# Package: libopenusd (via r-universe)

November 16, 2026

**Type** Package

**Title** 'OpenUSD' Scene Description Import and Export

**Version** 26.8-0

**Description** Provides the 'OpenUSD' 'C++' library for reading, composing, and writing Universal Scene Description files <https://openusd.org/>. Exposes stage import and export through R functions and a registered native interface for rendering packages.

**License** file LICENSE

**Copyright** file inst/COPYRIGHTS

**URL** https://github.com/tylermorganwall/libopenusd

**BugReports** https://github.com/tylermorganwall/libopenusd/issues

**Depends** R (>= 4.3.0)

**Imports** RcppParallel, Rcpp

**LinkingTo** Rcpp

**SystemRequirements** CMake (>= 3.22), GNU make

**Encoding** UTF-8

**NeedsCompilation** yes

**Roxygen** list(markdown = TRUE)

**Config/roxygen2/version** 8.0.0

**Config/pak/sysreqs** cmake make

**Repository** https://tylermorganwall.r-universe.dev

**Date/Publication** 2026-11-15 00:54:59 UTC

**RemoteUrl** https://github.com/tylermorganwall/libopenusd

**RemoteRef** HEAD

**RemoteSha** 4cfd44bbeb3804f4907ee7b419f1f9f21f02cc2b

# Contents

usd_inspect . . . . . 2
usd_messages . . . . . 3
usd_open . . . . . 3
usd_version . . . . . 4

**Index** **6**

---

`usd_inspect` *Inspect Typed USD Metadata*

---

## Description

Read prim custom data and selected properties through native OpenUSD queries. Values retain their numeric precision and array lengths.

## Usage

```
usd_inspect(filename, selectors = "", properties = character())
```

## Arguments

| | |
|---|---|
| `filename` | Path to a USD stage. |
| `selectors` | Default `""`. Prim paths to inspect. An empty string selects the default prim; a path ending in `/*` selects its immediate children. |
| `properties` | Default `character()`. Attribute names to read at default time. |

## Details

Supports dictionaries, boolean/integer/float/double/string/token scalars and arrays, unsigned-byte arrays, and asset paths. Other value types fail explicitly. This API and its registered `libopenusd_inspect_v1` native entry point use R objects and must only be called on R's main thread.

## Value

A list of prim records with paths, types, custom data, properties, and child paths. Asset values contain authored and resolved paths. Missing prims are omitted and missing properties return `NULL`.

## Examples

```
# Read a custom scalar without converting through a text serialization.
scene = usd_from_usda('#usda 1.0\ndef Xform "World" (customData = { int count = 4 }) {}')
path = tempfile(fileext = ".usdc")
usd_export(scene, path)
usd_inspect(path, "/World")[[1]]$custom_data$count
usd_close(scene)
unlink(path)
```

## usd_messages *Read Buffered OpenUSD Diagnostics*

### Description

Retrieve native library messages to investigate warnings and debug output produced while reading or writing scenes.

### Usage

```
usd_messages()
```

### Details

Messages from native workers are collected without calling R on those threads. The queue holds 128 messages of up to 2047 bytes each and reports overflow.

### Value

A character vector of buffered messages, clearing those messages from the queue. Returns `character(0)` when no messages are pending.

### Examples

```
# Inspect any diagnostics from earlier scene operations.
usd_messages()
```

## usd_open *Read and Write OpenUSD Stages*

### Description

Open, compose, and export USD scenes using the native OpenUSD library. Stage handles retain their layers until closed or garbage collected.

### Usage

```
usd_open(filename)

usd_create()

usd_from_usda(text)

usd_to_usda(stage, flatten = FALSE)

usd_export(stage, filename, flatten = FALSE)

usd_close(stage)
```

### Arguments

| | |
|---|---|
| filename | Path to a USD file. The extension selects the output format: .usda for text or .usdc for binary. Reading also supports USDZ archives. |
| text | One character string containing a complete USDA layer. |
| stage | A stage handle returned by these functions. |
| flatten | Default FALSE. Export only the root layer, preserving its composition arcs. Use TRUE to export composed opinions into a single layer. |

### Details

Handles belong to the current R process and cannot be restored with saveRDS(). Export a USD file and reopen it in the next session instead.

Export does not copy textures, referenced layers, volumes, or other assets. Root-layer export preserves authored relative asset paths, which must remain valid relative to the destination. Anonymous layers imported from text have no file location; use absolute asset paths or open a file with usd_open() when relative references are required. Flattening resolves composition but does not embed external assets. USDZ packaging is not provided by this API.

Compiled consumers can include libopenusd.h and obtain the versioned R_GetCCallable() function table documented in the installed headers.

### Value

usd_open(), usd_create(), and usd_from_usda() return an external pointer of class usd_stage. usd_to_usda() returns a character string. usd_export() invisibly returns the output path. usd_close() returns invisible NULL.

### Examples

```
# Write a native binary USD file and reopen it for a later rendering step.
scene = usd_from_usda('#usda 1.0\ndef Sphere "Ball" {\n double radius = 2\n}\n')
filename = tempfile(fileext = ".usdc")
usd_export(scene, filename)
restored = usd_open(filename)
cat(usd_to_usda(restored))
usd_close(scene)
usd_close(restored)
unlink(filename)
```

---

## usd_version *Report the Bundled OpenUSD Version*

---

### Description

Return the upstream library release used by the native stage interface.

## Usage

```
usd_version()
```

## Value

A character string containing the OpenUSD release version.

## Examples

```
# Distinguish the upstream release from the R packaging revision.
usd_version()
packageVersion("libopenusd")
```

# Index

usd_close *(usd_open)*, 3
usd_create *(usd_open)*, 3
usd_export *(usd_open)*, 3
usd_from_usda *(usd_open)*, 3
usd_inspect, 2
usd_messages, 3
usd_open, 3
usd_to_usda *(usd_open)*, 3
usd_version, 4