

Návod k SDK 3D Scény

Pro vytvoření 3D scény je možné použít SDK s předpřipravenou funkcionalitou, která odpovídá funkcím v ostatních typech scén. SDK je určeno pro Unity, avšak 3D scény lze vytvořit v libovolném prostředí.

Je také možné využít skripty a ostatní prvky přímo z repozitáře hlavního software, jsou zcela kompatibilní.

Požadavky

- Unity 2019, poslední LTS verze (2020.3.23f1 v době vydání této dokumentace)
- Poslední verze emt-sdk balíčku z GitHubu
- Poslední verze Unity SDK z GitHubu
- Použití URP rendereru (HDRP není podporované)
- Nainstalovaný Unity balíček pro export na Linux

Postup pro vytvoření scény

1. Vytvořte nový URP projekt v Unity
2. Přidejte stažený emt_sdk balíček v nabídce "Window -> Package Manager -> Add package from tarball..."
 - Pro bližší návod k přidávání balíčku je k dispozici dokumentace Unity zde (!!!odkaz!!! <https://docs.unity3d.com/Manual/upm-ui-tarball.html>)
3. Nahradte výchozí kameru prefabem *CameraRig* ze složky Prefabs obsažené v SDK, který realizuje transformaci obrazu

Komponenty

Konkrétní dokumentace k jednotlivým metodám a třídám je obsaženo přímo ve zdrojovém kódu pomocí XML dokumentace. Následují sekce v krátkosti popisují funkcionalitu nabízených komponentů dle jejich umístění v SDK.

Camera

Tyto komponenty slouží k aplikaci transformace obrazu pro IPW, konkrétně *DualCameraComponent* pro správu dvou přidružených kamer. Zároveň při startu scény načte IPW konfiguraci ze souboru (defaultně v ~/ipw.json) a aplikuje ji.

Pro samotné kamery je určen komponent *TransformCameraComponent*, který Unity kameře přidělí konkrétní konfiguraci displeje z IPW konfiguračního souboru.

Ve většině případů však stačí použít prefab s názvem *CameraRig*, který v sobě zahrnuje dvě perspektivní kamery se správnými komponenty, audio listener a jednoduchou reprezentaci avatara (kapsle s colliderem).

Content

Tyto komponenty se nevztahují přímo k IPW, jedná se o obecné komponenty pro zobrazení dat které jsou využité v ostatních datových typech. Jedná se o vlaječky v komponentu *FlagComponent*, které jsou stejné jako u 3D objektu a o komponent pro automatickou rotaci objektu směrem k jinému v *BillboardRotateComponent*. Tento efekt je vhodné použít například pro text.

Event

Tyto komponenty slouží k jednodušší interakci s event systémem. Základní verzí komponentu je *EventComponent*, který zaručuje bezpečné předání eventů napříč vlákny z emt-sdk do Unity. Tento komponent tyto eventy posílá jako *UnityEvent* dále do aplikace bez jakéhokoliv filtrování.

Dále jsou k dispozici specifické komponent umožňující filtrování dle například Typu pro snazší práci se specifický vstupem.

Pro ladění scény bez přístupu k IPW je možné použít debuggovací komponent *LocalInputComponent*, který sbírá data z klávesnice a myši a přeposílá je do event systému.

Extensions

Tyto třídy slouží ke konverzi datových typů mezi emt-sdk a Unity, jedná se zejména o vektory.

Materials

Zde naleznete referenční materiály použité v demo scénách.

Obsažen je materiál pro navigační čáry, který je viditelný i skrz geometrii.

Navigation

Tyto komponenty slouží k vytvoření scény s navigací po orientovaném grafu. Jedná se o podobný princip jako v interaktivních aplikacích typu Myst.

Export scény pro použití v IPW

1. Ujistěte se že používáte event relay pro komunikaci s IPW firmwarem a ne EventManager
2. Ideálně mějte pouze 1 (výchozí) scénu
3. Exportujte projekt skrz Unity pro platformu Linux