

Case Study:

# How Uplevel Reduces Short Fragments and Increases Deep Work for Better Focus

---

Uplevel helps Zipwhip find the true blocker to Deep Work at the company – Short Fragments – and helps implement a new meeting culture to improve Deep Work for every team in the company.



## Deep Work

---

# A key engineering metric

Zipwhip tracks a number of key engineering metrics to make sure they are shipping products on time with frequent check-ins with their Senior Leadership Team. These engineering metrics aren't what you might expect them to be. Alongside typical metrics like Uptime, Customer Support Escalations, and Cycle Time, they also track important People Health metrics to monitor how work is getting done. One key metric is Deep Work, a measure of uninterrupted working time for developers to reach true, deep focus.

Uplevel believes that you can't meet your product release goals without focusing on both product and people. Zipwhip partnered with Uplevel in the fall of 2020, specifically interested in honing in on people metrics and improving them across the company.

Zipwhip now uses Uplevel to monitor and improve Deep Work patterns while engaging across the organization. At the executive level, Uplevel shares reports of how Deep Work is trending by team, with an expanded timeline that shows month-to-month comparisons. Every manager and developer has access to data that shows how their team is tracking, as well as their personal Deep Work metrics. Beyond measuring Deep Work time, Uplevel also worked with the leadership team to implement cultural changes that would further improve Deep Work time. The result? The creation of a company-wide Focus Time block on the calendar, when developers are free to dive deep into projects and are not expected to attend meetings or respond to Slack messages.

---

## Deep Work

*a measure of uninterrupted working time (2+ hours)  
for developers to reach true, deep focus*



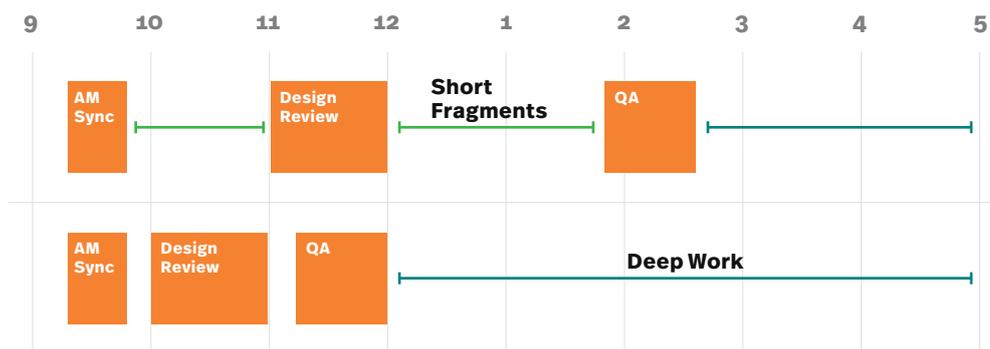
## Short Fragments

# A concerning trend

Despite initiatives to protect Deep Work, monthly reports were showing that developers had less Deep Work than in the previous periods. Zipwhip was growing rapidly and many new developers had an increase in meetings—but this wasn't the only problem.

Deep Work is defined as uninterrupted blocks of two or more hours without meetings or incoming Slack messages. Its counterpart is Short Fragments. Think of that 30-minute break you have between meetings—you might catch up on Slack messages, but don't have enough time to accomplish a task that requires intense focus, like reviewing a colleague's PR or fixing that sneaky bug that you just can't quite find.

Uplevel found that the true upward trend at Zipwhip was a rise in Short Fragments, meaning that developers had more windows of two hours or less, which break working time into small windows of surface-level focus. This was caused in part by the increase in meetings, which cut into developers' Deep Work time. Developers might only have two hours of meetings, but their scattered times throughout the day split up available Deep Work time, instead increasing the rate of Short Fragments on the calendar. The time they had to work was brief, preventing true focus.



## Short Fragments

*a measure of brief working windows (<2 hours)  
that lack sufficient time to reach true, deep focus*



# A new approach

Leveraging these Uplevel insights, Zipwhip changed their strategy and instead focused efforts on reducing Short Fragments on the calendar. This meant rearranging some meetings to preserve those precious open blocks on the calendars. Zipwhip found success with the following tactics:

### **Dedicate a single Meeting Day.** (Zipwhip chose Thursday.)

Consolidate most recurring meetings to a single day of the week to prevent fragmenting on other days. Try to schedule ad hoc meetings on this day whenever scheduling permits. This affords more consistent Focus Time for the whole team on other days of the week.

### **When scheduling, extend meeting time that is already blocked.**

When possible, schedule new meetings as extensions of other meeting blocks. For example, find the half-hour after a weekly team check-in, rather than picking the middle of an open afternoon. This keeps people in “meeting mode” and preserves the other hours of the day for more focused work.

### **Send team representatives to divide the meeting load.**

For meetings that involve multiple teams, send a representative from the team to gather information and report back. This preserves Deep Work time for other team members that can simply read notes later. Be sure to rotate representatives so that this is equitable for the team.





### **Communicate freely, attend wisely.**

You can invite broadly to meetings so that everyone has the opportunity to participate, but encourage your team members to decline meetings that are not an effective use of their time. If it interrupts Deep Work and doesn't help productivity on a project, it might be okay to skip. Help others make decisions by marking the "optional" flag for those who are not required to attend a meeting.

### **Schedule design discussions.**

Set aside one hour per day as "office hours" where all developers are present. Use this time for developers to ask questions or get pair programming help on challenging work. This enables developers to get answers for their issues at a regular time and minimizes the need for random meetings.

### **Save Slack and email for Short Fragment time.**

While Slack messages and emails feel imminent, encourage your team to prioritize more complex work for Focus Time. Then, use fragmented time to catch up on smaller tasks, like responding to those messages.



## More than a feeling



**In a one-month period, every group at Zipwhip showed a noticeable increase in Deep Work**



**One group increased its Deep Work by 35%**



**Deep Work became the biggest component of developers' days, replacing the earlier predominance of Meetings or Short Fragments**



“The feedback I am getting from my team is that they indeed have more time to do high-quality work. The large blocks of Focus Time are important for my team to be able to do thorough design work upfront, thinking deeply about corner case scenarios rather than just the happy path scenarios. **This results in the delivery of high-quality features the first time around.** Consistent Focus Time also allows us to plan sprints more accurately and deliver results within a predictable timeline, because the developers actually have dedicated time to work on their assigned tasks, rather than being pulled around randomly in multiple directions.”

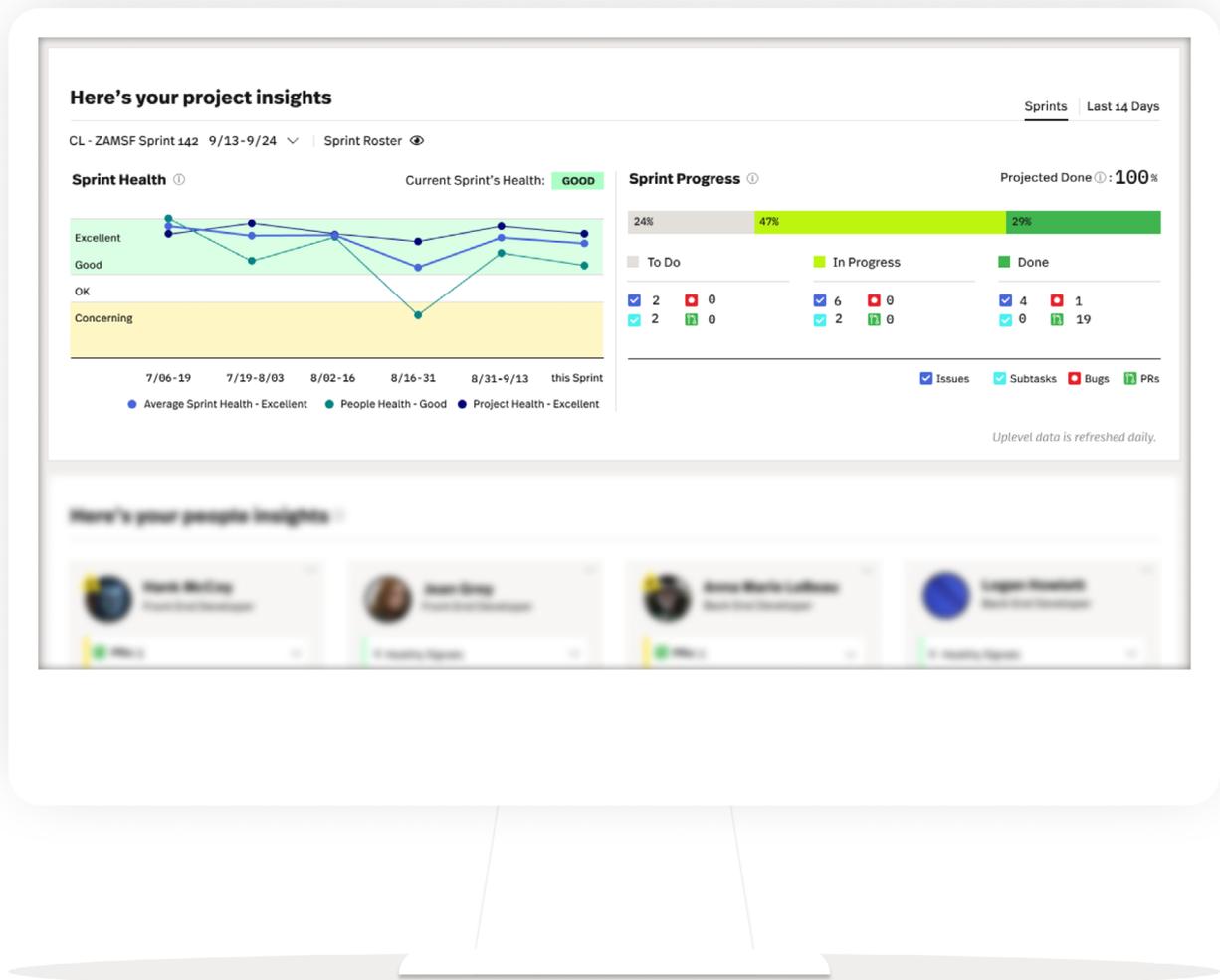
– Zipwhip Manager



## Further Benefits

# Higher Sprint Health

Two additional metrics that Zipwhip managers use to keep a pulse on their team are Sprint Health and Sprint Progress. Sprint Health combines both People Health and Project Health metrics to give visibility into how the current sprint is going and compares to past sprints for context. Managers use the Sprint Progress bar to understand the number and proportion of sprint goals that are To Do, In Progress, or Done. Based on current standing, Uplevel calculates a predicted percentage of “Projected Done” by the sprint end date. A recent look at this manager’s team shows a sprint with 100% projected completion, as well as Excellent or Good Sprint Health every sprint since July.



**“Our investments in the transformation journey have enabled us to deliver high-quality results consistently within a predictable timeframe, achieve a good work/life balance for the team, and increase the motivation of the team to welcome new challenges and opportunity to learn and grow.”**

**– Zipwhip Manager**

Ready to transform your team using Uplevel data?  
Schedule a demo today.

[uplevelteam.com](https://uplevelteam.com)

