Facilitorials

Jay Silver
Media Lab, MIT
20 Ames St.
Cambridge, MA 02139 USA
+1 321 223 3825
sil@mit.edu

ABSTRACT
Facilitorials are tutorials that try to act more like a facilitator. By mimicking a real facilitator, facilitorials end up prioritizing a healthy learning environment over maximal transfer of information. Facilitorials wait to be pulled rather than pushing themselves, and when they’re pulled they try to communicate by doing more often than by saying. This is the first introduction of the word facilitorial. The demo will include video facilitorials for the Scratch\(^1\) programming language. [1]

Keywords
Facilitorial, facilitator, tutorial, video, Scratch, program

ACM Classification Keywords
K.3.1 [Computers and Education]: Computer uses in education – computer assisted facilitation (CAF)

INTRODUCTION
Facilitorials can come in any of the formats that tutorials come in including: paper, video, computer, and live session. In this paper, I’ll focus on the format I have the most experience with (and plan to demo): video facilitorial.

Facilitators
When I refer to facilitators, I imagine a shop situation. The shop has all kinds of tools and a person who is there to help you get to know how to use the tools. That person is called a facilitator, and it is their job to create an environment where you can feel comfortable working on your project as you become more familiar with the tools of your art. A facilitator provides “…unobtrusive assistance, guidance, or supervision.” [2] When I facilitate workshops with Scratch, the primary tool is a computer programming language.

Creating an environment
I can’t possibly know what you want to do with the tools when I make a video [4]. So, rather than me deciding what set of information you ought to know, I want to create an environment, a shop, where it seems like a good idea to try stuff out. I keep this in mind whether I’m writing the script for the video, doing the voice-overs, or editing.

Communicating through actions
Take the example of making mistakes. Students are not explicitly told: “It’s not okay to make mistakes.” Rather, they are implicitly shown: presentations of new material are often so carefully rehearsed by expert teachers that mistakes are never seen. For example, it is quite rare indeed for someone to do something wrong in a tutorial video. If I want to communicate that it is okay to make mistakes, I will also have to speak the language of actions by making mistakes in my videos. And, I will have to really believe it, as a lot is communicated in as little as a glance [6].

METHOD
Software
In order to show parts of a computer program, I used a screen capture program\(^2\) called Camtasia. I edited the video in Windows Movie Maker. I also recorded the voice-overs in Movie Maker with a regular computer microphone. I try to keep things simple rather than flashy, and personal rather than professional. I highlight mouse clicks visually using a feature built into Camtasia and by recording the sound the clicks make too.

Show and Tell
One simple thing I do is just to think of the facilitorial as a time for show and tell, very casual. I use the first person perspective as I show a couple of different ways that I like to get started in Scratch. I show how I experiment with the blocks to see what happens as opposed to showing what the blocks themselves do. Incidentally, in the process, it does become apparent what some of the blocks do.

Applying Lessons from the Field
This semester I am facilitating a workshop on the Scratch software in conjunction with Citizen Schools. After each session, I am taking notes on specific techniques that can be used for facilitating Scratch. I then try to distill which aspects of the human facilitator could be embedded in

---

1 Scratch is a programming language where you can make stories, games, and interactive art by stacking Lego-like blocks.

2 A screen capture program lets you take video recordings of what happens on the computer screen.
a document such as a video. I am also introducing Scratch to creators (typically called learners) who have never seen it. I am recording this process using screen capture software to closely analyze which features of the program can easily be discovered and which concepts are more confusing. The current facilitorial is available on the Scratch website [1] and via streaming video on youtube.com where it has been watched more than 5,000 times.

AN INTRODUCTORY FACILITORIAL
Now I'll show you an excerpt from a facilitorial that I made [3] for introducing Scratch to people who had never seen it. In this script excerpt, I first demonstrate how I like to play around with the sample projects to see what happens:

“I'm going to change the code that runs the boom box. I'm just going to go through while it's running and change which drums are being played in the dropdown menu. I don't always know what will happen, but I just try it and see. I'm not worried about breaking anything. In fact I like to break the programs apart. I can even break the blocks apart while they're running to mess up the program and see what happens. That's one way I get started. But another way I get started is to start from a blank screen.”

I use the first person throughout as I describe what I do while I'm doing it. I poke and prod the code blocks to see what they do, and I show how I program without always knowing what is going to happen. As I click on the blocks in the video, some people will notice how the program flow changes as a result. Others will get a feel for how blocks can be dragged. There’s a lot of information, and it’s not important that everyone get the same thing from watching the process. Not everyone will feel comfortable starting this way, so I go on to talk about another way to get started.

ONGOING WORK
I am continuing to work on other facilitorials for Scratch. I am trying to make a set of very short and pithy facilitorials that hit on several commonly needed topics. I hope to combine this quick style with a slower paced style in a multi-speed video³, which lets the creator decide whether a shorter or longer explanation would be more helpful. In the work I’ve done or the work I plan to do, I want to be clear that I don’t think facilitorials should take the place of a human facilitator. However, I see facilitorials as helpful when no facilitator is present.

THE DEMO
The demonstration will consist of a video facilitorial combined with the option to try out the programming language. Ideally I will create a library of videos that you can refer to as you like while tinkering with the software. And, if we’re all lucky, there will be a multi-speed video to interact with.

THE GOAL
Facilitorials are meant, first, to grow curiosity, confidence, compassion, fearlessness... Yet I would offer a facilitorial in a place where you might find a tutorial, and a tutorial “provides practical information about a specific subject.” [2] Wouldn’t I like students to have information on the subject? Yes, but given the choice I’d rather provide an environment where creators can touch tools [5]. And, from experience and experiments creators can discover information in the midst of a wonderful idea. The facilitorial approach assumes that information is not a condensed list of facts, but rather part of a process.

ACKNOWLEDGMENTS
Thanks to the continual feedback from my group, Life Long Kindergarten, and my advisor, Mitch Resnick.

REFERENCES
1. Scratch programming language and the first video facilitorial available at http://scratch.mit.edu

³ By multi-speed I don’t just mean the same video played back faster or slower. I mean videos of different length and pace covering the same logical content.