



Group collaboration

CS 278 | Stanford University | Michael Bernstein

Last time

How to design for different kinds of social groups

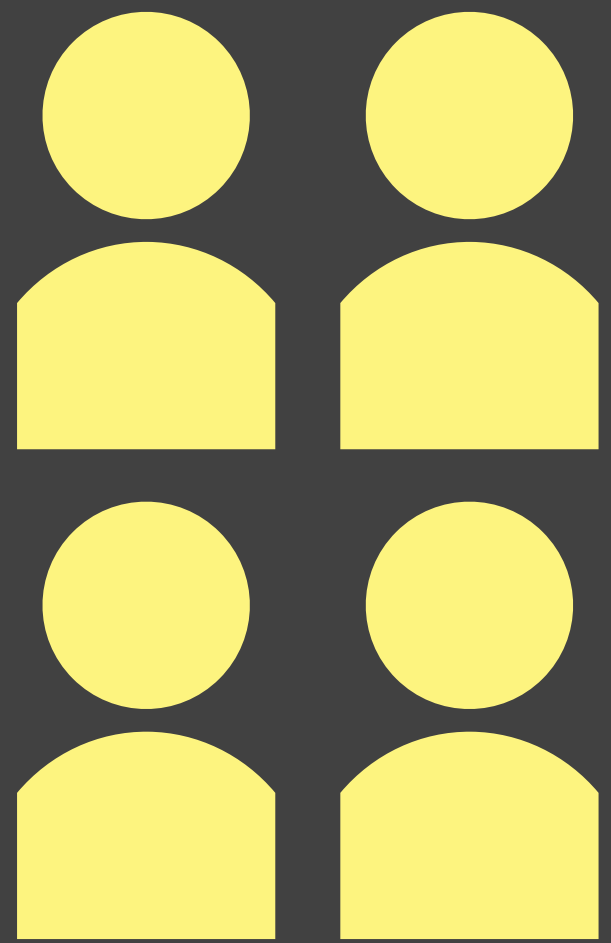
Strong ties: a few tight friends and family — design for honest signals

Weak ties: a wide variety of acquaintances — design for connectedness and to manage non-uniform contributions

Identity-based groups (no ties): brought together by a shared identity rather than pre-existing ties.

Today, a different kind of group: one brought together by shared purpose and goal.

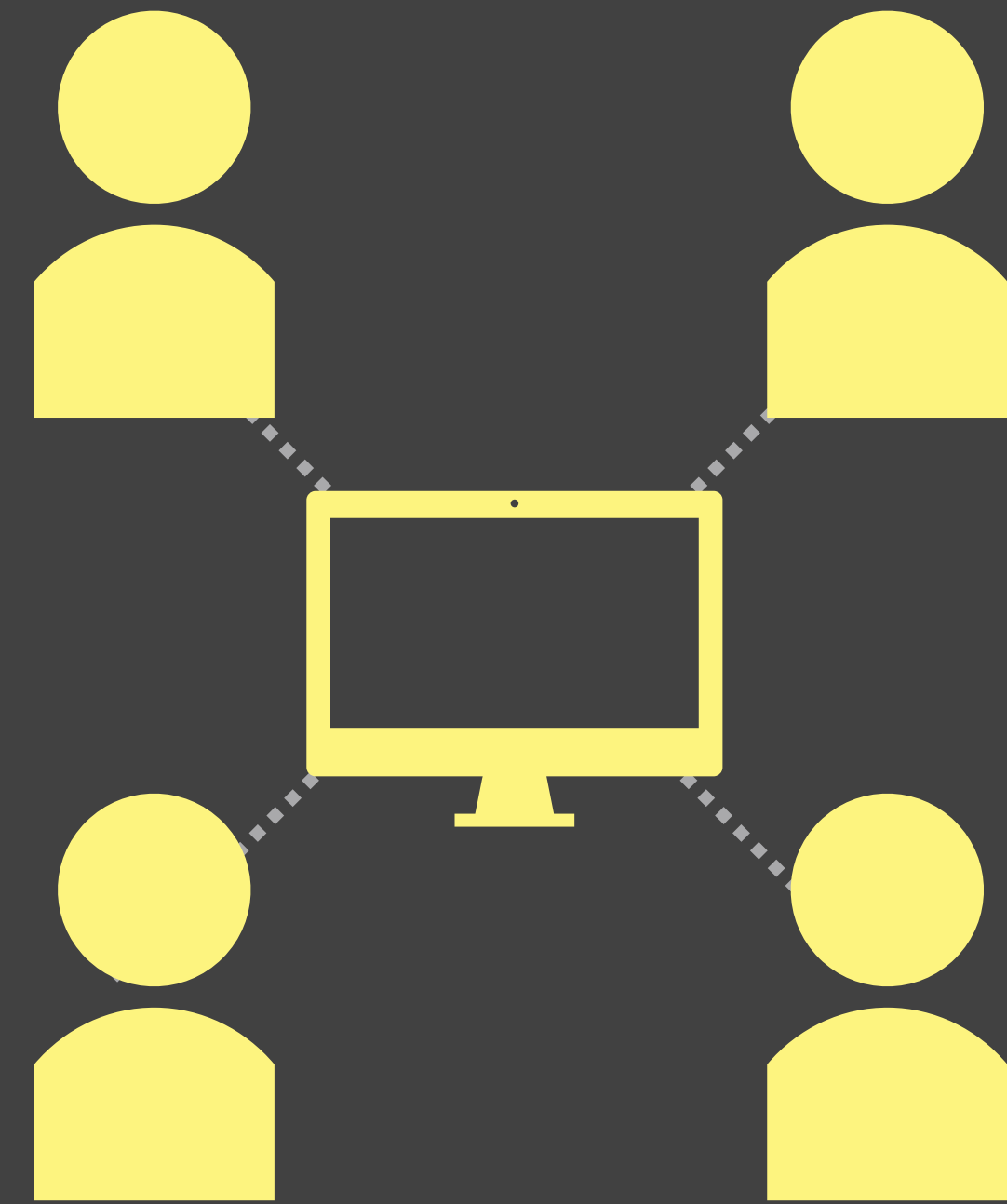
Which team is more effective?



Colocated team
has: a room

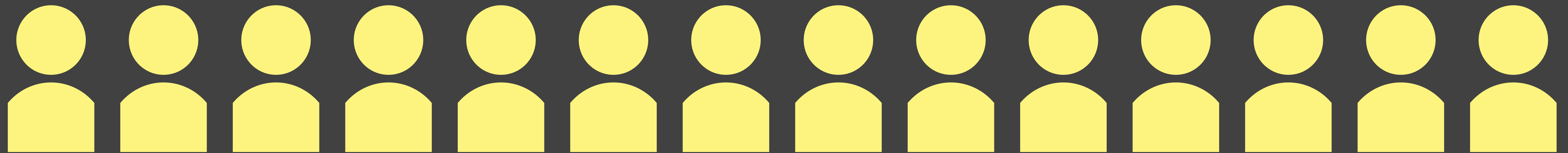
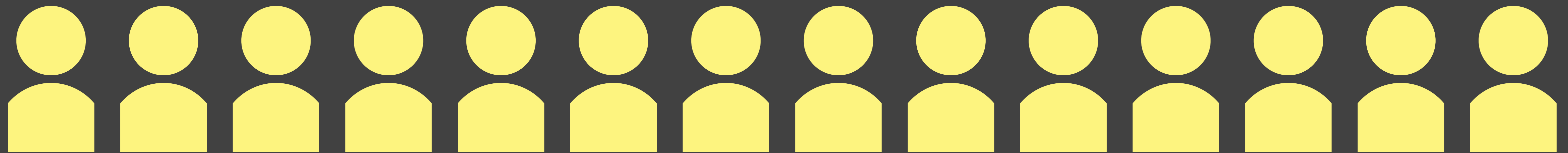
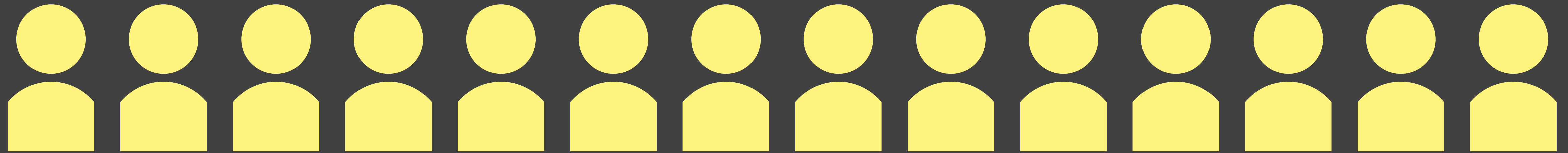
2:1 more effective
[Olson and Olson
2000; Cummings
2011]

Why? And what
can we do about it?

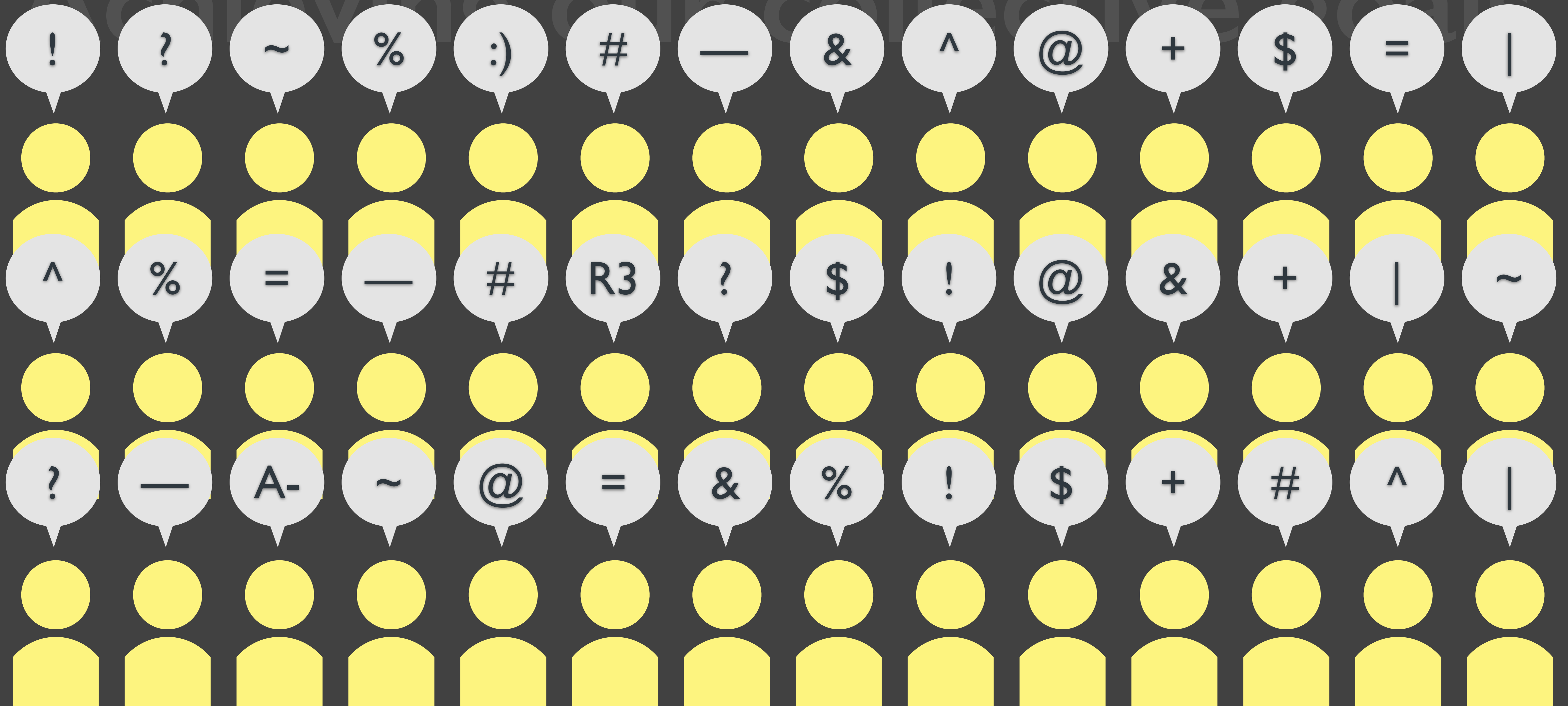


Distributed team
has: Skype, Slack, Trello,
Dropbox, GitHub, Asana,
Google Docs, Jira

Achieving our collective goals



Achieving our collective goals



Out of Sight, Out of Sync: Understanding
Conflict in Distributed Teams

COORDINATION NEGLECT: HOW LAY
THEORIES OF ORGANIZING
COMPLICATE COORDINATION IN
ORGANIZATIONS

The Mutual Knowledge Problem and Its
Consequences for Dispersed Collaboration

The team scaling fallacy: Underestimating the declining efficiency of larger teams

Who's in Charge Here? How Team Authority Structure Shapes Team Leadership

**Team Familiarity, Role
Experience, and
Performance: Evidence from
Indian Software Services**

The Influence of Shared Mental Models on Team Process and Performance

Some unintended consequences of
job design

Structure and Learning in Self-Managed Teams:
Why "Bureaucratic" Teams Can Be Better Learners

Out of Sight, Out of Sync: Understanding Conflict in Distributed Teams

COORDINATION NEGLECT: HOW LAY THEORIES OF ORGANIZING COMPLICATE COORDINATION IN ORGANIZATIONS

The Mutual Knowledge Problem and Its Consequences for Dispersed Collaboration

Failures to achieve our collective goals are rarely due to insufficient skills and increasingly due to fraught collaborations.

The team scaling fallacy: Underestimating the declining efficiency of larger teams

Who's in Charge Here? How Team Authority Structure Shapes Team Leadership

Team Familiarity, Role Experience, and

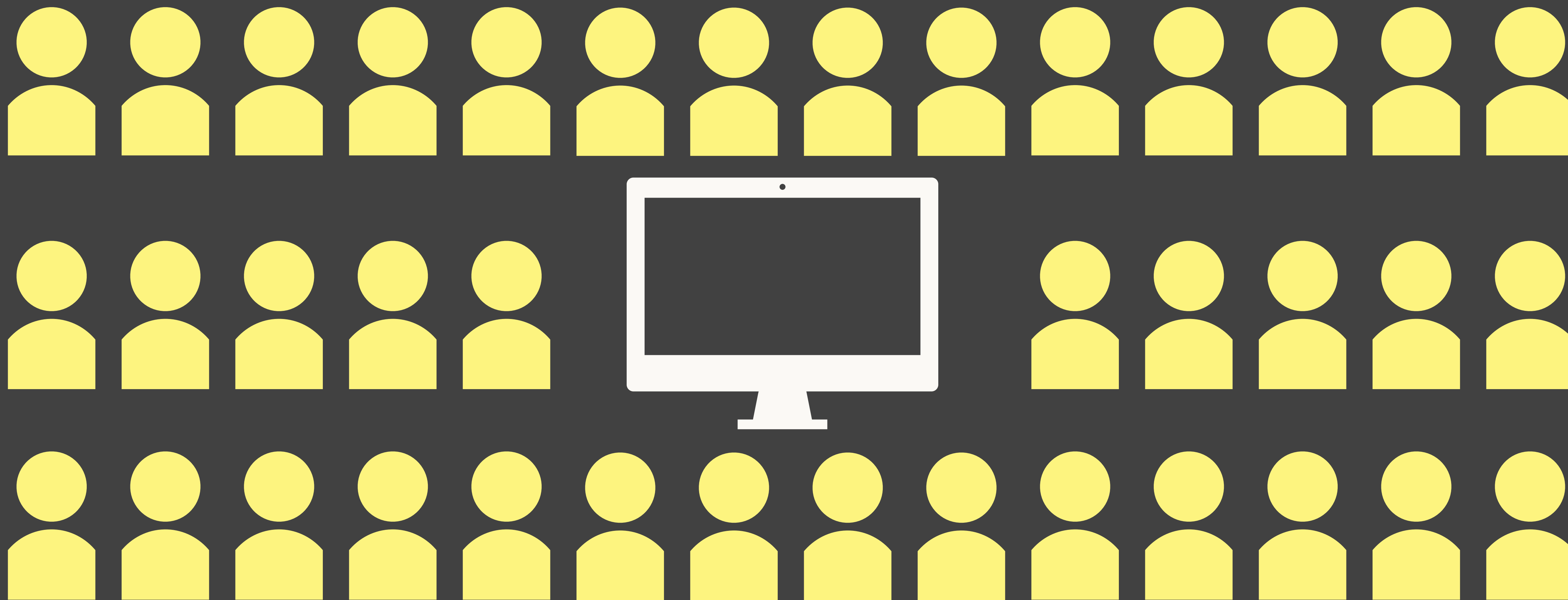
The Influence of Shared Mental Models on Team Process and Performance

Performance: Evidence from Indian Software Services

Some unintended consequences of job design

Structure and Learning in Self-Managed Teams: Why "Bureaucratic" Teams Can Be Better Learners

How might computing augment us in achieving our collective goals?



Today

How do we design tools so that distributed collaboration is as good as really being there?

Topics

- Social translucence

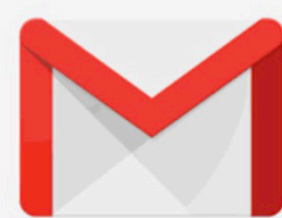
- Beyond being there

- Grudin's paradox

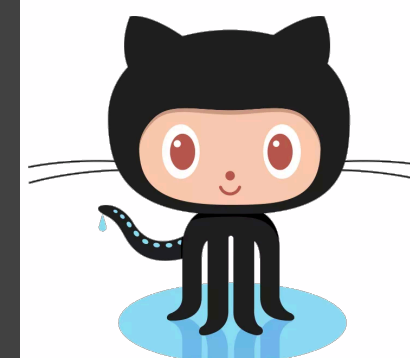
**Making distributed
collaboration as effective as
really being there**

What tools do we use?

Others?



Gmail



GitHub

What design patterns make them successful?
[2min]



Interdependence

What makes teamwork hard (and important) is that the group interactions are **interdependent**.

We can't just work in isolation: we need to engage in behaviors that are discretionary, pro-social, and non-programmed.

Sometimes those behaviors assume risk: asking questions, revealing ignorance, ceding power, putting in extra effort, monitoring each other, and holding each other accountable.

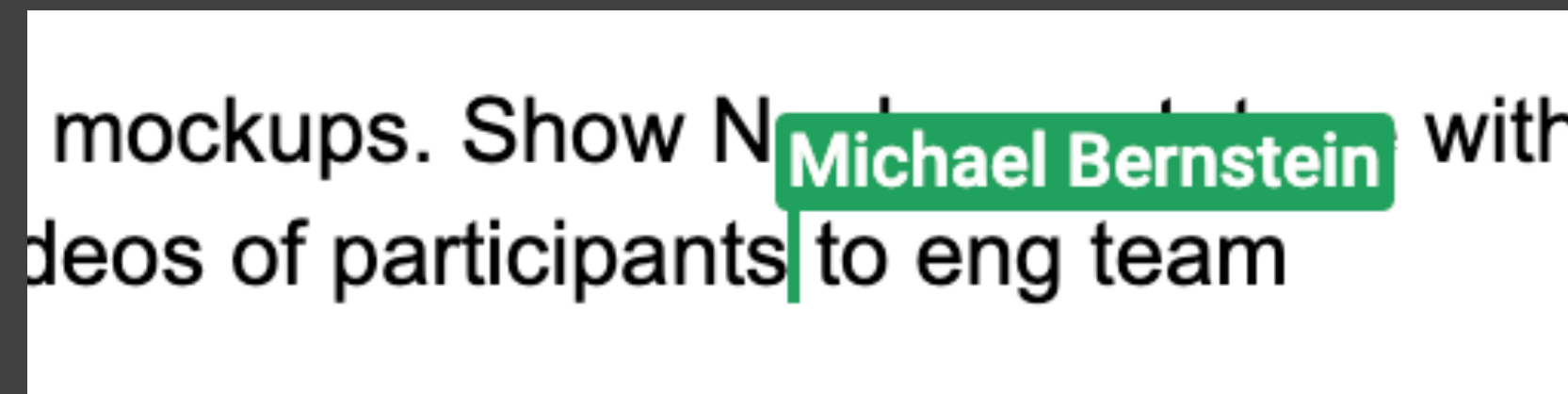
[h.t. Melissa Valentine]

What do we design to support interdependent collaboration?

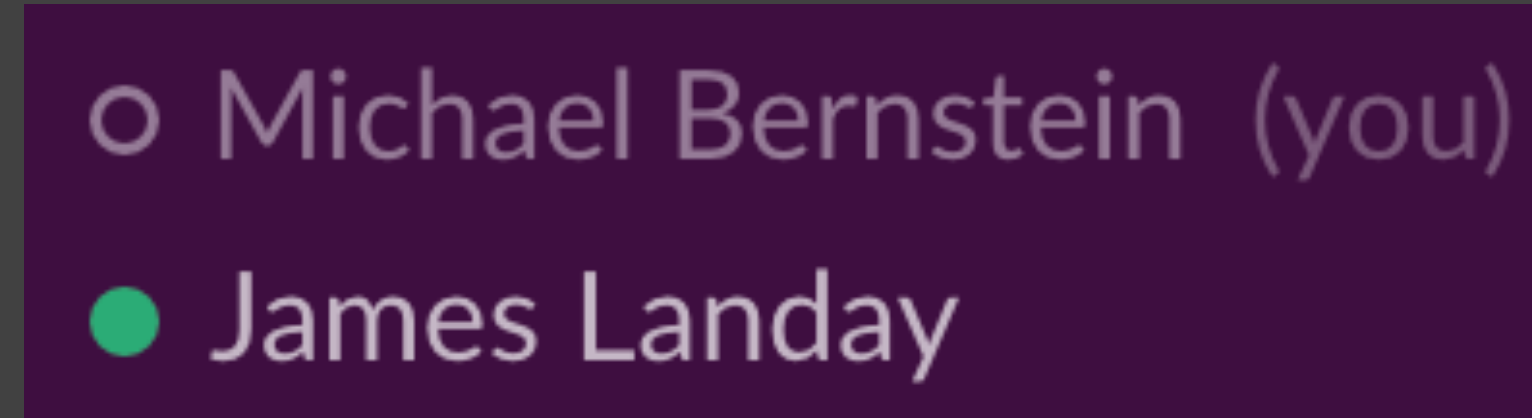
Awareness [Dourish and Bellotti 1992]

If interdependence is the key requirement, then design must allow people to understand each others' state and coordinate accordingly.

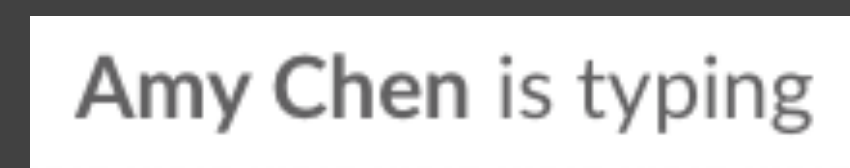
This goal is typically achieved through the design pattern of **awareness**: visualization of others' activities.



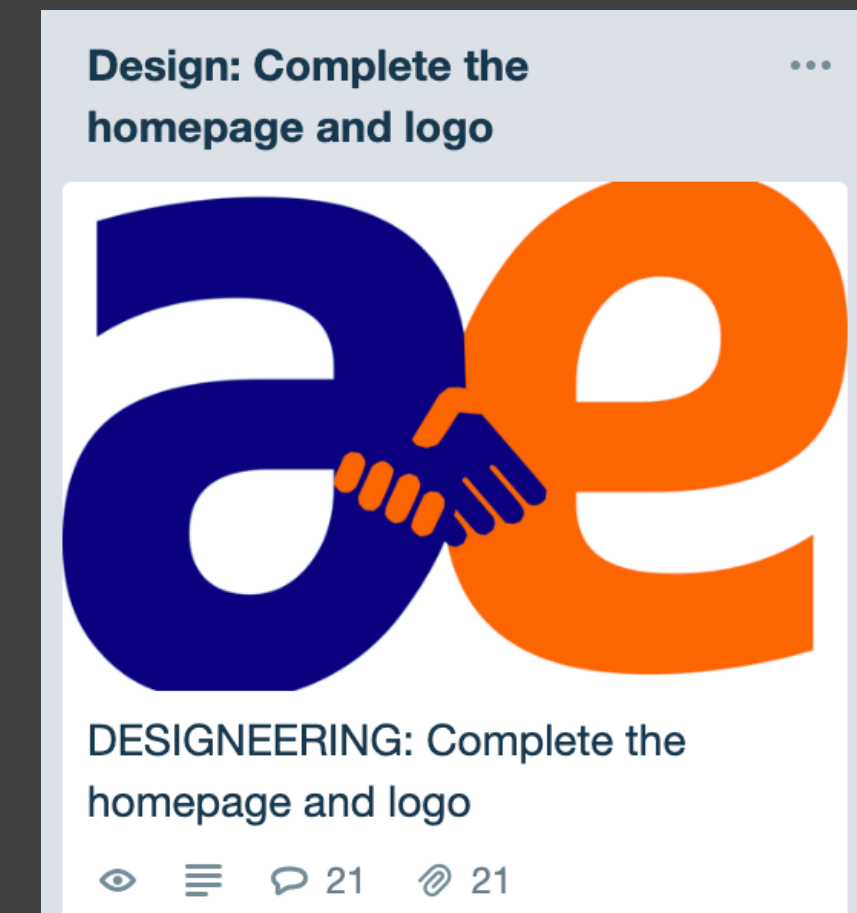
Google Docs



Slack



Messaging apps



Trello todos

But awareness can go too far

You don't want collaborators to know everything...

Whether you're working at every moment

Draft emails you wrote when you were angry but didn't send

Dumb bugs that you introduced into your code but fixed quickly before you made a git commit

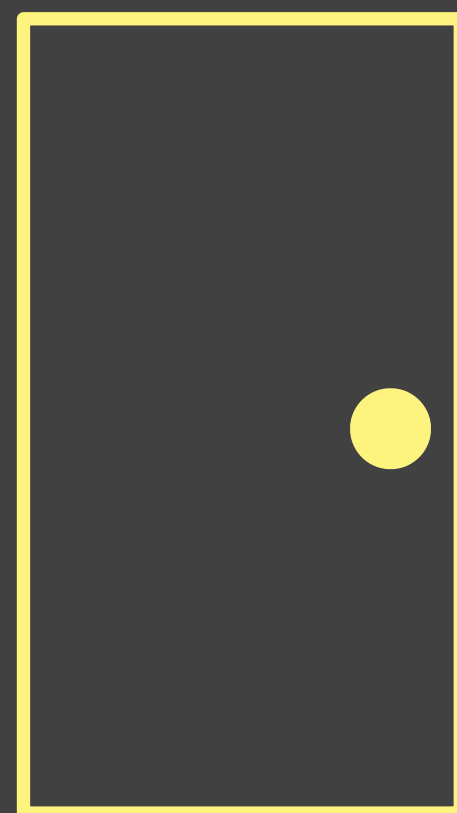
So how do we walk this line?

Social translucence

[Erickson and Kellogg 2000]

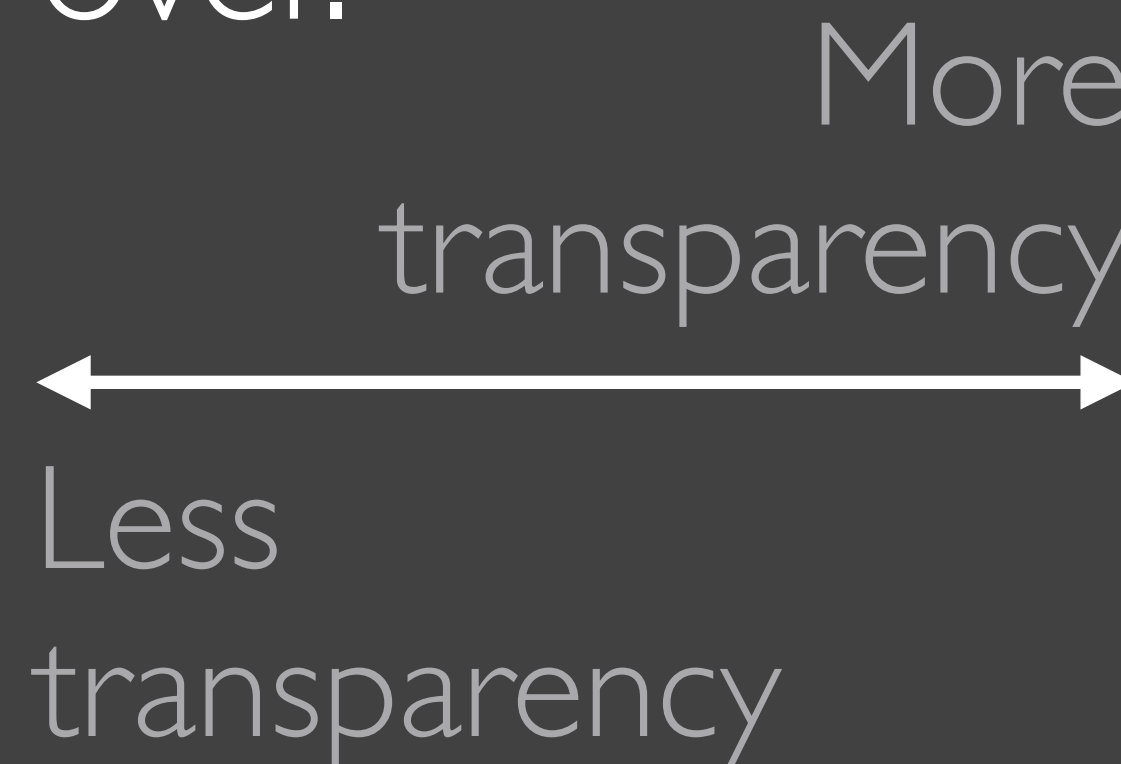
Aim for **socially translucent systems**: give enough information to let natural social cues take over.

Opaque systems:
no information



Solid door
to a trafficked
stairwell

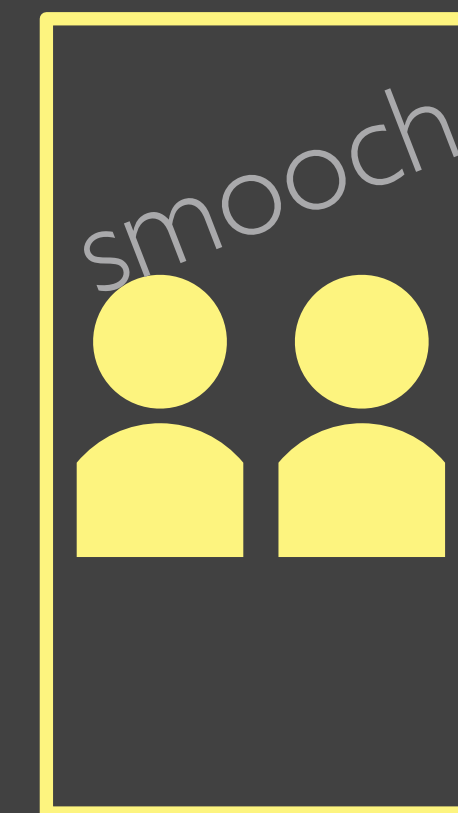
Door-in-the-
face situation



Less

transparency

Transparent systems:
total information



Glass door
to a trafficked
stairwell

Everybody feels
awkward

Social translucence

[Erickson and Kellogg 2000]

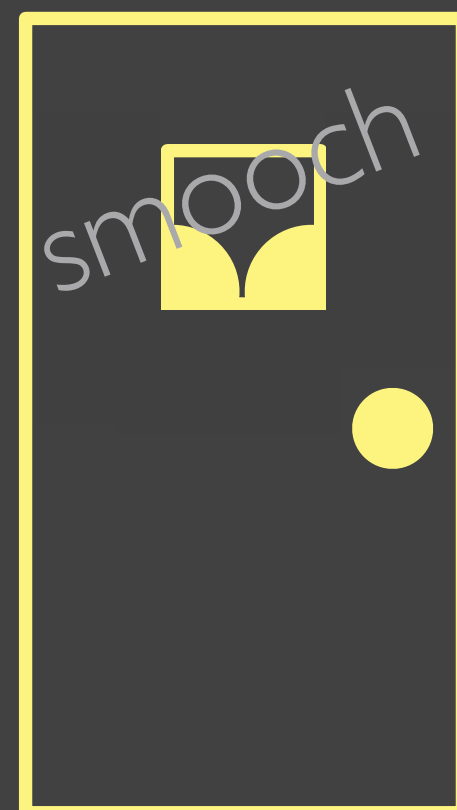
Aim for **socially translucent systems**: give enough information to let natural social cues take over.

Opaque systems:
no information

Solid door
to a trafficked
stairwell

Door-in-the-
face situation

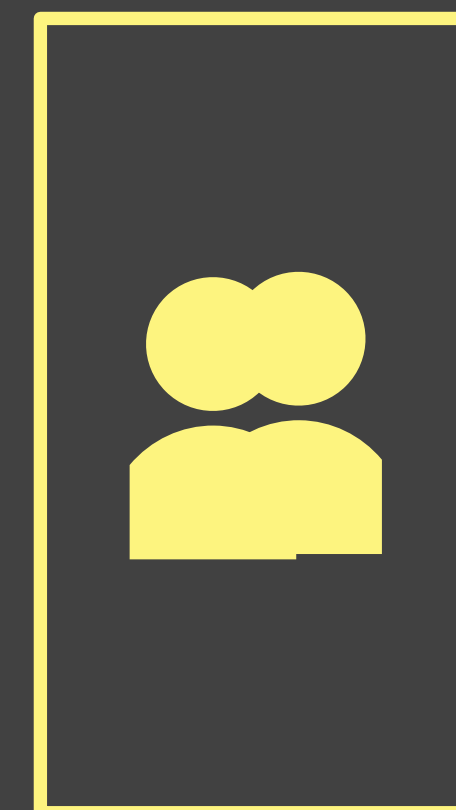
Translucent
systems



Windowed
door

Social cues
prevail

Transparent systems:
total information



Glass door
to a trafficked
stairwell

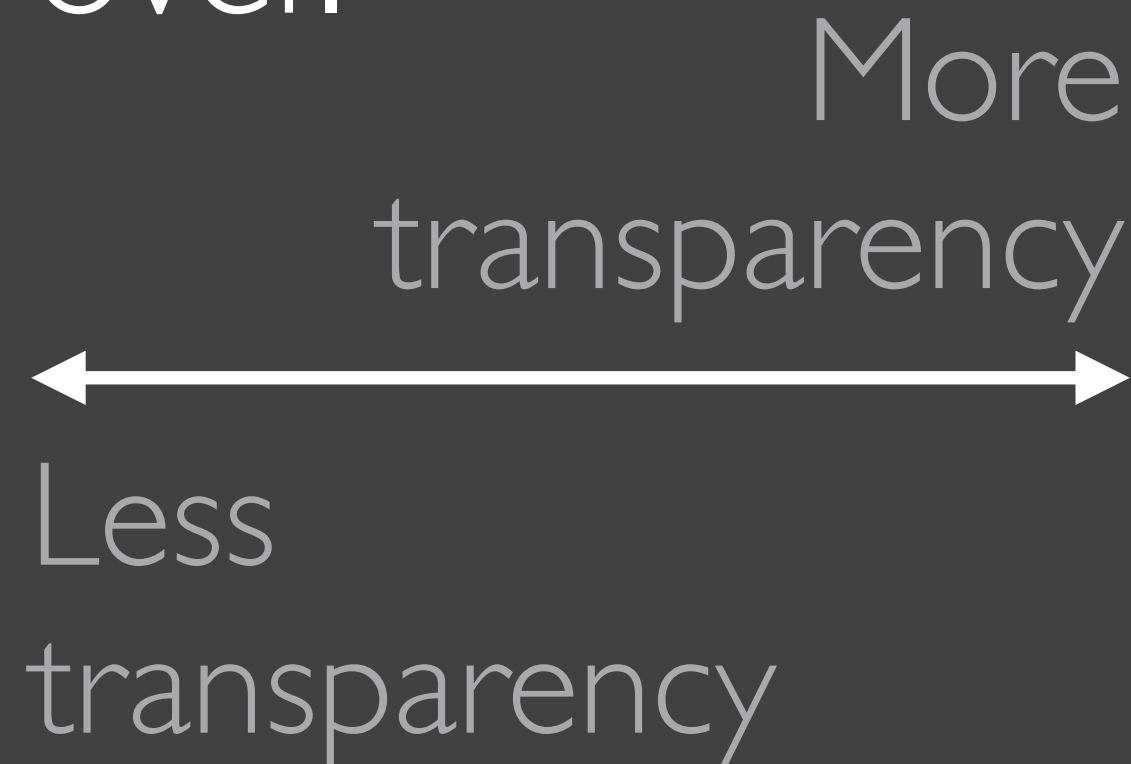
Everybody feels
awkward

Social translucence: example

[Erickson and Kellogg 2000]

Aim for **socially translucent systems**: give enough information to let natural social cues take over.

Opaque systems:
no information
Code isn't pushed yet...



Transparent systems:
total information
Michael Bernstein is editing
`importantfile.py`. He's typing
`i don't know how this works`
over and over into his code
editor.

Social translucence: example

[Erickson and Kellogg 2000]

Aim for **socially translucent systems**: give enough information to let natural social cues take over.

Opaque systems:
no information

Code isn't pushed yet...

Translucent
systems

Michael is working
on `importantfile.py`

Transparent systems:
total information

Michael Bernstein is editing
`importantfile.py`. He's typing
`i don't know how this works`
over and over into his code
editor

Social translucence

[Erickson and Kellogg 2000]

Two requirements for social translucence:

- 1) **Awareness**: others' activity can be seen — to an extent
- 2) **Accountability**: others know that their activity can be seen

If done correctly, social translucence supports interdependent work while maintaining plausible deniability when necessary.

If there's no plausible deniability in the system, people will abandon it.

Beyond being there

Goal: being there

Suppose that we've created a suite of collaboration tools that promote awareness and social translucence while allowing for plausible deniability when needed.

Now, our main goal is to increase fidelity: to try and make the channel have increased richness, allowing for more and more social cues. [Daft and Lengel 1986]

Let's make Skype and FaceTime have lower delays, higher resolution, and 3D VR or AR scenes

Let's make coding collaboration tools as effective as if we were pair programming

Beyond being there

[Hollan and Stornetta 1993]

“Being there” is the wrong goal.

We will never fully recreate the face-to-face experience. There are too many subtle cues for us to fully model or recreate them, even with hypothetical future technology.

Network lag, immersion and comfort issues in VR, lack of shared physical context, ...

So, stop trying.

Beyond being there

[Hollan and Stornetta 1993]

Instead of tilting at windmills to design experiences that are as good as being there, design for **beyond being there: experiences that could never have been created face-to-face.**

How could Skype bring you closer in ways that face-to-face collaboration never could?

How could online coordination tools help us be more effective planners than we ever could with whiteboards and gantt charts?

Examples

Skype translating between languages in real-time and producing foreign language speech in your own voice

Tools that help teams quickly identify if they should be flat or hierarchical, encouraging or critical, and enforcing equal turn-taking [Zhou, Valentine and Bernstein 2018]

Finding just the right person to answer the hard question you are facing, immediately [McDonald and Ackerman 2000]

What are some collaborative superpowers you have or could have?
[3min]

Grudin's paradox

Why do so many collaborative software systems get abandoned?

Dead wikis and documentation at work

Calendars not reflecting actual person or room availability

“Oh, I don’t use that. Just send me an email instead.”

...even though these systems may even provide social translucence and go beyond being there.

Grudin's paradox [Grudin 1998]

The socio-technical system may be benefiting everyone...except the people who are expected to use it.

What is in the product manager's interests may not be in the ordinary users' interests. [Ackerman and Halverson 2003]

Examples:

The manager wants everybody's calendars to be up-to-date...but the programmers don't care, and just want to work on the project.

We want an API to be documented and kept up-to-date, but the people who write and actively use the software don't need the documentation.

Being on Slack is distracting for the people who need to be reached

Grudin's paradox [Grudin 1998]

When a system falls prey to Grudin's paradox, it gets abandoned or circumvented.

How to avoid this? The system needs to provide benefit to all users, not asymmetric benefits.

...And not just perfunctory benefit — enough benefit to justify the work and distraction that using the system might entail.

Hate 'em, then love 'em

Irene Greif, who founded the field — and was the first woman to earn a PhD in CS from MIT — spent much of her career in industry research labs working on collaboration tools.

She notes that with each new generation of collaboration technology, companies are extremely wary: all they can see are the risks and the lawsuits.

Initially, even with something as simple as voicemail!

Collaboration benefits are much harder to quantify and put into dollar amounts, to balance against the risk. Only later do companies see the value and buy in.

So where are we going?



Facebook
Spaces: VR
remote
conversations

Using today's
concepts: will
this succeed?
[2min]

So where are we going?



Beam: robot
telepresence
robot

Using today's
concepts: will
this succeed?
[2min]

Michael's take

All the tools that we talked about today take the organizational structures as given: the team, the teams, the hierarchy, and so on.

e.g., Skype already assumes the members of the team are set

My opinion: the important technologies from here on out will help aid the authoring and evolution of these structures more directly.

Who should be working with who? And how?

What's the best way for this team to be working together?

Can we recover if we get into conflict and fracture?

Best

memes

As voted by the class.

Assignment 1: Go Viral

Due: Tuesday, April 9 at 11:59pm. Submit on Canvas.



Alex Pham

April 5 at 4:23 PM

Shitty Stanford WiFi reaccs only

Stanford:



Stanford University

Stanford, CA

#1 in Computer Science (tie)

Also Stanford:



Liza Hafner, Kathy Tran and 611 others

36 Comments

Like

Comment

Share

Alex Pham



KA



Avni Kakkar



😎👏⚠️ LIFE HACK ⚠️👏😎

Wear your Tuesday socks on Monday to cope with the fact that it's a Monday and convince yourself otherwise

MONDAY

Claire Rosenfeld

Mom: How did I raise a child with no discernable talents or passions.

Me:



Debugging heap allocator with 5 minutes left



Jack
Joseph
Gartland



Sho Arora

Overheard at Stanford: “I’d rather not go to the gym at all than show up without my AirPods”



Summary

Group and team collaboration requires interdependence, which leads to a distinct set of design constraints and affordances.

Social translucence is a general principle for designing these systems with awareness and accountability.

Aiming just to replicate the experience of being there is quixotic; better to aim for beyond being there by looking for affordances unique to the digital realm.

If incentives are misaligned, these systems will get abandoned.

Social Computing

CS 278 | Stanford University | Michael Bernstein

Creative Commons images thanks to Kamau Akabueze, Eric Parker, Chris Goldberg, Dick Vos, Wikimedia, MaxPixel.net, Mescon, and Andrew Taylor.

Slide content shareable under a Creative Commons Attribution-NonCommercial 4.0 International License.