
ONE AGENT PACKAGE · ONE GUIDED WORKFLOW

1102tools Agent Setup

Choose the right agent, install its tested integrations, and know the federal API-key requirements before work begins. The agent checks locally and explains missing or limited access.



Codex

Desktop and CLI



Claude Code

Desktop and CLI

Covers

Three stable agents

Codex Desktop · Codex CLI · Claude Desktop · Claude Code CLI

Updated

August 2026

1102tools.com

CONTENTS

This guide is only for packaged 1102tools agents. You do not need to choose or install their internal components separately.

The two-minute version	3
Data access readiness	3
Choose your agent	5
What every menu choice produces	6
Install in Codex	8
Install in Claude Code (Desktop or CLI)	10
Start the workflow	11
Keys, verification, and troubleshooting	13
Support boundary	15

The two-minute version

1. Choose one agent from the table on the next page.
2. Choose one supported client: Codex Desktop or CLI, or Claude Desktop or Claude Code CLI.
3. Run the two install commands for that client.
4. Start a fresh session and open the client picker: type `@` in Codex or `/` in Claude Code.

No special prompt is required to start

Start typing the agent name and wait for autocomplete. Press Enter once to load the agent or workflow into the prompt box, then press Enter again to start it. The agent checks data readiness, shows its menu, and names the default product before intake. The optional [Agent Playbook](#) shows ambitious requests when you want to see how far a route can go.

You are installing an **agent package**, not assembling a collection of technical parts. Each package contains its workflow, validation instructions, and tested federal source integrations. Some providers require a free key that you configure outside chat.

Three stable agents are available through the 1102tools GitHub repository marketplace. Install only the agent you need. Source, qualification evidence, and testing records live at:

github.com/1102tools-dev/federal-contracting-agents

Before you begin

You need Git, Python 3.10 or later, and `uvx`. On macOS or Linux, confirm with `git --version`, `python3 --version`, and `uvx --version`. If `uvx` is missing, install [uv](#) first.

Data access readiness

No 1102tools account is required; some federal providers require a free account or API key. Every affected agent checks credential presence locally before its menu or routed response. The check never displays, transmits, logs, or validates the value.

AGENT	CREDENTIAL AND NO-KEY ACCESS	WHAT THE AGENT SAYS AT STARTUP
GovCon Growth	<code>SAM_API_KEY</code> Required for SAM.gov opportunity, entity, registration, exclusion, certification, award, hierarchy, and subaward operations.	Reports a missing required credential before the menu, makes no SAM.gov data call, and never labels the missing key as an outage.

AGENT	CREDENTIAL AND NO-KEY ACCESS	WHAT THE AGENT SAYS AT STARTUP
Pre-Award	BLS_API_KEY PERDIEM_API_KEY BLS is limited to 25 requests per day and 10 years per query without its optional key. Per Diem uses a shared DEMO_KEY limited to approximately 10 requests per hour.	Discloses both limited fallbacks before workflow routing and permits bounded work. Scope-only work and GSA CALC+ remain keyless.
Other Transaction	BLS_API_KEY PERDIEM_API_KEY BLS is limited to 25 requests per day and 10 years per query without its optional key. Per Diem uses a shared DEMO_KEY limited to approximately 10 requests per hour.	Discloses both limited fallbacks before workflow routing and permits bounded work. Project-description-only work and GSA CALC+ remain keyless.

No user key: USASpending, GSA CALC+.

Get a key: [SAM.gov Help](#) · [BLS registration](#) · [api.data.gov signup](#). Configure it outside chat, fully restart the client, and rerun the workflow.

Never paste a key into chat

Configure it outside the conversation, fully restart the client, and rerun the workflow. The readiness check sees only whether a variable is present; it never displays, transmits, logs, or validates the value.

Choose your agent

AGENT	USE IT FOR	EXPECTED RESULT
GovCon Growth govcon-growth-agent	Opportunity, competitor, recompile, teaming, agency, market, and pricing intelligence.	Sourced findings or an optional GovCon Growth Brief.
Pre-Award pre-award-agent	SOW/PWS development, FFP/LH/T&M/CR IGCEs, or the complete scope-to-pricing workflow.	A validated requirements document, an auditable estimate, or both.
Other Transaction other-transaction-agent	Milestone-based OT project descriptions, cost analysis, or end-to-end scope-to-cost work.	A validated project description, cost workbook, or both.

If two agents appear relevant

Start with the earlier job in your workflow. A Pre-Award scope can feed its own pricing workflow, and an approved OT project description can feed OT cost analysis. The agents will tell you when a different package is the better next step.

Research approval: GovCon Growth asks before every public-web run. The choices are Native web only (Recommended), Native web with Tavily fallback, Tavily only, or No public web. Tavily is an optional third-party provider and is never selected silently.

What every menu choice produces

You do not have to invent a report. After a productive menu selection, the agent begins with these four labels before intake or artifact preflight:

Recommended outcome: <named chat product, .docx, .xlsx, or combined package>
Includes: <major contents>
Boundary/default: <recommended default and reserved decisions>
Next: <first required fact, document, authority choice, or approval>

A legitimate gate is specific

Good: "Next: provide the requirement or existing acquisition documents," "confirm FFP, LH/T&M, CR, or hybrid pricing," or "approve the proposed research plan." Bad: "What kind of report do you want?" after you already selected a report route.

GovCon Growth Agent

CHOICE	DEFAULT PRODUCT AND CONTENTS	FIRST GATE AND BOUNDARY
1	Federal Opportunity Shortlist with fit, timing, competition clues, and evidence gaps.	Company/offering and search scope, then research-plan approval.
2	Opportunity Evidence Screen separating public evidence from internal facts.	Notice and internal context; no public-data-only bid verdict.
3	Competitor/Incumbent Intelligence Profile.	Company/entity match and approved plan; label inference.
4	Recompete Pipeline with end dates, follow-ons, incumbents, vehicles, and confidence.	Market/company and horizon; do not treat stale awards as open targets.
5	Partner Shortlist , or a Due-Diligence Profile when a company is named.	Pursuit/candidate criteria; no responsibility or teaming determination.
6	Agency/Market Intelligence Snapshot.	Agency/market and intended positioning use.
7	Labor-Rate/Pricing Context Table.	Roles, locations, and comparison frame; no reasonableness verdict.
8	Refreshed prior product plus change log.	Prior product and as-of date.
9	One diagnosed route recommendation , with at most one alternative.	Objective, available material, and intended use.

Pre-Award Agent

CHOICE	DEFAULT PRODUCT AND CONTENTS	FIRST GATE AND BOUNDARY
1	Validated SOW/PWS .docx plus staffing and Section B chat handoffs.	Requirement/documents; no pricing content in the work statement.

CHOICE	DEFAULT PRODUCT AND CONTENTS	FIRST GATE AND BOUNDARY
2	Routed IGCE .xlsx , separated by confirmed method or hybrid CLIN.	Confirm FFP, LH/T&M, CR, or hybrid; no fair-and-reasonable determination.
3	SOW/PWS, approved handoffs, and routed IGCE workbook(s).	Approve scope and handoffs before pricing.
4	Affected artifact rebuild plus before/after change register.	Approved prior package and change request.

Other Transaction Agent

CHOICE	DEFAULT PRODUCT AND CONTENTS	FIRST GATE AND BOUNDARY
1	Validated OT Project Description .docx plus milestone chat handoff.	Project facts and authority choices; no cost content or AO determination.
2	Milestone-based OT Cost Analysis .xlsx .	Approved milestones; no participant, contribution, or price determination.
3	Both artifacts with the approved milestone handoff carried forward.	Approve project description and handoff before costing.
4	Affected artifact rebuild plus before/after milestone register.	Approved prior package and change request.

Install in Codex

This path supports Codex Desktop and Codex CLI. Open Terminal and add the 1102tools repository marketplace once:

```
codex plugin marketplace add 1102tools-dev/federal-contracting-agents --ref main
```

Then install the agent you selected. Use `govcon-growth-agent`, `pre-award-agent`, or `other-transaction-agent`:

```
codex plugin add AGENT@1102tools
```

Example:

```
codex plugin add govcon-growth-agent@1102tools
```

Keep shared data connections complete

Codex users who install several 1102tools agents must also install the maintained host profile. It owns the complete five-server federal MCP inventory and allowlists credential variable names without storing their values:

```
curl -fsSL https://raw.githubusercontent.com/1102tools-dev/federal-contracting-agents/main/config/codex/1102tools-host.config.toml -o ~/.codex/1102tools-host.config.toml  
codex --profile 1102tools-host
```

Codex Desktop: profile files are selected by the CLI. For Desktop, copy the complete `[mcp_servers...]` tables from that downloaded file into your user-level `~/.codex/config.toml` without replacing unrelated settings, then restart the app. The source file contains package pins and credential variable names, never credential values.

Finish the installation

1. Close the current Codex task.
2. Start a new task, type `@`, begin typing the agent name, and wait for autocomplete.
3. Press Enter once to load the agent into the prompt box. Press Enter again to start immediately, or type a normal request before the second Enter.

Update a Codex agent

Codex refreshes the marketplace separately from the installed package. Upgrade the marketplace, remove the installed agent, then add it again:

```
codex plugin marketplace upgrade 1102tools  
codex plugin remove AGENT@1102tools  
codex plugin add AGENT@1102tools
```

Start a new task after reinstalling. Codex does not currently expose a separate plugin-update command.

Install in Claude Code

Run these two commands in a terminal. They install the package for Claude Code. After installation, open a fresh Claude Code session in Claude Desktop or the CLI and verify that the agent appears.

```
claude plugin marketplace add 1102tools-dev/federal-contracting-agents
```

Then install the agent you selected. Use `govcon-growth-agent`, `pre-award-agent`, or `other-transaction-agent`:

```
claude plugin install AGENT@1102tools
```

Example:

```
claude plugin install govcon-growth-agent@1102tools
```

Finish the installation

1. Start a fresh Claude Code session in Claude Desktop or the CLI.
2. Type `/`, begin typing the agent name, and wait for its workflow to appear.
3. Press Enter once to load the workflow into the prompt box. Press Enter again to start immediately, or type a normal request before the second Enter.

Update a Claude agent

For a manual update, refresh the marketplace, update the selected package, and then start a new session:

```
claude plugin marketplace update 1102tools  
claude plugin update AGENT@1102tools
```

Inside an active Claude Code session, `/reload-plugins` applies plugin changes without a full restart.

Optional automatic updates: third-party marketplaces do not auto-update by default. Open `/plugin`, choose *Marketplaces*, select `1102tools`, and enable auto-update. At startup, Claude Code can then refresh the marketplace and its installed agents; if one changes, reload the plugins when prompted.

Start the workflow

Load it, then launch it

Codex Desktop and CLI: type @ and begin typing the agent name. Claude Desktop and Claude Code CLI: type / and begin typing the agent name. In either client, wait for autocomplete, press Enter once to load the selection into the prompt box, then press Enter again to send and start it.

You can launch the agent without writing a prompt. If you already know the job, type a plain request after the first Enter and send it with the second:

- "Find and evaluate federal opportunities for a small cybersecurity firm."
- "Help me create a help-desk PWS and associated IGCE."
- "Help me develop and cost a prototype Other Transaction."

Ordinary request or ambitious request

An ordinary request is enough to enter the right workflow. An ambitious request helps a less-confident user see the full shape of the product without changing the agent's gates.

AGENT	ORDINARY START	AMBITIOUS REQUEST AFTER ROUTE SELECTION
GovCon Growth	"Help me find opportunities."	"Build a prioritized shortlist with fit signals, timing, competition clues, evidence gaps, and the internal facts needed before pursuit."
Pre-Award	"Help me create a PWS and IGCE."	"Run the full scope-to-pricing workflow, validate the work statement first, carry approved handoffs forward, and build the correctly routed workbook."
Other Transaction	"Help me develop and cost a prototype OT."	"Build the milestone project description first, obtain approval of its handoff, then produce the auditable milestone cost analysis."

For route-by-route power requests, use the optional [Recommended Agent Playbook](#).

Inside the selected agent, the local readiness check runs first. Any missing-key or limited-fallback warning appears before the complete menu or a directly routed response. The agent then asks for missing documents, facts, dates, source approvals, and user decisions. Do not depend on ambient routing to choose the package for you.

Optional direct workflow names

The client picker is the easiest start. Power users can invoke a packaged workflow directly with these names:

AGENT	CODEX	CLAUDE CODE
GovCon Growth	<code>\$govcon-growth-workflow</code>	<code>/govcon-growth-agent:govcon-growth-workflow</code>
Pre-Award	<code>\$pre-award-workflow</code>	<code>/pre-award-agent:pre-award-workflow</code>
Other Transaction	<code>\$other-transaction-workflow</code>	<code>/other-transaction-agent:other-transaction-workflow</code>

Keep approvals in the conversation

The agent may stop for scope approval, source-plan approval, pricing-method confirmation, public-web approval, or artifact approval. Those bounded stops are part of the workflow, not errors. An open-ended request for you to invent the product is not.

Keys, verification, and troubleshooting

Add credentials only when the job needs them

The complete agent-by-agent matrix appears near the beginning of this guide. Use the current registration links there. A missing required SAM key blocks only SAM-dependent work; BLS and Per Diem disclose their bounded fallbacks.

Do not paste credentials into an ordinary chat message. Add them to your client's credential configuration instead.

Claude Desktop and Claude Code CLI. Add an `env` block to `~/.claude/settings.json`. This is the method that works on both surfaces, because they share one settings file. Restart the client afterward.

```
{
  "env": {
    "SAM_API_KEY": "your-key",
    "BLS_API_KEY": "your-key",
    "PERDIEM_API_KEY": "your-key"
  }
}
```

If you use Claude Desktop, exporting keys in a terminal will not work

A desktop application launched from the Dock or Start menu does not read your shell profile, so an `export` line never reaches it. Use the settings file above.

Codex CLI. Export the key in the shell that starts the client, then launch with `codex --profile 1102tools-host`.

```
export SAM_API_KEY="your-key"
export BLS_API_KEY="your-key"
export PERDIEM_API_KEY="your-key"
```

Codex Desktop. The same variable names must exist in the environment that launches the desktop app. A shell export reaches only a client started from that shell. Restart the app after changing its launch environment; do not paste keys into chat.

Quick checks

- **No menu:** start a new session and use the explicit workflow name.
- **Command not found:** confirm the client, Git, and `uvx` are on `PATH`.

- **Missing-key warning:** this is expected before the workflow choices. Configure the named credential outside chat, fully restart the client, and rerun the workflow. SAM-dependent work makes no upstream call while `SAM_API_KEY` is absent.
- **Readiness operation missing:** the MCP package or complete shared host profile is outdated or incomplete. Update all three agents and the `1102tools-host` profile, restart, and try again.
- **Spreadsheet cannot be built in Codex CLI:** accept JSON, Markdown, CSV, or a Codex Desktop handoff. The agent must stop early rather than fabricate an XLSX.
- **Another artifact cannot be built:** let the agent offer a reduced-scope chat result; do not accept a fabricated file.

Support boundary

1102tools maintains two client families across four tested surfaces:

- **Codex Desktop and Codex CLI**
- **Claude Desktop and Claude Code CLI**

Artifact capability by tested surface

SURFACE	STABLE CAPABILITY
Codex Desktop	Guided chat, federal research, DOCX, and XLSX workflows when the required artifact tools are available.
Codex CLI	Chat, research, routing, and DOCX. Without a spreadsheet runtime, the agent must stop before XLSX and offer JSON, Markdown, CSV, or a Desktop handoff.
Claude Desktop	Full artifacts when the required document and spreadsheet tools are available.
Claude Code CLI	Full artifacts when the required document and spreadsheet tools are available.

Other clients may support compatible agent-package conventions. They are not currently part of the 1102tools test matrix or installation support commitment. If you use another client, installation and troubleshooting are self-supported.

Stable support boundary

These packages assist with documented research, workflow control, and validated artifacts. They do not replace a Contracting Officer, Agreements Officer, legal counsel, or another authorized decision-maker. Keyed sources depend on credentials and upstream availability. Tavily requires explicit approval and is never a silent fallback.

The supported distribution source is the public 1102tools GitHub repository marketplace. This guide does not claim that the agents are listed in an official OpenAI, Codex, Anthropic, or Claude storefront.

Stable release:

github.com/1102tools-dev/federal-contracting-agents/releases/latest

Agent source and testing records:

github.com/1102tools-dev/federal-contracting-agents