

Why we stream

📌 Objectives

- Learn the general history of CodeRefinery streaming.
- Discuss the benefits of streaming and recording
- Discuss the downsides and difficulties

Instructor note

- Discussion: 10 min
- Q&A: 5 min
- Exercises: 0 min

This is a general discussion of the topics of the day, focused on the history of why we stream and record, how we got here, and how people feel about it. We won't focus on how anything is done.

Icebreaker questions

- What is the most people you have taught for?
- What are the distinct parts of teaching, that can be separated? (teaching, helping, etc)

What is streaming and recording?

- Streaming is mass communication: one to many
 - Interaction audience→presenters is more limited (but different)
- Using consumer-grade tools, normal people can reach huge audiences by Twitch/YouTube/etc.
- This isn't actually that hard: people with much less training than us do it all the time.
- They reach huge audiences and maintain engagement for long events.

Recording and rapid video editing is useful even without streaming.

History

- In-person workshops
 - 3 × full day, required travel, infrequent, one-shot
- Covid and remote teaching

- Traditional “Zoom” teaching several times
- Mega-CodeRefinery workshop
 - 100-person Zoom teaching
 - Emphasis on teams
- Research Software Hour
 - Livestream free-form discussions on Twitch
- Streamed “HPC Kickstart” courses

Benefits and disadvantages

Benefits:

- Larger size gives more (but different) interaction possibility
 - “Notes” for async Q&A
- Recording (with no privacy risk) allows instant reviews
- Stream-scale allows for many of the things you have learned about in days 1-3.

Disadvantages:

- Requires training for using the tools
- Requires a certain scale to be worth it
- Coordination is much harder for big events

When would I recommend it?

- No: For small courses
- Questionable: Medium sized courses with no videos
- Yes: When you want the largest audience
- Yes: When you want without registration required
- Yes: When you want good reusability/fast videos

Future prospects (briefly)

- Streaming probably stays as a CodeRefinery tool
- We *can* scale our courses much larger than they are now. Why don't we, together with others?
- These tools are useful in other places too
 - I've used them to record my own talks to make videos of my single-person presentations or record conference talks nicer than Zoom.

Q&A

Q&A from audience

📌 Keypoints

- Streaming optimizes a course for different things
- The disadvantages can be compensated for
- There are benefits and disadvantages