

Silicon Sandbox: Mastering Mac virtualisation for Jamf workflows



Rob Potvin
Senior Consulting Engineer
@Jamf

Recipe

Preheating the **Oven**

Choosing Our **Ingredients**

Measuring and **Prepping**

Mixing the **Batter**

Baking and **Sharing**

The Taste **Test**

Serving **Suggestions**



Preheating the Oven

Warming Up with Apple's Virtualisation Framework

What is this Virtual Machine

- A fully functional macOS environment running inside another macOS machine (host) on Apple Silicon
- VMs are created and launched using the Apple Virtualisation Framework
- Free, built into the OS

Apple silicon

macOS

.....

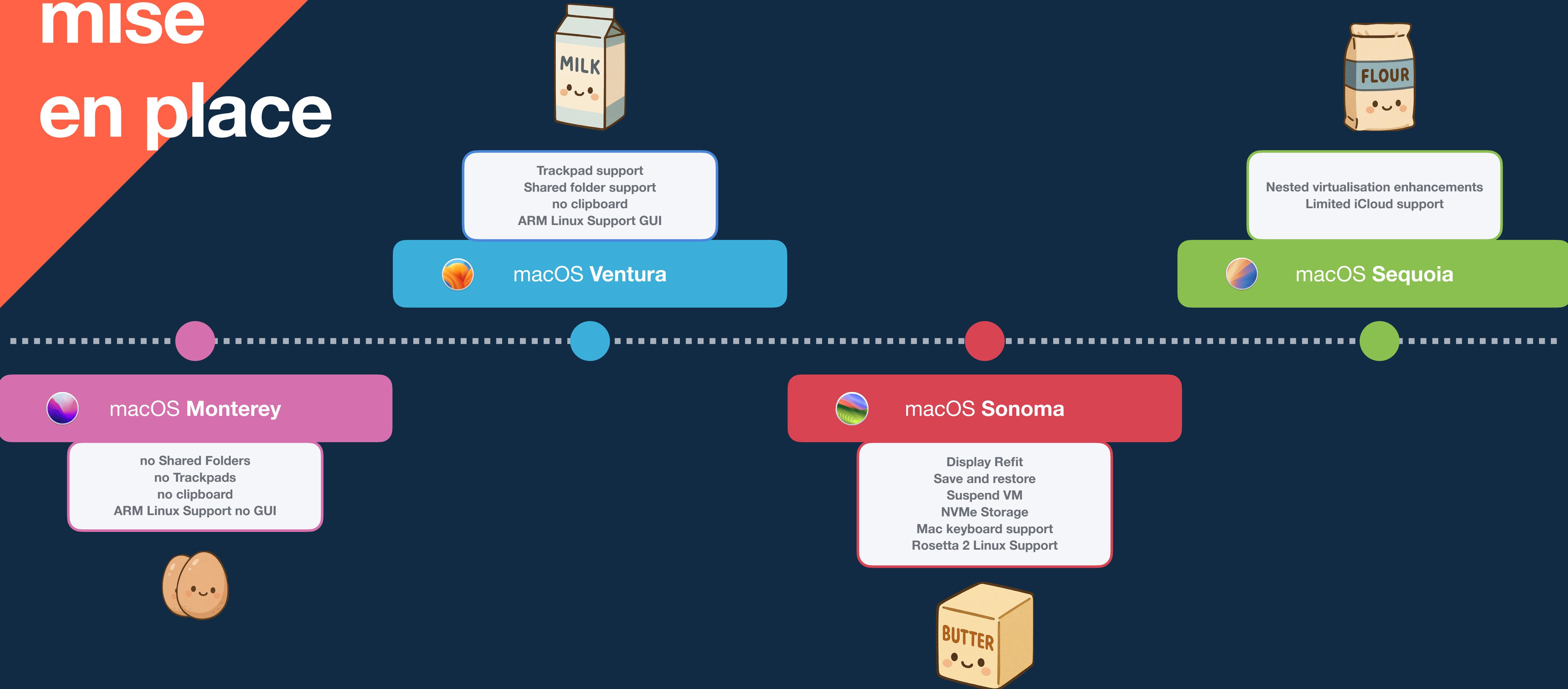
Virtualisation framework

Hypervisor framework

macOS kernel

Hardware

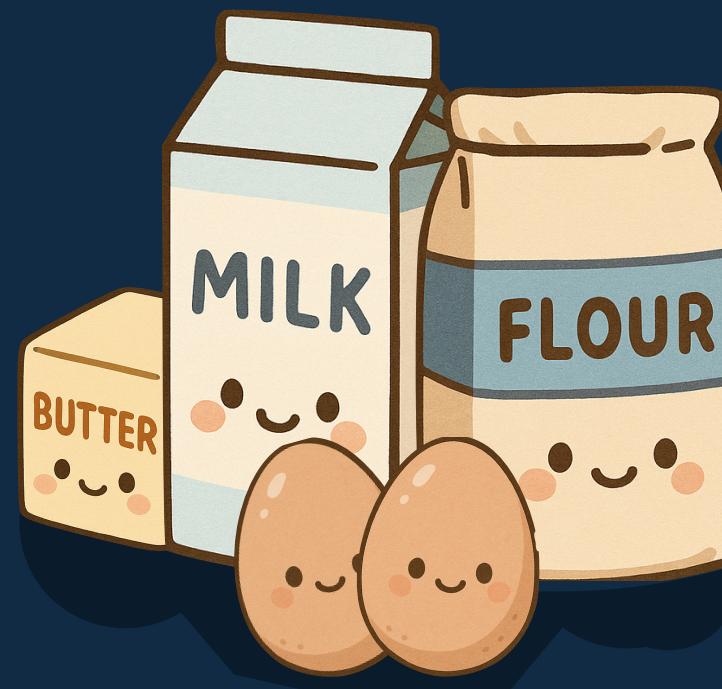
mise en place



Half-Baked Recipe: What's Missing from the Mix

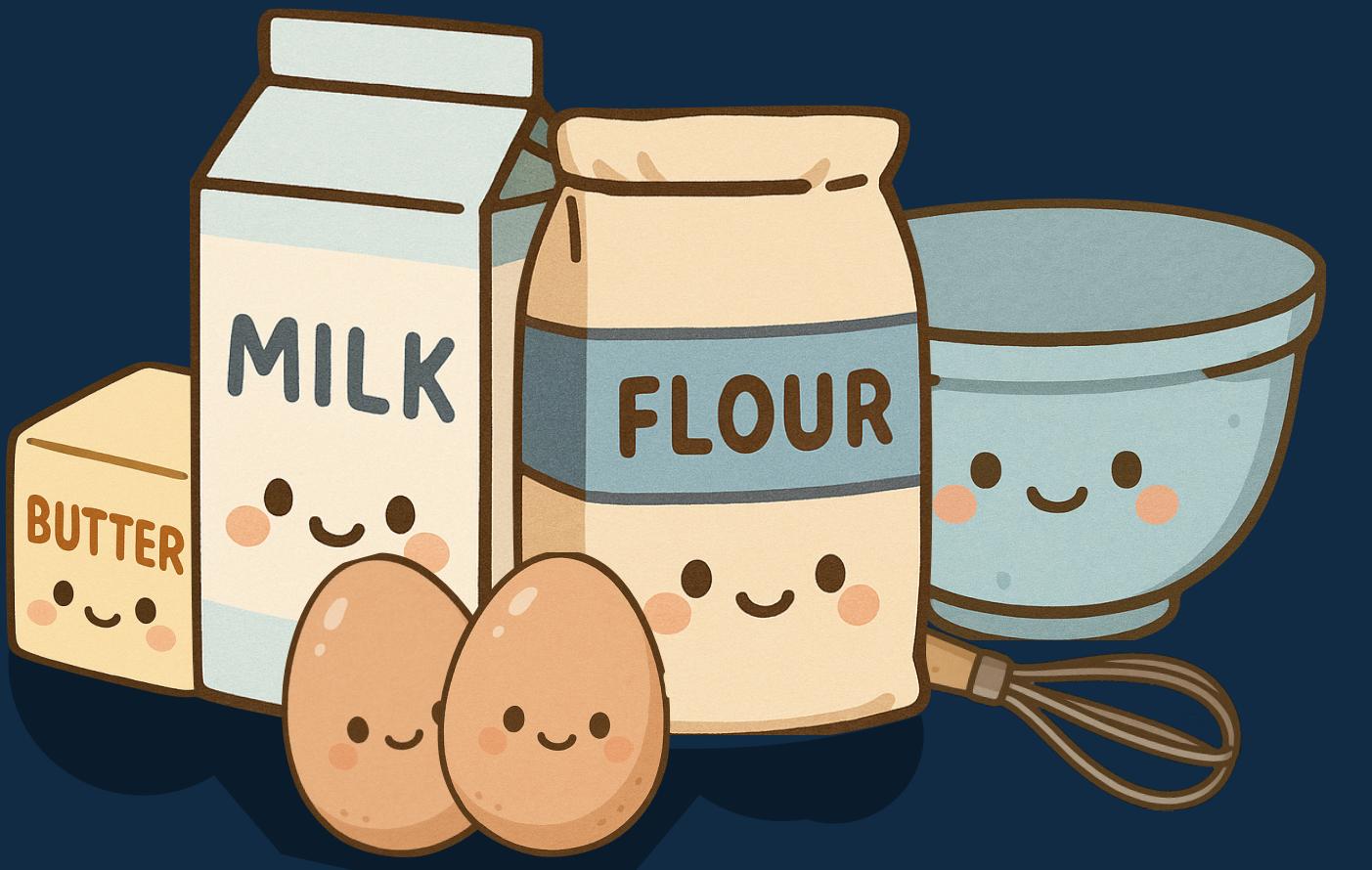
- No Appstore
- No Apps & Books
- No TouchID
- No Automated Enrollment
- Limited iCloud Support*
- Limited Mouse / Trackpad / Keyboard





Choosing Our Ingredients

Virtualisation Pantry



jamf



VirtualBuddy

VirtualBuddy can virtualize macOS 12 and later on Apple Silicon, with the goal of offering features that are useful to developers who need to test their apps on multiple versions of macOS.



Viable

*Viable uses lightweight virtualisation in macOS Monterey 12.4 or later to build, install and run macOS virtual machines (VMs) on Apple silicon Macs. Has a cool little sister app **Vimy** with a linux variant called*



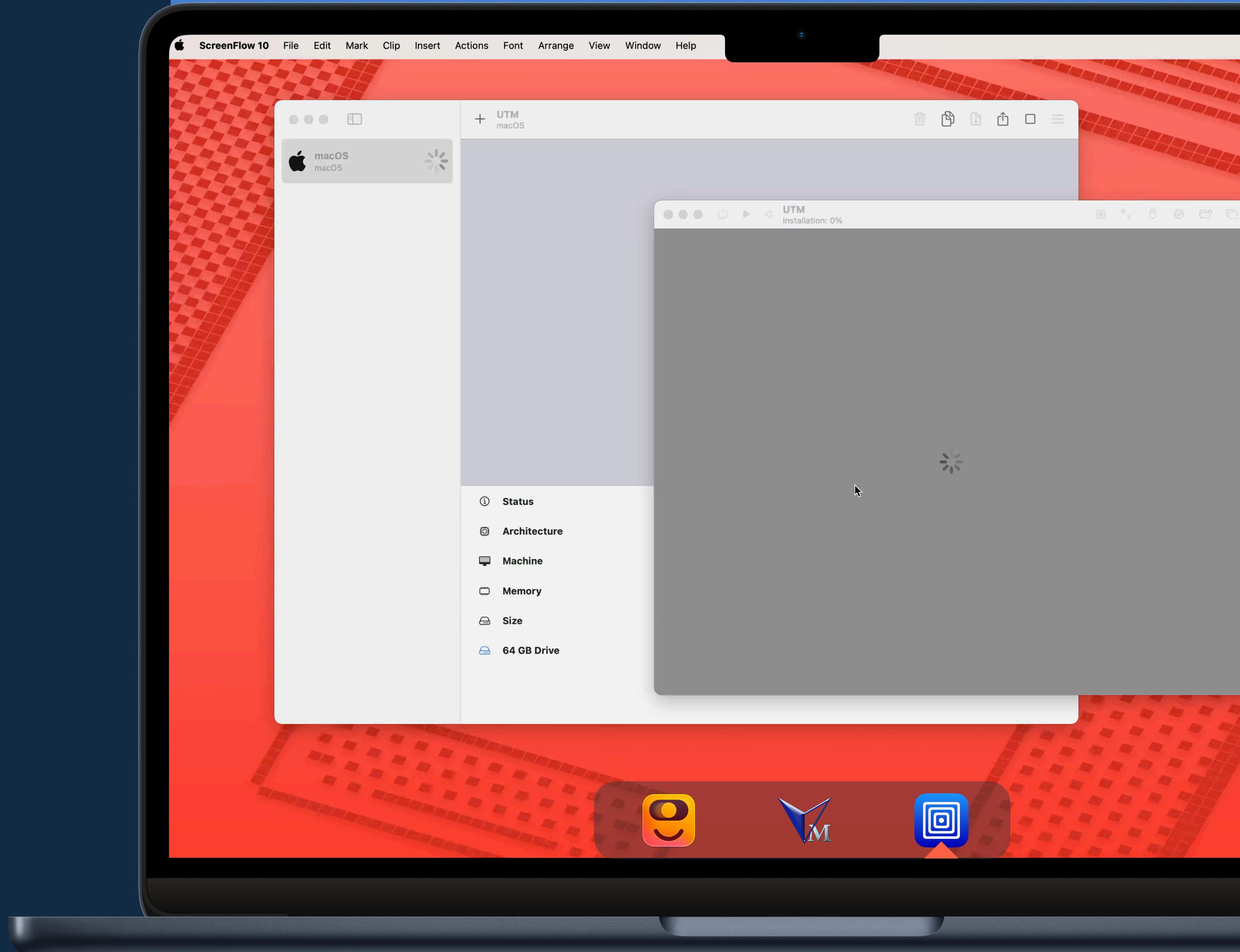
UTM

UTM is a full featured system emulator and virtual machine host for iOS and macOS. It is based off of QEMU. In short, it allows you to run Windows, Linux, and more on your Mac, iPhone, and iPad.



Building a VM

- ▶ Install from **IPSW**
- ▶ Run through **Setup Assistant**
- ▶ **Enroll Device**
- ▶ **Use once and delete**





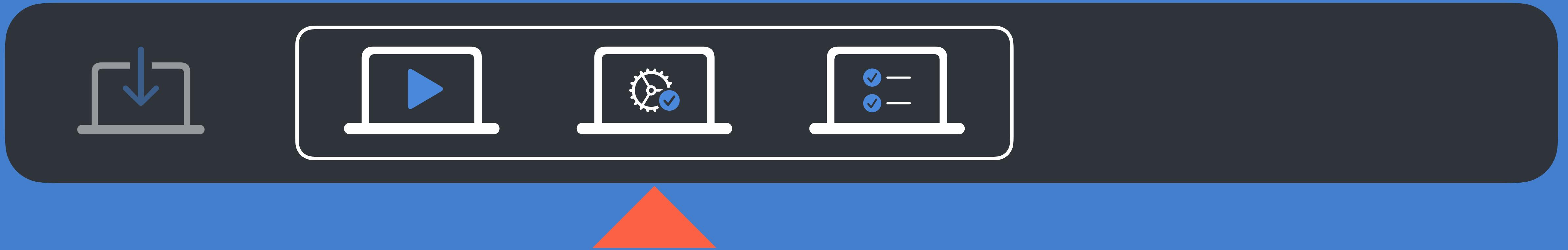
Traditional VM Managers

Where 'instant deployment' means 'grab a coffee first', click through the setup assistant. Many ingredients, many steps and then and only then will you be ready for enrollment.

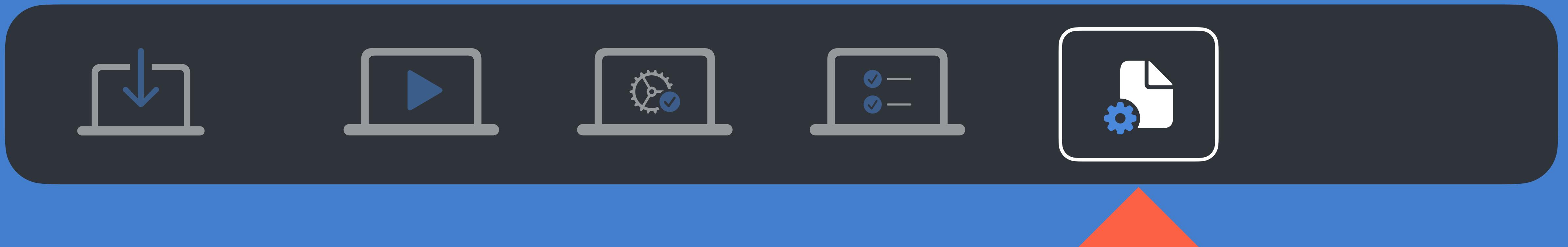
Restore



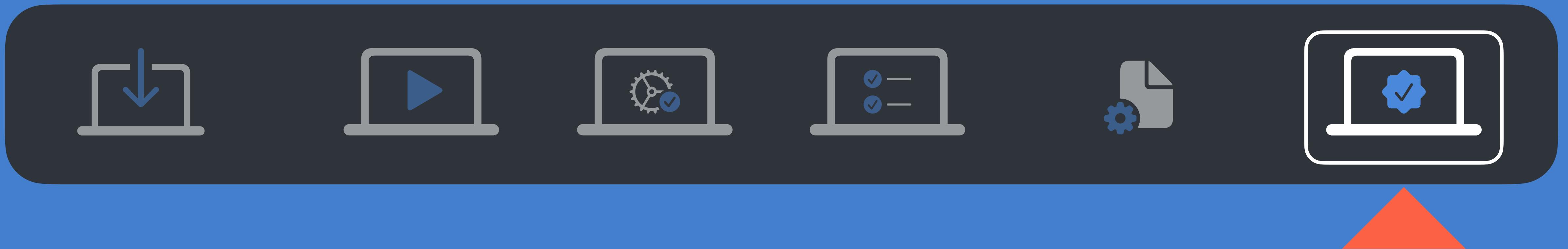
Apple Setup Assistant



Enrollment



Testing





tart

Tart is like your ingredients pre-mixed for virtualisation-think of it as your secret recipe for baking up quick, fresh, isolated macOS environments right from the command line.



Measuring Getting Baking:



Tart Bake-Off

Pre-Mixed, Ready in Minutes

- Command Line Tool
- Tart.app for output
- Install via homebrew
 - `brew install cirruslabs/cli/tart`
- `tart.run`



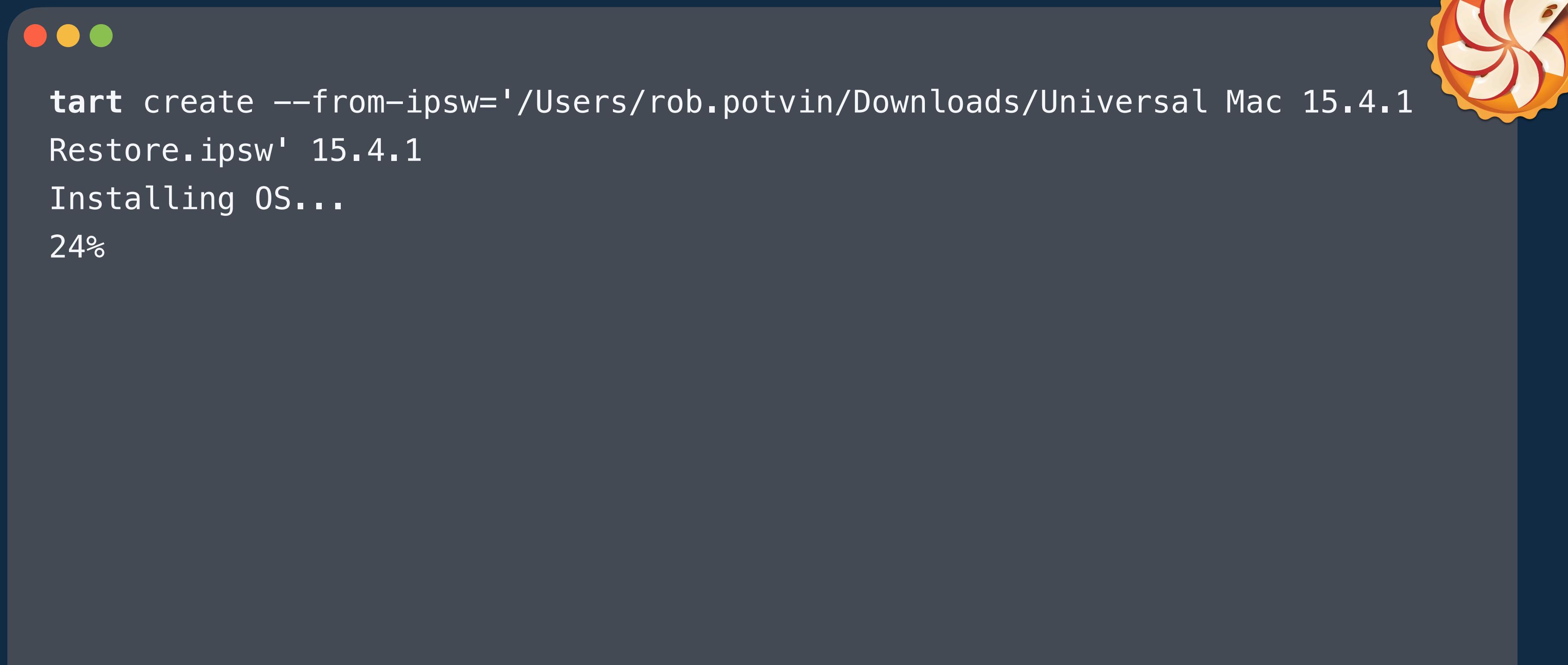
Tart Ingredients

Part cmd, part app

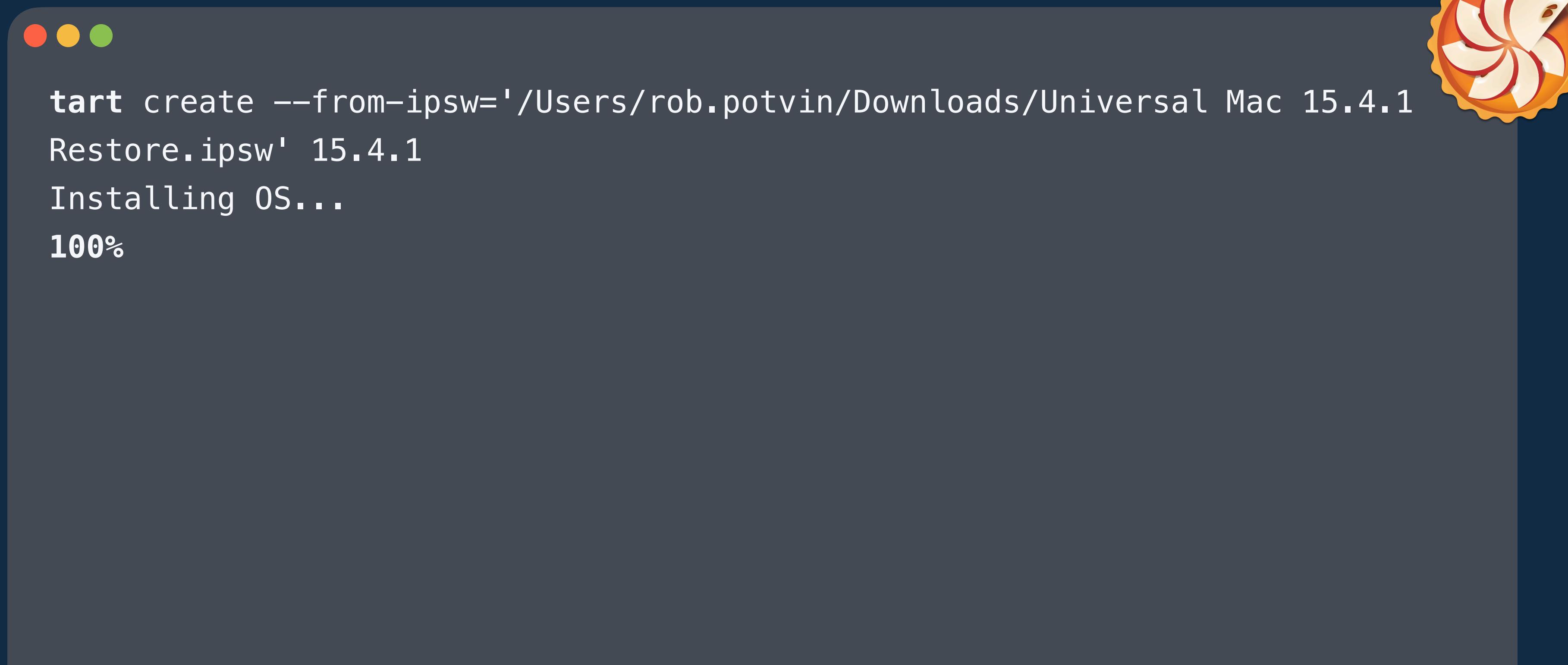
- Build from IPSW
- Easy Clone / Easy Import / Export
- Pull / Push VM images to / from OCI registry



Measuring and Prepping Getting to Know Tart



Measuring and Prepping Getting to Know Tart



Measuring and Prepping Getting to Know Tart

```
tart list
Source Name          Disk Size SizeOnDisk State
local   15.4.1
                    50    19      19      stopped
```



Measuring and Prepping Getting to Know Tart



```
tart tlist 15.4.1 newtest
Source Name          Disk Size SizeOnDisk State
local   15.4.1
                                50    19     19      stopped
local   newtest
                                50    19     19      stopped
```

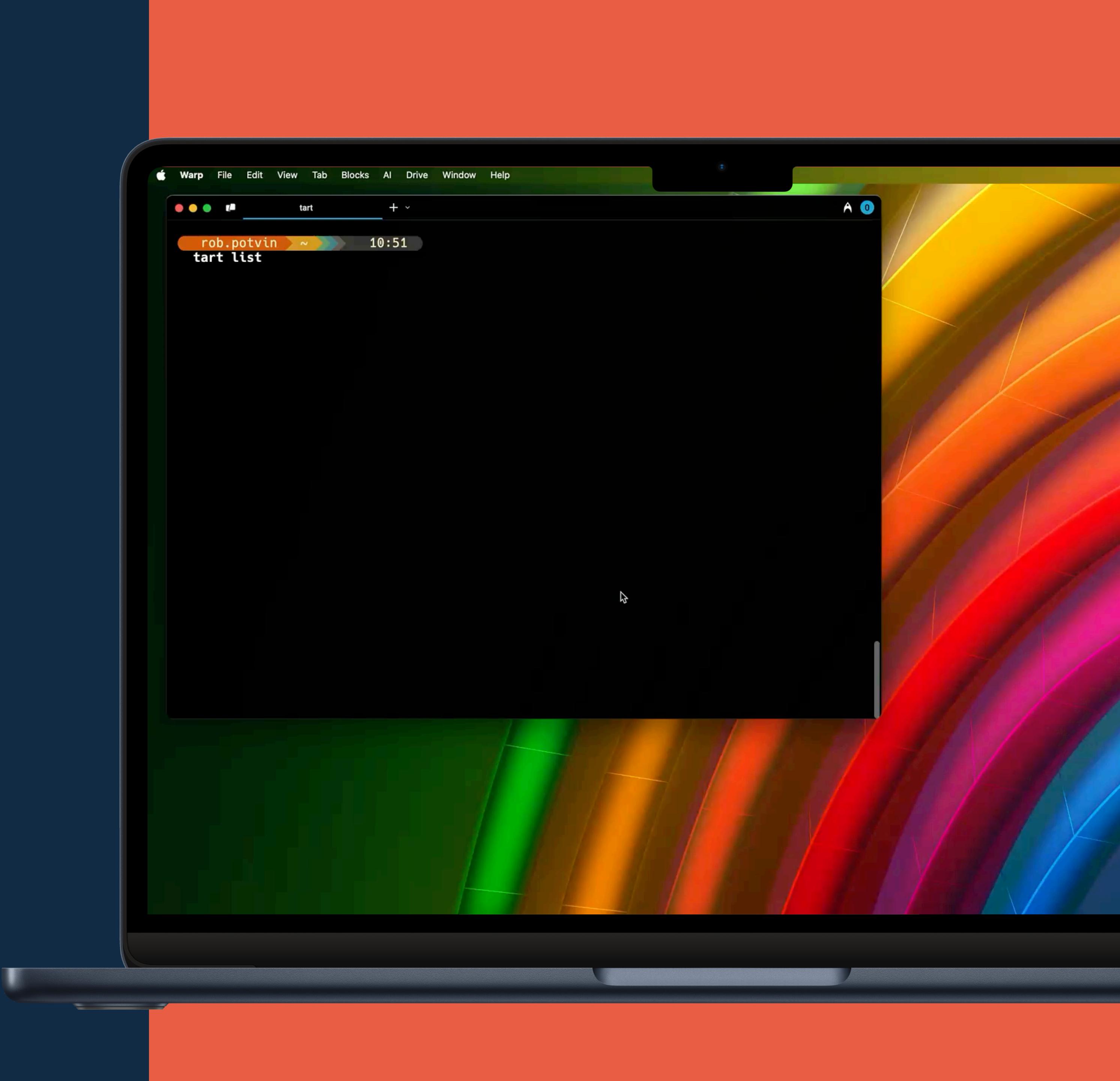
Great but where is the Instant Cake Mix?

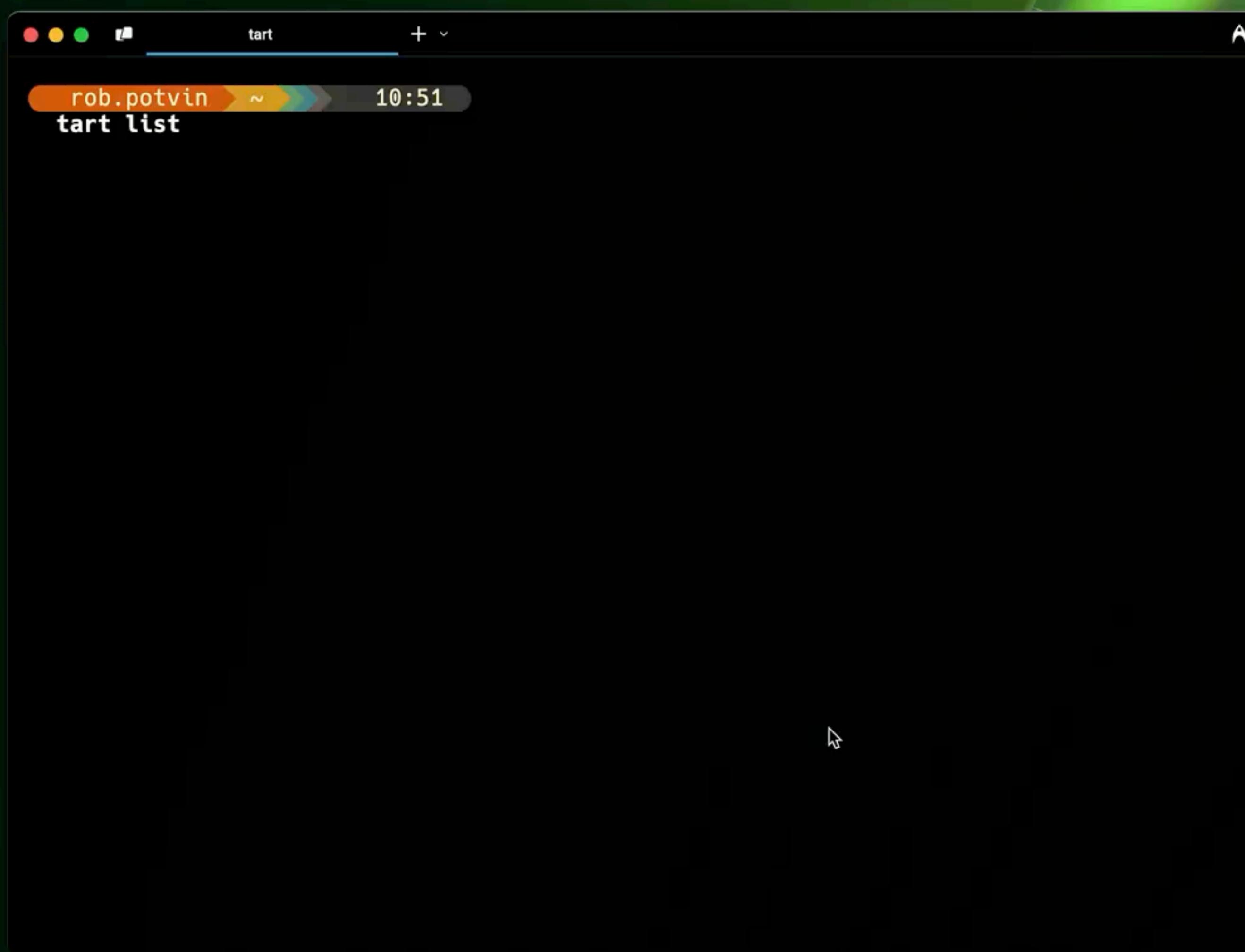
```
tart pull --quiet --no-progress --no-skip-existing newvmtest
  Pulling image: ghcr.io/cirruslabs/macos-sequoia-vanilla:latest...
  Disk Size: 21.3 GB
  Locally available: 0 B
  Locally available on disk: 0 B
  State: stopped
  OCI manifest: ghcr.io/cirruslabs/macos-sequoia-vanilla:latest
```



New Clone Running in tart

- Instant cake mix is **instant**
- Using the **tart set command**
- **--display-refit**
- Launches in 20-30 seconds
- Simple clone from the image downloaded from the **OCI server**





What do you think about Our Instant Cake Mix?

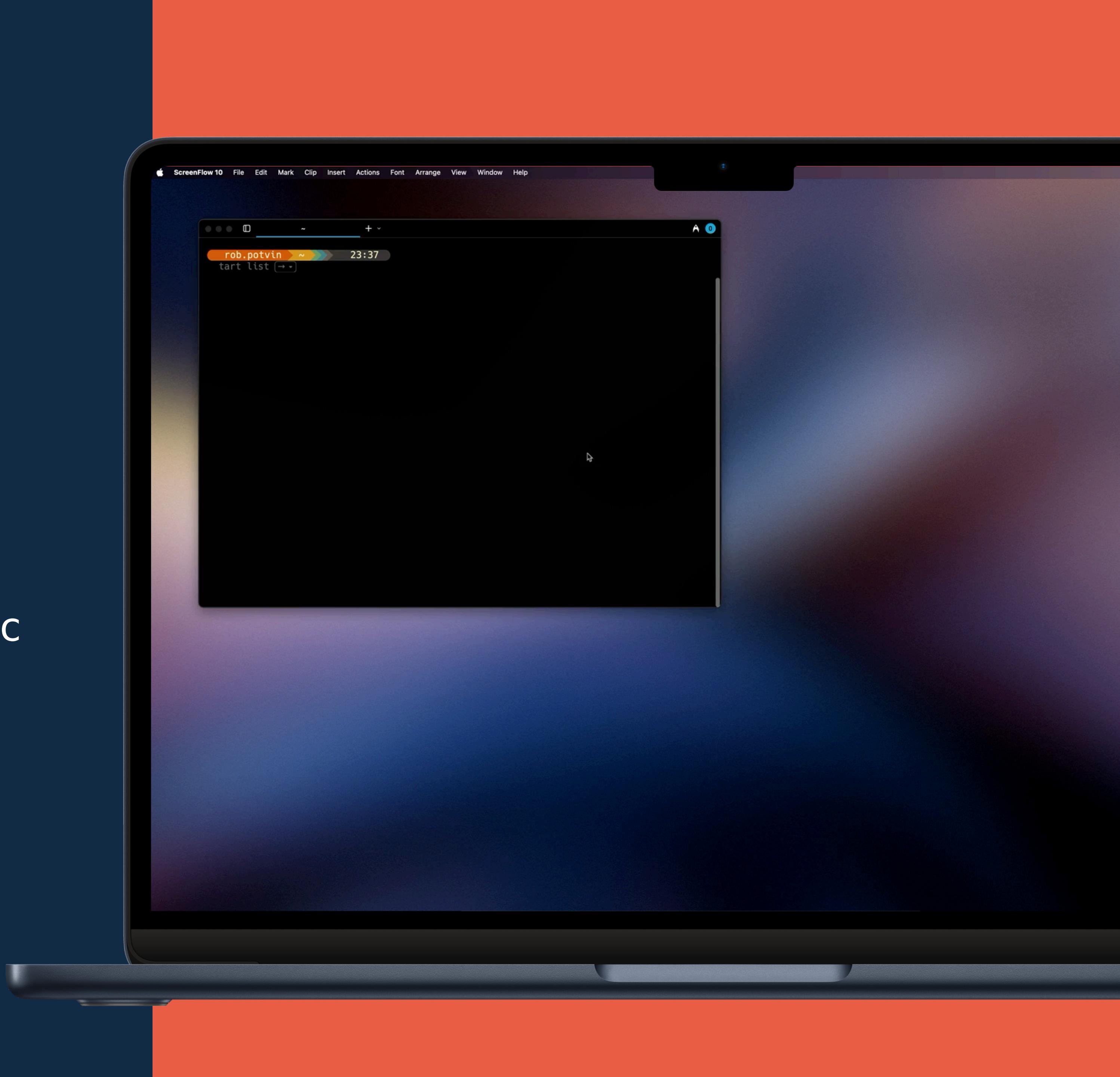
- One OCI macOS image hosted
- Many clones, many enrollments
- Easy to create new serial numbers
 - `tart set --random-serial --random-mac`



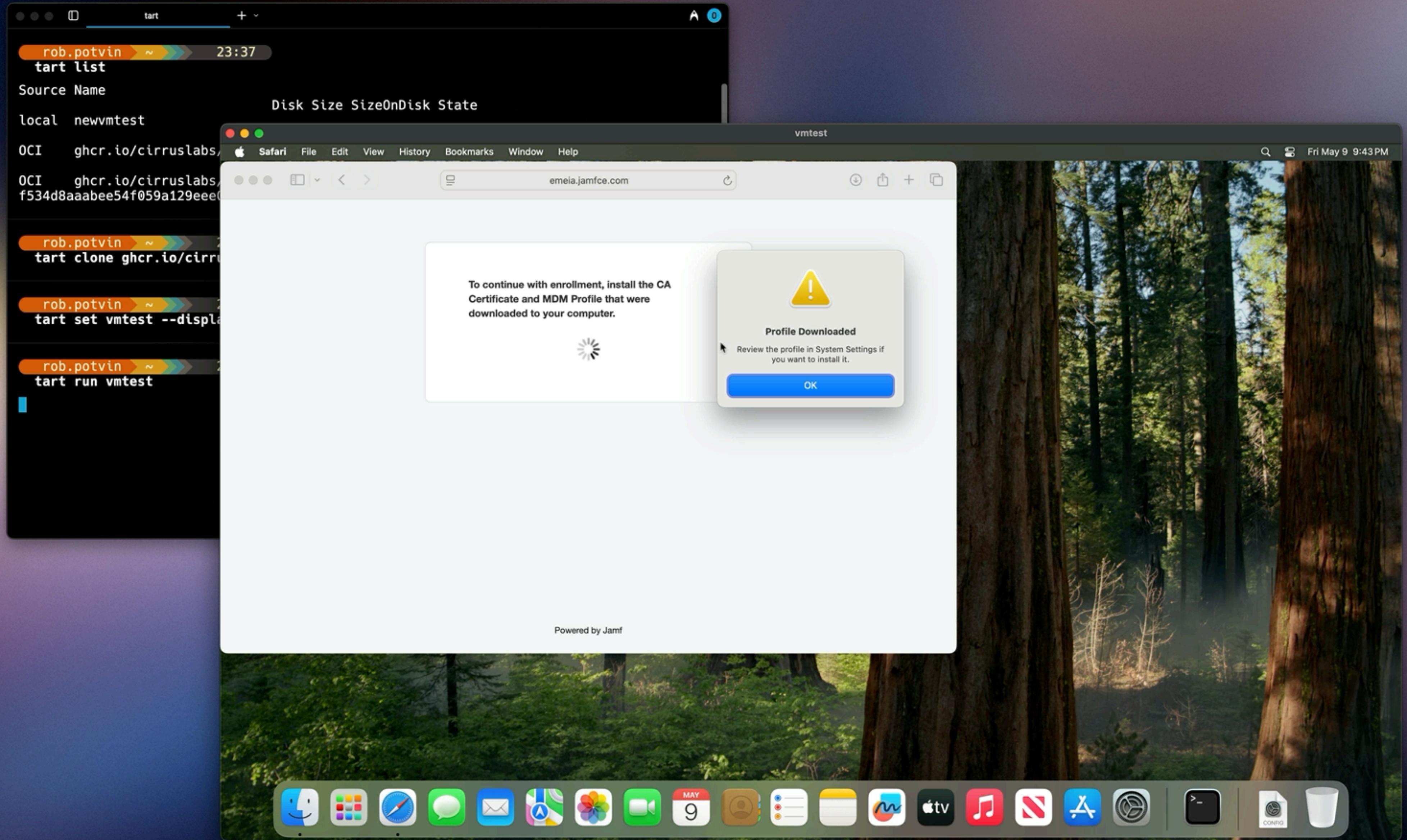
Enroll the VM

Up and running

- ▶ Cloning a new VM
- ▶ Using the **tart set** command
- ▶ **--random-serial --random-mac**
- ▶ **Enrollment** with User Initiated Enrollment

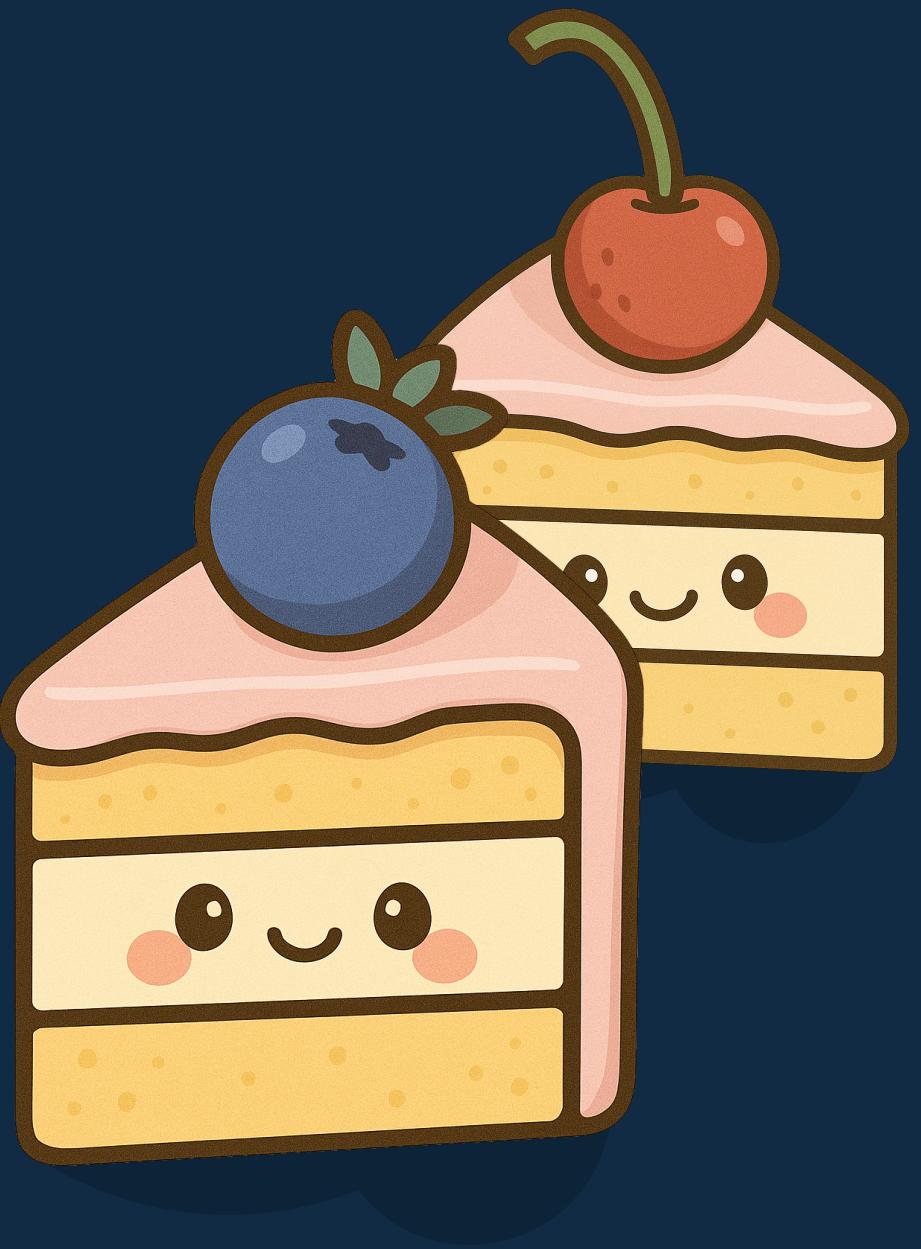






Enrolled the VM Could it be faster?

- Clone from OCI
- Easy to create new unique VMs for enrollment
- `tart set --random-serial --random-mac`
- Enrollment is a bit slow
- Could it be custom?





Mixing the Batter: Crafting Custom Images with Packer

Key Ingredient **Packer, just a tablespoon**

- Packer lets you create identical machine images from a single source template.
- Used a lot in CI/DI
- Easy install with homebrew
 - `brew tap hashicorp/tap`
 - `brew install hashicorp/tap/packer`
- Simple template to create a macOS VM



github.com

Product Solutions Resources Open Source Enterprise Pricing

Search or jump to... / Sign in Sign up

cirruslabs/macos-image-templates Public

Code Issues 2 Pull requests Discussions Actions Projects Security Insights

main 31 Branches 32 Tags Go to file Code About

No description or website provided.

fkorotkov Bring back iOS 17.5 ✓ 6f66644 · 2 days ago 398 Commits

.ci Bring back iOS 17.5 2 days ago

ansible Ignore errors for "install available update" Ansible task (#...) 3 months ago

data Bring back iOS 17.5 2 days ago

scripts Enable UI automation by default (#137) last year

templates Xcode 16.4-beta (#233) 3 days ago

.cirrus.star Updated runners push logic 3 days ago

.cirrus.yml Refactored release scripts 7 months ago

.gitignore install anka builder script 7 years ago

DEVELOPMENT.md Xcode 15.4 (#154) last year

LICENSE Initial commit 7 years ago

README.md swich to Xcode 16.3 release (#225) last month

Readme MIT license

Activity Custom properties 252 stars 12 watching 70 forks Report repository

Releases 32 Xcode 16.4 beta 1 Latest 3 days ago + 31 releases

Packages 13 macos-ventura-xcode

Personal > JS

Full Jamf Pro > 1

Dashboard

Version 11.15.2-t1743086653526

Inventory Information

Computers	Devices
6 Managed	1 Unmanaged
7 Managed	1 Unmanaged

Smart Computer Groups

Apple Silicon - VMs
3 Computers

Smart Mobile Device Groups

Jamf Trust Workflow
0 Devices

Policy Statuses

experiencejamf Install Excel Install Outlook Install Install Word

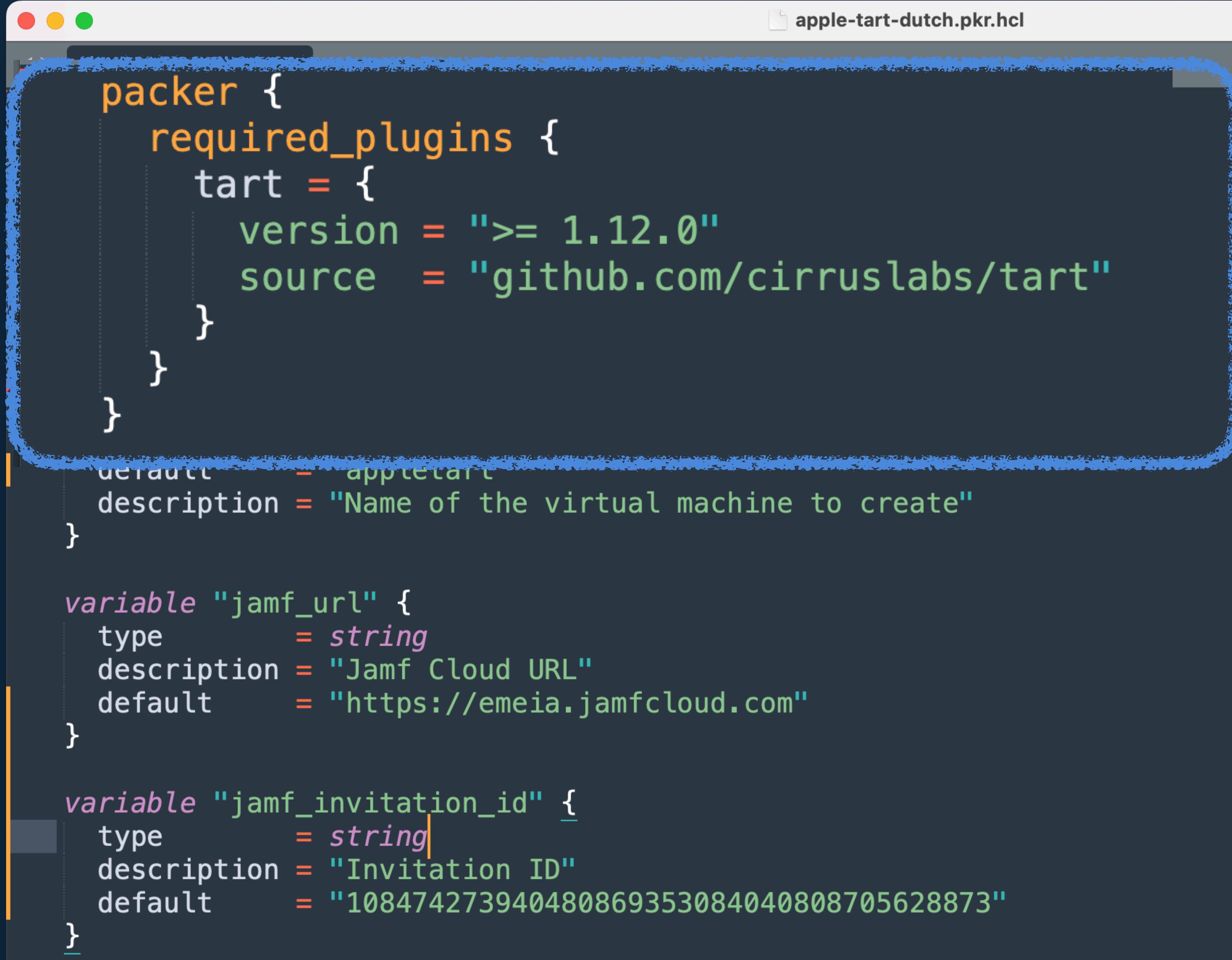
The screenshot shows the Jamf Pro dashboard interface. On the left is a sidebar with navigation links: Dashboard (selected), Computers, Devices, Users, Blueprints, Compliance, Compliance readiness, and Settings. The main content area is titled 'Dashboard' and displays 'Inventory Information' with two cards: one for 'Computers' (6 Managed, 1 Unmanaged) and one for 'Devices' (7 Managed, 1 Unmanaged). Below this is a section for 'Smart Computer Groups' showing 'Apple Silicon - VMs' with 3 computers. There's also a section for 'Smart Mobile Device Groups' showing 'Jamf Trust Workflow' with 0 devices. At the bottom, there's a 'Policy Statuses' section with five cards: 'experiencejamf', 'Install Excel', 'Install Outlook', 'Install', and 'Install Word'. The top bar includes standard Mac OS X icons for window control, file operations, and system status.

Enrollment Ingredient Prepping Jamf Pro



- Create an enrollment invitation in Jamf Pro
- Set the limits on that invitation
- Add to enrollment URL - right to MDM
 - [https://name.jamfcloud.com/enroll?invitation=\[id\]](https://name.jamfcloud.com/enroll?invitation=[id])
- Keep you ID safe

Packer Build



The screenshot shows a terminal window with the title bar "apple-tart-dutch.pkr.hcl". The window contains the following Packer build configuration:

```
packer {
  required_plugins {
    tart = {
      version = ">= 1.12.0"
      source  = "github.com/cirruslabs/tart"
    }
  }
  default = "apple-tart"
  description = "Name of the virtual machine to create"
}

variable "jamf_url" {
  type      = string
  description = "Jamf Cloud URL"
  default    = "https://emeia.jamfcloud.com"
}

variable "jamf_invitation_id" {
  type      = string
  description = "Invitation ID"
  default    = "108474273940480869353084040808705628873"
}
```

Packer Build



```
variable "jamf_url" {
  type      = string
  description = "Jamf Cloud URL"
  default    = "https://emeia.jamfcloud.com"
}

variable "jamf_invitation_id" {
  type      = string
  description = "Invitation ID"
  default    = "108474273940480869353084040808705628873"
}

variable "insw_url" {
```

A screenshot of a terminal window titled "apple-tart-dutch.pkr.hcl". The window displays a portion of a Packer configuration file. It includes a "packer" block with a "required_plugins" block containing a "tart" plugin entry. Below this, there are two "variable" blocks: one for "jamf_url" and one for "jamf_invitation_id". The "jamf_invitation_id" variable has a very long, randomly generated value assigned to its "default" field.

Packer Build



```
source "tart-cli" "tart" {
    from_ipsw      = var.ipsw_url
    vm_name        = var.vm_name
    cpu_count      = 4
    memory_gb      = 8
    disk_size_gb   = 50
    ssh_password   = "admin"
    ssh_username   = "admin"
    ssh_timeout    = "300s"
    boot_command   = [
        # hello, hola, bonjour, etc.
        # to the "English (US)", which is not what we want. To solve this, we switch to some other
        # language first, e.g. "Italiano" and then switch back to "English". We'll then jump to the
        # first entry in a list of "english"-prefixed items, which will be "English".
        #
        # [1]: should be named "English (US)", but oh well 🤷
        "<wait30s>italiano<esc>english<enter>",
        # Select Your Country or Region
        "<wait30s>united states<leftShiftOn><tab><leftShiftOff><spacebar>",
        # Transfer Your Data to This Mac
        "<wait10s><tab><tab><tab><spacebar><tab><tab><spacebar>",
        # Written and Spoken Languages
        "<wait10s><leftShiftOn><tab><leftShiftOff><spacebar>",
        # Accessibility
        "<wait10s><leftShiftOn><tab><leftShiftOff><spacebar>",
        # Data & Privacy
        "<wait10s><leftShiftOn><tab><leftShiftOff><spacebar>",
        # Create a Mac Account
        "<wait10s>Managed via Tart<tab>admin<tab>admin<tab>admin<tab><spacebar><tab><tab><spacebar>",
        # Enable Voice Over
        "<wait120s><leftAltOn><f5><leftAltOff>",
        # Sign In with Your Apple ID
    ]
}
```

Packer Build



```
apple-tart-enrollment-url.pkr.hcl
```

```
UNREGISTERED
```

```
source "tart-cli" "tart" {  
    from_ipsw      = var.ipsw_url  
    vm_name        = var.vm_name  
    cpu_count      = 4  
    memory_gb      = 8  
    disk_size_gb   = 50  
    ssh_password   = "admin"  
    ssh_username   = "admin"  
    ssh_timeout    = "300s"  
    boot_command   = [  
        # hello, hola, bonjour, etc.  
        "<wait60s><spacebar>",  
        # Language: most of the times we have a list of "English"[1], "English (UK)", e  
        # "English" language already selected. If we type "english", it'll cause us to  
        # to the "English (UK)", which is not what we want. To solve this, we switch to  
        # language first, e.g. "Italiano" and then switch back to "English". We'll then  
        # first entry in a list of "english"-prefixed items, which will be "English".  
        #  
        # [1]: should be named "English (US)", but oh well 🙃  
        "<wait30s>italiano<esc>english<enter>",  
        # Select Your Country or Region  
        "<wait30s>united states<leftShiftOn><tab><leftShiftOff><spacebar>",  
        # Transfer Your Data to This Mac  
        #<wait10s>Managed via Tart<tab>admin<tab>admin<tab>admin<tab><spacebar><tab><spacebar>  
        # Enable Voice Over  
        "<wait120s><leftAltOn><f5><leftAltOff>",  
        # Sign In with Your Apple ID  
    ]  
}
```

```
INSERT MODE, Line 1, Column 1
```

```
main
```

```
Spaces: 2
```

```
HCL
```

Packer Build



```
# Enable passwordless sudo, auto-login, disable screensaver, prevent sleep, enable safari au
provisioner "shell" {
  inline = [
    "set -euxo pipefail",
    // Enable passwordless sudo
    "echo admin | sudo -S sh -c \"mkdir -p /etc/sudoers.d/; echo 'admin ALL=(ALL) NOPASSWD: /bin/login /sbin/reboot' > /etc/sudoers.d/admin_login_reboot\"",
    // Enable auto-login
    //
    // See https://github.com/xfreebird/kcpassword for details.
    "echo '00000000: 1ced 3f4a bcba ba2c caca 4e82' | sudo xxd -r - /etc/kcpassword",
    "sudo defaults write /Library/Preferences/com.apple.loginwindow autoLoginUser admin",
    // Disable screensaver at login screen
    "sudo defaults write /Library/Preferences/com.apple.screensaver loginWindowIdleTime 0",
    // Disable screensaver for admin user
    "defaults -currentHost write com.apple.screensaver idleTime 0",
    // Prevent the VM from sleeping
    "sudo systemsetup -setsleep Off 2>/dev/null".
      SAFARI_PID=$!
      "disown",
      "sleep 30",
      "kill -9 $SAFARI_PID",
      // Enable Safari's remote automation
      "sudo safaridriver --enable".
```

```
// A (hopefully) temporary workaround for Virtualization.Framework's
// installation process not fully finishing in a timely manner
create_grace_time = "30s"

// Keep the recovery partition, otherwise it's not possible to "softwareupdate"
recovery_partition = "keep"
}
```

```
apple-tart-enrollment-url.pkr.hcl
```

```
UNREGISTERED
```

```
+ ▾
```

```
INSERT MODE, Line 1, Column 1
```

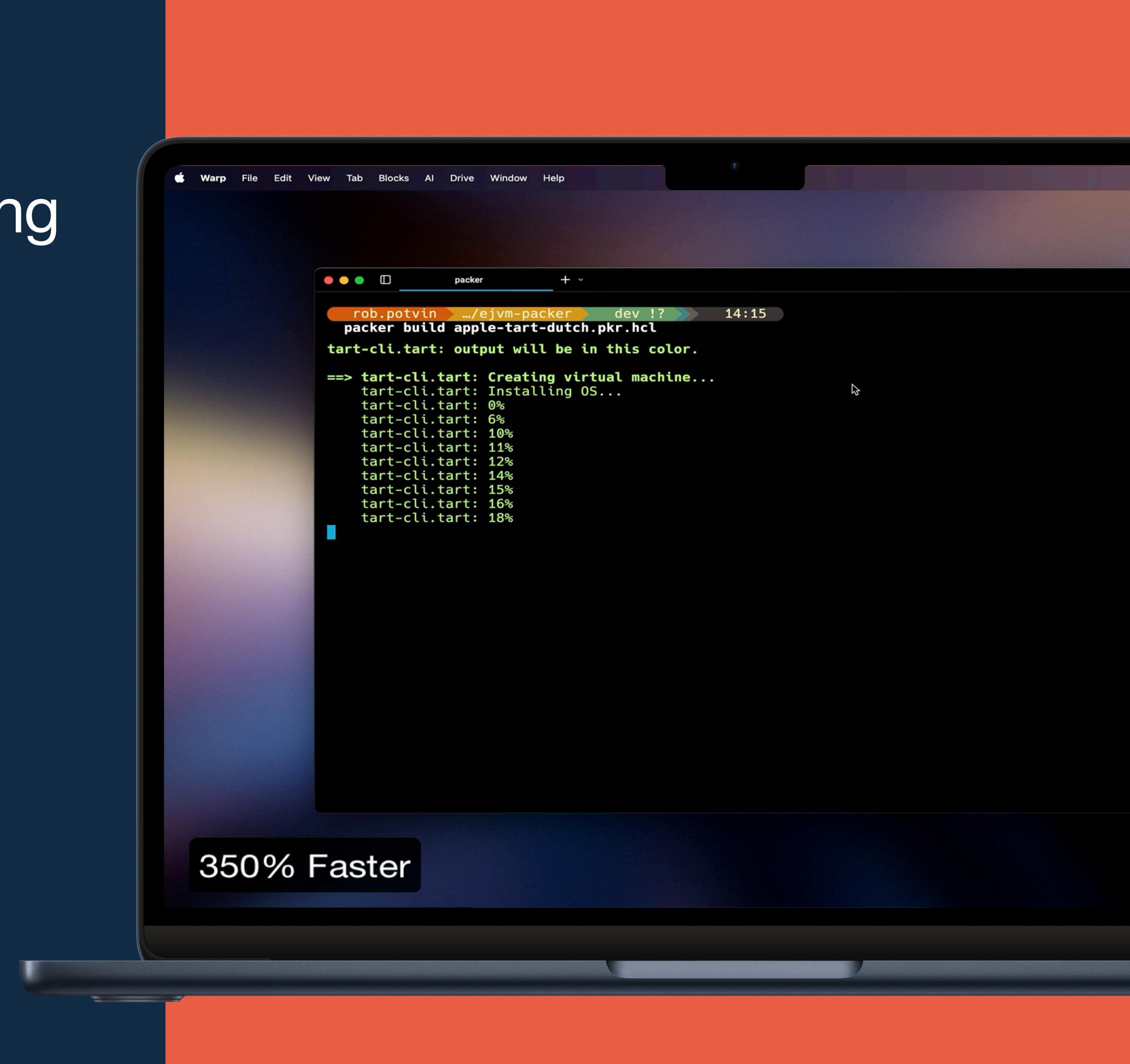
Packer Build



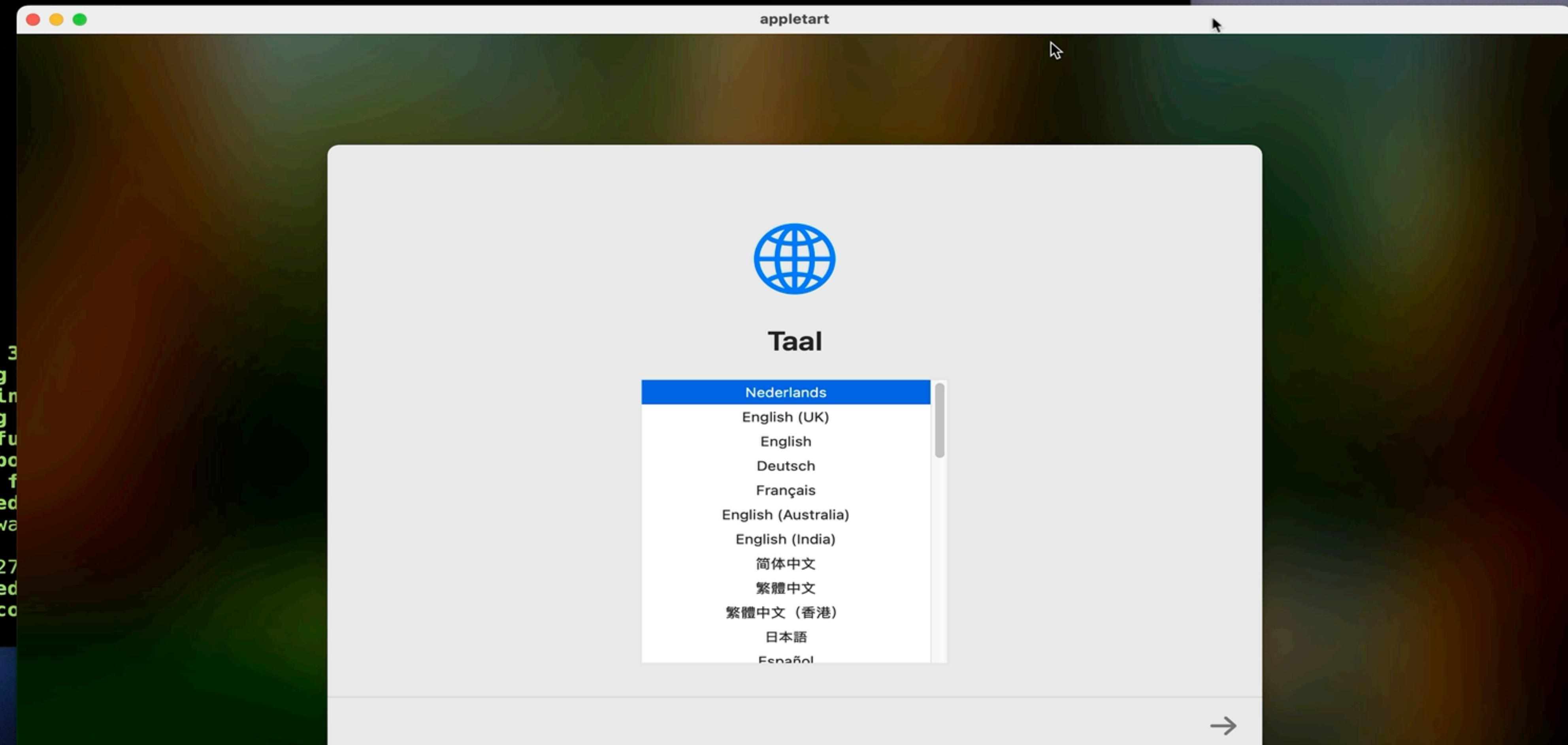
```
# Create a webloc file on the desktop for Jamf Pro enrollment
provisioner "shell" {
  inline = [
    "set -euxo pipefail",
    "cat << EOF > ~/Desktop/Enroll_Your_Mac.webloc",
    "<?xml version=\"1.0\" encoding=\"UTF-8\"?>",
    "<plist version=\"1.0\">",
    "<dict>",
    "  <key>URL</key>",
    "  <string>${var.jamf_url}/enroll?invitation=${var.jamf_invitation_id}</string>",
    "</dict>",
    "</plist>",
    "EOF",
  ]
}
```

Not-Live Cooking: Baking the Packer Template

- ▶ packer / tart installed
- ▶ packer init command
- ▶ packer fmt command
- ▶ packer validate command
- ▶ packer build command



```
● packer + ~  
tart-cli.tart: 40%  
tart-cli.tart: 42%  
tart-cli.tart: 43%  
tart-cli.tart: 44%  
tart-cli.tart: 46%  
tart-cli.tart: 48%  
tart-cli.tart: 49%  
tart-cli.tart: 51%  
tart-cli.tart: 53%  
tart-cli.tart: 55%  
tart-cli.tart: 57%  
tart-cli.tart: 61%  
tart-cli.tart: 64%  
tart-cli.tart: 67%  
tart-cli.tart: 70%  
tart-cli.tart: 73%  
tart-cli.tart: 76%  
tart-cli.tart: 79%  
tart-cli.tart: 82%  
tart-cli.tart: 84%  
tart-cli.tart: 87%  
tart-cli.tart: 90%  
tart-cli.tart: 100%  
==> tart-cli.tart: Waiting 3  
==> tart-cli.tart: Updating  
==> tart-cli.tart: Inspecting  
==> tart-cli.tart: Starting  
==> tart-cli.tart: Successful  
==> tart-cli.tart: Typing bo  
==> tart-cli.tart: Waiting f  
==> tart-cli.tart: Retrieved  
    tart-cli.tart: If you wa  
-next" to  
    tart-cli.tart: vnc://127  
==> tart-cli.tart: Connected  
==> tart-cli.tart: Typing co
```



Door gebruik te maken van deze software ga je akkoord met de voorwaarden van de licentieovereenkomst voor de software. Je kunt deze voorwaarden ook lezen op <https://www.apple.com/legal/sla/> (Engelstalig)

350% Faster

```
● ○ □ ~/Git/ejvm-packer + ~  
tart-cli.tart:  
tart-cli.tart: System shutdown time has arrived  
==> tart-cli.tart: Waiting for the tart process to exit...  
Build 'tart-cli.tart' finished after 16 minutes 642 milliseconds.  
  
==> Wait completed after 16 minutes 643 milliseconds  
  
==> Builds finished. The artifacts of successful builds are:  
--> tart-cli.tart: appletart  
  
rob.potvin ➤ .../ejvm-packer ➤ dev !? ➤ 14:31  
tart clone appletart dutchapple  
  
rob.potvin ➤ .../ejvm-packer ➤ dev !? ➤ 14:32  
tart list  
  
Source Name  
Disk Size SizeOnDisk State  
local appletart  
 50 22 22 stopped  
local dutchapple  
 50 22 22 stopped  
local newvmtest  
 50 24 24 stopped  
local vmtest  
 50 25 25 stopped  
OCI ghcr.io/cirruslabs/macos-sequoia-vanilla:latest  
 50 24 24 stopped  
OCI ghcr.io/cirruslabs/macos-sequoia-vanilla@sha256:01123a82f0507f3d005dfe8a0abf8f3d6f534d8aaabee54f059a129eee0706  
13 50 24 24 stopped  
  
rob.potvin ➤ .../ejvm-packer ➤ dev !? ➤ 14:32  
tart set experiencejamf --display-refit --random-mac --random-serial
```

350% Faster

tart + ^

```
==> Wait completed after 16 minutes 643 milliseconds  
==> Builds finished. The artifacts of successful builds are:  
--> tart-cli.tart: appletart
```

rob.potvin > .../ejvm-packer dev 1.2 14:31
tart clone appletart dutchapple

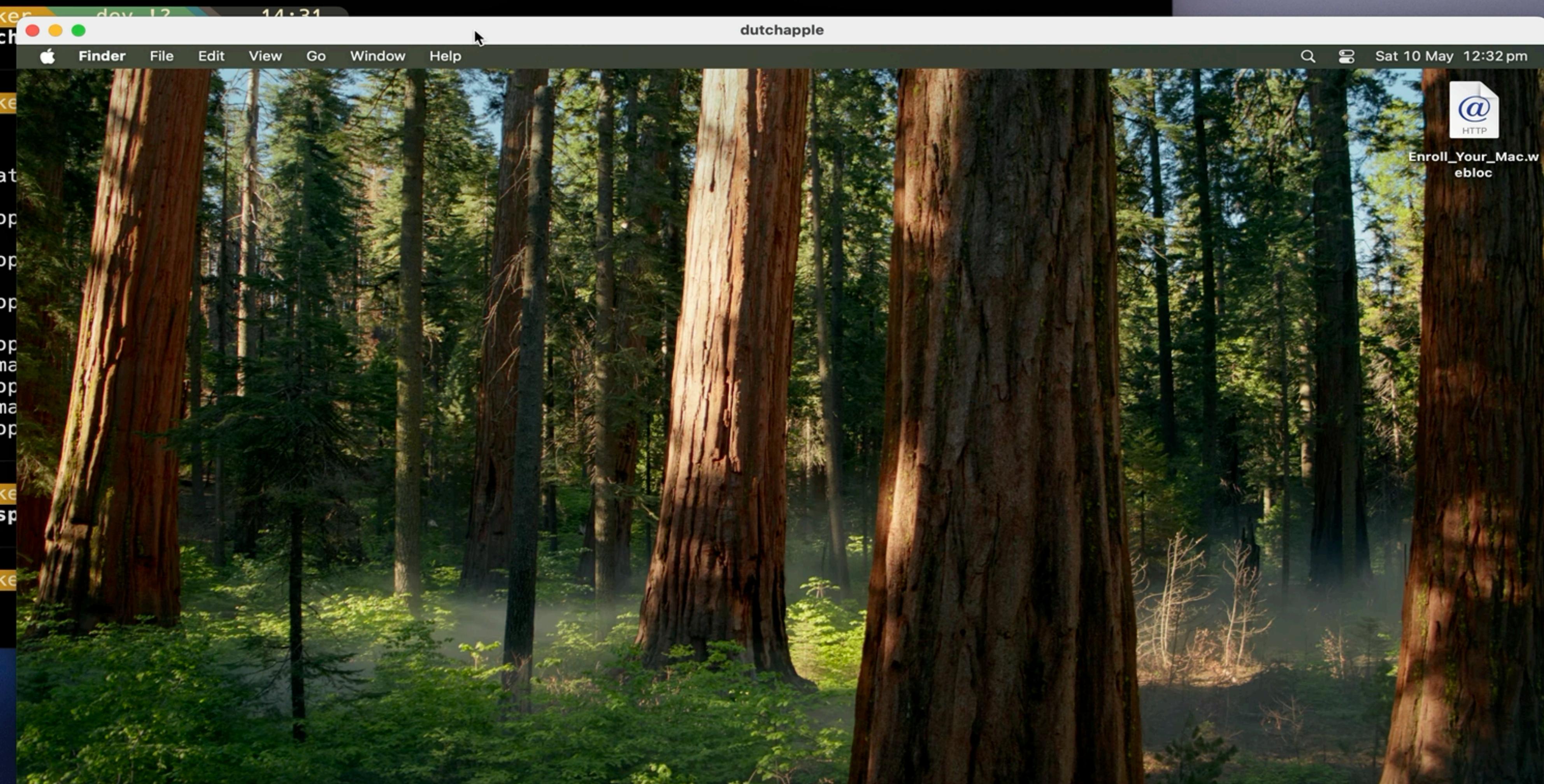
Finder File Edit View Go Window Help dutchapple

rob.potvin > .../ejvm-packer
tart list

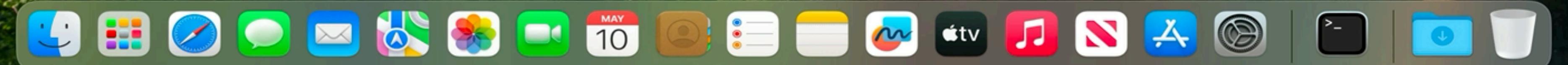
Source Name	Disk	Size	SizeOnDisk	Stat
local appletart	50	22	22	stop
local dutchapple	50	22	22	stop
local newvmtest	50	24	24	stop
local vmtest	50	25	25	stop
OCI ghcr.io/cirruslabs/maven	50	24	24	stop
OCI ghcr.io/cirruslabs/maven	13	50	24	24
				stop

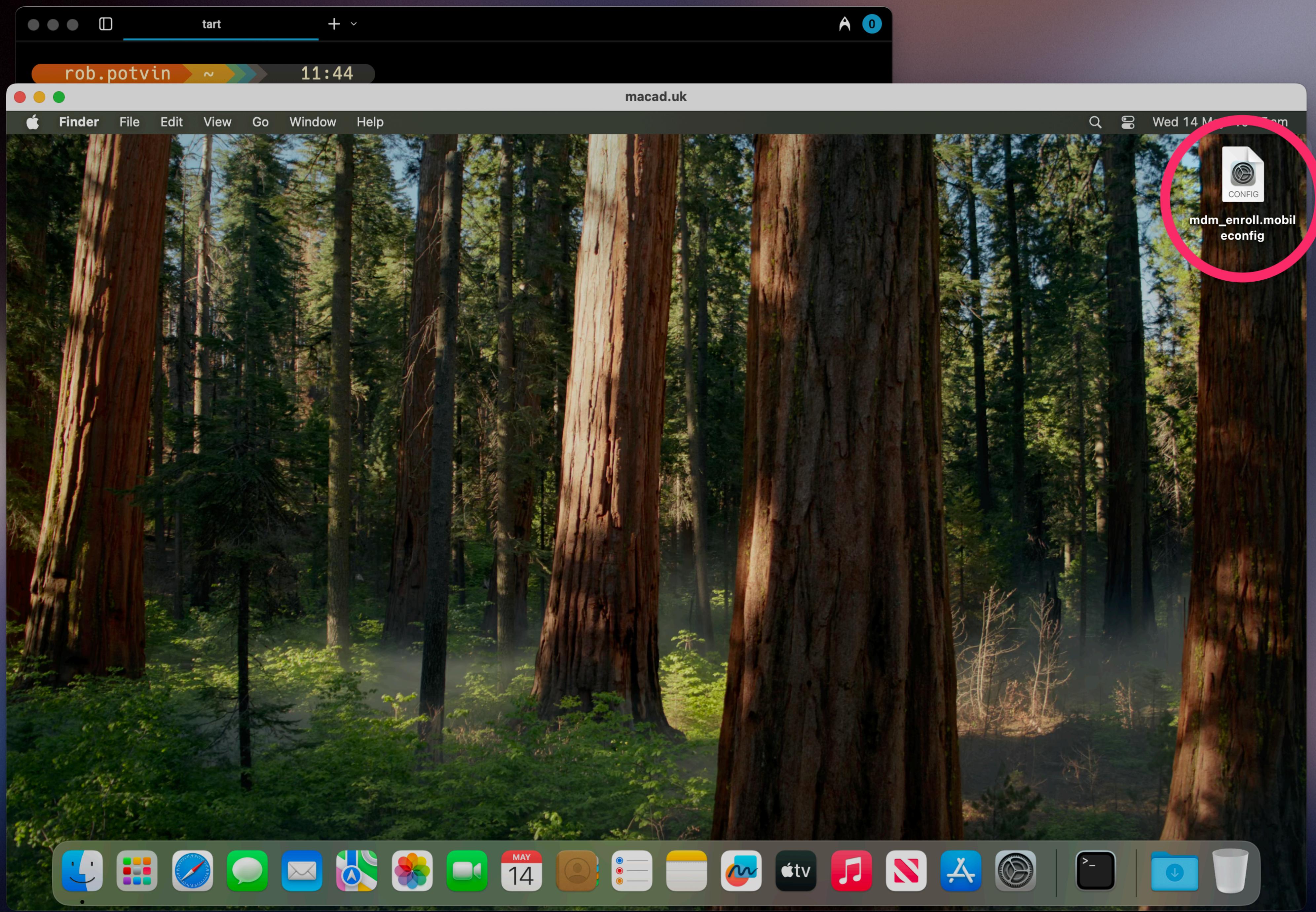
rob.potvin > .../ejvm-packer
tart set dutchapple --display

rob.potvin > .../ejvm-packer
tart run dutchapple



350% Faster

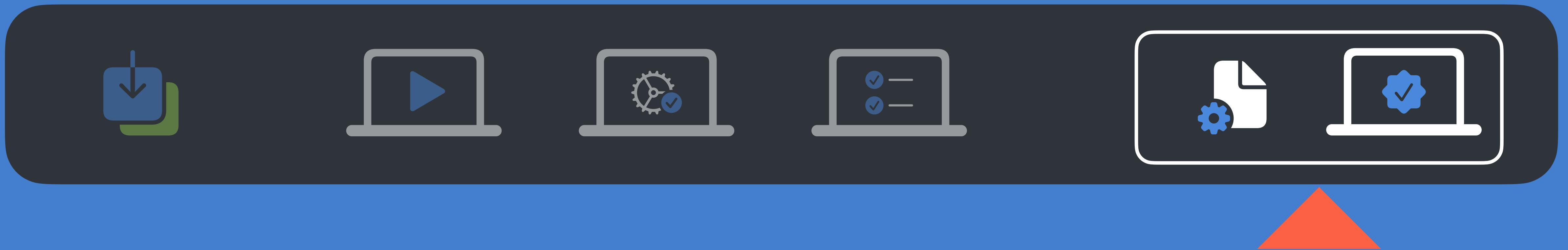


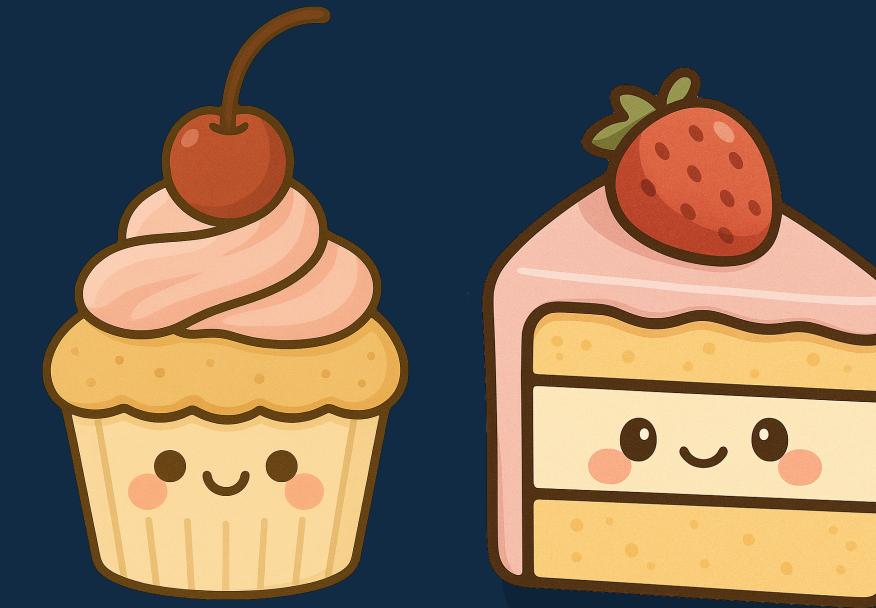


Clone



Testing





Baking and Sharing: Serving Up VMs with OCI Registries

Sharing is caring

Upload the image

- Use custom-baked images for rapid Jamf Pro workflow testing.
- Amazon ECR to host images
- Minimize manual steps
- Tag each image





Amazon ECR

github.com

aws | Search [Option+S]

Amazon ECR > Private registry > Repositories

Amazon Elastic Container Registry

Private registry

- Repositories
- Features & Settings

Public registry

- Repositories
- Settings

ECR public gallery

Amazon ECS

Amazon EKS

Getting started

Documentation

Private repositories (2)

Search by repository substring

Repository name	URI	Created at	Tag immutability	Encryption type
jamf/macaduk	106284622126.dkr.ecr.eu-central-1.amazonaws.com/jamf/macaduk	May 10, 2025, 21:13:34 (UTC+02)	Mutable	AES-256

C View push commands Delete Actions Create repository



Amazon ECR

Personal > Search [Option+S] AWS GitHub.com

Amazon ECR > Private registry > Repositories > jamf/macaduk

Amazon Elastic Container Registry

Private registry

- Repositories
- Summary
- Images**
- Permissions
- Lifecycle Policy
- Repository tags
- Features & Settings

Public registry

- Repositories
- Settings

ECR public gallery

Amazon ECS

Amazon EKS

Getting started

Documentation

Images (1)

<input type="checkbox"/>	Image tag	Artifact type	Pushed at	Size (MB)	Image URI	Digest	Last recorded pull time
<input type="checkbox"/>	latest	Image	May 10, 2025, 21:26:17 (UTC+02)	19699.96	<input type="checkbox"/> Copy URI	<input type="checkbox"/> sha256:f0fc1ce6a85aa38...	-

View push commands

Quick Whip, Fast Bake: Deploying with Tart

```
tart push 106284622126.dkr.ecr.eu-central-1.amazonaws.com/jamf/  
macaduk:latest  
pushing dutchapple to 106284622126.dkr.ecr.eu-central-1.amazonaws.com/jamf/  
macaduk:latest...  
pushing config...  
pushing disk... this will take a while...  
100%  
pushing NVRAM...  
pushing manifest for latest...
```





The Taste Test: **Rapid MDM Enrollment & Workflow Testing**

macad.uk Personal emeia.jamfce.com Full Jamf Pro 1

All Managed Clients

Computer Group Criteria Reports

AND/OR	CRITERIA	OPERATOR	VALUE	
	Operating System	not like	server	<input type="button" value="Delete"/>
and	Application Title	is not	Server.app	<input type="button" value="Delete"/>
and	Model	is not	VirtualMac2,1	<input type="button" value="Delete"/>

+ Add

macad.uk Personal < > emeia.jamfce.com 1 match Begins with ↗ < > Done

Full Jamf Pro 1

Computers : Smart Computer Groups

← Apple Silicon - VMs

Computer Group Criteria Reports

AND/OR	CRITERIA	OPERATOR	VALUE	...	▼	Delete
▼	Model	like	VirtualMac2,1	...	▼	+ Add

<>

The screenshot shows a web browser window for 'emeia.jamfce.com' with a search bar containing 'vm'. The main content area displays a 'Smart Computer Groups' search result titled 'Apple Silicon - VMs'. The interface includes a sidebar with icons for different sections like Computer Group, Criteria, Reports, and Settings. The 'Criteria' tab is active, showing a single search criterion: 'Model' set to 'like' operator with the value 'VirtualMac2,1'. There are buttons for adding more criteria ('+ Add') and deleting existing ones ('Delete'). A blue box highlights the title 'Apple Silicon - VMs'.

emeia.jamfce.com

Full Jamf Pro ▾  

Pro

Computers

Inventory

Search Inventory

Search Volume Content

Licensed Software

Content Management

Policies

Configuration Profiles

Software Updates

Restricted Software

Mac Apps

Patch Management

eBooks

Groups

Smart Computer Groups

Static Computer Groups

Computer Group Criteria Reports

Show in Jamf Pro Dashboard

Display Name
Display name for the smart computer group
Apple Silicon - VMs

Send email notification on membership change
When group membership changes, send an email notification to Jamf Pro users with email notifications enabled. An SMTP server must be set up in Jamf Pro for this to work

Site
Site to add the smart computer group to
None

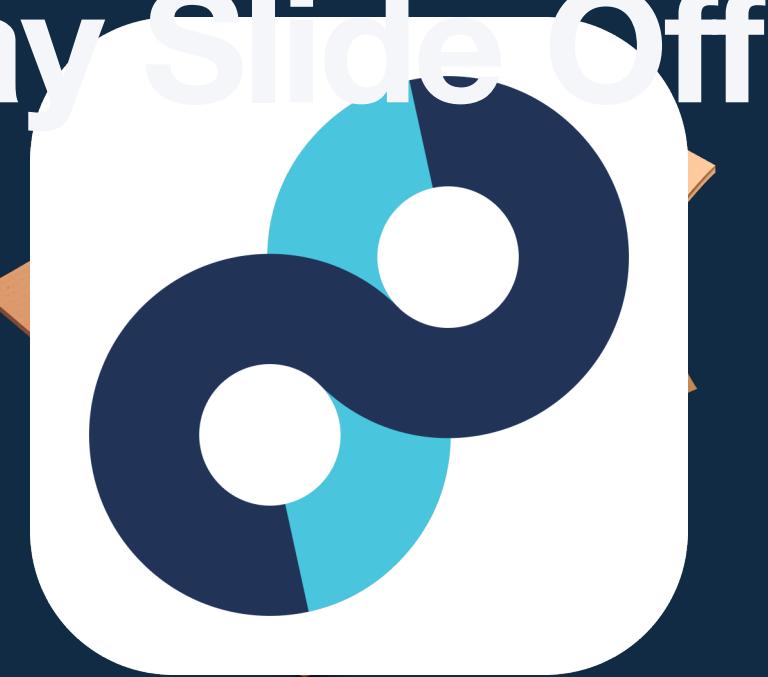
History View Clone Delete Edit

Computers : Smart Computer Groups

Apple Silicon - VMs

The Cake's Baked, But the Sprinkles May Slide Off

- Attestation
- Certificates
- Package deployment for VPP
- Anything USB (no mappings to hardware)
- Video (no cameras)
- And all that other stuff



Screen Sharing: Because Every Recipe Needs a Little Copy-Paste

- Packer template enabled
- Easy access VM via VNC
- Run VM headless
- Clipboard works flawlessly



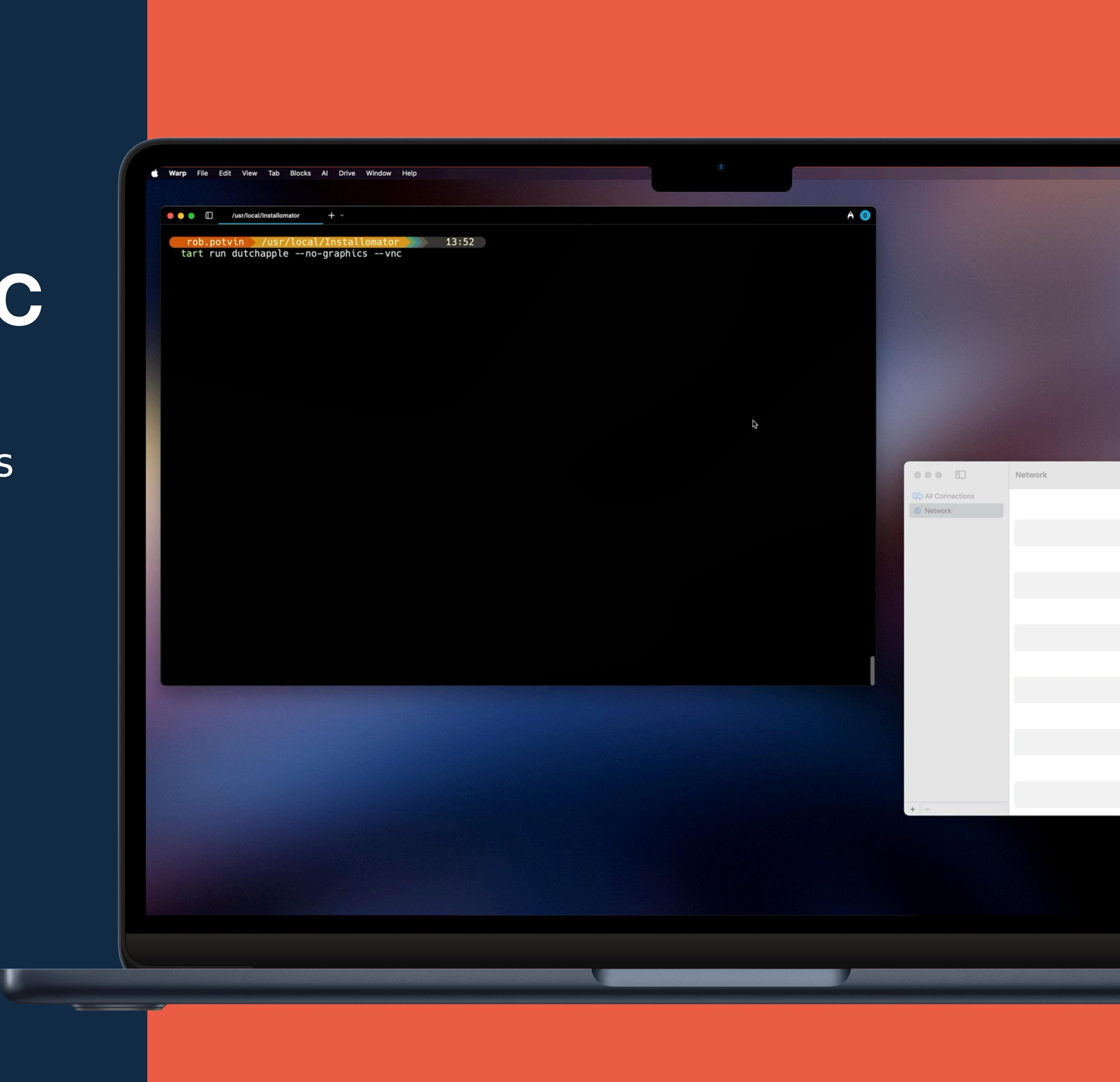
We Knead That Clipboard – Enter VNC

- ▶ `tart run --vnc --no-graphics`

Starts headless with vnc

- ▶ `tart run --vnc-experimental`

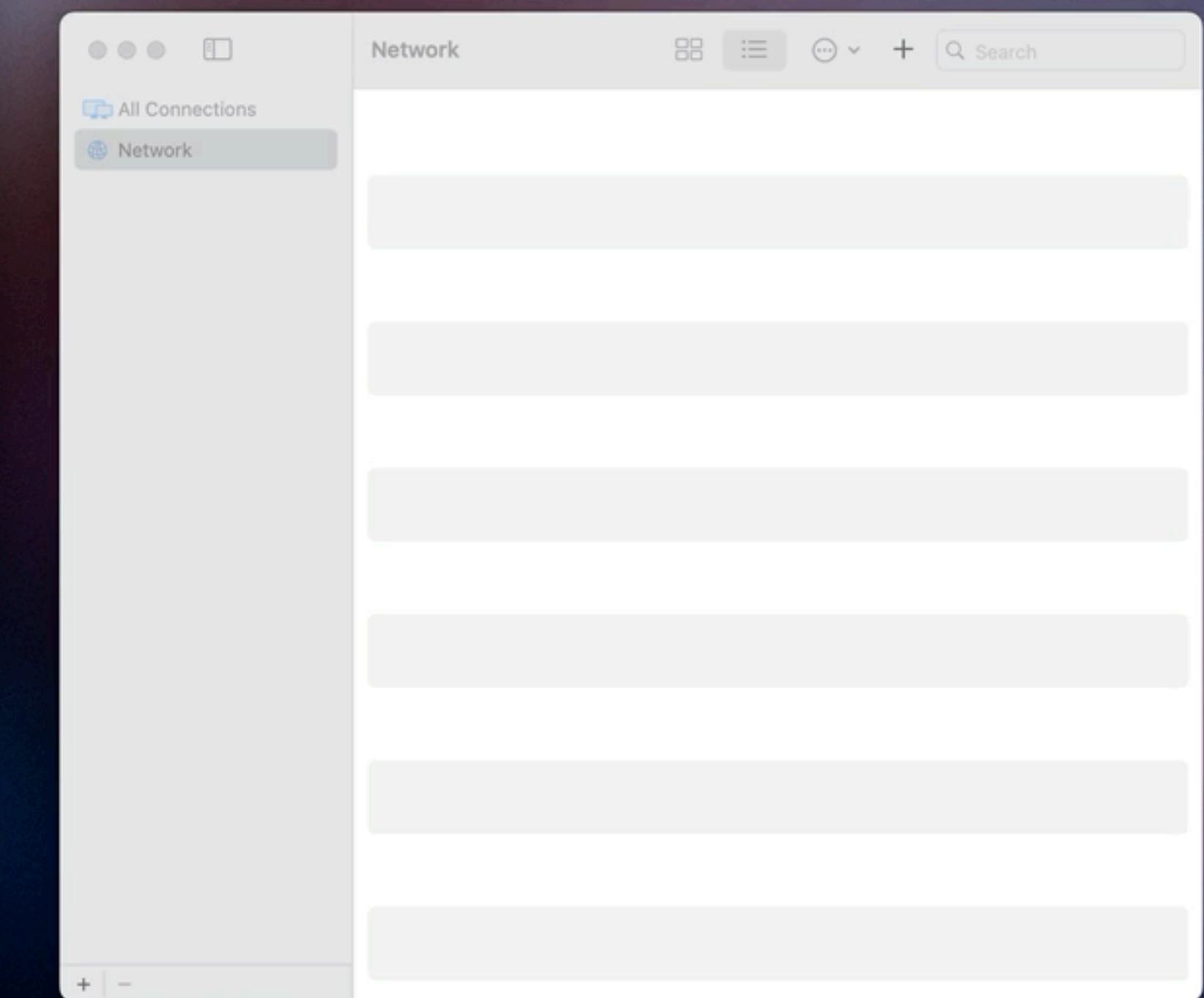
Tool for recovery partition



/usr/local/Installomator +

rob.potvin > /usr/local/Installomator 13:52

```
tart run dutchapple --no-graphics --vnc
```

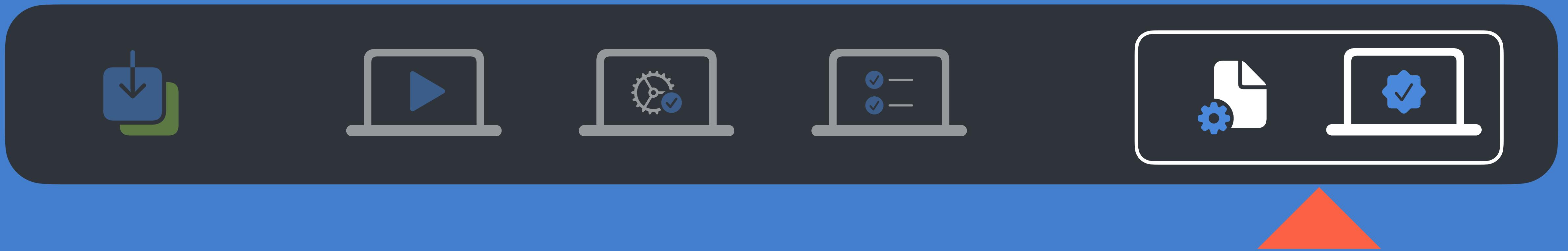




Serving Suggestions: **Takeaways for the Modern Mac**

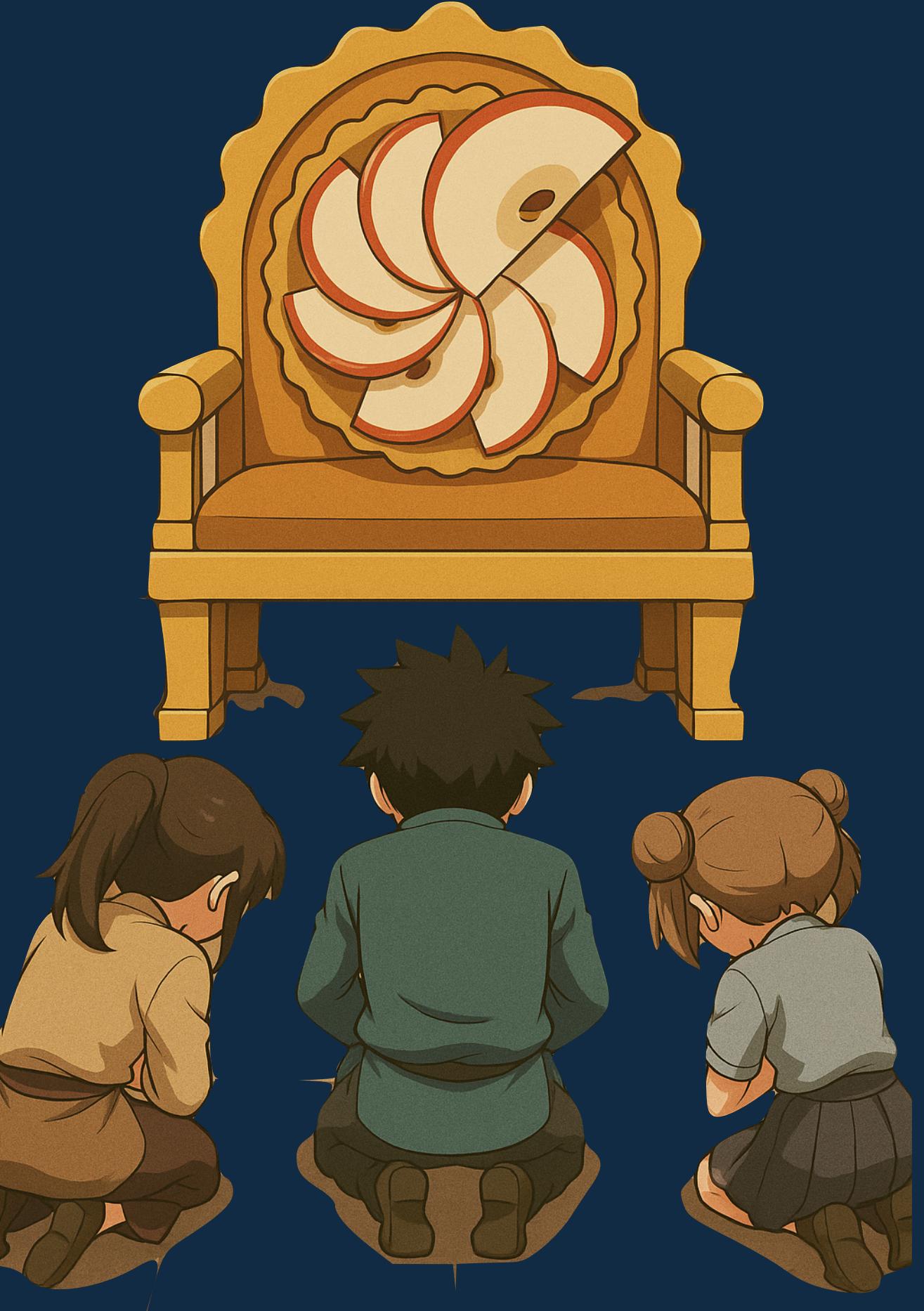
Admin Baker

Testing



Cooling on the Rack: Your Workflow in a Nutshell

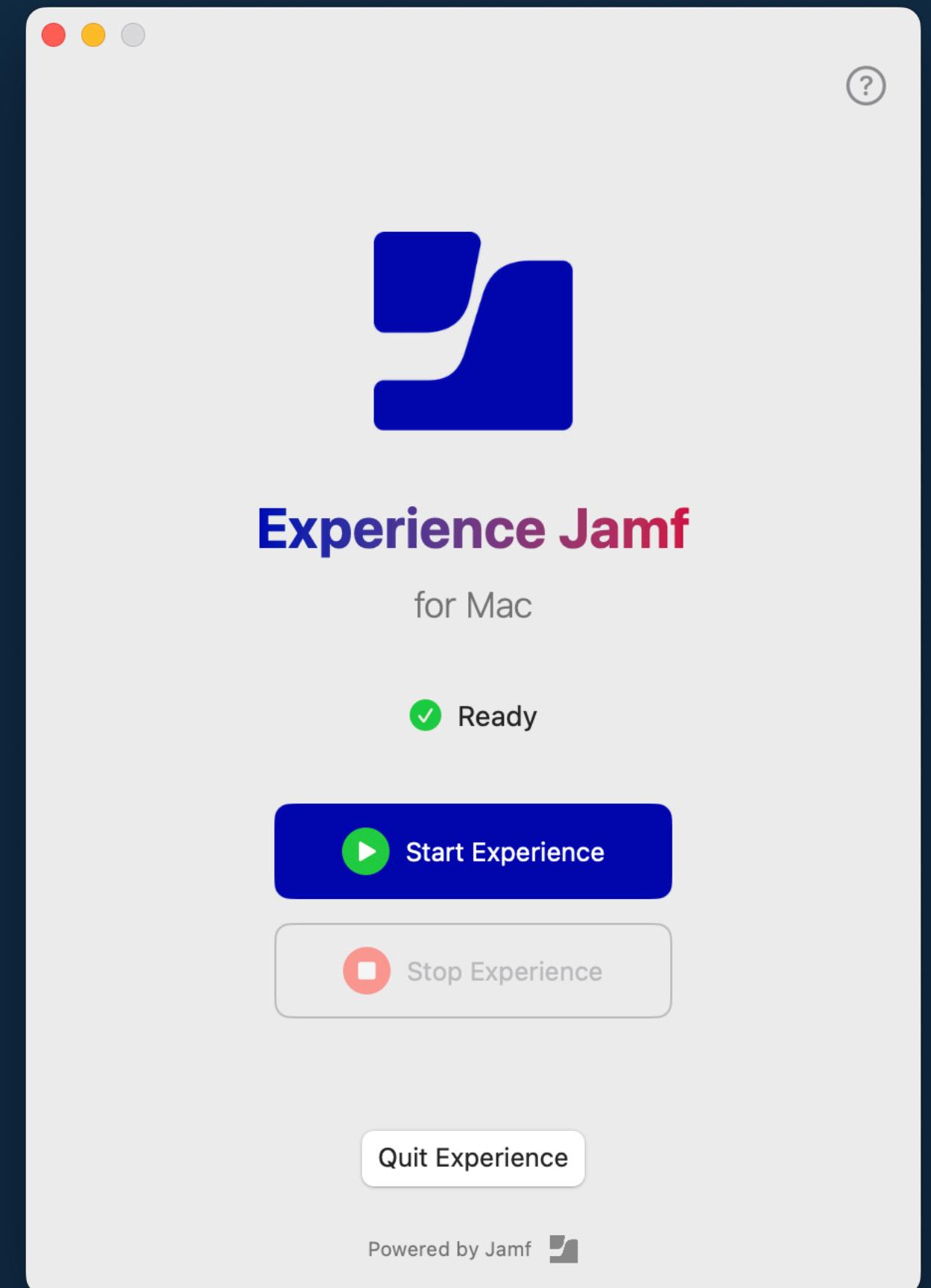
- tart is awesome
- CI/DI - but for macadmins
- VMs are disposable
- Test workflows
- Use Self Service Onboarding or Setup Manager
- This is just the start with tart

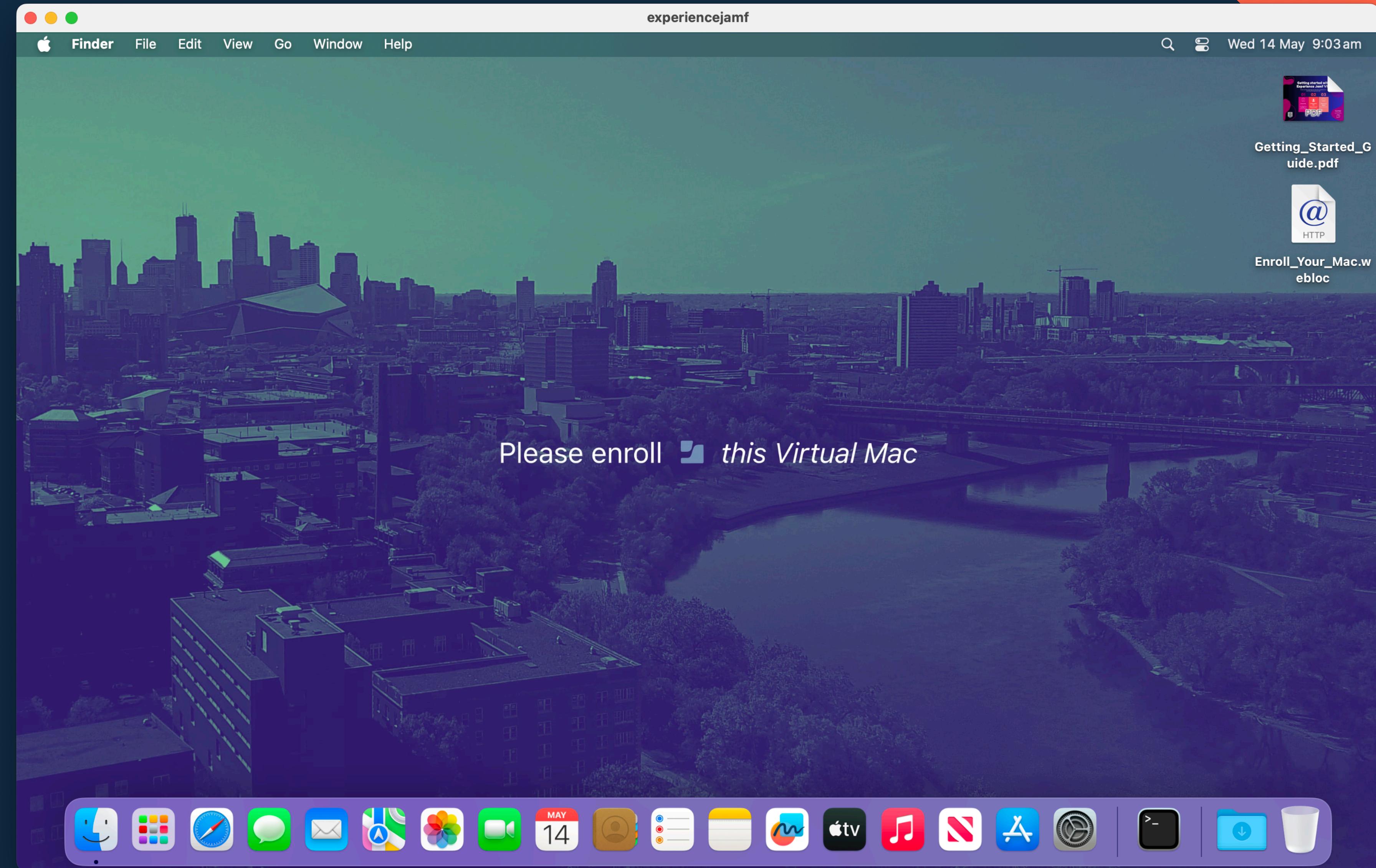


Tart: Peeling Back the Layers to the Good Bits

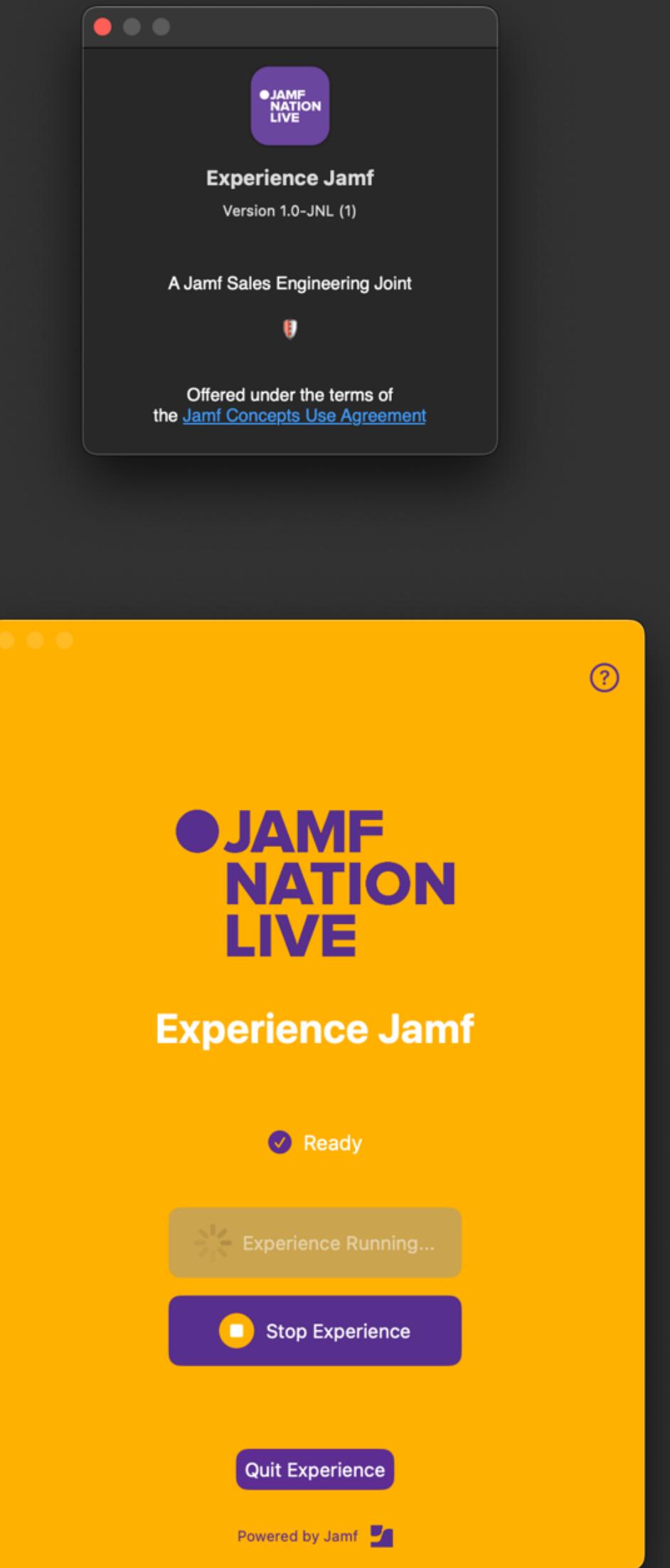
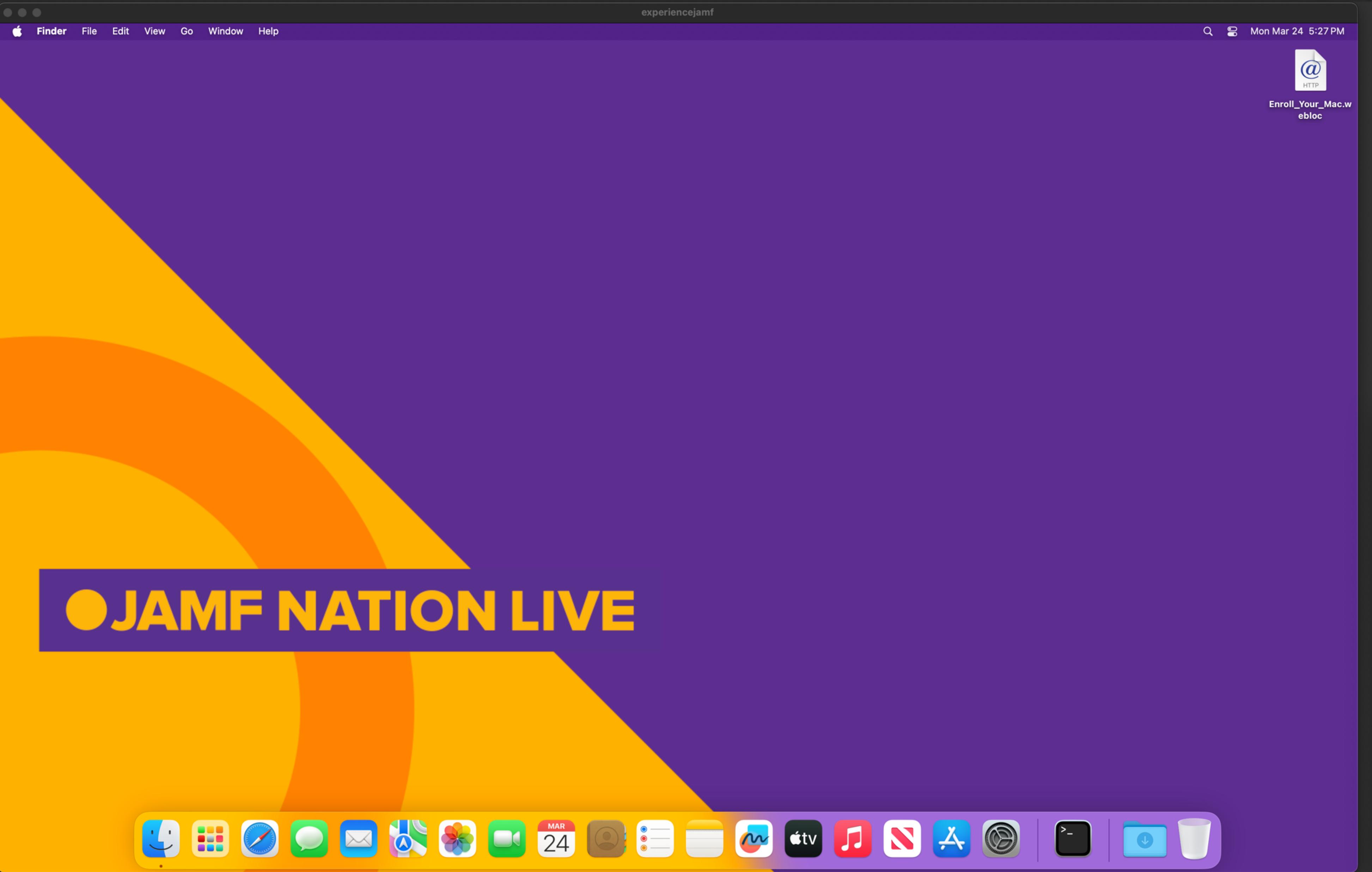
- Using Xcode and SwiftUI with tart
- Using Python flask with tart

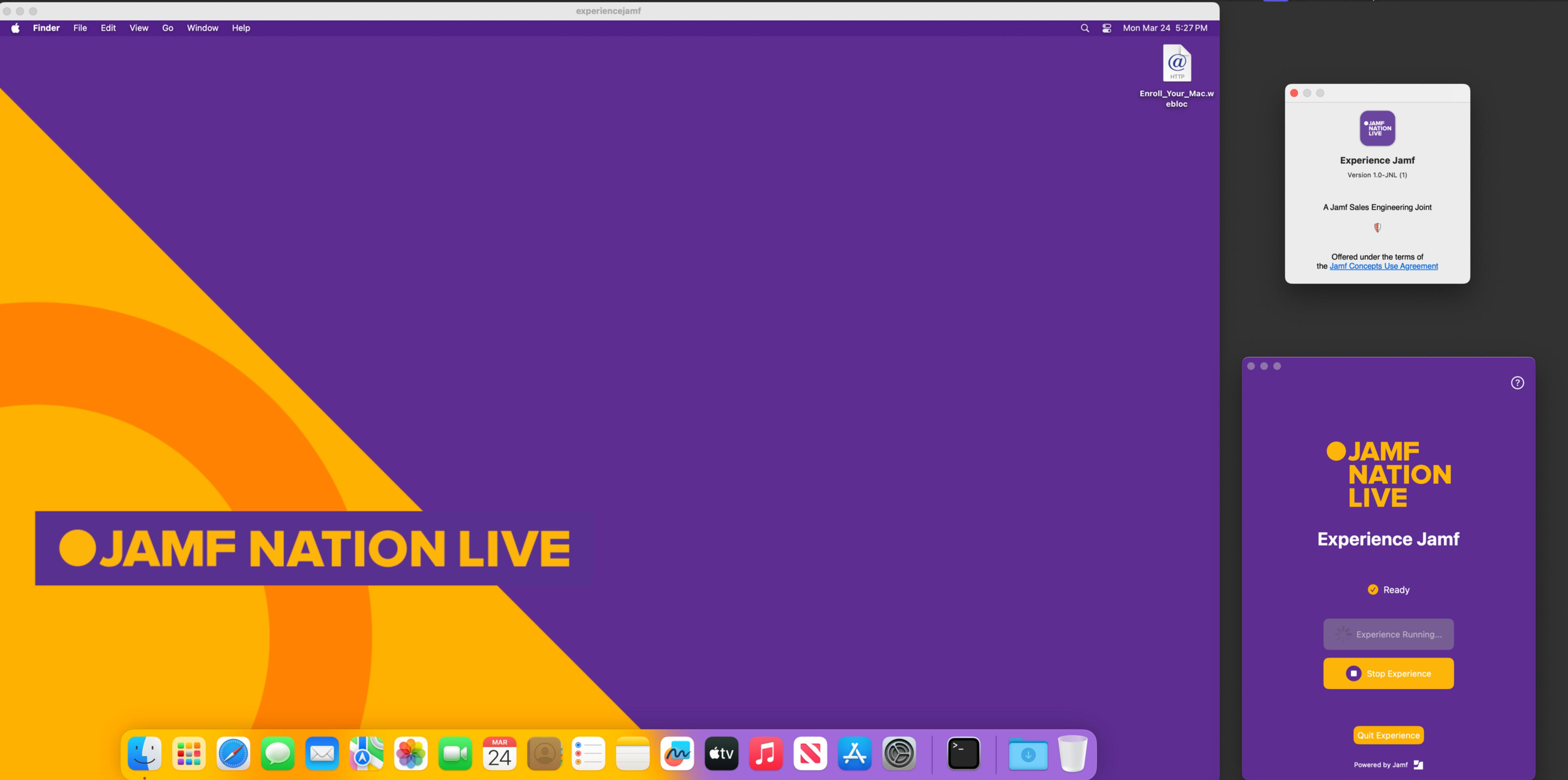






jamf





Jamf 127.0.0.1 Open in Work - Jamf

macOS VM Manager

Virtual Machines

Manage your macOS Virtualization Framework VMs

Grid List

experiencejamf
macOS Unknown
Stopped

4 CPUs
8 GB RAM
30 GB Disk

Test
macOS Unknown
Stopped

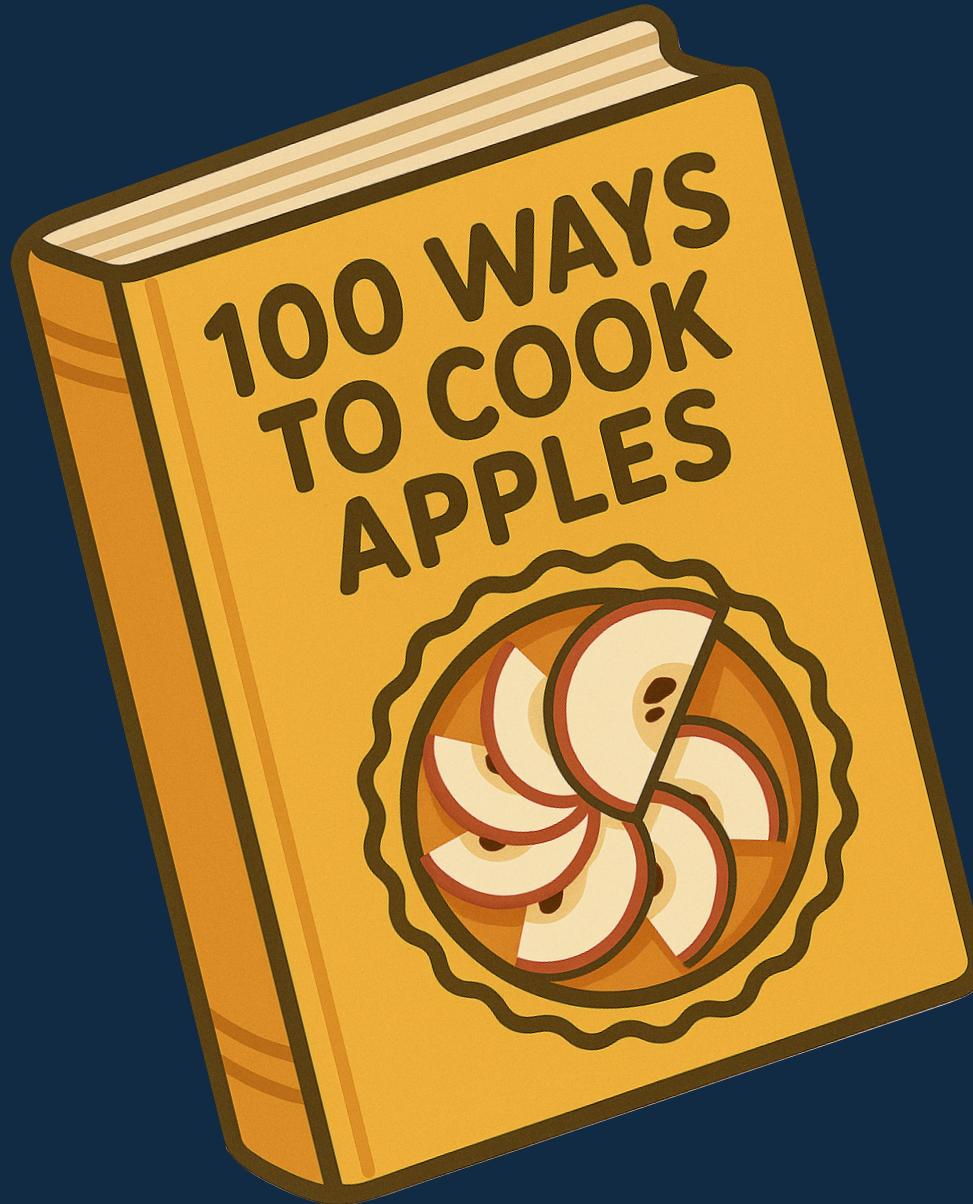
4 CPUs
8 GB RAM
50 GB Disk

Start Details ... Start Details ...

© 2025 macOS VM Manager - Powered by Tart and Flask

Leftovers You'll Love: Blog Preview & Handy Resources

- Cook Book online
 - motionbug.com
- Recipes for what you need
- My daughter hopes that I win the bakeoff



Q&A



Thank You