


The Future of Hybrid Software Development: Challenging Current Assumptions

Kieran Conboy , University of Galway and Lero

Nils Brede Moe , SINTEF

Viktor Stray , University of Oslo

Jan Henrik Gundelsby , Knowit

THE GLOBAL COVID-19 pandemic has been challenging for the industry, including software development organizations, causing many to revisit their procedures and assumptions about how work is conducted.¹ In particular, there was a dramatic rise in working from home (WFH), first driven by necessity and law due to the pandemic but now due to preference.^{2, 3} It is also clear that WFH is not something that will dissipate in the long term. A recent survey of 1,380 software developers found that only 3% plan to return to the office full time, 25% will remain fully remote, and 56% favor a hybrid approach, returning to the office regularly but not daily.³ Consequently, companies, such as Facebook, Square, Shopify, and Slack, have established policies of long-term and even permanent WFH.⁴ Spotify announced a work-from-anywhere (WFX) policy that allows employees to choose how often they prefer to be in the office or at home, even permitting them to move to a country of their choice.⁵ Therefore, many software development environments are, and will increasingly be, places of “hybrid software development.” While the concept of hybridity is often interpreted in many ways, for the purposes of this editorial, we use the following definition: hybrid software development is where some team members work mostly or completely from home, others mostly or completely from the traditional office, and others in some combination of the two—not quite distributed and not quite colocated but, rather, individuals working from anywhere and touching base with the office intermittently. Team members will probably have the flexibility to choose, at least to some degree, between remote and office-based work,¹ and in a large-scale context, teams will be set up differently.

Discussions around hybrid development are emerging through a variety of forums, such as journal papers, blogs, tracks at workshops and conferences, and “pop-up” events. As shown in Figure 1, the number of journal and conference papers has rapidly increased since 2020 as the research community rapidly responded to the new work situation caused by the pandemic. The search also showed the extent of research undertaken in different countries. Although the majority of the articles originate in the United States, India, and Western Europe, it has been a research theme on all continents, in a total of 89 countries. Many publications are suggesting new ways of working in a hybrid environment.

However, despite this dramatic rise in research activity, there is still a lot more to be understood about hybrid development to provide rigorous and relevant evidence-based guidance for practice. Also, in the preparatory reading that we did for this special issue and from our collective experiences working in and with organizations living and breathing hybrid development, we found that there are a number of fundamental assumptions that, for various reasons, now guide and

shape the research and practice of hybrid development.

This special issue addresses these assumptions in three ways. First, the articles selected for this special issue cover a wide variety of exciting topics regarding the future of hybrid software development and do so in an evidence-based manner. These articles will be discussed later. Second, we have included, in this editorial, two invited reflections from practitioners who have carefully reflected on hybrid development and its underlying implications for our community. Third, we have reviewed current research and practice to critically examine these prevailing assumptions, drawing on problematization approaches used by other seminal critical researchers.⁶ We are not saying these assumptions are necessarily wrong, at least in all hybrid settings, but we do argue that there is a need to at least challenge and refine them so that future developers and development managers can be more informed when thinking about organizing and executing hybrid development. These emerging assumptions are now discussed in turn, and we propose a new recommendation for each assumption (see Table 1).

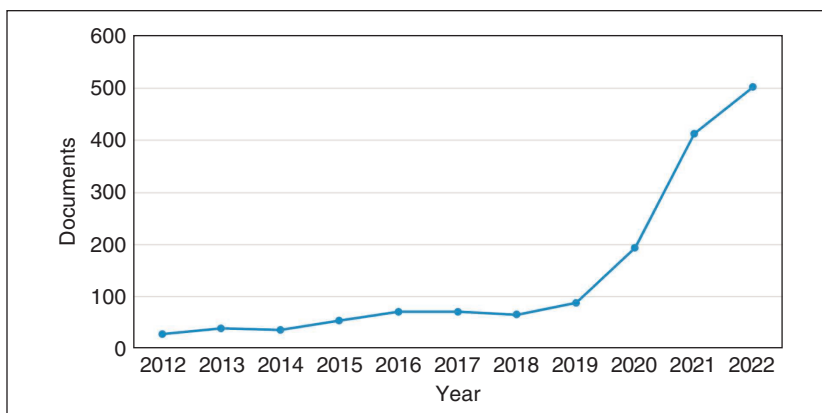


FIGURE 1. The total papers on WFH, hybrid work, and related topics, 2012–2022.

Recommendation 1: Consider Event-Based Hybrid Organization With or Instead of Calendar-Based Organization

Current approaches to hybrid work typically center on a calendar approach, e.g., “work from home on Fridays” and “work mornings in the office.” Various human resource (HR) legislation around the world follows a similar trend, where employees are entitled to work remotely for a certain number of hours or days per week or weeks per month or year.⁷ However, an alternative mode of organization is one that is event based, where a team and its activities are organized in response to certain trigger events rather than a particular time of day or week. A medical emergency, an earthquake, and a sudden fire are examples of event-based activity because people use the event as a reference point for things that happen before and after. Software development is, in many ways, an event-based activity, where teams organize their activities in response to certain events, e.g., a sudden systems failure, a changed customer requirement, and a new person joining the team. Therefore, we suggest that, rather than organizing

hybrid work around certain times and days, it may be advisable to encourage everyone to be on site when or immediately after certain events occur or, indeed, that everyone can work from home during and after other activities and events. In this special issue, Sporse et al.^{A1} provide strong advice for event-based hybrid work, where the use of communication technology is tailored and used specifically to allow team interaction around unscheduled and, often, emergency events.

Recommendation 2: Think About More Than One Way of Doing Hybrid Development

Many companies want to understand what the best practices are for doing hybrid: how many days at home, what processes to use, and what technology to use. However, within an organization, people are doing different tasks and have different preferences when it comes to flexibility. The way of doing hybrid must be adjusted to the people and task. And you can't know before you have experimented with an approach. Experiments need to be performed at the individual, team, and organization level, and they need to be data driven.

We need to move away from best practice thinking to continuous experimentation (based on data). To help, Wang et al.,^{A2} in this special issue, introduce a novel approach that can help organizations create their own paradigm of hybrid work via a bottom-up approach and further encourage a continuous improvement of an organization's own way of work. The article shows examples from author's own findings, such as the importance of mixing highly collaborative and close-proximity work with deep work, as this is substantially ingrained in their organization's culture.

Recommendation 3: Consider the Fluidity of Hybrid Development

Once an appropriate method is selected, there is a need to consider the fluid and dynamic nature of hybrid work. Many discussions assume that hybrid is some fixed and binary concept, that, for example, everyone in a particular office may work from home on Fridays and that everyone is in the office on certain days and at certain times. The reality is that development is highly fluid. Let's take a 20-person team as an example. On any given day, it is very unlikely all 20 member will be in the office and, indeed, that all 20 will work from home. Nineteen may be in, and one may be remote and vice versa. There may be a relatively even split between office and home. This may change throughout the day. People may drop out multiple times from their office or remote location. This is exacerbated in large-scale multiteam environments. Unfortunately, most hybrid methods, processes, and tools tend to be implemented with a particular fixed hybrid profile that is unlikely to be as effective for a certain mix of office/WFH mixes as others. We

Table 1. A new set of recommendations for hybrid software development.

Existing assumptions	New recommendations
1. The careful organization and synchronization of calendars and development cycles are necessary for hybrid development to work.	1. Consider event-based hybrid organization with or instead of calendar-based organization.
2. There is one best way of doing hybrid development.	2. Think about more than one way of doing hybrid development.
3. Hybrid is a fixed and binary concept.	3. Consider the fluidity of hybrid development.
4. Hybrid methods should be implemented as closely as possible to original guidelines.	4. Consider hybrid work as an ongoing experiment.

PRACTITIONER OPINION: HYBRID WORKING IS NOW THE NEW NORM



Patrick Kua, Chief Technology Officer Coach and Technical Leader/Engineering Manager Trainer

William Gibson is well quoted: “The future is already here—it’s just not very evenly distributed.” Remote working has always been here, but COVID and the global lockdown accelerated its spread, and I believe it is here to remain. Prelockdown, we had companies, such as GitLab and Automattic, that were well-known for being remote-first organizations. Even in companies that mandated working from the office, good engineering managers allowed individuals to work from home, recognizing that there were always circumstances when it was needed, all without a negative impact on work outcomes.

I have been working in the software industry for more than 20 years. In one of my earliest jobs, I worked in an open-space office with other developers from my team. Although colocated was our main way of working, it did not exclude variations of remote work. In this particular organization, we all had laptops instead of desktops, as we had the option of WFH in case we needed to receive a package or take care of some personal chore. But remote working was considered an exception. In this particular organization, we had another form of remote working, which involved collaborating with other remote teams. Our team was based in Australia, and we worked with other teams in locations such as the United States and India. For such a complex software system under development, it was impossible to be truly colocated.

Fast-forward many years later, when the world experienced COVID-19 and a global lockdown. We could consider this global lockdown the world’s largest remote working experiment. Where most companies allowed remote working as an exception, or a few companies embraced remote first, circumstances forced everyone to work from home abruptly. This experiment also underscored that teams in a fully remote world didn’t halt productivity. In fact, some research, such as a report by Van Bommel,⁸ demonstrated that remote work options could boost productivity. Despite some managers’ fears that work would not be completed if people were not in an office environment, the evidence was clear: remote work worked.

Unlike some industries and jobs that literally require humans to interact with a physical machine (although that, too, is changing), knowledge-based work, such as producing software, does not inherently demand a physical presence. Combined with fast and accessible Internet connectivity, the reduction in the size and cost of decent video and audio equipment, and more and more software being built remotely in the cloud, people around the world started to see what is possible with more remote-friendly work options.

But many organizations still struggle with what remote work policy to adopt. In my line of work, I train early stage

technical leaders and engineering managers and coach chief technology officers and vice presidents of engineering. A common theme all of them currently face is helping their companies agree to and communicate about an official office policy. Some digital-native companies went full remote first, such as Spotify’s WFX.^{S1} Other companies struggle to arrive at a decision, not wanting to create stronger divides across departments. For example, one company I work with manufactures industrial components. Managers in that company need to answer “Why can the IT department work from home when others can’t?” Software managers must find answers that balance reality, a perception of equity, and the need to collaborate in various remote ways.

In reality, regardless of what top management decides and dictates, my anecdotal observations with the organizations I work with show that most teams have arrived at some form of hybrid working. Even if top management decrees “Everyone must come into the office,” engineering managers still make exceptions for their teams. My sense, after talking to managers, is that some remote work options are now a standard employee benefit and that they fear they will lose good employees to other companies that embrace this. As an example, I see hybrid and remote being added to more and more job advertisements for software engineers. Given the demand for excellent software talent, those who cannot provide this benefit will struggle even more to attract good talent. COVID-19 and the global lockdown tipped the scale, and I think there is no going back.

For software professionals, the question should not be “Should we mandate going back to the office?” or “How many days in the week should be mandated?” I think this has been answered by the actions of managers and demand from employees. Instead, we need to ask ourselves questions, such as

- “How do we build better relationships in a remote/hybrid environment?”
- “How can tools improve remote collaboration?”
- “How do we create an engaging and inclusive work environment, given that some employees will be WFH with varied office space?”
- “How can managers detect when remote individuals need support without micromanaging?”

This is why I’m excited by this issue of *IEEE Software* that explores these questions.

Reference

- S1. “Work isn’t somewhere you go, it’s something you do: We give our people the freedom to work where they work best, wherever that may be.” Spotify. Accessed: Dec. 19, 2022. [Online]. Available: <https://www.lifeatspotify.com/being-here/work-from-anywhere>

suggest that decisions around hybrid methods, processes, and tools consider this wide and constantly changing fluid mix of WFH and office.

Further, we will need tools that support these new methods. In this special issue, John et al.^{A4} describe the use of an employee experience man-

We encourage developers and the general hybrid development community to continually challenge recommendations as they apply and tailor them in the future.

Recommendation 4: Consider Hybrid Work an Ongoing Experiment

Contemporary software development methods, such as agile and flow-based development, were designed for colocated on-site teams or, at the very least, distributed development in a controlled office setting rather than large-scale WFH. It is clear that the extensive use of remote and hybrid work challenges the fundamental assumptions of methods from the pre-COVID era, such as agile, which assumed that a team was colocated or at least distributed across offices specifically equipped with communication technology. Further, before the lockdowns, teams chose processes that were best suited to solve customer problems. Now, the process also needs to support building the team, strengthening the team, and allowing the hybrid setup to change. Therefore, we argue that adaptation of an existing method is necessary. As an example, Jackson et al.,^{A3} in this special issue, suggest eight approaches that hybrid software teams use to preserve, structure, and promote creativity as part of their day-to-day work,

agement (EXM) platform to promote employee engagement within the hybrid work model.

The way of identifying new methods and tools is by experimenting, trying out new processes and tools and then gathering evidence of how they work. As Marcin Floryan, director of engineering at Spotify, concluded in an opinion piece, "Hybrid work should be seen as an ongoing experiment with the potential to shape the future of software development."

Future Directions

While critically surfacing and challenging assumptions are indeed interesting and effective, they are limited in that they identify only a snapshot of assumptions at a given point in time. Therefore, we encourage developers and the general hybrid development community to continually challenge recommendations as they apply and tailor them in the future. Future research therefore needs to continue to explore the area and suggest a research agenda to guide the research community.

Overview of the Special Issue Articles

This special issue covers a range of exciting topics regarding the future of hybrid software development. We applied a rigorous review process to each article, including a review by at least three reviewers from industry and academia. We would like to extend our sincere gratitude to all those reviewers, who were so giving of their time and provided excellent guidance for the authors.

The first article, by Smite et al.,^{A5} provides a typology of different types of hybrid development, which again addresses the false assumption that there is a standard type or way of doing hybrid development. This is based on practical insights from Alphabet, Intersoft, Valtech, IBM, Brandwatch, and Ericsson, and the types of hybrid work include hybrid teams, partially aligned teams, and, more importantly, variegated teams with a fully aligned alternation of office presence.

Next, Sporse et al.^{A1} show that unscheduled meetings are just as important to distributed hybrid software development teams as to those that are colocated. By drawing on the experiences of four development teams in Norwegian companies NAV and Entur, they provide three recommendations for how teams can use virtual rooms to disclose whether it is appropriate to interrupt a colleague for an unscheduled meeting, how Slack channels can be customized so that developers feel safe to ask questions, and how teams can experiment to find new ways and tools for maintaining unscheduled meetings.

In the next article, Wang et al.^{A2} introduce a novel workshop approach to codesigning for a hybrid workplace experience. The article describes their practice of designing a hybrid

PRACTITIONER OPINION: THE PANDEMIC'S SILVER LINING—RETHINKING TEAMWORK IN SOFTWARE DEVELOPMENT



Marcin Floryan, Director of Engineering, Spotify

The global pandemic created a chance to challenge established beliefs about teamwork and colocation. Now, companies must explore the benefits and challenges of hybrid work. This presents an opportunity to reflect on the value and effectiveness of face-to-face interactions and address the pitfalls of remote work, such as a sense of isolation and lack of collaboration. Hybrid work should be seen as an ongoing experiment in how to best organize teamwork while providing flexibility to individuals with the potential to shape the future of software development.

Over the past 20 years, certainly since the publication of the Agile manifesto, software development has steadily become a more collaborative, cross-functional, and colocated endeavor. A “two-pizza team” sitting together in the same physical space, often with a board, visualizing work, has become the norm and a model for the most productive way of building software. Many organizations have gone to great lengths to hire folks around the globe and relocate them to corporate hubs. Against the backdrop of this global trend, very few companies—the likes of Buffer, GitLab, and Automattic—decided to bet on building their businesses in a fully distributed way. In early 2020, this predominant work mode faced a significant challenge as company after company started sending its employees home amid national lockdowns, thereby starting a worldwide experiment in distributed work.

Despite the concerns of developers and managers alike, the work didn't grind to a halt. Surprisingly, there were more developers reporting unchanged and even improved perceived productivity than those who felt negatively impacted—regardless of an anxiety-inducing global pandemic, new parental duties, housing challenges, and economic uncertainty. Initially, we did not really have a choice. Now, the opening up of societies has created an opportunity to decide whether we want to return to the past or explore what will work better. Some companies simply demand that their employees return to offices; others reflect on and evaluate the new reality. It's worth examining the underlying beliefs of those

decisions. Are these beliefs exposing a desire to return to a pretense of control, a wish to avoid the complexity of dealing with distributed work and the value and effectiveness of face-to-face interactions? Regardless of what stands behind them, the positive aspects of this work-from-home experiment are hard to miss, giving us a chance to reflect on past dogma. Many were able to adjust more freely their work schedule, integrating work life and personal life better. Most avoided the cost, in time, money, and environmental impact, of commuting to the office. Organizations that chose to embrace remote work gained access to new pools of employees who would have not otherwise been willing and able to uproot their lives. Families benefited, local communities benefited, the environment benefited and, above all, many of us benefited from the increased sense of agency, the freedom to choose what works best for us.

It's worth recognizing that our ability to remain productive while being more isolated physically was, to some extent, enabled by technology. The past two years have seen a clear boost in the innovation and development of both established and new tools. I, however, would not be looking forward to what the metaverse and virtual reality have to offer for hybrid work but, rather, toward the continued increase in the quality of existing technologies, which will eventually break the threshold of becoming more seamless and natural to use.

WFH, especially during a global pandemic, and a future opportunity for hybrid work are not without pitfalls. We must acknowledge the sense of isolation many felt, the tensions that sometimes flared up at home, and the longer hours some put in to prove they were still making a valuable contribution. Cultivating a sense of belonging, equitable treatment in terms of access to information, and recognition and support of people's mental health are some of the important topics to keep a close eye on.

Consider hybrid work in software engineering an ongoing experiment, where we have an opportunity both to examine the changing reality and shape it. For example, we experimented with building a fully remote gathering called a “homesite” with a mixture of online activities and one-to-one “walk and talk” sessions spread over a couple

(Continued)

PRACTITIONER OPINION: THE PANDEMIC'S SILVER LINING—RETHINKING TEAMWORK IN SOFTWARE DEVELOPMENT (CONT.)

of days. A mix of onscreen and real-world environments made it sustainable by maintaining good energy levels. We also sent everyone the same package with “surprise” content and opened it together at the same time; hearing people’s reactions and appreciation contributed to a sense of belonging. This worked during the lockdown and was repeated for hybrid teams.

Now that offices are open again, we have organized an “anchor week” for part of the organization, a time when everyone is encouraged to come to the office, be present, and work face-to-face with their colleagues. There are also extra activities, such as the joint building of large Lego sets, after-work sessions, and a treasure hunt, that add an element of fun and community.

To leverage the hybrid nature of work, we moved away from synchronous progress updates to short (3–5 min)

videos distributed using Loom, where people can watch and easily comment on the content, creating lively discussions. Many developers have used the growing ecosystem of plug-ins, extending their integrated development environments with screen and keyboard sharing facilities for online pairing sessions.

I want to see “the whole greater than the sum of its parts” being the goal of hybrid software engineering. I encourage us to embrace and take advantage of diverse skills, perspectives, and personalities, bringing them together in distributed yet connected collaborative work. We must continue experimenting with our practices. While it may seem to require more thought, more effort, and more attention, the effects, in terms of our own well-being as well as the outcomes of what we do together, will be well worth it.

workspace at one branch office of SAP Labs. Consisting of three workshops, with a plan for following a continuous improvement process, this framework can help other organizations create their own paradigm of hybrid work via a bottom-up approach.

Jackson et al.^{A3} then argue that creativity can flourish even in widely distributed hybrid settings. However, little research has been performed on creativity in hybrid and even all-remote software teams. By interviewing more than 20 practitioners from the tech industry in the United States, United Kingdom, Canada, and Argentina, the authors identified eight approaches hybrid software teams use to preserve, structure, and promote creativity as part of their day-to-day work. Further, they present an overview of the

suitability of each approach along with example scenarios and a summary of each approach’s strengths and weaknesses.

Finally, John et al.^{A4} investigated how employee EXM platforms may foster employee engagement and increase awareness of work habits in a hybrid work model (see “Practitioner Opinion: Hybrid Working Is Now the New Norm,” and “Practitioner Opinion: The Pandemic’s Silver Lining—Rethinking Teamwork in Software Development”). Based on the qualitative analysis of documents and 26 semistructured interviews in a multinational software company, they found that such platforms may help employees connect with one another and improve individual well-being. The platforms can provide insights into team habits that

can lead to stress and burnout. The downsides of using EXM were also analyzed. 📄

Acknowledgment

The work in this special issue was partially supported by the Research Council of Norway through the 10xTeams project (Grant 309344) and the Transformit project (Grant 321477). It was also supported by Science Foundation Ireland Grant 13/RC/2094_2 and co-funded under the European Regional Developmental Fund through the Southern and Eastern Regional Operational Programme to Lero—the Science Foundation Ireland Research Centre for Software (www.lero.ie). We are very grateful to the authors and to the reviewers who have contributed greatly to this special issue.

References

1. I. Ozkaya, "The future of software engineering work," *IEEE Softw.*, vol. 38, no. 5, pp. 3–6, Sep./Oct. 2021, doi: 10.1109/MS.2021.3089729.
2. D. Smite, A. Tkalic, N. B. Moe, E. Papatheocharous, E. Klotins, and M. P. Buvik, "Changes in perceived productivity of software engineers during COVID-19 pandemic: The voice of evidence," *J. Syst. Softw.*, vol. 186, Apr. 2022, Art. no. 111197, doi: 10.1016/j.jss.2021.111197.
3. "15th Annual State of Agile Report," Digital AI, Boston, MA, USA, 2021. Accessed: Mar. 12, 2022. [Online]. Available: <https://info.digital.ai/rs/981-LQX-968/images/RE-SA-15th-Annual-State-Of-Agile-Report.pdf>
4. K. Stoller, "Never want to go back to the office? Here's where you should work," *Forbes*, Jan. 31, 2021. Accessed: Feb. 18, 2022. [Online]. Available: <https://www.forbes.com/sites/kristinstoller/2021/01/31/never-want-to-go-back-to-the-office-heres-where-you-should-work/>
5. D. Chai and S. Park, "The increased use of virtual teams during the COVID-19 pandemic: Implications for psychological well-being," *Human Resource Develop. Int.*, vol. 25, no. 2, pp. 199–218, 2022, doi: 10.1080/13678868.2022.20470.
6. M. Alvesson and J. Sandberg, "Generating research questions through problematization," *Acad. Manage. Rev.*, vol. 36, no. 2, pp. 247–271, 2011.
7. D. Smite, N. B. Moe, J. Hildrum, J. Gonzalez Huerta, and D. Mendez, "Work-from-home is here to stay: Call for flexibility in post-pandemic work policies," *J. Syst. Softw.*, vol. 195, Jan. 2023, Art. no. 111552, doi: 10.1016/j.jss.2022.111552.
8. T. Van Bommel, "Remote-work options can boost productivity and curb burnout," *Catalyst*, New York, NY, USA. Accessed:

ABOUT THE AUTHORS



KIERAN CONBOY is a professor of business information systems at the School of Business and Economics, University of Galway, H91 TK33 Galway, Ireland, and a co-principal investigator at Lero, the Irish software research center. Contact him at kieran.conboy@universityofgalway.ie.



NILS BREDE MOE is a chief scientist with SINTEF, 7034 Trondheim, Norway. Contact him at nils.b.moe@sintef.no.



VIKTORIA STRAY is an associate professor of software engineering at the University of Oslo, 0373 Oslo, Norway. Contact her at stray@ifi.uio.no.



IAN HENRIK GUNDELSBY is the head of research and development at Knowit, 0561 Oslo, Norway. Contact him at jhg@knowit.no.

Appendix: Related Articles

- A1. T. Sporsem, A. F. Strand, and G. K. Hanssen, "Unscheduled meetings in hybrid work," *IEEE Softw.*, vol. 40, no. 2, pp. 42–49, Mar./Apr. 2023, doi: 10.1109/MS.2022.3229554.
- A2. Z. Wang et al., "Co-designing for a hybrid workplace experience in software development," *IEEE Softw.*, vol. 40, no. 2, pp. 50–59, Mar./Apr. 2023, doi: 10.1109/MS.2022.3229894.
- A3. V. Jackson, R. Prikladnicki, A. van der Hoek, and L. Marshall, "Team creativity in a hybrid software development world," *IEEE Softw.*, vol. 40, no. 2, pp. 60–69, Mar./Apr. 2023, doi: 10.1109/MS.2022.3229353.
- A4. B. John, Z. Alsamarra'i, and N. Panteli, "Enhancing employee experience in the era of hybrid work," *IEEE Softw.*, vol. 40, no. 2, pp. 70–79, Mar./Apr. 2023, doi: 10.1109/MS.2022.3229956.
- A5. D. Smite, E. L. Christensen, P. Tell, and D. Russo, "The future workplace," *IEEE Softw.*, vol. 40, no. 2, pp. 34–41, Mar./Apr. 2023, doi: 10.1109/MS.2022.3230289.