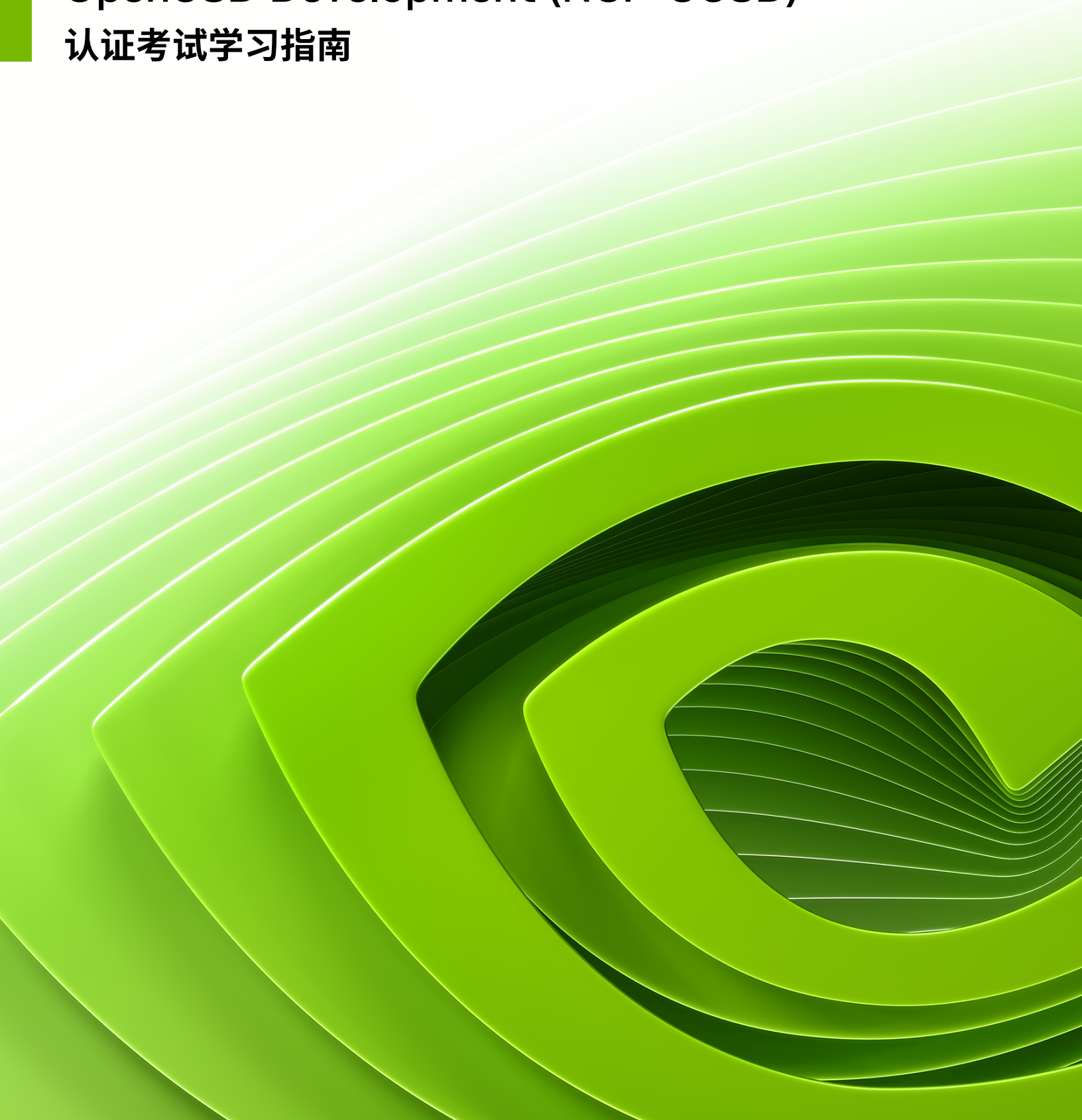




NVIDIA-Certified Professional: OpenUSD Development (NCP-OUSD)

认证考试学习指南





NVIDIA-Certified Professional: OpenUSD Development (NCP-OUSD) 认证考试学习指南

目录

合成：考试权重 23%	4
内容聚合：考试权重 10%	6
定制化 USD：考试权重 6%	7
数据交换：考试权重 15%	8
数据建模：考试权重 13%	9
调试和故障排除：考试权重 11%	10
workflow 开发：考试权重 14%	11
可视化：考试权重 8%	12

针对 NVIDIA-Certified Professional: OpenUSD Development (NCP-OUSD) 认证考试，本学习指南包含认证所涵盖的各个技术主题的介绍，并推荐了相关培训课程和阅读资料。

考试适用人群

OpenUSD 开发者，使用 OpenUSD 构建、维护和优化 3D 内容创建 workflow。从事 3D 内容开发和软件工程技术的专业人士，集成复杂的工具和工作流，确保艺术家、开发者和技术总监等团队高效协同。

从事于动画、视觉效果、游戏、制造、AECO 和虚拟生产等行业。深入理解 3D 数据结构、渲染系统和 OpenUSD 框架，可实现高效的资产交换、协作和场景管理。

相关工作职责示例

- 开发和维护基于 OpenUSD 的 3D 内容创建 workflow。
- 将 OpenUSD 与现有工具集成，并优化 3D 资产性能。
- 与领域专家协作，自动执行任务并解决 workflow 问题。
- 记录流程并培训团队成员使用 OpenUSD workflow。
- 及时了解 OpenUSD 进展，推动创新和可扩展性。

建议具备的知识和经验

- 具备 OpenUSD 框架和 Python/C++ 编程方面的专业知识。
- 具备 3D 内容创建、问题解决和版本控制 (Control) 技能。
- 跨团队沟通和对渲染系统的理解。
- 注重性能优化，具备持续学习思维模式。

认证主题和参考资料

合成：考试权重 23%

创作、设计或调试合成弧。考生需要了解所有合成弧的工作原理和方式，以及每一种合成弧的适用场景。具备调试复杂的 LIVERPS 场景的能力。

- 1.1 更改设定值的强度
- 1.2 根据数据规模选择合适的实例化策略
- 1.3 比较 referencing、payloads 和 sublayers 的差异，并识别它们各自的使用场景
- 1.4 创建一个场景，其中多个元素以不同的时间偏移引用同一个动画资源
- 1.5 设计可支持多用户协作的场景分层策略
- 1.6 用通俗语言解释 LIVERPS
- 1.7 判断在不同情况下是否适合使用 Variants 构建资产
- 1.8 识别某一图层中的 opinion 未在合成 stage 中生效的原因
- 1.9 准备将内部资产交付给外部团队
- 1.10 在装配 stage 中移除实例化组件 prim 的属性
- 1.11 拆分单一资产以支持多协作工作流

NVIDIA 培训推荐 (可选) 及更多资源

推荐课程

- [OpenUSD: 创建合成弧](#)

学习资料

- [Referencing Layers](#)
- [Authoring Variants](#)

术语表

- [LIVERPS](#)
- [Sublayers](#)
- [Variant Set](#)
- [Reference](#)
- [Payload](#)
- [Timesample](#)
- [Value Clips](#)
- [Layer Offset](#)
- [Edit Target](#)
- [Change Processing](#)
- [Flatten](#)

API 文档

- [Scenegraph Instancing](#)
- [Basic Datatypes for Scene Description](#)
[Provided by Sdf](#)
- [Edit Target - Detailed Description](#)
- [UsdStage::Flatten \(\)](#)
- [UsdReference - Expressing references](#)
[without prim paths](#)

更多资源

- [USD Composition - SIGGRAPH 2019](#)
- [USD Book-Default Prim](#)
- [OpenUSD Code Samples - Add a Payload](#)

内容聚合：考试权重 10%

构建模块化、可复用组件，利用实例化 (原生实例化和点实例化) 来优化场景，并应用不同的策略来覆盖实例化资产，从而实现高效、优化且协同的资产 (模型) 聚合，用于构建大型场景。

- 2.1 向 pointInstancer 添加新的原型 (prototype) 对象
- 2.2 在不改变实例化状态的前提下，修改 instanceProxy 网格的颜色属性
- 2.3 高效隐藏 pointInstancer 实例
- 2.4 实现并管理 USD 实例化机制，在大规模场景中高效复用资产
- 2.5 在装配阶段，从实例化组件 prim 中移除属性

NVIDIA 培训推荐 (可选) 及更多资源

推荐课程

- [OpenUSD：理解 Model Kinds](#)
- [OpenUSD：资产结构原理和内容聚合](#)

技术参考资料

- [Modularity and Content Reuse Best Practices - Instancing](#)

API 文档

- [Scenegraph Instancing](#)
- [Kind : Extensible Categorization - The Core Kind Hierarchy](#)
- [场景说明的重要属性 - 模型层次结构：含义和目的](#)
- [UsdShadeNodeGraph - 详细说明](#)
- [UsdGeomPointInstancer - Detailed Description](#)

定制化 USD：考试权重 6%

理解 USD 插件开发，以扩展 USD 的功能，包括创建自定义 schema、自定义文件格式插件、自定义模型类型，以及变体回退机制的配置。

- 3.1 基于指定版本的 USD 构建插件
- 3.2 从零开始构建 USD，并集成自定义依赖项
- 3.3 适时创建自定义 model kinds
- 3.4 为专有数据模型创建自定义 Schema
- 3.5 集成自定义解析器，实现资产路径的动态管理
- 3.6 使用 USD 模式支持导入 / 导出过程中的非标准属性或结构
- 3.7 编写 SceneIndex 插件，直接生成可渲染的 Hydra 图元
- 3.8 编写资产解析器，实现内存中可渲染图元的生成

NVIDIA 培训推荐 (可选) 及更多资源

推荐课程

- [OpenUSD：理解 Model Kinds](#)

学习资料

- [Generating New Schema Classes](#)

术语表

- [Model Hierarchy](#)

白皮书和博客

- [What Are OpenUSD Schemas?](#)

视频

- [Universal Scene Description \(OpenUSD\): Custom Schemas](#)
- [USD in Production](#)

技术参考资料

- [USD FAQ - Isn't USD just another file format?](#)

API 文档

- [Creating a File Format Plugin](#)
- [Creating New Schema Classes with usdGenSchema](#)
- [Kind : Extensible Categorization](#)
- [PlugRegistry - Registering Plug - ins](#)
- [Ar: Asset Resolution](#)

数据交换：考试权重 15%

创建概念性数据映射文档、自定义导入器、导出器以及用于与 OpenUSD 进行数据交换的脚本。

- 4.1 将 OpenUSD 资产转换为常见的 3D 格式 (例如 glTF)，同时确保转换过程中的数据保真性
- 4.2 记录 USD 与其他数据模型 (如 MaterialX) 之间的概念性映射关系
- 4.3 解释 USDC 与 USDA 格式之间的权衡 (包括性能、可读性、归档等方面)
- 4.4 在 DCC 中实现 USD 的双向数据交换流程 (round-trip)
- 4.5 使用 USD schemas 支持导入 / 导出过程中的非标准属性或结构
- 4.6 编写验证器, 确保从 DCC 导出的 OpenUSD 资产具备结构和数据完整性
- 4.7 开发用于生成 USD 格式的导出器或转换器
- 4.8 在 DCC 中写入或扩展通用场景描述导入器

NVIDIA 培训推荐 (可选) 及更多资源

推荐课程

- [OpenUSD：开发数据交换 workflow](#)

学习资料

- [Traversing a Stage](#)
- [Converting Between Layer Formats](#)

术语表

- [Over](#)
- [Stage Traversal](#)

技术参考资料

- [Conceptual Data Mapping](#)
- [USD Toolset - usdchecker](#)
- [USD Toolset - usdcat](#)
- [USD Toolset - usdzip](#)

技术参考资料 (续)

- [USD FAQ - What file format is my .usd file?](#)
- [USDZ File Format Specification](#)
- [Maximizing USD Performance](#)

API 文档

- [Encoding Stage UpAxis](#)
- [Creating New Schema Classes with usdGenSchema](#)
- [Common Idioms and Examples](#)
- [UsdGeomPoints - Detailed Description](#)
- [UsdProperty::GetNamespace \(\)](#)
- [UsdGeomModelAPI::GetExtentsHint \(\)](#)
- [UsdGeomBoundable::ComputeExtentFromPlugins \(\)](#)
- [UsdGeomBoundable::GetExtentAttr \(\)](#)

数据建模：考试权重 13%

掌握 Usd 与 Sdf 框架中的核心数据结构与类型，包括 prims、属性 (attributes/relationships)、primvars、valueTypes (如 float、token、matrix4d 等)、timeSamples，以及内建的 USD schemas。

- 5.1 在网格几何体上添加 primvar 属性
- 5.2 为属性数据选择适当的 valueTypes 进行存储
- 5.3 表示自定义元数据
- 5.4 检索 prim 属性
- 5.5 掌握引发视觉异常结果的常见原因及其排查方法
- 5.6 在更新网格的点数据后，更新其 extent 属性

NVIDIA 培训推荐 (可选) 及更多资源

推荐课程

- [OpenUSD：学习 Stages, Prims 和 Attributes](#)
- [OpenUSD：学习使用 Prims 和默认 Schemas](#)
- [学习 OpenUSD：使用属性](#)
- [OpenUSD：构建基本动画](#)

学习资料

- [Transformations, Time-sampled Animation, and Layer Offsets](#)
- [Generating New Schema Classes](#)

用户指南

- [Rendering User Guide - Working with Primvars](#)

术语表

- [TimeCode](#)
- [Layer Offset](#)
- [List Editing](#)
- [Primvar](#)

技术参考资料

- [USD Proposals - Spline Animation in USD](#)
- [Maximizing USD Performance](#)

API 文档

- [Basic Datatypes for Scene Description Provided by Sdf](#)
- [Encoding Stage Linear Units](#)
- [UsdAttribute - Attribute Interpolation](#)
- [UsdGeomPrimvar - Detailed Description](#)
- [UsdGeomPrimvar - Interpolation of Geometric Primitive Variables](#)
- [Sdf : Scene Description Foundations - Types](#)
- [Creating New Schema Classes with usdGenSchema](#)
- [UsdGeomBoundable](#)
- [UsdGeomXformable](#)
- [UsdGeomXformCache](#)
- [UsdStage:SetInterpolationType \(\)](#)

调试和故障排除：考试权重 11%

分析 USD stage，定位组合异常、识别不良数据编写，并提升加载与渲染效率。

6.1 识别何时使用 SdfChangeBlocks 提升性能

6.2 识别某一层的 opinion 未在最终组合 stage 中生效的原因

6.3 解决资产管理相关的问题

6.4 理解导致意外视觉结果的原因

6.5 使用诊断工具进行调试与性能分析 (TfDebug、diagnostic delegates、Trace、TfMallocTag 等)

NVIDIA 培训推荐 (可选) 及更多资源

推荐课程

- [OpenUSD：创建合成弧](#)

学习资料

- [Inspecting and Authoring Properties](#)
- [Traversing a Stage](#)

术语表

- [LIVERPS](#)
- [Reference](#)
- [Visibility](#)
- [IsA Schema](#)
- [Inherits](#)

技术参考资料

- [Maximizing USD Performance](#)
- [USD Toolset - usdcats](#)
- [通 USD FAQ - What's the difference between an "over" and a "typeless def" ?](#)
- [usdview LayerStack tab](#)

API 文档

- [UsdPrim](#)
- [UsdReference - Expressing references without prim paths](#)
- [UsdGeomModelAPI - Draw Modes](#)
- [UsdProperty::GetPropertyStack \(\)](#)
- [UsdUtilsCoalescingDiagnosticDelegate](#)
- [TfDiagnosticMgr::Delegate](#)
- [UsdStage::Open \(\) - pathResolverContext](#)
- [UsdStage - Variant Management](#)
- [UsdStage - Working Set Management](#)
- [UsdStage::MuteLayer \(\)](#)
- [UsdReferences - Reasons why adding a reference may fail...](#)
- [Pcp : PrimCache Population \(Composition\) - Errors](#)

workflows开发：考试权重 14%

OpenUSD 开发者或架构师需要掌握以下高阶任务，涵盖从管线设计到资产管理、版本控制、图示绘制、文档编写、用户界面与用户体验优化、编写 USD 导出钩子以转换数据结构、构建配置管理，以及资产扁平化处理与去除专有依赖。这些任务不仅涉及技术实现，还要求具备系统性思维与跨团队协作能力，是构建高效、可扩展 USD 工作流程的核心组成部分。

7.1 将 OpenUSD 资产转换为常见的 3D 格式 (例如 glTF)，并确保视觉保真度

7.2 编写资产结构指南

7.3 说明 USDC 与 USDA 格式之间的权衡，包括性能、可读性、归档等方面

7.4 在 DCC 工具与 USD 之间实现 round-trip 管线

7.5 集成自定义解析器，动态管理资产路径

7.6 表示自定义元数据

7.7 验证资产路径的格式是否正确

7.8 在 DCC 工具中编写或扩展 USD 导入器

NVIDIA 培训推荐 (可选) 及更多资源

推荐课程

- [OpenUSD：学习 Stages, Prims 和 Attributes](#)
- [OpenUSD：遍历 Stages](#)
- [OpenUSD：创建合成弧](#)
- [OpenUSD：资产结构原理和内容聚合](#)

技术参考资料

- [USD FAQ - I have some layers I want to combine: Should I use SubLayers or References?](#)
- [Principles of Scalable Asset Structure in OpenUSD](#)
- [ASWF Asset Structure Guidelines](#)

学习资料

- [Authoring Variants](#)

术语表

- [Subcomponent](#)
- [Crate File Format](#)
- [TimeCod](#)
- [Flatten](#)
- [Edit Target](#)

API 文档

- [UsdStage::Flatten \(\)](#)
- [Edit Target - Detailed Description](#)
- [UsdStage::Traverse \(\)](#)
- [UsdPhysics : USD Physics Schema](#)
- [Encoding Stage Linear Units](#)
- [Encoding Stage UpAxis](#)
- [SdfChangeBlock](#)
- [UsdNotice](#)
- [UsdGeom : USD Geometry Schema - Applying Timesampled Velocities to Geometry](#)

可视化：考试权重 8%

与 UsdGeom、UsdShade 和 UsdLux 等 USD 域相关的任务 (例如网格、相机、材质和灯光)。UsdGeom、UsdShade 和 UsdLux 是 USD 中的核心域，涵盖网格、相机、材质和灯光等内容。考生需要熟悉相关任务，这些内容在 USD 的实际应用中具有广泛覆盖面和重要性。

8.1 向网格添加 primvar 属性

8.2 为资产可视化分配 UsdPreviewSurface 材质

8.3 将材质绑定到网格

8.4 创建一个 UsdPreviewSurface 着色网络，用于从 primvar 中读取漫反射颜色

NVIDIA 培训推荐 (可选) 及更多资源

推荐课程

- [OpenUSD：学习 Stages, Prims 和 Attributes](#)
- [OpenUSD：学习使用 Prims 和默认 Schemas](#)

用户指南

- [Rendering with USD](#)

术语表

- [Gprim](#)

API 文档

- [UsdShadeMaterial - Detailed Description](#)
- [UsdGeomSubset - Detailed Description](#)
- [UsdShadeMaterialBindingAPI - Detailed Description](#)
- [UsdGeomMesh - Detailed Description](#)
- [UsdLuxLightAPI](#)
- [UsdLuxShadowAPI](#)
- [UsdLuxDistantLight](#)
- [UsdLuxShapingAPI](#)
- [UsdGeom : USD Geometry Schema - Imageable Purpose](#)

遇到问题？

问题咨询，请点击[联系我们](#)。