Towards Generalist Game Players: A Survey of Foundation Models in the Game Multiverse

Anonymous authors
Paper under double-blind review

Abstract

The real world unfolds along a single set of physics laws, yet human intelligence demonstrates a remarkable capacity to generalize experiences from this singular physical existence into a multiverse of games, each governed by entirely different rules, aesthetics, physics, and objectives. This omni-reality adaptability is a hallmark of general intelligence. As Artificial Intelligence progresses towards Artificial General Intelligence, the multiverse of games has evolved from mere entertainment into the ultimate ground for training and evaluating AGI. The pursuit of this generality has unfolded across four eras: from environment-specific symbolic and reinforcement learning agents, to current large foundation models acting as generalist players, and toward a future creator stage where the agent both creates new game worlds and continually evolves within them. In this survey, we trace the full lifecycle of a generalist game player along four interdependent pillars: Dataset, Model, Harness, and Benchmark. Every advance across these pillars can be read as an attempt to break one of five fundamental trade-offs that currently bound the whole system. Building on this end-to-end review, we chart a five-level roadmap, progressing from single-game mastery to the ultimate creator stage in which the agent simultaneously creates and evolves within the theoretical game multiverse. Taken together, this survey offers a unified lens onto a rapidly shifting field, and a principled path toward the omnipotent generalist agent capable of seamlessly mastering any challenge within the multiverse of games, thereby paving the way for AGI.

1 Introduction: From a Single Worldline to the Multiverse of Games

> *"Man only plays when in the full meaning of the word he is a man, and he is only completely a man when he plays."*
>
> — **Friedrich Schiller**

Games are far from mere pastimes; they are profound abstractions of reality—originating from life, yet transcending it. Collectively, they constitute a vast ***multiverse***, where each universe is governed by entirely distinct rules, aesthetics, physics and objectives, different from the single reality.

From a cognitive science perspective, human intelligence was developed to navigate the survival constraints of a single physical reality—a worldline characterized by consistent gravity, linear time, shared environment and fixed material properties (Cosmides & Tooby, 1994; Gigerenzer & Goldstein, 1996). However, armed with the singular set of real-world experiences, humans can seamlessly adapt to the infinite ***multiverse of games*** (Dubey et al., 2018; Lake et al., 2017), from managing block-based ecosystems in Minecraft (Fan et al., 2022) to orchestrating grand strategies in StarCraft (Ma et al., 2024) and Civilization (Qi et al., 2024). Within minutes, a human can adapt to a digital universe where the rules are different, the aesthetics surreal, the physics inverted, and the objectives entirely abstracted from physical survival. The ***omni-reality adaptability*** to utilize localized experience to rapidly master a multiverse of entirely novel environments is the ultimate symbol of generalized intelligence (Lake et al., 2017; Legg & Hutter, 2007; Turing, 1953).

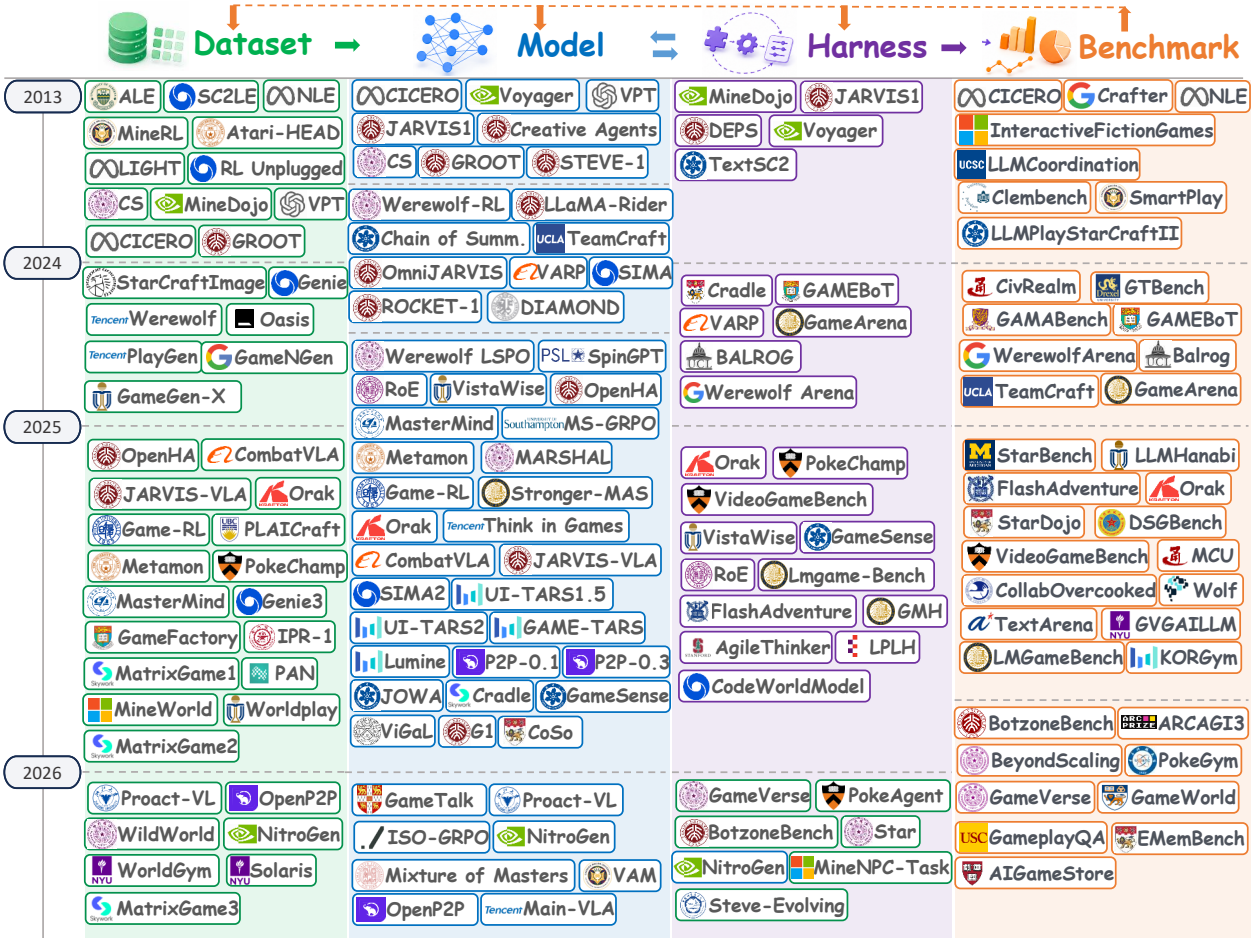

Figure 1: **A holistic overview of the research landscape toward generalist game players.** It organizes representative works along a temporal axis and unifies them into four key pillars: **Dataset, Model, Harness, and Benchmark**. By illustrating the evolution and interplay of these components over time, it reveals the field's transition from isolated, single-game studies to a unified paradigm of cross-game, multimodal generalist agents.

Driven by the pursuit of this generalized intelligence, researchers naturally turned to these digital games as the ultimate testbed for AI since its inception (Campbell et al., 2002; Turing, 1953; Laird & Lent, 2001). This coupling between games and the pursuit of general intelligence is a premise the community has shared for decades: from Turing's early treatment of chess as a probe of machine intelligence (Turing, 1953), through Deep Blue (Campbell et al., 2002), AlphaGo (Silver, 2016), AlphaStar (Vinyals, 2019), and OpenAI Five (OpenAI et al., 2019), to today's foundation-model agents such as Voyager (Wang et al., 2024a), Game-TARS (Wang et al., 2025f), and Lumine (Tan et al., 2025b). Digital worlds offer high quality, dynamic, visually rich, and long-horizon environments that demand a synthesis of perception, planning, reasoning, and motor control, while AI pushes the boundaries of what can be automated and solved within these worlds. For decades, the AI community has utilized games as a crucible to train and measure capability (Bellemare et al., 2013; Johnson et al., 2016; Fan et al., 2022; Towers et al., 2024; Ahn et al., 2025; Hu et al., 2025a; Park et al., 2026; Zhang et al., 2026a; Ying et al., 2026; Campbell et al., 2002; Silver, 2016; Vinyals, 2019; OpenAI et al., 2019; Wang et al., 2025f), yet it has historically failed to replicate this omni-reality adaptability. Generally, past paradigms have largely produced ***specialists***. Despite achieving superhuman performance through environment-specific Symbolic and Reinforcement Learning (RL) in certain complex games like DeepBlue in Chess (Campbell et al., 2002), AlphaGo in Go (Silver, 2016), AlphaStar in StarCraft II (Vinyals, 2019) and Openai Five in Dota II (OpenAI et al., 2019), these systems were fundamentally brittle specialists.

Table 1: Comparison with representative surveys on foundation models and games. ✓: covered; ∼: partial; ×: not covered.

Survey	Model families	Dataset	Model	Harness	Benchmark	Trade-offs	Roadmap
Gallotta et al. (2024)	LLM	×	✓	∼	∼	×	✓
Hu et al. (2024)	LLM	×	✓	✓	∼	×	∼
Ours	LLM / VLM / VLA / WM	✓	✓	✓	✓	✓ (5)	✓ (5-level)

Typically, these models are optimized from scratch for a singular game worldline with simplified interfaces and interactions. While they achieve superhuman performance within such highly structured environments, they fail completely to transfer their knowledge to another game worldline—rendering them paradoxically powerful yet profoundly fragile. Moreover, these models almost operate as black boxes, heavily obscuring the underlying mechanisms that drive their decision-making processes (Greydanus et al., 2018; Mott et al., 2019). Ultimately, systems confined to this paradigm fall fundamentally short of the true AGI that AI community pursues.

Recently, the advent of Large Foundation Models (LFMs), including Large Language Models (LLMs) (Brown et al., 2020; Ouyang et al., 2022; Bai et al., 2023; Yang et al., 2025), Vision-Language Models (VLMs) (OpenAI et al., 2024; Comanici et al., 2025; Bai et al., 2025a; SIMA-team et al., 2025), Vision-Language-Action Models (VLAs) (Zitkovich et al., 2023; Chen et al., 2025c; Li et al., 2025a; Wang et al., 2025f), and World Models (WMs) (Ha & Schmidhuber, 2018; Decart & Julian Quevedo, 2024; Zhang et al., 2025d; He et al., 2025a; Ball et al., 2025), has sparked a transformative paradigm shift. Rather than mastering a single game through billions of trial-and-error episodes from scratch, LFMs were born with vast open-world knowledge and emergent reasoning capabilities. By treating games not as isolated optimization problems, but as diverse instances of interactive environments, these models show the potential of ***omni-reality adaptability*** as a true ***generalist*** game player.

To capture this rapid evolution, this survey provides a comprehensive pipeline-oriented perspective towards generalist game player. ***To the best of our knowledge, this is the first survey to systematically investigate Large Foundation Models (LFMs) as generalist game players through a comprehensive, end-to-end lifecycle.*** Unlike prior surveys that cover a single slice—e.g., LLMs and games broadly (Gallotta et al., 2024) or LLM-based game agents (Hu et al., 2024)—we couple all four pillars into one loop with the widest model coverage (LLM/VLM/VLA/WM), five fundamental trade-offs, and a five-level roadmap, as summarized in Table 1. ***Our contributions are threefold:***

- **An evolution framework for game-playing AI.** We trace the development of game-playing AI through four eras and unify them under a Goal-Conditioned POMDP formulation. This formulation makes explicit how each transition reshapes the elements of the POMDPs tuple $\mathcal{M}$, providing a principled basis for understanding the ongoing shift from the brittle specialists to generalist game players can be understood.
- **A four-pillar pipeline for generalist game players.** We organize the literature around four interdependent pillars, **datasets, models, harness, and benchmarks**, that together constitute the full lifecycle of a generalist game-playing system. Within this pipeline, datasets fuel model training, trained models are deployed by the harness to interact with game environments, benchmarks measure the resulting capabilities and expose limitations, and these findings motivate the collection of new data and further model improvement.
- **A research roadmap with identified open challenges.** Building on the systematic review, we identify current bottleneck in each pillar. We highlight where current methods cluster and where significant gaps remain, aiming to inform future research priorities in this rapidly evolving field.

Scope of evidence. In this fast-moving area, much of the frontier is released as technical reports, preprints, or blog posts rather than peer-reviewed papers; we prioritize widely recognized, high-impact works and report their numbers under the original conditions (model, interface, metric), treating them as indicative of capabilities and failure modes rather than as settled findings.

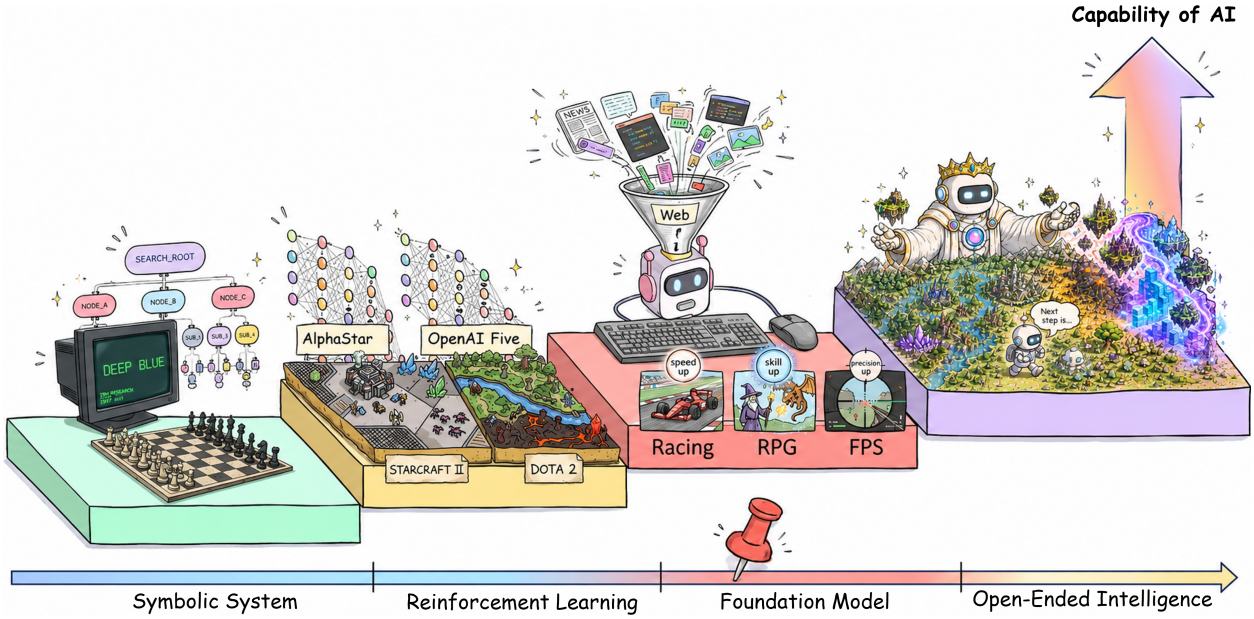

Figure 2: **The evolution of game-playing AI paradigms towards AGI.** We categorizes the timeline into four eras: (1) Symbolic Systems, relying on hard-coded heuristics within isolated environments; (2) Deep Reinforcement Learning (DRL), achieving superhuman mastery in specific domains, yet constrained as narrow experts; (3) Large Foundation Models (LFMs), emerging as generalist agents capable of reasoning and adaptation across the human-crafted multiverse of games; and ultimately, (4) The Creator, the future where AI transcends playing to become the simulator itself, autonomously generating and evolving infinite game multiverses.

2 Preliminary: AI, Games, and the Paradigm Shift

2.1 The Symbiosis and Formalization of AI in Games

The quest for AI has always been closely linked to games. From the early days of chess programs (Campbell et al., 2002) to modern highly complex strategy games (Ma et al., 2024), games serve as a standardized, quantifiable proxy for real-world decision-making. They include diverse challenges, such as partial observability, long-horizon planning, and real-time multimodal processing, while without the physical risks and costs of real-world robotics.

To establish a rigorous foundation, we formalize the interaction between an AI agent and a game environment as a Partially Observable Markov Decision Processes (POMDPs) (Smallwood & Sondik, 1971). A game can be described as a tuple $\mathcal{M} = \langle G, S, A, T, R, \Omega, O, \gamma \rangle$, where:

- G is a set of potential goals or objectives. Each $g \in G$ is a task objective (e.g., natural language prompts, target images, or specific sub-tasks) that dictate the agent's current mission.
- S is a set of states. Each $s \in S$ represents the internal state of the environment.
- A is a set of actions. Each action $a \in A$ can be a combination of textual reasoning, retrieval of external knowledge and memory, tool calls, and execution action.
- $T : S \times A \to \Delta(S)$ is the state transition probability function which takes a state-action pair (s, a) and outputs the probability distribution $T(s' \mid s, a)$ of the next state.
- $R : S \times A \times G \to \mathbb{R}$ is the feedback/reward function, conditioned on the specific goal $g \in G$. The feedback $r = R(s, a, g)$ typically takes the form of a scalar score or textual or image feedback.
- Ω is a set of observations accessible to the agent.

- $O : S \times A \to \Delta(\Omega)$ is the observation probability function which takes a state-action pair (s, a) and outputs the probability distribution $O(o' \mid s, a)$ of the next observation for the agent.
- $\gamma \in [0, 1)$ is the discount factor.

Under this framework, by explicitly introducing G, the agent aims to find a goal-conditioned policy $\pi(a_t|o_{\leq t}, g)$ that maximizes the expected discounted return $E[\sum_{t=0}^{\infty} \gamma^t R(s_t, a_t, g)]$.

2.2 Operationalizing the Generalist Game Player

The tuple $\mathcal{M}$ lets us define the generalist target precisely, rather than by genre or visual style. Let $\mathfrak{M}$ denote a *multiverse*: a space of games $\mathcal{M} = \langle G, S, A, T, R, \Omega, O, \gamma \rangle$, equipped with a distribution $\mathcal{D}$ over it. For a goal-conditioned policy π and a goal $g \in G$, we write the expected return on game $\mathcal{M}$ as

$$V_{\mathcal{M}}(\pi, g) \;=\; \mathbb{E}\Big[\textstyle\sum_{t=0}^{\infty} \gamma^t R(s_t, a_t, g) \;\Big|\; \pi, \mathcal{M}\Big]. \tag{1}$$

To make scores comparable across heterogeneous games, we normalize against a random policy π^{rand} and a reference expert (or human) policy $\pi^{\star}_{\mathcal{M}}$:

$$\mathcal{C}_{\mathcal{M}}(\pi) \;=\; \mathbb{E}_{g \sim G}\left[\frac{V_{\mathcal{M}}(\pi, g) - V_{\mathcal{M}}(\pi^{\text{rand}}, g)}{V_{\mathcal{M}}(\pi^{\star}_{\mathcal{M}}, g) - V_{\mathcal{M}}(\pi^{\text{rand}}, g)}\right] \;\in\; (-\infty, 1], \tag{2}$$

so that $\mathcal{C}_{\mathcal{M}}(\pi) = 1$ matches expert competence and $\mathcal{C}_{\mathcal{M}}(\pi) = 0$ matches random play.

Definition 1 ((ϵ, δ)-Generalist Game Player)**.** *A single policy π_θ with* fixed *parameters θ is an (ϵ, δ)-generalist over $(\mathfrak{M}, \mathcal{D})$ if, without any per-game parameter update,*

$$\Pr_{\mathcal{M} \sim \mathcal{D}}\left[\mathcal{C}_{\mathcal{M}}(\pi_\theta) \geq 1 - \epsilon\right] \;\geq\; 1 - \delta. \tag{3}$$

The defining constraint is that the same θ must serve every $\mathcal{M} \sim \mathcal{D}$: a specialist re-optimized per game does not qualify. What makes $(\mathfrak{M}, \mathcal{D})$ demanding is how its games differ. We call two games *structurally distinct* when they differ in observation channel (Ω, O), action semantics (A), transition dynamics (T), goal and reward specification (G, R), temporal scale, or required world prior; these are the axes induced by $\mathcal{M}$, and we use them—rather than surface appearance—to measure the diversity of $\mathfrak{M}$.

This distinction separates two claims. Let $\mathfrak{M}_{\text{tr}}$ be the games seen in training. *Within-family* generalization holds when test games share (Ω, A) with $\mathfrak{M}_{\text{tr}}$ up to relabeling and differ only in surface content. *Cross-world* generalization requires competence on games that differ from every $\mathcal{M} \in \mathfrak{M}_{\text{tr}}$ along at least one of A, O, or T. We quantify the shortfall by the generalization gap

$$\Delta(\pi_\theta) \;=\; \mathbb{E}_{\mathcal{M} \sim \mathcal{D}_{\text{tr}}}\big[\mathcal{C}_{\mathcal{M}}(\pi_\theta)\big] \;-\; \mathbb{E}_{\mathcal{M} \sim \mathcal{D}_{\text{test}}}\big[\mathcal{C}_{\mathcal{M}}(\pi_\theta)\big], \tag{4}$$

and a genuine generalist keeps Δ small even when $\mathcal{D}_{\text{test}}$ is cross-world. Our five-level roadmap (Section 7.2) instantiates this hierarchy: Level 2 varies surface content at fixed (Ω, A), Level 3 varies A, O, and T jointly, and Level 4 requires online inference of an unknown $\mathcal{M}$.

2.3 The Evolution of Game-Playing AI Paradigms

The evolution of game-playing AI can be mapped to how agents interact with, and control the elements of the POMDPs tuple $\mathcal{M}$. This progression marks a fundamental shift from rigid rule execution in constrained state spaces to open-ended multiverse simulation. Table 2 visualizes this paradigm shift across four distinct eras:

We assign a system to an era by which components of $\mathcal{M}$ it treats as fixed versus agent-controlled (Table 2), not by publication date; the eras are analytical stages rather than disjoint periods, and a system may combine features of adjacent eras.

Table 2: The Paradigm Shift in Game-Playing AI Formalization. We frame the evolution through the lens of a Goal-Conditioned POMDP $\mathcal{M} = \langle G, S, A, T, R, \Omega, O, \gamma \rangle$, highlighting the transition from rigid, pre-defined components to agent-generated open-ended dynamics.

Feature	**Era 1: Symbolic**	**Era 2: Deep RL**	**Era 3: Foundation Models**	**Era 4: The Creator, Demiurge**
Timeline	Pre-2010	2010-2022	2022-Present	Post-Future
Scope	Single Universe	Single Universe	Human-crafted Multiverse	Theoretical Multiverse
Mechanism	Rules $\rightarrow$ Search	State/Pixels $\rightarrow$ Policy	Knowledge $\rightarrow$ Reason $\rightarrow$ Action	Simulate $\rightarrow$ Generate $\rightarrow$ Evolve
Goal (G)	Fixed Condition	Maximize Reward	Dynamic Tasks	Self-assigned & Evolving
Action (A)	Symbolic API	Structured API	Human-like Interface	**Unconstrained / Generative**
Dynamics (T)	Deterministic	Fixed Black-box	Fixed Black-box	**Dynamically Generated**
Reward (R)	Terminal / Heuristic	Env-defined Scalar	Env Scalar + Semantic Evaluation	**Agent-generated**
Obs. (O, Ω)	Bypassed (Direct S)	Pixels / Structured ($\Omega_{px/vec}$)	Multimodal ($\Omega_{text,vision,audio}$)	**Omniscient / Bird's-eye**
World Prior	Hand-craft Logic	Tabula Rasa	Pre-trained Knowledge	Self-defined Logic
Target	Specialist	Specialist	**Generalist Player**	**Omnipotent Creator**

Era 1: Symbolic Systems - *Know the Rules, Search the Trees.* Prior to 2000, early AI relied on the hard-coded rules and heuristic search algorithms (Knuth & Moore, 1975; Campbell et al., 2002). In this era, the observation function O was often bypassed, giving the agent direct access to a fully observable and simplified state space S (e.g., the exact positions of the chess pieces). The action space A was rigidly defined by game-specific logical rules, and the goal G was a singular, fixed terminal state, such as checkmate in chess. These systems required domain experts to hand-craft features and lacked the ability of true perception and generalization.

Era 2: Deep Reinforcement Learning - *Learn from Scratch via Self-Play.* Since 2010s, the integration of deep neural networks with RL shifted the paradigm toward handling partial observability (Silver, 2016; Vinyals, 2019; OpenAI et al., 2019). Agents learned policies $\pi(a_t|o_{\leq t})$ from high-dimensional observation spaces Ω (such as raw pixels or structured input). However, the action space A remained a fixed, structured API as pre-defined control vectors in the specific task, and the goal G was implicitly bound to a static, environment-provided scalar reward R. While eliminating hand-crafted rules, these models still learned from scratch for each game $\mathcal{M}$, producing highly capable but narrow specialists with no omni-reality adaptability.

Era 3: Foundation Models - *Understand, Reason, Then Act.* Now, the current era is defined by models pre-trained on internet-scale multimodal data (Wang et al., 2025f; Tan et al., 2025b; Wang et al., 2025c; Yue et al., 2026; Magne et al., 2026). Here, the observation space Ω expands to include rich multimodal inputs (vision, audio, text). Crucially, the goal space G becomes open-ended, allowing agents to pursue complex, dynamic natural language instructions. Furthermore, the action space A shifts from game-specific APIs to a universal, human-like interface (emulating keyboard and mouse actions). By leveraging pre-trained world knowledge to infer underlying transition dynamics T, these models transition the field toward generalist game player in hand-crafted multiverse of games.

Era 4: The Creator, Demiurge - *Simulate, Generate, Then Evolve.* We use Era 4 (the Demiurge) as a forward-looking framing device marking the endpoint of the $\mathcal{M}$-control trajectory in Table 2, not a claim of imminent capability; world models (Ball et al., 2025; Decart & Julian Quevedo, 2024) and embodied agents are early, active steps in this direction. In the future, the AI transcends the role of a participant constrained by a fixed game $\mathcal{M}$ (SIMA-team et al., 2025; Ball et al., 2025). Instead of merely optimizing a policy within predefined boundaries, the agent becomes the simulator itself. It possesses the capability to actively generate and expand the state space S and action space A dynamically. It invents its own evolutionary goals G, and constructs the transition dynamics T and reward structures R. The interface becomes completely free-form, shifting from existing games to simulating, evolving, and directing the progression of entire theoretical multiverses.

2.4 The Current Frontier: The Nexus of Era 2 and Era 3

Situating the current landscape within the evolutionary trajectory of game-playing AI, we find ourselves at a critical stage: transitioning from the specialized optimization of Era 2 to the generalist paradigms of Era 3. While Deep RL in Era 2 has yielded specialists capable of mastering complex POMDPs, these systems often remain confined by the generalization bottleneck and the cost of learning π from scratch for each game $\mathcal{M}$ is becoming increasingly prohibitive.

Consequently, the field is pivoting toward Era 3: Foundation Model as generalists. By leveraging the world knowledge priors of Large Foundation Models, emerging frameworks demonstrate remarkable multimodal integration, zero-shot adaptation and goal-conditioned reasoning. However, significant challenges remain across the board, spanning accurate perception, sparse reasoning, memory retrieval, precise control, and efficiency, affecting virtually every layer of the system.

This survey specifically focuses on this paradigm-shifting frontier. We comprehensively review the ongoing transition, exploring the datasets, models, harness, benchmarks, and persistent challenges that define this pivotal moment in AI evolution, ultimately laying the groundwork for the distant but inevitable realization of Era 4.

3 Datasets: The Foundation Beneath Generalist Game Player

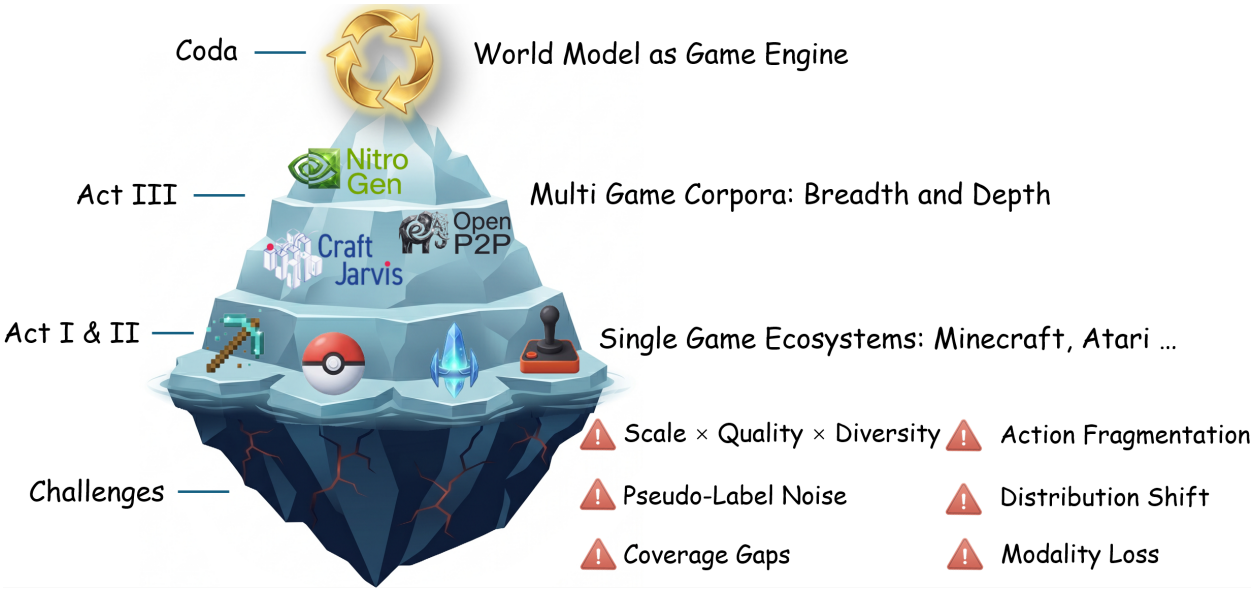

Figure 3: **Overview of Dataset in Game-playing AI.** The trajectory of data curation is shifting from isolated, single-game ecosystems (Acts I & II) to comprehensive, multi-game corpora (Act III) in both depth and breadth, tailored for generalist agents. We highlight the future where world model can replace the game engine, and we show the critical bottlenecks in current dataset construction.

In terms of the POMDP tuple $\mathcal{M}$, a dataset is a finite sample of interaction trajectories over observations Ω and actions A, optionally labeled with goals G or rewards R. Among the four fundamental pillars of our pipeline, datasets occupy a uniquely structural position. They serve not merely as the starting point of the development loop but rather as the constraint surface that shapes all downstream processes. The model architectures a community can explore, the harness designs that prove viable, and the benchmark dimensions that remain measurable are mostly predetermined, frequently in an implicit manner, by the data available at the onset of training.

In the DeepRL Era 2 paradigm, agents (Silver, 2016; Vinyals, 2019; OpenAI et al., 2019) generated their own training data through direct environment interaction: each self-play loop was self-contained, the data it

Table 3: Summary of representative large datasets in gaming AI

Paper	Modality	Data Source	Label Method	Scale	Game Set
ALE (2013)	Frame + action	Atari 2600 ROMs	Env-provided reward	57 games	57 Atari games
SC2LE (2017)	State + actions	Ladder replays	Auto-recorded	∼800K replays	StarCraft II
LIGHT (2019)	Text dialogues	Crowdsourced roleplay	Human-authored	663 scenes, 11K episodes	Text adventure
MineRL (2019)	RGB + actions	Crowdsourced play	Auto-recorded	60M frame-action pairs	Minecraft
Atari-HEAD (2020)	Frame + gaze + action	Human play + eye-tracker	Sync-recorded	117 hrs, 328M gaze	20 Atari games
NLE (2020)	Symbol + action	Procedural generation	Env-provided reward	Infinite (procedural)	NetHack
RL Unplugged (2020)	Frame + action	RL replay buffers	Auto-recorded	46 games × 200M frames	46 Atari games
CS Deathmatch (2022)	RGB + kbd/mouse	Server spectation	Rule-based IDM	5.77M frames, 100 hrs	CS:GO
VPT (2022)	Video + actions	Web videos	IDM pseudo-label	70K hrs (clean)	Minecraft
MineDojo (2022)	Video + text	YouTube/Wiki/Reddit	None (natural)	730K videos, 300K hrs	Minecraft
CICERO (2022)	Text + board state	webDiplomacy.net	Auto-aligned	40K+ human games	Diplomacy
StarCraftImage (2023)	Rendered images	Match replays	Auto-generated	3.6M imgs / 60K replays	StarCraft II
GROOT (2024)	Video + actions	Contractor + YouTube	IDM pseudo-label	30 tasks (benchmark)	Minecraft
Werewolf-18K (2024a)	Text dialogues	Human play	Auto-recorded	18.8K sessions	Werewolf
Genie (2024)	Unlabeled video	Web videos	None (unsupervised)	30K hrs (filtered)	100+ platformers
GameGen-X (2025)	Video + captions	Web game videos	GPT-4o captioning	1M+ clips, 150+ games	150+ games
GameNGen (2024)	Frame + action	RL-agent gameplay	Auto-recorded	20 FPS, multi-minute play	DOOM
PlayGen (2024)	Frame + action	Agent-generated gameplay	Auto-recorded	20 FPS, 1K+ Gen. frames	Mario & DOOM
OpenHA (2025e)	RGB + actions	VPT trajectories	Rule-based pipeline	5.5B tokens, 1K tasks	Minecraft
JARVIS-VLA (2025b)	Image + actions	Scripted agent	Self-supervised tokenizer	3.78M samples, 1K tasks	Minecraft
PLAICraft (2025b)	5-modal aligned	Human play	None (time-aligned)	10K hrs, 10K players	Minecraft
Game-RL (2025)	Image + QA	Code generation	Code-verified	140K QA pairs	30 games
Metamon (2025)	RL trajectories	Platform replays	Auto-parsed	2M human + 18M self-play	Pokémon
PokéChamp (2025)	Text battle logs	Platform replays	Reverse-engineered	3M+ battles	Pokémon
Mastermind (2025b)	Text (encoded)	Algorithm synthesis	Algorithm-labeled	1.7M + 288K samples	2 Chess games
CombatVLA (2025c)	Screenshot + actions	Expert human play	Human AoT labels	25K imgs, 200 hrs	Black Myth
GF-Minecraft (2025)	Video + kbd/mouse	Programmatic capture	Auto-recorded	70 hrs, 2K clips	Minecraft
Orak (2026)	Text/Image/Both	LLM-generated	LLM + human verified	11K fine-tuning samples	12 games
Matrix-Game (2025d)	Video + kbd/mouse	MineDojo + auto-collected gameplay	Auto-labeled actions	∼ 400 frames	Minecraft
Matrix-Game 2.0 (2025a)	Video + kbd/mouse	UE + GTA5 auto-production	Auto annotations	25 FPS, minute-level gen	GTA5 + UE scenes
WildWorld (2026d)	Video + state	ARPG auto-capture	Auto-extracted	108M frames, 450+ actions	Monster Hunter
NitroGen (2026)	Video + gamepad	Web videos	CV overlay extraction	40K hrs, 1K+ games	1,000+ games
OpenP2P (2026)	Video + kbd/mouse	Expert human play	Recorded + VLM text	8.3K hrs, 600M pairs	45+ games
Proact-VL (2026)	Video + ASR text	YouTube streams	Multi-stage pipeline	561 hrs, 128K samples	12 games
Solaris (2026)	Video + actions	Automated multiplayer collection	Sync-recorded	Minute-level play	Minecraft
Matrix-Game 3.0 (2026c)	Video + action + pose + text	UE + auto collection + augmentation	Auto-generated	40 FPS; minute-long play	UE + AAA games

produced was inseparable from the game that generated it, and nothing transferred elsewhere. Era 3 breaks this closure. Foundation models require corpora that are collected before training begins, that span multiple games, and that carry value beyond the domain they came from. The question is no longer "*how many environment steps can the agent take?*" but "*what kind of data should we prepare, and where does it come from?*"

This section answers that question through three chapters and a coda.

- **Act I — Requisition: Learning from Human Play.** Act I examines how the field learned to extract training signals from human gameplay, progressing from initial structured demonstration datasets to internet-scale pseudo-labeled video and finally to multimodal time-aligned streams.
- **Act II — Excavation: Mining the Competitive Archive.** Act II investigates a complementary data source encompassing decades of replays archived on gaming platforms, alongside algorithmic synthesis methods that distill strategic knowledge directly from game mechanics.
- **Act III — Scaling: Crossing the Single-Game Boundary.** Act III faces the fundamental challenge of transcending the single-game boundary, documenting the dual pathways of broad automated collection and deep structured curation that the community has pursued to construct multi-game corpora.
- **Coda — Creation: The World Model as Data Engine**. Finally, the Coda looks beyond these three phases to the emerging paradigm of the *world-model-as-data-engine*, effectively connecting the datasets pillar to the Era 4 Demiurge vision that frames our comprehensive survey.

3.1 Act I — Requisition: Learning from Human Play

Human players generate enormous behavioral traces every day, and they constitute the most natural source of training data for game-playing agents. The challenge of learning from human play is a tension between

two quantities that move in opposite directions: the scale of available gameplay footage and the cost of attaching action labels to it.

The earliest efforts resolved this tension by choosing precision over scale. MineRL (Guss et al., 2019) compiled over 60 million state-action pairs from human Minecraft demonstrations. A parallel effort collected four million frames of online Counter-Strike gameplay through the same strategy (Pearce & Zhu, 2022). Atari-HEAD (Zhang et al., 2020) captured not only frames and actions but also human gaze data via eye-tracking across 20 Atari games. More recently, CombatVLA (Chen et al., 2025c) invested in collecting 200 hours of Black Myth: Wukong aligned at sub-15-millisecond precision, further annotated with Action-of-Thought labels. These datasets founded imitation learning for game agents, yet their construction cost grew linearly with data volume, leaving the resulting corpora orders of magnitude smaller than the total gameplay available on the internet.

To decouple data scale from human labeling costs, recent works increasingly employ learned models as automated annotators at both perceptual and semantic levels. At the perceptual level, Video PreTraining (VPT) (Baker et al., 2022) utilizes an Inverse Dynamics Model (IDM) to infer low-level actions from large-scale unlabeled internet videos, a paradigm also adopted by GROOT (Cai et al., 2024) to curate task-oriented benchmarks. Complementing this, semantic-level approaches leverage the reasoning capabilities of foundation models: Orak (Park et al., 2026) deploys strong LLMs to autonomously generate structured expert trajectories across various game genres, while Proact-VL (Yan et al., 2026) and GameGen-X (Che et al., 2025) employs VLMs to autonomously label and caption the video. Despite enabling internet-scale data collection, these automated methods share a critical limitation: residual noise, from IDM inaccuracies or LLM hallucinations, inevitably propagates into the training objective, degrading downstream policies over long horizons.

While VPT demonstrated that vision and action can be recovered at scale, human gameplay is far richer than frames paired with keyboard inputs. Atari-Head (Zhang et al., 2020) collects 117 hours of Atari-2600 games with human K&B action and gaze. PLAICraft (He et al., 2025b) pushes the boundary by collecting 10,000 hours of Minecraft gameplay with five modalities aligned in time: video, audio, natural language (voice and text), actions, and K&M signals. They captures not only what the player did but also what the player said and aspects of the player's cognitive state. Training a model requires observations spanning the full perceptual space the agent will operate in. Multimodal temporal alignment is therefore not optional but a prerequisite for agents that perceive and reason about the game world in a human-like manner.

At the opposite end of the cost spectrum, OpenP2P (Yue et al., 2026) invested in over 8,300 hours of meticulous human annotation across more than 45 3D games. Its core contribution is empirical: by varying model scale and data volume, the authors showed that behavior cloning follows a predictable scaling law and that increasing both dimensions leads to emergent causal reasoning. This reframes the quality-versus-quantity debate. Internet-scale pseudo-labeled data and expertly annotated data at moderate scale are not interchangeable. They serve different roles in the training pipeline. The former provides broad behavioral coverage at low per-sample cost, while the latter supplies the high-fidelity signal needed to unlock qualitatively new capabilities. ***How to optimally combine these two data regimes remains one of the central open problems in corpus design for game-playing agents.***

Viewed as a whole, the evolution of human-sourced game data follows a clear trajectory: from expensive synchronous annotation (Guss et al., 2019; Pearce & Zhu, 2022), through internet-scale pseudo-labeling (Baker et al., 2022), to multimodal temporal alignment (He et al., 2025b) and quality-centric scaling experiments (Yue et al., 2026). Each advance was driven by a specific limitation of its predecessor, and the limitations that remain collectively, most notably the noise ceiling of pseudo-labels and the prohibitive cost of high-quality annotation at scale, motivate the alternative data sources explored in the following sections.

3.2 Act II — Excavation: Mining the Competitive Archive

Competitive gaming platforms have accumulated decades of match replays, battle logs, and ranked statistics, forming a vast reservoir whose value for training game agents has only recently been recognized. Unlike the human demonstrations of Section 3.1, competitive replays are byproducts of the competitive ecosystem,

not produced for teaching AI. Yet they carry a distinctive advantage: match outcomes and player rankings provide implicit quality signals requiring no additional annotation.

The most direct use of this archive is to extract structured training data from historical replays. The practice traces back to SC2LE (Vinyals et al., 2017), which released approximately 800,000 StarCraft II ladder replays as a standardized research resource, establishing the template of leveraging platform recording infrastructure and competitive outcomes as implicit supervision. StarCraftImage (Kulinski et al., 2023) later drew on 60,000 replays from the same ladder to compile 3.6 million images with spatial reasoning labels derived from the game state, using the ladder's built-in rating system as a natural skill-level filter. Metamon (Grigsby et al., 2025) mines a full decade of Pokémon Showdown competitive replays, capturing the complete evolutionary record of the metagame—from casual play to tournament-level strategy through repeated cycles of innovation, counter-adaptation, and balance patches. This combination of quality stratification and temporal diversity enables robust policy learning without active environment interaction. The common thread is that competitive replays embed implicit reward signals—rankings and win rates—that directly inform offline RL reward shaping, bypassing both the expensive manual annotation and the noisy pseudo-labeling.

Beyond replays that preserve visual or state-level information, a parallel line of work extracts strategic knowledge from textual battle logs and dialogue records at even larger scale. Pokéchamp (Karten et al., 2025) compiled over three million competitive Pokémon battle logs, one of the largest corpora of structured competitive interactions. CICERO's Diplomacy corpus (Meta Fundamental AI Research Diplomacy Team (FAIR) et al., 2022) contributes over 40,000 full-press Diplomacy games in which natural language negotiation messages are aligned with board-state actions, encompassing the full spectrum of strategic communication. Werewolf-18K (Wu et al., 2024a) provides 18,800 sessions of nine-player social deduction games with complete dialogue logs and voting records. At this scale, strategic patterns and team-composition preferences emerge from aggregate distributions rather than manual expert identification.

For rule-complete games with tractable state spaces, training data can be algorithmically synthesized or simulated, bypassing the need for human or competitive data collection. Prominent examples include RL Unplugged (Gulcehre et al., 2020), which compiles diverse agent-generated replay buffers across Atari games; NLE (Küttler et al., 2020), which provides procedurally generated NetHack episodes with automated reward annotations; Game-RL (Tong et al., 2025), which utilizes Code2Logic to programmatically generate verifiable QA pairs; and Mastermind (Wang et al., 2025b), which synthesizes Q-value-annotated trajectories and optimal moves for Doudizhu and Go via extensive game-tree search. By enumerating optimal solutions, synthesized data often surpasses human demonstrations in quality, offering verifiable correctness without expensive manual annotation or noisy pseudo-labeling. However, its application is fundamentally bounded by computational tractability. As environments become partially observable, continuous, or open-ended, algorithmic synthesis becomes prohibitive. Consequently, this strategy serves as a robust complement to human- and competition-sourced data, rather than a universal replacement.

Taken together, the data sources examined in this section, including competitive replays with implicit quality gradients (Vinyals et al., 2017; Kulinski et al., 2023; Grigsby et al., 2025), large-scale battle logs and dialogue archives (Karten et al., 2025; Meta Fundamental AI Research Diplomacy Team (FAIR) et al., 2022; Wu et al., 2024a), agent-generated replay buffers and procedural environments (Küttler et al., 2020; Gulcehre et al., 2020), and algorithmic synthesis with verifiable labels (Tong et al., 2025; Wang et al., 2025b), substantially expand data volume and diversity. Yet they introduce new challenges: distribution shift from game updates, limited applicability of algorithmic synthesis to open-ended games, and the single-game focus that characterizes most competitive archives. These limitations motivate the cross-game scaling in the following section.

3.3 Act III — Scaling: Crossing the Single-Game Boundary

The datasets of Sections 3.1 and 3.2 share a common limitation: most are confined to a single game or a narrow family of related games. For Era 3's goal of building generalist agents, this single-game isolation is a fundamental bottleneck. Two complementary strategies have emerged for constructing multi-game corpora: a breadth path that maximizes game coverage through automated collection, and a depth path that maximizes annotation quality within a carefully curated set of games.

The breadth path has progressed through several generations of increasingly ambitious collection. ALE (Bellemare et al., 2013) first standardized 57 Atari 2600 titles under a uniform frame-action-reward interface, establishing the principle that a common data format spanning dozens of games could accelerate research. Genie (Bruce et al., 2024) scaled this to the internet era, harvesting over 30,000 hours of unlabeled 2D platformer videos spanning hundreds of titles and using unsupervised learning to discover latent action spaces without action annotation. NitroGen (Magne et al., 2026) and OpenP2P (Yue et al., 2026) represent the current frontier, assembling over 40,000 hours across more than 1,000 games through automated pipelines combining web-scale video harvesting with computer-vision-based action extraction. NitroGen's significance lies in demonstrating that heterogeneous multi-game corpora can serve as the data foundation for generative pre-training architectures, including diffusion and autoregressive models). Yet breadth comes at a cost: as game coverage grows, action heterogeneity grows correspondingly. An input in CS carries entirely different semantics from the same in Dota II. Unifying these disparate action representations remains a pressing open challenge, directly motivating the unified action tokenization.

The depth path takes the opposite stance, focusing on fewer games where every task provides rich, verifiable training signals. Game-RL (Tong et al., 2025) instantiates this with a corpus spanning 30 games and 158 tasks, each paired with a Code2Logic verifiable reward function—eliminating the reward misspecification that destabilizes RL on noisy data. OpenHA (Wang et al., 2025e) and JARVIS-VLA (Li et al., 2025a) together contribute hierarchical VLA trajectories totaling approximately 4.2 billion tokens across 800+ tasks, with layered annotation from high-level natural-language goals to atomic keyboard commands. This hierarchy provides the training signal for hierarchical policy learning at multiple temporal and semantic granularities. The value of the depth path is a quality leverage effect: verifiable signals eliminate the ambiguity that leads to reward hacking, so even moderately sized datasets can produce training stability that much larger noisy corpora cannot match.

Surveying the cross-game coverage landscape reveals a highly uneven map. The Minecraft ecosystem (Guss et al., 2019; Fan et al., 2022; Baker et al., 2022; Cai et al., 2024; Wang et al., 2025e; Li et al., 2025b) is the most mature, forming a multi-year chain of increasingly sophisticated resources. Text strategic games (Urbanek et al., 2019; Wu et al., 2024a; Karten et al., 2025; Grigsby et al., 2025; Meta Fundamental AI Research Diplomacy Team (FAIR) et al., 2022) have likewise received substantial coverage, whose text-native interfaces align naturally with LLM input formats. The Atari ecosystem (Bellemare et al., 2013; Zhang et al., 2020; Küttler et al., 2020; Gulcehre et al., 2020) provides another mature data chain available for Era 3 reuse. By contrast, several important categories remain underrepresented. Real-time 3D games, including FPS, MOBAs, and action RPGs, demand millisecond-level response times and high-dimensional continuous observations yet lack large-scale annotated datasets comparable to Minecraft. WildWorld (Li et al., 2026d) and CombatVLA (Chen et al., 2025c) represent early steps toward filling this gap for the ARPG subdomain, but FPS and MOBA remain data-scarce. Continuous action-space games pose a related challenge: the majority of existing datasets encode discrete keyboard inputs, leaving the continuous control regime of racing games, flight simulators, and physics-based environments with almost no dedicated training data. Social deduction games, despite session-level datasets such as Werewolf-18K (Wu et al., 2024a), lack the temporally extended records needed for online adaptation and real-time belief modeling. These gaps reflect structural mismatches between data collection infrastructure and domain requirements, and closing them will likely require new recording frameworks for high-frequency, continuous-control, and socially interactive environments.

In summary, the breadth path (Bellemare et al., 2013; Bruce et al., 2024; Magne et al., 2026; Yue et al., 2026) provides the data substrate for generative pre-training across a thousand games but exposes the action-space unification problem. The depth path (Fan et al., 2022; Tong et al., 2025; Li et al., 2025b; Wang et al., 2025e) provides stable, verifiable reward signals for reliable RL but within a narrower scope. The human annotation or model-as-judge paradigm (Fan et al., 2022; Che et al., 2025; Park et al., 2026; Yu et al., 2025) offers a scalable middle ground but remains bounded by teacher-model capability. No single corpus achieves simultaneous leadership in scale, annotation quality, and game diversity. This three-way trade-off is the defining unresolved tension of the dataset pillar, driving the exploration of world-model-based data generation in the next section.

3.4 Coda — Creation: The World Model as Data Engine

Game datasets in Era 3 are still largely produced by human-authored, code-based game engines. This substrate naturally limits scale by engine-side collection efficiency, annotation quality by the gap between the AI learner and the game engine, and diversity by the closed set of already existing human-made games. Therefore, to push beyond the current boundary of game data acquisition, *the key is no longer to propose another dataset, but to propose another data engine*—one in which AI itself becomes the driver of data generation. This paradigm shift essentially unifies data generation and model learning, creating a self-amplifying data flywheel that can jointly expand scale, improve quality, and broaden diversity.

Breaking the Code-Engine Ceiling. Manual code engines are hard-coded systems to ensure a stable gameplay experience: they provide human-friendly control interfaces, maintain preset game world boundaries for robust interaction, and are typically optimized for a single game world. By contrast, game world models are learned neural simulators that model game dynamics and predict action-conditioned futures, thereby making data generation more flexible and opening a path to relaxing the ceilings imposed by code engines.

First, they relax the *interface ceiling*. In code-based engines, collectable trajectories are restricted to predefined keyboard, mouse, controller, or API interfaces. Recent world models begin to absorb action conditioning into the model itself. *Genie* (Bruce et al., 2024) is an early landmark in this direction, as it learns interactive environments with latent actions from unlabeled videos rather than relying on explicitly recorded engine-side controls. *MineWorld* (Guo et al., 2025b) and *Matrix-Game 2.0* (He et al., 2025a) further strengthen this trend by emphasizing action following and action injection under explicit control signals, while *PAN* (PAN Team Institute of Foundation Models, 2025) extends the action channel beyond fixed low-level interfaces to language-conditioned actions. In this sense, what counts as an admissible action for data generation is increasingly less confined to traditional low-level interfaces, even if some systems still rely on predefined controls.

Second, world models loosen the *preset-boundary ceiling*. Code engines can only operate within mechanics, maps, and interaction patterns that have been explicitly authored in advance. By contrast, recent generative systems begin to instantiate and recombine interactive situations beyond the exact boundaries of logged gameplay. *GameFactory* (Yu et al., 2025) explicitly frames this as creating new games through generative interactive videos, while *GameGen-X* (Che et al., 2025) pushes toward open-world interactive generation with multimodal control. *PAN* (PAN Team Institute of Foundation Models, 2025) broadens this further to general, interactable, and long-horizon world simulation, and *IPR-1* (Zhang et al., 2025c) complements this trend by moving toward game-to-unseen generalization with prediction-reinforced physical reasoning. These works indicate that training worlds no longer need to remain fully confined to the exact environments and interaction patterns specified in advance.

Third, world models begin to push against the single-game ceiling. Traditional datasets are ultimately limited by the finite set of already existing human-made games, whereas recent neural game engines increasingly expand toward more diverse and open-ended worlds. *GameFactory* (Yu et al., 2025), *GameGen-X* (Che et al., 2025), and *PAN* (PAN Team Institute of Foundation Models, 2025) all broaden the scope of generated environments beyond narrow single-game settings, while *Solaris* (Savva et al., 2026) extends this expansion to multiplayer shared-world simulation with cross-player consistency.

Towards Self-Amplifying Data Flywheel. Beyond serving as a data engine that relaxes the constraints of code-based environments, the deeper promise of world models lies in entering the training loop itself as a self-amplifying data flywheel. In this setting, the world model is no longer a passive predictor of next states; instead, it becomes a training component that supports *adaptive generation* tailored to the player's demands, while also enabling *closed-loop co-evolution* with the player model.

Early evidence in robotics has already shown that such a paradigm is not only feasible, but also powerful. On the one hand, recent works of *adaptive generation* suggest that: rather than producing a fixed pool of synthetic rollouts, the world model could generate data conditioned on the learner's weaknesses and demands, thereby improving policy models in long-tail failure-prone scenarios. Ctrl-World (Guo et al., 2026) shows that imagined successful trajectories can be selectively synthesized for policy improvement, while

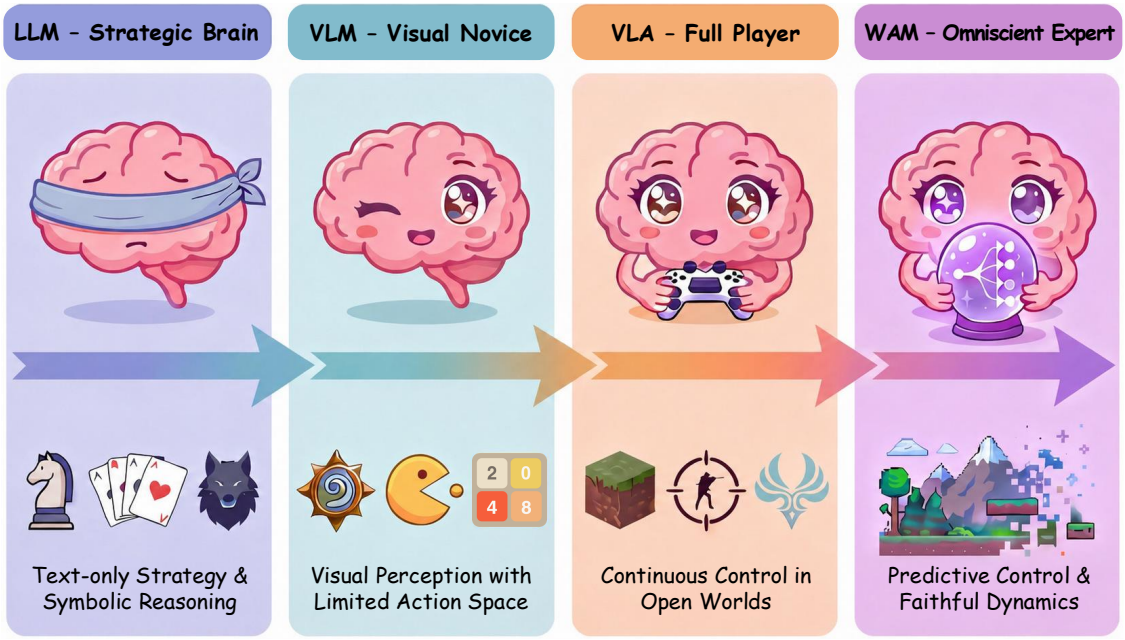

Figure 4: **Overview of Models in Game-playing AI.** Game-playing models progress from LLMs as strategic brains for text-based reasoning, to VLMs as visual novices that perceive game screens, then to VLAs as full players that execute low-level controls, and finally to WAMs as predictive experts that internalize game dynamics for world-aware decision-making.

WMPO (Zhu et al., 2026) demonstrates that on-policy post-training can be carried out inside a learned world model, allowing the generated training distribution to evolve together with the policy. On the other hand, world models can also support *closed-loop co-evolution* once they become part of the training loop itself. World-Gymnast (Sharma et al., 2026) explicitly couples world-model rollouts with policy refinement in a reinforcement learning loop, showing that simulator quality and policy capability can be iteratively improved together. A similar perspective is also emerging in embodied systems in Psi-R2 and Psi-W0 (Psibot Team, 2026), where the world model is not treated as a standalone predictor, but as an integrated component for policy evaluation, optimization, and autonomous self-improvement within a closed-loop pipeline. This evidence demonstrates the power of world models to function as a self-amplifying data flywheel that co-evolves with policy models.

Game LFM training naturally calls for a data flywheel that is tailored to and evolves with the model. Games are goal-directed, feedback-driven environments that naturally push players to improve through interaction. This makes adaptive training data especially valuable, as it can target the player's current weaknesses, failure modes, and learning demands. Since game difficulty often evolves with player competence, games also naturally motivate player–simulator co-evolution. However, current game world models still face substantial challenges before they can serve as such a data flywheel. High-quality training data require neural game engines to maintain temporal persistence and spatial consistency across the game multiverse, rather than merely generating locally plausible clips. Moreover, shared-world modeling that supports coherent multi-player rollouts remains underexplored, yet it is essential for scaling the flywheel toward industrial-level rollout throughput and richer collaborative or competitive training scenarios. Further advances in game world models will enable neural data engines to internalize more faithful game dynamics, ultimately forming a self-amplifying data flywheel that is tailored to and co-evolves with the game LFMs.

4 Models: The Evolving Brain of Generalist Game Player

Having established the paradigm shift towards generalist game players, we now turn to the cognitive core of these agents: the **Model**. In terms of $\mathcal{M}$, the model is the policy $\pi(a \mid o_{\leq t}, g)$ that maps observations Ω to

actions A; world models additionally learn the transition dynamics T. In traditional paradigms, game AI was locked to rigid interfaces. A chess engine takes a fixed board vector and returns a move index; a StarCraft agent reads a state tensor and emits an action ID. These systems are powerful within their contracts, but the contract is the cage. Change the board size, swap the piece types, or ask the engine to explain its reasoning in natural language, and the entire architecture collapses. Its input parser, action head, and reward function are all fixed to a single game. In contrast, building foundation agents requires architectural unification, including designing a singular model capable of processing unconstrained multimodal observations (Ω), reasoning over open-ended goals (G), and executing through universal human-computer interfaces (A).

Within the current landscape, this pursuit of a unified architecture has evolved through four progressive stages. Each stage addresses a critical bottleneck in closing the perception-action loop:

- **Large Language Models (LLMs):** Acting as the foundational cognitive engine, LLMs introduce world knowledge and strategic planning to games. However, restricted to textual inputs and outputs, they remain visually disconnected—capable of complex reasoning but blind to raw visual environments.
- **Vision-Language Models (VLMs):** By integrating visual encoders, VLMs equip the agent with direct pixel-level perception. While achieving multimodal grounding, their outputs typically remain semantic, creating a disconnect between visual understanding and continuous motor execution, such as keyboard, mouse and controller.
- **Vision-Language-Action Models (VLAs):** VLAs bridge the execution gap by mapping multimodal observations directly to low-level continuous control (e.g., unified keyboard and mouse operations). This end-to-end architecture seamlessly unifies perception, reasoning, and action within a single stream.
- **World Action Models (WAMs):** Representing the emerging frontier, WAMs move beyond reactive control by internalizing the environment's transition dynamics (T). By simulating future states and evaluating potential trajectories, these models enable predictive, model-based planning prior to execution.

We assign a model to a class by the most advanced stage of the perception–action loop it closes *natively*: text reasoning (LLM), visual perception (VLM), low-level action A (VLA), or learned dynamics T (WAM). These classes are cumulative rather than mutually exclusive, so we classify a hybrid system by its most advanced native capability and cross-reference it under the others; a pipeline that merely wraps a VLM with external memory or planning is treated as a Harness contribution (Section 5), not a new model class.

In the following subsections, we systematically review these four architectural paradigms. We explore their structural designs, training methodologies, perception-action loop designs, and the inherent limitations that continue to drive the evolution of generalist models.

4.1 LLMs as Game Brain: Reasoning the Strategies in the Textual Multiverse

Large Language Models (LLMs) (Brown et al., 2020; Ouyang et al., 2022; Bai et al., 2023) remove the constraint in Deep Reinforcement Learning. They take free-form text in and produce free-form text out. The same model can read a chess position (Tong et al., 2025), a Werewolf transcript (Wu et al., 2024a), a Pokemon battle log (Zhang et al., 2026b), or a Minecraft inventory (Feng et al., 2024), all without architectural change. Both observation and action live in natural language, making them in principle unbounded. Moreover, LLMs also bring broad world knowledge. Pre-training on game manuals, strategy forums, tournament commentary, and community wikis means the model already understands bluffing before it plays its first poker hand, and knows where diamonds spawn before it enters a Minecraft cave (Wang et al., 2024a). These two properties, ***flexible interface*** and ***pre-trained knowledge***, open the possibility of a single reasoning engine that generalizes across the textual multiverse: all games whose states and actions can be faithfully expressed in text.

Table 4: Summary of representative large foundation models in gaming AI

Paper	Arch.	Base Model	Input	Action	Training Recipe	Game Set
CICERO (2022)	LLM	BART-like 2.7B	Text state + dialogue	NL messages + orders	RL + planning (piKL)	Diplomacy
Voyager (2024a)	LLM	GPT-4	Text API	Code (skills)	Training-free	Minecraft
Werewolf-RL (2024)	LLM	GPT-4 + RL Net	Text log	Dialogue	MAPPO	Werewolf
LLaMA-Rider (2024)	LLM	LLaMA-2-70B	Text API	Text API calls	SFT	Minecraft
Chain of Summ. (2024)	LLM	GPT-3.5/4 + Qwen-7B	Text (TextSC2)	RTS commands	SFT (open-source)	StarCraft II
Werewolf LSPO (2025)	LLM	LLaMA-3-8B	Text dialogue	Strategic utterances	CFR + DPO	Werewolf
Refl. of Episodes (2026a)	LLM	GPT-3.5-Turbo	Text (keyframes)	RTS commands	Training-free	StarCraft II
Mastermind (2025b)	LLM	LLaMA-2-7B	Text board state	Game moves	SFT (synth. data)	Doudizhu, Go
Metamon (2025)	LLM	Transformer 200M	Text battle state	Move selection	Offline RL	Pokémon
Orak (2026)	LLM	Qwen-2.5-72B	Text / image	Game actions	SFT	12 genres
Think in Games (2025)	LLM	Qwen-2.5-7B/32B	Text state	Game policies	SFT + RL	Multiple
SpinGPT (2025)	LLM	LLaMA-3	Text game state	Poker decisions	SFT + RL	Poker
MS-GRPO (2025)	LLM	Qwen2.5-3B	Text state	Sequential decisions	MS-GRPO	Snake, Frozen Lake
MARSHAL (2025)	LLM	Qwen3-4B	Text scenarios	Strategic decisions	Self-play RL	Multiple
Stronger-MAS (2025)	LLM	Qwen3-1.7B/8B	Text prompts	Multi-agent actions	AT-GRPO	Multiple
GameTalk (2026)	LLM	LLaMA-3-3B	Text dialogue	Strategic dialogue	GRPO + DPO	Dialogue games
VAM (2026c)	LLM	Qwen2.5-3B/7B	Text (chess pos.)	Chess moves	RL + action masking	Chess
ISO-GRPO (2026)	LLM	Qwen2.5-0.5B	Text game state	Poker decisions	SFT + ISO-GRPO	Poker, Pokémon
Mixture of Masters (2026)	LLM	Custom GPT (small)	Text (chess pos.)	Chess moves	RL + MoE routing	Chess
Creative Agents (2023)	VLM	GPT-4V + Diffusion	Image+Text+Goal image	Code action	Prompting	Minecraft
TeamCraft (2024)	VLM	Vicuna-7B/13B	Vision + text	Multi-agent actions	Imitation Learning	Minecraft
VARP (2024a)	VLM	3 Models	Game Screenshot	Python kbd/mouse actions	Prompting+Retrieval	Black Myth: Wukong
ROCKET-1 (2025)	VLM	GPT-4o + SAM-2	Pixel image+Mask	Keyboard&Mouse	SFT	Minecraft
GameSense (2025)	VLM	Qwen2.5-VL	Pixel image	Keyboard&Mouse	Prompting+Module RL	3 action games
Cradle (2025c)	VLM	GPT-4V	GUI Screenshot	Keyboard&Mouse code	Prompting	4 commercial games
ViGaL (2026a)	VLM	Qwen2.5-VL-7B	Image + text	Game actions	RL	Arcade games
G1 (2025b)	VLM	Qwen2.5-VL-7B	Game Screenshot	Text action	SFT+GRPO	4 visual games
CoSo (2025)	VLM	LLaVA-1.6	Screenshot+Text	Text action	Online RL	Gym Cards
Game-RL (2025)	VLM	Qwen2.5-VL-7B	Game image+Text QA	Text answer	GRPO	30 games
VistaWise (2025)	VLM	GPT-4o	Text API + KG	API calls	Training-free	Minecraft
Orak (2026)	VLM	Qwen-2.5-72B	Text / image	Game actions	SFT	12 genres
VL-DAC (2025)	VLM	Qwen2-VL-7B	Screenshot+Text	Text action	PPO	MiniWorld&Gym-Cards
Proact-VL (2026)	VLM	LiveCC-7B-Base	Video stream	Commentary / guidance	SFT	12 live-streamed games
BC-CSGO (2022)	VA	EfficientNet&ConvLSTM	Pixel image	Keyboard&Mouse	SFT	CSGO
VPT (2022)	VA	from scratch(Transformer-XL-style)	Pixel image	Keyboard&Mouse	PT+SFT+PPG	Minecraft
GROOT (2024)	VA	VPT	Pixel image	Keyboard&Mouse	SFT	Minecraft
P2P(0.1) (2025a)	VA	from scratch(Decoder-only Transformer)	Pixel image	Keyboard&Mouse	PT	Roblox & MS-DOS games
NitroGen (2026)	VA	Siglip&DiT	Pixel image	Gamepad	PT+SFT	1000+ video games
STEVE-1 (2023)	VLA	VPT&MineCLIP	Pixel image+Text	Keyboard&Mouse	SFT	Minecraft
OmniJarvis (2024b)	VLA	LLaVA-7B	Pixel image+Text+Action	Action token	SFT	Minecraft
SIMA (2024)	VLA	SPARC&Phenaki&Transformer-XL	Pixel image+Text	Keyboard&Mouse	SFT	Multiple 3D games
P2P(0.3) (2025b)	VLA	from scratch(EfficientNet&Transformer)	Pixel image+Text	Keyboard&Mouse	PT+SFT	Roblox & MS-DOS & FPS games
SIMA2 (2025)	VLA	Gemini Flash-Lite	Pixel image+Text	Keyboard&Mouse	SFT+RL	Multiple 3D games
JarvisVLA (2025b)	VLA	Qwen2-VL-7B	Pixel image+Text	Action token	CPT+SFT	Minecraft
OpenHA (2025c)	VLA	Qwen2-VL-7B	Pixel image+Text	Multiple action space	PT+SFT	Minecraft
CombatVLA (2025c)	VLA	Qwen2.5-VL-3B	Pixel image+Text	Text Action	SFT	Wukong & Sekiro
Lumine (2025b)	VLA	Qwen2-VL-7B	Pixel image+Text	Text K&M Action	CPT+SFT	Genshin Impact
Game-TARS (2025f)	VLA	Qwen2.5-VL-7B&Seed-VL-1.5	Pixel image+Text	Code Action	CPT+SFT+RFT	500+ video games
UI-TARS1.5 (2025)	VLA	Qwen2.5-VL-7B	Pixel image+Text	Code Action	SFT+RL	14 diverse games
UI-TARS2 (2025c)	VLA	Qwen2.5-VL-7B	Pixel image+Text	Text Action	SFT+PPO	15+ 2D games
OpenP2P (2026)	VLA	from scratch(Decoder-only Transformer)	Pixel image+Text	Keyboard&Mouse	PT+SFT	Multiple 3D games
Main-VLA (2026)	VLA	Qwen2-VL-7B	Pixel image+Text	Keyboard&Mouse	SFT	Minecraft & Peace & Valorant
Genie (2024)	WM	Tokenizer + Transformer	Unlabeled video	Latent actions	Video pretraining	2D platformers
Oasis (2024)	WM	Autoregressive Transformer	Game frames	Keyboard&Mouse	Autoregressive video modeling	Minecraft-like world
GameNGen (2025)	WM	Diffusion model	DOOM frames + actions	Game actions	Diffusion next-frame modeling	DOOM
MineWorld (2025b)	WM	Autoregressive Transformer	Frames + actions	Keyboard&Mouse	Action-conditioned pretraining	Minecraft
GameFactory (2025)	WM	Video diffusion	Text/image/video prompts	Interactive actions	Diffusion fine-tuning	Generated games
Matrix-Game 3.0 (2026c)	WM	Diffusion Transformer	Frames + actions + memory	Keyboard&Mouse	AR diffusion + distillation	Minecraft
GameGen-X (2025)	WM	Video diffusion	Multimodal prompts	Interactive actions	Diffusion fine-tuning	Open-world games
PAN (2025)	WM	Long-horizon WM	History + text actions	Language actions	Long-horizon simulation	General worlds
WorldCam (2026)	WM	Autoregressive Transformer	Frames + camera pose	Camera controls	Pose-aware AR training	3D game worlds
DreamerV3 (2025)	WAM*	RSSM	Pixels / states	Game actions	World-model RL	Atari, Minecraft, DMLab
JOWA (2025)	WAM*	World-action Transformer	Offline trajectories	Future actions	Joint WM-action pretraining	Atari
DIAMOND (2024)	WAM*	Diffusion WM	Atari pixels	Atari actions	In-model policy training	Atari

* Prototype WAMs: they couple world prediction with action/policy learning, but are not fully unified gaming WAMs with large foundation models.

This section traces how far training-based methods have pushed LLMs toward this goal, along two threads: natively textual games, where the challenge is strategic depth; and visual games accessible through text, where the challenge shifts to model design under perceptual constraints.

The Native Territory: Text Multiverse of Games. In text games, the observation is text, the action is text, and the core challenge is reasoning. This makes them the natural arena for sharpening LLM strategic capability through training.

Social deduction games demand unconstrained natural language as the action space: the agent persuades, deceives, and detects lies through dialogue, making LLMs irreplaceable. The landmark is CICERO (Meta Fundamental AI Research Diplomacy Team (FAIR) et al., 2022), which first achieved human-level performance in Diplomacy via game-theoretic planning. Subsequent shifts moved from modular designs (Xu et al., 2024) to unified alignment, where Werewolf LSPO (Xu et al., 2025) and GameTalk (Vendrell et al., 2026) integrate strategies directly into LLMs via DPO and GRPO.

In imperfect information games (e.g., Poker), research focuses on long-horizon credit assignment. The GRPO (Guo et al., 2025a) family, including MS-GRPO (Dilkes et al., 2025) and ISO-GRPO (Xia et al., 2026), addresses sparse rewards, while SpinGPT (Maugin & Cazenave, 2025) standardizes the SFT-RL pipeline.

Offline learning also thrives, with Metamon (Grigsby et al., 2025) reaching expert levels in Pokemon via ranked replays. For board games, models like Mastermind (Wang et al., 2025b) and Mixture of Masters (Frisoni et al., 2026) leverage algorithmic synthesis and MoE architectures to provide explainable strategic reasoning. VAM (Zhang et al., 2026c) masks illegal chess moves during RL to focus exploration on strategic quality. However, text games also expose reasoning limits. Credit assignment over hundreds of turns remains unsolved.

Moreover, MARSHAL (Yuan et al., 2025) and Stronger-MAS (Zhao et al., 2025) train on self-play from multiple games, testing cross-domain transfer. Game-RL (Tong et al., 2025) spans 30 games with Code2Logic verifiable rewards; ViGaL (Xie et al., 2026a) and Think in Games (Liao et al., 2025) show that game RL improves general reasoning beyond games, suggesting games are training grounds for broader intelligence. Yet, multi-agent coordination degrades with agent count and belief-nesting depth. Most fundamentally, tracking what each player knows, believes, and intends across long interactions imposes a combinatorial cognitive load that current context windows struggle to support. These reasoning bottlenecks define the frontier in the textual multiverse.

Expanding the Frontier: LLMs in Visual Games. Most modern games are visual. LLMs cannot see pixels but can play when game states reach them as text, through APIs (Mineflayer (contributors, 2013), TextStarCraft II (Ma et al., 2024)), text protocols (Pokemon Showdown), or perception modules. In this pipeline, the LLM acts as the reasoning core for planning and decomposition. Minecraft remains the central benchmark: Voyager (Wang et al., 2024a) introduced autonomous skill libraries, followed by LLaMA-Rider (Feng et al., 2024) for exploration and JARVIS-1 (Wang et al., 2023b) for memory-augmented planning. Advances in VistaWise (Fu et al., 2025) and TeamCraft (Long et al., 2024) further integrated cross-modal knowledge and multi-agent scaling. In complex RTS environments like StarCraft II, Chain of Summarization (Ma et al., 2024) and Reflection of Episodes (Xu et al., 2026a) tackle context limits through hierarchical abstraction and episodic memory.

Despite successes in multi-game training (Park et al., 2026), ***a perceptual ceiling persists***. Textual mediation often discards critical spatial relations and real-time visual cues, limiting the agent's ability to master continuous dynamics. This bottleneck necessitates the transition to VLMs and VLAs, which integrate pixel-level perception and end-to-end motor control.

4.2 VLMs as Game Novice: Opening the Eyes to the Visual Multiverse

While LLM-based agents excel at planning and reasoning in text-mediated games, their reliance on text-converted state representations inevitably discards the spatial, temporal, and perceptual details that human players routinely extract from visual feedback. This perceptual bottleneck limits generalization across visually diverse game environments. Vision-Language Models (VLMs) (Li et al., 2024; Bai et al., 2025b;a) address this gap by integrating visual encoders with language understanding, enabling agents to perceive and reason about game worlds directly from pixels.

VLMs as Perceptual Reasoners for Game Decision-Making. An early exploration in this direction is MineDojo (Fan et al., 2022), which trains a contrastive video-language model, MineCLIP, to score the alignment between an agent's video trajectory and a language instruction, serving as a dense reward for reinforcement learning without manual reward design. This work demonstrates that vision-language alignment can effectively guide agents across diverse open-ended tasks, establishing the potential of joint vision-language learning for gameplay.

Subsequent works further leverage VLMs as the perceptual and reasoning backbone. Creative Agents (Zhang et al., 2023) pairs language and diffusion-based goal image generation with a VLM code controller for creative building in Minecraft. Cradle (Tan et al., 2025c) introduces the General Computer Control setting, where a VLM reasons over screenshots and outputs keyboard-mouse actions as code, generalizing across commercial games without game-specific APIs. VARP (Chen et al., 2024a) applies VLM-driven action planning to ARPG combat in *Black Myth: Wukong*, combining a visual action planning module with a visual trajectory tracking system that jointly translates raw screen observations into timed combat decisions. ROCKET-1 (Cai et al., 2025) addresses the spatial information bottleneck of language by introducing visual-temporal context prompting, where a VLM communicates interaction targets to a low-level policy via segmentation

masks. GameSense (Lu et al., 2025) shifts the VLM from a per-step controller to a developer that synthesizes reusable action-feedback modules for real-time play. Proact-VL (Yan et al., 2026) further extends VLMs to real-time streaming, building a proactive video language model that processes continuous gameplay and autonomously decides when to respond, serving as an interactive AI companion.

Reinforcement Learning for Sharper Visual Game Reasoning. The recent success of reinforcement learning (RL) in improving VLM reasoning has also influenced game agent research. G1 (Chen et al., 2025b) trains VLMs via RL self-evolution in a multi-game environment, showing that perception and reasoning mutually bootstrap during training. CoSo (Feng et al., 2025) improves exploration efficiency by using counterfactual reasoning to focus RL updates on action-critical tokens. Game-RL (Tong et al., 2025) synthesizes verifiable reasoning data from game source code and applies GRPO-based training (Shao et al., 2024), finding that RL on game data alone transfers to broader vision-language benchmarks. VL-DAC (Bredis et al., 2025) decouples token-level PPO (Schulman et al., 2017) updates from environment-step value estimation, demonstrating that RL training in lightweight simulators generalizes to real-image agentic control. Collectively, these efforts confirm that RL post-training consistently sharpens VLM decision-making in game settings and can yield benefits beyond the game domain.

Despite these advances, VLM-based game agents still rely on indirect execution mechanisms such as code generation, API calls, or predefined skill libraries and functions to translate high-level understanding into concrete actions. This decoupled architecture introduces latency, limits control granularity, and confines VLM agents to the role of slow-frequency planners. Realizing a truly general game agent thus still requires closing the perception-action loop within an end-to-end framework.

4.3 VLAs as Game Player: Closing the Perception-Action Loop

While VLMs provide visual perception, their outputs remain semantic and require external modules for motor execution. Vision-Language-Action models (VLAs) (O'Neill et al., 2024; Kim et al., 2024) close this gap by mapping multimodal observations directly to low-level control (Chi et al., 2023; Su et al., 2025) within end-to-end architectures. This paradigm builds on a sustained line of vision-to-action (VA) research that learns sensorimotor policies from gameplay.

Scaling Vision-to-Action Pre-training from Videos. Early VA research has explored video-based pre-training as a scalable route to sensorimotor policy learning. CSGO (Pearce & Zhu, 2022) demonstrates that behavioral cloning (BC) Jang et al. (2021) can scale to millions of human gameplay frames to produce competent FPS agents. VPT (Baker et al., 2022) then extends BC by leveraging an inverse dynamics model to pseudo-label massive internet gameplay videos and training a causal policy network on the resulting corpus, confirming the potential of internet-scale video pre-training for action generation. STEVE-1 (Lifshitz et al., 2023) builds on VPT with goal-conditioned fine-tuning, showing that the paradigm scales to multi-step task compositions. GROOT (Cai et al., 2024) extends goal conditioning by using reference gameplay videos as instructions, demonstrating that video pre-training can simultaneously provide the policy prior. NitroGen (Magne et al., 2026) further broadens the scope of VA paradigm, training a joint ViT-DiT vision-action architecture on 40K hours of gameplay spanning over 1,000 titles to achieve general-purpose continuous action generation for smoother and higher-resolution control. In light of the above VA advances, Pixels2Play (Yue et al., 2026) provides a systematic investigation of BC scaling laws, showing that increasing model capacity and data volume not only improves imitation fidelity but also promotes the emergence of causal reasoning over spurious correlation.

Leveraging Multimodal Understanding for Action Generation in VLAs. Building on the VA foundations, contemporary VLAs retain the end-to-end sensorimotor backbone while further integrating language comprehension into a unified multimodal policy, OmniJARVIS (Wang et al., 2024b) pioneers the paradigm by processing multimodal inputs through a shared encoder-decoder architecture and training action outputs via BC, with SIMA (Raad et al., 2024) validating the generality of this paradigm by training a language-conditioned agent across diverse commercial video games. This framework is quickly inherited by JARVIS-VLA (Li et al., 2025b) and OpenHA (Wang et al., 2025e), which further leverage pre-trained VLMs as the perceptual encoder and append lightweight policy decoders to generate actions.

The adoption of VLMs raises whether reinforcing their domain-specific understanding can benefit downstream action generation. CombatVLA (Chen et al., 2025c) distills tactical reasoning into learned representations through a progressive *Action-of-Thought* curriculum; MAIN-VLA (Zhou et al., 2026) prunes redundant visual and linguistic semantics into compact, action-critical features; and Lumine (Tan et al., 2025b) unifies language reasoning with action generation in a closed-loop framework, achieving instruction-adherent long-horizon control with cross-game transfer. These results indicate that deepening the perceptual and reasoning capacity of VLM encoders consistently improves both long-horizon interaction and cross-game generalizability.

Building on these supervised foundations, recent work introduces RL post-training to enable self-improvement beyond demonstration data—whether through self-generated gameplay across 3D virtual worlds (SIMA-team et al., 2025) or multi-turn RL within GUI environments at scale (Wang et al., 2025c). This supervised pre-training followed by RL post-training pipeline has become a prevalent recipe for generalist VLA agents, yielding policies that continuously adapt across heterogeneous interactive environments.

Toward Real-Time Control: Evolving Action Representations. To achieve better real-time control, the development of game VLAs has also focused on improving action representation. Early VLAs such as JARVIS-VLA (Li et al., 2025b) and Game-TARS (Wang et al., 2025f) discretize control into per-step tokens generated autoregressively alongside language output. To reduce the resulting inference overhead, subsequent work shifts toward *action chunks*: CombatVLA (Chen et al., 2025c) applies truncated decoding of action-of-thought sequences for a 50-fold speedup in 3D ARPG combat, while Lumine (Tan et al., 2025b) predicts six 33,ms textual chunks per step to sustain 30,Hz keyboard-mouse control. This trajectory culminates in NitroGen (Magne et al., 2026), which bypasses language-space encoding entirely and employs a flow-matching diffusion transformer to generate 16-step chunks of *continuous* gamepad actions from a single RGB frame. Collectively, these advances chart a path from discrete, per-token action decoding toward unified continuous representations capable of accommodating diverse control modalities at real-time rates.

4.4 WAMs as Game Expert: Internalizing the Dynamics of the Game Multiverse

VLAs have achieved strong semantic reasoning and fine-grained control. However, their understanding of game-world evolution still remains largely semantic, and their control generation is predominantly reactive. WAMs offer a new paradigm to overcome these limitations in two aspects:

- **Predictive Control:** By internalizing action-conditioned dynamics, WAMs support *prediction before action*, so that decisions are guided by future anticipation rather than only current states.
- **Faithful Dynamics:** WAMs learn *world models* of games from visual and control signals, leading to more faithful dynamics modeling of how game worlds evolve under interaction.

More importantly, these two properties also make WAMs inherently more *general*: instead of merely fitting game-specific action patterns, they capture transferable regularities of how game worlds evolve under intervention across diverse games. In this sense, WAMs shift game agents from reactive perceive-and-act systems to predictive plan-and-execute systems, marking a crucial step toward more generalist game players.

Early Prototypes of Predictive Control. Early world-model-based game agents marked the first step toward predictive game playing by introducing dynamics internalization and future modeling into control. In this sense, they can be viewed as prototypes of gaming WAMs. Action-Conditional Video Prediction (Oh et al., 2015) first made the dependency between future frames and control variables explicit, training convolutional and recurrent predictors to roll out Atari futures conditioned on candidate actions. World Models (Ha & Schmidhuber, 2018) compressed visual observations with a VAE, learned latent dynamics with an MDN-RNN, and evolved a compact controller inside the learned "dream" environment. SimPLe (Kaiser et al., 2020) turned this idea into an iterative Atari training loop, repeatedly fitting a stochastic video model from real interaction and training PPO on short imagined rollouts. The Dreamer series (Hafner et al., 2020; 2021; 2025) jointly trains an encoder, representation function, and dynamics function within recurrent latent state-space models, enabling implicit predictive control through imagined actor-critic learning across increasingly

diverse control tasks. IRIS (Micheli et al., 2023) replaced recurrent latent prediction with a transformer world model over discrete image tokens, and DIAMOND (Alonso et al., 2024) further showed that diffusion-based dynamics can preserve action-relevant visual details and train Atari agents entirely inside a learned world model.

Their shared contribution lies not in a modern unified world-action architecture that jointly maps historical observations and actions to future observations and actions, but in demonstrating that prediction-guided control can substantially improve control effectiveness. By grounding control in predicted futures, these works establish the early prototype of gaming WAMs. Nevertheless, limited generative capacity and insufficient dynamics-modeling ability prevented these early models from fully internalizing complex, visually grounded dynamics across a broader set of modern games with complex spatial layouts and rich interaction mechanisms.

Gaming World Models with Faithful Dynamics. While early prototypes had already demonstrated the effectiveness of predictive control, recent gaming WMs have strengthened the dynamics model itself. Genie (Bruce et al., 2024) established this direction by learning latent actions and autoregressive dynamics from unlabeled videos, showing that interactive environments can emerge without explicit action labels. This autoregressive paradigm was later grounded in Minecraft-like sandbox worlds by Oasis, MineWorld, and the Matrix-Game series (Decart & Julian Quevedo, 2024; Guo et al., 2025b; Zhang et al., 2025d; He et al., 2025a; Wang et al., 2026c), which progressively improved action-conditioned generation, action following, streaming rollout, and long-horizon memory, moving gaming WMs from video continuation toward controllable neural environments. As domains become visually richer, diffusion- or flow-style neural engines further improve fidelity and controllability. GameNGen (Valevski et al., 2025) shows that a diffusion model trained on DOOM traces can operate as a real-time neural game engine for FPS dynamics, while GameFactory and GameGen-X (Yu et al., 2025; Che et al., 2025) use video-generation backbones and multi-stage training to extend world modeling toward open-domain, modern 3D, and open-world games. Beyond real-time controllability, recent works increasingly target persistent dynamics. PAN and Model as a Game (PAN Team Institute of Foundation Models, 2025; Chen et al., 2025a) emphasize long-horizon interaction, mechanic consistency, and numerical/spatial reliability; WorldPlay and WorldCam (Sun et al., 2025b; Nam et al., 2026) strengthen geometric consistency and spatial revisitation, with WorldCam using camera-pose control for long-horizon 3D grounding; and Solaris (Savva et al., 2026) extends world modeling to synchronized multiplayer Minecraft with coherent shared states. These advances suggest that dynamics modeling of modern games with complex spatial structures and interaction mechanisms is no longer fragile. Rather, world modeling for modern games is becoming increasingly reliable, providing the foundation on which gaming WAMs can internalize, predict, and ultimately control.

Towards Modern Gaming World Action Models. World Action Models (WAMs) jointly model future world states and future actions from interaction history, enabling prediction and control to be learned within a unified framework rather than as separate modules. Promising evidence has already emerged in robotics, where works such as DreamZero (Ye et al., 2026) couple predictive world prediction with action generation, improving embodied control and cross-task generalization. In games, the Dreamer series (Hafner et al., 2020; 2021; 2025) provides an important prototype of this paradigm. Although it does not fully unify world-action anticipation, it demonstrates the effectiveness of acting on latent states shaped by internalized future dynamics. JOWA (Cheng et al., 2025) takes a more direct step toward modern gaming WAMs by jointly pretraining world and action predictions, showing that world-action coupling can improve decision-making in game environments. Nevertheless, gaming WAMs remain largely underexplored. Existing efforts have not yet fully leveraged the stronger world-model substrate emerging from recent gaming WMs, where faithful dynamics are learned from visual and control signals across richer environments. In this setting, WAMs could further enhance predictive control by grounding action generation in deeper anticipations of world evolution. More importantly, it may enable agents to learn transferable dynamics rather than game-specific action patterns, moving gaming WAMs toward a more fundamental world understanding and ultimately toward *omni-reality adaptability* across the game multiverse.

Table 5: Summary of representative harness designing in gaming AI.

Paper	Perception	Action	Reasoning	Real-time Reactivity	Memory	Adaptive Learning
DEPS (2023a)	Textual Env.	Semantic	Workflow	–	Working Only	Self-Reflection
AgentVerse (2024b)	Textual Env.	Semantic	Multi-Agent	–	Working Only	–
Voyager (2024a)	Textual Env.	Code Control	Workflow	–	Both	Evolving Skills
StarCraft Bench (2024)	Textual Env.	Semantic	Prompt ENGR	–	Working Only	–
Cradle (2025c)	Raw	GUI	Workflow	Pausing	Both	Self-Reflection
GAMEBoT (2024)	Textual Env.	Semantic	Prompt ENGR	–	Working Only	–
GameSense (2025)	Visual Scaffold	Code Control	Workflow	Decoupling Reasoning from Action	Both	Self-Reflection & Evolving Skills
VideoGameBench (2025a)	Raw	GUI	Workflow	Pausing	Working Only	–
LPLH (2025)	Textual Env.	Semantic	Workflow	–	Both	Self-Reflection
Orak (2026)	Visual Scaffold & Textual Env.	Semantic	Workflow	Pausing	Both	Self-Reflection
GMH (2025f)	Visual Scaffold & Textual Env.	Semantic	Workflow	–	Working Only	Prompt Opt.
VistaWise (2025)	Visual Scaffold	GUI	Prompt ENGR	Pausing	Both	–
FlashAdventure (2025)	Raw	GUI	Workflow	–	Both	Self-Reflection
TITAN (2025a)	Textual Env.	Semantic	Workflow	–	Both	Self-Reflection
CWM (2025)	Textual Env.	Code Control	Workflow	–	Both	Self-Reflection
AgileThinker (2025)	Textual Env.	Semantic	Workflow	Dual Thread	Working Only	–
PERIL (2025)	Textual Env.	Semantic	Prompt ENGR	–	Working Only	–
DSGBench (2025)	Textual Env.	Semantic	Workflow	Pasuing	Working Only	Self-Reflection
Lmgame-Bench (2025a)	Textual Env.	Semantic	Prompt ENGR	Pausing	Working Only	Prompt Opt. & Self-Reflection
DeepPHY (2026b)	Visual Scaffold	Semantic	Prompt ENGR	–	Working Only	Self-Reflection
Steve-Evolving (2026b)	Visual Scaffold	Code Control	Workflow	–	Both	Evolving-Skills
PokeAgent Challenge (2026)	Textual Env.	Sematic	Multi-Agent	–	Both	Self-Reflection
STAR (2026c)	Textual Env.	Sematic	Prompt ENGR	Batched Command Execution	Working Only	–
NitroGen (2026)	Raw	GUI	–	Pausing	Working Only	–
MineNPC-Task (2026)	Textual Env.	Code Control	Workflow	–	Both	Self-Reflection
BotzoneBench (2026a)	Textual Env.	Sematic	Prompt ENGR	–	Working Only	–
GameVerse (2026a)	Raw	GUI/Sematic	Workflow	–	Both	Self-Reflection

5 Harness: The Nervous System of Generalist Game Player

In terms of $\mathcal{M}$, the harness supplies what the policy lacks natively: a belief over the state S from interaction history (memory), search over the transition function T (planning), perception of the observation Ω, and grounding of decisions into executable actions A.

Classical cognitive science frameworks, such as the Unified Theories of Cognition (UTC), posit that intelligence is fundamentally a closed-loop system intertwined with essential functions like perception, memory, reasoning, and action (Newell, 1994; Laird et al., 2017). However, although the foundation models demonstrate powerful representation capabilities, they are stateless by design and suffer from inherent limitations, such as memory volatility (Kim, 2025; Tang et al., 2026). Consequently, they fail to function as complete intelligent systems capable of navigating dynamic and complex environments independently. To bridge this gap, researchers have engineered various Modular Harnesses designed to decouple and augment the foundational functions required by autonomous agents.

In the current landscape, the most prominent harnessing components can be categorized into six core dimensions:

- **Perception:** The perception module extracts multi-modal information from raw environmental observations (Ω) and transforms it into structured representations—such as textual prompts—that the underlying reasoning engine (e.g., LLMs) can process.
- **Action:** The action module translates the agent's internal decisions into executable operations. It involves defining the allowable action space (A), designing the invocation mechanisms (e.g., APIs, code execution, or GUI operations), and applying these actions to alter the environment state.
- **Reasoning:** Serving as the cognitive core, the reasoning module synthesizes the predefined goals (G) with perceptual inputs to estimate the current state (S). Instead of executing actions directly, it performs task decomposition, multi-step planning, and selects the optimal action trajectory.
- **Real-time Reactivity:** This mechanism is specifically designed to mitigate the critical gap between the inherent inference latency of foundation models and the temporal constraints of dynamic, real-time environments, ensuring timely and robust responses.
- **Memory:** To overcome the stateless nature of foundation models, the memory module persistently stores and manages key interaction histories, environment knowledge, and procedural experiences. It provides the agent with a coherent context for consistent decision-making.

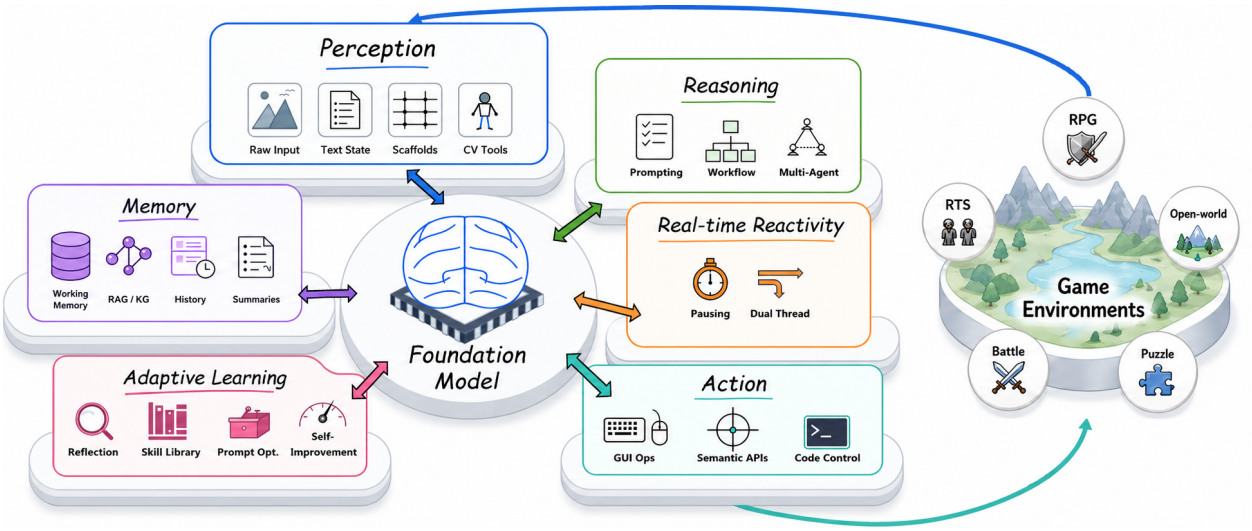

Figure 5: **Overview of Harness in Game-playing AI.** The harness bridges foundation models and game environments through six interdependent dimensions: Perception parses raw observations into model-readable representations, Action grounds decisions into executable interfaces, and Reasoning synthesizes goals with perceived states into multi-step plans. Real-time Reactivity mitigates inference latency in dynamic environments, Memory persists interaction histories beyond stateless inference, and Adaptive Learning distills trajectories into reusable experiences for continual self-evolution.

- **Adaptive Learning:** The adaptive learning module drives the continuous evolution of the agent. By reflecting on successes and failures, it abstracts interaction trajectories into reusable heuristic experiences, enabling the system to self-correct and autonomously adapt to novel or dynamic environments over time.

In the following sections, we systematically review the harness designs, practical efficacies, and current limitations corresponding to each of these six functional dimensions.

5.1 Perception

In game interactions, vision serves as the most fundamental perceptual channel for humans, enabling them to observe dynamic environments, read textual information, confirm game states, and guide precise action execution. However, this natural process poses a significant challenge for foundation models. Large Language Models (LLMs) are inherently restricted to the text modality; meanwhile, although Vision-Language Models (VLMs) can process visual inputs and comprehend global semantics, they still exhibit a substantial gap compared to human visual adaptation, particularly in fine-grained spatial alignment. Recent studies reveal that when raw game screenshots are used directly as input, VLMs frequently encounter issues such as spatial localization bias, semantic ambiguity, and severe perceptual hallucinations (Zhang et al., 2026a; 2025a). To bridge this "spatial-semantic gap" and alleviate the models' cognitive load, researchers have designed various perception harnesses. These module designs primarily include: **Textual Environment** and **Visual Scaffolding**.

Textual Environment. The textual environment paradigm bypasses the inherent spatial grounding limitations of vision models by converting complex, high-dimensional game states into structured natural language, tabular formats, or symbolic code representations. Recent studies implement this paradigm through distinct, framework-specific designs to optimize cognitive load. For instance, the LPLH framework utilizes Interactive Fiction games, which are natively text-based, thereby entirely circumventing visual perception challenges and allowing the agent to focus purely on semantic understanding and narrative reasoning (Zhang & Long, 2025). For visually complex Real-Time Strategy games, DSGBench and Reflection of Episodes lever-

age the TextStarCraft II environment to parse dense environmental states—such as unit counts, resource levels, and building statuses—into comprehensive textual summaries (Ma et al., 2024; Xu et al., 2026a; Tang et al., 2025). Similarly, The PokeAgentChallenge and Pokéchamp Agent abstract the intricate battle mechanics of Pokémon into structured text via the Pokémon Showdown simulator (Karten et al., 2025; 2026). For spatially structured tasks, frameworks like LmgameBench extract backend data from grid-based games (e.g., Sokoban, 2048, Tetris) to convert visual layouts into textual tables, explicitly listing object coordinates and their attributes (Hu et al., 2025a). For open-world 3D sandbox games, frameworks like Voyager and the Orak benchmark utilize the Mineflayer JavaScript API to entirely bypass raw visual processing (Wang et al., 2024a; Park et al., 2026; contributors, 2013). Taking abstraction a step further, Code World Model translates natural language game rules and historical trajectories into an executable Python-based world model. This approach transforms the game environment into formal code, serving as a reliable simulator to support rigorous algorithmic planning, such as Monte Carlo Tree Search (Lehrach et al., 2025).

Visual Scaffolding. In scenarios where visual modalities need to be retained, frameworks deploy visual scaffolding or external computer vision (CV) tools to bridge the spatial-semantic gap. To directly enhance the raw visual input, General Modular Harness explicitly overlays grid lines and coordinate labels onto the game images to reduce perception errors (Zhang et al., 2025f). Alternatively, Steve-Evolving anchors continuous visual experiences into structured state snapshots (e.g., precise coordinates, inventory summaries, health, and GUI states), pairing them with fine-grained execution diagnoses to facilitate reliable recall and agent evolution (Xie et al., 2026b). In privilege-free settings, modules employ external CV models as visual parsers. For example, GameSense operates entirely on real-time game screenshots without API access, deploying a suite of CV tools (e.g., Grounding DINO for object detection and OpenCV for state reading) to parse highly dynamic environments in FPS and action games (Lu et al., 2025). Similarly, VistaWise and the Orak benchmark deploy lightweight object detection models (such as YOLOv10 and YOLOv11) to extract bounding boxes and entity positions from raw screenshots (Fu et al., 2025; Park et al., 2026). Crucially, they perform perception abstraction to translate continuous spatial data into discrete semantic concepts (e.g., categorizing the distance between fighters as "very close" or "far").

5.2 Action

When playing games, human players seamlessly translate high-level semantic intentions (e.g., "aim and shoot the enemy") into fine-grained motor executions (e.g., moving the mouse to specific coordinates and left-clicking). While current foundation models exhibit robust reasoning and tool-invocation capabilities, they lack native motor skills to directly manipulate digital environments. Therefore, defining an appropriate action space and encapsulating environmental interactions into executable interfaces serves as the critical bridge connecting the model's cognition to the game engine. As action spaces have rapidly evolved from predefined function calls toward more flexible interface-level navigation and code execution, we categorize current Action modules into three progressive paradigms: **GUI-level operations** via General Computer Control (GCC) (Tan et al., 2025c), **Semantic-designed action spaces** through API and skill encapsulation, and **Programmatic control** via automated code generation.

GUI-level Operations General Computer Control (GCC), a paradigm formalized by the Cradle framework, aims to build foundational agents that master computer tasks via a universal human-style interface—receiving inputs from screens and audio, and outputting native keyboard and mouse actions (Tan et al., 2025c). A representative design of such a GCC action space used by GameVerse is listed in Table.6 (Zhang et al., 2026a). Frameworks such as Cradle demonstrate that, by utilizing pure GUI actions, agents can navigate complex commercial games like Red Dead Redemption 2 without relying on any built-in APIs (Tan et al., 2025c). Similarly, benchmarks like FlashAdventure evaluate agents on completing full story arcs in diverse web-based games using mouse and keyboard interfaces (Ahn et al., 2025). While this paradigm offers ultimate generalizability across any software, it exposes a critical limitation of current Vision-Language Models (VLMs) known as the "knowing-doing gap" or "semantic-execution gap" (Zhang et al., 2025a; 2026a). As observed in evaluations, even when an agent correctly reasons about the optimal high-level strategy, it frequently fails to map this semantic intent to the precise action with the right parameters, such as coordinates. This profound disconnect between strategic reasoning and action execution leads to significant performance

Table 6: Example of GUI Action Space

Action Type	Description	Parameters
Mouse Movement		
MOVE_TO	Move to position	`x`, `y` (int, req): Target coordinates (window relative)
MOVE_BY	Move relatively	`dx`, `dy` (int, req): Offset; `duration` (float, opt): Seconds
Mouse Click		
CLICK	Click mouse	`x`, `y` (opt); `button` (left/right/middle); `num_clicks`
RIGHT_CLICK	Right click	`x`, `y` (int, opt): Click coordinates
DOUBLE_CLICK	Double click	`x`, `y` (int, opt): Click coordinates
Mouse Drag		
MOUSE_DOWN	Press button	`button` (str, opt): "left", "right", or "middle"; `duration` (float, opt)
MOUSE_UP	Release button	`button` (str, opt): "left", "right", or "middle";
DRAG_TO	Drag to target	`x`, `y` (int, req): Target coordinates
Mouse Scroll		
SCROLL	Scroll wheel	`dx`, `dy` (int, req): Scroll amount (+/-); `duration` (float, opt)
Keyboard Input		
TYPING	Type text	`text` (str, req); `interval` (float, opt)
PRESS	Press key	`key` (str, req); `duration` (float, opt)
KEY_DOWN	Key down	`key` (str, req): Key name; `duration` (float, opt)
KEY_UP	Key up	`key` (str, req): Key name;
HOTKEY	Key combo	`keys` (list, req): e.g., `['ctrl', 'c']`; `duration` (float, opt)
Control Flow		
WAIT	Wait time	`duration` (float, req): Seconds to wait
DONE	Task success	No parameters
FAIL	Task failure	No parameters

degradation, particularly in real-time or precision-heavy scenarios. Furthermore, GUI operations like `drag` and `hold_key` pose significant spatial and temporal challenges, as the drag trajectory and holding time require spatial and temporal intelligence, or even a combination of both. VideoGameBench noted that current foundation models perform poorly in the corresponding game environments (Zhang et al., 2025a).

Semantic-designed Actions Semantic actions are mostly pre-defined according to the common operational logic within the environment, utilizing conceptual or discrete parameters (e.g., target IDs or semantic locations) instead of precise pixel coordinates and physical distances (Zhang et al., 2026a; Wang et al., 2025a; Zhang et al., 2025f; Lin et al., 2024). For instance, in open-world MMORPGs, the TITAN framework abstracts the overwhelmingly vast continuous action space into high-level templates such as `Moveto(Location)` and `Attack(target)` (Wang et al., 2025a). Similarly, the Orak benchmark abstracts the exact frame-perfect joystick combinations required in fighting games (e.g., Street Fighter III) into interpretable, discrete semantic commands like "Move Closer" or character-specific skills like "Fireball" (Park et al., 2026). To systematically implement these semantic invocations, recent advancements leverage standardized tool-calling interfaces. In the PokéAgent Challenge and Orak, the semantic actions are wrapped into independent Model Context Protocol (MCP) server to achieve plug-and-play functionality (Karten et al., 2026; Park et al., 2026).

Programmatic Control Programmatic control decouples high-level reasoning from low-level execution by utilizing executable code as the action space. Unlike semantic APIs that rely on predefined functions, this paradigm allows foundation models to synthesize their own execution logic. For instance, to mitigate the inference latency associated with direct key-mouse control in fast-paced environments, the GameSense

framework utilizes VLMs to generate specialized Python scripts known as "Game Sense Modules" (GSMs). These scripts encapsulate real-time interactive logic and integrate external vision tools, running locally to manage tasks such as combat (Lu et al., 2025). In open-world settings, Voyager leverages the Mineflayer API to generate JavaScript control primitives, verifying and storing these code snippets in an extensible skill library for future reuse (Wang et al., 2024a). Expanding this approach to environment simulation, Code World Models (CWM) prompts LLMs to translate natural language game rules and trajectories into an executable Python world model. This synthesized code includes functions for state transitions, legal move enumeration, and termination checks, functioning as a simulation engine to facilitate planning algorithms like Monte Carlo Tree Search (MCTS) (Lehrach et al., 2025).

5.3 Reasoning

While perception modules extract environmental states and action modules execute commands, reasoning and planning modules serve as the central cognitive engine that bridges the two. Foundation models typically excel at rapid, intuitive responses but often struggle with the deliberate, multi-step logic required for complex strategic tasks. To elicit higher-order reasoning, recent research leverages various inference-time cognitive harnesses. Based on the structural design of the reasoning process, we categorize current Reasoning modules into three progressive paradigms: **Prompt Engineering**, **Workflow Orchestration**, and **Multi-Agent Systems**.

Prompt Engineering At the most fundamental level, reasoning capabilities are elicited through carefully designed prompt formulations. Techniques such as Chain-of-Thought (CoT) prompting explicitly guide the model to generate intermediate reasoning traces before executing an action. For instance, the GAMEBoT framework leverages domain-specific CoT prompts to guide LLMs in addressing predefined modular sub-problems prior to action selection (Lin et al., 2024). Similarly, DEPS provides few-shot demonstrations to prompt the model into generating self-explanations for complex open-world tasks (Wang et al., 2023a), while environments like `lmgame-Bench` are specifically designed to support modern reasoning paradigms by allowing models to be evaluated with or without long chain-of-thought (long-CoT) reasoning (Hu et al., 2025a). Furthermore, to steer strategic reasoning in adversarial or cooperative settings, techniques like Persona Prompting inject specific strategic profiles into the model to reshape its decision-making process, as demonstrated in strategic board games like PERIL (Licato & Steinle, 2025).

Workflow Orchestration To tackle long-horizon tasks and dynamic environments, raw reasoning is structured into systematic workflows. A foundational paradigm is ReAct, which interleaves reasoning traces with executable actions, allowing the agent to continuously ground its thoughts in environmental observations rather than relying purely on internal logic (Yao et al., 2022). For complex goals, explicit task decomposition is employed to break down macro-objectives into manageable sub-goals . For instance, DEPS incorporates an interactive planning process with a trainable goal selector to rank parallel candidate sub-goals (Wang et al., 2023a), while Voyager utilizes an iterative prompting mechanism that incorporates environment feedback, execution errors, and self-verification to build and refine an executable skill library (Wang et al., 2024a). Similarly, TITAN employs a plan-execute-memorize-diagnose-replan workflow that tracks action histories and state coverage to diagnose execution stalls and dynamically adjust testing strategies in complex MMORPGs (Wang et al., 2025a). Taking structural reasoning a step further, frameworks incorporate traditional search algorithms directly into the LLM's reasoning loop, such as PokéChamp, which utilizes LLMs for heuristic position evaluation to guide depth-limited minimax search (Karten et al., 2025).

Multi-Agent Systems Shifting the focus from individual intelligence to collective rationality, reasoning is distributed across multi-agent ecosystems. In complex environments, architectures often adopt an orchestrator with specialized sub-agents. The PokéAgent Challenge implements a central orchestrator that maintains high-level route plans while dynamically dispatching specialized sub-agents for battle strategy, puzzle-solving, and self-reflection (Karten et al., 2026). Beyond internal cognitive division, AgentVerse deploys multiple independent agents in Minecraft (Chen et al., 2024b). In this shared open-ended environment, this multi-player collaboration enables agents to efficiently tackle multi-step tasks while exhibiting emergent social behaviors, such as volunteering unallocated time and resources to accelerate team progress.

5.4 Real-time Reactivity

The inherent inference and execution latency of foundation models severely restricts their performance in real-time, dynamic environments. To mitigate this latency in interactive games, environment pausing is conventionally used to skip the reasoning time. For instance, Cradle, Orak, and VideoGameBench Lite pause the game during reasoning (Tan et al., 2025c; Park et al., 2026; Zhang et al., 2025a), while NitroGen intercepts the system clock of the game engine to freeze simulation time (Magne et al., 2026). To bypass the reliance on environment pausing, recent frameworks focus on decoupling reasoning from action execution. In GameSense, by shifting the VLM's role from a direct controller to a module developer, the VLM generates task-specific executable code, termed Game Sense Modules (GSMs), which autonomously take over high-frequency real-time operations (Lu et al., 2025). Besides, AgileThinker introduces a dual-thread architecture consisting of a planning thread for multi-step reasoning and a parallel reactive thread for outputting actions in real-time (Wen et al., 2025). Furthermore, to achieve real-time action generation, Lumine utilizes hybrid thinking to skip unnecessary reasoning and employs action chunking for high-frequency control, alongside latency optimizations including StreamingLLM, tensor parallelism, W8A8 quantization, and speculative decoding (Tan et al., 2025b; Xiao et al., 2024; Leviathan et al., 2023; Xiao et al., 2023).

5.5 Memory

Since foundation models are inherently stateless, memory modules are essential to store and maintain information for multi-step interactions of agents with dynamic environments (Sumers et al., 2024). Following the Cognitive Architectures for Language Agents (CoALA) framework, we systematically organize memory modules into **Working Memory** and **Long-Term Memory**.

Working Memory Working memory maintains active and readily available information, persists across foundation model calls, and enables the models to be stateful agents (Sumers et al., 2024). A common approach for working memory is keeping a fixed steps of recent states and actions using a sliding window, such as the memory modules in lmgame-Bench and GMH (Hu et al., 2025a; Zhang et al., 2025f). Furthermore, VistaWise designs the memory stack based on the concept of Last-In-First-Out (LIFO), allowing the agent to recall decisions from the most recent to earlier ones with controllable recall steps (Fu et al., 2025). Another form is history summarization, which maintains a fixed-length or compacted summary from history to avoid context overflow, as utilized in GameVerse and the PokéAgent Challenge (Zhang et al., 2026a; Karten et al., 2026).

Long-Term Memory Following the CoALA architecture, long-term memory can be systematically categorized into semantic, episodic, and procedural memory (Sumers et al., 2024). Semantic memory provides the agent with factual background knowledge about the environment. For instance, LPLH and VistaWise utilize knowledge graphs to store world knowledge, entities, and connections, dynamically evolving them with the agent's exploration (Zhang & Long, 2025; Fu et al., 2025). Similarly, to mitigate the lack of domain-specific knowledge, the PokéAgent Challenge injects usage statistics and competitive strategies directly from Smogon, an online Pokémon community (Karten et al., 2026). Episodic memory records the agent's past experiences and historical trajectories. To effectively maintain and retrieve these experiences, GameVerse employs a vector database (ChromaDB) to store the multimodal experiences that the agent decides to save (Zhang et al., 2026a). Finally, procedural memory stores the executable skills and action patterns. For example, Voyager maintains an ever-growing skill library of executable code to retain useful skills for future interactions (Wang et al., 2024a), while GameSense constructs a procedural memory database to store and retrieve historical action implementations and their corresponding execution codes (Lu et al., 2025).

5.6 Adaptive Learning

In a training-free paradigm, agents must adapt to dynamic environments entirely through their external harness rather than parameter updates. We categorize these adaptive learning mechanisms into three hierarchical levels: **Self-Reflection** for text-based experiential learning, **Evolving Skills** for executable action-level evolution, and **Prompt Optimization** for system-level instruction refinement.

Self-Reflection Reflection utilizes the in-context learning capacity of foundational models, prompting the agent to analyze historical information and generate reflective text, which is then injected into the next-turn prompt to guide future actions. For state-level diagnosis, Orak compares the current state, previous state, and executed actions to generate reflections on state transitions (Park et al., 2026). Similarly, TITAN generates diagnostic reflections by evaluating the abstracted state, current screenshots, action history, and task objectives (Wang et al., 2025a). To further augment the reflection process, recent frameworks incorporate expert experiences as reference baselines. Reflection of Episodes (ROE) combines pre-defined expert experiences with self-experiences based on keyframe selection, generating updated self-experiences through post-episode reflection (Xu et al., 2026a). Furthermore, GameVerse employs Vision-Language Models (VLMs) to simultaneously analyze the agent's failure videos and human expert tutorials, contrasting the two visual trajectories to generate concentrated experience prompts for policy refinement (Zhang et al., 2026a).

Evolving Skills To better adapt to dynamic game environments, agents can continuously learn and memorize new skills. For example, in Voyager, once a generated piece of code is self-verified to achieve its target goal, it is treated as a new skill and stored in an ever-growing skill library for future reuse (Wang et al., 2024a). Furthermore, Steve-Evolving advances this concept by structurally anchoring interactive experiences with fine-grained diagnostic signals, including state differences and enumerated failure causes (Xie et al., 2026b). Based on these anchored experiences, successful actions are distilled into reusable skills with explicit preconditions, while failed trajectories are distilled into executable guardrails to strictly prevent the agent from repeating similar mistakes.

Prompt Optimization Agents' performance heavily depends on the quality of prompts. To address this, the General Modular Harness (GMH) introduces a two-stage prompt optimization pipeline leveraging objective reward signals from the environment (Zhang et al., 2025f). In the first stage, GMH applies empirical prompt engineering to construct a baseline template that integrates trajectory histories (past states, actions, and rewards) and reflective summaries. In the second stage, it standardizes the optimization process using the DSPy framework and the SIMBA algorithm. By treating the game's cumulative reward as the objective function, this process iteratively refines the baseline prompt templates.

6 Benchmarks: The Diverse Arena for Game-Playing Agents

In terms of $\mathcal{M}$, a benchmark exposes chosen observation Ω and action A channels and turns the reward or outcome R into a score, probing competence under controlled variation of $\mathcal{M}$.

The rapid advancement of large language and vision-language models has brought a fundamental evaluation challenge to the forefront: how should we assess systems that are increasingly studied not merely as answer generators, but as agents that must act over time? While many widely used benchmarks remain valuable for testing bounded knowledge, logical reasoning, or one-shot multimodal understanding, they provide only partial evidence of the intelligence required for continuous interaction. True agentic competence emerges in closed-loop settings, where a system must track an evolving state, make decisions under uncertainty, adapt to feedback, and recover from mistakes while pursuing a goal across multiple steps. The central evaluation gap, therefore, is not simply that existing tests are too easy or too narrow, but that many of them abstract away the dynamic structure in which intelligent behavior must actually unfold.

Games provide a compelling substrate for this kind of evaluation because they preserve that dynamic structure in forms natively designed for human play. Rather than mere collections of themed tasks, games are carefully engineered challenge systems characterized by explicit rules, goals, feedback loops, state progression, and consequences that accumulate over time. These properties make games natural probes of human-relevant capability. They effectively expose a broad range of reasoning demands, including rule understanding, planning, memory, uncertainty handling, coordination, and learning from interaction, while also supporting visually grounded agent tasks in which success depends on sustained perception, action, and progress toward an objective (Wu et al., 2024b; Duan et al., 2024; Paglieri et al., 2025). In this sense, games are valuable not because they are simply "harder" than static benchmarks, but because they retain the interactive, goal-directed, and human-oriented structure that static benchmarks frequently eliminate.

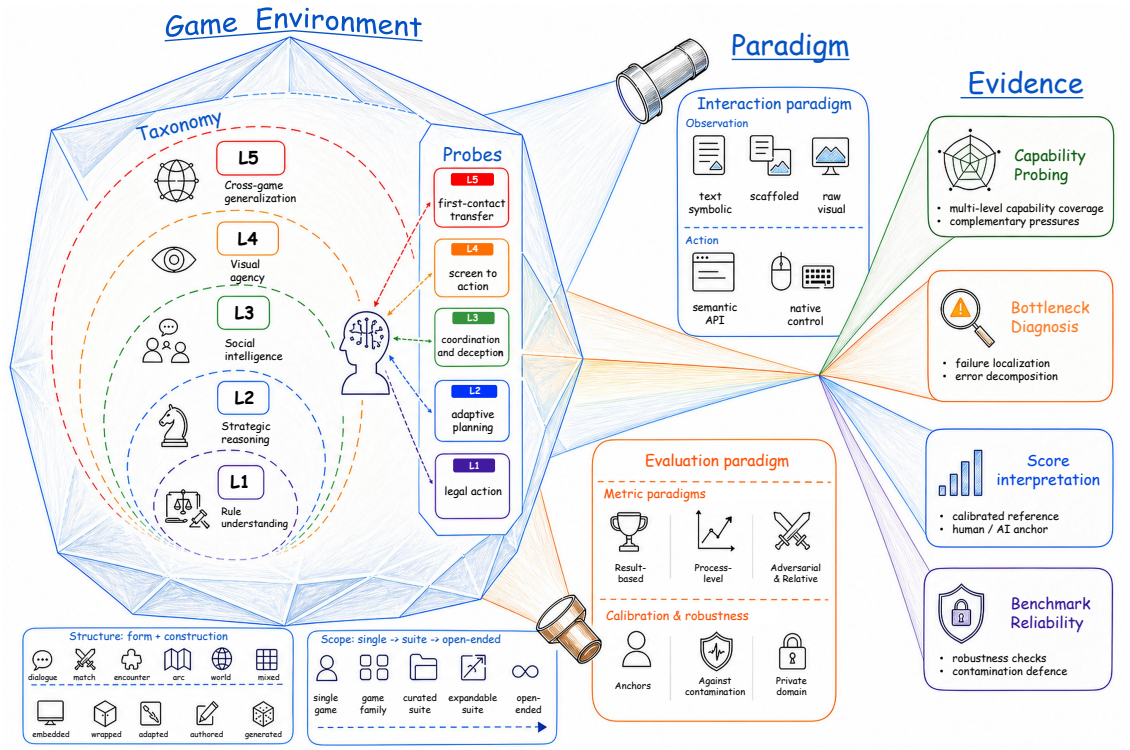

Figure 6: **Overview of conceptual framework for game benchmarks**. We frame game benchmarks through three connected dimensions: five-level environment organization, capability probing over agents, and the interaction and evaluation paradigms that shape gameplay into benchmark evidence.

This perspective also changes how we should understand game diversity. In the context of this survey, the value of games does not lie in assembling a loose catalog of genres, but in viewing the multiverse of human games as a structured space of capability coverage. Different game structures impose different demands on reasoning, memory, coordination, adaptation, and long-horizon control. Thus, diversity matters not as surface-level variation, but as a principled mechanism for applying complementary cognitive and operational pressures on AI agents. This is precisely why games are unusually promising as benchmarks: they span a broad, continually expanding, and difficult-to-saturate design space. As highlighted by AI GameStore (Ying et al., 2026), the point is not to identify a small set of canonical titles to solve once and for all, but to treat the multiverse of games as an extensible evaluation substrate for increasingly general, human-relevant intelligence.

However, a game environment does not automatically constitute a benchmark. It becomes one only when a protocol specifies how gameplay is presented, what qualifies as progress, and how behavior is converted into evidence. This section develops that argument through three dimensions:

- **Taxonomy** examines the types of game benchmarks the field has developed, organizing them into an evolutionary design space that spans rule-bounded settings, broader reasoning abilities, and increasingly open-ended environments.

- **Purpose** investigates what these environments are designed to measure, mapping different game structures to their motivating capability targets—ranging from rule grounding and strategic reasoning to social intelligence, visual agency, and cross-game generalization.

- **Paradigm** explores how gameplay is operationalized into benchmark evidence, analyzing the interaction interfaces and evaluation metrics that determine what a benchmark score truly represents.

Table 7: Taxonomy of representative game benchmarks

Paper	Date	Level	Form	Build	Scope	Input	Action	Game Set
Jericho (2020)	2019/09	L2 Strategy	Arc	Wrapped	Curated	Text	Semantic	56 IF games
NLE (2020)	2020/06	L2 Strategy	World	Wrapped	Single	Text	Semantic/native	NetHack
Crafter (2022)	2021/09	L4 Visual	World	Authored	Single	Visual	Native	22 achievements
CICERO (2022)	2022/11	L3 Social	Dialogue	Embedded	Single	Text	Semantic	Diplomacy
clembench (2023)	2023/05	L3 Social	Dialogue	Authored	Curated	Text	Semantic	7 dialogue datasets
SmartPlay (2024b)	2023/10	L1 Rule	Mixed	Wrapped	Curated	Text	Semantic	6 games
AvalonBench (2023)	2023/10	L3 Social	Dialogue	Adapted	Single	Text	Semantic	Avalon
LLM StarCraft II (2024)	2023/12	L2 Strategy	Match	Wrapped	Single	Text	Semantic	StarCraft II
CivRealm (2024)	2024/01	L2 Strategy	World	Wrapped	Open-ended	Text	Semantic	Civilization
GTBench (2024)	2024/02	L1 Rule	Match	Adapted	Curated	Text	Semantic	10 game-theory tasks
GAMABench (2025)	2024/05	L2 Strategy	Match	Adapted	Curated	Text	Semantic	8 scenarios
GameBench (2024)	2024/06	L2 Strategy	Match	Wrapped	Curated	Text	Semantic	9 games
Werewolf Arena (2024)	2024/07	L3 Social	Dialogue	Adapted	Single	Text	Semantic	Werewolf
BALROG (2025)	2024/11	L4 Visual	Mixed	Wrapped	Curated	Mixed	Semantic	6 env. families
TeamCraft (2024)	2024/12	L3 Social	World	Adapted	Single	Mixed	Semantic	Minecraft
GameArena (2025b)	2024/12	L3 Social	Dialogue	Adapted	Curated	Text	Semantic	3 live games
GAMEBoT (2024)	2024/12	L2 Strategy	Match	Wrapped	Curated	Text	Semantic	8 games
PokerBench (2025)	2025/01	L2 Strategy	Puzzle	Adapted	Single	Text	Semantic	11K poker spots
Collab-Overcooked (2025a)	2025/02	L3 Social	Encounter	Adapted	Single	Text	Semantic	30 tasks
DSGBench (2025)	2025/03	L2 Strategy	Mixed	Wrapped	Curated	Text	Semantic	6 games
LVLM Game Players (2025d)	2025/03	L4 Visual	Mixed	Adapted	Curated	Mixed	Semantic	6 games
LLM-Coordination (2025)	2025/04	L3 Social	Match	Adapted	Curated	Text	Semantic	4 games
TextArena (2025)	2025/04	L5 General	Mixed	Authored	Expandable	Text	Semantic	57+ games
KORGym (2025)	2025/05	L2 Strategy	Mixed	Authored	Curated	Mixed	Semantic	51 games
LMGAME-BENCH (2025a)	2025/05	L4 Visual	Mixed	Wrapped	Curated	Mixed	Semantic	6 games
VideoGameBench (2025a)	2025/05	L4 Visual	Mixed	Embedded	Curated	Visual	Native	23 games
MCU (2025b)	2025/05	L5 General	World	Generated	Open-ended	Visual	Native	3452 tasks
Orak (2026)	2025/06	L5 General	Mixed	Wrapped	Curated	Mixed	Semantic	12 games
TextAtari (2025c)	2025/06	L5 General	Encounter	Adapted	Curated	Text	Semantic	23 Atari games
TextQuests (2025)	2025/07	L5 General	Arc	Wrapped	Curated	Text	Semantic	25 IF games
StarDojo (2025a)	2025/07	L5 General	World	Adapted	Open-ended	Mixed	Semantic	1000 tasks
GVGAI-LLM (2025d)	2025/08	L5 General	Mixed	Generated	Expandable	Text	Semantic	118 games
FlashAdventure (2025)	2025/09	L4 Visual	Arc	Embedded	Family	Visual	Native	34 games
StarBench (2025b)	2025/10	L4 Visual	Encounter	Embedded	Single	Visual	Mixed	RPG battles
LLM-Hanabi (2025)	2025/10	L3 Social	Match	Adapted	Single	Text	Semantic	Hanabi
PuzzlePlex (2025)	2025/10	L2 Strategy	Puzzle	Authored	Curated	Mixed	Semantic	15 puzzle types
WOLF (2025)	2025/12	L3 Social	Dialogue	Adapted	Single	Text	Semantic	Werewolf
BotzoneBench (2026a)	2026/01	L1 Rule	Match	Wrapped	Curated	Text	Semantic	8 games
Strategic Hanabi (2026)	2026/01	L3 Social	Match	Adapted	Single	Text	Semantic	Hanabi
EMemBench (2026b)	2026/01	L4 Visual	Puzzle	Generated	Curated	Mixed	Mixed	16 games
MineNPC-Task (2026)	2026/01	L2 Strategy	World	Adapted	Single	Text	Semantic	Minecraft
AI GAMESTORE (2026)	2026/02	L5 General	Mixed	Generated	Expandable	Mixed	Native	100 games
Beyond Scaling (2026c)	2026/03	L2 Strategy	Match	Authored	Single	Text	Semantic	Zero-sum game
GameVerse (2026a)	2026/03	L5 General	Mixed	Wrapped	Curated	Mixed	Mixed	15 games
ARC-AGI-3 (2026)	2026/03	L5 General	Puzzle	Authored	Curated	Visual	Mixed	135 envs.
GameplayQA (2026b)	2026/03	L4 Visual	Puzzle	Adapted	Curated	Visual	Semantic	9 games
GameWorld (2026)	2026/04	L4 Visual	Mixed	Wrapped	Curated	Visual	Mixed	34 games
PokeGym (2026b)	2026/04	L4 Visual	World	Embedded	Single	Visual	Mixed	30 tasks

Together, these three sections move from benchmark structure, to capability target, to measurement logic. This framing provides a comprehensive understanding of how existing game benchmarks are organized and how the field has evolved. It also helps clarify how benchmark design can more effectively reveal model capabilities and limitations, thereby informing future model and benchmark development.

6.1 Taxonomy: The Evolutionary Levels of Game Environments

While conventional game genres describe games from the perspective of human play, a benchmark taxonomy must describe what an environment preserves, abstracts, and makes measurable for an AI agent. The same genre label can hide very different evaluation contracts, and very different games can impose similar benchmark pressures once they are converted into text states, semantic actions, GUI control, or generated task instances. We therefore treat game benchmarks as a multidimensional design space rather than as a genre catalogue. Its primary axis is an *evolutionary spine*, which situates benchmarks by their dominant ambition; three secondary axes then specify how each benchmark instantiates gameplay as an evaluable object: *Structure*, *Scope*, and *Modality*.

The primary axis of this design space is the evolutionary spine, which tracks the field's expanding benchmark ambition. *Level 1* corresponds to early benchmarks built around formal, rule-grounded interaction, evaluating whether an agent can understand rules, track state, and select legal or useful actions in formalized interfaces (e.g., Wu et al., 2024b; Duan et al., 2024). *Level 2* reflects a shift toward broader strategic decision-making in evolving environments, pushing the ambition from basic rule compliance to dynamic planning and interactive reasoning (e.g., Zhuang et al., 2025; Tang et al., 2025; Li et al., 2026c). *Level 3* marks

the integration of multi-agent social reasoning, where the evaluation target becomes the ability to cooperate, negotiate, or deceive within language-mediated arenas (e.g., Bailis et al., 2024; Agarwal et al., 2025). *Level 4* represents a leap toward ecological validity, shifting the environmental form to preserve more of the visual and interface burdens of human play while testing whether models can operate as grounded visual agents (e.g., Paglieri et al., 2025; Zhang et al., 2025b). Finally, *Level 5* extends the frontier from competence in a single environment to adaptability across broader game spaces, generated task distributions, or first-contact settings (e.g., Zhang et al., 2026a; Ying et al., 2026; Foundation, 2026). This progression highlights a continuous unified trajectory: as models evolve, the game environments used to benchmark them tend to abstract away fewer of the native complexities of human play, or make the remaining abstractions more explicit.

Structure specifies how gameplay becomes an evaluable unit through two fields: *Form* and *Construction*. Form identifies the playable unit on which progress, completion, and failure are defined. A *Match* has bounded contests and clear outcomes, as in formal and strategic game suites (e.g., Duan et al., 2024; Costarelli et al., 2024). A *Puzzle* isolates a constrained solving instance or question-like interactive problem (e.g., Long et al., 2025; Peper et al., 2026; Foundation, 2026). *Dialogue* benchmarks treat language interaction itself as the game loop, where debate, negotiation, reference, or persuasion drives the episode forward (e.g., Bailis et al., 2024; Chalamalasetti et al., 2023). *Encounter* benchmarks focus on bounded tactical segments, while *Arc* benchmarks evaluate longer story or quest progressions (e.g., Zhang et al., 2025b; Ahn et al., 2025). *World* benchmarks use persistent environments in which exploration, resources, tasks, or achievement graphs define progress (e.g., Zheng et al., 2025b; Qi et al., 2024; Long et al., 2024). *Mixed* applies when a suite intentionally spans several such playable units rather than fitting one dominant form (e.g., Wu et al., 2024b; Paglieri et al., 2025; Ouyang et al., 2026; Park et al., 2026).

Construction records the benchmark's relationship to the original game substrate. *Embedded* benchmarks preserve the native client or live play surface as much as possible, increasing ecological fidelity while also increasing execution noise and evaluation cost (e.g., Zhang et al., 2025b; Ahn et al., 2025). *Wrapped* benchmarks reuse existing games or environments through a shared harness, API, or sandbox, improving instrumentation and comparability while abstracting parts of the native interface (e.g., Wu et al., 2024b; Paglieri et al., 2025; Ouyang et al., 2026; Park et al., 2026). *Adapted* benchmarks convert existing games, rules, scenes, or logs into targeted evaluation tasks, such as game-theoretic settings, social-deduction arenas, or grounded rule questions (e.g., Duan et al., 2024; Bailis et al., 2024; Peper et al., 2026). *Authored* benchmarks create game-like environments specifically for measurement, while *Generated* benchmarks make new tasks, rules, levels, or game instances part of the benchmark's scaling strategy (e.g., Li et al., 2026c; Long et al., 2025; Ying et al., 2026; Zheng et al., 2025b; Li et al., 2025d). This distinction is central because ecological validity, repeatability, contamination risk, and measurement cost are all shaped by how far the benchmark artifact sits from the original game.

A separate axis is Benchmark *Scope*, which describes how a benchmark packages diversity to evaluate distinct operational scales. *Single game* benchmarks support deep diagnosis within one environment, often with clearer metrics and richer failure analysis (e.g., Bailis et al., 2024; Zhang et al., 2025b; Li et al., 2026c). *Game family* benchmarks expand within a relatively coherent family of related environments, maintaining recognizable interfaces and evaluation logic (e.g., Ahn et al., 2025). *Curated suites* intentionally assemble a fixed set of diverse games to evaluate general competence and cover multiple capability slices under a unified protocol (e.g., Wu et al., 2024b; Costarelli et al., 2024; Tang et al., 2025; Paglieri et al., 2025; Park et al., 2026). *Expandable suites* treat continuous growth as a core design principle, shifting the generalization focus from mastering a fixed game list to handling new instances, levels, or rules produced by the benchmark platform (e.g., Ying et al., 2026; Li et al., 2025d). Finally, *open-ended tasks* derive their breadth from large, combinatorial, or effectively unbounded task spaces within a persistent environment (e.g., Zheng et al., 2025b; Long et al., 2024; Qi et al., 2024). This axis prevents an important overclaim: broad coverage is not automatically cross-game generalization, and single-world task diversity should not be conflated with transfer across unrelated games.

The final axis is *Modality*, which records which parts of the human perception-action loop remain, are scaffolded, or are abstracted away in the benchmark. On the observation side, benchmarks may be *text-symbolic*, converting game states into natural language or structured data to maximize diagnostic clarity (e.g., Wu et al., 2024b; Duan et al., 2024; Tang et al., 2025; Li et al., 2025d); *visual*, using raw screenshots

or pixel streams to retain more of the native perception burden (e.g., Zhang et al., 2025b; Ahn et al., 2025; Zhang et al., 2026b); or *mixed*, providing multi-modal inputs or offering both raw and scaffolded tracks for comparison (e.g., Paglieri et al., 2025; Ouyang et al., 2026; Zhang et al., 2026a). On the action side, benchmarks may rely on *semantic controls*, where the agent outputs high-level commands, action tuples, or API calls (e.g., Wu et al., 2024b; Duan et al., 2024; Park et al., 2026), or *native control*, where the agent must execute human-like GUI, keyboard, or mouse actions (e.g., Zhang et al., 2025b; Ahn et al., 2025; Zheng et al., 2025b). At the taxonomic level, *Modality* is a factual coding of what burden the agent actually faces, not yet an explanation of how that burden changes score interpretation.

Taken together, these dimensions elevate the taxonomy table from a superficial genre catalog into a comprehensive design-space map. The five-level *evolutionary spine* traces the field's historical progression toward increasingly unconstrained generalization, while the *Structure*, *Scope*, and *Modality* axes deconstruct exactly how these environments are engineered. With this architectural framework established, the next section moves a step forward to address the core objective of these environments: what specific capabilities game benchmarks actually measure, and why they serve as a structured substrate for rigorous capability probes.

6.2 Purpose: Core Capabilities Evaluated by Games

The taxonomy above organizes game benchmarks by how they instantiate playable environments. Purpose shifts from that structural map to the capability claims those environments are meant to support. The same game substrate can become a different capability probe depending on which pressure the benchmark design foregrounds: rule exposure, state dynamics, information constraints, social interdependence, perception-action burden, or task novelty. This section explains why those structures can make particular capabilities visible, and where each kind of capability claim should remain bounded.

The central question is not whether games are harder than static tasks, but how a benchmark design makes a target capability necessary for play. Capability probing depends on design choices such as how rules are exposed, how states change, how opponents or collaborators constrain action, how much perception and control burden is preserved, and how task variation prevents success from collapsing into memorized routines. These are purpose-level questions: they explain why a game can serve as a probe for an ability.

Accordingly, each subsection below follows three guiding questions:

- Measurement target: What capability is the benchmark trying to put under pressure, and what should not be inferred from that target?
- Game affordance and boundary: Which properties of this game structure make the capability surface during play, and which aspects of real gameplay are abstracted away or only weakly tested?
- Probe design mechanisms: Which benchmark design choices make the target capability operationally necessary, such as rule-state-action formalization, information asymmetry, scenario selection, role structure, difficulty variation, scaffold contrasts, or generated task variation?

6.2.1 Level 1: Rule Understanding

At the first level, game benchmarks treat games as rule-governed action systems rather than as demonstrations of broad intelligence. The target is the model's ability to enter the game correctly: interpret rules or manuals, map the current state to those rules, maintain state across turns, produce legal actions, and satisfy the output or tool protocol through which the environment accepts moves. This is an entry condition for later capability claims. If a model cannot stay inside the legal action space or reliably update the state after its own moves, poor downstream performance should not yet be read as weak strategy, social reasoning, or visual agency (Wu et al., 2024b; Duan et al., 2024; Kolasani et al., 2025).

The strength of Level 1 settings is formal containment. Rules, states, actions, and outcomes can be made explicit enough that invalid moves, malformed outputs, tool-use failures, and weak move quality are separated instead of collapsed into a generic loss signal. Chess, grid-game, and game-theoretic benchmarks are useful precisely because simulators or engines can check participation while still leaving room for strategic choice.

The same clarity also defines the boundary of the evidence: many such benchmarks expose manuals, state variables, histories, or legal moves through text or API interfaces, removing much of the perceptual and interface-discovery burden of human play (Wu et al., 2024b; Duan et al., 2024; Kolasani et al., 2025; Topsakal et al.).

At the mechanism level, Level 1 benchmarks make rule understanding observable through three connected designs. Rule-state-action containment makes rules, current states, histories, and admissible actions explicit enough that rule interpretation, state maintenance, and legal-action generation become the minimum conditions for entering the game system (Wu et al., 2024b; Duan et al., 2024; Li et al., 2026a). Legal participation diagnostics then separate malformed outputs, illegal moves, tool-call failures, timeouts, and legal but strategically weak actions, preventing low scores from being prematurely interpreted as strategic failure (Kolasani et al., 2025; Topsakal et al.; Zhang et al., 2025e). Finally, situated rule-application probes ask whether models can bind rules to the particular state now in front of them, using rulebook modality, board or grid representations, valid-action questions, and short-horizon optimization tasks (Wang et al., 2025d). RuleOracles (Peper et al., 2026) makes this boundary especially clear: richer rulebook access can improve rule retrieval without guaranteeing correct rule application in a concrete game state.

6.2.2 Level 2: Reasoning

Level 2 shifts the target from legal participation to interactive decision quality. The question is not simply whether a model can produce a valid move, but whether it can sustain useful choices as states evolve, consequences are delayed, information remains incomplete, and opponents or environments respond. In this sense, reasoning is evaluated as a trajectory property: a plan must survive the action-consequence loop rather than remain plausible only as a one-step explanation (Duan et al., 2024; Costarelli et al., 2024; Tang et al., 2025; Li et al., 2026c).

The affordance of games at this level is that they turn reasoning into a sequence of commitments. A stated plan must survive contact with changing resources, opponent behavior, partial observability, spatial constraints, and accumulated earlier mistakes. This makes games useful probes of adaptive planning, opponent-aware choice, executable spatial reasoning, and long-horizon strategy. Yet the evidence remains bounded by the interface: many Level 2 benchmarks rely on textified states, API wrappers so they should be read as controlled probes of strategic pressure rather than direct evidence of human-like play or validated cognitive factors (Costarelli et al., 2024; Tang et al., 2025; Shi et al., 2025).

At the mechanism level, Level 2 benchmarks first use formal strategic pressure to make reasoning consequential. Game-theoretic environments (Duan et al., 2024; tse Huang et al., 2025) expose incentive structure, information regime, and opponent response in a controlled form, while specialist poker settings (Zhuang et al., 2025; Provost et al., 2026) add solver or strong-agent references for hidden-state decisions and action sizing. Coverage-oriented suites then broaden the pressure set across hidden information, stochasticity, cooperation, communication, adaptation, and planning under shared protocols, revealing uneven capability profiles rather than a single strategic score (Costarelli et al., 2024; Tang et al., 2025; Shi et al., 2025). Finally, spatial and temporal execution probes test whether plans remain effective as they become trajectories: maze and traversal tasks isolate map maintenance and loop avoidance, while real-time or strategy-world environments expose timing, resource allocation, state growth, and the gap between legal and useful actions (Einarsson, 2025; Nasir et al., 2024; Li et al., 2026c; Wang et al., 2026a; Qi et al., 2024).

6.2.3 Level 3: Social Intelligence

Level 3 treats social intelligence as interdependence in play, not as generic knowledge about social situations. A decision is good only relative to what other agents know, want, say, and may do next. The evidence is therefore behavioral: whether a model can turn social inference into consequential play, by identifying hidden roles, calibrating trust, coordinating with partners, or using language to shape later actions (Bailis et al., 2024; Light et al., 2023; Liang et al., 2025; Agashe et al., 2025).

Social games are useful because communication enters the causal loop of the game rather than remaining commentary about it. In deduction games, speech changes suspicion and votes; in cooperative games, a hint, request, or silence can determine whether partners converge on a shared plan; in Diplomacy-style

play, negotiation matters only when it remains consistent with executable orders. The boundary is equally important: most current benchmarks make this loop measurable by textifying dialogue, fixing roles or turns, adding rule-based moderators, or scoring labels and offline human references. They are strong probes of instrumented social gameplay, not direct proxies for unconstrained human social competence (Bailis et al., 2024; Liang et al., 2025; Meta Fundamental AI Research Diplomacy Team (FAIR) et al., 2022; Agarwal et al., 2025; Song et al., 2025).

At the mechanism level, designs make social reasoning consequential rather than merely reportable. Hidden-role settings turn private identity and role incentives into observable suspicion, voting, and deception dynamics; process-oriented variants then separate deception production, detection, calibration, and human-aligned judgment from final win rate (Light et al., 2023; Bailis et al., 2024; Agarwal et al., 2025; Song et al., 2025). Cooperative settings shift the pressure from deception to alignment: Hanabi, coordination games, and Overcooked-style tasks test whether agents infer partner knowledge and convert communication into joint action, with ToM scores, CoordQA, and initiating/responding metrics explaining failures beyond task success (Liang et al., 2025; Agashe et al., 2025; Sun et al., 2025a). Negotiation-heavy play then bridges language and strategy by showing that promises and persuasion matter only when coupled to plans that other agents can later act on, while also exposing the standardization cost of ecological human play (Meta Fundamental AI Research Diplomacy Team (FAIR) et al., 2022).

6.2.4 Level 4: Visual Agency

Level 4 targets the knowing-doing gap: whether a model can convert visual game states into effective action across time. The issue is not perception or planning in isolation, but their coupling with UI grounding, action localization, timing, memory, and recovery. A model may identify the right goal yet still fail by clicking the wrong region, misreading a status cue, repeating stale actions, becoming physically stuck, or losing track of quest progress (Zhang et al., 2025b;a; 2026b).

Visual games are useful because they preserve the screen-to-action loop that textified or API-mediated benchmarks often remove. The agent must decide from a changing visual surface and execute through an interface where small grounding errors alter later states. This makes static visual understanding insufficient: perception, control, and trajectory repair become observable behavioral failures. The boundary, however, is that ecological fidelity also makes scores harder to interpret. Low performance may reflect visual misrecognition, UI grounding, latency, action-format errors, planning weakness, memory collapse, or failed recovery. Level 4 is therefore best understood not as "more realistic is always better," but as the point where interface privilege determines which part of visual agency a benchmark can actually claim to measure (Zhang et al., 2025b; Paglieri et al., 2025; Ouyang et al., 2026).

Current benchmarks make this capability visible through three complementary designs. Perception-first probes remove or constrain control to show that gameplay-specific visual grounding and temporal attribution are already brittle before full agency is required (Wang et al., 2026b; 2025d; Zheng et al., 2025a). Matched-interface benchmarks then compare raw or computer-use control with semantic, tool-assisted, or harnessed variants, exposing how much apparent competence depends on the observation and action channel rather than on the game alone (Zhang et al., 2025b; Ouyang et al., 2026; Hu et al., 2025a; Zhang et al., 2026a). Finally, long-horizon GUI and 3D-world benchmarks turn local mistakes into trajectories, using milestones, progress scores, deadlock categories, or achievement structures to test whether agents can remember, recover, and keep advancing after errors accumulate (Ahn et al., 2025; Zhang et al., 2026b; Hafner, 2022).

6.2.5 Level 5: Cross-Game Generalization and Open-Ended Task Generalization

Level 5 should not be equated with "many games." Its target is whether an agent can preserve useful game competence beyond a fixed, known task: when rules, levels, mechanics, task combinations, or even the game substrate change. Rather than naming a single evaluation format, Level 5 marks a family of generalization pressures: retaining competence as the task space expands, as the form of play changes, or as the agent encounters unfamiliar mechanics for the first time (Park et al., 2026; Ying et al., 2026; Foundation, 2026).

Games are suitable for this purpose because their design space can expand in several non-equivalent directions. A benchmark may enlarge the task distribution inside one world, assemble heterogeneous games

under a common protocol, generate new rules or levels, or protect future tests through private and out-of-distribution splits. As a result, current claims about game-agent generalization are still methodologically uneven: the field is moving toward broader and less saturable evaluation, but different benchmarks operationalize that goal through different kinds of novelty, breadth, and openness.

Within this family, three mechanisms should be distinguished. Single-world open-ended benchmarks use rich environments such as Minecraft, Stardew Valley, or survival worlds to multiply goals, resource chains, social or collaborative situations, and recovery demands; they are strongest as evidence for intra-world task generalization, memory, exploration, repair, and long-horizon task composition, not for cross-game transfer (Zheng et al., 2025b; Tan et al., 2025a; Long et al., 2024; Hafner, 2022). Broad curated suites instead assemble multiple games or genres under a shared harness, asking whether a model, scaffold, or interface remains robust as the form of play changes; their main contribution is coverage across known game forms, so transfer claims should remain bounded by the fixed suite (Park et al., 2026; Zhang et al., 2026a; Phan et al., 2025; Li et al., 2025c; Guertler et al., 2025). Generated, living, or first-contact benchmarks push the pressure further by making new rules, levels, private games, or hidden mechanics part of the evaluation protocol; these designs are the closest current evidence for anti-saturation, mechanic induction, and first-contact adaptation, although the novelty may still be symbolic, procedural, or abstract rather than unrestricted commercial-game transfer (Li et al., 2025d; Ying et al., 2026; Foundation, 2026).

The five-level discussion above explains why different game structures make different capabilities necessary for play. Table 8 selects representative reported results from each level to show what current systems can and cannot do under each benchmark's own protocol. The Condition column states the relevant setting or reference before the Result column reports the key performance signal. The table shows a recurring pattern: current models can often enter controlled game systems and sometimes perform well under privileged text or semantic-action interfaces, but their performance drops sharply when success requires search-level planning, social information conversion, visual grounding, long-horizon recovery, or first-contact adaptation.

6.3 Paradigm: From Interaction to Assessment

A game environment does not automatically constitute a benchmark merely by being interactive or difficult. It becomes an evaluation instrument only when gameplay is mediated by an interaction contract and an evaluation contract. The former determines what part of the human play loop is exposed to the model, while the latter determines what kind of evidence can be extracted from the resulting behavior. In this sense, the methodological challenge of game benchmarking is not simply to select harder games, but to transform gameplay into a measurement pipeline that is interpretable, comparable, and sustainable as model capabilities evolve.

6.3.1 Interaction

Interface design operates as the front end of this evaluation pipeline. It specifies what the model is allowed to see, how it is allowed to act, and which cognitive or operational burdens of human play are preserved, simplified, or removed. A textified or API-mediated game can be a precise instrument for reasoning, while a visual native-control game can be a stronger test of end-to-end agency; neither is intrinsically superior unless its privilege level matches the claim being made. In this context, interaction design should be treated as a core part of benchmark validity rather than as an implementation detail.

Observation channels trace one of the clearest paradigm shifts in game benchmarks: the transition from abstract textual or symbolic state representations to raw visual streams. Textified interfaces align naturally with language-centric LLMs and excel in reasoning-oriented evaluations. By explicitly exposing rules, state variables, histories, and action constraints, they support highly controlled diagnostic probing (Wu et al., 2024b; Duan et al., 2024; Tang et al., 2025). However, scaling such interfaces to complex visual or commercial games necessitates bespoke APIs, wrappers, or state-extraction procedures. This fundamentally bypasses the challenge of inferring game states directly from the play surface.

Therefore, the shift toward raw visual input is not simply a change in modality; it is a commitment to more human-like, end-to-end agent evaluation. Models are forced to interpret screen pixels, track dynamic

Table 8: Reported performance signals of current models on representative game benchmarks

Level	Benchmark	Condition	Result	Signal
L1	SmartPlay (2024b)	HN = human-normalized score, with human baseline = 1.00.	GPT-4-0613 reaches HN 1.00 on Bandit and 0.91 on RPS, but drops to 0.83 on Hanoi, 0.61 on Minecraft, and 0.26 on Crafter	Textified rule play is feasible, but planning, recovery, spatial reasoning, and long-horizon control remain far below humans.
L1	GTBench (2024)	Opponent varies between random/MCTS/Tit-for-Tat and LLM opponents; NRA $\in [-1, 1]$ means normalized relative advantage against the opponent.	LLM agents pass the completion-rate sanity check with at least 90% completion overall, but in complete deterministic games against MCTS, all tested LLM agents report NRA $= -1$	Legal participation and strategic strength separate sharply; rule-following does not imply search-level play.
L2	KORGym (2025)	Text/API leaderboard reports average percentage over five reasoning dimensions; visual tasks are reported separately.	Text/API Avg.: o3-mini 82%, Gemini-2.5-Pro 79%, GPT-4o 22%. In visual tasks, Gemini-2.5-Pro reaches 90% on Bubble Ball but only 2% on Square Addition.	Standardized game APIs reveal reasoning profiles, while visual grounding remains uneven even in controlled game tasks.
L2	PuzzleJAX (2025)	BFS = breadth-first search with a 1M-step cap	BFS solves all levels in 8/12 example games. LLMs are mostly 0% across games; o3-mini reaches 100% on Slidings but only 50% on Sokoban Basic and 25% on Sokoban Match3.	Puzzle games still favor explicit search; current LLM/RL agents struggle with rule interactions and deadlock-avoiding plans.
L3	Collab-Overcooked (2025a)	Levels 1–6 increase collaboration complexity; SR = success rate, PC = progress completeness.	Claude Sonnet 4 SR falls from 100/100/96/92/74/58% across Levels 1–6; GPT-4o falls to 2% at Level 5 and 4% at Level 6. Humans under a 10s step limit still reach Level-6 SR 90% and PC 85.6.	Collaboration fails under complexity before basic task understanding disappears; humans remain far more stable.
L3	Werewolf Arena (2024)	No-information and trusted-Seer simulations provide anchors; Seer reveal metrics measure whether social information is disclosed, correct, and believed.	No-information villagers win only 1.2%; if the Seer automatically reveals a discovered Werewolf and villagers trust it, villager wins rise to 100%. In played games, believed Seer reveal rates vary sharply.	Social deduction depends not only on hidden information, but on when it is revealed, whether it is trusted, and whether persuasion converts knowledge into votes.
L4	Flash-Adventure (2025)	SR = success rate, MCR = milestone completion rate; COAST = native computer-use agent plus clue-memory Seek–Map–Solve scaffold.	Under a 1,000-step cap, COAST reaches highest SR 5.88%, and best MCR 19.89%, while human anchors reach SR/MCR 50.98/78.98% with the same cap and 97.06/100.00% without the cap.	Full-story GUI play remains far below human performance; clue memory improves progress slightly, but long-horizon observation–behavior gaps still dominate.
L4	LMGame-Bench (2025a)	No harness vs harness with perception, memory, and reasoning modules; Glass's δ = standardized distance from random baseline.	Without harness, 40% of non-text-only game runs fail to beat random; with harness, 86.7% beat random. Positive Glass's δ: 38/40 harnessed model-game pairs vs 26/40 without harness.	Harnesses can turn near-random play into discriminative scores, so reported ability is strongly scaffold-dependent.
L5	ARC-AGI-3 (2026)	Official = no harness on semi-private environments; RHAE = Relative Human Action Efficiency.	Humans solve 100% of included environments. The best official release score is only 0.37% RHAE.	First-contact mechanic induction is still largely unsolved.
L5	AI Game-Store (2026)	Scores are normalized to per-game human median and aggregated by geometric mean over 100 games.	Best reported model performance remains below 10% of the human-relative scale: GPT-5.2 is reported at 8.5/100. Across models, roughly 30–40% of games fall below 1% of the median human score.	Broad human-game coverage exposes a large human-relative gap, especially in memory, planning, and world-model learning.

visual changes, and make decisions under severe perceptual uncertainty. Unsurprisingly, current systems struggle in these settings, as raw visual observations reintroduce the state noise, grounding ambiguity, and temporal instability that text abstractions had neatly excised (Zhang et al., 2025a; 2026b; Wang et al., 2026b). To bridge this gap, hybrid designs—such as OCR-assisted, detector-assisted, or dual text-and-image tracks—occupy a critical middle ground. They recognize that while pure vision best approximates human play, providing intermediate perceptual scaffolding yields a more diagnostic benchmark, isolating higher-order reasoning failures from low-level perceptual bottlenecks (Zhang et al., 2025b; Park et al., 2026).

Action channels determine how a model's decisions manifest within the game environment. Semantic actions regularize the agent's output space by restricting choices to high-level moves, skills, targets, or predefined commands that are seamlessly parsed and executed by the benchmark harness (Wu et al., 2024b; Duan et al., 2024; Tang et al., 2025; Park et al., 2026). By eliminating the friction of motor execution and UI manipulation, this design directs evaluation toward pure reasoning, planning, and decision quality. Native control, by contrast, requires the model to interact through interfaces originally designed for human players, such as mouse, keyboard, controller, or GUI actions. This elevates the agentic requirements of the task: the model must map an intended strategy onto executable and time-sensitive interactions. Furthermore, native-control interfaces provide a scalable pathway for evaluating commercial and browser games. By leveraging standardized, human-facing control channels, researchers can bypass the prohibitive engineering cost of developing custom symbolic APIs for every new game environment (Zhang et al., 2025a; Ahn et al., 2025; Zhang et al., 2025b). Dual-interface benchmarks show why this distinction is substantive rather than cosmetic: the same or similar gameplay can look much easier when semantic actions replace low-level control (Zhang et al., 2026a; Ouyang et al., 2026).

Observation and action choices together form an interface-privilege hierarchy. High-privilege settings are valuable when the research question is strategic reasoning, rule application, or diagnostic comparability; low-privilege settings are valuable when the target is end-to-end gameplay which preserve visual grounding, UI operation, timing, and recovery. The risk is overclaiming across that boundary. Scores in high privilege settings weakens claims about whether a model can understand and act within an unconstrained environment as presented to humans, while a low score in a raw visual benchmark may conflate planning failure with perception, latency, control formatting, or recovery failure (Zhang et al., 2025a; 2026b; 2025b; Ouyang et al., 2026). A rigorous game benchmark must explicitly document its privilege level and, ideally, provide comparative tracks between assisted and human-like settings to successfully disentangle reasoning competence from grounding and execution failures.

6.3.2 Evaluation

The evaluation contract determines how agent behavior is synthesized into evidence. Games naturally contain goals, rewards, progress, and failure states, but these native signals do not automatically constitute a rigorous benchmark. A game becomes a benchmark only when its outcomes are translated into metrics that are interpretable, comparable, and robust against the noise of stochastic interactions.

The simplest contract is result-based evaluation. Win rate, score, reward, survival, completion, and normalized progress preserve the native objective structure of play and are easy to automate at scale, which explains their prevalence from textified strategic suites to broader agent benchmarks (Wu et al., 2024b; Duan et al., 2024; Paglieri et al., 2025; Ouyang et al., 2026). Their weakness is diagnostic opacity. The same low score may reflect poor planning, rule misunderstanding, visual misrecognition, invalid actions, latency, memory failure, or weak recovery. This problem becomes especially severe in visually grounded or long-horizon games, where current models often achieve extremely low native progress and the metric loses diagnostic resolution (Zhang et al., 2025a;b). Furthermore, native scores suffer from weak semantic portability: a score in one game rarely rarely maps to the same score in another.

Process-level evaluation addresses that opacity by instrumenting the trajectory rather than only the endpoint. These metrics decompose gameplay into intermediate milestones, trajectory analysis, sub-skill scores and reasoning checks. Validated trajectory scores can reveal whether agents fail through bad decisions, lost state, weak control, or poor recovery (Lin et al., 2024; Tang et al., 2025; Ouyang et al., 2026; Zheng et al., 2025b; Zhang et al., 2026a). This makes process metrics especially valuable for long-horizon and open-ended

games, where binary success hides most of the behavior. The trade-off is that decomposition is itself a design choice: once a benchmark decomposes gameplay into subskills or milestones, the decomposition itself reflects the benchmark creator's prior assumptions about positive behavior. Process metrics are therefore best treated as an explanatory layer around outcomes, not as a universal cross-game currency.

Relative and adversarial evaluation adds a different kind of evidence. By evaluating agents against other models, fixed opponents, or live human participants, these protocols expose capabilities that only appear in interaction, such as opponent modeling, deception, negotiation, and adaptation (Duan et al., 2024; Bailis et al., 2024; Hu et al., 2025b). Consequently, adversarial frameworks maintain benchmark freshness and capture dimensions of social intelligence that isolated, single-agent tasks inherently miss. Yet the resulting scores are relational by construction. A tournament rank or arena win rate reports performance against a particular opponent pool under a particular protocol. Adversarial evaluation is therefore powerful for dynamic stress testing, but weak as an absolute measure unless its competitive context is made explicit.

Calibration is therefore necessary for giving game scores external semantics. By introducing fixed AI anchors, explicitly tiered opponent ladders, or calibrated solver references, benchmark designers can transform fluctuating relative performances into interpretable claims (Wu et al., 2024b; Li et al., 2026a; Foundation, 2026; Ouyang et al., 2026). This ensures that the same numerical score is not overinterpreted across different games or varying interface privileges within a suite. A 60% win rate, a progress score, or an Elo-like rating is meaningful only relative to the baseline, interface privilege, and task distribution that produced it.

Robustness then asks whether that meaning survives reasonable variation and benchmark pressure. Stronger protocols now use repeated runs, seeded duplicate matches, private or held-out environments, refreshed instances, contamination checks, scaffold ablations, and public/private leaderboard separation to reduce noise, leakage, and benchmark-specific optimization (Foundation, 2026; Hu et al., 2025a; Beyer et al., 2024; Ouyang et al., 2026). The important point is not that any single defense solves robustness, but that evaluation design must report which threats it addresses and which remain. This is particularly important in game benchmarks, where interaction traces, agent scaffolds, and public gameplay knowledge can all become part of the measured system.

Taken together, these paradigms show that game evaluation should be read as a measurement stack rather than a scoreboard. Outcome metrics preserve the game's objective; process metrics explain the route through the game; adversarial protocols introduce interactive pressure; calibration supplies external reference points; and robustness protects those interpretations over time. The most rigorous game benchmarks are therefore not simply those built on difficult games, but those that make explicit what their scores can and cannot support as evidence about agent capability.

7 Challenge and Future

The preceding sections have traced the full pipeline of large foundation models in the multiverse of games, from datasets through models and harness to benchmarks. At every stage, a consistent pattern has emerged: progress on one axis exposes a bottleneck on another. This section distills these recurring bottlenecks into five fundamental trade-offs that define the current frontier, and then charts a five-level roadmap toward the generalist game player envisioned by this survey.

7.1 Five Fundamental Trade-Offs

The challenges facing game-playing foundation models are not isolated problems awaiting point solutions. They are structural tensions in which improving one side inevitably stresses the other. We identify five such trade-offs, each grounded in evidence from the preceding sections.

7.1.1 Scale vs. Fidelity vs. Diversity: The Data Trilemma

Section 3 established that no existing dataset simultaneously leads in scale, annotation quality, and game diversity. NitroGen (Magne et al., 2026) assembles 40,000 hours across 1,000+ games through automated controller-overlay extraction, but its button accuracy of 0.96 and joystick R^2 of 0.84 mean that roughly one

Figure 7: **Overview of five-level roadmap towards generalist game player.** The progressive roadmap illustrating the evolution from narrow, single-game mastery (Level 1) to fully generalist agents capable of cross-task transfer within genres (Level 2), cross-genre generalization (Level 3), and lifelong adaptation to unseen environments in human-craft game universe (Level 4), culminating in the "Demiurge" stage (Level 5), where agents transcend gameplay to generate, simulate, and evolve entire game worlds. This hierarchy reflects an increasing degree of agency, from solving tasks to constructing and expanding the game multiverse.

in twenty-five discrete actions and one in six continuous inputs are wrong. These errors compound over long trajectories, precisely the regime where fidelity matters most. At the other extreme, OpenP2P (Yue et al., 2026) invests 8,300 hours of meticulous human annotation across 45 games and demonstrates that behavior cloning follows a predictable scaling law: increasing both model capacity and data volume leads to the emergence of causal reasoning. Yet the linear cost of human annotation makes NitroGen-scale labeling economically prohibitive. Meanwhile, VPT's (Baker et al., 2022) 70,000 hours of IDM-pseudo-labeled data cover only Minecraft, and PLAICraft's (He et al., 2025b) five-modality temporally aligned corpus, though high in quality, is equally confined to a single game.

The tension is not merely practical. OpenP2P (Yue et al., 2026) shows that annotation quality enables qualitative capability jumps; NitroGen(Magne et al., 2026) shows that data scale enables cross-game transfer. Neither substitutes for the other. Bridging this gap will likely require either learned annotation models that approach human fidelity at machine cost, or world-model-based data engines that generate trajectories with built-in ground-truth labels.

7.1.2 Breadth vs. Depth: The Heterogeneity Wall

A human player adapts to a new game within minutes. Current models cannot. The core obstacle is game heterogeneity: the same keyboard key carries entirely different semantics across games. "W" means move forward in an FPS, build a worker in an RTS, and nothing at all in a card game. This action-space fragmentation means that scaling to more games does not automatically yield a better agent in any single game.

The evidence is consistent. Per-game specialists reach strong performance ceilings: Metamon (Grigsby et al., 2025) achieves human-level Pokemon through offline RL on a decade of ranked replays; CombatVLA (Chen et al., 2025c) attains human-level ARPG combat with sub-15ms action alignment. Cross-game systems cover far more titles but at substantially lower competence: NitroGen (Magne et al., 2026) spans 1,000+ games yet produces only reactive motor mimicry without goal-directed behavior; Game-TARS (Wang et al., 2025f) trains on 500+ games and approaches fresh-human performance on unseen web games, but requires game pausing during reasoning in fast-paced settings. Lumine (Tan et al., 2025b) represents the most promising middle ground, achieving zero-shot transfer of multi-hour storyline completion across structurally similar anime RPGs (Genshin Impact, Honkai: Star Rail, Wuthering Waves), but this transfer is confined to games sharing similar UI conventions and interaction patterns, and has not been demonstrated on mechanically distinct titles, such as Black Myth: Wukong.

Attempts to unify action representations, whether through NitroGen's (Magne et al., 2026) 20-dimensional gamepad vector or OmniJARVIS's (Wang et al., 2024b) FSQ tokenization, inevitably lose game-specific precision. The field has not yet found an action abstraction that preserves enough information for expert play while remaining general enough for cross-game transfer.

7.1.3 Reasoning vs. Reactivity: The Latency-Intelligence Dilemma

Strategic games and action games impose opposite demands on the same model. CICERO's (Meta Fundamental AI Research Diplomacy Team (FAIR) et al., 2022) multi-turn Diplomacy campaigns and LSPO's (Xu et al., 2025) equilibrium-finding in Werewolf require deep, multi-step reasoning that consumes hundreds of tokens. FPS and MOBA games require responses within 100 milliseconds.

Current systems cannot satisfy both demands. On VideoGameBench (Zhang et al., 2025a), the strongest frontier VLM, Gemini 2.5 Pro (Comanici et al., 2025), completes only 0.48% of game tasks when operating in real time; the benchmark had to introduce a game-pausing mode to reach even 1.6%. Existing solutions trade one end for the other: CombatVLA (Chen et al., 2025c) truncates its Action-of-Thought chain to achieve a 50× speedup but sacrifices interpretability and strategic depth; Lumine (Tan et al., 2025b) reasons at 5Hz and acts at 30Hz by invoking deliberation only when needed; NitroGen's (Magne et al., 2026) flow-matching DiT generates 16-step action chunks in parallel, eliminating the autoregressive bottleneck but also eliminating reasoning entirely. Game-TARS (Wang et al., 2025f) introduces a "Greedy Thinking" strategy, but finds that excessive reasoning in fast-paced games triggers hallucinated reasoning loops that degrade performance below the no-reasoning baseline.

The fundamental issue is architectural. Autoregressive language models couple reasoning depth to inference latency: more thinking means more tokens means more time. No current architecture provides a principled mechanism for an agent to deliberate deeply on strategic decisions and react reflexively to split-second events within a single forward pass.

7.1.4 Modular Workflow vs. Model-as-Whole: The Harness Paradox

Section 5 documented the harness as the nervous system connecting foundation models to game environments. The paradox is that current models need this external scaffolding to function, yet the scaffolding itself introduces fragility and limits generality.

End-to-end models struggle without harness support. On VideoGameBench (Zhang et al., 2025a) and GameVerse (Zhang et al., 2026a), frontier VLMs applied directly to game screenshots fail to progress beyond the opening minutes of most games. By contrast, modular harnesses achieve qualitatively stronger results: Cradle (Tan et al., 2025c) completes 40-minute missions in Red Dead Redemption II by composing perception, memory, planning, and action modules around a VLM core. Voyager (Wang et al., 2024a) builds an open-ended skill library in Minecraft through retrieval-augmented generation and self-verification. The gap demonstrates that capabilities such as persistent memory, self-correction, and low-latency reflexes remain beyond the native capacity of current foundation models. Yet modularity has costs. Each module introduces extra engineering, and the interfaces create information bottlenecks. Section 5.2 documented the "knowing-doing gap": even when an agent correctly identifies the optimal strategy, it frequently fails to translate that

strategy into the precise action parameters required. This gap between semantic understanding and motor execution is a direct consequence of the modular boundary between the reasoning and action components.

The ideal resolution is a model that natively possesses memory, reflection, and fine-grained control, making the external harness unnecessary. Current VLAs represent a step toward this goal by unifying perception and action, but they still lack persistent memory and self-correction. Closing this gap is a prerequisite for moving from game-specific agent pipelines to a universal game player.

7.1.5 Code Engine vs. World Model: The Simulation Gap

Most training and evaluation depend on code-based game engines. These engines provide solid physics, deterministic signals, and arbitrarily long rollouts, but they impose three ceilings: the action space is limited to predefined interfaces, the game diversity is limited to what humans have built, and each new game requires dedicated integration effort. World models offer a path beyond these ceilings. Genie (Bruce et al., 2024) discovers latent action spaces from unlabeled video, removing the interface constraint. GameFactory (Yu et al., 2025) and GameGen-X (Che et al., 2025) generate interactive environments beyond the boundaries of existing games, pushing toward the theoretical multiverse of Era 4. PAN (PAN Team Institute of Foundation Models, 2025) extends action conditioning to natural language, and Solaris (Savva et al., 2026) introduces multi-agent shared worlds with cross-player consistency. Most significantly, SIMA 2 (SIMA-team et al., 2025) has been successfully positioned inside Genie 3 generated worlds (Ball et al., 2025) and showed positive transfer to held-out tasks, providing the first evidence that world models can serve as viable training environments.

However, current world models remain far from replacing code engines. Oasis (Decart & Julian Quevedo, 2024) maintains consistency for only a few seconds before errors compound into visible drift. Genie 3 (Ball et al., 2025) generates playable environments but is limited to 60-second rollouts with imperfect physics adherence. GameNGen (Valevski et al., 2025) sustains human-indistinguishable quality for several minutes in DOOM, but only for a single, visually simple game. More fundamentally, world models lack the deterministic reward signals that code engines provide. When the environment itself is generated by a neural network, the definition of success becomes ambiguous: how does one verify that an agent has "won" a game whose rules are themselves approximate?

Bridging this gap requires advances on three fronts: temporal consistency over rollouts of thousands of steps, verifiable reward signals within model-generated environments, and multi-game world models that can instantiate diverse game universes rather than imitating a single title.

7.2 Five-Level Roadmap to the Future

The trade-offs above define why the challenges is hard. This section defines what progress looks like. We organize the path toward the generalist game player into five levels of increasing generalization, from mastering tasks within a single game to becoming the game environment itself. At each level, we identify the current frontier, its limits, and the trade-offs that must be resolved to advance further.

7.2.1 Level 1: Single-Game Task Mastery

"Complete all tasks within a single game, from atomic actions to long-horizon objectives."

In developmental psychology (Fitts & Posner, 1967; Anderson, 1982), skill acquisition within a single rule system is the foundation of all higher-order transfer. A child must first master the rules, controls, and feedback loops of one game before any cross-game generalization becomes meaningful. For AI agents, this corresponds to achieving robust competence across the full task distribution of a single game environment, including both short-horizon atomic skills and long-horizon composite objectives that chain these skills over hundreds of sequential decisions.

The most extensively studied game environment is Minecraft, where over seven years of sustained research have produced an unparalleled data-model ecosystem (MineRL (Guss et al., 2019), MineDojo (Fan et al., 2022), VPT (Baker et al., 2022), GROOT (Cai et al., 2024), OpenHA (Wang et al., 2025e), JARVIS-VLA (Li et al., 2025a), PLAICraft). Yet even in this richest ecosystem, single-game mastery remains incomplete. OpenHA's inference speed of 0.98 FPS is far below real-time play. Out-of-distribution tasks cause performance drops of up to 30 percentage points. Combat scenarios exhibit high variance ($\pm 43.5\%$), with the agent unable to track fast-moving targets. Most critically, no existing system has attempted a full long-horizon objective such as progressing from an empty world to defeating the Ender Dragon, a task requiring thousands of sequential decisions spanning resource gathering, crafting, exploration, and combat. Even in the most mature game ecosystem, the gap between completing atomic tasks and mastering the full game experience remains wide.

Advancing to single-game mastery requires resolving Trade-off 3, reasoning vs. reactivity for long-horizon planning and Trade-off 4, harness-supported memory and self-correction vs. end-to-end efficiency.

7.2.2 Level 2: Cross-Task Transfer Within Genre

> *"Transfer learned competence across games that share visual style, interface conventions, or interaction patterns."*

Analogical reasoning research (Gentner, 1983; Gick & Holyoak, 1983) shows that humans transfer skills most readily between domains that share surface features and relational structure. A player fluent in one open-world RPG navigates a new RPG almost immediately, because they share the similar interaction grammar. Level 2 tests whether AI agents can exploit this same structural overlap: given mastery of one game, can they generalize to a second game within the same genre without retraining?

Lumine (Tan et al., 2025b) demonstrates the current frontier at this level. Trained on the Mondstadt region of Genshin Impact, it achieves zero-shot completion of the five-hour main storyline in Honkai: Star Rail and a 100-minute mission in Wuthering Waves. Its 5Hz perception and 30Hz action pipeline operates in real time without game-specific adaptation, and its selective reasoning mechanism activates deliberation only when needed.

The limits are equally clear. Simple instruction-following tasks succeed at over 80%, but performance drops sharply on puzzles, flying enemies, and non-flat terrain. Approximately half of all errors stem from failures in multimodal understanding, particularly the detection of small or environmentally blended objects. More fundamentally, these three games share a common genre of anime-style open-world RPGs with similar minimap layouts, dialogue systems, and quest markers. Transfer to mechanically distinct titles such as action RPGs with different combat systems, inventory designs, and camera conventions has failed. And even within the supported genre, multi-hour storyline completion covers only a fraction of the hundreds of hours of content that each game offers.

Breaking through Level 2 requires addressing Trade-off 2 (breadth vs. depth): the agent must generalize beyond UI-similar games without losing the depth needed for complex in-game tasks.

7.2.3 Level 3: Cross-Genre Generalization

> *"Operate across games with fundamentally different action spaces, visual styles, time scales, and mechanics."*

This level corresponds to what game studies call "ludic literacy" (Juul, 2005; Salen & Zimmerman, 2003): the ability to parse an unfamiliar rule system by recognizing abstract patterns (resource management, spatial navigation, turn economy) beneath surface-level differences. A chess expert picking up a new strategy board game does not start from zero, because the underlying combinatorial reasoning transfers even when pieces, boards, and victory conditions change entirely. For AI, Level 3 demands that a single model handles games

in which $\mathcal{A}$, $\mathcal{O}$, and $\mathcal{T}$ are all different, requiring an invariant representation of agency that abstracts over game-specific details.

Several systems in this level have reached varying degrees of competence. Game-TARS (Wang et al., 2025f), pre-trained on 500B+ tokens from 500+ games, doubles prior state-of-the-art performance on Minecraft, exceeds GPT-5 and Gemini-2.5-Pro (Comanici et al., 2025) on ViZDoom FPS maps, and approaches fresh-human performance on unseen web games. NitroGen (Magne et al., 2026) trains a DiT-based architecture on 40,000 hours of gameplay across 1,000+ titles, demonstrating that heterogeneous multi-game corpora can support generative pre-training. OpenP2P (Yue et al., 2026) validates behavior-cloning scaling laws across 45 3D games and observes the emergence of causal reasoning with increasing model and data scale.

However, competence at this level is qualitatively shallower than at Levels 1 and 2. NitroGen (Magne et al., 2026) produces plausible motor patterns, such as combat reactions and navigation, but has no capacity for goal-directed planning, language understanding, or strategic reasoning. Game-TARS requires game pausing during reasoning steps, and its "Greedy Thinking" strategy triggers hallucinated reasoning loops in fast-paced settings. OpenP2P (Yue et al., 2026) follows only simple instructions and exhibits behavioral artifacts including wall collisions and off-target shooting. The consistent pattern is that as game coverage increases, the level of mastery in each game decreases. No current system achieves both broad coverage and deep competence.

Advancing beyond Level 3 requires resolving Trade-off 1, large-scale, high-fidelity data for diverse games and Trade-off 2, a unified action representation that preserves game-specific precision.

7.2.4 Level 4: Lifelong Adaptation

> *"Rapidly adapt to entirely new environments through self-directed exploration and continuous self-improvement."*

Levels 1 through 3 evaluate competence on a fixed distribution of games seen during training or closely related to them. Level 4 introduces a qualitatively different requirement: ***online learning in an unknown environment.*** This mirrors what cognitive scientists call "learning to learn" (Harlow, 1949; Flavell, 1979), the meta-cognitive ability to form hypotheses about new rules, test them through exploration, and revise them from sparse feedback. A human player dropped into a game with no manual does exactly this: within minutes, she infers the control mapping, the objective structure, and the penalty conditions through trial and error. Current foundation models, by contrast, are static after training and cannot update their policies from a handful of in-game experiences.

SIMA 2 (SIMA-team et al., 2025) provides the clearest prototype for this level. Its bootstrapped improvement cycle begins with human demonstrations augmented by Gemini-generated labels, transitions to self-directed play in unseen games without human data, and uses its own experience to train successive agent generations. On held-out environments never seen during training, SIMA 2 roughly doubles the success rate of its predecessor, completing 26 out of 50 MineDojo task categories (versus 2 for SIMA 1 (Raad et al., 2024)) and autonomously progressing through 15 to 20 minutes of a previously unseen story-driven game, demonstrating skills that were never present in its training data.

The gap to true lifelong adaptation remains substantial. SIMA 2 (SIMA-team et al., 2025) still requires a full SFT and RL training phase before self-improvement becomes possible; it is not a zero-shot or few-shot learner. Its context window limits working memory, causing performance degradation on tasks requiring long-horizon reasoning. Precise motor control, particularly in combat requiring split-second timing, remains a weakness. And the self-improvement loop depends on Gemini-based feedback, inheriting the reasoning costs and failure modes of the evaluator model. The broader challenge is that current foundation models lack the mechanisms for rapid online adaptation: they cannot update their behavior from a handful of in-game experiences in the way a human player adjusts strategy after a single failed attempt.

Reaching Level 4 requires resolving Trade-off 1 to 4. An agent who reaches Level 4 is a really generalist player.

7.2.5 Level 5: The Demiurge, From Player to Creator

"Become the environment itself: generate, reshape, and evolve game worlds."

The preceding levels concern agents that play within fixed game environments. Level 5 envisions a qualitative shift in the agent's ontological role. In game design theory (Huizinga, 1938; Salen & Zimmerman, 2003), the distinction between player and designer is fundamental: a player optimizes within the rules; a designer constructs the rules themselves. This is also the distinction between intelligence that adapts to a given world and intelligence that creates worlds. Within the POMDP formalism, the agent no longer optimizes a policy within a predefined game $\mathcal{M}$, but actively generates the state space $\mathcal{S}$, action space $\mathcal{A}$, transition dynamics $\mathcal{T}$, and reward structures $\mathcal{R}$. This is the Era 4 Demiurge outlined in Figure 2.

Early steps toward this vision have begun to converge. Genie 3 (Ball et al., 2025) generates diverse, interactive 3D environments in real time across multiple locomotion modes, producing worlds that are navigable by both humans and autonomous agents. SIMA 2 (SIMA-team et al., 2025) has explored inside Genie 3-generated worlds and demonstrated positive transfe, providing the first closed-loop evidence that a world model can serve as both training environment and evaluation substrate for an embodied agent.

The distance to the full vision remains large. Genie 3 (Ball et al., 2025) is limited to 60-second rollouts, lacks persistent world state, and does not guarantee physical consistency. No current world model supports the generation of verifiable game rules, win conditions, or reward functions. The multi-agent setting, in which multiple agents co-inhabit and co-shape a generated world, has been explored only by Solaris (Savva et al., 2026) and remains at the proof-of-concept stage. Realizing the Demiurge requires bridging all five trade-offs simultaneously: large-scale and high-fidelity data to train the world model (Trade-off 1), broad and deep game understanding to generate diverse yet coherent worlds (Trade-off 2), real-time generation with long-horizon consistency (Trade-off 3), native model capabilities that replace external scaffolding (Trade-off 4), and world models that match code-engine reliability while surpassing their diversity (Trade-off 5).

The five levels above are not merely a ladder of increasing difficulty. They represent a progression in the degree of agency: from executing predefined tasks, through adapting to new contexts, to constructing the contexts themselves. Each level subsumes the capabilities of those below it. An agent that can generate and evolve game worlds (Level 5) must also be able to adapt to new environments (Level 4), generalize across genres (Level 3), transfer within a genre (Level 2), and master individual games (Level 1). In this sense, the roadmap defines both the incremental milestones and the ultimate destination of the future.

References

Mrinal Agarwal, Saad Rana, Theo Sundoro, Hermela Berhe, Spencer Kim, Vasu Sharma, Sean O'Brien, and Kevin Zhu. Wolf: Werewolf-based observations for llm deception and falsehoods, 2025. URL `https://arxiv.org/abs/2512.09187`.

Saaket Agashe, Yue Fan, Anthony Reyna, and Xin Eric Wang. LLM-coordination: Evaluating and analyzing multi-agent coordination abilities in large language models. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 8053–8072, 2025. doi: 10.18653/v1/2025.findings-naacl.448. URL `https://aclanthology.org/2025.findings-naacl.448/`.

Jaewoo Ahn, Junseo Kim, Heeseung Yun, Jaehyeon Son, Dongmin Park, Jaewoong Cho, and Gunhee Kim. Flashadventure: A benchmark for gui agents solving full story arcs in diverse adventure games. In *EMNLP*, 2025.

Eloi Alonso, Adam Jelley, Vincent Micheli, Anssi Kanervisto, Amos J. Storkey, Tim Pearce, and Francois Fleuret. Diffusion for world modeling: Visual details matter in atari. In *Advances in Neural Information Processing Systems*, 2024.

John R. Anderson. Acquisition of cognitive skill. *Psychological Review*, 89:369–406, 1982.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Shengguang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang Zhu. Qwen technical report. *arXiv:2309.16609*, 2023.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, and Keqin Chen et al. Qwen3-vl technical report. *arXiv:2511.21631*, 2025a.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report, 2025b. URL `https://arxiv.org/abs/2502.13923`.

Suma Bailis, Jane Friedhoff, and Feiyang Chen. Werewolf arena: A case study in llm evaluation via social deduction, 2024. URL `https://arxiv.org/abs/2407.13943`.

Bowen Baker, Ilge Akkaya, Peter Zhokov, Joost Huizinga, Jie Tang, Adrien Ecoffet, Brandon Houghton, Raul Sampedro, and Jeff Clune. Video pretraining (VPT): Learning to act by watching unlabeled online videos. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022.

Philip J. Ball, Jakob Bauer, Frank Belletti, Bethanie Brownfield, Ariel Ephrat, Shlomi Fruchter, Agrim Gupta, Kristian Holsheimer, Aleksander Holynski, Jiri Hron, Christos Kaplanis, Marjorie Limont, Matt McGill, Yanko Oliveira, Jack Parker-Holder, Frank Perbet, Guy Scully, Jeremy Shar, Stephen Spencer, Omer Tov, Ruben Villegas, Emma Wang, Jessica Yung, Cip Baetu, Jordi Berbel, David Bridson, Jake Bruce, Gavin Buttimore, Sarah Chakera, Bilva Chandra, Paul Collins, Alex Cullum, Bogdan Damoc, Vibha Dasagi, Maxime Gazeau, Charles Gbadamosi, Woohyun Han, Ed Hirst, Ashyana Kachra, Lucie Kerley, Kristian Kjems, Eva Knoepfel, Vika Koriakin, Jessica Lo, Cong Lu, Zeb Mehring, Alex Moufarek, Henna Nandwani, Valeria Oliveira, Fabio Pardo, Jane Park, Andrew Pierson, Ben Poole, Helen Ran, Tim Salimans, Manuel Sanchez, Igor Saprykin, Amy Shen, Sailesh Sidhwani, Duncan Smith, Joe Stanton, Hamish Tomlinson, Dimple Vijaykumar, Luyu Wang, Piers Wingfield, Nat Wong, Keyang Xu, Christopher Yew, Nick Young, Vadim Zubov, Douglas Eck, Dumitru Erhan, Koray Kavukcuoglu, Demis Hassabis, Zoubin Gharamani, Raia Hadsell, Aäron van den Oord, Inbar Mosseri, Adrian Bolton, Satinder Singh, and Tim Rocktäschel. Genie 3: A new frontier for world models, 2025. URL `https://deepmind.google/blog/genie-3-a-new-frontier-for-world-models/`.

M. G. Bellemare, Y. Naddaf, J. Veness, and M. Bowling. The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research*, 47:253–279, jun 2013.

Anne Beyer, Kranti Chalamalasetti, Sherzod Hakimov, Brielen Madureira, Philipp Sadler, and David Schlangen. clembench-2024: A challenging, dynamic, complementary, multilingual benchmark and underlying flexible framework for llms as multi-action agents, 2024. URL `https://arxiv.org/abs/2405.20859`.

George Bredis, Stanislav Dereka, Viacheslav Sinii, Ruslan Rakhimov, and Daniil Gavrilov. Enhancing vision-language model training with reinforcement learning in synthetic worlds for real-world success, 2025. URL `https://arxiv.org/abs/2508.04280`.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, and Jared D et al. Kaplan. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pp. 1877–1901. Curran Associates, Inc., 2020. URL `https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf`.

Jake Bruce, Michael D Dennis, Ashley Edwards, Jack Parker-Holder, Yuge Shi, and Hughes et al. Genie: Generative interactive environments. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 4603–4623. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/bruce24a.html.

Shaofei Cai, Bowei Zhang, Zihao Wang, Xiaojian Ma, Anji Liu, and Yitao Liang. GROOT: Learning to follow instructions by watching gameplay videos. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=uleDLeiaT3.

Shaofei Cai, Zihao Wang, Kewei Lian, Zhancun Mu, Xiaojian Ma, Anji Liu, and Yitao Liang. Rocket-1: Mastering open-world interaction with visual-temporal context prompting. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 12122–12131, 2025.

Murray Campbell, A.Joseph Hoane, and Feng hsiung Hsu. Deep blue. *Artificial Intelligence*, 134(1):57–83, 2002. ISSN 0004-3702. doi: https://doi.org/10.1016/S0004-3702(01)00129-1. URL https://www.sciencedirect.com/science/article/pii/S0004370201001291.

Kranti Chalamalasetti, Jana Götze, Sherzod Hakimov, Brielen Madureira, Philipp Sadler, and David Schlangen. Clembench: Using game play to evaluate chat-optimized language models as conversational agents, 2023. URL https://arxiv.org/abs/2305.13455.

Haoxuan Che, Xuanhua He, Quande Liu, Cheng Jin, and Hao Chen. Gamegen-x: Interactive open-world game video generation. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=8VG8tpPZhe.

Jingye Chen, Yuzhong Zhao, Yupan Huang, Lei Cui, Li Dong, Tengchao Lv, Qifeng Chen, and Furu Wei. Model as a game: On numerical and spatial consistency for generative games. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, pp. 1958–1967, October 2025a.

Liang Chen, Hongcheng Gao, Tianyu Liu, Zhiqi Huang, Flood Sung, Xinyu Zhou, Yuxin Wu, and Baobao Chang. G1: Bootstrapping perception and reasoning abilities of vision-language model via reinforcement learning, 2025b. URL https://arxiv.org/abs/2505.13426.

Peng Chen, Pi Bu, Jun Song, Yuan Gao, and Bo Zheng. Can vlms play action role-playing games? take black myth wukong as a study case, 2024a. URL https://arxiv.org/abs/2409.12889.

Peng Chen, Pi Bu, Yingyao Wang, Xinyi Wang, Ziming Wang, and Jieet al. Guo. Combatvla: An efficient vision-language-action model for combat tasks in 3d action role-playing games. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 10919–10928, October 2025c.

Weize Chen, Yusheng Su, Jingwei Zuo, Cheng Yang, Chenfei Yuan, Chi-Min Chan, Heyang Yu, Yaxi Lu, Yi-Hsin Hung, Chen Qian, Yujia Qin, Xin Cong, Ruobing Xie, Zhiyuan Liu, Maosong Sun, and Jie Zhou. Agentverse: Facilitating multi-agent collaboration and exploring emergent behaviors. In *The Twelfth International Conference on Learning Representations*, 2024b. URL https://openreview.net/forum?id=EHg5GDnyq1.

Jie Cheng, Ruixi Qiao, YINGWEI MA, Binhua Li, Gang Xiong, Qinghai Miao, Yongbin Li, and Yisheng Lv. Scaling offline model-based RL via jointly-optimized world-action model pretraining. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=T1OvCSFaum.

Cheng Chi, Siyuan Feng, Yilun Du, Zhenjia Xu, Eric Cousineau, Benjamin Burchfiel, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, 44:1684 – 1704, 2023. URL https://api.semanticscholar.org/CorpusID:257378658.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

PrismarineJS contributors. PrismarineJS/mineflayer: Create Minecraft bots with a powerful, stable, and high-level JavaScript API. `https://github.com/PrismarineJS/mineflayer`, 2013. GitHub repository.

Leda Cosmides and John Tooby. Beyond intuition and instinct blindness: Toward an evolutionarily rigorous cognitive science. *Cognition*, 50(1-3):41–77, 1994.

Anthony Costarelli, Mat Allen, Roman Hauksson, Grace Sodunke, Suhas Hariharan, Carlson Cheng, Wenjie Li, Joshua M Clymer, and Arjun Yadav. Gamebench: Evaluating strategic reasoning abilities of LLM agents. In *Language Gamification - NeurIPS 2024 Workshop*, 2024. URL `https://openreview.net/forum?id=qrzKE533Jr`.

Decart and Spruce Campbell Xinlei Chen Robert Wachen Julian Quevedo, Quinn McIntyre. Oasis: A universe in a transformer. 2024. URL `https://oasis-model.github.io/`.

Jim Dilkes, Vahid Yazdanpanah, and Sebastian Stein. Reinforced language models for sequential decision making. *arXiv preprint arXiv:2508.10839*, 2025.

Tamil Sudaravan Mohan Doss, Michael Xu, Sudha Rao, Andrew D Wilson, and Balasaravanan Thoravi Kumaravel. Minenpc-task: Task suite for memory-aware minecraft agents. *arXiv preprint arXiv:2601.05215*, 2026.

Jinhao Duan, Renming Zhang, James Diffenderfer, Bhavya Kailkhura, Lichao Sun, Elias Stengel-Eskin, Mohit Bansal, Tianlong Chen, and Kaidi Xu. GTBench: Uncovering the strategic reasoning capabilities of LLMs via game-theoretic evaluations. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL `https://openreview.net/forum?id=ypggxVWIv2`.

Rachit Dubey, Pulkit Agrawal, Deepak Pathak, Tom Griffiths, and Alexei A. Efros. Investigating human priors for playing video games. In *ICML*, pp. 1348–1356, 2018. URL `http://proceedings.mlr.press/v80/dubey18a.html`.

Hafsteinn Einarsson. Mazeeval: A benchmark for testing sequential decision-making in language models, 2025. URL `https://arxiv.org/abs/2507.20395`.

Linxi Fan, Guanzhi Wang, Yunfan Jiang, Ajay Mandlekar, Yuncong Yang, Haoyi Zhu, Andrew Tang, De-An Huang, Yuke Zhu, and Anima Anandkumar. Minedojo: Building open-ended embodied agents with internet-scale knowledge. *Advances in Neural Information Processing Systems*, 35:18343–18362, 2022.

Lang Feng, Weihao Tan, Zhiyi Lyu, Longtao Zheng, Haiyang Xu, Ming Yan, Fei Huang, and Bo An. Towards efficient online tuning of VLM agents via counterfactual soft reinforcement learning. In *Forty-second International Conference on Machine Learning*, 2025. URL `https://openreview.net/forum?id=H76PMm7hf2`.

Yicheng Feng, Yuxuan Wang, Jiazheng Liu, Sipeng Zheng, and Zongqing Lu. Llama-rider: Spurring large language models to explore the open world. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pp. 4705–4724, 2024.

Paul M Fitts and Michael I Posner. Human performance. 1967.

John H Flavell. Metacognition and cognitive monitoring: A new area of cognitive–developmental inquiry. *American psychologist*, 34(10):906, 1979.

ARC Foundation. Arc-agi-3: A new challenge for frontier agentic intelligence. *arXiv preprint arXiv:2603.24621*, 2026.

Giacomo Frisoni, Lorenzo Molfetta, Davide Freddi, and Gianluca Moro. Mixture of masters: Sparse chess language models with player routing, 2026. URL https://arxiv.org/abs/2602.04447.

Honghao Fu, Junlong Ren, Qi Chai, Deheng Ye, Yujun Cai, and Hao Wang. Vistawise: Building cost-effective agent with cross-modal knowledge graph for minecraft. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 21895–21909, 2025.

Roberto Gallotta, Graham Todd, Marvin Zammit, Sam Earle, Antonios Liapis, Julian Togelius, and Georgios N. Yannakakis. Large language models and games: A survey and roadmap. *arXiv preprint arXiv:2402.18659*, 2024.

Dedre Gentner. Structure-mapping: A theoretical framework for analogy. *Cognitive science*, 7(2):155–170, 1983.

Mary Gick and Keith J. Holyoak. Schema induction and analogical transfer. *Cognitive Psychology*, 15:1–38, 1983.

Gerd Gigerenzer and Daniel Goldstein. Reasoning the fast and frugal way: Models of bounded rationality. *Psychological Review*, 62:650–669, 10 1996. doi: 10.1093/acprof:oso/9780199744282.003.0002.

Samuel Greydanus, Anurag Koul, Jonathan Dodge, and Alan Fern. Visualizing and understanding atari agents. In *International Conference on Machine Learning*, pp. 1792–1801. PMLR, 2018.

Jake Grigsby, Yuqi Xie, Justin Sasek, Steven Zheng, and Yuke Zhu. Human-level competitive pokémon via scalable offline reinforcement learning with transformers. In *Reinforcement Learning Conference*, 2025.

Leon Guertler, Bobby Cheng, Simon Yu, Bo Liu, Leshem Choshen, and Cheston Tan. Textarena. *arXiv preprint arXiv:2504.11442*, 2025.

Caglar Gulcehre, Ziyu Wang, Alexander Novikov, Thomas Paine, Sergio Gómez, Konrad Zolna, et al. Rl unplugged: A suite of benchmarks for offline reinforcement learning. volume 33, pp. 7248–7259, 2020.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, et al. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638, 2025a.

Junliang Guo, Yang Ye, Tianyu He, Haoyu Wu, Yushu Jiang, Tim Pearce, and Jiang Bian. Mineworld: a real-time and open-source interactive world model on minecraft. *arXiv preprint arXiv:2504.08388*, 2025b.

Yanjiang Guo, Lucy Xiaoyang Shi, Jianyu Chen, and Chelsea Finn. Ctrl-world: A controllable generative world model for robot manipulation. In *The Fourteenth International Conference on Learning Representations*, 2026.

William H. Guss, Brandon Houghton, Nicholay Topin, Phillip Wang, Cayden Codel, Manuela Veloso, and Ruslan Salakhutdinov. Minerl: a large-scale dataset of minecraft demonstrations. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, IJCAI'19, pp. 2442–2448. AAAI Press, 2019. ISBN 9780999241141.

David Ha and Jürgen Schmidhuber. Recurrent world models facilitate policy evolution. volume 31, 2018.

Danijar Hafner. Benchmarking the spectrum of agent capabilities, 2022.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. In *International Conference on Learning Representations*, 2020.

Danijar Hafner, Timothy Lillicrap, Mohammad Norouzi, and Jimmy Ba. Mastering atari with discrete world models. In *International Conference on Learning Representations*, 2021.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse control tasks through world models. *Nature*, 640:647–653, 2025. doi: 10.1038/s41586-025-08744-2.

Harry F Harlow. The formation of learning sets. *Psychological review*, 56(1):51, 1949.

Matthew Hausknecht, Prithviraj Ammanabrolu, Marc-Alexandre Côté, and Xingdi Yuan. Interactive fiction games: A colossal adventure. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 7903–7910, 2020.

Xianglong He, Chunli Peng, Zexiang Liu, Boyang Wang, Yifan Zhang, Qi Cui, et al. Matrix-game 2.0: An open-source, real-time, and streaming interactive world model. *arXiv preprint arXiv:2508.13009*, 2025a.

Yingchen He, Christian D Weilbach, Martyna E Wojciechowska, Yuxuan Zhang, and Frank Wood. Plaicraft: Large-scale time-aligned vision-speech-action dataset for embodied ai. *arXiv preprint arXiv:2505.12707*, 2025b.

Lanxiang Hu, Mingjia Huo, Yuxuan Zhang, Haoyang Yu, Eric P. Xing, Ion Stoica, Tajana Rosing, Haojian Jin, and Hao Zhang. lmgame-bench: How good are llms at playing games?, 2025a. URL `https://arxiv.org/abs/2505.15146`.

Lanxiang Hu, Qiyu Li, Anze Xie, Nan Jiang, Ion Stoica, Haojian Jin, and Hao Zhang. Gamearena: Evaluating LLM reasoning through live computer games. In *The Thirteenth International Conference on Learning Representations*, 2025b.

Sihao Hu, Tiansheng Huang, Fatih Ilhan, Selim Tekin, Gaowen Liu, Ramana Kompella, and Ling Liu. A survey on large language model-based game agents. *arXiv preprint arXiv:2404.02039*, 2024.

Johan Huizinga. *Homo Ludens: A Study of the Play-Element in Culture*. Wolters-Noordhoff, Groningen, 1938. English translation: Routledge & Kegan Paul, London, 1949.

Eric Jang, Alex Irpan, Mohi Khansari, Daniel Kappler, Frederik Ebert, Corey Lynch, Sergey Levine, and Chelsea Finn. BC-z: Zero-shot task generalization with robotic imitation learning. In *5th Annual Conference on Robot Learning*, 2021.

Matthew Johnson, Katja Hofmann, Tim Hutton, and David Bignell. The malmo platform for artificial intelligence experimentation. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, IJCAI'16, pp. 4246–4247. AAAI Press, 2016. ISBN 9781577357704.

Jesper Juul. *Half-Real: Video Games between Real Rules and Fictional Worlds*. MIT Press, Cambridge, MA, 2005. ISBN 978-0-262-10110-3.

Lukasz Kaiser, Mohammad Babaeizadeh, Piotr Milos, Blazej Osinski, Roy H. Campbell, Konrad Czechowski, Dumitru Erhan, Chelsea Finn, Piotr Kozakowski, Sergey Levine, Afroz Mohiuddin, Ryan Sepassi, George Tucker, and Henryk Michalewski. Model-based reinforcement learning for atari. In *International Conference on Learning Representations*, 2020.

Seth Karten, Andy Luu Nguyen, and Chi Jin. Pokéchamp: an expert-level minimax language agent, 2025.

Seth Karten, Jake Grigsby, Tersoo Upaa Jr, Junik Bae, Seonghun Hong, Hyunyoung Jeong, Jaeyoon Jung, Kun Kerdthaisong, Gyungbo Kim, Hyeokgi Kim, et al. The pokeagent challenge: Competitive and long-context learning at scale. *arXiv preprint arXiv:2603.15563*, 2026.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, Quan Vuong, Thomas Kollar, Benjamin Burchfiel, Russ Tedrake, Dorsa Sadigh, Sergey Levine, Percy Liang, and Chelsea Finn. Openvla: An open-source vision-language-action model, 2024. URL `https://arxiv.org/abs/2406.09246`.

Myung Ho Kim. Bridging symbolic control and neural reasoning in llm agents: The structured cognitive loop. *arXiv preprint arXiv:2511.17673*, 2025.

Donald E. Knuth and Ronald W. Moore. An analysis of alpha-beta pruning. *Artificial Intelligence*, 6(4): 293–326, 1975. ISSN 0004-3702. doi: https://doi.org/10.1016/0004-3702(75)90019-3.

Sai Kolasani, Maxim Saplin, Nicholas Crispino, Kyle Montgomery, Jared Quincy Davis, Matei Zaharia, Chi Wang, and Chenguang Wang. Llm chess: Benchmarking reasoning and instruction-following in llms through chess, 2025. URL https://arxiv.org/abs/2512.01992.

Sean Kulinski, Nicholas R. Waytowich, James Z. Hare, and David I. Inouye. Starcraftimage: A dataset for prototyping spatial reasoning methods for multi-agent environments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2023)*, pp. 22004–22013, 2023.

Heinrich Küttler, Nantas Nardelli, Alexander H. Miller, Roberta Raileanu, Marco Selvatici, Edward Grefenstette, and Tim Rocktäschel. The nethack learning environment. In *Advances in Neural Information Processing Systems 33 (NeurIPS 2020)*, 2020.

Heinrich Küttler, Nantas Nardelli, Alexander H. Miller, Roberta Raileanu, Marco Selvatici, Edward Grefenstette, and Tim Rocktäschel. The nethack learning environment, 2020. URL https://arxiv.org/abs/2006.13760.

John Laird and Michael Lent. Human-level ai's killer application: Interactive computer games. *AI Magazine*, 22:15–26, 06 2001.

John E Laird, Christian Lebiere, and Paul S Rosenbloom. A standard model of the mind: Toward a common computational framework across artificial intelligence, cognitive science, neuroscience, and robotics. *Ai Magazine*, 38(4):13–26, 2017.

Brenden M. Lake, Tomer D. Ullman, Joshua B. Tenenbaum, and Samuel J. Gershman. Building machines that learn and think like people. *Behavioral and Brain Sciences*, 40:e253, 2017. doi: 10.1017/S0140525X16001837.

Shane Legg and Marcus Hutter. Universal intelligence: A definition of machine intelligence. *Minds and Machines*, 17:391–444, 2007.

Wolfgang Lehrach, Daniel Hennes, Miguel Lazaro-Gredilla, Xinghua Lou, Carter Wendelken, Zun Li, Antoine Dedieu, Jordi Grau-Moya, Marc Lanctot, Atil Iscen, et al. Code world models for general game playing. *arXiv preprint arXiv:2510.04542*, 2025.

Yaniv Leviathan, Matan Kalman, and Yossi Matias. Fast inference from transformers via speculative decoding. In *International Conference on Machine Learning*, pp. 19274–19286. PMLR, 2023.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024.

Lingfeng Li, Yunlong Lu, Yuefei Zhang, Jingyu Yao, Yixin Zhu, KeYuan Cheng, Yongyi Wang, Qirui Zheng, Xionghui Yang, and Wenxin Li. Botzonebench: Scalable llm evaluation via graded ai anchors, 2026a. URL https://arxiv.org/abs/2602.13214.

Muyao Li, Zihao Wang, Kaichen He, Xiaojian Ma, and Yitao Liang. JARVIS-VLA: Post-training large-scale vision language models to play visual games with keyboards and mouse. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 17878–17899, 2025a.

Muyao Li, Zihao Wang, Kaichen He, Xiaojian Ma, and Yitao Liang. JARVIS-VLA: Post-training large-scale vision language models to play visual games with keyboards and mouse. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 17878–17899, Vienna, Austria, 2025b. Association for Computational Linguistics. doi: 10.18653/v1/2025.findings-acl.920. URL https://aclanthology.org/2025.findings-acl.920/.

Wenhao Li, Wenwu Li, Chuyun Shen, Junjie Sheng, Zixiao Huang, Di Wu, Yun Hua, Wei Yin, Xiangfeng Wang, Hongyuan Zha, and Bo Jin. Textatari: 100k frames game playing with language agents, 2025c. URL https://arxiv.org/abs/2506.04098.

Xinze Li, Ziyue Zhu, Siyuan Liu, Yubo Ma, Yuhang Zang, Yixin Cao, and Aixin Sun. Emembench: Interactive benchmarking of episodic memory for vlm agents, 2026b. URL https://arxiv.org/abs/2601.16690.

Yang Li, Xing Chen, Yutao Liu, Gege Qi, Yanxian BI, Zizhe Wang, Yunjian Zhang, and Yao Zhu. Beyond scaling: Assessing strategic reasoning and rapid decision-making capability of llms in zero-sum environments, 2026c. URL https://arxiv.org/abs/2603.09337.

Yuchen Li, Cong Lin, Muhammad Umair Nasir, Philip Bontrager, Jialin Liu, and Julian Togelius. Gvgai-llm: Evaluating large language model agents with infinite games, 2025d. URL https://arxiv.org/abs/2508.08501.

Zhen Li, Zian Meng, Shuwei Shi, Wenshuo Peng, Yuwei Wu, Bo Zheng, Chuanhao Li, and Kaipeng Zhang. Wildworld: A large-scale dataset for dynamic world modeling with actions and explicit state toward generative arpg, 2026d.

Fangzhou Liang, Tianshi Zheng, Chunkit Chan, Yauwai Yim, and Yangqiu Song. Llm-hanabi: Evaluating multi-agent gameplays with theory-of-mind and rationale inference in imperfect information collaboration game, 2025. URL https://arxiv.org/abs/2510.04980.

Yi Liao, Yu Gu, Yuan Sui, Zining Zhu, Yifan Lu, Guohua Tang, Zhongqian Sun, and Wei Yang. Think in games: Learning to reason in games via reinforcement learning with large language models, 2025.

John Licato and Stephen Steinle. Do persona-infused llms affect performance in a strategic reasoning game? In *Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics*, pp. 3497–3528, 2025.

Shalev Lifshitz, Keiran Paster, Harris Chan, Jimmy Ba, and Sheila McIlraith. Steve-1: A generative model for text-to-behavior in minecraft. In *Advances in Neural Information Processing Systems*, volume 36, pp. 69900–69929. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/dd03f856fc7f2efeec8b1c796284561d-Paper-Conference.pdf.

Jonathan Light, Min Cai, Sheng Shen, and Ziniu Hu. From text to tactic: Evaluating LLMs playing the game of avalon. In *NeurIPS 2023 Foundation Models for Decision Making Workshop*, 2023. URL https://openreview.net/forum?id=ltUrSrySOK.

Wenye Lin, Jonathan Roberts, Yunhan Yang, Samuel Albanie, Zongqing Lu, and Kai Han. GAMEBOT: Gaming arena for model evaluation - battle of tactics, 2024. URL https://openreview.net/forum?id=dr0s6aGYb7.

Qian Long, Zhi Li, Ran Gong, Ying Nian Wu, Demetri Terzopoulos, and Xiaofeng Gao. TeamCraft: A benchmark for multi-modal multi-agent systems in Minecraft, 2024.

Yitao Long, Yuru Jiang, Hongjun Liu, Yilun Zhao, et al. Puzzleplex: Benchmarking foundation models on reasoning and planning with puzzles, 2025. URL https://arxiv.org/abs/2510.06475.

Wenxuan Lu, Jiangyang He, Zhanqiu Zhang, Yiwen Guo, et al. Cultivating Game Sense for Yourself: Making VLMs Gaming Experts. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2025. doi: 10.18653/v1/2025.acl-long.643. URL https://aclanthology.org/2025.acl-long.643/.

Weiyu Ma, Qirui Mi, Yongcheng Zeng, Xue Yan, et al. Large language models play starcraft ii:benchmarks and a chain of summarization approach. In *Advances in Neural Information Processing Systems*, volume 37, pp. 133386–133442, 2024. doi: 10.52202/079017-4240. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/f0ebc318e2df08360b2df559e81602e5-Paper-Conference.pdf.

Loïc Magne, Anas Awadalla, Guanzhi Wang, Yinzhen Xu, et al. NitroGen: An open foundation model for generalist gaming agents, 2026. URL https://arxiv.org/abs/2601.02427.

Narada Maugin and Tristan Cazenave. SpinGPT: A large-language-model approach to playing poker correctly. In *Advances in Computer Games (ACG 2025)*, Lecture Notes in Computer Science. Springer, 2025.

Meta Fundamental AI Research Diplomacy Team (FAIR), Anton Bakhtin, Noam Brown, Emily Dinan, et al. Human-level play in the game of Diplomacy by combining language models with strategic reasoning. *Science*, 378(6624):1067–1074, 2022. doi: 10.1126/science.ade9097. URL https://www.science.org/doi/10.1126/science.ade9097.

Vincent Micheli, Eloi Alonso, and Francois Fleuret. Transformers are sample-efficient world models. In *The Eleventh International Conference on Learning Representations*, 2023.

Alexander Mott, Daniel Zoran, Mike Chrzanowski, Daan Wierstra, and Danilo Jimenez Rezende. Towards interpretable reinforcement learning using attention augmented agents. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32, 2019.

Jisu Nam, Yicong Hong, Chun-Hao Paul Huang, Feng Liu, JoungBin Lee, Jiyoung Kim, Siyoon Jin, Yunsung Lee, Jaeyoon Jung, Suhwan Choi, Seungryong Kim, and Yang Zhou. Worldcam: Interactive autoregressive 3d gaming worlds with camera pose as a unifying geometric representation. *arXiv preprint arXiv:2603.16871*, 2026.

Muhammad Umair Nasir, Steven James, and Julian Togelius. Gametraversalbenchmark: Evaluating planning abilities of large language models through traversing 2d game maps. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 31813–31827. Curran Associates, Inc., 2024. doi: 10.52202/079017-1000. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/3852c8254bc6d904c09db9921157f59b-Paper-Datasets_and_Benchmarks_Track.pdf.

Allen Newell. *Unified theories of cognition*. Harvard University Press, 1994.

Junhyuk Oh, Xiaoxiao Guo, Honglak Lee, Richard L. Lewis, and Satinder Singh. Action-conditional video prediction using deep networks in atari games. In *Advances in Neural Information Processing Systems*, 2015.

OpenAI, Christopher Berner, Greg Brockman, Brooke Chan, Vicki Cheung, Przemysław Dębiak, and Christy Dennison et al. Dota 2 with large scale deep reinforcement learning, 2019. URL https://arxiv.org/abs/1912.06680.

OpenAI, Aaron Hurst, Adam Lerer, Adam P. Goucher, and Adam Perelman et al. Gpt-4o system card. *arXiv:2410.21276*, 2024.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, and Mishkin et al. Training language models to follow instructions with human feedback. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA, 2022. Curran Associates Inc. ISBN 9781713871088.

Mingyu Ouyang, Siyuan Hu, Kevin Qinghong Lin, Hwee Tou Ng, and Mike Zheng Shou. Gameworld: Towards standardized and verifiable evaluation of multimodal game agents, 2026. URL https://arxiv.org/abs/2604.07429.

Abby O'Neill, Abdul Rehman, Abhiram Maddukuri, and Gupta Abhishek et al. Open x-embodiment: Robotic learning datasets and rt-x models : Open x-embodiment collaboration0. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6892–6903, 2024.

Davide Paglieri, Bartłomiej Cupiał, Samuel Coward, Ulyana Piterbarg, Maciej Wolczyk, Akbir Khan, Eduardo Pignatelli, Łukasz Kuciński, Lerrel Pinto, Rob Fergus, Jakob Nicolaus Foerster, Jack Parker-Holder, and Tim Rocktäschel. BALROG: Benchmarking agentic LLM and VLM reasoning on games. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=fp6t3F669F.

PAN Team Institute of Foundation Models. Pan: A world model for general, interactable, and long-horizon world simulation. *arXiv preprint arXiv:2511.09057*, 2025.

Dongmin Park, Minkyu Kim, Beongjun Choi, Junhyuck Kim, Keon Lee, Jonghyun Lee, Inkyu Park, Byeong-Uk Lee, Jaeyoung Hwang, Jaewoo Ahn, Ameya Sunil Mahabaleshwarkar, Bilal Kartal, Pritam Biswas, Yoshi Suhara, Kangwook Lee, and Jaewoong Cho. Orak: A foundational benchmark for training and evaluating LLM agents on diverse video games. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=H1ncX6O6Yh.

Tim Pearce and Jun Zhu. Counter-strike deathmatch with large-scale behavioural cloning. In *2022 IEEE Conference on Games (CoG)*, pp. 104–111, 2022. doi: 10.1109/CoG51982.2022.9893617.

Joseph J Peper, Sai Krishna Gandra, Yunxiang Zhang, Vaibhav Chennareddy, Shloki Jha, Ali Payani, and Lu Wang. LLMs as Rules Oracles: Exploring Real-World Multimodal Reasoning in Tabletop Strategy Game Environments. In *International Conference on Learning Representations*, 2026.

Long Phan, Mantas Mazeika, Andy Zou, and Dan Hendrycks. Textquests: How good are llms at text-based video games?, 2025. URL https://arxiv.org/abs/2507.23701.

Marc-Antoine Provost, Nejc Ilenic, Christopher Solinas, and Philippe Beardsell. Gto wizard benchmark, 2026. URL https://arxiv.org/abs/2603.23660.

Psibot Team. From human skill to robotic mastery, March 2026. URL https://cypypccpy.github.io/tech-blog.github.io/.

Siyuan Qi, Shuo Chen, Yexin Li, Xiangyu Kong, Junqi Wang, Bangcheng Yang, Pring Wong, Yifan Zhong, Xiaoyuan Zhang, Zhaowei Zhang, Nian Liu, Yaodong Yang, and Song-Chun Zhu. Civrealm: A learning and reasoning odyssey in civilization for decision-making agents. In *The Twelfth International Conference on Learning Representations*, 2024.

Maria Abi Raad, Arun Ahuja, Catarina Barros, Frederic Besse, Andrew Bolt, Adrian Bolton, et al. Scaling instructable agents across many simulated worlds. *arXiv preprint arXiv:2404.10179*, 2024.

Mahesh Ramesh, Kaousheik Jayakumar, Aswinkumar Ramkumar, Pavan Thodima, Aniket Rege, and Emmanouil-Vasileios Vlatakis-Gkaragkounis. Sparks of cooperative reasoning: Llms as strategic hanabi agents, 2026. URL https://arxiv.org/abs/2601.18077.

Katie Salen and Eric Zimmerman. *Rules of Play: Game Design Fundamentals*. MIT Press, Cambridge, MA, 2003. ISBN 978-0-262-24045-1.

Georgy Savva, Oscar Michel, Daohan Lu, Suppakit Waiwitlikhit, Timothy Meehan, Dhairya Mishra, Srivats Poddar, Jack Lu, and Saining Xie. Solaris: Building a multiplayer video world model in minecraft. *arXiv preprint arXiv:2602.22208*, 2026.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

ByteDance Seed. Ui-tars-1.5. https://seed-tars.com/1.5, 2025.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models, 2024.

Ansh Kumar Sharma, Yixiang Sun, Ninghao Lu, Yunzhe Zhang, Jiarao Liu, and Sherry Yang. World-gymnast: Training robots with reinforcement learning in a world model, 2026.

Jiajun Shi, Jian Yang, Jiaheng Liu, Xingyuan Bu, Jiangjie Chen, Junting Zhou, Kaijing Ma, Zhoufutu Wen, Bingli Wang, Yancheng He, Liang Song, Hualei Zhu, Shilong Li, Xingjian Wang, Wei Zhang, Ruibin Yuan, Yifan Yao, Wenjun Yang, Yunli Wang, Siyuan Fang, Siyu Yuan, Qianyu He, Xiangru Tang, Yingshui Tan, Wangchunshu Zhou, Zhaoxiang Zhang, Zhoujun Li, Wenhao Huang, and Ge Zhang. Korgym: A dynamic game platform for llm reasoning evaluation, 2025. URL https://arxiv.org/abs/2505.14552.

Huang-A. Maddison C. et al. Silver, D. Mastering the game of go with deep neural networks and tree search. *Nature*, 529:484–489, 2016.

SIMA-team, Adrian Bolton, Alexander Lerchner, Alexandra Cordell, Alexandre Moufarek, Andrew Bolt, Andrew Lampinen, Anna Mitenkova, Arne Olav Hallingstad, Bojan Vujatovic, Bonnie Li, Cong Lu, Daan Wierstra, Daniel P. Sawyer, Daniel Slater, David Reichert, Davide Vercelli, Demis Hassabis, Drew A. Hudson, Duncan Williams, Ed Hirst, Fabio Pardo, Felix Hill, Frederic Besse, Hannah Openshaw, Harris Chan, Hubert Soyer, Jane X. Wang, Jeff Clune, John Agapiou, John Reid, Joseph Marino, Junkyung Kim, Karol Gregor, Kaustubh Sridhar, Kay McKinney, Laura Kampis, Lei M. Zhang, Loic Matthey, Luyu Wang, Maria Abi Raad, Maria Loks-Thompson, Martin Engelcke, Matija Kecman, Matthew Jackson, Maxime Gazeau, Ollie Purkiss, Oscar Knagg, Peter Stys, Piermaria Mendolicchio, Raia Hadsell, Rosemary Ke, Ryan Faulkner, Sarah Chakera, Satinder Singh Baveja, Shane Legg, Sheleem Kashem, Tayfun Terzi, Thomas Keck, Tim Harley, Tim Scholtes, Tyson Roberts, Volodymyr Mnih, Yulan Liu, Zhengdong Wang, and Zoubin Ghahramani. Sima 2: A generalist embodied agent for virtual worlds. *arXiv:2512.04797*, 2025.

Richard D Smallwood and Edward J Sondik. The optimal control of partially observable markov processes. *Stanford University*, 1971.

Zirui Song, Yuan Huang, Junchang Liu, Haozhe Luo, Chenxi Wang, Lang Gao, Zixiang Xu, Mingfei Han, Xiaojun Chang, and Xiuying Chen. Beyond survival: Evaluating llms in social deduction games with human-aligned strategies, 2025. URL `https://arxiv.org/abs/2510.11389`.

Yue Su, Xinyu Zhan, Hongjie Fang, Han Xue, Hao-Shu Fang, Yong-Lu Li, Cewu Lu, and Lixin Yang. Dense policy: Bidirectional autoregressive learning of actions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 14486–14495, October 2025.

Theodore Sumers, Shunyu Yao, Karthik R Narasimhan, and Thomas L. Griffiths. Cognitive architectures for language agents. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL `https://openreview.net/forum?id=1i6ZCvflQJ`. Survey Certification, Featured Certification.

Haochen Sun, Shuwen Zhang, Lujie Niu, Lei Ren, Hao Xu, Hao Fu, Fangkun Zhao, Caixia Yuan, and Xiaojie Wang. Collab-overcooked: Benchmarking and evaluating large language models as collaborative agents. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 4922–4951, 2025a.

Wenqiang Sun, Haiyu Zhang, Haoyuan Wang, Junta Wu, Zehan Wang, Zhenwei Wang, et al. Worldplay: Towards long-term geometric consistency for real-time interactive world modeling. *arXiv preprint arXiv:2512.14614*, 2025b.

Weihao Tan, Changjiu Jiang, Yu Duan, Mingcong Lei, Li JiaGeng, et al. Stardojo: Benchmarking open-ended behaviors of agentic multimodal LLMs in production–living simulations with stardew valley. In *First Workshop on Multi-Turn Interactions in Large Language Models*, 2025a. URL `https://openreview.net/forum?id=R0mmX6BEau`.

Weihao Tan, Xiangyang Li, Yunhao Fang, Heyuan Yao, Shi Yan, Hao Luo, et al. Lumine: An open recipe for building generalist agents in 3d open worlds. *arXiv preprint arXiv:2511.08892*, 2025b.

Weihao Tan, Wentao Zhang, Xinrun Xu, Haochong Xia, Ziluo Ding, et al. Cradle: Empowering foundation agents towards general computer control. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 58658–58725. PMLR, 13–19 Jul 2025c. URL `https://proceedings.mlr.press/v267/tan25h.html`.

Guoqin Tang, Qingxuan Jia, Gang Chen, Tong Li, Zeyuan Huang, Zihang Lv, and Ning Ji. Vlm-dewm: Dynamic external world model for verifiable and resilient vision-language planning in manufacturing. *arXiv preprint arXiv:2602.15549*, 2026.

Wenjie Tang, Yuan Zhou, Erqiang Xu, Keyan Cheng, Minne Li, and Liquan Xiao. Dsgbench: A diverse strategic game benchmark for evaluating llm-based agents in complex decision-making environments, 2025. URL https://arxiv.org/abs/2503.06047.

Jingqi Tong, Jixin Tang, Hangcheng Li, Yurong Mou, Ming Zhang, Jun Zhao, et al. Game-rl: Synthesizing multimodal verifiable game data to boost vlms' general reasoning. *arXiv preprint arXiv:2505.13886*, 2025.

Oguzhan Topsakal, Edell Colby, and Harper Jackson. Evaluating the performance of large language models (llms) through grid-based game competitions: An extensible benchmark and leaderboard on the path to artificial general intelligence (agi). *The Journal of Cognitive Systems*, 9(2):8–19.

Mark Towers, Ariel Kwiatkowski, Jordan Terry, and John U. Balis et al. Gymnasium: A standard interface for reinforcement learning environments. *arXiv:2407.17032*, 2024.

Jen tse Huang, Eric John Li, Man Ho Lam, Tian Liang, Wenxuan Wang, Youliang Yuan, Wenxiang Jiao, Xing Wang, Zhaopeng Tu, and Michael R. Lyu. How far are we on the decision-making of llms? evaluating llms' gaming ability in multi-agent environments, 2025. URL https://arxiv.org/abs/2403.11807.

Alan M. Turing. Digital computers applied to games. In *Faster Than Thought.* Pitman, London, 1953.

Jack Urbanek, Angela Fan, Siddharth Karamcheti, Saachi Jain, Samuel Humeau, Emily Dinan, Tim Rocktäschel, Douwe Kiela, Arthur Szlam, and Jason Weston. Learning to speak and act in a fantasy text adventure game. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP 2019)*, pp. 673–683, 2019.

Dani Valevski, Yaniv Leviathan, Moab Arar, and Shlomi Fruchter. Diffusion models are real-time game engines. *arXiv preprint arXiv:2408.14837*, 2024. URL https://arxiv.org/abs/2408.14837.

Dani Valevski, Yaniv Leviathan, Moab Arar, and Shlomi Fruchter. Diffusion models are real-time game engines. In *International Conference on Learning Representations*, 2025.

Victor Conchello Vendrell, Max Ruiz Luyten, and Mihaela van der Schaar. GameTalk: Training LLMs for strategic conversation, 2026.

Babuschkin I. Czarnecki W.M. et al. Vinyals, O. Grandmaster level in starcraft ii using multi-agent reinforcement learning. *Nature*, 575:350–354, 2019.

Oriol Vinyals, Timo Ewalds, Sergey Bartunov, Petko Georgiev, Alexander Sasha Vezhnevets, Michelle Yeo, and et al. Starcraft II: A new challenge for reinforcement learning, 2017.

Chengjia Wang, Lanling Tang, Ming Yuan, Jiongchi Yu, Xiaofei Xie, and Jiajun Bu. Leveraging llm agents for automated video game testing. *arXiv preprint arXiv:2509.22170*, 2025a.

Dawei Wang, Chengming Zhou, Di Zhao, Xinyuan Liu, Marci Chi Ma, Gary Ushaw, and Richard Davison. Towermind: A tower defence game learning environment and benchmark for llm as agents, 2026a. URL https://arxiv.org/abs/2601.05899.

Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandlekar, Chaowei Xiao, Yuke Zhu, and et al. Voyager: An open-ended embodied agent with large language models. *Trans. Mach. Learn. Res.*, 2024, 2024a. URL http://dblp.uni-trier.de/db/journals/tmlr/tmlr2024.html#WangX0MXZFA24.

Haolin Wang, Xueyan Li, Yazhe Niu, Shuai Hu, and Hongsheng Li. Empowering llms in decision games through algorithmic data synthesis, 2025b.

Haoming Wang, Haoyang Zou, Huatong Song, and Jiazhan Feng et al. Ui-tars-2 technical report: Advancing gui agent with multi-turn reinforcement learning. *arXiv:2509.02544*, 2025c.

Xinyu Wang, Bohan Zhuang, and Qi Wu. Are large vision language models good game players? In *The Thirteenth International Conference on Learning Representations*, 2025d.

Yunzhe Wang, Runhui Xu, Kexin Zheng, Tianyi Zhang, Jayavibhav Niranjan Kogundi, Soham Hans, and Volkan Ustun. Gameplayqa: A benchmarking framework for decision-dense pov-synced multi-video understanding of 3d virtual agents, 2026b. URL https://arxiv.org/abs/2603.24329.

Zihao Wang, Shaofei Cai, Guanzhou Chen, Anji Liu, Xiaojian (Shawn) Ma, and Yitao Liang. Describe, explain, plan and select: Interactive planning with llms enables open-world multi-task agents. In *Advances in Neural Information Processing Systems*, volume 36, pp. 34153–34189, 2023a. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/6b8dfb8c0c12e6fafc6c256cb08a5ca7-Paper-Conference.pdf.

Zihao Wang, Shaofei Cai, Anji Liu, Yonggang Jin, Jinbing Hou, Bowei Zhang, and et al. JARVIS-1: Open-world multi-task agents with memory-augmented multimodal language models, 2023b.

Zihao Wang, Shaofei Cai, Zhancun Mu, Haowei Lin, Ceyao Zhang, Xuejie Liu, and et al. Omnijarvis: Unified vision-language-action tokenization enables open-world instruction following agents. In *Advances in Neural Information Processing Systems 37 (NeurIPS 2024)*, 2024b. doi: 10.52202/079017-2331.

Zihao Wang, Muyao Li, Kaichen He, Xiangyu Wang, Zhancun Mu, Anji Liu, and et al. Openha: A series of open-source hierarchical agentic models in minecraft. *arXiv preprint arXiv:2509.13347*, 2025e. URL https://arxiv.org/abs/2509.13347.

Zihao Wang, Xujing Li, Yining Ye, Junjie Fang, Haoming Wang, and et al. Game-tars: Pretrained foundation models for scalable generalist multimodal game agents. *arXiv:2510.23691*, 2025f.

Zile Wang, Zexiang Liu, Jiaxing Li, Kaichen Huang, Baixin Xu, Fei Kang, and et al. Matrix-game 3.0: Real-time and streaming interactive world model with long-horizon memory. *arXiv preprint arXiv:2604.08995*, 2026c.

Yule Wen, Yixin Ye, Yanzhe Zhang, Diyi Yang, and Hao Zhu. Real-time reasoning agents in evolving environments. *arXiv preprint arXiv:2511.04898*, 2025.

Shuang Wu, Liwen Zhu, Tao Yang, Shiwei Xu, Qiang Fu, Yang Wei, and et al. Enhance reasoning for large language models in the game werewolf, 2024a. URL https://arxiv.org/abs/2402.02330.

Yue Wu, Xuan Tang, Tom Mitchell, and Yuanzhi Li. Smartplay : A benchmark for LLMs as intelligent agents. In *The Twelfth International Conference on Learning Representations*, 2024b. URL https://openreview.net/forum?id=S2oTVrlcp3.

Boyang Xia, Weiyou Tian, Qingnan Ren, Jiaqi Huang, Jie Xiao, and Shuo et al. Lu. Implicit strategic optimization: Rethinking long-horizon decision-making in adversarial poker environments. *arXiv preprint arXiv:2602.08041*, 2026.

Guangxuan Xiao, Ji Lin, Mickael Seznec, and Wu et al. SmoothQuant: Accurate and efficient post-training quantization for large language models. In *Proceedings of the 40th International Conference on Machine Learning*, 2023.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=NG7sS51zVF.

Yunfei Xie, Yinsong Ma, Shiyi Lan, Alan Yuille, Junfei Xiao, and Chen Wei. Play to generalize: Learning to reason through game play. In *The Fourteenth International Conference on Learning Representations*, 2026a. URL https://openreview.net/forum?id=u1tsgXPh2o.

Zhengwei Xie, Zhisheng Chen, Ziyan Weng, and Tingyu Wu et al. Steve-evolving: Open-world embodied self-evolution via fine-grained diagnosis and dual-track knowledge distillation, 2026b. URL https://arxiv.org/abs/2603.13131.

Xiaojie Xu, Zongyuan Li, Chang Lu, Runnan Qi, and Yanan Ni et al. Reflection of episodes: Learning to play game from expert and self experiences, 2026a. URL https://arxiv.org/abs/2502.13388.

Xinrun Xu, Pi Bu, Ye Wang, Börje F. Karlsson, Ziming Wang, and Tengtao Song et al. Deepphy: Benchmarking agentic vlms on physical reasoning. In *AAAI*, pp. 34160–34168. AAAI Press, 2026b.

Zelai Xu, Chao Yu, Fei Fang, Yu Wang, and Yi Wu. Language agents with reinforcement learning for strategic play in the werewolf game. In *ICML*, Proceedings of Machine Learning Research, pp. 55434–55464. PMLR / OpenReview.net, 2024.

Zelai Xu, Wanjun Gu, Chao Yu, Yi Wu, and Yu Wang. Learning strategic language agents in the Werewolf game with iterative latent space policy optimization. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, 2025.

Weicai Yan, Yuhong Dai, Qi Ran, and Haodong Li et al. Proact-vl: A proactive videollm for real-time ai companions, 2026. URL `https://arxiv.org/abs/2603.03447`.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, and Bo Zheng et al. Qwen3 technical report, 2025. URL `https://arxiv.org/abs/2505.09388`.

Mingyu Yang, Junyou Li, Zhongbin Fang, Sheng Chen, Yangbin Yu, Qiang Fu, Wei Yang, and Deheng Ye. Playable game generation, 2024. URL `https://arxiv.org/abs/2412.00887`.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. In *The eleventh international conference on learning representations*, 2022.

Seonghyeon Ye, Yunhao Ge, Kaiyuan Zheng, Shenyuan Gao, Sihyun Yu, George Kurian, Suneel Indupuru, You Liang Tan, Chuning Zhu, Jiannan Xiang, Ayaan Malik, Kyungmin Lee, William Liang, Nadun Ranawaka, Jiasheng Gu, Yinzhen Xu, Guanzhi Wang, Fengyuan Hu, Avnish Narayan, Johan Bjorck, Jing Wang, Gwanghyun Kim, Dantong Niu, Ruijie Zheng, Yuqi Xie, Jimmy Wu, Qi Wang, Ryan Julian, Danfei Xu, Yilun Du, Yevgen Chebotar, Scott Reed, Jan Kautz, Yuke Zhu, Linxi "Jim" Fan, and Joel Jang. World action models are zero-shot policies, 2026.

Lance Ying, Ryan Truong, Prafull Sharma, Kaiya Ivy Zhao, Nathan Cloos, Kelsey R. Allen, Thomas L. Griffiths, Katherine M. Collins, José Hernández-Orallo, Phillip Isola, Samuel J. Gershman, and Joshua B. Tenenbaum. Ai gamestore: Scalable, open-ended evaluation of machine general intelligence with human games. *arXiv:2602.17594*, 2026.

Jiwen Yu, Yiran Qin, Xintao Wang, Pengfei Wan, Di Zhang, and Xihui Liu. Gamefactory: Creating new games with generative interactive videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 11590–11599, 2025.

Huining Yuan, Zelai Xu, Zheyue Tan, Xiangmin Yi, Mo Guang, Kaiwen Long, Haojia Hui, Boxun Li, Xinlei Chen, Bo Zhao, Xiao-Ping Zhang, Chao Yu, and Yu Wang. MARSHAL: Incentivizing multi-agent reasoning via self-play with strategic LLMs, 2025.

Yuguang Yue, Chris Green, Samuel Hunt, Irakli Salia, Wenzhe Shi, and Jonathan J Hunt. Pixels to play: A foundation model for 3d gameplay. In *2025 IEEE Conference on Games (CoG)*, pp. 1–4. IEEE, 2025a.

Yuguang Yue, Irakli Salia, Samuel Hunt, Christopher Green, Wenzhe Shi, and Jonathan J Hunt. Learning to play: A multimodal agent for 3d game-play. *arXiv preprint arXiv:2510.16774*, 2025b.

Yuguang Yue, Irakli Salia, Samuel Hunt, Chris Green, Wenzhe Shi, and Jonathan J. Hunt. Scaling behavior cloning improves causal reasoning: An open model for real-time video game playing. *arXiv:2601.04575*, 2026.

Alex L. Zhang, Thomas L. Griffiths, Karthik R. Narasimhan, and Ofir Press. Videogamebench: Can vision-language models complete popular video games?, 2025a. URL `https://arxiv.org/abs/2505.18134`.

Chi Zhang, Penglin Cai, Yuhui Fu, Haoqi Yuan, and Zongqing Lu. Creative agents: Empowering agents with imagination for creative tasks. *arXiv preprint arXiv:2312.02519*, 2023.

Haoran Zhang, Chenhao Zhu, Sicong Guo, Hanzhe Guo, Haiming Li, and Donglin Yu. Starbench: A turn-based rpg benchmark for agentic multimodal decision-making and information seeking, 2025b. URL https://arxiv.org/abs/2510.18483.

Jinming Zhang and Yunfei Long. Learning to play like humans: A framework for llm adaptation in interactive fiction games. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 10188–10205, 2025.

Kuan Zhang, Dongchen Liu, Qiyue Zhao, Jinkun Hou, Xinran Zhang, Qinlei Xie, Miao Liu, and Yiming Li. Gameverse: Can vision-language models learn from video-based reflection?, 2026a. URL https://arxiv.org/abs/2603.06656.

Mingyu Zhang, Lifeng Zhuo, Tianxi Tan, Guocan Xie, Xian Nie, Yan Li, Renjie Zhao, Zizhu He, Ziyu Wang, Jiting Cai, and Yong-Lu Li. Ipr-1: Interactive physical reasoner. *arXiv preprint arXiv:2511.15407*, 2025c.

Ruizhi Zhang, Ye Huang, Yuangang Pan, Chuanfu Shen, Zhilin Liu, Ting Xie, Wen Li, and Lixin Duan. Pokegym: A visually-driven long-horizon benchmark for vision-language models, 2026b. URL https://arxiv.org/abs/2604.08340.

Ruohan Zhang, Calen Walshe, Zhuode Liu, Lin Guan, and Karl S. Muller et al. Atari-head: Atari human eye-tracking and demonstration dataset. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI 2020)*, pp. 6811–6820, 2020. doi: 10.1609/aaai.v34i04.6161.

Yifan Zhang, Chunli Peng, Boyang Wang, Puyi Wang, Qingcheng Zhu, Fei Kang, Biao Jiang, Zedong Gao, Eric Li, Yang Liu, and Yahui Zhou. Matrix-game: Interactive world foundation model. *arXiv preprint arXiv:2506.18701*, 2025d.

Yinqi Zhang, Xintian Han, Haolong Li, Kedi Chen, and Shaohui Lin. Complete chess games enable llm become a chess master, 2025e. URL https://arxiv.org/abs/2501.17186.

Yuxuan Zhang, Haoyang Yu, Lanxiang Hu, Haojian Jin, and Hao Zhang. General modular harness for llm agents in multi-turn gaming environments. *arXiv preprint arXiv:2507.11633*, 2025f.

Zhicheng Zhang, Ziyan Wang, Yali Du, and Fei Fang. VAM: Verbalized action masking for controllable exploration in RL post-training – a chess case study, 2026c.

Yujie Zhao, Lanxiang Hu, Yang Wang, Minmin Hou, Hao Zhang, Ke Ding, and Jishen Zhao. Stronger-MAS: Multi-agent reinforcement learning for collaborative LLMs, 2025.

Xiangxi Zheng, Linjie Li, Zhengyuan Yang, Ping Yu, Alex Jinpeng Wang, Rui Yan, Yuan Yao, and Lijuan Wang. V-mage: A game evaluation framework for assessing vision-centric capabilities in multimodal large language models. *arXiv:2504.06148*, 2025a.

Xinyue Zheng, Haowei Lin, Kaichen He, Zihao Wang, QIANG FU, Haobo Fu, Zilong Zheng, and Yitao Liang. MCU: An evaluation framework for open-ended game agents. In *Forty-second International Conference on Machine Learning*, 2025b. URL https://openreview.net/forum?id=hrdLhNDAzp.

Zheyuan Zhou, Liang Du, Zixun Sun, Xiaoyu Zhou, Ruimin Ye, Qihao Chen, Yinda Chen, and Lemiao Qiu. MAIN-VLA: Modeling abstraction of intention and environment for vision-language-action models. *arXiv preprint arXiv:2602.02212*, 2026. URL https://arxiv.org/abs/2602.02212.

Fangqi Zhu, YAN Zhengyang, Zicong Hong, Quanxin Shou, Xiao Ma, and Song Guo. WMPO: World model-based policy optimization for vision-language-action models. In *The Fourteenth International Conference on Learning Representations*, 2026.

Richard Zhuang, Akshat Gupta, Richard Yang, Aniket Rahane, Zhengyu Li, and Gopala Anumanchipalli. Pokerbench: Training large language models to become professional poker players, 2025. URL https://arxiv.org/abs/2501.08328.

Brianna Zitkovich, Tianhe Yu, Sichun Xu, Peng Xu, Ted Xiao, Fei Xia, Jialin Wu, Paul Wohlhart, Stefan Welker, Ayzaan Wahid, Quan Vuong, Vincent Vanhoucke, Huong Tran, Radu Soricut, Anikait Singh, Jaspiar Singh, Pierre Sermanet, Pannag R. Sanketi, Grecia Salazar, Michael S. Ryoo, Krista Reymann, Kanishka Rao, Karl Pertsch, Igor Mordatch, Henryk Michalewski, Yao Lu, Sergey Levine, Lisa Lee, Tsang-Wei Edward Lee, Isabel Leal, Yuheng Kuang, Dmitry Kalashnikov, Ryan Julian, Nikhil J. Joshi, Alex Irpan, Brian Ichter, Jasmine Hsu, Alexander Herzog, Karol Hausman, Keerthana Gopalakrishnan, Chuyuan Fu, Pete Florence, Chelsea Finn, Kumar Avinava Dubey, Danny Driess, Tianli Ding, Krzysztof Marcin Choromanski, Xi Chen, Yevgen Chebotar, Justice Carbajal, Noah Brown, Anthony Brohan, Montserrat Gonzalez Arenas, and Kehang Han. Rt-2: Vision-language-action models transfer web knowledge to robotic control. In Jie Tan, Marc Toussaint, and Kourosh Darvish (eds.), *Proceedings of The 7th Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, pp. 2165–2183. PMLR, 06–09 Nov 2023.

A Appendix