

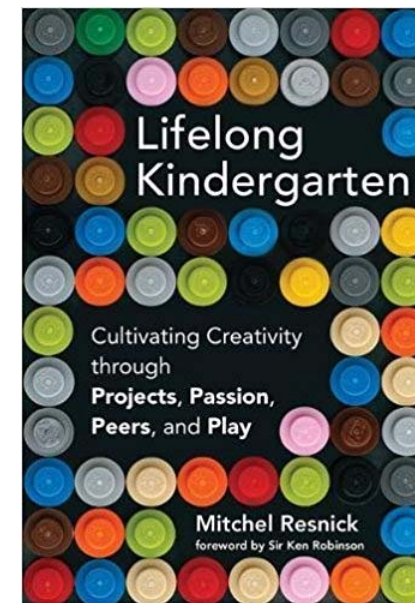
---

## Scratch vs. learn-to-code-websites

---

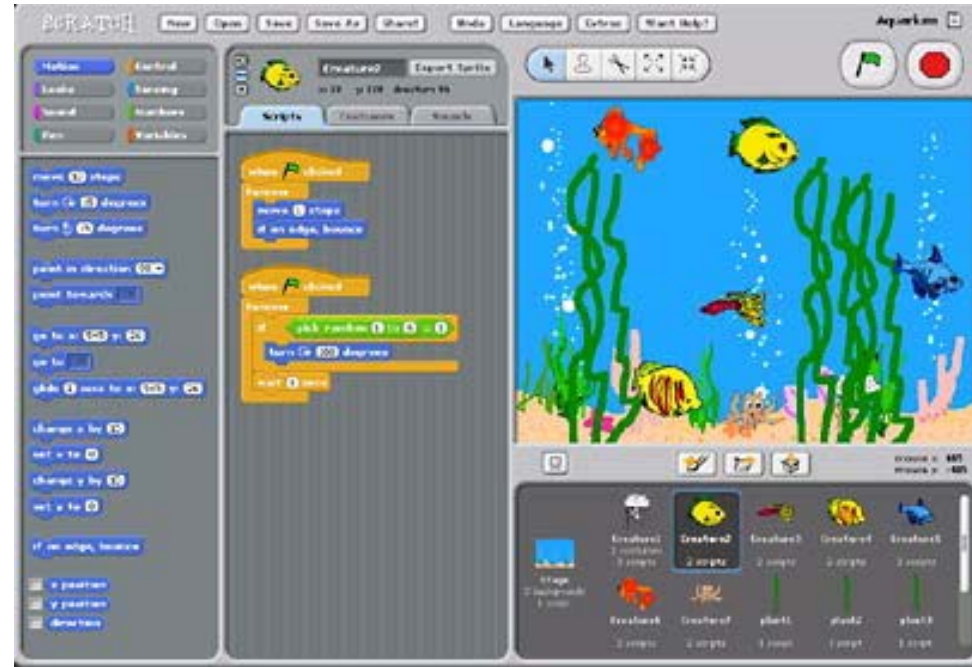
*“The Scratch approach to personalization stands in striking contrast to most learn-to-code websites, which introduce kids to coding through a series of puzzles to solve. Because the puzzles are standardized, the sites can track kids’ progress and offer personalized instruction and advice—but they provide kids with little or no opportunity for personal expression.*

*The Scratch approach is diametrically opposed. Because kids can create anything they want with Scratch, it’s difficult to automatically provide feedback or guidance — but the payoff comes in connecting with kids’ interests and catalyzing their imaginations.”*



Warum

# SCRATCH ?



You are here: [Beats Weblog](#) » Gedanken zu Programmierumgebungen für "Medien und Informatik"

## Gedanken zu Programmierumgebungen für "Medien und Informatik"

📅 21 May 2016 - 15:16 | 🗨️ Version 4 | 👤 [TobiasH](#)

In den deutschsprachigen Kantonen und Pädagogischen Hochschulen laufen derzeit die Vorbereitungen zur Einführung des **Lehrplans 21** ([Biblionetz:w02172](#)) auf Hochtouren, insbesondere auch für das neue Modul "Medien und Informatik". Eine nicht zentrale, aber trotzdem nicht ganz unwichtige Frage ist die nach Programmierumgebungen.

Als Diskussionsbasis in verschiedenen Kontexten habe ich mich mal hingesetzt und meine diesbezüglichen Überlegungen etwas strukturiert. Es handelt sich somit um eine Konkretisierung der allgemeinen Frage **Welche Programmiersprache eignet sich für die Schule?** ([Biblionetz:f156](#)) auf die aktuelle Situation in der Schweiz und dem bevorstehenden Lehrplan 21.

(direkt kommentiert werden kann dieser Text drüber bei [Google Docs](#))

- ↓ Gedanken zu Programmierumgebungen für "Medien und Informatik"
  - ↓ 1. Worum geht es?
  - ↓ 2. Wie sieht die Ausgangslage aus?
  - ↓ 3. Warum braucht es Empfehlungen?
  - ↓ 4. Eine oder mehrere Umgebungen?
  - ↓ 5. Was sind die Anforderungen an eine Programmierumgebung für das Modul Medien und Informatik?
    - ↓ 5.1 Didaktisierte oder professionell genutzte Sprache?
    - ↓ 5.2 Textbasierte Sprache oder Blocksprache?
  - ↓ 6. Empfehlung: Scratch & scratchkompatible Sprachen
    - ↓ 6.1 Was ist Scratch?
    - ↓ 6.2 Warum Scratch?

---

# Voraussetzungen

---

- **Benötigt wird eine Programmierumgebung für die 5./6. Klasse Primarschule und die Sekundarstufe I**
- **Es geht um Allgemein- und nicht Berufsbildung**
- **In der 5./6. Klasse keine Fachlehrpersonen**
- **Derzeit wenig bis kein Informatikvorwissen bei den Lehrpersonen**
- **Beschränkte Ressourcen zur Weiterbildung von Lehrpersonen, die "Medien und Informatik" unterrichten werden**
- **Derzeit wenig kohärentes, zum Modullehrplan "Medien und Informatik" passendes Unterrichtsmaterial**
- **Heterogene Infrastrukturvoraussetzungen in den Schulen.**

---

## Warum braucht es Empfehlungen?

---

- **Mangelndes Vorwissen der Lehrpersonen:** Weil die meisten Lehrpersonen bisher über wenig Informatik-Kompetenzen verfügen, wären sie mit der Auswahl selbst überfordert.
- **Wenig Material vorhanden:** Weil bisher wenig stufenspezifisches Unterrichtsmaterial existiert, fördern konkrete Umgebungsempfehlungen die Nutzbarkeit des zu entwickelnden Materials

---

## Eine oder mehrere Umgebungen?

---

- Parallel mehrere Umgebungen?
- Andere Umgebung in der Primarschule als auf der Sek-Stufe?

---

# Anforderungen an eine Sprache für M+I

---

- **Motivierend**
- **Leichter Einstieg**
- **Vielseitig verwendbar**
  - **Sprachumfang/Ausrichtung**
  - **Technische Verfügbarkeit**
  - **Lizenzrechtliche Verfügbarkeit**
- **Längerfristig verfügbar**
- **Mächtig**
- **Enaktiv**

# Didaktisiert oder professionell genutzt?

The image displays a screenshot of an IDE (likely Eclipse) showing a Java web application project. The interface is divided into several panes:

- Top Left:** A window titled "Kara, der programmierbare Märkteifer" showing a lifecycle diagram with a "start" button, a "track" circle, and a "Stop" circle.
- Top Middle:** A "Project Explorer" showing the project structure, including "Web Pages", "images", "javascripts", "stylesheets", "index.html", "Source Packages", "Libraries", "Configuration File", "Bean Validation API", and "Source Packages".
- Top Right:** A "Source" editor showing the code for "AjaxCometServlet.java". The code includes annotations like "@WebServlet" and "public class AjaxCometServlet extends HttpServlet".
- Bottom Left:** A "Members" pane showing the methods of the "AjaxCometServlet" class, including "destroy()", "doGet()", "doPost()", "escape()", "init()", "notify()", "toJson()", "BEGIN\_SCRIPT\_TAG", and "END\_SCRIPT\_TAG".
- Bottom Middle:** A "Javadoc" pane showing the documentation for the "destroy()" method: "Called by the servlet container to indicate to a servlet that the servlet is being taken out of service. See Servlet#destroy."
- Bottom Right:** A "Hierarchy" pane showing the class hierarchy for "AjaxCometServlet", including "HttpServlet", "GenericServlet", "Object", "Serializable", and "Servlet".



---

# Grafisch/Blockbasiert oder textuell?

---



Python script

```
1 def azureml_main(expr_as_frame):
2     import pandas as pd
3     expr = expr_as_frame.iat[0,0]
4     result = pd.DataFrame({'Expr': [expr], \
5                             'Result': [eval(expr)]})
6     return result,
```

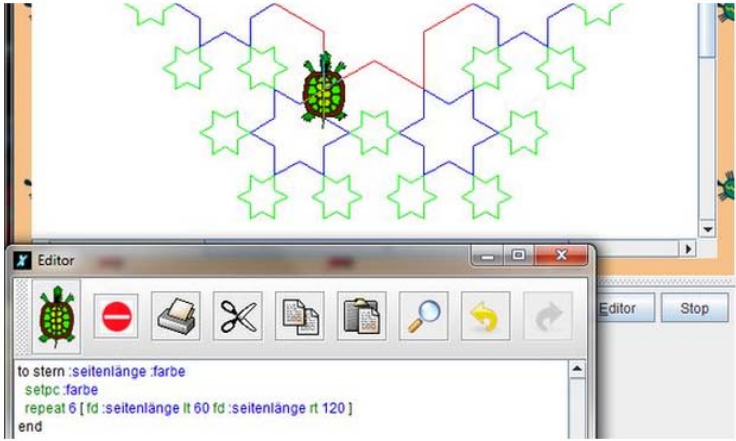
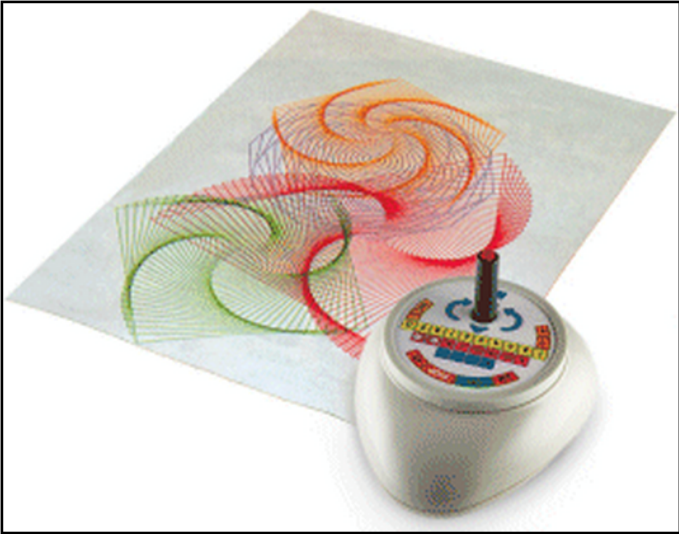
[doebe.li/a1273](https://doebe.li/a1273) Blockbasierte Sprachen verhindern Syntaxfehler

[doebe.li/a1274](https://doebe.li/a1274) Blockbasierte Sprachen erleichtern den Einstieg ins Programmieren

---

phsz

# Edu-Programmiersprachen gibt es wie Sand am Meer...



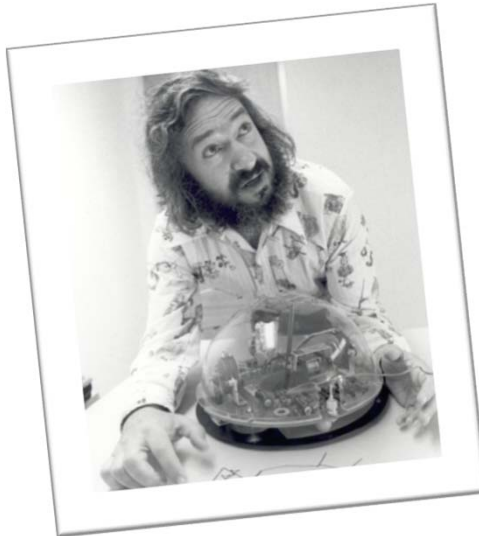
---

# Die Geschichte von Scratch

---



Jean Piaget:  
Konstruktivismus



Seymour Papert:  
Konstruktivismus  
& Logo



forward 50



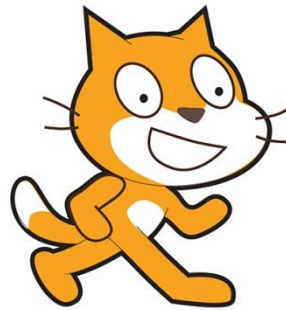
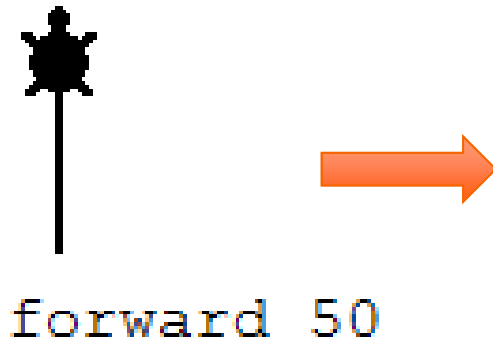
Mitchel Resnick:  
NetLogo &  
Scratch



---

# Von Logo zu Scratch

---



Mitchel Resnick (2012)

***“Learning lessons from Papert’s experiences of Logo, we’ve designed Scratch to move beyond Logo along three dimensions, making programming more tinkerable, more meaningful, and more social.”***

---

# Grundprinzipien von Scratch

---

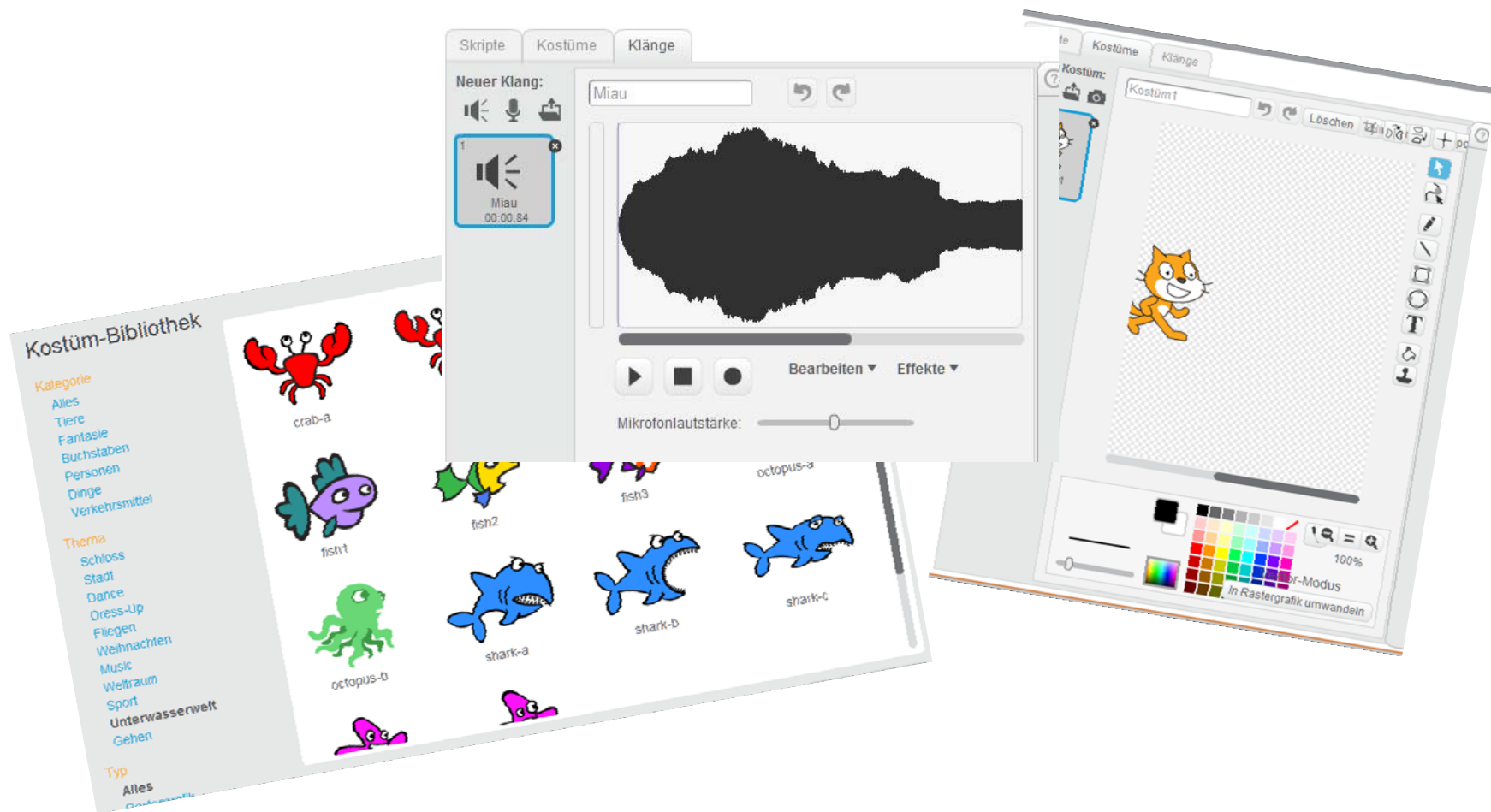


«more meaningful»

«more tinkerable»

«more social»

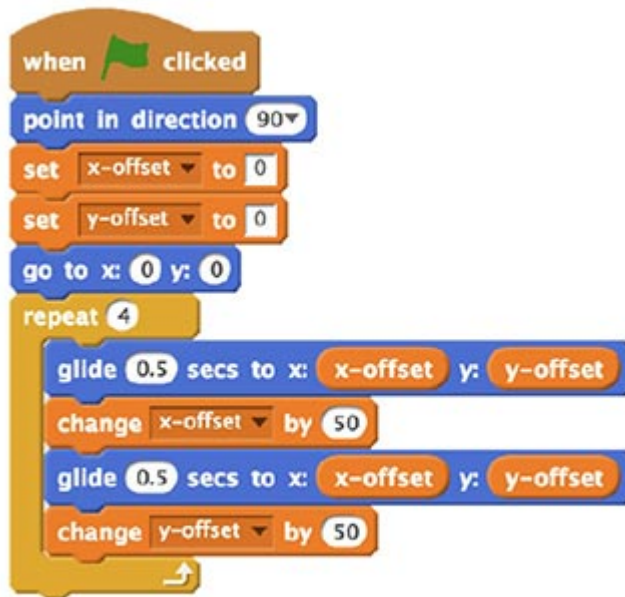
# imagine : More meaningful



---

## program : More tinkerable: block vs. text

---



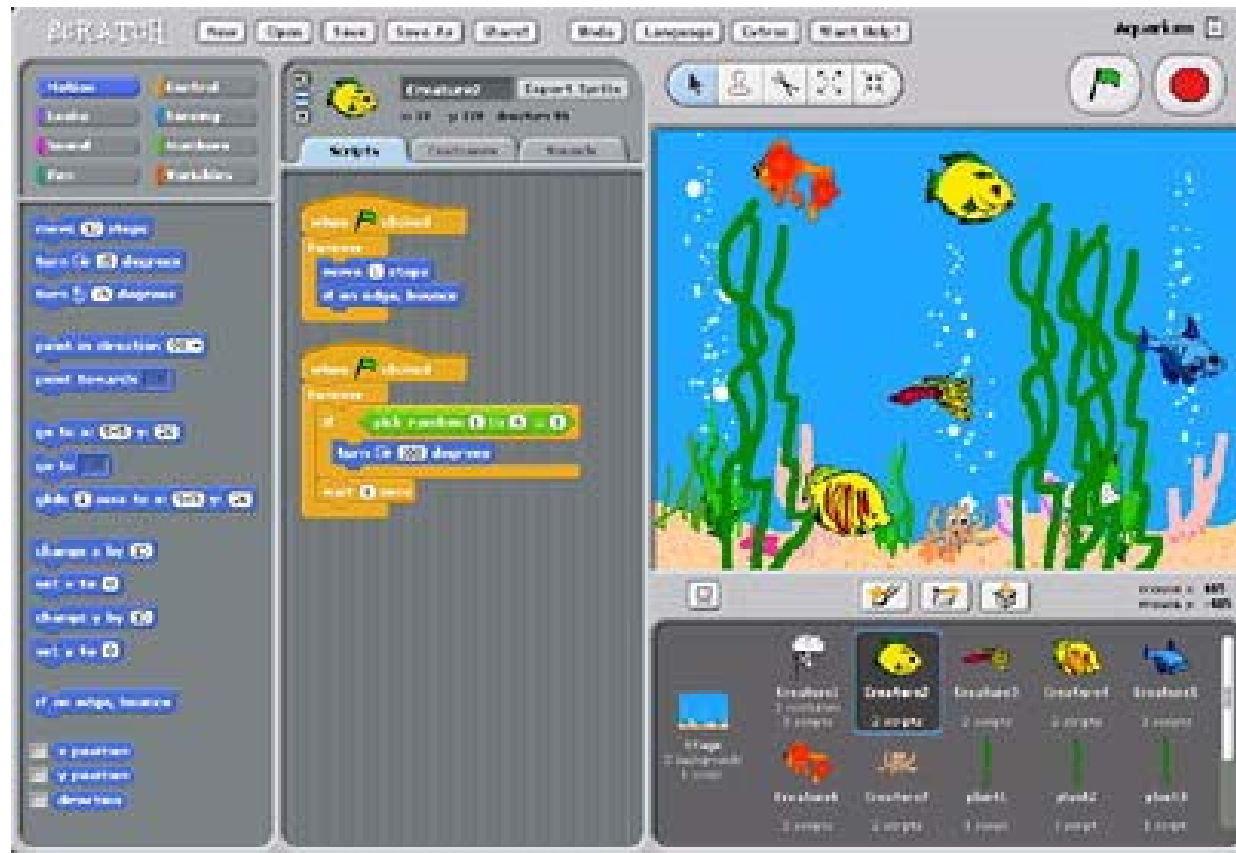
```
$(document).ready(function(){
  var $sprite = $('sprite');
  var xOffset = parseInt($('sprite').css('left').
    replace(/px/, ''), 10);
  var yOffset = parseInt($('sprite').css('top').
    replace(/px/, ''), 10);
  var i;

  for (i = 0; i < 4; i++) {
    xOffset += 50;
    yOffset -= 50;
    $sprite.animate({
      left: xOffset
    }).animate({
      top: yOffset
    });
  }
});
```

---

# program : More tinkerable: live coding

---

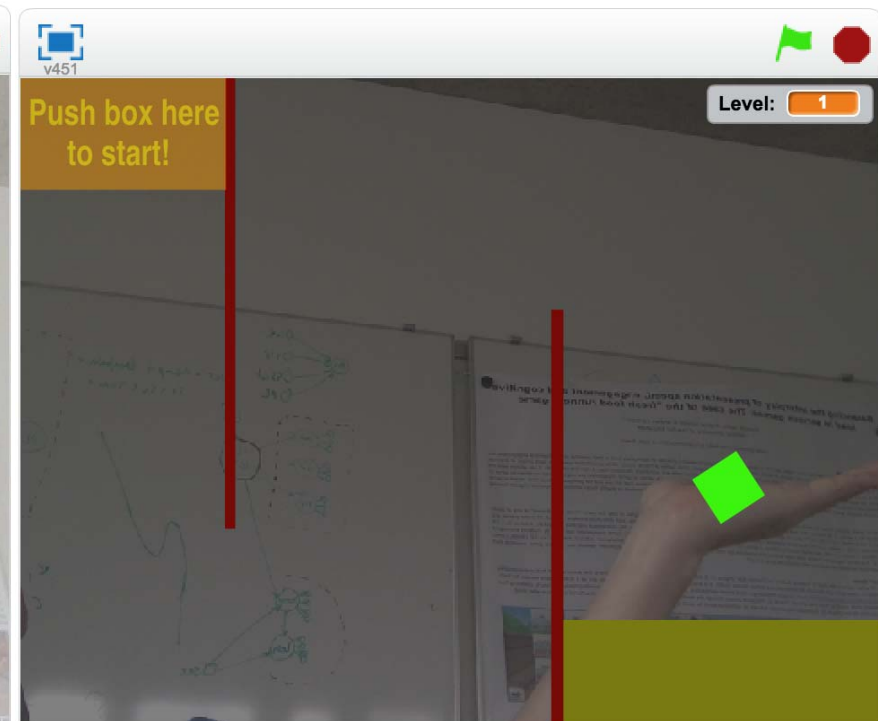
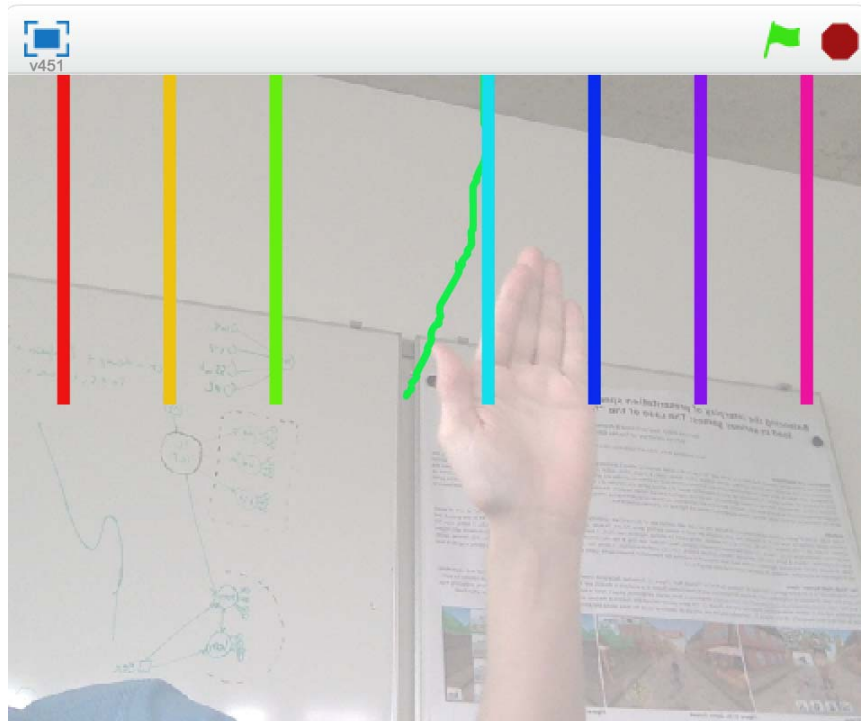


Herangehensweise: Bottom-up statt top-down





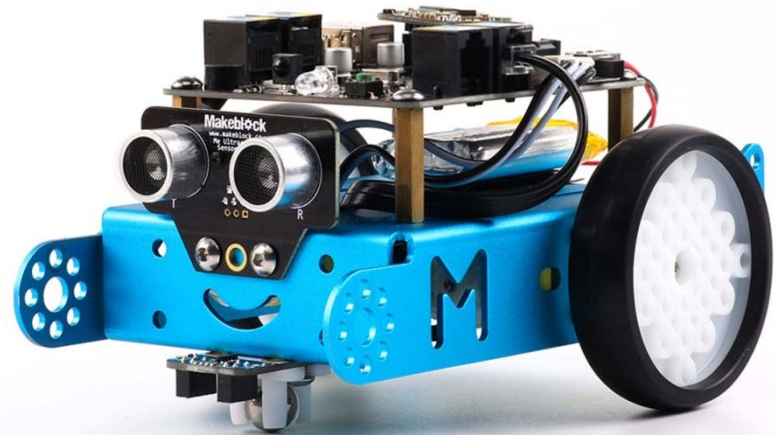
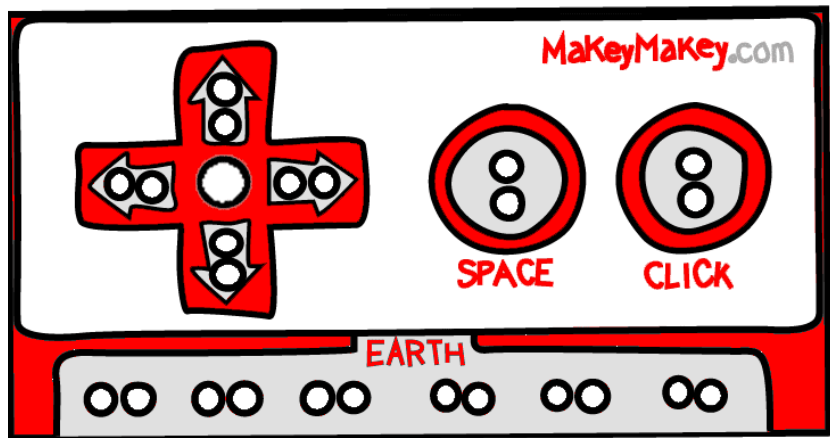
## : More tinkerable: Sensoren und Aktoren



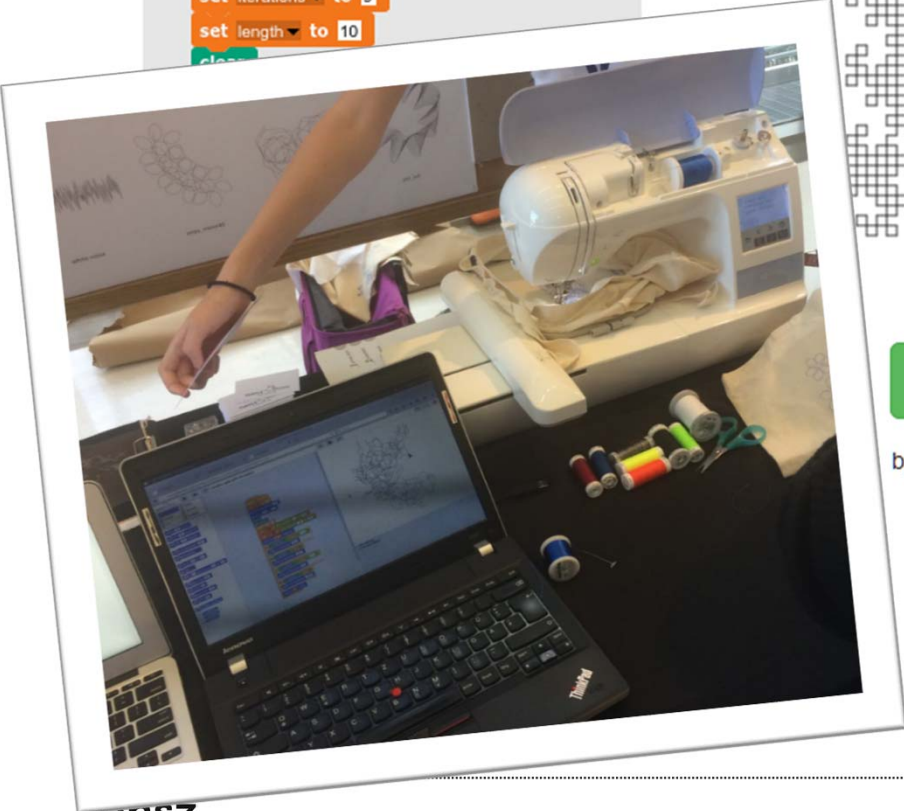
<https://scratch.mit.edu/studios/201435/>

**program**

# : More tinkerable: Sensoren und Aktoren



code! - draw! - stitch!



Run now!  
by STITCHCC





# : More sociable: ScratchR

Erschaffe Geschichten, Spiele, und Animationen  
Teile sie mit anderen weltweit



Eine kreative Lerngemeinschaft mit **17.758.764** geteilten Projekten

[ÜBER SCRATCH](#) | [FÜR LEHRKRÄFTE](#) | [FÜR ELTERN](#)

### Vorgestellte Projekte



### Vorgestellte Studios



### Projekte vorgestellt von Kurator ForeverArtist

[Mehr erfahren](#)



## Community statistics at a glance

- 17'731'978** projects shared,
  - 14'435'279** users registered,
  - 93'664'866** comments posted,
  - 2'970'475** studios created
- ...and growing!

## Website traffic last month

- 144'777'540** pageviews
- 27'845'820** visits
- 13'540'550** unique visitors

scratch.mit.edu: Das YouTube des Programmierens



## : More sociable: ScratchR

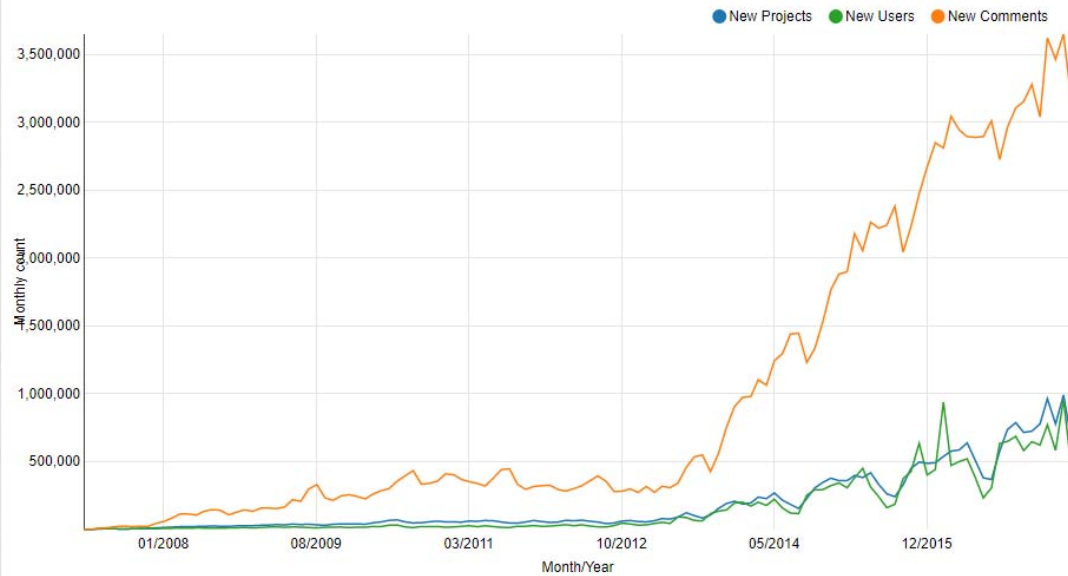
### Community statistics at a glance

- 🐱 24.028.090 projects shared,
  - 🐱 19.961.818 users registered,
  - 🐱 123.327.576 comments posted,
  - 🐱 3.633.643 studios created
- ...and growing!

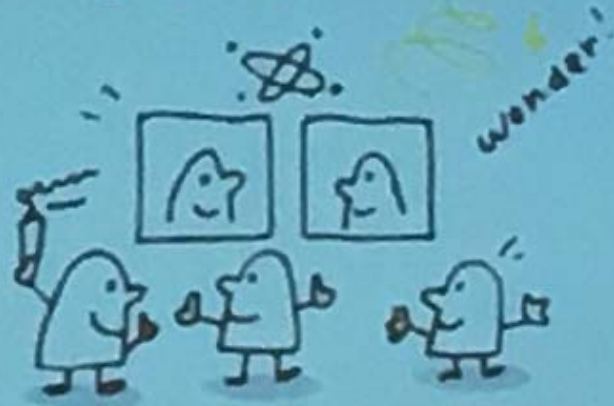
### Website traffic last month

- 🐱 100.126.650 pageviews
- 🐱 17.559.420 visits
- 🐱 9.076.330 unique visitors

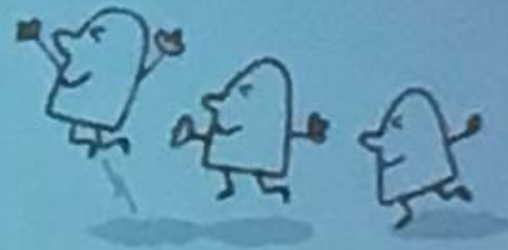
### Monthly Activity Trends



projects



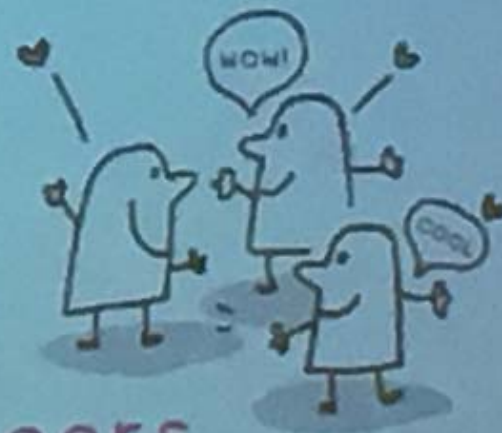
passion



play



peers



# Passion

it is all about falling in Love with the material!

EDUCATION HAS VERY LITTLE TO DO WITH EXPLANATION



Seymour grew up in South Africa - he wanted education to be available for ALL CHILDREN



Good morning



IF THINGS DON'T WORK THE WAY YOU LIKE, IT IS AN OPPORTUNITY TO LEARN...

change things, TRY AGAIN!

# Play

ENGAGE KIDS TO EXPERIMENT & TINKER

Purpose: Give a voice to everyone

BUILDING AUTOGRADERS FOR **SNAP** TO INTEGRATE THEM INTO YOUR COURSE...!

SNAP SHOULDN'T LOOK REALLY DIFFERENT TO STUDENTS...

by Michael Ball, Dan Garcia & Lauren Mock

WE GIVE THEM FEEDBACK AS THEY GO ON.



# Scratch

BORDEAUX 2017

GIVE THEM 3 PERSPECTIVES, IF YOU WANT TO EXPLAIN:

- 1 How does it work?
- 2 What are the effects?
- 3 How do I use it?

IN BETWEEN IS THE POLITICIAN. (who is NOT an expert!)

THE **Scratch** way  
"Just do it!"

## 21 cantons

EACH ONE HAS ITS OWN EDUCATION SYSTEM... OOF!

So...

"MEDIEN & INFORMATIK" can be a very different thing for 2 swiss children

IN SWITZERLAND THEY KNOW HOW TO EXPLAIN DIGITALIZATION TO POLITICIANS!



by Beat Döbeli Honegger



DON'T TOUCH THAT COMPUTER, YOU DON'T HAVE YOUR COMPUTER DRIVING LICENSE YET!

picknick sur l'herbe



HEY GUYS, THE FOOD TRUCK IS HERE AGAIN!

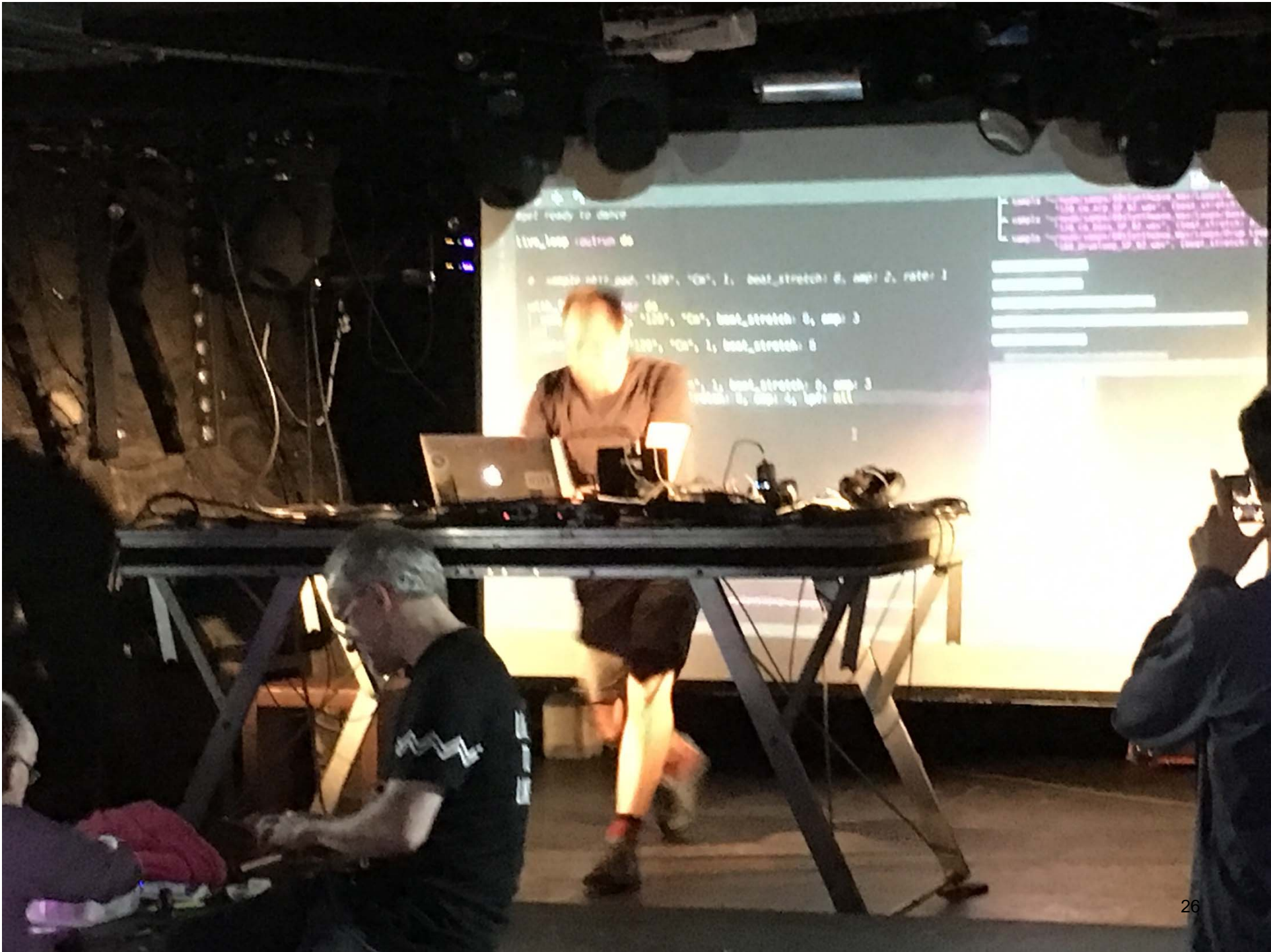


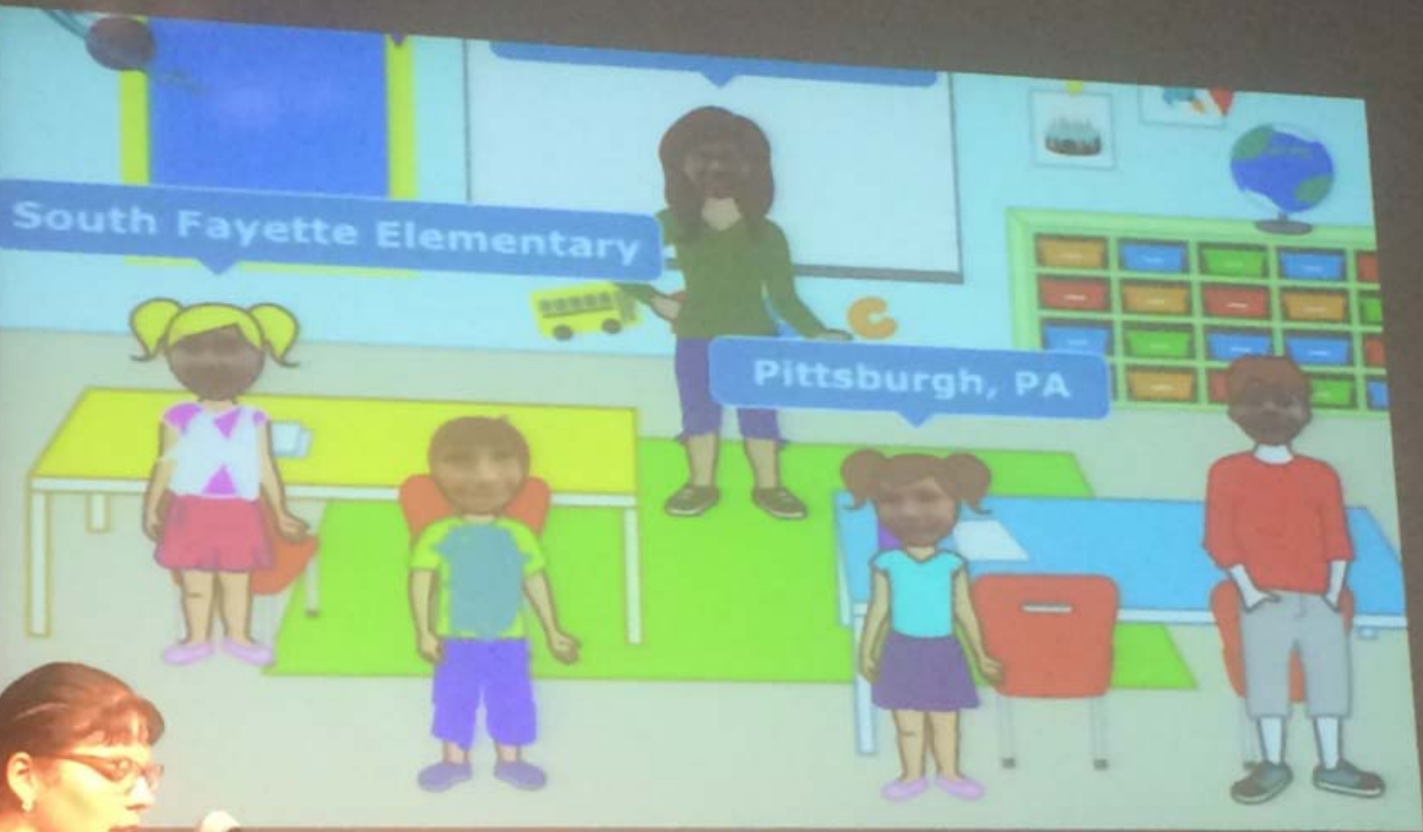


COCENDI-IPN:  
Mexican Experience  
in Early Childhood Education  
with Technology

Ecogsten for the educational technology  
development and training teams  
in science, technology and  
innovation

Cocendi-IPN: A Mexican  
experience in early  
childhood education  
technology

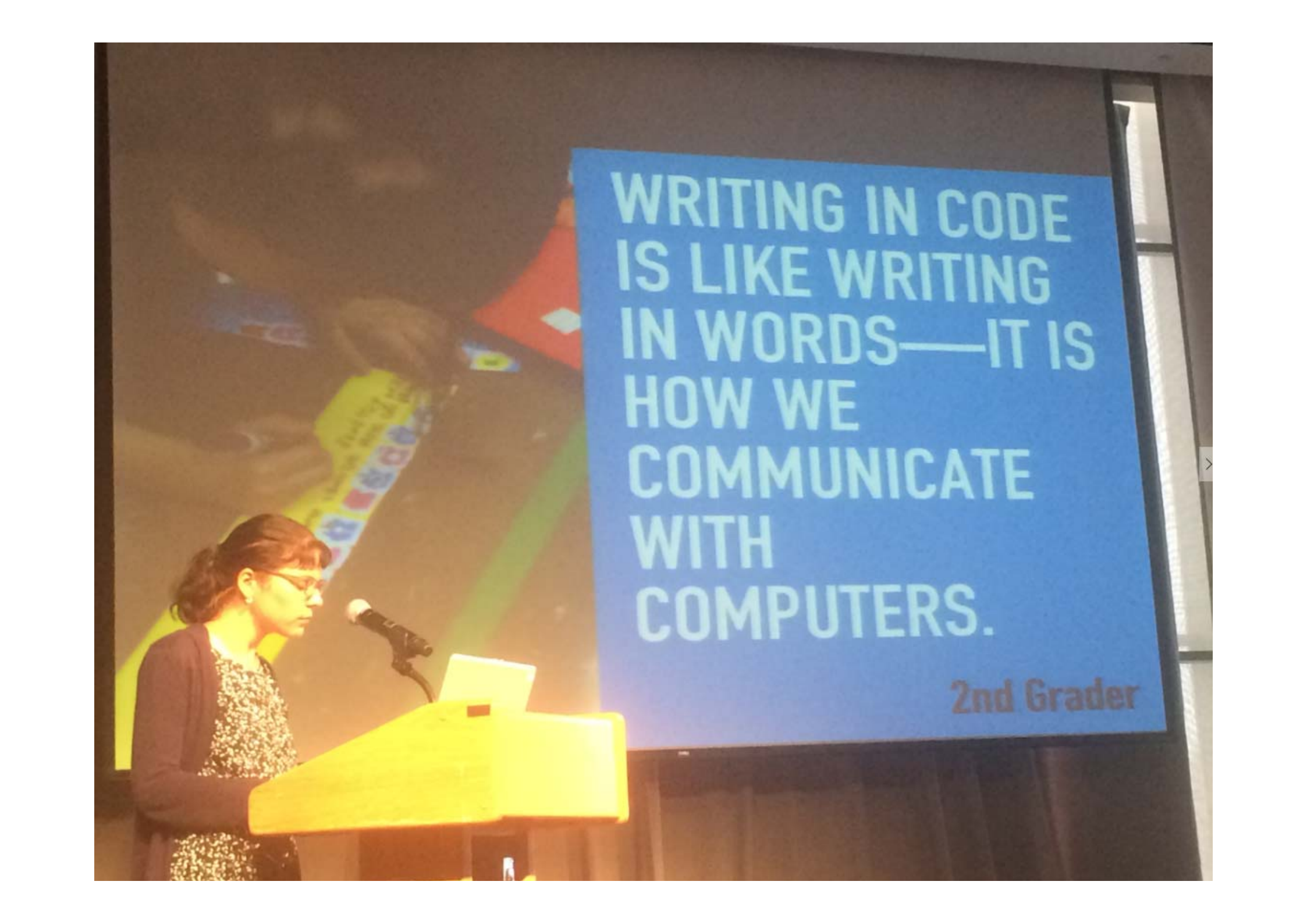




K-2 CODE KIDS

[SOUTHFAYETTE.ORG](http://SOUTHFAYETTE.ORG)



A woman with glasses and a dark cardigan over a patterned top stands at a yellow podium, speaking into a microphone. Behind her is a large blue screen with white text. The screen also shows some faint, colorful graphics on the left side, including a red triangle and a yellow banner with illegible text. The scene is lit with warm stage lights.

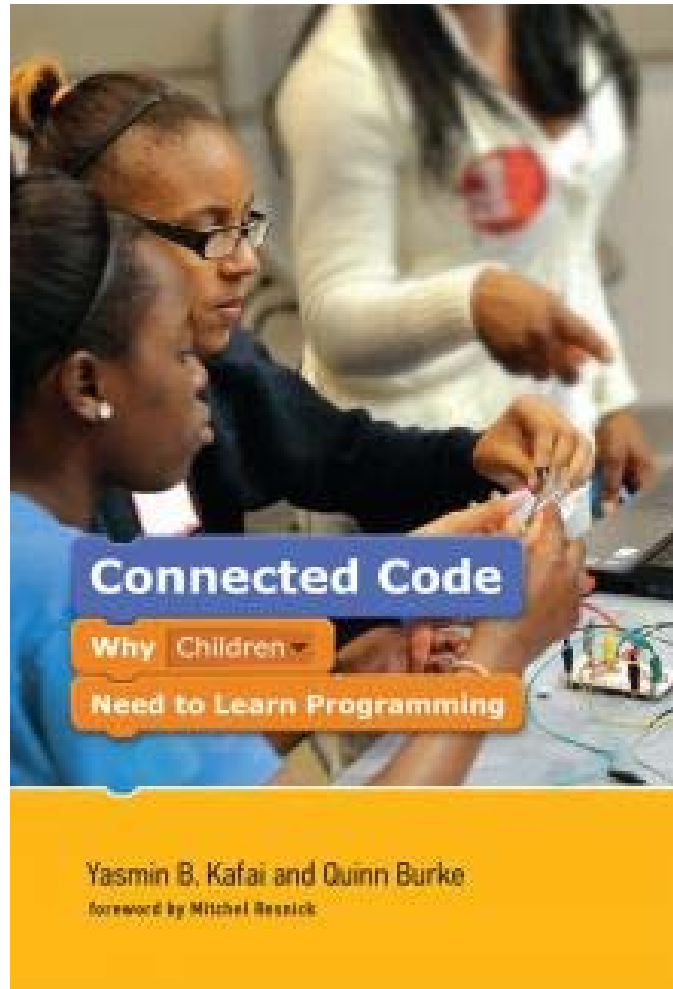
WRITING IN CODE  
IS LIKE WRITING  
IN WORDS—IT IS  
HOW WE  
COMMUNICATE  
WITH  
COMPUTERS.

2nd Grader

---

# Literaturtips

---



«computational participation»

[doebe.li/b5889](https://doebe.li/b5889)