TECHNICAL WHITEPAPER WP-06 · INTENT-TO-GOAL COMPILATION, ARAZZO 1.1 DAGS & MULTI-PROVIDER VIRTUAL MCP (VMCP)

Intent-to-Goal & Cross-Provider Virtual MCP (vMCP) Compilation: From Plain-English Business Outcome to Governed Arazzo 1.1 Workflows

Arcverin compiles business goals into verified Arazzo 1.1 workflow DAGs, enforces strictest-tier autonomy inheritance across multi-provider vMCP servers, and governs Single, Actor, and Daemon agents.

4-Step CLI	Arazzo 1.1	Strictest Tier	~380 Tokens
INTENT-TO-VMCP COMPILER	**MULTI-STEP WORKFLOW DAGS**	**vMCP AUTONOMY INHERITANCE**	**PROGRESSIVE 1-2-3 BASELINE**
intent profile -> plan create -> capability new -> capability compile.	Deterministic step dependencies, polling & pre-retry reconciliation.	Cross-provider virtual tools inherit the lowest/strictest step tier.	Upfront O(1) discovery footprint across multi-provider API catalogs.

1. The 4-Step Deterministic Intent-to-Goal Compilation Pipeline

Instead of letting LLMs stitch fragile API calls together at runtime, Arcverin compiles business outcomes into verified Arazzo 1.1 DAGs and Worker IR via a 4-step deterministic pipeline:

Pipeline Step	Deterministic CLI Command	Compiled Artifact & Verification Role
1. Goal Profiling	arcverin intent profile <spec> "<goal>" --output ./out	Produces intent.json binding the business outcome to qualified OpenAPI operations.
2. Plan Synthesis	arcverin plan create | arcverin plan validate	Builds and validates plan.json step ordering, data flow & compensation paths.
3. Contract & Overlay Repair	arcverin capability new --outcome | decisions answer <cap>	Emits capability.yaml with OpenAPI 1.2.0 Overlays & signed Owner Decisions.
4. Arazzo 1.1 & vMCP Compilation	arcverin capability compile <cap>	Compiles workflow.arazzo.json, mcp-tool.json & executable worker.ir.json.

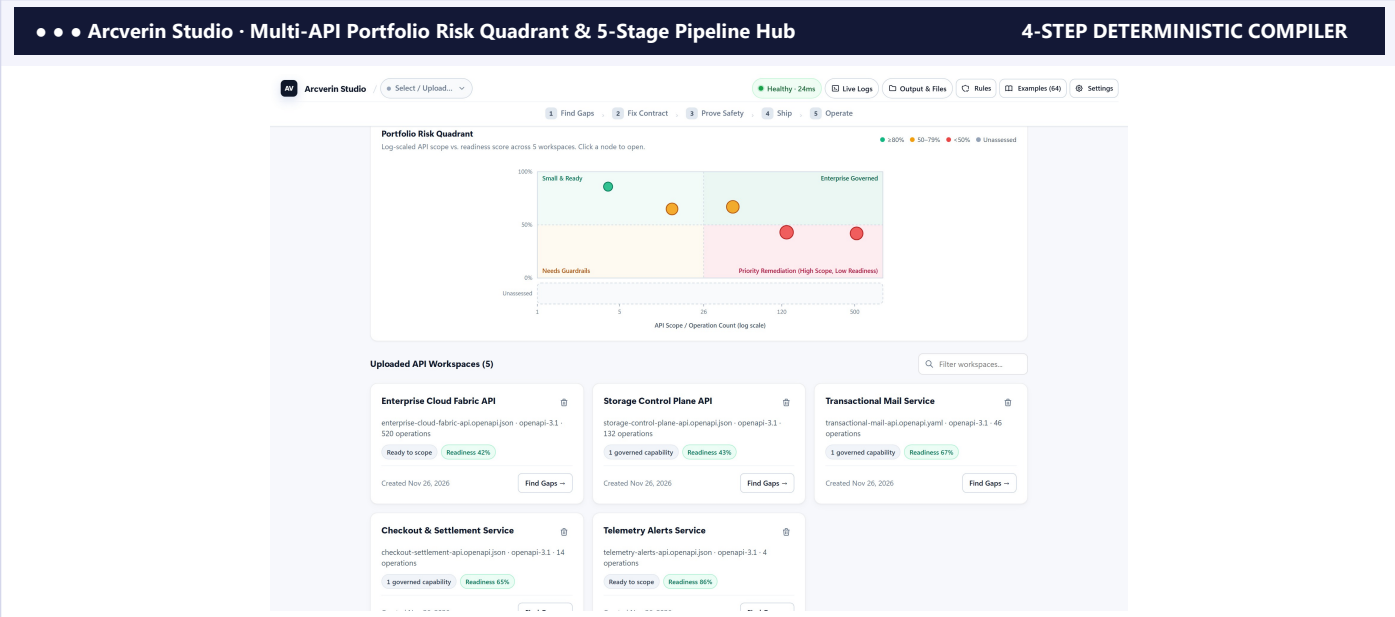


Figure 1: Arcverin Studio Home surfacing portfolio-wide OpenAPI readiness and token footprint across assessed APIs as the entry point to the 5-stage deterministic compilation pipeline.

2. Cross-Provider Virtual MCP (vMCP) Composition & Strictest-Tier Autonomy Inheritance

Enterprise workflows span multiple internal and external services (e.g., Settlement Ledger + Incident Ticketing + Identity Governance). Under the hood, vMCP builds on Arazzo 1.1 sourceDescriptions (referencing OpenAPI and AsyncAPI specs, $self anchors, jsonpath/xpath/jsonpointer selectors, and dependsOn DAGs), compiles into unified WorkerIr, and serves via arcverin-runner serve --vmcp-profile (or serve-vmcp) with Control-Plane compose-and-promote-vmcp. Three mechanics separate vMCP from a raw multi-server MCP aggregator-domain-outcome compaction, strictest-tier autonomy inheritance (if any step is AR2 or AR0, the composed tool inherits that limit), and per-provider Vault isolation across all three agent models:

Agent Architecture	Trigger & Execution Model	Blast-Radius & Approval Gate	Arazzo 1.1 Recovery & Policy Journal
Single Agent	Synchronous request-response tool call via MCP 2026-07-28	Strictest-tier inheritance; inline AR2 approval prompt	Pre-retry GET lookup; single-run SHA-256 journal entry
Actor Agent	Stateful per-entity workflow actor with durable step state	Per-actor Pillar 4 rate/spend cap & session tenant binding	Resumes async 202 polling & compensations across restarts
Daemon Agent	Continuous event/cron-driven autonomous background worker	Requires AR3 or AR4, dual-control commissioning & kill-switch	Deduplicates webhooks & streams OTel/SIEM policy-journal.jsonl

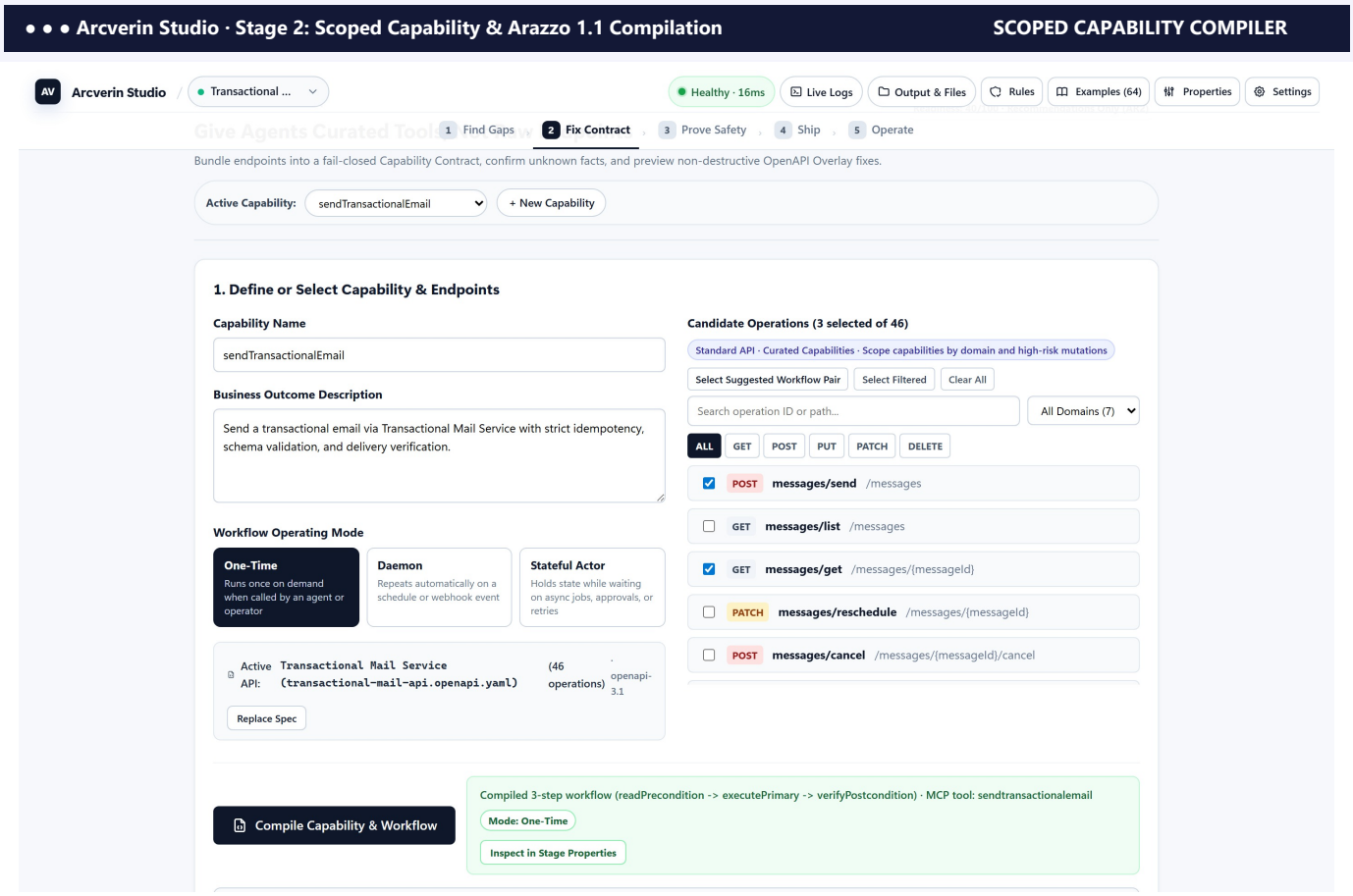


Figure 2: Stage 2 (2. Fix Contract) scoping selected operations from a 46-operation Transactional Mail Service spec into a bounded curated MCP tool and compiled Arazzo 1.1 workflow.

3. Air-Gapped Offline Deterministic Engine + Optional 11-Provider BYOK Semantic Co-Pilot

The Intent-to-Goal compiler runs 100% Offline Deterministic by default for air-gapped defense, finance, and sovereign workloads-zero external network or LLM calls required. Teams can optionally enable any of 11 cloud or local LLM providers (OpenAI, Claude, Gemini, Azure OpenAI, Bedrock, OpenRouter, Groq, Mistral, DeepSeek, Ollama/vLLM, or Custom BYOK) as a Semantic Co-Pilot while deterministic AR0-AR4 rules stay strictly authoritative.

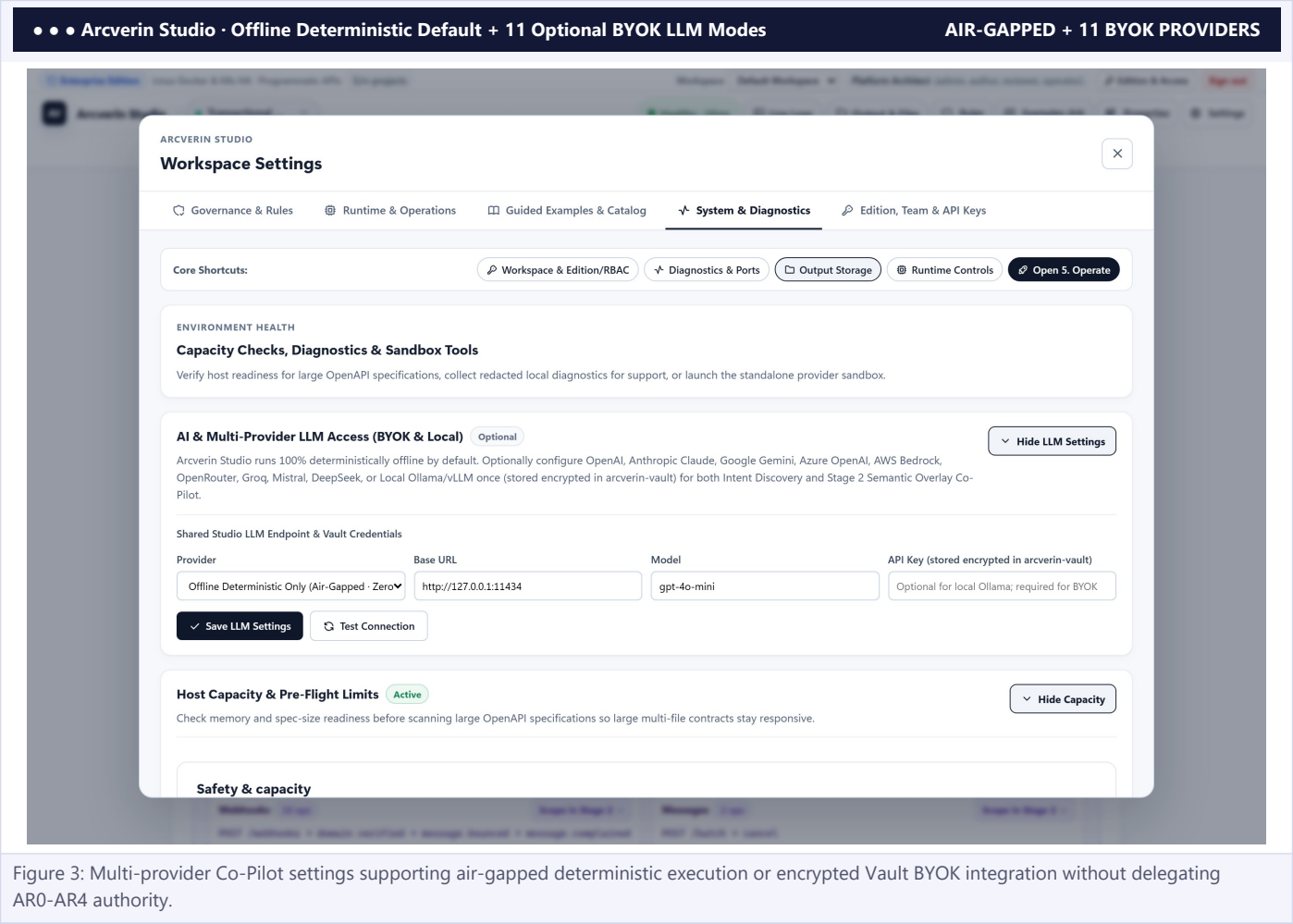


Figure 3: Multi-provider Co-Pilot settings supporting air-gapped deterministic execution or encrypted Vault BYOK integration without delegating AR0-AR4 authority.

4. Practical Engineering Caveats, vMCP Architectural Limits & When to Skip vMCP

Pre-compiled Intent-to-Workflow and Enterprise vMCP DAGs are built for repeatable, high-consequence enterprise procedures-not unbounded ad hoc exploration or implicit distributed transactions. Engineering and security teams should keep these operational boundaries explicit:

Architectural Boundary	What Actually Happens & Recommended Engineering Practice
Offline Lexical Matching	Deterministic intent profile matches OpenAPI paths, operationIds, and descriptions offline without an LLM. Engineers should always review intent.json and plan.json on ambiguous specs or refine scoping in capability.yaml.
Design-Time != Runnable	Compiling plan.json or workflow.arazzo.json does not authorize live traffic. Production runs still require owner attestations, Stage 3 verification (mock, read-only staging, or customer-run imports), DSSE packaging, Vault binding, and Stage 5 commissioning.
Arazzo 1.1 Portability	API steps, polling loops, and pre-retry GET lookups compile into standard Arazzo 1.1; Arcverin runtime guardrails travel in x-arcverin-dw-* extensions paired with x-arcverin-context v1.1.0 (tenantBoundary, recovery) in Overlays. Third-party Arazzo runners execute standard steps while ignoring custom extensions unless enforced via exported OPA Rego v1, Cedar, or gateway policies.
No Automatic 2PC	OpenAPI and Arazzo 1.1 describe call sequences and dependencies, not distributed two-phase commit (2PC). If Step 1 commits on Provider A and Step 2 fails permanently on Provider B, author explicit compensating endpoints or export an Enterprise Multi-System vMCP Temporal Saga bundle (export-only TypeScript & Python .zip with LIFO compensation hooks, N-of-M SoD quorum, and per-provider Vault/OPA hooks) to run on your own Temporal workers.
Enterprise-Only vMCP Gate	Cross-provider vMCP and compose-and-promote-vmcp require Enterprise Edition (Inquire for Pricing), returning HTTP 403 (verifiedVmcpRequiresEnterprise) on Personal and Team. Evaluate single-API workflows on Personal Edition ($99 1-Month Pass / $49/mo billed annually · $588/yr) first.
When to Skip vMCP	Skip vMCP if you already standardize cross-service orchestration in Temporal or an internal aggregation/BFF API (qualify and expose one coarse-grained endpoint instead), or for mostly read-only exploratory queries better served by an MCP gateway or Code Mode sandbox.

Explore the Arcverin Technical Documentation & CLI Reference at https://www.arcverin.com/docs/ →

https://www.arcverin.com/docs/

Compiler determinism note: Intent-to-Goal compilation produces identical workflow.arazzo.json, mcp-tool.json, and worker.ir.json outputs offline with zero LLM dependency; optional BYOK Co-Pilot outputs remain subject to deterministic AR0-AR4 verification.