The Best of Both Worlds: Unlocking the Potential of Hybrid Work for Software Engineers

Brian Houck[1], Henry Yelin[2], Jenna Butler[1], Nicole Forsgren[1], Alison McMartin[1]

[1]Microsoft, One Microsoft Way, Redmond, WA

[2]Vista Equity Partners, 401 Congress Avenue, Suite 3100, Austin, TX

Everything we thought we knew about *how* developers work is in a state of change. The Era of Hybrid Work has begun, and it brings new challenges and opportunities along with it. In this paper, we will explore the top challenges that developers are facing in their jobs, the biggest barriers to their productivity and what individuals, teams and organizations can do about it. The SPACE framework posits that developer productivity spans many dimensions[1], and these next pages show that the challenges developers face are similarly multi-faceted.

Since the pandemic, software engineering has changed in many ways, but one of the most notable changes is the ability for people to choose where they work. Companies now understand the importance of offering flexibility in work location to attract and retain top talent. In fact, our research found that developers who are dissatisfied with their ability to choose when and where to work are more than two times as likely to be actively seeking new employment opportunities. However, allowing flexibility in work location comes with tradeoffs and unique challenges, and companies must be equipped to address these to maintain a successful and efficient workforce. In general, there is an opportunity to further accentuate the positive elements of hybrid work, while also addressing some of hybrid work's unique challenges.

Current research shows we have not yet cracked the code on hybrid work. It still holds the promise of the best of both worlds – a vibrant, connected work life with social work relationships and productive impact, and an interwoven life and work balance that allows for loads of laundry between meetings and attending your kid's soccer game – but we haven't realized that promise yet. Instead, many people feel the tension: going to the office when no one is there, unable to separate work from life, feeling "always on" and experiencing "productivity paranoia" (a commonly reported tension[5] between what managers think is happening and what employees are actually doing). More research is needed to understand how hybrid work can empower everyone to both live the life they want and be productive at work. This study aims to identify the unique challenges of hybrid work in software engineering by

analyzing the results of over 3,400 survey responses conducted across 28 companies in seven countries, asking developers not just where they work, but is it really working for them?

Study Design

To identify challenges and opportunities facing developers in the hybrid world, Microsoft partnered with Vista Equity Partners to jointly design and conduct the research. Vista Equity Partners is a leading global asset manager with more than two decades of experience investing exclusively in enterprise software, data, and technology-enabled organizations. Vista pioneered a systematic approach to value creation that strengthens core operating principles to fuel product innovation and accelerate growth in each portfolio company. Dynamic and designed to evolve with shifts in markets and technology, this approach helps founders and executives create better businesses with enduring market value. Twenty-seven software companies from Vista's portfolio opted into this research study alongside several divisions of Microsoft.

To design the survey, eight interviews were conducted across eight companies. Questions such as "What are your company's hybrid work policies" and "How has hybrid work impacted your company's productivity" were asked to a diverse population of individuals from these companies including Chief Technology Officers, managers and individual contributors. Based on the results of these interviews, a survey was created to gain a more comprehensive understanding of the experiences of developers. The survey was carried out over two weeks between November 7th and November 18th, 2022. We received 3,456 total responses from individuals working at 28 different companies. Our target audience was Software Engineers who are individual contributors and not people managers. 2,315 of our survey responses were from this target audience, and except where explicitly noted, the analysis in this article is scoped to those 2,315 responses. The other responses came from people managers and non-developers. The companies included in the survey ranged dramatically in size from companies with a dozen total engineers to large companies with thousands (or in the case of Microsoft, tens of thousands) of engineers. These companies spanned industries and geographies. They encompass a range of corporate policies, workplace cultures, and engineering maturity. This breadth of participants provides a uniquely holistic view of how hybrid work is impacting developer experiences, across companies and workplace cultures, worldwide.

The surveys covered a range of topics about survey respondents' work. We asked every respondent about how productive they've felt over the last six months, about how satisfied they are with their jobs and whether they are looking for jobs at other companies. We also asked questions pertaining to hybrid work practices, and top challenges being faced. The rest of this paper discusses our findings.

The Great Debate: In-Office vs. Remote vs. Blended Work Models

Our research aims to identify which form of work – either fully in office, fully at home, or blended – has the highest level of reported productivity and job satisfaction in developers. Finding the most productive work model can be elusive; this may be because prior research has shown that the optimal way of working is incredibly personal[2]. Some people are more productive working remotely, while others are more productive working in-office[3]. With hybrid work, individuals can take advantage of both remote and in-office modalities to suit the kind of tasks they're doing on a given day. This allows for the best of both worlds and should theoretically lead to increased productivity and job satisfaction[4]. Figure 1 shows the breakdown of work locations for the population in our study:
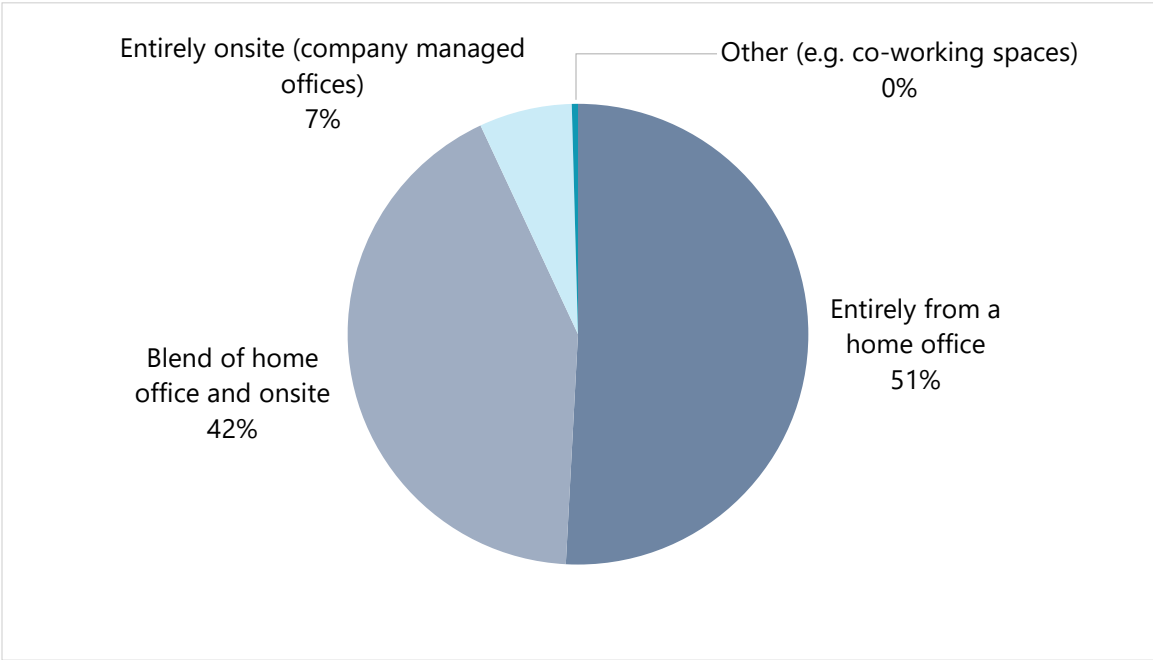

Figure 1: Response to question "Please describe where you physically do your work"

While hybrid work policies are popular, the majority of study respondents work entirely from a home office. This may be because the company operates as fully remote, or developers choose not to go into the office at all when working in a hybrid environment. While this shows us that a fully remote work is now the dominant model amongst the software engineers we studied, the question remains: Is remote work the "most productive"? To help answer this question, we analyzed responses to the question: "I have experienced high levels of productivity in the last six months," which was answered on a scale of 1 = strongly disagree to 5 = strongly agree. Average responses were compared to physical work locations reported by developers.

Microsoft   V I S T A

Work Model	Average Productivity Rating (Out of 5 Point Scale)
Entirely remote	4.18
Blend of home and in-office	3.95
Entirely in-office	3.87

Table 1: Average productivity rating by work model (self-reported).

Our analysis finds that those working entirely remotely report the highest level of productivity, followed by those working from a mix of home and office, and finally those working fully onsite report the lowest level of productivity (as shown in Table 1). The difference in productivity rating between those individuals working entirely remotely and those blending home/office are statistically significant, as are the difference between fully remote and entirely in-office ($p <$ .001). There are many factors that we thought might explain this:

- Workers who are at different stages of their career may be more likely to work in one model versus another.
- Companies focused on certain types of work may be more likely to support fully remote work, where the majority of work tasks are supported by technology and do not require physical interaction with customers or physical goods.
- Companies with long traditions of remote work may see more success with this work style.

However, employees' relative ranking of productivity holds even after we controlled the data for these factors: Fully remote developers feel more productive than blended developers and feel more productive than fully in-office developers, regardless of career level, industry, location, or any other factor we investigated.

It may be tempting to look at the figure above and jump to the conclusion that remote work is "better". It is more likely an indication of two critical concepts: 1) that remote work does not equate to lower productivity, and 2) that hybrid work is more challenging to optimize. The evidence continues to mount that remote developers can be highly productive, and companies shouldn't fear supporting remote developers. Hopefully we will start seeing fewer companies trapped by what Microsoft has dubbed: "productivity paranoia" – a phenomenon Microsoft identified in previous research in which a large percent of employees report working productively, but only a small percent of leaders report having full confidence that their team is productive[5]. What we believe we're seeing in this study's data is an acknowledgment that hybrid is hard, and we haven't figured out all the challenges that come with hybrid work.

Microsoft   VISTA

The Struggle is Real: Top Work Challenges Faced by Developers in Hybrid Work

The COVID-19 pandemic forced many technology companies to adopt remote work policies. While this has allowed developers to work from the comfort of their own homes, it also brought about a new set of challenges[6,7]. Some of the challenges engineers reported in past studies of remote and hybrid work included missing social interactions, difficulty communicating with colleagues, and insufficient hardware[2,8]. As companies begin to transition to hybrid work models, we wanted to explore how the top challenges have changed as developers were able to start blending home and in-office work.

To uncover the key challenges facing developers at work, we asked survey respondents, "What have been the biggest work challenges that you have experienced over the last 6 months" and had them select one to three challenges from a pre-determined list compiled from our interviews and from top reported challenges from previous studies[2]. We chose a six-month reflection period because many of the developers in the study use 6-month development cycles, and we did not want to limit the barriers we heard to the activities happening during a subset of the cycle. The below figure shows the challenges listed in order of frequency based on survey responses:
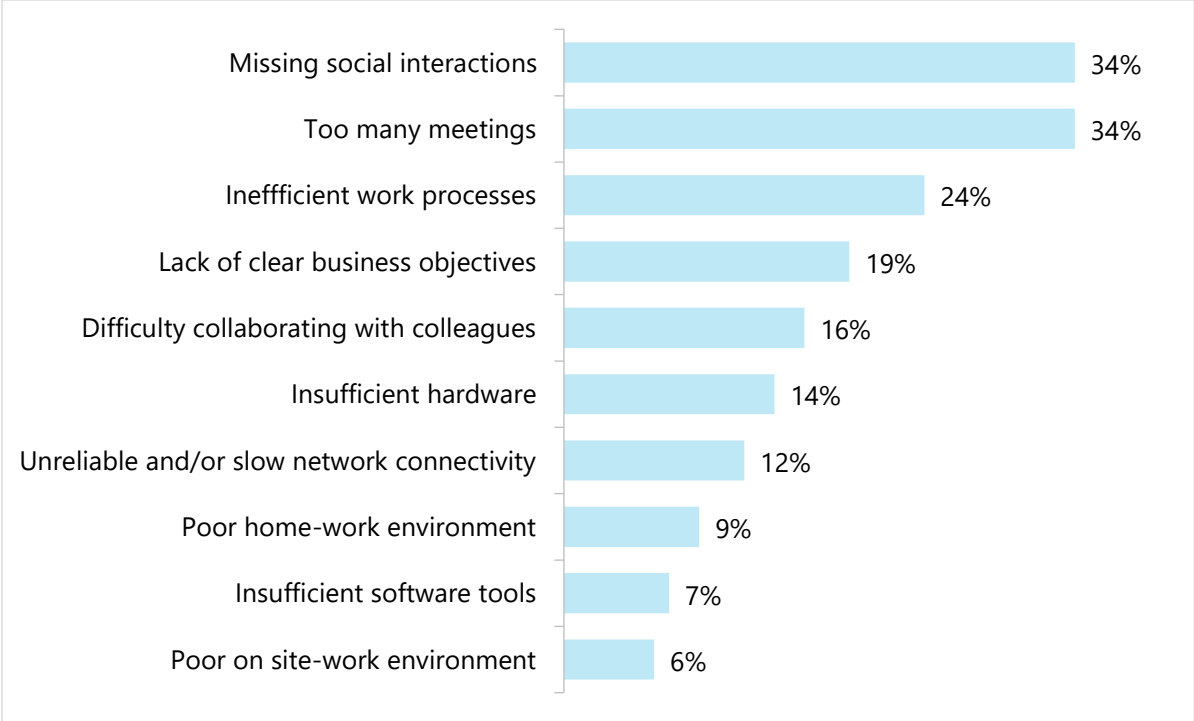

Challenge	Percentage
Missing social interactions	34%
Too many meetings	34%
Ineffficient work processes	24%
Lack of clear business objectives	19%
Difficulty collaborating with colleagues	16%
Insufficient hardware	14%
Unreliable and/or slow network connectivity	12%
Poor home-work environment	9%
Insufficient software tools	7%
Poor on site-work environment	6%

Figure 2: Biggest work challenges reported over last 6 months

Microsoft　VISTA

You will notice the above sums to greater than 100% as most developers selected more than one challenge.

Missing social interactions

Surprisingly, we see that the most frequently cited challenge is "missing social interactions," even though many of the developers we surveyed now have the option to come into the office to see their peers face-to-face in addition to working from home. This challenge is not limited to remote workers. In fact, fully onsite developers are 8% more likely to miss social interactions than fully remote developers. It is unclear exactly why this is, but we have the following hypotheses: 1) people come into the office for meetings, but meetings do not fill a need for social interaction since they are inherently transactional. 2) people come to the office because of other reasons (an appointment or errand near the building, for example) but people they want to meet aren't there at that time for a variety of reasons. And 3) related to number 2, companies do not yet have schedules for in-office days, making it difficult to serendipitously see people.

It should be noted that individuals who report missing social interactions are just as likely to be feeling productive as those who do not report missing social interactions. Our analysis also finds that individuals who report missing social interactions are no more likely to be looking for jobs at other companies than those who do not report missing social interactions. This suggests that social interaction is an important human need but may not have business impacts – yet.

To help alleviate this challenge, teams should organize social rituals, which provide both moments to socialize and a sense of ongoing team culture[3,9].

> **"**
>
> **[In office, I appreciate the] interactions with other people. The conversations that get sparked. The camaraderie.**
>   -S1841

Too many meetings

During the shift to remote work, developers' calendars exploded as most interactions became meetings. One in three developers report having too many meetings. What used to be a five-minute hallway conversation turned into a 30-minute meeting, and those meetings stacked up back-to-back-to-back. Not only did we have more meetings, but those meetings were more exhausting than discussing the same topic in-person[10]. As many individuals have shifted to a hybrid way of working, the challenge with meetings has not been alleviated because even if some team members are in the office, others are at home and the default interaction remains

Microsoft VISTA

the 30-minute meeting. Too many meetings can lead to burnout[11] (particularly if they stretch your day longer) and come at the expense of more fulfilling work[3]. For teams and organizations to be successful in a hybrid world, they'll need to figure out how to decrease meeting load by shifting some meetings to asynchronous communication and reducing redundant or low-quality meetings.

> **"**
>
> **Too many meetings impacts the ability to have focused work for longer duration and also causes inability to plan a day or tasks in advance.**
>   -S1948

Inefficient work processes

The third most frequently cited challenge may be the costliest to developer productivity and satisfaction. Individuals who reported inefficient work processes as a top challenge were more than 2x as likely to say they were feeling unproductive, compared to the average developer. They were also 68.6% more likely to say they were dissatisfied with their jobs. Additionally, they were 66.8% more likely to say they were looking for jobs at other companies. No challenge in our study appears to be a bigger risk to retention than inefficient work processes. And by definition, inefficient work processes are inefficient.

One might think only large companies face this challenge, since greater size could lead to red tape and bureaucratic overhead. However, inefficient work processes were still the 3rd most frequently cited challenge even for developers in the smallest companies included in the study.

Developers want to avoid toil and bureaucratic processes that distract from delivering innovation, and they'll leave their jobs to avoid these processes. The impact of inefficient work processes suggests that engineering teams should constantly evaluate their work processes and find ways to reduce toil and friction within them. Not all toil can be avoided; many processes are needed to ensure compliance requirements are met or to ensure the quality bar is maintained for customers. Process comes with a cost, and teams should deliberately decide when they are willing to pay it.

Lack of clear business objectives

Alignment on business objectives can be a challenge for developers. According to our survey, it was the 4th most frequently cited challenge, with nearly 1 in 5 respondents saying it was a top challenge. This challenge correlated to lower self-reported productivity and job satisfaction. In addition, respondents who felt business objectives were unclear were 42% more likely to be looking for another job.

Microsoft　VISTA

We conducted additional analyses to see how this challenge impacted developers based on their time working at their organization. Relatively fewer new hires felt like a lack of clear business objectives was a problem. This isn't necessarily a surprise, since junior developers are often focused on onboarding and ramping up knowledge in their new environment, such as understanding their new codebase and toolchains. In this scenario, clear business objectives are important to doing their job well, but not the most important. However, 13% of new hires felt like they weren't getting the clarity they desired around business objectives. The impact on their productivity from facing this challenge was nearly 3x greater than the impact to the average developer.

To help alleviate this challenge, companies should work to ensure that their business objectives are clear and well-communicated. This can allow developers to understand how their work fits into the bigger picture, helping keep them motivated and engaged.

Difficulty collaborating with colleagues

If missing social interactions fills a human need, collaborating with colleagues clearly fills a business need. Developers who reported a challenge collaborating with colleagues were 1.75x more likely to say they were feeling unproductive and 13% more likely to say they were looking for another job.

To help alleviate this challenge, companies should work to ensure that their developers have the tools and resources they need to collaborate effectively. Additionally, companies should encourage collaboration by creating a culture of openness and transparency, where developers feel comfortable sharing their ideas and working together to solve problems. Using a communications contract, such as the one found here, can help teams improve their collaboration and communication.

Key takeaways from the top challenges

When we analyzed the top challenges, three key points came to mind:

1) These challenges are not limited to certain companies or certain career stages. The order of the list was remarkably similar when we scoped to small companies and to new hires. In fact, these challenges are quite similar for non-developers (we analyzed responses we received from non-developers and found similar patterns). This suggests that these challenges aren't unique to single organizational culture or engineering tech stack. They aren't unique to a certain mode of working (e.g. remote versus in-office). These aren't even "developer" challenges. These hybrid work challenges are human challenges.

2) Our changing work modalities have had a substantial impact on *how* we collaborate. Remote and hybrid work impacts the network of people we interact with[13], and they also impact the outcomes of those interactions. If what might have been an informal hallway conversation now moves to email, the chances for misunderstandings (which could result

Microsoft   VISTA

in more meetings to add clarity) go up, and the odds of receiving help go down[14]. This requires developers to be more careful with their email communication than they would've needed to be in-person. As we learn to navigate these new modalities, we see a relationship between many of the top challenges related to how we interact.

3) Unfortunately, many of the challenges we identified are in tension with each other, creating what we have deemed the "Meeting Paradox.". Developers reported they want more social interactions to fill a human need and also that they require collaboration with colleagues to fill a business need; conversely, they do not want more meetings. Meetings can provide accelerated / deepened / impactful collaboration as well as create a feeling of social connectedness (especially in-person meetings) but will, by definition, increase the already high meeting load. This "Meeting Paradox" warrants further research, but for now, here are some best practices that could help:

   a. Utilizing asynchronous collaboration methods: try collaborating virtually over a shared document or slide deck. Not everything that requires multiple people working together needs to be a meeting. Sending out a link to a document, asking for specific feedback or help, leaving comments, and tracking changes are all ways we can collaborate without *being* together.

   b. Following hybrid meeting best practices: meetings are less taxing when they are well run, and this includes things like having an agenda, making sure everyone is included, and, most importantly, only having a meeting when you actually need a meeting (see the above point!). For research-backed tips and best practices, check out this [Hybrid Meeting Guide.](#)

   c. Planning your time with your team: Be planful with your time and your team: if you are utilizing asynchronous ways of collaborating and making sure meetings are reduced to the group and cadence necessary, there might be extra time for social activities that can re-build social capital without burning people out. Consider starting every meeting five minutes after the hour. People who want to catch up socially can join the meeting early and chit chat before the official agenda begins (remember doing that in real life in a conference room?). Also, research has shown that teams that play games together are more productive[12]. Even in the hybrid world, tools like [Games for Work](#) allow everyone to play, potentially boosting both productivity and social connectedness.

As this research has shown, developers continue to face a diverse set of challenges, from missing social interactions to challenging home-work environments. Addressing these challenges will be essential for maximizing all modes of hybrid work.

With many companies transitioning to hybrid work, we'll explore the role the office will play in providing a flexible and collaborative work environment.

Microsoft VISTA

The Office: A Place for Collaboration or a Relic of the Past?

What value do developers find in the physical office? We asked developers who came into the office at least occasionally: "What attributes of the office do you most value?". After reading through all the open text comments, we were able to categorize the results and analyze the top themes expressed by developers.

Attribute	Frequency Mentioned	Representative Quote
The People	61%	**Seeing colleagues face-to-face** *boosts my morale since it's often fun to catch up with colleagues and hear about their lives...* (S214)
Office Design	22%	*Having **a conference room** where the team can meet to hash out ideas and problems.* (S298)
Food/Coffee	19%	**Free coffee and drinks**. *I might come in more often if the office had a Taco Tuesday or Pizza Wednesday. It would be nice if we had free snacks.* (S533)
Workstation / Equipment	13%	*...having **two monitors** and a **comfortable chair**.* (S1368)
Dedicated Space	7%	*I like the fact I have **a place that's where I work** and a place where I unwind.* (S688)
Climate Control	5%	*In the summertime, **air conditioning**...* (S448)
Network Speed	5%	*Stable and fast **internet connectivity.*** (S2459)
Other	10%	*I don't have to fight with my cats.* (S6)

Table 2: Results from " What attributes of the office do you most value?" based on thematic analysis with representative quotes for each topic.

> **In office, I value the high-speed internet, chance to collaborate with other employees, the free coffee, and office amenities.**
>   -S570

Many developers think of the office as a place to connect and collaborate with their peers, implying that they're looking for the in-office experience to solve some of the top challenges we discussed in the previous section. While developers may be enjoying the flexibility of remote work, it seems that we have not been able to find digital replacements for the benefits of working together in the physical world.

Microsoft   VISTA

Many different aspects of building design were mentioned in the 22% of responses that touched on design. Developers value whiteboards for brainstorming in person (though it should be noted that developers who mentioned whiteboards were 2.5x more likely to be 7+ years into their career, so there could be a generational divide on that aspect). One company offered gym and sauna amenities in their office, and of the nine responses that mentioned this as the attribute of the office they most valued, none of those individuals said they were looking for other jobs. For those employees where gym and sauna are an ingrained part of the culture, an office designed to meet those local cultural values may provide a big retention boost.

Nearly one-in-five developers mentioned free food and coffee as the attribute they most valued. It should be noted however that those individuals were no more likely to report feeling productive, nor where they any less likely to be looking for another job. This implies that while people like free food and it can contribute to morale, it doesn't necessarily correlate to reported productivity.

To come or not to come to the office – that is the question

Since most developers in our study wanted the office to fill a human need for social interaction and collaboration, we wanted to understand what caused someone to decide whether to work at home or in the office on a given day. Were they explicitly coming into the office on days when they had a lot of meetings scheduled (to interact with their peers in-person), or was the decision primarily based on how much focused work they had to do that day?
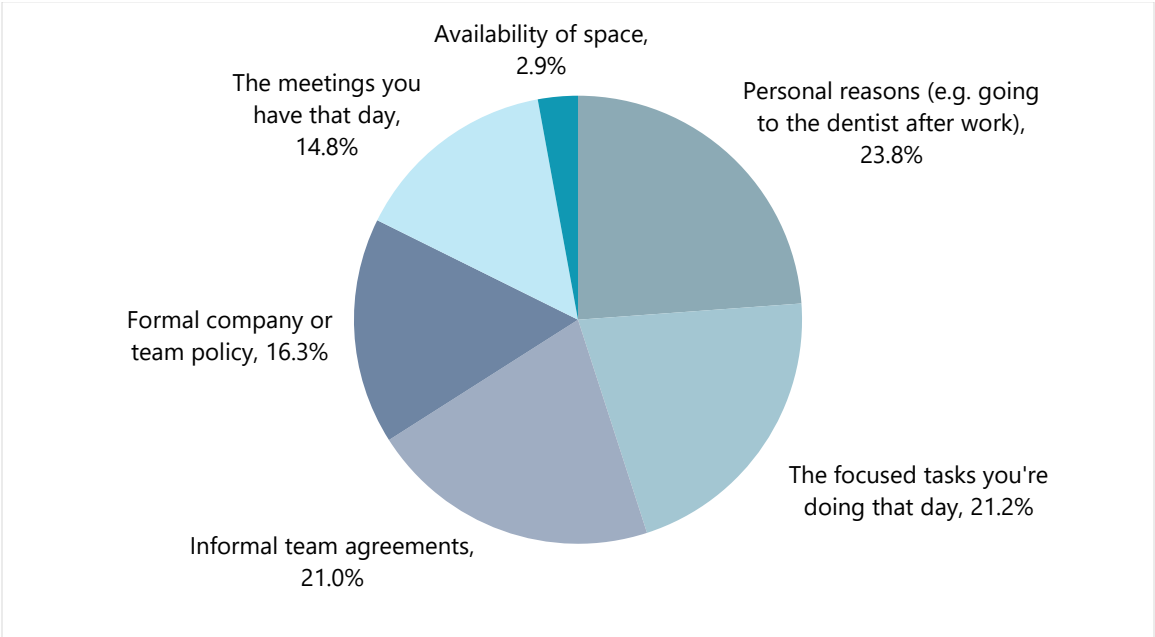

Figure 3: Responses to "What causes you to decide which days to work in the office versus at home?"

Interestingly, we see that the most frequently cited reason to choose where to work has nothing to do with the work a developer is doing on a given day but rather personal reasons. If developers are choosing where to work based on what is happening in their lives outside of work, then it might explain why they are still missing social interactions even when working from the office – when choosing where to work for personal reasons, it's essentially random if you'll overlap with other colleagues.

One of the reasons why hybrid is theoretically an optimal way of working, is that not all spaces need to be equally tailored to every kind of task. For many developers, their at-home work environment has been specifically tailored to their individual preferences and work style. This makes it a superior place to perform focused work. In contrast, the office may be tailored for collaboration and creative brainstorming activities. Our study provides evidence to support this hypothesis. Developers who chose where to work based on the focused tasks they were doing that day were 12% more likely to feel productive than developers who chose where to work based on personal reasons. Similarly, developers who chose where to work based on the meetings they had were 11% more likely to feel productive. Since not all spaces are created equal, choosing where to work based on non-work reasons can result in sub-optimal productivity. If you don't optimize, you get non-optimal outcomes.

While remote work has its benefits, for many developers the office still has a role to play in the hybrid work model. In the next section we'll explore how developer tools and processes will need to improve to address the needs of hybrid developers. We'll also look at what it will take for remote development environments to solve some of the uniquely hybrid problems.

The Future of Development: Overcoming the Biggest Barriers to Productivity in a Hybrid World with Better Developer Tools and Processes

Developers are facing barriers to productivity due to their tools and processes, and many of these barriers may be exacerbated by hybrid work. Insufficient software tools were a challenge for new hire developers in particular. New hire developers are 40% more likely to say that "Insufficient software tools" was a top workplace challenge, and the impact to their productivity is 37% greater than it is for more seasoned developers. While tooling can be a challenge for senior developers also, more senior developers may have developed workarounds or shortcuts for the tooling they use. Improvements in tooling would likely improve productivity for all developers.

The table below shows the results of a thematic analysis of the following question: "When thinking about your development tools and processes, what is the biggest barrier to your productivity?"

Barrier	Frequency Mentioned	Representative Quote
Developer Tools	14%	**_Usability of the tools_**_. Most internal tools are not user friendly. The UI is very limited and rough. Compare with Visual Studio or Visual Studio Code; both have strong UI investments and are user friendly to a diverse customer base._ (S408)
Meetings/Collaboration	13%	**_Too many meetings_** _are a barrier causing context switching to be slow, when that discussion could have definitely been an email._ (S412)
Compliance	11%	**_Compliance_** _related processes._ (S81)
Developer Education (documentation/learning/ support)	11%	_The biggest barrier to my productivity is_ **_my own skillset_**_. Still being relatively new to software development, there are gaps in my knowledge that can sometimes slow down or impede my work._ (S2269)
Connectivity	10%	_When working remotely,_ **_latency and disconnects_** _when remote connecting to my_ [development environment] (S738)
Hardware	8%	**_Performance of my work machine_**_. CPU and Memory constraints drastically increase compile build performance._ (S847)
Local Build	7%	_The lag time in_ **_building_** _a small, local change in code._ (S130)
Deployment	4%	**_Deployments are blocked too often_**_, causing delays getting new functionality to customers._ (S749)

Table 3: Results from "When thinking about your development tools and processes, what is the biggest barrier to your productivity?" based on thematic analysis with representative quotes for each topic.

There is no silver bullet. The barriers developers identified cover many dimensions. All the dimensions of the SPACE framework[1] are covered by the responses we saw to this question. Reducing friction in developer tools, the build process, and deployment are important. So too is improving collaboration (without simply adding more meetings).

Microsoft   VISTA

🏢 Policy leading to satisfaction & wellbeing

Developers who are satisfied with their ability to choose when and where to work are nearly half as likely to be looking for another role, and report feeling more productive.

💬 Communication & Collaboration	⚙️ Tooling – Creating efficiency & flow
61% of developers say that <u>seeing colleagues face to face</u> is the most valued part of going into the office.	Developers who experience <u>difficulties collaborating</u> are **13.1% more likely** to be looking for other roles.
	Developers who experience <u>inefficient work processes</u> are **66.8% more likely** to be looking for other roles.
41% of developers say '<u>missing social connections</u>' have been their biggest challenge in the past 6 months.	New hire developers are **40% more likely** to say '<u>insufficient software tools</u>' is a top workplace challenge and the impact it has on their productivity is **37% greater** than it is for more seasoned developers.

Figure 4: Summary of research results

This challenge of moving fluidly between locations requires more portability in developer computing power. The days of a developer being able to put a beefy desktop machine under their desk and use that as their primary environment are gone with the introduction of hybrid work. Cloud-based remote development environments like Microsoft's Dev Box, Google's Cloud Workstations, or Amazon's WorkSpaces are potential solutions to this need for portability while also increasing productivity through the scalable power offered by cloud platforms. From the perspective of individual developers, we wanted to know what these remote environment solutions needed to offer for them to choose to adopt them, so we asked: "What would a remote development environment need to offer for you to choose to use it?" The table below shows how developers answered this question. We only asked developers who are not currently using any cloud-based solution as their primary development environment.

Feature	Frequency Mentioned	Representative Quote
Performance	46%	***Faster*** *builds than I can do on my dev box* (S35)
Ease of Use	28%	*Be able to have all of the daily tools/libraries I use daily,* ***already installed*** *on the remote dev environment* (S353)
Latency/Responsiveness	23%	*Source code editor needs to* ***respond*** *to my keyboard and mouse as quickly as a desktop PC does* (S405)
Reliability	19%	*A better guarantee of* ***no network or system issues****. We have had too many of our remote employees using virtual machines unable to work for days at a time due to things out of their control* (S2565)
Configurability/Control	10%	*Faster, more reliable than my current set up, and* ***just as configurable*** *in terms of software I can install/use. Would take a lot to make me want to choose a remote env* (S665)
Nothing	9%	***I would never voluntarily choose*** *a remote development environment. I've had extremely negative experiences using remote development environments* (S2491)
Other	10%	*Multimonitor support would be the most important, ability to connect Surface headphones via a USB link as well* (S80)

Table 4: Themes seen in responses to the question "What would a remote development environment need to offer for you to choose to use it?" and their frequency, along with a representative quote for each major theme.

Developers were unequivocal that they won't adopt remote development environments for the sake of it. Those environments need to provide value above and beyond what they're using today. They don't just want responsive and available environments to author, test, debug and deploy their code, they want them to be configurable and easy to set up and use.

Hybrid work has introduced several challenges for developers, but by improving developer tooling and processes, organizations can help developers overcome these challenges and thrive in a hybrid work environment. Remote development environments are one potential solution to the challenge of moving fluidly between locations and can offer increased productivity through the scalable power offered by cloud platforms. By understanding what developers are looking for in a remote development environment, organizations can create environments that meet these needs and help developers work more effectively.

Microsoft   V I S T A

We studied:

3,456 individuals across

28 companies of various sizes around the globe

We surveyed them about the biggest challenges and opportunities developers are facing in a hybrid world, as well as their job satisfaction and productivity over the last 6 months.

Top hybrid challenges	Most valued attributes of the office
Missing social interactions **34%**	The people **61%**
Too many meetings **34%**	Office Design / Amenities **22%**
Inefficient work processes **24%**	Food/Coffee **19%**

Figure 5: Summary of the research process and results

Conclusion

This joint research between Microsoft and Vista Equity Partners sought to answer the question of what is hindering productivity in developers and how companies can address these challenges in a hybrid world. We studied 28 companies of varying sizes and work location modalities to understand the challenges developers face on a day-to-day basis. We learned that many of the challenges faced by developers are not developer specific, but rather human challenges that exist in the hybrid world. One of the top challenges was missing social interactions and struggling to collaborate with colleagues. In fact, those experiencing difficulty with collaboration were 1.75x more likely to feel unproductive and 13% more likely to be looking for another job. Compounding this challenge with a lack of collaboration and connection is the desire for fewer meetings. This meeting paradox can be solved if meetings are carefully scheduled – only used when needed, not used when asynchronous methods would do, and perhaps include dedicated social and collaboration time. We also found that the usual suspects of slow build speed, low compute power and latency can impact developer productivity. It is possible that remote development environments can help with some of the challenges of hybrid

Microsoft   VISTA

work (no matter where a developer is, their workstation remains the same) however developers are not likely to move to this type of environment unless it is better than their current set up.

In conclusion, the challenges developers are facing in our hybrid world are not unique to them – they are issues every company and manager needs to address in order to realize the promise of best-of-both-worlds work. By carefully planning in-office days (if they exist), choosing work location based on type of work, and being deliberate about connecting with others, managers and leaders can help address the challenges faced by information workers in our new world and unlock more productivity for workers in any location.

Acknowledgments

References

1. Forsgren, N., Storey, M.-A., Maddila, C., Zimmermann, T., Houck, B., Butler, J. 2021. The SPACE of Developer Productivity: There's more to it than you think. *ACM Queue;* https://dl.acm.org/doi/10.1145/3454122.3454124
2. Ford, D., Storey, M.-A., Zimmermann, T., Bird, C., Jaffe, S., Maddila, C., Butler, J., Houck, B., Nagappan, N. 2021. A Tale of Two Cities: Software Developers Working from Home during the COVID-19 Pandemic. *ACM Transactions on Software Engineering and Methodology;* https://dl.acm.org/doi/full/10.1145/3487567
3. Teevan, J., Baym, N., Butler, J., Hecht, B., Jaffe, S., Nowak, K., Sellen, A., and Yang, L. (Eds.). 2022. Microsoft New Future of Work Report 2022. *Microsoft Research Tech Report;* https://aka.ms/nfw2022
4. Bloom, N. 2021. Hybrid Work is Here to Stay. Now What? (Back to Work, Better). HBR IdeaCast; https://hbr.org/podcast/2021/06/hybrid-work-is-here-to-stay-now-what
5. Microsoft. 2022. Hybrid Work is Just Work: Are We Doing it Wrong?. *Microsoft WorkLab;* *https://www.microsoft.com/en-us/worklab/work-trend-index/hybrid-work-is-just-work*
6. Butler, J., Jaffe, S. 2021. Challenges and Gratitude: A Diary Study of Software Engineers Working From Home During Covid-19 Pandemic. *ICSE-SEIP;* https://www.microsoft.com/en-us/research/publication/challenges-and-gratitude-a-diary-study-of-software-engineers-working-from-home-during-covid-19-pandemic/

Microsoft ⋁ VISTA

7. Butler, J., Yeh, C. 2022. Walk a Mile in Their Shoes: The Covid pandemic through the lens of four tech workers. *ACM Queue*; https://dl.acm.org/doi/10.1145/3534860

8. Nolan, A. *et al.* 2021. To Work from Home (WFH) or Not to Work from Home? Lessons Learned by Software Engineers During the COVID-19 Pandemic. In: Yilmaz, M., Clarke, P., Messnarz, R., Reiner, M. (eds) *Systems, Software and Services Process Improvement. EuroSPI 2021. Communications in Computer and Information Science, vol 1442. Springer, Cham*; https://doi.org/10.1007/978-3-030-85521-5_2

9. Methot, J., Rosado-Solomon, E., Downes, P., Gabriel, A. 2021. Office Chitchat as a Social Ritual: The Uplifting Yet Distracting Effects of Daily Small Talk at Work. *Academy of Management Journal*; https://doi.org/10.5465/amj.2018.1474

10. Shoshan, H., Wehrt, W. 2021. Understanding "Zoom fatigue": A mixed-methods approach. *International Association of Applied Psychology*; https://doi.org/10.1111/apps.12360

11. Morshed, M., Hernandez, J., McDuff, D., Suh, J., Howe, E., Rowan, K., Abdin, M., Ramos, G., Tran, T., Czerwinski, M. 2022. Advancing the Understanding and Measurement of Workplace Stress in Remote Information Workers from Passive Sensors and Behavioral Data. *10th International Conference on Affective Computing and Intelligent Interaction (ACII)*; doi: 10.1109/ACII55700.2022.9953824.

12. Keith, M., Andersen, G., Gaskin, J., Dean, D. 2018. Team Video Gaming for Team Building: Effects on Team Performance. *AIS Transactions on Human-Computer Interaction*; https://doi.org/DOI: 10.17705/1thci.00110

13. Yang, L., Holtz, D., Jaffe, S., Suri, S., Sinha, S., Weston, J., Joyce, C., Shah, N., Sherman, K., Hecht, B., Teevan, J. 2021. The effects of remote work on collaboration among information workers. *Nature Human Behavior*; https://doi.org/10.1038/s41562-021-01196-4

14. Roghanizad, M., Bohns, V. 2022. Should I Ask Over Zoom, Phone, Email, or In-Person? Communication Channel and Predicted Versus Actual Compliance. *Social Psychological and Personality Science*; https://doi.org/10.1177/19485506211063259