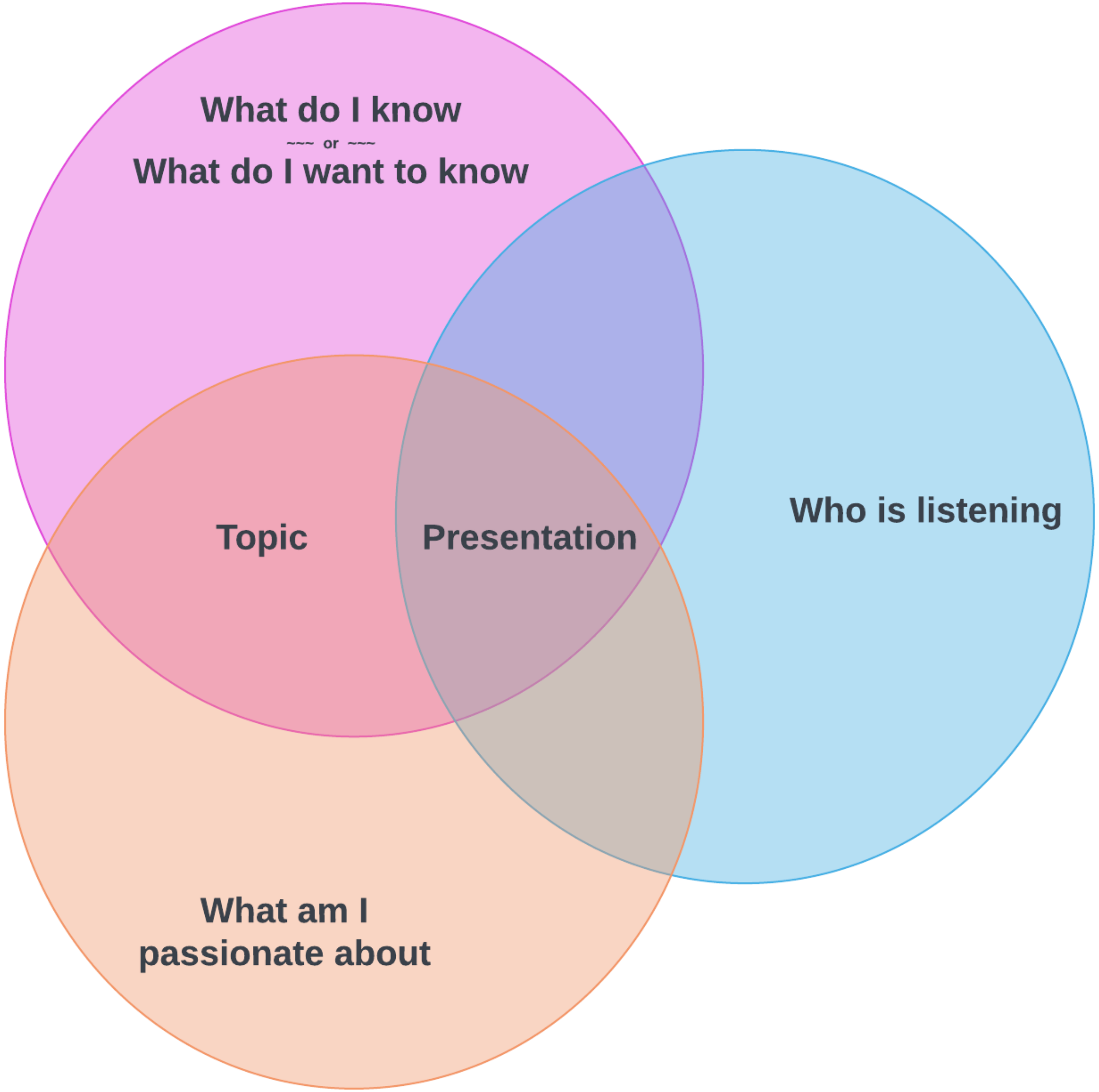
So, you want to give a tech talk...

Let's goooo!

…What do I talk about?



What do I know

~~~ or ~~~

What do I want to know

Topic

Presentation

Who is listening

What am I passionate about
~~~

Topic

What do I Know

- Bourbon
- Physics
- Stats
- APIs
- Node
- Roofing

What Am I passionate about

- Bourbon
- Numbers
- APIs

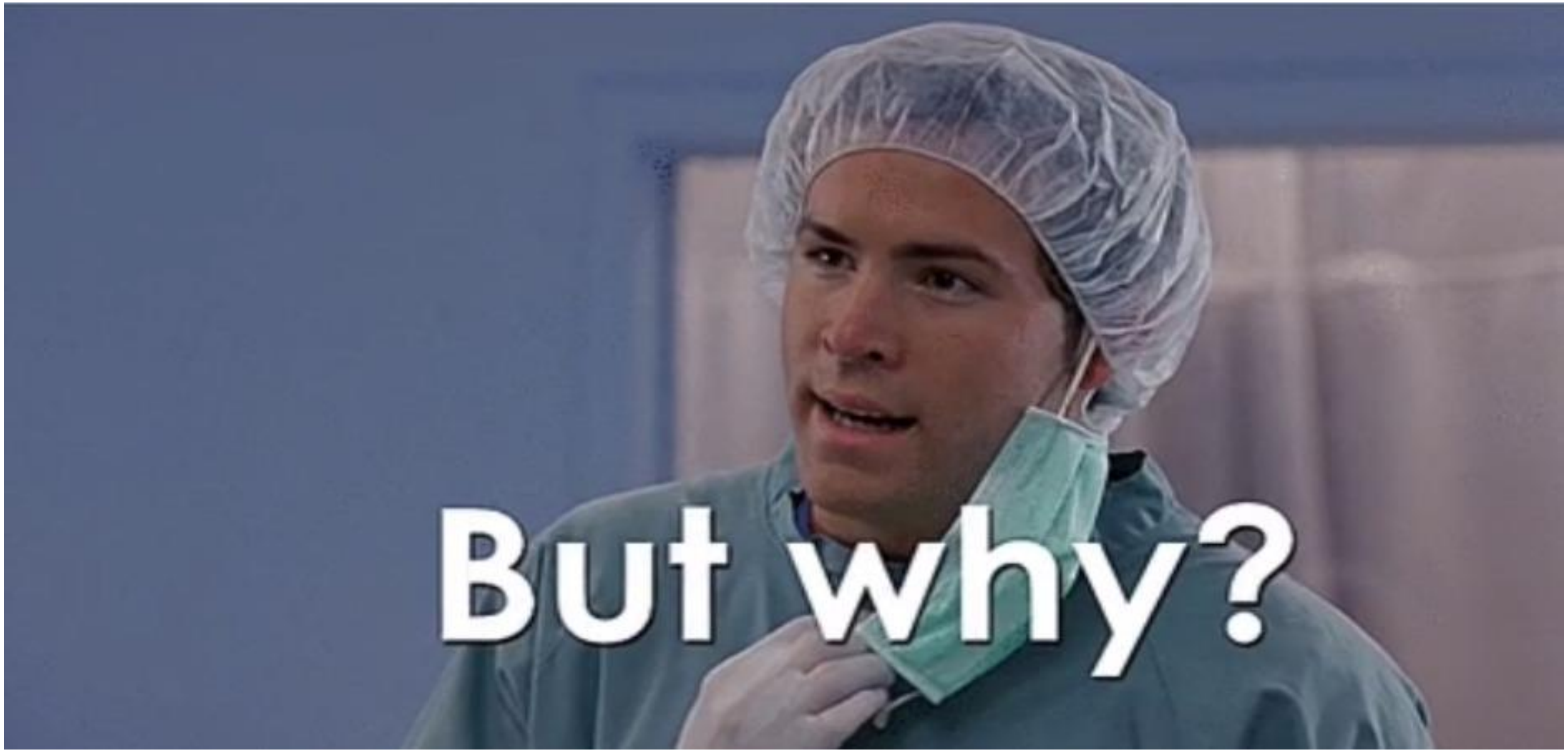
Who is listening

- Developers
- Jeff

API that deals with numbers

Experiments Api
(I happen to be working on this)

Wait a sec...



What problem does this solve?

- Current experimentation involves the developer turning an experiment on/off or updating an experiments scope with a code release. This includes adding drugs to a list, or toggling flipper settings where available. Experiments are done without maintaining a record of what user is seeing what experiment. Logging is done through datadog or other existing systems that are more difficult to tie into analytics.

What you want to say

Get your thoughts out there

What's the goal?

- We set out to replace the functionality of Form_Search_Logs which is designed to help track the user journey through their formpick experience. Specifically, it's being used to keep track of how many alternatives are used.
- So with the formpick rewrite, we needed a way to update the logging aspect (which pendo/datadog/etc were not an option). Since we have to greenfield the app, the question became "How do we abstract this to make it more usable for more people." We set out to solve three main issues:
- - Decouple toggling or modifying an experiment from a code release
- - Log a dynamic object defined by the experiment
- - Make the end user have a consistent experience
- So we created a structure that lets the database define an Experiment. The key things it includes are a start on, end on, and logged object pattern. So at the experiment level we can say this functionality turns on Monday at 8AM, and runs until Friday at 5PM. Furthermore, we can say "if we're logging an object for this experiment, it needs to contain X, Y, and Z".
- So form search logs can be recreated as this experiments' logged object, and we can set it up to not have an end date, so we can start an experiment "form search logs" and log to our hearts content. YAY! 2/3rds complete.
- But this is an experiment, how do we know what experience the user should get? I'm glad you asked! We set it up where an experiment can have Experiment groups. Think "A, B, C, …". And each experiment group can have n% of the traffic go to it. So when you ask the api "what group does user X belong to?", if we have a record we'll say "oh that user always gets experience B". But if it's a user's first time, we say "well what experiment group needs volume? that user is now assigned to that group". So at the data level, a user's experience is consistent for the workflow.

How you want to say it

Avoid wall of text

Three main points

- Give a custom logging experience that analytics can tap into
- Decouple experiment modification from developer releases
- Lock down user experience when experimenting

How you want the listener to digest it

Less is more

Custom Logs

Experiment maintenance without release

Consistent End User Experience

How to present it

Coherent thoughts is hard

LIVE DEMO TIME!

That didn't quite go as planned...

Live Demo

- High Risk
- Live Demo vs Live Debugging
- Can show "ease of use"

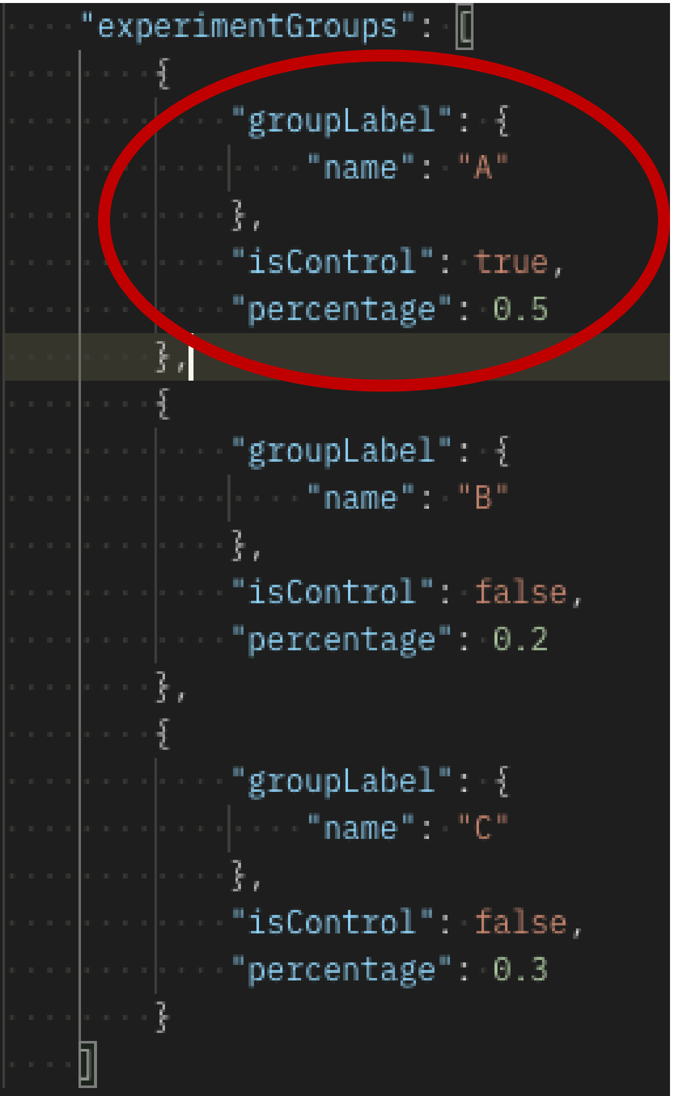
Pre-set up

- Pictures examples ready to go don't fail
- Videos can curate expected outcomes

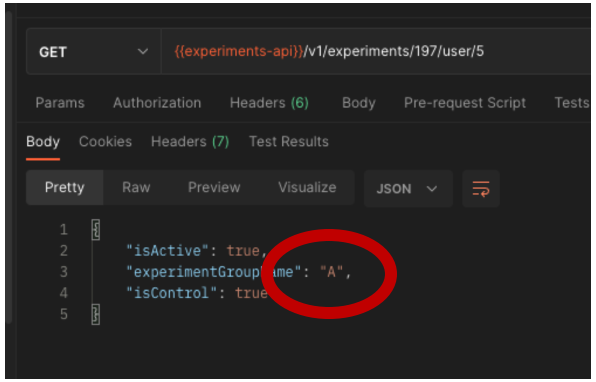
Time to Refine

Iteration zero is going to suck

Here's One I prepared in advance!

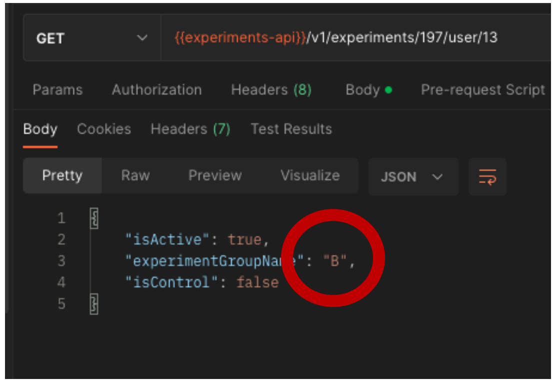


```
"experimentGroups": [
    {
        "groupLabel": {
            "name": "A"
        },
        "isControl": true,
        "percentage": 0.5
    },
    {
        "groupLabel": {
            "name": "B"
        },
        "isControl": false,
        "percentage": 0.2
    },
    {
        "groupLabel": {
            "name": "C"
        },
        "isControl": false,
        "percentage": 0.3
    }
]
```



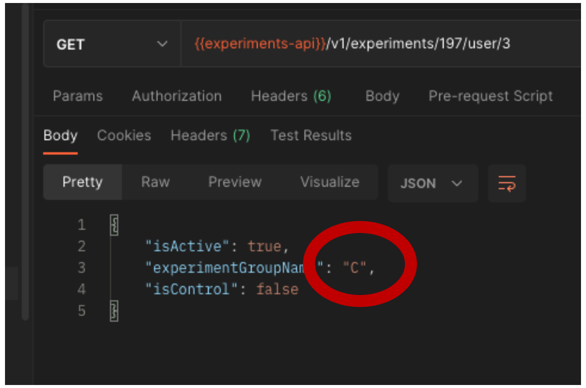
GET {{experiments-api}}/v1/experiments/197/user/5

```
{
    "isActive": true,
    "experimentGroupName": "A",
    "isControl": true
}
```



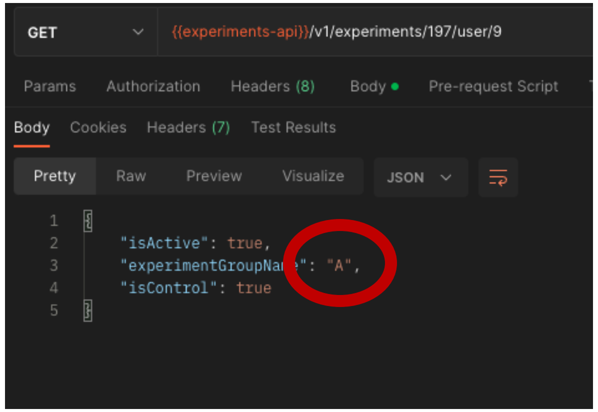
GET {{experiments-api}}/v1/experiments/197/user/13

```
{
    "isActive": true,
    "experimentGroupName": "B",
    "isControl": false
}
```



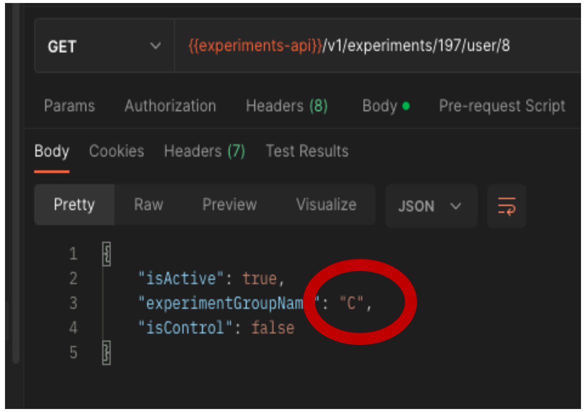
GET {{experiments-api}}/v1/experiments/197/user/3

```
{
    "isActive": true,
    "experimentGroupName": "C",
    "isControl": false
}
```



GET {{experiments-api}}/v1/experiments/197/user/9

```
{
    "isActive": true,
    "experimentGroupName": "A",
    "isControl": true
}
```

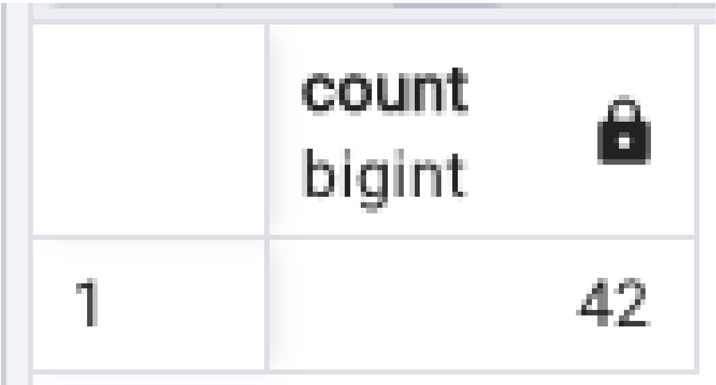


GET {{experiments-api}}/v1/experiments/197/user/8

```
{
    "isActive": true,
    "experimentGroupName": "C",
    "isControl": false
}
```

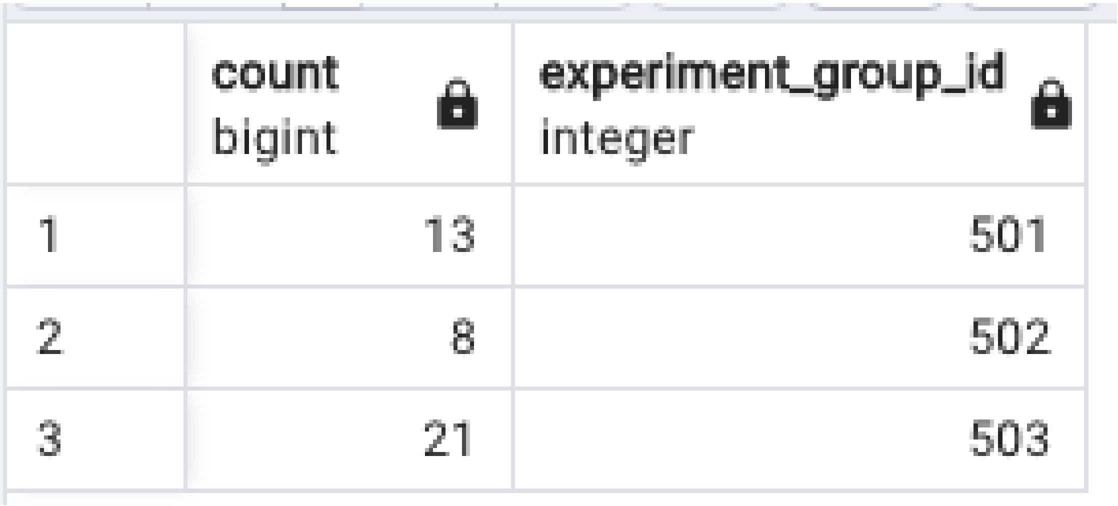
Well, that was hard to read...

Total Users



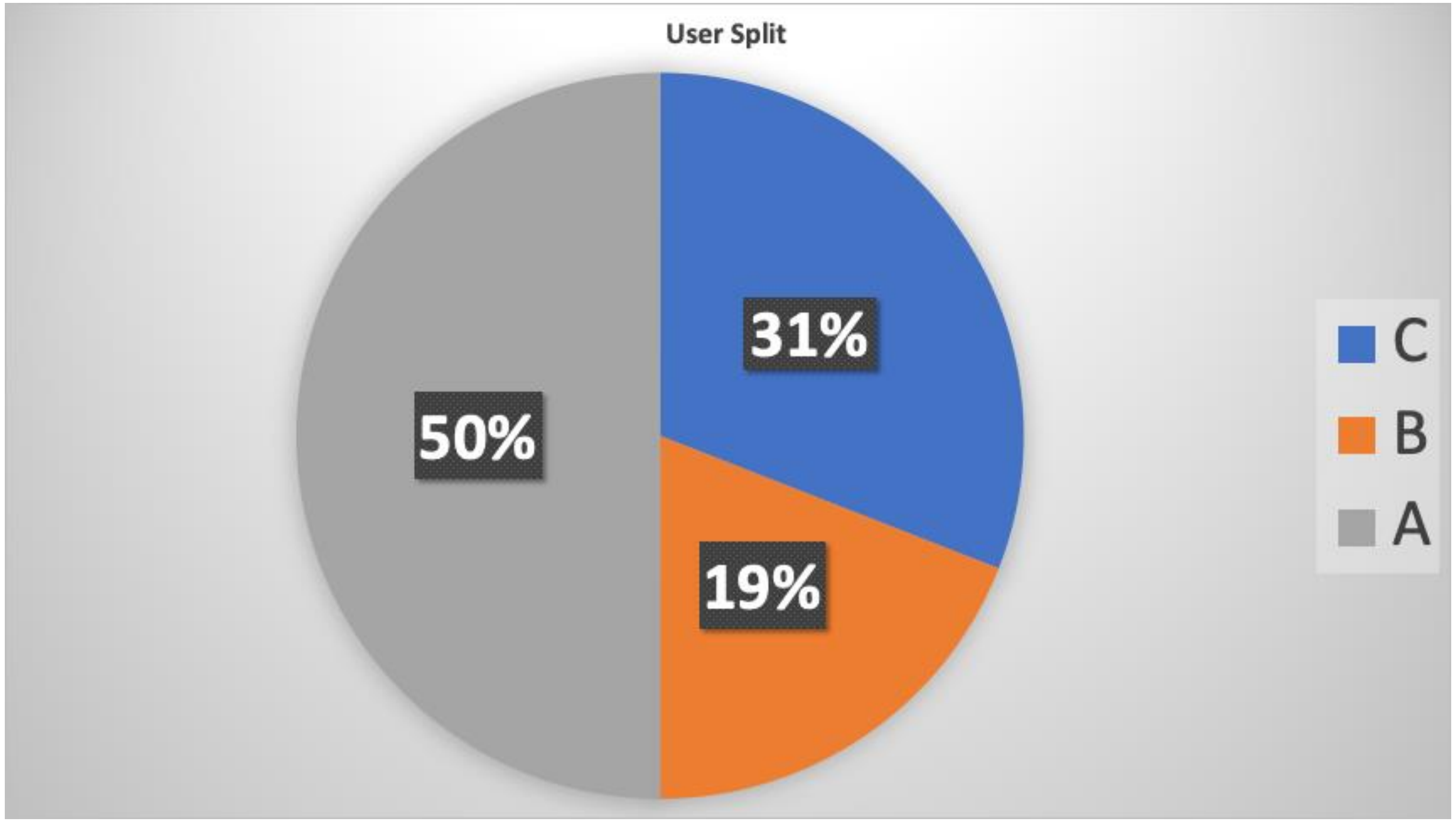
	count bigint
1	42

Users Per Group



	count bigint	experiment_group_id integer
1	13	501
2	8	502
3	21	503

New traffic split into buckets



The designer is your friend

CMM has a style guide

Time to put the PowerPoint together!

Framing Devices

We help people get the medicine they need to live healthier lives

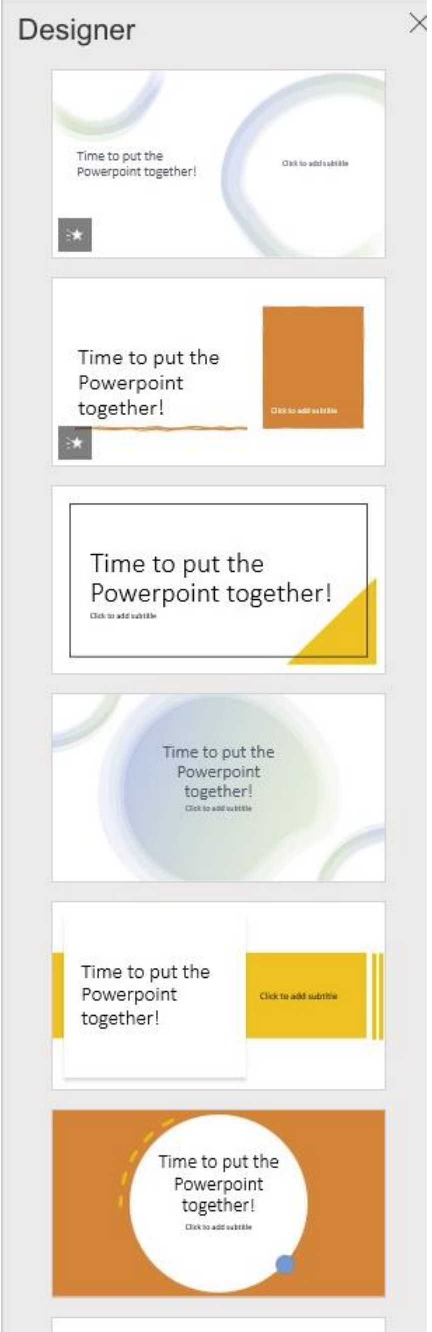
Clearing the path *for patients*

DISAPPOINTING
DISHEARTENING

27% of prescription abandonment is related to *access challenges.*

Our solutions leverage an *unmatched network*

Unifying Strengths


Designer

Time to put the Powerpoint together!

Click to add subtitle

Time to put the Powerpoint together!

Click to add subtitle

Time to put the Powerpoint together!

Click to add subtitle

Time to put the Powerpoint together!

Click to add subtitle

Time to put the Powerpoint together!

Click to add subtitle

Time to put the Powerpoint together!

Click to add subtitle

Nuanced control over logs and experiences

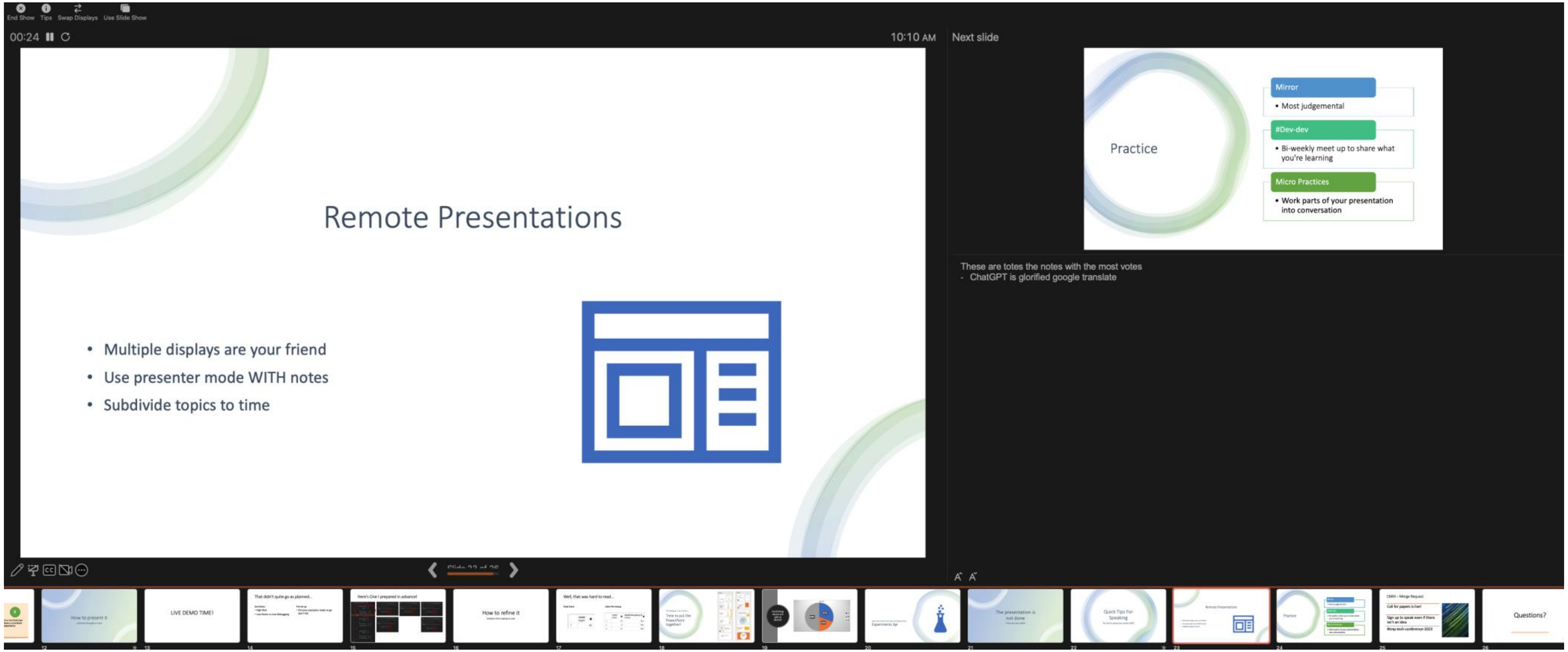
Experiments Api

Quick Tips For Speaking

Try not to poop your pants (Jeff)

Remote Presentations

- Multiple displays are your friend
- Use presenter mode WITH notes
- Subdivide topics to time



Remote Presentations

- Multiple displays are your friend
- Use presenter mode WITH notes
- Subdivide topics to time

Practice

Mirror

- Most judgemental

#Dev-dev

- Bi-weekly meet up to share what you're learning

Micro Practices

- Work parts of your presentation into conversation

These are totes the notes with the most votes

- ChatGPT is glorified google translate

Practice

Mirror

- Most judgemental

#Dev-dev

- Bi-weekly meet up to share what you're learning

Micro Practices

- Work parts of your presentation into conversation

CMM – Merge Request (Coming Mid June!)

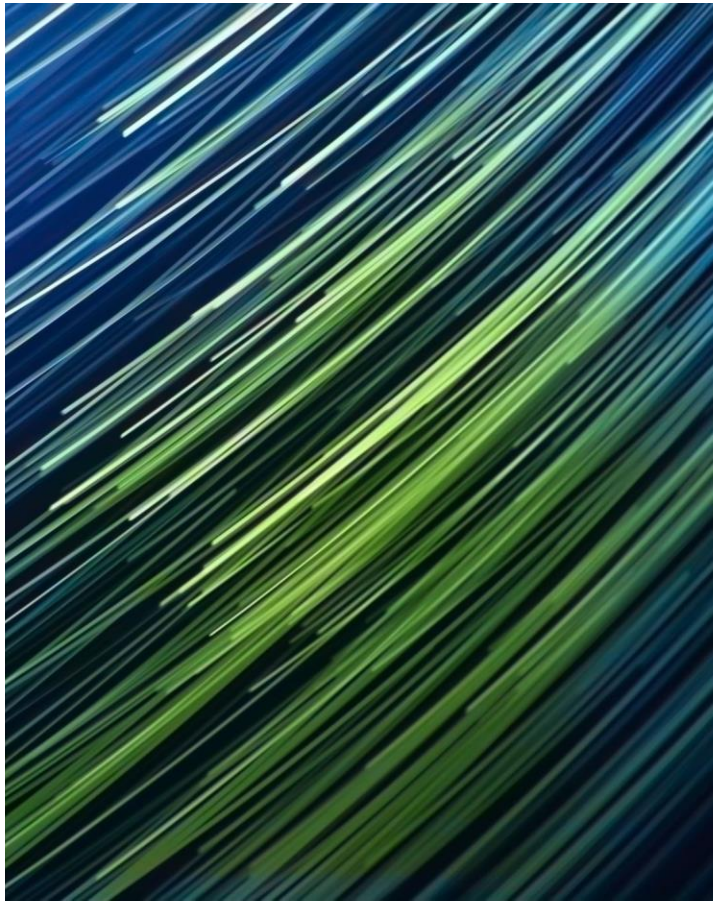
Virtual Tech Conference

Call for papers is live!

Link in #CMM-developers

Sign up to speak even if there isn't an idea

#tmp-tech-conference-2023



In Summation

WHAT DO YOU WANT TO SAY?

HOW DO YOU WANT TO SAY IT?

HOW DO YOU WANT IT DIGESTED?

The presentation is not done

Time was just called

Questions?