On-device AI that scales

Glenn Sonna · Xybrid



Cloud: one key, one call.
Local: which model, runtime, format, chip?

CLOUD



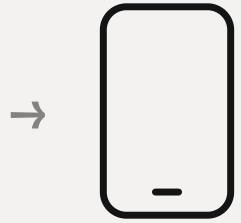
LOCAL, INSIDE AN APP

which model?

which runtime?

which format?

which chip?

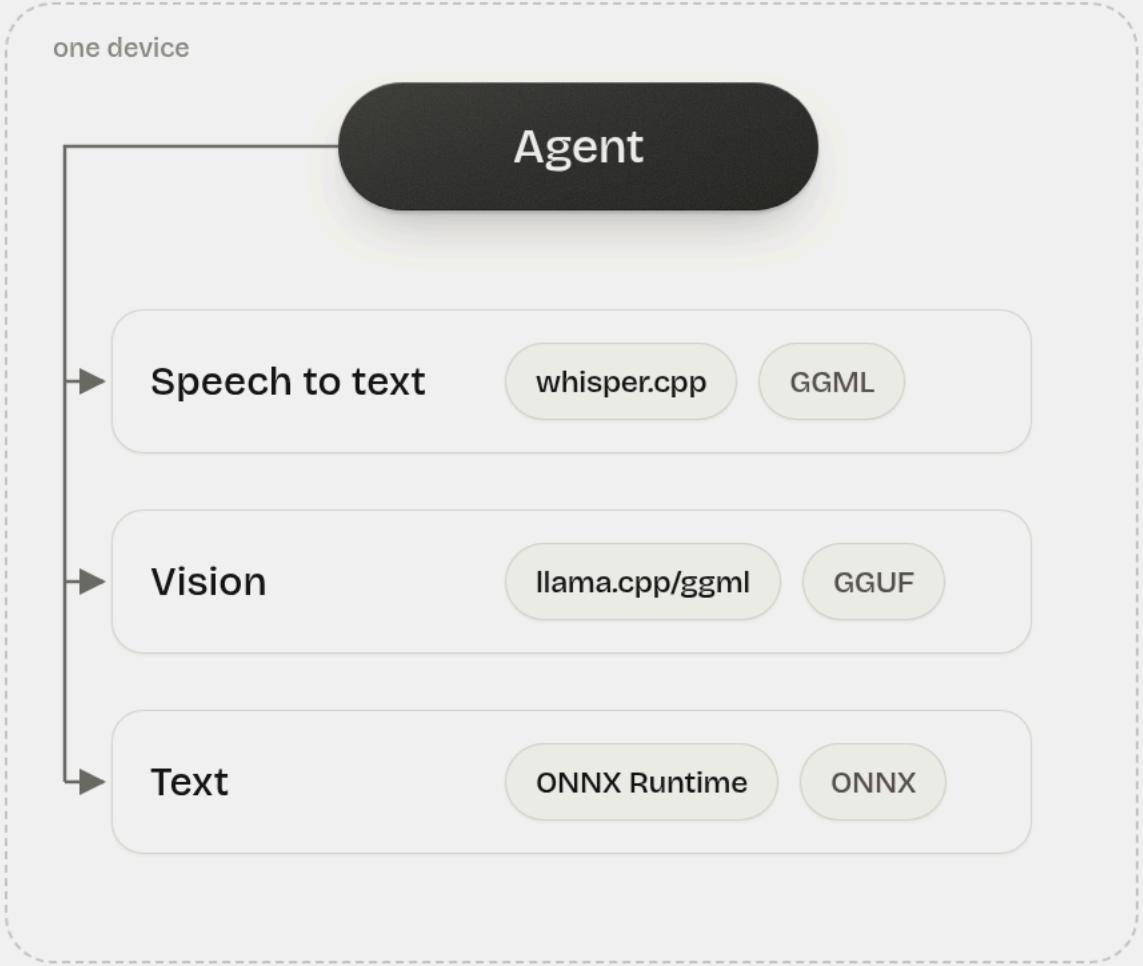


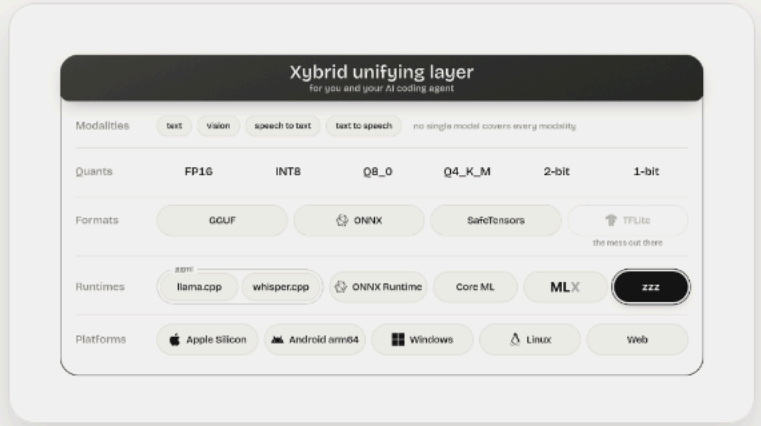
Xybrid unifying layer

for you and your AI coding agent

Modalities	text	vision	speech to text	text to speech	no single model covers every modality	
Quants	FP16	INT8	Q8_0	Q4_K_M	2-bit	1-bit
Formats	GGUF	ONNX	SafeTensors	TFLite	the mess out there	
Runtimes	ggml: llama.cpp, whisper.cpp	ONNX Runtime	Core ML	MLX	zzz	
Platforms	Apple Silicon	Android arm64	Windows	Linux	Web	

One agent, one device, several models.





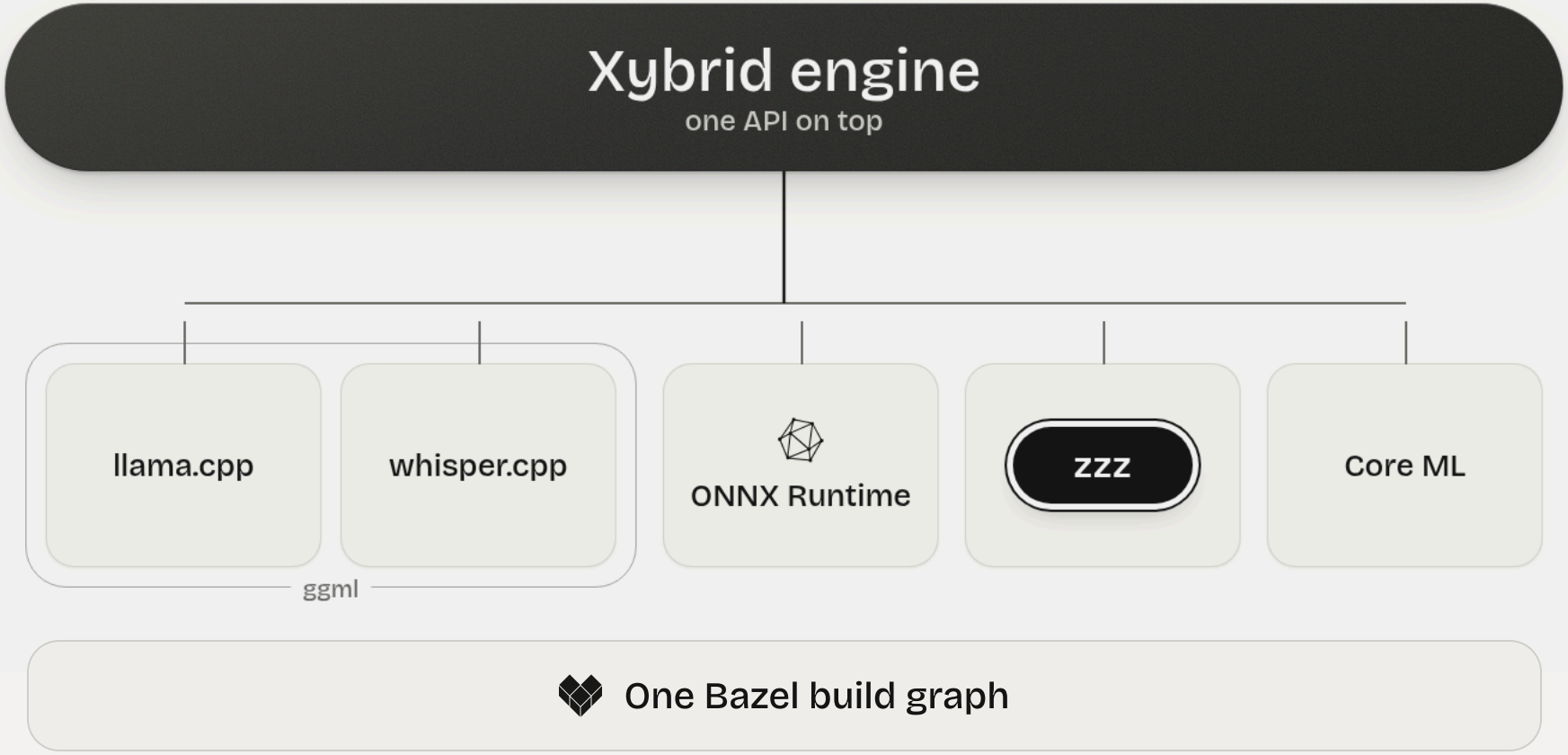
Each model brings its own layers.



We are Xybrid.

Make on-device AI ubiquitous.

Inside Xybrid: one engine, pluggable runtimes.



We sit above the engines. We don't replace them.

Runs where you ship.

SDKs	Flutter, Swift, Kotlin, React Native, Unity, Rust
Platforms	iOS, Android, macOS, Linux, Windows

In the cloud you see every request.
On a device, you ship and go blind.

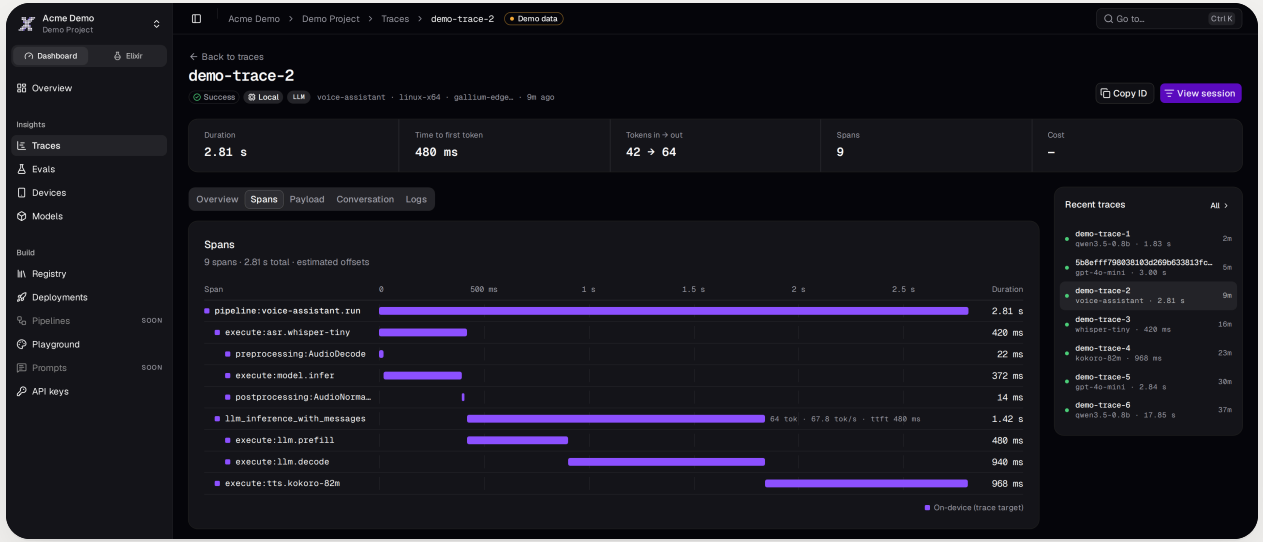
Observability across the stack

model load · inference · tools · workflows

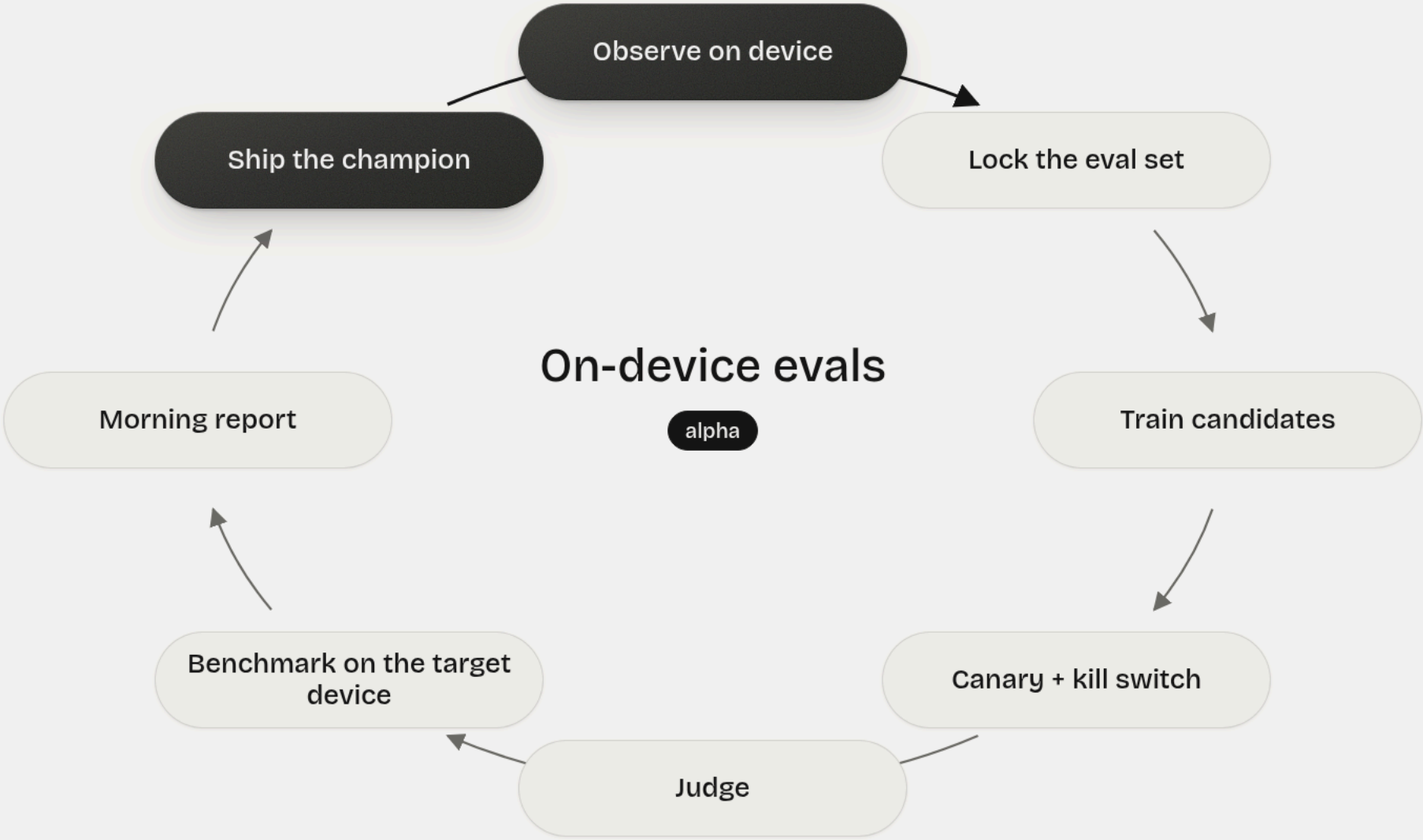
Open standard: OpenTelemetry

opt-in export · nothing leaves the device unless you turn it on

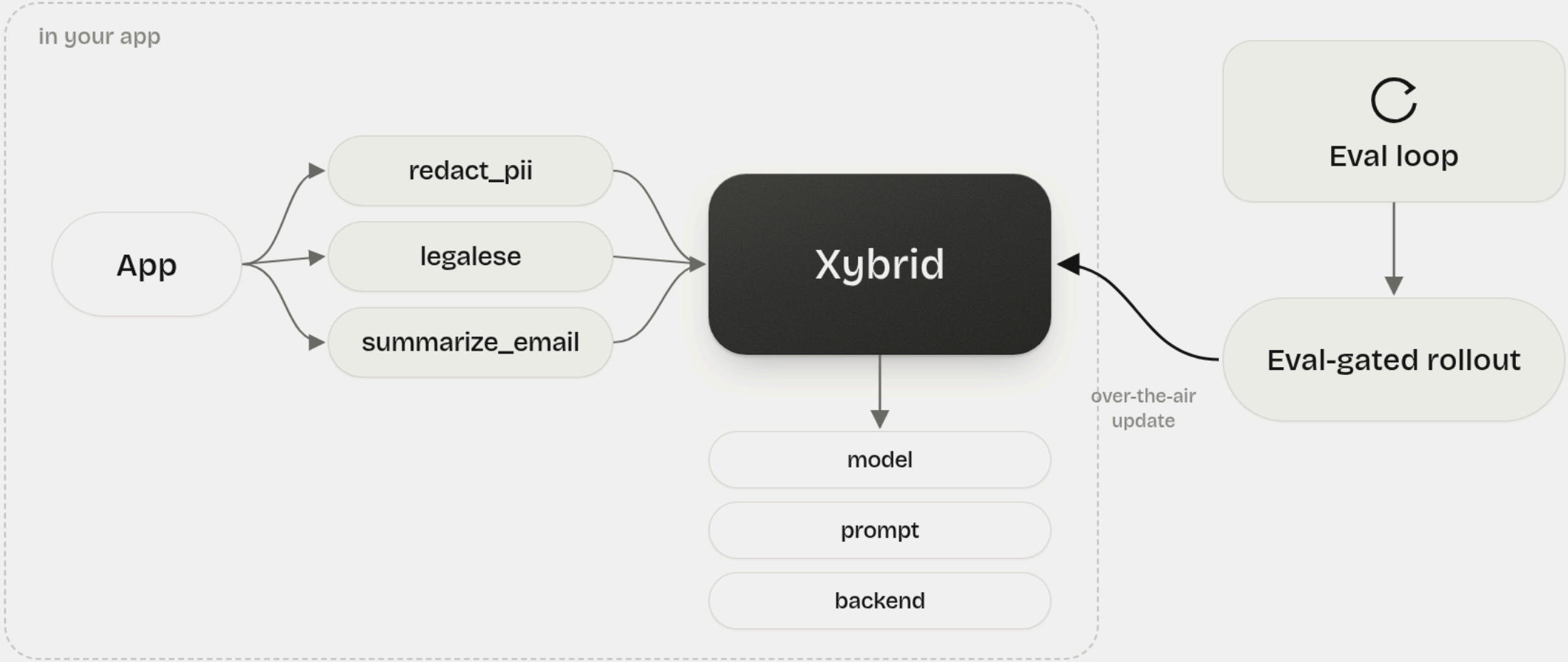
In the SDK today



Observe and evaluate
in one loop.



Tasks: your app picks one, Xybrid resolves the rest.



Resolved for each device.

zzz our engine for tiny devices.

New models, supported fast

Performance we control

No third-party runtime in the way

~1 MB

what zzz adds to the Android app library

zzz our engine for tiny devices

Phonon

17×

real time

KittenTTS 2, speech-token stage

2.4×

vs the lab's reference runtime

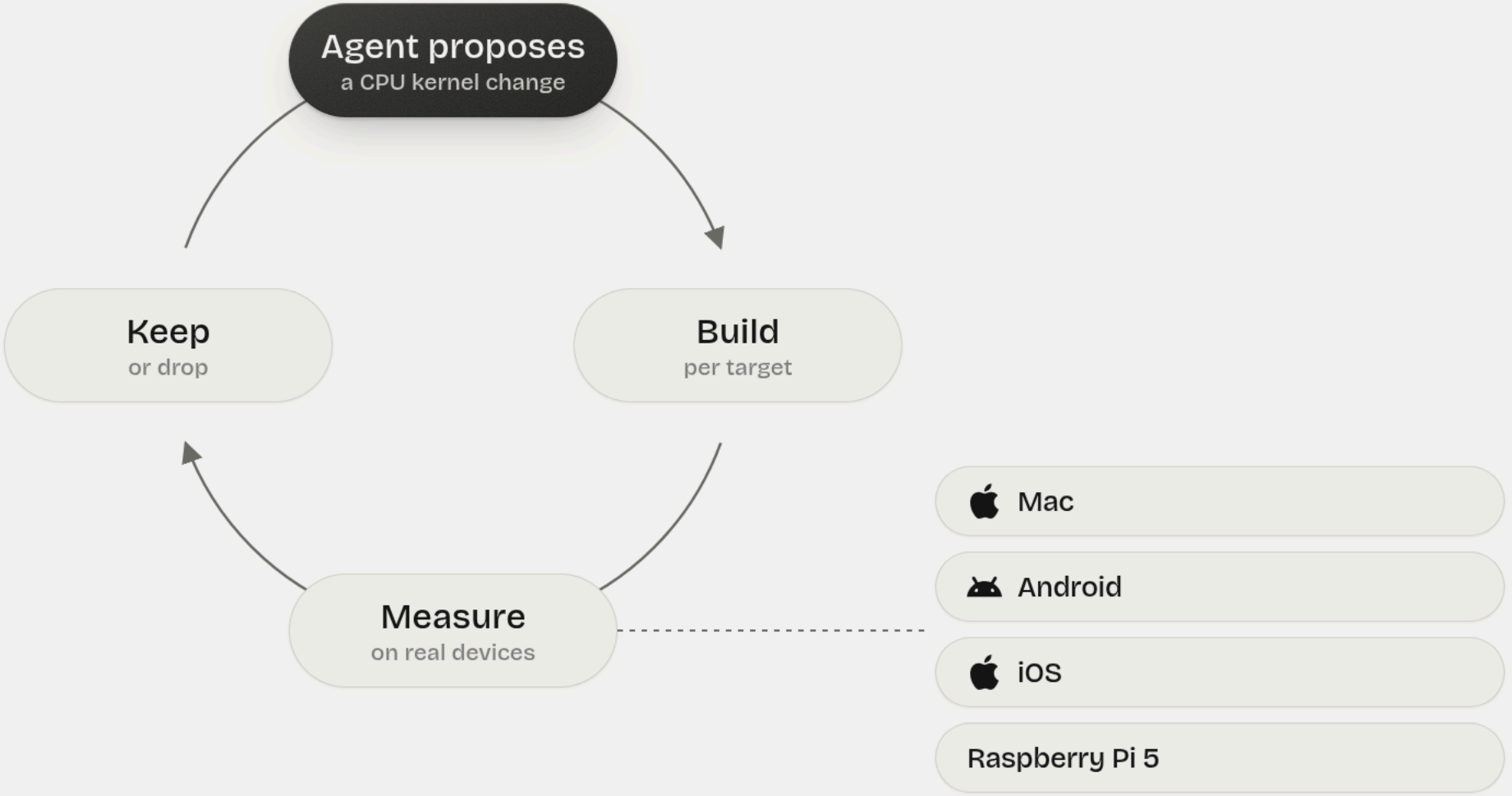
Bonsai 1.7B, text generation

2.35×

vs llama.cpp

Public bench: github.com/xybrid-ai/zzz

Agentic kernel optimisation.



On-device AI is a new topic.

There's still a lot to figure out.

All these experiments have one goal: push the boundaries of what's possible on-device.



Three models, one flow, **fully on-device.**

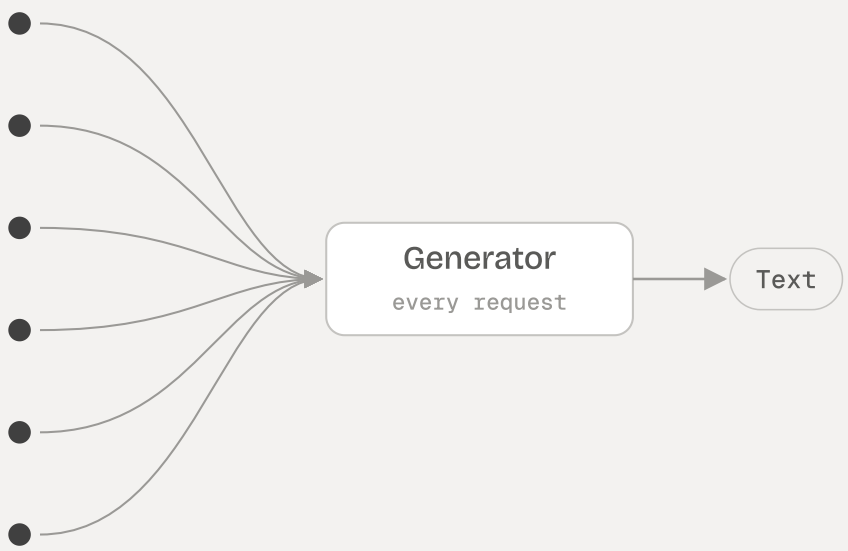
Xybrid coordinates a Liquid model, Whisper and Phonon, with tool calls, on an M1 MacBook.



Decisions instead of generation.

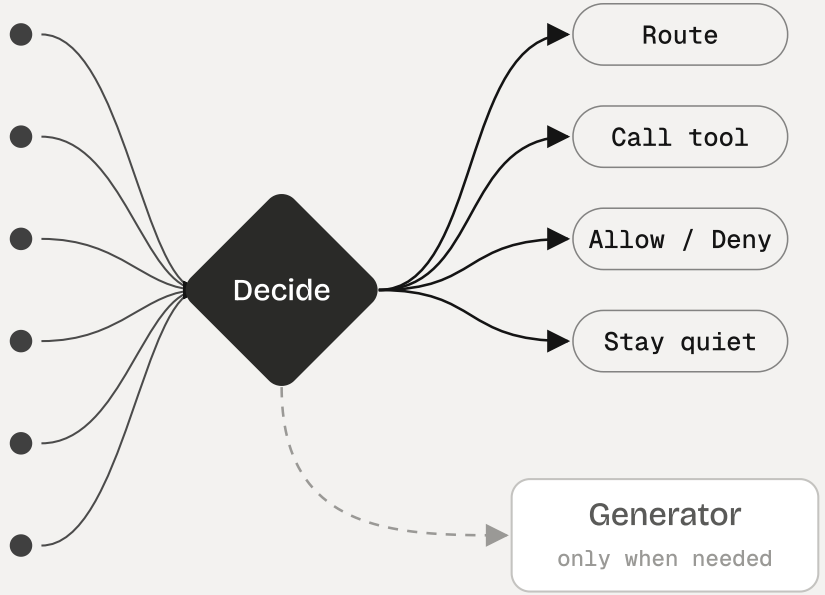
A small decision model answers the closed questions. Generate only what's left.

Generate everything



every request pays for generation

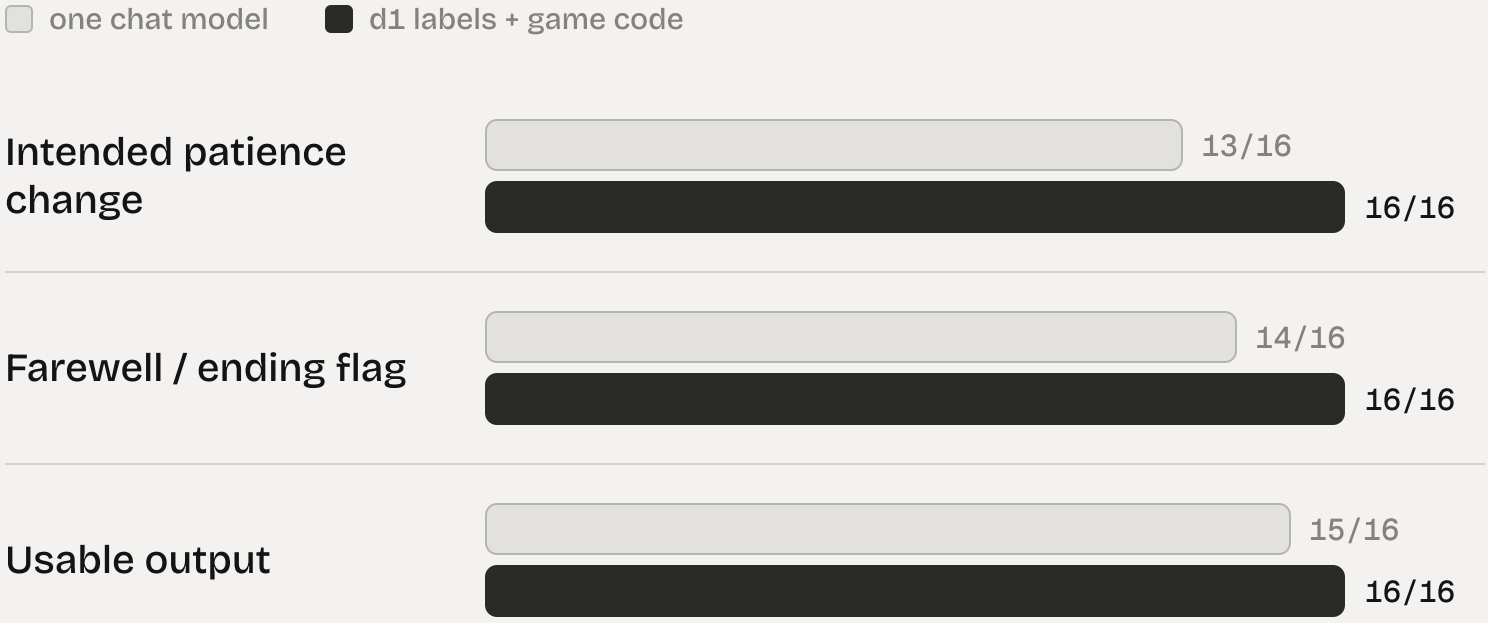
Decide, then generate



generate only what's left

Let code own the effects: 16 of 16.

Trola, a troll gatekeeper in a small game. The same 16 player lines, two ways.



16 player lines · answers written down first · one seed each

7

questions per player line, answered in one pass

All

of d1's labels matched the answers written down in advance

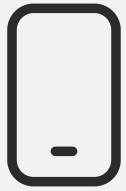
Real-time reactions: presence and emotion.

An experiment: lip-sync from the voice model's phoneme timings.



AI will be everywhere.

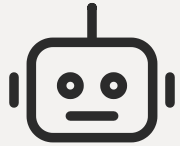
For every app and every agent, on every device.



Mobile



Wearables



Robotics



xybrid.ai

Make on-device AI ubiquitous.

github.com/xybrid-ai/xybrid