iLab:

A Scalable Architecture for Sharing Online Experiments

An MIT iCampus Project

MIT

V. Judson Harward, Jesús A. del Alamo, Steven R. Lerman, et al.
MIT

ICEE 2004, Gainesville
18 October 2004

Statement of the Problem

- There is enormous educational value in hands-on laboratory experiences, but…
- … conventional laboratories are expensive and have complex logistics:
  - Scheduling, equipment cost, lab space, lab staffing, training, safety
- … conventional labs don't scale well and can't easily be shared
  - All institutions must own all labs

Solution: Online Laboratories

- *Online laboratory ("iLab" or "WebLab"):* a real laboratory that is accessed through the Internet from anywhere at any time
  - Not a "virtual laboratory" (simulations)
  - Not a "canned experiment" (a "one-click" lab)
- Online laboratories can deliver many of the educational benefits of hands-on experimentation

iLabs at MIT

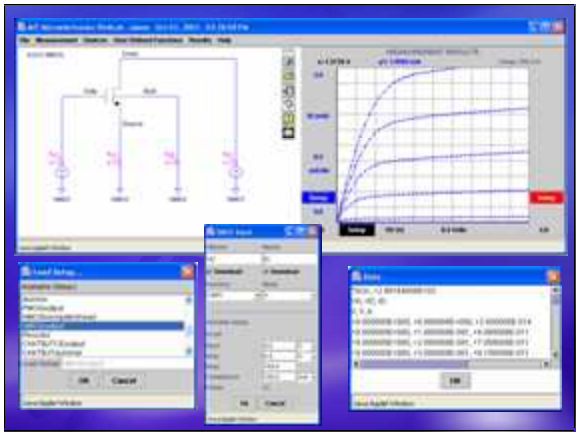
Dynamic signal analyzer *(EECS, to be deployed 2004)*

Polymer crystallization *(Chem. E., deployed 2003)*

Shake table *(Civil Eng., to be deployed 2004)*

Microelectronics device characterization *(EECS, deployed 1998)*

Heat exchanger *(Chem. E., deployed 2001)*



Educational Experiments

- MIT graduate and undergraduate courses since Fall 1998
- NUS (Singapore, 11 time zones), Fall 2000-03 (20-30 st/yr)
- Chalmers U. (Sweden, 6 time zones), Spring 2003-04 (250 st/yr)
- NTU Athens (Greece, 8 time zones), Spring 2004 (35 st/yr)
- Over 2000 student users (for credit) since 1998

Revolutionary consequences

- Order-of-magnitude more laboratory experiences available to students
- Can afford sophisticated labs involving:
  - advanced instrumentation
  - rare materials
  - unreachable locations
- iLabs embedded inside rich educational platforms containing:
  - visualization tools, simulations, data processing
  - remote collaboration and tutoring.

Revolutionary consequences

- iLabs will spawn communities of learners to share:
  - hardware
  - *and* educational content
- Institutions in the *developed* world can support educational needs of the *developing* world.

What Have We Learned?

Ad-hoc iLab development and management does not scale

- *iLab developers* are not IT specialists and want to minimize development work (want to reuse generic lab components)
- *iLab managers* do not want to deal with individual user management
- *iLab consumers* want to see single portal to multiple labs

➢ Need an *iLab Shared Architecture*

Shared Architecture Design Goals

- Scaling usage of online labs to a large number of users
- Allowing universities to share access to equipment
- Single sign on to labs at multiple universities
- Freeing lab owner/operator from administration (i.e. authentication, authorization, storage of results, archiving of data, etc.) of users from other universities
- Allowing universities with diverse network infrastructures to interoperate and share resources

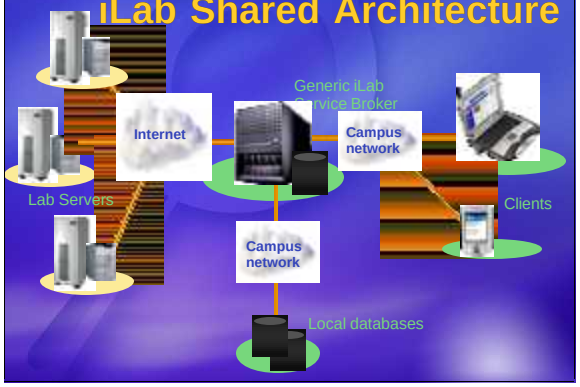
Project Boundaries

- Our architecture doesn't deal with specific hardware and software interfaces to lab equipment
- Our architecture is intended to be compatible and complementary with commercial software such as National Instruments LabView and analysis packages like Matlab

iLab Generic Services

- User authentication (and registration)
- User authorization and credential (group) management
- Experiment specification and result storage
- Lab access scheduling

iLab Shared Architecture



The Case for Web Services

- Web services represent the latest version of an old concept -- they allow one computer to invoke a procedure (method) on another.
- They are platform and vendor independent (we have already successfully bridged a Java client ⇔ a Windows XP/.NET Service Broker ⇔ a Windows 2000 lab server (with NI GPIB).
- Web services are self-describing and offer the promise of runtime discovery.
- Because they are usually based on http that we all use to access the web, they work well with campus networks.
- The iLab Shared Architecture builds on top of the current generation of web services.

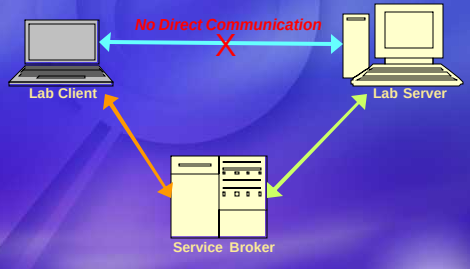
iLab Experiment Typology, 1

- Batched Experiments (2003-2004):
  - The entire specification of the experiment is determined before execution begins.
  - The user need not remain online while experiment executes.
- Interactive Experiments (2004-2005):
  - The student client portrays virtual lab equipment (GUI).
  - The student can interact with experiment throughout its course.

iLab Experiment Typology, 2

- Sensor Experiments (2005-2006):
  - Publish and subscribe based architecture
  - Triggers and event-driven data monitoring
  - Flexible data analysis
  - Data archive

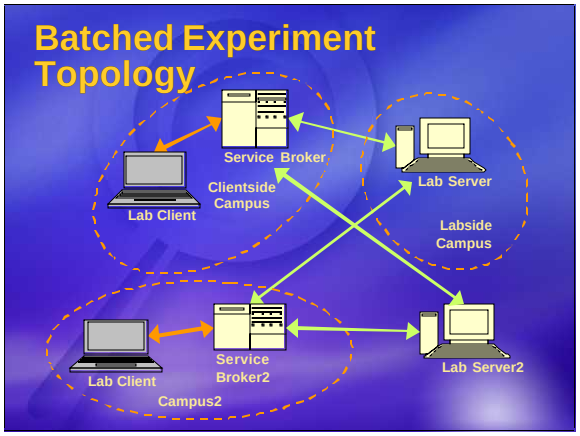
Batched Experiment Topology



What a Lab Provider Does Not Want To Do

- Register 100's of student accounts for other people's students.
- Store experiment results for students from other institutions and decide when they can be deleted or how to archive them.
- Decide who can view whose experiment results, especially when it involves setting policy for another university's courses.

Batched Experiment Topology



Service Broker Administrative Services

- Adding, modifying, and removing lab servers and clients.
- Adding, removing, or confirming user access.
- User management including assigning users to groups and modifying access rights.
- Managing experiment records.
- End of semester cleanup.

iLab Shared Architecture:

Beta-Test, Spring 2004

Lab: MIT Microelectronics WebLab, V. 6

Class: 6.012 Microelectronics Devices and Circuits (junior level in EECS)

Students: about 100

Assignments: 2

No major problems or down time

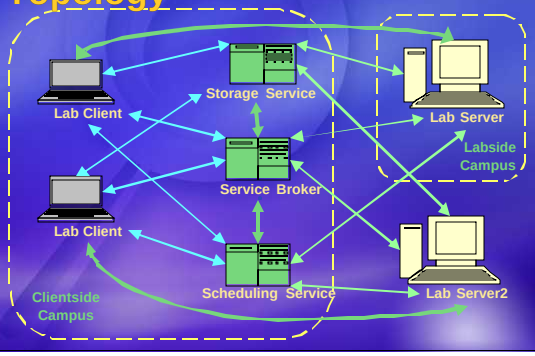
Preliminary Interactive Design

- We have been working on an extension of the Service Broker architecture for interactive experiments since February.
- Interactive experiments typically have duration; they must be scheduled rather than queued.
- Routing all client – lab server communication through the Service Broker is expensive and limits flexibility in client design.

New Services for Interactive Experiments

- A Scheduling Service
  - allows lab servers to let register blocks of time along with a signup policy;
  - allows client-side faculty to add further signup rules;
  - allows students to schedule lab sessions;
  - confirms student appointments on login;
- An Experiment Storage Service
  - allows any pre-registered iLab process (client or server) to store experiment records
- A Ticketing Service (ServiceBroker)
  - authorizes server/server and client/server communication.

Preliminary Interactive Topology



Future iLab Directions

- Our long term vision
- Plans for releasing iLab software and documentation in AY2005
- Collaborations with other universities
- Longer term iLab dissemination plans

Our Long Term Vision

- Creating a movement within higher education (and potentially other levels) leading to global sharing of laboratory experiments over the net
- Creating an informal "barter economy" to facilitate sharing of lab equipment
- Sharing beyond access to lab equipment to include pedagogical materials and teaching experiences
- "iLab-ready" experimental equipment and software
- Sharing of time on national and international experimental equipment such as space-based experiments
- Improving education through expansion of lab-based learning opportunities around the world

Making iLab software available

- iLab software and documentation will be made publicly available
- "for comment" postings followed by formal releases
- Release under an open source license
- Welcome comments and advice from anyone interested

Collaborations

- We are actively working with universities in Singapore, Sweden, Taiwan, Lebanon, Uganda, Nigeria, Tanzania, Italy, Colombia, Greece on sharing iLab experiments
- Seeking to create a larger community of lab developers and users
- Design of batched experiment APIs reflect needs of low-bandwidth environments

Longer Term Plans

- Fall 2004 – complete specification for interactive experiments
- Spring 2005 – implementation and testing of APIs for interactive experiments
- Early Fall 2005 – release of interactive experiment specifications and code for comment
- Jan-Feb 2006 – full release of interactive experiment code, documentation, "how to" manual and other materials
- 2006-2007 – repeat cycle for sensor-based experiments

Contact Information

- Jesus del Alamo (alamo@mit.edu)
- V. Judson Harward (jud@mit.edu)
- Steven Lerman (lerman@mit.edu)