DTrack Plugin for the Unity Game Engine 2019.x

This is a component for Unity 2019.1 or later with the purpose of native integration of the Advanded Realtime Tracking (ART) DTrack (versions 2 and 3) tracking solutions. This Unity Asset provides access to DTrack tracking data, that is send over network using UDP/IP datagrams. Each UDP packet contains one frame of tracking data including all outputs activated via the DTrack software (see Section **DTrack**). This package currently supports the DTrack body 6d and flystick 6df2 data formats.

Download

You can download or clone sources for this Asset package at GitHub

Prerequisites

To use this Asset the following components are required.

- Unity Editor 2019.1 or later
- Windows 64 bit, Linux 64 bit
- ART Tracking system and DTrack2/DTrack3
- UnityDTrackPlugin package (Asset)
- DTrack User's Guide (optional)
- DTrack Programmer's Guide (optional)

Creating the Unity DTrack Plugin package from sources

In order to create a Unity package from the provided sources, follow the steps below.

- 1. Download or clone sources for this Asset package (see Download)
- 2. Launch Unity
- 3. Create new Unity project (e.g., "MyUnityProject")
- 4. Extract the asset package, if applicable
- 5. Copy the directory *UnityDTrackPlugin* to your projects asset directory (/path/to/unity/projects/*MyUnityProject*/Assets/**DTrack**)
- 6. Export package (right-click on DTrack in the **Project** window and select **Export Package...**)

Importing Unity package

- 1. Launch Unity
- 2. Create/Open Unity project
- 3. Import package (Assets \rightarrow Import Package... \rightarrow Custom Package...)

DTrack Configuration

Find here a quick-start guide to DTrack. For details, please, refer to your DTrack User's Guide and DTrack Programmer's Guide, that is shipped with the DTrack distribution. In this section we assume that the ART tracking system is properly set up and a room calibration was done. Further, a set of 6DOF targets and flysticks are calibrated.

Room Calibration

For general information about the DTrack room calibration and room adjustment see the DTrack User's Guide. Here we discuss details relevant for use with the Unity Engine.

The calibration angle which comes with your ART tracking system defines the coordinate system layout in your tracking area. It consists of four retroreflective or active markers mounted onto an L-shaped frame.



The marker on top of the edge of this L-shape by default designates the origin of the DTrack coordinate system. When using the *Normal* calibration mode (see Figure below), the long arm of this L-shape corresponds to the X axis and the short arm to the Y axis. DTrack coordinates refer to a right-handed coordinate system, so when the angle is placed flat on the ground with the markers pointing up the Z axis points upwards.

wand length [mm]						
410,00						
	marker distances	Room Calibration Set 410 (high)				
z 🕇	A (1-2) [mm]	384,0				
	B (1-4) [mm]	114,0 225,0				
1.	C (1-3) [mm]					
x	height [mm]	43,0				
coordinate system						
Normal						
re-calibration (without angle too	1)					
Show <u>d</u> etails <u>T</u> ransfer		<u>C</u> alibrate E <u>x</u> it				

The plugin transforms a right-handed position of a DTrack 6DOF measurement to a left-handed Unity position by switching the Y and Z axes, i.e.,

(X_{Unity} , Y_{Unity} , Z_{Unity}) = (X_{DTrack} , Z_{DTrack} , Y_{DTrack}).



DTrack offers a multitude of ways to adjust coordinate systems for room and bodies, e.g., offsets, scaling, additional rotations, or shifting the origin of bodies. Consult your manual for details on *Room adjustment* and *Body adjustment*.

Setting outputs

To configure the tracking data stream generated by DTrack, execute these steps:

- 1. Activate a channel if needed
- 2. Fill in the IP/Port of the device receiving tracking data
- 3. Select outputs you are interested in (i.e., currently frame counter fr, time stamp ts, 6DOF standard body 6d and flystick 6df2 are supported) via the menu *Tracking* \rightarrow *Output* (DTrack3) or *Settings* \rightarrow *Output* (DTrack2), respectively.

Chann	nel 1 Cł	hannel 2									
✓ <u>a</u> ct	ive										
	<u>s</u> end	<u>U</u> DP port									
	my.u	my.unity.host.ip 5000									
	this computer send data divis										
multicast (224.0.1.0 - 239.255.255.255)											
Ide	entifier			Description							
		frame co	unter								
ts		timestan	np								
60	Ical	number	of calibrated b	odies							
√ 6d	1	6DOF sta	andard body								
30	I	3DOF ma	arker								
√ 6d	lf2	Flystick									
60	Imtr	Measure	ment Tool refe	rence							
gl		Fingertra	icking hand								
gle	cal	number	of calibrated Fi	ingertracking h	ands						
60	li	6D inerti	al body								
60	lmt2	Measure	ment Tool								
act a	s <u>r</u> outer fo	or tracking	output								
					OK Cancel						

Data display (6DOF)

Enable the *data display* view ($View \rightarrow Data$) to find assigned body IDs. In this table, both position and rotation for standard bodies can be observed while moving the target inside the tracking volume.

<u>D</u> Track	3 H <u>a</u> n	dware	<u>T</u> racking	<u>V</u> iew <u>T</u>	ools De <u>b</u> u	ig <u>H</u> elp							
Stop		c01 10)/10 c	02 10/10	0 🛛 🗍 syn	c 120 H	Hz I	2 0			∕ ×		
Data Dis	play												đ×
Bod	y ID	Name	Filter	x [mm]	y [mm]	z [mm]	rx [deg]	ry [deg]	rz [deg]	3DOF ID	x [mm]	y [mm]	z [mm]
	L		active	81.77	664.99	77.88	66.55	-22.58	49.96				
1	2		active	0.00	0.00	0.00	0.00	0.00	0.00				

Data display (Flystick)

Enable the *flystick* view ($View \rightarrow Flystick$) to find assigned flystick IDs. Listed flystick IDs are prefixed with a capital **F**. When referencing flysticks from within Unity in the DTrackReceiverFlystick mask, this prefix must be removed. In addition to position and rotation data, button presses (*b1,...,b6*), joystick movement (*jx,jy*) are illustrated in the table.

<u>D</u> Track3	H <u>a</u> rdware	<u>T</u> rackin	g <u>V</u> iew	Tools	De <u>b</u> ug	<u>H</u> elp											
Stop	🛛 c01	10/3	c02	9/4	sync	120 Hz		1 34 1	• 4					((<u>1</u>)) >>			
Flystick Di	splay																đX
Flystick	head	b1	b2	b3	b4	b5	b6	j×	jy	filter	x [mm]	y [mm]	z [mm]	rx [deg]	ry [deg] r.	z [deg]	
F1								-0.80	-0.55	active	175.73	569.84	198.68	73.59	6.87 -1	146.79	
F2								0.00	0.00	active							

Plugin Configuration

Streaming position, rotation and button events data from DTrack tracking systems to objects in your scene, requires appropriate network settings. In your scene add an *Empty* game object and give it a name, e.g., **DTrackSource**. To this object attach the **DTrack** script via *Add Component* \rightarrow *Scripts* \rightarrow *DTrack* \rightarrow *DTrack*. Set *Listen Port* number matching the setting for DTrack (see Section **Setting outputs** below). Note that 3D position data in the DTrack output stream have unit millimeters. The DTrack Unity Plugin converts such values to unit meter.

6 Inspector						ì .	=
Tag Untagged			aye		Sta		
▼ 人 Transform					2	킕	\$,
Position	х						
Rotation	х						
Scale	x						
🔻 💀 🔽 D Track (Script					1	킕	\$,
		DTrack					
Listen Port		000					
Element 0		CylinderT		(Flystick)			
Element 1	1	CubeTwo	(6[DOF)			
Element 2		CubeOne	(6[DOF)			
Element 3	1	CylinderC		(Flystick)			
,	٩d	d Compoi	ner	ıt			

Applying 6DOF Body data

In your scene attach via *Add Component* the DTrack script DTrackReceiver6Dof to an object you want to receive positional and rotational data. In the DTrackReceiver6Dof mask type in the ID that was assigned to the body by DTrack (see Section **Data display (6DOF)**).

 Inspector 						ê	≖≡
👕 🔽 CubeOne (6[00	F)				Statio	
Tag Untagged							
▼ 👃 Transform							! \$,
Position		0.312457		0.463391		0.3264	1
Rotation		74.44601		-3.629		-177.8	21
Scale							
🕨 🧱 🛛 Cube (Mesh Fi	lte	r)				1	! \$,
🕨 鼳 🔽 Mesh Renderer						R -	! \$,
🕨 💗 V Box Collider						1	! \$,
🔻 📾 🔽 D Track Recei	ve	r 6 Dof (Sc	rip	it)		8 -	ļ \$,
Script		DTrackRei					
Body Id							
Frame		988					
Apply Position	V	1					
Apply Rotation	V	1					
Default-Mate						1	\$,
Shader Stan							
	Ad	ld Compor	ner				
		110					

When the ART tracking system is running, you should now be able to see *Position* and *Rotation* data in the **Transform** box, as soon as you switch to *Play* mode.

Applying Flystick data

In your scene attach via *Add Component* the DTrack script DTrackReceiverFlystick to an object you want to receive positional and rotational data as well as interactive button and joystick events. In the DTrackReceiverFlystick mask type in the ID that was assigned to the flystick by DTrack (see Section **Data display (Flysticks)**).

 Inspector 						i +≡			
👕 🔽 CylinderOne	(Flystick)				Stat	ic 🔻			
Tag Untagged		Laye			ŧ				
V h Transform Position Rotation	X 0.753 X -5.83	579 Y 3 Y	0.309315 -63.532		0.615 -51.7	;: ☆, 5033 205			
Scale	× 20		20		20				
Cylinder (Mesh	Filter)				181				
Mesh Renderer					191				
E Capsule Collid					121	<u>_</u> ! ≎,			
🔻 📠 🗹 D Track Receiv	er Fly St	ick (Sc	ript)		3	⊒! \$,			
	DTrac	kReceiv				•			
Flystick Id									
Frame	5812								
Apply Position									
Apply Rotation									
Controller 1	-0.97								
Controller Speed 1	0.8								
Controller 2	-0.21								
Controller Speed 2	0.8								
Number Of Buttons									
Button 1	\checkmark								
Button 2									
Button 3									
Button 4									
Button 5									
Button 6									
Default-Mater					1	a \$,			
► Shader Stand									
	Add Com	iponer	ıt						

Attaching 6DOF Targets to Camera

For non-static, point-of-view cameras, you can attach a DTrack Receiver with positional and rotational data, e.g., 6DOF body or flystick.

 Inspector 						
👕 🦳 Camera (PO					Static	
Tag Untagged		‡ La	aye			
🔻 📕 Transform					1	\$,
Position		-0.18630		0.316571	0.2019	
Rotation		-41.452		99.17101	75.742	
Scale						
🕨 🖆 🔽 Camera						\$,
🔻 📾 📝 D Track Rece	ive	r 6 Dof (Sc	rip	ot)	R 1	\$,
Script		DTrackRe				
Body Id						
Frame						
Apply Position	V	1				
Apply Rotation	V	1				
	Ad	d Compor		ıt		