

Orbi360

Product sheet for a person deciding whether this belongs in the organisation, and for the people who will live with it. Plain language. The technical contract is in the companion sheet.

Orbi360 0.4.0, read from the source on 1 October 2026. Orbi is the coworker app on the OpenBot platform, tenant name Orbi, package 0.0.15.

Your tools.
Your memory.
Your boundaries.

One place to decide what an assistant can know, remember, and ask for.

The short version

Most assistants want a key, and then they can do everything that key can do. That is simple to set up. It is also a wide door.

Orbi360 is the door in front of the toolbox.

You connect a service once. You look at the individual actions it offers, and you publish the ones you have reviewed. You decide, person by person, the most they are allowed to touch. When Orbi, Claude, ChatGPT, or Notion asks for something, Orbi360+ looks at that request and releases only the piece that fits inside the permission already given. Ordinary software checks the permission again before anything leaves. A receipt is written. The receipt says what was released. It does not keep a copy of the file.

The model helps choose. The software decides. A skill, a prompt, or a confident answer cannot widen a permission.

Three names

Name	In ordinary words	What you do there
Orbi360	The administration portal.	Connect services. Set what each person may use. Issue a connection for a tool you already have, or add an agent to that person's Orbi area. Read receipts. Pause the instance when you need data operations to stop.
Orbi360+	The model inside the gateway.	On each request, chooses the smallest set of records that will answer it, inside the permission. It cannot raise the permission, approve a memory, or see the owner credential.
Orbi	The app people talk to.	An agent you add appears in that person's own area. The person talks to it there. Its company tools still come through Orbi360.

Orbi used to be built and described as OpenBot. In this checkout the product name on the tenant package is Orbi. The app is still the coworker platform: a person signs in, opens a channel, and talks to an agent that has a standing role.

Orbi is one client of the gateway. A team can use Orbi360 with the chat tools they already pay for, and add an Orbi agent when a standing role is worth having.

One afternoon, as an example

A company wants people to ask questions of a Notion workspace, from the assistants they already use, with a limit on each person.

1. An administrator connects Notion in Orbi360. The Notion credential stays in the vault on that instance. It is not pasted into Claude, and it is not pasted into Orbi.
2. Notion offers many actions. The administrator publishes the ones they have read. Each published action is marked read-only, given a label, and limited to the fields that may come back. An action left unpublished does not exist for any caller.
3. The administrator creates a person and sets a ceiling. For this example the ceiling allows that one published search, titles of records labelled personal, and no memory proposals.
4. The person needs Claude. The administrator writes a permission inside that ceiling. Orbi360 refuses anything larger. It then issues a client credential and shows it once. That credential goes into Claude, pointed at the company's Orbi360 address. Orbi360+ does not write the permission. It selects inside a permission that already exists.
5. The same person needs a standing agent. The administrator pushes one. It appears in the person's existing Orbi area, with its own credential. Claude's credential and the agent's credential are different. Turning one off leaves the other alone.
6. Claude asks a question. Orbi360 builds the small set the permission allows. Orbi360+ picks the relevant rows from that set, or holds the call if it is unsure. The software checks the permission again, writes a receipt, and returns the rows. Claude never holds the Notion key.

A second person gets a second ceiling. They do not inherit the first.

What the administrator can do

The console is the owner's. People do not sign into Orbi360. They sign into Orbi, or into the chat tool they were given a credential for.

Connect a known service. The catalogue is a list shipped with the instance: Notion, GitHub, Linear, Stripe, HubSpot, Figma, Jira and Confluence, and others of the same kind. There are 43 connectable recipes in the current list. Ten further entries are references you can read, including filesystem, Git, and web fetch. Those references do not connect.

Publish one action at a time. A later change at the service invalidates that review. A renamed or widened action does not keep the old permission.

Set a ceiling. The most one person may have: which actions, which local records, titles or bodies or both, which labels, which kinds of memory, whether the caller may suggest a memory, and how much one release may contain. The person, the agent, and the external tool can be made smaller. They cannot be made larger than the ceiling.

Issue two kinds of caller. An external tool, or an Orbi agent. Each has its own permission.

Read receipts. You can answer "what did this tool receive?" from the receipt: who asked, which action, whether it was allowed or refused, which records, which fields, and how large the answer was. You do not open the file to answer that.

Pause. One switch stops the next data operation. You can look, fix a permission, and resume. Work already sent to a service cannot be pulled back.

Remove a person. Their credentials stop. If Orbi is connected, the removal is agreed by both sides. If one side cannot complete it, the other side leaves the person in place. An administrator address that the Orbi deployment has pinned cannot be removed this way.

Review memory. Suggestions wait. Approving one makes it available to permissions that include it. Rejecting one leaves it out.

What a person experiences

In Orbi, the person signs in as themselves and talks in a channel. An agent the administrator pushed is in that person's area. It has a name, a title, and a role. The role is the standing instruction. The role does not grant a tool.

When the agent needs a company record or a published action, it asks Orbi360 with its own credential. If the permission says no, the agent is told to say so. The governed setup tells it not to look for another door to the same system.

The person can also keep using Claude, ChatGPT, or Notion. Those products stay themselves. They reach the company only as far as their own credential allows.

Skills are instructions for a repeated job. A saved skill does nothing until it is attached, and attaching it does not open a service. In the Orbi package, the skills for writing a skill and for drafting a coworker name no vendor tool. The draft says, in the product's own words, that connectors are granted afterwards, through Orbi360.

Labels, fields, and memory

Three labels. **Public** is material you are willing for a model to handle. **Personal** is a named person's material, still inside the ceiling. **Restricted** stays out unless that caller's permission and the model's own ceiling both include it. A newly published action starts as restricted, so it stays dark until someone chooses otherwise.

Two fields. A caller can be allowed to see that a record exists, by its title, and still be refused the body.

Three kinds of memory. A preference, a constraint, or an episode. The agent may suggest one when the permission allows suggestions. The suggestion stays a candidate until a person with the owner credential approves it. The agent cannot approve its own memory.

Learning a way of working

This function is specified. It is not a switch in the 0.4.0 console. When it is built, it is part of the same gateway. There is no new service to connect, and no new tool appears in Claude, ChatGPT, Notion, or Orbi.

An administrator turns Learning on for the workspace, then allows it per person. Both start off. A person without that allowance is unchanged.

The first time an agent does a job of at least three governed steps, Orbi360 can draft a way of working: the steps that succeeded, what a good result looks like, and what must stay out. The draft waits with the other suggestions. The agent cannot approve it. A “yes” in the chat is not approval. The administrator approves it, or rejects it, in the same place they review memory.

The next time that same job comes back, an Orbi360+ run may be shown the approved steps before it starts. The steps count toward the same size limit as any other context. They do not add a permission. A coworker in Orbi is not given a new tool. If that coworker was already allowed to read memory, it can read the approved steps there. Its private reasoning inside Orbi is not something this function can see. It learns from the calls that came through Orbi360.

Each repeated job leaves a line on the activity record. A later refusal after those steps were used counts against that way of working. The agent cannot mark its own work as successful. Turning Learning off stops new drafts and stops the steps being offered. Anything already approved stays in the workspace and is simply not used.

A concrete case. Someone asks their Orbi coworker, three times in a week, for a short brief from the same published search. After the administrator approves the draft, the next Orbi360+ run of that job starts from those steps. The search is still the search they published. The ceiling is still that person’s ceiling. The receipt still says what was released.

On the console this is “Learning is on” or “Learning is off”, a checkbox “May learn a way of working”, and a Memory section “Ways of working”. It does not have its own page.

Why the arrangement is different

The caller receives a permission, not the workshop. Connecting a service makes it available to Orbi360.

Publication is per action. A person's ceiling is smaller than the catalogue. An agent or an external tool is smaller again. Orbi360+ is smaller again, for that one call.

The same door serves the tools people already use. Orbi is convenient when you want an agent in a person's area. Claude, ChatGPT, and Notion can be clients of the same permissions, the same receipts, and the same pause switch.

Software checks the permission. Orbi360+ can suggest and select. The release happens after the live permission, the pause state, and the current version of the record have been checked again. A malformed or incomplete model decision holds the operation. There is no automatic retry that might send it twice.

Keys stay on the instance. Connector secrets are encrypted with a vault key that lives with the database. The model provider key stays in the backend and is sent only to the model endpoint you configured. The console can show which model recipient a conversation is using. It does not show the key.

One connection, many callers. You sign a service in once. Each person, each agent, and each external tool then gets a permission of its own.

What you can rely on

- The instance is yours. One deployment holds one organisation's catalogue, vault, and receipts.
 - The database and the vault key are a pair. Back them up together. Losing the key means the connector credentials cannot be read. The receipts can.
 - The model is whoever you configured. A hosted endpoint, or a server you run, as long as it speaks the Chat Completions shape Orbi360 uses. If the first provider fails, Orbi360 does not quietly try a second one.
 - Chat with the in-console helper is short-lived. It keeps a few recent exchanges, expires after a quiet half hour, and is discarded when the process stops. Receipts remain.
 - In the current gateway, a published action is a read. The administrator confirms that when they publish it. This release does not publish write actions.
-

What this product keeps outside its promise

- Orbi360 governs what Claude, ChatGPT, Notion, and Orbi can reach. It does not replace those products.
 - Publishing a service does not publish every action it has.
 - A skill does not grant a tool.
 - An agent does not approve memory, approve a way of working, change permissions, or administer the portal.
 - Learning, once it is switched on, remembers a way of working inside a permission that already exists. It does not discover a service or raise a ceiling.
 - People do not sign into the Orbi360 console. Administration is the owner credential. The person's sign-in is Orbi's, or the external product's.
 - The gateway does not enrol a new Orbi user. The person already belongs to the Orbi deployment before an agent is pushed.
 - Voice, a browser on a private computer, and scheduled routines exist in the wider coworker platform. The Orbi package shipped here does not turn them on, and Orbi360 does not grant them. An administrator who enables them on the Orbi deployment is opening a door this portal does not supervise.
 - A call already sent to a provider stays sent. Pause and revoke apply to the next check.
-

Who it is for

A person who uses more than one assistant and wants each of them to hold a permission instead of a raw key.

A team that will let people have agents, and wants those agents added to the area the person already uses, under a ceiling an administrator set.

An operator who wants one instance, one vault, one pause switch, and a receipt for each release, with the model provider of their choosing.

It is a poor fit for a shared host that keeps several companies in one room, for a model that is allowed to ignore a permission, or for a chat product with no record of what it was shown.

In this release

You can run the gateway and the console today. You can connect a catalogue recipe, publish reviewed read-only actions, keep local records, store and review memory, issue client credentials, set a person's ceiling, pause the instance, and read receipts. With Plus enabled, the console assistant and the governed reads use one Chat Completions model. External MCP clients call the gateway with a client credential. The owner credential is for the console.

You can push an agent into an Orbi deployment that already has that person, and you can remove the person from both sides together, when the two portal credentials are configured. The Orbi app they land in is the coworker platform, named Orbi in this tenant.

Learning a way of working, described above, is the specified next function of this gateway. The 0.4.0 console does not yet show the switch.

The locked line under the mark, for a cover and only for a cover:

INTELLIGENCE, GOVERNED.