CF

CUTTLEFISH
ENTERPRISE RUNTIME

MICROSOFT MARKETPLACE OVERVIEW

Governed execution for enterprise AI.

Connect AI to real systems without handing models unrestricted authority. Cuttlefish places policy, approvals, evidence, durable state, and model-independent continuity around consequential work.

Explicit authority
Only declared, resolved capabilities are exposed.

Approval-gated action
Material operations pause for the correct human authority.

Evidence-backed outcomes
Execution is paired with verification and reviewable records.

Model freedom
Cloud, local, private, and sovereign model routes remain replaceable.

EXECUTIVE OVERVIEW

01

AI can reach tools. Enterprise work requires custody.

Cuttlefish is the governed environment runtime between AI cognition and operational systems. Models can reason, draft, and propose. Cuttlefish determines what is visible, what may be prepared, what can execute, what requires approval, and what becomes durable organizational state.

One control plane

Policy, identity, approvals, evidence, runtime state, and model routing.

Many model routes

Use approved cloud, local, private, or sovereign inference without transferring authority.

Verifiable work

Preserve the request, decision, execution, evidence, verification, and outcome.

What changes for the enterprise

01 / CONTROLLED REACH

Expose capability, not raw power.

Register approved systems, browser targets, files, APIs, and applications as bounded capabilities rather than unrestricted model tools.

02 / ACCOUNTABLE ACTION

Separate preparation from execution.

Inspection, planning, drafting, and staging remain reversible. Material actions cross an explicit policy and approval boundary.

03 / DURABLE CONTINUITY

Keep the institution outside the model.

Operational memory, artifacts, evidence, graph context, run history, and reusable work surfaces persist across model changes.

Designed for enterprises, MSPs, platform teams, DevOps and GitOps organizations, security-conscious operators, and technical service firms that need automation velocity without surrendering control.

OPERATING MODEL

02

A focused workspace above a governed runtime.

Users work through familiar surfaces. The runtime appears only when needed - as approval requests, evidence drawers, governed apps, companion windows, receipts, and administrative controls.

H

Home

Conversation, current work, approvals, results, handoffs, recent evidence, and outcome-oriented activity.

W

Workspace

Create, inspect, validate, version, and promote documents, workflows, generated tools, and installable workspace applications.

B

Browser

Operate registered web targets, verify outcomes, capture page evidence, and preserve run state through operator handoffs.

A

Apps and companions

Launch connected applications, role-specific operator surfaces, companion windows, permissions, and application health.

S

Settings and administration

Manage model routes, connections, privacy, identity, policy, budgets, memory controls, diagnostics, and tenant governance.

THE OPERATIONAL BOUNDARY

The model is a cognition engine - not the authority owner.

Cuttlefish owns the work boundary: exposure, identity, policy, approval, execution, evidence, memory, and continuity. Models can be routed or replaced without losing the organization's control state.

1

Environment connector runtime

Compiles declared APIs, browser targets, context routes, evidence routes, and connector actions into normalized capabilities.

2

Policy and authority engine

Classifies actions by effect, risk, source, identity, authority, and execution boundary before they can run.

3

Evidence and receipt system

Preserves what was approved, denied, executed, verified, and reconciled as reviewable operational records.

4

Durable intelligence layer

Maintains memory proposals, graph context, world state, artifacts, and workflow continuity independently of the active model.

GOVERNED EXECUTION LIFECYCLE

03

From environment discovery to proven outcome.

Connecting a system is not permission to act. Compiling a capability is not execution. Cuttlefish separates discovery, preparation, authority, execution, and proof.

01 / CONNECT

Register the environment.

Declare the system, browser target, source, application, or operator request that defines the work boundary.

02 / RESOLVE

Resolve identity and context.

Determine tenant, user, role, target state, model route, and the governing authority boundary.

03 / COMPILE

Compile capabilities.

Normalize approved operations with source, effect, risk, approval, credential, and evidence metadata.

04 / PREPARE

Stage reversible work.

Inspect, summarize, plan, draft, build, and preview without crossing the execution boundary.

05 / AUTHORIZE

Approve and execute.

Route consequential operations through policy, explicit approval, credential isolation, and runtime controls.

06 / PROVE

Verify and reconcile.

Capture evidence, receipts, run history, durable state, reusable surfaces, and the verified outcome.

Proof is part of completion.

A task is not complete merely because a tool returned success. The runtime records what was requested, what authority was used, what changed, and whether the intended outcome was actually established.

READ-ONLY

Inspect under established permission.

Query approved state, review a file, summarize a page, or search retained knowledge.

PREPARE-ONLY

Create staged or reversible work.

Draft a response, build a plan, generate a preview, or prepare a change package.

APPROVAL-REQUIRED

Pause for explicit authority.

Send, publish, spend, write externally, or perform privileged and material changes.

PLATFORM CAPABILITIES

04

One runtime for connection, execution, verification, and reusable work.

The platform combines governed environment access with a workspace that can build, install, and operate role-specific applications bound to real runtime objects.

Environment manifests

Describe systems, contracts, browser targets, allowed origins, health checks, context resolvers, evidence routes, and execution limits in a governed contract.

Capability compilation

Normalize approved operations into semantic capabilities with source, risk, side-effect, approval, credential, and evidence metadata.

Workspace App Foundry

Convert repeated work into governed applications and operator surfaces bound to actual environment state rather than placeholder data.

Governed browser runtime

Operate registered web targets, inspect page structure, verify outcomes, capture evidence, and preserve state across human handoffs.

Ambient and voice invocation

Start work from selected text, files, folders, browser pages, screen regions, hotkeys, or voice while retaining the same policy boundary.

Model-independent routing

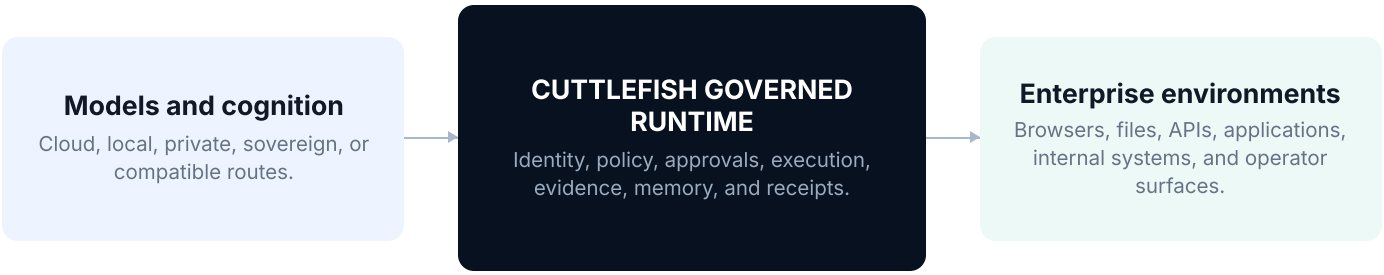
Use approved cloud, local, private-cloud, sovereign, or compatible endpoints without transferring policy ownership to the model.

Enterprise Surface Packs

Project the runtime into role-native experiences for IT, support, platform, compliance, operations, or executive review.

Continuity and operational learning

Preserve outcomes, evidence, memory candidates, graph relationships, world state, and reusable workflows across model replacement.



Generated UI is never sovereign.

Workspace applications declare what they need and request host-mediated actions. They do not receive unrestricted shell, file, browser, desktop, memory, graph, or governance authority.

Open model choice under one consistent control plane.

Deployment adapts to where data, inference, execution, memory, evidence, and audit records are permitted to live.

Deployment patterns

Local-first

Keep approved work and inference on the user's machine where required, under the same governance model.

Cloud-connected

Use approved external providers and shared services selectively with explicit route and task-level control.

Private and hybrid

Combine local, tenant-hosted, private-cloud, and external providers by data class, risk, cost, or residency policy.

Self-hosted and sovereign

Keep prompts, retrieval, inference, execution, memory, evidence, and audit within the organization's boundary.

Credentials remain isolated.

Secrets stay outside model prompts, generated applications, and ordinary logs. Revocation can withdraw connections, grants, routes, and capabilities.

Enterprise administration

Identity	Domain control, single sign-on, provisioning, role-based access, and dedicated organization administration.
Policy	Tenant-wide capability rules, approval boundaries, blocked operations, private routes, redaction, and network constraints.
Operations	Environment health, run visibility, evidence review, emergency stop, and revocation controls.
Usage and spend	Provider usage visibility, team budgets, metering, and allocation across approved model routes.
Assurance	Read-oriented audit views, evidence-backed outcome review, receipt verification, export, and retention controls.
Rollout	Pilot scoping, integration planning, security review, deployment design, enablement, and operational support.

Deployment scope, available connectors, administrative features, retention, assurance requirements, and commercial terms are confirmed during solution design and reflected in the applicable agreement.

OPERATIONAL FIT

06

Built for teams responsible for real environments.

The strongest first deployments are organizations that already manage systems, permissions, approvals, change control, customer boundaries, and evidence.

MSP and managed IT

Operate across customer environments through explicit client context, manifest-defined boundaries, approval paths, evidence packs, and reusable operator workflows.

Platform engineering

Expose internal platforms through governed capability maps instead of raw model tools while retaining policy, verification, and administrative control.

DevOps and GitOps

Prepare and execute operational change with registered targets, staged work, approval gates, outcome evidence, and rollback awareness.

Enterprise AI operations

Standardize multi-model work across teams under tenant identity, role-native surfaces, budgets, approval rules, and reviewable operating history.

Technical service firms

Package repeatable delivery work into governed applications that preserve client scope, evidence, review, and handoff.

Security and review teams

Inspect what ran, what was approved, what changed, and what evidence supports the result without reconstructing raw logs.

START WITH ONE REAL ENVIRONMENT

Prove the control model on a material workflow.

A Cuttlefish engagement defines the environment boundary, compiles approved capabilities, establishes policy and approvals, runs a governed workflow, verifies the outcome, and produces a reusable operator surface.

getcuttlefish.app/enterprise
support@getcuttlefish.app

This Microsoft Marketplace listing supports a separately scoped enterprise engagement. It does not automatically deploy software or create an unrestricted subscription. Access is provisioned after scope, security requirements, and commercial terms are confirmed.