THE UNREAL RADAR · ENGINE RADAR

The Unreal Engine Feature Radar

Which of Epic's UE5 subsystems are production-ready, which to pilot, which to watch, and which to leave for new work — judged against the current 5.7 release.

Engine Radar · Edition 2026.1

2026-06-13

Edited by Phil, MythicLemon

mythiclemon.com/radar

Editor’s note

Welcome to the first edition of the Unreal Engine Feature Radar. The idea is simple: for the subsystems that actually shape how a UE5 project is built, say plainly whether we'd ship on them today. We rate **Epic's own engine** here — not anyone's plugins or assets — so there is nothing to sell you and no rival to mark down. Every verdict is anchored to Epic's published maturity status and to building on these systems ourselves.

This edition is cut against **Unreal Engine 5.7**, the current stable release, with one eye on the **5.8 preview**. 5.7 was a genuinely consequential release for this radar: **Substrate**, **PCG** and **Motion Design** all crossed into production-ready, **MegaLights** reached beta, and an experimental **Nanite Foliage** path arrived. 5.8 (in preview as we write, with a stable window around mid-2026) promises to push MegaLights to production and add a Lumen medium-quality mode — but a preview is a preview, and you'll see us hold those one ring back until they ship.

A word on the rings. **Adopt** means we'd reach for it on a new production without hesitation. **Trial** means it's ready enough to commit to on a project that can absorb a little risk. **Assess** means it's worth your time to understand, but not yet to depend on. **Hold** means: fine where you already use it, but don't start new work on it. These are production-readiness calls, not quality judgements — a feature we place in Assess may be the most exciting thing in the engine; it just isn't something we'd bet a shipping date on yet.

Because this is the first edition, nothing carries a movement arrow — there is no previous radar to have moved from. From the next edition on, an arrow will mark where *we* changed our minds, which is a different thing from where Epic changed a label. See the matrix below for exactly where our verdict and Epic's status diverge today.

— Phil, MythicLemon

> **This is our view, not a measurement.** The rings are a considered production-readiness opinion — not a quality score. We rate Epic’s engine, asset categories and practices, never a named third-party product. Where a category overlaps something we sell, we say so.

The radar at a glance

Rendering & Lighting · Worldbuilding & Streaming · Gameplay Frameworks · Animation, Audio & VFX

Adopt
We'd use it on a new production without hesitation.

1 Nanite · 2 Lumen · 3 Virtual Shadow Maps · 7 World Partition · 8 PCG (Procedural Content Generation) · 11 Enhanced Input · 12 Gameplay Ability System · 17 Niagara · 18 MetaSounds · 19 Control Rig · 22 Chaos Physics

Trial
Ready enough to commit to where you can absorb some risk.

4 Substrate · 5 MegaLights · 6 Hardware Ray Tracing · 13 State Tree · 15 CommonUI · 20 Motion Matching · 21 MetaHuman

Assess
Worth understanding now — not yet worth depending on.

9 Nanite Foliage · 10 Mesh Terrain · 14 Mass Entity · 16 Verse (in mainline UE)

Hold
Fine where you use it; don't start new work on it.

— nothing here this edition —

Where our verdict diverges from Epic's status

Feature	Epic says (5.7)	We say	Why the gap
Substrate	Production	**Trial**	Framework is ready; third-party content and migration cost lag a release behind.
MegaLights	Beta	**Trial**	Beta in the *stable* engine; we won't promote it on a 5.8 preview. Likely Adopt when 5.8 ships.
PCG	Production	**Adopt**	We agree — the core framework earned the label; only the newer biome/vegetation edges are less settled.
State Tree	Production	**Trial**	Solid, but newer to most teams' toolkits — commit after a project's experience.
MetaHuman	Production	**Trial**	Core is production-grade; the in-engine toolset is still settling release-to-release.
Verse (mainline)	Experimental	**Assess**	A shipping reality in UEFN; still experimental and rolling out for standalone UE projects.
Nanite Foliage	Experimental	**Assess**	We agree it's not production yet — strong trajectory, worth prototyping.

Epic status as of UE 5.7 (current stable); 5.8 is in preview. "We say" is our production-readiness opinion, not a measurement.

The evidence (live online)

AS OF MAY 2026

UE5 is where new Unreal content is being made. Why this radar matters commercially: on the Fab marketplace, UE5 now accounts for the overwhelming majority of each month's new releases. The features above are what that majority is built on.

↗ Live, interactive figure (with downloadable data): https://www.mythiclemon.com/data/dashboards/engine-migration

The verdicts

Rendering & Lighting

How the frame gets drawn and lit. The most mature corner of UE5 — and the one moving fastest at the high end.

1 Nanite ADOPT Epic: Production

The virtualised geometry system is the safe default for static detail.

Nanite has been production-grade since UE5.0 and has only widened its remit since. For static, high-density meshes it is the default we reach for — it removes the LOD-authoring grind and lets artists work at source fidelity. 5.7's move to bring **first-person rendering with Nanite** to production closes one of the last awkward gaps.

The judgement to make is no longer *whether* to use Nanite but *where*: skeletal meshes, world-position-offset foliage and heavy translucency remain the places to measure rather than assume. Profile, don't guess — but start from Nanite-on.

Sources: Epic · UE 5.7 release notes

2 Lumen ADOPT Epic: Production

Dynamic global illumination that's matured into a dependable default.

Lumen is the other half of the modern UE5 look and is firmly production-ready. Software Lumen scales down to current-gen consoles at 60Hz with care; hardware ray tracing buys sharper reflections and cleaner interiors where you have the GPU budget, and 5.7 continued to improve both.

It is not free — Lumen rewards teams who treat lighting as a performance discipline, budget for it early and validate on target hardware. But as a starting point for dynamic lighting on a new project, it's the right one.

Sources: Epic · UE 5.7 release notes

3 Virtual Shadow Maps ADOPT Epic: Production

The shadowing path designed for Nanite-scale geometry.

Virtual Shadow Maps are the shadowing counterpart to Nanite and Lumen, and like them they're production-ready and the expected default in a modern UE5 renderer. They deliver consistent, high-resolution shadows across large scenes without the manual cascade tuning of the past.

Cost lives in overlapping casters and heavy foliage; the mitigations are well understood and documented. Adopt, and keep an eye on the VSM stats when scenes get dense.

Sources: Epic · UE 5.7 release notes

4 Substrate TRIAL Epic: Production

The new layered material framework — production in 5.7, but give your pipeline a beat to catch up.

Substrate replaces the fixed shading models with a layered, principled framework, and Epic moved it to **production-ready in 5.7**. The capability is real and the future of UE materials clearly runs through it.

This is the sharpest **Epic-says-Production, we-say-Trial** call on the radar. The framework is ready; the *ecosystem* around it — third-party materials, team muscle memory, and the cost of converting an existing project — is still catching up. For a brand-new project, trial it with intent. For a mid-flight one, plan the migration deliberately rather than flipping it on.

Sources: Epic · UE 5.7 release notes

5 MegaLights TRIAL Epic: Beta

Many shadow-casting dynamic lights, cheaply — beta now, production in the 5.8 preview.

MegaLights lets you place far more shadow-casting dynamic lights than was previously affordable, and it has moved quickly: experimental, then **beta in 5.7** with directional lights, Niagara particle lights, translucency and hair support. The 5.8 preview pushes it to production-ready.

We place it in **Trial**: beta in the current *stable* engine is exactly the 'commit where you can absorb risk' case, and we don't promote a feature on the strength of a preview build. When 5.8 ships, expect this to move to Adopt — it's one of the most promising lighting changes in years.

Sources: Epic · UE 5.7 release notes

6 Hardware Ray Tracing TRIAL Epic: Production

Worth it at the high end; budget the GPU before you commit.

Hardware Lumen and ray-traced reflections/shadows are production-ready and deliver a visible quality step on capable GPUs. 5.7's reflection and interior-lighting improvements make the case stronger again.

We hold it at **Trial** rather than Adopt only because the decision is so target-dependent: it's an easy yes for high-end PC and current-gen console showcases, and an easy no for the lower end of your hardware spread. Decide by your minimum spec, not your hero shot.

Sources: Epic · UE 5.7 release notes

Worldbuilding & Streaming

Building and streaming large worlds: partitioning, procedural generation and the new foliage and terrain paths.

7 World Partition ADOPT Epic: Production

The standard way to build and stream a large UE5 world.

World Partition, with Data Layers and Level Instances, is the production-ready, expected approach to large worlds in UE5 — it replaced the old sublevel workflow and is what Epic's own large-world tooling is built around. 5.6 and 5.7 both invested in making big worlds run smoothly at 60Hz.

Adopt for anything open-world or simply large. The learning curve is real (One File Per Actor changes how teams collaborate in source control), but it's the right default and the ecosystem assumes it.

Sources: Epic · UE 5.7 release notes

8 PCG (Procedural Content Generation) ADOPT Epic: Production

Epic's procedural framework crossed into production-ready in 5.7.

PCG graphs let you scatter, place and procedurally build world content non-destructively, and Epic designated the framework **production-ready in 5.7**, alongside continued work on biome and vegetation tooling. It has gone from an interesting experiment to a dependable part of the worldbuilding pipeline in a short time.

We agree with the Production label and **Adopt** it. Treat the newer biome/procedural-vegetation editors as the less-settled edge (see Nanite Foliage), but the core PCG framework is ready for real work.

Sources: Epic · UE 5.7 release notes

9 Nanite Foliage ASSESS Epic: Experimental

Promising experimental path for Nanite-density vegetation — not yet to depend on.

Nanite Foliage and the accompanying Procedural Vegetation Editor arrived as **experimental** features in 5.7, aiming to bring Nanite's density advantages to vegetation — historically Nanite's hardest case. The early results are exciting and the direction is clearly right.

Experimental means exactly that: **Assess** it, prototype with it, file feedback — but don't build a shipping vegetation pipeline on it this release. This is a strong candidate to climb the radar quickly over the next version or two.

Sources: Epic · UE 5.7 release notes

10 Mesh Terrain ASSESS Epic: Experimental (5.8 preview)

A mesh-based large-landscape system — preview-only, watch it land first.

Mesh Terrain is a new experimental system in the **5.8 preview** for building large-scale landscapes from a 3D mesh rather than a heightfield — a meaningful rethink of the terrain pipeline if it matures.

It is **Assess**, and barely on the board: it exists only in a preview build today. We're flagging it so you know it's coming, not recommending it for any current work. Revisit when 5.8 ships.

Sources: Epic · UE 5.7 release notes

Gameplay Frameworks

The systems you build mechanics on. Powerful, long-lived, and uneven on documentation.

11 Enhanced Input

ADOPT | Epic: Production

The modern input system — the default for any new project.

Enhanced Input is the production-ready, expected way to handle input in UE5, with input actions, mapping contexts and modifiers replacing the legacy action/axis mappings. It's mature, well-trodden and assumed by most current learning material.

There's no reason to start a new project on the old system. Adopt without reservation.

Sources: Epic · UE 5.7 release notes

12 Gameplay Ability System

ADOPT | Epic: Production

Battle-tested for abilities and stats — just budget for the learning curve.

The Gameplay Ability System is shipped-in-AAA, production-grade code for abilities, attributes and effects, with networking handled. For anything with skills, cooldowns, buffs or replicated stats, it saves you reimplementing a hard problem badly.

Our one honest caveat keeps it from being an unqualified Adopt for every team: official documentation remains thin, so the real cost is the ramp-up. With a team willing to climb that curve (or lean on the strong community material), Adopt.

Sources: Epic · UE 5.7 release notes

13 State Tree

TRIAL | Epic: Production

A clean, general state machine that's matured into production use.

State Tree gives you a hierarchical, data-oriented state machine usable well beyond AI — UI flow, gameplay states, sequencing. It reached production-ready maturity over recent releases and is increasingly Epic's recommended structure for this kind of logic.

We place it at **Trial**: it's solid and we'd commit to it, but it's newer to most teams' toolkits than the systems above it, so spend a project getting the patterns right. Pairs naturally with Mass and GAS.

Sources: Epic · UE 5.7 release notes

14 Mass Entity ASSESS Epic: Experimental

Data-oriented ECS for huge agent counts — powerful, still experimental.

Mass is UE5's data-oriented (ECS-style) framework for simulating very large numbers of entities — crowds, traffic, swarms — and it's genuinely capable at scale. Epic uses it in its own large-world demos.

It remains **experimental**, the API still shifts between versions, and the documentation is sparse — so it's an **Assess**. If you have a real high-agent-count problem and engineers comfortable on the bleeding edge, prototype with it; otherwise watch it and don't build core gameplay on it yet.

Sources: Epic · UE 5.7 release notes

15 CommonUI TRIAL Epic: Production

The right tool for multi-platform, controller-friendly UI — with a real ramp.

CommonUI is production-ready and the sensible foundation for UI that must work across mouse, gamepad and touch with proper focus navigation and input routing — the things plain UMG leaves you to solve yourself. Epic's own titles ship on it.

We say **Trial** rather than Adopt because the up-front complexity is significant and overkill for a small single-platform UI. For anything console or cross-platform, it's worth the climb; size the decision to your UI's real scope.

Sources: Epic · UE 5.7 release notes

16 Verse (in mainline UE) ASSESS Epic: Experimental

Epic's new language is real in UEFN — but experimental for a standard UE project.

Verse is Epic's programming language and a clear strategic direction for the engine's future. Inside **UEFN / Fortnite** it's a shipping reality. For a standard, standalone Unreal Engine project, however, its availability is still rolling out and experimental.

So the verdict splits by context. If you're building for UEFN, Verse is simply part of the platform. If you're building a standalone UE5 game, treat Verse as **Assess** — learn it, watch the roadmap, but keep C++ and Blueprint as your production languages for now. This is the entry most likely to be re-judged as the rollout proceeds.

Sources: Epic · UE 5.7 release notes

Animation, Audio & VFX

The content pipelines: particles, sound, rigging, motion and digital humans.

17 Niagara ADOPT Epic: Production

The mature, capable VFX system — the default for particles and effects.

Niagara is the production-ready VFX system in UE5 and has been for the whole 5.x line. It's deep, GPU-capable, and the assumed target for modern Unreal effects work; the old Cascade system is legacy (see our Migration radar).

Adopt without hesitation for any new effects work. It rewards investment in reusable module and emitter libraries.

> ⚖ We sell Niagara VFX packs on Fab, so we have a commercial interest in Niagara's health. We'd place it in Adopt regardless — it's Epic's production VFX system and the only sensible default — but you should know the overlap.

Sources: Epic · UE 5.7 release notes

18 MetaSounds ADOPT Epic: Production

Procedural audio done right — the default sound source in UE5.

MetaSounds is the production-ready, node-based audio system that replaced the old Sound Cue workflow, giving you sample-accurate, fully procedural control over sound generation. It's mature and the expected way to author audio in UE5.

Adopt for new audio work. It's one of the cleaner 'new system fully replaced the old one' stories in the engine.

> ⚖ We sell audio packs on Fab. The Adopt verdict reflects MetaSounds' maturity, not our catalogue — but the overlap is worth disclosing.

Sources: Epic · UE 5.7 release notes

19 Control Rig ADOPT Epic: Production

In-engine rigging and procedural animation, production-ready and widely used.

Control Rig is the production-ready system for building rigs and procedural animation directly in-engine, underpinning a lot of UE5's modern animation workflow (including much of the MetaHuman and motion tooling).

Adopt for in-engine rigging and runtime procedural animation. It's well-established and the surrounding animation tools assume it.

Sources: Epic · UE 5.7 release notes

20 Motion Matching TRIAL Epic: Production

Data-driven locomotion that's gone from demo to dependable.

Pose Search / Motion Matching gives you high-quality, data-driven locomotion without hand-built blend trees, and Epic's Game Animation Sample turned it from a curiosity into something teams genuinely ship. The tooling has matured well across recent releases.

We place it at **Trial**: it's ready to commit to, but it changes how your animation team works and needs a good motion dataset to shine, so adopt it deliberately on a project that can invest in the pipeline.

Sources: Epic · UE 5.7 release notes

21 MetaHuman TRIAL Epic: Production

Now an in-engine, free character pipeline — strong, with a settling toolset.

MetaHuman matured significantly when its creation tools moved in-engine (from 5.6 onward) and the licensing opened up, making high-quality digital humans and their animation pipeline far more accessible. The character and animation tooling around it keeps improving each release.

We say **Trial**: the core is production-grade and we'd happily build on it, but the in-engine toolset is still settling release-to-release, so pin your engine version and validate your pipeline rather than assuming stability across upgrades.

Sources: Epic · UE 5.7 release notes

22 Chaos Physics ADOPT Epic: Production

The engine's physics — production for the core, more variable at the edges.

Chaos is UE5's physics system and fully replaced PhysX; the core rigid-body simulation is production-ready and the default. For standard physics it's a non-decision — it's simply what the engine uses.

Adopt for core physics. The specialised sub-systems (destruction, cloth, flesh) vary in maturity from solid to experimental — judge those case by case rather than assuming the whole Chaos umbrella is equally mature.

Sources: Epic · UE 5.7 release notes

FAQ

Which engine version is this edition based on?

Unreal Engine 5.7, the current stable release as of June 2026, with notes on the 5.8 preview where a feature is about to change status. We won't promote a feature to a higher ring on the strength of a preview build alone.

Why don't you rate any plugins or marketplace assets here?

We sell on Fab, so rating rival products would be marking our own homework. This radar rates Epic's own engine, where we have no commercial stake. See our charter for the full rule.

What's the difference between 'Hold' and a feature simply not appearing?

Hold means a feature is being superseded and we wouldn't start new work on it — those mostly live in our companion Migration & Deprecation radar. A feature absent here just isn't covered this edition; the radar grows each release.

How often does this come out?

Roughly twice a year, tracking Epic's major releases. From the next edition, movement arrows will show where we've changed our minds — which is deliberately separate from where Epic changed a label.

HOW TO CITE

```
MythicLemon, "The Unreal Engine Feature Radar", edition 2026.1 (2026-06-13). The Unreal Radar. https://www.mythiclemon.com/radar/engine-radar-2026-1
```