

Infrastructure Overview

Runbook: Local Infrastructure

Platform: Mac Studio M3 Ultra, 96GB RAM, macOS **Services:** Docker containers, LaunchAgents, Cloudflare tunnels

Docker Services

Status Check

```
docker ps --format 'table {{.Names}}\t{{.Status}}\t{{.Ports}}'
```

Services

Container	Image	Port	Health
mattermost	mattermost/mattermost-enterprise	8065	healthcheck
mattermost-db	postgres:13	5432 (internal)	—
planka	ghcr.io/plankanban/planka	3100→1337	healthcheck
planka-db	postgres:15-alpine	5433 (internal)	healthcheck
documenso	documenso/documenso	3003	—
documenso-db	postgres	5434 (internal)	healthcheck
bookstack	lscr.io/linuxserver/bookstack	6875→80	—
bookstack_db	lscr.io/linuxserver/mariadb	3306 (internal)	—

Restart a container

```
docker restart <container_name>
# Example: docker restart mattermost
```

Restart all

```
# Mattermost stack
cd ~/system/services/mattermost && docker compose down && docker compose up -d

# Planka stack
cd ~/system/services/planka && docker compose down && docker compose up -d

# Documenso
cd ~/system/services/documenso && docker compose down && docker compose up -d

# BookStack
cd ~/system/services/bookstack && docker compose down && docker compose up -d
```

View logs

```
docker logs <container_name> --tail 50
docker logs <container_name> -f # follow
```

Disk cleanup (if disk >90%)

```
docker system prune -f          # Remove unused images, containers, networks
docker volume prune -f         # Remove unused volumes (CAREFUL: data loss)
```

Cloudflare Tunnels

Config

```
cat ~/.cloudflared/config.yml
```

Routes

Hostname	Target	Service
mm.basicconsulting.no	DECOMMISSIONED 2026-05-18	Mattermost (retired)
boards.alai.no	localhost:3100	Planka
sign.alai.no	localhost:3003	Documenso

Status

```
cloudflared tunnel info mattermost
```

Restart tunnel

```
# Tunnel runs as LaunchAgent  
launchctl unload ~/Library/LaunchAgents/com.cloudflare.tunnel.plist  
launchctl load ~/Library/LaunchAgents/com.cloudflare.tunnel.plist
```

LaunchAgents (Daemons)

List all custom daemons

```
launchctl list | grep -E "com\.(john|edita|cloudflare)"
```

Expected daemons

Daemon	Interval	Location
com.john.ops-agent	5 min	~/Library/LaunchAgents/
com.edita.autowork	30 min	~/Library/LaunchAgents/
com.john.mc-dashboard	always	~/Library/LaunchAgents/
com.john.mc-session-worker	on events	~/Library/LaunchAgents/

Load/unload

```
launchctl load ~/Library/LaunchAgents/<plist-name>.plist  
launchctl unload ~/Library/LaunchAgents/<plist-name>.plist
```

Ollama (Local AI)

Status

```
curl -s http://localhost:11434/api/tags | python3 -c "import sys,json; [print(m['name']) for m in json.load(sys.stdin)['models']]"
```

Models

Model	Size	Use
llama3.1:8b	5GB	Fast classification (ops-agent)
qwen2.5-coder:32b	19GB	Code generation, contextual responses
llama3.1:70b	40GB	Research, writing

Restart Ollama

```
# Ollama runs as macOS app  
killall ollama 2>/dev/null  
open -a Ollama
```

Mission Control Dashboard

Status

```
curl -s http://localhost:3030 | head -1
```

Restart

```
launchctl unload ~/Library/LaunchAgents/com.john.mc-dashboard.plist  
launchctl load ~/Library/LaunchAgents/com.john.mc-dashboard.plist
```

Full Health Check

```
# Human-readable
node ~/system/tools/health-check.js

# JSON (programmatic)
node ~/system/tools/health-check.js --json

# Quick (HTTP only)
node ~/system/tools/health-check.js --quick
```

After System Reboot

All LaunchAgents with `RunAtLoad: true` start automatically. Verify:

```
# 1. Check Docker is running
docker ps

# 2. Check all daemons
launchctl list | grep -E "com\.(john|edita|cloudflare)"

# 3. Run health check
node ~/system/tools/health-check.js

# 4. If anything missing, load it
launchctl load ~/Library/LaunchAgents/<missing>.plist
```

Created: 2026-02-10 **Last Updated:** 2026-02-10

Revision #6

Created 2026-02-20 09:38:28 UTC by John

Updated 2026-06-21 20:01:46 UTC by John