

Valentin Bonnet

San Francisco, CA | valentinvbbonnet@gmail.com | linkedin.com/in/valentinbonnet | github.com/vbonnet | vbonnet.ai

SUMMARY

Staff-level Tech Lead with 12+ years building developer infrastructure at Google and Alphabet. Shipped core systems powering 100K+ build targets, migrated YouTube backends from monolith to microservices, and led company-scale IAM and access governance for Verily's developer platform. Led cross-organizational technical programs — aligning 7 organizations around a 127-page architecture design, managing a strategic vendor relationship, and shaping Verily's Agentic SDLC initiative for 2026. Current focus: AI agent infrastructure — agentic coding harnesses, evaluation frameworks, and safety protocols that give non-deterministic LLM output measurable correctness guarantees. Writing about AI agent engineering at vbonnet.ai.

SKILLS

Languages: Go, Python, TypeScript, Java, C++, SQL

AI / Agent: LLM integration, agentic patterns (tool use, orchestration, context mgmt), MCP, prompt engineering, agent safety

Infrastructure: SDK/API design, distributed systems, Bazel, CI/CD, gRPC, Protobuf, GCP

Security: IAM, RBAC, identity providers, SCIM, access governance, compliance frameworks

Dev Tools: CLI tooling, developer platform engineering, IDE extensions, SDK design

EXPERIENCE

Senior Software Engineer / Area Tech Lead, Verily (Alphabet)

2022 – Present · Developer Platform · Hands-on technical leadership

- AI coding experiments influenced the definition of Verily's Agentic SDLC workstream — a company-level engineering initiative launching in 2026 to adapt the software development lifecycle for AI agents.
- Led technical strategy and architecture for **Grouper**, Verily's company-wide replacement for Aegis (~1,400 employees), producing a 127-page TDD that aligned 7 organizations with conflicting requirements (WIAM, SecOps, Cloud, IT, Compliance, OneVerily, Dev Platform) and directly unblocked a working prototype.
- Drove all key architectural decisions — Opal vendor selection (over build-in-house), custom orchestration layer, multi-phase rollout — and served as Verily's primary technical contact with Opal: ran weekly design sessions, shaped their product roadmap, and co-designed platform features to meet healthcare compliance requirements.
- Owned IAM and Data Access Governance architecture for Verily's developer platform, significantly reducing access-review cycle time through automated policy enforcement.
- Architected MCP server boundaries, permission model, and evaluation harness for ViDA — an AI-powered developer assistant providing agents secure, scoped access to internal systems.
- Shipped evaluation framework for non-deterministic LLM output — product teams now gate agentic features on measurable correctness thresholds before release, measurably improving pre-production regression coverage.
- Created the Feature Development Lifecycle framework (H1 2025), adopted by the Dev Platform team in H2 with positive feedback; mentored multiple junior engineers and led architecture reviews on cross-team projects.

Tech Lead, Google — Core Infrastructure (Boq)

Nov 2021 – Aug 2024

- Led the Boq Conformance & Isolation team within Google's build and deployment platform.
- Designed, built, and maintained Bazel Rules powering 100K+ build targets across Google.
- Developed multi-quarter roadmaps to catch errors earlier in the dev loop, lowering CI/CD costs org-wide.

Tech Lead, Google — YouTube Gaming

May 2018 – Nov 2021

- Led migration of YouTube Gaming backends from Python monolith into C++ microservices. Improved release velocity from weekly to daily.

Software Engineer, Google — YouTube Paid Digital Goods

Jan 2016 – May 2018

- Designed and launched backends for YouTube Live monetization (Super Chat, Channel Memberships) — sub-second latency to millions of concurrent global requests.

Software Engineer, Google — YouTube Live / Google+

Jan 2013 – Jan 2016

- Developed PlayStation and third-party API integrations. Built shared code registry framework adopted by all Google iOS apps.

PERSONAL PROJECTS & WRITING

dear-agent — github.com/vbonnet/dear-agent

Open-source meta-harness for autonomous AI coding agents (Claude Code, Gemini CLI, Codex CLI). SDK-style adapter interface, closed-loop safety protocol (DEAR: Define → Enforce → Audit → Resolve), persistent cue-based memory, MCP plugin architecture. Go, Apache 2.0.

Technical Writing — vbonnet.ai/blog