Physical AGI (PAGI) Network

Physical AGI Foundation

Litepaper v1.0 | October 2026

Abstract

Real-world data for physical AI is bought as an input. A buyer acquires recordings, annotation labor, or compute time, and carries the risk that what is delivered does not satisfy the requirement it was acquired for. Defects are difficult to detect at the point of purchase and often surface only after training or in deployment. Correctness is therefore assumed on the basis of a supplier's reputation rather than established by evidence, and the cost of the resulting rework is absorbed as a normal expense of building these systems.

PAGI is a decentralized network in which the definition of an acceptable result is fixed before work begins, the work is carried out by independent *agents and operators*, and the evidence that the definition was met travels with the output. A buyer receives a result that can be checked trustlessly and an operator is paid for completion of work following the definition, regardless of the intrinsic quality of asset. The network ensures the validity, integrity, and usefulness of sovereign actors in this system with financial stakes and incentives. A buyer receives a result that can be checked. An operator is paid for work that can be shown to be correct, including work whose correct outcome is that an asset should be rejected.

1. The problem

Physical AI systems learn from recorded experience of the world. A robot can perform a task because that task, or something close enough to it, was captured and turned into training data. Data is the primary input to these systems and the limit on what they can be taught to do.

Teams acquire that data through in-house collection operations, licensing agreements, and annotation vendors. They acquire the compute to process it through cloud providers and GPU markets. Both are functioning markets with established suppliers and understood pricing. A team can specify the hardware it needs and the volume of footage it needs, and receive both.

Neither market can establish whether the material satisfies the requirement it was bought for. A compute receipt establishes that a process ran. A data delivery establishes that files were transferred.

Neither one establishes that hands remain visible through the moment of contact, that a demonstration contains the task it is labeled with, that timestamps across sensors are aligned, or that a recording was captured under the conditions the buyer specified.

Those properties determine whether the data is usable, and they are the properties hardest to check on receipt. A recording can parse correctly and be worthless. An annotation can be plausible and be wrong. The failures surface later, in training runs that underperform for reasons that take weeks to isolate, or after deployment, when a system behaves incorrectly in a warehouse or a home because it learned from a defect that nobody detected at purchase.

Because correctness cannot be established at the point of purchase, it is assumed. The assumption rests on the reputation of the supplier, which makes the quality of a dataset a function of which vendor a team decided to trust. Rework is treated as a cost of doing business rather than as a failed transaction. That arrangement holds while data volumes are modest and the systems trained on them stay contained. It becomes harder to sustain as physical systems move into settings where their errors have physical consequences.

What is required is a way to purchase a defined result rather than an input. The definition has to be fixed before the work begins and stated precisely enough that a result can be checked against it. The evidence of that check has to travel with the output, so that a downstream user can confirm the result without repeating the work. An operator that carries out the work correctly has to be paid for doing so, including when the correct outcome is a rejection.

PAGI is a network built on that arrangement. The sections that follow describe what a team has to build today in the absence of it, how work is defined and allocated, why a buyer would use the network, who does the work and how they are paid, how correctness is established, and how the money behind the work is funded.

2. What a team has to build today

The absence described above is not a gap in the market. Teams fill it, and the way they fill it is by building a processing operation of their own.

That operation has a predictable set of parts. Someone has to write acceptance criteria precise enough that two reviewers reach the same verdict on the same clip. Someone has to recruit and qualify the people who do the work, then keep them, because a trained reviewer who leaves can take the calibration with them. Someone has to build quality assurance that catches what the criteria missed, since the first version of any spec fails on edge cases nobody anticipated. Someone has to run calibration against known answers, or the scores drift without anyone noticing. Someone has to

handle disputes, because reviewers and buyers disagree about what passed. And someone has to pay everyone on time, in the right currencies, or the supply leaves. None of this work is specific to the product the team is building, and all of it sits between the team and a usable dataset.

COMPONENT	BUILD IT YOURSELF	USE THE NETWORK
ACCEPTANCE CRITERIA	Build and maintain	Supplied
OPERATOR POOL AND QUAILIFICATION	Build and maintain	Supplied
CALIBRATION AGAINST KNOWN ANSWERS	Build and maintain	Supplied
QUALITY ASSURANCE	Build and maintain	Supplied
DISPUTE HANDLING	Build and maintain	Supplied
CROSS-BORDER PAYMENT	Build and maintain	Supplied
	The build repeats for each new specification	The buyer still states what it needs, precisely enough to be checkable.

Figure 1: What a team builds, and what the network supplies

The cost shows up in three ways that are easy to underestimate from the outside.

Setup is the expensive part, and it is a cost that repeats itself. A new specification is not a parameter change, but it means new recruiting, new briefing, new scripts, new qualification, and a new round of calibration before the first accepted unit arrives. A team that has built a pipeline for one kind of capture has not built one for the next kind.

Volume does not predict usable delivery. Sign-ups, active participants, qualified participants, and accepted output are four different numbers, and the gap between them widens as volume grows. Invalid submissions rise with scale.

Quality problems surface late. When scores, reviewers, and buyers disagree, the disagreement is usually discovered downstream, after the data has been accepted and sometimes after it has been trained on. Missing calibration does not announce itself. It produces a dataset that looks complete and performs badly.

The clearest evidence that this is a real constraint rather than an inconvenience comes from the best-resourced buyers. Figure, building its Index program, reported that it had tried buying data and that vendors could not meet the throughput, diversity, and quality bar its systems required, so it built

the pipeline itself. That pipeline now runs 44,000 weekly contributors across 108 countries, with $15M paid to contributors.

That is the correct response for a company with the capital and the engineering depth to do it. It is not available to most teams building physical AI systems, and it is not a good use of the capital of those that could. A team that builds its own processing operation has solved a problem that is not specific to its product, in a way that no other team can reuse, and that it must maintain for as long as it needs data.

The question this leaves is whether the operation itself can be bought, rather than the inputs to it.

3. The PAGI Network

PAGI coordinates data work between buyers who need a defined result and operators who can produce it. Three ideas are the core tenets of the design.

The work is defined before it is done

Work is organized into **Verifiable Data Networks**. A data network exists for a kind of work, operates under a written and version-numbered policy, and admits participants who qualify against that policy. The policy states what counts as an acceptable result, how a result is checked, and what happens when a result is disputed. It is published before work begins and it is versioned, so that a result can always be traced to the exact rules it was produced under.

Each network is run by an **Operator**, the identifiable person or organization accountable for it. The Operator writes and maintains the policy, admits participants, administers commercial terms, and is responsible for delivery, billing, and collection. Workers, verifiers, and payment providers are Operators of their own functions. Keys or machines under common economic control count as one Operator group, which is what stops a single party from splitting itself to occupy several independent seats.

Inside a data network, work is divided into **service classes**. A service class is the unit that is measured, funded, and paid for. One network running video work might carry qualification as one service class and annotation as another, each with its own acceptance criteria and its own price. The separation lets a buyer purchase one stage of a workflow rather than committing to all of it.

A network is opened when demand for it appears, by an Operator who takes responsibility for running it. It scales with demand and closes when the demand ends. Policy and participant histories stay available after a network closes, so work done under it remains verifiable afterwards.

Who can join, and how work is assigned

Admission criteria are published before anyone applies. In a network with open participation, an Operator does not refuse an applicant who posts the required deposit and meets those published criteria. A rejection or removal has to cite the policy rule it rests on, and can be appealed.

The first networks assign work by round robin. As participants build a performance record, a network can move to assignment weighted by reputation or by deposit, which routes work toward contributors who have demonstrated they can do it. What a deposit never does is change how the resulting work is judged or what it is worth. Each policy discloses which functions its Operator runs directly and how that is expected to change.

Buyers are customers, not participants. They post no deposit and take on no obligations beyond their commercial terms.

The result has its own evidence

When work is completed, the record of the work travels with the output. A buyer receives the result, the policy version it was produced under, and the record of the checks that were performed against it.

This is the part that distinguishes a defined result from an input. A buyer who receives a dataset with no evidence has to either trust the supplier or repeat the verification. A buyer who receives the evidence can confirm the result directly, and so can anyone the data is passed to afterwards.

Settlement data, evidence records, and work outputs become available progressively on chain or on decentralized storage, so that a party other than the Operator can verify what was settled.

How a result is checked

Checking happens three ways, and no single one is relied on alone.

A share of completed work is sampled at random and re-examined independently. Items with known answers are seeded into the work queue without being identified as such, so that an operator processing them cannot tell them apart from live work, and an operator's accuracy against those items is measured continuously rather than at onboarding. And where the work runs as software, the operator can supply attested computation, which is evidence that the specified program ran under the required conditions.

A sample establishes something about a batch and nothing about any particular uninspected item. A reviewer who inspects 100 records drawn at random from 1,000 of equal value and finds 96 valid has evidence suggesting around 960 are valid. That is not proof that exactly 960 passed, it does not

identify which of the remainder failed, and it does not justify treating the estimate as certain. The verification plan therefore settles on a supported lower figure and records what the sample can and cannot show.

The sampling rate, the proportion of seeded items, and the evidence each criterion requires are set per service class and published in its policy.

How the sample is drawn carries as much weight as how large it is. The plan divides work into non-overlapping groups by difficulty, privacy tier, operator, and reported verdict, then draws from each so the sample reflects the characteristics of the full set. Where harder work is inspected more often, each observation is weighted by the share of the population it stands for, so that closer scrutiny of difficult cases does not make them appear more common than they are.

A sample also has to establish quality without exposing the value of the material it is drawn from. Confidential evidence is examined through an authorized access route, and the policy sets limits on what the verification output discloses, since repeated queries or narrow aggregates can reveal a holding even when the examination itself is protected.

Checking the work and checking the claim are separate jobs

Two kinds of verification run in the network and they answer different questions.

Service verification asks whether the output is correct. It runs inside the data network, against the policy, and it is what the buyer cares about.

Protocol verification weighs the relative value of work across networks and asks whether a reward claim from a data network is valid. It confirms that the work being paid for was ordered, performed, and funded. It runs independently of the data networks that produced the work.

Keeping these separate means that an operator cannot establish the correctness of its own output and also establish the validity of its own claim for payment. Protocol verification is itself specialized, with domain checks and financial checks carried out by different parties, so that someone competent to judge whether a video meets a specification is not thereby trusted to judge whether the money behind it was real. A financial verifier's vote cannot fill an empty domain seat.

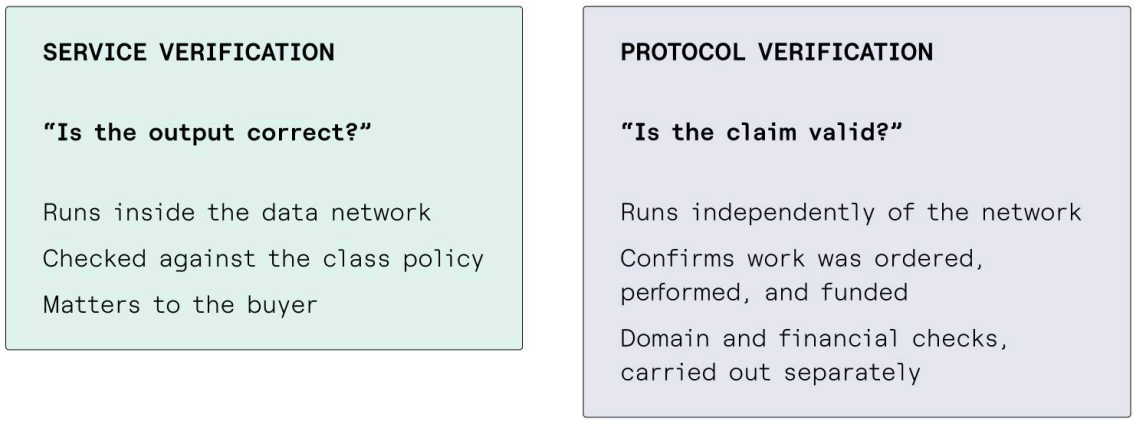


Figure 2: The two verifications

One unit of work from start to finish

The following traces a single unit through the network.

A buyer funds work in a service class and states the volume it needs. The service class policy is already published and version-numbered, so the acceptance criteria are fixed before anything is assigned. An operator qualified in that service class receives a unit and performs the work, reaching a verdict against the stated criteria. The output and the record of how it was produced are submitted together. A service verifier, drawn from a different party than the operator, checks the submission against the same policy version. The unit enters the sample pool, where it may be re-examined at random or found to be a seeded known-answer item. A protocol verifier confirms that the unit was ordered by a funded buyer and that the work recorded actually took place. Payment settles to the operator and the verifier against the accepted unit, and the buyer receives the result together with the policy version and the check record.

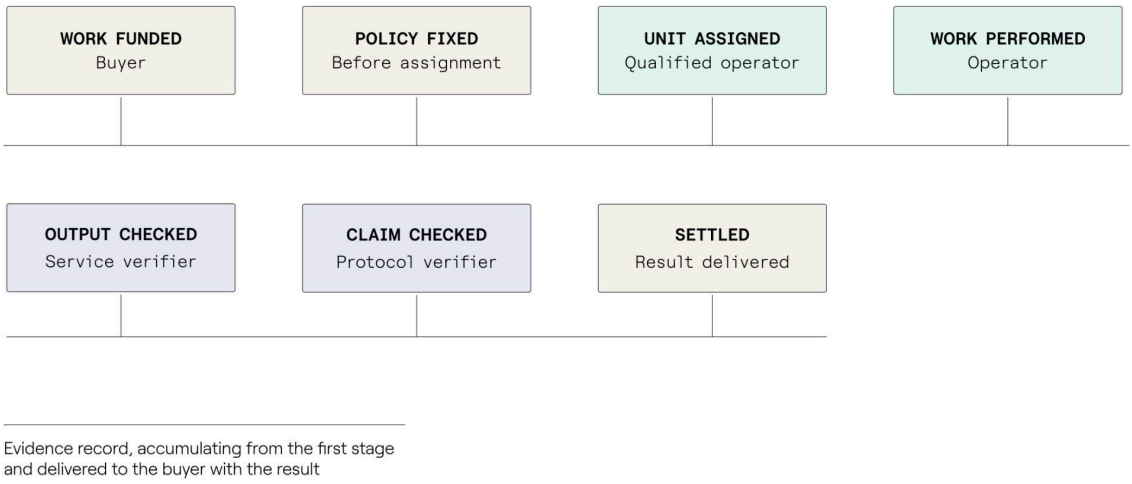


Figure 3: Lifecycle of one unit of work

What PAGI does not do

PAGI does not sell datasets. It coordinates the work of ingesting, processing, qualifying, annotating, and evaluating data, and it settles payment for that work. The data belongs to whoever supplied it, under whatever terms they supplied it.

4. Why a buyer would use it

A buyer evaluating PAGI is comparing it against two alternatives: building the operation described in section 2, or buying from a managed vendor. There are two reasons to prefer the network.

The pipeline is already built

The first and larger reason is that a buyer using PAGI does not construct a processing operation. The acceptance criteria exist and are published. The operator pool is qualified and the calibration runs against known answers. The dispute path is defined before the first dispute and the payment system handles operators across jurisdictions.

A team that wants qualified video next month can have work underway without recruiting anyone, writing a brief, or building a review process. A team that wants a different kind of work the month after that does not start over, because the setup cost sits in the network rather than in the buyer.

A cloud provider sells a team the compute to run a pipeline the team has to build. A managed vendor sells a team a finished delivery, with the pipeline staying inside the vendor where the buyer cannot inspect it or substitute parts of it. PAGI makes the pipeline itself purchasable, as defined services the buyer composes.

The practical consequences for a buyer are specific. Integration happens once and holds across different kinds of work. A provider that stops performing can be replaced without rebuilding the workflow around it, because the acceptance terms belong to the network rather than to the provider. Capacity can expand without a new commercial negotiation. And the record of what was checked is consistent across suppliers, so a buyer running work with four operators receives one kind of evidence rather than four.

The cost structure is different

The second reason is cost. The saving is in how the supply is organized, not in the rate a buyer manages to extract.

Centralized processing prices capacity that was provisioned in advance and must be paid for whether or not it is used. The buyer absorbs the cost of the provider's idle capacity through the rate. Distributed capacity is priced against its own alternative, which is sitting idle and earning nothing. An operator with spare capacity will accept work at a price a provider with a utilization target cannot offer.

The same holds on the labor side. A team running its own operation carries the cost of keeping a trained reviewer pool available between campaigns. A network distributes that cost across every buyer using the pool. These are differences in what is being paid for, and they persist rather than eroding under competitive pressure.

What a buyer still has to do

A buyer using PAGI still has to state what it needs precisely enough to be checkable. That is real work and the network does not remove it. What the network removes is everything between a clear specification and a verified result.

5. Who does the work and how they are paid

PAGI pays for work that someone ordered and that passed its checks.

Who earns

Workers perform the service units: qualifying footage against a specification, annotating it, running an evaluation, processing a file. They are paid per accepted unit, at the rate the service class specifies.

Service verifiers check the output against the policy. They are paid for review, and their own accuracy is measured against seeded known-answer items, so that a reviewer who approves everything does not earn the same as one who judges correctly.

Protocol verifiers check reward claims rather than output. They confirm that work was ordered, performed, and funded before payment is released. Domain and financial checks are carried out by different parties and compensated separately.

Network Operators open and run data networks. They write the policy, admit participants, and take responsibility for the network's delivery, billing, and collection. They earn operating fees from the work their network settles, and can receive development or adoption awards against stated milestones with a finite budget and an expiry.

ROLE	WHAT THEY DO	WHAT THEY ARE PAID FOR
WORKER	Perform service units	Each accepted unit, rejections included
SERVICE VERIFIER	Check output against the class policy	Each review, scored on seeded items
PROTOCOL VERIFIER Domain	Confirm the work was ordered and performed	Each claim checked
PROTOCOL VERIFIER Financial	Confirm the funding behind the claim	Each claim checked
NETWORK OPERATOR	Write the policy, run a data network	Operating fees, plus capped milestone awards

Figure 4: Who earns and what for

Correct rejections are paid

A worker who examines a file and correctly determines that it does not meet the specification has done the work the network needed. Paying only for accepted material would create pressure to pass marginal cases, since a rejection would mean unpaid effort. So valid rejections are paid at the same rate as valid acceptances. What the network buys from a worker is a correct decision about the data, and a correct decision to reject carries the same value as a correct decision to accept.

Verification is designed so that neither guessing nor blanket abstention pays. A worker is not rewarded for returning a verdict it cannot support, and a worker that abstains on every case does not earn either. Both accepted and rejected verdicts are sampled for audit.

Payment follows work, not holdings

The network pays against work that someone ordered and that passed its checks. Holding tokens earns no share of a network's commercial margin, governance rights earn no staking income, and an idle deposit earns no fee.

Participants post deposits sized to the work they take on, which can be lost if they fail their obligations. The deposit secures performance. A larger deposit does not increase the quantity of work credited to its holder, and it does not raise the quality score that work receives.

New issuance is capped per period and is directed at work in networks that are still establishing demand. As a network's buyers cover its costs, that support tapers. The purpose of issuance is to carry a network to the point where the demand pays for it.

What an operator gets that is not money

Three things accumulate for an operator working through the network, which reduces the cost of the next engagement.

Work arrives under a specification that already exists, which removes the negotiation and scoping that normally precedes a contract. Payment terms are defined before the work starts, and settlement follows acceptance instead of a client's internal approval cycle. And a record of accepted work builds up that every buyer on the network can see, so an operator that performs well becomes easier to find and qualifies for more work without selling into each buyer separately.

6. When a check fails

Section 3 describes how a result is checked. This section covers what happens when a check does not pass, because how a network behaves under disagreement says more than how it behaves when everything works.

Disagreement is expected and handled

Some acceptance criteria can be checked mechanically. Others require judgment about meaning, relevance, or whether supporting evidence is adequate. The service class policy states which kind each criterion is, so neither the operator nor the buyer discovers after the fact that a judgment call was treated as a measurement.

Where judgment is involved, more than one reviewer scores the work against the published rubric. Reviewers are assigned at random, commit their scores before seeing anyone else's, and are subject to limits on common ownership, so that a group cannot quietly coordinate. Eligibility, the number of reviewers required, how their scores combine, and the acceptance threshold are all fixed before the work begins.

Two things are deliberately excluded from how much a reviewer's opinion counts: the size of their deposit, and whether they agree with the current majority. Where a policy weights reviewers at all, the weight comes from measured performance on calibration work, and the rule has to be published before it is applied.

A minority view can be right

If reviewers disagree materially, if too few are available, or if evidence is missing, the result does not pass by default. The policy states what happens instead. The work is corrected, referred to adjudication, or closed with only the separately completed parts accepted.

A reviewer who dissents from the majority, with support for their position, has still performed a valid review and is paid for it. Their calibration record reflects how they performed against cases that were later resolved independently, so being in the minority and being right improves standing.

Missing a deadline and breaking a rule are different failures

A participant who misses an assignment, submits late, or fails to vote forfeits the reward for that piece of work. The deposit stays intact.

Deposits are at risk only for a defined and narrow set of breaches: fabricated evidence, signed records that contradict each other, fabricated output, copying another reviewer's verdict, submitting a review without performing the required inspection, a disqualifying interest that was concealed, or demonstrated manipulation of a review. Each has to be declared in advance, evidenced, and subject to appeal. A penalty cannot be invented after the fact for conduct that was not named before the work started. Honest disagreement, ordinary model error, and correct rejection of unsuitable work carry no penalty.

Where a deposit is forfeited, part goes to the party that brought the evidence, as payment for that work, and part to the protocol treasury.

Two further constraints keep the incentives clean. A reviewer cannot authorize additional paid review for themselves, which prevents anyone manufacturing their own workload. And an adjudicator's fee cannot increase with the size of the penalty they impose, which removes the reason to impose a larger one.

Challenges

A challenge names the specific record, the duty alleged to have been breached, who is affected, the evidence, and the correction being sought. It suspends only the contribution it concerns, leaving the rest of the batch to settle.

The adjudicator distinguishes between a service error, a procedural breach, and insufficient evidence to decide, because those three carry different consequences. Deadlines, the appeal route, any award to a successful challenger, and where a penalty goes are all set by rules published before the work.

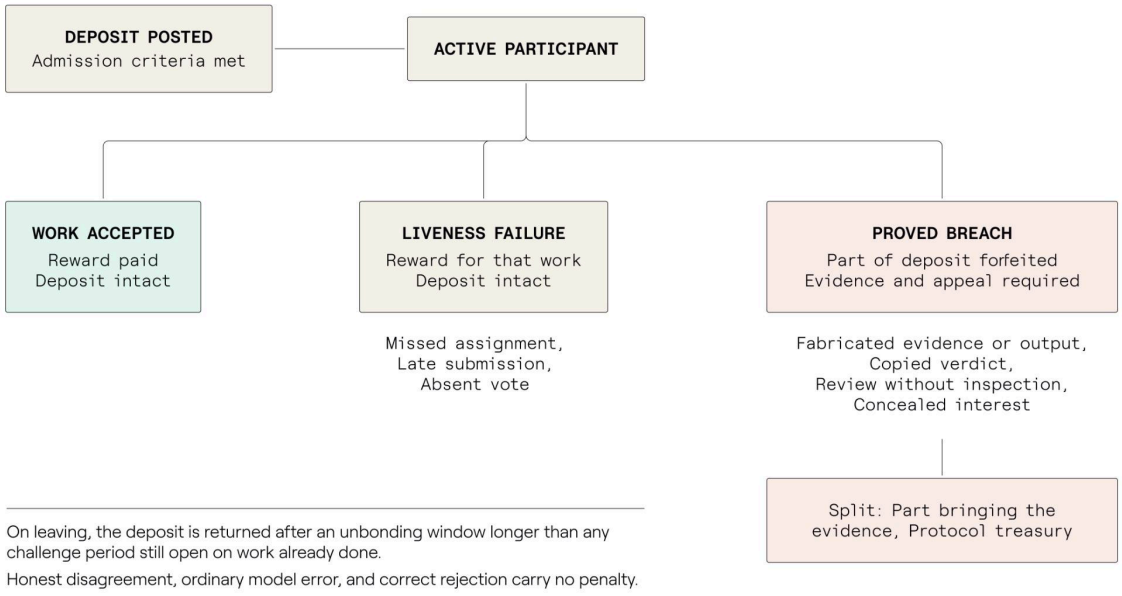


Figure 5: Stake lifecycle. A liveness failure loses only the reward.
Only a proved malicious act or reward gaming causes a slash.

7. How the money holds together

A network can look healthy while paying out against work nobody funded. The checks in section 6 establish that the work was done correctly. Establishing that the money behind it was real takes separate machinery, which this section describes.

A worker knows the terms before starting

Before a participant accepts a unit, the terms are fixed and recorded: what one unit is, the criteria it will be judged against, the maximum payable reward, the deposit required, the challenge period, and when settlement happens. The record separates the part of that reward already backed from the part that depends on allocation still to be decided, so a participant can see which is which before committing.

Once work is accepted, the Operator cannot change the rule it will be paid under, and cannot change it partway through a settlement period. Where a network commits to a fixed reward rate for a period, it has to commit the collateral behind that rate in advance.

Buyers pay through one of several arrangements, including a funded balance in advance and approved accounts billed after delivery. Where a buyer is billed after delivery, the network or its funding provider supplies the contribution covering that work before it is dispatched, so the

obligation to participants does not depend on the collection succeeding. If a buyer later fails to pay, the unpaid invoice stays a receivable of the Operator. It cannot be recorded as available inventory for reward purposes, it cannot reduce an award already final, and it cannot be recovered by claiming a subsidy that was not disclosed.

What counts as demand

The network measures verified collections attributable to delivered service. That figure is narrower than a customer account balance and narrower than revenue an operator might report in its own accounts.

A paid-service record ties together the buyer's authorisation, the commercial policy in force, the service units delivered, the acceptance record, the payment source and amount, how it was attributed, whether it was collected, any refunds, and the relationship between the parties.

The measure excludes:

- Money sitting in an account that has not bought anything yet
- Invoices that were issued but not paid
- The same payment counted against more than one piece of work
- Amounts that were later returned
- Transfers presented as independent demand when they come from parties under common control
- Funding passed from one service class to a neighbouring class inside the same network, which is internal funding and not an additional outside payment

A payment counts as independent demand only to the extent its economic origin and the service actually delivered support it.

Attribution ceiling

One payment can fund several pieces of work, but the parts attributed to each cannot add up to more than the amount actually collected. The controls that prevent double-counting cover public, confidential, and offchain payment routes together, closing the gap that opens when each route is policed in isolation.

A buyer can purchase a bundle, such as qualifying a batch of clips and then annotating the ones that pass. Part of that payment is attributed to the qualification stage only where the buyer expressly bought that deliverable and the attribution rule was fixed before the work. An internal transfer cannot create that attribution afterwards.

Refunds, reversals, and corrections update the record going forward without reaching back into unrelated settled awards.

Who gets what from collected cash

Collected cash meets payment costs, tax and refund obligations, external service expenses, and settlement to the participants who did the work. What remains is the operating margin of the party that holds the customer contract, and holding tokens creates no claim on it.

What issuance is for

New issuance is capped and directed at networks still establishing demand. As a network's buyers cover its costs, that support reduces.

The onboarding path makes this explicit. A new network or a materially new service class starts under operator or sponsor funding, moves through probation and expansion with progressively larger ceilings on public support, and reaches normal operation only after meeting stated conditions on review samples, error rates, settlement timeliness, deposits, and demonstrated commercial use.

Elapsed time alone does not qualify a network. A history of operation carries through a rename, a policy update, or a migration, so a network cannot restart its trial support by re-registering under a new name.

8. How the network is governed

Protocol-level parameters, contract upgrades, and emergency controls start under a multisig stewarded by the foundation. Control moves to token-holder voting as the network matures and demonstrates it can attract capable Operators without central coordination. The quorum, thresholds, and timing for that transition are still being defined.

An Operator controls only its own network policy, inside the bounds that protocol governance sets.