# Building an Ecology of Online Labs

*V. Judson Harward[1], Jesus A. del Alamo[1], Kayode Ayodele[2], Philip H. Bailey[1], Kimberly DeLong[1], James Hardison[1], Bryant Harrison[1], Olusoji Ilori[2], Adnaan Jiwaji[1], Steven R. Lerman[1], Phillip D. Long[3], Rahul Shroff[1], Hamidou Soumare[1]*

Massachusetts Institute of Technology[1], Obafemi Awolowo University[2], The University of Queensland[3]

**Key words:** *e-learning, Internet, laboratories*

## Abstract:

When the iLab Project at MIT began, the concept of sharing expensive lab equipment between universities was so natural and persuasive that we assumed lab providers and users would flock to join the effort once the enabling software infrastructure was developed. Now six years later, we realize that while such an infrastructure is necessary, it is not sufficient to foster the growth of a community of interest focused on sharing online labs. Such a self-perpetuating and evolving community requires at least three other foundations: (1) a deep understanding of the needs of different student audiences, (2) the identification of standard lab platforms that encourage collaboration as well as unique labs that extend the field, and (3) a mechanism for measuring and assigning value to the contributions of all participants in the online lab community.

## 1 Introduction

The MIT iLab project began over a decade ago with the overall goal of increasing the laboratory experiences available to university students worldwide [1, 2]. The project has been well published (most recently [3, 4], but most discussions have hitherto focused on the technical solutions that the iLab Project has developed to make online labs more scalable and more pedagogically effective.

As the project matures and grows, it has become apparent that the iLab Project and other notable online lab efforts in Europe and elsewhere face challenges that are more pedagogical, economic, political, or social than they are technical. This paper will explore some of these newly emerging issues, particularly those that arise from encouraging collaboration between institutions developing and using the same iLabs.

## 2 The Initial Goals of the iLab Project

We accept as a premise that laboratory experience plays a crucial role in a student's learning in many domains of science and engineering [5]. But as valuable as such experience may be, traditional laboratories present many difficulties. They are expensive to create and maintain, and they require much physical space that is hard to put to any additional use. The students' use of the labs is difficult to schedule and manage, and may raise safety and security issues.

The MIT iLab Project began with the realization that for many classes of experiments, providing access to real labs through a web-accessible software interface might prove a reasonable substitute for hands-on labs. In fact, we have found that these online labs that we call *iLabs* offer many advantages to the student and the teacher. The first and often most important advantage is that because the labs can usually be made available 24 hours a day, students can often spend more time exploring an iLab than the corresponding hands-on traditional lab. Second, the software interface through which the student operates the lab can combine multiple components including visualizations, numerical analysis tools, a companion simulation, intelligent online tutors, as well as collaboration and chat facilities that allow students to discuss any issues they are having with the lab. In a remarkable case study, faculty and students at the University of Queensland discovered that student success in performing a classic control experiment involving an inverted pendulum increased from 4% to 73% when the experiment was converted to an iLab [3, 6]. Part of the reason for this dramatic increase in student success lay in the increased amount of time students could spend with the experiment. But the iLab version of the experiment also supplied the student with a better set of tools to analyze the execution of the lab. The iLab version included a state diagram synched to a visualization of the motion of the pendulum (Figure 1). The playback of the visualization could be slowed down or paused allowing the student to examine the performance of his or her control script in detail. The interface also allowed the student to compare different runs of the experiment by overlaying their traces on the same graph.

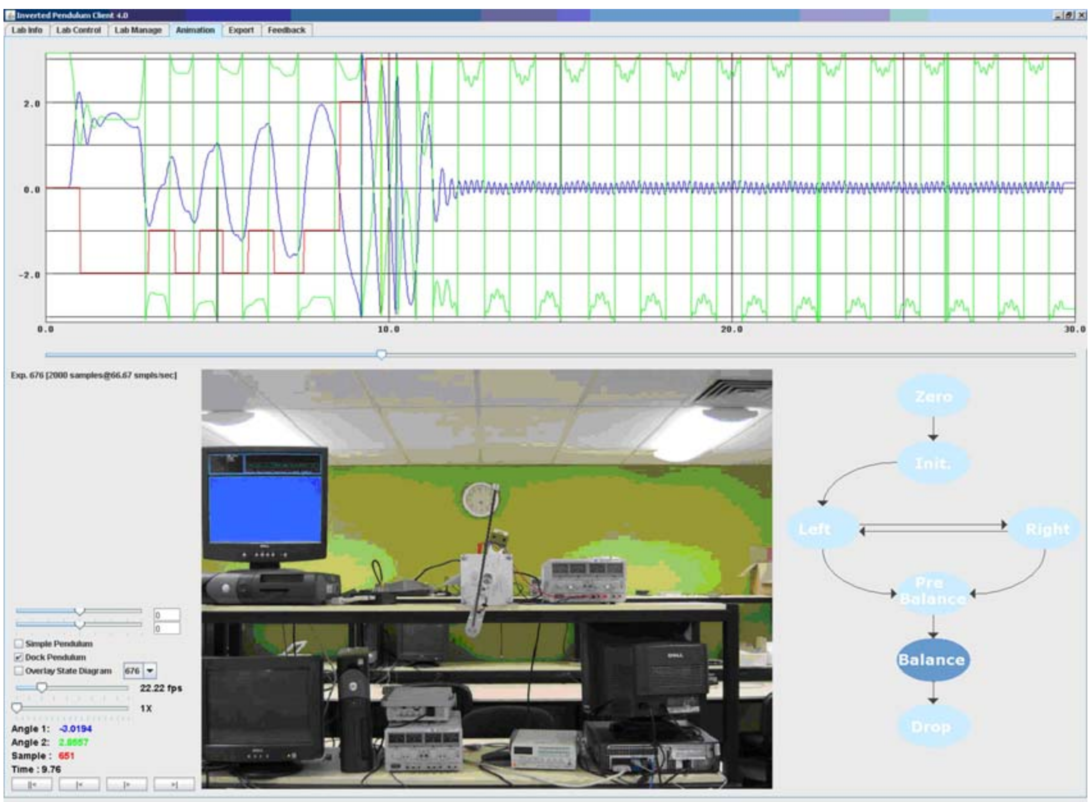


**Figure 1: The University of Queensland Inverted Pendulum iLab Client Interface.**

Because iLabs provide remote access and encourage the sharing of labs between institutions, they allow students access to equipment that is expensive, rare or located in unique or hard to reach locales such as Antarctica or the International Space Station. MIT has recently completed the implementation of an iLab that permits students to execute three classic experiments using the neutron beam from MIT’s nuclear reactor: a characterization of the thermal distribution of neutrons in the beam using a time of flight measurement, a neutron absorption experiment, and a neutron diffraction experiment using a copper crystal to select “monochromatic” neutrons. This new iLab will permit US high school students to perform at least two of the neutron beam experiments although they would never be permitted to physically enter the nuclear containment vessel for safety and security reasons.

# 3 The iLab Shared Architecture

Making a lab available to students over the web is straightforward provided that laboratory apparatus can interface to a computer server, often just a standard PC. One simply makes the server connected to the lab equipment, and thus known as the *lab server*, also act as a web server (Figure 2). Then the student need only login to the combined lab server and web server to gain access to the lab. The student controls the experiment through a piece of software known as the *lab client*. The client is delivered by the lab server and typically consists of a Java applet or active server page. Usually the same team develops both the lab server and client because they are so tightly coupled.

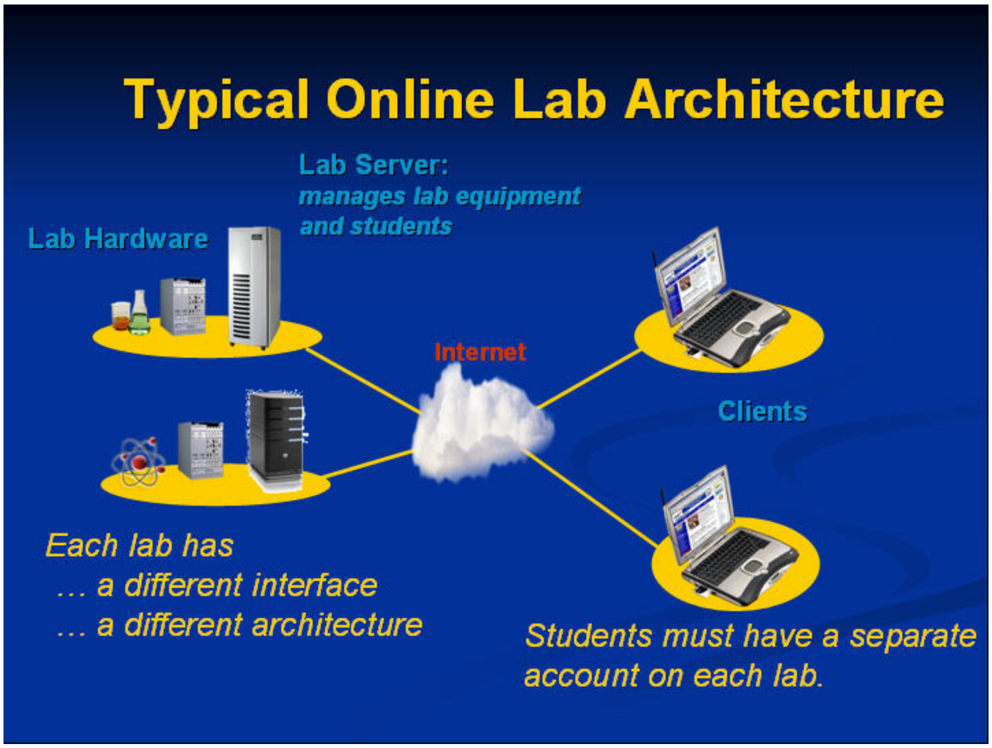


**Figure 2: A Simple Online Lab Strategy**

This simple approach has a number of drawbacks. First, while the developer may borrow an implementation from a previously implemented online lab and adapt it, first time developers usually start from scratch. The technologies that must be combined to succeed in this effort are diverse. The developer must be familiar with both the lab server and client environments. He or she must not only know how to program the interface to the laboratory devices but must also be familiar with standard issues of web programming including authentication and authorization. Most implementations will require local database support in order to save user data and history. The second drawback is more subtle. The lab developer has also become the lab administrator for the students using the experiment. If the lab is popular, the lab developer may be required to create and manage hundreds if not thousands of student accounts. The faculty teaching the students should be performing this role, but this simple implementation requires the staff member managing the students to have privileged access to the lab server. Teaching faculty, especially if they come from another institution, usually will not have this privileged access. Third, from the students' point of view, each lab has a separate interface and requires a separate account. They must use separate mechanisms to retrieve their data from each lab server. Thus this natural approach does not scale well for either the lab manager or the students who use the lab.

The iLab Project early on decided to tackle this problem by developing a basic software infrastructure to support the development of online labs. Lab developers create the lab server and usually also supply the lab clients used by students to access the lab. But the students' access to the lab is controlled through a totally generic server known as the *service broker* that is provided on an open source basis by the iLab Project. An iLab service broker normally runs on each campus from which students execute iLabs. Student accounts are managed by their own faculty. The service brokers provide the backbone of the system. If students need access to a new lab server located at another institution, the connection can be made through an automated protocol run between the student campus's service broker and the lab server's service broker. In effect, the iLab Shared Architecture creates standard roles for the different parts of the system. Lab servers, clients and service brokers can be administratively linked like snapping LEGO components together as in Figure 3 [7, 8].

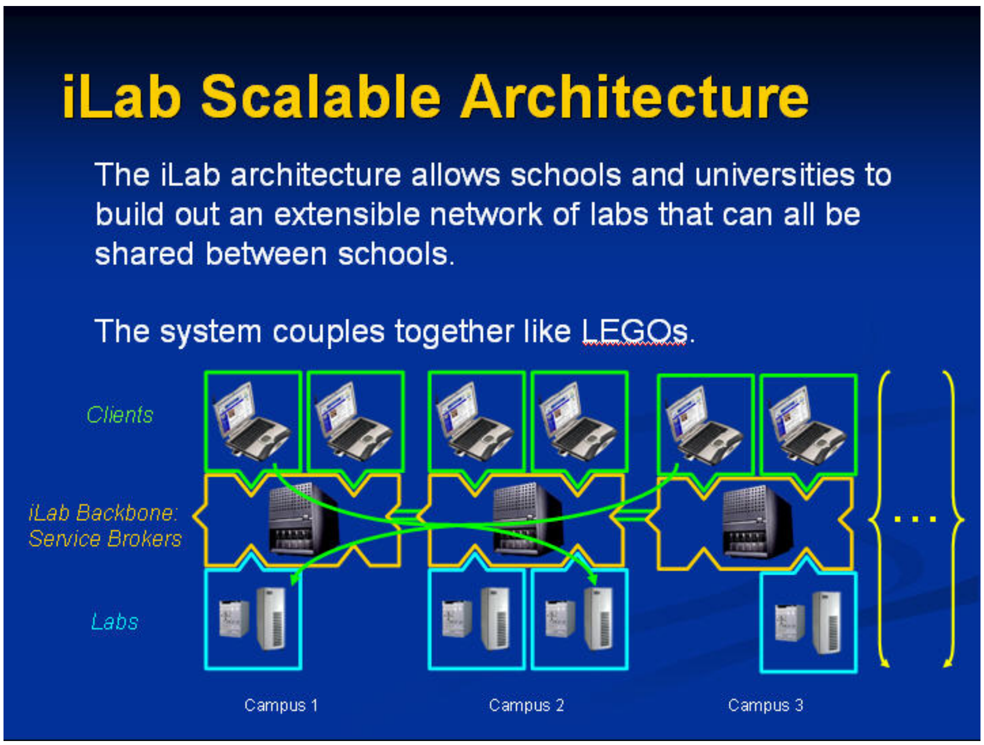


**Figure 3: The Scalability of the iLab Shared Architecture.**

## 4 New Challenges

For the past six years, the iLab Project has been occupied addressing issues of scalability and efficiency. We have also produced two versions of the architecture that we are now merging. The first supports "batched" experiments, those in which the student specifies all the parameters of the lab to be run before experiment execution begins [8]. The student does not interact with such labs while they are executing. He or she simply submits the experiment specification and then retrieves the results when they are complete. The second, more complicated version of the iLab architecture supports interactive experiments in which the student continuously monitors the course of the experiment and can modify parameters and take other actions while the experiment is progressing [4].

Now as more institutions are adopting the iLab approach, we are encountering an interesting set of issues that go beyond the technical and are encouraging us to think about new ways to structure the overall project.

### 4.1 Tailoring the Student Experience of a Lab

Up until recently we have taken it as a given that the same team if not the very same developers would be building the lab server and associated lab client(s). We have also come to realize that in many if not most cases, it is the design and construction of the backend lab and associated lab server that is the most expensive part of building a new iLab. We have developed two strategies to mitigate this cost.

The first recognizes that in certain fields like Electrical Engineering, one may be able to contruct a lab platform that is capable of performing a wide range of individual experiments. Consider a dynamic signal analyzer that allows a student to examine the frequency domain performance of a particular circuit. The creation of the dynamic signal analyzer platform may be initially expensive and time consuming, but a single platform should then be able to allow the examination of a wide range of circuits [9]. Students in different courses can use the same basic platform to examine different circuits of varying complexity, while students in the same course can examine different circuits over the course of the same semester. If the initial lab platform has been well designed, changing circuits can be as simple as swapping a circuit breadboard and entering a new XML-based circuit description on the lab server.

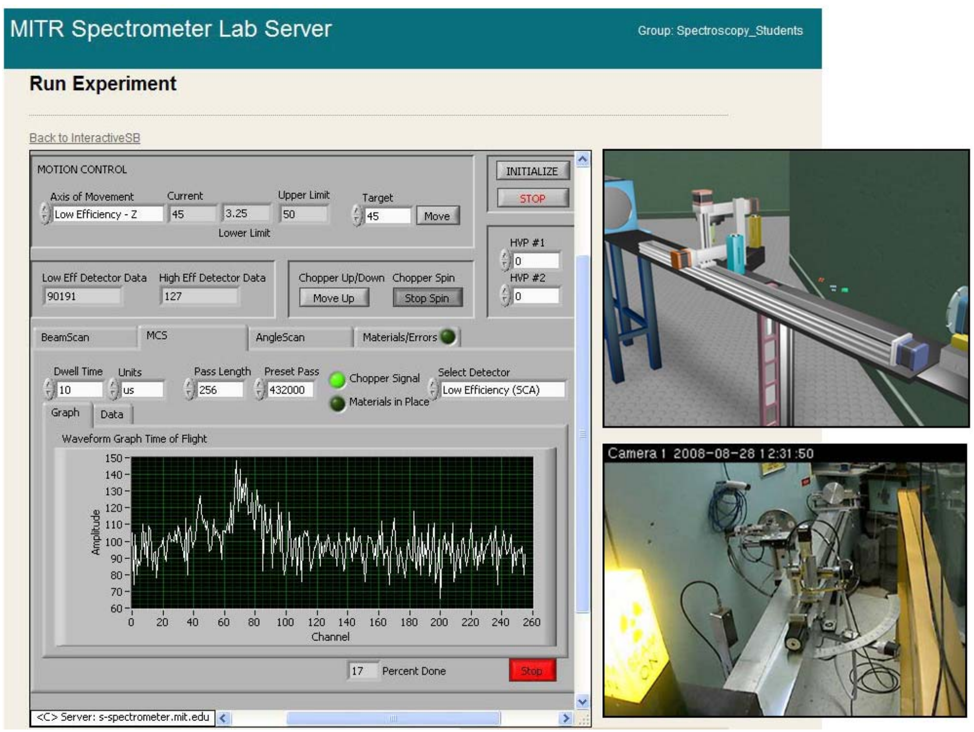


**Figure 4: The MIT Neutron Beam Lab**

The second strategy recognizes that certain sophisticated experiments can still be appreciated and understood at a simpler level by less advanced students. The MIT neutron beam experiments are a good example of this. The original remote lab implementation was designed for MIT students in the third year Physics or the Nuclear Engineering programs (Figure 4). Recently, a team of secondary school teachers collaborating with Northwestern University has selected the two simpler neutron beam experiments to be adapted to a US high school audience. This adaptation will require the redesign of the student client to provide a simpler, more focused graphic user interface. High school faculty will work with Northwestern University staff to generate a new description of the lab and instructions for its execution by the new high school audience. At least in the US, secondary school curricula tend to be tightly specified since secondary school students are so heavily tested, particularly if they are university-bound. High school faculty will help the Northwestern team integrate the revised lab closely with the standard high school advanced placement Physics curriculum. The high school student's experience of the lab will depend as much on these three revised components — user interface, experiment description, and revised curriculum — as on the original backend implementation, which will remain constant. Northwestern will be able to develop what is in many ways a new lab directed at a new audience, but at a fraction of the original cost because neither the lab server nor the hardware implementations will need to be changed.

### 4.2 Encouraging Collaboration Between Institutions

In the opening years of the iLab Project, we imagined that institutions would find the underlying goals and technology so natural that each institution would find it easy to create their own iLabs and contribute them to the growing community effort. In fact, we have found that any effort at standardization to encourage sharing comes at a cost. What may appear perfectly natural to one software engineer looks perverse to another. And as the iLab architecture has grown in functionality, it has also grown in complexity. Thus, as the community of iLab users has expanded, it has become increasingly important to foster collaboration between the development teams and faculty teaching with iLabs.

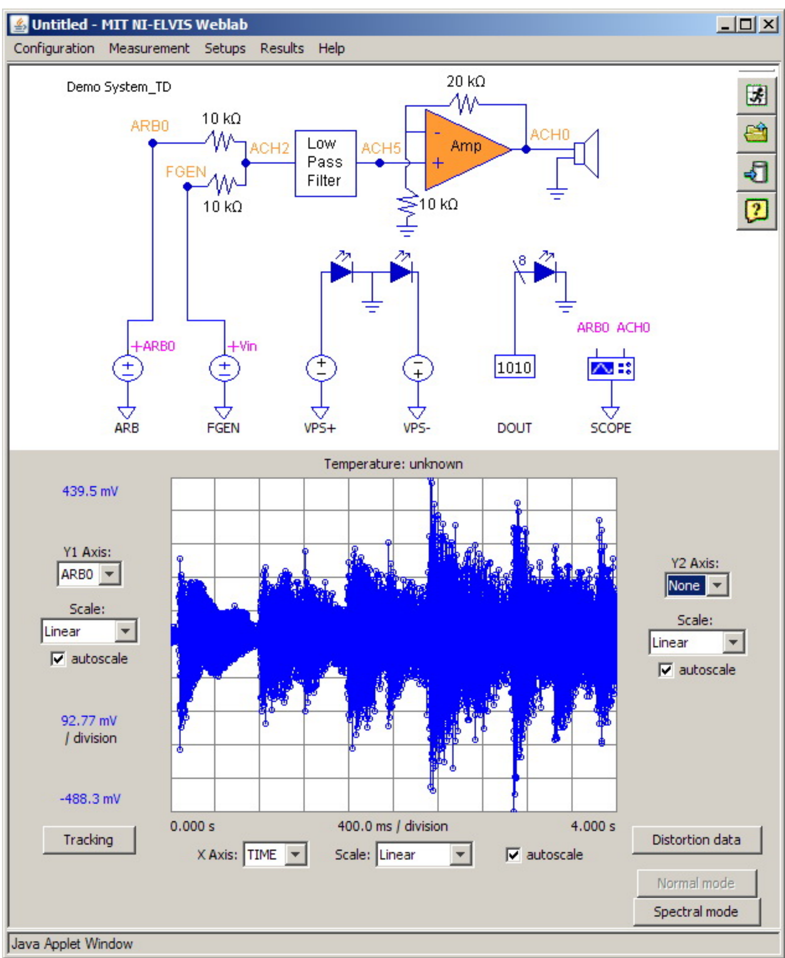


**Figure 5: The MIT-MATEC ESyst Lab with Dynamic Test Points.**

There have been splendid examples of such collaboration. MIT's concept of a general lab platform for Electrical Engineering (EE) experiments was adopted by our colleagues at Obafemi Awolowo University (OAU) in Ile Ife, Nigeria. Their work extended the platform by adding the functionality of switching matrices. This permitted students not only to analyze the behavior of individual circuits, but by using switches to incorporate circuit elements in parallel in the backend lab server and by using an innovative user interface to allow students to select components for inclusion in a circuit, the OAU team was able to introduce an element of design into such EE labs [10]. MIT then extended this approach to allow students to specify the test points at which they wanted to examine a circuit using an oscilloscope or digital multimeter instrument. This has led to the creation of labs that focus on diagnosis and troubleshooting from a high level, system-oriented perspective (Figure 5). The group of institutions designing and using EE iLabs that employ switching matrices has expanded to include the community colleges of the Maricopa Advanced Technology Center (MATEC), Makerere University in Uganda, and the University of Dar es Salaam in Tanzania.

But this collaboration has been enabled by the joint acceptance of a standard lab platform, in this case the low-cost ELVIS workbench manufactured by National Instruments for hardware, and the LabVIEW programming environment for software development on the lab server. While we hope that sharing expensive, unique labs will always play an important role in the iLab community, a simple low-cost platform is far more effective in encouraging cross-institution collaboration and joint development.

### *4.3 Creating an Experiment Economy*

Increasing the availability of iLabs depends in large part on finding new ways to finance the creation and maintenance of the iLabs. We are only at the very early stage in this process in part because the paradigm of sharing iLabs between institutions is so different from that of creating standard hands-on labs for a single institution. In the latter case, the actual "cost" of the lab is often never calculated. Department staff implement the lab and install it in department space. When a lab is used by participants from outside an institution, there is often an understandable pressure to calculate the cost of such use and to fund or recover it. But as we have noted multiple institutions may be involved in the creation of an iLab. The lab provider almost always creates the backend lab and the initial client software, but partners from other institutions may provide new client software, experiment descriptions, and curricular frameworks for the labs. Careful evaluation using questionnaires and observation of students can also be extremely useful in improving individual iLabs or the underlying infrastructure.

This leads to the fundamental question of how to encourage all these contributions and activities. Should all participants or institutions be compensated monetarily and should users contribute funds that are then redistributed? How can such an economy fairly value the very different types of contributions necessary to create a successful iLab? There are really two separate questions. There is the question of how to assign and balance value. Can value always be reduced to a monetary quantity or to a certain number of hours of effort? Each approach by itself seems inadequate. And neither takes into account the principle of demand. If one university creates a very expensive lab, largely for its own use, it should not expect other members of the iLab community who have no interest in the lab to help defray its cost. The second question is one of policy. Once a contribution has been assigned a value, the iLab community needs to decide how the contributor should be rewarded for providing that contribution in order to ensure that other members of the community will make similar contributions in the future.

The issues may appear abstract, but as work on online labs becomes more widespread it is also becoming more difficult to justify such efforts as pure research. Funders, whether they are public or private, are becomg increasingly concerned that the developed labs are sustainable, and that will require a far deeper understanding of the economic issues.

# 5 Conclusion

We feel the greatest challenges facing the further development of online labs lie less in technology choices than in an improved understanding of the political organization, the pedagogy, and the economy of online labs. There has been an organic element to the evolution of online labs at MIT and elsewhere that is important and healthy. Our recent experiences reinforce our recognition that the lab developer is only one of the significant roles necessary for the creation of a community of online labs. Some of these roles will be more academic and theoretical. We need a research economist to help us define and study the principles and metrics of the economy of online labs. We need far more work on the evaluation of labs and cognitive study of the human factors in good online lab design. And it is because of this realization that we feel the overall goal of the online lab community should be not just simply the creation of an *economy* of online labs, but more holistically the recognition and study of the emerging *ecology* of online labs with all the roles and relations that will foster and enable it.

## References:


[1] del Alamo, J. A., Chang, V., Brooks, L., McClean, C., Hardison, J., Mishuris, G., and Hui, L., “MIT Microelectronics WebLab,” in Lab on the Web, Tor. A. Fjeldly and Michael S. Shur, Hoboken, N. J.: IEEE Press and John Wiley & Sons, 2003, ch. 2, pp. 49-87.

[2] del Alamo, J. A., Brooks, L., McLean, C., Hardison, J., Mishuris, G., Chang, V., and Hui, L., “The MIT Microelectronics WebLab: a Web-Enabled Remote Laboratory for Microelectronics Device Characterization.” 2002 World Congress on Networked Learning in a Global Environment, Berlin (Germany), May 2002.

[3] Harward, V. J., del Alamo, J. A., Lerman, S. R., Bailey, P., Carpenter, J, et al., "The iLab Shared Architecture: A Web Services Infrastructure to Build Communities of Internet Accessible Laboratories," Proceedings of the IEEE, Vol 96, No. 6, June 2008, 931-950.

[4] Hardison, J. L., DeLong, K., Bailey, P., Harward, V. J., "Deploying Interactive Remove Labs Using the iLab Shared Architecture", to be presented at Frontiers in Education 2008, Saratoga Springs, New York, October 22-25, 2008.

[5] Ma, J. and Nickerson, Jeffrey V., “Hands-On, Simulated, and Remote Laboratories: A Comparative Literative Review,” ACM Computing Surveys, Vol. 38, No. 3, Article 7, September 2006.

[6] Hovland, G. "Evaluation of an Inverted Pendulum Control Experiment", IEEE Transactions on Education, Vol. 51, No. 1, pp.114-122, Feb. 2008.

[7] Hardison, J. L., Zych, D., del Alamo, J. A., Harward, V. J., Lerman, S. R., Wang, S. M., Yehia, K., and Varadharajan, C. “The Microelectronics WebLab 6.0 – An Implementation Using Web Services and the iLab Shared Architecture,” presented at the International Conference on Engineering Education and Research 2005, Tainan, Taiwan, March 1-5, 2005.

[8] Harward, V. J., del Alamo, J. A., Choudary, V. S., DeLong, K., Hardison, J. L., Lerman, S. R., Northridge, J., Varadharajan, C., Wang, S., Yehia, K., and Zych, D. “iLabs: A Scalable Architecture for Sharing Online Laboratories,” presented at the International Conference on Engineering Education 2004, Gainesville, Florida, October 16-21, 2004.

[9] Viedma, Gerardo, Dancy, Isaac J., and Lundberg, Kent H. "A Web-Based Linear-Systems iLab," presented at the American Control Conference 2005, Portland, Ore., June 8-10, 2005.

[10] Ayodele, Kayode P., Kehinde, Lawrence O., Jonah, Olutola, Ilori, Olusoji, Ajayi, E. O. B., and Osasona, Oladipo, “Development of an Operational Amplifier Virtual Laboratory Based on iLab Architecture and NI ELVIS,“ presented at the American Society of Engineering Education Annual Conference, 2008, Pittsburgh, Pennsylvania, June 22-25, 2008.

## Author(s):

V. Judson Harward
Massachusetts Institute of Technology, Center for Educational Computing Initiatives
Cambridge, Massachusetts, USA
jud@mit.edu

Jesus A. del Alamo
Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science
Cambridge, Massachusetts, USA
alamo@mit.edu

Kayode Ayodele
Obafemi Awolowo University, Department of Electrical and Electronic Engineering
Ile-Ife, Nigeria
kayodele@oauife.edu.ng

Philip H. Bailey
Massachusetts Institute of Technology, Center for Educational Computing Initiatives
Cambridge, Massachusetts, USA
pbailey@mit.edu

Kimberly DeLong
Massachusetts Institute of Technology, Center for Educational Computing Initiatives
Cambridge, Massachusetts, USA
kirky@mit.edu

James Hardison
Massachusetts Institute of Technology, Center for Educational Computing Initiatives
Cambridge, Massachusetts, USA
hardison@mit.edu

Bryant Harrison
Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science
Cambridge, Massachusetts, USA
bryant_h@mit.edu

Olusoji Ilori
Obafemi Awolowo University, Department of Electrical and Electronic Engineering
Ile-Ife, Nigeria
sojilori@oauife.edu.ng; sojilori@yahoo.com

Adnaan Jiwaji
Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science
Cambridge, Massachusetts, USA
adnaan@mit.edu

Steven R. Lerman, Vice-Chancellor and dean for Graduate Studies

Massachusetts Institute of Technology
Cambridge, Massachusetts, USA
lerman@mit.edu

Phillip D. Long
The University of Queensland, Center for Educational Innovation & Technology
St. Lucia, Brisbane, Queensland, Australia
longpd@uq.edu.au; longpd@mit.edu

Rahul Shroff
Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science
Cambridge, Massachusetts, USA
rshroff@mit.edu

Hamidou Soumare
Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science
Cambridge, Massachusetts, USA
hsoumare@mit.edu