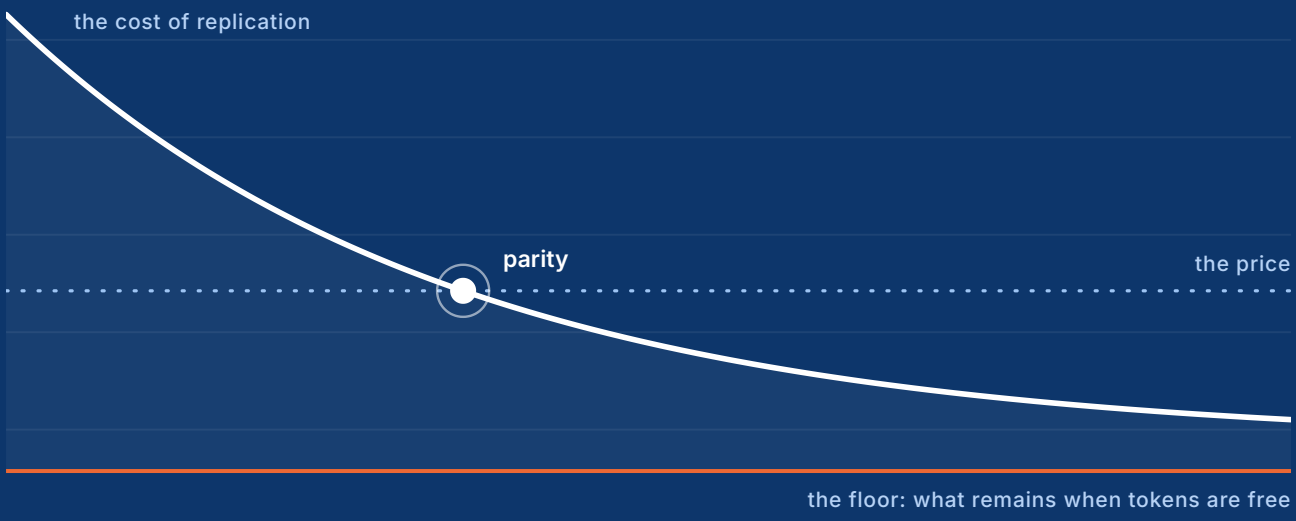
A FRAMEWORK · VERSION 0.1, FOR COMMENT

Open Replication Cost

A framework for peace between creation and reverse engineering



Illustrated edition · 13 November 2026

With a worked example: COTBase.com

Contents

SECTIONS

	Summary	3
1	Where the thesis holds, and where it needs sharpening	5
2	Foundations in economics and law	6
3	What a subscription sells: six layers	8
4	The cost model	10
5	Three parity prices and the decision rule	12
6	Stability: the invariants, the Index and the floor	15
7	The checker	17
8	The peace	20
9	Gaming and failure modes	22
10	Worked example: COTBase.com	23
11	Governance and versioning	27
12	Limits and open questions	28
13	Roadmap	29
A	The reference model	30
B	Glossary	31
	Sources	32

FIGURES

1	The decision rule	3
2	Four public rebuilds	5
3	Fifty years of precedent	7
4	Human and AI rebuild cost	7
5	Six layers	8
6	Fidelity levels	9
7	Anatomy of the cost	10
8	Observability grades	12
9	Three replicators	13
10	Three clocks	15
11	Measuring the Index	16
12	Parity runway	17
13	The checker protocol	18
14	A buyer card	19
15	The peace mechanism	20
16	Commitments on both sides	21
17	Cost composition	25
18	Replication multiples	26
19	Shared replica by group size	26
20	Roadmap	29

TABLES

1	Five findings	4
2	The public rebuild cases	5
3	Foundations in economics	6
4	The legal ground	8
5	Data classes	9
6	Notation	11
7	The decision rule	14
8	Invariants, Index and score	15
9	Evidence grades	18
10	What each reader gets	18
11	Peace instruments	21
12	Attacks and safeguards	22
13	Example: published facts	23
14	Example: layer map	23
15	Example: assumptions	24
16	Example: the arithmetic	24
17	Example: results	25
18	Governance rules	27
19	The five steps	29
20	Reference model outputs	30

OVERVIEW

Summary

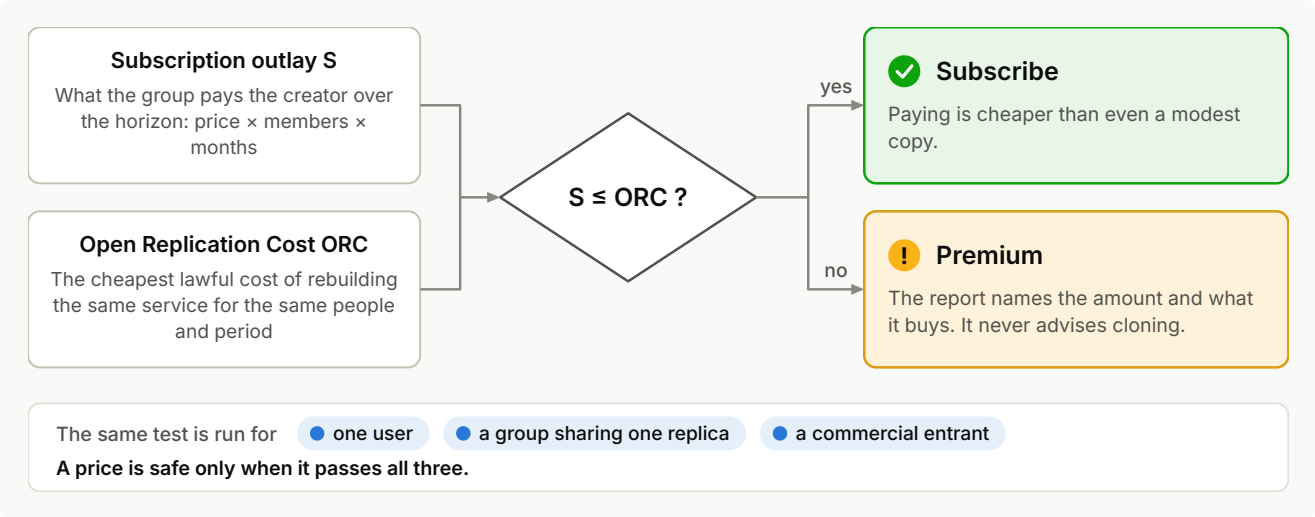
Publishing the cost of rebuilding a service, and comparing it with its price, can settle most of the conflict between creators and reverse engineers.

The idea

This document turns that idea into a standard called Open Replication Cost (ORC). It keeps the original rule, subscribe when the subscription is cheaper, and adds four things the rule needs to hold up.

FIGURE 1

The rule: subscribe when the subscription costs no more than replication



The checker advises in one direction only. Above replication cost it reports the premium and what it buys.

FOUR THINGS THE RULE NEEDS

1 Cost the whole service, not the code

A subscription sells six layers. Tokens buy one of them, and only in part.

2 Test three replicators, not one

A single user, a group sharing a replica and a commercial entrant face costs that differ by orders of magnitude. A price is safe only if it passes all three.

3 Give advice in one direction only

Below replication cost the checker says subscribe. Above it, the checker reports the premium and what it buys, and never advises cloning.

4 Separate what is stable from what decays

Functional size and layer classes are fixed by the standard. Unit costs come from a dated public index. Every score also states its floor: the cost that remains when tokens are free.

TABLE 1 **Five findings**

FINDING	EVIDENCE
The test already exists in regulation	US rail rates for captive shippers are capped at the cost of a hypothetical efficient replica railroad.[1] A US court capped a patent royalty at the cost of the non-infringing alternative.[2]
Replication cost is now measurable	Most of the Next.js API was rebuilt for about $1,100 in tokens,[3] and a C compiler for under $20,000.[4] A public benchmark now grades black-box rebuilds.[5]
Today tokens are the small part	In the worked example they are 13% to 20% of the cost of a personal copy. Human attention is the rest (Figure 17).
COTBase Premium passes clearly today	At $14 a month, a personal copy of the core costs 5 to 10 times the subscription over three years. A shared replica matches the price at 35 to 58 members (Figure 18 and Figure 19).
For pure function the protection is temporary	At full automation the personal copy costs about two thirds of the subscription. What then holds the price is the living service, the content and the trust (Figure 12).

THE WORKED EXAMPLE IN FOUR NUMBERS

Replication multiple
personal copy of the Premium core, 36 months

5.3× to 10.2×

What a copy costs, as a multiple of the $504 subscription

Token share
of that replication cost

13% to 20%

The rest is human oversight and attention

Group parity
members sharing one living replica

35 to 58

The group size at which each member pays the plan's $14 a month

At full automation
the same personal copy

0.66×

About two thirds of the subscription: $335 against $504

The peace

The peace is contestability without the contest. Once the cost is public, the threat of replication disciplines the price, and nobody needs to spend tokens rebuilding what already exists.

Status

This is version 0.1. The public index it depends on does not exist yet, and the worked example uses bootstrap unit costs.

SECTION 1

Where the thesis holds, and where it needs sharpening

The premise is right for one layer of a product, its software function, and incomplete for everything else a subscription sells.

The thesis

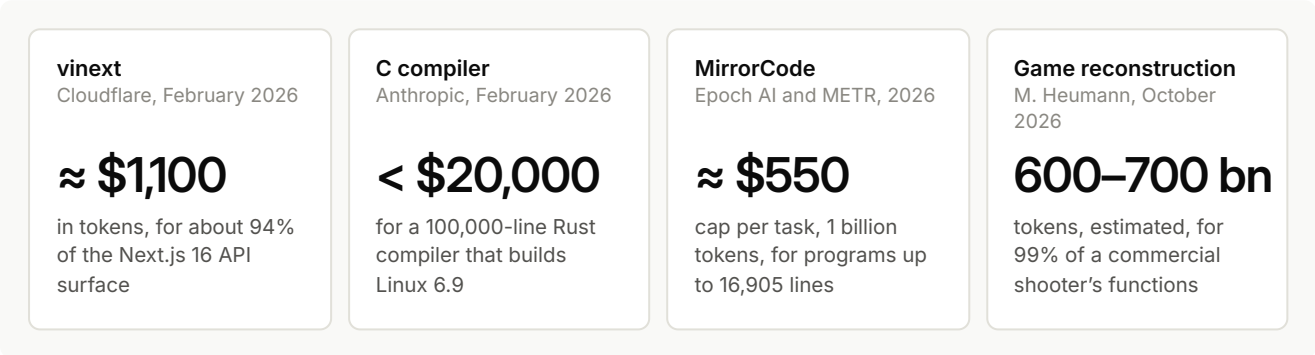
The starting thesis is that, with the AI of today and tomorrow, reverse engineering anything becomes a question of token cost. If so, the cost can be measured openly, set beside the subscription price, and turned into advice: where subscribing is cheaper, subscribe.

What holds

Function is being rebuilt in public at costs that can be read off an invoice.

FIGURE 2

Four public rebuilds in 2026: the cost of function can be read off an invoice



Reported costs, as published. Table 2 gives the effort behind each figure and the conditions that made it cheap. Sources: Cloudflare;[3] Anthropic;[4] Epoch AI;[5] the Heumann account.[6]

TABLE 2 **What was rebuilt, what it cost, and what made it cheap**

CASE	WHAT WAS REBUILT	REPORTED COST	WHAT MADE IT CHEAP
vinext, Cloudflare, February 2026[3, 7]	About 94% of the Next.js 16 API surface	About $1,100 in tokens; one engineer; under a week; over 800 sessions	A documented API, the Next.js test suite as specification, Vite underneath
C compiler, Anthropic, February 2026[4]	A 100,000-line Rust compiler that builds Linux 6.9	Just under $20,000; 16 agents; nearly 2,000 sessions; two weeks	Compiler test suites, and GCC as a known-good oracle
MirrorCode, Epoch AI and METR, 2026[5]	Command-line programs up to 16,905 lines, from execute-only access	A cap of 1 billion tokens per task, about $550	The original program as an exact oracle. A 61,461-line target stayed unsolved
Game reconstruction, M. Heumann, October 2026[6]	A commercial shooter from its binary: 99% of functions, 83% byte-exact	About three months; an estimated 600 to 700 billion tokens	A disassembler and a byte-for-byte comparison harness

What needs sharpening

1 **Every cheap case had a checkable specification.** Cloudflare names three conditions: a documented API, a comprehensive test suite and a solid build tool.[3] The compiler's author says the verifier must be 'nearly perfect'.[4] Epoch says success depends on a detailed, checkable specification.[5] A

closed service publishes none of these, so discovery and verification set the cost, not code generation.

2. **Tokens are not the whole bill.** vinext also took an engineer's week. Most of the compiler author's effort went into the test environment.
3. **The results are not at parity.** Cloudflare calls vinext experimental. The compiler emits slower code than GCC with optimisation switched off. Fidelity has to be part of any honest cost.
4. **Code is one of six layers.** Data, operations, future improvements, content and trust are costed differently, or cannot be bought at all (Section 3).
5. **Who replicates matters as much as what.** A Harvard study prices rebuilding the most widely used open source once at $4.15 billion, and at $8.8 trillion if every firm rebuilt it for itself.[8]

Why now

In October 2026 Eric S. Raymond argued that shipped binaries keep their secrets only until someone points a model at them.[6] The Dow Jones U.S. Software Index fell about 36% between October 2025 and April 2026 on fears of AI substitution.[9] The law on AI-assisted rebuilding is untested (Section 2). The tension is real, and it is unpriced.

SECTION 2

Foundations in economics and law

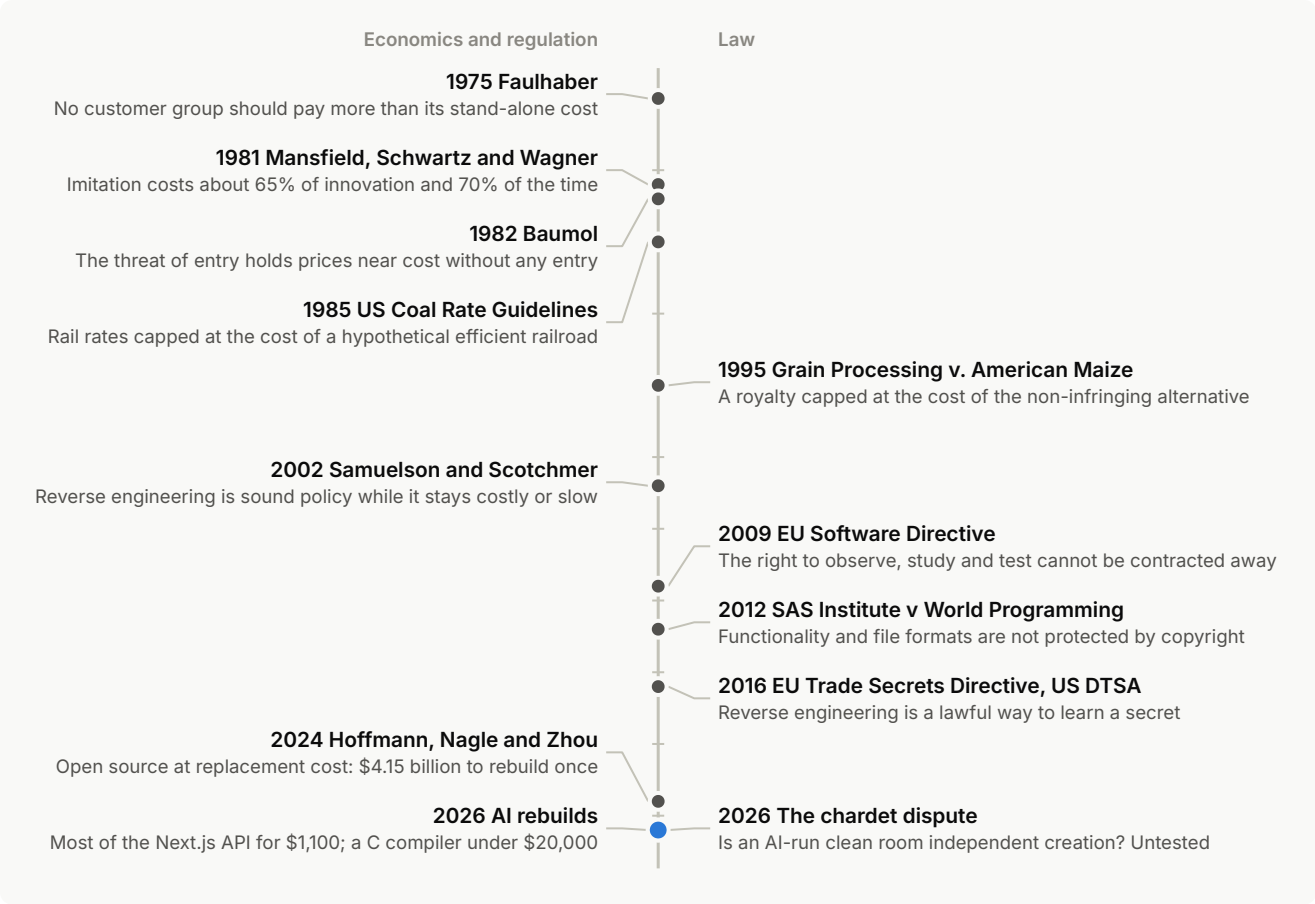
Regulators, courts and economists already use replication cost as a price ceiling. Nobody has yet applied it to software in the open.

TABLE 3 **What ORC takes from each source**

SOURCE	RESULT	WHAT ORC TAKES FROM IT
Samuelson and Scotchmer, 2002[10]	Permitting reverse engineering is economically sound while it is costly or slow enough for innovators to recoup their investment	Cost is the hinge. When it collapses, the settlement has to be renegotiated
Mansfield, Schwartz and Wagner, 1981[11, 12]	Across 48 product innovations, imitation cost about 65% of innovation cost and took about 70% of the time	The historical ratio. The cases in Table 2 put AI rebuilding of well-specified software near 1% of a human rebuild
Faulhaber, 1975, restated 2005,[13] **and the US rail stand-alone cost test, 1985**[1]	Prices are subsidy-free when no group of customers pays more than the cost of serving that group alone. A captive shipper may design a hypothetical efficient railroad, and its rate may not exceed what that railroad would need to charge	The test itself, and a warning: the rail version grew so elaborate that a court feared 'a full employment bill for economists'
Baumol, 1982[14]	Where entry is easy, the threat of entry holds prices near cost without entry taking place	The peace mechanism
Grain Processing v. American Maize, 1995[2]	A licensee would never pay a royalty above the extra cost of its non-infringing alternative	The parity licence
Hoffmann, Nagle and Zhou, 2024[8]	Software can be valued at replacement cost, using COCOMO II and wage data	The measurement precedent

FIGURE 3

Fifty years of precedent: replication cost as a ceiling, and reverse engineering as a lawful route



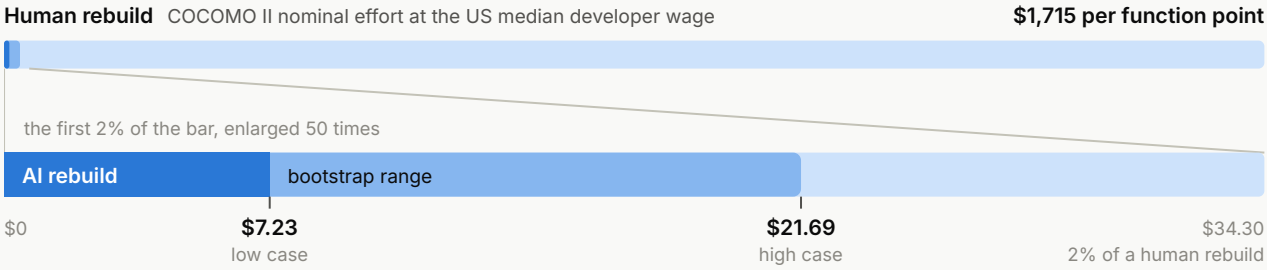
The axis is to scale. Left: economics and regulation. Right: law. Table 3 and Table 4 give the detail and the sources.

The ratio has moved

Mansfield and his co-authors found imitation at about two thirds of the innovator's cost. For software with a checkable specification, the ratio between an AI rebuild and a human rebuild is now of a different order (Figure 4).

FIGURE 4

An AI rebuild of well-specified software costs about 1% of a human rebuild



0.4% to 1.3% of a human rebuild, which costs 79 to 237 times as much

Cost per COSMIC function point. Human: COCOMO II nominal effort for a 100-point program at about 50 source lines per point, priced at the US median developer wage.[15, 16, 17] AI: the bootstrap unit costs of Section 10, tokens plus human oversight, which bracket the published cases. These are declared assumptions (grade C), not measurements.

TABLE 4 **The legal ground, briefly**

WHERE	WHAT THE SOURCES SAY
EU	A lawful user may observe, study and test a program to find its underlying ideas, and contract terms to the contrary are null and void.[18] Functionality, programming languages and data file formats are not protected by copyright.[19] Observing, studying or testing a lawfully acquired product is a lawful way to learn a trade secret, absent a valid duty not to.[20] Decompilation is allowed only for interoperability.
US	Federal trade-secret law excludes reverse engineering from improper means.[21] A licence clause that forbids it can still be enforced as a contract.[22]
Unsettled	Whether an AI-run clean room counts as independent creation when the model may have trained on the original, as in the chardet dispute.[23] In the US, code written by AI with little human contribution may not be copyrightable, so a cloner may be unable to protect the clone.[24]

This is a map, not legal advice. It needs a lawyer's review before anyone relies on it.

> **The route the standard prices**
> The standard prices one route only: rebuilding from lawful observation and public information. It excludes copied code, content or protected data, decompilation outside the interoperability exception, and anything that needs a breach of access terms.

SECTION 3

What a subscription sells: six layers

A subscription is a bundle of six things, and tokens buy only the first.

FIGURE 5

A subscription sells six layers, and a replicator obtains each in a different way

	Layer	What the subscriber gets	How a replicator obtains it	Cost class
1	**Function**	Features, screens, calculations	Tokens, human oversight, access to observe the original	Token-priced
2	**Data**	Inputs and accumulated history	A pipeline for public data, fees for licensed data; recorded history cannot be bought	Market-priced; Not for sale
3	**Operations**	Uptime, updates, security, support	Hosting, and someone's attention	Market-priced
4	**Stream**	Improvements that have not shipped yet	Only by following the original, always late	Token-priced
5	**Content and judgement**	Teaching, analysis and curation by a person	Cannot be replicated lawfully, only substituted with original work	Not for sale
6	**Trust and network**	Track record, accountability, community, other users	Earned over time, or assembled person by person	Not for sale

Token-priced — falls as AI improves; measured by the Index

Market-priced — hosting, licences and wages; moves slowly

Not for sale — time-locked; cannot be bought at any price

The cost class decides how a layer's replication cost behaves as AI improves.

What the cost follows

Each layer's cost follows something different. Function follows functional size, observability and fidelity, and falls with AI. Data follows its data class and is unchanged by AI. Operations follow scale, and a floor remains. The stream follows the creator's rate of change. Content and judgement are unchanged by AI: a reputation is not for sale. Trust and network follow calendar time and coordination.

Three cost classes

Token-priced costs fall as AI improves. Market-priced costs, such as hosting, licences and wages, move slowly. Not-for-sale assets cannot be bought at any price because they are time-locked: a recorded history, a track record, an audience.

Data classes

The checker assigns each data set one class.

TABLE 5 **Data classes**

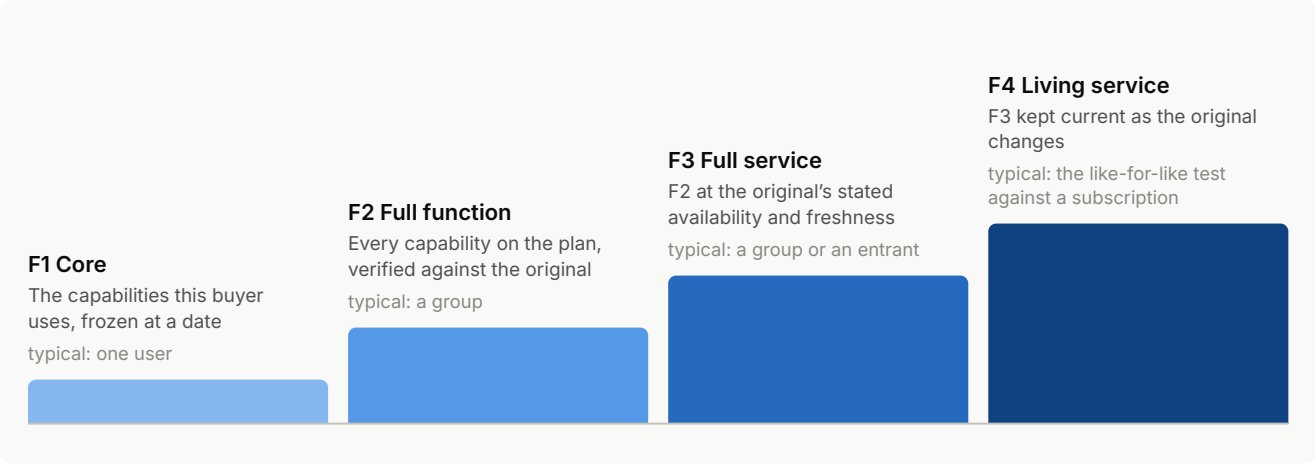
CLASS	MEANING	REPLICATION COST
D0	Public and free	The pipeline only
D1	Available under licence	The licence fees, for as long as the replica runs
D2	Derived from D0 or D1 by the service's own logic	Counted as function
D3	Time-locked: recorded by the service and no longer obtainable	None at any price. Reported as not replicable
D4	Contributed by the service's users	None for one replicator. Reported as not replicable

Fidelity levels

A cost means nothing without the standard it meets. Every ORC figure names one of four levels.

FIGURE 6

Four fidelity levels: a cost is always stated at one of them



F4 is the like-for-like comparison, because a subscription is itself a living service.

SECTION 4

The cost model

Open Replication Cost is the cheapest demonstrated lawful cost of supplying k users with the service at fidelity F for T months.

Definition

The cost has three parts: a build, an entry cost for each member, and a running cost for every month of the horizon.

$$\mathrm{ORC}(F,k,T) \;=\; \underbrace{\frac{B}{\pi}}_{\text{build, risk-adjusted}} \;+\; \underbrace{k\,a}_{\text{entry}} \;+\; \underbrace{A_T\,R(k)}_{\text{running}} \tag{1}$$

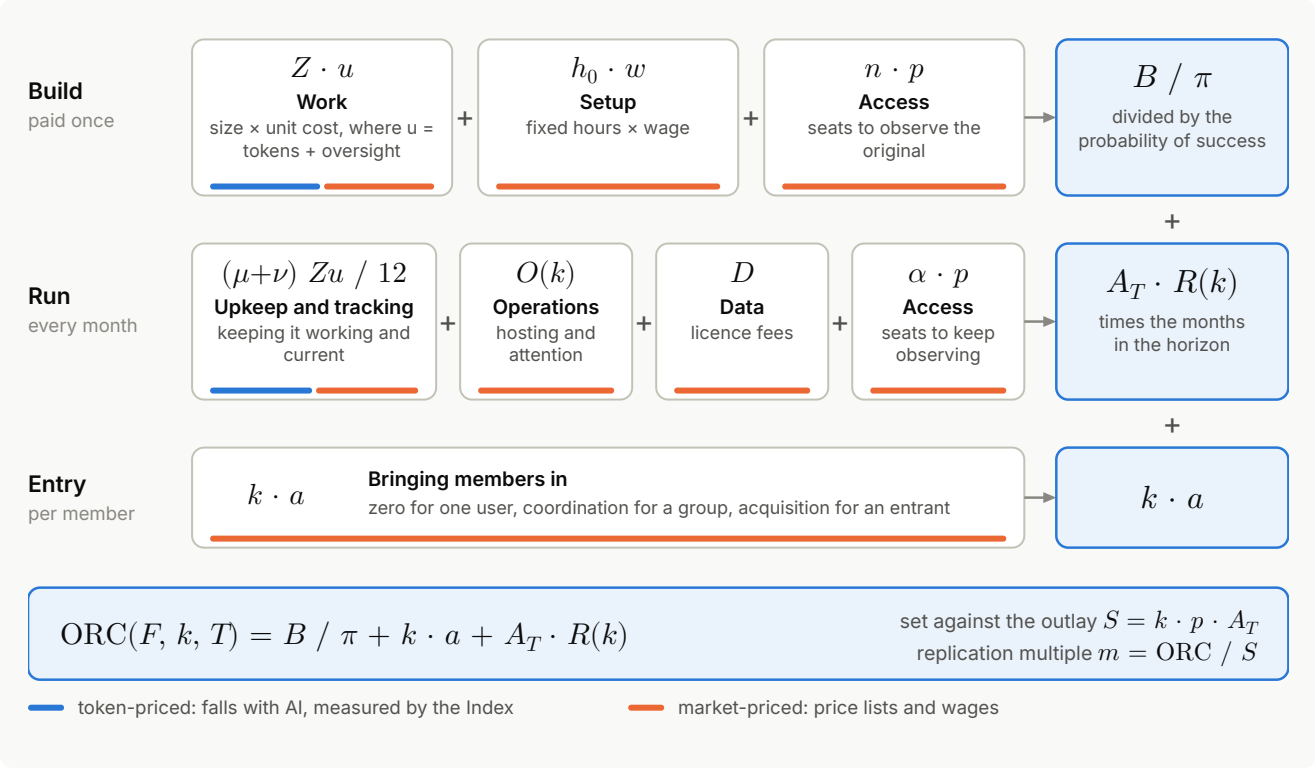
The build is paid once and the running cost every month.

$$B \;=\; Z\,u \;+\; h_0\,w \;+\; n\,p, \qquad u \;=\; \tau\,q \;+\; \eta\,w \tag{2}$$

$$R(k) \;=\; \frac{(\mu+\nu)\,Z\,u}{12} \;+\; O(k) \;+\; D \;+\; \alpha\,p, \qquad A_T \;=\; \sum_{t=1}^{T} \delta^{t} \tag{3}$$

FIGURE 7

How the cost is assembled: a build paid once, a running cost paid monthly, an entry cost per member



Only the terms marked token-priced fall as AI improves. The others are read from price lists and wage statistics.

TABLE 6 **Notation**

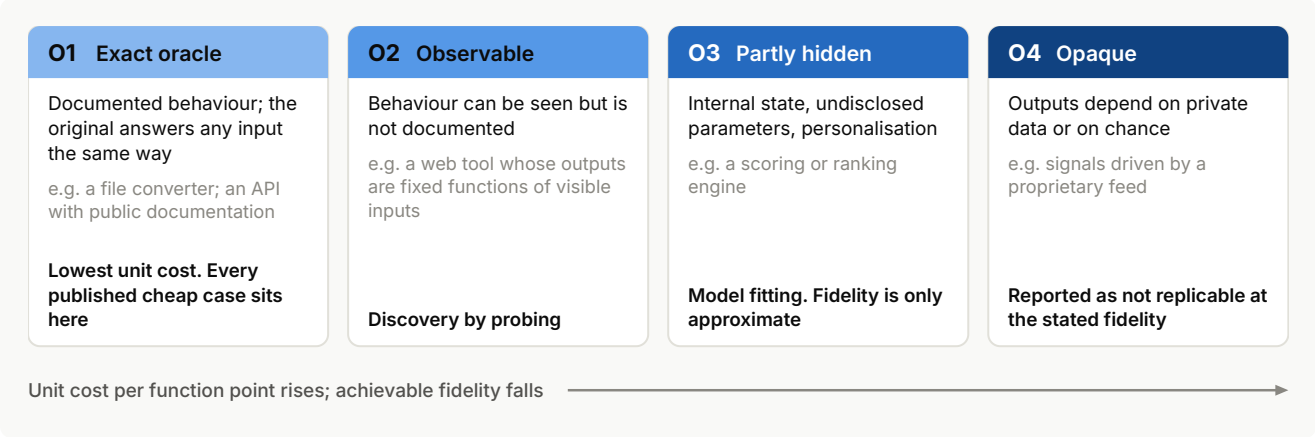
SYMBOL	MEANING	SOURCE
Z	Functional size of the capabilities in scope, in COSMIC function points[25]	Counted from the public surface
τ, η, π	Tokens per point, oversight hours per point and probability of success, by system class, observability and fidelity	The Index
q	Price per token of the models the Index used	Providers' price lists[26]
w	Hourly wage of the skill required	Public wage statistics[17]
h_0	Fixed setup hours	The Index
n, α	Months of access needed to observe the original during the build; share of months needed afterwards	Protocol defaults
p	The plan's price per month	The creator's price list
μ, ν	Yearly upkeep rate; the original's yearly rate of functional change	The Index; the target's changelog
$O(k)$	Hosting, operator attention and coordination for k users	Cloud price lists,[27] wage
D	Data licence fees per month	Vendors' price lists
a	One-off cost of bringing in each member: zero for self-supply, coordination for a group, acquisition for an entrant	Declared
δ	Monthly discount factor	A public reference rate
u, B, $R(k)$	Unit cost per function point; build cost; running cost per month	Equations (2) and (3)
A_T	The months of the horizon, discounted. Equal to T when there is no discounting	Equation (3)

Five rules keep the number honest

1. **Cheapest demonstrated route.** ORC is the minimum over published, reproducible methods, including adopting an existing open equivalent. It is an upper bound that tightens as cheaper routes are shown.
2. **Lawful route only.** The cost of an unlawful shortcut is never counted.
3. **Stated fidelity.** Every figure names its level, F1 to F4.
4. **Risk-adjusted.** Build cost is divided by the probability that the attempt reaches the stated fidelity.
5. **Observability graded.** How much of the behaviour can be seen from outside is the largest cost driver, so it is graded explicitly (Figure 8).

FIGURE 8

Observability is the largest cost driver, so it is graded



Every published low-cost rebuild had an exact oracle. A closed service usually offers less.

Three derived figures

The comparison uses three derived figures.

$$S(k,T) \;=\; k\,p\,A_T \qquad\qquad m \;=\; \frac{\mathrm{ORC}(F,k,T)}{S(k,T)}$$

$$\mathrm{ORC}_{\infty} \;=\; \mathrm{ORC}(F,k,T)\Big|_{\tau=0,\;\eta=0} \tag{4}$$

S is what the group pays the creator. The replication multiple m says how many times dearer replication is than subscribing. The floor, written ORC_{∞}, is what replication still costs when tokens and oversight are free.

SECTION 5

Three parity prices and the decision rule

One formula yields three thresholds, and a price is safe only below all of them.

The parity price

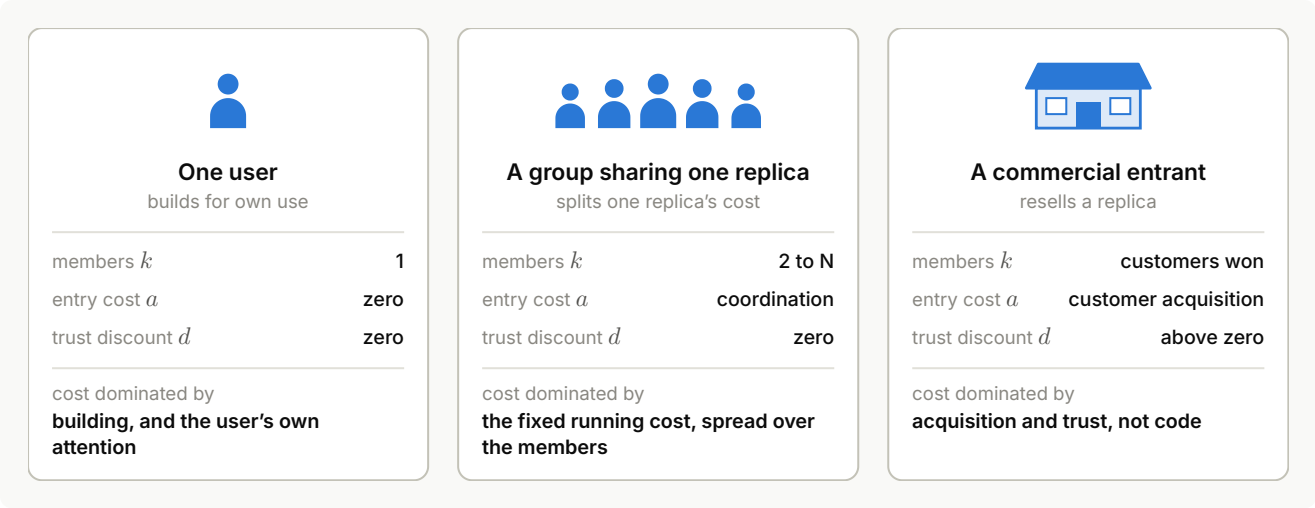
The parity price is the price at which subscribing and replicating cost the same.

$$p^{*}(k) \;=\; \frac{\dfrac{B'/\pi}{k\,A_T} \;+\; \dfrac{a}{A_T} \;+\; \dfrac{R'(k)}{k}}{1 \;-\; d \;-\; \dfrac{\alpha}{k} \;-\; \dfrac{n}{\pi\,k\,A_T}} \tag{5}$$

In the formula, B' and R' are the build and running costs without the observation seats, and d is the discount buyers demand before they trust a newcomer over the original. The formula is evaluated three times, once for each kind of replicator (Figure 9).

FIGURE 9

Three replicators face three different costs



The same formula is evaluated with each replicator's members, entry cost and trust discount.

Four results

Four results follow directly.

1 Peace condition

If the outlay S is at or below ORC for every group size and horizon, no buyer or group gains by replicating and no entrant can profitably undercut. These are the subsidy-free prices of Faulhaber's test.[13]

$$S(k,T) \;\le\; \mathrm{ORC}(F,k,T) \qquad \text{for every group size } k \text{ and horizon } T \tag{6}$$

2 Floor

If the price is at or below a replica's running cost per member, replication never pays for that group size, even when building is free. Such a price is immune to falling token prices.

$$p \;\le\; \frac{R(k)}{k} \quad\Longrightarrow\quad S \;=\; k\,p\,A_T \;\le\; A_T\,R(k) \;\le\; \mathrm{ORC} \tag{7}$$

3 Living service

If following the original needs continuous access, so that α is 1, the denominator of the parity price is zero or negative for a single user. One user can never replicate a living service for less than its subscription. A lone replicator can win only by accepting a frozen copy.

$$\alpha = 1,\;\; k = 1 \quad\Longrightarrow\quad 1 - d - \frac{\alpha}{k} - \frac{n}{\pi\,k\,A_T} \;\le\; 0 \tag{8}$$

4 Rent bound

An entrant that serves N customers at a running cost of c each can undercut only a price above the limit price p_L.

$$p_L = \frac{1}{1-d}\left[c + \frac{a}{A_T} + \frac{B/\pi}{N\,A_T} \right] \tag{9}$$

The build term shrinks towards zero as customers grow, so what remains is acquisition and the trust discount.

$$p_L \longrightarrow \frac{c + a/A_T}{1-d} \qquad \text{as } N \to \infty \tag{10}$$

At market scale the most a creator can earn above its own running cost is a replica's entry cost per customer. The return to having written the software disappears. The return to being known, trusted and current does not.

The decision rule

The decision rule runs in one direction.

TABLE 7 **The decision rule**

RESULT FOR THE BUYER'S OWN CASE	VERDICT	WHAT THE REPORT SAYS
Outlay at or below ORC	**Subscribe**	Paying is cheaper than even a modest copy
Outlay above ORC	**Premium, with its amount**	What the premium buys: the living service and the layers tokens cannot buy. The judgement is the buyer's

Why one direction

The checker never advises replication, for three reasons. ORC leaves out the layers it cannot cost, so it understates the true cost of an equivalent. Lawfulness depends on jurisdiction and terms. And a public standard should not be a cloning service.

What creators read

Creators read three numbers instead of one: the individual multiple, the group size at which a shared replica matches the price, and the group size that stays safe at the floor. A shared replica matches the price at the group size k^*, where R_{fix} is the replica's fixed running cost per month and r is what each further member adds.

$$k^* = \frac{(B/\pi)/A_T + R_{\text{fix}}}{p - r} \tag{11}$$

The tier test

A fourth test covers pricing tiers. The price of each upgrade step, from p_1 to p_2, is compared with the ORC of the capabilities the step adds.

$$m_\Delta = \frac{\text{ORC}(\Delta Z)}{(p_2 - p_1)\,A_T} \tag{12}$$

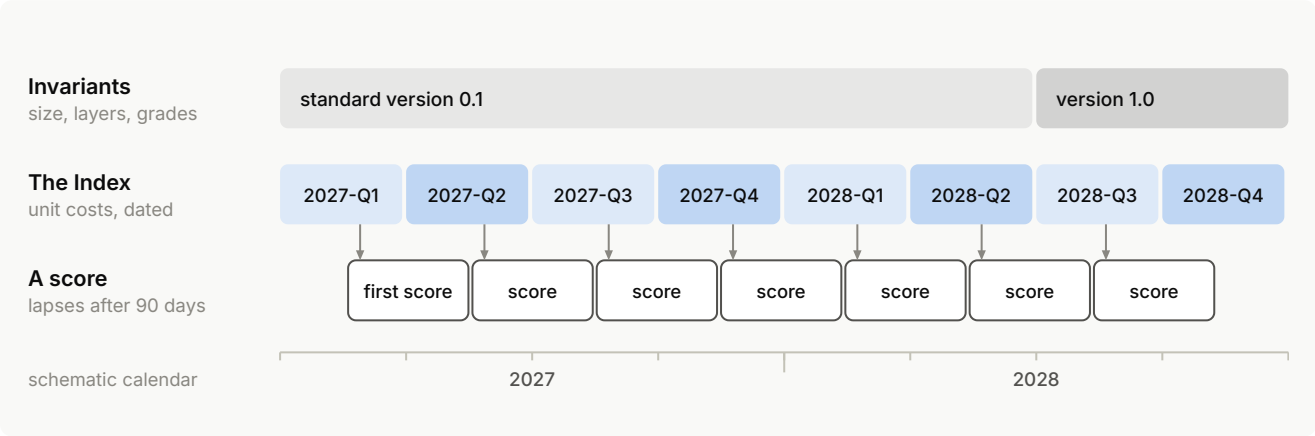
SECTION 6

Stability: the invariants, the Index and the floor

The score stays meaningful as AI improves because it keeps apart three things that change at different speeds.

FIGURE 10

Three clocks: the standard changes rarely, the Index every quarter, and a score lapses after 90 days



Schematic. Each score is stamped with the standard version, the Index epoch, the manifest version and its date.

TABLE 8 **What is fixed, what is republished and what expires**

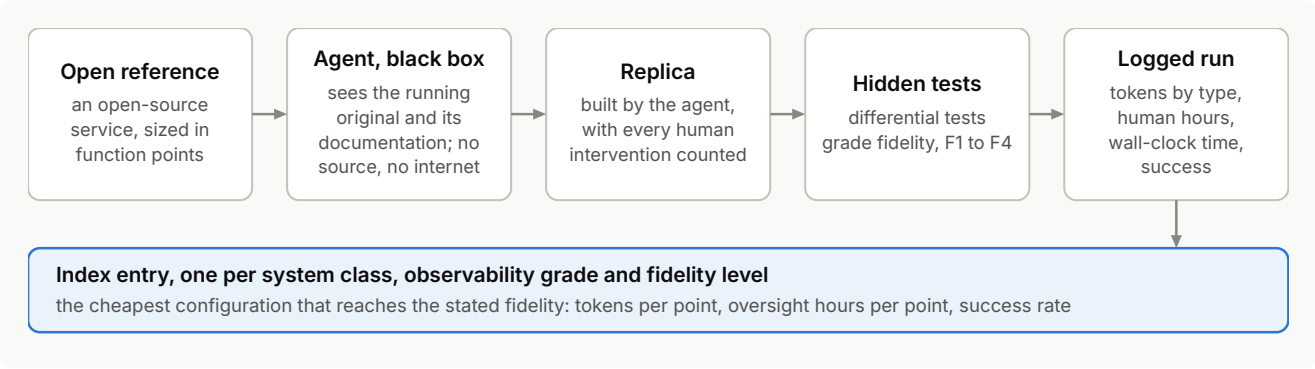
PART	CONTENTS	HOW IT CHANGES
Invariants	Functional size under ISO/IEC 19761,[25] layer and data classes, fidelity and observability grades	Only with a new version of the standard
The Index	Tokens and oversight hours per function point, success rates, token prices, wage, hosting	Republished every quarter, dated
A score	The invariants priced with one Index epoch	Stamped with standard version, Index epoch, manifest version and date. Lapses after 90 days

How the Index is measured

Agents rebuild open-source reference programs under black-box conditions: the running original and its documentation, no source, no internet. Hidden differential tests grade fidelity. Each run logs tokens by type, wall-clock time and human interventions, and each reference is sized in function points. The published unit cost is the cheapest configuration that reaches the stated fidelity.

FIGURE 11

The Index is measured on open-source references, never on the target



One entry per system class, observability grade and fidelity level.

What exists already

Most of this exists already. MirrorCode tests execute-only rebuilds against held-out tests, and its best model scores 56%.[28] Behavior2Code grades 60 programs against 6,872 hidden differential tests.[29] GameReplica covers visual black-box replication and finds that agents copy appearance more readily than rules.[30] The Index adds four things: web services with accounts and data pipelines, cost accounting, functional sizing, and the higher fidelity levels.

Why nothing is hard-coded

The price of a fixed level of model capability has been falling 5 to 10 times a year.[31] The length of task an agent completes alone has doubled every 131 days since 2023.[32] Any constant written into the standard would be wrong within a quarter.

The floor

Every score reports what remains at the zero-token limit: observation seats, hosting, data licences, coordination or acquisition, a minimum of human attention, and calendar time. Three readings follow.

- A price at or below floor parity is durable.
- A price between floor parity and today's parity is safe now and erodes at the Index's pace. The parity runway is the number of months until the multiple reaches 1 on the Index trend.
- For a product whose value is function over public data, the individual floor is close to zero. The framework says so plainly.

The parity runway

If every AI-priced cost halves each h months, replication cost falls from today's value towards the floor,

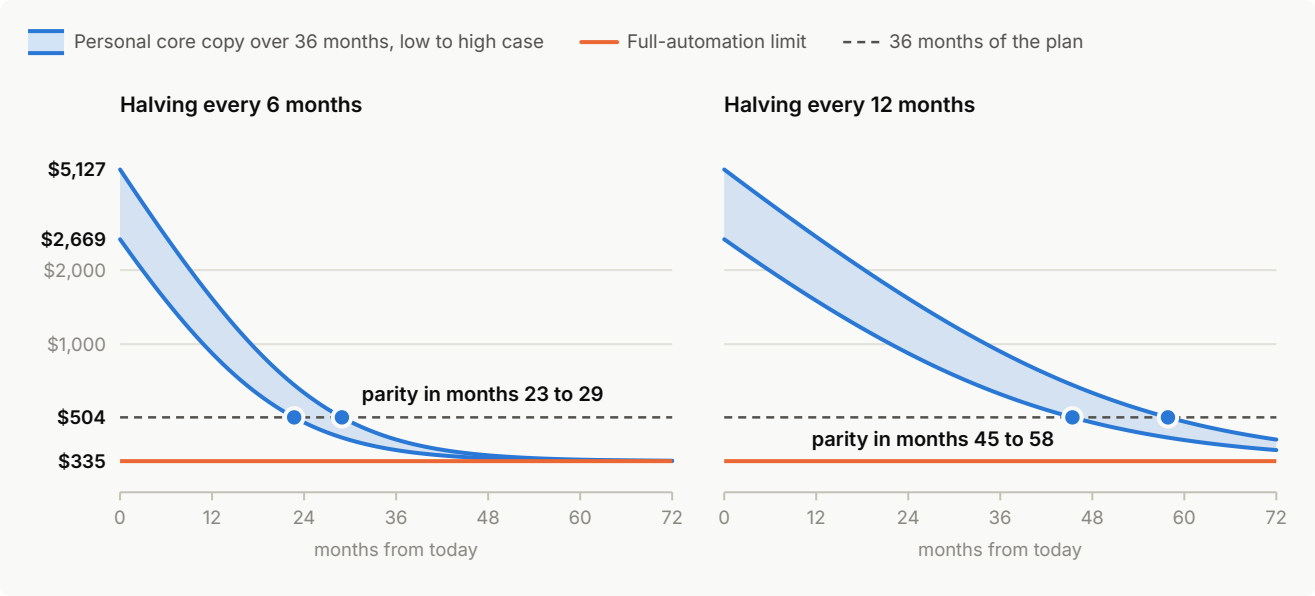
$$\mathrm{ORC}(t) \;=\; \mathrm{ORC}_{\infty} + \left(\mathrm{ORC}(0) - \mathrm{ORC}_{\infty}\right) 2^{-t/h} \tag{13}$$

and it meets the outlay S after t^{*} months.

$$t^{*} \;=\; h \,\log_{2} \frac{\mathrm{ORC}(0) - \mathrm{ORC}_{\infty}}{S - \mathrm{ORC}_{\infty}} \tag{14}$$

FIGURE 12

Parity runway: if AI-priced costs halve every 6 to 12 months, the worked example reaches parity in two to five years



Personal copy of the COTBase Premium core over 36 months (Section 10), low and high case. Parity is reached in months 23 to 29 if AI-priced cost halves every 6 months, and in months 45 to 58 if every 12. The cost never falls below the full-automation limit of $335. Log scale. Grade C.

SECTION 7

The checker

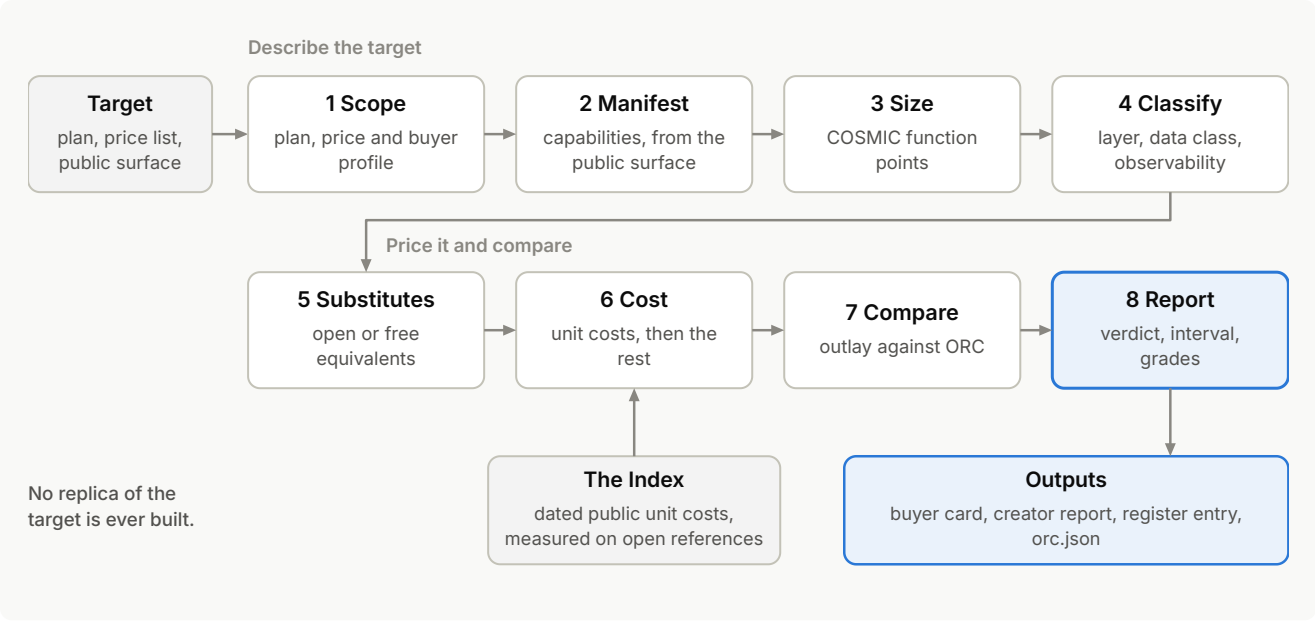
The checker turns a public product page into a score in eight steps, and it never builds a replica of the target.

The eight steps

1. **Scope.** Fix the plan, its price and the buyer profile: the capabilities used and the horizon.
2. **Manifest.** List every capability the plan offers, from the public site, documentation, feature matrix, changelog and free tier. The manifest is published, and the creator may correct it.
3. **Size.** Count each capability's data movements. Each Entry, Exit, Read or Write is one COSMIC function point.[25] Mark the core subset by usage.
4. **Classify.** Give each capability a layer, a data class and an observability grade, and set the fidelity target.
5. **Substitutes.** Search for open or free equivalents, and measure how much of the manifest each one covers.
6. **Cost.** Apply the Index's unit costs. Add access, operations, data, upkeep and tracking. Divide the build by its probability of success.
7. **Compare.** Compute outlay, ORC and the multiple for one user, the curve for groups, the floor, the token share and the runway.
8. **Report.** State the verdict with its interval and evidence grades, notify the creator, and publish after the notice period.

FIGURE 13

The checker protocol: eight steps, with the Index as the only outside input



The target is described and priced, never rebuilt.

Evidence grades Every input carries one.

TABLE 9 **Evidence grades**

GRADE	MEANING	EXAMPLES
A	A published fact	Price lists, wage statistics, statutes
B	Measured on the reference suite	Unit costs and success rates from the Index
C	A declared assumption	A size counted without the creator's confirmation; a bootstrap unit cost

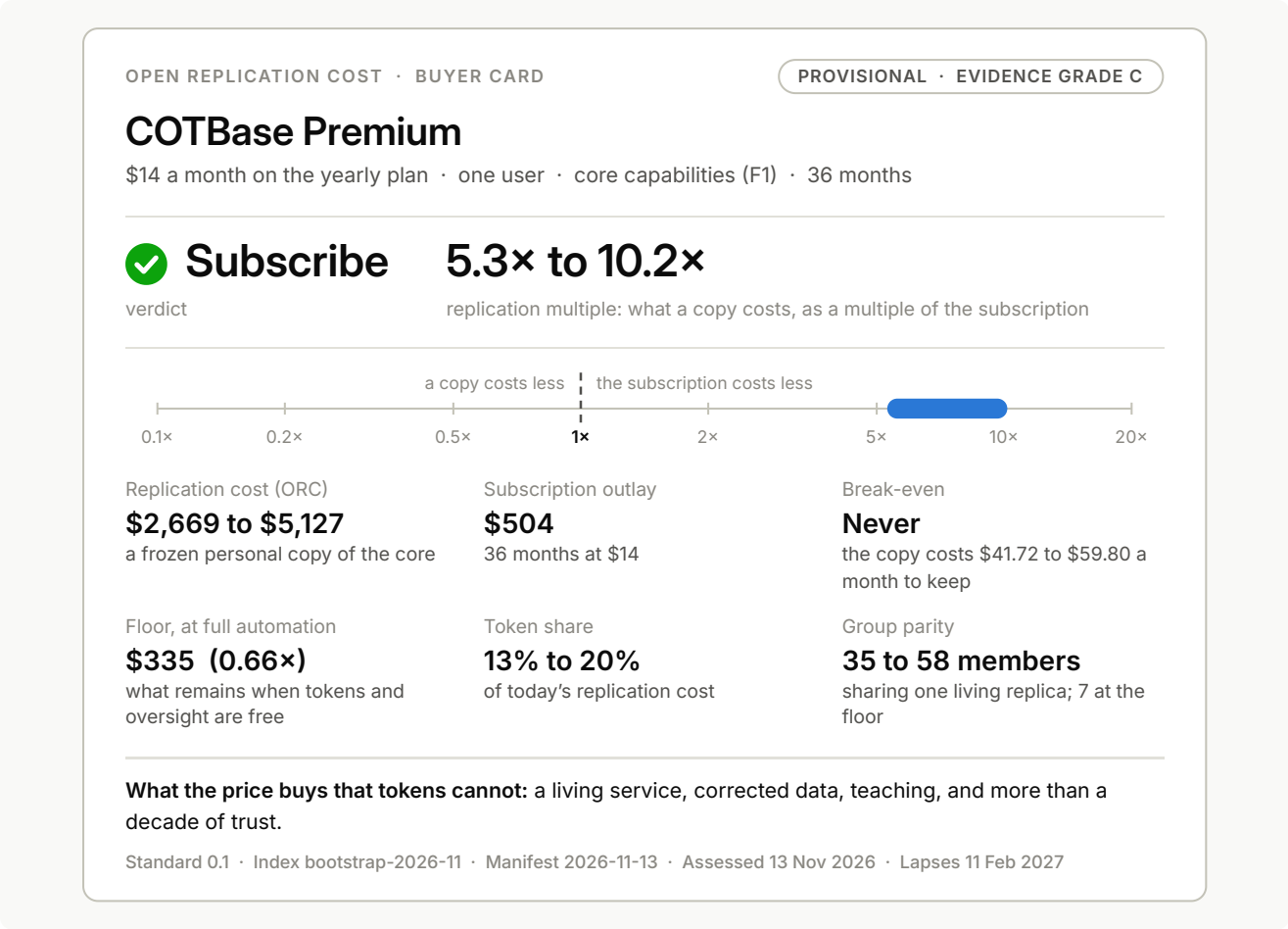
The headline uses A and B inputs. C inputs widen the interval, and a score that rests on them is labelled provisional.

TABLE 10 **What each reader gets**

READER	CONTENTS
Buyer	Verdict, multiple, break-even horizon, the premium and what it buys
Creator	The three thresholds, the floor, the token share, the runway, the capabilities that carry the cost, and the share due to obscurity
Public register	Verdict band, interval, version stamps and manifest hash. No build plan

FIGURE 14

A buyer card: the worked example as the checker would report it



Mock-up. The figures are those of Section 10 and are provisional: they rest on declared assumptions.

A machine-readable statement

Creators would publish the same facts at `/.well-known/orc.json`: plans and prices, the manifest, declared data classes and time-locked assets, a self-assessed score, independent attestations, and the terms of a parity licence. Buyers' agents will soon run a build-or-buy check before every renewal. An open statement gives them a fair one to read. Listing 1 shows the statement for the worked example in this document.

LISTING 1 **The worked example as an orc.json statement**

```
{
  "orc_version": "0.1",
  "index_epoch": "bootstrap-2026-11",
  "assessed": "2026-11-13",
  "plan": {"name": "Premium", "usd_per_month": 14, "billing": "yearly"},
  "size_cfp": {"core": 100, "full": 300},
  "data_classes": ["D0", "D2"],
  "time_locked": ["operating history", "audience"],
  "score": {
    "fidelity": "F1", "group": 1, "horizon_months": 36,
    "orc_usd": [2669, 5127], "multiple": [5.3, 10.2], "floor_usd": 335,
    "token_share": [0.13, 0.20], "verdict": "subscribe",
    "grade": "C", "provisional": true
  },
  "group": {"matches_price_at": [35, 58], "safe_below": 7}
}
```

Cheap by design

A check is cheap, because it sizes and prices but builds nothing. That is the lesson of the rail test: a ceiling that needs a team of economists for each case protects nobody.

SECTION 8

The peace

The conflict is over a rent nobody can see. Once its size is public, both sides do better by pricing it than by fighting over it.

Without a public cost

Creators guess how exposed they are and buyers guess what a rebuild would take. Wrong guesses produce real clones, which duplicate work, and defensive moves: terms against reverse engineering, withheld documentation, closed test suites, lock-in. SQLite's code is public domain, while its TH3 test harness, which gives 100% branch coverage, is proprietary.[33] Each defence makes software worse for the people who pay.

With a public cost

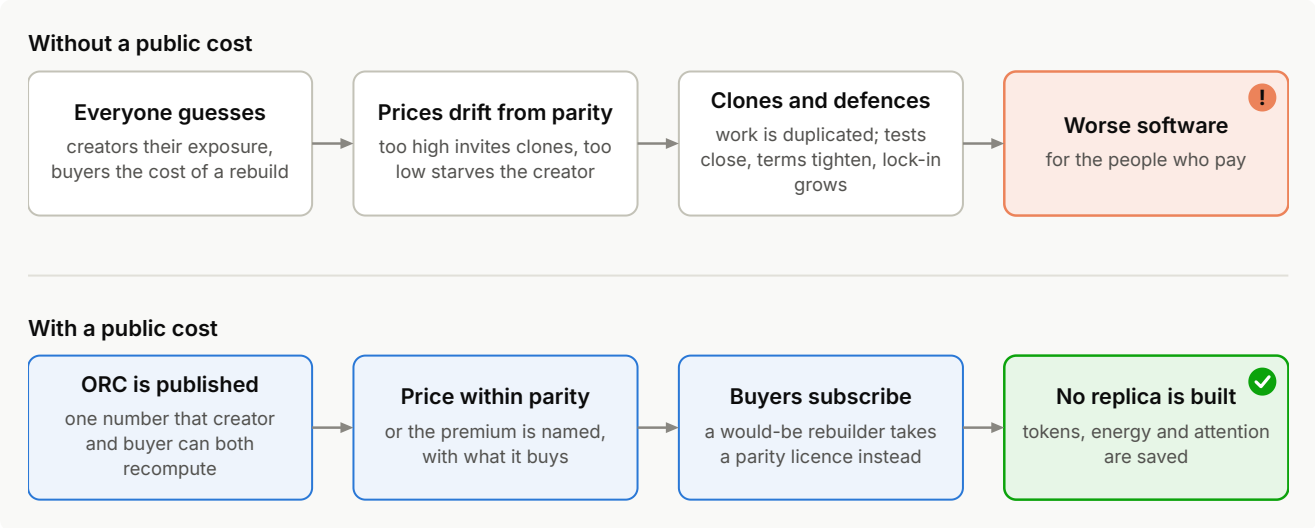
The creator's best move is to price within parity and the buyer's best move is to subscribe. The replica is never built. The threat disciplines the price as well as the act would, and the tokens, energy and attention a clone would burn are saved.

When a price exceeds parity

The same number tells the creator to lower the price or to move value into layers tokens cannot buy. It tells the buyer exactly what the premium is for.

FIGURE 15

Contestability without the contest: the threat disciplines the price, and the replica is never built



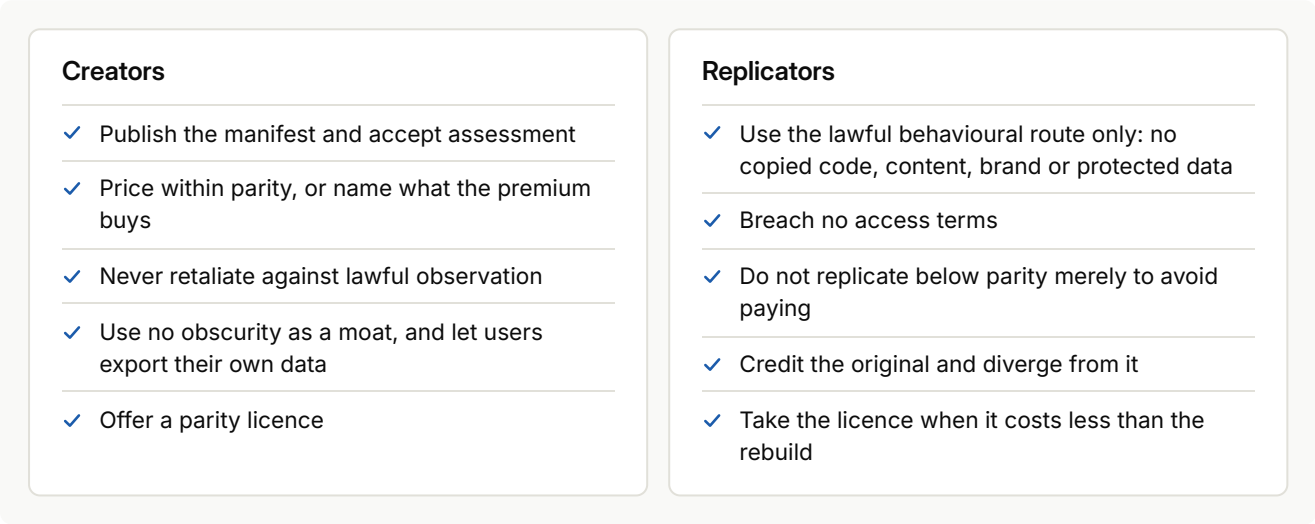
The mechanism is Baumol's: where entry is easy, the threat of entry does the work of entry.[14]

TABLE 11 **Six instruments**

INSTRUMENT	OFFERED BY	WHAT IT DOES
Parity pricing	Creator	Publishes the score beside the price, and keeps the price within parity for the intended buyer
Tier parity	Creator	Makes each upgrade step pass the incremental test
Parity licence	Creator	Sells source or a self-host licence for less than the licensee's ORC. Both sides gain against a rebuild, by the logic of the design-around cap[2]
Parity statement	Creator	Publishes orc.json, so that purchasing agents can check before they build
Notice	Checker	Gives the creator time to respond before a first score is published
Divergence	Replicator	Adds something the original lacks. A replica that does so is creation

FIGURE 16

Commitments on both sides



The norm binds only replication whose purpose is to avoid paying.

> **One exemption matters**
> Replication for interoperability, security research, preserving a discontinued service, accessibility or learning is legitimate at any price. The norm applies only to replication whose purpose is to avoid paying.

SECTION 9

Gaming and failure modes

Any published score will be gamed, so the standard assumes it.

TABLE 12 **Nine attacks and their safeguards**

ATTACK	BY	SAFEGUARD
Obscurity: hide documentation, add noise or block export to raise discovery cost	Creator	Report essential cost, as if the product were documented, apart from friction. The fairness verdict uses essential cost, and the friction share is published
Padding: add unused features to raise size	Creator	The individual verdict uses the usage-weighted core
Churn: change cosmetics often to inflate tracking	Creator	Count functional change in used capabilities only, with a cap
Flattering checker	A checker the creator pays	Open method and inputs, so anyone can recompute. Funding disclosed. A challenge window
Phantom cheap route: claim a cheaper method to depress a rival's score	Competitor	A route counts only when reproduced on the reference suite at the stated fidelity, or when a substitute's coverage of the manifest is measured
Overfitting the Index	Model and tool makers	Rotating references, hidden tests, contamination screens, several harnesses
Target list: a public register of cheap-to-clone products guides cloners	Anyone	A notice period. Opt-in below a size threshold. The register shows bands, never a build plan
Bundles and price discrimination	Creator	Each plan and each upgrade step is tested separately
Profile picking: choose a heavy-use buyer to enlarge the core	Creator	Standard buyer profiles, plus the buyer's own

SECTION 10

Worked example: COTBase.com

At $14 a month COTBase Premium passes the individual test by a factor of 5 to 10, a shared replica would match its price at 35 to 58 members, and the Pro plan sits at the edge.

> **An illustration, not a certified score**
> The sizes are a rough count from the public site, and the unit costs are bootstrap values. Every result below carries evidence grade C.

TABLE 13 **Published facts (grade A)**

INPUT	VALUE	SOURCE
Premium	$19.99 a month; $14 a month on the yearly plan; $12 on the two-year plan	cotbase.com[34]
Pro	$99 a month; $83 yearly; $69 two-year. Adds a NinjaTrader indicator, API access, downloads and all data sets	cotbase.com[34]
Underlying data	CFTC reports, public and free, released on Fridays with Tuesday's positions. A twice-weekly schedule is announced for the end of 2026	CFTC,[35] World Grain[36]
Free substitutes for basic charts	The CFTC's own reporting environment, Barchart, and COTBase's free tier	Barchart[37]
Developer wage	$65.38 an hour, from a median of $135,980 a year	BLS[17]
Smallest cloud server	€5.49 a month, about $6.37 at 1.16 dollars to the euro	Hetzner[27]

TABLE 14 **The six layers at COTBase**

LAYER	AT COTBASE	CLASS
Function	Net-position charts, corrected and combined series, market complexes, options-only view, analysis views	Token-priced. Observability O1 to O2, since outputs are fixed functions of public data
Data	CFTC reports, and a derived, corrected history	D0 and D2: pipeline cost only
Operations	Weekly ingest and hosting	Market-priced and small
Stream	New tools as they ship	Tracking
Content and judgement	The 'On COT' guides, the blog, the YouTube channel, consultations	Not replicable
Trust and network	More than a decade in operation, and an audience	Not for sale
iCOT	An intraday estimate built from market data by a proprietary algorithm	O3 to O4: the algorithm is hidden, so a rebuild cannot be verified. Reported as not replicable at the stated fidelity

TABLE 15 **Assumptions (grade C)**

ASSUMPTION	VALUE	BASIS
Size	100 function points for a personal core; 300 for the full Premium service; 30 for what Pro adds for one user	A rough count of the public feature list
Tokens per point	$2 to $6	Brackets the published cases at about 50 lines per point:[16] MirrorCode's cap works out at $1.63, the C compiler at $10
Oversight per point	0.08 to 0.24 hours	One engineer-week for vinext
Unit cost	$7.23 to $21.69 per point	A human rebuild under COCOMO II[15] costs about $1,715 per point, 79 to 237 times more
Success probability	0.8 for the core; 0.6 for the full service	Declared
Upkeep; tracking	15% and 10% of build work per year	Declared
Attention	0.5 hours a month for a personal copy; 4 hours a month for a shared one	Declared
Full automation	One hour to commission the build; 0.1 hours a month afterwards	Declared
Horizon	36 months, no discounting	Declared

TABLE 16 **How the numbers add up**

LINE	CALCULATION	LOW CASE	HIGH CASE
PERSONAL COPY OF THE CORE: ONE USER, FIDELITY F1, 36 MONTHS			
Build work: tokens	100 points at $2 or $6	$200.00	$600.00
Build work: oversight	100 points at 0.08 or 0.24 hours, at $65.38 an hour	$523.00	$1,569.00
Setup	3 hours at $65.38	$196.12	$196.12
Access to observe the original	1 month at $14	$14.00	$14.00
Build, risk-adjusted: B/π	The four lines above, divided by 0.8	$1,166.41	$2,973.91
Upkeep	15% of the build work a year, for 3 years	$325.35	$976.05
Attention	0.5 hours a month at $65.38, for 36 months	$1,176.75	$1,176.75
Replication cost: ORC		**$2,668.51**	**$5,126.71**
Outlay S	36 months at $14	$504.00	$504.00
Replication multiple m	ORC divided by the outlay	**5.29×**	**10.17×**

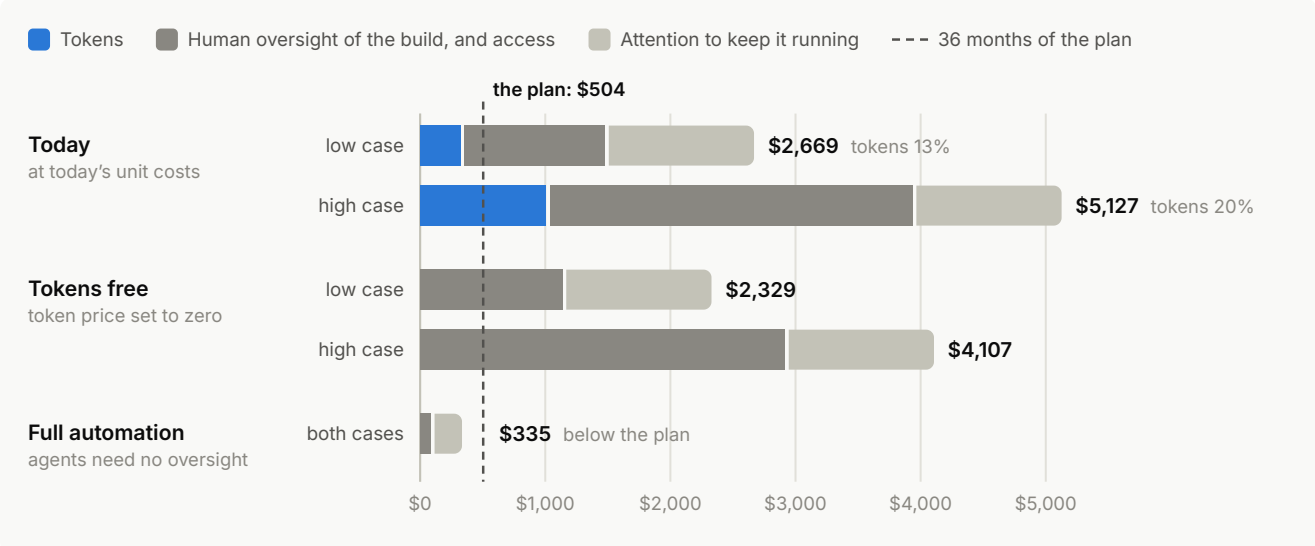
LINE	CALCULATION	LOW CASE	HIGH CASE
SHARED LIVING REPLICA: k MEMBERS, FIDELITY F4, 36 MONTHS			
Build, risk-adjusted: B/π	300 points at the unit cost, 8 setup hours and 3 months of access, divided by 0.6	$4,556.67	$11,786.67
Upkeep and tracking, a month	25% of the build work a year	$45.19	$135.56
One seat, hosting and an operator, a month	$14, $6.37 and 4 hours at $65.38	$281.87	$281.87
Fixed running cost a month: R_{fix}	The two lines above	$327.06	$417.43
Members at parity: k^*	Equation (11), with $1 a month of coordination per member	**34.9**	**57.3**

TABLE 17 **Results**

TEST	LOW CASE	HIGH CASE	AGAINST THE PRICE
Personal frozen copy of the core, 36 months	$2,669	$5,127	$504 on the yearly plan: 5.3 to 10.2 times dearer
Its running cost per month	$41.72	$59.80	Above $14, so it never breaks even
The same copy with the user's own time valued at zero	$358	$1,038	0.7 to 2.1 times. Break-even after 23 to 118 months
The same copy at full automation	$335	$335	0.66 times
Members at which a shared living replica matches $14	35	58	15 to 38 with a volunteer operator. 7 at full automation
Pro, whole plan, 36 months	$3,123	$6,319	$2,988 on the yearly plan: 1.05 to 2.1 times
Pro upgrade step alone	$369	$1,106	$2,484: 0.15 to 0.45 times
Parity runway for Premium	23 to 45 months	29 to 58 months	If AI-priced cost halves every 6 to 12 months

FIGURE 17

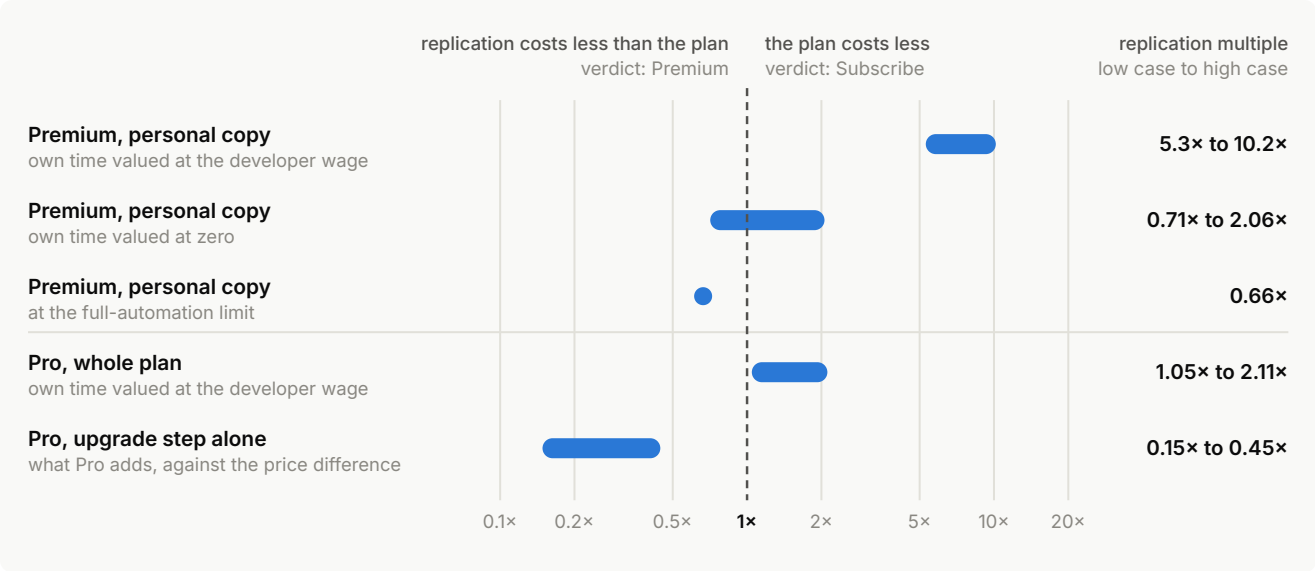
Tokens are 13% to 20% of what a personal copy costs today



Personal copy of the Premium core over 36 months. Removing the token bill barely moves the total. Removing the need for human attention drops it below the subscription. Reference model v0.1, grade C.

FIGURE 18

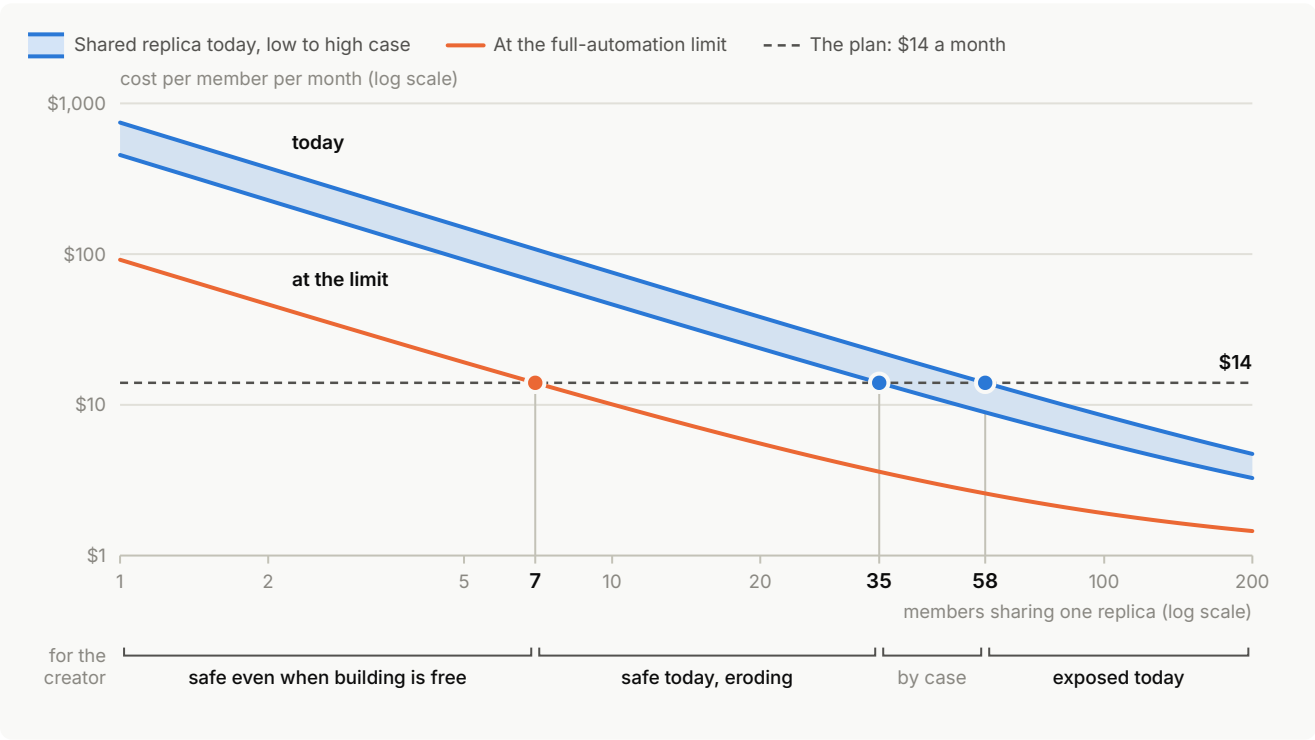
Premium passes the individual test by 5 to 10 times, and the Pro upgrade step fails it



Replication multiple: replication cost divided by the subscription outlay over 36 months, on the yearly plans. Bars span the low and the high case. Log scale. Reference model v0.1, grade C.

FIGURE 19

A shared living replica matches the $14 plan at 35 to 58 members today, and at 7 at the limit



Cost per member per month of one shared living replica over 36 months. The space between the two curves is the protection that AI is eroding. Groups too small to reach the lower curve stay safe even when building is free. Between 35 and 58 members the answer depends on the case. Reference model v0.1, grade C.

What the numbers say

1. **Premium: subscribe.** For anyone who values their time, a personal copy costs several times the subscription, and its upkeep alone costs more than the plan.
2. **Attention protects Premium, not tokens.** Tokens are 13% to 20% of the cost. As agents need less supervision that protection fades. At full automation a frozen copy costs about two thirds of the plan.

3. **Groups are the nearer threat.** A few dozen subscribers could share a replica at the same cost per head today, and seven could at the limit.
4. **Function is already at the floor.** Basic COT charts are free elsewhere. Premium's price is carried by the corrected data, the convenience, the teaching and more than a decade of trust.
5. **Pro fails the tier test.** The step from Premium to Pro costs 2 to 7 times what the added capabilities would cost a technically able user to add. Pro's buyers are the users most able to do that.
6. **iCOT is the defensible product, and an instructive one.** The site describes it as reverse-engineering COT data into an intraday estimate, which the CFTC does not publish.[34] That is replication as creation: it adds what the original lacks.

Robustness

The verdict on Premium is robust to these inputs: in both cases a personal copy's running cost exceeds the price, whatever the build costs. The Pro and group results are not robust. They move with the size count and the oversight hours.

SECTION 11

Governance and versioning

The standard needs a neutral keeper, because a score published by an interested party will not be believed.

TABLE 18 **Governance rules**

ELEMENT	RULE
Keeper	A body with creators, users and independent researchers, and no controlling vendor
Changes	By public proposal and comment. Each release is numbered
Stamps	Every score carries standard version, Index epoch, manifest version and date, and lapses after 90 days
Reproducibility	All inputs are published. Two independent counts of the same manifest must agree within a stated tolerance, or the manifest goes to dispute
Disputes	A creator may contest the manifest and its classifications during the notice period. Disputes are settled by recounting against the public surface
Independence	Checkers disclose their funding. Scores commissioned by the creator are labelled
Notice	A first score goes to the creator before publication. Scores the creator requests publish at once
Scope	Paid digital services. Version 0.1 excludes hardware, pure content libraries and services whose product is a regulatory licence

SECTION 12

Limits and open questions

The framework measures a moat. It does not create one, and it cannot guarantee that creating stays worth doing.

Ten limits

- **No Index yet.** Version 0.1 numbers are orders of magnitude, not measurements.
- **Size is not value.** COSMIC counts data movements, not algorithmic depth. Algorithm-heavy systems are under-sized until the Index carries class multipliers.
- **The outside view misses hidden work.** Edge cases, scale and back-office functions do not show on a product page. ORC is therefore understated, which is why the verdict runs one way.
- **Appropriability is not solved.** Under parity pricing a creator recovers at most what a replica would cost. Where creating costs far more than rebuilding, creating is under-rewarded, and transparency alone does not fix that. Samuelson and Scotchmer expected regulation in exactly this case.[10]
- **The first public clone.** Once an open replica exists, everyone's ORC falls to the cost of running it. For a popular product built on function alone, assume that day comes and price against the floor.
- **People differ.** Some value their time at zero. Some build for pleasure or on principle.
- **Trust and network are flagged, not priced.** The premium they justify stays a judgement.
- **The law is unsettled and varies by country.** A score is not a licence.
- **Publication can hurt small creators.** Notice and opt-in reduce the harm. They do not remove it.
- **Replication is sometimes the public good.** Competition, interoperability and preservation depend on it. A norm against replicating below parity must not grow into a norm against those.

Four open questions

1. How should observability be graded so that two checkers agree?
2. How should a creator's rate of functional change be measured from outside?
3. How long should the notice period be?
4. Should creators declare their creation cost beside ORC, so that the ratio between creating and rebuilding becomes visible?

SECTION 13

Roadmap

Five steps take this from a document to a working standard, and the first two can start with COTBase.

FIGURE 20

Five steps to a working standard

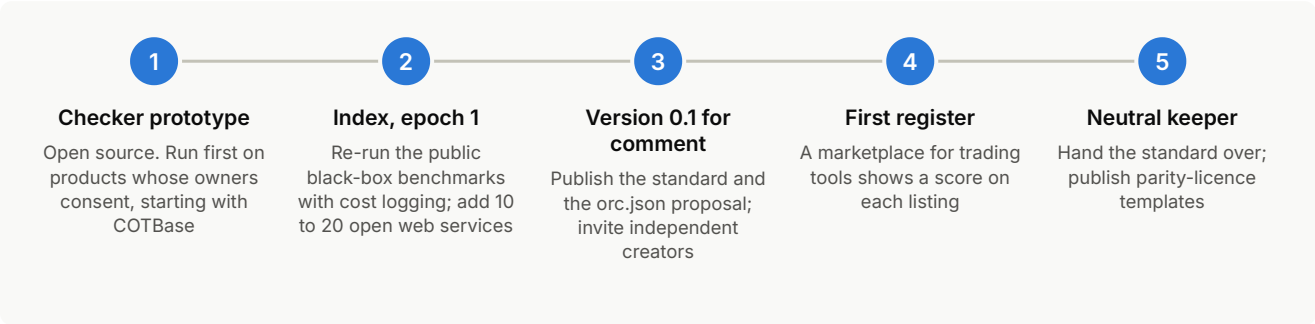


TABLE 19 **The five steps**

STEP	WORK	RESULT
1	Build the checker as open source and run it first on products whose owners consent, starting with COTBase	Scores with no notice problem, and a tested protocol
2	Bootstrap the Index: re-run the public black-box benchmarks with cost logging, and add 10 to 20 open-source web services sized in function points	Index epoch 1
3	Publish version 0.1 and the orc.json proposal for comment, and invite a handful of independent creators	A corrected standard
4	Open the first register where products are already listed and compared. A marketplace for trading tools, such as Traders Marketplace, can show a score on each listing	Adoption with a built-in audience
5	Hand the standard to a neutral keeper, and publish parity-licence templates	A standard nobody owns

APPENDIX A

The reference model

Every figure in the worked example comes from a short Python script, orc_model.py, which is attached to this PDF. The table lists what it computes.

TABLE 20 **Reference model v0.1: outputs**

QUANTITY	LOW CASE	HIGH CASE
INPUTS		
Developer wage, an hour (grade A)	$65.38	$65.38
Smallest cloud server, a month (grade A)	$6.37	$6.37
Unit cost per function point: tokens plus oversight	$7.23	$21.69
Human rebuild per function point under COCOMO II, 100 points	$1,714.98	$1,714.98
PERSONAL COPY OF THE PREMIUM CORE, 36 MONTHS		
Tokens	$340.00	$1,020.00
Human oversight	$1,134.26	$2,912.46
Access	$17.50	$17.50
Attention	$1,176.75	$1,176.75
Replication cost	$2,668.51	$5,126.71
Replication multiple, yearly plan at $14	5.29×	10.17×
Replication multiple, monthly plan at $19.99	3.72×	7.13×
Replication multiple, two-year plan at $12	6.17×	11.86×
Running cost a month	$41.72	$59.80
Own time valued at zero: replication cost	$357.50	$1,037.50
Own time valued at zero: multiple	0.71×	2.06×
Own time valued at zero: break-even, months	23.26	118.08
Full automation: replication cost	$334.57	$334.57
Full automation: multiple	0.66×	0.66×
SHARED LIVING REPLICA		
Build, risk-adjusted	$4,556.67	$11,786.67
Fixed running cost a month	$327.06	$417.43
Members at parity with $14	34.89	57.30
Members at parity, volunteer operator	14.78	37.18
Members at parity, full automation	6.98	6.98

QUANTITY	LOW CASE	HIGH CASE
PRO PLAN, 36 MONTHS, YEARLY PRICE OF $83		
Whole plan: replication cost	$3,123.49	$6,319.15
Whole plan: multiple against $2,988	1.05×	2.11×
Upgrade step: replication cost of what Pro adds	$368.73	$1,106.19
Upgrade step: multiple against $2,484	0.15×	0.45×
PARITY RUNWAY FOR THE PERSONAL COPY, MONTHS		
AI-priced cost halves every 6 months	22.70	28.93
AI-priced cost halves every 12 months	45.41	57.86

Reproducing the figures

The script states the evidence grade of every input in its header. Changing an assumption there and running it again reproduces every table and chart of Section 10.

APPENDIX B

Glossary

Open Replication Cost (ORC)
The cheapest demonstrated lawful cost of supplying a group of users with a service at a stated fidelity for a stated period.

Outlay
What the same group pays the creator over the same period: price, times members, times months.

Replication multiple
ORC divided by the outlay. Above 1, subscribing is cheaper.

Parity price
The price at which subscribing and replicating cost the same.

Floor
ORC when tokens and oversight are free. What is left is market-priced: seats, hosting, licences, attention.

Parity runway
The months until the multiple reaches 1 if AI-priced costs keep falling at the Index's pace.

Fidelity, F1 to F4
How complete and how current the replica must be, from a frozen personal core to a living service.

Observability, O1 to O4
How much of the original's behaviour can be seen from outside, from an exact oracle to opaque.

The Index
Dated public unit costs, measured by rebuilding open-source references under black-box conditions.

Manifest
The published list of a plan's capabilities, sized in function points and classified by layer.

COSMIC function point
The unit of functional size under ISO/IEC 19761: one data movement, an Entry, Exit, Read or Write.

Evidence grade
A for a published fact, B for a measurement on the reference suite, C for a declared assumption.

Parity licence
A licence to the source, or to self-host, priced below the licensee's ORC.

Token-priced, market-priced, not for sale
The three cost classes: falling with AI, moving slowly, and time-locked.

REFERENCES

Sources

Numbers match the superscripts in the text, in order of first appearance. Every figure in the worked example comes from the reference model, orc_model.py, which is attached to this PDF.

1 R. Pittman, "Against the Stand-Alone-Cost Test in U.S. Freight Rail Regulation", US Department of Justice. usdoj.gov/atr/public/eag/257888.pdf

2 Grain Processing Corp. v. American Maize-Products Co., 893 F. Supp. 1386 (N.D. Ind. 1995). law.justia.com/cases/federal/district-courts/FSupp/893/1386/1794994/

3 Cloudflare, "How we rebuilt Next.js with AI in one week", The Cloudflare Blog, February 2026. blog.cloudflare.com/vinext

4 Anthropic, "Building a C compiler with a team of parallel Claudes", Anthropic Engineering, February 2026. anthropic.com/engineering/building-c-compiler

5 Epoch AI, "MirrorCode: preliminary results", 2026. epoch.ai/blog/mirrorcode-preliminary-results

6 "AI decompilation and software copyright, part 1", hddlawfirm.com, 10 October 2026. For the Raymond and Heumann accounts. hddlawfirm.com/news/2026/10/10/compiled-code-used-to-keep-secrets-ai-just-taught-it-to-talk

7 The Pragmatic Engineer, "The Pulse: Cloudflare rewrites Next.js as AI rewrites commercial open source". blog.pragmaticengineer.com/the-pulse-cloudflare-rewrites-next-js-as-ai-rewrites-commercial-open-source/

8 M. Hoffmann, F. Nagle and Y. Zhou, "The Value of Open Source Software", 2024. papers.ssrn.com/abstract=4693148

9 Yahoo Finance, "What lies ahead for software stocks". finance.yahoo.com/technology/ai/articles/lies-ahead-software-stocks-etfs-110000811.html

10 P. Samuelson and S. Scotchmer, "The Law and Economics of Reverse Engineering", Yale Law Journal, 2002. yalelawjournal.org/index.php/article/the-law-and-economics-of-reverse-engineering

11 E. Mansfield, M. Schwartz and S. Wagner, "Imitation Costs and Patents: An Empirical Study", Economic Journal, 1981. doi.org/10.2307/2232499

12 Federal Reserve Bank of New York, Staff Report 44, which summarises the Mansfield figures. newyorkfed.org/medialibrary/media/research/staff_reports/sr44.pdf

13 G. Faulhaber, "Cross-subsidy analysis with more than two services", Journal of Competition Law and Economics, 2005, on his 1975 tests. econpapers.repec.org/RePEc:oup:jcomle:v:1:y:2005:i:3:p:441-448.

14 "Contestable market", Wikipedia, after W. Baumol, American Economic Review, 1982. en.wikipedia.org/wiki/Contestable_market

15 "COCOMO II model manual", University of Southern California. student.cs.uwaterloo.ca/~cs430/CII_modelman2000.0.pdf

16 QSM, "Function point languages table". qsm.com/node/68

17 US Bureau of Labor Statistics, "Software developers", Occupational Outlook Handbook. bls.gov/ooh/computer-and-information-technology/software-developers.htm

18 Directive 2009/24/EC on the legal protection of computer programs, Article 8. legislation.gov.uk/eudr/2009/24/article/8/data.pdf

19 Society for Computers and Law, "SAS Institute Inc v World Programming Ltd: report and analysis". scl.org/2451-sas-institute-inc-v-world-programming-ltd-report-and-analysis/

20 Directive (EU) 2016/943 on trade secrets, Chapter II. legislation.gov.uk/eudr/2016/943/chapter/II/adopted

21 18 U.S.C. 1839, definitions. law.cornell.edu/uscode/text/18/1839

22 "Bowers v. Baystate Technologies, Inc.", Wikipedia. en.wikipedia.org/wiki/Bowers_v._Baystate_Technologies,_Inc.

23 H. Meeker, "The chardet controversy: open source and the AI clean room", 9 April 2026. heathermeeker.com/2026/04/09/the-chardet-controversy-open-source-and-the-ai-clean-room/

24 "The clean room problem", Lexology. lexology.com/library/detail.aspx?g=3ef72a54-2b2f-4bd0-9072-b71fbf477c9b

25 COSMIC, "The basics of COSMIC sizing", on ISO/IEC 19761. cosmic-sizing.org/cosmic-sizing/intro/

26 Anthropic, "Claude API pricing", as an example of a public token price list. platform.claude.com/docs/en/about-claude/pricing

27 Hetzner, "Cloud", price list. hetzner.com/cloud

28 The MirrorCode paper, arXiv 2606.30182, 2026. arxiv.org/html/2606.30182

29 Turing, "Behavior2Code", benchmark page. labs.turing.com/benchmarks/behavior2code

30 "GameReplica", arXiv 2609.22308, 2026. arxiv.org/pdf/2609.22308

31 Gundlach, Lynch, Mertens and Thompson, "The Price of Progress", arXiv 2511.23455. arxiv.org/abs/2511.23455

32 METR, "Time Horizon 1.1", 29 January 2026. metr.org/blog/2026-01-29-time-horizon-1-1/

33 SQLite, "TH3", the SQLite test harness. sqlite.org/th3.html

34 COTBase, plans and product pages, read 13 November 2026. cotbase.com/

35 US Commodity Futures Trading Commission, "Commitments of Traders". cftc.gov/MarketReports/CommitmentsofTraders/index.htm

36 World Grain, "Twice-weekly COT reports coming". world-grain.com/articles/23385-twice-weekly-cot-reports-coming

37 Barchart, "Commitment of Traders charts". barchart.com/futures/commitment-of-traders