Down With the Corporate Ethos, Up With the Sunrise

Inspiring a New Generation of Hackers

But first, some light entertainment!

[redacted]

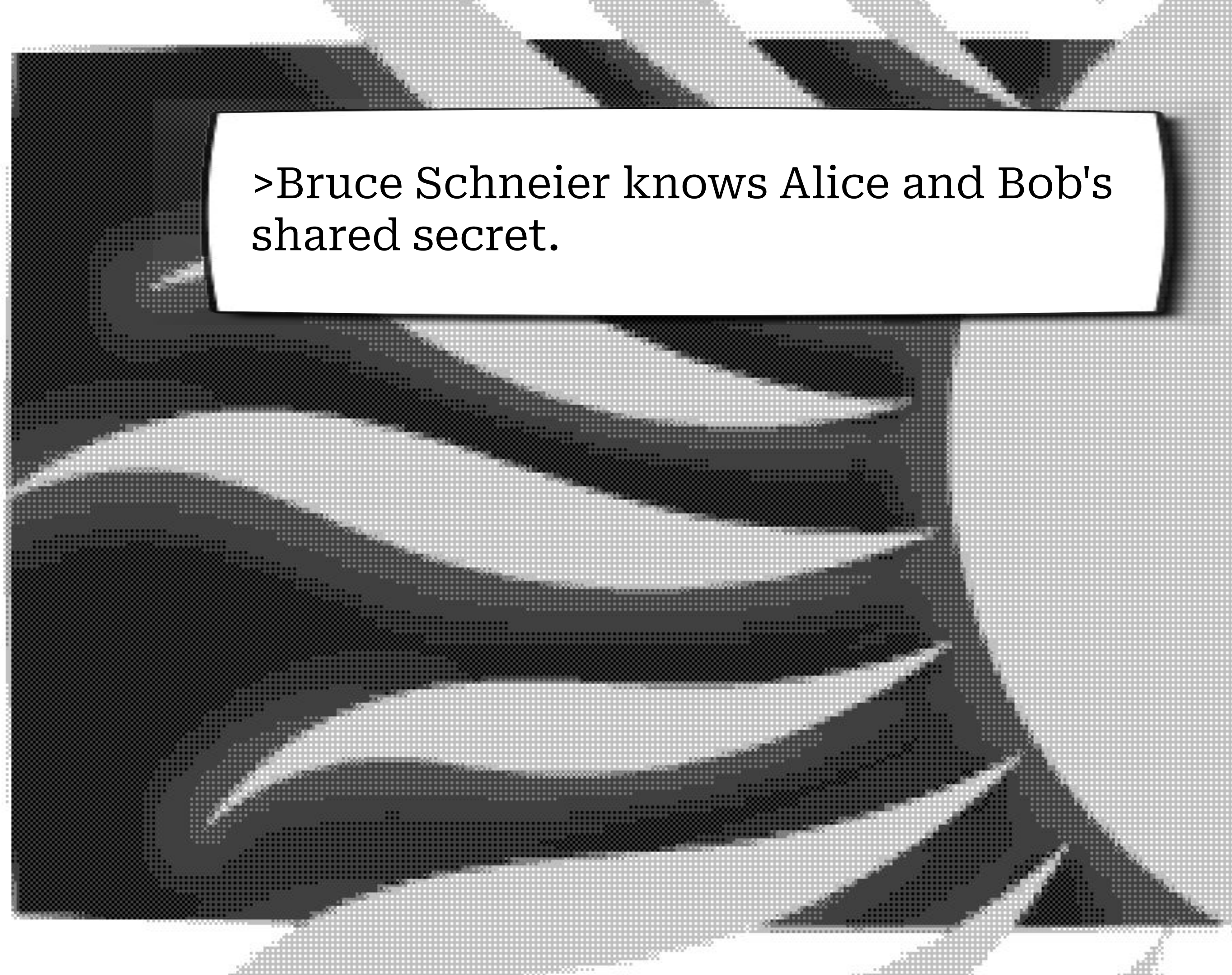
[redacted]

> The manual also contains a "tonsil", as explained in this footnote:

"4) Since all other reference manuals have appendices, it was decided that the INTERCAL manual should contain some other type of removable organ."

> PHP's parser refers to e.g. :: internally as T_PAAMAYIM_NEKUDOYATIM

> Btrfs can survive acts of god. XFS is resistant to acts of god. FAT32 is too dumb to realize what is before it is an act of god, making it similarly resistant for all the wrong reasons.



>Bruce Schneier knows Alice and Bob's shared secret.

>Bruce Schneier knows Alice and Bob's shared secret.

>Doug McIlroy can change file permissions using a magnet and a pin.

>Bruce Schneier knows Alice and Bob's shared secret.

>Doug McIlroy can change file permissions using a magnet and a pin.

>Doug McIlroy can read data from /dev/null.

>Bruce Schneier knows Alice and Bob's shared secret.

>Doug McIlroy can change file permissions using a magnet and a pin.

>Doug McIlroy can handle SIGKILL.

The Problem

(1) Students

The Problem

1. **Students**

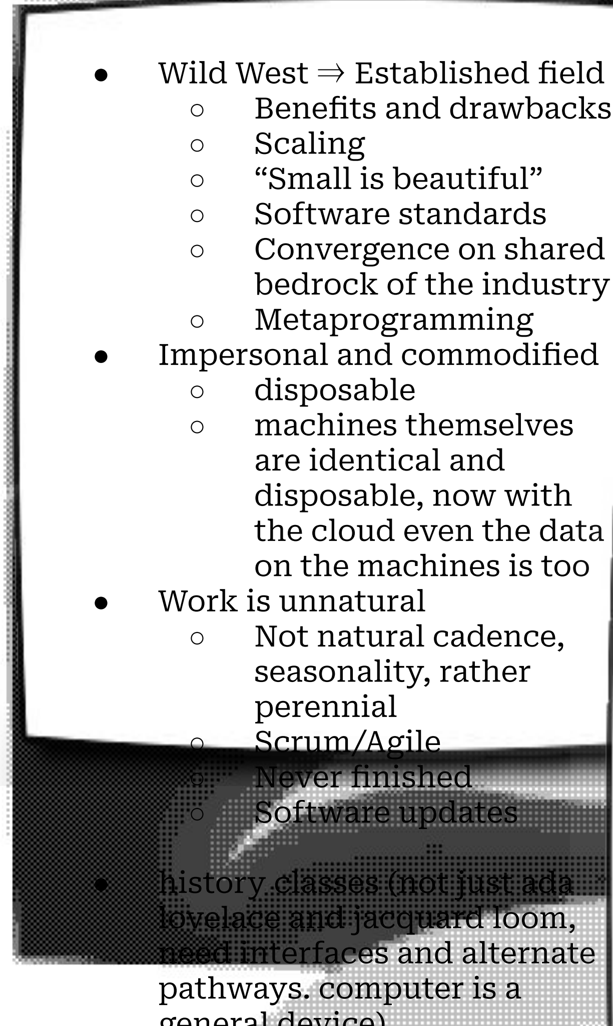
- Corporate poisoning
  - Leetcode
  - Resume
  - Dead-on-arrival portfolio projects
  - Take easiest classes for GPA
  - Standardized testing incentivizes cheating and rote memorization

- No wholesome incentives to write programs, just scare tactics
  - At times, it seems like everything has been invented already
  - Internet

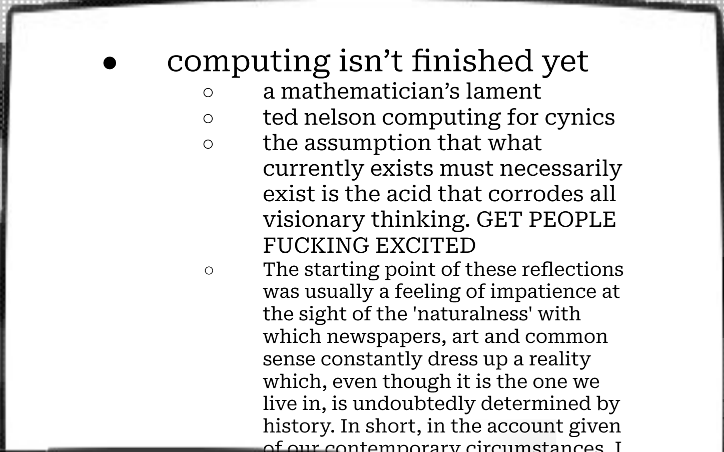
- Why should you care?
  - “The world in a raindrop”

The Problem

(2) Computing



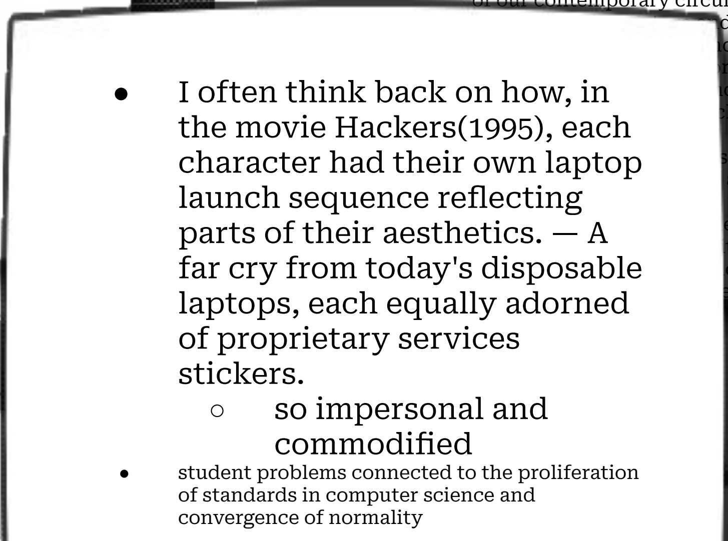
- Wild West ⇒ Established field
  - Benefits and drawbacks
  - Scaling
  - “Small is beautiful”
  - Software standards
  - Convergence on shared bedrock of the industry
  - Metaprogramming
- Impersonal and commodified
  - disposable
  - machines themselves are identical and disposable, now with the cloud even the data on the machines is too
- Work is unnatural
  - Not natural cadence, seasonality, rather perennial
  - Scrum/Agile
  - Never finished
  - Software updates
- history classes (not just ada lovelace and jacquard loom, need interfaces and alternate pathways. computer is a general device)



- computing isn’t finished yet
  - a mathematician’s lament
  - ted nelson computing for cynics
  - the assumption that what currently exists must necessarily exist is the acid that corrodes all visionary thinking. GET PEOPLE FUCKING EXCITED
  - The starting point of these reflections was usually a feeling of impatience at the sight of the 'naturalness' with which newspapers, art and common sense constantly dress up a reality which, even though it is the one we live in, is undoubtedly determined by history. In short, in the account given of our contemporary circumstances, I ... d History ... d I wanted ... rative ... out-saying, ... ch, in my ... st always ... a ‘natural ... presented ... e to be a ... t must make ... ed to be ... e

lem

ing



- I often think back on how, in the movie Hackers(1995), each character had their own laptop launch sequence reflecting parts of their aesthetics. — A far cry from today's disposable laptops, each equally adorned of proprietary services stickers.
  - so impersonal and commodified
- student problems connected to the proliferation of standards in computer science and convergence of normality

Th

- newcomers to the internet are often startled to discover themselves not so much in some soulless colony of technocrats as in a kind of cultural Brigadoon - a flowering remnant of the 60s, when hippie communalism and liberarian politics formed the roots of the modern cyberrevolution - Stewart Brand
  - now it’s exactly a colony of technocrats and their companies and adverts

“On the Internet, nobody knows you’re a dog.”

The Problem

(3) Society

- we don’t trust each other anymore, partisan breakdown
- cold rationalism
- raise children to be automatons
- I grew up in a world in which it seemed like we already knew the answers to all of the questions. Instead of being told about the sorts of ongoing problems and unknowns my generation of children could potentially grapple with throughout our lifetime, the most existential thing we were presented with was "what do you want to be when you grow up?"
- taught to exert ourselves in contexts that don't matter and dissociate when confronted with the ones that do
  - i try so hard at work for something that doesn’t matter but im scared of emotional duress or homeless people on the street
- aliens are hte last socially acceptable mythology

- attempts to re-enchant the world after the death of god
- Capitalist Realism: Is There No Alternative?" by Mark Fisher
- it's easier to imagine end of the world than the end of capitalism
- He defines capitalist realism as "the predominant conception that capitalism is the only viable economic system, and thus there can be no imaginable alternative." "Fisher likens capitalist realism to a "pervasive atmosphere" that affects areas of cultural production, political-economic activity, and general thought." I see the same attitude in the world of technology
-

Rose tinted glasses

(the past)

- Timesharing
  - community/fellowship
  - What we wanted to preserve was not just a good environment in which to do programming, but a system around which a fellowship could form. We knew from experience that the essence of communal computing [...] is not just to type programs into a terminal instead of a keypunch, but to encourage close communication. - DMR
- ARPA/DARPA money
  - open-ended research
  - venture capital
  - The industrial mindset -- short-term, results-driven, immediately-applicable -- is generally hostile to long-term, exploratory, foundational research
  - how can we get this without the backdrop of war?

bsd history (bill joy, berkeley university unix, etc)

The story I told in the talk -- "they didn't know what they were doing, so they tried everything" -- was essentially that programming at the time was in the "pre-paradigm phase", as defined by Thomas Kuhn in The Structure of Scientific Revolutions. This is the period of time before researchers reach consensus on what problems they're actually trying to solve. The establishment of distinct programming paradigms (e.g., functional, logic, etc.) led into Kuhn's "normal science" phase (or as Sussman put it, "diddling with details") where the foundations of the subject tend to be taken for granted.

The hacker ethic that played such a large part in advancing computer science, building gcc, building Linux, indeed building the world's computer systems and engineering the biggest peaceful economic boom in history, is more than just a thirst for knowledge about computers. It's the obsessive belief that knowledge exists to be shared, that helping someone by making their computer run better (or their air conditioner) is one of life's joys, and that the rules that prevent sharing and helping exist to be broken. [...] The hero is the one who knows how to fix things, and fixes them -- despite not being "authorized." The evil is the paperwork we construct around ourselves, the forms and regulations that take the place of people freely helping each other.

- in title i used the word hacker, i’m evoking the traditional meaning originating from the mit AI lab, e.g. see “hackers heroes of the computer revolution by steven levy”
- we modern students and programmers can take inspiration from this ethos

aaaaa

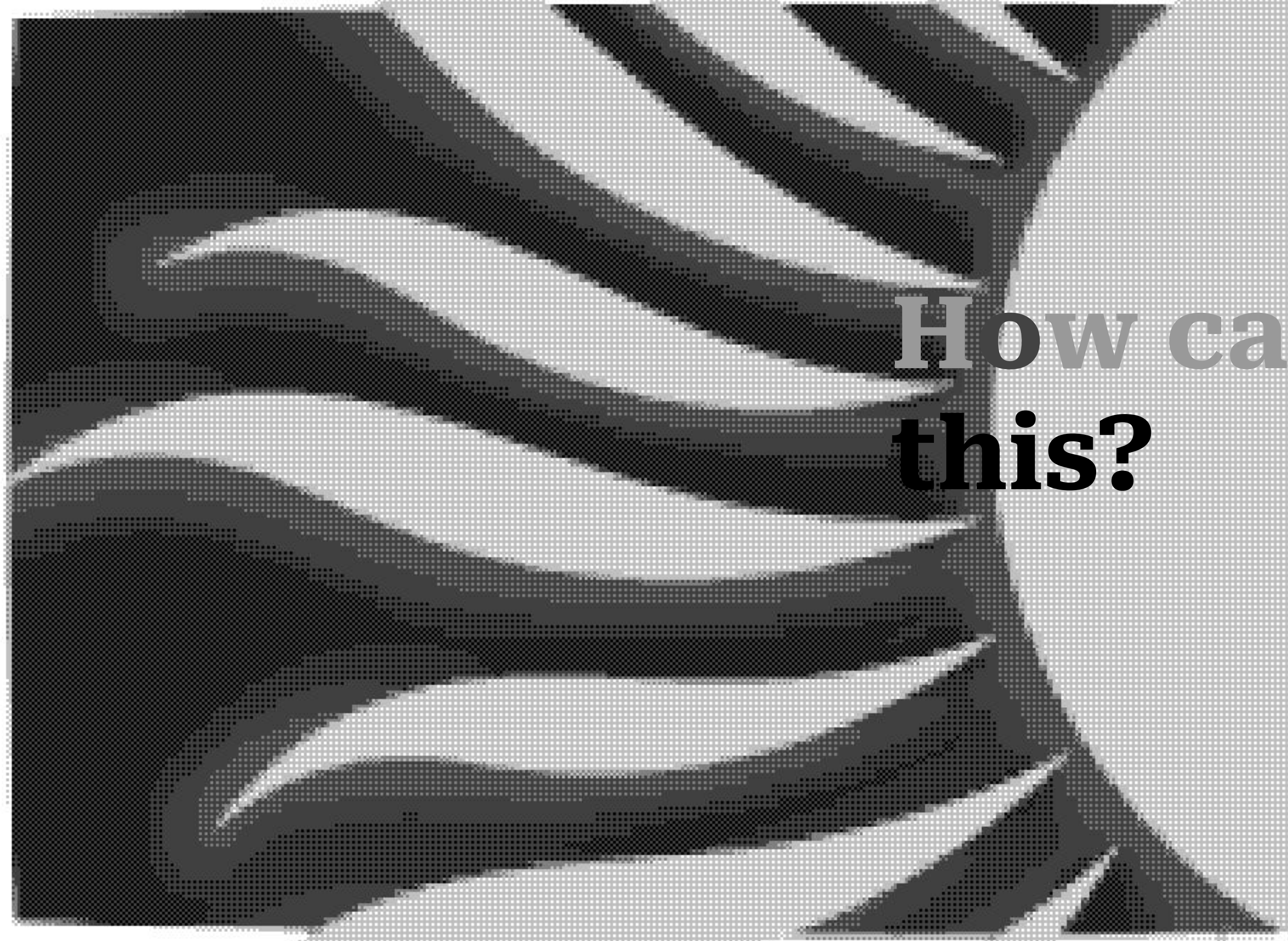
Rose tinted glasses

(the past)

Important: I learned a lesson about this. Don’t be bitter about the present. Yearn for the past but with the most optimistic eye towards the future.

Religions are fun and you learn a lot but don’t get stuck.

I sent an email to somebody to the effect of “help i was born in the wrong generation where is my Xerox Parc Bell Labs UNIX LISP machine inc?” expecting affirmation but he saw through my illness and gave me some rational advice.



How can we fix this?

Romanticism

Ro

- poetry
- communicate the full richness
- If you want to build a ship, don't drum up the men to gather wood, divide the work and give orders. Instead, teach them to yearn for the vast and endless sea.
- a rosy future
- myth spirituality
- fascism
    - home steading website farm

there are some things that can only be evoked or danced around
paintings
film
cappuccino at night
sunrise in the morning

"People can't get a joke by you explaining it to them, they have to experience getting it for themselves."
I lot of experience is just like this.

Long ago, I observed that Edsger Dijkstra’s dining philosopher’s problem received much more attention than I thought it deserved. For example, it isn’t nearly as important as the self-stabilization problem, also introduced by Dijkstra, which received almost no notice for a decade. I realized that it was the story that went with it that made the problem so popular.

the warm stream of computing
Marxism is well known for its two components: a “cold stream” which concerns objective scientific analysis; and a “warm stream” that concerns enthusiasm and hope and leads to commitment to Marxism as a cause.
more metaphors

CAFFE REGGIO ORIGINAL CAPPUCCINO
SINCE 1927
ORIGINAL CAPPUCCINO
CAFE
Caffe Reggio
Original Cappuccino Since 1927
Caffe Reggio
Original Cappuccino Since 1927
Caffe Reggio

ONE WAY

Help people help each other

- in constrains i find freedom / put students in an environmen together where they help each other

- If you wrote a program, you were contributing to your community's infrastructure, building it up with more and more utilities over time. Every program you wrote bettered the system, extending the capabilities of whomever you were sharing your system with. Systems themselves were simpler, built from primitives one could reasonably wrap their head around, so adding an impactful change was possible.

UNIX is a time sharing system, we're all logged in to one big machine and sharing its resources. (The operating system is the arbiter of this process, by the way). Since we're all logged in, let's compute together!
who/write/wall/etc
how can we make this more like a community/village?

H

- students dont think cli programs are real programs at first
  - enchant them with language and composable spells

Emulating the Past
Why can't these circumstances can't be reproduced. By simulating a small community where all students reside and interact, sharing programs, improving infrastructure and working on the community itself as they chug along, we can replace the hollow drive of modern society with a wholesome ethic of helping one another learn and be better.. We all crave this human connection that we are so deprived of.
it's like playing a multiplayer game but in real life

get students to program outside of class for a good cause, and make them feel good for doing so (all while earning new stuff). Coming together and improving the infrastructure of the school is a great way of doing so.

shared /usr/local/bin, throw stuff in there