

**NO WORKFLOW CAN  
EVER BE ENOUGH:  
HOW CROWDSOURCING WORKFLOWS  
SUPPORT AND CONSTRAIN COMPLEX WORK**

Daniela Retelny, Michael S. Bernstein, Melissa A. Valentine

**CSCW 2018**

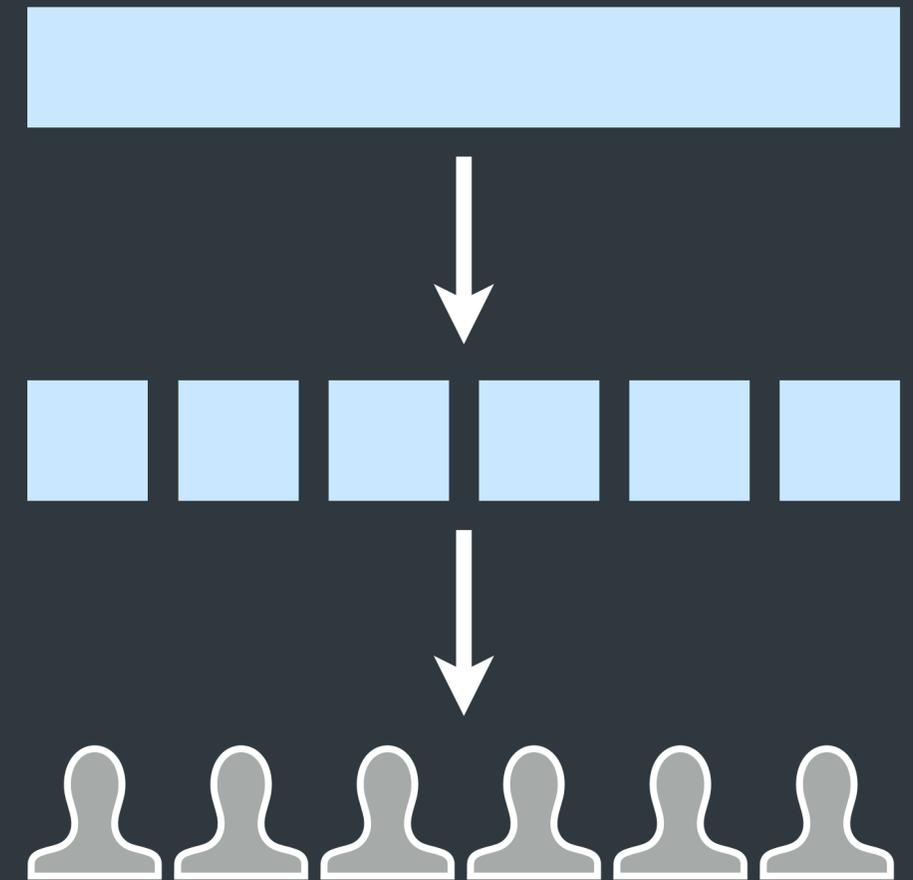
# CROWDSOURCING: SMALL TASKS, MANY PEOPLE

Combine many small tasks  
completed by independent  
workers

e.g., text shortening [Bernstein et al., 2010]

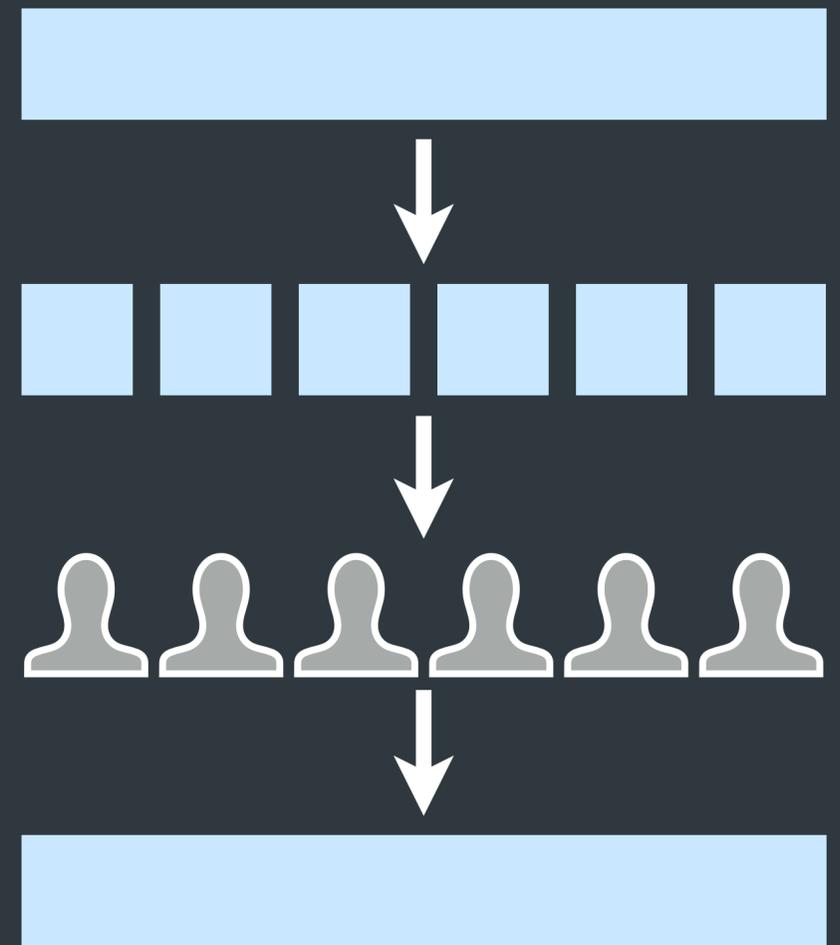
e.g., image labeling [Sorokin & Forsyth, 2008]

e.g., translation [Hu et al., 2011]



# CROWDSOURCING WORKFLOWS

Pre-specified and computationally sequenced set of decomposed tasks that are assigned to distributed workers and combined to reach a final goal [Weld & Bragg, 2010, Little et al., 2010].



# WORKFLOWS ENABLE CROWDS TO ACHIEVE LARGE SCALE GOALS

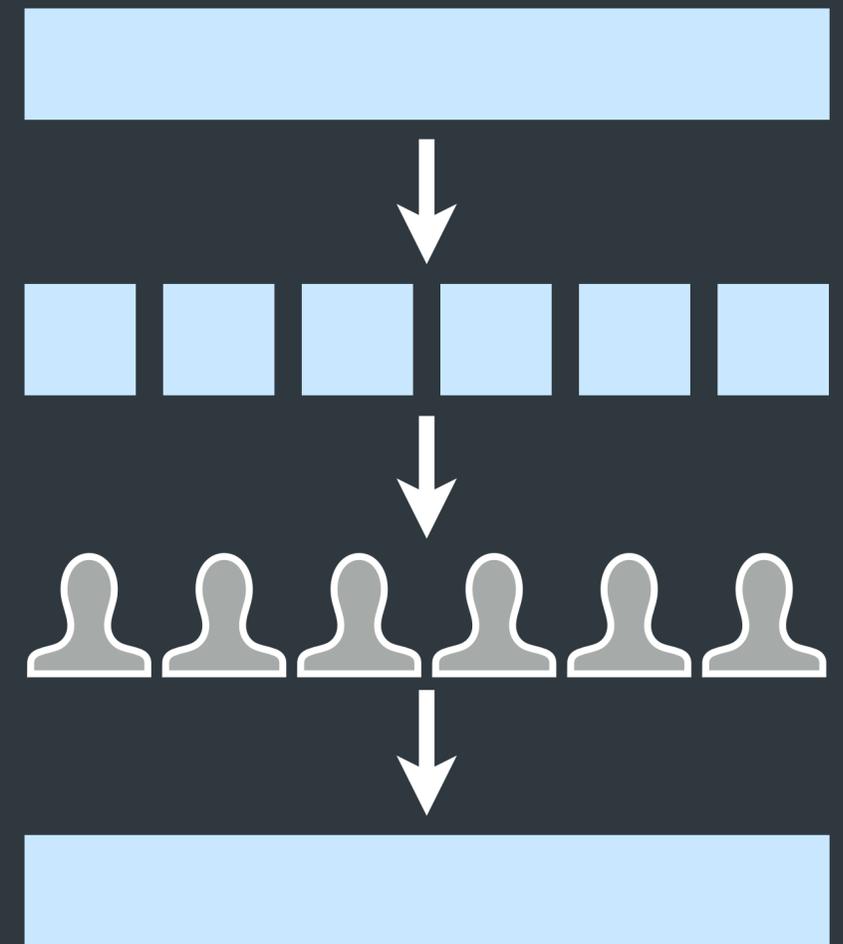
Crowd workflows have been used for:

document editing [Bernstein et al., 2010]

email management [Kokkalis et al., 2013]

text translation [Hu et al., 2011]

software development [LaToza et al., 2014]



# COMPLEX GOALS REMAIN A STRUGGLE FOR CROWDSOURCING

Complex goals such as innovation remain an open challenge for workflow based crowdsourcing [Kittur et al., 2011, Kittur et al., 2013].

Require diverse expertise (e.g., can't be completed by anyone)

Highly interdependent, uncertain and dynamic

**Our objective is to understand:**

Why do complex goals remain a struggle?

To what extent do complex goals represent fundamental limits of workflow-based crowdsourcing?

HOW DO CROWDSOURCING  
WORKFLOWS **SUPPORT** AND  
**CONSTRAIN** COMPLEX  
CROWD WORK?

# OUR ARGUMENT: WORKFLOWS ARE A DOUBLE-EDGED SWORD

**Method:** inductive study of workflow- and role-based expert crowd work.

Our results show: crowdsourcing workflows **support coordination** but **inhibit adaptation**.

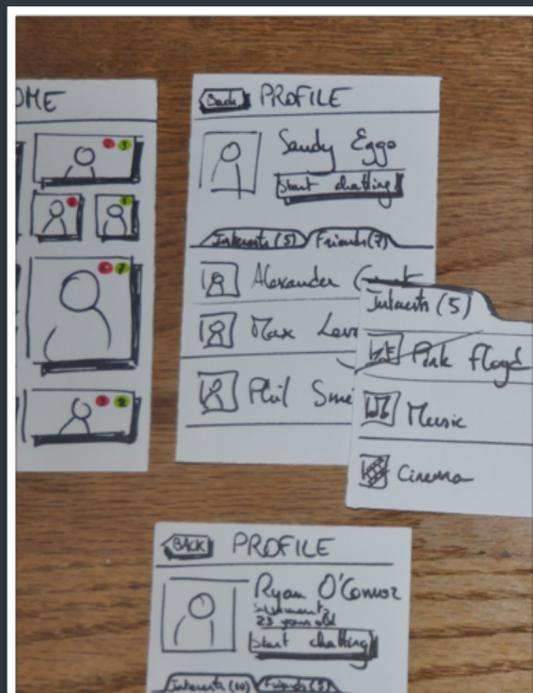
Pre-specified workflows better support coordination and require less communication but inhibit teams from adapting.

Minimally specified work plans are easier to adapt but make coordination more difficult, require more communication and don't ensure the most effective adaptations are pursued.

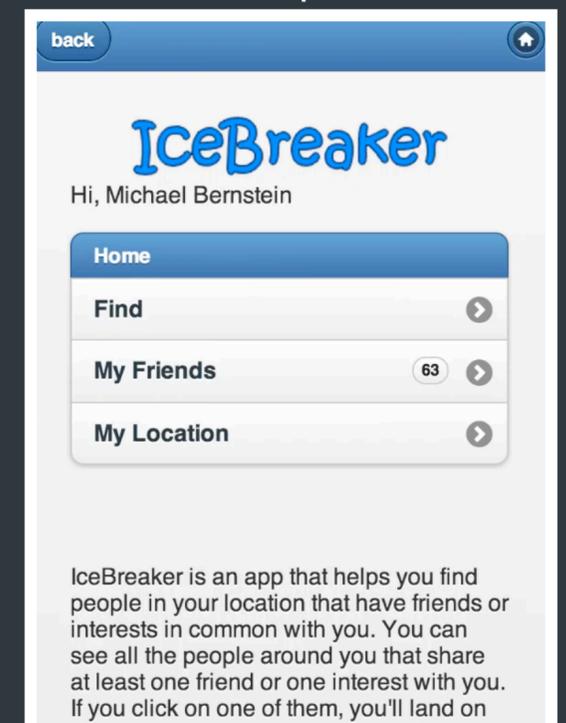
# FLASH TEAMS

Computationally-guided teams of crowd experts supported by **lightweight, reproducible and scalable team structures** [Retelny et al., 2014].

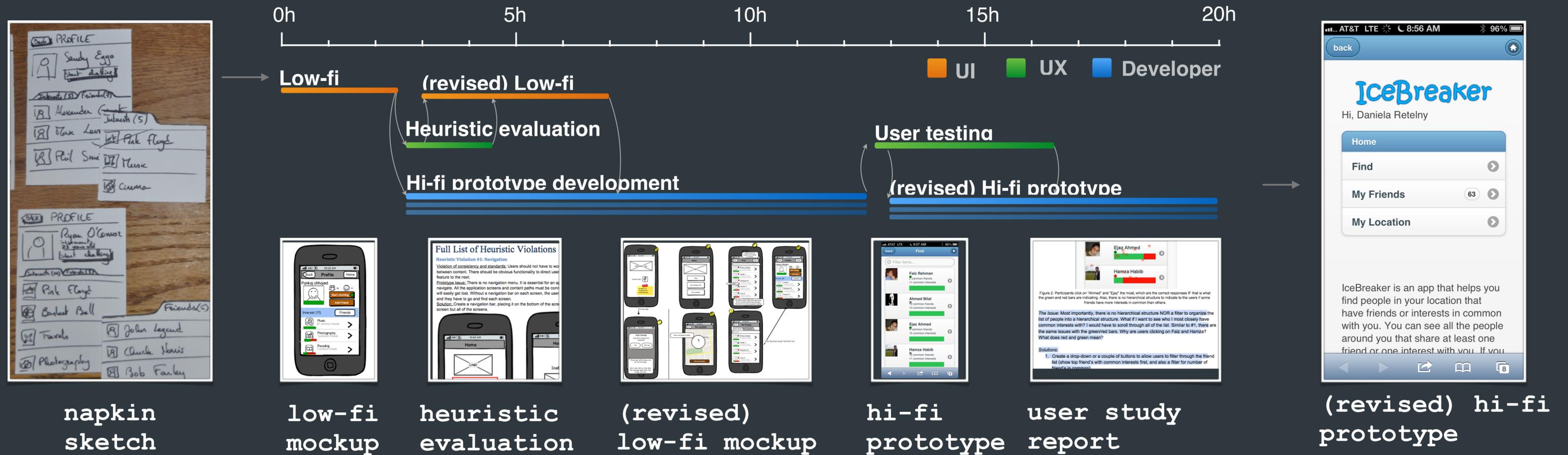
Input



Output



# FLASH TEAMS



[Retelny et al., 2014]

# METHOD: CASE COMPARISON

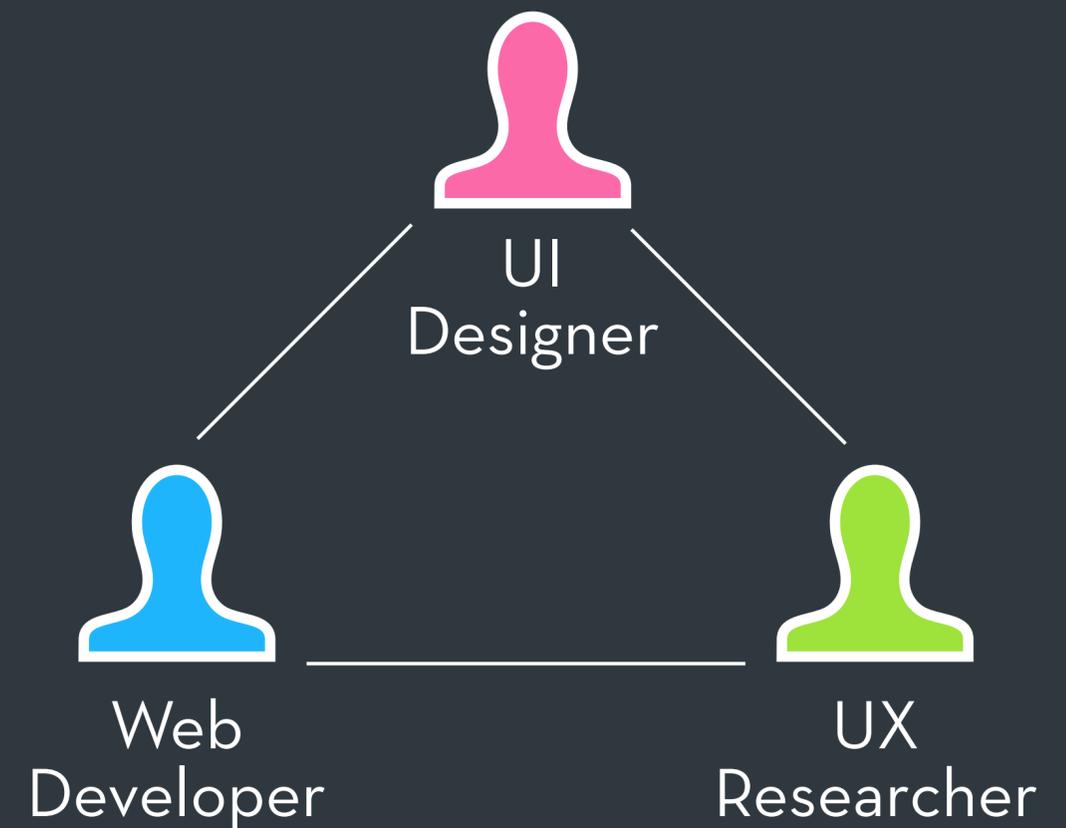
## SIX TEAMS OF CROWD WORKERS

Recruited 22 crowd workers from Upwork and randomly assigned them into six teams

On average, the workers had a rating of 4.66/5, 767 total work hours, an hourly rate of \$22.22, were mostly male (19 males, 3 females) and came from 8 countries.

Task: create a party-planning task manager mobile web app starting from input sketch

Target deadline: 13 hours



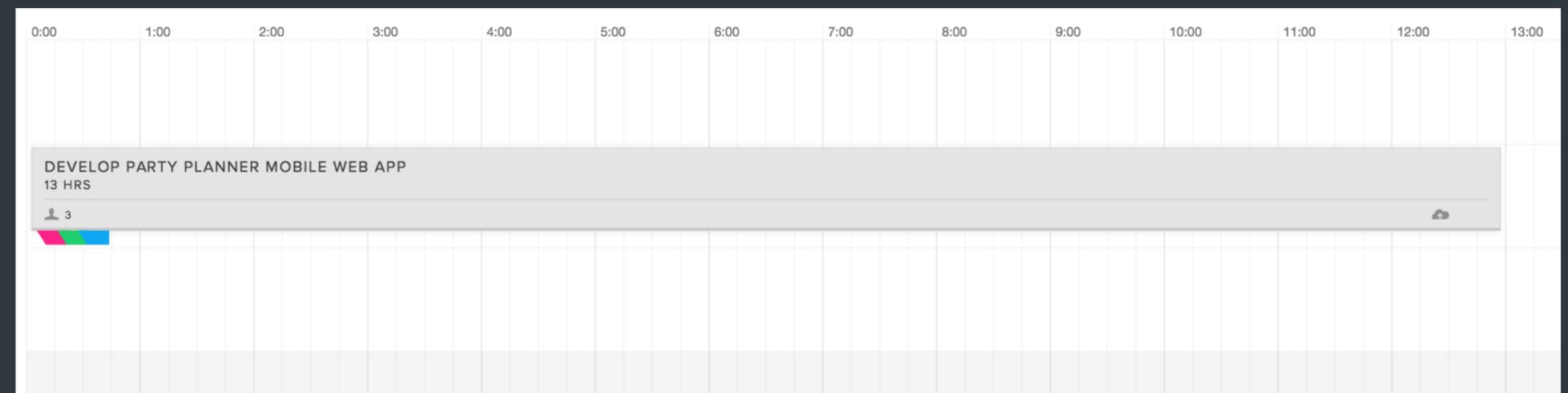
# METHOD: CASE COMPARISON

## WORKFLOW-BASED VS. ROLE-BASED TEAMS

**3 workflow-based teams:**  
enacted pre-specified  
workflows



**3 role-based teams:**  
enacted minimally  
specified work plans



# DATA COLLECTION & ANALYSIS

## WORKFLOW-BASED VS. ROLE-BASED TEAMS

### Data Collection:

- 22 workers across 6 teams
- Real-time observation notes, chat data, open-ended survey data
- Foundry team data, Upwork work diaries, task deliverables

### Data Analysis:

- Coded and analyzed data for each team individually in NVivo
- Compared themes that emerged across teams and conditions
- Returned to research question and literature to further refine codes
- Once qualitative analysis was complete, we analyzed the work diary data from Upwork to better understand what might explain the differences we were seeing

# DATA COLLECTION & ANALYSIS

## WORKFLOW-BASED VS. ROLE-BASED TEAMS

### Data Collection:

- 22 workers across 6 teams
- Real-time observation notes, chat data, open-ended survey data
- Foundry team data, Upwork work diaries, task deliverables

### Data Analysis:

- Coded and analyzed data for each team individually in NVivo
- Compared themes that emerged across teams and conditions
- Returned to research question and literature to further refine codes
- Once qualitative analysis was complete, we analyzed the work diary data from Upwork to better understand what might explain the differences we were seeing

# KEY TAKEAWAYS

Neither pre-specified workflows nor minimally specified work plans are fully sufficient for orchestrating complex and interdependent crowd goals. Goals of this nature require:

- 1) **Coordinating** interdependent contributions from diverse workers [Kittur et al., 2013, Okhuysen & Bechky, 2009]
- 2) **Adapting** work structures and deliverables in response to unplanned contingencies and emergent opportunities [Brown & Eisenhardt, 1997, Suchman, 2007]

For crowds to take on complex goals, crowdsourcing workflows need to support both coordination and adaptation

FINDING #1: WORKFLOWS SUPPORT COORDINATION

FINDING #2: WORKFLOWS INHIBIT ADAPTATION

FINDING #1: WORKFLOWS SUPPORT COORDINATION

FINDING #2: WORKFLOWS INHIBIT ADAPTATION

# WORKFLOWS SUPPORT COORDINATION

The pre-specified workflows:

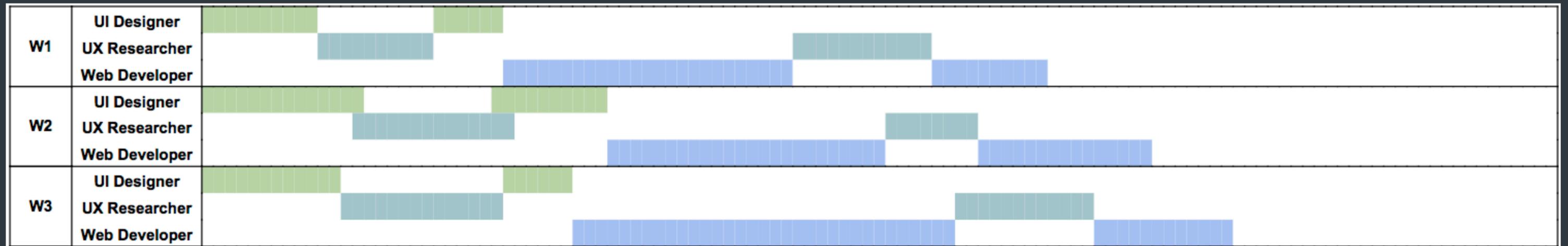
- encoded best practices, ensuring that the workflow-based teams did not make obvious missteps

- clarified the order and relationship of tasks, reducing the amount of explicit coordination and communication needed between workers

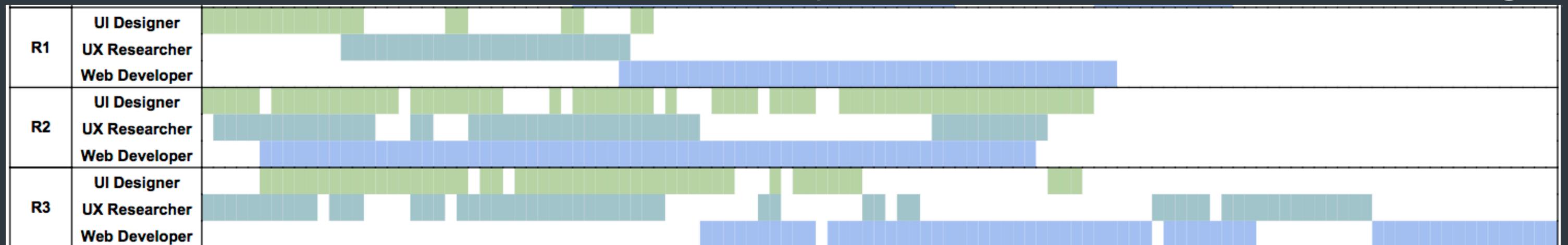
In contrast, the minimally specified work plans required the role-based teams to decide among themselves how to proceed, leading to inefficient coordination and incorrect work.

# WORKFLOWS ENCODE BEST PRACTICES & ORIENT ACTIONS

**Workflow-based teams:** coordinated effectively by following same work plan



**Role-based teams:** enacted different work plans and faced coordination challenges



# WORKFLOWS + COORDINATION

## IMPLICATIONS OF CONCURRENT WORK

	Total Cost	Chat Word Count	Active Work Time
Workflow-based Teams	mean: \$256.36 sd: \$83.19	mean: 3,340.67 sd: 471.36	mean: 12:34:12 sd: 1:27:22
Role-based Teams	mean: \$457.09 sd: \$134.23	mean: 6,424.67 sd: 1,864.33	mean: 23:30:56 sd: 8:03:11

One-tail t-test:

p < .05

p < .05

p < .05

FINDING #1: WORKFLOWS SUPPORT COORDINATION

FINDING #2: WORKFLOWS INHIBIT ADAPTATION

# WORKFLOWS INHIBIT ADAPTATION

In contrast to the minimally specified work plans, the rigid structure imposed by the pre-specified workflows:

- makes it difficult for crowds to respond to contingencies

- prevents crowd workers from pivoting when a better opportunity presents itself

These challenges prevented the workflow-based teams from adapting their work structures and goals.

# WORKFLOWS + ADAPTATION

## RESPONDING TO CONTINGENCIES

### Incompatible deliverables

- caused by workers' decisions, assumptions and errors, which are difficult to predict a priori
- workflow-based teams struggled to catch incompatibilities in time and couldn't adapt when they occurred
- role-based teams caught incompatibilities in real-time and could adapt when they occurred

Decision: using different programming language than requested

Assumption: native app (e.g., iPhone app) instead of mobile web app (e.g., responsive website)

Error: missing fields or error messages in the mockups

# WORKFLOWS + ADAPTATION

## RESPONDING TO CONTINGENCIES

### Incompatible deliverables

- caused by workers' decisions, changes, assumptions and error, which are difficult to predict a priori
- workflow-based teams struggled to catch incompatibilities in time and couldn't adapt when they occurred
- role-based teams caught incompatibilities in real-time and could adapt when they occurred

*“And one more thing. I did not follow any heuristic approach, I followed what makes it easy for the user to use the app. I have the capacity to hypothesize about the reasons behind the actions that people take. I am able to see things from another person's perspective. And most UX designers do more than think about what people do. It's a purely intellectual job. Hope you understand.”*

# WORKFLOWS + ADAPTATION

## RESPONDING TO CONTINGENCIES

### Incompatible deliverables

- caused by workers' decisions, changes, assumptions and error, which are difficult to predict a priori
- workflow-based teams struggled to catch incompatibilities in time and couldn't adapt when they occurred
- role-based teams caught incompatibilities in real-time and could adapt when they occurred

When UI Designer finished the first set of mockups, the UX Researcher quickly reviewed them before conducting the heuristic evaluation and noticed that the wireframes were missing error messages for incomplete fields. This allowed the UI Designer to quickly revise the mockups before the heuristic evaluation began.

*“There was constant communication through Foundry; Feedback and iterations were happening right away, making it possible to execute the project well.”*

# WORKFLOWS + ADAPTATION

## PIVOTING WHEN OPPORTUNITIES EMERGE

All of the teams faced situations in which their work structures or plans were not best suited for their local context or current status.

The constraints and structure enforced by workflows inhibited crowds from considering alternative solutions and pivoting when better opportunities or approaches emerged.

In contrast, the lack of structure and constraints in the role-based teams allowed them to adapt their work plans when better alternatives were available (although they didn't always adapt in most effective way).

# WORKFLOWS + ADAPTATION

## PIVOTING WHEN OPPORTUNITIES EMERGE

### Revising work structures

Workflow

One of the workflow-based team's developer **wanted to work in parallel to minimize the delays** caused by the UI Designer not showing up and try to keep project on schedule **but could not revise the work structure.**

Role-based

When time was running out or workers were overwhelmed, **the role-based teams adapted their work structure by prioritizing key tasks and collaborating** to ensure they got done.

### Pursuing alternative opportunities

The workflow-based teams attempted to add features such as delete, search, pagination and sort to their mobile web apps were unable to successfully add them due to the strict constraints embedded in the workflows.

While the role-based teams attempted to add fewer features, they were more successful when doing so - two teams added delete features and one added data storage feature .

# WORKFLOWS + ADAPTATION

## PIVOTING WHEN OPPORTUNITIES EMERGE

### Revising work structures

Workflow

One of the workflow-based team's developer wanted to work in parallel to minimize the delays caused by the UI Designer not showing up and try to keep project on schedule but could not revise the work structure.

Role-based

When time was running out or workers were overwhelmed, the role-based teams adapted their work structure by prioritizing key tasks and collaborating to ensure they got done.

### Pursuing alternative opportunities

The **workflow-based teams attempted to add features** such as delete, search, pagination and sort to their mobile web apps were **unable to successfully add them due to the strict constraints embedded in the workflows.**

While **the role-based teams attempted to add fewer features, they were more successful when doing so:** two teams added delete features and one added data storage feature.

# DISCUSSION

## WHY CROWDS STRUGGLE WITH COMPLEX GOALS

Crowdsourcing workflows are incompatible with complex work.

Only as effective as the contingencies and adaptations they manage to encode [Van de Ven, 1976, Herbsleb & Grinter, 1999].

It is impossible to predict all possible contingencies and adaptations.

There is no such thing as a perfect workflow for complex work.

The attributes and uncertainty of the task and environment determine the best way to coordinate and design the work [Lawrence & Lorsch 1967, Van de Ven, 1976, Okhuysen & Bechky, 2009].

Workflow paradox: double-edged sword of crowdsourcing workflows

# DISCUSSION

## WHY CROWDS STRUGGLE WITH COMPLEX GOALS

Crowdsourcing workflows are incompatible with complex work.

Only as effective as the contingencies and adaptations they manage to encode [Van de Ven, 1976, Herbsleb & Grinter, 1999].

It is impossible to predict all possible contingencies and adaptations.

There is no such thing as a perfect workflow for complex work.

The attributes and uncertainty of the task and environment determine the best way to coordinate and design the work [Lawrence & Lorsch 1967, Van de Ven, 1976, Okhuysen & Bechky, 2009].

Workflow paradox: double-edged sword of crowdsourcing workflows

# DISCUSSION

## WHY CROWDS STRUGGLE WITH COMPLEX GOALS

Crowdsourcing workflows are incompatible with complex work.

Only as effective as the contingencies and adaptations they manage to encode [Van de Ven, 1976, Herbsleb & Grinter, 1999].

It is impossible to predict all possible contingencies and adaptations.

There is no such thing as a perfect workflow for complex work

The attributes and uncertainty of the task and environment determine the best way to coordinate and design the work [Lawrence & Lorsch 1967, Van de Ven, 1976, Okhuysen & Bechky, 2009].

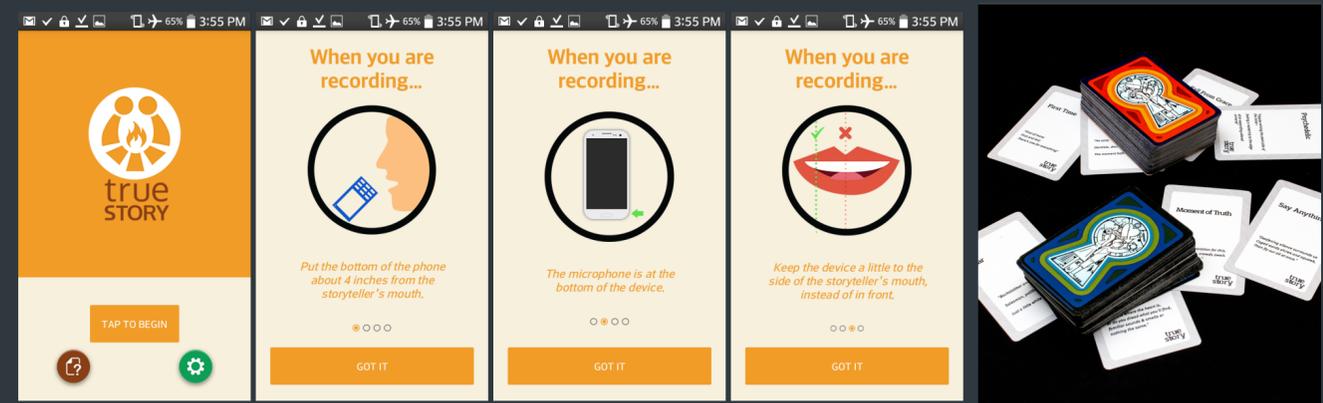
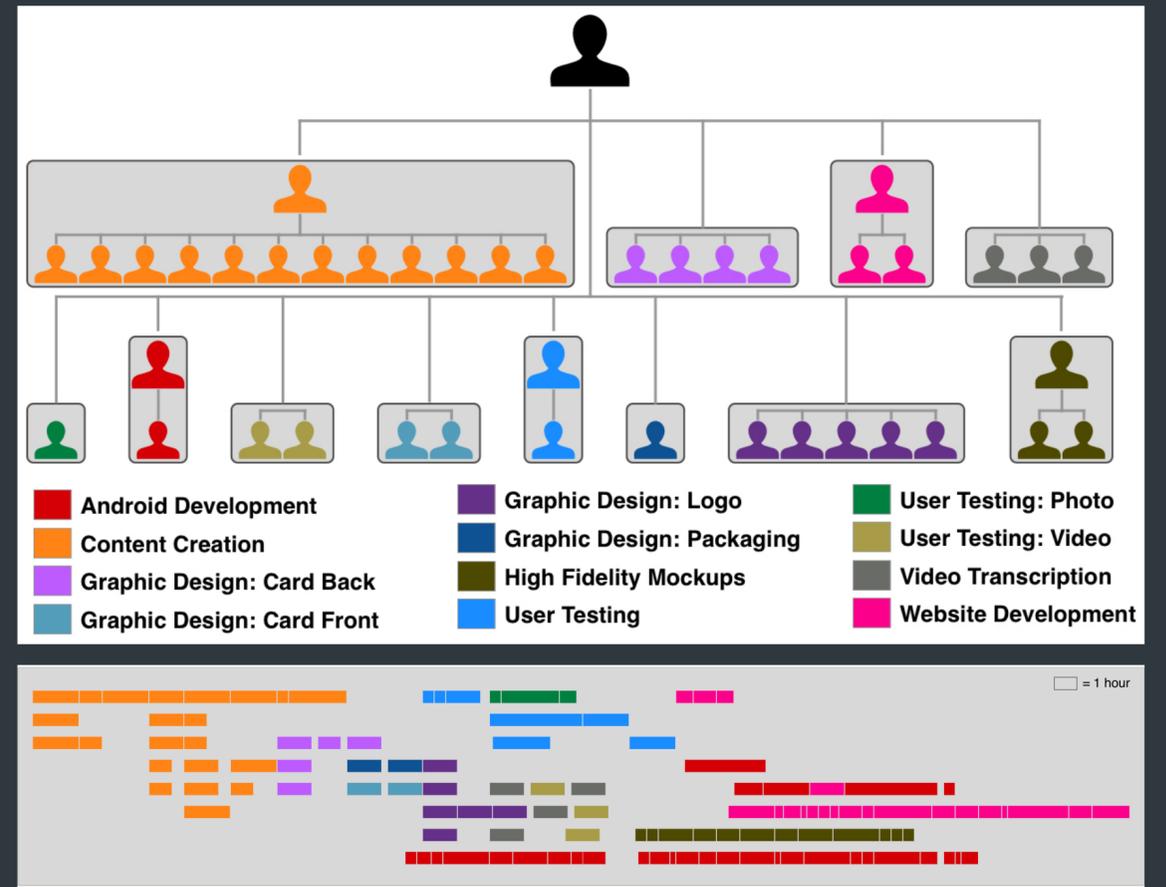
Workflow paradox: double-edged sword of crowdsourcing workflows

# CROWDSOURCING SANS WORKFLOWS?

What now, if workflows are fundamentally incompatible with complex work?

This reflection led our team to an alternative architecture: collaborations structured not as algorithmic workflows, but as **computationally augmented organizations**.

**Flash organizations** [CHI 2017]: rapidly assembled and reconfigurable organizations composed of online crowd workers



**NO WORKFLOW CAN EVER  
BE ENOUGH:  
HOW CROWDSOURCING WORKFLOWS  
SUPPORT AND CONSTRAIN COMPLEX WORK**

Thank you for funding from the NSF, DARPA, the Stanford Cyber Initiative, the Stanford Institute for Research in the Social Sciences, and a Stanford Interdisciplinary Graduate Fellowship.