API Changes from 25.X to 26.X

Many user facing changes are listed in the Release Notes.

This document lists the main issues for developers who are migrating from 25 to 26

For migration from 23 to 25, please refer to Migrating from 23.X to 25.

Riley

Please note that as of PRMan 26.0, Riley **should still be considered an unstable interface**.

- The Riley header file version has been incremented to 0.4.
- An inconsistency in const correctness for ModifyRenderOutput() has been corrected.
- New methods have been added for creating, modifying, and deleting instances in batches: CreateGeometryInstances(), ModifyGeometryInstances(), DeleteGeometryInstances(), CreateLightInstances(), ModifyLightInstances(), and DeleteLightInstances(). Currently, these routines do not have any significant advantages over their non-batched versions (and both RIS and XPU have had significant speed improvements when using the non-batched calls); however, in future releases, these methods may provide significant speedups for operating on large numbers of instances. Note that the Create and Modify routines have specific requirements for memory allocation; for more information, please refer to the Riley.h header.

RixRiCtl

- PRManBegin() and PRManEnd() have been augmented with new separated calls for "system" and "render". The old entrypoints are still there for standalone renders, but under the hood have been rewritten to use the new split API. These routines allow DCCs to intermingle rendering via the USD Hydra delegate and via the traditional direct calls.
    - PRManSystemBegin() and PRManSystemEnd() is for application startup and shut down. Nested system begin calls are reference counted.
    - PRManRenderBegin() and PRManRenderEnd() is for render startup and shutdown. Nested render begin calls are not allowed; PRManRenderEnd() must be called before starting another render.

Deprecated APIs

The following interfaces should be considered deprecated, and will be removed in a future release:

- dtex
- RixDeepTexture
- RixSymbolResolver
- RixResourceResolver
- RixStorage
- RixMutex
- RixThreadUtils