

Humanoid Training Loop: First Principles to Commercial Proof

Oct 2, 2026 · @julian gilyadov

Summary

Build a measurement loop first and a robot second: one Unitree G1 humanoid (or the wheeled G1-D if the buyer's task needs no walking), a cheap two-arm station, open models you fine-tune, and an evaluation system that turns every robot-hour into trustworthy numbers. You know you have something potentially commercial when, on a replica of a real buyer's station, the robot succeeds at least 95% of the time with a 90% lower bound, goes an hour or more between human interventions, and the value of its work covers its full cost per hour (robot, supervision, failures) about twice over, and the buyer agrees to pay for a pilot.

The setup.

- **Body:** Unitree G1 EDU with 29 joints and tactile Dex3-1 hands, the only humanoid where the whole open stack already works, plus a YAM two-arm station where manipulation methods iterate cheaply.
- **Control:** NVIDIA's SONIC whole-body controller at 50 Hz on the robot; the learned policy runs on a workstation GPU over Ethernet; every sensor on one clock.
- **Data:** VR teleop (PICO 4 Ultra and trackers) through SONIC, plus robot-free capture rigs; raw MCAP logs, LeRobot v3 datasets, everything indexed in ClickHouse.
- **Models:** GR00T N1.7 as the primary policy and $\pi 0.5$ as the comparison; operator corrections, then RL on the real robot, to climb from "usually" to "almost always".
- **Simulation:** Isaac Lab to train and gate the controller, MuJoCo as a second engine, splat twins of your own cell, and world models only to rank candidates.
- **Evaluation:** your own frozen task suite, blind paired trials with a canary policy, sequential tests, and cheap proxies used only after they match real results.
- **Safety:** hoist, two independent stops, a fenced zone, and the learned policy kept out of the safety chain; pilots run in a guarded cell with an independent field evaluation.

What you optimize. Net value per robot-hour on the buyer's task: throughput relative to a trained person times the buyer's labor cost, minus the robot's full cost per hour. Its four drivers form the dashboard: success with self-recovery, throughput, interventions per

hour times minutes per intervention, and productive hours per day. Public benchmarks are smoke tests.

The thresholds.

Gate	Where	Success	Mean time between interventions	Throughput and hours	Buyer
Working	Your lab	≥80% over ≥50 blind trials	≥10 min	≥40% of a person	—
Potentially commercial	Replica of a buyer's station	≥95%, lower bound ≥90% (96/100)	≥1 h, ≥80% of failures self-recovered	Work worth ≥2.1× its full hourly cost; ≥8 h a day	Signs a paid pilot with their own acceptance test
Commercial	The buyer's site	≥98–99% (99% takes 299 straight successes)	≥1 shift	At takt, or work worth ≥2.1× its full hourly cost; 16–20 h a day	Signs a contract after an independent safety evaluation

Timeline and cost. About a year if every gate passes on time: the measurement loop by week 8, Working by week 18, Potentially commercial by week 30, Commercial at the pilot site by week 42, and a second site by week 52. Hardware for one robot costs about \$143k as specified (\$94k with grippers and no research hand); year one costs about \$0.55M with three hires, a part-time safety contractor and a second robot.

Four conclusions that drive everything else.

1. Measurement comes before scale. Each trial is a coin flip, robot time is scarce and every decision rests on the numbers, so build the evaluation loop before collecting large data (A4, A5, A10).
2. Supervision and hours decide break-even more than robot price. With one minder per robot, even a \$50k robot must outwork a person; at ten robots per supervisor it breaks even at a third of a person's speed.
3. Interventions, not accuracy, set the success rate you need. At 60 attempts an hour, a full shift without a person needs 99.8% success unless the robot recovers from its own failures, so self-recovery is a skill to train and measure.
4. Own the interfaces, data, evaluation and buyer relationship, and rent the rest. Bodies, hands, models and GPUs improve on other people's money, so everything you build must survive swapping them.

Axioms and irreducible constraints

Eleven constraints fix the shape of any humanoid training loop. Each follows from physics, information, statistics or arithmetic, so better models will not remove them. Every later choice in this document is derived from them.

A1. A robot acts only through contact, and contact force is invisible to cameras. A humanoid changes the world only by pushing on it, and force does not show up in pixels. Joint torque, motor current, wrist force sensors and touch sensors are the only ways to know it.

- Consequence: log joint position, velocity and torque (or current) at control rate in every episode. Add measured touch for contact-heavy tasks. A video-only pipeline caps contact precision.

A2. A policy cannot know more than its data and priors contain. No training step creates information about the deployment world that is absent from the data, the pretrained model or the robot's own interaction. In the largest controlled study ([Lin et al., 2024](#): 40,000+ demonstrations, 15,000+ real rollouts), success in new environments followed a power law in the number of distinct environments and objects, while extra demonstrations per environment saturated.

- Consequence: buy diversity (rooms, objects, people) before repetition, and always test in places absent from training.

A3. Closed-loop errors compound. An imitation-trained policy has only seen states the demonstrator visited. One small error pushes it off that path, where errors grow; for behavior cloning the worst-case cost grows with the square of task length ([Ross et al., 2011](#)).

$$p_{\{\text{task}\}} = \prod_{i=1}^n p_i \quad \text{qqquad} 0.99^{20} \quad \text{approx} \quad 0.82 \quad \text{qqquad} 0.95^{1/20} \quad \text{approx} \quad 0.9974$$

- Consequence: a 20-step task at 95% needs every step at 99.7%. The loop needs recovery data (operator interventions), action chunking, and reinforcement learning on the real robot for the last points.

A4. A success rate is a coin you can only estimate by flipping it. Each trial is a binomial draw, so certainty costs trials. At 20 trials and 50% success, the 95% interval runs from 30% to 70%. With zero failures in n trials, the failure rate is below 3/n at 95% confidence.

$$\text{Wilson interval: } \frac{\hat{p} + \frac{z^2}{2n}}{\sqrt{\frac{\hat{p}(1-\hat{p})}{n} + \frac{z^2}{4n^2}}} \pm \frac{z}{\sqrt{n}} \approx \text{rule of three: } p_{\text{fail}} < \frac{3}{n}$$

- Consequence: proving 99% takes about 300 straight successes; 99.9% takes about 3,000. Comparisons must be paired (same start states), pre-registered and sized before running.

A5. Robot time is the binding resource. A trial plus reset takes 1 to 3 minutes, so one robot yields 20 to 60 trials an hour at best. Charging, repairs and resets cut a working day to roughly 100 to 200 trials, and data collection competes for the same hours.

- Consequence: maximise information per robot-hour. Use paired designs, automatic resets, and cheap proxies (offline metrics, simulation, world models) that have been validated against real trials. A second robot often beats a bigger model.

A6. A standing humanoid is an inverted pendulum with a battery. If control or power fails it falls, and falls break hardware and can hurt people. Batteries last a few hours ([NVIDIA's reference humanoid](#): about 3).

- Consequence: a gantry or harness, a wireless stop, and fixed-base experiments for anything that is not about legs.

A7. Control has hard time budgets. A standing G1's fall develops in about a quarter of a second (the pendulum time constant $\sqrt{h/g}$ for a center of mass about 0.7 m up), so balance needs tens of corrections a second. Learned trackers such as SONIC run at 50 Hz and feed joint loops at 500 Hz to 1 kHz. Large models replan only a few times a second: GR00T's G1 workflow defaults to 2.5 Hz, and the world-action model [Cosmos 3](#) Edge takes 1.53 s per action chunk on Jetson Thor. At 1 m/s, 50 ms of delay is 5 cm of error.

- Consequence: a three-layer stack. A slow learned policy emits targets, a whole-body controller tracks them at 50 Hz or faster, and joint loops run at 500 Hz to 1 kHz. All sensor streams need one clock, with every frame's capture time logged.

A8. Hardware drifts and wears out. Leading dexterous hands last 300,000 to 500,000 grip cycles, about a month at parcel-sorting pace (some 1,270 grips an hour, 10 hours a day) and 8 to 14 months at 60 grips an hour, 20 hours a day ([Deep News](#)). Tendons stretch, and two units of the same robot behave differently.

- Consequence: log hardware health on every trial, recalibrate on a schedule, keep spares, and interleave the policies you compare so drift cannot pose as progress.

A9. A robot is commercial only when its value per hour beats its full cost per hour and the alternative.

$$\begin{aligned} \text{cost/h} &= \frac{\text{price}}{\text{service hours}} + \\ &\text{maintenance/h} + \text{supervision/h} + \text{energy/h} + \\ &\text{software/h} \quad \text{value/h} = \text{tasks/h} \times \\ &\text{value/task} \end{aligned}$$

- Consequence: from the first week, measure tasks per hour against a person, interventions per hour (they set supervision cost), uptime and mean time between failures. The numbers are in Commercial thresholds.

A10. Progress is set by the loop's cycle time. If one experiment takes T days, a quarter tests 90/T ideas, so halving T doubles how fast the company learns.

- Consequence: build the measurement loop before collecting large data, and aim for under 48 hours from a change to a decision.

A11. Hardware and compute get cheaper; data, evaluation and customer access

compound. Unitree's entry humanoid fell from 99,000 yuan (G1, 2024) to about \$5,900 (R1, 2025) ([Interesting Engineering](#)). Chinese five-finger hands fell from RMB 50,000–100,000 to about RMB 20,000 ([ITES](#)), and NVIDIA's open model went from GR00T N1 to N1.7 in 13 months.

- Consequence: buy hardware and assume it will be replaced. Keep everything you build independent of one robot: action formats, data pipeline, evaluation suite and task library.

What the loop must contain

The axioms force twelve components; drop any one and the loop either cannot learn, cannot measure, or cannot be trusted. The future-proofing rule: own the interfaces and the measurement, and buy or rent everything that improves on someone else's cost curve.

Component	Forced by	Must deliver	Future-proof choice	Likely obsolete
Robot body	A1, A6, A11	Cameras, joints and torques you can read and command at control rate	A bought, widely used platform with an open low-level SDK and a published simulation model	Building your own humanoid; closed bodies without torque access
Hands	A1, A8	Enough independent joints for the tasks; touch where force matters	Hands with a simulation model and a retargeting map from the human hand	Unmodeled hands that cannot be simulated or retargeted
Sensors and clock	A1, A7	Head and wrist cameras, joint state, force or touch, all on one clock (timestamps within 1 ms)	Hardware timestamps; camera layout that matches the target robot class	Unsynced webcams; video-only logs
Control stack	A6, A7	Balance plus tracking of targets at 50–500 Hz	A learned whole-body controller trained in simulation and commanded by targets	Hand-coded state machines; scripted motions per task
Teleop and capture	A2, A3	Robot demonstrations, operator interventions, scalable human data	VR or glove teleop through the same controller; a human capture rig matched to the robot	Rigs tied to one robot body

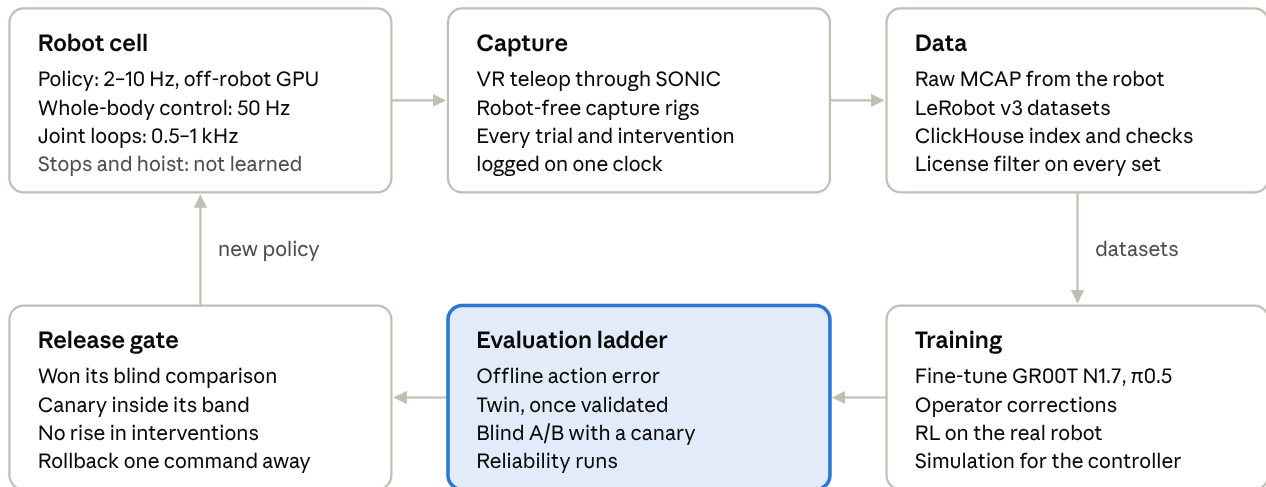
Component	Forced by	Must deliver	Future-proof choice	Likely obsolete
Data infrastructure	A5, A10	Every episode logged, synced, checked and searchable	Open formats (LeRobot v3, MCAP), automatic quality checks, metadata on every trial	Ad hoc folders; ROS 1 bags (ROS 1 reached end of life in May 2025)
Policy model	A2, A3, A11	A pretrained generalist adapted with your data	Open-weight foundation models with relative end-effector actions	Small single-task policies trained from scratch as the product
Reliability training	A3, A9	A way from "usually works" to "almost always"	RL from interventions and outcomes on the real robot; simulation RL for low-level skills	Only adding demonstrations
Simulation and twins	A5, A10	Cheap pre-screening and controller training, checked against reality	GPU physics plus simulated copies of your test cells	Trusting simulation without measured agreement with real trials
Evaluation protocol	A4, A5, A8	Trustworthy numbers per robot-hour	Paired, pre-registered trials with confidence intervals; validated proxies	Demo videos; saturated simulation benchmarks as goals
Safety and facility	A6	Testing without injuries or broken robots	Gantry, wireless stop, marked cell, standards-aware procedures	Untethered testing near people

Component	Forced by	Must deliver	Future-proof choice	Likely obsolete
Commercial instrumentation	A9	Throughput, interventions per hour, uptime and cost per hour from week one	A dashboard tied to one buyer's KPI	Measuring success rate alone

The control stack has three layers, each at its own speed (A7). A learned policy turns images, language and joint state into chunks of targets, replanning a few times a second. A whole-body controller turns targets into joint commands at 50 Hz or faster while keeping balance. Joint loops on the robot run at 500 Hz to 1 kHz, and the motor drivers' current loops run faster still. Keep the interface between the first two layers stable (end-effector and base targets, or motion targets), so either side can be swapped as better ones ship.

Every change reaches the robot only through the evaluation ladder

The robot runs three control layers; everything else is the loop around them.



The loop this document builds · six stages, one release gate

Read it clockwise from the robot: sessions become logged data, data becomes training candidates, candidates climb the evaluation ladder, and only a winner returns to the robot.

What to own, what to buy.

- Own: the data pipeline and schema, the task library and evaluation protocol, the adapters that map any robot's actions into one format, and the relationship with the buyer whose KPI you measure.
- Buy or rent: bodies, hands, base models, physics engines, GPUs. Each is on a falling cost curve driven by companies with far more capital.
- Watch, don't build yet: world-action models, learned simulators and tactile skins. Design the interfaces so they can drop in later.

Robot body

Buy a Unitree G1 EDU with 29 joints and pair it with a \$12,347 YAM two-arm station. The G1 is the only humanoid where the whole open stack already works; on any other body a solo founder spends months porting it. The YAM station is where manipulation methods iterate cheaply.

Why the ecosystem beats the spec sheet (A10). Porting a whole-body controller, teleop, a simulation model and a model adapter to a new body takes weeks to months of work that teaches nothing. That costs more than any \$10,000–30,000 price difference. The G1 is the only body that has all of these at once:

- training data tags in NVIDIA's GROOT N1.7 model (`UNITREE_G1` , `UNITREE_G1_SONIC`);
- NVIDIA's SONIC whole-body controller;
- a native LeRobot driver;
- Isaac Lab tasks and a MuJoCo model that speaks the robot's own command API;
- Unitree's VR teleop.

Buy only a developer (EDU) unit. The base G1, G1+, H2 and R1 cannot be programmed. Only EDU units expose the 500 Hz joint interface (position, velocity, torque and gains per joint) through [unitree_sdk2](#).

Humanoid	Price (US, Oct 2026)	Developer access	Ecosystem	Verdict
Unitree G1 EDU (23–43 joints, ~3 kg per arm, Orin NX, ~2 h battery)	\$43,900 (23 joints) · \$53,900 (29) · \$65,900–67,900 with Dex3-1 hands (RoboStore) · \$74,138 with Inspire hands (Stemfinity)	Full: 500 Hz joint commands, C++/Python, ROS 2	GR00T tags, SONIC, LeRobot, Isaac Lab, MuJoCo, VR teleop	Buy
Unitree H2 EDU (31 joints, 1.82 m, 7 kg rated per arm)	\$66,800; \$67,960–158,200 with Thor and hands	EDU only	Unitree teleop and RL repos; no official SONIC	Second body later; expensive to crash
H2 Plus , NVIDIA's GR00T reference humanoid (75 joints with two Sharpa hands, Thor)	\$100,000 list; up to \$240,000 configured	NVIDIA tools preinstalled	The future reference platform	Watch; ships late 2026
Unitree R1 (20–26 joints)	\$4,900–5,900; EDU price on request	EDU only	Teleop, simulation model	Arms too weak (5 joints), ~1 h battery
Booster T1 / T2	\$41,224 / \$32,999–54,999	Open SDK, ROS 2	Isaac Lab training repos	Locomotion research; T1 arms weak, T2 new
AgiBot X2 / A2	\$27,300 (backordered) / €119,999	Development on the motion computer prohibited	GR00T tag covers AgiBot's wheeled robot, not X2	Avoid as first body

Humanoid	Price (US, Oct 2026)	Developer access	Ecosystem	Verdict
EngineAI PM01 , LimX Oli , Fourier GR-3	\$39,500 / \$21,800 (China) / ~\$80,000 (unverified)	SDKs exist	Thin	Avoid as first body
1X NEO	\$20,000	No SDK	Closed	Avoid
Open kits (Berkeley Humanoid Lite , ToddlerBot)	~\$4,300 in parts	Full	Research code	Locomotion experiments only; K-Scale shut down in November 2025

The cheap manipulation station. Most manipulation learning does not need legs (A6), so iterate on arms and confirm on the humanoid.

Station	Price	Why it matters
I2RT YAM leader-follower pair	~\$12,300 + cameras; turnkey station \$23,999	GR00T N1.7 has a YAM tag; ABC-130K gives 3,553 hours of data and a MuJoCo twin on this exact setup
Trossen Stationary AI	\$23,996	Turnkey ALOHA-style, native LeRobot
Unitree G1-D	\$38,800–66,300	The G1's own arms on a wheeled base with a lift column
Galaxea R1 Pro	\$69,999	EgoScale's robot and the BEHAVIOR challenge platform
xArm 7	\$11,000	A mount for a research hand

Known risks to plan for.

- A [forum report](#) describes broken wrist gears and overheated windings after collisions during RL testing. Budget about 10% of the robot's price for spares.
- [Alias Robotics](#) found a Bluetooth root exploit and telemetry sent every 5 minutes. Keep Unitree robots on an isolated network with Bluetooth off.

- Legs or wheels is still open: wheeled bases with the same arms (G1-D, Galaxea) hedge it.

Hands and touch

Start with Unitree's tactile three-finger Dex3-1 on the G1, and add one high-DoF five-finger research hand on an arm for dexterity experiments. Don't buy a \$50,000 Sharpa first. The Dex3-1 matches the seven joints per hand that GR00T's G1-with-SONIC embodiment outputs, and touch gets logged from day one (A1).

What decides the hand (from the axioms).

- **A1 (force is invisible).** Hands with touch sensors give you the only direct record of contact. Log them even before a model uses them.
- **A8 (wear).** Durability decides how many trials you get. Linkage and direct-drive hands survive collisions better than unmodeled tendon hands.
- **A2 (data).** Human data maps most cleanly onto human-like hands; EgoScale used a 22-joint five-finger hand as its shared action space.
- **The hedge.** Five fingers are not guaranteed: Boston Dynamics' new Atlas hand (1 October 2026) has a thumb and three fingers, 13 joints, and was designed for simulation-trained control.

Hand	Price (Oct 2026)	Active joints	Touch	Simulation and ecosystem	Verdict
Unitree Dex3-1	\$7,680; \$8,680 tactile (SVRC ; the listing doesn't say whether that is per hand)	7, three fingers	33 sensors	URDF; fits GR00T's G1+SONIC tag (7 + 7 hand joints)	First hand on the G1
Unitree Dex1-1 gripper	\$380–1,080	1	—	Native	Cheapest and most reliable if the task allows a gripper
Inspire RH56 (DFX / F1 / E2)	\$10,500 / \$5,500 / \$10,900	6 motors driving 12 joints	8–17 sensors on tactile versions	Isaac Lab ships a G1 + Inspire task; maker claims 10,000 hands delivered in 2025	Five-finger option on the G1 (Ultimate C/D)

Hand	Price (Oct 2026)	Active joints	Touch	Simulation and ecosystem	Verdict
Unitree Dex5-1 / 5-1P	\$25,000 / \$29,000 (per hand or pair unclear)	16 + 4 passive, backdrivable	94 sensors (5-1P)	Unitree stack	Later, when tasks need it
Wuji Hand 2	\$14,600	20, direct-drive, backdrivable	Fingertip force/torque	Ships with SDK, ROS 2, simulation model and teleop	Best research hand to simulate
LEAP v1 / v2 Advanced	\$2,066 kit / ~\$3,000	16 / 17 motors	—	MuJoCo Menagerie model; Dynamixel parts swap in minutes	Cheapest dexterity testbed
ORCA Touch	parts under 2,000 CHF; built \$6,100–7,100	17, tendon-driven	351 taxels	MuJoCo models, MIT license; 10,000 cycles without failure	Good open alternative
BrainCo Revo 2	\$4,700	6 motors, 11 joints	Optional	Unitree teleop support	Budget five-finger
Linker L20 / O6	\$9,999 / \$912–2,000	16 + 5 passive / 6	—	G1 option	Cheap; thin evidence
Sharpa Wave	\$50,000	22	1,000+ taxels per fingertip at 180 Hz	EgoScale's canonical hand; on NVIDIA's reference humanoid; durability numbers are the vendor's	Not for a first rig

Touch sensors to add to any hand.

Sensor	Price	What it gives
AnySkin	\$128	Magnetic skin, replaceable; mounts for LEAP and SO-101
GelSight Mini	\$510	High-resolution contact images; ROS 2
PaXini kit	\$1,299	Six-axis Hall-effect taxels at 1 kHz
Daimon DM-Tac	\$180–280	Low-cost fingertip pads

Sensor	Price	What it gives
Digit 360	Not sold	Free research units through a call for proposals

Decision rule.

- If the buyer's task can be done with a gripper (totes, kitting, part transfer), start with grippers: they're cheapest, most reliable and best covered by open data.
- If the task needs fingers, use the Dex3-1 on the G1 and run dexterity research on a Wuji Hand 2 or LEAP v2 mounted on an arm.
- Record every hand's raw joint state and touch at full rate, so the data can be retargeted to whatever hand wins.

Sensors, onboard compute and the latency budget

Run the learned policy off the robot on a workstation GPU over wired Ethernet. Keep the whole-body controller and the stops on the robot, and put every stream on one hardware clock. This meets the time budget (A7) from day one without a \$5,499 Jetson Thor.

Layer	Rate	Measured latency	Where it runs
Motor drivers	~1 kHz	Firmware	Robot
Joint commands (unitree_sdk2 , EDU)	500 Hz	—	Robot
Whole-body controller (SONIC)	50 Hz	C++ and TensorRT on the G1's Orin	Robot
Policy: GR00T N1.7 , 4 denoising steps, 1 camera	Chunks of 40 actions	27.9 ms on H100 · 38.4 ms on L40 · 93.8 ms on Jetson Thor (TensorRT, ~10.7 Hz)	Workstation now; Thor later
Policy: π0.5	Chunks	132 ms on Thor in BF16; 49–54 ms with TensorRT FP8	Workstation or Thor
World-action models	Chunks	Cosmos 3 Edge : 1.53 s per 32-action chunk on Thor (covering 2.13 s of motion) · DreamZero : ~0.6 s on GB200, ~3 s on H100	Not for real-time control yet

Layer	Rate	Measured latency	Where it runs
Teleop network	—	SONIC guidance : under 10 ms good, 10–30 ms acceptable, over 30 ms "increased stumbling risk"	Wired or dedicated Wi-Fi

GR00T's G1 workflow already uses this split: the policy runs on a remote server at [2.5 Hz by default](#) over ZMQ, and actions stream to SONIC at 50 Hz.

Latency budget. Keep camera exposure to executed action under 150 ms at the 99th percentile, and compute the next chunk while the current one plays. Put a watchdog in your bridge to SONIC: if no fresh chunk arrives in time, command a standing pose and log a fault. Test it on every release by killing the policy server, pulling the cable and adding 100 ms of network delay; the robot must hold its stance, and both stops must still work.

Onboard compute. The G1's Orin NX (16 GB) runs the controller but cannot run a 3B-parameter policy at control rate. NVIDIA [raised the Jetson AGX Thor dev kit](#) from \$3,499 to \$5,499 on 22 July 2026 (T5000 module: \$4,999). Buy one only for untethered runs or to measure on-robot latency.

Workstation GPU. Isaac Sim needs RTX cards (16 GB or more, ray-tracing cores) and [does not support A100 or H100](#), while fine-tuning GR00T N1.7 needs 40 GB or more. A 48–96 GB RTX-class card (RTX 6000 Ada or RTX PRO 6000) does both on one machine. The cheaper alternative is an RTX 5090 (32 GB) for simulation plus rented H100s for fine-tuning.

Sensors to log on every episode.

- Built in on the G1: head depth camera, 3D LiDAR, IMU, joint encoders, motor current, motor temperatures, battery state.
- Add: two wrist cameras on fixed mounts (buy the same models for the human capture rig), and one third-person camera for scoring trials that never feeds the policy.
- From the hands: joint state and touch at full rate.

One clock (A7).

Mechanism	What it achieves
PTP with hardware timestamps	On Thor only hardware PTP works; software transmit timestamps are broken on kernel 6.8.12-tegra

Mechanism	What it achieves
GMSL2 cameras on one capture card	Same electrical trigger, microsecond-level skew
Holoscan Sensor Bridge	Host and sensor agree within 1 μ s; under 20 ms end to end at 4K
USB depth cameras on free-running clocks	Drift a few ms over tens of minutes ; needs a sync cable or trigger

Alarm when timestamp error exceeds 1 ms. Free-running cameras still capture up to half a frame apart (17 ms at 30 fps), so trigger cameras together where alignment matters and log every frame's capture time.

Measure latency, then measure it again after every hardware or software change. Store the results per robot and per software version.

- Camera: film a monitor showing a rolling timestamp QR code ([UMI method](#)).
- Actuators: command a sine wave and fit the lag between commanded and measured motion.
- Software chains: [CARET](#) or ros2_tracing.
- Network: a policy hop adds roughly 13 ms ($\pi 0$ measured 73 ms onboard against 86 ms over one hop).

Whole-body control

Use NVIDIA's SONIC as the bottom layer on the G1, and route both teleop and the learned policy through it. It is the only open humanoid controller with commercial-use weights, active releases (June, July and August 2026) and a ready workflow for GR00T policies. Keep the interface above it in whole-body joint references or end-effector keypoints, because those survive a controller swap.

Why a learned tracker sits at the bottom (A6, A7). Balance must run at 50 Hz or faster and never wait for a 100 ms policy. Learned motion trackers trained in simulation have replaced hand-written control at the leading labs. Figure's System 0 [replaced 109,504 lines of C++](#), and Boston Dynamics [trains Atlas with RL](#). Your policy should say where the hands and body go, not how to stay upright.

Controller	What the layer above sends	Robots	License and activity	Verdict
SONIC (GR00T-WholeBodyControl)	Human pose, G1 motion reference, VR head and hands, or gamepad; encoded to a 64-D token, decoded to 29 joints at 50 Hz	G1 (official)	Code Apache-2.0; weights allow commercial use with attribution; needs exactly TensorRT 10.7	Use first
BeyondMimic	Reference motion clips; a distilled diffusion policy steered by joystick or waypoints	G1	MIT; Isaac Lab or mjlab, then MuJoCo and ROS 2	Best agile results; example checkpoints trained on non-commercial motion data
TWIST2	Root velocity, height and orientation plus whole-body joint positions	G1 + Dex3-1	MIT; built on legacy Isaac Gym	Fast teleop data; motion data license risk
AGILE	Velocity, height, stand and sit, motion tracking	G1, Booster T1	Apache-2.0/BSD-3; maintained	Small community
HANDOFF (June 2026)	Base velocity, height, two wrist targets	G1	Code promised	Larger robust workspace than SONIC (0.31 vs 0.26 m ³)

Controller	What the layer above sends	Robots	License and activity	Verdict
HOVER, OmniH2O, HOMIE, AMO, ExBody2	Various	H1 or G1	Dormant, non-commercial (CC BY-NC) or no code	Avoid

The interface above the controller is the decision that lasts.

Interface	Who uses it	Evidence	Durability
Whole-body joint references (joint angles plus root pose)	Figure Helix 02 (200 Hz into a 1 kHz tracker); OpenHLM ($\pi 0.5$ into SONIC); TWIST2	OpenHLM: 87.5% task progress against 57.5% for GR00T N1.6's decoupled interface, with 1.14 h of demonstrations against 2.70 h, on one task the authors chose	Durable
End-effector and body keypoints	Atlas's behavior model; HuMI (5 Hz into a 50 Hz controller); GR00T N1.7's default relative end-effector actions	Human data maps onto it directly	Durable
Controller tokens	GR00T N1.7 UNITREE_G1_SONIC (64-D token plus 7 + 7 hand joints)	Works out of the box	Tied to one controller checkpoint
Separate navigation and arm commands	GR00T N1.6 UNITREE_G1 (35-D)	Lost to whole-body references in OpenHLM	Fading

Recommendation: start with N1.7's SONIC tag, but log every representation. At each 50 Hz tick, record:

- the raw operator stream (human pose or tracker poses);
- the retargeted joint and root reference;

- the token actually sent;
- the measured robot state;
- the controller checkpoint and gains.

SONIC changed its per-motor gains on 31 August 2026, and a different controller produces a different training distribution. Keypoints can be derived later by forward kinematics, and tokens can be re-encoded from the logged references when SONIC changes, so the token interface is a shortcut you can leave and this data survives whichever interface wins.

Bring-up gates, in order.

1. Simulated gantry: Isaac Sim's [virtual gantry](#) holds the robot while you test.
2. Sim-to-sim: run the same deployment binary in MuJoCo after Isaac Lab, with runtime versions pinned (SONIC requires JetPack 6 and [TensorRT 10.7](#) exactly).
3. Physical hoist: hoist, zero torque, debug mode, default pose, lower slowly, run, then damping to stop.
4. Stops: on SONIC, all four face buttons on the PICO controllers or the O key. A damping stop means the robot falls, so keep the hoist slack short.
5. Match the robot's pose before switching control modes; mismatches "trigger aggressive motions".
6. Falls: learned get-up policies such as [HumanUP](#) for the G1.

Gate to pass: on your robot, 100 held-out motions on the hoist with at most one tracking failure and no fall (SONIC reports 99.2% at 25.7 mm error in its own evaluation), then an hour of test motions in the fenced zone without a fall.

When you train your own tracker (a second body, or a custom skill), start from [mjlab](#) (Apache-2.0, MuJoCo Warp), [unitree_rl_mjlab](#) (G1, H1_2, H2, R1), [holosoma](#) or [ProtoMotions](#). Train only on motion data you are licensed to use commercially.

Teleoperation and data capture

Start with a PICO 4 Ultra headset and two motion trackers (about \$1,000) driving SONIC in whole-body pose mode. That setup produced the best published humanoid demonstrations, and it records through the controller you will deploy. Then add a robot-free capture rig for two to three times the throughput, and add gloves only once finger dexterity is the bottleneck.

The constraints. Demonstrations used as your robot's actions must come through the deployed controller, because a different controller is a different data distribution (A2). Data from people or other robots still helps, but train it as its own embodiment rather than mixing it into your robot's actions. Throughput per hour sets data cost (A5). Lag above about 30 ms makes the robot stumble (A7).

System	Devices	Robots and coverage	Measured	License
SONIC VR teleop	PICO 4 / 4 Pro, 2 controllers, 2 ankle trackers	G1: whole-body pose, planner, or head-and-hands mode	Network under 10 ms good; over 30 ms risks stumbling	Apache-2.0
TWIST2	PICO 4U + 2 calf trackers (~\$1,000); 2-DoF neck (\$250) with ZED Mini (\$400)	G1 + Dex3-1, whole body	Under 0.1 s delay; 100 demos in 15 min on short pick-and-place	MIT
OpenHLM	PICO 4U + 2 leg trackers → GMR retargeting → SONIC	G1 with grippers	40 demos in 21 min of teleop for loco-manipulation, not counting resets (~114 an hour)	Code, data, checkpoints released
Unitree xr_teleoperate v1.6	Vision Pro, PICO 4 Ultra Enterprise, Quest 3	G1, H1, H1_2, H2, R1; Dex1, Dex3, Inspire, BrainCo hands	Upper body; walking via a flag	Open source
Isaac Teleop 1.3	Vision Pro, Quest 2/3/3S, PICO 4 Ultra, MANUS gloves, pedals	Isaac Lab G1 tasks; SONIC path requires a G1 with a Thor backpack	MCAP record and replay; ROS 2 node	Apache-2.0

System	Devices	Robots and coverage	Measured	License
Mocap suits (Xsens Link , Rokoko)	17 inertial sensors at 240 Hz, ~20 ms (Xsens); 200 fps (Rokoko)	Any body via GMR	Xsens ships humanoid teleop software	Price on request

Throughput, from the papers.

- TWIST2: about 400 demonstrations an hour on short pick-and-place.
- OpenHLM whole-body teleop: about 114 an hour.
- HuMI robot-free capture: about 340 an hour (these three rates were compiled by OpenHLM's authors and come from different tasks).
- On HuMI's own whole-body task, HuMI collected 62 episodes per 15 minutes against TWIST2's 28, with 96.7% accepted against 64.3%.

Known failure modes.

- Loose clothing hides the foot trackers (SONIC asks for tight trousers).
- Wi-Fi lag makes the robot stumble.
- Switching modes with a pose mismatch causes aggressive motions.
- Dropped frames record empty human poses.
- Motors overheat during long sessions.

Force feedback matters for contact-heavy dexterity. In one study it raised a teleoperated task from 4/10 to 10/10, but none of these gloves is integrated with an open humanoid whole-body stack yet.

- [DOGlove](#): 21-DoF capture, 5-DoF force feedback, MIT, under \$600 in parts.
- [SenseGlove Nova 2](#): up to 20 N per finger.
- MANUS Metagloves Pro Haptic: vibration only.
- HaptX: microfluidic.

Robot-free capture rigs (more data per hour, no robot time).

Rig	What it is	Evidence
HuMI (Feb 2026)	Two handheld grippers with GoPros plus five Vive Ultimate trackers (grippers, waist, feet); fits in a backpack	70% success in unseen environments; co-training reached 84% task progress in OpenHLM
HumanoidUMI (June 2026)	VR plus handheld grippers, turned into keypoints, then whole-body references	New
EgoScale-style aligned rig	The robot's camera layout, MANUS gloves and Vive wrist trackers	A small aligned set (about 30 human and 5 robot demos per task) made 20,854 hours of human video transfer

Motion data licenses.

- [AMASS](#) bans training networks for commercial use.
- LAFAN1 is CC BY-NC-ND.
- BONES-SEED is free only for academics and companies under \$1M in revenue.

Checkpoints trained on these carry the same risk. Record your own motion with an Xsens suit and retarget it with [GMR](#) (MIT) for data you own outright.

Gates: at least 90% of episodes accepted by quality checks, with demonstrations per operator-hour tracked from the first session.

Data infrastructure

Record raw MCAP on the robot, build LeRobot v3 datasets from it as reproducible artifacts, and index everything in ClickHouse. That matches where the field has converged, and it turns a data-infrastructure background into the loop's speed advantage (A5, A10).

Layer	Choice	Why	Avoid
Raw log	MCAP : per-message nanosecond timestamps, schema per channel, zstd chunks	Immutable record; ROS 2's default bag format since Iron; read by Foxglove, Rerun, FiftyOne and Encord	Ad hoc folders
Training format	LeRobot dataset v3 (Parquet plus AV1 video, episode metadata, streaming; lerobot 0.6.1, August 2026)	GR00T, openpi and RLinf read it; Unitree, NVIDIA and DROID publish in it; depth stored as lossless 12-bit HEVC	Writing new RLDS (archived November 2025) or HDF5
Index, quality checks, analytics	ClickHouse, with the video kept as MP4 on object storage	<code>ASOF JOIN</code> aligns streams at different rates; the <code>s3()</code> table function reads LeRobot Parquet in place; vector index finds duplicates	Video inside the database
Middleware	ROS 2 Jazzy (LTS to May 2029) with <code>rmw_zenoh</code> ; Lyrical Luth (LTS to May 2031) later	Zenoh avoids multicast discovery problems on Wi-Fi; raise its <code>wait_before_drop</code> from 1 ms or large messages drop silently	ROS 1 (end of life May 2025); the 500 Hz joint loop inside ROS
Viewers	Rerun 0.38 (free; reads LeRobot), Foxglove (free 10 GB; Pro \$20/month plus usage), Grafana's ClickHouse plugin	Episode debugging, remote viewing, operations dashboards	Building your own viewer

Layer	Choice	Why	Avoid
Quality checks	Copy the checks in HFlow (Hebbian Robotics; Apache-2.0, pre-v1); label success with Cosmos-Reason2-8B (commercial use allowed, 32 GB+ GPU)	Catches broken episodes before they poison training	Depending on a two-person pre-v1 project

The schema (one table per row type).

- **episodes** : robot and body serial, operator, task, environment, objects, initial-state ID, controller checkpoint and gains hash, calibration version, **license**, success label and its source, intervention list, quality status, MCAP and LeRobot locations.
- **trials** : policy hash, canary flag, session, train or held-out environment, milestones reached, intervention class, battery level, motor temperatures, calibration residual.
- **frames** and **signals** : sequence number, sensor time, receive time, bytes, keyframe flag, file offset; joint and touch signals ordered by robot, episode and time.
- **qa_checks** , **labels** (span, labeling model and version, human-verified flag), **evals** (confidence intervals computed in SQL), **embeddings** .

Checks at ingest.

- Frame gaps over 1.5× the nominal period, and sequence holes.
- Frozen or black frames.
- Timestamps that go backwards, and drifting clock offsets.
- Joint-limit violations, velocity spikes and NaNs.
- Idle time at the start and end of episodes.
- Reprojection error against a fiducial marker on the body (catches calibration drift).
- Duplicate episodes.
- Disagreement between the operator's success flag and a vision-language model's, with humans reviewing every disagreement.
- Usable-episode yield per operator.

Storage. Budget 10–20 GB per robot-hour. Measured rates from published datasets: G1 whole-body teleop with four 640×480 cameras is at most 5.1 GB/h; ABC-130K is 6.5 GB/h; AgiBot World with depth and touch is 16–17 GB/h.

20 TB (~1,000 robot-hours)	Per month	Egress
S3 Standard	\$460	\$0.09/GB: one full read to an outside GPU cloud costs about \$1,800
Cloudflare R2	\$300	Free
Backblaze B2	\$139	Free up to 3× stored
Hugging Face private	\$360	Included

Keep hot copies where you train (egress, not storage, is the real cost) and raw MCAP in cold storage.

Open data, filtered by license. Enforce this through the `license` column when building any training set you might ship.

Usable commercially	Not usable commercially
Unitree's G1 datasets (Apache-2.0) · Humanoid Everyday (Apache-2.0, G1 with touch) · RoboMIND (Apache-2.0, gated) · DROID (CC BY 4.0) · GR00T-Teleop-G1 and GR00T-X-Embodiment-Sim (CC BY 4.0) · ABC-130K · Build AI Egocentric-100K (Apache-2.0) · Lightwheel EgoSuite (training only)	AgiBot World (CC BY-NC-SA) · Galaxea Open-World · Fourier ActionNet · EgoDex · AMASS · LAFAN1 · NVIDIA's GR00T-Teleop-Sim (CC BY-NC)

Models and training

Fine-tune NVIDIA's GR00T N1.7 as the primary policy and keep $\pi 0.5$ as the comparison. GR00T N1.7 has 3 billion parameters, weights licensed for commercial use, a native G1-with-SONIC embodiment tag, and fine-tunes in 2–3 hours on one 48 GB GPU. Then climb from "usually works" to "almost always works" with operator corrections and reinforcement learning (RL) on the real robot, not with ever more demonstrations (A3).

Why a pretrained generalist (A2, A11). A pretrained model brings priors no small dataset can supply. Training from scratch reaches 19% on tasks where Generalist's GEN-1 reaches 99% (company claim). The base models improve every quarter for free, so your effort goes into data, adaptation and evaluation.

Open models you can ship commercially.

Model	Size	Weights license	GPU memory (fine-tune / run)	Speed	Humanoid fit	Evidence
GR00T N1.7	3B	NVIDIA Open Model License (commercial use)	40 GB+ / 16 GB+	27.9 ms on H100; 93.8 ms on Thor	Relative end-effector actions; tags for G1, G1 with SONIC, and YAM	~32,000 h of real and human data plus ~8,000 h simulated; NVIDIA's own evaluations
π0.5 (openpi)	~4B	Gemma license (check commercial terms)	LoRA 22.5 GB+, full 70 GB+ / 8 GB+	49-54 ms on Thor	Drove a G1 through SONIC in OpenHLM	87.5% task progress against 57.5% for GR00T N1.6 on OpenHLM's task
LingBot-VLA 2.0	6B	Apache-2.0	n/a	~130 ms on RTX 4090D	20 robot types incl. humanoids and dexterous hands	60,000 h (50,000 robot, 10,000 human); self-reported wins
Being-H0.5	2B	Apache-2.0	n/a	n/a	Ran on a G1; grippers and dexterous hands	35,000 h incl. 16,000 h of human video
X-VLA	0.9B	Apache-2.0	Small	n/a	Per-robot learned prompt	Cloth folding at 33 folds an hour after 1,200 demos
SmoIVLA	0.45B	Apache-2.0	One GPU (~4 h on an A100)	Asynchronous	Arm-centric	Needs 50+ episodes per task

Model	Size	Weights license	GPU memory (fine-tune / run)	Speed	Humanoid fit	Evidence
Cosmos 3 Edge/ Nano policies	4B / 16B	OpenMDW-1.1 (commercial)	DROID post-training took 17,400 GB200-hours	1.53 s per chunk on Thor	Arm (DROID)	Nano ranked first on RoboArena (NVIDIA)

Not usable commercially, or not downloadable:

- Closed: π 0.6 and π 0.7; GR00T N2 (not released, due by the end of 2026); Gemini Robotics On-Device (trusted testers only); GEN-1 (early access).
- Non-commercial licenses: AgiBot GO-1, InternVLA, Ψ 0 (a G1 model reporting 40% higher success than N1.6), Galaxea G0 Plus and G0.5.
- Unclear: [DreamZero](#) (Apache-2.0 on GitHub, CC BY-NC on its Hugging Face card). [FLUX 3 Action](#) is offered under a non-commercial or a paid commercial license.

Fine-tuning: what the measurements say.

Source	Demonstrations	Compute	What matters
GR00T FAQ	~100 for simple tasks; 500+ multi-step; 2,000+ for high-DoF humanoid tasks; 20–50 per lighting condition	—	Start at ~100, then correction rounds; no zero-shot transfer to new bodies
GR00T N1.7 on a G1 (Isaac Lab-Arena example)	200	One 48 GB RTX 6000 Ada, 20,000 steps, 2–3 h	Language model frozen; the 40-step action horizon must match between training and serving

Source	Demonstrations	Compute	What matters
OpenHLM ($\pi 0.5$ into SONIC)	40 per task; 10/20/40 demos gave 60/85/91%	4 A800s for ~24 h	Starting from $\pi 0.5$ gave 91% against 60% from the plain vision-language model; 10 flow steps beat 1 (87% vs ~67%)
EgoScale	100 per task (20 for deformables); one-shot shirt folding 88%	Pretraining on 256 GB200s	A small aligned set (50 h human, 4 h robot) bridges human video to the robot

Fine-tuning pitfalls:

- Training and serving disagree on action horizon, normalization statistics, or absolute versus relative actions.
- The teleop controller differs from the deployment controller.
- The policy leans on joint state; use state dropout of 0.2–0.8.
- Overfitting; RDT2 advises under 5 epochs.
- Selecting checkpoints on validation loss. It does not track real success; action error on held-out episodes does (Pearson correlation $r = -0.89$ in the ABC study).

From "usually" to "almost always": post-training.

Method	What it does	Data and time	Measured gain
HG-Dagger	Operator takes over when the policy goes wrong; corrections are trained in	After ~100 demos	The standard first loop
Recap ($\pi * 0.6$)	Learns which actions were better than average from demos, rollouts and corrections, and conditions on it	600 demos + 360 corrections per round on 3 robots	Over 2× throughput, about half the failures

Method	What it does	Data and time	Measured gain
RL Token (PI, Apr 2026)	A small actor-critic learns on a summary of the large model's internal state, staying close to its actions	15 min–5 h, 400–1,000 episodes	Screw insertion 20% → 65%; up to 3× faster
HIL-SERL	Per-task RL on the real robot with a learned reward and human interventions	1–2.5 h	Near-perfect success; 2× success and 1.8× faster than imitation
DSRL	RL over the input noise of a frozen diffusion or flow policy	Under 50 episodes	One case 20% → 90%
Simulation RL for large policies (πRL , SimpleVLA-RL)	PPO or binary-reward RL in a simulator	Simulation GPU time	π0.5: 77.1% → 98.3% on LIBERO; only worth it with a faithful twin
Robometer reward model	Scores task progress from video (4B; in LeRobot 0.6)	Trained on 1M+ trajectories	Online RL success 0.85 against 0.55 with RoboReward

[RLinf](#) (Apache-2.0) is the one framework that runs both simulated and real-robot RL for GR00T N1.5–N1.7 and π0/π0.5. It covers PPO, SAC, RLDP, DSRL, Recap, RL Token and HG-Dagger.

Compute (2 October 2026). GPU rents have risen about 40% since the October 2025 low, so budget for rising prices.

GPU	On-demand median	Floor
H100	\$3.49/h	\$1.79/h
H200	\$4.29/h	\$2.60/h
B200	\$6.89/h	\$3.69/h
RTX PRO 6000	\$2.09/h (RunPod)	—

Sources: [GPU price index](#), [RunPod](#).

At those prices a GR00T fine-tune costs under \$10. An OpenHLM-style $\pi 0.5$ run (about 96 GPU-hours) costs about \$170–335, and a full DROID-scale openpi run about \$1,300. Post-training a world-action model like Cosmos 3 Edge on DROID cost 17,400 GB200-hours, at least \$64,000–120,000 even at the B200 rates above, so don't.

Upgrade path.

- GR00T N2 when its weights ship, after checking its license matches N1.7's.
- World-action models once per-chunk latency on Thor drops below about 150 ms and they win blind contact-rich trials.
- EgoScale-style aligned human data once your tasks stabilise.

Likely obsolete within 12–24 months:

- 7B autoregressive or regression policies (OpenVLA-class), and the $\pi 0$, $\pi 0$ -FAST and GR00T N1.5/N1.6 checkpoints.
- LIBERO and SimplerEnv as decision metrics.
- On-policy PPO on hardware.
- Pipelines that generate video first and label it afterwards.
- Unlabeled datasets of absolute joint angles.
- Orin-class onboard compute.

Simulation, digital twins and world models

Use simulation for three jobs:

1. Training and gating the whole-body controller, in Isaac Lab on PhysX now and Isaac Lab 3.0 with Newton once it reaches general release.
2. A MuJoCo sim-to-sim check before anything touches hardware.
3. Gaussian-splat twins of your own test cell, to pre-screen manipulation policies.

Use world models only to rank candidates. Every published simulation-to-real correlation comes from tabletop arms; none exists yet for humanoid whole-body tasks, so you must measure your own (A4, A5).

Engine	Version (date)	License	Throughput	Use it for	Verdict
Isaac Lab on PhysX	2.3.2 stable (Feb 2026); 3.0 early access 16 Sep 2026, general release targeted end of October	BSD-3	G1 rough-terrain training at 82,000 steps/s on one RTX 4090 ; Shadow-hand cube at 170,000	Controller training; Isaac Lab-Arena evaluation	Default
Newton (inside Isaac Lab 3.0)	1.0 GA 16 Mar 2026; now 1.6	Apache-2.0, Linux Foundation	NVIDIA claims 252× (locomotion) and 475× (manipulation) for MuJoCo Warp, baseline unnamed	One task API over PhysX, MuJoCo Warp and others; pressure-patch contact; cloth and cables	Move when 3.0 is GA; expect breaking changes
MuJoCo + MuJoCo Warp + Playground	3.14 (22 Sep 2026)	Apache-2.0	G1 and T1 walking policies train in under 30 min on two 4090s	Sim-to-sim gate; vendor-neutral controller training via mjlab	Second engine
Drake	1.57 (Sep 2026)	Open source	CPU, deterministic	Reference for contact accuracy (TRI's 48-task benchmark)	Specialist
ManiSkill3	3.0.1 (Apr 2026)	Code Apache-2.0; assets CC BY-NC	30,000+ frames/s with RGB-D on one 4090	Vision RL on arms	Optional
Genesis World	1.4.3 (Sep 2026)	Apache-2.0	Claims 10–80× faster; an independent test of the 2024 release measured 0.29M against 43M claimed frames/s	—	Avoid as main engine

Twins of your own cell (for manipulation).

Method	Effort	Agreement with real results
PolaRiS (Gaussian splats into Isaac Sim, open tools)	Under 20 min of human work and under 1 h wall time per environment	$r \approx 0.9$ average (worst 0.81); co-training on ~350 simulated demos closes the visual gap
RialTo (phone-scan twin, RL fine-tuning)	~15 min active work per scene	Over 67% more robust than imitation alone
ABC's MuJoCo twin of the YAM station	Released with the dataset	$r = 0.85$ strict success, 0.91 progress (12 checkpoints)

Generated data that has paid off.

- [HumanoidMimicGen](#) (May 2026): one VR demonstration becomes 1,000 trajectories per task on a G1. Policies reached 89%, against 48% from 100 human demos, and co-training added 20% on hardware.
- RoboCasa365: simulated pretraining raised four real tasks from 61.8% to 79.8%.
- RoboTwin 2.0: 10 real plus 1,000 synthetic demos added 24.4%.
- For visual variety, use Cosmos 3. The [Cosmos Transfer 2.5 repository](#) has been frozen since 1 June 2026.

World models as evaluators: rank, never report.

Model	Access	Agreement with real	Cost
RoboWorld	Paper open	$r = 0.989$ on RoboArena's 8 policies	100 H100-hours for the whole leaderboard
GigaWorld-1	Open (CC BY 4.0)	14.9% better agreement than Cosmos-Predict2.5; consistency of actions matters more than visual realism	1.3B or 5B
Runway GWM-Robotics	By request	$r = 0.95$ on RoboArena	Real-time rollouts
Google DeepMind's Veo evaluator	Internal	$r = 0.88$ on normal scenes, 0.56 with novel objects	—

Model	Access	Agreement with real	Cost
WorldGym	Open	$r = 0.78$	Under 1 GPU-hour

World models do not model balance or forces, and they hallucinate object contact. Use them to order candidates for real trials, never as a success rate.

The rule that keeps simulation honest. Before any proxy influences a decision, measure its agreement with your own real trials on at least 15 checkpoints: the 95% interval of the correlation must stay above 0.5, and the proxy must pick the real winner in at least 80% of pairs that differ by 10 points or more. With 8 checkpoints, an observed r of 0.8 has an interval of 0.22–0.96 and proves little. Checkpoints accumulate from ordinary A/B runs; until 15 agree, use a proxy only to drop clear losers. Re-check after every engine upgrade: NVIDIA's RoboLab authors warn that contact behavior changes between Isaac Sim versions.

Avoid:

- Isaac Gym and `legged_gym` (legacy).
- Genesis as the main engine.
- MJX where throughput matters; MuJoCo's docs say MuJoCo Warp "scales better than MJX on nearly every workload".
- Comparing scores across simulator versions.
- World-model scores as absolute success rates, or for contact-rich or whole-body tasks.

Evaluation

Evaluate on a ladder that spends robot time only on finalists. Each rung is cheaper and less trustworthy than the one above it, and each cheap rung must first be checked against the expensive one (A4, A5).

Rung	Cost	Answers	Trust rule
1. Offline action error on held-out episodes (fixed denoising steps and seed)	Minutes	Which checkpoint within one recipe	ABC: action error tracked real success with a Pearson correlation of $r = -0.89$; validation loss did not ($r = -0.04$)
2. Simulated twin of your cell	GPU-hours	Which recipe or data mix	Use only after it passes the agreement rule in Simulation (15+ of your checkpoints)
3. World-model ranking	~100 H100-hours per leaderboard	Order of candidates	Never report its number as a success rate
4. Paired, blind real trials	Robot-hours	Is B better than A	Pre-registered; interleaved bundles with a canary policy
5. Reliability runs	Robot-days	Is it $\geq 95-99\%$	Consecutive successes; sequential tests
6. Pilot at the buyer's site	Weeks	Does it meet the buyer's KPI	Buyer-defined acceptance test

Methods worth copying.

- TRI's [large-behavior-model study](#) is the design template. It ran:
 - 50 real rollouts per task, policy and condition, blind, from matched initial states;
 - trials in bundles of one rollout per policy in random order, which absorbs drift;
 - Beta posteriors and sequential tests with Bonferroni correction;
 - a milestone rubric, with QA re-scoring 27% of rollouts (2.31% disagreement on success).
- [STEP](#): a sequential test on success and failure that stops early with controlled error; up to 32% fewer trials.
- [N-SCORE](#) (RSS 2026): an anytime-valid sequential test on partial-credit scores; up to 70% fewer trials than a fixed-size test.

- [RoboArena](#): pairwise blind comparisons with progress scores. A Bradley-Terry model with task difficulty ranks policies well within ~100 comparisons.

The formulas.

$$n_{\text{per arm}} = \frac{\left[z_{1-\alpha/2} \sqrt{2 \bar{p} \bar{q}} + z_{1-\beta} \sqrt{p_1 q_1 + p_2 q_2} \right]^2}{(p_2 - p_1)^2} \approx n_{\text{zero failures}} = \frac{\ln 0.05}{\ln p_0} \approx \frac{3}{1 - p_0} \approx T_{\text{robot-hours}} \geq \frac{\chi^2_{0.95, 2k+2}}{2}, M$$

Question	Numbers ($\alpha = 0.05$, 80% power)
Is B better than A? Trials per arm	40→60%: 97 · 80→90%: 199 · 90→95%: 435 · 95→99%: 285 · 98→99.5%: 861
Does it meet target p_0 ? Consecutive successes with zero failures	95%: 59 · 98%: 149 · 99%: 299 · 99.5%: 598 · 99.9%: 2,995
Same, allowing 1 or 2 failures	98%: 236 / 313 · 99%: 473 / 628 · 99.5%: 947 / 1,258
Sequential go/no-go (Wald SPRT, $\alpha = \beta = 0.05$), 90% against 97%	Accept after 40 straight successes; each failure costs about 16 more
Mean time between interventions $\geq M$ hours	Robot-hours needed: $3.0 \cdot M$ with 0 interventions, $4.74 \cdot M$ with 1, $6.30 \cdot M$ with 2 (10 h needs 30 clean robot-hours)

Comparing 90% against 95% at 435 trials an arm and 3 minutes a trial costs about 22 robot-hours per arm. Sequential tests cut that by up to a third (STEP) and partial-credit tests by up to 70% (N-SCORE), so use them for decisions and keep fixed-size tests for claims you publish.

What a week can detect. One robot at 2–3 hours a day runs about 100 trials in two days, or about 50 per arm. That detects 50% → 77% or 70% → 92%, and nothing above 90%. A 10-point gain at 80% needs about 200 trials per arm, a week or more per task; 90% → 95% needs 435 per arm, 3–4 weeks or a second robot. So batch small changes, let validated proxies decide routine ones, and spend real trials on changes big enough to see.

Gate statistics. Gate claims use one-sided 95% exact (Clopper–Pearson) lower bounds: 96/100 gives 91.1%, and 299 successes in 299 trials gives 99.0%. Spread gate trials over at least three days with fresh start states, and bootstrap the interval over sessions, because

trials in one session are correlated. Above 90% success, use exact tests rather than the normal formula above.

Log on every trial.

- Outcome: success on the first try, success after the robot's own retry within the time limit, or failure needing a person; milestones reached; time to completion.
- Interventions, by class: safety stop, fall, assist or reset, remote takeover, hardware fault, each with call, start and end times, so minutes per intervention and waiting time can be computed. Log planned stops (battery changes, scheduled maintenance) separately; they cut productive hours but don't count against MTBI.
- Throughput: attempts per hour, successes per hour, availability.
- Covariates: policy hash, canary flag, session, initial-state ID, train or held-out environment, battery level, motor temperatures, calibration residual.

Control drift (A8).

- Put a frozen canary policy in every bundle and halt when its success rate leaves its control band ($\bar{p} \pm 3\sqrt{\bar{p}\bar{q}/n}$), judged on all canary trials in the week; a CUSUM chart catches slow drift sooner. The five daily pre-flight canary trials only catch a broken setup.
- Analyze with session as a stratum (Cochran–Mantel–Haenszel, or logistic regression with session effects).
- Have a second person score about 25% of rubric labels.

Derived metrics (all computed in SQL from the trial log):

- Interventions per hour = interventions ÷ autonomous robot-hours, and its inverse, mean time between interventions (MTBI), both with Poisson bounds. Falls get their own rate and bound.
- Effective units per hour = attempts per hour × success rate × availability, measured against a trained person at the same station.
- Hardware mean time between failures, tracked separately from the policy. At BMW, Figure's top failure point was the forearm.

Benchmarks

No public benchmark measures what a humanoid buyer pays for, so your own task suite is the primary benchmark. Public benchmarks are smoke tests and a way to earn credibility. LIBERO is saturated: policies scoring 90–95% on it [span the full range of real-robot](#)

[performance.](#)

Benchmark	Type	Status (Oct 2026)	Numbers	Use
LIBERO	Simulation, one arm	Saturated	OpenVLA-OFT 97.1%, Cosmos Policy 98.5%	Smoke test only
LIBERO-PRO / LIBERO-Plus	LIBERO with perturbations	Exposes memorization	Over 90% falls to about 0% when object positions change; ~95% falls below 30% with camera shifts	Robustness check
RoboCasa365	Simulation, mobile manipulator, 365 tasks	Still separates models	Best 58.1% overall, 36.7% on unseen composite tasks	Credibility for arm models
RoboTwin 2.0	Simulation, two arms, 50 tasks	Still separates models	$\pi 0$ drops from 46.4% (easy) to 16.3% (hard)	Bimanual sanity check
RoboLab-120	Simulation, DROID-style arm	Still separates models	$\pi 0.5$ 28.0%, GR00T N1.6 7.2%, Cosmos 3 Nano 36.8%, DreamZero 25.7%	Nightly regression
BEHAVIOR-1K Challenge	Simulation, wheeled humanoid (Galaxea R1 Pro)	Far from solved	2025 winner 26%; 2026 has 100 tasks, deadline 16 Oct 2026	Aim for 2027

Benchmark	Type	Status (Oct 2026)	Numbers	Use
Isaac Lab-Arena G1 Factory; SIMPLE (60 whole-body tasks); HumanoidToolBench (18 G1 tasks, sim and hardware, 1 Oct 2026)	Simulation, humanoid	New	SIMPLE: 80% zero-shot real on pick-and-place and handover	Humanoid regression and credibility
RoboArena	Real, blind pairwise	Active	7 institutions, 4,284 rollouts; DROID Franka arms only	Only if your model also runs on DROID
RoboChallenge	Real robots driven remotely	Active	π 0.5 led Table30 at 43.7%	Arm credibility
AgiBot World Challenge 2026	Online qualifying, real final	ICRA 2026	526 teams; final ran on AgiBot's G2 humanoid; winner 43.47 points	External credibility
HumanoidBench, CALVIN, VLABench	RL benchmark; old PyBullet suite; no leaderboard	Low value for this loop	—	Skip

Design your own suite as the main benchmark. Each task gets a written specification with:

- the start-state distribution, plus at least 20 photographed initial states;
- the success definition and milestone rubric, with a time limit;
- the environment list, split into training and held-out environments;
- the mapping to the buyer's KPI (units per hour, accuracy, cycle time).

Pick 6–8 tasks:

- 2 control tasks that a gripper solves easily, so you can tell model regressions from task difficulty;

- 2–3 tasks that are the buyer's core work;
- 1–2 dexterity stretch tasks;
- 1 loco-manipulation task (walk, pick, carry, place).

Freeze the suite's version, and change it only with a new version number, so results stay comparable over time.

Safety, facility and compliance

Treat every new controller as able to fall (A6). Test it on a gantry with two independent stops inside a fenced zone of about 3 m, keep learned software out of the safety chain (a learned controller may perform the crouch, but no stop may depend on it), and plan for a guarded cell plus an independent field evaluation before any paid deployment. The first humanoid-specific standard, ISO 25785-1, is still a committee draft.

Lab practice.

- **Gantry or hoist** that catches the robot before its knees touch, for every new controller and every power-cut test. A locked pose is unstable without power.
- **Two stops.** A wireless stop for the spotter that commands a crouch or damping through the controller, and a radio emergency stop that cuts motor power without passing through SONIC or any learned software. A power cut drops a biped, so use it with the robot on the hoist, or off the hoist only with nobody inside the fence. Before the first untethered session, ask Unitree how to cut an EDU unit's motor power from outside the robot.
- **Exclusion zone** of at least height + reach + 1 m (about 3 m). The operator and the spotter are different people, so robot sessions need two.
- **Record every stop** as an intervention in the trial log.
- **Simulation first:** a virtual gantry and a sim-to-sim check before the physical hoist (Whole-body control).
- **Network:** wired, or dedicated Wi-Fi under 10 ms. Over 30 ms of lag makes SONIC-driven robots stumble.
- **Security:** isolate Unitree robots on their own network and turn Bluetooth off.
- **Falls:** learned get-up policies (HumanUP), or an arm with a force sensor acting as an active gantry ([Robot-Trains-Robot](#)).

Standards that apply.

Standard	Status (Oct 2026)	What it means for you
ISO/CD 25785-1 , dynamically stable industrial mobile robots	Committee draft registered 8 May 2026; comments closed 8 Jul 2026; publication likely 2027 or later	First humanoid-specific standard; reportedly covers controlled descent on power loss and contact limits
ISO 10218-1/-2:2025 and ANSI/A3 R15.06-2025	Published	Every safety function needs a performance level under ISO 13849. They assume losing power causes no unexpected motion, which a falling humanoid breaks
ISO/TS 15066	Under revision as ISO/AWI 15066-1	Force and pressure limits per body region
ANSI/RIA R15.08	Published	Mobile manipulators; treats instability only as payload tipping
ISO 13849-1:2023 / IEC 61508	Current	Performance level d $\approx$ SIL 2. Balance is itself a safety function, so keep the learned policy out of the safety chain; Agility's Digit uses an independent safety controller, a safety PLC, a Category 1 stop and a pendant
OSHA and NRTL field evaluation	No humanoid standard; production sites need a field audit	Digit was the first humanoid to pass , Nov 2025
EU Machinery Regulation 2023/1230	Applies from 20 Jan 2027, replacing the Machinery Directive (CE marking is required under both)	Covers AI-driven safety functions and cyber-safety
EU AI Act , Annex I high-risk	Applies from 2 Aug 2028	AI safety components of machinery

Standard	Status (Oct 2026)	What it means for you
IEEE Humanoid Study Group report (Sep 2025)	Report	Stop levels from power cut (0) to planned move to a safe state on backup power (3-1); proposes fall-recovery tests

What this means commercially. Humanoids deployed today work behind panels or laser barriers, separated from people. Expect your first pilot in a guarded cell, with an ISO 12100 risk assessment, an independent Category 1 stop at performance level d, and an NRTL field evaluation at each site. Build the performance level d function into the cell, not the robot’s learned software: guard interlocks command the crouch, and a safety relay cuts power after a fixed delay whether or not the crouch has finished. Working beside people without fencing waits for ISO 25785-1 and safety-rated human detection. No humanoid-specific insurance product could be verified; Agility calls being insurable a precondition.

Commercial thresholds

It is potentially commercial when, on a task a buyer already pays people to do, measured success, throughput, interventions and hours put your cost per good unit below theirs with room for both margins, at 95% confidence. In numbers: at least 95% success with a lower bound of 90%, at least an hour between interventions with most failures recovered by the robot itself, and throughput about twice break-even. It is commercial when the same holds at the buyer’s site with 98–99% success and a full shift between interventions.

The inequality (A9). Per robot operating hour, the work done must pay for the robot, its supervision, its failures and both parties’ margins:

$$\eta \backslash, W \backslash ; \backslash ge \backslash ; \backslash frac{1}{(1-g)(1-b)} \backslash left [\backslash frac{P}{L} + \backslash frac{\sigma P}{H} + e + W_s \backslash, \backslash lambda \backslash, t_i + (1-s) \backslash, a \backslash, c_f \backslash right] \quad \eta = \backslash frac{a \backslash, s}{h}$$

- **Value.** η is throughput relative to a trained person: a attempts per hour times success s , over the person’s h units per hour. W is the buyer’s loaded labor cost: \$36.38/h for US transportation and material-moving jobs (BLS, June 2026).
- **Robot.** Price P over a life of L operating hours, yearly service at a fraction σ of the price spread over H operating hours a year, and e for compute, energy and software.

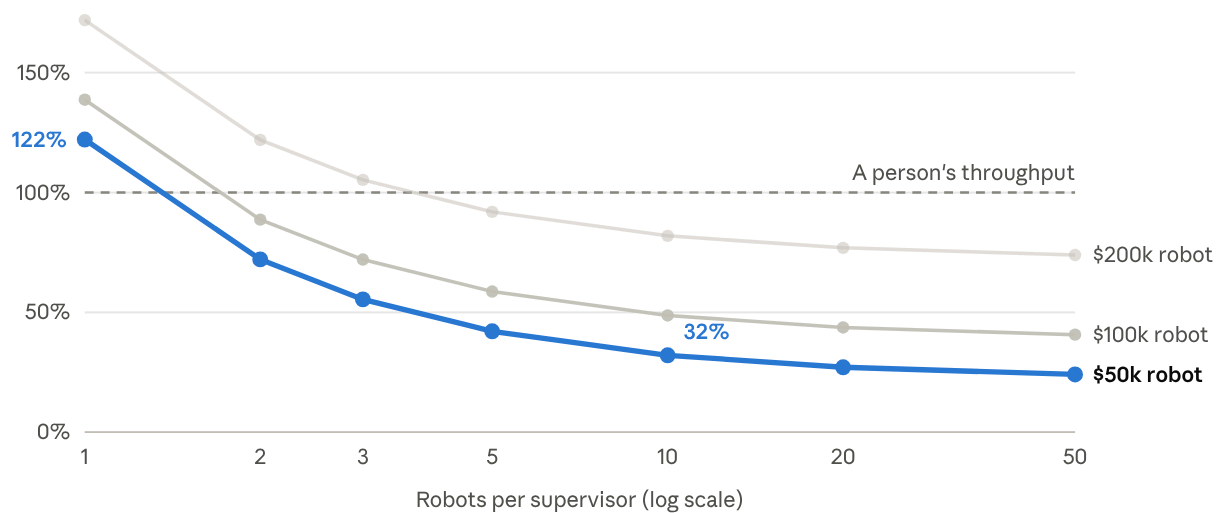
- **Supervision.** Supervisor cost W_s times interventions per robot-hour λ ($1/\text{MTBI}$, the mean time between interventions) times hours per intervention t_i . One person covers at most $1/(\lambda \cdot t_i)$ robots. Calls arrive at random, so plan for about 70% utilization, where a robot waits one to two and a half intervention-lengths for help, and count the wait as downtime.
- **Failures.** c_f is the rework or scrap cost of one failed attempt.
- **Margins.** g is your gross margin and b the savings a buyer needs before switching. At $g = 40\%$ and $b = 20\%$ the bracket must be covered 2.1 times. The 20% is an assumption; no primary source states what buyers require.

1. Supervision decides break-even before robot price does.

Robots per supervisor decide break-even more than robot price

A \$50k robot needs 122% of a person's throughput with 1 per supervisor, 32% with 10

Throughput needed to break even, as a share of a person's (before margins)



Computed here · robot cost per hour = price ÷ 12,000 h of life + 15% of price a year for service at 4,000 h a year + \$2/h for compute, energy and software (assumptions) · person and supervisor at \$36.38/h (BLS ECEC, transportation and material moving, June 2026) · break-even before margins

With both margins ($\times 2.1$), a \$50k robot at 10 robots per supervisor needs about 67% of a person's throughput, and a \$200k robot about 171%, at 16 hours a day; at 8 hours a day they need 78% and 214%. With one minder per robot, even a free robot would need 2.2 times a person's throughput, so $\log \lambda$ and t_i from the first week.

2. Hours per day matter as much as speed. A fixed price spreads over the hours worked. Agility's [RaaS terms](#), \$8,500 a month plus a ~\$25k deployment fee spread here over 36 months, give:

Robot hours a year	Cost per robot-hour	Fee-only break-even against a \$30.50/h person (before supervision and margins)
6,240 (20 h × 6 days)	\$17.68	58%
4,000 (16 h × 250 days)	\$27.58	90%
2,000 (one shift)	\$55.17	181%

The G1 runs about 2 hours per battery, so a 16-hour day needs about 8 battery changes. Log them as planned stops: they cut productive hours and add supervisor minutes, but they don't count against MTBI, which measures failures. Decide in phase 4 how the pilot robot gets power: a tether at a fixed station, charging contacts, a swap station, or a body that docks itself.

3. Interventions, not accuracy, set the success rate you need. If a fraction f of failures needs a person, an MTBI of M hours needs success $s \geq 1 - 1/(a \cdot M \cdot f)$. Faster work needs higher success unless the robot catches and retries its own failures:

Attempts per hour	MTBI ≥ 1 h, every failure needs a person	MTBI ≥ 1 h, 80% self-recovered	MTBI ≥ 8 h, every failure needs a person	MTBI ≥ 8 h, 80% self-recovered
20	95.0%	75.0%	99.38%	96.9%
43 (BMW's 84 s cycle)	97.7%	88.4%	99.71%	98.55%
60 (Siemens pilot, tote moves)	98.3%	91.7%	99.79%	98.96%
240	99.58%	97.9%	99.95%	99.74%
1,270 (Figure's parcel rate)	99.92%	99.61%	99.99%	99.95%

BMW's specification makes the same point. An 84 s cycle gives 429 cycles in a 10-hour shift, so more than 99% success per shift allows 4 failures, and zero interventions per shift means the robot recovers from each of them itself. Train and measure self-recovery (detect, retry, re-grasp, reset) as a skill in its own right.

The gates. The thresholds are judgments anchored to the published numbers below; trial counts come from Evaluation, and lower bounds are one-sided 95% exact bounds (96/100 gives 91.1%; 299/299 gives 99.0%).

Metric	Gate 1: Working (lab, own suite)	Gate 2: Potentially commercial (replica of a buyer's station)	Gate 3: Commercial (buyer's site, paid)
Success, blind, from held-out start states	$\geq 80\%$ over ≥ 50 trials (40/50 gives 67–89%)	$\geq 95\%$ with lower bound $\geq 90\%$ (96/100), or a sequential test	≥ 98 –99% on the buyer's acceptance test; 99% takes 299 trials with no failure, or 473 with one
Self-recovery	Logged	$\geq 80\%$ of failures recovered without a person	Enough that failures needing a person meet the MTBI target
MTBI	≥ 10 min	≥ 1 h (3 robot-hours without an unplanned intervention show it), so one person can cover about 8 robots at 5 min per intervention	≥ 1 shift (8–10 h; 24–30 clean robot-hours), and zero per shift if the buyer asks
Falls	Logged	None in the endurance runs	None in 24–30 robot-hours, with a Poisson bound
Throughput against a trained person	$\geq 40\%$	$\geq 2.1\times$ break-even for your costs (about 67% for a \$50k robot at 10 per supervisor and 16 h a day; 78% at 8 h)	The buyer's takt on paced lines; $\geq 2.1\times$ break-even on buffered work
Hours	1–2 h per battery	≥ 8 h a day, battery changes logged	16–20 h a day with automatic charging or swaps

Metric	Gate 1: Working (lab, own suite)	Gate 2: Potentially commercial (replica of a buyer's station)	Gate 3: Commercial (buyer's site, paid)
Cost per good unit	Not yet	Projected from measured η , λ and hours; below the buyer's	Measured on site, and projected at the fleet ratio that measured λ and t_i allow; below the buyer's after both margins
Safety	Gantry, two stops, fenced zone	ISO 12100 risk assessment; independent PL d Category 1 stop; guarded cell	NRTL field evaluation at the site; for EU sales, CE marking (under the Machinery Regulation from 20 Jan 2027)
Buyer	Interviews	The buyer writes the acceptance test and agrees to a paid pilot	Signed contract after the pilot
Unlocks	Commit to the method	A paid pilot, and fundraising on evidence	Revenue

Scaling (Gate 4). A business needs each site and each task to cost less than the one before:

- a second site with the same task reaches Gate 3 in two weeks or less, for about \$25k of deployment work or less (Agility's fee; Siemens's pilot ran two weeks);
- a new task in the same family reaches Gate 2 with fewer demonstration-hours than the task before it. Figure's logistics policy went from 88.2% to 94.4% correctly oriented packages, and from 6.84 s to 4.31 s per package, as demonstrations grew from 10 to 60 hours ([Figure](#));
- fleet MTBI and hardware mean time between failures are tracked separately, with Poisson bounds;
- gross margin is at least 40%; UBTech's company-wide gross margin reached 44.7% in H1 2026, lifted mainly by its full-size humanoids.

Published reference points.

Program	Published	Not published
Agility Digit, paid RaaS (deck)	98% accuracy on tote work; \$8,500 a month plus ~\$25k deployment; the model assumes a \$30.50/h person and 6,240 h a year; potential customer ROI 2.0× (RaaS) and 2.5× (purchase, 1.1-year payback); v4 bill of materials ~\$125k; ~31,000 h service life; 65,000+ operating hours across 9 sites; \$300M+ of Digit 5 orders from one customer	Uptime, interventions, throughput. 2025 revenue was \$1.8M against a \$140M operating loss (The Robot Report)
Figure 02 at BMW	Specification: 84 s cycle (37 s to load), more than 99% success per shift, zero interventions per shift, 5 mm placement in 2 s. Ran 11 months: 1,250+ h, 90,000+ parts	Rates achieved. The forearm was the top hardware failure
Humanoid's HMND 01 at Siemens (wheeled)	Two-week pilot, Jan 2026: 60 tote moves an hour, more than 90% success, over 30 min of continuous autonomy, over 8 h uptime	Interventions, cost
Figure F.03 livestream, May 2026	10 h against a person: 12,732 parcels against 12,924, or 98.5% of the person's throughput	Error rates and fleet size; later totals conflict between outlets
UBTech humanoids	H1 2026: 921 humanoids, RMB 590M (\$87.6M) of humanoid revenue, about \$95k per unit; company-wide gross margin 44.7%, lifted mainly by humanoids	Uptime, success

What does not count.

- Teleoperated or remotely assisted minutes counted as autonomous. Log every assist as an intervention.
- Demos and livestreams that report no error rates or interventions.
- Success from training start states, or on trials the operator picked.
- One good session. Drift (A8) shows up over days, so evidence spans several sessions with a canary.
- Letters of intent without a price and an acceptance test.

The number to optimize. Net value per robot-hour: $\eta \cdot W$ minus the bracket above, computed per task from the trial log on the frozen suite. Its four drivers form the dashboard: success with self-recovery, throughput ratio, interventions per hour times minutes per intervention, and productive hours per day. Ship a release when its targeted metric improves at 95% confidence, net value and interventions per hour are no more than 10% worse at 95% confidence (a non-inferiority test, Holm-corrected when several candidates compete), and no safety metric gets worse. Track one more number for the business: demonstration-hours and calendar days to take a new task to Gate 2. A generalist model should make it fall with every task; if it stays flat, the company is selling integration services.

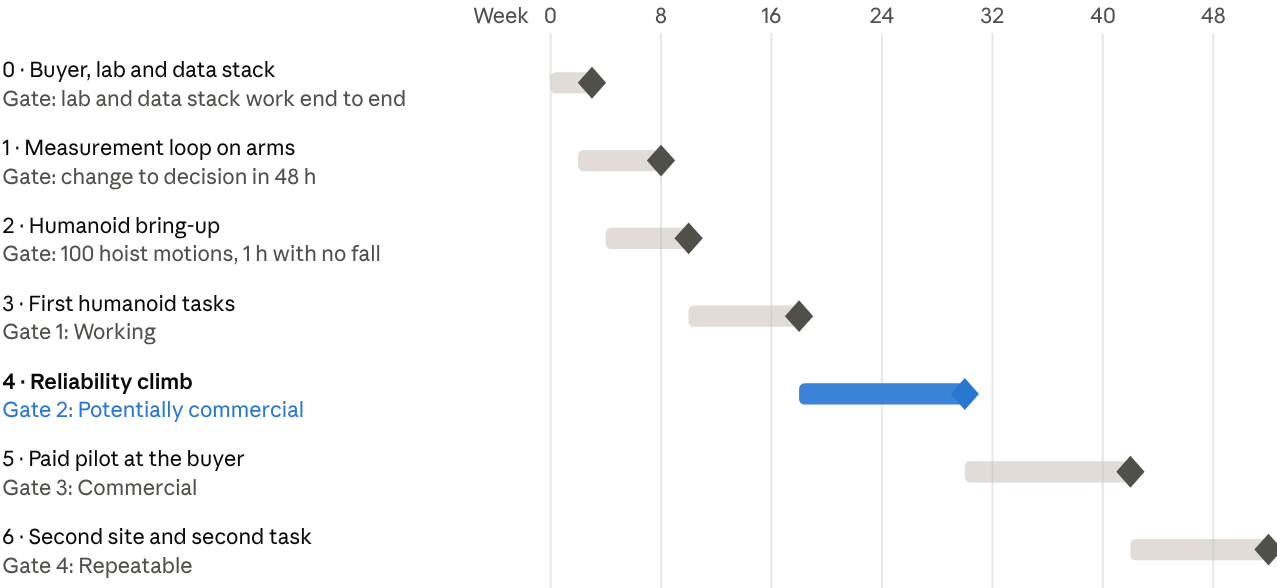
Where the inequality is easiest. Buffered rather than paced work, so η below 1 still pays; two or three shifts a day; failures that are cheap and can be retried; and a buyer who already has a written acceptance test. Tote moving, destacking and parts sequencing fit. Paced assembly stations, which need takt and zero interventions, come later.

End-to-end runbook

Run seven phases, each ending in a gate. Phases on different hardware overlap (the humanoid's bring-up runs while the arm loop is being validated), but a phase's main work starts only when the gate before it passes, and no humanoid policy experiment runs before the loop's gate. Spend the first eight weeks making the measurement loop work on cheap arms, because every later decision rests on its numbers (A4, A10). The humanoid's first useful work is in a replica of one buyer's station, and the year ends with a paid pilot and a second site.

Seven phases and their gates over the first year

Gate 2: Potentially commercial is targeted for week 30; the last gate for week 52
Gates decide when to move on; the weeks are targets.



Plan in this document · the weeks are targets; gate thresholds are in Commercial thresholds and the sections each phase names

The weekly loop (from week 2). Every release climbs the evaluation ladder, cheapest rung first:

When	Step	Output
Monday	Freeze 2–5 candidates, each with a policy hash, data manifest and config	Candidates
Monday–Tuesday	Offline action error on held-out episodes; the twin and world-model ranking once validated	1–2 finalists
Wednesday–Thursday	Blind paired bundles with the canary; a sequential test stops early when it can	Winner, with a confidence interval
Thursday–Friday	Reliability run of the winner; every failure and intervention tagged by class	Failure counts by class
Friday	Ship, iterate or roll back; plan next week’s data from the top failure classes	Decision log and data plan

Small changes (a bug fix, a data filter) go through the same ladder in under 48 hours, but two days of real trials detect only large effects (What a week can detect, in Evaluation), so validated proxies decide the rest.

Daily pre-flight, about 10 minutes before the first trial:

1. Test both stops with the robot on the hoist.
2. Check calibration: fiducial reprojection error, timestamp error under 1 ms, a latency probe.
3. Log battery, motor temperatures, and firmware and controller versions.
4. Run 5 canary trials as a smoke test, and end the session if most fail; drift is judged on the week's pooled canary trials.

Tests at each layer.

Layer	What it catches	When it runs
Unit and integration tests	Action horizon, normalization statistics or action type differing between training and serving; schema and timestamp errors	Every commit
Replay test	The policy run on logged episodes, its actions compared with the recorded ones	Every candidate
Sim-to-sim	A controller or deployment binary that works in only one engine	Every controller change
Pre-flight	Calibration drift, latency, broken stops	Every session
Blind paired bundles	Whether B beats A	Every release
Reliability and endurance runs	MTBI, battery, heat, wear	Weekly from phase 4
The buyer's acceptance test	The buyer's KPI	The pilot

Phase 0 · Weeks 0–3 · Buyer, lab and data stack.

1. Interview operations managers at 15–20 warehouses and factories. For 2–3 candidate tasks, get the acceptance test in writing, a trained person's units per hour, shifts per day, the cost of a failure, and photos and dimensions of the station. Rank the tasks with

the inequality in Commercial thresholds.

2. Order the YAM pair with cameras, a PICO 4 Ultra with two trackers, a workstation (an RTX 5090 at ~\$11k plus rented H100s, or a 96 GB RTX PRO 6000 on quote), a hoist, fencing, wireless stops and a radio power-cut stop. Order the body once the tasks are ranked, by week 2: the G1 EDU with 29 joints (Dex3-1 hands or grippers, per Hands and touch) if the top task needs walking between stations, steps or a deep squat; otherwise consider the wheeled G1-D, which has the same arms, stays standing when power is cut and runs 2–6 hours per battery, but has no SONIC whole-body teleop and must be fine-tuned as a new embodiment. Add 10% of the robot's price for spares.
3. Set up the lab: a fenced 3 m zone, the hoist, two stops, an isolated network with Bluetooth off, and a PTP clock.
4. Build the data stack: MCAP recording, a LeRobot v3 build job, the ClickHouse schema, ingest checks, and a Grafana dashboard that computes Wilson and Poisson intervals in SQL.
5. Write the protocol: a task specification template, a trial runner that randomizes bundles and hides the policy from the scorer, and the intervention classes.

Gate: a test episode flows from recorder to dashboard with every check passing, both stops work, and one buyer's acceptance test is on paper.

Phase 1 • Weeks 2–8 • Measurement loop on arms.

1. On the YAM station, set up the two control tasks from the suite and an arm version of the buyer's task.
2. Collect 100–200 demonstrations per task; at least 90% must pass the quality checks.
3. Fine-tune GR00T N1.7 (YAM tag) and $\pi 0.5$ (LoRA). Match the action horizon and normalization between training and serving, and use state dropout.
4. Score every checkpoint by offline action error on held-out episodes.
5. Run blind bundles of 50 trials per task and policy, with the canary, and compute intervals and sequential tests from the trial table.
6. Validate the proxy: compare offline action error (and ABC's MuJoCo twin of the YAM, if used) with real success as checkpoints accumulate, until at least 15 meet the agreement rule in Simulation.
7. Time the loop from a change to a decision.

Gate: a change reaches a decision within 48 hours; the proxy meets the agreement rule, or is used only to drop clear losers; the canary has been stable for two weeks; control tasks are at 80% or better.

If it fails: fix the loop before any humanoid policy work. The usual causes are a training-serving mismatch, calibration drift and inconsistent resets.

Phase 2 · Weeks 4–10 · Humanoid bring-up.

1. In simulation: the G1 in Isaac Lab and SONIC on the virtual gantry, then the same deployment binary in MuJoCo, with JetPack 6 and TensorRT 10.7 pinned. On a G1-D, skip the balance steps.
2. On the hoist, in the order given in Whole-body control. Test the stops, and match the pose before every mode switch.
3. Measure the latency chain (camera, actuators, network) and timestamp error, and store both per software version.
4. Teleop: PICO 4 Ultra and trackers into SONIC. Train the operators and log every representation at 50 Hz. Build the robot-free capture rig and its retargeting job too (GMR into the robot's references, plus gloves if the hands have fingers).
5. Hands: calibrate the Dex3-1 and log touch at full rate.
6. Endurance: battery runtime under load, motor temperatures, and a timed battery change.

Gate: 100 held-out motions on the hoist with at most one tracking failure and no fall, then an hour of test motions without a fall; the latency watchdog test passes; at least 90% of teleop episodes accepted; demonstrations per operator-hour measured; network latency under 10 ms (under 30 ms is acceptable).

Phase 3 · Weeks 10–18 · First humanoid tasks, to Gate 1.

1. Build a replica of the buyer's station: the same totes, shelf or conveyor heights, and lighting range, plus fixtures and a scripted routine that reset the scene without a person. Photograph at least 20 start states per task, and split environments into training and held-out.
2. Freeze suite v1 (6–8 tasks, as in Benchmarks).
3. Collect about 200 teleop demonstrations per buyer task through SONIC, and add robot-free capture for 2–3× the throughput. Train human data and open G1 data recorded through other controllers as separate embodiments, and keep each only if a blind ablation shows it helps.
4. Fine-tune GR00T N1.7 on the UNITREE_G1_SONIC tag (2–3 hours on one 48 GB GPU), and train $\pi_{0.5}$ into SONIC with the OpenHLM recipe as the comparison. The two use different interfaces (tokens against joint references), so this picks a stack; to compare models alone, train both on the same logged references.

5. Make a Polaris twin of the replica cell, and let it screen candidates only once it meets the agreement rule in Simulation.
6. Evaluate in blind bundles of 50 trials per task and candidate, from held-out start states, with the canary. Time a trained person doing 30+ units at the same station to get η .
7. Start the failure taxonomy: perception, grasp, placement, balance, recovery, hardware.

Gate 1: at least 80% over 50+ trials on two buyer tasks, MTBI of 10 minutes or more, and η of at least 40%.

If it fails by week 24: swap the hand for a gripper if grasping dominates the failures; shorten the task; audit data quality and training-serving consistency; try whole-body joint references (OpenHLM) if the SONIC token interface is the limit.

Phase 4 · Weeks 18–30 · Reliability climb, to Gate 2.

1. Run correction rounds (HG-Dagger): the operator takes over at each failure from the robot's current pose (hand motion relative to where the hands are, with a clutch, blended over about half a second; rehearse on the hoist), and the policy retrains weekly on 300+ new rollouts.
2. Teach self-recovery: collect demonstrations of noticing and fixing the common failures (missed grasp, dropped tote, misplaced item), and flag failures with a progress model (Robometer) once its flags match human labels on your tasks (check precision and recall).
3. Run real-robot RL on the bottleneck skills through RLinf: RL Token or HIL-SERL at a few hours per skill, DSRL on the frozen policy, and Recap-style advantage conditioning across rollouts and corrections. Freeze SONIC during RL; clamp workspace, speed, force and motor temperature; and count resets as robot time.
4. Run 4–8-hour endurance sessions, logging battery changes as planned stops. Record MTBI and falls with Poisson bounds, and hardware failures separately. Decide how the pilot robot gets power: a tether, charging contacts, a swap station or a self-docking body.
5. Push speed only once success is above 90%: cut pauses, tune chunk length and execution speed, and re-measure η .
6. Decide releases with sequential tests, and confirm Gate 2 with a fixed 100-trial run per task.
7. Prepare the site: an ISO 12100 risk assessment of the pilot cell, an independent PL d Category 1 stop built into the cell (interlocks command the crouch; a safety relay cuts power after a fixed delay), and the guarded-cell layout.

8. Sell: show the buyer the replica's numbers, and agree the paid pilot and its acceptance test.

Gate 2, the potentially commercial column of the gate table: at least 95% success with a 90% lower bound on the buyer's tasks, 80% of failures self-recovered, MTBI of at least an hour, no falls in the endurance runs, η at least $2.1\times$ break-even, 8 hours a day, and a signed paid pilot.

If it fails by week 36: if one failure class dominates, attack only that; if η is too low, choose another task, or a wheeled body (G1-D) for work that doesn't need legs; if hardware limits MTBI, fix the hardware first.

Phase 5 · Weeks 30–42 · Paid pilot, to Gate 3.

1. Survey the site, install the guarded cell, update the risk assessment, book the NRTL field evaluation, and train the buyer's staff on the stops.
2. In the first two weeks on site, collect 100–200 demonstrations per task in the real cell, build a twin of it, and release daily through the ladder.
3. Run two shifts a day, each with two people: one of yours plus a trained member of the buyer's staff, or two of yours. Log every intervention with its class, review failures daily and release weekly.
4. Run the acceptance test exactly as the buyer defined it, for example three consecutive shifts at 99% with no interventions.
5. Measure cost per good unit against the buyer's.

Gate 3: acceptance test passed, 98–99% success, MTBI of a full shift (24–30 clean robot-hours) with no falls, 16–20 hours a day, NRTL evaluation passed, contract signed.

Phase 6 · Weeks 42–52 · Second site and second task, to Gate 4.

1. Deploy the same task at a second site, recording days and dollars to Gate 3.
2. Add a new task from the same family, recording demonstration-hours and days to Gate 2.
3. Run at least two robots, and measure robots per supervisor and remote-assist time.

Gate 4: the second site reaches Gate 3 within two weeks for about \$25k or less, demonstration-hours per new task are falling, and gross margin is at least 40%.

Robot time (A5; estimates).

Phase	Robot-hours	People present
1 Measurement loop	~40–80 on the arms (about 1,000 trials and 300–600 demonstrations)	Founder alone; the operator joins around week 4
2 Bring-up	~40	Operator and spotter
3 First tasks	~100–130; one evaluation round of 5 tasks × 3 policies × 50 trials at 3 minutes is ~38 h	Operator and spotter
4 Reliability climb	~150–200, about 2–3 hours a day on one robot; a second G1 halves the calendar time	2–3
5 Pilot	~640 (16 h × 5 days × 8 weeks)	2 per shift (one of yours plus trained buyer staff, or four of yours across two shifts), plus remote monitoring

Release checklist before anything runs at a buyer's site:

- the replay test passes: same action horizon, normalization and action type as in training;
- latency stays under 150 ms at p99 on the deployment hardware, and the watchdog test passes;
- today's stop tests passed;
- the release won its blind comparison, with the canary inside its band;
- interventions per hour did not rise on the last reliability run;
- the previous release is one command away.

Budget, team and what a solo founder can run

Year one costs about \$0.55M before the founder's own pay and the pilot site's costs: about \$143k of hardware for one robot, \$68k for a second robot in phase 4, \$12–24k of compute and storage, and about \$320k for three hires, which makes people the largest line. One person can run phases 0 and 1 alone; from the first humanoid session on, every robot session needs two people.

Hardware. Prices are from Robot body, Hands and touch, and Sensors; items marked estimate are not quoted prices.

Item	Lean set	Recommended set
Humanoid	G1 EDU Plus, 29 joints: \$53,900	G1 EDU Ultimate B with tactile Dex3-1 hands: \$67,900
End effectors	2 Dex1-1 grippers: \$2,160	Dex3-1 included; Wuji Hand 2 on an xArm 7 for dexterity research: \$25,600
Arm station	YAM leader-follower pair: \$12,347, plus 3 cameras	Same
Compute	RTX 5090 workstation: ~\$10,996; rent H100s for fine-tuning, since 32 GB is short of N1.7's 40 GB	Same, or a 96 GB RTX PRO 6000 workstation (get a quote); Jetson AGX Thor dev kit for on-robot latency: \$5,499
Touch	2 AnySkin and a GelSight Mini: \$766	Plus a PaXini kit: ~\$2,100 in all
Teleop and capture	PICO 4 Ultra and 2 trackers: ~\$1,000	Plus a HuMI-style capture rig: ~\$3,000 (estimate)
Lab safety	Hoist or gantry, fencing, wireless and power-cut stops: \$5,000–10,000 (estimate)	Same
Spares	10% of the robot: \$5,390	\$6,790
Total	~\$94,000 plus cameras	~\$143,000 plus cameras

Running costs, year one (estimates).

Item	Cost	Basis
Cloud compute	\$10,000–20,000	GR00T fine-tunes cost under \$10 each, π 0.5 runs \$170–335, and a world-model ranking ~100 H100-hours (\$350 at \$3.49/h)
Storage	\$2,000–4,000	10–20 TB at \$139–300 a month

Item	Cost	Basis
Second G1 EDU (phase 4)	\$53,900–67,900	Halves phase 4's calendar time and gives the first fleet numbers
Pilot site	Get quotes in phase 4	Guarded cell, charging or swap station, NRTL field evaluation, travel, insurance

Team. Loaded costs use BLS's private-industry ratio of total compensation to wages (1.43).

Role	Starts	Why	US cost
Founder	Week 0	Buyer discovery, the data and evaluation stack, training	—
Robot operator and technician	Week ~4	Teleop, resets, pre-flight, spares; the second person every robot session needs	Median wage \$73,900 (BLS , electro-mechanical and mechatronics technicians, May 2025); ~\$106k loaded
Robotics ML engineer	Week ~12	Post-training and RL, the controller, simulation	Median \$140,300 (BLS , computer and information research scientists); ~\$200k loaded; robotics specialists cost more
Second operator	Phase 4	Two-shift endurance runs, then the pilot	As the first operator
Safety engineer, part-time	Phase 4	ISO 12100 risk assessment, PL d stop design, NRTL preparation	Contract

What one person can run.

- Alone: phases 0 and 1, simulation and twin work, every fine-tune and analysis, the data stack, and buyer discovery.
- Not alone: any humanoid session (operator and spotter are different people), endurance runs longer than a few hours, and the pilot.

- Teleop is physical work. Jim Fan puts it at about 3 hours per robot per day (reported in a talk summary), so each operator caps the data rate (A5); robot-free capture rigs and correction rounds stretch it.
- Build the measurement loop yourself, because it compounds and it is where a data-infrastructure founder is fastest (Data infrastructure, Evaluation). Buy or rent everything else.

Cheaper paths.

- Start with the lean set and add the research hand only when a buyer's task needs fingers.
- If the buyer's task needs no walking, the wheeled G1-D (\$38,800–66,300) carries the same arms on a lift column, with fewer falls and simpler safety.
- Defer the second G1 until phase 4 shows that robot time, not data or method, is the bottleneck.

Bets and hedges

Invest only in what every leading lab has converged on and what follows from the axioms: a learned whole-body controller, pretrained generalist policies that output chunks of state-relative actions, human data joined to a small robot bridge, RL on your own rollouts, and evaluation built before scale. Hedge the parts still in contest (world-action models, the hand, legs or wheels) by keeping data and interfaces neutral. Everything you build should survive swapping the body, the hand, the model and the controller.

Safe bets (from the research for this document; each with the evidence that would overturn it).

Bet	Evidence	What would change it
A learned whole-body controller at the bottom, with teleop routed through it	Figure's S0 replaced 109,504 lines of C++ ; Boston Dynamics trains Atlas with RL; SONIC	A single policy running on the robot at 500 Hz or faster beats a tracker in blind trials, or certification demands non-learned balance control

Bet	Evidence	What would change it
Flow or diffusion heads that output chunks of state-relative end-effector and hand actions	GR00T N1.6 and N1.7 moved to state-relative action chunks ; $\pi 0$ through $\pi 0.7$; Atlas outputs poses	World-action models return chunks fast enough for closed-loop control (under ~150 ms on Thor) and win contact-rich tasks
A data portfolio: broad human and egocentric data, a small aligned robot bridge, your own teleop, all in LeRobot and MCAP	EgoScale: 20,854 hours of human video made useful by about 4 hours of aligned robot data; Generalist reports ~1 hour of robot data per task on a 500,000-hour model (GEN-1 , company claim)	Independent replications show human-data gains flattening beyond about 20,000 hours
RL post-training on your own rollouts and interventions	Recap more than doubled throughput and roughly halved failures; $\pi 0.7$ matches RL-tuned specialists (company claim)	A generalist reaches 99% zero-shot on unseen tasks
Evaluation before scale: blind real trials plus validated proxies	Saturated benchmarks collapse under perturbation (LIBERO-Plus : from 95% to below 30%)	A neutral commercial evaluator appears; then buy it
A commodity body and rented compute, budgeted for rising rents	Unitree's average humanoid price fell from about \$85k to \$25k between 2023 and 2025 (Rest of World); H100 rents rose from \$2.57 to \$2.75 an hour between May and July 2026 (Silicon Data)	The neocloud median H100 rent (Silicon Data) falls back below about \$2 an hour
ROS 2 long-term releases with MCAP	ROS 1 reached end of life in May 2025; Lyrical Luth is supported to May 2031	None expected

Hedges (stay compatible, don't commit yet).

Contest	Today	Commit when	Drop when
World-action models (GR00T N2, Cosmos 3, DreamZero) as the policy	Cosmos 3 Edge takes 1.53 s per action chunk on Thor; GR00T N2's weights are unreleased	Open weights return a chunk in under ~150 ms on Thor and win blind contact-rich trials	Still over ~1 s per chunk and losing on precision by mid-2027
What the policy is built from: a vision-language model, human video from scratch, or one stream	GR00T N1.7 and $\pi 0.5$ start from vision-language models; Figure's Helix 2.5 trains from scratch	The from-scratch recipe is replicated in the open	—
High-DoF hands with touch	Hands are cheap (Linkerbot's O6 Lite at ¥3,999, about \$560, after subsidy; 36Kr), but touch data does not pool across hands, and Atlas's new hand has four digits	A standard tactile data format, or blind dexterity wins from glove data	Grippers keep winning the buyer's tasks
Legs or wheels	The Siemens pilot, Sunday's Memo and PI's platforms are wheeled or fixed; Rodney Brooks expects wheels	Commit to legs when legged robots show measured task-time gains	Move to wheels (G1-D) when your buyers' tasks need no walking
An open model or a licensed one	Open weights (GR00T N1.7, $\pi 0.5$) are enough to start; GEN-1 and $\pi 0.6/\pi 0.7$ are closed, and a model vendor can become a competitor	A licensed model wins your blind trials by enough to justify the dependence	— (either way, keep two model families running and own the adapters)

Stop investing in these.

- ROS 1, ad hoc data folders, and new RLDS or HDF5 pipelines.
- LIBERO and SimplerEnv as decision metrics: LIBERO-PRO saw scores over 90% fall to 0% when objects moved.
- Single-task policies trained from scratch: Generalist reports 99% where training from scratch reaches 19% (company claim).
- Absolute joint-angle actions tied to one body: human data reaches them only by retargeting to that body, and they don't carry over to the next one.
- Hand-written rewards for manipulation RL; learned progress and value models replace them.
- MPC-only humanoid control, now replaced by learned controllers at Figure, Boston Dynamics and Agility.
- Teleop rigs that record through a different controller than the one deployed. EgoScale's teleop was 4 hours, under 0.1% of its training mix.
- Hand-built simulation scenes as the main data source; generated worlds are taking over, and AMD is [buying World Labs for \\$8.2B](#).
- 7B autoregressive policies (OpenVLA-class), older checkpoints (π_0 , π_0 -FAST, GR00T N1.5 and N1.6), and Orin-class onboard compute.

Signals to check each quarter.

Signal	Today	Threshold	Action
World-action latency on Thor	1.53 s per 32-action chunk	Under ~150 ms, plus a blind contact-rich win	Move the policy layer to a world-action model
GR00T N2	Due by the end of 2026, unreleased	Open weights with N1.7's license terms	Swap it in and rerun the frozen suite
Zero-shot reliability of open generalists	Far below 99%	99% on unseen tasks	Shift effort from RL post-training to deployment and evaluation
Human-data scaling	Log-linear gains to ~20,000 hours (EgoScale)	Flattening in replications	Move budget back to robot data

Signal	Today	Threshold	Action
ISO 25785-1	Committee draft (May 2026)	Published	Redo the risk assessment and plan work without fencing
Onboard compute	G1 runs Orin NX; Thor dev kit \$5,499	EDU bodies with Thor at G1 prices	Run inference on the robot
H100 rent	~\$2.75 an hour (neocloud, Jul 2026)	Under \$2 (neocloud median)	Rent more, buy less

Where informed people disagree. Sergey Levine argues that any hand-designed component eventually becomes the bottleneck ([Sporks of AGI](#)). Jim Fan expects teleop data to become “negligible within a year or two” (talk summary). Rodney Brooks predicts that deployable dexterity will still be poor beyond 2036 ([scorecard](#)), and The Economist doubts a commercial humanoid market emerges this decade. The loop in this document holds under all four: it learns everything above the safety chain, works with human or robot data, and measures commercial value instead of assuming it.

Sources

Every figure links to its source where it appears; this list groups the primary sources. Research ran to 2 October 2026, and prices were checked on 1–2 October 2026. Company claims are marked where they are used, and numbers marked estimate or unverified were not confirmed.

Learning, statistics and evaluation

- Scaling and compounding error: [Lin et al. 2024, data scaling laws for imitation learning](#) · [Ross et al. 2011, DAgger](#)
- Real-robot evaluation: [TRI large-behavior-model study](#) · [STEP sequential test \(RSS 2025\)](#) · [N-SCORE \(RSS 2026\)](#) · [RoboArena](#) · [Kress-Gazit et al. 2024, reporting robot-learning evaluations](#)
- Benchmarks: [LIBERO-PRO](#) · [LIBERO-Plus](#) · [RoboCasa365 leaderboard](#) · [RoboLab leaderboard](#) · [BEHAVIOR-1K Challenge](#) · [Isaac Lab-Arena](#) · [AgIBot World Challenge 2026](#)

Models and post-training

- Policies: [GR00T N1.7](#) · [GR00T FAQ](#) · [GR00T N1.6](#) · [openpi \(\$\pi\$ 0.5\)](#) · [\$\pi\$ 0.5 on Thor](#) · [LingBot-VLA 2.0](#) · [Being-H0.5](#) · [X-VLA](#) · [SmolVLA](#) · [GEN-1](#)

- World and world-action models: [Cosmos 3](#) · [Cosmos 3 Edge post-training](#) · [Cosmos 3 Edge DROID policy](#) · [DreamZero](#) · [FLUX 3 Action](#)
- Post-training: [HG-Dagger](#) · [RL Token](#) · [HIL-SERL](#) · [DSRL](#) · [\$\pi\$ RL](#) · [SimpleVLA-RL](#) · [Robometer](#) · [RLinf](#)
- Fine-tuning recipes: [Isaac Lab-Arena G1 example](#) · [OpenHLM paper](#) and [project](#)

Whole-body control and teleoperation

- Controllers: [SONIC code, docs](#) and [weights](#) · [BeyondMimic](#) · [AGILE](#) · [HANDOFF](#) · [HumanUP](#) · [Figure Helix 02](#) · [Boston Dynamics on training Atlas](#)
- Training your own: [mjlab](#) · [unitree_rl_mjlab](#) · [holosoma](#) · [ProtoMotions](#)
- Teleop and capture: [SONIC VR teleop](#) · [TWIST2 paper](#) and [code](#) · [HuMI](#) · [xr_teleoperate](#) · [Isaac Teleop](#) · [GMR retargeting](#) · [DOGlove](#) · [SenseGlove Nova 2](#) · [Xsens Link](#) · [AMASS license](#)

Simulation and twins

- Engines: [Isaac Lab releases](#) and [benchmarks](#) · [Isaac Sim requirements](#) · [Newton](#) · [MuJoCo changelog](#) · [MuJoCo Playground](#) · [Drake](#) · [ManiSkill3](#) · [Genesis](#) and an [independent test](#)
- Twins, generated data and world-model evaluators: [Polaris](#) · [RialTo](#) · [HumanoidMimicGen](#) · [RoboWorld](#) · [GigaWorld-1](#) · [Runway GWM-Robotics](#)

Data infrastructure and datasets

- Formats and tools: [MCAP spec](#) · [LeRobot dataset v3](#) · [RLDS \(archived\)](#) · [HFlow](#) · [Cosmos-Reason2-8B](#) · [Rerun](#) · [Foxglove](#)
- Datasets: [Unitree's G1 datasets](#) · [Humanoid Everyday](#) · [RoboMIND](#) · [GR00T-Teleop-G1](#)
- Storage prices: [S3](#) · [R2](#) · [B2](#) · [Hugging Face](#)

Hardware, sensors and timing

- Bodies and prices: [unitree_sdk2 G1 low-level example](#) · [G1 configurations guide](#) · [RoboStore G1 EDU](#) · [Stemfinity G1 EDU Ultimate](#) · [H2 EDU](#) · [H2 Plus](#) · [G1-D](#) · [I2RT YAM](#) · [Trossen Stationary AI](#) · [Galaxea R1 Pro](#) · [NVIDIA's reference humanoid](#)
- Hands and touch: [Dex3-1](#) · [Dex5-1](#) · [Wuji Hand 2](#) · [LEAP v2 Advanced](#) · [ORCA durability paper](#) · [Sharpa Wave](#) · [AnySkin](#) · [GelSight Mini](#) · [PaXini](#) · [Digit 360](#)
- Compute, clocks and latency: [Jetson price increase](#) · [Thor PTP thread](#) · [Holoscan](#) · [Sensor Bridge latency](#) · [UMI latency method](#) · [CARET](#) · [GPU price index](#) · [RunPod pricing](#) · [Silicon Data GPU rents](#)

- Reliability and security: [hardware wear \(Deep News\)](#) · [G1 arm failures \(forum\)](#) · [Alias Robotics findings](#)

Safety and standards

- [ISO/CD 25785-1](#) · [ISO 10218:2025 FAQ](#) · [IEEE Spectrum on Digit's NRTL evaluation](#) · [IEEE humanoid study_group](#) · [EU Machinery Regulation](#) · [EU AI Act timeline](#) · [Robot-Trains-Robot](#)

Economics, deployments and trends

- Labor costs: [BLS ECEC, June 2026](#) · [BLS electro-mechanical technicians](#) · [BLS computer and information research scientists](#)
- Deployments: [Agility investor deck \(SEC\)](#) · [Agility's 2025 financials](#) · [Figure at BMW](#) · [Figure logistics scaling](#) · [Humanoid's pilot at Siemens](#) · [UBTech H1 2026](#)
- Curves and views: [Unitree's prices \(Rest of World\)](#) · [Unitree R1 price](#) · [hand prices \(ITES\)](#) · [hand market \(36Kr\)](#) · [AMD and World Labs](#) · [Levine, Sporks of AGI](#) · [Brooks's 2026 scorecard](#)