



FactVerse Omniverse Adaptor

使用指南

目录

1. 概述	3
1.1 核心功能	3
1.2 使用场景示例	4
1.3 插件加载方式说明	4
2. 系统环境要求	5
2.1 开发与依赖项	5
2.2 推荐硬件配置（单机部署）	5
3. 准备工作	5
3.1 获取测试平台及账号信息	5
3.2 下载插件工具包	6
3.3 搭建数字孪生场景	6
3.4 传 USD 模型资源	9
3.5 接入数据	11
3.6 安装 Omniverse Nucleus	15
3.6.1 安装步骤	15
3.6.2 配置 Nucleus 服务地址	17
4. 测试连接与数据驱动效果	19
4.1 导入插件	19
4.2 验证连接与数据驱动	21
5. 插件集成与定制开发	24
5.1 创建 USD Viewer Streaming 应用	25
5.2 集成 Adaptor 插件和 Messaging 插件	26
5.3 开发与调试	28
5.4 Web 前端部署	30

1. 概述

FactVerse Omniverse Adaptor 是一款连接 FactVerse 平台与 NVIDIA Omniverse 的扩展插件，旨在帮助用户将数字孪生场景快速导入 Omniverse 环境，并结合 DFS（Data Fusion Services）实现 USD 场景中的动态仿真与联动。

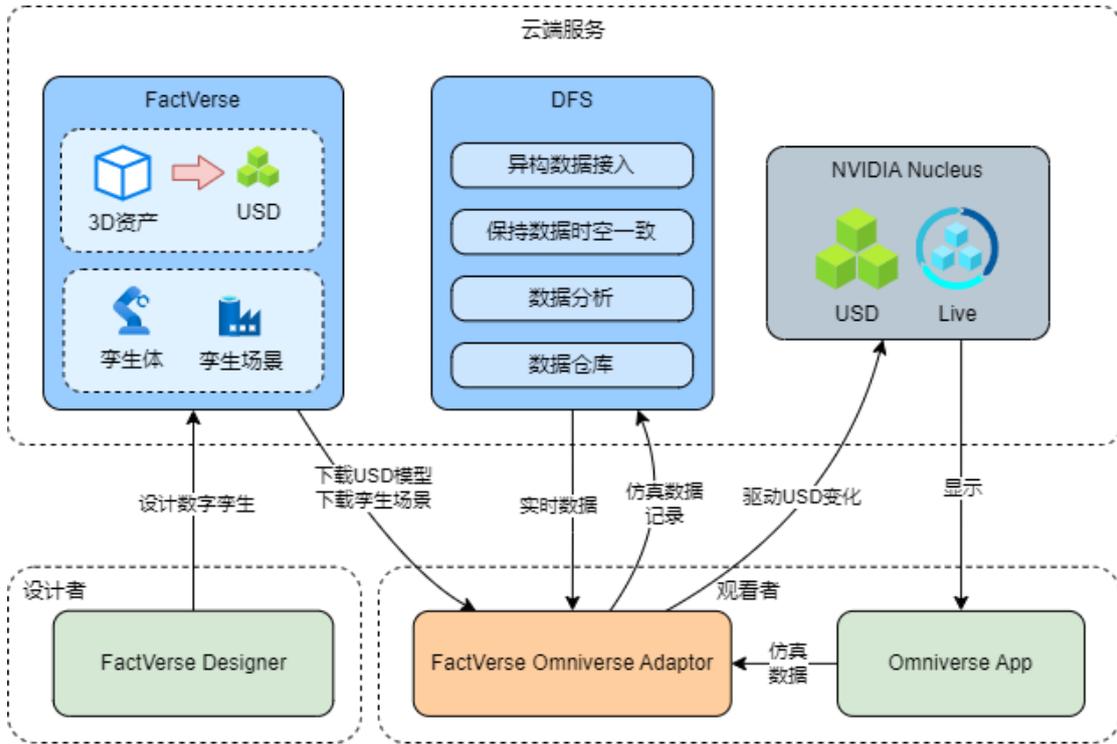


图 1 FactVerse × Omniverse 数据驱动协同架构图

1.1 核心功能

该插件在 Omniverse 应用中提供以下核心能力：

- **快速导入：** 从 FactVerse 平台加载 USD 模型和配置；
- **实时驱动：** 接入 DFS 数据，动态驱动场景变化；
- **多应用兼容：** 可集成至 USD Composer、Viewer、Explorer 等应用中使用；
- **灵活部署：** 支持本地运行或嵌入定制应用；
- **Web 可视化：** 搭配 WebRTC 与 Web 前端，实现浏览器内远程交互与查看。

1.2 使用场景示例

插件适用于以下典型场景：

使用场景	描述
场景导入与开发集成	将在 FactVerse 中构建的数字孪生场景快速导入 Omniverse 应用（如 USD Composer 或 Isaac Sim），支持在 Omniverse 内进行进一步开发。
实时数据可视化	接入 DFS 平台提供的实时或模拟数据，驱动场景中对象属性（如位置、角度、状态等）动态变化，实现数据与 3D 场景的联动展示。
仿真逻辑演示	插件内置行为树引擎可在本地运行场景中的逻辑（如设备状态变化、自动化流程等），并将结果同步渲染到 Omniverse 中，用于功能验证与场景仿真演示。
远程可视化展示	插件集成于 Streaming 应用中，结合 WebRTC 向浏览器端推送画面，实现 Web 远程交互。

1.3 插件加载方式说明

加载方式	适用场景	操作说明
Import Extension	<input checked="" type="checkbox"/> 测试连接与功能验证 适用于 USD Composer / Isaac Sim 环境下的手动导入	通过菜单 Window > Extensions > Options > Import Extension 导入 .zip 插件包，快速测试插件是否连接平台、数据是否驱动成功
Extension Search Path	<input checked="" type="checkbox"/> 代码开发与调试阶段 适用于本地自定义应用调试环境	在 USD Composer 或 Kit 应用中配置 Extension Search Path 指向 source/extensions/，便于热更新插件 Python 代码、查看实时日志输出
作为依赖配置	<input checked="" type="checkbox"/> 项目初始化阶段 适用于定制 App（如 USD Viewer Streaming App）	在 .kit 应用配置文件及插件 extension.toml 中添加依赖声明，构建时统一加载插件模块

2. 系统环境要求

在开发或测试环境中，建议将以下组件安装在同一台高性能工作站上，以保证通信链路稳定、渲染性能充足：

2.1 开发工具与依赖项

工具	说明
Git	用于下载 kit-app-template 模板和插件代码库
Omniverse Launcher	安装 USD Composer、Nucleus 等 Omniverse 核心服务
VS Code (推荐)	插件开发调试环境，Omniverse 提供官方 VS Code 插件支持 Kit 开发
Node.js + npm (可选)	构建 Web Client (远程交互与 WebRTC 可视化)

2.2 推荐硬件配置 (单机部署)

组件类型	推荐硬件配置
GPU	NVIDIA GeForce RTX 4090 D (VRAM 24G)
内存	32GB (2400 MHz)
CPU	英特尔 12th Gen Core i7-12700KF (12 核)
硬盘	SSD 512GB 以上

3. 准备工作

在正式接入 FactVerse Omniverse Adaptor 插件前，请完成以下准备流程，确保环境配置完整、资源可用、数据接通。

3.1 获取测试平台及账号信息

为了进行插件测试与场景搭建，需要使用 DataMesh 提供的 FactVerse 平台与 DFS 平台。

平台信息

平台名称	访问地址
FactVerse 平台	https://dtcs-local-demo.datamesh.com/login
DFS 平台	https://dfs-local-demo.datamesh.com/login

账号申请与资源获取

- 请发送邮件至 lab@datamesh.com 申请试用账号及测试素材，也可通过相关动手实验室下载测试资源。
- **注意：**上述平台仅限使用配套测试素材。如需使用**企业自有数据**进行测试，或需搭建**内部测试环境**（私有部署），请联系 lab@datamesh.com 申请相关支持方案，以确保数据安全合规。

3.2 下载插件工具包

请访问以下链接，下载并解压所需插件工具包：[FactVerse-Omniverse-Adaptor-v1.0.1.zip](#)

工具包内容：

文件名	简称	功能说明
datamesh.factverse_ovadaptor_extension-1.0.1.zip	Adaptor 插件	连接 FactVerse，驱动 USD 场景并写入至 Nucleus
datamesh.factverse_viewer_messaging_extension-0.1.1.zip	Messaging 插件	处理 Web 请求，控制场景加载与状态反馈
USDConverter_v0.1.0_20250526.zip	USD 转换器	将常见 3D 格式（如 FBX）转换为 USD 模型文件

3.3 搭建数字孪生场景

在搭建孪生体模板与数字孪生场景前，请安装以下必备设计工具：

- **DataMesh Importer:** 用于导入、预处理 3D 模型文件；
- **FactVerse Designer:** 用于模板创建、场景搭建和交互逻辑配置。

操作步骤

1. 准备模型：

- a) 在 DataMesh Importer 中打开模型。
- b) 根据实际运动需求，选择模型层级中的节点，以用于数据驱动。
- c) 上传模型至您的 FactVerse 资源库。

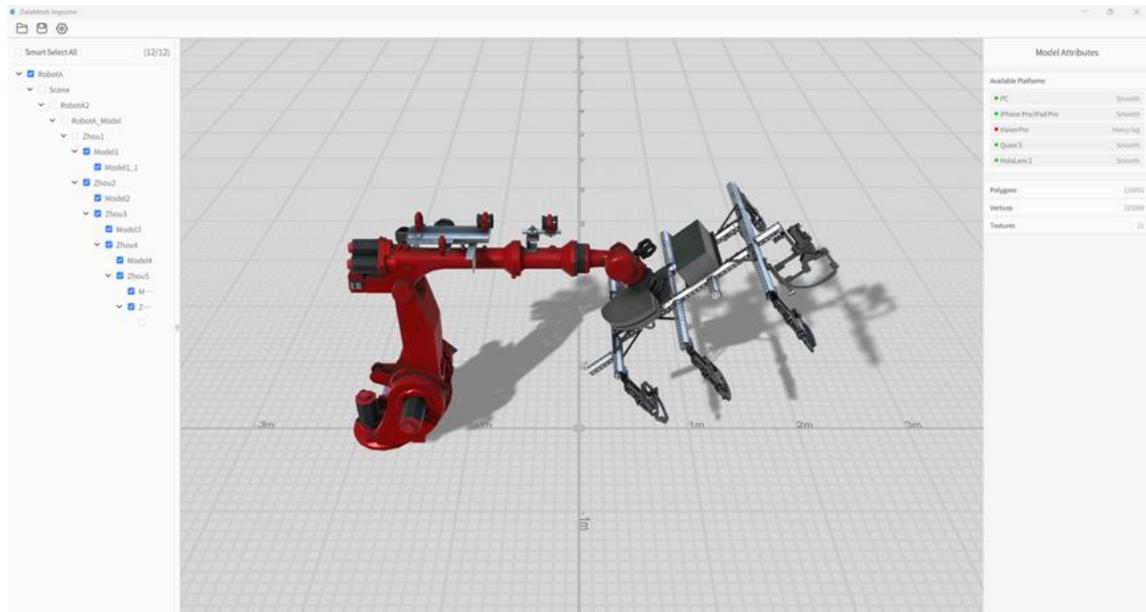


图 2 选择模型节点

2. 创建并配置模板：

- a) 在 FactVerse Designer 中新建模板。
- b) 拖入模型，添加属性。例如旋转角度。

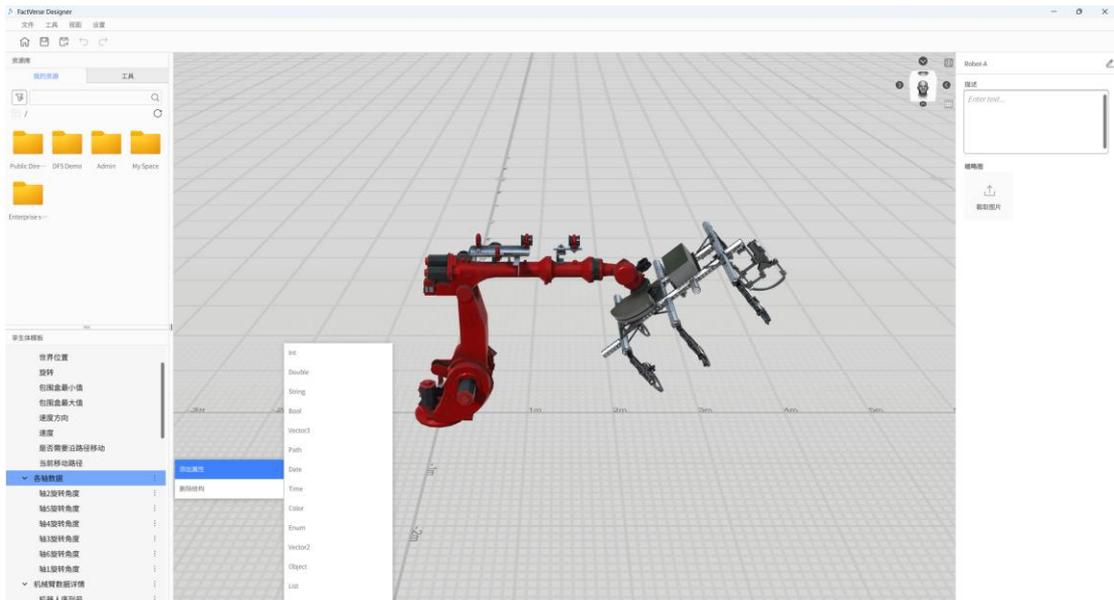


图 3 添加属性

c) 在数据绑定区域，将模型层级与属性进行关联，完成后保存模板。



图 4 绑定模型节点与属性

3. 搭建孪生场景：

- a) 新建孪生场景，根据已创建的模板批量生成孪生体。
- b) 孪生体即代表实际设备，注意明确命名，便于后续数据绑定。
- c) 调整孪生体位置和布局，完成基础场景构建并保存。

3.4 传 USD 模型资源

将 3D 模型转换为 .usdz 格式并上传至平台资源库。

操作步骤

1. **模型格式转换：**使用 DataMesh 提供的转换工具将常见格式模型（如 FBX、OBJ）转换为 .usdz 文件。

- a) 解压插件工具包中的 USDConverter_v0.1.0_20250526.zip
- b) 运行 USDConverter.exe

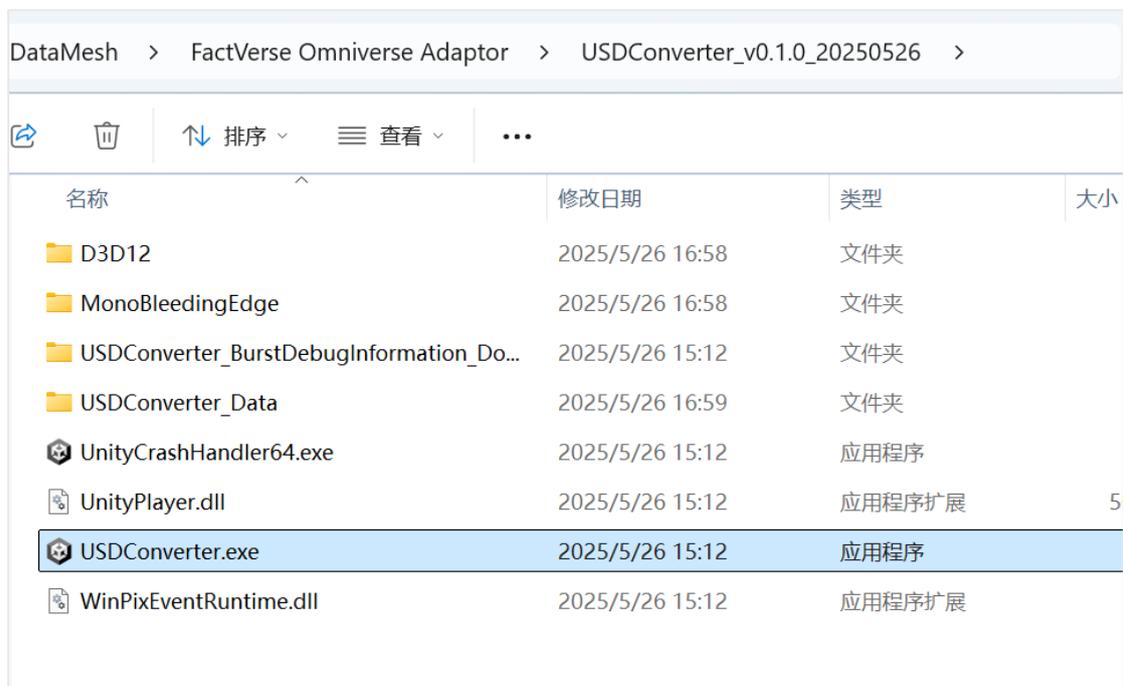


图 5 运行转换工具

- c) 打开原始模型

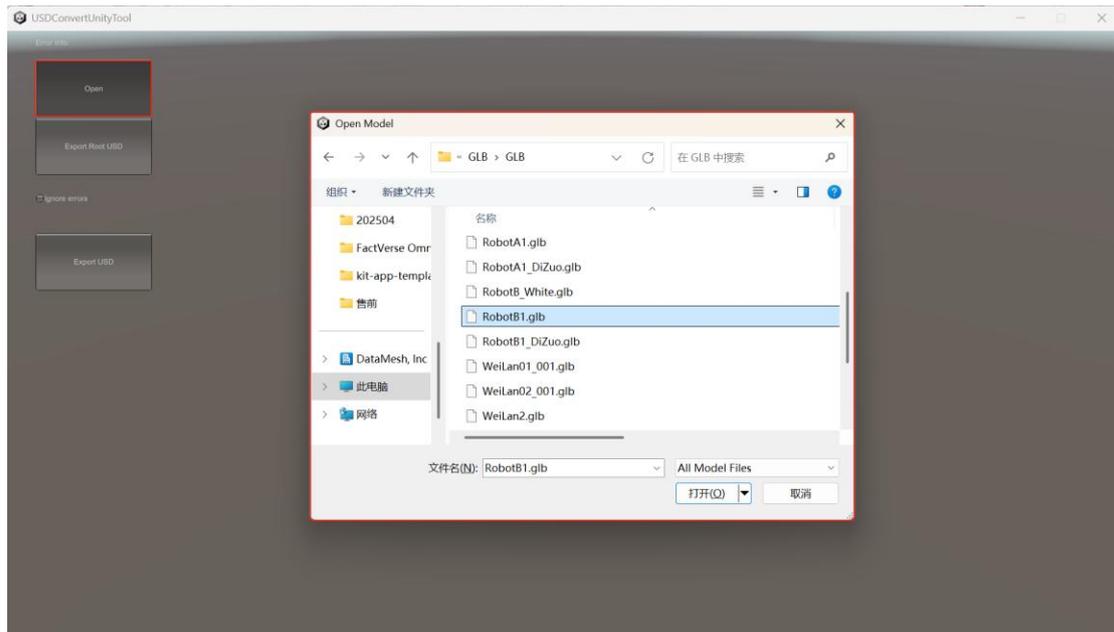


图 6 选择模型

d) 打开模型后，点击【Export USD】导出 .usdz 格式文件。



图 7 Export USD

2. 上传 .usdz 文件

- a) 登录 FactVerse 平台。
- b) 在数字资源的资源库中，选择对应模型，进入资源详情页面。

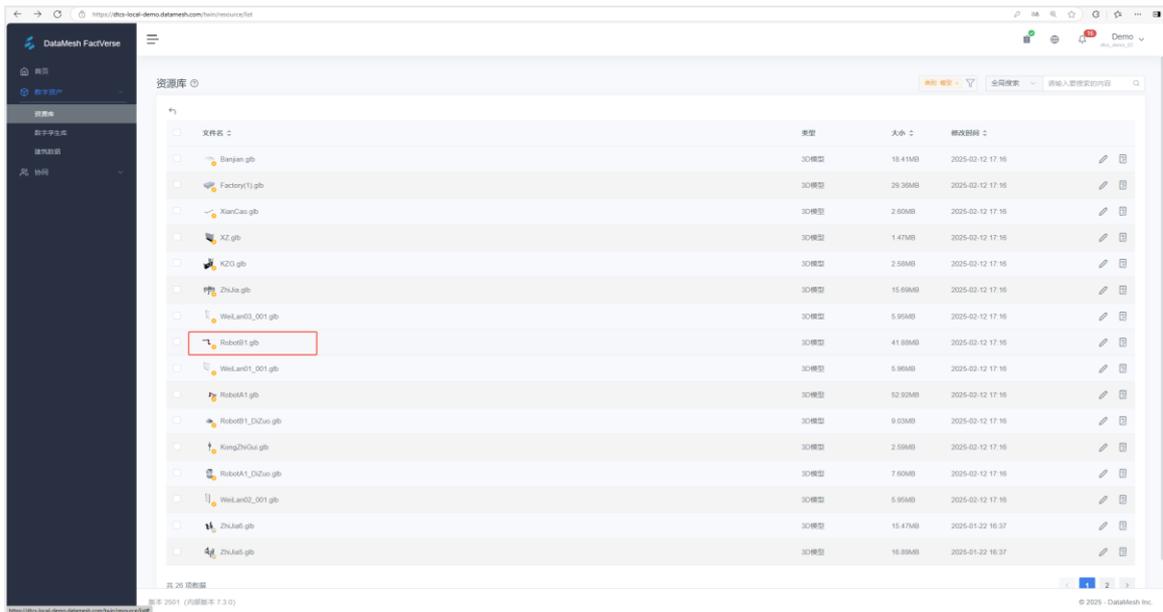


图 8 打开模型详情页面

c) 点击【Omniverse】上传对应的 .usdz 格式资源文件。

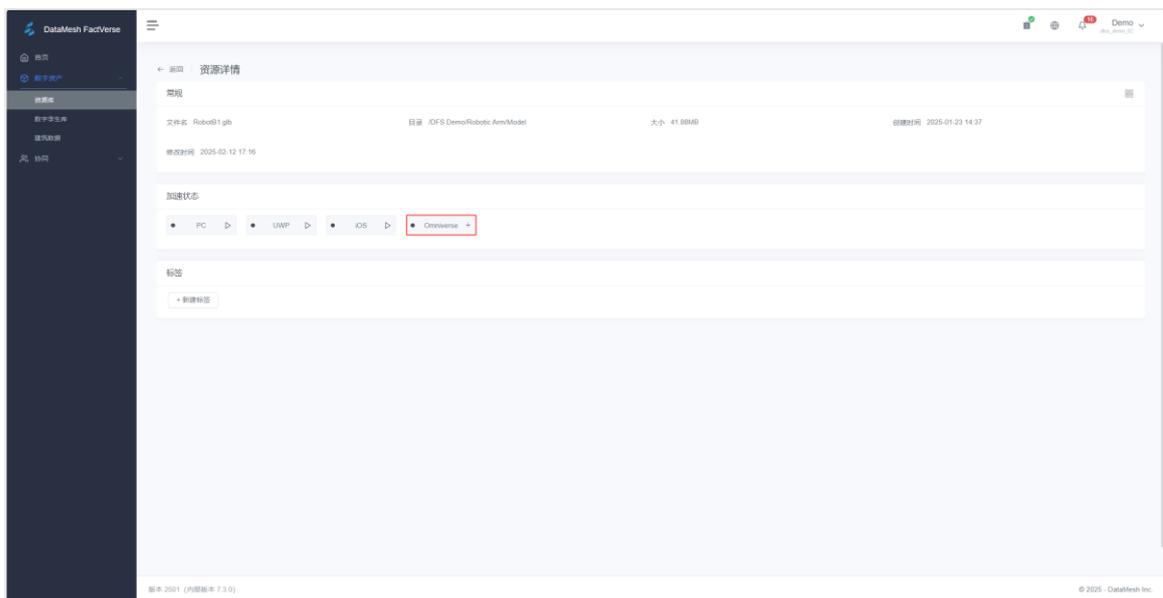


图 9 上传.usdz 格式文件

3.5 接入数据

通过 DFS（Data Fusion Service）平台接入设备数据，完成孪生场景数据驱动配置。

操作步骤

1. 创建模拟数据源：

- a) 登录 DFS 平台，进入【数据源管理】>【模拟数据源】，新建任务并上传数据。
- b) 为数据绑定对应的设备，并设置数据的循环频率。



图 10 创建模拟数据源

2. 导入孪生场景：在 DFS 平台进入【孪生体关联】>【场景配置】页面，点击【导入】，选择需要绑定数据的孪生场景。

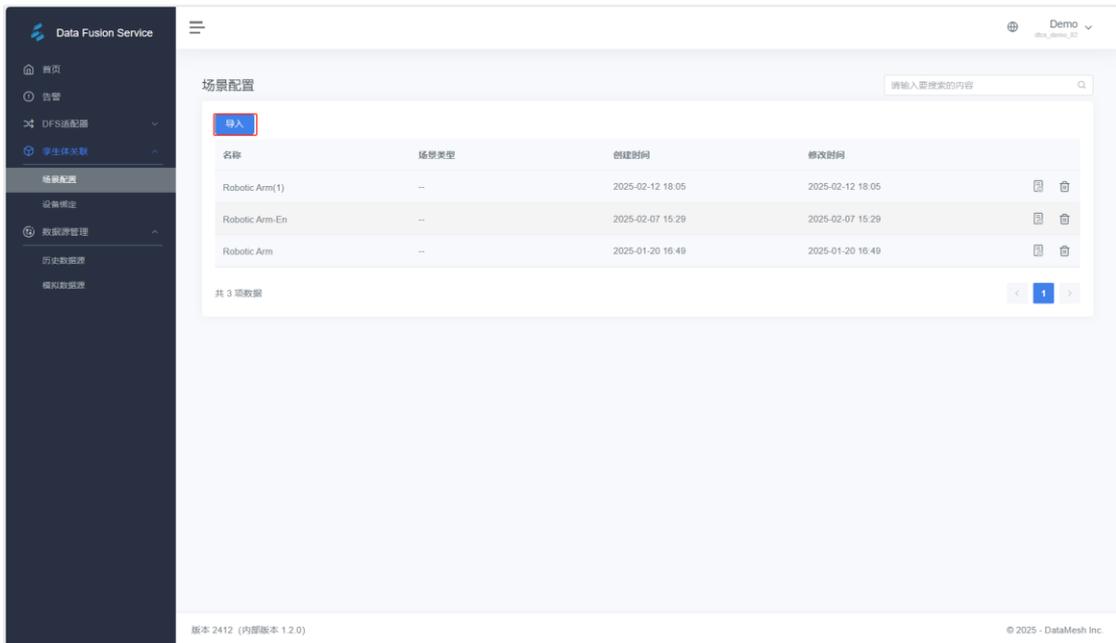


图 11 导入场景

导入完成后，可在场景详情页查看孪生体列表。

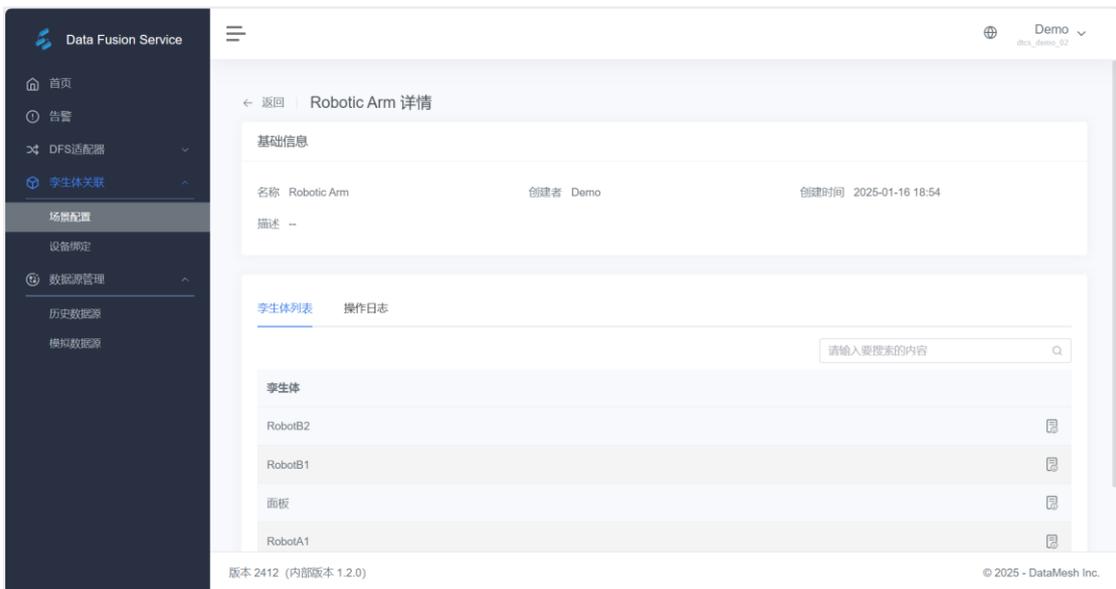


图 12 孪生体列表

3. 绑定数据到孪生体:

- a) 进入【孪生体关联】>【设备绑定】页面，找到要绑定的设备，例如设备 Robot，点击相应的详情按钮，打开设备的详情页。

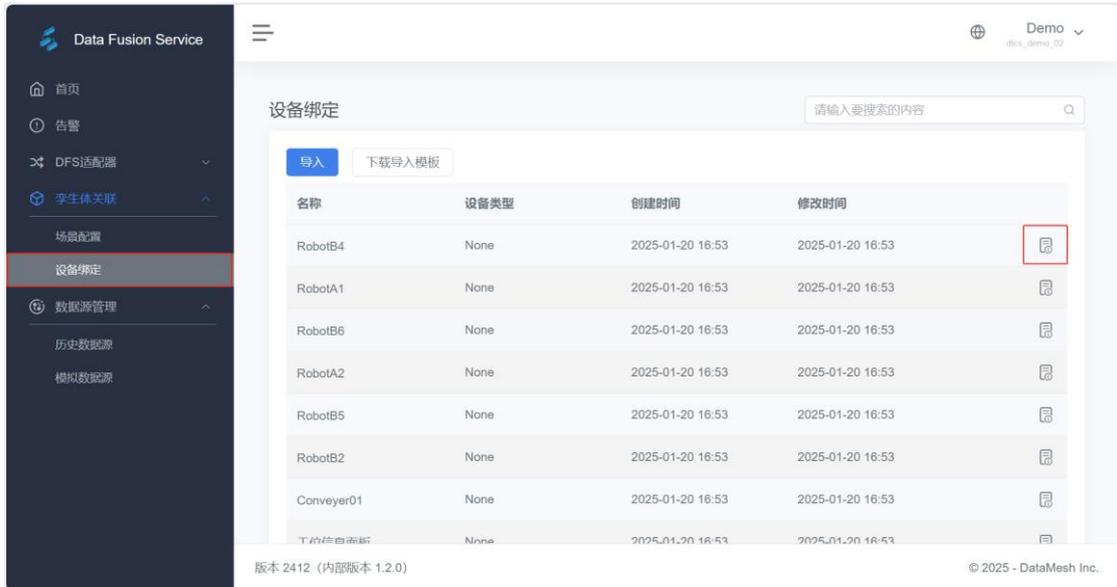


图 13 查看设备详情

- b) 在**设备配置**栏里，点击【**配置孪生体**】按钮，打开**配置孪生体**窗口，绑定对应孪生体。



图 14 配置孪生体

- c) 在**属性**栏里，点击设备属性的绑定图标，将设备属性与孪生体属性逐一绑定。
4. **场景验证**：返回 FactVerse Designer，打开已绑定数据的孪生场景，点击播放，观察孪生体的运动效果。

更多信息参见《DataMesh FactVerse DFS 用户指南》。

3.6 安装 Omniverse Nucleus

在正式使用 Adaptor 插件前，必须安装并配置 Omniverse Nucleus。该组件在插件运行中承担两个关键角色：

- **角色一：USD 场景资源服务器**
负责存储并同步插件生成的 USD 场景文件，供 Omniverse 应用加载和渲染。
- **角色二：提供本地 API 接口**
Adaptor 插件运行时会调用本地 Nucleus 接口以完成上传、资源查询等操作。

部署方式说明

部署方式	配置说明
插件与资源服务器部署在同一台机器上	只需安装一个 Nucleus 实例，该实例既作为资源服务器，也同时提供 API 接口。插件配置中的 Nucleus IP 可设为 localhost。
插件与资源服务器部署在不同机器上	需做两项配置： <ol style="list-style-type: none">1. 在插件所在机器上安装一个本地 Nucleus，仅用于提供 API 接口；2. 插件配置中的 Nucleus IP 必须填写远程 Nucleus 服务的地址（如 192.168.1.1），并确保远程服务器已通过 Add Server 添加该地址。

注意： 插件运行的机器必须能正常访问资源服务器，否则会导致场景加载和数据驱动失败。

3.6.1 安装步骤

1. 打开 Omniverse Launcher，进入 **Nucleus** 标签页。
2. 点击 **Add Local Nucleus Service**。

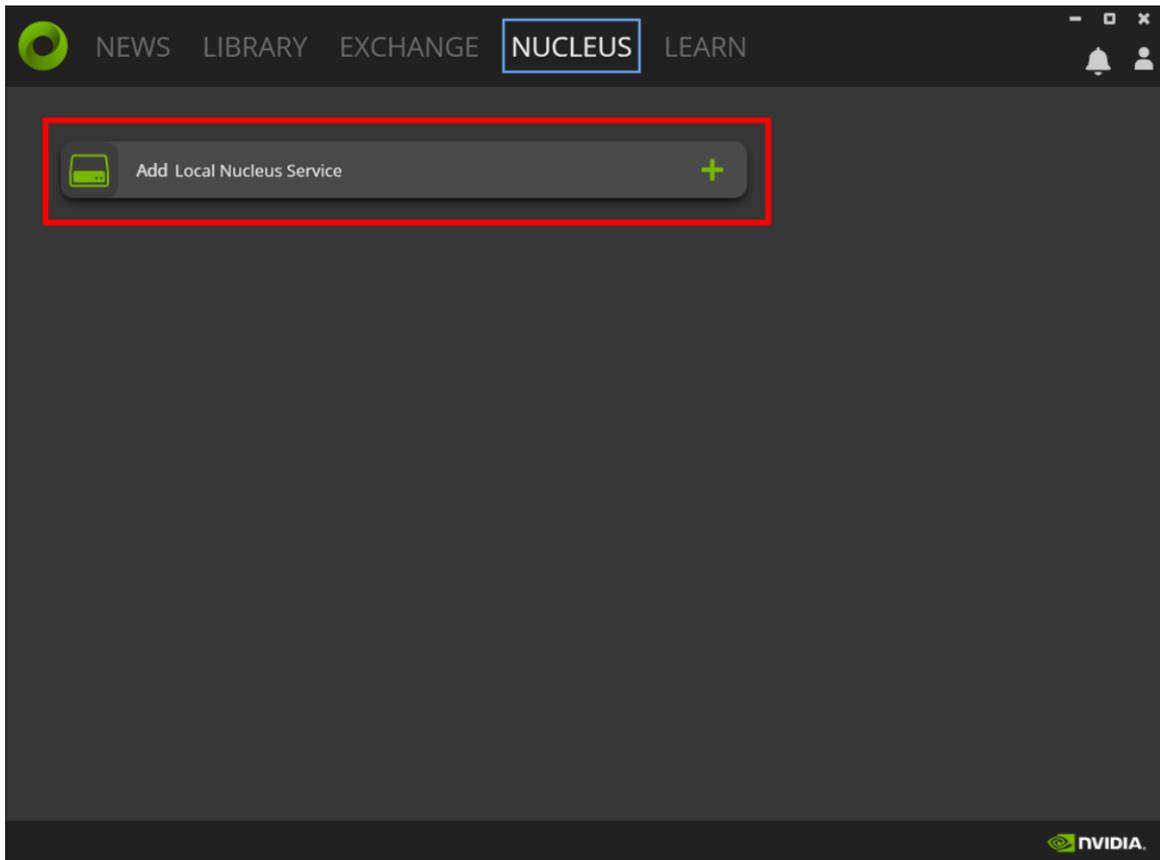


图 15 Add Local Nucleus Service

3. 设置管理员账号

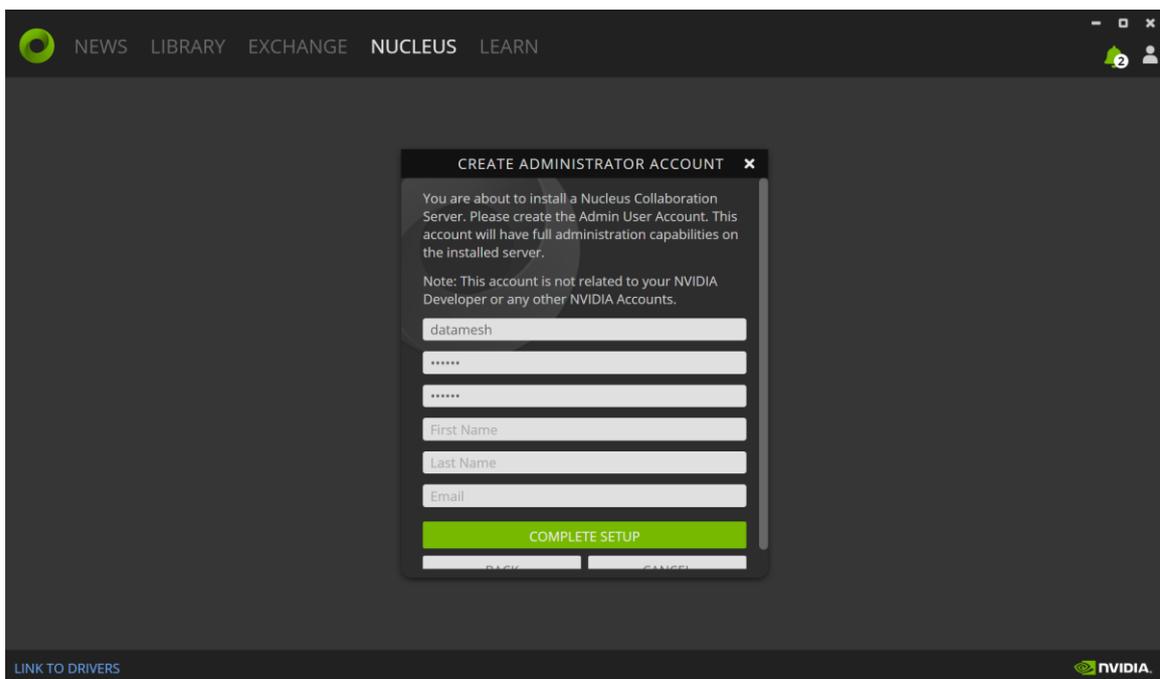


图 16 设置管理员账号

3.6.2 配置 Nucleus 服务地址

配置步骤

1. 访问 <http://localhost:3080/> 打开 Nucleus 服务管理页面。
2. 在 **Apps** 页面中，点击 **【Enable sharing】**。

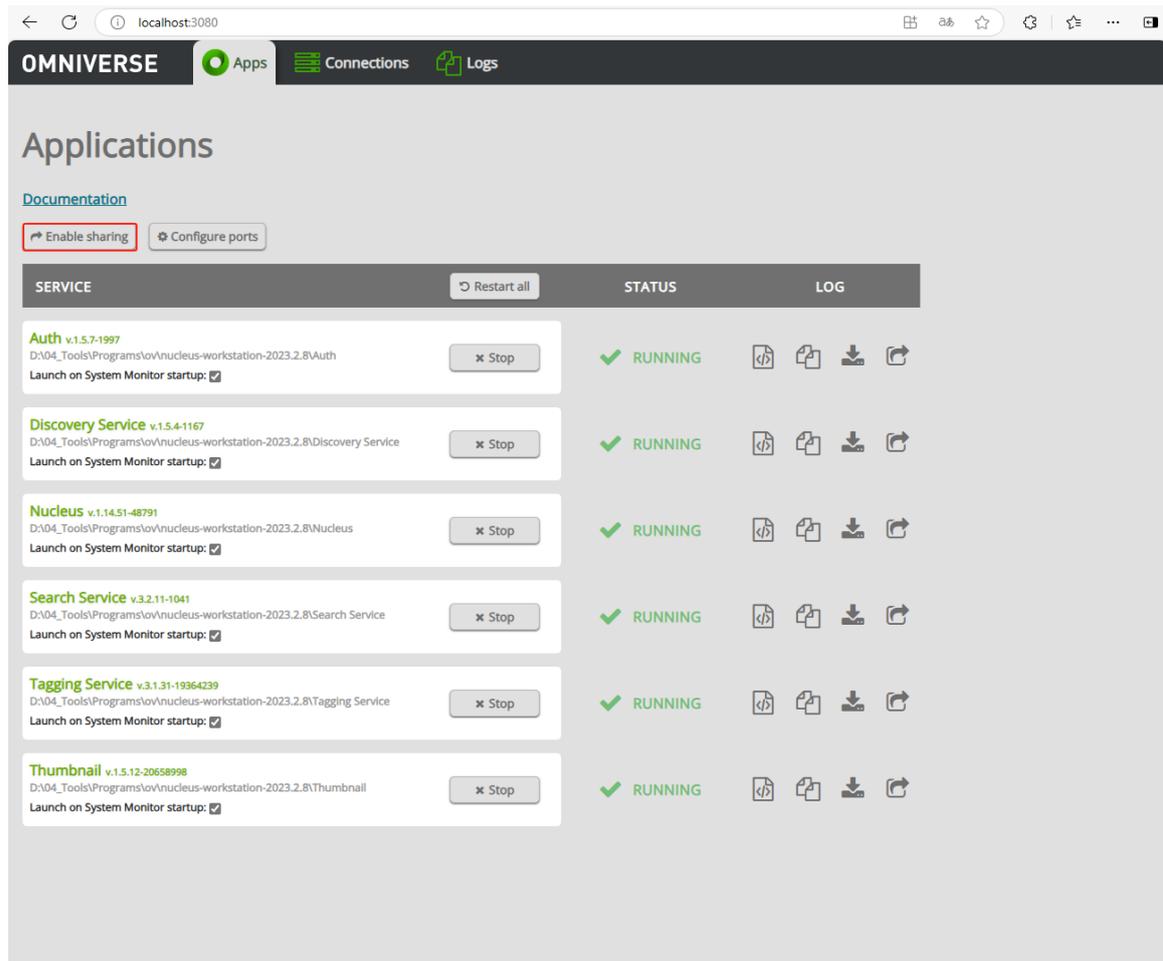


图 17 Enable Sharing

3. 启动 Nucleus 后，在服务器列表中通过 **【Add Server】** 添加 Nucleus 服务器地址，即本机地址。

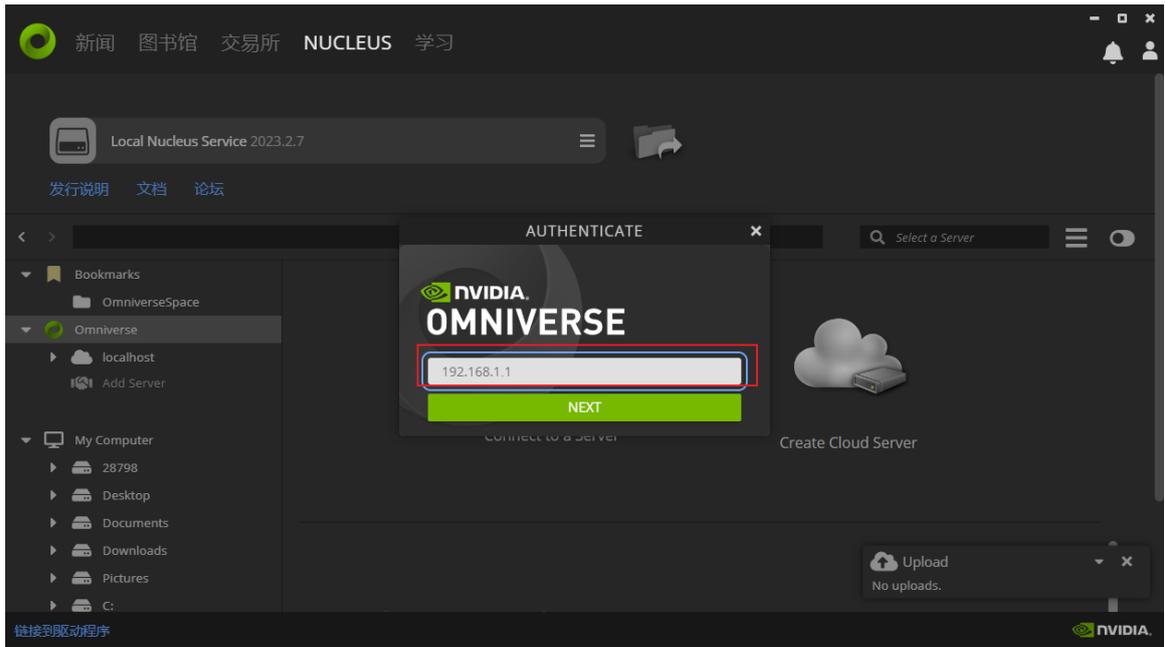


图 18 添加服务器 IP 地址

4. 完成配置后，在服务器列表中确认已成功添加。

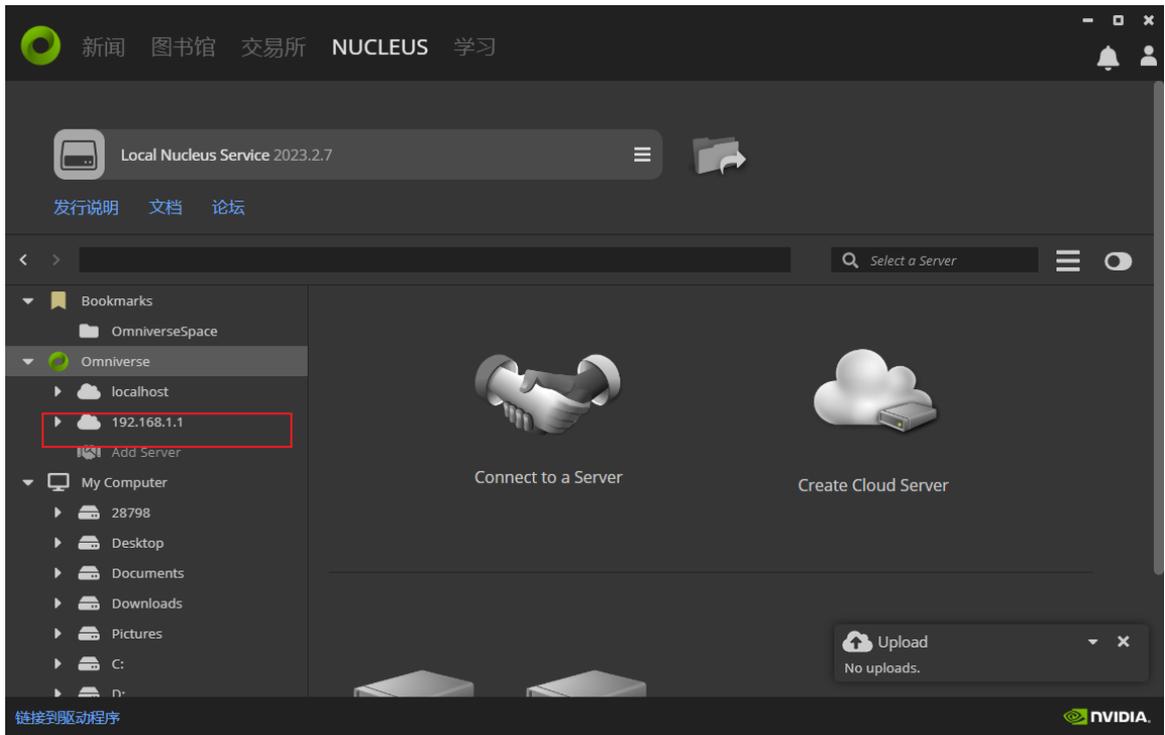


图 19 完成添加

4. 测试连接与数据驱动效果

您可通过 Omniverse 官方应用（如 USD Composer 或 Isaac Sim）导入 Adaptor 插件，用于验证与 FactVerse 平台的连接能力、DFS 数据驱动效果以及场景是否正确加载。

适用场景：

- 验证插件是否能成功连接 FactVerse 平台；
- 测试 DFS 数据驱动是否生效（如机械臂是否正常运动）；
- 为后续远程部署、自定义 App 开发提供可视化验证支持。

4.1 导入插件

以 USD Composer（**USD Composer 2023.2.3**）为例，导入插件步骤如下：

1. 安装 **USD Composer**。
2. 导入插件：
 - a) 打开 USD Composer，进入菜单 **Window > Extensions**。
 - b) 点击右上角 **Options > Import Extension**。

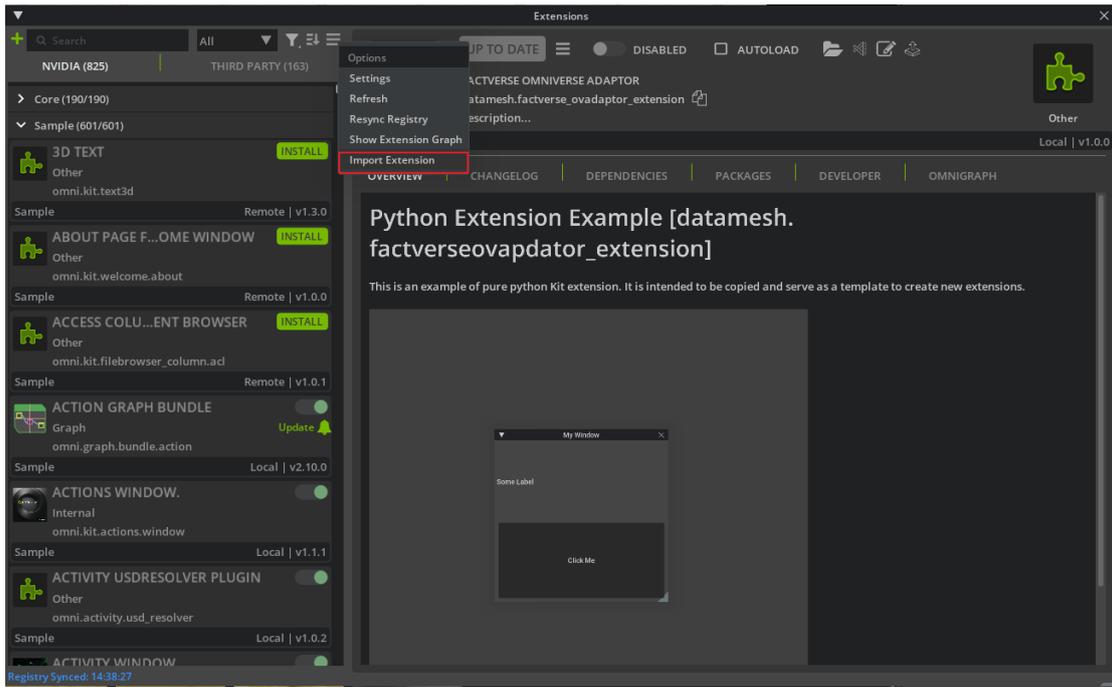


图 20 Import Extension

- c) 选择 `datamesh.factverse_ovadaptor_extension-1.0.1.zip` 文件，点击【Import】导入。

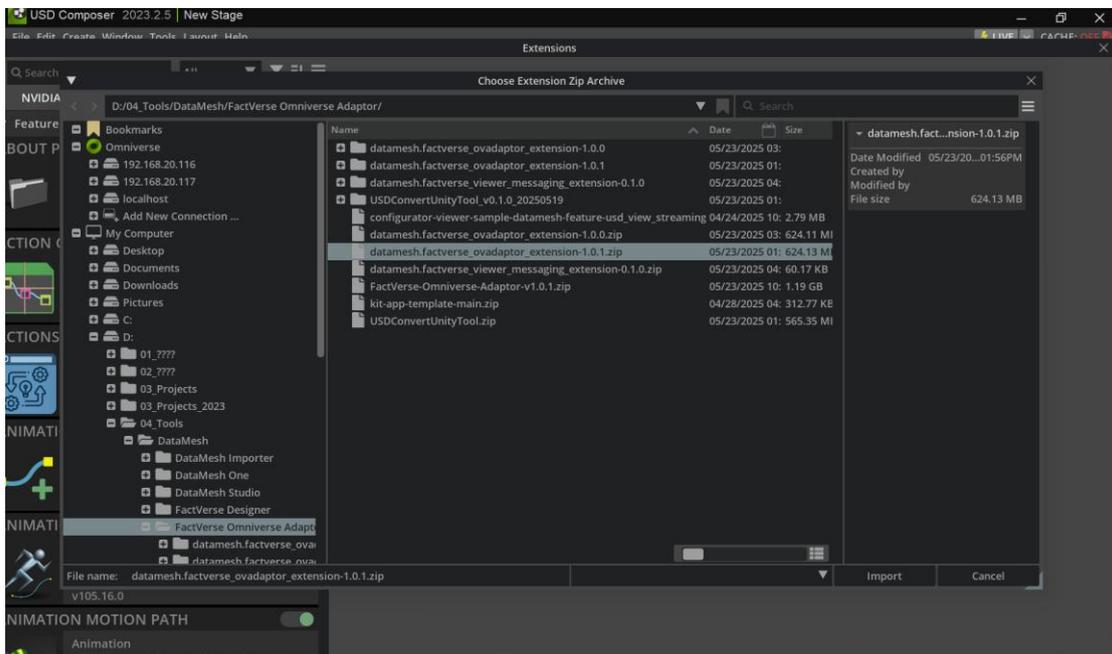


图 21 点击【Import】

- d) 在 Extension 搜索框中输入 DataMesh，找到“FACTVERSE OMNIVERSE ADAPTOR”，点击启用。

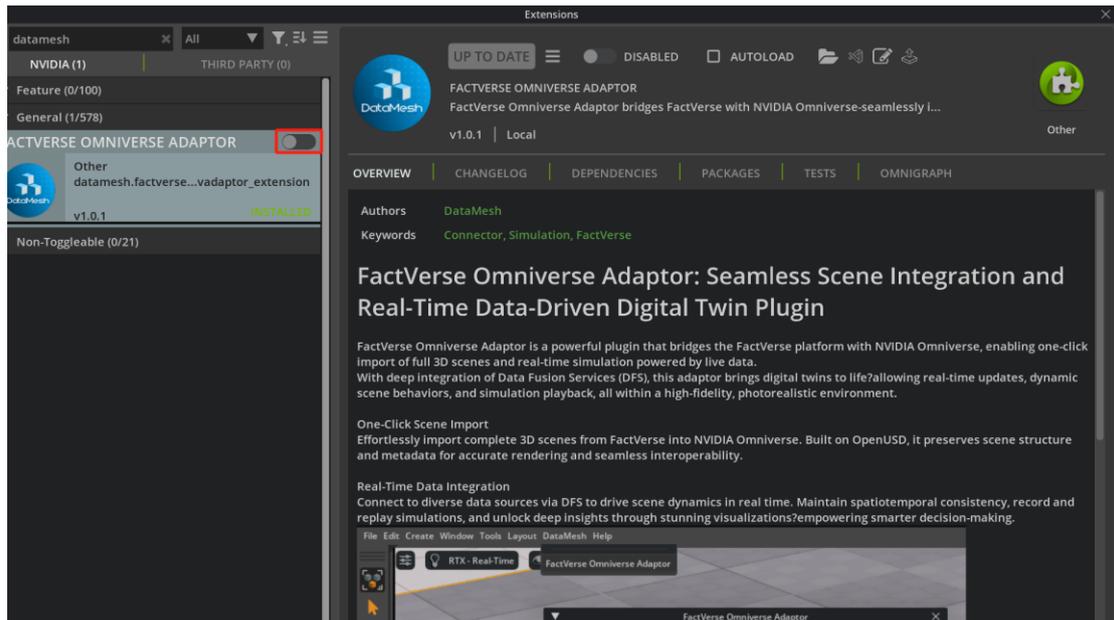


图 22 启用插件

4.2 验证连接与数据驱动

注意事项： 请确保在进行场景导入和数据集成时，处于编辑模式下。

1. 插件启用后，顶部菜单将新增 **【DataMesh】** 菜单项。点击 **【DataMesh】** > **【FactVerse Omniverse Adaptor】** 打开连接窗口。

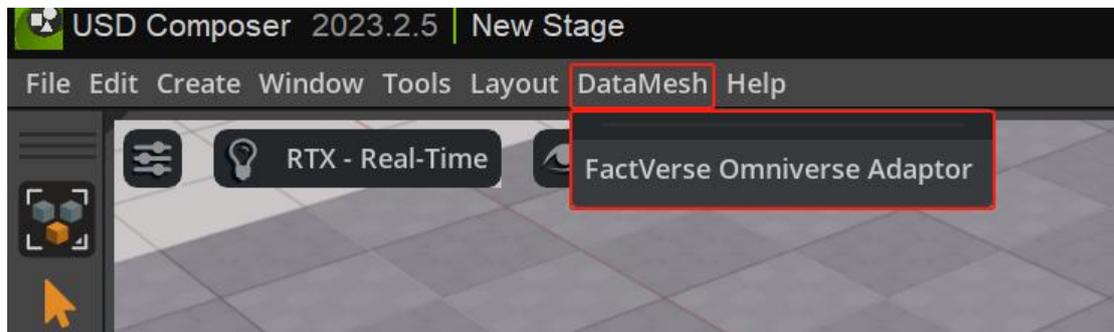


图 23 点击 **【DataMesh】** > **【FactVerse Omniverse Adaptor】**

2. 在连接窗口中填写以下信息：

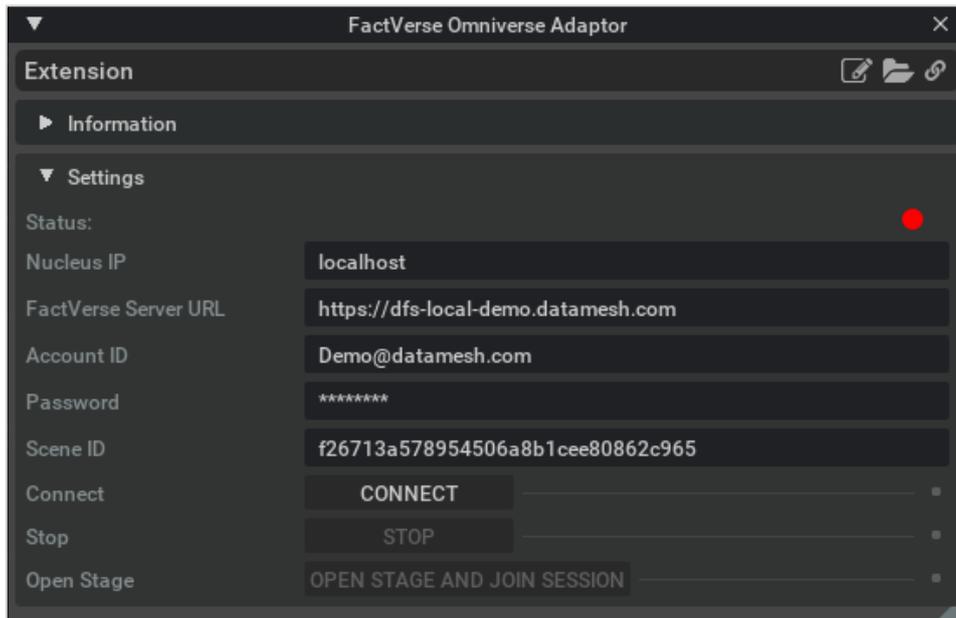


图 24 连接窗口

- **Nucleus IP:** 插件访问的资源服务器地址。若插件与插件访问的资源服务器在同一主机，可填写 localhost；若为远程部署服务，应填写其服务地址（如 192.168.1.1）
- **FactVerse Server URL:** FactVerse 平台访问地址
- **Account ID:** FactVerse 账号
- **Password:** 登录密码
- **Scene ID:** 孪生场景 ID

如何获取 Scene ID:

- a) 在 FactVerse 平台中打开目标孪生场景的详情页面。
- b) 复制 URL 中类似以下格式 ID:

<https://dtcs-local-demo.datamesh.com/.../6fa55ac9f44f4523ae0c166e299176d9/...6fa55ac9f44f4523ae0c166e299176d9> 即为 Scene ID。

3. 点击【CONNECT】按钮后，插件将自动加载场景并绑定数据。

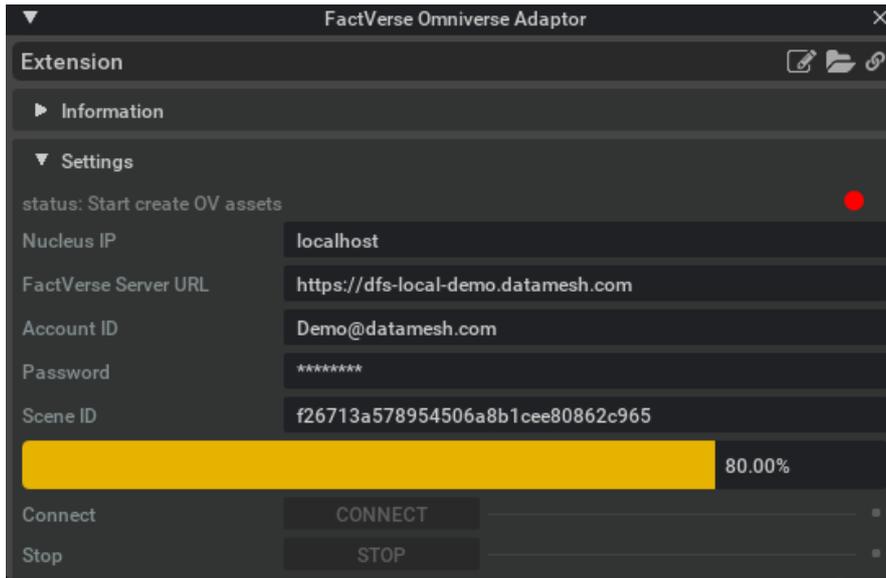


图 25 连接中

4. 完成连接后点击【OPEN STAGE AND JOIN SESSION】。

以上步骤完成后，即可在 USD Composer 中看到基于 FactVerse 实时数据驱动的孪生场景效果，为后续远程发布与调试提供验证基础。

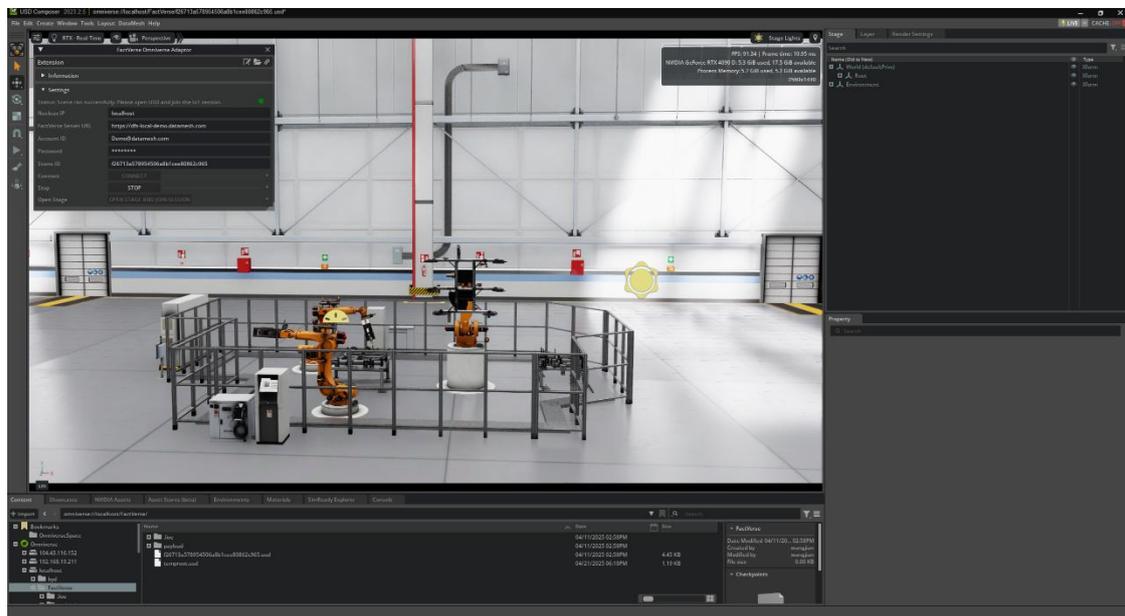


图 26 连接完成展示场景

5. 测试完成点击【STOP】按钮停止 Adaptor 运行。

5. 插件集成与定制开发

本章将介绍如何基于 **Omniverse Kit** 框架开发一个定制化的 **USD Viewer Streaming** 应用，并在其中集成 **Adaptor** 插件。通过该集成，用户可以在应用内实现以下关键能力：

- 从 **FactVerse** 平台加载孪生场景；
- 绑定 **DFS**（实时或模拟）数据，驱动场景动态变化；
- 在 **USD Viewer Streaming** 应用中完成场景渲染；
- 通过 **Web 浏览器** 实现远程访问与交互控制。

该方案适用于企业级部署、远程可视化展示、项目交付与多用户交互等场景，是插件从本地调试走向正式生产环境的主要集成方式。

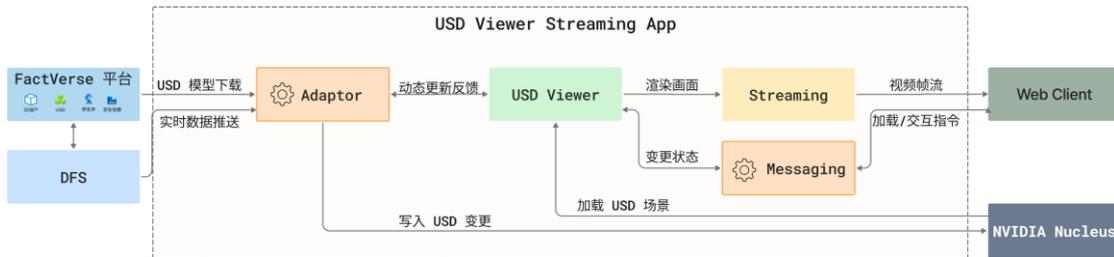


图 27 USD Viewer Streaming 应用中插件集成与数据流动图

数据流说明：

1. **Adaptor** 插件从 FactVerse 平台下载 USD 模型和孪生场景配置，并负责向本地 Nucleus 写入场景内容与数据变更；
2. **DFS** 持续推送设备或传感器数据至 Adaptor 插件；
3. **Adaptor** 插件将数据按映射关系写入场景中的 USD 属性（如位置、角度、颜色等），实现对场景的动态驱动；
4. **USD Viewer** 从 Nucleus 加载并渲染当前场景，反映 Adaptor 写入的实时变化
5. **Messaging** 插件响应 Web 请求，通过调用 USD Viewer 接口加载目标场景，并监听场景中的事件与状态变化；

6. **Web Client** 发起加载/控制指令至 Messaging 插件，获取反馈信息，实现远程场景控制与状态同步。
7. **Streaming 模块** 捕获 USD Viewer 的渲染画面，将其编码为视频流，通过 WebRTC 推送至 Web Client，实现浏览器中的 3D 场景可视化。

5.1 创建 USD Viewer Streaming 应用

1. 初始化 App 项目：

```
git clone https://github.com/NVIDIA-Omniverse/kit-app-template

cd kit-app-template

.\repo.bat template new
```

2. 按提示填写：

- 类型：Application
- 基于模板：[omni_usd_viewer]
- 应用名：datamesh.my_usd_viewer
- 版本号：0.1.0

3. 配置扩展插件信息：

在初始化流程中，需要配置以下两个核心扩展插件：

a) USD Viewer Messaging Extension

此插件负责 Web 客户端与 Viewer 应用之间的指令通信和状态反馈。

项目	配置示例值
文件名	datamesh.factverse_viewer_messaging_extenslon
名称	My USD Viewer Messaging Extension
版本号	0.1.1

b) USD Viewer Setup Extension

此插件负责处理应用内的配置、通信、布局和其他设置等。

项目	配置示例值
文件名	datamesh.my_usd_viewer.setup

名称	My USD Viewer Setup Extension
版本号	0.1.0

4. 启用流式传输

在初始化 USD Viewer Streaming 应用时，请在 Streaming 配置步骤中，选择以下模块：

- **[omni_default_streaming]: Omniverse Kit App Streaming (Default)**

该模块将自动为应用集成 WebRTC 流式传输功能，支持将 USD Viewer 的渲染画面实时推送至浏览器前端，从而实现远程可视化与交互控制能力。

5. 构建项目

```
.\repo.bat build
```

5.2 集成 Adaptor 插件和 Messaging 插件

1. 解压插件至 **extensions** 目录：将以下两个 zip 文件解压至 `source/extensions/` 目录中。

- `datamesh.factverse_ovadaptor_extension-1.0.1.zip`
- `datamesh.factverse_viewer_messaging_extenslon-0.1.1.zip`

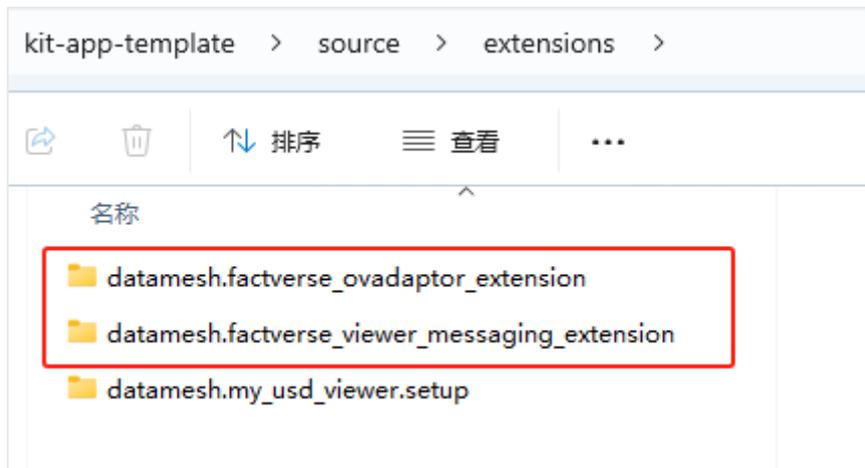


图 28 插件目录结构示意图

2. 配置依赖：

- a) 在 `datamesh.usd_viewer.setup/config/extension.toml` 中增加 Messaging 插件依赖。

```
[dependencies]
"omni.kit.usd.layers" = {}
"omni.activity.ui" = {order=1000} # Progress activity messages
"omni.kit.quicklayout" = {}
"omni.kit.viewport.utility" = {}
"datamesh.factverse_viewer_messaging_extension" = {} # Required messaging extension
```

- b) 编辑 `datamesh.factverse_viewer_messaging_extension/config/extension.toml` 中增加 Adaptor 插件依赖。

```
[dependencies]
"datamesh.factverse_ovadaptor_extension" = {} # Required Adaptor extension
```

- c) 在 `source/apps/datamesh.my_usd_viewer_streaming.kit` 中添加 Live Session 功能的依赖

```
[dependencies]
...
"omni.kit.collaboration.channel_manager" = {}
"omni.kit.collaboration.presence_layer" = {}
"omni.kit.collaboration.selection_outline" = {}
"omni.kit.collaboration.telemetry" = {}
"omni.kit.widget.live" = {}
"omni.kit.livestream.webrtc" = {} # Livestream extension
```

3. 配置连接参数：编辑 `datamesh.factverse_ovadaptor_extension\config\extension.toml`

连接参数说明

参数项	说明
id	FactVerse 平台账号（如：admin@datamesh.com）
pass	登录密码
sceneId	要连接的孪生场景 ID，可在平台 URL 中获取（详见第 4.2 节）

ovserver	<p>Nucleus 所在服务器的 IP 地址：</p> <ul style="list-style-type: none"> • 若插件与 Nucleus 资源服务部署在同一台机器，可使用 localhost； • 若资源存储在其他机器，必须填写其实际 IP 且保证可访问； • 在 Nucleus 资源服务器中必须通过【Add Server】方式添加此 IP 地址（详见 3.6.2）。
DCSDomain	DFS 平台地址，如： https://dfs-local-demo.datamesh.com
ui_visible	<p>是否显示插件 UI</p> <ul style="list-style-type: none"> • true 表示插件界面可视； • false 表示插件以隐藏方式运行，适用于定制应用中自动加载的场景

4. **建立硬链接（可选）：** 为避免 Windows 对路径长度的限制，可使用硬链接方式缩短路径。

```

mklink /J "路径/_build/.../exts/datamesh.factverse_ovadaptor_extension" "路径
/source/extensions/datamesh.factverse_ovadaptor_extension"
mklink /J "路径/_build/.../exts/datamesh.factverse_viewer_messaging_extension" "路径
/source/extensions/datamesh.factverse_viewer_messaging_extension"

```

5.3 开发与调试

1. 在 USD Composer 中添加 Extension Search Path 挂载 Adaptor 插件源码，例如：
d:/temp/kit-app-template/source/extensions。

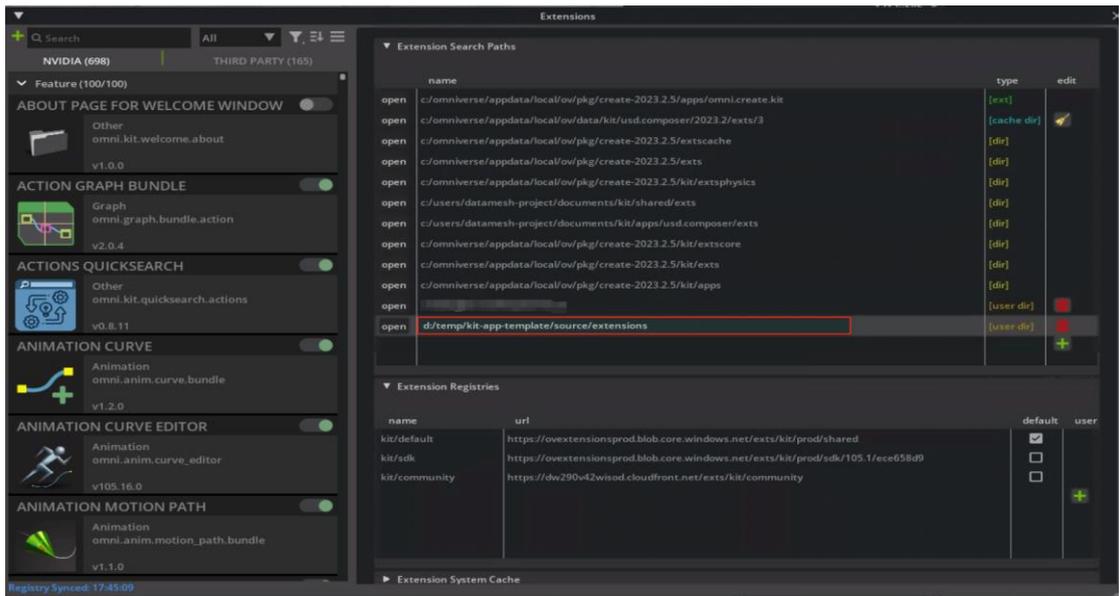


图 29 导入插件

2. 启用插件

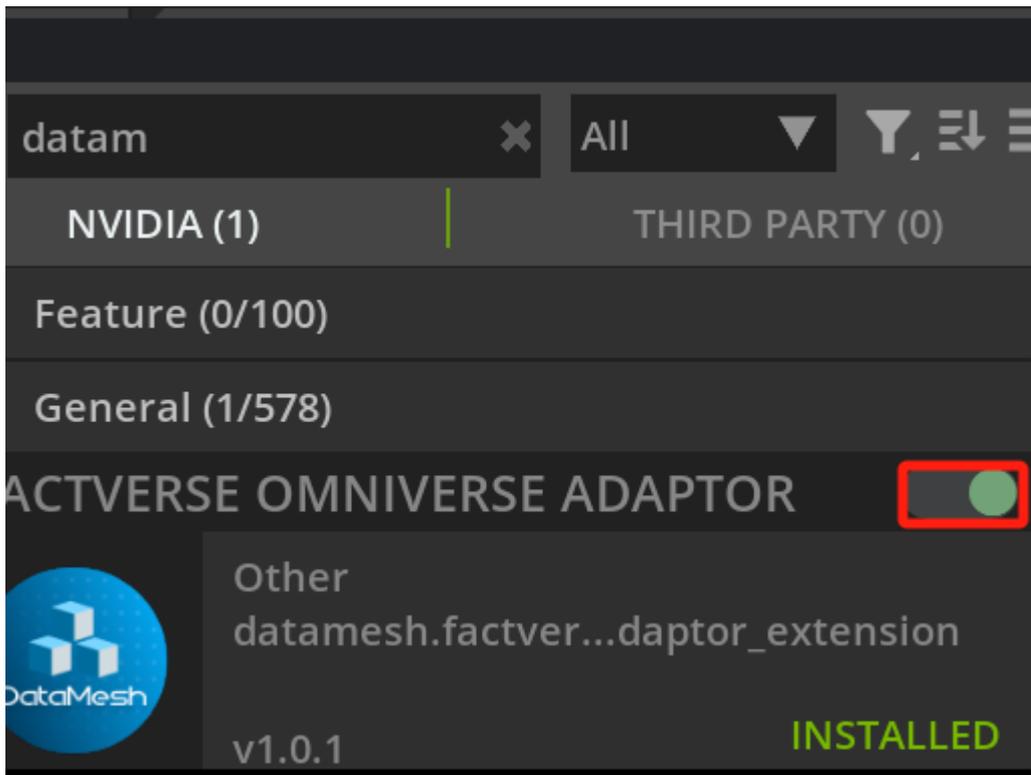


图 30 启用插件

3. 使用 VSCode 打开插件源码，进入调试开发。

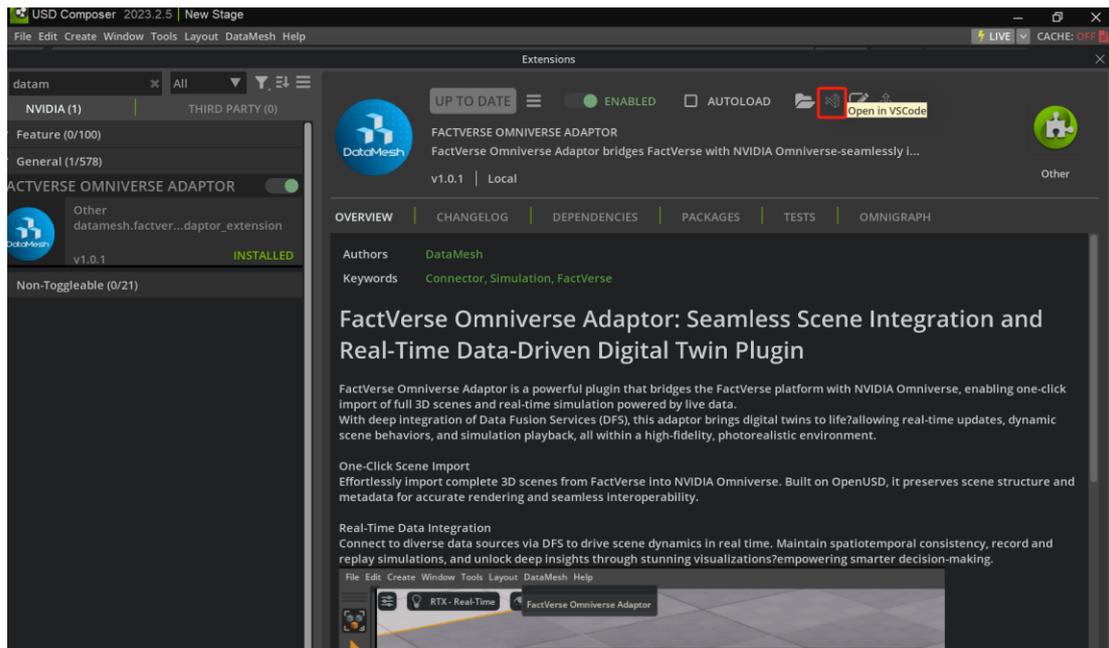


图 31 Open in VSCode

4. 以管理员权限启动 USD Viewer Streaming 应用

```
.\repo.bat launch datamesh.my_usd_viewer_streaming.kit
```

确认开发调试无误后，可执行以下命令打包发布应用：

```
.\repo.bat package
```

5.4 Web 前端部署

在完成插件集成并启动 USD Viewer 应用后，若希望通过浏览器远程访问场景并进行交互操作，可部署配套的 Web 前端项目。该前端基于现代框架（如 Vite + React），可通过 WebRTC 实时显示 USD 渲染画面，并与插件通信实现数据控制。

参考示例：[Omniverse Configurator Sample Application](#)（由 DataMesh 提供）。

1. 修改场景 URL：

打开前端项目目录中的：

```
..\configurator-viewer-sample\src\Window.tsx
```

在 `_onClickOpenNucleusUSD()` 和 `_onClickJoinSession()`方法中，将 URL 替换为目标场景的实际地址。

```
/**
 * @function _onClickOpenNucleusUSD
 * =====2.打开Nucleus 某个usd =====
 * Send a request to load an asset based on the currently selected asset
 */
private _onClickOpenNucleusUSD(): void {
  //url:"./samples/stage01.usd"
  //var url = "./samples/stage01.usd"
  var url = "omniverse://localhost/FactVerse/f26713a578954506a8b1cee80862c965.usd"
  console.log("Sending request to open asset: ${url}.");

  const message: AppStreamMessageType = {
    event_type: "openStageRequest",
    payload: {
      url: url
    }
  };
  AppStream.sendMessage(JSON.stringify(message));
}

/**
 * @function _onClickJoinSession
 * =====2.加入Nucleus 某个Session =====
 */
private _onClickJoinSession (): void {

  var url = "omniverse://localhost/FactVerse/f26713a578954506a8b1cee80862c965.usd"
  var sessionName = "iot_session"

  const reset_message: AppStreamMessageType = {
    event_type: "JoinSessionRequest"
  };
}
```

图 32 修改场景 URL

2. 启动前端应用

```
npm install
npm run dev
```

默认访问地址为：<http://localhost:5173>，在浏览器中打开即可进入 Web 操作界面。

3. 启动 Adaptor 插件

点击界面上的 **【Start Adaptor】** 按钮，启动后台数据驱动插件。

4. 查询插件状态

点击 **【Get Adaptor Status】** 按钮，检查 Adaptor 是否已成功启动。

5. 加载 USD 场景

当确认 Adaptor 运行正常后，点击 **【Open USD】** 和 **【Join Session】** 按钮加载场景，USD Viewer 将从 Nucleus 读取场景并渲染，数据驱动自动生效。

加载完成后，页面将显示 USD Viewer 渲染画面，并支持与场景进行交互操作。

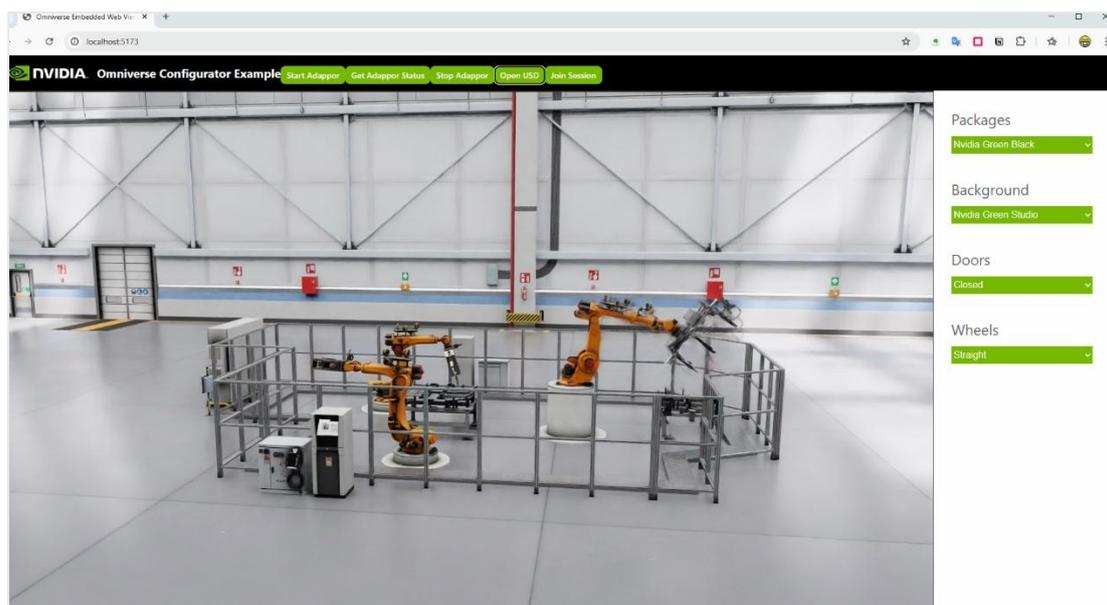


图 33 web 端展示效果