

**Collaborative Interaction And Display Space Organization**  
**In Large High-Resolution Environments**

BY

RATKO JAGODIC

B.A., Computer Science, Chemistry, Lake Forest College, 2003

THESIS

Submitted as partial fulfillment of the requirements  
for the degree of Doctor of Philosophy in Computer Science  
in the Graduate College of the  
University of Illinois at Chicago, 2011

Chicago, Illinois

Defense Committee:

Andrew Johnson, Chair and Advisor  
Jason Leigh  
Luc Renambot  
Thomas Moher  
Erik Hofer, University of Michigan

## ACKNOWLEDGEMENTS

First and foremost I have to thank my advisor Andy as he has contributed the most to the success of this work. He was never too busy to discuss ideas or provide feedback. Although at times brutally honest, his feedback has contributed greatly to my ability to observe different perspectives. This is something I will benefit from for the rest of my life. Similarly, Jason was always there to provide high-level guidance with as little as a sentence or an eye-opening question that made me rethink ideas from scratch. Much thanks also has to go to Luc for his unprecedented expertise in pretty much everything and his willingness to share it. When it comes down to getting the needed equipment and getting it running, Lance and Alan have been the most knowledgeable I've ever had a pleasure to meet. I have bothered them countless times and have always received prompt attention, despite their busy schedules. Although my writing skills leave much to be desired, Maxine has always happily read my materials and even encouraged me to ask her for help. Her personal attention to every person in the lab is largely responsible for keeping EVL together as a family. She has also been of tremendous help in my job searching process, without which I would be swimming alone and lost. Taking care of administrative tasks would be impossible without Dana's expertise and unlimited energy she brings to the lab.

I also have many friends to thank for helping me preserve sanity during the PhD process. Sungwon has been a true friend with whom I was able to share honest opinions and thoughts, as he understood my worries the best. Often, I would selfishly demand his attention, which he never refused to give, even when he needed it himself. Numerous

philosophical discussions with Khairi have been extremely entertaining while also being educational. Relating to Julian on many issues has given me hope and guidance, especially during the final days of PhD that we were both going through. Similarly, my cousin Hrvoje has been a sort of a mentor during most of my life, unselfishly sharing priceless experiences and suggestions with me. Endless photography-related discussions with Lance, Luc and James have done a great deal to distract and relieve my mind from the PhD work. But undeniably the biggest supporter has been Vita, who stuck with me during my darkest days and had to endure my numerous self-centered moments. Her unconditional encouragement kept pushing me forward at times when I had no desire to continue. Of course, I wouldn't even be here writing this without endless support, on so many levels, from my parents and my dear sister who always believed in me even when I didn't.

This dissertation is based on work supported in part by the National Science Foundation (NSF), awards OCI-0225642 (OptIPuter) and OCI-0943559 (SAGE), Sharp Laboratories of America, and, the King Abdullah University of Science and Technology (KAUST) award number US 2008-107/SA-C0064. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the author and do not necessarily reflect the views of the funding agencies and companies.

RJ

# TABLE OF CONTENTS

<u>CHAPTER</u>	<u>PAGE</u>
<b>1. INTRODUCTION .....</b>	<b>1</b>
1.1 SUMMARY OF CONTRIBUTIONS.....	7
1.2 DOCUMENT STRUCTURE.....	8
<b>2. BACKGROUND .....</b>	<b>9</b>
2.1 LARGE, HIGH-RESOLUTION DISPLAYS.....	9
2.2 SCALABLE ADAPTIVE GRAPHICS ENVIRONMENT (SAGE).....	11
2.3 EARLY EXPERIENCES WITH SAGE .....	14
2.3.1 <i>Desktop Interfaces – Indirect Interaction</i> .....	14
2.3.2 <i>Early Experiences</i> .....	15
2.4 SUMMARY .....	18
<b>3. RELATED WORK .....</b>	<b>20</b>
3.1 ENABLING DIRECT USER INTERACTION ON LARGE, HIGH-RESOLUTION DISPLAYS	20
3.2 SINGLE AND MULTI-USER INTERACTION TECHNIQUES .....	22
3.3 DISPLAY SPACE ORGANIZATION.....	24
3.4 RELATED WORK SUMMARY .....	27
<b>4. MULTI-USER, MULTI-MODAL INTERACTION FRAMEWORK .....</b>	<b>29</b>
4.1 DESIGN GOALS .....	29
4.1.1 <i>Distributed</i> .....	30
4.1.2 <i>Scalable</i> .....	30
4.1.3 <i>Multi-user</i> .....	30
4.1.4 <i>Multi-modal</i> .....	31
4.2 DIRECT INTERACTION MANAGER (DIM).....	31
4.2.1 <i>Device Manager</i> .....	33
4.2.2 <i>Event Manager</i> .....	34
4.2.3 <i>Overlays and Drawing</i> .....	35
4.3 SAGE WIDGETS.....	36
4.3.1 <i>The Anatomy of a Distributed Widget</i> .....	37
4.3.2 <i>Widget Creation</i> .....	38
4.3.3 <i>Widget Types</i> .....	40

## TABLE OF CONTENTS (CONTINUED)

4.3.4	<i>Event Handling Scenario</i> .....	42
4.4	ATTAINING DESIGN ASPECTS .....	43
4.5	SUMMARY .....	45
<b>5.</b>	<b>ITERATIVE DESIGN</b> .....	<b>47</b>
5.1	CYBER-COMMONS – THE SECOND WALL DISPLAY .....	48
5.2	FIRST DESIGN PHASE – DIRECT, MULTI-MODAL INTERACTION .....	49
5.2.1	<i>Physical Interaction Devices – Interaction From a Distance</i> .....	50
5.2.2	<i>SAGE Pointer – Interaction From Personal Laptops</i> .....	52
5.2.3	<i>Summary of Design Improvements</i> .....	54
5.3	OBSERVATIONS .....	55
5.3.1	<i>Classroom – Visual Analytics Class #1, Half-wall</i> .....	55
5.3.2	<i>Lessons</i> .....	60
5.4	SECOND DESIGN PHASE – ENHANCING DISPLAY ORGANIZATION.....	62
5.4.1	<i>Off-loading Information – Minimize</i> .....	62
5.4.2	<i>Maximize</i> .....	63
5.4.3	<i>Tiling</i> .....	64
5.4.4	<i>Multi-window Manipulation</i> .....	64
5.4.5	<i>Push-to-back</i> .....	65
5.4.6	<i>Persistence</i> .....	66
5.4.7	<i>Summary of Design Improvements</i> .....	67
5.5	OBSERVATIONS .....	69
5.5.1	<i>Meetings</i> .....	69
5.5.2	<i>Analysis Task – “Image of Research” Judging</i> .....	71
5.5.3	<i>Lessons</i> .....	75
5.6	THIRD DESIGN PHASE – DISPLAY PARTITIONING AND UP-CLOSE INTERACTION....	76
5.6.1	<i>Touch Interaction</i> .....	76
5.6.2	<i>Display Sections</i> .....	79
5.6.3	<i>Summary of Design Improvements</i> .....	81
5.7	OBSERVATIONS .....	84
5.7.1	<i>Classroom – Visual Analytics Class #2, Full Wall</i> .....	84
5.7.2	<i>Presentation - Art Seminar</i> .....	87
5.8	SUMMARY OF DESIGN ITERATIONS.....	89
5.8.1	<i>Design Needs not Addressed</i> .....	91
<b>6.</b>	<b>EVALUATION - COLLABORATIVE ANALYSIS USER STUDY</b> .....	<b>93</b>
6.1	STUDY METHODOLOGY .....	93
6.1.1	<i>Task Description</i> .....	94
6.1.2	<i>Setup and Equipment</i> .....	95

## TABLE OF CONTENTS (CONTINUED)

6.1.3	<i>Participants</i> .....	95
6.1.4	<i>Procedure and Data Gathering</i> .....	96
6.2	OBSERVATIONS .....	97
6.2.1	<i>Overall Workflow</i> .....	98
6.2.2	<i>Display Space Organization</i> .....	98
6.2.3	<i>Fundamental Organization Techniques</i> .....	104
6.2.4	<i>User Interaction and Control</i> .....	108
6.2.5	<i>Awareness / Implicit Learning</i> .....	112
6.3	DESIGN IMPLICATIONS – FROM USER FEEDBACK .....	113
6.3.1	<i>Improved Feedback and Awareness</i> .....	113
6.3.2	<i>The Learning Curve</i> .....	114
6.3.3	<i>Easier Control Switching</i> .....	114
6.3.4	<i>Annotation</i> .....	114
6.4	DESIGN IMPLICATIONS – FROM OBSERVATIONS .....	115
6.4.1	<i>Individual Workspaces</i> .....	115
6.4.2	<i>Structure + Freedom</i> .....	115
6.4.3	<i>Allow Transient Organization</i> .....	116
6.4.4	<i>Undo</i> .....	116
6.4.5	<i>Assist With Organization</i> .....	117
<b>7.</b>	<b>CONCLUSION AND FUTURE WORK</b> .....	<b>118</b>
7.1	DIGITAL WAR ROOMS .....	118
7.1.1	<i>Physical Interaction Zones and Multi-modal Interaction</i> .....	120
7.1.2	<i>Interaction for Everyone</i> .....	122
7.1.3	<i>Partitioning the Display Space</i> .....	123
7.1.4	<i>Organization Strategies</i> .....	124
7.1.5	<i>Persistence</i> .....	124
7.2	FUTURE DIRECTIONS .....	125
7.3	SUMMARY .....	127
	<b>CITED LITERATURE</b> .....	<b>128</b>
	<b>VITA</b> .....	<b>134</b>

## LIST OF FIGURES

<b><u>FIGURE</u></b>	<b><u>PAGE</u></b>
Figure 1. A war room of a South African design company <i>Flow</i> . War rooms are often used for collaboratively managing large amounts of information using walls covered with paper-based information that serves as tangible representation of cognitive artifacts.....	3
Figure 2. Examples of various wall displays ranging from projector-based to LCD-based and the newest thin-border LCD displays (time is not linear). .....	10
Figure 3. LambdaVision, EVL's first generation wall-sized display, 105 megapixels in resolution. ....	11
Figure 4. Main components of SAGE.....	13
Figure 5. SAGE UI – a desktop-based, cross-platform user interface.....	15
Figure 6. SAGE Web UI – A browser-based user interface. ....	15
Figure 7. During the weekly technical research meetings all students show their laptop screens on the tiled-display to share ideas and promote discussion.....	16
Figure 8. The anatomy class study session where students were asked to answer questions using the displayed images on LambdaVision.....	18
Figure 9. The most representative, highly related work in user interaction and display space organization on wall displays. ....	28
Figure 10. Direct Interaction Manager integrated with SAGE. Device Manager handles multiple physical interaction devices, Event Manager performs event distribution for device events, individual event handlers manage the logic and overlays manage the drawing of objects. The components in gray are all a part of the Direct Interaction Manager.....	32
Figure 11. The three-step process of creating an application widget, a button in this case.....	40
Figure 12. Event handling scenario for a mouse click. Dotted lines denote a network connection. ....	43
Figure 13. Cyber-Commons, EVL's second generation, 20-foot, thin-bezel wall.....	49
Figure 14. The Gyromouse (made by Gyration) is used for enabling interaction from a distance.....	52

## LIST OF FIGURES (CONTINUED)

Figure 15. SAGE Pointer is a simple cross-platform that allows users to control SAGE walls from their laptops, share their desktop, media files and documents.....	53
Figure 16. The first design phase focused on enabling direct interaction. Observations and the resulting framework improvements are highlighted in blue. ....	55
Figure 17. Visual Analytics class taught on the Cyber-commons wall. The left half contained the lecture notes while the right half ran SAGE for juxtaposing media relevant to the lecture. ....	57
Figure 18. Juxtaposing media externalizing students' memory and allows direct comparison.....	58
Figure 19. The media browser, developed using SAGE widgets, is a tool for easily browsing and showing multimedia files on large high-resolution displays. ....	66
Figure 20. The second design phase focused on enhancing the display space organization through easier window manipulation techniques. Observations and the resulting framework improvements are highlighted in blue.....	68
Figure 21. The Image of Research judging process made use of the Cyber-Commons wall for juxtaposing all the submissions on the left while providing the research statement on the right. ....	72
Figure 22. For the lack of explicit support, information was categorized using spatial grouping during the Image of Research judging process.....	74
Figure 23. During the final stage of the Image of Research judging, the committee took advantage of the physical navigation affordances of wall displays and used post-it notes for placing votes. ....	74
Figure 24. Three display sections supporting various organization strategies. The left section is automatically tiled while the rest are manually organized. ....	81
Figure 25. The summary of the three commonly used interaction zones in wall display environments, each with their own benefits and drawbacks. ....	82
Figure 26. The third design phase brought the touch interface as another interaction zone and display sections for space partitioning. Observations and the resulting framework improvements are highlighted in blue. ....	83

## LIST OF FIGURES (CONTINUED)

Figure 27. During the visual analytics class, the Cyber-Commons wall was partitioned into three regions, one for juxtaposing media, one for the Cintiq tablet and one for the lecture notes.....	85
Figure 28. During the art seminar presentation, the wall was split into a working region with all the media and the context region with the presentation slides. ....	88
Figure 29. During the discussion stage, a structured display space quickly turned into a freeform one with many temporary window organizations. ....	89
Figure 30. Digital war rooms are a combination of traditional war rooms and large, high-resolution displays, which provide the benefits of virtual interaction and reconfigurable space.....	90
Figure 31. Sections explicitly defined personal workspaces and allowed participants to employ individual organization strategies. ....	99
Figure 32. Implicitly defined personal workspaces loosely correspond to users' seating positions, require more physical distance for clarity, and necessitate manual window organization.....	100
Figure 33. Typical window layout during filtering stage consisted of the job posting and the supporting information in focus on top of the structured layout underneath. ....	103
Figure 34. To get around the limitation of the automatic tiling algorithm and maximize space usage, users would sometimes group windows of similar aspect ratios into separate sections.....	105
Figure 35. Physical navigation proved more natural and required minimal window manipulations. ....	111
Figure 36. Digital war rooms are a combination of traditional war rooms and large, high-resolution displays, which provide the benefits of virtual interaction and reconfigurable space. The arrows outline how the techniques serve to support the traditional war room affordances. ....	119
Figure 37. The three commonly used interaction zones in wall display environments, each with their own benefits and drawbacks.....	122

## LIST OF ABBREVIATIONS

SAGE	Scalable Adaptive Graphics Environment
EVL	Electronic Visualization Laboratory
LCD	Liquid Crystal Display
SAIL	SAGE Application Interface Library
VNC	Virtual Network Computing
UI	User Interface
SDG	Single Display Groupware
MDG	Multi-Display Groupware
DIM	Direct Interaction Manager
API	Application Programming Interface
XML	Extensible Markup Language
DVI	Digital Visual Interface
VGA	Video Graphics Array
HD	High-definition
PDF	Portable Document Format
IR	Infrared

## SUMMARY

In recent years, we have seen a steady increase in the resolution and the amount of collected data that needs to be analyzed or presented. Often, the complexity of the problems that drive this increase requires that multidisciplinary teams work collaboratively towards a solution. Traditional paper-based war rooms have been shown to double the productivity of radically collocated teams when addressing such problems. Large, high-resolution displays have the potential to transform paper-based war rooms into more dynamic digital war rooms by enabling virtual interaction and reconfigurable space. Although of great potential, these affordances alone will not result in increased productivity since our current knowledge of virtual interaction and display space organization is based on contemporary desktop systems with vastly different requirements, necessitating new techniques tailored for wall displays.

This dissertation first presents a unique interaction framework that allows multi-user, multi-modal control of wall displays along with scalable, distributed user interface widgets for building novel interfaces for collaborative large, high-resolution environments. The framework enabled subsequent investigations of the user interaction and display space organization issues through numerous real-world usage scenarios. The investigations were aimed at shedding light on two specific questions. First, what is a set of interaction requirements and modalities appropriate for collaborative work across a broad range of wall display use cases? Second, what display space organization techniques and behaviors do users employ in collaboratively managing a large amount of information on wall displays? Based on observations from a number of use cases, a

## **SUMMARY (CONTINUED)**

sample set of tools was iteratively developed for addressing the issues of input control and display space organization. Lastly, the developed tools were evaluated in a simulated collaborative analysis task for the purposes of understanding how the developed techniques serve to support traditional war rooms.

# 1. INTRODUCTION

Imagine the process of writing a research proposal. To start off, typically a group of researchers brainstorm their ideas in order to formulate a feasible and worthy research question. This involves generating numerous ideas in various forms: sketches, mindmaps, notes, graphs and tables. Even reviewing existing work involves finding and evaluating potentially hundreds of relevant publications, news articles, pictures and videos. The reoccurring thought process in writing a research proposal is the assimilation of large amounts of heterogeneous information followed by decision-making or rationalizing based on the relationships among the different pieces. Tasks that require dealing with information in such a way are increasingly common across other domains as well. For instance, scientific instruments today record an unprecedented amount of data, which is eventually transformed into numerous visual representations, requiring interpretation by groups of scientists. Even in education there has been an increase in the use of multimedia to support the increasingly complex lectures. Notes are often accompanied by multiple images, videos, websites, and drawings that provide context for students and help clarify concepts. Aside from brainstorming, other areas frequently dealing with large amounts of visual information include design and monitoring tasks, emergency response environments, and various planning activities. Generally, working with data in a visual domain provides a common denominator for communication and analysis across multiple disciplines and heterogeneous data types. However, successfully dealing with so much information is ultimately bounded by our finite cognitive ability, which leaves the advancement in the supporting tools as the only path to improvement.

Often, such information-intensive tasks bring together multiple users into teams for collaboratively solving problems. Traditionally, war rooms have been an environment that fostered group work through radical collocation. The main features of every war room are walls covered with paper artifacts that serve as tangible representations of cognitive artifacts for facilitating social interactions among group participants (Figure 1). Teasley et al. (Teasley et al. 2000) observed doubling of productivity among such radically collocated teams when compared to group members working individually. The increase in productivity was attributed to constant awareness and implicit learning, easy transitions in and out of spontaneous meetings, immediate access to information and its persistence. A similar study also pointed out that spatial memory is frequently used for recalling artifacts in paper-based war rooms (Teasley et al. 2002). Despite their advantages, traditional paper-based war rooms require printing of artifacts for posting on walls, at a time when the majority of information is digital. This severely limits the opportunity for re-arrangement or dynamic update of information as well as retrieval of information for individual work outside the war room. Therefore, the question remains of how to enable digital war rooms while preserving the affordances found responsible for productivity improvements.



Figure 1. A war room of a South African design company *Flow*. War rooms are often used for collaboratively managing large amounts of information using walls covered with paper-based information that serves as tangible representation of cognitive artifacts.

Performing information intensive tasks in desktop environments is possible but often cumbersome, and not optimal. The biggest drawback is the low resolution of the display, which forces us to display only a few pieces of information at a time and resort to frequent context switching. Frequent context switching limits our ability to see the broader picture and make sense of our data. Additionally, larger, more complex problems often require that teams of users work collaboratively towards a solution, which is beyond the capabilities of desktop environments (hence called “personal computers”).

Large high-resolution display environments, also referred to as wall displays, empower users to better cope with this increase in the amount of visual information. The benefits over contemporary desktop environments are realized in the two main physical characteristics of wall displays. The high resolution enables considerably more

information to be displayed simultaneously, which reduces context switching and enables juxtaposing of relevant pieces of information for direct comparison (Ball et al. 2005; Czerwinski et al. 2003). The large size of such displays makes them an excellent collaborative environment, promotes a more natural physical navigation, and improves spatial performance when analyzing high-resolution datasets (Ball et al. 2007). These affordances make them ideal for enabling digital war rooms; however, taking advantage of them does not come for free. As prior research has pointed out (Ball and North 2005; Czerwinski et al. 2006), desktop interaction techniques generally do not scale well to wall displays, limiting their potential in supporting visual exploration. Although areas for improvement are numerous, in this document the focus is on addressing the issues of user interaction and display space organization in collaborative large, high-resolution display environments.

To even begin applying wall displays in real-world environments, we need appropriate software infrastructure that will allow us to take advantage of all the affordances mentioned previously. In itself this is not an easy task since high-resolution is currently achieved by tiling multiple off-the-shelf displays and driving them with a cluster of computers. This eliminated the possibility of running contemporary desktop operating systems to create a seamless display surface since they were originally designed for single-user, single-computer environments. Furthermore, as the OptIPuter project (Smarr et al. 2003) envisioned, scientists are beginning to collaborate in large high-resolution environments serving as visualization instruments that tap into remote storage and computing resources over high-speed networks. Given the decreasing costs of bandwidth, accessing remote resources is more cost effective than maintaining local

storage and computing systems. The first step towards this vision was the Scalable Adaptive Graphics Environment (SAGE) (Jeong et al. 2006) that turns any combination and configuration of displays driven by a cluster of computers into a seamless, fully distributed environment where windows of locally or remotely rendered applications can be moved and resized as if they were on a single local desktop. However, SAGE itself is only a middleware that manages delivery and display of remotely rendered application pixels across tiled-displays and as such is of little value without appropriate user interaction capabilities. Therefore, before investigating interaction and organization paradigms, the interaction system for SAGE is presented, which enables truly multi-user and scalable user interaction with distributed data in large high-resolution display environments, effectively turning the OptIPuter vision into reality. This provided a solid platform for supporting and observing real-world collaborative use cases and iteratively designing interaction and organization techniques derived from the observations.

The first focus is on issues of virtual interaction. Given the appropriate interaction framework, the lowest requirement for usefulness of any new system is the ability to interact with it. On desktop systems, the mouse has been the standard interaction device for almost three decades. While appropriate for such environments, it limits physical navigation by confining users to a hard surface. Given the wide range of potential wall display use cases, it is easy to envision scenarios that necessitate mobility or multi-user interaction, rendering the mouse inadequate. For instance, in an educational setting, professors often walk in front of the class while explaining the material, making the mouse less than ideal for such interaction. Ultimately, before being able to address any further human-factors research questions in wall display environments, appropriate

means for interaction must be provided, especially for investigating use cases outside the controlled laboratory studies. This motivated the exploration of the interaction requirements that various real-world use scenarios pose in the context of wall displays. Among others, the interaction modality criteria guiding the investigation were the number of users needing interaction capabilities, the mobility requirements and the target display size.

The second focus is display space organization. Having the ability to display large amounts of information does not necessarily mean that there will be an instantaneous increase in productivity. Mostly-overlapping windows on desktop displays are becoming mostly-visible windows on wall displays. While mostly-visible windows are certainly desirable, the problem of window switching now gave way to the problem of window organization. Compounded by the fact that a large screen real estate will inevitably entice users to display a large amount of information, we must look into better ways to help users organize this information into meaningful arrangements. However, given the novelty of large high-resolution display environments, there is little knowledge on how users actually make use of this reconfigurable space for dealing with large amounts of visible information. Collaboration and multi-user interaction requirements further add to the complexity of the display space organization by potentially necessitating dynamic task and display partitioning. The aim was therefore to identify most commonly used wall display organization paradigms and propose design guidelines for building more appropriate tools for successfully managing a large amount of visible information in collaborative high-resolution environments.

## **1.1 Summary of Contributions**

Although the interaction framework was designed with remote collaboration in mind, and many findings presented here will carry over to distributed use cases, that aspect is beyond the scope of this document. Instead, the focus is on collocated, single and multi-user work in the context of large, high-resolution display environments. For the purposes of this thesis, displays are considered large if they can comfortably accommodate multiple standing users. This is typically the case in wall configurations where users can move about in front of the display. High resolution refers to displays that have significantly more resolution than a contemporary desktop display, typically over 10 Megapixels. The main contributions of this thesis follow.

First, a novel interaction framework is presented for supporting simultaneous multi-user, multi-modal interaction in large high-resolution environments. Together with SAGE, this makes it a functional instantiation of the OptIPuter model, which is ideal for empowering collaboration in distributed, wall display environments for the purposes of further research in potential applications and human factors issues in a broad range of real-world scenarios, beyond the ones presented in this document.

Second, observations from a broad range of real-world use cases are described, along with the interaction and displays space organization behaviors employed in the use cases. From these observations, interaction paradigms and a sample set of display space organization tools are proposed and iteratively developed for collaboratively managing large amounts of information on wall-sized displays.

Third, the developed tools and techniques are evaluated in a user study of a collaborative analysis task. Evaluation in a simulated war room environment provides

insight into how the developed techniques support traditional war room affordances responsible for increased productivity. Lastly, general guidelines for successfully dealing with large amounts of information on wall displays are summarized.

## **1.2 Document Structure**

Chapter 2 presents an evolution of wall-sized displays and the early experiences as the motivation for this work. Chapter 3 gives an overview of the current research in the areas of user interaction and display space organization in the context of wall displays. The rest of this document is organized along the three main contributions of this work. Chapter 4 describes the multi-user, multi-modal interaction framework for SAGE that enabled further investigation of interaction and display space organization issues in the real-world use cases. The iterative design phase for development and advancement of interaction and organization techniques is presented in Chapter 5. A collaborative analysis user study for evaluating previously described improvements in the freeform, collaborative analysis task context is presented in Chapter 6. Finally, the summary of findings and the general design guidelines derived from the observations are given in Chapter 7 along with the possible future work directions.

## 2. BACKGROUND

This chapter begins by providing a brief overview of the evolution of wall-sized displays, from the low-resolution, projector-driven walls 20 years ago to the latest, high-resolution, thin-bezel LCD tiled-displays. Following is the description of the Scalable Adaptive Graphics Environment, EVL's middleware for turning such displays into a single, contiguous display surface. Lastly, several preliminary use cases of SAGE are presented which motivated the work presented in this document.

### 2.1 Large, High-resolution Displays

As mentioned previously, displays are considered large if they can comfortably accommodate multiple standing users. This is typically the case in wall configurations where users can move about in front of the display. High resolution refers to displays that have significantly more resolution than a contemporary desktop display, typically over 10 Megapixels.

In the past however, achieving both, the large size and high resolution, has been difficult given the technological limitations. Although the high pixel density of LCD displays makes them a preferred method for increasing resolution, the significant costs of early LCD display technology encouraged the adoption of projectors for building wall displays. Tiling projectors also provided the benefit of having a seamless display, although it requires complex alignment procedures. In contrast, early LCD displays typically had very large borders and were rather small in size, yielding a significant border-to-display ratio when tiled (i.e. unusable-to-usable area). As the cost of LCD

displays kept decreasing, the early projector-based tiled displays slowly gave way to large, LCD-based tiled displays with resolutions exceeding 100 Megapixels. Moreover, recent technological advancements significantly reduced the size of display borders, enabling the construction of near-seamless tiled walls effectively eliminating their greatest drawback, although currently at somewhat lower resolutions. It is already apparent that the trend will continue to even smaller border sizes and even higher resolutions and such displays will be commonly found outside the laboratory settings: in offices, classrooms (Leigh and Brown 2008) and even homes (Go Big or Go Home). This further emphasizes the need for the work presented in this dissertation.

The following chart outlines some of the representative wall displays and the evolution from projector-based to LCD-based tiled displays (Figure 2).

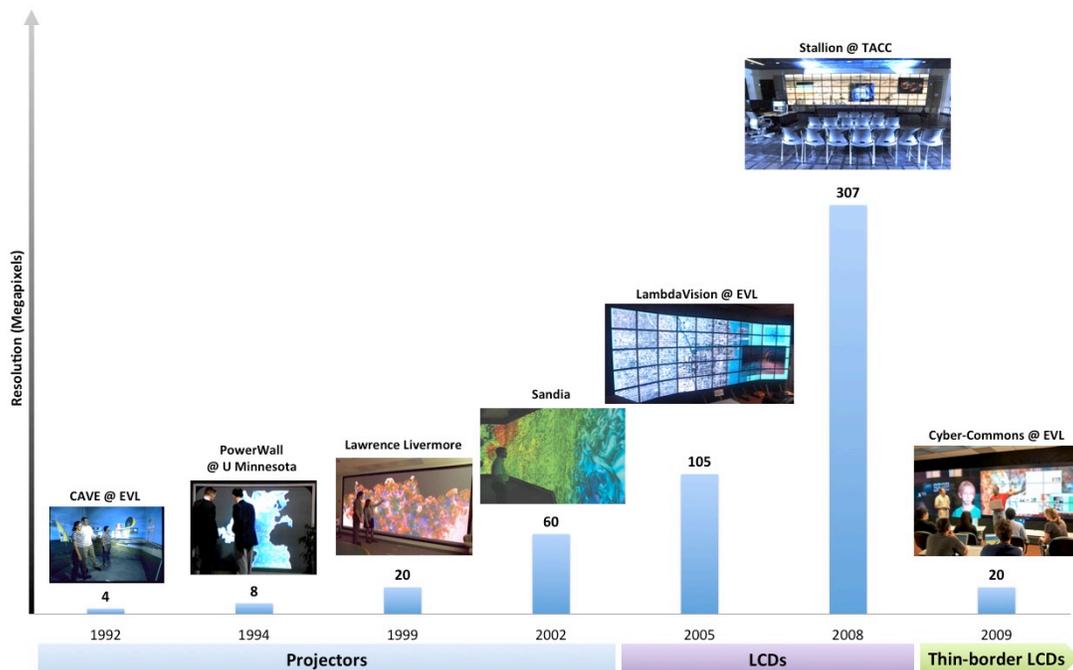


Figure 2. Examples of various wall displays ranging from projector-based to LCD-based and the newest thin-border LCD displays (time is not linear).

LambdaVision is EVL's first iteration of wall-sized, high-resolution displays that motivated the work presented in this document. It was constructed in 2005 from 55 tiled 21-inch LCD monitors for a total resolution of 105 megapixels. The tiling resulted in roughly one-inch bezels between adjacent screens. Physically, the wall was 17 feet wide, 6 feet high and driven by a cluster of 28 computers (Figure 3).



Figure 3. LambdaVision, EVL's first generation wall-sized display, 105 megapixels in resolution.

## **2.2 Scalable Adaptive Graphics Environment (SAGE)**

The first step towards enabling the type of work paradigms envisioned by the OptIPuter project, was the supporting software infrastructure to enable distributed, collaborative work on large high-resolution displays driven by a cluster of computers. Scalable Adaptive Graphics Environment (Jeong et al. 2006) (SAGE) is a middleware that turns any combination and configuration of displays driven by a cluster, into a

seamless, fully distributed environment where windows of locally or remotely rendered applications can be moved and resized as if they were on a single local desktop. The need for a seamless high-resolution environment was motivated by EVL's prior work in the Amplified Collaborative Environments (Park and Cho 2005).

Figure 4 shows the main components of SAGE. SAGE Application Interface Library (SAIL) is responsible for communication between the application and SAGE. Every application gives its rendered pixels to SAIL which then streams them to the appropriate Display Nodes depending on the current size and position of the application window on the display. The transparent separation of display from rendering is achieved because the application doesn't know anything about the size and position of its window on the display (or the location of the display for that matter). An important remark has to be made about window size. When a window on a SAGE-driven display is resized, its contents are scaled as well (as opposed to resizing the "viewable area" which can then be used to display more content, similar to resizing a web browser window).

There is typically one Display Node per computer in a cluster running the display. Each Display Node is responsible for receiving application pixels sent by SAIL and drawing them correctly on the displays physically connected to this node. Multiple applications can be displayed at the same time because each Display Node can simultaneously receive multiple pixel streams.

Finally, the FreeSpace Manager (fsManager) is the central component of SAGE that keeps track of the current display parameters and the arrangement of the application pixels on the display. Based on the requested arrangement, fsManager directs SAIL to split up application's pixels and send each section to the appropriate Display Nodes. The

only exposed fsManager commands for user interaction are moving, resizing and closing application windows, which have to be issued through a network interface. Since SAGE is not bound by the traditional operating system input event processing, it has all the prerequisites to support simultaneous multi-user interaction.

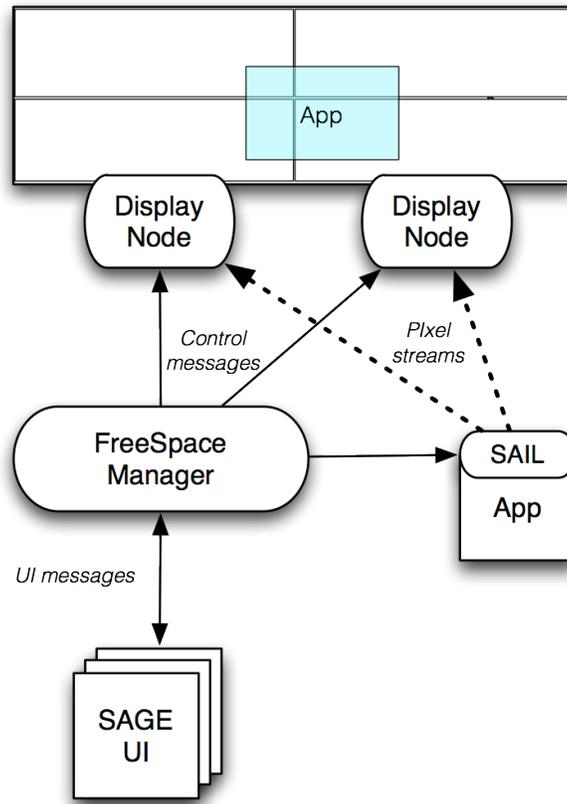


Figure 4. Main components of SAGE.

## **2.3 Early Experiences with SAGE**

### **2.3.1 Desktop Interfaces – Indirect Interaction**

The first interface to SAGE was SAGE UI, a desktop-based interface (Figure 5). It is multi-user, multi-platform and distributed which allows each user to run a copy from their laptop and use the tiled-display simultaneously. This was also the most comprehensive interface with personal desktop sharing using VNC, sharing media files via drag-and-drop, browsing previously shared media, saving and loading sessions, application management and other administrative features. To make the interaction more accessible for new and casual users, a limited web-based version of the interface was also available. SAGE Web UI (Figure 6) supports basic window manipulation and desktop sharing as the only available application. Users can simply point their browser to a web page, pick the SAGE display they want to control and they will receive an overview of that display in the browser where windows can be manipulated (in Figure 5 and Figure 6 a 2x2 display is shown running 2 applications). Interaction through SAGE UI and SAGE Web UI is labeled as *indirect* since windows are manipulated through their representations as opposed to directly on the wall.

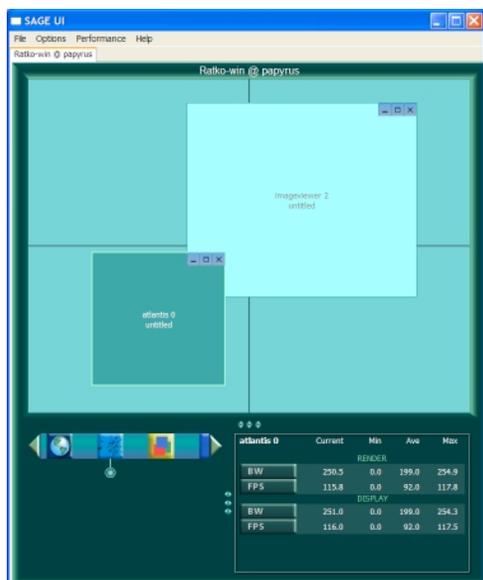


Figure 5. SAGE UI – a desktop-based, cross-platform user interface.



Figure 6. SAGE Web UI – A browser-based user interface.

### 2.3.2 Early Experiences

We started regularly using SAGE during our weekly technical meetings where every participant would simultaneously share their laptop screen with the group by showing it on EVL’s 105-megapixel tiled-display. At the beginning of the meeting, every user would position their small window along the periphery of the display using SAGE UI running on their laptop, typically resulting in approximately 10 windows on the wall (Figure 7). This was by no means required but instead seemed to be a natural layout that we implicitly agreed upon not long after we started using the display in our meetings. This created a large empty space in the middle of the display that is typically used for maximizing the window of the person currently speaking. We noticed that users rarely reposition or resize other people’s windows even though no such restrictions were

imposed by the system. This behavior further supported our hypothesis on the importance of simultaneous multi-user interaction where every user has the ability to control the display. Since all the windows are already present on the display, switching between speakers is only a matter of bringing the correct window into focus by resizing and repositioning it using SAGE UI (as opposed to switching video cables in a more traditional single-projector environment). This proved to be a significant benefit for the dynamics of the meetings as it encouraged discussion and information sharing. During the discussions, there is generally a need to point at the display, which had to be done by either walking up to the display or passing a laser pointer between the participants. This suggested that giving every user a more direct interaction and pointing capability, beyond the desktop-based SAGE UI, is needed.



Figure 7. During the weekly technical research meetings all students show their laptop screens on the tiled-display to share ideas and promote discussion.

Besides the regular meetings, an anatomy class professor at the university wanted to hold an exam study session using the display. Numerous medical images, pertinent to that day's discussion, were preloaded and shown on the display before the beginning of class. Because large bezels made small text unreadable, the study notes were shown on a single large screen neighboring the LambdaVision wall. The students gathered around the wall and the professor would spur the discussion by asking questions that required the students to use the appropriate image on the screen for answering the question. The student that made a first attempt at answering usually ended up walking up to the display to point out the details and explain his/her answer (Figure 8). Meanwhile, the professor had to remotely point at the display in order to guide the student or to raise specific questions. While this was easily accomplished using a laser pointer, manipulation of the windows on the display was still mediated by an operator because not every student (nor the professor) had a laptop during the class to make use of the Web UI or the SAGE UI. This supported the need for the direct interaction without a laptop since one could envision many such use cases where a desktop-based user interface isn't appropriate. Furthermore, that interface had to be intuitive and with a small learning curve to encourage a more casual use of the display by non-expert users.

The manipulations typically involved bringing a window into focus by either maximizing it or manually setting size and position. In more than one case, multiple images had to be used for comparison that demanded side-by-side windows of equal proportions, which could only be arranged by hand. On occasion, multiple windows also had to be positioned lower on the screen to make the top of the window reachable for

pointing. These and similar behaviors indicated the need for more appropriate window manipulation techniques that will assist in juxtaposition of information.



Figure 8. The anatomy class study session where students were asked to answer questions using the displayed images on LambdaVision.

## **2.4 Summary**

Preliminary observations and experiences with SAGE and desktop-based interfaces confirmed the need for more appropriate direct interaction capabilities and display space organization paradigms, beyond the ones typically found in desktop environments. Specifically, there was a need to point and interact with the wall from a distance and while sitting with a laptop. As observed during the weekly meetings, true multi-user interaction can promote discussion and collaboration. Furthermore, while frequent manipulations and juxtapositions of windows called for more powerful display organization techniques, it was not yet clear which techniques are necessary and how they should be developed in the context of a collaborative environment. Successfully

investigating and addressing such issues first demanded direct interaction capabilities, which motivated the development of the interaction framework presented in this document. The interaction framework then enabled further studies of wall displays in the real-world use cases for investigating and enhancing interaction and display space organization.

### 3. RELATED WORK

This chapter presents the current state of the knowledge on collaborative work in large, high-resolution displays. First, various systems for supporting concurrent, collocated, collaborative work are surveyed. Numerous proposed single-user and multi-user interaction techniques are reviewed in the following section. Lastly, current advancements in the display space organization are presented in the last section. The chapter concludes with the summary of the most representative prior work and paves the way for the work presented in this document.

#### **3.1 Enabling Direct User Interaction on Large, High-resolution Displays**

High-resolution displays are becoming more widespread in scientific laboratories (Stolk and Wielinga 2006; DeFanti et al. 2009) and will eventually become pervasive at the office, in the cubicle, the meeting room (Leigh and Brown 2008), and ultimately at home (Go Big or Go Home). The OptIPuter project (Smarr et al. 2003) explored the use of high-resolution display environments for scientific discovery under the prediction that seemingly unlimited amount of network bandwidth delivered by progress in optical networking would change the way science is conducted between geographically distributed teams. However, the hardware is only half of the equation and it requires a matching software infrastructure and appropriate user interaction techniques to fully take the advantage of the bleeding-edge technology affordances in a real-world setting. Although several real-world applications of large displays have been demonstrated (Fitzmaurice et al. 2005; Buxton et al. 2000; Richards and Mantey 2006; Olsen et al.

2008), they are highly specific to the application domain and often without true multi-user interaction capabilities.

The ubiquitous computing community on the other hand, has put more focus on distributed systems. Typically, that line of work involved integrating multiple displays and devices into a unified environment where users can interact with the system (Roman et al. 2002; Tandler 2001). However, since such environments are still desktop-oriented, they often impose the limitation that there can be only one active user per display or application (Johanson and Fox 2002). While conceptually our goals were similar, our target environment was a seamless large high-resolution desktop where remotely rendered applications are not displayed on a single screen but rather are freely movable and resizable across any number of displays. Additionally, we must enable simultaneous multi-user interaction with all the applications on the display.

Numerous widget frameworks already exist for desktop environments, such as Qt (Qt) or wxWidgets (wxWidgets). However, these are designed for desktop operating systems and therefore inherit their limitations. For example, multi-user interaction typically isn't possible and they are often tightly integrated with the operating system's device events (mouse and keyboard). This low-level coupling with the existing single-user operating systems would make it difficult to adapt the toolkits to work with multiple users simultaneously (i.e. interaction devices). Though toolkits built on top of X Window System are distributed, their typical single client - single server setup doesn't translate to an arbitrary number of clients and servers (e.g. a parallel application being displayed remotely on a tiled-display driven by a cluster of computers). Lastly, they aren't scalable and are therefore inappropriate for displaying on very high-resolution displays where

visibility and usability becomes an issue. For instance, a 100-megapixel display with a touch screen input device requires different widget parameters than a 10-megapixel display with a traditional mouse.

We found that the area of concurrent multi-user interaction in distributed, large high-resolution environments is largely unexplored, even though recent trends identify the need for it. However, in order to make advances in this area, appropriate software infrastructure has to exist that will support the type of interaction scenarios as envisioned by the OptIPuter project. The software infrastructure presented in this document, specifically the interaction framework, is currently the only one supporting the OptIPuter work paradigms and it was necessary for further investigation of interaction and display space organization issues in the real-world environments.

### **3.2 Single and Multi-user Interaction Techniques**

There is a large body of research that highlighted the positive impacts that large tiled LCD displays have on human performance. Large displays offer a wider field of view, which in turn improves spatial performance while the high resolution affords more content to be displayed simultaneously, which reduces context switching and promotes physical navigation (Ball and North 2008; Ball et al. 2007; Ball et al. 2005; Booker et al. 2007; Czerwinski et al. 2003; Shupp et al. 2006; Tan et al. 2003; Tan et al. 2006; Yost et al. 2007). Previous research has pointed out that traditional interaction methods do not scale well to large high-resolution environments and that new, more appropriate methods are needed (Ball and North 2005; Czerwinski et al. 2006). For example, there has been significant research effort aimed towards addressing the issue of target acquisition

(Baudisch et al. 2004; Bezerianos and Balakrishnan 2005; Blanch et al. 2004; Grossman and Balakrishnan 2005; Malik et al. 2005; Szvedo et al. 2006; Forlines et al. 2006; Tse et al. 2007; Vogel and Balakrishnan 2005), cursor tracking (Baudisch et al. 2003; Khan et al. 2005) and various interaction devices (Malik et al. 2005; Cao and Balakrishnan 2003; Jiang et al. 2006; McCallum and Irani 2009). However, most of these techniques were developed with single users in mind and focus on the low-level aspects of user interaction. The question also remains whether and how these techniques work outside the controlled study settings with real-world users and applications that may pose different requirements.

Some prior work has investigated the issue of control in environments supporting simultaneous multi-user interaction. Birnholtz et al. (Birnholtz et al. 2007) compared group performance in a negotiation task under two distinct input configurations: single mouse and multiple mouse (one per user). The results of their study suggest that giving control to every user allows for more parallel work, however the quality of discussion was higher with only one mouse per group. Foster et al. (Foster et al. 2010) report similar findings with gesture-based controllers. We were, however, interested in how the group might coordinate their input control and choose interaction modes on their own given a persistent, multi-user, multi-modal interaction capability. As Vogt et al. (Vogt et al. 2004) pointed out, every device has different collaborative factors as well as physical characteristics, which we were aiming to explore in various usage scenarios. Wallace et al. (Wallace et al. 2009) examined collaborative work in single-display (SDG) and multi-display groupware (MDG) environments. They reported that SDG configurations provided more awareness but also more distraction and interference from other

participants concurrently working on a single large display. On the other hand, MDG configurations afforded less awareness but provided personal space on users' laptops, which resulted in lower cognitive load caused by distractions.

Collaborative work has also been studied in the context of tabletops. Isenberg et al. (Isenberg et al. 2010) identified eight collaboration styles that define how closely users work together while problem solving, which in turn impacts the group's success in the task. Generally, they found that closer collaboration led to improved task performance. Morris et al. (Morris et al. 2010) presented and evaluated WeSearch, a collaborative web search and sensemaking tool for tabletop displays. The key issues that users reported had to do with clutter reduction, virtual keyboard usability and document orientation while support for awareness, sensemaking, and division of labor was positively received. Although tabletops are naturally a collaborative environment, they have different affordances from wall displays. Their display space is still rather limited which requires specific clutter reduction strategies, while various orientations of information result in different group dynamics.

### **3.3 Display Space Organization**

Aside from control, another focus of this work was display space organization (also called window management or window layout). Various organization techniques have been proposed for easier management of an increasing number of windows. For instance, (Robertson et al. 1998) presented a system called "Data Mountain" that allows users to place representations of documents in a 3D environment. They also showed that spatial memory extends to this 3D environment. Besides the typical manual organization

of windows, various attempts have been made at automating the process to some extent. Robertson et al. (Robertson et al. 2004) introduced a technique called “Scalable Fabric” that attempts to make use of a focus plus context concept by automatically shrinking windows as they are moved away from the center of the display. Another approach for managing multiple applications was taken by (Kandogan and Shneiderman 1997). The authors introduced organization techniques that attempt to automate tiling, grouping and hierarchical organization of documents based on user’s roles (e.g. teacher, researcher, personal). (Bell and Feiner 2000) illustrated a system for automatically managing rectangular empty space that could be used for positioning and sizing windows or user interface elements. An automated approach based on purely spatial constraints was demonstrated by (Badros et al. 2001) where users could define spatial relationships between windows and easily invoke the layout algorithm that will obey those relationships (e.g. “keep one window always to the left or above another”).

To improve automated information presentation using knowledge about the data itself (content-awareness), (Zhou and Feiner 1996) suggested a data categorization process. They identified certain categories that any type of data should fit in (e.g. domain of data, attributes, relations, roles, data sense). The concept was demonstrated by displaying a patient’s medical record that consisted of heterogeneous pieces of information. An even more data-specific approach has been taken by (Ishak 2007). The author presented automated user interaction techniques that are content-aware as opposed to only context-aware as in previous research. Content-aware approach considers the actual content, the region of interest in the application and the empty space in order to make better decisions on layout, scrolling and overlapping windows. However, the

authors only considered text-based data and used string-search as the only action for content-aware layout.

All of the approaches listed above pertain to desktop environments where only a limited amount of information is visible. On large high-resolution displays, the problem of display space organization becomes even more difficult primarily because of the significant increase in the amount of information and the large size. This makes manual organization tedious, including manually defining constraints for the system to conform to. Hutchings et al. (Hutchings and Stasko 2004) studied display space management on single vs. multi-monitor desktops and found out that in multi-monitor cases users resort to different context-switching methods, more display space is wasted, windows were generally larger and found out that ultimately, the visibility of a window is a better measure of importance than just activity. Users were broadly classified as maximizers, near-maximizers and careful coordinators, along with several window management techniques. They also noticed that overlapping windows are a rare occurrence, which is somewhat contradictory to the previous study (Jakobsen and Hornbaek 2010), indicating a change in display space organization approaches as screens get larger. To address the problem of manual content organization on large high-resolution displays, (Bezerianos 2007) proposed various techniques that focus on reaching distant windows (i.e. an extension of a target acquisition problem). Bi et al. (Bi and Balakrishnan 2009) expanded on this topic by investigating display space management in the context of large high-resolution displays when used for daily work on a Windows-based system. They observed more manual window management operations and less maximizing and minimizing. Furthermore, there was a tendency to organize the display into focal and

peripheral regions. A similar study used more complex intelligence analysis tasks to examine the utility of large high-resolution displays when compared to a single 17" desktop monitor (Andrews et al. 2010). The study showed how large display space serves to support external memory as well as provide meaning through window size and position on the display. For instance, overlapping windows were considered potentially interesting while tiled (non-overlapping) windows were considered highly important.

### **3.4 Related Work Summary**

Prior work has highlighted the importance of display space organization and user interaction in the environments with ever-increasing size and resolution. However, the focus has either been on multi-user low-resolution environments or single-user high-resolution environments. The true benefit of wall-sized displays lies in supporting collaborative work at the intersection of high-resolution and large size. The work presented in this document aims to investigate display space organization and user interaction in large, high-resolution, multi-user, multi-modal environments.

Figure 9 lists the most representative, highly related work in the area of user interaction and display space organization on wall displays. Primarily, previous work has addressed interaction needs for single-user environments including target acquisition challenges and novel input devices. Additionally, studies have pointed out the need for encouraging physical while enabling virtual interaction as it is more natural and provides benefits for spatial tasks.

No studies thus far have investigated collaborative use of large, high-resolution display space. However, single-user studies have shown a rapid increase in the number of

window manipulations and careful window arrangements. Given the single-user tasks, the center of the wall was always classified as the focal region while the periphery was the context. Lastly, a dramatic increase in the amount of available display real estate prompted the use of space for categorization of information and for supporting external memory.

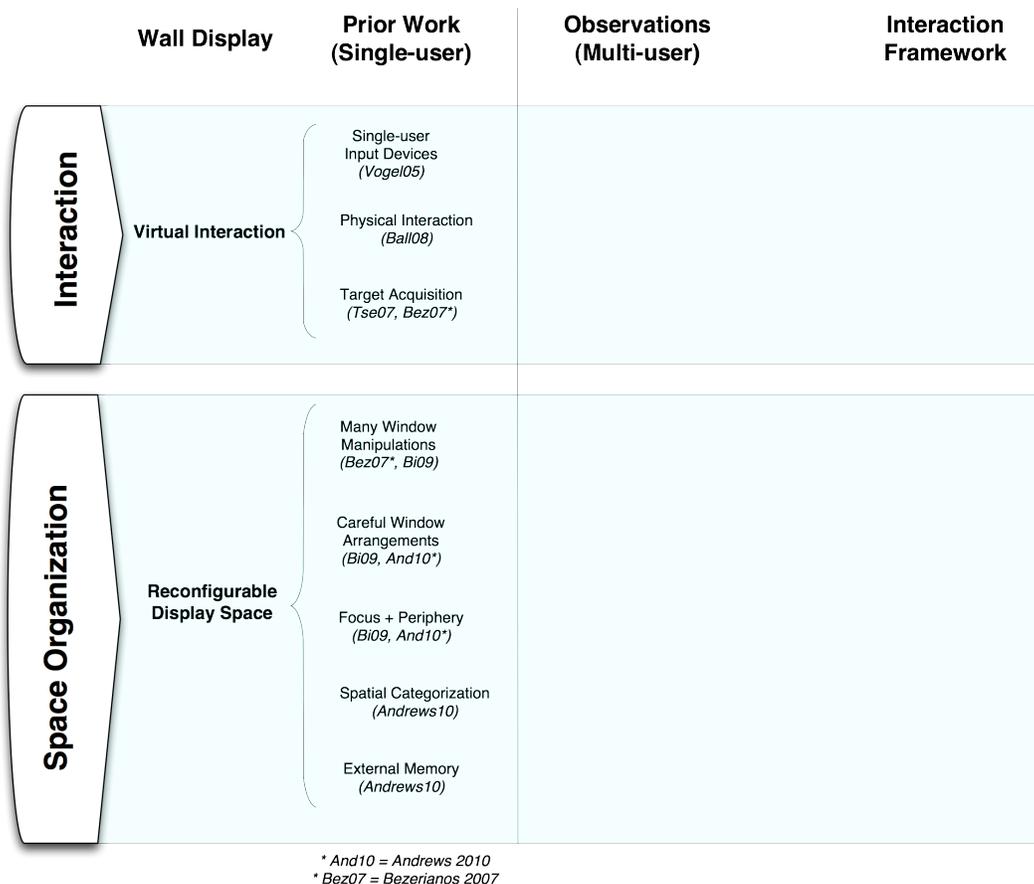


Figure 9. The most representative, highly related work in user interaction and display space organization on wall displays.

## **4. MULTI-USER, MULTI-MODAL INTERACTION**

### **FRAMEWORK**

This chapter presents an interaction framework built on top of SAGE that allows multiple users to simultaneously interact with local or remote media and applications, using a variety of physical interaction devices and scalable, distributed user interface widgets. Besides providing a more intuitive interaction modality for experienced and casual users of wall displays, having such framework enabled further study of high-resolution environments in real-world scenarios, uncovering novel work paradigms and development of more appropriate interaction and display space organization techniques. This would not be possible using existing interaction systems since they do not fully exploit the affordances of high-resolution environments (e.g. high-resolution, multi-user interaction, distributed nature).

First, four major design goals are presented resulting from the preliminary experiences and from the need to preserve the primary benefits of traditional war rooms. Second, the core component of the framework is presented as the Direct Interaction Manager (DIM). Building on top of DIM, SAGE Widgets are described as the widget toolkit for developing distributed, scalable and multi-user wall display user interfaces.

#### **4.1 Design Goals**

Based on the war room affordances and the preliminary experiences with SAGE and the LambdaVision, the following fundamental design aspects of the direct interaction framework are identified. They are believed to be necessary in order to fully exploit the affordances of the large high-resolution displays.

#### **4.1.1 Distributed**

The framework should allow for display of user interface elements across any number of displays driven by any number of machines. Additionally, application interfaces should be visible on any remote display without the application actually being aware of this rendering-display separation. Similarly, any physical interaction device should be able to control any remote application without being concerned with its rendering location.

#### **4.1.2 Scalable**

Since large high-resolutions displays are currently assembled as tiled LCD panels or projectors, their size and resolution can vary significantly depending on the target application. Therefore, it is imperative that the interaction system automatically adapts to the target display size and resolution. This adaptation is necessary from both, the visibility and usability perspective. For instance, the physical interaction devices should adapt their speed and sensitivity (control-display gain) and the interactive objects should adjust their size to be visible and be easily target by the devices.

#### **4.1.3 Multi-user**

Large high-resolution displays are well suited for collaborative work since their large size easily accommodates multiple users and their high-resolution allows much more content to be displayed simultaneously. As initially observed, giving control to every user simultaneously can promote discussion and collaboration. Additionally, one of

the primary benefits of traditional war rooms is the ability to join and leave discussions with ease, which wouldn't be possible if users had to resort to turn-taking during participation. Therefore, to take advantage of these affordances, it will be necessary to support concurrent multi-user interaction, without turn-taking. This interaction could be on the same display or even within the same application.

#### **4.1.4 Multi-modal**

Even though the mouse is the de-facto standard for desktop systems, often it is inappropriate for large high resolution displays because it confines the user to a surface. The research community has introduced many novel and promising interaction devices however, no single device has emerged as the clear winner for the large high-resolution displays. Therefore, it is beneficial to have an easy way of integrating new devices into the system without requiring changes to the applications or the display interface. Depending on the display size, resolution and room configuration, some devices may be more appropriate than others. Additionally, different use cases will pose different interaction requirements based on the number of users and the location of users in front of the wall. Already in the preliminary observations the need for two interaction zones has been identified: from a laptop while sitting and from a distance.

## **4.2 Direct Interaction Manager (DIM)**

DIM is the core of the interaction system that contains the following components: Device Manager to manage physical interaction devices, Event Manager for delivering device events to appropriate interactive objects and the Overlay Manager which manages

the drawing of objects on the display. Each component is described in more detail below and their integration into the existing SAGE architecture is depicted in Figure 10.

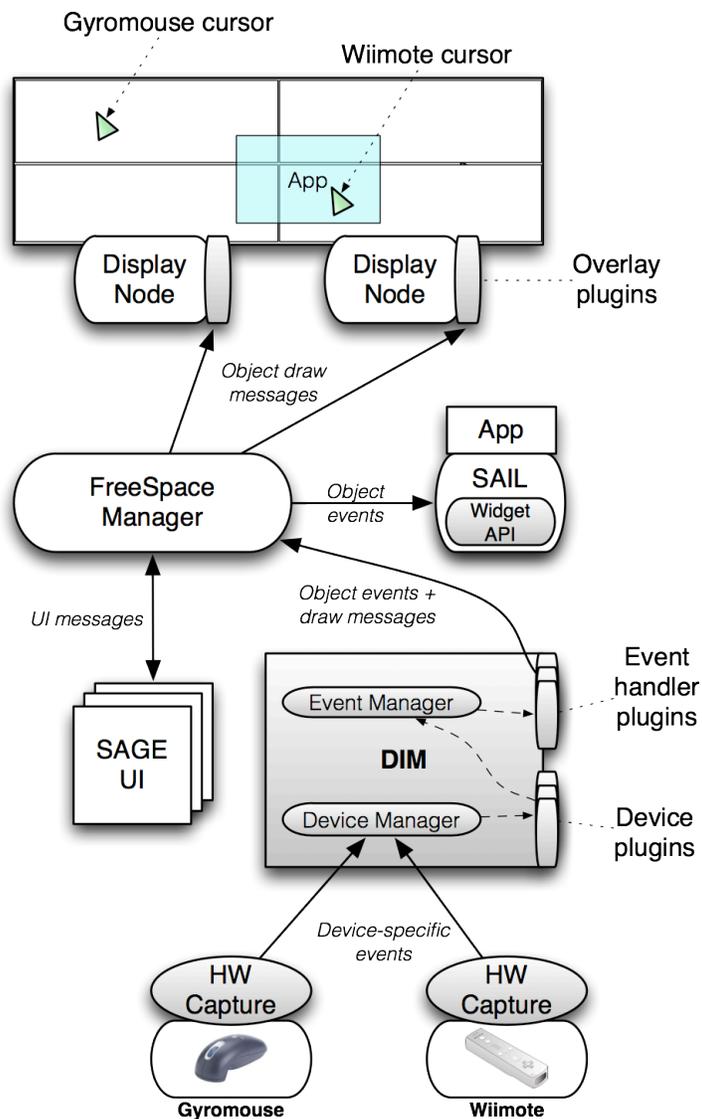


Figure 10. Direct Interaction Manager integrated with SAGE. Device Manager handles multiple physical interaction devices, Event Manager performs event distribution for device events, individual event handlers manage the logic and overlays manage the drawing of objects. The components in gray are all a part of the Direct Interaction Manager.

### 4.2.1 Device Manager

To successfully enable multi-user interaction, the Device Manager accepts events from multiple physical interaction devices through a network interface. If multiple devices are used in a system, especially if they are of different types, it is often the case that they are connected to multiple computers receiving their events (e.g. a special machine for the camera tracker), which necessitates network-based communication. A simple Hardware Capture (HWcapture) component contains plugins for each device that captures the device-specific events, which are then delivered to Device Manager (Figure 10). The event delivery (messaging) library is already provided so developers only need to create the plugin to capture device-specific events. To provide robust service, Device Manager will dynamically add new devices and remove old ones if they fail or disconnect. Allowing devices to connect or disconnect at any point in time, essentially allows users to freely join and leave the space as they please, which mimics traditional war room environments.

So far no single device has emerged as the best tool for large high-resolution display environments even though studies have been performed using laser pointers, hand gestures, touch screens and traditional mice. Therefore, we found it necessary to allow the use of a wide variety of devices even though they may generate different events. When these device-specific events are delivered to the Device Manager, they are passed onto a device plugin that describes the conversion of device-specific events into a generic set of events that are then further used within the system. This effectively makes all the devices appear equal in view of the interaction system and therefore allows new types of devices to be added without changing the user interface objects or applications

themselves. In order to add support for a new type of device, the following two steps are necessary:

- Create a HWcapture plugin that grabs device-specific events from the physical device (e.g. touches from a touch screen)
- Create a device plugin to DIM that describes the conversion between device-specific and generic events (e.g. a touch to EVT\_CLICK)

#### **4.2.2 Event Manager**

As mentioned previously, all device-specific events are converted to a generic set of events, which in turn allows heterogeneous devices to interact with the display equally. Naturally, some devices are more powerful than others, so they can generate a larger subset of the generic events. These generic events are then put in an event queue where the Event Manager processes them in order, trying to find an appropriate event handler for each. If an event handler is found the event is passed onto the handler, otherwise the event is discarded. Event handler is the base class for any interactive object that wants to receive events. Essentially, it is a rectangular container that has size, position and depth parameters, knows how to receive events and it contains other properties common to all widgets (e.g. tooltips, labels, visibility flags). There is also an event handler for every application, which will deliver interaction events to the actual application, whether it is local or remote (for instance, a dragging event can be delivered to a map application). Since the applications can be freely resized and repositioned on the display, all the event coordinates are normalized before being sent to the applications, as they are unaware of their own window position and size. While an event handler is processing an event, the

Event Manager will lock it in order to prevent competition from different devices (for instance, if a button is held down or something is being dragged). However, no such restrictions are imposed between different event handlers, which allows for true multi-user interaction.

### **4.2.3 Overlays and Drawing**

Overlay refers to any visible object on the display. The label “overlay” is used because Display Nodes themselves draw only application pixels on the display so any other object is actually an overlay drawn on top of the application pixels. If an event handler is drawn on the display, it is also an overlay. On the other hand, objects such as cursors are just overlays and not event handlers because they do not react to any events.

There are two components to every overlay. One on the DIM side, which simply retains the current state of the drawn object and sends drawing commands to the Display Nodes. The second component is the overlay plugin on the Display Nodes that describes how the overlay should be drawn in certain visual states (e.g. button is down or up) based on the drawing commands. Each plugin is a subclass of a base overlay class, which contains basic draw parameters (depth-ordering, transparency, size, position, visibility etc.) and contains a generic draw method that each object has to implement. Draw methods provided by plugins on the Displays Nodes are actually just snippets of OpenGL code, which are called at the appropriate time during the draw loop.

### **4.3 SAGE Widgets**

With the introduction of DIM, we had a framework for integrating multiple devices and functional, though very limited, direct interface to SAGE. It replicated the basic window manipulation capabilities of the old desktop-based SAGE UI and SAGE Web UI (move and resize windows), except that interaction was now *direct* since users would have a pointer on the wall that directly manipulates the windows as opposed to their representations. However, there was no way to create a more complete interface either for the wall or for applications (e.g. buttons). Consequently, we developed SAGE Widgets which allows all applications, and the display itself, to use the same set of widgets without dealing with the event handling, the drawing, the interaction devices being used or the distributed nature of the environment. It is important to note that users are not able to edit information in SAGE, only display it (e.g. editing a text document on the wall was not feasible). SAGE Widgets were never conceived as a functionally complete set of widgets for building complex application interfaces, but rather as a test bed for exploring further human-factors issues that traditional widget toolkits would not allow.

One may argue that SAGE application developers will be hesitant to redesign their applications to use these newly developed SAGE widgets. While this may be true in some cases, it is also important to note that typically, current user interfaces were developed for desktop systems and therefore aren't usable on a much larger and higher resolution displays. This is primarily because they rely on the accuracy of the mouse to acquire smaller targets. However, the mouse isn't appropriate anymore for these new environments since it confines users to a hard surface whereas users often want to be

mobile in front of the display. Additionally, the applications themselves are often custom developed for the large high-resolution displays, again because current desktop applications rarely take advantage of the vastly increased display resolution available to them. Therefore, these custom applications are often still in development when the cost of adding a more suitable user interface is not tremendous.

#### 4.3.1 The Anatomy of a Distributed Widget

The widget framework was designed to be compatible with the existing DIM architecture where each widget is actually a collection of several distributed elements.

- An *event handler plugin* to DIM. This component contains the logic, the API for creating this widget and all the parameters of the widget. It also listens for device events and decides how to act on each.
- An *overlay plugin* to Display Nodes. This component contains the drawing code for the widget for all possible states (e.g. how to draw a button when it's pressed or not)
- *Widget stubs plugin* in SAIL. This small component allows the application to create the widget through a simple API (same API as the event handler plugin). It also keeps track of and initiates widget event callbacks that the application is interested in.

The Event Manager then ensures the proper routing of events from devices to widgets (event handler) and applications, and the Overlay Manager handles the routing of drawing commands from the widget to the proper overlay plugin on the Display Node that actually draws the widget. This plugin-oriented architecture ensures we can easily

add new widget types to the system. Therefore, in order to create a new type of widget, one has to write the three plugins described above. While it may appear complicated, the process is in fact the same process as writing a custom widget in any of the desktop-based widget toolkits. The only difference is that the code for the widget is split across three different locations as opposed to only one. Nevertheless, this has certain benefits. For instance, we can change the logic or drawing of widgets without recompiling applications, as long as the API stays the same.

### 4.3.2 Widget Creation

Even though all the widgets, events and devices are managed by DIM, widgets can be created by DIM itself or by the application through a simple API that is similar to desktop-based wxWidgets framework. The process for creating widgets is the same for DIM and applications. Figure 11 illustrates a case where an application creates a widget in three steps:

- A widget is created and its parameters (e.g. size, position, label) are set using the API, which then creates an XML description of the widget. This XML description represents the widget in the network communication between DIM, the application and the Display Node. When a widget is created by the application, the widget stubs plugin exposed in SAIL only generates appropriate XML file and executes event callbacks as the events are received from DIM.
- SAIL then sends the widget XML description to the FreeSpace Manager, which distributes it to every Display Node and also to DIM.

- Every Display Node creates an overlay object from a plugin of the correct type based on the XML description. This overlay simply describes how the widget should be drawn in different visual states (e.g. button down). DIM, on the other hand, creates an event handler from the event handler plugin of the correct type based on the XML description. This event handler performs all of the widget's event handling and functionality. Entities that create the widgets (DIM or application) can also register callbacks to receive widget events when they take place (e.g. button click, menu selection). These events will be automatically delivered to the appropriate event callback even if the widget was created by an application running on a distant rendering cluster. A more detailed description of an event-handling scenario is given in the next section.

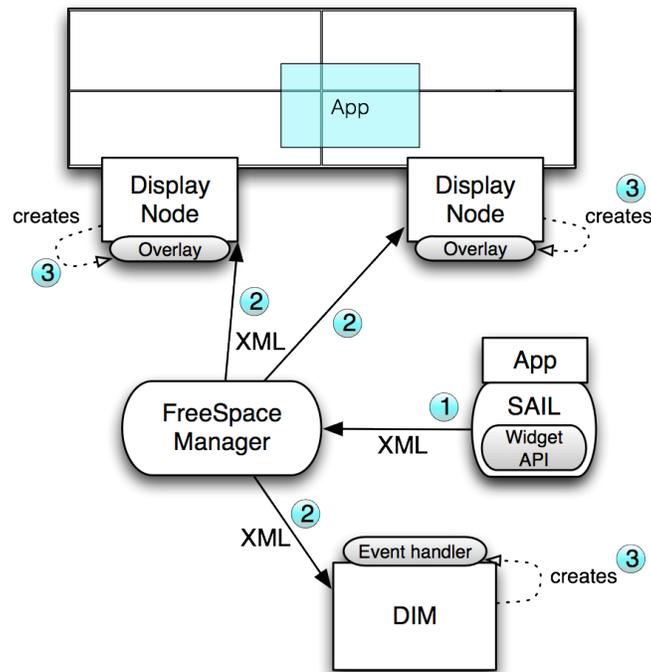


Figure 11. The three-step process of creating an application widget, a button in this case.

### 4.3.3 Widget Types

Although there is only one set of plugins for each widget, they could be created as one of the three different types that ultimately determine how they are resized and positioned on the wall and their priority during the event handling.

**Global widgets** – created by DIM, one set of those exists for each display environment and make up the main display user interface. Examples would be some icons on the desktop or a main menu for starting applications. Global widgets are

positioned relative to the display bounds and are resized appropriately for the current display size.

**Per-application widgets** – created by DIM, one set of those exists for every application on the display. These are always positioned relative to the application (they follow them around). An example would be the borders and the title bar of every application window that is provided by the system itself. In SAGE, this functionality is provided by an event handler that is automatically created for every application. This widget is always the size of the application pixels as drawn by SAGE and positioned so that it exactly covers them. It is completely transparent except for the borders. Then, by receiving events from devices, we are able to manipulate the application pixels as if it were an actual window in a window manager (e.g. EVT\_DRAG that is delivered to this application window widget, will move the window around on the display). The size of the per-application widgets can be relative to the display bounds, as in global widgets, or relative to the application itself, as in application widgets.

**Application widgets** – created by the application. This is the most common way of using widgets in modern widget toolkits. The application developer uses these widgets for creating a user interface for the application. These widgets are always positioned and resized relative to the application itself in order to keep their interface consistent with the designer's intentions. For instance, movie player controls are an example of application widgets.

#### 4.3.4 Event Handling Scenario

Figure 12 shows the propagation and the event handling for a hardware event, in this case a mouse click. First, a mouse click occurs on some remote machine and a hardware specific event is generated. The hardware specific event is sent over the network to DIM where the Device Manager loads an appropriate device plugin for this particular interaction device, a mouse in this case. The plugin knows how to convert the hardware specific event to a generic event, `EVT_CLICK`. The Event Manager in DIM then attempts to find the event handler that is listening for this type of event. In this case, a button event handler plugin is found at the current event position and the event is passed onto it. The event handler determines that the click event altered the button state (into a down state) and the appropriate `BUTTON_DOWN` widget event is generated. This `BUTTON_DOWN` event is now sent to the Display Node to update the button appearance using the button overlay plugin and to the application to execute the callback that was originally set for this button.

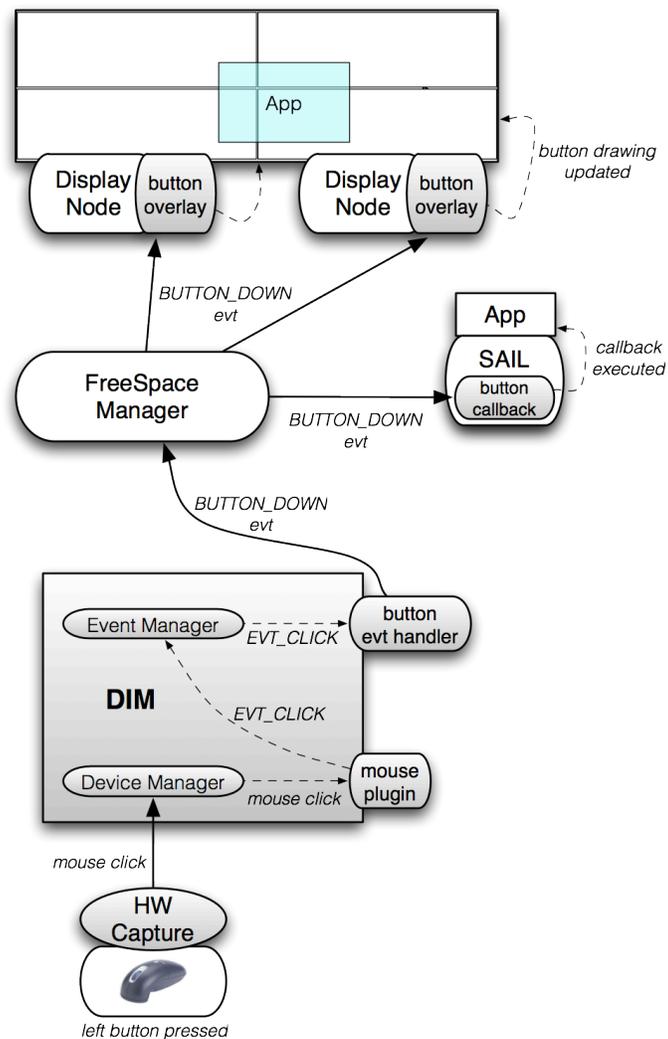


Figure 12. Event handling scenario for a mouse click. Dotted lines denote a network connection.

#### 4.4 Attaining Design Aspects

##### Distributed

By decoupling the widget logic, drawing and creation, we achieved a truly distributed widget framework. Widgets can be seamlessly drawn across any number of

displays driven by any number of Display Nodes. Applications located on distant rendering resources can present their user interface on any display by requesting widgets and receiving events over the network all without ever being aware of this separation. The separation means that we can develop new interaction techniques that are more appropriate for the large high-resolution displays without requiring any modifications to the applications themselves. For example, as display resolution dramatically increases, target acquisition becomes much more difficult. One approach is to dynamically resize targets as cursors get near, which is something we were able to easily implement without changing or recompiling any of the applications.

### **Scalable**

During startup, DIM first collects information about the display environment and calculates the appropriate widget scale factor to adjust for usability (ease of clicking on a target with a physical interaction device) and visibility (font size visible on the display). To provide truly scalable drawing, we use dynamically resizable vector fonts and large raster images that are typically scaled down to reduce aliasing. It would be straightforward to add support for vector images as well. Since we always assume that applications are rendered remotely, we cannot assume that every image used in the application interface is already present on the display side. Therefore, the API automatically embeds necessary images in the XML widget description, which is then delivered to the Display Nodes for drawing. This gives application developers freedom to design custom interfaces.

### **Multi-user**

This design aspect was rather easy to achieve since we designed DIM to support multiple interaction devices while giving each device a separate cursor. Additionally, while rules exist to prevent multiple devices from interacting with one widget at the same time, no such rules are enforced between widgets; hence multiple users can interact with different widgets simultaneously, even within the same application.

### **Multi-modal**

As mentioned earlier, it is possible that new physical interaction devices will surface that perform better on large high-resolution displays or as we have personally experienced, that multiple input modalities may be more intuitive for different use cases. New devices can be added to SAGE by simply providing a plugin that describes the conversion between device-specific events and the generic set of events. By using this generic set of events and therefore removing any specifics of each device, we can treat all of them equally for event handling purposes.

## **4.5 Summary**

The plugin-based architecture of DIM and SAGE Widgets is ideal for adding and tweaking overlays and devices given how little is known about wall display usage in the real-world scenarios. Decoupling the event handling logic, drawing and creation, achieved a truly distributed interaction framework. Widgets can be seamlessly drawn across any number of displays driven by any number of display nodes. Widgets will automatically scale to any target display size and resolution. Furthermore, since there is no theoretical limit on the number of connected physical interaction devices, concurrent multi-user interaction is natively supported by providing a separate cursor for each

device. While rules exist to prevent multiple devices from interacting with the same widget simultaneously, no such rules are enforced between widgets; hence multiple users can interact with different widgets simultaneously, even within the same application. Together with SAGE, the interaction framework makes a functional instantiation of the OptIPuter model, which is ideal for empowering collaboration on wall displays for the purposes of further research of potential applications and human factor issues in digital war rooms. This would not be possible using existing interaction systems since they do not fully exploit the affordances of such displays.

## 5. ITERATIVE DESIGN

Given the interaction framework for building wall display interfaces, the question remains of how to actually design the multi-user interaction and display space organization techniques for managing numerous pieces of information. As mentioned previously, desktop techniques do not always scale to large sizes and high resolutions. Aside from simply developing more appropriate techniques, the fundamental goal is to design digital war rooms that take advantage of wall display affordances while complementing the traditional war room affordances for preserving the productivity benefits of radical collocation.

This chapter presents an iterative design approach consisting of a series of real-world use case observations within EVL, the resulting lessons from the observed behaviors and the design and development of the techniques supporting the observed behaviors. Three main design phases are presented. First design phase resulted in the initial direct interaction wall interface and several physical interaction devices for the two interaction zones as identified in the early experiences with SAGE. The second design phase focused on enhancing the display space organization through a series of manual and automatic window manipulation techniques. The last phase introduced a third interaction zone and the consequent touch interaction along with the display partitioning techniques for supporting several observed uses of display space. The iterative design process was accelerated by EVL's second-generation wall, the thin-bezel tiled display, providing a near-seamless environment appropriate for displaying text and therefore enabling a number of previously impractical use cases.

## **5.1 Cyber-Commons – The Second Wall Display**

Until now, we have been using LambdaVision, our 100-megapixel display described previously. While large and very high in resolution, it came at a cost of one-inch bezels between adjacent screens. By accounting for the pixels that would have been underneath the bezels, we can mitigate their effects (e.g. lines crossing the bezels will appear straight as opposed to broken). Although this to an extent works for visual information, it makes text unreadable unless significantly enlarged, which severely reduced the number of potential use cases.

EVL's second wall display, the 20-foot Cyber-Commons, was therefore constructed of 18, thin-bezel LCD displays (Figure 13). While lower in resolution (20 megapixels), the borders between adjacent screens were now only 7mm, which alone dramatically increased its usefulness in real-world use cases. Displaying text was not an issue anymore and almost universally users treated it as a single seamless display surface, as opposed to a "tiled-display". Technological improvements in graphics and computing power also allowed us to drive the display with a single computer as opposed to a cluster. The benefit was the ability to run a standard desktop operating system (Linux and Windows) across the whole wall. Instead of allowing us to overcome the downsides of SAGE by taking advantage of a rich collection of available applications, this only confirmed that desktop operating systems do not scale well to wall-sized displays. For instance, they do not support simultaneous multi-user interaction, the interfaces rely too much on the accuracy of interaction devices and UI elements are difficult to see. Similar issues have also been reported by the research community (Ball and North 2005;

Czerwinski et al. 2006), which motivated us to develop advanced user interaction techniques in SAGE to fully take advantage of the wall display affordances.

Another useful feature of the new wall was the potential to accept video signal from standard DVI and VGA devices and spread it across either half of the wall automatically, although at a maximum resolution of 1920 x 1080 pixels. This effectively serves the same purpose as an HD projector for plugging in laptops and other devices.



Figure 13. Cyber-Commons, EVL's second generation, 20-foot, thin-bezel wall.

## **5.2 First Design Phase – Direct, Multi-modal Interaction**

The first step in enabling the direct interaction required appropriate physical interaction devices for the two interaction zones identified earlier: interaction from a distance and interaction from a laptop while sitting. Although interaction from a laptop was possible through the initial SAGE UI and SAGE Web UI interfaces, they were

*indirect* interfaces as windows were manipulated through their representations in the desktop interface, which did not allow for pointing while seated. *Direct* interaction on the other hand, allows for pointing and manipulation of windows directly on the wall. Interaction from a distance on the other hand assumes users aren't seated and therefore cannot make use of a laptop. For this case, several handheld interaction devices were evaluated.

### **5.2.1 Physical Interaction Devices – Interaction From a Distance**

In order to fully exploit the affordances of large high-resolution displays, users should ideally be mobile in front of the display (Ball et al. 2007). While the mouse has been the de facto standard for desktop interaction, it will not cover all applications of large high-resolution displays primarily because it confines users to a hard surface. Although, the research community has investigated various novel interaction devices (Ahlborn et al. 2005; Baudisch et al. 2006; Forbrig et al. 2008; Malik et al. 2005), many of them require elaborate setups and equipment for which most end users will not have time or resources to set up or maintain (e.g. expensive camera trackers). Furthermore, most of these devices do not work well or at all in collaborative settings where true multi-user interaction is desired. Therefore, the physical interaction device independence attained in the design of the interaction framework, proved to be invaluable since it enabled rapid experimentation with various devices and ultimately allowed further study of human-factors issues presented in this document. Various off-the-shelf devices such as joysticks, Gyromouse, trackpads, 6 degree-of-freedom tracked wand and the Wiimote were evaluated.

After much experimentation with various devices, the Gyromouse seemed to be the most suitable one for interaction on large, high-resolution displays (Figure 14 illustrates the controls used for interaction). It was relatively inexpensive, readily available and since it was essentially a mid-air mouse, it was easy to connect to the system and allow users to be completely mobile in front of the wall. However, it has not seen as much use in our lab as expected. We believe this to be because it was still a relative pointing device, which meant it required frequent “clutching” to traverse larger distances, especially on the 100-megapixel LambdaVision display. Whereas traditional clutching is performed by lifting and repositioning the mouse on a surface, Gyromouse clutching is performed by letting go of the trigger button enabling the movement of the cursor, moving the mouse in mid-air and again pressing the trigger. This often confused users since they were used to the mechanics of the traditional desktop mouse. To make matters worse, dragging an object on the display meant holding two buttons at the same time: the trigger to enable cursor movement and a button to grab the object on the screen. If a clutching action was required in the middle of a drag operation, the user had to remember to let go of the trigger while still holding the button that is grabbing the object. Although with little practice, users would get used to this, the mere presence of learning curve sometimes discouraged novice users.

Unlike the Gyromouse, Wiimote was an absolute pointing device that does not require “clutching” for traversing the whole range of the wall. This effectively eliminated the learning curve making it more suitable for novice users. However, in its default configuration, Wiimote is tailored to the typical home-theater environments and does not scale well to wall-sized displays. To mitigate this problem, an improved tracking

approach was developed that effectively doubled its operating range and off-axis performance. Nevertheless, even with the improvements, it depended on the line-of-sight for operation and was therefore susceptible to interruptions if users obstructed its view in collaborative usage scenarios. While appropriate for use cases where such obstructions were rare, Gyromouse remained the preferred device for interaction from a distance.

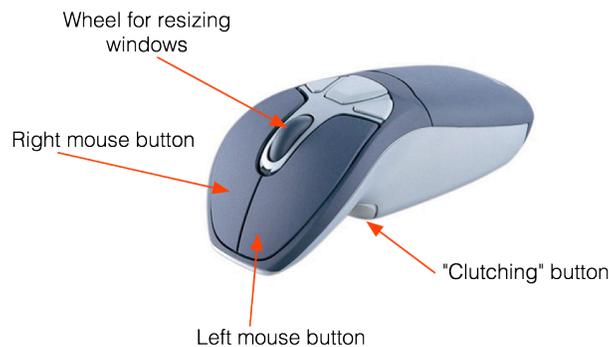


Figure 14. The Gyromouse (made by Gyration) is used for enabling interaction from a distance.

### 5.2.2 SAGE Pointer – Interaction From Personal Laptops

As reported previously, there is a need for interaction modalities appropriate for a large number of users however, providing every user with a physical interaction device was impractical. While SAGE UI could have easily been retrofitted with direct interaction capability, it was too full-featured for a casual user and could potentially be confusing with a mix of direct and indirect interaction capabilities. Therefore, SAGE Pointer was developed, a small cross-platform application (Windows and Mac) that allows users to share a variety of image and video formats, PDF documents and personal

desktops using a VNC protocol (Figure 15). It was conceived as a lightweight replacement to the original SAGE UI while empowering every user with direct wall interaction from their laptop. It allows users to simultaneously manipulate windows on the wall by providing every user with their own pointer, controlled by the laptop's trackpad or mouse. To gain control of the wall, users simply move their cursor to the top portion of the laptop screen at which point their cursor shows up on the wall and disappears from the laptop screen. This metaphor is similar to the one of extended monitor when a secondary display is connected to a laptop.

Besides being able to share local media files and documents using a simple drag-and-drop interface, users can also share images and YouTube videos directly from a web browser, eliminating the need to first download them onto their computer. Furthermore, for information that cannot be uploaded to SAGE, users have an option to take a local screenshot, which can automatically be shared on the wall by the SAGE Pointer application. It is important to reiterate that users are not able to edit information in SAGE, only display it (e.g. editing a text document on the wall is not feasible).

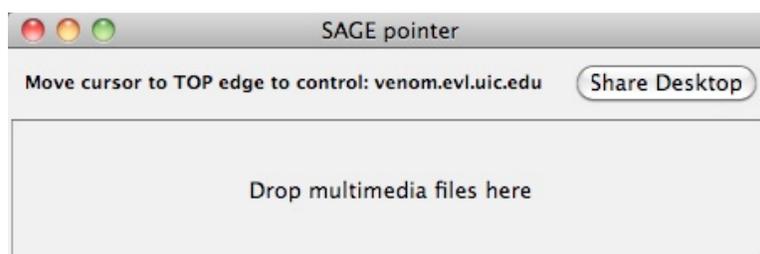


Figure 15. SAGE Pointer is a simple cross-platform that allows users to control SAGE walls from their laptops, share their desktop, media files and documents.

### 5.2.3 Summary of Design Improvements

Figure 16 summarizes the current state of the interaction framework with the early SAGE observations and the resulting features highlighted in blue. The early observations of SAGE use in weekly meetings and the Anatomy class indicated a clear need for direct interaction capabilities for multiple users simultaneously. This was the primary motivation for the development of the interaction framework that enabled the variety of use cases presented in this document.

Furthermore, the early observations also revealed two distinct interaction zones that users were found in: from a distance and while sitting. Direct interaction from a distance was supported by various physical interaction devices, of which Gyromouse and the Wiimote were the most suitable. Because of its large operating range, Gyromouse was particularly appropriate for encouraging physical navigation, which has been previously shown to improve performance while being more natural (Ball and North 2008). Interaction while sitting was enabled through SAGE Pointer, a lightweight replacement for the original SAGE UI, which enabled direct interaction and pointing from a laptop. Although prior work has proposed various novel interaction devices for wall displays, they were all targeted at single-user environments. While the Gyromouse or the Wiimote weren't novel input devices, they did allow concurrent multi-user interaction, which was necessary for studying real-world collaborative use cases.

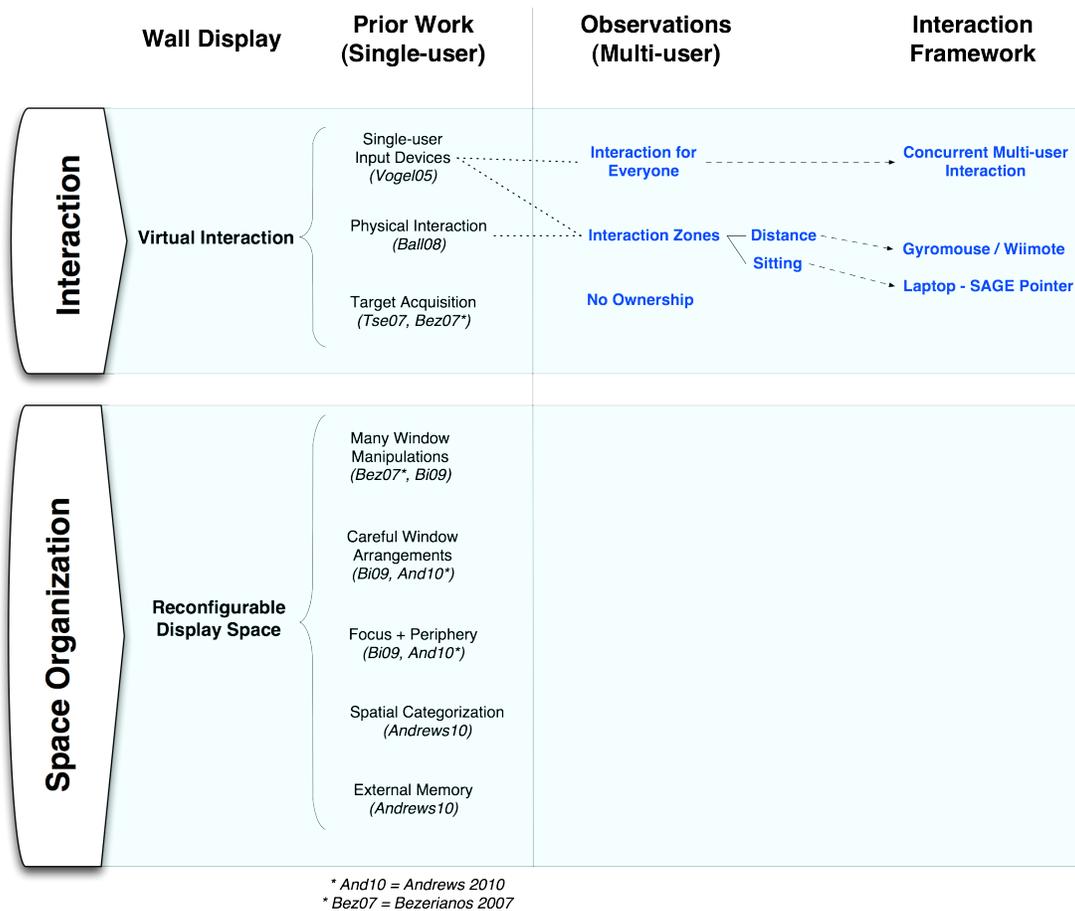


Figure 16. The first design phase focused on enabling direct interaction. Observations and the resulting framework improvements are highlighted in blue.

### 5.3 Observations

#### 5.3.1 Classroom – Visual Analytics Class #1, Half-wall

Besides the experience with the new interface during our meetings, a semester-long computer science class on visualization and visual analytics was also taught using the Cyber-Commons wall. The focus was on teaching various visualization techniques

through existing visualization examples. Because of the thin bezels, viewing text was not problematic anymore, unlike during the anatomy class which used the previous generation LambdaVision wall. Therefore, the professor would hold lectures from a website prepared for the class displayed on one half of the wall through a laptop connected by a video cable, essentially mimicking a typical projector setup. On the other half of the wall, SAGE was used for showing media pertinent to the discussion (Figure 17). There were two main reasons for this separation as opposed to using SAGE for the whole wall. Firstly, we had no native web browser for SAGE, which meant screen sharing was our only other option for showing webpages. At the time however, it was limited to a few frames per second, which is normally enough for meetings but not enough for interactive visualizations often shown in the class. The second reason for separation was the inability to logically partition the wall to support two organization modes simultaneously: one side where the lectures notes are always maximized and the other where media can be organized independently without affecting the lecture notes (e.g. if one wanted to tile the images, it would tile the lecture notes as well).



Figure 17. Visual Analytics class taught on the Cyber-commons wall. The left half contained the lecture notes while the right half ran SAGE for juxtaposing media relevant to the lecture.

Although the media was already embedded in the lecture notes, displaying them separately on a high-resolution display allowed the professor to juxtapose them, for instance, for comparing and contrasting various visualization solutions to the same problem (Figure 18). Ultimately, this externalizes students' memory since all examples were visible simultaneously eliminating the need to remember the characteristics of each visualization. Students confirmed this benefit during a post-semester survey where eighty percent of the students felt that they were learning significantly more in this classroom space than they would have in a more traditional classroom equipped with whiteboards and a single projector.

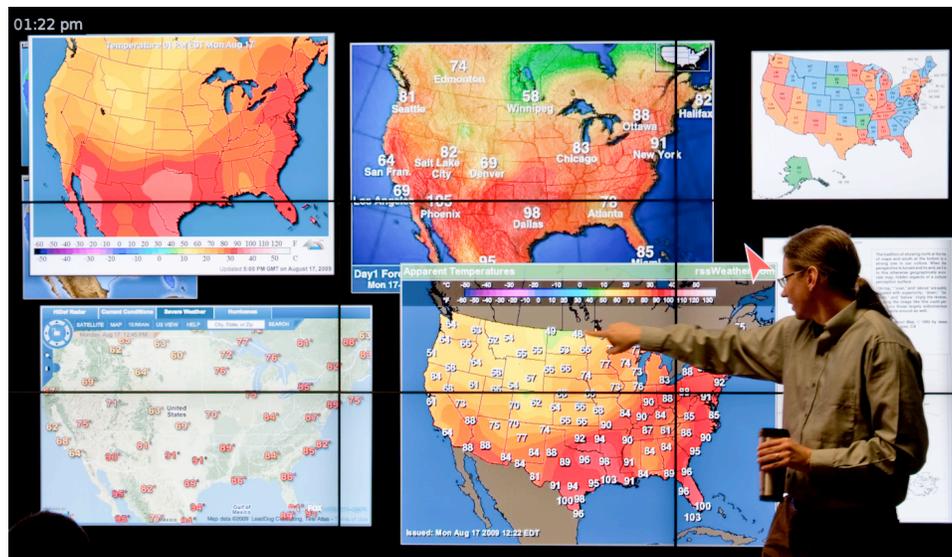


Figure 18. Juxtaposing media externalizing students' memory and allows direct comparison.

To show the media files on the SAGE display, the professor would drag and drop them either directly from the web browser (e.g. the lecture notes), or the local hard drive, onto the SAGE Pointer. A Gyromouse connected to the same laptop was used to control the web browser on the computer and also to control the SAGE display. The switching between the control modes was done using the SAGE Pointer application, which assigned one of the auxiliary buttons on the Gyromouse to do the switching (this was prior to implementing the extended monitors metaphor for sharing the cursor). Although the professor was skilled at using the Gyromouse, switching between the modes remained confusing throughout the semester, which ultimately motivated the shift to the extended monitors metaphor. It is important to note that the Gyromouse was used because it allows for the greatest mobility in front of the wall, which was necessary since the professor spent a considerable amount of time directly at the wall, explaining the visualizations and

pointing out the details using hand gestures. This was an obvious indication that physical navigation is preferred for being more natural even though it may be slower and requires more physical movement. Ideally then, one would be able to manipulate the objects on the wall directly, through a touch interface for instance, eliminating the need for the Gyromouse and the awkward mode switching.

Besides the lectures, the students also used the wall for paper presentations during the semester. The presentation slides were shown from the laptop connected through a video cable, while on the other half of the wall running SAGE, they would often preload pertinent media in order to bring it up more easily at the time of the presentation. The reason for showing media outside the presentation slides was the same as during the class, to more effectively explain the characteristics of related visualizations. As more students wanted to preload their media, the display quickly became crowded and it was more difficult to find and organize the appropriate windows for each student. This was partly the motivation for the minimize feature described earlier, which allowed students to preload their media on the wall without taking up much space.

The same split-wall setup was used during the project presentations. While students were presenting their visualizations from the laptop on one half, the other half was running SAGE where the professor would upload a screenshots of all the projects. This again had the benefit of spurring discussion by making the strengths and weaknesses of each project evident when compared to others. During the project presentations, the professor would move to the back of the classroom and instead of using the Gyromouse to control the wall, he would use the SAGE Pointer from his laptop. This simply adds to

the substantial amount of evidence supporting the need for multi-modal interaction, even for the same use case.

### **5.3.2 Lessons**

#### ***5.3.2.1 Multi-modal Interaction***

Absolute pointing devices (e.g. laser pointer) directly reflect user's motion with the cursor over the entire display surface, without requiring "clutching" (repeating the gesture in order to cover more distance). Relative pointing devices require clutching, which is understood with a desktop mouse but confusing if done in mid-air (e.g. Gyromouse). This perhaps indicates that the problem is not the presence of clutching itself but rather in the mechanics of it. For instance, one can imagine mid-air hand gestures where clutching is performed quite naturally while turning a virtual object (similarly to how we would repeatedly turn a physical object). In our case, given the inexpensive off-the-shelf devices, a more appropriate choice for novice users was the Wiimote, even though it had a somewhat limited operating range. On the other hand, more experienced users also had Gyromouse as an option, which did not have any practical operating range limitations. In fact, it was the only device available if up-close interaction was desired, even though it would not be intuitive since small movements of the hand would result in large movements of the cursor when viewed from up-close. These two devices however, are far from constituting a complete set of interaction modalities appropriate for all use cases. For instance, while enabling mobility in front of the wall for several users, they might not be practical in scenarios involving a large number of users or users that are mostly sitting down (e.g. our weekly meetings).

Therefore, the solution currently does not lie in a single ideal device but rather in a set of devices with various interaction affordances. One might have to consider criteria such as the number of simultaneous users, the operating range, the necessary accuracy and the speed of the device.

### ***5.3.2.2 Easy Window Manipulations***

Increasing the size and resolution of displays also increases the number of visible windows. This is in contrast to desktop displays where windows are mostly overlapping and there is rarely more than one window visible at a time. This means that the problem of window switching on desktop displays is becoming more of a window organization problem, requiring frequent window manipulations. Simplified window manipulation encourages users to experiment more with the organization of windows, potentially providing new perspective on their problems. Besides merely increasing the ease of manual window manipulation, there is great potential for automating certain frequently used paradigms using more powerful window organization tools. For instance, bringing multiple windows into focus or moving and resizing multiple windows simultaneously appeared to be a common behavior.

### ***5.3.2.3 Off-loading Information***

A naïve argument may be that given “enough screen space”, all the windows will be visible and tiling will become a prevalent method of window organization. In reality, given more screen space, we tend to display more information, which eventually necessitates moving it aside while keeping it within reach. Another reason for off-loading is the visual overload when a large amount of information is displayed simultaneously.

#### **5.3.2.4 Persistence**

Although off-loading information serves to maximize the amount usable wall display space, it does not lend itself for supporting long-term work. In such cases it will be crucial to save and reload the state of the wall at a later time. For instance, in classrooms, lectures sometimes spread over multiple class sessions, which would require reloading the material from the previous class to continue where the class left off. Such persistence features are increasingly important as the amount of visible information increases since manually rearranging it would be impractical and could therefore discourage further use of wall displays for long-term work.

### **5.4 Second Design Phase – Enhancing Display Organization**

As noted previously, managing an increasing amount of visible information requires more powerful window organization techniques. Several opportunities for improvement have been observed, which motivated the development of the following initial set of organization techniques:

#### **5.4.1 Off-loading Information – Minimize**

Although similar to the iconize feature found on desktops, minimize feature was tailored for wall-sized displays. The minimize bar was positioned at the bottom, along the whole width of the display. To minimize a window, users would simply drag the window down, onto the minimize bar. It is important to note that minimized windows were not “iconized” as they usually are in desktop operating systems and instead are simply rescaled to a smaller size. For instance, if a video was minimized, it will still be playing

(albeit at a lower framerate to preserve system resources). Given a large number of windows, this aids significantly in window identification, especially since the window aspect ratio is preserved as well. To take advantage of user's spatial memory, a minimized window was positioned at the same location where the user dropped it onto the minimize bar. This further facilitated window identification. Windows minimized in the same area are also tiled, preventing overlap. Though tiling will alter the window's minimized position slightly, it will be not be significant enough to invalidate user's spatial memory. A single click on the minimized window restores it to its original position.

#### **5.4.2 Maximize**

Perhaps another term for maximize in the context of wall displays would be "bring into focus" as maximizing a window over the whole display is rarely desired. Furthermore, one might imagine the utility of maximizing a window only within a physical display (i.e. one tile) (Grudin 2001), however, in case of Cyber-Commons and its thin bezels, users typically regarded the whole display as one seamless surface without paying much attention to the physical screens. Therefore, since we are dealing mostly with media files in SAGE, maximize feature simply brings the window into the center of the display and makes it as large as possible while preserving the window's aspect ratio. Double-click anywhere on the window maximizes it while the second double-click restores the window to its original position.

### 5.4.3 Tiling

Although large size and high resolution enables us to juxtapose many pieces of information for direct comparison, it is very time consuming to do manually. We have therefore developed a tiling feature that will organize all the visible, un-minimized windows into a grid, with each cell being of equal proportions. The final cell size is determined from a combination of the average window aspect ratio, the aspect ratio of the display and on the number of visible windows. Although not optimal, it provides fairness given a variety of window sizes and their aspect ratios. Originally we had a second tiling mode, which uses physical screen boundaries as cells, however, this has not seen any use on the Cyber-Commons display. Perhaps this is another hint that 7mm bezels are small enough for perceiving the display as seamless.

Tile mode can be either turned on or off. Even when on, windows can still freely be moved and resized manually, allowing for further layout customization. When on, upon adding new content, the display will automatically retile itself. Turning the tiling mode off will simply disable new window tiling but it will preserve the current organization of windows. The order of windows in the grid is based on the age of the window (i.e. how long the window has been on the display), the oldest being first, the youngest last. This is beneficial when new content is added to the display as it shows up at the end of the grid without destroying the ordering and therefore spatial memory.

### 5.4.4 Multi-window Manipulation

Given that it is tedious to manipulate a large number of windows individually, we have enabled simultaneous multi-window manipulation, much like with icons on the

desktop. This applies to moving, resizing, maximizing, minimizing and closing actions. Using the secondary click (e.g. right-click), one can drag-select, click-select or use a combination of both actions to select multiple windows. Moving, resizing and closing simply replicates the action to multiple windows similarly to icons on the desktop. Minimizing multiple windows simultaneously will bring all the windows into the same area within the minimize bar which serves as a temporal reminder that a certain set of windows was minimized together. Maximizing a set of selected windows will bring all of the windows on top of the others and tile them using the same algorithm described previously. This essentially juxtaposes all selected windows for direct comparison. Multiple users can make their own selections and perform multi-window manipulations independently.

#### **5.4.5 Push-to-back**

In a desktop environment, reaching for a window completely hidden below the window in focus can easily be done using the ubiquitous ALT-TAB window switching operation. On wall displays however, we need a different window switching operation since the user mobility requirements make keyboards impractical. Although the top window can simply be moved aside to reveal hidden windows, it could also be pushed below all the windows without a change in size or position. While similar to the ALT-TAB operation, it does not switch to the most recent window but simply pushes the top window all the way into the background, revealing the window immediately underneath. This way the operation is localized and does not affect the layout of windows in other areas of the display.

### 5.4.6 Persistence

Although low-level support for persistence has already been developed and found rather useful during the anatomy class, we now had the ability to simplify its use further by designing its interface using SAGE Widgets. We have therefore designed a media library that keeps track of all the files shared by the users (Figure 19). This allows users to go back in time and browse older media directly on the wall, which is presented as thumbnails sorted by type and the time of upload. Besides browsing the media, users can preserve the current state of the wall, which saves the position, size and the loaded files of every application window currently running. Saved states can later be easily identified by the wall screenshot captured during saving. For reliability reasons, the system also automatically saves the state after every window manipulation, which can be reloaded in case of crashes to bring the wall to the exact state it was in right before the crash.



Figure 19. The media browser, developed using SAGE widgets, is a tool for easily browsing and showing multimedia files on large high-resolution displays.

### 5.4.7 Summary of Design Improvements

Figure 20 summarizes the current state of the interaction framework with the latest observations and the resulting features highlighted in blue. There was a clear need to support up-close interaction with the wall and various prior research efforts have investigated this modality. However, unlike projected displays used in prior research, Cyber-Commons was an LCD-based tiled display, which prohibited the use of camera-based techniques for tracking touches.

Prior research has also pointed out that users frequently perform many window manipulations and careful window arrangements on wall-sized displays, which was further supported by our observations. To assist window manipulations, several manual and automatic organization techniques have been developed. While not perfect, the automatic organization techniques often eliminate the need for careful manual window arrangements, greatly saving time and effort.

As reported by (Andrews et al. 2010) and observed during the Visual Analytics class, a large amount of screen space serves to support external memory by eliminating the need to memorize information. However, even high-resolution displays are not enough for keeping all the information visible, which necessitates off-loading of information. The developed minimize feature serves the external memory by keeping the off-loaded information around, in the exact region where it was minimized. To further assist window identification, minimized windows still have the same aspect ratio as their un-minimized state. Lastly, their content is still displayed fully (e.g. a movie is still visibly playing while minimized). When off-loading information isn't enough, the whole

state of the display can be saved and recalled at a later time by browsing screenshots of the saved states.

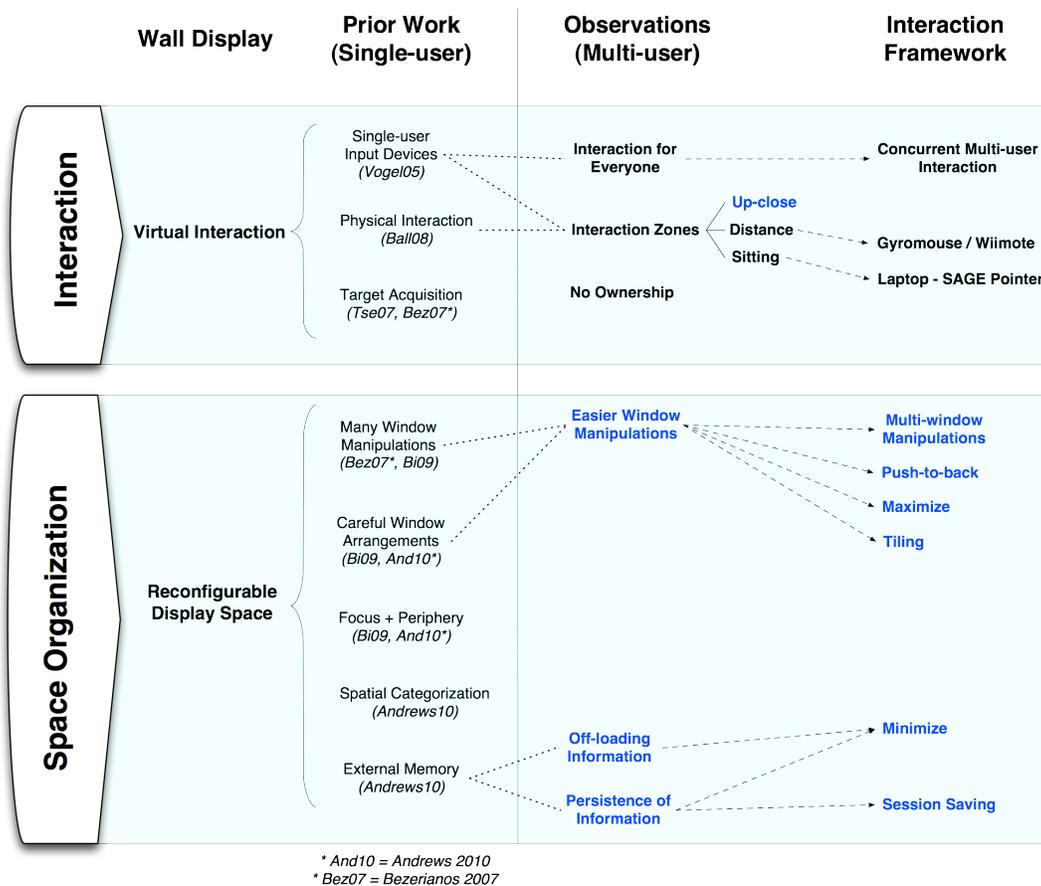


Figure 20. The second design phase focused on enhancing the display space organization through easier window manipulation techniques. Observations and the resulting framework improvements are highlighted in blue.

## **5.5 Observations**

### **5.5.1 Meetings**

We continued to hold our weekly meetings in front of the display however, as previously observed, neither Gyromouse nor the improved Wiimote were ideal for direct interaction given the practical limit to the number of devices. This motivated the development of the aforementioned SAGE Pointer application to allow direct control of the wall from the laptop. Although again, not ideal for all scenarios (e.g. the anatomy class), it was appropriate for our weekly meetings since every participant already used a laptop to share their desktop and present their work.

The extended monitor metaphor for sharing cursors on the wall was generally well accepted and understood nevertheless, there were several occasions when users were confused about the location of their cursor (i.e. wall or laptop). This appears to be more of a usability issue as opposed to a paradigm flaw, as several subsequent usability improvements have reduced the number of such occurrences dramatically. Ultimately, giving users the full ability to control the wall from their laptop as if they were using a handheld interaction device, ended up being highly appreciated by the users, especially given the ability to simultaneously and independently manipulate information. Such interaction paradigm proved so natural in fact, when users did not have their laptops and therefore, the ability to point at and interact with the display, certain subtle frustration became evident as they could not easily convey their ideas and thoughts relative to the existing conversation and content on the display. Clearly, this would not be possible without integrating true multi-user and multi-modal capabilities into the interaction

framework. Since new devices (i.e. pointers) can be added dynamically, users can casually switch back and forth between controlling their laptop and controlling the SAGE display, and would do so numerous times during the meetings. The need to casually join and leave meetings was also observed previously by Teasley et. al (Teasley et al. 2000) in the context of traditional war rooms.

Window manipulation was fast and there was virtually no learning curve associated with the SAGE Pointer since it used a standard laptop trackpad, which inherits all the accuracy and speed advantages of the desktop mouse. The window layout during the meetings remained the same as before, with personal desktops as small windows along the periphery and one or more large windows in the center of the display serving as the focus of the discussion (Figure 7). In addition, given a more seamless display and a more intuitive user interface, users were now more readily sharing media on the wall pertinent to the discussion. Typically this ended up being a video or a few images but occasionally a large number of images would be shared, posing a problem of organization. Although the tiling feature was available, it was not utilized because this would tile all the windows, including other users' shared desktops, which they did not want tiled. While manual workarounds were possible (e.g. minimizing everyone's window first and then tiling just the images), it was easier for the users to simply forgo sharing media on the wall and display them directly on their shared desktop. This was a hint that the whole display sometimes should not be treated as a single contiguous surface but rather partitioned as needed to support individual or transient organization structures. Alternatively, the information on the wall could be explicitly grouped and manipulated as such.

Generally, the SAGE Pointer application seemed to be well accepted since it was a much simpler interface to SAGE and at the same time more powerful than the old SAGE UI. Interestingly, these benefits encouraged further, both scheduled and impromptu, uses of the Cyber-Commons wall as a shared collaborative medium. This is perhaps a testament to the true merit of the large high-resolution environments and a hint that we were moving in the right direction in making the most of the affordances of such environments.

### **5.5.2 Analysis Task – “Image of Research” Judging**

Image of Research is an annual competition held by the University of Illinois at Chicago, which allows students to submit images representative of their research in order to receive cash prizes and public recognition. A panel of seven judges is assembled into a committee every year that picks the top six submissions based on a variety of criteria. In prior years, the submissions were reviewed on a laptop connected to a projector while this year the committee was invited to use our new Cyber-Commons wall. A judge familiar with the technology took the lead in controlling the wall from his laptop, using SAGE Pointer, and began by tiling all the submitted images on the wall (close to 50 images). Besides the visual appeal of the image, another ranking criterion was the research represented by the image, as described in the submission statement. Therefore, the judge in control also shared his laptop screen on the wall with the website containing all the research statements. To make sure this information was visible at all times, the shared desktop window was maximized on the right, while the tiled images were manually moved to the left half of the wall (Figure 21). This way the tiling mode was still

utilized but further optimization of the window organization was done manually. The multi-window manipulation proved very useful here since whole columns of tiled images could be moved simultaneously, as opposed to moving every image individually. The ability to take advantage of the tiling mode while still being able to make manual adjustments was quoted as a valuable feature. This was perhaps an indication that the system can provide a degree of structure in organization while still allowing enough freedom for manually fine-tuning the layout. Alternatively, the wall could be partitioned into regions for supporting varying organizations, entirely eliminating the need for manual adjustments.



Figure 21. The Image of Research judging process made use of the Cyber-Commons wall for juxtaposing all the submissions on the left while providing the research statement on the right.

During the first stage of the review process, images were sorted into two categories, “keep” and “discard”. Discarded images were minimized while images for keeping were simply left in their original position on the wall. Since minimized images were still easily recognizable, it allowed the committee to refer to them for comparing

similar submissions. Upon deciding on the top 20 submissions, the images were categorized by the domain they were representing (e.g. humanities, science, art etc.). This was done in order to get an overview of how well each domain was represented and to try and nominate submissions from every domain. The categorization was denoted using spatial grouping, for the lack of explicit support for such organizational structure (Figure 22). To determine the top six submissions, judges voted for their own top picks, which determined the final ranking based on the number of votes each submission received. For the lack of annotation support, the committee decided to treat the Cyber-Commons wall as an actual wall where votes were placed by attaching post-it notes on top of the digital images (Figure 23). Although this move ended the possibility for any future virtual interaction and replaced it with physical interaction, it was perfectly acceptable and natural this being the final stage of the judging process. It became obvious that even novice users grasp the physical navigation affordances of wall displays and will attempt to utilize it whenever possible. Ultimately, this is another indication that we should bridge physical and virtual worlds and combine the benefits that each offers.



Figure 22. For the lack of explicit support, information was categorized using spatial grouping during the Image of Research judging process.



Figure 23. During the final stage of the Image of Research judging, the committee took advantage of the physical navigation affordances of wall displays and used post-it notes for placing votes.

### **5.5.3 Lessons**

#### ***5.5.3.1 Up-close Interaction***

As observed in the visual analytics class and the Image of Research judging process, physical navigation is an affordance of wall displays, which users embrace naturally and take full advantage of. While users are always free to walk up to the wall to see the detail and physically move around to analyze the information, it has to be combined with the ability to manipulate the information on the wall. Although the Gyromouse allows for complete freedom of movement, it is not very natural when used up close to the display since it feels like an intermediary device when there is no need for one. Preferably one would manipulate the objects directly on the wall (e.g. through a touch interface).

#### ***5.5.3.2 Display Partitioning***

Although the Cyber-Commons wall was practically a seamless surface, spatial grouping of information on the wall has been observed in almost every use case so far. Either different regions of the wall had distinct purpose or information was spatially grouped based on a commonality. For instance, during the visual analytics class, there was a need to keep lecture notes separately from the media used for discussion, each calling for a different window layout on different regions of the wall. On the other hand, during the Image of Research judging, images were spatially categorized according to their domain. These behaviors indicate that large walls, although seamless, are logically partitioned and will need to provide explicit support for various simultaneous organizational structures. Although separation into peripheral and focal regions has been

observed previously in the context of single-user daily work (Bi and Balakrishnan 2009) and multi-monitor environments (Grudin 2001), our experiences indicate the need for more freely reconfigurable regions, especially given the collaborative nature of our environment. For instance, regions could be defined for supporting individual or transient organizations of information.

## **5.6 Third Design Phase – Display Partitioning and Up-close Interaction**

### **5.6.1 Touch Interaction**

Although the Gyromouse was used for the bulk of the interaction, when users are at arm's-length from the display, it becomes somewhat confusing to use because small movements of the mouse result in large movements on the display. In such cases, directly interacting with the wall using touch gestures is more natural since it takes advantage of physical navigation, as observed in most of the previously described use cases. Although enabling touch interaction on rear-projected walls can be achieved using cameras behind the screen, this technique is not applicable to LCD-based walls. However, PQLabs offers a solution using a simple frame around the wall, consisting of IR emitters and receivers, which can detect up to 32 touches simultaneously (PQ Labs). Having physical interaction device independence in our interaction framework again proved invaluable as integrating a touch screen device into DIM simply required converting raw touches into our generic set of events in a device plugin. Furthermore, the multi-user attribute of the interaction framework allowed us to design and prototype multi-user, multitouch gestures. Although users now have the ability to interact from a distance (Gyromouse, Wiimote or SAGE Pointer) and interact from up-close using touch input, each input modality has different

strengths and weaknesses. This exposes a question of how to design the wall user interface so that it is appropriate for all input modalities? For instance, Gyromouse and Wiimote, although imprecise, are able to quickly traverse large distances. On the other hand, touch input can be rather precise but it could take a long time to drag an object across a 20-foot wall. We began by replicating the existing window manipulation functionality using touch gestures:

### **Move**

The intuitiveness of touch interfaces comes from direct relationship between the action and the result. For instance, if a finger moves 2 inches, the object being acted upon should move 2 inches as well, as it would in real life. However, on a 20-foot wall like Cyber-Commons, this means the user would have to walk 20 feet in order to move a window from one side of the wall to the other, which would quickly become tedious given a large number of such actions. As a better approach, we designed two alternative solutions, an imprecise one and a precise one. An imprecise solution involves a single-finger flick gesture that launches the windows as if they were thrown across the display. The speed of the flick determines how far the window will travel while the direction of the flick determines the direction of the throw. This approach retains its roots in the real world and is easily understood since it mimics throwing real objects. The second approach allows users to precisely position a window anywhere on the wall, without having to walk the distance. A single-finger drag motion will drag a window at an accelerated pace according to a gain factor. For instance, given a gain factor of two, one-inch movement of the finger will move the window two inches. The gain factor is determined dynamically based on the display size, satisfying the scalability

characteristics of the interaction framework. Some users expressed skepticism about this method, mentioning that it will not be very intuitive. However, after using the wall themselves, they quickly realized its benefit and noted that the gesture actually makes a lot of sense.

### **Resize**

Resizing is performed using a familiar two finger-pinch gesture. However, during pilot testing, users would naturally attempt a pinch gesture with two hands at a distance significantly larger than the distance between two fingers on the same hand. Therefore, the size range of acceptable pinch gestures was increased to include two-handed attempts as well as the standard two-finger ones.

### **Close**

As described previously, closing windows using physical interaction devices was accomplished by dragging and dropping the windows on close button at the top of the wall. Although possible to do using the touch interface as well, we felt there should be an easier way of accomplishing the same task. Therefore, a five-finger hold on the window will start fading the window. If the window reaches complete transparency after about one second, it will be closed, otherwise the close operation will be canceled and the window will return to normal opacity. Although not rooted in the real-life gesture, it has the benefit of requiring five fingers, which are rarely used during regular window manipulations and therefore not easily initiated by accident.

### **Maximize**

The maximize is initiated using a double-tap, much like the double-click using other interaction devices.

### **Minimize**

Windows are minimized by dragging or flicking them to the bottom minimize bar. While the close button also requires dropping a window onto it, it is significantly smaller in size than the display-sized minimize bar, which was the reason for introducing a separate closing gesture. In order to make space on the display, we have developed a gesture for minimizing all the windows at once. A five-finger downward swipe will begin “pulling a curtain” over the windows and if the swipe is continued and the curtain reaches the bottom, all the windows will be minimized.

### **Push-to-back**

Pushing windows to the back is performed using a big tap, a tap with a large surface area. Users would typically use the whole palm or the bottom of their fist. This somewhat mimics real-life gestures where a bigger push is expected to move objects further, and since a tap is not directional, the object being acted upon could only go into the screen.

## **5.6.2 Display Sections**

Prior research has pointed out that in multi-monitor systems users take advantage of the physical bezels for partitioning their workspace, typically into focal and peripheral regions (Grudin 2001). Despite the small physical bezels, the Cyber-Commons wall was treated as a seamless, contiguous surface. Nevertheless, we have still observed logical partitioning of the wall into different sections independently of the bezels. As these near-seamless and seamless large high-resolution displays are becoming more prevalent, we will not be able to rely on the organizational benefits of distinct physical screens.

Therefore, to aid in spatial organization on such displays, we developed sections for explicitly partitioning the wall into any configuration of rectangular regions. This approach is similar to organizing documents (or files) in a folder, however, information on the wall is still visible which gives it a dual purpose: it acts as a document itself while its section membership embeds a secondary meaning (e.g. time or category).

Sections are resizable rectangular regions allowing for various simultaneous organizational strategies on one seamless wall surface (Figure 24). To create a section, one simply splits any section into two, vertically or horizontally. This process can be repeated recursively allowing for any arrangement of sections on the wall. Removing a section is done using a small close button in each section, which simply assigns all its windows to the parent section. Each section can be tiled independently and will be retiled automatically if a new window is dropped into it while the tiling mode is turned on. When a window is dropped into a different section, it is automatically resized to fit within that section. Similarly, when sections are resized, all the windows within the section and the neighboring sections are moved and resized accordingly. The section membership is determined by the position of window's center. Manually moving and resizing windows within a section is still possible, although if the center of the window happens to fall outside the current section during a resize operation, the window will be automatically fitted into the new section.

With the introduction of sections, several previously described features were adjusted to better support selective organization strategies that sections offer. Minimizing multiple windows using a five-finger swipe gesture now acts only on the section where it was initiated, minimizing the windows within that section only. Similarly, a double-click

(or tap) will first maximize the window within its section while a second consecutive double-click will maximize the window over the whole display. Lastly, the persistence feature will also save the current state of sections, even if no windows on the display are present. This allows users to preserve the current section arrangements as templates for later use.

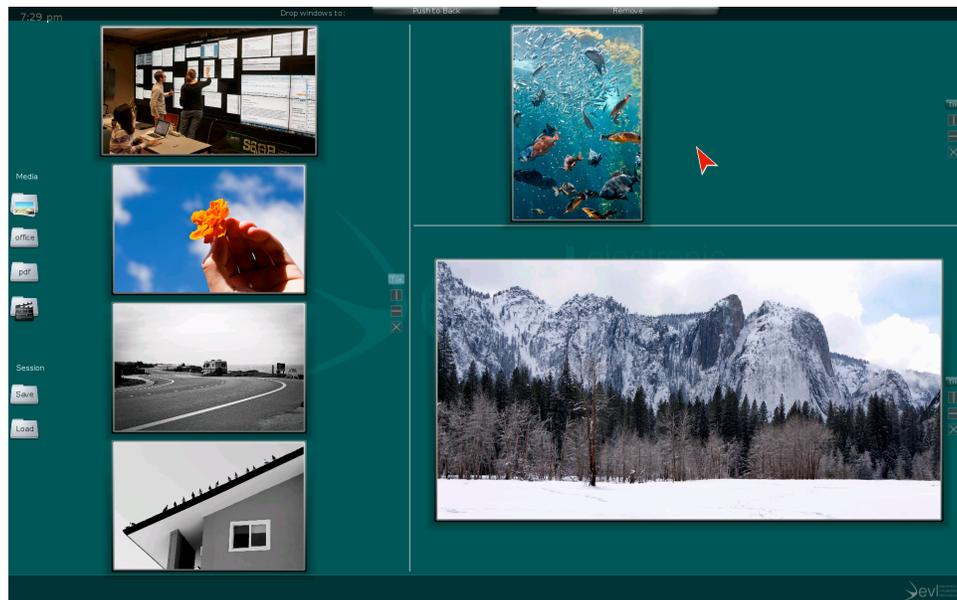


Figure 24. Three display sections supporting various organization strategies. The left section is automatically tiled while the rest are manually organized.

### 5.6.3 Summary of Design Improvements

Figure 26 summarizes the current state of the interaction framework with the latest observations and the resulting features highlighted in blue. Several prior observations have clearly pointed out the need for up-close interaction with the wall. Luckily, recent technological advancements by PQ Labs have made it possible to add

touch capability to the LCD-based Cyber-Commons wall. Although an early prototype with limited response rate, it was sufficient for devising several crucial touch gestures for supporting concurrent, multi-user up-close interaction. Perhaps most importantly, the addition of the touch input modality has the potential to further encourage the more natural physical navigation. The summary of all interaction zones and their affordances is presented below in Figure 25.

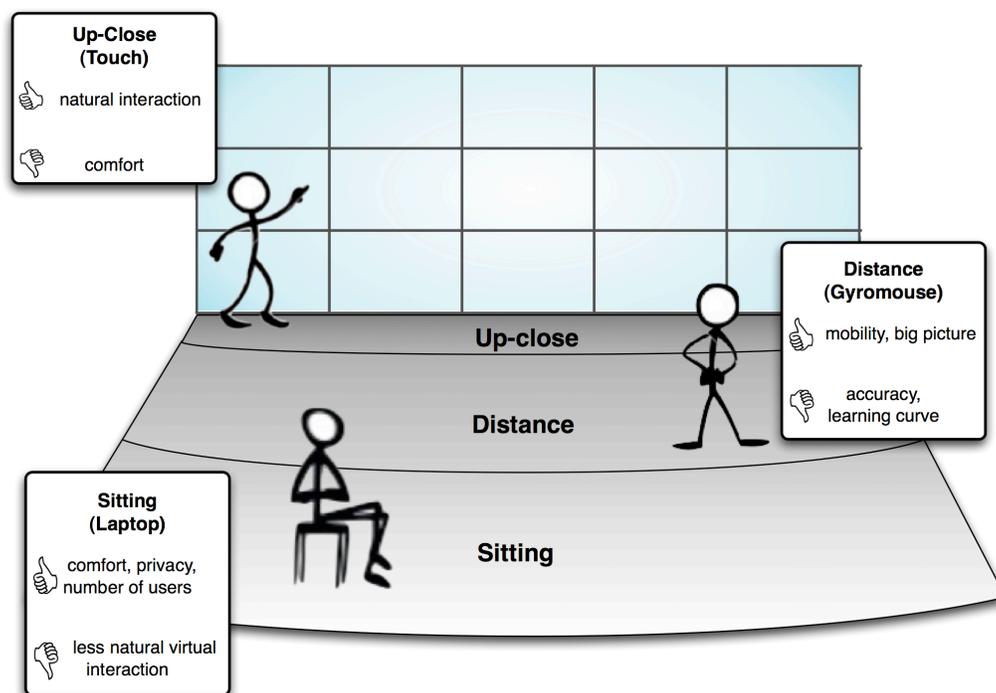


Figure 25. The summary of the three commonly used interaction zones in wall display environments, each with their own benefits and drawbacks.

Although the Cyber-Commons wall was now treated as a contiguous display thanks to its thin bezels, users still logically divided the display space to serve several different needs. Aside from spatial categorization, space was also used for maintaining various organizations of information and for supporting focal and context regions during

tasks. Unlike prior work, which has reported that in single-user environments the center is the focal region, the observed collaborative use cases indicated that focal regions change often and depend on the use case. Display sections (i.e. rectangular, resizable regions) were developed in order to support such uses of display space.

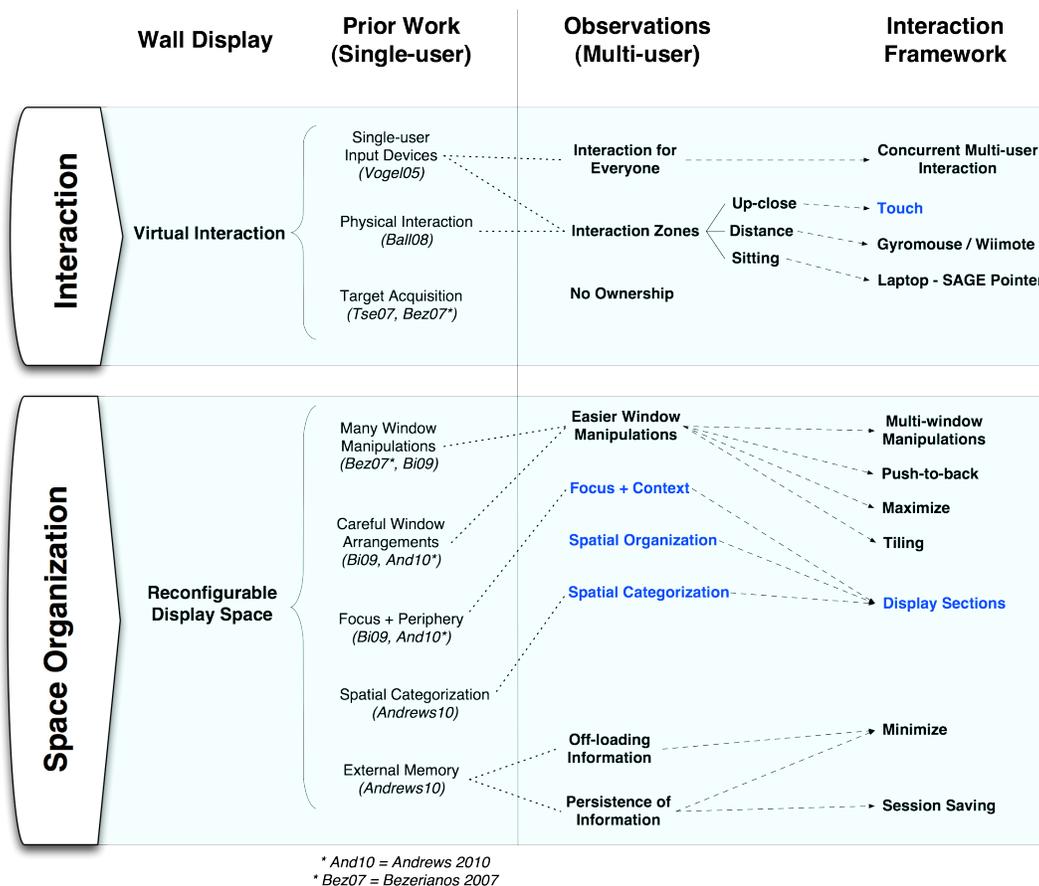


Figure 26. The third design phase brought the touch interface as another interaction zone and display sections for space partitioning. Observations and the resulting framework improvements are highlighted in blue.

## **5.7 Observations**

### **5.7.1 Classroom – Visual Analytics Class #2, Full Wall**

Following the success of visual analytics class the first semester, the course was taught using the Cyber-Commons wall the second semester as well. In the meantime however, we have improved SAGE desktop sharing to run at 25 frames per second making it appropriate for showing real-time information, such as animated visualizations and videos. Together with the touch capability and the display partitioning features, we could now use SAGE across the whole wall without being bound by the speed of desktop sharing or the inability to organize media independently of the lecture notes. The wall was now partitioned into three sections, roughly corresponding to the thirds of the display: the left section was used for juxtaposing media from the lecture notes, the middle was reserved for the shared desktop with the lecture notes while the right section was showing another shared desktop of a Cintiq tablet (made by Wacom) which imitated the capabilities of a whiteboard (Figure 27). Although displaying lecture notes in the center of the wall makes sense for visibility, the choice of putting media on the left was purely practical. Since the initial position of windows in SAGE is always the bottom-left corner of the wall, it made sense to designate the left section for juxtaposing media. While acceptable in this case, one might imagine other cases when this is not the most optimal initial position of windows.



Figure 27. During the visual analytics class, the Cyber-Commons wall was partitioned into three regions, one for juxtaposing media, one for the Cintiq tablet and one for the lecture notes.

Having a touch screen for up-close interaction allowed the professor to use it for manipulating media on the wall while the Gyromouse was still used for controlling the web browser with the lecture notes. This allowed the professor to stand in front of the wall during the whole lecture while completely eliminating the need to switch Gyromouse modes as done previously. This combination of the Gyromouse and the touch input was received quite naturally, despite the need to frequently switch between the two input modalities. In fact, after a class when the professor had to revert to using the Gyromouse only due to technical problems with the touch screen, he proclaimed not even being aware of how much he appreciated the touch screen. This further supports the case for multi-modal interaction and demonstrates that using two input modalities is acceptable and even preferred to a single one, as long as each modality is the right tool for the task. Naturally, this argument likely would not extend to two input modalities

which both require holding a physical interaction device. Although one may certainly argue that exclusively using a touch screen would be an even better approach, it still may not be suitable in all cases. For instance, if the professor could control the web browser and the lecture notes through a touch interface, the window would have to be low enough to be completely reachable which would reduce the visibility for students in the back rows. Furthermore, during the project presentations, the professor would still control the wall from the back of the classroom for displaying individual images, which obviously could not be accomplished using the touch interface.

During the lecture, there were two most common reasons for displaying media on the SAGE wall as opposed to from the lecture notes. First, very large images could be viewed in higher resolution than in a web browser. Those were typically maximized within a section, or less frequently, maximized across the whole wall for an even bigger window. Second, as mentioned previously, media could be juxtaposed for easier comparison. This was often accomplished by simply leaving the tile mode on which would automatically juxtapose media as it was dragged from the web browser. Occasionally, there was a need to emphasize details in one of the windows, which was easily accomplished by maximizing the window, even while in tile mode. Ability to manually manipulate windows while still in tile mode again proved very useful. As various topics or a set of examples came up during the lecture, the left section of the display was cleaned up and the new images were brought up. As switching topics occurred frequently, it was useful to take advantage of the minimize-all gesture described earlier.

Overall, the second semester of the class was a significant improvement over the first semester, primarily due to the addition of the touch interface for up-close interaction, which is rooted in the traditional classrooms where professors teach in front of blackboards. Although counter-intuitive, the addition of the second input modality only simplified the environment resulting in more natural interaction. Additionally, the ability to partition the wall using sections offered finer control over various window layouts.

### **5.7.2 Presentation - Art Seminar**

Although presentations are traditionally delivered from a set of presentation slides, the high-resolution environments open new opportunities for enhancement. While slides are still valuable for providing the general flow of a presentation, they are quite limiting in elaborating the presented concepts using digital media, primarily because written explanations have to be sacrificed in order to display the media. Wall displays however, offer the opportunity to juxtapose all relevant media while still taking advantage of presentation slides for providing the flow and structure to a presentation.

We have had the opportunity to observe a student presentation during an art seminar class. The wall was organized similarly to the visual analytics class, with the slides maximized in a section on one side, while the rest of the wall was dedicated to the section containing tiled presentation images (Figure 28). As the images were referred to during the presentation, they were simply maximized within that section. To control the wall, the presenter used the Gyromouse and the touch input interchangeably throughout the presentation. This provided him with the mobility in front of the wall while also being a natural interaction method when up-close.



Figure 28. During the art seminar presentation, the wall was split into a working region with all the media and the context region with the presentation slides.

During the discussion following the presentation, certain questions prompted the presenter to share additional images and videos on the wall to clarify answers or further support discussion. Eventually, the presenter even shared his desktop on the wall in order to bring up information from a web browser. Since these actions were unplanned and their sole purpose was to display information as easily as possible, the shared media simply ended up being thrown on the display and either maximized or quickly resized for visibility. Essentially, this instantly turned a structured and organized environment into an unstructured one with very temporary organizations (Figure 29). However, since the sections used in the presentation were still present on the wall, on a few occasions they interfered with the temporary window enlargements by automatically fitting a window when not desired. This behavior suggested that although providing structure is useful in managing large amounts of information, freedom of manual window manipulation still

has to be allowed. Although it is already possible to move windows manually, even in tile mode, it is sometimes difficult to do without the interference from the underlying structure provided by the sections.



Figure 29. During the discussion stage, a structured display space quickly turned into a freeform one with many temporary window organizations.

## **5.8 Summary of Design Iterations**

This chapter presented an iterative design approach to understanding the most common interaction and display space organization behaviors users exhibit while managing a large amount of information on wall displays. From the observations of various use cases, a number of techniques and tools were developed for supporting such behaviors, which can broadly be grouped into three categories. First, multi-modal interaction for input from three interaction zones: up-close, distant and sitting. Second, improved window manipulation techniques for managing a large amount of information on the wall. Third, display partitioning features for supporting spatial categorization and

spatial organization of information. Figure 30 presents the final set of techniques that have been developed for supporting virtual interaction and reconfiguration of space in the context of digital war rooms.

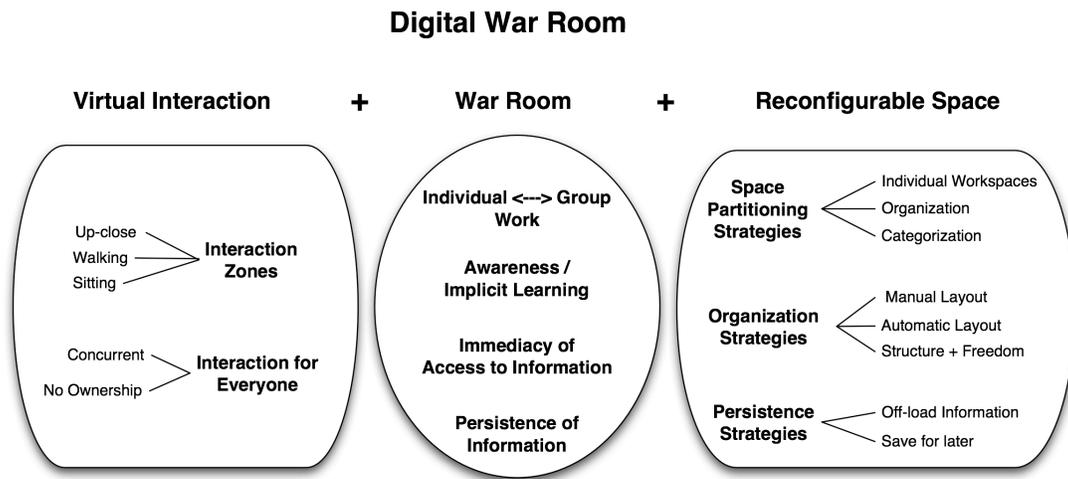


Figure 30. Digital war rooms are a combination of traditional war rooms and large, high-resolution displays, which provide the benefits of virtual interaction and reconfigurable space.

Although each of the presented techniques was developed with multi-user interaction in mind, few uses cases actually demanded concurrent multi-user interaction. Instead, there was frequently one user manipulating the wall while others were participating in the discussion. Therefore, the question remains of how these developed techniques support use cases that are closer to the traditional war room environments with frequent simultaneous multi-user interaction. Investigating such use cases would also reveal how well the presented techniques complement the primary affordances of traditional war rooms, which are responsible for the productivity benefits of radical collocation.

## **5.8.1 Design Needs not Addressed**

The last several use cases highlighted the benefit of the developed techniques in managing information on wall displays. However, they also pointed out further needs that have not been addressed in this work but are presented below.

### ***5.8.1.1 Structure + Freedom***

As observed in the art presentation scenario and the Image of Research judging case, providing organization support (e.g. sections and tiling) is useful when managing a large amount of information with ease. However, we must not forget to permit manual manipulations for fine-tuning the layout or enabling transient organizations, as these situations will inevitably occur given the number of potential applications of wall displays. Although manual window manipulations were allowed in sections and in tile mode, a better support for manually fine-tuning the layout is still needed.

### ***5.8.1.2 Initial Window Position***

Clearly, there is no real reason for initially placing windows in the bottom-left corner and in fact it might not work well in all cases. But the question then becomes, what is a better initial position? Some options are picking a different initial place, allowing users to define the initial position or putting windows in a temporary bin where they do not take up much space. The matter is complicated further in a collaborative scenario and it ultimately may require scenario-specific approach.

### ***5.8.1.3 Annotation***

Annotation is useful in any analytical environment, and it becomes even more essential on wall displays with a large amount of visible information. Although making so much information visible externalizes users' memory, it could also lead to visual overload and difficulty in identifying windows. While spatial organization of windows mitigates the problem to some extent, it may not be enough and users may need to provide further categorization via other means, especially in a collaborative scenario where different users may want to categorize information differently. Annotation could help tremendously since it not only allows highlighting of important details but also allows users to attach additional meanings to individual pieces of information, either for categorization or identification purposes. Unlike spatial categorization, this would also extend to collaborative scenarios since each user can add annotations independently. For instance, in the Image of Research judging scenario, instead of voting using physical post-it notes, users could make virtual notes on each window. This would not have been possible using spatial categorization given that each committee member had to provide his or her own input. Part of the reason for omitting annotation from the set of techniques presented in this document was limited refresh rate of the physical touch screen used in Cyber-Commons.

## **6. EVALUATION - COLLABORATIVE ANALYSIS USER STUDY**

Scenarios such as presentations, meetings, classrooms and a simple analysis task, presented the need for manipulation of numerous pieces of information, which informed the development of interaction and display space organization techniques described previously. While most of these use cases have been collaborative, only the weekly meetings called for simultaneous multi-user interaction. Furthermore, all were rather structured and required little in terms of planning the approach to the task. On the other hand, the greatest benefits of war rooms are realized for more complex tasks involving multiple stages and varying group dynamics. For such tasks, war rooms can double the productivity of teams when compared to individuals working separately (Teasley et al. 2000). Therefore, a more freeform collaborative analysis task was devised in order to investigate how can wall displays and their benefits of virtual interaction and reconfigurable space be used to create digital war rooms. More specifically, how do the developed interaction and display space organization techniques support war room behaviors that necessitate concurrent multi-user interaction, reorganization of space and varying group dynamics?

### **6.1 Study Methodology**

The goals of the study were to try to answer some of the following questions:

- How is the display space organized when the work is divided among users (i.e. when they work separately)?

- How is the display space organized when users work together to reach an agreement?
- What do the group dynamics look like in wall display environments?
- How do users coordinate input control, and which modes of interaction were used?
- What were the fundamental window organization techniques employed during the study?
- How can we design wall display spaces that facilitate collaborative analysis and help users better manage large amounts of information?

### **6.1.1 Task Description**

The task chosen for the study was job searching which involved surveying the then-current job market and finding the top five job openings given a variety of career interests and job criteria. This task was chosen because it required little specialized domain knowledge; it was simple and doable in a reasonable amount of time and potentially involved reviewing many pieces of information. The task was split into two parts in order to explore the two modes of group work we were interested in: working independently (dividing the work) and working together.

In the first part of the task, nine related career interests of the hypothetical job seeker were given to the group and their task was to find as many job openings that match one or more of these interests. They were given 30 minutes to accomplish this part. Eight job searching websites were given to them as a starting point. It is important to note

that the participants were never explicitly asked to work independently on this part, although we anticipated most groups would split up the work.

In the second part of the task, they were given six criteria to judge the previously found job openings by, and were asked to pick the top five jobs, as a group. Their final product was to be a digital poster on the wall listing the top five choices and any relevant information they could find which supported their decisions. For this part, they were given one hour and the only explicit requirement was to agree on the top five choices as a group.

### **6.1.2 Setup and Equipment**

The study was performed using the Cyber-Commons running SAGE across the whole wall, with the three participants seated at a table 6 feet away and centered in front. Each participant was required to have a laptop, either their own or one that was provided. All the features of SAGE were available to the participants, including the touch screen, the Gyromouse and SAGE Pointer on every laptop for interaction. It is important to reiterate that users were not able to edit information in SAGE, only display it (e.g. editing a text document on the wall was not feasible).

### **6.1.3 Participants**

Eighteen (4 female, 14 male) graduate computer science students from our laboratory were recruited for the study, and organized into six groups with three participants in each group. There were two main reasons for recruiting participants from our lab.

Firstly, they all had enough experience with the wall display and SAGE to comfortably share information on the wall and perform basic window manipulations from lab meetings and classes held using the wall in the room. However, they had very little experience with using the screen space for dealing with large amounts of information, especially in a collaborative setting. This was desirable since it allowed us to investigate novel behaviors that emerged during the study, as opposed to behaviors based on preconceived ideas and experiences.

The second reason for recruiting participants from our lab was to minimize social effects on group dynamics by ensuring participants were comfortable in front of each other. The groups were formed where all the participants were friends and have possibly collaborated on a project previously. Attention was also paid to the cultural aspects, where participants of similar cultures and backgrounds were grouped together. Since humans naturally tend to interact with other individuals of similar cultures and backgrounds, this requirement was rather easy to meet.

#### **6.1.4 Procedure and Data Gathering**

Prior to explaining the task to the participants, they were given a 15-20 minute demonstration of all the SAGE features and were encouraged to try them out and ask questions. Following the training period, the job-searching task was explained and they were given a handout with the task specifics and eight websites as the starting point.

Although the most valuable information was gathered by the observing researcher, the study was also video and audio recorded and all the wall interaction was logged as a series of events. Following the study completion, the observing researcher

carried out a 15-30 minute group interview. Every group was asked a common set of questions about their positive and negative experiences with the system and any suggestions they may have for improvement. The second part of the interview was specific to each group and was used to gather a deeper understanding of various behaviors observed during the study by asking the participants to provide reasoning behind their actions. No quantitative metric was employed to evaluate each group's performance since we were primarily interested in the process itself as opposed to the outcome. Moreover, it would be rather difficult to isolate variables responsible for the groups' performance in an exploratory study.

## **6.2 Observations**

Given the exploratory nature of the study and the lack of previous research on the topic, most of our findings are qualitative. However, as mentioned previously, we logged all the user interaction and this data was used to support certain observations or further explain various behaviors.

In the following sections we present observations unique to our study, primarily related to its collaborative aspect, however we also briefly make note of observed behaviors previously reported in single-user studies (Andrews et al. 2010; Bi and Balakrishnan 2009). We believe the fact that we have also observed some of the same behaviors further strengthens the body of knowledge on large high-resolution displays, which will ultimately benefit the community trying to design such systems.

### 6.2.1 Overall Workflow

All the groups took a similar, high-level, approach to the task:

1. *Split the work* by search sites. Every user would take a few and search for all the career interests at those sites.
2. *Independently find* job postings. Although not explicitly mentioned, we expected this behavior.
3. *Collect* all job postings on the wall as PDF documents or screenshots. The number of postings found by the group ranged greatly, from 9 to 32 (average was 19).
4. *Present* individual findings to the group, on the wall.
5. *Filter* the presented job postings through group discussion and narrow the choices down to the top 5.
6. *Make the poster*. All groups split up the work again, with each user taking a job posting and finding related media and information.

### 6.2.2 Display Space Organization

The main reason behind dividing the task up into two different parts was to study behavior of a group when members are working individually and when they are working collaboratively, and how this switching between two different modes takes place.

#### 6.2.2.1 Part One – Individual Web Searching and Data Gathering

There were essentially two distinct approaches employed by all the groups: *explicit* and *implicit* separation of workspace. Three of the six groups decided to explicitly define personal workspaces by splitting the wall into three equal parts using

sections, each part corresponding to the user that was sitting in front of that section. From that point on, all job postings found by a certain participant went to their personal section. This allowed them to maintain any window layout they preferred without interfering with their co-workers. For instance, in Figure 31 the middle user had all his windows tiled while the user on the right had all his windows piled on top of one another.

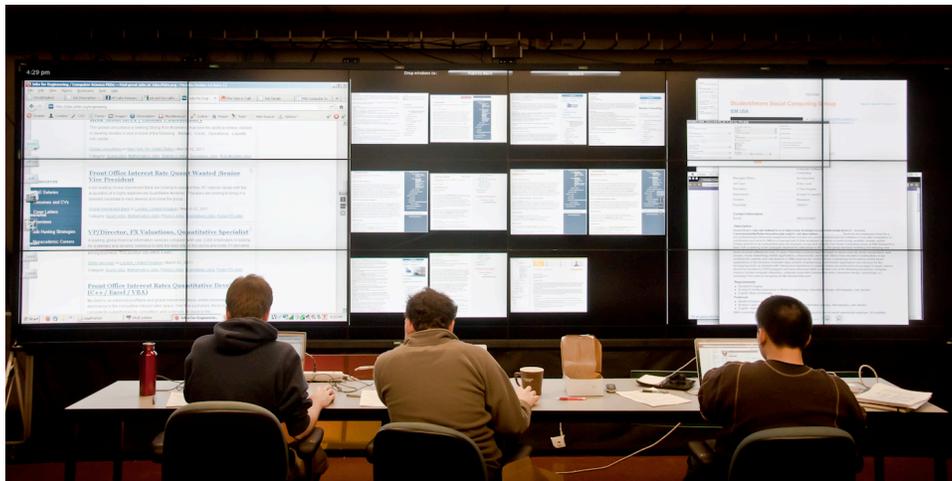


Figure 31. Sections explicitly defined personal workspaces and allowed participants to employ individual organization strategies.

The other three groups simply started working by putting job postings on the wall as they found them. Every user still had their personal space on the wall, however this space was implicit and loosely defined. The separation between each user's space was now denoted by physical distance between their collection of windows. In this case, the positioning of the users' windows was loosely related to their seating position. Such implicit organization had several other consequences. First, to make the separation and ownership of space evident, there had to be an obvious distance between the groups of windows from different users (Figure 32), much more so than when sections were used

since the section splitter provided a clear separation clue. Simply speaking, this distance requirement meant that more display space was wasted. The second consequence of not using sections was the inability to use the tiling algorithm, so users often resorted to manual window alignment (Figure 32). Lastly, given enough windows, users would eventually intrude on each other's territory, which meant they would have to re-negotiate the use of space. This happened with group 5 which eventually resorted to using sections to provide each user with their own workspace.



Figure 32. Implicitly defined personal workspaces loosely correspond to users' seating positions, require more physical distance for clarity, and necessitate manual window organization.

Interestingly, no group decided to keep all their findings together. After all, the first part of the task was to *collect* job postings for the second part, which required them to filter all the findings *as a group* (meaning, the data will be aggregated anyway). Moreover, all groups decided to split the task by job searching website, rather than career interests, even though their chosen strategy was, in theory, more likely to result in duplicates. Keeping their findings separate allowed each participant to focus their

attention only on their own work, which meant that duplicates weren't identified until the group started filtering through the data in part two. In practice, however, this proved to be an insignificant problem since less than 10 duplicates were found across all groups.

This desire for clear separation further strengthens the point that when given a choice, users prefer taking their own approach to the problem. Although taking individual approaches to solving a problem isn't new, users are now able to extend this behavior to personalizing a portion of *shared* space, given that the space is easily re-configurable.

#### ***6.2.2.2 Part Two – Collaborative Analysis and Filtering***

In part one, some users decided to keep their findings on their laptop, typically as open tabs in a web browser, instead of posting them on the wall. Since part two of the task required them to collaboratively filter the collected information, they quickly realized the benefit of having all the postings on the wall and decided to share their findings as well. Displaying all of the job postings on the wall externalized the group's working memory and therefore reduced the users' cognitive load because every window on the wall embodied multiple pieces of information, eliminating the need for memorization. Primarily, each window was a document itself, listing the actual job posting. However, it also acted as a proxy for the supporting information. For instance, window position described the current state of the job posting in the filtering process ("keep", "maybe" or "not yet reviewed"). This made it easy to directly compare similar postings, to keep track of the progress or find duplicates. This was another example of distributed cognition, also observed by Andrews et al. (Andrews et al. 2010).

Once all job postings were on the wall, every group took a moment to rearrange the display space. In the first part of the study, grouping of windows was used to denote

ownership (through sections or physical separation). In the second part, grouping was used for categorization, with the categories being “keep”, “maybe” and “not yet reviewed” (rejected job postings were simply removed). Although every group had the same categories, they again used implicit or explicit grouping (sections or physical separation) to denote categories. Interestingly, group 3 used sections for the first part and spatial grouping for the second part, while group 6 did the opposite. No matter which organization approach was chosen, the categories “keep” and “not yet reviewed” were always completely separate, which makes sense since those were the two most important categories. On the other hand, the category “maybe” was considered as a temporary one and therefore didn’t receive its own section. Instead, it was typically kept as a pile on the side somewhere. This was an indication that users needed some “scratch space” to temporarily place items in, similar to scratch paper we commonly use to write down short-lived ideas.

During the filtering process, most groups would evaluate one job posting at a time by bringing its window into focus, either by resizing it manually or using the maximize feature to enlarge it. Quite often another piece of information was needed to effectively make a decision, from either the web or another job posting which was similar to the one currently being examined. To display this supporting information, users typically made use of spatial memory, recalling where this information was last seen on the display and bringing it into focus. The desired layout would be all relevant windows juxtaposed on top of all the categorized windows underneath (Figure 33). This presented a technical problem in groups that used sections for categorization since sections were designed to provide structure by automatically fitting a window inside if its center happens to fall

over that section. This automatic re-arrangement of windows resulted in user frustration. What users really desired was temporary freedom to lay windows out manually in a layer above all the sections. This need for *structure plus freedom* was repeatedly observed in all the groups and later confirmed in the post-study interviews.

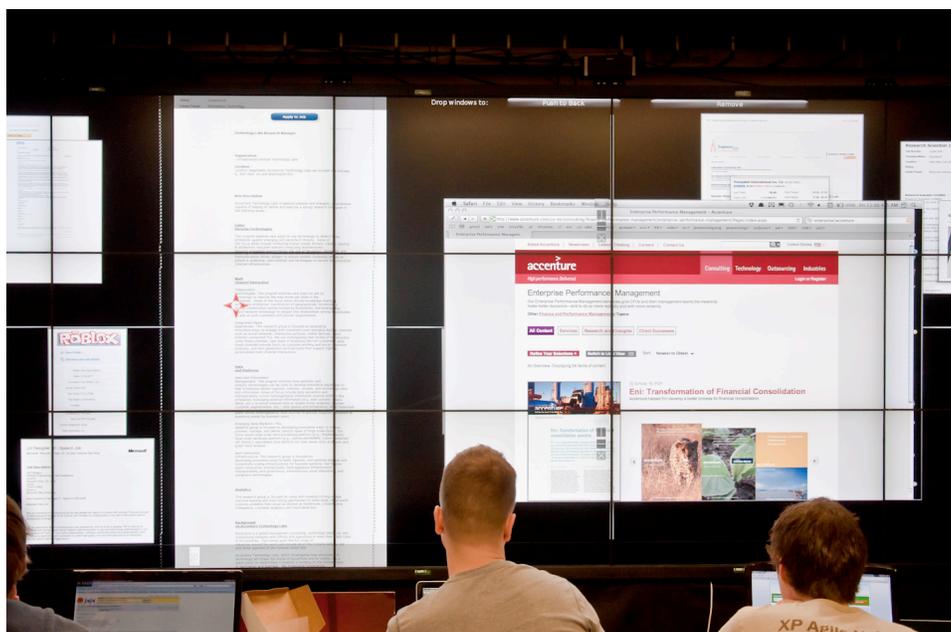


Figure 33. Typical window layout during filtering stage consisted of the job posting and the supporting information in focus on top of the structured layout underneath.

Another interesting behavior observed in all the groups was “*stop and organize*” which means briefly pausing the filtering process to organize the workspace. This organization could be as insignificant as repositioning a few windows inside a category or completely rearranging the whole workspace. This occurred either when a certain region became too cluttered, or the space went out of balance as windows were moved from one section to the other during filtering. Small changes to the organization were performed by individuals without consulting other group members while bigger changes

would be initiated by individual participants, but discussed with the rest of the group before being carried out. This was especially apparent with group 4 which took an iterative approach to filtering, possibly because they had the largest number of job postings. Prior to each iteration, they would completely re-arrange the sections and windows in order to give emphasis to the job postings still to be evaluated.

These behaviors identify space organization as an important part of the workflow on large high-resolution displays, which is a notable departure from the typical desktop environments where low resolution doesn't necessitate significant space organization. A positive side effect of frequent organization is that it gives users an opportunity to take a step back, re-evaluate and clarify their focus and direction.

### **6.2.3 Fundamental Organization Techniques**

Previously, we discussed the general layout of the wall, however at any point in time the users employed several fundamental organization techniques. We list the most commonly used ones.

#### ***6.2.3.1 Tiling***

Virtually every group used the automatic tiling feature since it provides a quick way to lay out all windows without overlap. No overlap meant the effect of external memory was maximized since every window was fully visible. However, there were several issues associated with automatic tiling.

The tiling algorithm itself is not optimal because the size of each cell depends on the average window aspect ratio, the aspect ratio of the section and on the number of

windows. This works best if all windows have a similar aspect ratio (Figure 31). On the other hand, given windows of varying aspect ratios, tiling tends to yield a lot of empty space between the windows. To get around this drawback, users sometimes split the current section into two and divided the windows between the sections based on their aspect ratios (Figure 34). Other times, they would simply resize the section several times, hoping to manually optimize the tiling algorithm, which often yielded successful results without much work. However, making one section smaller would automatically enlarge neighboring sections, which could alter their tiling arrangement. This would obviously present a problem when sections are owned by different users. Although never explicitly mentioned during the training, this effect was understood by the users and therefore section resizing was only performed in part two of the task where consequences did not directly affect other participants.



Figure 34. To get around the limitation of the automatic tiling algorithm and maximize space usage, users would sometimes group windows of similar aspect ratios into separate sections.

The question of relative order of windows within a tiled section is also of importance. Ideally, we would like to minimize wild re-arrangements of windows since

that could invalidate users' spatial memory. The current algorithm orders the windows based on the time each window spent on the wall, oldest being first in the grid. This makes sense when new information is added to the wall since the windows are placed at the end of the grid. However, in part two of the task, the age of windows was not of importance anymore. In fact it was forgotten, which would occasionally result in unintentional re-arrangements that users did not understand. They quickly worked around this by turning the tiling feature off once windows were tiled, preventing further automatic re-arrangements.

Ultimately, tiling was a much-appreciated feature and its benefits outweighed the drawbacks. Perhaps it becomes even imperative given a large number of windows because manual organization becomes too tedious. This was especially evident with group 4, which had the most windows and coincidentally made extensive use of the tiling feature, even trying to optimize its effectiveness to make the best use of space.

#### ***6.2.3.2 Piling***

Another commonly used organization approach was piling, or rough grouping of windows with some overlap. Throughout the study, there were several reasons why users resorted to piling. It was used to denote ownership of windows, to associate one or more items, or to create more free space.

While piling is very natural, in our case the technology needed improvement to support this. For instance, finding specific items in a pile required many manual window manipulations and potentially some free space. This is somewhat similar to the problem of switching windows in any contemporary operating system, albeit a multi-user one with multiple piles which complicates the problem significantly. Moving a whole pile is

another issue. Although users could simply drag-select the whole pile and move it as a group, many simply forgot that this was possible and therefore ended up moving every window manually. Perhaps this is a clue that users were treating documents as windows from a typical desktop, which cannot be selected and moved as a group. This is a part of the learning curve associated with these environments of notably different affordances. Lastly, what identifies a pile is the top-most window. This however was difficult to maintain since the piles typically started with the most important window, the job posting itself, and therefore adding any supporting information to the pile would cover up the posting. Clearly, more advanced form of grouping support is needed.

#### ***6.2.3.3 Aligning***

Aligning refers to manually positioning a window close to another one while leaving a small amount of empty space to visually separate the two. Mostly, this behavior was observed in the first part where individuals could keep to their own organization strategies and the number of windows per user was lower. Perhaps this behavior emphasizes the fact that each individual brings their personal preferences to the group and would like to adhere to them if possible (e.g. neatness). Manual alignment was primarily done to achieve a cleaner organization, but another side effect was better space utilization since manually arranging windows yielded closer packing than the automatic tiling could, especially in cases with varying window aspect ratios. It would be interesting to explore whether cleaner display space organization or aesthetics leads to better performance or understanding of information.

Although users didn't seem to mind the extra effort required to align windows neatly, the question arises how well this approach scales to a large number of windows. It

is probably not a coincidence that group 4, which had the largest number of windows, didn't align windows manually. Regardless, the system could assist this behavior with either smarter tiling algorithms or even some simple snapping features.

#### **6.2.4 User Interaction and Control**

SAGE supports truly multi-user, multi-modal interaction, which provided a unique environment for studying control coordination among participants. Every user embraced this affordance naturally and took full advantage of it. Each user's presence on the wall was denoted by a cursor, which could have a user-defined color and a label. Although not every user picked a color for their cursor, they did assign a label, which proved sufficient in differentiating cursors among participants.

##### ***6.2.4.1 Sense of Ownership***

SAGE imposes no ownership restrictions, which meant that any user is *allowed* to control anything on the display at any point in time. Having control from their laptops meant they were also *able* to control anything on the display, since all windows on the display were reachable and had the same orientation for all users (unlike tabletop displays).

During part one, there seemed to be an implicitly defined ownership of information since users were not touching each other's windows or personal space. Exceptions to this were accidents when a user mistook somebody else's window for his. In this case, they would even apologize for this mistake and correct it by returning the

window where they found it. This behavior was independent of whether the group used sections or simply spatial distance to indicate personal space.

In part two, although every user would present his/her own findings, the windows weren't always strictly controlled by the user who was presenting (i.e. the user who found the job posting). In fact, one group even decided to have a single user control the display during the filtering process. This agreement didn't last very long however. Very quickly, other users wanted to express their opinions or point things out which meant they instantly, and mostly subconsciously, grabbed their own cursor to accomplish what they intended. Generally, it was easier to accomplish smaller tasks yourself than to explain to others what you wanted. This means that there has to be a very low barrier to entry for taking control of the display. Although there were some usability issues with switching control between laptops and SAGE, for the most part users found the control easy enough to use it continuously in such manner.

Although a sense of ownership may be important in a more professional setting, we believe implicit social protocols may be enough to enforce this ownership. This further supports our prior observations from the weekly meetings.

#### ***6.2.4.2 Division of Tasks***

Occasionally, during the filtering process there was a need for further information before the decision could be made. Often, individuals would volunteer to find this information while others were still reviewing the job posting in question. However, at times, tasks were specifically dispatched by one user, either to a specific individual (e.g. "Could you find it on the map?"), or to the group as a whole expecting somebody to take the responsibility for the task (e.g. "Can we Google it?"). In either case, the supporting

information was consistently shared on the wall as opposed to leaning over or turning laptop screens towards others. This indicated that users naturally adopted the wall as a shared medium and that they found it relatively easy to share the information.

A degree of parallelism also occurred with organization. While two users were reviewing the materials (manipulating the focus), one user could be cleaning up the wall (manipulating the context). This never resulted in large reorganizations of the space since most of the space was taken up by the windows in focus and because individual participants would not undertake a larger restructuring without first consulting the rest of the group.

#### ***6.2.4.3 Physical and Virtual Navigation***

Every group except for group 3 exclusively used laptops and virtual navigation for controlling the wall. Group 3 however, performed the bulk of the filtering process on the wall, using the touch screen. Upon gathering all the job postings on the wall, they eliminated all the sections and decided to individually review each posting (Figure 35). However, instead of enlarging it, the user that was physically closest to the window, would use a combination of physical and virtual navigation to bring it to the group (i.e. walking and dragging the window closer to the center of the group). Windows were rarely maximized or even enlarged much beyond what was necessary for the group to be able to read it. Such behaviors were a clear indication that physical navigation was more natural than virtual navigation, even if it demanded more physical work [3]. In the post-study interview, members of this group noted that while using the touch screen felt slower, it was much more natural. Interestingly, this group was actually the fastest in narrowing down their findings to the top five while having the second largest number of

job postings (28)! The average time for filtering was 30 minutes while this group finished in 16 minutes. This group even did more comparisons among individual jobs during the filtering process (however, this was only observed, not quantified). This is possibly a hint that the quality of analysis improves with physical navigation. Perhaps this is another proof that physical navigation is more natural since it was easier to walk to another window than having to manipulate multiple items into focus, especially considering the virtual navigation learning curve.

During the interview, most groups expressed curiosity about the touch interaction although they mentioned doubt about its usefulness as the primary reason for not trying it. Although these observations hint at the benefits of touch interaction, more work is needed to quantify these benefits or determine which input modality is better suited for which type of tasks. A longer study would also be required to investigate the effects of comfort and fatigue on the choice of input modality.



Figure 35. Physical navigation proved more natural and required minimal window manipulations.

### 6.2.5 Awareness / Implicit Learning

Large high-resolution displays are particularly well suited for supporting awareness since they offer a single large display surface visible to all participants in the group. We have observed several instances of users benefitting from increased awareness, primarily during the first part of the study when users worked separately.

Almost universally, participants had their personal desktops shared on the wall during search. On several occasions, participants adopted their group member's search websites and strategies by simply glancing onto the wall and noticing the web page they were on. Similarly, one user decided to highlight the important bits in the web page before taking the screenshot and sharing it on the wall. Highlights persisted in the screenshot and made it easier to later quickly recall what this job posting was about. His group member noticed this and immediately adopted the technique without ever mentioning anything. During the interview however, she felt slightly embarrassed at admitting this behavior. This means that supporting awareness on a large shared surface has potential to also reduce social awkwardness by allowing users to glance at each other's progress without physically leaning over to see their laptop screen.

Besides adopting each other's search techniques, group members adopted also adopted organization strategies. In group 6, the middle user was neatly aligning his job postings on the wall. Having noticed that, the other two group members decided to align theirs similarly, without ever mentioning anything explicitly (Figure 32). The reason behind this behavior may have been of social nature rather than purely practical, since users with overlapping windows perhaps felt that they needed to contribute to the overall "cleanliness" of the display space, especially since this group did not use sections to

separate individual workspaces. On the other hand, a user in a group that did use sections, decided to manually align her windows neatly while others kept their own organization strategies. Perhaps simply having separate workspaces also affects the social dynamics within a group, possibly distancing the members from each other. However, a more focused study would be required to investigate this effect further.

### **6.3 Design Implications – From User Feedback**

Even though some users in the beginning were skeptical about the advantages of a wall display in this task, in the end, impressions and feedback from all the groups was very positive. They stated that the ability to simultaneously interact with and view all the information at once helped immensely in this collaboration. Since the study, we have even observed increased use of the wall during regular lab activities such as classes and meetings. However, every group also noted some suggestions for improvement. Although sometimes conflicting, most of the suggestions revolved around usability issues of SAGE. They can broadly be summarized into several categories:

#### **6.3.1 Improved Feedback and Awareness**

Given such a large working surface, it is often difficult to make note of all the activity on the wall and therefore it would be useful to animate changes. Besides the obvious aesthetic benefits, this would have the potential to increase awareness since human peripheral vision is naturally sensitive to motion. For instance, users noted that it would have been helpful to animate the tiling process in order to more easily keep track

of automatic rearrangements and minimize the effect on spatial memory. Similarly, one could animate new windows showing up on the display.

### **6.3.2 The Learning Curve**

The learning curve does not necessarily refer to learning to use the tools, but rather remembering all the capabilities and understanding how it could help them manage the display space. All users reported being more comfortable with the technology after the study and having a better idea of how to organize information with so much space available. Most claimed they would have paid more attention to organization and made more use of sections and tiling features if they had to do it again. Additionally, virtually every group overlooked a certain available feature, which could have simplified some actions or even result in new task strategy.

### **6.3.3 Easier Control Switching**

Although the “extended desktop” metaphor for switching control from laptops to SAGE made sense, users were still sometimes confused about which display they were currently controlling. This was mostly due to technical reasons such as insufficient feedback and responsiveness.

### **6.3.4 Annotation**

On a really large surface with many windows, identifying windows becomes difficult especially when documents are visually similar as in our case (primarily white, text documents). Users quoted annotation and editing on the wall as much needed

features to mitigate this problem. This would also provide the means for finer classification of information beyond the spatial categorization.

## **6.4 Design Implications – From Observations**

Aside from the user feedback, we have abstracted several higher-level design goals to better support collaborative work on wall displays.

### **6.4.1 Individual Workspaces**

Participants seemed to embrace the ability to organize their own display space without interfering with others. We believe that if we allow users to exercise their individual preferences instead of forcing structure on them, they are more likely to adopt large high-resolution displays as a collaborative medium. We have observed a similar behavior during our weekly meetings when users wanted to display a large number of images by organizing it in their own workspace, without interfering with others.

### **6.4.2 Structure + Freedom**

Managing a large number of windows manually proved tedious for many users, which encouraged them to use sections and automatic tiling features. Although sections proved very useful in structuring the environment, they frequently got in the way of accomplishing simple, mostly temporary, tasks. This led to user frustration and diminished the benefit of sections; occasionally prompting users to remove sections altogether in order to gain complete freedom of interaction. Perhaps a layer of unrestricted window layout could be superimposed on top of a structured layout to allow

a degree of freedom. The art seminar presentation and the Image of Research judging scenario called for a similar functionality.

### **6.4.3 Allow Transient Organization**

Even in a task as simple as this one, users sometimes attempted various layouts that better suit the current subtask. In a more complicated analytics task, gaining new insights about data could greatly depend on your ability to represent it in many different forms. Therefore, reorganizing information quickly and switching between different layouts would be crucial.

### **6.4.4 Undo**

Since organization is now a task in itself, users will inevitably make mistakes. Or even worse, the system will make mistakes if it attempts to be intelligent. Undoing previous action or a set of actions is imperative. However, limiting the feature to undoing only the last window manipulation is not sufficient. For instance, during automatic tiling, dozens of window manipulations could occur in an instant. Perhaps a timeline approach would be more appropriate. Additionally, users might have many organizational structures on the display at once, which means it should be possible to undo actions selectively on certain portions of the display. Given a large number of windows and difficulty in identifying them, windows get closed by accident, which should be reversible as well.

#### **6.4.5 Assist With Organization**

Considering that users sometimes forget to organize until clutter becomes obvious, the system could continuously monitor the space, learn about users actions and detect the state of clutter. It could either remind users to organize or even offer suggestions based on the gathered data. Defining metrics for identifying the state of clutter would depend on the use case, but could include the amount of overlap in windows, the number of windows, window sizes, amount of empty space and possibly the number of interactions.

## 7. CONCLUSION AND FUTURE WORK

The traditional war rooms have the potential to significantly increase the productivity of teams through radical collocation. Walls covered with paper artifacts are the primary tools that foster social interactions in such environments. The ultimate goal of this work was to begin a shift towards digital war rooms using wall displays that enhance traditional paper-based environments with virtual interaction and reconfigurable space for collaboratively managing large amounts of visual information. Therefore, this dissertation sought to understand the user interaction and display space organization behaviors in collaborative large, high-resolution display environments and investigate how to best take advantage of these affordances. To enable investigation in the context of real-world scenarios, a multi-user, multi-modal interaction framework was developed along with distributed, scalable user interface widgets. The framework, tailored for wall displays, provided the necessary development flexibility during iterative design of the interaction and display organization techniques. Following the three iterative design phases, the techniques were evaluated in a collaborative analysis task resembling a digital war room environment. A summary of the most important techniques and behaviors is presented below along with the discussion of how they complement the traditional war room environments. The chapter is concluded with ideas for future research directions.

### **7.1 Digital War Rooms**

As mentioned previously, (Teasley et al. 2000) identified four fundamental affordances of war rooms responsible for productivity increase in teams. They were listed

as constant awareness and implicit learning, easy transitions in and out of spontaneous meetings, immediate access to information and its persistence. Substituting paper-based walls with large, high-resolution displays introduces two additional affordances, among others. First, digital walls enable virtual interaction with the artifacts, enriching interaction opportunities through various interaction zones. Second, digital space is easily reconfigurable, unlike walls covered with paper artifacts, which allows for more elaborate and dynamic display organizations. Combining these affordances begins to paint the picture of digital war rooms (Figure 36). The most important techniques in interaction and display space organization are restated below, along with how they complement the traditional war rooms.

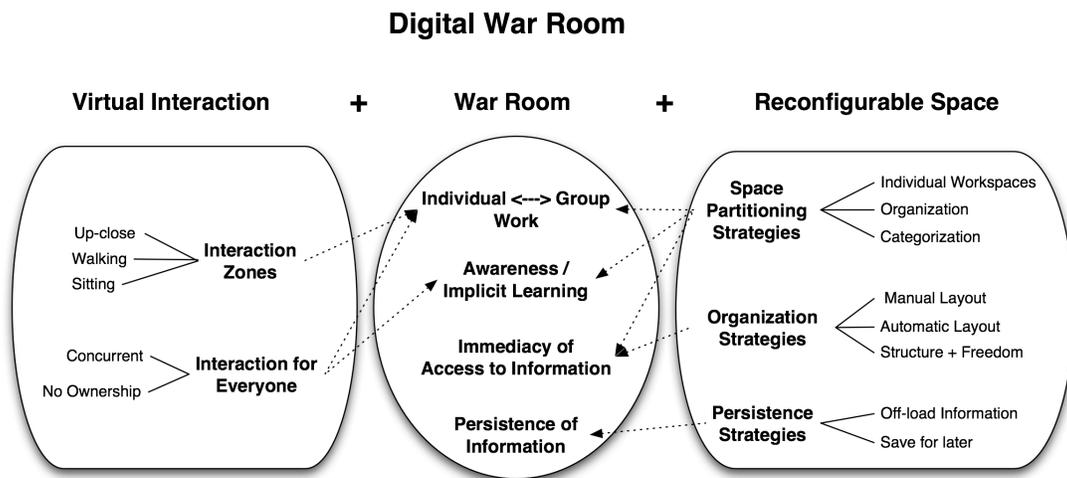


Figure 36. Digital war rooms are a combination of traditional war rooms and large, high-resolution displays, which provide the benefits of virtual interaction and reconfigurable space. The arrows outline how the techniques serve to support the traditional war room affordances.

### 7.1.1 Physical Interaction Zones and Multi-modal Interaction

Different use cases call for different interaction zones, or areas where the users are physically located in front of the display during work. From the various observed uses cases, three frequently used interaction zones were identified: up-close, distance, sitting. Figure 37 illustrates the zones while the description of each is given below.

#### **Up-close**

Interaction directly on the wall is referred to as up-close interaction (e.g. using a touch screen). This modality takes advantage of physical navigation, which reduces dependence on virtual navigation and results in more natural interaction, much like in traditional war rooms with paper-based artifacts. It may also be more appropriate for localized interaction and focusing on details. However, the disadvantage is lack of comfort and therefore it may be unreasonable to expect users to work in this mode for prolonged periods of time (Beynon 2001). Additionally, it may require special attention for very large displays since not all areas will be within users' reach.

#### **Distance**

This zone affords the greatest mobility in front of the wall, taking advantage of physical navigation. Users also have an option of sitting down or walking up to the display while still being able to interact with the wall. This mode may be most appropriate for making sense out of information (e.g. seeing the overview of the data). In traditional war rooms, users would frequently have to move to the wall for re-arranging the artifacts, and stand back from the wall in order to see the bigger picture. Virtual interaction from a distance therefore has the potential to enhance this aspect of war rooms. However, this freedom currently seems to come at a cost, either in intuitiveness,

accuracy, speed or the number of users supported simultaneously. It is yet to be seen whether any interaction technique or device can satisfy all these requirements simultaneously.

### **Sitting**

This zone is preferred during longer meetings where comfort and private/public work modes are desired. If every user has a laptop, it provides for easy switching between individual and collaborative work, which has been found to be a crucial aspect of war rooms. In fact, it offers more than traditional war rooms since laptops can be used for enabling interaction for every single participant at no cost. However, this mode does not naturally allow for physical interaction, which means fast methods for virtual navigation are imperative.

Most use cases will ideally span across two interaction zones, some even all three. Although in practice it would be difficult to bridge all the interaction zones using a single interaction modality, one approach is to make use of two input modalities. As observed in the classroom scenario, making use of two modalities simultaneously is not as confusing as it may seem, as long as one of them is a natural interaction method (e.g. touch, hand gestures).

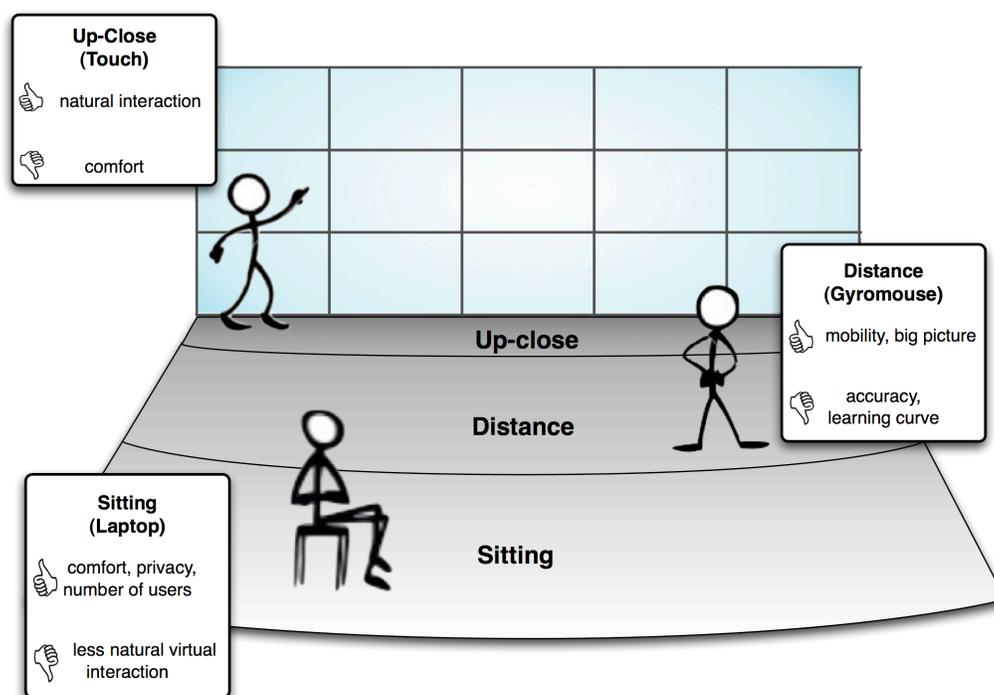


Figure 37. The three commonly used interaction zones in wall display environments, each with their own benefits and drawbacks.

### 7.1.2 Interaction for Everyone

Although some systems are collaborative in the sense that they allow multiple users to interact with the system, the greatest benefit comes from the ability to interact with the system *simultaneously*. As we have observed in the collaborative analysis study, turn taking may still take place and even be agreed upon explicitly. However, even after agreeing on turn taking, the users appreciated the ability to gain control and provide input at any point in time. Generally, empowering every user with control minimizes social barriers for participation by eliminating the need to ask for control or interfering with

other users. This in turn promotes discussion, fosters collaboration and allows for easy transitions between individual and group work often exhibited in war rooms. Additionally, individual control also facilitates task division among group members, potentially benefitting the group performance through a greater degree of parallelism and therefore increasing awareness and opportunities for implicit learning.

### **7.1.3 Partitioning the Display Space**

As display size and resolution increases, the position of information on the display encodes additional meanings, raising the need to partition the display space for supporting such behaviors. Several reasons for partitioning the space have been observed: categorization of information, the need for individual workspaces and supporting various organizational structures. Display space can be partitioned explicitly (e.g. using sections) or implicitly (e.g. using spatial grouping). Benefits to explicitly partitioning the space include greater freedom in various organization strategies, easier reconfiguration of space and better support for individual workspaces. Individual workspaces serve to directly support seamless transitions between individual and group work, and therefore awareness as well, since information used by individuals in their tasks is always present on the wall for everyone to see. Using sections for maintaining various organization strategies is usually employed for increasing information visibility through the use of selective tiling. Increased visibility equates to increased immediacy of access to information, which has been proved essential in war rooms.

#### **7.1.4 Organization Strategies**

As prior work in wall displays has pointed out, large screen real estate results in numerous window manipulations as well as frequent careful window alignments. To assist such behaviors, several enhanced manual window manipulation techniques have been developed. Nevertheless, manually managing a large number of windows proved tedious for many users, which motivated the development of automated organization features such as tiling. Tiling organizes windows in a grid without overlap, which has the benefit of increasing the visibility of information. This again complements the traditional war rooms as it allows immediate access to information.

Sections provide another form of automated organization. Although they proved very useful in structuring the environment, they frequently got in the way of accomplishing simple, mostly temporary, organizational tasks. This led to user frustration and diminished the benefit of sections; occasionally prompting removal of sections altogether for gaining complete freedom of interaction. Therefore, automatic organization techniques, while necessary, must allow for manual window adjustments (i.e. offer structure while providing the freedom in organization).

#### **7.1.5 Persistence**

One of the observed organization behaviors is topic switching, or in other words, displaying new information loosely related to the currently displayed one. There are two ways in which this typically occurred. More commonly, users simply started displaying new information on top of the old one, usually resulting in clutter. In other cases, users first removed or minimized all the windows (or windows within a certain section) and

subsequently displayed the new information. On the other hand, saving the state of the whole display was used for supporting long-term use. This allows the group to revisit prior stages of their work or even time-share the war room with other groups. This kind of dynamic use clearly isn't practical in paper-based environments.

## **7.2 Future Directions**

The work presented in the document offers several tools and techniques for collaboratively managing numerous pieces of information in digital war rooms. However, it has made no attempt at trying to quantify the actual benefits of wall displays equipped with these techniques versus without them. Furthermore, it is still unknown as to what extent digital war rooms enhance information-intensive work when compared to traditional war room environments. The general benefit of digital information and environments has already been taken for granted, nevertheless, at times the ease of use and the tangible qualities of paper artifacts are preferred despite their shortcomings. The affordances of paper are perhaps of even greater significance since traditional war rooms are primarily paper-based. Although there is a substantial amount of evidence highlighting the benefits of wall displays for organizing large amounts of digital information despite the lack of the natural, paper-based interaction, a more comprehensive investigation is needed to directly compare the traditional and digital war rooms.

As mentioned previously, virtual interaction and reconfigurable space are just two of the many benefits digital war rooms have over paper-based environments. For instance, easy sharing, dynamic update and easy retrieval of information are all

affordances of digital spaces, which have not been discussed here. Although some support for information sharing has been enabled in SAGE, the ultimate goal is to share information with the same ease as writing and posting paper notes on a physical wall. While paper artifacts are difficult to modify after creation, digital information can be updated dynamically, sometimes completely changing representations or content. This will necessitate appropriate techniques modifying content on the wall and for notifying users of the changes in the content, especially in the long-term, collaborative use cases that can result in asynchronous updates. Digital artifacts also enable retrieval of information for individual work outside the war room, which is difficult in collaborative paper-based environments. However, given that the relative organization of digital artifacts encodes additional meaning, that also presents challenges when moving information to smaller screens for work outside the war room.

Although, the interaction framework has been designed with geographically distributed work in mind, such use cases have not been evaluated in the current scope of this work. Undeniably however, there is a significant push for distributed group work as the financial and environmental burdens of travel are increasing. Much like collaborative work has added a new dimension to the complexity of single-user environments, distributed work adds a new complexity dimension to the collaborative work. Sharing information across multiple sites, effective communication, group dynamics, privacy issues and asynchronous work will all become issues needing attention before these display-rich environments can become an equally functional alternative to radical collocation.

Ultimately, the question is how to use wall display environments for collaboratively addressing the problems of scale and complexity. The work in this document focused on the display space organization and virtual interaction while some of the additional affordances have also been mentioned. Providing tools for enabling these affordances, and discovering new ones, has to be followed by iterative evaluation and improvement in the specific context in order to create an environment that lends itself for collaboratively managing the problems of scale and complexity.

### **7.3 Summary**

The presented techniques, although far from constituting a complete set, offer direction for future investigation of wall displays in war room environments. Furthermore, while the developed features clearly had space for usability improvements, even as such, they transformed the collaborative use cases at EVL to the point where it is naturally and readily adopted as a shared display medium for facilitating group work. This is perhaps a proof that this work has set off in the right direction as well as a real-world testament to the benefits of wall displays in collaboratively managing numerous pieces of information. As the technology needed for building such environments becomes more affordable, we will likely see wall displays permeate a broad range of everyday use cases that will give further importance to the related human-factors issues.

## CITED LITERATURE

Ahlborn, Benjamin, David Thompson, Oliver Kreylos, Bernd Hamann, and Oliver Stadt. 2005. "A practical system for laser pointer interaction on large displays." *VRST '05: Proceedings of the ACM symposium on Virtual reality software and technology* (November 1).

Andrews, Christopher, Alex Endert, and Chris North. 2010. "Space to think: large high-resolution displays for sensemaking." *CHI '10: Proceedings of the 28th international conference on Human factors in computing systems* (April 1).

PQ Labs, <http://multi-touch-screen.com/>.

Qt, <http://qt.nokia.com/>.

wxWidgets, <http://www.wxwidgets.org/>.

Badros, Greg J, Jeffrey Nichols, and Alan Borning. 2001. Scwm: An Extensible Constraint-Enabled Window Manager. In *Proceedings of the FREENIX Track: 2001 USENIX Annual Technical Conference*. USENIX Association, June.

Ball, Robert, Michael Varghese, Bill Carstensen, E Dana Cox, Chris Fierer, Matthew Peterson, and Chris North. 2005. "Evaluating the Benefits of Tiled Displays for Navigating Maps." *Human-Computer Interaction - INTERACT*.

Ball, Robert, Chris North, and Doug A Bowman. 2007. Move to improve: promoting physical navigation to increase user performance with large displays. In *CHI '07: Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM Request Permissions, April. doi:10.1145/1240624.1240656.

Ball, Robert, and Chris North. 2005. "Analysis of User Behavior on High-Resolution Tiled Displays." *Human-Computer Interaction - Interact 2005*.

Ball, Robert, and Chris North. 2008. "The effects of peripheral vision and physical navigation on large scale visualization." *GI '08: Proceedings of graphics interface 2008* (May 1).

Baudisch, Patrick, Edward Cutrell, and George Robertson. 2003. "High-density cursor: a visualization technique that helps users keep track of fast-moving mouse cursors." *Human-Computer Interaction - INTERACT*.

Baudisch, Patrick, E Cutrell, K Hinckley, and R Gruen. 2004. "Mouse ether: accelerating the acquisition of targets across multi-monitor displays." *Conference on Human Factors in Computing Systems*.

Baudisch, Patrick, Mike Sinclair, and Andrew Wilson. 2006. "Soap: a pointing device

- that works in mid-air.” *Proceedings of the ACM symposium on User interface software and technology*.
- Bell, Blaine, and Steven Feiner. 2000. Dynamic space management for user interfaces. In *UIST '00: Proceedings of the 13th annual ACM symposium on User interface software and technology*. ACM Request Permissions, November. doi:10.1145/354401.354790.
- Beynon, C. 2001. “Spinal shrinkage during a seated break and standing break during simulated nursing tasks.” *Applied Ergonomics* 32 (6) (December): 617–622. doi:10.1016/S0003-6870(01)00038-2.
- Bezerianos, Anastasia. 2007. “Using alternative views for layout, comparison and context switching tasks in wall displays.” *Human-Computer Interaction - INTERACT* (November 1).
- Bezerianos, Anastasia, and Ravin Balakrishnan. 2005. “The vacuum: facilitating the manipulation of distant objects.” *Conference on Human Factors in Computing Systems*.
- Bi, Xiaojun, and Ravin Balakrishnan. 2009. “Comparing usage of a large high-resolution display to single or dual desktop displays for daily work.” *CHI '09: Proceedings of the 27th international conference on Human factors in computing systems*: 1005–1014. doi:http://doi.acm.org/10.1145/1518701.1518855.
- Birnholtz, Jeremy P, Tovi Grossman, Clarissa Mak, and Ravin Balakrishnan. 2007. An exploratory study of input configuration and group process in a negotiation task using a large display. In *CHI '07: Proceedings of the SIGCHI conference on Human factors in computing systems*, 91–100. CHI '07. New York, NY, USA: ACM. doi:http://doi.acm.org/10.1145/1240624.1240638.
- Blanch, R, Y Guiard, and M Beaudouin-Lafon. 2004. “Semantic pointing: improving target acquisition with control-display ratio adaptation.” *Proceedings of the 2004 conference on Human factors in computing systems*.
- Booker, John, Timothy Buennemeyer, Andrew Sabri, and Chris North. 2007. “High-resolution displays enhancing geo-temporal data visualizations.” *ACM-SE 45: Proceedings of the 45th annual southeast regional conference* (March 1).
- Buxton, W, G Fitzmaurice, and R Balakrishnan. 2000. “Large displays in automotive design.” *Computer Graphics and Applications*.
- Cao, Xiang, and Ravin Balakrishnan. 2003. VisionWand: interaction techniques for large displays using a passive wand tracked in 3D. In *UIST '03: Proceedings of the 16th annual ACM symposium on User interface software and technology*. ACM Request Permissions, November. doi:10.1145/964696.964716.
- Czerwinski, Mary, Greg Smith, Tim Regan, Brian Meyers, George Robertson, and Gary

- Starkweather. 2003. "Toward characterizing the productivity benefits of very large displays." *Human-Computer Interaction - INTERACT*.
- Czerwinski, Mary, George Robertson, Brian Meyers, Greg Smith, Daniel Robbins, and Desney Tan. 2006. "Large display research overview." *CHI '06: CHI '06 extended abstracts on Human factors in computing systems* (April). doi:10.1145/1125451.1125471.
- DeFanti, Thomas A, Jason Leigh, Luc Renambot, Byungil Jeong, Alan Verlo, Lance Long, Maxine Brown, et al. 2009. "The OptiPortal, a scalable visualization, storage, and computing interface device for the OptiPuter." *Future Generation Computer Systems* 25 (2) (February 1): 114–123. doi:10.1016/j.future.2008.06.016.
- Fitzmaurice, G, A Khan, G Kurtenbach, and G Binks. 2005. "Cinematic meeting facilities using large displays." *Computer Graphics and Applications*.
- Forbrig, P, Fabio Paternò, and Annelise Mark Pejtersen. 2008. "Multitouch Sensing for Collaborative Interactive Walls." *Proceedings of the 1st Tc 13 Human-Computer Interaction Symposium (Hcis 2008), September 7-10, 2008, Milano, Italy* 272. doi:10.1007/978-0-387-09678-0.
- Forlines, Clifton, Daniel Vogel, and Ravin Balakrishnan. 2006. HybridPointing: fluid switching between absolute and relative pointing with a direct input device. In *UIST '06: Proceedings of the 19th annual ACM symposium on User interface software and technology*. ACM Request Permissions, October. doi:10.1145/1166253.1166286.
- Foster, Christopher, Liz Burd, and Andrew Hatch. 2010. Exploring concepts collaboratively: considering how Wii interact. In *AVI '10: Proceedings of the International Conference on Advanced Visual Interfaces*. ACM Request Permissions, May. doi:10.1145/1842993.1843100.
- Grossman, Tovi, and Ravin Balakrishnan. 2005. The bubble cursor: enhancing target acquisition by dynamic resizing of the cursor's activation area. In *CHI '05: Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM Request Permissions, April. doi:10.1145/1054972.1055012.
- Grudin, Jonathan. 2001. "Partitioning digital worlds: focal and peripheral awareness in multiple monitor use." *CHI '01: Proceedings of the SIGCHI conference on Human factors in computing systems*. Proceedings of the SIGCHI conference on Human factors in computing systems: 458–465. doi:10.1145/365024.365312.
- Hutchings, Dugald, and John Stasko. 2004. "Revisiting Display Space Management: Understanding Current Practice to Inform Next-generation Design." *GI '04: Proceedings of Graphics Interface 2004* (May 1).
- Isenberg, Petra, Danyel Fisher, Meredith Ringel Morris, Kori Inkpen, and Mary

- Czerwinski. 2010. An Exploratory Study of Co-located Collaborative Visual Analytics Around a Tabletop Display. In *2010 IEEE Symposium on Visual Analytics Science and Technology (VAST)*, 179–186. IEEE, October 24. doi:10.1109/VAST.2010.5652880.
- Ishak, Edward. 2007. “Content-Aware Interaction in User Interfaces.” *Thesis*.
- Jakobsen, Mikkel Rønne, and Kasper Hornbaek. 2010. Piles, tabs and overlaps in navigation among documents. In *NordiCHI '10: Proceedings of the 6th Nordic Conference on Human-Computer Interaction: Extending Boundaries*, 246–255. NordiCHI '10. New York, NY, USA: ACM. doi:http://doi.acm.org/10.1145/1868914.1868945.
- Jeong, Byungil, Luc Renambot, Ratko Jagodic, Rajvikram Singh, Julieta Aguilera, Andrew Johnson, and Jason Leigh. 2006. “High-performance dynamic graphics streaming for scalable adaptive graphics environment.” *SC '06: Proceedings of the 2006 ACM/IEEE conference on Supercomputing* (November 1). doi:10.1145/1188455.1188568.
- Jiang, Hao, Eyal Ofek, Neema Moraveji, and Yuanchun Shi. 2006. “Direct pointer: direct manipulation for large-display interaction using handheld cameras.” *CHI '06: Proceedings of the SIGCHI conference on Human Factors in computing systems* (April 1).
- Johanson, B, and A Fox. 2002. “The Event Heap: a coordination infrastructure for interactive workspaces.” *Mobile Computing Systems and Applications, 2002. Proceedings Fourth IEEE Workshop on* (May 31): 83–93. doi:10.1109/MCSA.2002.1017488.
- Kandogan, Eser, and Ben Shneiderman. 1997. “Elastic Windows: evaluation of multi-window operations.” *CHI 97: Proceedings of the SIGCHI conference on Human factors in computing systems*. Proceedings of the SIGCHI conference on Human factors in computing systems: 250–257. doi:10.1145/258549.258720.
- Go Big or Go Home, <http://www.forbes.com/forbes/2007/0604/070.html>.
- Khan, Azam, Justin Matejka, George Fitzmaurice, and Gordon Kurtenbach. 2005. Spotlight: directing users' attention on large displays. In *CHI '05: Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM Request Permissions, April. doi:10.1145/1054972.1055082.
- Leigh, Jason, and Maxine Brown. 2008. “Cyber-commons: merging real and virtual worlds.” *Communications of the ACM* 51 (1).
- Malik, S, A Ranjan, and R Balakrishnan. 2005. “Interacting with large displays from a distance with vision-tracked multi-finger gestural input.” *Proceedings of the ACM symposium on User interface software and technology*.

- McCallum, David, and Pourang Irani. 2009. "ARC-Pad: absolute+relative cursor positioning for large displays with a mobile touchscreen." *UIST '09: Proceedings of the 22nd annual ACM symposium on User interface software and technology* (October 1).
- Morris, Meredith, Jarrod Lombardo, and Daniel Wigdor. 2010. "WeSearch: supporting collaborative search and sensemaking on a tabletop display." *CSCW '10: Proceedings of the 2010 ACM conference on Computer supported cooperative work* (February 1). doi:10.1145/1718918.1718987.
- Olsen, B, S Dhakal, O Eldevik, and P Hasvold. 2008. "A large, high resolution tiled display for medical use: experiences from prototyping of a radiology ...." *Stud Health Technol Inform.*
- Park, K, and Y Cho. 2005. "Designing Display-Rich Environments for Collaborative Scientific Visualization." *LECTURE NOTES IN COMPUTER SCIENCE.*
- Richards, J, and P.E Mantey. 2006. Large Classroom Experience with an Interactive Tiled Display Mural. In *Frontiers in Education Conference, 36th Annual*, 15–20. Frontiers in Education Conference, 36th Annual IS . doi:10.1109/FIE.2006.322559.
- Robertson, G, M Czerwinski, K Larson, and D Robbins. 1998. "Data mountain: using spatial memory for document management." *Proceedings of the ACM symposium on User interface software and technology.*
- Robertson, G, E Horvitz, M Czerwinski, and P Baudisch. 2004. "Scalable Fabric: A Flexible Representation for Task Management." *Advanced Visual Interfaces.*
- Roman, M, C Hess, R Cerqueira, and A Ranganathan. 2002. "Gaia: A Middleware Infrastructure to Enable Active Spaces." *IEEE Pervasive Computing.*
- Shupp, Lauren, Robert Ball, Beth Yost, John Booker, and Chris North. 2006. "Evaluation of viewport size and curvature of large, high-resolution displays." *GI '06: Proceedings of Graphics Interface 2006* (June 1).
- Smarr, Larry, Andrew Chien, Tom DeFanti, Jason Leigh, and Philip Papadopoulos. 2003. "The OptIPuter." *Communications of the ACM* 46 (11) (November 1). doi:10.1145/948383.948410.
- Stolk, Bram, and Paul Wielinga. 2006. "Building a 100 Mpixel graphics device for the OptIPuter." *Future Generation Computer Systems* 22 (8) (October).
- Szwedo, Michael, Chris North, and Robert Ball. 2006. "Dynamic size and speed cursor for large, high-resolution displays." *Computer Science.*
- Tan, Desney, D Gergle, P Scupelli, and R Pausch. 2003. "With similar visual angles, larger displays improve spatial performance." *Proceedings of the SIGCHI*

- conference on Human factors in computing systems* (April 1).
- Tan, Desney, D Gergle, P Scupelli, and R Pausch. 2006. "Physically large displays improve performance on spatial tasks." *ACM Transactions on Computer-Human Interaction*.
- Tandler, Peter. 2001. "Software Infrastructure for Ubiquitous Computing Environments: Supporting Synchronous Collaboration with Heterogeneous Devices." *Proceedings of UbiComp 2001: Ubiquitous Computing, 2001* (November 28).
- Teasley, Stephanie, Lisa Covi, M. S. Krishnan, and Judith S. Olson. 2000. *How does radical collocation help a team succeed?* ACM, December 1. doi:10.1145/358916.359005.
- Teasley, Stephanie, L.A. Covi, M. S. Krishnan, and J.S. Olson. 2002. "Rapid software development through team collocation." *Software Engineering, IEEE Transactions on* 28 (7). Software Engineering, IEEE Transactions on: 671–683. doi:10.1109/TSE.2002.1019481.
- Tse, Edward, Mark Hancock, and Saul Greenberg. 2007. "Speech-filtered bubble ray: improving target acquisition on display walls." *ICMI '07: Proceedings of the 9th international conference on Multimodal interfaces* (November 1).
- Vogel, Daniel, and Ravin Balakrishnan. 2005. Distant freehand pointing and clicking on very large, high resolution displays. In *UIST '05: Proceedings of the 18th annual ACM symposium on User interface software and technology*. ACM Request Permissions, October. doi:10.1145/1095034.1095041.
- Vogt, F, J Wong, B.A Po, R Argue, S.S Fels, and K.S Booth. 2004. Exploring collaboration with group pointer interaction. In *Computer Graphics International, 2004. Proceedings*, 636–639. doi:10.1109/CGI.2004.1309280.
- Wallace, James R, Stacey D Scott, Taryn Stutz, Tricia Enns, and Kori Inkpen. 2009. "Investigating teamwork and taskwork in single- and multi-display groupware systems." *Personal and Ubiquitous Computing* 13 (8) (June 25): 569–581. doi:10.1007/s00779-009-0241-8.
- Yost, B, Y Haciahmetoglu, and C North. 2007. "Beyond visual acuity: the perceptual scalability of information visualizations for large displays." *Proceedings of the SIGCHI conference on Human factors in computing systems*.
- Zhou, Michelle, and Steven Feiner. 1996. "Data characterization for automatically visualizing heterogeneous information." *Information Visualization '96, Proceedings IEEE Symposium on*: 13–20. doi:10.1109/INFVIS.1996.559211.

# VITA

## EDUCATION

---

Ph.D., Electronic Visualization Laboratory (EVL), Computer Science Department,  
University of Illinois at Chicago [ 2003 – present ]

B.A. Computer Science and Chemistry, Lake Forest College [ 1999 – 2003 ]

## PROFESSIONAL EXPERIENCE

---

Research Assistant - EVL, University of Illinois at Chicago [ 2005 – present ]

Research Internship - Sharp Laboratories of America [ Summer 2008 ]

Programmer - Visual Solutions [ 2002 - 2003 ]

Lake Forest College [ 1999 - 2003 ]

## TEACHING EXPERIENCE

---

Lecturer: University of Illinois at Chicago [Summer 2004]

- Introduction to Computing and Programming

Teaching Assistant: University of Illinois at Chicago [2003-2005]

- User Interface Design
- Computer Graphics I
- Computer Architecture II

## PROFESSIONAL ACTIVITIES

---

### Demonstrations:

- *Supercomputing 2010, New Orleans, LA.*
- *Disney Studios, Burbank, CA, 2010.*
- *AAAS 2009, Chicago, IL.*
- *Supercomputing 2008, Austin, TX.*
- *Supercomputing 2007, Reno, NV.*
- *iGrid 2005 Workshop, UCSD, San Diego, CA.*

### Presentations / Talks:

- *IBM Almaden Research Center, San Jose, CA, Jun 2011.* – “Designing collaborative large display environments.”
- *NASA Ames Research Center, Moffet Field, CA, Jun 2011.* – “Designing collaborative large display environments.”
- *SAGE BoF Supercomputing 2010, New Orleans, LA, Nov 2010.* – “Designing

usable wall-display environments.”

- *ON\*VECTOR Workshop 2010, UCSD, San Diego, CA, Feb 2010.* – CSCW panelist, “CSCW in Ultra High-Resolution Display Environments.”
- *SAGE BoF Supercomputing 2009, Portland, OR, Nov 2009.* – “Multi-user, multi-modal user interaction on wall-sized displays.”

#### **Art Collaborations:**

- *Cernoch, A., Jagodic, R. “Replication of System”, 2008.*
- *Cernoch, A., Jagodic, R. “BendyBots”, 2008.*
- *Cernoch, A., Jagodic, R. “Infinity Gliders”, 2007.*

---

### **INVENTION DISCLOSURES**

**Jagodic R.**, Deshpande S., Yuan C. “Distributed, Scalable, Multi-User Widget Framework for Large, High Resolution Displays.” *Sharp Laboratories of America*, August 2008.

---

### **JOURNAL PUBLICATIONS**

1. Leigh, J., Johnson, A., Renambot, L., Peterka, T., Jeong, B., **Jagodic, R.**, Nam, S., Hur, H., Sun, Y. “Scalable Resolution Displays.”, *To appear in Proceedings of the IEEE, Special Issue: Beyond HTDV*, 2012.
2. **Jagodic, R.**, Renambot, L., Johnson, A.E., Leigh, J., Deshpande, S., “Enabling Multi-User Interaction in Large High-Resolution Distributed Environments.” *Future Generation Computer Systems*, Elsevier, 11/21/2010
3. Jeong, B., Leigh, J., Johnson, A., Renambot, L., Brown, M., **Jagodic, R.**, Nam, S., Hur, H. “Ultrascale Collaborative Visualization Using a Display-Rich Global Cyberinfrastructure.” *IEEE Computer Graphics and Applications*, doi:10.1109/MCG.2010.45, 05/01/2010 - 06/01/2010
4. Leigh, J., Renambot, L., Johnson, A., Jeong, B., **Jagodic, R.**, Schwarz, N., Svistula, D., Singh, R., Aguilera, J., Wang, X., Vishwanatha, V., Lopez Silva, B.A., Sandin, D., Peterka, T., Girado, J., Kooima, R., Gea, J., Long, L., Verlo, A., DeFanti, T., Brown, M., Cox, D. “The Global Lambda Visualization Facility: An International Ultra-High-Definition Wide-Area Visualization Collaboratory.” *Future Generation Computer Systems*, 10/01/2006 - 10/31/2006

1.

---

### **CONFERENCE PUBLICATIONS**

1. **Jagodic, R.**, Johnson, A., Leigh, J., “An Exploratory Study of Collaborative Work on Large High-Resolution Displays”, *Submitted to CSCW 2012*, 02/11/2012 – 02/15/2012
2. Chen, Y.C., **Jagodic, R.**, Johnson, A., Leigh, J., “Cross-Cultural Scientific Collaboration Case Studies,” *Workshop on The Changing Dynamics of Scientific Collaborations, 44th Hawaii International Conference on System Sciences 2011, Koloa, Hawaii*, 01/04/2011
3. Leigh, J., Renambot, L., Johnson, A., **Jagodic, R.**, Hur, H., Hofer, E., Lee, D. “Scalable Adaptive Graphics middleware for visualization streaming and collaboration in ultra resolution display environments.” *Proceedings of the Workshop on Ultrascale Visualization, 2008. UltraVis 2008, Austin, TX*, 11/16/2008 - 11/16/2008

4. Tsukishima, Y., Hirano, A., Taniguchi, A., Imajuku, W., Jinno, M., Hibino, Y., Takigawa, Y., Hagimoto, K., Wang, X., Renambot, L., Jeong, B., **Jagodic, R.**, Nam, S., Leigh, J., DeFanti, T., Verlo, A. "The First Optically-Virtual-Concatenated Lambdas over Multiple Domains in Chicago Metro Area Network Achieved Through Interworking of Network Resource Managers." *Proceedings of the 12th Optoelectronics & Communications Conference/16th International Conference on Integrated Optics & Optical Fiber Communication, Pacifico Yokohama, Japan, 07/01/2007 - 07/01/2007*
5. Leigh, J., Johnson, A., Renambot, L., DeFanti, T., Brown, M., Jeong, B., **Jagodic, R.**, Krumbholz, C., Svistula, D., Hur, H., Kooima, R., Peterka, T., Ge, J., Falk, C. "Emerging from the CAVE: Collaboration in Ultra High Resolution Environments." *Proceedings of the First International Symposium on Universal Communication, Kyoto, Japan, 06/14/2007 - 06/15/2007*
6. Jeong, B., Renambot, L., **Jagodic, R.**, Singh, R., Aguilera, J., Johnson, A., Leigh, J. "High-Performance Dynamic Graphics Streaming for Scalable Adaptive Graphics Environment." *Proceedings of SC06, Tampa, FL, 11/11/2006 - 11/17/2006*
7. Wang, X., Vishwanath, V., Jeong, B., **Jagodic, R.**, He, E., Renambot, L., Johnson, A., Leigh, J. "LambdaBridge: A Scalable Architecture for Future Generation Terabit Applications." *Broadnets 2006 - Third International Conference on Broadband Communications, Networks, and Systems, San Jose, CA, 10/01/2006 - 10/05/2006*
8. Renambot, L., Jeong, B., **Jagodic, R.**, Johnson, A., Leigh, J., Aguilera, J., "Collaborative Visualization using High-Resolution Tiled Displays," *CHI 06 Workshop on Information Visualization and Interaction Techniques for Collaboration Across Multiple Displays, Montreal, Canada, 04/22/2006 - 04/27/2006*
9. Jeong, B., **Jagodic, R.**, Renambot, L., Singh, R., Johnson, A., Leigh, J. "Scalable Graphics Architecture for High-Resolution Displays." *Proceedings of IEEE Information Visualization Workshop 2005, Minneapolis, MN, 10/23/2005 - 10/25/2005*

---

## PROFESSIONAL SKILLS

### Evaluations:

Qualitative and quantitative user studies, longitudinal observations, focus groups, surveys.

### Programming:

Python, C/C++, Java, GUI toolkits, multi-touch gestures, OpenGL, sockets, multithreading, RPC.

### Electronics:

Wiring and Arduino microcontrollers, electronic circuit design, analog and digital input sensors.

### Media editing:

Adobe (Premiere, Photoshop, Lightroom), Audacity.

---

## PROFESSIONAL MEMBERSHIPS

ACM Member [2009 – present]; ACM SIGCHI [2009 – present]