Skills Usage and Control

Overview

Skills are quickly becoming one of the most important ways to make local AI agents actually useful in practice. At their core, skills are reusable bundles of instructions, workflows, and supporting context that let an agent perform a task more reliably and with less repeated prompting. But early adopters are running into a real tension: the more a skill is individualized to a single developer's setup, preferences, and tooling, the more powerful it becomes for that person, while the less portable it is across a team or company. Tools like Claude Code have made this tension especially visible, because they make it easy to create highly personalized workflows before organizations have agreed on what the shared best practices should be.

Skills We Use Today

Vault & Notes

- obsidian-read-vault — Load vault structure, areas, projects, people
- obsidian-meeting-note — Create meeting notes
- obsidian-project-note — Create project notes (redirects to cop-manager)
- obsidian-project-status — Project status report (redirects to cop-brief)
- people-note-creator — Create person notes for relationship tracking
- capture-to-inbox — Quick capture to inbox
- create-daily-note — Generate daily note and habit tasks

COPs & Reviews

- cop-manager — Create, update, assess, and brief on COPs
- weekly-review — Weekly review with memory compaction and self-improvement
- memory-compact — Compact session scratch into Memory Bank
- inbox-triage — Triage Inbox Tasks using PARA method

Task Management

- helm-task-writer — Create structured tasks for Helm's queue
- helm-task-executor — Execute tasks from Helm Tasks.md
- clickup — ClickUp workspace (tasks, time tracking, docs)

Google Workspace

- google-workspace — Gmail, Calendar, Drive, Docs, Sheets via MCP
- gslides-generator — Google Slides generator

Salesforce & Revenue

- commission-calculator — Calculate 1% commission splits, write to Salesforce
- monthly-report — Monthly KPI report (Salesforce + Harvest data)
- bva-builder — Business Value Assessments for Salesforce engagements

cirriussolutions

Individual Skills vs. Shared Skills

Content & Docs
• sop-creator — Runbooks, playbooks, technical documentation
• helm-reference-guide — Reference guides for Helm context
• excalidraw-diagram — Visual diagrams (workflows, architectures)
• cirrius-ops-newsletter — Weekly newsletter
Communication
• telegram-bridge — Telegram bridge to Helm
• standup-cirrius — Daily Cirrius standup for Slack

Development & Meta
• skill-creator — Create new skills
• command-create — Create new slash commands
• release-notes — Generate release notes for the ops agent repo
• publish-cirrius-ops-agent — Publish the ops agent
• feature-submission — Submit features/bugs to GitHub
• claude-api — Build apps with Claude API / Anthropic SDK

When to Use Individual Skills
• Tied to personal workflow, preferences, or working style
• Dependent on a specific local setup (file structure, tools, environment)
• Reflect individual coding/debugging habits or heuristics
• Used by a single person or very small, specialized sub-team
• Still experimental or evolving rapidly
• Optimized for speed and flexibility over consistency
• Capturing "how I get things done" rather than "how we should do things"

When to Use Team Skills
• Represent repeatable workflows others will benefit from
• Used across multiple people, teams, or projects
• Require consistency to maintain quality (e.g., releases, support, security)
• Encode best practices the organization wants to standardize
• Useful for onboarding or reducing ramp-up time
• Capture institutional knowledge, not just individual habits
• Proven to work and stable enough to scale

A Practical Rule of Thumb
A useful rule is: keep skills individual when they are still exploratory, personal, or highly context-specific; share them when they become repeatable, teachable, and broadly applicable. In other words, personal skills are for discovering what works, while shared skills are for standardizing what has already been proven to work.

BVA Builder Example

Overview

The BVA Builder creates, refines, and maps Business Value Assessments for Salesforce implementation engagements. A BVA is the strategic backbone of a Salesforce project — it proves that every dollar of SOW spend serves a defined business outcome, gives the client executive-level justification, and gives the delivery team measurable success criteria. It works directly in Google Sheets (via MCP) as the source of truth — no JSON Intermediaries.

Example Inputs

To start a BVA build, the skill asks for:

- Spreadsheet ID — existing Google Sheet or creates a new one
- SOW/Scope document — Google Sheet ID, Google Doc ID, or local file path
- Client name and industry vertical — e.g., "Acme Corp / Home Services"
- Transcripts or meeting notes — optional evidence sources (discovery calls, etc.)

Example Output

The BVA Builder outputs a multi-tab Google Sheet containing the full strategic hierarchy (Executive Priorities → Goals → Objectives → Capabilities → Metrics), a SOW mapping tab that traces every capability to its corresponding SOW line item with hour reconciliation, and a Coverage Report tab with quality check findings rated by severity. It also populates Salesforce formula drafts for each success metric and selects Golden Metrics (1 per goal) for an executive dashboard. The final product is a single, client-ready artifact that proves every dollar of implementation spend serves a defined business outcome.