01 What People Mean

13 MIN READ · INTERACTIVE

> The phrase now covers at least five different things, and the people using it rarely say which. A field guide to telling them apart before you read another paper about them.

A clip goes past on your timeline. Somebody is walking through a landscape with the arrow keys, and the caption says it was generated, not built. It looks like a video game nobody made.

The replies call it a world model. So you go and look the phrase up.

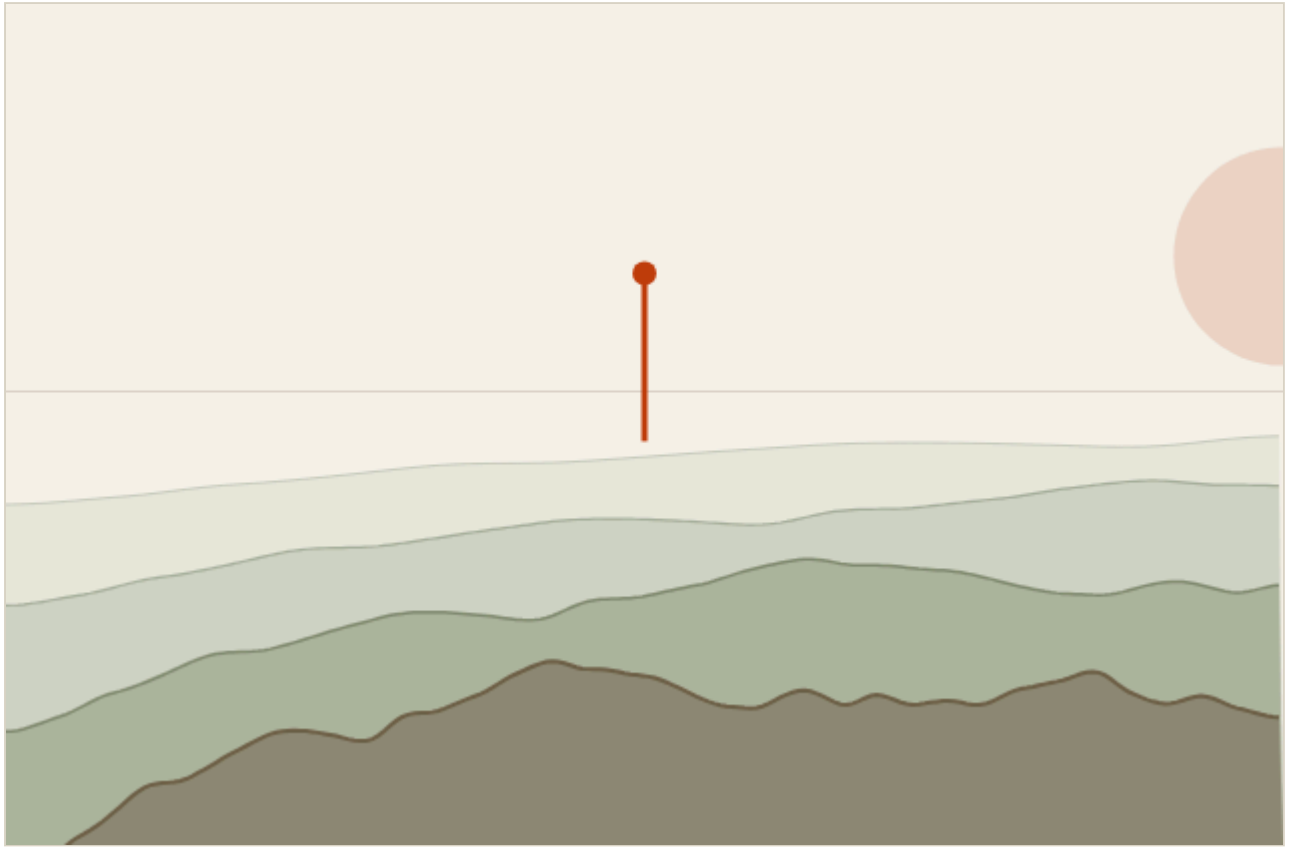


FIG. 1.1 Not a recording. The ridgelines are computed from your heading the moment they are drawn, and nothing behind the picture stores a landscape. Walk away from the marker and come back with the switch off, then again with it on.

Ten minutes later there are five tabs open and they describe five different machines. One generates video you steer with the arrow keys. One exports geometry into Blender. One is a paper about predicting embeddings that never shows you a video at all.

That is not a research failure. Between roughly 2018 and 2024, computer vision, reinforcement learning, robotics and interpretability all reached for the same two words, and every one of them was describing something real. By 2026 the labs had started publishing dictionaries.

In this chapter I'll deal with the five definitions in current use, where each came from, and the one test that separates them. Chapters 2 through 6 build the machinery underneath.

“world model” 5 INCOMPATIBLE ANSWERS

A DeepMind demo where somebody walks through a generated landscape with the arrow keys.

Genie 3 · Google DeepMind ↗

TURNS OUT TO BE

Renderer
https://deepmind.google/blog/genie-3-a-new-frontier-for-world-models/

A Meta paper about predicting video embeddings that never shows you a video.

V-JEPA 2 · Meta AI ↗

TURNS OUT TO BE

Representation
https://ai.meta.com/blog/v-jepa-2-world-model-benchmarks/

A 3-D environment you can load straight into Blender.

Marble · World Labs ↗

TURNS OUT TO BE

Simulator
https://www.worldlabs.ai/blog/marble-world-model

A reinforcement learning result from 2018 about a car in a racing game.

World Models · Ha & Schmidhuber ↗

TURNS OUT TO BE

Dynamics Model
https://worldmodels.github.io/

An argument, conducted mostly at volume, about whether a language model that has never seen a chessboard has one inside it anyway.

Emergent World Representations · Li et al. ↗

TURNS OUT TO BE

```
Implicit Model
https://arxiv.org/abs/2210.13382
```

FIG. 1.2 Five results, five different classes of system, all of them called world models by the people who built them.

Origins

The term started somewhere specific. In control theory, and later in reinforcement learning (training an agent by letting it act, watching what happens, and rewarding what worked), a world model is a learned transition function. A state and an action go in. The next state comes out.

Written down it looks like this.

$$p(s_{t+1} \mid s_t, a_t)$$

the chance of what happens next, given where you are now and what you do

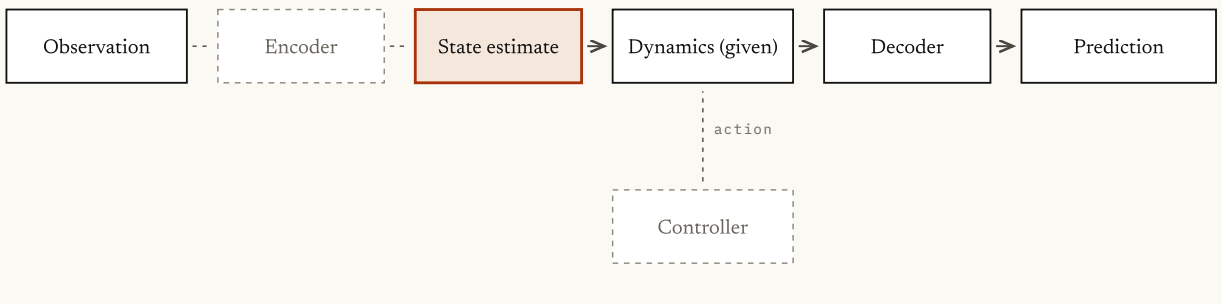
FIG. 1.3 Every system in this chapter is an argument about what belongs in place of *s*: pixels, geometry, a compact vector, or an embedding. That single choice is what separates the five definitions.

Kalman's 1960 filter nailed one half of it: working out a hidden state from noisy measurements (it estimates state, but never asks what happens if I act, so it is ancestry rather than membership). Around 1990 Schmidhuber built neural systems in two parts, where one network modelled the world and another chose actions, and the first predicted what the second's choices would do. Sutton's Dyna did something related. It alternated between learning from real experience and learning from experience the model made up.

Ha and Schmidhuber's *World Models* (2018) made the label popular. Their system had three pieces: an encoder (a network that squeezes a video frame down to a short list of numbers), a dynamics model that predicted where those numbers go next, and a small controller trained almost entirely inside the model's own imagined rollouts.

PlaNet (2018) learned those dynamics straight from pixels and planned in latent space (latent just means the model's own compressed description: a few hundred numbers instead of a million pixels). Dreamer (2019) used the same machinery to learn behaviour from imagined trajectories.

Then the word travelled. JEPA stopped predicting what the next frame looks like and started predicting a summary of it. Genie, Cosmos and Marble pushed the other way, toward worlds you can see and walk through. Four traditions, each holding a different piece of one problem. They only noticed they were neighbours when the outputs started to resemble each other.



Recover a hidden state from noisy measurements by running a forward model and correcting it against what you observe. The dynamics are supplied, not learned, and nothing here asks what happens if I act.

FIG. 1.4 Scrub the years. Nothing is replaced along the way, only added, relabelled or retargeted, which is why systems built for different reasons ended up sharing a name.

The five definitions

These are contracts, not cages. They describe what a system promises, and modern work crosses between them routinely.

FIG. 1.5 Four are systems you can run, ordered by how abstract the predicted object is. The fifth is not a system, which is why it sits off the axis.

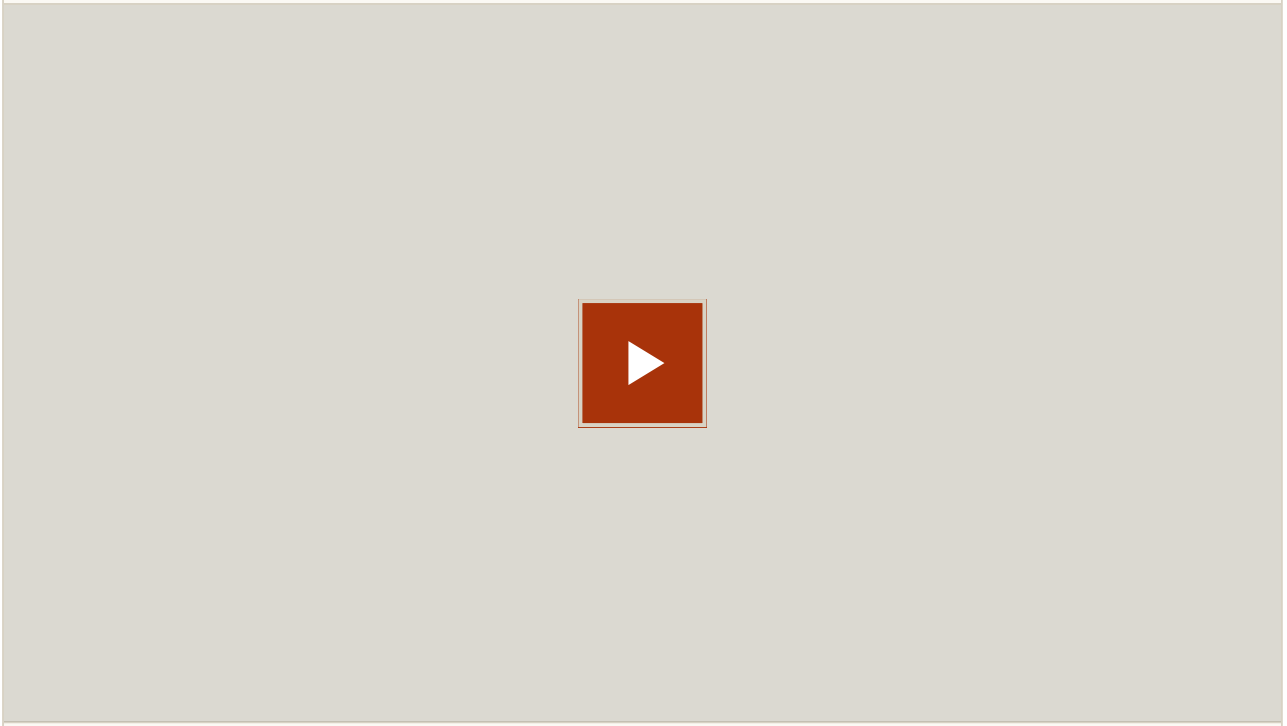
The Renderer predicts observations, usually pixels. GameNGen generates DOOM frames conditioned on previous frames and inputs. Genie 3 is reported to generate interactive video at 720p and 24 frames per second, to stay coherent for several minutes, and to take natural-language interventions mid-session: change the weather, add a flock of birds. Persistence exists, but it is learned through generation rather than guaranteed by stored geometry.

The Simulator predicts queryable structure. Marble takes text, an image or a rough 3-D layout and returns two representations at once. Gaussian splats (millions of coloured blobs, cheap to render, nothing to bump into) carry the appearance. Triangle meshes carry the structure, collider meshes included (the invisible geometry a physics engine actually collides against). World Labs shipped it to limited beta in November 2025 and general availability in February 2026. NVIDIA open-sourced Cosmos, which generates physically-aware video specifically as robot and autonomous-vehicle training data.

Cosmos is the case that keeps the map honest, so it is worth slowing down for. NVIDIA describes Cosmos Predict as generative video, while Omniverse supplies the explicit 3-D simulation. File the whole platform under "simulator" and you hide the distinction worth learning.

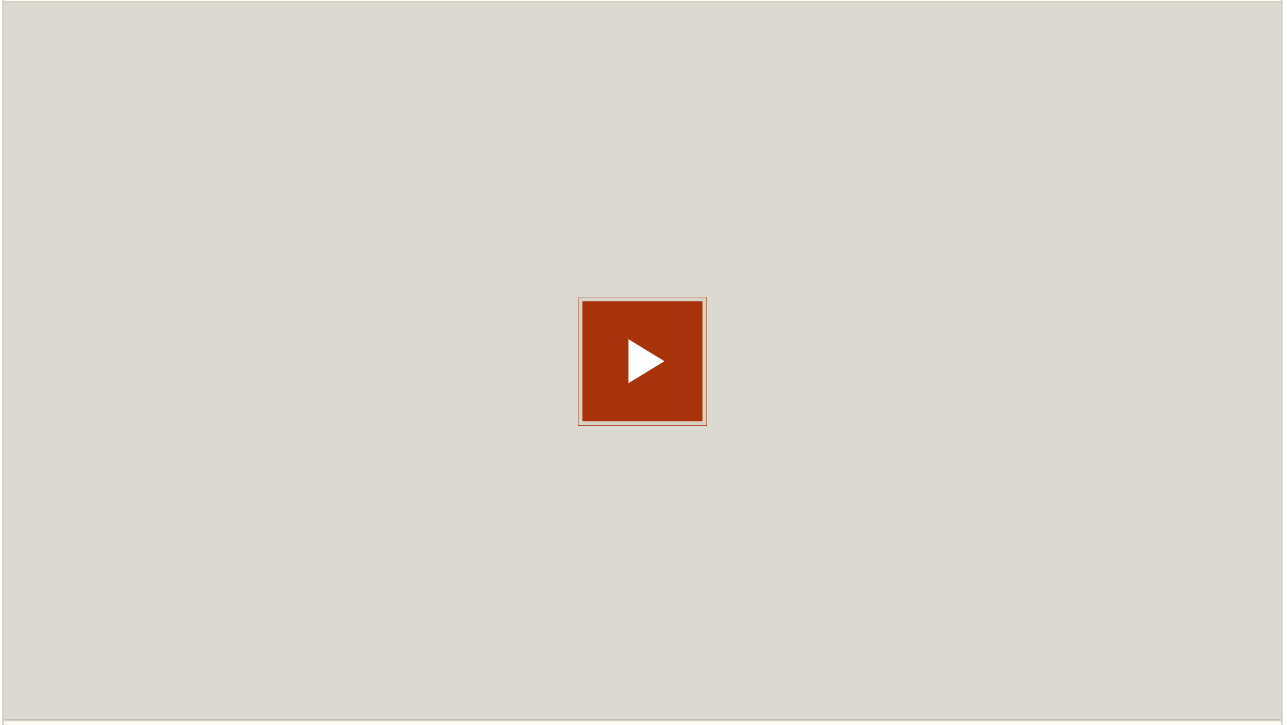
The wider lesson is that a platform is not a category. One product can expose a pixel predictor, a 3-D simulator and an action-conditioned model at the same time, and asking which of the five it *is* will get you nowhere. Ask which one you are being sold, and which interface you are about to build against. The categories describe contracts, and a large enough system signs more than one.

RENDERER / PREDICTS OBSERVATIONS



GOOGLE DEEPMIND Genie 3: Creating dynamic worlds that you can navigate in real-time

SIMULATOR / PREDICTS STRUCTURE



WORLD LABS Introducing Marble by World Labs

FIG. 1.6 Not pass/fail. Marble hands you geometry another program can open, which you can check for yourself; Genie 3's coherence is reported by the lab that built it and cannot be. The difference is what each promises, and how much of it you can verify.

The Dynamics Model predicts compact state and reward under actions. PlaNet plans online in latent space; Dreamer learns behaviour from trajectories imagined through those dynamics. What matters here is not how real it looks. It is whether you can search inside it. Try a hundred possible action sequences, score them, and keep the best one.

A warning about the name, because it trips people who go on to the papers. Ha and Schmidhuber's system has three modules, and they call the third one the controller: the small policy that picks actions. The world model is the module beside it, the one the controller plans inside. So this category is the model, not the controller, and naming it "the Controller" would name it after the one piece that is explicitly not a world model.

The Representation predicts embeddings (a list of numbers standing in for an image or a clip, arranged so similar things sit close together) and then throws the prediction away. I-JEPA hides part of an image and predicts the embedding of the missing piece, rather than redrawing the pixels. V-JEPA 2 pre-trains on more than a million hours of video with no actions involved. A second stage post-trains an action-conditioned variant for model-predictive control (plan a few steps, execute one, re-plan). Meta reports it driving a Franka arm in a lab it never saw during training, and planning around 30× faster than Cosmos. Treat that number as a comparison between different contracts rather than a ranking.

Notice where that second stage lands. Action-conditioned, rolled forward, searched over: that is the Dynamics Model's contract, reached from the other direction. Which is what the categories are for. They describe what a system was trained to predict, not a permanent identity, and mature systems migrate. V-JEPA 2 begins as a Representation and grows a Dynamics Model on top of it.

The Implicit Model is a different kind of claim. Othello-GPT was trained only to predict legal Othello moves and never shown a board. Li et al. (2022) found board state represented internally, recoverable by a probe (a small classifier trained to read one specific fact out of a network's activations). Intervening on that representation

changed the moves the model went on to make. Nanda et al. (2023) later found a closely related linear representation. Nobody handed the model a callable function named `world_model`. The phrase describes evidence about what arose inside another system.

The test

You have already run it.

Figure 1.1 has a switch marked *hold the world*. Turn it off, walk away from the marker, come back: it has moved. Turn it on and the marker is exactly where you left it.

That suggests an easy way to tell these systems apart.

Turn around. Walk away. Turn back. Is it the same room?

It ought to work. Something that really holds a world inside it should keep the furniture where you left it. Something that is only drawing pictures should not be able to.

It does not work, and the reason is worth a minute.

There are two ways to pass this test. You can store the room somewhere and look it up when the viewer turns back, which is what my switch does. Or you can get very, very good at drawing a room that matches the one you drew a moment ago.

From the outside, those are identical. The furniture is still there either way, and nothing you can see from where you are standing tells you which just happened.

A large model trained on a lot of video learns the second way. Nobody gave it a room to keep. It got good at continuing what it had already started, and continuing consistently is part of that.

> DeepMind reports this of Genie 3: scenes holding together over several minutes, and details coming back when you return to them. Take it as their report rather than something anyone outside the lab can check.

So the question cannot sort them. Ask it anyway, because it points at something more useful than the answer it gives.

When you turn away from the chair, the chair does not stop existing. It stops being *visible*. Those are two different things, and nearly everything difficult about this subject lives in the gap between them.

What is actually there is called the **state**. The part of it you can see right now is called the **observation**.

> State is the underlying reality of the world; complete in principle, but never directly visible to any agent inside it. Observations are an agent's partial view of that reality.
>
> World Labs on the distinction underneath the taxonomy (A Functional Taxonomy of World Models, June 2026)
> https://www.worldlabs.ai/blog/taxonomy-of-world-models

You never get the state. You get observations, and you have to work backwards from them. The chair behind you, a ball still rolling somewhere out of shot, whether the cupboard is open: all real, none of it visible. Working out the rest from the part you can see is called **partial observability**, and Chapters 2 through 6 are largely about coping with it.

None of this is new. A Cambridge psychologist described the fix in 1943, long before there was anything to build it out of.

> If the organism carries a 'small-scale model' of external reality and of its own possible actions within its head, it is able to try out various alternatives, conclude which is the best of them, react to future situations before they arise... and in every way to react in a much fuller, safer, and more competent manner to the emergencies which face it.
>
> Kenneth Craik on internal models, forty years before anyone trained one (The Nature of Explanation, 1943)

Carry a small model of the world in your head, and you can test an action before you take it. That is the whole idea, and every definition on the map is a different bet on what the model should contain.

The loop

One object sits underneath all five definitions.

An environment has a state. An agent receives observations revealing part of it. From observations and memory it builds an internal state. A model predicts how things could evolve. The agent acts. The world changes. A new observation arrives.

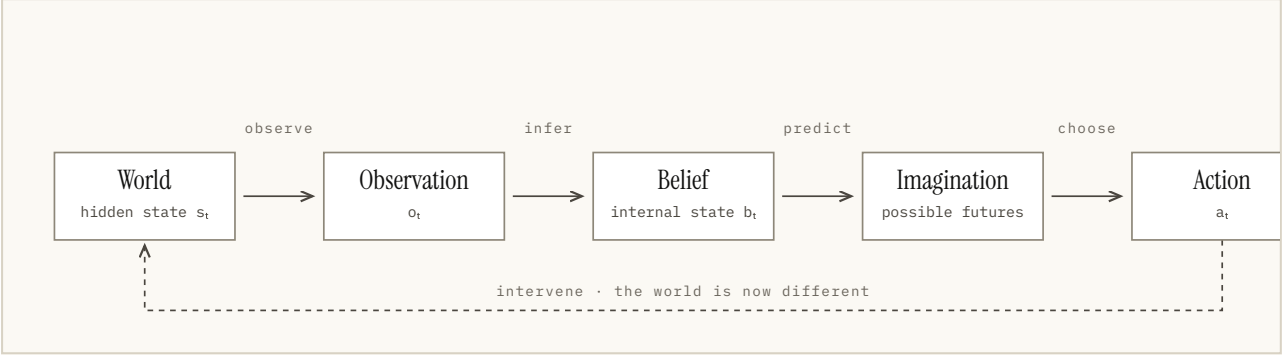


FIG. 1.7 The agent–environment loop, with each definition shown as a specialist on one arc. This is why apparently incompatible systems inherit the same name.

Three questions follow directly, and each gets a chapter.

How sure should it be? Throw a ball behind a wall and ask where it is now. A single exact position is the wrong shape of answer, because you genuinely cannot know. It might have bounced off something, or stopped against a kerb, or still be rolling. A model that hands back one confident coordinate has quietly thrown away the fact that it was guessing.

PlaNet carries two things at once instead. A deterministic part (computed the same way every time, so it always follows from the last state) holds whatever reliably follows from what came before. A stochastic part (sampled rather than computed, so the same input can produce several different futures) holds whatever could still go either way. Being unsure is not the failure. Being sure without grounds is.

Do actions change the prediction? *What happens next* and *what happens if I push this* are different questions. Only the second one lets you compare options before choosing. A model that answers it is called action-conditioned (it is told which action was taken, so it can answer what-if instead of only what-next), and that is what control needs.

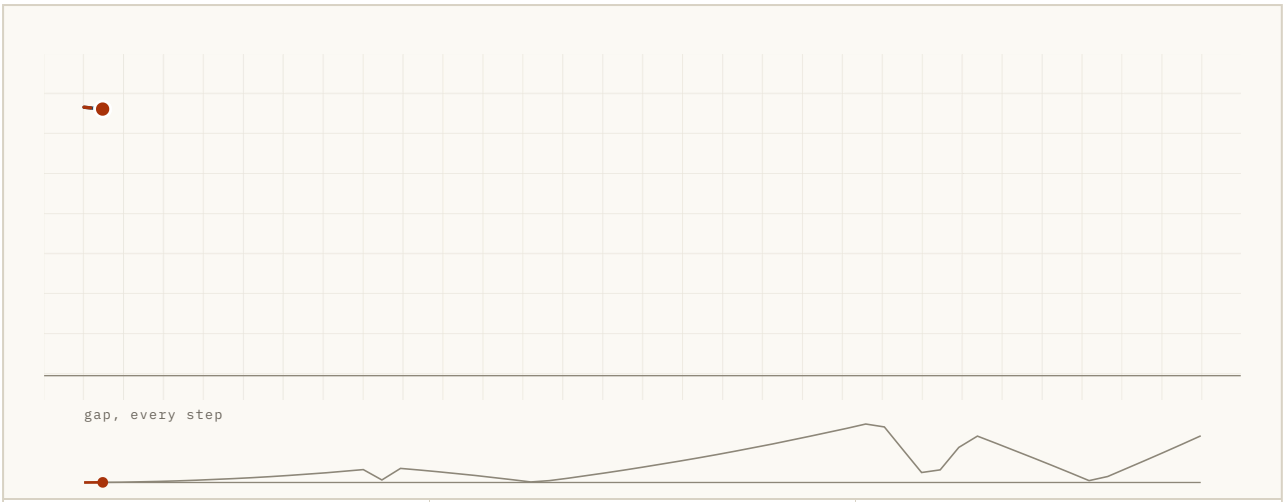
How far can it roll forward? One prediction is rarely enough. To plan, a model has to feed its own answer back in and predict again, and again, on top of what it just made up. The number of steps you ask for is called the **horizon**.

$$p(s_{t+1} \mid s_t, a_t) \implies p(s_{t+1:t+H} \mid s_t, a_{t:t+H-1})$$

Ask for the next state instead of every state from here to H, and you still only get the same single state you are in and a whole plan.

FIG. 1.8 The same model asked for a stretch of future instead of a single step. Two things grow and one does not, which is the whole difficulty in one line.

Errors pile up along that stretch. Nothing dramatic happens at any single step, which is what makes it hard to notice: the model is only ever slightly wrong. Get step one slightly wrong and step twenty can be somewhere else entirely.



STEPS PREDICTED	GAP RIGHT NOW	WORST GAP SO FAR
1	0.0 step-widths	0.0 step-widths

FIG. 1.9 Both dots start in the same place under the same rules, except one is running dynamics that are a few percent off. Drag to about 8 and they are still together. Keep going. Nothing breaks at any step; the gap is the accumulation.

Dreamer exists to learn behaviour across many imagined steps. DeepMind names long-horizon consistency as a central open problem for generated worlds. Same enemy, opposite ends of the field.

Now go back to the clip. It is a Renderer, and you can say why: it predicts observations, it holds the room because it learned to, and nothing in it promises geometry underneath.

The next one that goes past your timeline will be called a world model too. The useful question is not whether it is impressive. It is which of the five it is, and whether the person posting it could tell you.

Where a claim only holds under one of the five, this course says which.

Try it

1 It takes one photo of a kitchen and returns a mesh you can import into a game engine, with collision volumes on the worktops.

`The Renderer · The Simulator · The Dynamics Model · The Representation · The Implicit Model`

> ANSWER The Simulator. Something else can open it and compute against it. Collision volumes are the giveaway: they exist for a physics engine to bump into, not for you to look at.

2 You are watching a camera feed of a room. There is a chair behind the camera. Where does that chair sit?

a. Outside the state, because nothing can see it
b. In the state but not the observation
c. In the observation but not the state
d. In neither, until the camera turns

> ANSWER b. In the state but not the observation. Turning away does not delete furniture. The chair is part of what is actually there, and simply not part of what you can currently see. Every hard problem in this course lives in that gap.

3 You hold a key and it streams video of a city that has never existed, a frame at a time, reacting to which way you steer.

`The Renderer · The Simulator · The Dynamics Model · The Representation · The Implicit Model`

> ANSWER The Renderer. The output is the picture. It may well stay consistent as you drive, but nothing underneath is obliged to, and there is no city to hand anyone.

4 A system keeps the room consistent when you turn away and turn back. What has that proved?

a. It is storing the room
b. It is not storing the room
c. Nothing on its own
d. It must be a Simulator

> ANSWER c. Nothing on its own. There are two ways to pass that test, and from the outside they are identical. You can keep the room, or you can be very good at redrawing it. The result is the same, so the result cannot tell you which.

5 It is trained only to predict the next move in chess games. Researchers later probe it and find it tracks where the pieces are.

`The Renderer · The Simulator · The Dynamics Model · The Representation · The Implicit Model`

> ANSWER The Implicit Model. Nobody built a chess model here and nobody can run one. The claim is about structure found inside a network trained for something else, which is a claim of a different kind from all the others.

6 A model is slightly wrong at every step. You feed its own output back in twenty times. What happens to the error?

a. It stays about the same
b. It roughly doubles
c. It grows, unevenly, and can end up somewhere else entirely
d. It cancels out over enough steps

> ANSWER c. It grows, unevenly, and can end up somewhere else entirely. Each imagined state becomes the input to the next prediction, so mistakes are built on. Nothing dramatic happens at any single step, which is exactly what makes it hard to catch.

7 It hides part of a video and learns to predict a summary of the hidden part. Once trained, the predictions are thrown away and the rest is bolted onto a robot.

`The Renderer · The Simulator · The Dynamics Model · The Representation · The Implicit Model`

> ANSWER The Representation. The forecast was scaffolding. What survives training is the way it learned to describe things, which is the product.

8 Going from predicting one step to predicting H steps, what does NOT get bigger?

a. The stretch of future you are asking for
b. The number of actions you have to supply
c. What you are given to start from
d. The number of ways it can go wrong

> ANSWER c. What you are given to start from. However far ahead you ask, you are still standing in exactly one place with one observation of it. The question grows; the evidence does not.

9 Given the current sensor reading and a motor command you are considering, it returns the sensor reading you would get next. A search loop calls it a few thousand times a second.

`The Renderer · The Simulator · The Dynamics Model · The Representation · The Implicit Model`

> ANSWER The Dynamics Model. Small, fast, and useful only because you can roll it forward under actions nobody has taken yet. Fidelity is beside the point; searchability is the whole point.

10 Why does the Kalman filter count as ancestry rather than as one of the five?

a. It is too old to count
b. Its dynamics are supplied rather than learned
c. It estimates state but is never asked what happens if you act
d. It only works on linear systems

> ANSWER c. It estimates state but is never asked what happens if you act. It does half the job beautifully: work out a hidden state from noisy measurements. What it never does is answer what-if, and conditioning on actions is what makes the rest of these useful for choosing.

11 One era in the history removed a part instead of adding one. Which, and why?

a. The encoder, because pixels stopped mattering
b. The decoder, because the prediction target moved off the pixels
c. The controller, because planning was abandoned
d. The dynamics, because they became implicit

> ANSWER b. The decoder, because the prediction target moved off the pixels. JEPA predicts a summary of the next frame rather than the frame, so nothing needs to turn the prediction back into pixels. That is the same reason the forecast can be discarded and the features kept.

12 A lab generates photorealistic video of motorway driving to train a self-driving stack. It is marketed for robotics.

`The Renderer · The Simulator · The Dynamics Model · The Representation · The Implicit Model`

> ANSWER The Renderer. The trap. Being aimed at robots suggests a Simulator, but the output is still video, with no geometry anyone can collide against. What a system is for is a weaker clue than what it hands you. Not a complete answer either: a large platform can ship several interfaces, so the real question is which one you are about to build against.

13 A lab reports its system stays coherent for several minutes. You cannot run it yourself. How should that sit in your notes?

a. As a fact, since they built it
b. As reported, not checked
c. As false until proven
d. As irrelevant to the category

> ANSWER b. As reported, not checked. Not scepticism for its own sake. Some claims you can open and verify, like a mesh you can load; others you can only receive. Knowing which is which is part of reading this field.

FIG. 1.10 Thirteen questions across the whole chapter: placing systems it never named, and the ideas the figures were built to teach. One is a trap, and it is the mistake worth making here rather than in a meeting.

Thanks for reading. Chapter 2 deals with the Dynamics Model in full: what a learned simulator is, what having one buys you, and why the idea is eighty years old and only recently any good.

Sources

A Functional Taxonomy of World Models WORLD LABS, 2026 ↗

First-party renderer/simulator/planner split, derived from the agent loop.

https://www.worldlabs.ai/blog/taxonomy-of-world-models

A New Approach to Linear Filtering and Prediction Problems KALMAN, 1960 ↗

The hidden-state ancestor. Paywalled.

https://doi.org/10.1115/1.3662552

Recurrent world models for planning and curiosity SCHMIDHUBER, 1990 ↗

A recurrent model predicting the consequences of a controller's actions.

https://people.idsia.ch/~juergen/world-models-planning-curiosity-fki-1990.html

World Models HA & SCHMIDHUBER, 2018 ↗

The paper that popularised the modern label.

https://arxiv.org/abs/1803.10122

Learning Latent Dynamics for Planning from Pixels (PlaNet) HAFNER ET AL., 2018 ↗

Raw pixels to stochastic latent state to online planning.

https://arxiv.org/abs/1811.04551

Dream to Control (Dreamer) HAFNER ET AL., 2019 ↗

Behaviour learned from multi-step latent imagination.

https://arxiv.org/abs/1912.01603

I-JEPA ASSRAN ET AL., 2023 ↗

Predicting representations of masked regions, not pixels.

https://arxiv.org/abs/2301.08243

V-JEPA 2 META AI, 2025 ↗

Action-free pre-training, then action-conditioned control.

https://ai.meta.com/blog/v-jepa-2-world-model-benchmarks/

Genie: Generative Interactive Environments BRUCE ET AL., 2024 ↗

Action-controllable generated environments.

https://arxiv.org/abs/2402.15391

Genie 3 GOOGLE DEEPMIND, 2025 ↗

Reports recalling previously seen detail over multi-minute interaction.

https://deepmind.google/blog/genie-3-a-new-frontier-for-world-models/

Marble WORLD LABS, 2025 ↗

Gaussian splats plus collider meshes: an explicit structural export.

https://www.worldlabs.ai/blog/marble-world-model

Cosmos NVIDIA ↗

A boundary case: predictive video worlds beside explicit simulation.

https://www.nvidia.com/en-us/ai/cosmos/

Emergent World Representations LI ET AL., 2022 ↗

Board state found and causally manipulated inside Othello-GPT.

https://arxiv.org/abs/2210.13382

Othello-GPT has a linear emergent world representation NANDA ET AL., 2023 ↗

The follow-up that sharpened the finding.

https://arxiv.org/abs/2309.00941

FIG. 1.11 Ordered for someone building on this rather than for historical completeness. First-party material preferred throughout.