NERO PROJECT

Agrippina - AI Voice and Personality of the House

Text it at the store. Ask about it at the sink.

Paul Schindler | Built with Claude | November 2026

Executive summary

Store-bought voice assistants can set a timer and read the weather, but they know little of the household they serve. I gave our house one voice with a character. Agrippina is an imperial Roman matriarch with a quiet delight in the household's routines, and she speaks through voice pucks in three rooms. Her speech recognition and her voice run on my own hardware, and she shares one memory with the always-on messaging agent on my phone, so an item texted from the store is on the list when someone asks at the kitchen sink. I designed it, set the rules and verified it by ear; AI agents wrote the code.

3 rooms
voice pucks on one shared agent

1 memory
shared by voice and text

8 voices
rendered; 6 auditioned live on a puck

Live
since June 2026

What is in this report

Page	Contents
2	The problem, what I ruled out and why the house got a character
3	How it works - one brain, two mouths
4	Designing the character, her name and her voice
5	Eight safeguards and two reliability lessons
6	Results, status, what is next and what it shows

What a member of the household does

- Asks what happened ("When did the heat run last night?") and hears an answer read from real history.
- Texts the same agent from a store and asks about it later by voice.

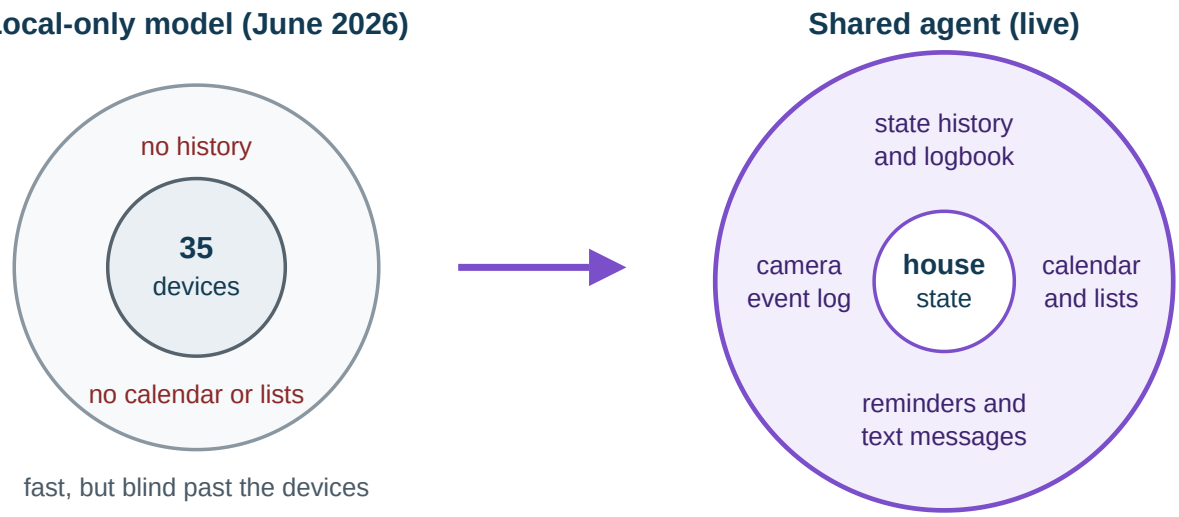
What the design rules out

- An AI that disarms the alarm on its own judgment.
- A voice that calls the house secure when a sensor is silent.

Details of the home, its network and its security equipment are left out of this report on purpose.

1. The problem and what I ruled out

I wanted a voice assistant that knew what the rest of our system knew, and I began designing one in my notes in December 2025. The first build, in June 2026, ran entirely on my own hardware. A small open model with Home Assistant tools answered in 0.6 seconds, but it could see 35 devices and nothing else. It failed the questions we actually asked, such as "When were the geese last seen?" and "What is on the calendar?"



The small local model answered fast but saw only the devices exposed to it. Routing voice to the agent that already answers my texts gave it everything that agent can reach.

Option	Why I ruled it out
Store-bought smart speaker	It knows little of the household, and what it learns sits on the maker's servers.
Small local model only	Fast, but limited to 35 devices. It could not reach history, the calendar or the camera event log.
A separate agent just for voice	Two agents to harden and two memories to keep in step. The text agent already reached the calendar, reminders, lists and the house.
Replies through the living room's good speakers	The soundbar input cannot be controlled, and casting to the television interrupts whatever is playing. Replies stay on the puck.
Keep the default voice	The stock voice engine is built for speed, not warmth. The household called it "very mechanical."

Why give the house a character

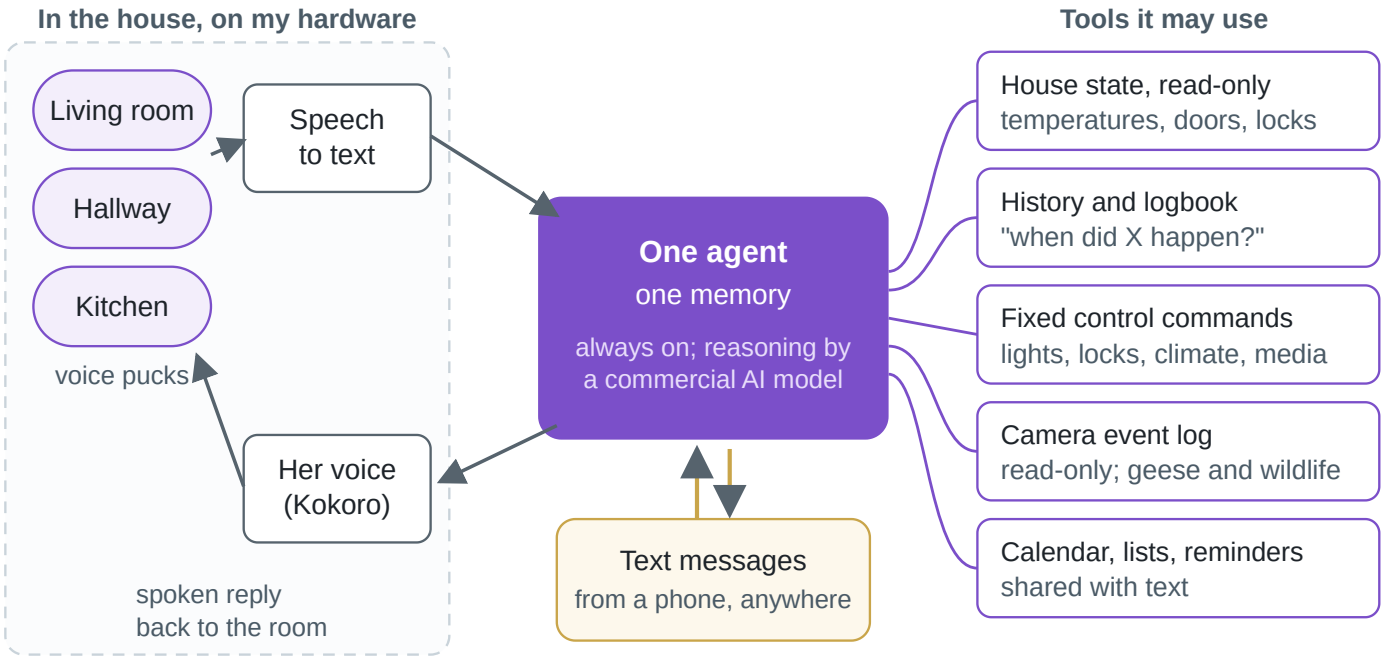
A house voice speaks to everyone in the household many times a day, so how it sounds and how it phrases things matter as much as what it knows. The diagnosis in July 2026 named two separate dials. The words come from the instructions the AI follows, and the sound comes from the voice engine. A character prompt changes how the voice feels more than any change of timbre, and costs nothing to run. We turned both dials, one change at a time, and verified each by speaking to a puck rather than by reading configuration.

The decision

Voice became a second door into the same always-on agent that answers my text messages, with one memory behind both. The small local model stays installed as a fallback the household can switch back to in one setting.

2. How it works - one brain, two mouths

1. Wake	2. Transcribe	3. One agent	4. Look or act	5. In character	6. Speak
The puck listens for its wake phrase on the device itself.	Whisper turns speech into text on my workstation's graphics card.	The text joins the same session that answers my messages, tagged with the room.	The agent reads house state and history, or acts through a fixed command.	Agrippina composes one short spoken sentence.	Kokoro renders her voice locally and the puck plays it.



The pucks and the phone are two doors into one agent. Speech recognition and her voice stay on my hardware; the agent's reasoning runs on a commercial AI model.

Questions she answers from real history

Asked at a puck	What she reads before answering
"When were the geese last on the dock?"	The camera event log, for the most recent bird detection.
"When did the heat run last night?"	Thermostat history, as heating cycles with start times, typical length and total hours (added November 2026).
"Are the doors shut?"	The door sensors. If one is not reporting, she names it and when it was last heard.
"Turn on the light."	Nothing to read. She assumes the room the request came from and runs one fixed command.
"What did I add to the list?"	The same list the text agent wrote to when the item was texted from the store.

One session, one memory - and the trade

Voice and text run in one always-on session, so the agent handles one request at a time and remembers both. That is why a texted item can be asked about at the sink. The cost is speed. In June 2026 testing, replies that used tools took 10 to 37 seconds, so I raised the time limit to 42 seconds and she says "working on it" first when a task runs long.

3. Designing the character, her name and her voice

The character brief is a few paragraphs of instructions the agent loads for every voice request. It sets who she is and, just as important, how little of the character to use.

> *"You are Agrippina, the intelligence of this house - imperial Roman matriarch in composure, with a quiet, curious delight in the patterns of the household... Rank shows as calm economy ("It is done."), never as coldness."*
>
> Character brief, July 2026. Asked who she is, she answers "I am Agrippina. I keep the pattern of this house."

The register adapts to the moment



At most one flourish per reply. Most replies carry none.

The character lives in word choice, not added sentences. A person waiting at the sink for the temperature gets the temperature.

Rules that keep replies short

1. One spoken sentence is the norm; no monologue and no theatrics.
2. No markdown, lists or web links read aloud.
3. No honorifics. Never "madam" or "sir."
4. Names of the household used sparingly and warmly.
5. The text agent has its own personality; it never leaks into her voice.

How we chose the name

We screened 20+ names against four tests. A name must be easy to call out loud, collide with no everyday phrase, carry no trademark risk from a living fiction franchise and fit the house. Theodora failed ("the door" is said daily in a smart home), Alexandra contains another assistant's name and Livia sounds like "Olivia." Agrippina was the mother of the emperor Nero, and Nero is the name of the platform she runs on.

From a mechanical voice to a chosen one

	Before (June 2026)	After (July 2026)
Engine	Piper, the stock local engine, built for speed	Kokoro, a more natural open-source engine, also local
How it was picked	Default setting	8 voices rendered from one script; 6 auditioned live on a puck, then a head-to-head of the final two. The household chose a voice called Heart.
Where it is set	Per assistant path	One setting for the pucks and the hallway display
Speed	-	First audio in 0.3 to 0.6 seconds for a short sentence, on the processor alone
Undo	-	The old engine stays installed; switching back takes four clicks

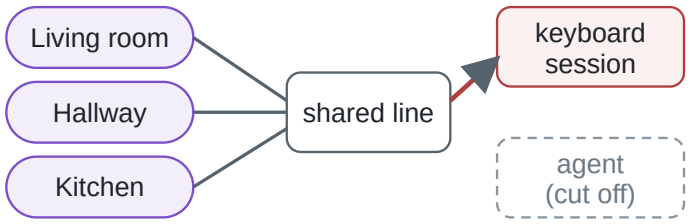
4. Safeguards

A voice that can unlock doors needs limits written in code, not in good intentions. These are the eight that apply to every voice request.

#	Safeguard	What it does
1	**Wake on the device**	Wake-phrase detection runs on the puck itself.
2	**Speech stays home**	Speech recognition and her voice run on my own workstation. Only the reasoning step uses a commercial AI model.
3	**Short-range microphones**	We tested the eavesdropping concern directly. The pucks pick up speech only indoors and at short range.
4	**Fixed commands only**	House queries are read-only. Control runs through a short list of fixed verbs for lights, switches, scenes, climate, media and locks. Anything outside the approved list stops for a person's approval.
5	**Alarm rule in code**	She can arm the alarm. She can disarm it only when the code confirms a household phone is home. The script decides, not the AI (ruled September 2026).
6	**Honest door answers**	If a door sensor is not reporting, she says which one and when it was last heard, and she never calls the house secure (October 2026).
7	**One set of limits**	The same written limits govern voice and text, so any new device connected later inherits them.
8	**Every request gets an answer**	A room hears only what she sends through a dedicated speak command. Early tests produced silent non-answers, so the instructions now require that command for every voice request.

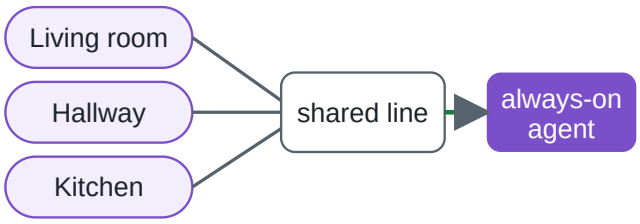
5. Two reliability lessons

Before - any session could grab the line



all three rooms went silent at once

After - one owner for the shared line



keyboard sessions can no longer open it

One shared line for the whole house (July 2026)

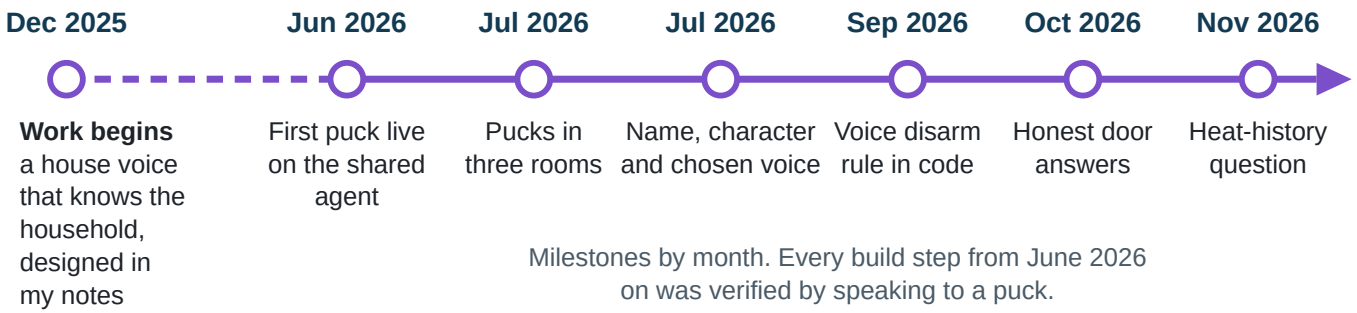
Every keyboard session opened on the workstation started its own copy of the voice connection, and that copy took the line the pucks use. All three rooms went silent. The fix gives the line one owner, the always-on service. When three rooms share one path, one fault silences all of them, so the shared path needs a single owner.

One voice, set in one place (July 2026)

The first switch to the new voice landed on the fallback path, so the pucks never used it, and a hidden per-path setting would have overridden the choice anyway. We moved the voice to a single setting and confirmed it by listening at a puck. A change is done when you hear it in the room.

6. Results and status

Agrippina is live in three rooms. The work began in my notes in December 2025, and the first puck went live in June 2026. Each build step shipped as its own reversible change, with the previous setting kept as the undo.



Milestones by month. Every build step from June 2026 on was verified by speaking to a puck.

Item	Status	Detail
Voice pucks on the shared agent	Live	Since June 2026; three rooms since July 2026.
Character and chosen voice	Live	Since July 2026, after a live audition on a puck.
Questions from real history	Live	Door and lock history, the camera event log and, since November 2026, heating cycles.
Custom wake phrase in her name	Not deployed	The pucks still trigger on television audio at their least sensitive setting. A first training run for a custom phrase did not finish. The stock phrase stays until a custom one triggers falsely less often in testing.
Instant path for simple commands	Planned	Let the house handle "turn on the lights" locally, without the agent, to cut the wait.
A "thinking" sound	Designed	A short in-character clip to fill the pause while she works on a longer answer.

What is next

The open work is speed and the wake phrase, not reach. Answers from history already work; the long pause on simple commands is the main thing a member of the household notices, so the instant path for simple commands comes first.

7. What this shows about how I work

I treated a voice assistant as a program with users, rules and a release process rather than as a gadget. I wrote down what she may do, set the limits in code and kept every change reversible. I chose one shared system over two parallel ones because two systems drift, and I verified each change the way a user would, by asking a question in the room. The same habits apply to running a program with partners or helping an organization adopt AI. Write the rules first, keep one source of truth and judge the result where the user meets it.

My part

I specified that every device would share one memory, set the limits and the alarm rule, and verified the results by voice. The household chose the name and the voice. AI coding agents wrote the connection between voice and text, the tools and the voice service.