Bowling starter kit

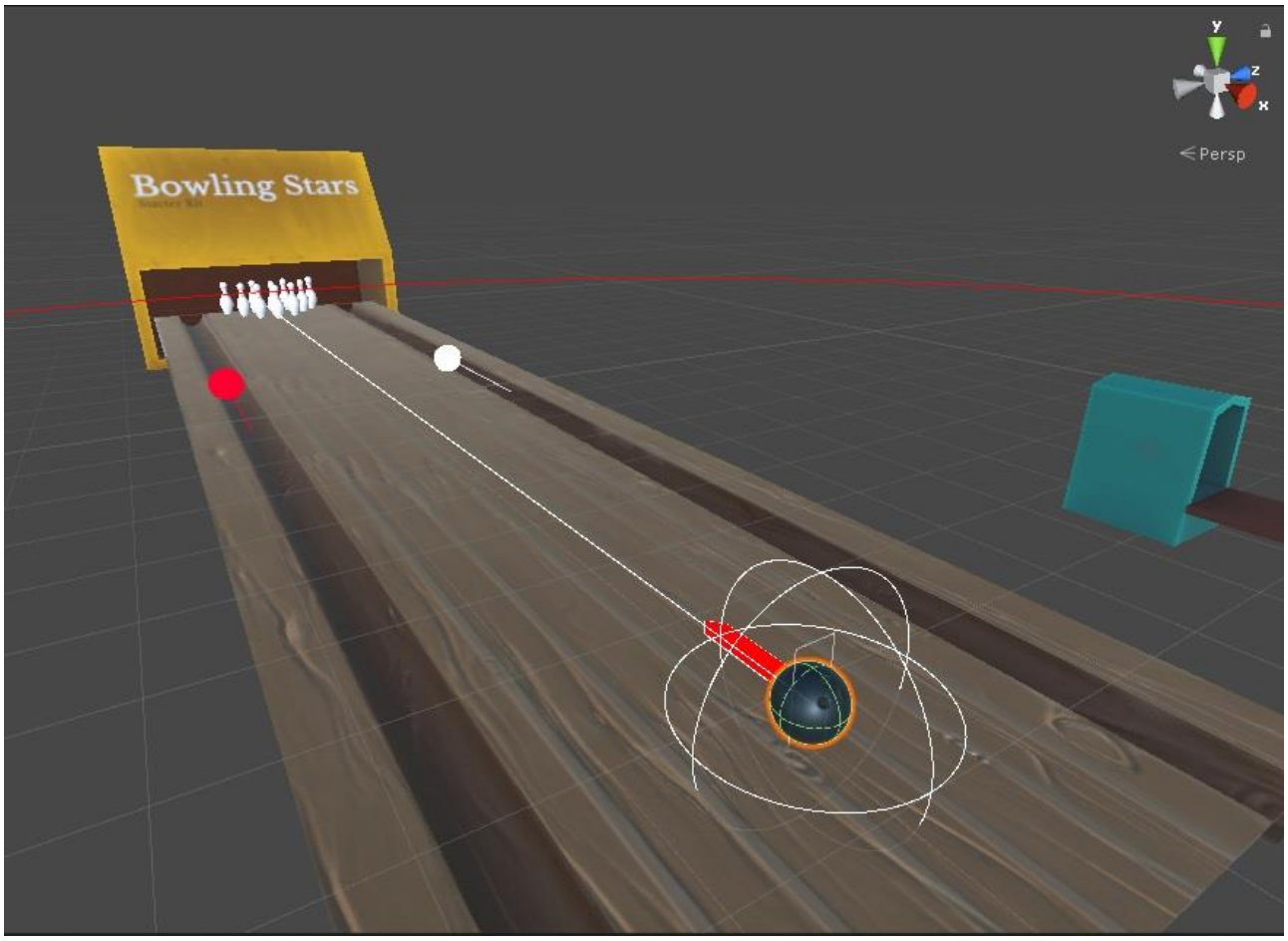
This system will allow you to create simple, cartoony bowling system. It includes additional physic functions to enhance ball controlling and managing pin(s). this system will control bowling game events include developer events.

There you have all scripts to work:

- BowlingGameManager
- BowlingGameManager Editor
- BowlingBall
- BowlingBallEditor
- BowlingPin
- BowlingPinEditor
- AcceleratorSensor
- BallHolder
- BowlingObjectStopper
- -
- PivotAngleHandler
- MainMenu
- MainMenuItem
- BowlingMenu
- -
- EditorTools
- Extensions
- Globals

How it works?

Ball force, UI, directions, physics, hole system will calculate by system. The potential force will change by moving touch position or by player tapping moments. After release, the potential force converted to dynamic force and the logics will calculate.

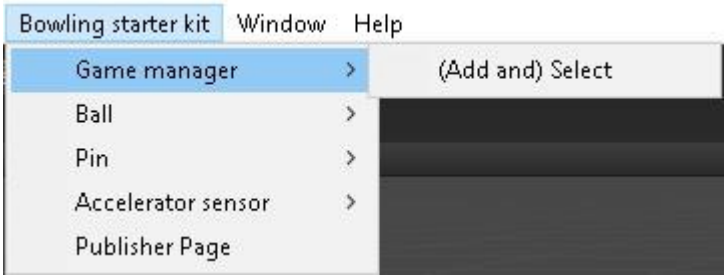


How to use menu item?

Here the menu items explained quickly but in Scripts title its explained with more details.

Game Manager

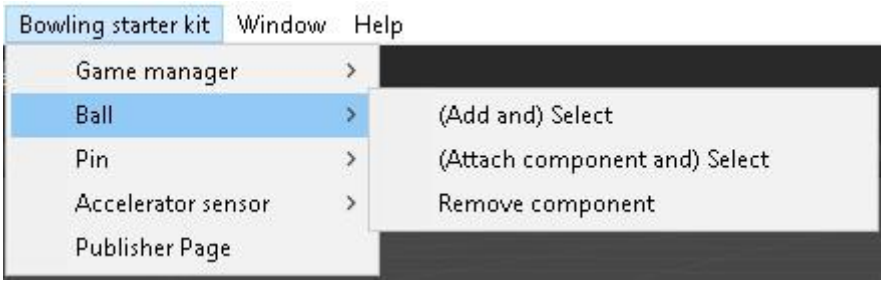
Game Manager is the core for handling game logic.



In “(Add and) Select” item you can select or add game manager script if not exists.

Ball

Ball is the core for bowling ball.



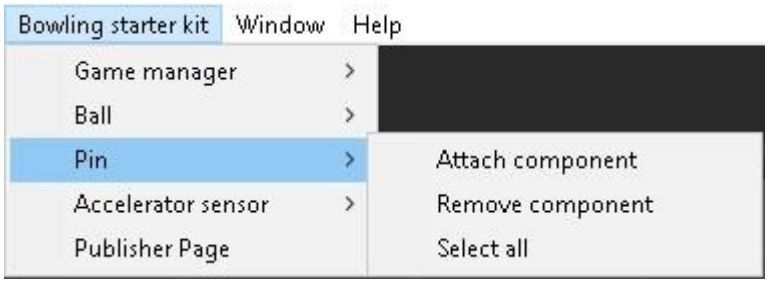
In “(Add and) Select” item you can select or create ball script if not exists.

In “(Add and) Select” item you can select or add ball script if not exists.

In “Remove component” item you can remove ball script component from all game object(s) that you selected.

Pin

Pin is the core for bowling pin.



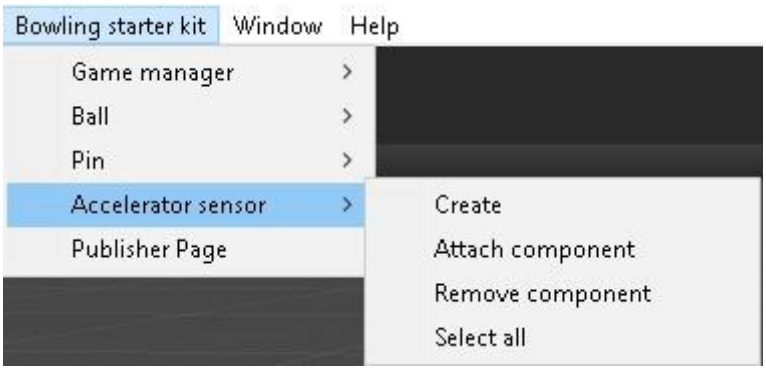
In “Attach component” item you can attach bowling pin script to selected object(s).

In “Select all” item you can select all object(s) that included bowling pin script.

In “Remove component” item you can remove bowling pin script component from all game object(s) that you selected.

Accelerator sensor

Accelerator sensor is useful for add some physic force to game.



In “Create” item you can create new accelerator sensor.

In “Attach component” item you can attach accelerator sensor script to selected object(s).

In “Select all” item you can select all object(s) that included Accelerator sensor script.

In “Remove component” item you can remove accelerator sensor component from all game object(s) that you selected.

Basic scripts and their parameters

BowlingGameManager

It`s the main script to create your game logic. The game manager script will get you complete information for handling game logic.

Parameter(s)	
Start from ball holder	In game start, bowling ball loaded from ball holder
Real-time pin state updating	Each pin state(s) will be updated in real time. It's useful for debugging and better to check this state(s) after game over.
Infinities shoot(s)	Allow player to have infinities shoots
Max shoots	If player don't be allowed to have infinities shoots you can defend maximum player shoots count
Overriding grand mask	Override ground mask for all pin(s)
Ground mask	Ground layer mask
Event(s)	
Game is over delay	Delay for invoke in second
Game is over event	Game over event
Game firs start event delay	Delay for invoke in second
Game firs start event	Game firs start event. It's just has one invoke in starting the game
Ball loaded from ball holder delay	Delay for invoke in second
Ball loaded from ball holder event	This event will be invoked when ball loaded from ball holder
Delay for Shoot(s) over	Delay for invoke in second
Shoot(s) over	Game over event when player shoots are over
All pins fallen delay	Delay for invoke in second
All pins fallen event	This event just invoke when all pins fallen
Information	
Game is over	Boolean value to explain game is over or not
Current shoot(s)	Player shoots count
Game state	Bowling game state
Fallen pin(s) count	Fallen pin(s) count
Standed pin(s) count	Standed pin(s) count

Note: the game is over when:

1. All pin(s) are fallen
2. Or The player shoots are over

bowling **Game states** are:

- Null: not a valid game state
- None: there is no special state founded yet
- Shoots over: the game is over because player shoots are over
- All pins fallen: all pin(s) are fallen

This script has some useful functions that help game designer to develop:

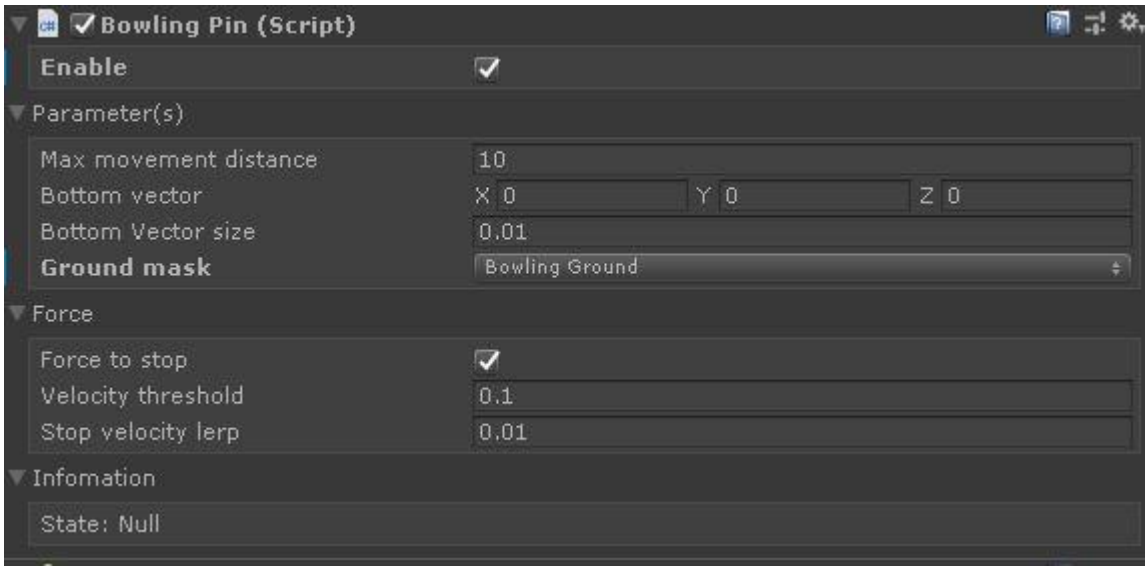
Parameters	
`GameIsOver`	`Read-only Boolean value for game is over or not`
`GameState`	`Read-only game state`
`CurrentShoots`	`Read-only player current shoots count`
`StandedPinCount`	`Read-only standed pin(s) count`
`FallenPinCount`	`Read-only fallen pin(s) count`
Functions	
`ResetToStart()`	`Reset the game manager`
`TakeBallToStart()`	`Reset the ball and transferring it to ball holder`
`pinStateIsExist(state)`	`Return Boolean of atlases one pin equal to state`
`pinsStateCount(state)`	`Return count of pin(s) are equals to state`
`pinsStateEquals(state)`	`Return Boolean of all pin(s) are equals to state`

BowlingGameManagerEditor

Editor for BowlingGameManager script class.

BowlingPin

It`s the main script to create bowling pin.

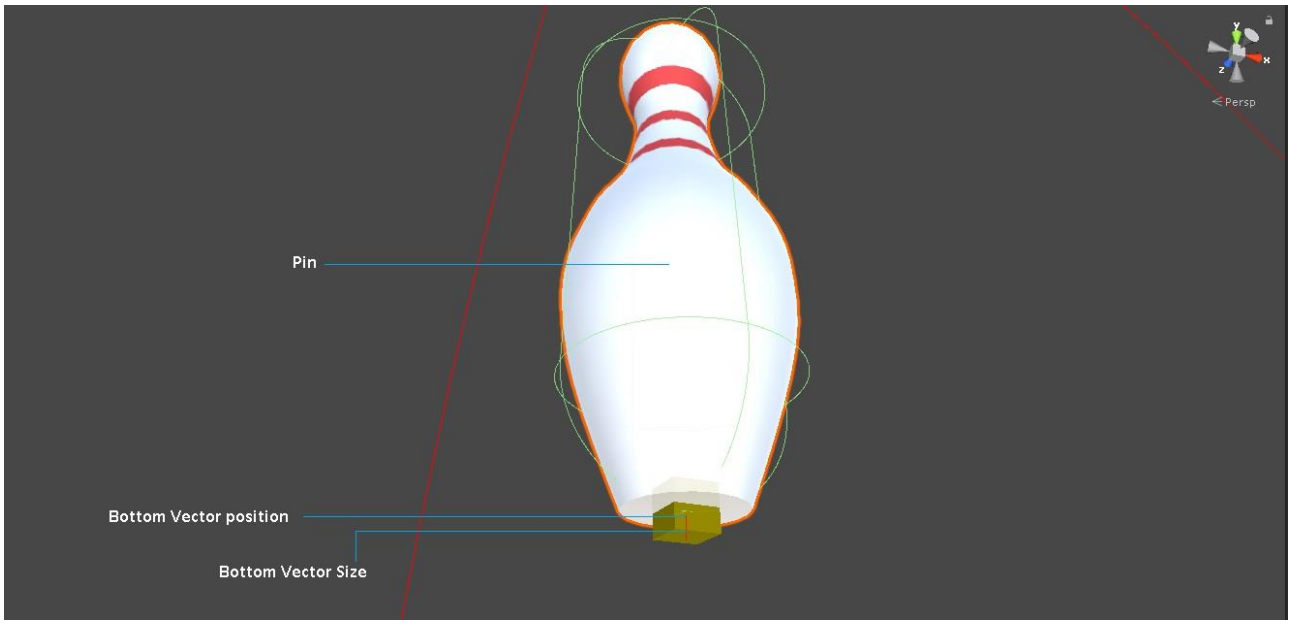


Enable	Make all functions calculation enable or disable
Parameter(s)	
Max movement distance	The distance of region between pin center of mass and hit able position
Bottom vector	Bottom vector position that is connector between ground and pin
Bottom vector size	The size or radius of connector point
Ground mask	The ground mask
Force	
Force to stop	The Boolean value to cause pin more rapidly stopped
Velocity threshold	The value that if ball velocity is less than this value, cause to stop pin
Stop velocity lerp	The lerp to stop pin velocity
Information	
State	Pin state

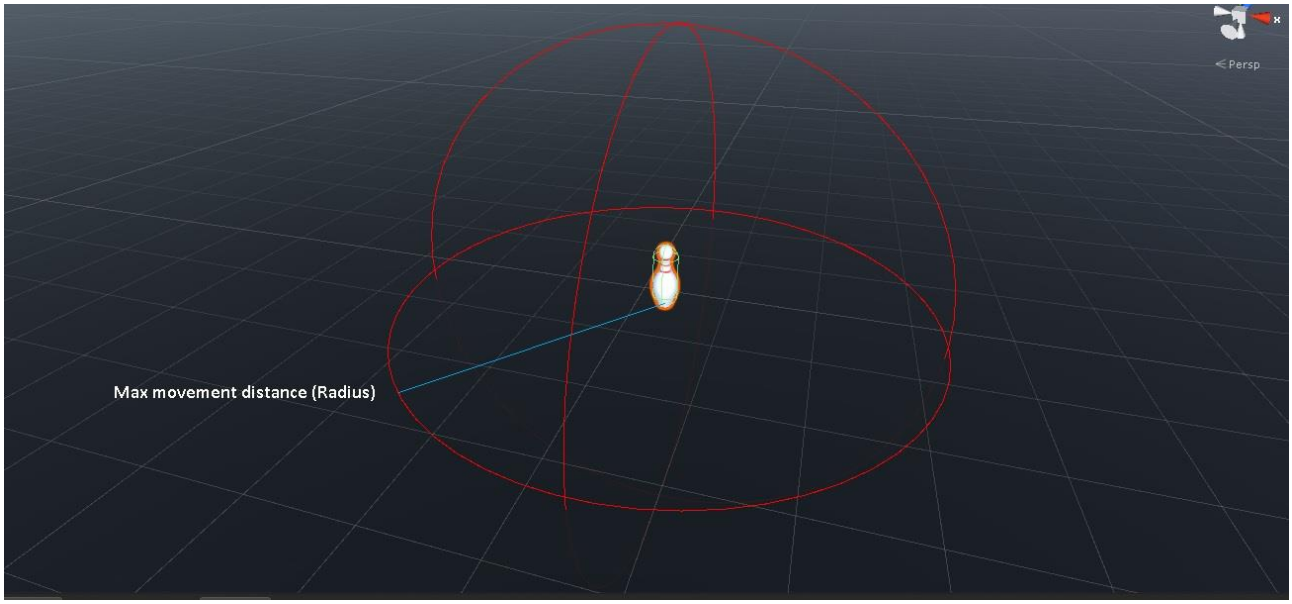
bowling pin **states** are:

- Null: not a valid game state
- `Involved`: there is no special state founded yet but its steel sensing
- Standed: the pin is touching the ground
- Fallen: the pin is not touching the ground

You can see pin bottom vector and ground touching position in this image:



You can see a single pin configuration mode in this image:

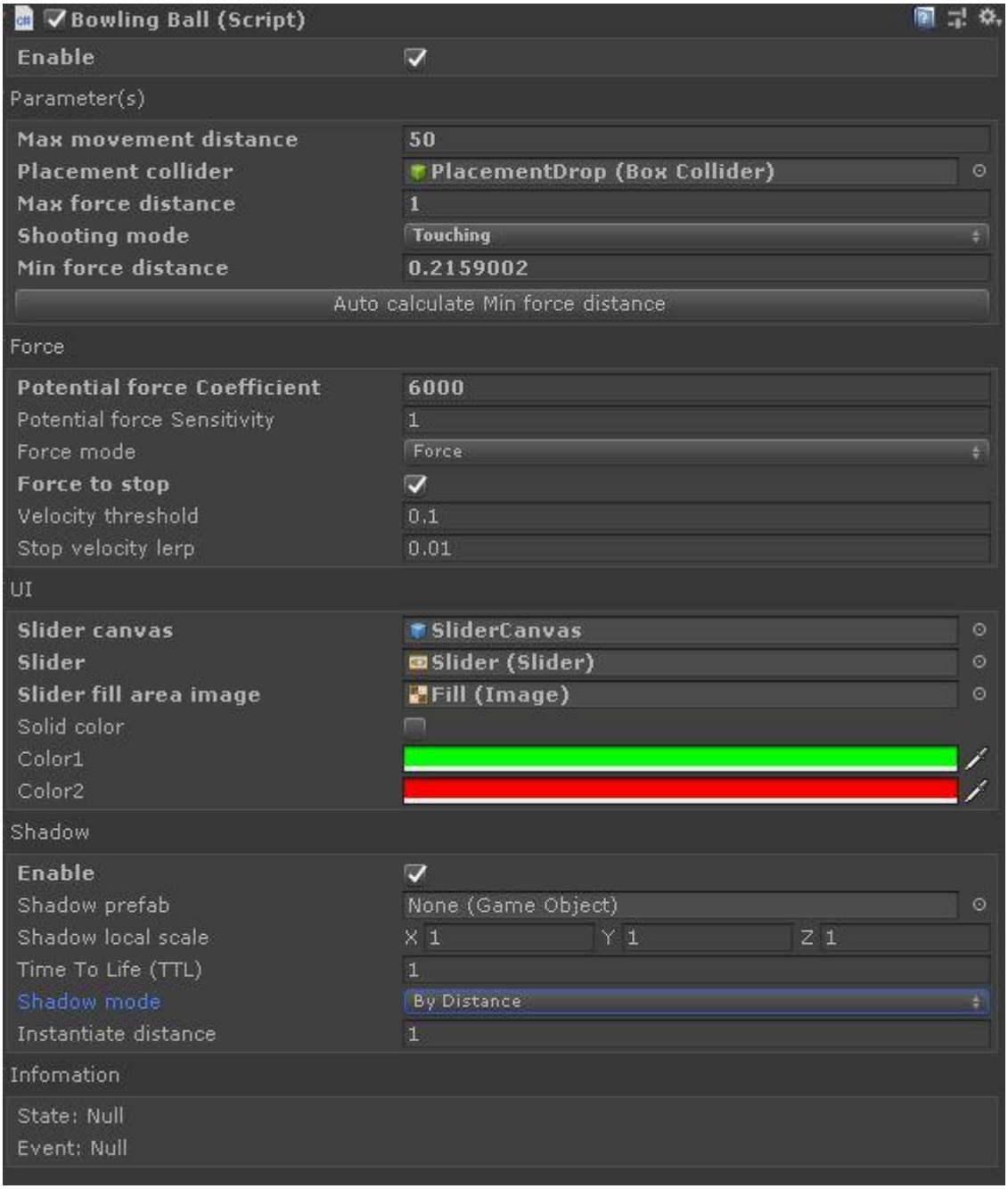


BowlingPinEditor

Editor for BowlingPin class.

BowlingBall

This script is handling bowling ball logics.



Enable	Enable or disable
Parameter(s)	
Max force distance	The distance to end of hit able position to calculate potential force
Max movement distance	Maximum distance that ball can be move
Placement collider	The collider that ball can be drag and drop over that collider
Shooting mode	The ball shooting mode
Shooting mode – Touching	Shooting ball by touch
- Min force distance	The distance of region between ball center of mass and hit able position to calculate potential force
Shooting mode – Tapping	Shooting ball by tapping
Pivot angle	The pivot direction that ball is shooting
Theta	The deviation from the pivot angle Angle = [pivot angle-theta, pivot angle + theta]
Orbit dampening	the orbiting dampening
Power dampening	The power dampening that effected to potential force by delta time
Force	
Potential force coefficient	The value that be multiplied by potential force value
Potential force sensitivity	The value that cause force to potential force to more recently be over flow
Force mode	Unity force mode
Force to stop	The Boolean value to cause ball more rapidly stopped
Velocity threshold	The value that if ball velocity is less than this value, cause to stop ball
Stop velocity lerp	The lerp to stop ball velocity
UI	
Slider canvas	The canvas game object that is slider`s parent
Slider	Slider
Slider fill area image	The image that show potential power
Solid color	Boolean value to show potential power color
Color1	Main potential power color
Color2	High value color to show potential power
Shadow	
Enable	Enable shadow
Shadow prefab	Shadow prefab or game object
Shadow local scale	Shadow local scale
Time to life (TTL)	Shadow time to life in float
Shadow mode	
-By Distance	New shadow will be instanced by delta position
Instance distance	The shadow will be instanced by delta position more than this value
-By Time	New shadow will be instanced by delta time
Instance time	The shadow will be instanced by delta time more than this value
Information	
State	Ball state
Event	Ball event

Golf **Ball state**s are:

- Null: not a valid state
- Idle: ball is waiting for player action
- PositionSelection: ball is waiting to drop on new position by player
- Tapped_FindDelta: ball is generating new angle direction
- Tapped_FindPower: ball is generating new potential power
- Touched: ball is touched by player
- Release: ball is released
- Finish: ball state is finish and waiting for game manager orders

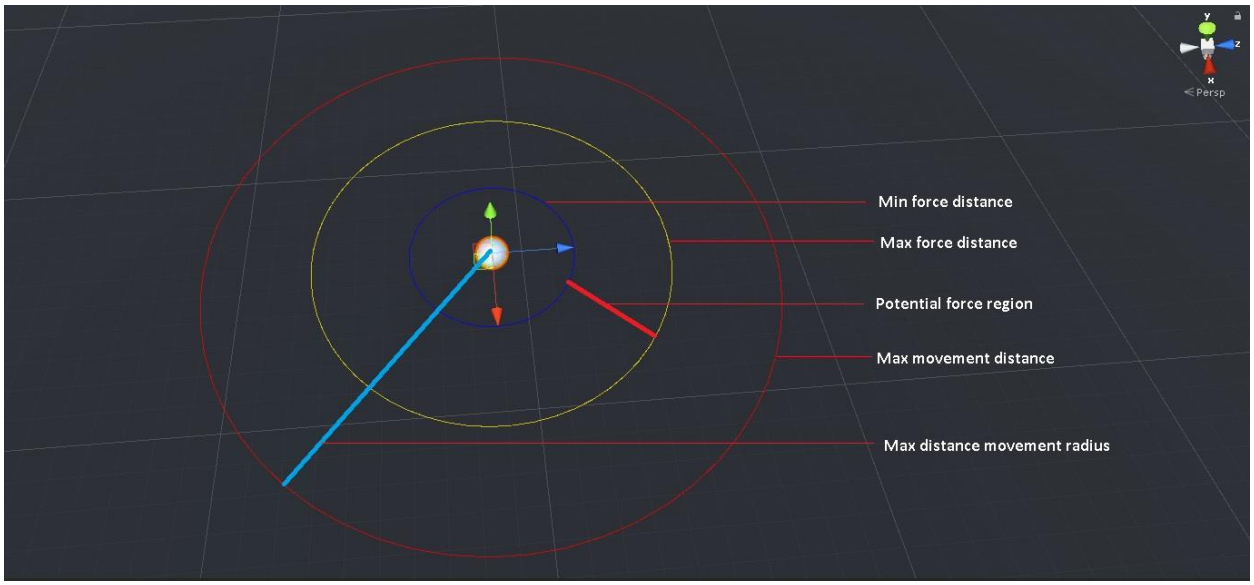
bowling **Ball Event**s are:

- Null: not a valid event
- None: no special event founded
- Far distance: ball is moved for the distance more than max distance value
- Ball stopped: ball is stopped after releasing

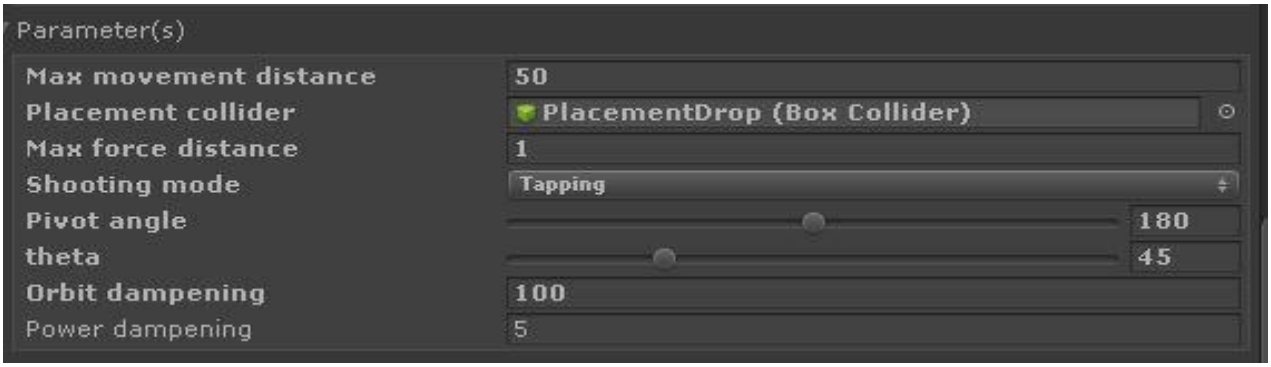
This script has some useful functions that help game designer to develop:

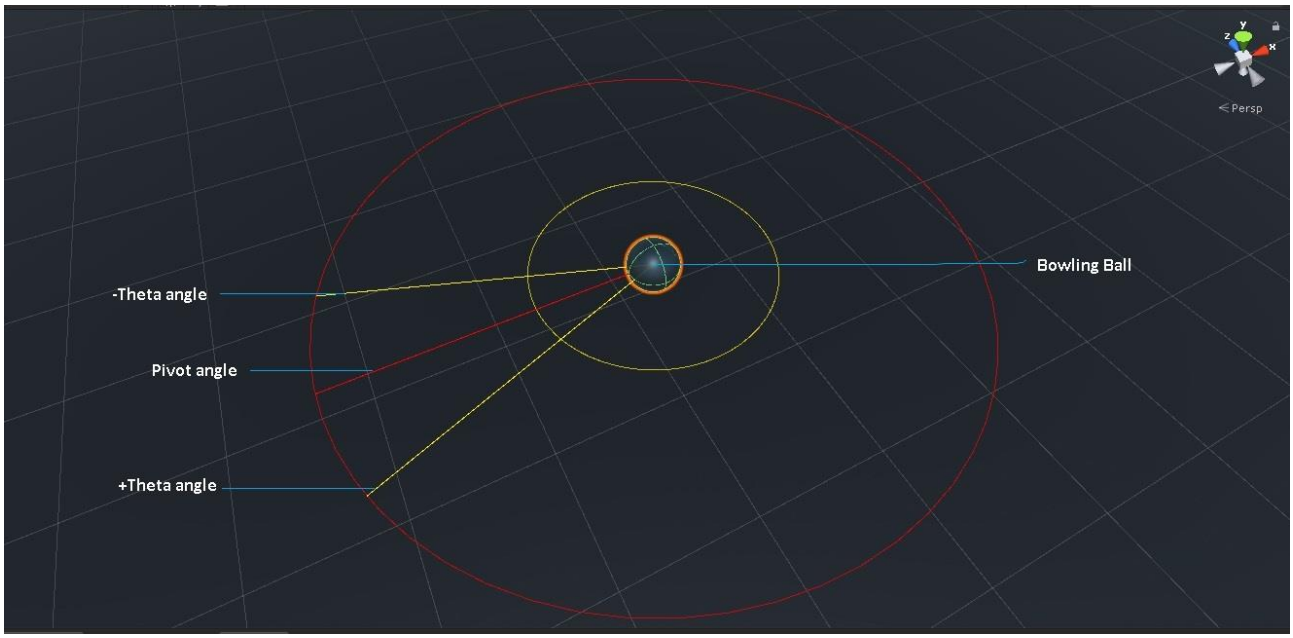
Parameters	
`BallState`	`Read-only current ball state`
`BallEvent`	`Read-only current ball event`
`PotentialEnergy`	`Read-only ball potential energy`
`PotentialForceDirection`	`Read-only ball potential energy direction`
Functions	
`ResetBall()`	`Reset the ball`
`AddForce(Vector3 force)`	`Add force to ball`

You can see bowling ball in configuration mode in this picture by touching shooting mode:



You can see bowling ball in configuration mode and its parameters in this picture by tapping shooting mode:





BowlingBallEditor

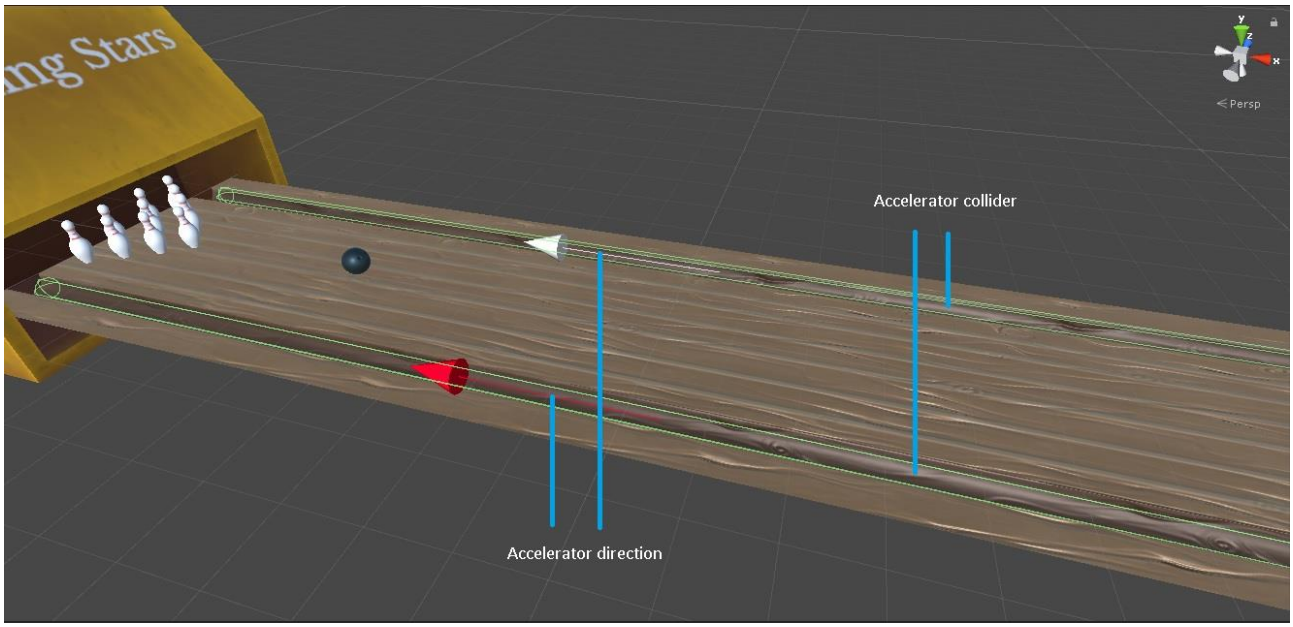
Editor for BowlingBall script class.

AcceleratorSensor

This script is a force direction that if bowling ball stay in this object`s collider, then a force in same direction of arrow direction is generated. It's very useful for take the ball to hole rapidly.

Force power	The force power
Handle arrow size	Arrow size
Arrow color	Arrow color

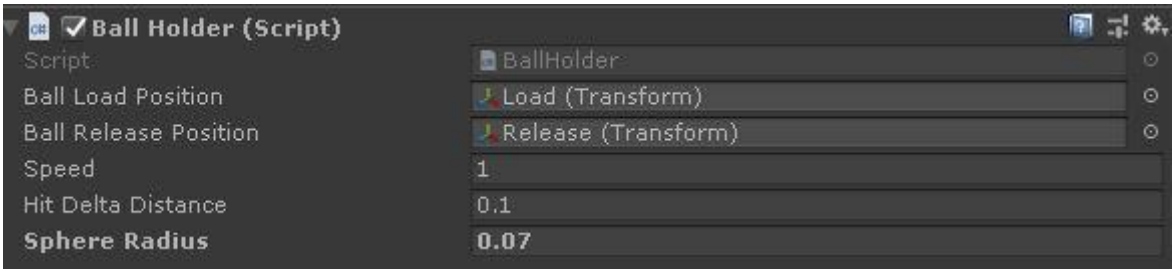
You can see two accelerator objects in this image:



In above image each accelerator can guide ball to the bowling hole.

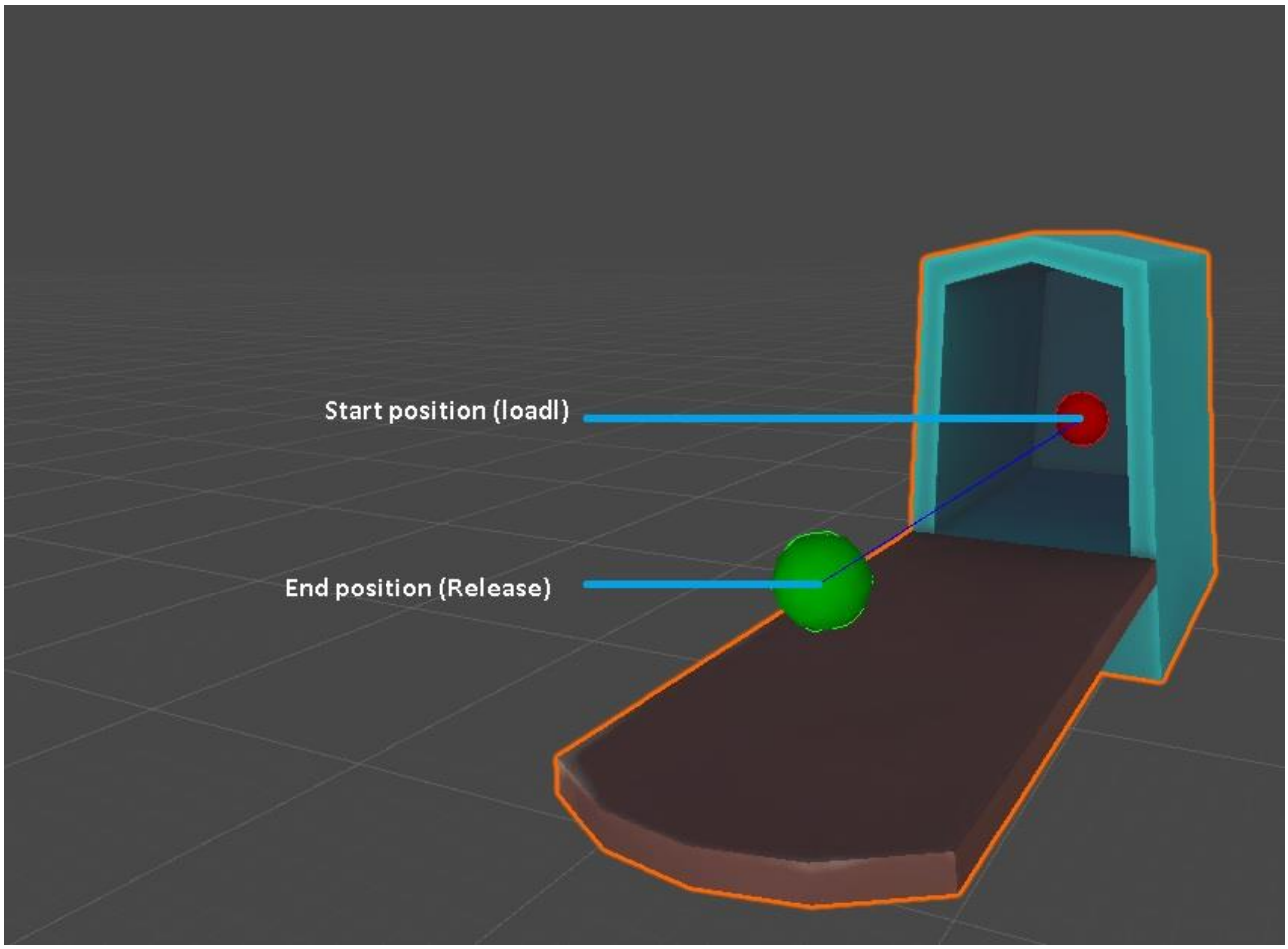
BallHolder

Ball holder is the place that bowling ball after falling in bowling ball, will be loaded from this object and player can drag and drop it to the ground and shooting.



Ball load position	The ball load position
Ball release position	The position that ball transported from load position to this position
Speed	Transporting speed
Hit delta distance	The distance between ball position and release position for free the ball
Sphere radius	The radius to show positions in gizmo for helping and debugging

You can see a ball holder object in this image:



BowlingObjectStopper

Basic class stop pin and ball velocity. When pin or ball stay in this object`s collider then its work.

MainMenu

Basic class for main menu.

MainMenuItem

Include main menu item(s).

BowlingMenu

Main class for main menu.

EditorTools

Main class for create editor.

Extensions

Helper class with useful function(s).

Globals

This is static class for some references.