YOM

Infrastruttura GPU edge decentralizzata

GPU accanto a ogni utente, pagate per sessione o a ore

Documentazione tecnica e whitepaper:
la rete, il protocollo e come guadagnano
gli operatori.

V1.0 · 21 NOVEMBRE 2026 · YOM.NET/DOCS

Indice

PROBLEMA

L'endgame post-cloud

Origini e visione

SOLUZIONE

Come funziona

Blockchain

Rete

Operatori

Pagamenti

GUADAGNA

NANO

Licenze dei nodi

Delega NaaS

Massimizzare i pagamenti

Referral

COSTRUISCI

Panoramica

Avvio rapido

Integrazione JS

- Autenticazione personalizzata
- Gestione degli eventi
- Layout dinamici
- Hook UI semplici
- Configurazione multi-stream

INFORMAZIONI

Allocazione dei token

Roadmap

Social media

FAQ

Chiarimento sui crediti del Compute Pass

SEZIONE

problema

l'endgame post-cloud

La narrazione secondo cui servono *più* data center è una bugia.

Il mondo non soffre di mancanza di potenza di calcolo. Sta annegando in potenza di calcolo allocata male. La capacità grezza necessaria per far girare l'AI live e la prossima generazione di giochi AAA esiste già. Gran parte di essa è spenta, nelle case delle persone.

YOM non sta costruendo un data center migliore. Stiamo costruendo il livello che rende il data center opzionale.

Il gigante dormiente: la capacità inattiva nelle case

La maggior parte della potenza delle GPU consumer del mondo resta inattiva per gran parte della giornata.

Milioni di RTX 3090, 4080 e di rig da gaming di fascia alta fanno girare schede del browser o nulla. È uno dei più grandi bacini di calcolo del pianeta, e i provider di cloud centralizzato lo trattano come se non esistesse. Noi lo trattiamo come il lato dell'offerta.

Attivare quell'hardware non *aggiunge* capacità. Mette al lavoro una risorsa già prodotta, già distribuita e già pagata.

E le GPU consumer non sono l'unico lato dell'offerta. Le GPU nei tower site delle telecomunicazioni aggiungono una copertura edge di livello carrier dove serve. YOM ha gestito 168 sessioni di gioco live in 20 giorni su tower site di Cornerstone, la più grande società di torri mobili del Regno Unito (oltre 16,500 siti), nata come joint venture di condivisione di rete tra Vodafone e O2, con StonesThro e Zadara. Una topologia ibrida, non un marketplace a fonte unica.

Il pavimento economico che il cloud centralizzato non può superare

Per scalare, un hyperscaler deve spendere. Raddoppiare la capacità significa versare miliardi in silicio, cemento, terreni e raffreddamento, e quel mutuo fissa un pavimento rigido sotto i suoi prezzi. Un'ora su una GPU da data center riservata per il picco costa circa US$1.23, perché qualcuno deve pagare l'edificio e le ore in cui resta inattivo.

YOM non ha un edificio.

Non compriamo hardware. Non versiamo cemento. Orchestriamo un protocollo su hardware che esiste già, il che porta una sessione Standard fino a un'ora a US$0.08. Non è uno sconto. È una struttura di costo diversa, e matematicamente fuori portata per chiunque porti sulle spalle il capex dei data center.

Sconfiggere la velocità della luce

Anche il cloud centralizzato combatte contro la fisica, e perde.

Non si può ottimizzare la velocità della luce. Se il server è a mille chilometri di distanza, la latenza è incorporata, e nessuna fibra può cambiare il fatto che la macchina è semplicemente troppo lontana dal giocatore. YOM risolve la latenza ignorando la geografia: ogni PC da gaming della rete è un potenziale nodo edge, il che sposta il calcolo nel quartiere stesso dell'utente. Il segnale percorre la strada invece di attraversare il paese, e la rete punta a una latenza mediana sotto i 10 ms con nodi entro ~50 km dal giocatore.

È la strada verso la reattività di una console locale: prossimità invece di forza bruta, con la latenza misurata pubblicata dal vivo su source of truth.

Cosa resta scarso quando tutto il resto va a zero

Il cambiamento più profondo è che il software stesso sta perdendo valore. Chiunque abbia un modello di frontiera può generare la propria app in un fine settimana. Quando tutto può essere generato, l'intelligenza diventa abbondante e il software costa poco. Restano scarse le cose che non si possono far esistere con un prompt: **latenza, località, energia, hardware e presenza fisica.**

È il terreno su cui poggia YOM. Possedere l'informazione conta meno quando generare costa poco; far funzionare sistemi a bassa latenza vicino al giocatore conta di più. L'economia che ne deriva premia chi contribuisce davvero in luoghi reali, non monopoli lontani. Miner, operatori retail e telco contribuiscono sullo stesso substrato. La rete è aperta a chiunque abbia una macchina idonea.

Il cambiamento è strutturale

Stiamo assistendo al passaggio da piattaforme di rendita a reti partecipative.

- **Vecchio mondo:** paghi una corporation per affittare un computer lontano. La corporation si tiene il margine; il pianeta paga per nuova produzione.
- **Nuovo mondo:** paghi un pari per un computer vicino. Chi contribuisce si tiene la ricompensa; non serve costruire nulla di nuovo.

YOM è l'infrastruttura per quella transizione. Partiamo dal cloud gaming perché il gaming è il problema di streaming in tempo reale più difficile che esista: latenza ultra bassa, fedeltà AAA e picchi serali in cui tutti i giocatori arrivano insieme. Risolvi il gaming e la stessa rete serve conversazioni AI live, prima di tutto la voce, sugli stessi nodi e nelle stesse sessioni.

Il primo passo è il gaming. Il secondo è l'AI live.

origini e visione

YOM non è nata in una sala riunioni. È nata in produzione. Il nostro fondatore Jorrit Velzeboer dirigeva un'agenzia di streaming 3D, dove ha toccato con mano il costo dello streaming di carichi GPU sul cloud centralizzato: nonostante un'ottimizzazione estrema, ogni ora di utente portava con sé il prezzo di una GPU da data center, tenuta lontano dall'utente e pagata anche mentre aspettava il picco. Il cloud centralizzato è stato costruito per i carichi di lavoro aziendali, non per lo streaming interattivo di massa. Così ci siamo messi a risolvere un problema che gli hyperscaler non avevano alcun motivo di risolvere.

La cronologia

- **2021:** Costruito un backend di pixel streaming su misura. L'MVP ha dimostrato che le GPU consumer vicine al giocatore possono trasmettere giochi in streaming, l'intuizione su cui si regge l'intera azienda.
- **2022:** Concluso che solo una rete di GPU di proprietà di chi le gestisce scala in modo economicamente efficiente. Rohan Solanki entra per progettare HyperOrch, l'orchestratore edge di YOM, e NodeOS, il sistema operativo dei nodi.
- **2023:** Vinto il Seal of Excellence della Commissione Europea per l'innovazione deep-tech. Prime conversazioni go-to-market con Andrew Pringle e Jeff Outlaw.
- **2024:** Formalizzato il team di leadership con veterani del gaming. L'alpha di HyperOrch trasmette la sua prima demo UE5. Round Pre-Seed con Outlier Ventures e Borderless Capital.
- **2025:** Rilasciato il primo lotto di NANO. Best Tech Startup, Gamescom 2025.
- **2026:** Token e mainnet attivi a giugno. Walter trasmette in streaming i giochi di Steam ed Epic che i giocatori già possiedono. Oltre 46,000 account registrati. Sui tower site di Cornerstone, YOM ha gestito 168 sessioni di gioco live in 20 giorni. La rete in produzione pubblica dal vivo, su source of truth, la latenza mediana per i giocatori entro 50 km da un nodo; sotto i 10 ms è l'obiettivo.

Una rete di proprietà di chi la gestisce

Stiamo costruendo verso un'economia della partecipazione in tempo reale, non verso un'economia di credenziali astratte o di intelligenza centralizzata.

Mentre internet evolve da un **internet dei file** a un **internet degli eventi**, il valore si sposta con esso. L'intelligenza diventa abbondante; il software costa poco. Restano scarse le cose che l'astrazione non può cancellare: **latenza, località, energia, hardware e presenza fisica.** In quel mondo il potere non appartiene più a chi possiede l'informazione. Appartiene a chi sa far funzionare sistemi live vicini al tempo umano, entro i limiti ferrei della fisica e della percezione.

YOM esiste per un mondo in cui:

- **Le persone sono operatori e proprietari** dell'infrastruttura che fa girare l'esperienza digitale, non soltanto suoi utenti.
- **Il valore va a chi contribuisce, non alla piattaforma.** Sostituiamo la dipendenza da monopoli lontani con un'economia diretta che premia le persone per l'hardware, l'energia e la cura che mettono a disposizione.
- **La base di chi contribuisce è aperta.** Miner, operatori retail e telco fanno girare nodi sullo stesso substrato. La rete è aperta a chiunque abbia una macchina idonea.
- **La realtà digitale resta ancorata:** un sistema che deve essere gestito, misurato, verificato e pagato da persone reali che contribuiscono in luoghi reali, non uno spettacolo da consumare.

Non è teoria. Un PC host guadagna trasmettendo potenza di calcolo ai giocatori vicini. Uno stream Standard guadagna circa ~US$53 al mese all'obiettivo di utilizzo della rete del 50%; lo stimatore di backend calcola la capacità supportata da ciascun host a partire dal suo hardware misurato (cifre live su /operators).

La missione

Trasformare l'hardware reale, di proprietà locale, in infrastruttura economica attiva.

Coordiniamo una rete globale di dispositivi edge di proprietà della community, orchestrati da HyperOrch, il nostro orchestratore edge, e ricompensati tramite la condivisione dei ricavi con gli operatori, per farne un'infrastruttura che diventa più distribuita man mano che cresce. L'opposto del cloud centralizzato.

Partiamo dal cloud gaming perché è il problema di streaming in tempo reale più difficile che esista: latenza ultra bassa, fedeltà AAA e picchi serali in cui tutti i giocatori arrivano insieme. Il pixel streaming vive in modo nativo sull'edge, e la località è tutto il gioco. Risolto quello, la stessa rete serve conversazioni AI live, prima di tutto la voce, e ogni altro carico di lavoro in tempo reale in cui i data center centralizzati toccano il loro tetto.

Esistiamo per:

1. Mantenere **economicamente prezioso l'agire umano** in un mondo di cognizione automatizzata.
2. Convertire **l'hardware inattivo in capacità produttiva pagata e verificabile.**
3. Ancorare le economie digitali a **località e uptime**, non a una concentrazione di capitale lontana.

Leadership

Uniamo l'ingegneria deep-tech all'esperienza commerciale nel gaming AAA.

- **Jorrit Velzeboer (Founder & CEO):** Imprenditore seriale ed ex venture capitalist. Ha progettato il modello di business e la direzione tecnica di YOM.
- **Jeff Outlaw (Co-Founder & COO):** Ex Product Manager di Oculus VR. Veterano dei giochi che collega YOM agli studi e alle community di giocatori.
- **Rohan Solanki (Co-Founder & CTO):** Architetto di HyperOrch e NodeOS. Specialista di sistemi distribuiti su larga scala.
- **Andrew Pringle (Co-Founder & CCO):** Decenni di strategia retail, commerciale e distributiva.
- **John Kavanagh (Business Development Director):** Oltre 20 anni di relazioni nel mondo della proprietà intellettuale e dell'intrattenimento. Guida la domanda di publisher e wholesale.
- **Jeroen Elout (CXO):** Specialista di crescita e user experience; segue conversione e retention per Walter.

Consigliati dall'ex COO di OnLive e CEO di Eidos Interactive, dall'ex responsabile globale di PlayStation Store e dal CEO di PLAYERUNKNOWN Productions.

Stiamo costruendo l'infrastruttura del futuro, ancorata ai valori umani.

SEZIONE

soluzione

come funziona

Far funzionare una rete edge decentralizzata per carichi di lavoro interattivi richiede tre cose che lavorano insieme: uno stack tecnico serio, un'economia on-chain e un'esperienza che per giocatori e sviluppatori risulti senza sforzo.

YOM ha costruito la propria orchestrazione, il proprio sistema operativo dei nodi e il proprio stack di streaming per rendere lo streaming sicuro, efficiente e facile da usare; la tecnologia è in attesa di brevetto. Il prossimo capitolo descrive l'architettura.

Quando clicchi «Play»

Dal punto di vista del giocatore non c'è nessun download e nessuna installazione. Clicchi un link e nel giro di pochi secondi stai giocando, con il rendering eseguito da remoto e lo streaming sul tuo browser o dispositivo. Per i giocatori questo significa:

- Giocare subito, anche con hardware di fascia bassa.
- Giocare su Walter ai giochi che già possiedi da US$4.99 al mese.
- Bassa latenza grazie alla collocazione: un obiettivo di latenza mediana sotto i 10 ms per i giocatori entro 50 km da un nodo, misurata dal vivo su source of truth.

Per studi e publisher, eliminare installazioni, patch e limiti hardware apre pubblici molto più ampi e una conversione più alta.

Cosa succede davvero

A livello tecnico, ecco cosa succede quando premi «Play»:

1. Il giocatore richiede un gioco tramite una web app, un link incorporato o un portale partner.
2. HyperOrch, il nostro orchestratore edge, invia la sessione al nodo più vicino che ha la capacità GPU richiesta e spazio libero.
3. Quel nodo, un PC da gaming avviato con NANO o un rack telco che esegue lo stesso software di nodo, lancia il gioco in un container nuovo, lo codifica con l'encoder hardware della GPU (NVENC) e lo trasmette via WebRTC, così il giocatore non installa nulla.
4. Il gameplay viaggia peer-to-peer da un nodo vicino al giocatore, entro circa 50 km dove la rete ne ha uno.
5. Quando la sessione termina, i ricavi vengono divisi automaticamente:
    - 60% va all'operatore del nodo
    - 30% finanzia lo sviluppo e la crescita della rete
    - 5% è condiviso con tutti coloro che conservano il proprio pagamento
    - 5% viene ritirato, riducendo la supply di token
6. Ogni sessione viene regolata tramite smart contract.

La YOM Foundation gestisce il settlement e governa il programma che fa crescere la supply, e passa il controllo alla community per gradi: prima i voti dei detentori di token sul programma della supply, poi i membri del consiglio eletti.

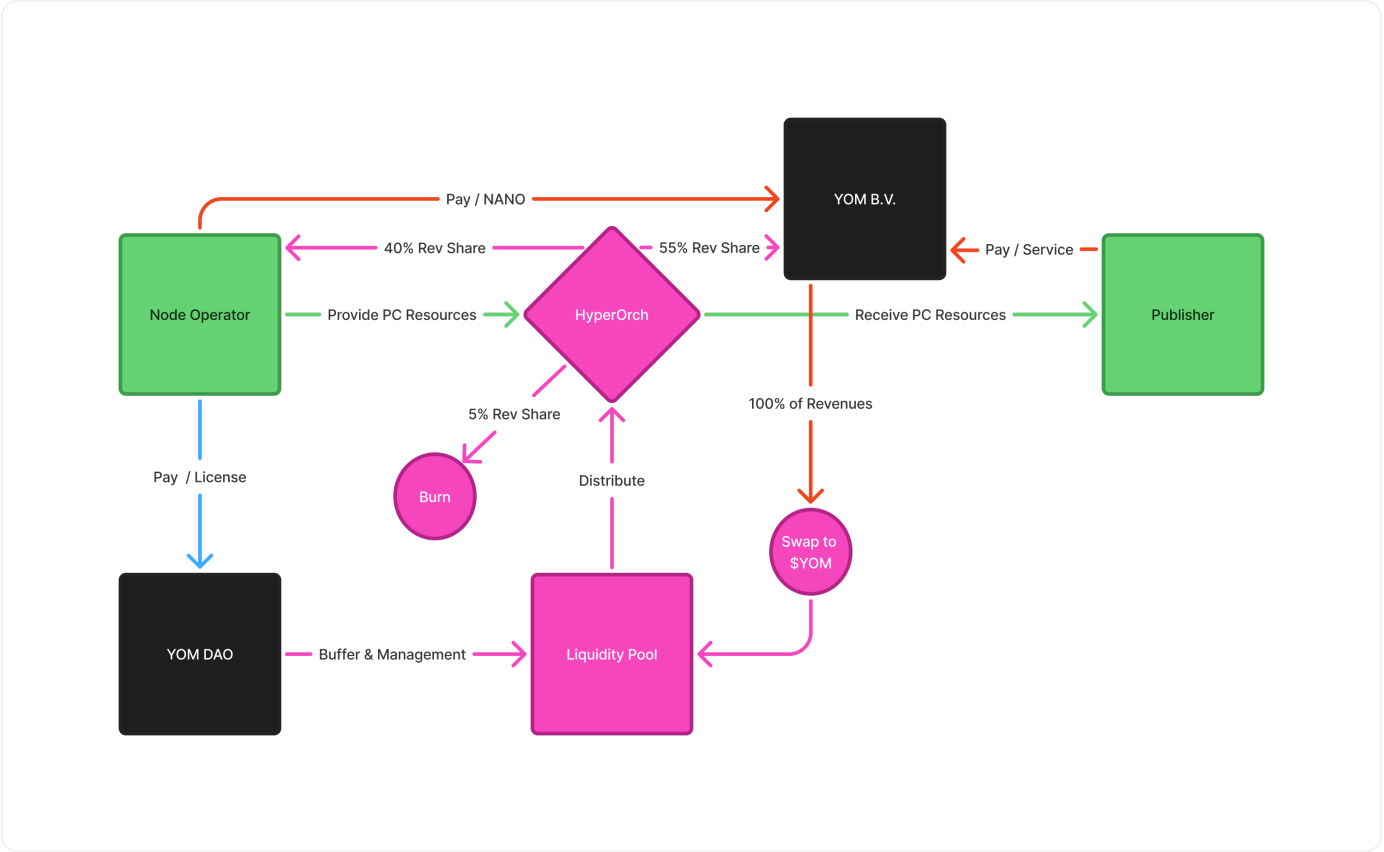
Un'economia che si autoregola

Ciò che tiene YOM in equilibrio è una struttura circolare e autoregolante: i giocatori creano la domanda, i nodi la servono, i publisher la finanziano e ogni partecipante condivide il risultato.

Il valore ($YOM) scorre secondo prezzi fissi per livello di sessione e ripartizioni a livello di protocollo: ricompense degli operatori, sviluppo della rete, la quota della community e il ritiro. Ogni tariffa di sessione viene regolata automaticamente.

La governance passa alla community per fasi. I voti dei detentori di token saranno ponderati in base alle disponibilità di $YOM e agli XP, un punteggio di contributo non trasferibile che si guadagna contribuendo alla rete. Le licenze vengono rilasciate in lotti man mano che la domanda riempie la rete, così la nuova offerta segue l'utilizzo reale. Il risultato:

- Gli operatori sono pagati per sessione servita, in automatico, a prezzi fissi per livello.
- I nodi che non superano la verifica vengono messi in quarantena automaticamente.



Per gli operatori di nodi

Gli operatori mettono a disposizione il proprio hardware usando NANO, un dispositivo di avvio plug-and-play che trasforma un normale PC da gaming in un nodo della rete in pochi minuti: avvia un sistema di sola lettura dal proprio drive cifrato, separato dal sistema operativo del proprietario.

Gli operatori ottengono:

- Una quota del 60% di ogni tariffa di sessione che servono, fissata in dollari e pagata in $YOM: circa ~US$53 al mese per NANO (una sessione Standard alla volta) all'obiettivo di utilizzo della rete del 50%, prima dell'elettricità. I guadagni totali seguono la capacità host calcolata dallo stimatore di backend (live su /operators; vedi Massimizzare i pagamenti per il dettaglio), al netto della loro elettricità.
- L'ingresso in un'economia di proprietà della community, in cui controllano il proprio hardware e le proprie ore.
- XP per le macchine affidabili e per il lavoro nella community, un punteggio di contributo non trasferibile che peserà sui voti dei detentori di token (vedi XP).

Gli operatori sono liberi di entrare o uscire in qualsiasi momento, il che allinea naturalmente l'offerta alla domanda.

Per publisher e studi

YOM offre ai publisher un nuovo canale per distribuire e monetizzare contenuti interattivi. Eliminare le barriere di infrastruttura e di hardware amplia il pubblico e riduce il costo di distribuzione.

I publisher possono:

- Portare la propria build, realizzata in qualsiasi engine, e inserire una sessione giocabile nei propri siti web, app e canali: un link incorporato, una pagina con il proprio brand o il proprio streamer eseguito come sessione gestita (panoramica della build).
- Ridurre il costo di infrastruttura da circa US$1.23 per un'ora su una GPU da data center riservata a livelli di calcolo per sessione da US$0.04 a US$0.24, con una sessione che dura fino a un'ora.
- Mantenere il pieno controllo della monetizzazione: premium, abbonamento, finanziata dalla pubblicità o free-to-play.

Con YOM i publisher raggiungono nuovi pubblici senza sostenere il costo del data center.

blockchain

Quando un giocatore preme **Play**, un nodo si sveglia, una GPU si accende, i pacchetti scorrono e il valore si muove, regolato end-to-end on-chain. Al centro c'è il **token $YOM**: l'unità di conto del calcolo, la ricompensa degli operatori e la supply che si riduce a ogni sessione.

YOM oggi regola su **Avalanche C-Chain**, costruito come token omnichain LayerZero (OFT); il deployment su **Base** è il prossimo. In tutta la rete si usa un unico token $YOM globale. È una scelta deliberata: al volume attuale Avalanche C-Chain è più che sufficiente, e una migrazione prematura non farebbe che frammentare la liquidità. Un L1 sovrano con token regionali è nella roadmap (vedi sotto), non è lo stato attuale.

Coordinamento del calcolo, non una chain generalista

$YOM non è una scommessa sugli smart contract generalisti. È il livello di settlement e di coordinamento per una rete globale di GPU. Ogni frame renderizzato e ogni sessione trasmessa è una micro-economia di calcolo, banda e ricompensa.

Sotto il cofano, **HyperOrch**, l'orchestratore edge di YOM, invia ogni sessione al nodo più vicino che ha la capacità GPU richiesta e spazio libero. Il nodo lancia il gioco nel proprio container, la sessione viene trasmessa al giocatore e la tariffa si regola in modo nativo on-chain.

Le ricompense sono legate alle **sessioni servite, non ai blocchi validati**, e prezzate in USD tramite livelli di calcolo fissi (vedi Pagamenti). Questo permette alla rete di pagare ricompense stabili e prevedibili senza dipendere dal prezzo spot del token. Gli operatori stipulano un contratto con il protocollo; il protocollo distribuisce 60% all'operatore, 30% allo sviluppo della rete, 5% a tutti coloro che conservano il proprio pagamento, e ritira 5%.

Gli operatori guadagnano servendo sessioni. Separatamente, **5% di ogni tariffa di sessione è condiviso con tutti coloro che conservano il proprio pagamento**. Durante il periodo iniziale di trading, il pool dell'ecosistema finanzia le ricompense di liquidità sugli exchange decentralizzati.

Deflazionistico per costruzione

Il token ha una struttura rigorosamente deflazionistica: **non si potranno mai coniare nuovi token** (supply fissa). A ogni sessione, **5%** della tariffa viene ritirato, rimuovendo supply in modo permanente. Man mano che l'utilizzo cresce, il flottante diventa più scarso, allineando i detentori del token all'attività continua della rete invece che alla speculazione.

Gestione e governance

Il protocollo e il token sono gestiti dalla **YOM Foundation**, che gestisce il settlement della rete e governa il programma che fa crescere la supply. La Foundation amministra la tesoreria e incarica market maker professionali per la liquidità del token.

Il controllo passa alla community per fasi. Prima, i detentori di token votano sul programma della supply: tassi di incentivo per gli operatori, allocazione della tesoreria, standard hardware e sovvenzioni. Poi i membri del consiglio eletti dalla community iniziano a sostituire gli attuali rappresentanti dei fondatori, a partire dal secondo anno dopo il lancio. I voti saranno ponderati **per il 67% in base agli XP**, un punteggio di contributo non trasferibile, e **per il 33% in base alle disponibilità di $YOM**, in modo che a guidare la rete siano le persone che fanno il lavoro, non solo i maggiori detentori.

Audit e amministrazione dei contratti

Hashlock, una società indipendente di sicurezza per smart contract, ha sottoposto ad audit il codice del contratto del token ad aprile 2026 e il codice dei contratti di pagamento e staking a ottobre 2026. Ogni report è pubblico, indica la versione del codice esaminata ed elenca ogni risultato con il suo stato: i report di Hashlock su YOM.

L'accesso amministrativo al contratto del token sta dietro un **multisig 2 su 3** e un **timelock di 48 ore**, così una modifica è visibile su Avalanche prima di avere effetto. Chiunque può verificare entrambi sull'explorer pubblico: il timelock e il multisig. Il contratto del token è `0xb6314518b61b4864162c7aE7fdc36261e0A14C4b`.

Roadmap: L1 sovrano e token regionali

Con la crescita del volume di transazioni, YOM prevede di migrare a un **L1 sovrano** e di introdurre **token regionali** (ad es. $yEU, $yCHN), ciascuno in un rapporto di burn/mint rispetto a $YOM, che permette ai mercati locali di prezzare il calcolo in modo indipendente mentre $YOM cattura valore a livello globale. La migrazione è condizionata al fatto che il volume di transazioni raggiunga il punto in cui il gas sulla chain condivisa supera la soglia che giustifica un L1 dedicato, con obiettivo **2027+**. Fino ad allora, tutto il settlement resta su Avalanche C-Chain con un unico $YOM globale.

Per il modello completo del token, l'allocazione e i dettagli del TGE, vedi Allocazione dei token.

rete

YOM è una rete GPU edge decentralizzata, costruita per orchestrare le GPU consumer inattive del mondo in un'unica griglia di calcolo che serve ogni sessione da un nodo vicino all'utente, entro circa 50 km dove la rete ne ha uno, a una frazione del costo del cloud centralizzato. Il gaming è il primo carico di lavoro perché è il più difficile: budget di latenza al millisecondo, fedeltà AAA e sessioni stateful che non si possono mettere in pausa né ripetere. La stessa rete fa girare conversazioni AI live come sessioni, e lo streamer o il modello di un partner passa per lo stesso percorso di sessione una volta revisionato e fissato.

Lo stack comprende una soluzione di pixel streaming su misura, un edge computing bare-metal sicuro e HyperOrch, l'orchestratore edge di YOM. Insieme, questi componenti trasmettono giochi in streaming su qualsiasi dispositivo con un browser, con bassa latenza edge e sicurezza dall'avvio allo stream, a una frazione del costo del cloud gaming centralizzato.

Il protocollo si basa su tre componenti: **HyperOrch** (l'orchestratore edge), **NodeOS** (il sistema operativo di sola lettura del nodo) e lo **Universal Streamer** (pixel streaming nel browser). I nodi girano su **bare metal, senza Kubernetes e senza macchine virtuali**, così ogni sessione raggiunge la GPU senza un hypervisor di mezzo.

Stateful per progettazione

La maggior parte delle reti GPU distribuite vende ore di GPU per lavori batch e stateless: instrada il calcolo verso qualsiasi GPU sia libera. Le sessioni in tempo reale sono fondamentalmente diverse. Richiedono un'esecuzione stateful: lo stato del gioco persiste da un frame all'altro e i cicli di input si misurano in millisecondi.

YOM è costruita per la sessione. Lo stesso modello di sessione porta giochi e conversazioni AI live, compresi lo streamer o il modello di un partner, ciascuno tenuto per un solo utente accanto a quell'utente. Così cattura il lavoro in tempo reale che le reti solo stateless, per struttura, non possono servire.

Pixel streaming peer-to-peer

YOM distribuisce un proprio servizio leggero di pixel streaming che fa girare i giochi su GPU di classe consumer remote, codifica ogni frame con l'encoder hardware della GPU (NVENC, H.264) e lo restituisce ai giocatori via WebRTC, così il giocatore non installa nulla. Una volta approvata una sessione, il gameplay viene consegnato su una connessione peer-to-peer diretta tra il nodo e il dispositivo del giocatore, ove possibile, e tramite un relay TURN quando un firewall o un NAT blocca il percorso diretto, così la sessione si connette comunque.

Video, audio e input viaggiano cifrati in transito con il DTLS-SRTP standard di WebRTC, riducendo al minimo i passaggi e mantenendo privato lo stream sulle reti pubbliche.

Universal Streamer e middleware

Ogni nodo YOM esegue un **livello middleware** che orchestra l'intero ciclo di vita della sessione: dalla preparazione del gioco e l'avvio di un container accelerato da GPU (Docker con NVIDIA Container Toolkit), fino alla gestione di tunnel, controlli di stato e chiusura della sessione. Il middleware preleva le proprie istruzioni da HyperOrch tramite una connessione in uscita, così il nodo non ha bisogno di nessuna porta in ingresso aperta, e fa in modo che ogni container sia monitorato e pronto allo streaming nel giro di pochi secondi.

All'interno di ogni container di gioco subentra lo **Universal Streamer**. Questo client di streaming leggero gestisce tutto ciò che serve al giocatore per interagire con il gioco: video e audio WebRTC, acquisizione degli input a bassa latenza (mouse, tastiera, gamepad, microfono) e streaming a bitrate adattivo, il tutto direttamente nel browser, senza nulla da

installare.

È lo Universal Streamer a rendere YOM indipendente dal gioco. I giocatori portano le librerie Steam ed Epic che già possiedono; gli studi caricano una build. Nessuna integrazione SDK richiesta. Lo streamer avvolge l'output renderizzato del gioco, stabilisce una connessione peer con il browser del giocatore e avvia lo streaming. Per gli sviluppatori, integrare un nuovo titolo è semplice come caricare una build e incorporare un iframe.

Insieme, il middleware e lo Universal Streamer formano la spina dorsale operativa di ogni nodo, trasformando la potenza GPU grezza in sessioni nel browser su larga scala.

Verifica dei nodi e telemetria

Ogni nodo si registra nella rete e comunica la propria classe di GPU, la posizione e il carico. I nodi NANO controllano i propri file di avvio rispetto a hash noti a ogni avvio; HyperOrch considera solo i nodi che superano il controllo, e un nodo che fallisce viene messo in quarantena automaticamente.

Durante il funzionamento live il nodo comunica FPS, latenza, bitrate, perdita di pacchetti e temperature, e il rig check gratuito misura quante sessioni una macchina può reggere prima ancora di servirne una.

Ogni sessione gira in un container pulito che viene scartato quando finisce, così nulla di un utente passa al successivo. Il disco di un NANO è interamente cifrato e a ogni avvio si sblocca solo tramite il servizio di chiavi di YOM; un dispositivo che YOM revoca resta bloccato.

I nodi NANO avviano un'immagine Linux minimale di sola lettura dal proprio drive, separata dal sistema operativo del proprietario. Al primo avvio il sistema operativo:

- Rileva automaticamente l'hardware e installa driver GPU firmati, così la scheda discreta è esposta alle massime prestazioni (nessun intervento dell'utente richiesto).
- Esegue un modulo di registrazione che collega la macchina al suo operatore e comunica a HyperOrch la capacità GPU, la posizione e la capacità della macchina.
- Esegue ogni sessione nel proprio container, distrutto al termine della sessione, condividendo la GPU con le altre sessioni tramite il time-slicing del driver.

HyperOrch: l'orchestratore edge

L'allocazione dei carichi di lavoro nella rete YOM è gestita da HyperOrch, l'orchestratore edge di YOM, oggi in produzione.

HyperOrch invia ogni sessione al nodo più vicino che ha la capacità GPU richiesta e spazio libero. HyperOrch filtra i nodi che sono online, hanno il gioco della sessione e dispongono di una GPU idonea e di capacità libera, poi invia l'istruzione di avvio al più vicino. I nodi senza una GPU idonea o senza capacità libera vengono scartati, quindi il nodo più vicino può perdere contro uno che è idoneo.

Sul nodo, oggi più sessioni condividono una stessa scheda consumer tramite il time-slicing del driver GPU.

Se il nodo scelto non risponde, la sessione passa al nodo idoneo successivo. Se un nodo cade a metà sessione, il giocatore riparte sul prossimo nodo vicino.

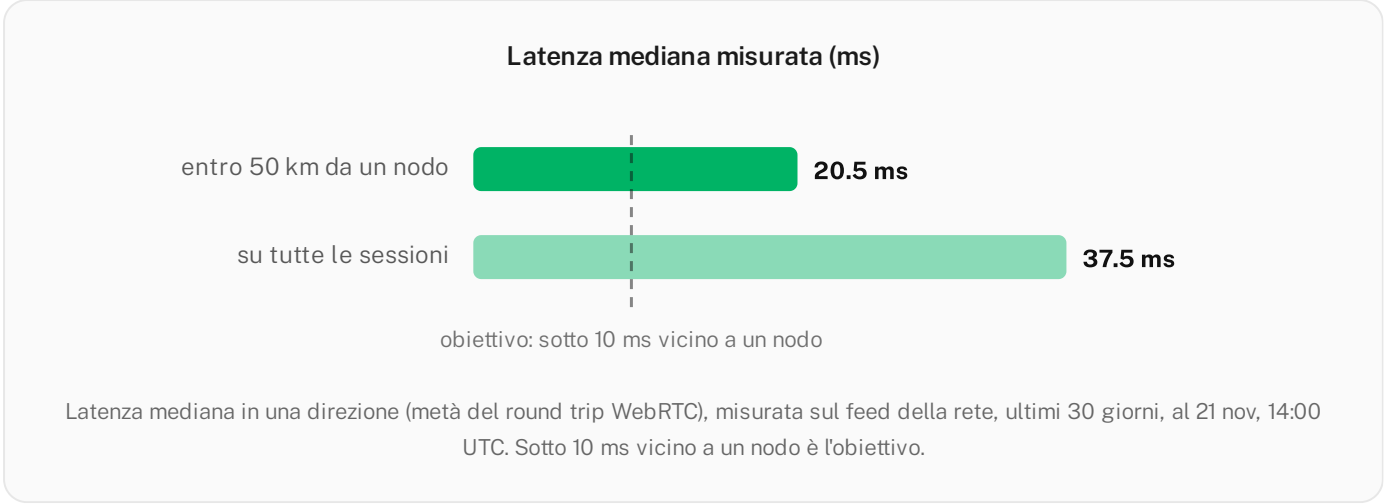
Capacità di overflow e livelli di servizio

L'offerta primaria della rete sono le GPU consumer gestite dalla community, ma in ogni nuova regione la densità richiede tempo per costruirsi. Dove l'offerta degli operatori è scarsa, le sessioni Walter girano su GPU noleggiate su richiesta, circa il 13% delle sessioni Walter nel primo anno del piano, in calo al 3% entro l'anno cinque man mano che cresce la copertura degli operatori.

I livelli di servizio fissano l'offerta dietro ogni sessione, e ciascuno è espresso come tasso di completamento delle sessioni: la quota di sessioni di un'ora che terminano senza una disconnessione causata dalla rete, non l'uptime. Sulle GPU domestiche (Basic, best effort) l'obiettivo di progetto è il 98% di sessioni senza una caduta causata dalla rete. Protected punta al 99.5% di sessioni di un'ora senza una caduta causata dalla rete, con la maggior parte delle sue sessioni su rack telco, e Carrier punta al 99.9% solo su rack telco.

Brevetti

Tecnologia in attesa di brevetto per il supporto di avvio sicuro, il pixel streaming e l'orchestrazione.



Latenza mediana in una direzione (metà del round trip WebRTC), misurata sul feed della rete, ultimi 30 giorni, al 21 nov, 14:00 UTC. Sotto 10 ms vicino a un nodo è l'obiettivo.

operatori

Nell'ecosistema YOM i **nodi** sono la spina dorsale di una rete GPU edge decentralizzata. Ogni nodo è una GPU di proprietà di un operatore che trasmette sessioni di gioco live. A differenza dei "nodi" crypto tradizionali, che spesso restano inattivi o dipendono da emissioni speculative di token, i nostri nodi svolgono lavoro reale e vengono pagati per sessione servita.

Che cos'è un nodo?

Un nodo è la macchina di un operatore al servizio della rete: un computer da gaming o un server la cui GPU esegue sessioni per i giocatori vicini. Quando un giocatore lancia un gioco sulla rete, il gioco viene renderizzato sull'hardware di un operatore di nodo e trasmesso in tempo reale al dispositivo del giocatore, ogni sessione nel proprio container, distrutto al termine della sessione.

In termini più semplici, far girare un nodo YOM è come ospitare un piccolo server di gioco a cui i giocatori si collegano per una sessione, con la differenza che il giocatore vede soltanto il flusso video. Poiché YOM è decentralizzata, questi nodi possono essere gestiti da operatori comuni sul proprio hardware. Ogni volta che il tuo nodo serve una sessione, vieni pagato per quella sessione.

Chiunque abbia l'hardware adatto e una licenza di nodo può far girare un nodo. La capacità di una macchina, misurata dal suo rig check, si conta in unità: una sessione Light ne usa una, Standard due, Premium quattro e Ultra sei. Una licenza di nodo sblocca un'unità di capacità, ogni NANO Node Runner include due licenze (sufficienti per una sessione Standard), e le licenze aggiuntive mettono in servizio più capacità di una macchina, fino a quanto mostra il suo rig check.

Le licenze sono il modo in cui la rete controlla l'offerta. Nuovi lotti di Node Runner e licenze di nodo si aprono quando l'utilizzo, previsto a tre mesi, raggiunge il 60%, così la nuova capacità segue la domanda e mantiene occupati gli host già presenti nella rete.

Quante sessioni per host?

Una macchina fisica può servire **più sessioni contemporaneamente**, in qualsiasi combinazione compatibile con la sua capacità, quando le sue risorse e la sua banda le supportano. Parti da un NANO Node Runner e usa il controllo PC gratuito per valutare la capacità prima di aggiungere licenze di nodo.

Ogni sessione gira nel proprio container. Un operatore con capacità sufficiente può servire più giocatori contemporaneamente da una sola macchina.

Guadagni degli operatori

Gli operatori guadagnano una quota fissa di ogni tariffa di sessione, fissata in dollari e pagata in $YOM. All'obiettivo di utilizzo della rete del 50%, un NANO (una sessione Standard alla volta) guadagna circa **~US$53 al mese** prima dell'elettricità. I guadagni totali dell'host dipendono dalla capacità che lo stimatore di backend assegna all'hardware misurato. Un NANO Node Runner costa **US$349** con due licenze incluse; una nuova licenza di nodo costa **US$125**. Il rientro dell'investimento è di circa **~7 months** all'obiettivo di utilizzo della rete del 50%.

All'obiettivo di utilizzo della rete del 50%	Guadagni mensili	Hardware	Rientro
Un NANO, una sessione Standard	~US$53	US$349	~7 months
Capacità host misurata	Risultato dello stimatore di backend	US$349 + US$125 per ogni licenza aggiuntiva	vedi /operators

I guadagni crescono con le sessioni che la tua macchina serve. Le licenze di nodo restano sul tuo account, quindi puoi spostarle su un'altra macchina.

XP

Gli XP sono un punteggio di contributo non trasferibile che peserà sui voti dei detentori di token. Riconoscono le macchine affidabili e il lavoro nella community; non cambiano l'assegnazione delle sessioni né i guadagni. Gli XP si guadagnano attraverso:

1. **Operatività della rete.** I proprietari di nodi che fanno girare macchine stabili e con alto uptime accumulano XP in proporzione all'affidabilità dei loro nodi.
2. **Partecipazione alla community.** La partecipazione attiva su Discord o sul forum, guide o tutorial di alta qualità e il sostegno sui social generano XP. La web app e Discord danno ai membri un facile accesso alle attività in cui possono guadagnare XP.
3. **Lavori per la community.** Eventi periodici organizzati dalla YOM Foundation o dai comitati della community possono offrire XP bonus per attività come lo stress test di nuove funzionalità o l'aiuto nell'accoglienza dei nuovi membri.

pagamenti

Un nodo su YOM guadagna una tariffa per ogni sessione di gioco che serve, pagata a un prezzo fissato dal livello di carico della sessione. Niente farming di token inattivi e niente misurazione al minuto: la sessione è l'unità atomica che i giocatori consumano, gli operatori servono e il protocollo regola. Una sola grandezza attraversa l'intero sistema, ed è ciò che rende ogni pagamento verificabile da un capo all'altro.

Livelli di calcolo: un'esperienza erogata, non una scheda tecnica hardware

Ogni sessione si regola a una tariffa fissa per sessione, stabilita dal livello di carico. Un livello è un **contratto di esperienza erogata**: la risoluzione, il framerate e la reattività che il giocatore riceve davvero, non il nome di un modello di GPU. La rete misura se una macchina può rispettare il contratto di un livello prima ancora di assegnarglielo.

Livello	Crediti Walter per sessione	Tariffa di rete per sessione
Light	1	US$0.04
Standard	2	US$0.08
Premium	4	US$0.16
Ultra	6	US$0.24

Vedi i livelli di calcolo attuali per la risoluzione, il frame rate e l'allocazione di risorse di ciascun livello. I crediti Walter pagano l'accesso dei consumatori; la tariffa di rete paga la sessione di calcolo. Il prezzo dell'abbonamento e la tariffa di rete sono importi separati.

Le tariffe crescono con le risorse che una sessione riserva. A parità di durata della sessione e di utilizzo, una capacità riservata equivalente guadagna la stessa tariffa per sessione in tutti i livelli. A ogni gioco viene assegnato un livello, e l'operatore non lo sceglie; attualmente ogni gioco gira al livello Standard. I guadagni effettivi dell'host dipendono anche dalla domanda, dalla durata delle sessioni e dalla combinazione che il suo hardware può servire.

Ogni sessione, un gioco o una conversazione AI live, è fatturata per sessione, e una sessione dura fino a un'ora: una sessione Standard costa US$0.08, contro circa US$1.23 per la stessa ora su una GPU da data center riservata per il picco.

Crediti di sessione e tempo di gioco

Il Compute Pass di Walter include un'allocazione mensile di crediti. Una sessione usa i crediti indicati sopra per il suo livello e può durare fino a **60 minutes**. L'abbonamento da US$9.99 include 70 crediti, che i giocatori possono spendere tra livelli diversi. I crediti extra costano US$0.175 ciascuno.

Il tetto di sessione descrive quanto può durare una singola sessione. La previsione usa una durata ipotizzata di 20 minutes per stimare l'utilizzo tipico e i ricavi per ora; non fissa la durata della sessione di un giocatore. Le sessioni più lunghe entro il tetto distribuiscono lo stesso addebito di sessione su più tempo di gioco.

Flusso del valore

Ogni sessione regolata attraverso il protocollo YOM paga la tariffa del suo livello, e la rete la divide quando la sessione si chiude. Gli operatori stipulano un contratto con il **protocollo**, non con i singoli publisher, il che rende i pagamenti stabili e prevedibili a prescindere da chi ha venduto la sessione.

Parte interessata	Ripartizione	Descrizione
Operatori di nodi	60%	Pagata direttamente agli operatori che hanno servito la sessione.
Sviluppo della rete	30%	Finanzia lo sviluppo e la crescita della rete.
Quota dei detentori	5%	Condivisa con tutti coloro che conservano il proprio pagamento.
Ritirata	5%	Ritirata in modo permanente dalla supply.

Questa ripartizione 60% / 30% / 5% / 5% è l'invariante economico del protocollo. Si applica in modo uniforme: le sessioni Walter, le sessioni enterprise e qualsiasi aggregatore terzo che regoli attraverso la rete pagano tutti la stessa ripartizione, sugli stessi livelli.

Il pagamento in due parti

Il reddito dell'operatore ha due componenti, e la distinzione conta quando stai decidendo se collegare la macchina:

1. **Pagamento a consumo**: la quota del 60% di ogni tariffa di sessione che la tua macchina serve. Cresce con la domanda della rete e con la capacità misurata della tua macchina.
2. **Bonus mensile**: US$74 per NANO ogni mese, in aggiunta ai guadagni di sessione. US$37 sono garantiti quando la macchina è online almeno 6 ore al giorno, e US$37 vanno in un pool condiviso, diviso in base alle sessioni che ogni macchina serve. Resta finché un sistema di sussidio migliore non lo sostituisce, con annuncio anticipato.

All'obiettivo di utilizzo della rete del 50% un NANO (una sessione Standard alla volta) guadagna circa ~US$53 al mese prima dell'elettricità. I guadagni totali dell'host crescono con la capacità che lo stimatore di backend assegna all'hardware misurato; le cifre live sono su /operators e crescono con la domanda della rete.

Riprezzamento prevedibile, non deriva silenziosa

Le tariffe di sessione sono calibrate su una durata di sessione ipotizzata e poi confrontate con l'utilizzo osservato. Man mano che la libreria di giochi migliora e i giocatori restano più a lungo, la tariffa per sessione può essere ricalibrata in modo che i guadagni dell'operatore per ora-macchina restino stabili. **Le modifiche alle tariffe vengono annunciate prima che entrino in vigore**: gli operatori devono poter pianificare in base al proprio tasso di pagamento come pianificano in base alla bolletta della luce.

Prezzi

Le tariffe di sessione sono prezzate in USD e pagate in **$YOM**, con un unico token globale. Questo mantiene le ricompense degli operatori stabili e prevedibili senza dipendere dal prezzo spot del token. Vedi Tokenomics per i dettagli sul token.

Perché, per struttura, batte il cloud centralizzato

L'endgame non è uno sconto; è una curva di costo diversa. Il prezzo riservato di US$1.23 all'ora di una GPU da data center nasce dal capex delle GPU su un orologio di ammortamento quinquennale, dagli immobili del data center, dal raffreddamento e dal networking enterprise: costi sostenuti che qualcuno stia giocando o no. L'offerta di YOM è **silicio consumer marginale che esiste già**: il costo orario della rete è l'elettricità dell'operatore più l'incentivo che rende conveniente collegare la macchina, su hardware che comunque era stato comprato per giocare.

Servire sessioni su hardware consumer esistente riduce il costo di capitale che ogni sessione deve recuperare. I guadagni degli operatori dipendono comunque dalla domanda, dall'utilizzo e dalla durata di sessione effettivamente servita.

In breve: **il tuo nodo guadagna servendo sessioni, prezzate per livello di carico**, con una ripartizione uniforme e prevedibile che non dipende da chi ha venduto la sessione né da dove si trova il giocatore.

SEZIONE

guadagna

nano

Il nostro **NANO** plug-in è un SSD avviabile e cifrato che trasforma un PC da gaming in un nodo della rete, con pieno controllo e una configurazione semplice: metti in funzione il tuo nodo senza installare software complesso. Il NANO costa **US$349** e include **2 licenze di nodo**, sufficienti per una sessione Standard. È prodotto e distribuito dal nostro partner hardware **SIMMS** e venduto tramite rivenditori autorizzati.

Se preferisci non usare un dispositivo dedicato, su rig idonei è supportata anche una **configurazione di nodo solo software**.

Passo 1: rig check

Il rig check gratuito stabilisce se il tuo rig è idoneo e ne misura la capacità: quanta della sua potenza la rete può mettere al lavoro contemporaneamente, in unità. Come indicazione, cerca:

Requisiti di sistema:

- **GPU:** Una scheda NVIDIA RTX desktop. Più VRAM significa più capacità.
- **CPU:** Intel i5/i7 o AMD Ryzen 7/9 per il multitasking e la stabilità.
- **RAM:** 16GB bastano nelle configurazioni più piccole; si consigliano 32GB o più.
- **Internet:** Una connessione Ethernet cablata con almeno 20 Mbps di upload misurato.
- **Raffreddamento:** Un flusso d'aria e un raffreddamento adeguati per evitare il surriscaldamento.

Passo 2: ordina il NANO

Quando il rig check conferma che il tuo sistema è idoneo, puoi ordinare il **NANO** a **US$349** (prezzo indicativo; possono applicarsi imposte regionali). Il dispositivo è costruito e distribuito dal nostro partner hardware **SIMMS** e venduto tramite webshop di rivenditori autorizzati come Freshminers.com.

Passo 3: configura il tuo nodo

Quando il dispositivo arriva, segui questi passaggi per metterlo in funzione:

1. **Collega il NANO al tuo PC.**
2. **Avvia dal NANO.**
   - Riavvia il PC e apri il BIOS premendo `F2`, `F12` o `DEL` durante l'avvio.
   - Imposta l'SSD YOM come dispositivo di avvio principale.
   - Salva ed esci dalle impostazioni del BIOS per continuare.
3. **Segui la configurazione a schermo:**
   - Quando il sistema si avvia in YOM Node OS, segui la configurazione guidata.
   - Scansiona il codice sulla scatola con l'app per operatori per collegare la macchina al tuo account. L'app crea il tuo wallet per te.

Ecco fatto: il tuo nodo è pronto a servire.

licenze dei nodi

Dopo aver configurato il tuo NANO, le licenze di nodo decidono quanta della capacità della tua macchina è in servizio. La capacità è quanta della potenza di una macchina la rete può mettere al lavoro contemporaneamente, misurata in unità: una sessione Light usa 1 unità, Standard 2, Premium 4 e Ultra 6. La tua scheda grafica, il processore, la memoria e la velocità di upload determinano la capacità della tua macchina, e il rig check gratuito la misura. Una macchina serve qualsiasi combinazione di sessioni compatibile. Un host ha bisogno di una connessione cablata con almeno 20 Mbps di upload misurato.

Una licenza sblocca un'unità di capacità. La tua macchina serve fino alla propria capacità o alla capacità che le sue licenze sbloccano, a seconda di quale sia la più bassa. Le licenze limitano quanta capacità è in servizio, mai quanti giocatori: 5 licenze servono una sessione Premium e una Light contemporaneamente, oppure due Standard e una Light, oppure cinque Light. Non c'è un tetto alle licenze per macchina, e una macchina più capace ha più capacità. Ogni NANO include **2 licenze**, sufficienti per una sessione Standard.

Quando questo entra in vigore, e che cosa significa per le licenze che già possiedi: dagli slot alla capacità: ogni licenza diventa due.

Una licenza è il diritto di far girare capacità sulla rete, e resta sul tuo account.

Dalla vista di gestione dei nodi nella web app, assegni le tue licenze al tuo NANO.

Come ottenere una licenza di nodo?

Una nuova licenza costa **US$125**. Le licenze vengono rilasciate in lotti nell'app per operatori: un nuovo lotto di licenze e NANO si apre quando gli host della rete sono occupati al 60%, così la nuova capacità si aggiunge man mano che la domanda cresce e gli host che già servono restano occupati.

delega naas

Node-as-a-Service (NaaS) è un modo per mettere al lavoro le licenze di nodo senza far girare un proprio PC. Un provider fa girare le licenze per il proprietario e addebita una commissione per la capacità che gestisce.

Perché delegare

La delega si adatta ai titolari di licenze che non possiedono un PC da gaming o preferiscono non usare il proprio per la rete. Il provider si occupa di configurazione, monitoraggio e manutenzione. Il proprietario riceve la quota dell'operatore di ogni sessione servita dalle licenze, fissata in dollari e pagata in $YOM, meno la commissione del provider.

Ciò che rende una licenza dipende da quanto bene lavora il provider: i provider che collocano la capacità dove sono i giocatori e la tengono online servono più sessioni.

Domande frequenti

1. **Perché i provider contano per la rete?** I provider aggiungono capacità dove il picco di domanda supera le GPU domestiche vicine, così i giocatori di quelle regioni hanno comunque un nodo vicino.
2. **Posso far girare io le mie licenze?** Sì. Una licenza resta sul tuo account e puoi sempre farla girare sul tuo PC con un NANO.
3. **Ci sono commissioni?** Sì. I provider addebitano una commissione per la capacità che gestiscono per te. Controlla prima le condizioni del provider.

massimizzare i pagamenti

Ogni sessione si regola attraverso il protocollo con una ripartizione uniforme (60% all'operatore, 30% allo sviluppo della rete, 5% condiviso con tutti coloro che conservano il proprio pagamento e 5% ritirato), quindi il tuo pagamento non dipende mai da chi ha venduto la sessione né da dove si trova il giocatore. La tua quota è una parte fissa di ogni tariffa di sessione, fissata in dollari e pagata in $YOM. Ciò che controlli tu è quanta parte di quel flusso cattura la tua macchina. I tuoi guadagni sono il prodotto di due cose: **quanta capacità ha onestamente il tuo hardware e quanto spesso viene riempita**. Questa guida tratta entrambe. A ogni gioco viene assegnata una dimensione di sessione, e non la scegli tu; attualmente ogni gioco gira alla dimensione Standard.

Costruisci la tua capacità

Un host può far girare più sessioni contemporaneamente quando le sue risorse misurate le supportano. La sua capacità si misura in unità: una sessione Light ne usa 1, Standard 2, Premium 4 e Ultra 6, e una macchina serve qualsiasi combinazione compatibile. Il rig check calcola la capacità supportata a partire da CPU, memoria, GPU, encoder e capacità di upload. Ogni licenza di nodo sblocca un'unità, e il tuo NANO ne include due, sufficienti per una sessione Standard.

Aggiungi capacità quando la tua macchina può reggerla. La capacità è limitata dalla prima risorsa che si esaurisce:

- **Margine della GPU e VRAM:** ogni livello ha un budget fisso per stream.
- **Sessioni dell'encoder e throughput:** la GPU deve codificare ogni sessione entro i propri limiti.
- **Banda di upload:** la stima include la quota di upload del livello e il margine di rete.
- **CPU, memoria e archiviazione:** devono starci il gioco completo, Steam e lo stack di streaming.

Ordine pratico delle operazioni: controlla prima la velocità di *upload* sostenuta, poi aggiungi capacità un passo alla volta mentre la tua dashboard conferma che ogni sessione mantiene la propria qualità. Le licenze oltre la capacità reale della tua macchina non guadagnano: la rete assegna le sessioni in base alla capacità misurata, non al numero di licenze.

Riempi la tua capacità

HyperOrch, l'orchestratore edge di YOM, invia ogni sessione al nodo più vicino che ha la capacità GPU richiesta e spazio libero. Quindi il modo per servire più sessioni è essere online, con capacità libera, quando i giocatori vicini a te giocano.

Conta il momento: la domanda di gaming raggiunge il picco la sera e nei fine settimana. Un nodo che è online in modo affidabile in quelle ore cattura molte più sessioni di uno online a orari casuali.

Evita le disconnessioni brusche. Usa hardware affidabile e una connessione cablata stabile. Se un nodo non risponde, la sessione passa al nodo idoneo successivo, e i nodi che continuano a fallire vengono messi in quarantena automaticamente, quindi il vantaggio duraturo è l'affidabilità.

Farlo girare

Guadagni facendo girare licenze sul tuo PC con un NANO.

Caratteristica	NANO
Conoscenze tecniche	Minime
Tempo di configurazione	Moderato
Cosa ottimizzi	Essere online quando i giocatori vicini a te giocano
Hardware	PC da gaming
Costo di esercizio	La tua elettricità
Impegno	Nessuno: puoi fermarti quando vuoi. I guadagni seguono l'elettricità e la domanda locale

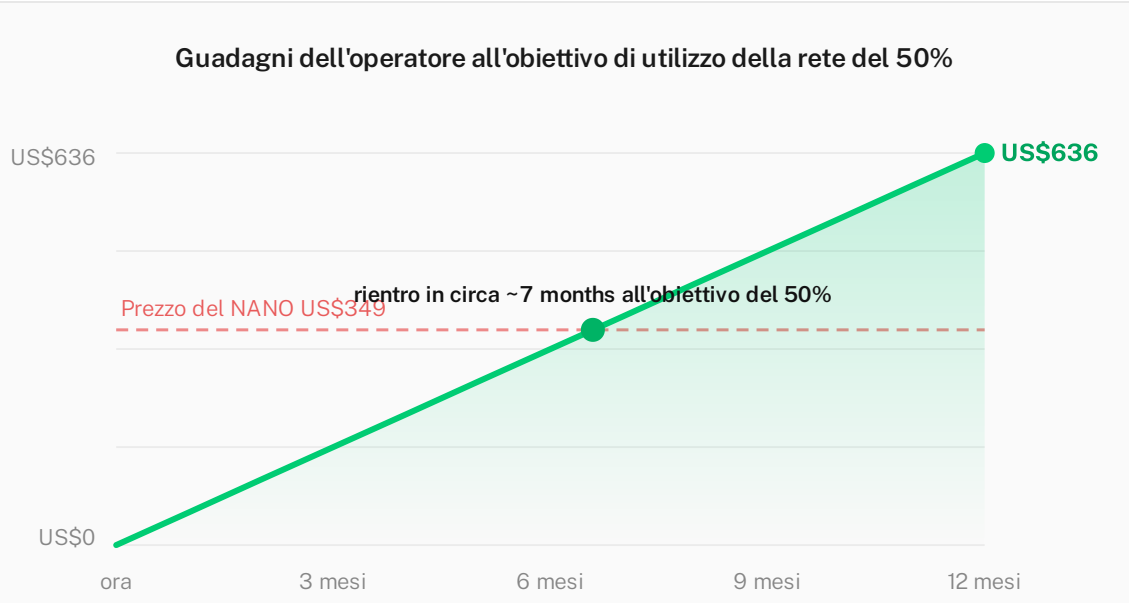
Aspettative di guadagno: esempi pratici

La domanda più frequente è «**quanto posso guadagnare?**» La risposta onesta è un intervallo, non una promessa, perché i guadagni crescono con la domanda live della rete nella tua zona.

Il modello della rete calcola i guadagni all'**obiettivo di utilizzo della rete del 50%**: una macchina occupata per circa metà delle ore in cui è disponibile, con sessioni di durata tipica. Su questa base un NANO (una sessione Standard alla volta) guadagna **~US$53 al mese** e ogni unità di capacità circa ~US$26, prima dell'elettricità. Il tuo totale segue la capacità che il rig check misura sul tuo host (stima su /operators). Sono stime, non garanzie: la rete sta ancora crescendo verso quel livello, quindi i guadagni di oggi sono inferiori, e la tua dashboard mostra le tue sessioni.

Due componenti formano quel reddito: il **pagamento a consumo** (la tua quota del 60% di ogni tariffa di sessione servita) e il **bonus mensile**, US$74 per NANO ogni mese in aggiunta ai guadagni di sessione. US$37 sono garantiti quando la macchina è online almeno 6 ore al giorno, e US$37 vanno in un pool condiviso, diviso in base alle sessioni che ogni macchina serve. Resta finché un sistema di sussidio migliore non lo sostituisce, con annuncio anticipato.

Rientro, su hardware che già possiedi: un NANO costa **US$349** e include due licenze, sufficienti per una sessione Standard. All'obiettivo di utilizzo della rete del 50% un NANO guadagna ~US$53 al mese, quindi rientra dell'investimento in ~7 months prima dell'elettricità. Nuovi lotti di licenze si aprono solo quando la domanda riempie gli host che già servono. L'elettricità pesa di più nelle regioni con tariffe alte: un mese all'obiettivo costa circa ~US$12 in Europa, meno nella maggior parte degli altri mercati, quindi controlla la tua tariffa locale. Se dovessi prima comprare una GPU, il rientro è più lungo: l'economia è pensata per la macchina che hai già sotto la scrivania.



Guadagni cumulativi in 12 mesi per un NANO che serve una sessione Standard alla volta all'obiettivo di utilizzo della rete del 50%, prima dell'elettricità. Rientro in circa ~7 months all'obiettivo di utilizzo del 50%. I guadagni effettivi seguono le sessioni che una macchina serve.

referral

Programma referral

Guadagna ricompense condividendo YOM con la tua rete. Quando qualcuno acquista un NANO YOM (**US$349**, due licenze di nodo incluse) usando il tuo codice referral, risparmia e tu guadagni una commissione.

Come funziona

1. **Ottieni il tuo codice referral:** Trovi il tuo codice univoco nella dashboard Referral
2. **Condividilo:** Invia il tuo link ad amici, pubblicalo sui social o condividilo nelle community
3. **Guadagna ricompense:** Quando qualcuno fa un acquisto usando il tuo codice, ne traete vantaggio entrambi

Ricompense

Vantaggio	Descrizione
Sconto per l'acquirente	Percentuale sul prezzo del prodotto
Commissione per chi segnala	Percentuale del prezzo del prodotto (IVA esclusa)

Il tuo tasso referral può variare. Controlla la tua dashboard Referral per le percentuali attuali di sconto e di commissione.

Come si calcola la commissione

La commissione si calcola sul prezzo del prodotto IVA esclusa. Il prezzo di listino UE è €369 IVA del 21% inclusa (€305 IVA esclusa), e l'esempio svolto qui sotto lo usa.

Esempio: Un acquirente compra un NANO YOM usando un codice referral con un tasso del 10%.

Passaggio	Calcolo	Importo
Prezzo al pubblico (IVA del 21% inclusa)		€369.00
Prezzo del prodotto (IVA esclusa)	€369.00 ÷ 1.21	€305.00
Sconto per l'acquirente (10%)	€369.00 × 10%	-€36.90
Spedizione		€18.15
L'acquirente paga	€369.00 -€36.90 + €18.15	€350.25
Chi segnala guadagna (10%)	**€305.00 × 10%**	**€30.50**

La commissione si basa sul prezzo del prodotto IVA esclusa perché l'IVA viene riscossa per conto delle autorità fiscali e non fa parte dei ricavi del prodotto.

Quando le ricompense vengono confermate

La tua commissione viene confermata quando si verificano entrambe le condizioni:

- Il periodo di recesso di 14 giorni è trascorso
- Il NANO acquistato è stato attivato

Puoi seguire le ricompense in sospeso e quelle confermate nella tua dashboard Referral.

Pagamenti

Le commissioni maturate vengono pagate alla fine di ogni mese sul tuo wallet collegato.

Condividere il tuo codice

Puoi condividere direttamente il tuo link referral oppure usare i pulsanti di condivisione rapida per X (Twitter) e Telegram nella dashboard.

Classifica

I migliori referrer sono mostrati nella classifica. Scala le posizioni portando più operatori di nodi nella rete YOM.

SEZIONE

costruisci

panoramica

YOM trasmette in streaming qualsiasi gioco compilato per Linux su qualsiasi browser. Niente app store, niente download, nessun requisito sul dispositivo. I giocatori cliccano un link e giocano.

Di cosa si occupa YOM per te:

- Streaming video/audio WebRTC (H.264, VP8, VP9, Opus)
- Acquisizione degli input: mouse, tastiera, gamepad, microfono
- Adattamento automatico tra dispositivi e dimensioni dello schermo
- Gestione delle sessioni e allocazione dei nodi: HyperOrch, l'orchestratore edge di YOM, invia ogni sessione al nodo più vicino che ha la capacità GPU richiesta e spazio libero
- Analytics tramite l'integrazione con Google Tag Manager

Tre modi per costruire su YOM

Scegli il percorso adatto alle tue esigenze:

Percorso	Cosa ti serve	Tempo per andare live	SDK richiesto?
Avvio rapido	Compilare il gioco per Linux, caricarlo, incorporare un iframe	~15 minuti	No
Integrazione web	UI personalizzata, pagine multi-stream, comandi dal web al gioco	~1 ora	Script di embed JS
Il tuo streamer o modello	Il tuo streamer o modello vocale, revisionato e fissato, poi eseguito come sessione gestita	Integrazione partner, a partire da un pilota circoscritto	Parlane con noi

Avvio rapido: trasmetti qualsiasi gioco

Carica il tuo binario Linux compilato e incorpora un solo iframe. Lo streamer universale di YOM gestisce tutto il resto: WebRTC, input, video, audio. Nessuna modifica al codice del tuo gioco.

Ideale per: demo, vetrine, esperienze single-player, prototipazione rapida.

Integrazione web: portali personalizzati

Usa lo script di embed di YOM (`yom-embed.js`) per costruire portali di gioco personalizzati con più stream, layout dinamici, comunicazione dal web al gioco e flussi di onboarding con il tuo brand.

Ideale per: portali di gioco, aggregatori di contenuti, vetrine con il proprio brand, pagine di marketing interattive.

Il tuo streamer o modello: sessioni gestite

Una piattaforma che trasmette giochi ai propri utenti, o un'azienda che fa girare un proprio modello vocale, lo distribuisce su YOM come sessione gestita. YOM revisiona il tuo streamer o modello e fissa la versione revisionata; da quel momento gira solo attraverso il percorso di sessione. HyperOrch colloca ogni sessione su un nodo vicino all'utente, e NodeOS la esegue nel proprio container, creato per quella sessione e scartato quando si chiude. Il rapporto con l'utente e il brand restano tuoi.

Le piattaforme lo acquistano come **YOM Games** e i team vocali come **YOM AI**, entrambi per sessione agli stessi livelli di calcolo, da US$0.04 a US$0.24 a sessione.

Ideale per: piattaforme di streaming che fanno girare un proprio streamer di giochi e aziende con modelli di linguaggio e voce propri.

Come si vende ciascun percorso

Avvio rapido e Integrazione web si inseriscono nel modo in cui **YOM Games** viene venduto, per sessione: tramite Walter, il servizio consumer di YOM, o tramite il tuo canale. Le istruzioni qui sopra sono le stesse; cambia il modo in cui lo porti sul mercato e il funzionamento della fatturazione.

- **Discovery** alimenta le demo incorporate gratuite. Inserisci una demo giocabile su Reddit, Discord, Telegram, una landing page o uno spazio pubblicitario usando Avvio rapido o Integrazione web. Nessuna commissione di servizio; le sessioni girano al costo di calcolo di base. È il cuneo che trasforma un clic in un giocatore.
- L'**Hosting Slot** è il piano per publisher a **US$89 al mese**: 2,300 crediti (1,150 sessioni Standard) e un carico di lavoro, con l'utilizzo oltre quel limite a US$0.048 per credito. I piani a pagamento partono da un pilota circoscritto o dall'Hosting Slot. Qui la maggior parte degli studi usa Avvio rapido.
- Il **Compute Pass** è l'abbonamento consumer di Walter (fatturazione da gennaio 2027): US$4.99 / US$9.99 / US$19.99 al mese con 30 / 70 / 140 crediti. Ogni sessione usa 1 crediti a Light, 2 a Standard, 4 a Premium o 6 a Ultra, per un massimo di 60 minutes. L'allocazione mensile può coprire una combinazione di livelli di sessione; i crediti extra costano US$0.175 ciascuno. I giocatori collegano Steam o Epic e trasmettono in streaming i giochi che già possiedono (bring-your-own-game). I publisher indirizzano la propria base di giocatori al Compute Pass come livello di streaming, mantenendo il proprio store e i propri ricavi.

Vedi i crediti del Compute Pass per come l'allocazione mensile si rapporta ai livelli di sessione e al tempo di gioco.

Qualunque sia il modello che ti si adatta, il lavoro di integrazione di questa pagina è ciò che ti porta lì: YOM è infrastruttura, non una vetrina che trattiene una percentuale. Mantieni il controllo della monetizzazione, che sia un abbonamento, un titolo premium, una demo finanziata dalla pubblicità o una vetrina gratuita.

Progetta per il gioco immediato

I giochi in cloud streaming vivono su pagine web e sono accessibili subito da qualsiasi dispositivo. Per sfruttare al meglio il gioco immediato:

- **Salta i menu principali:** i giocatori si aspettano di buttarsi direttamente nel gameplay.
- **Progetta in modo responsive:** il tuo gioco girerà su telefoni, tablet e desktop.
- **Pensa in modo nativo per il web:** usa widget web e la comunicazione bidirezionale tra gioco e web al posto della UI di gioco dove ha senso.

avvio rapido

Metti un gioco in streaming su YOM in due passaggi: carica il tuo binario, poi incorpora un iframe.

> **Stato della rete:** La rete di produzione è attiva dal 5 giugno 2026. Il flusso di Avvio rapido qui sotto è attivo oggi. Se un passaggio ti appare diverso nella dashboard man mano che la rete scala, scrivici a info@yom.net.

Passo 1: distribuisci il tuo stream

1. Compila il tuo gioco per **Linux x86_64**. Va bene qualsiasi engine: Unreal Engine, Unity, Godot o personalizzato.
2. Accedi ad app.yom.net/streams.
3. Crea un nuovo stream:
   - **Stream Name**: un identificativo univoco (ad es. `my-game-demo`)
   - **Build Target**: scegli il tuo ambiente di destinazione (ad es. Development)
   - **Upload Binaries**: trascina i file del tuo gioco compilato
   - **Choose Network**: seleziona la rete che preferisci (ad es. Test Net)
4. Controlla la configurazione e clicca **Pay + Deploy**.
5. Attendi che i file del server e del client finiscano di caricarsi. Non aggiornare la pagina e non uscire durante il caricamento.

Una volta distribuito, riceverai un **Game ID** e uno **Stream ID**.

Passo 2: incorpora lo stream

Aggiungi questo iframe alla tua pagina web:

```
<iframe
  src="https://player.yom.net?gameId=YOUR_GAME_ID&streamId=YOUR_STREAM_ID&origin=https://yoursite.com"
  width="1280"
  height="720"
  frameborder="0"
  allowfullscreen
  allow="autoplay *; microphone *; clipboard-write *; pointer-lock"
  style="border: none;">
</iframe>
```

Sostituisci `YOUR_GAME_ID`, `YOUR_STREAM_ID` e `https://yoursite.com` con i tuoi valori reali.

Ecco fatto. I giocatori visitano la tua pagina, cliccano play e il gioco viene trasmesso direttamente al loro browser.

Parametri facoltativi dell'iframe

Aggiungili all'URL `src` dell'iframe come parametri di query:

Parametro	Valori	Descrizione
`guestName`	stringa	Precompila il nome del giocatore
`requestMic`	`true` / `false`	Richiede l'accesso al microfono
`forceMic`	`true` / `false`	Richiede il microfono per procedere
`fullscreen`	`none` / `window` / `native`	Modalità a schermo intero (`window` funziona su iOS)
`autoInitiate`	`true` / `false`	Salta il pulsante play, avvia subito
`requiresName`	`true` / `false`	Richiede l'inserimento del nome del giocatore (predefinito: true)
`lang`	codice ISO-639-1	Codice della lingua (ad es. `eng`)
`publisher`	stringa	Nome del publisher per le analytics
`coppa`	`0` / `1`	Flag di conformità COPPA
`genre`	stringa	Genere del gioco per il targeting pubblicitario

Sicurezza: header CSP

Se il tuo sito usa header di Content Security Policy, consenti l'origine del player YOM:

```
Content-Security-Policy: frame-src 'self' https://player.yom.net;
```

Oppure tramite un meta tag:

```
<meta http-equiv="Content-Security-Policy" content="frame-src 'self' https://player.yom.net;">
```

Risoluzione dei problemi

Se riscontri problemi:

1. Controlla la console del browser per eventuali messaggi di errore.
2. Verifica che il tuo `gameId` e il tuo `streamId` siano corretti.
3. Assicurati che i tuoi header CSP consentano `frame-src https://player.yom.net` .
4. Conferma che il binario del tuo gioco sia stato compilato per Linux x86_64.

E dopo?

- **Integrazione web:** Aggiungi UI personalizzata, pagine multi-stream e comunicazione dal web al gioco usando lo script di embed di YOM.

Per assistenza, scrivici a info@yom.net.

integrazione js

Lo script di embed di YOM (`yom-embed.js`) ti dà il pieno controllo sull'esperienza di streaming. Gestisce la creazione dell'iframe, gli eventi del ciclo di vita del player, il comportamento a schermo intero, l'attivazione e disattivazione della visibilità e la comunicazione bidirezionale tra la tua pagina web e il gioco.

Per iniziare

Includi lo script di embed nella tua pagina:

```
<script src="https://cdn.yom.net/yom-embed.js"></script>
```

Poi aggiungi gli elementi HTML richiesti:

```
<!-- Player container: the iframe will be injected here -->
<div class="yom-player-placeholder"
     data-yom-stream-id="1"
     data-game-id="YOUR_GAME_ID"
     style="width: 100%; aspect-ratio: 16/9;">
</div>

<!-- Launch button -->
<input class="yom-name-input" data-yom-stream-id="1" placeholder="Your name">
<button class="yom-launch-button" data-yom-stream-id="1">Play</button>
```

Quando il giocatore clicca il pulsante, `yom-embed.js` crea un iframe dentro il div segnaposto, si collega a YOM e avvia lo stream del gioco.

Concetti fondamentali

Classi CSS richieste

Classe	Scopo
`.yom-player-placeholder`	Contenitore in cui viene creato l'iframe dello stream
`.yom-launch-button`	Pulsante che avvia lo stream
`.yom-name-input`	Campo di testo facoltativo per il nome del giocatore

Attributi data richiesti

Attributo	Su	Descrizione
`data-yom-stream-id`	Tutti gli elementi	Collega gli elementi a una specifica istanza di stream
`data-game-id`	`.yom-player-placeholder`	L'identificativo univoco del tuo gioco

Attributi data facoltativi

Impostali su `.yom-player-placeholder` per configurare il comportamento:

Attributo	Valori	Descrizione
`data-request-mic`	`true` / `false`	Richiede l'accesso al microfono
`data-force-mic`	`true` / `false`	Richiede il microfono per procedere
`data-fullscreen`	`none` / `window` / `native`	Modalità a schermo intero (predefinito: `none`)
`data-auto-initiate`	`true` / `false`	Avvio automatico senza clic sul pulsante
`data-require-name`	`true` / `false`	Richiede l'inserimento del nome (predefinito: `true`)
`data-hook`	Selettore CSS	Applica la classe `.yom-active` a questo elemento durante lo stream
`data-callback`	Nome di funzione	Validazione personalizzata prima dell'avvio
`data-lang`	ISO-639-1	Codice della lingua
`data-publisher`	stringa	Nome del publisher
`data-coppa`	`0` / `1`	Conformità COPPA
`data-genre`	stringa	Genere del gioco

Esempio completo

```
<!DOCTYPE html>
<html>
<head>
  <meta http-equiv="Content-Security-Policy"
        content="frame-src 'self' https://player.yom.net;">
  <style>
    .yom-toggle-visibility { transition: opacity 0.3s ease; }
  </style>
</head>
<body>
  <!-- Stream container -->
  <div class="yom-player-placeholder"
       data-yom-stream-id="demo"
       data-game-id="my-game"
       data-fullscreen="window"
       style="width: 100%; aspect-ratio: 16/9;">
  </div>

  <!-- Pre-stream UI (hidden when stream starts) -->
  <div class="yom-toggle-visibility" data-yom-stream-id="demo">
    <input class="yom-name-input" data-yom-stream-id="demo" placeholder="Enter your name">
    <button class="yom-launch-button" data-yom-stream-id="demo">Start Game</button>
  </div>

  <!-- In-stream controls (shown when stream starts) -->
  <div class="yom-toggle-visibility" data-yom-stream-id="demo" style="display: none;">
    <button onclick="window.postToPlayer('demo', 'SpawnBot', {amount: '1'})">Add Bot</button>
    <button onclick="window.postToPlayer('demo', 'ClearBot', {})">Clear Bots</button>
  </div>

  <script src="https://cdn.yom.net/yom-embed.js"></script>
</body>
</html>
```

E dopo?

- **Autenticazione personalizzata:** Aggiungi una validazione prima dell'avvio dello stream
- **Gestione degli eventi:** Invia comandi al gioco e ascolta gli eventi del gioco
- **Layout dinamici:** Attiva o disattiva la visibilità della UI in base allo stato dello stream
- **Hook UI semplici:** Modalità a schermo intero, ridimensionamento automatico e hook dei contenitori
- **Configurazione multi-stream:** Esegui più stream di gioco su una sola pagina

autenticazione personalizzata

Puoi eseguire una validazione personalizzata prima che lo stream parta, ad esempio controllando una password, verificando un token o subordinando l'accesso all'invio di un modulo.

Configurazione

Aggiungi `data-callback` al pulsante di avvio con il nome della tua funzione di validazione:

```
<div class="yom-player-placeholder"
     data-yom-stream-id="1"
     data-game-id="my-game"
     style="width: 100%; aspect-ratio: 16/9;">
</div>

<input type="text" class="yom-name-input" data-yom-stream-id="1" placeholder="Your name">
<input type="password" id="password" placeholder="Enter password">
<button class="yom-launch-button"
        data-yom-stream-id="1"
        data-callback="validatePassword">
  Start Game
</button>
```

Definisci la callback

La tua funzione riceve due parametri: `onSuccess` e `onFailure`. Chiamane uno in base al risultato della validazione.

```
function validatePassword(onSuccess, onFailure) {
    const password = document.getElementById('password').value;

    if (password === 'correct-password') {
        onSuccess();   // Stream launches
    } else {
        onFailure();   // Launch is cancelled, button resets
    }
}
```

La callback viene invocata quando il giocatore clicca il pulsante di avvio, **prima** che l'iframe venga creato. Se la funzione genera un errore, l'avvio viene annullato e l'errore viene registrato nel log.

Casi d'uso

- **Accesso protetto da password:** Limita gli stream agli utenti che hanno un codice
- **Validazione di moduli:** Richiedi la compilazione dei campi prima dell'avvio
- **Autenticazione esterna:** Chiama la tua API per verificare l'utente, poi chiama `onSuccess()` nel gestore della risposta
- **Verifica dell'età:** Chiedi una conferma dell'età prima di consentire il gioco

gestione degli eventi

YOM supporta la comunicazione bidirezionale tra la tua pagina web e lo stream di gioco. Invia comandi al gioco e ascolta gli eventi che tornano indietro.

Invio di comandi al gioco

Usa `window.postToPlayer()` per inviare comandi a uno stream in esecuzione:

```
window.postToPlayer(streamId, commandName, data);
```

Parametro	Tipo	Descrizione
`streamId`	stringa	Deve corrispondere al `data-yom-stream-id` dello stream di destinazione
`commandName`	stringa	Il nome della funzione registrata nel tuo gioco
`data`	oggetto	Coppie chiave-valore passate alla funzione del gioco

Esempio: controlli dei bot

```
<button onclick="window.postToPlayer('1', 'SpawnBot', {amount: '1'})">Add Bot</button>
<button onclick="window.postToPlayer('1', 'RemoveBot', {amount: '1'})">Remove Bot</button>
<button onclick="window.postToPlayer('1', 'ClearBot', {})">Clear All Bots</button>
```

Esempio: impostazioni della camera

```
window.postToPlayer('1', 'SetCameraSensitivityX', { MouseSensitivityX: '0.5' });
window.postToPlayer('1', 'SetCameraSensitivityY', { MouseSensitivityY: '0.5' });
```

Esempio: controllo dello stream

```
window.postToPlayer('1', 'pause', {});
window.postToPlayer('1', 'resume', {});
```

Ascolto degli eventi dal gioco

Il player YOM invia eventi alla tua pagina tramite `postMessage`. Ascoltali sull'oggetto `window`:

```
window.addEventListener('message', function(event) {
    // Only trust messages from the YOM player
    if (event.origin !== 'https://player.yom.net') return;

    const { event: eventName, streamId, data } = event.data;

    switch (eventName) {
        case 'playerReady':
            console.log(`Stream ${streamId} is ready`);
            break;
        case 'streamActive':
            console.log(`Stream ${streamId} is now playing`);
            break;
        case 'streamClosed':
            console.log(`Stream ${streamId} ended`);
            break;
        case 'error':
            console.error(`Stream ${streamId} error:`, data.message);
            break;
    }
});
```

Eventi disponibili

Evento	Quando scatta
`playerReady`	L'iframe dello stream è caricato e pronto
`streamActive`	Il gioco è partito, il giocatore è nell'esperienza
`streamClosed`	Il giocatore è uscito o lo stream è terminato
`error`	Qualcosa è andato storto
`requestFullscreen`	Il player ha richiesto lo schermo intero
`exitFullscreen`	Il player è uscito dallo schermo intero

Integrazione con Google Tag Manager

Tutti gli eventi del player vengono inviati automaticamente a `window.dataLayer` per l'integrazione con GTM/GA:

```
// Automatically pushed by yom-embed.js
window.dataLayer.push({
    'event': 'streamActive',
    'category': 'yom-event',
    'streamId': '1',
    'sessionId': 'abc123'
});
```

Configura il tuo container GTM per catturare gli eventi della categoria `yom-event` ai fini del tracciamento analytics.

layout dinamici

Mostra e nascondi gli elementi della pagina in base al fatto che uno stream sia attivo. Così puoi costruire UI di benvenuto pre-stream che scompaiono quando il gioco parte, e controlli in-stream che compaiono durante il gameplay.

Attivazione della visibilità

Aggiungi la classe `.yom-toggle-visibility` e un `data-yom-stream-id` a qualsiasi elemento che vuoi attivare o disattivare:

```
<!-- Visible before stream, hidden during stream -->
<div class="yom-toggle-visibility" data-yom-stream-id="1">
    <h2>Welcome! Click below to start playing.</h2>
    <button class="yom-launch-button" data-yom-stream-id="1">Play Now</button>
</div>

<!-- Hidden before stream, visible during stream -->
<div class="yom-toggle-visibility" data-yom-stream-id="1" style="display: none;">
    <p>Game is running. Use the controls below:</p>
    <button onclick="window.postToPlayer('1', 'SpawnBot', {amount: '1'})">Add Bot</button>
</div>
```

Come funziona:

1. Quando lo stream diventa attivo (`playerReady`), gli elementi si invertono: quelli visibili sfumano fino a sparire, quelli nascosti compaiono con una dissolvenza.
2. Quando lo stream si chiude (`streamClosed`), gli elementi tornano al loro stato originale.

Transizione CSS

L'animazione di dissolvenza dura 300ms. Aggiungi questo CSS per rendere fluida la transizione:

```
.yom-toggle-visibility {
    transition: opacity 0.3s ease;
}
```

Hook dei contenitori

Usa `data-hook` su `.yom-player-placeholder` per applicare una classe `.yom-active` a qualsiasi elemento contenitore mentre lo stream è in esecuzione:

```
<div id="game-section">
    <div class="yom-player-placeholder"
         data-yom-stream-id="1"
         data-game-id="my-game"
         data-hook="#game-section"
         style="width: 100%; aspect-ratio: 16/9;">
    </div>
</div>
```

Ora puoi dare uno stile allo stato attivo con il CSS:

```css
#game-section.yom-active {
    background: #000;
    padding: 0;
    max-width: 100%;
}
```

La classe `.yom-active` viene aggiunta quando lo stream parte e rimossa quando termina, dandoti pieno controllo CSS sui cambiamenti di layout.

hook ui semplici

Lo script di embed di YOM gestisce in automatico lo schermo intero, il ridimensionamento del rapporto d'aspetto e il ciclo di vita dello stream. Ecco come configurare ciascun comportamento.

Modalità a schermo intero

Imposta la modalità a schermo intero tramite `data-fullscreen` sul tuo `.yom-player-placeholder`:

```
<div class="yom-player-placeholder"
     data-yom-stream-id="1"
     data-game-id="my-game"
     data-fullscreen="window">
</div>
```

Modalità	Comportamento	Supporto iOS
`none`	Nessuno schermo intero (predefinito)	N/A
`window`	Schermo intero basato su CSS (`position: fixed` , riempie il viewport)	Sì
`native`	Fullscreen API (vero schermo intero, nasconde la UI del browser)	No (ripiega su `window`)

La **modalità window** è consigliata per il supporto multipiattaforma. Funziona su iOS (dove la Fullscreen API nativa non è disponibile) e gestisce le trasformazioni CSS degli elementi antenati che altrimenti romperebbero `position: fixed` .

Ciclo di attivazione dello schermo intero

Quando il giocatore richiede ripetutamente lo schermo intero, le modalità si alternano in ciclo:

- Desktop: `none` → `window` → `native` → `none`
- iOS: `none` → `window` → `none` (native viene saltata)

Ridimensionamento automatico

Lo stream si ridimensiona automaticamente per riempire il proprio contenitore mantenendo il corretto rapporto d'aspetto. Quando la modalità a schermo intero cambia o la finestra viene ridimensionata, lo script di embed notifica al player le nuove dimensioni del viewport, e la risoluzione del gioco si adegua di conseguenza.

Nessuna configurazione necessaria. Funziona subito.

Ciclo di vita dello stream

Quando lo stream è attivo, lo script di embed gestisce automaticamente diverse cose:

1. **Creazione dell'iframe:** Creato dentro `.yom-player-placeholder` all'avvio, con i permessi corretti (`autoplay` , `microphone` , `clipboard-write` , `pointer-lock`).
2. **Dissolvenza in apertura dell'opacità:** L'iframe appare gradualmente in 300ms una volta pronto.
3. **Attivazione della visibilità:** Gli elementi con `.yom-toggle-visibility` vengono attivati o disattivati all'avvio e alla chiusura dello stream.

4. **Hook dei contenitori:** L'elemento indicato da `data-hook` riceve la classe `.yom-active`.
5. **Notifiche del viewport:** Al ridimensionamento o al cambio di schermo intero, il player viene informato delle nuove dimensioni.

Avvio automatico

Salta del tutto il pulsante di avvio e fai partire lo stream non appena la pagina si carica:

```
<div class="yom-player-placeholder"
     data-yom-stream-id="1"
     data-game-id="my-game"
     data-auto-initiate="true">
</div>
```

Il nome del giocatore può essere precompilato tramite il parametro iframe `guestName` o il campo `.yom-name-input`.

Osservabilità

Lo script di embed include un'integrazione Grafana Faro nativa per il monitoraggio degli utenti reali. Log della console, errori e tracce vengono catturati automaticamente, senza alcuna configurazione da parte tua.

Gli errori vengono inoltre segnalati al backend di YOM a scopo di debug.

configurazione multi-stream

YOM supporta più stream di gioco indipendenti su una singola pagina. Ogni stream ha il proprio contenitore, il proprio pulsante di avvio e il proprio canale di comunicazione.

Configurazione

Assegna a ogni stream un `data-yom-stream-id` univoco:

```
<!-- Stream 1 -->
<div class="yom-player-placeholder"
     data-yom-stream-id="1"
     data-game-id="racing-game"
     style="width: 100%; aspect-ratio: 16/9;">
</div>
<input class="yom-name-input" data-yom-stream-id="1" placeholder="Name for Racing">
<button class="yom-launch-button" data-yom-stream-id="1">Start Racing</button>

<!-- Stream 2 -->
<div class="yom-player-placeholder"
     data-yom-stream-id="2"
     data-game-id="fps-game"
     style="width: 100%; aspect-ratio: 16/9;">
</div>
<input class="yom-name-input" data-yom-stream-id="2" placeholder="Name for FPS">
<button class="yom-launch-button" data-yom-stream-id="2">Start FPS</button>
```

Ogni stream opera in modo indipendente, con avvio, arresto e comunicazione separati.

Invio di comandi a stream specifici

Usa `window.postToPlayer()` con l'ID dello stream di destinazione:

```
// Send command to stream 1
window.postToPlayer('1', 'SpawnBot', { amount: '1' });

// Send command to stream 2
window.postToPlayer('2', 'SetCameraSensitivityX', { MouseSensitivityX: '0.3' });
```

Ascolto degli eventi da stream specifici

Gli eventi del player includono uno `streamId`, così puoi distinguere da quale stream provengono:

```
window.addEventListener('message', function(event) {
    if (event.origin !== 'https://player.yom.net') return;

    const { event: eventName, streamId } = event.data;

    if (eventName === 'streamActive') {
        console.log(`Stream ${streamId} started playing`);
    }
});
```

Attivazione della visibilità per stream

Le attivazioni di visibilità hanno come ambito il rispettivo `data-yom-stream-id`:

```
<!-- Only toggles when stream 1 changes state -->
<div class="yom-toggle-visibility" data-yom-stream-id="1">
    Controls for Racing Game
</div>

<!-- Only toggles when stream 2 changes state -->
<div class="yom-toggle-visibility" data-yom-stream-id="2">
    Controls for FPS Game
</div>
```

Casi d’uso

- **Portali di gioco:** Mostra più giochi su un’unica landing page
- **Pagine di confronto:** Lascia che i giocatori provino due esperienze fianco a fianco
- **Aggregatori di contenuti:** Cura stream di publisher diversi
- **Eventi:** Fai girare più stream di tornei su un’unica dashboard

SEZIONE

informazioni

allocazione dei token

Attributo	Valore
Token totali	750,000,000
Smart contract	0xb6314518b61b4864162c7aE7fdc36261e0A14C4b
Chain di origine	Avalanche C-Chain
Chain di settlement	Avalanche C-Chain
Attivo dal	5 giugno 2026

Dove va ogni tariffa di sessione

Ogni tariffa di sessione si divide a condizioni fisse: 60% all'operatore la cui GPU ha servito la sessione, 30% allo sviluppo della rete, 5% condiviso con tutti coloro che conservano il proprio pagamento e 5% ritirato in modo permanente.

Allocazione dei token

Categoria	Allocazione	Token	Calendario di sblocco
Tesoreria	37.50%	281,250,000	20% al TGE, cliff di 9 mesi, 36 mesi lineari
Ecosistema	25.00%	187,500,000	20% al TGE, cliff di 6 mesi, 24 mesi lineari
HODL	17.31%	129,790,869	cliff di 9 mesi, 16 mesi lineari
Pool di liquidità	9.72%	72,934,111	35% al TGE, 24 mesi lineari
Vendita Seed	2.00%	15,000,000	cliff di 3 mesi, 10% sbloccato al cliff, 6 mesi lineari
Vendita privata	1.50%	11,250,000	cliff di 3 mesi, vesting lineare di 12 mesi
Round KOL	0.48%	3,571,429	20% al TGE, cliff di 3 mesi, 6 mesi lineari
Team	3.00%	22,500,000	cliff di 9 mesi, 20 mesi lineari
Community	2.78%	20,860,735	cliff di 3 mesi, 5 mesi lineari
Exchange	0.03%	200,000	100% al TGE
Round della community (i)	0.10%	714,286	10% al TGE, cliff di 3 mesi, 6 mesi lineari
Round della community (ii)	0.10%	714,286	10% al TGE, cliff di 3 mesi, 6 mesi lineari
Round della community (iii)	0.10%	714,286	10% al TGE, cliff di 3 mesi, 6 mesi lineari
Quest	0.40%	3,000,000	cliff di 3 mesi, 2 mesi lineari

- **Tesoreria (37.50%)**: Alimenta l'adozione, le quotazioni sugli exchange, le sovvenzioni agli studi e le operazioni quotidiane.

- **Ecosistema (25.00%)**: Riservato agli operatori di nodi: ricompense di bootstrap che si riducono man mano che il pagamento di sessione copre gli operatori, più airdrop, programmi per ambassador e sovvenzioni e incentivi di liquidità.
- **HODL (17.31%)**: Detentori dedicati che si sono impegnati a conservare per un periodo molto lungo, con un calendario di vesting esteso.
- **Pool di liquidità (9.72%)**: Dedicato alla liquidità su DEX ed exchange. Garantisce pool di trading profondi su sedi centralizzate e decentralizzate.
- **Vendita Seed (2.00%)**: Investitori istituzionali in equity e in token warrant fin dal primo giorno, più investitori seed in token. Cliff di 3 mesi, il 10% si sblocca alla fine del cliff, mentre il resto matura linearmente in 6 mesi.
- **Vendita privata (1.50%)**: Investitori privati in token con vesting esteso. Cliff di 3 mesi senza sblocco alla fine del cliff; l'intero saldo matura linearmente in 12 mesi.
- **Round KOL (0.48%)**: Campagna Key Opinion Leader per partner influencer e KOL ingaggiati per la portata di marketing del lancio del token. Il 20% si sblocca al TGE per una promozione di lancio allineata; il restante 80% matura linearmente in 6 mesi dopo un cliff di 3 mesi.
- **Team (3.00%)**: Allocato tra tutti i membri del team centrale. Matura su un orizzonte lungo per garantire un profondo allineamento con il successo della rete.
- **Community (2.78%)**: Investitori della community che hanno creduto nel progetto fin dall'inizio. Questo pool rappresenta i nostri detentori OG che hanno scelto di conservare e fare il bridge.
- **Exchange (0.03%)**: Residuo riservato ai partner di lancio e alle quotazioni sugli exchange.
- **Round della community (i / ii / iii) (0.10% ciascuno, 0.30% in totale)**: Tre tranche indipendenti di round della community, una per piattaforma partner, aperte a maggio 2026. Matura il 10% al TGE, poi cliff di 3 mesi e maturazione lineare di 6 mesi.
- **Quest (0.40%)**: Alimenta le competizioni in classifica e le partnership con influencer. Supporta le collaborazioni di marketing che fanno crescere notorietà e adozione.

roadmap

YOM sta costruendo l'edge di sessione: capacità GPU accanto a ogni giocatore e a ogni utente AI, venduta per sessione, su GPU che i gamer e le telco già possiedono. Gli operatori servono sessioni reali fin dal primo giorno, e la rete cresce con le sessioni che serve.

Cosa fa girare YOM

Rilasciato e in funzione sulla rete YOM:

Rete e token

- **Rete di produzione**: la rete serve sessioni live ogni giorno
- **Token $YOM**: gli operatori sono pagati in $YOM, con il token attivo su Avalanche. La storia è nell'aggiornamento sul TGE e nell'aggiornamento strategico
- **Ripartizione della tariffa su ogni sessione**: 60% all'operatore che l'ha servita, 30% allo sviluppo della rete, 5% condiviso con tutti coloro che conservano il proprio pagamento e 5% ritirato
- **Paga degli operatori fissata in dollari**: una quota fissa di ogni tariffa di sessione, fissata in dollari e pagata in $YOM

Streaming e giochi

- Gioco immediato nel browser dei giochi Steam ed Epic che possiedi, su qualsiasi dispositivo, senza download
- Streamer universale con video e audio WebRTC e pieno supporto degli input (mouse, tastiera, gamepad, microfono)
- Roughneck Rumble di Skyhook Games gira in streaming sulla rete

Rete di nodi

- **HyperOrch**, l'orchestratore edge di YOM, invia ogni sessione al nodo più vicino che ha la capacità GPU richiesta e spazio libero
- **Stack di nodo containerizzato**: il software di nodo gira come container sui server dei partner e sui rack telco, con lo stesso software delle GPU domestiche
- **Edge telco nel Regno Unito**: 168 sessioni di gioco live in 20 giorni su tower site di Cornerstone, la più grande società di torri mobili del Regno Unito, con StonesThro e Zadara
- **Registri dell'esito delle sessioni** e un ciclo di test automatizzato per le release dei nodi
- NANO: collegalo a un PC da gaming, scansiona un codice QR e il PC è pronto a servire sessioni
- Aggiornamenti over-the-air per i dispositivi NANO sul campo
- Pagamenti per sessione agli operatori di nodi
- Più sessioni per GPU, condivise tramite il time-slicing del driver
- Scansione hardware dei rig e verifica automatizzata
- Registrazione e monitoraggio di nodi e sessioni

Strumenti per gli operatori

- Portale operatori app.yom.net
- stats.yom.net, statistiche pubbliche della rete
- Programma referral con tracciamento delle commissioni

Sicurezza

- Penetration test esterno dell'immagine di nodo NANO
- Ogni sessione gira nel proprio container pulito, scartato alla chiusura della sessione
- NANO avvia un sistema di sola lettura dal proprio drive cifrato, che si sblocca solo tramite il servizio di chiavi di YOM
- I nodi NANO controllano i propri file di avvio rispetto a hash noti a ogni avvio, e i nodi che falliscono vengono messi in quarantena automaticamente
- Gli stream sono cifrati in transito (WebRTC, DTLS-SRTP)
- API di backend protetta (rate limiting, rafforzamento dell'autenticazione, validazione degli input)

Il nostro approccio

La roadmap di YOM segue tre principi:

1. **L'economia prima del volume.** Ogni prodotto dimostra il proprio prezzo con un partner a basso volume prima che il volume venga spinto.
2. **Walter per primo su ogni capacità.** Walter, il servizio consumer di YOM, fa girare ogni nuova capacità in sessioni live prima che i partner la acquistino.
3. **Offerta e mercati si aprono per soglie.** I nuovi lotti di NANO e i nuovi mercati si aprono in base all'uso misurato e alla capacità testata. Collega un NANO ed è pronto a servire sessioni; i guadagni seguono le sessioni che serve.

social media

Segui la rete: lanci, numeri live e novità per gli operatori. Ogni canale ha il proprio mix di contenuti e community, e questi sono gli account ufficiali di YOM.

- X
- Telegram
- Discord
- YouTube
- Medium
- LinkedIn
- Instagram

faq

1. Domande generali

Che cos’è YOM?

YOM mette le GPU accanto a ogni utente: una rete edge decentralizzata che trasforma le GPU consumer inattive e i rack telco in un’unica griglia di calcolo, senza nulla prenotato in anticipo. YOM vende sessioni gestite in tempo reale, due prodotti su un’unica rete, entrambi pagati per sessione: **YOM Games**, il game streaming live per i giocatori tramite Walter, per i publisher e per le piattaforme che trasmettono giochi ai propri utenti; e **YOM AI**, sessioni AI live, prima di tutto la voce, per le aziende che fanno girare propri modelli di linguaggio e di voce. YOM collega le risorse GPU inattive di nodi in self-hosting e delegati a studi, brand ed emittenti, a una frazione del costo del cloud centralizzato.

Come funziona YOM?

YOM gira su una rete di macchine da gaming (nodi) che eseguono il nostro **Node OS** su misura, e trasmettono sessioni agli utenti finali. Ogni sessione gira sullo stack gestito di YOM nel proprio container, creato per quella sessione e scartato quando si chiude. Giocatori, publisher, piattaforme e aziende AI pagano per sessione. L’operatore la cui GPU ha servito una sessione guadagna il 60% della sua tariffa: una quota fissa, fissata in dollari e pagata in $YOM.

Chi può trarre vantaggio da YOM?

- **Operatori di nodi:** Guadagna il 60% di ogni sessione che la tua GPU serve.
- **Membri della community:** Guadagna XP, conserva il tuo pagamento per partecipare al pool della community e vota sul programma della supply quando si apriranno le votazioni dei detentori di token.
- **Studi e piattaforme:** Usa YOM Games per trasmettere i tuoi giochi su qualsiasi browser, accanto ai tuoi giocatori, a una frazione del costo di un data center.
- **Aziende AI:** Usa YOM AI per far girare i tuoi modelli vocali in sessioni live accanto a chi parla.

2. Operatori di nodi

Come posso iniziare a far girare un nodo YOM?

1. **Self-hosting (disponibile ora):**
   - Collega un **NANO** al tuo PC da gaming, oppure configura il nodo tramite **installazione manuale** sul tuo rig da gaming.
2. **Licenze extra:**
   - Le licenze mettono in servizio più capacità della tua macchina. Una nuova licenza costa US$125.

Quali sono i requisiti hardware per il self-hosting?

- **GPU:** Qualsiasi configurazione NVIDIA RTX (2060RTX e superiori) con molta VRAM per massimizzare la capacità del nodo.
- **CPU:** Intel i5/i7 o AMD Ryzen 7/9 per un multitasking e una stabilità migliori.
- **RAM:** 16GB sarebbero sufficienti nelle configurazioni più modeste, ma si consigliano 32GB+.
- **Archiviazione:** Nessuna archiviazione necessaria per la soluzione plug and play. Per un deployment di avvio Linux personalizzato: servono 1TB+ di spazio disponibile.
- **Internet:** Connessione Ethernet cablata con almeno 20 Mbps di upload misurato.

Quanto posso guadagnare facendo girare un nodo?

I guadagni dipendono dalle sessioni che il tuo nodo serve. All'obiettivo di utilizzo della rete del 50%:

- **Per NANO:** circa ~US$53 al mese prima dell'elettricità.
- **Per host:** calcolato in base alla capacità misurata del tuo hardware. Esegui il controllo PC gratuito per ottenere il tuo risultato.
- **Rientro:** circa ~7 months per un NANO. Le cifre live sono su /operators e seguono ogni release del modello.

Un NANO costa US$349 e include 2 licenze, sufficienti per una sessione Standard; ogni nuova licenza (US$125) sblocca un'unità di capacità, la dimensione di una sessione Light. La paga dell'operatore è una quota fissa di ogni tariffa di sessione, fissata in dollari e pagata in $YOM, e cresce con le sessioni che il tuo nodo serve. Sono stime, non garanzie: la rete sta ancora crescendo verso il suo obiettivo, quindi i guadagni di oggi sono inferiori. Come funzionano capacità e licenze: Licenze dei nodi.

Quali costi devo considerare con il self-hosting?

I costi comprendono:

- **Hardware:** un rig nuovo o il PC da gaming che già possiedi.
- **Elettricità:** l'energia che la tua GPU assorbe mentre serve sessioni.
- **Licenze:** ogni licenza di nodo mette in servizio un'altra unità della capacità della tua macchina.

3. Studi e piattaforme

Perché dovrei usare YOM per il cloud gaming?

YOM offre:

- **Efficienza di costo:** Paghi per sessione, da US$0.04 a US$0.24 in base al livello di calcolo: una sessione Standard fino a un'ora costa US$0.08, contro circa US$1.23 per la stessa ora su una GPU da data center riservata per il picco.
- **Scalabilità:** Usi una rete di nodi in crescita senza investimenti in infrastruttura.
- **Facilità di integrazione:** Incorpori gli stream direttamente in siti web, app e piattaforme con un iframe.
- **Il tuo brand:** Presenti lo stream con il tuo brand.

Come viene garantita la performance di gioco su dispositivi diversi?

HyperOrch, l'orchestratore edge di YOM, invia ogni sessione al nodo più vicino che ha la capacità GPU richiesta e spazio libero. I giocatori entro 50 km da un nodo vedono una latenza mediana in una direzione di circa 15 ms; sotto i 10 ms è l'obiettivo.

Quale modello di prezzo usa YOM?

YOM prezza **ogni sessione** a un prezzo fisso per livello di calcolo, sia per YOM Games sia per YOM AI:

- **US$0.04 / US$0.08 / US$0.16 / US$0.24 per sessione** per una sessione Light / Standard / Premium / Ultra, un gioco o una conversazione AI live; una sessione dura fino a un'ora.
- **Hosting Slot a US$89 al mese:** 2,300 crediti (1,150 sessioni Standard) e un carico di lavoro; l'utilizzo oltre quel limite costa US$0.048 per credito.

Le sessioni possono essere finanziate dalla pubblicità?

Gli annunci giocabili e le sessioni finanziate dalla pubblicità sono nella roadmap, non attivi oggi. Lo stesso streaming incorporabile che alimenta le demo gratuite (Discovery) può ospitare formati pre-sessione o di annunci giocabili, e amplieremo i modelli finanziati dalla pubblicità quando la densità della rete lo permetterà. Fino ad allora i publisher pagano per sessione o tramite l'Hosting Slot, e gli abbonamenti Walter (il Compute Pass) vengono fatturati da gennaio 2027.

---

4. Tokenomics e ricompense

Che cos'è $YOM e come funziona?

$YOM regola ogni sessione sulla rete:

- **Paga degli operatori:** gli operatori sono pagati in $YOM, una quota fissa di ogni tariffa di sessione fissata in dollari.
- **Quota della community:** il 5% di ogni tariffa di sessione è condiviso con tutti coloro che conservano il proprio pagamento.
- **Ritirata:** il 5% di ogni tariffa di sessione viene ritirato in modo permanente.
- **Liquidità:** $YOM ha il suo mercato primario su Avalanche.

Che cosa sono gli XP e qual è il loro ruolo nell'ecosistema?

Gli XP sono un **punteggio di contributo non trasferibile** che si guadagna contribuendo alla rete. Servono a:

- Classificare chi contribuisce nella classifica.
- Pesare i voti dei detentori di token sul programma della supply quando le votazioni si apriranno.

Gli XP non cambiano quali sessioni serve il tuo nodo né quanto guadagna.

La supply di $YOM cresce?

Non vengono coniati nuovi token oltre la supply totale fissa di 750,000,000 $YOM, e il 5% di ogni tariffa di sessione viene ritirato in modo permanente.

Come posso guadagnare di più?

I guadagni seguono le sessioni che il tuo nodo serve:

- Mantieni il tuo nodo online e affidabile, su una connessione cablata.
- Una macchina più capace ha più capacità, e il rig check gratuito la misura.
- Ogni licenza di nodo extra mette al lavoro un'altra unità di quella capacità.

---

5. Governance e community

Come funziona la governance nell'ecosistema YOM?

Il voto dei detentori di token sul programma della supply è il prossimo passo della governance (vedi la roadmap). Quando si aprirà, i detentori voteranno su:

- Tassi di incentivo per gli operatori.
- Allocazione della tesoreria.
- Standard hardware.
- Approvazione delle sovvenzioni.

Chi governa il protocollo oggi, e come cambierà?

Oggi un consiglio di tre rappresentanti dei fondatori gestisce il contratto di settlement della rete e fissa i parametri del programma della supply, l'allocazione della tesoreria e le sovvenzioni. Il voto dei detentori di token viene dopo, con ogni voto ponderato al 67% in base agli XP e al 33% in base alle disponibilità di $YOM, così gli operatori che fanno il lavoro contribuiscono a guidare la rete. I membri del consiglio eletti dalla community iniziano poi a sostituire i rappresentanti dei fondatori.

Come posso partecipare alla community di YOM?

Puoi interagire con YOM attraverso:

- La partecipazione alle discussioni su **Discord e Telegram.**
- Guadagnare XP, che peseranno sul tuo voto quando si apriranno le votazioni dei detentori di token.
- L'organizzazione di eventi della community per accogliere nuovi utenti e ampliare l'ecosistema.

---

6. Aspetti tecnici e risoluzione dei problemi

Cosa devo fare se il mio nodo non compare nella dashboard?

- Assicurati che il tuo wallet sia collegato correttamente.
- Verifica la tua connessione internet e riavvia il sistema.
- Contatta il supporto YOM se il problema persiste.

Come ottimizzo il mio rig per prestazioni migliori?

- Assicura un raffreddamento adeguato e un'efficienza energetica corretta.
- Usa connessioni internet cablate ad alta velocità.
- Monitora regolarmente le metriche di performance dalla dashboard.

chiarimento sui crediti del compute pass

17 settembre 2026

Il prezzo pubblicato del Compute Pass di Walter usa un'allocazione mensile di crediti. Ogni sessione consuma crediti in base al suo livello di calcolo:

Livello di sessione	Crediti per sessione
Light	1
Standard	2
Premium	4
Ultra	6

Questo corregge la spiegazione precedente secondo cui un credito comprava qualsiasi sessione. Un credito copre una sessione Light; Standard, Premium e Ultra ne consumano di più. I giocatori possono spendere l'allocazione mensile su una combinazione di livelli.

Gli abbonamenti pubblicati restano US$4.99 per 30 crediti, US$9.99 per 70 crediti e US$19.99 per 140 crediti al mese. I crediti aggiuntivi costano US$0.175 ciascuno. Ogni sessione può durare fino a 60 minutes.

La previsione ipotizza una durata di sessione di 20 minutes per stimare l'utilizzo e l'economia oraria. L'allocazione mensile si spende per sessione alla tariffa in crediti del livello, e non come un contatore orario. La documentazione sui pagamenti ora distingue quell'ipotesi dal tetto di sessione e rimanda alle specifiche attuali dei livelli.

Walter su walter.gg si può provare gratuitamente. Questa pagina spiega i prezzi degli abbonamenti pubblicati.