Graduate Students: Project Reminder

Midpoint due is on Nov. 15 (< 2 weeks from now)

Midpoint presentations on Mon, Nov. 15.

Make progress every day.

Keep a notebook & write as you go*, so that you are not writing both the report and making the slides at the last minute.*

Connect with me if you are stuck!

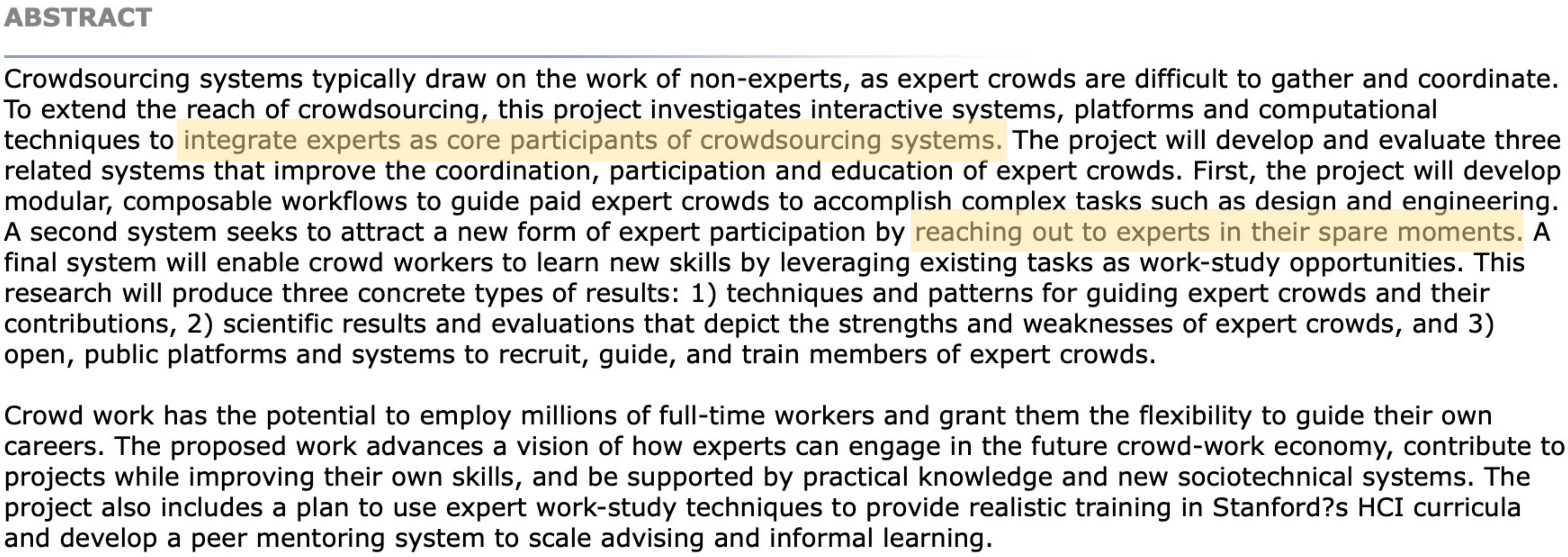
CS 295B/CS 395B Systems for Knowledge Discovery

Systems Considerations of Crowdsourcing Platforms

The University of Vermont

Dynamo & Daemo:
A story of best intentions

Background: Bernstein CAREER (2014)



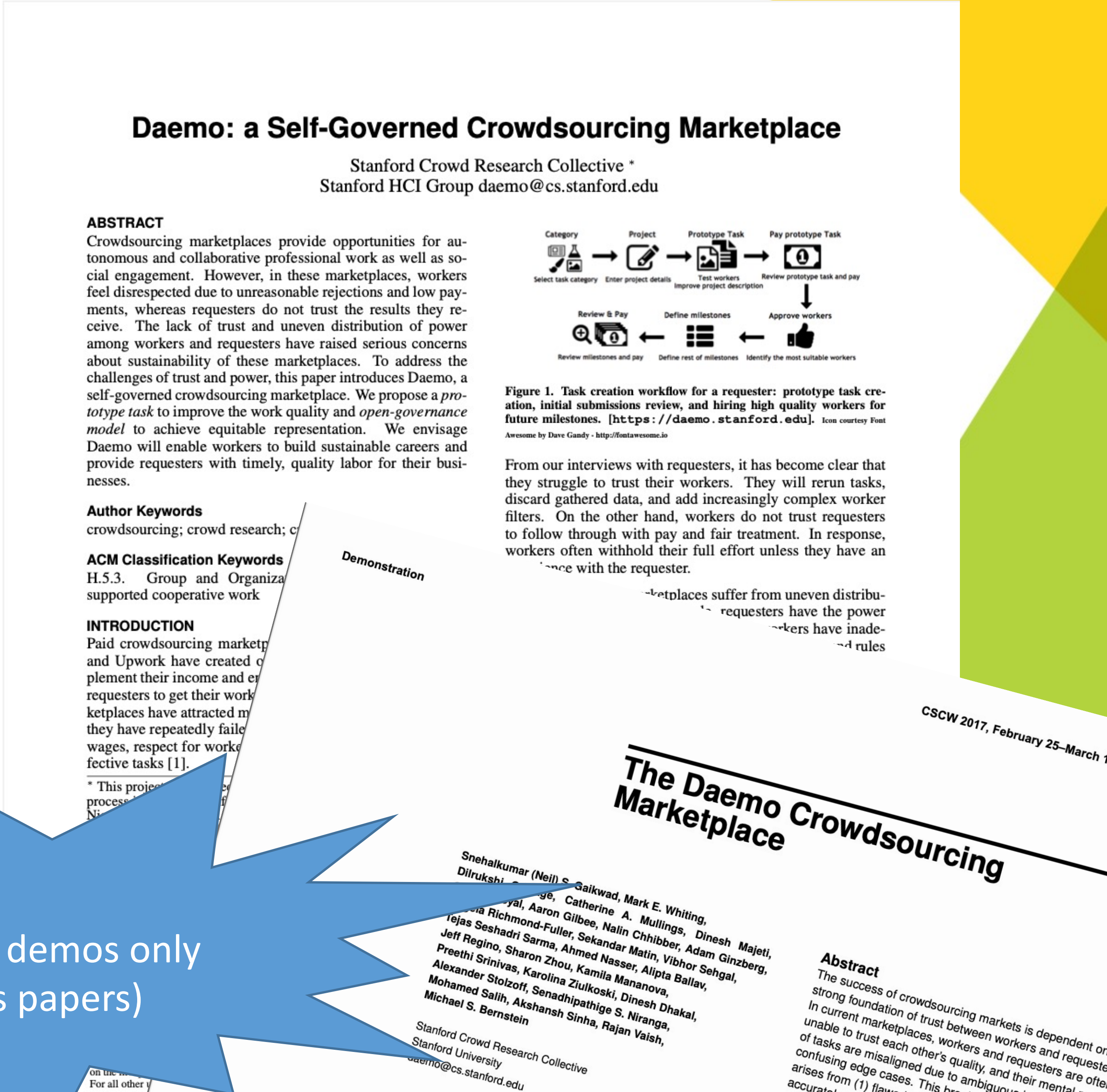
ABSTRACT

Crowdsourcing systems typically draw on the work of non-experts, as expert crowds are difficult to gather and coordinate. To extend the reach of crowdsourcing, this project investigates interactive systems, platforms and computational techniques to integrate experts as core participants of crowdsourcing systems. The project will develop and evaluate three related systems that improve the coordination, participation and education of expert crowds. First, the project will develop modular, composable workflows to guide paid expert crowds to accomplish complex tasks such as design and engineering. A second system seeks to attract a new form of expert participation by reaching out to experts in their spare moments. A final system will enable crowd workers to learn new skills by leveraging existing tasks as work-study opportunities. This research will produce three concrete types of results: 1) techniques and patterns for guiding expert crowds and their contributions, 2) scientific results and evaluations that depict the strengths and weaknesses of expert crowds, and 3) open, public platforms and systems to recruit, guide, and train members of expert crowds.

Crowd work has the potential to employ millions of full-time workers and grant them the flexibility to guide their own careers. The proposed work advances a vision of how experts can engage in the future crowd-work economy, contribute to projects while improving their own skills, and be supported by practical knowledge and new sociotechnical systems. The project also includes a plan to use expert work-study techniques to provide realistic training in Stanford?s HCI curricula and develop a peer mentoring system to scale advising and informal learning.

Daemo

- Group tried to build a crowdsourced-crowdsourcing system
- Appealed for input via mailing list
- Designed by workers, social scientists, HCI researchers



Daemo: a Self-Governed Crowdsourcing Marketplace

Stanford Crowd Research Collective *
Stanford HCI Group daemo@cs.stanford.edu

ABSTRACT

Crowdsourcing marketplaces provide opportunities for autonomous and collaborative professional work as well as social engagement. However, in these marketplaces, workers feel disrespected due to unreasonable rejections and low payments, whereas requesters do not trust the results they receive. The lack of trust and uneven distribution of power among workers and requesters have raised serious concerns about sustainability of these marketplaces. To address the challenges of trust and power, this paper introduces Daemo, a self-governed crowdsourcing marketplace. We propose a *prototype task* to improve the work quality and *open-governance model* to achieve equitable representation. We envisage Daemo will enable workers to build sustainable careers and provide requesters with timely, quality labor for their businesses.

Author Keywords

crowdsourcing; crowd research; c

ACM Classification Keywords

H.5.3. Group and Organiza
supported cooperative work

INTRODUCTION

Paid crowdsourcing marketp
and Upwork have created o
plement their income and er
requesters to get their work
ketplaces have attracted m
they have repeatedly faile
wages, respect for worke
fective tasks [1].

Figure 1. Task creation workflow for a requester: prototype task creation, initial submissions review, and hiring high quality workers for future milestones. [`https://daemo.stanford.edu`]. Icon courtesy Font Awesome by Dave Gandy - http://fontawesome.io

From our interviews with requesters, it has become clear that they struggle to trust their workers. They will rerun tasks, discard gathered data, and add increasingly complex worker filters. On the other hand, workers do not trust requesters to follow through with pay and fair treatment. In response, workers often withhold their full effort unless they have an
ence with the requester.

ketplaces suffer from uneven distribu-
requesters have the power
rkers have inade-
d rules

* This proje
process
Ni

on the
For all other

Demonstration

CSCW 2017, February 25–March 1

The Daemo Crowdsourcing Marketplace

Snehalkumar (Neil) S. Gaikwad, Mark E. Whiting,
Dilrukshi ge, Catherine A. Mullings, Dinesh Majeti,
yal, Aaron Gilbee, Nalin Chhibber, Adam Ginzberg,
ela Richmond-Fuller, Sekandar Matin, Vibhor Sehgal,
Tejas Seshadri Sarma, Ahmed Nasser, Alipta Ballav,
Jeff Regino, Sharon Zhou, Kamila Mananova,
Preethi Srinivas, Karolina Ziulkoski, Dinesh Dhakal,
Alexander Stolzoff, Senadhipathige S. Niranga,
Mohamed Salih, Akshansh Sinha, Rajan Vaish,
Michael S. Bernstein

Stanford Crowd Research Collective
Stanford University
daemo@cs.stanford.edu

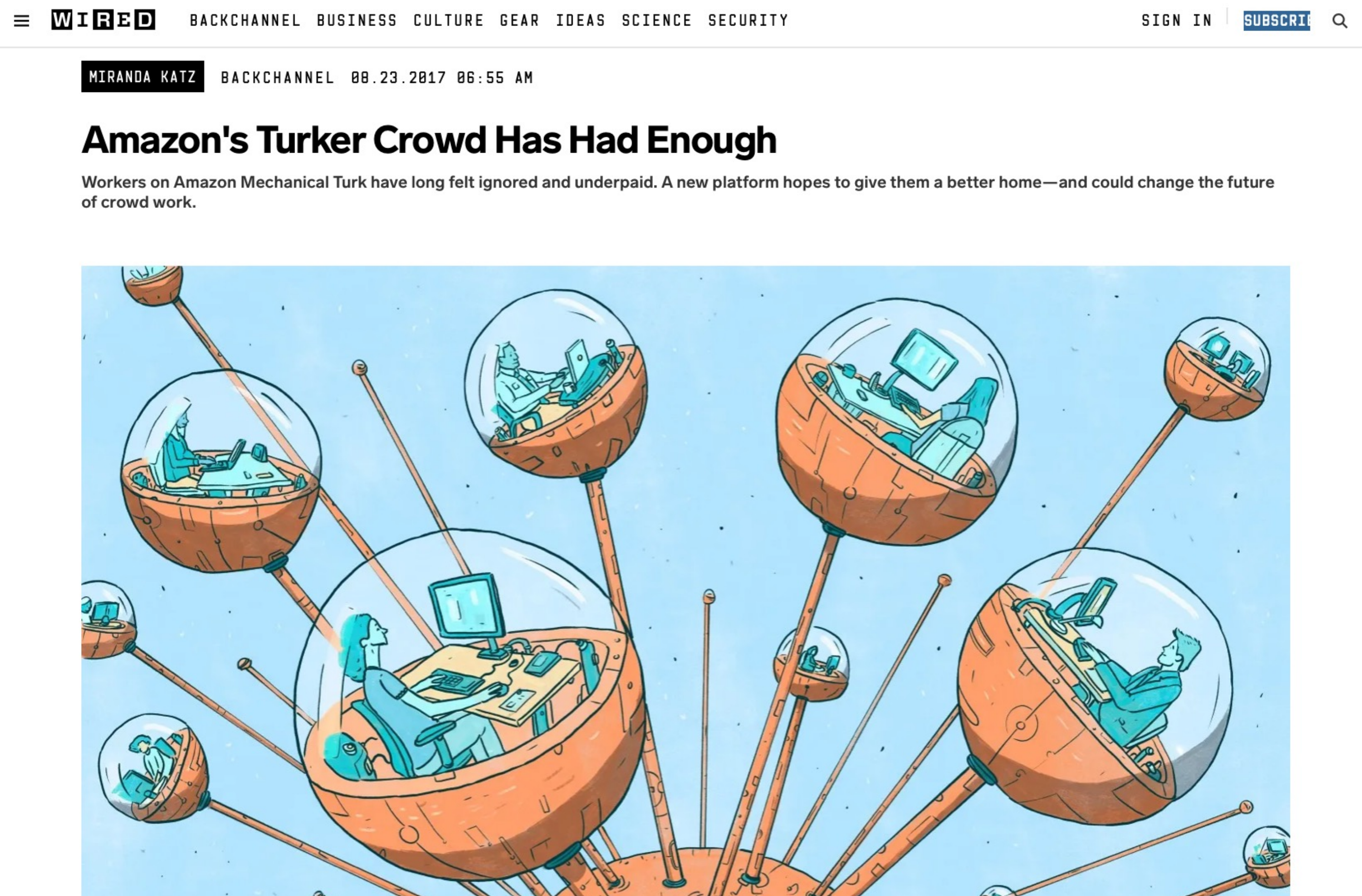
Abstract

The success of crowdsourcing markets is dependent on
strong foundation of trust between workers and requeste
In current marketplaces, workers and requesters are ofte
unable to trust each other's quality, and their mental
of tasks are misaligned due to ambiguous
confusing edge cases. This bro
arises from (1) flaw
accurat

Short papers + demos only
(no systems papers)

Press: replacement for AMT

- Idea: Amazon nonresponsive
- Solution: build a new system
- Challenge: buy-in
- Open source
- Appears to link AMT
- Details?

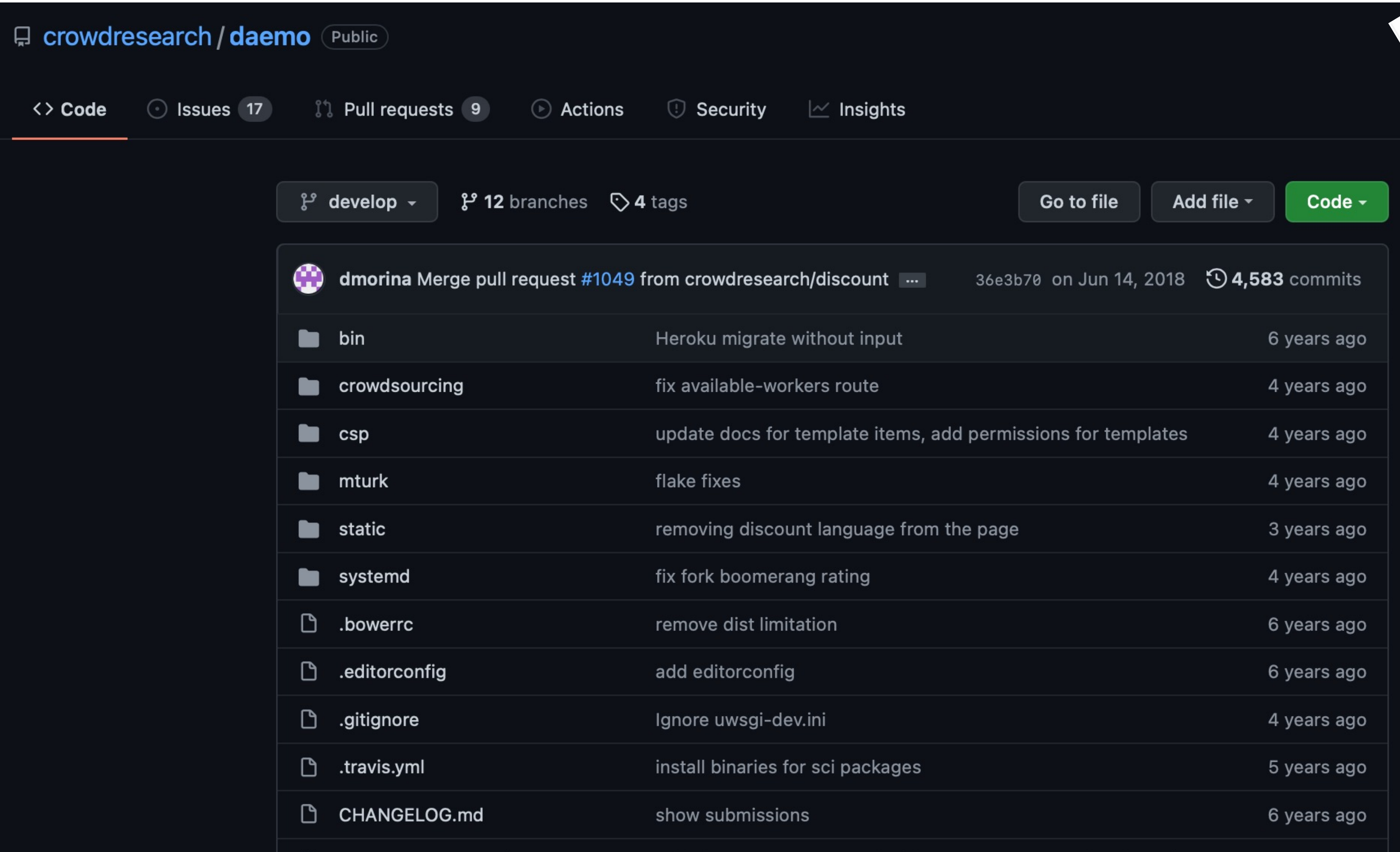


MIRANDA KATZ BACKCHANNEL 08.23.2017 06:55 AM

Amazon's Turker Crowd Has Had Enough

Workers on Amazon Mechanical Turk have long felt ignored and underpaid. A new platform hopes to give them a better home—and could change the future of crowd work.

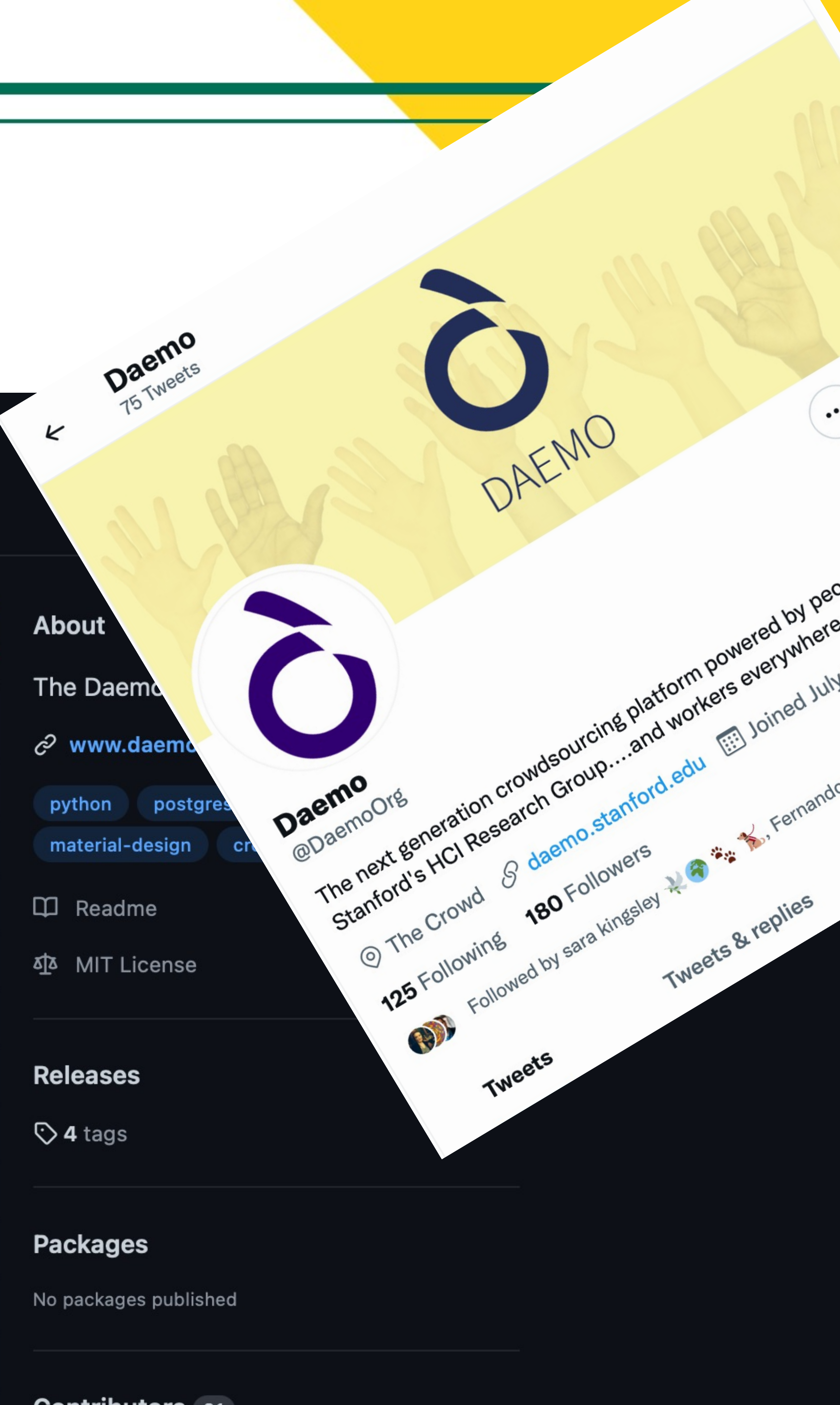
Daemo: defunct



crowdresearch / daemo Public

dmorina Merge pull request #1049 from crowdresearch/discount ... 36e3b70 on Jun 14, 2018 4,583 commits

bin	Heroku migrate without input	6 years ago
crowdsourcing	fix available-workers route	4 years ago
csp	update docs for template items, add permissions for templates	4 years ago
mturk	flake fixes	4 years ago
static	removing discount language from the page	3 years ago
systemd	fix fork boomerang rating	4 years ago
.bowerrc	remove dist limitation	6 years ago
.editorconfig	add editorconfig	6 years ago
.gitignore	Ignore uwsgi-dev.ini	4 years ago
.travis.yml	install binaries for sci packages	5 years ago
CHANGELOG.md	show submissions	6 years ago



About

The Daemo

www.daemo

Readme

MIT License

Releases

4 tags

Packages

No packages published

Daemo
75 Tweets

Daemo
@DaemoOrg

The next generation crowdsourcing platform powered by peo
Stanford's HCI Research Group....and workers everywhere

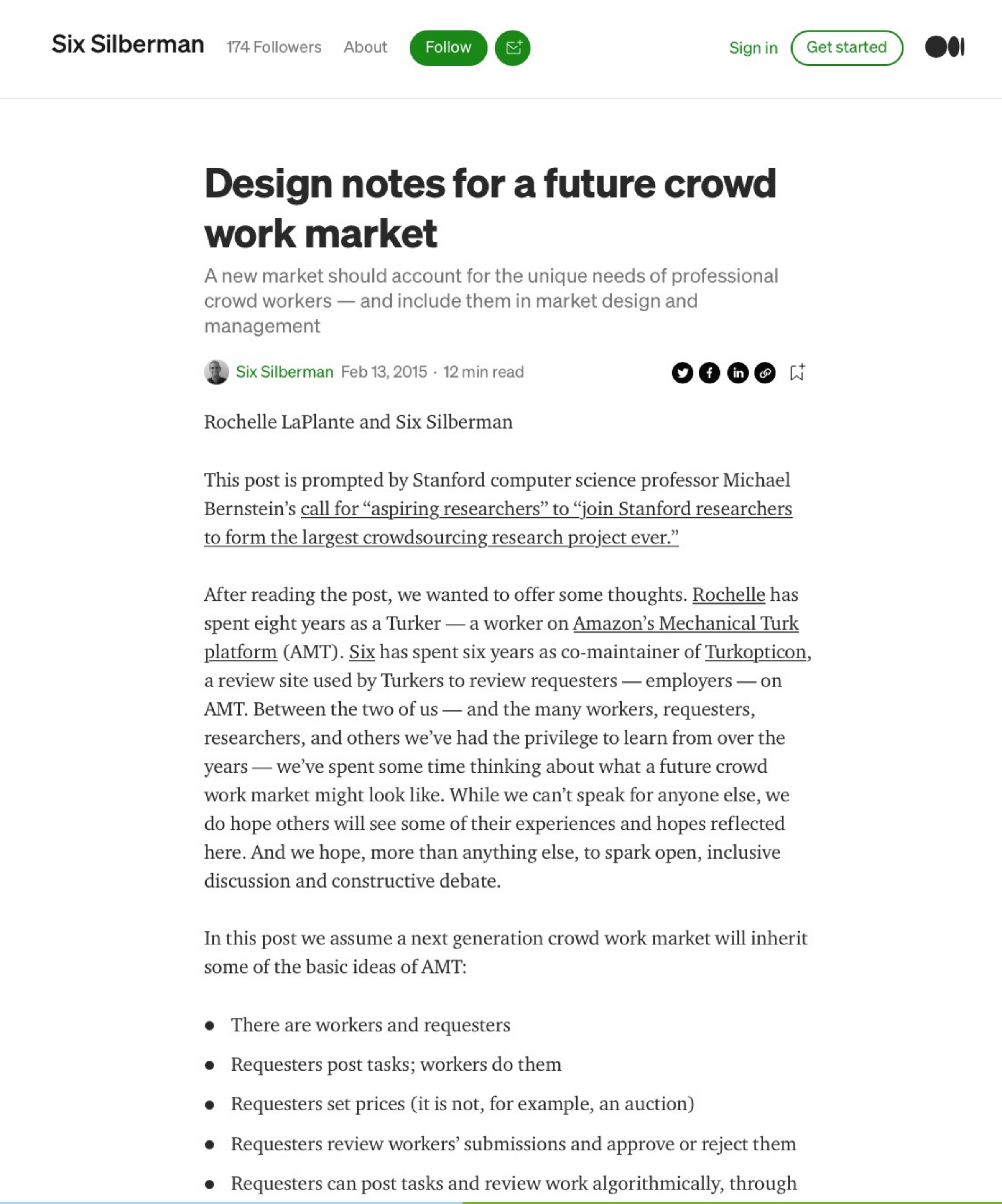
The Crowd daemo.stanford.edu Joined July

125 Following 180 Followers

Followed by sara kingsley 🕊️🌍🐾🦌, Fernando

Why?

- Feb 13, 2015 – Silberman warns about pitfalls
- Feb 20, 2015 – Bernstein responds, involves workers
- 2016 – Daemo launches
- 2017 – Daemo gets press



Design notes for a future crowd work market

A new market should account for the unique needs of professional crowd workers — and include them in market design and management

Six Silberman Feb 13, 2015 · 12 min read

Rochelle LaPlante and Six Silberman

This post is prompted by Stanford computer science professor Michael Bernstein's call for "aspiring researchers" to "join Stanford researchers to form the largest crowdsourcing research project ever."

After reading the post, we wanted to offer some thoughts. Rochelle has spent eight years as a Turker — a worker on Amazon's Mechanical Turk platform (AMT). Six has spent six years as co-maintainer of Turkopticon, a review site used by Turkers to review requesters — employers — on AMT. Between the two of us — and the many workers, requesters, researchers, and others we've had the privilege to learn from over the years — we've spent some time thinking about what a future crowd work market might look like. While we can't speak for anyone else, we do hope others will see some of their experiences and hopes reflected here. And we hope, more than anything else, to spark open, inclusive discussion and constructive debate.

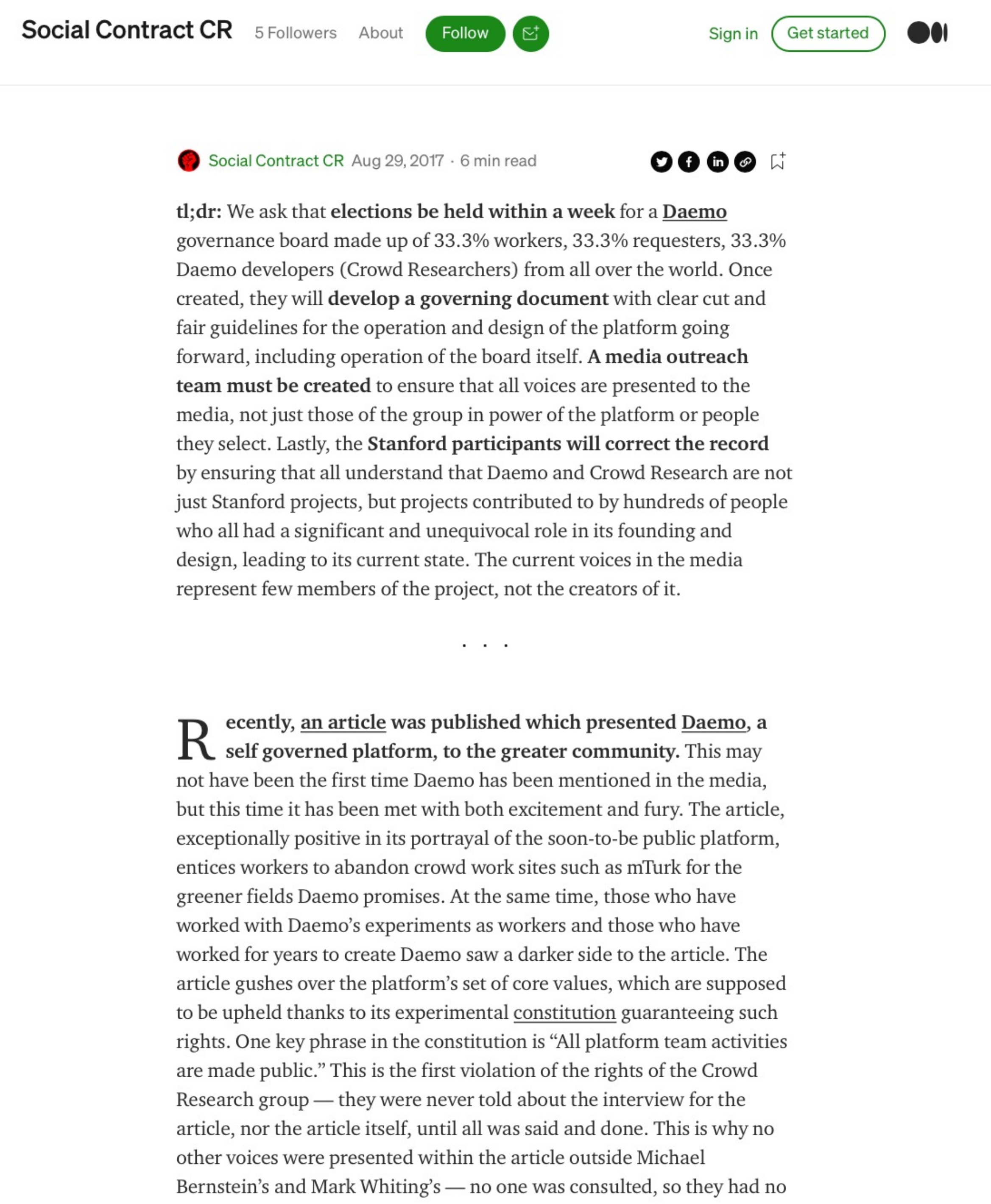
In this post we assume a next generation crowd work market will inherit some of the basic ideas of AMT:

- There are workers and requesters
- Requesters post tasks; workers do them
- Requesters set prices (it is not, for example, an auction)
- Requesters review workers' submissions and approve or reject them
- Requesters can post tasks and review work algorithmically, through

Why?

- Also 2017: workers respond to press
- Issues:
  - Lack of worker recognition
  - Misrepresentation of stakeholders in media
  - Alleged gatekeeping
  - Misaligned goals and values

Specific issues arise due to incentives of work vs. research

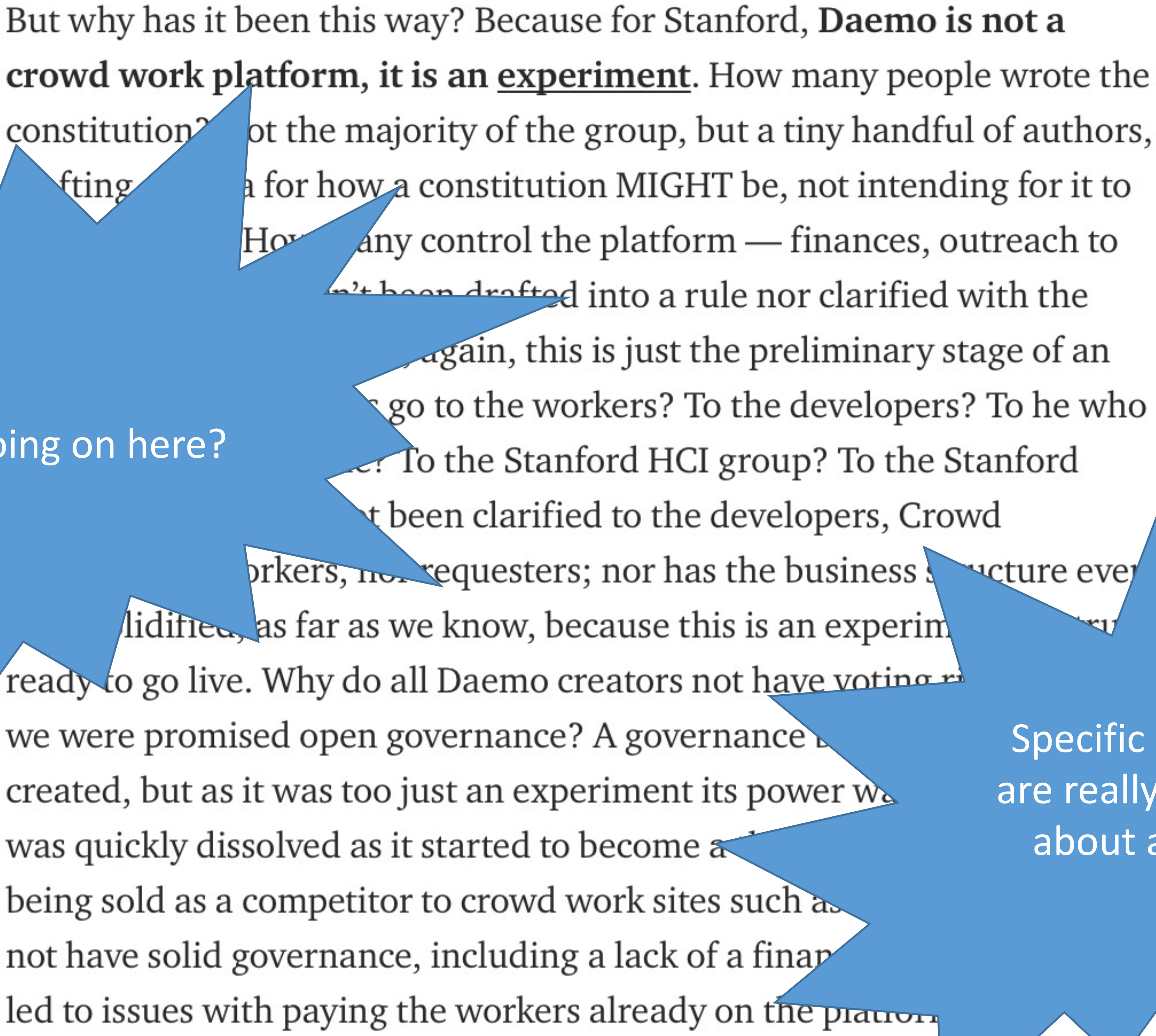


Social Contract CR Aug 29, 2017 · 6 min read

tl;dr: We ask that **elections be held within a week** for a Daemo governance board made up of 33.3% workers, 33.3% requesters, 33.3% Daemo developers (Crowd Researchers) from all over the world. Once created, they will **develop a governing document** with clear cut and fair guidelines for the operation and design of the platform going forward, including operation of the board itself. **A media outreach team must be created** to ensure that all voices are presented to the media, not just those of the group in power of the platform or people they select. Lastly, the **Stanford participants will correct the record** by ensuring that all understand that Daemo and Crowd Research are not just Stanford projects, but projects contributed to by hundreds of people who all had a significant and unequivocal role in its founding and design, leading to its current state. The current voices in the media represent few members of the project, not the creators of it.

. . .

Recently, an article was published which presented Daemo, a self governed platform, to the greater community. This may not have been the first time Daemo has been mentioned in the media, but this time it has been met with both excitement and fury. The article, exceptionally positive in its portrayal of the soon-to-be public platform, entices workers to abandon crowd work sites such as mTurk for the greener fields Daemo promises. At the same time, those who have worked with Daemo's experiments as workers and those who have worked for years to create Daemo saw a darker side to the article. The article gushes over the platform's set of core values, which are supposed to be upheld thanks to its experimental constitution guaranteeing such rights. One key phrase in the constitution is "All platform team activities are made public." This is the first violation of the rights of the Crowd Research group — they were never told about the interview for the article, nor the article itself, until all was said and done. This is why no other voices were presented within the article outside Michael Bernstein's and Mark Whiting's — no one was consulted, so they had no



But why has it been this way? Because for Stanford, **Daemo is not a crowd work platform, it is an experiment**. How many people wrote the constitution? ot the majority of the group, but a tiny handful of authors, fting a for how a constitution MIGHT be, not intending for it to Ho any control the platform — finances, outreach to n't been drafted into a rule nor clarified with the again, this is just the preliminary stage of an go to the workers? To the developers? To he who e? To the Stanford HCI group? To the Stanford t been clarified to the developers, Crowd orkers, nor requesters; nor has the business ucture ever lidified, as far as we know, because this is an experim ready to go live. Why do all Daemo creators not have voting ri we were promised open governance? A governance created, but as it was too just an experiment its power wa was quickly dissolved as it started to become a being sold as a competitor to crowd work sites such as not have solid governance, including a lack of a finan led to issues with paying the workers already on the platfor

What’s going on here?

Specific complaints are really complaints about academia.

Academic model vs. client model

Academic model:

- Asymmetric hierarchy
- Transient
- Discretionary

Client model:

- Symmetric marketplace
- Long-term stability
- Necessary

Exercise: What would a client model look like?

Most crowdwork is HCI

- Few systems-level approaches
- Some recent gestures in this direction (CSCW2021)

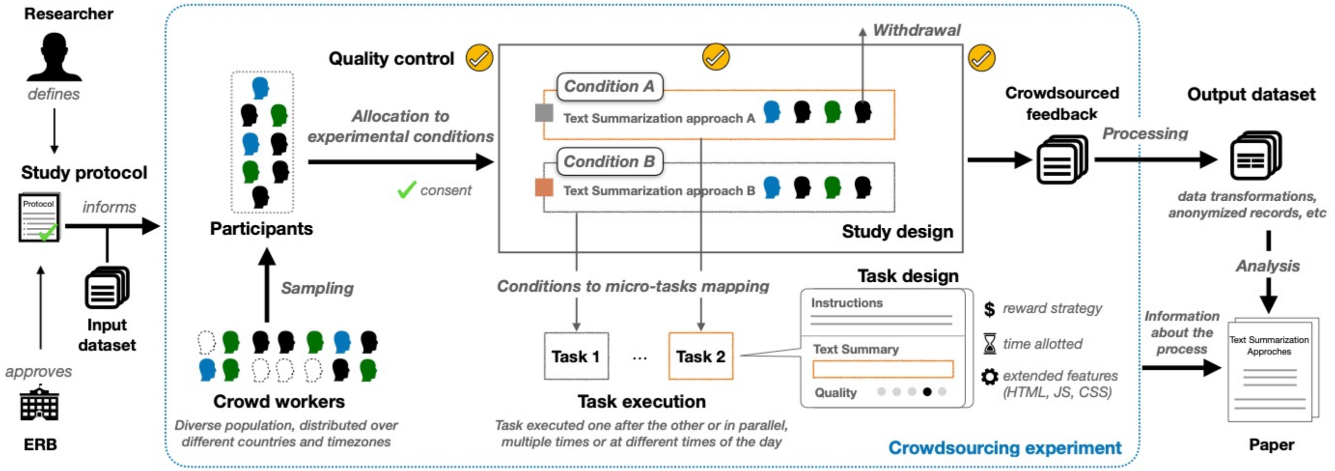
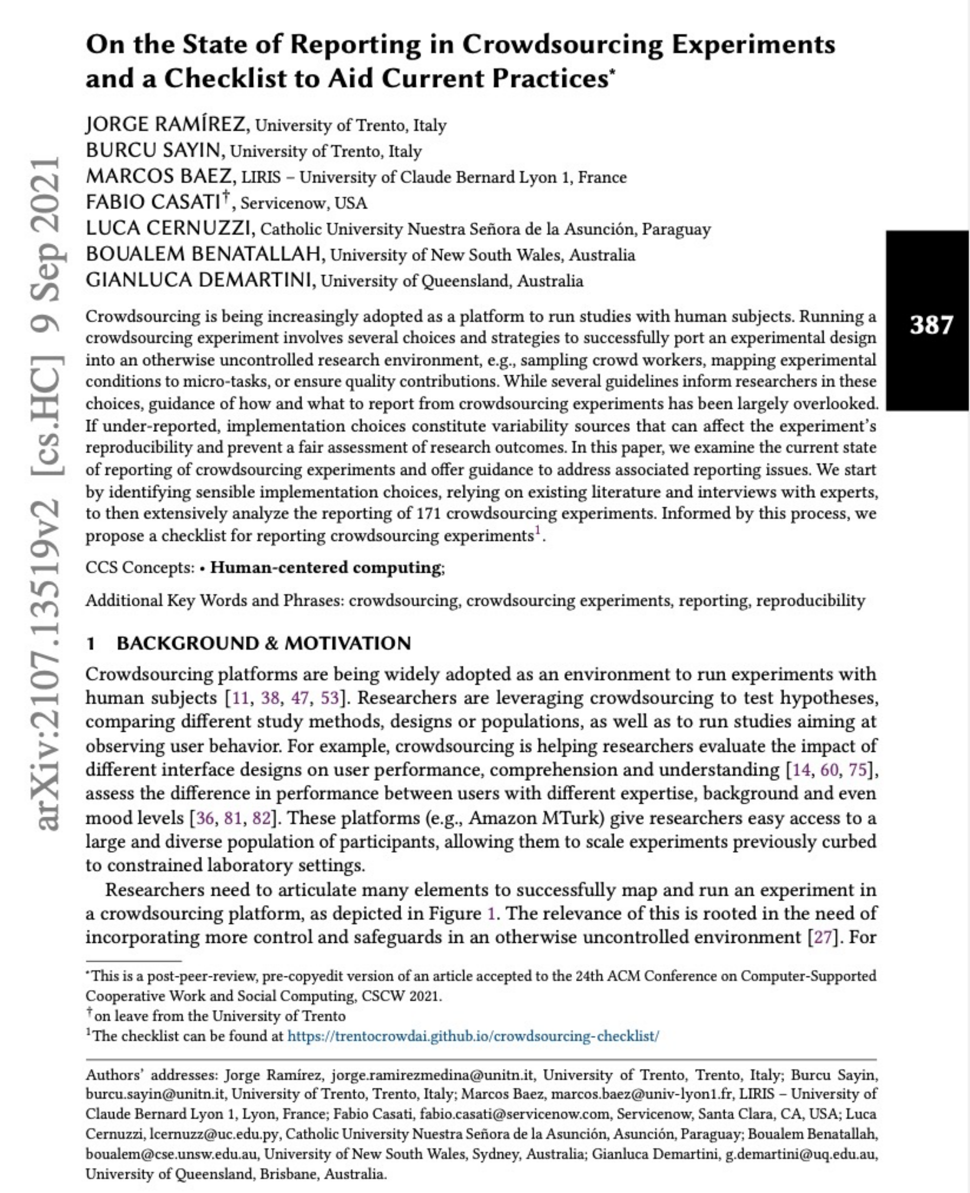

Researcher
defines
Study protocol
Protocol
informs
Input dataset
approves
ERB
Participants
Sampling
Crowd workers
Diverse population, distributed over different countries and timezones
Quality control
Allocation to experimental conditions
consent
Withdrawal
Condition A
Text Summarization approach A
Condition B
Text Summarization approach B
Study design
Conditions to micro-tasks mapping
Task 1
...
Task 2
Task execution
Task executed one after the other or in parallel, multiple times or at different times of the day
Task design
Instructions
Text Summary
Quality
reward strategy
time allotted
extended features (HTML, JS, CSS)
Crowdsourcing experiment
Crowdsourced feedback
Processing
Output dataset
data transformations, anonymized records, etc
Analysis
Information about the process
Text Summarization Approches
Paper

Fig. 1. Mapping an experimental design to a crowdsourcing platform involves articulating many elements (none of which have a unique implementation). These elements constitute sources of variability if not properly reported.



On the State of Reporting in Crowdsourcing Experiments and a Checklist to Aid Current Practices*

JORGE RAMÍREZ, University of Trento, Italy
BURCU SAYIN, University of Trento, Italy
MARCOS BAEZ, LIRIS – University of Claude Bernard Lyon 1, France
FABIO CASATI†, Servicenow, USA
LUCA CERNUZZI, Catholic University Nuestra Señora de la Asunción, Paraguay
BOUALEM BENATALLAH, University of New South Wales, Australia
GIANLUCA DEMARTINI, University of Queensland, Australia

Crowdsourcing is being increasingly adopted as a platform to run studies with human subjects. Running a crowdsourcing experiment involves several choices and strategies to successfully port an experimental design into an otherwise uncontrolled research environment, e.g., sampling crowd workers, mapping experimental conditions to micro-tasks, or ensure quality contributions. While several guidelines inform researchers in these choices, guidance of how and what to report from crowdsourcing experiments has been largely overlooked. If under-reported, implementation choices constitute variability sources that can affect the experiment's reproducibility and prevent a fair assessment of research outcomes. In this paper, we examine the current state of reporting of crowdsourcing experiments and offer guidance to address associated reporting issues. We start by identifying sensible implementation choices, relying on existing literature and interviews with experts, to then extensively analyze the reporting of 171 crowdsourcing experiments. Informed by this process, we propose a checklist for reporting crowdsourcing experiments[1].

CCS Concepts: • **Human-centered computing**;

Additional Key Words and Phrases: crowdsourcing, crowdsourcing experiments, reporting, reproducibility

1 BACKGROUND & MOTIVATION

Crowdsourcing platforms are being widely adopted as an environment to run experiments with human subjects [11, 38, 47, 53]. Researchers are leveraging crowdsourcing to test hypotheses, comparing different study methods, designs or populations, as well as to run studies aiming at observing user behavior. For example, crowdsourcing is helping researchers evaluate the impact of different interface designs on user performance, comprehension and understanding [14, 60, 75], assess the difference in performance between users with different expertise, background and even mood levels [36, 81, 82]. These platforms (e.g., Amazon MTurk) give researchers easy access to a large and diverse population of participants, allowing them to scale experiments previously curbed to constrained laboratory settings.

Researchers need to articulate many elements to successfully map and run an experiment in a crowdsourcing platform, as depicted in Figure 1. The relevance of this is rooted in the need of incorporating more control and safeguards in an otherwise uncontrolled environment [27]. For

*This is a post-peer-review, pre-copyedit version of an article accepted to the 24th ACM Conference on Computer-Supported Cooperative Work and Social Computing, CSCW 2021.

†on leave from the University of Trento

[1] The checklist can be found at https://trentocrowdai.github.io/crowdsourcing-checklist/

Authors' addresses: Jorge Ramírez, jorge.ramirezmedina@unitn.it, University of Trento, Trento, Italy; Burcu Sayin, burcu.sayin@unitn.it, University of Trento, Trento, Italy; Marcos Baez, marcos.baez@univ-lyon1.fr, LIRIS – University of Claude Bernard Lyon 1, Lyon, France; Fabio Casati, fabio.casati@servicenow.com, Servicenow, Santa Clara, CA, USA; Luca Cernuzzi, lcernuzz@uc.edu.py, Catholic University Nuestra Señora de la Asunción, Asunción, Paraguay; Boualem Benatallah, boualem@cse.unsw.edu.au, University of New South Wales, Sydney, Australia; Gianluca Demartini, g.demartini@uq.edu.au, University of Queensland, Brisbane, Australia.

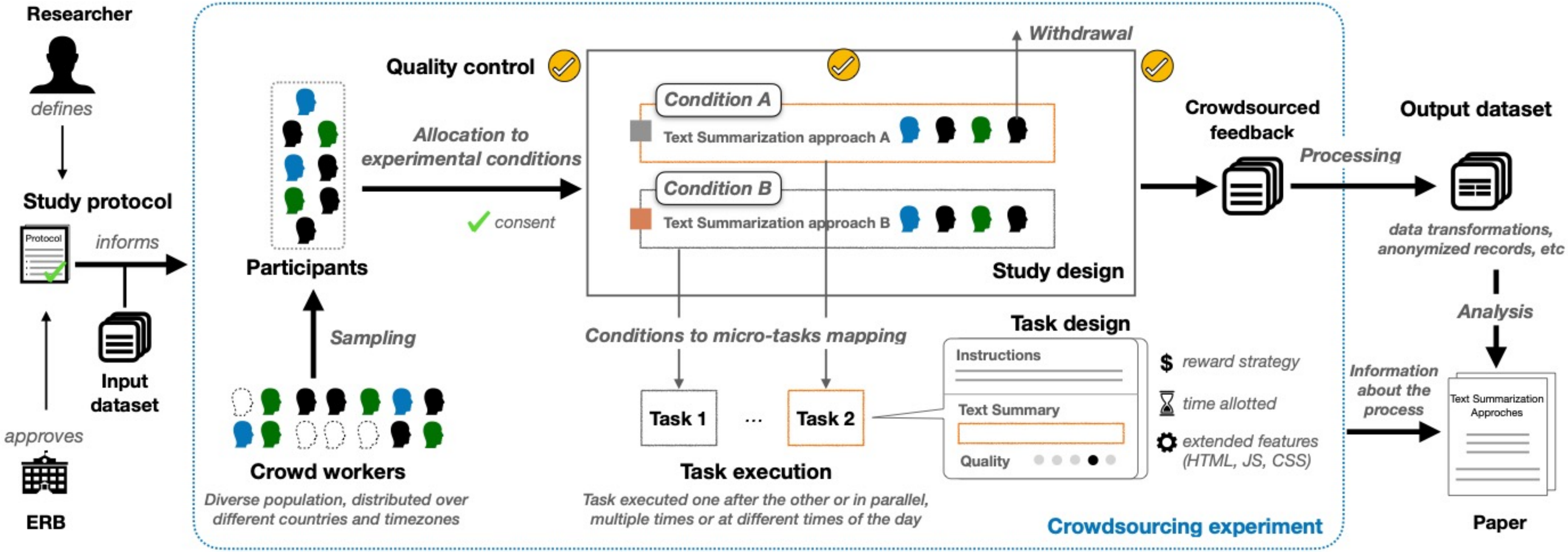

Researcher
defines
Study protocol
Protocol
informs
Input dataset
approves
ERB
Quality control
Participants
Allocation to experimental conditions
consent
Sampling
Crowd workers
Diverse population, distributed over different countries and timezones
Withdrawal
Condition A
Text Summarization approach A
Condition B
Text Summarization approach B
Study design
Conditions to micro-tasks mapping
Task 1
...
Task 2
Task execution
Task executed one after the other or in parallel, multiple times or at different times of the day
Task design
Instructions
Text Summary
Quality
reward strategy
time allotted
extended features (HTML, JS, CSS)
Crowdsourced feedback
Processing
Output dataset
data transformations, anonymized records, etc
Analysis
Information about the process
Text Summarization Approches
Paper
Crowdsourcing experiment

Fig. 1. Mapping an experimental design to a crowdsourcing platform involves articulating many elements (none of which have a unique implementation). These elements constitute sources of variability if not properly reported.

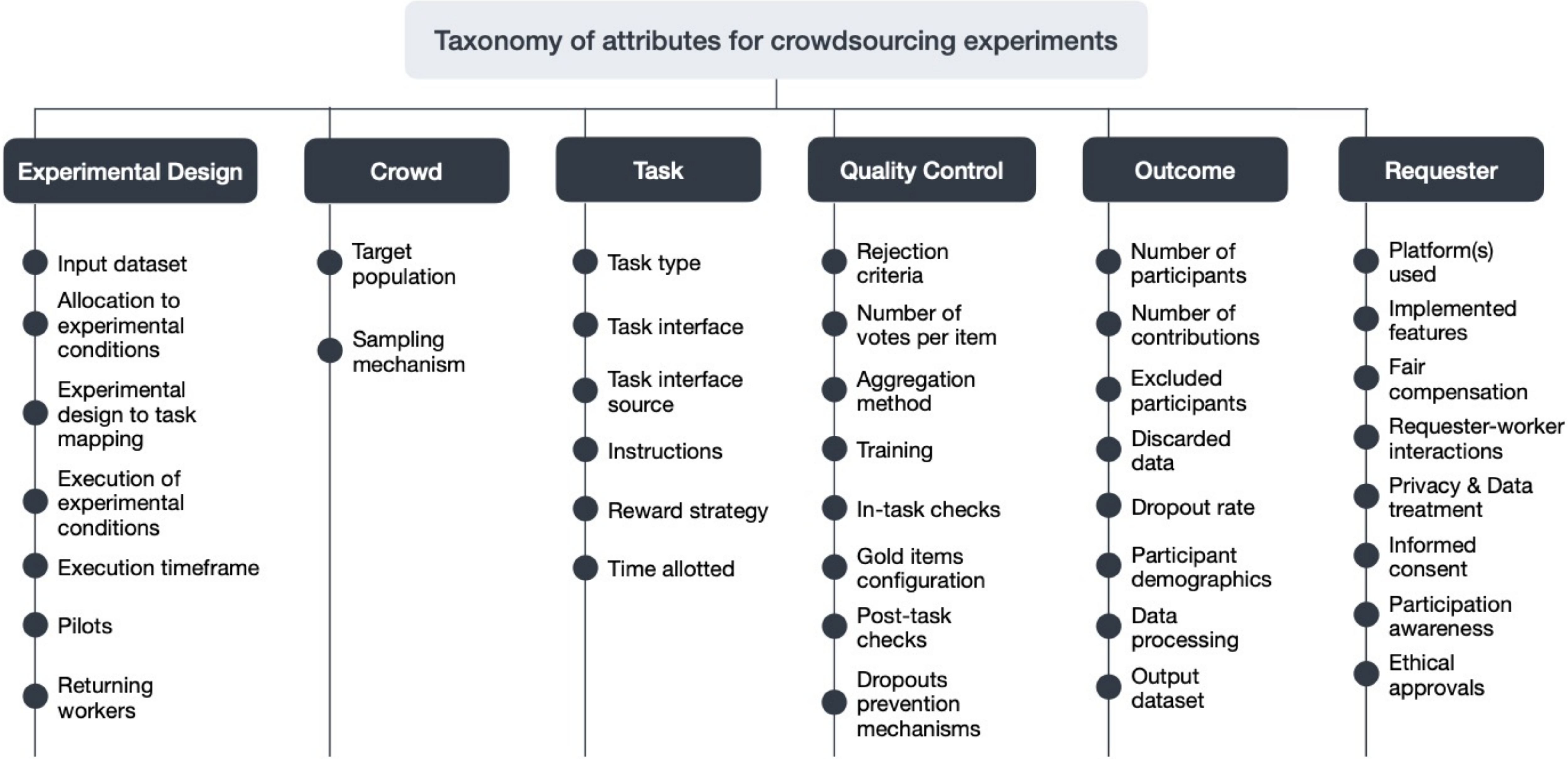
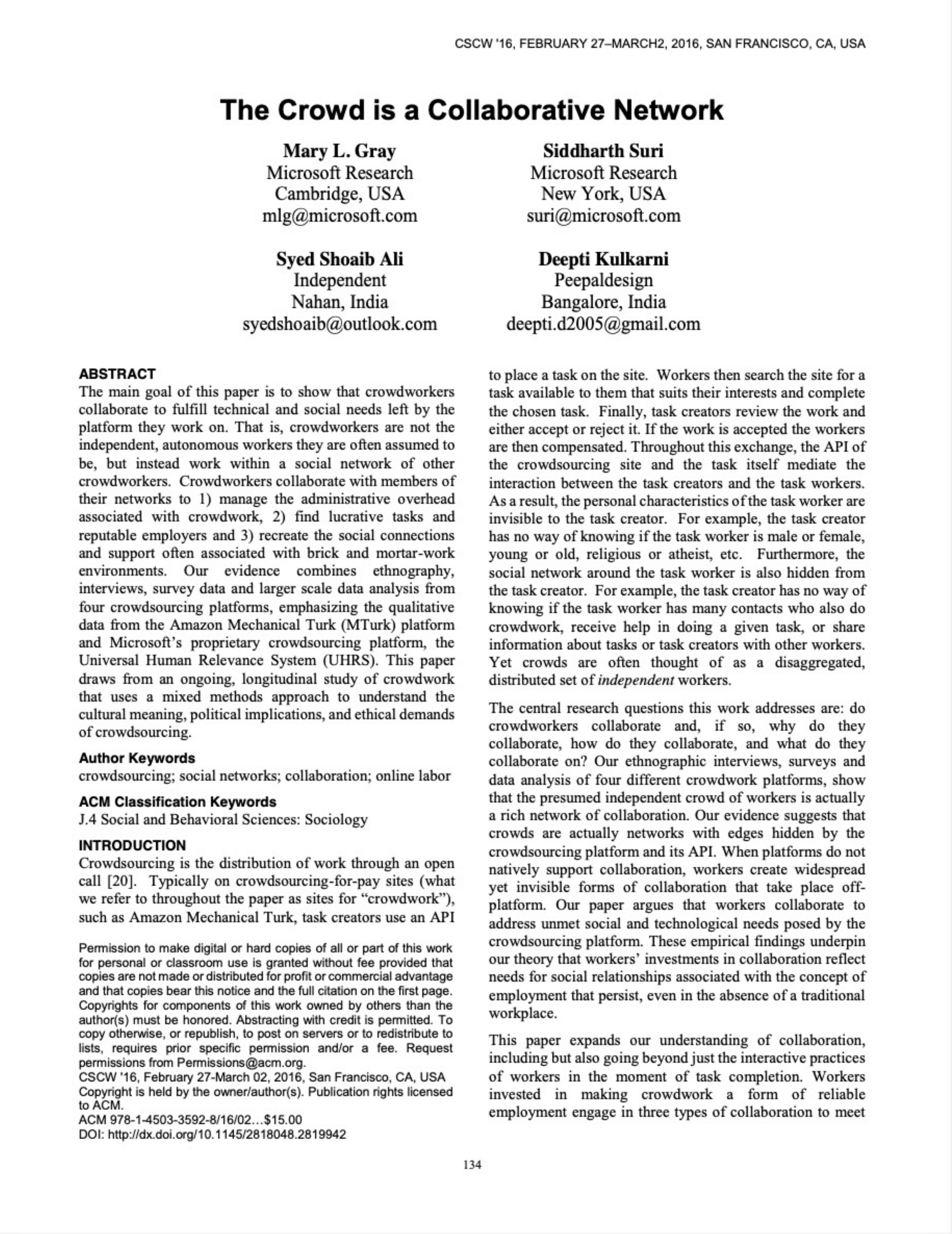


Fig. 2. A taxonomy of relevant attributes characterizing experiments in crowdsourcing.

Some HCI is social science

- Most qualitative social-science-y paper we will have read so far
- Two big ideas (why I chose this paper):
  - Computing is broad
  - Implications from this work for system design (relational)



The Crowd is a Collaborative Network

Mary L. Gray
Microsoft Research
Cambridge, USA
mlg@microsoft.com

Siddharth Suri
Microsoft Research
New York, USA
suri@microsoft.com

Syed Shoaib Ali
Independent
Nahan, India
syedshoaib@outlook.com

Deepti Kulkarni
Peepaldesign
Bangalore, India
deepti.d2005@gmail.com

ABSTRACT

The main goal of this paper is to show that crowdworkers collaborate to fulfill technical and social needs left by the platform they work on. That is, crowdworkers are not the independent, autonomous workers they are often assumed to be, but instead work within a social network of other crowdworkers. Crowdworkers collaborate with members of their networks to 1) manage the administrative overhead associated with crowdwork, 2) find lucrative tasks and reputable employers and 3) recreate the social connections and support often associated with brick and mortar-work environments. Our evidence combines ethnography, interviews, survey data and larger scale data analysis from four crowdsourcing platforms, emphasizing the qualitative data from the Amazon Mechanical Turk (MTurk) platform and Microsoft's proprietary crowdsourcing platform, the Universal Human Relevance System (UHRS). This paper draws from an ongoing, longitudinal study of crowdwork that uses a mixed methods approach to understand the cultural meaning, political implications, and ethical demands of crowdsourcing.

Author Keywords
crowdsourcing; social networks; collaboration; online labor

ACM Classification Keywords
J.4 Social and Behavioral Sciences: Sociology

INTRODUCTION

Crowdsourcing is the distribution of work through an open call [20]. Typically on crowdsourcing-for-pay sites (what we refer to throughout the paper as sites for "crowdwork"), such as Amazon Mechanical Turk, task creators use an API to place a task on the site. Workers then search the site for a task available to them that suits their interests and complete the chosen task. Finally, task creators review the work and either accept or reject it. If the work is accepted the workers are then compensated. Throughout this exchange, the API of the crowdsourcing site and the task itself mediate the interaction between the task creators and the task workers. As a result, the personal characteristics of the task worker are invisible to the task creator. For example, the task creator has no way of knowing if the task worker is male or female, young or old, religious or atheist, etc. Furthermore, the social network around the task worker is also hidden from the task creator. For example, the task creator has no way of knowing if the task worker has many contacts who also do crowdwork, receive help in doing a given task, or share information about tasks or task creators with other workers. Yet crowds are often thought of as a disaggregated, distributed set of *independent* workers.

The central research questions this work addresses are: do crowdworkers collaborate and, if so, why do they collaborate, how do they collaborate, and what do they collaborate on? Our ethnographic interviews, surveys and data analysis of four different crowdwork platforms, show that the presumed independent crowd of workers is actually a rich network of collaboration. Our evidence suggests that crowds are actually networks with edges hidden by the crowdsourcing platform and its API. When platforms do not natively support collaboration, workers create widespread yet invisible forms of collaboration that take place off-platform. Our paper argues that workers collaborate to address unmet social and technological needs posed by the crowdsourcing platform. These empirical findings underpin our theory that workers' investments in collaboration reflect needs for social relationships associated with the concept of employment that persist, even in the absence of a traditional workplace.

This paper expands our understanding of collaboration, including but also going beyond just the interactive practices of workers in the moment of task completion. Workers invested in making crowdwork a form of reliable employment engage in three types of collaboration to meet

CSCW '16, February 27-March 02, 2016, San Francisco, CA, USA

ACM 978-1-4503-3592-8/16/02…$15.00
DOI: http://dx.doi.org/10.1145/2818048.2819942

Exercise Proposal

Try to design a crowdwork system from a systems perspective.

Typical Stakeholders: Platform, Requesters, Workers

Daemo: both platform designers *and* requesters

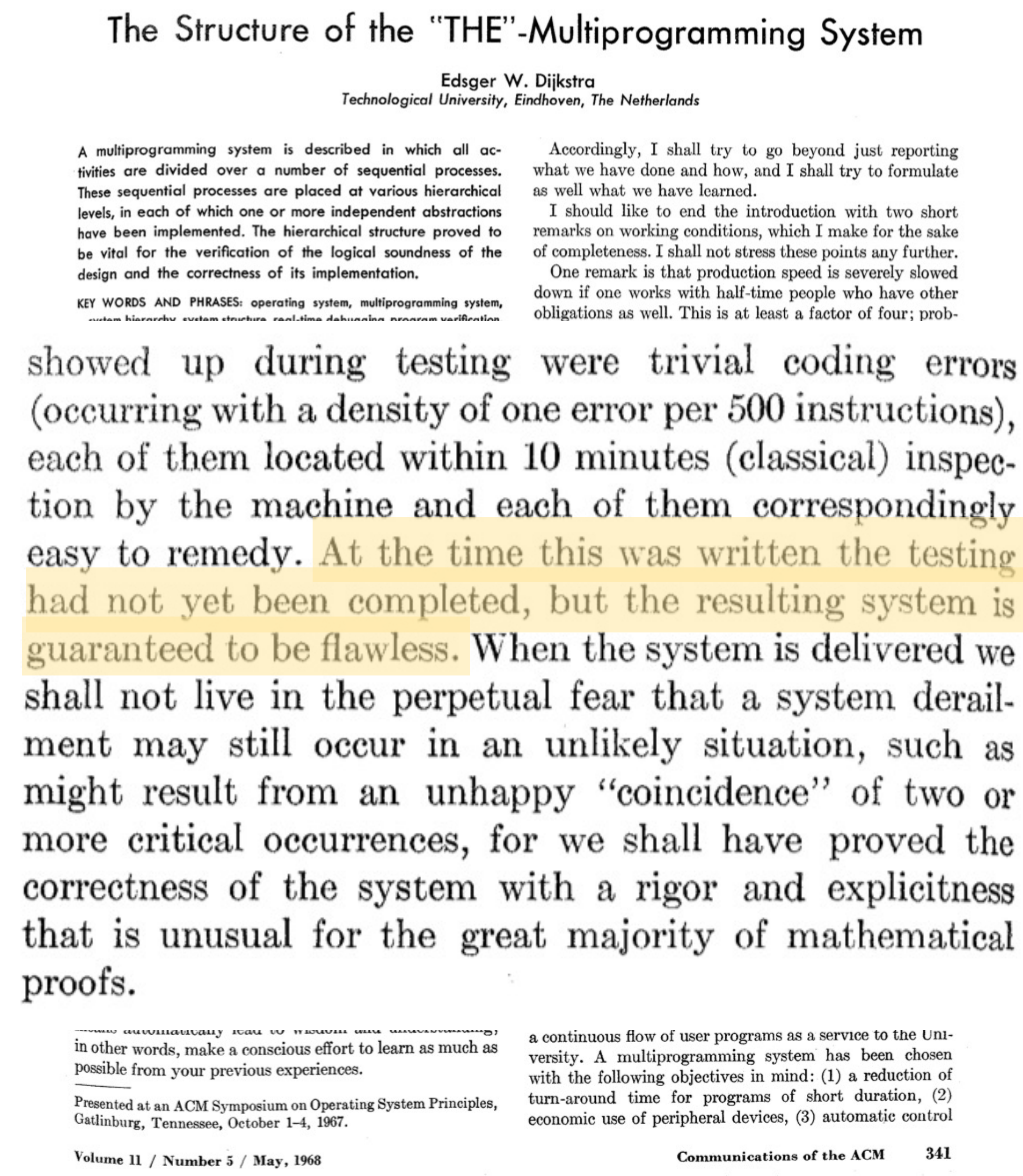
- Process of social contract design was itself research
- Always from the requesters' perspective
- What if we invert the process?

Take the THE approach*

Problem: treating AMT as opaque + assumptions

All problem solving has been *holistic about the system, component-wise about people*

Instead: holistic about people, component-wise about system



The Structure of the "THE"-Multiprogramming System

Edsger W. Dijkstra
Technological University, Eindhoven, The Netherlands

A multiprogramming system is described in which all activities are divided over a number of sequential processes. These sequential processes are placed at various hierarchical levels, in each of which one or more independent abstractions have been implemented. The hierarchical structure proved to be vital for the verification of the logical soundness of the design and the correctness of its implementation.

KEY WORDS AND PHRASES: operating system, multiprogramming system, system hierarchy, system structure, real-time debugging, program verification

Accordingly, I shall try to go beyond just reporting what we have done and how, and I shall try to formulate as well what we have learned.

I should like to end the introduction with two short remarks on working conditions, which I make for the sake of completeness. I shall not stress these points any further.

One remark is that production speed is severely slowed down if one works with half-time people who have other obligations as well. This is at least a factor of four; prob-

showed up during testing were trivial coding errors (occurring with a density of one error per 500 instructions), each of them located within 10 minutes (classical) inspection by the machine and each of them correspondingly easy to remedy. At the time this was written the testing had not yet been completed, but the resulting system is guaranteed to be flawless. When the system is delivered we shall not live in the perpetual fear that a system derailment may still occur in an unlikely situation, such as might result from an unhappy "coincidence" of two or more critical occurrences, for we shall have proved the correctness of the system with a rigor and explicitness that is unusual for the great majority of mathematical proofs.

means automatically lead to wisdom and understanding; in other words, make a conscious effort to learn as much as possible from your previous experiences.

a continuous flow of user programs as a service to the University. A multiprogramming system has been chosen with the following objectives in mind: (1) a reduction of turn-around time for programs of short duration, (2) economic use of peripheral devices, (3) automatic control

Presented at an ACM Symposium on Operating System Principles, Gatlinburg, Tennessee, October 1–4, 1967.

* In a previous lecture, I referred to Multics but actually meant the THE system

Proposal: High level

1. Enumerate required software components
2. Enumerate properties software components must have
3. Discuss implementation logistics
4. Compare against known issues without software solutions

Walk through what interaction with the system looks like, from a *worker's* perspective

Walk through what interaction with the system looks like, from a *requester's* perspective

Some possible components of a crowdwork system

- Identity & reputation management (both workers and requesters)
- HIT server/display (how available HITs are displayed)
- HIT bidding/claiming (how workers select jobs)
- HIT flaw reporting system
- Appeals system for rejected HITs
- Requester training/qualifications
- System-side estimates of return (ML cuts both ways!), ski rental problem
- Explicit support for collaboration

Proposal: High level

~~1. Enumerate required software components~~
2. Enumerate properties software components must have
3. Discuss implementation logistics
4. Compare against known issues without software solutions

Assertions and contracts

- Assertions: dynamic (as opposed to static) checks
- Preconditions (expects): assert s.t. is true before executing a block
- Postconditions (ensures): assert s.t. is true after executing a block
- Static analysis: “contracts,” Hoare logic ({P: expects} C {Q: ensures})

What properties must these components have?
(in the model of {P} C {Q})

Some possible properties

- Fariness properties of HIT display (both worker and requester perspectives)
- No cross-origin collusion
- Bound the information content of a HIT (atomicity of a task)
- HITs do not leak information about workers
- Target flow of tasks (juice/throttle?)

Proposal: High level

1. ~~Enumerate required software components~~
2. ~~Enumerate properties software components much have~~
3. Discuss implementation logistics
4. Compare against known issues without software solutions

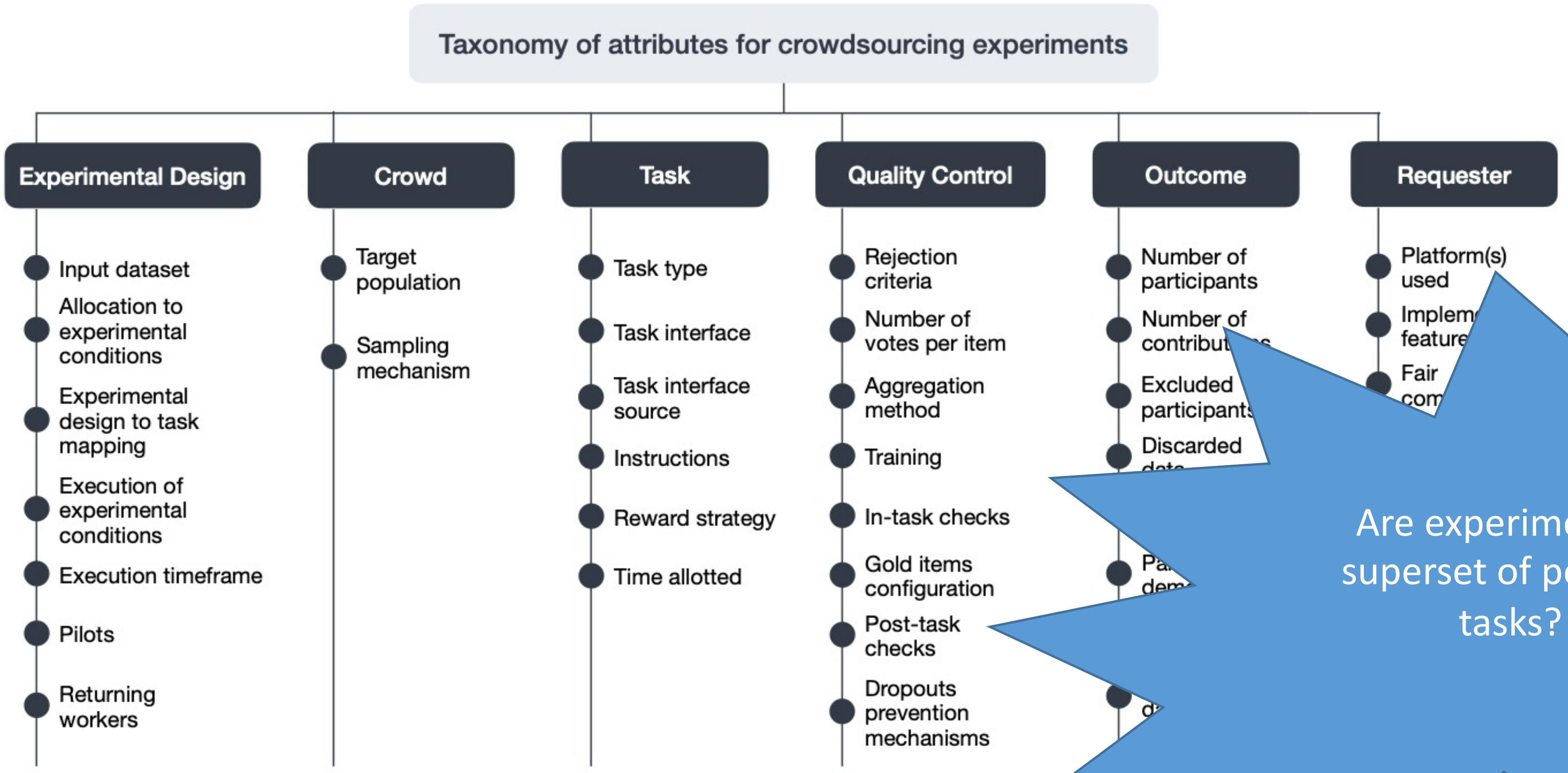
Which pieces happen server-side? Client-side?

Which components communicate with each other?

Which pieces are automated vs. human-in-the-loop?

Which pieces are pluggable/swappable and from whose perspective? Platform only? Worker? Requester?

Did we consider all of these?



Taxonomy of attributes for crowdsourcing experiments

- **Experimental Design**
  - Input dataset
  - Allocation to experimental conditions
  - Experimental design to task mapping
  - Execution of experimental conditions
  - Execution timeframe
  - Pilots
  - Returning workers
- **Crowd**
  - Target population
  - Sampling mechanism
- **Task**
  - Task type
  - Task interface
  - Task interface source
  - Instructions
  - Reward strategy
  - Time allotted
- **Quality Control**
  - Rejection criteria
  - Number of votes per item
  - Aggregation method
  - Training
  - In-task checks
  - Gold items configuration
  - Post-task checks
  - Dropouts prevention mechanisms
- **Outcome**
  - Number of participants
  - Number of contribut
  - Excluded participants
  - Discarded data
  - Pa dem
  - d
- **Requester**
  - Platform(s) used
  - Implem feature
  - Fair com

Fig. 2. A taxonomy of relevant attributes characterizing experiments in cr urcing.

Are experiments a superset of possible tasks?

Proposal: High level

1. ~~Enumerate required software components~~

2. ~~Enumerate properties software components much have~~

3. ~~Discuss implementation logistics~~

4. Compare against known issues without software solutions

Known worker concerns: McInnis et al.

Flaws in task or interface design

Unclear evaluation criteria

Unresponsive, arbitrary resolution of rejections

Lack of information on Requesters

Inexperienced and Unfamiliar Requesters

Tasks with poor return

Prioritizing efficiency over quality

Thematically: working conditions vis-à-vis pay and uncertainty

Known requester concerns

Difallah et al.

- Population skew of workers

Barowy et al.

- Random respondents

Tosch and Berger

- Inattentive respondents

Thematically: cost of unusable data

Concluding notes

Responsible testing

- Simulation-based
  - Model collective behavior
  - Treat as an iterative process
  - Transparency reports
- Paradigm shift: platform views workers & requesters as beta-testers for platform

Why crowdsourcing matters

Yes, it powers many data-driven applications

But also:

1. Technical + societal microcosm of the future of data+infrastructure
2. Cautionary tale of CS (and academic) hubris
3. Many core computer science problems not currently addressed as such