Damien House - Principal AI Platform Architect & Fractional CTO - Capability Statement

Enterprise AI agent platforms do not fail because organizations lack capable models. They fail because they lack the governed substrate required for safe operation at scale: controlled context surfaces, enforced tool boundaries, tenant-isolated memory, auditable authorization paths, and human-in-the-loop checkpoints. Damien House builds that substrate.

As a principal AI platform architect, fractional CTO, and senior engineering advisor, Damien helps organizations move from fragile prototypes to governed production systems that can be audited, secured, and scaled. Through 109.studio, he combines 20+ years of platform modernization, DevSecOps, Drupal architecture, and technical leadership with hands-on R&D in Agent Experience (AX), Model Context Protocol (MCP), context engineering, retrieval-backed systems, and AI-augmented delivery. He is strongest where enterprise complexity appears quickly: multi-step workflows, multi-tenant services, explicit tool boundaries, session state, privacy, security, auditability, PII/PHI boundaries, end-user safety, and compliance defensibility.

Key Capabilities

- **Multi-Tenant Agent Platform Architecture:** Design tenant-isolated agent services, MCP server registries, scoped tool catalogs, session-state boundaries, agent identity patterns, and namespace isolation models that reduce cross-domain data leakage risk and create reusable foundations for enterprise-scale assistants and workflow agents.
- **Context Engineering, RAG & Content Intelligence:** Treat the LLM context window as a governed enterprise resource using retrieval-backed grounding, source attribution, memory tiers, scratchpad isolation, context compaction, token-budget controls, and permissioned retrieval to reduce hallucination risk, token waste, and tenant bleed-through.
- **MCP Tool Exposure, Skill Governance & Interoperability:** Build MCP-compatible interfaces that expose structured content, internal APIs, and business actions through typed, authenticated, registry-governed tool surfaces; architect versioned skill packages with approval workflows, security vetting, usage telemetry, and deprecation paths.
- **Guardrails, Evaluation & Compliance Defensibility:** Combine schema validation, policy-as-code, static analysis, RBAC/ABAC, architecture constraints, and test execution with LLM-as-judge evaluation, human-in-the-loop escalation, approval gates, audit trails, and explainable context provenance.
- **Retrieval-Backed Knowledge & Content Operations:** Design RAG pipelines, source-grounded retrieval, metadata workflows, content audit systems, and human review queues that improve answer quality, reduce manual burden, and preserve provenance across large knowledge estates.
- **Cloud-Native Delivery, DevEx & DevSecOps:** Modernize legacy foundations, standardize environments, improve CI/CD, embed code quality and security gates, and leave teams with better tools, documentation, release confidence, and the engineering substrate required for safe agentic automation.

Differentiators

- **Governed Intelligence, Not AI Theater:** Builds agentic systems around explicit action surfaces, context controls, review paths, auditability, and operating safeguards instead of prompt-heavy wrappers that fail under production pressure.
- **Context Control as Enterprise Governance:** Applies namespace and tenant-isolation discipline one layer higher in the stack, making AI context provisioning as controlled, auditable, and defensible as identity and access management.
- **Platform Thinking, Not Prompt Engineering:** Brings systems architecture judgment to AI: multi-tenancy, skill lifecycle management, context budgeting, tool governance, evaluation harnesses, and operational controls.
- **Prototype Fast, Build to Last:** Uses AI-augmented execution to compress discovery and implementation cycles from 3-to-6 month timelines into 3-to-4 week execution paths while preserving durable architecture, review discipline, and clean operational boundaries.
- **Systems That Leave Teams Stronger:** Improves documentation, environments, reusable platform primitives, delivery workflows, and operational confidence so capability compounds after the engagement ends.

Value Proposition

- **Enterprise AI That Survives Production:** Design agent platforms that are observable, permission-aware, auditable, and operationally stable, with reusable foundations that hold up when real enterprise complexity arrives.
- **Faster Delivery with Lower Technical Risk:** Move from idea to stable system without compounding technical debt, governance gaps, fragile integrations, weak review paths, unmanaged automation, or unclear ownership boundaries.
- **Compliance Defensibility Without Slowing Delivery:** Build audit trails, context provenance, bounded autonomy, review checkpoints, PII/PHI-aware retrieval, and policy-aware tool execution into the platform substrate so AI workflows can move from pilot to production under strict regulatory compliance.
- **Reusable Foundations, Not One-Off Agents:** Establish skill registries, context hierarchies, MCP tool contracts, retrieval patterns, and delivery systems that reduce future implementation risk and improve platform-wide reuse across teams, product domains, and operational workflows.