

USD Transforms Stack



Take Away

- Understand the USD transform stack (xformOp)
- Know the available USD transform ops
- Know the structure of the USD transform stack
- Know how it maps to UFE
- Know the various implementations of xformOp in MayaUSD

What is a USD Transform stack

- Control the 3D position of a USD prim
- It is an ordered list of transform operations (xformOp): translation, rotation, scale, 4x4 matrix
- USD supports any number of xformOp, in any order
- An individual xformOp is kept in a USD attribute
- All transform-related attributes begin with the "xformOp:" prefix
- An xformOp attribute name has two or three parts: "xformOp:", the transform type, optional suffix
- For example: "xformOp:translate:pivot"
- The xformOp order is kept in a special attribute: "xformOpOrder"

Quick Recap on Pivot

Pivots...

- ... are generally neutral: they don't move the prim
- ... come as a pair of opposite translations, sandwiching other transforms
- ... are used in DCC to position the center of rotation and center of scaling

Quick Recap on Transform Maths

- A single matrix is equivalent to any number of chained transforms
- The inverse is not true
- Some matrices can't be expressed with translation, rotation and scaling...
- ... but they are generally considered degenerate and are rarely seen
- On the other hand, matrix cannot express pivot pairs
- Because pivot pairs are neutral, they don't move the prim at all
- In general, except for pivots, all transform representations are equivalent

USD Common Transform Stack

- USD provides a recommended, simple transform stack called "common API"
- The goal is to have a baseline transform stack that all DCC should support
- The USD common stack structure is: translate, pivot, rotate, scale
- In particular, the pivot wraps both the rotation and scaling, unlike Maya

UFE Transforms

UFE Transform3d...

- ... allows modifying a UFE scene item transforms
- ... is created by the registered UFE Transform3dHandler
- ... creates commands that when executed changes the transform
- ... does **not** prescribe how the various transforms interact
- ... is implicitly tied to Maya's view of how transforms are ordered

MayaUSD XformOp Implementations

- MayaUSD provides multiple UFE Transform3d implementations
- The implementations are:
 - point instance
 - Maya stack
 - USD common API
 - 4x4 matrix
 - no compendo + Maya stack
- For the rest of the presentation, we will ignore point instance and no compendo

MayaUSD XformOp Details

- Only the Maya stack supports all the UFE commands
- In particular, pivot commands
- Which one is used is decided at the time of the creation of the Transform3d
- Once decided, there is no turning back, we cannot switch dynamically
- The decision is based on what is already authored on the prim
- In case multiple implementations could be used, the priority order is:
 - Maya stack
 - common API
 - matrix
- The main goal was that we would privilege the Maya stack, but still support the others representations.