

# A Federation Architecture for DETER

Ted Faber, John Wroclawski, Kevin Lahey,  
John Hickey

`{faber,jtw,lahey,jhickey}@isi.edu`

University of Southern California  
Information Sciences Institute





# Outline

---

*The Federation Problem*

A Model for DETER Federation

A Simple Prototype





# Federation: What and Why

---

Sharing Distributed Experimental Resources  
Resource Admin and Policy Creates Specialization  
Examples

- Many DETER Nodes

- Customized “Lego” Apparatus

  - Malware Capture Facility and High-containment Testbed

  - Trace Collection Facility and Trace-driven Emulation





# Federation Challenges

---

## Shared Control of Resources

- Respect Local Admin/Policy

- Compose New Facilities

## Resource Discovery (at Scale)

- Describe Resources and Properties

- Find Resources with Properties



USC **Viterbi**  
School of Engineering



# More Federation Challenges

---

Distributed Composition of a Unified Facility

Create Consolidated Tool from Multiple Testbeds

Manipulate Distributed Resources Intuitively

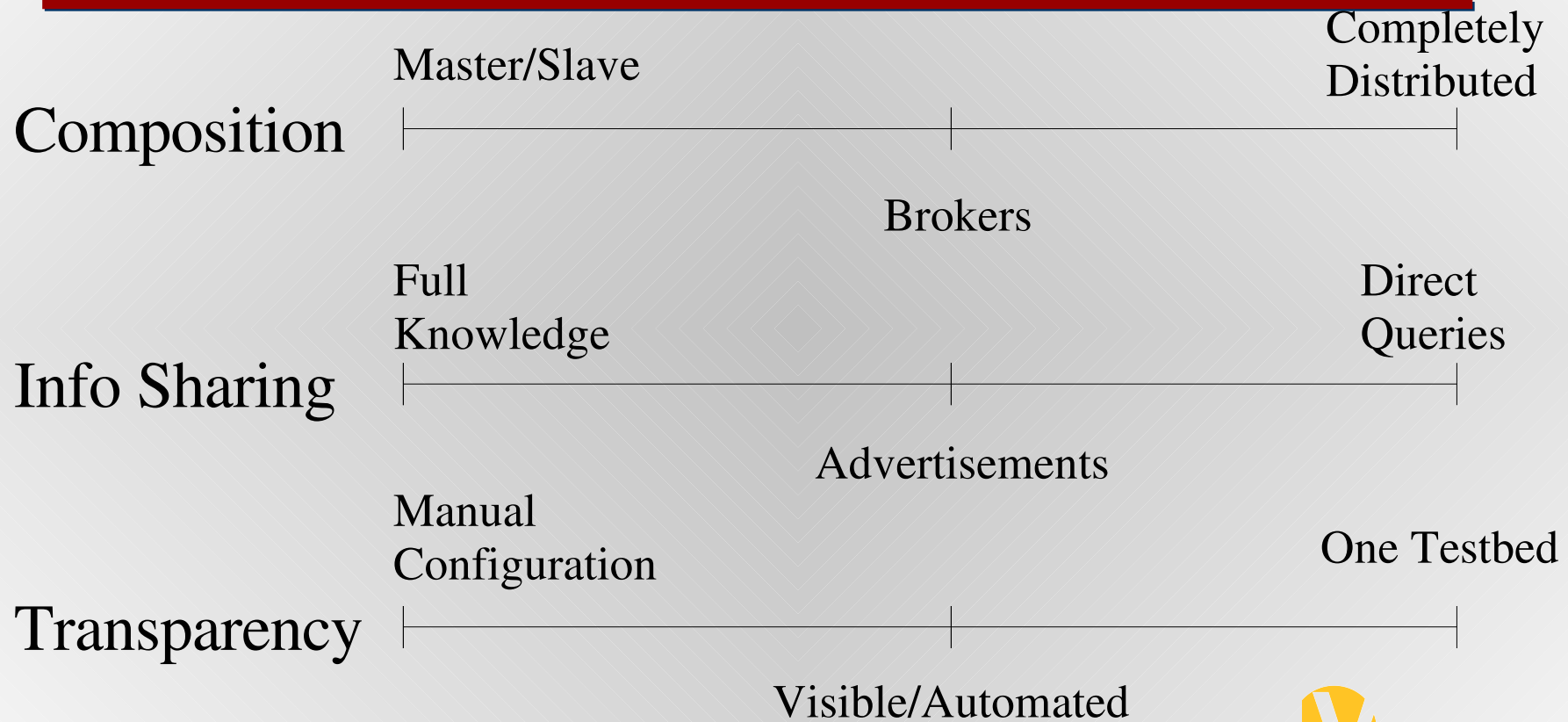
(ISI Prototyping Here)



USC **Viterbi**  
School of Engineering



# Federation Solution Space





# Outline

---

The Federation Problem

*A Model for DETER Federation*

A Simple Prototype





# A DETER Federation Architecture

---

Apparatus Configuration Layer

Resource Discovery Unbound

Distributed Administration Incorporated

Coordinates:

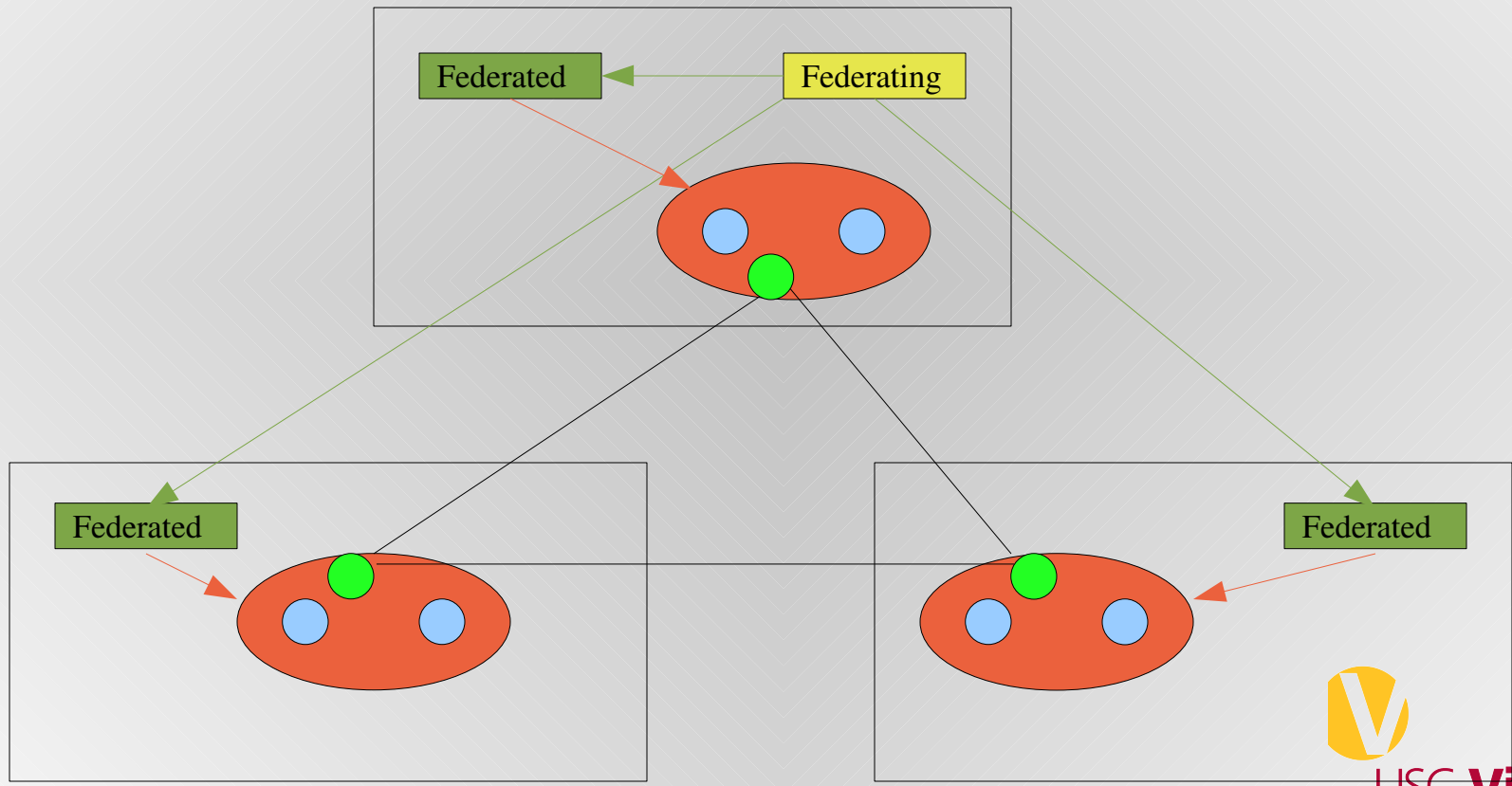
(Primary/Secondary, Direct Query, Visible)

Focus On Combining DETER Namespaces/Features





# Shape of the Architecture





# Testbed Features to Compose

---

## Features:

Connectivity (IP addresses & tunneling) (experiment)

Storage (shared filesystem) (project)

Principals (user IDs/accounts) (project)

Host Configuration (system images) (experiment)

Experiment Control (event system) (experiment)





# Outline

---

The Federation Problem

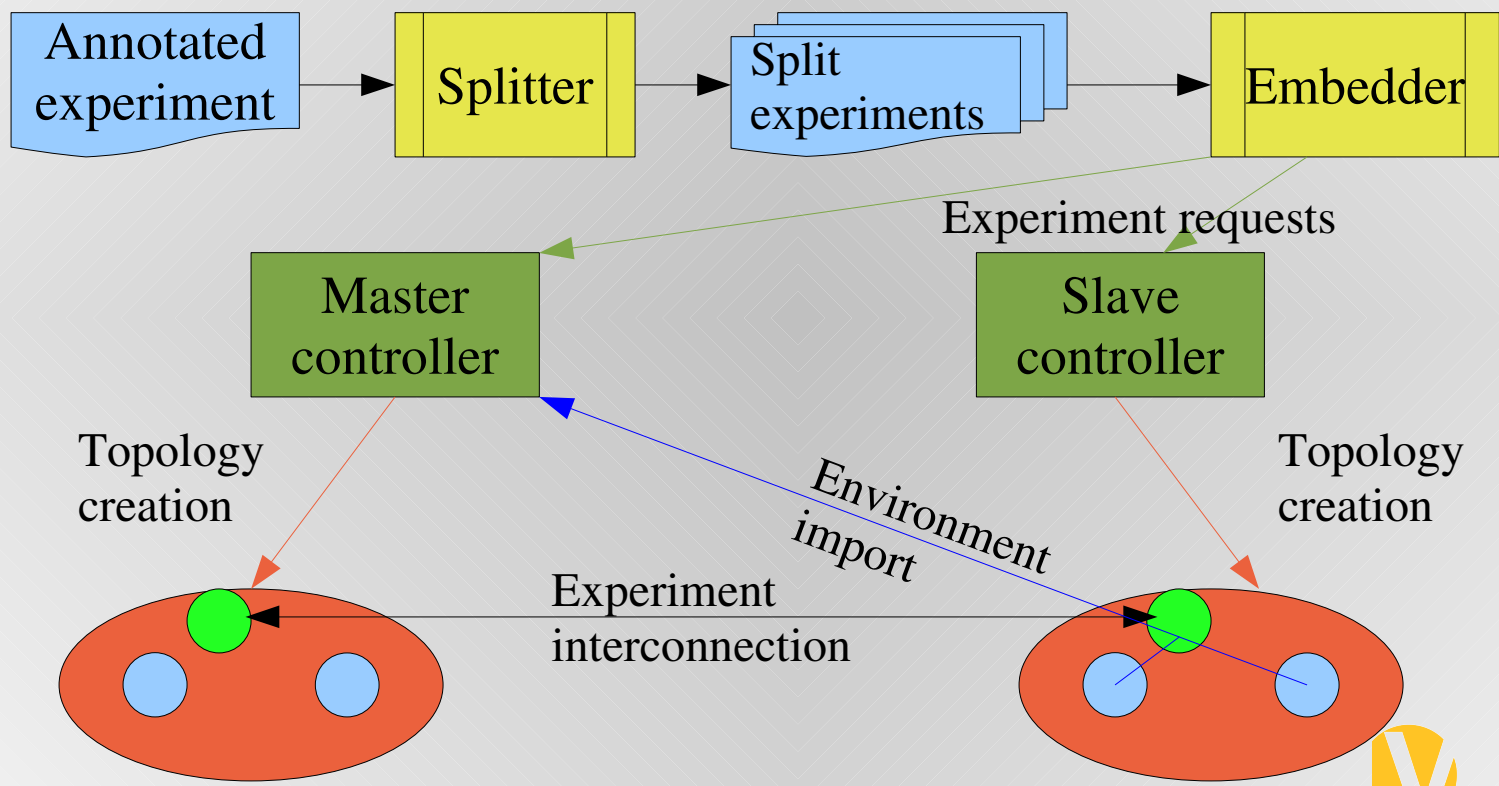
A Model for DETER Federation

*A Simple Prototype*





# The Prototype





# Prototype Components

---

## Experiment Partition

Parser splits annotated ns files into sub-experiments

Sub-experiments use local names

## Centralized Synchronization

Testbeds configured through SSH/XML-RPC

Basic failure handling



**USC Viterbi**  
School of Engineering



# More Prototype Components

---

## Connectors: Experiment and Control

- IP over SSH tunnel

- SSH tunnel to Primary Users/Boss

## Node Configuration

- Import user accounts

- Mount remote file systems





# Conclusions

---

## Simple Working Prototype

Prototype vs. implementation?

## Extending the Prototype

Event system

Connector interfaces

## Extending the Architecture

Multiple simultaneous embeddings/allocations

Higher level experiment descriptions



USC **Viterbi**  
School of Engineering



# Prototyped Combinations

---

## Connectivity

Merged IP space – IP tunneling

## Storage

Primary filesystem exported

## Principals

Primary testbed exported

Experiment Control (SEER/event system)

Host Configuration (system images)



USC **Viterbi**  
School of Engineering