

THE UNIFIED SPACE AND SCIENCE DATA PLATFORM

CosmicIntersection

One governed home for the whole research data lifecycle, from a raw observation to a citable result.

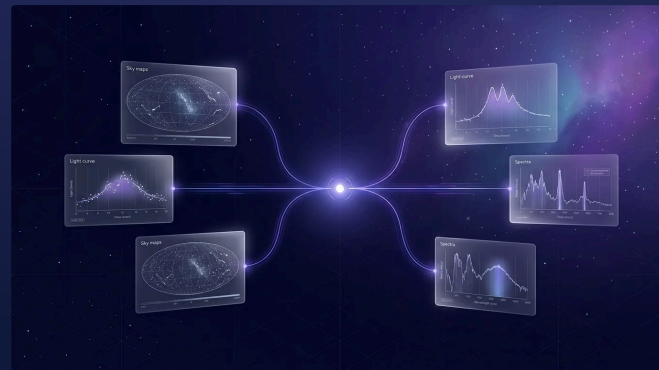
Data catalogue

Observation pipelines

Reproducible notebooks

Citable releases

Governed sharing

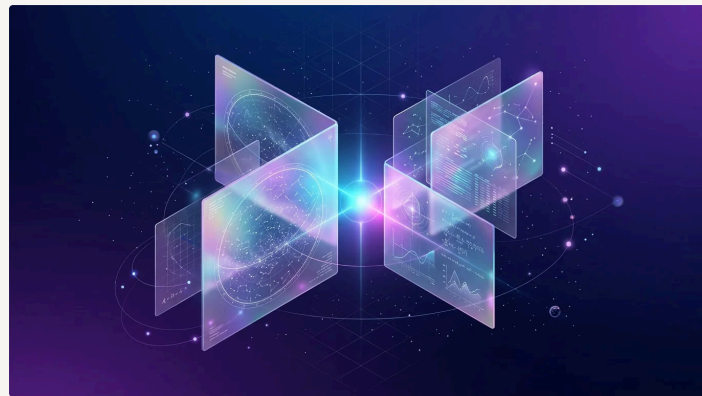


01 The short version

CosmicIntersection is one place for a research team to keep its data, process it, analyse it, share it and publish it, with the trail from the raw observation to the published figure kept intact the whole way.

THE PROMISE

Every result should be traceable back to the observation it came from, without anyone having to remember how.



1

One connected model

A source becomes a dataset, becomes a pipeline product, becomes an analysis, becomes a citable release. No copying between systems.

9

Things you govern

Sources, datasets, collections, pipelines, runs, notebooks, analyses, releases and access grants, each with its own permission level per role.

Day 1

Governance included

Roles, audit trail, quotas and tenant isolation come with the platform rather than arriving later as an enterprise upgrade.

\$0

Complete free tier

The whole product surface at starter quotas, no card required, sized to prove the workflow rather than to frustrate you.

01 Why it exists

TODAY

Data-heavy science runs on a sprawl of disconnected systems. Raw telescope frames sit on one file server, instrument readings in another archive, spacecraft telemetry somewhere else again. The calibration script that turned those into something usable lives on a laptop. The analysis is a notebook nobody else can run, and the finding ends up locked inside a PDF. Nothing lines up, results cannot be reproduced, and the people who collected the data struggle to find it again.

IN THE MIDDLE OF IT

CosmicIntersection sits in the middle of all that. Every feed you pull from becomes a source. Everything that lands becomes a dataset in one searchable catalogue, described down to the unit and origin of each column. Processing becomes a pipeline that is versioned and repeatable rather than a folder of scripts. Analysis happens in notebooks that record the exact software they ran against. Finished work becomes a release other people can cite, and restricted data is shared through grants that expire on their own.

WHY IT HOLDS

Because it is one connected model rather than nine tools bolted together, nothing has to be re-keyed and nothing quietly loses its history. A reviewer can walk from a published figure back to the observation it came from. A new student can see what the group holds on their first day. An institution can run many labs under one roof with the same rules applied to each.

02 Who it is for

Research scientists

You want to find the right data and get to an answer, not spend the week learning a fifth archive interface. The catalogue, the notebooks and the analysis record are built around that.

Observatories, surveys and national facilities

You run many instruments across many epochs. Bringing survey catalogues, light curves and spectra into one queryable archive is the point of the product, and many teams can sit under one tenant.

Research software and data engineers

You have written the same ingestion script a dozen times for a dozen archives. One typed source model and one API replace the bespoke scrapers, and runs record their own lineage so debugging starts from evidence.

Universities, labs and graduate researchers

Every group gets a governed home for data, processing and analysis, and a student can start doing real science on a free tier instead of spending three months on data plumbing.

Principal investigators and research leads

You answer to funders for reproducibility and to collaborators for access. You get a shared workspace, governed sharing with expiry, and citable output that strengthens the next grant application.

Data-intensive science beyond astronomy

Physics and instrumentation, Earth and remote sensing, and signal and time-series work all depend on the same ingest, process, analyse, publish loop, and the data model does not assume a telescope.

Data stewards and data centre directors

You are accountable for how the holding is described, versioned and released. Catalogue curation and access governance are a distinct job here, with their own role that never touches anyone's research.

None of the problems below are exotic. They are the ordinary daily friction of doing science with data, and they compound: each one makes the next harder to fix.

The data is scattered and nobody can find it again

Archives, file shares and instrument drops each have their own interface, query language and naming habits. Data is rarely deleted; it decays quietly into a folder with no metadata, no schema and no provenance.

- One catalogue across every source you connect, with instrument, target, coverage, format, access level and per-column provenance recorded as first-class metadata rather than as a README.

Results cannot be reproduced six months later

A reviewer asks you to re-run a figure from an older paper. The processing script has moved on, the software environment was never written down, and the parameters are a memory. The honest answer is that you cannot.

- Notebooks pin their runtime and package versions and bind to the datasets they used. Runs record their steps, parameters and what each step read and produced, so a result carries its own recipe.

Calibration lives in a folder of scripts

The moment reduction logic exists only on one person's machine, the group has lost both reproducibility and continuity. When that person moves on, the pipeline becomes an archaeology project.

- Processing becomes a versioned pipeline definition with typed steps and a trigger, so what produced a table is a record in the platform rather than a file somebody remembers to back up.

Sharing restricted data means copying a credential

Collaboration falls back to email attachments and shared drives. There is no version control, no expiry, no audit trail, and no way to withdraw access once it has been handed out.

- Scoped access grants issued to a named collaborator, with an embargo or expiry attached and revocation available at any moment. Source credentials are referenced, never copied onto a record.

Publishing is a project rather than a button

Data that cannot be cited rarely gets reused, so months of careful work stops being useful the day the paper is out. Preparing a proper deposit is its own separate chore nobody has time for.

- A working notebook and the datasets beneath it become a versioned release with an abstract, authors, a licence and a persistent identifier, moving through draft, review and published.

Governance is bolted on after something goes wrong

Access control, audit and multi-tenancy usually arrive as an enterprise add-on once a leak or a funder question forces the issue, and by then the habits are already set.

- Every object belongs to a workspace and every operation is permission-checked at the edge. The audit trail, the quotas and the tenant boundary exist from the first dataset onwards.

Onboarding a new researcher takes months

Every new student or postdoc spends the first weeks learning which portal holds what, whose script does which correction, and why last month's numbers do not match this month's.

- A shared workspace where the data, the processing, the analyses and the discussion sit together, so context is discoverable instead of transmitted by conversation.

Compute is a long way from the data

Analysing a large archive means downloading it first, then keeping a second copy in step, then a third for the collaborator abroad. The copies drift and the storage bill grows.

- The catalogue references artefacts in shared storage rather than hoarding copies of them, and co-located compute, so notebooks and pipelines run next to petabyte-scale archives, is committed roadmap.

The product is built from a small number of clearly named parts. Learning them takes an afternoon, and everything else in the platform is an arrangement of these.

CATALOGUE

Sources

An upstream feed the workspace pulls from: a telescope, a satellite downlink, a ground sensor, an archive, a simulation, an API, a file drop or a stream. Each carries its endpoint, a reference to a stored credential, connector settings, a last-synced time, and a live health state running from configuring through connected, syncing, degraded, error and disabled.

CATALOGUE

Datasets and schema fields

A dataset is a catalogued, queryable unit of data with coverage, instrument metadata, format, size, version, a lifecycle and an access level. Each column is described as a schema field with a type, a unit, nullability and a note on where it came from, which is what makes a dataset trustworthy at a glance.

CATALOGUE

Collections

A curated grouping of related datasets for a mission, a survey or a project, which can be marked public when the team is ready to share it openly.

PROCESSING

Pipelines

A reusable processing definition expressed as a declarative graph of typed steps: ingest, transform, calibrate, analyse, export and notify. Pipelines are versioned, carry a lifecycle, and run manually, on a schedule, when data lands, or from a webhook.

PROCESSING

Runs, steps and lineage

A run is one execution against an input dataset, with a run number, status, timings, resolved parameters and aggregated metrics. Ordered steps record exactly what each one read and produced, plus logs, errors and per-step telemetry, which is where provenance actually comes from.

RESEARCH

Notebooks

A reproducible notebook bound to one or more datasets, with its runtime captured and its environment pinned down to package versions and an image identifier, so it can be re-run faithfully. Notebooks can be shared by link and are the unit that becomes a release.

RESEARCH

Analyses

A single analysis run against a dataset or notebook, recording the method, structured results, detected anomalies and summary statistics, a reference to the rendered figure, and whether an AI assist was involved. Results are charted from the real numbers, with an honest empty state when there is nothing to plot.

PUBLISHING

Releases

A notebook and its datasets packaged as a citable publication with a version, abstract, authors, a licence and a persistent identifier, moving through draft, under review, published and retracted.

GOVERNANCE

Data access grants

A governed grant of dataset access to a named collaborator with explicit scopes such as read and query, an optional embargo or expiry, and a status running through pending, active, revoked and expired. No secret ever changes hands.

Fifteen situations the product was built around, drawn from the personas and journeys in its own specification. Where a capability is committed roadmap rather than available today, it says so.

RESEARCH SCIENTIST

Start the day from one console instead of nine tabs

The bind. Data work begins with a hunt across portals, drives and inboxes, and the first twenty minutes go on remembering where things are.

A single dashboard gathers your datasets, sources, pipelines, runs, notebooks and releases into summary tiles, alongside recent activity and quick actions such as register a dataset, connect a source or create a pipeline. Navigation collapses the whole product into four areas: catalogue, pipelines, research and data access.

- One front door for a researcher, an engineer and a lead alike, so the work starts where the data is.

RESEARCH SCIENTIST

Find the right dataset and know whether to trust it

The bind. Narrowing a question to the observations that answer it usually means learning a different query language for every archive that holds them.

Type a few words and narrow by instrument type, access level, format, collection and status, several filters at once, with the count beside each filter derived from the same live set as the results so the two always agree. Open a dataset to see its schema fields, their units and their provenance before you build anything on top of it.

- Discovery that reaches a working subset in minutes, and a schema preview that tells you whether the data means what you think it means.

RESEARCH SCIENTIST

Turn an exploration into work someone else can repeat

The bind. The notebook that produced the figure runs on your machine and nowhere else, and you are quietly afraid of being asked to prove it.

Launch a notebook bound to the datasets it uses, with the runtime captured and the environment pinned to exact package versions and an image identifier. Record analyses against it with their method, structured results and rendered output. Share the notebook by link so a collaborator opens precisely what you see.

- A result that carries its own recipe, so it can be re-run, audited and built on rather than taken on faith.

DATA ENGINEER

Bring telescope, sensor and mission feeds into one shape

The bind. You have written the same ingestion script a dozen times, each one slightly different, and each one now a thing you personally maintain.

Connect each feed as a typed source with its connector configuration and a reference to a stored credential, then register what lands as a dataset with its schema, coverage and access level attached. Sources are modelled for telescopes, satellites, ground sensors, archives, simulations, APIs, file drops and streams rather than forced into one generic shape.

- One integration pattern instead of a dozen bespoke scrapers, and observations that arrive already described.

DATA ENGINEER

Make processing a definition rather than a folder of scripts

The bind. Calibration and reduction live as scripts nobody has versioned, so nobody can say with confidence how last quarter's table was produced.

Compose calibration, reduction and enrichment into a versioned pipeline of typed steps, then set it to run manually, on a schedule, the moment data lands, or from a webhook. Start a run and watch its status, its resolved parameters and its aggregated metrics, with each step recording what it consumed and produced.

- Downstream consumers can see exactly how a table was made, and the pipeline outlives the person who wrote it.

DATA ENGINEER

Keep the institutional holding findable and affordable

The bind. Data is not lost by deletion, it is lost by neglect, and the storage bill grows whether or not anyone can still find what is on it.

Everything the workspace holds is catalogued with queryable metadata covering instrument, target, epoch, format, processing level and access policy, and moves through a governed lifecycle of draft, ingesting, available, deprecated and archived. Lifecycle-aware tiered storage, so cold archives move to cheaper tiers without leaving the catalogue, is committed roadmap.

- The catalogue that powers researcher discovery is the same one that keeps the holding governed and defensible to a funder.

RESEARCH SCIENTIST

Know at a glance which of your feeds is actually flowing

The bind. Data arrives from telescopes, satellites and ground stations at once, and you often discover a stalled feed only when an analysis comes back wrong.

Every connected source carries a live health state moving through configuring, connected, syncing, degraded, error and disabled, plus a last-synced time. Mark a source synced when a fresh pull lands and register the dataset straight into the catalogue. Streaming dashboards, throughput statistics and transient alerts are roadmap; connection, health and sync are what ships today.

- A stalled feed becomes visible on the day it stalls rather than in the post-mortem.

RESEARCH LEAD

Publish work others can find, reuse and credit

The bind. Data that cannot be cited rarely gets reused, and preparing a proper deposit is a separate project nobody has time for.

Turn a working notebook and the datasets beneath it into a versioned release carrying an abstract, an author list, a licence and a persistent identifier, then move it through draft, under review and published. The workflow ships today. Automated registration with an external registrar such as DataCite or Zenodo is roadmap, so the identifier is supplied at publish time for now.

- Good work becomes durable and findable beyond the group that produced it, and a stronger line in the next grant application.

DATA STEWARD

Share restricted data without handing over a secret

The bind. Too open and you leak; too closed and the collaboration stalls. The usual compromise is a shared drive nobody can audit.

Datasets carry an access level of public, workspace, restricted or embargoed. Issue a grant to a named collaborator with explicit scopes such as read and query, attach an embargo or expiry so access opens and closes on its own, and revoke at any point. Sources reference their credentials rather than storing them, so a grant is a permission and never a copied key.

- Who can see what is always answerable, with a status and an audit trail behind every grant.

RESEARCH LEAD

Give a distributed team one place to work

The bind. Collaborators in three time zones coordinate through email threads, and a new joiner spends weeks reconstructing what everyone else already knows.

A workspace owns the datasets, the pipelines, the notebooks, the releases, the members and their roles, with inline comments and discussion threads attached to the objects they are about. A new collaborator joins and sees the data, the processing and the context in one place.

- Onboarding archaeology stops being part of the job, and the conversation happens next to the data it concerns.

DATA ENGINEER

Let an assistant do the repetitive cataloguing

The bind. Registering a source, cataloguing a dataset and declaring a column at a time with type, unit and provenance is exactly the work you would rather describe once.

A Model Context Protocol server exposes the catalogue to an agent in your editor or desktop client as 42 tools spanning sources, datasets and their fields, collections, pipelines, runs and steps, notebooks, analyses, releases and grants. Two guided workflows ship with it: onboarding a dataset end to end, and taking research through to a release with a lineage summary at the finish.

- An agent works your catalogue through the same permission-checked operations the console uses, holding exactly the rights your role grants.

RESEARCH LEAD

Tell a machine-assisted finding from a hand-derived one

The bind. The hard question about AI in science is not whether a model can help, it is which published results a model touched and whether you can still defend them.

Starting an analysis lets you mark it AI-assisted and name the method. Both stay on the record, the row is badged wherever it appears, and completing the analysis captures the results and an optional output file, charted from the real numbers. Reviewing a release later, you can walk down to the analyses beneath it and see which were machine-assisted.

- Attribution that survives the session, so a reviewer months later can see exactly what a model contributed.

PRINCIPAL INVESTIGATOR

Separate the duties that should never sit with one person

The bind. One identity often wears several hats, and the usual outcome is that everyone ends up with more access than the job needs.

Six roles are seeded into every workspace automatically. The pipeline engineer who wires ingestion holds no permission at all on access grants. The data steward who governs sharing holds none on notebooks or analyses. A graduate contributor can run analyses but cannot stamp the group's work with a citable identifier, because releases are read-only for that role.

- Separation of duties expressed in the role catalogue itself rather than in a policy document nobody enforces.

REPRODUCIBILITY AUDITOR

Reproduce a result and review who saw what

The bind. A reviewer or a funder asks how a figure was produced and who held access to the underlying data, and answering means interviewing people.

A read-only auditor role reaches every resource, including sources and access grants, and holds write access to nothing. The activity timeline can be filtered by actor, action or date and exported, and every change is recorded with credential-shaped values stripped out and the entry size bounded.

- A figure can be traced to its inputs and the data-sharing trail shown to an auditor without loosening anyone's permissions.

INSTITUTION

Run many labs under one roof without changing product

The bind. An observatory with several instruments, or a university with dozens of groups, ends up with dozens of local answers to the same governance question.

Many workspaces sit under one tenant with consolidated governance and usage, so an administrator can compare teams and steward storage and compute centrally. Enforced single sign-on, directory provisioning, data residency, dedicated tenancy and self-hosting are the Enterprise shape. Cross-institution data federation is roadmap.

- The path from one graduate student to an institution-wide deployment does not require re-platforming.

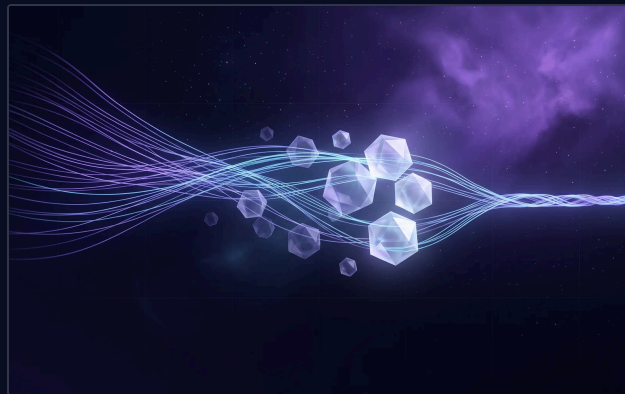
06 Built AI-native

Governed before it was generative, and labelled line by line

Most product assistants are a blank box. You open one and the first thing it asks for is the context already on your screen: the dataset identifier, the run number, the notebook you were reading. CosmicIntersection works the other way round. The assistant is handed your position in the product before your question reaches it, so a question about the thing in front of you needs no identifier at all.

That only works because the catalogue is structured. Sources, datasets, schema fields with units attached, runs, steps, analyses and releases are real objects with real relationships, not files in a folder with prose wrapped around them. An assistant can answer from them rather than around them, and the chain from source to dataset to run to notebook to analysis to release is a path it can actually walk. Where most science tooling can only summarise whatever documentation somebody wrote, this one reads the record.

The governance came first, which is unusual and deliberate. Permissions, the workspace boundary, the redacted audit trail and the AI-assist mark on an analysis were all in place before any model was wired in. The assistant and any outside agent are ordinary clients holding exactly the rights of whoever is calling, so there is no AI back door to audit later. Roughly half of what follows is still roadmap, and it is labelled as such rather than blurred into the shipped part.



06 Four pillars



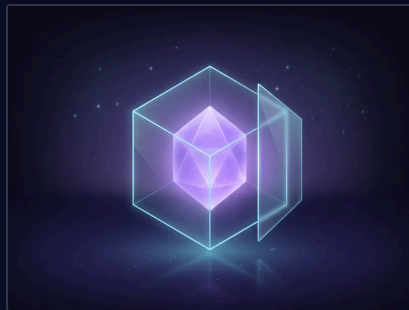
Structure, not a folder of files

Sources, datasets, fields, runs, analyses and releases are typed objects with units attached, so an assistant answers from your actual catalogue rather than from prose about it.



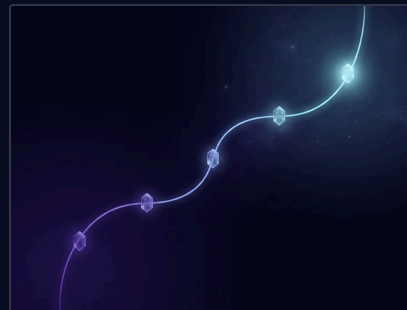
AI assistance stays on the record

An analysis records whether a machine helped and which method ran, badged everywhere it appears, so it never quietly becomes a hand-derived result.



Governed before it was generative

The assistant and any outside agent hold exactly your permissions, write only inside your workspace, and land in the same audit trail as anyone else.



Lineage is the grounding

Source to dataset to run to notebook to analysis to release is a chain you can walk today, and the same chain a drafted release would be assembled from.

An assistant on every page

Shipped. It sits on the dashboard, catalogue, discovery, pipelines, runs, notebooks, analyses, releases, lineage and data access screens, and the conversation follows you as you navigate.

You never have to leave the record you are reading in order to ask about it.

A lineage chain you can walk

Shipped. Source, dataset, run, notebook, analysis and release join into one clickable chain in either direction, built from the real links between your own records.

It is the honest answer to where a figure came from, and the grounding any future drafting would read.

This dataset means this dataset

Shipped. Eight kinds of detail page, covering dataset, source, collection, pipeline, run, notebook, analysis and release, are recognised by name and resolved into a typed reference before your question is sent.

No identifier pasting, and on pages that are not about one record it names the section rather than inventing one.

An AI-assist mark on every analysis

Shipped. A switch on the start-analysis form records that a machine helped and which method ran, and the row is badged on desktop tables and mobile cards alike.

A reviewer months later can tell a machine-assisted finding from a hand-derived one.

Faceted catalogue discovery

Shipped. Keyword search combines with filters on instrument type, access level, format, collection and status, and the count beside each filter comes from the same live set as the results.

Every number you see is the number of datasets you would actually get.

Results drawn from real numbers

Shipped. Completing an analysis records its outcome, its numbers and an optional output file, and the chart is drawn from that payload, falling back to a list and then to a clear no-results note.

You never see decorative sample data standing in for a result.

42 agent tools over your catalogue

Shipped. An outside agent can register sources, catalogue and publish datasets, declare fields, manage collections and pipelines, start and advance runs, write notebooks, run analyses, cut releases and manage access.

The repetitive onboarding sequence becomes something you describe once.

The model that computes an analysis

In progress. The record side is done: starting an AI-assisted analysis opens a running record that waits for a model or worker to report its results back. Choosing and integrating that provider is open work.

You bring the method today, and the platform holds the request, the result and the attribution around it.

Two guided agent workflows

Shipped. Rather than let an agent guess the order, a dataset-onboarding walkthrough and a research-to-release walkthrough ship with the agent server, the second ending in a summary of the lineage it created.

Agents follow the order of operations the product actually expects.

A confirmation gate before an agent deletes

In progress. The agent shell now refuses generic delete tools without an explicit confirmation. Archiving a dataset, pipeline, notebook or source, and revoking a grant, still need their destructive annotations before the gate catches them.

Until that lands, the safer posture is to give agents read-heavy roles.

Per-workspace metering of assistant turns

Shipped. A credit check runs before a turn and the reported token usage is posted back per message, charged to the workspace that made it.

AI spend is attributable rather than invisible.

A written playbook for this catalogue

Roadmap. Today the assistant is stronger at reading and explaining your catalogue than at driving long multi-step changes through it, because no product-specific agent guide has been written yet.

Once written, it gives the assistant the right order of operations rather than leaving it to infer one.

Asking for data in plain language

Roadmap. Observations of this region, in this band, during this epoch, returning ranked datasets, is the target. Matching today is on words and filters only, and nothing ranks by meaning.

It is the difference between filtering a catalogue and asking it a question.

Release drafting from lineage

Roadmap. Assembling a draft release, with its abstract, authors and dataset bundle, from the lineage already captured, for a lead to correct and publish.

Every fact a release form asks for already exists somewhere in the platform by the time someone sits down to write it.

Field and unit inference

Roadmap. Proposing a dataset's field types, units and provenance from the file itself, inside the fixed vocabulary of coordinate, spectrum, geometry, image reference, timestamp and the ordinary scalars.

Schema inference here is constrained classification over a typed vocabulary, not free-text generation.

Anomaly and transient detection

Roadmap. Detection across streaming and archived observations that flags the unexpected for human review with its supporting context attached, landing as AI-assisted analyses so attribution comes with it.

It is what would scale a small team's vigilance across a growing flood of observations.

Failed-run triage

Roadmap. Reading a failed run's step graph, timings, metrics and errors, grouping it with the failures it resembles, naming the step where the chain actually broke and proposing the next action.

A person would start from a hypothesis rather than from a log.

A cap on AI-assisted analyses

Roadmap. Assistant turns are counted, and creating a dataset, publishing one and starting a run all draw against quota. Starting an AI-assisted analysis is attributed but not yet capped.

Named openly, because an uncapped action is a budget question rather than a feature.

06 And what keeps it in bounds

1 of 2

AI gets your permissions, nothing more

The assistant and any outside agent are ordinary clients holding exactly the rights of whoever is calling. Every read and write passes the same permission check as the console, so there is no separate AI path to review.

Changes stay inside your workspace

Every write is confined to the caller's workspace, so an identifier guessed from another team's data returns nothing rather than altering it. This holds for a person, the assistant and an agent alike.

Every change is logged, safely

Changes are recorded with credential-shaped values stripped out and the entry size bounded, so a trail of AI-driven changes never becomes either a leak or a runaway log.

Machine assistance is on the record

An analysis keeps whether AI helped and which method ran. Working back from a citable release you can see which findings beneath it were machine-assisted, which is what makes the rest of the governance meaningful.

It proposes, you commit

The three planned judgements, field inference, run triage and release drafting, are each designed to arrive as a reviewable proposal. A drafted release would enter as a draft and nothing more.

Publishing stays a human act

Publishing a release stamps its persistent identifier and remains a permission-checked action taken by a person. An AI-drafted abstract that someone publishes is their statement, not the model's.

06 And what keeps it in bounds

2 of 2

Run lifecycles cannot be advanced quietly

Advancing a run through its states is a permission-checked change with validated transitions, so an agent cannot silently re-run production processing. Re-running is something a person authorises.

AI spend is attributed to a workspace

Assistant turns are counted and charged to the workspace that made them. Stated openly: starting an AI-assisted analysis is not yet counted against an allowance, and closing that gap is committed work.

Destructive agent actions are being gated

The agent shell enforces an explicit confirmation on generic delete tools. Product-specific archive and revoke tools still need their destructive annotations, so agents should hold read-heavy roles until that ships.

No model provider is connected yet

No AI model provider is wired into CosmicIntersection, so there is no training arrangement to describe and none will be claimed. The in-app assistant runs on shared platform infrastructure that reports a model name purely so turns can be counted. Bringing your own model or key is planned.

No accuracy figure is published

There is no benchmark, no metric and no defined fallback for AI output here yet, so no percentage appears anywhere. Building the evaluation set comes before publishing a number.

Analyses are workspace-scoped and audited

AI-assisted analyses are ordinary workspace-scoped records, checked at the edge on write and recorded by the same audit layer as everything else, with sensitive values redacted.

06 Reachable from, and in numbers

REACHABLE FROM

- The in-app assistant, on every CosmicIntersection screen, carrying your current record as context
- Any editor or desktop client that speaks the Model Context Protocol, through the agent server
- The federated API, which is what the assistant, the agents and the console all call
- The analysis record itself, where the AI-assist mark and the method are stored and badged
- The lineage view, which is the grounding a drafted release would be assembled from
- The workspace usage view, where assistant turns are counted against the workspace that made them
- The audit and activity timeline, where AI-driven changes appear alongside human ones

IN NUMBERS

- 42 agent tools across the catalogue, processing and research layers, plus the platform surfaces the product inherits
- 2 guided agent workflows: dataset onboarding, and research through to a release with a lineage summary
- 8 kinds of detail page recognised by name, so that this dataset resolves without an identifier
- 9 governed resource types, each with its own permission level per role
- 8 source types modelled: telescope, satellite, ground sensor, stream, simulation, archive, API and file drop
- 5 science-native field types beyond the ordinary scalars: coordinate, spectrum, geometry, image reference and timestamp
- 6 typed pipeline step kinds: ingest, transform, calibrate, analyse, export and notify
- 4 dataset access levels: public, workspace, restricted and embargoed
- 0 accuracy claims, because the evaluation set does not exist yet

STATED PLAINLY

What the AI does not do yet

What the AI does not do yet, stated plainly. It cannot rank the catalogue by meaning, so plain-language data search is not available and matching is on words and filters. It cannot infer a schema, triage a failed run or draft a release: those three judgements are designed but unbuilt, and deliberately sequenced behind a shared multi-model runtime rather than a bespoke integration that would have to be unpicked later. It does not perform anomaly or transient detection. It does not compute the results of an AI-assisted analysis; the platform records the request, the attribution and the returned numbers while the computation happens elsewhere. No model provider is connected, so no provider, no training terms and no bring-your-own-key option can be claimed today. Starting an AI-assisted analysis is not yet capped against an allowance, and the destructive-action confirmation gate does not yet cover the product-specific archive and revoke tools. There is no benchmark and no accuracy figure, which is why none appears anywhere in this document.

Six roles are seeded into every workspace automatically, in descending priority. Nobody creates them by hand. What a role does not list, it cannot reach, and several of the gaps below are deliberate separations of duty rather than oversights.

Principal Investigator

Reproducibility, who reaches embargoed data, and what leaves the group under the group's name

Owns the research programme end to end: the catalogue, the processing, the research output, the citable releases and every access grant. This is the only role that can retire a source, dataset, collection, pipeline or notebook.

- Full control of all nine resource types
- Archive sources, datasets and pipelines
- Issue and revoke access grants
- Publish citable releases
- Approve what the group puts its name to
- Govern embargoed and restricted data

Data Steward

How the holding is described, versioned and released, and what a funder will ask about it

Curates the catalogue and governs who may reach restricted and embargoed datasets. Owns sources, datasets, collections and grants outright, reads pipelines, runs and releases, and holds no permission at all on notebooks or analyses.

- Curate datasets and schema fields
- Organise collections by mission or survey
- Connect and retire sources
- Issue, scope and revoke grants
- Trace provenance without altering it
- Keep the catalogue lifecycle honest

Pipeline Engineer

Ingestion that stays up, processing that stays repeatable, and lineage that stays complete

Connects sources, builds the versioned pipelines that turn raw observations into analysis-ready datasets, and starts and monitors their runs. Deliberately holds no permission on access grants at all.

- Connect and configure sources
- Author pipelines and their step graphs
- Start, advance and cancel runs
- Append run steps and read logs
- Catalogue pipeline output as datasets
- Read notebooks, analyses and releases

Research Scientist

Getting from a question to a defensible answer, and getting credit for it

Runs reproducible notebooks and analyses against catalogue datasets and turns finished work into citable releases. Sources stay out of reach, because connector records carry upstream endpoints and credential references.

- Author and pin notebooks
- Run and complete analyses
- Mark an analysis AI-assisted
- Package work as a citable release
- Read datasets, collections and runs
- Share a notebook by link

Research Contributor

Doing real science on day one without being able to speak for the group

The graduate researcher. Works notebooks and analyses against the catalogue and reads the lineage behind a result, but cannot publish. Releases are read-only here, which is the single difference from the Research Scientist.

- Author and run notebooks
- Record analyses and results
- Read the catalogue and collections
- Follow lineage back to source
- Read releases without publishing
- Learn the workflow safely

Reproducibility Auditor

Whether a figure can be reproduced, and who was granted what and when it expired

Reviews lineage, results, releases and the data-sharing history across the workspace and changes nothing. The only read-only role that reaches every resource, including sources and access grants.

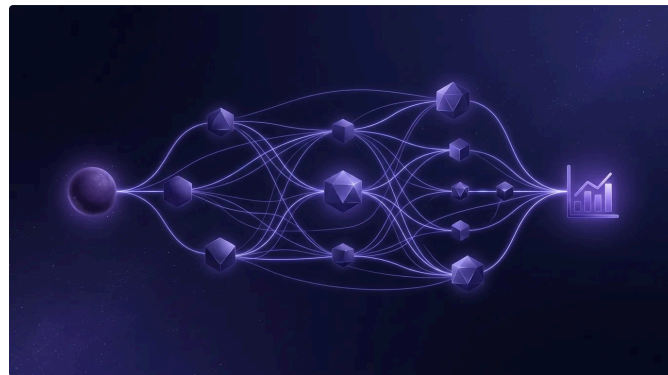
- Read every one of the nine resources
- Reproduce a result from its inputs
- Review grants, embargoes and revocations
- Filter and export the activity timeline
- Verify a release against its lineage
- Write nothing, anywhere

08 Under the hood

CosmicIntersection is API-first rather than screen-first. The catalogue, the processing layer and the research layer are exposed as one federated GraphQL API, and the web console is a client of it like any other. A team can drive the platform from a notebook, a dashboard, an automation job or a partner tool, and query relationships across datasets, runs and releases in a single request, instead of rebuilding data plumbing to reach its own records.

Everything belongs to a workspace. Identity is validated and permissions are enforced at the API edge, and every write is scoped to the caller's workspace with a strict single-row assertion, so an identifier guessed from another tenant cannot be altered. Product-owned types are namespaced so they compose cleanly alongside the rest of the platform rather than colliding with it.

The catalogue references rather than hoards. Raw artefacts, notebook documents and rendered figures live in the platform's shared file storage and are referenced by the catalogue, never copied into it, so a large artefact is stored once under one consistent storage policy. Identity, billing, files, tagging, conversations, notifications and audit are shared platform services rather than things this product rebuilt.



THE SPLIT THAT MATTERS

Recording versus executing. What ships is the record spine: pipeline definitions, runs, steps, validated state transitions, per-step lineage, metrics and pinned notebook environments. What is roadmap is the managed engine that executes step code and notebook kernels in-platform. Today a run advances through its states and records what an external worker reports, and a notebook's environment is pinned and versioned rather than executed for you. Saying so plainly matters more than the claim would.

08 Capability by area

Catalogue

Typed sources with health and sync state, datasets with lifecycle, access level, coverage, format, size and version, per-column schema fields with type, unit, nullability and provenance, and collections for missions, surveys and projects.

Research

Notebooks bound to datasets with runtime captured and environment pinned to package versions and an image identifier; analyses carrying method, structured results, an output reference and an AI-assist flag; link sharing for collaborators.

Discovery

Case-insensitive keyword search combined with facets on access level, format, source, collection and status, always scoped to your workspace, with filter counts derived from the same live set as the results.

Publishing

Releases bundling a notebook and its datasets with version, abstract, authors, licence and a persistent identifier, moving through draft, under review, published and retracted.

Processing

Declarative step graphs across six typed step kinds, four trigger modes covering manual, scheduled, on-ingest and webhook, versioned definitions, and a validated run state machine with per-run metrics and resolved parameters.

Governance

Permission directives on every operation enforced at the edge, workspace-scoped writes, quota checks at the action layer on dataset creation, dataset publication and run starts, and an audit trail with sensitive values redacted.

Lineage

Ordered run steps record exactly which datasets and files each step consumed and produced, alongside logs, errors and per-step telemetry, with a flag on each dataset recording whether its downstream provenance has been wired.

Events

Product events are published and consumed on a shared event stream, so pipeline completion, dataset publication and release events can drive notifications and downstream integrations.

08 Standards and integrations

OPEN STANDARDS IT SPEAKS

- GraphQL, as one federated API across the catalogue, processing and research layers
- OpenID Connect single sign-on for people
- OAuth 2.0 client credentials for integrations, and device-code login for terminal clients
- SAML and SCIM directory provisioning, enforced on Enterprise
- Model Context Protocol, for assistants and agents acting on the catalogue
- DOI-style persistent identifiers on citable releases
- FAIR data principles, findable, accessible, interoperable and reusable, as the metadata design goal
- Cron-style scheduling for pipeline triggers, and webhooks in both directions
- Semantic versioning on pipeline definitions, datasets and releases
- WCAG 2.1 AA as the accessibility conformance target, with a self-assessment already drafted

PLUGS INTO

- Telescope and observatory feeds, modelled as a first-class source type
- Satellite downlinks and spacecraft telemetry
- Ground sensors and instrument readings
- Public mission and survey archives, modelled as archive sources, with per-archive connectors being built out
- Simulation output and generated datasets
- Any REST or HTTP API, plus plain file drops and continuous streams
- Your identity provider, through single sign-on and directory provisioning
- DOI registrars such as DataCite and Zenodo for automated minting, on the roadmap
- Downstream systems, through webhooks and event-driven notification
- Editor and desktop AI assistants, through the agent server

08 How it deploys

Shared multi-tenant cloud

The default. Your workspace is isolated by construction, with the tenant boundary applied to every read and write, and nothing to operate yourself.

Dedicated tenancy

An Enterprise option for institutions whose procurement or security posture requires isolation beyond the shared environment.

Self-hosted and air-gapped

An Enterprise option for facilities that cannot send research data outside their own network at all, including fully disconnected deployments.

Data residency and multi-region

An Enterprise option where the jurisdiction the data rests in is a contractual requirement rather than a preference.

API-first from any surface

Because the console is a client of the same API you get, a team can build its own dashboards, notebooks and automation without waiting for a feature to reach the interface.

Custom domains

An Enterprise option for institutions that need the platform to sit under their own name.

Security here is not a module that arrived late. It is inherited from the shared platform the product sits on, so the same controls apply uniformly across every workspace, and it was in place before the first dataset was catalogued.

Permissions enforced at the edge

Every query and every change carries a permission directive that the API edge enforces from the composed schema. The product service never re-implements authorisation and never validates a token itself, so there is exactly one place where access is decided.

Workspace-scoped writes

Every change filters by workspace with a strict single-row assertion. An identifier guessed from another tenant returns nothing rather than being altered, for a person, the assistant and an outside agent alike.

Six seeded roles with real separation of duty

The engineer who wires ingestion holds no permission on access grants. The steward who governs sharing holds none on notebooks or analyses. A contributor cannot publish. Omitting a resource from a role is the access control.

Dataset access levels

Every dataset carries an access level of public, workspace, restricted or embargoed, enforced through the platform rather than by convention.

Scoped, time-bounded access grants

Grants are issued to a named collaborator with explicit scopes, an optional embargo or expiry, and a status lifecycle covering pending, active, revoked and expired. Every grant carries a reason and an audit trail.

Secrets stay in the vault

A source references its credential by reference, held in the workspace secret store. No secret is ever written onto a source record, which is also why connector records are withheld from both research roles.

Redacted, bounded audit

Every change is recorded with credential-shaped values stripped out and the entry size bounded, so the audit trail cannot itself become a leak or a runaway log.

Searchable, exportable activity timeline

Every action is attributable to an actor with its target and its result, filterable by actor, action or date, and exportable for an auditor or a funder review.

Multi-tenant isolation by default

Workspaces and organisations are isolated so a tenant only ever sees its own data, with dedicated tenancy and self-hosting available where procurement requires it.

Encryption in transit and at rest

Traffic between your device and the platform is encrypted with industry-standard transport security, and stored data is encrypted at rest on enterprise cloud infrastructure with geographic replication.

Multi-factor authentication

Accounts can be protected with additional verification factors including authenticator apps, with custom multi-factor and session policies and network restrictions available on Enterprise.

Quota enforcement at the action layer

Creating a dataset, publishing one and starting a pipeline run are each checked against workspace quotas at the moment of the action, so a limit surfaces then rather than as a surprise afterwards.

Least privilege as an operating habit

Workspace members, roles and dataset access are meant to be reviewed regularly, and the read-only auditor role exists so that a review never requires handing anyone write access.

Responsible disclosure with a paid bounty

Security reports go to a published address, with a bug bounty offering 50,000 rupees for critical findings, 25,000 for high severity and 10,000 for medium severity.

STATED PLAINLY

No third-party certifications today

CosmicIntersection holds no third-party security certifications today. It is pre-launch, and claiming otherwise would be the easiest lie in this document. What exists is a control environment modelled on the ISO 27001 framework, designed against SOC 2 criteria, and built to align with GDPR and India's DPDP Act from the start. What does not exist yet is independent assurance that it operates as described. The plan is public and dated: a SOC 2 Type I report with observation starting in the first quarter of 2027 and a Type II report targeted for the third quarter; an ISO 27001 certification audit targeted for the second quarter of 2027; and an ISO 27701 privacy audit able to run alongside it. Privacy readiness under DPDP and GDPR is in progress with an external assessment targeted for the first quarter of 2027; an internal WCAG 2.1 AA self-assessment is drafted with a third-party audit targeted for the second quarter of 2027; and EU AI Act classification work covering AI-assisted analysis is queued. Compliance evidence exports and streaming to a security information and event management system are roadmap, not shipped. Subprocessors are published on the website and updated as launch-critical providers are confirmed. If your procurement needs a certificate today, this is the honest answer: ask for the roadmap and the security exhibit, not for a badge we do not hold.

One product, many front doors

The web console is the complete product today. The other surfaces are specified and sequenced, and this list says plainly which is which rather than presenting everything as though it had shipped.

Web console

Available. The full product in a browser: catalogue, sources, collections, discovery, pipelines, runs, notebooks, analyses, releases, lineage and data access, responsive down to mobile, with no native install.

Product app shell

Available. The same experience composed into the wider Burdenoff workspace app, so CosmicIntersection sits alongside the other products a team uses under one identity.

Federated API

Available. The API every surface is built on, and the one you build against. Query relationships across datasets, runs and releases in a single request from notebooks, dashboards and partner tools.

In-app assistant

Available. A floating assistant on every screen that already knows which dataset, run or notebook you have open, with its turns counted against your workspace.

Agent server

Built. A Model Context Protocol server exposing 42 catalogue tools and two guided workflows to assistants in your editor or desktop client, authenticating as its own client so an agent holds exactly your rights.

Command-line client

Planned. Terminal-native automation and bulk operations against the same API, for scripting and continuous-integration use. Not yet published, and listed as planned for exactly that reason.

One product, many front doors

Node and Python SDKs

Planned. Typed programmatic access for automation and integrations, and an async-first Python client aimed at notebooks and data workflows. Neither is published to its registry yet.

Editor and browser extensions

Planned. An editor sidebar for sources, catalogue, pipelines and notebooks, and a browser side panel for quick catalogue access while you read a paper or a portal.

Mobile apps

In development. Android and iOS shells are built with push wiring, offline caching of researcher metadata and a working release pipeline, but neither is listed on an app store yet.

Documentation

In progress. Product guides, API reference and integration tutorials, published as the product moves towards general availability.

Changelog and roadmap

Available. Published openly, and the authoritative source for what has shipped and what is next. Where this catalogue and the changelog disagree, the changelog wins.

Community hub

Planned. Question-and-answer, discussion and user stories, signed in against your workspace identity.

11 Plans

Four tiers, with a clear path from one graduate student to an institution-wide deployment and no re-platforming on the way. The data model, the governance, the audit trail and the API are identical on every plan; what changes is scale, seats, and the isolation and compliance options. These are the planned launch prices. CosmicIntersection is pre-launch and has no customers, revenue or usage data.

FREE

\$0 forever

Individual researchers, graduate students and single-investigator projects evaluating the full platform.

- 3 data sources, 50 catalogue datasets, 5 GB ingest a month
- 2 pipelines with 5 runs a month, manual triggers only
- 3 notebooks, 20 analyses a month, 50 compute credits
- 5 scoped data-access grants and 10 GB shared storage
- The complete product surface, with no card required
- No citable releases or persistent identifiers; those unlock at Pro
- Exports are clean, with no watermarking

PRO

\$49/month, or \$39/month billed annually

Independent researchers, postdocs and single-investigator projects that have outgrown the free ceiling.

- 25 data sources, 1,000 datasets, 100 GB ingest a month
- 25 pipelines, 500 runs a month, 5 scheduled pipelines
- Unlimited notebooks under fair use, 1,000 analyses a month
- 25 citable releases and 5 persistent identifiers a month
- 100 active access grants with scopes and expiry
- 100 GB shared storage and 90-day audit retention
- Email support within 48 hours, 99.5% availability target

11 Plans

BUSINESS

\$399/month, or \$319/month billed annually, 10 seats included

Research groups, labs and small science teams collaborating across datasets and pipelines continuously.

- 10 seats, 3 workspaces, 25 custom roles
- 100 sources, 25,000 datasets, 1 TB ingest a month
- 500 pipelines, 10,000 runs a month, 100 scheduled
- 1,000 releases and 100 persistent identifiers a month
- 5,000 active grants, dataset embargoes, advanced lineage views
- 2 TB pooled storage and 365-day audit retention
- Priority support within 8 hours, 99.9% availability target

ENTERPRISE

From \$12/seat/month list, with a \$9/seat/month annual floor at 25 seats and above

Research institutions, observatories, space agencies, national labs and regulated or multi-lab organisations.

- Custom quotas on sources, datasets, ingest, runs, compute and releases
- Enforced single sign-on, directory provisioning, custom session policy
- Advanced audit, access review, security-event streaming, legal hold, custom retention
- Data residency, customer-managed keys, private ingress, credential vault controls
- Dedicated tenancy, self-hosting, air-gapped and multi-region deployment
- Consolidated multi-lab governance and billing across many workspaces
- Dedicated success contact, named channel, 99.95% or better availability target

11 How the commercials work

Add-ons expand any tier without forcing a jump

Capacity packs cover seats, sources, datasets, ingest, runs, compute credits, storage, API volume and integrations, from \$15 a month for 100 GB of storage up to \$349 for 10 TB of ingest a month. Capability packs cover an AI tier at \$79, a releases and identifiers pack at \$49, and sandbox identity test packs. Free users upgrade first, and no add-on unlocks production governance below Enterprise.

Bring your own AI key

AI-assisted analysis allowances are measured as orchestration operations rather than resold inference. You connect your own provider key or subscription, and the quota counts what the platform coordinates rather than what the model costs. Production single sign-on and directory enforcement stay an Enterprise capability; the test packs are sandbox connectors only.

How billing actually works

Free needs no card. The first paid attempt can start a 14-day Business trial. Business is a flat team fee with 10 seats included, monthly or annual. Enterprise is an annual per-active-seat contract, invoiced on 30-day terms by default, with compute and ingest above the contracted allowance metered through credit and volume packs. Regional price bands apply, and storage and shared API limits are pooled across Burdenoff products on the same account.

Services when you want a hand

A one-day quick-start workshop is \$2,500 and a standard implementation is \$12,000. Migration from an existing repository or legacy archive starts at \$30,000, integration work at \$7,500, and support blocks at \$750 for five hours. Academic and non-profit programmes apply to verified research and teaching use, and public-sector procurement follows the Enterprise quote path.

12 Why teams choose it

One model for the whole lifecycle, not four tools

Sources, datasets, schema fields, collections, pipelines, runs, notebooks, analyses, releases and grants sit in one connected graph. No competitor covers catalogue, processing, reproducible analysis and citable publishing end to end, which is why teams stitch four systems together and lose the joins between them.

Governance that came before the features

Roles, audit, quotas, tenant isolation and access grants were in place before the first model or the first agent. Most platforms sell this back later as an enterprise tier; here it is on the free plan.

Science-native semantics rather than generic tables

Fields can be typed as coordinate, spectrum, geometry, image reference or timestamp, and sources as telescope, satellite, ground sensor, stream, simulation, archive, API or file drop. Observational data is modelled as what it is, rather than remodelled into a lakehouse's idea of a table.

Sharing that expires on its own

Restricted and embargoed data moves through scoped, time-bounded, revocable grants with a full status lifecycle, so open-science releases and sensitive mission data can genuinely coexist in one workspace.

Reproducibility as structure, not discipline

Environment pinning, dataset binding and per-step lineage recording are properties of the objects themselves, so provenance is captured rather than maintained. A group does not have to be unusually rigorous for its results to be reproducible.

12 Why teams choose it

API-first, so it is infrastructure rather than a screen

The console is a client of the same federated API you get. A research software engineer can build the dashboard the group actually wants without waiting for it to reach the product, and query across datasets, runs and releases in one call.

The rest of the platform arrives with it

Single sign-on, billing, file storage, tagging, conversations, notifications, search, translation, accessibility, guided tours and a developer portal are shared services rather than a backlog. That is why a pre-launch product can credibly offer this much governance.

The catalogue references, it does not hoard

Artefacts, notebook documents and figures live once in shared storage and are referenced by the catalogue, which keeps the catalogue lean and the storage policy consistent instead of accumulating the blob sprawl standalone repositories tend towards.

Honest labelling as a product decision

Every AI capability, surface and trust control on the public site carries a shipped, in-progress or roadmap label, and the page counts are derived from that data so the product cannot overstate itself. It is a deliberate constraint, and it is why this document names its own gaps.

12 Honest answers to hard questions

1 of 2

You are pre-launch. Why would we bet on this?

Weigh that seriously. There are no customers, no revenue and no traction to cite, and this document deliberately contains none. What there is: a shipped catalogue, processing and research model with a working console, governance inherited from a platform that already runs other products, a free tier complete enough to test the workflow properly, and a public changelog you can hold us to. Start on Free, model your real holding, and decide from evidence.

The identifier on a release is not minted by a registrar

Also correct. The release workflow, the lifecycle and the identifier field are shipped, and minting has been built against a registrar sandbox, but production credentials are not in place, so today the identifier is supplied at publish time rather than registered automatically. Automated registration is on the roadmap and is a small piece of work next to the workflow around it.

You have no certifications

None. The control environment is modelled on ISO 27001 and designed against SOC 2 criteria, but no independent auditor has confirmed that yet. SOC 2 Type I observation is targeted to start in the first quarter of 2027 with Type II in the third, and ISO 27001 certification in the second quarter of 2027. If a current certificate is a hard procurement gate, we will not pass it today.

Your pipelines do not actually run the compute

Correct, and it is the most important limitation in the product. What ships is the record spine: definitions, runs, steps, validated transitions, parameters, metrics and lineage. A run advances through its states and records what an external worker reports. The managed engine that executes step code and notebook kernels in-platform is committed roadmap. If you need in-platform execution today, this is not yet the product for it.

Half your AI story is roadmap

It is, and it says so item by item. Shipped: the page-aware assistant, per-workspace metering, the AI-assist mark and results ledger on analyses, 42 agent tools with two guided workflows, faceted discovery and the lineage chain. Roadmap: plain-language search, anomaly detection and the three decision-engine judgements. We would rather you buy the shipped half than be surprised by the other one.

12 Honest answers to hard questions

2 of 2

We already have an institutional repository

Keep it. A repository archives finished outputs; it does not catalogue live sources, run processing, host reproducible analysis or capture lineage. CosmicIntersection is the active layer that produces what you deposit, and exporting or mirroring a release to your repository for long-term preservation is the intended pattern rather than a competing one.

Will we be locked in?

The mitigations are real but partial. Everything is reachable through one documented API rather than through screens only, sources are yours and connect to archives you already use, and releases are designed as standards-compliant, exportable packages. The product is not open source, and we do not claim it is. Exportable formats and clear APIs are the commitment we can actually make.

Is the free tier crippled to force an upgrade?

It is limited, not crippled. Free carries the whole product surface, including pipelines, notebooks and governed sharing, with no watermarking on exports. The deliberate exclusions are citable releases, scheduled pipeline triggers and production identity enforcement. It is sized to prove the workflow in the first sessions, not to run a research programme indefinitely.

Our data is enormous and must not move

Two honest answers. The catalogue references artefacts rather than copying them, so it does not force a second copy of your archive. Co-located compute next to petabyte-scale archives is roadmap rather than shipped. For facilities that cannot send data outside the network at all, self-hosted and air-gapped deployment is an Enterprise option.

Who maintains this if the team is small?

A fair question for any young product. The answer is leverage: identity, billing, files, notifications, tagging, conversations, audit, translation and accessibility are shared platform services used by other products in the same portfolio, so this team builds the science layer rather than rebuilding a software-as-a-service backbone. It is also why the governance is more complete than a product at this stage would usually manage.

Where it sits against the alternatives

VS CITIZEN-SCIENCE AND CROWDSOURCING PLATFORMS

Platforms such as Zooniverse are unmatched at public classification, and we have no volunteer crowd and never will. The complementary move is to ingest their labelled output into a governed workspace where processing, reproducible analysis and citable releases live together.

VS OPEN REPOSITORIES AND DOI ARCHIVES

Zenodo, Dataverse and similar repositories are free, durable and backed by preservation mandates we cannot match. They archive a finished file. They do not catalogue live sources, run pipelines, host reproducible analysis or capture lineage. Use us for the active lifecycle and mirror the release to them for preservation.

VS GENERAL-PURPOSE LAKEHOUSE AND DATA PLATFORMS

These are far ahead of us on raw compute scale and machine-learning tooling, and we will not pretend otherwise. What they do not have is any understanding of observations, instruments or celestial coordinates, any citable release layer, or predictable flat pricing for a research lab. Integrate them for petabyte workloads and keep the catalogue and lineage here.

Where it sits against the alternatives

VS ACADEMIC SCIENCE GATEWAYS AND HOSTED NOTEBOOK SERVICES

Gateways offering free institutional compute and multi-petabyte public holdings beat us on both counts today, and we offer no free academic virtual machines. What they typically lack is workspace-scoped roles, audit, billing, enforced single sign-on, versioned pipeline orchestration and a citable release workflow, which is precisely the governed layer we provide.

VS MISSION ARCHIVES AND OBSERVATORY PORTALS

An authoritative mission archive is authoritative, and we are not trying to replace one. We are the workspace that consumes it, alongside your private and instrument data, and turns it into analysis-ready datasets, reproducible notebooks and citable releases. Mission-specific query tooling is theirs; the lifecycle around it is ours.

VS AN IN-HOUSE PLATFORM BUILT BY YOUR OWN ENGINEERS

A good engineer can build a catalogue, and many have. What is rarely built alongside it is roles, audit, quotas, tenant isolation, single sign-on, billing and a release workflow, and what is never budgeted is the time to maintain all of it. The honest trade is that you give up total control of the data model in exchange for not owning the backbone.

START TODAY

Model your real holding on the free tier and judge it from evidence

CosmicIntersection is pre-launch and in waitlist mode ahead of a 2026 launch. The fastest way to evaluate it is to catalogue a real dataset, describe its schema, and see whether the trail it keeps is the one you have been missing.

- Join the waitlist and tell us which archives and instruments you work with
- Book a walkthrough tailored to your holding, your processing and your governance obligations
- Connect one source on the free tier, catalogue a dataset and describe its schema fields
- Run a pipeline, record an analysis, package it as a release, then follow the lineage back to the source

JOIN THE WAITLIST

cosmicintersection.com/waitlist

BOOK A DEMO

cosmicintersection.com/contact

BUSINESS AND ENTERPRISE

sales@cosmicintersection.com

RESPONSIBLE DISCLOSURE

security@cosmicintersection.com

EVERYTHING ELSE

contact@cosmicintersection.com



CosmicIntersection

CosmicIntersection is pre-launch and in waitlist mode. Capabilities marked roadmap or in progress are not available today. Prices are the planned launch configuration, are quoted before local taxes, and vary by regional band. Proprietary software of Algoshred Technologies Private Limited, part of the Burdenoff group.