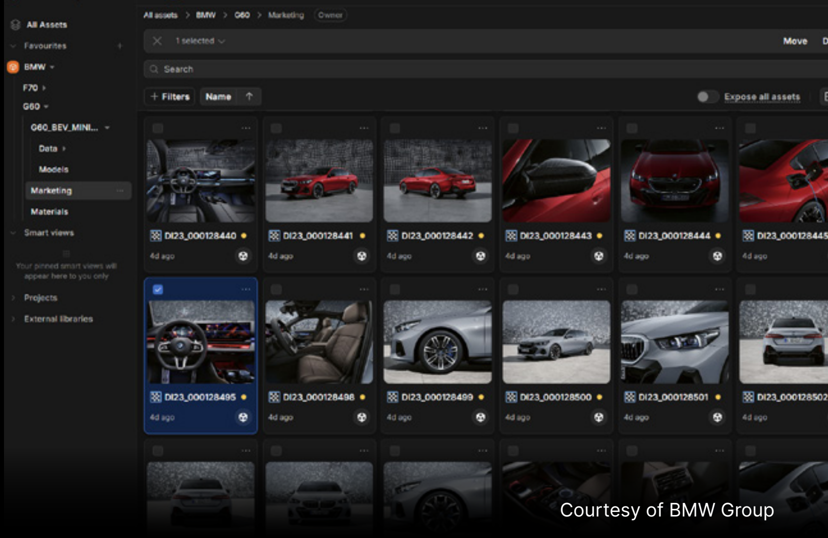
→ PLAYBOOK

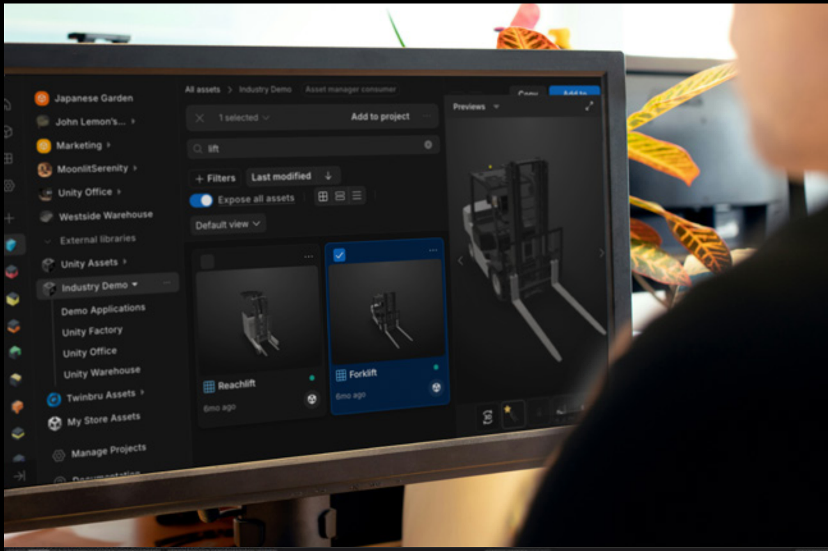
Establishing one source of truth for 3D asset data

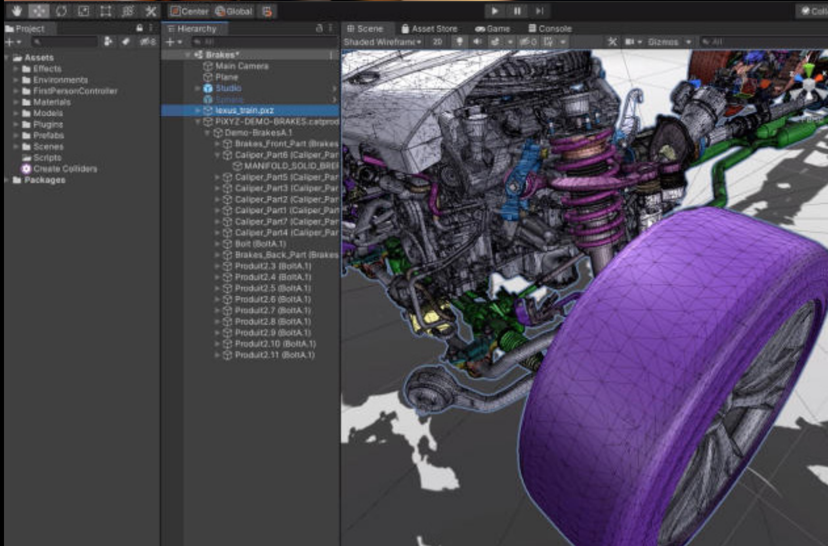
How to connect your teams to the data they need: A playbook for engineering, training and operations teams seeking to unify 3D assets across the enterprise.

Unity®



Courtesy of BMW Group





Introduction

Most industrial teams have a wealth of 3D asset data, but it's stubbornly scattered across silos, owned by different groups and hard to reuse. That's a problem because, by definition, when your data lives in different places, people make decisions based on different truths. For example, a trainer might hand a new hire a VR simulation without knowing that engineering updated the line layout sometime earlier. The trainer has to either stop to rebuild with the latest data or continue with content they can't trust. Both cost time and credibility.

The solution is a single repository for all 3D assets that keeps hierarchy and metadata intact and grants everyone who needs it access to approved models only. When everyone pulls assets from the same library, you eliminate version drift and rework, while also gaining visibility into what changed and who changed it.

This playbook shows how to connect all your 3D data sources into a common library that keeps what's important at the time of import and prepares assets for easy access and use across roles. The goal isn't to deploy a new tool for its own sake but to reduce the time spent on nonvalue-added work and avoid unnecessary management complexity when applying access and compliance policies.

The disconnect: Scattered data and duplicate effort

Industrial data is everywhere but rarely connected. “A designer or architect might have access to CAD or BIM, but an engineer might not have the software, so teams end up re-creating models from scratch,” said Sarah Lash, senior vice president and general manager, industry, at Unity.

Silos often occur because of operational or technological gaps. For example, engineering might store CAD files in a product life cycle management (PLM) system, whereas other teams export copies to separate drives or apps to the point you end up with multiple versions of the same asset. “The most common issue is working from two different versions of the same model, which practically guarantees rework,” Lash said.

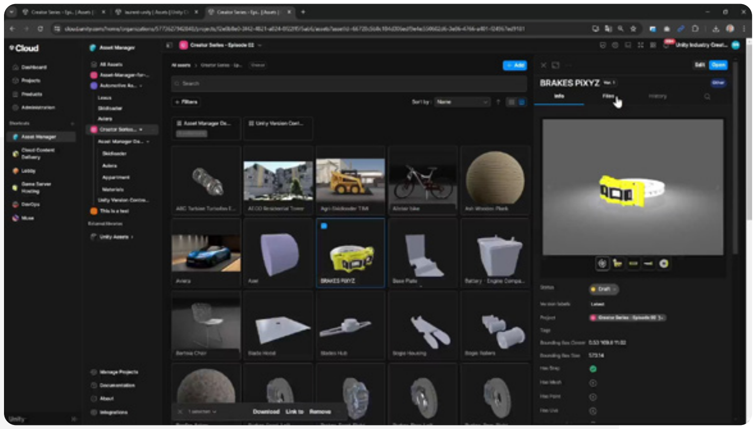
The use of multiple platforms, tools and formats, while often necessary, makes it difficult to track and share assets. This hampers productivity, with people spending hours searching for the right assets or even rebuilding ones that already exist because there’s no easy way to tell which one is the correct one.

As Henning Linn, senior director, industry customer success, at Unity, said: “If there’s a high risk of duplication, there’s often a need for rework, because assets aren’t aligned, and teams aren’t working from the same source file.”

The asset management complexity only increases as these mismatches surface downstream, such as during training and delivery, where they result in rework, missed deadlines and inconsistent experiences. But the cost isn’t just technical. It also undermines confidence between departments, as well as between brands and customers themselves. After all, trainers would rather avoid launching sessions using outdated or unverified models, even if it leads to extended and inconsistent onboarding for new hires. Other teams might end up building “shadow” libraries to stay productive, while IT scrambles to secure and govern an environment it can’t see. Without a common foundation, every new project has to start from scratch.

What to connect: Creating a centralized repository that preserves what matters

Your 3D asset library likely spans many tools and platforms, so the first step to leveraging the models you have is to import everything into a centralized repository like Unity Asset Manager. This way, you won't have to rebuild anything you already have. "We make sure we can import all the relevant data in a preferred and qualitative way, and there's practically no file format in the industry that we can't support," Linn said. "By converting the data into a harmonized format, enriching it as required and tracking every change, it becomes easier to manage those assets over their life cycles."



Before unifying your data, you need to understand what you're connecting and why. Most industrial 3D pipelines draw data from four main sources, each with different structures and priorities:

- Computer-aided design (CAD) models, typically stored in PLM systems, are the primary source of record for parts, assemblies and mechanical properties.
- Building information modeling/management (BIM) models include building and infrastructure data, are rich in spatial and compliance metadata, and are usually stored in BIM software or AEC (architecture, engineering and construction) repositories.
- Meshes from digital content creation tools include visualization assets used in areas such as marketing, training and user experience and are often optimized for visual fidelity rather than technical detail.
- Point clouds and scans, common in extended reality (XR) and virtual reality (VR) applications, include data captured from things such as laser scanning and photogrammetry.

Each source represents the same physical assets — a product, assembly line or entire facility — from a different perspective, such as function, space, appearance or, in the case of point clouds for XR, as-built physical reality.

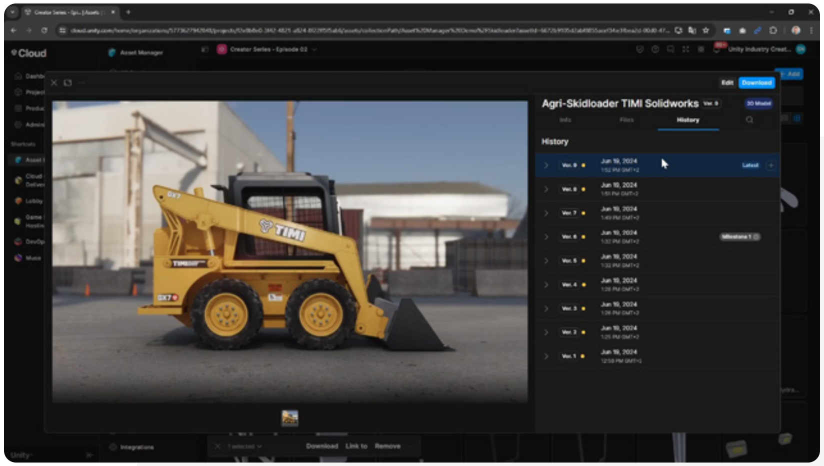
Start with the models your teams reuse most since you'll likely need to prioritize depending on the size of your asset library. A common requirement, no matter the source, is to preserve context during import, because losing vital metadata will only necessitate rework later. Before importing, identify the metadata fields you can't afford to lose. "Always keep identifiers and technical attributes, such as height and weight, and make sure to name and version everything so that assets stay traceable and usable," Linn recommended.

Some fields should always come through intact:

- Identifiers such as part numbers, version codes or unique IDs make it easier to trace assets back to their original sources for verifying their authenticity.
- Hierarchy and grouping metadata, rather than a merged mesh, show how parts fit together or how facilities are organized, allowing engineers to isolate and swap parts as needed.
- Material and unit information, such as density or tensile strength, and descriptive information, such as name or vendor, ensure assets maintain the correct appearance and behavior.

Preserving these details keeps the models useful across applications, whereas without them, they're little more than a picture for quick reference that's disconnected from its purpose.

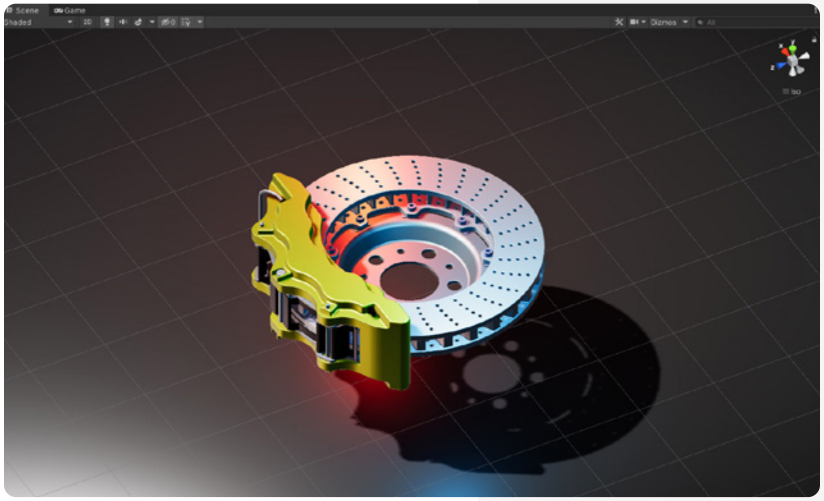
> "Always keep identifiers and technical attributes, such as height and weight, and make sure to name and version everything so that assets stay traceable and usable."
>
> Henning Linn
> Senior Director, Industry Customer Success, Unity

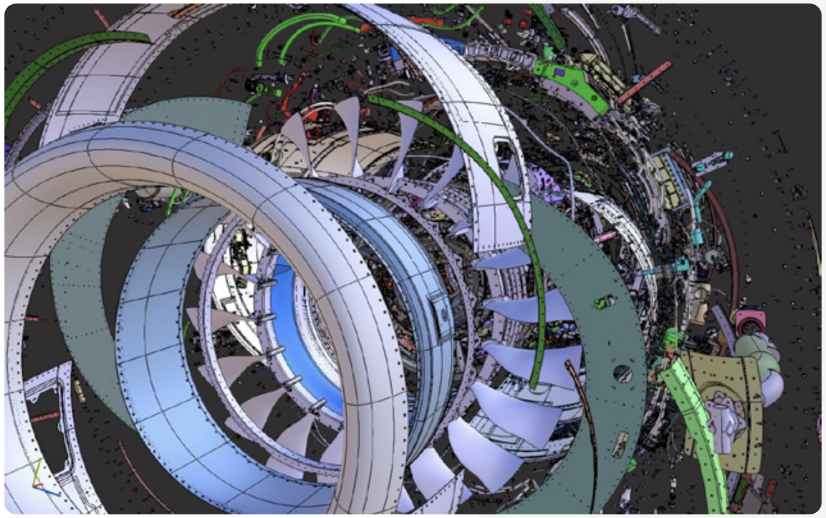

Cloud
Asset Manager
Creator Series - Episode 02
Unity Industry Creat...
Edit
Download
Agri-Skidloader TIMI Solidworks
Ver. 9
3D Model
Info
Files
History
History
Ver. 9 Jun 19, 2024 1:52 PM GMT+2 Latest
Ver. 8 Jun 19, 2024 1:51 PM GMT+2
Ver. 7 Jun 19, 2024 1:49 PM GMT+2
Ver. 6 Jun 19, 2024 Milestone 1
Ver. 5 Jun 19, 2024
Ver. 4 Jun 19, 2024
Ver. 3 Jun 19, 2024
Ver. 2 Jun 19, 2024 1:25 PM GMT+2
Ver. 1 Jun 19, 2024 12:58 PM GMT+2
Download
Link to
Remove

Preparing your assets for import

Assets must be properly prepared before and during import, and there are some common pitfalls to avoid to maintain the health of your repositories. That's especially true for large-scale manufacturing or construction, where 3D datasets can be enormously complex. For instance, a digital twin of an entire factory or a complete model of an automobile might consist of tens or even hundreds of thousands of parts. This can easily overwhelm even the most powerful software and hardware, so breaking everything down into smaller logical groups — such as microchips, connectors or mechanical components — helps keep imports manageable.

Software that prepares 3D data — such as Unity's Asset Transformer Toolkit — is designed to handle these challenges by supporting many widely used CAD and BIM formats, preserving structure and metadata and automatically simplifying and standardizing models during import as needed. For instance, a real-time use case such as an XR simulation for employee training doesn't need every bolt or rivet from the original CAD file. What matters in this case is that the level of realism supports the task. "Depending on the end point, polygon production can range from modest cuts to as much as 90%," Linn said.





The goal is to keep models as light as possible to optimize performance and usability:

- For training and simulation use cases, the goal is to achieve as much visual fidelity as possible at smooth frame rates. Aim for stable frame rates that match the headset's refresh rate, because anything lower can lead to user discomfort.

- 3D collaboration and design reviews typically require high functional detail, allowing engineers to inspect things such as fasteners and interfaces. To better manage the demand for high polygon counts, use levels of detail (LODs) for subassemblies that render only in close proximity.

- Human-machine interfaces such as embedded systems and industrial controls typically have limited graphics processing power, so it's best to target the lowest possible complexity, using prebaked lighting and simpler shaders where feasible.

- Customer experience applications can be difficult to optimize for because of the huge diversity in target devices. Aim for a compromise between visual fidelity and loading times, and validate on midtier mobile devices and popular web browsers.

As a rule of thumb, it's best to start with a simplified model and test it on the weakest device you plan to support. Then add detail only where performance allows to avoid having to fix a bloated model after it's already been deployed. For instance, you can generate LODs at import so every asset ships with scalable detail, allowing you to accommodate a wider range of devices and use cases later on as needed. There's no universal correct polygon count, however. What matters is the one that reliably hits your frame-rate and loading-time goals on the target devices. As Linn said, "The range can go from several million polygons to just a few 100,000 for the same asset. What's important is that you're still using the same source file, including all the metadata connected to it."

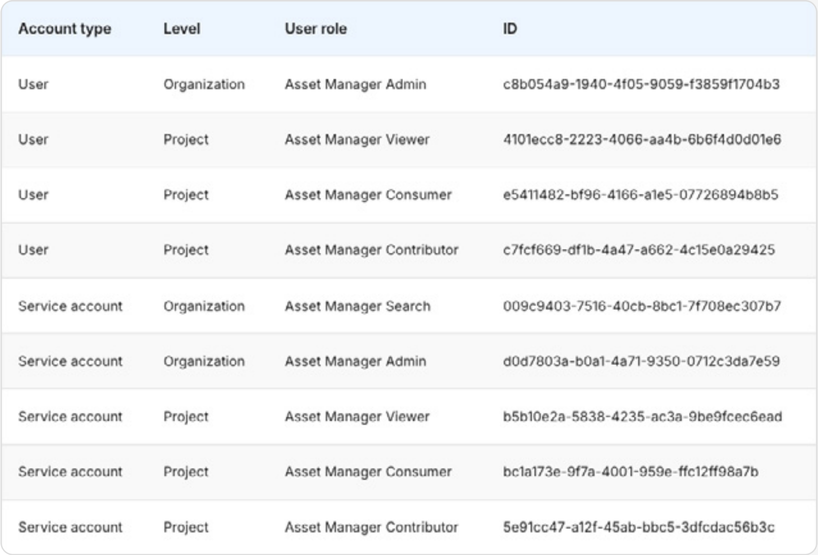
Courtesy of HERE HMI

Make it usable: Building in governance, accessibility and versioning

With your assets imported and right-sized, the next step is making them secure and discoverable for every role that needs them. The goal here is to curate a single library where your teams know where to look without being overburdened with assets that aren't relevant to them. You need a permission-based access model that doesn't slow people down and lets you maintain a complete audit trail to ensure that every update propagates across your projects.

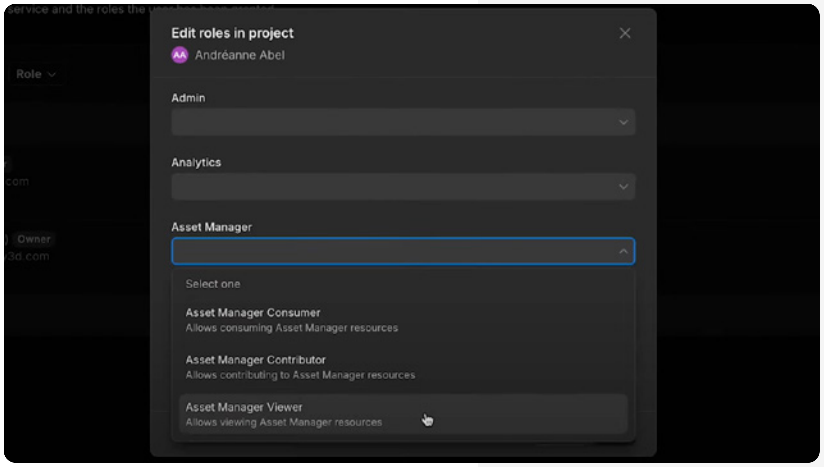
Role-based access controls (RBAC)

Linn recommended keeping access permissions simple. "You'll have an administrator that oversees the import and creation of assets, and then you'll have contributor and reviewer roles, and that's normally enough." For example, have an admin to define structure and standards, manage users and retention, and approve or archive versions. Keep this user group as small as possible.



Account type	Level	User role	ID
User	Organization	Asset Manager Admin	c8b054a9-1940-4f05-9059-f3859f1704b3
User	Project	Asset Manager Viewer	4101ecc8-2223-4066-aa4b-6b6f4d0d01e6
User	Project	Asset Manager Consumer	e5411482-bf96-4166-a1e5-07726894b8b5
User	Project	Asset Manager Contributor	c7fcf669-df1b-4a47-a662-4c15e0a29425
Service account	Organization	Asset Manager Search	009c9403-7516-40cb-8bc1-7f708ec307b7
Service account	Organization	Asset Manager Admin	d0d7803a-b0a1-4a71-9350-0712c3da7e59
Service account	Project	Asset Manager Viewer	b5b10e2a-5838-4235-ac3a-9be9fcec6ead
Service account	Project	Asset Manager Consumer	bc1a173e-9f7a-4001-959e-ffc12ff98a7b
Service account	Project	Asset Manager Contributor	5e91cc47-a12f-45ab-bbc5-3dfcdac56b3c

Next in the hierarchy, you might have designers, contributors and editors who can import new assets, edit metadata and publish updates, with admin approval if required. They'll be the ones maintaining the library day to day. Finally, you'll have your consumer roles: employees who can search for, preview and download approved assets but won't be able to modify or publish them.



Edit roles in project

AA Andréanne Abel

Admin

Analytics

Asset Manager

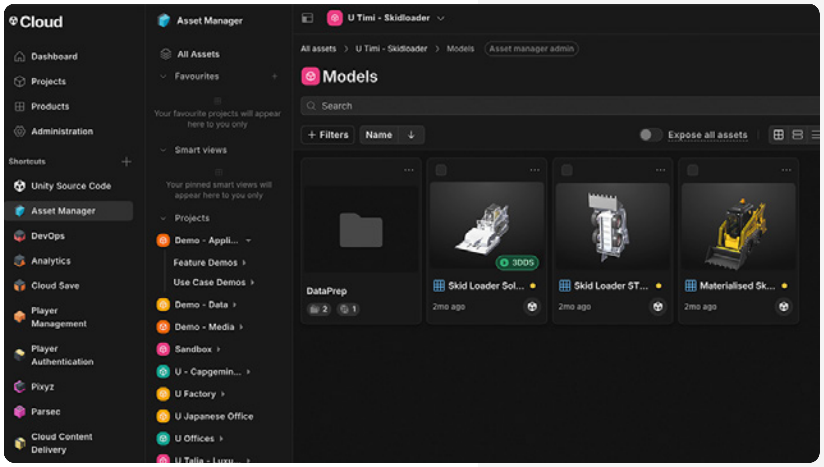
Select one

Asset Manager Consumer
Allows consuming Asset Manager resources

Asset Manager Contributor
Allows contributing to Asset Manager resources

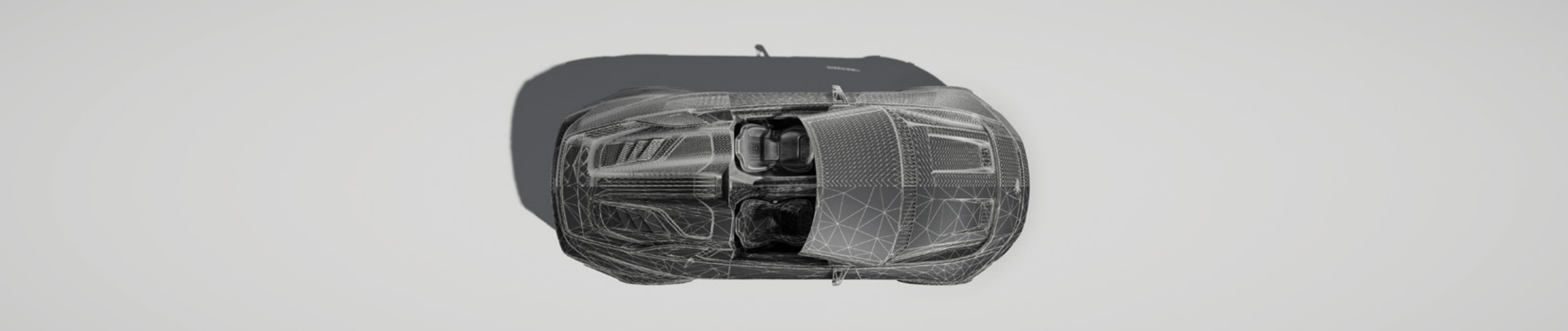
Asset Manager Viewer
Allows viewing Asset Manager resources

Role-based access controls are also critical for security. Many industrial 3D asset libraries contain highly sensitive information, some of which are subject to regulatory controls. For instance, export-controlled designs for government, aerospace or defense must conform with federal regulations that mandate a strict need-to-know posture, full auditability and deployment in tightly controlled environments. Linn recommended using a virtual private cloud deployment if you're operating in a highly regulated industry.



Regardless of your industry, there are some nonnegotiables:

- Always use RBAC with least-privilege defaults.
- Encrypt all data in transit and at rest in the platforms you use.
- Maintain comprehensive audit logs and approval statuses per version.
- Segment projects for sensitive work and, where needed, data residency options.



Versioning control and auditability

Governance isn't about just security and compliance but also standardizing asset management for ease of access and use. That said, most governance challenges stem from one of two extremes — everyone can change everything or no one can change anything. For instance, a trainer might overwrite a model the engineering team just approved. That's not necessarily due to carelessness, but more often a lack of clear versioning controls. "You'll often have multiple branching incarnations of the same asset, and it all needs to be managed cohesively and in a standardized manner," Linn said.

That's where versioning comes in. The goal is to ensure changes are clear, reversible and intentional. After all, any usable asset library is going to change regularly: A digital factory twin might pick up weekly layout tweaks, while a training program might be updated with new models ahead of a rollout of a next-gen XR headset.

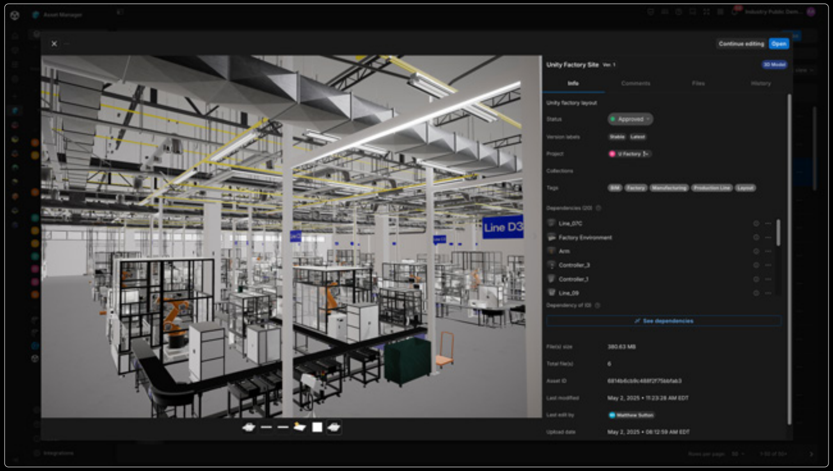
Treat your library as a single — albeit ever-changing — source of truth for real-time assets by applying a linear version history (v1.0, v1.1, etc.) and asset state (draft, in review, approved, retired, etc.).

In many industrial settings, engineering teams maintain source CAD files for official designs. However, visualization or training teams typically use real-time versions of those same models, optimized for their specific purposes. These assets should be linked rather than merged, otherwise it's impossible to tell which CAD version the visualization-ready model came from. For example, if engineering updates a drawing, a training model built months ago might still reflect the earlier version, and no one would know. That's a problem, because it breaks auditability and can lead to errors later on, such as training on the wrong configuration.

As for model variants, be sure to tag them according to their scope and purpose, instead of just making copies with new names. An automotive manufacturer might, for instance, tag them by geographical region, left- or right-hand drive, or form factor. That way, when a base part changes, you'll know exactly which variants need updating.

Similarly, if you have derivatives of certain base models, such as a simplified model used for training, label them as derivatives but maintain a reference to the original source and its CAD revision. This way, a trainer won't confuse a simplified mesh used for an XR/VR simulation with engineering's source of record.

Of course, such a granular level of versioning, while essential for usability, can be extremely difficult to achieve at scale, so automation is a necessity. Newer 3D asset managers typically include command line interfaces for performing bulk operations and event-based automation features for generating metadata, previews and tags for newly imported or updated assets.



Unity Factory Site Ver. 1

Unity factory layout

Status: Approved

Version labels: Stable, Latest

Project: U Factory

Collections

Tags: BIM, Factory, Manufacturing, Production Line, Layout

Dependencies (20)

Line_07C
Factory Environment
Arm
Controller_3
Controller_1
Line_09

Dependency of (0)

File(s) size: 380.63 MB

Total file(s): 6

Asset ID: 6814b6cb9c488f2f75bbfab3

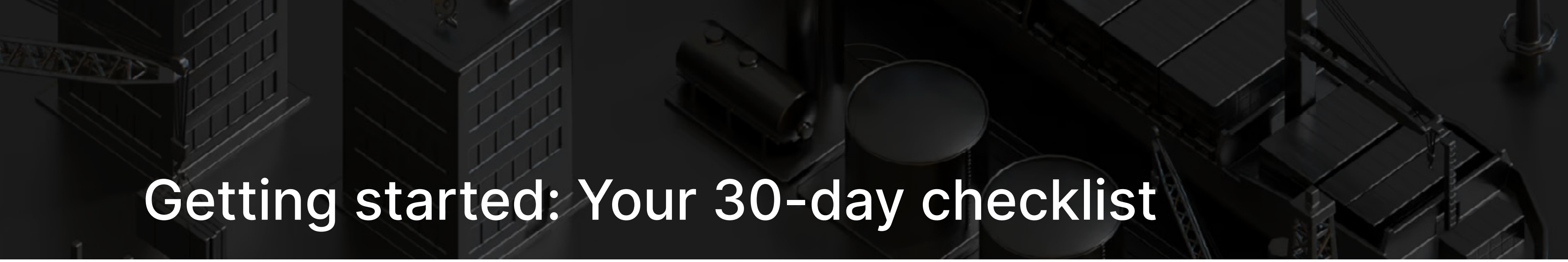
Last modified: May 2, 2025 • 11:23:28 AM EDT

Last edit by: Matthew Sutton

Upload date: May 2, 2025 • 08:12:59 AM EDT

Here's a quick checklist of operational changes you can make with the right asset manager:

- ✓ Adopt a three-role model, with admins, contributors and consumers. Add an approval role for projects that require additional sign-off.
- ✓ Map required metadata fields, such as asset identities, revision numbers, original sources, owners and approval states, and set approved-by-default views for most users.
- ✓ Link your CAD revisions to their real-time counterparts by labeling variants and derivatives clearly and automating updates where possible.



Getting started: Your 30-day checklist

Once your assets are imported, optimized and governed, the final step is putting your new system into daily use. Then you can quickly prove the value of a unified 3D asset library by creating the immersive experiences that drive real business results — whether it's for training, product development, customer experience or anything else.

Here's a quick recap of what you can do in just 30 days or less:

1. Inventory your data sources and their owners.
2. Select one or two representative models to pilot.
3. Determine which metadata fields to preserve.
4. Test the import to optimize your publishing cycle.
5. Set up your access controls and audit trails.

In our next playbook, we'll explore how to turn connected data into interactive 3D experiences — real-time experiences that enhance training programs, elevate product development, delight customers and more.

Are you ready to connect, create and deploy immersive experiences for your industry? Unity Industry provides a comprehensive suite of tools for training and guidance, 3D design collaboration, CX, XR, and HMI backed by three-year long-term support and a massive developer community.

Start your real-time 3D journey today!

GET STARTED →

Unity®

About Unity

Unity [NYSE: U] offers a suite of tools to develop, deploy, and grow games and interactive experiences across all major platforms from mobile, PC, and console, to extended reality. For more information, visit unity.com

LEARN MORE →