CSR2 Hub Streaming Data Documentation and Update Instructions

Table of Content

CarVisualData (CVD)

This section of the project is covering car specs by providing game assets that contain gacha config files for each car. The assets in the project folder are renamed to only be the cars crdb code name. When extracting these assets from the games UnityCache folder you need to look out for folders ending in “_common.ASTC”. If inside that folder multiple subfolders appear, then take last folder from the list as it’s the latest version of that asset. Rename the __data file to the crdb code name of the car before moving the file to the CVD folder of the project.

Gacha

This is the second part to the car specs system. In this folder we provide in game screenshots of each cars specs with a fixed angle. To generate these screenshots, you need a device or emulator rendering the game at 1920x1080. To have all specs of a car reliably you want to read the CVD file of the car with an asset bundle browser like AssetStudioGUI an asset bundle browser like AssetStudioGUI. In there you can find a MonoBehaviour named CarVisualsData. There under GachaConfig you can find each spec of a car. Switching to dump in AssetStudioGUI you can see the size of the section. Using Car support gifts you can generate the each spec by inserting the GachaConfig ID into gachaconfig in the support gift. To generate all specs you need support gifts with gachaconfig ranging from 0 to section size - 1. For example a car has 15 specs then 15 support gifts are needed with gacha config ranging from 0 to 14.

Icons

Icons of cars and brands can be found in UnityCache aswell and have the folder ending “_icon.ASTC”. If inside that folder multiple subfolders appear, then take last folder from the list as it’s the latest version of that asset. Open the Asset folder in an asset bundle browser like AssetStudioGUI. There will likely only be 1 asset inside being a Texture2D. Extract the asset as a png. The png file should automatically be named after the car crdb or brand name + “_icon”. Place the image in its respective folder with that naming scheme. Collections likely won’t get any more additions as the feature was never developed further.

Libraries

Libraries can be found in UnityCache. Get the __data file from the last subfolder inside a library folder and move it to the repo folder. Files from inside the NmgSvcs folder are gathered from Documents/NmgSvcsData/1/ ConfigData. OldCarsInThisAppVersionList.json is always ment to be the NewCarsInThisAppVersionList.json you replace with the latest one.

Snapshots

Snapshots are used to preview a car in the app. We use 2 types of snapshots. One being the normal one from the garage browser while the other is only generated by the game in Collections and support gifts. The file locations are CarSnapshots/<player_id>/Default and CarSnapshots/<player_id> respectively. The png files need to be renamed to snap and snap_full respectively. To save them properly in the repo move them to the cars brand folder and inside a subfolder named after the cars crdb code name.

TextData

TextData is a folder you will likely never have to touch as all its content is automatically uploaded and maintained through Google Sheets Appscripts.

aCRDB

Just like the TextData folder this file is automatically updated and you don’t need to touch it.