

Designing for Community Appropriation of Media

Ryan Shaw

School of Information Management and Systems, UC Berkeley
ryanshaw@sims.berkeley.edu

INTRODUCTION

My colleagues and I are currently in the planning stages of a project to create web-native infrastructure and tools for interacting with video information. Users will be able to annotate, re-edit and re-use video content in a networked, collaborative fashion. Key user groups include fan communities, video bloggers, and independent journalists. As we design the system the architecture, we are trying to design a system that not only enables appropriation of media, but is itself able to be appropriated for purposes we cannot foresee.

TOWARD SEMIOTIC DEMOCRACY

Currently not all members of society are able to participate equally in the production of media and culture. Professional producers dominate semiotic discourse, while the majority is restricted to a role of passive consumption. Although the emergence of the World Wide Web and the mainstreaming of web publishing software have started to change this situation, there is still a considerable gap between media producers and media consumers, especially in mediums like film and television. We are interested in building systems to enable consumers to shape media (and media technologies) to create their own meanings, what William Fisher has called “semiotic democracy” [4].

COMPUTATIONAL MEDIA

Our research into future systems for media production is founded upon the idea that media production can be viewed as a computational process, and that media assets can become parameterizable, dynamic pieces of software rather than static artifacts [2]. Past research has focused on tools for representing the semantic and syntactic structure of video content in order to make that content recombinable and adaptable [3]. As we investigate ways to bring these tools to the Web and make them scale to much larger communities of users, we are seeking to make not only media content, but the tools themselves recombinable and adaptable. This has led us to investigate current approaches to designing for appropriation, and to look at how these approaches apply to both the design of computational media objects and the design of systems for computational media production.

DESIGNING FOR APPROPRIATION

The clearest set of principles for designing software to be appropriated were expressed by Tom Moran in his 2002 keynote to the Designing Interactive Systems conference [5]. I will briefly discuss a few of those principles here, and how they apply to our project.

Modularity

By composing standard descriptors for annotating, retrieving, and repurposing video content, we can transform content in media archives into reusable parts for piecing together new productions. Likewise, a modular software architecture allows recombining and repurposing of media technologies to meet the needs of a wide variety of possible users and contexts of use [6]. The aim in both cases is to allow anyone, with a minimum of training, to quickly compose modules to meet their needs.

Openness

We want to make media content, the data created by users as they annotate and appropriate this content, and the services that allow this annotation and appropriation as open and accessible as possible. This openness is necessary for ensuring spontaneous collaboration [8] and ubiquity [9]. In particular, due to the current state of copyright law, it must be possible for users to appropriate content to which they have only read-access, and to do so without having to copy that content.

Transparency

Providing users with easy access to underlying system implementation details has proven to be a powerful way of spreading expertise in the workings of the World Wide Web. Unfortunately, multimedia has been the exception to this rule, with proprietary formats and closed protocols being the norm. We propose to make both media objects and the APIs for interacting with these objects as transparent and discoverable as possible, by following best practices like those espoused in [1] and [7].

Layered architecture

A layered, or orthogonal, architecture makes it possible to change one part of a system without requiring changes to the other parts. This allows some parts of the system to change quickly, adapting to rapid change, while other parts constrain change and provide continuity. Our framework for semantic annotation of media is divided into orthogonal facets, which allow the framework to be extended without losing compatibility with previous versions. The same principles can be applied (as they have in the design of the Web itself [1]) to ensure that the platform we are building is similarly extensible.

Process

Finally, processes must be developed to support appropriation, to educate users about how the system works and support them when they have questions or suggestions

for new features. In our system, since easy appropriation relies on quality annotation, there must a community process for determining whether annotations are of sufficiently high quality. Likewise, there should be a community process for deciding whether the benefits of proposed API extensions are worth the potential added complexity.

CONCLUSION

We believe that these principles for designing “hackable” systems are applicable not only to traditional software systems, but also to the self-describing, dynamic software objects that media assets will become. We hope to apply these principles to design a system that will be appropriated not only by traditional “hackers” (computer programmers) but also by “media hackers,” including fan communities, video bloggers, and independent journalists. Hopefully, as these communities collaborate and the distinctions among them blur, we will begin to see what possibilities a society of truly democratic media production may offer.

REFERENCES

1. Architecture of the World Wide Web, Volume One. Jacobs, I. and Walsh, N., eds. W3C Recommendation, 15 December 2004, [http://www.w3.org/TR/2004/REC-](http://www.w3.org/TR/2004/REC-webarch-20041215/)

[webarch-20041215/](http://www.w3.org/TR/webarch/). Latest version available at <http://www.w3.org/TR/webarch/>.

2. Davis, M. Garage Cinema and the Future of Media Technology. *Communications of the ACM (50th Anniversary Edition Invited Article)*, 40 (2 1997), 42-48.
3. Davis, M. Media Streams: An Iconic Visual Language for Video Annotation. *Teletronikk 4.93*, (1993), 59-71.
4. Fisher, W. Theories of Intellectual Property. <http://www.law.harvard.edu/faculty/dfisher/iptheory.html>.
5. Moran, T. Everyday Adaptive Design. <http://www.sigchi.org/DIS2002/resources/DIS2002TomMoranKeynote.ppt>
6. Udell, J. The Beauty of REST. <http://www.xml.com/pub/a/2004/03/17/udell.html>.
7. Udell, J. Tangled in the Threads: The power of the URL-line. <http://udell.roninhouse.com/bytecols/2001-08-15.html>.
8. Wilensky, R. Digital Library Resources as a Basis for Collaborative Work. *JASIS*, 51 (3 2000), 228-245.
9. Zawodny, J. Ubiquity in the Internet Age. <http://jeremy.zawodny.com/blog/archives/002931.html>.