# The MeetingMachine:  Interactive Workspace Support for Nomadic Users

John J. Barton, Tony Hsieh[1], Vikram Vijayaraghavan[1], Tomoto Shimizu[2], Brad Johanson[1], Armando Fox[1]
Mobile and Media Systems Laboratory
HP Laboratories Palo Alto
HPL-2003-128
June 23rd , 2003*

ubiquitous,
appliance
computing,
smart spaces
interactive
rooms,
conference
rooms

Ubiquitous computing embraces both nomadic computing and infrastructure-rich interactive workspaces.  Although most effort in these areas to date has focused narrowly on one domain, there are interesting challenges at their intersection. To start a discussion on how some of the benefits of interactive workspaces can be provided to nomadic users, we present MeetingMachine, a digital media projector that incorporates both interactive workspace and nomadic computing technologies in an appliance form factor. As a research effort, the MeetingMachine attempts to bridge the gap between space-embedded and mobile computing models of ubiquitous computing, specifically focusing on the task of sharing information brought into and created during meetings. As a prototype artifact, the design of the MeetingMachine seeks to place ubiquitous computing technology into an appliance, whose role would be similar to that of a digital projector or conference room telephone.

# The MeetingMachine: Interactive Workspace Support for Nomadic Users

John J. Barton§, Tony Hsieh*, Vikram Vijayaraghavan*, Tomoto Shimizu†, Brad Johanson*, and Armando Fox*

§Hewlett-Packard Laboratories, Palo Alto, CA
*Stanford University, Stanford, CA
†Stanford University Visiting Scholar on leave from Hitachi Ltd.

### Abstract

*Ubiquitous computing embraces both nomadic computing and infrastructure-rich interactive workspaces. Although most effort in these areas to date has focused narrowly on one domain, there are interesting challenges at their intersection. To start a discussion on how some of the benefits of interactive workspaces can be provided to nomadic users, we present MeetingMachine, a digital media projector that incorporates both interactive workspace and nomadic computing technologies in an appliance form factor. As a research effort, the MeetingMachine attempts to bridge the gap between space-embedded and mobile computing models of ubiquitous computing, specifically focusing on the task of sharing information brought into and created during meetings. As a prototype artifact, the design of the MeetingMachine seeks to place ubiquitous computing technology into an appliance, whose role would be similar to that of a digital projector or conference room telephone.*

## 1. Introduction

### 1.1. Motivation

Ubiquitous computing encompasses several overlapping visions of the future beyond personal computing. The field's mobile computing heritage is evident in areas such as nomadic user support and the extension of ubiquitous/environmental computing to handheld devices (see [18] and references therein). Other distinct work focuses on embedding computing technology into buildings [2], classrooms [3], or dedicated meeting rooms [14, 28]. We use the term *interactive spaces* to refer to this last category (or *interactive workspaces* to refer to Stanford specific research), usually characterized as dedicated, room-sized spaces with a rich embedded infrastructure including wireless networking, applications, and middleware.

We observe that the task of *supporting nomads at meetings* is a common real-life usage scenario that inherits from the domains of both nomadic computing and interactive spaces. On the one hand, by nature no infrastructure can be assumed to be present in the environment for nomadic computing; instead we would like to be able to deploy it on-demand. Nomadic computing also implies the need for seamless integration of both the media and the (heterogeneous) mobile devices of the nomadic users with minimal or no preparation on their part (such as pre-installing software). On the other hand, the

usage scenarios that have driven interactive space research, while requiring a fixed infrastructure, have frequently been focused on meeting support and other forms of group collaboration. As a result, interactive space research has produced a variety of tools, technologies, and middleware architectures well suited to that task domain. We would like to harness elements of both research areas in order to provide meaningful support for nomads in meetings.

The MeetingMachine (MM) is an attempt to do this by combining elements of the Stanford iROS (Interactive Room Operating System) software [27] with HP Labs' Web-based system for Nomadic Computing [18]. iROS provides coordination of multiple existing applications on multiple machines controlled by multiple users. The Web-based Nomadic Computing system supports physical-virtual connection and spontaneous interactions. We place the iROS coordination system in a single appliance-like device that connects to an ordinary electronic projector. The appliance runs some frequently-used software to support meeting activities, supports the temporary or permanent sharing of electronic media brought in by nomads, and aids aggregation the nomads' machines to allow cooperative, meeting-oriented tasks such as shared browsing and media exchange.

### 1.2. Contributions

***Contribution 1*** is an exploration of how to combine the elements of interactive spaces, which are infrastructure-rich and deal with a relatively static user population, with the constraints of nomadic meetings, which may be infrastructure-poor and consist of dynamic groups of users. The goals are (a) to bring at least two of the most-used aspects of interactive spaces—public browsing and a deliberately temporary shared media abstraction—to nomadic meetings; and (b) to reduce the gap between the physical meeting and the (usually electronic) meeting products by enabling connection between physical objects and virtual objects.

***Contribution 2*** is a specific architecture and prototype for an *appliance* that meets the above goals. As outlined in our earlier work on Appliance Computing [12], we chose to develop an appliance because it lets users deal with concrete media artifacts, such as USB and CompactFlash memory cards, and even non-computer physical tokens, rather than forcing them to deal directly with awkward computer-centric abstractions such as shared file systems. Our portable appliance plugs into any electronic projector and transforms a "generic" meeting space used by nomads into one that supports the above usage modalities. The same
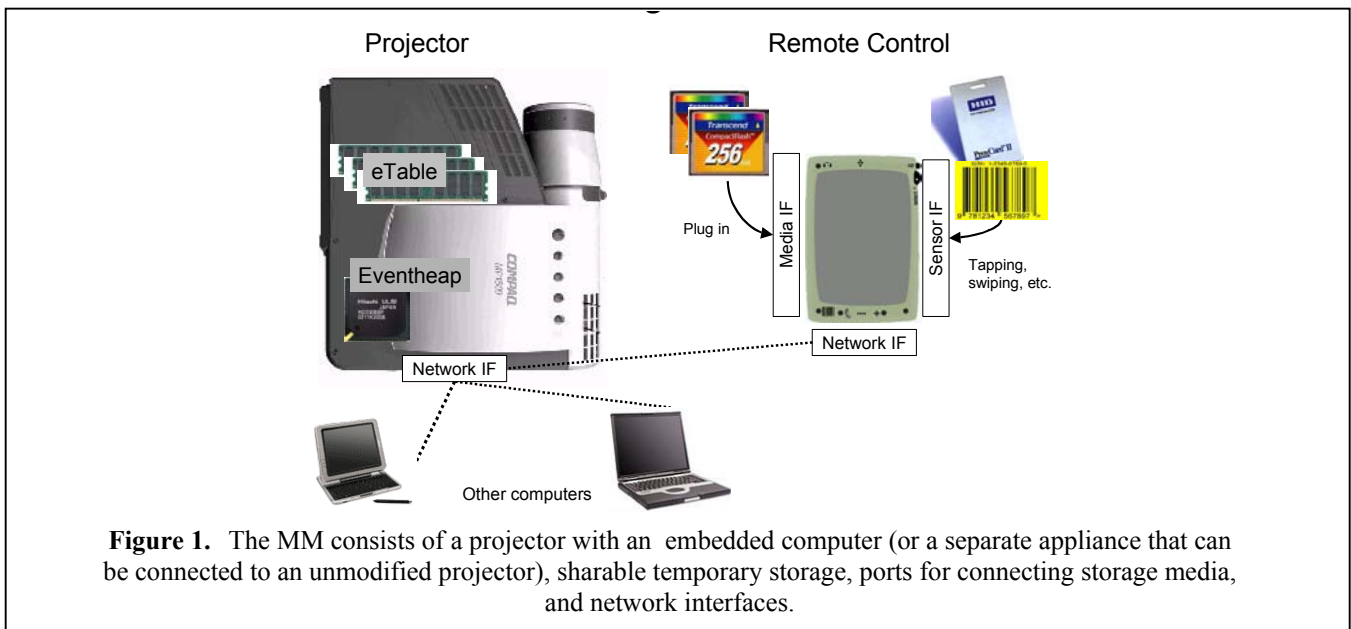
**Figure 1.** The MM consists of a projector with an embedded computer (or a separate appliance that can be connected to an unmodified projector), sharable temporary storage, ports for connecting storage media, and network interfaces.

elements could be merged into a portable projector for a completely nomadic device or they could be embedded in a conference room for a interactive workspace-like use model.

The rest of the paper proceeds as follows. We begin by discussing what happens when nomadic users have meetings, and what we can expect in the way of existing technology, to derive design constraints. We then describe an architecture that addresses these constraints in an appliance form factor, combining techniques from iROS and CoolTown to provide four successive layers of functionality. We describe our early prototype using the UbiWISE [6] prototyping environment, our physical prototype, and ongoing work.

## 2. Supporting Nomads in Meetings

Our goal for the MeetingMachine is to increase the productivity of meeting participants primarily by shortening the delay between the time of the physical meeting and the completion of the ultimate products of the meeting, typically the electronic documents that result from human collaborations (see as motivation [23, 24]).

To start we need an assessment of the state of meeting rooms. Our model for the people, where they meet, and what electronic media they use is based on the author's own experiences and personal observations with meetings, primarily at Stanford and Hewlett-Packard, and is not based on a structured study of meetings. Nonetheless, we believe these observations are broadly applicable in comparable organizations (see also [7]). We describe our informal model of these elements so that our user scenario and architecture can be better understood.

### 2.1. What happens in a meeting?

Typically our meetings involve 2-20 people, often co-

workers with whom we share a great deal of electronic infrastructure. Even then, some participants arrive with wireless laptops, some with paper, and some only with a cell phone. Meetings may directly or indirectly involve shared or to-be-shared electronic documents, though the electronic resource may or may not be used in the meeting. When meetings involve people from outside our group, a typical purpose is to exchange information through talking and showing presentation slides (e.g. PowerPoint) using a projector connected to a laptop computer. If the electronic version of the slides is passed around, it will usually be either to circumvent unusual circumstances—for example, when a visitor's laptop fails to connect to the local projector—or by a post-hoc link or attachment sent via email by the presenter or their host

Our meeting rooms have many sizes and shapes. Most have projection screens, many have built-in projectors, and almost all include a table or podium that could accommodate a machine the size of a large phone. Some rooms have computers connected to our protected intranets (and/or wireless/wired connectivity to it) and some have multiple projectors or displays. Our meetings occasionally involve video conferencing to remote rooms.

When we meet, we have an immediate goal of oral communications augmented with gestures and diagrams. But this immediate goal typically contributes to a larger goal that includes the production of electronic documents, such as publications, slides, schedules, calendars, agendas, computer programs, and so on. Some of the documents, like HTML web pages, are by their nature readable by almost everyone involved; others, like specialized 3D models, are readable by only a few people, because of the requirements of specialized software, limited access to the document's storage medium, or both. The access and storage model for these documents is not uniform within or without our organizations, and we believe the same is true for

comparable organizations.

We can summarize these observations as follows:

**1.** Meeting rooms are not uniformly equipped, and "generic" spaces are often turned into *ad hoc* meeting spaces on short notice.

**2.** Documents may be temporarily shared during a meeting, so we should make this easy to do. However, it is important to separate this *temporary* sharing from the ability to explicitly make a permanent record.

**3.** When shared documents are under discussion, it should be possible for all participants, not just the document owner, to manipulate and interact with the document and easily call attention to related documents.

**4.** Some users do not habitually bring laptops with them, relying instead on references to remote content, such as Web-posted documents referenced by the URL. We should provide a way for such users to incorporate documents into the discussion "by reference" rather than requiring them to bring equipment or electronic storage media with them.

## 2.2. Supporting Nomads in Meetings: Four Layers

Based on the above observations, we provide four conceptual layers to fill the gap between the physical meeting proceedings and the virtual meeting-products: a new projector-like digital appliance to provide a physical focus of attention for user interaction, mechanisms for electronic media interchange, tools for collaborative work on electronic media, and integration of physical and electronic media. We outline these layers next as a way of introducing the MeetingMachine functions.

### 2.2.1. Layer 1: A Unifying Appliance

Observation (1), the non-uniform equipping of meeting rooms, reinforces our starting belief that while a complete, multiscreen interactive space is attractive for meetings and can support sophisticated applications [14], the design time overhead and physical space remodeling for these custom installations requires a substantial investment (the Stanford iRoom cost in excess of US$100,000 for hardware alone). We chose instead to design the MM as a telephone-sized appliance. Although its implementation can be realized using a reconfigured PC, an appliance's specialization narrows the usage model to meeting-room-specific functionality, allowing the device to "disappear" into the meeting proceedings rather than merely providing another distraction for users [12] (most current interactive spaces, by comparison, require dedicated administrators). The appliance's functionality can focus on meeting-specific tasks with a simple user interface that avoids the complexities of generic PCs such as a keyboard, mouse, and sophisticated shared file system. Also, the appliance can host non-user-specific- but-infrastructure-critical applications so that individual users need not concern themselves with them (e.g. the infrastructure to support a temporary shared document space).

In our prototype the appliance does not have an integral projector since many meeting rooms already have built-in projectors with VGA drop cables. It would be straightforward, however, to integrate the functionality we describe into the same housing as a projector to create a standalone "smart projector".
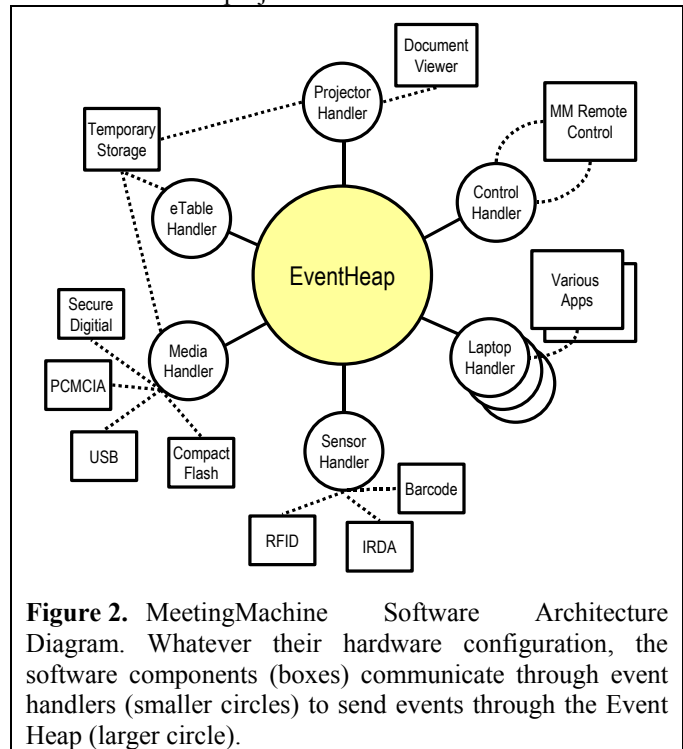


**Figure 2.** MeetingMachine Software Architecture Diagram. Whatever their hardware configuration, the software components (boxes) communicate through event handlers (smaller circles) to send events through the Event Heap (larger circle).

The base "appliance" consists of two parts. The *Projector Base* is a box about the size of a large phone that includes network interfaces (Ethernet and WiFi) for connection to the local intranet, and VGA output ports to display to one or more projectors. A practical implementation might provide an integral WiFi access-point and NAT functionality as well. Throughout this paper we call the image projected from the base the *main display*.
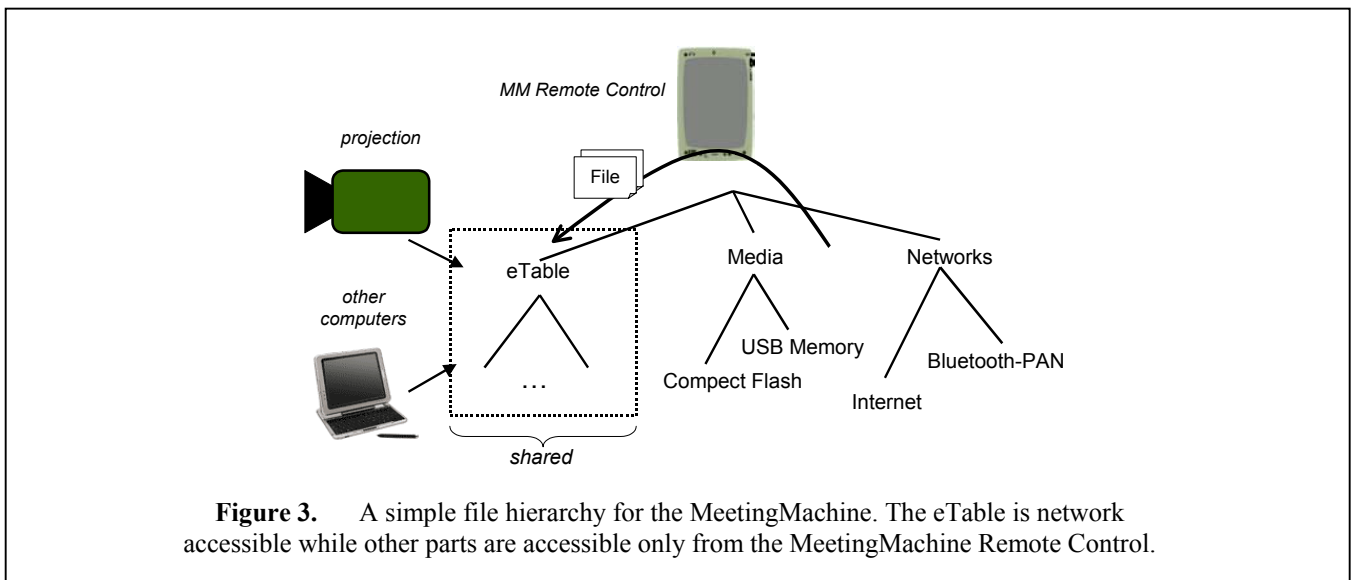
The second part is a handheld *Remote Control* unit that resembles an enhanced Personal Digital Assistant. (We will refer to this part of the appliances as the MM Remote Control below). It includes:

 (a) interfaces for portable storage media, including USB memory devices and CompactFlash CF memory cards;

 (b) WiFi connectivity for communicating with the MM base;

 (c) IRDA (infrared) I/O and RFID sensors.

We explain the usage models for each class of ports in the sequel.

### 2.2.2. Layer 2: temporary electronic media interchange.

The need for document sharing expressed in observation (2) leads us to create the electronic equivalent of the conference room table, which we call the *eTable*. It provides a public, sharable temporary storage and work space. Like the conference room table, the eTable is

**Figure 3.** A simple file hierarchy for the MeetingMachine. The eTable is network accessible while other parts are accessible only from the MeetingMachine Remote Control.

intentionally temporary: the MM Remote Control includes a Clear button that erases this storage. Just as people remove physical documents from the shared meeting table when a meeting ends, we expect people to clear electronic media from the eTable when their meeting ends.

*2.2.3. Layer 3: support for public browsing of shared documents.*

Observation (3), the need for all users to interact with content being discussed, leads us to add interactive workspace technology to interact with and display content to our media-exchange enhanced projector. While we ultimately hope to support a rich set of meeting applications, we start with a more modest goal of supporting cooperative browsing of electronic documents. Once a document is explicitly placed on the eTable, this allows three main capabilities: (a) in the case of slide presentations (Web-based or PowerPoint), any user can direct her laptop to "slave" to the main display (discussed as an example in [27]); (b) any user can direct the main display to show any shared document (multibrowsing [16]); (c) any user can manipulate the mouse pointer associated with the main display, via remote-control from her own laptop or handheld (PointRight [15]).

*2.2.4. Layer 4: linking physical and virtual.*

In our final layer we support access to virtual resources (e.g. documents, links) through resolution of physical identifiers that can be scanned by any of the sensor technologies integrated in the MM's Remote Control unit (currently RFID, barcode reader, and IRDA). The main purpose of this layer is to provide for physical tokens "bound" to documents. Users could, for example, associate a barcode with a documents URL while in their office and then just bring the token to the meeting, or bind a document created in the meeting to a "blank" RFID card to carry it out of the meeting. These techniques allow nomadic users to move digital content into or out of the room using a physical object [17, 22]. Users may choose whether to

make these documents available to other meeting participants using the media interchange layer.

The physical and virtual link layer is also inherent in the design of the MM: the physical MM appliance represents the underlying "virtual" software infrastructure it provides. As we will describe later in more detail, we use this mapping in a concrete way by allowing a user's mobile devices to discover the infrastructure by receiving an IRDA beacon with connection information from the physical MM device.

# 3. Architecture and Implementation

Having presented an overview of the functional layers of the MeetingMachine and a use model example, we now turn to the system architecture. Readers familiar with iROS and CoolTown technologies may skip the next section and proceed to section 3.2.

## 3.1. Background: iROS and CoolTown

iROS, the Interactive Room Operating System, provides a simple coordination-based programming model for interactive workspaces that is optimized for ease of integration and evolution [27]. In this model, applications are constructed by coordinating the behavior of a loosely-coupled ensemble of independent building blocks. Coordination occurs via an extended-tuplespace model called the Event Heap [13], a globally-visible "blackboard" over which messages are exchanged using a publish-subscribe scheme. The use of soft state and automatic message expiration make managing the Event Heap easy: in general, any process or application communicating with the Event Heap can be restarted without warning at any time, as can the Event Heap itself, without jeopardizing correctness and in most cases with minimal performance implications. The Event Heap's narrow API's are provided as libraries for a variety of languages that support TCP/IP sockets, and as a servlet that converts HTTP form submissions to event
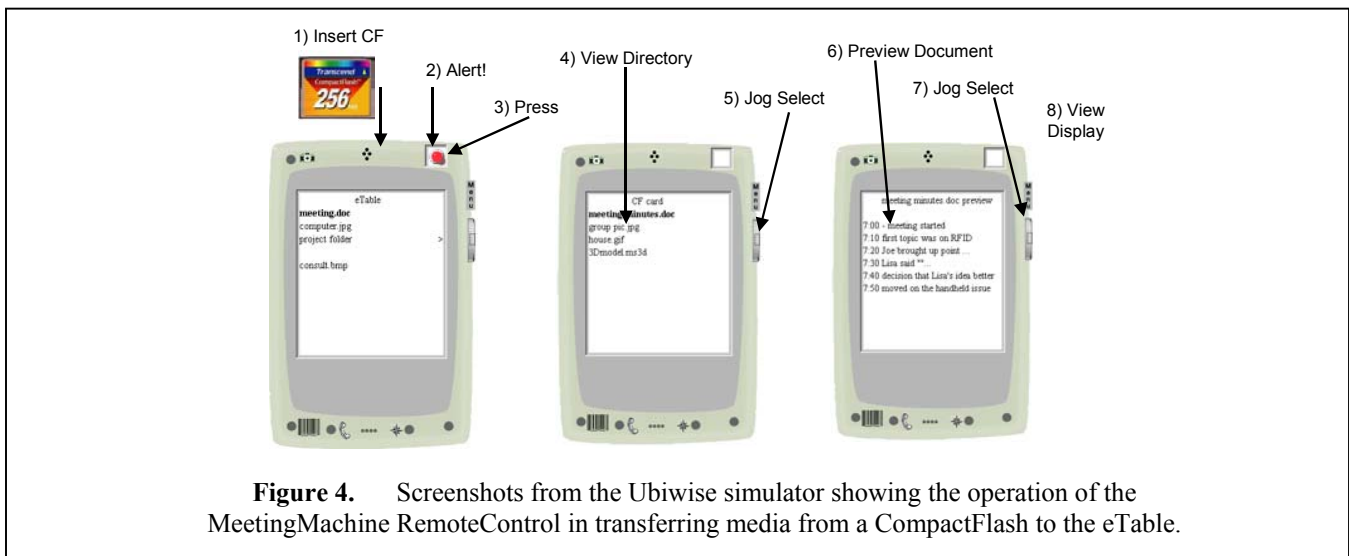
**Figure 4.** Screenshots from the Ubiwise simulator showing the operation of the MeetingMachine RemoteControl in transferring media from a CompactFlash to the eTable.

publications or subscription requests, thereby allowing any device that can run a Web browser to interact in a limited way with the Event Heap.

The Event Heap and several Event Heap-based applications are deployed in various interactive spaces including the Stanford iRoom, the KTH (Kista campus) iLounge, the HP Labs uRoom, and others. Two key applications we leverage for the present work are Multibrowsing and PointRight. Multibrowsing [16] aggregates all displays in an interactive space, whether on individual users' devices or large shared displays, and allows any user to "push" a document to any display or "pull" to her own a device a document being shown on any display. This is accomplished via the Event Heap, the Event Heap servlet, and a simple browser plug-in. PointRight [15] allows any pointer in the interactive space, whether a shared pointing device that is part of the infrastructure or the pointing devices on individual laptops, to range across all displays, thereby giving the illusion of a single pointer that any user can control at any time. (We rely on social etiquette for floor control.) When a user moves the mouse pointer off her own laptop display, it reappears on one of the public shared displays, in our case the projector connected to the MM. The geometry of how multiple shared displays are connected is configurable. Finally, SmartPresenter (described in [27]) allows multiple public displays to be aggregated for presenting PowerPoint slides and other content, allows remote control of the presentation from any handheld or laptop, and enables audience members to set their laptop to "slave" to the master presentation, automatically advancing the slide on each user's laptop display whenever the presenter advances the slide on the main shared display.

The nomadic computing technology we embed in the MeetingMachine grew out of the Cooltown project [18] at HP Labs. That work mapped physical tags and identifiers to web URLs, either directly or indirectly through a lookup service [20]. When read by a handheld device with a wireless LAN connection, the tag or identifier acts like a physical analog for a web page hyperlink. Like web pages, the meaning of physical hyperlinks can vary, but the use most pertinent to this work has the physical hyperlink refer to a web page describing a conference rooms services and environment or to a web-page of resources connected to a user [8, 9].

Therefore the design challenge we tackle is to embed the important ubiquitous computing hardware and software ideas from iROS and CoolTown into an appliance that is portable and fits into a typical meeting room, requires near-zero administration or configuration (like the projector to which it connects), and works well with the technology that people already use in meetings.

### 3.2. Basic Software Architecture

The MM runs an instance of the Event Heap, as shown in 0. All coordination among elements of the MM architecture occurs over TCP/IP connections to the Event Heap server. The Event Heap-centric design follows the Boundary Principle [21]: a device can participate in an MM-facilitated meeting if and only if it can communicate with that MM's Event Heap. Nomads who have installed the open-source iROS client software on their machines will automatically discover the Event Heap and establish an association with it, after which their machines will be incorporated as a part of iROS applications (e.g. automatically added to the PointRight configuration information). Nomads who have not previously installed the iROS client may install it by accessing a downloadable version stored on the web server associated with the MM.

### 3.3. eTable architecture

Functionally, the eTable is stored as a subdirectory in the files system of the MeetingMachine (0) and is also shared with other users in the space. Users view the eTable as one sub-tree in the interface provided on the Remote Control—mobile media and network content appear as

other sub-trees in the UI. To place media on the eTable users first insert the physical media into the Remote Control, using USB, CompactFlash, or Secure Digital Media. They then use the controls on the Remote Control device to navigate, preview, and select the documents they wish to transfer as shown in Fig. 4. We are still experimenting with the user interface for the remote control, but our current model uses a jog-dial with up, down and select to move through files in a directory or descend to a sub-tree, and a menu button to move up to the parent of the current sub-tree. Using the "select" operation on a document shows a preview, and a second "select" on the preview moves that content to the eTable and displays it on the main display.

Meeting participants using laptops or PDAs connected by networks can also access and add content to the eTable, but may not access the other content visible through the UI on the Remote Control. Keeping that content hidden from other meeting participants allows a user to privately browse content on the Remote Control before choosing to move some specific document onto the eTable. In addition to preserving the user's privacy, this also prevents him from distracting other meeting participants during the selection process.

## 3.4. Collaborative browsing architecture

Collaborative browsing is enabled by the global visibility of Event Heap events and existing iROS tools. Using SmartPresenter users may choose to slave their local laptop displays to the main display, allowing them to follow along with a presentation. Clients can also use Multibrowsing [16] to move content to the main display or other user's machines, or pull content from the same. Finally, using PointRight [15], user's can take turns controlling the pointer and keyboard on the main display. Thus, for example, a few people with laptops could discuss several electronic documents, taking turns controlling the pointer on the projected display or putting new documents up for discussion, with each user only having to interact directly with her own PC.

## 3.5. Physical/virtual interconnection architecture

We use physical identifiers in two specific ways: to allow the devices of nomadic users to locate the infrastructure provided by the MM and to move data on and off of the eTable using URLs bound to physical objects.

In the first case, we have an IRDA beacon in the MM continuously emitting a URL that points to a web server running in the MM projector. Laptops with IRDA can pick up the beacon and access a control web page on the MM over the wireless LAN. From this page users can configure the MM, establish a connection to the MM's Event Heap, gain access to the eTable, etc. Since IR is localized in space and the web server is in the room, the result is a place-embedded Web page, a "web-presence" for the room [8]. Since this page is generated dynamically within the MM, it can be used to "bootstrap" an arriving nomadic user

into the meeting, and even allow first time users to download the IROS client software if they so desire.

Second, RFID and bar-code sensors on the MeetingMachine Remote Control allow users to bring electronic documents in to a meeting using small physical objects like key chains, watches, or jewelry or using barcodes on paper. The sensors yield identifiers that are converted to URLs [17, 18] pointing to the documents as illustrated in Fig. 5. After conversion the documents are loaded into the MM Remote Control for preview and possible selection with the same user interface as the physical media. This allows papers, badges, maps, floor plans, musical scores, schedules, calendars and so on to be tagged with IDs matching virtual resources related to that physical artifact. These "by-reference" URL links to online media complement the "by-value" copying of media and provide an alternative path for input and output to the eTable .
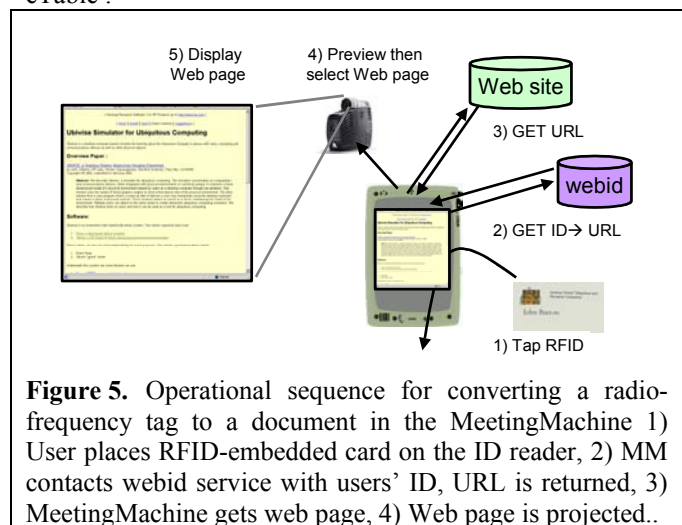


**Figure 5.** Operational sequence for converting a radio-frequency tag to a document in the MeetingMachine 1) User places RFID-embedded card on the ID reader, 2) MM contacts webid service with users' ID, URL is returned, 3) MeetingMachine gets web page, 4) Web page is projected..

Effective use of this feature requires more infrastructure in the environment of the MM. To show online media, the MM would need connection to the intranet/Internet. The users of MeetingMachine equipped environments would also need to add a small compatible physical tag-reader to their desktop machine. In this way they could mark documents at their desk and carry the token to a meeting for quick recall and discussion.

## 3.6. User Scenario and Supporting Features

### 3.6.1. User Scenario

Now that we have described the four layers of functionality and the architecture behind them, we try to convey a sense of how the appliance might be used by a group of experienced users. Our scenario concerns four people working on a 3D "walk-through" of a proposed office-technology upgrade. The storyboard for this scenario is shown in Fig. 6. One person is an outside consultant bringing "textures" (images) of possible office fabrics and finishes for the walkthrough. The other three people work together routinely. To make the story easier to understand,

we assume that these three share access from their desktop personal computers to a file system that is mapped on to a web server. Several such mechanisms for shared access exist, including WebDAV [11], and we hope to more thoroughly investigate them as we continue with this work.
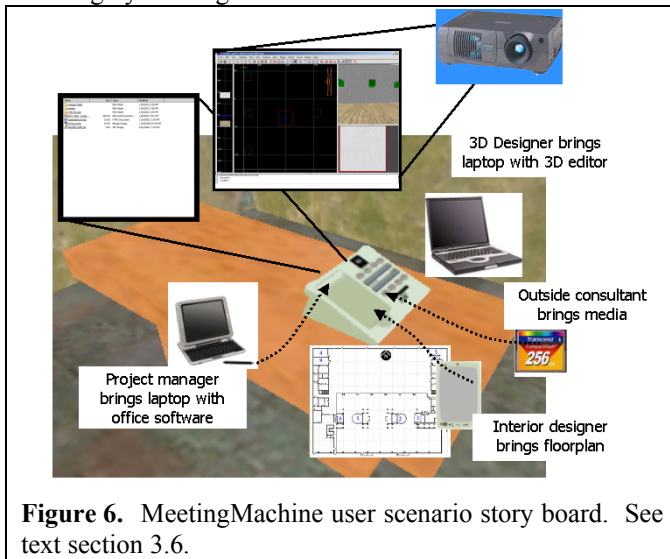


**Figure 6.** MeetingMachine user scenario story board. See text section 3.6.

**1.** Today's meeting is run by Patty, the project manager. She welcomes Andy, an outside consultant working on texture images, and Dan, the interior designer, to the meeting room where she connects a MeetingMachine (MM) to the room's projector and powers both up.

**2.** Dan spreads a (paper) floor plan on the table. The floor plan is marked with bar codes for each room or corridor area. Dan uses the MM Remote Control's bar-code sensor attachment to scan the bar code for a particular room. Since Dan associated these bar codes with web pages [17] which had been set up for each room, a web page with a set of links corresponding to various electronic media related to this room appears on the handheld display. Dan previews one of the links to verify that it is his (electronic) sketch of the room interior, then he selects it so that it is displayed on the projector and moved onto the eTable. Everyone agrees that this is the correct room for today's work.

**3.** Now Dan passes the MM Remote Control to Andy, the texture consultant, who puts a USB memory device into it and uses its controls to view thumbnails of the texture files on his memory device. He selects one using the jog-dial on the handheld. The file is transferred to the eTable and shown on the projector. The meeting participants discuss this texture and decide it seems suitable for the room decoration.

Until now we have shown how the MM works in a meeting even with no laptops. Next we extend the previous scenario by adding laptops to show more capability of the MM.

**4.** During the first part of the meeting, Patty's laptop established network connections to the MM. The laptop discovered the MM's Web server via an IR beacon and displayed a web page for the room which included a button

that allows devices to connect to the MM Event Heap. As Patty examines her project management documents, a note reminds her that a previous design choice for the room may conflict with the new texture. She goes back to the MM web page window and clicks to connect to the Event Heap. Then she uses her laptop's Multibrowse function (a right-click menu option) to push the conflicting texture onto the main display for others to view. Now the team is uncertain about the room decoration.

**5.** A fourth person, Victor, the 3D designer, arrives to show 3D virtual image of the office to help them visualize the conflict. Victor plugs his laptop's VGA port into a second (unmodified) projector in the room and connects to the Event Heap in the same way as Patty did. To save time Victor asks Dan to bring up the team's page for the room on the eTable. Dan clicks the floorplan as before (see item 2.), and the links page for the room is once again displayed (but is not moved to the eTable again since it is already present there). Both Patty's and Victor's laptops display an alert that the main display has changed. Victor pulls the web page to his laptop using multibrowsing and clicks on a link to a 3D model file he had previously linked there. His 3D CAD application automatically launches and displays the model of the room, which everyone else can also see on the second projector.

**6.** Victor, the 3D designer, next looks at the eTable directory listing (available whenever a laptop is connected to the MM). He clicks on the texture file Andy left there, loads the result in to his editor, and places the texture on appropriate parts of the room's visualization.

**7.** This process – transferring a texture and trying it on 3D model – is repeated until the team is satisfied with the design. During the review the other team members use their own pointer controls to reposition the view being displayed by Victor's laptop and bring up and interact with content on the main display using PointRight through their connection to the MM.

**8.** The team finally agrees on one texture for the virtual room under discussion. Patty copies the selected texture data from the eTable of the MM into her machine for later transfer to the team's disk storage. The meeting adjourns and Patty presses "clear" button on the MM Remote Control, erasing the temporary files on the eTable.

### 3.6.2. Supporting Features

The meeting room has two built-in projectors so it has some basic digital interactive capability. When Patty starts the meeting she is bringing interactive workspace technology into the room inside the MeetingMachine (step 1); she is nomadic but she is creating an interactive space. Andy, the outside consultant is completely nomadic; the others use a mixture of nomadic and interactive space technologies.

When Dan, the interior designer, selects IDs in the form of bar-codes bound to physical paper with the MM's portable sensor attachment (step 2), we have used the basic projector control with electronic media interchange (section

3.3. ) initiated from a physical-virtual connection (section 3.5. ). The web browser in the MM resolved the ID within the company's global webid server [17] and loaded the returned URL to the MM Remote Control. Neither the URL nor the page content is "part of the room". It is visible only to the user of the Remote Control. The final "select" operation transfers the URL to the MM eTable and selects it for presentation. At this point the information is "public" and visible both through projection and through the temporary file system of the MeetingMachine.

Andy's transfer of the texture file (step 3) uses the same steps for physical media rather than the physical-virtual interconnection feature.

To connect laptops to the MM in the room (step 4), physical-virtual connection technology is used to assist the user in aggregating her device with the room appliance. The MM runs a web server [9] and beacons URL out over its infrared port, allowing Patty and Victor to use physical discovery [5, 19] to connect to the MM Web server. The infrared beacon transfers a URL to their laptops; the resulting web page represents the room and features a link to trigger connection to the Event Heap. Patty's connection to the MM and her Multibrowse client allows her to send links to the eTable for shared viewing via the projector.

Step 5 shows the shared nature of the workspace; Dan's operation affects Patty and Victor's synchronization to the meeting. When users of the MM Remote Control display a file on the main display or transfer a file onto the eTable, both the MM projector and the laptops in the room respond. This happens because the MM Remote Control posts events that include the link to the media contents into the Event Heap. The MM Event Heap client on each laptop listens to these events and shows an alert window with a link allowing the user to display the content locally.

In Step 6, we have an example of a highly specialized application – 3D editing – being brought into the space on a laptop, showing on a public display, yet it is synchronized with the other machines. The first kind of synchrony allows files from the eTable to be moved into applications running on the laptop. In Step 7, the MM Event Heap is transferring mouse events from other devices to control Victor's laptop; the iROS system also supports other shared I/O controls, so that any user can use their laptop, PDA, or other device to control the shared projector or any of the devices that users are willing to share [15].

As the meeting ends (Step 8), the eTable is cleared and the Event Heap is shut down by "clear" button. This again illustrates the appliance character of our device. As we discussed when describing the eTable, the social etiquette for the internal storage of the MeetingMachine is intended to resemble a conference room table. They will need to clear off the eTable when they finish, by pressing the Clear button. As with today's digital projectors, users shouldn't expect that documents used with the eTable are securely removed when they leave. However, they should expect that any remaining digital traces of these documents can only be obtained by extraordinary effort on someone's part.

## 3.7. Prototype with UbiWISE

Our initial design ideas were simulated in Ubiwise [6] using synthetic images for the MM Remote Control and digital photographs of CompactFlash and RFID objects. Several designs were discussed among ourselves and shown to our colleagues; the physical device design changed dramatically in the process. Our software became progressively more realistic once the physical design was clearer. Initially the simulated devices communicated internally through function calls; in the second iteration the simulated devices communicated via the Event Heap. Then laptop computers equipped with IROS software were integrated with the simulated devices. The final prototype uses a backPaq [1] equipped PDA running Java over Linux for the MM Remote Control and a laptop connected to a conventional projector for the MM Projector.

## 4. Discussion and Related Work

### 4.1. MeetingMachine vs. Interactive Workspaces

Full interactive workspaces using iROS enjoy a variety of sophisticated applications that can run over the Event Heap, whereas the MeetingMachine in its current form supports only Multibrowsing, PointRight and SmartPresenter. On the other hand, the MeetingMachine is a zero-administration appliance: turning it on is sufficient to start a meeting with this basic functionality. If the MM software should fail or wedge, "rebooting" the MM by turning it off and on presents no problem (once it reboots, the MM applications will pick up where they left off due to the Event Heap's design) and, if the fault is transient, this action will likely cure it. We have sought to package a small but useful subset of full interactive workspace technology in a dedicated appliance.

### 4.2. MeetingMachine vs. Purely-Nomadic Solutions

For nomadic users a peer-to-peer solution that requires no room-based technology could be considered. Despite the apparent architectural appeal, a centralized model is preferable for several reasons. First, as argued in [27], a centralized model is easier to design, implement, and reason about. Second, a centralized model provides a natural mapping onto the Boundary Principle, which states that the application-level boundary of a ubicomp environment must be made explicit to applications running in that environment. In our case, the Event Heap centrally defines that boundary. Third, in the case of a meeting, there already exists logical centralization in the form of a shared public display that is the focus of attention and physical centralization (the projector, screen, physical table or eTable for shared document storage, and other artifacts that are shared during the meeting but dissociated from the meeting once it ends). The mapping from the inherent centralization of a meeting to a centralized appliance is a natural one.

## 4.3. Related Work

As a physical environment, a conference room equipped with a MeetingMachine most closely resembles the Pebbles work of Brad Myers et al. [25]. That system has PDA-like devices controlling a computer dedicated to controlling a shared projector. We also dedicate a PDA-like device to be the projector control and it could therefore host Pebbles-like functionality. Our system supports lower function interaction – memory media and id tags—as well as higher function interaction in the form of coordinated laptops.

Trevor et al. discuss issues in personalizing shared pervasive devices [29]. They use tags carried by users that that resolve to personalization information for pervasive devices the user may encounter. This use model is quite close to part of our user scenario. While we have only a single device, it has more modes of interacting with users.

Wakikawa et al. [31] describe the use of "webphones" to control appliance functions in rooms. These handheld devices are considered to be ubiquitous personal devices for interacting with pervasive infrastructure. The Satchel project [10] used media and media references in an office setting, but it dealt more with enabling a single person's use of a ubiquitous infrastructure than meetings.

The use of physical objects with electronic identifiers to move media around a workspace has been explored in several projects including mediaBlocks [30], Passage [22], Parctab [32], and Paper++ [26]. Carrying URLs around in PDAs or cellphones [20] has a similar flavor. We view our work as a promising application of these technologies.

## 4.4. Existing products

Two products currently (5/2003) on the market add some support to projectors for wirelessly connected laptop users. The **Katun Komatsu AirProjector Wireless Presentation System**, [http://www.katun.com] offers wireless transmission of video from a laptop to a projector. Because video is transmitted, only wireless laptop users are supported and then only to present slides. The **Linksys Wireless Presentation Player (WPG12)** [http://www.linksys.com] offers wireless transmission of images of slides into a device that attaches to a projector. Because only images rather than full documents are transmitted, no animation or other special effects that might be used in a presentation are available. No data can be read back so file sharing is not supported; neither media users nor ID tags can be used. While these products have limited functionality their appearance on the market does suggest that users are interested in conference room devices to support meetings.

The set of products from Colligo [http://www.colligo.com/] are designed to enable collaboration among peer-to-peer ad hoc groups of users with wireless PCs and PDAs. They provide the tools for creating a local virtual sub-net among the aggregated devices so that users can transfer files, share printers, chat, or engage in NetMeeting sessions, when no fixed network infrastructure is present or when users are members of different organizations. Notable differences from MeetingMachine are the lack of focus on presentation and shared control, and no binding between physical and virtual entities. Also, as discussed in section 4.2, we believe having a central infrastructure piece is a better approach than peer-to-peer in meeting situations.

## 5. Conclusions and Future work

Our main challenge in this work was to try and combine the tools developed for ubiquity on a global scale with those for ubiquity in a meeting room, and then to shrink them into a couple of small, practical boxes. This juxtaposition may seem extreme. On the one hand our appliance is "just a projector control box"; on the other we claim to bring an entire ubiquitous computing experience into a room by putting that box on a table. We hope that this juxtaposition becomes our contribution and that our efforts will encourage others to adapt some of the many promising ideas in ubiquitous and pervasive computing into practical devices that can be deployed.

As a fusion of interactive workspace and nomadic computing technology, we have turned some existing ideas around. Like interactive workspaces, we support multiple users interacting through coordinated applications. Unlike interactive workspaces, we expect most of the applications to be running on laptops that appear in the space only for a short time. Like the nomadic sensor-enhanced mobile web browser [4], we support resolution of physical hyperlinks to virtual media as a integral part of the meeting experience. Unlike the mobile web browser, our browser is integrated with the shared Event Heap so the mobile (and fixed) devices in the room see the media as well. Furthermore we use technology of physical hyperlinks to help nomadic users get connected to room-based infrastructure quickly and easily.

While we believe we have a solid and practical starting point in our MeetingMachine prototype, numerous challenges and opportunities remain. As designers we are satisfied that these challenges have been met in our prototype. Of course that does not mean users will find utility in our design. As an appliance, the user interface and functional design of the MeetingMachine require modification and validation with careful user studies. Therefore our next step will be to create a fully functional deployment of the MeetingMachine. Beyond this, we hope to expand the scope of interactive room technologies that we can host in this appliance on several fronts. First we can attempt to import and adapt technologies from the ongoing work in interactive workspaces [14]. Second we can push forward on technologies specific to the use model we have in mind, such as more spontaneous interaction between applications running on laptops. Finally, we can push more nomadic computing ideas into our prototype to try and support a more diverse set of scenarios.

# References

1. Mercury BackPAQ Research Prototype, http://www.handhelds.org/z/wiki/BackPAQ.
2. MIT Project Oxygen : Software Environment, . 2001, MIT Laboratory for Computer Science, MIT AI Laboratory http://oxygen.lcs.mit.edu/Software.html.
3. Abowd, G., J. Brotherton, and J. Bhalodia. Classroom 2000: a system for capturing and accessing multimedia classroom experiences. in CHI 98: Human Factors in Computing Systems. 1998. Los Angeles, CA USA: Association for Computing Machinery: p. 20-21.
4. Barton, J., et al. Sensor-enhanced Mobile Web Clients: an XForms Approach. in WWW2003. 2003. Budapest, Hungary: ACM Press: p.
5. Barton, J., T. Kindberg, and S. Sadalgi, Physical registration: configuring electronic directories using handheld devices. IEEE Wireless Communications, 2002. 9(1): p. 30-8.
6. Barton, J. and V. Vijayaraghavan, UBIWISE, A Simulator for Ubiquitous Computing Systems Design, HPL-2003-93. 2003, Palo Alto, CA: Hewlett-Packard Labs. 17. http://www.hpl.hp.com/techreports/2003/HPL-2003-93.html.
7. Bellotti, V. and S. Bly. Walking away from the desktop computer: distributed collaboration and mobility in a product design team. in Conference on Computer Supported Cooperative Work 1996 (CSCW-96). 1996. Boston, MA, USA: ACM Press: p. 209-218.
8. Caswell, D. and P. Debaty. Creating Web representations for places. in Handheld and Ubiquitous Computing Second International Symposium HUC 2000. 2000. Bristol, UK: Berlin, Germany : Springer-Verlag, 2000: p. 114-26.
9. Debaty, P. and D. Caswell, Uniform Web presence architecture for people, places, and things. IEEE Personal Communications, 2001. 8(4): p. 46-51.
10. Flynn, M., et al., The Satchel system architecture: mobile access to documents and services. Mobile Networks and Applications, 2000. 5(4): p. 243-58.
11. Goland, Y., et al., HTTP Extensions for Distributed Authoring - WEBDAV. Request For Comments (RFC), RFC-2518. 1999: Internet Engineering Task Force. http://www.webdav.org.
12. Huang, A.C., et al. Running the Web backwards: appliance data services. in Ninth International World Wide Web Conference. 2000. Amsterdam, Netherlands: Netherlands : Elsevier, 2000: p.
13. Johanson, B. and A. Fox. The Event Heap: a coordination infrastructure for interactive workspaces. in Fourth IEEE Workshop on Mobile Computing Systems and Applications. 2002. Callicoon, NY, USA: Los Alamitos, CA, USA : IEEE Comput. Soc, 2002: p. 83-93.
14. Johanson, B., A. Fox, and T. Winograd, The Interactive Workspaces project: experiences with ubiquitous computing rooms. IEEE Pervasive Computing, 2002. 1(2): p. 67-74.
15. Johanson, B., et al. PointRight: Experience with Flexible Input Redirection in Interactive Workspaces. in ACM Symposium on User Interface Software and Technology (UIST-2002). 2002. Paris, France: p. 227-234.
16. Johanson, B., et al. Multibrowsing: Moving Web Content across Multiple Displays. in Ubicomp 2001. 2001. Atlanta, GA, USA: p. 256-272.
17. Kindberg, T. Implementing Physical Hyperlinks Using Ubiquitous Identifier Resolution. in WWW2002. 2002. Honolulu, HI, USA: ACM Press: p.
18. Kindberg, T. and J. Barton, A Web-based nomadic computing system. Computer Networks, 2001. 35(4): p. 443-56.
19. Kindberg, T., et al. People, places, things: Web presence for the real world. in Third IEEE Workshop on Mobile Computing Systems and Applications. 2000. Los Alamitos CA USA: Los Alamitos, CA, USA : IEEE Comput. Soc, 2000: p. 19-28.
20. Kindberg, T., et al., People, places, things: Web presence for the real world. Mobile Networks and Applications, 2002. 7(5): p. 365-76.
21. Kindberg, T. and A. Fox, System Software for Ubiquitous Computing, in IEEE Pervasive Computing. 2002. p. 70-81
22. Konomi, S., C. Muller-Tomfelde, and N.A. Streitz. Passage: physical transportation of digital information in cooperative buildings. in Cooperative Buildings, Integrating Information, Organizations and Architecture, Second International Workshop, (CoBuild'99). 1999. Pittsburgh, PA, USA: Berlin, Germany : Springer-Verlag, 1999: p. 45-54.
23. Mark, G., Collaborative Design Within and Between Warrooms. submitted to the Human-Computer Interaction Journal, 2002.
24. Mark, G., Extreme Collaboration. Communications of the ACM, 2002. 45(4).
25. Myers, B., Using Handhelds and PCs Together, in Communications of the ACM. 2001. p. 34-41
26. Norrie, M.C. and B. Signer, Web-Based Integration of Printed and Digital Information. Lecture Notes in Computer Science, LNCS 2590, February, 2003: p. 200-219.
27. Ponnekanti, S., et al. Portability, Extensibility and Robustness in iROS. in 1st IEEE International Conference on Pervasive Computing and Communications (PerCom 2003). 2003. Dallas-Fort Worth, Texas, USA: IEEE: p. 11-19.
28. Streitz, N., et al. i-LAND: An interactive Landscape for Creativity and Innovation. in ACM Conference on Human Factors in Computing Systems (CHI'99). 1999. Pittsburgh, PA, USA: ACM Press, New York, NY, USA: p. 120-127.
29. Trevor, J., D.M. Hilbert, and B.N. Schilit. Issues in personalizing shared ubiquitous devices. in UbiComp 2002: Ubiquitous Computing. 4th International Conference. Proceedings. 2002. Goteborg, Sweden: Berlin, Germany : Springer-Verlag, 2002: p. 56-72.
30. Ullmer, B., H. Ishii, and D. Glas. mediaBlocks: physical containers, transports, and controls for online media. in SIGGRAPH 98: 25th International Conference on Computer Graphics and Interactive Techniques. 1998. Orlando, FL, USA: New York, NY, USA : ACM, 1998: p. 379-86.
31. Wakikawa, R., et al. Roomotes: ubiquitous room-based remote control over web phones. in CHI 2001: Conference on Human Factors in Computing Systems. 2001. Seattle, WA, USA: New York, NY, USA: ACM 2001: p. 239-240.
32. Want, R., et al. Bridging Physical and Virtual Worlds with Electronic Tags. in ACM Conference on Human Factors in Computing Systems (CHI'99). 1999. Pittsburgh, PA, USA: ACM Press, New York, NY, USA: p. 370-377.