

CSCI 1377

Tools for Thought

Artificial Intelligence II

Weird Futures

“The forced matings of minds and electrons succeed and fail with equal spectacle. Our hybrids become as brilliant as savants, and as autistic. [...] Computers bootstrap their own offspring, grow so wise and incomprehensible that their communiqués assume the hallmarks of dementia: unfocused and irrelevant to the barely-intelligent creatures left behind.”

— Peter Watts, *Blindsight* (2006)

It's 1439. Johannes Gutenberg just invented the printing press. What happens next?

- 1500 (61y): printing presses have produced ~10 million books
- 1517 (78y): printing spreads heterodoxy, facilitating the Protestant Reformation
- 1577 (138y): printing allows the sharing of astronomical observations of a comet
- 1605 (166y): printing spreads news from the world's first newspaper
- 1618 (179y): printing spreads propaganda, facilitating the Thirty Years War
- 1820 (381y): global literacy rate estimated at 12% of world's population
- 1833 (394y): the "penny press" brings ad-subsidized news to the lower classes
- 1983 (544y): global literacy rate estimated at 70% of world's population

What do you want to know?

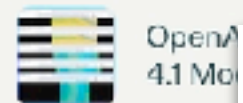
Ask anything...

PDF

Deep Research



Join the waitlist to get early access to Comet
Introducing Comet, a new browser for agentic search



META

Bitcoin



Ask Meta AI...



Canvas

BETA

Imagine



Imagine a paper airplane city

Help me write a cover letter

What can I help with?

Ask anything



Search

Reason



What can I help you

Ask v0 to build...

No project selected

Clone a Screenshot

Import from Figma

Landing Page

Sign Up Form

Calculate Factorial

Hey, what's on your mind

Hello, night owl

How can I help you today?



Claude 3.7 Sonnet



Get advice

Learn something new

Create an image

Make a plan

Brainstorm

Practice a language

Take a quiz



Hi, I'm DeepSeek.

How can I help you today?

Message DeepSeek

DeepThink (R1)

Search

Build something Lovab

Idea to app in seconds, with your personal full stack engineer

Create an app for selling and buying goods using Stripe payment processing

Attach

Job board

E-commerce product page

Social media feed

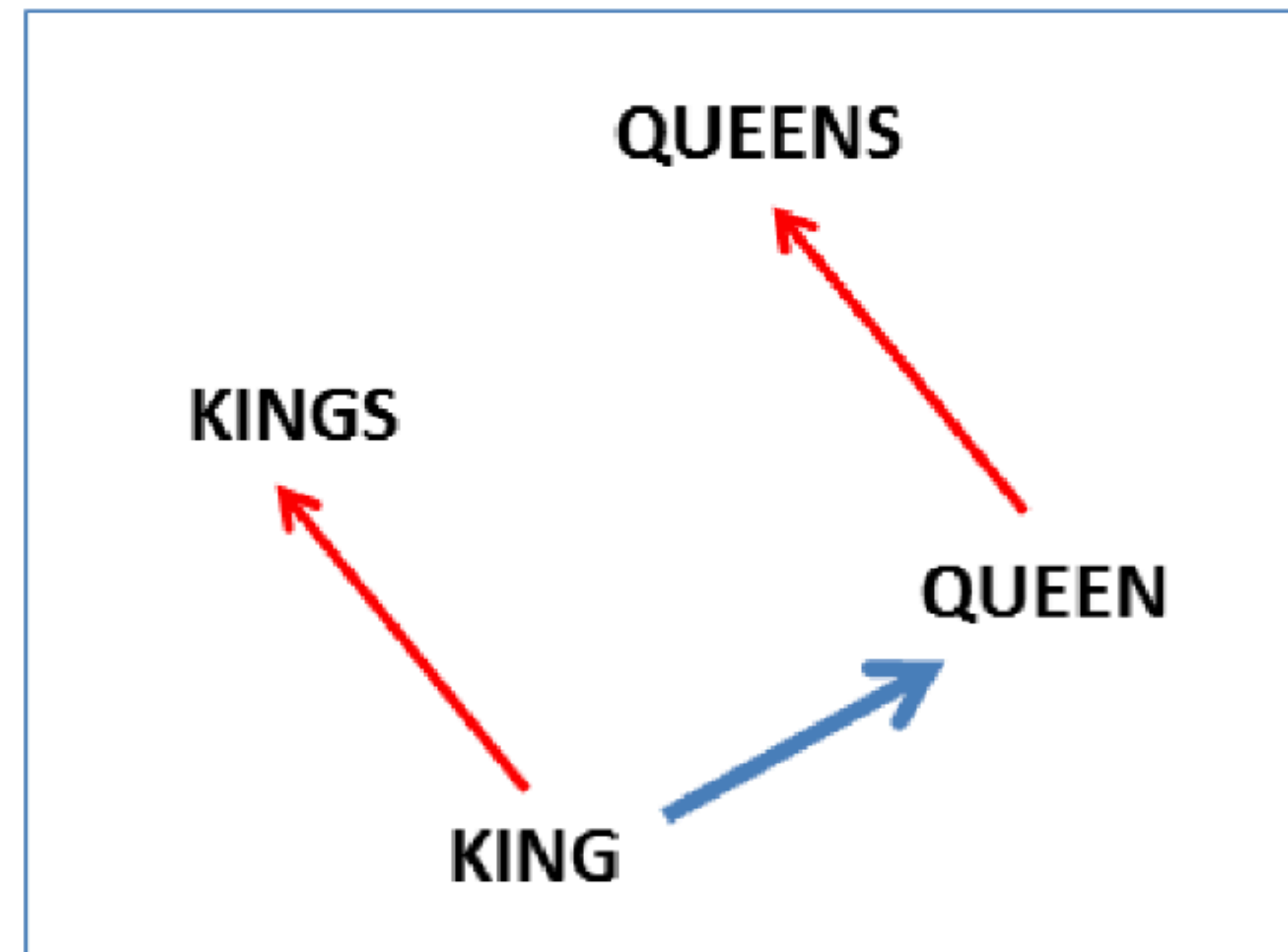
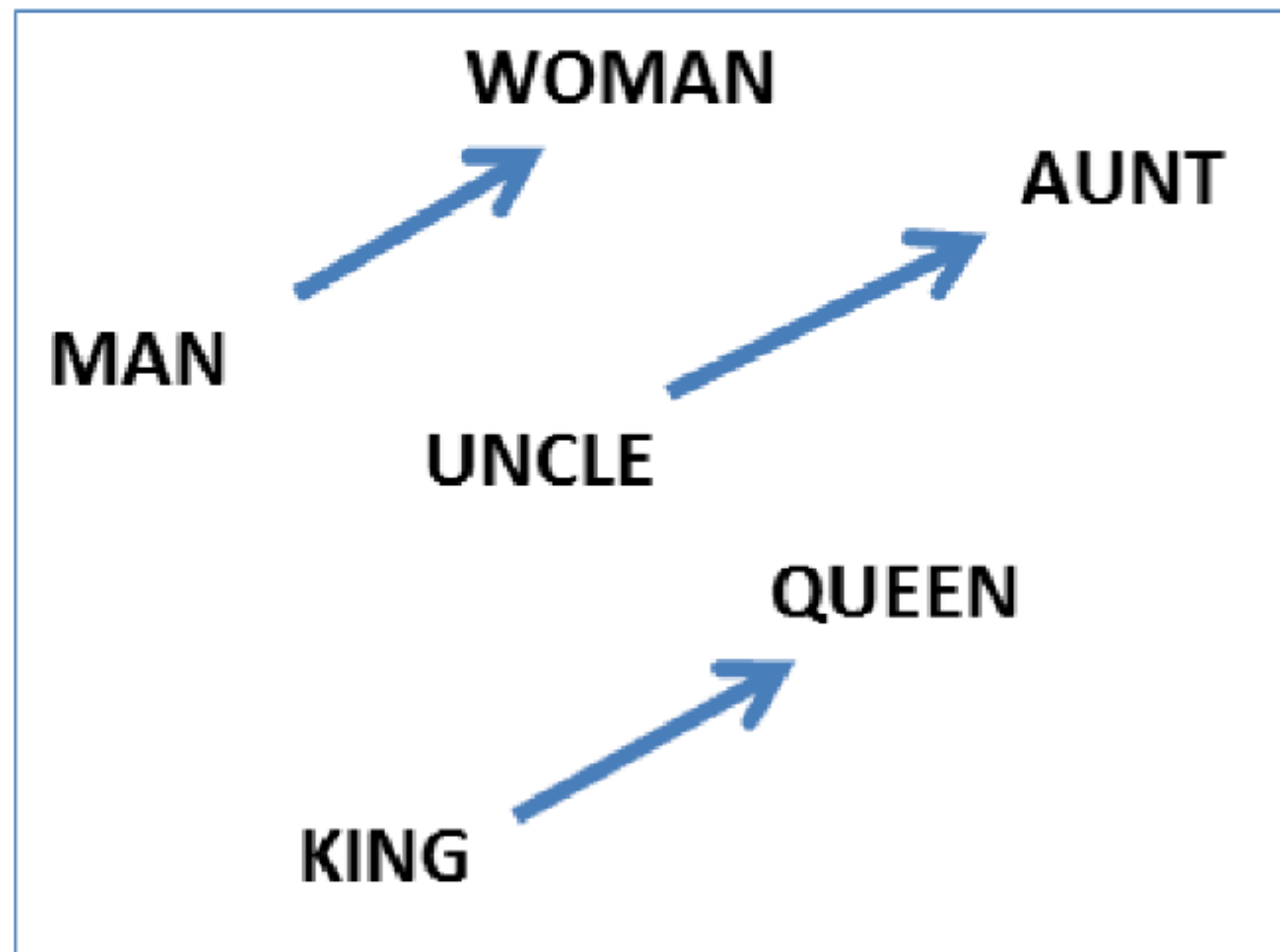
SaaS landing

Learning from superintelligence

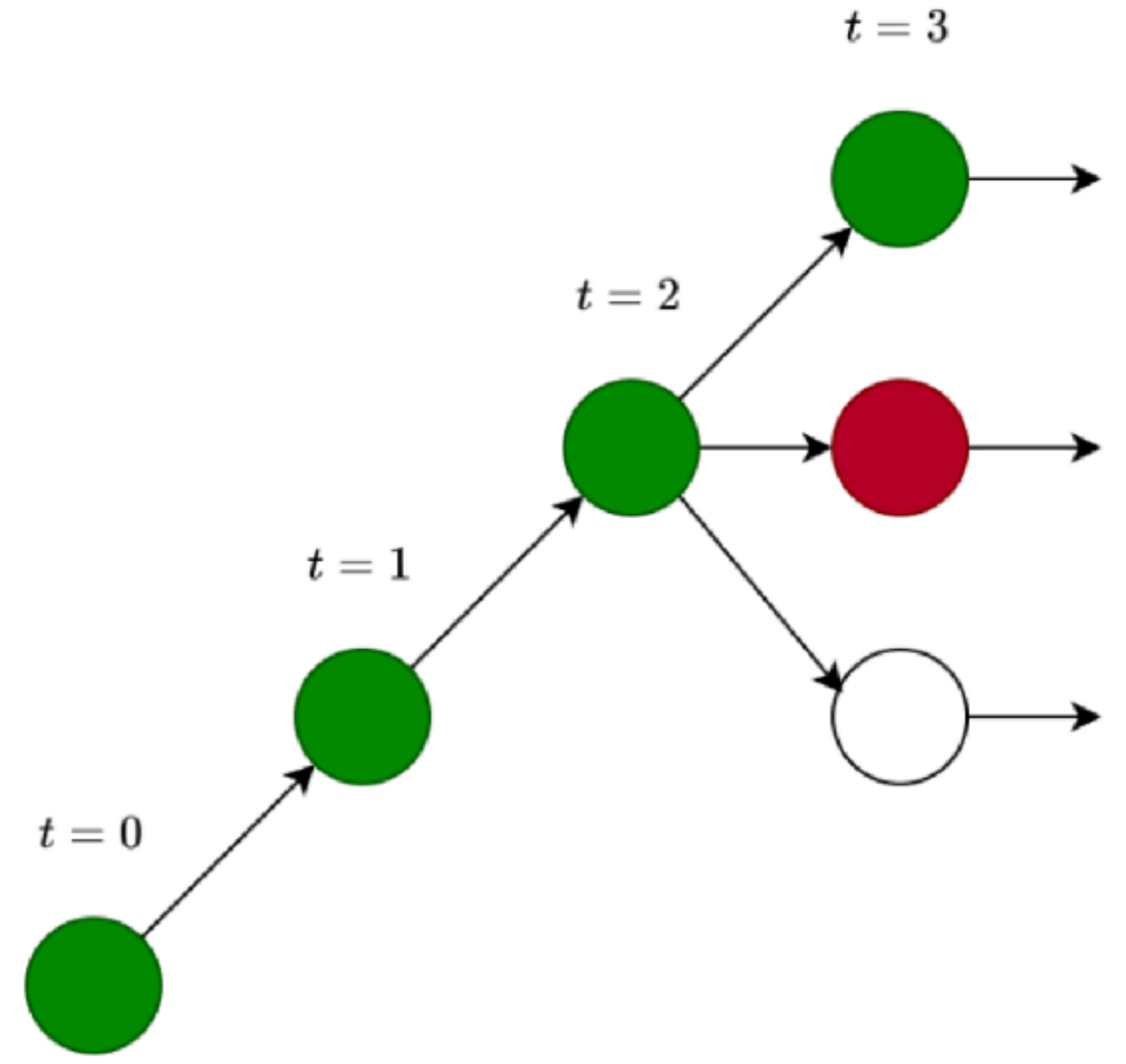
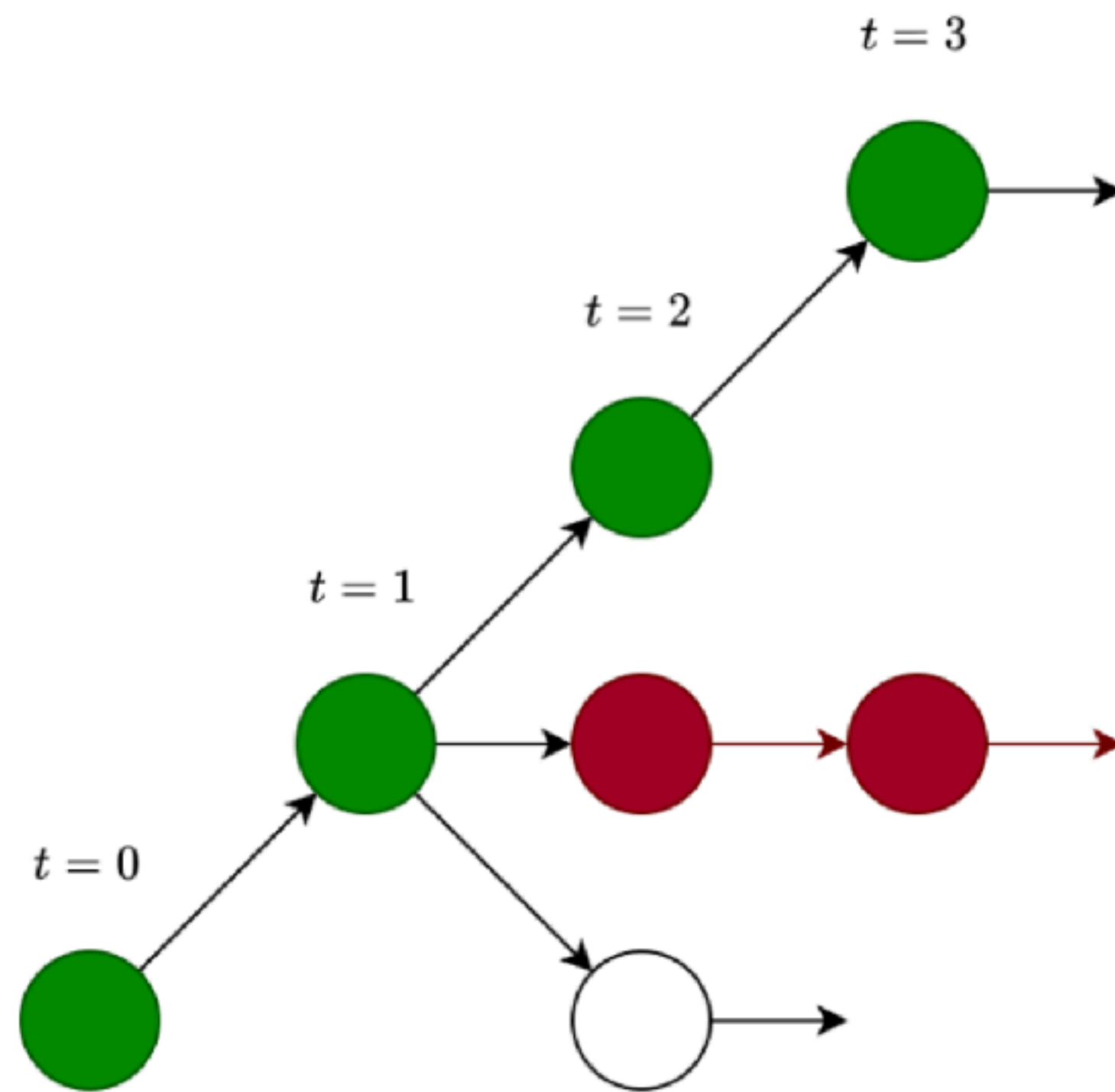
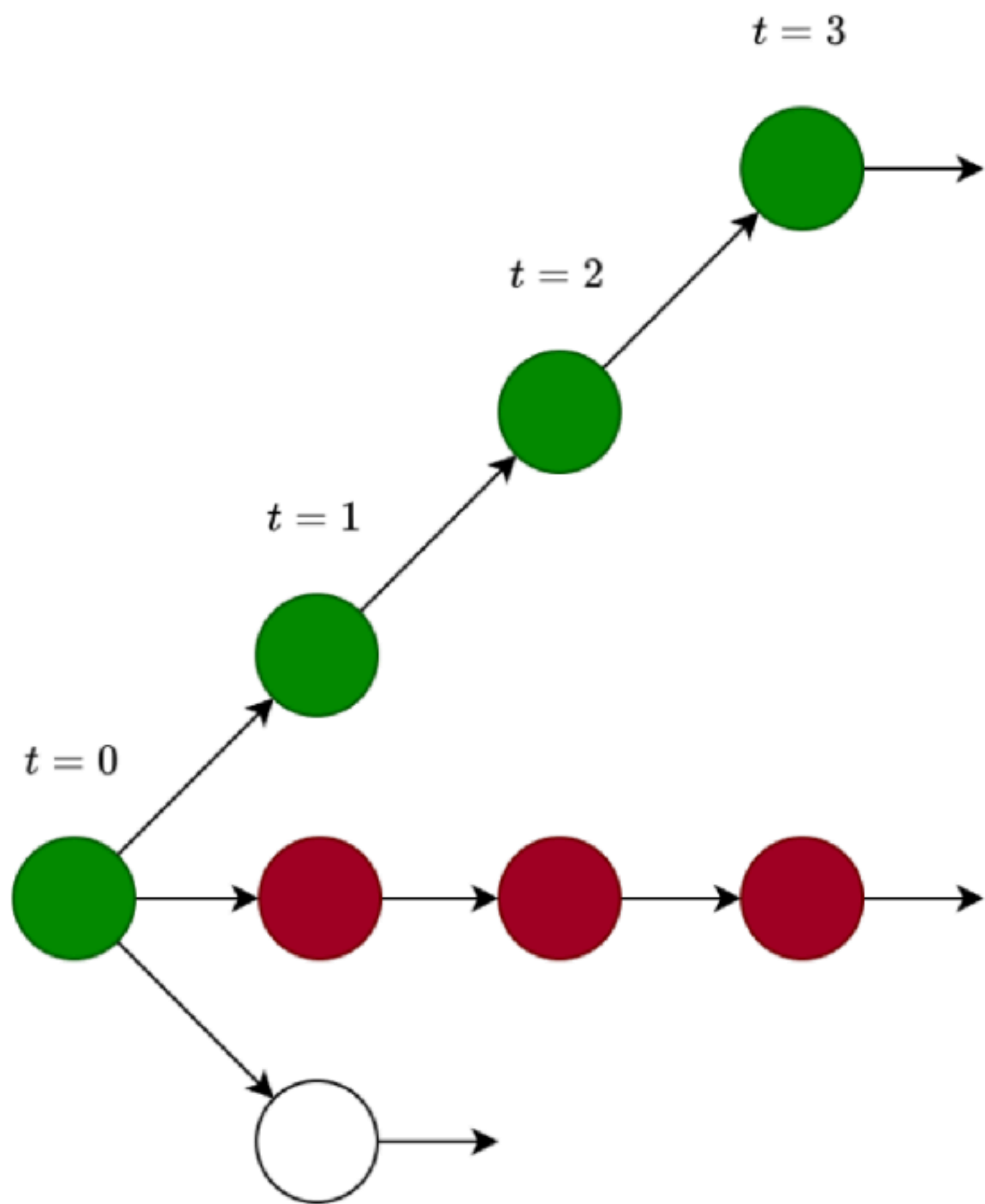
Vectors between embeddings can represent concepts

$$\text{forward}_{\ell}(\text{"King"}) = \overbrace{[0.8, 1.2, \dots, -9.3]}^{|\ell| \text{ numbers}}$$

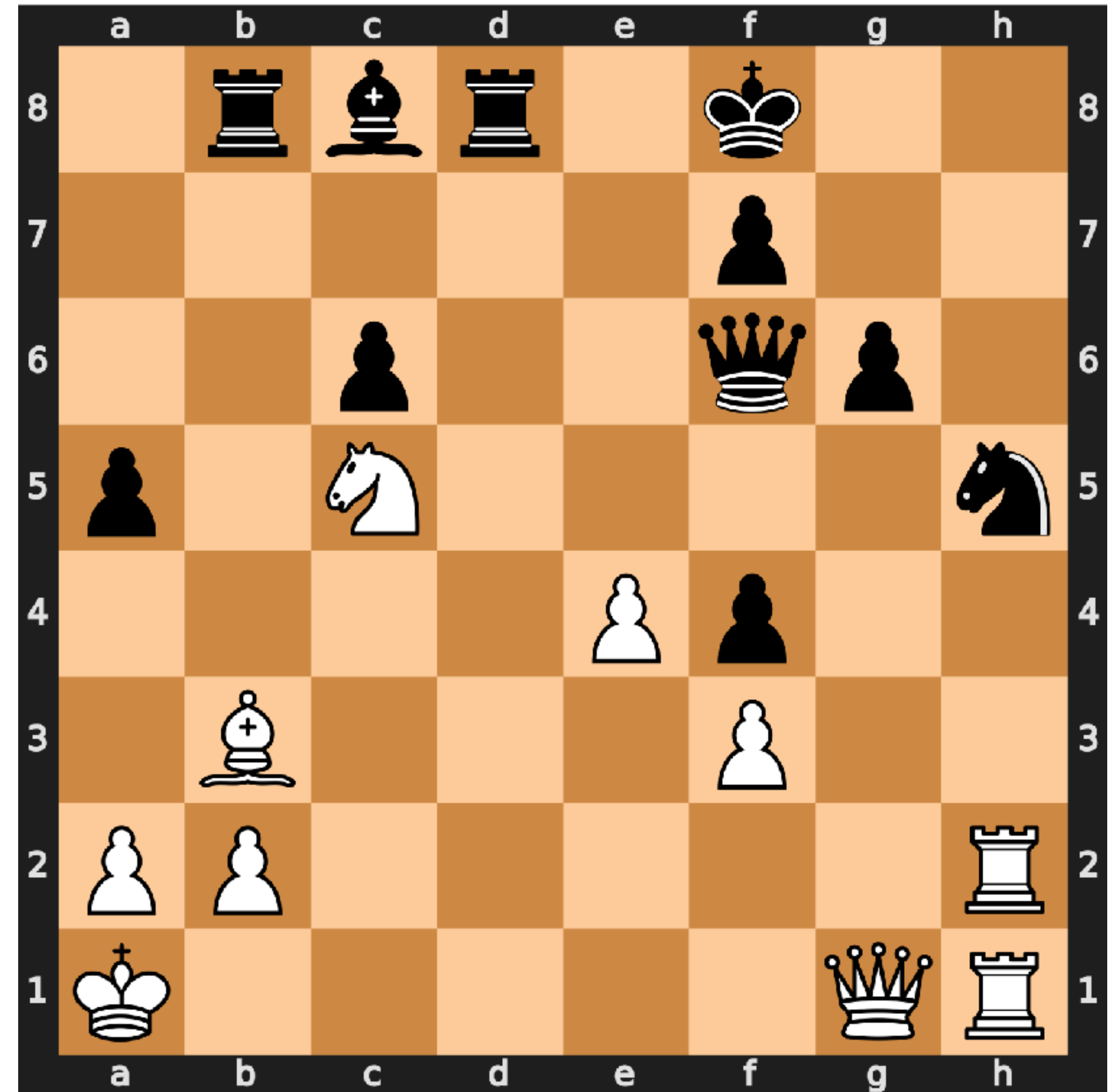
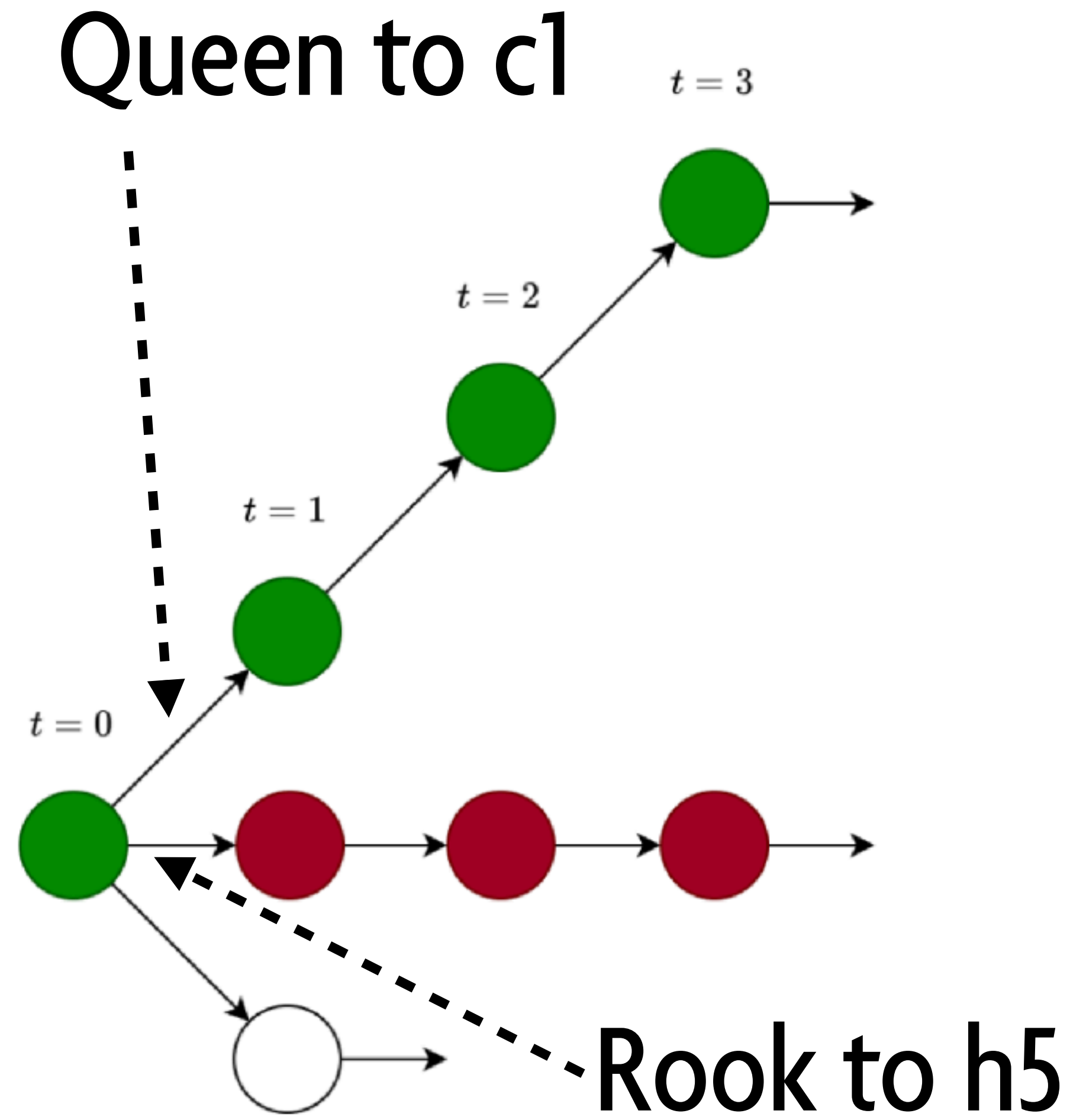
$$\text{forward}_{\ell}(\text{"King"}) - \text{forward}_{\ell}(\text{"Man"}) \approx \text{royalty}$$



AZ evaluates a sequence of board positions to decide the optimal move



Concepts should explain why AZ makes one move and not the other



Concept discovery process:

- 1. Discover which vectors are meaningful to decisions**
- 2. Determine whether the concepts can be taught**
- 3. Estimate whether the concept is shared or AI-specific**
- 4. Show boards w/ teachable & novel concepts to humans**

Concepts are vectors differentially active in preferred board positions

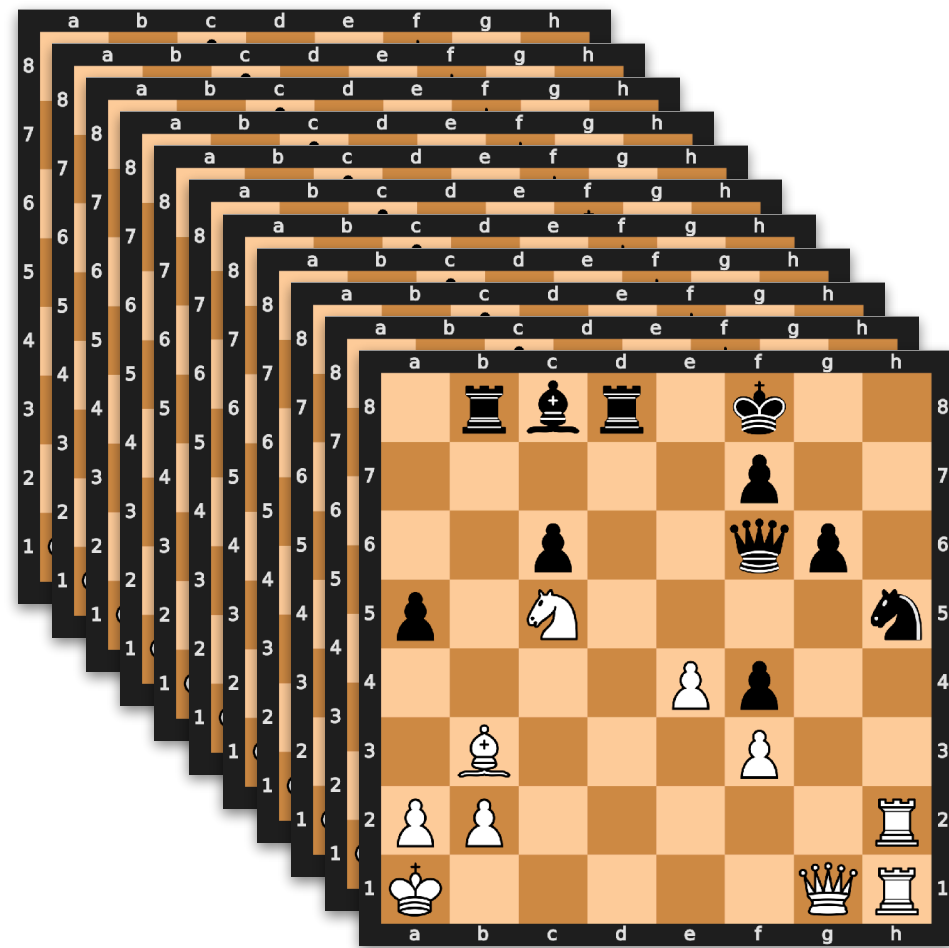
Queen to c1 = \mathbf{z}^+

Rook to h5 = \mathbf{z}^-

Goal: find the smallest vector \mathbf{v} such that

$$\mathbf{v} \cdot \mathbf{z}^+ \geq \mathbf{v} \cdot \mathbf{z}^-$$

(and for all steps to a max depth T)

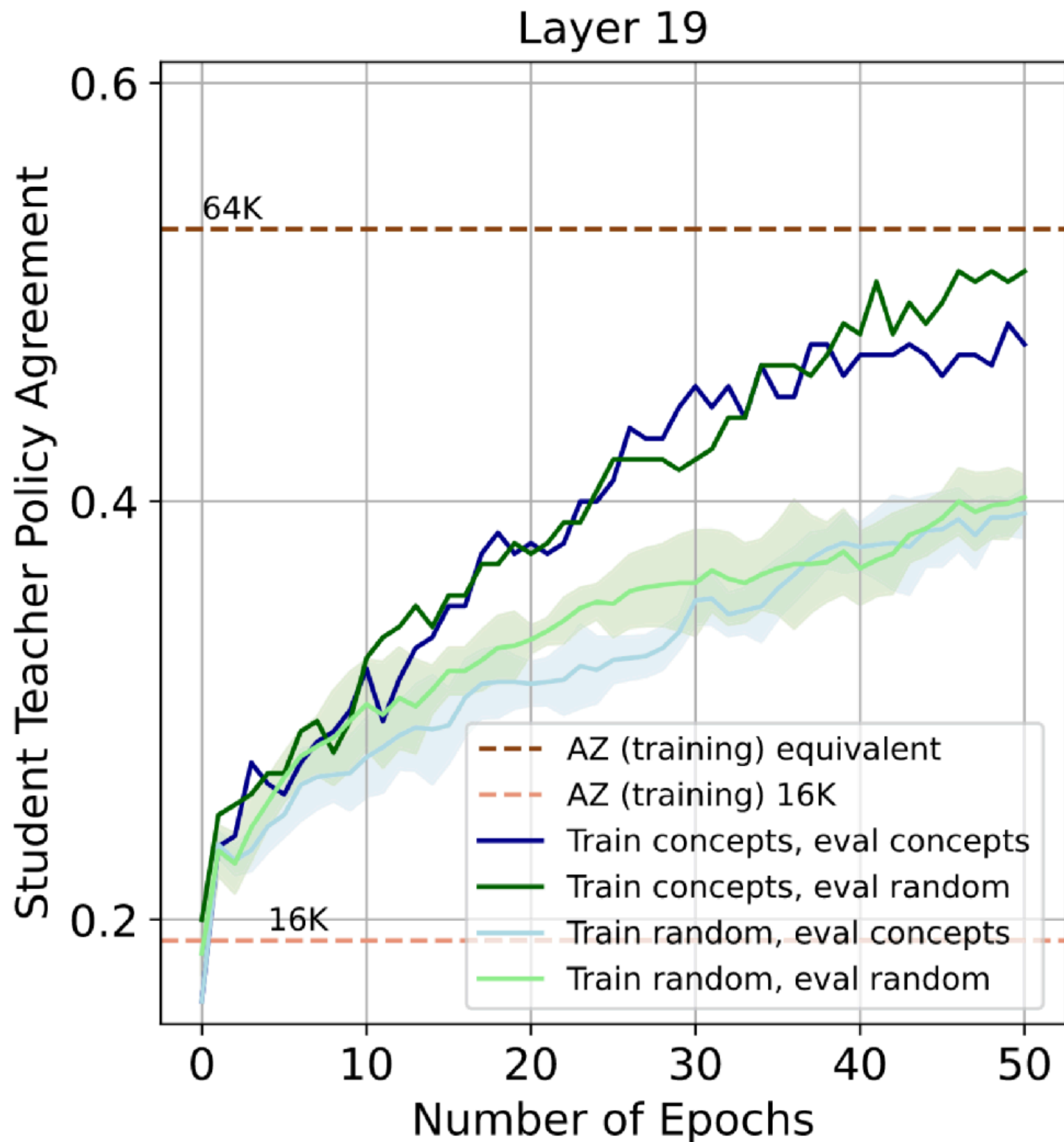


For every board position in the database, compute its \mathbf{z}^+ and \mathbf{z}^- , and find the corresponding concept \mathbf{v}

Evaluate teachability by training an early checkpoint of AZ on the concept

1. Find every board position in the database satisfying $\mathbf{v} \cdot \mathbf{z}^+ \geq \mathbf{v} \cdot \mathbf{z}^-$
2. Find a checkpoint of AZ such that it disagrees w/ the final AZ for at least 80% of the board positions
3. Train the checkpoint for 50 epochs on the board positions against the final AZ's answers

The model learns better training on concept-related positions than random positions



Evaluate novelty by finding concepts that are not well reconstructed by the human concept basis

1. Build a matrix of embeddings for every position played by AI or humans

$$\mathbf{Z}_{\text{AI}} = \begin{bmatrix} \cdots & \mathbf{z}_{\text{AI}}^1 & \cdots \\ & \vdots & \\ \cdots & \mathbf{z}_{\text{AI}}^n & \cdots \end{bmatrix} \quad \mathbf{Z}_{\text{HUM}} = \begin{bmatrix} \cdots & \mathbf{z}_{\text{HUM}}^1 & \cdots \\ & \vdots & \\ \cdots & \mathbf{z}_{\text{HUM}}^m & \cdots \end{bmatrix}$$

2. By SVD, compute the orthonormal row basis \mathbf{U}_{AI} and \mathbf{U}_{HUM}

3. $\text{novelty}(\mathbf{v}) = \text{reconstruction-error}(\mathbf{v}, \mathbf{U}_{\text{HUM}}) - \text{reconstruction-error}(\mathbf{v}, \mathbf{U}_{\text{AI}})$

World-class chess players improved play on concept-related boards through training

Study: 4 grandmasters shown concept-related positions, evaluated before and after for agreement w/ AZ on concept-related puzzles

Grandmaster	Puzzles solved		↑
	Phase 1	Phase 3	
1	0/12	5/12	+42%
2	4/12	7/12	+25%
3	3/12	5/12	+16%
4	6/16	7/15	+6%

Hacking our brains

Thickness, curvature, vertical symmetry reduce the memorability of a symbol

Visual PC1: Thickness

Bottom 10

± ÷ ≠ = + £ × # € ¥

Top 10

▶▶ 🔊♥♣ 🔊◀◀♦▶ 🔔⬛

Visual PC2: Curvature

Bottom 10

▶▶◀◀▶ 🔊📶♦✈️ 🔊⏸️🔔

Top 10

🔄🚫⚙️📄@☯️☮️©®☢️

Visual PC3: Vertical Symmetry











































Bottom 10

°C Ω α ¢ € \$ Σ £ & β

Top 10

+ = ÷ ≠ ± × # † ⏸️ ♀

DALL-E 3 can generate more or less memorable symbols

Word	Memorable	Forgettable	Word	Memorable	Forgettable	Word	Memorable	Forgettable
impression			opinion			revenge		
intelligence			opportunity			risk		
journey			patience			sin		
kind			personality			society		
knowledge			pity			soul		
lack			pleasure			success		
language			possibility			talent		

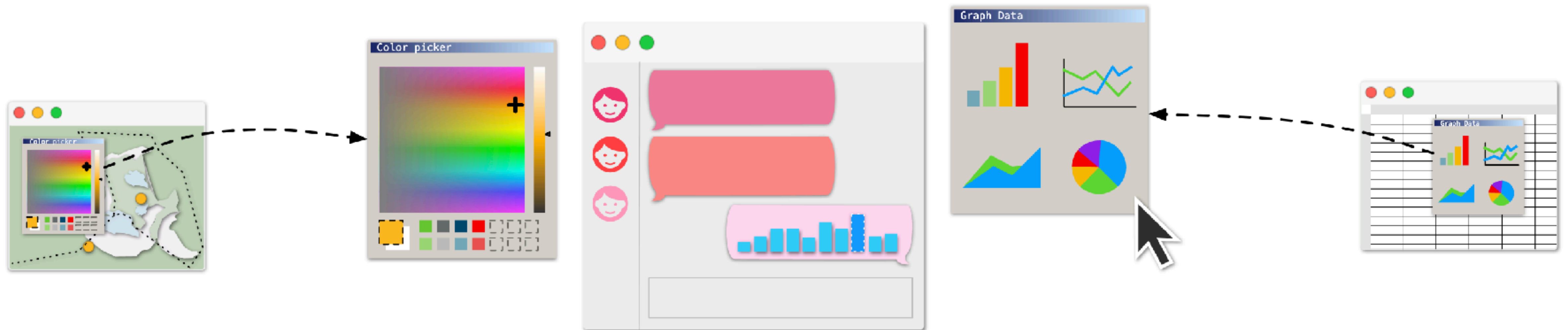
**Generative interfaces,
malleable systems**

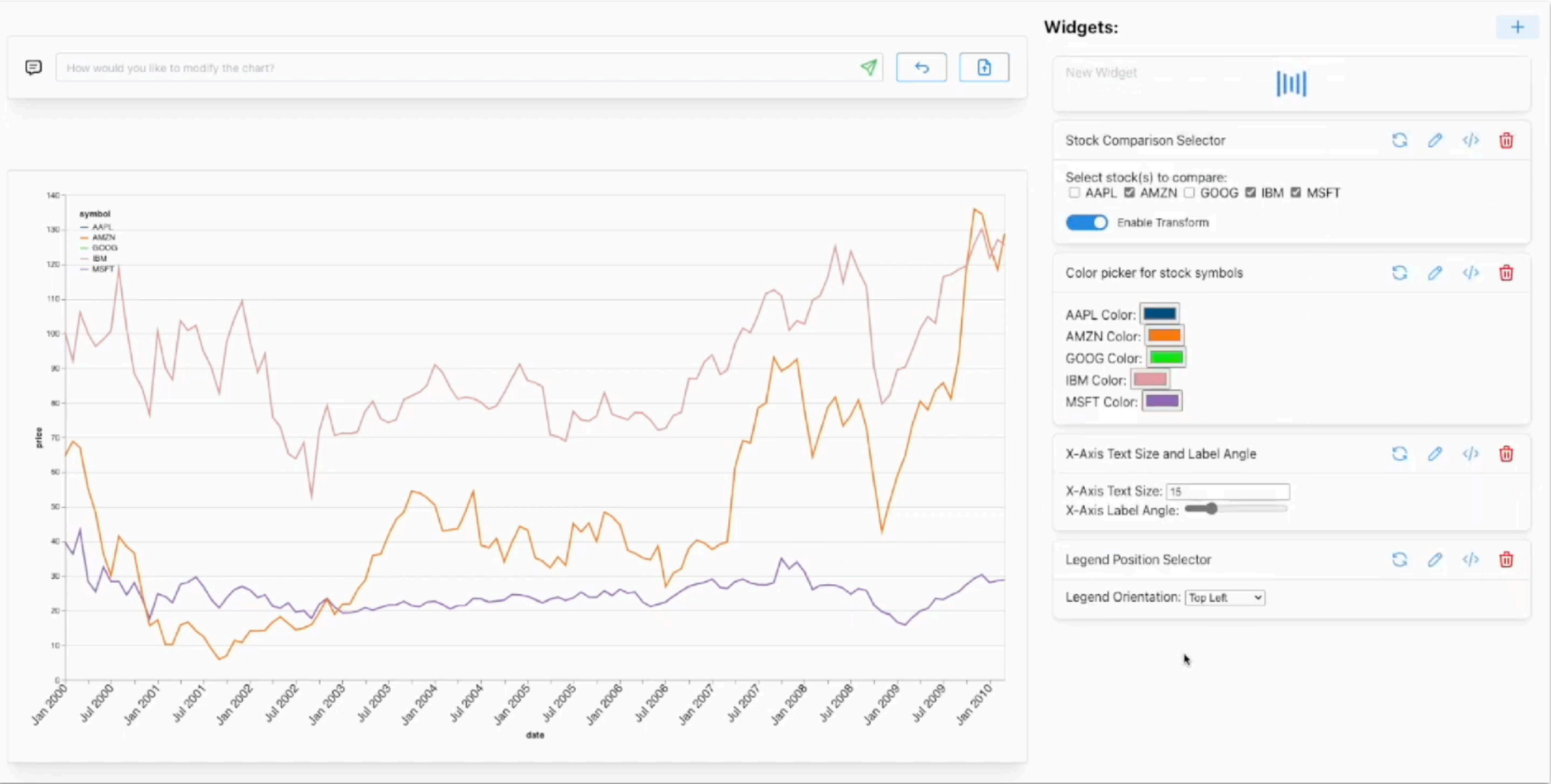
We live in a world of apps

- Spreadsheets are an app, document editing is an app, social media is an app, etc. etc.
- Apps only talk to each other through pre-baked integrations
- Every sufficiently complex desktop app implements keybindings, tabs, window management, folders, tags, ...
 - And each implementation is slightly different from the others!
- If I want a spreadsheet and a document editor together, that's a whole new app!

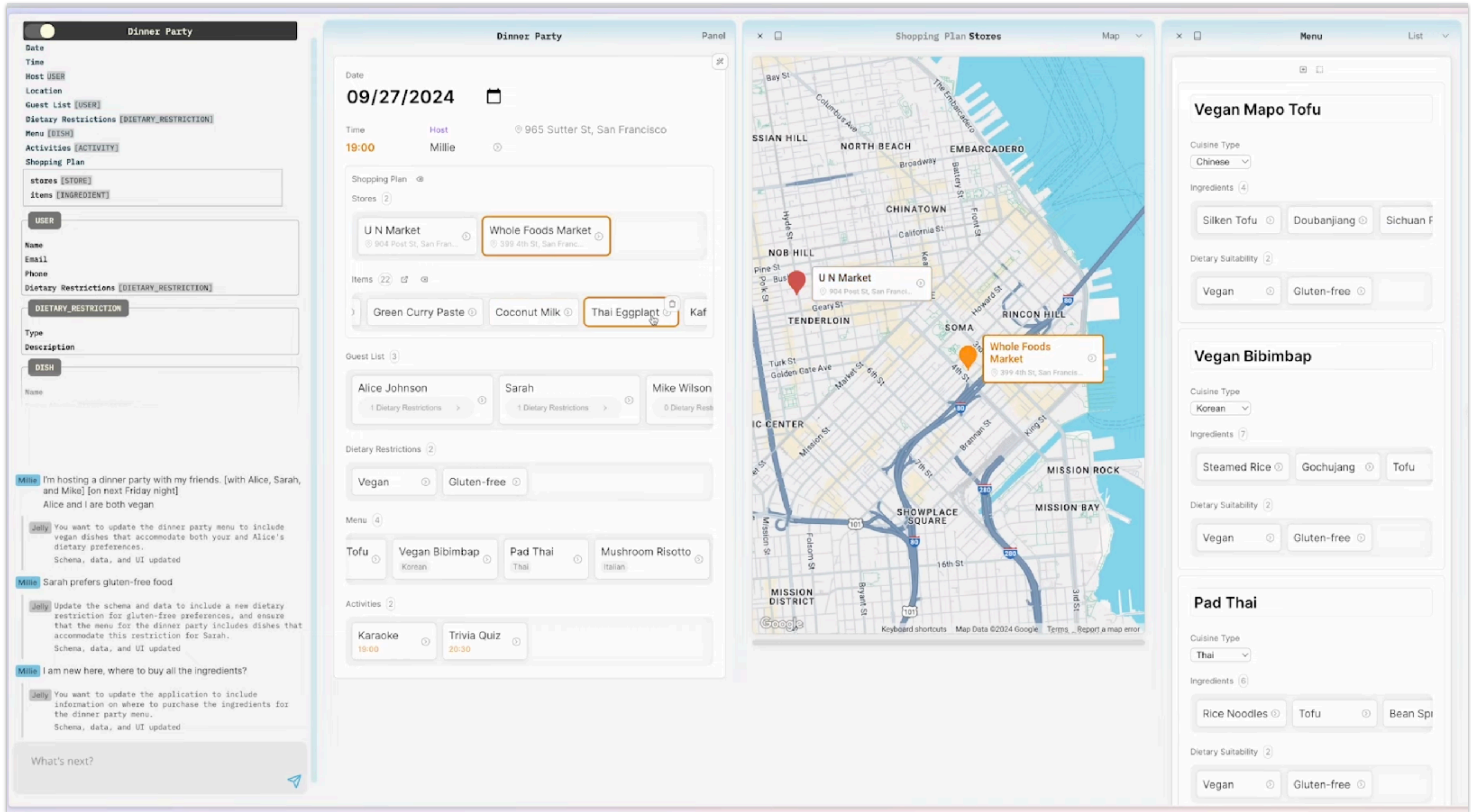
Malleable software aims to increase the power of existing adaptation behaviors by allowing users to pull apart and re-combine their interfaces at the granularity of individual UI elements, such as toolbars, widgets, menus, documents, and devices. In other words, the goal is to erase the boundaries between apps and create an end-user accessible “physics of interfaces” that dictate how different interfaces and documents can be assembled.

— Philip Tchernavskij, *Designing and Programming Malleable Software* (2019)





youtu.be/8Z4yKqm0AAg



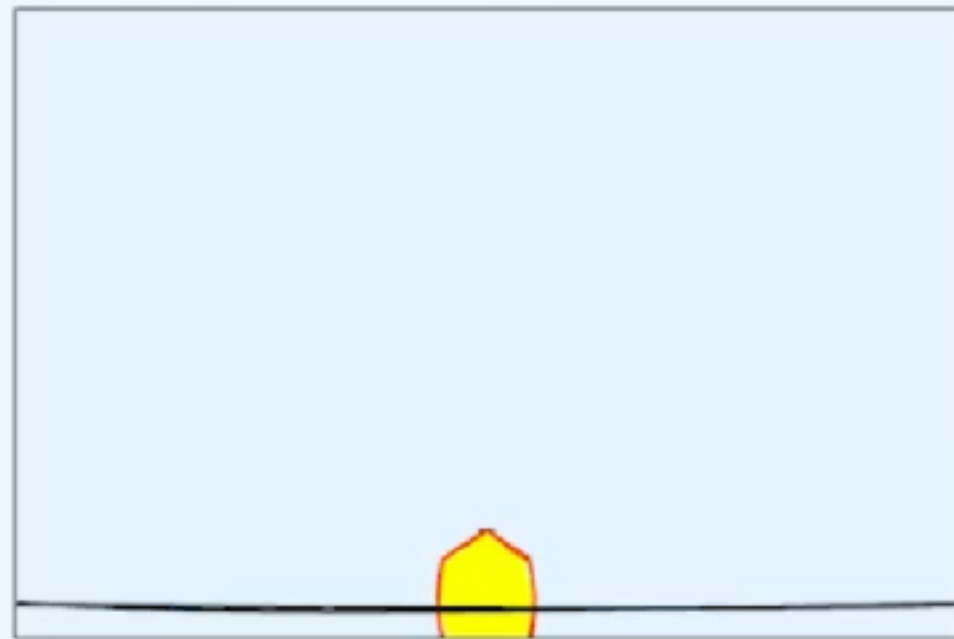
youtu.be/X3cf1U1dFGQ

Back



Paper Lantern Experiment

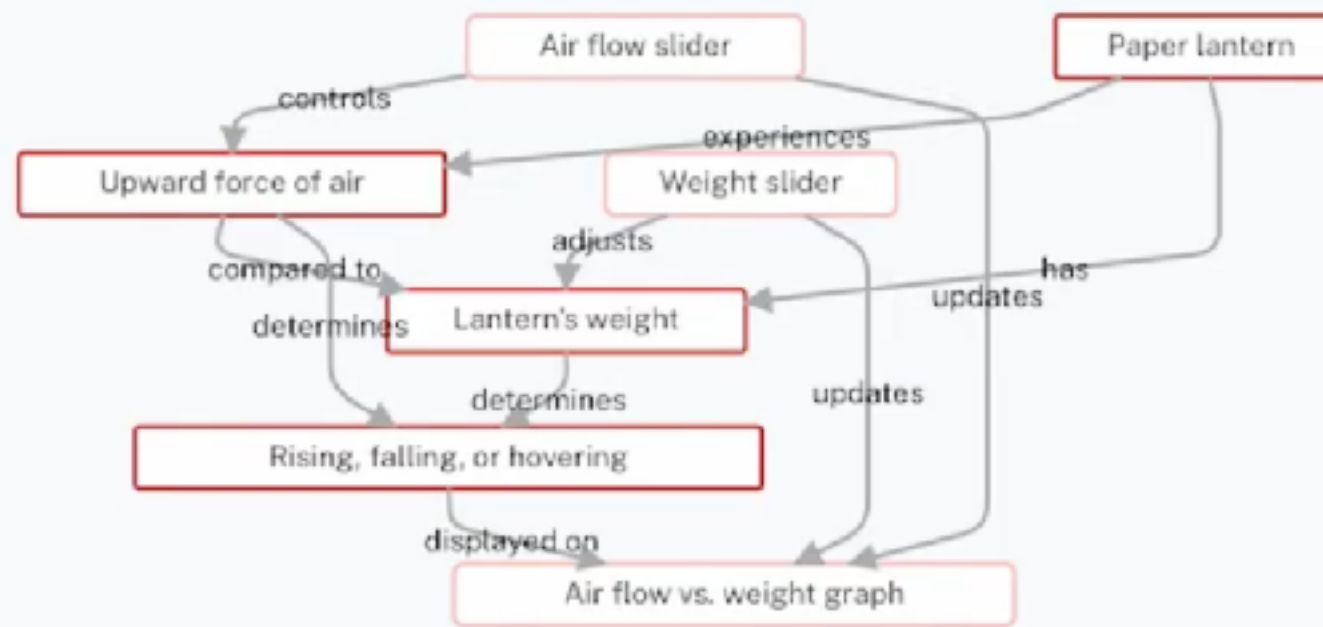
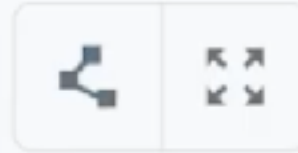
Welcome to the Paper Lantern Experiment! In this simulation, you'll explore how air flow and weight affect the behavior of a paper lantern. Adjust the sliders to change the air flow and the lantern's weight, and observe how these factors influence whether the lantern rises, falls, or hovers in place. Pay attention to the balance between the upward force of air and the lantern's weight!



Air Flow (m/s) 5
Lantern Weight (g) 50



Add Node



Next

Ask me to correct or change anything in this simulation. Annotate the simulation or select a subgraph to help me understand what you want changed.

Test 1/10

Increase air flow to 8 m/s

Expected Result:
The lantern should rise higher in the canvas

Play Test

Commit

Abort

Type your message here...

Send





Brief discussion

- How would you want to grow or bend your day-to-day tools?
- Where do you think these demos will fall apart?
- What software architecture do we need to sustain malleability?

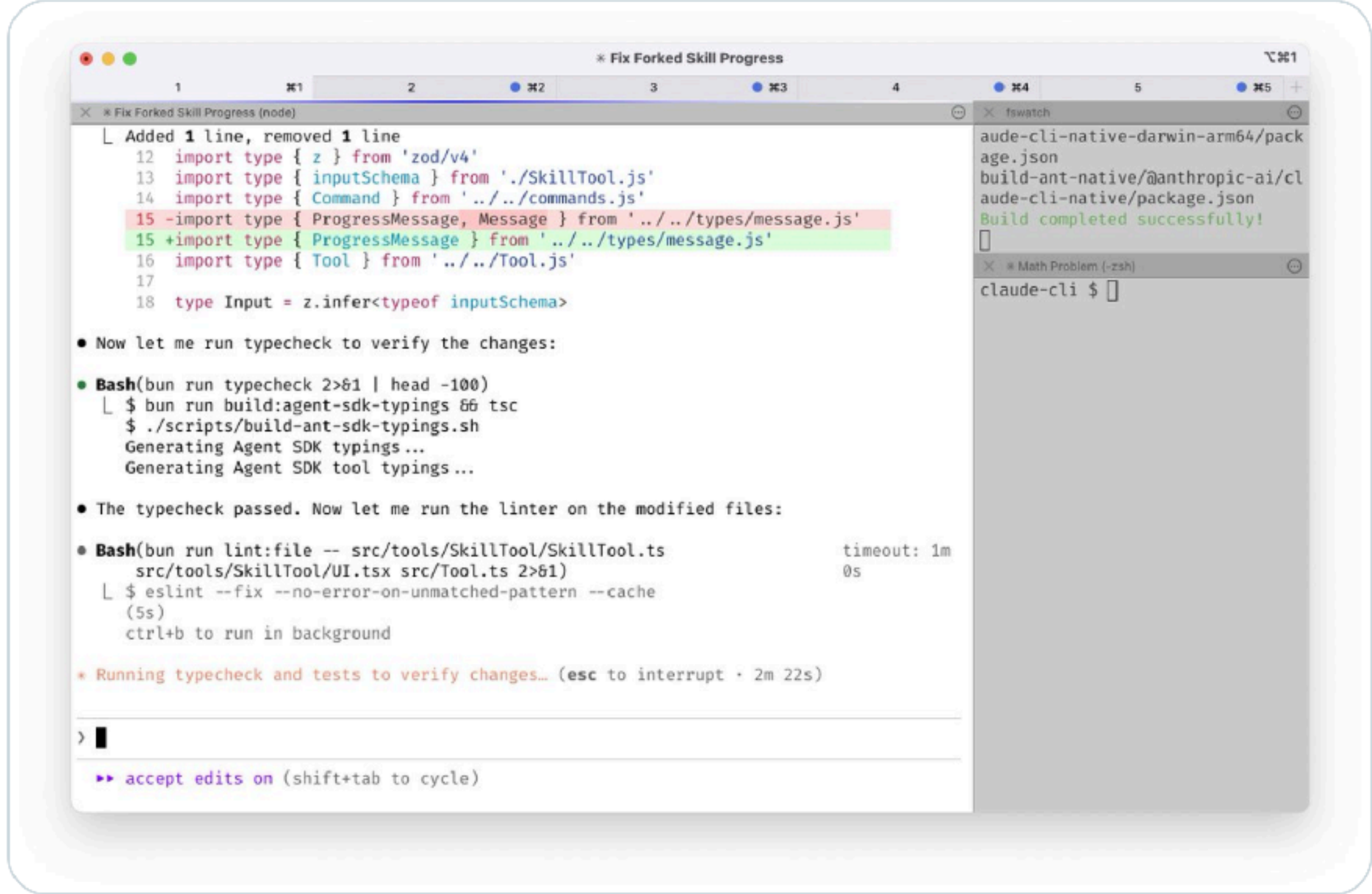
**In the future,
everyone is a manager**

What will the coding interface of 2030 look like?

- Last year has dramatically shifted towards agents and conversational UI for coding
 - Claude Code runs in the terminal but could easily be elsewhere. It's a React app for god's sake...
- Managing multiple agents is currently “more terminal tabs”

 **Boris Cherny** 
@bcherny

1/ I run 5 Claudes in parallel in my terminal. I number my tabs 1-5, and use system notifications to know when a Claude needs input code.claude.com/docs/en/termin...



```
Fix Forked Skill Progress (node)
┌ Added 1 line, removed 1 line
12 import type { z } from 'zod/v4'
13 import type { inputSchema } from './SkillTool.js'
14 import type { Command } from '../commands.js'
15 -import type { ProgressMessage, Message } from '../types/message.js'
15 +import type { ProgressMessage } from '../types/message.js'
16 import type { Tool } from '../Tool.js'
17
18 type Input = z.infer<typeof inputSchema>

• Now let me run typecheck to verify the changes:

• Bash(bun run typecheck 2>&1 | head -100)
┌ $ bun run build:agent-sdk-typings && tsc
└ $ ./scripts/build-ant-sdk-typings.sh
  Generating Agent SDK typings...
  Generating Agent SDK tool typings...

• The typecheck passed. Now let me run the linter on the modified files:

• Bash(bun run lint:file -- src/tools/SkillTool/SkillTool.ts src/tools/SkillTool/UI.tsx src/Tool.ts 2>&1)
┌ $ eslint --fix --no-error-on-unmatched-pattern --cache
└ (5s)
  ctrl+b to run in background

* Running typecheck and tests to verify changes... (esc to interrupt · 2m 22s)

> |

** accept edits on (shift+tab to cycle)
```

2:58 PM · Jan 2, 2026 · 1.1M Views

Intent IDE

The screenshot displays the Intent IDE interface. On the left, a sidebar titled "Public gallery view" shows a progress bar at 50% and a "Changes" tab with 1 change. Below this, it indicates "1 file changed in Space" and shows a commit history with a commit titled "Backend: Open read-only screenshot end...". A "Create PR" button is visible, and a PR description is partially visible: "Allows for public viewing of the".

The central pane shows the code editor for "routes.ts". The code includes imports for security, express-validator, sanitize-html, and authentication middleware. A key change is highlighted: the import of `optionalAuth` from `./middleware/auth` instead of `requireAuth`. The code also shows route definitions for `/api/screenshots/search` and `/api/screenshots/filter`.

On the right, a "Spec" pane contains a "Goal" section: "Allow non-logged-in users to browse the screenshot gallery and change thumbnail grid sizes, while preventing them from adding, editing, or deleting photos. Logged-in users retain full functionality." Below the goal is a "Tasks" list with two items: "Backend: Open read-only screenshot end..." (checked) and "Frontend: Hide auth-only UI for anyony..." (unchecked). The "Acceptance Criteria" section lists several conditions, such as "Unauthenticated users can load / and see the screenshot grid without being redirected to /auth".

For next time...



<https://docs.google.com/document/d/1rAzujTFcmQyVEbTSGmu3OH43BpXju4qEuYPXOPU9vRE/edit?usp=sharing>