

DevOps for research

Jun Kato

AIST, Arch Inc.

#non_research_tips #情報科学研究補助技法

Icons by FontAwesome, licensed under SIL Open Font License 1.1. For more information, visit: <https://fontawesome.com/license/free>

Links to open-source code in this lecture

- 🔗 **arcatdmz/non-research-tips-gitlab-ci**
 - GitLab CI/CD simple example
 - <https://gitlab.com/arcatdmz/non-research-tips-gitlab-ci>
- 🔗 **arcatdmz/non-research-tips-devcontainer**
 - VS Code DevContainer for consistent development experience
 - Node.js + MySQL + Prisma
 - <https://github.com/arcatdmz/non-research-tips-devcontainer>
- 🔗 **arcatdmz/acm2024-primary-article-template**
 - Editing, linting, and sharing papers
 - Overleaf + GitHub Actions + GitHub Pages + Slack
 - <https://github.com/arcatdmz/acm2024-primary-article-template>
- 🔗 **arcatdmz/arcatdmz.github.io**
 - Single source of truth for multiple views
 - Gulp + Pug + BibTeX etc.
 - <https://github.com/arcatdmz/arcatdmz.github.io>
- 🔗 **arcatdmz/research-assistant-playbook**
 - Agentic development of agentic research assistant
 - <https://github.com/arcatdz/research-assistant-playbook>

DevOps for research: agenda

- Development environment setup
- Continuous integration/delivery
- Web technologies for development/public relations
- Agentic development of your agentic assistant

Development environment setup

DevOps for research

Development environment setup

- Package managers for Windows/macOS
- Two-factor authentication for improved security
- SSH config for custom networking
- DevContainer for consistent development experience

Package managers for Windows/macOS

- Windows
 - winget [**recommended**]
 - scoop, Chocolatey
- macOS
 - Homebrew [**recommended**]
 - macPorts

Two-factor authentication for improved security

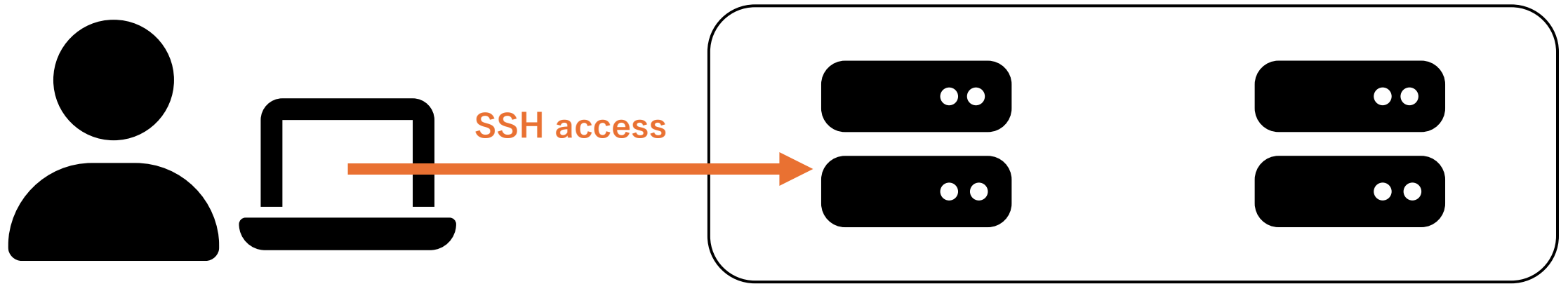
- Microsoft Authenticator
- Authy [**recommended**]
- 1Password

SSH config

- Cf. Week 1: SSH + server coding
- You can do a lot with SSH, probably more than you'd expect

SSH config case studies 1:

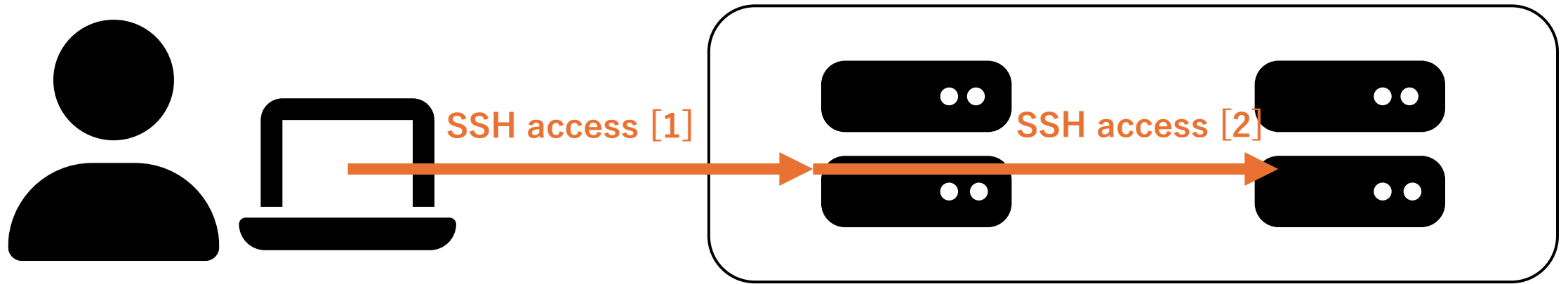
ProxyJump for jumping hosts



- Suppose you have a server that can only be accessed from within a restricted network that includes another server
- You have SSH keys to access both the bastion host (intermediate server) and the final destination server

SSH config case studies 1:

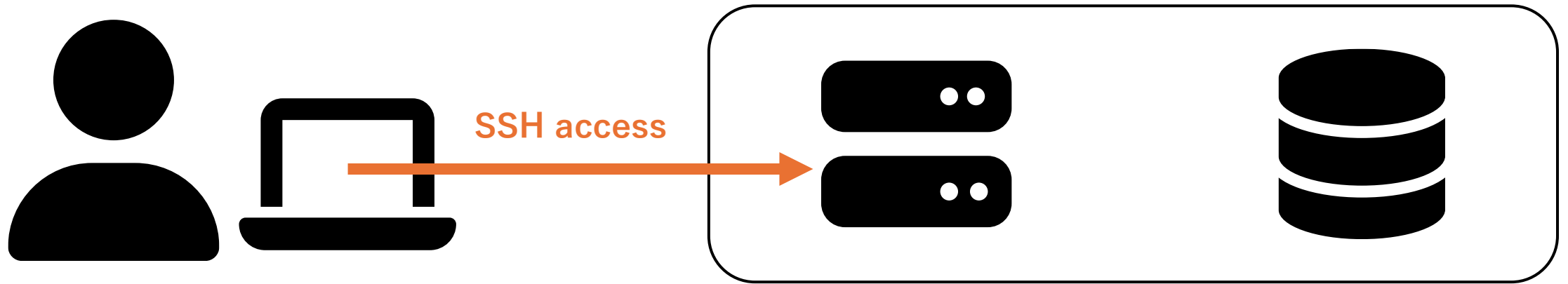
ProxyJump for jumping hosts



```
1 Host ssh-config-proxy-jump
2   HostName xxx.xxx.xxx.xxx
3   ProxyJump ssh-config-bastion
4 Host ssh-config-bastion
5   HostName final-destination
```

SSH config case studies 2:

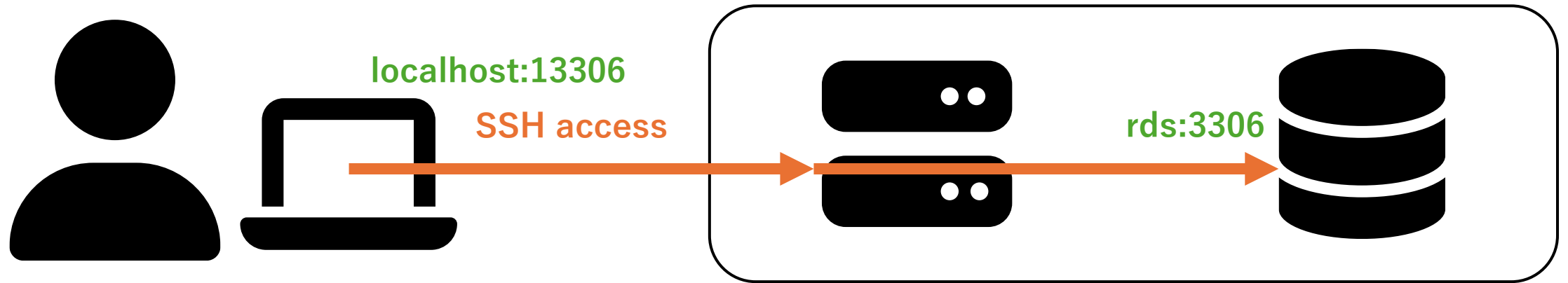
LocalForward for remote service access



- Suppose you have a database server that can only be accessed from within a restricted network that includes an app server
- You have SSH access to the app server, but not to the database server

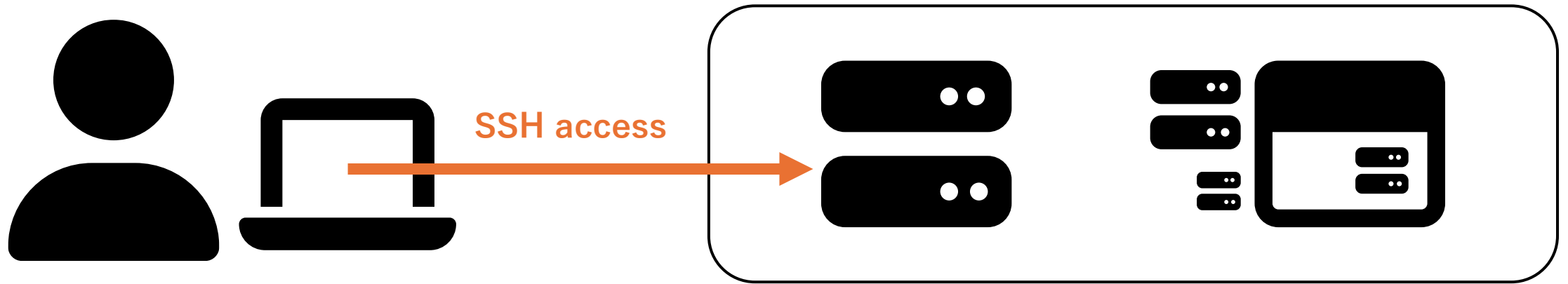
SSH config case studies 2:

LocalForward for remote service access



```
1 Host ssh-config-localforward
2   User ec2-user
3   HostName xxx.xxx.xxx.xxx
4   IdentityFile ~/.ssh/xxx.pem
5   LocalForward 13306 rds:3306
```

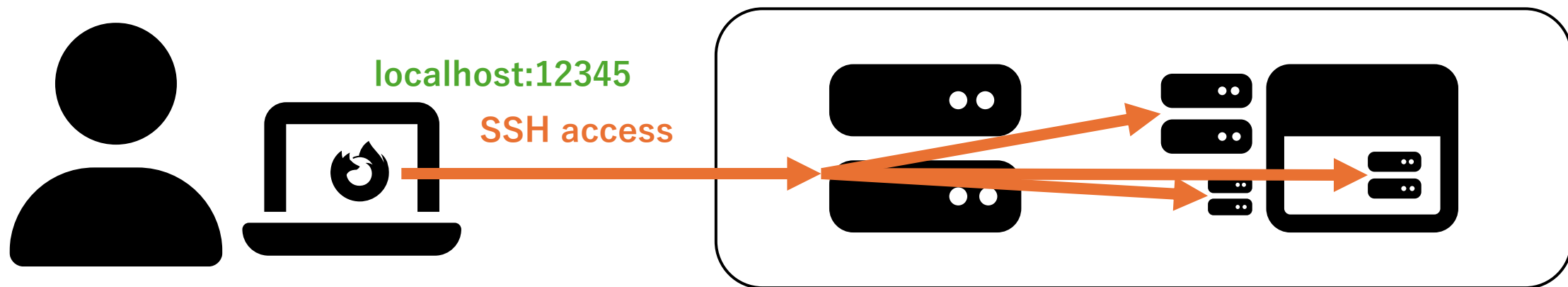
SSH config case studies 3: **DynamicForward** for SOCKS v5 proxy



- Suppose you have a service that can only be accessed from within a restricted network that includes an app server
- You have SSH access to the app server, but the service runs on multiple servers to be accessed via a web browser

SSH config case studies 3:

DynamicForward for SOCKS v5 proxy

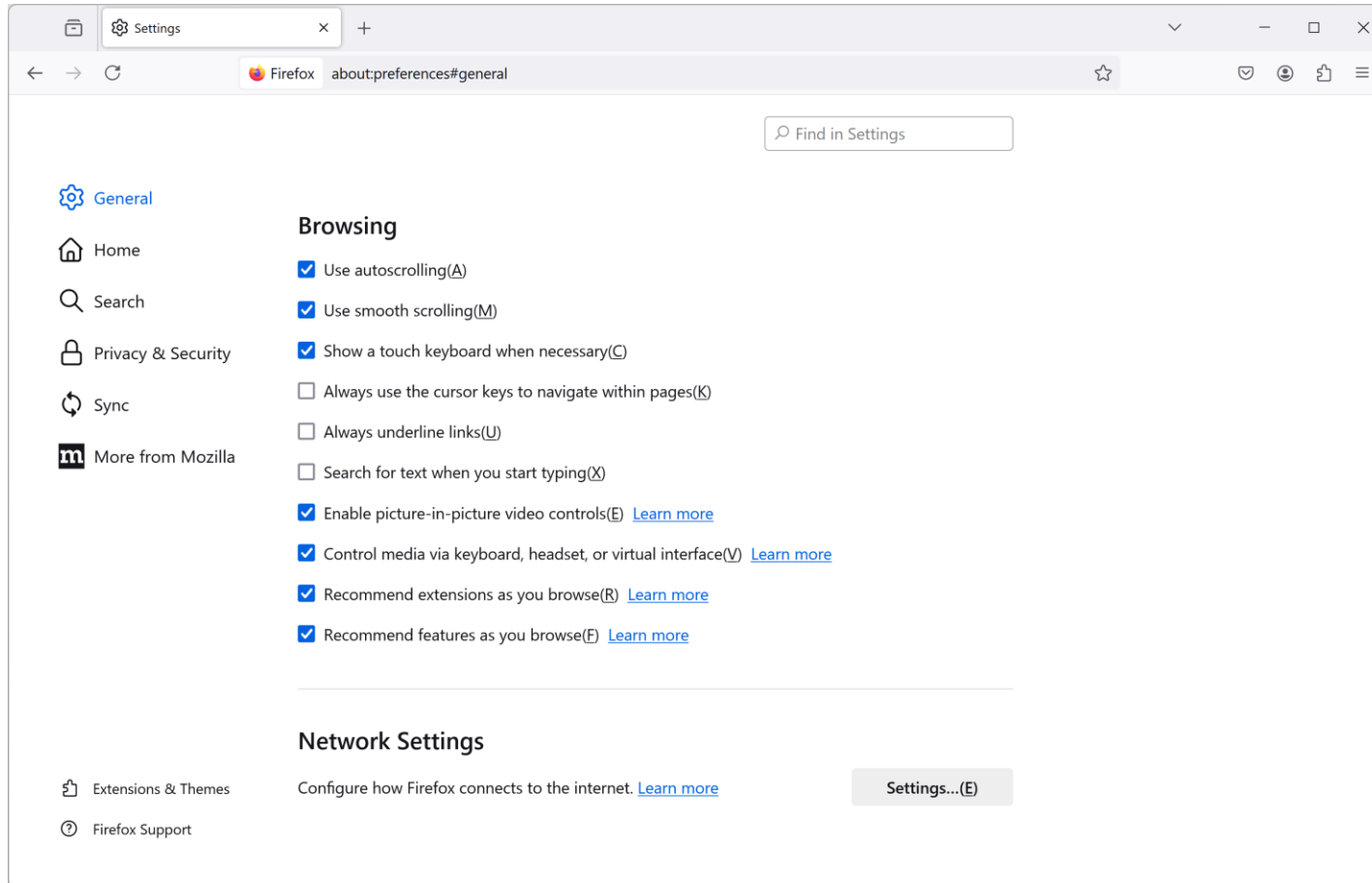


```
1 Host ssh-config-dynamicforward
2   User remote-user
3   HostName xxx.xxx.xxx.xxx
4   IdentityFile ~/.ssh/xxx.pem
5   DynamicForward 12345
```

SSH config case studies 2:

DynamicForward for SOCKS v5 proxy

Firefox browser
config:
Network Settings



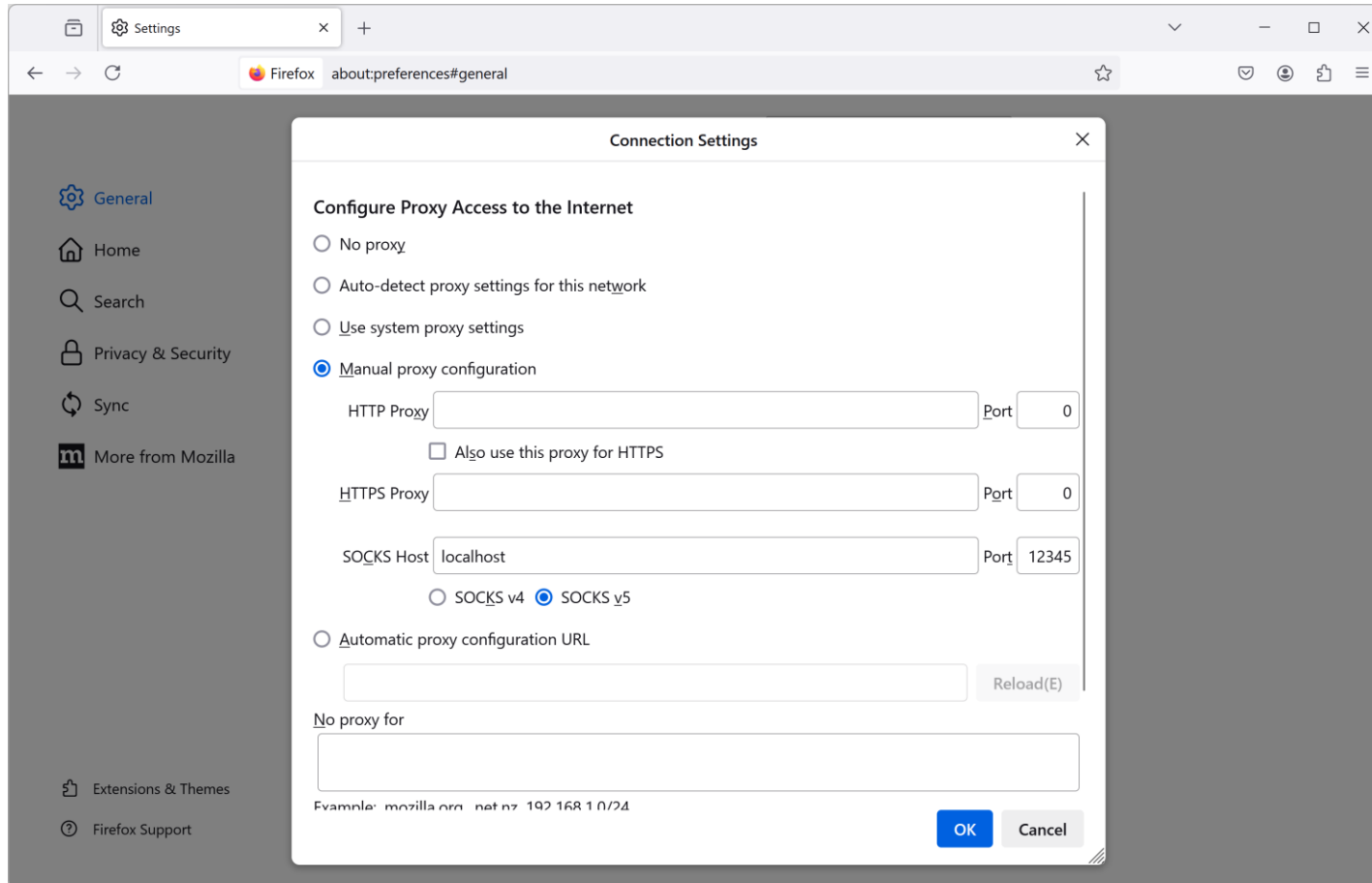
SSH config case studies 2:

DynamicForward for SOCKS v5 proxy

Firefox browser
config:

Network Settings >
Manual proxy
configuration >

**SOCKS Host, Port,
SOCKS v5**



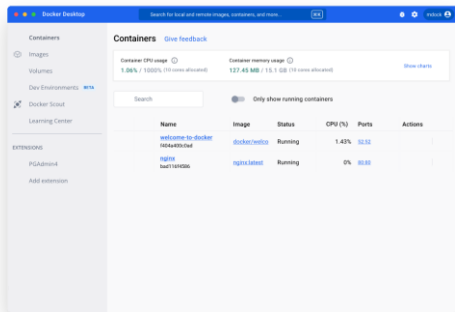
DevContainer (VS Code feature)

- Prerequisite: Docker Desktop (or Rancher Desktop)
- Useful feature for Windows: WSL2
- DevContainer for consistent development experience

DevContainer (VS Code feature): Prerequisite

Docker Desktop

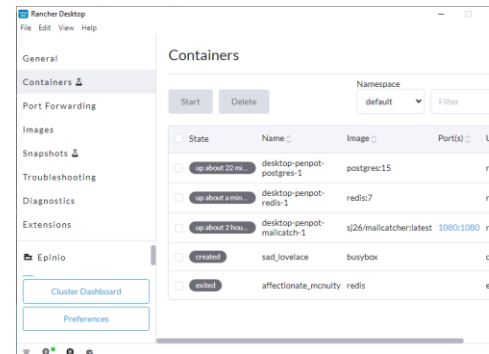
- Free for educational, non-commercial, personal, and small business use
- **docker** and **docker-compose** command support



<https://www.docker.com/wp-content/uploads/2023/08/docker-desktop-hero-v2.svg>

Rancher Desktop

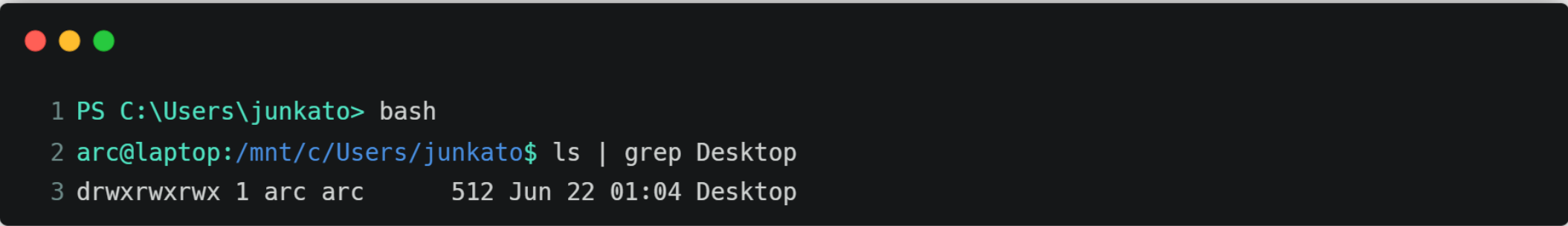
- Completely OSS, free for anyone who agrees the terms of use



https://suse-rancher-media.s3.amazonaws.com/desktop/v1.14/ui-main/Windows_Containers.png

DevContainer (VS Code feature): Useful Windows feature: WSL2

- Windows Subsystem for Linux 2
- To install and start using bash on Ubuntu:
 - `wsl --install` in the administrator privileged console
 - Restart system
 - `wsl --set-default-version 2` in the admin console
 - `bash` in console



```
1 PS C:\Users\junkato> bash
2 arc@laptop:/mnt/c/Users/junkato$ ls | grep Desktop
3 drwxrwxrwx 1 arc arc      512 Jun 22 01:04 Desktop
```

DevContainer (VS Code feature) for consistent development experience

- Use the same runtime (e.g., Node.js)
- Use the same package manager (e.g., npm)
- Use the same backend server (e.g., MySQL)
- Use the same VS Code extensions, editor config, etc.

...

Want a consistent development experience across machines?

DevContainer has you covered!

DevContainer (VS Code feature) for consistent development experience

```
1 {  
2   "name": "Example dev container",  
3   "dockerComposeFile": ["/docker-compose.yml"],  
4   "service": "node",  
5   "workspaceFolder": "/workspace",  
6   "customizations": { "vscode": {  
7     "extensions": [  
8       "esbenp.prettier-vscode",  
9       "EditorConfig.EditorConfig",  
10    ]  
11  } },  
12  "forwardPorts": [3000, "mysql:3306", "prisma:5555"],  
13 }
```

Example [devcontainer.json](#)



DevContainer (VS Code feature) for consistent development experience

```
1 version: "3.7"
2 services:
3   mysql:
4     image: mysql:8
5     healthcheck: { test: ["CMD", "mysqladmin ping -h localhost"] }
6   prisma:
7     # mysql client
8   node:
9     build:
10       context: ../
11       dockerfile: ../devcontainer/Dockerfile
12     command: sleep infinity
13     # more config entries omitted
```

Example [docker-compose.yml](#)



By the way...

Share beautiful images of your code



```
1 # What does this do?
2 ## Carbon lets you create and share beautiful images of your source
3
4 You know all of those code screenshots you see on Twitter? Although
  thought there was room for improvement in the aesthetic department.
  try it out and impress all your developer and designer friends. 🎨
5
6 ## Import
7 There are a few different ways to import code into Carbon:
8
9 - Drop a file into the editor
10 - Append a GitHub gist id to the url (example)
11 ...
```

<https://carbon.now.sh/>



Continuous integration/ continuous delivery

DevOps for research

Continuous integration/continuous delivery

- GitHub or GitLab?
- Self-hosted runners
- CI/CD use cases 1: linting papers
- CI/CD use cases 2: running heavy workloads on clouds
- CI/CD use cases 3: ChatGPT integration
- CI/CD use cases 4: publishing websites

GitHub or GitLab?

from CI/CD perspective



GitHub Actions

- Not much difference for basic usage
- writes config under `~/.github/workflows/`
- **Pros:** Has a vast library of community-defined “Actions” useful for various purposes
- **Cons:** You’re vendor-locked!



GitLab CI/CD

- writes config in `~/.gitlab-ci.yml`
- **Pros:** Can be hosted entirely on your own servers without worrying much about resource limits
- **Cons:** Maintenance costs

GitLab CI/CD basic example

.gitlab-ci.yml

```
1 image: docker:dind
2
3 before_script:
4   - docker login -u $CI_REGISTRY_USER -p $CI_REGISTRY_PASSWORD $CI_REGISTRY
5
6 stages:
7   - build
8   - deploy
9
10 variables:
11   IMAGE_PATH: ${CI_REGISTRY}/${CI_PROJECT_NAMESPACE}/${CI_PROJECT_NAME}
12   CONTAINER_NAME: ${CI_PROJECT_NAME}-${CI_JOB_NAME}
13 # more config entries omitted
```

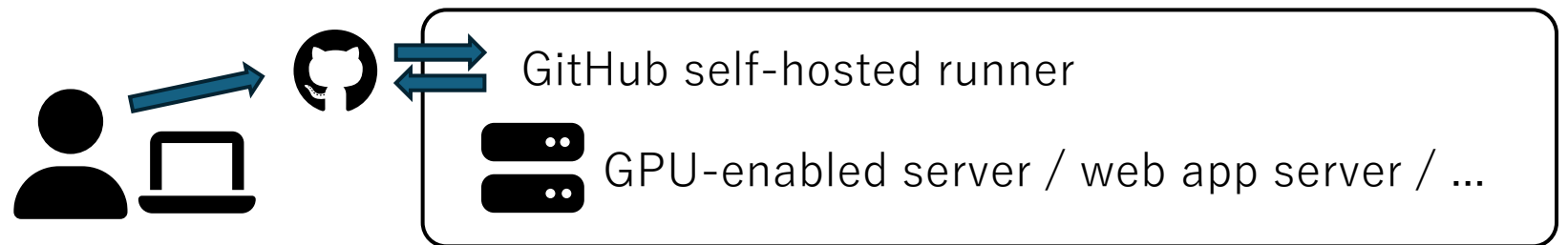
Example [.gitlab-ci.yml](#)



Self-hosted runners

Basics and usages

- GitHub Actions and GitLab CI/CD both support self-hosted runners – CI/CD workers that run on machines under your control, such as **modern NAS devices** and various servers
- These are especially useful if
 - you have **a handy machine (or some special machines)** and want to avoid being charged for running an automated process
 - you have **a deployment target on cloud computing platforms** and want to securely update running services on each push



CI/CD case studies 1:

Editing, linting, and sharing papers

Overleaf + GitHub Actions + GitHub Pages + Slack

- Overleaf has a good GitHub integration by default
- GitHub Actions can automatically process pushed TeX files
- GitHub Actions can upload the end result to GitHub Pages
- GitHub Actions can post a message to Slack upon completion

<https://github.com/arcatdmz/acm2024-primary-article-template>



Let's look at the actual code on GitHub:

CI/CD case studies 2:

Running heavy workloads on clouds

Amazon EC2 + GitHub + GitHub Actions

- Automated self-hosted runner management with GitHub personal access token of **repo** scope
- Automated EC2 instance management with AWS access keys with permissions for **ec2:*** actions
- On-demand tests/experiments with EC2 instance!
<https://github.com/machulav/ec2-github-runner>

CI/CD case studies 3:

ChatGPT integration

GitHub + GitHub Actions + OpenAI Platforms API

For each GitHub comment,

- look for harmful text ([moderations](https://github.com/takahiroanno2024/election2024/blob/main/.github/workflows/comment-review.yml#L89-L96) endpoint)
<https://github.com/takahiroanno2024/election2024/blob/main/.github/workflows/comment-review.yml#L89-L96>
- look for URLs and check if they look harmful
<https://github.com/takahiroanno2024/election2024/blob/main/.github/workflows/comment-review.yml#L54-L87>

CI/CD case studies 4: Publishing web services

To be covered in the next section!

Web technologies for development/public relations

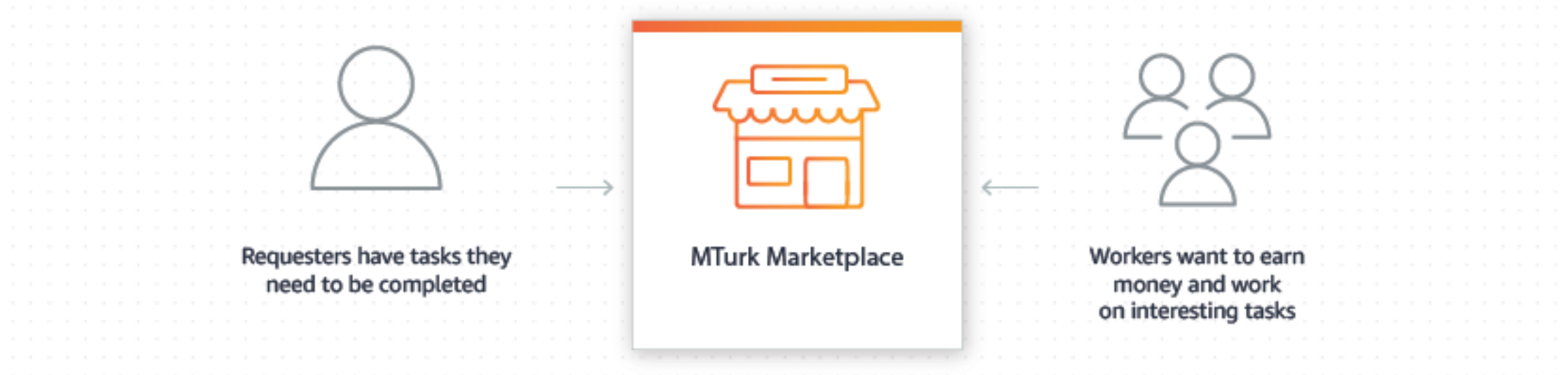
DevOps for research

Web technologies for development/public relations

- Crowdsourcing
- npx + Prettier for code beautification
- Infrastructure for web deployments
- Languages and frameworks
- Tips for making informative portfolio pages

Crowdsourcing

<https://www.mturk.com/>



- Amazon Mechanical Turk, Yahoo! Crowdsourcing cf. [acarin website](#)
- Crowdsourcing and LLMs (search [llm modeling human behaviors to replace crowdsourcing](#) or visit [HCOMP website](#))

npx + Prettier for code beautification

- `npx` comes with Node.js by default and Prettier is a well-known OSS code formatter
- `npx prettier --write {your files}` is useful, even if you don't write JavaScript (e.g, HTML and Markdown files; more languages if combined with plugins)
- `npx http-server {your directory}` is also useful
- See also: `localtunnel`, `cpx`, `cross-env`, ...

Tell me your favorite npm package!

Infrastructure for web deployments

- For static sites: GitHub Pages and Cloudflare Pages
- For a bit more complex sites: Netlify and Vercel
- For backends: PlanetScale TiDB and... Google Sheets!
- For professional-minded developers: AWS, GCP, Azure

Web technologies case studies 1: Cloudflare Pages + Google Sheets

- Google Forms ➡ Google Sheets
- Google Sheets ➡ Cloudflare Pages

<https://github.com/ChezTachibana/website>



Let's look at the actual code on GitHub:

Languages and frameworks

- HTML, Pug, Markdown
- React (JSX/TSX), MDX, and more!
- Responsive design for free with CSS frameworks
- Pretty icons with FontAwesome and alternatives

Languages and frameworks

HTML, simplified

Pug

- **html**
 - **body**
 - **h1#mytitle** Main title
 - **p** This is the main paragraph
 - **ol.myclass**
 - **li** This is the first item
 - **li** second item
 - **li** third item
 - **p** You can add
 - **a(href="#mytitle")** links
 - **|** pretty easily!

Markdown

- **#** Main title **{#mytitle}**
- This is the main paragraph
 - **1.** This is the first item
 - **2.** second item
 - **3.** third item
- You can add **[links](#mytitle)** pretty easily!

Languages and frameworks

HTML, evolved

React (*.jsx)

```
// Every user interface component is a function
export function MyButton = () => {
  const handleClick = useCallback(() => { ... }, []);
  return <Button onClick={handleClick}
    className="my-button">My button!</Button>
}
```

MDX

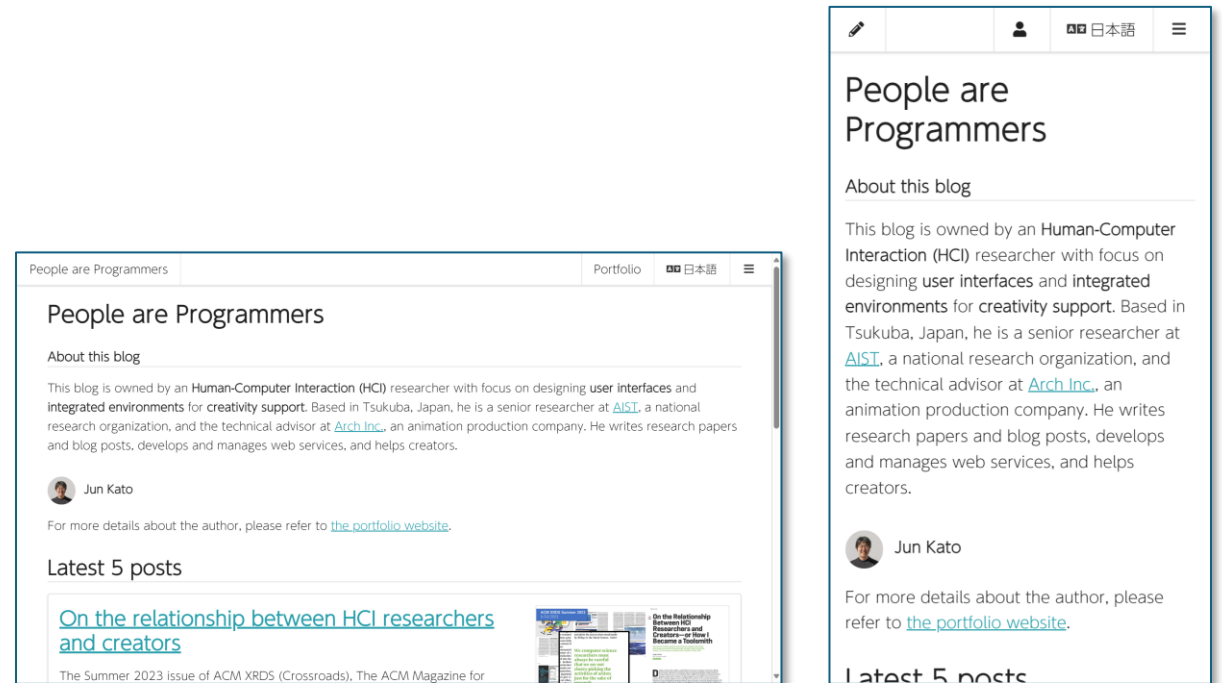
```
import { MyButton } from
"../src/MyButton";
```

This is the main paragraph
with a custom `<MyButton />`.

Languages and frameworks

CSS frameworks for responsive design

- Bootstrap
- Semantic UI / Fomantic UI
- and more!



Search for “CSS frameworks”

Tips for making informative portfolio pages

- Stick to static sites, allow easy copy and paste
- Show the story, not only the papers
- Single source of truth, multiple views for multiple purposes

Web technologies case studies 2: Portfolio web page

<https://github.com/arcatdmz/arcatdmz.github.io>



Let's look at the actual code on GitHub:

Agentic Development of Your Agentic Assistant

DevOps for Research

Build Your Own Agentic Assistant

- Organize PDF files
- Filter your X bookmarks to find books to buy
- Plan your day
- Review what you did that day/week/month

 [2023, Borowski et al] Between Principle and Pragmatism - Reflections on Prototyping Computational Media with W...	✓	2024/06/01 13:14
 [2023, Chang et al] The Second Workshop on Intelligent and Interactive Writing Assistants (3544549.3573826-2).pdf	✓	2024/10/09 17:01
 [2023, Devendorf et al] Towards Mutual Benefit - Reflecting on Artist Residencies as a Method for Collaboration in DI...	✓	2024/11/26 13:42
 [2023, Gobert] Lorgnette - Creating Malleable Code Projections (uist2023-lorgnette).pdf	✓	2024/08/01 1:13
 [2023, Ito et al] AI-powered Rewriting Support Software in Context with Other Tools (2023jp-06).pdf	✓	2023/11/03 11:15
 [2023, Li et al] Beyond the Artifact - Power as a Lens for Creativity Support Tools (PREPRINT-power-csts).pdf	✓	2024/10/24 13:52
 [2023, Lottridge] Perceptions of Colour Pickers in Virtual Reality Art-Making (WSCG2023_ColourPicker_CameraReady...	✓	2025/05/01 18:26
 [2023, Oc et al] An HCI-Centric Survey and Taxonomy of Human-Generative-AI Interactions (2310.07127).pdf	✓	2023/10/30 2:39
 [2023, Wan and Lu] Investigating VTubing as Reconstruction of Streamer Self-Presentation (wan2023vtubing).pdf	✓	2025/06/10 18:57

 **codex-nas** アプリ 06:00
TODO sync completed (54s)
flush=ok, touched=11
- Slack completion sync: added 0; already present 29
- Cosense completion sync: updated 0; already marked 5; skipped missing 0
- Slack live refresh: checked 0 newly completed item(s) from 4 candidate(s)
- Last refreshed: [2026-06-26](https://scrapbox.io/arcatdmz/2026-06-26)
- Active: 25
- Completion candidates: 0
- Checked / completed: 29 (編集済み)

 **codex-nas** アプリ 07:00
6月26日・過去の記録
- 2009年: xreality が発足。加えて、キャプチャ用クラスの整理やマルチタッチ関連の作業が進んでおり、実世界・拡張現実系の実装コミュニティ活動が動き始めていた。
- 2010年: プロトタイピングに関する発言整理を参照しており、制作・試作をどう研究や教育の言葉にするかへの関心が見える。
- 2011年: Google App Engine に触れながらコーディングを続けていた。Webサービス実装の手触りを得ている時期。
- 2013年: 「ポンチ絵」が仮完成。研究構想や提案を図としてまとめる作業が進んでいた。
- 2014年: HAI の査読を完了し、GCCF 査読や構成学関連の発表資料作成に取り組んでいた。研究コミュニティ運営と発信準備の日。

Build Your Own Agentic Assistant

- Add text labels to PDF file(s)
- Manage to-dos across services
- Visualize your paper submission history
- Organize your presentation history
- etc.

202210 LIVE 2022 2022-09 to 2022-12		Invited
202211 CHI 2023 2022-09 to 2023-04		Paper
202301 CHI勉強会 会議レポート 2023-01 to 2023-01		Article
202302 ヒューマンインタフェース学会学会誌 2023-02 to 2023-02		Article
202303 ACM (People of, XRDS) 2023-03 to 2023-03		Article
202304 SAS 2023 2023-04 to 2023-06		Paper
202306 MLSE 2023夏 2023-06 to 2023-06		Panel
202309 CHI 2024 2023-09 to 2024-05		Paper

Invited Talk Timeline

26 candidates / 2 upcoming / 22 accepted / 2 needs review / 2 money unknown / 8 gift tentative / 678 Cosense evidence / 304 file evidence / 10 Outlook evidence / generated 2026/6/27 23:00:49

Confidence

Timing

Review

Money

Gift Report

high

completed

accepted

none

unknown

seeded

2025-09-26

UIST 2025 Longitudinal AI workshop ミニ基調講演 UIST 2025 / 18 evidence

high

completed

accepted

none

unknown

2025-10-16

北海道大学招待講演 北海道大学 / 55 evidence

high

completed

needs-review

confirmed

unknown

情報処理学会 セミ

スレッド

Jun Kato (PI) 12分前

@codex-nas 今日10:25から東大の本郷キャンパスで講義。あざみ野から本郷三丁目まで移動。経路探してScrapboxの今日のページに書いておいて。

codex-nas アプリ 12分前

Codex on wakaba is working on it.

確認しました。今日(2026-07-01)のScrapbox/Cosense ページには、すでに経路が入っています。

<https://scrapbox.io/arcatdmz/2026-07-01>

経路は、あざみ野 09:26 発 → 本郷三丁目 10:20 着、54分、497円、乗換2回です。Yahoo!路線の結果 URL もページ内に記録されています。

B I U

返信する...

☐ 以下にも投稿する p-junkato

+ Aa ...

▶

Agentic Development of Your Agentic Assistant

Design Principles

- Rely on interpretive skills when needed
- Single Source of Truth
- Start small and turn repetitive tasks into CLIs
- Record as much as possible; it's like tending bonsai

Rely on Interpretive Skills When Needed

- Although Codex is very good at coding, it can also handle interpretive tasks, such as reading papers
- You can even run Codex on your NAS and have it respond to your requests whenever needed

```
• Searching the web
• Searched the web for site:dl.acm.org/doi 10.1145/306686.306690
• Searching the web
• Searched the web for "Incremental Formalization with the Hyper-Object Substrate"

• The extracted text already contains complete citation blocks for the four ACM papers. Web search is not returning reliable primary pages for all of them through the available search tool, so I'll use the PDF ACM Reference Format / DOI blocks as the primary metadata and record DOI URLs as external metadata where present. I'll inspect the method/results/conclusion portions before writing summaries.

• I have enough metadata. I'm reading result/discussion sections for the newer papers now so the summaries are interpretive rather than abstract-only.
```

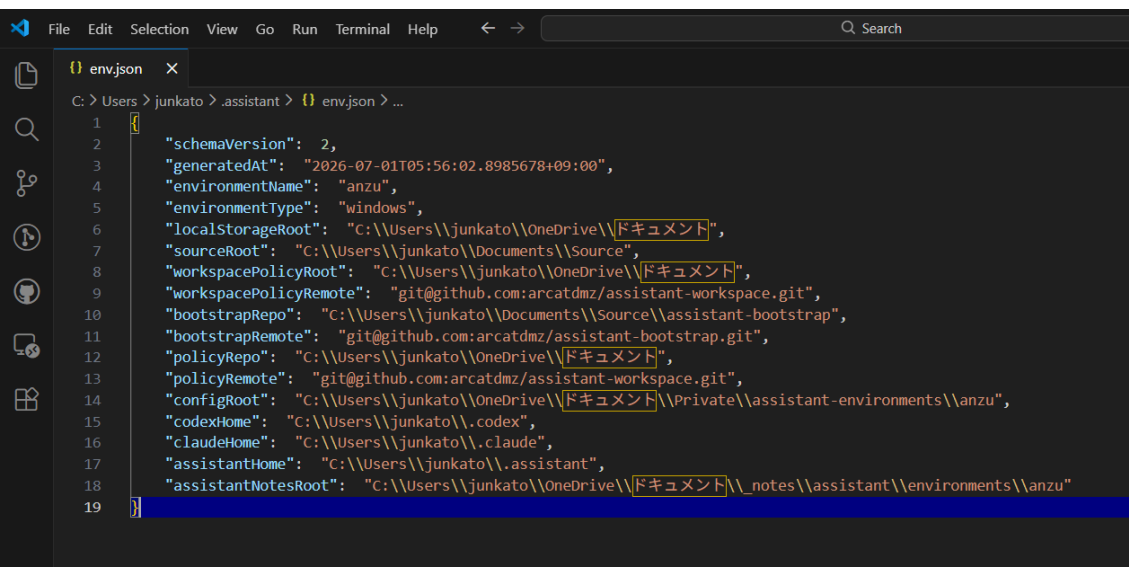
Single Source of Truth

- Store code on GitHub and data in the cloud, and treat them as your single source of truth
- This way, you can run multiple instances of your assistant across different machines and environments

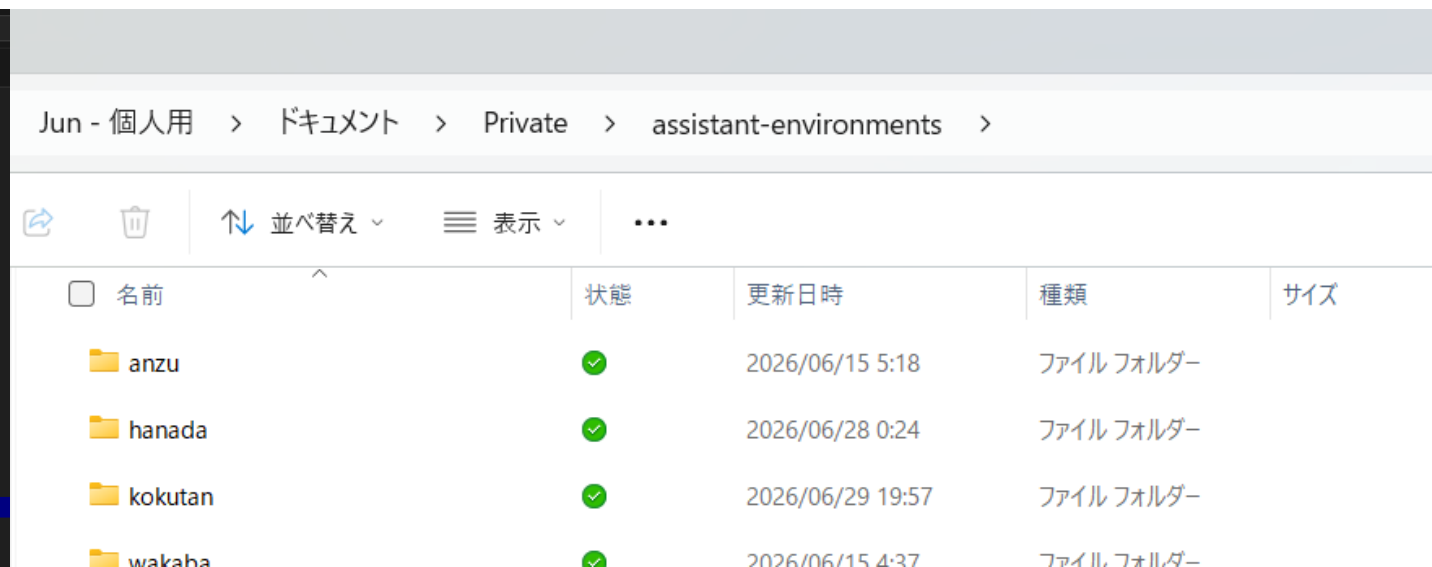


Single Source of Truth / Multiple Instances

- Create a bootstrap document and script
- Use discoverable paths (e.g., ~/.assistant) for config storage
- Use the machine name for machine-specific config paths



```
1 {
2   "schemaVersion": 2,
3   "generatedAt": "2026-07-01T05:56:02.8985678+09:00",
4   "environmentName": "anzu",
5   "environmentType": "windows",
6   "localStorageRoot": "C:\\Users\\junkato\\OneDrive\\ドキュメント",
7   "sourceRoot": "C:\\Users\\junkato\\Documents\\Source",
8   "workspacePolicyRoot": "C:\\Users\\junkato\\OneDrive\\ドキュメント",
9   "workspacePolicyRemote": "git@github.com:arcatdmz/assistant-workspace.git",
10  "bootstrapRepo": "C:\\Users\\junkato\\Documents\\Source\\assistant-bootstrap",
11  "bootstrapRemote": "git@github.com:arcatdmz/assistant-bootstrap.git",
12  "policyRepo": "C:\\Users\\junkato\\OneDrive\\ドキュメント",
13  "policyRemote": "git@github.com:arcatdmz/assistant-workspace.git",
14  "configRoot": "C:\\Users\\junkato\\OneDrive\\ドキュメント\\Private\\assistant-environments\\anzu",
15  "codexHome": "C:\\Users\\junkato\\.codex",
16  "claudeHome": "C:\\Users\\junkato\\.claude",
17  "assistantHome": "C:\\Users\\junkato\\.assistant",
18  "assistantNotesRoot": "C:\\Users\\junkato\\OneDrive\\ドキュメント\\_notes\\assistant\\environments\\anzu"
19 }
```



Jun - 個人用 > ドキュメント > Private > assistant-environments >				
📁 名前 状態 更新日時 種類 サイズ				
anzu	✓	2026/06/15 5:18	ファイル フォルダー	
hanada	✓	2026/06/28 0:24	ファイル フォルダー	
kokutan	✓	2026/06/29 19:57	ファイル フォルダー	
wakaba	✓	2026/06/15 4:37	ファイル フォルダー	

Start Small and Turn Repetitive Tasks into CLIs

- OpenAI Codex: `codex`
- Claude Code: `claude`
- You can ask anything you would ask in their web interfaces
- "How can I get to Hongo-sanchome Station from ...?"
- When you notice that you are asking the same kind of question or performing the same kind of task repeatedly, turn it into a reproducible command-line interface (CLI)

Record as Much as Possible; It's Like Tending Bonsai

- Record your activities for later reuse
- Record agents' notes for transparency and reproducibility
- Ask agents not only to visualize data, but also to implement UIs for verification and modification

202210 LIVE 2022

LIVE 2022 keynote / talk workspace; official LIVE page lists Jun Kato as organizer; no direct public paper entry matched

202210-live-2022

Invited

202211 CHI 2023

Matched: `chi2023-kato-lyric-app-framework` and `chi2023-kato-sigccc`, CHI '23

202211-chi-2023

Paper

202208 SIGGRAPH 2022

Talk / slide workspace; no direct public entry matched

202208-siggraph-2022

Talk

202203 UIST 2022

Source

Search source records

Overleaf

uist2026, qualitative, repository



Linked source records

8997 / Lyric App Framework: A Web-based Framework for Developing Interactive Lyric-driven Musical Applications / CHI 2023 Papers / Accepted
PCS / 8997



2030 / Griffith: A Storyboarding Tool ... / CHI 2024 Papers / Accepted
PCS / 2030



Agentic Development of Your Agentic Assistant

Design Principles

- Rely on interpretive skills when needed
- Single Source of Truth
- Start small and turn repetitive tasks into CLIs
- Record as much as possible; it's like tending bonsai

Build Your Own Assistant!

DevOps for research: agenda

- Development environment setup
- Continuous integration/delivery
- Web technologies for development/public relations
- Agentic development of your agentic secretary

DevOps for research