

Reflections on Remote Working from Cloud Enablement

Introduction

Cloud Enablement are a distributed team with one team member working from the ~~Netherlands~~ Netherlands and other UK based members working from home frequently. Although most team members commute into Bracken House, we have been working in a remote first manner for several years. This document was produced during a session where we reflected on what we do. Hopefully it helps others.

Tools we use

Visual Studio Code with Live Share - Coding IDE

Everyone on the team uses [VSCode](#) along with the [Live Share Extension](#). The live share extension provides capabilities like google docs collaboration but for code. Multiple people can all collaborate on the same source code. You can also follow someone which is useful to navigate as a group between files.

Only source files are shared, the IDE config is local. We have a [standardised VSCode environment](#) and [config settings](#) to ensure everyone is using the same validators / linters / config. Collaborating on code with different validator / linter config can be frustrating and unproductive.

There is a problem with sharing the terminal in VSCode with live share - people with different window sizes get weird wrapping making the terminal unusable (half sentences, cursor not in the right position). See Tmate below for solution.

The Live Share extension will ask all users joining a sharing session to login either to their GitHub or Microsoft account. This is required so the other users can identify you when you edit the code - in a similar manner as a Google Doc. It is possible to not login to either but you'll be identified as a Guest in the session.

Tmate - Terminal Sharing - <https://tmate.io/>

Easily setup a shared unix terminal between multiple team members, based on the [tmux](#) terminal multiplexer. Compensates for poor terminal sharing experience in VSCode. TMatе only gives a single common screen view, so scrollback history is possible only by piping your command into `|more` or `|less`. TMatе session can also be shared as a read-only terminal.

- For additional security we are using a private/public SSH key when setting up the session.

- `tmate -k tmk-UNIQUEKEY -n mobbing -a ~/.ssh/team-key.pub`

- Allowing a consistent entry point for the shared named session.
 - `ssh cloud-enablement/mobbing@lon1.tmate.io`

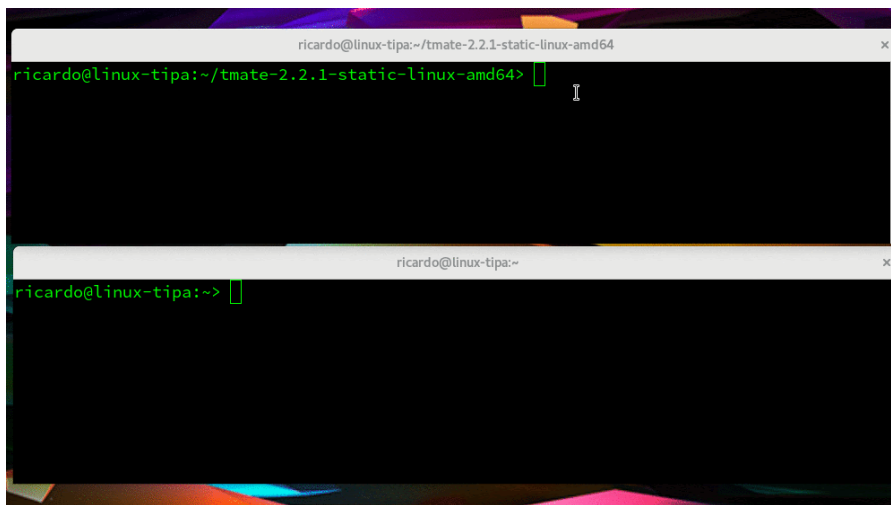
View when starting a new Session:

```
Tip: if you wish to use tmate only for remote access, run: tmate -F
To see the following messages again, run in a tmate session: tmate show-messages
Press <q> or <ctrl-c> to continue

-----

Connecting to ssh.tmate.io...
Note: clear your terminal before sharing readonly access
ssh session read only: ssh ro-UcWMSDUnt5TZ3dhDYtWENHYX7@lon1.tmate.io
ssh session: ssh cloud-enablement/mobbing@lon1.tmate.io
```

Example of how a shared session looks:



Slack

The team has a private slack channel (#cloud-enablement-team) and a separate public channel to provide (#cloud-enablement) support for our users.

When working remotely we make sure our presence is known using Slack to chat and keep the environment close to what it is at the office.

Google Docs

A Google doc is great for collaborating on non-code stuff e.g. email comms we want to send, design docs, todo lists etc. Google drawing / lucid charts are also useful for sketching up architecture and other sorts of diagrams.

JIRA

Manage initiatives, stories, support tickets, issues, planning, tracking, etc

Timer

[Timer \(OS X only\)](#) - for tracking mob programming intervals / driver rotation.

Lastpass

Use Shared folders in LastPass to organise secrets that need to be shared with all team members.

Audio / Visual

Hangouts

[Google Meet](#) is the FT preferred option for remote meetings. It is good for large groups and mobbing in a room with 1 or 2 remote individuals. However image quality issues such as code blurring when scrolling and screenshare losing focus can be annoying and are amplified on a big screen for coding work.

Another option is [whereby.com](#) for video calls. The quality of screenshare is better and there is minimal blurring when moving / scrolling around code. The Audio quality is also better in our opinion. Whereby allows you to use the same URL link for all of your meetings so to setup you just use your team link and this starts the session. When it is time for a remote meeting you can simply go to your team link and then put the team link in slack so other members of your team can join you. The only downside is that the free option is restricted to only 4 people.

Headsets

- FT supplied headsets are good when working amongst other people or there's background noise, and better than most others for mic quality.
- Otherwise the mac internal mic is good enough if you have room to yourself and there's minimal background noise.
- We have noticed that high CPU load on mac can cause fans to kick in which can cause background noise and lower quality audio experience for others on call if using the mac laptop mic.
- Using [Mac Fan Control](#) can help to monitor fan speed and temps.
- It's not a typing competition, be aware of how noisy your typing is.
- If your typing is loud, you can mute your microphone whilst you are typing.

External Monitor

- It's good to be able to see everything you need including your teammates faces.
- An external monitor makes things easier when you need real estate for multiple windows e.g. code environment, tmate (terminal) , screenshare, hangout for people faces
- Can also use a TV with a suitable adaptor (e.g. usb to HDMI or chromecast/appleTV)

Desk and Environment

- Use a desk, table or other hard surface - compared to working with a laptop on lap it minimises shuffling noise and video jitter which can be annoying for others. Also better for posture versus sitting on the sofa.
- Find a quiet room if you can't use a headset to minimise background noise.

How we work

Daily Stand Ups

Quick 10 minute sync each morning.

Tracking and Planning sessions

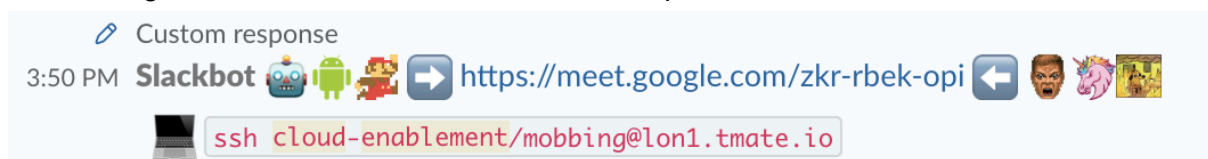
We have scheduled 1 hour team sessions Monday and Thursday each week where we

- a) review progress on OKRs, initiatives, stories etc
- b) plan future work, break initiatives into smaller units of work, define desired outcomes & tasks, discuss options and technical solutions / designs, prioritisation, etc.

Daily Mob Programming Sessions

We all typically swarm on a single work item at a time using a technique called [mob programming](#).

- Agree when to mob, have regular slots in calendars
- Announce that mobbing is starting on slack, provide links for screenshare, vscode, tmate, hangout etc. We created a custom slack response.



- Agree what the next step is when you start a mob session and also when you finish a step.
- Keep track of progress e.g. ticking off items in Jira story / text file in project / google doc.
- Driver shares their screen - keeps everyone engaged and provides context
- The rest of the team are navigators, deciding as a group which way to go
- Have timed driving intervals, suggest rotation of driver every 10 minutes - use a timer - keeps everyone focussed and engaged, sessions feel productive
- Swap screen share whenever driver rotates, everyone sees drivers screen
- Have an order for driver rotation - e.g. alphabetical order by first name
- Check in periodically on where the team are heading, is everyone on same page
- Take breaks when you need to. Remember that not everyone will need breaks at the same time but this is fine, not everyone needs to be present all the time.

- Say hi when you join, bye when you leave and consider pausing to set context for new joiners or people rejoining after a break - working remotely needs more context so people are aware if others are around, have left and what's being worked on etc.
- Conclude a mob session by writing up some session notes. This provides context for the next mob session.

Other Tips

- Try and keep people's faces on screen. Prefer face to face over audio only.
- Plug in your laptop, video hangouts will drain a mac battery in 1-2 hours
- When working from home remember to get some exercise / get outside every day since you aren't commuting, go for a walk at least
- Have conversations that aren't about work with your colleagues as you would do in the office

Other External References

- [Remote Mob Programming @ INNOQ](#)
- [Cloud Posse Developer Hub info about TMate](#)