Nicolas Maman Duffour

Senior Backend Engineer · Go, Erlang · Low-Latency Distributed Systems

Montevideo, Uruguay (Remote, UTC-3) · nicolasmd87@gmail.com · linkedin.com/in/nmd1987 · github.com/nicolas-maman · nicolas-maman.github.io

PROFESSIONAL SUMMARY

Senior backend engineer with 15+ years building low-latency, high-throughput distributed systems in Go and Erlang. Owned the core service of a real-time ad-spend pacing system handling 1,000,000 requests per second at 40-microsecond average latency, designed real-time pricing for a live-betting product generating approximately EUR 20M a week, and built the backends of the Central Bank of Ecuador's digital wallet and of an AI document-intelligence platform for oil and gas. Led a team of four engineers and a monolith-to-microservices migration. Creator and co-maintainer of Aether, an open-source actor language and lock-free multi-core runtime written in C.

TECHNICAL SKILLS

Programming Languages: C, Go (Golang), Erlang, Elixir, Java, C#, Python, SQL

Compilers & Tooling: Compiler design, lexer and parser construction, type inference and type checking, C code generation, runtime systems, language servers (LSP), toolchain and SDK development, cross-compilation, FFI bindings

Performance & Concurrency: Multithreading, SIMD (AVX2, NEON), lock-free data structures (SPSC queues, mailboxes), work-stealing and locality-aware scheduling, actor-model concurrency, arena allocators, NUMA-aware allocation, profiling and benchmarking (cycle counting, fprof)

Backend & Distributed Systems: Distributed systems, microservices, API design, REST, event streaming, WebSocket, HTTP/2, Linux, Docker, Kubernetes, AWS (CloudFormation)

Data, Observability & CI/CD: Kafka, RabbitMQ, NoSQL (Aerospike, DynamoDB), Oracle, Elasticsearch, Prometheus, Grafana, GitLab CI, GitHub Actions

Security: Capability-based sandboxing, LD_PRELOAD libc interposition, TLS

AI-Assisted Engineering: Agentic coding with Claude Code, with agents working behind engineering gates: tests and mutation testing, golden traces, benchmarks against reference implementations, and leak checks

Graphics & GPU: OpenGL, Vulkan, Unity, real-time 3D rendering

PROFESSIONAL EXPERIENCE

Software Engineer, Factored Jun 2023 to Present

Current client: Infersoft, AI document-intelligence platform for the oil and gas industry (250M+ pages processed, 20M+ leases reviewed)

- Built the backend that processes millions of land, title, and operational documents (leases, joint operating agreements, pooling orders), classifying 50+ document types and extracting up to 350 data points per document.
- Linked every extracted value to its source page and paragraph, giving land and title teams an audit trail and cutting document review from months to hours.

Previous client: programmatic advertising platform for a global consumer-electronics company (USD 300M+ annual revenue)

- Owned the core service of the platform's real-time budget-pacing system, sustaining 1,000,000 TCP requests per second at 40-microsecond average latency.
- Built the pacing services as Go microservices in a stream-oriented distributed system (Kafka, Kubernetes, Docker, Prometheus, Grafana), modeling linear and non-linear spend curves to set each campaign's budget and track lifetime spend accurately.
- Converted each campaign's paced budget into auction eligibility, deciding when its ads could enter real-time ad auctions so spend stayed on plan for the whole campaign.
- Led the transition from a monolith to a microservice architecture and owned the platform's reliability and scalability through continuous refactoring.
- Mentored engineers and served as rotating Scrum Master in an agile team.

Software Engineer, Superbet / Happening Dec 2019 to Apr 2023

Real-time sports-betting pricing platform (UK market; live-betting product generating approximately EUR 20M in weekly revenue)

- Designed and maintained real-time live-pricing systems in Erlang (with Kafka, AWS, and Elasticsearch) that adjust sports odds based on market demand across all sports and leagues worldwide.
- Met a hard performance budget of 10,000+ requests per second at under 250 ms end-to-end latency. Fed ~10 MB payloads to the downstream mathematical pricing models.
- Designed and delivered production REST APIs in Go exposing pricing-model parameters; automated deployments with GitLab CI pipelines and CloudFormation on AWS.
- Ran the Dockerized production environment and its Prometheus and Grafana monitoring.

Software Engineer / Technical Lead, IoSports Nov 2017 to Dec 2019

- Built the original proof of concept for a real-time live-betting system, the basis for IoSports' acquisition by Superbet / Happening.
- Led a technical team of four senior data and software engineers; owned technical scope and requirements.
- Built REST APIs and WebSocket services delivering real-time prices, counters, and configurable rule-based alerts to professional traders.

Software Engineer, Inswitch Solutions Aug 2010 to Jun 2017

- Built the digital wallet for the Central Bank of Ecuador, the first nationwide centralized digital-currency and mobile-payments platform, handling hundreds of transactions per second in Erlang on Oracle Database.
- Led a performance-driven refactor (v2.0) using profiling data (fprof, observer) to improve throughput and usability.

Game Developer, Razor Edge Games Jul 2015 to Aug 2016

- Designed a modular game-rules engine and a Unity SDK (C#) enabling 60+ distributed developers to collaborate on a shared codebase.
- Built real-time weather, wind, and cloth-rigging systems in the Unity engine, plus an XML-based interface making game rules fully moddable.

OPEN SOURCE PROJECTS

Aether: Compiled Programming Language & Runtime (Creator and Co-Maintainer, since 2025)

Open-source (MIT) actor language that compiles to C · github.com/aether-lang-dev/aether · aether-lang.dev

- Designed and built a statically typed, compiled language from scratch: lexer, parser, type checker with type inference, and a C code-generation backend chosen for portability, native performance, and direct C interoperability.
- Designed and shipped its ownership and memory-safety model from a written specification, verified by a generated test grid under Valgrind and macOS leaks.
- Engineered a multi-core partitioned runtime: locality-aware actor scheduling with work stealing, lock-free SPSC queues and cross-core mailboxes, arena allocators and memory pools, NUMA-aware allocation, and AVX2 SIMD batch paths with scalar fallback, tuned with per-core cycle counting (message batching, coalescing, zero-copy delivery).
- Built the developer tooling: a language server (LSP), the VS Code extension, the ae CLI (build, run, test, format, packages), and a cross-language benchmark suite comparing Aether's actor runtime with 10 other languages on the Savina patterns.
- Co-developed the SDK and security layers: --emit=lib shared libraries with typed Python, Java, and Ruby bindings; a capability sandbox spanning compile-time module gating, lexical scope denial, and an LD_PRELOAD libc shim; and a standard library with an HTTP/1.1 and HTTP/2 server, TLS, WebSocket, cryptography, and JSON.

EDUCATION

Software Engineering coursework (40 of 60 credits), Universidad ORT Uruguay

REFERENCES

Verifiable peer references at nicolas-maman.github.io/references.html