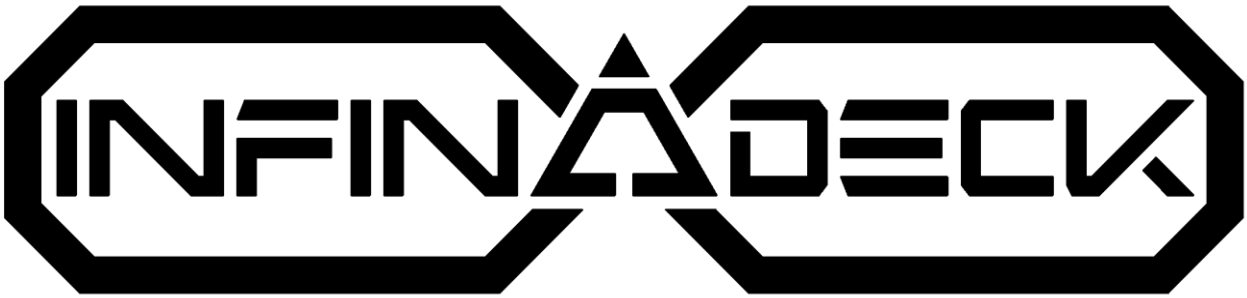


Infinadeck

Unreal Setup & Installation Manual

Original Release for Plugin Version 3.2.0

Manual Version 2.1

10-31-2022

VERSION HISTORY

Version #	Implemented By	Revision Date	Approved By	Approval Date	Reason
1.0	G. Brunner	10-22-2018	B. Freeman	10-22-2018	Initial Release
1.1	G. Brunner	11-2-2018	B. Freeman	11-2-2018	Updated Supported Unity Versions
1.2	G. Brunner	11-14-2018	B. Freeman	11-14-2018	Clarified Vive Tracker Requirements
1.3	G. Brunner	11-26-2018	B. Freeman	11-26-2018	Unreal Release
1.4	G. Brunner	12-21-2018	B. Freeman	12-21-2018	Clarified Unreal Documentation
1.5	G. Brunner	1-16-2019	B. Freeman	1-16-2019	Separated IK Documentation
1.6	G. Brunner	6-1-2020			Simplification of Plugin Install and Use, Separation between Unity and Unreal documents
2.0	G. Brunner	1-6-2022			Updated for major changes of 3.0 release; keybinds info, reference panel, runtime edition
2.1	G. Brunner	10-31-2022			Updated for IDA release; keybind details matched to InfinaKEYBIND documentation

Infinadeck Template Version: 2018-06-20

TABLE OF CONTENTS

1 INTRODUCTION

This guide covers the installation and use of the Infinadeck software and the Infinadeck Unreal plugin. Implementation of this system into a game will allow said game to be played with the use of an Infinadeck omnidirectional treadmill.

Please make a backup of your game files before you begin implementing our plugin. If there are issues using our plugin, follow the Uninstall Guide (see Section 6, Uninstall Guide) to properly remove it.

2 REQUIREMENTS

The following hardware and software components are required to run the Infinadeck SDK:

- Unreal Version: 4.24 or higher (tested through 4.27) **(C++ Project required)**

3 SOFTWARE DOWNLOAD

Infinadeck Desktop Application

The latest version of the Infinadeck Desktop Application software may be found on the Infinadeck GitHub. Please follow the link to download the latest files. Demo projects and a short video to help with the installation and setup for Unity and Unreal may be found in the same location.

- GitHub: https://github.com/Infinadeck/InfinadeckSDK

The following items will be included with the download:

- Infinadeck Desktop Application Installer
- SDK Documentation

Unreal Plugin

The latest version of the Infinadeck Unreal Plugin may be found on either the Unreal Engine Marketplace or the InfinadeckUnrealPlugin GitHub. Please follow either link to download the latest files.

- UE Marketplace: https://www.unrealengine.com/marketplace/en-US/product/infinadeck-plugin
- GitHub: https://github.com/Infinadeck/InfinadeckUnrealPlugin

The following items will be included with the download:

- Unreal Plugin

All documentation for the Unreal Plugin can be found on the Infinadeck Unreal Plugin GitHub.

4 INSTALLATION GUIDE

Once you have downloaded the files (see Section 3, Software Download) you are ready to begin the installation process.

4.1 INFINADECK APP

1. Double click *[DownloadLocation]/IDA_[VersionNumber].exe* to begin installing the Infinadeck Desktop Application.
2. Follow any on-screen prompts to finish the installation.

4.2 UNREAL PLUGIN

4.2.1 Plugin Installation

The Unreal Plugin can be added to your Unreal Engine project in multiple ways.

Install thru Epic Games Launcher:

1. Find **Infinadeck Plugin** on the Unreal Engine Marketplace and click **Add to Cart**, then **Check Out** via the shopping cart icon in the upper right corner.
2. In the Epic Games Launcher, **find Unreal Engine > Library > Vault.**
3. Click **Add to Project** to pull up the **Select the Project to Add the Asset To** menu.
4. Navigate to your C++ project and click **Add to Project** (if your project is not visible, please make sure to check “Show all projects” and that the correct engine version is selected in the bottom left corner).

Manual Install:

1. Under your new or existing Project’s game files, look for a folder named *Plugins*. If this folder does not exist yet, create it.
2. Navigate to *[DownloadLocation]/**InfinadeckUnrealPlugin_[VersionNumber].zip*** and unzip it.
3. Locate the folder named *InfinadeckSDK* within the unzipped folder.
4. Copy this folder and paste it into *[ProjectLocation]/Plugins*.

4.2.2 C++ Building

To verify that the plugin is included in the project, do the following:

1. Right click **[ProjectName].uproject** & select **"Generate Visual Studio project files".**
2. Double click **[ProjectName].sln** to open it in Visual Studio.
3. Right click **[ProjectName]** in the Solution Explorer and select **Build**.
4. Once Visual Studio is done (check the Output Tab), close Visual Studio.

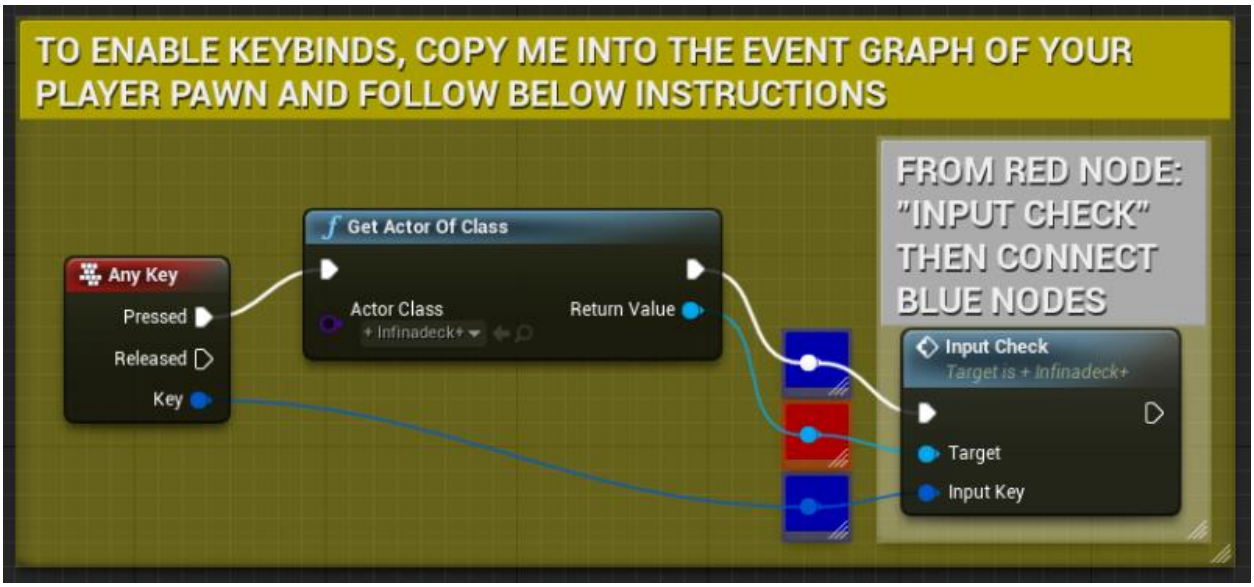
4.2.3 Blueprint Addition

1. After installing the plugin, click on the eye in the corner of the Content Browser and ensure that Plugin Content is set to Visible.
    - If you can't see a large folder named *InfinadeckSDK Content* in your Content Browser, try **closing** and **reopening** your project.
    - If you still can't see the folder, navigate to **Edit > Plugins** and verify that the InfinadeckSDK plugin is **Enabled**.
2. In the *Content Browser/InfinadeckSDK Content/Blueprints*, find the blueprint named **+Infinadeck+.**
3. Grab this blueprint and drag it anywhere in your game world.

4.2.4 Wire Blueprint for Interaction

After clicking on the **+Infinadeck+** blueprint instance, the Details pane for the instance should pop open. In the Details section, set the component values as follows:

- **CameraRig:** Set this to your VR Camera Rig, as in the effective "Room Setup." This is the object the **+Infinadeck+** blueprint will parent itself to at runtime and will be the object moved through the game world.
- **Headset:** Set this to your VR Camera, as in the effective "Headset." This is the object the **+Infinadeck+** blueprint will refer to for appropriate display of reference objects.



To enable keybinds, find the above nodes in the +Infinadeck+ Event Graph and copy them into your player Pawn's Event Graph (for XR, this is typically called MotionControllerPawn- it is also likely your CameraRig!) Make sure to add the Input Check function to the Event Graph as shown, or else inputs will not translate.

4.3 TEST SYSTEM

The system is now ready to run. To test run the software:

1. Open **Infinadeck.exe** from your install location of the Infinadeck Desktop Application. Press the Windows key and type “Infinadeck” to find it with ease.
    - If you are running the system without a treadmill present, please refer to document 910-51-00010: Infinadeck Software Manual – Emulator Mode.
2. With your game window partially visible, click on **Infinadeck.exe**, and move over to the **Manual** tab. Click the Start button. When you move the virtual joystick, the movement should translate to motion in the game.

4.4 REQUIRED CODING PRACTICES

If you have implemented our system into your game, any call of a popup, menu screen, level load, or similar action **must** be accompanied by a call of the function SetTreadmillPause() to guarantee the user does not enter a situation where the deck is active in a non-movement environment.

Not following this step is a direct violation of our Terms and Conditions.

5 TROUBLESHOOTING

The following table outlines common installation issues and solutions:

Installation Issue	**Solution**
Unreal Won't Run Post Plugin Installation	Check to make sure you fit within the required versions listed for this Plugin. If the problem persists, report the error to us via email.
The Game Crashes	Open **Infinadeck.exe** and let it run idly in the background.
The Ring Turns Green BUT the Reference Objects Don't Move	Make sure that **MovementLevel** is set to true.
The Reference Objects Move, But My Room Setup Doesn't Move With It	Make sure that the blueprint's CameraRig is set correctly. Whatever device it is set to will experience the motions you're seeing in the Reference Objects.
I Can't See A Reference Object I Want To See	Check your system's InfinaDATA settings for the Infinadeck Plugin, found here: [*UserDirectory]/Documents/My Games/Infinadeck/Core/infPlugin_preferences.ini*. It is quite likely the desired object is just disabled here.

If you have additional questions, please email *info@infinadeck.com* with the words "PLUGIN HELP REQUESTED" in the subject line.

6 UNINSTALL GUIDE

To remove each of the components of the system:

6.1 INFINADECK DESKTOP APPLICATION

1. Press the Windows key and type **add or remove**, then press Enter.
2. In the pop-up window, scroll to find **Infinadeck Desktop Application [Version Number]. Select it.**
3. Press **Uninstall** to remove the software from your machine.

6.2 UNREAL PLUGIN

1. Delete any and all instances of **+Infinadeck+** from your scene.
2. Remove any custom dependencies between Infinadeck Plugin objects and functions.
3. Delete the **InfinadeckSDK** folder under *[ProjectLocation]/Plugins.*

7 USAGE

There are several ways to interface with the Infinadeck system once it's been installed (see Section 4, Installation Guide).

7.1 PLUGIN SETTINGS

There are several values you can change within your blueprint to make the system function as desired. Separate blueprints from the default system may be saved.

Optional Settings

- **AutoStart(**bool**):** Only set to FALSE if you will later call Boot() on **+Infinadeck+.** Delays the boot cycle of the plugin to keep things lightweight until necessary.
- **FirstLevel(**bool**):** Set to TRUE if this is your initial spawn in-scene. This will spawn an splashscreen with important spec information about the treadmill being used and give awareness that the plugin is hooked into the game. Otherwise, this can be set to FALSE.
- **MovementLevel(**bool**):** Set to TRUE if this is a level where the user should be able to walk. Otherwise, this can be set to FALSE.
- **GuaranteeDestroyOnLevelLoad(**bool**):** Set to TRUE if this actor is getting attached to something that enters DontDestroyOnLoad, to guarantee that it is not maintained between scenes. Otherwise, this can be set to FALSE.

Advanced Settings

- **SpeedGain(**float**): Do not tweak this value without first setting your WorldScale. Can be used to artificially adjust player's motion vector such that they can move through the game world faster (or slower).** Value is only referenced in the +Infinadeck+ blueprint's Boot() function, and as such, if you want to modify this value during runtime, draw reference to InfinadeckLocomotion, which should spawn in the level during runtime, and directly modify its SpeedGain.
- **WorldScale(**Vector3**):** This value should be set to the appropriate value to match the world's Scale. At initial settings of 1x1x1, the system expects that the world is set to default settings, such that 1m =100 Unreal Units.
- **CorrectPosition(**bool**), CorrectRotation(**bool**), CorrectScale(**bool**):** Set these to TRUE if the position, rotation, or scale of the actor is incorrect at runtime. This will appropriately fix the position, rotation, or scale after the actor is reparented.
- **ShowCollisions(**bool**):** Set to TRUE to get a Print comment every frame where the CameraRig is moved by something other than Infinadeck Locomotion.
- **ShowTreadmillVelocity(**bool**):** Set to TRUE to get a Print comment every frame, reporting back the treadmill speed according to Infinadeck Locomotion.

7.2 INFINADATA SETTINGS

Many settings used by this plugin are stored via InfinaDATA, which allows the values to be modified via a custom INI file, accessible across game engines.

7.2.1 General Preferences

The general preferences for this application are saved under the following directory:
*User/Documents/My Games/Infinadeck/Core/**infPlugin_preferences.ini.***

[KeybindPreferences]

- These are all detailed in Section 7.3.1, InfinaKEYBIND Universal Keybind System.

7.2.2 Reference Object Preferences

The Reference Object preferences for this application are saved under the following directory:
*User/Documents/My Games/Infinadeck/ReferenceObjects/**infPlugin_refObj_preferences.ini.***

[ReferenceObjects]

- **ringVisibility(**bool**):** Set to TRUE to enable the reference ring. Otherwise, set to FALSE.
- **ringModel(**int**):** Set to the index of the desired model (0-5).
- **enableFixedRingHeights(**bool**):** Set to TRUE to enable fixing the reference ring's height as per the pre-allocated pinned ring positions. Otherwise, set to FALSE.
- **baseFixedRingHeight(**float**):** Set to the inch-value representation of the minimum height of the ring, when placed at the lowest setting.
- **modulusOfFixedRingHeight(**float**):** Set to the inch-value representation of the change in height per pin position of the ring.
- **indexOfFixedRingHeight(**int**):** Set to the pin number for the ring's current use (0-4).
- **centerVisibility(**bool**):** Set to TRUE to enable the center mark. Otherwise, set to FALSE.
- **centerModel(**int**):** Set to the index of the desired model (0-4).
- **forceCenter(**bool**):** Set to TRUE to force controlling the position of the reference objects through this settings file. Otherwise, set to FALSE.
- **centerX(**float**):** Set to the X dimension of the reference object's position (if **forceCenter**).
- **centerY(**float**):** Set to the Y dimension of the reference object's position (if **forceCenter**).
- **centerZ(**float**):** Set to the Z dimension of the reference object's position (if **forceCenter**).
- **edgeVisibility(**bool**):** Set to TRUE to enable the reference edge. Otherwise, set to FALSE.
- **deckVisibility(**bool**):** Set to TRUE to enable seeing a virtual copy of the Infinadeck in-game. Otherwise, set to FALSE. Note: If the Speed Gain in-game is set to 1, the deck's surface should appear to move one-to-one.

- **deckHeadingVisibility (**bool**):** Set to TRUE to enable seeing a heading indicator along the edges of the deck surface in-game, better indicating where the treadmill is moving. Otherwise, set to FALSE.

[TreadmillConfiguration]

- **ringDiameter(**float**):** Set to the meter-value of the physical ring's major diameter.
- **ringThickness(**float**):** Set to the meter-value of the physical ring's major diameter.
- **walkingSurfaceWidth(**float**):** Set to the meter-value of the walking surface width.
- **walkingSurfaceEdgeThickness(**float**):** Set to the desired meter-value of the walking surface edge thickness.

[DynamicPanel]

- **dynamicRingPanel(**bool**):** Set to TRUE to enable the "dynamic ring panel", or the reference panel. Otherwise, set to FALSE.
- **panelWidthM(**float**):** Set to the meter-value of the desired horizontal depth of the panel.
- **panelHeightM(**float**):** Set to the meter-value of the desired vertical height of the panel.
- **panelDiameterM(**float**):** Set to the meter-value of the desired inner diameter of the panel.
- **bandThicknessPercent(**int**):** Set to the desired thickness of the dynamic band (0-100).
- **topBoundaryThicknessPercent(**int**):** Set to the desired thickness of the top band (0-100).
- **bottomBoundaryThicknessPercent(**int**):** Set to the desired thickness of the lower band (0-100).
- **dynamicBackdrop(**bool**):** Set to TRUE to make the backdrop grow and shrink with the deck speed. Otherwise, set to FALSE.
- **panelPalette(**int**):** Set to the desired panel palette (0-4)
- **colorblindMode(**bool**):** Set to TRUE to enable seeing reference symbols that better indicate deck state, both for colorblind users and those who want more reference visibility. Will force-enable the reference panel. Otherwise, set to FALSE.
- **dynamicColorblindElements(**bool**):** Set to TRUE to make the colorblind elements move dynamically, based on the treadmill's run state. Otherwise, set to FALSE.
- **dynamicColorblindFrames(**int**):** Set to the desired number of frames required to move the dynamic colorblind elements through a full loop.
- **maxTreadmillSpeedMetersPerSecond(**float**):** Set to the current maximum speed of the treadmill. This number does not affect treadmill performance but does affect the dynamic panel's illustration of deck speed.

7.3 KEYBINDS

7.3.1 InfinaKEYBIND Universal Keybind System

The keybinds used by this plugin are part of the InfinaKEYBIND Universal Keybind System, which allows these keys to be accessible in any project, no matter what keys have already been bound.

This is done through "**links**". **Linked functions** are established by the plugin developer, and **linked keys** are alterable by the end user, without requiring any explicit development by the project developer.

This system supports the following, all exposed via the InfinaDATA settings file at [*UserDirectory]/Documents/My Games/Infinadeck/Core/infPlugin_preferences.ini.*

[Keybind Preferences]

- **keyboardInputEnabled(**bool**):** Set to TRUE to enable input; Otherwise, set to FALSE.
- **exportBindings(**bool**):** when TRUE, any input detected by the project during runtime will cause link01 through link12 to be freshly exported to the settings file.

[Keybinds]

- **keybindProfile(**string**):** Assign the following **linked keys** to **link01 thru link12** based on your desired input scheme:
  - FUNC
    - F1, F2, F3, F4, F5, F6, F7, F8, F9, F10, F11, F12
  - 1234
    - Alpha1, Alpha2, Alpha3, Alpha4, Alpha5, Alpha6, Alpha7, Alpha8, Alpha9, Alpha0, Minus (-), Equals (=)
  - #PAD
    - Keypad1, Keypad2, Keypad3, Keypad4, Keypad5, Keypad6, Keypad7, Keypad8, Keypad9, KeypadDivide (/), KeypadMultiply (*), KeypadMinus (-)
  - STND
    - LeftShift, LeftControl, LeftAlt, Space, RightShift, RightControl, RightAlt, Return, BackQuote (`), Tab, Backslash (\), Backspace
  - CPAD:
    - LeftArrow (←), DownArrow (↓), RightArrow (→), UpArrow (↑), Delete, End, PageDown, Insert, Home, PageUp, ScrollLock, Pause
  - QWER
    - Q, W, E, R, T, Y, U, I, O, P, LeftBracket ([), RightBracket (])
  - ASDF
    - A, S, D, F, G, H, J, K, L, Semicolon (;), Quote (‘), Slash (/)
  - Custom: Custom string of keybinds set with **customBinding**

- **customBinding(**string**):** If **keybindProfile** is set to “Custom”, this string of keybinds will be read in for use by the engine.
    - Separate keybinds with a ‘ - ’ character.
    - Example Custom Keybind: Q-W-E-R-T-Y-U-I-O-P-LeftBracket-RightBracket
    - Any keybind listed in the keyBindProfiles can be used in a custom binding.
- **link01 thru link12(**string**):** the exported list of current keybinds being used by the system. Modifying these values in the settings file do nothing; they will only be written to if **exportBindings** is set to TRUE.

7.3.2 Keybound Functions Accessible to this Plugin

The default **keybindProfile** for this file is #PAD (for more information, see **keybindProfile** in Section 7.3.1, InfinaKEYBIND Universal Keybind System).

- **link01 (Keypad1)- Toggle Deck Ring**- inverts the visibility of the reference ring.
- **link02 (Keypad2)- Toggle Deck Edge**- inverts the visibility of the reference edge.
- **link03 (Keypad3)- Toggle Deck Center**- inverts the visibility of the reference center.
- **link04 (Keypad4)- Toggle Reference Panel**- inverts the visibility of the reference panel.
- **link05 (Keypad5)- Toggle In-Engine Deck**- inverts the visibility of the in-engine treadmill.
- **link06 (Keypad6)- Toggle Heading**- inverts the visibility of the treadmill heading.
- **link07 (Keypad7)- Toggle Colorblind**- inverts the visibility of the colorblind elements.
- **link08 (Keypad8)- Cycle Panel Theme**- cycles through the available panel themes.
- **link09 (Keypad9)- Boot/Shutdown/Reboot Plugin**- Used to physically enable and disable the plugin during runtime. When disabled, all subcomponents will be deleted; when re-enabled, they will all be respawned. Useful for extra-lightweight testing.
- **link10 (Keypad0)- Start/Stop Treadmill-** Starts the treadmill if it is currently stopped and stops the treadmill if it is currently started.
- **link11 (KeypadPeriod)- Import Preferences-** Manually imports preferences directly from the settings files. Good for bypassing the default long “check if file updated” cycle.
- **link12 (KeypadEnter)- Reset Preferences-** Manually pushes default preferences onto the settings files. This is equivalent to stopping the project, deleting the settings files, and starting the project back up, but without any downtime or loss of progress.

7.4 COMMON GAME FUNCTIONALITY IMPLEMENTATIONS

Given that the base functionality of this UI may not really allow for implementation in a more complex game, a walkthrough on the implementation of several common features has been included.

7.4.1 Non-Walking Scenes

In the plugin settings, set **MovementLevel** to FALSE and **GuaranteeDestroyOnLevelLoad** to TRUE for each scene that should have no walking.

7.4.2 Fast-Traveling

While the Infinadeck can let the user walk anywhere, it may be beneficial in larger scenes to allow the user to teleport to various locations (think the size of Egyptian Pyramids). This can be done with either a submenu that pulls up various desirable fast travel locations or traditional VR teleportation.

The only rule to keep in mind is that the user should not be capable of moving on the deck while no visual is displayed on their screen. An instantaneous teleport, or <50ms black screen teleport should be fine, but anything longer should be accompanied by the appropriate calls to the Infinadeck API's SetTreadmillPause().

7.4.3 Make Infinadeck a Setting that can be Enabled from the menu

In the plugin settings, set AutoStart to FALSE. Give the menu object a direct link to the instance of +**Infinadeck+** in the game world. The menu object needs to call the Boot() function on +**Infinadeck+** to get it to start up appropriately.

8 ADVANCED DOCUMENTATION

The following subsections go in-depth for more sophisticated concepts than those covered elsewhere in this guide. They exist to give creators a better understanding of how our system works so that they may better interface and work with our system.

8.1 REFERENCE POINTS

There are a couple critical reference points that users need to have visible to help ground the use to the real world while in virtual reality.

8.1.1 Ring

The virtual ring serves two primary purposes for the user. First, it is a physical object that any user who is feeling afraid, unsafe, or unstable can grab onto. Second, the ring serves as a visual grounding point. Whenever the deck speed is at a mismatch to the user, the user gains a greatly increased chance of nausea that is mitigated by the existence of the ring. As the ring is tied to the deck and not the user, the brain can easily see how much mismatch exists between the deck and the user, and thus prevent any locomotion-based nausea.

8.1.2 Deck Edge/Center Mark

The deck edge serves a very similar purpose to the ring, functionally being a less-critical floor-based reference point. Given that the system has a speed limit, users can look to see exactly where their stride lands, so that they comfortably and intuitively know the system's limits. The center mark lets users know both how far they are from center.

8.1.3 Reference Panel

Some users like having visibility of their current speed relative to the current maximum treadmill speed, to better gauge how fast they can go. The reference panel dynamically illustrates the relative speed in a visual way and is highly customizable to a user's preferences.

8.1.4 Colorblind Elements

The treadmill's state is usually clarified by changing the color of reference objects. When this isn't sufficient, these symbols can be enabled to clearly tell what state the Infinadeck is in at any given time.

8.1.5 Virtual Deck/Deck Heading Indicator

The virtual copy of the deck allows for checking for parity between deck motion and world translation. In addition, both it and the Deck Heading Indicator give the user a clearer idea of exactly how the deck is moving under their feet, which can be a useful reference point for new users.

9 MISSING A FEATURE?

Missing a feature that you'd like from our plugins? Please contact us at info@infinadeck.com with the words "PLUGIN FEATURE REQUEST" and detail your requested feature as clearly as possible. The more information provided, the better. If the feature makes sense in our roadmap, we'll integrate it; otherwise, we'll try to help support your unique use case directly.

10 KNOWN BUGS

Below are a list of Known Bugs and Fixes we are currently working on. Please contact us at info@infinadeck.com with the words "PLUGIN BUG DISCOVERED" in the subject line if you find any additional bugs or encounter any issues.