

FIELD NOTES · AI INFRASTRUCTURE

TPU Rationing Is an Authority Problem

How Google decided who gets scarce compute — and where the rest of the demand went.



RACK2CLOUD
Think Like An Architect.
Build Like An Engineer.

Think Like An Architect. Build Like An Engineer.

The Shortage Is Real

PHYSICAL CONSTRAINT

High-bandwidth memory supply
from a small number of manufacturers

"the supply-constrained environment"

— Anat Ashkenazi, Alphabet CFO

Not a signal problem. Not an unvalidated forecast.
A physical ceiling — genuine, and shared across the industry.



Someone Has to Decide Where It Goes

TIER 1 — AGI FRONTIER RESEARCH

"The foundation for everything we do."

TIER 2 — CORE PRODUCTS

Search, YouTube

TIER 3 — CLOUD CUSTOMERS

Whatever capacity remains



OVERFLOW — DISPLACED DEMAND

Sundar Pichai stated this hierarchy directly to analysts on Alphabet's Q2 2026 earnings call.

The Decision Has a Downstream Address



March 2026 — Google told Meta it could not supply the Gemini capacity Meta had requested.

Meta instructed staff to conserve their own AI usage in response.

Allocation authority doesn't just decide who's served first — it decides who has to look elsewhere.

Bridge Capacity Is the Admission

\$920M / month

~110,000 Nvidia GPUs · xAI's data centers

Google's own term: "bridge capacity"

A bridge connects two points that aren't naturally joined. Calling this bridge capacity is Google's own admission that its internal architecture can't absorb all the demand it's being asked to serve.

This Is Not Phantom Capacity

PHANTOM CAPACITY

Framework #171

A capacity signal is treated as validated demand when nothing has actually checked it.

A story about trust.

TPU RATIONING

This Post

Capacity and competing demand are both real. Supply just can't cover them at the same time.

A story about arithmetic.

Related resource. Different failure.



Compute Can Always Be Found Elsewhere.

What changes, once the original allocation can't satisfy demand, is who ends up paying for the gap — and on whose terms.

