

# what makes software work?

Daniel Jackson · Google · May 14, 2024



When you go to design a house you talk to an architect first, not an engineer. Why is this?

Because the criteria for what makes a good building fall outside the domain of engineering.

Similarly, in computer programs, the selection of the various components must be driven by the conditions of use.

How is this to be done? By software designers.

Mitchell Kapor, *A Software Design Manifesto* (1996)

what are these components?  
what is software design?

what does  
success look like?

story #1:  
TBL's Web  
(1989)

# what was novel about the web?

## World Wide Web

The WorldWideWeb (W3) is a wide-area [hypermedia](#) information retrieval initiative aiming to give universal access to a large universe of documents.

Everything there is online about W3 is linked directly or indirectly to this document, including an [executive summary](#) of the project, [Mailing lists](#) , [Policy](#) , November's [W3 news](#) , [Frequently Asked Questions](#) .

### [What's out there?](#)

Pointers to the world's online information, [subjects](#) , [W3 servers](#), etc.

### [Help](#)

on the browser you are using

### [Software Products](#)

A list of W3 project components and their current state. (e.g. [Line Mode](#) ,X11 [Viola](#) , [NeXTStep](#) , [Servers](#) , [Tools](#) , [Mail robot](#) , [Library](#) )

### [Technical](#)

Details of protocols, formats, program internals etc

### [Bibliography](#)

Paper documentation on W3 and references.

### [People](#)

A list of some people involved in the project.

### [History](#)

A summary of the history of the project.

### [How can I help ?](#)

If you would like to support the web..

### [Getting code](#)

Getting the code by [anonymous FTP](#) , etc.

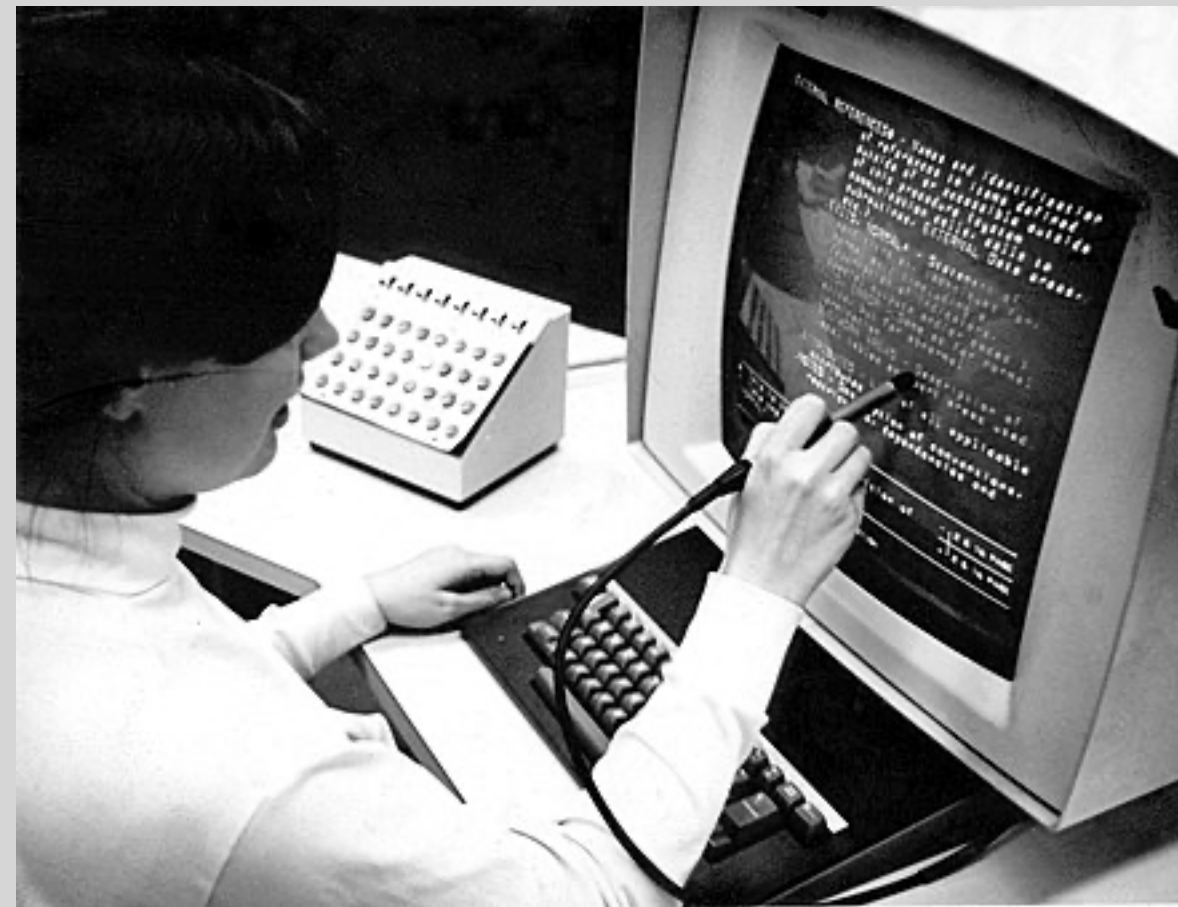
<http://info.cern.ch/hypertext/WWW/TheProject.html>

the technology?

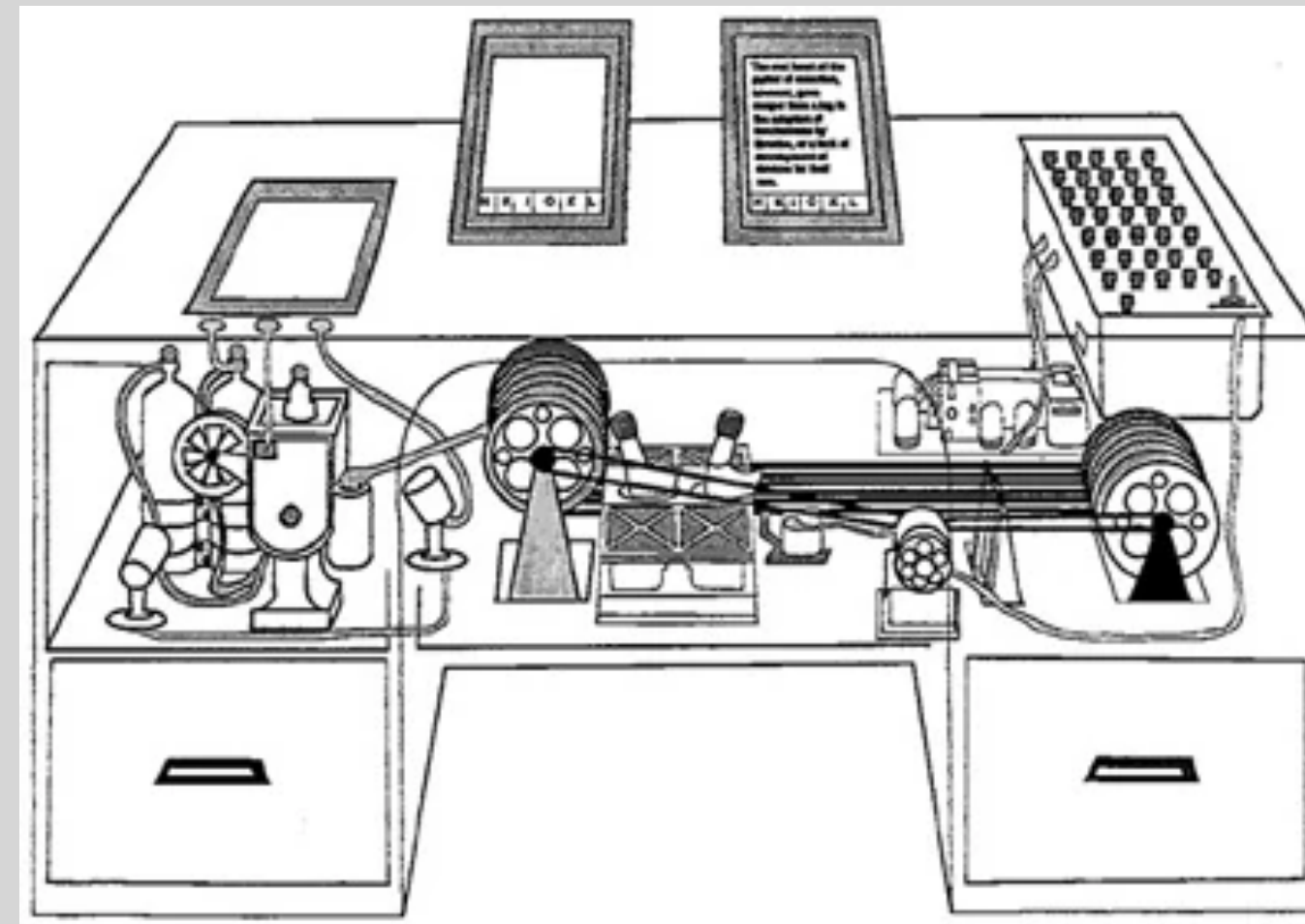


NeXT computer  
660MB hard disk  
Motorola 68030, 25MHz  
17" monitor with Display Postscript  
built in ethernet connectivity

# hypertext?



Hypertext Editing System  
(Nelson & van Dam, 1967)

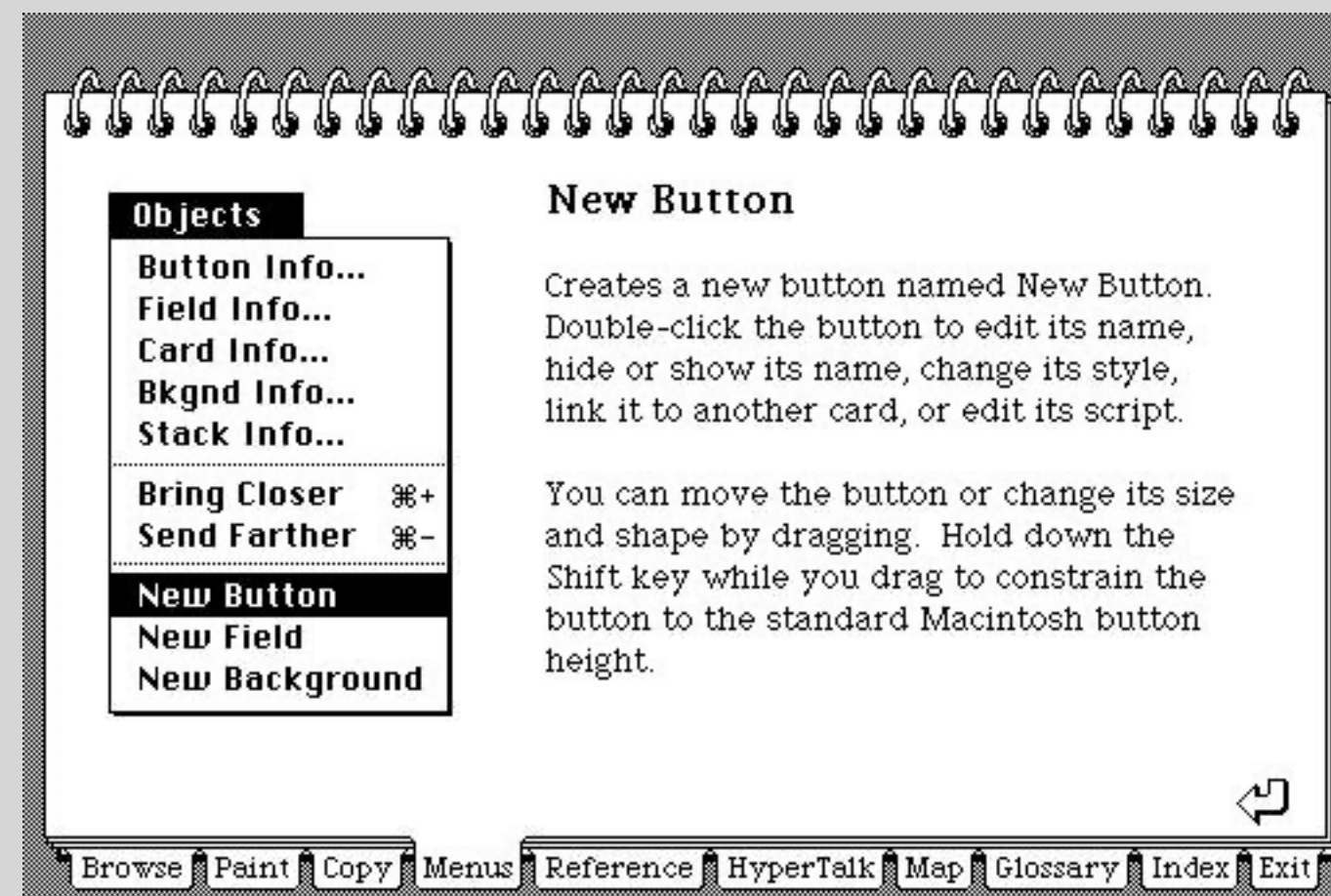


Memex  
(Vannevar Bush, 1945)

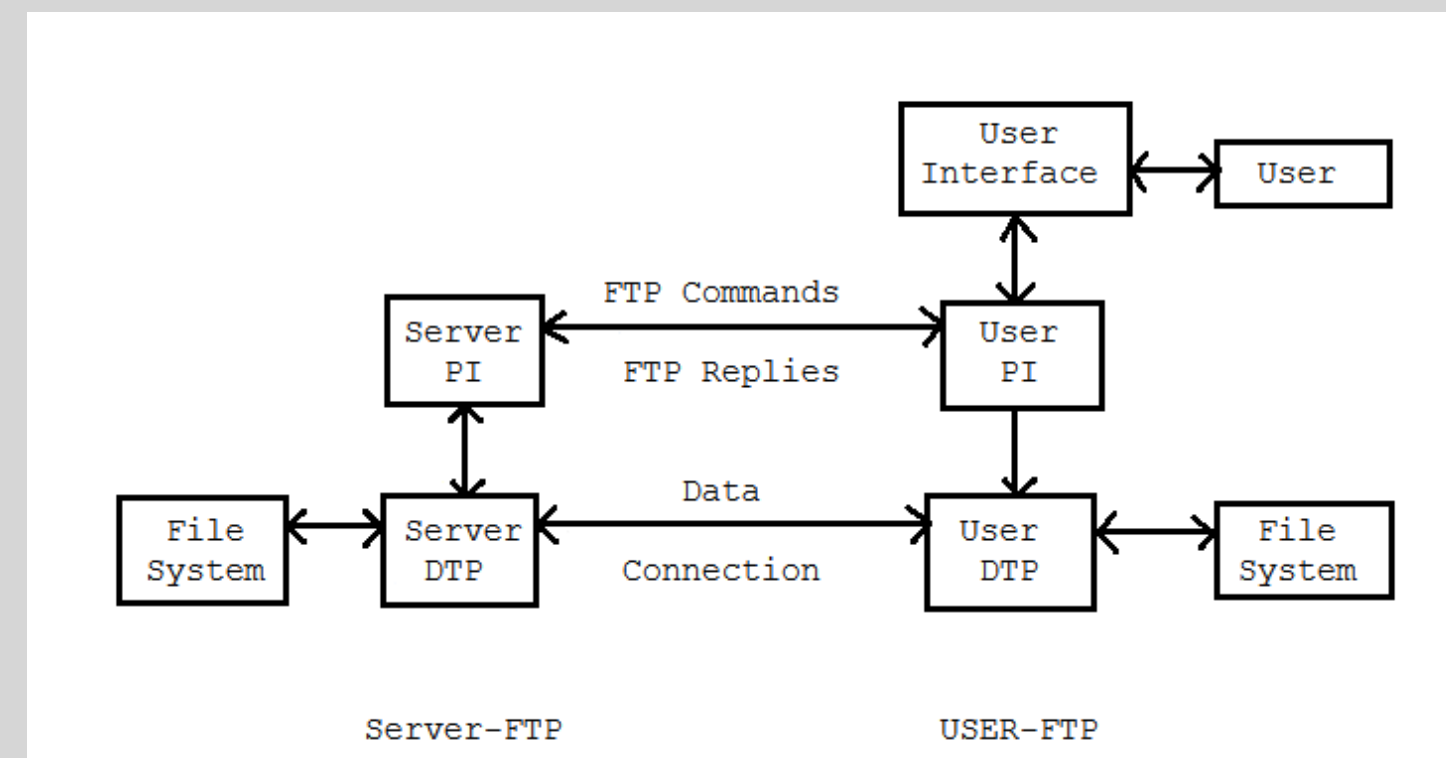
```
<!DOCTYPE motd [ <!E
<motd>
<!-- created: 2003-12-12-->
<sentence>Do not throw
out the <keep>baby</>
with the
<refuse>dirty</>,
<refuse>stinky</>,
<refuse>bathwater</>.
</>
<!-- finish this later-->
</motd>
```

**SGML**

SGML  
(Charles Goldfarb, 1986)

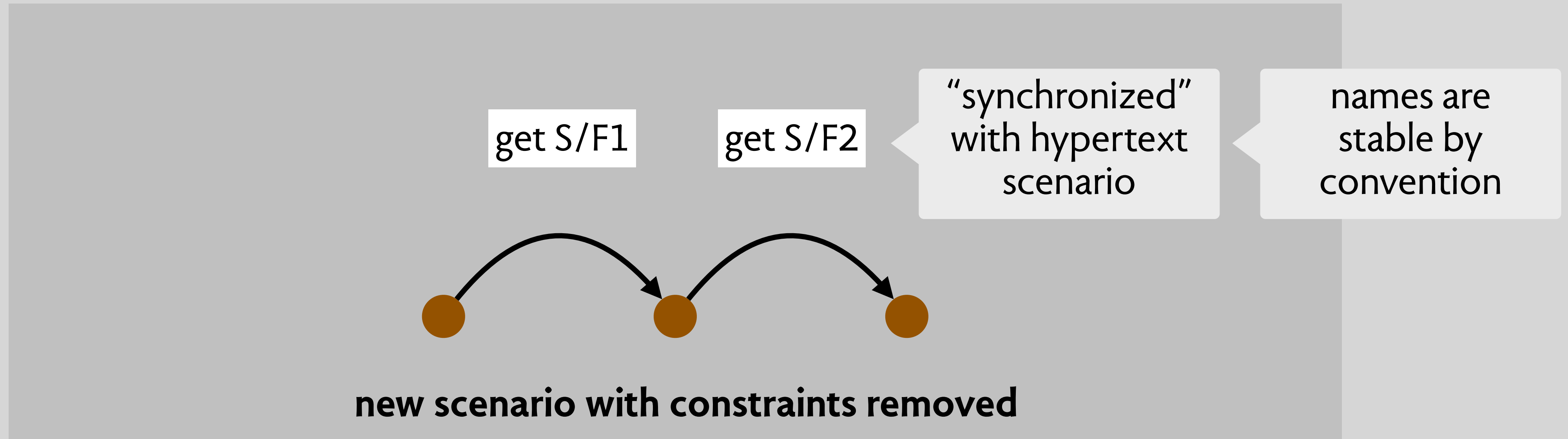
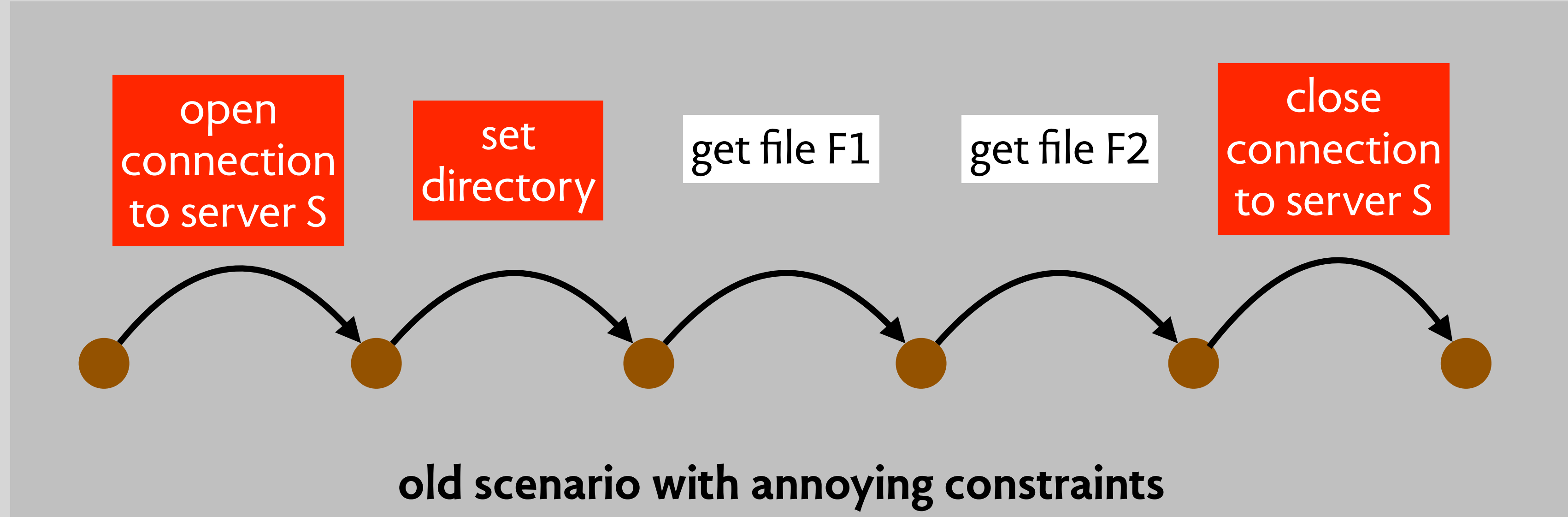


Apple HyperCard  
(Bill Atkinson, 1987)



File Transfer Protocol  
(Abhay Bhushan, 1971)

# a new way to get content





this new scenario is embodied in the concept of the URL

## World Wide Web

The WorldWideWeb (W3) is a wide-area [hypermedia](#) information retrieval initiative aiming to give universal access to a large universe of documents.

Everything there is online about W3 is linked directly or indirectly to this document, including an [executive summary](#) of the project, [Mailing lists](#) , [Policy](#) , November's [W3 news](#) , [Frequently Asked Questions](#) .

### [What's out there?](#)

Pointers to the world's online information, [subjects](#) , [W3 servers](#), etc.

### [Help](#)

on the browser you are using

### [Software Products](#)

A list of W3 project components and their current state. (e.g. [Line Mode](#) ,X11 [Viola](#) , [NeXTStep](#) , [Servers](#) , [Tools](#) , [Mail robot](#) , [Library](#) )

### [Technical](#)

Details of protocols, formats, program internals etc

### [Bibliography](#)

Paper documentation on W3 and references.

### [People](#)

A list of some people involved in the project.

### [History](#)

A summary of the history of the project.

### [How can I help ?](#)

If you would like to support the web..

### [Getting code](#)

Getting the code by [anonymous FTP](#) , etc.

<http://info.cern.ch/hypertext/WWW/TheProject.html>

*story #2:*  
iPod (2001)

what was novel about the iPod?



Apple iPod  
(2001)

# a new industrial design?



**Apple iPod**  
(Jony Ive, 2001)



**Braun Pocket Radio T3**  
(Dieter Rams, 1958)

# a technology breakthrough?



iPod 5GB drive  
(Toshiba, 2001)

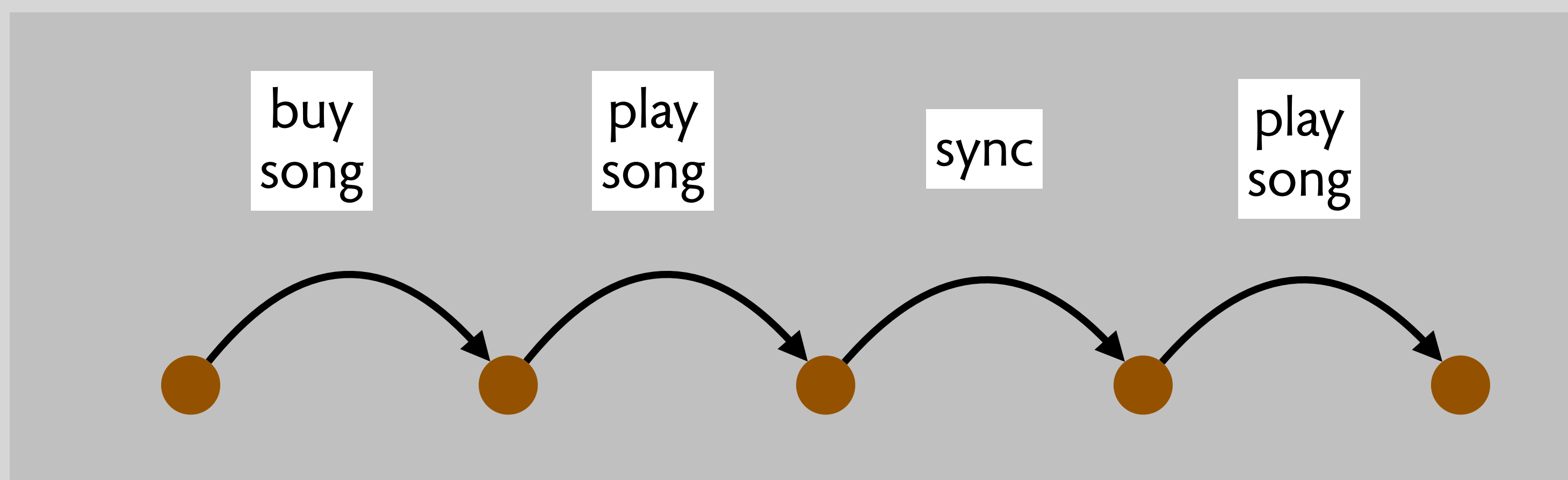
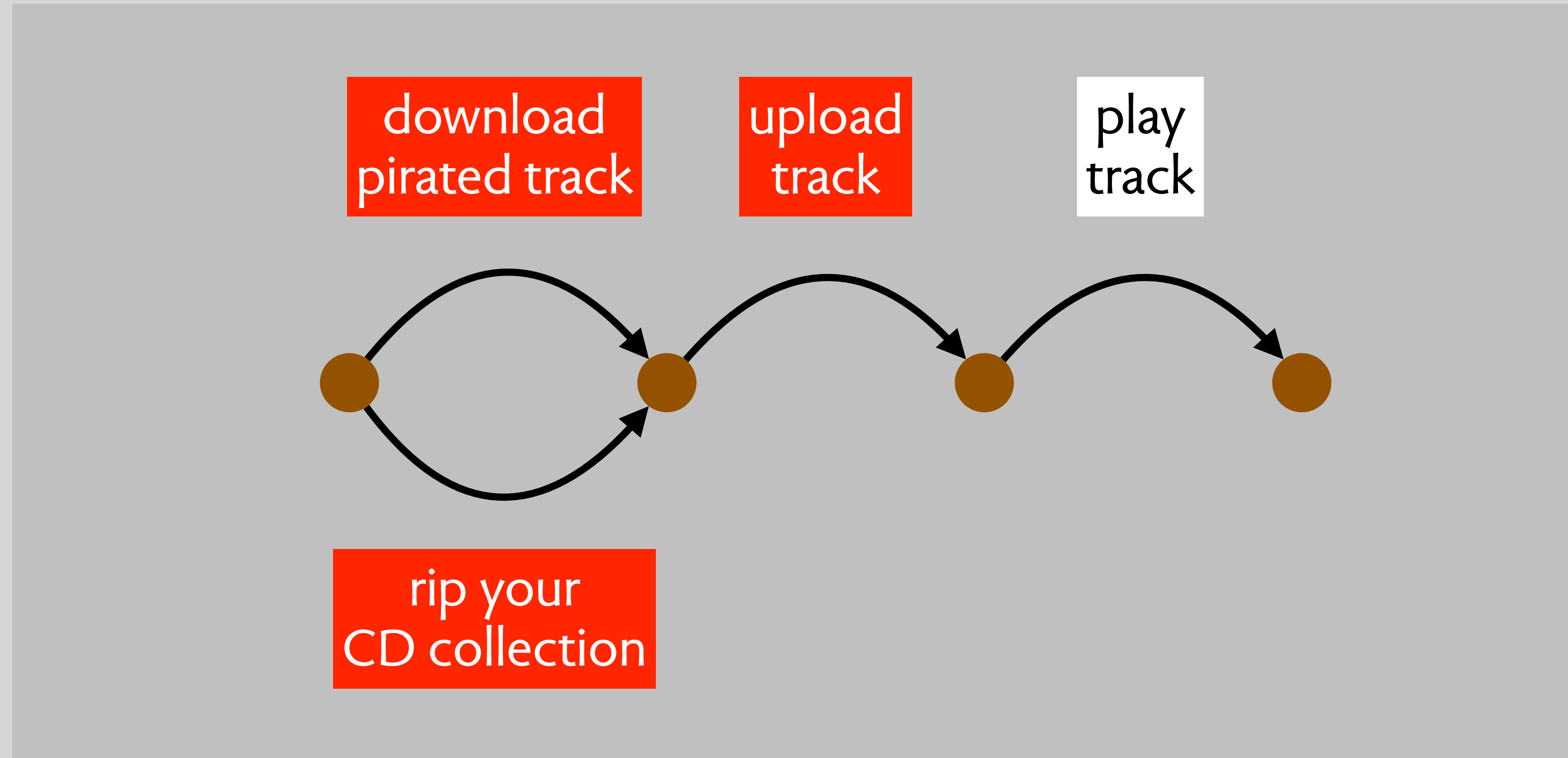


FireWire serial bus  
(Apple, 1995)

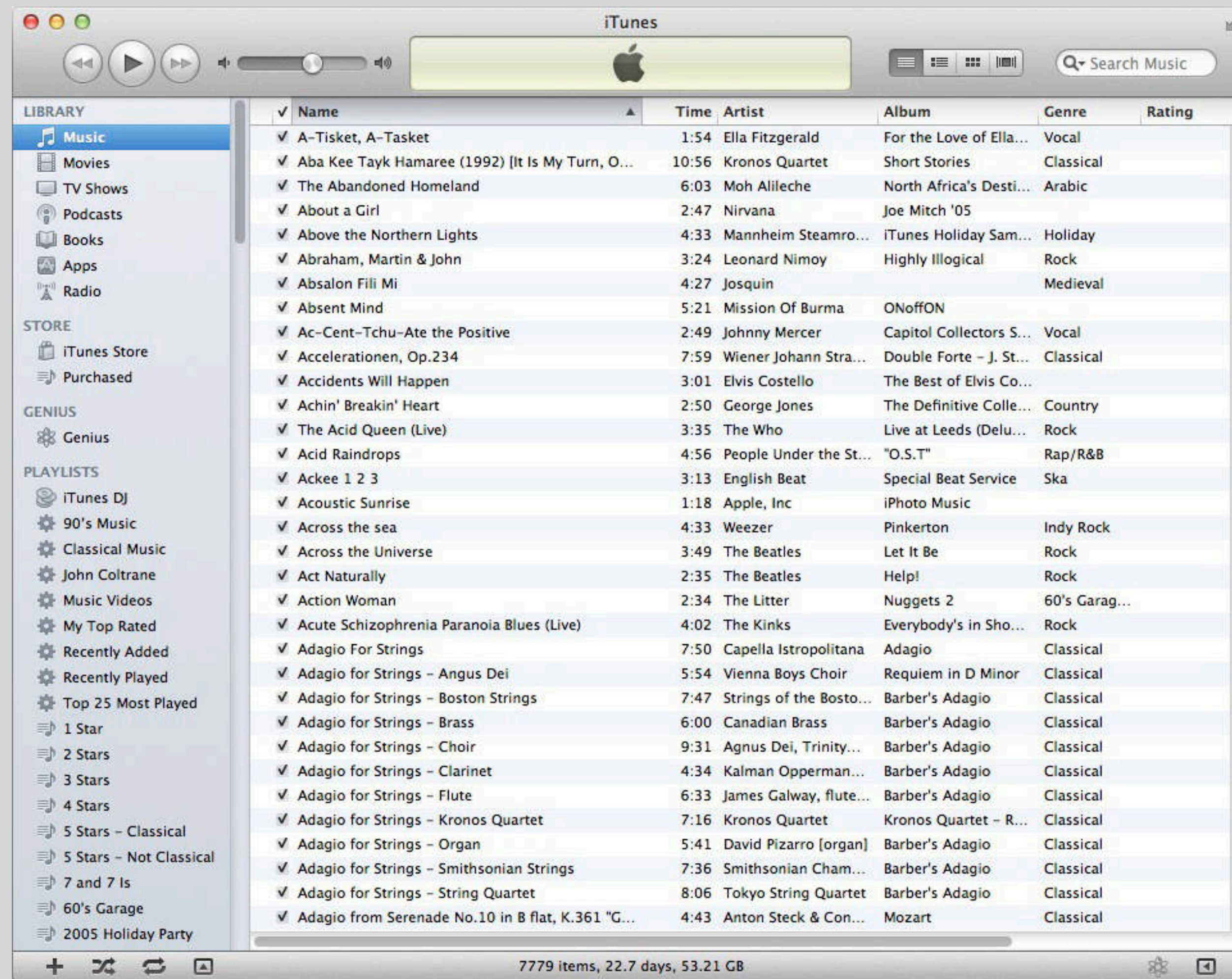


Personal Jukebox 5GB drive  
(DEC, 1999)

# the ipod scenario



# supporting cast: iTunes



iTunes Media Player (2001); iTunes Music Store (2003)

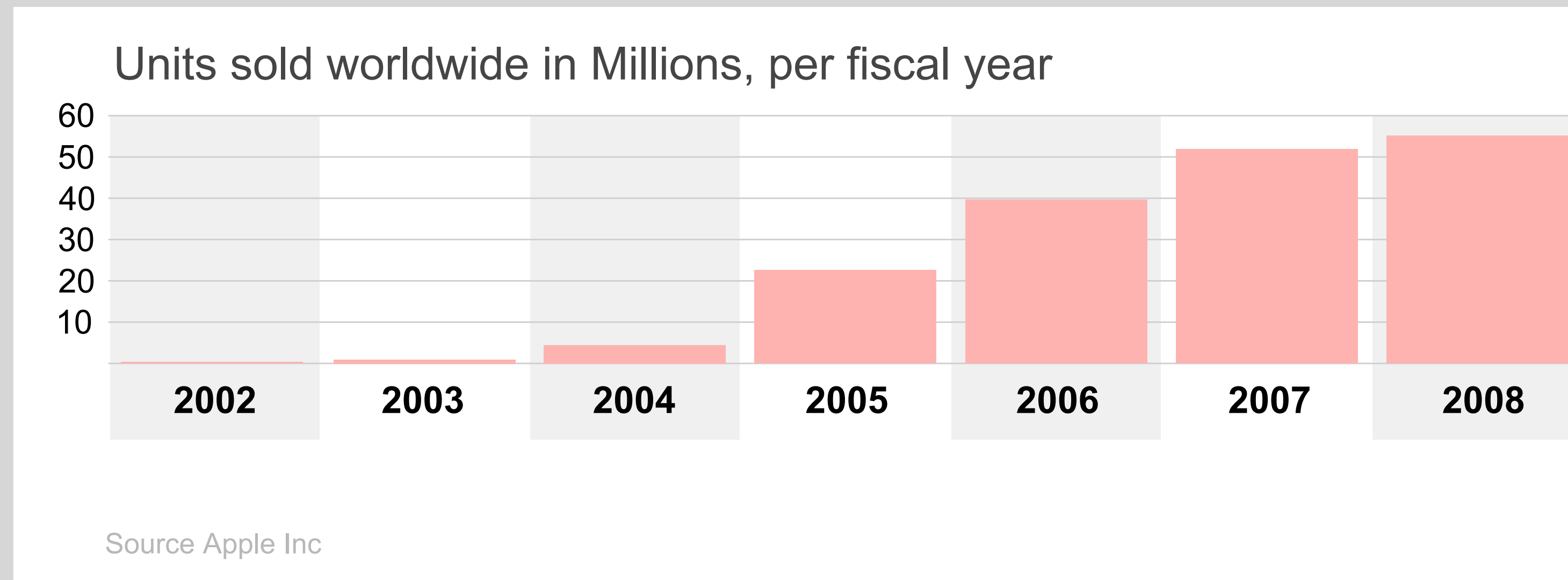
# explosion in sales came after iTunes took off

iPod  
appears

iTunes  
store

100m  
songs sold

iPod  
dominates






# how did Sony fail?




## 1 Connecting your Network Walkman to the computer

For details, refer to the instruction manual of your Network Walkman.

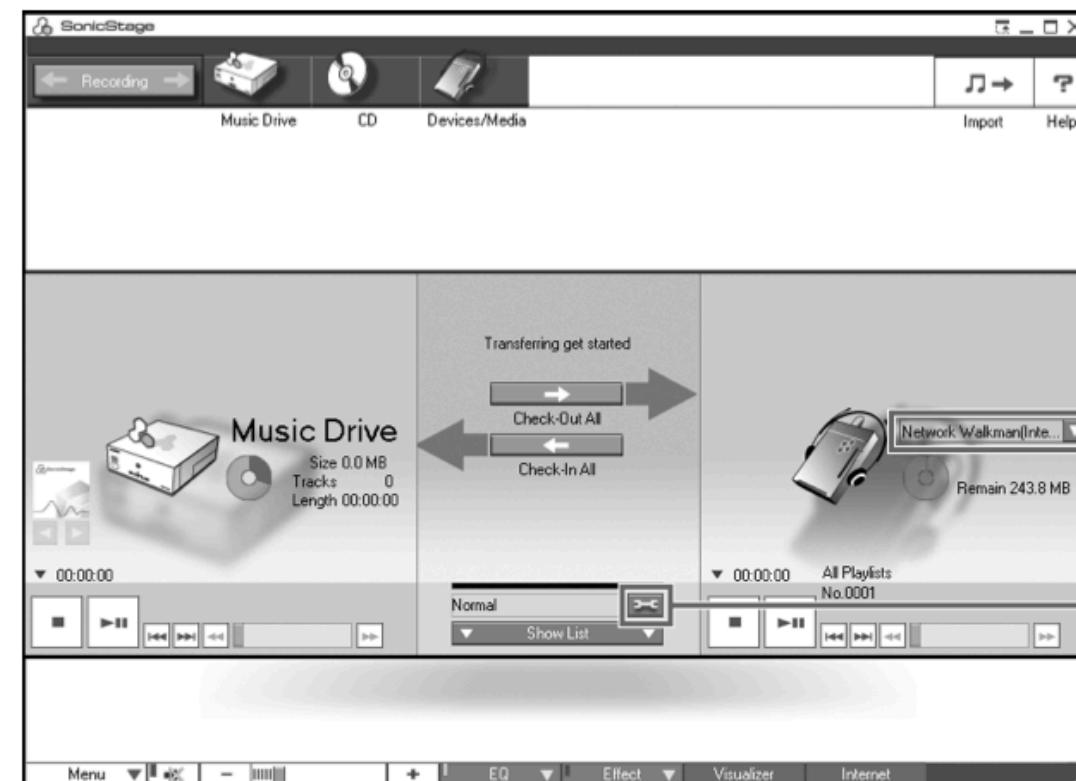
When using a device supporting “MG Memory Stick”, insert the “MagicGate Memory Stick” media into the device before connecting to the computer.

## 2 Transferring audio files to your Network Walkman – Check-out

1 Click  to display the Recording Assistant Window.

2 Click and select  (Music Drive) for [1 Sources] and  (Device/Media) for [2 Targets], and then click 3  (Jump to Record Window).

The Record/Transfer window is displayed.



Device/media  
Drop-down list

Transfer mode button

3 Select the device/media from the drop-down list at the right of the window.

When using the Network Walkman supporting two or more medium types, select the target medium.

When you check out to the built-in flash memory, select “Network Walkman (Internal),” and when you check out to the “Memory Stick”, select “Network Walkman (Memory...)”.

all ingredients of success (1999)

Network Walkman: a digital audio player

Bitmusic: a store for downloading songs

what went wrong?

proprietary compression scheme (ATRAC)?


store limited to release in Japan?

DRM controls made trouble for users?

no simple usage scenario?

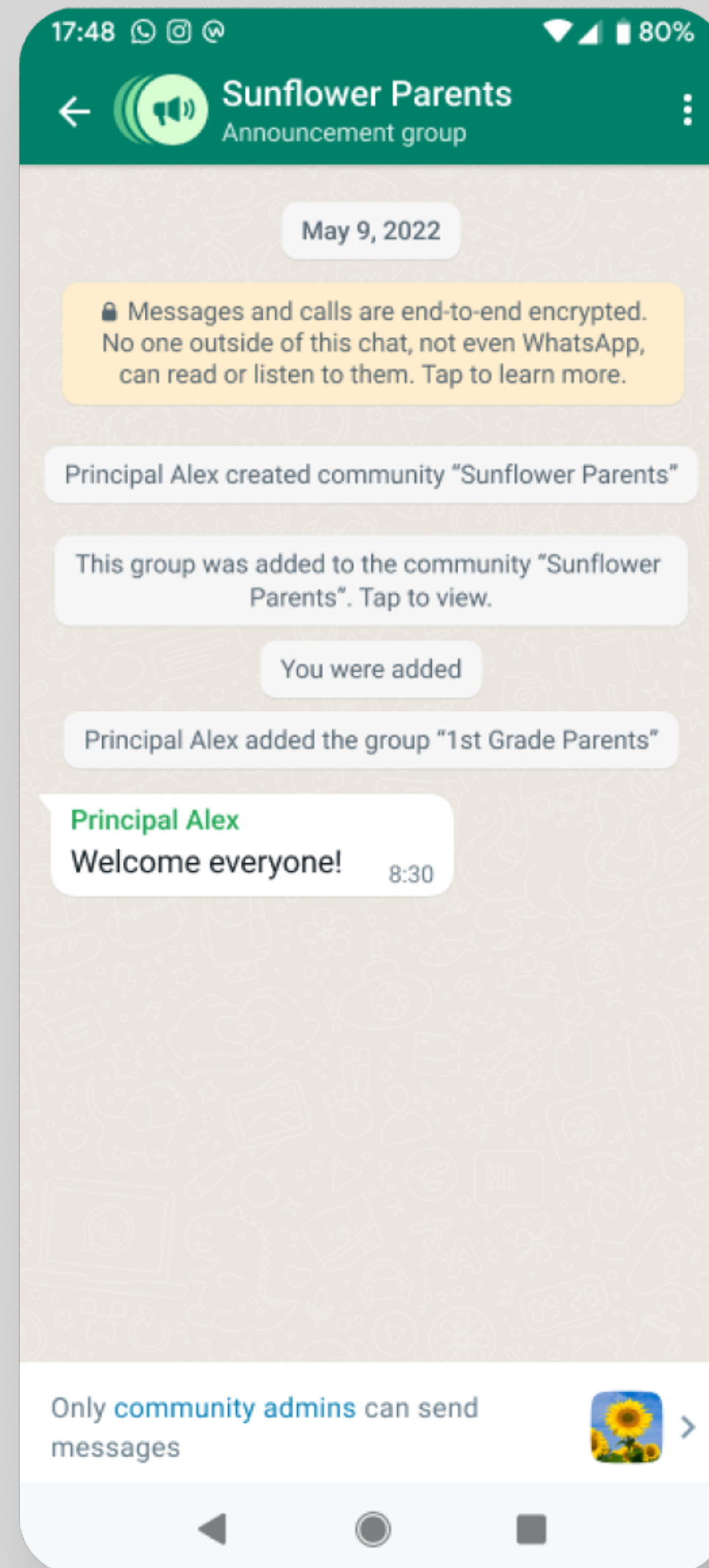


RIP 2d gen iPod Shuffle

Put some music on.  
Built-in clip. 240 songs. Up to 12-hour battery. Just \$79. One size fits all.  iPod shuffle

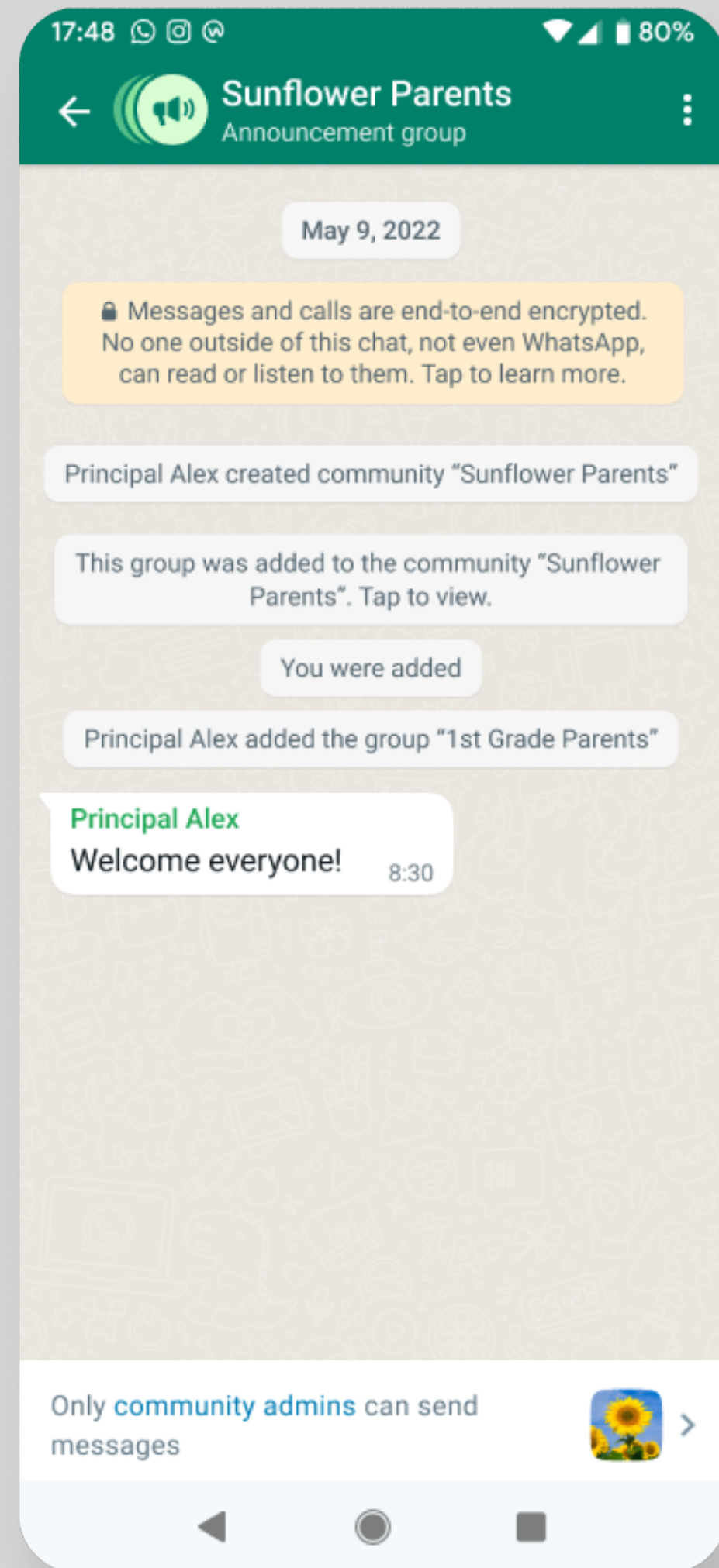
story #3  
WhatsApp  
(2009)

# what was novel about WhatsApp?



WhatsApp (2009)

# was it free texting?

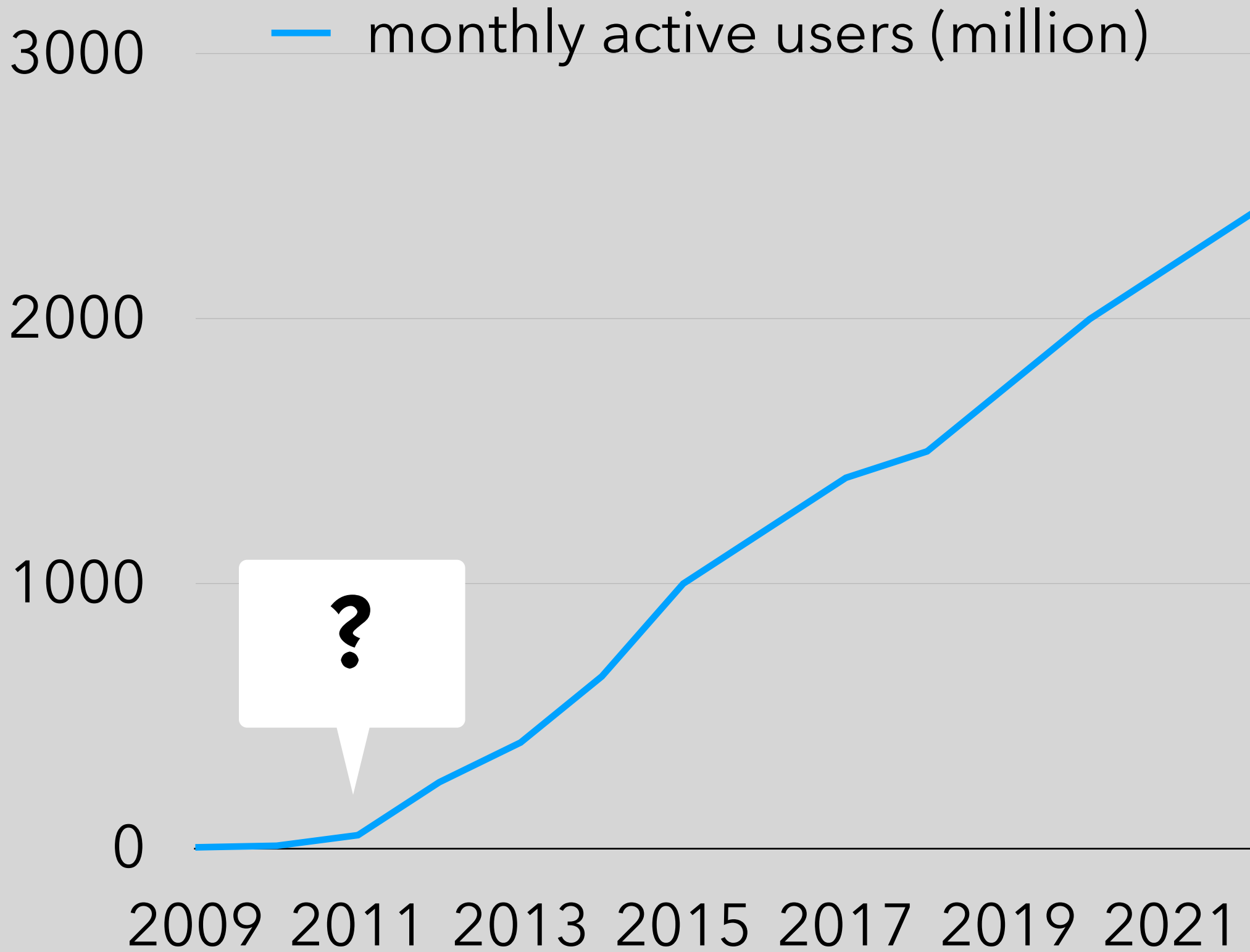
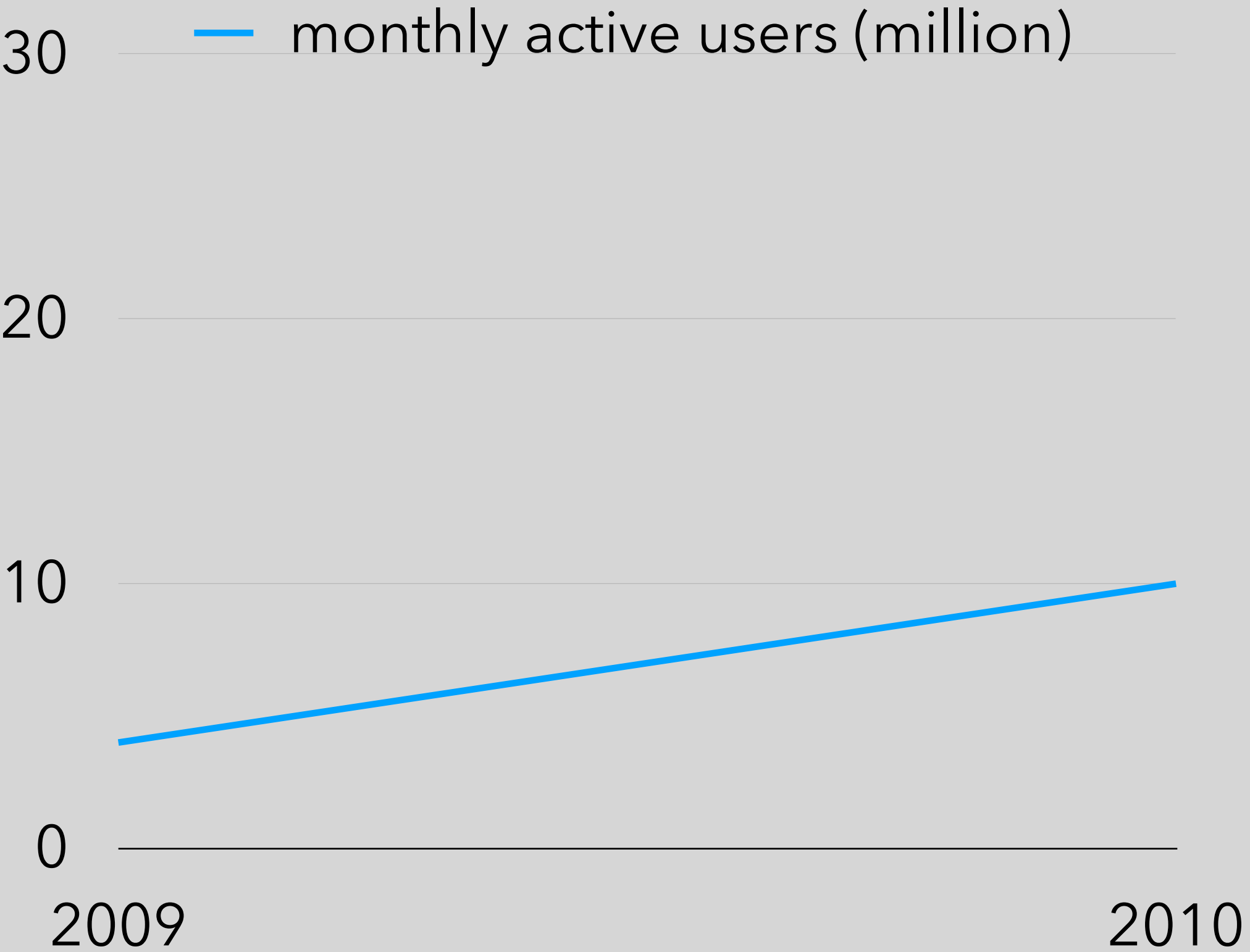
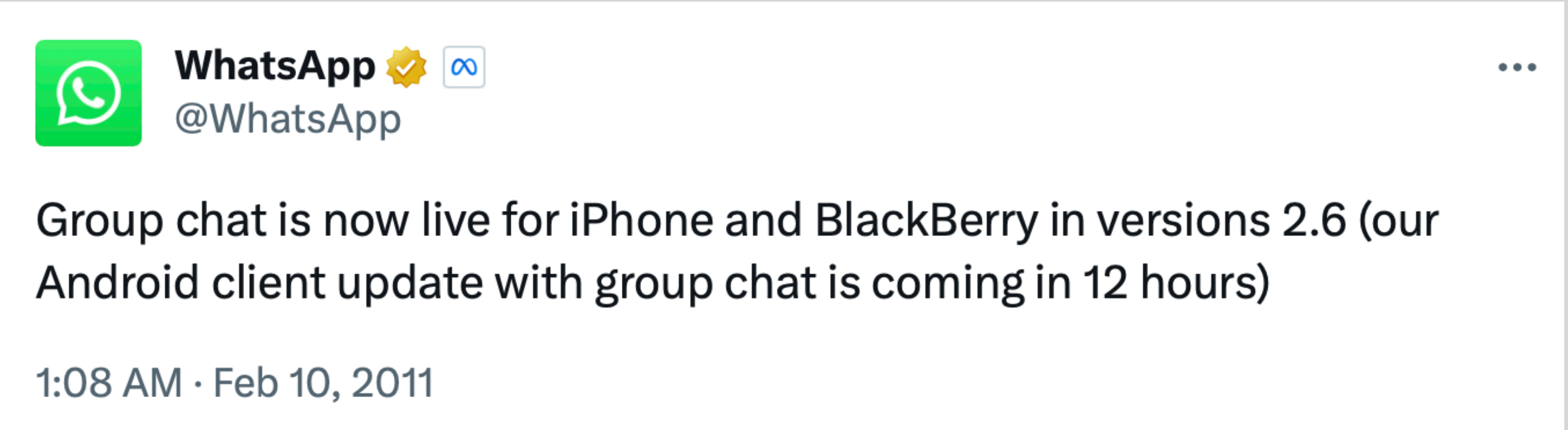


WhatsApp (2009)

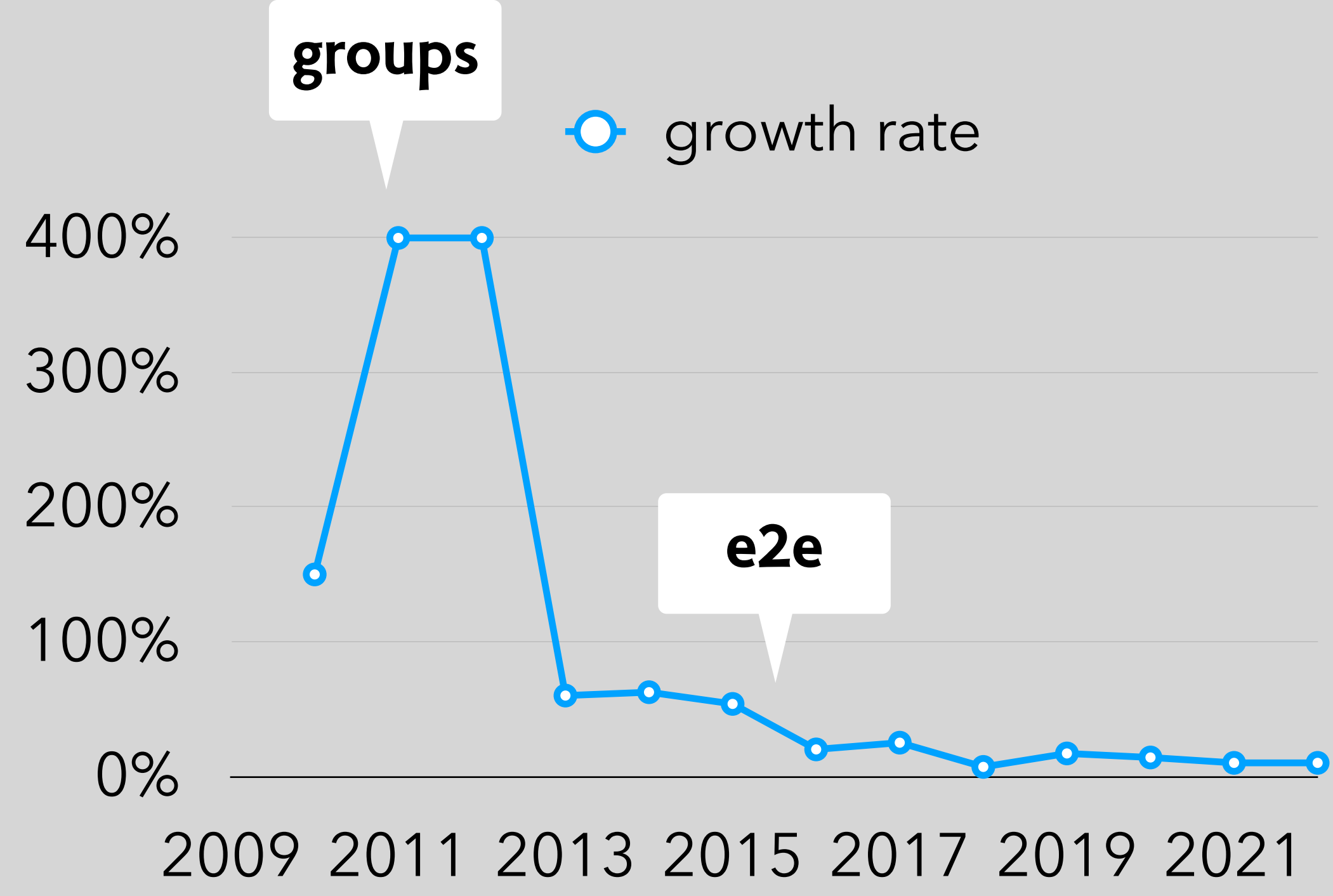
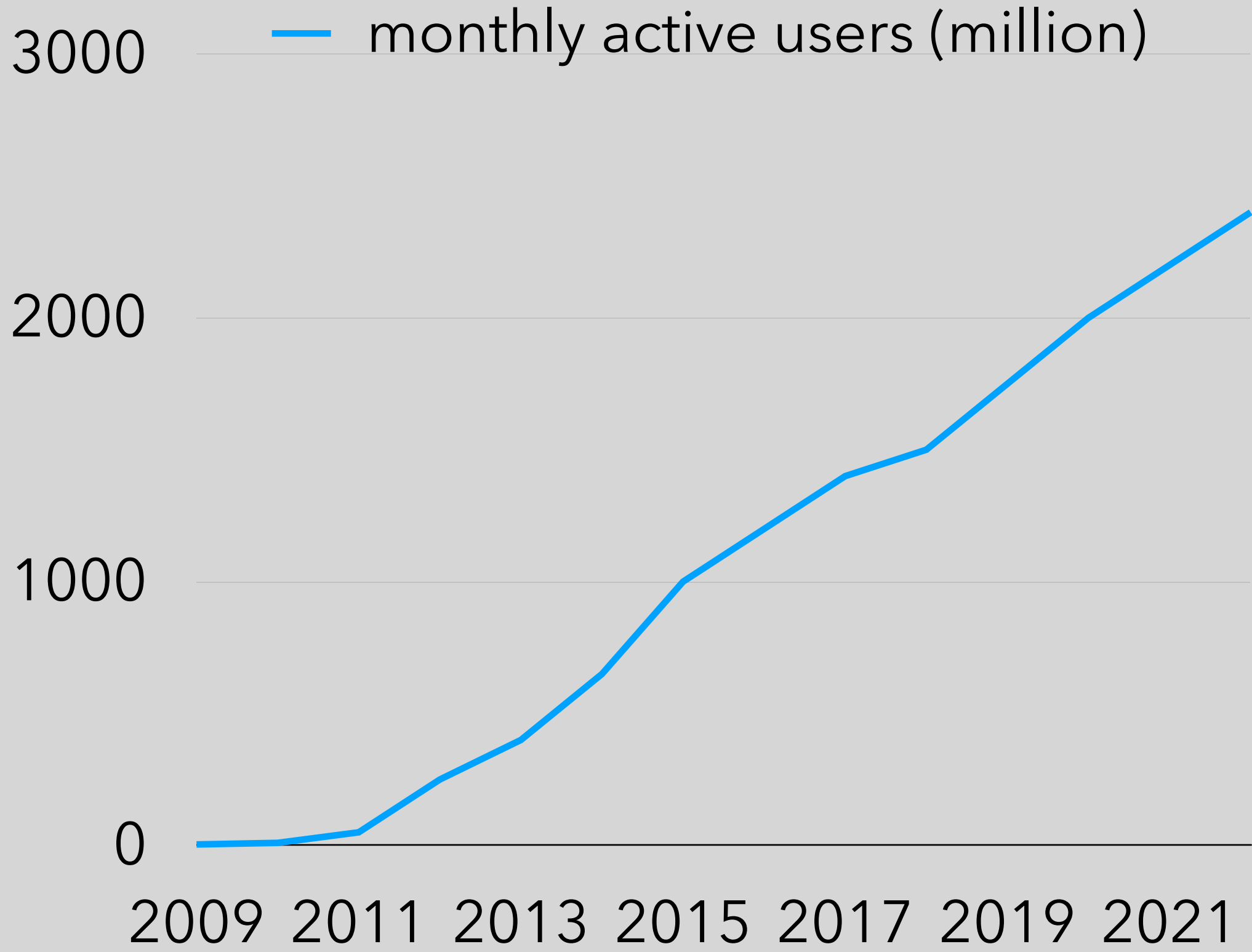


**first free-texting app**  
TextFree/Pinger (2006)

# whatsapp's big moment



# growing with groups



# what was happening in 2011

## *Start-Ups Are in a Rush to Bring the Chat Room to the Smartphone*

Share full article



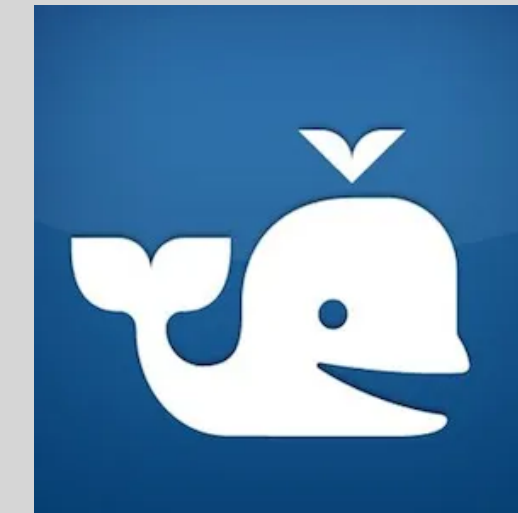
Caleb Elston, left, and David Kasper, co-founders of the social messaging site Yobongo, in the company's offices in San Francisco. Jim Wilson/The New York Times

By Jenna Wortham

March 10, 2011



GroupMe



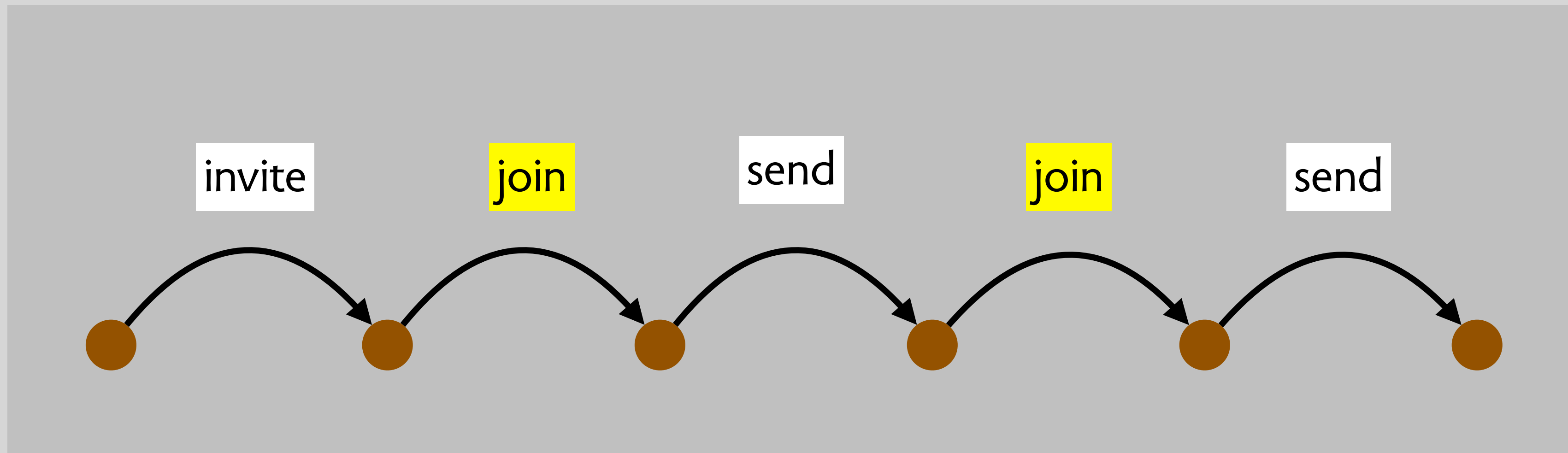
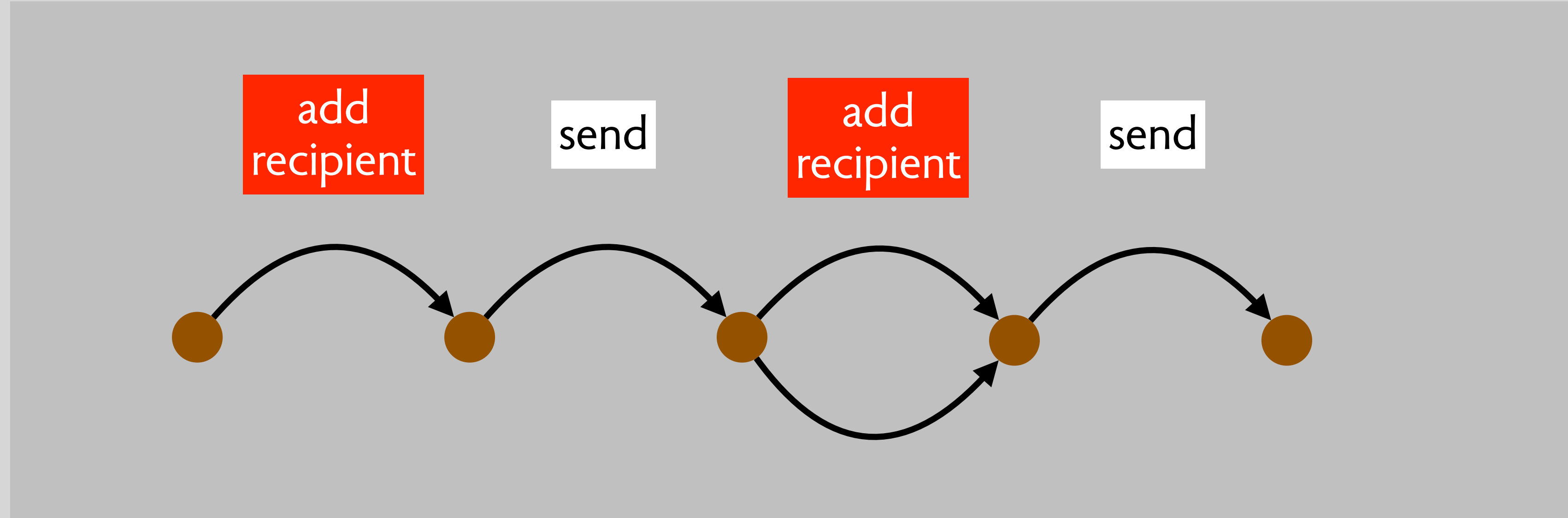
Beluga



Yobongo



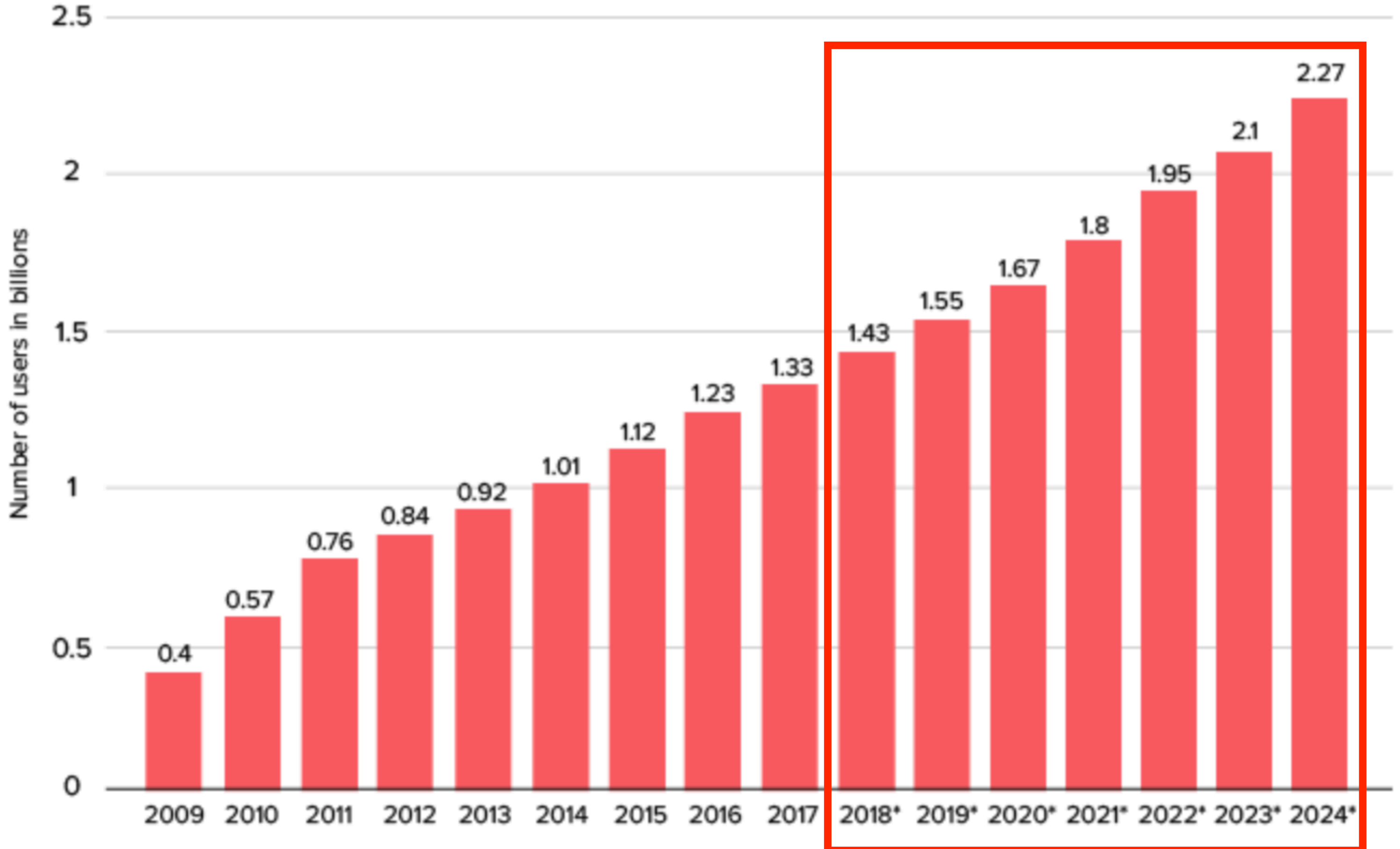
# the group scenario



story #4

Zoom (2013)

### Number of estimated Skype users registered worldwide from 2009 to 2024 (in billions)

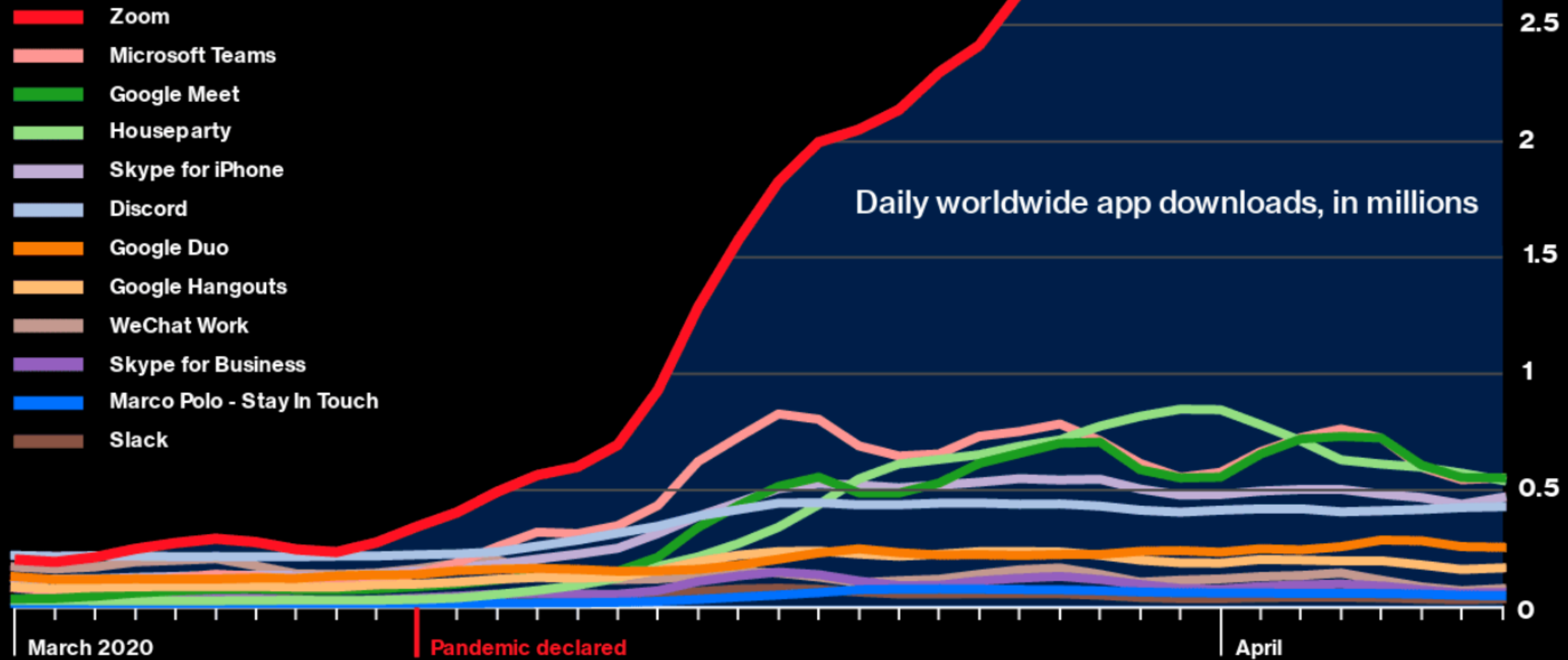


from Statistica: 2018-2024 estimated

# Covid-19 / Zoom-a-Zoom-Zoom

BBOViz  
Covid-19

There's only one winner in the  
work-from-home meeting app wars



Source: Apptopia

BEN SCHOTT

# what was novel about Zoom?



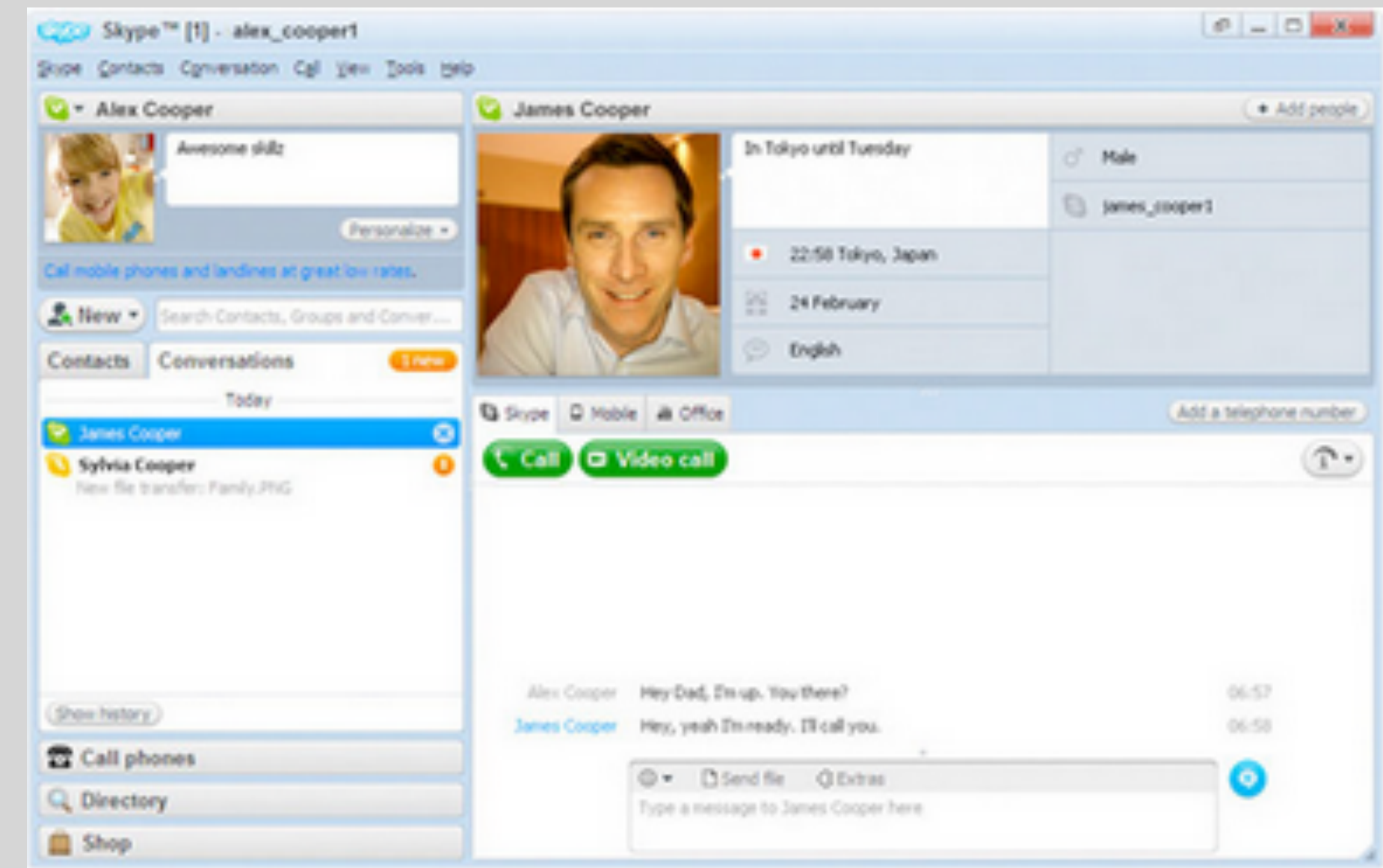
shares meeting id!

March 31, 2020

# were video calls new?



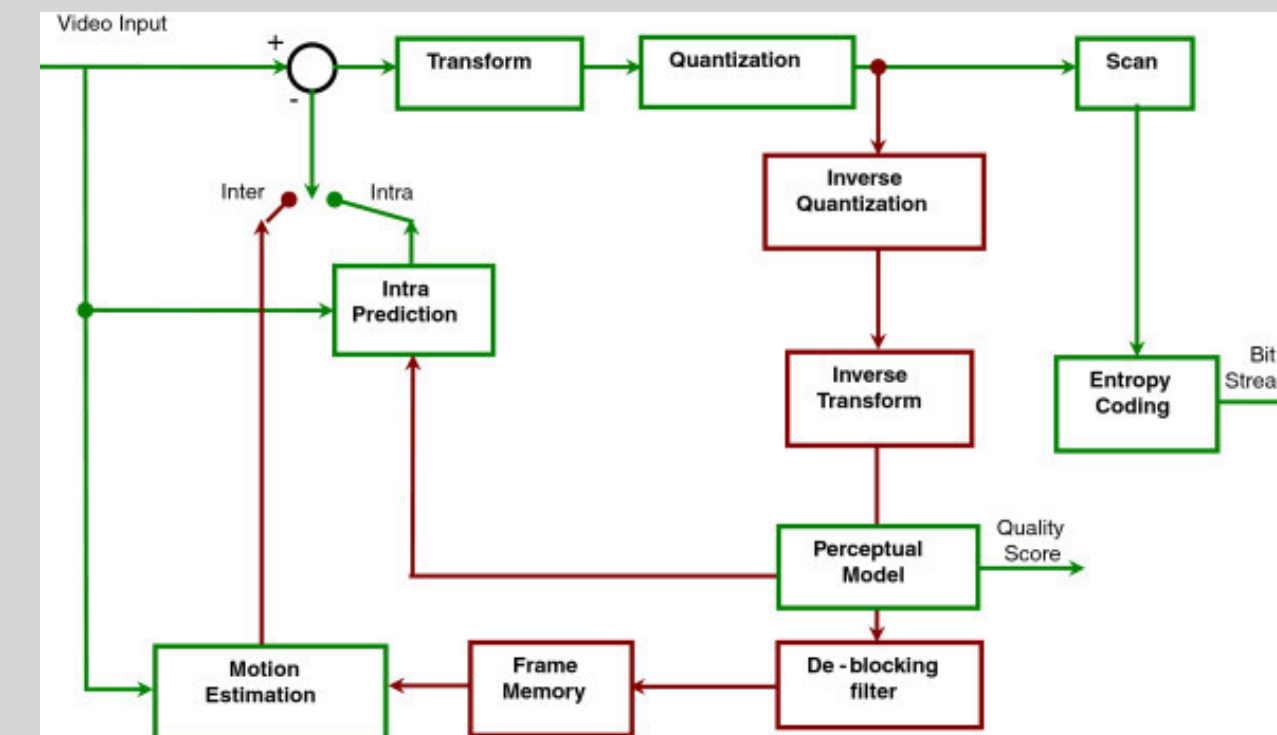
Picturephone (1964)



Skype (2009)

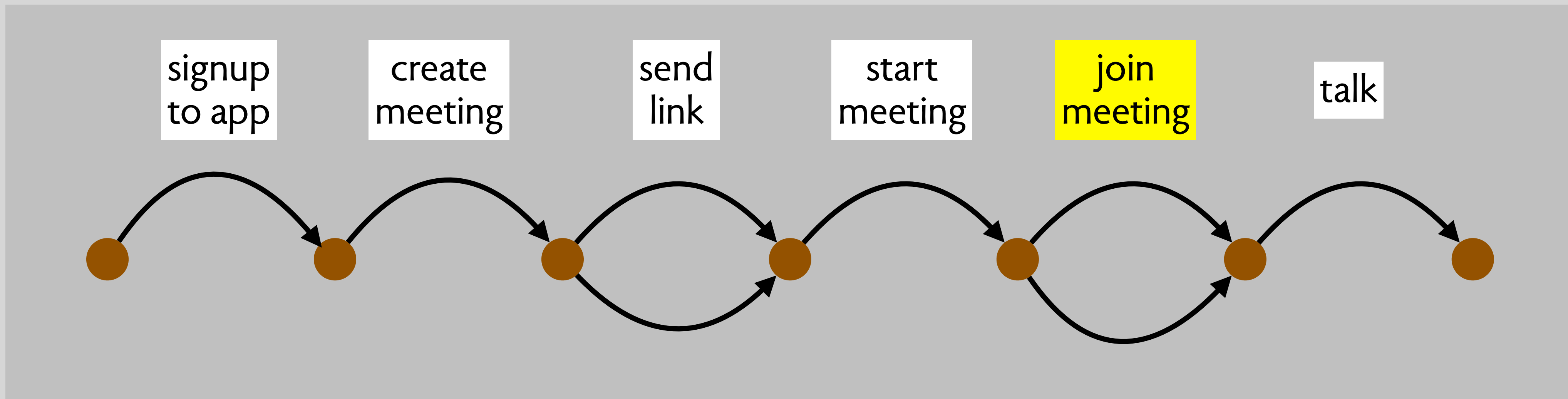
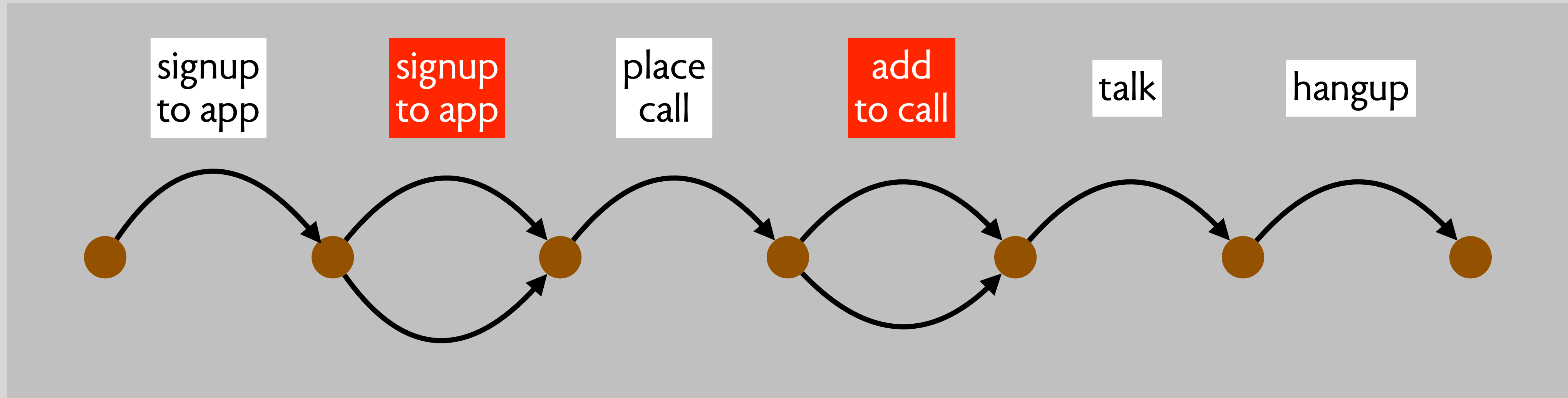


QuickCam (1994)  
first commercial webcam



H.264 Video Codec (2003)

# the meeting scenario



# tracing zoom's meeting concept



**Skype**  
initially P2P (2003)  
Microsoft (2013)



**FaceTime**  
Apple (2010)



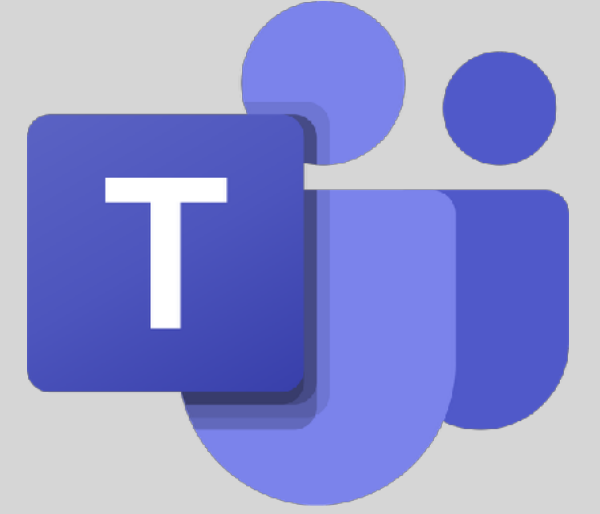
**Google Hangouts**  
in Google+ (2011)  
own product (2013)  
Duo replaces (2016)



**Zoom**  
Eric Yuan (2013)

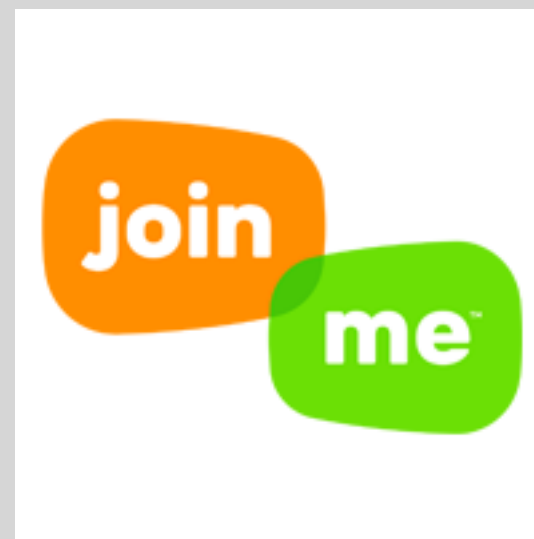


**Google Meet**  
launched (2017)  
absorbs Duo (2022)



**Microsoft Teams**  
launched (2017)

meeting concept  
added to Skype  
April 2020



**Join.me**  
LogMeIn (2010)

meeting scenario

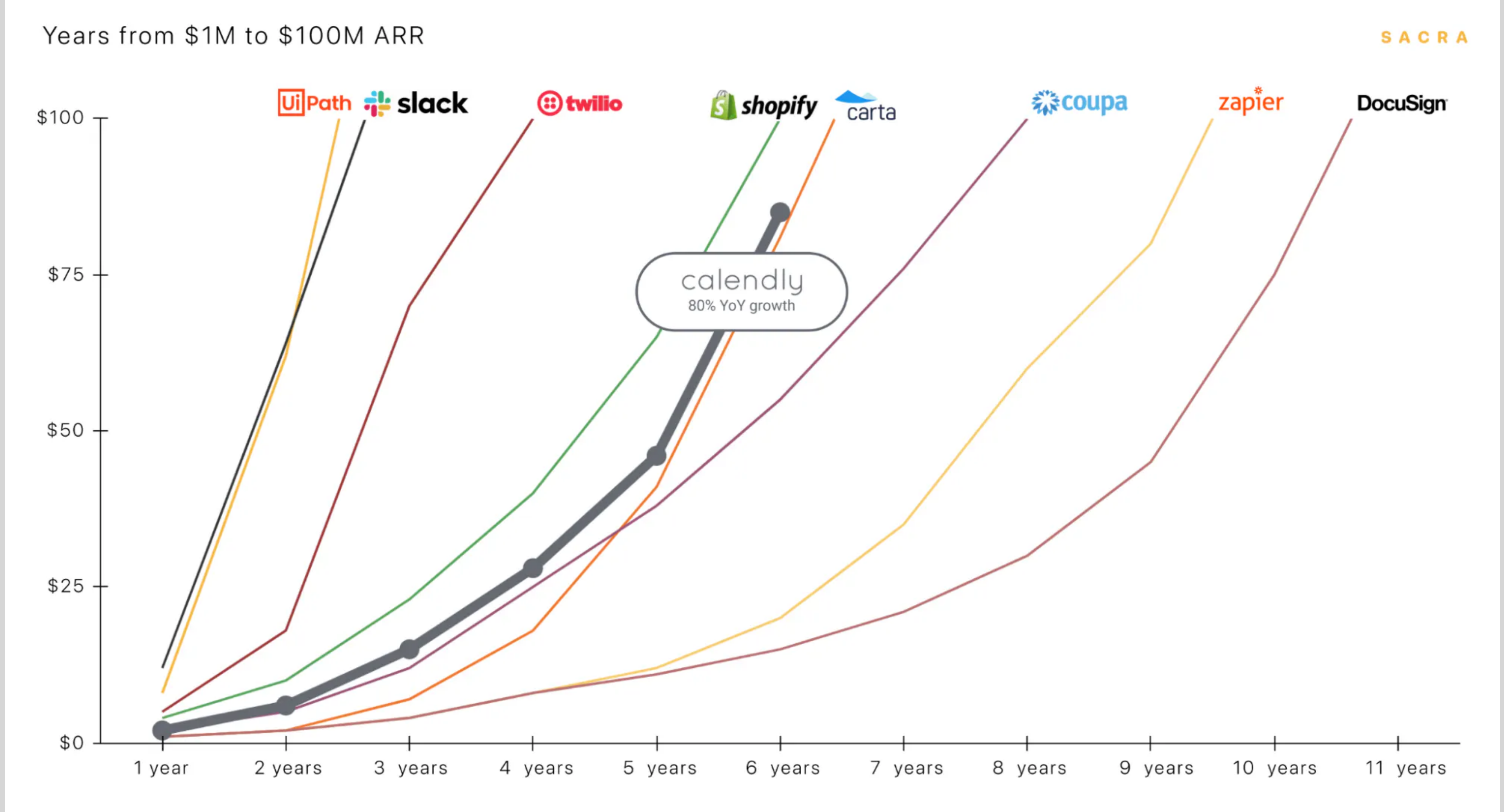
meeting scenario  
added to Teams  
June 2022?




story #5

Calendly (2013)


# "the \$4b Docusign of scheduling"




# scheduling yourself



## 30 Minute Meeting

 30 min


 Web conferencing details provided upon confirmation.


[Cookie settings](#)

**Select a Date & Time**

May 2022 < >

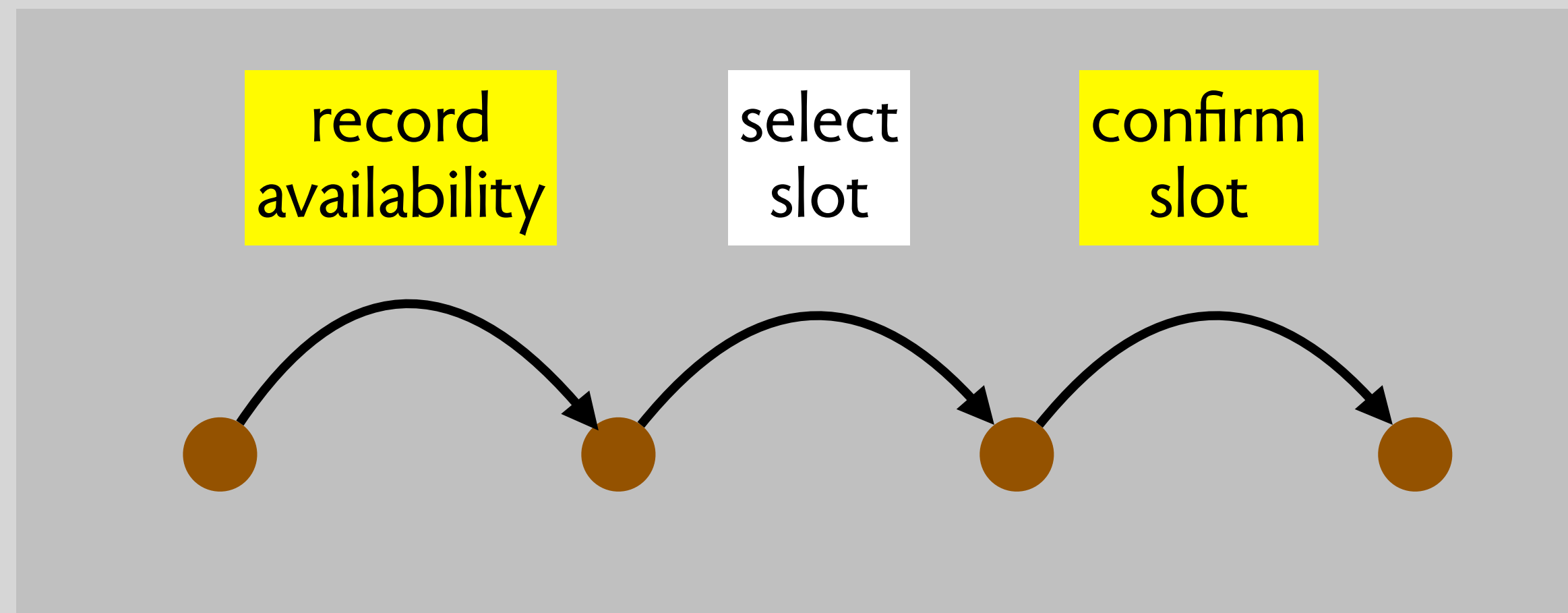
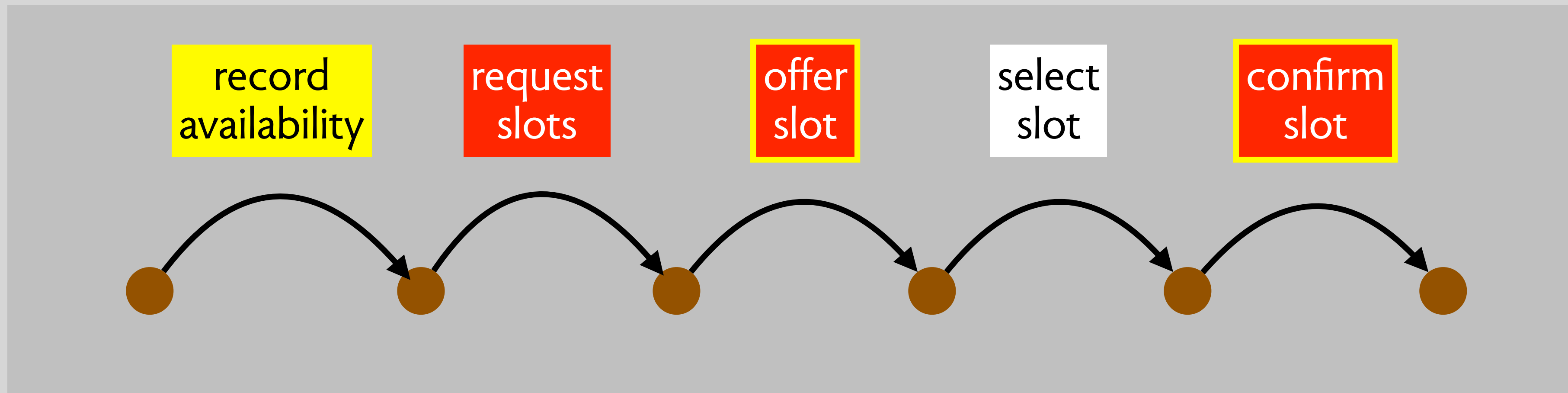
SUN	MON	TUE	WED	THU	FRI	SAT
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

 Pacific Time - US & Canada (4:24pm) ▼

 [Troubleshoot](#)

POWERED BY  
**Calendly**

# fixing the scenario: scheduling a slot



# tracing self scheduling



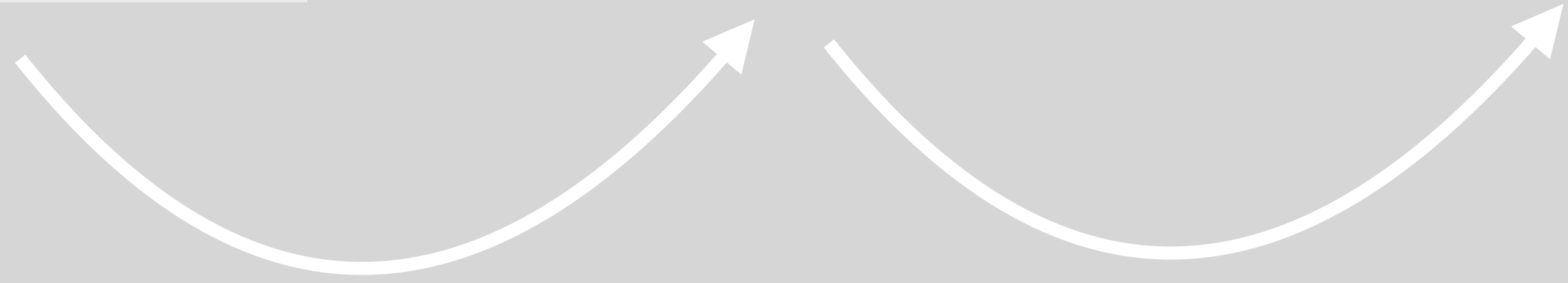
**Acuity (2006)**  
small business focus  
to Squarespace (2018)



**Calendly (2013)**  
sole proprietor focus



**Google Calendar**  
"slots" (2011)  
"schedules" (2022?)



self-scheduling scenario

where we  
are so far

# lessons so far

## **a product is defined by a scenario**

a compelling story of how to use it  
a social protocol and a service API at once  
hints at why the user's purpose is fulfilled  
an archetypal usage, not the only one

## **innovations (almost) never enable new things**

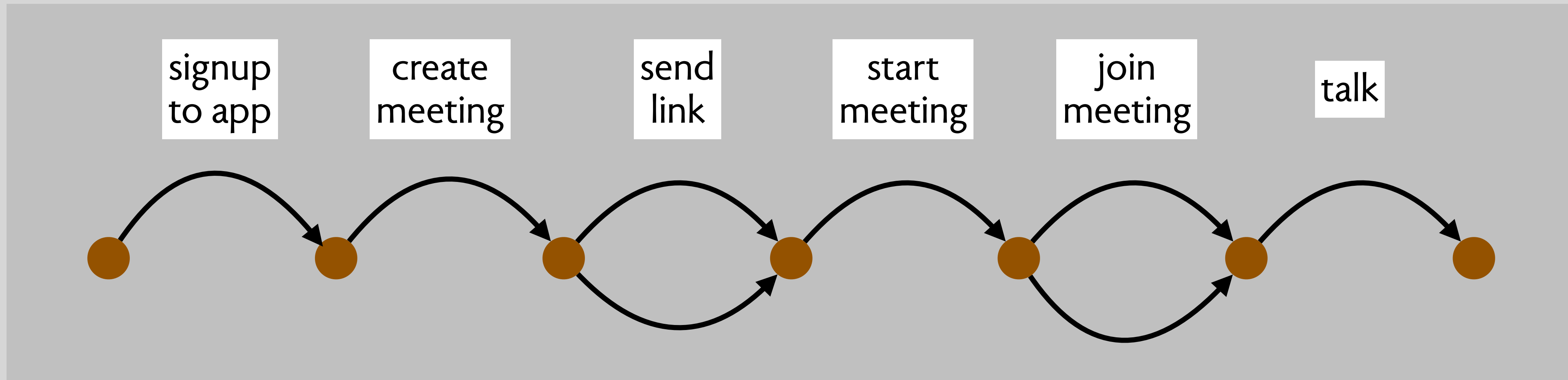
they just make them easier to do  
an old scenario with pain points is replaced by new one

scenarios for  
design, not  
specification!

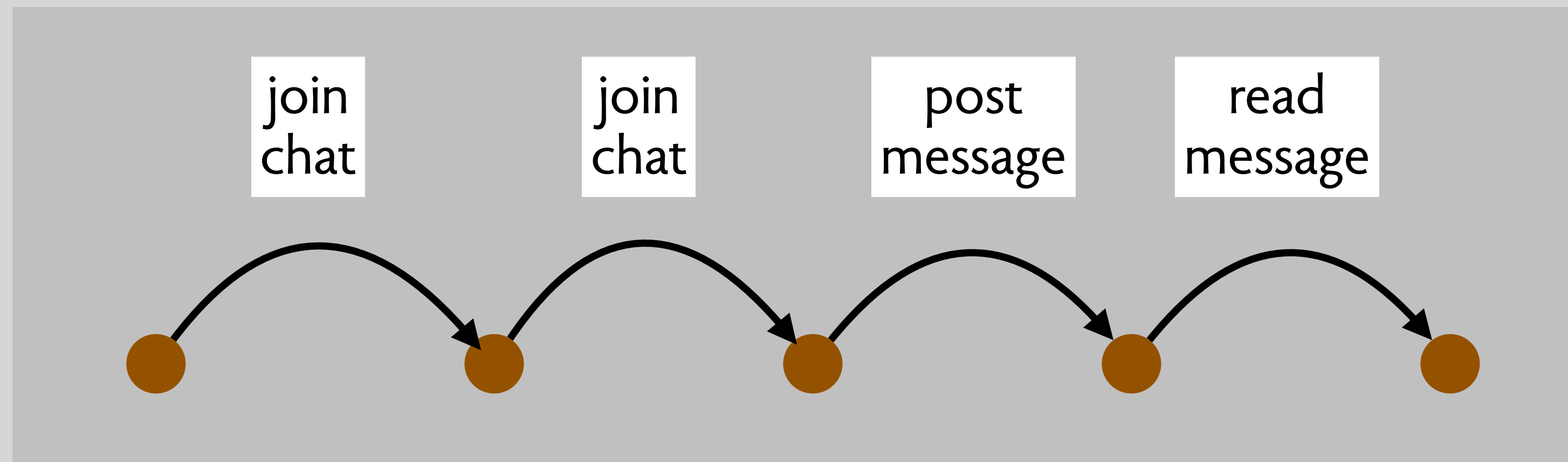
*from scenarios  
to concepts*



# zoom isn't just the meeting scenario



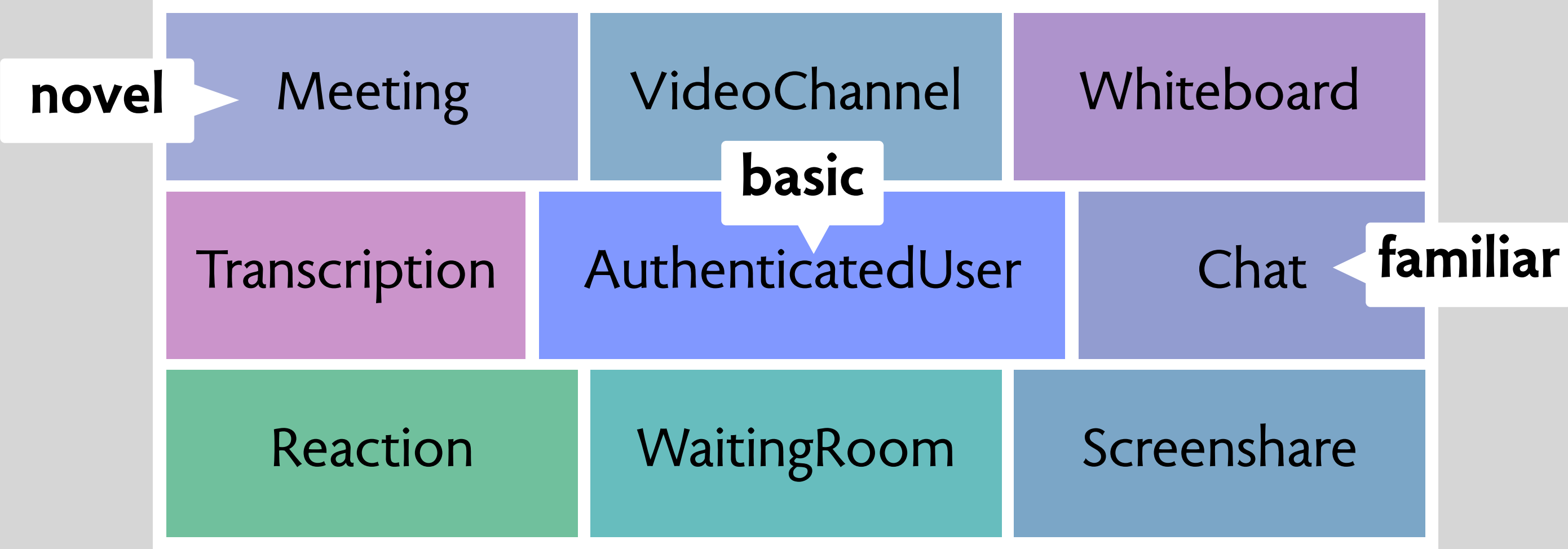
**Zoom's meeting scenario**



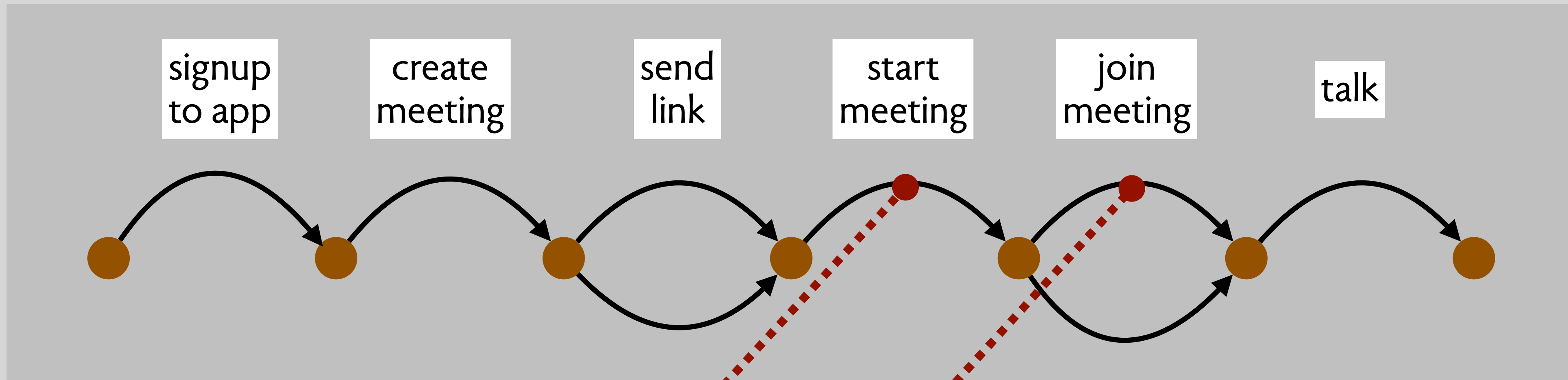
**Zoom's chat scenario**

# software = concepts

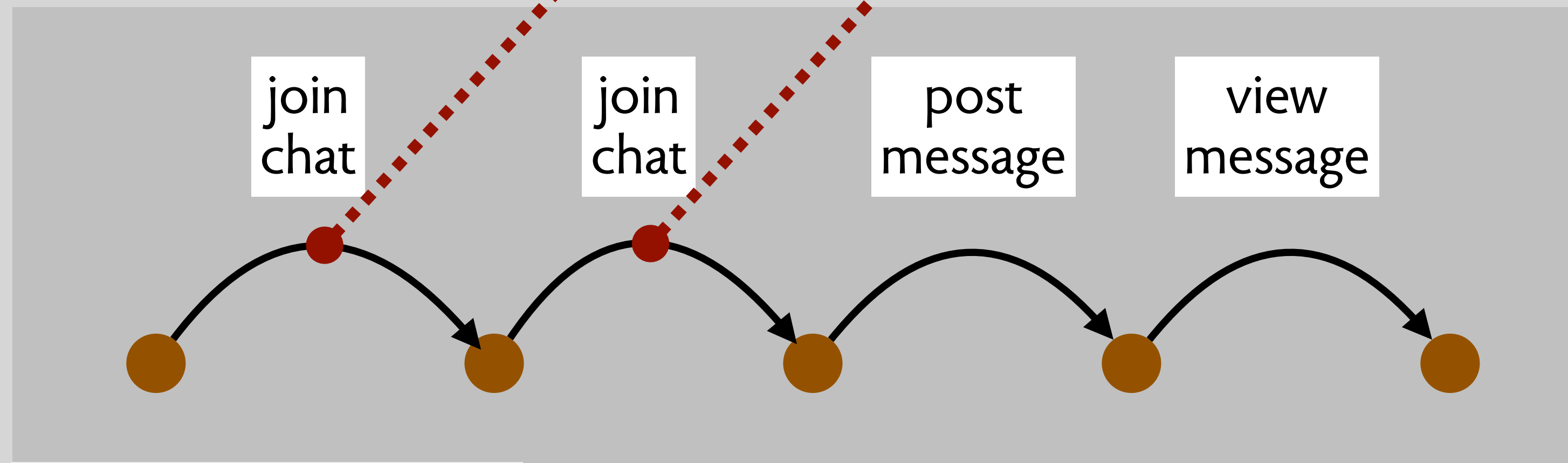
## Zoom



# how are concepts composed? by interleaving & synchronization

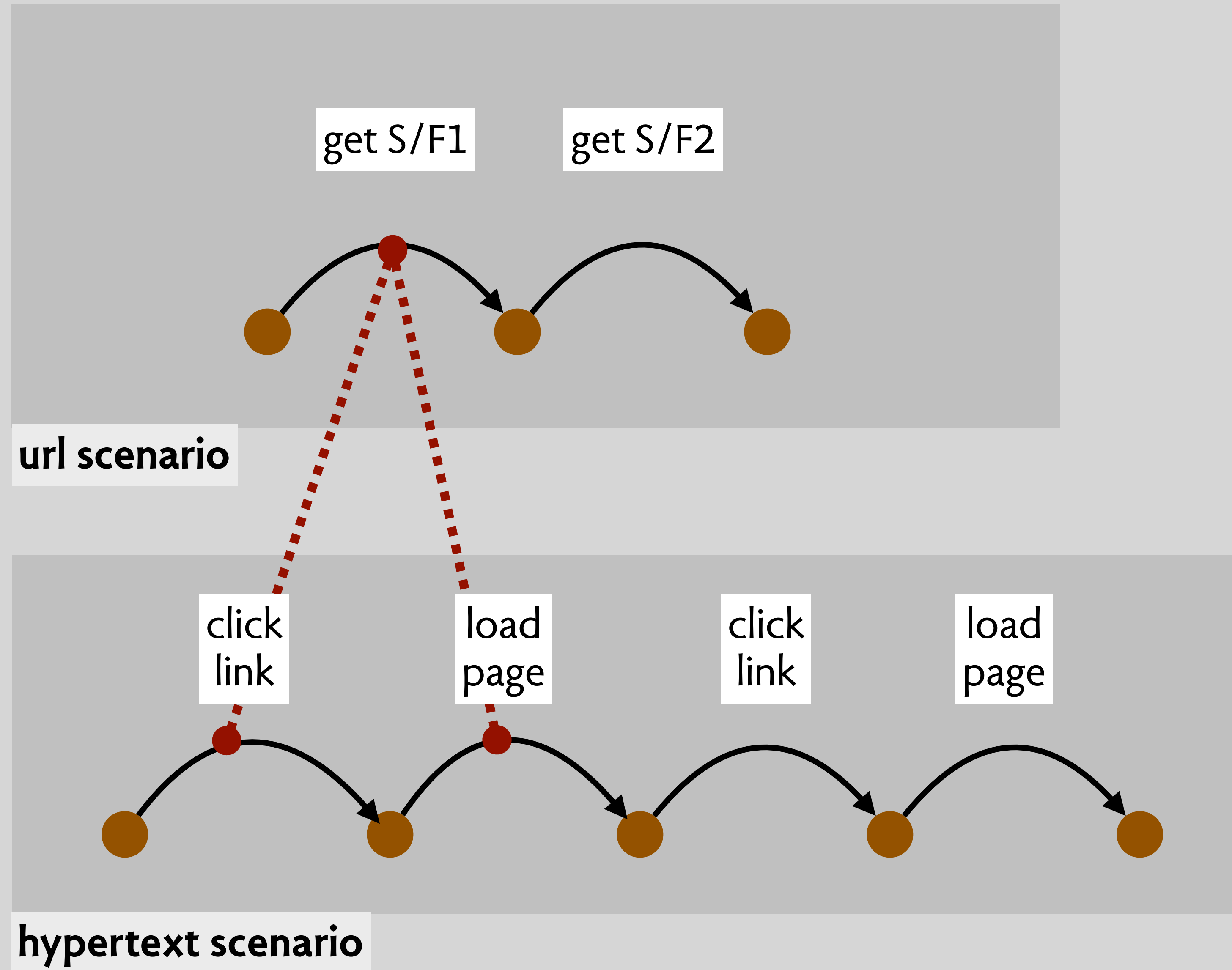


**Zoom's meeting scenario**



**Zoom's chat scenario**

# the synchronization that makes URLs work



# essential concept compositions

Zoom = MeetingLink + VideoConference

Calendly = SelfSchedule + Notify

iPod = Song + Store + Sync

# designing a concept

**concept** Chat

**purpose** share short messages in group

**principle** after two users join, if one posts a message, the other can read it

## state

msgs: set Msg

content: Msg -> one String

posted: Msg -> one Date

joined: User -> one Date

## actions

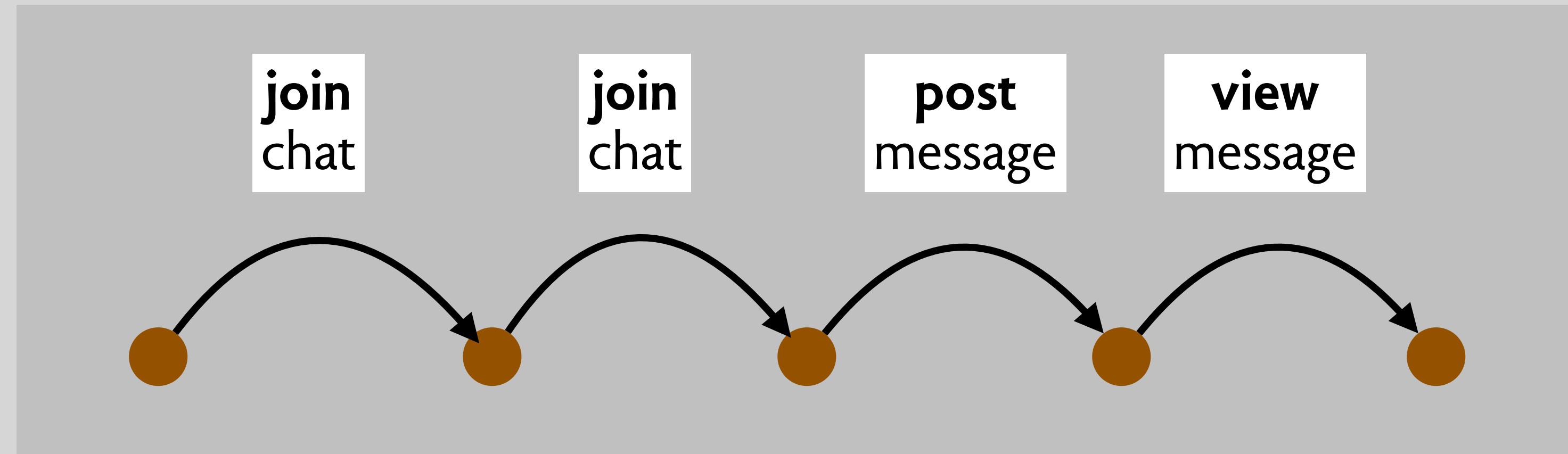
join (u: User)

post (u: User, s: String, out m: Msg)

view (u: User, out msgs: set Msg)

// returns all messages posted after

// u's joining date



actually a serious design flaw

# how many “differentiator” concepts?



Gmail: email + search



Zoom: meeting



Photoshop: pixel array,  
layer, mask, channel, ...



Arc: tabs as bookmarks



WhatsApp: group



Quark: flow, kerning,  
bezier curves, ...

**no new concepts**  
combinational creativity

**one new concept**  
a better way to work

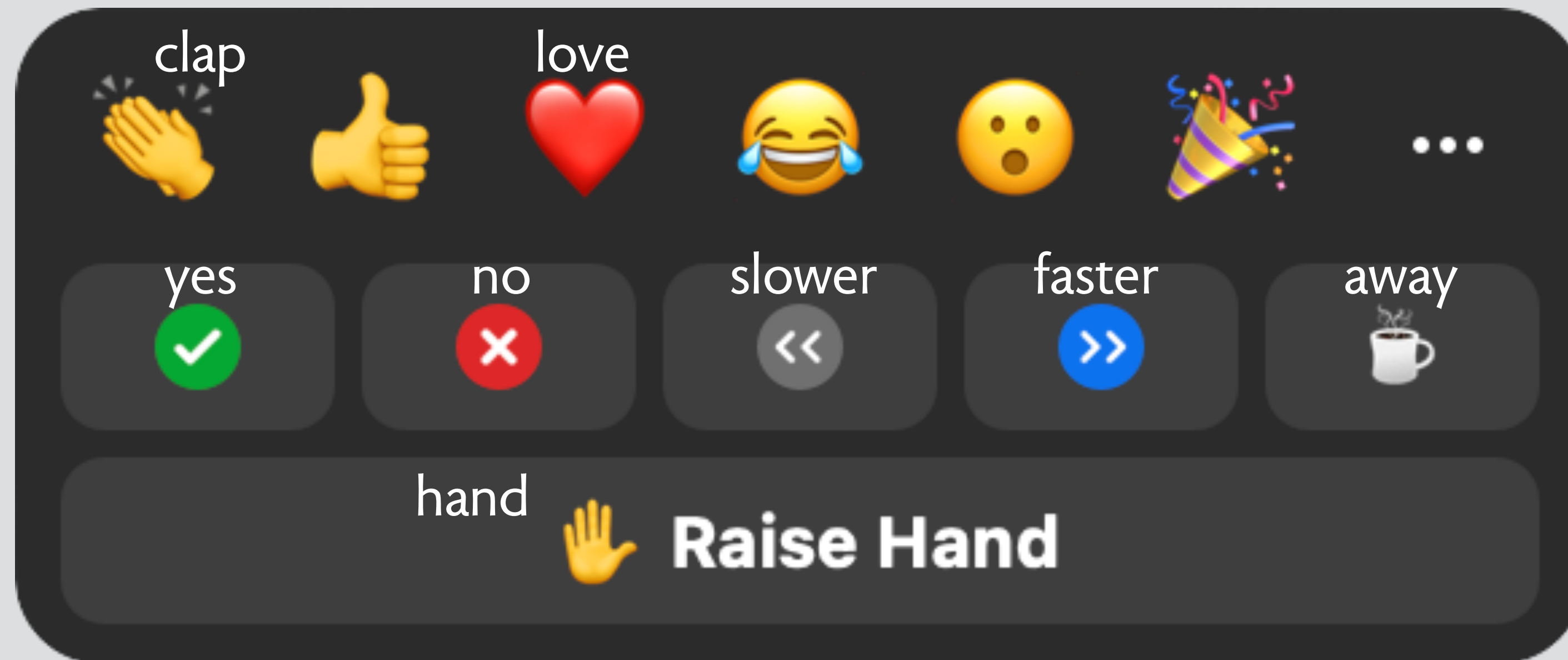
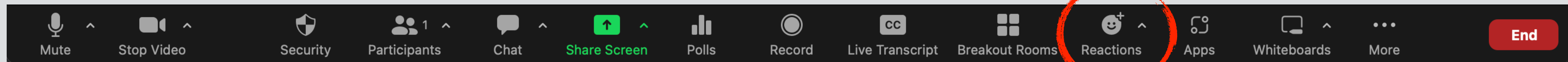
**many new concepts**  
a new paradigm

how can you  
use these ideas?

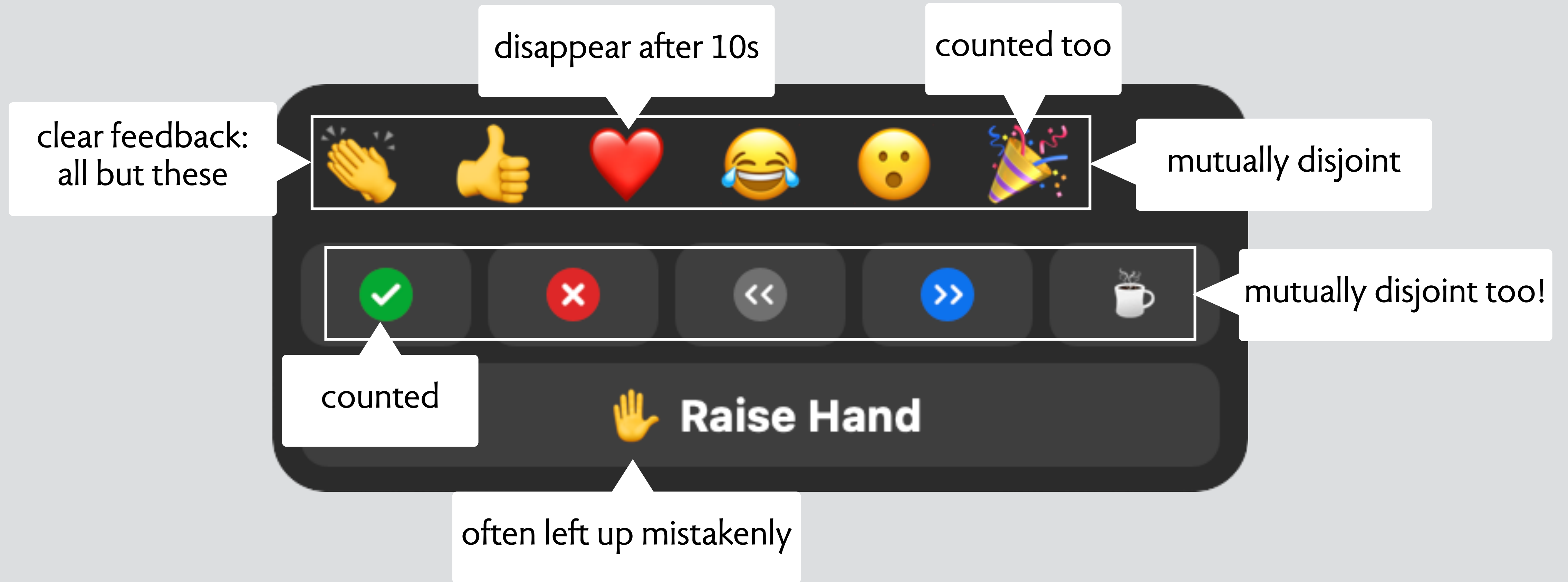


design  
by concept

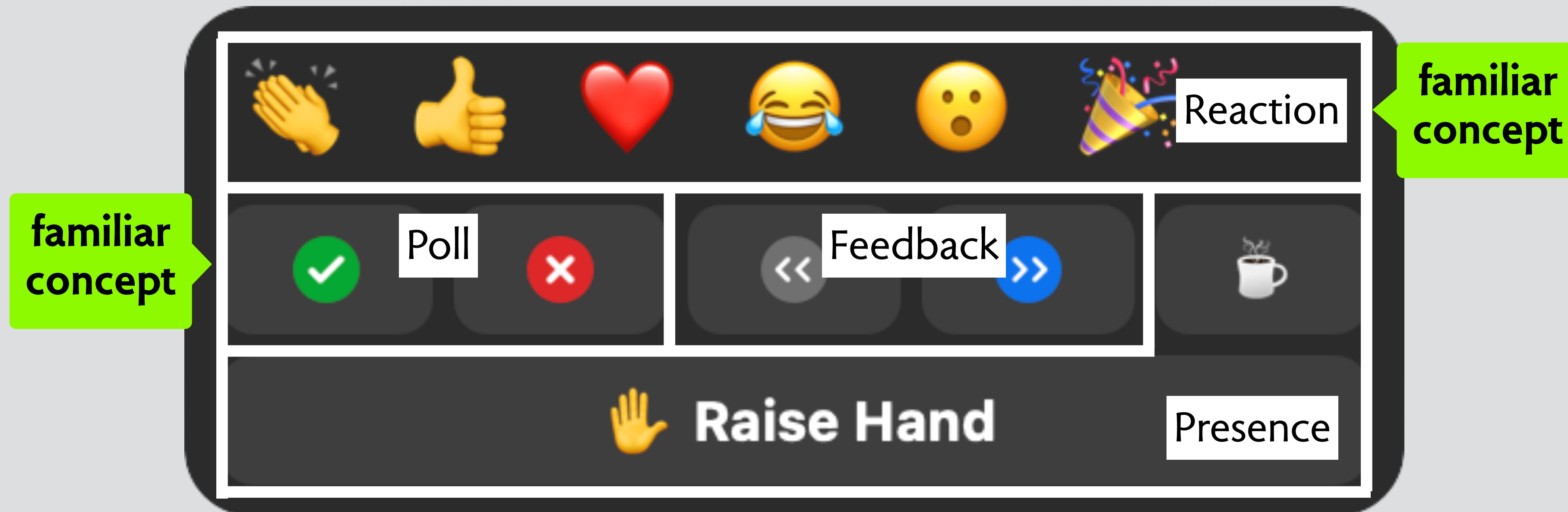
# Zoom's reactions

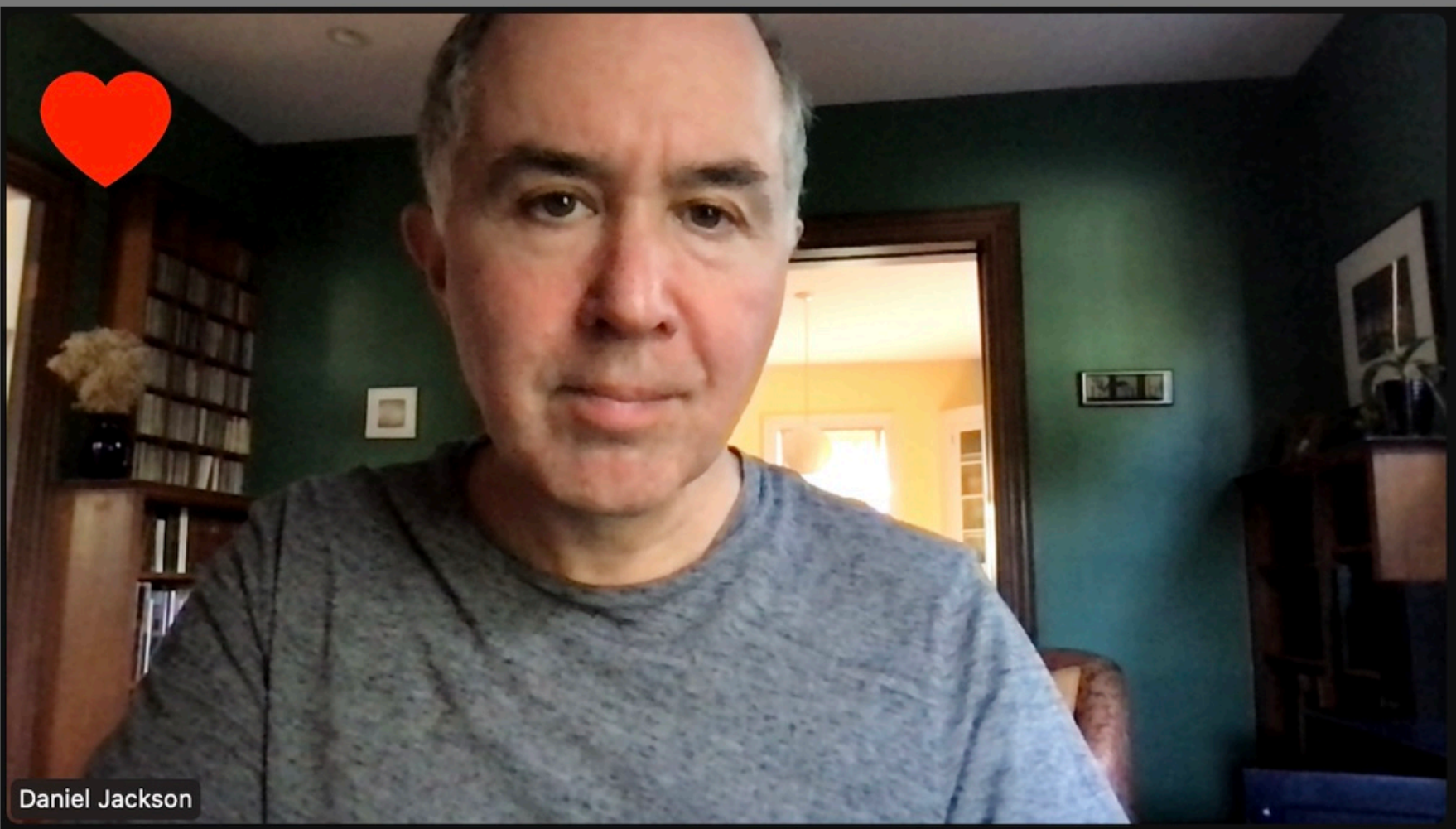


# anomalous behaviors



# splitting into coherent concepts





**Presence**

**Chat**

**Reaction**

**Feedback**

Request to speak

Watching/listening

Speaking

I'm away

Audience

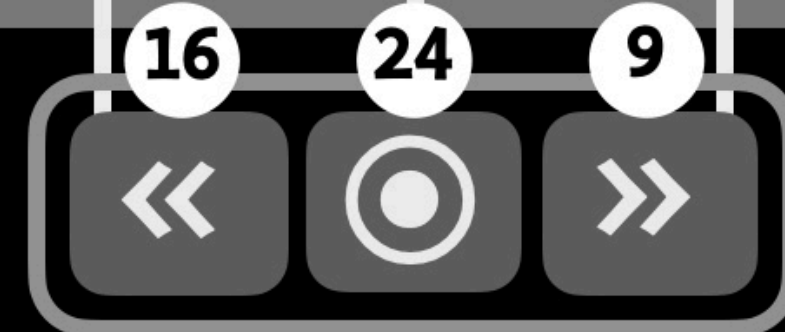
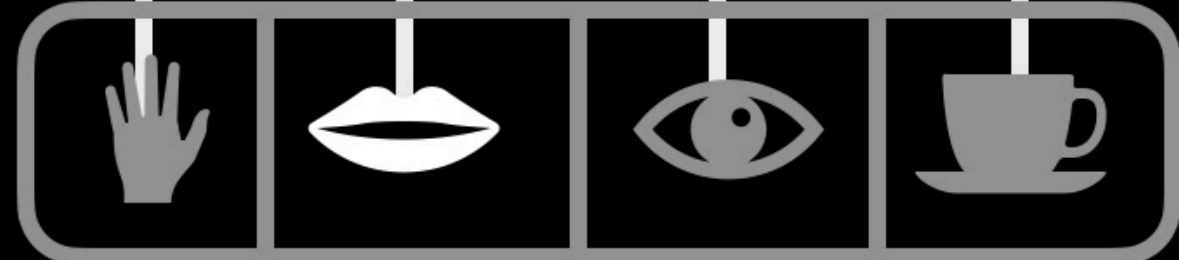
Recent emoji

Other emoji

Slow down

Just right

Speed up



**Reaction**

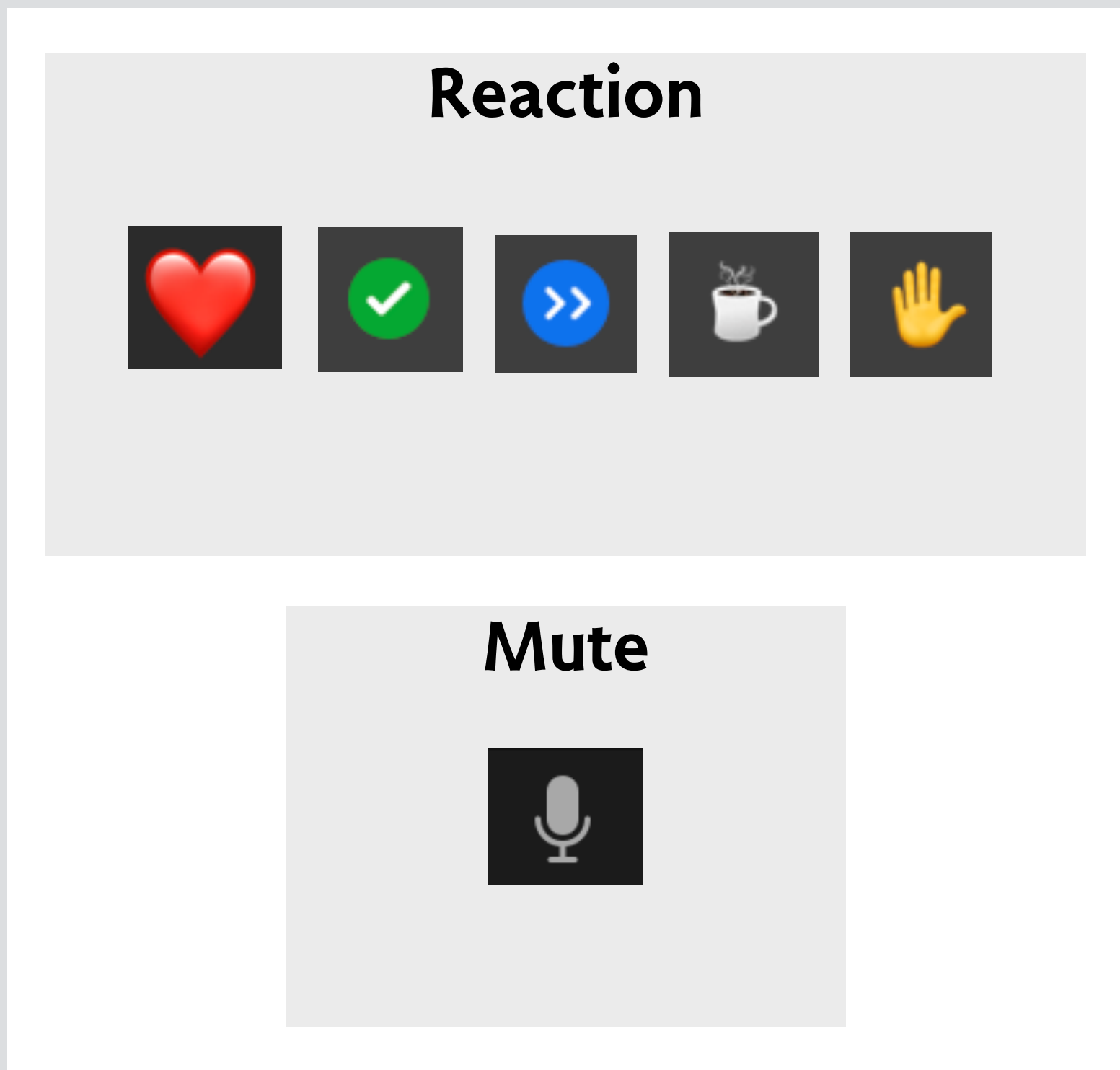
Reaction

Reaction

Reaction

Mute

Mute



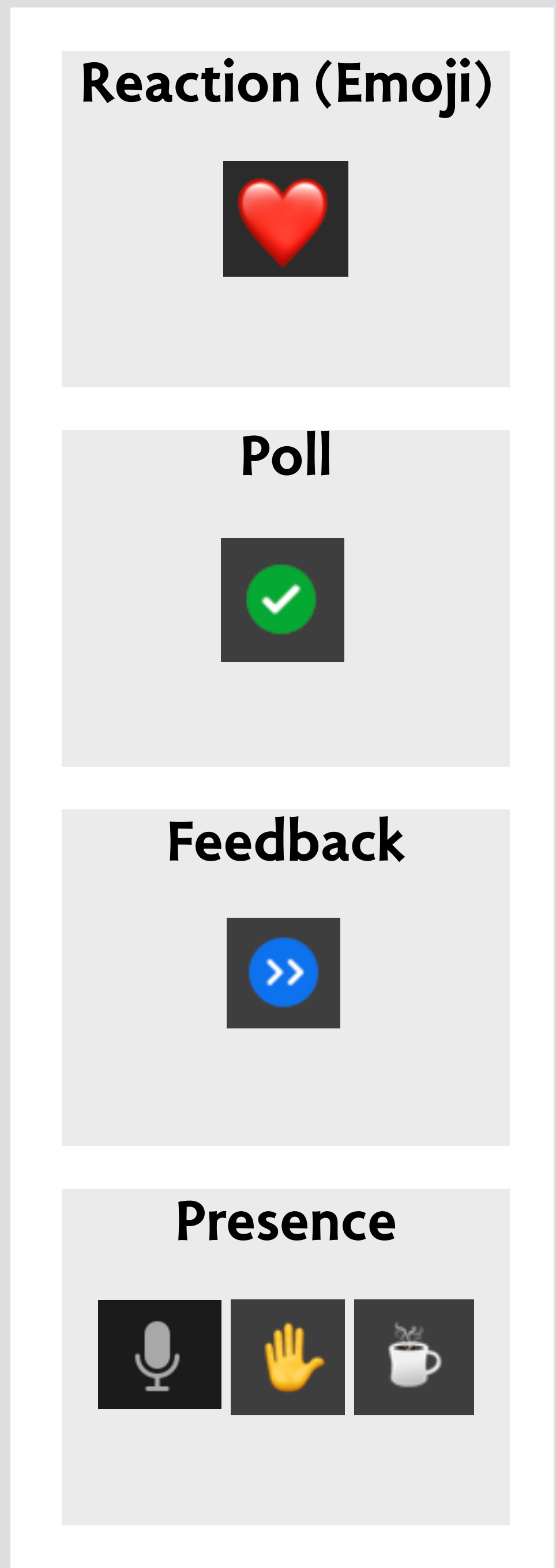
**Reaction (Emoji)**

Reaction (Emoji)

Poll

Feedback

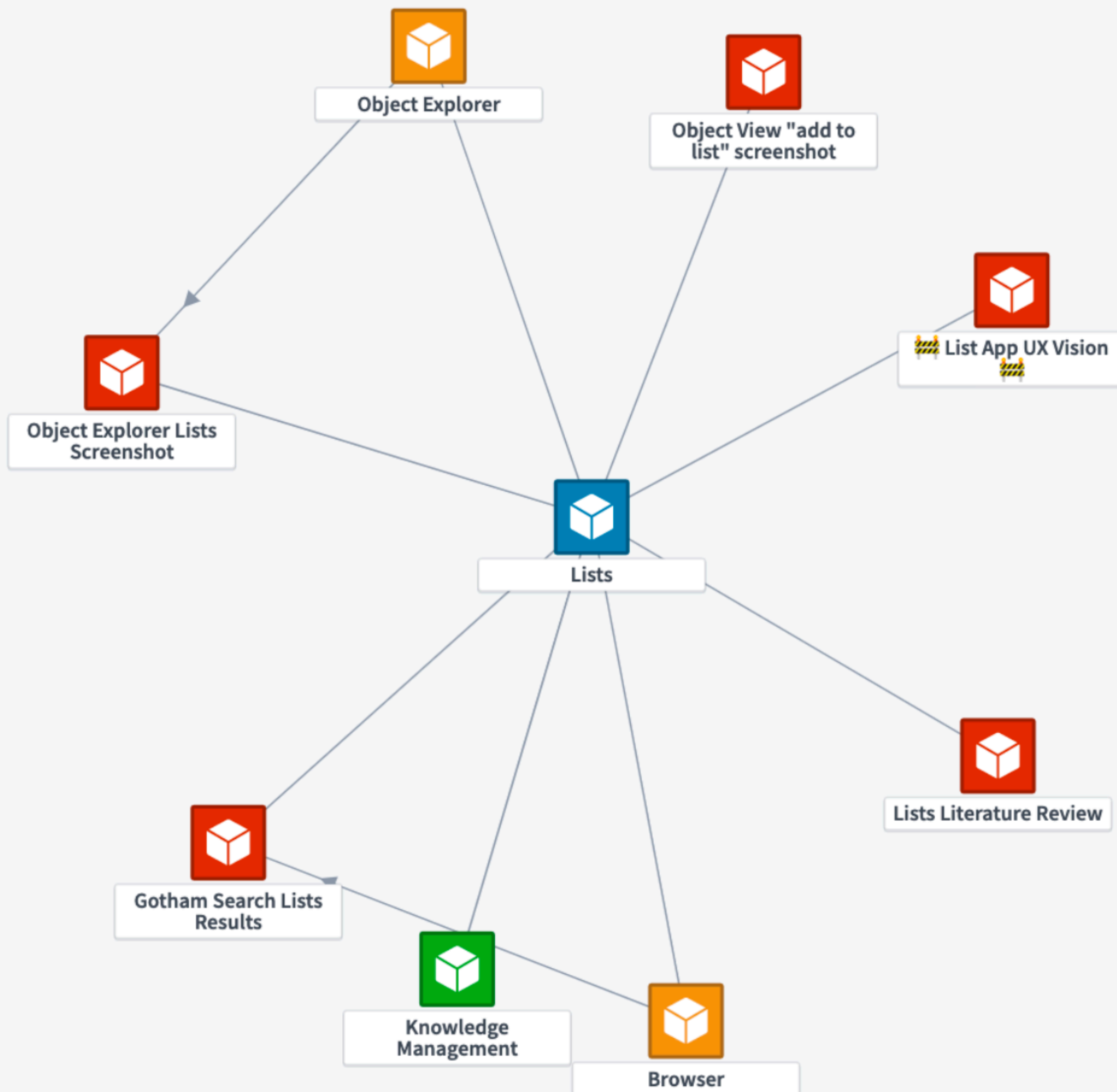
Presence



align products  
in suite

# concepts at Palantir (2023)

Wilczynski et al, [arxiv.org/abs/2304.14975](https://arxiv.org/abs/2304.14975)



## challenges they were facing

issues not attributable to modules or even products  
inconsistent UX across products for similar functions  
“conceptual entropy”: growing complexity

## what they did

integrated concepts into company knowledge base  
leaders bootstrapped by writing initial concepts  
exploiting existing documents  
now 200 concepts recorded, 280 regular users

## concepts go beyond engineering

concepts used in marketing; IP lawyers interested too

## concepts empower PMs

new career path: PMs given ownership of concepts

## anticipated impacts

cataloging key assets & avoiding rework  
aligning concepts across products, reuse  
aligning marketing/design/engineering



# modularize your code

*with Abutalib Namazov*

▲ Jackson structured programming (wikipedia.org)

106 points by haakonhr 63 days ago | hide | past | favorite | 69 comments

Post

Session

Upvote

Favorite

▲ danielnicholas 63 days ago [-]

user: danielnicholas

created: 63 days ago

karma: 11

Comment

Karma

you might find helpful an annotated version [0] of Hoare's explanation of JSP that I edited for a Michael Jackson festschrift

I'd point to these ideas as worth knowing:

...ing problem that involves traversing structures can be solved very systematically. HTDP addresses this class, but bases code structure only on input structure; JSP synthesized it.

- The archetypal problems that, however you code, can't be pushed under the rug—most notably structure clashes—and just recognizing them

- Coroutines (or code transformation) let you structure code more cleanly when you need to read or write more than one structure. It's why real iterators (with yield), which offer a limited form of this, are (in my view) better than Java-style iterators with a next method.

- The idea of viewing a system as a collection of asynchronous processes (Ch. 11 in the JSP book, which later became JSD) with a long-running process for each real-world entity. This was a notable contrast to OOP, and led to a strategy (seeing a resurgence with event storming for DDD) that began with events rather than objects.

[0] <https://groups.csail.mit.edu/sdg/pubs/2009/hoare-jsp-3-29-09...>

▲ ob-nix 63 days ago [-]

... this brings back memories! In the late eighties I, as a teenager, found a Jackson Struct. Pr. book at the town library. I remember I was amazed at the text and wondered why I hadn't heard about the method before.

If I remember correctly did the book clearly point out backtracking as a standard method, while mentioning that most languages lacked that, so it had to be implemented manually.

▲ CraigJPerry 63 days ago [-]

This is referenced(1) as a core inspiration in the preface to "How to Design Programs" but i never researched it further because i've found the "design recipes" approach in htdp to be pretty solid in real life problems

# “combinational creativity”

## **post concept**

just a title and a URL

home page list shows posts at most 4 days old

## **comment concept**

can delete within 2 hours of writing

no new comments after 2 weeks

## **upvote concept**

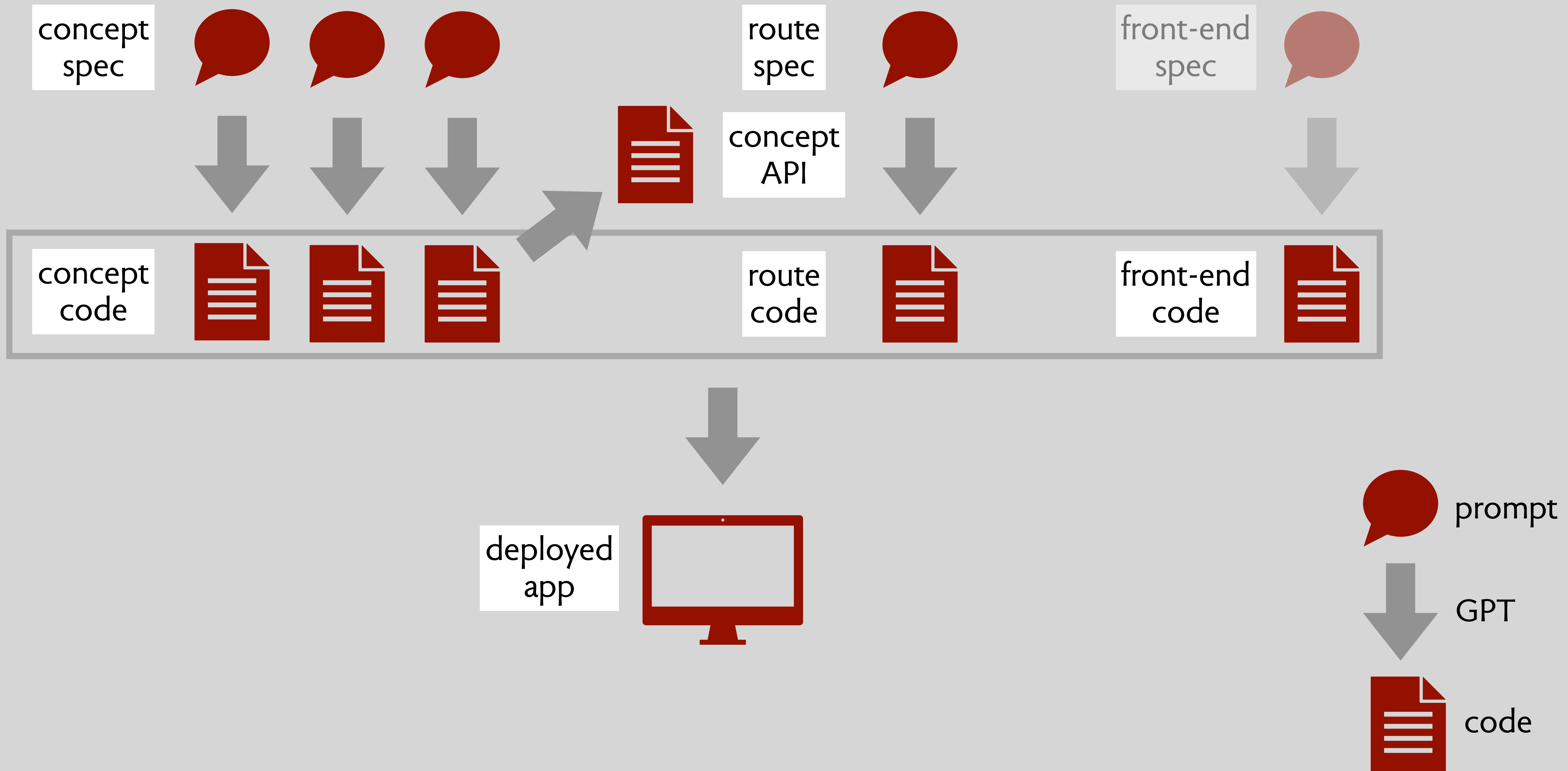
can only downvote with enough karma

## **ask and show**

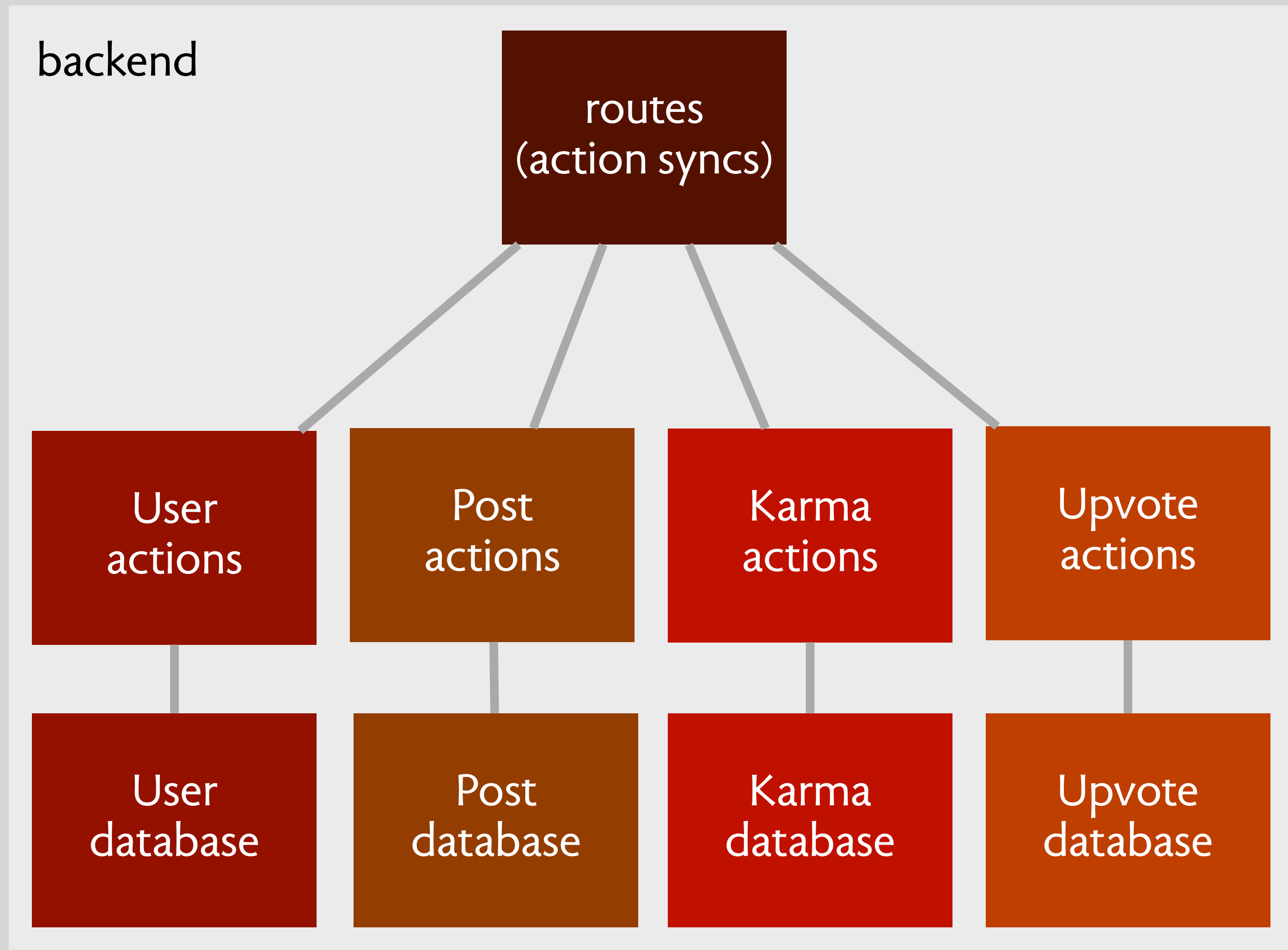
have “Ask HN” and “Show HN” prefix in title

...

# our approach




# structuring the code with concepts




no dependencies between concepts!

## Project: empty\_project

 Dependencies not installed  
node\_modules directory does not exist. They are needed to run your backend.

Install dependencies

Delete the project


 Project Status  
Project is stopped.


Start running

> Environment Variables

> File Directory


### Concepts

 Create a new concept

 user.ts

 websession.ts

### App Definition


 Edit the app definition

built-in  
concepts

# Project: hackernews

Delete dependencies

Delete the project


 Project Status  
Project is stopped.

Start running

> Environment Variables

> File Directory

## Concepts


 Create a new concept


 comment.ts

 karma.ts

 mark.ts

 post.ts

 user.ts


 vote.ts


 websession.ts


## App Definition

user adds concepts

## Concepts

 Create a new concept

 comment.ts

 karma.ts

You can update the concept here. Currently, renaming a concept is not supported.

Revision prompt

Please Input

Update

Prompt history:

Each karma should have user (generic) and points (number). Actions should be:

- increase(user, x)
- decrease(user, x)
- get(user),
- isAllowed(user, threshold), throws error if user doesn't have at least threshold

defining  
Karma



generated  
concept  
code

Source code:

```
1 import { ObjectId } from "mongodb";
2 import DocCollection, { BaseDoc } from "../framework/doc";
3 import { NotAllowedError } from "../framework/errors";
4
5 interface KarmaDoc extends BaseDoc {
6   user: ObjectId;
7   points: number;
8 }
9
10 export default class KarmaConcept {
11   public readonly karmas: DocCollection<KarmaDoc>;
12
13   constructor(collectionName: string) {
14     this.karmas = new DocCollection<KarmaDoc>(collectionName);
15   }
16
17   async increase(user: ObjectId, x: number) {
18     const karma = await this.karmas.readOne({ user });
19     if (karma) {
20       await this.karmas.collection.updateOne({ user }, { $inc: { points: x } });
21     } else {
22       await this.karmas.createOne({ user, points: x });
23     }
24   }
25
26   async decrease(user: ObjectId, x: number) {
27     const karma = await this.karmas.readOne({ user });
28     if (karma) {
29       await this.karmas.collection.updateOne({ user }, { $inc: { points: -x } });
30     } else {
```

generated  
concept spec

Generated spec:

```
KarmaDoc = {  
  user: ObjectId;  
  points: number;  
}  
  
KarmaConcept = {  
  constructor: (collectionName: string) => KarmaConcept; // instantiates a new KarmaConcept with the  
given collection name  
  async increase: (user: ObjectId, x: number) => void; // increases the karma points for a user; cre  
ates a new karma record if not exists  
  async decrease: (user: ObjectId, x: number) => void; // decreases the karma points for a user; thr  
ows NotAllowedError if resulting in negative karma  
  async get: (user: ObjectId) => number; // returns the karma points for a user; returns 0 if the us  
er has no karma record  
  async isAllowed: (user: ObjectId, threshold: number) => void; // throws NotAllowedError if user's  
karma points do not meet the threshold  
}
```

generated  
route code

✓ createUser (Register a new user and log in)

Delete

```
1 @Router.post("/register")
2   async createUser(session: WebSessionDoc, username: string, password: string) {
3     WebSession.assertLoggedOut(session);
4     const { msg, user } = await User.create(username, password);
5     await Karma.increase(user._id, 1);
6     WebSession.start(session, user._id);
7     return { msg };
8   }
```

# hacked news: home page

1. ▲ Show HN: Kodless -- make websites with AI (github.com)  
1 point by dnj a day ago | [hide](#) | [discuss](#)
2. ▲ Monolith -- CLI tool for saving web pages as a single HTML file (crates.io)  
1 point by rust 9 days ago | [hide](#) | [discuss](#)
3. ▲ Show HN: Kodless -- make websites with AI (github.com)  
3 points by kodless 9 days ago | [hide](#) | [2 comments](#)
4. ▲ Software = concepts (essenceofsoftware.com)  
6 points by dnj 9 days ago | [hide](#) | [2 comments](#)
5. ▲ Jobs HN: I am hiring a wine expert  
1 point by recruiter 10 days ago | [hide](#) | [discuss](#)
6. ▲ Show HN: Fuiz -- free, open-source and privacy-friendly alternative to Kahoot (fuiz.us)  
3 points by best\_dev 10 days ago | [hide](#) | [7 comments](#)
7. ▲ Ask HN: How many bugs do you have per line of code?  
1 point by big\_asker 10 days ago | [hide](#) | [discuss](#)
8. ▲ Github -- Use this website to share your code (github.com)  
2 points by barish2 10 days ago | [hide](#) | [discuss](#)

# hacked news: thread

Software = concepts (essenceofsoftware.com)

6 points by dnj 9 days ago | [unvote](#) | [hide](#) | [favorite](#) | [2 comments](#)

[add comment](#)

\* [kodless](#) 9 days ago | [next \[-\]](#)

dnj, you should check out my platform Kodless -- it helps you generate software with concepts without writing any code.

[reply](#)

dnj 9 days ago | [next \[-\]](#) [unvote](#)

That's great to see! Have you seen GPT-powered concept tutor? <https://essenceofsoftware.com/studies/larger/tutor/>

[reply](#)

Search:

# hacked news: user profile

user: kodless

created: 2024-03-25T19:14:36.316Z

karma: 3

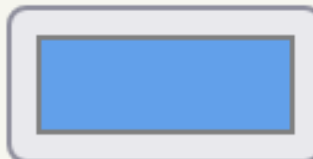
about:

[submissions](#)

[threads](#)

[favorites](#)

[hidden](#)



change top bar color

Search:

# hacked news: posts by date

[previous day](#) | [next day](#) | 2024-03-25

1. [Software = concepts](#) (essenceofsoftware.com)  
6 points by dnj 9 days ago | [unvote](#) | [hide](#) | [2 comments](#)
2. \* [Show HN: Kodless -- make websites with AI](#) (github.com)  
3 points by kodless 9 days ago | [hide](#) | [2 comments](#)
3. [Show HN: Fuiz -- free, open-source and privacy-friendly alternative to Kahoot](#) (fuiz.us)  
3 points by best\_dev 10 days ago | [unvote](#) | [hide](#) | [7 comments](#)
4. [Github -- Use this website to share your code](#) (github.com)  
2 points by barish2 10 days ago | [unvote](#) | [hide](#) | [discuss](#)
5. ▲ [Ask HN: How many bugs do you have per line of code?](#)  
1 point by big\_asker 10 days ago | [hide](#) | [discuss](#)

---

[Guidelines](#) | [FAQ](#) | [Lists](#) | [API](#) | [Security](#) | [Legal](#) | [Apply to YC](#) | [Contact](#)

Search:

# the good, the bad & the ugly

spec of HackedNews is < 1,000 words, 100 lines  
this **includes** specs of all concepts  
**runs the first time**, only minor tweaks needed  
**recursive comments** coded from mention of "comment tree"

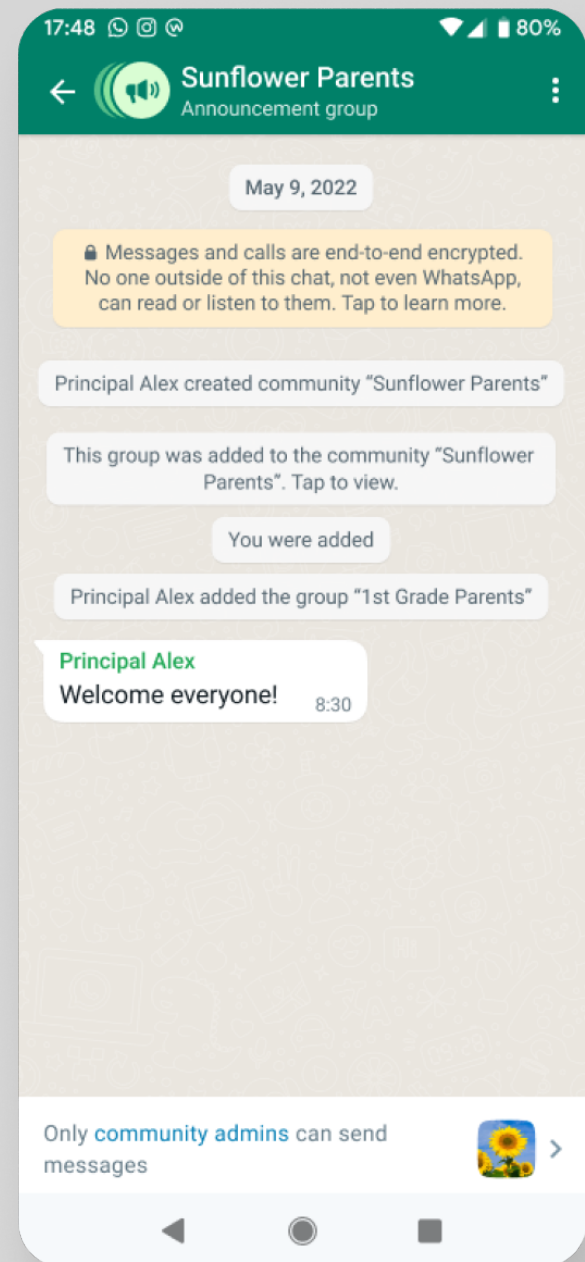
**frontend** is not yet built automatically, but we're close  
sometimes **code quality** of routes isn't great (but they work)  
sometimes generated code contains **type errors** but still works  
some **routes are complicated** to describe

GET /posts/sort?count=<count>&page=<page>&prefix=<prefix> – For this query, only consider posts from last 4 days. Sort them by:  $(p-1)/(hours\_since\_creation)^{1.8-5f}$  formula where p is points of this post and f is the number of times it got flagged. Returns posts with their points, number of comments, author's user profile, and if the currently logged in user voted (if no logged in user, skip that). Do not include hidden posts. Use default of count 30 and page 1, no prefix. All query params are optional.

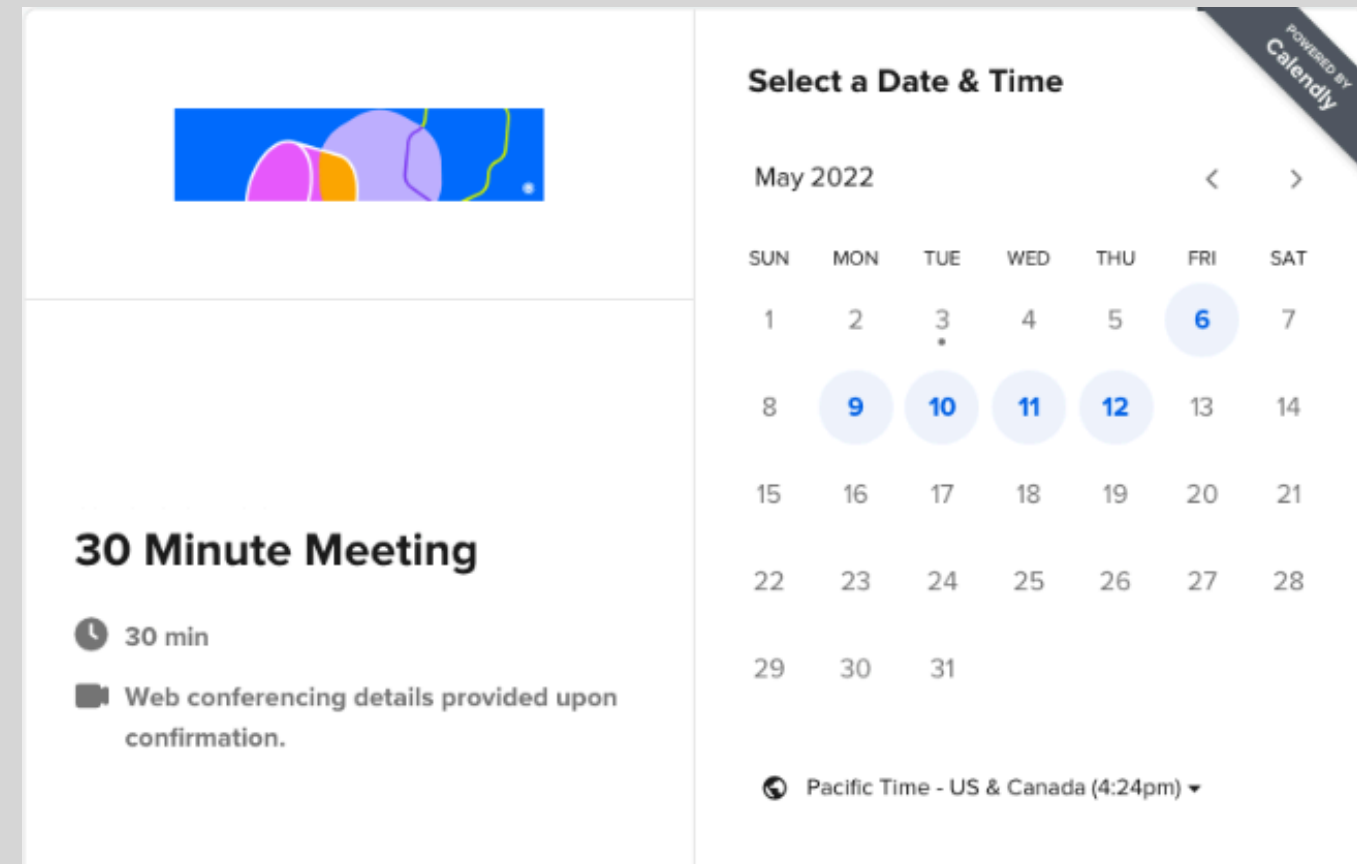


*takeaways*

# apps are characterized by their concepts, sometimes just one



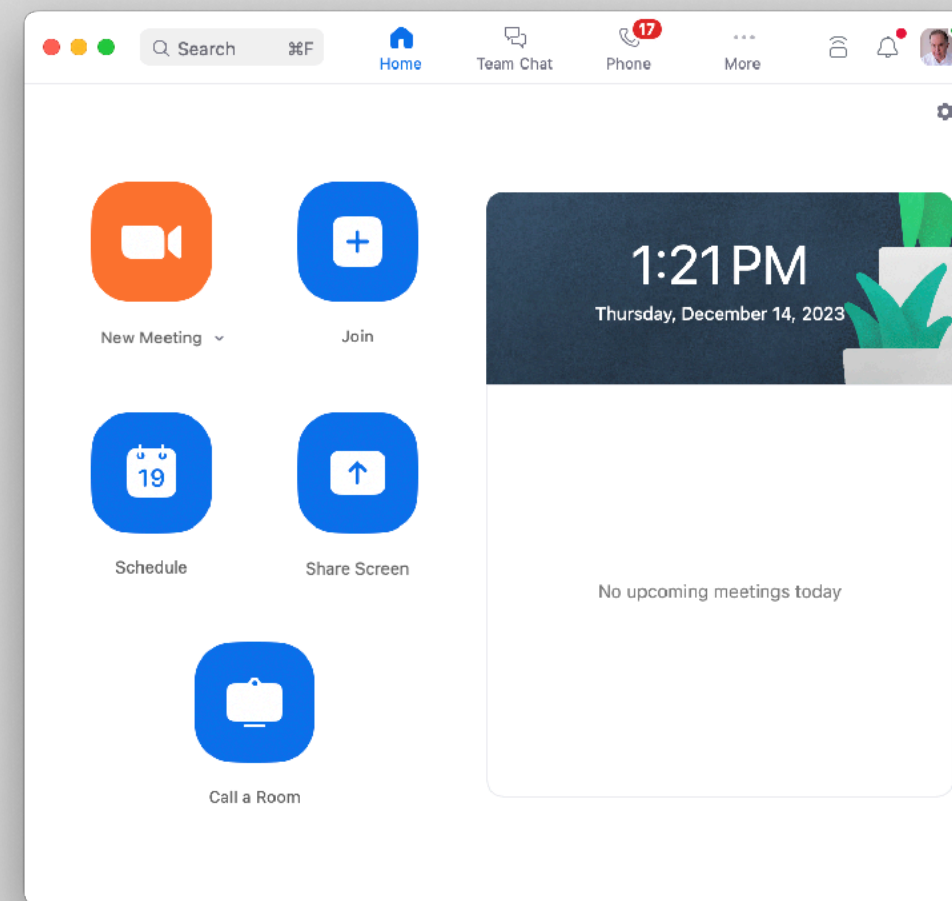
Group



SelfSchedule



Song



MeetingLink

## World Wide Web

The WorldWideWeb (W3) is a wide-area [hypermedia](#) information retrieval initiative aiming to give universal access to a large universe of documents.

Everything there is online about W3 is linked directly or indirectly to this document, including an [executive summary](#) of the project, [Mailing lists](#), [Policy](#), November's [W3 news](#), [Frequently Asked Questions](#).

### [What's out there?](#)

Pointers to the world's online information, [subjects](#), [W3 servers](#), etc.

### [Help](#)

on the browser you are using

### [Software Products](#)

A list of W3 project components and their current state. (e.g. [Line Mode](#), [X11 Viola](#), [NeXTStep](#), [Servers](#), [Tools](#), [Mail robot](#), [Library](#))

### [Technical](#)

Details of protocols, formats, program internals etc

### [Bibliography](#)

Paper documentation on W3 and references.

### [People](#)

A list of some people involved in the project.

### [History](#)

A summary of the history of the project.

### [How can I help?](#)

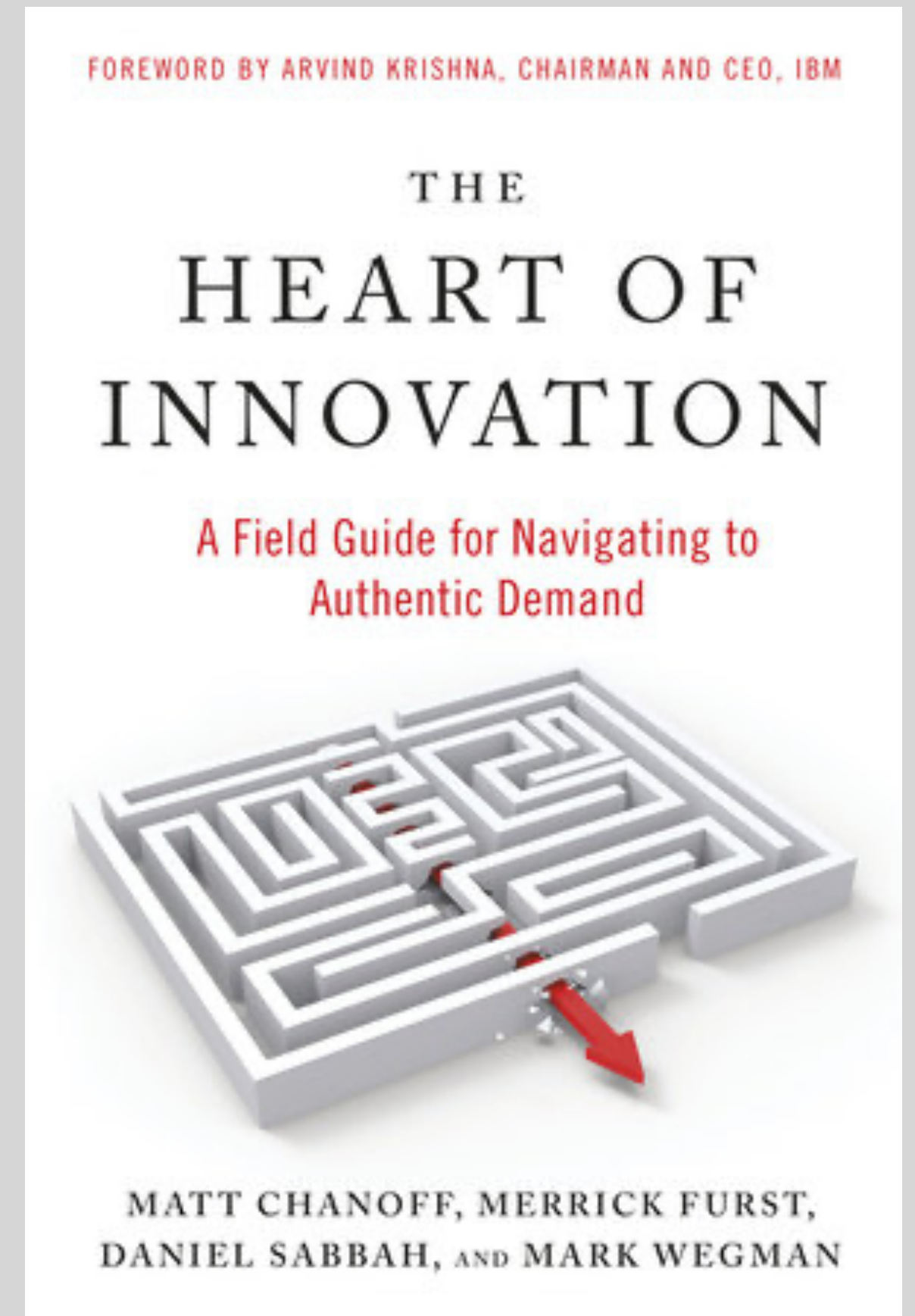
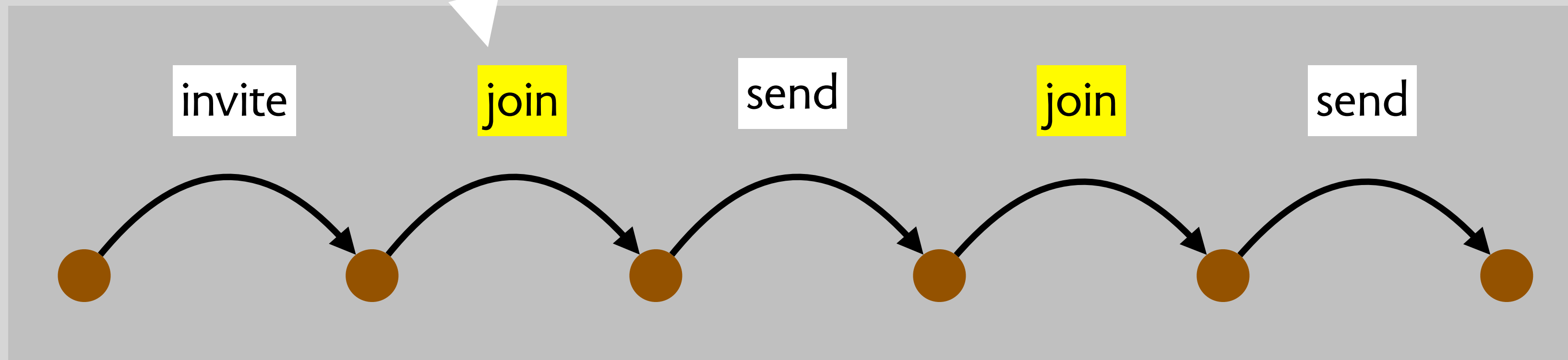
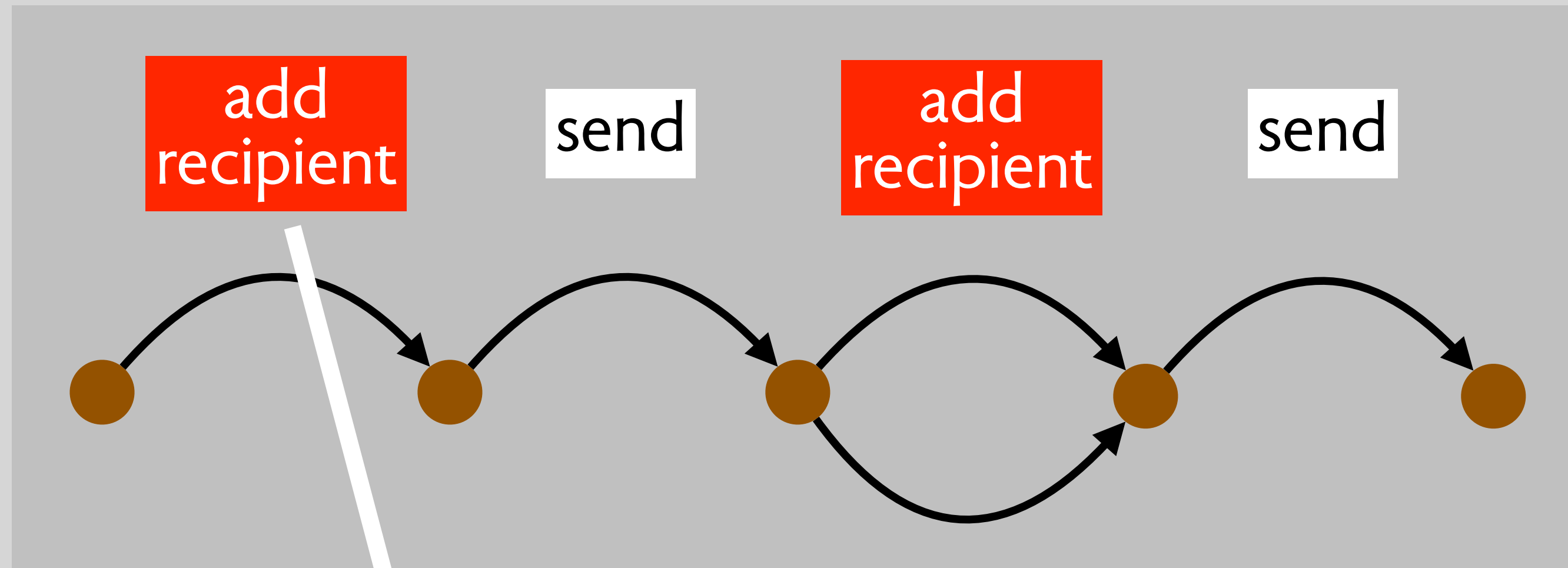
If you would like to support the web..

### [Getting code](#)

Getting the code by [anonymous FTP](#), etc.

URL

# innovations simplify scenarios



“not nots”:  
it's **not** OK to **not** have  
group functionality

**modularity is key**

concepts  
are fully  
independent

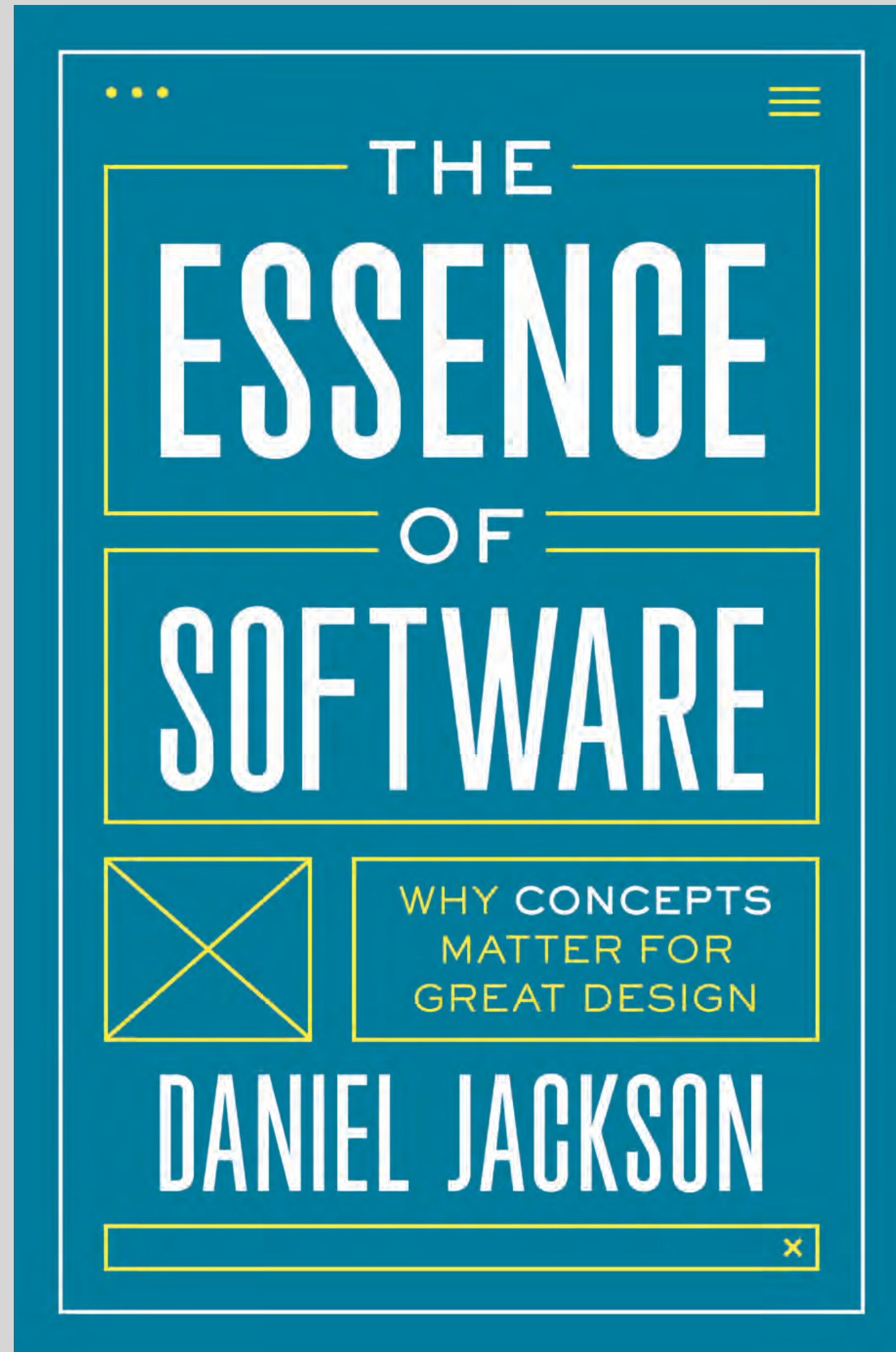
**factor out familiar parts**

upvote  
untainted by  
karma rules

**goodbye agile**

now it's specs  
that matter,  
not code

much more in book & website



[essenceofsoftware.com/subscribe](https://essenceofsoftware.com/subscribe)

why a missing concept could be Zoom's downfall

dark concepts: going rogue

how HCI got misled about conceptual models