A Raspberry Pi 5 inside a telephone booth

Edge architecture, field notes and reliability of a physical voice agent

Cris Olmedo and Kurt Sommer
Divina Machina Research Group
Working paper · Technical note
DM-2026-07, version 1.0 · 30 October 2026

Abstract. A voice agent inside a physical object has to decide what runs on the device and what in the cloud. We document the GAIA 2147 booth at Barcelona Design Week: a Raspberry Pi 5 handling everything that can be touched, the hook switch, audio, light, display and paper, and two cloud voice platforms, ElevenLabs and Vidiv, with automatic failover between them.

We give the full specification, eleven failure modes found in the field with their cause and fix, and the reliability recorded in the cloud. In Barcelona, 0.7 % (95 % CI 0.3–1.9 %) of 543 calls ended with an abnormal disconnect and 0.4 % with a technical failure. The first audio of the greeting left the server in a median of 95 ms with the connection prepared before pick-up. In one window of booth logs, 21 % (95 % CI 15–29 %) of hang-ups ran out a two-second wait in the audio thread, a failure still open.

The rule we propose is simple: the edge handles what is touched and what must happen in milliseconds; the cloud handles language, voice and safety, which change every few months.

Keywords: edge computing; Raspberry Pi; voice agents; reliability; interactive installations; failure mode analysis.

1. Introduction

A telephone booth at a trade fair has simple needs and hard constraints. It has to look like a telephone, start on its own and run all day, survive the hall's wifi and open no port to the internet. We built it on a Raspberry Pi 5: quad-core Cortex-A76 at 2.4 GHz, dual-band wifi, Bluetooth 5.0, two USB 3 ports and the 40-pin header, with a 5 V, 5 A supply [1]. That is enough for the booth's physical world. It is not where the voice should be made.

This note covers the Barcelona Design Week version. The logs do not document the hardware at Sónar+D and IFA; reliability figures for those events come from the cloud and do not compare hardware [MED].

2. Architecture

The Pi handles everything that can be touched and the call state (figure 1). It plays the dial tone, opens the conversation, carries audio to and from the handset, drives light, display and printer, and keeps rules and quotas so nothing runs away. The cloud handles voice, language and safety. We always work with two platforms, ElevenLabs and Vidiv; if one fails, the booth moves to the other without intervention [MED]. A Cloudflare Worker is the far end of a tunnel opened by the booth, through which tools and settings come down and state goes up.

3. Failure modes

Table 2 records what happened to us, in the format of a failure mode analysis. These are observations from building and operating the booth, not counts [MED].

Two deserve an equation. The WS2812B datasheet requires a high level of at least $0.7\,V_{DD}$ [2]:

$$V_{IH,\min} = 0.7 \times 5\,\mathrm{V} = 3.5\,\mathrm{V} > V_{OH,\mathrm{Pi}} \approx 3.3\,\mathrm{V}. \quad (1)$$

Many strips work anyway; ours did not, and we never fitted the shifter. Figure 2 shows the proposed fix, a 74AHCT-family buffer powered at 5 V, whose input threshold accepts the Pi's 3.3 V and whose output reaches V_{DD} [HIP].

And the audio thread wait: in a three-and-a-half-hour window of booth logs, 21 % (95 % CI 15–29 %) of 136 hang-ups ran out a two-second wait [MED]. That count comes from the Pi logs, which are not part of the gaia-2147 dataset.

4. Reliability

Table 3 gives the reliability recorded in the cloud for the three events [MED]. Barcelona had the lowest share of abnormal disconnects, 0.7 % (95 % CI 0.3–1.9 %), against 7.4 % (95 % CI 5.7–9.6 %) at Sónar+D, whose hardware we did not document. The first audio of the greeting fell from 287 ms (synthesis only, Sónar+D) to 95 ms in Barcelona, with the signed connection prepared before pick-up. These are server times: the Pi and the loudspeaker add their own delay, for which we only have scattered readings of 0.55 to 0.73 s from opening the conversation to the first sound.

The load is real (figure 3): in Barcelona the booth reached 75 calls in one hour. One afternoon it took more than a hundred calls without going down; the only interventions were restarts we chose to make. Cloud cost was 0.10 USD per minute in Barcelona and 0.12 USD at IFA; we did not cost the hardware.

5. What the edge should do

- Everything physical and everything that must happen in milliseconds: hook switch, tones, light, display, paper.
- The call state, to know when it is safe to print, restart or update.
- Platform failover: ElevenLabs and Vidiv together, with automatic switching.
- Privacy by default: local history, upload only if enabled and with a visible notice.

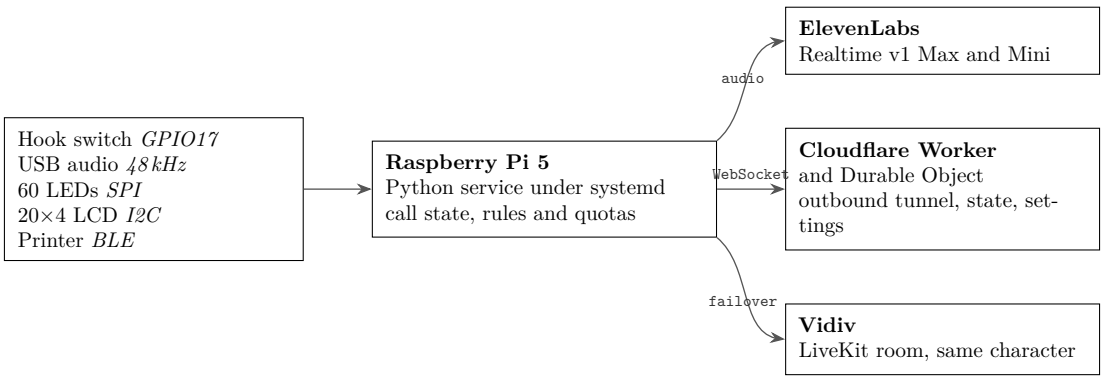


Figure 1. What runs where. The edge carries the physical and the state; the cloud carries language, voice and safety. The booth never opens a port: everything goes outbound.

Table 1. Booth specification at Barcelona Design Week, taken from its configuration and code.

Subsystem	Part	Interface	Setting
Computer	Raspberry Pi 5		Python service under systemd with automatic restart
Audio	USB sound card	USB, ALSA	48 kHz mono, 20 ms blocks; mixer at 98 %; gain 3.5; buffer target 40 ms, ceiling 30 s
Hook switch	Original switch	GPIO17	Polarity inverted in software; 120 ms debounce; 0.4 s confirmation
Display	20×4 LCD	I2C 0x27	Murmurs and short messages with per-call limits
Light	60 WS2812E LEDs	SPI, GPIO10	Brightness 0.35; idle colour read from the cloud every 10 s
Paper	Thermal printer	Bluetooth LE	384 dots per line; raised print energy; prints on hang-up
Voice	ElevenLabs Realtime v1	WebSocket	Signed connection prepared before pick-up
Voice, failover	Vidiv	LiveKit	Same character; automatic switch
Link	Cloudflare Workers	Outbound WebSocket	State every few seconds; settings every 10 s
Software	elevenlabs 2.66.0, websockets 16.0		Pinned versions; a silent update can undo a fix

Table 2. Failure modes found in the field.

Symptom	Cause	Fix	Status
Picking up did nothing	Hook switch with reversed polarity and bounce	Inverted reading, debounce and second check	Solved
Choppy voice	Bursty audio and a small buffer	Buffer ceiling to 30 s, target 40 ms	Solved
Volume reset on boot	The USB card changed number	Find it by name	Solved
Distortion on laughter	Too much gain in the handset	Mixer at 98 %, gain 3.5	Solved
Blank paper	Low default print heat	More print energy	Solved
Printer busy	A single Bluetooth connection	The service owns it	Solved
Printing broke the audio	Wifi and Bluetooth share the radio module	Print only on hang-up	Solved
Bluish white light	3.3 V data against a 3.5 V threshold	Level shifter and common ground	Open
GAIA answered itself	The microphone picked up the loudspeaker	Noise gate and lower microphone while speaking	Mitigated
Pick-up took about 2 s longer	The audio thread sometimes failed to confirm pause or resume	Stop without draining and time each step	Open
Late cloud setting	Stale copy in the tunnel object	Poll settings every 10 s	Solved

Table 3. Reliability and greeting by event, from the cloud logs. Brackets, Wilson or bootstrap 95 % CI.

Event	Calls	Abnormal disconnect	Technical failure	Greeting (ms)
Sónar+D	701	52 · 7.4 % [5.7; 9.6]	8 · 1.1 % [0.6; 2.2]	287 (TTS)
IFA	274	3 · 1.1 % [0.4; 3.2]	3 · 1.1 % [0.4; 3.2]	179 (e2e)
BDW	543	4 · 0.7 % [0.3; 1.9]	2 · 0.4 % [0.1; 1.3]	95 (e2e)

Abnormal disconnect, WebSocket close code 1006 or 1011. Technical failure, failed status without a safety guardrail. Greeting, server-side median of first audio in turn 0; Sónar+D only records the first synthesis byte.

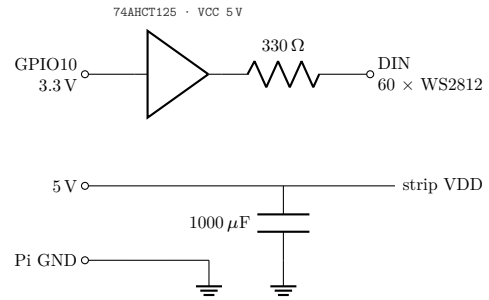


Figure 2. Proposed level shifter for the light strip, with common ground. Drawn with circuitikz; not installed in Barcelona.

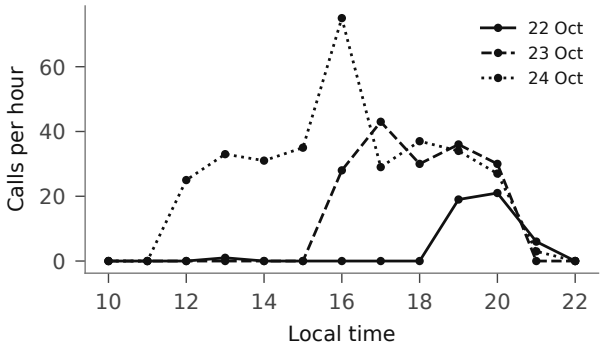


Figure 3. Calls per hour at Barcelona Design Week.

And what it should not: recognition, language model, voice and safety filters. They change every few months and the booth should not need a new image when they do [INT].

6. Threats to validity

A single version at a single event. Field notes are not counted. Reliability at Sónar+D and IFA does not correspond to documented hardware. Times to first sound at the booth are scattered readings. The count of expired waits covers one log window, not the event.

7. Recommendations for the community

1. Fit a level shifter on any WS2812 strip powered at 5 V from day one.
2. Give the printer its own radio or move to USB; printing must not compete with the voice.
3. Log on the device, every hour, uptime, restarts, first sound and echo.
4. Restart only with the handset on the hook, checking twice.
5. Pin library versions and keep a second card ready with a single install script.

8. Data and code

Reliability and server times from the gaia-2147 dataset [3]; figures from `analysis/analyze.py`. Specification and failure modes from the booth's configuration and logs.

9. Conclusions

A small computer is enough for an object to converse if it does not try to make the voice. The booth carried what is touched; the cloud, what is said. What failed was almost always physical: polarities, shared radios, voltages. That is the part no product sheet mentions.

Signed. Cris Olmedo and Kurt Sommer, Divina Machina Research Group. *How to cite*: Olmedo, C. and Sommer, K. (2026). A Raspberry Pi 5 inside a telephone booth. Working paper DM-2026-07, v 1.0. Divina Machina Research Group. Divina Machina by Qs Ventures.

References

Sources consulted up to 29 October 2026.

[1] Raspberry Pi Ltd. Raspberry pi 5. Product page, 2023. URL https://www.raspberrypi.com/products/raspberry-pi-5/.

[2] Worldsemi. WS2812B. intelligent control LED integrated light source. Datasheet, University of Cambridge copy. URL https://www.cl.cam.ac.uk/teaching/1516/ECAD+Arch/files/pcb/WS2812B.pdf.

[3] C. Olmedo and K. Sommer. GAIA-2147. data descriptor of 1527 conversations with a voice agent in public spaces. Working paper DM-2026-09. Divina Machina Research Group, 2026.