

CV2 Combat

Design: [REDACTED]

Engineering: [REDACTED]

*"cv2 made scoreboard cv2 made teams cv2 made health and damage
cv2 made rules cv2 made hud/ui cv2 made EVERYTHING" ~ From RR CV2 Discord*

CV2 Combat is an effort consisting of smaller projects that enable creators the ability to create their own compelling PvP experiences without requiring the CV1 Game Rules Chip.



It is **Possible** and it is **Good**

In the context of CV2 PvP, **possible** means that players can achieve the goal of creating their own PvP systems without CV1. **Good** is about providing gameplay feedback that is at least as effective as what is available in the Game Rules chip so that capable creators will not understandably default back to our old system.

Examples:

- *"I can script my own health system into my game (possible) and the players know what their health is (good)."*
- *"A player has been shot and takes damage (possible) so their screen flashes red (good)"*

Goals

1. Give creators the foundation to begin creating PvP game rules on their own
 - a. It is **Possible** to recreate Paintball in its entirety.
2. Give creators tools to provide **good** player feedback for combat.
 - a. Players understand the state of the game.
 - b. Players understand their own state.
 - c. Players understand other players' states.
3. Give creators inventions to improve CV2 PvP adoption.
 - a. Paintball Game Manager (Creative Team certified invention)
 - i. Rules
 - ii. Team Manager
 - iii. Player Health
4. The network reliability of the Projectile Hit event must be up to par with RROs.
5. Expand the utility of the Player Definition Board
6. Adhere to Operation: Clean Break principles as much as possible
7. Don't break any of our CV1 PvP rooms

Non-Goal

- Overtake the Game Rules chip immediately.
 - The early release of this will be prohibitively complicated to Tracy outside of the certified invention for paintball. The goal is to expand the possibility space for PvP rooms first.

CV2 Combat Rev 1 Deliverables

Required for MVP

1. Projectile Hit Event
2. CV2 Equipment
3. CV2 Player HUD

4. CV2 Player World GUI
5. CV2 Damage Flash Color Chip
6. Respawn Chip (On its way!)

Post-Release

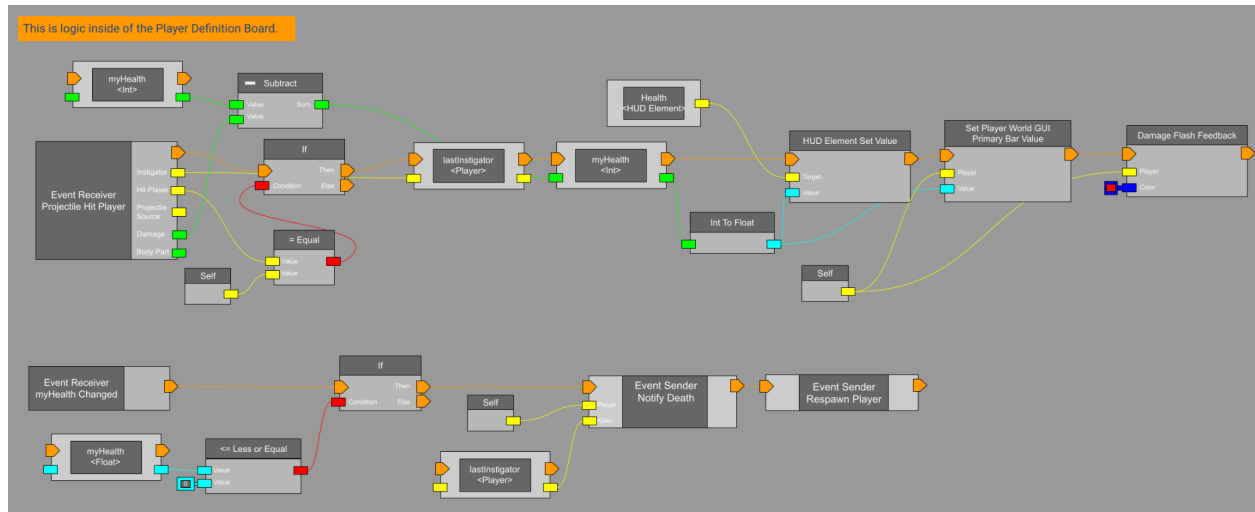
Along with the items in the following list, Operation: Clean Break outlines an audit of many different mechanics whose porting should be prioritized based on community response to the MVP.

1. Color Grade Chip (for grayscale when knocked out effect + more flexible uses)
2. CV2 Teams Invention
3. CV2 Health/Damage Manager Invention
4. CV2 Paintball Game Manager Invention (probably contains the previous 2 inventions)
5. Projectile Spawner Gadget

Other Projects That Matter

- CV2 Color type
 - Prerequisite for Player HUD and Player World GUI chip color inputs
 - *Recolor Prop Chip* from in the Color Type document can recolor relevant props like weapons to team colors.
- CV2 Roles
 - Inventions will require role definitions. As of this writing, CV2 game manager inventions would have to contain CV1 Role Chips.
- Switch Chips
 - QoL improvement that will work well with Projectile Hit Player Event.
- Object Model Milestone 1
 - Required to have a hypothetical projectile spawner gadget be part of an interesting weapon that can return an instigating Player without hardcoded hackery.

Projectile Hit Player Event



Ready Fire Aim (MVP)

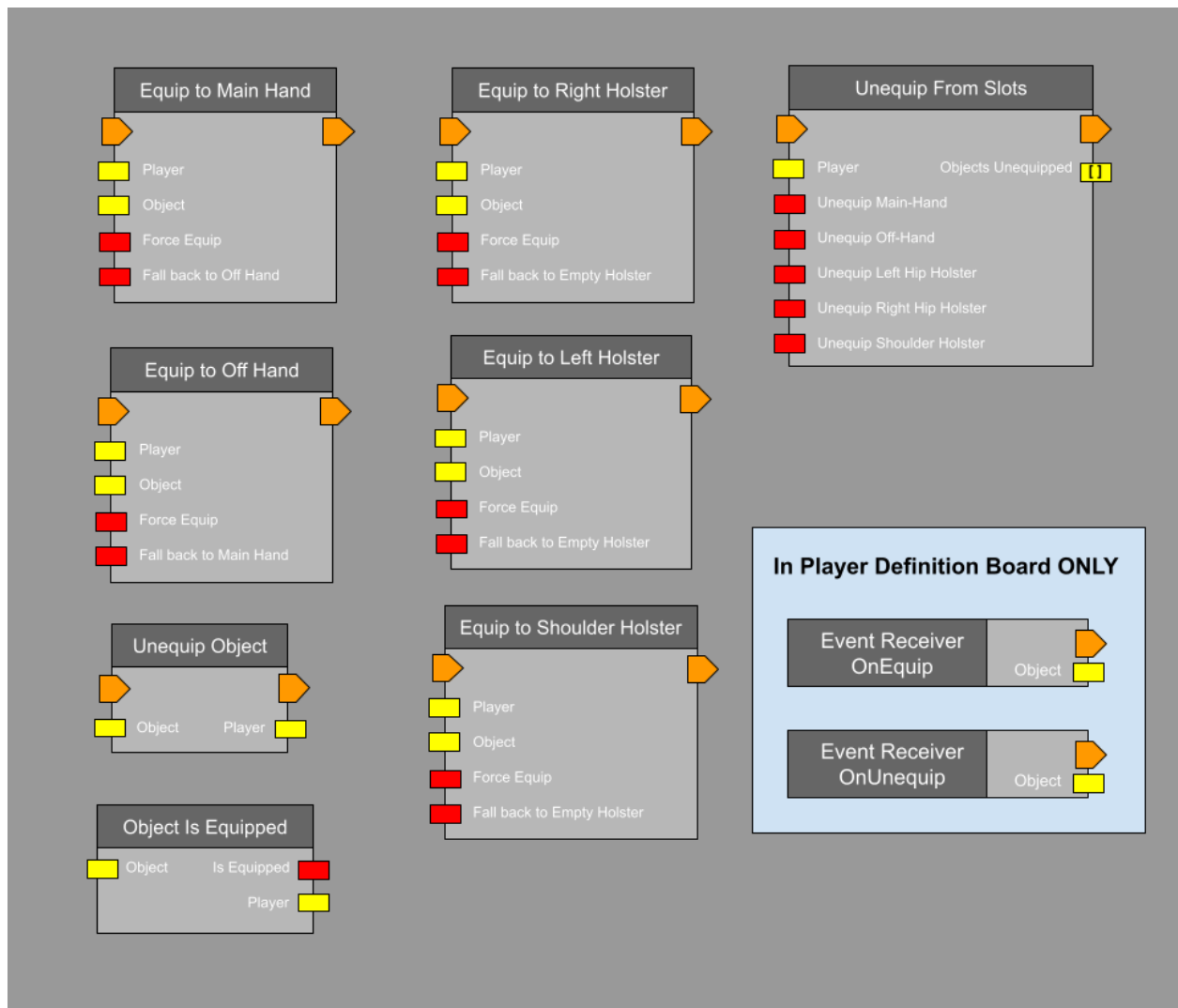
	<p>Instigator <Player>: player who spawned the projectile</p> <p>Hit Player <Player>: player that was hit by the projectile</p> <p>Projectile Source <RRObj>: Whichever object spawned the projectile</p> <p>Damage <Int>: Damage as defined by the spawner (C# for RRO weapons, projectile spawner config for UGC weapons)</p> <p>Body Part <int>: 0 = head; 1 = torso; 2 = right hand; 3 = left hand. Information must be in the chip description. Plays well with Switch Chips</p>
--	--

A projectile here is defined as any object that inherits from *Bullet.cs*. In the *OnPlayerHit* event we can return the weapon as *<RecRoom Object>* and the Instigator as *<Player>*.

Repeat (wishlist)

- Create a UGC projectile spawner. On hit, *Projectile Source* is the object that the spawner is a child of. (requires OM)
- Return a 'Hit' struct that could give us information such as distance travelled, impact normals, impact point, etc.

CV2 Equipment



Notes:

- OnEquip/OnUnequip events only exist in the Player Definition Board.
- Using these chips should work reliably over network
 - My local player should be able to call this on others
- If an object is not equippable (*can be picked up*), send a warning to the log.

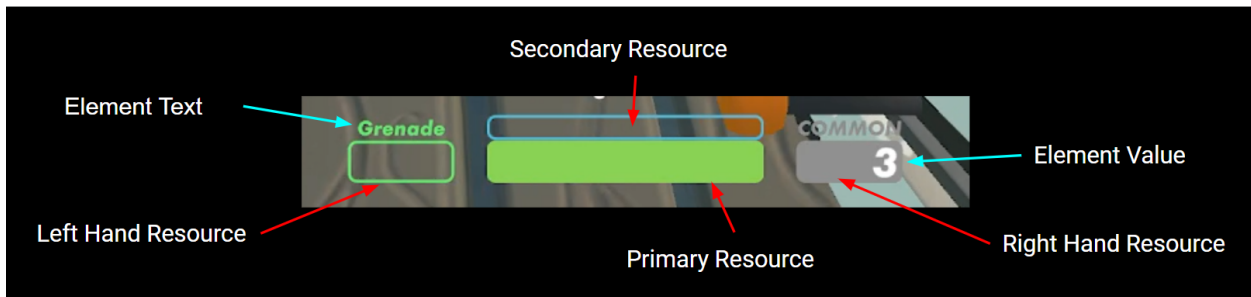
- Unequipping an item causes it to drop in place.
- These play well with existing chips *PlayerGetEquippedObjects* and *GetPlayerLastEquippedObject*
- If force equip on, then you can equip a weapon held/holstered by someone else to someone else
- “Fall back to Empty Holster” boolean falls back in a predictable order:
 - *Right-Hand -> Left-Hand -> Shoulder -> Right-Hand*

Combat Feedback Parity

It is important for creators to have access to some of the feedback that we created for RROs. Without HUD, Health Bars, and damage received feedback it would be hard to justify a creator opting for CV2 Combat over the old Game Rules chip. We do not and will not have tools available to creators to create their own feedback any time soon so we should expose some of the functionality of our first-class x-plat solutions to CV2.

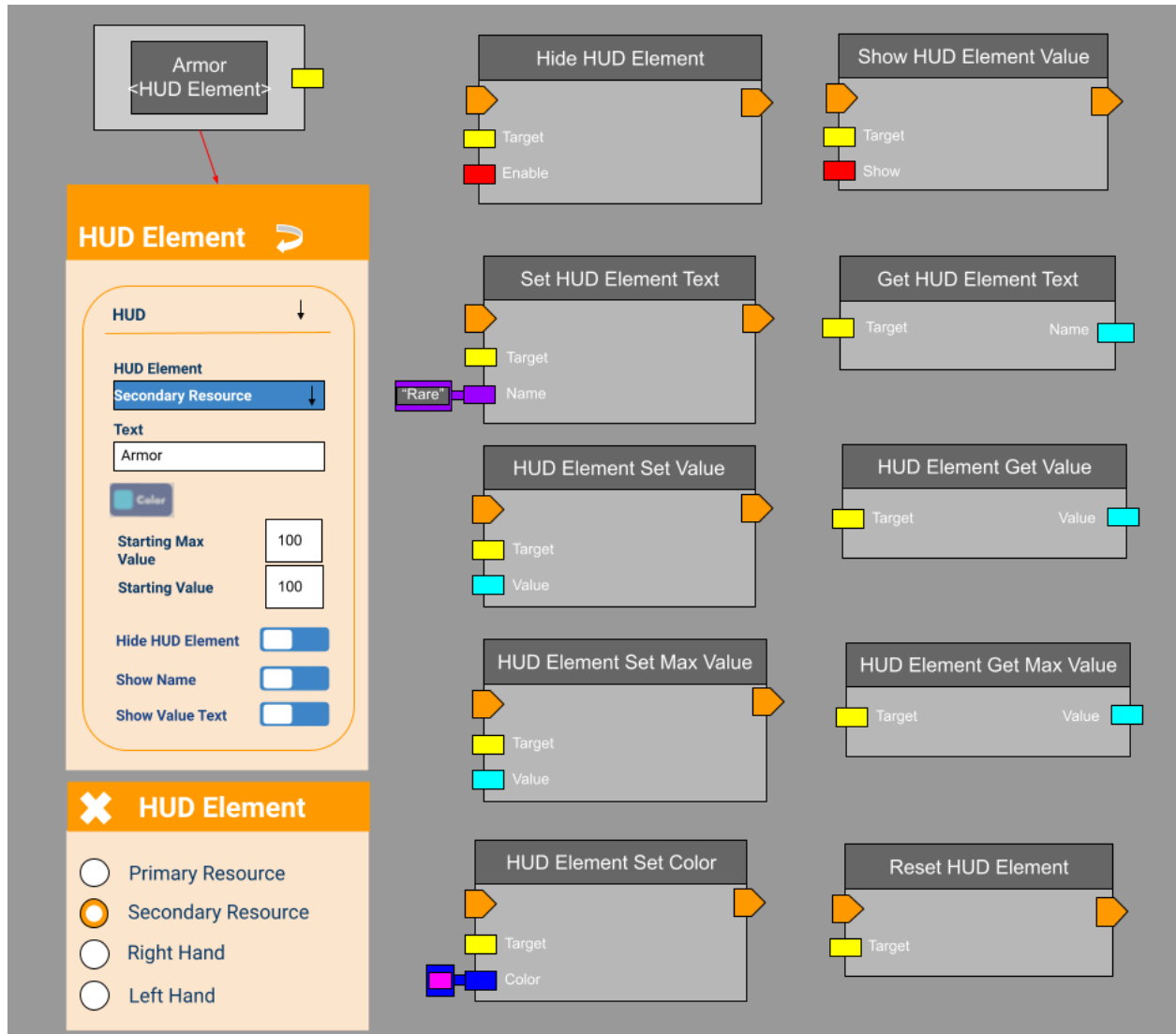
Player HUD

Expanding the player HUD provides greater breadth of expression to creators outside of PvP scenarios and makes a case for choosing CV2 for feedback over the existing Game Rules chip.



General Notes:

- For the interest of a consistent x-plat experience, the size and location of these elements cannot be altered by a creator.
- HUD Element *text* and *value* have strict character limits.

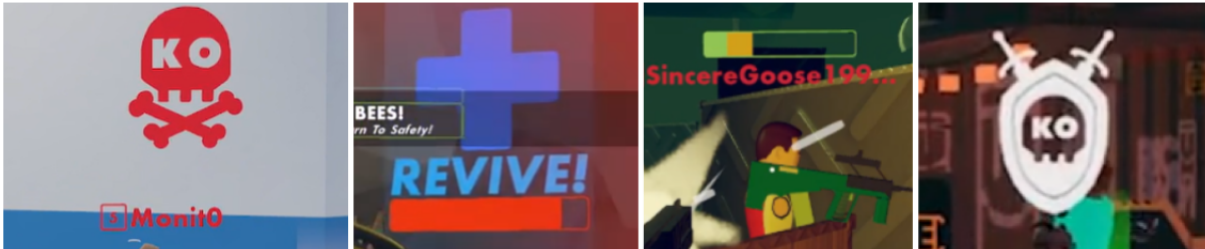


CV2 Chip Notes:

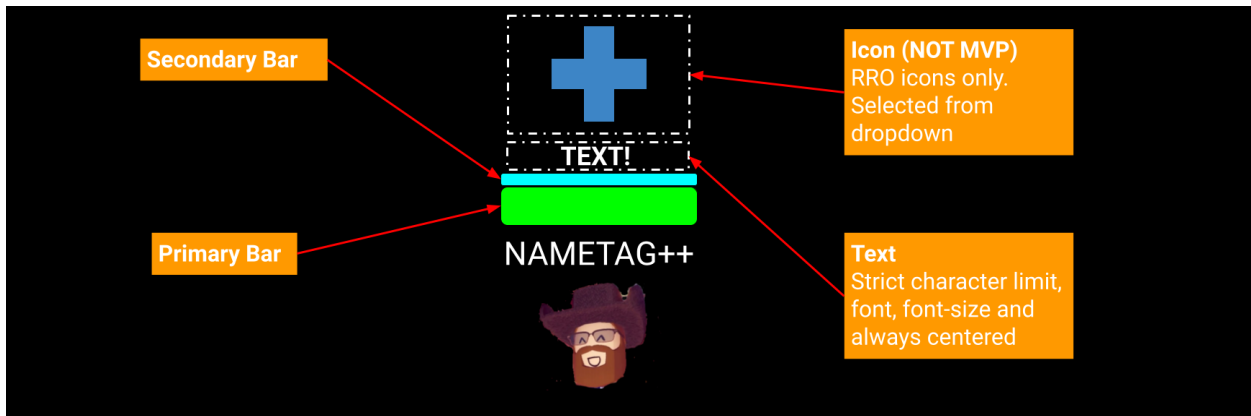
- All HUD chips are local only.
- If an object has predefined HUD logic from RROs, overwrite any player CV2 HUD logic with the proprietary stuff.
 - **Example:** If a *Right* or *Left Hand* HUD element exists, picking up a paintball or laser gun will overwrite whatever was there previously with their ammo.
 - Dropping the weapon/object will restore the previous state of the HUD element.
- *Reset HUD Element* restores Home Values.
- Only one instance of a HUD element can be active in a room.

- If there already exists a Primary Resource HUD Element, the next spawned HUD Element constant chip would default as the next item in the list. In this case: Secondary Resource.
- If a HUD element of a certain type already exists, gray out that option from the drop down unless the pre-existing constant is deleted.
- A fifth HUD element cannot be spawned. Send warning to log if attempted.

Player World GUI

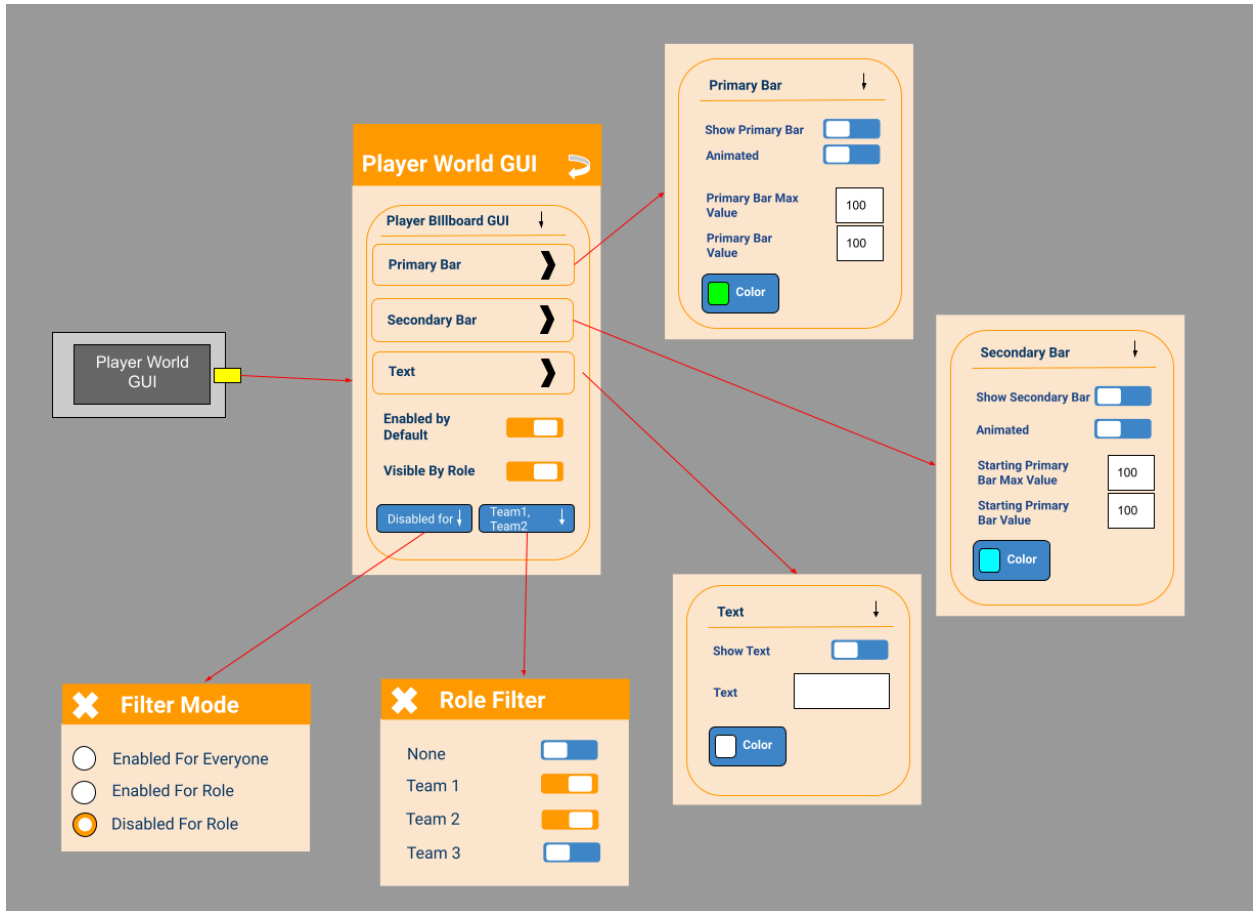


Player World GUI elements are used to provide feedback to other players about the local player's state. Like with the HUD, expanding the player GUI provides greater breadth of expression to creators and makes a case for choosing CV2 for feedback over the existing Game Rules chip.




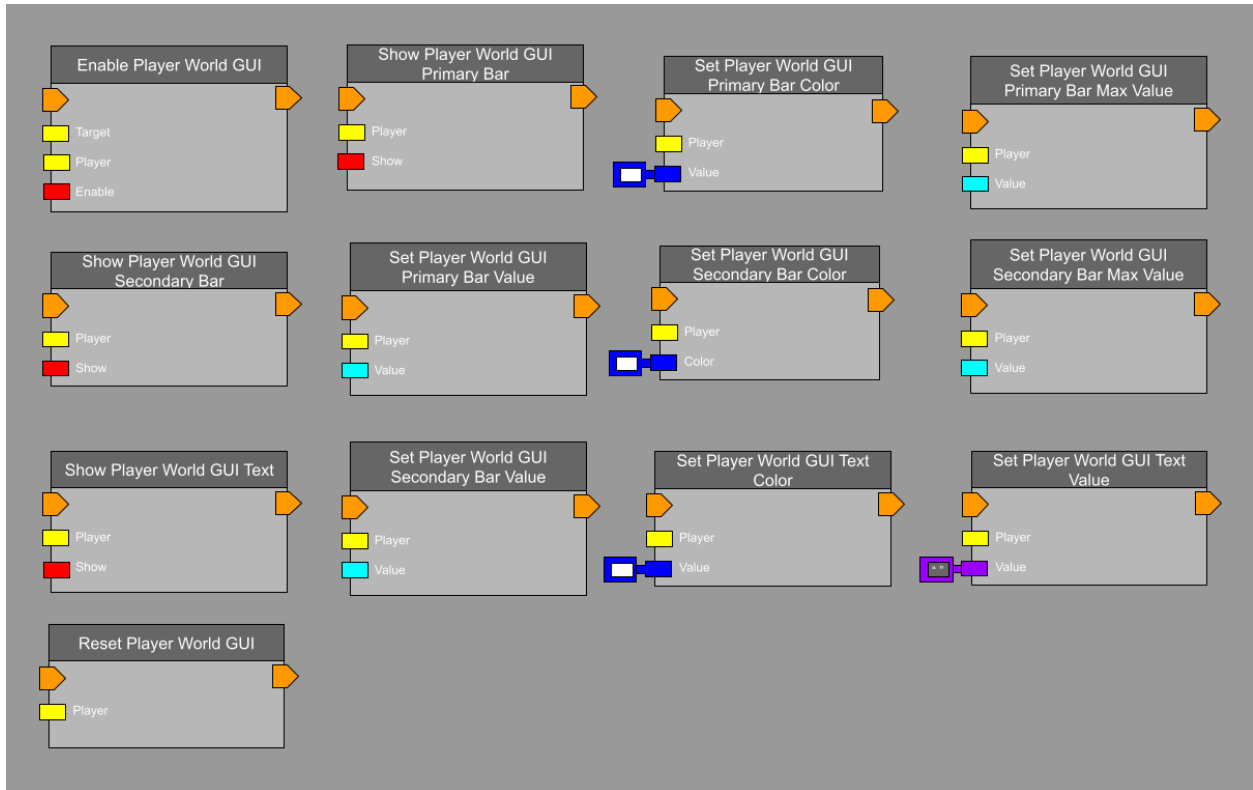
General Notes:

- Player World GUI is always synced.
- They are always above the player's name tag even if the name tag is hidden.
- Each individual player's World GUI element values should not be making RPC calls. All players' elements should be managed by a singular manager to bundle these updates together.



Player World GUI Constant Notes:

- Only one Player World GUI instance can be enabled for a player at any time.
- The CV2 Player World GUI chips will only work if a Player World GUI constant has been enabled through its config menu or *Enable Player World GUI* chip.
- “Animated” toggle refers to when the value is decreased, it shows a preview of the damage before shortening the bar:
 - 
- The role filter logic and UI should map 1:1 with Interaction Roles
 - Interaction Roles will most likely be shipped before this work starts, so better to add detail here later.



CV2 Chip Notes:

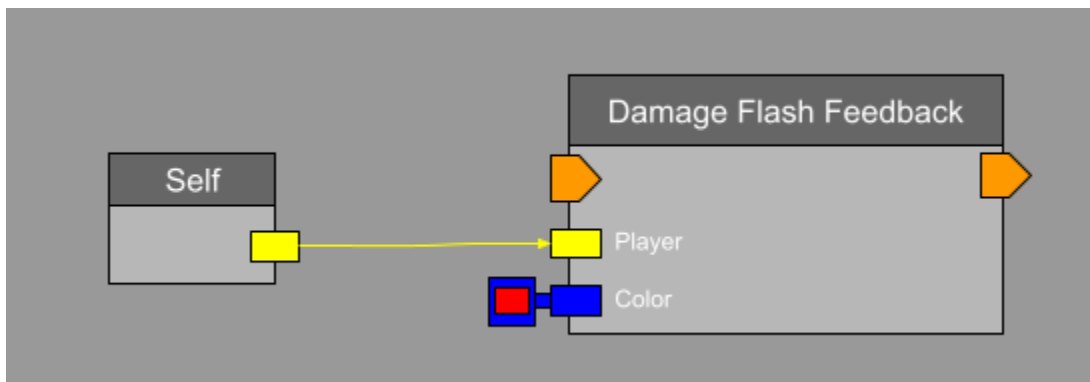
- Attempting to call Player World GUI logic on a player that does not have a GUI enabled will send an error to the log.
- Attempting to enable a new GUI with the *Enable Player World GUI* chip will send an error if the player currently already has a World GUI enabled.
- *Reset Player World GUI* resets the target player's active World GUI to its home values.

Additional Notes:

- A future generic "World GUI" gadget would be great to apply to any object in the game.
 - We could limit its use with ink cost
 - Requires OM
 - Does not affect **Player** World GUI

Damage Flash Feedback

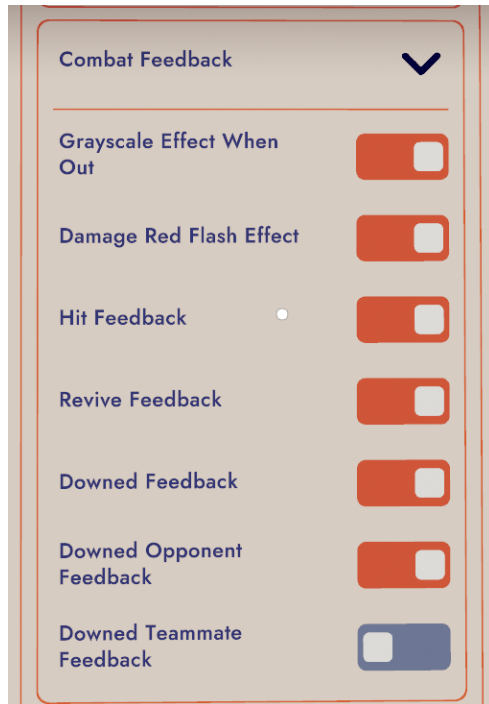
Damage Flash Red is an option in the Game Rules chip under “Combat Feedback”. It is a pre-existing x-plat solution that has already been solved. It can already handle rapid-fire weapons without the need to look further into trust and safety for screen flashing.



Notes:

- Can send flash feedback over network
- Will be most reliable if called on local player
 - After getting Projectile Hit Event, for example.

Combat Feedback We Are Not Doing for MVP



Besides *Damage Red Flash* and *Hit Feedback*, the rest of these are specific to Rec Royale and don't necessarily achieve our Goal of "creators can make paintball" and are likely to create more debt than help to port them in some fashion.

Notes to self *ignore if not [REDACTED]*

- Must do CV2 Roles
- pickups event + chips (buffs including the ones already in palette)
- explosives event + chips (aoe damage system for grenade types)