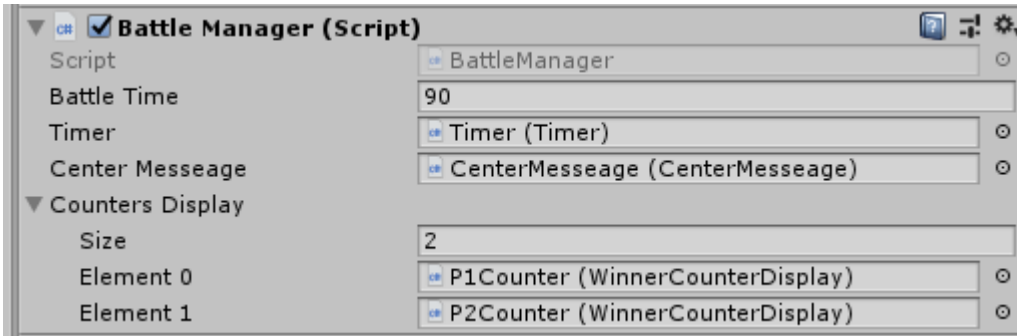


システム設定項目

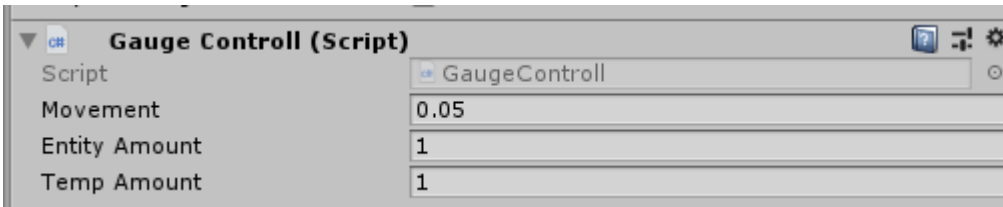
root/BattleManager



- ・ BattkeTime 対戦時間

UI への参照は面倒なのでシリアライズ使ってます。今回使ったポーズの仕様上 Canvas が root 直下に置けません。CounterDisplay は PlayerNum で識別できるので順番は適当でいい

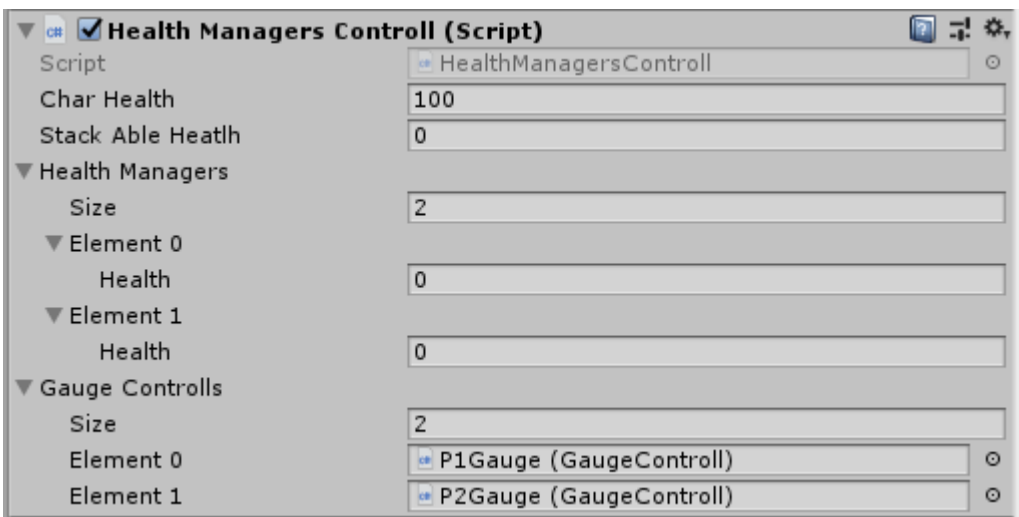
Canvas/Top/P{n}gauge/GaugeControll



- ・ Movement fillamount の 1 フレーム辺りの変化量

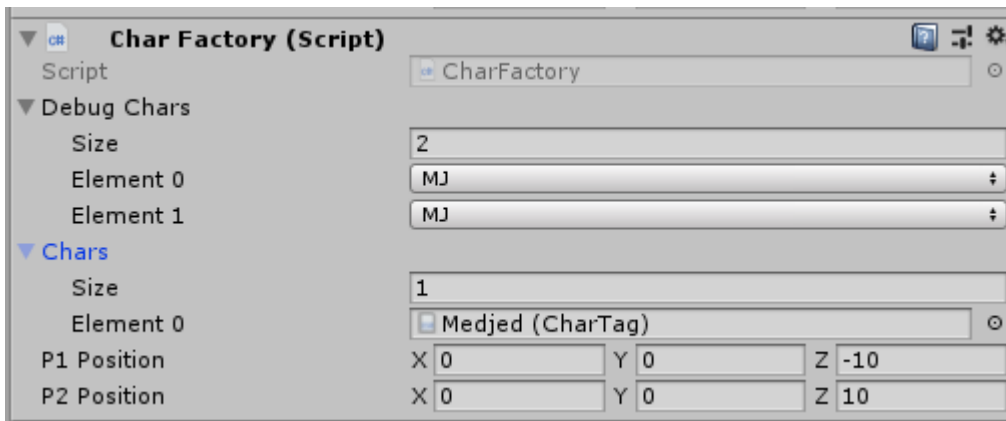
色見れば分かるけど temp が仮ゲージ。temp とそれ以外の Image を取得するので本物の方は名前適当で可

root/HealthManager/HealthManager



- ・ CharHealth キャラの体力
- ・ StackAbleHealth 2D フェイズから 3D に戻るために必要な閾値
- ・ HelathMangers 動的生成されま
- ・ GaugeControlls これはインデックス 0 が P1、1 が P2 なんて順番気を付けて

root/CharFactory/CharFactory



- ・ DebugChars GameStateManager がない時はこのタグを使ってキャラを生成する
- ・ Chars Prefab への参照。CharTag クラスでシリアライズしてるけどオブジェクトをドロップして大丈夫
- ・ P{ n }Position 初期座標

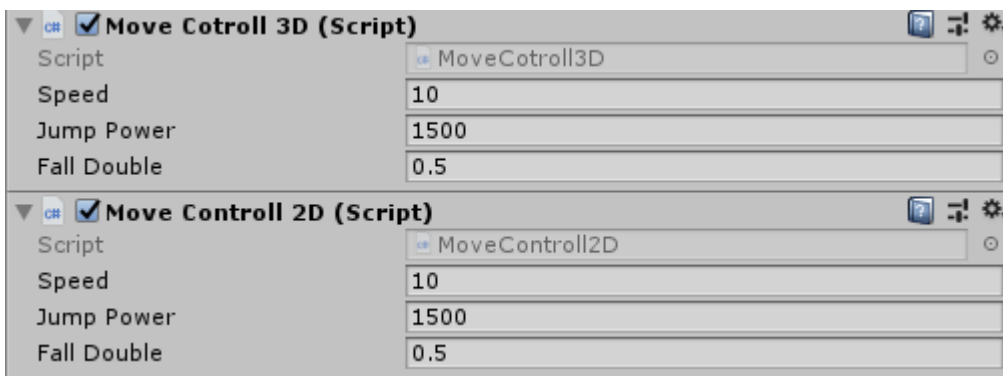
キャラ設定項目

[キャラの root]

PlayerRoot

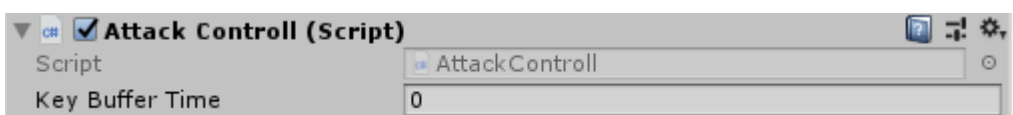
必須コンポーネント全部 RequireComponent してあるからまずこれアタッチして

MoveControll



- ・ speed 移動速度
- ・ jumpPower ジャンプ力
- ・ fallDouble 落下時にジャンプ力を下向きにかける際の倍率
両方共同じ数値でいいと思いますまる

AttackControll



- ・ keyBufferTime 複数入力用にキーがバッファを保持する時間。但し各コマンド毎にこの項目は設定できるのでこれは攻撃に関係ない移動キーなどにのみ適応される数値。

DamageControll

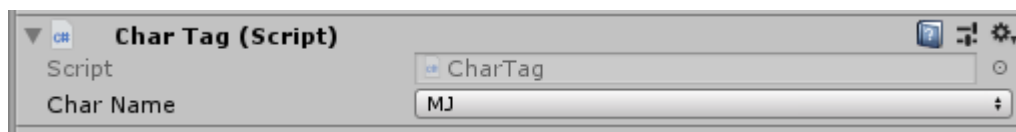


- ・BlowPowerBorder どれくらいのノックバックを受けると強被ダメージモーションになるかどうか
- ・ReductionRate 強攻撃をガードした時の軽減率

GuardControll

こいつはあくまでガードのモーションとステートを操作するだけでダメージとかには一切関与しない

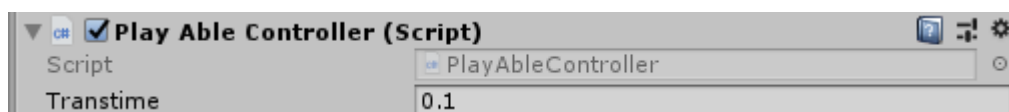
CharTag



ファクトリから参照するために CharName 保持するだけ。

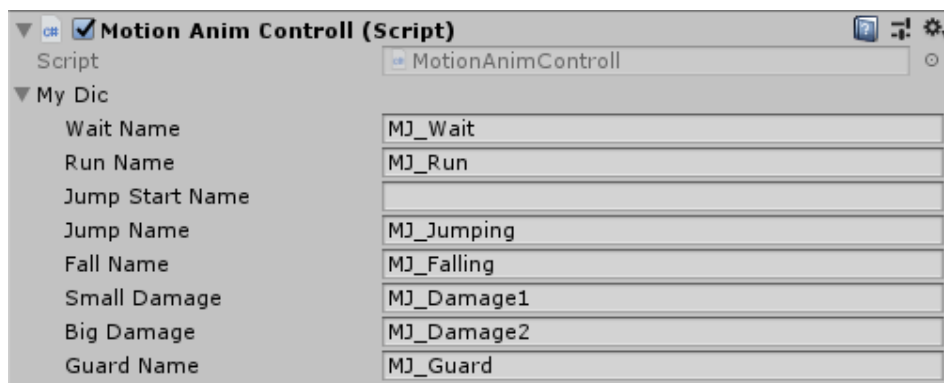
[ボーンの root]

PlayAbleController



- ・transTime モーションブレンドにかかる時間

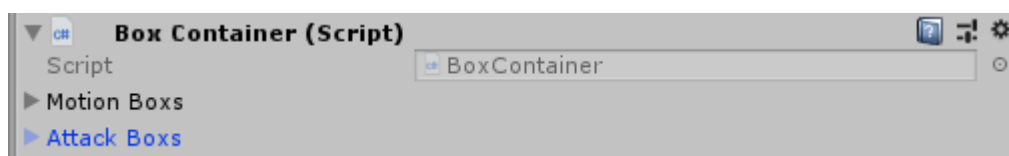
MotionAnimControll



- ・MyDic

攻撃以外のモーションはここで AnimationClip の名前を登録する。JumpStart だけではなくても動くが他は必須

BoxContainer



モーション全部保持します以上

[AttackBox]

- ・clip クリップ
- ・delayTime モーションが終わってもこの時間中は硬直し、コンボの受付ができる
- ・BufferTime キーがバッファされる時間。この時間中に複数入力 of モーションが入力された場合はこのモーションがキャンセルされて入力されたモーションが実行される
- ・AttackInputInfo/KeyCodes 実行に必要なキー情報。現状 2 入力までしか対応していない
- ・AttackInputInfo/CommandType {0:Normal,1:Jump,2:Chain}Jump はジャンプ中のみ実行される。Chain は単独では実行されずコントローラー側からのみ実行される（掴み攻撃など連続モーションに用いる）
- ・AttackInputInfo/ApplyPhase {0:Both,1:P2D,2:P3D}そのモーションが実行されるフェイズ。もし同じキーでこの項目も同じモーションがあるとインデックスが小さい方が実行されると思います
- ・DelayTimeForTools 入力があってから AttackTool が起動するまでの時間。当たり判定の発生遅らせたり
- ・AttackTools 攻撃に使用する AttackTool 全部。0 でもエラーとかは起きない
- ・AttackDamageBox/Damage ダメージ
- ・AttackDamageBox/AttackType {0:Weak,1:Strong,2:Finish,3:Range,4:Grab,5:Empty}弱攻撃,強攻撃,コンボフィニッシュ,遠距離攻撃,掴み攻撃。現状遠距離は特殊な処理がないので遠距離攻撃は Weak or Strong（ガードに対しての処理が変わる）。掴み攻撃は初動と投げ両方共 Grab を設定してください
- ・AttackDamageBox/KnockBackPower ノックバック量。ヒットした対象の方向ベクトルの反対を基準に発生する
- ・NextAttack コンボ、及び派生攻撃で使用する。ここから Enum のプルダウンが使えなくなるので Enum の項目には上記の対応する数字を入れてください

[AttackTools]

AttackCollider

起動するとコライダがアクティブになるだけのシンプルツール。コライダは Trigger をお願いします

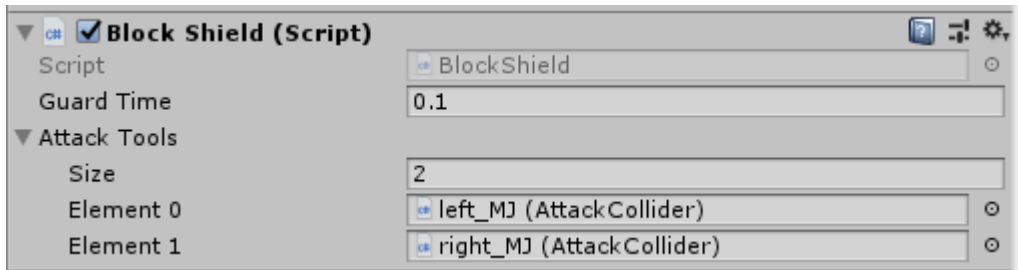
BulletFactory

設定項目はｼﾗｽ。自身の座標基準にするから空の Object にアタッチしてボーン以下に置くのがよさげ

Grab

初動の当たり判定、及び投げ動作の両方を行う。また投げは自身の座標を使用するので、Grab 用の Object をキャラの前に配置し、投げを行うオブジェクトの下に置き、コライダだけそっちに合わせるという特殊な構造が必要になる（メジエドだと右腕に置いてある

BlockShield



- ・ GuardTime ガード状態を維持する時間。
- ・ AttackTools ガード状態が終わった後起動するツール

ガード攻撃用。GuardControll にアクセスするのでキャラの Root にアタッチしてください

AttackSoundPlayer

起動時に音鳴らすだけ。適当にアタッチして鳴らす攻撃の AttackTools に登録するだけ

AttackTool 自体は攻撃毎の攻撃開始/終了処理を委譲するためのものなので AttackSoundPlayer みたいに見栄えのために継承して使ってもよし。エフェクト出すのも IAttackTool 継承したクラス使う方が楽かも

当たり判定である HitStream は AttackToolEntity は Start で登録されるので動的生成しないでください。ショットなどは AttackToolEntity で弾などを動的生成しその当たり判定を自身の HitStream に流すみたいな形での実装が推奨