the essence of software

Daniel Jackson, MIT CSAIL · ACM Tech Talk · Dec 1, 2021

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** and elements of the application must be driven by the conditions of use. **How is this to be done?** By software designers.

A Software Design Manifesto
Mitchell Kapor, 1996
paraphrasing slightly

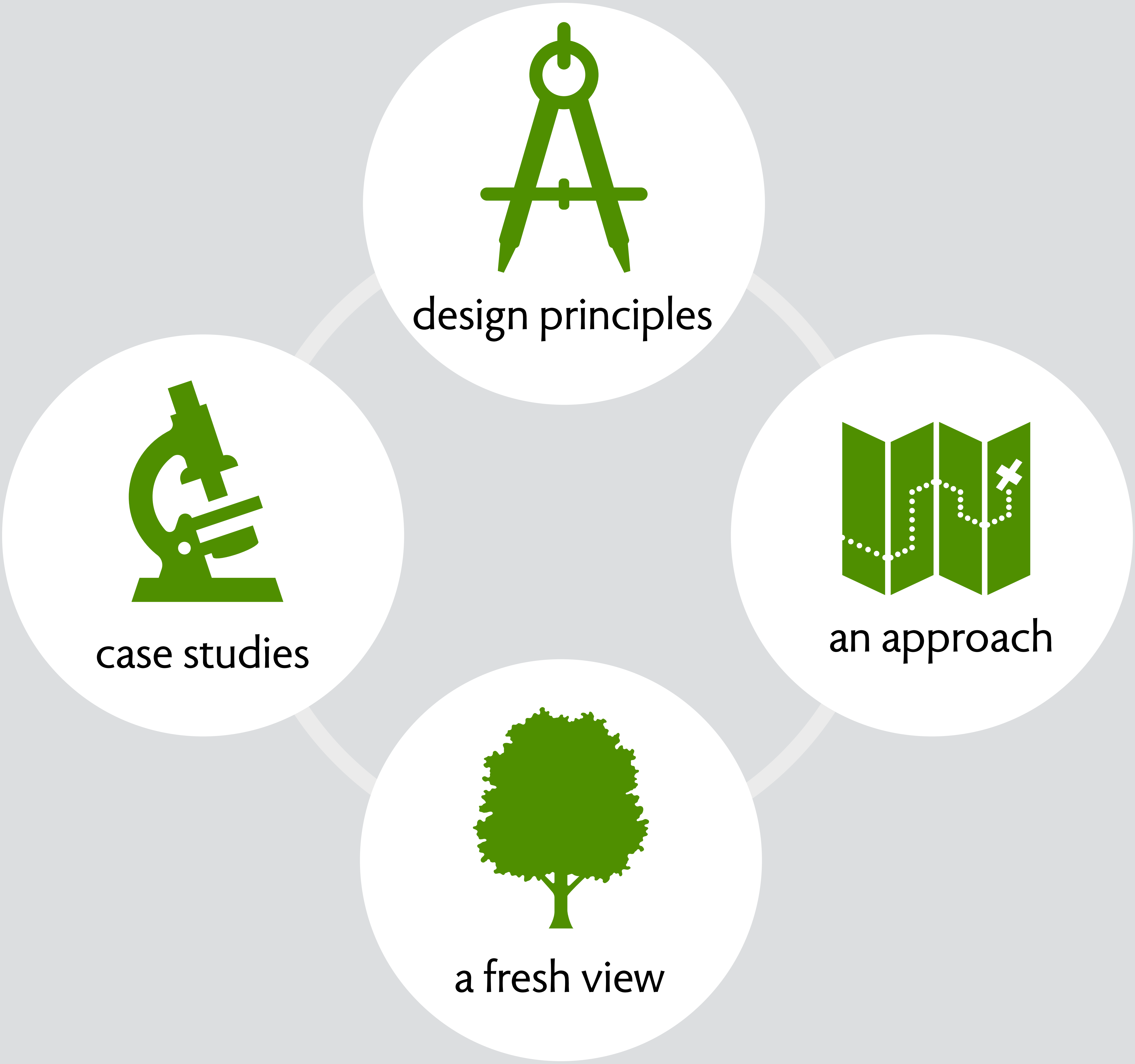
“ make sure the user has the right conceptual model ”

“ get the concepts straight and everything else will fall into place ”

but what exactly are concepts? and how do get them straight?

this talk:

just a flavor, but hopefully enough to give you a new way to think about software



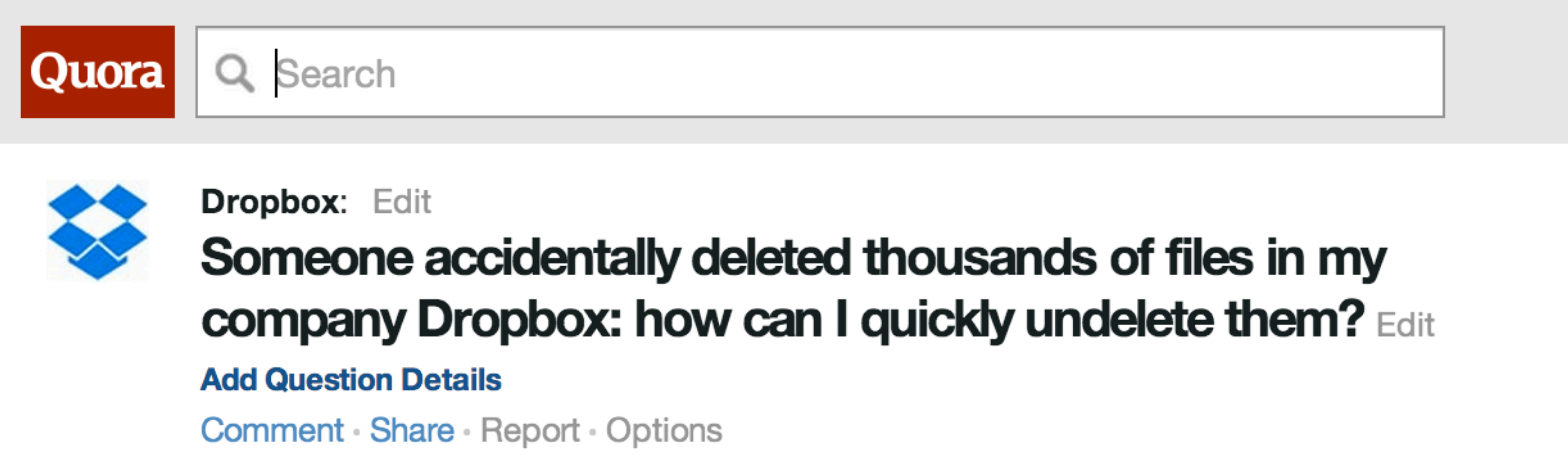
design principles

an approach

a fresh view

case studies

puzzle #1: Dropbox & the case of the disappearing files

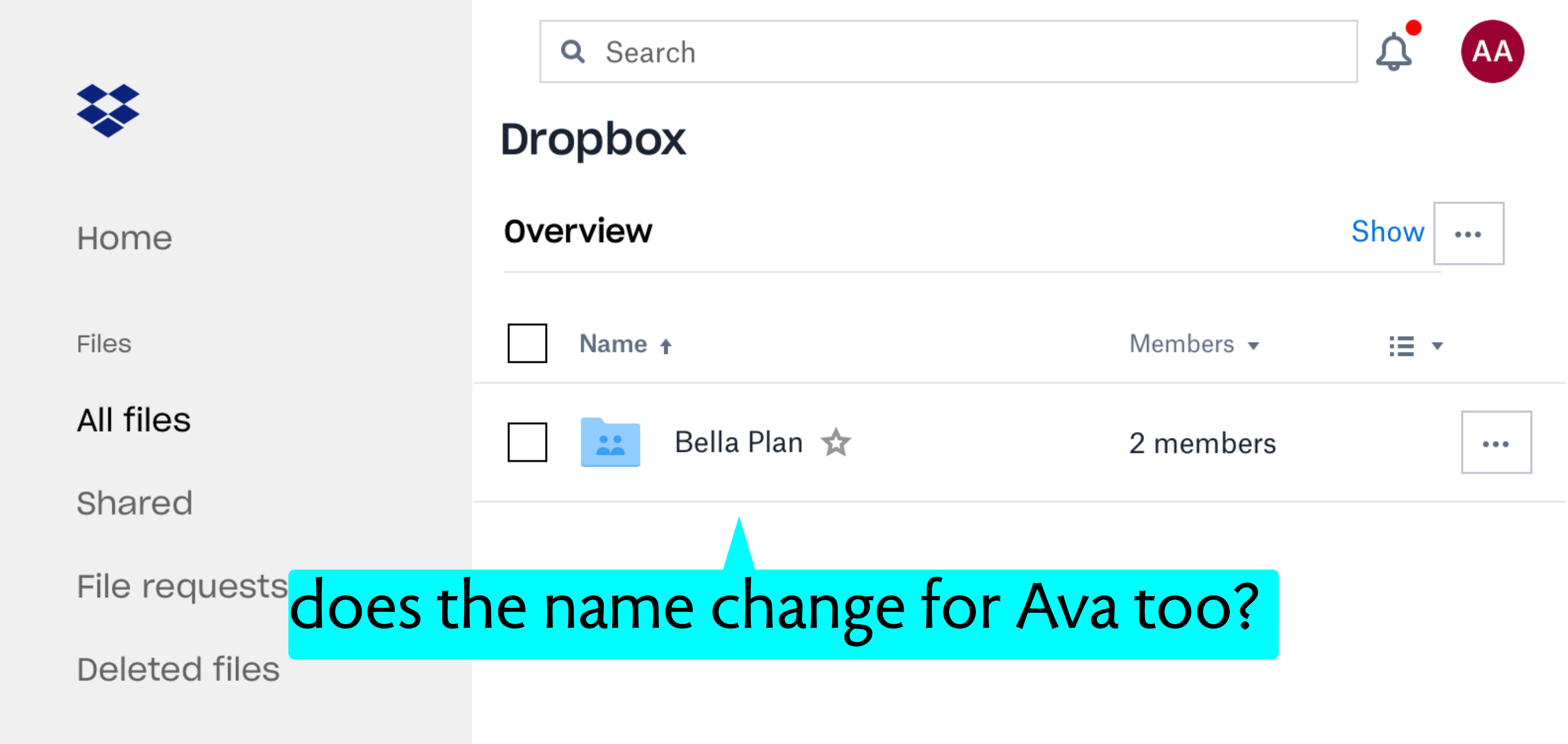


Dropbox:

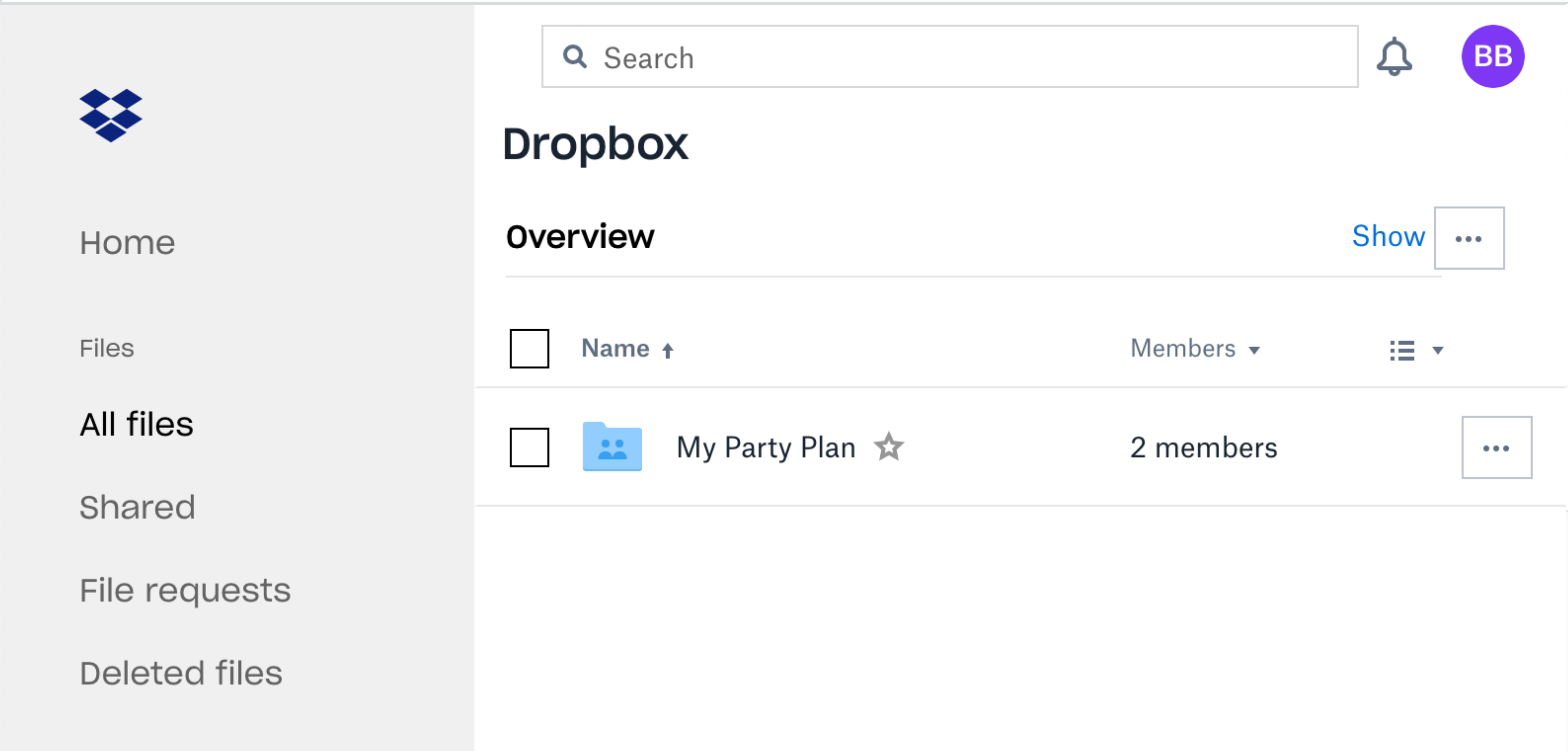
Someone accidentally deleted thousands of files in my company Dropbox: how can I quickly undelete them?

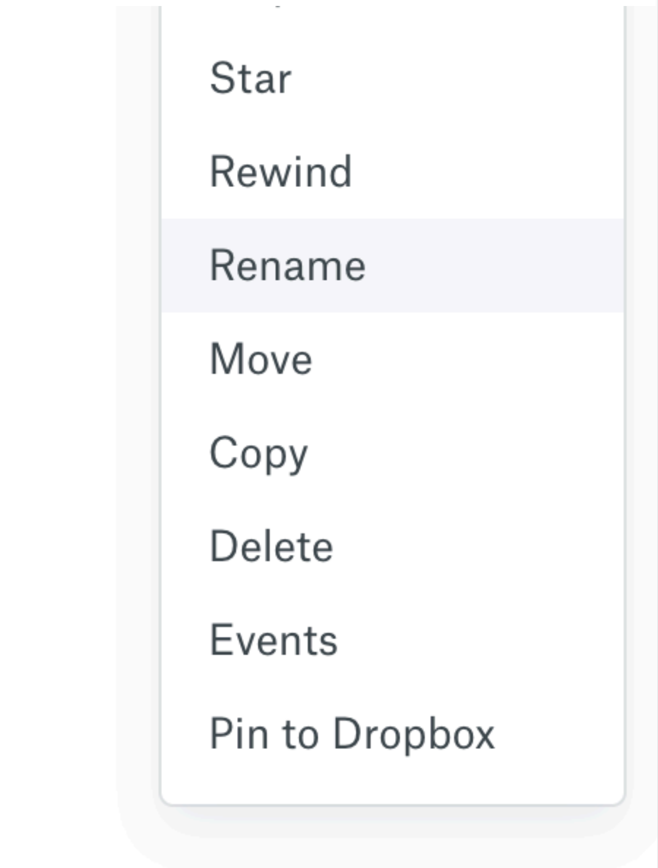
Ava is a party planner

Bella is having a party



does the name change for Ava too?



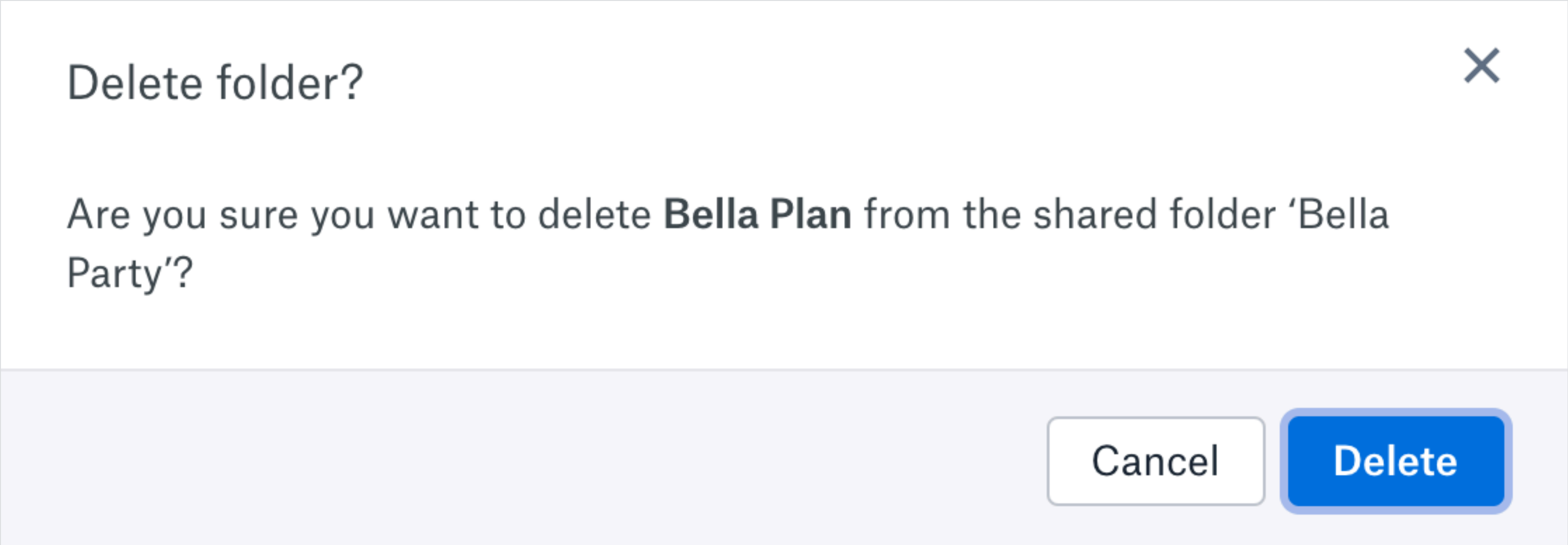


answer: it depends

if Ava just shares Bella Plan with Bella
and Bella renamed the folder, Ava sees no change

if Ava shared a folder containing Bella Plan
then Ava does see the change

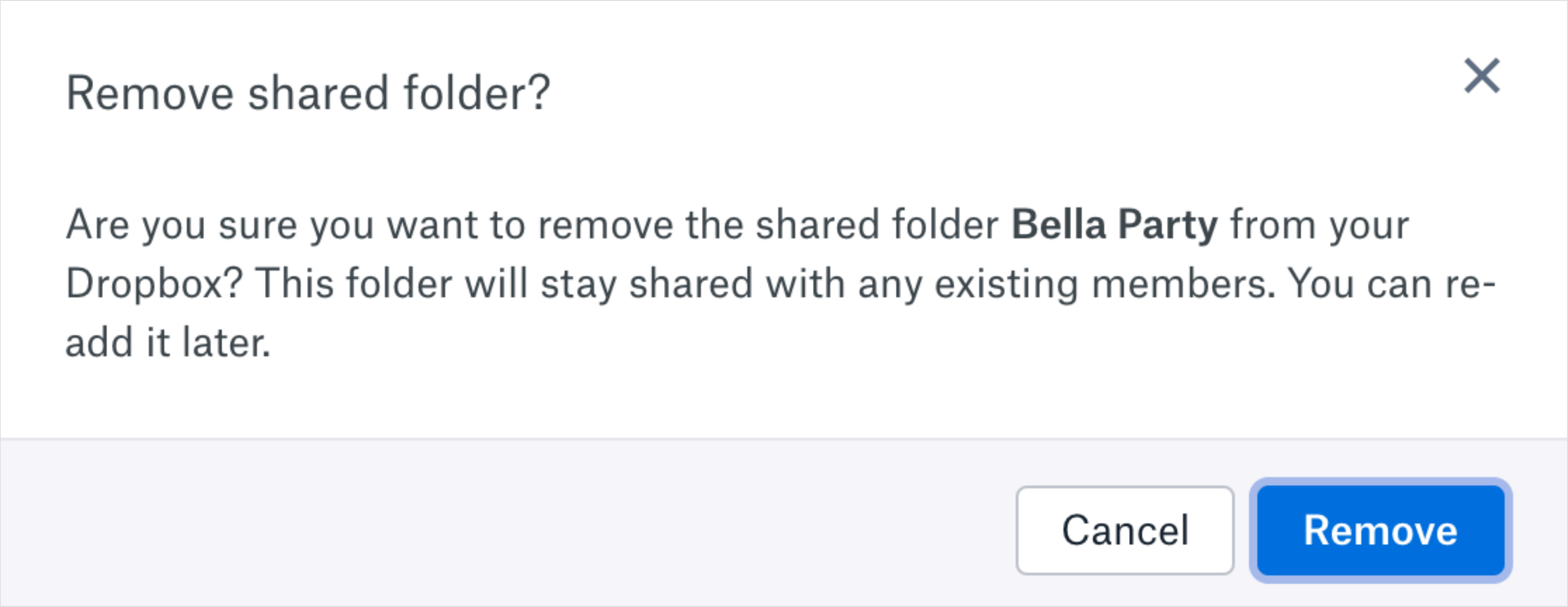
Bella deletes Bella Plan from shared folder Bella Party



Delete folder?

Are you sure you want to delete **Bella Plan** from the shared folder 'Bella Party'?

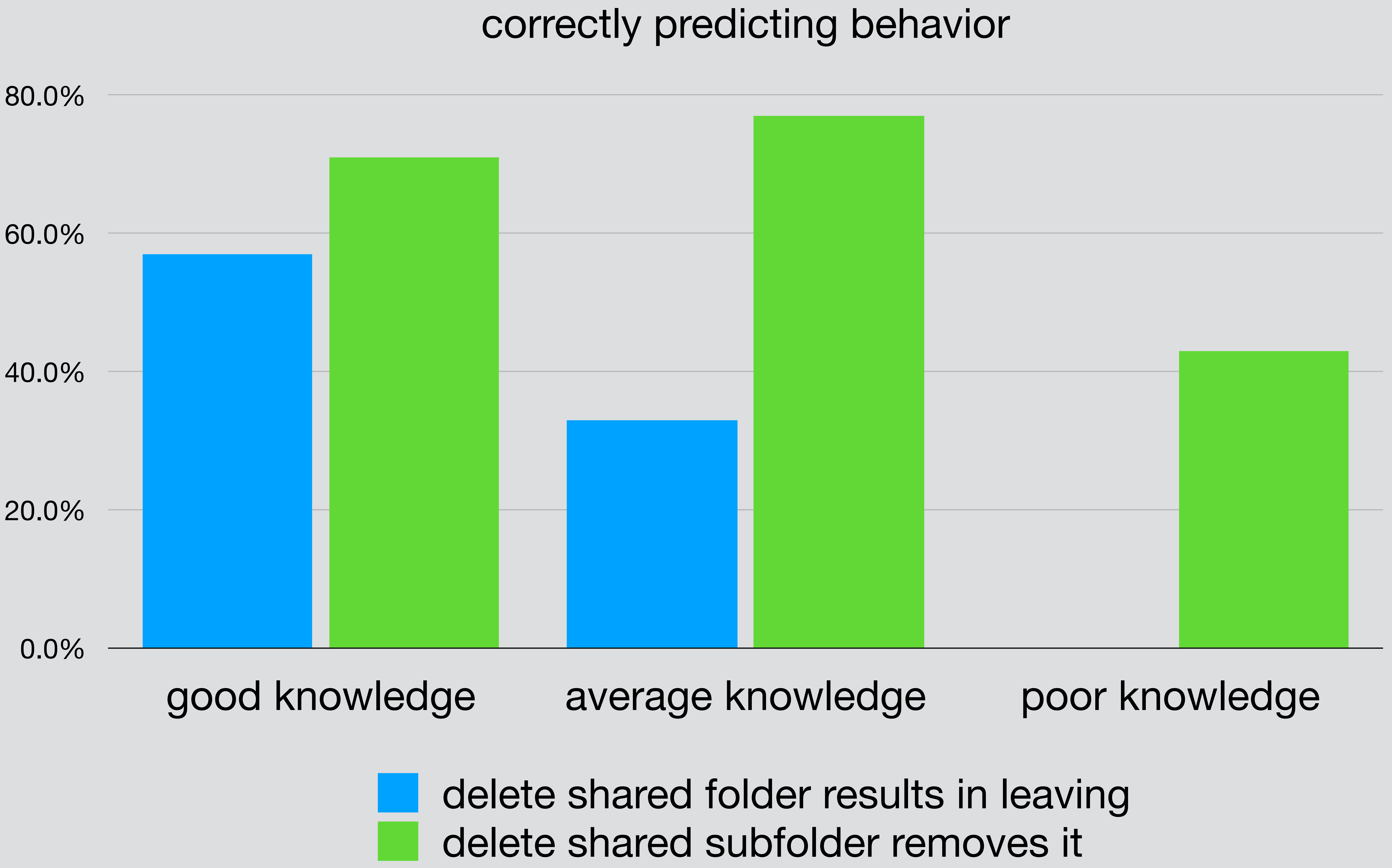
Bella deletes shared folder Bella Party



Remove shared folder?

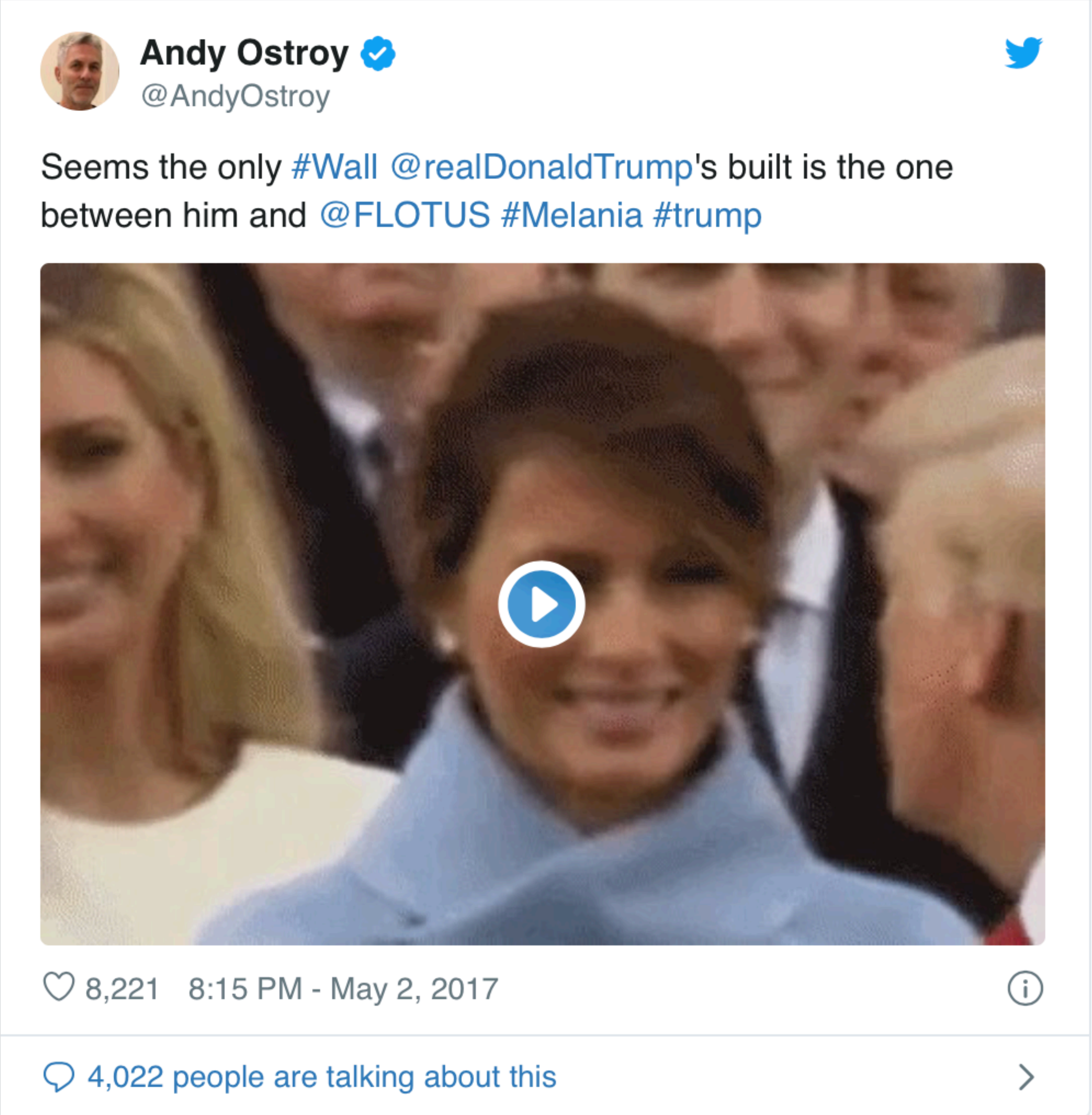
Are you sure you want to remove the shared folder **Bella Party** from your Dropbox? This folder will stay shared with any existing members. You can re-add it later.

survey of dropbox users (MIT CS undergrads)



Kelly Zhang

puzzle #2: Twitter & the case of the surprised first lady

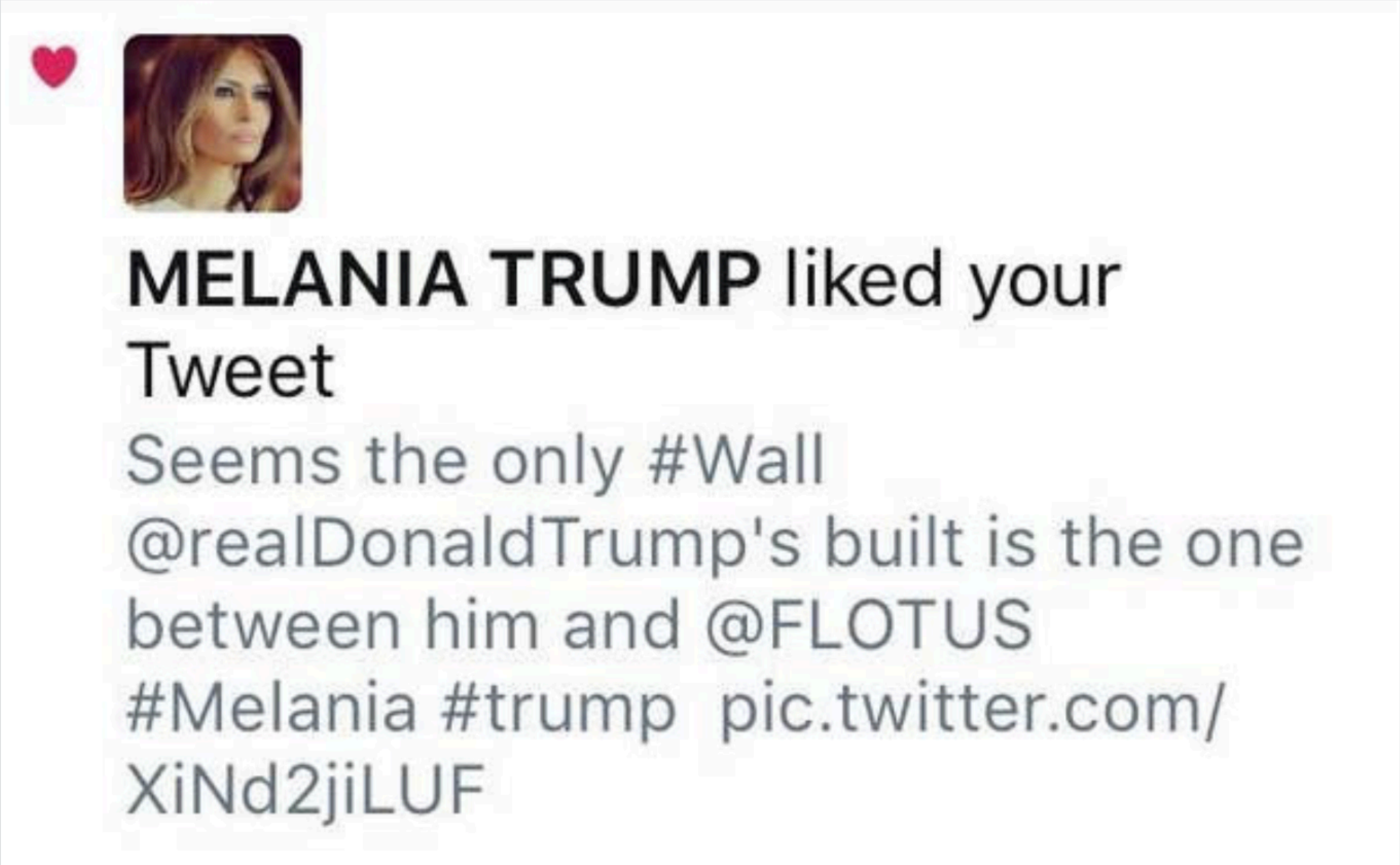


Andy Ostroy
@AndyOstroy

Seems the only #Wall @realDonaldTrump's built is the one between him and @FLOTUS #Melania #trump

♡ 8,221 8:15 PM - May 2, 2017

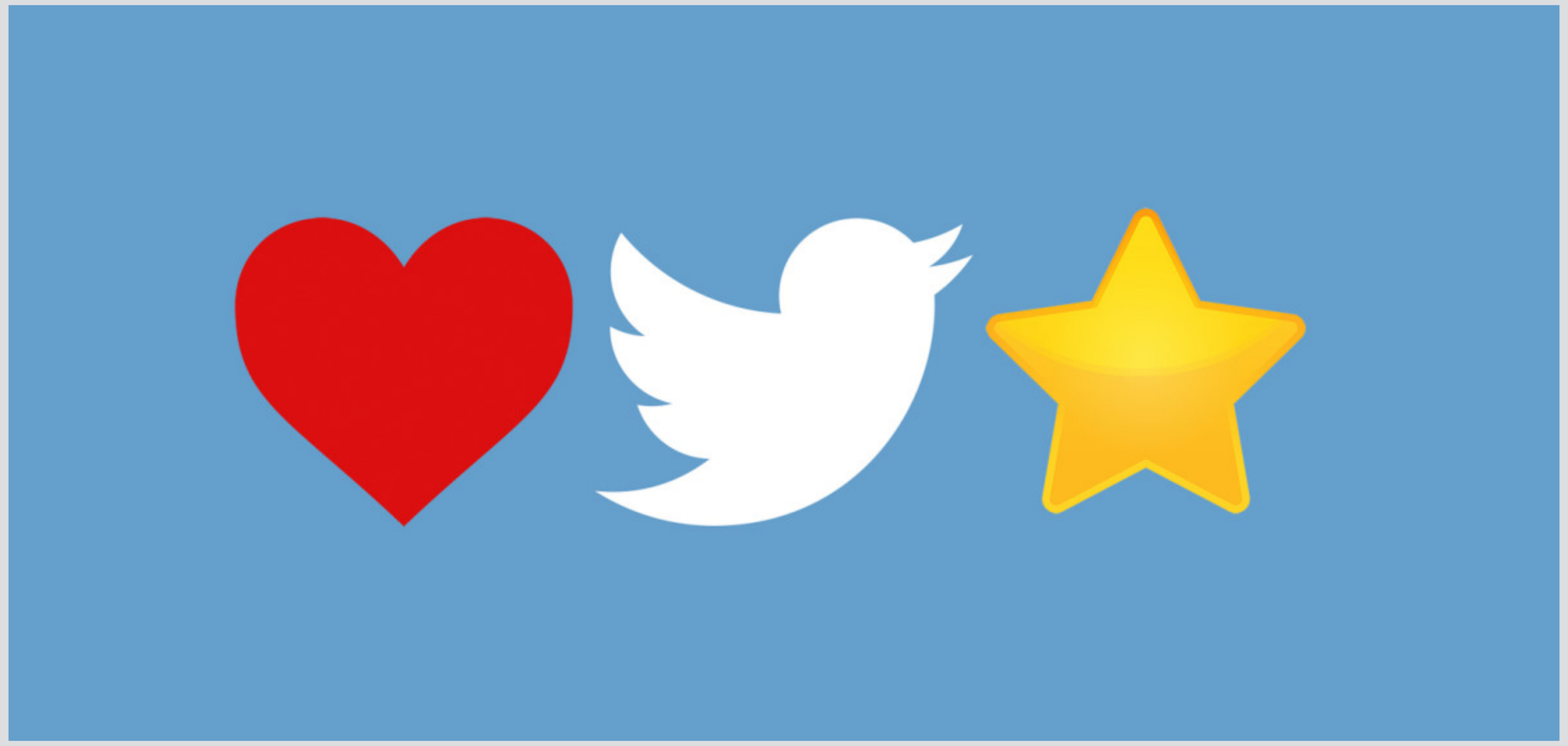
4,022 people are talking about this



♥

MELANIA TRUMP liked your Tweet

Seems the only #Wall @realDonaldTrump's built is the one between him and @FLOTUS #Melania #trump pic.twitter.com/XiNd2jiLUF

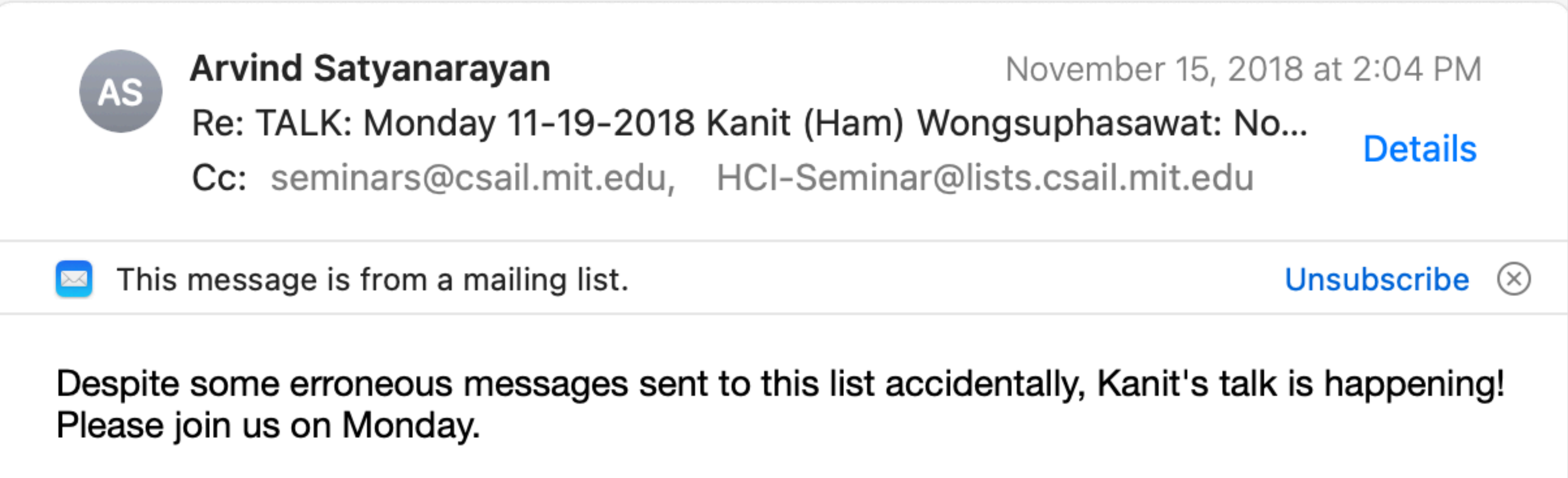


Nov 2, 2015: Twitter changes Favorite (Star) to Like (Heart)

We are changing our star icon for favorites to a heart and we'll be calling them likes... **We know that at times the star could be confusing, especially to newcomers**. You might like a lot of things, but not everything can be your favorite. *Twitter press release*

puzzle #3: Google Calendar

& the case of the mysterious cancellations



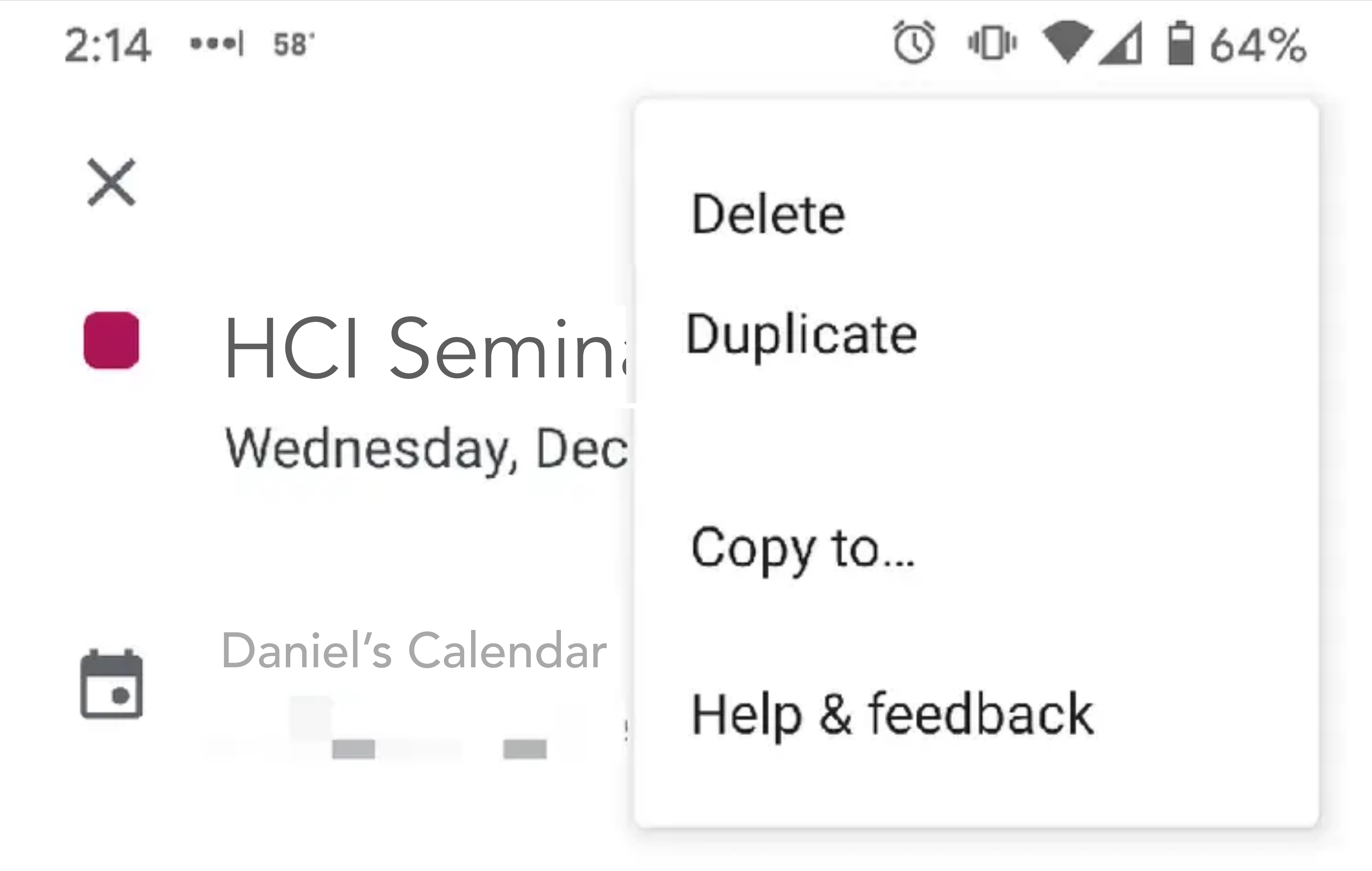
Arvind Satyanarayan November 15, 2018 at 2:04 PM

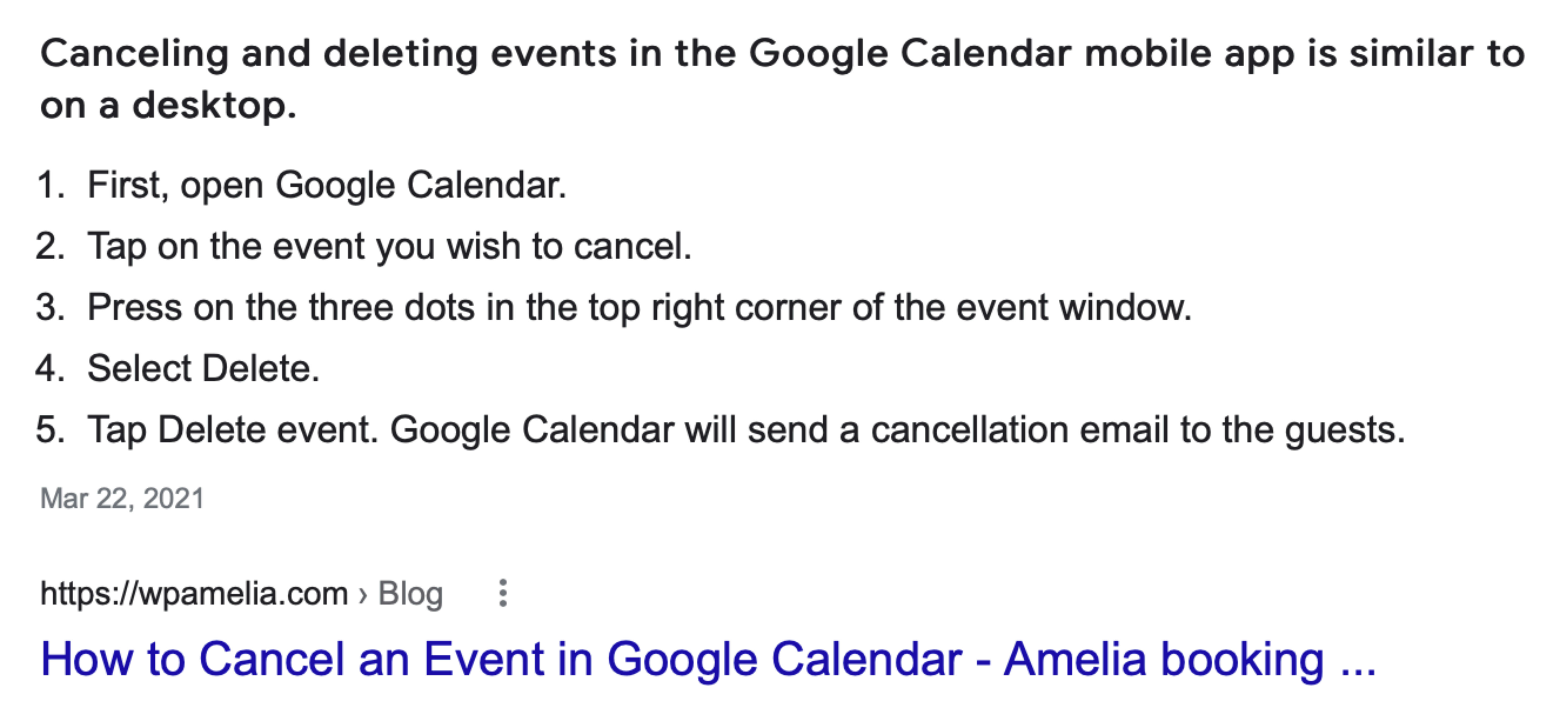
Re: TALK: Monday 11-19-2018 Kanit (Ham) Wongsuphasawat: No...

Cc: seminars@csail.mit.edu, HCI-Seminar@lists.csail.mit.edu

This message is from a mailing list.

Despite some erroneous messages sent to this list accidentally, Kanit's talk is happening! Please join us on Monday.





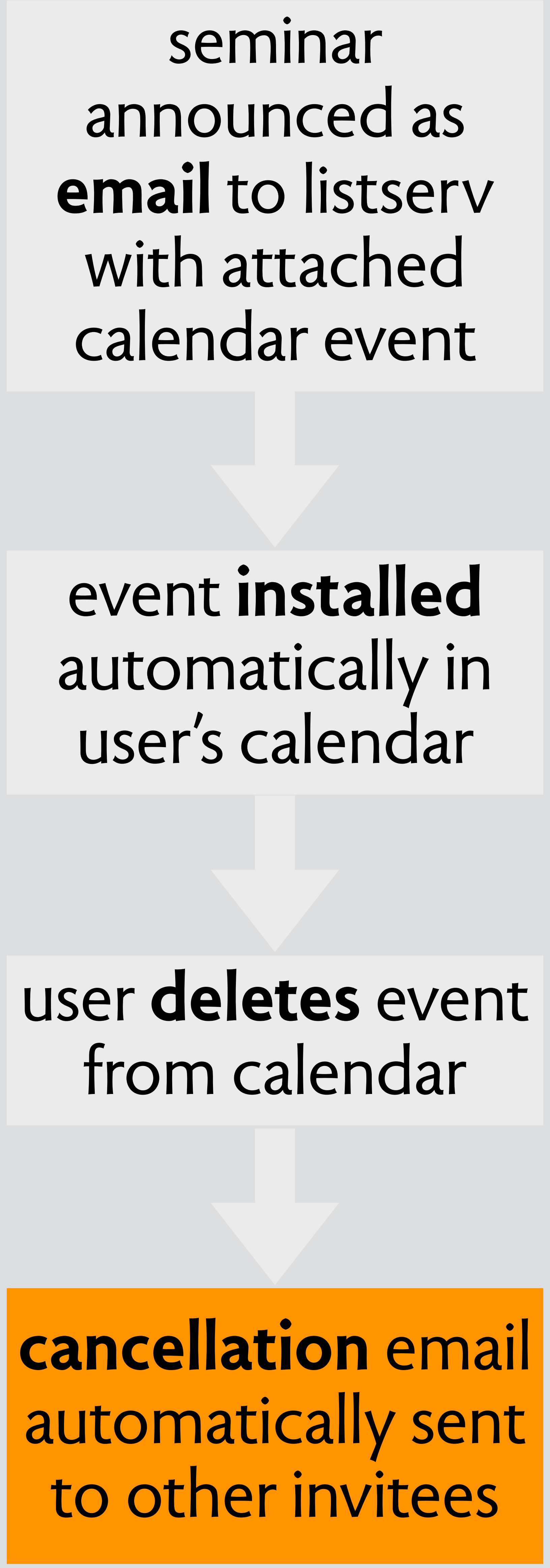
Canceling and deleting events in the Google Calendar mobile app is similar to on a desktop.

1. First, open Google Calendar.
2. Tap on the event you wish to cancel.
3. Press on the three dots in the top right corner of the event window.
4. Select Delete.
5. Tap Delete event. Google Calendar will send a cancellation email to the guests.

Mar 22, 2021

https://wpamelia.com › Blog

How to Cancel an Event in Google Calendar - Amelia booking ...



seminar announced as **email** to listserv with attached calendar event

event **installed** automatically in user's calendar

user **deletes** event from calendar

cancellation email automatically sent to other invitees

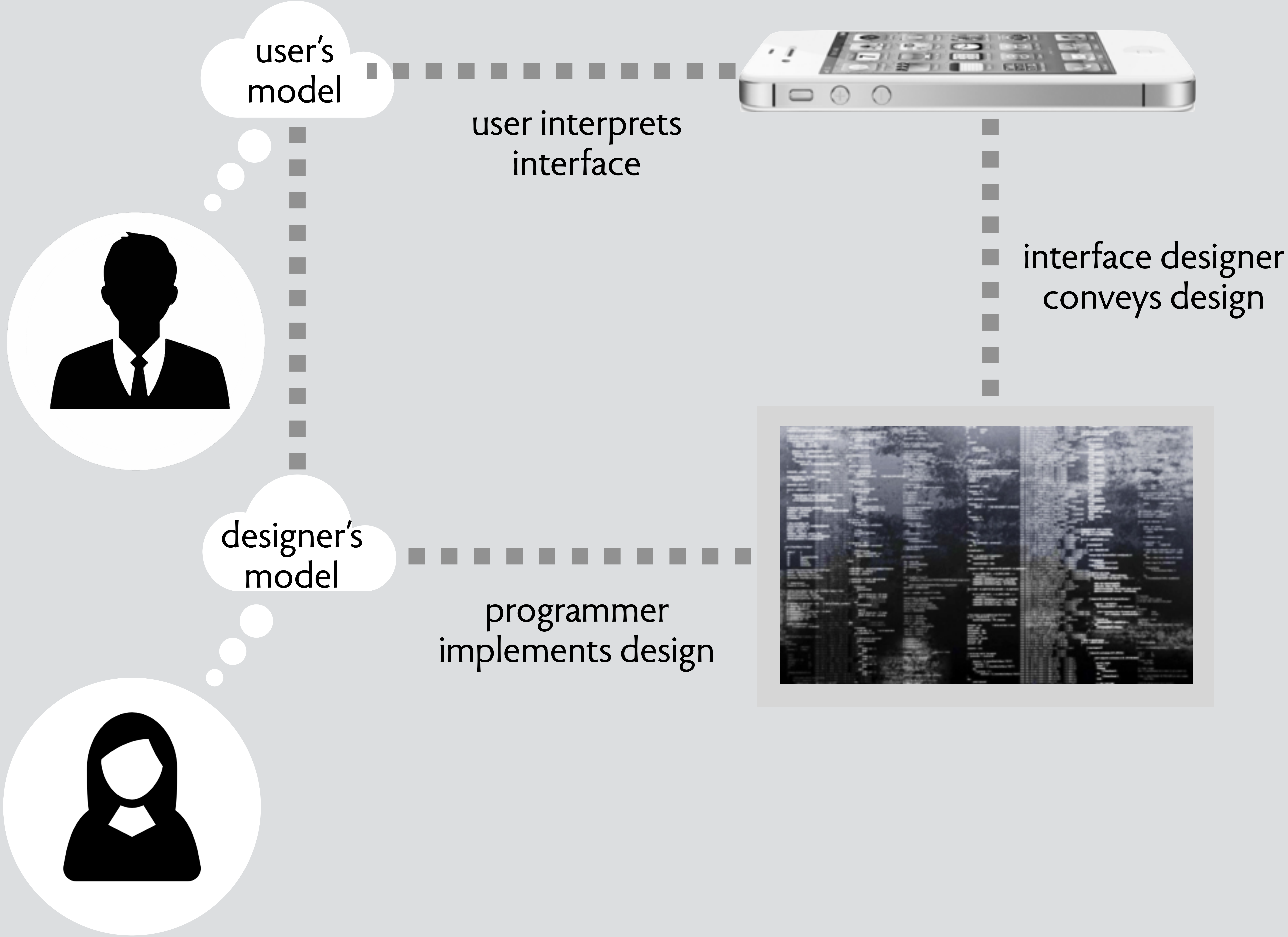
a new path
inspired by old ideas

what kinds of problems are these?

not just human errors

not bugs in the code

not classic UI design flaws



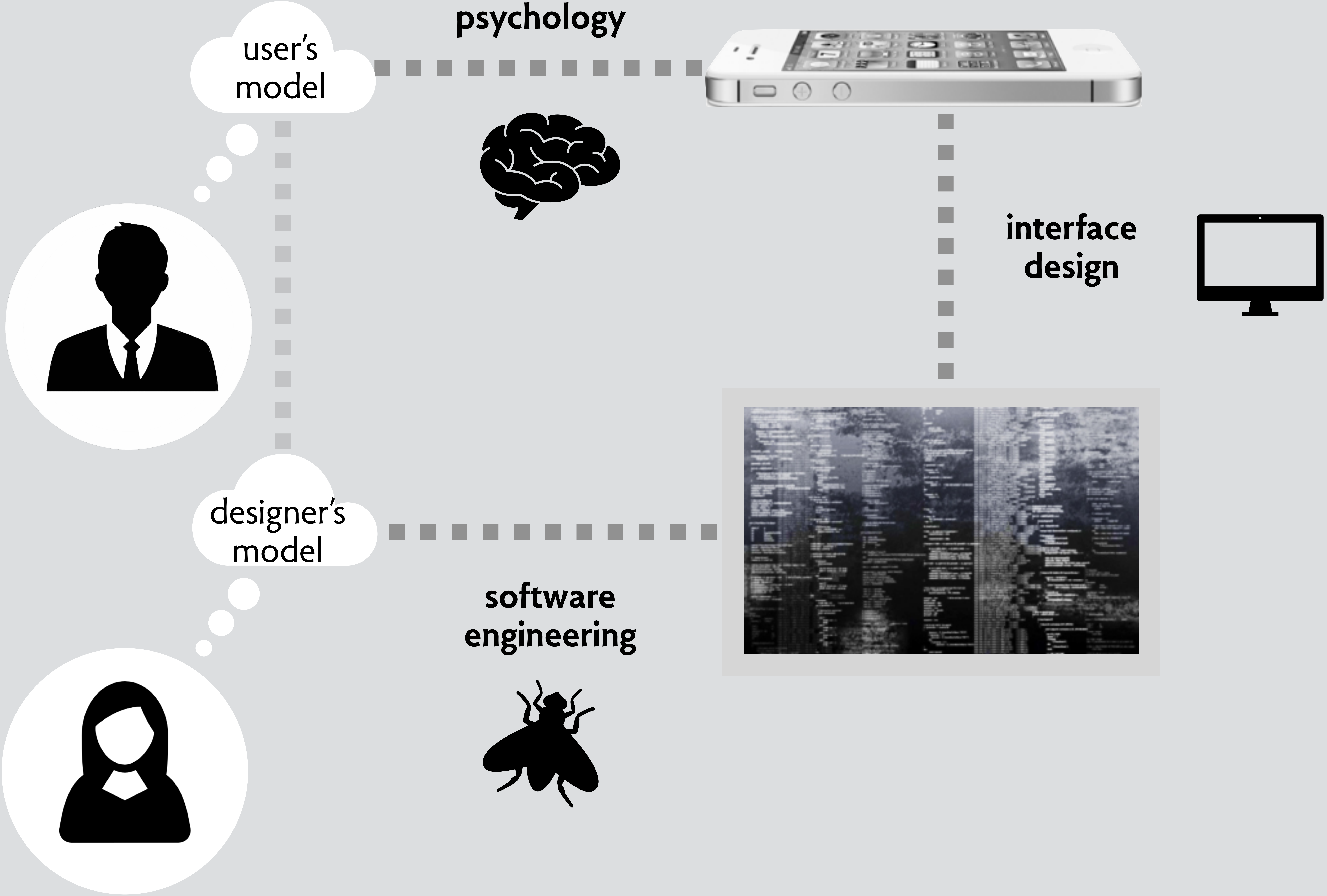
user’s model

user interprets interface

interface designer conveys design

designer’s model

programmer implements design



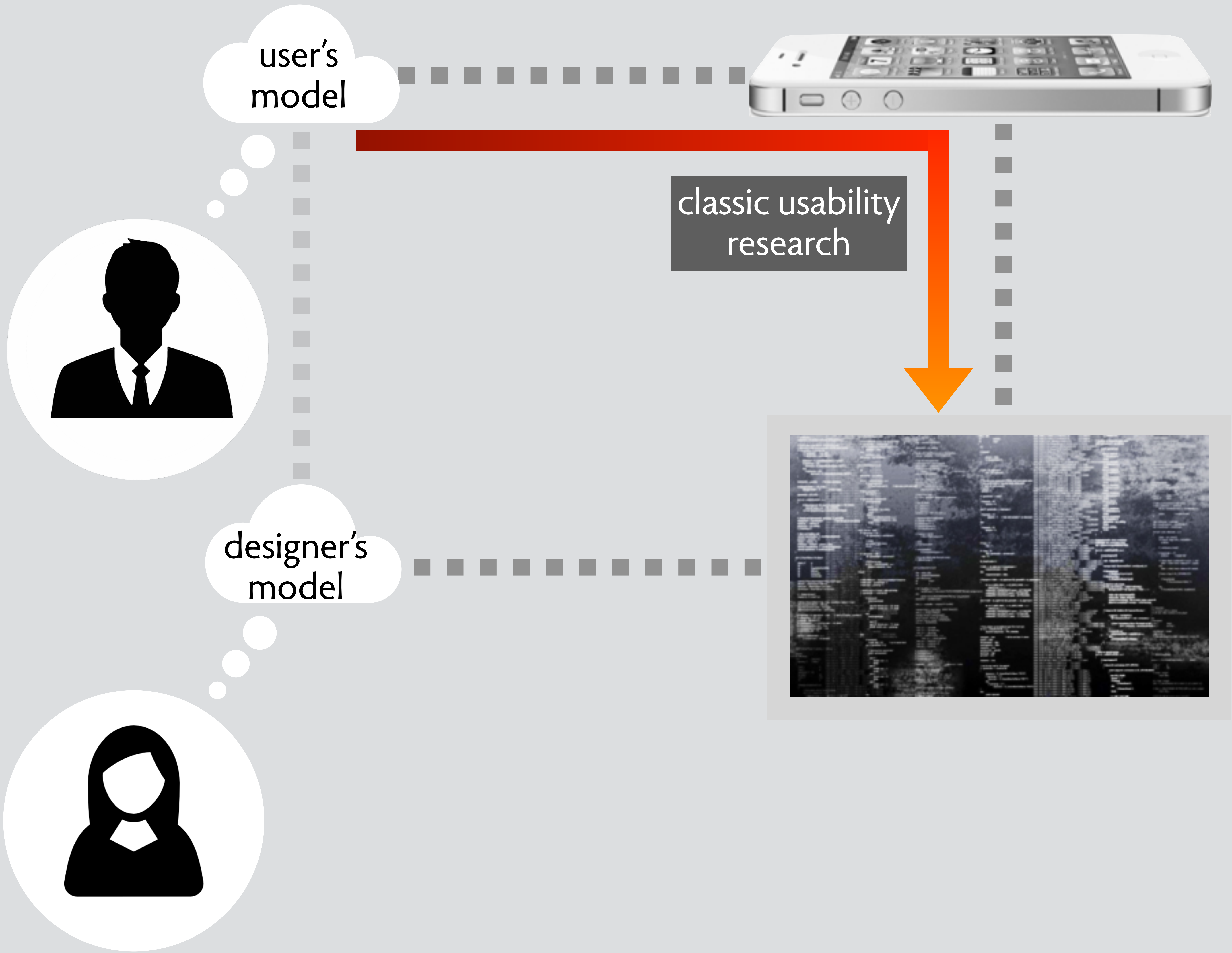
user's model

psychology

interface design

designer's model

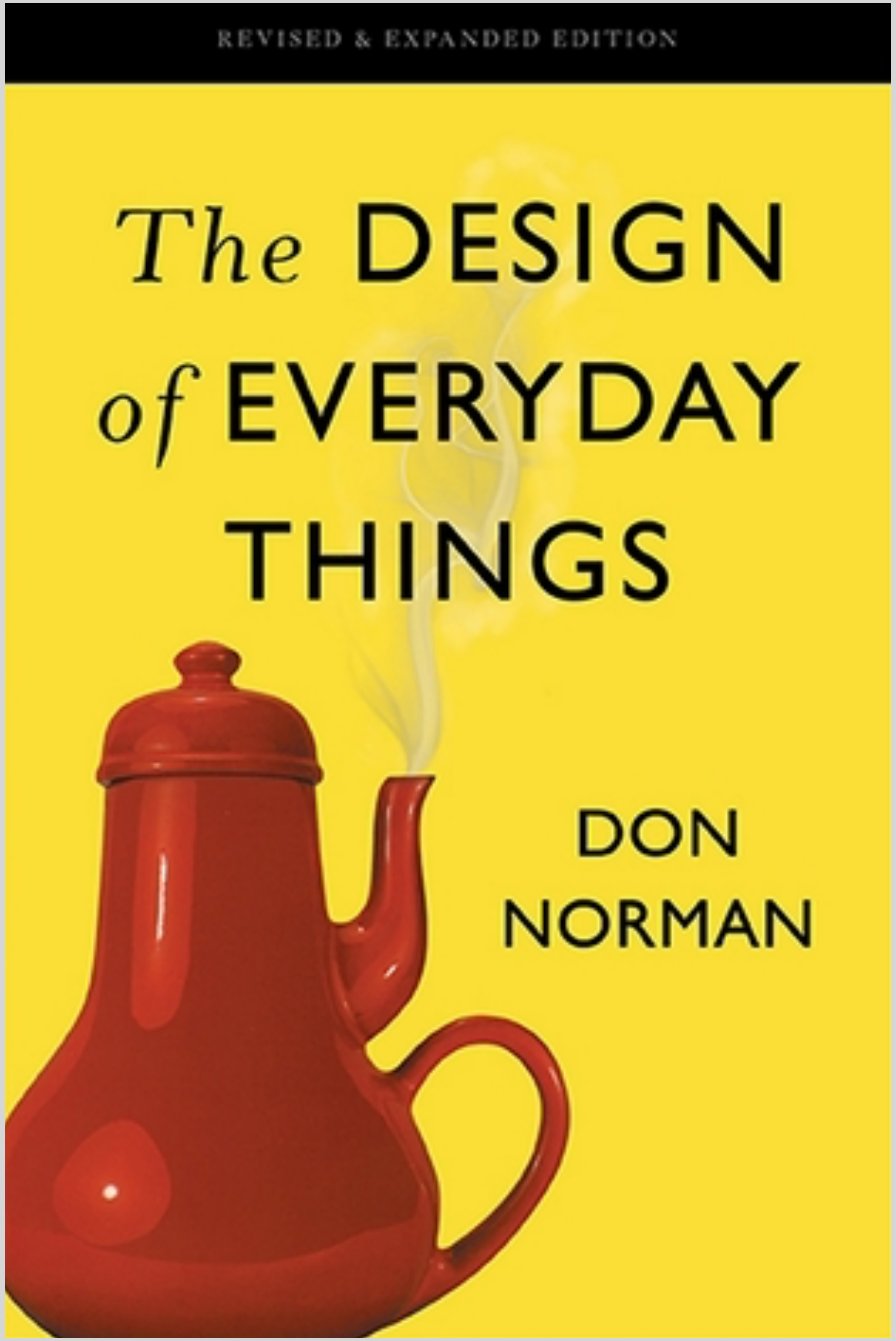
software engineering



user’s model

classic usability research

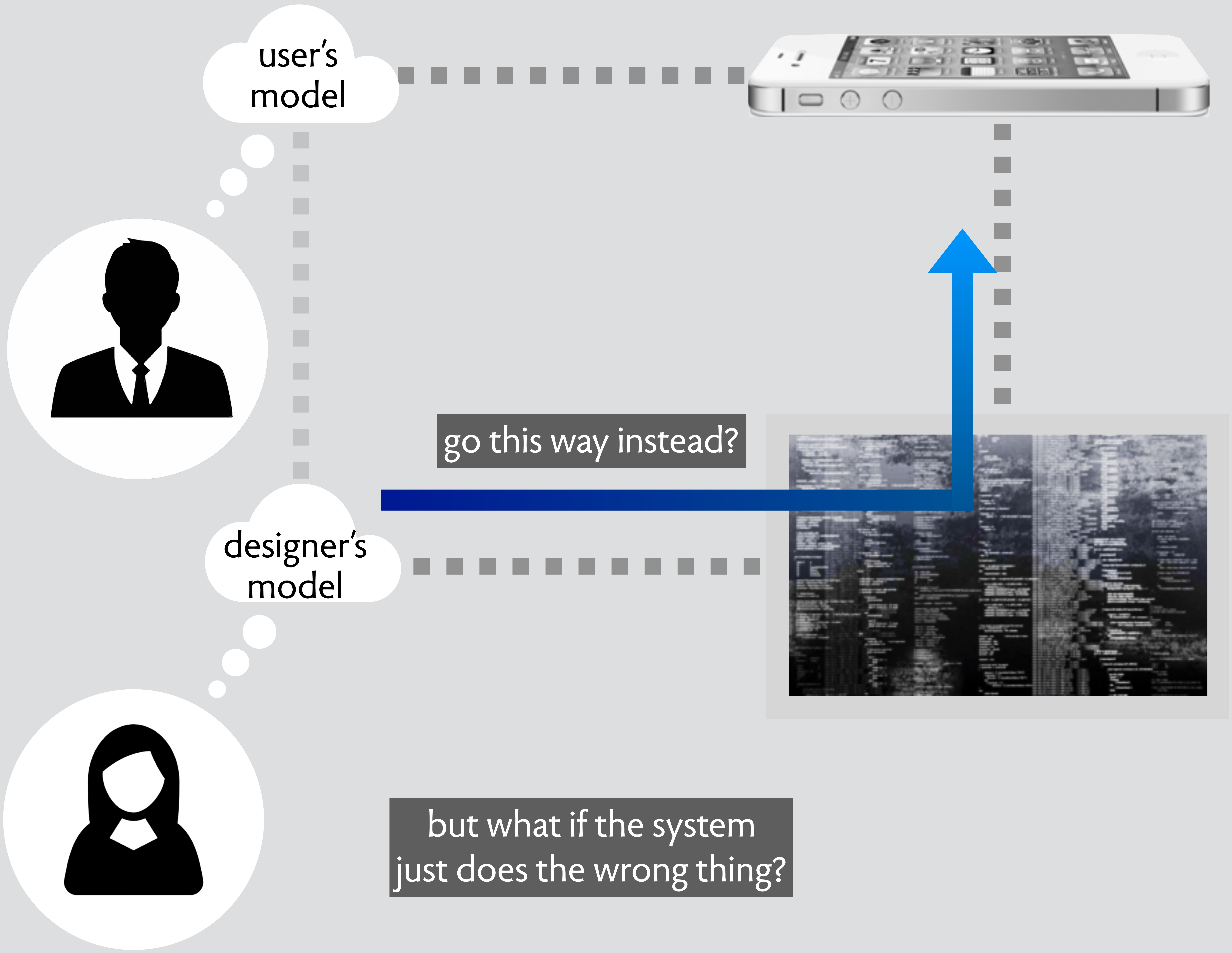
designer’s model



focus on faithful projection of design model
mapping, affordance/signifiers, gulfs

theory turned into practice through heuristics
8 Golden Rules (Shneiderman)
First Principles of Interaction Design (Tognazzini)
10 Usability Heuristics (Nielsen)

result: huge improvements in usability
anyone can design a great UI



user’s model

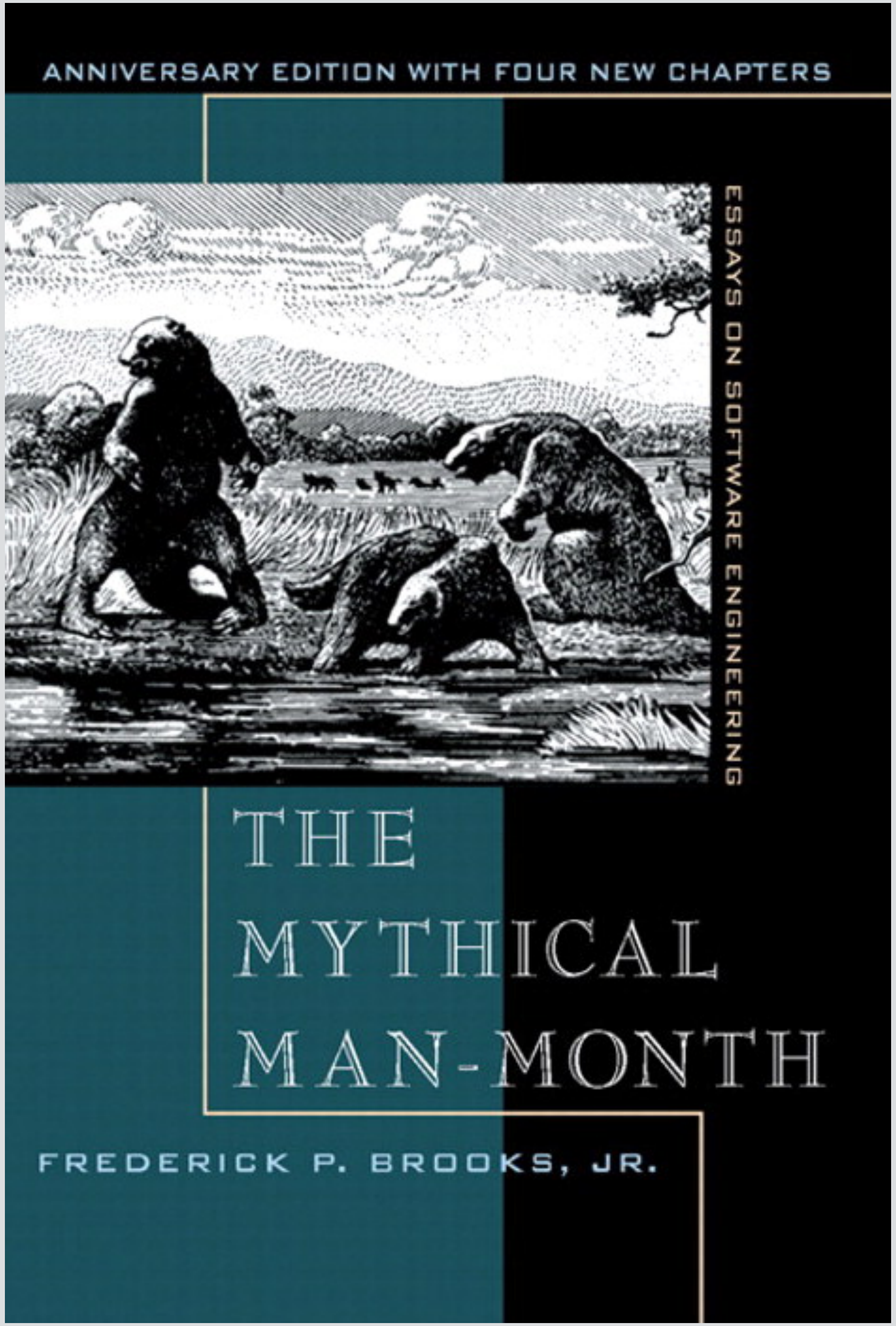
designer’s model

go this way instead?

but what if the system just does the wrong thing?

If a simple model is not explicitly or implicitly provided, users formulate their own myths about how the system works... The system has to be designed with an **explicit conceptual model** that is easy enough for the user to learn.

Stuart Card & Thomas Moran (1986)


ANNIVERSARY EDITION WITH FOUR NEW CHAPTERS
ESSAYS ON SOFTWARE ENGINEERING
THE MYTHICAL MAN-MONTH
FREDERICK P. BROOKS, JR.

> Conceptual integrity is the most important consideration in system design.

Fred Brooks, Mythical Man Month (1975)

> I am more convinced than ever. Conceptual integrity is central to product quality.

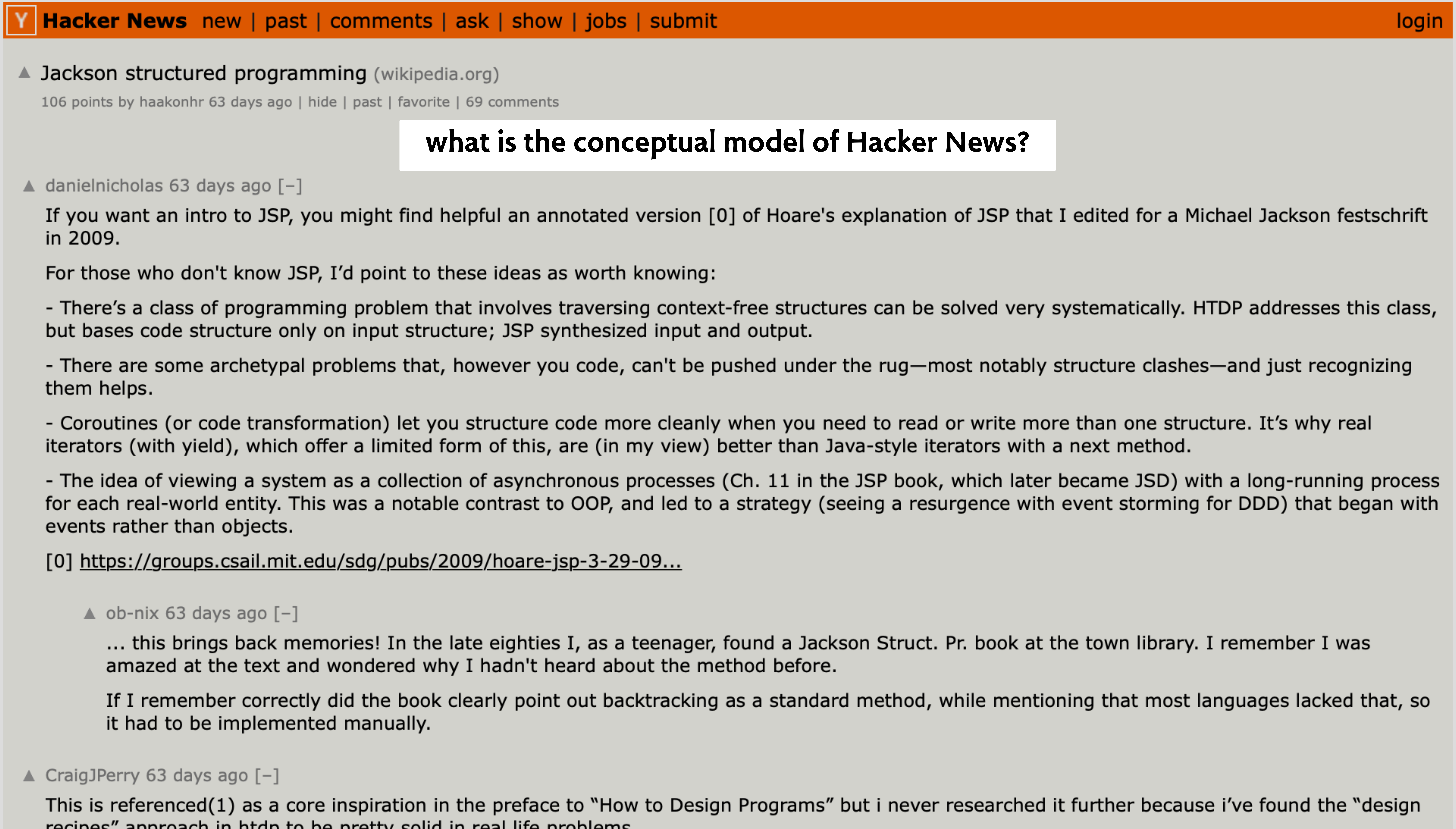
Mythical Man Month Anniversary Edition (1995)

> The essence of a software entity is a construct of interlocking concepts… I believe the hard part of building software to be the specification, design, and testing of this conceptual construct, not the labor of representing it…

No Silver Bullet (1986)

challenges

or, why it's not easy



▲ Jackson structured programming (wikipedia.org)

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

what is the conceptual model of Hacker News?

▲ danielnicholas 63 days ago [–]

If you want an intro to JSP, you might find helpful an annotated version [0] of Hoare's explanation of JSP that I edited for a Michael Jackson festschrift in 2009.

For those who don't know JSP, I'd point to these ideas as worth knowing:

- There's a class of programming problem that involves traversing context-free structures can be solved very systematically. HTDP addresses this class, but bases code structure only on input structure; JSP synthesized input and output.

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

- 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...

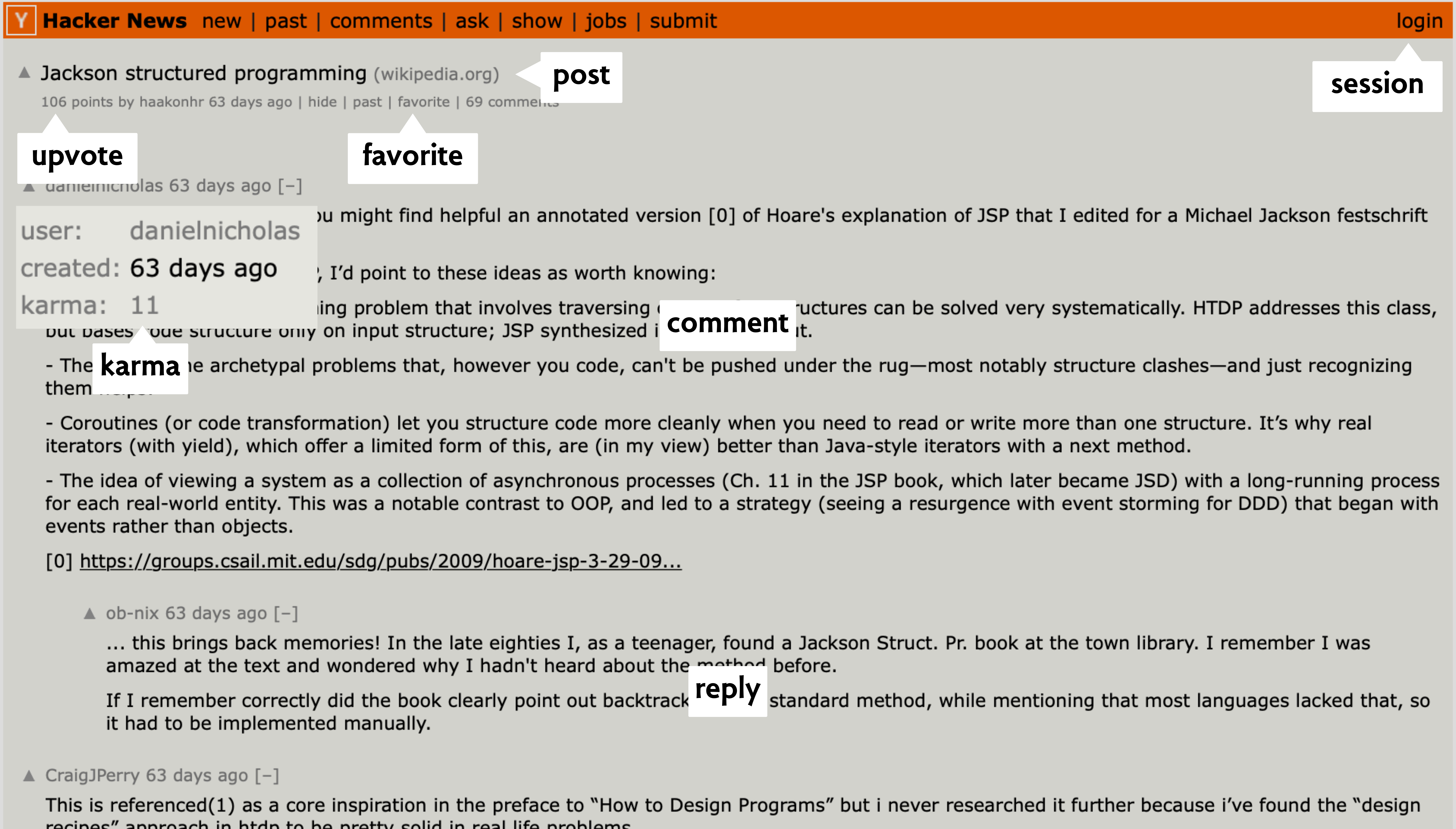
▲ ob-nix 63 days ago [–]

... 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



Y **Hacker News** new | past | comments | ask | show | jobs | submit login

▲ Jackson structured programming (wikipedia.org)

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

▲ danielnicholas 63 days ago [–]

user: danielnicholas
created: 63 days ago
karma: 11

u 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 ructures can be solved very systematically. HTDP addresses this class, but bases ode structure only on input structure; JSP synthesized i it.

- The e 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 before.

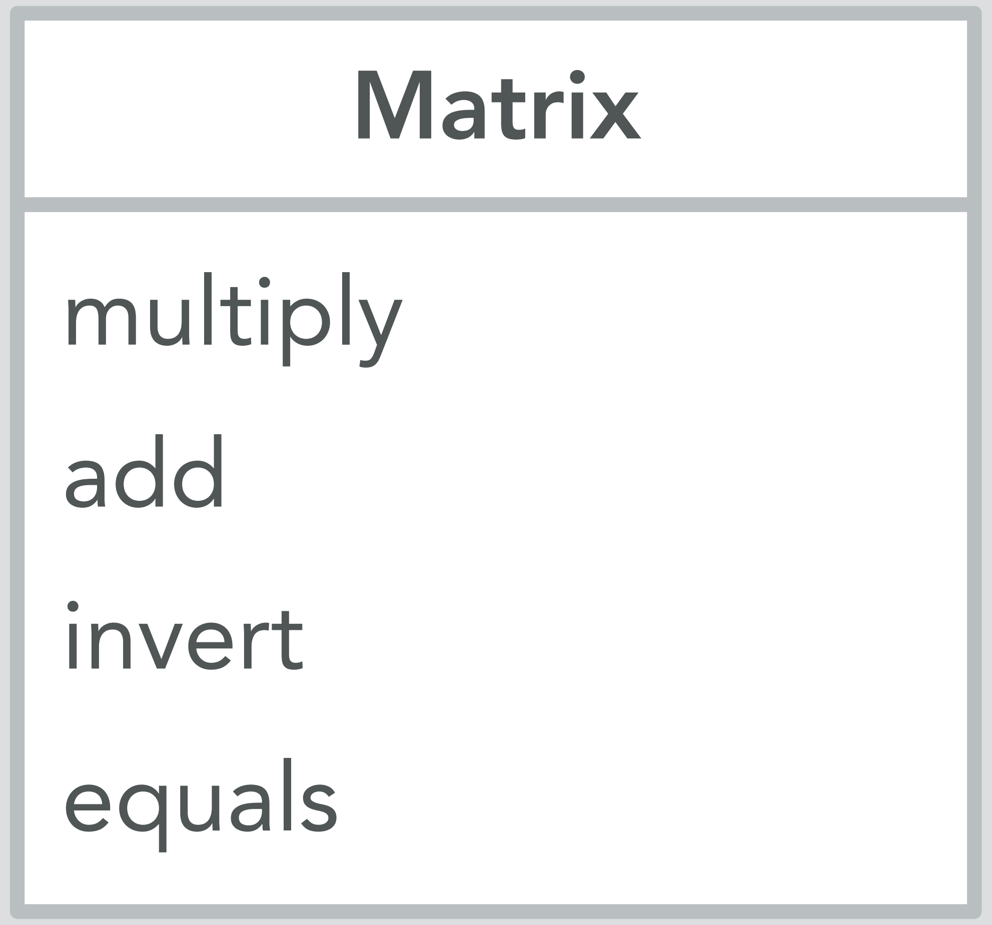
If I remember correctly did the book clearly point out backtrack standard method, while mentioning that most languages lacked that, so it had to be implemented manually.

▲ 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

but what’s a concept? three things it isn’t

abstract type, class/object



Matrix

multiply

add

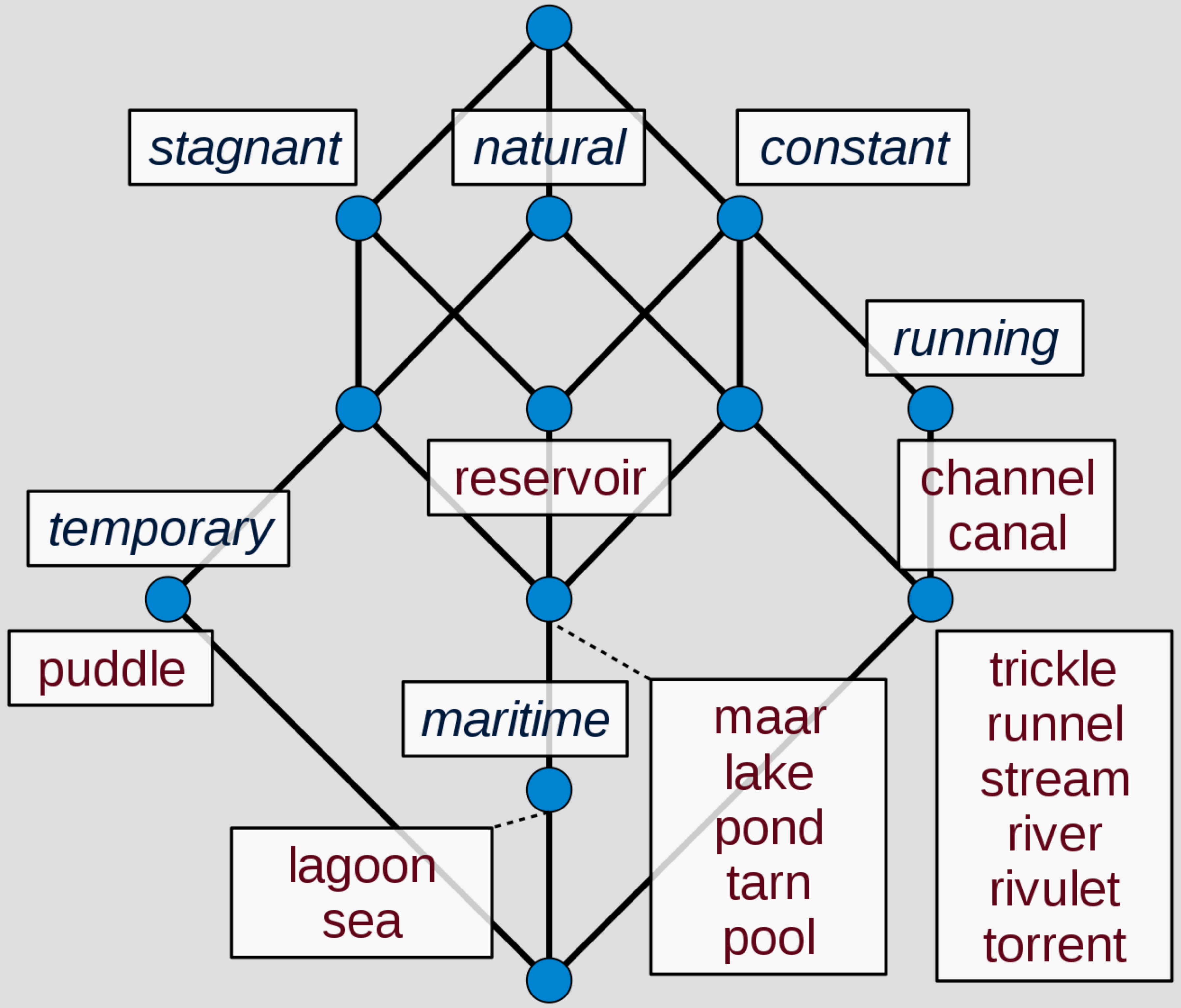
invert

equals

not limited to built-in types
encapsulate representation
defined by operations alone

what operations can
you do on an upvote?

concept lattice



stagnant

natural

constant

running

reservoir

channel
canal

temporary

puddle

maritime

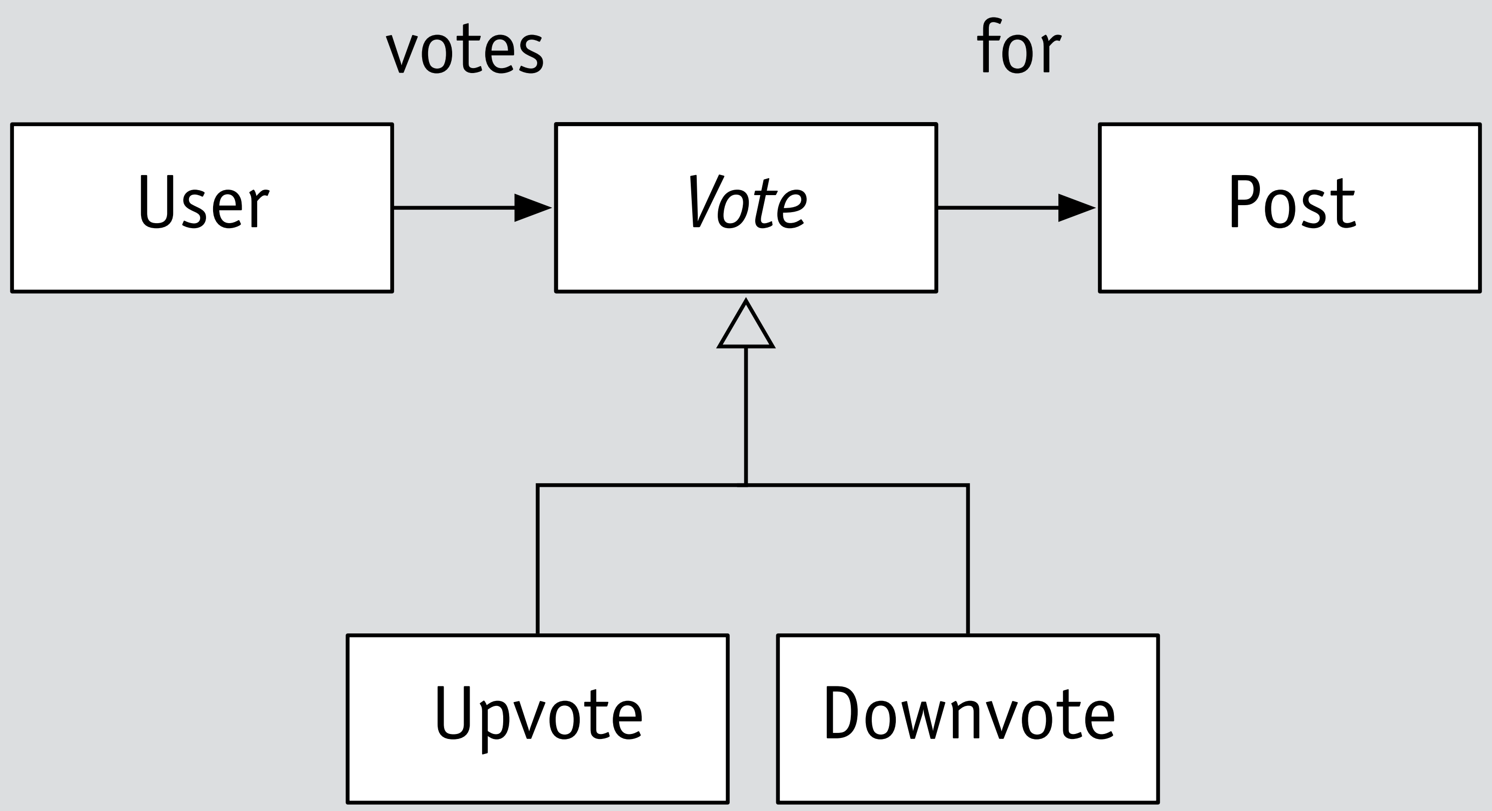
maar
lake
pond
tarn
pool

trickle
runnel
stream
river
rivulet
torrent

lagoon
sea

upvotes and downvotes
are votes and then what?

entity in data model



votes

for

User

Vote

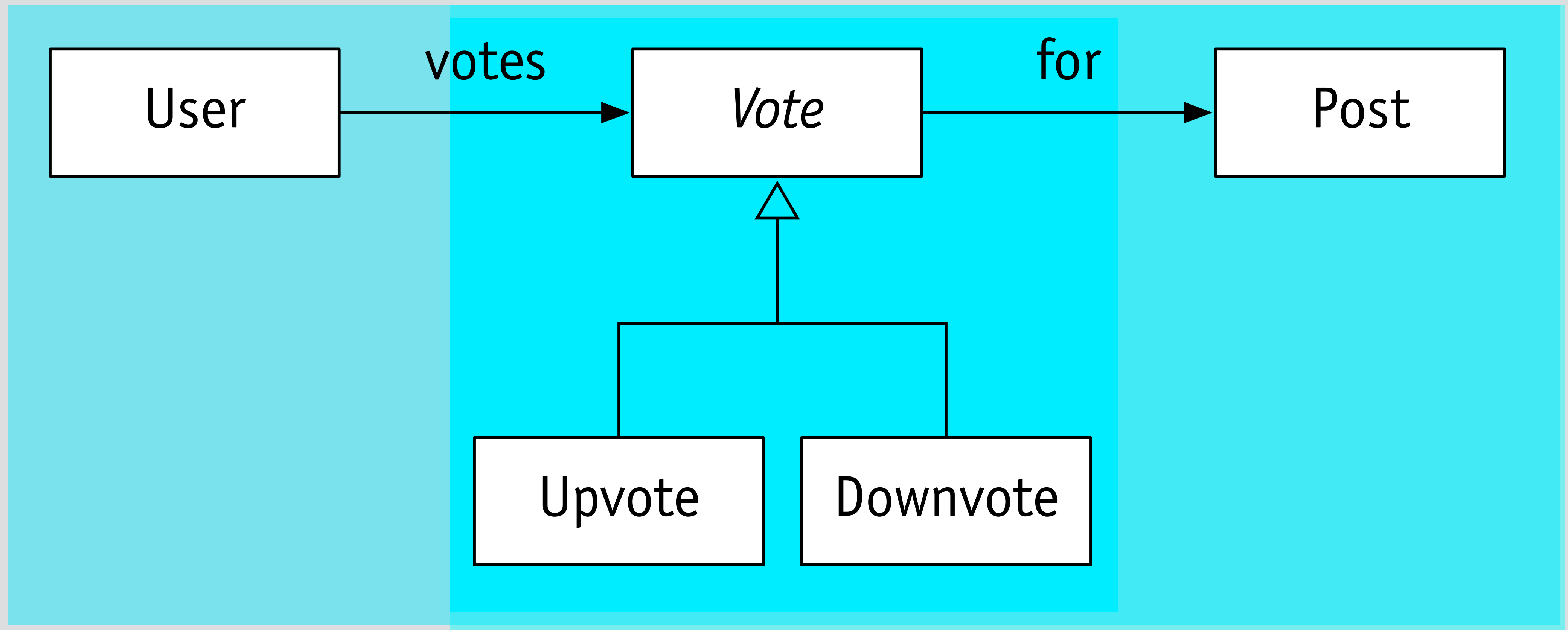
Post

Upvote

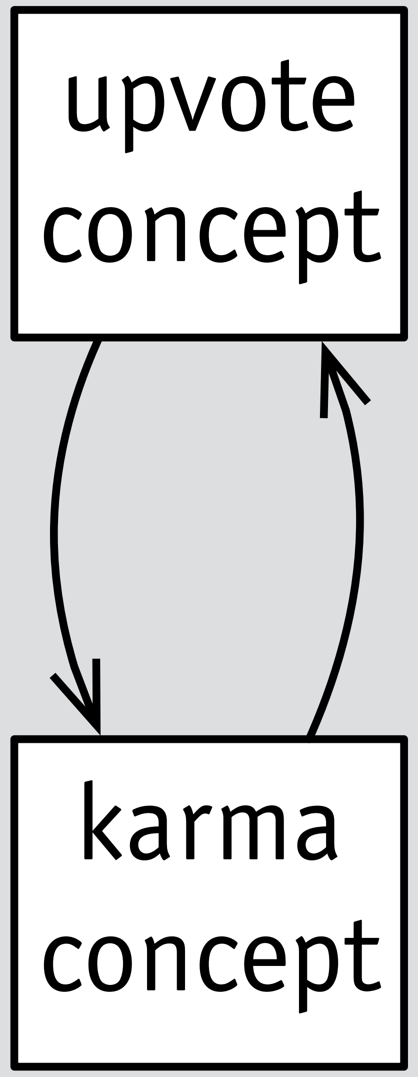
Downvote

but concept is in the
relationships, not the entities!

two more problems



where's the boundary?
if concept is in relationships too, which ones?
for upvote, are posts in the concept? users?



are concepts coupled?
eg, upvotes add to karma points
need karma points to downvote
so are these concepts dependent?

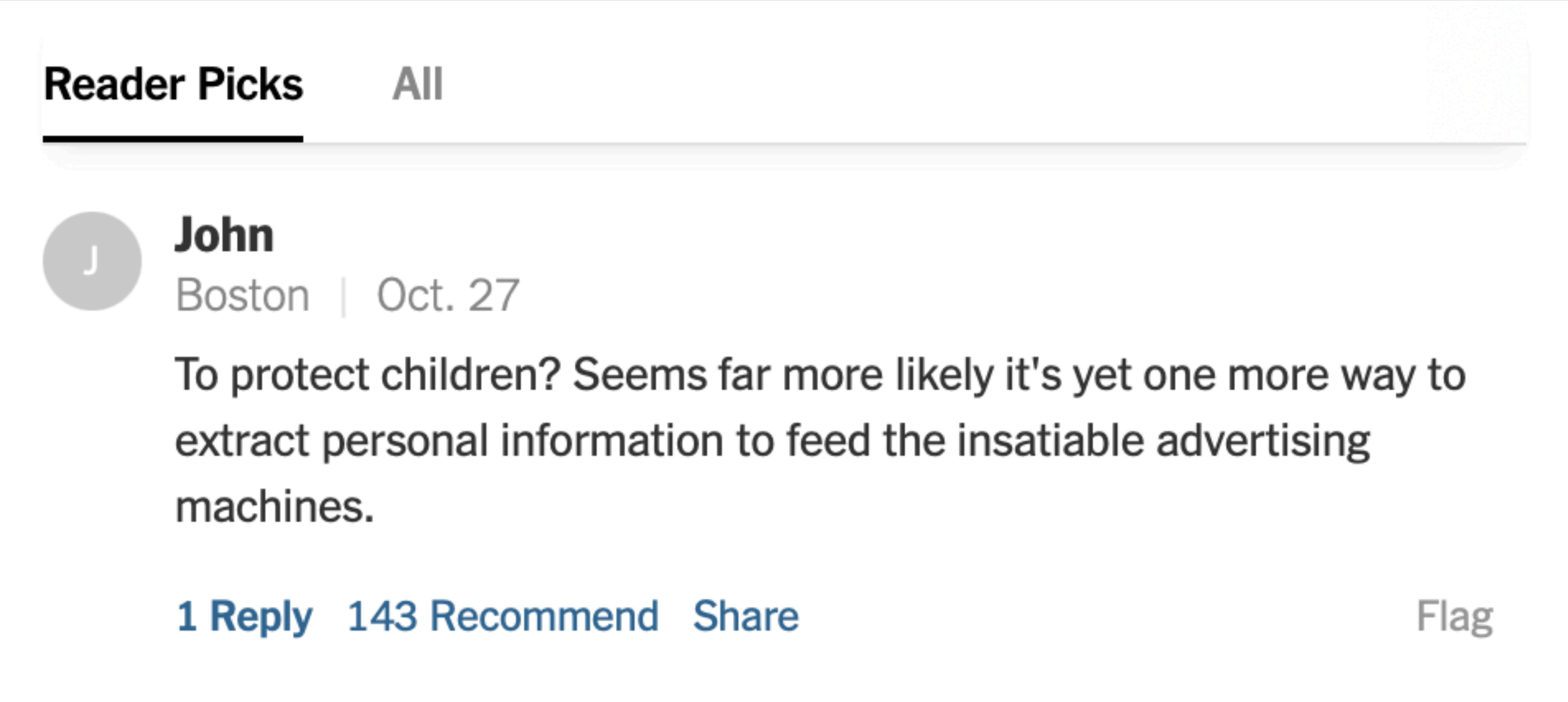
an example concept

finally!

a concept has a name

concept Upvote

same concept in HackerNews, NYTimes comment section, StackOverflow, etc



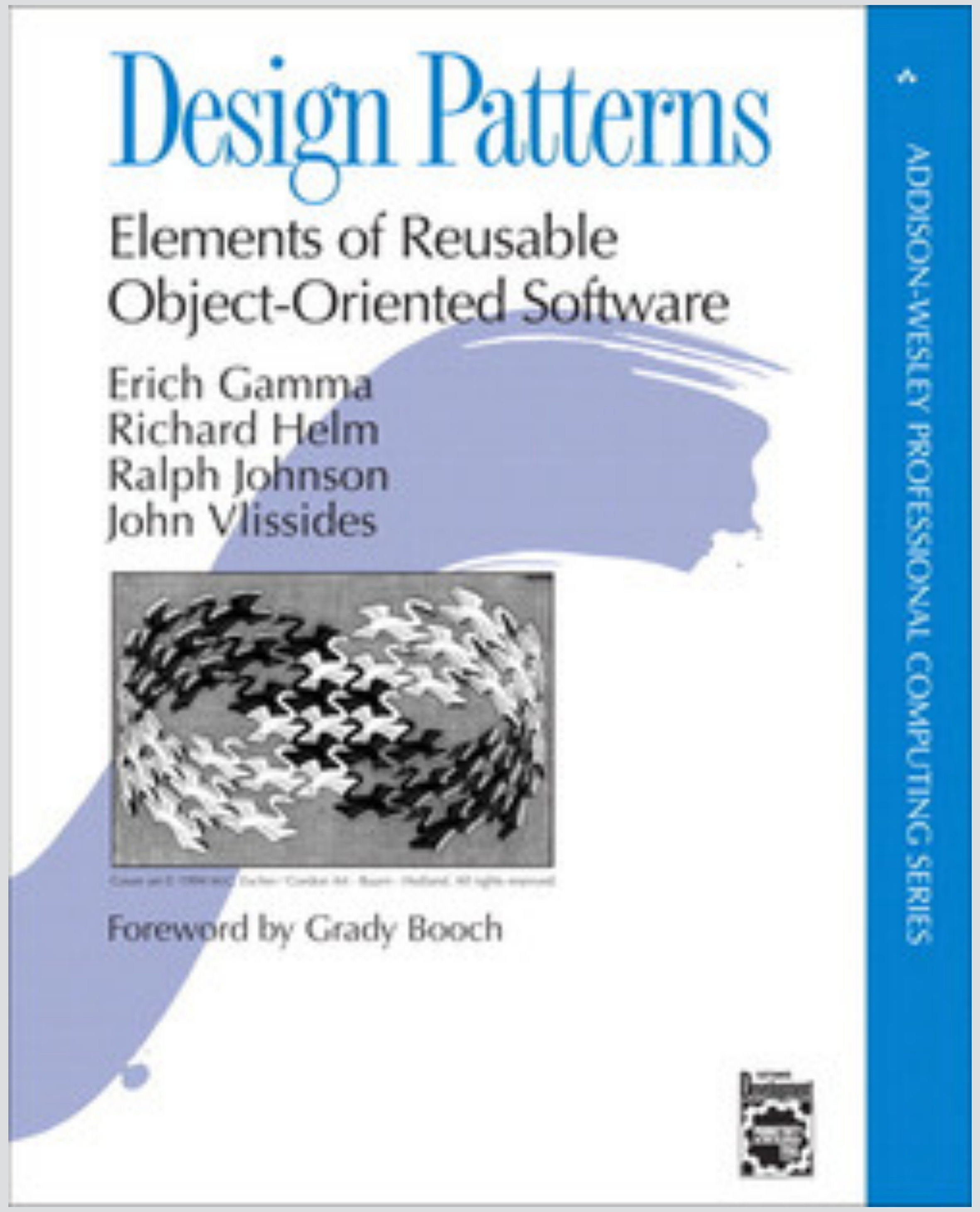
Reader Picks All

J **John**

Boston | Oct. 27

To protect children? Seems far more likely it's yet one more way to extract personal information to feed the insatiable advertising machines.

1 Reply 143 Recommend Share Flag

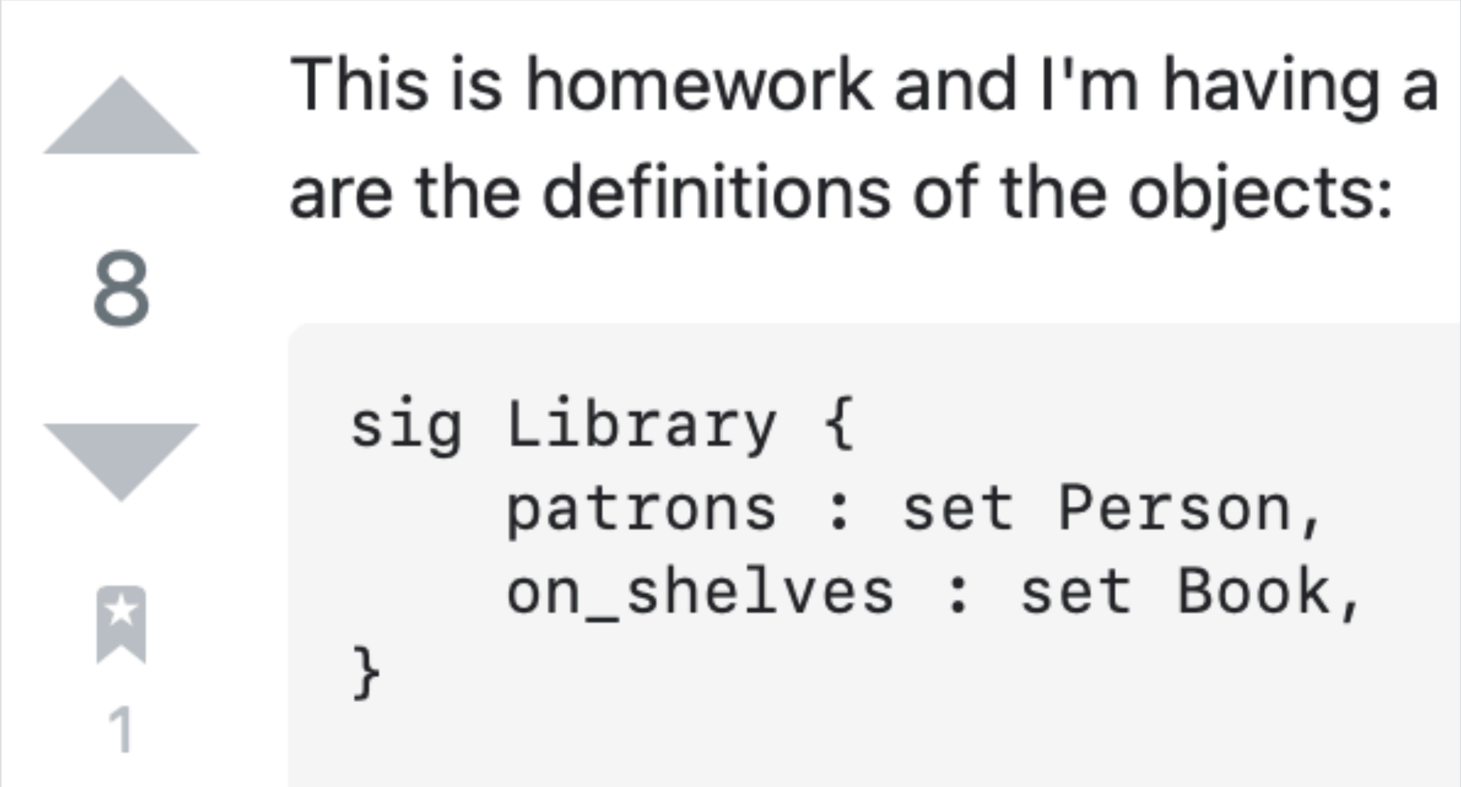

Design Patterns
Elements of Reusable Object-Oriented Software
Erich Gamma
Richard Helm
Ralph Johnson
John Vlissides
Foreword by Grady Booch
ADDISON-WESLEY PROFESSIONAL COMPUTING SERIES

we learned from the GOF patterns book how essential names are for sharing expertise

a concept has a purpose

concept Upvote

purpose rank items by popularity

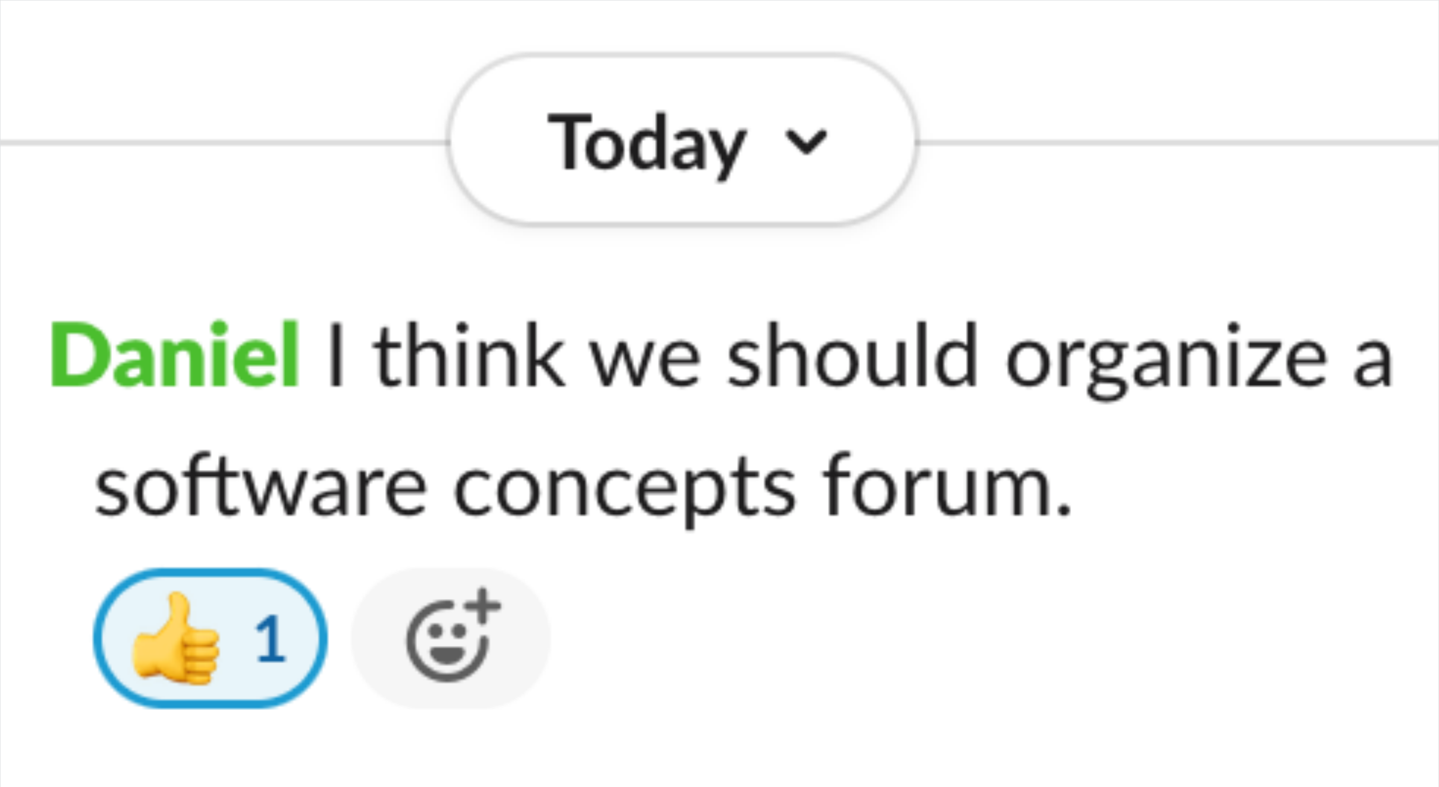


This is homework and I'm having a are the definitions of the objects:

```
sig Library {
    patrons : set Person,
    on_shelves : set Book,
}
```

concept Reaction

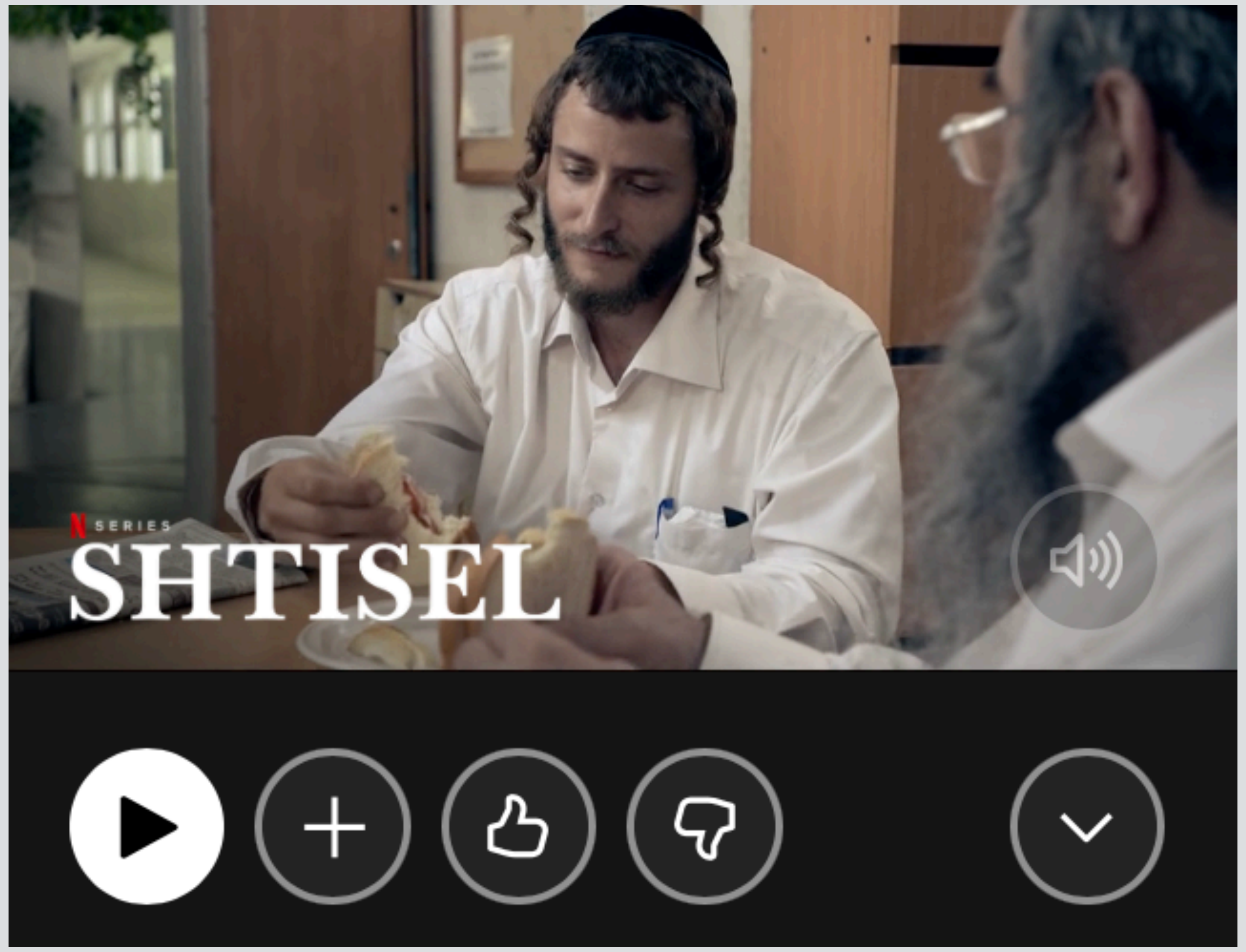
purpose send reactions to author



Daniel I think we should organize a software concepts forum.

concept Recommendation

purpose use prior likes to recommend



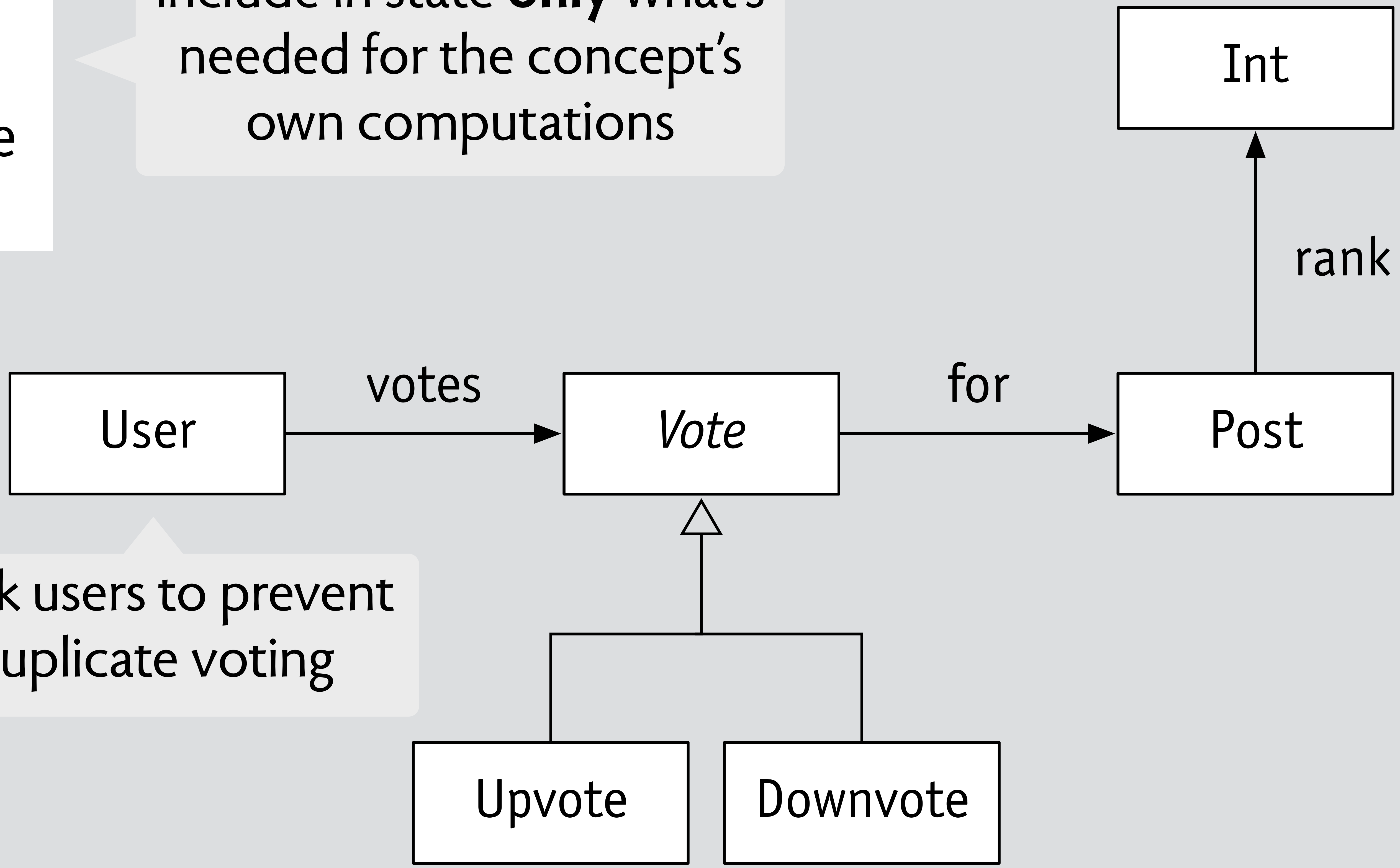
a concept has a **state**

concept Upvote

purpose rank items by popularity

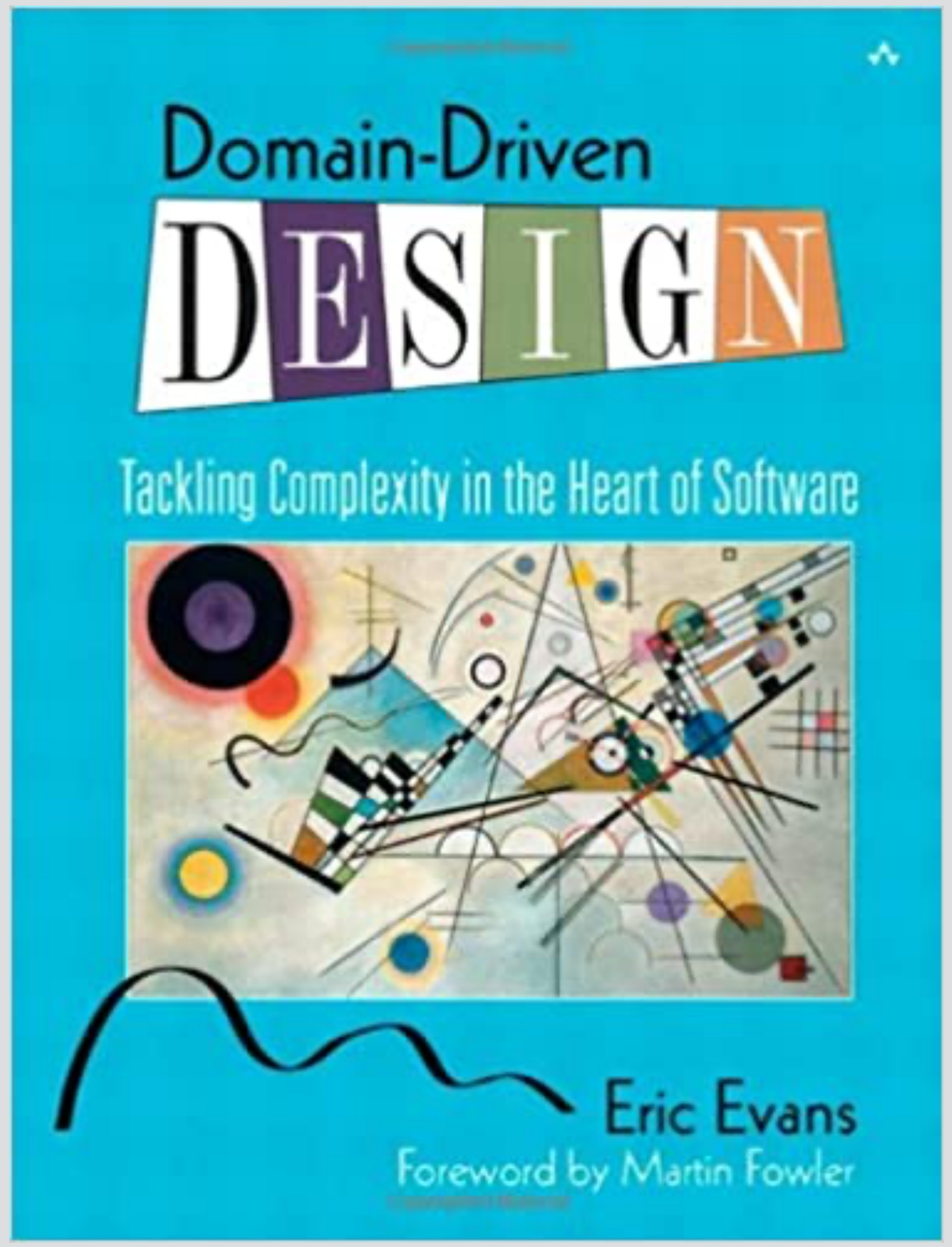
state
votes: User -> set Vote
for: Vote -> one Item
Upvote, Downvote: set Vote
rank: Item -> one Int

include in state **only** what's needed for the concept's own computations



User
votes
Vote
for
Post
rank
Int
Upvote
Downvote

track users to prevent duplicate voting


Domain-Driven
DESIGN
Tackling Complexity in the Heart of Software
Eric Evans
Foreword by Martin Fowler

like bounded context in DDD, but even more localized

a concept has actions

concept Upvote

purpose rank items by popularity

state
votes: User -> set Vote
for: Vote -> one Item
Upvote, Downvote: set Vote
rank: Item -> one Int

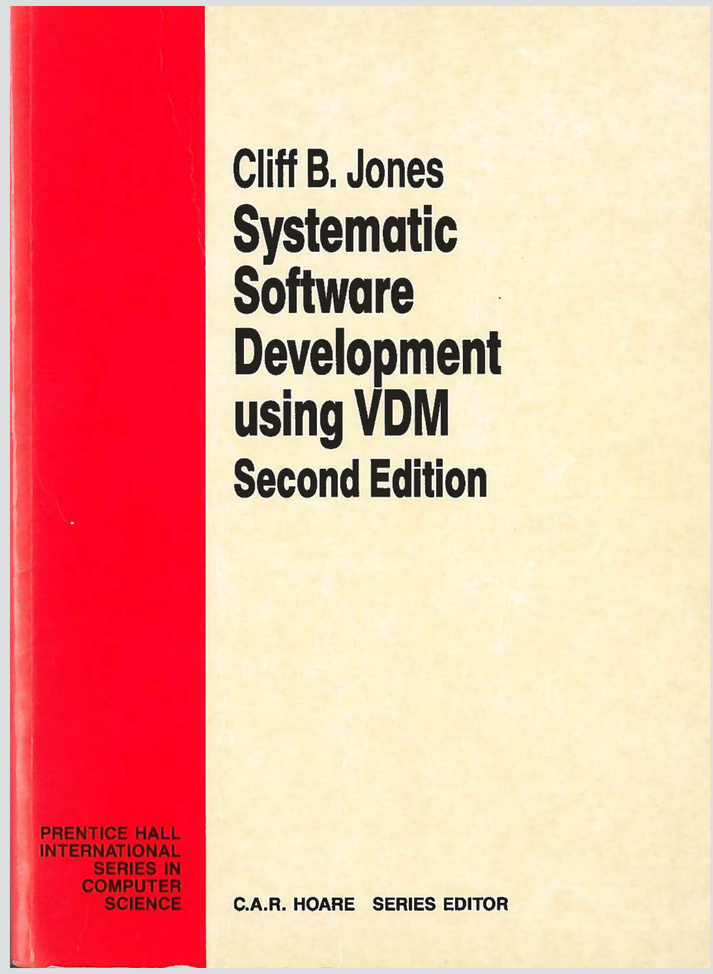
actions
upvote (u: User, i: Item)
downvote (u: User, i: Item)
unvote (u: User, i: Item)

need **unvote** in HackerNews (eg) since only high karma users can downvote

actions capture the concept **behavior in full**

downvote (i: Item, u: User)

```
  // no existing Downvote for i in u.votes
  // remove any Upvote for i from u.votes
  // add a Downvote for i in u.votes
  // update i.rank ...
```



Cliff B. Jones
Systematic Software Development using VDM
Second Edition

PRENTICE HALL INTERNATIONAL SERIES IN COMPUTER SCIENCE

C.A.R. HOARE SERIES EDITOR

succinct specification as actions on states
VDM (1986)
Z (1992)
Larch (1993)
Event-B (2006)
Alloy (2006)

a concept has an operational principle

concept Upvote

purpose rank items by popularity

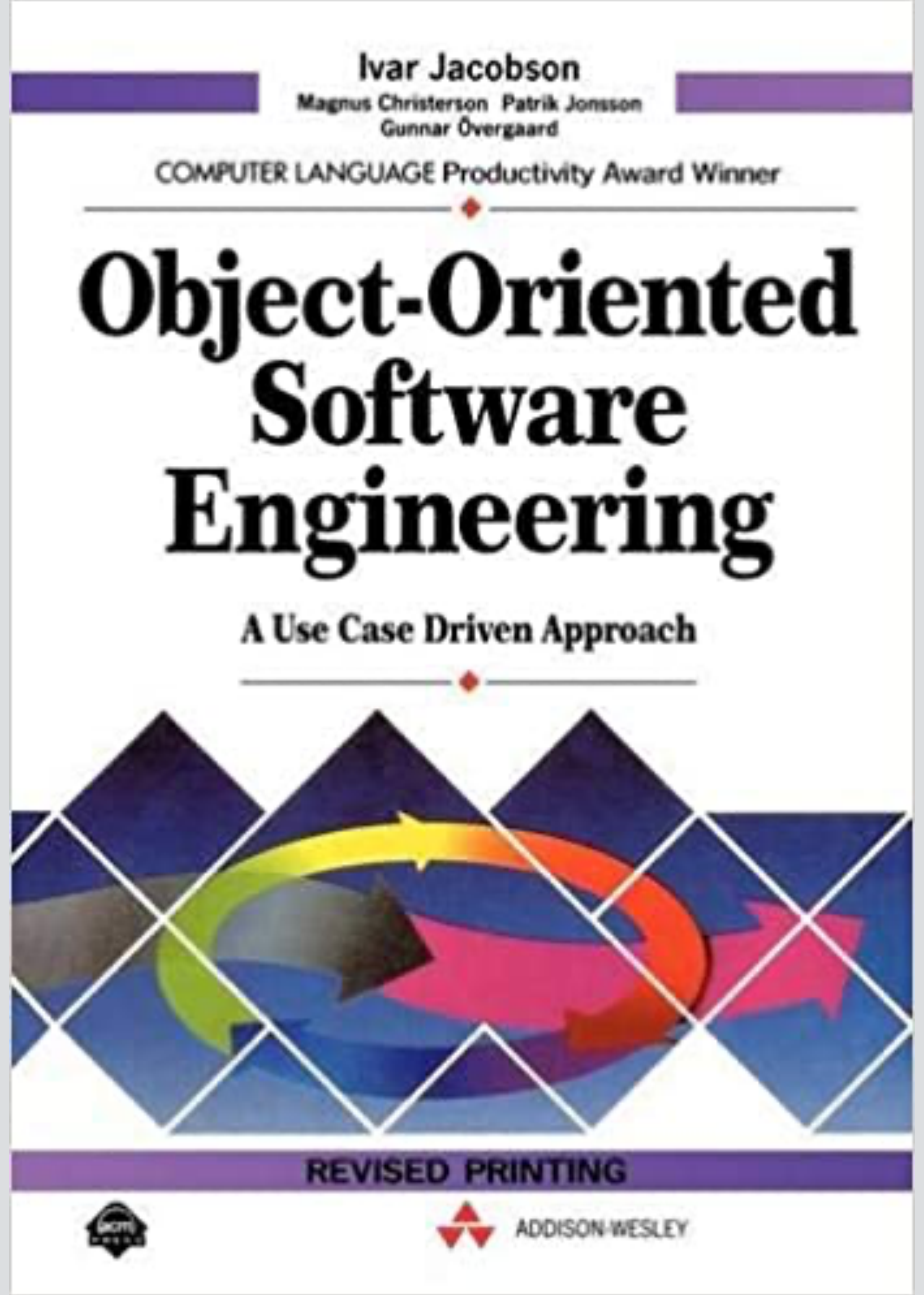
state
votes: User -> set Vote
for: Vote -> one Item
Upvote, Downvote: set Vote
rank: Item -> one Int

actions
upvote (u: User, i: Item)
downvote (u: User, i: Item)
unvote (u: User, i: Item)

operational principle
after sequence of upvote (…)’s,
if item i has more than item j,
then i.rank < j.rank

an archetypal scenario that captures the essence of how the concept fulfills its purpose

Michael Polanyi


Ivar Jacobson
Magnus Christerson Patrik Jonsson
Gunnar Övergaard
COMPUTER LANGUAGE Productivity Award Winner
Object-Oriented Software Engineering
A Use Case Driven Approach
REVISED PRINTING
ADDISON-WESLEY

use cases are actually more like actions

a concept handbook entry

related concepts
Recommendation, Reaction, ...

design variants
downvote as unvote
use age in ranking
weigh downvotes more

known issues
preventing double votes
(require login, use IP address, save cookie)
saving storage space
(freeze old posts and from user info)

typical uses
social media posts
comments on articles
Q&A responses

often used with
Karma, Session, ...

how to make an app

concept composition

how to extend behavior?

concept Upvote

actions
upvote (u: User, i: Item)
downvote (u: User, i: Item)
unvote (u: User, i: Item)

suppose I want this behavior:
you can't downvote an item
until you've received
N upvotes on your own items

define a new concept!
a hint: not just used by upvote

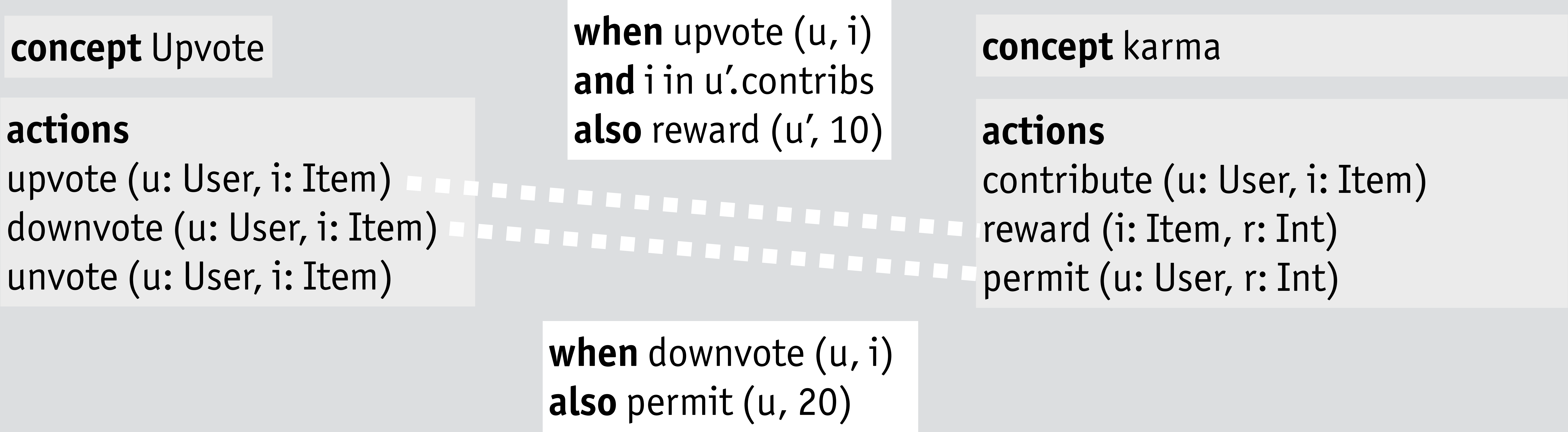
concept Karma

purpose privilege good users

state
karma: User -> one Int
contribs: User -> set Item

actions
contribute (u: User, i: Item)
reward (i: Item, r: Int)
permit (u: User, r: Int)

operational principle
allow permit (u, R) if sum of
rewards for u.items ≥ R



concept Upvote

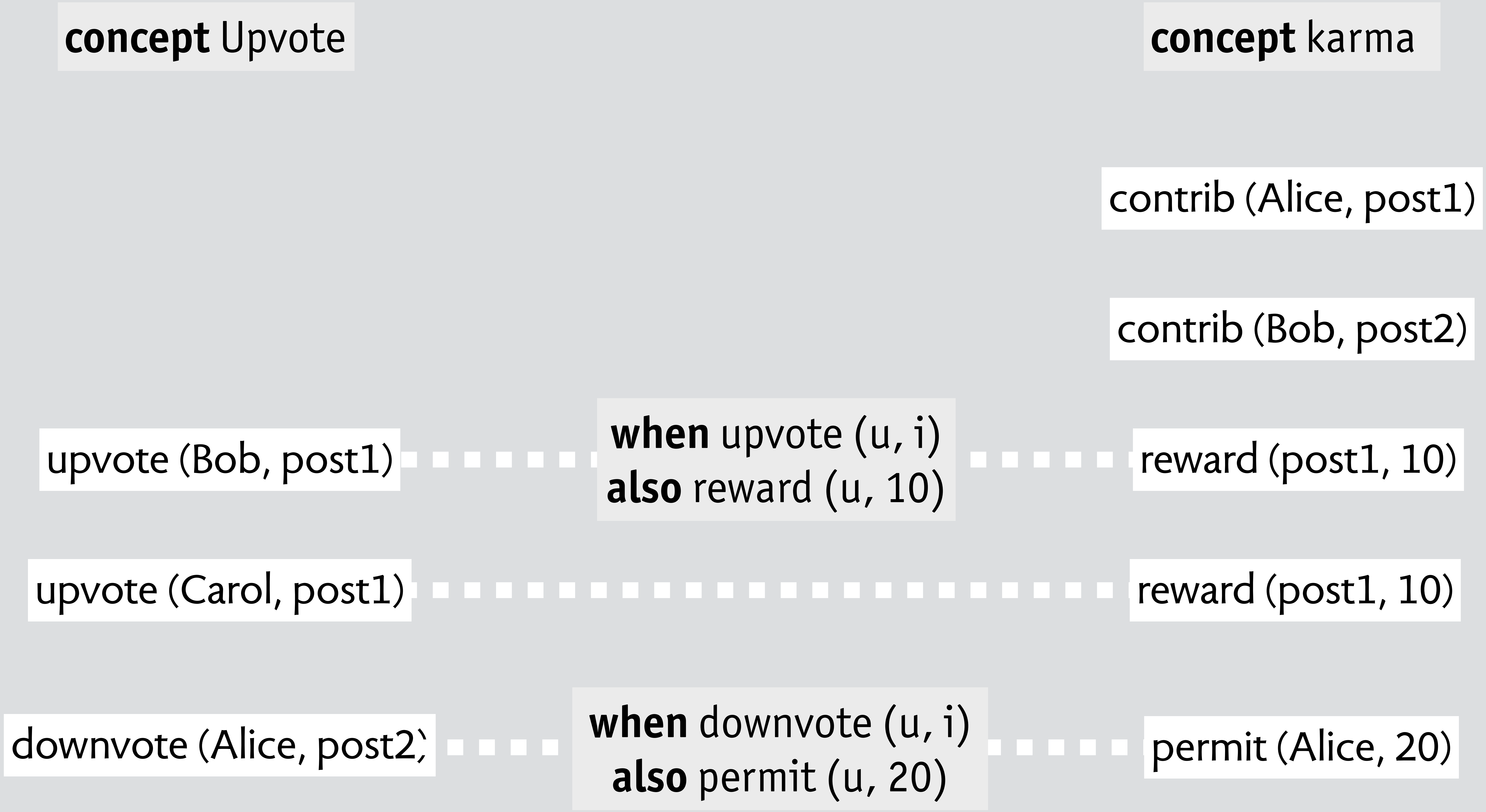
actions
upvote (u: User, i: Item)
downvote (u: User, i: Item)
unvote (u: User, i: Item)

when upvote (u, i)
and i in u’.contribs
also reward (u’, 10)

when downvote (u, i)
also permit (u, 20)

concept karma

actions
contribute (u: User, i: Item)
reward (i: Item, r: Int)
permit (u: User, r: Int)



concept Upvote

concept karma

contrib (Alice, post1)

contrib (Bob, post2)

upvote (Bob, post1)

when upvote (u, i)
also reward (u, 10)

reward (post1, 10)

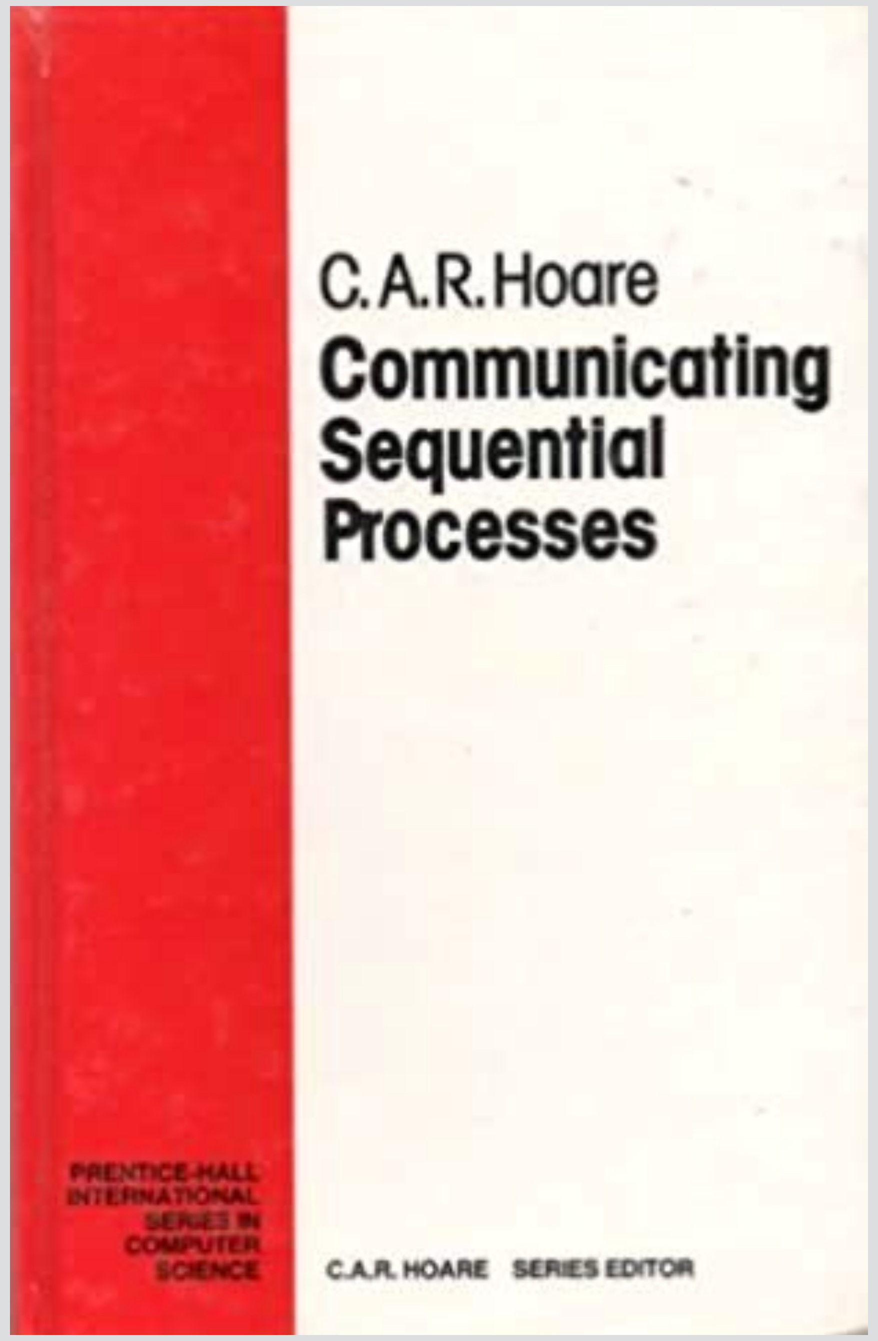
upvote (Carol, post1)

reward (post1, 10)

downvote (Alice, post2)

when downvote (u, i)
also permit (u, 20)

permit (Alice, 20)


C.A.R.Hoare
Communicating Sequential Processes
PRENTICE-HALL INTERNATIONAL SERIES IN COMPUTER SCIENCE
C.A.R. HOARE SERIES EDITOR

composition uses
event sync from
Hoare's CSP

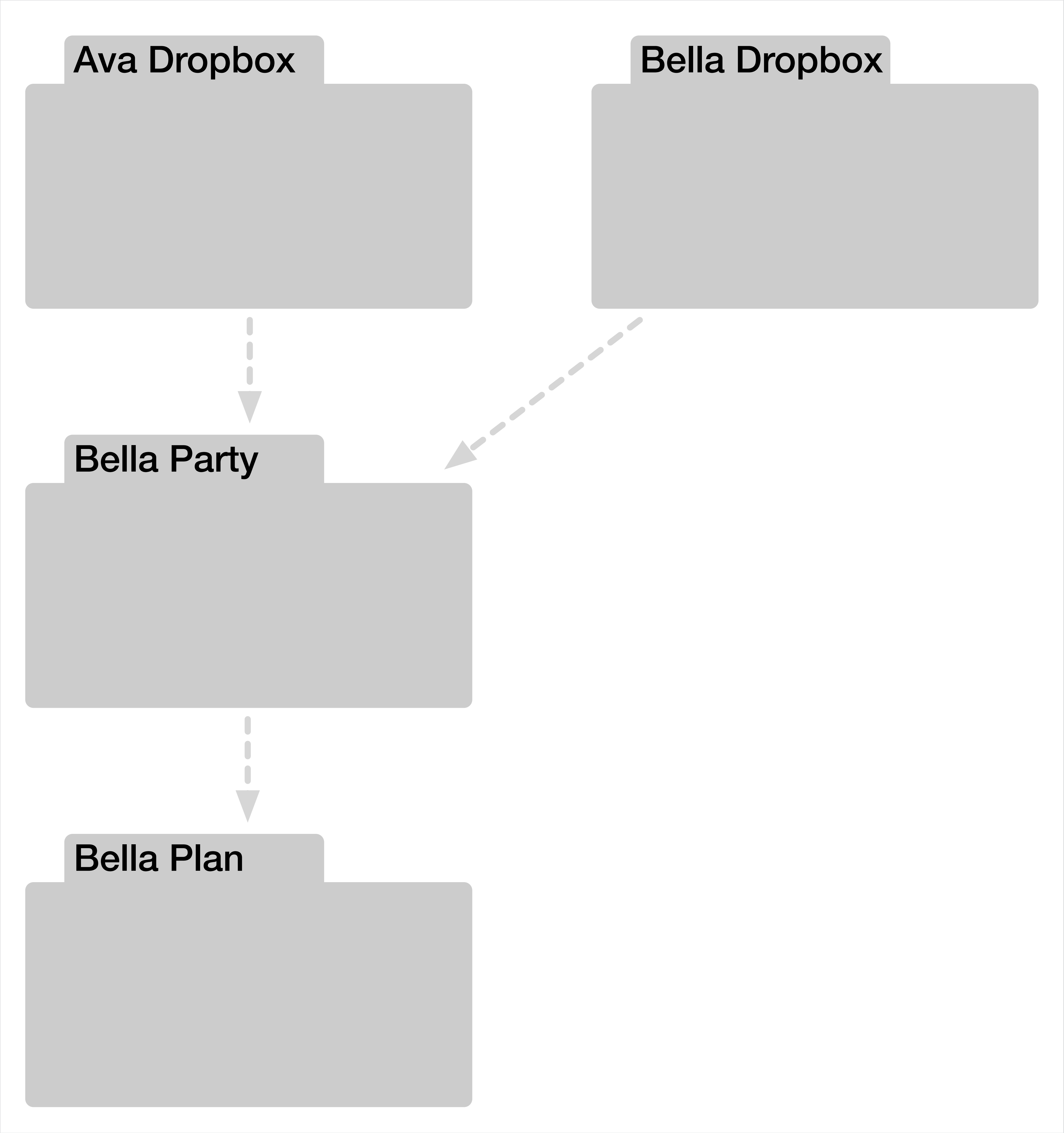
no concept coupling
concepts preserve properties

resolving the puzzles

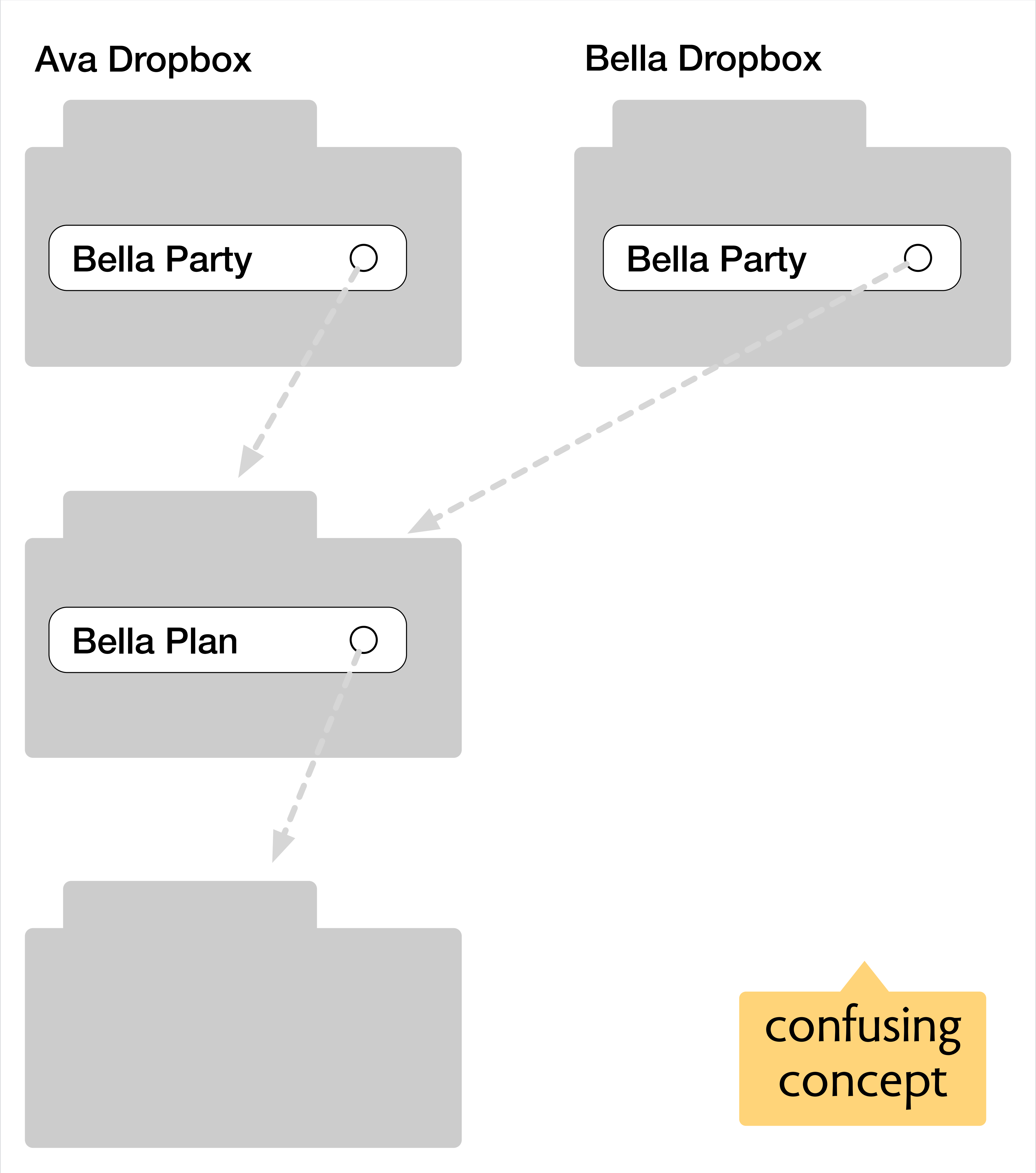
concept design in action

#1: dropbox

a troubled concept



how many users believe Dropbox is structured



how Dropbox is actually structured

concept metadata

purpose tag items with properties for lookup

state

properties: Item -> set Property
key: Property -> one Key
val: Property -> one Val

concept folder

purpose organize items into disjoint categories

state

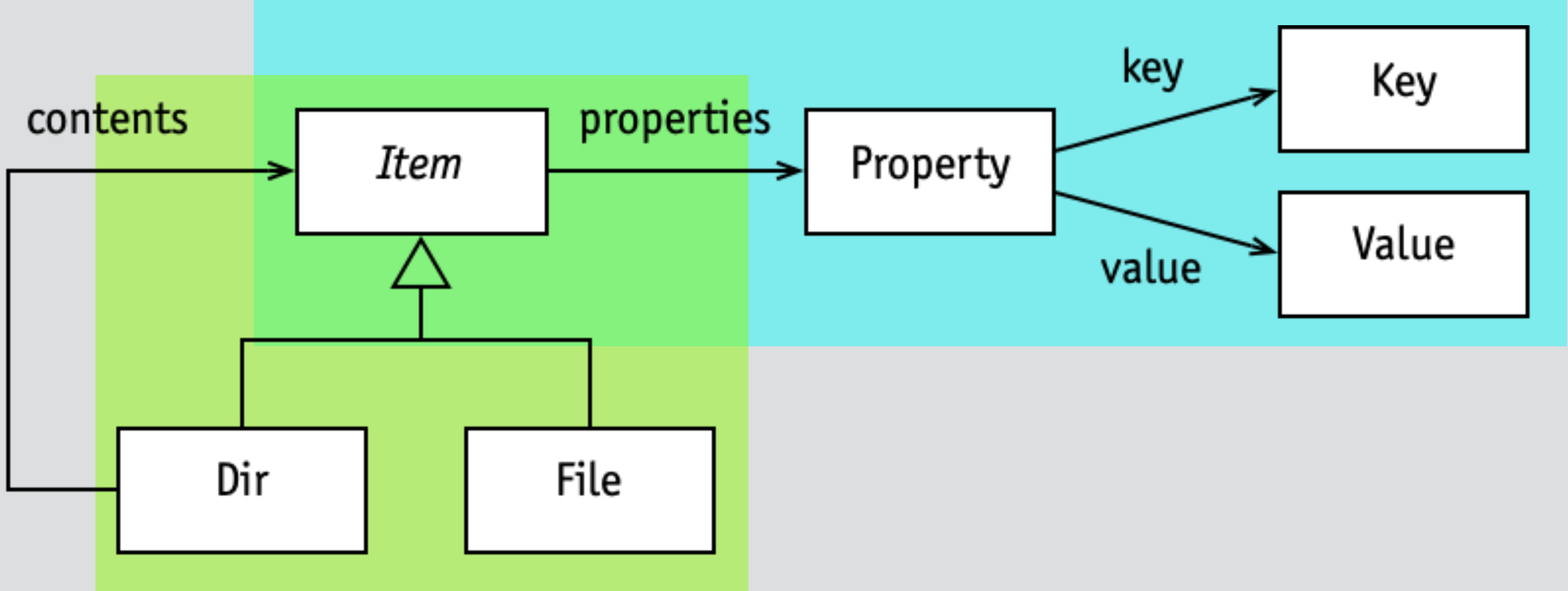
contents: Folder -> set (Folder + Item)
root: one Folder

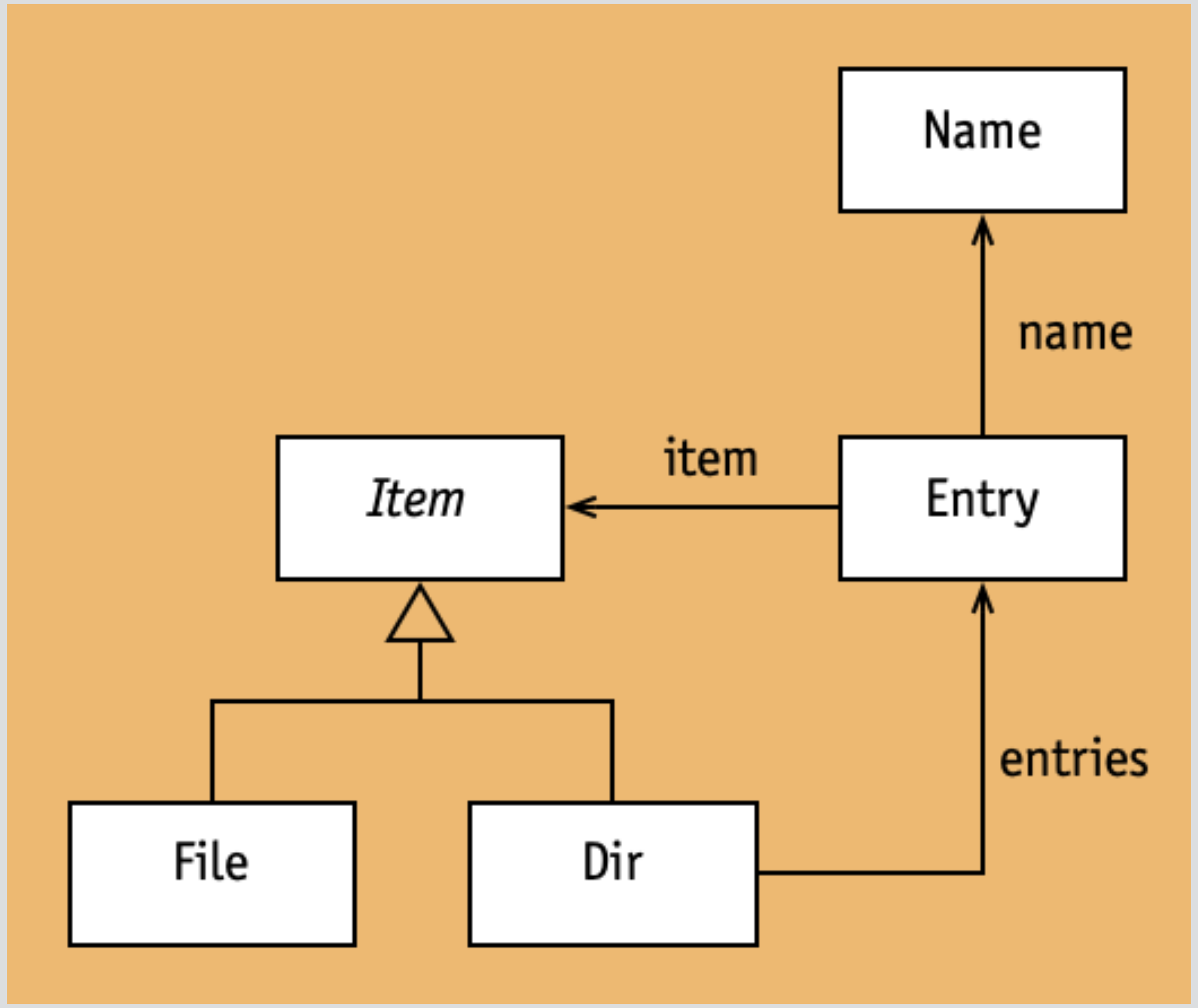
concept unix directory

purpose organize items into overlapping categories

state

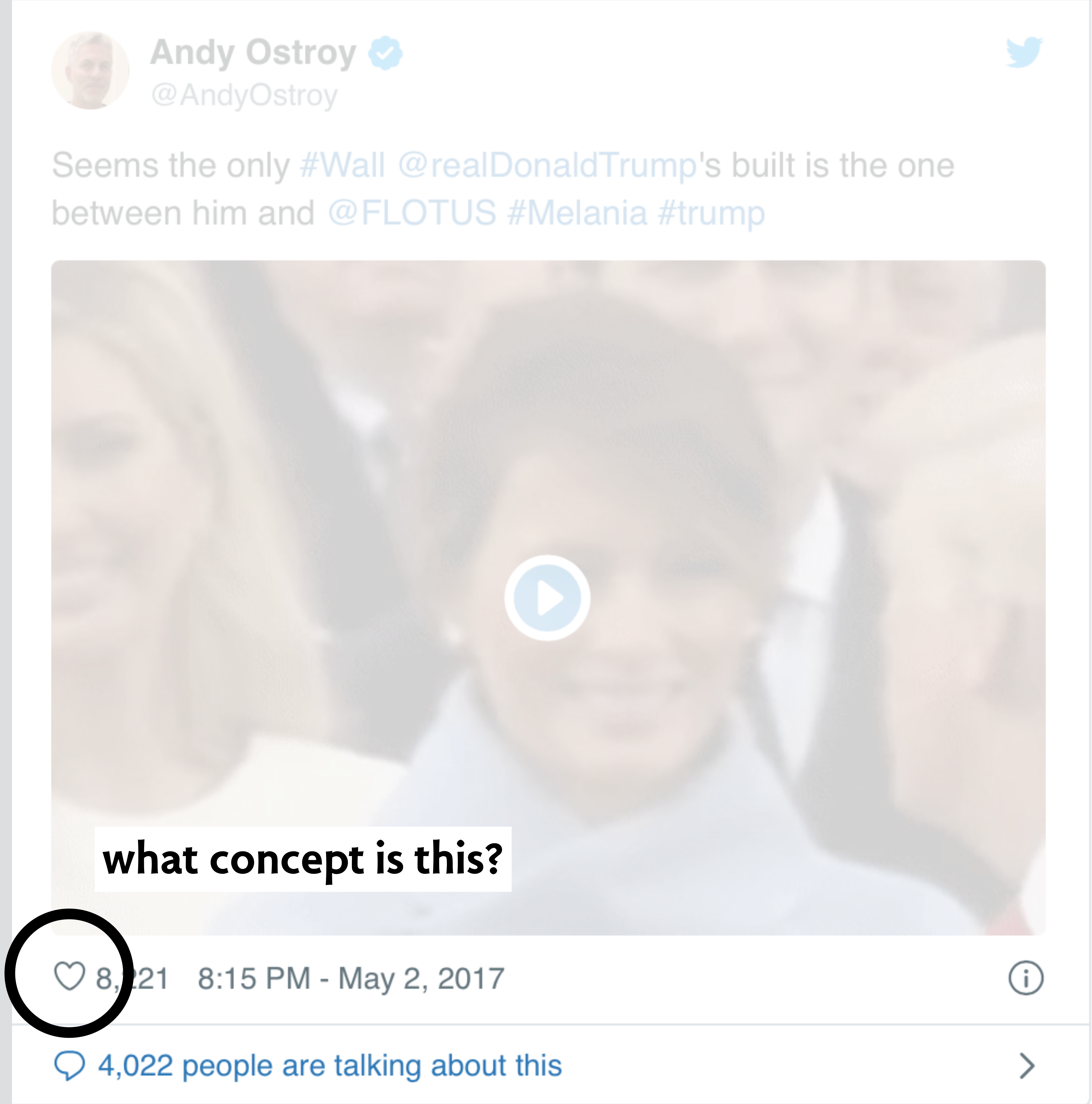
entries: Dir -> set Entry
name: Entry -> one Name
item: Entry -> one (File + Dir)





#2: twitter

a missing concept



Andy Ostroy
@AndyOstroy

Seems the only #Wall @realDonaldTrump's built is the one between him and @FLOTUS #Melania #trump

what concept is this?

♡ 8,?21 8:15 PM - May 2, 2017

4,022 people are talking about this

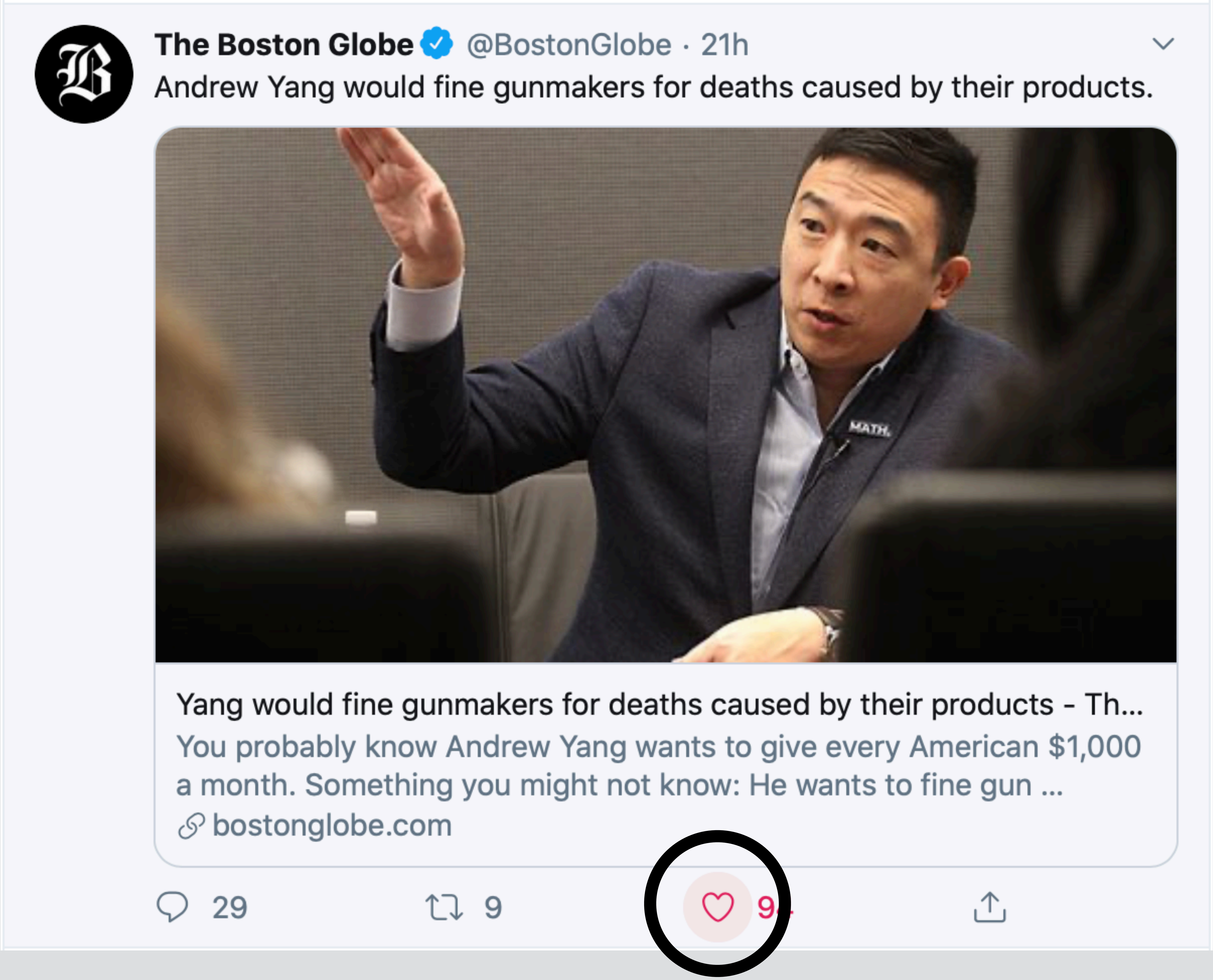
concept Upvote
purpose rank items by popularity

?

concept Bookmark
purpose save items to revisit

missing concept!

twitter adds a new concept (2018)



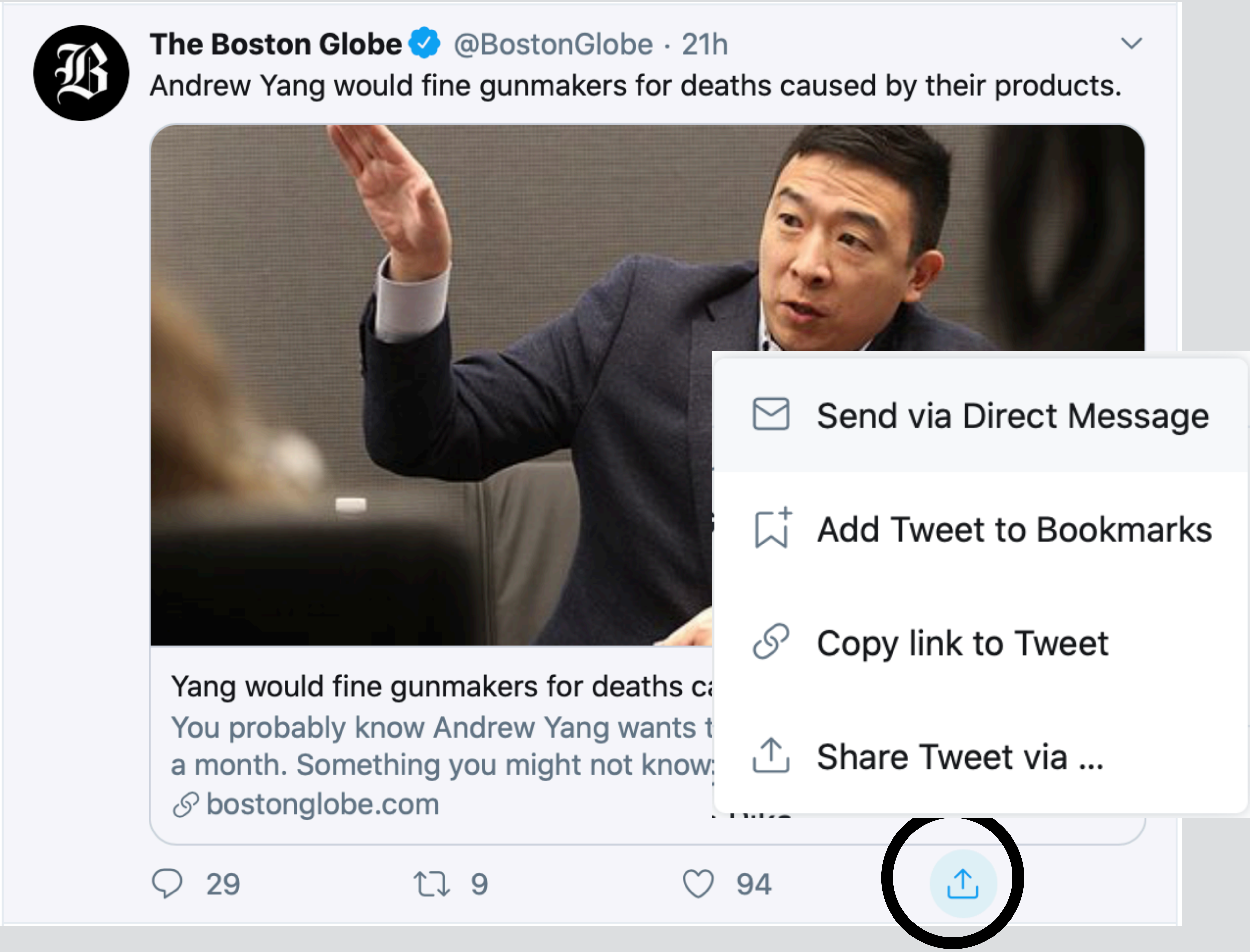
The Boston Globe @BostonGlobe · 21h

Andrew Yang would fine gunmakers for deaths caused by their products.

Yang would fine gunmakers for deaths caused by their products - Th...

You probably know Andrew Yang wants to give every American $1,000 a month. Something you might not know: He wants to fine gun ...

bostonglobe.com



The Boston Globe @BostonGlobe · 21h

Andrew Yang would fine gunmakers for deaths caused by their products.

Yang would fine gunmakers for deaths ca

You probably know Andrew Yang wants t

a month. Something you might not know

bostonglobe.com

concept Upvote

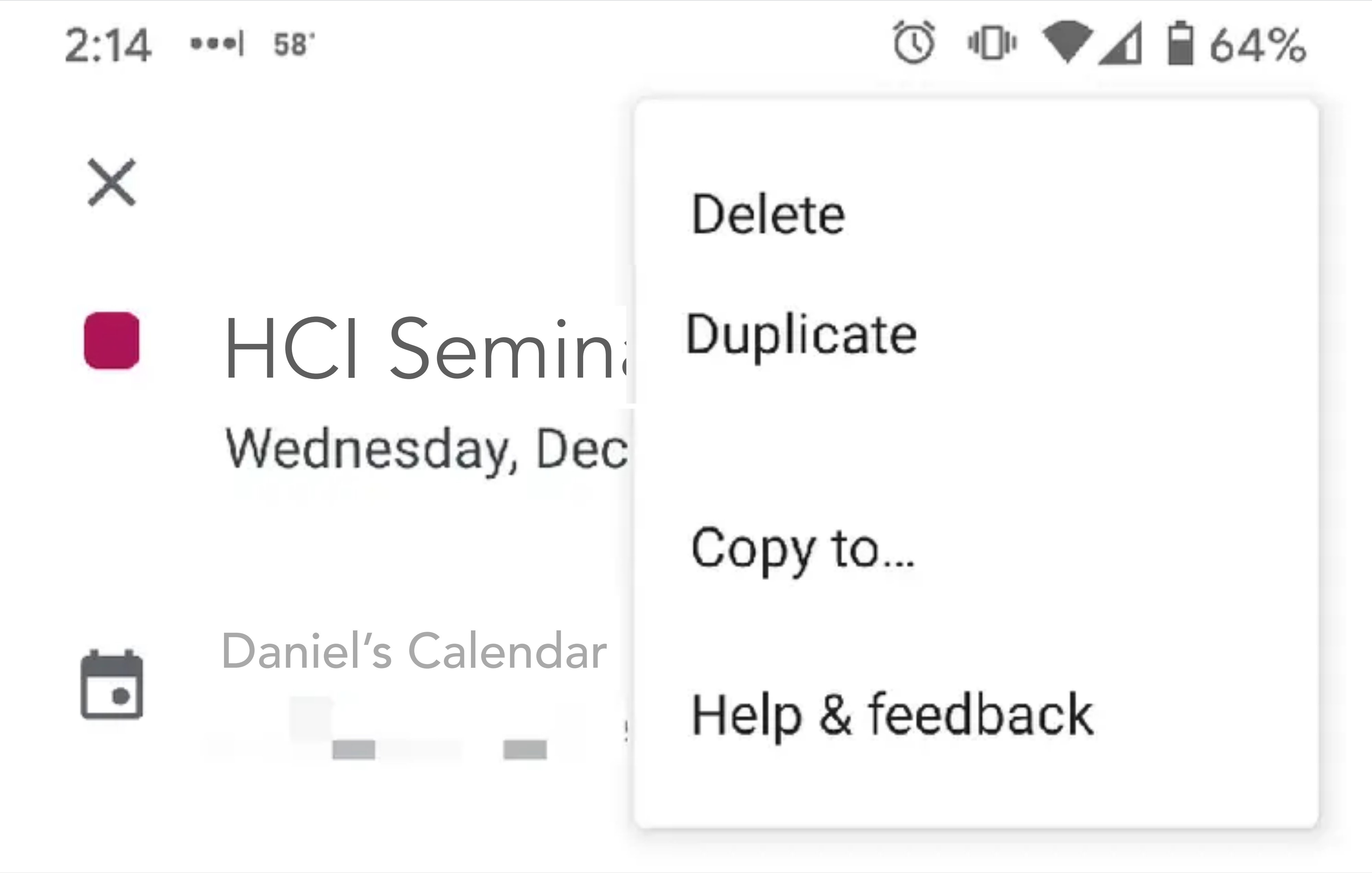
purpose rank items by popularity

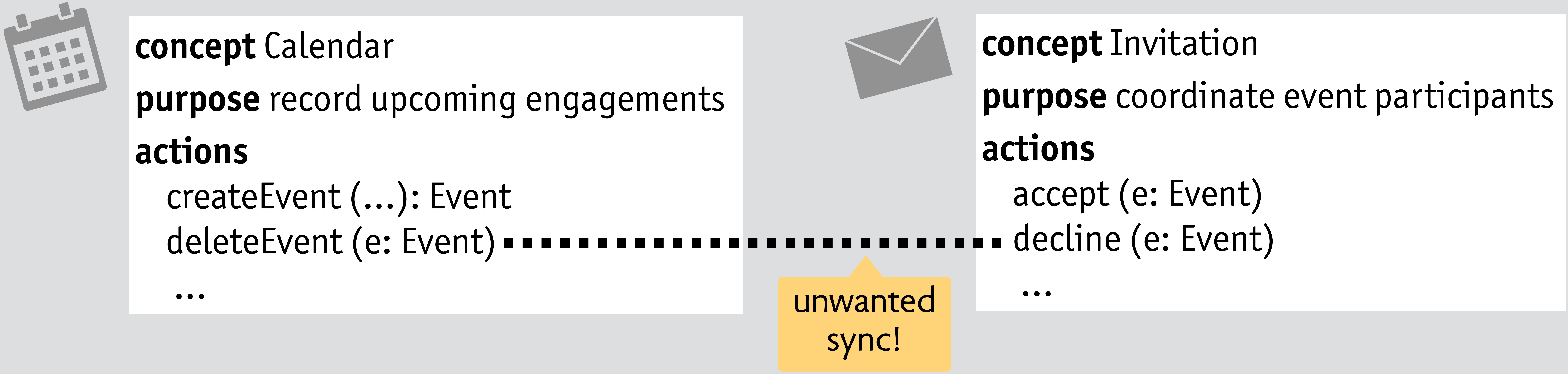
concept Bookmark

purpose save items to revisit

#3: Google Calendar

a bad synchronization





concept Calendar

purpose record upcoming engagements

actions

createEvent (...): Event

deleteEvent (e: Event)

...

concept Invitation

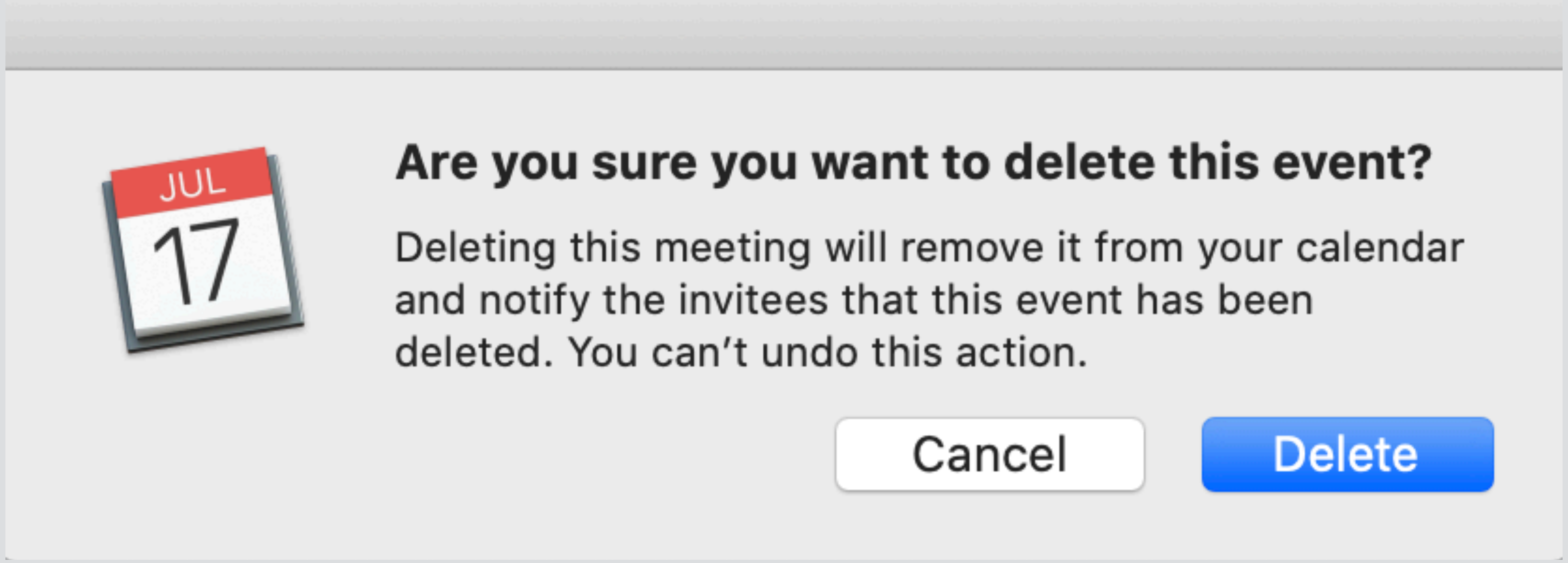
purpose coordinate event participants

actions

accept (e: Event)

decline (e: Event)

...

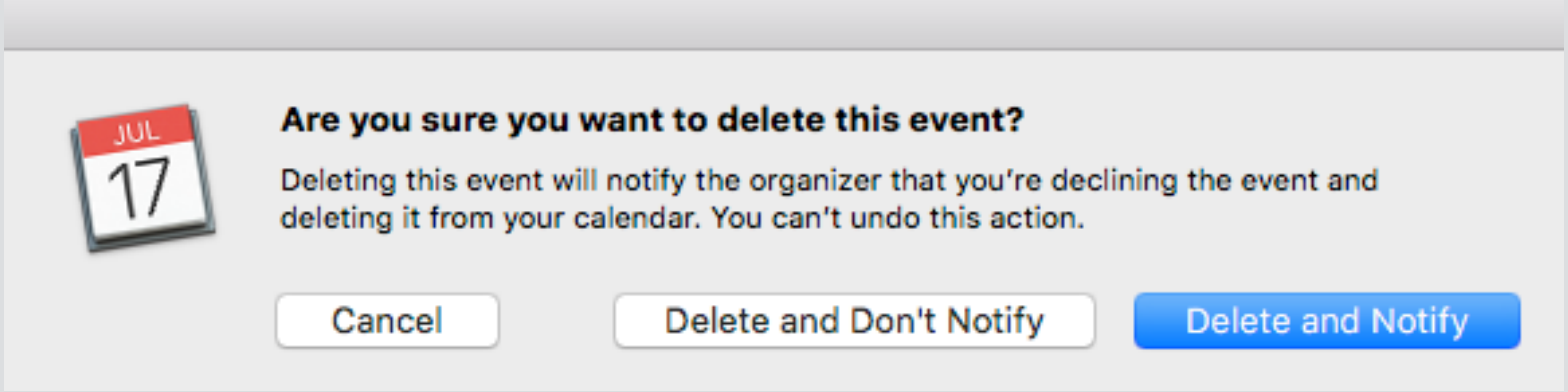


Are you sure you want to delete this event?

Deleting this meeting will remove it from your calendar and notify the invitees that this event has been deleted. You can't undo this action.

a long time problem in iCal too

how to delete spam calendar events?



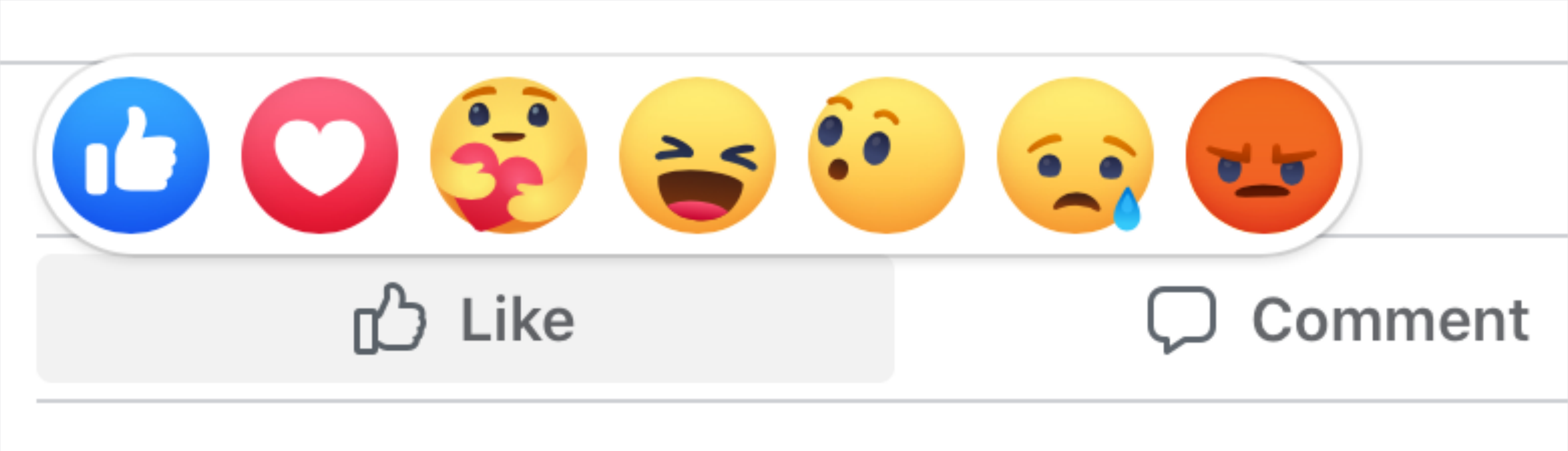
Are you sure you want to delete this event?

Deleting this event will notify the organizer that you're declining the event and deleting it from your calendar. You can't undo this action.

resolution to design problem

make sync optional

should facebook concepts be desynchronized too?



concept Upvote

purpose rank items by popularity

actions

  upvote (u: User, i: Item)

  …

concept Reaction

purpose convey emotion to author

actions

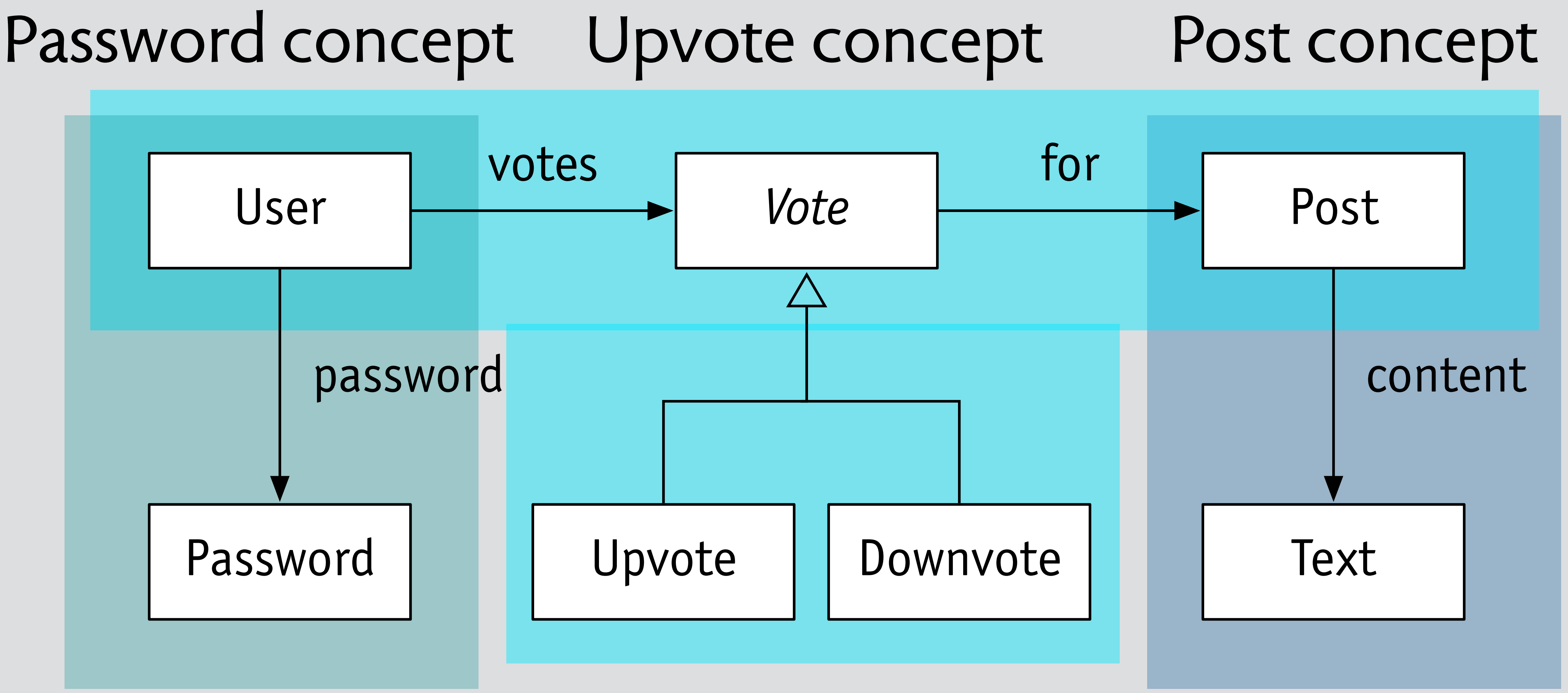
  reactAngry (u: User, i: Item)

  …

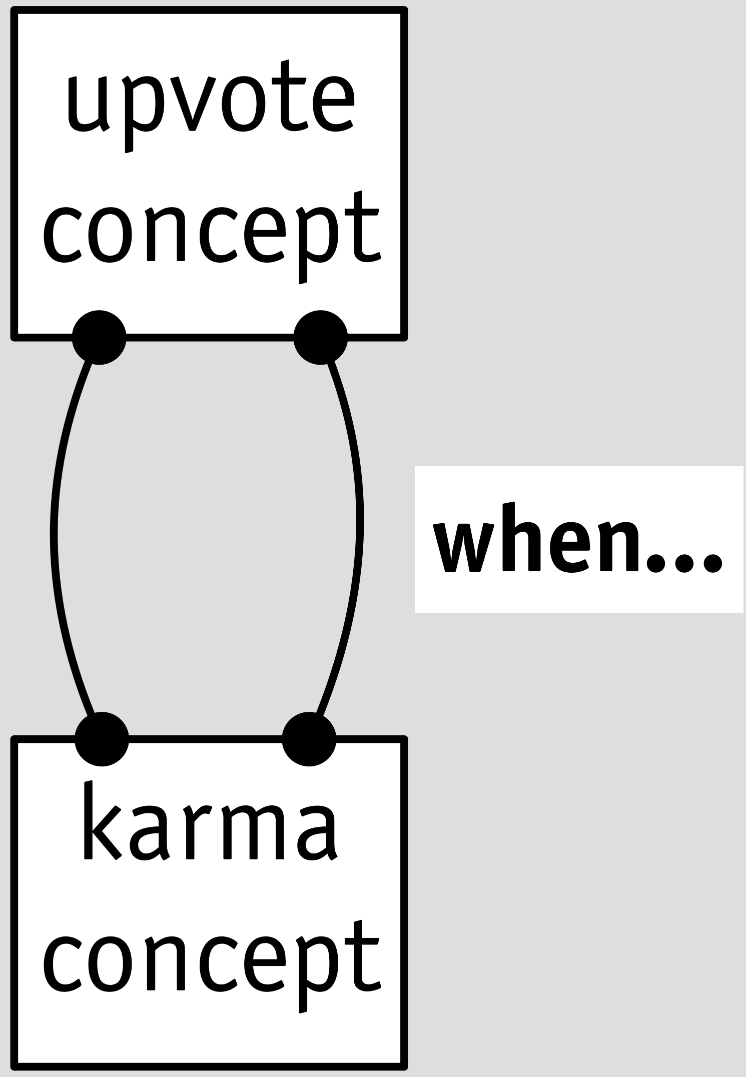
unwanted sync?

lessons
& takeaways

how we resolved the challenges



where's the boundary?
a concept is more like a microservice
than a class, datatype or entity.
call it a "nanoservice"?



are concepts coupled?
no! synchronize when composing
so concepts are free-standing

practical lessons

Twitter

do you have the **right concepts?**

inventory your app’s concepts

identify **familiar** concepts to reuse

Dropbox

is each concept **fit for purpose?**

convey the **purpose**

localize the data model

Calendar

is the **composition** right?

compose by **sync**

watch for **over & under sync**

the costs of bad concept design

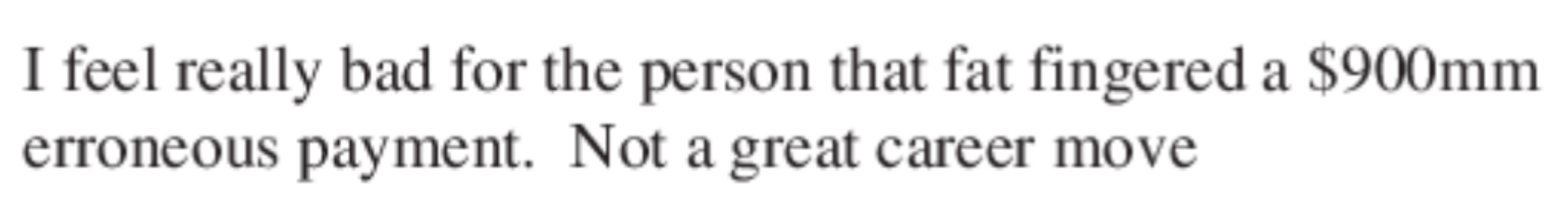
unclear concepts makes life less pleasant and upsets customers

a concept design flaw reported at https://googledrivesucks.com

tricky concepts cause users to limit their use of an app

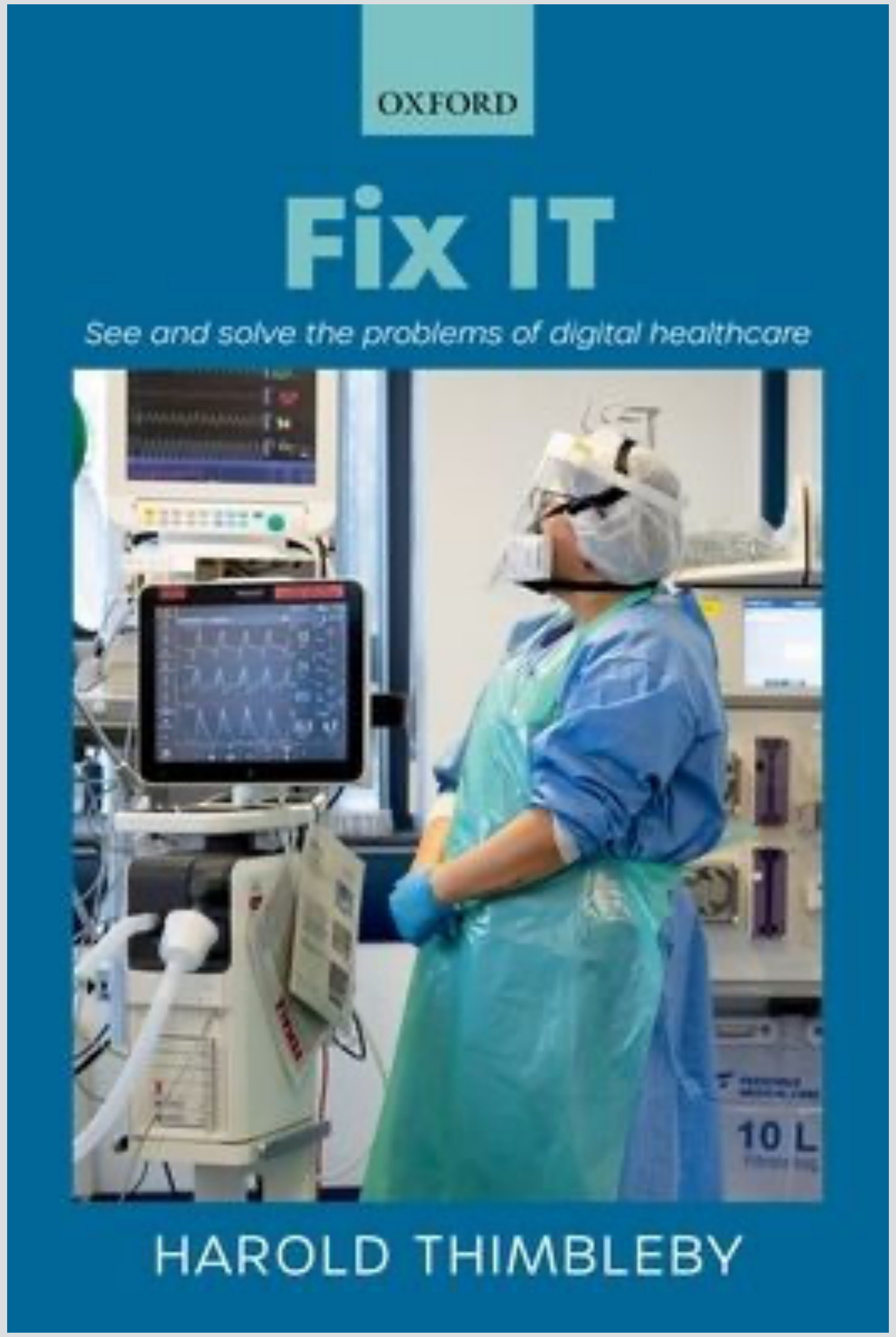
Me: Have you ever encountered these Dropbox problems?
Computer scientist: No, I don't use sharing. Too risky.

confusing concepts lead to costly mistakes



I feel really bad for the person that fat fingered a $900mm erroneous payment. Not a great career move

in re Citibank, 2020



would better concepts
prevent dosage errors?

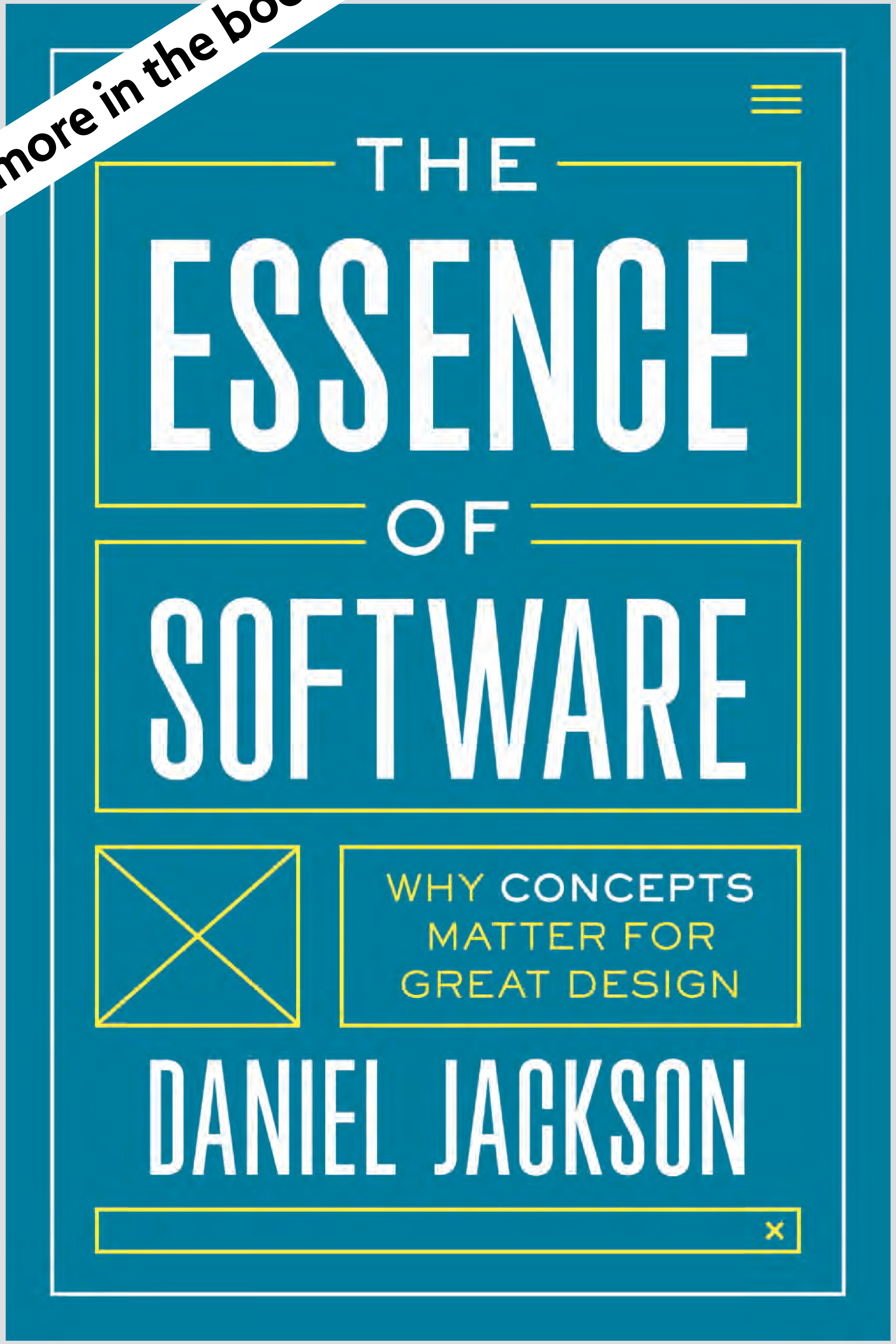
the potential benefits of good concept design

design focus
inform scope choices
new modularity
helps divide work

design ideas
record knowledge
share it with others
reuse designs

design principles
avoid user testing
complement code
& UI heuristics

much more in the book



essenceofsoftware.com

join the discussion
about concept design!
forum.softwareconcepts.io