Games That Don't Die: Operator-Independent Competitive Infrastructure for Abandoned Multiplayer Games

Félix Francia
Independent researcher · Uruguay
Preprint, draft v0.2 · doi:10.5281/zenodo.23875636

Abstract

When an online multiplayer game loses its operator, communities can revive the servers, but its competitive life — rankings, match history, titles — dies with whoever kept the database. In June 2026 the European Commission declined to require publishers to keep games playable after support ends, leaving preservation to player communities. We argue that what those communities lack is not servers but *operator-independent competitive infrastructure*: a ranking and a history that no single party, including its founders, can corrupt or take down.

We present Proof of Battle, a design for such infrastructure, and study it in a real revival: Warband Premier, a competitive matchmaking network for *Mount & Blade: Warband* (2010) that went silent in May 2026 and was rebuilt as a network of community-hosted rooms feeding a single rating. The game client cannot be modified, and each match is reported by exactly one host who controls the server machine.

Our core technical contribution is **single-reporter truth discovery**: jointly estimating player skill and host reliability when no match is ever observed by more than one reporter. Identifiability comes instead from players who play across many hosts. We combine it with trust propagation over the match graph against Sybil farming, and with a survivability layer that commits the rating state to a public ledger so that anyone can rebuild it, and challenge it with a one-step fraud proof. Persistent achievements are issued as non-transferable (soulbound) credentials, and a per-match token is issued in proportion to the information each verified result adds to the rating. We prove that only relative host honesty is identifiable from such data, which yields a per-player honest-majority condition, a rating analogue of the 51% assumption of blockchains. On the network's 1,435 real matches with injected attacks, a lying host is caught in 97–100% of trials once it flips half of a target's losses, while plain Elo lets the target climb 10 to 20 places; without cross-host overlap, honest hosts are falsely accused in up to 31% of trials, as our non-identifiability result predicts. A joint skill-and-bias model under a median-like gauge removes half to three quarters of the rank a lie buys and flagged no honest host in 324 trials; trust propagation provably bounds what a farm of fake accounts can capture by its contact with real players, which held its token share to at most 3.2% in our runs against 17% without it.

Draft v0.2, November 2026: design, seven propositions and a lemma with proofs, and results on the full real match history. Code and data: github.com/felixfrancia27/games-that-dont-die.

1. Introduction

Online games die when their operator stops paying for servers. The Stop Killing Games campaign turned this into policy: its European Citizens' Initiative collected 1,294,188 verified signatures and was submitted on 26 January 2026. On 16 June 2026 the Commission concluded that a legal obligation to keep games playable would not be proportionate, and proposed instead a voluntary industry code for games at their "end of life" (Lewis Silkin; Flux). For the foreseeable future, games will outlive their publishers only if their communities keep them alive.

Communities already do this for servers. What they cannot easily preserve is the *competitive layer*. A ladder is a shared memory: who beat whom, who held which title in which season. It normally lives in one database run by one party. When that party leaves, the memory goes with it, and a revived community starts from zero. A recent formal model of player-base collapse states that "community-operated post-shutdown infrastructure" re-

quires modifying its axioms and is "outside the present scope", naming community-run private servers such as SWGEmu as future work (arXiv 2604.13390).

Moving the ranking to many community hosts creates the opposite problem. Every host controls its own machine and reports its own results. Nobody else observes the match. A host can invent matches, alter scores, or collude with players, and the classic remedy of asking several referees is unavailable.

We make four contributions:

1. **Problem.** We define *operator-independent competitive infrastructure* and its requirements: the ranking must survive the disappearance of any party, and must resist untrusted reporters without trusted hardware or client changes.
2. **Single-reporter truth discovery.** A model that jointly estimates player skill and host bias when each match has exactly one reporter. We prove that a single host's lies are unidentifiable, that only relative honesty across hosts is observable, and that detection succeeds with a bounded number of matches under cross-host overlap and a per-player honest majority.
3. **Survivability layer.** A public, deterministic match log with periodic commitments on an Ethereum layer-2, one-step fraud proofs, soulbound achievement credentials, and an information-proportional per-match token.
4. **Case study.** A real deployment on *Mount & Blade: Warband*, whose competitive network collapsed in May 2026 and relaunches on 1 December 2026, with an evaluation that replays its real history under injected attacks.

2. Background and related work

Each building block exists; none addresses a single untrusted reporter per event in a game that cannot be modified. The table places our work against each line (preliminary search, to be extended with ACM DL, IEEE Xplore and Scholar).

Line of work	What it offers	Gap for our setting
Game preservation (arXiv 2604.13390; Winget, *Digital Graveyards*)	Models of player-base collapse; legal remedies such as preservation patches	Community servers after shutdown fall outside the models; competitive state not addressed
Skill rating (Elo; TrueSkill; Elo-MMR)	Bradley–Terry style skill estimation; incentive-compatible variants	Outcomes assumed true
Crowdsourced ranking (Crowd-BT; factorBT; Agarwal et al. 2020)	Joint estimation of item strength and annotator reliability, including adversarial annotators	Relies on several annotators judging the same pair
Truth discovery (survey)	Source reliability without ground truth	Sparse sources named the hard case; one source per event unstudied
Graph-based Sybil defence (SybilBelief; trust leaks; Sybilproof reputation)	Trust propagation from honest seeds across sparse cuts	Applied to social and P2P graphs, not to rating updates
Result oracles (Chainlink; Elympics; witness patent)	Optimistic proposals with dispute windows; witness nodes	Trusted game server or official API assumed
Fraud proofs (Dave; incentive non-compatibility)	Cheap verification of off-chain computation	Verifier's dilemma when fraud is rare
Peer prediction (Robust BTS; elicitation without ground truth)	Truthful reports with $n \geq 3$ and no verification	Assumes reporters have no stake in the label; players do
Soulbound credentials (Weyl, Ohlhaver and Buterin 2022; Binance Research)	Non-transferable, publicly visible credentials	No rigorous study of skill credentials in games
Play-to-earn (SoK; Hype and Reality; empirical study)	Evidence on farming, Sybil airdrops, token collapse	Issuance tied to activity, not to information

FIDE has recorded chess ratings on Algorand since 2020, which shows ledgers can hold ratings but not that ratings can be made verifiable by replay. Riot Games detects boosting and win-trading in ranked play but does not publish its method (Riot).

3. Case study: Warband Premier

Warband Premier is a competitive matchmaking network for *Mount & Blade: Warband* (TaleWorlds, 2010) that collapsed and is being rebuilt as a network of community rooms. It is a natural laboratory: an old game whose client cannot change, a real history of seasons, and a documented death and revival.

The game. Warband is a medieval multiplayer game with dedicated servers and a server-side scripting system (modules). Competitive play is organised as team matches between players from the queue, with map vote and captain pick. The server can report events over HTTP, but nothing on the client can be changed or attested.

History. The network ran a beta and two seasons. Season 2 opened in April 2026 with little activity, and the last real match was played on 8 May 2026. The site then remained online with no games: the competitive layer had effectively died while its data survived only in one operator's database.

Revival as a network of rooms. The relaunch, scheduled for 1 December 2026, replaces central servers with *rooms*:

- any logged-in user creates a room on the website and receives a per-room key;
- the host installs a kit (TaleWorlds' official dedicated server, the Premier module, and a reporting agent) on their own PC or VPS;
- the room appears in a public server browser by region (SA, NA, EU, RU, AS, OC);
- every room feeds one global rating; the room owner moderates their room;

- an automatic integrity engine validates reported rounds, scores and connected players, voids suspicious matches without rating changes, and suspends rooms with escalating strikes;
- player identity in the game is bound to Warband's `unique_id` on first use.

This deployment already removes the operator from match hosting. It does not yet remove the operator from *truth*: the integrity engine and the rating database still run on one central API. The rest of this paper is about closing that gap.

In total the network recorded 1,725 matches by 140 players between 20 February and 8 May 2026, 1,435 of them with a winner, all on servers run by the operator.

4. System and threat model

We assume no trusted hardware, no client changes and no trusted host; we assume only that most matches are played honestly and that honest players play across several hosts.

Actors. *Players* hold an account bound to an in-game identity. *Hosts* run rooms and report each match (roster, rounds, score). The *coordinator* runs the website, queue and rating computation today. *Watchers* are anyone who downloads the public log and recomputes the state.

Adversaries.

Adversary	Capability	Goal
Lying host	Fabricates matches, alters scores or rosters; modifies its own agent and module	Raise chosen players, earn host rewards
Colluding players	Throw matches, win-trade, confirm false reports	Boost one account
Sybil farmer	Creates many accounts and rooms, plays among them	Mint tokens or rating from nothing
Interested witnesses	Players who confirm or dispute results in their own favour	Overturn losses
Vanishing or captured coordinator	Stops, loses data, or rewrites history	End or bias the ladder

Out of scope: in-game cheating by clients (aimbots and similar), denial of service, and attacks on the underlying ledger.

Design goals.

1. *Survivability*: the full competitive state can be rebuilt by anyone from public data if the coordinator disappears.
2. *Bounded distortion*: a lying host or Sybil cluster can move ratings and issuance by at most an amount bounded by its connection to the honest network.
3. *No-referee operation*: no step requires several independent reports of the same match.
4. *Low friction*: players need no wallet or per-match signature to play.
5. *Small-community robustness*: the mechanism must not punish a legitimate community of 50 players who mostly play each other.

5. Single-reporter truth discovery

The central claim: when no match has two reporters, a host's honesty can still be estimated from how its players perform everywhere else, provided players overlap across hosts. We combine three signals: host reliability,

match-graph trust, and testimony against interest.

5.1 Joint model of skill and host reliability

Each player i has latent skill s_i. For a match m between teams A and B hosted by h(m), the true outcome follows a Bradley–Terry model on team strength:

$$P(z_m = A) = \sigma\big(S_A - S_B\big), \qquad S_T = \tfrac{1}{|T|}\sum_{i \in T} s_i$$

The host reports y_m. With probability ρ_h it reports the truth; otherwise it reports in favour of a side chosen by its own (unknown) agenda, modelled per host as a bias toward specific players:

$$P(y_m = A \mid z_m) = \rho_h \, 1[z_m = A] + (1 - \rho_h)\, \sigma\big(b_{h,A} - b_{h,B}\big)$$

This is the Crowd-BT likelihood with one change that matters: every match has exactly one annotator. Parameters (s, ρ, b) are fitted jointly by EM or variational inference, with a prior on ρ_h from the integrity engine's per-room signals (voided matches, strikes, roster anomalies).

5.2 Identifiability through player overlap

Within a single host, a strong player and a lying host produce the same data. The confound breaks only when the same players also play elsewhere. For each host h we define its *overlap* as the share of its participants' matches played on other hosts. A leave-one-host-out estimate of skill then predicts h's outcomes; systematic deviation lowers ρ_h.

We now make this precise. For the analysis, a lying host h favouring player j adds a bias $\beta > 0$ to the log-odds of j's side in the matches it reports (a continuous version of the flip model above). Write x_m for the true log-odds of j's side in match m and n_T for the fixed team size.

Proposition 1 (no overlap, no identification). If every match of player j is reported by host h, then the parameters (s, β) and (s′, 0), where s′ equals s except $s'_j = s_j + n_T \beta$, induce the same distribution over all reports.

Proof. In a match containing j, raising s_j by $n_T \beta$ raises the mean strength of j's team by β, so the log-odds of j's side becomes $x_m + \beta$, exactly the biased report. Matches without j are unchanged. Since j plays nowhere else, no other term of the likelihood depends on s_j. The two parameter sets give identical likelihoods for every possible data set. ∎

So a host that lies for a player who plays only there is indistinguishable from an honest host with a stronger player. Any detector must then either miss the lie or accuse honest hosts of strong players at the same rate.

Proposition 2 (detection with overlap). Let host h report n matches containing j, with outcomes independent given skills. Suppose the skills estimated *without* h's reports predict each of these matches with error $|\hat{p}_m - p_m| \le \varepsilon$. Let $D = (1/n) \Sigma (y_m - \hat{p}_m)$, where $y_m = 1$ if j's side is reported as winner, and flag h when $D > \varepsilon + t$ with $t = \sqrt{(\ln(1/\delta) / 2n)}$. Then:

1. an honest host is flagged with probability at most δ;
2. a lying host with average gap $g = (1/n) \Sigma (E[y_m] - p_m)$ is flagged with probability at least $1 - \delta$ whenever

$$n \;\ge\; \frac{2\ln(1/\delta)}{(g - 2\varepsilon)^2}, \qquad g > 2\varepsilon.$$

Proof. Under honesty $E[D] \le \varepsilon$ and, by Hoeffding's inequality for n independent outcomes in [0, 1], $P(D - E[D] > t) \le \exp(-2nt^2) = \delta$. Under the attack $E[D] \ge g - \varepsilon$, so the test misses only if D falls below its mean by at least

g − 2ε − t. When the bound on n holds, g − 2ε ≥ 2t, so this deviation is at least t and has probability at most δ. ∎

Scanning K (host, player) pairs keeps the overall false-positive rate at δ by testing each at δ/K (union bound).

Bounding ε. Proposition 2 assumes the skill estimate computed without h predicts h's matches within ε. The next lemma reduces this to the accuracy of the skill estimate itself.

Lemma 1 (from skill error to prediction error). For any estimates ĉ and ŝ, write e = ŝ − s. In a match between teams of size n_T,

$$|\hat{p}_m - p_m| \;\le\; \tfrac{1}{4}\left(|\hat{c} - c| + \sqrt{2/n_T}\,\|e\|_2\right).$$

Proof. The logistic function is 1/4-Lipschitz, so $|\hat{p}_m - p_m| \le |\hat{\eta}_m - \eta_m| / 4$. The error in the log-odds is (ĉ − c) plus (1/n_T) times the sum of e_i over team A minus the sum over team B. Its absolute value is at most |ĉ − c| + (1/n_T) Σ $|e_i|$ over the $2n_T$ players of the match, and by Cauchy–Schwarz that sum is at most $\sqrt{(2n_T)}\,\|e\|_2$. ∎

For pairwise Bradley–Terry, Shah et al. (2016) bound $\|e\|_2$ by a quantity that shrinks with the number of comparisons and grows as the comparison graph's spectral gap closes. Extending that bound to team-mean designs is an open step we do not claim. It is also where overlap enters: once h's reports are removed, player j stays connected to the rest of the graph only through j's matches on other hosts, and with none the gap is zero.

Removing ε altogether: a model-free test. If rooms are assigned without regard to who plays, no skill model is needed.

Assumption A (random hosting). Match compositions are drawn independently; given the composition, the host that reports the match is chosen independently of it and of the true outcome.

Proposition 3 (model-free detection). Under Assumption A, let $\bar{W}_h$ be player j's win rate over the n_1 matches reported by h, and $\bar{W}_{-h}$ over the n_2 matches reported by other hosts, all honest. Flag h when $\bar{W}_h - \bar{W}_{-h} > t$, with

$$t = \sqrt{\tfrac{\ln(1/\delta)}{2}\left(\tfrac{1}{n_1} + \tfrac{1}{n_2}\right)}.$$

Then an honest h is flagged with probability at most δ, and a host that raises j's win probability by g on average is flagged with probability at least 1 − δ whenever

$$\frac{1}{n_1} + \frac{1}{n_2} \;\le\; \frac{g^2}{2\ln(1/\delta)}.$$

Proof. Under Assumption A and honesty, j's $n_1 + n_2$ win indicators are independent with the same mean, because the matches h reports are a random subsample of j's matches. The statistic $D = \bar{W}_h - \bar{W}_{-h}$ is a weighted sum of these indicators with weights $1/n_1$ and $-1/n_2$, whose squares sum to $1/n_1 + 1/n_2$. Hoeffding's inequality gives $P(D - E[D] \ge t) \le \exp(-2t^2 / (1/n_1 + 1/n_2)) = \delta$, and E[D] = 0 under honesty. Under the attack E[D] ≥ g; when the condition holds, g ≥ 2t, so a miss needs a deviation of at least t, which has probability at most δ. ∎

Overlap now appears explicitly: n_2 is the number of j's matches elsewhere. With $n_2 = 0$ the threshold is infinite and the test abstains, which is Proposition 1 again, except that the test *knows* it cannot decide and so never accuses. For a fixed total, $1/n_1 + 1/n_2$ is smallest when $n_1 = n_2$, so the best case for detection is a player who splits matches evenly across hosts.

On the real history (same setup as Section 9.1, scanning every host–player pair at 1% overall), the test made no false accusation in 180 attack-free trials at any level of overlap. It caught a host flipping all of a target's losses in 100% of trials for f ≤ 0.75 and 86% at f = 0.9, and abstained at f = 1. For half the losses, detection was 33%,

69%, 50%, 6% and 0% at f = 0.25, 0.5, 0.75, 0.9 and 1, peaking at the even split as the bound predicts. The price of needing no model is power: the model-based test caught 97–100% of those moderate lies. Assumption A fails when players choose rooms by strength; in deployment the test should compare within region and time window, or fall back to Proposition 2.

Corollary (undetectable lies are small). A lie that escapes the test has $g \le 2\varepsilon + 2t$, so it shifts j's expected win rate on h by at most that amount. Lies large enough to matter are the ones the test catches.

From one host to many: only relative honesty is observable. Proposition 1 generalises. In the joint model every (host, player) pair may carry its own bias $\beta_{h,j}$. Write the log-odds that team 1 wins match m as

$$\eta_m = c + \frac{1}{n_T} \sum_i \sigma_{m,i}\, s_i + \sum_j \sigma_{m,j}\, \beta_{h(m),j}$$

where $\sigma_{m,i}$ is +1 if player i is on team 1, −1 if on team 2, and 0 otherwise.

Proposition 4 (gauge invariance). For any player j and any constant c, replacing s_j by $s_j + n_T\, c$ and every $\beta_{h,j}$ by $\beta_{h,j} - c$ leaves every η_m, and hence the likelihood, unchanged.

Proof. In a match without j nothing changes. In a match with j, the skill term changes by $\sigma_{m,j}\, (n_T\, c) / n_T = \sigma_{m,j}\, c$ and the bias term by $-\sigma_{m,j}\, c$; the two cancel. ∎

So the data reveal only the *differences* $\beta_{h,j} - \beta_{h',j}$: how differently two hosts treat the same player. If every host j plays on favours j by the same amount, no method can tell, because that is observationally the same as j being stronger. Proposition 1 is the special case of a single host. Any decentralised rating must therefore *fix a gauge*: state which hosts define "honest" for each player.

Honest majority of matches. The natural gauge is the weighted median over hosts of $\beta_{h,j}$, with weights equal to j's number of matches on each host, set to zero.

Proposition 5 (a 51% bound for ratings). Suppose the relative biases are known exactly (the large-sample limit) and the median gauge is used. If honest hosts ($\beta_{h,j} = 0$) report strictly more than half of j's matches, the gauge assigns zero bias to them, and every lying host's bias is recovered exactly. If hosts reporting more than half of j's matches all favour j by the same β, the gauge instead absorbs β into j's skill, and the honest minority appears biased by −β.

Proof. The weighted median of a set of values is any value whose cumulative weight reaches one half from both sides. In the first case the value 0 carries more than half of the weight, so it is the unique weighted median and setting it to zero leaves each $\beta_{h,j}$ unchanged. In the second case β carries more than half of the weight, so the median equals β; fixing it to zero shifts every bias by −β and, by Proposition 4, adds $n_T\, \beta$ to s_j. ∎

This is the rating analogue of the honest-majority assumption of blockchains, but it is local: it is counted *per player*, over that player's matches. A coalition of hosts must capture most of a specific player's matches to rewrite that player's rating. In a network where players spread their matches across rooms, this is much harder than capturing a majority of all hosts.

Our implementation in Section 9 uses a ridge penalty on β, which amounts to a mean gauge rather than a median one. The mean is not robust: one liar shifts it by β divided by the number of hosts. A median (L1) gauge is the principled choice and is part of future work.

The consequence for small networks is direct: **with a single host, or hosts that share no players, Proposition 1 applies and trust reduces to the coordinator.** The method starts to work once two or more hosts share players. Section 9 measures how this plays out on real data.

5.3 Match-graph trust against Sybil farming

Islands are exactly what Sybil farms look like. We weight each match by trust propagated over the player–match graph from a seed set, using personalized PageRank in the style of SybilRank. Seeds grow in two phases. During the bootstrap months after relaunch, every account is a seed with uniform trust, including the **players with history before the collapse.** At that size the community knows its members, and fixed issuance (Section 8) caps what any farm can gain. Once public thresholds are met (for example, several independent hosts), trust weighting switches on, and the seeds become the bootstrap accounts that played enough matches against enough distinct opponents. The switch is auditable from the public log.

Trust and its bound. Trust is personalized PageRank with restart probability α on the bipartite graph of players and matches, restarting at the seed players. Player i's trust per match is p(i)/d(i), where d(i) is i's number of matches. A match's trust τ_m is the smallest per-match trust among its participants, normalised by the honest median and capped at 1. Repeated opponents are not penalised by themselves, which protects small honest communities (goal 5).

Proposition 6 (a farm's trust is bounded by its contact with the community). Let S be any set of nodes that contains no seed, for example a farm's accounts and the matches they play among themselves. Let F be the trust flowing into S along cut edges, the sum over edges (u, v) with u outside S and v inside S of p(u)/d(u). Then

$$p(S) \;\le\; \frac{1-\alpha}{\alpha}\,F,$$

and the total trust of the farm's matches, before normalisation, is at most p(S).

Proof. PageRank satisfies p = αs + (1 − α) pP, where s is the seed distribution and P moves from a node to each neighbour with probability 1/d. Summing over the nodes of S, and using s(S) = 0, the mass entering S from inside is at most p(S) and the mass entering from outside is exactly F. Hence p(S) ≤ (1 − α)(p(S) + F), which rearranges to the bound. For the second claim, each farm match contains some account i in S and has trust at most p(i)/d(i). Charging every match to one such account, the total is at most the sum over i in S of d(i) · p(i)/d(i), which is at most p(S). ∎

F grows with the number of cut edges, that is, with how many matches the farm plays against real players, and not with how many matches it plays among itself. Combined with part 4 of Proposition 7, a farm's token share is bounded by its contact with the honest community.

Trust gate for the ranking. Weighting is not enough for the ranking (Section 9.3): even low-weight farm matches still teach the model that one fake account beats the others. So a player appears in the public ranking only when their per-match trust reaches a threshold θ of the honest median. By Proposition 6, the number of farm accounts that can pass the gate is at most p(S) divided by θ times the median trust per match times the smallest number of matches a farm account plays.

5.4 Interested witnesses and testimony against interest

Players can confirm or dispute each result, but unlike crowd workers they care about the outcome. Peer prediction mechanisms assume reporters with no stake in the label, so we borrow a principle from evidence law instead: a *declaration against interest* carries more weight than a self-serving one.

- A confirmation from the **losing** side is strong evidence that the report is true.
- A dispute from the **winning** side is strong evidence that it is false.
- A dispute from the losers, or a confirmation from the winners, carries little weight alone.

Within each team, Robust BTS-style scoring can reward the testimony that agrees with teammates' predictions, which works with six or more reporters per side. Silence counts as weak confirmation only after a window, so

that the rating can be shown provisionally at once and settled later.

6. Survivability layer

The ledger's job is not to compute the ranking but to make it impossible to lose or rewrite: the coordinator becomes a replaceable convenience, like a block producer in an optimistic rollup.

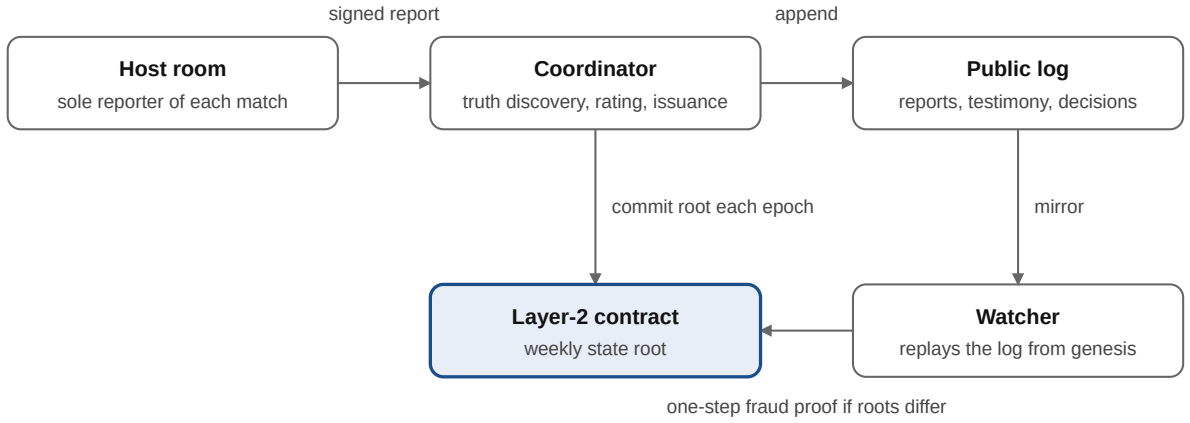


Figure 1. The survivability layer: one host reports each match, the coordinator computes and commits, and anyone holding the log can check and challenge.

The coordinator is the only box that can disappear without loss: the log and the committed roots let a watcher rebuild the state and take over.

Public log. Every host report, testimony and integrity decision is appended to a public, content-addressed log (match record, host signature with its room key, timestamps). Anyone can mirror it.

Deterministic replay. The rating and issuance rules of Sections 5 and 8 are published as code with fixed iteration order and fixed-point arithmetic. Replaying the log from genesis yields the same state on any machine. Inference that is not naturally deterministic (EM) is run with fixed seeds and a fixed number of iterations per epoch.

Periodic commitments. At the end of each epoch (one week), the coordinator posts a Merkle root of the state (ratings, host reliabilities, balances, credentials) and a hash of the log segment to a contract on an Ethereum layer-2 such as Base. Cost is a few cents per epoch, and no player needs a wallet.

One-step fraud proofs. Most of the epoch is a sequence of per-match updates. A watcher who disagrees with a posted root locates the first diverging match and submits it with Merkle proofs of the pre-state. The contract re-executes that single update, which is cheap enough on-chain. If the root was wrong, it is rejected and the challenger rewarded. The joint inference of Section 5.1 is too heavy for this; it is committed per epoch and challenged by bisection over iterations, as in interactive fraud-proof games (Dave).

The verifier's dilemma. If fraud never happens, nobody is paid to watch, and watching stops (arXiv 2312.01549). Our setting softens this: the people with the strongest motive to verify are players checking their own rating, and hosts checking their reliability score. We additionally plan a small standing reward for verified replays, funded from the common pool of Section 8.

Exit. If the coordinator stops, any party holding the log can resume from the last committed root, post new roots under a fresh coordinator key, and prove continuity. A future publisher shutdown, or our own, does not

erase the ladder.

7. Persistent identity: soulbound credentials

Achievements are issued as non-transferable credentials, because a competitive title that can be bought no longer certifies skill.

What is issued. Season placements, regional titles, clan championships, and milestones such as a first ranked match after the revival. Each credential is a leaf in the epoch's state tree (Section 6) and references the log segment that justifies it, so its evidence is checkable.

Claim without friction. Credentials exist from the moment they are earned, bound to the player's account. A player who later wants them on a public address claims them with a Merkle proof into a soulbound token, in the sense of Weyl, Ohlhaver and Buterin (2022): visible, non-transferable, revocable only by the issuing rule (for example, if a fraud proof voids the matches behind it).

Why not transferable NFTs. A rating is a claim about the person who holds it. If it can be sold, the claim breaks, and the market rewards farming and account trading, the failures play-to-earn studies document. Transferable items, if any, belong to cosmetics and stay outside the rating.

Portability. Because credentials reference a public log and an open rule set, other communities can read them. A player's history in one revived game can count as reputation in the next one, which is a small step toward the open ranking protocol discussed in Section 10.

8. Information-proportional issuance

Each epoch issues a fixed amount of a per-match token, split among verified matches in proportion to how much each one could teach the rating, so farming dilutes others instead of creating value.

Weight of a match. Before the result, the model predicts $\hat{p}_m = P(z_m = A)$. The match's informativeness is the binary entropy of that prediction, scaled by its trust weight from Section 5.3:

$$I_m = \tau_m \cdot \Big(-\hat{p}_m \log_2 \hat{p}_m - (1-\hat{p}_m)\log_2(1-\hat{p}_m) \Big)$$

A balanced match carries up to 1 bit; a mismatch carries almost nothing. The weight is fixed *before* the outcome, so neither side can raise it by choosing who wins.

Epoch emission. A fixed amount E_t is issued per epoch t and halves after a fixed number of cumulative verified matches, with a hard cap on total supply. Match m receives $E_t \cdot I_m / \Sigma I$. Inside a match, a starting split for discussion: winners 45%, losers 25%, host 20%, common pool 10%.

Host bond. The host's share accumulates as a bond, not a payout, and is slashed when a fraud proof or the reliability model (ρ_h) shows fabrication. Hosts build collateral by hosting, so opening a room needs no upfront stake.

Properties.

1. *No creation by farming*: total issuance per epoch is fixed, so fake matches only dilute.
2. *Outcome neutrality*: a match's total share does not depend on the reported result.
3. *Lopsided matches pay little*: smurfing against weak players earns near zero.
4. *Sybil bound*: a farm's share is bounded by its trust, hence by its cut to the honest graph.

Proposition 7 (issuance). Let W be the total weight $\Sigma\, I_m$ of the honest matches of an epoch, and let an attacker add matches F with total weight W_F. Then:

1. total issuance in the epoch is E_t whatever F is, and the attacker receives at most a share $W_F / (W + W_F)$;
2. a match's total share depends only on $\hat{p}_m$ and τ_m, not on the reported result;
3. a match predicted at $\hat{p} \le 1/2$ for the weaker side has weight at most $\tau_m \cdot \hat{p} \log_2(e / \hat{p})$, for example 0.29 bits at $\hat{p} = 0.05$ against 1 bit for an even match;
4. since $I_m \le \tau_m$, the attacker's weight W_F is at most the sum of the trust of its matches, so any bound on a farm's trust (Section 5.3) bounds its issuance.

Proof. (1) Match m receives $E_t\, I_m / \Sigma\, I$, and these shares sum to E_t; the attacker's matches sum to $E_t\, W_F / (W + W_F)$. (2) I_m is computed from $\hat{p}_m$ and τ_m, both fixed before the result is reported. (3) For the binary entropy, $(1 - p) \ln(1/(1 - p)) \le p$, because $-\ln(1 - p) \le p / (1 - p)$; hence $H(p) \le p \log_2(1/p) + p \log_2 e = p \log_2(e/p)$. (4) $H(p) \le 1$ for every p. ∎

Part 2 matters most: lying about the winner can move tokens *between* the two sides of a match, which the rating model of Section 5 polices, but it cannot increase what the match issues in total.

Transferability. The balance is non-transferable at first and could become transferable only after public maturity thresholds (number of independent hosts and active players), subject to legal review. The paper's claims do not depend on the token having market value.

9. Evaluation

We evaluate on two fronts: a replay of the network's real match history with injected attacks, and a longitudinal study of the revival from 1 December 2026.

Data. Every decided match of the beta, Season 1 and Season 2, in order, with teams and winner only (Section 9.1). The live network after relaunch adds real community hosts. The project's emulator (`wb sim`) and chaos tester generate synthetic matches and adversaries at known ground truth.

Attacks injected.

Attack	Injection	Expected defence
Lying host	One host flips a share of its results toward chosen players	ρ_h falls through cross-host overlap (5.2)
Ghost matches	A host reports matches that never happened among its regulars	Low trust and no testimony against interest
Sybil farm	k new accounts and rooms play only among themselves	Trust bounded by cut edges (5.3); issuance diluted (8)
Win-trading	Two accounts alternate wins	Near-zero information per match after the first few
Smurfing	Strong player on a new account against weak ones	Lopsided matches pay little
Coordinator rewrite	Altered history after commitment	Fraud proof against the committed root (6)
Coordinator exit	Coordinator stops	Third party rebuilds state from log and last root

Baselines. Plain Elo, TrueSkill, Crowd-BT with one annotator per match, and the current integrity engine alone.

Metrics. Rating distortion (rank correlation and mean absolute error against the clean replay); time and number of matches until a lying host is detected; share of issuance captured by attackers; false positives on honest hosts

and on the small honest community; cost per epoch on the layer-2; time for a third party to rebuild the state.

Hypotheses.

1. With two or more overlapping hosts, the joint model detects a host flipping 20% of its results within a bounded number of matches, while plain Elo absorbs the distortion.
2. A Sybil farm captures issuance proportional to its cut edges, not to its match count.
3. Honest small communities are not flagged at a higher rate than large ones.
4. Anyone can rebuild the full ladder from public data alone.

Revival study. From relaunch, we measure the number of independent hosts, cross-host overlap over time (the quantity that makes the method work), retention, disputes, and player attitudes toward the ledger through a short survey.

9.1 Preliminary results on real matches

On the network's full history of 1,435 decided matches, a lying host was caught in 97–100% of trials once it flipped half of a target's losses. Without overlap, honest hosts of strong players were falsely accused in up to 31% of trials, as Proposition 1 predicts.

Data. The production database holds 1,725 matches by 140 players (20 February to 8 May 2026). Of these, 1,435 have a winner: 385 in the beta, 836 in Season 1 and 214 in Season 2. The rest were voided or drawn. Player identifiers were replaced by salted hashes inside the server, so no raw identifier left production.

The real network is small and dense. The median player played 35 matches and faced 55 distinct opponents. 3,479 pairs of players met as opponents, a median of 6 times each, and one pair met 146 times. The top 10% of players fill 38% of all match slots. A rule that penalised repeated opponents would punish this entire legitimate community, which is why Section 5.3 normalises by community size instead.

Setup (semi-synthetic). All historical matches ran on the operator's own servers, so hosts are simulated. Each match is assigned to one of 4 hosts. Host 0 lies for a target player: a share f of the target's matches goes to host 0, and it flips each of the target's losses into a win with probability λ. Real outcomes serve as ground truth. Targets are the 12 most active players (374 to 605 matches each), 3 seeds each, so 36 trials per cell. Detection uses the leave-one-host-out test of Proposition 2, scanning every (host, player) pair with at least 5 matches at an overall level of 1% (Bonferroni), then refitting while excluding hosts already flagged. With 36 trials, a rate near 50% has a 95% binomial interval of about ±16 points, so small differences between rows are within sampling noise.

Share f of target's matches on the lying host	Caught, λ = 0.25	Caught, λ = 0.5	Caught, λ = 1	Honest host falsely accused, no attack
0.25	22%	97%	100%	0%
0.50	58%	100%	100%	0%
0.75	75%	100%	100%	3%
0.90	92%	100%	100%	28%
1.00	92%	100%	100%	31%

Readings.

- *Plain Elo absorbs the lie.* Replaying the same reports through team Elo (K = 32), the attack lifted targets by 10 to 20 ranking places on average, depending on f and λ.

- *Overlap buys fairness, not power.* Detection of the lie barely depends on overlap: concentrating the target's matches on the liar gives the test more evidence. What overlap controls is false accusation. With no attack, an honest host of a strong player was flagged in 0–3% of trials when most of that player's matches were elsewhere, and in 31% when all of them were on that host. That is Proposition 1 measured on real data: without overlap, real skill looks like a lie.
- *Calibration.* In a separate null run with no attack (60 trials), the test flagged any host in 3.3% of trials at f = 0.25 against a nominal 1%, and in 23% at f = 1; the test is slightly anti-conservative on real data even with overlap, which we attribute to model misspecification (real matches are not independent draws from a fixed Bradley–Terry model).
- *Contamination.* Under the strongest attack, the liar's reports distort the skills used to test other hosts: honest hosts were flagged in up to 25% of trials. Refitting without flagged hosts lowered this at the strongest attacks (25% to 14%) but did not help in general: across settings, 3–17% of trials still flagged an honest host after refitting, partly because each refit uses less data. This motivates the full joint model of Section 5.1, which down-weights a host by its estimated reliability instead of dropping it.

Design implication. A deployed system should never punish on the test alone when a player's matches are concentrated on one host. It should first measure that player's overlap, and treat low-overlap flags as requests for more evidence, such as matches on other hosts or testimony against interest.

9.2 The joint model: protecting the ranking, not only flagging the host

With a median-like (L1) gauge, the joint model cut the rank a lie buys from 12–21 places to 4.5–6 whenever the target also played elsewhere, and it never flagged an honest host. In preliminary runs not included in the released code, a version with a ridge penalty (a mean gauge) failed, as Proposition 5 predicts: at f = 0.25 it caught 25% of moderate liars and left the rank gain at 14–20 places, no better than plain Elo, because it absorbed the lie into the target's skill.

Setup. Same data, hosts and attacks as in 9.1. The model fits player skills and a bias for every (host, player) pair with at least 5 matches. Biases carry an L1 penalty (iteratively reweighted least squares), which drives most of them to zero and so approximates the honest-majority gauge. A host is flagged when one of its biases exceeds a Bonferroni threshold at 1% overall. "Rank gain" is how many places the target rose compared with a fit on clean data; 36 trials per row.

f	λ	Leave-one-out test: liar caught / honest host flagged	Joint model: liar caught / honest host flagged	Rank gain, Elo	Rank gain, Bradley–Terry	Rank gain, joint model
0.25	0	0% / 3%	0% / 0%	0	0	1.5
0.25	0.5	78% / 3%	0% / 0%	12.4	12.3	6.0
0.25	1	100% / 11%	97% / 0%	18.9	19.2	5.9
0.50	0	0% / 0%	0% / 0%	0	0	1.1
0.50	0.5	100% / 3%	0% / 0%	18.8	19.1	4.5
0.50	1	100% / 19%	100% / 0%	20.3	20.5	4.6
1.00	0	28% / 3%	3% / 0%	0	0	−5.5
1.00	0.5	100% / 3%	8% / 0%	20.3	20.4	0.4
1.00	1	100% / 3%	100% / 0%	20.3	20.8	2.5

Rows with λ = 0 have no attack: there, "liar caught" means host 0 was falsely flagged.

Readings.

- *The two methods do different jobs.* The leave-one-out test is the better alarm: it catches moderate lies (λ = 0.5) that the joint model's bias statistic misses. The joint model is the better *rating*: it removes between half and three quarters of the rank a lie buys and never flagged an honest host in 324 trials.
- *The mean gauge fails, the median gauge works.* This is Proposition 5 in practice. Under a mean gauge the lie is split between the host's bias and the target's skill; under a median-like gauge the honest majority of the target's matches pins the skill and the lie stays with the host.
- *Without overlap, someone pays.* With every match of a strong player on one honest host (f = 1, no attack), the leave-one-out test accused the host in 28% of trials, while the joint model lowered the honest player by 5.5 places instead. Proposition 1 says some error is unavoidable here; the design choice is who bears it.
- *Residual noise.* Without any attack the joint model moves targets by 1.1 to 1.5 places relative to a plain fit, the price of letting every host carry its own bias.

Recommended design. Rate with the joint model under a median gauge; raise alarms with the leave-one-out test; and treat any alarm on a player with low cross-host overlap as a request for more evidence, never as a sanction.

Code: `src/joint.py`; results in `results/joint.json`.

9.3 Sybil farms

A farm of 12 fake accounts captured 17% of tokens under per-match issuance and 19% under information weighting alone, but at most 3.2% once trust was applied, and no fake account passed the ranking's trust gate.

Setup. On top of the 1,435 real matches, 12 fake accounts play 300 matches among themselves. In the *token* attack they pick results at random, which keeps every match maximally informative. In the *rank* attack one account always wins. They also sneak into a matches of the real history by replacing one real participant (attack matches, the cut edges of Proposition 6). The real players are the seeds, as in the bootstrap phase of Section 5.3; α = 0.15; 3 seeds per row.

Attack matches with real players	Farm token share: per match	Information only	Trust × information	Farm trust mass / bound of Prop. 6	Fake account's rank: Elo	Trust-weighted rating	Fake account's trust vs honest median
0	17.3%	18.8%	0.00%	0 / 0	1	unrated	0
10	17.3%	18.8%	0.14%	0.0014 / 0.0014	1	35	0.7%
50	17.3%	18.8%	0.77%	0.0064 / 0.0067	1	7	3.4%
200	17.3%	18.8%	3.24%	0.0238 / 0.0278	1	2	11%

Readings.

- *Information weighting alone makes farming worse.* A farm that randomises its results produces balanced, maximally informative matches. Proposition 7's outcome neutrality protects the total, but only trust protects against farms.
- *The bound holds and is tight.* The farm's trust mass stays under the bound of Proposition 6 in every run, within 15% of it. The token share grows with the number of matches played against real players, not with the 300 matches played inside the farm.
- *Weighting cannot protect the ranking; the gate can.* With trust-weighted matches the always-winning account still climbed to 2nd place at 200 attack matches. Its trust, however, never exceeded 11% of the honest

median, so a gate at $\theta = 0.5$ kept every fake account out of the ranking while excluding none of the real players with at least 10 matches.

- *Collateral.* Real matches that a fake account joined inherit its low trust: 0.7%, 3.5% and 14% of real matches at 10, 50 and 200 attack matches, exactly the infiltrated ones. Players in those matches lose some issuance, which gives the community a direct reason to report infiltrators.

Code: `src/sybil.py`; results in `results/sybil_*.json`.

Code and data availability. All code and the anonymised match data are public at github.com/felixfrancia27/games-that-dont-die. One command (`./run_all.sh`) regenerates every table in this section; the numbers reported here are exactly those it produces.

10. Discussion and limitations

The design shifts what must be trusted from one operator to a property of the community, cross-host overlap, and is only as strong as that property.

Policy. The Commission's voluntary end-of-life code is being drafted with industry and consumer groups through the end of 2026. Operator-independent competitive infrastructure is a concrete, low-cost practice such a code could recommend: publish the match log and rating rules at shutdown, so a community can carry the ladder forward.

Generalisation. Nothing in Sections 5 to 8 is specific to Warband. Any game with dedicated servers that can report results fits, which suggests an open ranking protocol shared across revived classics.

Limitations.

- *Small networks.* With one host, or hosts that share no players, truth discovery is impossible and trust falls back to the coordinator (5.2). At launch, most rooms may belong to the founders, so decentralisation is a trajectory, not a starting state.
- *Skilled lying hosts.* A host controlling its machine can fabricate results consistent with players' skill. Such lies are hard to detect, but they also move ratings little, which bounds their damage.
- *No replays.* Warband does not record matches, so disputes rest on testimony and statistics, not evidence.
- *Determinism versus inference.* Replaying statistical inference bit-for-bit across machines needs fixed-point arithmetic and fixed iteration budgets; this constrains model choice.
- *Legal risk.* A transferable token can be treated as a financial asset or raise anti-money-laundering duties depending on jurisdiction; we keep balances non-transferable and require legal review before any change.
- *Community acceptance.* Part of the gaming community rejects cryptocurrency. The design keeps the ledger invisible to players who do not care, with no wallet needed to play.
- *Novelty claims.* Our literature search is preliminary; claims of novelty are "to our knowledge" until a systematic review is done.

11. Conclusion

A game survives its publisher only if its community can carry both its servers and its memory. We proposed infrastructure that makes a competitive ladder outlive any operator: truth inferred from untrusted single reporters through cross-host overlap and match-graph trust, a state anyone can rebuild and challenge, credentials that cannot be bought, and issuance that rewards information rather than activity. The revival of Warband Premier

on 1 December 2026 lets us test it on a real community, starting from the hardest case: a small network that must grow into its own decentralisation.

References

- Agarwal et al. Rank Aggregation from Pairwise Comparisons in the Presence of Adversarial Corruptions. ICML 2020.
- Chen, Bennett, Collins-Thompson, Horvitz. Pairwise Ranking Aggregation in a Crowdsourced Setting. WSDM 2013.
- Aggregation of pairwise comparisons with reduction of biases (factorBT). arXiv 1906.03711.
- A Survey on Truth Discovery.
- SybilBelief. arXiv 1312.5035.
- Minimizing Trust Leaks for Robust Sybil Detection. ICML 2017.
- Sybilproof reputation mechanisms.
- Ebtekar, Liu. Elo-MMR. WWW 2021.
- Nehab, Coutinho de Paula, Teixeira. Dave: a decentralized, secure, and lively fraud-proof algorithm. arXiv 2411.05463.
- Incentive Non-Compatibility of Optimistic Rollups. Landis, arXiv 2312.01549, 2023.
- Witkowski, Parkes. A Robust Bayesian Truth Serum for Small Populations. AAAI 2012.
- Mechanisms for belief elicitation without ground truth. Lehmann, arXiv 2409.07277, 2024.
- Negahban, Oh, Shah. Rank Centrality: Ranking from Pairwise Comparisons. Operations Research, 2017.
- Shah et al. Estimation from Pairwise Comparisons: Sharp Minimax Bounds with Topology Dependence. JMLR, 2016.
- Hoeffding. Probability Inequalities for Sums of Bounded Random Variables. JASA, 1963.
- Nakamoto. Bitcoin: A Peer-to-Peer Electronic Cash System. 2008.
- SWGEmu Project. Core3. 2011–present.
- Weyl, Ohlhaver, Buterin. Decentralized Society: Finding Web3's Soul. 2022.
- Soulbound Tokens. Binance Research.
- SoK: Play-to-Earn Projects. Yu, Zhang, Chen, Fang, arXiv 2211.01000, 2022.
- Gaming and Blockchain: Hype and Reality. arXiv 2407.12134.
- NFT Games: an Empirical Look into the Play-to-Earn Model. Gao et al., arXiv 2602.13882, 2026.
- A Formal Framework for Critical-Mass Collapse in Online Multiplayer Games. Sheta, arXiv 2604.13390, 2026.
- Winget. Digital Graveyards. N.C. J. Law & Technology 27(4).
- European Commission response to Stop Destroying Videogames. Lewis Silkin, June 2026.
- Elympics In-Game Oracle; Chainlink: results data; USPTO 10572872.