



≡ macOS companion app > Menu Bar Icon

macOS companion app

Menu Bar Icon

Author: steipete · Updated: 2025-12-06 · Scope: macOS app (apps/macos)

Idle: Normal icon animation (blink, occasional wiggle).

Paused: Status item uses `appearsDisabled` ; no motion.

Voice trigger (big ears): Voice wake detector calls `AppState.triggerVoiceEars(ttl: nil)` when the wake word is heard, keeping `earBoostActive=true` while the utterance is captured. Ears scale up (1.9x), get circular ear holes for readability, then drop via `stopVoiceEars()` after 1s of silence. Only fired from the in-app voice pipeline.

Working (agent running): `AppState.isWorking=true` drives a “tail/leg scurry” micro-motion: faster leg wiggle and slight offset while work is in-flight. Currently toggled around WebChat agent runs; add the same toggle around other long tasks when you wire them.

Wiring points

Voice wake: runtime/tester call `AppState.triggerVoiceEars(ttl: nil)` on trigger and `stopVoiceEars()` after 1s of silence to match the capture window.

Agent activity: set `AppStateStore.shared.setWorking(true/false)` around work spans (already done in WebChat agent call). Keep spans short and reset in `defer` blocks to avoid stuck animations.

Shapes & sizes



Base icon drawn in

```
CritterIconRenderer.makeIcon(blink:legWiggle:earWiggle:earScale:earHoles:)
```

Ear scale defaults to 1.0 ; voice boost sets earScale=1.9 and toggles earHoles=true without changing overall frame (18×18 pt template image rendered into a 36×36 px Retina backing store).

Scurry uses leg wiggle up to ~1.0 with a small horizontal jiggle; it's additive to any existing idle wiggle.

Behavioral notes

No external CLI/broker toggle for ears/working; keep it internal to the app's own signals to avoid accidental flapping.

Keep TTLs short (<10s) so the icon returns to baseline quickly if a job hangs.

< Health Checks

macOS Logging >

Powered by [mintlify](#)