Augmented Meeting Spaces

Thomas Mauzer
RWTH Aachen University
thomas.mauzer@rwth-aachen.de

Michael Schlimnat
RWTH Aachen University
michael.schlimnat@rwth-aachen.de

ABSTRACT

This paper gives an overview on augmented meeting spaces, which apply computer technology to traditional meeting rooms. We collect design requirements and considerations, present different implementations and compare them with each other by focusing on how well they fulfill our requirements.

Author Keywords

co-located collaboration, shared workspaces, large displays, multi-monitor interfaces, pen-based interfaces, gestural input, multi-user input, digital meeting rooms, meeting spaces, multi-display environments, virtual environments, roomware, meeting tools, ubiquitous computing, window manager, sharing, exchange, comparison, overview

ACM Classification Keywords

H.5.3 Group and Organization Interfaces: Computer Supported Collaborative Work.

INTRODUCTION

Meeting spaces are environments where workers come together for discussions, meetings, collaboration or to work on their own. Work there is traditionally supported by devices like projectors, whiteboards, smart boards, paper documents, pictures, and Post-its. Workers normally take their personal devices with them such as laptops, tablets, and smart phones. An augmented meeting room supports workers with a large screen area, connection abilities to personal devices, sharing information capabilities, interfaces for analogue formats, etc. Typically, these spaces are multi-screen environments and devices in such rooms are connected with each other.

In this Paper we give an overview on augmented meeting spaces. First, design requirements and considerations for such spaces are identified, then, room implementations, an interactive board implementation and software implementations are presented. For that, we have a view on software, hardware and some technical aspects in room implementations. Furthermore, we will point out the most important results of the evaluations and how these were performed. We also describe which solutions of implementations are well designed to support meeting participants. A comparison of implementations, where we concentrate on the question if the implementations are conform to the requirements, follows. Additionally, we have a short look on the evaluations methods and on the questions how significant the results are and if the evaluations help to compare the rooms with reference to efficiency of performing tasks.

Seminar Post-Desktop User Interfaces
The Media Computing Group
Prof. Dr. Jan Borchers
RWTH Aachen University

DESIGN REQUIREMENTS AND CONSIDERATIONS

Design Requirements for Augmented Meeting Spaces

To evaluate and compare different room and software implementations, we present our design considerations for building augmented meeting spaces. The social and technology routines occurring in meeting spaces have been analyzed in [13] and the results are used as basis of our design requirements. Further, the results from a field study at a large steel company conducted by the developers of the NiCE discussion room [5] and the paper about WeSpace [22] extend, beside others, the basis of our requirements.

Use of Personal Devices

The use of personal devices has been observed in [13]: Devices like notebooks and cell phones are frequently brought in by individuals and their usage is a common routine. Notebooks are used ephemerally in meetings, e.g., to take a closer look at presented documents or to check and respond to email periodically. Calls with cell phones were made to quickly gain information and rely it to other individuals. One reason for the use is that mobility was inherent in many positions. In addition, personal devices are also used for data sharing, e.g., showing a document from one's notebook to the other participants [20]. Thus, it is important to support the possibility to connect and reconnect devices at the beginning and while meetings.

Sufficient Space on Shared Screens to Show Persistent Information

The feature of showing persistent information on boards or on a second screen is often used. A second display enhances existing routines, so users opt to use it [13]. A second screen is also used by participants who did not adopted a particular meeting style yet [12]. E.g., while one display is used for presentations and collaboration documents, a second shared display can provide supporting or reference material. If persistent information moves from physical walls to shared display, it will be possible to provide access to team members who are not in the room. Thus, meeting spaces should be able to provide such information either in digital or analog formats.

The effects of the presence and placement of displays on collaborative work in a meeting room has been explored in [12]. Three display configurations are compared: a *single display*, *side-by-side dual displays*, and *opposing dual displays* (see Figure 1). The authors conducted a study where participants, who were divided into groups that used one of the three configurations, had to solve an intellective sense-making task. The collaborative work was best supported by the side-by-side dual display configuration (most "insights" measured), worse by the opposing dual display configuration, and worst by the single display configuration. The groups with dual displays used the second display to compare data and to show additional information (instead of explaining it to others), whereas single display groups also used a whiteboard more frequently to minute persistent information important for the task.

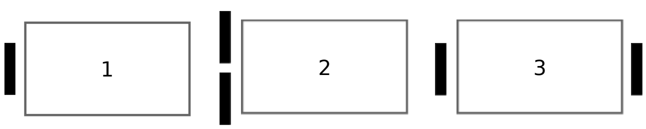

Figure 1. The three shared display configurations: single (1), side-by-side dual displays (2), and opposing dual displays (3).

Since the participants where mostly students they did not adopted a particular meeting style yet. That is why the study may not show how a second display may improve collaboration at real environments in corporations where people have already adopted a particular meeting style; therefore, they may not use the second display. Although, a second display is used in real environments in a similar way described above [13].

Physicality and Visibility of Connections
Physical connection, e.g., with a VGA cable, is a comfort factor as it provides assurance that the user would not show private information when disconnected, and it offers benefits for troubleshooting as a hardware failures like defective VGA cables can be resolved quickly (some attendees feel apprehensive about software-based display connection solutions because of often occurring malfunctions that cannot be fixed in a timely fashion) [13]. Thus, it is important to make connections visible and it must be ensured that users can troubleshoot and fix problems. But, too much physicality can lead to complexity.

Convenient Space
Non-technical factors like lighting, paint colors, seating, and physical layout of the space are critical and attendees desire technology improvements infrequently [13], so designers should consider the physical aspects of a space.

Group Creation and Reachability
Attendees reach for objects when using tangible and table top devices [19], so they must be accessible to users. A meeting space should also provide group creation proximity for groups [20, 16]. The physical arrangement is also important. When an user is seated away from a display, it appears non-interactive to him [11]. The relative seating position to a display influences the interaction ability [23].

Shared Workspaces
Shared content is fundamental [14, 4] and shared spaces should provide both individual and shared workspaces. To create shared content, it is important to have a workplace that provides creating and manipulating shared content collectively.

Individual Workspaces
If the meeting space provides individual workspaces as a socially safe area, users will be able to work undisturbed and unobserved, and it is important that users can prepare ideas before presenting them to the group [6, 24]. The importance of privacy has also been shown in [13].

Smooth Transitions between Individual and Shared Spaces
The collaborative environment should be integrated. Therefore, the transition of data between individual and shared places should be smooth [15].

Support for Multiple and Interrelated Content Types
Content types in group meetings and work is often extensive and heterogenous. For example, software development produces code that may be interrelated with Word documents from the requirement engineering and handwritten notes with implementations ideas.

Content Sharing between Personal Devices and Shared Workplaces
Attendees should be able to make data on their personal devices available to other participants. Meetings are usually part of overarching activities and it is important that users can easily transition from one activity type to another [20]. Digital personal data is traditionally provided by personal devices like notebooks or cell phones.

Egalitarian Input
Discussion groups are often diverse, where each participants has its own expertise. Interviews with users have shown that group members must have the same opportunity for control of the discussion all the time [22].

Recording Ability
Working meetings often involve collaborative work products, e.g., diagrams, which are often drawn on a whiteboard. Attendees want to have a possibility to store work products [22].

Optional: Support for Existing (Off-the-Shelf) Applications
In some environments it is crucial that users can use their own software tools, e.g., the physicists at WeSpace, but this feature has not to be important in other environments. Therefore, it is not a requirement for all spaces.

Versatility and Compelling Benefits

Meeting spaces should support a wide-range of tasks and should have different—sometimes redundant—methods and tools for supporting tasks and routines. An example is given in [13]: A software failure—a remote software lost its connection and the connection could not be reestablished—was solved by using a network drive as backup system. Existing paradigms may only be replaced by new ones if these provide compelling benefits and if people understand how to use provided technologies. E.g., one attendee used his cell phone for capturing the whiteboard content because he did not know how to use the integrated capturing system to digitalize the analog whiteboard content [13].

If the target group of a meeting room is more familiar with computer technology, we think that methods can also be more software-orientated and more complex with a more efficient use after a short learning process. Backup systems and redundancy may be a more important requirement in real meeting rooms outside of lab environments but not all rooms we present have been deployed in real environments over a long time.

Challenges of the Real World

When designing an interactive workspace it is also important to consider where and how a new system is deployed. An example of how a system failed in a real world environment is shown in [6]. The authors introduce the MERBoard, a touch-sensitive plasma screen that was designed and deployed for the Mars Exploration Rover Mission of NASA Jet Propulsion Labs to facilitate collaboration. For the mission several scientists, who did not knew each other, came together to work co-located within a large room, which provided two projectors and a MERBoard for each group working in the room. The MERBoard was not used as expected from the developers. For example, the whiteboard function received less use. Only the long-term planning group used the MERBoard often because it provided a certain application that was useful for the tasks this group had to fulfill.

The users were not trained in using the MERBoard and did not experimented and explored on it because they did not feel a sense of ownership over this shared resource. Furthermore, the scientists just had no time to learn how to use the MERBoard. Everything the MERBoard can do could be done with other more traditional techniques except the functions of SolTree, the application used by the long-term planning group. SolTree allowed the users to easily draw tree structures to plan actions for the rovers. Another problem was that the firewall prevented the scientists to directly access their PCs from the MERBoard. Users who wanted to use the board for collaboration sometimes did not know if the MERBoard was in use because the board was often used to display a fullscreen mars-time clock which could be seen from the personal workspaces. The developers had no chance to adjust the system to the needs of the scientists because it was forbidden to change systems during the mission.

The example of the MERBoard shows that users will not use a new (shared) system when they are not shown how to use it. Additionally, users will more easily use a new system when it has functionalities that traditional systems do not have.

ROOM IMPLEMENTATIONS

We present several implementations of augmented meeting rooms from 1987 to 2010. They implement hardware and software.

Early Approaches at Xerox PARC

Colab

In 1987, the Colab [17]—an experimental meeting room at Xerox PARC—has been presented. The room is designed for the use of two to six persons and consists of personal workstations (one for each person), a large touch-sensitive screen, and a stand up keyboard, which are connected with each other over a local network. The same software is running on each device. Different tools support group interaction and problem solving in meetings: *Boardnoter* is an informal meeting tool for freestyle sketching and imitates a chalkboard, *Cognoter* is used for organizing ideas to prepare presentations collectively, and *Argnoter* has been developed for presenting and evaluating alternate proposals.

Users can act simultaneously and the data is stored in a shared database. Due to limited computer and infrastructure performance, each machine has a copy of the database and changes are broadcasted to ensure short delays in the interface. Conflicts have to be manually resolved or prevented by social constraints and verbal clues. In *Cognoter*, actual editing is done in private windows but finished text is broadcasted to all co-participants, whereas *Argnoter* also provides private windows. The authors use the term WYSIWYS ("what you see is what you see") which means that users have the impression to work with shared and tangible objects. Strict WYSIWYS would give everyone the same image on their displays. Since this would be very limiting, the Colab uses a relaxed version so that private windows are allowed. Early observations—small sets of controlled experiments with several pairs of student collaborators—have shown that the interface of *Cognoter* needs practice to be used effectively.

Liveboard

The Liveboard [3] from 1992 is an interactive display system for computer-supported meetings and complements personal computer devices as it provides a shared workspace for collaboration. It is based on a rear-projection screen with a display surface of 1.2 m x 0.8 m. People can use a cordless pen for input which allows remote pointing, gestural input, and the functionality of a three-button mouse to be used with existing software. The Interface is called *BoardWalk*; the *BoardWalk* control panel is displayed in the lower left corner of the display. It contains a list of planks, which are a set of applications that automatically opens when a user choose a plank. E.g., the *Meeting*-plank includes a whiteboard application, a text editor, and a clock. The *SlideShow*-plank opens the *SlideShow* presentation tool. All sheets of the whiteboard are remembered and can be saved to a file or printed, while saved sheets can be retrieved on any Liveboard. The whiteboard can be used both for taking notes at informal meetings, and for presentations.

12 Prototypes were built and ten placed at PARC and two at XEROX. They were used by different user groups, e.g., managers and software developers. The survey consists of an e-mail questionnaire. 34% of people using the Liveboard did not know how to turn it on and did not want to try things out—some because they where afraid of making mistakes and damaging it. Most often it was used in meetings and mostly used for the whiteboard, which was the default application that came up after starting *BoardWalk*. Also mentioned was that people wanted to have a better image quality and the pen to be more accurate. The problem that people did not use the Liveboards was solved by starting the Liveboards before people arrived and to let *BoardWalk* start automatically. This is quite a bad idea when you want to save energy, and a better placed power-on button or a moving sensor may be better solutions. Furthermore, the machine is automatically restored after being idle for a specific time. Thus, users are enabled to solve software failures by restarting or waiting for a restore.

The Liveboard realizes a good trade-off between ease-of-use and the complexity of the applications. The developers payed attention to let the Liveboard be usable by non-computer-professionals and the pen allows familiar control, but the Liveboard itself provides no connection possibility neither for personal devices nor for extra displays to display additional or persistent information. Ubicomp (see next section) uses Liveboards and connects them with other devices.

Ubiquitous Computing (Ubicomp)

Mark Weiser presented the idea of ubiquitous computing in 1991 [21]. The idea is that computer technologies disappear in fabric of everyday life, become invisible, and augment the reality. The devices that are used at Xerox PARC are tabs (centimeter scale, 100 per room), pads (decimeter scale, dozens per room), and boards (meter scale, one or two per room); wired and wireless networks link these devices. The Boards are Live Boards which has been described above. Rooms with Boards can be used as meeting spaces and are also augmented with tabs and tablets.

i-LAND

The i-LAND [18] is a space for collaborative work which integrates the real architecture space into the whole implementation and has focused on the support for creative teams when it was presented. The scientists conducted an empirical study with creatively working teams at five companies from the automobile and oil industry. The result was that “the teams wanted to have much freedom in (re-)configuring their physical environment and their information environment”. The implementation consists of several *roomware* components which are “computer-augmented objects resulting from the integration of room elements, e.g., walls, doors, furniture (tables, chairs, etc.) with computer-based information devices”. In the current implementations these are an interactive wall called *DynaWall*, a mobile interactive table called *InteracTable*, and mobile chairs called *CommChairs* with integrated interactive devices. The software of the i-LAND is called BEACH.

The Dynawall provides the ability to work individually or collaboratively on a 4.5 m x 1.1 m display area. To drag an object or a window, it provides the mechanisms *take and put* allowing to take an information object at one screen position, walk over without contact to the DynaWall and put it somewhere else on the screen, and *shuffle* allowing to throw objects—with different accelerations—which can be caught by others. The CommChairs exist in two versions, one with a docking facility for a laptop and one with an integrated pen-based computer. Users have a private space for making personal notes and can also interact remotely on shared workspaces, e.g., making annotations on the DynaWall. Thus, users have the ability to communicate and share information with other people. Each CommChair has a wireless network connection and an independent power supply for flexibility and mobility. The InteracTable provides the ability for group work. Users can write and draw with a pen, interact via finger or pen gestures, and use a wireless keyboard for extensive text input. For easy viewing from all perspectives it provides gestures for rotating and shuffling single information or groups of information. The *passage concept* allows to connect digital information with a physical object, e.g., a watch, and carry it physically to a new location by putting it on a device called *bridge*. Such a physical object is called *passenger* and must be recognizable by the bridge and be unique.

Flexible and dynamic creation of workspaces is also possible, e.g., users can form a subgroup by moving chairs together. The i-LAND is a first implementation of the *roomware* approach which is part of *collaborative buildings*, a more general framework. The flexibility and mobility of the components together with the flexible and dynamic creation of workspaces are very interesting because static rooms may prevent subgroups from sitting in close proximity to each other and, thus, prevent collaboration.

iRoom

The interactive workspace project [9] at Stanford University is a long-term project which addresses solutions for an augmented meeting room. The authors constructed a prototype interactive workspace, the iRoom, and created a software infrastructure called *iRoomOS* (*iROS*). The test setting of the iRoom used by the developers consists of three smart boards, a 1.8 m diagonal display, a 0.6 m x 1.2 m display table, cameras, microphones, wireless LAN support, programable wireless buttons, and other interaction devices. The iRoom can easily be changed to other layouts what makes it deployable at various environments. The *iROS* is a meta-OS that ties together devices which have their own low-level OS. Components of the *iROS* are for example the *Data Heap*, *iCrafter* and the *Event Heap*.

Applications running on the machines which are part of the iRoom can place data into the *Data Heap*. The *Data Heap* stores a number of attributes for each date to characterize it which makes it file system independent. It automatically transforms data to the best format supported by the retrieving application (e.g., PowerPoint files to JPEG if an application does not support presentations). The *iCrafter* provides service advertisement and invocation as well as a user interface generator which returns the best interface for a selected service for the user's device. The *iCrafter* makes it possible to control the iRoom from every device connected to the room.

The *Event Heap* [8] provides application coordination for applications in the interactive workspace. Therefore, the infrastructure of the *Event Heap* is derived from a tuplespace model. Tuples can be placed into the tuplespace and can be accessed by patterns of the wanted tuples. The *Event Heap* is language independent, so application written in multiple programming languages can implement it. Applications can place events in the *Event Heap* which can be read by other applications. Since applications do not communicate directly with each other, the system provides good failure isolation. Furthermore, that makes it possible for applications to receive messages even when the sender is not active anymore. If the *Event Heap* server fails, it is easily restartable and clients automatically reconnect after a server restart, so an inexperienced user should easily be able to just restart the server if something fails to fix a problem. The iRoom has already a set of tools implemented for some standard tasks, navigation [10], and room control, but it also relys on user applications which are modified to implement the *Event Heap* to integrate it into the interactive workspace.

The iRoom was used by a group of civil engineers working on construction management, student project groups in courses, design firms for brainstorming meetings, school principals for training-simulation meetings, and the developers themselves for their meetings. The civil engineers used a software suite they modified to integrate it into the iRoom. The iRoom was actually used and accepted in a real world environment for collaborative work; the iRoom and the *iROS* can be a good framework to design an interactive workspace for the special needs of a group working together in a certain field of work.

WeSpace

Another interactive workspace that was actually used in a real world environment is the WeSpace [22]. The authors put a multi-surface environment into actual use by a scientific user group—a group of three astrophysicists. The WeSpace uses a large shared high-resolution display and a multi-touch display table for shared user input. The high-resolution display makes it possible to analyze and work with visual data. The table makes input more egalitarian and visible to other users, and it facilitate face to face collaboration. Users can connect their laptops to the WeSpace and control them via the tabletop display after installing a lightweight client. They can share, show, and, most important, work with data directly without sending it to other computers. If the touch input is not sufficient, the laptops can also be controlled by their traditional input method.

The advantage of working with your own computer compared to working with your data on a shared machine is that own applications can be used. This is important when collaborating in a special field of work where special applications are needed. The WeSpace can render multiple screens simultaneously, so it is possible to overlay different laptop screens to compare and analyze them. The WeSpace API allows developers to make their own applications for it. The developers of the WeSpace already implemented two applications: The *Layout Manager* and *LivOlay*. The *Layout Manager* allows to have an overview over the connected laptops and lets the user choose which screen should be in focus on the large display and which laptop should be controlled over the tabletop surface. *LivOlay* lets the users overlay their live laptop screens. For that, they have to select landmark points on their screen which should be overlaid. *LivOlay* transforms the screens automatically such that corresponding landmark points are overlaid.

The WeSpace was in actual use by a group of astrophysicists and iterative designed to fit their needs. The WeSpace changed and improved their workflow by making collaborated work possible in the first place. Especially, the overlay function helped them in research and even helped them making new discoveries. The tabletop display made it possible to use physical tools like markers or rulers, to move virtual objects more naturally, and to write notes with a stylus. Thus, the WeSpace shows that a multi-touch tabletop display combined with a large high-resolution wall-display can facilitate collaboration when the right tools for certain tasks are provided by the system. In the case of the astrophysicists this tool would be the overlay function.

The NiCE Discussion Room

The latest attempt to build an interactive meeting room is the NiCE Discussion Room [5]. The room consists of three sketching walls, a paper interface, tangible palettes for interacting with the whiteboard, and movable furniture. Users can put media on the sketching wall and organize or annotate it. To put media on the wall, the user can write directly on it with a stylus, connect a laptop via VGA cable to the system to capture the screen, or use the paper interface. The advantage of using a simple VGA connection to capture the screen is that the users do not have to install a client on their laptop. The sketching walls consist of the whiteboards and short throw projectors above them. The whiteboard and the paper interface use Anoto digital ink pens and Anoto pattern paper to digitalize user input.

One of the benefits of using a paper interface is that sketches can first be drawn in private, and the user can decide if she or he want to share it or throw it away. Another advantage of the paper interface and the laptop input is that the participants of a meeting can remain seated while contributing to a discussion. The room should facilitate different kinds of meetings, so the movable furniture should provide that. The digital whiteboard can be controlled with embedded magnets which can be put on the whiteboard or used remotely. There are tangible menus for changing the function of the pen (e.g., drawing, erasing, highlighting, pick and drop) and creating new layers. Instead of the tangible menus, digital pie menus can also be used. There are magnets for three types of layers: the overview layer, the screen capture layer, and paper layer. The overview layer shows thumbnails of all canvases and highlights the currently active canvas. The different layers can be moved, written on, etc. by the pen.

To evaluate the NiCE discussion room, thirty-nine participants from a local software engineering company had to perform a collaborative design task in groups of three. Most groups used all features of the room. Only about the half of the participants were satisfied with the whiteboards interaction. The problem was that the surface was too large: If the group members stood in front of the whiteboard, they had no overview over the other group members activities. Overall, the data transfer was easy for the participants. The most problems were related to the use of overlays as users did not understand this feature.

The study shows that the room is not ready to be used in a real world environment and needs to be improved in some places, but, when the developers fix the problems that the participants of the study had, the system can maybe prove to be useful for actual meetings. Particularly, the paper interface can prove itself as an intuitive input method for such systems.

SOFTWARE IMPLEMENTATIONS

Several implementations of software could be used in augmented meeting rooms. They provide, e.g., window relocating or data sharing.

ARIS

ARIS [1] is an interactive space window manager and is used in the interactive workspace *Active Spaces*. The interface allows users to relocate an application and redirect local input. The user can combine these tasks or perform them separately.

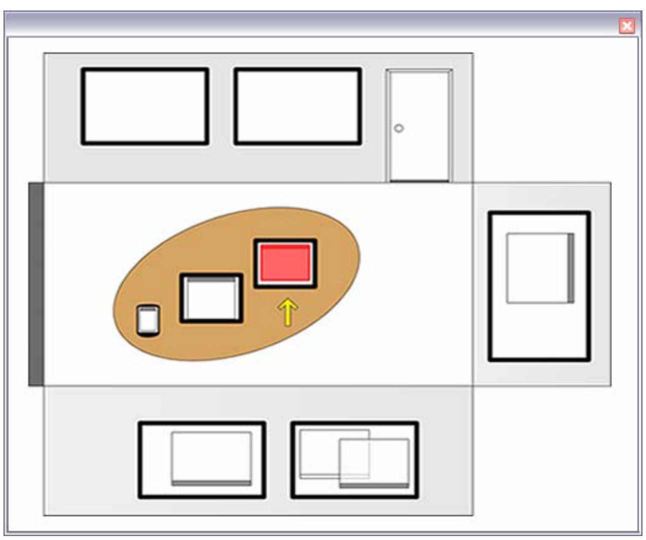

Figure 2. The iconic map of ARIS.

ARIS uses an iconic map (see Figure 2) which shows not only all screens in the space like plasma screens, PDAs, laptops, tablets, and more, but also the walls and doors for orientation. Furthermore, a direction arrow, which shows the position and orientation of the user in the space, is displayed. It also shows the application windows with a bar, which represent orientation (the bar is displayed at the top, bottom, or side of the windows) and focus (the bar is drawn darker or lighter). When a user wants to relocate a window, she or he selects a button on the title bar of the window. ARIS starts, the iconic map appears, and the user can drag the representation of the window to the destination screen, on which an outline of the window to relocate appears. The users releases the pointer, ARIS closes, and the relocation is finished.

The development followed an iterative design process, thus, the solution is more or less based on user needs. We approve the good orientation-supporting design of the iconic map so that the user has a connection to the environment. Supporting small up to big-screen devices matches the requirement of supporting personal devices. All screens can be used for everything and input can be redirected, therefore, this system is quite flexible. Disadvantages are the necessity of installing ARIS on every device and the necessity of tracking the devices if the iconic map should represent the environment dynamically.

Dynamo

Dynamo [7] is a system for public meeting spaces for users outside their familiar organizational setting to share and exchange digital media cooperatively. Dynamo is designed to support a wide range of meeting styles, e.g., spontaneous ad-hoc meetings as well as long planned ones. It supports multiple displays and input devices, and users can use their laptops or PDAs for input. The system is able to access data from USB storage devices, laptops, and PDAs.

Dynamo provides a shared desktop where users can claim areas for their own use. Digital media can be dragged from own storages to the surface to share or show them. Furthermore, it is possible to create notes and leave them on the surface. Dynamo supports a range of standard media formats for video, image, or presentation files. It has a built-in browser and it is possible to connect printers or cameras to the system. Since multiple users can control the Dynamo surface at the same time, their cursors have to be distinguishable. Therefore, each user registered to the system has her or his own color and the user's cursor has this color. User profiles and private spaces can be stored for later use, but it is also possible to use Dynamo as a guest user without having to create an own profile. Users who carved-off part of the main surface for their private space can manage which user can access this area and the media within it. Multiple media items can be bundled in parcels for further share. The creator can set the permission for opening and accessing them.

An early version of Dynamo was tested at a workshop. The system was placed in the foyer that was open for everyone to test. The carve technique and drag-and-drop metaphor which the system uses needed explicit explanation. Thus, in a real world environment users maybe need an explanation before using the system what can make them not wanting to use Dynamo at all. The traditional alternative to Dynamo is showing digital media directly on the users laptop or on a projector and sharing it via network or USB storage devices.

IMPROMPTU

IMPROMPTU [2] is an interaction framework for collaborating in multiple display environments. It supports off-the-shelf applications without modifications. Thus, users can share a great amount of information types. The framework also supports joint interaction on both personal and shared displays, and multitasking among shared applications. The authors speak of a lightweight interface, and we agree with this attribute. The interface provides a visual representation of group members and contains the *Collaboration Control*, *Collaborator Bar*, and *Shared Screen Docks*. Figure 3 shows the user interfaces.

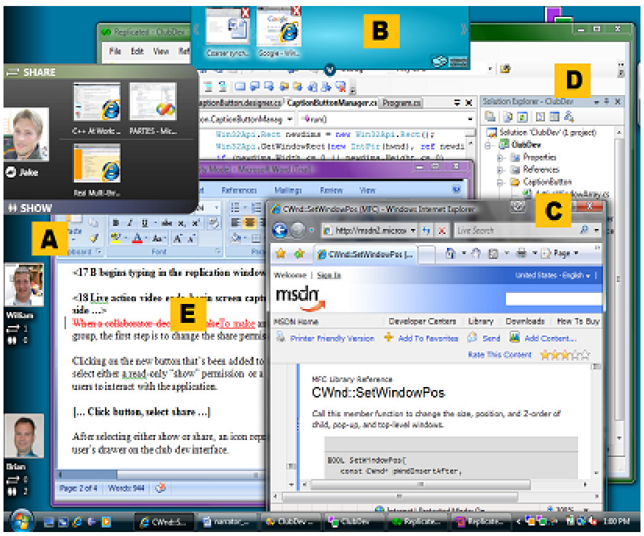

Figure 3. The IMPROMPTU user interface with replicated and local application windows on a user's machine, the collaborator bar (A), the shared screen dock (B), control for availability to the group and level of control (C), a replicated window in share mode (D) and a replicated window in show mode (E).

The *Collaboration Control* is located on the title bar of every top-level application window as a button and lets the user choose an entry from a list wether the application should be not available to group members (*Do not show or share*), available to group members in a view-only mode (*Show*), or available to group members and any group member can interact with the content of the window (*Share*). The icon of the button represents the current sharing mode. The *Collaborator Bar* provides a representation of each group member. Each representation has a drawer with two rows that animates out when moving over the corresponding one. The top row shows thumbnails of applications that have been set to *Share*, while the bottom row displays applications that have been set to *Show*. Users can drag and drop a thumbnail onto the desktop, initiating a replication of the window, and see the cursors of each other's within replicated windows.

Each shared display in the space has its own *Shared Screen Dock*. While minimized by default, it opens and shows thumbnails of all windows from left to right on the corresponding display when the user moves the cursor over it. Expanded via a button at the bottom, the dock shows a miniature representation of the display content, where users can adjust position and z-order of windows. Any application from one group member's *Collaborator Bar*, including the own representation, can be placed onto a shared display by drag and drop. A shared display can contain replicated windows from different users at the same time. Users can redirect their local input to a shared display by pressing the redirecting button on the dock and can return input to the local device by a key sequence.

A field study conducted with development teams from Microsoft Corp. has shown that users shared not only the code editor but many other applications. Thus, the replication model that allows the use of off-the-shelf-application was a good decision. They appreciated that team members can manipulate one application in parallel and that users can multitask with applications while sharing windows. Furthermore, users appreciated the ability to easily made content available to the group by putting it onto a shared display. This was also useful to see where people are collaborating and, thus, to know where to jump in. The input relocation functions was rarely used, which shows that users prefer performing input actions on their local device. The behavior of performing tasks did not change in a detectable way when using the framework, so it seems IMPROMPTU supports natural collaborative routines. Users also mentioned that it allows new opportunities for collaboration. A main criticism was that users cannot independently manipulate the view of a shared application. The authors argue that the framework is for "opportunistic, short-lived collaborative engagements", so "it was important for users to maintain the *same* view". Overall, the users find value in using IMPROMPTU.

We like the lightweight and clearly arranged user interface. The user interface gives a people-orientated view rather than a strict spatial representation of the space, which was a lesson learned from working with users in the design process. We would agree with this, but a more spatial representation of shared displays like in ARIS could be better in environments with many shared screens. It is thinkable to utilize IMPROMPTU in different meeting spaces for collaboration tasks and for presentation tasks as well.

COMPARISON OF IMPLEMENTATIONS

In the following, we want to compare the software implementations Dynamo, ARIS, IMPROMPTU, as well as Colab, the Liveboard, i-LAND, iRoom, WeSpace and NiCE with each other. Ubicomp is mentioned in various passages. We describe similarities and differences, check how the implementation match the design requirements and how the evaluations were performed.

Comparison of Software Implementations

While ARIS and IMPROMPTU provide window relocation, Dynamo provides data sharing.

Use of Personal Devices

All implementations support this requirement, but ARIS and IMPROMPTU users must install the related software on their device. Additionally, ARIS users have to make it trackable.

Sufficient Space on Shared Screens to Show Persistent Information

This requirement is room dependent. ARIS and IMPROMPTU as window relocating implementations provide this, Dynamo provides notes that can be left on screens with multiple media items organized on it.

Physicality and Visibility of Connections

Devices at ARIS and IMPROMPTU normally stay physically connected, while Dynamo users can physically plug in USB data storages, but they have to connect laptops, desktops, and PDAs remotely.

Convenient Space

Convenience is room-dependent, but all implementations have a more or less intuitive interface. As mentioned before, we especially like the interface of IMPROMPTU.

Group Creation and Reachability

ARIS does not support group formation but only input relocation without access control. Dynamo supports private spaces which can only be accessed by authorized users, therefore, it is possible to create groups. IMPROMPTU provides groups with a shared space. Additionally, the owner of an application has control over the access to her or his applications (*not show*, *share*, or *show and share*).

Shared Workspaces

ARIS offers only interaction with an application for one user at the same time, whereas IMPROMPTU supports shared screens and allows users to interact simultaneously with applications. Dynamo is not a system for collaborative interaction but for showing and sharing data (which definitely facilitate collaborative work).

Individual Workspaces

Only IMPROMPTU users have individual workplaces where they can work unobserved and undisturbed because they have full control over access of own applications. ARIS does not provide access control and Dynamo has no workspaces at all, but it has private areas.

Smooth Transitions between Individual and Shared Spaces

ARIS has no shared workplaces, but transitions can be performed by window and input relocation. The evaluation has shown that IMPROMPTU matches this requirement as it allows parallel manipulation of an application without blocking one user when working on her or his machine. Dynamo combines private and public spaces on one surface for smooth transitions.

Support for Multiple and Interrelated Content Types

Both ARIS and IMPROMPTU fulfill this requirement via supporting off-the-shelf software. Dynamo supports a wide range of media.

Content Sharing between Personal Devices and Shared Workplaces

Both ARIS and IMPROMPTU match this requirement via application sharing. Dynamo is designed for content sharing but does not offer collaborative interaction.

Egalitarian Input

All implementations fulfill this requirement. At ARIS everyone can relocate windows and input, Dynamo provides an own cursor for every user, and IMPROMPTU gives each user controls to access to own applications.

Recording Ability

None of the three implementations provide recording.

Versatility and Compelling Benefits

The iconic map of ARIS could be very useful in spaces with many displays distributed in the whole room, and Dynamo allows dynamic and graphically supported sharing and showing data on one shared surface with private areas. Both ARIS and IMPROMPTU facilitate sharing of a great amount of information types via supporting off-the-shelf applications.

Comparison of Room Implementations

Use of Personal Devices

Every implementation supports the use of personal devices except of Colab and the Liveboard, but workstations can connect and reconnect at any time at Colab. i-LAND provides its *CommChairs*, at iRoom applications on personal devices have to implement the *Event Heap* to communicate with the room, whereas at WeSpace laptops and desktops can be brought into the space on-the-fly, and at NiCE own laptops can be connected via VGA cable to capture their screen content.

Sufficient Space on Shared Screens to Show Persistent Information

This requirement is fulfilled by all implementations except of Colab and the Liveboard as they provide only one shared screen that is not big enough to replace two or more displays. i-LAND offers the DynaWall, iRoom has multiple screens, WeSpace allows to show the content of multiple laptop screens on a tabletop and a wall display, and NiCE provides its large wall display.

Physicality and Visibility of Connections

The workstations at Colab stay permanently physically connected by a network cable, and the Liveboard does not support the connection of devices. i-LAND allows personal devices to be plugged in at the *CommChairs*, provides its *passenger concept*, and allows flexible and dynamic creation of workspaces by physical arrangement of *roomware* components. At iRoom laptops can be connected via VGA connections, but the main communication to the room is invisible. WeSpace offers, beside WiFi, Ethernet cable connections, and NiCE connects laptops with VGA connections like iRoom.

Convenient Space

The Liveboard was used in different meeting spaces and *iROS* from iRoom can be deployed in different rooms. i-LAND offers convenience as the physical arrangement can be customized (*CommChairs* and *InteracTable* are mobile), which allows flexible and dynamic creation of workspaces. WeSpace offers face-to-face collaboration by sitting around a table, and NiCE has moveable furnitures and looks convenient on picture and video.

Group Creation and Reachability
The shared display at Colab can be controlled from the workstation. Users stand in front of the Liveboard and use context menus, but there are no remote controls from other devices (but Ubicomp provides such inputs). i-LAND offers group creation by a flexible and dynamic creation of workspaces and virtual locations, and iRoom and WeSpace offer touchable devices for egalitarian and visible input and let users sit around a tabletop display (face-to-face). NiCE provides tangible menus and a paper interface, and it allows users sitting in distance to the shared display to interact with it.

Shared Workspaces
The Liveboard itself is a shared workplace. Colab provides its shared screen and most actions are public there (WYSIWYS). At iRoom users can work together depending on the used applications. i-LAND and WeSpace offers display walls (interactive at i-LAND and remotely controlled at WeSpace) and tabletops, and NiCE its interactive whiteboards.

Individual Workspaces
This requirement is fulfilled by every implementation except for the Liveboard as it does not provide individual workspaces (Ubicomp provides them by providing tabs and tablets). Colab supports private windows, but the main focus is on collaboration (surely, this is application depending).

Smooth Transitions between Individual and Shared Spaces
This requirement is not applicable to the Liveboard, but the other implementations match it. Most actions in Colab are public (WYSIWYS) so there is a direct transition. Users have remote control from the CommChairs and can use the passenger concept at i-LAND, the other implementations allow the use of own laptops (and the paper interface at NiCE) to create content before sharing it.

Support for Multiple and Interrelated Content Types
Colab uses a database for all contents of its applications, and no further support for different content types was mentioned. The Liveboard supports existing applications, but we do not know how well the special applications provide support for different content types. The format support of i-LAND applications was not mentioned in the corresponding paper. The Data Heap performs conversations to supported formats of a device automatically at iRoom. The supported content types of WeSpace and NiCE depend on the used applications on the users' laptops as the screen content is captured from them.

Content Sharing between Personal Devices and Shared Workplaces
This requirement is not applicable to the Liveboard and Colab. i-LAND provides remote control from the *CommChairs* and has the *passenger concept*, iRoom implements the *Data Heap*, whereas WeSpace and NiCE stream laptop screen content to shared devices (with input redirection at WeSpace).

Egalitarian Input
The interactive boards or walls (Liveboard, i-LAND and NiCE) and the tabletops (i-LAND, iRooms and WeSpace) provide egalitarian input. At Colab, all participants have the same rights and conflict solving relies on social constraints.

Recording Ability
Colab stores content in its database, the Liveboard and i-LAND also support storing information, and NiCE facilitate the recording of sessions. At iRoom, not everything can be recorded but it may be possible with own applications (probably specialized for own software suite). However, WeSpace has no record possibilities.

Versatility and Compelling Benefits
Each implementation has benefits. Colab supports an already existing meeting style with different tools, thus, it can be seen as a special purpose system, whereas WeSpace introduces new routines by providing specific tools for the target group and by supporting existing (scientific) tools; these new routines has offered the target group new possibilities to work. The Liveboard, compared, e.g., with a whiteboard or a projector, allows storing documents and offers different functions for different tasks, e.g, for presentations. i-LAND implements the great idea of intensely integrating the real architecture, while iRoom is customizable by user applications and has a range of pre-implemented tools. NiCE provides input via the Anoto pen on the whiteboard or paper and combines the different input methods.

Measurement
A lot of implementations were evaluated. Various observations, interviews, questionnaires and quantitative measures were performed in different environments. Some evaluations were conducted in lab studies, others in (long-term) field studies in real world environments. The result indicates if an implementation supports participants and if it has a compelling benefit in the evaluation environment or similar ones. The results of the evaluations were quite positive in general. However, the wide range of different measure methods does not allow a good comparison between the implementations.

CONCLUSION AND FUTURE WORK
We collected design requirements and considerations and presented software, board, and room implementations. The versatility has risen over time and technical limitations decreased, e.g., the iRoom is more versatile than the Colab and has less technical limitations. Our design requirements were mostly fulfilled and we have seen many solutions for our design requirements. Further, the evaluations resulted in positive feedback, but we miss evaluations of the implementations at different real world environments because it would be interesting to see the differences of using them in different

environments. Future approaches could focus on more flexible and more universal meeting room implementations that can be deployed in any meeting spaces, while still allowing an easy and intuitive use.

REFERENCES

1. J. T. Biehl and B. P. Bailey. Aris: an interface for application relocation in an interactive space. In *GI '04*, pages 107–116, 2004.

2. J. T. Biehl, W. T. Baker, B. P. Bailey, D. S. Tan, K. M. Inkpen, and M. Czerwinski. Impromptu: a new interaction framework for supporting collaboration in multiple display environments and its field evaluation for co-located software development. In *CHI '08*, pages 939–948, 2008.

3. S. Elrod, R. Bruce, R. Gold, D. Goldberg, F. Halasz, W. Janssen, D. Lee, K. McCall, E. Pedersen, K. Pier, J. Tang, and B. Welch. Liveboard: a large interactive display supporting group meetings, presentations, and remote collaboration. In *CHI '92*, pages 599–607, 1992.

4. W. Hacker. Konstruktives entwickeln: Psychologische grundlagen. In *Denken in der Produktentwicklung. Psychologische Unterstützung der frühen Phasen*, pages 11–26. 2002.

5. M. Haller, J. Leitner, T. Seifried, J. R. Wallace, S. D. Scott, C. Richter, P. Brandl, A. Gokcezade, and S. Hunter. The nice discussion room: Integrating paper and digital media to support co-located group meetings. In *CHI '10*, pages 609–618, 2010.

6. E. M. Huang, E. D. Mynatt, and J. P. Trimble. When design just isn't enough: the unanticipated challenges of the real world for large collaborative displays. *Personal Ubiquitous Comput.*, 11(7):537–547, 2007.

7. S. Izadi, H. Brignull, T. Rodden, Y. Rogers, and M. Underwood. Dynamo: a public interactive surface supporting the cooperative sharing and exchange of media. In *UIST '03*, pages 159–168, 2003.

8. B. Johanson and A. Fox. The event heap: A coordination infrastructure for interactive workspaces. In *WMCSA '02*, page 83, 2002.

9. B. Johanson, A. Fox, and T. Winograd. The interactive workspaces project: Experiences with ubiquitous computing rooms. *IEEE Pervasive Computing*, 1(2):67–74, 2002.

10. B. Johanson, G. Hutchins, T. Winograd, and M. Stone. Pointright: experience with flexible input redirection in interactive workspaces. In *UIST '02*, pages 227–234, 2002.

11. R. L. Mandryk, S. D. Scott, and K. Inkpen. Display factors influencing co-located collaboration. 2002.

12. C. Plaue and J. Stasko. Presence & placement: exploring the benefits of multiple shared displays on an intellective sensemaking task. In *GROUP '09*, pages 179–188, 2009.

13. C. Plaue, J. Stasko, and M. Baloga. The conference room as a toolbox: technological and social routines in corporate meeting spaces. In *C&T '09*, pages 95–104, 2009.

14. G. Schwabe. Gemeinsames material und gruppengedächtnis. In *CSCW-Kompendium : Lehr- und Handbuch zum computerunterstützten kooperativen Arbeiten*, pages 447–454. August 2001.

15. S. D. Scott, K. D. Grant, and R. L. Mandryk. System guidelines for co-located, collaborative work on a tabletop display. In *ECSCW 2003*, pages 159–178, 2003.

16. R. Sommer. Personal space. the behavioral basis of design., 1969.

17. M. Stefik, G. Foster, D. G. Bobrow, K. Kahn, S. Lanning, and L. Suchman. Beyond the chalkboard: computer support for collaboration and problem solving in meetings. *Commun. ACM*, 30:32–47, January 1987.

18. N. A. Streitz, J. Geißler, T. Holmer, S. Konomi, C. Müller-Tomfelde, W. Reischl, P. Rexroth, P. Seitz, and R. Steinmetz. i-land: an interactive landscape for creativity and innovation. In *CHI '99*, pages 120–127, 1999.

19. A. Toney and B. H. Thomas. Considering reach in tangible and table top design. *TABLETOP 2006*, pages 57–58, 2006.

20. J. Wallace and S. Scott. Contextual design considerations for co-located, collaborative tables. In *TABLETOP 2008*, pages 57 –64, 2008.

21. M. Weiser. The computer for the 21st century. *Scientific American*, pages 94–104, September 1991.

22. D. Wigdor, H. Jiang, C. Forlines, M. Borkin, and C. Shen. Wespace: the design development and deployment of a walk-up and share multi-surface visual collaboration system. In *CHI '09*, pages 1237–1246, 2009.

23. D. Wigdor, C. Shen, C. Forlines, and R. Balakrishnan. Effects of display position and control space orientation on user preference and performance. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*, CHI '06, pages 309–318, 2006.

24. S. Wilson, J. Galliers, and J. Fone. Not all sharing is equal: the impact of a large display on small group collaborative work. In *Computer supported cooperative work*, pages 25–28, 2006.