

The iLab Service Broker: What it is and How it works

1st iLab Europe Workshop
November 16 - 18, 2009



The Role of the iLab Service Broker

The iLab Service Broker...

- ▶ Provides generic support services for iLabs
- ▶ Provides a standard, distributed platform for lab deployment, eases sharing
- ▶ Enables single sign-on for students who use multiple iLabs



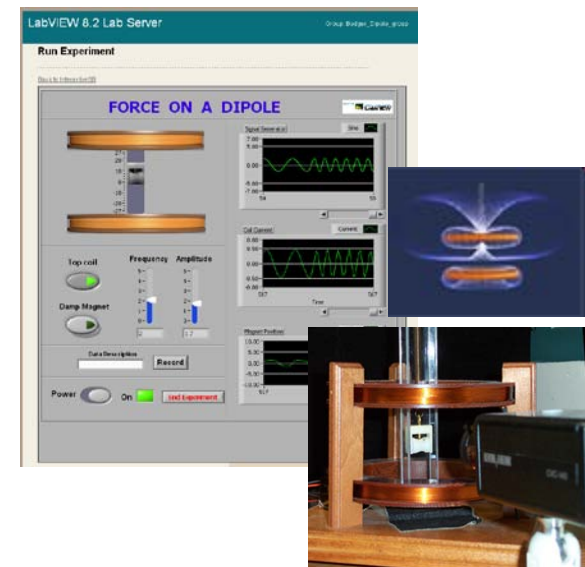
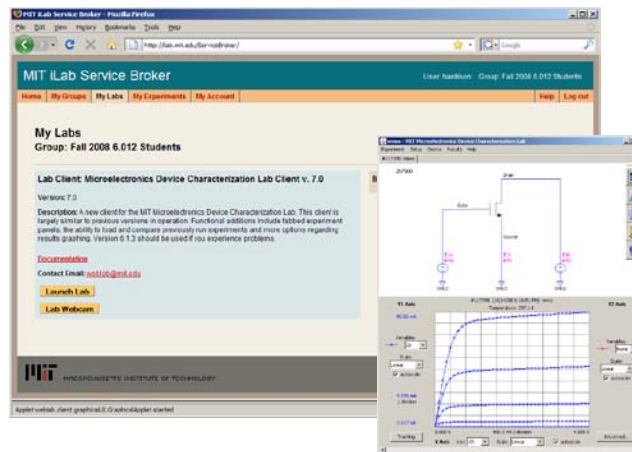
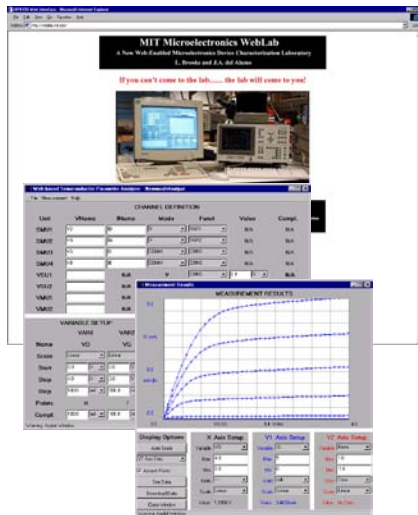
A Brief History of the iLab Project

1998: First Individual Labs Developed

2004: iLab Shared Architecture (ISA), Compliant Batched Labs Released

2007: Support for Interactive Experiments Added to ISA

Adoption of ISA increasing worldwide



Batched and Interactive iLabs

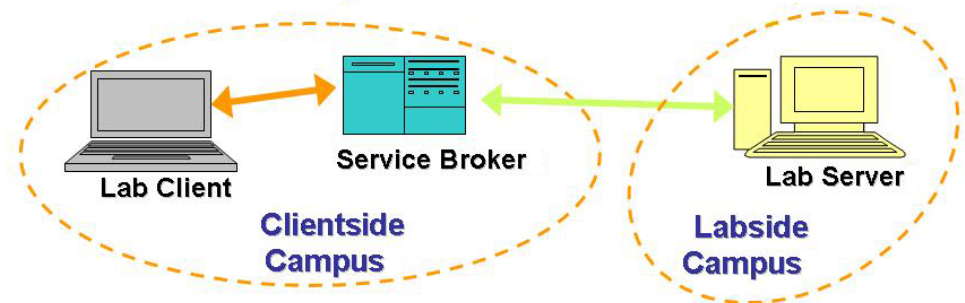
A Batched iLab does not require real-time control of lab equipment, permits certain simplifying assumptions.

- ▶ **Communication model**

- ▶ Web services can be used for lab-specific communications
- ▶ All client-lab server communications routed through Service Broker – authentication and data storage become easy

- ▶ **Execution model**

- ▶ Generally, experiments execute quickly; can be queued



Interactive iLabs are a little more complex

Supporting Interactive Experiments

Interactive experiments, by their nature...

- ▶ ...require real-time control.

Direct client to lab server connection needed
(possibly high-bandwidth, possibly proprietary)

- ▶ ...are performed in human-time.

Longer periods of single user control

A revision to the iLab Shared Architecture (and the Service Broker) was necessary to provide services for interactive experiments.

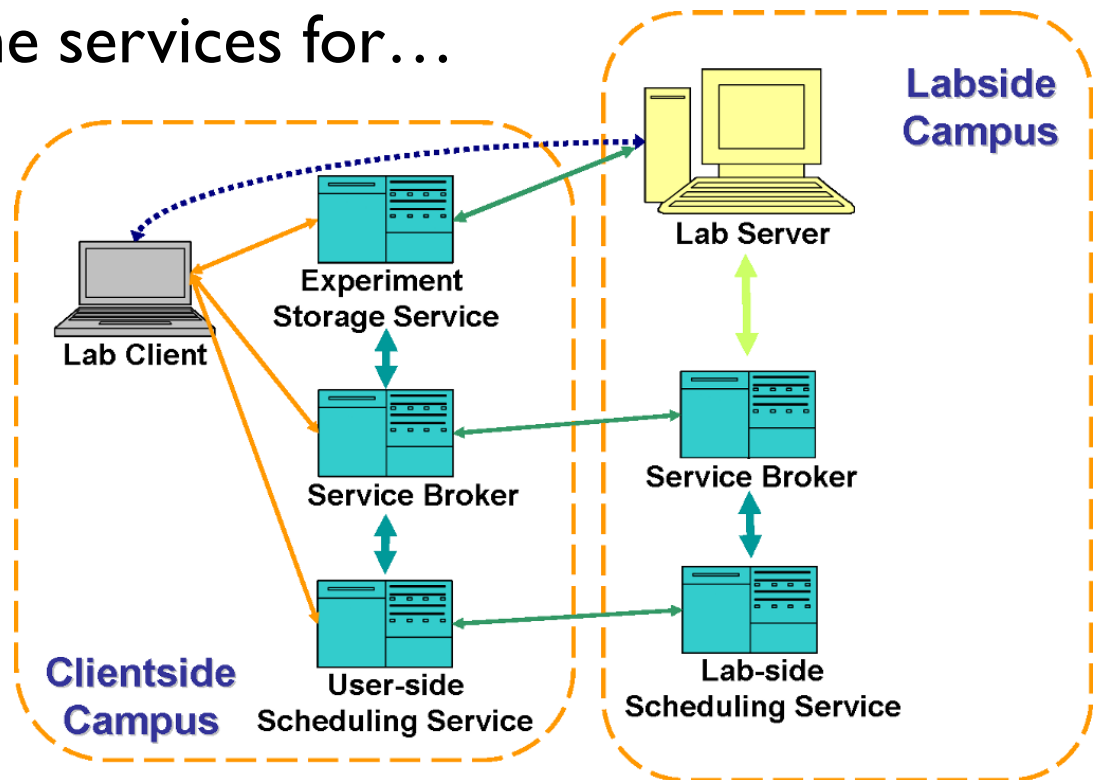


Topology of Interactive Experiments in the iLab Shared Architecture

iLab Service Broker orchestrates access to a set of distributed, stand-alone services for...

- ▶ Experiment Storage
- ▶ Scheduling
- ▶ Cross-service authentication (Ticketing)

Service Broker sets up the experiment session and then steps out of the way.



The Experiment Storage Service (ESS)

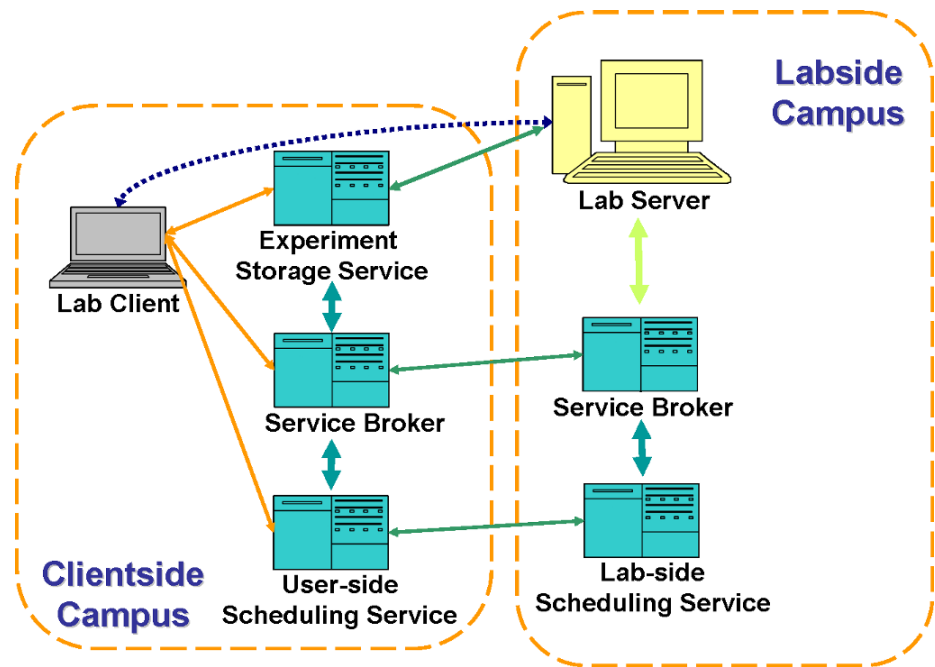
A generic stand-alone service responsible storing student experiment information

- ▶ Lab configurations, input parameters, results

Service Brokers, Lab Servers and Clients all interface to an ESS

- ▶ Service Broker responsible for record management
- ▶ Clients & Lab Servers contribute to the record

A single ESS may service many iLab installations

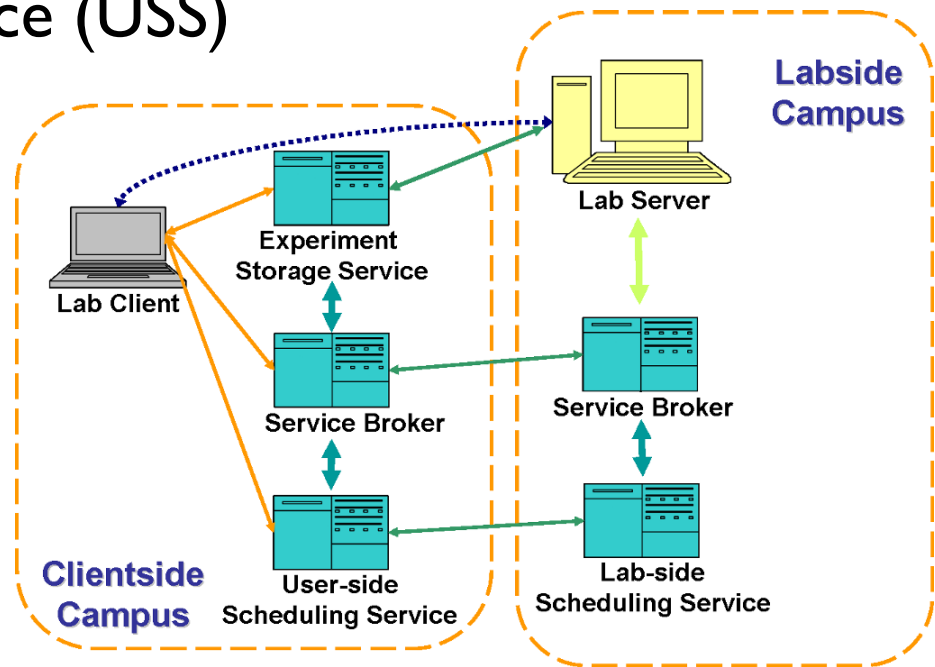


Scheduling Services

Used to manage student access to the lab. Composed of two complementary services:

- ▶ **Lab-side Scheduling Service (LSS)**
 - ▶ Allocates available lab time to sets of students.
- ▶ **User-side Scheduling Service (USS)**
 - ▶ Reserves blocks of available time for individual student use.

Service Brokers manage connections between USS and LSS pairs



One Service Broker to Rule them All...

Initial Interactive-capable Service Brokers did not support Batched iLabs – not good!

Recently, a “merged” Service Broker has been released

- ▶ Contains new support services, uses as appropriate
- ▶ Provides support for both Interactive and Batched iLabs
- ▶ Enables newer Batched labs to leverage new services

Reference implementation still under active development, but all development and deployments will be based on this framework.

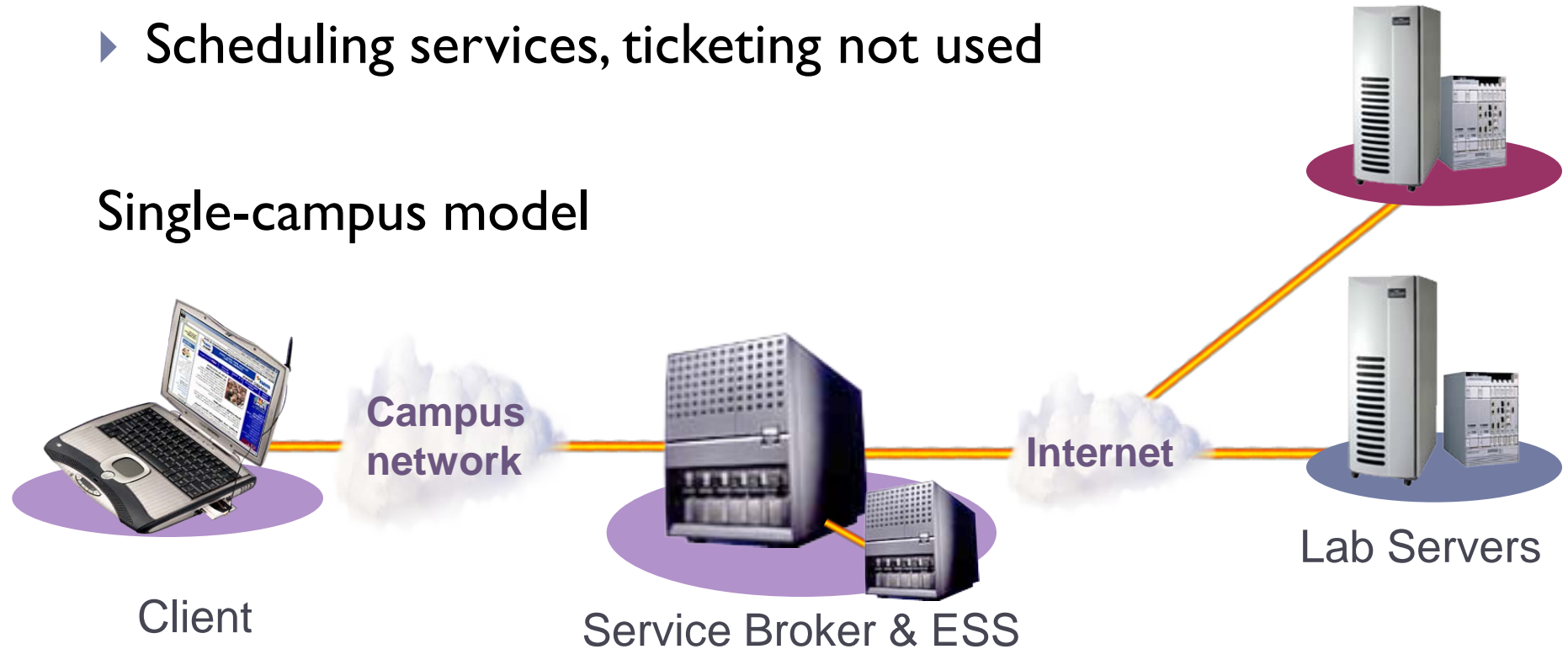


Supporting Batched iLabs

Merged Service Broker operates similarly to the original broker implementations

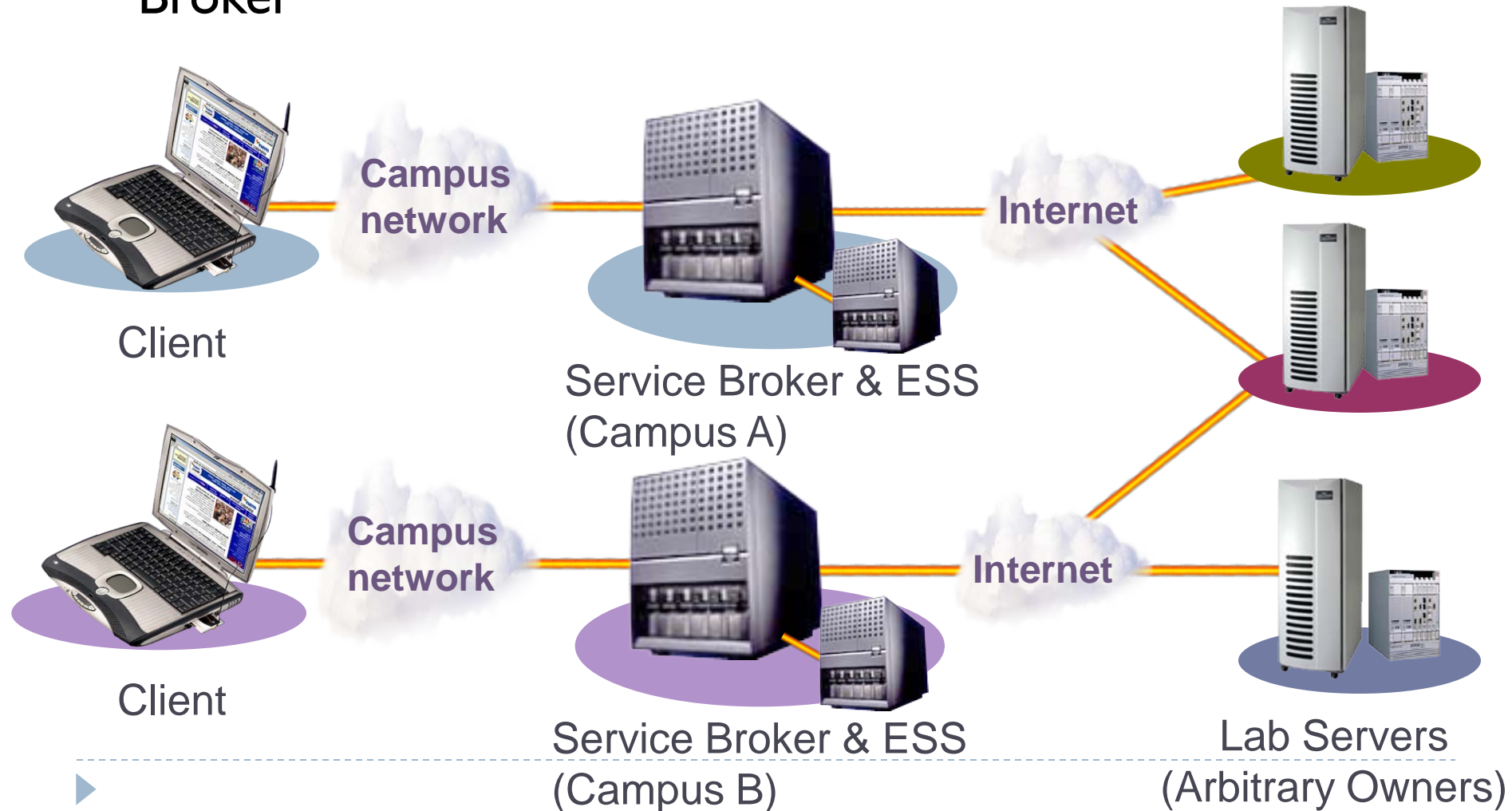
- ▶ Lab Servers/Clients are registered in a similar manner
- ▶ Scheduling services, ticketing not used

Single-campus model



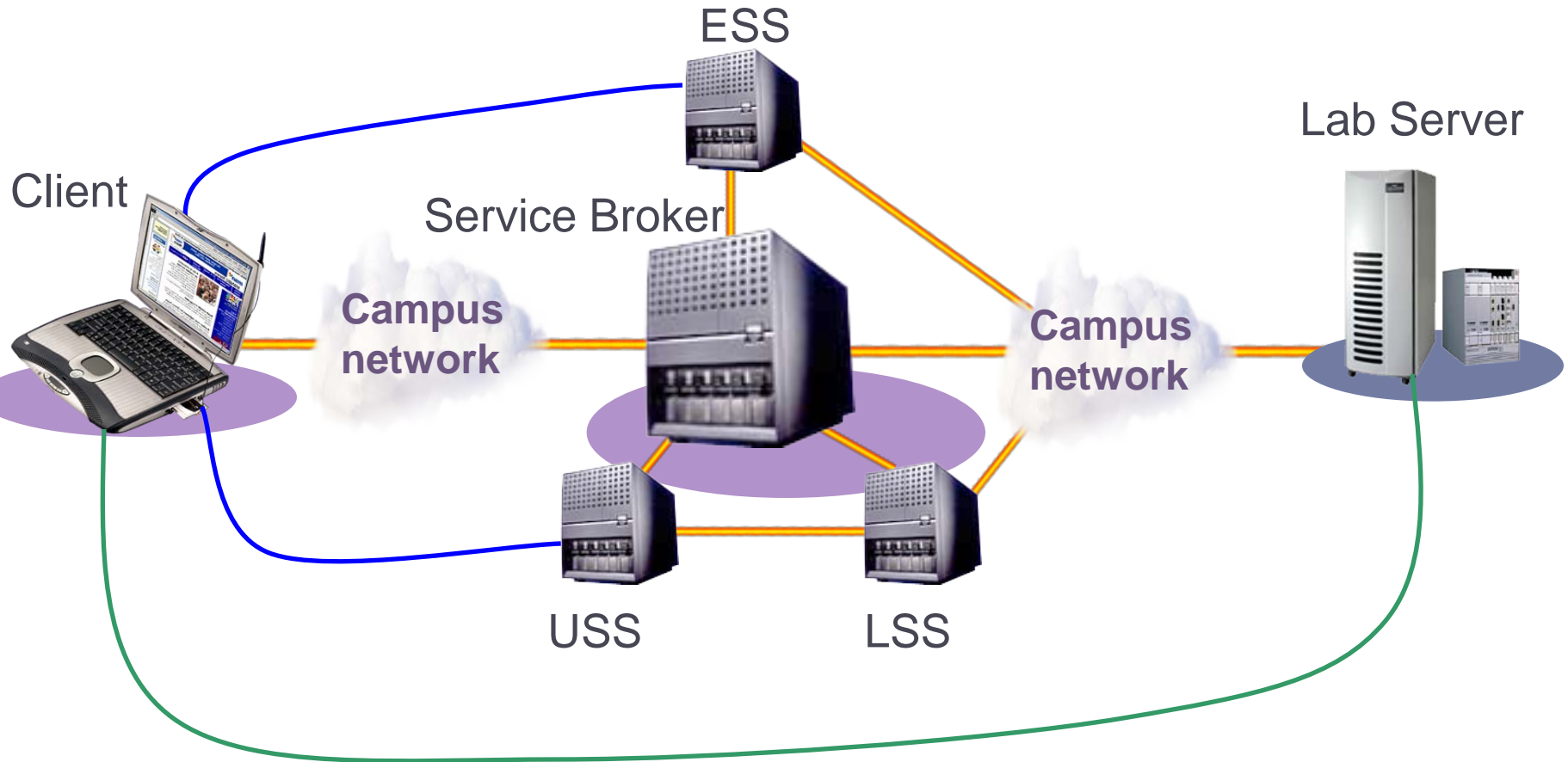
Supporting Batched iLabs

Multi-campus model – same as with “batched” Service Broker



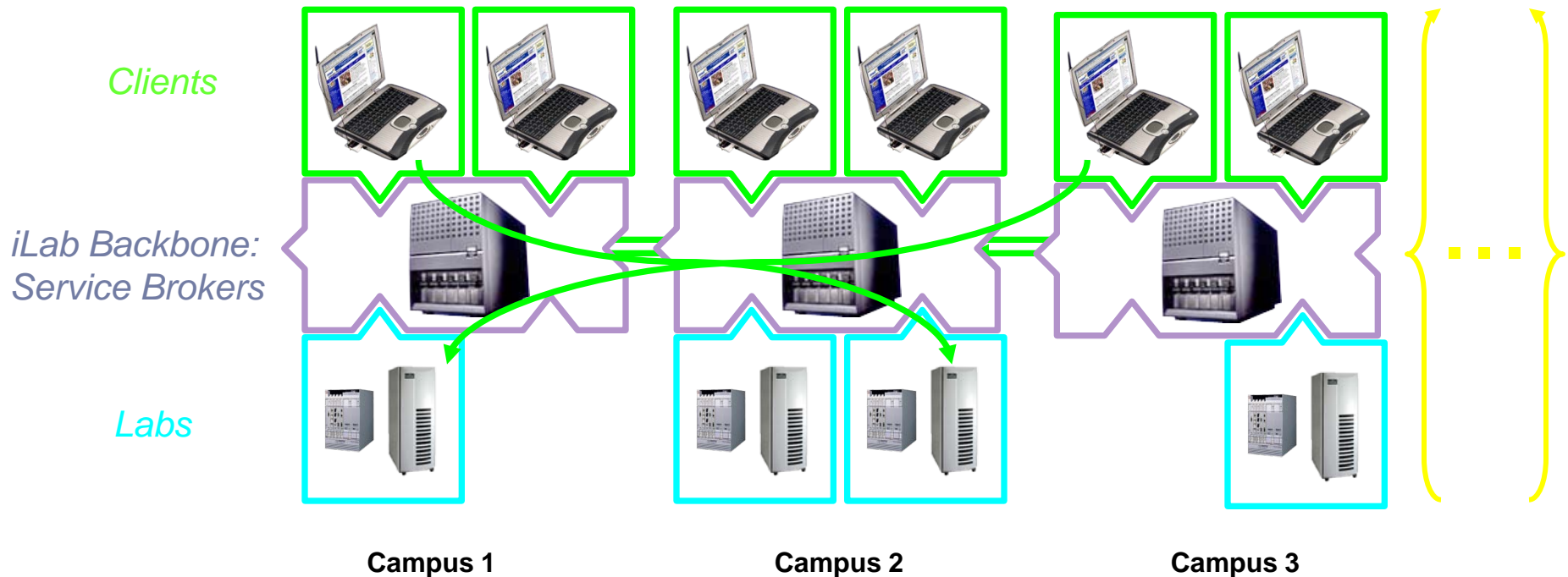
Supporting Interactive iLabs

Single-campus model – uses all support services



Supporting Interactive iLabs

Multi-campus case – trust relationships are set up between Service Brokers. Brokers act as iLab communication backbone



Demonstration

- ▶ Batched and Interactive iLabs on a “merged” Service Broker.
- ▶ Tomorrow, we’ll go into more detail on how to install and configure the Service Broker and its associated services.

