Discovering Your Software Umwelt

Rebecca Wirfs-Brock *(Wirfs-Brock Associates)*

Allen Wirfs-Brock *(Wirfs-Brock Associates)*

Jordan Wirfs-Brock *(Whitman College)*

Rebecca

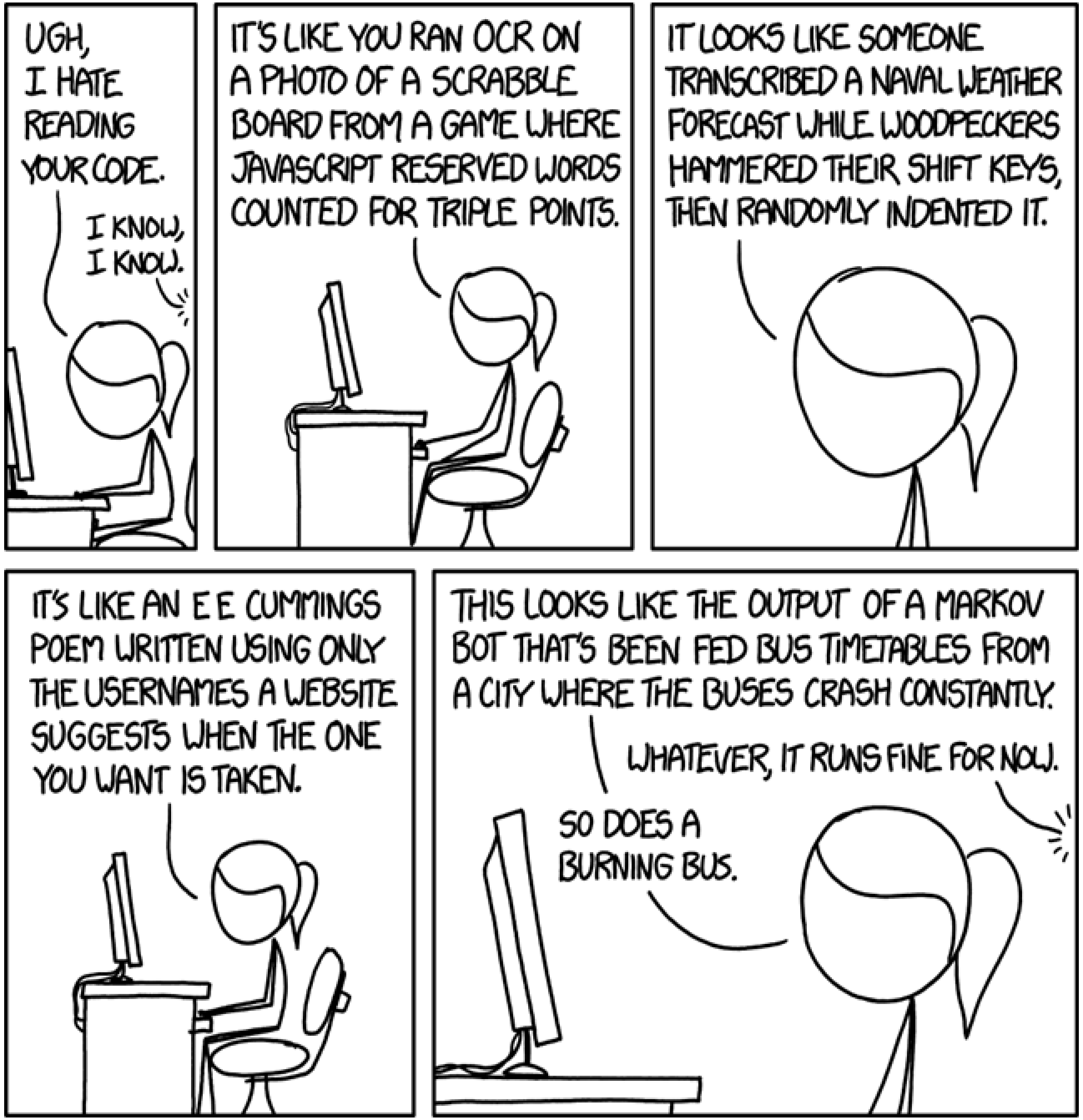
embedded software, object design pioneer, books, consulting, modeling, design, architecture, communication, and thinking skills

Allen

system programmer, language design and implementation, compilers, Smalltalk VMs & GCs
Smalltalk & JavaScript standards

Jordan

aerospace engineering undergrad, data journalist, Info Science PhD, CS Prof, HCI researcher, data sonification


UGH, I HATE READING YOUR CODE.
I KNOW, I KNOW.
IT'S LIKE YOU RAN OCR ON A PHOTO OF A SCRABBLE BOARD FROM A GAME WHERE JAVASCRIPT RESERVED WORDS COUNTED FOR TRIPLE POINTS.
IT LOOKS LIKE SOMEONE TRANSCRIBED A NAVAL WEATHER FORECAST WHILE WOODPECKERS HAMMERED THEIR SHIFT KEYS, THEN RANDOMLY INDENTED IT.
IT'S LIKE AN E E CUMMINGS POEM WRITTEN USING ONLY THE USERNAMES A WEBSITE SUGGESTS WHEN THE ONE YOU WANT IS TAKEN.
THIS LOOKS LIKE THE OUTPUT OF A MARKOV BOT THAT'S BEEN FED BUS TIMETABLES FROM A CITY WHERE THE BUSES CRASH CONSTANTLY.
WHATEVER, IT RUNS FINE FOR NOW.
SO DOES A BURNING BUS.

Why can’t software developers understand each other?

https://xkcd.com/1695/

One reason we have trouble communicating with each other is we each have a different…

Umwelt:

our unique perceptions of our world and the actions we deem appropriate to affect it.

*Our umwelts are **personal** and **unique**.*

Umwelt: A Deeper Look

An Immense World by Ed Yong

An Immense World: How Animal Senses Reveal the Hidden Realms Around Us

ISBN-13: 9780593133231

“Our Umwelt is still limited; it just doesn’t feel that way. ***To us, it feels all-encompassing****. It is all that we know, and so we easily mistake it for all there is to know. This is an illusion...”*

— Ed Yong, *An Immense World: How Animal Senses Reveal the Hidden Realms Around Us*

“A moth will never know what a zebra finch hears in its song, a zebra finch will never feel the electric buzz of a black ghost knifefish, a knifefish will never see through the eyes of a mantis shrimp, a mantis shrimp will never smell the way a dog can, and a dog will never understand what it is to be a bat. ***We will never fully do any of these things either, but we are the only animal that can try.****”*

— Ed Yong, *An Immense World: How Animal Senses Reveal the Hidden Realms Around Us*

The Dilemma

You can’t see your own eyes or hear your own ears….

So how do you come to **understand your own *umwelt*?**

Our Experiment

Let’s write narrative **stories about our personal software umwelts** and **share them** with each other.

What can we learn???

Articulating our umwelts through key experiences...

We compared our umwelt narratives (talked about them together), realized that we had these similar threads and salient moments – origins, how our practice grew, pivots, etc.

We turned those into **prompts** that can help others **identify and reflect on their own umwelt**.

Jordan’s Personal Umwelt Summary

“My software umwelt is tightly coupled with my familial umwelt, and is a matter of legacy and identity. It has resulted in ***pride and curiosity****. Writing software requires a* ***type of thinking that we don’t often encounter*** *in other dimensions of our lives. We need to* ***empathize*** *with how other people relate to programming. There are many ways to program, and embracing them leads to broader and deeper perspectives and ways of doing things.* ***Software development is a sociotechnical system****. I’m always looking at systems, from a standpoint of trying to figure out what people and tools are required to navigate them. But the real power of software is that we can make things that people use—****technologies that change the world****. But how do we know that we are making something people want? That they need? That doesn’t harm them? To consider these questions, we must always remember to to* ***pull back and see the bigger picture****.”*

Allen’s Personal Umwelt Summary

“Software is **my creative media** *and* **programming is an intellectual activity** *that can be done even in the absence of a physical computer. I’m a* **systems programmer***. I value technical diversity,* **expect** *ongoing evolutionary* **change***, and* ***relish radical innovations****.* ***Lateral thinking is essential*** *for software innovation. I believe that s****oftware exists to support humans*** *and human activities. It should be approachable, understandable, consistent, and comfortable to the humans that will interact with it. I’ve come to understand that* ***programming is model building****. I can build software* ***models of real and imaginary worlds****. It’s all the same to a computer. Programs are built from layers of* ***abstractions*** *that are ultimately* ***rooted in the numeric manipulation capabilities of computers****.”*

Rebecca’s Personal Umwelt Summary

*“The foundation of my software umwelt is **curiosity**. It drives me to explore new things and **immerse myself in new environments**. I can be uncomfortable and uncertain of my abilities, while at the same time believing in my ability to successfully navigate new environments. Becoming **comfortable with being uncomfortable** leads me to find ways to adapt, grow, and stretch my ability to perceive and act. I never hesitated to research design options or ask other developers questions. **My ego isn’t tied to figuring things out myself.** I’m good at pinpointing key aspects in new environments, processes, or practices and explaining them to myself and others. My **outsider’s perspective**, along with my design values and practices offer unique and valuable insights. I can zoom out to establish the bigger picture of the overall system architecture, or dive into details and tweak the software design, processes, techniques, or practices. And then as a **communicator**, I write papers, essays, and books.”*

Narrative Threads & Prompts

- Origins & nurturing — first exposure to programming

  What is your earliest memory of writing code? What type of device did you use? What activities did you do? Who did you do it with? How did you learn more? What did you learn about yourself in the process?

- Going to work — first practical experiences

  When did you first feel competent/confident in your programming skills? What things did the "real world" teach you that you hadn't studied? What was the first software project you worked on that made you feel proud, and how did it unfold?

- Finding proficiency — through practice

  Describe how your relationship to software changed as your expertise grew. Did you become a generalist or specialist? How did your software development values and practices change?

- Pivots — major direction changes

  Describe a significant life or career pivot. Why did you make it? How did this change your views on software?

Narrative Threads & Prompts

- Origins & nurturing — first exposure to programming

  What is your earliest memory of writing code? What type of device did you use? What activities did you do? Who did you do it with? How did you learn more? What did you learn about yourself in the process?

- Going to work — first practical experiences

  When did you first feel competent/confident in your programming skills? What things did the "real world" teach you that you hadn't studied? What was the first software project you worked on that made you feel proud, and how did it unfold?

- Finding proficiency — through practice

  Describe how your relationship to software changed as your expertise grew. Did you become a generalist or specialist? How did your software development values and practices change?

- Pivots — major direction changes

  Describe a significant life or career pivot. Why did you make it? How did this change your views on software?

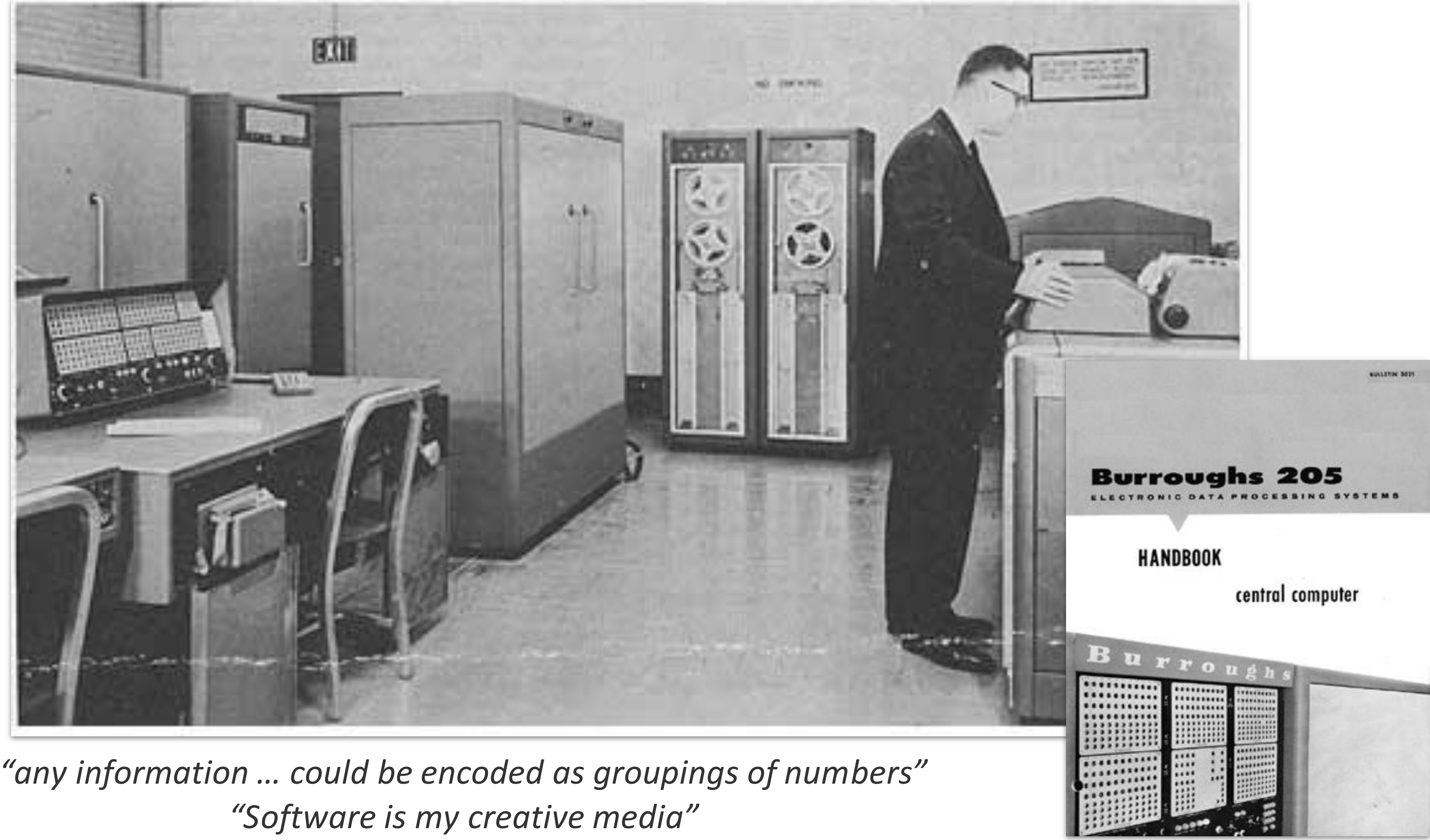
Origins & Nurturing Thread

- Origins — first exposure to programming

  What is your earliest memory of writing code? What type of device did you use? What activities did you do? Who else was involved in this memory?

- Nurturing — learning about software development

  How did you learn more about programming? What did you learn about yourself in the process? What were the “sticky” things you learned and why?



BULLETIN 3021

Burroughs 205

ELECTRONIC DATA PROCESSING SYSTEMS

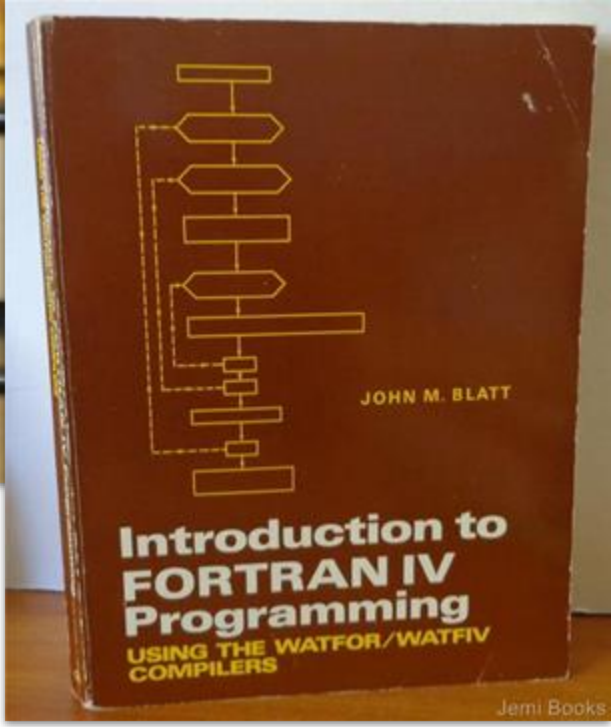
HANDBOOK

central computer

“any information … could be encoded as groupings of numbers”

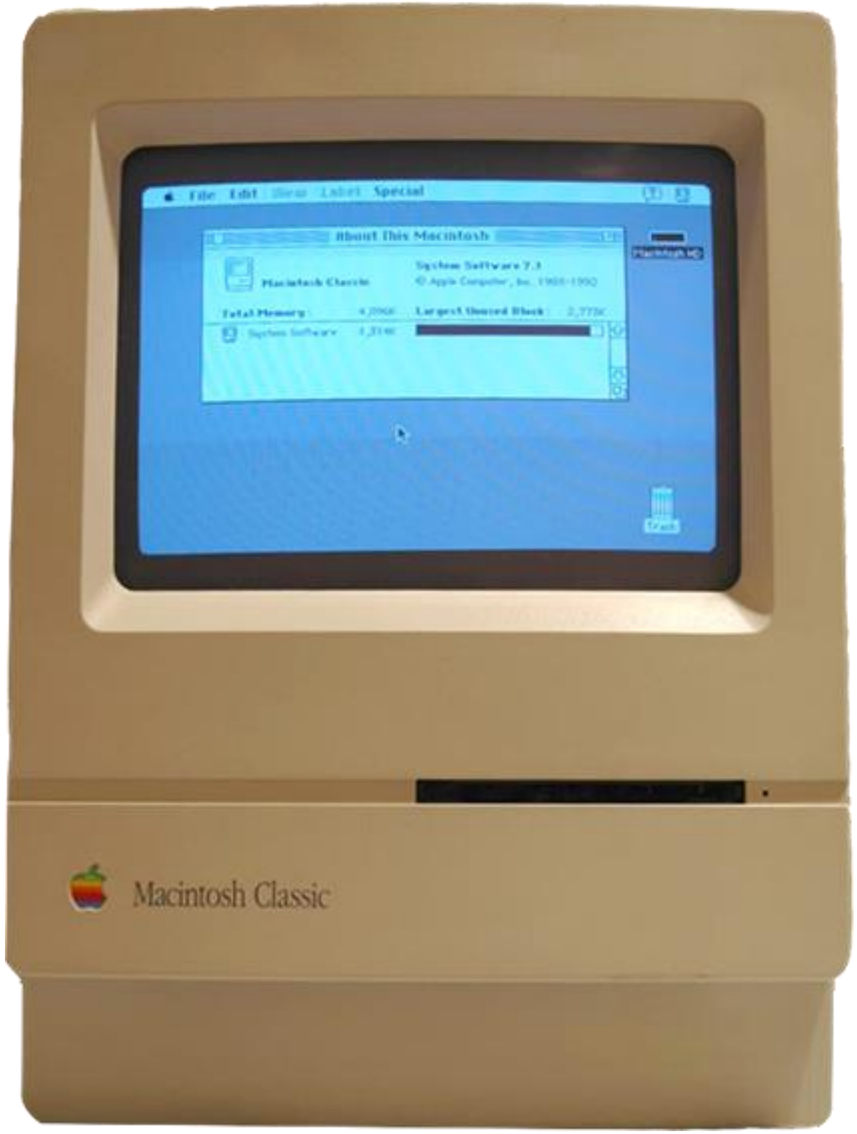
“Software is my creative media”

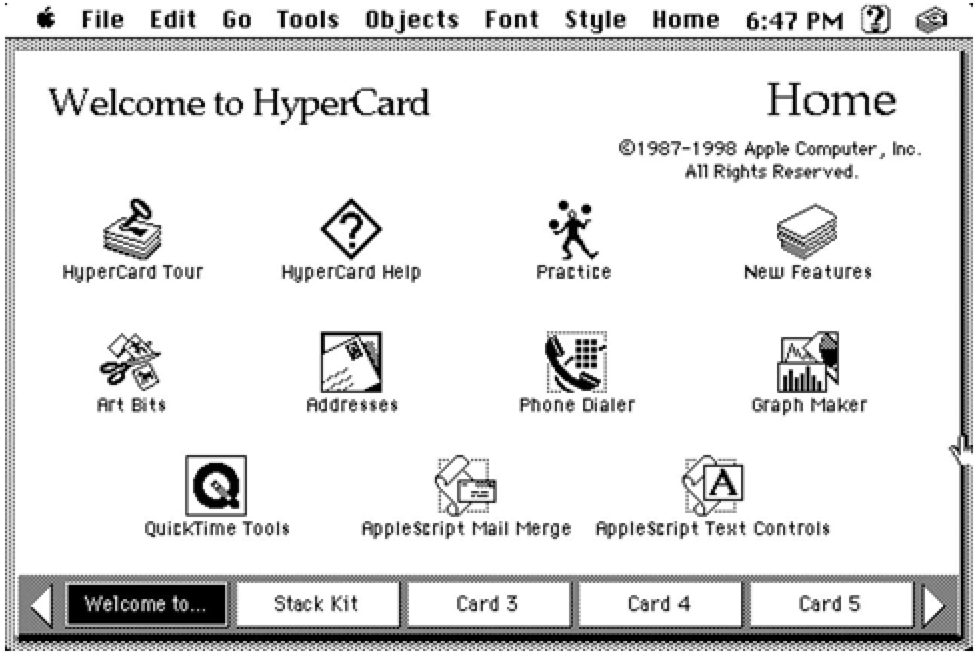
IBM 360/50


JOHN M. BLATT
Introduction to FORTRAN IV Programming
USING THE WATFOR/WATFIV COMPILERS
Jemi Books

“The foundation of my software umwelt is curiosity.”

“I can be uncomfortable and uncertain… at the same time believing in my ability to … navigate new environments.”





Welcome to HyperCard

Home

©1987-1998 Apple Computer, Inc.
All Rights Reserved.

“My software umwelt is tightly coupled with my familial umwelt”
“a type of thinking that we don’t often encounter in other dimensions of our lives.”

Our origins contributed to our umwelts

Allen

Software is my creative media …

Programs are … ultimately rooted in the numeric manipulation capabilities of computers.

Rebecca

The foundation of my software umwelt is curiosity.

I can be uncomfortable and uncertain… at the same time believing in my ability to successfully navigate new environments.

Jordan

My software umwelt is tightly coupled with my familial umwelt…

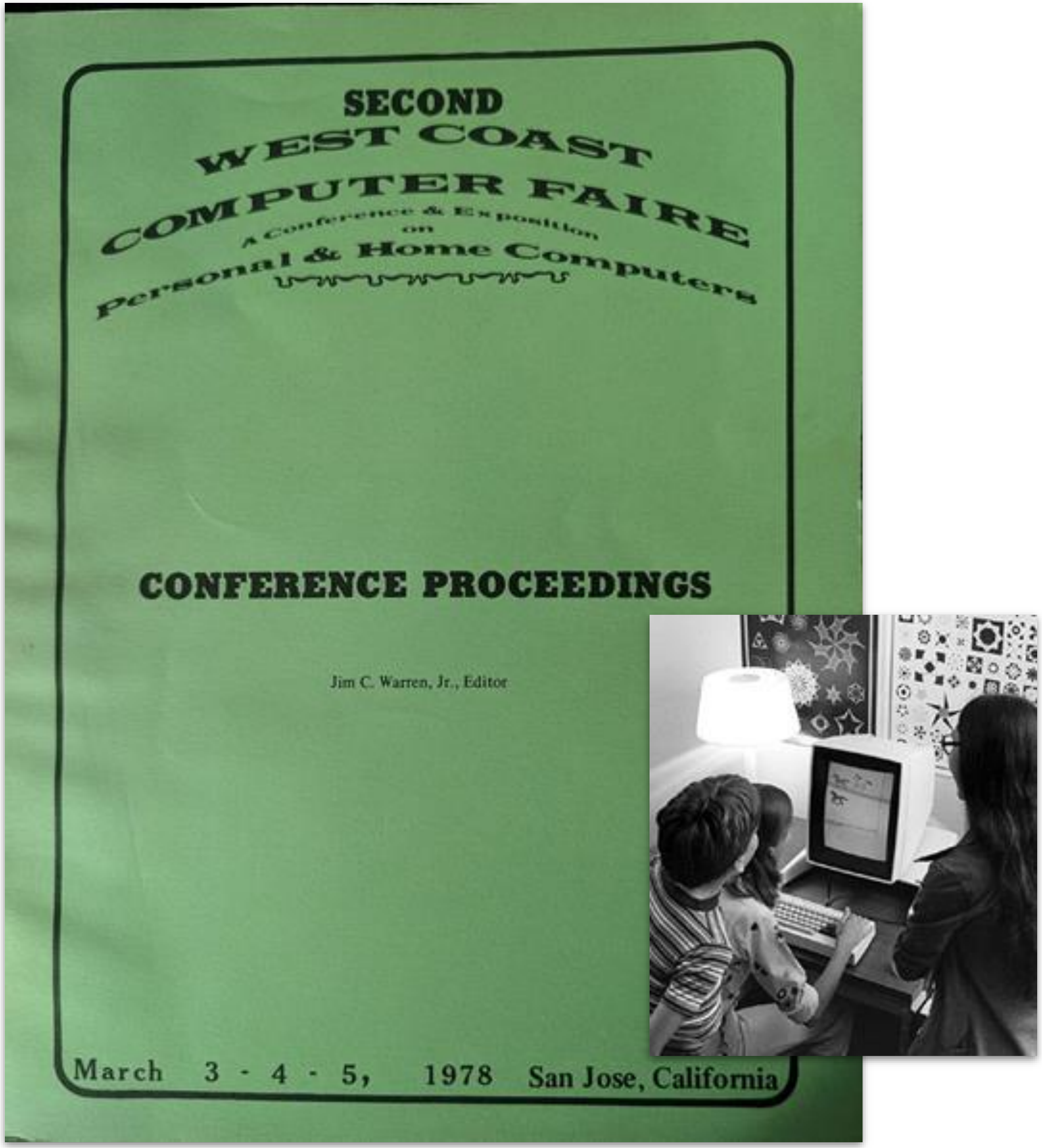
Writing software requires a type of thinking that we don’t often encounter in other dimensions of our lives.

Pivots Thread

For each of us, distinct events significantly impacted the trajectory of our software careers and umwelts.

- Pivots — major direction changes

  Describe a significant life or career pivot. Why did you make it? How did this change your views on software?

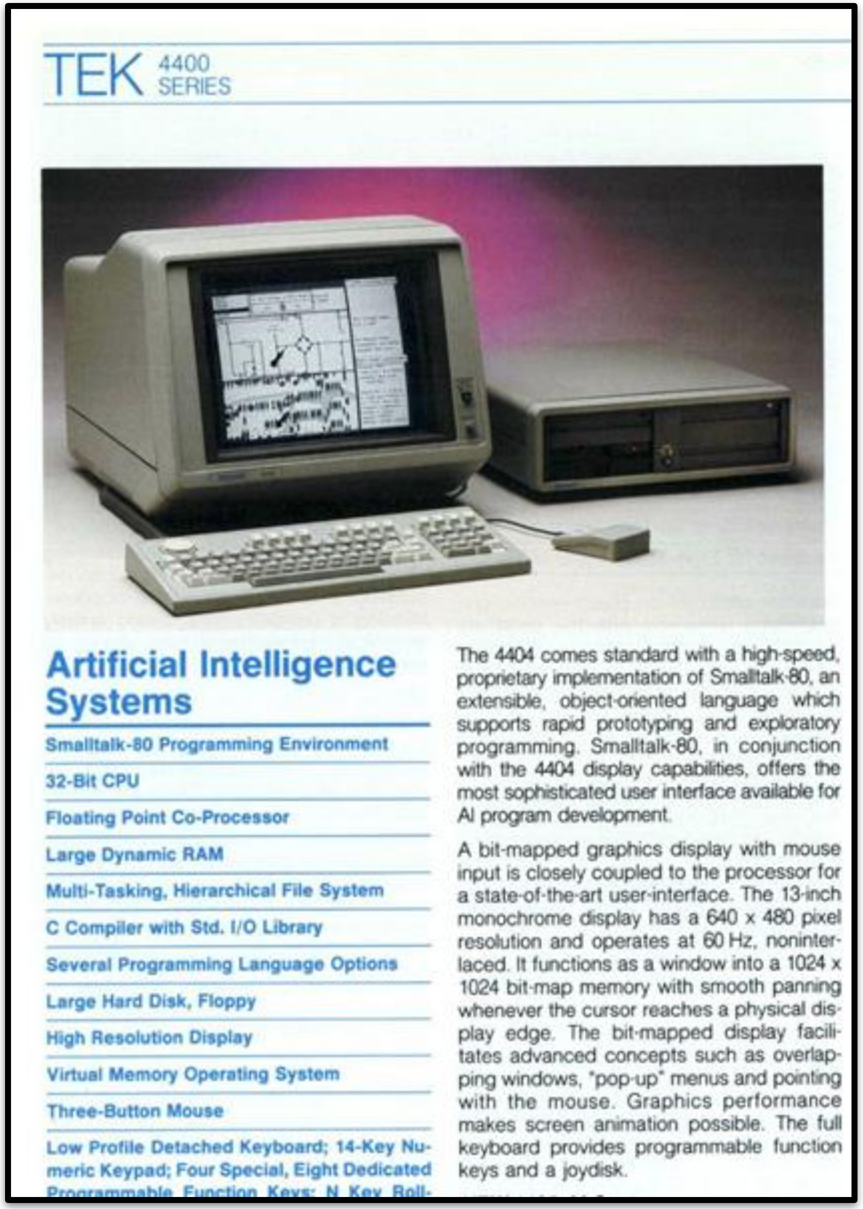


SECOND
WEST COAST
COMPUTER FAIRE
A Conference & Exposition
on
Personal & Home Computers

CONFERENCE PROCEEDINGS

Jim C. Warren, Jr., Editor

March 3 - 4 - 5, 1978 San Jose, California



TEK 4400 SERIES

Artificial Intelligence Systems

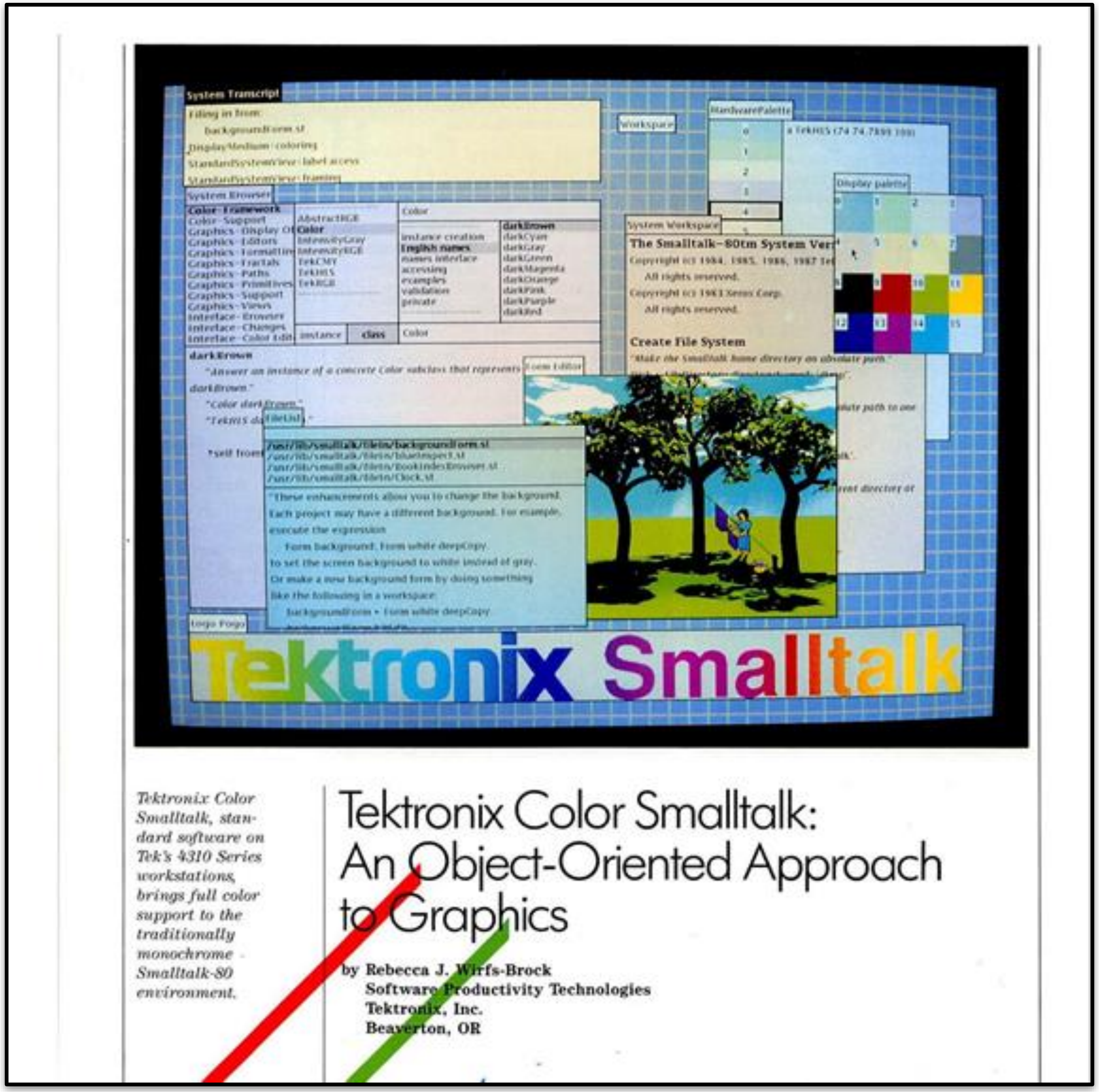
- Smalltalk-80 Programming Environment
- 32-Bit CPU
- Floating Point Co-Processor
- Large Dynamic RAM
- Multi-Tasking, Hierarchical File System
- C Compiler with Std. I/O Library
- Several Programming Language Options
- Large Hard Disk, Floppy
- High Resolution Display
- Virtual Memory Operating System
- Three-Button Mouse
- Low Profile Detached Keyboard; 14-Key Numeric Keypad; Four Special, Eight Dedicated Programmable Function Keys; N Key Roll-

The 4404 comes standard with a high-speed, proprietary implementation of Smalltalk-80, an extensible, object-oriented language which supports rapid prototyping and exploratory programming. Smalltalk-80, in conjunction with the 4404 display capabilities, offers the most sophisticated user interface available for AI program development.

A bit-mapped graphics display with mouse input is closely coupled to the processor for a state-of-the-art user-interface. The 13-inch monochrome display has a 640 x 480 pixel resolution and operates at 60 Hz, noninterlaced. It functions as a window into a 1024 x 1024 bit-map memory with smooth panning whenever the cursor reaches a physical display edge. The bit-mapped display facilitates advanced concepts such as overlapping windows, "pop-up" menus and pointing with the mouse. Graphics performance makes screen animation possible. The full keyboard provides programmable function keys and a joydisk.

"I relish radical innovations."

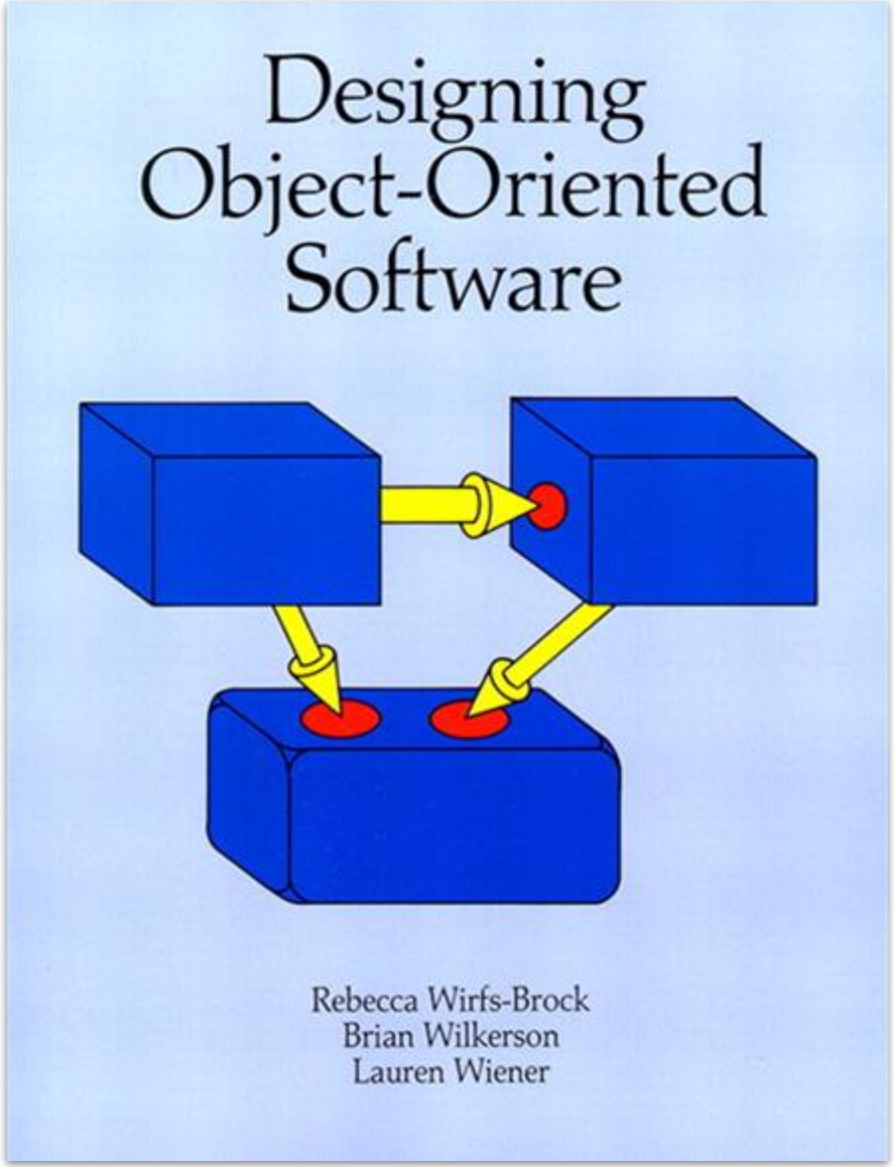
"[Software] should be approachable, understandable, consistent, and comfortable to the humans that will interact with it."


Tektronix Smalltalk

Tektronix Color Smalltalk, standard software on Tek's 4310 Series workstations, brings full color support to the traditionally monochrome Smalltalk-80 environment.

Tektronix Color Smalltalk: An Object-Oriented Approach to Graphics

by Rebecca J. Wirfs-Brock
Software Productivity Technologies
Tektronix, Inc.
Beaverton, OR

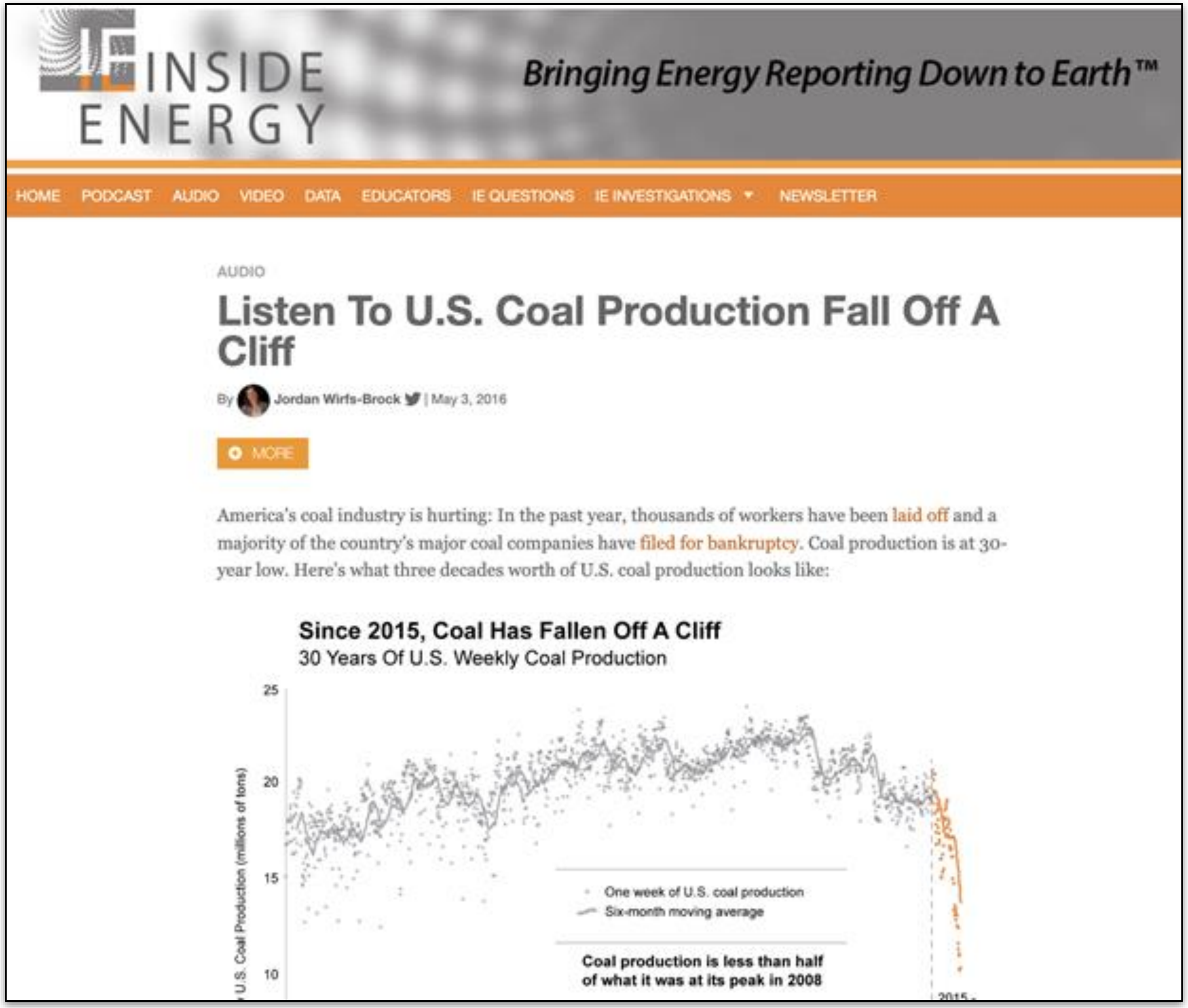


Designing Object-Oriented Software

Rebecca Wirfs-Brock
Brian Wilkerson
Lauren Wiener

"I can zoom out to establish the bigger picture of the overall system architecture, or dive into details and tweak the software design, processes, techniques, or practices."

"And then as a communicator, I write papers, essays, and books."



Listen To U.S. Coal Production Fall Off A Cliff

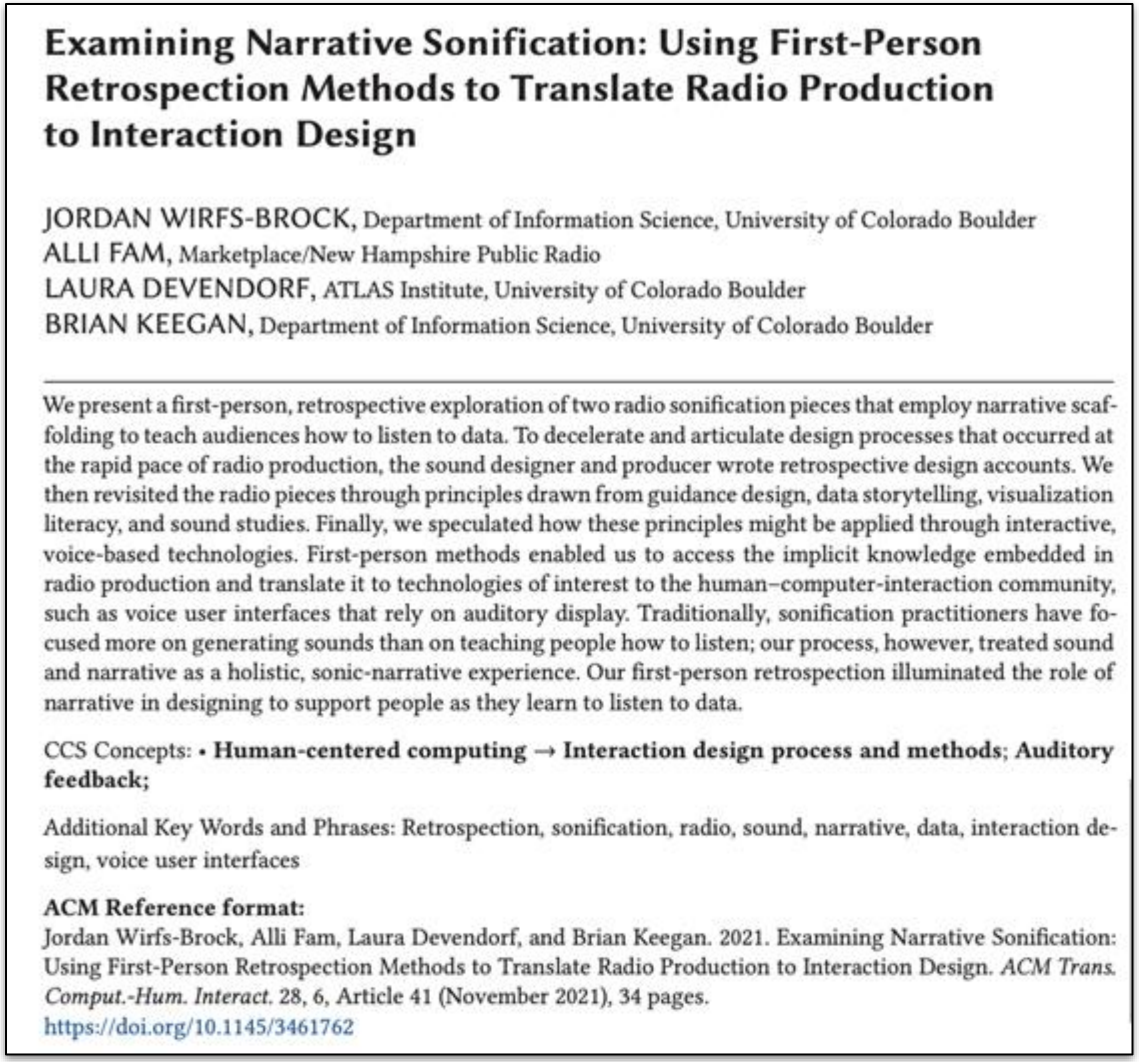
By Jordan Wirfs-Brock | May 3, 2016

America's coal industry is hurting: In the past year, thousands of workers have been laid off and a majority of the country's major coal companies have filed for bankruptcy. Coal production is at 30-year low. Here's what three decades worth of U.S. coal production looks like:

Since 2015, Coal Has Fallen Off A Cliff

30 Years Of U.S. Weekly Coal Production

25
20
15
10
U.S. Coal Production (millions of tons)
One week of U.S. coal production
Six-month moving average
Coal production is less than half of what it was at its peak in 2008
2015 -



Examining Narrative Sonification: Using First-Person Retrospection Methods to Translate Radio Production to Interaction Design

JORDAN WIRFS-BROCK, Department of Information Science, University of Colorado Boulder
ALLI FAM, Marketplace/New Hampshire Public Radio
LAURA DEVENDORF, ATLAS Institute, University of Colorado Boulder
BRIAN KEEGAN, Department of Information Science, University of Colorado Boulder

We present a first-person, retrospective exploration of two radio sonification pieces that employ narrative scaffolding to teach audiences how to listen to data. To decelerate and articulate design processes that occurred at the rapid pace of radio production, the sound designer and producer wrote retrospective design accounts. We then revisited the radio pieces through principles drawn from guidance design, data storytelling, visualization literacy, and sound studies. Finally, we speculated how these principles might be applied through interactive, voice-based technologies. First-person methods enabled us to access the implicit knowledge embedded in radio production and translate it to technologies of interest to the human–computer-interaction community, such as voice user interfaces that rely on auditory display. Traditionally, sonification practitioners have focused more on generating sounds than on teaching people how to listen; our process, however, treated sound and narrative as a holistic, sonic-narrative experience. Our first-person retrospection illuminated the role of narrative in designing to support people as they learn to listen to data.

CCS Concepts: • **Human-centered computing → Interaction design process and methods**; **Auditory feedback**;

Additional Key Words and Phrases: Retrospection, sonification, radio, sound, narrative, data, interaction design, voice user interfaces

ACM Reference format:
Jordan Wirfs-Brock, Alli Fam, Laura Devendorf, and Brian Keegan. 2021. Examining Narrative Sonification: Using First-Person Retrospection Methods to Translate Radio Production to Interaction Design. *ACM Trans. Comput.-Hum. Interact.* 28, 6, Article 41 (November 2021), 34 pages.
https://doi.org/10.1145/3461762

“Software development is a sociotechnical system. I’m always looking at systems, from a standpoint of trying to figure out what people and tools are required to navigate them.”

Our pivots contributed to our umwelts

Allen

I relish radical innovations.

[Software] should be approachable, understandable, consistent, and comfortable to the humans that will interact with it.

Rebecca

…we can't simply flush out our existing umwelt and start over.

… Becoming comfortable with being uncomfortable leads us to find ways to adapt, grow, and stretch our ability to perceive and act.

Jordan

…looking at systems, trying to figure out what people and tools are required

How do we know that we are making something people want? That they need? That doesn't harm them?

Now you try...

Pick **one** of the threads:

- ❑ **Origins & Nurturing**

  *What is your earliest memory of writing code? What details do you remember (what, who, where, why)? **How was writing code different from other things you did?** Once you started programming, how did you learn? **Why did you continue?***

- ❑ **Pivots**

  *Describe a significant life or career pivot. Why did you make it? What were the consequences of it? **How did this change your approach to software development?***

THINK / PAIR / SHARE

- 5 minutes: Think and write;
- 3 minutes: Talk with the person next to you, or paste your notes into Discord chat

Join Us in Discovering Your Software Umwelt

Tips:

- Use the prompts as starting points & modify them to fit your context.
- Start describing memorable events. Read what you’ve written and reflect.
- Do this with a buddy or two. Build a reflective community of people you trust.
- This work is never done. Step away from it, then come back.
- Alternate between modes of inquiry (writing, discussing, reading).

When you do, please share them with us—we can’t wait to perceive them.

Discovering Your Software Umwelt

Rebecca Wirfs-Brock *(Wirfs-Brock Associates)*
Allen Wirfs-Brock *(Wirfs-Brock Associates)*
Jordan Wirfs-Brock *(Whitman College)*

umwelt@wirfs-brock.com