

DEEP RESEARCH REPORT · OCTOBER 2026

Vision-language-action models graduate from demos to shifts

Architectures, action heads, data, RL post-training, world models, evaluation, deployment and industry for robot foundation models

>2×

Throughput on PI's hardest tasks after RECAP RL ($\pi^*0.6$)

91% → 99%

OpenVLA-OFT on LIBERO after SimpleVLA-RL

32% → 62%

RT-2 vs RT-1 on unseen scenarios from web co-training

~84 → 69.6

OpenVLA-OFT, clean LIBERO-Spatial vs perturbed LIBERO-Plus

10,306 words ~45 min read 163 linked sources 12 figures & tables

Prepared with Claude Code deep research · [sugeerth/job-search](#)

Contents

01	Four generations in four years: tokens, experts, hierarchies, steerable generalists	4
02	A 3–5B VLM plus a flow-matching expert is now the default stack	6
	Action heads: why flow matching won, and where tokens still help	8
	Cross-embodiment designs absorb heterogeneity with padding, prompts and mixtures	9
03	Fast and slow systems split into three tiers	10
04	Reasoning became conditioning: from chain-of-thought to steerable prompts	12
05	Data corpora jumped from 10k to a claimed 270k hours, and diversity still beats volume	13
	The data pyramid formalizes three tiers of supervision	14
	Web co-training protects generalization, not in-distribution skill	15
	Scaling laws: diversity in academia, hours in industry	15
06	RL post-training doubled real-world throughput, mostly with humans in the loop	16
	RECAP: advantage-conditioned learning from experience and corrections	16
	Simulation RL saturates LIBERO and rescues one-demo policies	17
	Real-world RL means RL plus interventions	18
07	World models, reward models and verifiers are the new compute sinks	19
08	LIBERO is saturated, and robustness suites expose the gap	21
09	Deployment hides latency in the execution layer	23
10	Safety and security lag capability by a wide margin	24
11	Capital prices the robot brain separately from the robot body	25
12	Seven open problems define the 2027 research agenda	27
13	What our own lab reproduces at small scale	28
14	Conclusion	30
15	Sources	31

EXECUTIVE SUMMARY

By October 2026 the frontier vision-language-action (VLA) model is a 3–5B-parameter vision-language backbone with a separate flow-matching "action expert". It emits chunks of continuous actions at 50–200 Hz, sits inside a hierarchy of slower planners and faster controllers, and is improved after pretraining by **reinforcement learning from its own deployments plus human corrections**. Physical Intelligence's $\pi 0.7$ (April 2026) is the clearest example: one 5B generalist that reportedly matches $\pi^*0.6$'s RL-tuned specialists on laundry, espresso and box building, and that is steered by prompts carrying subtasks, world-model-generated subgoal images and data-quality metadata ([arXiv 2604.15483](#)). Google DeepMind's Gemini Robotics 2 (July 2026) extends the same pattern to whole-body humanoid control under one policy, and NVIDIA's GR00T N1.7 and Hugging Face's SmolVLA give practitioners open weights for the same design. The competitive edge has moved from architecture to **data and post-training**. Proprietary corpora grew from about 10k hours ($\pi 0$, 2024) to more than 100k hours (Xiaomi-Robotics-1, 2026) and a claimed 270k hours (Generalist GEN-0), mostly collected with wearable or handheld devices rather than robot teleoperation. RL recipes such as RECAP more than doubled throughput on PI's hardest tasks. Evaluation is the weak link. Clean LIBERO is saturated at 94–99%, perturbation suites expose drops of 15+ points, the best BEHAVIOR-1K score is about 0.26, and almost every real-world deployment number is vendor-reported. Capital follows the "robot brain" story: Figure at USD 39B, Skild at USD 14B+, Physical Intelligence at USD 5.6B confirmed and about USD 11B reported. Paid deployments are still pilots, and the first consumer humanoid depends on remote teleoperation. For a practitioner building a VLA today, the decisive choices are a pretrained VLM backbone protected by knowledge insulation, a flow-matching chunked action head with asynchronous or real-time-chunked execution, diversity-first data collection with a human-video base, and an intervention-plus-advantage post-training loop. Our own CPU-scale lab reproduces several of these lessons in miniature, and the report closes with what those toy results do and do not tell us.

HOW TO READ THIS REPORT

How to read this report. The research tools could not open arXiv, Hugging Face or pi.website directly in this session. Most 2026 numbers come from search-result excerpts of primary pages, and some well-known 2023–2025 figures come from primary papers the researchers knew but could not re-open. Numbers from a primary paper or official page are untagged. Everything else carries an inline tag: **SEC** secondary source (press, blog, third-party summary or reimplementation); **INF** inference by the research team or the author; **UNV** unverified (single source, title only, or recalled and not re-checked); **VENDOR** company self-report with no independent replication; **CONFLICT** sources conflict, with both values given. "Toy-scale" marks results from our own lab, which are primary for us but say nothing directly about frontier systems.

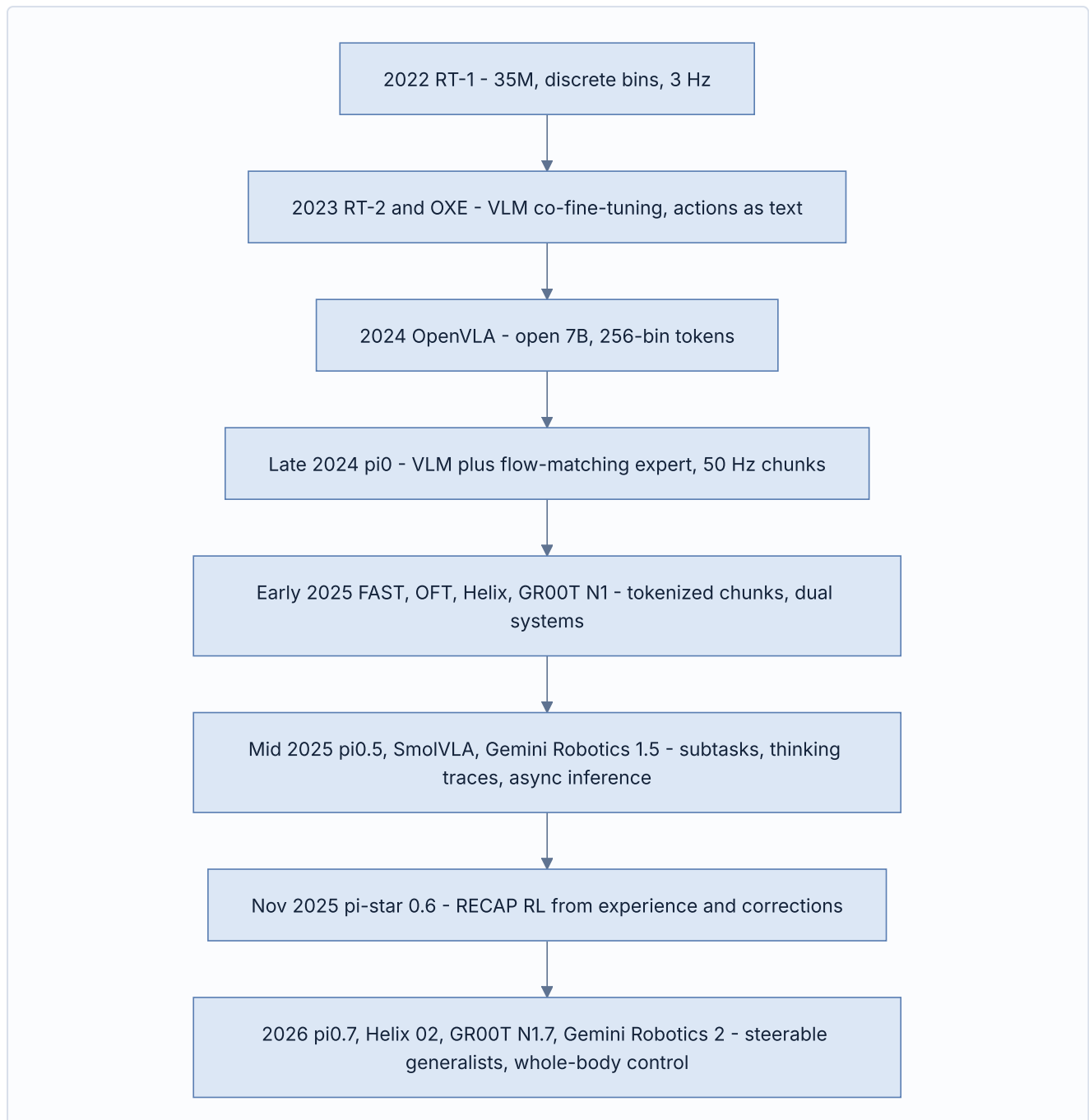
01

Four generations in four years: tokens, experts, hierarchies, steerable generalists

The VLA story is a four-step lineage, and each step answered a specific failure of the one before. The **discrete-token era (2022–2024)** treated actions as language. RT-1 was a 35M-parameter robotics transformer: an EfficientNet-B3 encoder with FiLM language conditioning, TokenLearner, and 256 bins for each of 11 action dimensions, running at 3 Hz and trained on about 130k episodes covering 700+ tasks ([RT-1](#), [arXiv 2212.06817](#)). RT-2 then co-fine-tuned PaLI-X (5B and 55B) and PaLM-E (12B) to emit actions as text tokens. It roughly doubled success on unseen scenarios, **from RT-1's 32% to 62%**, and showed emergent semantic reasoning, but the 55B model ran on cloud TPUs at only about 1–3 Hz ([RT-2](#), [arXiv 2307.15818](#); [DeepMind blog](#)). Open X-Embodiment pooled more than 1M trajectories from 22 embodiments and 34 labs, and RT-2-X trained on it was about 3x as successful as RT-2 on emergent skills ([OXE paper](#); [DeepMind blog](#)). OpenVLA made the recipe open in 2024: a 7B Prismatic model (Llama 2 with fused DINOv2 and SigLIP encoders) with 256-bin action tokens, trained on about 970k OXE trajectories, that beat the 55B RT-2-X by **16.5 points** with 7x fewer parameters ([OpenVLA](#), [arXiv 2406.09246](#)).

Token-by-token decoding was too slow and too coarse for dexterous, high-frequency control, so the **action-expert era (late 2024–2025)** split the model in two. Octo had already used a diffusion head on a small open transformer ([Octo](#), [arXiv 2405.12213](#)). π_0 made the split canonical: a PaliGemma-3B backbone plus a roughly 300M flow-matching expert that outputs **50-step chunks at up to 50 Hz** (π_0 , [arXiv 2410.24164](#)). Even the autoregressive camp adopted chunks. OpenVLA-OFT replaced token decoding with parallel decoding, continuous actions and an L1 loss, reaching **97.1% on LIBERO with about 26x faster action generation** ([OpenVLA-OFT](#), [arXiv 2502.19645](#)). FAST compressed chunks with a DCT plus byte-pair encoding, so autoregressive models could finally fit high-frequency data and train about 5x faster than π_0 ([FAST](#), [arXiv 2501.09747](#)).

The **hierarchy era (2025)** separated slow thinking from fast acting across processes and clock rates: Figure's Helix (S2 at 7–9 Hz, S1 at 200 Hz), NVIDIA GR00T N1 (VLM at about 10 Hz, flow DiT at about 120 Hz), and Gemini Robotics 1.5 (an embodied-reasoning orchestrator over a VLA) ([Figure Helix](#); [GR00T N1](#), [arXiv 2503.14734](#); [Gemini Robotics 1.5](#), [arXiv 2510.03342](#)). The **steerable-generalist era (late 2025–2026)** added RL from experience ($\pi^*0.6$ with RECAP), conditioning on structured prompts and generated subgoals ($\pi_0.7$), whole-body control (Helix 02, Gemini Robotics 2) and egocentric human-video pretraining at scale (GR00T N1.7 with EgoScale).



The pattern across these steps matters more than any single model. Each generation kept the previous one's pretrained knowledge and added a component that runs at a different rate or trains on a different signal. Reasoning moved up the stack, high-rate control moved down it, and learning moved from offline imitation to online experience [INF](#) .

02

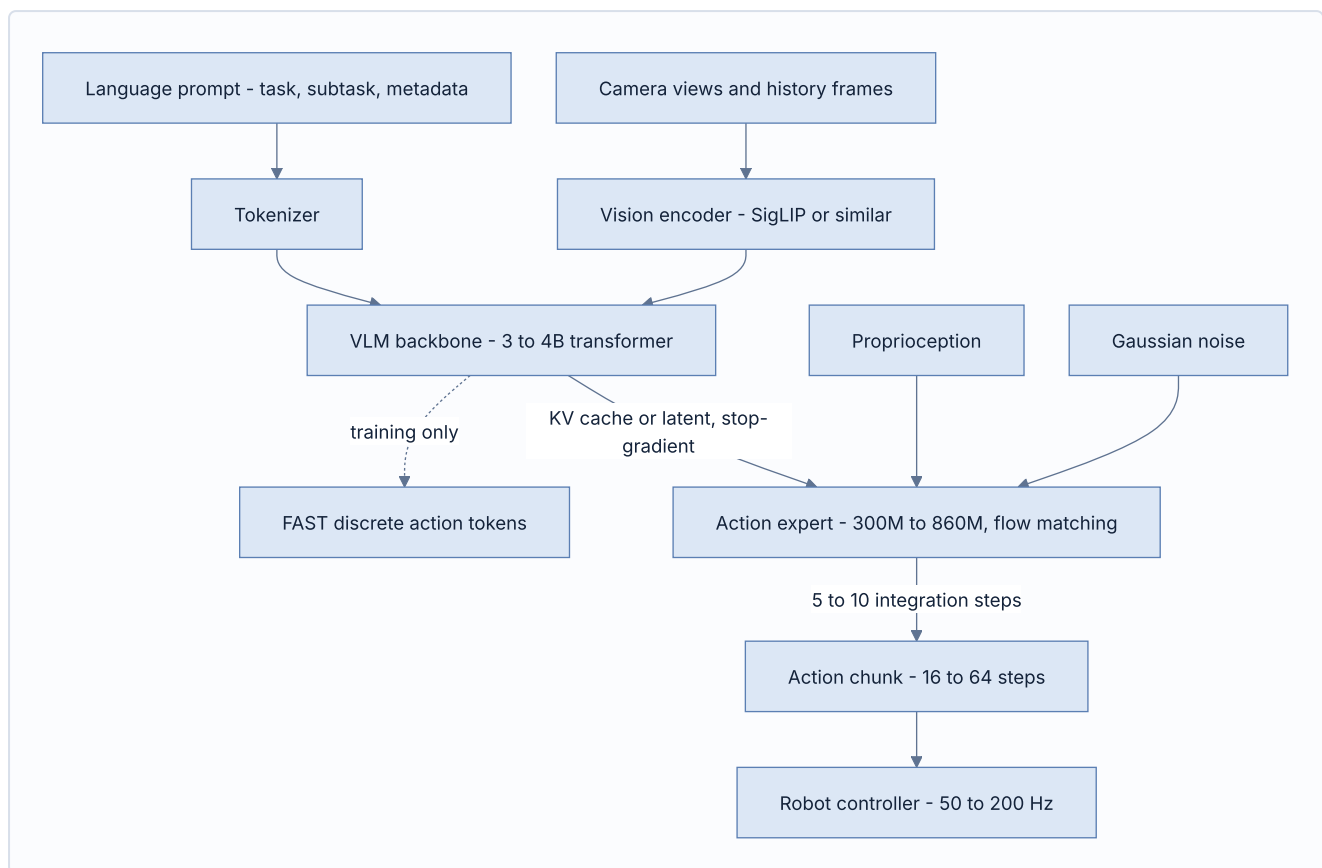
A 3–5B VLM plus a flow-matching expert is now the default stack

The field has converged on a **pretrained VLM backbone feeding a separate continuous action expert**. The expert is usually trained with flow matching (the π family, GR00T, SmolVLA, X-VLA) or diffusion (Octo, CogACT, RDT, DexVLA) and predicts chunks of roughly 16–64 steps. Discrete autoregressive tokens survive in two niches: as a *training signal* (FAST tokens in $\pi 0.5/\pi 0.6$ with knowledge insulation) and in unified reasoning-plus-action streams (Galaxea G0.5). The table summarizes the major models.

Model (date)	Backbone	Params	Action head	Chunk / control rate	Open weights
RT-1 (2022)	EfficientNet-B3 + FiLM + TokenLearner	~35M	256-bin discrete tokens, 11 dims	1 step / 3 Hz	Code only
RT-2 (2023)	PaLI-X 5B/55B, PaLM-E 12B	5–55B	Actions as text tokens	1 step / 1–5 Hz	No
Octo (2024)	Custom transformer	27M / 93M	Diffusion	Short chunks (length not verified)	Yes
OpenVLA (2024)	Llama 2 7B + DINOv2/SigLIP	7B	256-bin AR tokens	1 step / slow	Yes
OpenVLA-OFT (2025)	Same as OpenVLA	7B	Parallel decoding, L1 regression	Chunked / ~26x faster	Yes
RDT-1B (2024)	SigLIP + T5-XXL	1.2B	Diffusion transformer	64 steps	Yes
CogACT (2024)	VLM "cognition" feature	7B	Diffusion transformer	Chunked	Yes
$\pi 0$ (Oct 2024)	PaliGemma 3B	~3.3B (300M expert)	Flow matching, ~10 steps	50 steps / up to 50 Hz	Yes (openpi)
$\pi 0$ -FAST (Jan 2025)	PaliGemma 3B	~3B	AR over DCT+BPE tokens	Chunked	Yes
$\pi 0.5$ (Apr 2025)	PaliGemma	~3B	FAST in pretraining, flow expert added in post-training	Chunked	Yes (openpi)
$\pi 0.6$ / $\pi^*0.6$ (Nov 2025)	Gemma 3 4B	~4.9B (860M expert)	Flow expert, stop-gradient, plus FAST tokens	Not confirmed	No SEC
$\pi 0.7$ (Apr 2026)	Gemma 3 4B + video-history encoder	~5B SEC	Flow expert, 860M SEC	Not confirmed	No
GR00T N1 (Mar 2025)	Eagle-2 VLM	2.2B	Flow-matching DiT	16 steps / ~120 Hz	Yes
GR00T N1.6 (2025)	Cosmos-Reason-2B variant vs Eagle-Block2A-2B CONFLICT	~3B	32-layer DiT	16 steps	Yes

GR00T N1.7 (Apr 2026)	Cosmos-Reason2-2B (Qwen3-VL) SEC	~3B SEC	DiT, 32 vs 16 layers CONFLICT	Not confirmed	Yes, commercial license SEC
Helix (Feb 2025)	Onboard VLM, ~7B SEC	7B + 80M S1 SEC	Latent-conditioned visuomotor transformer	Continuous / 200 Hz	No
Gemini Robotics 1.5 / 2	Gemini 2.0-class	Undisclosed	Local action decoder	~50 Hz effective (GR 1)	No
SmoVLA (Jun 2025)	SmoVLM2, layer- skipped	450M	Flow-matching expert	50 steps, async	Yes
X-VLA (ICLR 2026)	Soft-prompted transformer	0.9B	Flow matching	Chunked	Yes
InternVLA-M1 (2025)	Qwen2.5-VL	~4B	Spatially guided action module	Chunked	Yes
Galaxea G0.5 (2026)	Unified AR decoder	Not verified	Interleaved reasoning and action tokens	Not verified	Not verified

The π family shows where growth goes. From π_0 to $\pi_{0.6}$ the base VLM moved only from PaliGemma 3B to **Gemma 3 4B**, while the action expert nearly tripled, **from about 300M to about 860M**, trained end to end on continuous actions and discretized tokens with a stop-gradient that keeps the expert from disturbing the backbone ($\pi_{*0.6}$, [arXiv 2511.14759](#)). A third-party reimplementation citing PI's November 2025 model card adds that $\pi_{0.6}$ raised SigLIP input resolution to 448×448, **cut flow denoising steps from 10 to 5**, and matched expert depth to the backbone's 34 layers ([OpenTau docs](#)) SEC. NVIDIA did the same: GR00T N1.6 doubled its DiT action head from 16 to 32 layers, replaced N1.5's 4-layer post-VLM adapter by unfreezing the top 4 VLM layers, and trains on joint flow-matching and world-modeling objectives ([HF model card](#)). Sources disagree on N1.6's backbone. The model card describes a Cosmos-Reason-2B variant, while a Chinese write-up says the config lists `nvidia/Eagle-Block2A-2B-v2` and that Cosmos-Reason2 arrived only with N1.7 ([CSDN](#)) CONFLICT. Total parameter counts stay at **0.5–5B** for frontier VLAs, an order of magnitude below frontier LLMs, because the model has to run within a robot's latency budget INF.



Action heads: why flow matching won, and where tokens still help

Each action representation trades expressiveness against speed and ease of RL. **Discrete bins** (RT-2, OpenVLA) inherit the LLM's training pipeline and give exact log-likelihoods, which suits policy-gradient RL. But they need one forward pass per action dimension, quantize away precision, and handle multimodal high-frequency data poorly. **Compressed tokens** (FAST) fix the length problem by turning a chunk into a few DCT coefficients, and FAST+ is a universal tokenizer trained on about 1M trajectories ([FAST](#)). **L1 regression with parallel decoding** (OFT) is the simplest fast option and is very strong on unimodal benchmarks like LIBERO, but it averages across modes. **Diffusion and flow heads** model the full action distribution in continuous space and need only a handful of integration steps. Flow matching won production because it combines diffusion-quality multimodality with fewer, deterministic integration steps, and because it trains well as a small expert attached to a frozen or insulated backbone [INF](#).

The open question was how to keep a powerful expert from wrecking the backbone's web knowledge. **Knowledge insulation** answers it. The backbone trains on discrete FAST action tokens (plus VLM data), the flow expert reads the backbone's representations, and a stop-gradient keeps the expert's gradients out of the backbone. This gives faster training and preserves the VLM's semantics ([Knowledge Insulating VLA, arXiv 2505.23705](#)). $\pi 0.6$ confirms the stop-gradient in production ([arXiv 2511.14759](#)). Hybrids continue: HybridVLA ensembles autoregressive and diffusion outputs inside one LLM ([arXiv 2503.10631](#)), and Galaxea G0.5 returns to a single autoregressive stream that interleaves reasoning and action tokens under one objective ([arXiv 2608.11739](#)). A 2026 paper titled "The Compression Gap" argues that discrete tokenization limits VLA scaling ([awesome-physical-ai list](#)) [UNV](#). The safe bet for a new system is a continuous flow expert, with discrete tokens used as an auxiliary signal for the backbone [INF](#).

Cross-embodiment designs absorb heterogeneity with padding, prompts and mixtures

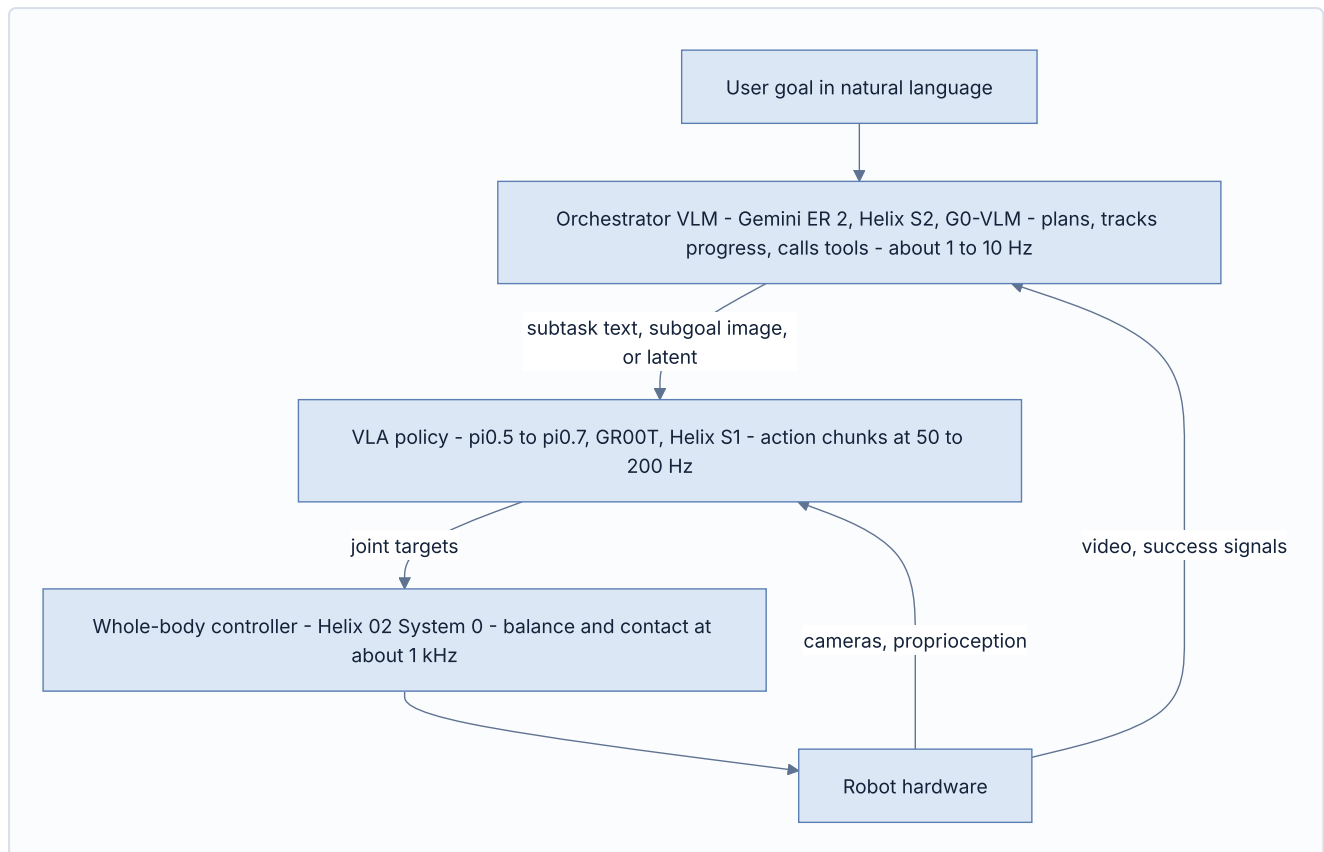
Most generalists handle different robots crudely, and it works. π^0 zero-pads every robot's state and action vectors to the largest dimension in its mixture (18) and reweights task-robot combinations by $n^{-0.43}$ to down-weight over-represented ones (π^0) [UNV](#) . X-VLA learns a soft-prompt embedding per data source. It adapts to a new domain by freezing the backbone and learning new prompts, and a LoRA that updates about **9M parameters (about 1%)** reaches 93% on LIBERO ([arXiv 2510.10274](#)). Gemini Robotics 1.5's "Motion Transfer" lets one checkpoint control ALOHA, a bi-arm Franka and the Apollo humanoid without robot-specific fine-tuning or explicit action-space alignment ([arXiv 2510.03342](#)). Galaxea G0 goes the other way: its three-stage curriculum found single-embodiment pretraining critical ([arXiv 2509.00576](#)). The two views fit together. Cross-embodiment data builds broad priors, and a phase on the target body supplies the precise dynamics [INF](#) .

03

Fast and slow systems split into three tiers

Two families implement the System 1 / System 2 idea. **Explicitly separated systems** run distinct networks at distinct rates. Figure's Helix runs an onboard internet-pretrained VLM (reported as 7B) at 7–9 Hz and an 80M visuomotor transformer at 200 Hz. The fast system turns the slow system's latent vector into continuous upper-body actions, and the two are trained end to end with gradients flowing from S1 into S2 ([Figure Helix](#); sizes via [The Humanoid Hub](#) SEC). GR00T N1 pairs an Eagle-2 VLM at about 10 Hz with a flow-matching DiT producing 16-step chunks at about 120 Hz ([arXiv 2503.14734](#)). Galaxea G0 uses a planner VLM that breaks instructions into subtasks above an executor VLA ([arXiv 2509.00576](#)). Gemini Robotics 1.5 makes the split agentic. ER 1.5 orchestrates: it plans, detects success, and calls tools such as web search and code. GR 1.5 executes ([arXiv 2510.03342](#)). **Single-network systems** ($\pi 0.5$ through $\pi 0.7$) get much of the same separation inside one model, through the backbone/expert split and knowledge insulation.

The 2026 trend adds a third tier below the VLA. **Helix 02 (January 2026)** adds "System 0" at 1 kHz for balance and whole-body coordination. S1 stays at 200 Hz but now outputs full-body joint targets, and S2 handles scene and language understanding. The demo was a 4-minute autonomous dishwasher unload and reload ([Figure Helix 02](#); [eWeek](#)). **Gemini Robotics 2 (July 30, 2026)** comes as three models. The GR 2 VLA controls legs, torso, arms and fingers under one policy. ER 2 adds real-time video understanding, task-progress tracking, multi-robot task division and resuming from the last successful step, and is available through the Gemini API. On-Device 2 is built from Gemini Robotics 1.5 technology and on-device Gemma models ([DeepMind On-Device 2 model card](#); [Robotics & Automation News](#)) SEC. One outlet says Gemini Robotics 2 was never announced ([Scalewise](#)). That contradicts DeepMind's own model card and the rest of the coverage and is probably outdated CONFLICT.



The three-tier split is driven by physics, not fashion. Balance and contact need millisecond loops that no 3B transformer can serve. Language understanding and planning tolerate hundreds of milliseconds. The VLA sits in between, converting intent into motion at the rate where manipulation happens [INF](#). A practitioner should design the interfaces between tiers carefully: latent vectors (Helix), text subtasks ($\pi 0.5$, G0), subgoal images ($\pi 0.7$) or tool calls (Gemini ER). Those interfaces decide what can be inspected, steered and trained separately.

04

Reasoning became conditioning: from chain-of-thought to steerable prompts

Embodied reasoning has gone through three stages. ECoT trained OpenVLA to write out a plan, the current subtask, a movement description, the gripper position and object bounding boxes before acting, and reported about a **28-point absolute** gain over OpenVLA (ECoT, [arXiv 2407.08693](#)). The cost was latency, because every control step waited for generated text. $\pi 0.5$ made reasoning hierarchical inside one model. It first predicts a high-level text subtask ("pick up the pillow"), then low-level actions through the flow expert, and is co-trained on multi-robot data, web data, subtask labels and verbal instructions ($\pi 0.5$, [arXiv 2504.16054](#)). Gemini Robotics 1.5 "thinks before acting", interleaving actions with multi-level natural-language thinking traces. An ablation shows better multi-step progress with thinking turned on, though the researchers did not retrieve the exact numbers ([arXiv 2510.03342](#)).

The 2026 stage is **steerability**, best shown by $\pi 0.7$. Its prompts carry far more than task text: subtask instructions, subgoal images, episode metadata (speed discretized in 500-step bins, a quality score from 1 to 5, annotated mistake segments) and control mode. At test time a lightweight world model initialized from BAGEL generates the subgoal images, near-future frames showing what success looks like. Conditioning on metadata lets the model keep improving on mixed-quality data instead of being dragged down by bad demonstrations, because at inference it can be asked for "fast, quality 5, no mistakes" ([arXiv 2604.15483](#); [swiftscholar](#) SEC). Architecturally, $\pi 0.7$ reportedly totals about 5B parameters. It pairs a Gemma 3 4B backbone with a "MEM-style" video-history encoder and an 860M action expert, and accepts up to 4 camera views with up to 6 history frames each plus up to 3 subgoal-image views (Terry Um summary; [hyper.ai](#)) SEC. PI claims that this single generalist matches the RL-tuned $\pi^* 0.6$ specialists on laundry, espresso and box building, and folded shirts zero-shot on a UR5e arm with no laundry data from that platform ([oodaloop/TechCrunch](#)) VENDOR. The Decoder's critical coverage notes that the generalization arrives with LLM-like failure modes ([The Decoder](#)).

Steerability turns reasoning into structured conditioning signals that can come from humans, VLM planners or world models INF. It is a better engineering abstraction than free-form chain-of-thought. A subgoal image or subtask string is cheap to produce, can be checked by a separate model, and can be supplied by an operator when the planner is wrong. Two neighbouring threads support the same conclusion. **Spatial grounding** gives the policy explicit "where" signals. InternVLA-M1 is a roughly 4B Qwen2.5-VL model that grounds spatially before generating actions and reports a **95.9% LIBERO average** ([arXiv 2510.13778](#)). Gemini Robotics-ER provides 2D/3D pointing, boxes and trajectories ([arXiv 2503.20020](#)), and SpatialVLA adds Ego3D position encodings ([arXiv 2501.15830](#)). **Latent actions** let models learn from video without action labels. LAPA pretrains on VQ-VAE latent actions learned from frame pairs ([arXiv 2410.11758](#)), UniVLA learns task-centric latents in DINOv2 space ([arXiv 2505.06111](#)), and AgiBot GO-1 inserts a latent action planner between VLM and expert ([arXiv 2503.06669](#)). **Memory** is the newest addition: $\pi 0.7$'s history encoder is among the first production moves beyond single-frame or short-history policies. Its "MEM" design has not been confirmed from the primary source UNV.

05

Data corpora jumped from 10k to a claimed 270k hours, and diversity still beats volume

Robot data has grown faster than models. **Open** data went from about 1M pooled OXE trajectories (2023) to single-platform corpora of about 1M trajectories (AgiBot World, 2025). **Proprietary** corpora, counted in hours rather than trajectories, are one to two orders of magnitude larger. Because the two are counted in different units, the comparisons below are approximate.

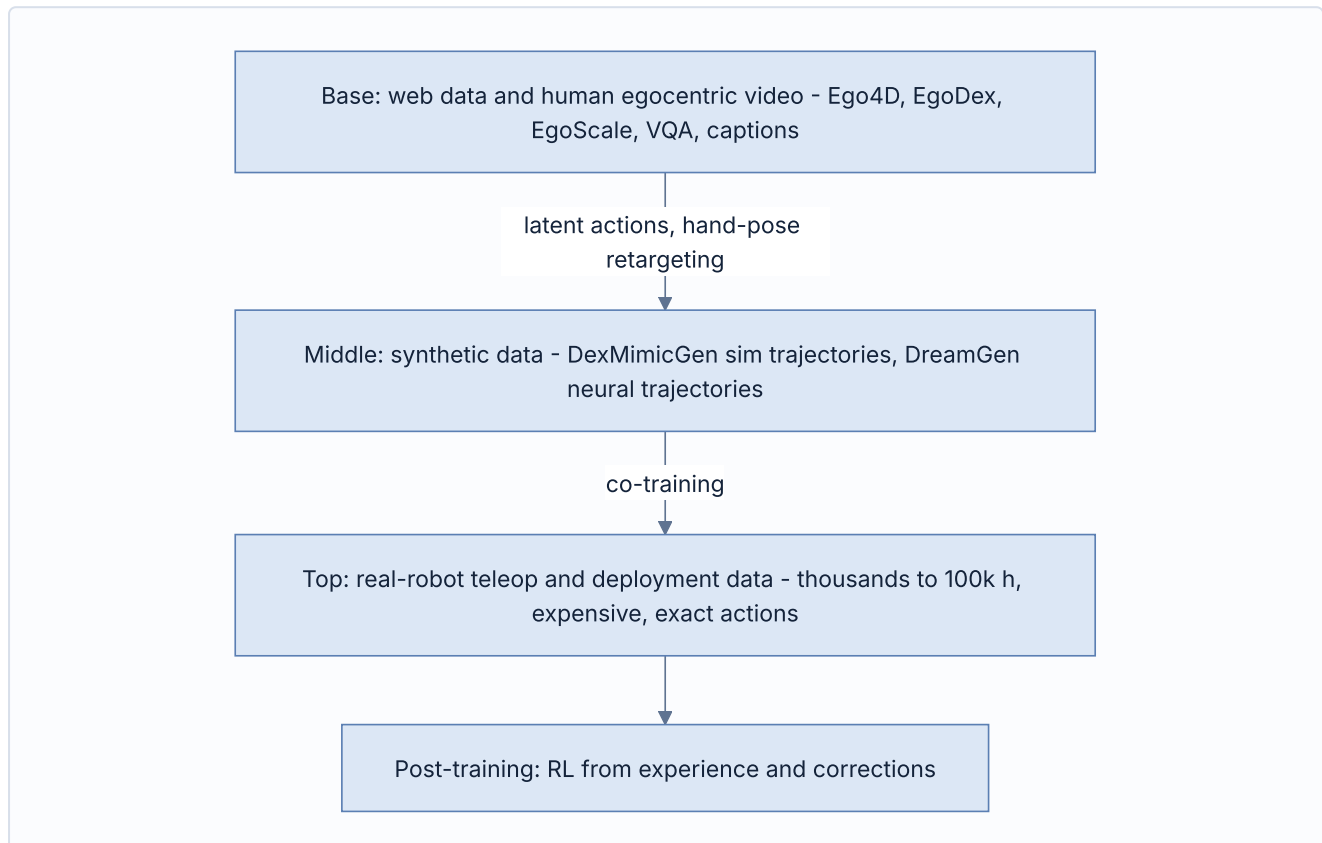
Dataset (year)	Scale	Embodiments / collection	Provenance
Open X-Embodiment (2023)	1M+ trajectories, 60 datasets, 34 labs	22 embodiments	Primary; one survey says 2M+ CONFLICT
BridgeData V2 (2023)	60,096 trajectories, 24 environments	WidowX	Primary
DROID (2024)	~76k trajectories, 564 scenes, ~350 h	Franka, 52 buildings	SEC
AgiBot World (2025)	1,003,672 trajectories (~43.8 TB), 217 tasks; curated Alpha 92,214	100 robots, 4,000 m² facility	Primary
Galaxea Open-World (2025)	500+ h, 150+ tasks, 50 scenes	Single dual-arm mobile body	Primary hours; task counts SEC
LeRobot community v3	791 datasets, ~25.97M frames	46 embodiments, mostly low-cost	Primary (HF page)
π0 mixture (2024)	10,000 h + OXE/DROID/Bridge	7 configurations, 68 tasks	Primary; ~91% proprietary split UNV
Xiaomi-Robotics-1 (2026)	100k+ h pretraining; 7,200+ h robot + 1,000+ h UMI post-training	Mainly UMI-style handheld	Primary
GEN-0 (Nov 2025)	~270,000 h, +10,000 h per week	Thousands of homes and workplaces	VENDOR
EgoDex (2025)	829 h, 338k episodes, ~90M frames	Apple Vision Pro, 3D hand poses	Primary
EgoScale (2026)	20,854 h egocentric, action-labeled	Human video	SEC

Sources: [OXE](#), [BridgeData V2](#), [DROID summary](#), [AgiBot World](#), [GalaxeaVLA README](#), [LeRobot community v3](#), [π0](#), [Xiaomi-Robotics-1](#), [GEN-0](#), [EgoDex](#), [Data Pyramid survey](#).

Two shifts stand out. First, the two largest corpora (Xiaomi at 100k+ hours, Generalist at a claimed 270k) are built on **wearable or handheld collection rather than robot teleoperation**. Pretraining data is moving off the robot INF. Second, the open/closed gap is now roughly **100–1000x in hours**: DROID is about 350 h and Galaxea 500+ h, against hundreds of thousands in industry INF. Academic labs cannot win on volume, so their best contributions are mixture methods, data-efficient post-training and evaluation.

The data pyramid formalizes three tiers of supervision

NVIDIA's GR00T N1 paper formalized a **data pyramid**. The base holds the most abundant data, web data and human video. The middle holds synthetic data from simulation and neural-generated trajectories. The top holds the rarest and most valuable data, real-robot teleoperation (GR00T N1; [NVIDIA whitepaper](#)). A 2026 survey is titled "Data Pyramid for Embodied Manipulation", which shows how far the framing has spread ([arXiv 2607.24744](#)).



The base is growing fastest. **EgoDex** offers 829 h of 1080p/30 Hz egocentric video with tracked 3D hand and finger joints across 194 tabletop tasks, from tying shoelaces to folding laundry ([EgoDex](#); [Apple ML](#)). **PH2D** collected about 3.02M frames and 26,824 human demos on consumer VR headsets, against EgoMimic's 432k frames, and co-trains a Human Action Transformer on human and humanoid data ([arXiv 2503.13441](#)). **EgoScale** scales action-labeled egocentric pretraining to **20,854 hours** and reportedly finds consistent gains from 1k to 20k hours, described as a log-linear scaling law between human data volume and robot performance ([Data Pyramid survey](#); [Not All Actions Are Equal](#)) [SEC](#). GR00T N1.7 builds EgoScale into its pretraining for dexterity ([GitHub mirror](#)) [SEC](#). In about a year, human-video datasets went from hundreds of hours to tens of thousands. Together with the wearable-gripper corpora, "human-embodied" data is becoming the main pretraining source [INF](#).

Collection hardware has collapsed in price. ALOHA cost about USD 20k as published in 2023, Mobile ALOHA about USD 32k, and the U-ARM paper puts dual-arm ALOHA above USD 50,000 at current prices. A GELLO leader arm costs about USD 150–300, and a UMI gripper about USD 371 in parts ([dexset.ai](#); [U-ARM](#); [dexset.ai 2026 guide](#)) [SEC](#). UMI-style grippers underlie both the Lin et al. scaling-law study and Xiaomi's 100k-hour corpus.

The middle of the pyramid is generated. GR00T N1 used DexMimicGen to produce **780,000 simulation trajectories in 11 hours**. NVIDIA describes this as equivalent to 6,500 hours of human demonstrations, which is a time-equivalence estimate rather than recorded hours ([GR00T N1](#)) [VENDOR](#). DreamGen fine-tunes a video world model on the target robot, prompts it for new behaviors, labels the videos with pseudo-actions from a latent-action or inverse-dynamics model, and trains on the result. Starting from pick-and-place teleop in one environment, a GR1 humanoid learned 22 new behaviors. It scored **43.2% on new behaviors in seen environments and 28.5% in unseen ones**, where GR00T N1 trained on pick-and-place alone scored near zero ([DreamGen](#)). Neural trajectories can teach new verbs, but the modest absolute success says they extend the real data rather than replace it [INF](#).

Web co-training protects generalization, not in-distribution skill

Co-training on web data consistently helps on unseen objects and instructions while leaving in-distribution performance roughly unchanged. RT-2's jump from 32% to 62% on unseen scenarios was the first evidence ([RT-2](#)). In $\pi 0.5$'s "no web data" ablation, removing captioning, VQA and detection data makes little difference on the aggregate mock-home test but hurts on unseen objects. The authors attribute this to web knowledge of object categories that never appear in robot data ([\$\pi 0.5\$](#)). Secondary write-ups put the out-of-distribution cost at about 14–20 points, for example 94% vs 74% OOD success, but nobody checked these figures against the paper ([robo2u](#); [ay-robots](#)) [SEC](#). The lesson is that VQA data protects the backbone's open-vocabulary grounding, the part of the model that robot data alone cannot supply.

Scaling laws: diversity in academia, hours in industry

There are two kinds of scaling evidence, and they answer different questions. The academic result is Lin et al. (ICLR 2025 oral). Training Diffusion Policy on 40,000+ UMI demonstrations and evaluating with 15,000+ real rollouts, they found generalization follows roughly a **power law in the number of environments and objects, not in the number of demos**. Diversity is "far more important" than demo count, and returns per environment saturate past a threshold. Four collectors working one afternoon gathered enough data for about 90% success on two tasks in new environments with unseen objects ([arXiv 2410.18647](#)). The study covers single-task policies only. ReMix makes the same point from the mixture side. Learned domain weights on OXE beat uniform weights by more than 40% and human-curated weights by more than 20%, and reach competitive performance with only **25% of the data** ([ReMix](#), [PMLR](#)). SmolVLA shows that small, noisy, diverse community data still helps a lot: pretraining on about 10M frames from 487 LeRobot community datasets lifted SO-100 suite success **from 51.7% to 78.3%** ([SmolVLA](#)).

The industry results claim that sheer hours also scale. Generalist reports that GEN-0's downstream post-training error follows a power law in pretraining data, which one summary gives as about $D^{-0.5}$ ([Generalist blog](#); [aibase](#)) [VENDOR](#). It also claims a model-size threshold below which models "ossify" and stop absorbing data. The Batch places the threshold at **7B parameters**, while 36Kr describes **10B+** models adapting with less post-training data ([The Batch](#); [36Kr](#)) [CONFLICT](#) [VENDOR](#). Xiaomi-Robotics-1 reports, in a paper rather than a blog, that more pretraining data or a bigger model consistently lowers action-prediction error at the 100k-hour scale ([arXiv 2607.15330](#)). Action-prediction error is not task success, and as of October 2026 **no peer-reviewed, independently replicated scaling law for generalist VLAs exists at 100k+ hours** [INF](#). Nobody has published a compute-optimal (Chinchilla-style) analysis either. The two kinds of evidence are compatible. Diversity buys generalization, and volume pays off once diversity is present and the model has the capacity to absorb it [INF](#).

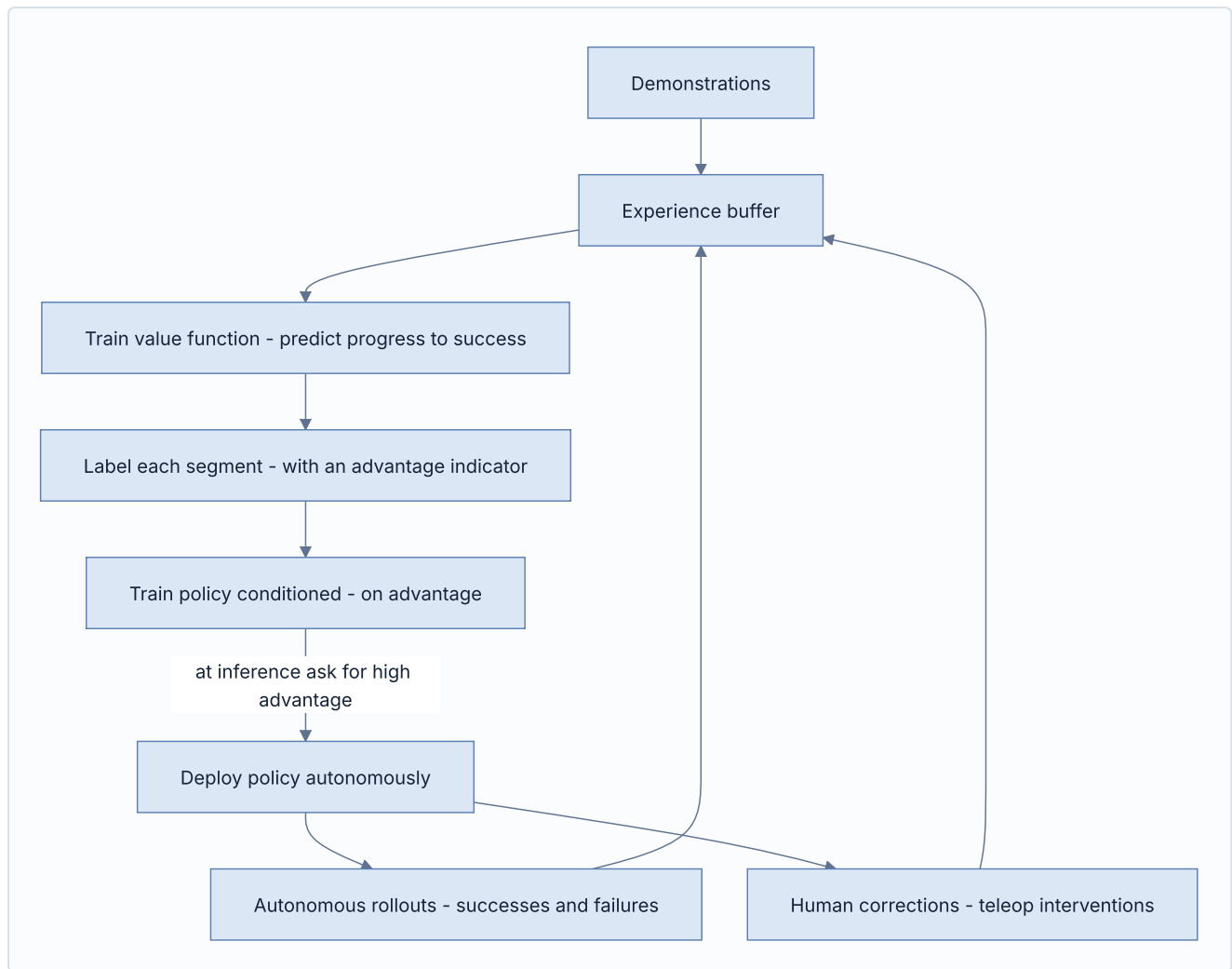
06

RL post-training doubled real-world throughput, mostly with humans in the loop

Reinforcement learning became the main lever after pretraining in 2025–2026, and the strongest real-world results all combine RL with human corrections.

RECAP: advantage-conditioned learning from experience and corrections

Physical Intelligence's $\pi^*0.6$ trains $\pi0.6$ with **RECAP**, "RL with Experience & Corrections via Advantage-conditioned Policies". The training data has three sources: demonstrations, expert teleoperated corrections during autonomous runs, and the robot's own autonomous rollouts. A learned value function scores each segment, and the policy is conditioned on an advantage indicator so it learns to produce "high-advantage" actions. At deployment it is asked for good actions only ([arXiv 2511.14759](#); [PI PDF](#)). PI reports that RECAP "can more than double the throughput on some of the hardest tasks and decrease failure rates by 2x or more". The largest gains came on diverse laundry folding and espresso making, with box assembly as the third task. The robustness demos include folding novel laundry in a new home for over 2 hours without interruption, and espresso for **13 hours straight** according to the paper. A secondary source says 18 hours ([arXiv 2511.14759](#); [postcutoff.com](#)) **CONFLICT**. The 13-hour figure from the paper should be preferred. No weights or API were released ([sudoremove](#)) **SEC**.



RECAP is closer to offline RL than to classic online RL. It conditions on advantage instead of running on-policy PPO, so it works more like "filtered or weighted behavior cloning with a learned critic" [INF](#). That is why it fits a 5B flow-based model trained on messy real-world mixtures. It never needs flow-policy likelihoods, it reuses every interaction, and corrections enter as ordinary data. $\pi_{0.7}$ carries the idea further by conditioning on quality and mistake metadata, which is a hand-labeled cousin of the advantage indicator [INF](#).

Simulation RL saturates LIBERO and rescues one-demo policies

In simulation, outcome-reward online RL (GRPO, RLOO, PPO) applied to supervised VLAs saturates LIBERO. The gains are largest with very few demonstrations.

Method	Base policy	Algorithm	Headline result
SimpleVLA-RL (ICLR 2026)	OpenVLA-OFT	GRPO, binary outcome reward	LIBERO 91% → 99%; 1-demo 48.9% → 96.9%; LIBERO-Long 1-demo 17.3% → 91.7%; RoboTwin 1.0 39.8 → 70.4
RIPT-VLA	QueST 20M; OpenVLA-OFT 7B	REINFORCE leave-one-out	QueST +10.9 pts to 93.6%; OFT 97.5%; 1-demo 4% → 97% in 15 iterations
RL4VLA (NeurIPS 2025)	OpenVLA	PPO vs GRPO vs DPO	PPO best; gains in semantic and execution generalization, not visual robustness
π RL (RLinf)	π_0 , $\pi_{0.5}$	PPO / GRPO with Flow-Noise, Flow-SDE	Few-shot π_0 57.6% → 97.6%; $\pi_{0.5}$ 77.1% → 98.3%; $\pi_{0.5}$ 1-shot LIBERO-Long 43.9% → 94.0%
GRAPE	OpenVLA	Trajectory-level preference (DPO-style)	+51.79% in-domain, +58.20% unseen, -37.44% collisions when safety-aligned

Sources: [SimpleVLA-RL](#); [RIPT-VLA](#); [RL4VLA](#); [\$\pi\$ RL](#); [GRAPE](#).

Three lessons come out of this table. First, the headline value of RL is **data efficiency**. Policies trained on one demo go from unusable to more than 90%, and SimpleVLA-RL observed a "pushcut" behavior, an action pattern never seen in the demos ([SimpleVLA-RL](#)). Second, RL improves **semantic and execution generalization but not visual robustness** ([RL4VLA](#)), so it does not replace diverse visual data. Third, the move from token VLAs to flow VLAs made RL harder, because flow policies have no cheap log-likelihood. π RL turns denoising into a discrete-time MDP with a learnable noise network (Flow-Noise) or converts the ODE to an SDE for exploration (Flow-SDE). RLinf-VLA describes it as the first policy-gradient method for flow-matching VLAs ([RLinf-VLA](#)). The RLinf docs report slightly different numbers for another PPO configuration (π_0 at 96.0% average, $\pi_{0.5}$ at 97.9%) ([RLinf docs](#)), a reminder that LIBERO differences of one point are configuration noise [INF](#).

Real-world RL means RL plus interventions

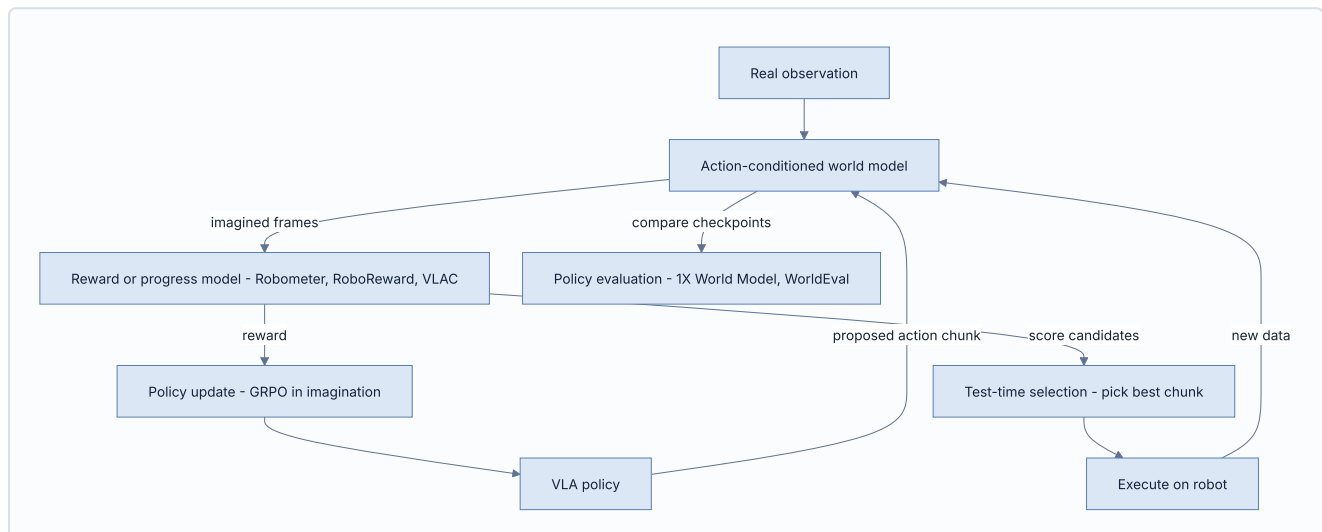
On real hardware, human-in-the-loop off-policy RL is now reliable for single contact-rich tasks. **HIL-SERL** reaches near-perfect success with fast cycle times after **1 to 2.5 hours** of real-world training. Against HG-Dagger, its average cycle time was 5.4 s vs 9.6 s, and the HG-Dagger baseline averaged 49.7% success ([Science Robotics](#); [arXiv 2410.21845](#)). **ConRFT** brings this to VLAs. Applied to Octo, it combines Q-learning and BC losses offline, then fine-tunes online with task rewards and interventions, reaching **96.3% average success across 8 real tasks within 45–90 minutes**. Only the action head is tuned, with the backbone frozen ([arXiv 2502.05450](#)). **DSRL** needs only black-box access. It runs RL over the *input noise* of a frozen diffusion or flow policy. Summaries report LIBERO going from about 20% to about 100% within roughly 10k samples, and one π_0 -DROID real task going from 20% to 90% in under 50 episodes ([arXiv 2506.15799](#); [secondary](#)) [SEC](#).

Human corrections appear in every strong real-world result: HIL-SERL, ConRFT and RECAP [INF](#). For anyone building a VLA, the intervention interface is therefore a first-class piece of infrastructure, not a debugging tool. It needs a teleop override, logging of who intervened and when, and a pipeline that turns corrections into training data within hours.

07

World models, reward models and verifiers are the new compute sinks

Learned world models now play three roles in VLA development: an RL environment, a data generator and a policy evaluator. Reward and progress models are the component that makes the first and third roles work.



As an RL environment. VLA-RFT uses a data-driven world model as a controllable simulator with verified rewards and fine-tunes with GRPO. It reports **+4.5 points average LIBERO success with fewer than 400 steps** (arXiv 2510.00406) SEC. World-Env does RL post-training entirely inside a world model under extreme data scarcity (arXiv 2509.24948). These gains are much smaller than simulator RL gains, which points to world-model fidelity and reward verification as the bottleneck INF. **As a data generator,** DreamGen (above) is the reference. GR00T N1.5 reached 38.3% across 12 DreamGen tasks vs 13.1% for N1, and co-training improved RoboCasa by 4–9 points at 30–300 demos (GR00T N1.5; DreamGen). **As a planner,** Meta's V-JEPA 2-AC is an action-conditioned latent world model post-trained on under 62 hours of unlabeled DROID video. Planning zero-shot toward image goals on Franka arms in two labs, it reached 80% pick-and-place success for cups and 65% for boxes (arXiv 2506.09985). An independent re-run in the AD-WM paper got only 19/45 for V-JEPA 2-AC vs 32/45 for AD-WM (arXiv 2609.30264), the usual drop when a method leaves its authors' lab INF. **As an evaluator,** 1X's world model compares policies under identical conditions for checkpoint selection, architecture comparison and dataset curation. 1X reports high correlation with real-world results, improving with more autonomous-rollout data (1X World Model) VENDOR. Academic counterparts are WorldEval (arXiv 2505.19017) and dWorldEval (arXiv 2604.22152). Evaluation is the most mature of the four roles, because ranking policies tolerates more model error than training on imagined rewards does INF. $\pi 0.7$'s use of a BAGEL-initialized world model to generate subgoal images is a fifth role: imagination as an interface rather than a training signal.

Reward and progress models matured quickly in 2025–2026. GVL prompts a VLM with shuffled frames to estimate per-frame progress, scored by Value-Order Correlation. VLAC, trained on about 300k human and robot trajectories, predicts the relative progress between two frames. Robometer is a Qwen3-VL-4B model trained on trajectory comparisons that outputs per-frame progress, success and preferences. RoboReward fine-tunes Qwen3-VL to output a 1–5 progress score on OXE and RoboArena data (Robometer; RoboReward).

In a third-party comparison across six robot setups, average scores were **GVL 0.199**, **VLAC-8B 0.049**, **RoboReward-4B 0.502** and **Robometer 0.655** ([RynnValue](#), [arXiv 2608.09853](#)) **SEC**. Trained trajectory-comparison models clearly beat prompted VLMs. Reward models are what make real-world RL possible without hand-built success detectors, and what turns world-model RL into "verified" RL **INF**.

Test-time compute is the third axis. RoboMonkey samples candidate actions from the VLA, perturbs them with Gaussian noise, and picks one with a VLM verifier. It reports **+25 points on out-of-distribution tasks and +9 in distribution** over OpenVLA, but 16 candidates take about 650 ms (about 1.5 Hz) on one H100 ([arXiv 2506.17811](#); [PMLR](#)). V-GPS re-ranks sampled actions with an offline-RL value function and improved 5 different generalist policies across 12 tasks ([arXiv 2410.13816](#)). The caveat matters: a 2026 paper reports that RoboMonkey "fails catastrophically on most tasks — often performing even worse than the base pi0 model" in its setup ([arXiv 2602.12281](#)). Verifier quality matters more than sample count. Test-time selection helps weak policies most and can hurt strong ones when the verifier is miscalibrated **INF**. Our own lab found the same thing in miniature (below). τ 0-VLA moves world-model-guided test-time computation into a hierarchical model ([arXiv 2608.16885](#)) **UNV**.

08

LIBERO is saturated, and robustness suites expose the gap

Benchmarks have not kept up with models. **Clean LIBERO** averages now sit at 94–99%, inside the run-to-run and reproduction noise of the leading methods, and 2026 papers routinely call it saturated.

Model	Spatial	Object	Goal	Long	Average	Source table
OpenVLA-OFT	97.6	98.4	97.9	94.5	97.1	StarVLA
π 0	96.8 / 98.0	98.8 / 96.8	95.8 / 94.4	85.2 / 88.4	94.1 / 94.4 CONFLICT	StarVLA / PhysBrain
π 0.5	98.8	98.2	98.0	92.4	96.9	PhysBrain
GR00T N1.5	92.0	92.0	86.0	76.0	86.5	StarVLA
GR00T N1.6	97.7	98.5	97.5	94.4	97.0	PhysBrain
InternVLA-M1	98.0	99.0	93.8	92.6	95.9	InternVLA-M1 paper
Wall-OSS-0.5	—	—	—	—	96.5	G0.5 paper
SimpleVLA-RL (OFT + RL)	—	—	—	—	~99	SimpleVLA-RL

Sources: [StarVLA](#), [arXiv 2604.05014](#); [PhysBrain 1.0](#), [arXiv 2605.15298](#); [InternVLA-M1](#); [G0.5](#); [SimpleVLA-RL](#). The values come from search excerpts of each paper's re-tabulation, and the same model differs by 0.3–1 point across papers.

LIBERO-Long is the only suite that still separates models. The perturbation suites tell the real story. **LIBERO-PRO** shows that high LIBERO scores "often fail to translate" to tasks with minor variations, and frames the problem as memorization rather than understanding ([arXiv 2510.03827](#)). **LIBERO-Plus** perturbs 7 dimensions (camera, robot initial state, language, lighting, background, noise and layout) across 21 sub-components. OpenVLA-OFT drops from about 84 on its clean comparison to **69.6**, and OpenVLA falls to **15.6** ([arXiv 2510.13626](#)) **UNV**. An OpenPI robustness average of 85.31% on LIBERO-Plus appears in IndustrialVLA-Bench ([arXiv 2609.25562](#)) **SEC**. On SimplerEnv-WidowX, PhysBrain 1.0 claims an 80.2% average, 23.1 points above π 0.5 and GR00T-N1.6-Bridge (which implies about 57% for both), and OA-WAM claims 79.3 ([PhysBrain](#); [OA-WAM](#)) **VENDOR**. MemoryVLA's table footnotes that its π 0 SimplerEnv number was reproduced with extra proprioception and is not directly comparable ([arXiv 2508.19236](#)). Footnotes like this are why cross-paper tables need care.

Long-horizon household tasks are far from solved. The **NeurIPS 2025 BEHAVIOR-1K Challenge** ranks by Q-score, which gives partial credit for satisfied goal conditions. The winners were Robot Learning Collective, Comet and SimpleAI Robot, with held-out Q-scores of about **0.260**, **0.251** and **0.159** ([BEHAVIOR 2025 archive](#); scores via [benchmarklist](#) **SEC**). A 2026 edition is running ([BEHAVIOR challenge](#)).

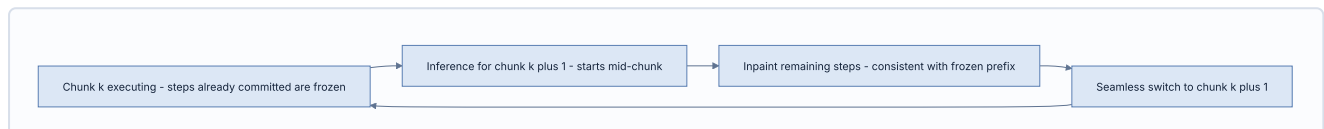
Real-world evaluation is the deepest problem. Labs choose their own tasks, run small numbers of trials on hardware nobody else has, and publish vendor-run demos **INF**. **RoboArena** is the main proposed fix. Evaluators pick their own tasks and environments and run double-blind pairwise A/B comparisons that are aggregated into a global ranking. Its first study covered 7 institutions on the DROID platform, with 600+

pairwise episodes across 7 generalist policies ([RoboArena](#), [arXiv 2506.18123](#)). Together with world-model evaluators (1X, WorldEval), this points to an "arena plus simulator" evaluation stack replacing fixed benchmarks for generalists **INF**.

09

Deployment hides latency in the execution layer

A 3–5B model cannot produce a fresh action every 5 ms, so deployment has converged on **chunking plus overlap**. The policy predicts a chunk of future actions, and the next chunk is computed while the current one executes. Synchronous chunking pauses between chunks and produces jerky motion, and mobile robots that use a remote server add network latency on top ([PI: Real-Time Chunking](#)).



Real-Time Chunking (RTC) from Physical Intelligence freezes the actions guaranteed to execute during inference and inpaints the rest, so consecutive chunks join smoothly. It works with any diffusion- or flow-based VLA, including $\pi 0.5$, with no change to training. As injected latency grows, it holds up better than naive async execution, Bidirectional Decoding and Temporal Ensembling. On real robots it survived +100 ms and +200 ms of injected delay, where Temporal Ensembling failed ([arXiv 2506.07339](#), NeurIPS 2025). PI later released a training-time variant of RTC and used it in the $\pi^*0.6$ espresso demo ([PI RTC page](#)). **SmolVLA's asynchronous inference** separates prediction from execution in the open-source stack. Hugging Face reports about 30% faster response and 2x task throughput, single-GPU training, and deployment on consumer GPUs or CPU ([HF blog](#)) [VENDOR](#). A 2026 analysis compares asynchronous inference methods for VLAs systematically ([arXiv 2605.08168](#)).

On-device inference is the second front. Gemini Robotics already split a cloud backbone (under about 160 ms) from a local action decoder, for about 250 ms end to end and about 50 Hz effective control through chunks ([arXiv 2503.20020](#)) [UNV](#). Gemini Robotics On-Device (June 2025) moved a VLA onto bi-arm robots, and On-Device 2 (July 2026) builds on on-device Gemma models ([The Robot Report](#); [DeepMind model card](#)). Open tooling is catching up. Jetson-PI reports control-frequency gains of **8.66x over naive PyTorch** and **5.41x over vla.cpp** on Jetson Orin, plus +14.8% over VLASH on LIBERO ([alphaxiv 2607.12659](#)) [UNV](#). vla.cpp runs SmolVLA and GR00T N1.5/1.6/1.7 on CPU, Apple Silicon, CUDA and Jetson-class boards ([GitHub](#)). Model-side efficiency appears in $\pi 0.6$'s cut from 10 to 5 denoising steps [SEC](#) and SmolVLA's layer skipping. The research team found no primary source comparing quantization or distillation across VLAs. The deployment pattern is now stable: a slow reasoning model (cloud or onboard), a fast reactive expert, and RTC or async overlap to hide the gap. Hardware vendors are pushing the whole stack onboard [INF](#).

10

Safety and security lag capability by a wide margin

Semantic safety evaluation is led by Google DeepMind. **ASIMOV** pairs an LLM-generated "robot constitution" with a semantic-safety benchmark ([Sermanet et al., arXiv 2503.08663](#)). The Gemini Robotics 1.5 report released **ASIMOV-2.0**, which covers the long tail of real injury scenarios better (drawing on NEISS data) and adds video inputs. It describes layered safety: high-level semantic reasoning about whether an action is safe, plus low-level collision avoidance ([arXiv 2510.03342](#)). Gemini Robotics 2 added **ASIMOV-Agentic** for multi-step agentic behavior ([A3/automate.org](#)) **SEC**. The researchers retrieved no ASIMOV scores this session. GRAPE shows that safety can be a post-training objective: aligning for a collision cost cut collisions by about 37–44% ([GRAPE](#)).

The security picture is worse. VLAs inherit LLM jailbreaks and add physical attack surfaces. **Textual jailbreaks** adapted from GCG, applied once at the start of a rollout, give broad control over the robot's action space ([Jones et al., arXiv 2506.03350](#); [Gray Swan](#)). **Physical and digital adversarial patches** cut task success by up to 100% in simulation ([Wang et al., ICCV 2025](#)). A model-agnostic embedding-disruption patch attack comes with a proposed defense ([arXiv 2510.13237](#)). **FreezeVLA** makes robots stop acting, with a 76.2% average attack success rate across 3 VLAs and 4 benchmarks ([arXiv 2509.19870](#)). Every architectural trend in this report widens the attack surface. Richer prompts, tool-calling orchestrators, subgoal images from generative models and cloud backbones all give an attacker another input to poison **INF**. The practical defense today is architectural, not learned: a non-learned safety controller in the lowest tier, constraints that semantic reasoning cannot override, and human confirmation for irreversible actions **INF**.

11

Capital prices the robot brain separately from the robot body

Funding is concentrated in a few companies, and foundation-model companies are now valued like hardware-heavy humanoid makers. Confidence varies a lot by row, so the table carries flags.

Company	Latest reported round / valuation	Date	Confidence
Figure AI	Series C over USD 1B at USD 39B post-money	Sept 2025	SEC ; no 2026 round found
Skild AI	Series C USD 1.4B at USD 14B+, led by SoftBank	Jan 14, 2026	SEC ; Sacra says USD 15B CONFLICT
Physical Intelligence	Series B USD 600M at USD 5.6B, led by CapitalG	Nov 2025	Confirmed
Physical Intelligence	In talks for ~USD 1B at more than USD 11B (Bloomberg)	Mar 2026	Reported (TechCrunch); a May 2026 Series C filing is UNV ; a USD 18.8B secondary mark is low-confidence SEC
Unitree	STAR Market IPO at 150.8 yuan/share, ~USD 9.0B, ~USD 904M raised	Aug 6, 2026	SEC ; other reports say USD 5.9–7B CONFLICT
Apptроник	~USD 935–966M total incl. USD 520M Series A-X, ~USD 5–5.5B	Feb 2026	SEC
Generalist AI	USD 400M Series B at USD 2B, then ~USD 200M at ~USD 3B	Jun / Aug 2026	Reported (TechCrunch, regulatory filing)
Agility Robotics	SPAC with Churchill Capital XI at USD 2.5B pre-money	2026	SEC ; close not confirmed
Dyna Robotics	USD 120M Series A	Sept 2025	Confirmed (law-firm release); one outlet says Series B at USD 600M+ CONFLICT

Sources: [Wikipedia: Physical Intelligence](#); [TechCrunch 2026-03-27](#); [Private Market View](#); [teahose: Figure](#); [Sacra: Skild](#); [KraneShares: Unitree](#); [humanoidindex](#); [TechCrunch 2026-08-25](#); [teahose: Agility](#); [Gunderson Dettmer](#); [GetCoAI](#).

Deployment lags the capital. Figure's BMW Spartanburg pilot reportedly ran about 10 months of 10-hour shifts and loaded 90,000+ sheet-metal parts that went into 30,000+ X3s. That figure is company-reported via secondary coverage, and sources disagree on whether the pilot moved from Figure 02 to Figure 03 ([iiot-world](#)) VENDOR SEC. 1X began full-scale NEO production in Hayward, California, with consumer shipments planned for 2026 ([GlobeNewswire](#)). It prices NEO at USD 20k or USD 499 per month ([The Robot Report](#)), and claims more than 10,000 first-year units sold in 5 days VENDOR SEC. As of mid-July 2026 there was no verified customer delivery, and NEO relies on scheduled remote teleoperation for tasks beyond its onboard autonomy ([theplanettools](#)) SEC. Tesla's Q4 2025 8-K plans a first Optimus production line before the end of 2026, with eventual capacity of 1M per year, and its Q2 2026 10-Q says only that it is "preparing and investing for large-scale" production ([Tesla 8-K](#); [10-Q](#)). Google DeepMind keeps the Gemini Robotics 2 VLA in early access with partners such as Apptроник. NVIDIA plays platform vendor, shipping GR00T weights, Cosmos world models and Jetson hardware.

The market structure is splitting into three layers: robot-brain companies (PI, Skild, Generalist, Dyna, DeepMind), vertically integrated humanoid makers (Figure, Tesla, 1X, Apptronik, Agility, Unitree), and platform vendors (NVIDIA, Hugging Face for open tooling) [INF](#). Valuations at the brain layer depend on the scaling claims in the data section. Those claims are mostly vendor-reported, which makes them the single largest source of valuation risk [INF](#).

12

Seven open problems define the 2027 research agenda

The first open problem is **evaluation validity**. Clean benchmarks are saturated, perturbation suites show double-digit drops, real-world numbers are self-reported, and cross-paper tables disagree by up to a point for the same model ([LIBERO-PRO](#); [LIBERO-Plus](#)). RoboArena-style pairwise evaluation and world-model evaluators are promising, but neither is yet a shared standard. The second is **long-horizon competence and failure recovery**. A best BEHAVIOR-1K Q-score of 0.26 says household-length tasks are unsolved, and recovery benchmarks like LIBERO-RECOVER are only now appearing ([arXiv 2609.05178](#)). ER 2's ability to resume from the last successful step is a vendor claim with no public numbers [SEC](#). The third is **robustness to distribution shift**, particularly visual robustness, which RL does not fix ([RL4VLA](#)). The fourth is **cross-embodiment transfer at low data**. Gemini Robotics 2's claim of fewer than 200 examples for a new body comes from a single outlet [UNV](#) [VENDOR](#), and Galaxea's finding that single-embodiment pretraining is critical cuts the other way.

The fifth is **reliable reward and world models**. World-model RL gains are small, V-JEPA 2-AC fell from 80% to 19/45 in an independent re-run, and the best progress model scores 0.655 on a cross-setup comparison [SEC](#). Until imagined rollouts can be trusted, RL stays tied to real robots and human supervisors [INF](#). The sixth is **safety and security**: jailbreaks, patches and freezing attacks all work, and defenses are early. The seventh is **the deployment gap and the economics of autonomy**. Consumer humanoids still need teleoperation, factory pilots are vendor-reported, and no independent study has measured cost per successful task against human labor [INF](#). Two structural gaps sit underneath all seven: no independently replicated scaling law at industry scale and no compute-optimal analysis for VLAs, so it is unknown whether today's capital is buying a GPT-3 moment or a plateau [INF](#).

13

What our own lab reproduces at small scale

Our [VLA lab](#) builds a complete toy stack on a 4-core CPU with no pretrained weights. It has a FiLM-CNN backbone, a VLM, a VLA with action chunking and a flow-matching head, DAgger, a Dreamer-style world model, a language-conditioned reward model and imagination-based planning. Everything is trained on a 64×64 2D tabletop with three objects and two zones. **Everything in this section is toy-scale.** The results are primary for us but show only that a mechanism *can* appear, not that it does at frontier scale. The final policy, J3, succeeds on **100.0% / 99.75% / 99.85%** of episodes in distribution, on held-out target objects and on held-out instruction phrasing (mean of two seeds, 1,000 episodes per split). Plain behavior cloning scores 62.0% / 46.5% / 64.9%. Six findings connect to the field.

The flow head needed noisy-end timestep sampling, as π_0 does. Our first flow-matching head sampled timesteps toward the clean end. It learned to denoise from the noisy action alone, ignored the observation, and reached **0.3%** success. Switching to $t = 1 - \text{Beta}(1.5, 1)$, which puts more training mass near pure noise, fixed it. With the same recipe, the flow head then beat regression at **98.3 vs 93.7** in distribution and **79.3 vs 61.7** on held-out targets (toy-scale). This is the same reason π_0 samples flow timesteps from a shifted Beta distribution that emphasizes noisier timesteps ([\$\pi_0\$](#)). At high noise the only usable information is the observation, so emphasizing that regime forces the expert to condition on it [INF](#). It also suggests why $\pi_{0.6}$ can cut integration steps from 10 to 5 [SEC](#): if the expert is well conditioned at the noisy end, most of the trajectory is easy [INF](#).

Fine-tuning a pretrained backbone at full learning rate destroyed the policy, which is the problem knowledge insulation solves. Fine-tuning our small VLM backbone at the action head's learning rate collapsed the policy to 3.7%. At 0.2x it recovered to 86.0% but still lost to a from-scratch backbone with auxiliary grounding (89.7%), and lost worst on paraphrases (73.7 vs 87.3). This is the toy version of the failure that knowledge insulation and $\pi_{0.6}$'s stop-gradient prevent ([KI](#); [\$\pi_{0.6}\$](#)) [INF](#). The lab also adds a caveat the field mostly avoids: our VLM's language covered *questions*, not *instructions*, so pretraining only pays when the backbone's language and visual distribution cover the downstream task. Web-scale VLMs mostly satisfy that condition. Small in-domain VLMs do not.

Auxiliary grounding beat every policy-side trick, which parallels ECoT, spatial grounding and Gemini ER. Regressing the target and goal positions from the summary token added about **30 points on every split** (60.0 → 89.7 in distribution). Action chunking alone added under 2 points, within noise. Failure analysis showed why: 33.5% of baseline episodes never grasped anything, so the bottleneck was localization, not control or language (toy-scale). This matches the field's move toward explicit spatial intermediates such as ECoT's bounding boxes and gripper positions, InternVLA-M1's spatially guided stage and Gemini Robotics-ER's pointing ([ECoT](#); [InternVLA-M1](#); [Gemini Robotics](#)) [INF](#). Round 2 pushed the point further. A compositional heat map that was **100% correct on held-out targets** still left the policy at 62–66%, because the policy took a shortcut through the language-conditioned trunk. Removing language from the visual trunk, so object identity could enter only through factorized target and goal heat maps, took held-out success to **100%** (J2). Adding verb-dropping syntax augmentation fixed the phrasing split (J3). The field's equivalent is a hierarchy that sends grounding through an inspectable interface, such as ER points, $\pi_{0.7}$ subgoal images or $\pi_{0.5}$ subtask text, rather than trusting one monolithic trunk to compose [INF](#). One caveat: our J2 bakes in a (color, shape) prior and simulator labels, which real systems have to replace with open-vocabulary detectors.

Dagger with an omniscient expert is not RECAP, and the difference shows. With a regression head, each DAgger round traded held-out generalization for in-distribution polish (held-out targets 77.0 $\rightarrow$ 61.7 while in-distribution rose 89.7 $\rightarrow$ 93.7). Every relabeled state used a training-distribution instruction, so the data drifted toward what the model had already seen. With the flow head, held-out success did not erode (72.7 $\rightarrow$ 79.3) (toy-scale). RECAP differs from our DAgger in three ways that matter. Its corrections are sparse and human-initiated rather than dense and scripted. It keeps the robot's own autonomous experience instead of replacing labels. And a learned value function decides which data to imitate ($\pi^*0.6$). Our result predicts that dense relabeling narrows a policy unless the head models the full action distribution, and that matches the field's preference for generative heads plus advantage weighting over plain aggregation [INF](#). HIL-SERL's comparison, in which the HG-Dagger baseline averaged 49.7% success against near-perfect RL, points the same way at real scale ([HIL-SERL](#)).

Imagination planning helps only with on-manifold proposals, as with test-time verifiers. At each re-plan, our planner imagines 4 steps for each of 8 candidate chunks with the world model, scores the final frames with the reward model, and executes the best. Argmax over *perturbed* regression actions exploited model errors, worst out of distribution (held-out targets 77.0 $\rightarrow$ 53.7). A trust region removed the damage but added nothing. Proposals sampled from the flow policy itself stayed on the data manifold and gave small, consistent, statistically non-significant gains (+0.4 / +2.3 / +2.0 points) (toy-scale). This mirrors the RoboMonkey story closely. Sample-and-verify helps weak policies, and it can do worse than the base model when the verifier is miscalibrated on the candidates it sees ([RoboMonkey](#); [arXiv 2602.12281](#)) [INF](#). Our bottleneck was the reward model on *imagined* frames: success AUC fell from **0.75 on real frames to 0.59 on imagined ones**, because the deterministic world model blurs exactly the moment that matters, a small object released under the gripper. That is the same fidelity-and-verification bottleneck that keeps world-model RL gains like VLA-RFT's +4.5 points small ([VLA-RFT](#)) [INF](#), and it explains why trajectory-trained reward models like Robometer beat prompted ones [INF](#).

What the lab does not show. It has one embodiment, 2D dynamics with attachment-only physics, a scripted expert, 64×64 renders and about 0.9M parameters. It cannot speak to scaling laws, cross-embodiment transfer, latency, real-world RL or safety. Its value is as a fast test bench for mechanisms (timestep schedules, insulation, grounding interfaces, proposal distributions for planning) before spending GPU hours on them.

14

Conclusion

The VLA field has settled the architecture question faster than anyone expected and moved the fight to data and feedback. The real innovations of 2026 are not new heads. They are new *interfaces*: advantage and quality tokens that let a model learn from imperfect experience, subgoal images that let a world model steer a policy, and a three-tier split that lets each part of the stack run at its natural rate. The consequence is that VLA progress now looks like the LLM post-training era. The organizations with fleets, intervention pipelines and reward models compound their lead, while the open community competes on efficiency, evaluation and recipes. Our predictions for the next 12 months, with roughly stated confidence, follow. At least one more frontier lab will publish a RECAP-style real-world RL result with multi-hour autonomous runs (high). An open-weight model of 3–5B with an RL post-training recipe will match $\pi 0.5$ on real-world household benchmarks (medium-high). RoboArena-style pairwise evaluation or world-model evaluators will replace LIBERO as the headline metric in major papers (medium). Human egocentric and wearable data will make up most of the pretraining hours in every new frontier corpus (high). An independently replicated scaling law at 100k+ hours will still be missing by October 2027 (medium). Consumer humanoids will ship in small numbers with substantial teleoperation behind them (high).

For a practitioner building a VLA today, the recommendations are concrete. **Start from an open, insulated backbone and a flow expert:** openpi's $\pi 0/\pi 0.5$ or GR00T N1.7 if you have GPUs, SmolVLA if you do not. Freeze or stop-gradient the backbone, sample flow timesteps toward the noisy end, and run with asynchronous or real-time-chunked execution from day one. **Collect for diversity, not volume.** Many environments and objects, a cheap UMI or GELLO rig, egocentric human video as the base, and VQA co-training to protect open-vocabulary grounding. **Build the intervention loop before the model is good.** Log every teleop correction and autonomous rollout, train a progress model on trajectory comparisons, and use advantage-conditioned or weighted BC before trying on-policy RL. **Give grounding an explicit, inspectable interface** (points, boxes, subtask text or subgoal images) rather than hoping the trunk composes. **Treat test-time sampling and world-model planning as optional extras:** use them only with on-policy proposals and a calibrated verifier, and measure on held-out conditions where they tend to hurt. **Evaluate on perturbations and pairwise real-world comparisons,** never on clean LIBERO alone. And **put a non-learned safety layer at the bottom of the stack,** because every published attack on VLAs works.

15

Sources

Models and architectures: [RT-1](#) · [RT-2](#) · [RT-2 blog](#) · [Octo](#) · [OpenVLA](#) · [OpenVLA-OFT](#) · [\$\pi 0\$](#) · [FAST](#) · [\$\pi 0.5\$](#) · [Knowledge Insulation](#) · [\$\pi^*0.6\$](#) · [OpenTau \$\pi 0.6\$ docs](#) · [\$\pi 0.7\$](#) · [PI \$\pi 0.7\$ blog](#) · [Terry Um \$\pi 0.7\$ summary](#) · [hyper.ai \$\pi 0.7\$](#) · [swiftscholar \$\pi 0.7\$](#) · [oodaloop/TechCrunch \$\pi 0.7\$](#) · [The Decoder \$\pi 0.7\$](#) · [GR00T N1](#) · [GR00T N1.6 model card](#) · [CSDN on N1.6 backbone](#) · [GR00T N1.7 mirror](#) · [agentmarketcap N1.7](#) · [Gemini Robotics](#) · [Gemini Robotics On-Device](#) · [Gemini Robotics 1.5](#) · [On-Device 2 model card](#) · [Gemini Robotics 2 coverage](#) · [MarkTechPost GR2](#) · [Scalewise GR2](#) · [Figure Helix](#) · [Helix 02](#) · [The Humanoid Hub](#) · [SmolVLA](#) · [SmolVLA blog](#) · [X-VLA](#) · [Galaxea G0](#) · [Galaxea G0.5](#) · [InternVLA-M1](#) · [RDT-1B](#) · [CogACT](#) · [DexVLA](#) · [HybridVLA](#) · [AgiBot GO-1 / World](#) · [ECoT](#) · [LAPA](#) · [UniVLA](#) · [SpatialVLA](#)

Data and scaling: [Open X-Embodiment](#) · [OXE blog](#) · [DROID](#) · [DROID summary](#) · [BridgeData V2](#) · [AgiBot World GitHub](#) · [GalaxeaVLA](#) · [LeRobot community v3](#) · [Xiaomi-Robotics-1](#) · [GEN-0](#) · [The Batch on GEN-0](#) · [36Kr on GEN-0](#) · [aibase on GEN-0](#) · [ReMix](#) · [EgoDex](#) · [PH2D / HAT](#) · [Data Pyramid survey](#) · [Not All Actions Are Equal](#) · [GR00T whitepaper](#) · [dexset.ai teleop comparison](#) · [dexset.ai 2026 guide](#) · [U-ARM](#) · [DreamGen](#) · [GR00T N1.5](#) · [\$\pi 0.5\$ PDF](#) · [robo2u](#) · [ay-robots](#) · [Lin et al. scaling laws](#) · [awesome-physical-ai](#)

RL, world models, reward models and test-time compute: [\$\pi^*0.6\$ HTML](#) · [\$\pi^*0.6\$ PDF](#) · [postcutoff on \$\pi^*0.6\$](#) · [sudoremove on \$\pi^*0.6\$](#) · [SimpleVLA-RL](#) · [RIPT-VLA](#) · [RL4VLA](#) · [\$\pi\$ RL](#) · [RLinf docs](#) · [RLinf-VLA](#) · [GRAPE](#) · [HIL-SERL](#) · [ConRFT](#) · [DSRL](#) · [DSRL summary](#) · [VLA-RFT](#) · [World-Env](#) · [V-JEPA 2](#) · [AD-WM](#) · [1X World Model](#) · [WorldEval](#) · [dWorldEval](#) · [Robometer](#) · [RoboReward](#) · [RynnValue](#) · [RoboMonkey](#) · [RoboMonkey PMLR](#) · [RoboMonkey failure report](#) · [V-GPS](#) · [\$\tau 0\$ -VLA](#)

Benchmarks, deployment and safety: [StarVLA](#) · [PhysBrain 1.0](#) · [OA-WAM](#) · [MemoryVLA](#) · [LIBERO-PRO](#) · [LIBERO-Plus](#) · [IndustrialVLA-Bench](#) · [LIBERO-RECOVER](#) · [BEHAVIOR 2025](#) · [BEHAVIOR leaderboard](#) · [benchmarklist BEHAVIOR](#) · [BEHAVIOR 2026](#) · [RoboArena](#) · [PI Real-Time Chunking](#) · [RTC paper](#) · [PI RTC training-time](#) · [Async inference analysis](#) · [Jetson-PI](#) · [vla.cpp](#) · [ASIMOV](#) · [A3 on GR2 safety](#) · [VLA jailbreaks](#) · [Gray Swan](#) · [Adversarial patches](#) · [Embedding Disruption Patch](#) · [FreezeVLA](#)

Industry: [Wikipedia: Physical Intelligence](#) · [revenuememo](#) · [TechCrunch on PI](#) · [Private Market View](#) · [teahose: Figure](#) · [iiot-world BMW pilot](#) · [Sacra: Skild](#) · [TechCrunch: Skild 2025](#) · [TechCrunch: Generalist](#) · [Gunderson: Dyna](#) · [Verdict: Dyna](#) · [GetCoAI: Dyna](#) · [1X factory](#) · [1X pricing](#) · [1X teleop caveat](#) · [Tesla 8-K](#) · [Tesla 10-Q](#) · [KraneShares: Unitree](#) · [humanoidindex](#) · [teahose: Agility](#)

Our lab: [VLA lab README](#) (toy-scale, CPU-only, primary for this repository)