Proceedings of the 2006 Winter Simulation Conference
L. F. Perrone, F. P. Wieland, J. Liu, B. G. Lawson, D. M. Nicol, and R. M. Fujimoto, eds.

EUCALYPTUS: INTELLIGENT INFRASTRUCTURE ENABLED PARTICIPATORY DESIGN STUDIO

Michael Jemtrud, Philam Nguyen

Carleton Immersive Media Studio
Carleton University
1125 Colonel By Drive
Ottawa, ON K1S5B6, CANADA

Bruce Spencer, Martin Brooks, Sandy Liu,
Yong Liang, Bo Xu, and Libo Zhang

Institute for Information Technology
National Research Council Canada
46 Dineen Drive
Fredericton, NB E3B9W4, CANADA

ABSTRACT

A new notion of participation is at stake with advances in technologically mediated work environments. Insufficient bandwidth and insufficiently powerful, crudely coordinated tools resulted in distributed task-based modes of collaboration that did not allow full participation by members of distributed design teams. The emergence of Service Oriented Architectures (SOA) and User-Controlled Light-Paths (UCLP) herald the beginning of a new age where fully participatory multi-site design may become possible.

While most service-oriented solutions are developed for gluing systems or data together, the Participatory Design Studio (PDS) Eucalyptus is developed to manage and configure the resources needed by users engaging in a participatory design session, such as a videoconference application, and a visualization server. Harnessing the power of UCLP, Eucalyptus strives to provide a set of upper layer services for non-technical users to provision devices and applications running on high-speed broadband networks, in addition to conventional TCP/IP networks.

1 INTRODUCTION

The current design context from an educational and professional perspective is highly digital-mediated and increasingly distributed. Collaboration between design studios, architectural and engineering firms, fabricators and consultants, etc. often traverses (and collapses) the space and time of the globe characterized as the so-called "24/7 office". Of course this context is by no means limited to architecture and is common in industries such as aerospace, automotive, medical, petroleum, entertainment, and defence. The project, discussed herein, intends to address this development and participate in the burgeoning research and development initiative around the next generation network and the instruments and middleware that constitute this networked infrastructure.

The "User Controlled Lightpath enabled Participatory Design Studio" (UCLP-PDS or Eucalyptus) is designed for users who are interested in collaborating with each other in participatory sessions, using high-end tools that run on broadband high speed networks, and other tools that enhance collaboration. A session consists of a set of resources including such collaborative tools and a set of users across the network. Typically a service-oriented solution is developed for integrating a system together by packaging application data through a standardized interface; this system does not deliver the collaborative application data such as video or audio streams. Instead, Eucalyptus is aimed at the meta-level; and it's role is to build a system that allows a user to *specify*, *invoke* and *manage* these sessions. The sessions themselves are not delivered through Web Services, but they are managed through Web Services.

Canarie's User-Controlled Lightpath Provisioning (UCLP) (Wu et al. 2003) is an SOA for provisioning lightpaths to users of CA*net4. A *lightpath* is an abstraction of connection between two or more switches in an optical network, and typically connects two points on the network at speeds up to 10 gigabits per second. The UCLP project inspired and supports this work. The UCLP provisioning Web Services allow users to dynamically assemble a set of lightpaths into a private end-to-end optical network, a so-called APN (Articulated Private Network). An APN network, formed by lightpaths, is one of the resources that is included in the collaborative sessions.

The theoretical and intentional underpinnings of the current research recognizes and attempts to identify characteristics of the biased nature of electronic modes of making and seeing but asks the question as to *what is possible only in the network driven digital realm* rather than lament on what is presumably lost from location-based collaboration. It provisionally accepts the seemingly enhanced features of a digital mediated environment such as a more thorough integrative mode, increased interactivity and responsive-

1-4244-0501-7/06/$20.00 ©2006 IEEE

ness, and greater immersion in the process (Al-Qawasmi 2000).

Eucalyptus strives to facilitate the digital mediated design process characterized by the manipulation, sharing, and visualizing of large and heterogeneous data sets. In this project, the users require many tools to be integrated, including video-conferencing devices and applications, 3D modeling and animation software such as AutoDesk's Maya, and visualization tools such as IBM's DCV - Deep Computing Visualization. The highspeed low-latency APN removes the bandwidth bottleneck. Eucalyptus provides a user-friendly Dashboard for architects to control these tools and instruments with the support of an SOA.

In the subsequent sections, the overall design, including services for managing sessions, users, workflows and resources, which are the major elements in Eucalyptus will be discussed. Also discussed are the implementation methods, followed by a discussion on issues and pedagogical and practical implications.

2 EUCALYPTUS (UCLP-PDS) SYSTEM DESIGN

UCLP promises to overcome several limitations in previous digital mediated distributed modes of collaboration that necessarily strive for low-bandwidth solutions due to limited bandwidth considerations. It utilizes existing fibre network infrastructure provisioned and controlled by UCLP software designed to enable end-users to create their own discipline or application-specific IP network whose topology and architecture is optimized for their particular applications needs and requirements. To put this in context, the UCLP controlled 10gb lambda network is referred to as the "third generation" network representing a significant 30-year transformation in how we use and conceptualize the Internet.

The PDS network consists of lightpaths spanning CA*net4 associated with one Gb/s channels that are bonded (i.e. grouped) to achieve channels with an effective bandwidth of 4-10 Gb/s. Through this high-speed network, the PDS gathers a variety resources, assets, and expertise in a manner that effectively collapses the space and time of the work environment and creates a "next door" phenomenon over large geographical distances.

Lightpaths offer high bandwidth and low latency; Eucalyptus utilizes this combination by deploying distributed tool configurations for which would be prohibited by a layer 3 gigabit network latency. One example is transmission of uncompressed high-definition video using Pleora Technologies' EtherCast™ devices which are ordinarily designed for deployment on a LAN. Another example is a distributed configuration of a PC cluster supporting IBM's Deep Visualization Computing where geometry and pixels are computed on clustered machines usually through an InifiBand interconnect that equals typical bus speeds; in PDS they will be connected by a lightpath. In addition there will be exchanges of data between the Render Farm, SAN, and a variety of communication platforms located at the various sites through application and desktop sharing.

All the core functions in Eucalyptus will be provided by Web Services, either as a single service or a combination of services. The services are divided into two groups: task oriented services and management services. Management services are generic and provide support and management for the task-oriented services. For instance, the Resource Management Web Service (WS) is responsible for managing all the resources that made available by Eucalyptus. It also serves as a registry for resources where the user can check the status of a particular resource. The *PDS Dashboard* is an integrated service client that provides a graphical interface for users to effectively use resources, and provision participatory design sessions. As one of the goals of Eucalyptus is to provide easy access remotely to many high-end resources that are typically not available in one single lab or facility, the PDS Dashboard hides the complexities of configuring the resources required in a participatory design session. There are three types of computers in Eucalyptus:

1. **PDSF** refers to **PDS Framework** computers. These computers have the Web Services platform installed and are generally used for exposing resources in Eucalyptus.

2. **PDSC** refers to **PDS Client** computers. These computers have the *PDS Dashboard* installed and are typically used by end users.

3. **PDSB** refers to **PDS Backend** computers or devices such as the Deep Computing Visualization Server (an IBM BladeServer) and the Rendering Farm, which is a network of high performance computers devoted to rendering.

3 SERVICE ORIENTED ARCHITECTURE (SOA)

Users configure the PDS network; i.e. it makes use of user-controlled lightpaths (Wu 2003). A SOA and graphical user interface referred to as the PDS Dashboard makes participation, control, and intelligence possible. The SOA and web services (WS) is the key development of the present research which renders the network transparent to the user and enables him/her to easily configure and access the resources available within the APN.

SOA is an application architecture that invokes interfaces to accomplish coordinated tasks in which the interconnected protocols and basic processes are established by the SOA. Web Services is a way of integrating web-based applications that allows the applications to automatically interface. The UCLP provisioning WS allow users to dynamically assemble a set of lightpaths into a private end-to-end optical network, or APN. SOA and WS are operating system independent. PDS Web Services are implemented in Business Process Execution Language (BPEL), authored with IBM WebSphere Integration Developer and executed with WebSphere Process Server (v6.0).

SOA middleware is the one of the most innovative components of PDS, providing a new state-of-the-art trade-off between system flexibility and ease of use. All the core functions of PDS are provided by WS, either as a single service or a combination of services.

The PDS SOA hide the tools' logistical complexities from users, allowing them to simply select the combinations most suited to the task at hand through the dashboard. PDS innovations include enabling commercial tools for long-distance use, and structuring each tool's computational components as WS resources so that that many different complex WS can be built from them, nevertheless providing the user with simple selections from preconfigured solutions.

SOA and WS allows for a heterogeneous composition of network-enabled resources that "uncouple applications and data from any specific machine or location" (St. Arnaud 2004). Thus, SOA allows the integration of applications and data to the network rather than being tightly bound to operating systems. The workstation becomes subservient to the network rather than the reverse situation typified with the current network configuration.

It also increases the level of control by the end users, since APN creation is no longer dictated by network administrators, but by the user (e.g. the design teams) possibly with the assistance of the technical staff on site. The high-speed low-latency APN removes the bandwidth bottleneck. Nevertheless, the design team requires many tools to be integrated, including video-conferencing devices and applications, 3D modeling/animation software, rendering management software, simulation applications and cluster resources, and visualization tools such as IBM's DCV. Inspired by CANARIE's vision (St. Arnaud 2003), PDS will provide a user-friendly dashboard for architects to control these tools and instruments with the support of an SOA.

3.1 Resource Management Web Service

One of the key motivation for developing Eucalyptus is to provide users with seamless access to resources located locally or remotely. We consider any non-human object required in a participatory design session as a resource, which could be anything ranging from a physical device (e.g. a visualization server) to a software application (e.g. Maya).

Thus, resource management plays an important role in Eucalyptus. Each resource is assigned a Resource ID. In addition, it is described by a resource name, description, physical location, owner, access restrictions, current status, resource type, whether it can accommodate multiple sessions, a list of parameters and allowable values respectively, and a calendar showing its usage schedule. We categorize resources into different resource types according their functionalities: communication tools, visualizing tools, management tools and others. For example, Isabel is a kind of communication tool. This categorization allows us to retrieve available resources by type and also facilitates our implementation which will be mentioned in a later section.

The Resource Management WS provides services to add new resources and modify the properties of existing resources. It also acts as a registry, where one can look up available resources by its properties, and get the current status of different resources.

3.2 User Management Web Services

There are three categories of users identified: the physical network administrator, the UCLP user, and the end-user. The end-users are students, architects and designers who are participating in the design. The physical network administrators are responsible for administrating the optical network and managing the lightpath resources. The UCLP user works with the end-users, and is capable of assembling the lightpath resources to create an APN for the end-users. A user's access restrictions are based on his/her role within the project.

The User Management WS provides services to add new users, modify and delete existing users. It also keeps track of the contact information, login status, user scheduling for all Eucalyptus users. Thus, it can also serve as a universal address book for Eucalyptus users. Typically each resource specifies the access restriction. When a resource is being required, the Resource Management WS will contact the User Management WS to verify if the user has the proper permission to use that resource.

3.3 Session Management Web Service

A session is a collection of people and resources, including at least one user and at least one resource. A session starts when a user engages to have a conversation or to use resources in Eucalyptus. A user can potentially be involved in multiple sessions. For example, a user can participate in a session with two other users who are text messaging, while s/he is video-conferencing through Isabel.

A session is proposed by one person through the Dashboard by selecting the resources and people involved, according to what kinds of interactions should occur among these people.

To reserve a session, a thread-safe check is made that each included resource is available to all included people. Then the session can be reserved. In more detail, a resource is available to a person if at least the following conditions and perhaps others are met:

1. The resource can be provided to that person via their PDSC computer, or via another computer in the same room.
2. The resource is permitted to be used by this person using some access control scheme.
3. The resource is not otherwise allocated to that person through some other session, unless that resource can be part of two different sessions. Text messaging systems can be part of multiple sessions, as one can text messaging in more than one conversation at the same time, but two different Isabel video-conference sessions cannot be hosted by one computer.

The setting of a session can be saved for future use. Suppose an architecture team located in Ottawa needs to have a meeting with another team located in Montreal for a particular project on a regular basis and they meet through videoconference and also use a shared desktop application to discuss a building design. The setup for the first meeting can be saved and given a name. From then on, the user can launch this saved session without re-defining the setup for every meeting. In addition a user can also define a session via pre-defined workflows, which are managed by Workflow Management Web Service as defined in the following sub-session.

3.4 Workflow Management Web Service

A workflow clearly defines the sequence of activities that must be performed in order to accomplish a certain task, and for each activity, it defines the preconditions required. In the context of Eucalyptus, activities are performed by Web Services. The Web Service workflow can be defined by a workflow language such as WS-BPEL, previously called BPEL4WS (T.A. 2003). Users can orchestrate a set of Web Services together to perform certain tasks. For example, a user can specify a workflow for rendering a file as follows:

1. Select a set of files for rendering and transfer the files to the file server through the File Management Web Service if they are already there;
2. Start a rendering task managed by the Rendering Web Service;
3. When the rendering task is completed, display the output to the user via the Visualization Web Service.

4 PDS DASHBOARD

The PDS dashboard is implemented as a web application. A user can access it from any workstation connected to the Internet (or the APN). The functions of the PDS dashboard are supported by a set of underlying services. The services have been divided into two groups: task-oriented services and support/utility services. Support/utility services are generic and support the task-oriented services. For instance, the User Management Service, which is a utility service, authenticates each user; users with the proper security certificates then are granted access to task-oriented services, such as accessing files through the File Management Service.

The dashboard, which can be project or group specific, is a flexible, customizable GUI that allows each user to create the context in which he/she is working. It functions by adding intelligence to the mediated environment and remove actions such as configuration, establishing protocols, and the logical launching of applications in a coordinated manner. Once logged in, the user sees the resources, assets, and people that comprise his/her work environment.

The dashboard monitors resources and gives permissions if those resources, such as the rendering farm or work file, are available. Communication options from text messaging to High-Definition video conferencing and display are made invoked through the GUI as long as the request does not violate rules of availability. It can be temporally customized to allow more direct access to resources that are conditioned by the process such as training material at the beginning or texture folders during the compositing process. Applications are launched within their proprietary interface although the WS manages some configuration and preferences.

In the Eucalyptus project, it was decided to develop the integrated client as a desktop application as opposed to a web application for several reasons. 1) Web applications have limitations on client functionality. There is no easy way to access the local file systems. 2) The implementations of the HTML, CSS, DOM and some other tools are browser specific and they often act inconsistently in different browsers. 3) It is less convenient to maintain an accurate reflection of status of all the resources.

To maintain a desktop application over many computers is normally not an easy task. However, with the help of Java Web Start, the deployment and maintenance of
Java desktop applications become easier. The advantages of
Java Web Start include automatic application update, desktop integration, platform independence, Java runtime environment management, and security.

The Dashboard interface is carefully designed to be unobtrusive and user-friendly. Inspired by DragThing http://dragthing.com, it is implemented as a floating dock, similar to the Mac OS X system dock. It contains four tabs:

People, Resource, Session, and Workflow. If the user selects the People Tab, for example, s/he can view the set of all people that are registered in the Eucalyptus system and distinguish who is currently logged in.

To compose a session using the People Tab, the user selects from among the people currently logged in, by selecting a set of people to be included. Then, the user is shown a list of the resources that are available to all selected users. The composing user can then select from among these resources to complete the definition of the session. The Eucalyptus system then attempts to book this session.

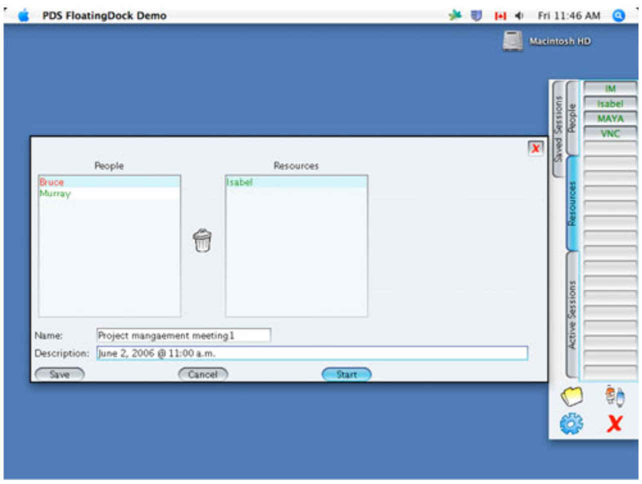

Figure 1: PDS Dashboard working prototype

5 RESOURCE WRAPPING: A GENERIC APPROACH

To make existing resources available on the Web, one way is to rewrite the existing applications and develop Web Service versions. However, rewriting all the code is an unacceptably inefficient approach. Therefore a simple alternative is presented. Our approach allows applications to be configured at start up according to a set of parameters. For example, in a multipoint videoconference, each PDSF computer included in the session needs to be configured to connect to the conference host. This approach is also useful for wrapping commercial applications for which no source code or API (Application Program Interface) is available. An example of wrapping the Isabel resource as service is illustrated in the following. The resources in each category have similar functions, which means every resource in the same category has similar operations to be exposed to the user. For example, in communication tools, all resources should have following functions: `startResource()`, `stopResource()`, `getStatus()`. Different resources have different input and output parameters in its functions. To make the generic WS interface extensible to all of the more specific resource type, we declare all the input parameters and the return type as `String`. A Web Service Description Language (WSDL) (Christensen et al 2001) file contains the operations of the WS and the arguments to invoke these operations. When resource functions are wrapped as WS, the interface of the resource WS is described in a WSDL file.

Since many commercial applications do not provide open APIs, we need to figure out, first of all, how to interface with these applications through the command line interface, which includes the execution commands and all of the input and output parameters. We then decide which functions of the application should be made available in the related Task-oriented WS. Subsequently, we develop a Java program as a wrapper to make calls to those functions. We use the Java runtime class to interface with the application resource, such as Isabel video-conferencing system, that is running outside of the Java Virtual Machine (JVM). Once the wrapper program is created and can make calls to the native application to perform different functions
locally, we expose the wrapper as a WS.

Now have the WS co-located with the supporting resource, *e.g. Isabel*, in the same PDSF computer. However, this WS can be invoked only by the Resource Management WS. If someone requests an Isabel session, the request will be first handled by the Session Management WS. The Session Management WS will contact the Resource Management WS and the User Management WS prior to the invocation of the Isabel Video-Conference WS to ensure that only authenticated and authorized user have access to it.

6 EUCALYPTUS DEPLOYED: FIRST STEPS

6.1 CIMS-LaJolla: iGrid 2005

Initial configuration and testing occurred in September 2005 as a demonstration for the iGrid2005 conference where the Jonas Salk Institute for Biological Studies, built by architect Louis Kahn in 1965 in La Jolla, California was digitally reconstructed over a 4-day period. It consisted of a small team "on site" with minimal resources (two laptops, 8mg pixel camera, video camera) connected to a central lab with significant compute and personnel resources. The UCLP configured lightpath connected CIMSlab-Ottawa to the conference site.

Graduate students from the Carleton University School of Architecture were deployed at both sites; they collaborated on construction of a 3D model of the Salk Institute according to a pre-establish 3D imaging and modeling protocol (Jemtrud 2005). Prior to the demo, the Ottawa students used architectural reference materials to construct a basic 3D model of the Salk Institute buildings. The students in San Diego visited the Salk Institute to further develop the model and animation through photogrammetric modeling, to gather high-resolution textures and site information, verify accuracies, and give concrete feedback to the entire team.

At iGrid, the two groups of students worked together in real-time to model in Maya and ShapeCapture, to render scenes and composite the animation. "Being there" was critical in order to experientially characterize the site for effects, ambience, and in storyboarding the HD animation. Distance collaboration was facilitated by high definition videoconference and shared access to applications running from the host lab through DCV, desktop sharing, and VNC. The demonstration was an important milestone in the evolution of the project, which has evolved from the relatively simple demonstration to an operational distributed environment.

The lessons learned from this initial proof-of-concept were many. The major realization concerned the "problem of the speed of light". Lightpaths' high bandwidth results in a large bandwidth-delay product (St. Arnaud 2004). Since many of the PDS tools will use TCP/IP to transport large datasets, optimization of the relevant TCP/IP implementation parameters will be essential.

The positive value of the PDS scenario and workflow was equally evident. The minimal on-site resources proved sufficient because access to intensive compute resources in Ottawa was immediate through the lightpath. The team in La Jolla was able to set up the animation, use the rendering farm, and transfer large data sets as if within the internal network. The video conferencing systems and desktop sharing made effective communication and participation by all the students. If additional information was needed during the construction it was simply a short drive away. As a result, a sophisticated, accurate, and high-resolution artifact was created efficiently with more effective results.

Upon completion of its first stage in December 2006, PDS will be made available to the university and research community for practical use and further evolution.

6.2 CIMS-Montréal: Boulevard St. Laurent

Presently, two studio sites (CIMSlab-Ottawa and CIMSlab-Montréal) and the CRC BADLAB are connected by way of lightpaths in the current PDS. NRC-Ottawa and NRC-Fredericton are connected to the PDS through a standard CA*net4 connection. The host site, CIMSlab-Ottawa, contains the main infrastructure including: concentration of designers and applied researchers; a 14-blade cluster (rendering farm); high-performance visualization cluster (IBM Deep Computing Visualization); application server and a central archives/storage; reference material, peripherals, as well as standard tools such as drawings and digital whiteboards. A range of video conferencing systems from individual to H323 and high definition systems are available between all sites. A visualization cluster drives immersive environments located at the remote site from the host site over the high-speed network. CIMSlab-Montréal is located at the Society for Arts and Technology (SAT) and contains the same video conferencing systems, two workstations, and three immersive environment configurations.

Presently, work is being done between the two sites in constructing the urban model in much the same way the Salk Institute was created according to the 3D imaging and modeling protocol. Laser scanning will be done in the late spring that will challenge the collaborative work over the network due to the large file sizes and complexity of manipulation. The fusion of the various data sets will occur through the participation of people at differing sites with varying skill sets and expertise. Development of the SOA, web services, and dashboard will continue and progressively include more sites and clients through 2007.

7 ISSUES AND DISCUSSIONS

The design and development of Eucalyptus are challenging. Beyond simply wrapping existing equipment and applications, our design accommodates changing demands and includes innovations for issues that are not normally encountered in SOA implementation projects. In this section, a few issues that have been encountered and some distinct features of this project are discussed.

Most Web Services are set up to bridge communication between different applications. Consider a Web Service that accepts purchase orders from different clients, where the message content would primarily consist of the data required to complete the business transaction. The message may be large and it is desirable to minimize the number of nested service calls, since each requires the data to be wrapped and unwrapped. In the case of Eucalyptus, the Web Services are used mostly for provisioning and monitoring of the tools, so the message contents are relatively lean, consisting mostly of control parameters. Even though we have some nesting of calls through the various management services, a provisioning service should be able to call a number of different associated services with relatively little delay.

The services provision different tools, and for each tool there must exist a service platform to provide access to it. Using Web Services, the straightforward way is to have both an HTTP server and an application server running on the machine that co-hosts the application controlling the resource. This is a labor-intensive approach as it requires installation of the required servers, and the required set of related libraries. It also requires configuration of the runtime environment on every machine that interfaces with one or more resources. As the number of Web Services increases, the maintenance work is at risk of becoming unmanageable. There are also security risks. As the number of entrances into the network increases, more security holes are opened for malicious access (Birman 2006). To counter these risks, for each network domain we set up an HTTP server outside the firewall, and use it to funnel the application server traffic behind the firewall by a private channel (e.g. SSH). The application server then connects to the local resources.

To better manage the communications overhead, the Resource Management WS keeps track of the states of different resources. It is more efficient to communicate this information from this single site to the various PDSC computers, rather than having each PDSC computer regularly communicate to all the PDSF computers to keep this status information current. This arrangement also simplifies our workflow.

Most resources are not designed to work as a service, and many runtime parameters are set by applications that run these resources. This can make accessing the resource complicated because the appropriate access method depends on the current state of the resource. One way to get around this is to initialize the resource to a ready state before performing any action. If this is possible, there is no need to keep track of the state, achieving the stateless principle (Erl 2006).

To reduce the cost of maintaining the client desktop application, it is desirable to design a generic GUI panel for all resources. But since various resources have different input parameters, it seems to be required to generate a separate GUI panel for each new resource. One possible solution is to design an XML schema that defines the layout of the panel in XML syntax. An XML file compliant with this schema describes the panel for an individual resource. Then the XML file can be parsed and rendered at the client side (Yan et al 2005). Eucalyptus applies SOA in a novel usage area. It also provides a testbed for future research in process management and service computing.

8 PEDAGOGICAL AND PRACTICAL IMPLICATIONS

The pedagogical and professional implications of this new vision of network-based work environment cannot be understated. By seeing work environments as truly network-enabled, the present research questions collaborative and participatory work in general. Assuming such network capability is a thing of the near future, the value of the research is in a new conception of working. The way in which this environment and its tools are configured and constructed impacts the very nature of how we see, think, and make the world *together*.

The implications of the UCLP-enabled PDS for distributed design scenarios are varied and significant. No longer is network-based collaboration subservient to local workstations, application, and operating systems that ultimately require low bandwidth solutions for network-based collaboration. Designers can use the tools they would typically use in a proximate situation over a distributed network.

Development of the SOA and web services will allow easy configuration and access to pooled resources at multiple sites creating a global, design-specific APN ("next door" phenomenon). Sophisticated and phenomenologically rich communication scenarios, access to wide ranging expertise and deployment infrastructure including large scale augmented environments establishes a creative environment unavailable to location dependent studios. The question is not "what is lost" from distance but what is only possible in such a situation where, through technology, time and space are collapsed in the service of creative and productive goals.

The possibility of sharing compute resources at a truly effective level (i.e., not limited to low-bandwidth solutions) will allow organizations (universities, institutions, offices) to have access to infrastructure that is cost prohibitive from an acquisition and support perspective. Those institutions, countries, and companies who place a priority on the network rather than the "black hole" of hardware can share access to network-capable resources, typically unavailable at any single institution.

As seen in both examples, "on-site" work requires minimal resources when being there is critical to the successful completion of the project. The compute load is placed on the network thus relieving the burden of locale based and consolidated infrastructure.

Real-time interactivity and a more immersed and experientially rich design process are possible. Collaborative tools beyond the typical screen must be developed to respond to this larger environment. With "intelligent" SOA and web services designers can customize teams and work environments making present technical hurdles transparent to the user thus allowing the team to more freely participate in creative activity. Dashboard and customized GUI will allow for a spatialization of a global work environment that is more "immersive" and multi-modal including sound, haptics, fabrication, etc. The expansion of presentation and deployment of robust assets is a larger discursive sphere for a greater amount of stakeholders from professionals to the general public.

The ability to freely involve experts located at geographically remote locations for design development and review provides exciting possibilities for education and professional activity in training and design. Greater expertise can be culled between stakeholders who can interact with experts distributed across the globe. In fact, this is already the case to some extent when a large architectural firm has offices in LA, Beijing and engineers in New York and London, fabricators in Spain, urban planners in Tokyo. At present, such collaboration is typically limited to video conferencing and transfer of files at best. It is based on a conventional understanding of communication and task-based collaboration.

What is being suggested here is that not only can one achieve a more robust technologically mediated and experiential work environment, but that given this emerging paradigm of work, we must also re-think the fundamental nature of participation.

REFERENCES

Al-Qawasami, J 2000. Learning virtually: a paradigm shift in design education. In *Proceedings of the Fifth Conference of the Association for Computer Aided Architectural Design Research in Asia,* 123-133, Singapore.

Birman K. 2006. The untrustworthy web services revolution. In *IEEE Computer*, 39(2):98–100.

Christensen, E, F. Curbera, G. Meredith, and S. Weerawarana 2001. Web services description language (WSDL) 1.1. Available online via <http://www.w3.org/TR/wsdl>.

Erl, T 2006. *Service-Oriented Architecture: Concepts, Technology, and Design*. Prentice Hall.

Jemtrud, M. 2005. Technological mediation and digital craft: Diverse 3D data integration in architectural and urban reconstruction. In *Practices: A Journal of the Center for the Study of Practice*, 9/10.

St.Arnaud, B. 2004. *CA*net4 research program update – UCLP roadmap: Web Services workflow for connecting research instruments and sensors to networks*. <http://www.canarie.ca>

T. A. et al 2003. *Business Process Execution Language for Web Services version 1.1.*

Wu, J., S. Campbell, J.M. Savoie, H. Zhang, G. V. Bochmann, and B. St. Arnaud 2003. User-managed end-to-end lightpath provisioning over CA*net 4. In *Proceedings of the National Fiber Optic Engineers Conference (NFOEC)*, 275-282, Orlando, FL, USA.

Yan, Y, L. Yong, and, D. Xinge 2005. Controlling remote instruments using web services for online experiment systems. In *IEEE International Conference onWeb Services (ICWS'05), Orlando, FL, USA*, pp, 725–732.

AUTHOR BIOGRAPHIES

MICHAEL JEMTRUD is the Director of the Carleton Immersive Media Studio (CIMS) and an Assistant Professor for the School of Architecture at Carleton University (Ottawa, Canada). Professor Jemtrud holds a B.A. in Philosophy (Philosophy of Science and Technology), a Professional B.Arch from Pennsylvania State University, and an M.Arch in the History and Theory of Architecture from McGill University (Montreal, Canada). His current research interests focus on the digital tools, techniques, and artefacts used to see, think, and make the world - primarily in the area of high performance visualization, genetic representation, (Mediating) devices of design and e-centric Collaborative Work Environments.

BRUCE SPENCER is the founder and group leader of the Internet Logic Group at the Institute for Information Technology, National Research Council of Canada, which is Canada's national laboratory. The Internet Logic group specializes in e-Business and in particular in reasoning systems for Internet applications, which includes work in electronic marketplaces, automated policy evaluation and validation, monitoring and diagnosing e-business systems, and the development of systems for expressing and implementing rules governing Internet usage. He has a BSc (Dalhousie, 1980), a MMath (Waterloo 1983), and a Ph.D.(Waterloo, 1991) and is the current president of the Canadian Society for the Computational Study of Intelligence.