

# 数字孪生中三维仿真引擎 解耦方法设计与应用

## Design and Application of Decoupling Methods for Three-Dimensional Simulation Engines in Digital Twins

崔秋然,由志远,张 勇,田阿康,刘江锋(中讯邮电咨询设计院有限公司,北京 100048)

Cui Qiuran, You Zhiyuan, Zhang Yong, Tian Akang, Liu Jiangfeng (China Information Technology Designing & Consulting Institute Co., Ltd., Beijing 100048, China)

### 摘 要:

在三维场景仿真领域,不同仿真引擎各有利弊,对模型导入的要求也有所不同,为了使多种仿真引擎协同工作,提高协作效率,提出了一种引擎解耦方法。首先对现有三维场景仿真引擎进行介绍,分析其现状和局限性,突出引擎解耦的必要性。然后详细阐述了引擎解耦方法的实现步骤和技术细节,并将该方法应用到实际的项目中,通过实践验证该方法的有效性和实用性,提高了项目开发效率,降低了项目成本,为三维场景仿真提供了极大的便利。

### 关键词:

数字孪生;工业互联网;三维仿真;引擎解耦

doi: 10.12045/j.issn.1007-3043.2025.10.011

文章编号: 1007-3043(2025)10-0059-05

中图分类号: TN919

文献标识码: A

开放科学(资源服务)标识码(OSID):



### Abstract:

In the field of 3D scene simulation, different simulation engines have their own advantages and limitations, and the requirements for model importation also vary among different simulation engines. To enable multiple simulation engines to collaborate effectively and improve collaboration efficiency, an engine decoupling method is proposed. Firstly, it introduces the existing 3D scene simulation engines, analyzes their current status and limitations, and highlights the necessity of engine decoupling. Then, it elaborates on the implementation steps and technical details of the engine decoupling method. This method has been applied to actual projects, and its effectiveness and practicality have been verified through practical application, which enhances project development efficiency, reduces project costs, and provides significant convenience for 3D scene simulation.

### Keywords:

Digital twins; Industrial internet; 3D simulation; Engine decoupling

引用格式: 崔秋然,由志远,张勇,等. 数字孪生中三维仿真引擎解耦方法设计与应用[J]. 邮电设计技术, 2025(10): 59-63.

## 0 引言

随着科技的迅速发展和工业智能制造的快速推进,建筑信息模型(BIM)、数字孪生、物联网(IoT)等新兴技术被广泛应用于各行各业中<sup>[1]</sup>。数字孪生技术通过在虚拟空间中构建与实体一致的数字镜像或映射,

将真实世界中的实体以及实体的运行过程转化为虚拟世界中的数据,并基于物理模型、传感器数据以及历史运行数据驱动虚拟实体,使虚拟实体能够实时反映真实实体的当前状态、预测真实实体未来的行为和性能,从而实现对真实实体全生命周期的动态仿真<sup>[2]</sup>。

数字孪生的核心价值在于通过虚拟与真实系统的高度集成,实现对复杂系统的精准监测、预测性维护、优化设计以及决策支持,极大地提高了生产效率、

收稿日期: 2025-09-03

降低了运营成本,并增强系统的安全性和可靠性<sup>[3-5]</sup>。

## 1 数字孪生发展现状与挑战

数字孪生最早于2003年由Grievess教授提出,定义为包含实体产品、虚拟产品以及二者之间连接的三维模型<sup>[6]</sup>。2010年,美国国家航空航天局在太空技术路线图中首次引入了数字孪生的概念,期望基于数字孪生技术实现飞行系统的全面诊断维护<sup>[7]</sup>。2011年,美国空军研究实验室和NASA共同提出了构建未来飞行器的数字孪生体<sup>[8-9]</sup>。随后,Gabor等人提出数字孪生还应包含专家知识,以实现对真实实体的精准模拟<sup>[10]</sup>。Rios等人认为数字孪生应该面向更加广泛通用的产品<sup>[11]</sup>。2018年,陶飞等人将数字孪生模型扩展为五维结构:物理实体、虚拟模型、服务系统、孪生数据和连接<sup>[12]</sup>。

在数字孪生的构建过程中,模型的质量与兼容性关乎整个项目的成功与否。目前主流的三维仿真引擎大致可分为2类:一是以虚幻引擎(Unreal Engine)、Unity为代表的游戏引擎,二是Cesium、Three.js等面向轻量化应用的引擎。虚幻引擎和Unity具有强大的渲染能力,可构建高度逼真的三维场景,但对设备性能要求较高<sup>[13]</sup>。相比之下,Cesium与Three.js等轻量化引擎凭借其低门槛、广兼容的优势,被广泛应用于Web端三维可视化中<sup>[14-15]</sup>。

由于现有引擎难以兼顾渲染效果与性能,且模型与引擎间的紧耦合关系限制了项目的灵活性,若需更换引擎,将面临模型迁移复杂、数据同步困难等挑战。此外,工业数字孪生对实时数据的集成与反馈具有较高的要求,虚拟模型需要动态响应真实实体的变化。当前大多数系统使用的模型导入方式难以实现跨引擎数据的无缝流转与同步更新,增加了系统维护的复杂性和成本。

鉴于上述问题,本文提出了一种引擎解耦方法,基于数字孪生低代码平台构建统一的中央模型库,作为连接不同三维仿真引擎的桥梁。该模型库不仅负责存储和管理所有三维模型数据,还具备数据同步的能力,确保模型在不同引擎间的无缝迁移与实时更新。同时,该平台还集成了低代码开发环境,降低了数字孪生应用的开发门槛,提高了项目开发效率。

## 2 引擎解耦方法设计

本文所提出的引擎解耦方法通过搭建一个中央

模型库保存常用的设备和场景模型信息,利用Revit客户端中的模型轻量化插件,将从数据采集系统或者三维建模软件中获得的模型转化为统一的格式,并上传至中央模型库中存储管理,三维仿真引擎可以与中央模型库建立连接,获取模型信息,从而实现三维场景可视化,并在三维场景中实现用户交互、实时数据反馈等功能。引擎解耦技术路线如图1所示。

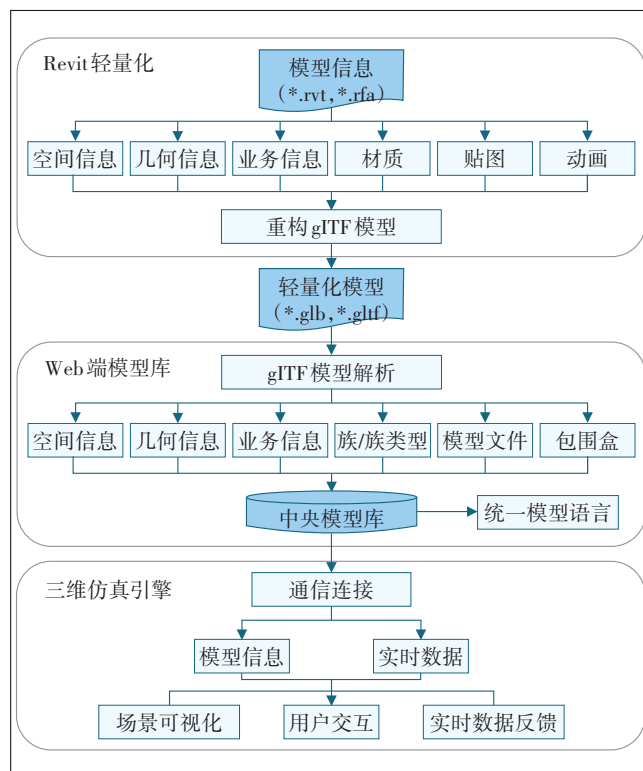


图1 三维仿真引擎解耦技术路线

### 2.1 模型轻量化处理

目前常用的建模软件有3DMax、Maya等,不同的建模软件导出的模型结构和格式差异较大,且不同的三维引擎支持的模型文件格式略有不同,这对同一模型在多个三维引擎中的复用十分不利。为了防止同一模型导入到不同三维引擎中出现加载错误、材质丢失等问题,需要对模型格式进行统一化处理。因此,为了实现三维模型在多个引擎中的复用,本平台对模型进行轻量化、标准化处理,将不同格式的模型转化为统一格式的glTF或glb文件。

glTF是一种基于JSON的文本文件格式,通过简洁的结构即可描述3D场景中的几何形状、材质、动画等信息<sup>[16]</sup>。glTF文件中各字段之间的关系如图2所示。其中scene是整个三维场景的入口点,描述了整个场

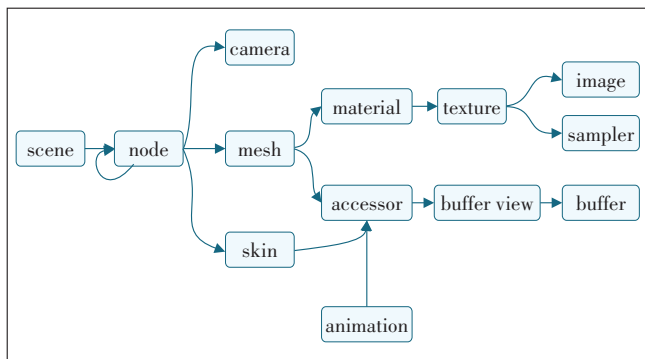


图2 glTF文件中各字段关系

景的组织结构;node是场景中的一个节点,包含位置变换信息,每个场景由多个节点构成;camera描述了此节点的相机所在位置;mesh描述场景中的几何物体;material描述模型的材质信息,包括材质的漫反射值、镜面反射值等;texture描述如何将纹理映射到相应的模型;image描述用于创建纹理的图像信息;animation描述模型的骨骼动画;skin描述模型绑定到骨骼上的参数;accessor描述如何从二进制数据源中获取数据;buffer view描述如何从buffer中读取数据;buffer指向包含实际数据的二进制文件<sup>[17]</sup>。

本平台利用Revit客户端的模型轻量化插件对模型进行轻量化处理,并将处理后的模型导出为通用的glTF或glb格式文件。标准化处理后的模型保留了模型的几何、位置、空间、材质等关键信息,降低了模型的复杂度,提高了模型的加载速度。模型可上传至中央模型库,并依据模型库规范对模型进行分类和存储。模型标准化处理既能够为不同的业务需求提供相应的数据信息,又有效解决了多个三维引擎之间模型难以互通的问题。

中央模型库分为场景模型库和设备模型库,场景模型库中保存了三维场景的建筑、环境等信息;设备模型库中保存了工业互联网中常用的设备信息,如机柜、空调等。

## 2.2 场景解析

传统的场景搭建方式通常是在三维建模软件中,对整个场景中的所有模型进行建模,然后将整个场景导入到三维仿真引擎中完成后续的开发工作,这种方式对模型的复用程度较低,项目后期维护成本较高,如果需要对模型进行更新,则需要在建模软件中修改模型后再次导入三维仿真引擎中。然而对于同类型的应用,同一模型可以在多个场景中复用。因此,本

文采用模块化设计理念,将场景分解为若干个可独立编辑的模型,每个模型作为独立的资源进行管理,当需要对某个模型进行更新时,只需要在模型库中修改对应的模型,所有引用该模型的场景将自动同步更新,无需重复导入导出,从而降低场景维护复杂度,提升场景搭建效率。

用户可以利用本平台开发的中央模型库中已有的模型快速搭建场景,通过数据采集系统或者建模软件获得场景的相关数据信息,将信息存储在JSON文件中并导入平台。平台解析JSON数据得到场景内模型的类型以及位置信息,然后在模型库中调用相关的模型,从而构建整个场景。

场景JSON数据各字段定义如表1所示。中央模型库中存储的模型具有唯一标识uuid,通过对场景JSON数据的解析,获取场景中模型的uuid,并利用uuid查找模型库中是否存在该模型,若存在,则直接使用模型库中的模型,若不存在,则从外部导入该模型并上传至模型库中。worldPosition、worldDirection和worldScale字段中存储了场景中模型的位置信息,利用该信息可以将模型放到场景中的对应位置。若一个模型内部还嵌套有其他子模型,可通过解析children字段获取子模型信息。

表1 场景JSON数据各字段定义

| 字段             | 字段类型   | 必填 | 备注            |
|----------------|--------|----|---------------|
| name           | string | 是  | 场景名称          |
| type           | string | 是  | 用于区分场景模型和设备模型 |
| uuid           | string | 是  | 模型唯一标识        |
| category       | string | 是  | 模型类型          |
| roomName       | string | 否  | 模型所属房间        |
| worldPosition  | vector | 是  | 模型位置信息        |
| worldDirection | vector | 是  | 模型朝向信息        |
| worldScale     | vector | 是  | 模型缩放信息        |
| children       | array  | 否  | 该模型内的其他子模型    |
| relateConfig   | array  | 否  | 模型实时数据        |

## 2.3 引擎间通信

为了兼顾渲染效果和系统性能,本平台采用UE+Three.js的设计方案,基于Three.js实现中央模型库的开发和场景数据解析等工作,模型库中的模型可以按照规定的JSON格式传输到虚幻引擎中,并在虚幻引擎中对场景进行还原。本平台基于WebSocket技术实现虚幻引擎和中央模型库之间场景数据和实时数据

的跨引擎传输。WebSocket 是一种在单个 TCP 连接上进行全双工通信的协议,该协议允许服务器主动向客

户端推送信息,客户端也可以主动向服务器发送信息<sup>[18]</sup>。WebSocket 通信流程如图 3 所示。

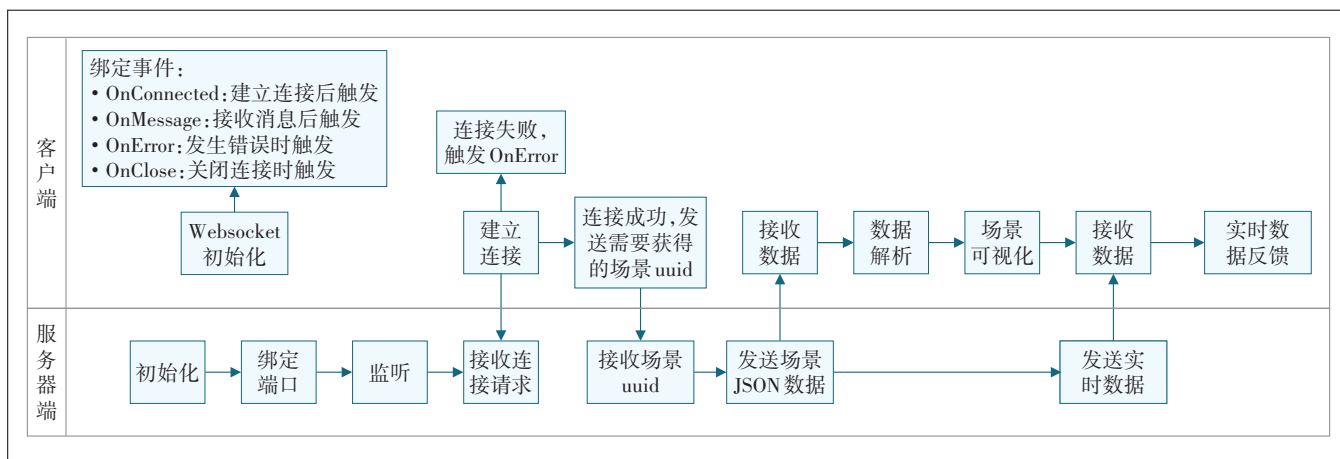


图3 WebSocket通信流程

本平台中 Three.js 端作为服务器端,创建 WebSocket 服务器实例,绑定特定端口并监听来自客户端的连接请求。虚幻引擎端作为客户端,创建 WebSocket 客户端组件,定义连接 URL、是否初始化连接等属性,并实现建立连接、断开连接、接收消息等绑定事件。客户端发送获取场景数据的请求到服务器。服务器接收到请求后,将相应的场景 JSON 数据通过 WebSocket 发送给客户端。客户端接收到这些数据后对数据进行解析,并据此进行场景的可视化渲染。

## 2.4 实时数据反馈

在数字孪生应用中,实时数据的及时反馈十分重要,表 1 中的 relateConfig 字段存储了模型与实时数据源之间的绑定关系,该数据通过采集真实设备数据获得,relateConfig 字段示例如图 4 所示。其中 uuid 为实时数据所绑定模型的唯一标识;showPanelType 为布尔类型,用于描述当前实时数据面板是否展示;showPanelConfig 为 JSON 数组类型,用于描述需要展示的实时数据,其中 attr 字段为数据名称,id 为该实时数据的唯一标识。实时数据按照规定的时间间隔发送至虚幻引擎端,虚幻引擎端对数据进行解析,并根据实时数据驱动模型发生变化,从而展示真实实体当前变化。

## 3 引擎解耦方法应用场景

### 3.1 智慧工厂应用

本文所提出的方法可应用于智慧工厂数字孪生研发过程中,在掌握数据采集技术的基础上,集成现



图4 relateConfig字段示例

有功能模块,完成机床加工中心、烘干中心、质量检测中心、立体仓储区、电子围栏等工厂模拟环境搭建,在虚拟环境中实现机械臂、电机、环境监测传感器、电子围栏等工厂设备实时数据的反馈,推动工业制造流程优化。图 5 所示为本平台在某智慧工厂项目中的应用效果。

### 3.2 数据中心应用

本文所提出的方法可应用于数据中心数字孪生项目中,通过与动环数据对接,获取设备的相关信息,

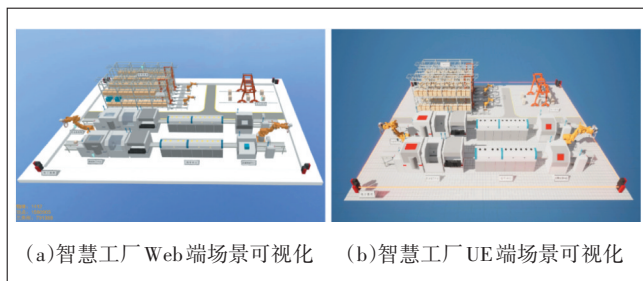


图5 智慧工厂应用效果

并将该数据以JSON格式传输给中央模型库,利用JSON数据和场景自动化构建数据,生成场景模型。设备的运行工况、能耗信息以及周围环境信息可以通过数据采集系统实时传输给虚幻引擎,并根据实时数据作出相应的反馈,有助于实现数据中心各设备的集中管理,改善数据中心效能问题。图6所示为本平台在某数据中心数字孪生项目中的应用效果。

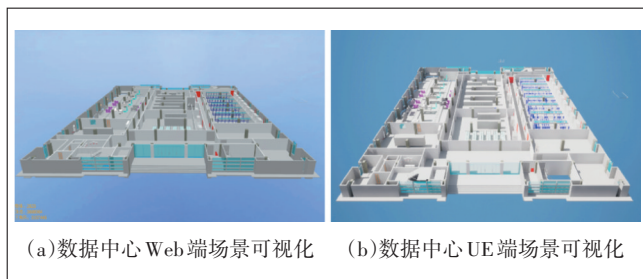


图6 数据中心应用效果

## 4 结束语

针对数字孪生领域中模型跨引擎难以互通这一问题,本文提出一种引擎解耦方法,并在多个应用场景下对该方法进行实验。经验证,该方法在一定程度上解决了多个引擎间模型信息互通的问题,提升了数字孪生应用中场景搭建的效率,降低了场景维护的成本。后续将持续对本平台的中央模型库进行扩充与优化,如增加通信行业等其他行业的常用模型等,以提升平台的普适性,扩大平台的使用范围。

## 参考文献:

- [1] TAO F, QI Q L. New IT driven service-oriented smart manufacturing: framework and characteristics [J]. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2019, 49(1): 81-91.
- [2] XIA L Y, LU J F, ZHANG H. Research on construction method of digital twin workshop based on digital twin engine [C]//2020 IEEE International Conference on Advances in Electrical Engineering and

- Computer Applications (AEECA). Dalian: IEEE, 2020: 417-421.
- [3] 刘彬, 张云勇. 基于数字孪生模型的工业互联网应用[J]. 电信科学, 2019, 35(5): 120-128.
- [4] 门志刚. 基于5G和工业互联网的智能制造系统探究[J]. 数字技术与应用, 2024, 42(5): 65-67.
- [5] 王晓颖, 顾钦子. 建筑全生命周期中的数字孪生技术应用研究[J]. 建筑科技, 2024, 8(7): 140-143.
- [6] GRIEVES M W. Product lifecycle management: the new paradigm for enterprises [J]. International Journal of Product Development, 2005, 2(1/2): 71-84.
- [7] TUEGEL E J, INGRAFFEA A R, EASON T G, et al. Reengineering aircraft structural Life prediction using a digital twin [J]. International Journal of Aerospace Engineering, 2011, 2011(1): 154798.
- [8] 杨林瑶, 陈思远, 王晓, 等. 数字孪生与平行系统: 发展现状、对比及展望[J]. 自动化学报, 2019, 45(11): 2001-2031.
- [9] 刘大同, 郭凯, 王本宽, 等. 数字孪生技术综述与展望[J]. 仪器仪表学报, 2018, 39(11): 1-10.
- [10] GABOR T, BELZNER L, KIERMEIER M, et al. A simulation-based architecture for smart cyber-physical systems [C]//2016 IEEE International Conference on Autonomic Computing (ICAC). Wuerzburg: IEEE, 2016: 374-379.
- [11] RIOS J, HERNANDEZ J C, OLIVA M, et al. Product avatar as digital counterpart of a physical individual product: literature review and implications in an aircraft [C]//22nd ISPE Inc. International Conference on Concurrent Engineering (CE2015). TU Delft: The Authors and IOS Press, 2015: 657-666.
- [12] 陶飞, 刘蔚然, 刘检华, 等. 数字孪生及其应用探索[J]. 计算机集成制造系统, 2018, 24(1): 1-18.
- [13] 李梅, 姜展, 满旺, 等. 基于虚幻引擎的智能矿山数字孪生系统云渲染技术[J]. 测绘通报, 2023(1): 26-30.
- [14] 任宏康, 祝若鑫, 李风光, 等. 基于Three.js的真实三维地形可视化设计与实现[J]. 测绘与空间地理信息, 2015, 38(10): 51-54.
- [15] 高云成. 基于Cesium的WebGIS三维客户端实现技术研究[D]. 西安: 西安电子科技大学, 2014.
- [16] 吕婧, 金浩然, 谭军, 等. glTF在BIM模型轻量化中的应用[J]. 科技创新与应用, 2020(6): 174-176.
- [17] 赵艳松. 基于glTF的三维模型服务关键技术研究[D]. 西安: 西安电子科技大学, 2017.
- [18] 李代立, 陈榕. WebSocket在Web实时通信领域的研究[J]. 电脑知识与技术, 2010, 6(28): 7923-7925, 7935.

## 作者简介:

崔秋然, 助理工程师, 硕士, 主要从事数字孪生研究工作; 由志远, 高级工程师, 硕士, 主要从事数字孪生研究和网络软件研发工作; 张勇, 高级工程师, 硕士, 主要从事数字孪生研究和可视化技术开发工作; 田阿康, 工程师, 硕士, 主要从事数字孪生、可视化平台研发工作; 刘江锋, 中级工程师, 学士, 主要从事智慧城市、智慧文旅等行业数字孪生产品研发工作。