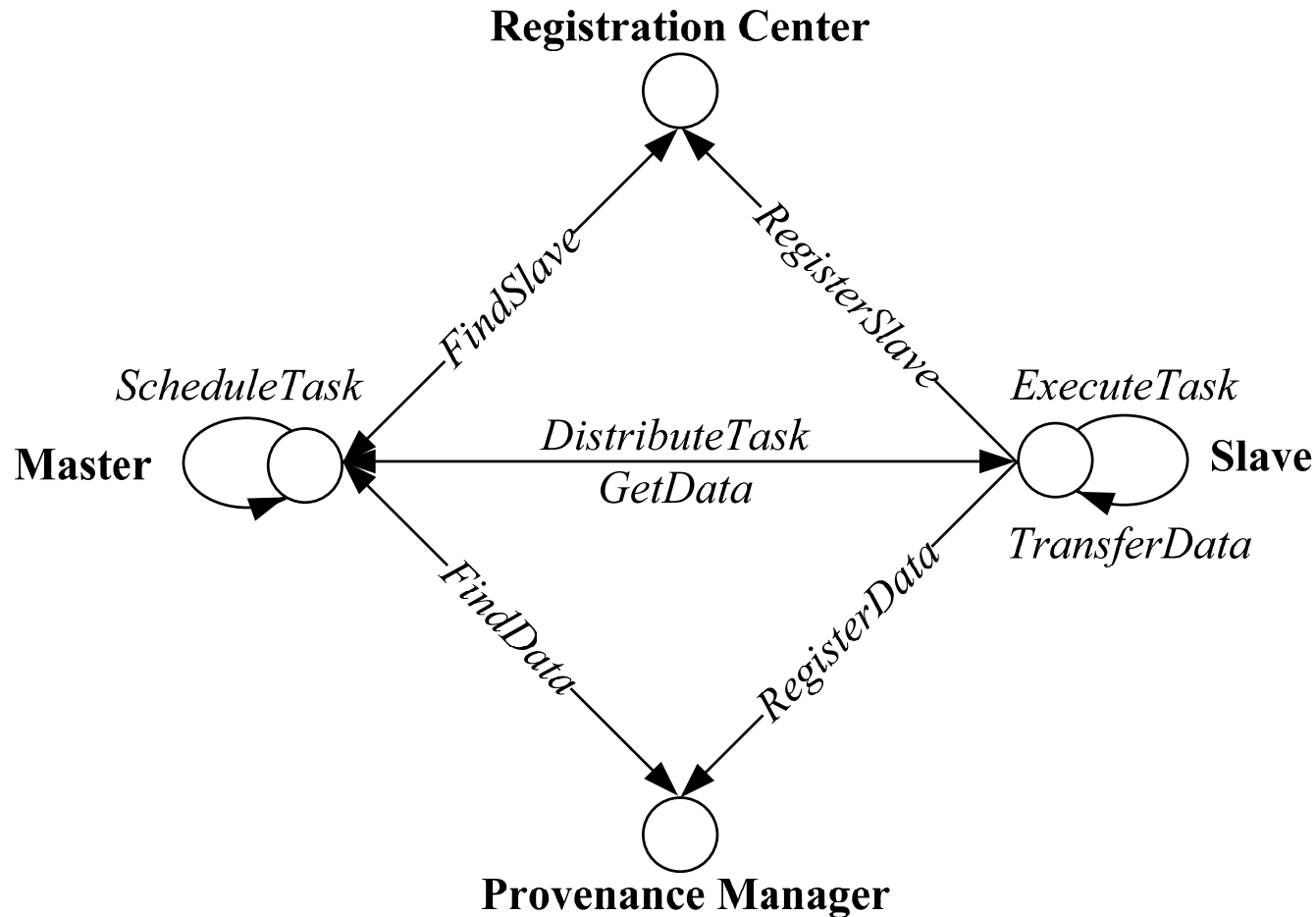 - Provenance Collection

Kepler

Background – Kepler

- ## Actor-oriented Modeling
  - All these actors inherit the same interfaces, such as *prefire()*, *fire()* and *postfire()*

- ## Model of Computation
  - Synchronous Data Flow (SDF) director: actors execute sequentially
  - Process Network (PN) director: each actor has its own execution thread and execute in parallel

Background – Conceptual Architecture for Workflow Distributed Execution

Registration Center

FindSlave

RegisterSlave

ScheduleTask

Master

DistributeTask
GetData

ExecuteTask

Slave

TransferData

FindData

*RegisterData*

Provenance Manager

Our Approach – Distributed Composite Actor

- As the role of Master, each token received by this Actor is distributed to a Slave node, executed, and the results returned.
- Different behavior with different computation models

A High-Level Distributed Execution Framework for Scientific Workflows
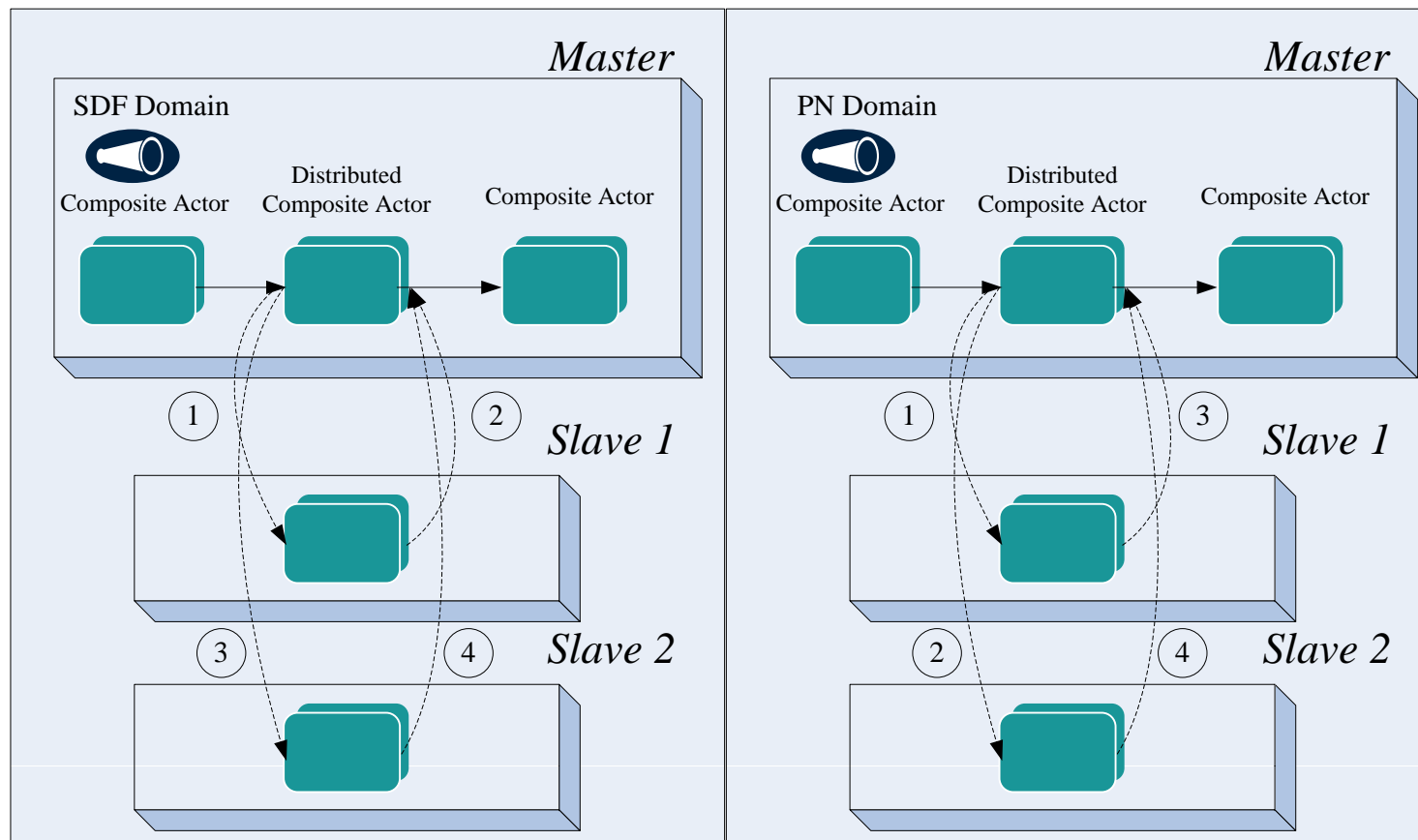
Our Approach – Provenance Collection

- By collecting workflow structure and executions, our provenance framework make it easier for users to track data files for large parameter sweeps
- It can be configured to support centralized or decentralized provenance information recording
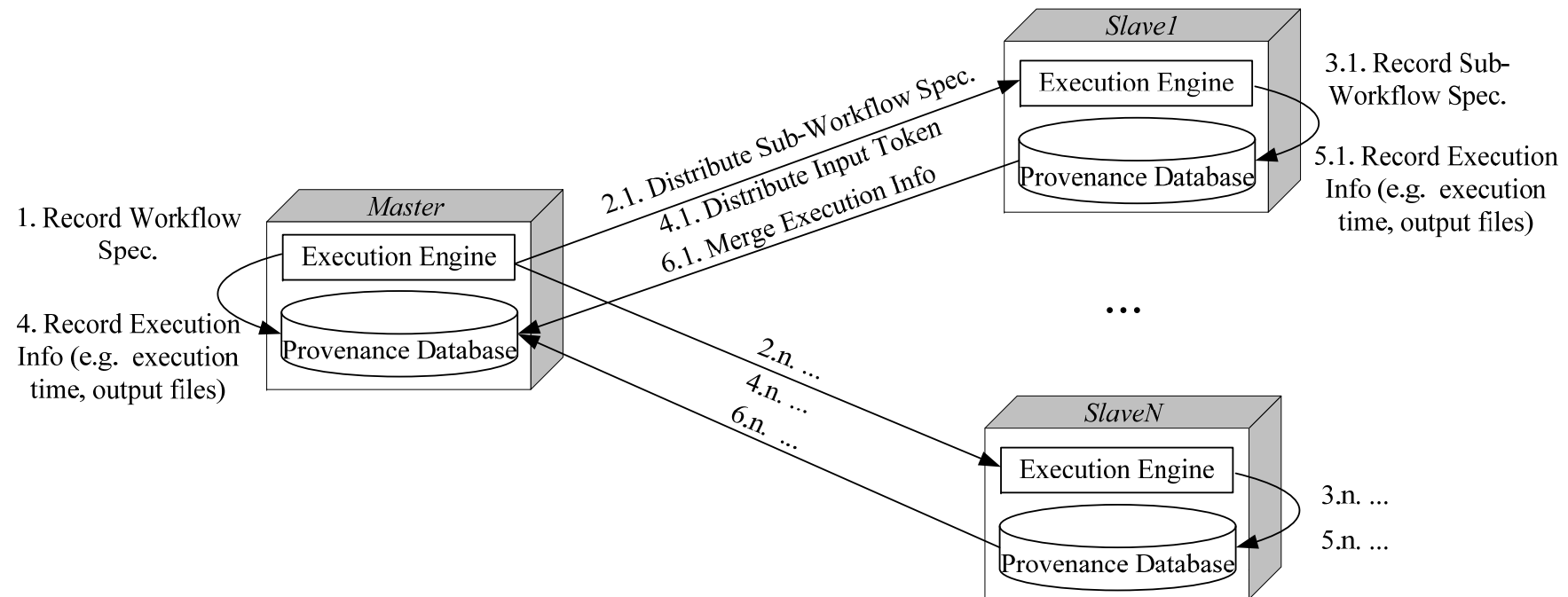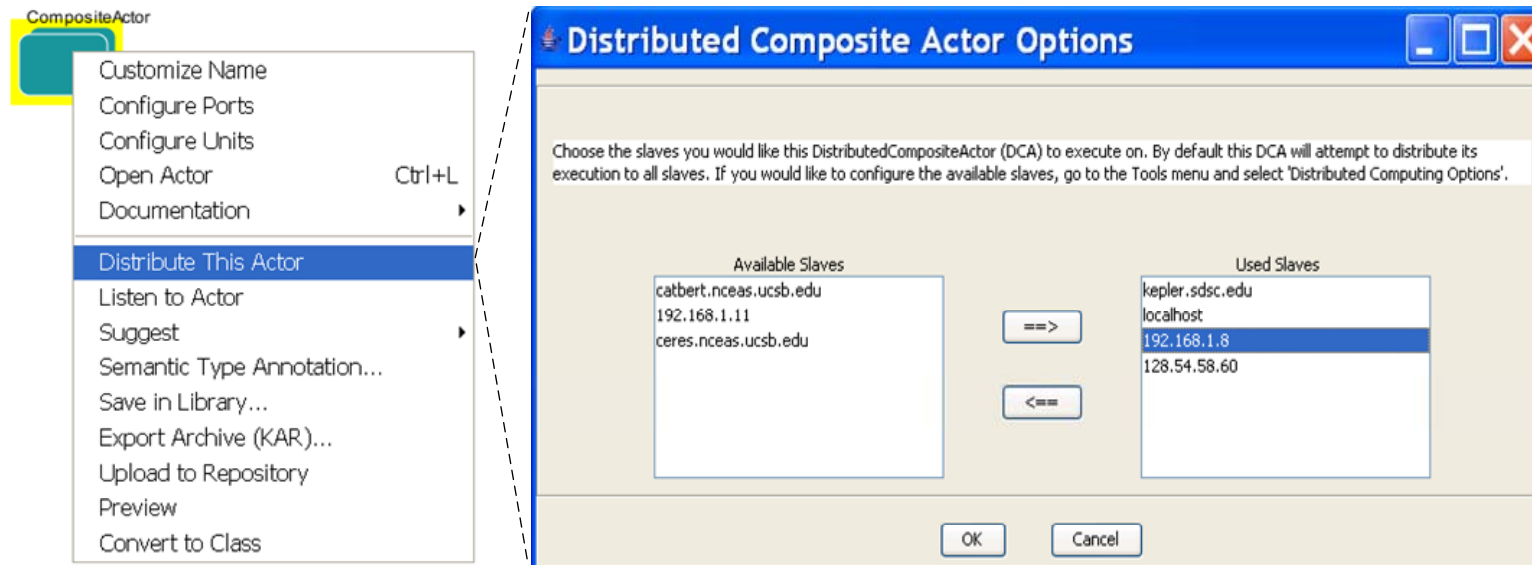
A High-Level Distributed Execution Framework for Scientific Workflows

Results – Workflow

Results – Usability

- **Users use the DistributedCompositeActor just like the common composite actor**

- **Interaction for execution environment transition**

A High-Level Distributed Execution Framework for Scientific Workflows

Results – Experiment

Parameters	Execution Time (minutes)			
	SDF locally	**PN locally**	**SDF Master-Slave**	**PN Master-Slave**
b=<0.1, 0.2, 0.3, 0.4>, d=<2.5, 3.5>, X=3, Y=3, E=4	0.39	0.35	0.60	0.52
b=<0.1, 0.2, 0.3, 0.4>, d=<2.5, 3.5>, X=8, Y=8, E=30	32.21	32.24	19.05	9.38
b=<0.1, 0.2, 0.3, 0.4>, d=<2.5, 3.5>, X=16, Y=16, E=30	502.2	510	309	147
Testbed Constitution				
	OS	Memory	CPU	
Notebook	Window XP	2 GB	2.00 GHz Duo Core	
Desktop	Mac OS X	2 GB	2.80 GHz Duo Core	
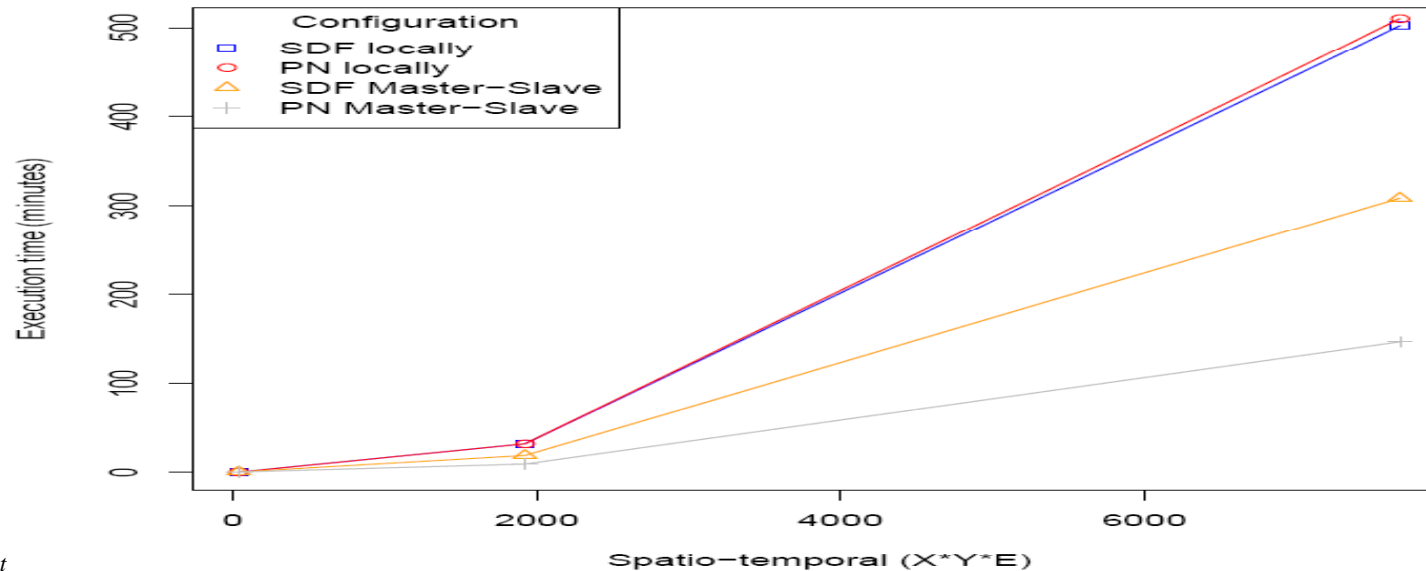
Conclusion and Future Work

- **A distributed execution framework in the Kepler**
  - Distribute sub-workflows to ad-hoc network computing resources
  - Applicable to parameter sweep applications to realize parallel independent execution

- **Future Work**
  - Generalize for Cluster, Grid, and Cloud platforms.
  - Categorize different distributed approaches in Kepler to match different requirements

A High-Level Distributed Execution Framework for Scientific Workflows

- **Thanks!**

- **For More Information:**
  - Distributed Execution Interest Group of Kepler: https://dev.kepler-project.org/developers/interest-groups/distributed

  - Contact: jianwu@sdsc.edu

Kepler