

M-Studio: an Authoring Application for Context-aware Multimedia

Pengkai Pan, Carly Kastner, David Crow, Glorianna Davenport

Room E15-351 Media Laboratory

Massachusetts Institute of Technology

20 Ames Street, MA, 02139, USA.

ppk@media.mit.edu, goodevil@mit.edu, crow@mit.edu, gid@media.mit.edu

ABSTRACT

Broadband wireless networks coupled with handheld computers and appropriate sensing technologies provide a channel for the delivery of mobile cinema. Mobile cinema changes the consumer experience of motion picture stories in that discrete cinematic sequences are delivered based on the consumer's location and the a story-real-time metric. The M-Studio authoring tool helps mobile story creators design, simulate and adjust mobile narratives. The tool provides the author with a graphical manipulation interface for linking content with a specific geographical space and a simulator allows the author to evaluate and iterate the content for continuity of story threads as they may be presented. The tool directly generates the code that is required for the server to deliver the cinematic sequences appropriately. This tool is discussed in the context of the two mobile narratives that have been created.

Keywords

Authoring tool, mobile multimedia, context-aware narrative, story making/sharing.

1. INTRODUCTION

Broadband wireless networks combined with handheld computers and various sensing technologies provide a new channel for the delivery of mobile cinema. This channel evolves in the social context established by mobile phone and text messaging technology. The M-Views research project explores how this channel can support the nascent mobile story format, which maps cinematic sequences to the movement of a consumer/viewer through space and time. Our exemplars to date include fiction and info-documentary story genres.

Mobile cinema is delivered on mobile PDA's using 802.11 network capability and spatial sensing such as GPS or infrared. Mobile stories are appropriate to many industries including entertainment, tourism and health.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

ACM Multimedia '02, December 1-6, 2002, Juan Les Pins, France.

Copyright 2002 ACM 1-58113-000-0/00/0000...\$5.00.



Figure 1: The M-Views Client

Like many interactive digital media forms that have preceded it, mobile cinema uses a database approach for storing cinematic sequences. The M-Views architecture supports a client-server delivery of cinematic segments and a peer-to-peer architecture for messaging between participants. Contextual sensing can be expanded from the current time and location sensing to include bio-sensors as well as historical data.

Mobile stories are generated by defining goals, style, characters, and action; these each need to be defined in light of technical constraints, including positioning of viewers. In order to visualize how stories will be delivered over this architecture, authors require a tool for simulating story play-out given the movements through time/space of any individual viewer. For this purpose, we have developed M-Studio, a tool-kit that enables authors to design, script, and simulate mobile narratives.

M-Studio supports authors throughout the complex story construction process beginning with initial sketching and storyboarding through to final simulation and testing. The tool provides authors with a visual direct manipulation interface allowing the author to focus on the narrative grid rather than on the machine coding requirements. While authoring tool does not lessen the challenge of creating interesting, complex, self-aware story lines, it does allow authors to visualize story structures, examine story threads and partially simulate the audience's experience in the mobile environment.

The current version of M-Views [Figure 2] consists of three sub-systems: M-Views client [Figure 1], M-Views server, and M-Studio. The client is context-aware, multimedia delivery software running on a Compaq iPAQ outfitted with a GPS receiver and

802.11b network card. The M-Views Server is built from Java Servlets that maintain an XML database of story and message events. M-Studio is a PC-based multimedia authoring tool for designing, editing and examining context-aware stories that are viewed in mobile environments.

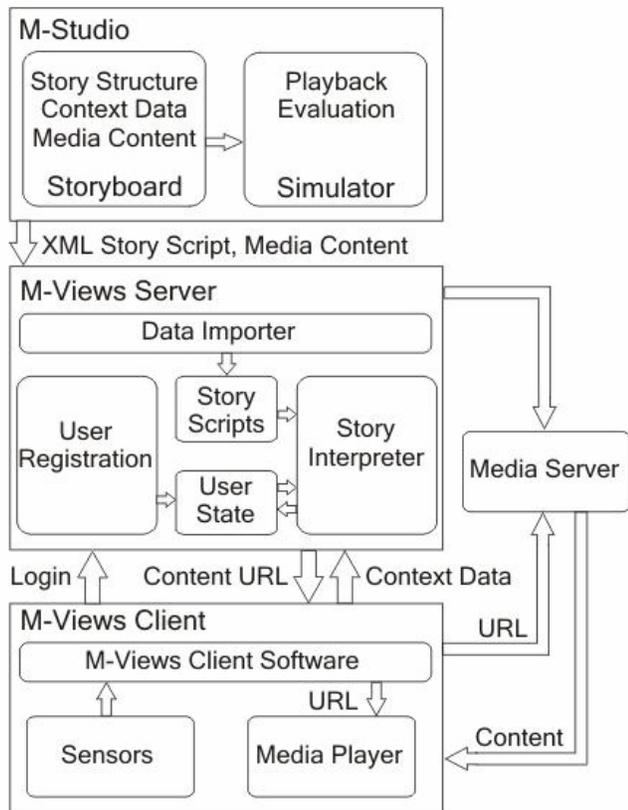


Figure 2: A Brief Overview of M-Views Architecture

There are several multimedia authoring tools, such as Authorware [5], Director [6], and HyperCard [1], and design tools, such as DEMAIS [2] and DENIM [4], that have been developed before. None of these tools account for the dynamic interactions of mobile environments. To design context-aware mobile multimedia, a creator needs to deal with more dynamic attributes of the viewing experiences than s/he does for desktop-based multimedia design. QuakeSim [7] is a Quake III Arena-based 3D simulator for GeoNotes [3] system, which allows users to annotate their present location with virtual Post-Its. The current version of QuakeSim supports neither multimedia authoring nor real context information simulation. M-Studio has much more comprehensive components, such as a Storyboard, Location Editor, Clip Editor, and simulators, to facilitate designing context-aware multimedia.

This remainder of the paper is organized as follows. In the next section, we focus on the design goals and implementation of M-Studio. We will discuss the system features with reference to “Another Alice” [8], a murder mystery developed for M-Views, which allowed the viewer to follow different characters and gather clues to catch a killer. In section 3, we present a preliminary evaluation of M-Studio. Finally, section 4 presents our conclusions with a view of ongoing and future work.

2. SYSTEM IMPLEMENTATION

The process of creating context-aware mobile stories is complex, iterative, and filled with challenges. M-Studio allows authors to visualize and build multithreaded, context-aware stories and provides the server with instructions regarding the presentation of these stories. The design of a non-linear story can be a difficult task, especially given the burden of functionally visualizing the impact of context information related to every scene. M-Studio simplifies these tasks, allowing authors to design the initial story with these factors in mind, and providing the tools to make the final story available to users. Our key design goals are:

- **Interaction:** There are two types of interactions between the audience and M-Views Client: external interactions, such as the audience clicking a button or walking to a particular location, and internal interactions, based on story structure, such as the characters met in the clip. M-Studio should help the creator be aware of these possible interactions and provide means to organize them.
- **Visualization:** Context-aware stories can be very complex because they contain multiple story locations, threads, and interactions. For example, “Another Alice” has four characters and three main parallel structures, but there are at least 12 possible story threads that the user may experience. Without an intuitive visualization tool, the design team will not be able to understand and discuss the whole picture of the story.
- **Simulation:** One of main difficulties for context-aware story creators examining mobile experiences from the desktop. An author cannot physically walk through the story every time it changes. Well-designed simulators help the creator to understand how a story might be viewed on mobile devices.
- **Integration:** Designing a context-aware story requires certain knowledge about how the client, server and authoring tool work. In order to make the authoring process as intuitive as possible, it is necessary to separate technical details from design. In M-Studio, authoring a story does not require any scripting language. However, advanced story creators can still configure and customize story setup in the M-Views server with XML.

M-Studio consists of two main components, the Storyboard and the Simulator [Figure 4]. Authors can use the Storyboard tool to layout clips and associate them with locations. Then, they can use the Simulator tools to model possible stories that can be created from these clips. Once satisfied with the results, the creator can use the Storyboard to export the story to the server. M-Views users can then contact the server with their clients to begin playing the story.

2.0 Storyboard

The Storyboard [Figure 3] allows authors to implement multi-threaded context-aware stories. This tool is similar to the video-clip sequencing tool common to random access editing products, but allows users to layout several parallel storylines thereby visualizing the temporal/spatial/character interactions relative to each other. Additionally, each video clip is also associated with metadata, which could be both external (like location or time) and internal (like characters meet in the clip) to the story.

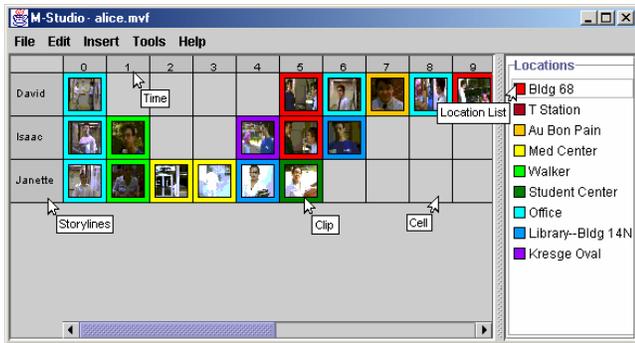


Figure 3: M-Studio Storyboard

The Storyboard is the central component to the authoring system, providing data to both the simulators and the server. It is the space where the author builds his or her story. It keeps track of the different locations where clips can take place, the possible storylines, and the positions of all the clips. The storyboard allows the author to visualize how multiple parallel storylines unfold in time.

Alternatively, the author can view the story layout from the perspective of location and time. This allows the story creator to observe the balance of story events across space and time. This is particularly useful if the author hopes to create a story that fully exploits its settings.

Parallel storylines are a useful abstraction for a multi-threaded or multiple-point-of-view story. They can represent completely independent, self-sufficient stories, or pieces of a story that can come together and diverge. In “Another Alice”, the viewer chooses to follow a particular character, so in M-Studio, we represent this by having each storyline correspond to the path one particular character took. The user can cross between storylines when different characters meet.

The Location Editor provides a convenient way for users to define and visualize the locations where their clips take place, using maps to provide a visual frame of reference for authors, allowing them to understand the settings for their stories. Once the author has selected a few starting locations, s/he can begin to enter clips. Clips encapsulate both context data and the media elements that will be presented to the end users.

To allow for greater control over story flow, clips are also associated with flags. Each clip has a set of global conditions that must be met for it to play and a set of changes it makes to the state of the story. For instance, a clip can require that the user has or has not seen a set of other clips. Flags allow an author to create internal variables that represent structural elements of the story. To ease the complexity of setting flags, the storyboard can automatically generate some of these flags based on a clip’s position in the storyboard, suggesting connections between clips to the author.

Clips can contain many types of media; in addition to video, they can also use text, images, or sound. This can be useful in the process of writing the initial story, allowing the author to plan a scene using text and images. This clip could later be replaced with a standard video file when shooting is complete.

When the author is satisfied with the work, s/he can upload the story to the server. The server uses an XML template for story scripts that treats clips as events with a context that require certain

conditions to occur and change the user state. M-Studio uses the clips’ location data and flags to generate this document. It then packages the XML with the video files to be uploaded and exports the data to the server, which sets up a story available for public viewing.

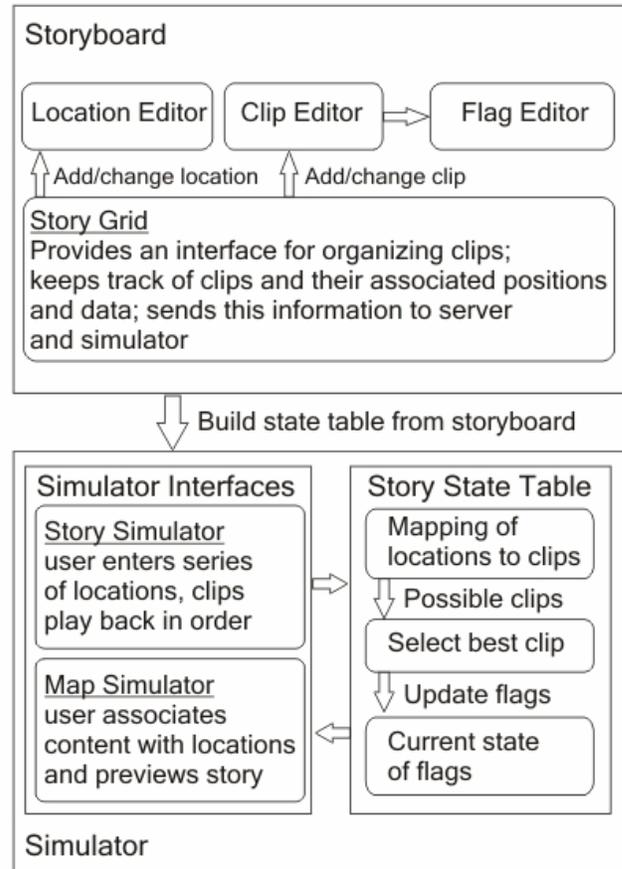


Figure 4: M-Studio Architecture Overview

2.1 Story Simulator

The Story Simulator is the second component to story authoring. Multi-threaded stories can become quite complicated as the possibility for crossover between threads increases. As production is an expensive component of creating these stories, the author will want to simulate the appropriateness and logic of the crossovers at several stages before finalizing the story and releasing it to the public. The simulator allows the author to evaluate story scripts as well as final performance in produced sequences. The Story Simulator allows an author to view a story given any timing and physical path that a viewer might follow.

To produce these simulations, the simulator maintains a mapping of locations to clips and a table of state flags. When a particular location is accessed, the simulator checks through all the possible clips for that location and returns one of it requirements are met. Then, it updates the state table, and goes on to the next location.

Using this interface, the simulator can also generate all possible paths through a story. The simulator examines all the clips and determines which ones could be played, given the current state of the flags, and creates new paths with their own state tables for each new possibility. The author can then examine each of these

storylines to confirm the current flag structure is producing the desired results.

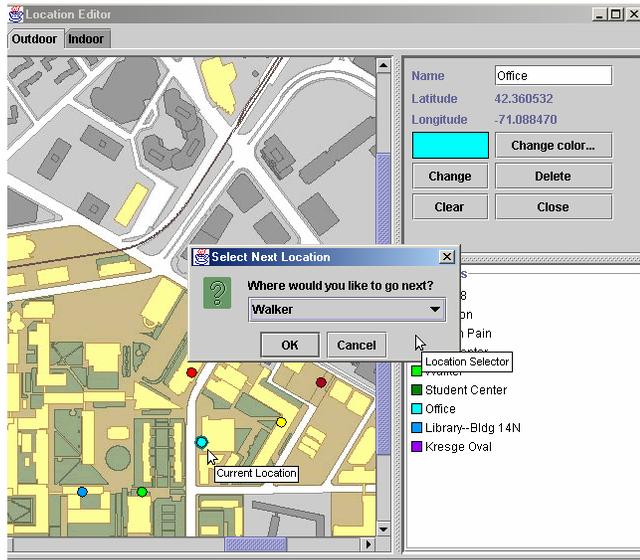


Figure 5: M-Studio Map Simulator

The Map Simulator [Figure 5] helps the creator understand the experience of actually playing the story. Once the author selects the starting storyline, this tool scrolls the map to the starting location, and plays the opening clip. After viewing the clip, the author is prompted to select the next location they would like to visit. If there is a clip at the selected location, it plays, just as it would if an end user visited that physical location.

3. PRELIMINARY EVALUATION

To date, evaluation of M-Studio has been limited to the development of a few location-aware stories that have been made at MIT and Harvard Square. Four story creators (only one creator used M-Studio to finish a whole context-aware mobile story, “Another Alice”) and three other MIT students have tested M-Studio from a story authoring and system design perspective.

In order to better understand the demands of mobile, location-aware narrative and to test the functionality of M-Studio, we designed and produced a location-aware video fiction story, “Another Alice”. In this “who-dun-it,” we cast the viewer in the role of investigator. The viewer experiences the story by following different characters to different locations. As the viewer arrives at each new location, s/he is able to view the relevant scene on M-Views client.

From concept through production, the creation of “Another Alice” provided a concrete mobile multimedia fiction on which to model the structure and features of M-Studio. One important lesson we learned from the creation of “Another Alice” was that the form and content of a story are shaped by dynamic attributes of the viewing experience—the audiences are navigating through space and time. The story must be structured in a mobile context, so that both location and time act as triggers to retrieve the right pieces of multimedia information from the server. Furthermore, unlike the PC-based multimedia authoring process, in which the authoring and consuming platforms are the same, context-aware multimedia exists on two different platforms: the desktop for authoring and the mobile environment for consuming. It is difficult for the

creator to imagine and examine multimedia content that will actually be viewed in mobile environments on the desktop.

Overall, our trial users were positive about their M-Studio experience, saying that it was easy to use and allowed for clear visualization of complex stories. In particular, users responded positively to M-Studio's usability, finding the interface intuitive and usable with only minimal instruction. People were particularly fond of the easy to understand transition from locations on the maps of the location editors to locations on the storyboard. They also thought the implementation of the story simulator was helpful, allowing users to browse through a sequence of clips.

4. CONCLUSIONS

In this paper, we discussed M-Studio in terms of motivation, design and implementation. According to our preliminary evaluation, M-Studio is useful for designing context-aware stories. In particular, it supports authors in the production phase of story development as well as in some of the initial creation, and it provides the instruction set for the runtime server.

While multi-threaded stories for ‘interactive’ entertainment have been envisioned over the past quarter-century, and many prototypes have been created, we are still in the early days of developing compelling forms. Wireless handheld computers that display video offer us a novel and exciting opportunity to create a mobile movie form that will provide a fun, compelling, and entertaining experience.

5. REFERENCES

- [1] Apple Computer. HyperCard: Reference Manual, 1993
- [2] Bailey, B.P., J.A. Konstan, and J.V. Carlis. DEMAIS: Designing Multimedia Applications with Interactive Storyboards. *Proceedings ACM Multimedia*, 2001.
- [3] Espinoza, F., Persson, P., Sandin, A., Nyström, H., Cacciatore, E. & Bylund, M. (2001) GeoNotes: Social and Navigational Aspects of Location-Based Information Systems, in Abowd, Brumitt & Shafer (eds.) *UbiComp 2001: Ubiquitous Computing*, International Conference Atlanta, Georgia, September 30 – October 2, Berlin: Springer, p. 2-17
- [4] Lin, J., M.W. Newman, J.I. Hong, and J.A. Landay. DENIM: Finding a Tighter Fit Between Tools and Practice for Web Site Design. *Conference Proceedings on Human Factors in Computing Systems*, 2000.
- [5] <http://www.macromedia.com/software/authorware/>
- [6] <http://www.macromedia.com/software/director/>
- [7] Markus Bylund and Fredrik Espinoza. Using Quake III Arena to Simulate Sensors and Actuators when Evaluating and Testing Mobile Services. *In Proceedings of the CHI 2001 Conference on Human Factors in Computing Systems*, March 31-April 5, 2001
- [8] Pengkai Pan, Christina Chen, and Glorianna Davenport. The Birth of "Another Alice" *Computers and Fun 4*. University of York, November 2001