It's Not the Whiteboard We Missed, It's How It Made Us Feel

Elaine M. Raybourn*

Applied Cognitive Science | Sandia National Laboratories

Abstract

In the domains of research software and science, geographically dispersed, virtual, and remote software teams are nothing new—and likely here to stay. Nevertheless, during the COVID-19 pandemic, many scientists and software teams who transitioned from co-located offices to working remotely fulltime often lamented about missing the whiteboard most of all during the pandemic. So, was it the whiteboard that we missed, or the *way* we used it? Could we even go so far to say that we missed how it made us feel? As some of us now head back to co-located or hybrid office configurations, let's contemplate the role of the whiteboard in many software teams and give it the props it deserves.

Keywords: Whiteboard, communication, software teams.

1 Introduction

Approximately 16 months ago and in response to the COVID-19 pandemic, the Exascale Computing Project's Interoperable Design of Extreme-scale Application Software (IDEAS-ECP) launched the panel series *Strategies for Working Remotely* [1]. The panels are designed to promote informal, cross-organizational dialog and community building. As one of the co-organizers and moderators of the panel series, I've had the pleasure of facilitating a variety of discussions around the topic of remote work—from parenting while working remotely, transitioning to and incorporating best practices of virtual (geographically dispersed) software teams, virtual onboarding and mentoring, virtual internships, and the impact of remote work on scientific creativity and innovation [2]. In almost every panel discussion in our series, the subject of the whiteboard came up. There were many questions about online collaborative tools and support for brainstorming or mind mapping. "The Whiteboard" was by far, the question on everyone's minds – and a satisfying answer eluded us all. I have yet to hear that someone has found an online collaboration tool whose use is as satisfying as that fuzzy feeling one gets from being at the whiteboard. We all missed it, apparently. But did we really? Isn't what we *really* missed about the whiteboard the kinds of interactions and experiences that happened with others while we were *together*, in front of it? Or did we just miss how it made us feel?

Being an artist turned social scientist, I am not only fascinated by designs and drawing, but I'm also fascinated by people, how we think, communicate, and the artifacts we produce. In this spirit I hope the reader will find this short piece introducing the some of the ways software teams use whiteboards and how they make us feel like kids again as much fun to read as it was for me to write.

2 The Whiteboard – Decades of CS Research

It may not surprise you to know that the whiteboard, or rather, computational support for interactive whiteboards, has been the subject of many dissertations and decades of computer science research—with some of the earliest exploration in computational tools for sketch recognition and management dating back from the early 1960s [3]. Teams of designers and developers over several decades have studied informal sketching practices at the whiteboard to inform the development of digital pens [4] collaborative tools [5], [6], models [7], large displays [8], and groupware [9]. Decades of research notwithstanding, interactive whiteboards are still not widely used in practice [8]. So, what gives?

2.1 How Software Teams Use Whiteboards

Software development is deeply social. Research conducted at Microsoft found that developers share and maintain mental models largely through face-to-face communication and the code itself [10]. Many participants in this study reported avoiding email or formal design documents (including bug reports, specs, etc.) to generate or transfer knowledge among teams. Developers in this study also reported rarely using IM for code-related tasks, and instead used IM to connect socially with colleagues or family. If they needed to work out a problem, they were most likely to interrupt developers who were most knowledgeable about the issue. These one-on-one developer conversations often happen at the whiteboard [5].

Informed by a review of the literature, Mangano and others distilled 14 whiteboard behaviors of software developers and designers engaged in informal software design [11]. According to the literature, teams of designers and developers use the whiteboard to communicate both synchronously and asynchronously by quickly sketching diagrams, models, and ideas while at the whiteboard, taking a step back to see the big picture, and evolving these informal designs as needed. Since software development is largely abstract, whiteboard sketches allow teams to align different perspectives, develop common mental models, review, and explore alternate hypotheses. Work at the whiteboard, while highly creative is also transient and ephemeral in that the visualizations often have little value after a task [5]. Nevertheless, the feeling of being unencumbered by an interface is valuable in and of itself. I would argue we are often in a moment of *flow* at the whiteboard—sharing ideas, seamlessly moving from thought to action as the whiteboard experience requires that we sit, stand, speak, look closely, step back and look again, grab a different colored marker, and erase. After working remotely for over 16 months, who wouldn't miss this—the joy of informality, immediacy, and impermanence—all the attributes that elude us, especially after working hour upon hour in front of a screen, often on video, and rarely sketching.

Over the 16-month period that most of us were working from home, I started to see more whiteboards emerge in my colleagues' backgrounds than guitars. Some even brought whiteboards from their office home. Analogous perhaps to Amazon's "empty chair" that signals a customer seat at the table [13], the whiteboard at home has become a prop recreating that sacred space welcoming one-on-one developer conversations.

* Send your whiteboard stories to remote@acm.org. Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA-0003525. This work was supported by the U.S. Department of Energy Office of Science, Office of Advanced Scientific Computing Research (ASCR), and by the Exascale Computing Project (17-SC- 20-SC), a collaborative effort of the U.S. Department of Energy Office of Science and the National Nuclear Security Administration. SAND2021-7382 O.

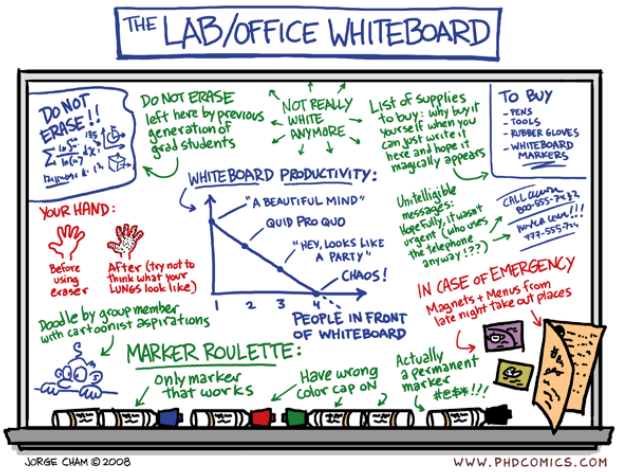

Figure 1: Raise your hand if your whiteboard has looked like this.

2.2 Whiteboards Make Us Feel Like Kids Again

Science is serious stuff but who's to say that the whiteboard experience doesn't remind us of those days in preschool or early elementary school when we were all artists, and nobody criticized our drawings? We were free to express ourselves. Perhaps now as adults, when we pick up a marker at the whiteboard it is almost like a crayon, and we feel like kids again. Think about it, when was the last time your diagram, or sketch at the whiteboard was criticized (not the idea, but the drawing)? In all seriousness though, informal design is an important, yet ephemeral, part of software development. It helps to know that the whiteboard sketches are not contracts.

Earlier in my career while a PhD student at the University of New Mexico, my colleagues and I conducted a study on single display groupware (SDG) that was informed by traditional sketching [9]. Single display groupware is a software model enabling users to simultaneously perform actions, each using input devices while at a single computer display. At the time of our research, computers in education were all about "one kid controlling the application while all others sit and wait" [9]. For our study, we employed the SDG architecture to develop a groupware version of Kidpad [12], a story creation tool for children. Our version of Kidpad allowed two users to simultaneously draw, move graphical shapes, erase, and configure tool color and pen width.

We then studied the collaborative benefits of the application with 72 elementary school children over 28 sessions We paired children together to engage in collaborative design tasks at computers that enabled one input device or two. In our control condition (one input device), passive users often attempted to physically manipulate screen objects, they barked out orders to the user with the input device, and they became irritated and frustrated at least once every 10 minutes. Children in the multi-input device condition solicited help and engaged in fewer verbal commands. Partners demonstrated how to perform actions, and if needed gave up control of their input device. In summary, we learned that no one, especially a kid, likes being passive in front of a display while collaborating [9].

3 Conclusion

The whiteboard has been the subject of decades of research, and cartoons (see Figure 1). Even so, it's magic continues to elude us. When you think about it, the whiteboard is the ultimate interface. We work alongside each other, without having to "give control" or worry that our sketches will be criticized. Heck, we can erase them immediately if we want. There is no expectation from the whiteboard and no moment more important than the one at hand. At the whiteboard we are free, we are kids again.

References

[1] E. M. Raybourn. Why we need strategies for working remotely. Preprint. DOI: https://doi.org/10.6084/m9.figshare.14681433.v1 figshare, 2021.

[2] E. M. Raybourn. Why we need strategies for working remotely: The Exascale Computing Project (ECP) panel series. State of the Practice Talk, International Conference for High Performance Computing, Networking, Storage, and Analysis, SC21. November 17, 2020.

[3] G. Johnson, M. D. Gross, J. Hong, and E. Yi-Luen Do. Computational Support for Sketching in Design: A Review. *Foundations and Trends in Human-Computer Interaction*, 2, 1 (January 2009), 1–93. DOI: https://doi.org/10.1561/1100000013

[4] M. Sra, A. Lee, S. Pao, G. Jiang, and H. Ishii. Point and share: from paper to whiteboard. In *Adjunct proceedings of the 25th annual ACM symposium on User interface software and technology* (UIST Adjunct Proceedings '12). ACM 2012. New York, NY, USA, 23–24. DOI: https://doi.org/10.1145/2380296.2380309

[5] M. Cherubini, G. Venolia, R. DeLine, and A. J. Ko. Let's go to the whiteboard: how and why software developers use drawings. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '07). ACM 2007, New York, NY, USA, 557–566. DOI: https://doi.org/10.1145/1240624.1240714

[6] S. Branham, G. Golovchinsky, S. Carter, and J. T. Biehl. Let's go from the whiteboard: supporting transitions in work through whiteboard capture and reuse. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '10). ACM. 2010. New York, NY, USA, 75–84. DOI: https://doi.org/10.1145/1753326.1753338

[7] A. Motta, N. Mangano, and A. van der Hoek. Lightweight analysis of software design models at the whiteboard. In *Proceedings of the 5th International Workshop on Modeling in Software Engineering* (MiSE '13). IEEE Press, 18–23. 2013.

[8] E. M. Huang, E. D. Mynatt, D. M. Russell, and A. E. Sue. Secrets to Success and Fatal Flaws: The Design of Large-Display Groupware. *IEEE Computer Graphics and Applications 26*, 1 (2006), 10–17.

[9] J. Stewart, E. M. Raybourn, B. Bederson, and A. Druin. When two hands are better than one: enhancing collaboration using single display groupware. In *CHI 98 Conference Summary on Human Factors in Computing Systems* (CHI '98). ACM, 1998. New York, NY, USA, 287–288. DOI: https://doi.org/10.1145/286498.286766

[10] T. D. LaToza, G. Venolia, and R. DeLine. Maintaining mental models: a study of developer work habits. In *Proceedings of the 28th international conference on Software engineering* (ICSE '06). ACM ICSE 2006, New York, NY, USA, 492–501. DOI: https://doi.org/10.1145/1134285.1134355

[11] N. Mangano, T. D. LaToza, M. Petre, and A. van der Hoek. Supporting informal design with interactive whiteboards. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '14). ACM, 2014. New York, NY, USA, 331–340. DOI: https://doi.org/10.1145/2556288.2557411

[12] A. Druin, J. Stewart, D. Proft, B. Bederson, and J. Hollan. 1997. KidPad: a design collaboration between children, technologists, and educators. In *Proceedings of the ACM SIGCHI Conference on Human factors in computing systems* (CHI '97). Association for Computing Machinery, New York, NY, USA, 463–470. DOI: https://doi.org/10.1145/258549.258866

[13] J. Koetsier. Why every Amazon meeting has at least 1 empty chair. Inc. https://www.inc.com/john-koetsier/why-every-amazon-meeting-has-at-least-one-empty-chair.html. Last retrieved June 21, 2021.