Uplevel

Data sources for Uplevel visualizations

The following include descriptions of all data sources for each visualization in the Uplevel dashboard. You will also find descriptions under each chat in the dashboard, but we've added a bit more detail here.

Deep Work:
- We track Deep Work as uninterrupted blocks of time of two or more hours during a normal workday. We deduct time when we've detected that you've been interrupted in Slack:
  - If a message comes to you and you respond right away with a short reply, we give a small penalty. If it's a back and forth conversation, we know this causes more context switching and give a much higher penalty + time to get back into what you're doing, based on academic research.
  - If you wait and respond, we don't count this as a penalty
- Additional time elements:
  - "Short Fragments" are defined as blocks of time less than two hours where you are not in a meeting.
  - "Busy/Blocked" is for lunch and other non-meeting on the calendar.
  - "Meetings" are defined as events on your calendar where there are at least two attendees.
  - Interruptions are based on time spent responding to incoming messages on Slack during otherwise Deep Work time.

Meetings:
- Based on your calendar data from the last two weeks where there are more than 1 person on the meeting invite (we don't count meetings you set with yourself).
- If you have two overlapping meetings confirmed at the same time, we do our best to guess which one you attended based on acceptance status and number of attendees. E.g. if you were tentative to one and accepted the other, we assume you went to the one you accepted.
- If you are tentative to a meeting, we treat as attended unless there is an overlapping meeting.
- If you accept a meeting and don't attend, there is no way for us to know that information and we count as attended.

Multitasking in meetings:
- In this case, "multitasking" means sending Slack messages during meeting time. We ignore quick messages sent during the first or last five minutes ("where u at?"), but anything longer counts.

Jira data:
- Pulled from your Jira account and shows the set of Jira tickets your team has touched (edited or been assigned) within the last two weeks. The colors show how the tickets change status over time. Work is grouped by its most recent status and sorted by ticket age, with work that has been open longest at the top.
- *Note: we've received feedback on the Jira data that is not reflected from some users and are working to resolve the issue.*

Pull request activity:
- This shows your team's workflow of pull requests over the course of two recent weeks. Note that we slightly approximated some dates in order to improve legibility of the data, but the volume of comments and updates is accurate.

Story points:
- Shows the number of recent tickets assigned to your team that have Story Point estimates recorded in Jira. This includes only Stories and Tasks that are currently open, and assigned to your team within the last two weeks.

Out of hours:
- Since schedules vary, we dynamically calculated the most common 9-hour working window for your team and simply indicated communication that occurred before or after that window. Data for your most recent week is updated daily, so if you view the data earlier in the week it will reflect fewer days than later in the week.
- If you change your workday, we will detect that over time.

Epics:
- This looks at the total number of ticket updates within each epic, such as marking a ticket as "In Progress" or adding a comment. The pie chart represents your team's weekly distribution of Jira activity, so you can see which epics are taking up the most time.

Team distribution of Jira tickets:
- The distribution of your team's recent Jira activity, divided by team member and type of task. Activity is defined as any activity on a Jira ticket, including opening, commenting, editing and closing a ticket. Data for your most recent week is updated daily, so if you view the data earlier in the week it will reflect fewer days than later in the week.

Code commits:
- You can see your team's code commits, organized above by developer and repo. Note that code commits appear in these results only if they eventually make it up to the default branch.

Cross team collaboration - Jira:
- Shows the relative distribution of other teams (labeled by manager) that your team is partnering with in Jira. We look at which teams are opening or updating tickets that your team is working on. Larger circles indicate more frequent partnership. Your own team is also shown to give context on how often your team collaborates within the team vs cross team.

Weekly collaboration - meetings:
- Shows the average time you're spending each week with colleagues and team members, based on two recent weeks of calendar data. Larger circles indicate more collaboration time in a week. Note: This only represents your meetings with engineering - not a partner in marketing or sales, for example.

Team 1:1s:
- Shows which weeks you held 1:1s with individual contributors who report directly to you. If you met informally or in a group setting, it won't show up here. Note that if you reset the start date of a recurring 1:1, the history won't be reflected in this chart.

Weekly facetime:
- We checked calendar data to calculate the average time per week that you're in scheduled meetings with each of the individual contributors who directly report to you. This does not include 1:1s, which are in the chart above, or large meetings with more than 20 attendees. Data for your most recent week is updated daily, so if you view the data earlier in the week it will reflect fewer days than later in the week.

Always on:
- Shows the estimated risk (Normal, Above Normal, or High) of being on for more than 8 hours per work day or on non-work days, potentially indicating difficulty turning off and possibly experiencing burnout.
  - Activities include:
    - Slack messaging, Jira activity, meetings, and source code (PR commits, PR open, PR activity [comment, request changes, approve], and PR merges)
      - Note: We do not account for activities for which we don't have data (e.g. email, presentation slides, documents, etc.), so if someone is doing a large amount of work in other places they may show up as being "Normal" risk, but actually be "High" risk, so make sure you check in with them regularly.
    - Activity is computed by estimating how many minutes each activity typically takes
  - Contributions to always on risk:

- Activity that occurs outside of the most active 8 hours on work days will contribute towards being always on
- Any activity on weekends, holidays, or full days OOO will contribute towards being always on
  - Risk buckets:
    - "Normal": A small amount of additional minutes of work day activity will show as a "Normal" risk of being always on
    - "Above Normal": A moderate amount of additional minutes of work day activity or a small amount of activity on non-work days will show as "Above Normal" risk of being always on
    - "High": A large amount of additional minutes of work day activity or at least a moderate amount of activity on non-work days will show as "High" risk of being always on

KTLO:
- Defined as any tickets with the phrase "KTLO" or "keep the lights on" in the title of part of an epic that has one of those in the title.
- Shows number of tickets open and assigned to your team within the last 2 weeks and how many are KTLO.

Context Switching:
- For each team member, shows the relative amount of context switching (Less, About the same, or More) sources participated in during the week compared to that individual's average amount over the prior 12 weeks. This is a proxy for how many different things people are focusing on throughout the day. Occasional periods of more context switching are normal, but sustained periods of increased context switching may be frantic and frustrating for some individuals and be a factor in burnout.
  - Sources include:
    - Number of PR repos where the individual opened, merged, approved, commented on, or requested changes on a PR
    - Number of Jira epics where the individual edited one or more of the status, resolution, summary, description, comment, attachment, or time spent
    - Number of P1 (high priority) bug issues where the individual edited one or more of the status, resolution, summary, description, comment, attachment, or time spent
    - Number of Slack channels where the individual was interrupted
    - Number of meetings the individual attended with 2 or more teams
    - Note: We do not account for sources for which we don't have data (e.g. email, presentation slides, documents, etc.), so an individual's participation in these sources is not directly captured. Therefore more or

less activity in those sources won't be reflected, so make sure you check in with them regularly.

- ○ Context Switching buckets:
  - ■ "Less": When the number of total sources an individual participated in for the week decreases by at least -20% compared to the average over the prior 8 weeks, the amount of context switching is "Less"
  - ■ "About the same": When the number of total sources an individual participated in for the week is within +/-20% of the average over the prior 8 weeks, the amount of context switching is "About the same"
  - ■ "More": When the number of total sources an individual participated in for the week increases by at least +20% compared to the average over the prior 8 weeks, the amount of context switching is "More"

Project Explorer:
- ● The Project Explorer feature gives managers and engineers visibility into key project details across team members and various date ranges. Specifically, managers can reduce execution risk by seeing if unplanned work is dominating a sprint, and if any engineers are at risk for burnout. Engineers can track and view past and current projects as well as understand collaboration pattern data to ask for help when needed.
- ● The following key insights and features are included in the Project Explorer:
  - ○ **Associated PRs:** PRs that are linked to a Jira ticket. We determine this by a Jira ticket ID appearing in the title, description, or branch name of a PR that was opened, commented on, and/or merged in the given date range.
  - ○ **Jira issues:** Any Jira issue with a ticket type other than bug. This does not include subtasks.
  - ○ **Jira bugs:** Any Jira ticket issue with a ticket type of a bug.
  - ○ **Assignee/author vs. contributor/reviewer**: The assignee/author is the person currently responsible for the Jira ticket or the author of a PR for the date range selected. A contributor/reviewer is someone who may have been assigned the ticket in the past and/or commented on the PR if there was a PR associated with that Jira ticket.
  - ○ **Date range**: Users have the ability to select a variety of date ranges. Once a date range is selected, the Project Explorer only shows Jira issues and/or PRs that have had activity during that date range (i.e. comment on a PR, status change on a Jira ticket, etc.) so this would not include tickets that are perhaps in a backlog with no activity.
- ● Additionally, Project Explorer has specific "**callouts**" that you can find in the details page of each epic by looking for the highlighted row with a yellow triangle caution symbol.
  - ○ "This item was added mid-sprint": This identifies issues added late in the sprint to get an idea of scope creep

- ○ "This item has spanned X sprints": Calling out work that's been open multiple sprints or stalled to see if your team needs help closing
- ○ "This PR is stalled": Brings attention to stuck pull requests to see if you need to provide extra assistance or check in
- ○ "This PR was merged without approval": Calls out PRs that were merged into the master branch without someone reviewing and approving
- ○ "Low complexity and highly discussed": Indicates that there might be an issue with a PR that could use assistance

Sprint Health Score

There are a variety of different factors that affect a sprint. In order to give managers a holistic view of their sprints that can be tracked over time, Uplevel created the Sprint Health Score. The score brings in information about Jira and PR activity during the sprint and combines that with people data to measure potential burnout. The specific data point that are tracked are:

Jira:
- Bugs - the fewer, the better
- Ticket Carryover - the fewer, the better
- Tickets added mid sprint - the fewer, the better
- Number of tickets closed - the more the better
- Story points and descriptions - the more the better

PRs
- PRs merged without approval - the fewer, the better
- PRs without reviewers - the fewer, the better
- Stuck PRs - the fewer, the better

People - *this includes everyone in the sprint, not just your team*
- Always On - the more people in the "Normal" range, the better
- Context Switching - the more people in the "Less" category, the better
- Deep Work - the fewer major changes, compared to your team's baseline, the better
- Slack Interruptions - the fewer major changes, compared to your team's baseline, the better

Sprint Progress Data Sources

The Sprint Progress feature breaks up the work that's assigned in your sprint into the following categories - pulled from your Jira tickets:
- To Do

- In Progress
- Done

This covers all work planned in this sprint, including unassigned work and work assigned to other teams. Adding or removing work from the sprint will update this projection. It then looks at your team's historical velocity and estimates how much of the assigned work will get completed by the end of the sprint—we call this number the Projected Done Percentage.

PR Cycle Time Data Sources

The PR Cycle time metric shows how long the average lifecycle of a PR is on your team. It breaks down the PR into stages, measured in hours:
- Dev: the time from the first commit, until the PR is opened. Draft PRs are considered to be still in the dev phase.
- Waiting for Review: the time from PR open until a first reply from a reviewer.
- Review: the time from first reply until the PR is merged.
- Release: the time from merge until it appears in a release branch.

Note: these summary median times only include PRs to the default branch, in order to not double count PRs that move code from the default branch to a release branch.