Code, Camera, Action!

How Software Developers Document and Share Program Knowledge Using YouTube

Laura MacLeod, Margaret-Anne Storey & Andreas Bergen
University of Victoria, BC, Canada

CHISEL
Computer Human Interaction & Software Engineering Lab

University of Victoria

Screencasting

> *I continue to be fascinated by this medium. The ability to **capture, narrate, and share software experiences**... enables an important mode of communication that we've barely begun to exploit.*

- Jon Udell

Software Development & Social Media

Previous work has explored:

Twitter	GitHub	Stack Overflow
Singer et al. 2014	Dabbish et al. 2012	Parnin et al. 2012

Software Development & Social Media

Awareness
Singer et al. 2014

Collaboration
Dabbish et al. 2012

Identity Building
Storey et al. 2014

Knowledge Foraging
Brandt et al. 2010

Knowledge Sharing
Parnin et al. 2012

But...
How do developers use YouTube to **share** software development **knowledge**?

Research Questions

RQ1: What Kinds of **Program Knowledge** Are Captured in Screencasts?

RQ2: What **Techniques** Do Developers Use to Document Code in Screencasts?

RQ3: **Why** Do Developers Create Code Screencasts?

RQ4: **How** Do Developers Produce Code Screencasts?

Methodology

Methodology

Phase 1

Answered through the analysis of screencasts

RQ1:
What Kinds of Program Knowledge Are Captured in Screencasts?

RQ2:
What Techniques Do Developers Use to Document Code in Screencasts?

Phase 2

Answered through interviews with screencast creators

RQ3:
Why Do Developers Create Code Screencasts?

RQ4:
How Do Developers Produce Code Screencasts?

Methodology

Data collected

Phase 1

Answered through the analysis of screencasts

Screencast sample

20 videos, 8+ hours of footage

Phase 2

Answered through interviews with screencast creators

Interviews

10 interviews with screencast creators

Findings

Findings

RQ1: What Kinds of **Program Knowledge** Are Captured in Screencasts?

1. Sharing Customization Knowledge

> *Hey. I'm just going to give a quick tour of some of the source code for Asteria and **how to mod it** and so forth and **customize it…***
>
> - *Video 7*

Findings

RQ1: What Kinds of **Program Knowledge** Are Captured in Screencasts?

2. Sharing Development Experiences

> *I didn't use multithreading. **At first I didn't have time for it,** and then I didn't feel like it was quite necessary. I mean, if you want to add that functionality, you can.*
>
> - *Video 19*

Findings

RQ1: What Kinds of **Program Knowledge** Are Captured in Screencasts?

3. Sharing Implementation Approaches

```
78                  logininfo.Close();
79              }
80              else
81              {
82                  StreamWriter logininfo = new StreamWriter(@"C:\GMODit\.minecraft\gmoditlastlogin"); // creates
83                  logininfo.Close();
84              }
85
86              bool ismodded = false;
87              bool clientExists = false;
88              string folderlocation = @"C:\GMODIT\.minecraft";
89              string modlocation = folderlocation + @"\mods";
90              string coremodlocation = folderlocation + @"\coremods";
91              string templocation = "c:\\temp\\";
92
93              if (Directory.Exists(folderlocation))
94              {
95                  clientExists = true;
96              }
97
98              if (!Directory.Exists(templocation))
99              {
100                 Directory.CreateDirectory(templocati
101             }
100 %
```

Minecraft Launcher - Code walkthrough

XstreamRockerr

Subscribe 34

14,406

Add to Share More 82 5

Published on 26 Apr 2013

This is a video that I threw together of how to make a custom minecraft launcher. I coded everything in C# and I can post a link to my code if there is a popular demand. Most of it is already in the video, but it can be a pain to type up.

Link to Code: https://dl.dropboxusercontent.com/u/6...

Findings

RQ1: What Kinds of **Program Knowledge** Are Captured in Screencasts?

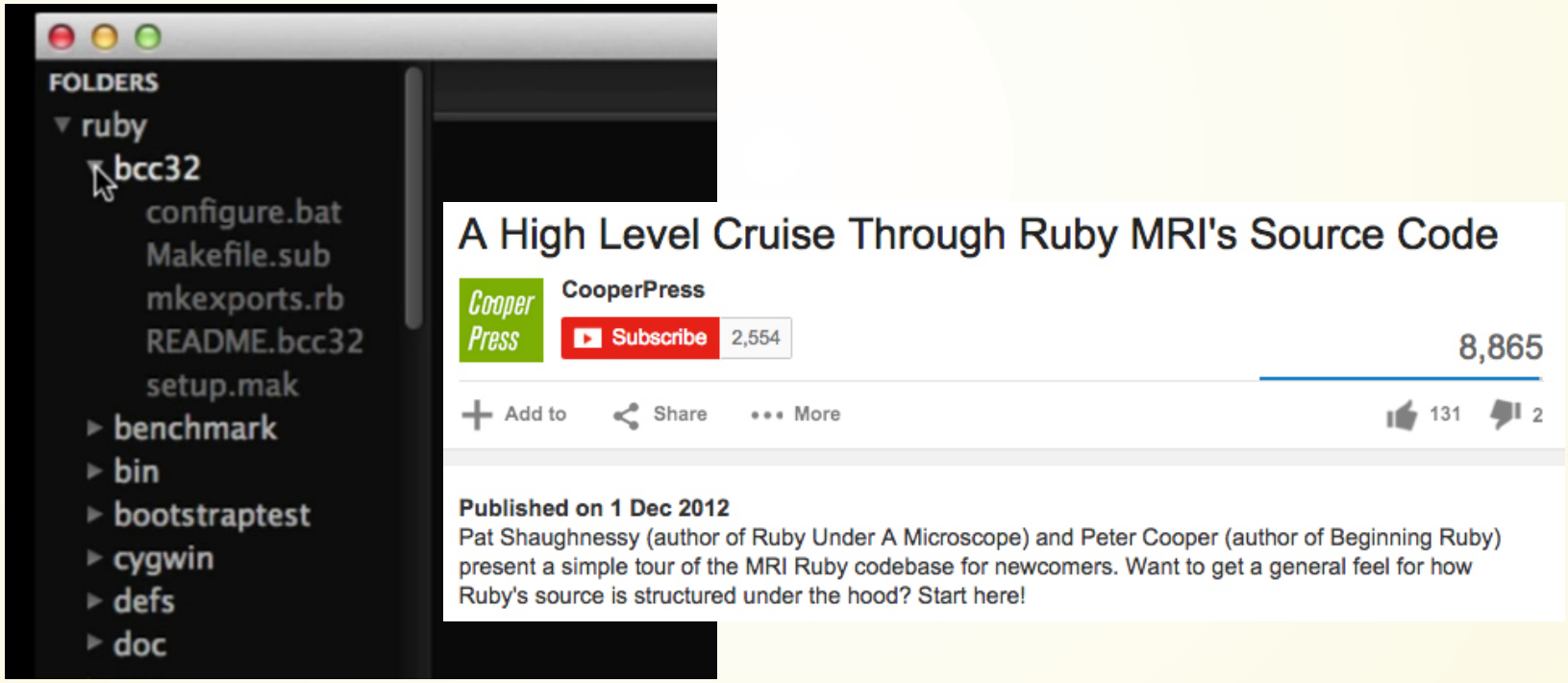
4. Demonstrating the Application of Design Patterns

> *Go through an example application so you can see the Lua code, you can see the Robotlegs implementations, and some of the various ways that **we can implement the patterns** that Robot-legs does.*
>
> - *Video 2*

Findings

RQ1: What Kinds of **Program Knowledge** Are Captured in Screencasts?

5. Explaining Data Structures



Findings

RQ2: What **Techniques** Do Developers Use to Document Code in Screencasts?

- Goal Setting
- Live Editing to Showcase Code Changes
- Demonstrations to Showcase the Execution of the Program
- Referencing Different Levels of Detail
- Browsing the Technical Environment
- Provisioning of Additional Resources
- Mapping Execution to Code and Code to Code

Findings

RQ2: What **Techniques** Do Developers Use to Document Code in Screencasts?

Goal Setting

“

Hey. I’m just going to give a quick tour of some of the source code for Asteria and how to mod it and so forth and customize it…

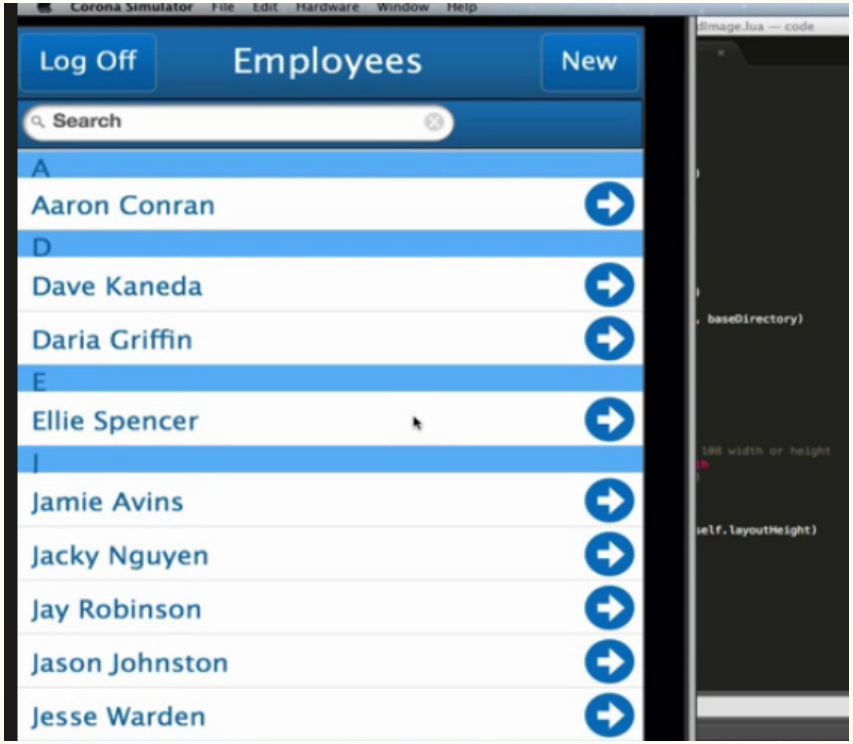
”

- *Video 7*

Findings

RQ2: What **Techniques** Do Developers Use to Document Code in Screencasts?

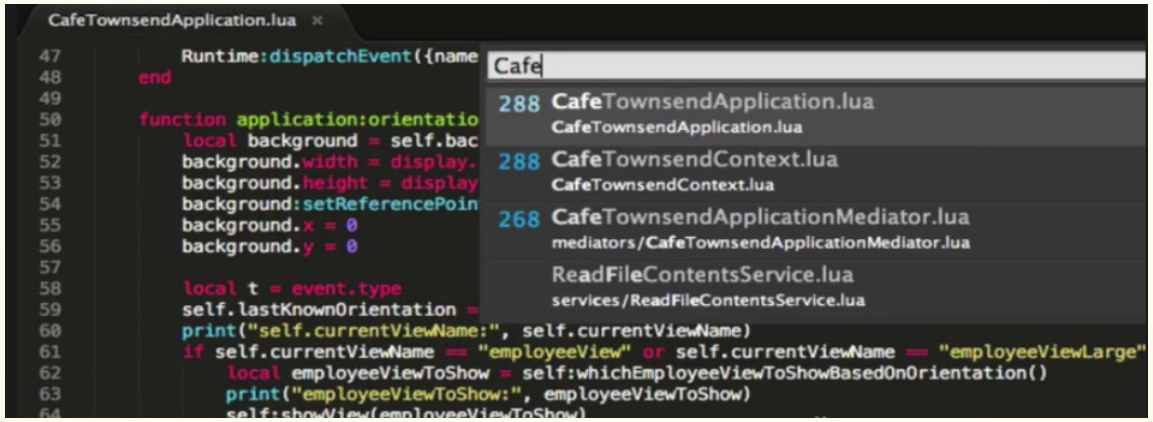
Demonstrations to Showcase the Execution of the Program



Findings

RQ2: What **Techniques** Do Developers Use to Document Code in Screencasts?

Mapping Execution to Code and Code to Code



Findings

RQ3: **Why** Do Developers Create Code Screencasts?

To Build an Online Identity

“ *So I've done that a couple times, where I've recorded the video, I felt really good, [but] I never released it. It was four hours and I was ready to go and I thought this sucks, or I don't like it because I know all the trolls on Reddit will have issue with it... So* ***I have to be very careful about what I release out there****.* ”

- *Interviewee 7*

Findings

RQ3: **Why** Do Developers Create Code Screencasts?

To Promote Themselves

“ *It went wild.* ***Like I got tens of thousands of people viewing these videos*** *and sending me personal messages about how amazing the course is and they want more... I have a lot of blog followers now and I have a couple thousand YouTube subscribers [up] from thirty.* ”

- *Interviewee 10*

Findings

RQ3: **Why** Do Developers Create Code Screencasts?

As a Learning Exercise

“ *I was doing it on a weekly basis. So once a week, ok lets try something new. So I just open up the Blender specs and* ***see what developers were working on*** *what was new, what people might have some issues with.* ”

- *Interviewee 1*

Findings

RQ3: **Why** Do Developers Create Code Screencasts?

To Give Back

“ *They said it was really helpful ... and I actually saw it reflect in the way they were writing their code, so* ***these are the things I wish I had*** *when I went through the same thing, and that’s my guide...* ”

- *Interviewee 3*

Findings

RQ3: **Why** Do Developers Create Code Screencasts?

As an Alternative to Blogging

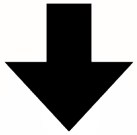
“ *If I had the choice to watch a video **I'll definitely watch the video**. It's way faster and a better, smoother, learning curve.* ”

- *Interviewee 3*

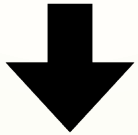
Findings

RQ4: **How** Do Developers Produce Code Screencasts?

Preparing the Screencast



Recording the Screencast



Post-production

Discussion

Screencasting Best Practices

1. Plan Ahead
2. Short and Focused
3. Provide the Source Code
4. Speak Clearly
5. Execute the Code

Threats to Validity and Limitations

Selection Bias

Researcher Bias

Generalizability of Findings

Future Work

Using Screencasts

Screencast Communities

Tool Support

Conclusion

- Explores screencasting for software developers
- Shows how developers share technical knowledge through YouTube hosted screencasts
- Puts forth techniques and motivations behind screencasts for developers
- Explores the process developers use to create screencasts

Questions?

Laura MacLeod
@LauraAnnMacLeod

Dr. Margaret-Anne Storey
@margaretstorey

References

1. Jon Udell. Name that Genre: Screencast, November 2004.

2. Leif Singer, Fernando Marques Figueira Filho, and Margaret-Anne Storey. Software Engineering at the Speed of Light: How Developers Stay Current Using Twitter. ICSE, pages 211–221, 2014.

3. Laura Dabbish, Colleen Stuart, Jason Tsay, and Jim Herbsleb. Social Coding in Github: Transparency and Collaboration in an Open Software Repository. In Proc. of the ACM 2012 conference on CSCW, pages 1277–1286. ACM, 2012.

4. Christoph Treude, Ohad Barzilay, and Margaret-Anne Storey. How do Programmers Ask and Answer Questions on the Web?. In 33rd Intl Conference on Software Engineering (ICSE), pages 804–807. IEEE, 2011.

5. Chris Parnin, Christoph Treude, Lars Grammel, and Margaret-Anne Storey. Crowd Documentation: Exploring the Coverage and the Dynamics of API Discussions on Stack Overflow. Georgia Institute of Technology, Technical Report, 2012.

6. Storey, Margaret-Anne, et al. “The (R) Evolution of Social Media in Software Wngineering.” Proceedings on the Future of Software Engineering, pages 110-116. ACM, 2014.

7. Joel Brandt, Mira Dontcheva, Marcos Weskamp, and Scott R Klemmer. Example-centric Programming: Integrating Web Search into the Development Environment. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pages 513–522. ACM, 2010.

Appendix A

Inter-rater reliability equation

$$Percentage of Agreement = \frac{Number of Agreements}{Number of Agreements + Number of Disagreements} \times 100$$

Themes	Code	Video
Goal Setting	Explaining the limitations (i.e., the scope) of the video and its intended audience	V1, V2, V3, V4, V5, V6, V7, V8, V10, V11, V13, V14, V15, V16, V17, V18, V20
	Verbally defining the video's purpose	V1, V2, V3, V4, V5, V6, V7, V8, V10, V11, V13, V15, V16, V17, V18, V19, V20
Live Editing to Showcase Code Changes	Live code changes	V2, V3, V5, V6, V7, V8, V10, V11, V13, V14, V19, V20
	Changing control flow or variables	V2, V13, V19, V20
	Introducing bugs	V2, V3, V5, V13, V15, V20
Demonstrations to showcase the execution of the program	Executing the program to demonstrate features to the audience	V1, V2, V3, V5, V6, V9, V13, V16, V17, V18, V19, V20
Referencing Different Levels of Detail	High-level code overview	V1, V2, V3, V4, V5, V6, V7, V8, V9, V10, V11, V12, V13, V14, V15, V16, V17, V18, V19, V20
	Medium-level focus on sub-block of code	V1, V2, V3, V4, V5, V6, V7, V8, V9, V11, V12, V13, V14, V15, V16, V17, V18, V19, V20
	Low-level focus on a single line of code	V1, V2, V3, V4, V5, V7, V8, V9, V10, V11, V12, V13, V14, V15, V16, V17, V18, V19, V20
	Picks out element identifier	V1, V2, V3, V4, V5, V6, V7, V8, V9, V10, V11, V12, V13, V14, V15, V16, V17, V18, V19, V20
	Return types/ parameters	V1, V2, V3, V4, V5, V7, V8, V10, V12, V13, V14, V15, V17, V18, V19, V20
	Line numbers	V1, V4, V5
Browsing the Technical Environment	Make use of file explorers	V1,V2, V4, V5, V6, V7, V9, V11, V13, V17
	Explaining the program structure	V1, V2, V4, V6, V7, V10, V13, V14, V16, V17, V18, V20
	Explaining the technical environment (including libraries, servers)	V1, V2, V3, V4, V5, V6, V8, V10, V11, V12, V13, V14, V16, V17, V20
Provisioning of Additional Resources	Webpages	V1, V2, V4, V6, V10, V11, V13, V16, V19, V20
	Diagrams	V16, V17, V18
	Source Code	V1,V2, V4, V6, V9, V10, V13, V18, V19, V20
	Visual Annotations	V1, V3, V6, V10, V11
Mapping Execution to Code and Code	Connection between the demonstration and the code	V2, V5, V6, V9, V13, V15, V16, V17, V18, V19, V20
	Linking code segments together	V1, V2, V3, V5, V7, V9, V10, V11, V12, V13, V15, V16, V17, V18, V19, V20

Appendix C

Excel example of open coding videos

Video Title:	V5				
HH:MM:SS	Code 1	Code 2	Code 3	Notes	Memos
00:00:00	C1 Prepping the user/ setting up expectations			Motivating for why he did what he did	Talking about previous coding experiences? What led them to these questions, their motivation for showing this code. Previous experiences impact the code they built and are therefore showing??
00:00:05					
00:00:10					
00:00:15					
00:00:20					
00:00:25					
00:00:30		B1 Slides for background story			

Appendix D

Interview Questions

1. Tell me a bit about yourself.
2. What do you do for a living?
3. (If not obvious from the above:) Would you consider yourself a developer?
4. Where are you located?
5. As a contributor how do you use YouTube?
6. How many videos do you have on YouTube?
7. What prompted you to start posting on YouTube?
8. How do you decide to make a video?
9. Think about the last video you made, can you describe the process?
10. Were these any different than other videos you have made?
11. If you have many videos, has your process changed over time?
12. Who do you think your audience is?
13. How do you picture them?
14. What do you think your audience is trying to get out of your videos?
15. What tools/ software do you use to make your videos?
16. Do you edit your videos?
17. If so, how long does it take?
18. Have you received any feedback from your videos?
19. If so what has it been like? What have people said?
20. What do you think makes a good code walkthrough video?

Appendix E

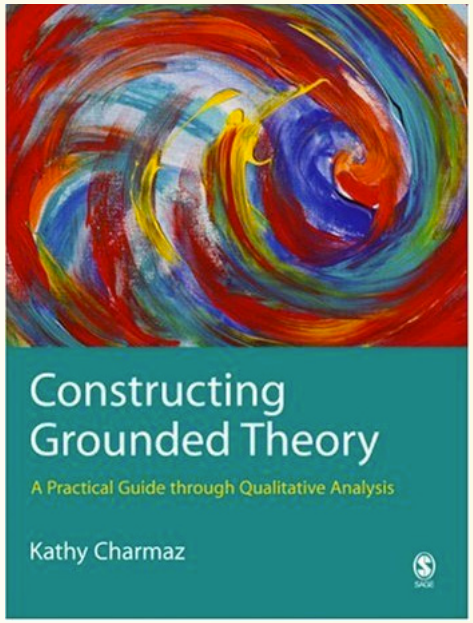
Coding Handbook Example

Diagrams

- Definition: Visual images used to explain the program to the audience
- Example:

Appendix F

Grounded theory



"

Stated simply, grounded theory methods consist of **systematic, yet flexible guidelines for collecting and analyzing qualitative data to construct theories 'grounded' in the data themselves…**

We study our early data and begin to separate, sort, and synthesize these data through qualitative coding. Coding means that we attach labels to segments of data that depict what each segment is about… **Grounded theorists emphasize what is happening in the scene when they code data…**

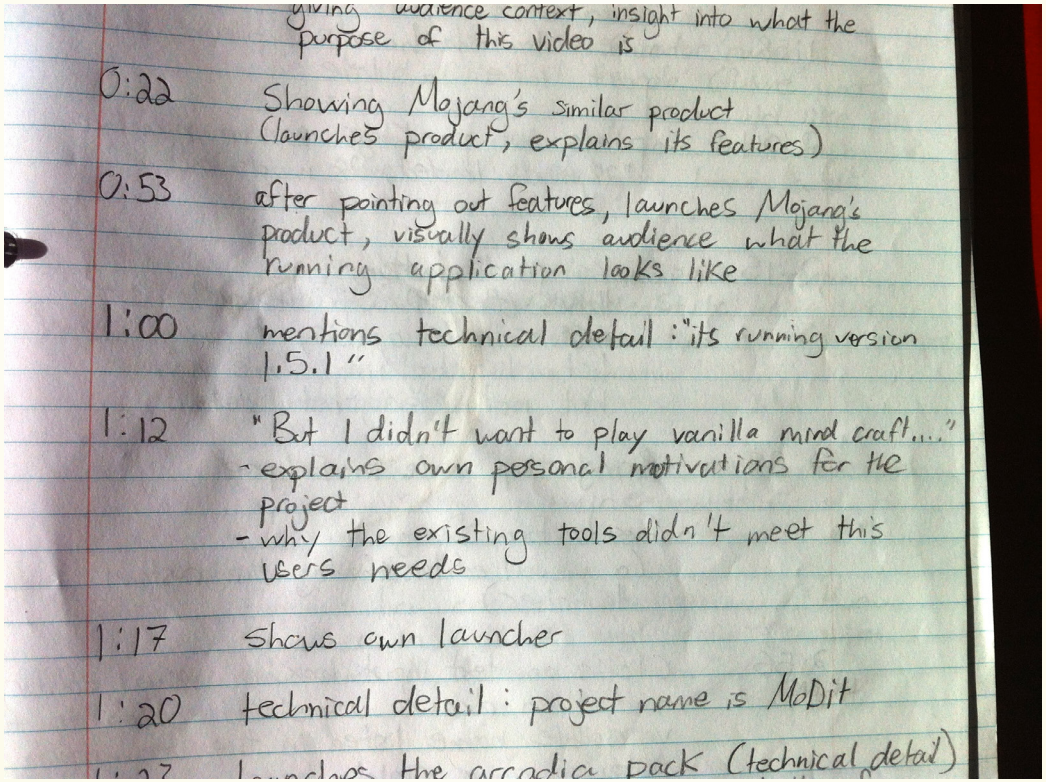
We build levels of abstraction directly from the data and, subsequently, gather additional data to check and refine our emerging analytic categories. **Our work culminates in a 'grounded theory,' or an abstract theoretical understanding of the studied experience.**

"

- Charmaz [2]

Appendix G

Open coding



giving audience context, insight into what the purpose of this video is

0:22 Showing Mojang's similar product (launches product, explains its features)

0:53 after pointing out features, launches Mojang's product, visually shows audience what the running application looks like

1:00 mentions technical detail: "its running version 1.5.1"

1:12 "But I didn't want to play vanilla mind craft...."
- explains own personal motivations for the project
- why the existing tools didn't meet this users needs

1:17 Shows own launcher

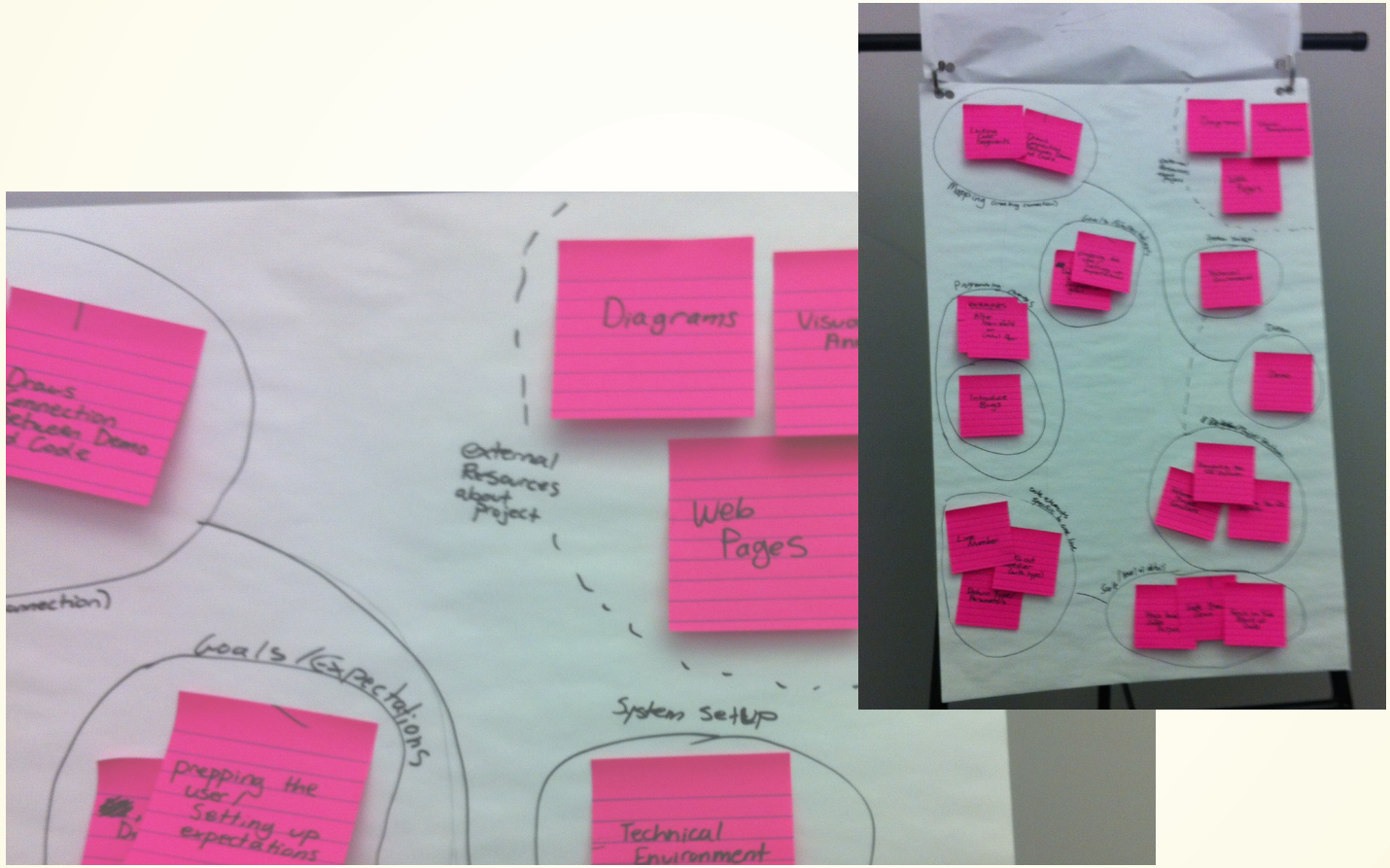
1:20 technical detail: project name is MoDit

1:27 launches the arcadia pack (technical detail)

An example of an initial attempt at open coding a video

Appendix H

Context Mapping



An example of context mapping done by two of the researchers involved in this study

Appendix I

Interviews

10 screencast creators

Demographics:

- All male, from North America, Europe and Australia
- English speaking
- All had some education in computer science
- Three were current or former educators