Review Instructions

Hi team,

Thanks for helping the Product team on this.

Why?

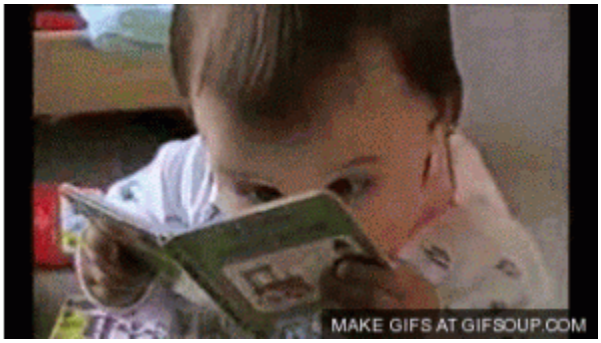
The goal of this documentation is to empower new users to have a smooth onboarding experience for our PS Tool. Once finalized, this documentation will be made publicly available on our website for everyone to use.

What?

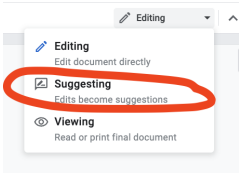
When reviewing this documentation, please keep asking yourself the following questions:

- If you are new to our tool, do you find our messaging clear? Is it helpful?
- Anything you feel important to add to the documentation?
- Anything you feel is redundant and should be removed?
- Anything you feel is misleading to our new users?If yes, how would you rewrite it?

How can you help?



Please turn on the suggesting feature of Google doc on the right hand side and make any comments or suggestions directly in this doc.



Once this has been reviewed, Product will make all the relevant updates and publish the documentation. Please review this by the end of the week (10/08/22).

Thanks so much,
Your Product Team

Getting started

Mobile App Performance Optimization – what is it?

Mobile app performance optimization refers to identifying and improving code that plays a significant role in user experience – such as app start time, the loading time between screens, or time spent completing tasks.

In the ever-growing app market, users can easily uninstall slow mobile apps and swiftly try the next app on the list. Mobile app performance doesn't just refer to codebase quality and infrastructure – the user experience of the application's responsiveness defines it.

Along with that, Android and iOS storefronts incentivize app performance by giving lower search rankings to less-performant apps. App stores do this because more-performant apps provide a better UX and deliver more business value which we'll see below.

Why is Mobile App Performance Optimization Important?

Brand Preference

52%+ users suggested they are less likely to interact with a brand after a poor mobile app experience.

Brand Reputation

36% of users who had experienced slow performance issues had a lower opinion of the company.

Customer Retention

96% of users say app performance factors such as speed and responsiveness matters when deciding whether to keep or uninstall an app.

What is PS Tool?

If you're unfamiliar with the concept of tracing, performance analysis can initially seem overwhelming. Our PS tool will change the way you think about performance engineering. With our visualizations, you will effortlessly understand how every function relates to another and identify relevant code optimization opportunities.

At Product Science (PS), we offer a set of tools for mobile app performance engineering that includes:

- Dynamic automatic code instrumentation via AI-powered plugins added to the build process;
- PS multi-threaded code profiling tool highlights critical functions and frameworks that impact user experience, revealing the root cause of problems;
- PS Companion mobile app to record and upload traces from your mobile device.

By replacing manual instrumentation and embedding right into the build processes, we enable anyone to identify points of app causation of performance issues with clear visualization.

Who is PS Tool for?

PS Tool empowers anyone who understand code to use our visualization tool powered by AI to find performance optimization insights.

How does PS Tool work?

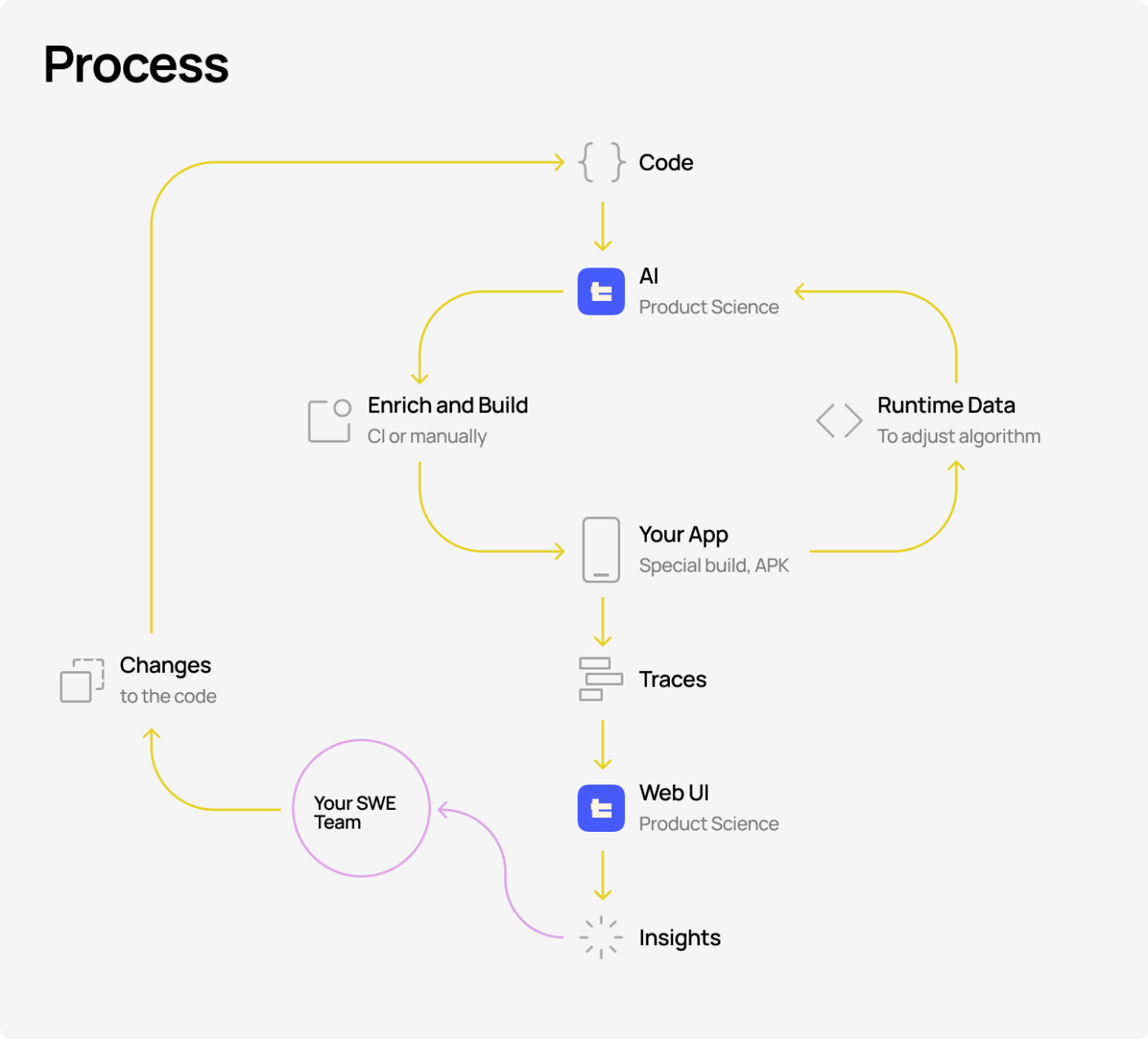
Start with instrumenting your app's code with our PS Gradle plugin / Xcode Code Injector powered by AI which only visualizes functions that impact your mobile app's performance.

PS Tool profiles and visualizes recorded (user) flows, and our AI will then suggest execution paths – the sequence of functions executed – that empower identification of performance opportunities.

> New to instrumentation? Learn what it is here

How to see performance optimization insights, step by step:

1. Use our plugin during the build process to instrument your code
   > Learn How To
2. Run the build on the target device
   > Learn How To
3. Record the trace and video while the app runs and walk through a flow that you want to optimize performance for
   > Learn How To
4. Upload the recorded trace and video using our PS Companion App
   > Learn How To
5. Visualize insights with PS Tool
   > Learn How To



Read Essential Steps to learn more

Essential Steps

To get performance optimization insights from your code with our PS Tool, follow the step-by-step below:

1. Instrument and build the app with our plugin to analyze your code.

How to Instrument

Android / Gradle
iOS

2. Record trace and video by running your instrumented app on the target device and walk through use flow you want to optimize.

> What is a flow? Learn more about it.
> How can screen recording helps to find insights faster? Learn more about it.

Android

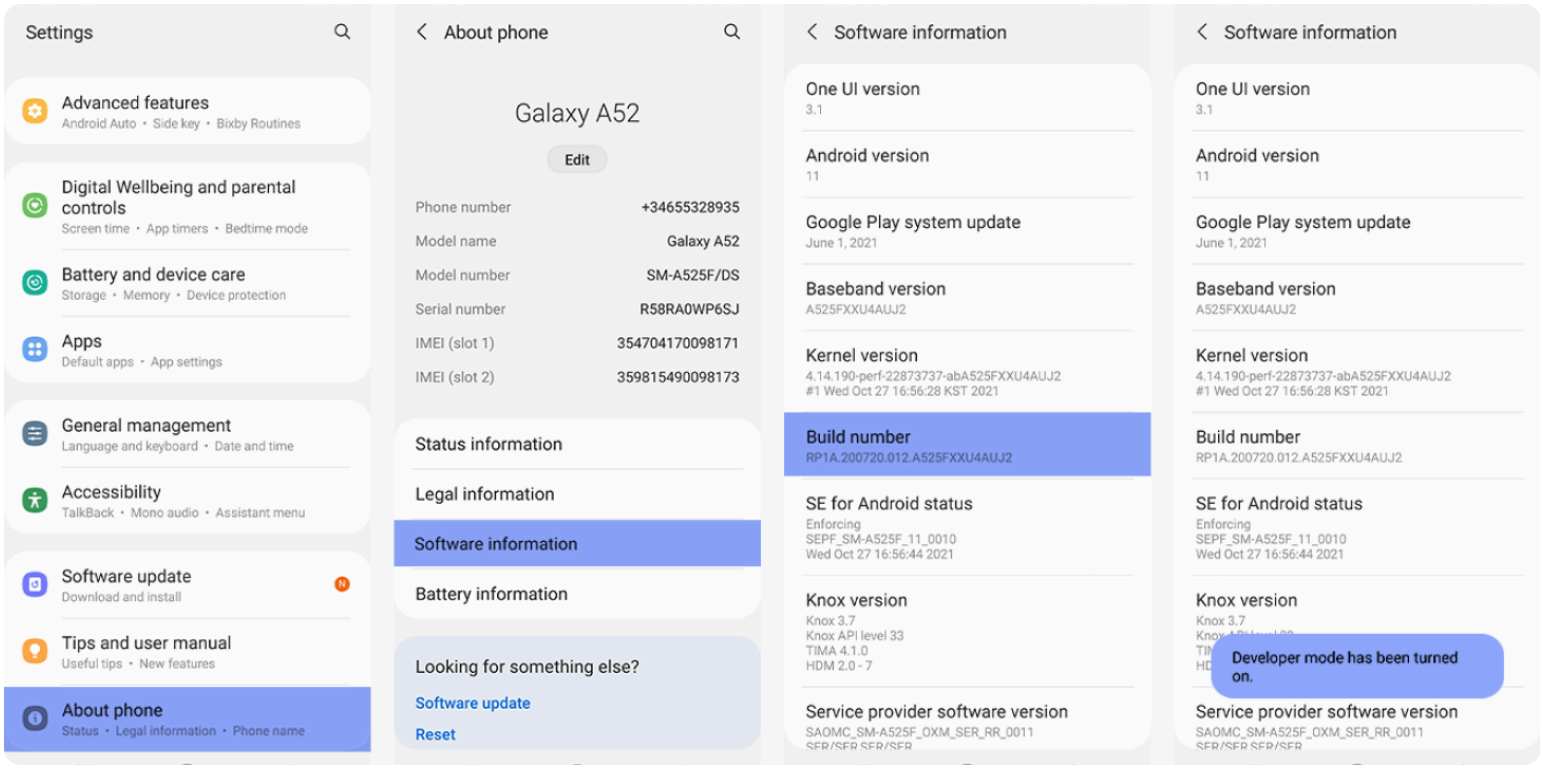
Preparation (Only Needs to be Done Once)

> Install the PS Companion app

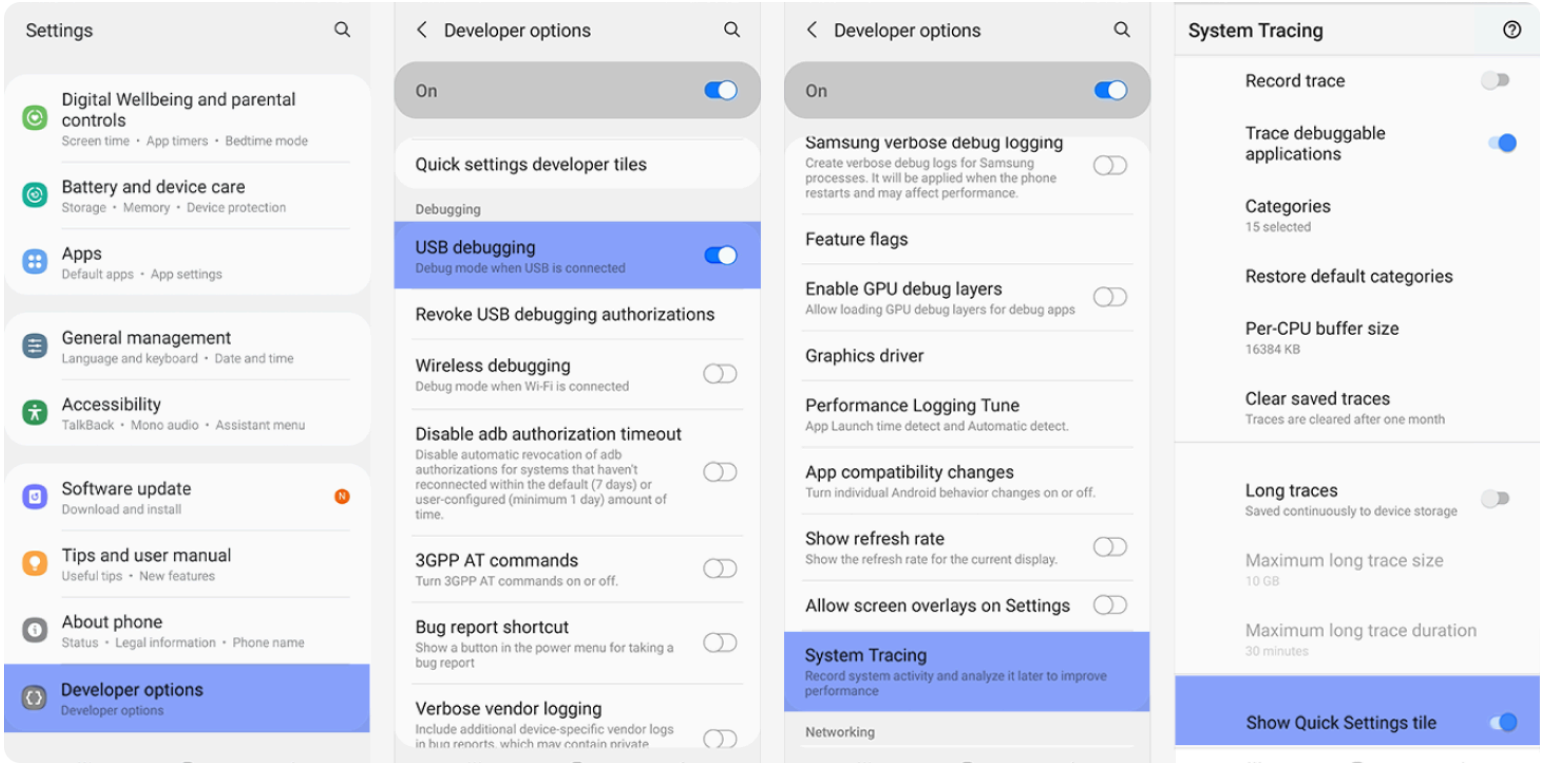
- Make sure to log in before moving on to the next step.

> Enable tracing

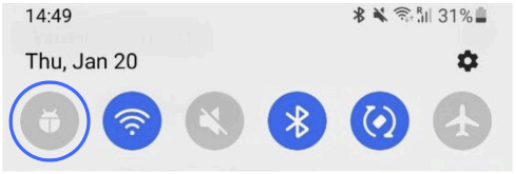
1. First, you'll need to enable developer options and USB debugging.
    - Settings > About Phone > Software information > Build Number.
    - Tap the Build Number option **7 times.**



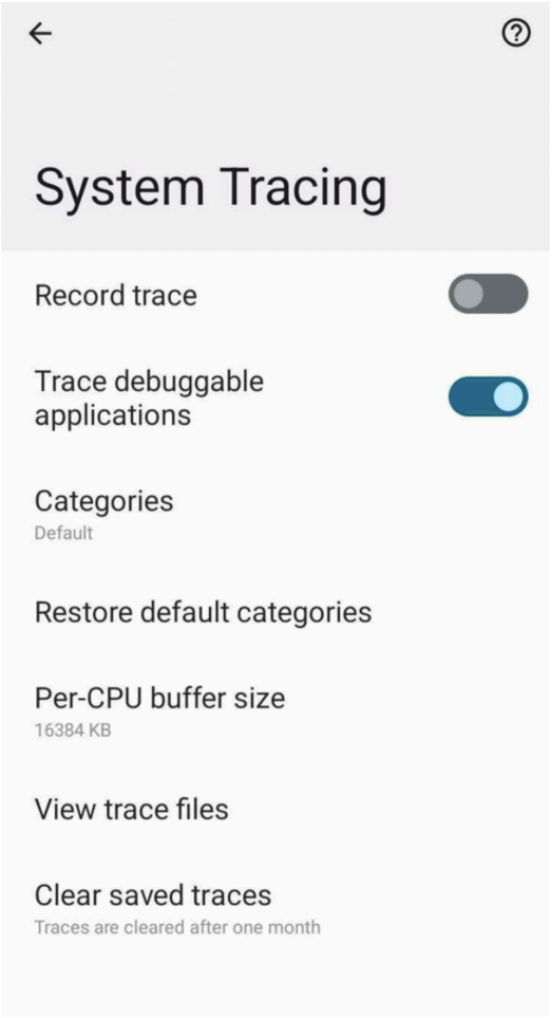
2. Then, enable tracing.
    - Settings > System > Developer Options > Debugging section> System Tracing.
    - Make sure to turn off all systems tracing categories except `view: View System`.



- Enable **Show Quick Settings tile**. This will add System Tracing tile to the Quick Settings panel, which appears as an upper panel:



- Hold down the ‘‘bug’ button to yield the full trace menu.
- Enable “Long traces”



Trace and Screen Recording

> Install instrumented app

- Install the build of the app instrumented in the previous steps.
  Make sure your app is instrumented!

> Screen recording (optional but highly recommend)

Capturing how your application functions in real life provides valuable context that helps to understand what's happening - even when the code is hard to follow. With screen recording, you can see when the user interacts with the application (action: start of the user flow) and when the application screen is updated (reaction: end of the user flow). We highly recommend you to record your screen to complement your trace analysis.
> Learn more about how to make the most out of videos.

Record your phone screen (Source: Google)

1. Swipe down twice from the top of your screen.
2. Tap Screen record .
   - You might need to swipe right to find it.
   - If it's not there, tap Edit and drag Screen record to your Quick Settings.
3. Choose what you want to record and tap Start. The recording begins after the countdown.
4. To stop recording, swipe down from the top of the screen and tap the Screen recorder notification .

How to record when there is no screen recorder feature

We recommend you to download 3rd party apps such as Screen Recorder - XRecorder.

Find screen recordings (Source: Google)

1. Open your phone's Photos app .
2. Tap Library > Movies.

> Record trace

Generally, we recommend recording traces with a cold start where everything will be initialized from zero. This type of app launch is usually the slowest because the system has a lot of expensive work to do compared to the other two states (warm and hot starts where the system brings the app running from the background to the foreground).

Optimizing any user flow with cold start enables you to improve all app processes that are being created from scratch which can enhance the same user flow's performance with warm and hot starts as well.

Step	To record App Start Flow	To record any Flow other than App Start
1	Kill the targeted app.	Kill the targeted app
2	Make sure you are logged in to our PS Companion App.	Make sure you are logged in to our PS Companion App.
3	Start screen recording (optional).*	Open the targeted app.
4	Start recording a trace by selecting the bug icon or 'record trace' button from the quick settings tile.	Perform the user actions from the (user) flow and **stop before the last step**.
5	Open the targeted app.	Start screen recording (optional).*
6	-	Start recording a trace by selecting the bug icon or 'record trace' button from the quick settings tile.
7	-	Perform the last step from the (user) flow
8	Once the main page is fully loaded, tap the bug icon or 'stop tracing' button to stop recording the trace.	Once the main page is fully loaded, tap the bug icon or 'stop tracing' button to stop recording the trace.
9	Stop screen recording (if needed).	Stop screen recording (if needed).
10	Wait for a "success" dialog drawn on top of your app.	Wait for a "success" dialog drawn on top of your app.

Best Practices

- Avoid doing any unnecessary actions outside of the flow.
- Click and wait for each screen to fully load and any functions to complete before clicking on the following button (if necessary).
- Avoid performing the flow too fast, and functions from different screens might overlap.

- Avoid performing the flow too slowly, and you are further away from understanding how a user will interact with the app.
- Record your screen!
  >Why? Learn more about it.

iOS

Preparation (Only Needs to be Done Once)

> Install our PS Companion App

- Make sure to log in before moving on to the next steps

> Customize share sheet (Optional)

To make it easier for exporting your traces in the upcoming steps, we recommend to customize your iPhone share sheet so that you can have the PS companion readily available

- Launch an app containing shareable content, such as Safari, Photos or Files app
- Tap the  icon
- Share sheet appears
- If PS Companion App is not shown, swipe to the right
- Tap more 

- Tap Edit on the top right
- Tap the green plus icon for the PS Companion app

Trace and Video Recording

> Install instrumented app

- Install the build of the app instrumented in the previous steps.
  Make sure your app is instrumented!

> Screen recording (optional but highly recommend)

Capturing how your application functions in real life provides valuable context that helps to understand what's happening - even when the code is hard to follow. With screen recording, you can see when the user interacts with the application (action: start of the user flow) and when the application screen is updated (reaction: end of the user flow). We highly recommend you to record your screen to complement your trace analysis.
> Learn more about how to make the most out of videos.

Record your phone screen (Source: Apple)

1. Go to Settings > Control Center, then tap the Add button  next to Screen Recording.
2. Open Control Center on your iPhone, or on your iPad.
3. Tap the gray Record button  , then wait for the three-second countdown.
4. Exit Control Center to record your screen.
5. To stop recording, open Control Center, then tap the red Record button  . Or tap the red status bar at the top of your screen and tap Stop.

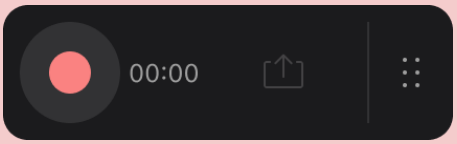
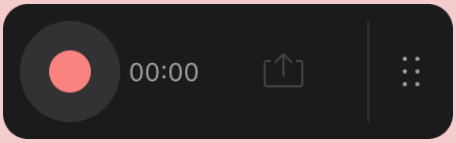
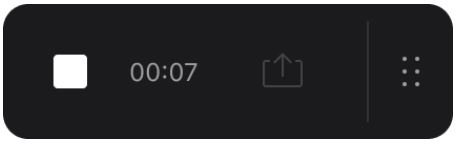
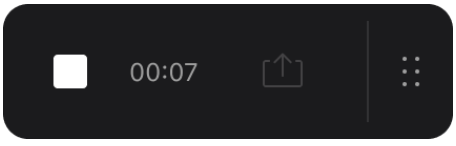
Find screen recordings

1. Go to the Photos app and select your screen recording.

> Record trace

Generally, we recommend recording traces with a cold start where everything will be initialized from zero. This type of app launch is usually the slowest because the system has a lot of expensive work to do compared to the other two states (warm and hot starts where the system brings the app running from the background to the foreground).

Optimizing any user flow with cold start enables you to improve all app processes that are being created from scratch which can enhance the same user flow's performance with warm and hot starts as well.

Step	To record App Start	To record any Flow other than App Start
1	Kill the targeted app.	Kill the targeted app.
2	Make sure you are logged in to our PS Companion App.	Make sure you are logged in to our PS Companion App.
3	Start screen recording (optional).*	Open the targeted app.
4	 Tap the button on PS Companion App to start recording	Perform the user actions from the (user) flow and **stop before the last step**.
5	Open the targeted app.	Start screen recording (optional).*
6	Perform the user actions from the (user) flow that you want to optimize.	 Tap the button on PS Companion App to start recording.
7	-	Perform the last step from the (user) flow.
	 Once the final step is fully loaded, tap the button to stop recording.	 Once the final step is fully loaded, tap the button to stop recording.

Best Practices

- Avoid doing any unnecessary actions outside of the flow.
- Click and wait for each screen to fully load and any functions to complete before clicking on the following button (if necessary).
- Avoid performing the flow too fast, and functions from different screens might overlap.

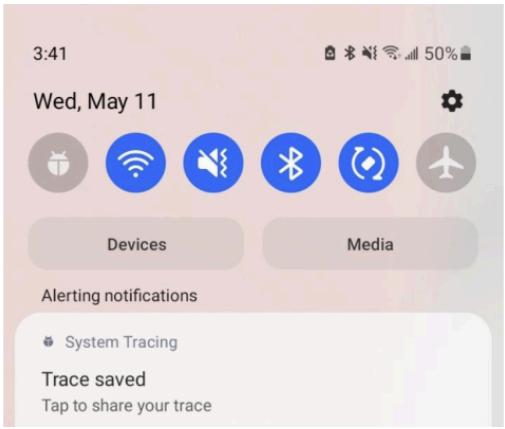
- Avoid performing the flow too slowly, and you are further away from understanding how a user will interact with the app.
- Record your screen!
  >Why? Learn more about it.

3. Upload recorded traces to PS Tool

Android

Upload trace to PS Tool

- After recording a trace file, tap the file to export it. You will have the option to save it or export it to different apps.



- Export your trace file to the PS Companion App
  - Alternatively, you can manually upload traces via web interface.
- Name the trace, assign it to the relevant flow and then upload it to our cloud.
- Uploaded trace will appear in your productscience.app Flow Library
  - *Alternatively, you can enable the "subscribe" functionality in the Flow Library to get an email containing a link to a new trace when it uploads.*
  - *Want to upload it via web interface? Learn how to.*
- For any errors, our system will return a message explaining what went wrong. If you can't resolve the issue, please contact us at suppot@productsience.ai.

PS Companion Mobile App Alternative

For the most seamless experience, we highly recommend using our mobile app for uploading traces. But in instances where that fails, you can also upload traces via:

> Export trace from the Files app

- On mobile devices running Android 10 (API level 29), traces are shown in the Files app

> Download traces with command line (Optional)

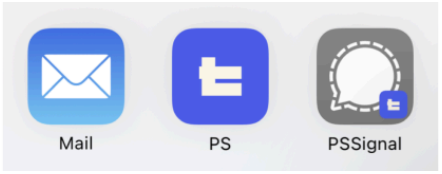
- Connect your android device to a computer via USB
- Command-line for downloading traces: adb pull /data/local/traces/

> Manual upload via Web Interface

- > Learn How To

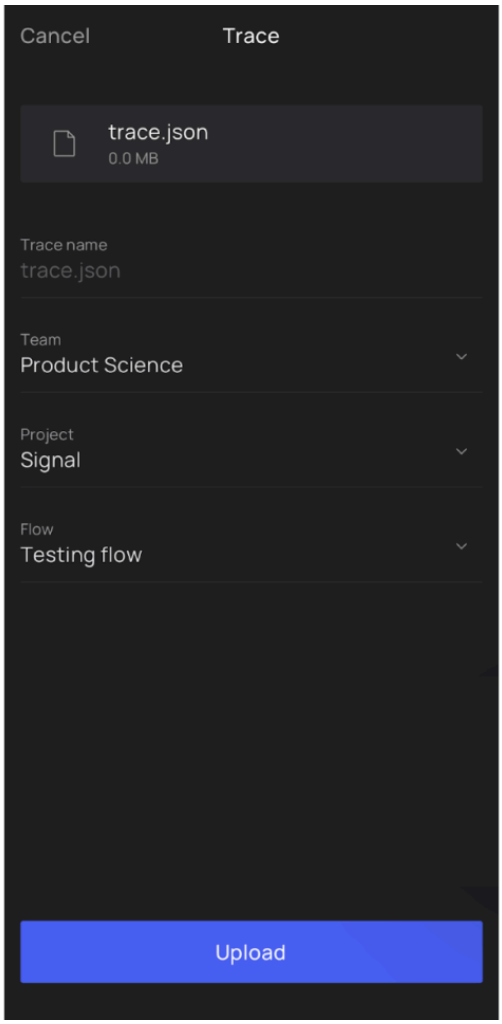
Upload trace to PS Tool with PS Companion App

- Tap  to export your trace file.
- Open it with our PS companion app.



Don't see PS Companion app? Customize the share sheet

- Name the trace, assign it to the relevant flow and then upload it to our cloud.



- Uploaded trace will appear in your productscience.app Flow Library.
  - *Alternatively, you can open the trace file via the link we sent to your email for every successful upload.*
  - Or, upload via web interface. Learn how to.
- For any errors, our system will return a message explaining what went wrong. If you can't resolve the issue, please contact us at suppot@productsience.ai.

View uploaded trace

- You can find previously recorded traces organized by flow under the Discover Projects tab.
- Select a flow to see the uploaded traces corresponding to that flow.

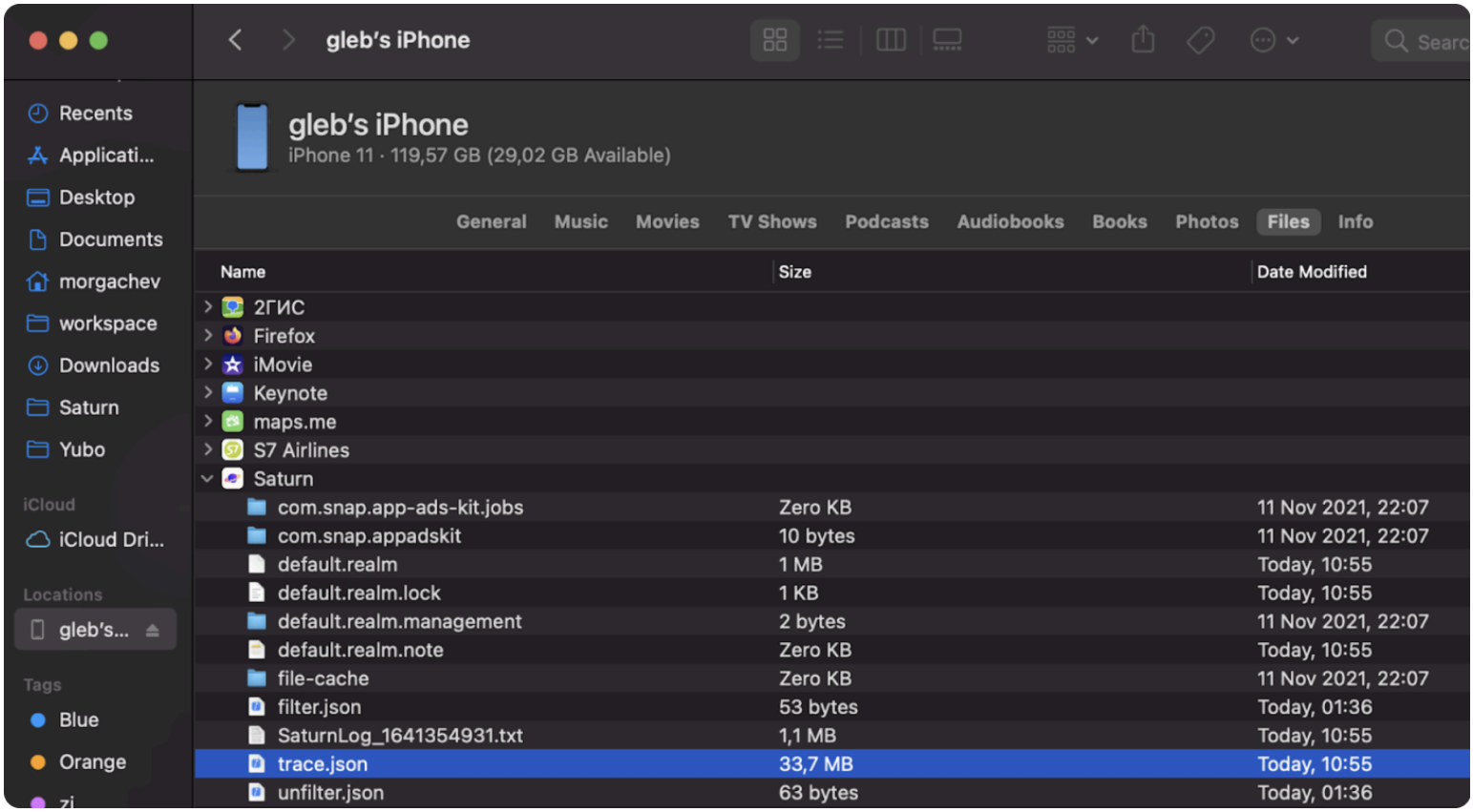
Delete trace

- To delete an uploaded trace, swipe left and select the trash icon.

PS Companion Mobile App Alternative

> Export trace file to a computer (Optional)

1. Open Finder > Locations > iPhone > Files
2. Select the Application
3. Airdrop or share the "trace.json" via iCloud to your computer



> Download with command line (Optional)

The open source app [ios-deploy](https://github.com/ios-control/ios-deploy) can be used for getting traces.

- Install tool with Homebrew:

```
brew install ios-deploy
```

- Download `trace.json`:

```
ios-deploy --bundle_id <APP'S BUNDLE ID>\\
           --download=/Documents/trace.json \\
           --to .
```

Manually upload trace to flow library

[> Learn How To]

Record Trace with Video

With the **Video synced with trace** feature you will be able to see what's happening on the phone screen at every point of the recorded trace.

Video synced with trace feature allows you to**:**

- Easily find the beginning and end of user flows
- See the user actions in real life
- Visually identify performance opportunities (like jitteriness, lags and long network requests)
- Visually identify when the screen was updated

The solid visual cues demonstrate precisely how the app works for your user or errors your app may be receiving that cannot be gained from the code alone.

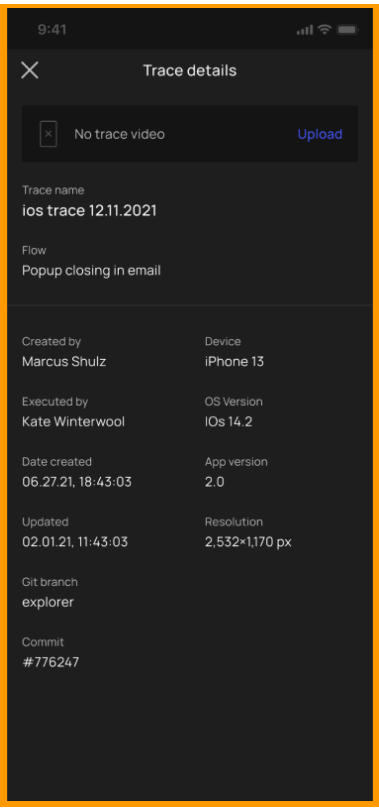
[> Learn how to record your screens. (Android)]
[> Learn how to record your screens. (iOS)]
[> Learn more about how to make the most out of the videos.]

Upload Screen Recordings

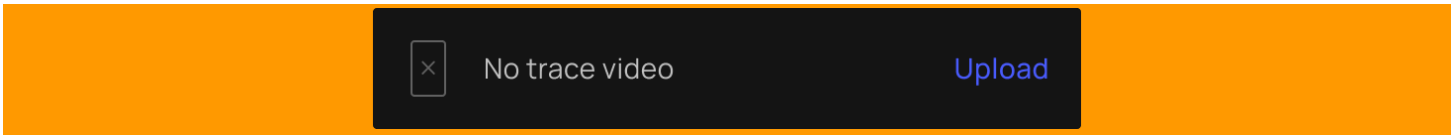
You can upload screen recordings to our tool either with our [companion app] or directly with [PS Tool].

Via PS Companion App

- In Flow > Select a user flow > Trace details screen opens.



- Tap 'Upload'.

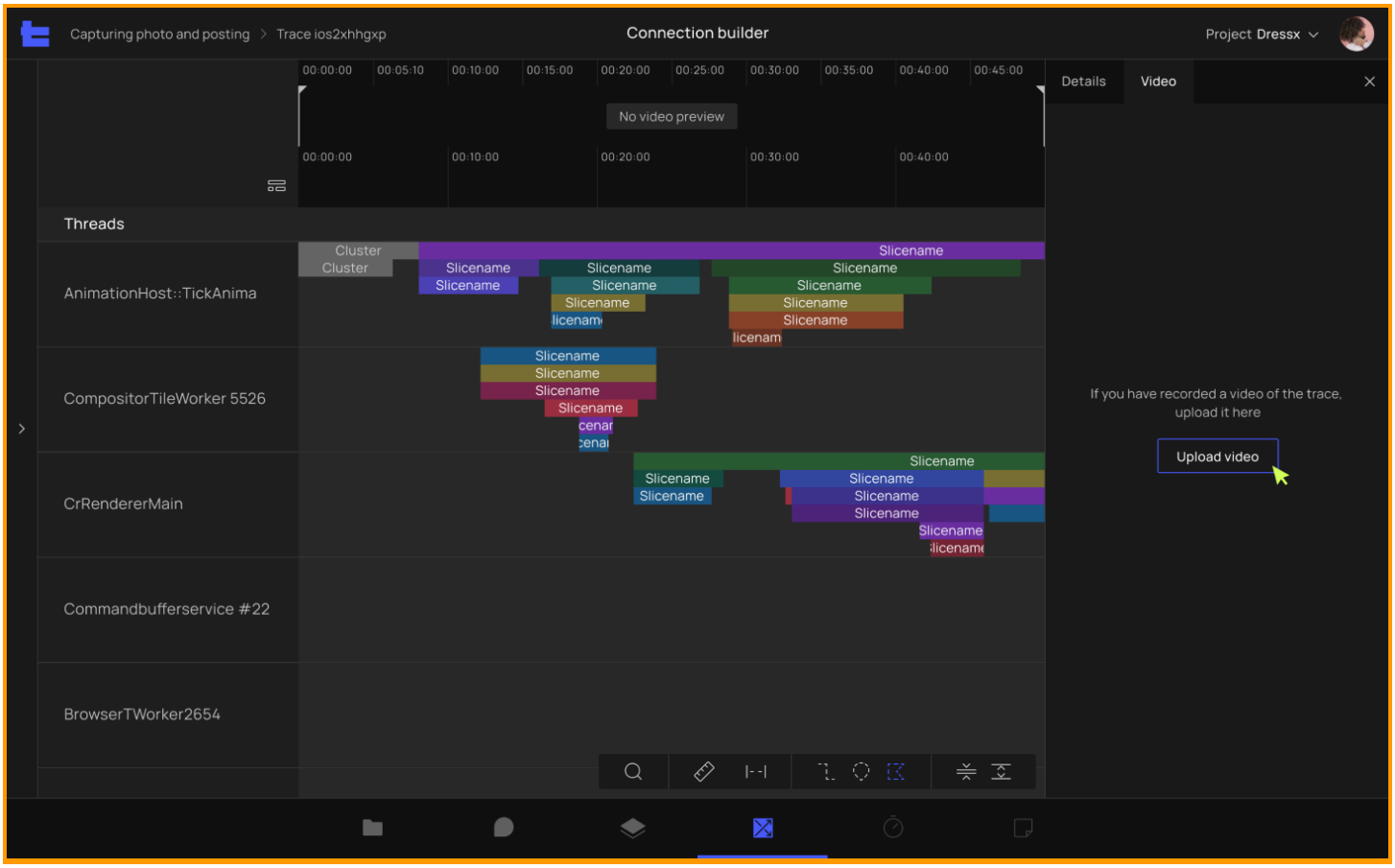


- Phone library opens > select the screen recording > ‘Upload’.
- Wait for the status bar to change from ‘Video processing in progress’ to ‘Video upload successfully’.
- Upload success. You can now view and annotate the video in the PS Tool.

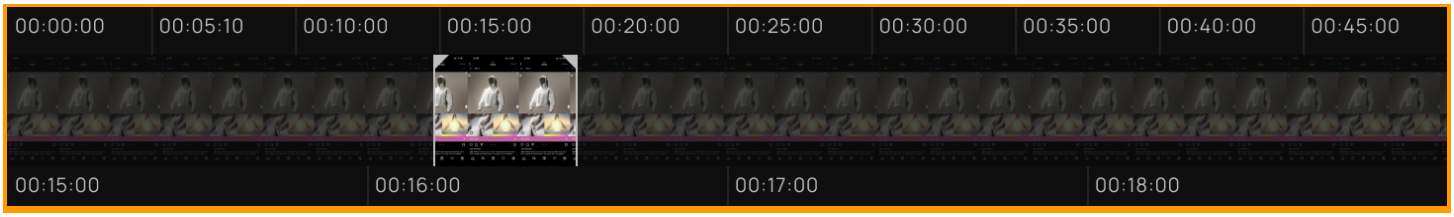
> Learn more about how to make the most out of the videos.

Via PS Tool

- In Trace Viewer > Open the Video Panel on the right.



- Click ‘Upload a video.’
- Drag and drop or browse to upload a screen recording.
- Once the upload is completed, you will see the global timeline replaced by the video.



> Learn more about how to make the most out of the videos.

Delete Screen Recordings

You can delete screen recordings uploaded either with our companion app or directly with PS Tool.

Via PS Companion App

- In Flow > Select a user flow > Trace details screen opens.
- Tap ‘Delete’.

Via PS Tool

- In Trace Viewer > open up the **Video** panel on the right.
- Click the trash icon.

> Learn more about how to make the most out of the videos.

Working with PS Tool

At Product Science, performance optimization starts with defining key (user) flows that bring the most value to users. We visualize your code that empowers you to capture optimization opportunities that impact user experiences. You can optimize how quickly the app starts up, how its processes perform, how smooth animation is, etc.

> Learn how to create flows and upload traces in the Flow Library section

At the center of the PSTool is the trace viewer which shows essential information, such as what functions were invoked, their duration, and execution paths. Visualizing execution paths here is critical because it allows you to see the sequence of functions and how they connect and highlight optimization opportunities.

> Learn how to work with traces in the Trace Viewer section

Accessing the PS Tool

Visit productscience.app and log in using your company email.

Flow Library

Flow library organizes all traces by (user) flow. This is where you can view, subscribe, manage and create flows. Like a folder in a file system, each flow contains traces of user flows that your team records and optimizes. Use Flow Library to group traces by flows that make the most sense to you and your team. Add a description to communicate the context of the flows.

Open a trace in PS Tool

Flow Library > Select a flow card > Flow Table > right click any trace > click “open”

Create new flow

1. In Flow Library > click “Create new flow”

2. Enter your flow name & description

(User) Flow Screen

>Add Flow Screen Image

The Flow Screen is where you can find all the traces uploaded associated with a single flow. This is where you can add, delete, edit, and assign your trace.

We encourage you to record more than one trace for every flow. More recordings can increase the statistical significance of your “findings”. Specifically for flows containing asynchronous I/O operations like network requests, recording multiple traces can increase your confidence in locating patterns of performance issues within the trace.

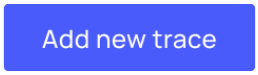
Add new trace

Via our PS Companion app

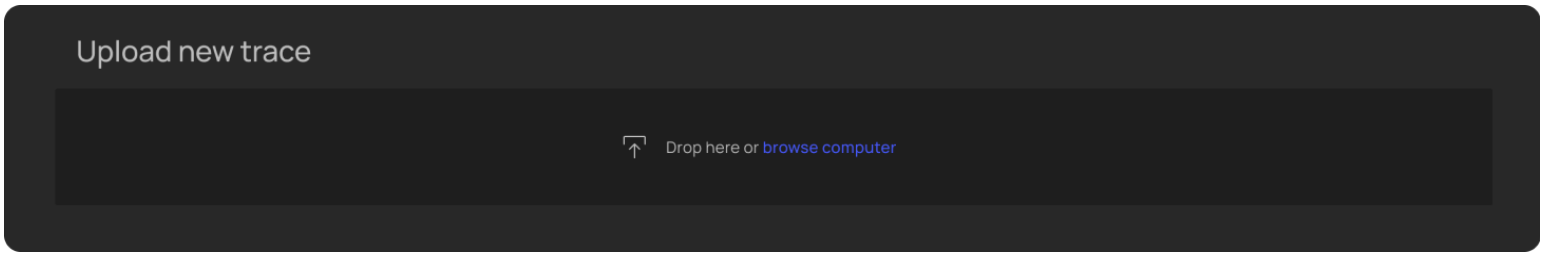
> Learn How To

Via manual upload

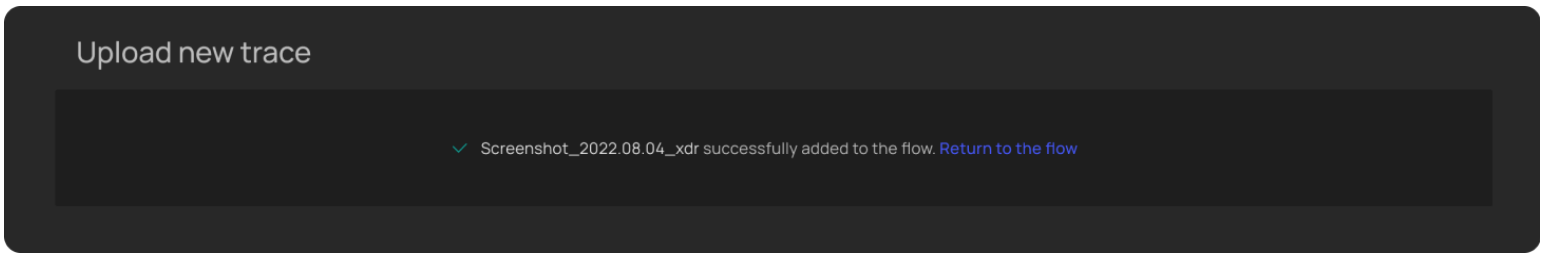
1. Click the “Add new trace” button on the Flow screen



2. Upload traces by dragging the trace or browse from computer



3. Once the trace is uploaded successfully, you shall see the message

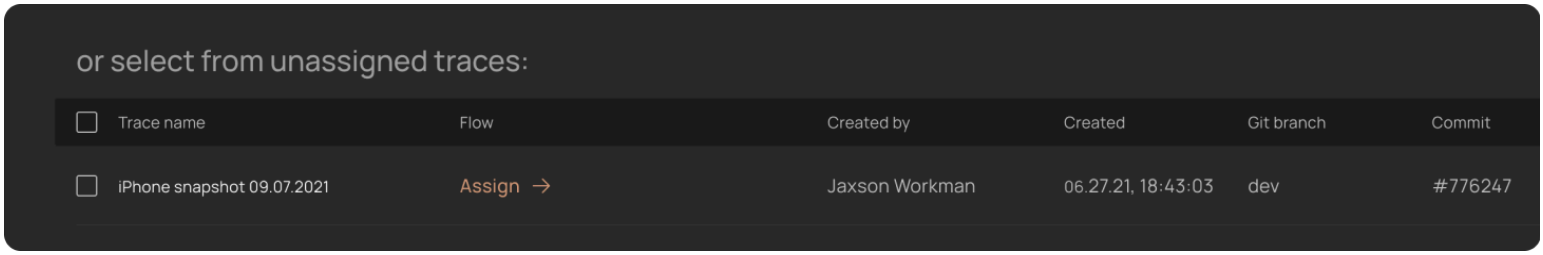


Via select from unassigned

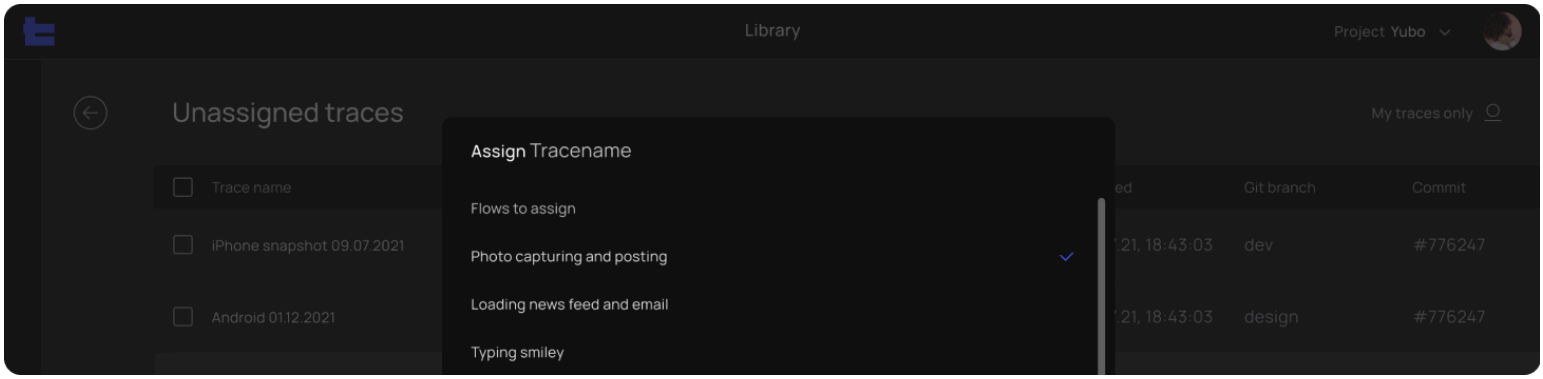
Alternatively, you can use traces that were previously uploaded but not assigned.

1. Click on the “Review and assign” button

2. Check the box for the trace you would like to assign

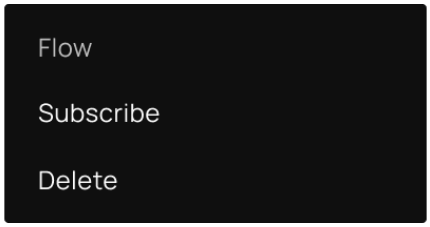


3. Select the flow you would like to flow to be assigned to



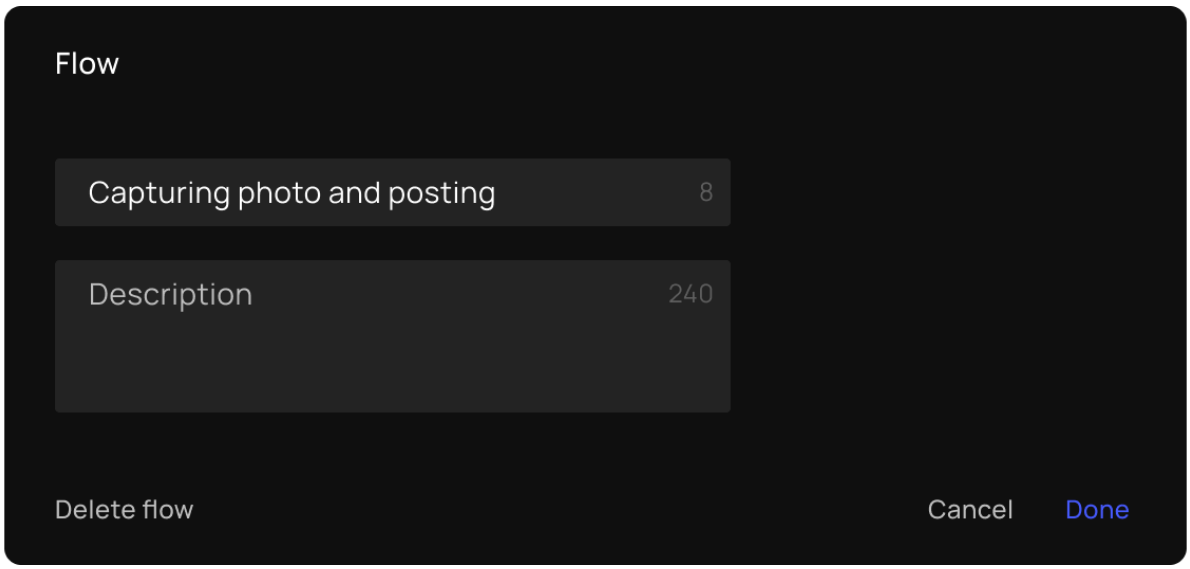
Delete flow

1. In Flow Library > hover over the bottom right corner of the flow
2. Click on the “...” button on the flow card



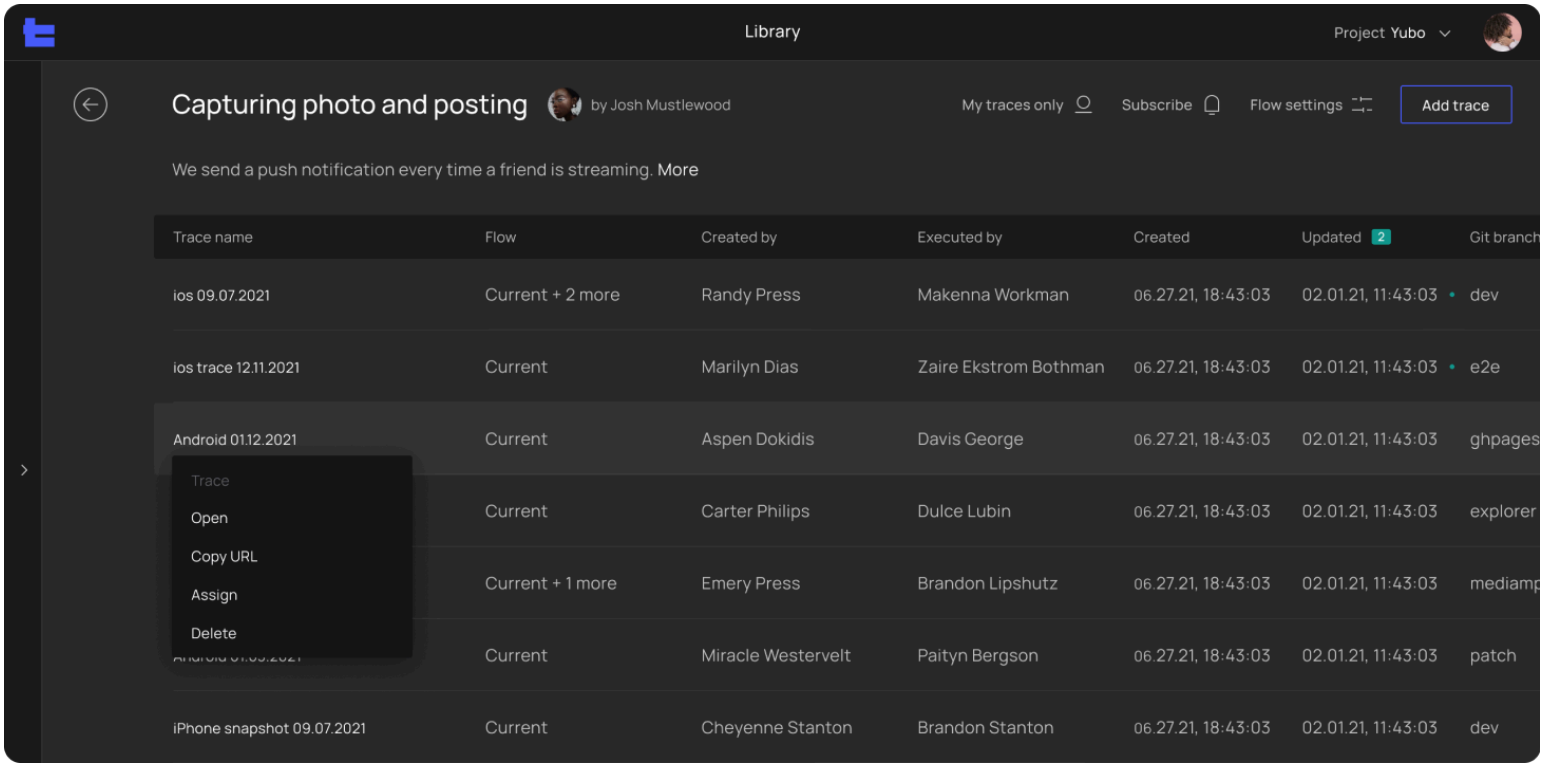
Edit flow description

1. Flow Library > Flow card > Flow settings

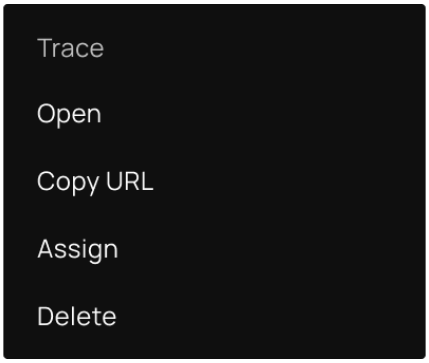


Assign trace to another flow

1. In the Flow screen

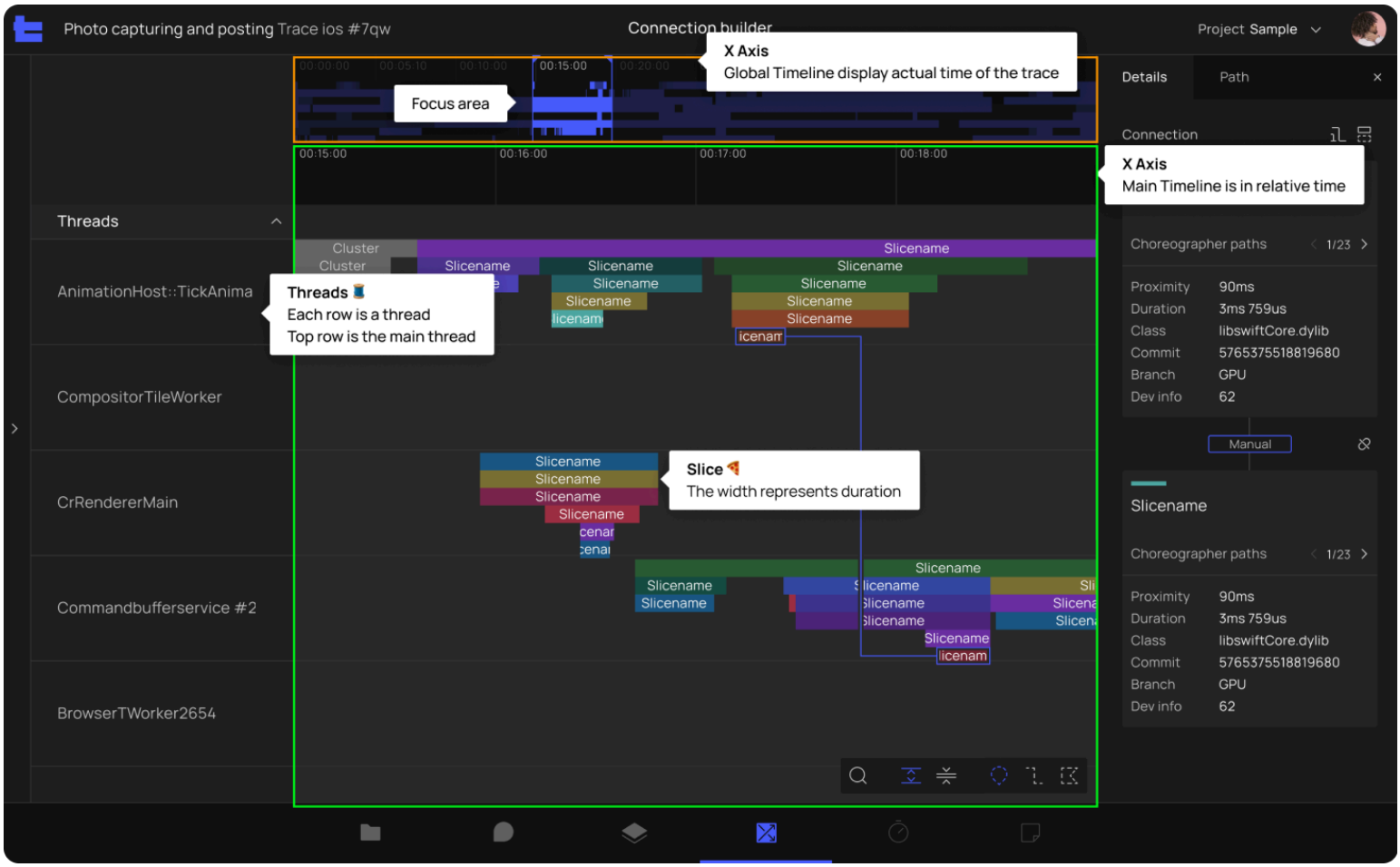


2. Click on the trace > menu will appear > click ‘Assign’



3. Select one or multiple flows you want to assign the trace to > Click ‘Done’

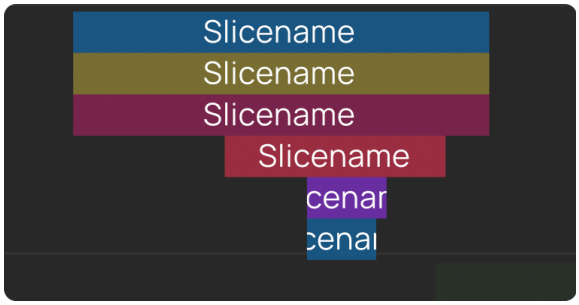
Trace Viewer



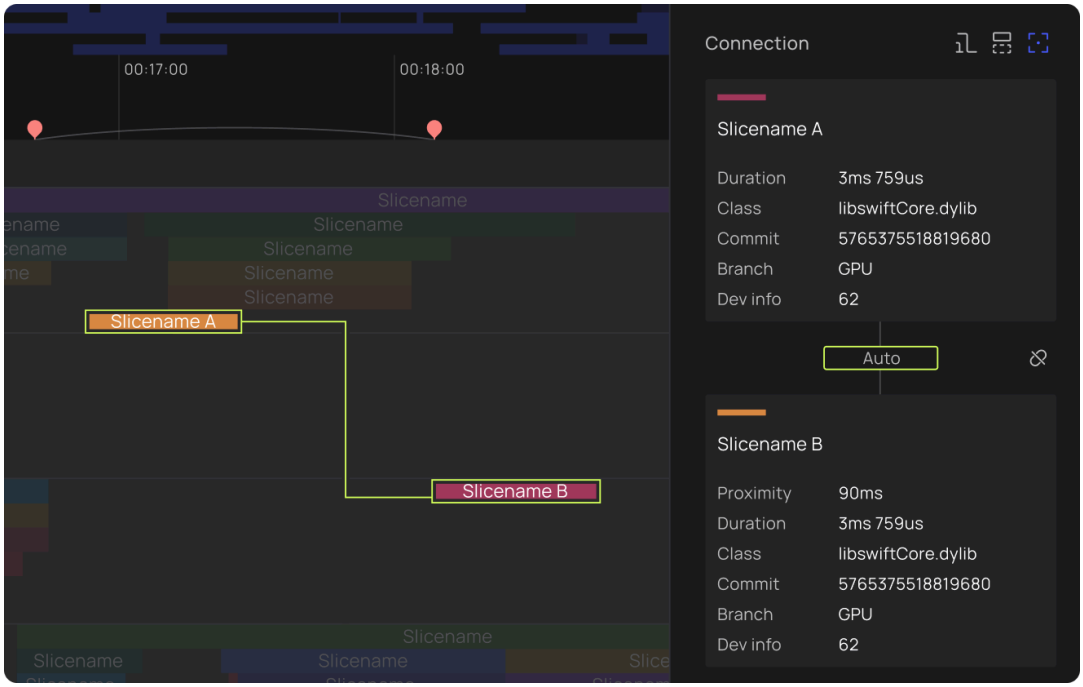
A trace is a time series snapshot of your app. Our dynamic, multi-threaded tracing tool captures and measures all functions executed during the recording. While millions of events and functions might be executed on the system level, our AI filters key functions that impact your user experience the most while filtering out all irrelevant information.

A slice represents a single unit of work done by the CPU, and the width (X-Axis) of the slice represents the duration of the executed process and its position indicates the start time.

The nesting (Y-Axis) of slices represents the call stack of a specific function.
In the image below, you can see how function B (Slice B) was called by function A (slice A)

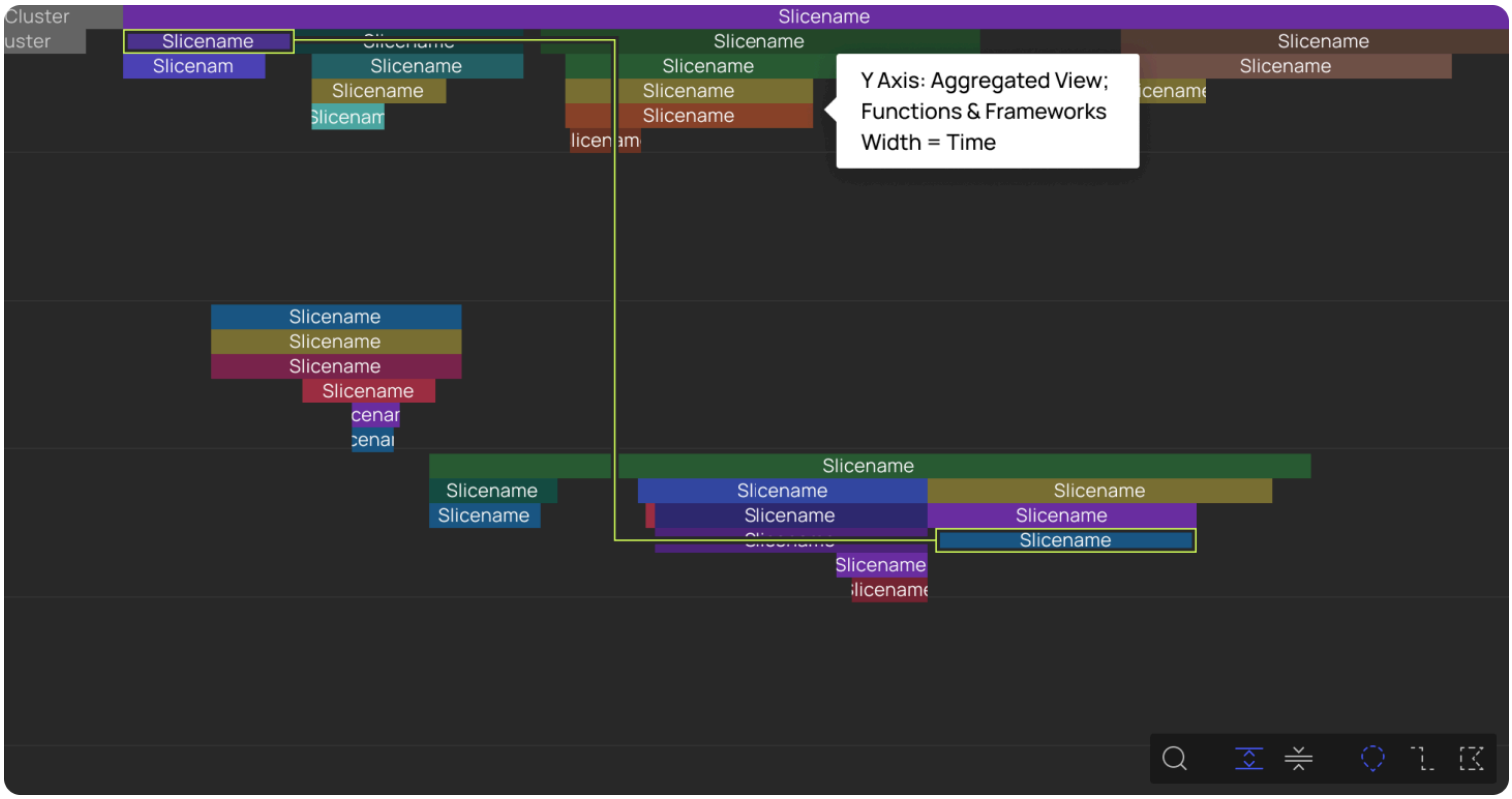


The execution path is the most helpful tool to determine where the delays are coming from. It shows which function call stacks are called by which function call stacks, so you can quickly see not just how individual functions are related but by how groupings of functions are related, creating an elevated perspective of the code which makes it easier to zoom-out and quickly find the root cause of a problem.



Navigate main timeline

The center of the dashboard is what we call *the main timeline* of your user flow. It tells you about your different processes at a specific time. Though you won't be able to view the details of slices here, you can see when your app is fully loaded and when it's idling.



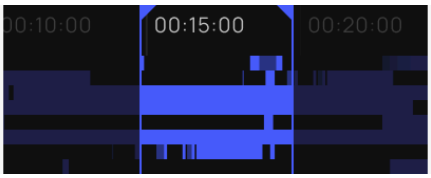
To navigate the main timeline view, in addition to your trackpad, you can use the following key commands: :
W - Zoom in

S - Zoom out
A - Move left
D - Move right

Navigate aggregation view (global timeline)

In the timeline near the top of the dashboard, you'll see what looks like a measuring tape.

- Numbers at the top indicate where you are based on the trace's timeline.
- Numbers at the bottom measure the distance between functions on the trace or the time it takes for a function to execute.



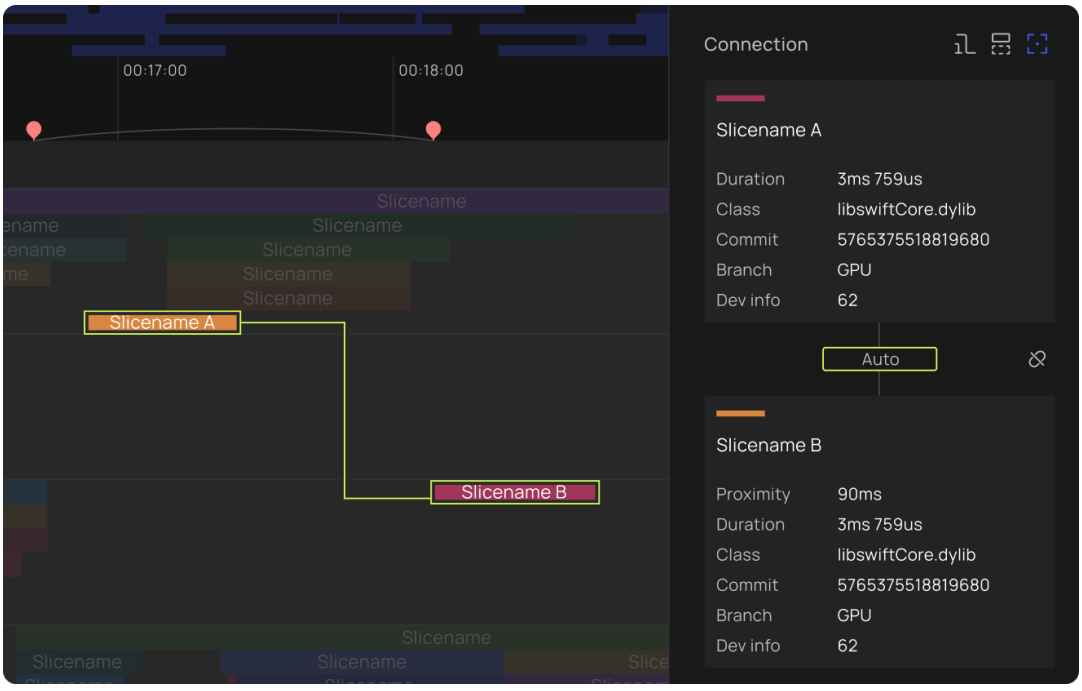
You can adjust the focus area by dragging the blue limit handles located on the edges.

Slices

The visual representation of a code's function/process. The length of the slice indicates how long it takes that process to execute.

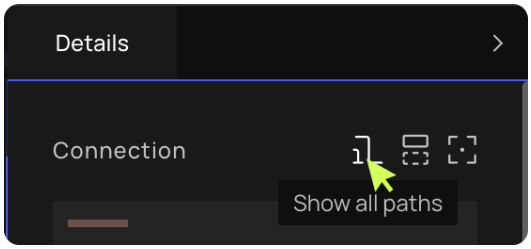
The width of the slice represents the duration of the executed process and its position indicates the start time.

Property Panel

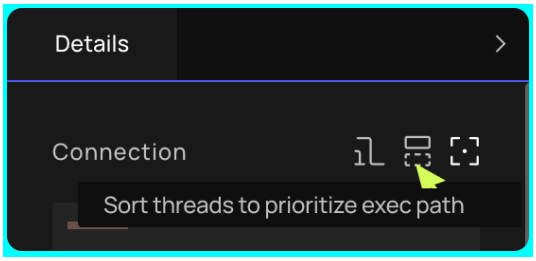


The property panel allows you and your contributors to:

- View and copy details of the slices including slice name, thread name, slice ID, object ID, start time, and duration.
- Create manual connections
- Find execution path-related tools such as:
    - Show/hide all paths



- Sort threads by execution path

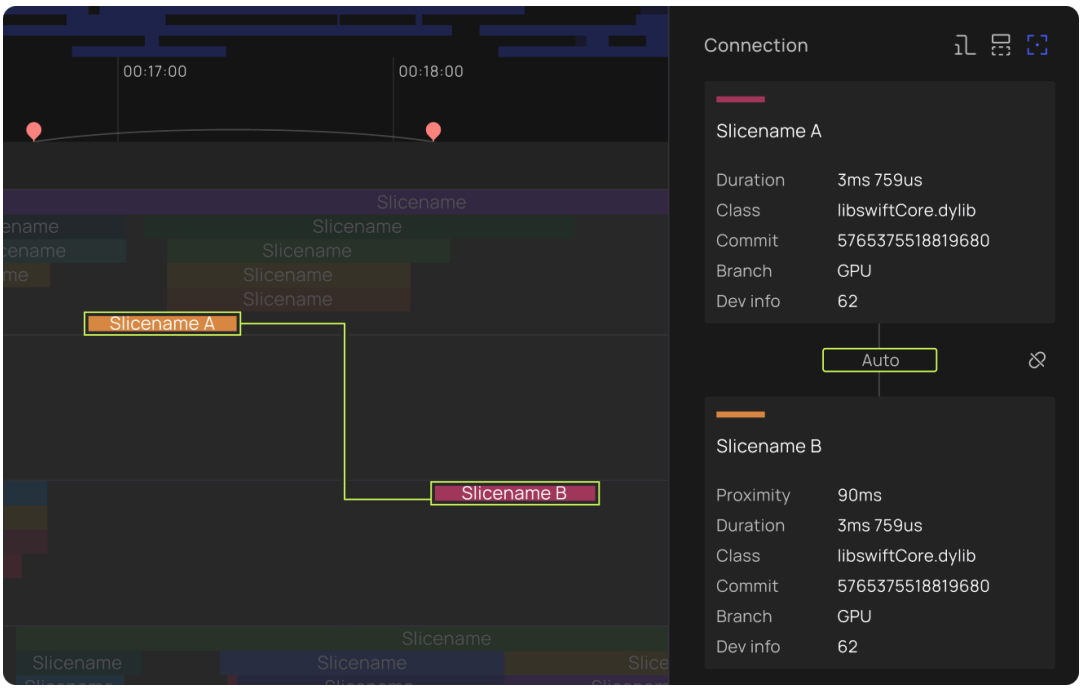


- Dim slices outside of execution path

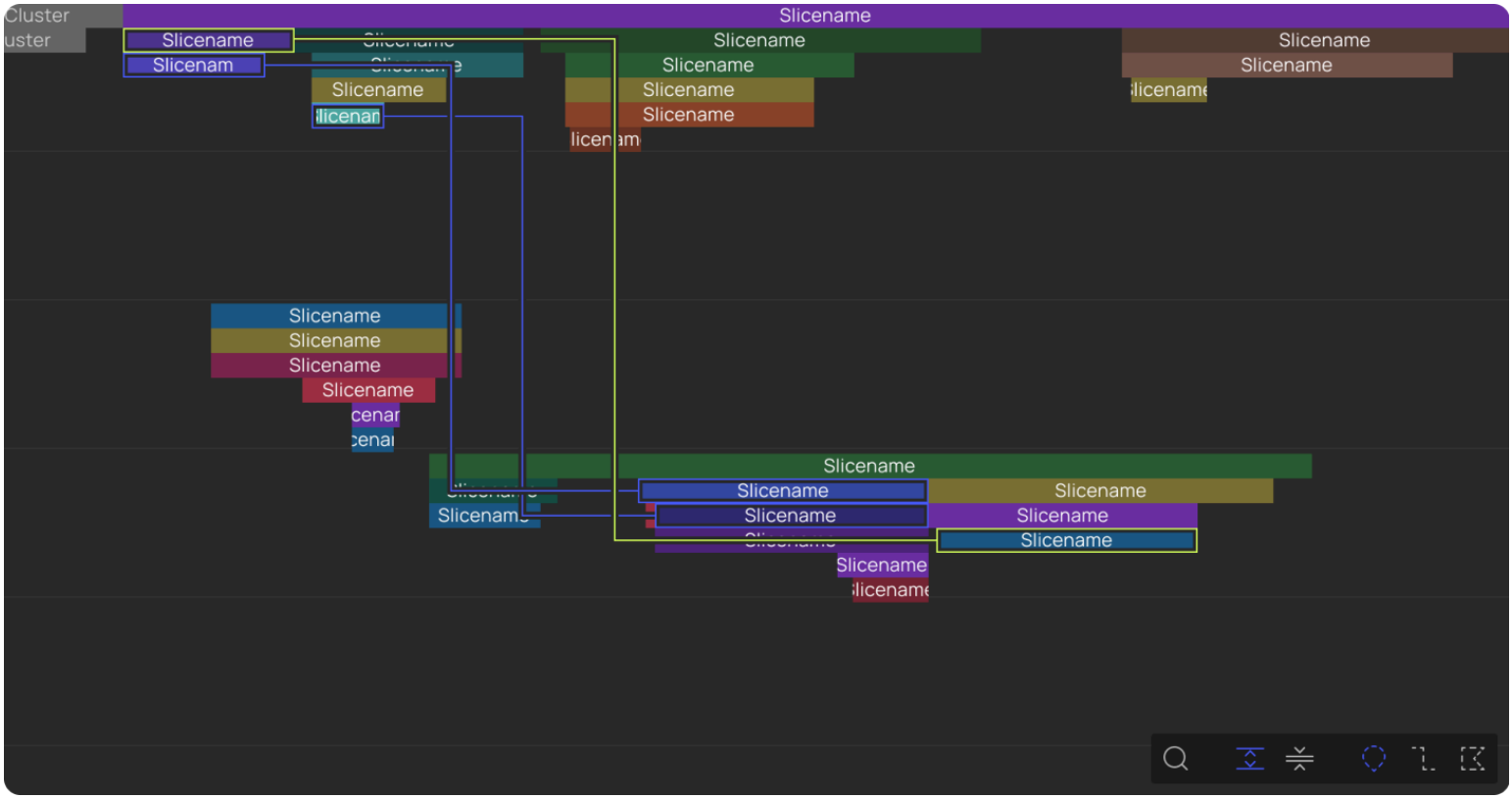
- Review connections
- Customize flags

Execution Paths

1. Click on any slice and our system will automatically check any connections linked to this slice. In other words, we highlight actions and processes that occurred resulting in the slice you chose.
2. Once an execution path – a one-directional linked list of slices – is found, you should see lines like the example below. For example, if Slice A calls Slice B, they will be connected.

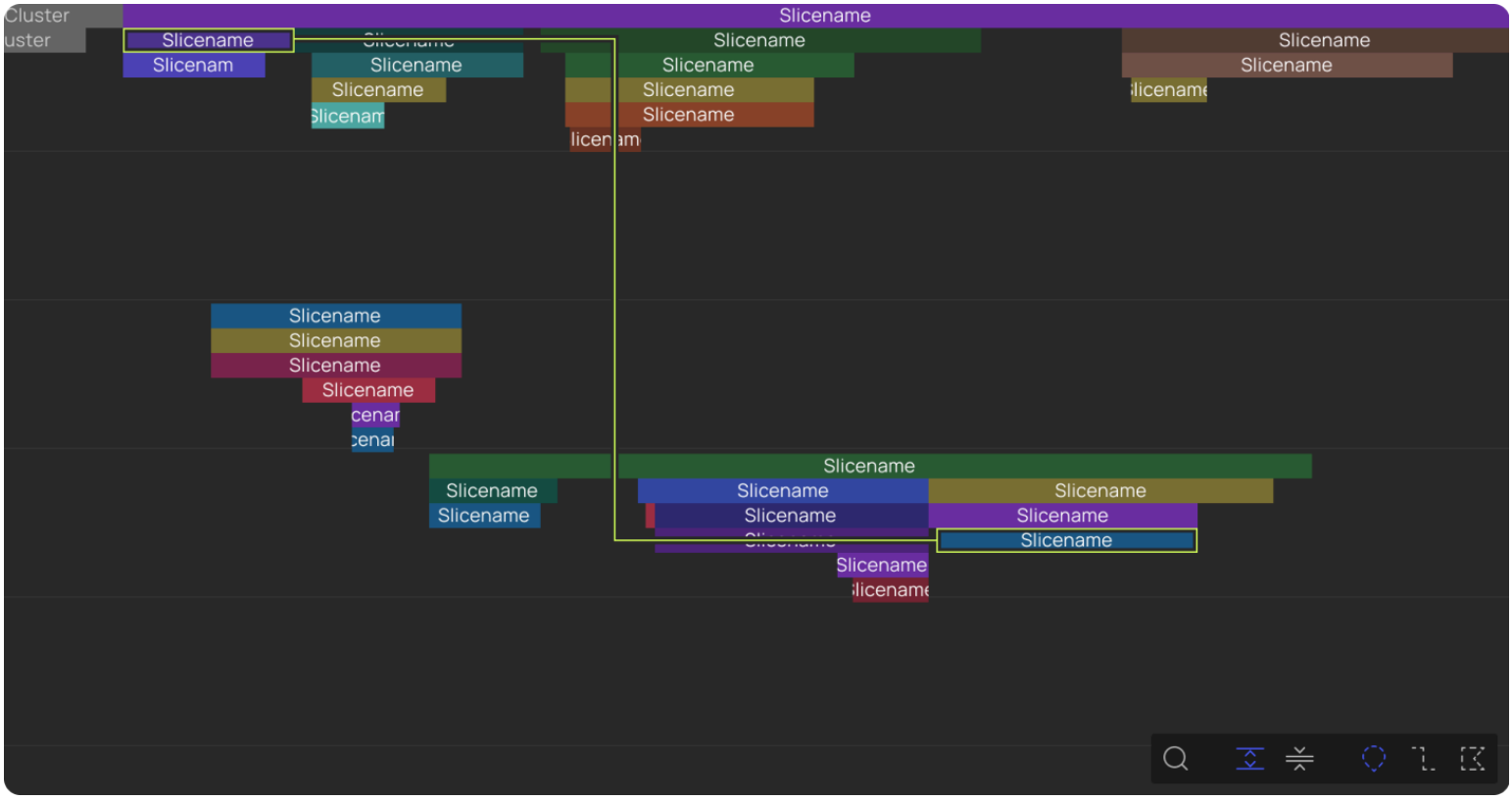


The green path is what we refer to as the **main** execution path; it usually contains more slices than the other lines and gives you a better image of where the delays are located.

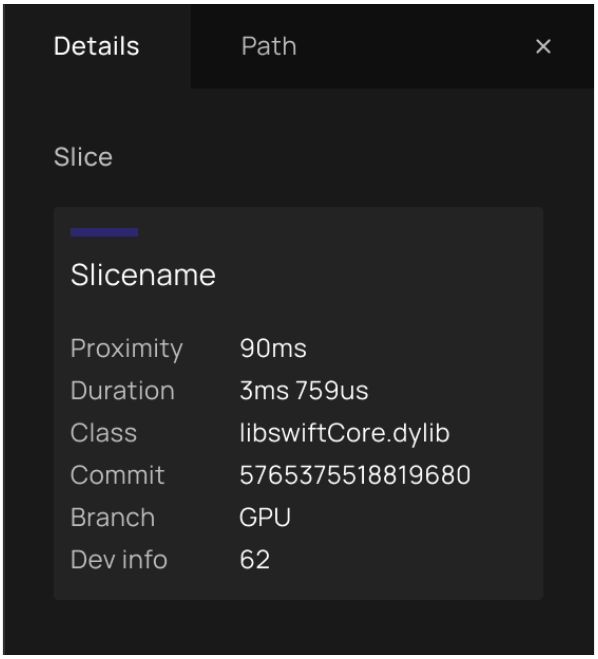


3. Open property panel
4. Click the button below to hide/show other non-main execution paths

Example of clicking the filtering button to only show the main execution path

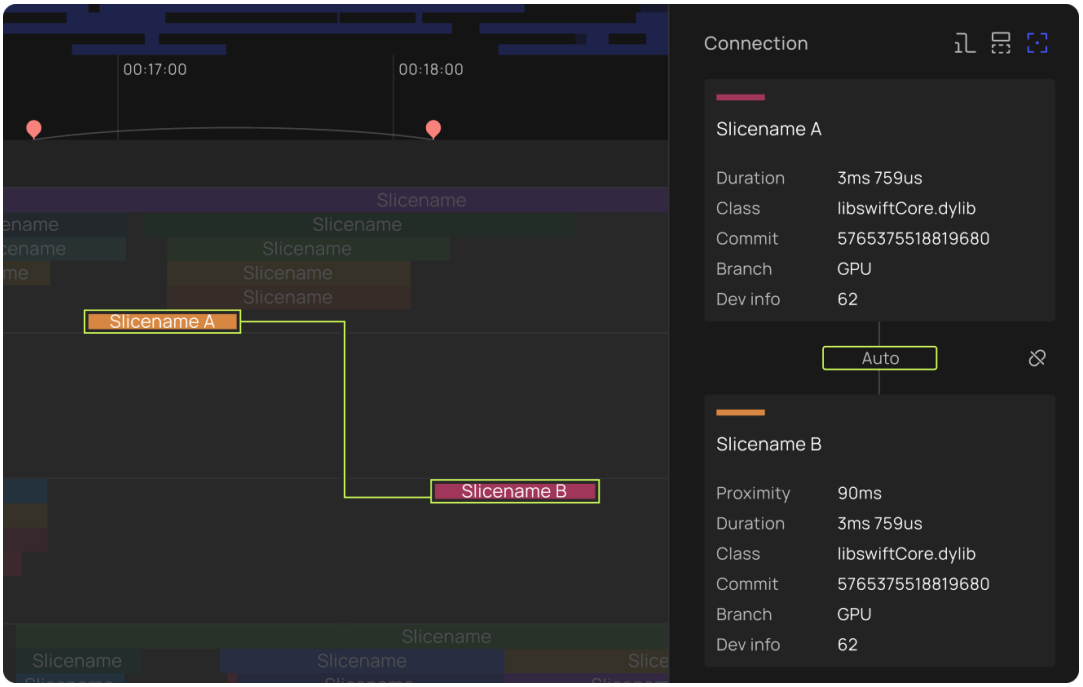


5. View slice and connection details on the property panel (right side)



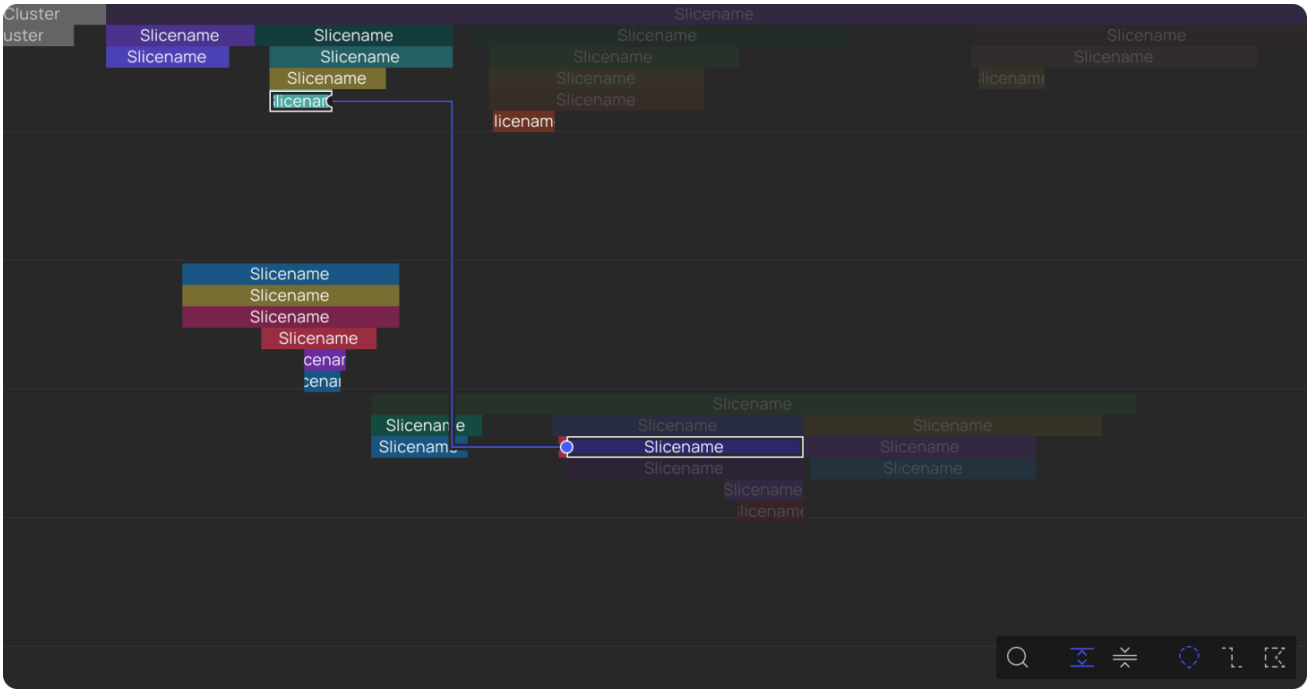
Create manual connections

1. Click on a slice > Slice details show in the property panel.



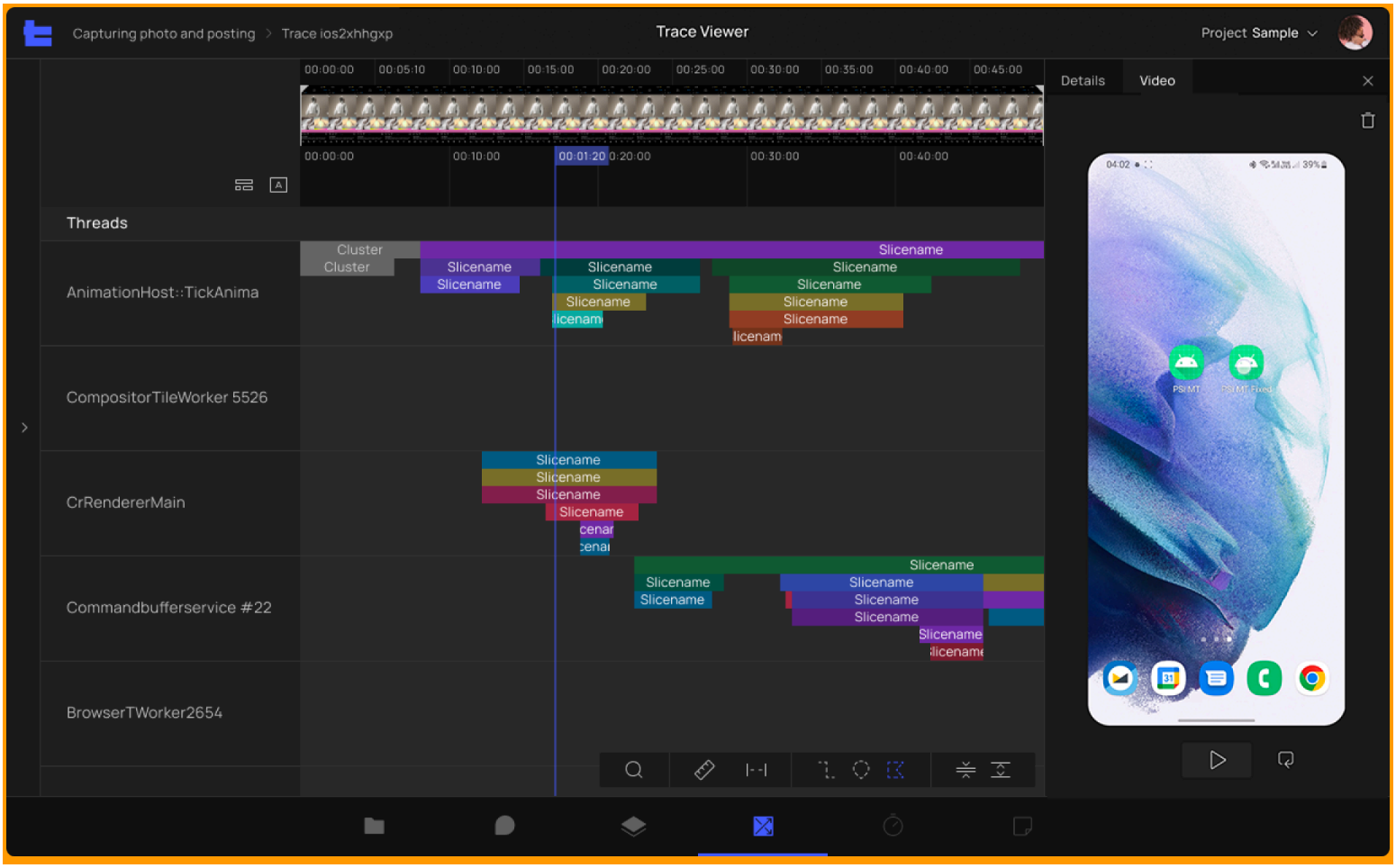
2. Click the ‘create connection’ icon next to the slice or press C while the slice is selected.

3. Slices that cannot be connected to the slice clicked are dimmed.



4. Repeat step 2 to connect more slices.

Working with Video



In the Trace Viewer, you can find the **Video** panel next to the **Details** panel where you can view, upload, play, pause and remove screen recording of your traces.

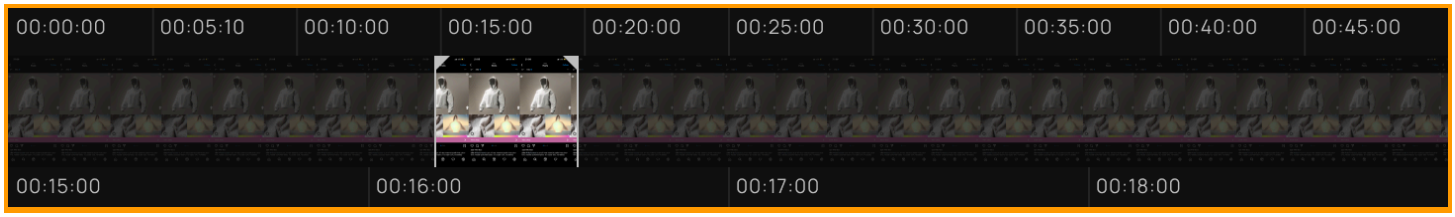
Switch between Aggregated and Video views

Option 1: Use the View Toggle [Soon to be released]

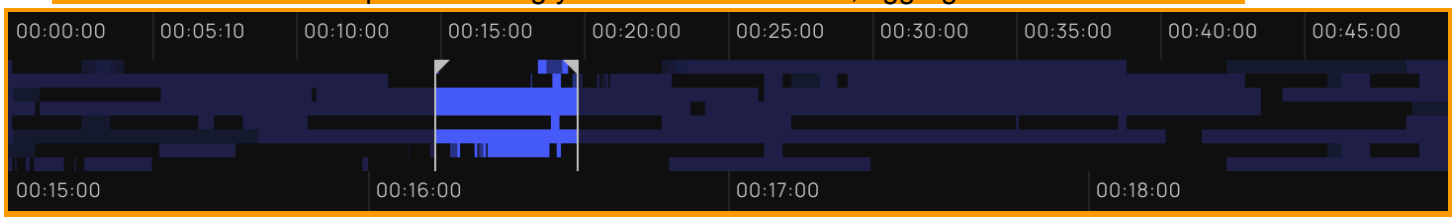
- Trace Viewer > Left of the global / aggregated view timeline;
- Click on the  toggle to switch between aggregated or video views.

Option 2: Switch between Panels

- Click on the **Video** panel > Bring you to the video view where the video frames replace the global timeline.



- Click on the **Details** panel > Bring you back to the default, aggregated view of the trace.

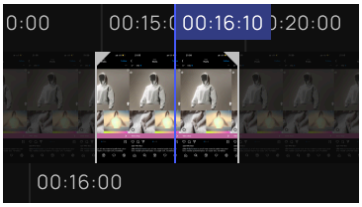


Set Focus Area

On Video Panel (Not for this released)

Trace Viewer > Video panel > Bottom of the video;

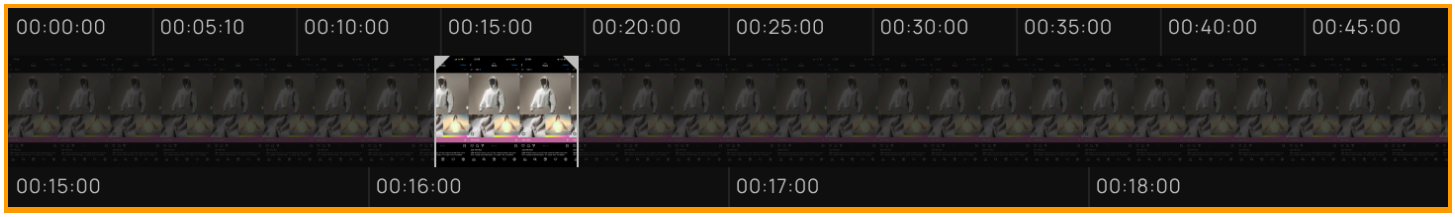
- Shift-drag the beginning or end of the focus area - it will adjust the beginning and the end of what you see on the main timeline.



- The focus area selected will automatically sync with the global timeline's.

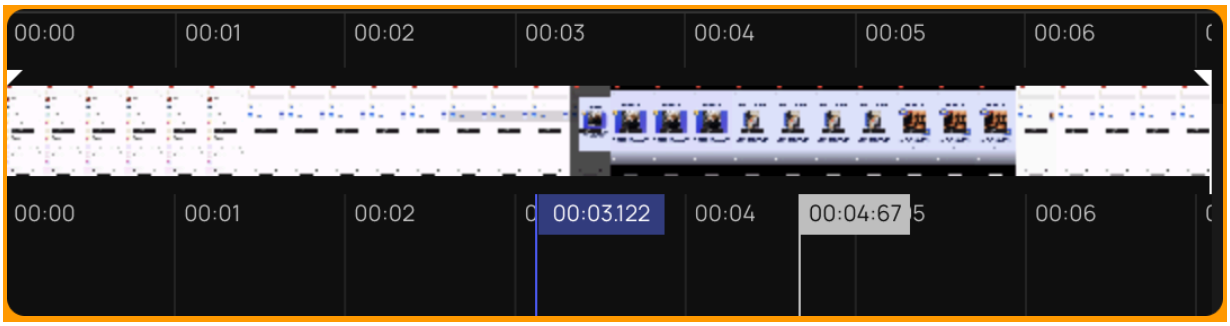
On the Main Timeline

- Trace Viewer > Enable video view.
- Shift-drag the beginning or end of the focus area - it will adjust the beginning and the end of what you see on the main timeline.



- The focus area selected will automatically sync with the Video panel.

Current Time Indicator



Current Time Indicator (the purple flag) highlights the current position of the screen recording video on the trace timeline. It is like a vertical ruler that enables you to see how all the threads and slices the current time indicator touches are associated with the video frames shown on the video panel.

Previewer

Hover the top part of the global timeline and you will see the Current Time Indicator Previewer (white flag) appears. Move the previewer along the timeline and you can preview the video on the video panel.

If you click while you're in preview mode - the current time indicator (purple flag) will move to the clicked position.

If you move your cursor out of the global timeline, the video preview in the video panel changes to the current-time indicator position.

Important Tips

- You can only set current time indicators within the focus area limits.

Play/ Pause Video

- Open the **Video** panel.
- Press ▷ at the bottom of the video.
- Screen recording and the current time indicator on the main timeline play in a loop within the limits of the focus area.
- Press ❙❙ at the bottom of the video to pause both the video and current time indicator.

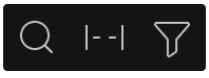
Loop Video

To make the video repeat continuously

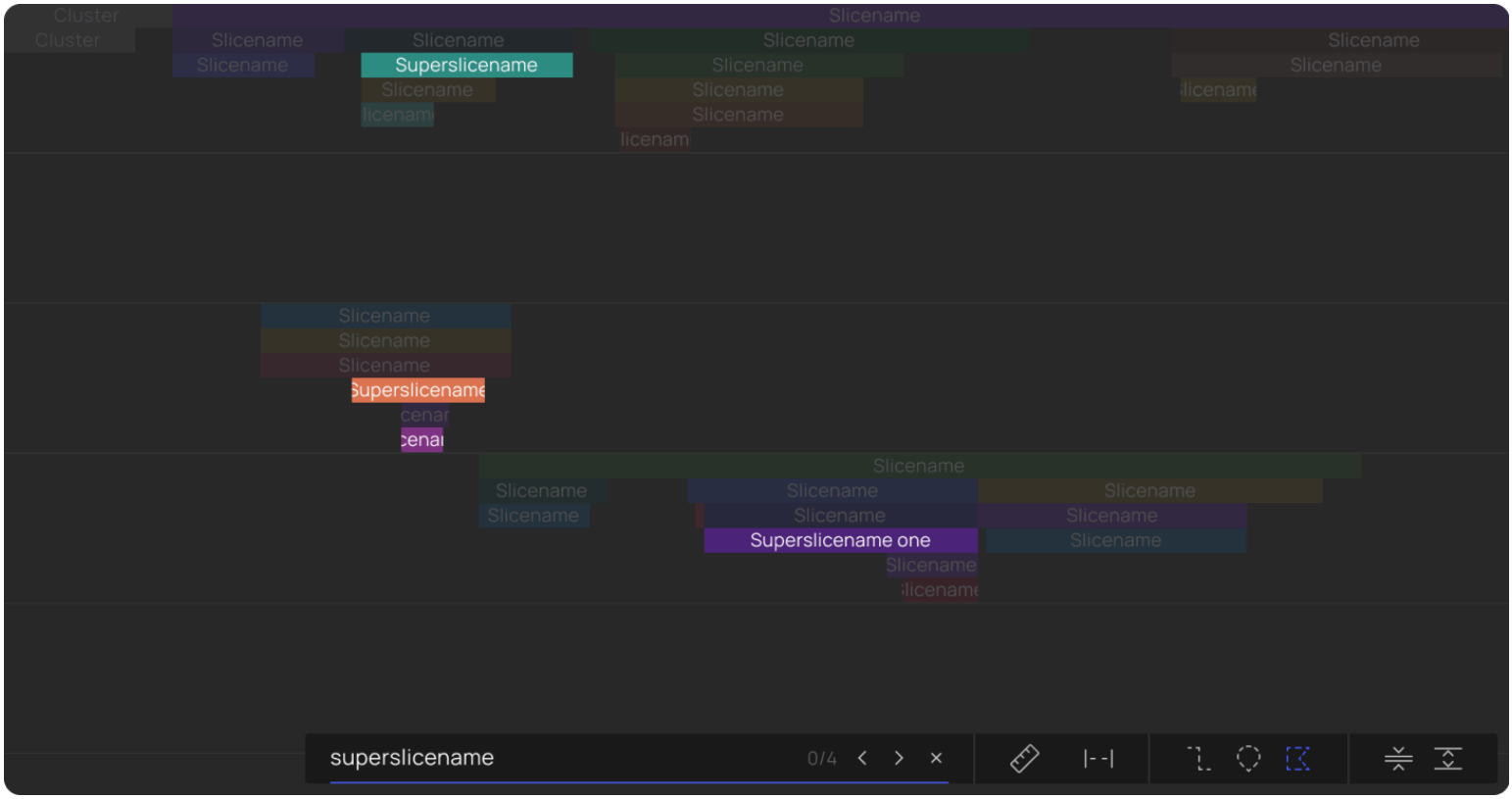
- For the video to loop from beginning to the end of the video - press .
- For the video to loop within the focus area - press the same button that has now turned blue.
- To turn the loop feature off - press the again.

Use Search

1. Search for any framework and function using the search box at the bottom right or press ctrl+f for Windows and cmd+f for Mac OS.



2. All related instances should be immediately highlighted (while all the other unrelated functions should be dimmed).



3. Use arrows inside the search box to navigate among the search results.

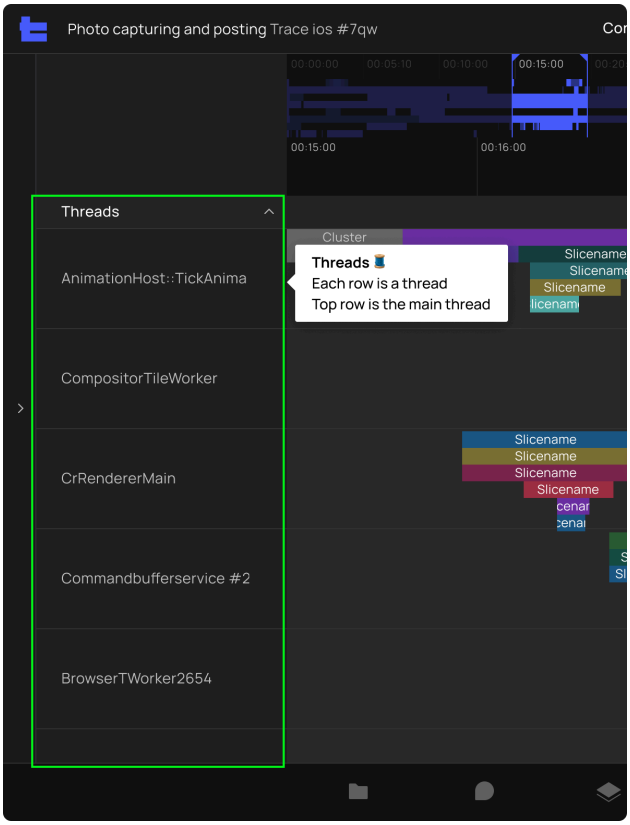
4. Press Esc or click ‘x’ to exit search.

Measure Time

To understand the duration of one function to another, you can use our measurement tool by holding **shift + click**.

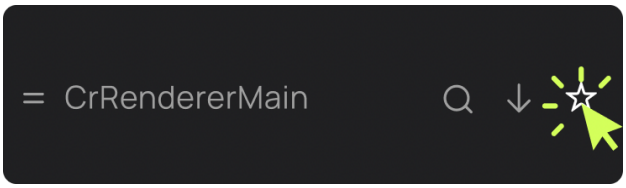
Threads

Smallest executable task of the system. PS multi-threaded code profiling tool highlights critical functions and frameworks that impact user experience, revealing the root cause of problems;



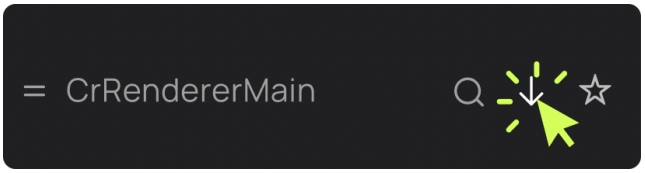
Pin Threads

Click the star to the right of a thread name to pin the thread to the top of the window for easy comparison and reference.



Deprioritize Threads

Click the downward button to the right of a thread name to pin the thread at the bottom of the window if you find that particular thread is less relevant.

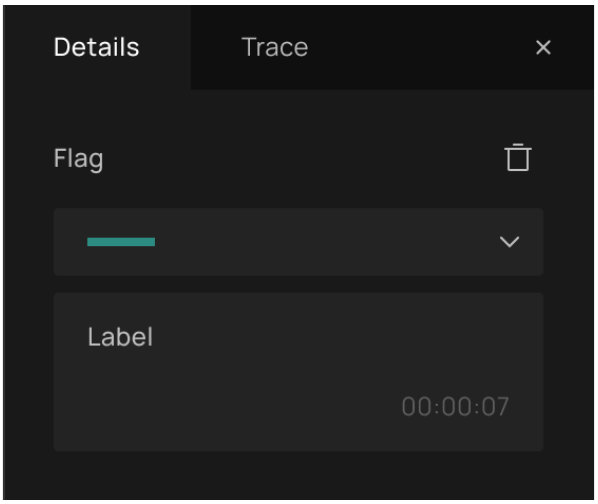


Place a flag

We recommend placing flags at the beginning and the end of a flow. You also get to choose colors of placed flags, which is helpful when you need to keep track of different information.

1. Hover over the timeline, and you'll see a gray flag.

2. Click on the timeline to place the flag.
3. Our tool will randomly pick a color for your flag, which you can manually edit on the right-hand side of the screen. This is also where you can add labels to your flag.

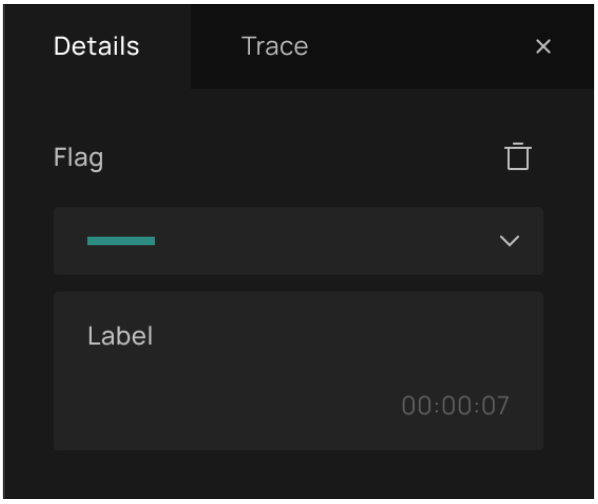


Pro tip: You can also use the label field for quick notes. Simply click on the “label”, press shift + enter and start typing. That way, the notes you typed won’t show in the timeline, but you can use the notes for future reference.

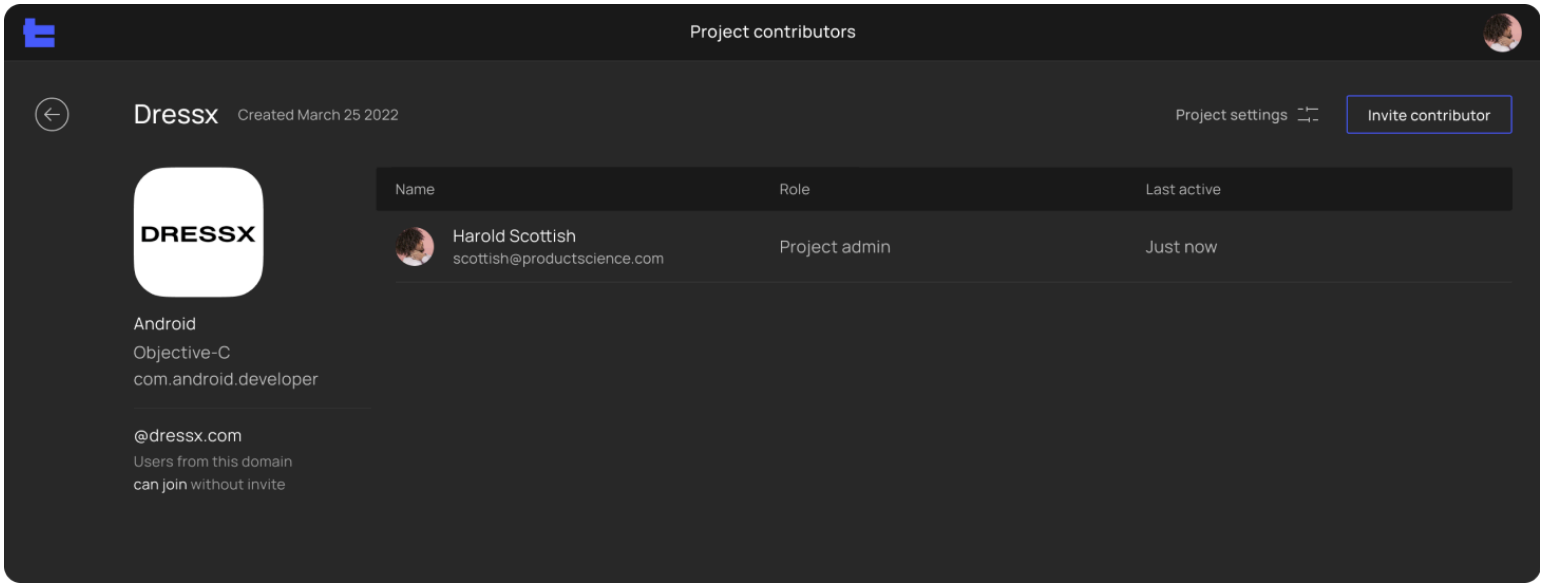
Delete a Flag

When a flag is no longer relevant, you can remove it by:

1. Clicking on the flag
2. Clicking on the ‘trash can’ button when the flag details menu is shown on the right; or pressing “Del” on your keyboard.



Admin Screen



Team

Team is the representation of a company. This is where you can create projects (per single app and operation system) that contain all flows and traces.

> Learn more about what you can do as a Team Admin.

Project

Project is where you can create different flows and traces for a single app. You can join and contribute via an invitation from the team's or project's admin.

> Learn more about what you can do as a Project Admin.

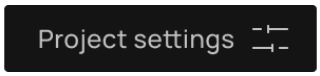
Collaborating

Roles and Permissions
Every team member can have team-level permissions that determine their default access to projects.

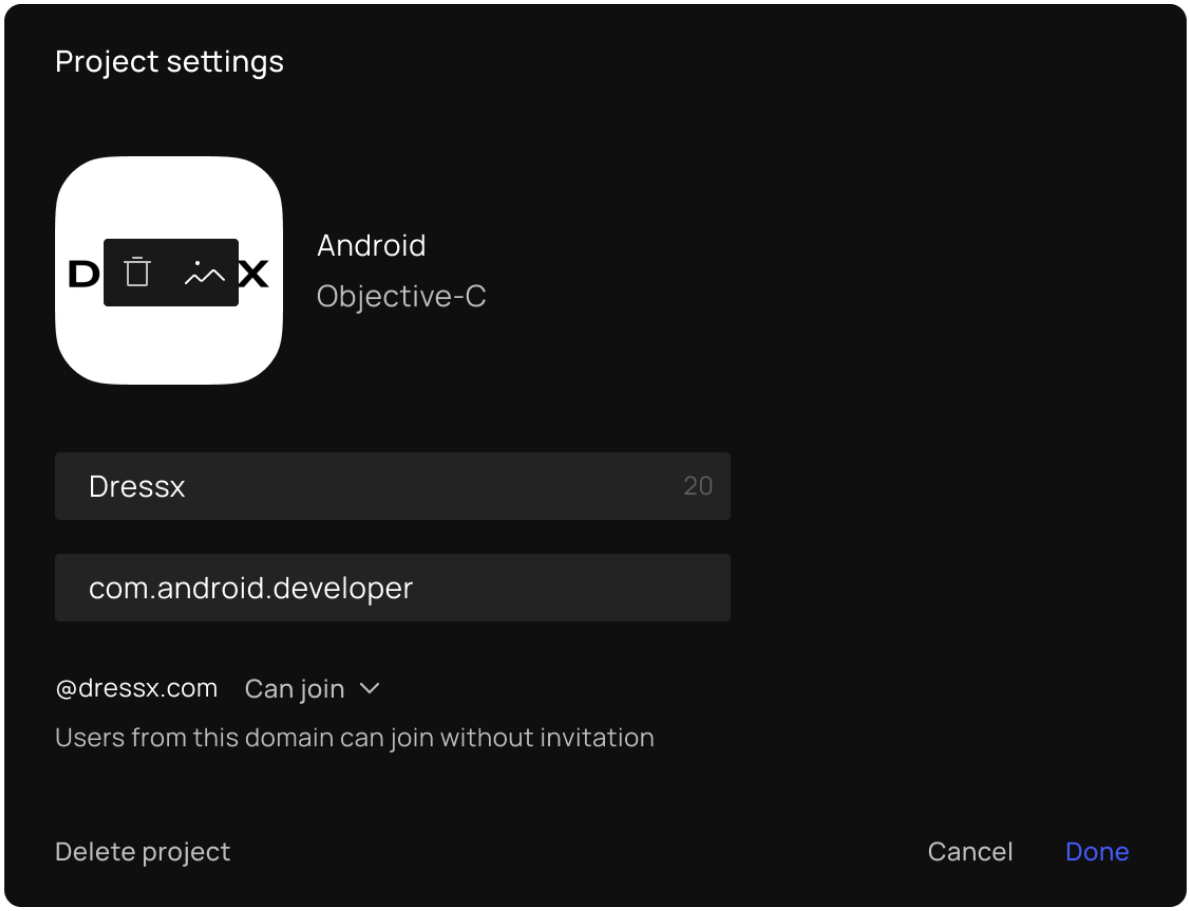
Team Admin

Update Project Icons and Details

1. In Project > click 'project settings'.

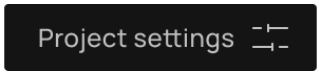


2. Hover over the thumbnail > click on the image icon.

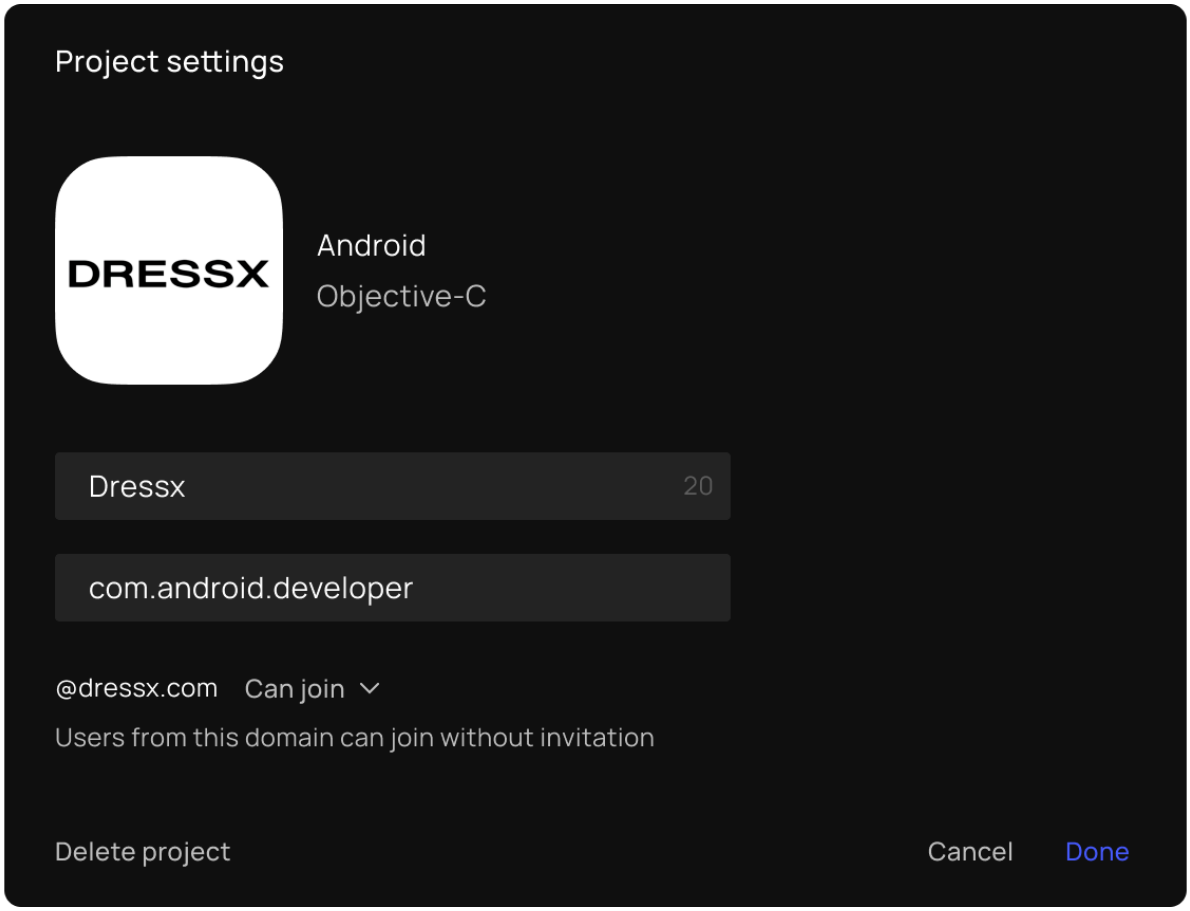


Remove a Project

1. In Project > click ‘project settings’.

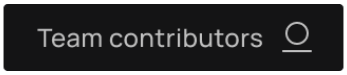


2. Click ‘delete project’.

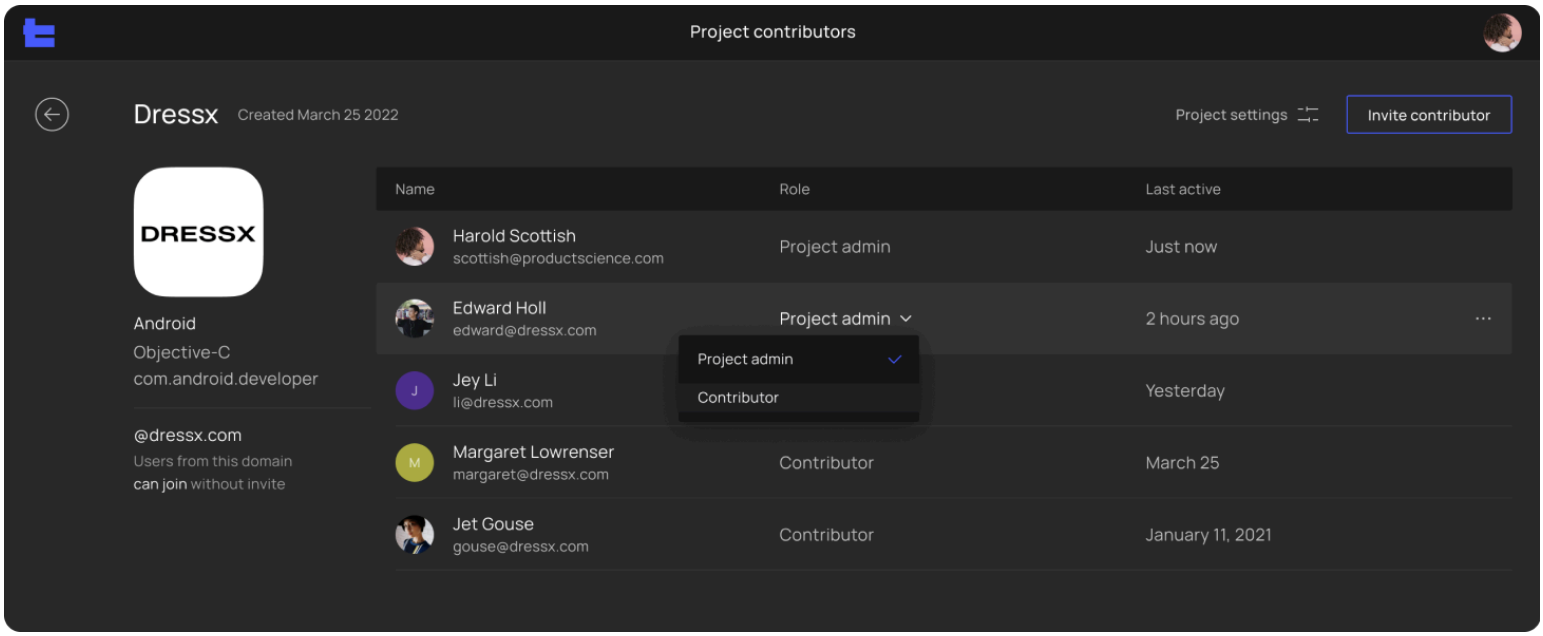


Update Team Member's Role

1. In Team > click on 'team contributors' to view the table.



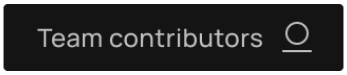
2. Click and choose a new role under the 'Role' Column.



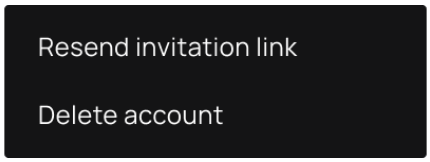
3. Click out to save.

Resend Invitation Link

1. In Team> Team contributors > click on 'team contributors' to view the table.

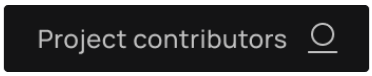


2. Click the '...' button on the right of the table > menu appears.
3. Click 'resend invitation link'.

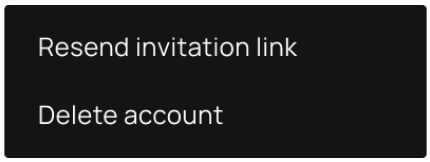


Remove a Team Member

1. Project screen > click on 'project contributors' to view the table.

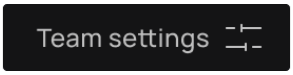


2. Click the '...' button on the right side of the table > menu will appear.
3. Click 'delete account'.

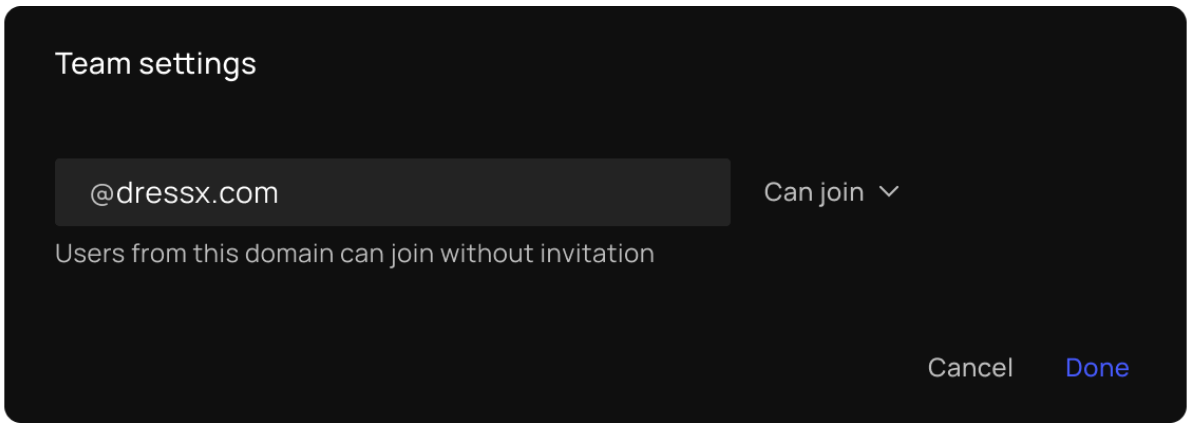


Update Email Domain

1. Home page > click 'team settings'.



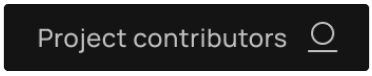
2. Update your company email domain so that everyone with the specified email domain can log in.



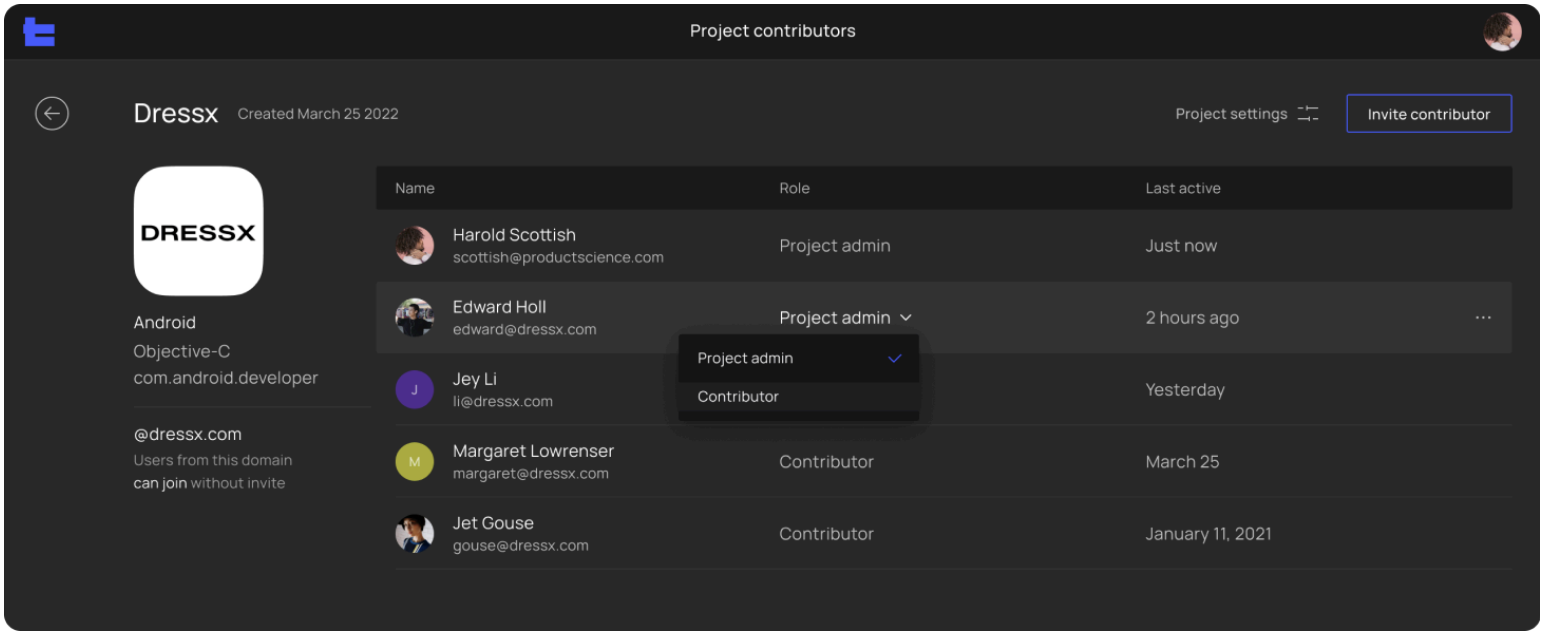
Project Admin

Update Team Member's Role in Project

1. In Project > click on 'project contributors' to view the table.



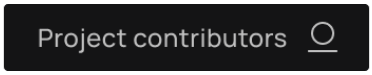
2. Click and choose a new role under the 'Role' Column.



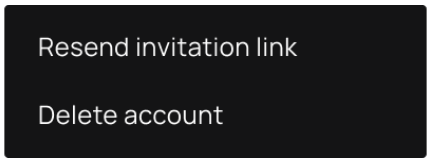
3. Click out to save

Remove Team Member

1. In Project > Project contributor > click on ‘project contributors’ to view the table.



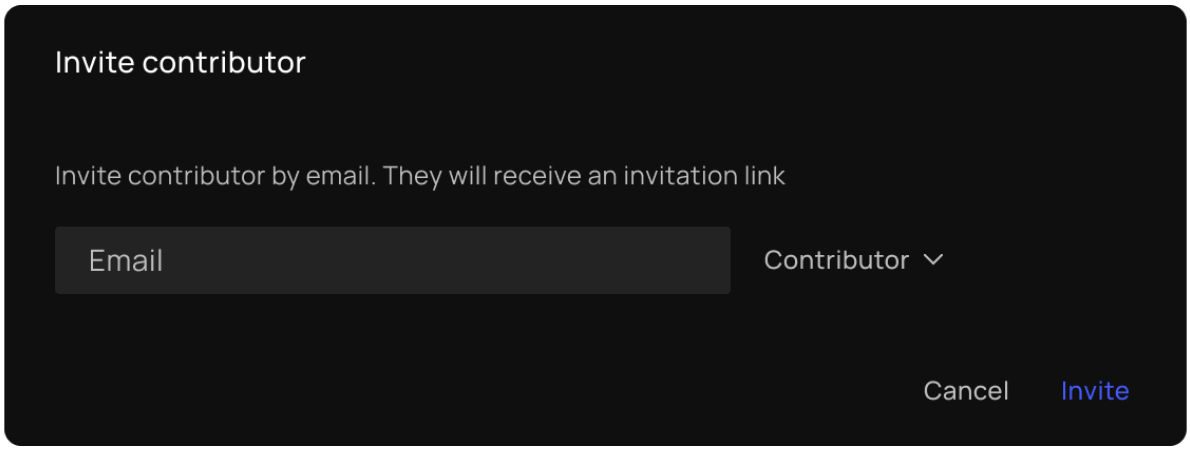
2. Click the ‘...’ button on the rightest of the table > menu appears
3. Click ‘delete account’.



Invite New Team Member

1. In Project > Project contributor > click ‘invite contributor’ on the top right corner.

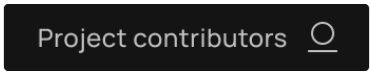
2. Input the email and select the role.



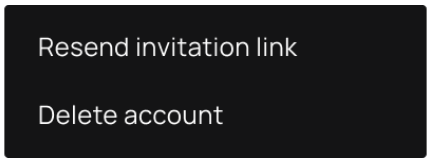
3. Click ‘Invite’.

Resend Invitation Link

1. In Project > Project contributor > click on ‘project contributors’ to view the table.



2. Click the ‘...’ button on the far right side of the table of the table > menu will appear.
3. Click ‘resend invitation link’.



Dictionary

Cold App Start

A cold start refers to an app's starting from scratch: the system's process has not, until this start created the app's process. It happens when your app is being launched for the first time since the device booted, or since the system killed the app.

Sample use case

- Hold your mobile device
- Close all your opened apps
- Click on your app icon from home
- Everything will be initialized from zero

According to Google, the ideal benchmark time should be:

- Cold startup takes 5 seconds or longer.
- Warm startup takes 2 seconds or longer.
- Hot startup takes 1.5 seconds or longer.

Connection

Line connecting two slices showing what previous function triggered the execution of the selected function.

Hot App Start

A hot start refers to when your apps' process is already running in the background and all the system does is bring your activity to the foreground. This process will bring back your app to the last state where you left without reinitializing any of the app assets.

Sample use case

- Hold your mobile device
- Close all your opened apps
- Click on your app icon from home
- Everything will be initialized from zero
    - This was so far for a cold start
- Go back to home
- Re-open the app again
- Your app is now bring back to the foreground without reinitialization

According to Google, the ideal benchmark time should be:

- Cold startup takes 5 seconds or longer.
- Warm startup takes 2 seconds or longer.
- Hot startup takes 1.5 seconds or longer.

Instrumentation

Instrumentation is the method in which our plugin can extract the necessary information about the runtime of

the application, such as, how long functions run, which processes they call (child or async) and which threads they run on. It runs during the build process, instrumenting by injecting 100% of all functions and frameworks automatically without sampling.

Slices

The visual representation of a code's function/process. The length of the slice indicates how long it takes that process to execute.

The width of the slice represents the duration of the executed process and its position indicates the start time.

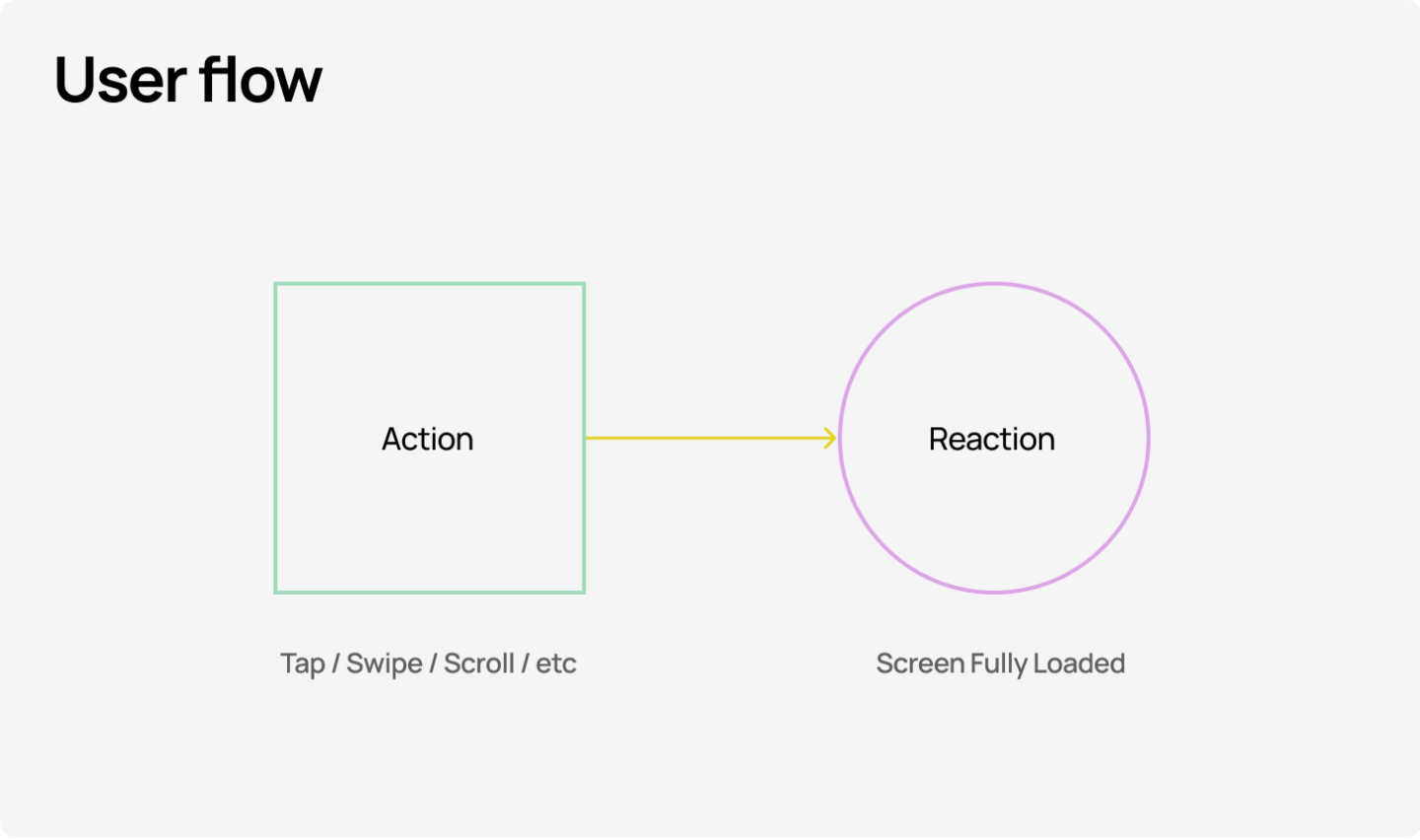
Trace

Think of a trace as the technical representation of a mirror showing us the user's journey. It includes hardware and software advances occurring through an entire app stack.

The trace viewer will show vital information such as what functions were invoked, their duration, and most importantly, their execution paths. The visualization of execution paths is crucial because it allows everyone to see the sequence of functions and how they connect and highlight optimization opportunities.

(User) Flow

Flow is the path a user takes to complete a task within the app. For example, the user types in "restaurant near me". The app then returns search results. Finally, the user viewing searched results is considered a flow.



[WIP] NOT TO BE LAUNCHED FOR 10/07/22

[WIP] Performance Engineering Methodology

The PSi Tool enables performance analysis that was previously impossible and enables a new methodology for performance engineering.

Performance Engineering Methodology

Before tackling a performance need, a good practice is to approach the task with a specific goal. Here is a list of examples:

- There is a visually obvious performance problem and there is a need to identify the code behavior causing it.
- A change has been made to the codebase and there is a need to investigate its impact on performance.
- An understanding needs to be developed about how the code design impacts performance and interacts with the phone.
- There is a suspect area of the codebase that may be impacting performance
- A developer that is onboarding needs to have a better understanding of how the codebase works.
- A devops alert, devops report, or customer support issue regarding performance has arisen regarding a user flow and an investigation must take place.

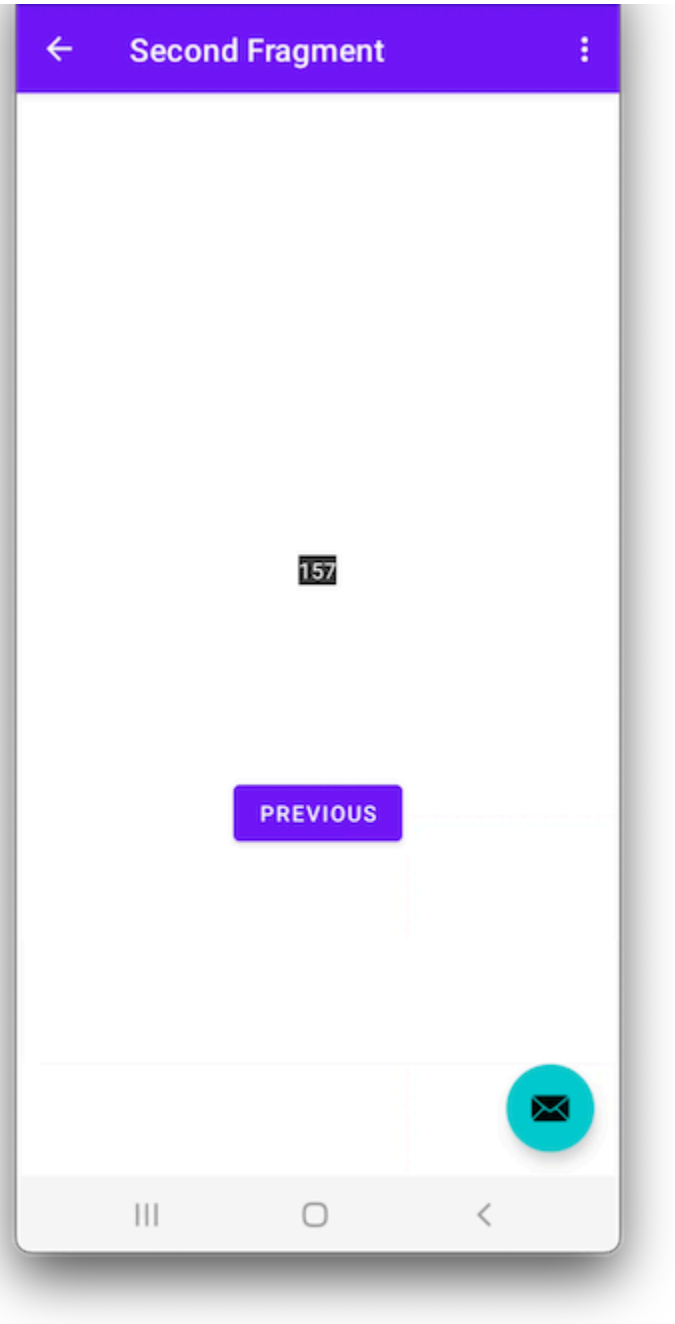
PSi instrumentation creates a frame counter (Android) or time code (iOS) that displays on an app to use for correlation with the trace timeline. This timing is related to when the app starts, not when the video is recorded or tracing starts.

Frame numbers are written also as traces so that you can search for them and correlate something seen while using the app to associated functions/traces.

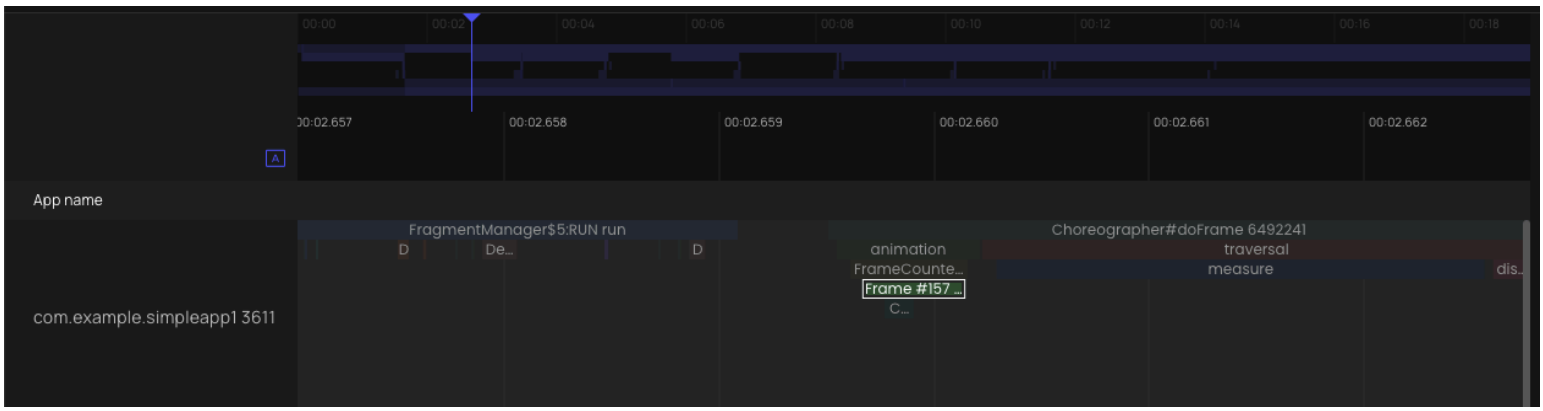
Frame Counter / Time Code Example

The example below shows an app with a lag between the NEXT and PREVIOUS buttons.

When playing the video of the recording, the lag ends at Frame #157 after clicking NEXT.

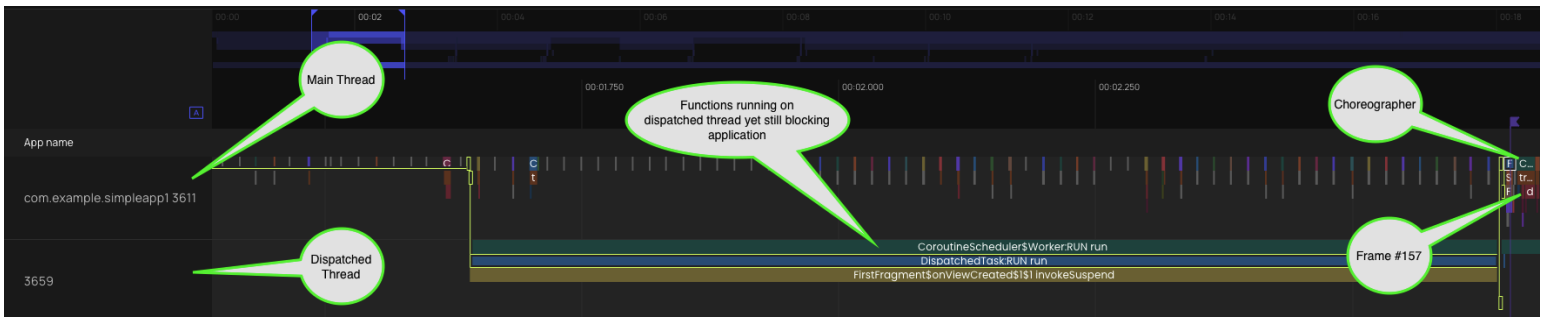


Zooming in around FRAME #157 shows a marker with the frame number right beneath the Choreographer function and its enclosed operations- highlighted function is the marker for Frame #157.



Clicking on the Fragment function left of Choreographer shows the user flow path of function connections that were executed up until this point via the yellow line aka “the path”- green arrow shows the Choreographer associated with Frame #157:

Zoom out to see the source of the lag:



💡 There are functions running in series blocking the smooth transition from the NEXT button to the PREVIOUS button. The functions are being dispatched to another thread - but are still blocking - so the thread dispatch code needs to be examined.

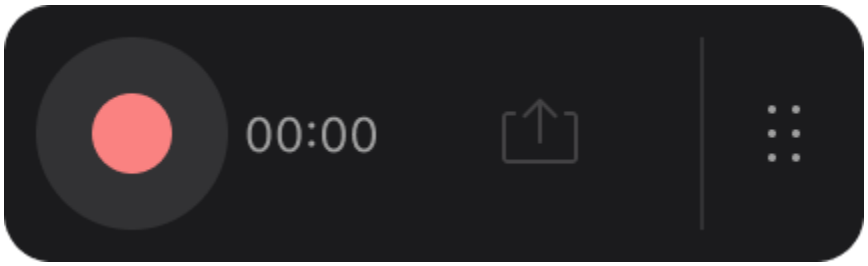
The callouts show the main and dispatched threads and the red area indicating delay. Zooming in will reveal more and code can be examined to find the fix.

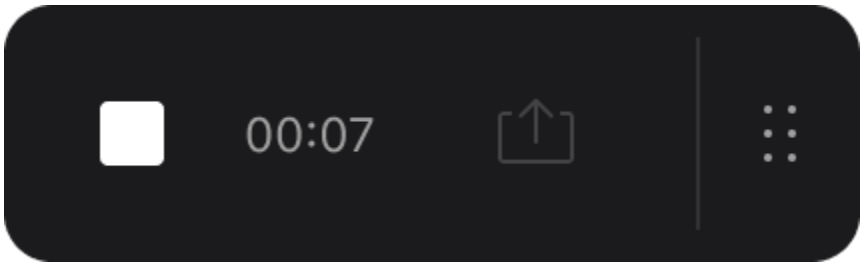
The yellow path shows the connections between functions based on the user flow and the user's actions.

This example including versions with lag and fixed are here: https://github.com/product-science/demoapps

Explain why we need screen recording

[WIP] Shortcuts


00:00


00:07