

Inventory: Agent Fleet

Agent Fleet Inventory — SENTINEL Audit 2026-05-09

Auditor: sentinel-architect

Scope: ~/.claude/agents/ vs specialist-mapping.json vs persona dirs vs chains vs definitions dual-store

Status: READ-ONLY. No files modified.

1. 66 vs 29 vs 12 Reconciliation

Raw counts (tool-verified)

Store	Count	Notes
<code>~/.claude/agents/*.md</code>	66	Includes 0.md, Explore.md, Plan.md as named agents
<code>specialist-mapping.json</code> mappings	29	Key: <code>mappings</code> object
<code>specialist-mapping.json</code> companies	9	ALAI, AgentForge, CodeCraft, Finverge, FlowForge, Proveo, Securion, Skybound, Vizu
Persona dirs in <code>~/system/agents/personas/</code>	12	AgentForge, Axiom, CodeCraft, Datavera, Finverge, FlowForge, Lexicon, Proveo, Resolver, Securion, Skybound, Vizu

Critical gap: 3 persona companies are completely absent from specialist-mapping.json:

- `Axiom` — not in company_summary, zero agents mapped
- `Datavera` — not in company_summary, zero agents mapped
- `Resolver` — not in company_summary, zero agents mapped
- `Lexicon` — not in company_summary, zero agents mapped (persona dir exists, skillforge.md maps to "Skillforge" not Lexicon)

So the real company gap is 4 out of 12 personas have no presence in specialist-mapping.json.

Mapped agents (29 in specialist-mapping.json)

Agent file	Company	On disk (~/.claude/agents/)?
alem-clone.md	ALAI	MISSING
angie-jones.md	Proveo	YES
anthropic-chief-architect.md	AgentForge	MISSING
brad-frost.md	Vizu	YES
bruce-momjian.md	CodeCraft	YES
builder.md	CodeCraft	YES
chip-huyen.md	AgentForge	YES
claude-code-guide.md	AgentForge	YES
codecraft.md	CodeCraft	YES
dorota-huizinga.md	Proveo	MISSING
georgi-gerganov.md	AgentForge	YES
hadi-hariri.md	CodeCraft	MISSING
james-bach.md	Proveo	MISSING
kelsey-hightower.md	FlowForge	YES
lea-verou.md	Vizu	YES
lee-robinson.md	CodeCraft	MISSING
lisa-crispin.md	Proveo	MISSING
markos-zachariadis.md	Finverge	YES
martin-kleppmann.md	CodeCraft	YES
parisa-tabriz.md	Securion	YES
paul-hudson.md	Skybound	YES
petter-graff.md	CodeCraft	YES
proveo.md	Proveo	YES
sentinel-architect.md	Securion	YES
sentinel-ba.md	Skybound	YES
sentinel-developer.md	CodeCraft	YES
sentinel-tester.md	Proveo	YES

Agent file	Company	On disk (~/.claude/agents/)?
sentinel-validator.md	Proveo	YES
skillforge.md	Skillforge	YES

7 agents mapped in specialist-mapping.json but MISSING from ~/.claude/agents/:

1. `alem-clone.md` — exists in definitions/, not synced to ~/.claude/agents/
2. `anthropic-chief-architect.md` — NOT in definitions/ either; completely phantom
3. `dorota-huizinga.md` — exists in definitions/, not synced
4. `hadi-hariri.md` — exists in definitions/, not synced
5. `james-bach.md` — exists in definitions/, not synced
6. `lee-robinson.md` — exists in definitions/, not synced
7. `lisa-crispin.md` — exists in definitions/, not synced

`anthropic-chief-architect.md` is the worst case: mapped in specialist-mapping.json, NOT in definitions/, NOT in ~/.claude/agents/ — fully phantom, cannot be dispatched.

42 unmapped agents (in ~/.claude/agents/ but NOT in specialist-mapping.json)

Classification: ORPHAN = nowhere used | DUPLICATE = covered by mapped peer | NEEDS-MAPPING = used in chains/skills but unmapped

Agent	Classification	Reasoning
<code>0.md</code>	ORPHAN	No name, no description, artifact
<code>agentforge.md</code>	NEEDS-MAPPING	Company persona file; Axiom/Datavera/Resolver equivalents all exist — AgentForge has a persona dir but no company-level mapping entry
<code>backend-builder.md</code>	DUPLICATE	Covered by builder.md (CodeCraft, mapped)
<code>backend-dev.md</code>	DUPLICATE	Covered by codecraft.md + builder.md
<code>baseline-comparator.md</code>	NEEDS-MAPPING	Active agent (Veritas baseline, MLX-backed); used in verify-fix-loop skill; no mapping
<code>code-reviewer.md</code>	DUPLICATE	Covered by petter-graff.md / sentinel-developer.md
<code>code-simplifier.md</code>	DUPLICATE	Covered by sentinel-developer.md
<code>database-dev.md</code>	DUPLICATE	Covered by bruce-momjian.md

Agent	Classification	Reasoning
<code>datavera.md</code>	NEEDS-MAPPING	Company persona file for Datavera (persona dir exists, 0 mapped agents)
<code>design-builder.md</code>	DUPLICATE	Covered by <code>brad-frost.md</code> / <code>lea-verou.md</code>
<code>devils-advocate.md</code>	NEEDS-MAPPING	Pre-action blocker used in 0 chain yamls but referenced in mehanik flow; unregistered
<code>devops-dev.md</code>	DUPLICATE	Covered by <code>kelsey-hightower.md</code>
<code>distiller.md</code>	NEEDS-MAPPING	Used in 21 chain yaml steps (highest after builder/validator); no mapping. CRITICAL gap.
<code>dr-sarah-chen.md</code>	ORPHAN	No description parsed; no chain/skill references found
<code>dzevad-jahic.md</code>	NEEDS-MAPPING	Bosnian linguistic QA (Lexicon company, per <code>CLAUDE.md</code>); not in <code>specialist-mapping.json</code> despite <code>CLAUDE.md</code> routing directive
<code>evidence-verifier.md</code>	NEEDS-MAPPING	Active Veritas agent (gemma-4-26b @ FORGE); triggers on <code>mc.js</code> done for H tasks; no mapping
<code>Explore.md</code>	ORPHAN	Capital E; appears to be a stub
<code>finverge.md</code>	NEEDS-MAPPING	Company persona file for Finverge; persona dir mapped but no company-level agent entry
<code>fix-builder.md</code>	NEEDS-MAPPING	Write-only counterpart to verifier; used in <code>verify-fix-loop</code> skill; no mapping
<code>flowforge.md</code>	NEEDS-MAPPING	Company persona file for FlowForge; only <code>kelsey-hightower.md</code> individual is mapped
<code>frontend-builder.md</code>	DUPLICATE	Covered by <code>lea-verou.md</code> / <code>lee-robinson.md</code>
<code>frontend-dev.md</code>	DUPLICATE	Covered by <code>lea-verou.md</code>
<code>fullstack-dev.md</code>	DUPLICATE	Covered by <code>codecraft.md</code>
<code>helixsupport.md</code>	ORPHAN	Role=coordinator; 0 skill/chain references found
<code>indy-dandev.md</code>	ORPHAN	AI research agent (Indian AI + Dan Abramov persona); no chain/skill references; not used in current system
<code>integration-dev.md</code>	DUPLICATE	Covered by <code>codecraft.md</code>

Agent	Classification	Reasoning
jake-wharton.md	NEEDS-MAPPING	Android/Kotlin expert (Jake Wharton persona); no AgentForge/Skybound mapping entry
lexicon.md	NEEDS-MAPPING	Company persona file for Lexicon (documentation company per CLAUDE.md); 0 agents in specialist-mapping.json
maria-santos.md	ORPHAN	No description parsed; no chain/skill references found
mehanik.md	NEEDS-MAPPING	Core orchestration gate; referenced in 7 skill files; CLAUDE.md cites /mehanik command as mandatory pre-dispatch gate; completely absent from specialist-mapping.json
meta-agent.md	ORPHAN	No chain/skill references found
Plan.md	ORPHAN	Capital P; appears to be a stub
proxima.md	NEEDS-MAPPING	Marketing/content agent; referenced in 10 skill files; no company assignment
rag-builder.md	ORPHAN	No chain/skill references; likely superseded by AgentForge rag-tuning-agent.yaml
redzo-reviewer.md	ORPHAN	No chain/skill references found
resolver.md	NEEDS-MAPPING	Company persona for Resolver (persona dir exists, 8 internal agents; 0 in specialist-mapping.json)
securion.md	NEEDS-MAPPING	Company persona for Securion; parisa-tabriz.md + sentinel-architect.md individually mapped, but no company-level dispatcher
skybound.md	NEEDS-MAPPING	Company persona for Skybound; individual members mapped but no company dispatcher
thaer-sabri.md	ORPHAN	No description parsed; no chain/skill references found
validator.md	NEEDS-MAPPING	Used in 44 skill files and 22 chain yaml steps; one of the most-used agents in the entire system; NOT in specialist-mapping.json. CRITICAL gap.
verifier.md	NEEDS-MAPPING	2 skill file references; verify-fix-loop skill; not mapped

Agent	Classification	Reasoning
vizu.md	NEEDS-MAPPING	Company persona for Vizu; brad-frost.md + lea-verou.md individually mapped, no company dispatcher

Summary of 42 unmapped:

- ORPHAN: 10 (0.md, dr-sarah-chen.md, Explore.md, helixsupport.md, indy-dandev.md, maria-santos.md, meta-agent.md, Plan.md, rag-builder.md, redzo-reviewer.md, thaer-sabri.md) — wait, 11 counting redzo
- Actually: 0.md, dr-sarah-chen.md, Explore.md, helixsupport.md, indy-dandev.md, maria-santos.md, meta-agent.md, Plan.md, rag-builder.md, redzo-reviewer.md, thaer-sabri.md = **11 ORPHAN**
- DUPLICATE: backend-builder.md, backend-dev.md, code-reviewer.md, code-simplifier.md, database-dev.md, design-builder.md, devops-dev.md, frontend-builder.md, frontend-dev.md, fullstack-dev.md, integration-dev.md = **11 DUPLICATE**
- NEEDS-MAPPING: **20** (agentforge, baseline-comparator, datavera, devils-advocate, distiller, dzevad-jahic, evidence-verifier, finverge, fix-builder, flowforge, jake-wharton, lexicon, mehanik, proxima, resolver, securion, skybound, validator, verifier, vizu)

Note: counts = 11+11+20 = 42. The original "37 unmapped" figure understates by 5 because it excludes `alem-clone.md` (mapped but disk-missing) and overcounts mapped agents that are actually absent.

2. Persona Dirs Deep Dive

All 12 persona dirs have a consistent structure: `agents/`, `blueprints/`, `brand/`, `CLAUDE.md`, `company.json`, `config.json`, `legal/`, `ops/`, `README.md`, `skills/`, `state/`, `tools/`.

Persona	Has README	Has CLAUDE.md	Has company.json	Agents inside (count)	Owner in company.json	In specialist-mapping.json
AgentForge	YES	YES	YES (domain: AI)	8	N/A	Partial (3 individuals mapped, no company dispatcher)
Axiom	YES	YES	YES (domain: ARCHITECTURE)	5	N/A	NO — completely absent
CodeCraft	YES	YES	YES (domain: DEVELOPMENT)	8	N/A	Partial (6 individuals mapped)

Persona	Has README	Has CLAUDE.md	Has company.json	Agents inside (count)	Owner in company.json	In specialist-mapping.json
Datavera	YES	YES	YES (domain: DATA)	8	N/A	NO — completely absent
Finverge	YES	YES	YES (domain: FINANCE)	9	N/A	Partial (1 individual mapped)
FlowForge	YES	YES	YES (domain: DEVOPS)	10	N/A	Partial (1 individual mapped)
Lexicon	YES	YES	YES (domain: DOCUMENTATION)	9	N/A	NO — skillforge.md maps to "Skillforge" not Lexicon
Proveo	YES	YES	YES (domain: QA)	8	N/A	Partial (6 individuals mapped)
Resolver	YES	YES	YES (domain: SYSTEMIC)	8	N/A	NO — completely absent
Securion	YES	YES	YES (domain: SECURITY)	8	N/A	Partial (2 individuals mapped)
Skybound	YES	YES	YES (domain: PRODUCT)	7	N/A	Partial (2 individuals mapped)
Vizu	YES	YES	YES (domain: DESIGN)	7	N/A	Partial (2 individuals mapped)

Structural finding: All company.json files report `owner: N/A`. No human/agent owner is recorded for any virtual company. This means there is no machine-readable way to route escalation or accountability.

Persona vs mapping mismatch:

- 87 total agents inside persona dirs (sum of agent subdirs across 12 companies) — none of these internal PI agents (builder.yaml, lead.yaml, reviewer.yaml, etc.) appear in specialist-mapping.json. specialist-mapping.json only tracks the "celebrity" individual agents, not the PI agent swarms inside each company.

3. Chain Coverage

Agents referenced in chains

Agent	Times referenced in chains	In specialist-mapping.json?	Disk present?
builder	25	YES	YES
validator	22	NO	YES
distiller	21	NO	YES
sentinel-validator	9	YES	YES
minion	5	NO	NOT in ~/.claude/agents/ (in definitions/ only)
planner	4	NO	NOT in ~/.claude/agents/ at all

Critical: `minion` and `planner` are referenced in chains but have NO corresponding `.md` in `~/.claude/agents/`.

- `minion.md` exists in `~/system/agents/definitions/` but was never synced forward
- `planner` does not exist in `definitions/` or `~/.claude/agents/` — it is a phantom agent referenced in 3 chains (`plan-build.yaml`, `plan-build-review.yaml`, `plan-review-plan.yaml`)

Dead chains (0 references anywhere in skills/ or system/)

Chains that are never invoked via skills or daemons:

Chain	Skill refs	System refs	Verdict
<code>codecraft-api-backend.yaml</code>	0	0	DEAD
<code>codecraft-nextjs-app.yaml</code>	0	0	DEAD
<code>full-review.yaml</code>	0	0	DEAD
<code>minion-bugfix.yaml</code>	0	0	DEAD
<code>minion-docs.yaml</code>	0	0	DEAD
<code>minion-one-shot.yaml</code>	0	0	DEAD
<code>minion-refactor.yaml</code>	0	0	DEAD
<code>minion-security-fix.yaml</code>	0	0	DEAD

Chain	Skill refs	System refs	Verdict
plan-build-review.yaml	0	0	DEAD
plan-build.yaml	~1 (plan-build-test skill ref)	0	BORDERLINE
plan-review-plan.yaml	0	0	DEAD
scout-flow.yaml	0	0	DEAD
securion-security-review.yaml	0	0	DEAD

Note: The `skill-*.yaml` chains in the `chains/` dir are not invoked by name in `skills/`. They appear to be template definitions, not live dispatch chains. Chains are not invoked via a chain runner — skills embed agents directly via `agent:` field inline. The chain YAML format appears to be an aspirational DAG definition language that has no runtime executor wired up.

Effectively ALL 35 chain YAMLs are dead — there is no chain runner in the skill system. Skills call agents directly, not via chain files.

4. Dual-Store Consistency

Files in both `~/.claude/agents/` and `~/system/agents/definitions/`

48 files exist in both stores. ALL 48 are byte-for-byte SYNCED (diff returned empty for every shared file). The sync script at `~/bin/agent-definitions-sync.sh` is working correctly for the files it covers.

Sync gaps

16 files ONLY in `~/.claude/agents/` (not in definitions/) — not covered by sync:

```
baseline-comparator.md
claude-code-guide.md
devils-advocate.md
dr-sarah-chen.md
dzevad-jahic.md
evidence-verifier.md
Explore.md
fix-builder.md
```

```

indy-dandev.md
jake-wharton.md
maria-santos.md
mehanik.md
Plan.md
redzo-reviewer.md
thaer-sabri.md
verifier.md

```

8 files ONLY in definitions/ (not synced to ~/.claude/agents/) — these agents are UNREACHABLE by Claude Code:

```

dorota-huizinga.md    ← mapped in specialist-mapping.json, should be in ~/.claude/agents/
hadi-hariri.md       ← mapped in specialist-mapping.json, should be in ~/.claude/agents/
james-bach.md        ← mapped in specialist-mapping.json, should be in ~/.claude/agents/
lee-robinson.md      ← mapped in specialist-mapping.json, should be in ~/.claude/agents/
lisa-crispin.md      ← mapped in specialist-mapping.json, should be in ~/.claude/agents/
minion.md            ← referenced in 5 chain yaml steps, unreachable
sentry-code-simplifier.md ← not in mapping, not in chains
sp-code-reviewer.md  ← not in mapping, not in chains

```

The first 5 are mapped and therefore expected to be dispatched — they cannot be. Any dispatch attempt for dorota-huizinga, hadi-hariri, james-bach, lee-robinson, or lisa-crispin will silently fail or fall back.

5. Skill ? Agent Linkage

Sample of 10 skills with agent dispatch analysis:

Skill	Agent referenced	Agent in ~/.claude/agents/?	In specialist-mapping.json?
hop-build	No sub-agent dispatch (marker-only skill)	N/A	N/A
build	<code>builder</code> (3 parallel), <code>rag-context-for-builder.js</code> (tool)	YES	YES
code-review	<code>code-reviewer</code> , <code>securion</code> sub-agent, <code>sentinel-architect</code>	code-reviewer YES (unmapped), securion YES (unmapped dispatcher), sentinel-architect YES (mapped)	

Skill	Agent referenced	Agent in ~/claude/agents/?	In specialist- mapping.json?
debugging	No agent dispatch found in instructions	N/A	N/A
deploy-verify	No agent (runs Playwright directly)	N/A	N/A
design-system	No agent dispatch	N/A	N/A
doc-coauthoring	No named agent dispatch	N/A	N/A
fiken-agent	Self-referential meta-skill; dispatches sub-task SKILL.md files	Indirect	N/A
financial-overview	No agent dispatch found	N/A	N/A
incident-response	References <code>securion</code> agent (remediation)	securion.md YES (unmapped dispatcher)	NO

Flags:

- `code-review` skill dispatches `code-reviewer` (unmapped, 44 skill refs) and `securion` (unmapped company dispatcher) directly by name
- `incident-response` references `securion` as a response agent — but `securion.md` is NOT in `specialist-mapping.json` (only individual members are mapped)
- `validator` is the most-used agent (44 skill files, 22 chain steps) with NO mapping entry

Open Questions

1. **Chain runner:** Is there a chain executor anywhere in the system (`~/system/tools/`, `~/projects/`, `pi-orchestrator`)? If not, the entire `chains/` directory is documentation-only, not executable automation.
2. **planner agent:** Referenced in 3 chains (`plan-build`, `plan-build-review`, `plan-review-plan`) but does not exist on disk anywhere. Was it renamed to `distiller` or `mehanik`?
3. **Axiom, Datavera, Resolver:** Three fully-formed virtual companies with persona dirs, README, CLAUDE.md, 5-8 internal agents each — but zero presence in `specialist-mapping.json`. Are these active companies being used via direct session invocation (not via John routing)?
4. **anthropic-chief-architect.md:** Mapped in `specialist-mapping.json`, absent from both `~/claude/agents/` AND `definitions/`. Was this agent removed intentionally or is it a sync failure?
5. **company.json owner=N/A:** All 12 companies have no human owner. Is there a separate ownership registry, or is this a gap in accountability chain?
6. **Lexicon vs Skillforge naming:** CLAUDE.md routing table names the company "Lexicon" and lists "Dževad Jahić" as its agent. `specialist-mapping.json` has `skillforge.md` mapping

to company "Skillforge". These are two different names for what appears to be the same documentation company. Which is canonical?

7. `~/claude/agents/*.md` priority: Claude Code loads subagents from `~/claude/agents/`. The `definitions/` store is a backup. But 8 mapped agents live only in `definitions/` and are therefore unreachable. Is `~/bin/agent-definitions-sync.sh` being run on any schedule?
-

Architectural Concerns (no auto-fix)

A. Mapping covers only 29 of 66 agents (44%) — the layer is too thin to be a reliable routing table.

The `specialist-mapping.json` is supposed to be John's source of truth for "who builds this?" routing. But the two highest-usage agents in the entire system (`validator` with 44 skill refs, `distiller` with 21 chain refs) are absent. Routing decisions based on this file are structurally incomplete.

B. 7 mapped agents unreachable at runtime.

Agents marked as mapped (`specialist-mapping.json` claims them) but missing from `~/claude/agents/` will fail silently when dispatched. The mapping implies reachability but does not enforce it. No health check validates the mapping → disk correspondence.

C. The chain YAML layer has no executor.

35 chain YAML files define multi-step agent pipelines, but skills invoke agents directly by name — not via the chain files. The `chains/` directory is a documentation artifact, not live infrastructure. All automation currently runs through inline skill → agent calls. This creates a documentation drift risk: chain files will diverge from actual behavior with no mechanism to detect it.

D. 4 virtual companies are phantom — infrastructure without routing.

Axiom, Datavera, Resolver, Lexicon each have: `persona` dir, `README`, `CLAUDE.md`, `company.json`, 5-9 internal agents. None appear in `specialist-mapping.json` or John's routing table. They consume disk and cognitive space but cannot be dispatched through the normal John → `discover.js` → `specialist` route. Direct session invocation (naming the company in a prompt) is the only access path — undocumented and unreliable.

E. Dual-store sync is manual and partial.

16 agents exist only in `~/claude/agents/` (single source of truth but no backup). 8 agents exist only in `definitions/` (backed up but unreachable). The sync script does not auto-run; it must be manually invoked. This creates continuous drift pressure.

F. `planner` is a phantom agent in live chains.

Three chains reference an agent named `planner` that has no `.md` file anywhere on disk. If these chains were ever executed, `planner` steps would fail with no error at the mapping layer.

G. No machine-readable owner for any virtual company.

`company.json` `owner: N/A` across all 12 companies means there is no way to auto-route escalation, billing, or accountability. This is a governance gap, not a code gap.

Revision #2

Created 2026-05-09 19:44:19 UTC by John

Updated 2026-06-14 20:02:54 UTC by John