1.4.0 Change Log

Primary Changes

1) The In-Editor Guide has been moved online and can now be found at https://deepspacelabs.net/html/sam/docs/documentation_main.html. The Help -> Deep Space Labs -> SAM Editor Guide option still exists, however it now opens a window that explains the move and provides the above link, as well as some general information.

2) Moved some settings from included Chunk Streamer implementations to a new global settings asset, as the settings are really intended to apply to specific types of streamers globally. This asset is created under Deep Space Labs/SAM/Resources the first time it is needed, which allows you to adjust the settings and not have them be overwritten the next time you update SAM.

   These moved settings will need to be reexamined as the move will reset them. These include **Set Load Priority** and **Load Priority** (shown only if Set Load Priority is enabled). These settings can be adjusted via any Chunk Streamer in your scenes.

   In addition, some new Scene Chunk Streamer settings have also been added to this asset, allowing you adjust some settings related to the sorting of Build Setting Scenes. Again, you also can adjust these via any Scene Chunk Streamer in your scenes.

3) Reworked the World Designer Tool UI to be a bit less cluttered. Advanced Operation Settings have been moved to their own window in the top center portion of the Tool window.

4) Added three new options for the **What To Assign** setting (used for Assignment Operations using the World Designer Tool). The new options **are Game Object With Tag**, **Game Objects On Layer**, and **Custom Assignment Selector**. The first two options should be self-explanatory. The third allows you to create custom scriptable assets to define unique custom logic for selecting the objects that will be assigned. All three new options work similar to the Regex based method, in that the hierarchy is searched in the same way.

5) Added a new World Designer setting (under General Settings in the Advanced Operation Settings window) called **Assignment/Transfer Position Offset**. This new setting lets you

add an offset to the position of each game object that is assigned during Assignment/Transfer operations. This can be used to fix some issues with floating point inaccuracies for objects on the borders of cells (where they might be assigned to the wrong cell). Mainly useful for terrain.

6) Deprecated the **LODCrossFadeTransitioner**, and instead moved its functionality into the **MatPropBlockTransitioner** and **PerMaterialTransitioner** classes. All you need to do is enable the **Use Unity Crossfade** option to make these transitioners work with materials setup to work with Unity's built in LODGroup component. Keep in mind that the **Visual State Of Stored Assets When Active** setting (found on Streamable Grid assets) needs to be set to **Invisible** when using **Use Unity Crossfade**. This was also true when using an LODCrossFadeTranstiioner as well, however it was not documented correctly.

7) Added a new Player Mover Component called FixedUpdatePlayerMover, which you can use to make SAM based player position changes only happen in FixedUpdate.

8) When creating both prefabs and scenes with a Default Asset Creator, you must now define which of the types is **Primary Asset** (depending on the Chunk Streamer that will be used to load the created assets at runtime). This ensures only the primary asset is created when the World Designer Tool request a test asset chunk (used to ensure the Creator has been configured correctly). Before this change, it was possible for the secondary asset to not get cleaned up properly, resulting in extra assets being left behind in your project.

9) When using the Default Asset Creator, the created assets will have a default position equal to the position that they were first created to be spawned at within the editor. The previous behavior was for their positions to be 0,0,0, which was not ideal for certain situations.

10) Changed the return type for the AssetCreator's **CreateAssetsForCellChunk** method from IEnumerator to void. This shouldn't affect the majority of users, only those who have created a custom Asset Creator (unlikely to be anyone).

11) The **IsGrouping3D** and **GetStreamableGridOfGrouping** from the World class have been renamed to **IsGrouping3D_PreInitSafe** and **GetStreamableGridOfGrouping_PreInitSafe** respectively. As the name implies, these methods are now safe to use before the World has been initialized.

Fixes

1) Fixed a significant bug that was introduced with 1.3.0 that caused prefabs loaded by the World Designer Tool to be immediately marked as having unsaved changes.

2) Fixed another bug (possibly introduced in 1.3.0) that would cause invalid scenes to be created when using a Default Asset Creator set to create both prefabs and scenes.

3) Fixed a bug in the Scene Chunk Streamer that would result in Build Setting scenes not being detected if the scenes were not in alphabetical order.

4) Fixed a bug that could cause errors in how Load Keys were generated when using appended data (could have effected Chunk Streamers and World Designer Tool). It is unlikely users were making use of appended data, so this should not have affected anyone.

5) Fixed an improperly thrown exception that could occur when using one of the Addressable Chunk Streamers with the World Designer Tool.

6) Fixed an issue with the LOD Filter not showing correctly on the Static Batcher Listener component.

7) Fixed Cell Visual Transition Controller Animation Curves not being set correctly to default values when first being added to the scene.

8) Fixed scenes not being dirtied properly when assigning objects contained in the scenes using the World Designer Tool.

9) Fixed an issue with the World component LOD Group error reporting where the LOD Group reported as having an error would be incorrect (LOD 1 was reported as LOD 2, 2 as 3, and so on).

Other

1) Added a public static method called **TryGetWorlDesignerInUseByWorld** to the **WorldDesignerWindow** class that allows you to try and find an open World Designer Window that is currently in use by a specific World (only useable in Editor).