# **Dave Sound Dokumentation:**

## 3.11.2022 – Audio-Busse für Spiel-Sound-Settings und Parameter-Test

* Umstellung der **FMOD-Session auf Stereo** (statt 5.1) *[generell aber auch sehr einfache Möglichkeit, verschiedene Audio-Systeme zur Auswahl zu geben]*
* **Funktioniert 3D-Tracking** (aka Spatializing) in Game? *[nachdem bei Oasis ja leider nicht ging, dacht ich mir, diesmal checken wir das gleich am Anfang]* -> Anleitung zum Testen (oder Discord-Call und gemeinsam mal ausprobieren):
  + Test-Objekt mit „3D“ im Namen laden (entweder TestVoice-3D oder TestUI-3D)
  + Mit Listener im Bereich von 0 bis 20 Units *(? Weiß leider nicht welcher Größe das entspricht oder ob das sinnvoll ist; kann ich aber anpassen, sobald wir ein Gefühl für die Abstände haben)* um das Objekt bewegen (oder das Objekt um den Listener)

-> Sound sollte sich im Stereo-Bild entsprechend von links nach rechts bewegen und entsprechend der Entfernung leiser werden

* Erste Einbindung von **Parametern zum Testen**:
  + Erklärung FMOD-„Parameter“:
    - Kommunikation von Veränderungen im Gameplay zwischen Unity und FMOD
    - Beispiele: Velocity eines Objekts, Zustand eines Objekts (aufgeladen/ deaktiviert), Zustand des Spielers (Tod/ Lebendig, …), Fortschritt innerhalb eines Levels, …
    - Distanz zw. Objekt und Listener (von Spatializing) wird bei FMOD intern auch als Parameter gehandhabt, muss aber innerhalb von Unity nicht zugewiesen werden
    - Je nach Einstellung hat jede Event-Instanz einen eigenen Parameter-Wert oder ruft einen allg. gültigen Parameter auf (local vs. global Parameter)
    - entspricht RTPCs von Wwise
  + (generelles) Einbindung von Parametern in Unity:
    - Hinzufügen eines FMOD Studio Event Emitters als Component zu einem Objekt
    - (Im hinzugefügten Component kann man bei Initial Parameter Value einen extra Start-Wert einstellen – mache ich aber eigentlich auch bei FMOD intern; gleichzeitig kannst du hier aber sehen, welche Parameter mit dem Event verknüpft sind)
    - Live-Anpassung von Parameter-Werten (soweit ich das sehe) leider nur über Script möglich, genauere Beschreibung s. [hier (unter 3.4, 16.)](https://www.fmod.com/docs/2.02/unity/integration-tutorial.html#adding-sounds-kart-engine) (relevante Syntax: „GetComponent<FMODUnity.StudioEventEmitter>()“ & Befehl „.SetParameter*(„FMOD-Parameter-Name“*, value)
  + **Konkretes Test-Beispiel**:
    - Parameter mit Namen „TestParameter“ verknüpft mit TestAtmo-2D
    - „TestParameter“ hat Werte zwischen 0 und 3, Initial Value 1
    - Effekt der Parameter: Bereich 1 – 2: Klang normal; Bereich 0 – 1: Klang dumpf („Unterwasser“); Bereich 2 – 3: keine tiefes Grundrauschen mehr
    - Außerdem: „TestAtmo-2D“ läuft im Loop -> müsste ewig weiterlaufen bzw. erst über den FMOD Studio Event Emitter – Stop Event Command händisch gestoppt werden
* Steuerbare Audio-Busse (= **Lautstärke-Regler**) für Spieler erstellt:
  + Einbindung in (eventuelles) Menü, damit Spieler Sound-Einstellungen vornehmen kann *[nicht schlimm, wenn erstmal nicht in Spiel eingebunden, war nur relevant für Setup der Session]*
  + Steuerbar über folgende (globale) Parameter:
    - [Parameter-Name] – [Beschreibung]
    - „UserVolume-Master“ – Gesamtlautstärke
    - „UserVolume-Ambience“ – Atmo-Lautstärke (z.B. Stadt, Wald, …)
    - „UserVolume-Voice“ – Stimmen-Lautstärke (falls evtl. kleine Voice-Lines von Charactern)
    - „UserVolume-Music“ – Musik-Lautstärke
    - „UserVolume-SoundFX“ – Soundeffekte-Lautstärke
  + Einbindung in Unity (vermutlich) über Script am Lautstärke-Menü oder Component „FMOD Studio Global Parameter Trigger“ – das weißt du aber wahrscheinlich viel besser