

Namespace DOTSDynamicBone

Classes

[DOTSDynamicBoneBaker](#)

[DOTSDynamicBoneClass](#)

This holds and displays all the data needs to modify to create the DOTSDynamicBone data

[DOTSDynamicBoneClass.EditorParticle](#)

[DOTSDynamicBoneComponent](#)

This is a component that creates a single DOTSDynamicBone onto an Entity. If you want more please use the DOTSDynamicBonesComponent or the IndependantDOTSDynamicBoneComponent

[ExcludeEntityFromSimulation](#)

This class holds data related to simulation exclusion

Structs

[DOTSDynamicBone](#)

This is a DOTSDynamicBone. This struct holds data used for calculations

[DOTSDynamicBoneCulling_Tag](#)

[DOTSDynamicBoneEntity](#)

This is a shared component that links all particles to it's respective DOTSDynamicBone

[DOTSDynamicBoneInitializerTag](#)

[DOTSDynamicBoneParticle](#)

[DOTSPlane](#)

A DOTS representation of a Unity Plane

[EditorDOTSDynamicBoneCullingSettings](#)

[LinkedDOTSDynamicBones](#)

Used in a DOTSDynamicBone Entity to link created Bones with the Original Entity.

[Particle](#)

This contains all the data related to a section the DOTSDynamicBone. Here the data is store and calculations are performed on the data

[ParticleTransform](#)

This is an imitation of a Unity Transform since RigidTransform didn't give me enough information

Enums

[DOTSDynamicBone.FreezeAxis](#)

[DOTSDynamicBone.UpdateMode](#)

[DOTSDynamicBoneClass.UpdateDataFrom](#)

[ParticleTransform.DOTSDynamicBoneColliderType](#)

Struct DOTSDynamicBone

Namespace: [DOTSDynamicBone](#)

Assembly: DOTSDynamicBone.dll

This is a DOTSDynamicBone. This struct holds data used for calculations

```
[Serializable]
[BurstCompile]
public struct DOTSDynamicBone : IComponentData, IQueryTypeParameter
```

Implements

IComponentData, IQueryTypeParameter

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Remarks

This is a DOTSDynamicBone. This struct holds data used for calculations

Fields

Disabled

Set this to true to disable calculations on the bone

```
public bool Disabled
```

Field Value

[bool](#)

set to true to disable all particle calculations within the bone set

UpdateLocalPositionOfParticles

```
public bool UpdateLocalPositionOfParticles
```

Field Value

[bool](#) ↗

If set to true then the local positions of the particles will be overwritten.

UpdateLocalRotationOfParticles

```
public bool UpdateLocalRotationOfParticles
```

Field Value

[bool](#) ↗

If set to true then the world positions of the particles will be overwritten.

UpdateTransformAsIfNoParent

```
public bool UpdateTransformAsIfNoParent
```

Field Value

[bool](#) ↗

UpdateTransformButFlipped

```
public bool UpdateTransformButFlipped
```

Field Value

[bool](#) ↗

UpdateWorldPositionOfParticles

```
public bool UpdateWorldPositionOfParticles
```

Field Value

[bool](#) 

If set to true then the world positions of the particles will be overwritten.

UpdateWorldRotationOfParticles

If set to true then the world rotations of the particles will be overwritten.

```
public bool UpdateWorldRotationOfParticles
```

Field Value

[bool](#) 

UseAnimatedLocalToRoot

```
public bool UseAnimatedLocalToRoot
```

Field Value

[bool](#) 

set this to true to have all calculated values be converted into AnimatedLocalToRootValues (required for it to not look weird, but disable it if you want, it's your game)

UseLocalTransformOnSkinMesh

```
public bool UseLocalTransformOnSkinMesh
```

Field Value

[bool](#) ↗

UseNaturalColliders

`public bool UseNaturalColliders`

Field Value

[bool](#) ↗

m_BoneTotalLength

`public float m_BoneTotalLength`

Field Value

[float](#) ↗

total length of the bone

m_Core

`public ParticleTransform m_Core`

Field Value

[ParticleTransform](#)

This is a ParticleTransform that stores the data related to the m_RigComponentEntity Transform

m_Damping

```
public float m_Damping
```

Field Value

[float](#)

How much the bones slowed down.

m_DistanceDisable

```
public bool m_DistanceDisable
```

Field Value

[bool](#)

m_DistanceToObject

```
public float m_DistanceToObject
```

Field Value

[float](#)

how far the Referenced Entity has to be in order for m_DistantDisable to be true

m_DistantDisabled

```
public bool m_DistantDisabled
```

Field Value

[bool](#)

m_Elasticity

```
public float m_Elasticity
```

Field Value

[float](#) ↗

How much the force applied to return each bone to original orientation.

m_EndLength

```
public float m_EndLength
```

Field Value

[float](#) ↗

If End Length is not zero, an extra bone is generated at the end of transform hierarchy.

m_EndOffset

```
public float3 m_EndOffset
```

Field Value

float3

If End Offset is not zero, an extra bone is generated at the end of transform hierarchy.

m_Force

```
public float3 m_Force
```

Field Value

float3

The force apply to bones.

m_FreezeAxis

```
public DOTSDynamicBone.FreezeAxis m_FreezeAxis
```

Field Value

[DOTSDynamicBone.FreezeAxis](#)

Constrain bones to move on specified plane.

m_Friction

```
public float m_Friction
```

Field Value

[float](#) ↗

How much the bones slowed down when collide.

m_GlobalForce

```
public float3 m_GlobalForce
```

Field Value

float3

Constant force applied by the world to all entities like gravity. NOTE: this is only used with DDB Collision

m_Gravity

```
public float3 m_Gravity
```

Field Value

float3

The force apply to bones. Partial force apply to character's initial pose is cancelled out.

m_Inert

```
public float m_Inert
```

Field Value

float ↗

How much character's position change is ignored in physics simulation.

m_LocalGravity

```
public float3 m_LocalGravity
```

Field Value

float3

local gravity of the bone

m_MaxRestLength

```
public float m_MaxRestLength
```

Field Value

[float](#)

NOT IMPLEMENTED, TODO: determine if I want to implement this...

m_ObjectMove

`public ParticleTransform m_ObjectMove`

Field Value

[ParticleTransform](#)

m_ObjectMovePrev

`public ParticleTransform m_ObjectMovePrev`

Field Value

[ParticleTransform](#)

m_ObjectScale

`public float3 m_ObjectScale`

Field Value

`float3`

scale of the Entity

m_ParentAnimatedLocalToRootIndex

`public int m_ParentAnimatedLocalToRootIndex`

Field Value

[int ↗](#)

This is the Root DynamicBone Particle's parent index in the AnimatedLocalToRoot DynamicBuffer

m_ParticleResetOptions

```
[SerializeField]  
public DOTSDynamicBoneParticleResetOptions m_ParticleResetOptions
```

Field Value

[DOTSDynamicBoneParticleResetOptions](#)

m_Radius

```
public float m_Radius
```

Field Value

[float ↗](#)

Each bone can be a sphere to collide with colliders. Radius describe sphere's size.

m_ReferenceObjectEntity

```
public Entity m_ReferenceObjectEntity
```

Field Value

Entity

The Entity to use in the DistantDisable determination

m_ReferenceObjectTransform

```
public ParticleTransform m_ReferenceObjectTransform
```

Field Value

[ParticleTransform](#)

A PArticleTransform that represents the ReferendEntity's Transform

m_RigComponentEntity

```
public Entity m_RigComponentEntity
```

Field Value

Entity

This is the entity Entity that contained the RigComponent. Most likely this will be the same as the m_RootEntity

m_RigRootEntity

```
public Entity m_RigRootEntity
```

Field Value

Entity

m_RootAnimatedLocalToRootIndex

```
public int m_RootAnimatedLocalToRootIndex
```

Field Value

[int ↗](#)

This is the index that the Root Entity's AnimatedLocalToRootIndex index. This will always most likely be 0 since it is the root

m_RootEntity

```
public Entity m_RootEntity
```

Field Value

Entity

This is the Entity if the root transform (not to be confused with the RootBone Entity)

m_RootParentEntity

```
public Entity m_RootParentEntity
```

Field Value

Entity

The root DynamicBone has a parent, and this Entity is that parent.

m_RootParentTransform

```
public ParticleTransform m_RootParentTransform
```

Field Value

[ParticleTransform](#)

m_Stiffness

```
public float m_Stiffness
```

Field Value

[float ↗](#)

How much bone's original orientation are preserved.

m_UpdateDataFrom

```
public DOTSDynamicBoneClass.UpdateDataFrom m_UpdateDataFrom
```

Field Value

[DOTSDynamicBoneClass.UpdateDataFrom](#)

An option to choose where the data updates from. This was implemented

m_UpdateMode

```
public DOTSDynamicBone.UpdateMode m_UpdateMode
```

Field Value

[DOTSDynamicBone.UpdateMode](#)

Internal physics simulation mode.

m_UpdateRate

```
public float m_UpdateRate
```

Field Value

[float ↗](#)

Internal physics simulation rate.

m_Weight

```
public float m_Weight
```

Field Value

[float](#) ↗

weight of bone.

Properties

Default

```
public static DOTSDynamicBone Default { get; }
```

Property Value

[DOTSDynamicBone](#)

Methods

Create(IBaker, DOTSDynamicBoneClass,
List<SkinnedMeshRenderer>, float)

V0.8.3+ Creates a DOTSDynamicBone using the given information

```
public static DOTSDynamicBone Create(IBaker baker, DOTSDynamicBoneClass boneComponent,  
List<SkinnedMeshRenderer> meshRenderers, float maxRestLength = 0.2)
```

Parameters

baker IBaker

`boneComponent` [DOTSDynamicBoneClass](#)

DOTSDynamicBoneClass

`meshRenderers` [List](#)<SkinnedMeshRenderer>

`maxRestLength` [float](#)

Returns

[DOTSDynamicBone](#)

DOTSDynamicBone

Disable(ref bool, ref DynamicBuffer<Particle>, int, int)

Disables the bone and all the given Particles. (NOTE: the particles should be associated with the bones)

```
[BurstCompile]
public static void Disable(ref bool Disabled, ref DynamicBuffer<Particle> m\_Particles, int
start, int end)
```

Parameters

`Disabled` [bool](#)

`m_Particles` DynamicBuffer<Particle>

particles to be disabled

`start` [int](#)

`end` [int](#)

Enable(ref bool, ref DynamicBuffer<Particle>, int, int)

Enables the bone and all the given Particles. (NOTE: the particles should be associated with the bones)

```
[BurstCompile]
public static void Enable(ref bool Disabled, ref DynamicBuffer<Particle> m\_Particles, int
```

```
start, int end)
```

Parameters

Disabled [bool](#)

m_Particles DynamicBuffer<[Particle](#)>

particles to be enabled

start [int](#)

end [int](#)

Enum DOTSDynamicBone.FreezeAxis

Namespace: [DOTSDynamicBone](#)

Assembly: DOTSDynamicBone.dll

```
public enum DOTSDynamicBone.FreezeAxis
```

Fields

None = 0

X = 1

Y = 2

Z = 3

Enum DOTSDynamicBone.UpdateMode

Namespace: [DOTSDynamicBone](#)

Assembly: DOTSDynamicBone.dll

```
public enum DOTSDynamicBone.UpdateMode
```

Fields

AnimatePhysics = 1

Default = 3

Normal = 0

UnscaledTime = 2

Class DOTSDynamicBoneBaker

Namespace: [DOTSDynamicBone](#)

Assembly: DOTSDynamicBone.dll

```
public class DOTSDynamicBoneBaker : Baker<DOTSDynamicBoneComponent>
```

Inheritance

[object](#) ← IBaker ← Baker<[DOTSDynamicBoneComponent](#)> ← DOTSDynamicBoneBaker

Inherited Members

IBaker.GetSceneGUID() , IBaker.GetComponent<T>() , IBaker.GetComponent<T>(Component) ,
IBaker.GetComponent<T>(GameObject) , [IBaker.GetComponents<T>\(List<T>\)](#) ,
[IBaker.GetComponents<T>\(Component, List<T>\)](#) ,
[IBaker.GetComponents<T>\(GameObject, List<T>\)](#) , IBaker.GetComponents<T>()
IBaker.GetComponents<T>(Component) , IBaker.GetComponents<T>(GameObject)
IBaker.GetComponentInParent<T>() , IBaker.GetComponentInParent<T>(Component) ,
IBaker.GetComponentInParent<T>(GameObject) , [IBaker.GetComponentsInParent<T>\(List<T>\)](#) ,
[IBaker.GetComponentsInParent<T>\(Component, List<T>\)](#) ,
[IBaker.GetComponentsInParent<T>\(GameObject, List<T>\)](#) , IBaker.GetComponentsInParent<T>()
IBaker.GetComponentsInParent<T>(Component) , IBaker.GetComponentsInParent<T>(GameObject) ,
IBaker.GetComponentInChildren<T>() , IBaker.GetComponentInChildren<T>(Component) ,
IBaker.GetComponentInChildren<T>(GameObject) , [IBaker.GetComponentsInChildren<T>\(List<T>\)](#) ,
[IBaker.GetComponentsInChildren<T>\(Component, List<T>\)](#) ,
[IBaker.GetComponentsInChildren<T>\(GameObject, List<T>\)](#) , IBaker.GetComponentsInChildren<T>()
IBaker.GetComponentsInChildren<T>(Component) ,
IBaker.GetComponentsInChildren<T>(GameObject) , IBaker.GetParent() , IBaker.GetParent(Component) ,
IBaker.GetParent(GameObject) , IBaker.GetParents() , IBaker.GetParents(Component) ,
IBaker.GetParents(GameObject) , [IBaker.GetParents\(List<GameObject>\)](#) ,
[IBaker.GetParents\(Component, List<GameObject>\)](#) ,
[IBaker.GetParents\(GameObject, List<GameObject>\)](#) , [IBaker.GetChild\(int\)](#) ,
[IBaker.GetChild\(Component, int\)](#) , [IBaker.GetChild\(GameObject, int\)](#) , [IBaker.GetChildren\(bool\)](#) ,
[IBaker.GetChildren\(Component, bool\)](#) , [IBaker.GetChildren\(GameObject, bool\)](#) ,
[IBaker.GetChildren\(List<GameObject>, bool\)](#) ,
[IBaker.GetChildren\(Component, List<GameObject>, bool\)](#) ,
[IBaker.GetChildren\(GameObject, List<GameObject>, bool\)](#) , IBaker.GetChildCount()
IBaker.GetChildCount(Component) , IBaker.GetChildCount(GameObject) , IBaker.GetName()
IBaker.GetName(Component) , IBaker.GetName(GameObject) , IBaker.GetLayer()
IBaker.GetLayer(Component) , IBaker.GetLayer(GameObject) , IBaker.GetTag()

IBaker.GetTag(Component) , IBaker.GetTag(GameObject) , IBaker.GetEntity() ,
IBaker.GetEntity(GameObject) , IBaker.GetEntity(Component) , IBaker.GetEntity(TransformUsageFlags) ,
IBaker.GetEntity(GameObject, TransformUsageFlags) ,
IBaker.GetEntity(Component, TransformUsageFlags) , IBaker.GetEntityWithoutDependency() ,
IBaker.IsActive() , IBaker.IsActive(Component) , IBaker.IsActive(GameObject) ,
IBaker.IsActiveAndEnabled() , IBaker.IsActiveAndEnabled(Component) , IBaker.IsStatic() ,
IBaker.IsStatic(Component) , IBaker.IsStatic(GameObject) , IBaker.IsClient() , IBaker.IsServer() ,
IBaker.DependsOn<T>(T) , IBaker.DependsOnComponentInParent<T>() ,
IBaker.DependsOnComponentInParent<T>(Component) ,
IBaker.DependsOnComponentInParent<T>(GameObject) ,
IBaker.DependsOnComponentsInParent<T>(Component) ,
IBaker.DependsOnComponentsInParent<T>(GameObject) ,
IBaker.DependsOnComponentInChildren<T>() ,
IBaker.DependsOnComponentInChildren<T>(Component) ,
IBaker.DependsOnComponentInChildren<T>(GameObject) ,
IBaker.DependsOnComponentsInChildren<T>() ,
IBaker.DependsOnComponentsInChildren<T>(GameObject) ,
IBaker.DependsOnComponentsInChildren<T>(Component) , IBaker.DependsOnLightBaking() ,
IBaker.AddBlobAsset<T>(ref BlobAssetReference<T>, out Hash128) ,
IBaker.AddBlobAssetWithCustomHash<T>(ref BlobAssetReference<T>, Hash128) ,
IBaker.TryGetBlobAssetReference<T>(Hash128, out BlobAssetReference<T>) ,
IBaker.AddComponent<T>() , IBaker.AddComponent<T>(in T) , IBaker.AddComponent<T>(Entity) ,
IBaker.AddComponent<T>(Entity, in T) , IBaker.AddComponent(ComponentType) ,
IBaker.AddComponent(Entity, ComponentType) , IBaker.AddComponent(in ComponentTypeSet) ,
IBaker.AddComponent(Entity, in ComponentTypeSet) , IBaker.AddComponentObject<T>(T) ,
IBaker.AddComponentObject<T>(Entity, T) , IBaker.AddSharedComponentManaged<T>(T) ,
IBaker.AddSharedComponentManaged<T>(Entity, T) , IBaker.AddSharedComponent<T>(T) ,
IBaker.AddSharedComponent<T>(Entity, T) , IBaker.AddBuffer<T>() , IBaker.AddBuffer<T>(Entity) ,
IBaker.SetComponent<T>(Entity, in T) , [IBaker.GetComponentEnabled<T>\(Entity, bool\)](#) ,
[IBaker.GetComponentEnabled<T>\(bool\)](#) , IBaker.SetSharedComponentManaged<T>(Entity, in T) ,
IBaker.SetSharedComponent<T>(Entity, in T) , IBaker.SetBuffer<T>() , IBaker.SetBuffer<T>(Entity) ,
IBaker.AppendToBuffer<T>(T) , IBaker.AppendToBuffer<T>(Entity, T) , IBaker.CreateAdditionalEntity() ,
[IBaker.CreateAdditionalEntity\(TransformUsageFlags, bool, string\)](#) ,
IBaker.RegisterPrefabForBaking(GameObject) , IBaker.AddTransformUsageFlags(TransformUsageFlags) ,
IBaker.AddTransformUsageFlags(Entity, TransformUsageFlags) , IBaker.IsBakingForEditor() ,
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

Methods

Bake(DOTSDynamicBoneComponent)

Called in the baking process to bake the authoring component

```
public override void Bake(DOTSDynamicBoneComponent authoring)
```

Parameters

authoring [DOTSDynamicBoneComponent](#)

The authoring component to bake

Remarks

This method will be called for every instance of TAuthoringType in order to bake that instance.

Class DOTSDynamicBoneClass

Namespace: [DOTSDynamicBone](#)

Assembly: DOTSDynamicBone.dll

This holds and displays all the data needs to modify to create the DOTSDynamicBone data

```
[Serializable]
public class DOTSDynamicBoneClass
```

Inheritance

[object](#) ← DOTSDynamicBoneClass

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

Remarks

This holds and displays all the data needs to modify to create the DOTSDynamicBone data

Constructors

DOTSDynamicBoneClass(bool, RigComponent)

Default Initializer

```
public DOTSDynamicBoneClass(bool useNaturalColliders = false, RigComponent rigComponent
= null)
```

Parameters

useNaturalColliders [bool](#)

rigComponent [RigComponent](#)

Fields

Name

```
[Tooltip("Just The name of the root bone (so you can see it in the editor easier, changing  
it won't break anything but may confuse you)")]  
public string Name
```

Field Value

[string](#)

eParticles

```
public List<DOTSDynamicBoneClass.EditorParticle> eParticles
```

Field Value

[List](#) <[DOTSDynamicBoneClass.EditorParticle](#)>

m_BoneTotalLength

```
public float m_BoneTotalLength
```

Field Value

[float](#)

m_Colliders

```
[Tooltip("Collider objects interact with the bones.")]  
public List<DOTSDynamicBoneCollider> m_Colliders
```

Field Value

[List](#) <[DOTSDynamicBoneCollider](#)>

m_CullingData

```
[Tooltip("If you want the Particle Physics to be Culled, then fill out the Data here.")]
public EditorDOTSDynnamicBoneCullingSettings m_CullingData
```

Field Value

[EditorDOTSDynnamicBoneCullingSettings](#)

m_Damping

```
[Tooltip("How much the bones slowed down.")]
[Range(0, 1)]
public float m_Damping
```

Field Value

[float](#)

m_DampingDistrib

```
[Tooltip("How much the bones slowed down over a distribution curve.")]
public AnimationCurve m_DampingDistrib
```

Field Value

[AnimationCurve](#)

m_Disable

```
[Tooltip("This disables all Particles within the DOTSDynamicBone from taking part in
Particle Physic Simulations.")]
public bool m_Disable
```

Field Value

[bool](#)

m_Elasticity

```
[Tooltip("How much the force applied to return each bone to original orientation.")]
[Range(0, 1)]
public float m_Elasticity
```

Field Value

[float](#)

m_ElasticityDistrib

```
[Tooltip("How much the force applied to return each bone to original orientation over a
distribution curve.")]
public AnimationCurve m_ElasticityDistrib
```

Field Value

AnimationCurve

m_EndLength

```
[Tooltip("If End Length is not zero, an extra bone is generated at the end of
transform hierarchy.")]
public float m_EndLength
```

Field Value

[float](#)

m_EndOffset

```
[Tooltip("If End Offset is not zero, an extra bone is generated at the end of  
transform hierarchy.")]  
public Vector3 m_EndOffset
```

Field Value

Vector3

m_Exclusions

```
[Tooltip("Bones excluded from physics simulation.")]  
public List<ExcludeEntityFromSimulation> m_Exclusions
```

Field Value

[List](#) <[ExcludeEntityFromSimulation](#)>

m_Force

```
[Tooltip("The force apply to bones.")]  
public Vector3 m_Force
```

Field Value

Vector3

m_FreezeAxis

```
[Tooltip("Constrain bones to move on specified plane.")]  
public DOTSDynamicBone.FreezeAxis m_FreezeAxis
```

Field Value

[DOTSDynamicBone.FreezeAxis](#)

m_Friction

```
[Tooltip("How much the bones slowed down when collide.")]
public float m_Friction
```

Field Value

[float](#)

m_FrictionDistrib

```
[Tooltip("How much the bones slowed down when collide over a distribution curve.")]
public AnimationCurve m_FrictionDistrib
```

Field Value

AnimationCurve

m_GlobalForce

```
[Tooltip("Constant force applied by the world to all entities like gravity. NOTE: this is
only used with DDB Collision")]
public Vector3 m_GlobalForce
```

Field Value

Vector3

m_Gravity

```
[Tooltip("The force apply to bones. Partial force apply to character's initial pose is
cancelled out.")]
public Vector3 m_Gravity
```

Field Value

Vector3

m_Inert

```
[Tooltip("How much character's position change is ignored in physics simulation.")]
[Range(0, 1)]
public float m_Inert
```

Field Value

[float](#)

m_InertDistrib

```
[Tooltip("How much character's position change is ignored in physics simulation over a
distribution curve.")]
public AnimationCurve m_InertDistrib
```

Field Value

AnimationCurve

m_Radius

```
[Tooltip("Each bone can be a sphere to collide with colliders. Radius describe
sphere's size.")]
public float m_Radius
```

Field Value

[float](#)

m_RadiusDistrib

```
[Tooltip("Each bone can be a sphere to collide with colliders. Radius describe sphere's size  
over a distribution curve.")]  
public AnimationCurve m_RadiusDistrib
```

Field Value

AnimationCurve

m_ResetOptions

```
[Tooltip("This is a bunch of options used to reset certain parameters in the  
DOTSDynamicBone systems.")]  
public DOTSDynamicBoneParticleResetOptions m_ResetOptions
```

Field Value

[DOTSDynamicBoneParticleResetOptions](#)

m_RigComponent

```
[Tooltip("This should be the GameObject this script is attached to")]  
public RigComponent m_RigComponent
```

Field Value

[RigComponent](#)

m_RootBone

```
[Tooltip("The root of the transform hierarchy to apply physics.")]  
public Transform m_RootBone
```

Field Value

Transform

m_Stiffness

```
[Tooltip("How much bone's original orientation are preserved.")]
[Range(0, 1)]
public float m_Stiffness
```

Field Value

[float](#)

m_StiffnessDistrib

```
[Tooltip("How much bone's original orientation are preserved over a distribution curve.")]
public AnimationCurve m_StiffnessDistrib
```

Field Value

AnimationCurve

m_UpdateDataFrom

```
[Tooltip("An option to choose where the data updates from. This was implemented but due to
me finding a lack of use it's options were removed leaving only AnimatedLocalToWorld. Maybe
I'll add it back but only at request and reason.")]
public DOTSDynamicBoneClass.UpdateDataFrom m_UpdateDataFrom
```

Field Value

[DOTSDynamicBoneClass.UpdateDataFrom](#)

m_UpdateLocalPositions

```
[Tooltip("Set to true to update the Local Postoitons for each Particle's
Corresponding Entity")]
```

```
public bool m_UpdateLocalPositions
```

Field Value

[bool](#) ↗

m_UpdateLocalRotations

```
[Tooltip("Set to true to update the Local Rotations for each Particle's  
Corresponding Entity")]  
public bool m_UpdateLocalRotations
```

Field Value

[bool](#) ↗

m_UpdateMode

```
[Tooltip("Internal physics simulation type.")]  
public DOTSDynamicBone.UpdateMode m_UpdateMode
```

Field Value

[DOTSDynamicBone.UpdateMode](#)

m_UpdateRate

```
[Tooltip("Internal physics simulation rate.")]  
public float m_UpdateRate
```

Field Value

[float](#) ↗

m_UpdateTransformAsIfNoParent

```
[Tooltip("Set to true if entities will be un-parented during conversion. If using  
Unity.Physics and are using Natural Collision then this should be set to true.")]  
public bool m_UpdateTransformAsIfNoParent
```

Field Value

[bool](#) ↗

m_UpdateTransformButFlipLocalAndWorld

```
[Tooltip("Set to true to swap the Local and World Transofm Updates for each Particle's  
Corresponding Entity")]  
public bool m_UpdateTransformButFlipLocalAndWorld
```

Field Value

[bool](#) ↗

m_UpdateWorldPositions

```
[Tooltip("Set to true to update the World Positions for each Particle's  
Corresponding Entity")]  
public bool m_UpdateWorldPositions
```

Field Value

[bool](#) ↗

m_UpdateWorldRotations

```
[Tooltip("Set to true to update the World Rotatoins for each Particle's  
Corresponding Entity")]  
public bool m_UpdateWorldRotations
```

Field Value

[bool](#) ↗

m_UseAnimatedLocalToRoot

```
[Tooltip("Set to true if the entity is Animated using the Package")]
public bool m_UseAnimatedLocalToRoot
```

Field Value

[bool](#) ↗

m_Weight

```
public float m_Weight
```

Field Value

[float](#) ↗

Properties

Default

```
public static DOTSDynamicBoneClass Default { get; }
```

Property Value

[DOTSDynamicBoneClass](#)

DefaultWithNaturalCollisions

```
public static DOTSDynamicBoneClass DefaultWithNaturalCollisions { get; }
```

Property Value

[DOTSDynamicBoneClass](#)

Methods

AddDOTSDynamicBoneCalculationCulling(EntityManager, Entity, ref DOTSDynamicBone, Entity, float)

```
public static void AddDOTSDynamicBoneCalculationCulling(EntityManager em, Entity e, ref DOTSDynamicBone bone, Entity reference_entity, float distance)
```

Parameters

em EntityManager

e Entity

bone [DOTSDynamicBone](#)

reference_entity Entity

distance float

AddDOTSDynamicBoneColliderComponentsToEntity(EntityManager, Entity, bool, bool)

adds some DynamicBoneColliderComponents to the given entity

```
public static void AddDOTSDynamicBoneColliderComponentsToEntity(EntityManager em, Entity ddb_entity, bool useNaturalColliders, bool excludePhysics)
```

Parameters

em EntityManager

EntityManager

ddb_entity Entity

Entity

`useNaturalColliders bool`

`excludePhysics bool`

AddDOTSDynamicBoneColliderComponentsToEntity(IBaker, Entity, bool, bool)

Adds DOTSDynamicBone collider components to the DOTSDynamicBone Entity

```
public static void AddDOTSDynamicBoneColliderComponentsToEntity(IBaker baker, Entity  
ddb_entity, bool useNaturalColliders, bool excludePhysics)
```

Parameters

`baker` IBaker

baker used during entity conversion

`ddb_entity` Entity

the DOTSDynamicBone entity

`useNaturalColliders bool`

BoneComponent.UseNaturalcolliders

`excludePhysics bool`

set to true to not include the DOTSDynamicBoneCollider_BufferElement buffer

InitializeDefault(bool, RigComponent)

```
public void InitializeDefault(bool useNaturalColliders = false, RigComponent rigComponent  
= null)
```

Parameters

`useNaturalColliders bool`

`rigComponent` [RigComponent](#)

RemoveDOTSDynamicBoneCalculationCulling(EntityManager, Entity)

```
public static void RemoveDOTSDynamicBoneCalculationCulling(EntityManager em, Entity e)
```

Parameters

`em` EntityManager

`e` Entity

RemoveDOTSDynamicBoneCalculationCulling(EntityManager, Entity, ref DOTSDynamicBone)

```
public static void RemoveDOTSDynamicBoneCalculationCulling(EntityManager em, Entity e, ref DOTSDynamicBone bone)
```

Parameters

`em` EntityManager

`e` Entity

`bone` [DOTSDynamicBone](#)

SetDOTSDynamicBoneCalculationCulling(ref DOTSDynamicBone, Entity, float, bool)

```
public static void SetDOTSDynamicBoneCalculationCulling(ref DOTSDynamicBone bone, Entity reference_entity, float distance, bool ActivateCulling = true)
```

Parameters

bone [DOTSDynamicBone](#)

reference_entity Entity

distance float

ActivateCulling bool

ToDOTSDynamicBone(IBaker, RigComponent, Entity, ref DynamicBuffer<Particle>)

Creates a valid DOTSDynamicBone. NOTE: This call will invalidate Particle DynamicBuffer.

```
public DOTSDynamicBone ToDOTSDynamicBone(IBaker baker, RigComponent RigComponent, Entity DOTSDynamicBoneEntity, ref DynamicBuffer<Particle> m_Particles)
```

Parameters

baker IBaker

RigComponent [RigComponent](#)

RigComponent to be used on the conversion

DOTSDynamicBoneEntity Entity

Entity where the root DOTSDynamicBone will affect

m_Particles DynamicBuffer<[Particle](#)>

Buffer of particles to populate with data.

Returns

[DOTSDynamicBone](#)

DOTSDynamicBone

ToDOTSDynamicBone(IBaker, Entity, ref DynamicBuffer<Particle>)

Creates a valid DOTSDynamicBone. NOTE: This call will invalidate Particle DynamicBuffer.

```
public DOTSDynamicBone ToDOTSDynamicBone(IBaker baker, Entity DOTSDynamicBoneEntity, ref  
DynamicBuffer<Particle> m_Particles)
```

Parameters

baker IBaker

DOTSDynamicBoneEntity Entity

Entity where the root DOTSDynamicBone will affect

m_Particles DynamicBuffer<[Particle](#)>

Buffer of particles to populate with data.

Returns

[DOTSDynamicBone](#)

DOTSDynamicBone

UpdateName()

Updates the name of the Name property. (Used only in the Editor)

```
public void UpdateName()
```

Class DOTSDynamicBoneClass.EditorParticle

Namespace: [DOTSDynamicBone](#)

Assembly: DOTSDynamicBone.dll

```
[Serializable]
public class DOTSDynamicBoneClass.EditorParticle
```

Inheritance

[object](#) ← DOTSDynamicBoneClass.EditorParticle

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

Fields

EEFS

```
[SerializeField]
public ExcludeEntityFromSimulation EEFS
```

Field Value

[ExcludeEntityFromSimulation](#)

parentIndex

```
[SerializeField]
public int parentIndex
```

Field Value

[int](#)

t

```
[SerializeField]  
public Transform t
```

Field Value

Transform

Enum DOTSDynamicBoneClass.UpdateDataFrom

Namespace: [DOTSDynamicBone](#)

Assembly: DOTSDynamicBone.dll

```
public enum DOTSDynamicBoneClass.UpdateDataFrom
```

Fields

AnimatedLocalToRoot = 2

AnimatedLocalToWorld = 0

None = 1

Class DOTSDynamicBoneComponent

Namespace: [DOTSDynamicBone](#)

Assembly: DOTSDynamicBone.dll

This is a component that creates a single DOTSDynamicBone onto an Entity. If you want more please use the DOTSDynamicBonesComponent or the IndependantDOTSDynamicBoneComponent

```
public class DOTSDynamicBoneComponent : MonoBehaviour
```

Inheritance

[object](#) ← Object ← Component ← Behaviour ← MonoBehaviour ← DOTSDynamicBoneComponent

Inherited Members

MonoBehaviour.IsInvoking() , MonoBehaviour.CancelInvoke() , [MonoBehaviour.Invoke\(string, float\)](#) ,
[MonoBehaviour.InvokeRepeating\(string, float, float\)](#) , [MonoBehaviour.CancelInvoke\(string\)](#) ,
[MonoBehaviour.IsInvoking\(string\)](#) , [MonoBehaviour.StartCoroutine\(string\)](#) ,
[MonoBehaviour.StartCoroutine\(string, object\)](#) , [MonoBehaviour.StartCoroutine\(IEnumerator\)](#) ,
[MonoBehaviour.StartCoroutine_Auto\(IEnumerator\)](#) , [MonoBehaviour.StopCoroutine\(IEnumerator\)](#) ,
MonoBehaviour.StopCoroutine(Coroutine) , [MonoBehaviour.StopCoroutine\(string\)](#) ,
MonoBehaviour.StopAllCoroutines() , [MonoBehaviour.print\(object\)](#) ,
MonoBehaviour.destroyCancellationToken , MonoBehaviour.useGUILayout ,
MonoBehaviour.runInEditMode , Behaviour.enabled , Behaviour.isActiveAndEnabled ,
[Component.GetComponent\(Type\)](#) , Component.GetComponent<T>() ,
[Component.TryGetComponent\(Type, out Component\)](#) , Component.TryGetComponent<T>(out T) ,
[Component.GetComponent\(string\)](#) , [Component.GetComponentInChildren\(Type, bool\)](#) ,
[Component.GetComponentInChildren\(Type\)](#) , [Component.GetComponentInChildren<T>\(bool\)](#) ,
Component.GetComponentInChildren<T>() , [Component.GetComponentsInChildren\(Type, bool\)](#) ,
[Component.GetComponentsInChildren\(Type\)](#) , [Component.GetComponentsInChildren<T>\(bool\)](#) ,
[Component.GetComponentsInChildren<T>\(bool, List<T>\)](#) ,
Component.GetComponentsInChildren<T>() , [Component.GetComponentsInChildren<T>\(List<T>\)](#) ,
[Component.GetComponentInParent\(Type, bool\)](#) , [Component.GetComponentInParent\(Type\)](#) ,
[Component.GetComponentInParent<T>\(bool\)](#) , Component.GetComponentInParent<T>() ,
[Component.GetComponentsInParent\(Type, bool\)](#) , [Component.GetComponentsInParent\(Type\)](#) ,
[Component.GetComponentsInParent<T>\(bool\)](#) ,
[Component.GetComponentsInParent<T>\(bool, List<T>\)](#) , Component.GetComponentsInParent<T>() ,
[Component.GetComponents\(Type\)](#) , [Component.GetComponents\(Type, List<Component>\)](#) ,
[Component.GetComponents<T>\(List<T>\)](#) , Component.GetComponents<T>() ,
Component.GetComponentIndex() , [Component.CompareTag\(string\)](#) ,

[Component.SendMessageUpwards\(string, object, SendMessageOptions\)](#) ,
[Component.SendMessageUpwards\(string, object\)](#) , [Component.SendMessageUpwards\(string\)](#) ,
[Component.SendMessageUpwards\(string, SendMessageOptions\)](#) ,
[Component.SendMessage\(string, object\)](#) , [Component.SendMessage\(string\)](#) ,
[Component.SendMessage\(string, object, SendMessageOptions\)](#) ,
[Component.SendMessage\(string, SendMessageOptions\)](#) ,
[Component.BroadcastMessage\(string, object, SendMessageOptions\)](#) ,
[Component.BroadcastMessage\(string, object\)](#) , [Component.BroadcastMessage\(string\)](#) ,
[Component.BroadcastMessage\(string, SendMessageOptions\)](#) , Component.transform ,
Component.gameObject , Component.tag , Object.GetInstanceID() , Object.GetHashCode() ,
[Object.Equals\(object\)](#) , Object.InstantiateAsync<T>(T) , Object.InstantiateAsync<T>(T, Transform) ,
Object.InstantiateAsync<T>(T, Vector3, Quaternion) ,
Object.InstantiateAsync<T>(T, Transform, Vector3, Quaternion) , [Object.InstantiateAsync<T>\(T, int\)](#) ,
[Object.InstantiateAsync<T>\(T, int, Transform\)](#) ,
[Object.InstantiateAsync<T>\(T, int, Vector3, Quaternion\)](#) ,
[Object.InstantiateAsync<T>\(T, int, ReadOnlySpan<Vector3>, ReadOnlySpan<Quaternion>\)](#) ,
[Object.InstantiateAsync<T>\(T, int, Transform, Vector3, Quaternion\)](#) ,
[Object.InstantiateAsync<T>\(T, int, Transform, ReadOnlySpan<Vector3>, ReadOnlySpan<Quaternion>\)](#) ,
Object.Instantiate(Object, Vector3, Quaternion) ,
Object.Instantiate(Object, Vector3, Quaternion, Transform) , Object.Instantiate(Object) ,
Object.Instantiate(Object, Scene) , Object.Instantiate(Object, Transform) ,
[Object.Instantiate\(Object, Transform, bool\)](#) , Object.Instantiate<T>(T) ,
Object.Instantiate<T>(T, Vector3, Quaternion) ,
Object.Instantiate<T>(T, Vector3, Quaternion, Transform) , Object.Instantiate<T>(T, Transform) ,
[Object.Instantiate<T>\(T, Transform, bool\)](#) , [Object.Destroy\(Object, float\)](#) , Object.Destroy(Object) ,
[Object.DestroyImmediate\(Object, bool\)](#) , Object.DestroyImmediate(Object) ,
[Object.FindObjectsOfType\(Type\)](#) , [Object.FindObjectsOfType\(Type, bool\)](#) ,
[Object.FindObjectsByType\(Type, FindObjectsSortMode\)](#) ,
[Object.FindObjectsByType\(Type, FindObjectsInactive, FindObjectsSortMode\)](#) ,
Object.DontDestroyOnLoad(Object) , [Object.DestroyObject\(Object, float\)](#) ,
Object.DestroyObject(Object) , [Object.FindSceneObjectsOfType\(Type\)](#) ,
[Object.FindObjectsOfTypeIncludingAssets\(Type\)](#) , Object.FindObjectsOfType<T>() ,
Object.FindObjectsByType<T>(FindObjectsSortMode) , [Object.FindObjectsOfType<T>\(bool\)](#) ,
Object.FindObjectsByType<T>(FindObjectsInactive, FindObjectsSortMode) ,
ObjectFindObjectOfType<T>() , [Object.FindObjectType<T>\(bool\)](#) ,
Object.FindFirstObjectByType<T>() , Object.FindAnyObjectByType<T>() ,
Object.FindFirstObjectByType<T>(FindObjectsInactive) ,
Object.FindAnyObjectByType<T>(FindObjectsInactive) , [Object.FindObjectsOfTypeAll\(Type\)](#) ,
[Object.FindObjectType\(Type\)](#) , [Object.FindFirstObjectByType\(Type\)](#) ,
[Object.FindAnyObjectByType\(Type\)](#) , [Object.FindObjectType\(Type, bool\)](#) ,

```
Object.FindFirstObjectByType\(Type, FindObjectsInactive\) ,  
Object.FindAnyObjectByType\(Type, FindObjectsInactive\) , Object.ToString() , Object.name ,  
Object.hideFlags , object.Equals\(object, object\) , object.GetType\(\) , object.MemberwiseClone\(\) ,  
object.ReferenceEquals\(object, object\)
```

Remarks

This is a component that creates a single DOTSDynamicBone onto an Entity. If you want more please use the DOTSDynamicBonesComponent or the IndependantDOTSDynamicBoneComponent

Fields

ApplyToAll

```
[Header("Apply To All Section")]
[Tooltip("Set this to have all valid data in the \"Apply To All Section\" applied to every
DOTSDynamicBoneClass in the DOTS Dynamic Bone List.")]
public bool ApplyToAll
```

Field Value

[bool](#)

m_ATAColliders

```
[Tooltip("A List of DOTSDynamicBoneCollider to every DOTSDynamicBoneClass if it doesn't
already contain it.")]
public List<DOTSDynamicBoneCollider> m_ATAColliders
```

Field Value

[List](#)<[DOTSDynamicBoneCollider](#)>

m_ATACullingData

```
[Tooltip("<value> Culling Data that will Applied to each Entity")]
[SerializeField]
```

```
public EditorDOTSDynnamicBoneCullingSettings m_ATACullingData
```

Field Value

[EditorDOTSDynnamicBoneCullingSettings](#)

m_ATAExclusions

```
[Tooltip("A List of ExcludeEntityFromSimulation to every DOTSDynamicBoneClass if it doesn't already contain it.")]  
public List<ExcludeEntityFromSimulation> m_ATAExclusions
```

Field Value

[List](#) <[ExcludeEntityFromSimulation](#)>

m_ATARadius

```
[Tooltip("Set the radius of every DOTSDynamicBoneClass. NOTE: the radius (m_ATARadius.y) will only be set if m_ATARadius.x is set to 1.0f")]  
public float2 m_ATARadius
```

Field Value

float2

m_DOTSDynamicBones

```
[Tooltip("This is a DOTSDynamicBoneClass used for Creating a DOTSDynamicBone")]  
public List<DOTSDynamicBoneClass> m_DOTSDynamicBones
```

Field Value

[List](#) <[DOTSDynamicBoneClass](#)>

Methods

Start()

```
public void Start()
```

Struct DOTSDynamicBoneCulling_Tag

Namespace: [DOTSDynamicBone](#)

Assembly: DOTSDynamicBone.dll

```
public struct DOTSDynamicBoneCulling_Tag : IComponentData, IQueryTypeParameter
```

Implements

IComponentData, IQueryTypeParameter

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Struct DOTSDynamicBoneEntity

Namespace: [DOTSDynamicBone](#)

Assembly: DOTSDynamicBone.dll

This is a shared component that links all particles to it's respective DOTSDynamicBone

```
[BurstCompile]
public struct DOTSDynamicBoneEntity : ISharedComponentData, IQueryTypeParameter
```

Implements

ISharedComponentData, IQueryTypeParameter

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Fields

entity

This is the Entity that has the DOTSDynamicBone related data

```
public Entity entity
```

Field Value

Entity

Struct DOTSDynamicBoneInitializerTag

Namespace: [DOTSDynamicBone](#)

Assembly: DOTSDynamicBone.dll

```
public struct DOTSDynamicBoneInitializerTag : IComponentData, IQueryTypeParameter
```

Implements

IComponentData, IQueryTypeParameter

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Struct DOTSDynamicBoneParticle

Namespace: [DOTSDynamicBone](#)

Assembly: DOTSDynamicBone.dll

```
public struct DOTSDynamicBoneParticle : IComponentData, IQueryTypeParameter
```

Implements

IComponentData, IQueryTypeParameter

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Fields

DOTSDynamicBoneEntity

```
public Entity DOTSDynamicBoneEntity
```

Field Value

Entity

boneIndex

```
public int boneIndex
```

Field Value

[int](#)

particleIndex

```
public int particleIndex
```

Field Value

[int ↗](#)

singleBone

```
public bool singleBone
```

Field Value

[bool ↗](#)

Struct DOTSPlane

Namespace: [DOTSDynamicBone](#)

Assembly: DOTSDynamicBone.dll

A DOTS representation of a Unity Plane

```
[BurstCompile]
public struct DOTSPlane : IComponentData, IQueryTypeParameter
```

Implements

IComponentData, IQueryTypeParameter

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Remarks

A DOTS representation of a Unity Plane that was made before I starting using the Unity.Physics.Plane....I'm still using this plane though cause it works

Constructors

DOTSPlane(float3, float)

```
public DOTSPlane(float3 inNormal, float distance)
```

Parameters

inNormal float3

distance [float](#)

DOTSPlane(float3, float3)

```
public DOTSPlane(float3 inNormal, float3 inPoint)
```

Parameters

inNormal float3

inPoint float3

DOTSPlane(float3, float3, float3)

```
public DOTSPlane(float3 a, float3 b, float3 c)
```

Parameters

a float3

b float3

c float3

Properties

distance

```
public float distance { get; set; }
```

Property Value

[float](#)

flipped

```
public DOTSPlane flipped { get; }
```

Property Value

[DOTSPlane](#)

normal

```
public float3 normal { get; set; }
```

Property Value

float3

Methods

ClosestPointOnPlane(float3)

```
public float3 ClosestPointOnPlane(float3 point)
```

Parameters

point float3

Returns

float3

Flip()

```
public void Flip()
```

GetDistanceToPoint(float3)

```
public float GetDistanceToPoint(float3 point)
```

Parameters

point float3

Returns

[float](#)

GetDistanceToPoint(in float3, in float, in float3, out float)

[BurstCompile]

```
public static void GetDistanceToPoint(in float3 m_Normal, in float m_Distance, in float3 point, out float output)
```

Parameters

m_Normal float3

m_Distance [float](#)

point float3

output [float](#)

GetSide(float3)

```
public bool GetSide(float3 point)
```

Parameters

point float3

Returns

[bool](#)

SameSide(float3, float3)

```
public bool SameSide(float3 inPt0, float3 inPt1)
```

Parameters

inPt0 float3

inPt1 float3

Returns

[bool](#)

Set3Points(float3, float3, float3)

```
public void Set3Points(float3 a, float3 b, float3 c)
```

Parameters

a float3

b float3

c float3

SetNormalAndPosition(float3, float3)

```
public void SetNormalAndPosition(float3 inNormal, float3 inPoint)
```

Parameters

inNormal float3

inPoint float3

Translate(DOTSPlane, Vector3)

```
public static DOTSPlane Translate(DOTSPlane plane, Vector3 translation)
```

Parameters

plane [DOTSPlane](#)

translation Vector3

Returns

[DOTSPlane](#)

Translate(Vector3)

```
public void Translate(Vector3 translation)
```

Parameters

translation Vector3

Struct EditorDOTSDynamicBoneCullingSettings

Namespace: [DOTSDynamicBone](#)

Assembly: DOTSDynamicBone.dll

```
[Serializable]
public struct EditorDOTSDynamicBoneCullingSettings
```

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Fields

Reference

```
public Transform Reference
```

Field Value

Transform

Transform that represents the Entity that the distance will be compared to.

m_Enable

```
public bool m_Enable
```

Field Value

bool

set this to true Disable physics simulation automatically if Entity is far from the Referencee Entity.

m_EnableGizmo

```
public bool m_EnableGizmo
```

Field Value

[bool](#) ↗

maxDisance

```
public float maxDisance
```

Field Value

[float](#) ↗

Maximumn Distance the DOTS Dynamic Bone can be from the Reference before Particle Physics get disabled.

Class ExcludeEntityFromSimulation

Namespace: [DOTSDynamicBone](#)

Assembly: DOTSDynamicBone.dll

This class holds data related to simulation exclusion

```
[Serializable]
public class ExcludeEntityFromSimulation : IEquatable<ExcludeEntityFromSimulation>
```

Inheritance

[object](#) ← ExcludeEntityFromSimulation

Implements

[IEquatable](#)<[ExcludeEntityFromSimulation](#)>

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

Fields

DisableChildrenFromParticlePhysics

```
[Tooltip("Set this to true to also disable the children of this transform from particle
physics simulation")]
public bool DisableChildrenFromParticlePhysics
```

Field Value

[bool](#)

DisableParticlePhysics

```
[Tooltip("Set this to true to exclude from normal particle physics simulations")]
public bool DisableParticlePhysics
```

Field Value

[bool](#) ↗

ExcludeChildrenFromCollision

```
[Tooltip("Set this to true to exclude the transform's children from collision as well")]
public bool ExcludeChildrenFromCollision
```

Field Value

[bool](#) ↗

ExcludeChildrenFromParticleCreation

```
public bool ExcludeChildrenFromParticleCreation
```

Field Value

[bool](#) ↗

ExcludeFromCollision

```
[Tooltip("Set this to true to exlude from Physic Collisions")]
public bool ExcludeFromCollision
```

Field Value

[bool](#) ↗

ExcludeFromParticleCreation

```
public bool ExcludeFromParticleCreation
```

Field Value

[bool](#) ↗

transform

```
[Tooltip("Transform to exclude")]
public Transform transform
```

Field Value

Transform

Properties

Null

```
public static ExcludeEntityFromSimulation Null { get; }
```

Property Value

[ExcludeEntityFromSimulation](#)

Methods

Copy()

```
public ExcludeEntityFromSimulation Copy()
```

Returns

[ExcludeEntityFromSimulation](#)

Equals(ExcludeEntityFromSimulation)

```
public bool Equals(ExcludeEntityFromSimulation other)
```

Parameters

other [ExcludeEntityFromSimulation](#)

Returns

[bool](#)

Struct LinkedDOTSDynamicBones

Namespace: [DOTSDynamicBone](#)

Assembly: DOTSDynamicBone.dll

Used in a DOTSDynamicBone Entity to link created Bones with the Original Entity.

```
public struct LinkedDOTSDynamicBones : IBufferElementData
```

Implements

IBufferElementData

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Remarks

Used in a DOTSDynamicBone Entity to link created Bones with the Original Entity.

Fields

DOTSDynamicBoneEntity

```
public Entity DOTSDynamicBoneEntity
```

Field Value

Entity

Struct Particle

Namespace: [DOTSDynamicBone](#)

Assembly: DOTSDynamicBone.dll

This contains all the data related to a section the DOTSDynamicBone. Here the data is store and calculations are performed on the data

```
[Serializable]
[BurstCompile]
public struct Particle : IBufferElementData
```

Implements

IBufferElementData

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Remarks

This contains all the data related to a section the DOTSDynamicBone. Here the data is store and calculations are performed on the data

Fields

AnimatedLocalToRootIndex

```
public int AnimatedLocalToRootIndex
```

Field Value

[int](#)

The index relative to the Core Entity's AnimatedLocalToRoot and AnimatedLocalToWorld DynamicBuffer

h_Damping

```
public float h_Damping
```

Field Value

[float](#) ↗

These are temporary values used when switching from a disabled/enabled mode

h_Elasticity

```
public float h_Elasticity
```

Field Value

[float](#) ↗

These are temporary values used when switching from a disabled/enabled mode

h_Friction

```
public float h_Friction
```

Field Value

[float](#) ↗

These are temporary values used when switching from a disabled/enabled mode

h_Inert

```
public float h_Inert
```

Field Value

[float](#)

These are temporary values used when switching from a disabled/enabled mode

h_Stiffness

`public float h_Stiffness`

Field Value

[float](#)

These are temporary values used when switching from a disabled/enabled mode

m_AnimatedLocalToWorldTransform

`public ParticleTransform m_AnimatedLocalToWorldTransform`

Field Value

[ParticleTransform](#)

The "Transform" of the Entity the Particle represent that has been converted to AnimatedLocalToRoot space.

m_BoneLength

`public float m_BoneLength`

Field Value

[float](#)

length of the bone at this particle

m_Bound

```
public TransformsExtensions.Bound m_Bound
```

Field Value

[TransformsExtensions.Bound](#)

type of bound for collision calculations.

m_Damping

```
public float m_Damping
```

Field Value

[float](#) ↗

How much the particle slowed down.

m_Elasticity

```
public float m_Elasticity
```

Field Value

[float](#) ↗

How much the force applied to return each particle to original orientation.

m_EndOffset

```
public float3 m_EndOffset
```

Field Value

float3

Used for internal calculations(do not touch unless you know what your doing)

m_ExcludeFromCollision

```
public bool m_ExcludeFromCollision
```

Field Value

[bool](#)

m_ExcludeFromParticlePhysics

```
public bool m_ExcludeFromParticlePhysics
```

Field Value

[bool](#)

m_Friction

```
public float m_Friction
```

Field Value

[float](#)

How much the particle slowed down when collide.

m_Inert

```
public float m_Inert
```

Field Value

[float](#) ↗

How much particle's position change is ignored in physics simulation.

m_InitLocalPosition

```
public float3 m_InitLocalPosition
```

Field Value

float3

Initial Local Position Used for internal calculations(do not touch unless you know what your doing)

m_InitLocalRotation

```
public quaternion m_InitLocalRotation
```

Field Value

quaternion

Used for internal calculations(do not touch unless you know what your doing)

m_InitWorldPosition

```
public float3 m_InitWorldPosition
```

Field Value

float3

initial position of particle used for internal calculations(do not touch unless you know what your doing)

m_InitWorldRotation

```
public quaternion m_InitWorldRotation
```

Field Value

quaternion

initial rotation of particle used for internal calculations(do not touch unless you know what your doing)

m_LastExcludeFromParticlePhysics

```
public bool m_LastExcludeFromParticlePhysics
```

Field Value

[bool](#)

m_LastParentParticleIndex

```
public int m_LastParentParticleIndex
```

Field Value

[int](#)

The last known index of the parent Particle (-1 = the parent info is in the bone data)

m_ParentEntity

```
public Entity m_ParentEntity
```

Field Value

Entity

the parent entity of this particle.

m_ParentParticleIndex

```
public int m_ParentParticleIndex
```

Field Value

[int](#) ↗

The index of the parent Particle (-1 = the parent info is in the bone data)

m_Position

```
public float3 m_Position
```

Field Value

float3

Current Position Used for internal calculations (do not touch unless you know what your doing)

m_PrevPosition

```
public float3 m_PrevPosition
```

Field Value

float3

Previous Position Used for internal calculations(do not touch unless you know what your doing)

m_PrevRotation

```
public quaternion m_PrevRotation
```

Field Value

quaternion

Previous Rotation Used for internal calculations (do not touch unless you know what your doing)

m_Radius

```
public float m_Radius
```

Field Value

[float](#) ↗

Each particle can be a sphere to collide with colliders. Radius describe sphere's size.

m_Rotation

```
public quaternion m_Rotation
```

Field Value

quaternion

Rotation Used for internal calculations (do not touch unless you know what your doing)

m_Stiffness

```
public float m_Stiffness
```

Field Value

[float](#) ↗

How much particle's original orientation are preserved.

m_Transform

```
public ParticleTransform m_Transform
```

Field Value

[ParticleTransform](#)

The "Transform" of the Entity the Particle represents

m_TransformEntity

```
public Entity m_TransformEntity
```

Field Value

Entity

Entity that the Particle represents

m_TransformLocalToWorldMatrix

```
public Matrix4x4 m_TransformLocalToWorldMatrix
```

Field Value

Matrix4x4

the "Transform's" LocalToWorld matrix

m_isCollide

```
public bool m_isCollide
```

Field Value

[bool](#)

Properties

Null

```
public static Particle Null { get; }
```

Property Value

[Particle](#)

returns a "Null" particle

Methods

CalculateParameters()

```
[BurstCompile]  
public void CalculateParameters()
```

CalculateParameters(ref Particle)

Calculates the parameters based on internal data values

```
[BurstCompile]  
public static void CalculateParameters(ref Particle p)
```

Parameters

[p Particle](#)

reference of a Particle

CheckExclude(ref List<Particle>)

```
public void CheckExclude(ref List<Particle> particles)
```

Parameters

particles [List](#)<[Particle](#)>

CheckExclude(in int, ref List<Particle>)

Handles Including/Excluding particles within a List

```
public static void CheckExclude(in int pIndex, ref List<Particle> particles)
```

Parameters

pIndex [int](#)

index of the Particle to start at

particles [List](#)<[Particle](#)>

particles to rummage through

CheckExclude(in int, ref DynamicBuffer<Particle>)

[BurstCompile]

```
public static void CheckExclude(in int pIndex, ref DynamicBuffer<Particle> _particles)
```

Parameters

pIndex [int](#)

_particles DynamicBuffer<[Particle](#)>

CheckExclude(ref DynamicBuffer<Particle>)

```
[BurstCompile]
public void CheckExclude(ref DynamicBuffer<Particle> particles)
```

Parameters

particles DynamicBuffer<[Particle](#)>

Create(IBaker, Transform, Transform,
List<SkinnedMeshRenderer>, RigComponent, float3, float, int,
ref float, bool, bool, int, float, float, float, float, float, float, bool)

Creates a Particle using the given arguments

```
public static Particle Create(IBaker baker, Transform t, Transform tParent,
List<SkinnedMeshRenderer> m_Renderers, RigComponent rig, float3 endOffset, float endLength,
int array_index, ref float m_BoneTotalLength, bool excludeFromParticlePhysics = false, bool
excludeFromCollision = false, int parentIndex = -1, float damping = 0, float elasticity = 0,
float stiffness = 0, float inert = 0, float friction = 0, float radius = 0, bool isCollide
= false)
```

Parameters

baker IBaker

IBaker

t Transform

the Referenced Transform that the particle will represent

tParent Transform

parent of t Transform

m_Renderers [List](#)<SkinnedMeshRenderer>

SkinnedMeshRenderer

rig [RigComponent](#)

RigComponent

endOffset float3

endOffset

endLength float[↗]

endLength

array_index int[↗]

index Particle will be in DynamicBuffer array

m_BoneTotalLength float[↗]

Total Length of the Bone (this is usually handled automatically)

excludeFromParticlePhysics bool[↗]

exclude the particle from calculations

excludeFromCollision bool[↗]

set this to true to exclude this particle from collision

parentIndex int[↗]

parent index of the particle within the DynamicBuffer. (This is setup automatically and should not be modified)

damping float[↗]

How much the particle slowed down.

elasticity float[↗]

How much the force applied to return each particle to original orientation.

stiffness float[↗]

How much particle's original orientation are preserved.

inert float[↗]

How much the particle slowed down.

friction float[↗]

How much the particle slowed down when collide.

radius [float](#)

Each particle can be a sphere to collide with colliders. Radius describe sphere's size.

isCollide [bool](#)

true if the particle is colliding

Returns

[Particle](#)

Particle

Exclude(ref Particle, ref List<Particle>)

Excludes a Particle from Particle Simulation

```
public static void Exclude(ref Particle p, ref List<Particle> particles)
```

Parameters

p [Particle](#)

the reference Particle

particles [List](#)<[Particle](#)>

a List of particles to rummage through

Exclude(ref Particle, ref DynamicBuffer<Particle>)

```
[BurstCompile]
```

```
public static void Exclude(ref Particle p, ref DynamicBuffer<Particle> particles)
```

Parameters

p [Particle](#)

```
particles DynamicBuffer<Particle>
```

Exclude(ref List<Particle>)

```
public void Exclude(ref List<Particle> particles)
```

Parameters

```
particles List<Particle>
```

Exclude(ref DynamicBuffer<Particle>)

Excludes the Particle from calculations

```
[BurstCompile]
public void Exclude(ref DynamicBuffer<Particle> particles)
```

Parameters

```
particles DynamicBuffer<Particle>
```

Include(ref Particle, ref List<Particle>)

Includes a Particle from Particle Simulation

```
public static void Include(ref Particle p, ref List<Particle> particles)
```

Parameters

```
p Particle
```

the reference Particle

```
particles List<Particle>
```

a List of particles to rummage through

Include(ref Particle, ref DynamicBuffer<Particle>)

```
[BurstCompile]
public static void Include(ref Particle p, ref DynamicBuffer<Particle> particles)
```

Parameters

p [Particle](#)

particles [DynamicBuffer<Particle>](#)

Include(ref List<Particle>)

```
public void Include(ref List<Particle> particles)
```

Parameters

particles [List<Particle>](#)

Include(ref DynamicBuffer<Particle>)

Includes the Particles in calculations

```
[BurstCompile]
public void Include(ref DynamicBuffer<Particle> particles)
```

Parameters

particles [DynamicBuffer<Particle>](#)

IsRootBone(ref Particle, in Entity, out bool)

Determines if a Particle is a RootBone

```
public static void IsRootBone(ref Particle p, in Entity rootBone, out bool m_isRootBone)
```

Parameters

p [Particle](#)

reference particle to check

rootBone Entity

RootBone for comparison

m_isRootBone [bool](#)

resulting output

IsRootBone(in Entity, out bool)

```
public void IsRootBone(in Entity rootBone, out bool m_isRootBone)
```

Parameters

rootBone Entity

m_isRootBone [bool](#)

IsRootLikeBone(ref Particle, out bool)

Determines if a Particle is RootLike (meaning it isn't the RootBone but has similar characteristics)

```
public static void IsRootLikeBone(ref Particle p, out bool m_isRootLike)
```

Parameters

p [Particle](#)

reference particle to check

m_isRootLike [bool](#)

resulting output

IsRootLikeBone(out bool)

```
public void IsRootLikeBone(out bool m_isRootLike)
```

Parameters

m_isRootLike [bool ↗](#)

SwapParentIndices()

```
public void SwapParentIndices()
```

ToggleExclude(ref Particle)

```
[BurstCompile]  
public static void ToggleExclude(ref Particle p)
```

Parameters

p [Particle](#)

Verify()

Verifies if some important values are valid.

```
public void Verify()
```

Struct ParticleTransform

Namespace: [DOTSDynamicBone](#)

Assembly: DOTSDynamicBone.dll

This is an imitation of a Unity Transform since Rigidbody didn't give me enough information

```
[Serializable]
[BurstCompile]
public struct ParticleTransform : IComponentData, IQueryTypeParameter
```

Implements

IComponentData, IQueryTypeParameter

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Remarks

This is an imitation of a Unity Transform since Rigidbody didn't give me enough information

Constructors

ParticleTransform(float3, float3, quaternion, quaternion, float3, float3, int)

creates a new ParticleTransform using the given arguments

```
public ParticleTransform(float3 position, float3 localPosition, quaternion rotation,
quaternion localRotation, float3 localScale, float3 lossyScale, int childCount)
```

Parameters

position float3

[psotopm]

localPosition float3

`;pca; position`

`rotation` quaternion

rotation

`localRotation` quaternion

local rotation

`localScale` float3

local scale

`lossyScale` float3

lossy scale

`childCount` [int](#)

amount of children the transform has

ParticleTransform(LocalToWorld, LocalTransform, int)

`public ParticleTransform(LocalToWorld ltw, LocalTransform lt, int childCount = 0)`

Parameters

`ltw` LocalToWorld

`lt` LocalTransform

`childCount` [int](#)

ParticleTransform(Transform)

creates a ParticleTransform from the provided Transform

`public ParticleTransform(Transform transform)`

Parameters

transform Transform

UnityEngine.Transform

Fields

childCount

`public int childCount`

Field Value

[int](#)

amount of children the transform has

localToRoot

`public float4x4 localToRoot`

Field Value

float4x4

localToWorld

`public float4x4 localToWorld`

Field Value

float4x4

Properties

Null

```
public static ParticleTransform Null { get; }
```

Property Value

[ParticleTransform](#)

A Null ParticleTransform

localPosition

```
public float3 localPosition { get; set; }
```

Property Value

float3

local position

localRotation

```
public quaternion localRotation { get; set; }
```

Property Value

quaternion

local rotation

localScale

```
public float3 localScale { get; set; }
```

Property Value

float3

lossy scale

lossyScale

```
public float3 lossyScale { get; set; }
```

Property Value

float3

local scale

position

```
public float3 position { get; set; }
```

Property Value

float3

position

rotation

```
public quaternion rotation { get; set; }
```

Property Value

quaternion

rotation

Methods

Equals(ref ParticleTransform)

Compares Particle Transform with another one

```
public bool Equals(ref ParticleTransform otherOne)
```

Parameters

otherOne [ParticleTransform](#)

Returns

[bool](#)

SetTransform(Transform)

Sets the value of the ParticleTransform using the given transform

```
public void SetTransform(Transform transform)
```

Parameters

transform [Transform](#)

[UnityEngine.Transform](#)

ToLocalToWorld()

converts the ParticleTransform into a LocalToWorld using the internal data values

```
public LocalToWorld ToLocalToWorld()
```

Returns

[LocalToWorld](#)

[LocalToWorld](#)

ToLocalToWorld(bool)

converts the ParticleTransform into a LocalToWorld using the internal data values

```
public LocalToWorld ToLocalToWorld(bool useOnlyLocalScale)
```

Parameters

useOnlyLocalScale [bool](#)

Returns

LocalToWorld

LocalToWorld

ToLocalTransform()

Converts the ParticleTransform into a LocalTransform using internal data values

```
public LocalTransform ToLocalTransform()
```

Returns

LocalTransform

LocalTransform

ToTransform(ParticleTransform)

This creates a UnityEngine.Transform from the ParticleTransform's data.

```
public static Transform ToTransform(ParticleTransform particleTransform)
```

Parameters

particleTransform [ParticleTransform](#)

ParticleTransform

Returns

Transform

Enum ParticleTransform.DOTSDynamicBone ColliderType

Namespace: [DOTSDynamicBone](#)

Assembly: DOTSDynamicBone.dll

```
public enum ParticleTransform.DOTSDynamicBoneColliderType
```

Fields

Cube = 2

Plane = 1

SphereLike = 0

Namespace DOTSDynamicBone.Collision

Classes

[DOTSDynamicBoneCollider](#)

This is a Preset collider for the DOTSDynamicBoneCollisionSystem

Structs

[DOTSDynamicBoneCollider_BufferElement](#)

This is a collider for the DOTSDynamicBoneSystem

Interfaces

[IDOTSDynamicBoneCollider](#)

Class DOTSDynamicBoneCollider

Namespace: [DOTSDynamicBone.Collision](#)

Assembly: DOTSDynamicBone.dll

This is a Preset collider for the DOTSDynamicBoneCollisionSystem

```
public class DOTSDynamicBoneCollider : MonoBehaviour
```

Inheritance

[object](#) ← Object ← Component ← Behaviour ← MonoBehaviour ← DOTSDynamicBoneCollider

Inherited Members

MonoBehaviour.IsInvoking() , MonoBehaviour.CancelInvoke() , [MonoBehaviour.Invoke\(string, float\)](#) ,
[MonoBehaviour.InvokeRepeating\(string, float, float\)](#) , [MonoBehaviour.CancelInvoke\(string\)](#) ,
[MonoBehaviour.IsInvoking\(string\)](#) , [MonoBehaviour.StartCoroutine\(string\)](#) ,
[MonoBehaviour.StartCoroutine\(string, object\)](#) , [MonoBehaviour.StartCoroutine\(IEnumerator\)](#) ,
[MonoBehaviour.StartCoroutine_Auto\(IEnumerator\)](#) , [MonoBehaviour.StopCoroutine\(IEnumerator\)](#) ,
MonoBehaviour.StopCoroutine(Coroutine) , [MonoBehaviour.StopCoroutine\(string\)](#) ,
MonoBehaviour.StopAllCoroutines() , [MonoBehaviour.print\(object\)](#) ,
MonoBehaviour.destroyCancellationToken , MonoBehaviour.useGUILayout ,
MonoBehaviour.runInEditMode , Behaviour.enabled , Behaviour.isActiveAndEnabled ,
[Component.GetComponent\(Type\)](#) , Component.GetComponent<T>() ,
[Component.TryGetComponent\(Type, out Component\)](#) , Component.TryGetComponent<T>(out T) ,
[Component.GetComponent\(string\)](#) , [Component.GetComponentInChildren\(Type, bool\)](#) ,
[Component.GetComponentInChildren\(Type\)](#) , [Component.GetComponentInChildren<T>\(bool\)](#) ,
Component.GetComponentInChildren<T>() , [Component.GetComponentsInChildren\(Type, bool\)](#) ,
[Component.GetComponentsInChildren\(Type\)](#) , [Component.GetComponentsInChildren<T>\(bool\)](#) ,
[Component.GetComponentsInChildren<T>\(bool, List<T>\)](#) ,
Component.GetComponentsInChildren<T>() , [Component.GetComponentsInChildren<T>\(List<T>\)](#) ,
[Component.GetComponentInParent\(Type, bool\)](#) , [Component.GetComponentInParent\(Type\)](#) ,
[Component.GetComponentInParent<T>\(bool\)](#) , Component.GetComponentInParent<T>() ,
[Component.GetComponentsInParent\(Type, bool\)](#) , [Component.GetComponentsInParent\(Type\)](#) ,
[Component.GetComponentsInParent<T>\(bool\)](#) ,
[Component.GetComponentsInParent<T>\(bool, List<T>\)](#) , Component.GetComponentsInParent<T>() ,
[Component.GetComponents\(Type\)](#) , [Component.GetComponents\(Type, List<Component>\)](#) ,
[Component.GetComponents<T>\(List<T>\)](#) , Component.GetComponents<T>() ,
Component.GetComponentIndex() , [Component.CompareTag\(string\)](#) ,
[Component.SendMessageUpwards\(string, object, SendMessageOptions\)](#) ,

[Component.SendMessageUpwards\(string, object\)](#) , [Component.SendMessageUpwards\(string\)](#) ,
[Component.SendMessageUpwards\(string, SendMessageOptions\)](#) ,
[Component.SendMessage\(string, object\)](#) , [Component.SendMessage\(string\)](#) ,
[Component.SendMessage\(string, object, SendMessageOptions\)](#) ,
[Component.SendMessage\(string, SendMessageOptions\)](#) ,
[Component.BroadcastMessage\(string, object, SendMessageOptions\)](#) ,
[Component.BroadcastMessage\(string, object\)](#) , [Component.BroadcastMessage\(string\)](#) ,
[Component.BroadcastMessage\(string, SendMessageOptions\)](#) , Component.transform ,
Component.gameObject , Component.tag , Object.GetInstanceID() , Object.GetHashCode() ,
[Object.Equals\(object\)](#) , Object.InstantiateAsync<T>(T) , Object.InstantiateAsync<T>(T, Transform) ,
Object.InstantiateAsync<T>(T, Vector3, Quaternion) ,
Object.InstantiateAsync<T>(T, Transform, Vector3, Quaternion) , [Object.InstantiateAsync<T>\(T, int\)](#) ,
[Object.InstantiateAsync<T>\(T, int, Transform\)](#) ,
[Object.InstantiateAsync<T>\(T, int, Vector3, Quaternion\)](#) ,
[Object.InstantiateAsync<T>\(T, int, ReadOnlySpan<Vector3>, ReadOnlySpan<Quaternion>\)](#) ,
[Object.InstantiateAsync<T>\(T, int, Transform, Vector3, Quaternion\)](#) ,
[Object.InstantiateAsync<T>\(T, int, Transform, ReadOnlySpan<Vector3>, ReadOnlySpan<Quaternion>\)](#) ,
Object.Instantiate(Object, Vector3, Quaternion) ,
Object.Instantiate(Object, Vector3, Quaternion, Transform) , Object.Instantiate(Object) ,
Object.Instantiate(Object, Scene) , Object.Instantiate(Object, Transform) ,
[Object.Instantiate\(Object, Transform, bool\)](#) , Object.Instantiate<T>(T) ,
Object.Instantiate<T>(T, Vector3, Quaternion) ,
Object.Instantiate<T>(T, Vector3, Quaternion, Transform) , Object.Instantiate<T>(T, Transform) ,
[Object.Instantiate<T>\(T, Transform, bool\)](#) , [Object.Destroy\(Object, float\)](#) , Object.Destroy(Object) ,
[Object.DestroyImmediate\(Object, bool\)](#) , Object.DestroyImmediate(Object) ,
[Object.FindObjectsOfType\(Type\)](#) , [Object.FindObjectsOfType\(Type, bool\)](#) ,
[Object.FindObjectsByType\(Type, FindObjectsSortMode\)](#) ,
[Object.FindObjectsByType\(Type, FindObjectsInactive, FindObjectsSortMode\)](#) ,
Object.DontDestroyOnLoad(Object) , [Object.DestroyObject\(Object, float\)](#) ,
Object.DestroyObject(Object) , [Object.FindSceneObjectsOfType\(Type\)](#) ,
[Object.FindObjectsOfTypeIncludingAssets\(Type\)](#) , Object.FindObjectsOfType<T>() ,
Object.FindObjectsByType<T>(FindObjectsSortMode) , [Object.FindObjectsOfType<T>\(bool\)](#) ,
Object.FindObjectsByType<T>(FindObjectsInactive, FindObjectsSortMode) ,
ObjectFindObjectOfType<T>() , [Object.FindObjectType<T>\(bool\)](#) ,
Object.FindFirstObjectByType<T>() , Object.FindAnyObjectByType<T>() ,
Object.FindFirstObjectByType<T>(FindObjectsInactive) ,
Object.FindAnyObjectByType<T>(FindObjectsInactive) , [Object.FindObjectsOfTypeAll\(Type\)](#) ,
[Object.FindObjectType\(Type\)](#) , [Object.FindFirstObjectByType\(Type\)](#) ,
[Object.FindAnyObjectByType\(Type\)](#) , [Object.FindObjectType\(Type, bool\)](#) ,
[Object.FindFirstObjectByType\(Type, FindObjectsInactive\)](#) ,

[Object.FindAnyObjectByType\(Type, FindObjectsInactive\)](#) , Object.ToString() , Object.name ,
Object.hideFlags , [object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,
[object.ReferenceEquals\(object, object\)](#)

Remarks

This is a Preset collider for the DOTSDynamicBoneCollisionSystem

Fields

Enable

```
[Tooltip("enables or disable the collider from collision calculations")]
public bool Enable
```

Field Value

[bool](#)

This represents the type of collider the DOTS Dynamic Collider represents

FallbackToDefaultColliderOnNullPhysicsCollider

```
[Tooltip("set to true to use the Particle's m_Radius to perform collisions when the
Particle's PhysicsCollider is null and UseParticlePhysicsCollider is true")]
public bool FallbackToDefaultColliderOnNullPhysicsCollider
```

Field Value

[bool](#)

Persistant

```
[Tooltip("set this to true to prevent the removal of this collider from the DynamicBuffer at
the end of the frame")]
public bool Persistant
```

Field Value

[bool](#) ↗

UseParticlePhysicsCollider

```
[Tooltip("set to true to use the Unity.Physics.PhysicsCollider data stored within  
a Particle")]  
public bool UseParticlePhysicsCollider
```

Field Value

[bool](#) ↗

UseUnityPhysicsCollider

```
[Tooltip("set to true to use the Unity.Physics.PhysicsCollider data generated by this  
GameObject's PhysicsShape & PhysicsBody")]  
public bool UseUnityPhysicsCollider
```

Field Value

[bool](#) ↗

m_Bound

```
[Tooltip("type fo collision bound. NOTE: Bound types isn't currently supported to  
with Unity.Physics.Collider")]  
public TransformsExtensions.Bound m_Bound
```

Field Value

[TransformsExtensions.Bound](#)

m_Center

```
[Tooltip("center of the collider")]
public float3 m_Center
```

Field Value

float3

m_Direction

```
[Tooltip("The up vector axis of the collider")]
public TransformsExtensions.Direction m_Direction
```

Field Value

[TransformsExtensions.Direction](#)

m_Height

```
[Tooltip(" height of collider ")]
public float m_Height
```

Field Value

[float](#) ↗

m_Radius

```
[Tooltip("radius of collider")]
public float m_Radius
```

Field Value

[float](#) ↗

Methods

ToDOTSDynamicBoneCollider_BufferElement(IBaker, Transform, bool)

Converts the collider into a DOTSDynamicBoneCollider_BufferElement

```
public DOTSDynamicBoneCollider_BufferElement ToDOTSDynamicBoneCollider_BufferElement(IBaker  
baker, Transform BoneRoot, bool IsCollided = false)
```

Parameters

baker IBaker

IBaker

BoneRoot Transform

Bone

IsCollided bool

Sets the IsCollided value in the DOTSDynamicBoneCollider_BUfferElement

Returns

[DOTSDynamicBoneCollider_BufferElement](#)

DOTSDynamicBoneCollider_BufferElement

Struct DOTSDynamicBoneCollider_BufferElement

Namespace: [DOTSDynamicBone.Collision](#)

Assembly: DOTSDynamicBone.dll

This is a collider for the DOTSDynamicBoneSystem

```
public struct DOTSDynamicBoneCollider_BufferElement : IDOTSDynamicBoneCollider,  
IBufferElementData
```

Implements

[IDOTSDynamicBoneCollider](#), IBufferElementData

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Remarks

This is a collider for the DOTSDynamicBoneSystem

Fields

IsCollided

```
public bool IsCollided
```

Field Value

[bool](#)

set this to true to have this collider always return true

boneEntity

```
public Entity boneEntity
```

Field Value

Entity

the bone entity associated with this collider

enabled

```
public bool enabled
```

Field Value

[bool](#)

whether or not the collider is enabled.

entity

```
public Entity entity
```

Field Value

Entity

the collider entity

fallbackToDefaultColliderOnNullPhysicsCollider

```
public bool fallbackToDefaultColliderOnNullPhysicsCollider
```

Field Value

[bool](#)

set to true to use the Particle's m_Radius to perform collisions when the Particle's PhysicsCollider is null and UseParticlePhysicsCollider is true

m_Height

```
public float m_Height
```

Field Value

[float](#)

The height of the capsule.

m_Radius

```
public float m_Radius
```

Field Value

[float](#)

The radius of the sphere or capsule.

persistent

```
public bool persistent
```

Field Value

[bool](#)

set this to true to prevent the removal of this collider from the DynamicBuffer at the end of the frame

transform

```
public ParticleTransform transform
```

Field Value

[ParticleTransform](#)

the entity's Transform

useUnityPhysicsCollider

```
public bool useUnityPhysicsCollider
```

Field Value

[bool](#) ↗

Set this to true to use the Unity.Physics.Collider in stead of the internally stored values

Properties

Null

```
[SerializeField]  
public static DOTSDynamicBoneCollider_BufferElement Null { get; }
```

Property Value

[DOTSDynamicBoneCollider_BufferElement](#)

m_Bound

```
[SerializeField]  
public TransformsExtensions.Bound m_Bound { get; set; }
```

Property Value

[TransformsExtensions.Bound](#)

Constrain bones to outside bound or inside bound

m_Center

```
[SerializeField]  
public float3 m_Center { get; set; }
```

Property Value

float3

The center of the sphere or capsule, in the object's local space.

m_Direction

```
[SerializeField]  
public TransformsExtensions.Direction m_Direction { get; set; }
```

Property Value

[TransformsExtensions.Direction](#)

The axis of the capsule's height.

Methods

Collide(ref float3, float)

Tests for a collision

```
public bool Collide(ref float3 particlePosition, float particleRadius)
```

Parameters

particlePosition float3

position

particleRadius [float](#)

radius

Returns

[bool](#)

Create(IBaker, DOTSDynamicBoneCollider, Transform, bool)

```
public static DOTSDynamicBoneCollider_BufferElement Create(IBaker baker,  
DOTSDynamicBoneCollider ddbc, Transform BoneRoot, bool IsCollided = false)
```

Parameters

baker IBaker

ddbc [DOTSDynamicBoneCollider](#)

BoneRoot Transform

IsCollided [bool](#)

Returns

[DOTSDynamicBoneCollider_BufferElement](#)

Interface IDOTSDynamicBoneCollider

Namespace: [DOTSDynamicBone.Collision](#)

Assembly: DOTSDynamicBone.dll

```
public interface IDOTSDynamicBoneCollider : IBufferElementData
```

Properties

m_Bound

```
TransformsExtensions.Bound m_Bound { get; set; }
```

Property Value

[TransformsExtensions.Bound](#)

Constrain bones to outside bound or inside bound

m_Center

```
float3 m_Center { get; set; }
```

Property Value

float3

The center of the sphere or capsule, in the object's local space.

m_Direction

```
TransformsExtensions.Direction m_Direction { get; set; }
```

Property Value

[TransformsExtensions.Direction](#)

The axis of the capsule's height.

Methods

Collide(ref float3, float)

Collide function

```
bool Collide(ref float3 particlePosition, float particleRadius)
```

Parameters

particlePosition float3

the particle's position

particleRadius [float](#)

the particle's radius

Returns

[bool](#)

Namespace DOTSDynamicBone.Rig

Classes

[RigComponent](#)

This is a copy of the Unity Entities Animation RigComponent. I only kept what I needed and modified it to simplify things.

Structs

[AnimatedLocalToRoot](#)

[AnimatedLocalToWorld](#)

[ExcludedBone](#)

Struct AnimatedLocalToRoot

Namespace: [DOTSDynamicBone.Rig](#)

Assembly: DOTSDynamicBone.dll

```
public struct AnimatedLocalToRoot : IBufferElementData
```

Implements

IBufferElementData

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Fields

Value

```
public float4x4 Value
```

Field Value

float4x4

Struct AnimatedLocalToWorld

Namespace: [DOTSDynamicBone.Rig](#)

Assembly: DOTSDynamicBone.dll

```
public struct AnimatedLocalToWorld : IBufferElementData
```

Implements

IBufferElementData

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Fields

Value

```
public float4x4 Value
```

Field Value

float4x4

Struct ExcludedBone

Namespace: [DOTSDynamicBone.Rig](#)

Assembly: DOTSDynamicBone.dll

```
[Serializable]
public struct ExcludedBone
```

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Fields

bone

```
public Transform bone
```

Field Value

Transform

excludeChildren

```
public bool excludeChildren
```

Field Value

[bool](#)

Class RigComponent

Namespace: [DOTSDynamicBone.Rig](#)

Assembly: DOTSDynamicBone.dll

This is a copy of the Unity Entities Animation RigComponent. I only kept what I needed and modified it to simplify things.

```
public class RigComponent : MonoBehaviour
```

Inheritance

[object](#) ← Object ← Component ← Behaviour ← MonoBehaviour ← RigComponent

Inherited Members

MonoBehaviour.IsInvoking() , MonoBehaviour.CancelInvoke() , [MonoBehaviour.Invoke\(string, float\)](#) ,
[MonoBehaviour.InvokeRepeating\(string, float, float\)](#) , [MonoBehaviour.CancelInvoke\(string\)](#) ,
[MonoBehaviour.IsInvoking\(string\)](#) , [MonoBehaviour.StartCoroutine\(string\)](#) ,
[MonoBehaviour.StartCoroutine\(string, object\)](#) , [MonoBehaviour.StartCoroutine\(IEnumerator\)](#) ,
[MonoBehaviour.StartCoroutine_Auto\(IEnumerator\)](#) , [MonoBehaviour.StopCoroutine\(IEnumerator\)](#) ,
MonoBehaviour.StopCoroutine(Coroutine) , [MonoBehaviour.StopCoroutine\(string\)](#) ,
MonoBehaviour.StopAllCoroutines() , [MonoBehaviour.print\(object\)](#) ,
MonoBehaviour.destroyCancellationToken , MonoBehaviour.useGUILayout ,
MonoBehaviour.runInEditMode , Behaviour.enabled , Behaviour.isActiveAndEnabled ,
[Component.GetComponent\(Type\)](#) , Component.GetComponent<T>() ,
[Component.TryGetComponent\(Type, out Component\)](#) , Component.TryGetComponent<T>(out T) ,
[Component.GetComponent\(string\)](#) , [Component.GetComponentInChildren\(Type, bool\)](#) ,
[Component.GetComponentInChildren\(Type\)](#) , [Component.GetComponentInChildren<T>\(bool\)](#) ,
Component.GetComponentInChildren<T>() , [Component.GetComponentsInChildren\(Type, bool\)](#) ,
[Component.GetComponentsInChildren\(Type\)](#) , [Component.GetComponentsInChildren<T>\(bool\)](#) ,
[Component.GetComponentsInChildren<T>\(bool, List<T>\)](#) ,
Component.GetComponentsInChildren<T>() , [Component.GetComponentsInChildren<T>\(List<T>\)](#) ,
[Component.GetComponentInParent\(Type, bool\)](#) , [Component.GetComponentInParent\(Type\)](#) ,
[Component.GetComponentInParent<T>\(bool\)](#) , Component.GetComponentInParent<T>() ,
[Component.GetComponentsInParent\(Type, bool\)](#) , [Component.GetComponentsInParent\(Type\)](#) ,
[Component.GetComponentsInParent<T>\(bool\)](#) ,
[Component.GetComponentsInParent<T>\(bool, List<T>\)](#) , Component.GetComponentsInParent<T>() ,
[Component.GetComponents\(Type\)](#) , [Component.GetComponents\(Type, List<Component>\)](#) ,
[Component.GetComponents<T>\(List<T>\)](#) , Component.GetComponents<T>() ,
Component.GetComponentIndex() , [Component.CompareTag\(string\)](#) ,

[Component.SendMessageUpwards\(string, object, SendMessageOptions\)](#) ,
[Component.SendMessageUpwards\(string, object\)](#) , [Component.SendMessageUpwards\(string\)](#) ,
[Component.SendMessageUpwards\(string, SendMessageOptions\)](#) ,
[Component.SendMessage\(string, object\)](#) , [Component.SendMessage\(string\)](#) ,
[Component.SendMessage\(string, object, SendMessageOptions\)](#) ,
[Component.SendMessage\(string, SendMessageOptions\)](#) ,
[Component.BroadcastMessage\(string, object, SendMessageOptions\)](#) ,
[Component.BroadcastMessage\(string, object\)](#) , [Component.BroadcastMessage\(string\)](#) ,
[Component.BroadcastMessage\(string, SendMessageOptions\)](#) , Component.transform ,
Component.gameObject , Component.tag , Object.GetInstanceID() , Object.GetHashCode() ,
[Object.Equals\(object\)](#) , Object.InstantiateAsync<T>(T) , Object.InstantiateAsync<T>(T, Transform) ,
Object.InstantiateAsync<T>(T, Vector3, Quaternion) ,
Object.InstantiateAsync<T>(T, Transform, Vector3, Quaternion) , [Object.InstantiateAsync<T>\(T, int\)](#) ,
[Object.InstantiateAsync<T>\(T, int, Transform\)](#) ,
[Object.InstantiateAsync<T>\(T, int, Vector3, Quaternion\)](#) ,
[Object.InstantiateAsync<T>\(T, int, ReadOnlySpan<Vector3>, ReadOnlySpan<Quaternion>\)](#) ,
[Object.InstantiateAsync<T>\(T, int, Transform, Vector3, Quaternion\)](#) ,
[Object.InstantiateAsync<T>\(T, int, Transform, ReadOnlySpan<Vector3>, ReadOnlySpan<Quaternion>\)](#) ,
Object.Instantiate(Object, Vector3, Quaternion) ,
Object.Instantiate(Object, Vector3, Quaternion, Transform) , Object.Instantiate(Object) ,
Object.Instantiate(Object, Scene) , Object.Instantiate(Object, Transform) ,
[Object.Instantiate\(Object, Transform, bool\)](#) , Object.Instantiate<T>(T) ,
Object.Instantiate<T>(T, Vector3, Quaternion) ,
Object.Instantiate<T>(T, Vector3, Quaternion, Transform) , Object.Instantiate<T>(T, Transform) ,
[Object.Instantiate<T>\(T, Transform, bool\)](#) , [Object.Destroy\(Object, float\)](#) , Object.Destroy(Object) ,
[Object.DestroyImmediate\(Object, bool\)](#) , Object.DestroyImmediate(Object) ,
[Object.FindObjectsOfType\(Type\)](#) , [Object.FindObjectsOfType\(Type, bool\)](#) ,
[Object.FindObjectsByType\(Type, FindObjectsSortMode\)](#) ,
[Object.FindObjectsByType\(Type, FindObjectsInactive, FindObjectsSortMode\)](#) ,
Object.DontDestroyOnLoad(Object) , [Object.DestroyObject\(Object, float\)](#) ,
Object.DestroyObject(Object) , [Object.FindSceneObjectsOfType\(Type\)](#) ,
[Object.FindObjectsOfTypeIncludingAssets\(Type\)](#) , Object.FindObjectsOfType<T>() ,
Object.FindObjectsByType<T>(FindObjectsSortMode) , [Object.FindObjectsOfType<T>\(bool\)](#) ,
Object.FindObjectsByType<T>(FindObjectsInactive, FindObjectsSortMode) ,
ObjectFindObjectOfType<T>() , [Object.FindObjectType<T>\(bool\)](#) ,
Object.FindFirstObjectByType<T>() , Object.FindAnyObjectByType<T>() ,
Object.FindFirstObjectByType<T>(FindObjectsInactive) ,
Object.FindAnyObjectByType<T>(FindObjectsInactive) , [Object.FindObjectsOfTypeAll\(Type\)](#) ,
[Object.FindObjectType\(Type\)](#) , [Object.FindFirstObjectByType\(Type\)](#) ,
[Object.FindAnyObjectByType\(Type\)](#) , [Object.FindObjectType\(Type, bool\)](#) ,

[Object.FindFirstObjectByType\(Type, FindObjectsInactive\)](#) ,
[Object.FindAnyObjectByType\(Type, FindObjectsInactive\)](#) , Object.ToString() , Object.name ,
Object.hideFlags , [object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,
[object.ReferenceEquals\(object, object\)](#)

Fields

Bones

`public Transform[] Bones`

Field Value

Transform[]

Properties

SkeletonRootBone

`public Transform SkeletonRootBone { get; set; }`

Property Value

Transform

Methods

ExcludedBonesContains(List<ExcludedBone>, Transform)

`public bool ExcludedBonesContains(List<ExcludedBone> excludedBones, Transform bone)`

Parameters

`excludedBones List<ExcludedBone>`

bone Transform

Returns

[bool](#)

GetExcludedBones()

```
public List<ExcludedBone> GetExcludedBones()
```

Returns

[List](#)<[ExcludedBone](#)>

GetExcludedBonesIndex(List<ExcludedBone>, Transform)

```
public int GetExcludedBonesIndex(List<ExcludedBone> excludedBones, Transform bone)
```

Parameters

excludedBones [List](#)<[ExcludedBone](#)>

bone Transform

Returns

[int](#)

Namespace DOTSDynamicBone.Systems

Classes

[DOTSDynamicBoneUpdateGroup](#)

A system group containing systems that process DOTSDynamicBone updates

Structs

[DOTSDynamicBoneAnimatedLocalToXPreUpdateSystem](#)

[DOTSDynamicBoneAnimatedLocalToXPreUpdateSystem.SetAnimatedLocalToXJob](#)

[DOTSDynamicBoneAnimatedLocalToXPreUpdateSystem.SetAnimatedLocalToXJob.InternalCompiler](#)

Internal structure used by the compiler

[DOTSDynamicBoneAnimatedLocalToXPreUpdateSystem.SetAnimatedLocalToXJob.InternalCompilerQueryAndHandleData](#)

Used internally by the compiler, we won't promise this exists in the future

[DOTSDynamicBoneAnimatedLocalToXPreUpdateSystem.SetAnimatedLocalToXJob.InternalCompilerQueryAndHandleData.TypeHandle](#)

[DOTSDynamicBoneCullingSystem](#)

This system updates the tracked bone.m_ReferenceObject transform

[DOTSDynamicBoneParticleResetOptions](#)

[DOTSDynamicBoneParticleResetSystem](#)

[DOTSDynamicBoneParticleResetSystem.DOTSDynamicBoneParticleResetJobOptimized](#)

This Job is responsible for Resetting parameters related to DOTSDynamicBone. Since particles use the resetOptions in DDB, this must execute after Particles execute

[DOTSDynamicBoneParticleResetSystem.DOTSDynamicBoneParticleResetJobOptimized.InternalCompiler](#)

Internal structure used by the compiler

[DOTSDynamicBoneParticleResetSystem.DOTSDynamicBoneParticleResetJobOptimized.InternalCompilerQueryAndHandleData](#)

Used internally by the compiler, we won't promise this exists in the future

[DOTSDynamicBoneParticleResetSystem.DOTSDynamicBoneParticleResetJobOptimized.InternalCompilerQueryAndHandleData.TypeHandle](#)

[DOTSDynamicBoneUpdateSystem](#)

[DOTSDynamicBoneUpdateSystem.DOTSDynamicBoneUpdateJob](#)

[DOTSDynamicBoneUpdateSystem.DOTSDynamicBoneUpdateJob.InternalCompiler](#)

Internal structure used by the compiler

[DOTSDynamicBoneUpdateSystem.DOTSDynamicBoneUpdateJob.InternalCompilerQueryAndHandleData](#)

Used internally by the compiler, we won't promise this exists in the future

[DOTSDynamicBoneUpdateSystem.DOTSDynamicBoneUpdateJob.InternalCompilerQueryAndHandleData.TypeHandle](#)

[DOTSDynamicBoneVisualPhysicalOverrideSystem](#)

[DOTSDynamicBoneVisualPhysicalOverrideSystem.UpdateXTransformJob](#)

[DOTSDynamicBoneVisualPhysicalOverrideSystem.UpdateXTransformJob.InternalCompiler](#)

Internal structure used by the compiler

[DOTSDynamicBoneVisualPhysicalOverrideSystem.UpdateXTransformJob.InternalCompilerQueryAndHandleData](#)

Used internally by the compiler, we won't promise this exists in the future

[DOTSDynamicBoneVisualPhysicalOverrideSystem.UpdateXTransformJob.InternalCompilerQueryAndHandleData.TypeHandle](#)

Struct DOTSDynamicBoneAnimatedLocalToXPreUpdateSystem

Namespace: [DOTSDynamicBone.Systems](#)

Assembly: DOTSDynamicBone.dll

```
[UpdateInGroup(typeof(DOTSDynamicBoneUpdateGroup))]
[UpdateBefore(typeof(DOTSDynamicBoneUpdateSystem))]
[WorldSystemFilter(WorldSystemFilterFlags.Default|WorldSystemFilterFlags.Editor,
WorldSystemFilterFlags.Default)]
[BurstCompile]
public struct DOTSDynamicBoneAnimatedLocalToXPreUpdateSystem :
ISystem, ISystemCompilerGenerated
```

Implements

ISystem, ISystemCompilerGenerated

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Methods

OnCreate(ref SystemState)

Called when this system is created.

```
[BurstCompile]
public void OnCreate(ref SystemState state)
```

Parameters

state SystemState

The Unity.Entities.SystemState backing this system instance

Remarks

Implement an **OnCreate** function to set up system resources when it is created.

OnCreate is invoked before the first time `Unity.Entities.ISystemStartStop.OnStartRunning(ref Unity.Entities.SystemState)` and `Unity.Entities.ISystem.OnUpdate(ref Unity.Entities.SystemState)` are invoked.

OnCreateForCompiler(ref SystemState)

Generated by compilation pipeline and used internally.

```
public void OnCreateForCompiler(ref SystemState state)
```

Parameters

state SystemState

The `Unity.Entities.SystemState` backing this system instance

OnUpdate(ref SystemState)

Implement **OnUpdate** to perform the major work of this system.

```
[BurstCompile]  
public void OnUpdate(ref SystemState state)
```

Parameters

state SystemState

The `Unity.Entities.SystemState` backing this system instance

Remarks

By default, the system invokes `OnUpdate` once every frame on the main thread. To skip `OnUpdate` if all of the system's `[EntityQueries]` are empty, use the `[RequireMatchingQueriesForUpdateAttribute]`. To limit when `OnUpdate` is invoked, you can specify components that must exist, or queries that match specific Entities. To do this, call `Unity.Entities.SystemState.RequireForUpdate<T>()` or `Unity.Entities.SystemState.RequireForUpdate(Unity.Entities.EntityQuery)` in the system's `OnCreate` method. For more information, see `Unity.Entities.SystemState.ShouldRunSystem()`.

You can instantiate and schedule an Unity.Entities.IJobChunk instance; you can use the [C# Job System] or you can perform work on the main thread. If you call Unity.Entities.EntityManager methods that perform structural changes on the main thread, be sure to arrange the system order to minimize the performance impact of the resulting [sync points].

UpdateAnimatedLocalToXValues(ref DynamicBuffer<Particle>, ref DynamicBuffer<AnimatedLocalToRoot>, ref DynamicBuffer<AnimatedLocalToWorld>)

```
[BurstCompile]
public static void UpdateAnimatedLocalToXValues(ref DynamicBuffer<Particle> particles, ref
DynamicBuffer<AnimatedLocalToRoot> altr, ref DynamicBuffer<AnimatedLocalToWorld> altw)
```

Parameters

particles DynamicBuffer<[Particle](#)>

altr DynamicBuffer<[AnimatedLocalToRoot](#)>

altw DynamicBuffer<[AnimatedLocalToWorld](#)>

UpdateAnimatedLocalToXValues_Old(ref DynamicBuffer<Particle>, ref DynamicBuffer<AnimatedLocalToRoot>, ref DynamicBuffer<AnimatedLocalToWorld>, in bool, in bool)

```
[BurstCompile]
public static void UpdateAnimatedLocalToXValues_Old(ref DynamicBuffer<Particle> particles,
ref DynamicBuffer<AnimatedLocalToRoot> altr, ref DynamicBuffer<AnimatedLocalToWorld> altw,
in bool UpdateTransformAsIfNoParent, in bool fliplocaltoworld)
```

Parameters

particles DynamicBuffer<[Particle](#)>

altr DynamicBuffer<[AnimatedLocalToRoot](#)>

altw DynamicBuffer<[AnimatedLocalToWorld](#)>

UpdateTransformAsIfNoParent [bool](#)

fliplocaltoworld [bool](#)

Struct DOTSDynamicBoneAnimatedLocalToXPreUpdateSystem.SetAnimatedLocalToXJob

Namespace: [DOTSDynamicBone.Systems](#)

Assembly: DOTSDynamicBone.dll

```
[BurstCompile]
public struct DOTSDynamicBoneAnimatedLocalToXPreUpdateSystem.SetAnimatedLocalToXJob : IJobEntity, IJobChunk
```

Implements

IJobEntity, IJobChunk

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Fields

GetLTR

```
[ReadOnly]
public ComponentLookup<LocalTransform> GetLTR
```

Field Value

ComponentLookup<LocalTransform>

GetLTWR

```
[ReadOnly]
public ComponentLookup<LocalToWorld> GetLTWR
```

Field Value

Methods

Execute(ref DOTSDynamicBone, ref DynamicBuffer<Particle>, ref DynamicBuffer<AnimatedLocalToWorld>, ref DynamicBuffer<AnimatedLocalToRoot>)

```
public void Execute(ref DOTSDynamicBone ddb, ref DynamicBuffer<Particle> particles, ref DynamicBuffer<AnimatedLocalToWorld> m_altw, ref DynamicBuffer<AnimatedLocalToRoot> m_altr)
```

Parameters

ddb [DOTSDynamicBone](#)

particles DynamicBuffer<[Particle](#)>

m_altw DynamicBuffer<[AnimatedLocalToWorld](#)>

m_altr DynamicBuffer<[AnimatedLocalToRoot](#)>

Execute(in ArchetypeChunk, int, bool, in v128)

Implement the `Execute` function to perform a unit of work on an `Unity.Entities.ArchetypeChunk` representing a chunk.

```
public void Execute(in ArchetypeChunk chunk, int chunkIndexInQuery, bool useEnabledMask, in v128 chunkEnabledMask)
```

Parameters

chunk ArchetypeChunk

An object providing access to the entities within a chunk.

chunkIndexInQuery [int](#)

useEnabledMask [bool](#)

If true, the contents of `chunkEnabledMask` describe which entities in the chunk match the provided `Unity.Entities.EntityQuery` and should be processed by this job. If false, all entities in the chunk match the provided query, and the contents of `chunkEnabledMask` are undefined.

`chunkEnabledMask` v128

If bit N in this mask is set, entity N in `chunk` matches the `Unity.Entities.EntityQuery` used to schedule the job. If bit N is clear, entity N does not match the query and can be skipped. If N is greater than or equal to the number of entities in the chunk, bit N will always be clear. If `useEnabledMask` is false, all entities in the chunk match the query, and the contents of this mask are undefined.

Remarks

The chunks selected by the `Unity.Entities.EntityQuery` used to schedule the job are the input to your `Execute` function. The `Execute` function is called once per matching chunk.

Note that `unfilteredChunkIndex` is not necessarily guaranteed to be a zero-based, tightly-packed index into the chunks the job actually runs on. For example, if the query matches 100 chunks, but the query's uses `Unity.Entities.EntityQuery.SetSharedComponentFilter<SharedComponent>(SharedComponent)` and the first 50 chunks get filtered out, the `unfilteredChunkIndex` will range from 50 to 99. If the index relative to the filtered chunk list is required, use `Unity.Entities.EntityQuery.CalculateFilteredChunkIndexArray(Unity.CollectionsAllocatorManagerAllocatorHandle)`

Run()

```
public void Run()
```

Run(EntityQuery)

```
public void Run(EntityQuery query)
```

Parameters

`query` EntityQuery

RunByRef()

```
public void RunByRef()
```

RunByRef(EntityQuery)

```
public void RunByRef(EntityQuery query)
```

Parameters

query EntityQuery

Schedule()

```
public void Schedule()
```

Schedule(EntityQuery)

```
public void Schedule(EntityQuery query)
```

Parameters

query EntityQuery

Schedule(EntityQuery, JobHandle)

```
public JobHandle Schedule(EntityQuery query, JobHandle dependsOn)
```

Parameters

query EntityQuery

dependsOn JobHandle

Returns

JobHandle

Schedule(JobHandle)

```
public JobHandle Schedule(JobHandle dependsOn)
```

Parameters

`dependsOn` JobHandle

Returns

JobHandle

ScheduleByRef()

```
public void ScheduleByRef()
```

ScheduleByRef(EntityQuery)

```
public void ScheduleByRef(EntityQuery query)
```

Parameters

`query` EntityQuery

ScheduleByRef(EntityQuery, JobHandle)

```
public JobHandle ScheduleByRef(EntityQuery query, JobHandle dependsOn)
```

Parameters

`query` EntityQuery

`dependsOn` JobHandle

Returns

JobHandle

ScheduleByRef(JobHandle)

```
public JobHandle ScheduleByRef(JobHandle dependsOn)
```

Parameters

`dependsOn` JobHandle

Returns

JobHandle

ScheduleParallel()

```
public void ScheduleParallel()
```

ScheduleParallel(EntityQuery)

```
public void ScheduleParallel(EntityQuery query)
```

Parameters

`query` EntityQuery

ScheduleParallel(EntityQuery, JobHandle)

```
public JobHandle ScheduleParallel(EntityQuery query, JobHandle dependsOn)
```

Parameters

query EntityQuery

dependsOn JobHandle

Returns

JobHandle

ScheduleParallel(EntityQuery, JobHandle, NativeArray<int>)

```
public JobHandle ScheduleParallel(EntityQuery query, JobHandle dependsOn,  
NativeArray<int> chunkBaseEntityIndices)
```

Parameters

query EntityQuery

dependsOn JobHandle

chunkBaseEntityIndices NativeArray<int>

Returns

JobHandle

ScheduleParallel(JobHandle)

```
public JobHandle ScheduleParallel(JobHandle dependsOn)
```

Parameters

dependsOn JobHandle

Returns

JobHandle

ScheduleParallelByRef()

```
public void ScheduleParallelByRef()
```

ScheduleParallelByRef(EntityQuery)

```
public void ScheduleParallelByRef(EntityQuery query)
```

Parameters

`query` EntityQuery

ScheduleParallelByRef(EntityQuery, JobHandle)

```
public JobHandle ScheduleParallelByRef(EntityQuery query, JobHandle dependsOn)
```

Parameters

`query` EntityQuery

`dependsOn` JobHandle

Returns

JobHandle

ScheduleParallelByRef(EntityQuery, JobHandle, NativeArray<int>)

```
public JobHandle ScheduleParallelByRef(EntityQuery query, JobHandle dependsOn,  
NativeArray<int> chunkBaseEntityIndices)
```

Parameters

query EntityQuery

dependsOn JobHandle

chunkBaseEntityIndices NativeArray<int>

Returns

JobHandle

ScheduleParallelByRef(JobHandle)

```
public JobHandle ScheduleParallelByRef(JobHandle dependsOn)
```

Parameters

dependsOn JobHandle

Returns

JobHandle

Struct DOTSDynamicBoneAnimatedLocal ToXPreUpdateSystem.SetAnimatedLocal ToXJob.InternalCompiler

Namespace: [DOTSDynamicBone.Systems](#)

Assembly: DOTSDynamicBone.dll

Internal structure used by the compiler

```
public struct  
DOTSDynamicBoneAnimatedLocalToXPreUpdateSystem.SetAnimatedLocalToXJob.InternalCompiler
```

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Methods

CheckForErrors(int)

```
[Conditional("ENABLE_UNITY_COLLECTIONS_CHECKS")]  
public static void CheckForErrors(int scheduleType)
```

Parameters

scheduleType [int](#)

Struct DOTSDynamicBoneAnimatedLocal ToXPreUpdateSystem.SetAnimatedLocal ToXJob.InternalCompilerQueryAndHandleData

Namespace: [DOTSDynamicBone.Systems](#)

Assembly: DOTSDynamicBone.dll

Used internally by the compiler, we won't promise this exists in the future

```
public struct  
DOTSDynamicBoneAnimatedLocalToXPreUpdateSystem.SetAnimatedLocalToXJob.InternalCompilerQueryA  
ndHandleData
```

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Fields

DefaultQuery

```
public EntityQuery DefaultQuery
```

Field Value

EntityQuery

__TypeHandle

```
public  
DOTSDynamicBoneAnimatedLocalToXPreUpdateSystem.SetAnimatedLocalToXJob.InternalCompilerQueryA  
ndHandleData.TypeHandle __TypeHandle
```

Field Value

[DOTSDynamicBoneAnimatedLocalToXPreUpdateSystem.SetAnimatedLocalToXJob.InternalCompilerQueryAndHandleData.TypeHandle](#)

Methods

AssignEntityManager(ref SetAnimatedLocalToXJob, EntityManager)

```
public void AssignEntityManager(ref  
DOTSDynamicBoneAnimatedLocalToXPreUpdateSystem.SetAnimatedLocalToXJob job,  
EntityManager entityManager)
```

Parameters

job [DOTSDynamicBoneAnimatedLocalToXPreUpdateSystem.SetAnimatedLocalToXJob](#)
entityManager EntityManager

Init(ref SystemState, bool)

```
public void Init(ref SystemState state, bool assignDefaultQuery)
```

Parameters

state SystemState
assignDefaultQuery [bool](#)

Run(ref SetAnimatedLocalToXJob, EntityQuery)

```
public void Run(ref DOTSDynamicBoneAnimatedLocalToXPreUpdateSystem.SetAnimatedLocalToXJob  
job, EntityQuery query)
```

Parameters

job [DOTSDynamicBoneAnimatedLocalToXPreUpdateSystem.SetAnimatedLocalToXJob](#)

query EntityQuery

Schedule(ref SetAnimatedLocalToXJob, EntityQuery, JobHandle)

```
public JobHandle Schedule(ref  
DOTSDynamicBoneAnimatedLocalToXPreUpdateSystem.SetAnimatedLocalToXJob job, EntityQuery  
query, JobHandle dependency)
```

Parameters

job [DOTSDynamicBoneAnimatedLocalToXPreUpdateSystem.SetAnimatedLocalToXJob](#)

query EntityQuery

dependency JobHandle

Returns

JobHandle

ScheduleParallel(ref SetAnimatedLocalToXJob, EntityQuery, Job Handle)

```
public JobHandle ScheduleParallel(ref  
DOTSDynamicBoneAnimatedLocalToXPreUpdateSystem.SetAnimatedLocalToXJob job, EntityQuery  
query, JobHandle dependency)
```

Parameters

job [DOTSDynamicBoneAnimatedLocalToXPreUpdateSystem.SetAnimatedLocalToXJob](#)

query EntityQuery

dependency JobHandle

Returns

JobHandle

UpdateBaseEntityIndexArray(ref SetAnimatedLocalToXJob, EntityQuery, ref SystemState)

```
public void Update BaseEntityIndexArray(ref  
DOTSDynamicBoneAnimatedLocalToXPreUpdateSystem.SetAnimatedLocalToXJob job, EntityQuery  
query, ref SystemState state)
```

Parameters

job [DOTSDynamicBoneAnimatedLocalToXPreUpdateSystem.SetAnimatedLocalToXJob](#)

query EntityQuery

state SystemState

UpdateBaseEntityIndexArray(ref SetAnimatedLocalToXJob, EntityQuery, JobHandle, ref SystemState)

```
public JobHandle Update BaseEntityIndexArray(ref  
DOTSDynamicBoneAnimatedLocalToXPreUpdateSystem.SetAnimatedLocalToXJob job, EntityQuery  
query, JobHandle dependency, ref SystemState state)
```

Parameters

job [DOTSDynamicBoneAnimatedLocalToXPreUpdateSystem.SetAnimatedLocalToXJob](#)

query EntityQuery

dependency JobHandle

state SystemState

Returns

JobHandle

Struct DOTSDynamicBoneAnimatedLocal ToXPreUpdateSystem.SetAnimatedLocal ToXJob.InternalCompilerQueryAndHandleData. TypeHandle

Namespace: [DOTSDynamicBone.Systems](#)

Assembly: DOTSDynamicBone.dll

```
public struct  
DOTSDynamicBoneAnimatedLocalToXPreUpdateSystem.SetAnimatedLocalToXJob.InternalCompilerQueryA  
ndHandleData.TypeHandle
```

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Fields

__DOTSDynamicBone_DOTSDynamicBone_RW_ComponentType
Handle

```
public ComponentTypeHandle<DOTSDynamicBone>  
__DOTSDynamicBone_DOTSDynamicBone_RW_ComponentTypeHandle
```

Field Value

ComponentTypeHandle<[DOTSDynamicBone](#)>

__DOTSDynamicBone_Particle_RW_BufferTypeHandle

```
public BufferTypeHandle<Particle> __DOTSDynamicBone_Particle_RW_BufferTypeHandle
```

Field Value

BufferTypeHandle<[Particle](#)>

__DOTSDynamicBone_Rig_AnimatedLocalToRoot_RW_BufferTypeHandle

```
public BufferTypeHandle<AnimatedLocalToRoot>
__DOTSDynamicBone_Rig_AnimatedLocalToRoot_RW_BufferTypeHandle
```

Field Value

BufferTypeHandle<[AnimatedLocalToRoot](#)>

__DOTSDynamicBone_Rig_AnimatedLocalToWorld_RW_BufferTypeHandle

```
public BufferTypeHandle<AnimatedLocalToWorld>
__DOTSDynamicBone_Rig_AnimatedLocalToWorld_RW_BufferTypeHandle
```

Field Value

BufferTypeHandle<[AnimatedLocalToWorld](#)>

Methods

Update(ref SystemState)

```
public void Update(ref SystemState state)
```

Parameters

state SystemState

__AssignHandles(ref SystemState)

```
public void __AssignHandles(ref SystemState state)
```

Parameters

state SystemState

Struct DOTSDynamicBoneCullingSystem

Namespace: [DOTSDynamicBone.Systems](#)

Assembly: DOTSDynamicBone.dll

This system updates the tracked bone.m_ReferenceObject transform

```
[WorldSystemFilter(WorldSystemFilterFlags.Default | WorldSystemFilterFlags.Editor,  
WorldSystemFilterFlags.Default)]  
[UpdateInGroup(typeof(DOTSDynamicBoneUpdateGroup))]  
[UpdateBefore(typeof(DOTSDynamicBoneUpdateSystem))]  
[UpdateAfter(typeof(DOTSDynamicBoneAnimatedLocalToXPreUpdateSystem))]  
[BurstCompile]  
public struct DOTSDynamicBoneCullingSystem : ISystem, ISystemCompilerGenerated
```

Implements

ISystem, ISystemCompilerGenerated

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Methods

OnCreate(ref SystemState)

Called when this system is created.

```
[BurstCompile]  
public void OnCreate(ref SystemState state)
```

Parameters

state SystemState

The Unity.Entities.SystemState backing this system instance

Remarks

Implement an **OnCreate** function to set up system resources when it is created.

OnCreate is invoked before the the first time `Unity.Entities.ISystemStartStop.OnStartRunning(ref Unity.Entities.SystemState)` and `Unity.Entities.ISystem.OnUpdate(ref Unity.Entities.SystemState)` are invoked.

OnCreateForCompiler(ref SystemState)

Generated by compilation pipeline and used internally.

```
public void OnCreateForCompiler(ref SystemState state)
```

Parameters

state SystemState

The `Unity.Entities.SystemState` backing this system instance

OnUpdate(ref SystemState)

Implement **OnUpdate** to perform the major work of this system.

```
[BurstCompile]  
public void OnUpdate(ref SystemState state)
```

Parameters

state SystemState

The `Unity.Entities.SystemState` backing this system instance

Remarks

By default, the system invokes `OnUpdate` once every frame on the main thread. To skip `OnUpdate` if all of the system's `[EntityQueries]` are empty, use the `[RequireMatchingQueriesForUpdateAttribute]`. To limit when `OnUpdate` is invoked, you can specify components that must exist, or queries that match specific Entities. To do this, call `Unity.Entities.SystemState.RequireForUpdate<T>()` or `Unity.Entities.SystemState.RequireForUpdate(Unity.Entities.EntityQuery)` in the system's `OnCreate` method. For more information, see `Unity.Entities.SystemState.ShouldRunSystem()`.

You can instantiate and schedule an `Unity.Entities.IJobChunk` instance; you can use the [C# Job System] or you can perform work on the main thread. If you call `Unity.Entities.EntityManager` methods that perform structural changes on the main thread, be sure to arrange the system order to minimize the performance impact of the resulting [sync points].

Struct DOTSDynamicBoneParticleResetOptions

Namespace: [DOTSDynamicBone.Systems](#)

Assembly: DOTSDynamicBone.dll

```
[Serializable]
[BurstCompile]
public struct DOTSDynamicBoneParticleResetOptions : IComponentData, IQueryTypeParameter
```

Implements

IComponentData, IQueryTypeParameter

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Fields

ResetALTR

```
public bool3 ResetALTR
```

Field Value

bool3

Resets the AnimatedLocalToRoot Data

ResetALTW

```
public bool3 ResetALTW
```

Field Value

bool3

Resets the AnimatedLocalToWorld Data

ResetLocalPosition

```
public bool3 ResetLocalPosition
```

Field Value

bool3

Resets the Local Position Data

ResetLocalRotation

```
public bool3 ResetLocalRotation
```

Field Value

bool3

Resets the Local Rotation Data

ResetMTransform

```
public bool3 ResetMTransform
```

Field Value

bool3

Resets the ParticleTransform Matrix

ResetPhysicsVelocity

```
public bool3 ResetPhysicsVelocity
```

Field Value

bool3

Resets the Physics Velocity Data

ResetWorldPosition

NOTE: bool3 x = whether to reset y = reset on every frame vs one time z = execute only on root particle

```
public bool3 ResetWorldPosition
```

Field Value

bool3

Resets the World PositionData

ResetWorldRotation

```
public bool3 ResetWorldRotation
```

Field Value

bool3

Resets the World Rotation Data

Methods

ExecuteResetALTR(in bool3, in int, in float3, in quaternion, ref DynamicBuffer<AnimatedLocalToRoot>, bool)

```
[BurstCompile]
public static void ExecuteResetALTR(in bool3 resetOptions, in int AnimatedLocalToRootIndex,
in float3 m_InitLocalPosition, in quaternion m_InitLocalRotation, ref
DynamicBuffer<AnimatedLocalToRoot> altr, bool isRootParticle)
```

Parameters

resetOptions bool3

AnimatedLocalToRootIndex [int](#)

m_InitLocalPosition float3

m_InitLocalRotation quaternion

altr DynamicBuffer<[AnimatedLocalToRoot](#)>

isRootParticle [bool](#)

ExecuteResetALTR(in bool3, ref DynamicBuffer<Particle>, ref DynamicBuffer<AnimatedLocalToRoot>)

```
[BurstCompile]
public static void ExecuteResetALTR(in bool3 resetOptions, ref DynamicBuffer<Particle>
particles, ref DynamicBuffer<AnimatedLocalToRoot> altr)
```

Parameters

resetOptions bool3

particles DynamicBuffer<[Particle](#)>

altr DynamicBuffer<[AnimatedLocalToRoot](#)>

ExecuteResetALTW(in bool3, in int, in float3, in quaternion, ref DynamicBuffer<AnimatedLocalToWorld>, bool)

```
[BurstCompile]
public static void ExecuteResetALTW(in bool3 resetOptions, in int AnimatedLocalToRootIndex,
```

```
in float3 m_InitWorldPosition, in quaternion m_InitWorldRotation, ref  
DynamicBuffer<AnimatedLocalToWorld> altw, bool isRootParticle)
```

Parameters

resetOptions bool3

AnimatedLocalToRootIndex [int](#)

m_InitWorldPosition float3

m_InitWorldRotation quaternion

altw DynamicBuffer<[AnimatedLocalToWorld](#)>

isRootParticle [bool](#)

ExecuteResetALTW(in bool3, ref DynamicBuffer<Particle>, ref DynamicBuffer<AnimatedLocalToWorld>)

```
[BurstCompile]  
public static void ExecuteResetALTW(in bool3 resetOptions, ref DynamicBuffer<Particle>  
particles, ref DynamicBuffer<AnimatedLocalToWorld> altw)
```

Parameters

resetOptions bool3

particles DynamicBuffer<[Particle](#)>

altw DynamicBuffer<[AnimatedLocalToWorld](#)>

ExecuteResetLocalTransform(in bool3, in bool3, ref Local Transform, in float3, in float3, in quaternion, in quaternion, in bool, bool)

```
[BurstCompile]  
public static void ExecuteResetLocalTransform(in bool3 ResetLocalPosition, in bool3  
ResetLocalRotation, ref LocalTransform lt, in float3 m_InitWorldPosition, in float3
```

```
m_InitLocalPosition, in quaternion m_InitWorldRotation, in quaternion m_InitLocalRotation,  
in bool UpdateTransformAsIfNoParent, bool isRootParticle)
```

Parameters

ResetLocalPosition bool3

ResetLocalRotation bool3

lt LocalTransform

m_InitWorldPosition float3

m_InitLocalPosition float3

m_InitWorldRotation quaternion

m_InitLocalRotation quaternion

UpdateTransformAsIfNoParent bool

isRootParticle bool

ExecuteResetMTransform(ref bool3, ref float3, ref quaternion, ref
float3, ref quaternion, ref LocalToWorld, in float3, in quaternion,
in float3, in quaternion, in bool)

[BurstCompile]

```
public static void ExecuteResetMTransform(ref bool3 resetOptionsResetMTransform, ref float3  
m_TransformPosition, ref quaternion m_TransformRotation, ref float3  
m_TransformLocalPosition, ref quaternion m_TransformLocalRotation, ref LocalToWorld  
m_TransformRawLTW, in float3 m_InitWorldPosition, in quaternion m_InitWorldRotation, in  
float3 m_InitLocalPosition, in quaternion m_InitLocalRotation, in bool isRootParticle)
```

Parameters

resetOptionsResetMTransform bool3

m_TransformPosition float3

m_TransformRotation quaternion

```
m_TransformLocalPosition float3  
m_TransformLocalRotation quaternion  
m_TransformRawLTW LocalToWorld  
m_InitWorldPosition float3  
m_InitWorldRotation quaternion  
m_InitLocalPosition float3  
m_InitLocalRotation quaternion  
isRootParticle bool
```

ExecuteResetWorldTransforms(in bool3, in bool3, in float3, in quaternion, ref LocalToWorld, bool)

```
[BurstCompile]  
public static void ExecuteResetWorldTransforms(in bool3 ResetWorldPosition, in bool3  
ResetWorldRotation, in float3 m_InitWorldPosition, in quaternion m_InitWorldRotation, ref  
LocalToWorld ltw, bool isRootParticle)
```

Parameters

```
ResetWorldPosition bool3  
ResetWorldRotation bool3  
m_InitWorldPosition float3  
m_InitWorldRotation quaternion  
ltw LocalToWorld  
isRootParticle bool
```

UpdateResetOptions()

```
[BurstCompile]
public void UpdateResetOptions()
```

UpdateResetOptions(ref bool3)

```
[BurstCompile]
public static void UpdateResetOptions(ref bool3 options)
```

Parameters

`options` bool3

UpdateResetOptions(ref bool3, ref bool3)

```
[BurstCompile]
public static void UpdateResetOptions(ref bool3 ResetLocalRotation, ref bool3
ResetLocalPosition, ref bool3 ResetALTR, ref bool3 ResetALTW, ref bool3 ResetMTransform, ref
bool3 ResetPhysicsVelocity, ref bool3 ResetWorldRotation, ref bool3 ResetWorldPosition)
```

Parameters

`ResetLocalRotation` bool3

`ResetLocalPosition` bool3

`ResetALTR` bool3

`ResetALTW` bool3

`ResetMTransform` bool3

`ResetPhysicsVelocity` bool3

`ResetWorldRotation` bool3

`ResetWorldPosition` bool3

Struct DOTSDynamicBoneParticleResetSystem

Namespace: [DOTSDynamicBone.Systems](#)

Assembly: DOTSDynamicBone.dll

```
[UpdateInGroup(typeof(SimulationSystemGroup))]
[UpdateBefore(typeof(TransformSystemGroup))]
[UpdateAfter(typeof(VariableRateSimulationSystemGroup))]
[WorldSystemFilter(WorldSystemFilterFlags.Default | WorldSystemFilterFlags.Editor,
WorldSystemFilterFlags.Default)]
[BurstCompile]
public struct DOTSDynamicBoneParticleResetSystem : ISystem, ISystemCompilerGenerated
```

Implements

ISystem, ISystemCompilerGenerated

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Methods

OnCreate(ref SystemState)

Called when this system is created.

```
public void OnCreate(ref SystemState state)
```

Parameters

state SystemState

The Unity.Entities.SystemState backing this system instance

Remarks

Implement an **OnCreate** function to set up system resources when it is created.

`OnCreate` is invoked before the first time `Unity.Entities.ISystemStartStop.OnStartRunning(ref Unity.Entities.SystemState)` and `Unity.Entities.ISystem.OnUpdate(ref Unity.Entities.SystemState)` are invoked.

OnCreateForCompiler(ref SystemState)

Generated by compilation pipeline and used internally.

```
public void OnCreateForCompiler(ref SystemState state)
```

Parameters

state SystemState

The `Unity.Entities.SystemState` backing this system instance

OnUpdate(ref SystemState)

Implement `OnUpdate` to perform the major work of this system.

```
[BurstCompile]  
public void OnUpdate(ref SystemState state)
```

Parameters

state SystemState

The `Unity.Entities.SystemState` backing this system instance

Remarks

By default, the system invokes `OnUpdate` once every frame on the main thread. To skip `OnUpdate` if all of the system's `[EntityQueries]` are empty, use the `[RequireMatchingQueriesForUpdateAttribute]`. To limit when `OnUpdate` is invoked, you can specify components that must exist, or queries that match specific Entities. To do this, call `Unity.Entities.SystemState.RequireForUpdate<T>()` or `Unity.Entities.SystemState.RequireForUpdate(Unity.Entities.EntityQuery)` in the system's `OnCreate` method. For more information, see `Unity.Entities.SystemState.ShouldRunSystem()`.

You can instantiate and schedule an `Unity.Entities.IJobChunk` instance; you can use the [C# Job System] or you can perform work on the main thread. If you call `Unity.Entities.EntityManager` methods that

perform structural changes on the main thread, be sure to arrange the system order to minimize the performance impact of the resulting [sync points].

Struct DOTSDynamicBoneParticleResetSystem.DOTSDynamicBoneParticleResetJobOptimized

Namespace: [DOTSDynamicBone.Systems](#)

Assembly: DOTSDynamicBone.dll

This Job is responsible for Resetting parameters related to DOTSDynamicBone. Since particles use the resetOptions in DDB, this must execute after Particles execute

```
[BurstCompile]
public struct DOTSDynamicBoneParticleResetSystem.DOTSDynamicBoneParticleResetJobOptimized : IJobEntity, IJobChunk
```

Implements

IJobEntity, IJobChunk

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Fields

GetDOTSDynamicBone

```
[ReadOnly]
public ComponentLookup<DOTSDynamicBone> GetDOTSDynamicBone
```

Field Value

ComponentLookup<[DOTSDynamicBone](#)>

GetParticles

```
[ReadOnly]
public BufferLookup<Particle> GetParticles
```

Field Value

BufferLookup<[Particle](#)>

Methods

Execute(in ArchetypeChunk, int, bool, in v128)

Implement the [Execute](#) function to perform a unit of work on an `Unity.Entities.ArchetypeChunk` representing a chunk.

```
public void Execute(in ArchetypeChunk chunk, int chunkIndexInQuery, bool useEnabledMask, in  
v128 chunkEnabledMask)
```

Parameters

chunk ArchetypeChunk

An object providing access to the entities within a chunk.

chunkIndexInQuery [int](#)

useEnabledMask [bool](#)

If true, the contents of `chunkEnabledMask` describe which entities in the chunk match the provided `Unity.Entities.EntityQuery` and should be processed by this job. If false, all entities in the chunk match the provided query, and the contents of `chunkEnabledMask` are undefined.

chunkEnabledMask v128

If bit N in this mask is set, entity N in `chunk` matches the `Unity.Entities.EntityQuery` used to schedule the job. If bit N is clear, entity N does not match the query and can be skipped. If N is greater than or equal to the number of entities in the chunk, bit N will always be clear. If `useEnabledMask` is false, all entities in the chunk match the query, and the contents of this mask are undefined.

Remarks

The chunks selected by the `Unity.Entities.EntityQuery` used to schedule the job are the input to your [Execute](#) function. The [Execute](#) function is called once per matching chunk.

Note that `unfilteredChunkIndex` is not necessarily guaranteed to be a zero-based, tightly-packed index into the chunks the job actually runs on. For example, if the query matches 100 chunks, but the query's

uses Unity.Entities.EntityQuery.SetSharedComponentFilter<SharedComponent>(SharedComponent) and the first 50 chunks get filtered out, the `unfilteredChunkIndex` will range from 50 to 99. If the index relative to the filtered chunk list is required, use Unity.Entities.EntityQuery.CalculateFilteredChunkIndex
Array(Unity.CollectionsAllocatorManagerAllocatorHandle)

Execute(ref LocalTransform, ref LocalToWorld, in DOTSDynamicBoneParticle, in DOTSDynamicBoneEntity)

```
public void Execute(ref LocalTransform lt, ref LocalToWorld ltw, in DOTSDynamicBoneParticle  
particleData, in DOTSDynamicBoneEntity ddbe)
```

Parameters

`lt` LocalTransform

`ltw` LocalToWorld

`particleData` [DOTSDynamicBoneParticle](#)

`ddbe` [DOTSDynamicBoneEntity](#)

Run()

```
public void Run()
```

Run(EntityQuery)

```
public void Run(EntityQuery query)
```

Parameters

`query` EntityQuery

RunByRef()

```
public void RunByRef()
```

RunByRef(EntityQuery)

```
public void RunByRef(EntityQuery query)
```

Parameters

query EntityQuery

Schedule()

```
public void Schedule()
```

Schedule(EntityQuery)

```
public void Schedule(EntityQuery query)
```

Parameters

query EntityQuery

Schedule(EntityQuery, JobHandle)

```
public JobHandle Schedule(EntityQuery query, JobHandle dependsOn)
```

Parameters

query EntityQuery

dependsOn JobHandle

Returns

JobHandle

Schedule(JobHandle)

```
public JobHandle Schedule(JobHandle dependsOn)
```

Parameters

`dependsOn` JobHandle

Returns

JobHandle

ScheduleByRef()

```
public void ScheduleByRef()
```

ScheduleByRef(EntityQuery)

```
public void ScheduleByRef(EntityQuery query)
```

Parameters

`query` EntityQuery

ScheduleByRef(EntityQuery, JobHandle)

```
public JobHandle ScheduleByRef(EntityQuery query, JobHandle dependsOn)
```

Parameters

`query` EntityQuery

`dependsOn` JobHandle

Returns

JobHandle

ScheduleByRef(JobHandle)

```
public JobHandle ScheduleByRef(JobHandle dependsOn)
```

Parameters

`dependsOn` JobHandle

Returns

JobHandle

ScheduleParallel()

```
public void ScheduleParallel()
```

ScheduleParallel(EntityQuery)

```
public void ScheduleParallel(EntityQuery query)
```

Parameters

`query` EntityQuery

ScheduleParallel(EntityQuery, JobHandle)

```
public JobHandle ScheduleParallel(EntityQuery query, JobHandle dependsOn)
```

Parameters

query EntityQuery

dependsOn JobHandle

Returns

JobHandle

ScheduleParallel(EntityQuery, JobHandle, NativeArray<int>)

```
public JobHandle ScheduleParallel(EntityQuery query, JobHandle dependsOn,  
NativeArray<int> chunkBaseEntityIndices)
```

Parameters

query EntityQuery

dependsOn JobHandle

chunkBaseEntityIndices NativeArray<int>

Returns

JobHandle

ScheduleParallel(JobHandle)

```
public JobHandle ScheduleParallel(JobHandle dependsOn)
```

Parameters

dependsOn JobHandle

Returns

JobHandle

ScheduleParallelByRef()

```
public void ScheduleParallelByRef()
```

ScheduleParallelByRef(EntityQuery)

```
public void ScheduleParallelByRef(EntityQuery query)
```

Parameters

`query` EntityQuery

ScheduleParallelByRef(EntityQuery, JobHandle)

```
public JobHandle ScheduleParallelByRef(EntityQuery query, JobHandle dependsOn)
```

Parameters

`query` EntityQuery

`dependsOn` JobHandle

Returns

JobHandle

ScheduleParallelByRef(EntityQuery, JobHandle, NativeArray<int>)

```
public JobHandle ScheduleParallelByRef(EntityQuery query, JobHandle dependsOn,  
NativeArray<int> chunkBaseEntityIndices)
```

Parameters

query EntityQuery

dependsOn JobHandle

chunkBaseEntityIndices NativeArray<int>

Returns

JobHandle

ScheduleParallelByRef(JobHandle)

```
public JobHandle ScheduleParallelByRef(JobHandle dependsOn)
```

Parameters

dependsOn JobHandle

Returns

JobHandle

Struct DOTSDynamicBoneParticleResetSystem. DOTSDynamicBoneParticleResetJobOptimized. InternalCompiler

Namespace: [DOTSDynamicBone.Systems](#)

Assembly: DOTSDynamicBone.dll

Internal structure used by the compiler

```
public struct  
DOTSDynamicBoneParticleResetSystem.DOTSDynamicBoneParticleResetJobOptimized.InternalCompiler
```

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Methods

CheckForErrors(int)

```
[Conditional("ENABLE_UNITY_COLLECTIONS_CHECKS")]  
public static void CheckForErrors(int scheduleType)
```

Parameters

scheduleType [int](#)

Struct DOTSDynamicBoneParticleResetSystem. DOTSDynamicBoneParticleResetJobOptimized. InternalCompilerQueryAndHandleData

Namespace: [DOTSDynamicBone.Systems](#)

Assembly: DOTSDynamicBone.dll

Used internally by the compiler, we won't promise this exists in the future

```
public struct  
DOTSDynamicBoneParticleResetSystem.DOTSDynamicBoneParticleResetJobOptimized.InternalCompiler  
QueryAndHandleData
```

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Fields

DefaultQuery

```
public EntityQuery DefaultQuery
```

Field Value

EntityQuery

__TypeHandle

```
public  
DOTSDynamicBoneParticleResetSystem.DOTSDynamicBoneParticleResetJobOptimized.InternalCompiler  
QueryAndHandleData.TypeHandle __TypeHandle
```

Field Value

[DOTSDynamicBoneParticleResetSystem](#).[DOTSDynamicBoneParticleResetJobOptimized](#).[InternalCompilerQueryAndHandleData](#).[TypeHandle](#)

Methods

AssignEntityManager(ref DOTSDynamicBoneParticleResetJobOptimized, EntityManager)

```
public void AssignEntityManager(ref  
DOTSDynamicBoneParticleResetSystem.DOTSDynamicBoneParticleResetJobOptimized job,  
EntityManager entityManager)
```

Parameters

job [DOTSDynamicBoneParticleResetSystem](#).[DOTSDynamicBoneParticleResetJobOptimized](#)
entityManager EntityManager

Init(ref SystemState, bool)

```
public void Init(ref SystemState state, bool assignDefaultQuery)
```

Parameters

state SystemState
assignDefaultQuery [bool](#)

Run(ref DOTSDynamicBoneParticleResetJobOptimized, EntityQuery)

```
public void Run(ref  
DOTSDynamicBoneParticleResetSystem.DOTSDynamicBoneParticleResetJobOptimized job,  
EntityQuery query)
```

Parameters

job [DOTSDynamicBoneParticleResetSystem.DOTSDynamicBoneParticleResetJobOptimized](#)
query EntityQuery

Schedule(ref DOTSDynamicBoneParticleResetJobOptimized,
EntityQuery, JobHandle)

```
public JobHandle Schedule(ref  
DOTSDynamicBoneParticleResetSystem.DOTSDynamicBoneParticleResetJobOptimized job, EntityQuery  
query, JobHandle dependency)
```

Parameters

job [DOTSDynamicBoneParticleResetSystem.DOTSDynamicBoneParticleResetJobOptimized](#)
query EntityQuery
dependency JobHandle

Returns

JobHandle

ScheduleParallel(ref DOTSDynamicBoneParticleResetJob
Optimized, EntityQuery, JobHandle)

```
public JobHandle ScheduleParallel(ref  
DOTSDynamicBoneParticleResetSystem.DOTSDynamicBoneParticleResetJobOptimized job, EntityQuery  
query, JobHandle dependency)
```

Parameters

job [DOTSDynamicBoneParticleResetSystem.DOTSDynamicBoneParticleResetJobOptimized](#)
query EntityQuery
dependency JobHandle

Returns

JobHandle

UpdateBaseEntityIndexArray(ref DOTSDynamicBoneParticleResetJobOptimized, EntityQuery, ref SystemState)

```
public void UpdateBaseEntityIndexArray(ref  
DOTSDynamicBoneParticleResetSystem.DOTSDynamicBoneParticleResetJobOptimized job, EntityQuery  
query, ref SystemState state)
```

Parameters

job [DOTSDynamicBoneParticleResetSystem.DOTSDynamicBoneParticleResetJobOptimized](#)

query EntityQuery

state SystemState

UpdateBaseEntityIndexArray(ref DOTSDynamicBoneParticleResetJobOptimized, EntityQuery, JobHandle, ref SystemState)

```
public JobHandle UpdateBaseEntityIndexArray(ref  
DOTSDynamicBoneParticleResetSystem.DOTSDynamicBoneParticleResetJobOptimized job, EntityQuery  
query, JobHandle dependency, ref SystemState state)
```

Parameters

job [DOTSDynamicBoneParticleResetSystem.DOTSDynamicBoneParticleResetJobOptimized](#)

query EntityQuery

dependency JobHandle

state SystemState

Returns

Struct DOTSDynamicBoneParticleResetSystem. DOTSDynamicBoneParticleResetJobOptimized. InternalCompilerQueryAndHandleData.Type Handle

Namespace: [DOTSDynamicBone.Systems](#)

Assembly: DOTSDynamicBone.dll

```
public struct  
DOTSDynamicBoneParticleResetSystem.DOTSDynamicBoneParticleResetJobOptimized.InternalCompiler  
QueryAndHandleData.TypeHandle
```

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Fields

_DOTSDynamicBone_DOTSDynamicBoneEntity_Shared ComponentTypeHandle

```
[ReadOnly]  
public SharedComponentTypeHandle<DOTSDynamicBoneEntity>  
_DOTSDynamicBone_DOTSDynamicBoneEntity_SharedComponentTypeHandle
```

Field Value

SharedComponentTypeHandle<[DOTSDynamicBoneEntity](#)>

_DOTSDynamicBone_DOTSDynamicBoneParticle_RO_ ComponentTypeHandle

```
[ReadOnly]  
public ComponentTypeHandle<DOTSDynamicBoneParticle>
```

```
__DOTSDynamicBone_DOTSDynamicBoneParticle_RO_ComponentTypeHandle
```

Field Value

```
ComponentTypeHandle<DOTSDynamicBoneParticle>
```

__Unity_Transforms_LocalToWorld_RW_ComponentTypeHandle

```
public ComponentTypeHandle<LocalToWorld>
__Unity_Transforms_LocalToWorld_RW_ComponentTypeHandle
```

Field Value

```
ComponentTypeHandle<LocalToWorld>
```

__Unity_Transforms_LocalTransform_RW_ComponentTypeHandle

```
public ComponentTypeHandle<LocalTransform>
__Unity_Transforms_LocalTransform_RW_ComponentTypeHandle
```

Field Value

```
ComponentTypeHandle<LocalTransform>
```

Methods

Update(ref SystemState)

```
public void Update(ref SystemState state)
```

Parameters

state SystemState

__AssignHandles(ref SystemState)

```
public void __AssignHandles(ref SystemState state)
```

Parameters

state SystemState

Class DOTSDynamicBoneUpdateGroup

Namespace: [DOTSDynamicBone.Systems](#)

Assembly: DOTSDynamicBone.dll

A system group containing systems that process DOTSDynamicBone updates

```
[WorldSystemFilter(WorldSystemFilterFlags.Default | WorldSystemFilterFlags.Editor,  
WorldSystemFilterFlags.Default)]  
[UpdateInGroup(typeof(SimulationSystemGroup))]  
[UpdateAfter(typeof(TransformSystemGroup))]  
public class DOTSDynamicBoneUpdateGroup : ComponentSystemGroup
```

Inheritance

[object](#) ← ComponentSystemBase ← SystemBase ← ComponentSystemGroup ← DOTSDynamicBoneUpdateGroup

Inherited Members

ComponentSystemGroup.EnableSystemSorting , ComponentSystemGroup.Created ,
ComponentSystemGroup.ManagedSystems ,
ComponentSystemGroup.GetUnmanagedSystems(Allocator) ,
ComponentSystemGroup.GetAllSystems(Allocator) , ComponentSystemGroup.OnCreate() ,
ComponentSystemGroup.OnDestroy() ,
ComponentSystemGroup.AddSystemToUpdateList(ComponentSystemBase) ,
ComponentSystemGroup.AddSystemToUpdateList(SystemHandle) ,
ComponentSystemGroup.RemoveSystemFromUpdateList(ComponentSystemBase) ,
ComponentSystemGroup.RemoveSystemFromUpdateList(SystemHandle) ,
ComponentSystemGroup.SortSystems() , ComponentSystemGroup.FixedRateManager ,
ComponentSystemGroup.RateManager ,
ComponentSystemGroup.SetRateManagerCreateAllocator(IRateManager) ,
ComponentSystemGroup.RateGroupAllocators , ComponentSystemGroup.OnUpdate() ,
SystemBase.Dependency , SystemBase.CheckedStateRef , SystemBase.CompleteDependency() ,
SystemBase.Entities , SystemBase.Job , SystemBase.Update() , SystemBase.GetComponent<T>(Entity) ,
SystemBase.SetComponent<T>(Entity, T) , SystemBase.HasComponent<T>(Entity) ,
SystemBase.HasBuffer<T>(Entity) , [SystemBase.GetComponentLookup<T>\(bool\)](#) ,
[SystemBase.GetComponentDataFromEntity<T>\(bool\)](#) , [SystemBase.GetBuffer<T>\(Entity, bool\)](#) ,
[SystemBase.GetBufferLookup<T>\(bool\)](#) , [SystemBase.GetBufferFromEntity<T>\(bool\)](#) ,
SystemBase.GetEntityStorageInfoLookup() , SystemBase.GetStorageInfoFromEntity() ,
SystemBase.Exists(Entity) , ComponentSystemBase.Enabled , ComponentSystemBase.EntityQueries ,
ComponentSystemBase.GlobalSystemVersion , ComponentSystemBase.LastSystemVersion ,

ComponentSystemBase.EntityManager , ComponentSystemBase.World ,
ComponentSystemBase.SystemHandle , ComponentSystemBase.SystemHandleUntyped ,
ComponentSystemBase.Time , ComponentSystemBase.WorldUpdateAllocator ,
ComponentSystemBase.OnStartRunning() , ComponentSystemBase.OnStopRunning() ,
ComponentSystemBase.ShouldRunSystem() ,
[ComponentSystemBase.GetComponentTypeHandle<T>\(bool\)](#) ,
ComponentSystemBase.GetDynamicComponentTypeHandle(ComponentType) ,
[ComponentSystemBase.GetBufferTypeHandle<T>\(bool\)](#) ,
ComponentSystemBase.GetSharedComponentTypeHandle<T>() ,
ComponentSystemBase.GetDynamicSharedComponentTypeHandle(ComponentType) ,
ComponentSystemBase.GetEntityTypeHandle() , ComponentSystemBase.RequireForUpdate(EntityQuery) ,
ComponentSystemBase.RequireAnyForUpdate(params EntityQuery[]) ,
ComponentSystemBase.RequireAnyForUpdate(NativeArray<EntityQuery>) ,
ComponentSystemBase.RequireForUpdate<T>() ,
ComponentSystemBase.RequireSingletonForUpdate<T>() , ComponentSystemBase.HasSingleton<T>() ,
ComponentSystemBase.GetSingleton<T>() , ComponentSystemBase.GetSingletonRW<T>() ,
[ComponentSystemBase.GetSingletonBuffer<T>\(bool\)](#) ,
ComponentSystemBase.TryGetSingleton<T>(out T) ,
ComponentSystemBase.TryGetSingletonBuffer<T>(out DynamicBuffer<T>) ,
ComponentSystemBase.SetSingleton<T>(T) , ComponentSystemBase.GetSingletonEntity<T>() ,
ComponentSystemBase.TryGetSingletonEntity<T>(out Entity) ,
ComponentSystemBase.GetEntityQuery(params ComponentType[]) ,
ComponentSystemBase.GetEntityQuery(NativeArray<ComponentType>) ,
ComponentSystemBase.GetEntityQuery(params EntityQueryDesc[]) ,
ComponentSystemBase.GetEntityQuery(in EntityQueryBuilder) , [object.Equals\(object\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

Remarks

A system group containing systems that process DOTSDynamicBone updates

Struct DOTSDynamicBoneUpdateSystem

Namespace: [DOTSDynamicBone.Systems](#)

Assembly: DOTSDynamicBone.dll

```
[UpdateInGroup(typeof(DOTSDynamicBoneUpdateGroup))]
[WorldSystemFilter(WorldSystemFilterFlags.Default | WorldSystemFilterFlags.Editor,
WorldSystemFilterFlags.Default)]
[BurstCompile]
public struct DOTSDynamicBoneUpdateSystem : ISystem, ISystemCompilerGenerated
```

Implements

ISystem, ISystemCompilerGenerated

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Methods

OnCreate(ref SystemState)

Called when this system is created.

```
[BurstCompile]
public void OnCreate(ref SystemState state)
```

Parameters

state SystemState

The Unity.Entities.SystemState backing this system instance

Remarks

Implement an **OnCreate** function to set up system resources when it is created.

OnCreate is invoked before the the first time `Unity.Entities.ISystemStartStop.OnStartRunning(ref Unity.Entities.SystemState)` and `Unity.Entities.ISystem.OnUpdate(ref Unity.Entities.SystemState)` are invoked.

OnCreateForCompiler(ref SystemState)

Generated by compilation pipeline and used internally.

```
public void OnCreateForCompiler(ref SystemState state)
```

Parameters

state SystemState

The Unity.Entities.SystemState backing this system instance

OnUpdate(ref SystemState)

Implement **OnUpdate** to perform the major work of this system.

```
[BurstCompile]  
public void OnUpdate(ref SystemState state)
```

Parameters

state SystemState

The Unity.Entities.SystemState backing this system instance

Remarks

By default, the system invokes `OnUpdate` once every frame on the main thread. To skip OnUpdate if all of the system's [EntityQueries] are empty, use the [RequireMatchingQueriesForUpdateAttribute]. To limit when OnUpdate is invoked, you can specify components that must exist, or queries that match specific Entities. To do this, call `Unity.Entities.SystemState.RequireForUpdate<T>()` or `Unity.Entities.SystemState.RequireForUpdate(Unity.Entities.EntityQuery)` in the system's OnCreate method. For more information, see `Unity.Entities.SystemState.ShouldRunSystem()`.

You can instantiate and schedule an `Unity.Entities.IJobChunk` instance; you can use the [C# Job System] or you can perform work on the main thread. If you call `Unity.Entities.EntityManager` methods that perform structural changes on the main thread, be sure to arrange the system order to minimize the performance impact of the resulting [sync points].

Struct DOTSDynamicBoneUpdateSystem. DOTSDynamicBoneUpdateJob

Namespace: [DOTSDynamicBone.Systems](#)

Assembly: DOTSDynamicBone.dll

```
[BurstCompile]
public struct DOTSDynamicBoneUpdateSystem.DOTSDynamicBoneUpdateJob : IJobEntity, IJobChunk
```

Implements

IJobEntity, IJobChunk

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Fields

GetLTW

```
[ReadOnly]
public ComponentLookup<LocalToWorld> GetLTW
```

Field Value

ComponentLookup<LocalToWorld>

deltaTime

```
[ReadOnly]
public float deltaTime
```

Field Value

[float](#)

fixedDeltaTime

```
[ReadOnly]  
public float fixedDeltaTime
```

Field Value

[float](#)

unscaledDeltaTime

```
[ReadOnly]  
public float unscaledDeltaTime
```

Field Value

[float](#)

Methods

Execute(ref DOTSDynamicBone, ref DynamicBuffer<Particle>, ref DynamicBuffer<AnimatedLocalToRoot>, ref DynamicBuffer<AnimatedLocalToWorld>)

```
public void Execute(ref DOTSDynamicBone bone, ref DynamicBuffer<Particle> particles, ref DynamicBuffer<AnimatedLocalToRoot> altr, ref DynamicBuffer<AnimatedLocalToWorld> altw)
```

Parameters

bone [DOTSDynamicBone](#)

particles [DynamicBuffer<Particle>](#)

altr [DynamicBuffer<AnimatedLocalToRoot>](#)

Execute(in ArchetypeChunk, int, bool, in v128)

Implement the [Execute](#) function to perform a unit of work on an `Unity.Entities.ArchetypeChunk` representing a chunk.

```
public void Execute(in ArchetypeChunk chunk, int chunkIndexInQuery, bool useEnabledMask, in v128 chunkEnabledMask)
```

Parameters

chunk ArchetypeChunk

An object providing access to the entities within a chunk.

chunkIndexInQuery [int](#)

useEnabledMask [bool](#)

If true, the contents of `chunkEnabledMask` describe which entities in the chunk match the provided `Unity.Entities.EntityQuery` and should be processed by this job. If false, all entities in the chunk match the provided query, and the contents of `chunkEnabledMask` are undefined.

chunkEnabledMask v128

If bit N in this mask is set, entity N in `chunk` matches the `Unity.Entities.EntityQuery` used to schedule the job. If bit N is clear, entity N does not match the query and can be skipped. If N is greater than or equal to the number of entities in the chunk, bit N will always be clear. If `useEnabledMask` is false, all entities in the chunk match the query, and the contents of this mask are undefined.

Remarks

The chunks selected by the `Unity.Entities.EntityQuery` used to schedule the job are the input to your [Execute](#) function. The [Execute](#) function is called once per matching chunk.

Note that `unfilteredChunkIndex` is not necessarily guaranteed to be a zero-based, tightly-packed index into the chunks the job actually runs on. For example, if the query matches 100 chunks, but the query's uses `Unity.Entities.EntityQuery.SetSharedComponentFilter<SharedComponent>(SharedComponent)` and the first 50 chunks get filtered out, the `unfilteredChunkIndex` will range from 50 to 99. If the index relative to the filtered chunk list is required, use `Unity.Entities.EntityQuery.CalculateFilteredChunkIndexArray(Unity.CollectionsAllocatorManagerAllocatorHandle)`

Run()

```
public void Run()
```

Run(EntityQuery)

```
public void Run(EntityQuery query)
```

Parameters

query EntityQuery

RunByRef()

```
public void RunByRef()
```

RunByRef(EntityQuery)

```
public void RunByRef(EntityQuery query)
```

Parameters

query EntityQuery

Schedule()

```
public void Schedule()
```

Schedule(EntityQuery)

```
public void Schedule(EntityQuery query)
```

Parameters

query EntityQuery

Schedule(EntityQuery, JobHandle)

```
public JobHandle Schedule(EntityQuery query, JobHandle dependsOn)
```

Parameters

query EntityQuery

dependsOn JobHandle

Returns

JobHandle

Schedule(JobHandle)

```
public JobHandle Schedule(JobHandle dependsOn)
```

Parameters

dependsOn JobHandle

Returns

JobHandle

ScheduleByRef()

```
public void ScheduleByRef()
```

ScheduleByRef(EntityQuery)

```
public void ScheduleByRef(EntityQuery query)
```

Parameters

query EntityQuery

ScheduleByRef(EntityQuery, JobHandle)

```
public JobHandle ScheduleByRef(EntityQuery query, JobHandle dependsOn)
```

Parameters

query EntityQuery

dependsOn JobHandle

Returns

JobHandle

ScheduleByRef(JobHandle)

```
public JobHandle ScheduleByRef(JobHandle dependsOn)
```

Parameters

dependsOn JobHandle

Returns

JobHandle

ScheduleParallel()

```
public void ScheduleParallel()
```

ScheduleParallel(EntityQuery)

```
public void ScheduleParallel(EntityQuery query)
```

Parameters

query EntityQuery

ScheduleParallel(EntityQuery, JobHandle)

```
public JobHandle ScheduleParallel(EntityQuery query, JobHandle dependsOn)
```

Parameters

query EntityQuery

dependsOn JobHandle

Returns

JobHandle

ScheduleParallel(EntityQuery, JobHandle, NativeArray<int>)

```
public JobHandle ScheduleParallel(EntityQuery query, JobHandle dependsOn,  
NativeArray<int> chunkBaseEntityIndices)
```

Parameters

query EntityQuery

dependsOn JobHandle

```
chunkBaseEntityIndices NativeArray<int>
```

Returns

JobHandle

ScheduleParallel(JobHandle)

```
public JobHandle ScheduleParallel(JobHandle dependsOn)
```

Parameters

dependsOn JobHandle

Returns

JobHandle

ScheduleParallelByRef()

```
public void ScheduleParallelByRef()
```

ScheduleParallelByRef(EntityQuery)

```
public void ScheduleParallelByRef(EntityQuery query)
```

Parameters

query EntityQuery

ScheduleParallelByRef(EntityQuery, JobHandle)

```
public JobHandle ScheduleParallelByRef(EntityQuery query, JobHandle dependsOn)
```

Parameters

`query` EntityQuery

`dependsOn` JobHandle

Returns

JobHandle

ScheduleParallelByRef(EntityQuery, JobHandle,
NativeArray<int>)

```
public JobHandle ScheduleParallelByRef(EntityQuery query, JobHandle dependsOn,  
NativeArray<int> chunkBaseEntityIndices)
```

Parameters

`query` EntityQuery

`dependsOn` JobHandle

`chunkBaseEntityIndices` NativeArray<int>

Returns

JobHandle

ScheduleParallelByRef(JobHandle)

```
public JobHandle ScheduleParallelByRef(JobHandle dependsOn)
```

Parameters

`dependsOn` JobHandle

Returns

Struct DOTSDynamicBoneUpdateSystem. DOTSDynamicBoneUpdateJob.InternalCompiler

Namespace: [DOTSDynamicBone.Systems](#)

Assembly: DOTSDynamicBone.dll

Internal structure used by the compiler

```
public struct DOTSDynamicBoneUpdateSystem.DOTSDynamicBoneUpdateJob.InternalCompiler
```

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Methods

CheckForErrors(int)

```
[Conditional("ENABLE_UNITY_COLLECTIONS_CHECKS")]
public static void CheckForErrors(int scheduleType)
```

Parameters

scheduleType [int](#)

Struct DOTSDynamicBoneUpdateSystem. DOTSDynamicBoneUpdateJob.InternalCompiler QueryAndHandleData

Namespace: [DOTSDynamicBone.Systems](#)

Assembly: DOTSDynamicBone.dll

Used internally by the compiler, we won't promise this exists in the future

```
public struct  
DOTSDynamicBoneUpdateSystem.DOTSDynamicBoneUpdateJob.InternalCompilerQueryAndHandleData
```

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Fields

DefaultQuery

```
public EntityQuery DefaultQuery
```

Field Value

EntityQuery

__TypeHandle

```
public  
DOTSDynamicBoneUpdateSystem.DOTSDynamicBoneUpdateJob.InternalCompilerQueryAndHandleData.Type  
Handle __TypeHandle
```

Field Value

[DOTSDynamicBoneUpdateSystem.DOTSDynamicBoneUpdateJob.InternalCompilerQueryAndHandleData.TypeHandle](#)

Methods

AssignEntityManager(ref DOTSDynamicBoneUpdateJob, EntityManager)

```
public void AssignEntityManager(ref DOTSDynamicBoneUpdateSystem.DOTSDynamicBoneUpdateJob job, EntityManager entityManager)
```

Parameters

job [DOTSDynamicBoneUpdateSystem.DOTSDynamicBoneUpdateJob](#)

entityManager EntityManager

Init(ref SystemState, bool)

```
public void Init(ref SystemState state, bool assignDefaultQuery)
```

Parameters

state SystemState

assignDefaultQuery [**bool**](#)

Run(ref DOTSDynamicBoneUpdateJob, EntityQuery)

```
public void Run(ref DOTSDynamicBoneUpdateSystem.DOTSDynamicBoneUpdateJob job, EntityQuery query)
```

Parameters

job [DOTSDynamicBoneUpdateSystem.DOTSDynamicBoneUpdateJob](#)

query EntityQuery

Schedule(ref DOTSDynamicBoneUpdateJob, EntityQuery, JobHandle)

```
public JobHandle Schedule(ref DOTSDynamicBoneUpdateSystem.DOTSDynamicBoneUpdateJob job,  
EntityQuery query, JobHandle dependency)
```

Parameters

job [DOTSDynamicBoneUpdateSystem.DOTSDynamicBoneUpdateJob](#)

query EntityQuery

dependency JobHandle

Returns

JobHandle

ScheduleParallel(ref DOTSDynamicBoneUpdateJob, EntityQuery, JobHandle)

```
public JobHandle ScheduleParallel(ref DOTSDynamicBoneUpdateSystem.DOTSDynamicBoneUpdateJob  
job, EntityQuery query, JobHandle dependency)
```

Parameters

job [DOTSDynamicBoneUpdateSystem.DOTSDynamicBoneUpdateJob](#)

query EntityQuery

dependency JobHandle

Returns

JobHandle

UpdateBaseEntityIndexArray(ref DOTSDynamicBoneUpdateJob, EntityQuery, ref SystemState)

```
public void UpdateBaseEntityIndexArray(ref  
DOTSDynamicBoneUpdateSystem.DOTSDynamicBoneUpdateJob job, EntityQuery query, ref  
SystemState state)
```

Parameters

job [DOTSDynamicBoneUpdateSystem.DOTSDynamicBoneUpdateJob](#)

query EntityQuery

state SystemState

UpdateBaseEntityIndexArray(ref DOTSDynamicBoneUpdateJob, EntityQuery, JobHandle, ref SystemState)

```
public JobHandle UpdateBaseEntityIndexArray(ref  
DOTSDynamicBoneUpdateSystem.DOTSDynamicBoneUpdateJob job, EntityQuery query, JobHandle  
dependency, ref SystemState state)
```

Parameters

job [DOTSDynamicBoneUpdateSystem.DOTSDynamicBoneUpdateJob](#)

query EntityQuery

dependency JobHandle

state SystemState

Returns

JobHandle

Struct DOTSDynamicBoneUpdateSystem. DOTSDynamicBoneUpdateJob.InternalCompiler QueryAndHandleData.TypeHandle

Namespace: [DOTSDynamicBone.Systems](#)

Assembly: DOTSDynamicBone.dll

```
public struct  
DOTSDynamicBoneUpdateSystem.DOTSDynamicBoneUpdateJob.InternalCompilerQueryAndHandleData.Type  
Handle
```

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Fields

__DOTSDynamicBone__DOTSDynamicBone_RW_ComponentTypeHandle

```
public ComponentTypeHandle<DOTSDynamicBone>  
__DOTSDynamicBone__DOTSDynamicBone_RW_ComponentTypeHandle
```

Field Value

ComponentTypeHandle<[DOTSDynamicBone](#)>

__DOTSDynamicBone_Particle_RW_BufferTypeHandle

```
public BufferTypeHandle<Particle> __DOTSDynamicBone_Particle_RW_BufferTypeHandle
```

Field Value

BufferTypeHandle<[Particle](#)>

__DOTSDynamicBone_Rig_AnimatedLocalToRoot_RW_BufferTypeHandle

```
public BufferTypeHandle<AnimatedLocalToRoot>
__DOTSDynamicBone_Rig_AnimatedLocalToRoot_RW_BufferTypeHandle
```

Field Value

BufferTypeHandle<[AnimatedLocalToRoot](#)>

__DOTSDynamicBone_Rig_AnimatedLocalToWorld_RW_BufferTypeHandle

```
public BufferTypeHandle<AnimatedLocalToWorld>
__DOTSDynamicBone_Rig_AnimatedLocalToWorld_RW_BufferTypeHandle
```

Field Value

BufferTypeHandle<[AnimatedLocalToWorld](#)>

Methods

Update(ref SystemState)

```
public void Update(ref SystemState state)
```

Parameters

state SystemState

__AssignHandles(ref SystemState)

```
public void __AssignHandles(ref SystemState state)
```

Parameters

state SystemState

Struct DOTSDynamicBoneVisualPhysicalOverrideSystem

Namespace: [DOTSDynamicBone.Systems](#)

Assembly: DOTSDynamicBone.dll

```
[UpdateInGroup(typeof(DOTSDynamicBoneUpdateGroup))]
[WorldSystemFilter(WorldSystemFilterFlags.Default | WorldSystemFilterFlags.Editor,
WorldSystemFilterFlags.Default)]
[UpdateAfter(typeof(DOTSDynamicBoneUpdateSystem))]
[BurstCompile]
public struct DOTSDynamicBoneVisualPhysicalOverrideSystem :
ISystem, ISystemCompilerGenerated
```

Implements

ISystem, ISystemCompilerGenerated

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Methods

OnCreate(ref SystemState)

Called when this system is created.

```
[BurstCompile]
public void OnCreate(ref SystemState state)
```

Parameters

state SystemState

The Unity.Entities.SystemState backing this system instance

Remarks

Implement an **OnCreate** function to set up system resources when it is created.

OnCreate is invoked before the first time `Unity.Entities.ISystemStartStop.OnStartRunning(ref Unity.Entities.SystemState)` and `Unity.Entities.ISystem.OnUpdate(ref Unity.Entities.SystemState)` are invoked.

OnCreateForCompiler(ref SystemState)

Generated by compilation pipeline and used internally.

```
public void OnCreateForCompiler(ref SystemState state)
```

Parameters

state SystemState

The `Unity.Entities.SystemState` backing this system instance

OnUpdate(ref SystemState)

Implement **OnUpdate** to perform the major work of this system.

```
[BurstCompile]  
public void OnUpdate(ref SystemState state)
```

Parameters

state SystemState

The `Unity.Entities.SystemState` backing this system instance

Remarks

By default, the system invokes `OnUpdate` once every frame on the main thread. To skip `OnUpdate` if all of the system's `[EntityQueries]` are empty, use the `[RequireMatchingQueriesForUpdateAttribute]`. To limit when `OnUpdate` is invoked, you can specify components that must exist, or queries that match specific Entities. To do this, call `Unity.Entities.SystemState.RequireForUpdate<T>()` or `Unity.Entities.SystemState.RequireForUpdate(Unity.Entities.EntityQuery)` in the system's `OnCreate` method. For more information, see `Unity.Entities.SystemState.ShouldRunSystem()`.

You can instantiate and schedule an `Unity.Entities.IJobChunk` instance; you can use the [C# Job System] or you can perform work on the main thread. If you call `Unity.Entities.EntityManager` methods that perform structural changes on the main thread, be sure to arrange the system order to minimize the performance impact of the resulting [sync points].

Struct DOTSDynamicBoneVisualPhysicalOverrideSystem.UpdateXTransformJob

Namespace: [DOTSDynamicBone.Systems](#)

Assembly: DOTSDynamicBone.dll

```
[BurstCompile]
public struct DOTSDynamicBoneVisualPhysicalOverrideSystem.UpdateXTransformJob : IJobEntity, IJobChunk
```

Implements

IJobEntity, IJobChunk

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Fields

GetLT

```
public ComponentLookup<LocalTransform> GetLT
```

Field Value

ComponentLookup<LocalTransform>

GetLTW

```
public ComponentLookup<LocalToWorld> GetLTW
```

Field Value

ComponentLookup<LocalToWorld>

Methods

Execute(ref DOTSDynamicBone, DynamicBuffer<Particle>, DynamicBuffer<AnimatedLocalToRoot>, DynamicBuffer<AnimatedLocalToWorld>)

```
public void Execute(ref DOTSDynamicBone bone, DynamicBuffer<Particle> particles, DynamicBuffer<AnimatedLocalToRoot> altr, DynamicBuffer<AnimatedLocalToWorld> altw)
```

Parameters

bone [DOTSDynamicBone](#)

particles DynamicBuffer<[Particle](#)>

altr DynamicBuffer<[AnimatedLocalToRoot](#)>

altw DynamicBuffer<[AnimatedLocalToWorld](#)>

Execute(in ArchetypeChunk, int, bool, in v128)

Implement the **Execute** function to perform a unit of work on an `Unity.Entities.ArchetypeChunk` representing a chunk.

```
public void Execute(in ArchetypeChunk chunk, int chunkIndexInQuery, bool useEnabledMask, in v128 chunkEnabledMask)
```

Parameters

chunk ArchetypeChunk

An object providing access to the entities within a chunk.

chunkIndexInQuery [int](#)

useEnabledMask [bool](#)

If true, the contents of `chunkEnabledMask` describe which entities in the chunk match the provided `Unity.Entities.EntityQuery` and should be processed by this job. If false, all entities in the chunk match the provided query, and the contents of `chunkEnabledMask` are undefined.

chunkEnabledMask v128

If bit N in this mask is set, entity N in `chunk` matches the `Unity.Entities.EntityQuery` used to schedule the job. If bit N is clear, entity N does not match the query and can be skipped. If N is greater than or equal to the number of entities in the chunk, bit N will always be clear. If `useEnabledMask` is false, all entities in the chunk match the query, and the contents of this mask are undefined.

Remarks

The chunks selected by the `Unity.Entities.EntityQuery` used to schedule the job are the input to your `Execute` function. The `Execute` function is called once per matching chunk.

Note that `unfilteredChunkIndex` is not necessarily guaranteed to be a zero-based, tightly-packed index into the chunks the job actually runs on. For example, if the query matches 100 chunks, but the query's uses `Unity.Entities.EntityQuery.SetSharedComponentFilter<SharedComponent>(SharedComponent)` and the first 50 chunks get filtered out, the `unfilteredChunkIndex` will range from 50 to 99. If the index relative to the filtered chunk list is required, use `Unity.Entities.EntityQuery.CalculateFilteredChunkIndexArray(Unity.CollectionsAllocatorManagerAllocatorHandle)`

Run()

```
public void Run()
```

Run(EntityQuery)

```
public void Run(EntityQuery query)
```

Parameters

`query` EntityQuery

RunByRef()

```
public void RunByRef()
```

RunByRef(EntityQuery)

```
public void RunByRef(EntityQuery query)
```

Parameters

query EntityQuery

Schedule()

```
public void Schedule()
```

Schedule(EntityQuery)

```
public void Schedule(EntityQuery query)
```

Parameters

query EntityQuery

Schedule(EntityQuery, JobHandle)

```
public JobHandle Schedule(EntityQuery query, JobHandle dependsOn)
```

Parameters

query EntityQuery

dependsOn JobHandle

Returns

JobHandle

Schedule(JobHandle)

```
public JobHandle Schedule(JobHandle dependsOn)
```

Parameters

`dependsOn` JobHandle

Returns

JobHandle

ScheduleByRef()

```
public void ScheduleByRef()
```

ScheduleByRef(EntityQuery)

```
public void ScheduleByRef(EntityQuery query)
```

Parameters

`query` EntityQuery

ScheduleByRef(EntityQuery, JobHandle)

```
public JobHandle ScheduleByRef(EntityQuery query, JobHandle dependsOn)
```

Parameters

`query` EntityQuery

`dependsOn` JobHandle

Returns

JobHandle

ScheduleByRef(JobHandle)

```
public JobHandle ScheduleByRef(JobHandle dependsOn)
```

Parameters

`dependsOn` JobHandle

Returns

JobHandle

ScheduleParallel()

```
public void ScheduleParallel()
```

ScheduleParallel(EntityQuery)

```
public void ScheduleParallel(EntityQuery query)
```

Parameters

`query` EntityQuery

ScheduleParallel(EntityQuery, JobHandle)

```
public JobHandle ScheduleParallel(EntityQuery query, JobHandle dependsOn)
```

Parameters

`query` EntityQuery

`dependsOn` JobHandle

Returns

JobHandle

ScheduleParallel(EntityQuery, JobHandle, NativeArray<int>)

```
public JobHandle ScheduleParallel(EntityQuery query, JobHandle dependsOn,  
NativeArray<int> chunkBaseEntityIndices)
```

Parameters

`query` EntityQuery

`dependsOn` JobHandle

`chunkBaseEntityIndices` NativeArray<int>

Returns

JobHandle

ScheduleParallel(JobHandle)

```
public JobHandle ScheduleParallel(JobHandle dependsOn)
```

Parameters

`dependsOn` JobHandle

Returns

JobHandle

ScheduleParallelByRef()

```
public void ScheduleParallelByRef()
```

ScheduleParallelByRef(EntityQuery)

```
public void ScheduleParallelByRef(EntityQuery query)
```

Parameters

`query` EntityQuery

ScheduleParallelByRef(EntityQuery, JobHandle)

```
public JobHandle ScheduleParallelByRef(EntityQuery query, JobHandle dependsOn)
```

Parameters

`query` EntityQuery

`dependsOn` JobHandle

Returns

JobHandle

ScheduleParallelByRef(EntityQuery, JobHandle, NativeArray<int>)

```
public JobHandle ScheduleParallelByRef(EntityQuery query, JobHandle dependsOn,  
NativeArray<int> chunkBaseEntityIndices)
```

Parameters

`query` EntityQuery

`dependsOn` JobHandle

`chunkBaseEntityIndices` NativeArray<[int](#)>

Returns

JobHandle

ScheduleParallelByRef(JobHandle)

`public` JobHandle **ScheduleParallelByRef**(JobHandle dependsOn)

Parameters

`dependsOn` JobHandle

Returns

JobHandle

Struct DOTSDynamicBoneVisualPhysicalOverrideSystem.UpdateXTransformJob.InternalCompiler

Namespace: [DOTSDynamicBone.Systems](#)

Assembly: DOTSDynamicBone.dll

Internal structure used by the compiler

```
public struct  
DOTSDynamicBoneVisualPhysicalOverrideSystem.UpdateXTransformJob.InternalCompiler
```

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Methods

CheckForErrors(int)

```
[Conditional("ENABLE_UNITY_COLLECTIONS_CHECKS")]  
public static void CheckForErrors(int scheduleType)
```

Parameters

scheduleType [int](#)

Struct DOTSDynamicBoneVisualPhysicalOverrideSystem.UpdateXTransformJob.InternalCompilerQueryAndHandleData

Namespace: [DOTSDynamicBone.Systems](#)

Assembly: DOTSDynamicBone.dll

Used internally by the compiler, we won't promise this exists in the future

```
public struct  
DOTSDynamicBoneVisualPhysicalOverrideSystem.UpdateXTransformJob.InternalCompilerQueryAndHandleData
```

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Fields

DefaultQuery

```
public EntityQuery DefaultQuery
```

Field Value

EntityQuery

__TypeHandle

```
public  
DOTSDynamicBoneVisualPhysicalOverrideSystem.UpdateXTransformJob.InternalCompilerQueryAndHandleData.TypeHandle __TypeHandle
```

Field Value

[DOTSDynamicBoneVisualPhysicalOverrideSystem.UpdateXTransformJob.InternalCompilerQueryAndHandleData.TypeHandle](#)

Methods

AssignEntityManager(ref UpdateXTransformJob, EntityManager)

```
public void AssignEntityManager(ref  
DOTSDynamicBoneVisualPhysicalOverrideSystem.UpdateXTransformJob job,  
EntityManager entityManager)
```

Parameters

job [DOTSDynamicBoneVisualPhysicalOverrideSystem.UpdateXTransformJob](#)

entityManager EntityManager

Init(ref SystemState, bool)

```
public void Init(ref SystemState state, bool assignDefaultQuery)
```

Parameters

state SystemState

assignDefaultQuery [bool](#)

Run(ref UpdateXTransformJob, EntityQuery)

```
public void Run(ref DOTSDynamicBoneVisualPhysicalOverrideSystem.UpdateXTransformJob job,  
EntityQuery query)
```

Parameters

job [DOTSDynamicBoneVisualPhysicalOverrideSystem.UpdateXTransformJob](#)

query EntityQuery

Schedule(ref UpdateXTransformJob, EntityQuery, JobHandle)

```
public JobHandle Schedule(ref  
DOTSDynamicBoneVisualPhysicalOverrideSystem.UpdateXTransformJob job, EntityQuery query,  
JobHandle dependency)
```

Parameters

job [DOTSDynamicBoneVisualPhysicalOverrideSystem.UpdateXTransformJob](#)

query EntityQuery

dependency JobHandle

Returns

JobHandle

ScheduleParallel(ref UpdateXTransformJob, EntityQuery, JobHandle)

```
public JobHandle ScheduleParallel(ref  
DOTSDynamicBoneVisualPhysicalOverrideSystem.UpdateXTransformJob job, EntityQuery query,  
JobHandle dependency)
```

Parameters

job [DOTSDynamicBoneVisualPhysicalOverrideSystem.UpdateXTransformJob](#)

query EntityQuery

dependency JobHandle

Returns

JobHandle

UpdateBaseEntityIndexArray(ref UpdateXTransformJob, EntityQuery, ref SystemState)

```
public void UpdateBaseEntityIndexArray(ref  
DOTSDynamicBoneVisualPhysicalOverrideSystem.UpdateXTransformJob job, EntityQuery query, ref  
SystemState state)
```

Parameters

job [DOTSDynamicBoneVisualPhysicalOverrideSystem.UpdateXTransformJob](#)

query EntityQuery

state SystemState

UpdateBaseEntityIndexArray(ref UpdateXTransformJob, EntityQuery, JobHandle, ref SystemState)

```
public JobHandle UpdateBaseEntityIndexArray(ref  
DOTSDynamicBoneVisualPhysicalOverrideSystem.UpdateXTransformJob job, EntityQuery query,  
JobHandle dependency, ref SystemState state)
```

Parameters

job [DOTSDynamicBoneVisualPhysicalOverrideSystem.UpdateXTransformJob](#)

query EntityQuery

dependency JobHandle

state SystemState

Returns

JobHandle

Struct DOTSDynamicBoneVisualPhysical OverrideSystem.UpdateXTransformJob.Internal CompilerQueryAndHandleData.TypeHandle

Namespace: [DOTSDynamicBone.Systems](#)

Assembly: DOTSDynamicBone.dll

```
public struct  
DOTSDynamicBoneVisualPhysicalOverrideSystem.UpdateXTransformJob.InternalCompilerQueryAndHandleData.TypeHandle
```

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Fields

__DOTSDynamicBone__DOTSDynamicBone_RW_ComponentTypeHandle

```
public ComponentTypeHandle<DOTSDynamicBone>  
__DOTSDynamicBone__DOTSDynamicBone_RW_ComponentTypeHandle
```

Field Value

ComponentTypeHandle<[DOTSDynamicBone](#)>

__DOTSDynamicBone_Particle_RW_BufferTypeHandle

```
public BufferTypeHandle<Particle> __DOTSDynamicBone_Particle_RW_BufferTypeHandle
```

Field Value

BufferTypeHandle<[Particle](#)>

__DOTSDynamicBone_Rig_AnimatedLocalToRoot_RW_BufferTypeHandle

```
public BufferTypeHandle<AnimatedLocalToRoot>
__DOTSDynamicBone_Rig_AnimatedLocalToRoot_RW_BufferTypeHandle
```

Field Value

BufferTypeHandle<[AnimatedLocalToRoot](#)>

__DOTSDynamicBone_Rig_AnimatedLocalToWorld_RW_BufferTypeHandle

```
public BufferTypeHandle<AnimatedLocalToWorld>
__DOTSDynamicBone_Rig_AnimatedLocalToWorld_RW_BufferTypeHandle
```

Field Value

BufferTypeHandle<[AnimatedLocalToWorld](#)>

Methods

Update(ref SystemState)

```
public void Update(ref SystemState state)
```

Parameters

state SystemState

__AssignHandles(ref SystemState)

```
public void __AssignHandles(ref SystemState state)
```

Parameters

state SystemState

Namespace DOTSDynamicBone.Systems.Baking

Structs

[DOTSDynamicBoneInitializationSystem](#)

[DOTSDynamicBoneInitializationSystem.DOTSDynamicBoneInitializationJob](#)

[DOTSDynamicBoneInitializationSystem.DOTSDynamicBoneInitializationJob.InternalCompiler](#)

Internal structure used by the compiler

[DOTSDynamicBoneInitializationSystem.DOTSDynamicBoneInitializationJob.InternalCompilerQueryAndHandleData](#)

Used internally by the compiler, we won't promise this exists in the future

[DOTSDynamicBoneInitializationSystem.DOTSDynamicBoneInitializationJob.InternalCompilerQueryAndHandleData.TypeHandle](#)

Struct DOTSDynamicBoneInitializationSystem

Namespace: [DOTSDynamicBone.Systems.Baking](#)

Assembly: DOTSDynamicBone.dll

```
[WorldSystemFilter(WorldSystemFilterFlags.Default | WorldSystemFilterFlags.Editor,
WorldSystemFilterFlags.Default)]
[UpdateInGroup(typeof(InitializationSystemGroup))]
[UpdateAfter(typeof(SceneSystemGroup))]
[BurstCompile]
public struct DOTSDynamicBoneInitializationSystem : ISystem, ISystemStartStop,
ISystemCompilerGenerated
```

Implements

ISystem, ISystemStartStop, ISystemCompilerGenerated

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Methods

OnCreate(ref SystemState)

Called when this system is created.

```
[BurstCompile]
public void OnCreate(ref SystemState state)
```

Parameters

state SystemState

The Unity.Entities.SystemState backing this system instance

Remarks

Implement an **OnCreate** function to set up system resources when it is created.

`OnCreate` is invoked before the first time `Unity.Entities.ISystemStartStop.OnStartRunning(ref Unity.Entities.SystemState)` and `Unity.Entities.ISystem.OnUpdate(ref Unity.Entities.SystemState)` are invoked.

OnCreateForCompiler(ref SystemState)

Generated by compilation pipeline and used internally.

```
public void OnCreateForCompiler(ref SystemState state)
```

Parameters

state SystemState

The `Unity.Entities.SystemState` backing this system instance

OnStartRunning(ref SystemState)

Called before the first call to `OnUpdate` and when a system resumes updating after being stopped or disabled.

```
[BurstCompile]  
public void OnStartRunning(ref SystemState state)
```

Parameters

state SystemState

The `Unity.Entities.SystemState` backing this system instance

Remarks

If the `Unity.Entities.EntityQuery` objects defined for a system do not match any existing entities then the system skips updates until a successful match is found. Likewise, if you set `Unity.Entities.SystemState.Enabled` to false, then the system stops running. In both cases, `Unity.Entities.ISystemStartStop.OnStopRunning(ref Unity.Entities.SystemState)` is called when a running system stops updating; `OnStartRunning` is called when it starts updating again.

OnStopRunning(ref SystemState)

Called when this system stops running because no entities match the system's Unity.Entities.EntityQuery objects or because you change the system Unity.Entities.SystemState.Enabled property to false.

```
[BurstCompile]
public void OnStopRunning(ref SystemState state)
```

Parameters

state SystemState

The Unity.Entities.SystemState backing this system instance

Remarks

If the Unity.Entities.EntityQuery objects defined for a system do not match any existing entities then the system skips updating until a successful match is found. Likewise, if you set Unity.Entities.SystemState.Enabled to false, then the system stops running. In both cases, OnStopRunning is called when a running system stops updating; Unity.Entities.ISystemStartStop.OnStartRunning(ref Unity.Entities.SystemState) is called when it starts updating again.

OnUpdate(ref SystemState)

Implement **OnUpdate** to perform the major work of this system.

```
[BurstCompile]
public void OnUpdate(ref SystemState state)
```

Parameters

state SystemState

The Unity.Entities.SystemState backing this system instance

Remarks

By default, the system invokes `OnUpdate` once every frame on the main thread. To skip OnUpdate if all of the system's [EntityQueries] are empty, use the [RequireMatchingQueriesForUpdateAttribute]. To limit when OnUpdate is invoked, you can specify components that must exist, or queries that match specific

Entities. To do this, call `Unity.Entities.SystemState.RequireForUpdate<T>()` or `Unity.Entities.SystemState.RequireForUpdate(Unity.Entities.EntityQuery)` in the system's `OnCreate` method. For more information, see `Unity.Entities.SystemState.ShouldRunSystem()`.

You can instantiate and schedule an `Unity.Entities.IJobChunk` instance; you can use the [C# Job System] or you can perform work on the main thread. If you call `Unity.Entities.EntityManager` methods that perform structural changes on the main thread, be sure to arrange the system order to minimize the performance impact of the resulting [sync points].

Struct DOTSDynamicBoneInitializationSystem.DOTSDynamicBoneInitializationJob

Namespace: [DOTSDynamicBone.Systems.Baking](#)

Assembly: DOTSDynamicBone.dll

```
[BurstCompile]
public struct DOTSDynamicBoneInitializationSystem.DOTSDynamicBoneInitializationJob : IJobEntity, IJobChunk
```

Implements

IJobEntity, IJobChunk

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Fields

ecb

```
public EntityCommandBuffer ecb
```

Field Value

EntityCommandBuffer

Methods

Execute(in ArchetypeChunk, int, bool, in v128)

Implement the [Execute](#) function to perform a unit of work on an Unity.Entities.ArchetypeChunk representing a chunk.

```
public void Execute(in ArchetypeChunk chunk, int chunkIndexInQuery, bool useEnabledMask, in
```

```
v128 chunkEnabledMask)
```

Parameters

chunk ArchetypeChunk

An object providing access to the entities within a chunk.

chunkIndexInQuery [int](#)

useEnabledMask [bool](#)

If true, the contents of **chunkEnabledMask** describe which entities in the chunk match the provided Unity.Entities.EntityQuery and should be processed by this job. If false, all entities in the chunk match the provided query, and the contents of **chunkEnabledMask** are undefined.

chunkEnabledMask v128

If bit N in this mask is set, entity N in **chunk** matches the Unity.Entities.EntityQuery used to schedule the job. If bit N is clear, entity N does not match the query and can be skipped. If N is greater than or equal to the number of entities in the chunk, bit N will always be clear. If **useEnabledMask** is false, all entities in the chunk match the query, and the contents of this mask are undefined.

Remarks

The chunks selected by the Unity.Entities.EntityQuery used to schedule the job are the input to your **Execute** function. The **Execute** function is called once per matching chunk.

Note that **unfilteredChunkIndex** is not necessarily guaranteed to be a zero-based, tightly-packed index into the chunks the job actually runs on. For example, if the query matches 100 chunks, but the query's uses Unity.Entities.EntityQuery.SetSharedComponentFilter<SharedComponent>(SharedComponent) and the first 50 chunks get filtered out, the **unfilteredChunkIndex** will range from 50 to 99. If the index relative to the filtered chunk list is required, use Unity.Entities.EntityQuery.CalculateFilteredChunkIndex Array(Unity.CollectionsAllocatorManagerAllocatorHandle)

Execute(Entity, in DOTSDynamicBone, in DynamicBuffer<Particle>, in DOTSDynamicBoneInitializerTag)

```
public void Execute(Entity e, in DOTSDynamicBone bone, in DynamicBuffer<Particle> particles,  
in DOTSDynamicBoneInitializerTag tag)
```

Parameters

e Entity

bone [DOTSDynamicBone](#)

particles DynamicBuffer<[Particle](#)>

tag [DOTSDynamicBoneInitializerTag](#)

Run()

```
public void Run()
```

Run(EntityQuery)

```
public void Run(EntityQuery query)
```

Parameters

query EntityQuery

RunByRef()

```
public void RunByRef()
```

RunByRef(EntityQuery)

```
public void RunByRef(EntityQuery query)
```

Parameters

query EntityQuery

Schedule()

```
public void Schedule()
```

Schedule(EntityQuery)

```
public void Schedule(EntityQuery query)
```

Parameters

query EntityQuery

Schedule(EntityQuery, JobHandle)

```
public JobHandle Schedule(EntityQuery query, JobHandle dependsOn)
```

Parameters

query EntityQuery

dependsOn JobHandle

Returns

JobHandle

Schedule(JobHandle)

```
public JobHandle Schedule(JobHandle dependsOn)
```

Parameters

dependsOn JobHandle

Returns

JobHandle

ScheduleByRef()

```
public void ScheduleByRef()
```

ScheduleByRef(EntityQuery)

```
public void ScheduleByRef(EntityQuery query)
```

Parameters

query EntityQuery

ScheduleByRef(EntityQuery, JobHandle)

```
public JobHandle ScheduleByRef(EntityQuery query, JobHandle dependsOn)
```

Parameters

query EntityQuery

dependsOn JobHandle

Returns

JobHandle

ScheduleByRef(JobHandle)

```
public JobHandle ScheduleByRef(JobHandle dependsOn)
```

Parameters

`dependsOn` JobHandle

Returns

JobHandle

ScheduleParallel()

```
public void ScheduleParallel()
```

ScheduleParallel(EntityQuery)

```
public void ScheduleParallel(EntityQuery query)
```

Parameters

`query` EntityQuery

ScheduleParallel(EntityQuery, JobHandle)

```
public JobHandle ScheduleParallel(EntityQuery query, JobHandle dependsOn)
```

Parameters

`query` EntityQuery

`dependsOn` JobHandle

Returns

JobHandle

ScheduleParallel(EntityQuery, JobHandle, NativeArray<int>)

```
public JobHandle ScheduleParallel(EntityQuery query, JobHandle dependsOn,
NativeArray<int> chunkBaseEntityIndices)
```

Parameters

query EntityQuery

dependsOn JobHandle

chunkBaseEntityIndices NativeArray<int>

Returns

JobHandle

ScheduleParallel(JobHandle)

```
public JobHandle ScheduleParallel(JobHandle dependsOn)
```

Parameters

dependsOn JobHandle

Returns

JobHandle

ScheduleParallelByRef()

```
public void ScheduleParallelByRef()
```

ScheduleParallelByRef(EntityQuery)

```
public void ScheduleParallelByRef(EntityQuery query)
```

Parameters

query EntityQuery

ScheduleParallelByRef(EntityQuery, JobHandle)

```
public JobHandle ScheduleParallelByRef(EntityQuery query, JobHandle dependsOn)
```

Parameters

query EntityQuery

dependsOn JobHandle

Returns

JobHandle

ScheduleParallelByRef(EntityQuery, JobHandle, NativeArray<int>)

```
public JobHandle ScheduleParallelByRef(EntityQuery query, JobHandle dependsOn, NativeArray<int> chunkBaseEntityIndices)
```

Parameters

query EntityQuery

dependsOn JobHandle

chunkBaseEntityIndices NativeArray<[int](#)>

Returns

JobHandle

ScheduleParallelByRef(JobHandle)

```
public JobHandle ScheduleParallelByRef(JobHandle dependsOn)
```

Parameters

dependsOn JobHandle

Returns

JobHandle

Struct DOTSDynamicBoneInitializationSystem.DOTSDynamicBoneInitializationJob.InternalCompiler

Namespace: [DOTSDynamicBone.Systems.Baking](#)

Assembly: DOTSDynamicBone.dll

Internal structure used by the compiler

```
public struct  
DOTSDynamicBoneInitializationSystem.DOTSDynamicBoneInitializationJob.InternalCompiler
```

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Methods

CheckForErrors(int)

```
[Conditional("ENABLE_UNITY_COLLECTIONS_CHECKS")]  
public static void CheckForErrors(int scheduleType)
```

Parameters

scheduleType [int](#)

Struct DOTSDynamicBoneInitializationSystem.DOTSDynamicBoneInitializationJob.InternalCompilerQueryAndHandleData

Namespace: [DOTSDynamicBone.Systems.Baking](#)

Assembly: DOTSDynamicBone.dll

Used internally by the compiler, we won't promise this exists in the future

```
public struct  
DOTSDynamicBoneInitializationSystem.DOTSDynamicBoneInitializationJob.InternalCompilerQueryAn  
dHandleData
```

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Fields

DefaultQuery

```
public EntityQuery DefaultQuery
```

Field Value

EntityQuery

__TypeHandle

```
public  
DOTSDynamicBoneInitializationSystem.DOTSDynamicBoneInitializationJob.InternalCompilerQueryAn  
dHandleData.TypeHandle __TypeHandle
```

Field Value

[DOTSDynamicBoneInitializationSystem.DOTSDynamicBoneInitializationJob.InternalCompilerQueryAndHandleData.TypeHandle](#)

Methods

AssignEntityManager(ref DOTSDynamicBoneInitializationJob, EntityManager)

```
public void AssignEntityManager(ref  
DOTSDynamicBoneInitializationSystem.DOTSDynamicBoneInitializationJob job,  
EntityManager entityManager)
```

Parameters

job [DOTSDynamicBoneInitializationSystem.DOTSDynamicBoneInitializationJob](#)
entityManager EntityManager

Init(ref SystemState, bool)

```
public void Init(ref SystemState state, bool assignDefaultQuery)
```

Parameters

state SystemState
assignDefaultQuery [bool](#)

Run(ref DOTSDynamicBoneInitializationJob, EntityQuery)

```
public void Run(ref DOTSDynamicBoneInitializationSystem.DOTSDynamicBoneInitializationJob  
job, EntityQuery query)
```

Parameters

job [DOTSDynamicBoneInitializationSystem.DOTSDynamicBoneInitializationJob](#)

query EntityQuery

Schedule(ref DOTSDynamicBoneInitializationJob, EntityQuery, JobHandle)

```
public JobHandle Schedule(ref  
DOTSDynamicBoneInitializationSystem.DOTSDynamicBoneInitializationJob job, EntityQuery query,  
JobHandle dependency)
```

Parameters

job [DOTSDynamicBoneInitializationSystem.DOTSDynamicBoneInitializationJob](#)

query EntityQuery

dependency JobHandle

Returns

JobHandle

ScheduleParallel(ref DOTSDynamicBoneInitializationJob, EntityQuery, JobHandle)

```
public JobHandle ScheduleParallel(ref  
DOTSDynamicBoneInitializationSystem.DOTSDynamicBoneInitializationJob job, EntityQuery query,  
JobHandle dependency)
```

Parameters

job [DOTSDynamicBoneInitializationSystem.DOTSDynamicBoneInitializationJob](#)

query EntityQuery

dependency JobHandle

Returns

JobHandle

UpdateBaseEntityIndexArray(ref DOTSDynamicBoneInitializationJob, EntityQuery, ref SystemState)

```
public void UpdateBaseEntityIndexArray(ref  
DOTSDynamicBoneInitializationSystem.DOTSDynamicBoneInitializationJob job, EntityQuery query,  
ref SystemState state)
```

Parameters

job [DOTSDynamicBoneInitializationSystem.DOTSDynamicBoneInitializationJob](#)

query EntityQuery

state SystemState

UpdateBaseEntityIndexArray(ref DOTSDynamicBoneInitializationJob, EntityQuery, JobHandle, ref SystemState)

```
public JobHandle UpdateBaseEntityIndexArray(ref  
DOTSDynamicBoneInitializationSystem.DOTSDynamicBoneInitializationJob job, EntityQuery query,  
JobHandle dependency, ref SystemState state)
```

Parameters

job [DOTSDynamicBoneInitializationSystem.DOTSDynamicBoneInitializationJob](#)

query EntityQuery

dependency JobHandle

state SystemState

Returns

JobHandle

Struct DOTSDynamicBoneInitializationSystem.DOTSDynamicBoneInitializationJob.InternalCompilerQueryAndHandleData.TypeHandle

Namespace: [DOTSDynamicBone.Systems.Baking](#)

Assembly: DOTSDynamicBone.dll

```
public struct  
DOTSDynamicBoneInitializationSystem.DOTSDynamicBoneInitializationJob.InternalCompilerQueryAn  
dHandleData.TypeHandle
```

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Fields

__DOTSDynamicBone__DOTSDynamicBoneInitializerTag_RO_ComponentTypeHandle

```
[ReadOnly]  
public ComponentTypeHandle<DOTSDynamicBoneInitializerTag>  
__DOTSDynamicBone__DOTSDynamicBoneInitializerTag_RO_ComponentTypeHandle
```

Field Value

ComponentTypeHandle<[DOTSDynamicBoneInitializerTag](#)>

__DOTSDynamicBone__DOTSDynamicBone_RO_ComponentTypeHandle

```
[ReadOnly]  
public ComponentTypeHandle<DOTSDynamicBone>  
__DOTSDynamicBone__DOTSDynamicBone_RO_ComponentTypeHandle
```

Field Value

ComponentTypeHandle<[DOTSDynamicBone](#)>

__DOTSDynamicBone_Particle_RO_BufferTypeHandle

[ReadOnly]

```
public BufferTypeHandle<Particle> __DOTSDynamicBone_Particle_RO_BufferTypeHandle
```

Field Value

BufferTypeHandle<[Particle](#)>

__Unity_Entities_Entity_TypeHandle

[ReadOnly]

```
public EntityTypeHandle __Unity_Entities_Entity_TypeHandle
```

Field Value

EntityTypeHandle

Methods

Update(ref SystemState)

```
public void Update(ref SystemState state)
```

Parameters

state SystemState

__AssignHandles(ref SystemState)

```
public void __AssignHandles(ref SystemState state)
```

Parameters

state SystemState

Namespace Egg

Classes

[StatusCodes](#)

Class StatusCodes

Namespace: [Egg](#)

Assembly: EggExtensions.dll

```
public static class StatusCodes
```

Inheritance

[object](#) ← StatusCodes

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

Fields

A_OK

```
public const int A_OK = 0
```

Field Value

[int](#)

BINDING_HAS_NO_VALUE

```
public const int BINDING_HAS_NO_VALUE = -6
```

Field Value

[int](#)

BINDING_IS_EMPTY

```
public const int BINDING_IS_EMPTY = -3
```

Field Value

[int ↗](#)

BINDING_IS_NULL

```
public const int BINDING_IS_NULL = -5
```

Field Value

[int ↗](#)

CONTROL_SCHEME_IN_USE

```
public const int CONTROL_SCHEME_IN_USE = -2
```

Field Value

[int ↗](#)

CONTROL_SCHEME_NOT_FOUND

```
public const int CONTROL_SCHEME_NOT_FOUND = -1
```

Field Value

[int ↗](#)

FAILED_TO_GET_VALID_INPUT_CONTROL_SCHEME

```
public const int FAILED_TO_GET_VALID_INPUT_CONTROL_SCHEME = -4
```

Field Value

[int↗](#)

NO_CAMERA_IN_SCENE

```
public const int NO_CAMERA_IN_SCENE = -100
```

Field Value

[int↗](#)

OPTION_NOT_SETUP_YET

```
public const int OPTION_NOT_SETUP_YET = -200
```

Field Value

[int↗](#)

VALUE_NOT_FOUND_IN_ARRAY

```
public const int VALUE_NOT_FOUND_IN_ARRAY = -201
```

Field Value

[int↗](#)

Namespace Egg.Extensions

Classes

[PhysicsExcludeComponent](#)

Add this to your GameObject to add the Unity.Physics.PhysicsExclude tag to your entity during conversion

[PhysicsExcludeComponentBaker](#)

Structs

[PhysicsExcludeRequestSystem](#)

[PhysicsExcludeRequest Tag](#)

Tag used in the PhysicsExcludeRequestSystem to identify which entities to remove from the Physics systems

[PhysicsExclude Tag](#)

Class PhysicsExcludeComponent

Namespace: [Egg.Extensions](#)

Assembly: EggExtensions.dll

Add this to your GameObject to add the Unity.Physics.PhysicsExclude tag to your entity during conversion

```
public class PhysicsExcludeComponent : MonoBehaviour
```

Inheritance

[object](#) ← Object ← Component ← Behaviour ← MonoBehaviour ← PhysicsExcludeComponent

Inherited Members

MonoBehaviour.IsInvoking() , MonoBehaviour.CancelInvoke() , [MonoBehaviour.Invoke\(string, float\)](#) ,
[MonoBehaviour.InvokeRepeating\(string, float, float\)](#) , [MonoBehaviour.CancelInvoke\(string\)](#) ,
[MonoBehaviour.IsInvoking\(string\)](#) , [MonoBehaviour.StartCoroutine\(string\)](#) ,
[MonoBehaviour.StartCoroutine\(string, object\)](#) , [MonoBehaviour.StartCoroutine\(IEnumerator\)](#) ,
[MonoBehaviour.StartCoroutine_Auto\(IEnumerator\)](#) , [MonoBehaviour.StopCoroutine\(IEnumerator\)](#) ,
MonoBehaviour.StopCoroutine(Coroutine) , [MonoBehaviour.StopCoroutine\(string\)](#) ,
MonoBehaviour.StopAllCoroutines() , [MonoBehaviour.print\(object\)](#) ,
MonoBehaviour.destroyCancellationToken , MonoBehaviour.useGUILayout ,
MonoBehaviour.runInEditMode , Behaviour.enabled , Behaviour.isActiveAndEnabled ,
[Component.GetComponent\(Type\)](#) , Component.GetComponent<T>() ,
[Component.TryGetComponent\(Type, out Component\)](#) , Component.TryGetComponent<T>(out T) ,
[Component.GetComponent\(string\)](#) , [Component.GetComponentInChildren\(Type, bool\)](#) ,
[Component.GetComponentInChildren\(Type\)](#) , [Component.GetComponentInChildren<T>\(bool\)](#) ,
Component.GetComponentInChildren<T>() , [Component.GetComponentsInChildren\(Type, bool\)](#) ,
[Component.GetComponentsInChildren\(Type\)](#) , [Component.GetComponentsInChildren<T>\(bool\)](#) ,
[Component.GetComponentsInChildren<T>\(bool, List<T>\)](#) ,
Component.GetComponentsInChildren<T>() , [Component.GetComponentsInChildren<T>\(List<T>\)](#) ,
[Component.GetComponentInParent\(Type, bool\)](#) , [Component.GetComponentInParent\(Type\)](#) ,
[Component.GetComponentInParent<T>\(bool\)](#) , Component.GetComponentInParent<T>() ,
[Component.GetComponentsInParent\(Type, bool\)](#) , [Component.GetComponentsInParent\(Type\)](#) ,
[Component.GetComponentsInParent<T>\(bool\)](#) ,
[Component.GetComponentsInParent<T>\(bool, List<T>\)](#) , Component.GetComponentsInParent<T>() ,
[Component.GetComponents\(Type\)](#) , [Component.GetComponents\(Type, List<Component>\)](#) ,
[Component.GetComponents<T>\(List<T>\)](#) , Component.GetComponents<T>() ,
Component.GetComponentIndex() , [Component.CompareTag\(string\)](#) ,

[Component.SendMessageUpwards\(string, object, SendMessageOptions\)](#) ,
[Component.SendMessageUpwards\(string, object\)](#) , [Component.SendMessageUpwards\(string\)](#) ,
[Component.SendMessageUpwards\(string, SendMessageOptions\)](#) ,
[Component.SendMessage\(string, object\)](#) , [Component.SendMessage\(string\)](#) ,
[Component.SendMessage\(string, object, SendMessageOptions\)](#) ,
[Component.SendMessage\(string, SendMessageOptions\)](#) ,
[Component.BroadcastMessage\(string, object, SendMessageOptions\)](#) ,
[Component.BroadcastMessage\(string, object\)](#) , [Component.BroadcastMessage\(string\)](#) ,
[Component.BroadcastMessage\(string, SendMessageOptions\)](#) , Component.transform ,
Component.gameObject , Component.tag , Object.GetInstanceID() , Object.GetHashCode() ,
[Object.Equals\(object\)](#) , Object.InstantiateAsync<T>(T) , Object.InstantiateAsync<T>(T, Transform) ,
Object.InstantiateAsync<T>(T, Vector3, Quaternion) ,
Object.InstantiateAsync<T>(T, Transform, Vector3, Quaternion) , [Object.InstantiateAsync<T>\(T, int\)](#) ,
[Object.InstantiateAsync<T>\(T, int, Transform\)](#) ,
[Object.InstantiateAsync<T>\(T, int, Vector3, Quaternion\)](#) ,
[Object.InstantiateAsync<T>\(T, int, ReadOnlySpan<Vector3>, ReadOnlySpan<Quaternion>\)](#) ,
[Object.InstantiateAsync<T>\(T, int, Transform, Vector3, Quaternion\)](#) ,
[Object.InstantiateAsync<T>\(T, int, Transform, ReadOnlySpan<Vector3>, ReadOnlySpan<Quaternion>\)](#) ,
Object.Instantiate(Object, Vector3, Quaternion) ,
Object.Instantiate(Object, Vector3, Quaternion, Transform) , Object.Instantiate(Object) ,
Object.Instantiate(Object, Scene) , Object.Instantiate(Object, Transform) ,
[Object.Instantiate\(Object, Transform, bool\)](#) , Object.Instantiate<T>(T) ,
Object.Instantiate<T>(T, Vector3, Quaternion) ,
Object.Instantiate<T>(T, Vector3, Quaternion, Transform) , Object.Instantiate<T>(T, Transform) ,
[Object.Instantiate<T>\(T, Transform, bool\)](#) , [Object.Destroy\(Object, float\)](#) , Object.Destroy(Object) ,
[Object.DestroyImmediate\(Object, bool\)](#) , Object.DestroyImmediate(Object) ,
[Object.FindObjectsOfType\(Type\)](#) , [Object.FindObjectsOfType\(Type, bool\)](#) ,
[Object.FindObjectsByType\(Type, FindObjectsSortMode\)](#) ,
[Object.FindObjectsByType\(Type, FindObjectsInactive, FindObjectsSortMode\)](#) ,
Object.DontDestroyOnLoad(Object) , [Object.DestroyObject\(Object, float\)](#) ,
Object.DestroyObject(Object) , [Object.FindSceneObjectsOfType\(Type\)](#) ,
[Object.FindObjectsOfTypeIncludingAssets\(Type\)](#) , Object.FindObjectsOfType<T>() ,
Object.FindObjectsByType<T>(FindObjectsSortMode) , [Object.FindObjectsOfType<T>\(bool\)](#) ,
Object.FindObjectsByType<T>(FindObjectsInactive, FindObjectsSortMode) ,
ObjectFindObjectOfType<T>() , [Object.FindObjectType<T>\(bool\)](#) ,
Object.FindFirstObjectByType<T>() , Object.FindAnyObjectByType<T>() ,
Object.FindFirstObjectByType<T>(FindObjectsInactive) ,
Object.FindAnyObjectByType<T>(FindObjectsInactive) , [Object.FindObjectsOfTypeAll\(Type\)](#) ,
[Object.FindObjectType\(Type\)](#) , [Object.FindFirstObjectByType\(Type\)](#) ,
[Object.FindAnyObjectByType\(Type\)](#) , [Object.FindObjectType\(Type, bool\)](#) ,

[Object.FindFirstObjectByType\(Type, FindObjectsInactive\)](#) ,
[Object.FindAnyObjectByType\(Type, FindObjectsInactive\)](#) , Object.ToString() , Object.name ,
Object.hideFlags , [object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,
[object.ReferenceEquals\(object, object\)](#)

Class PhysicsExcludeComponentBaker

Namespace: [Egg Extensions](#)

Assembly: EggExtensions.dll

```
public class PhysicsExcludeComponentBaker : Baker<PhysicsExcludeComponent>
```

Inheritance

[object](#) ← IBaker ← Baker<[PhysicsExcludeComponent](#)> ← PhysicsExcludeComponentBaker

Inherited Members

IBaker.GetSceneGUID() , IBaker.GetComponent<T>() , IBaker.GetComponent<T>(Component) ,
IBaker.GetComponent<T>(GameObject) , [IBaker.GetComponents<T>\(List<T>\)](#) ,
[IBaker.GetComponents<T>\(Component, List<T>\)](#) ,
[IBaker.GetComponents<T>\(GameObject, List<T>\)](#) , IBaker.GetComponents<T>()
IBaker.GetComponents<T>(Component) , IBaker.GetComponents<T>(GameObject)
IBaker.GetComponentInParent<T>() , IBaker.GetComponentInParent<T>(Component) ,
IBaker.GetComponentInParent<T>(GameObject) , [IBaker.GetComponentsInParent<T>\(List<T>\)](#) ,
[IBaker.GetComponentsInParent<T>\(Component, List<T>\)](#) ,
[IBaker.GetComponentsInParent<T>\(GameObject, List<T>\)](#) , IBaker.GetComponentsInParent<T>()
IBaker.GetComponentsInParent<T>(Component) , IBaker.GetComponentsInParent<T>(GameObject) ,
IBaker.GetComponentInChildren<T>() , IBaker.GetComponentInChildren<T>(Component) ,
IBaker.GetComponentInChildren<T>(GameObject) , [IBaker.GetComponentsInChildren<T>\(List<T>\)](#) ,
[IBaker.GetComponentsInChildren<T>\(Component, List<T>\)](#) ,
[IBaker.GetComponentsInChildren<T>\(GameObject, List<T>\)](#) , IBaker.GetComponentsInChildren<T>()
IBaker.GetComponentsInChildren<T>(Component) ,
IBaker.GetComponentsInChildren<T>(GameObject) , IBaker.GetParent() , IBaker.GetParent(Component) ,
IBaker.GetParent(GameObject) , IBaker.GetParents() , IBaker.GetParents(Component) ,
IBaker.GetParents(GameObject) , [IBaker.GetParents\(List<GameObject>\)](#) ,
[IBaker.GetParents\(Component, List<GameObject>\)](#) ,
[IBaker.GetParents\(GameObject, List<GameObject>\)](#) , [IBaker.GetChild\(int\)](#) ,
[IBaker.GetChild\(Component, int\)](#) , [IBaker.GetChild\(GameObject, int\)](#) , [IBaker.GetChildren\(bool\)](#) ,
[IBaker.GetChildren\(Component, bool\)](#) , [IBaker.GetChildren\(GameObject, bool\)](#) ,
[IBaker.GetChildren\(List<GameObject>, bool\)](#) ,
[IBaker.GetChildren\(Component, List<GameObject>, bool\)](#) ,
[IBaker.GetChildren\(GameObject, List<GameObject>, bool\)](#) , IBaker.GetChildCount()
IBaker.GetChildCount(Component) , IBaker.GetChildCount(GameObject) , IBaker.GetName()
IBaker.GetName(Component) , IBaker.GetName(GameObject) , IBaker.GetLayer()
IBaker.GetLayer(Component) , IBaker.GetLayer(GameObject) , IBaker.GetTag()

IBaker.GetTag(Component) , IBaker.GetTag(GameObject) , IBaker.GetEntity() ,
IBaker.GetEntity(GameObject) , IBaker.GetEntity(Component) , IBaker.GetEntity(TransformUsageFlags) ,
IBaker.GetEntity(GameObject, TransformUsageFlags) ,
IBaker.GetEntity(Component, TransformUsageFlags) , IBaker.GetEntityWithoutDependency() ,
IBaker.IsActive() , IBaker.IsActive(Component) , IBaker.IsActive(GameObject) ,
IBaker.IsActiveAndEnabled() , IBaker.IsActiveAndEnabled(Component) , IBaker.IsStatic() ,
IBaker.IsStatic(Component) , IBaker.IsStatic(GameObject) , IBaker.IsClient() , IBaker.IsServer() ,
IBaker.DependsOn<T>(T) , IBaker.DependsOnComponentInParent<T>() ,
IBaker.DependsOnComponentInParent<T>(Component) ,
IBaker.DependsOnComponentInParent<T>(GameObject) ,
IBaker.DependsOnComponentsInParent<T>(Component) ,
IBaker.DependsOnComponentsInParent<T>(GameObject) ,
IBaker.DependsOnComponentInChildren<T>() ,
IBaker.DependsOnComponentInChildren<T>(Component) ,
IBaker.DependsOnComponentInChildren<T>(GameObject) ,
IBaker.DependsOnComponentsInChildren<T>() ,
IBaker.DependsOnComponentsInChildren<T>(GameObject) ,
IBaker.DependsOnComponentsInChildren<T>(Component) , IBaker.DependsOnLightBaking() ,
IBaker.AddBlobAsset<T>(ref BlobAssetReference<T>, out Hash128) ,
IBaker.AddBlobAssetWithCustomHash<T>(ref BlobAssetReference<T>, Hash128) ,
IBaker.TryGetBlobAssetReference<T>(Hash128, out BlobAssetReference<T>) ,
IBaker.AddComponent<T>() , IBaker.AddComponent<T>(in T) , IBaker.AddComponent<T>(Entity) ,
IBaker.AddComponent<T>(Entity, in T) , IBaker.AddComponent(ComponentType) ,
IBaker.AddComponent(Entity, ComponentType) , IBaker.AddComponent(in ComponentTypeSet) ,
IBaker.AddComponent(Entity, in ComponentTypeSet) , IBaker.AddComponentObject<T>(T) ,
IBaker.AddComponentObject<T>(Entity, T) , IBaker.AddSharedComponentManaged<T>(T) ,
IBaker.AddSharedComponentManaged<T>(Entity, T) , IBaker.AddSharedComponent<T>(T) ,
IBaker.AddSharedComponent<T>(Entity, T) , IBaker.AddBuffer<T>() , IBaker.AddBuffer<T>(Entity) ,
IBaker.SetComponent<T>(Entity, in T) , [IBaker.GetComponentEnabled<T>\(Entity, bool\)](#) ,
[IBaker.GetComponentEnabled<T>\(bool\)](#) , IBaker.SetSharedComponentManaged<T>(Entity, in T) ,
IBaker.SetSharedComponent<T>(Entity, in T) , IBaker.SetBuffer<T>() , IBaker.SetBuffer<T>(Entity) ,
IBaker.AppendToBuffer<T>(T) , IBaker.AppendToBuffer<T>(Entity, T) , IBaker.CreateAdditionalEntity() ,
[IBaker.CreateAdditionalEntity\(TransformUsageFlags, bool, string\)](#) ,
IBaker.RegisterPrefabForBaking(GameObject) , IBaker.AddTransformUsageFlags(TransformUsageFlags) ,
IBaker.AddTransformUsageFlags(Entity, TransformUsageFlags) , IBaker.IsBakingForEditor() ,
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

Methods

Bake(PhysicsExcludeComponent)

Called in the baking process to bake the authoring component

```
public override void Bake(PhysicsExcludeComponent authoring)
```

Parameters

authoring [PhysicsExcludeComponent](#)

The authoring component to bake

Remarks

This method will be called for every instance of TAuthoringType in order to bake that instance.

Struct PhysicsExcludeRequestSystem

Namespace: [Egg.Extensions](#)

Assembly: EggExtensions.dll

```
[BurstCompile]
public struct PhysicsExcludeRequestSystem : ISystem, ISystemCompilerGenerated
```

Implements

ISystem, ISystemCompilerGenerated

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Methods

OnCreate(ref SystemState)

Called when this system is created.

```
[BurstCompile]
public void OnCreate(ref SystemState state)
```

Parameters

state SystemState

The Unity.Entities.SystemState backing this system instance

Remarks

Implement an **OnCreate** function to set up system resources when it is created.

OnCreate is invoked before the the first time `Unity.Entities.ISystemStartStop.OnStartRunning(ref Unity.Entities.SystemState)` and `Unity.Entities.ISystem.OnUpdate(ref Unity.Entities.SystemState)` are invoked.

OnCreateForCompiler(ref SystemState)

Generated by compilation pipeline and used internally.

```
public void OnCreateForCompiler(ref SystemState state)
```

Parameters

state SystemState

The Unity.Entities.SystemState backing this system instance

OnUpdate(ref SystemState)

Implement **OnUpdate** to perform the major work of this system.

```
[BurstCompile]  
public void OnUpdate(ref SystemState state)
```

Parameters

state SystemState

The Unity.Entities.SystemState backing this system instance

Remarks

By default, the system invokes `OnUpdate` once every frame on the main thread. To skip OnUpdate if all of the system's [EntityQueries] are empty, use the [RequireMatchingQueriesForUpdateAttribute]. To limit when OnUpdate is invoked, you can specify components that must exist, or queries that match specific Entities. To do this, call `Unity.Entities.SystemState.RequireForUpdate<T>()` or `Unity.Entities.SystemState.RequireForUpdate(Unity.Entities.EntityQuery)` in the system's OnCreate method. For more information, see `Unity.Entities.SystemState.ShouldRunSystem()`.

You can instantiate and schedule an `Unity.Entities.IJobChunk` instance; you can use the [C# Job System] or you can perform work on the main thread. If you call `Unity.Entities.EntityManager` methods that perform structural changes on the main thread, be sure to arrange the system order to minimize the performance impact of the resulting [sync points].

Struct PhysicsExcludeRequest_Tag

Namespace: [Egg.Extensions](#)

Assembly: EggExtensions.dll

Tag used in the PhysicsExcludeRequestSystem to identify which entities to remove from the Physics systems

```
public struct PhysicsExcludeRequest_Tag : IComponentData, IQueryTypeParameter
```

Implements

IComponentData, IQueryTypeParameter

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Struct PhysicsExclude_Tag

Namespace: [Egg.Extensions](#)

Assembly: EggExtensions.dll

```
public struct PhysicsExclude_Tag : IComponentData, IQueryTypeParameter
```

Implements

IComponentData, IQueryTypeParameter

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Namespace Egg.Extensions.Mathematics

Classes

[ManagedStaticMathematicsExtensions](#)

[MathematicsExtensions](#)

[StaticMathematicsExtensions](#)

Class ManagedStaticMathematicsExtensions

Namespace: [Egg.Extensions.Mathematics](#)

Assembly: EggExtensions.dll

```
public static class ManagedStaticMathematicsExtensions
```

Inheritance

[object](#) ← ManagedStaticMathematicsExtensions

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

Methods

ToFixedString(float4x4)

```
public static FixedString512Bytes ToFixedString(this float4x4 a)
```

Parameters

a float4x4

Returns

FixedString512Bytes

Class MathematicsExtensions

Namespace: [Egg.Extensions.Mathematics](#)

Assembly: EggExtensions.dll

```
[BurstCompile]
public class MathematicsExtensions
```

Inheritance

[object](#) ← MathematicsExtensions

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

Methods

ConvertToPersicion(float, int)

```
public static float ConvertToPersicion(float value, int percision)
```

Parameters

value [float](#)

percision [int](#)

Returns

[float](#)

ConvertToPersicion(in float3, int)

```
public static float3 ConvertToPersicion(in float3 value, int percision)
```

Parameters

value float3

percision [int](#)

Returns

float3

GetValue<T>(byte, float4)

```
public static T GetValue<T>(byte internalIndex, float4 Value)
```

Parameters

internalIndex [byte](#)

Value float4

Returns

T

Type Parameters

T

InRange(float, float, float)

returns true if x is greater than a and less than b

```
public static bool InRange(float x, float a, float b)
```

Parameters

x [float](#)

value to check

a [float](#)

a bound

b [float](#)

another bound

Returns

[bool](#)

RotatePointAroundPivot(in Vector3, in Vector3, in Quaternion, out Vector3)

[BurstCompile]

```
public static void RotatePointAroundPivot(in Vector3 point, in Vector3 pivot, in Quaternion  
angles, out Vector3 output)
```

Parameters

point Vector3

pivot Vector3

angles Quaternion

output Vector3

RotatePointAroundPivot(in Vector3, in Vector3, in Vector3, out Vector3)

[BurstCompile]

```
public static void RotatePointAroundPivot(in Vector3 point, in Vector3 pivot, in Vector3  
angles, out Vector3 output)
```

Parameters

point Vector3

pivot Vector3

angles Vector3

output Vector3

Class StaticMathematicsExtensions

Namespace: [Egg.Extensions.Mathematics](#)

Assembly: EggExtensions.dll

```
[BurstCompile]
public static class StaticMathematicsExtensions
```

Inheritance

[object](#) ← StaticMathematicsExtensions

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

Methods

Add(float4x4, in float4x4)

"Adds" a rotation to another as in start + diff = added_matrix

```
public static float4x4 Add(this float4x4 start, in float4x4 diff)
```

Parameters

start float4x4

diff float4x4

Returns

float4x4

Add(in float4x4, in float4x4, out float4x4)

[BurstCompiled] "Adds" a rotation to another as in start + diff = added_matrix

```
[BurstCompile]
public static void Add(in float4x4 start, in float4x4 diff, out float4x4 output)
```

Parameters

start float4x4

diff float4x4

output float4x4

Add(quaternion, in quaternion)

"Adds" a rotation to another as in start + diff = added_matrix

```
public static quaternion Add(this quaternion start, in quaternion diff)
```

Parameters

start quaternion

diff quaternion

Returns

quaternion

Add(**in** quaternion, **in** quaternion, **out** quaternion)

```
[BurstCompile]
public static void Add(in quaternion start, in quaternion diff, out quaternion output)
```

Parameters

start quaternion

diff quaternion

output quaternion

Add(Quaternion, in Quaternion)

"Adds" a rotation to another as in start + diff = added_matrix

```
public static Quaternion Add(this Quaternion start, in Quaternion diff)
```

Parameters

start Quaternion

diff Quaternion

Returns

Quaternion

Add(in Quaternion, in Quaternion, out Quaternion)

```
[BurstCompile]
```

```
public static void Add(in Quaternion start, in Quaternion diff, out Quaternion output)
```

Parameters

start Quaternion

diff Quaternion

output Quaternion

Diff(float4x4, in float4x4)

Gets the difference between 2 matrices

```
public static float4x4 Diff(this float4x4 to, in float4x4 from)
```

Parameters

to float4x4

what your rotating to

from float4x4

what your rotating from

Returns

float4x4

Diff(in float4x4, in float4x4, out float4x4)

```
[BurstCompile]
public static void Diff(in float4x4 to, in float4x4 from, out float4x4 output)
```

Parameters

to float4x4

from float4x4

output float4x4

Diff(quaternion, in quaternion)

Gets the difference between 2 quaternions

```
public static quaternion Diff(this quaternion to, in quaternion from)
```

Parameters

to quaternion

what your rotating to

from quaternion

what your rotating from

Returns

quaternion

Diff(in quaternion, in quaternion, out quaternion)

[BurstCompiled] Gets the difference between 2 quaternions

```
[BurstCompile]
public static void Diff(in quaternion to, in quaternion from, out quaternion result)
```

Parameters

to quaternion

what your rotating to

from quaternion

what your rotating from

result quaternion

Diff(Quaternion, in Quaternion)

Gets the difference between 2 quaternions

```
public static Quaternion Diff(this Quaternion to, in Quaternion from)
```

Parameters

to Quaternion

what your rotating to

from Quaternion

what your rotating from

Returns

Quaternion

Diff(in Quaternion, in Quaternion, out Quaternion)

```
[BurstCompile]
public static void Diff(in Quaternion to, in Quaternion from, out Quaternion output)
```

Parameters

to Quaternion

from Quaternion

output Quaternion

FromPositionRotation(ref float3x4, ref float3, ref quaternion)

[BurstCompiled] Create a 3x4 matrix representing both rotation and translation

```
[BurstCompile]
public static void FromPositionRotation(ref float3x4 transform, ref float3 position, ref
quaternion rotation)
```

Parameters

transform float3x4

transform to modify

position float3

rotation quaternion

MultiplyVector(ref float4x4, in float3, out float3)

[BurstCompiled] Transforms a direction by this matrix.

```
[BurstCompile]
public static void MultiplyVector(ref float4x4 matrix, in float3 vector, out float3 result)
```

Parameters

matrix float4x4

vector float3

result float3

Position(float4x4)

returns the position of the float4x4 if using it as a transform matrix

```
public static float3 Position(this float4x4 transform)
```

Parameters

transform float4x4

Returns

float3

position of transform matrix

Namespace Egg.Extensions.Transforms

Classes

[TransformsExtensions](#)

Enums

[TransformsExtensions.Bound](#)

[TransformsExtensions.Direction](#)

Class TransformsExtensions

Namespace: [Egg.Extensions.Transforms](#)

Assembly: EggExtensions.dll

```
public static class TransformsExtensions
```

Inheritance

[object](#) ← TransformsExtensions

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

Methods

Create(float3, quaternion)

```
public static LocalToWorld Create(float3 position, quaternion rotation)
```

Parameters

position float3

rotation quaternion

Returns

LocalToWorld

Create(float3, quaternion, float)

```
public static LocalToWorld Create(float3 position, quaternion rotation, float scale)
```

Parameters

```
position float3  
  
rotation quaternion  
  
scale float
```

Returns

LocalToWorld

Create(float3, quaternion, float3)

```
public static LocalToWorld Create(float3 position, quaternion rotation, float3 scale)
```

Parameters

```
position float3  
  
rotation quaternion  
  
scale float3
```

Returns

LocalToWorld

Create(float3, quaternion, float3, float3)

```
public static LocalToWorld Create(float3 position, quaternion rotation, float3 localScale,  
float3 lossyScale)
```

Parameters

```
position float3  
  
rotation quaternion  
  
localScale float3
```

lossyScale float3

Returns

LocalToWorld

FromToRotation(float3, float3)

```
public static quaternion FromToRotation(float3 from, float3 to)
```

Parameters

from float3

to float3

Returns

quaternion

GetTRSMatrix(RigidTransform, float3)

```
public static Matrix4x4 GetTRSMatrix(RigidTransform transform, float3 scale)
```

Parameters

transform RigidTransform

scale float3

Returns

Matrix4x4

GetTRSMatrix(float3, quaternion, float3)

```
public static Matrix4x4 GetTRSMatrix(float3 translation, quaternion rotation, float3 scale)
```

Parameters

translation float3

rotation quaternion

scale float3

Returns

Matrix4x4

GetTRSMatrix(Transform)

```
public static Matrix4x4 GetTRSMatrix(Transform transform)
```

Parameters

transform Transform

Returns

Matrix4x4

IsInvalidFloat3(float3)

checks if a float3 either has an Inf or a NaN.

```
public static bool IsInvalidFloat3(float3 f)
```

Parameters

f float3

float3 to check

Returns

[bool](#)

true if this float3 is invalid, false otherwise

IsInvalidQuaternion(quaternion)

Checks if all the float values are 0 or is any of them contain a NaN

```
public static bool IsInvalidQuaternion(quaternion q)
```

Parameters

q quaternion

quaternion to check

Returns

[bool](#)

true if this quaternion is invalid, false otherwise

RotateAround(float3, float3, quaternion)

```
public static float3 RotateAround(float3 position, float3 pivotPoint, quaternion rot)
```

Parameters

position float3

pivotPoint float3

rot quaternion

Returns

float3

RotateAround(LocalToWorld, float3, float3, float)

Rotates a transform around the given Pivot point by the given angle on the given axis.

```
public static LocalToWorld RotateAround(LocalToWorld transform, float3 pivotPoint, float3
axis, float angle)
```

Parameters

transform LocalToWorld

pivotPoint float3

axis float3

angle float

Returns

LocalToWorld

RotateAround(LocalToWorld, float3, quaternion)

Rotates a transform around the given Pivot point by the given rotation.

```
public static LocalToWorld RotateAround(LocalToWorld transform, float3 pivotPoint,
quaternion rot)
```

Parameters

transform LocalToWorld

pivotPoint float3

rot quaternion

Returns

LocalToWorld

RotateAround(Transform, Vector3, Quaternion)

Rotates a transform around the given Pivot point by the given rotation.

```
public static void RotateAround(Transform transform, Vector3 pivotPoint, Quaternion rot)
```

Parameters

transform Transform

pivotPoint Vector3

rot Quaternion

RotateAround(Transform, Vector3, Vector3, float)

Rotates a transform around the given Pivot point by the given angle on the given axis.

```
public static void RotateAround(Transform transform, Vector3 pivotPoint, Vector3 axis,
float angle)
```

Parameters

transform Transform

pivotPoint Vector3

axis Vector3

angle float

RotateDirectionVector(Vector3, Quaternion)

```
public static Vector3 RotateDirectionVector(Vector3 direction, Quaternion rot)
```

Parameters

direction Vector3

`rot` Quaternion

Returns

Vector3

RotateDirectionVector(Vector3, Vector3)

```
public static Vector3 RotateDirectionVector(Vector3 direction, Vector3 rotation)
```

Parameters

`direction` Vector3

`rotation` Vector3

Returns

Vector3

RotatePointAroundAxis(Vector3, float, Vector3)

```
public static Vector3 RotatePointAroundAxis(Vector3 point, float angle, Vector3 axis)
```

Parameters

`point` Vector3

`angle` [float](#)

`axis` Vector3

Returns

Vector3

RotatePointAroundPivot(float3, float3, quaternion)

```
public static Vector3 RotatePointAroundPivot(float3 point, float3 pivot, quaternion angles)
```

Parameters

point float3

pivot float3

angles quaternion

Returns

Vector3

RotatePointAroundPivot(Vector3, Vector3, Quaternion)

```
public static Vector3 RotatePointAroundPivot(Vector3 point, Vector3 pivot,  
Quaternion angles)
```

Parameters

point Vector3

pivot Vector3

angles Quaternion

Returns

Vector3

RotatePointAroundPivot(Vector3, Vector3, Vector3)

```
public static Vector3 RotatePointAroundPivot(Vector3 point, Vector3 pivot, Vector3 angles)
```

Parameters

point Vector3

pivot Vector3

angles Vector3

Returns

Vector3

ToEuler(float4x4)

```
public static float3 ToEuler(float4x4 q)
```

Parameters

q float4x4

Returns

float3

ToEuler(quaternion)

```
public static float3 ToEuler(quaternion q)
```

Parameters

q quaternion

Returns

float3

TransformDirection(quaternion, float3)

```
public static float3 TransformDirection(quaternion rot, float3 vec)
```

Parameters

rot quaternion

vec float3

Returns

float3

TransformDirection(Matrix4x4, float3)

```
public static float3 TransformDirection(Matrix4x4 a, float3 localDirectionDirection)
```

Parameters

a Matrix4x4

localDirectionDirection float3

Returns

float3

TransformPoint(RigidTransform, float3, float3)

```
public static float3 TransformPoint(RigidTransform transform, float3 vector,  
float3 localScale)
```

Parameters

transform RigidTransform

vector float3

```
localScale float3
```

Returns

```
float3
```

ZeroInvalidFloat3(float3)

```
public static float3 ZeroInvalidFloat3(float3 f)
```

Parameters

```
f float3
```

Returns

```
float3
```

toEuler(quaternion, RotationOrder)

```
public static float3 toEuler(quaternion q, math.RotationOrder order = RotationOrder.Default)
```

Parameters

```
q quaternion
```

```
order math.RotationOrder
```

Returns

```
float3
```

Enum TransformsExtensions.Bound

Namespace: [Egg.Extensions.Transforms](#)

Assembly: EggExtensions.dll

```
public enum TransformsExtensions.Bound
```

Fields

Inside = 1

Outside = 0

Enum TransformsExtensions.Direction

Namespace: [Egg.Extensions.Transforms](#)

Assembly: EggExtensions.dll

```
public enum TransformsExtensions.Direction
```

Fields

Dynamic = 3

X = 0

Y = 1

Z = 2

Namespace Unity.Entities.Animation

Classes

[AnimateBlendShapeAuthoring](#)

[AnimatePositionAuthoring](#)

[AnimateScaleAuthoring](#)

[ComputeSkinMatricesBakingSystem](#)

Class AnimateBlendShapeAuthoring

Namespace: [Unity.Entities.Animation](#)

Assembly: DOTSDynamicBone.dll

```
public class AnimateBlendShapeAuthoring : MonoBehaviour
```

Inheritance

[object](#) ← Object ← Component ← Behaviour ← MonoBehaviour ← AnimateBlendShapeAuthoring

Inherited Members

MonoBehaviour.IsInvoking() , MonoBehaviour.CancelInvoke() , [MonoBehaviour.Invoke\(string, float\)](#) ,
[MonoBehaviour.InvokeRepeating\(string, float, float\)](#) , [MonoBehaviour.CancelInvoke\(string\)](#) ,
[MonoBehaviour.IsInvoking\(string\)](#) , [MonoBehaviour.StartCoroutine\(string\)](#) ,
[MonoBehaviour.StartCoroutine\(string, object\)](#) , [MonoBehaviour.StartCoroutine\(IEnumerator\)](#) ,
[MonoBehaviour.StartCoroutine_Auto\(IEnumerator\)](#) , [MonoBehaviour.StopCoroutine\(IEnumerator\)](#) ,
MonoBehaviour.StopCoroutine(Coroutine) , [MonoBehaviour.StopCoroutine\(string\)](#) ,
MonoBehaviour.StopAllCoroutines() , [MonoBehaviour.print\(object\)](#) ,
MonoBehaviour.destroyCancellationToken , MonoBehaviour.useGUILayout ,
MonoBehaviour.runInEditMode , Behaviour.enabled , Behaviour.isActiveAndEnabled ,
[Component.GetComponent\(Type\)](#) , Component.GetComponent<T>() ,
[Component.TryGetComponent\(Type, out Component\)](#) , Component.TryGetComponent<T>(out T) ,
[Component.GetComponent\(string\)](#) , [Component.GetComponentInChildren\(Type, bool\)](#) ,
[Component.GetComponentInChildren\(Type\)](#) , [Component.GetComponentInChildren<T>\(bool\)](#) ,
Component.GetComponentInChildren<T>() , [Component.GetComponentsInChildren\(Type, bool\)](#) ,
[Component.GetComponentsInChildren\(Type\)](#) , [Component.GetComponentsInChildren<T>\(bool\)](#) ,
[Component.GetComponentsInChildren<T>\(bool, List<T>\)](#) ,
Component.GetComponentsInChildren<T>() , [Component.GetComponentsInChildren<T>\(List<T>\)](#) ,
[Component.GetComponentInParent\(Type, bool\)](#) , [Component.GetComponentInParent\(Type\)](#) ,
[Component.GetComponentInParent<T>\(bool\)](#) , Component.GetComponentInParent<T>() ,
[Component.GetComponentsInParent\(Type, bool\)](#) , [Component.GetComponentsInParent\(Type\)](#) ,
[Component.GetComponentsInParent<T>\(bool\)](#) ,
[Component.GetComponentsInParent<T>\(bool, List<T>\)](#) , Component.GetComponentsInParent<T>() ,
[Component.GetComponents\(Type\)](#) , [Component.GetComponents\(Type, List<Component>\)](#) ,
[Component.GetComponents<T>\(List<T>\)](#) , Component.GetComponents<T>() ,
Component.GetComponentIndex() , [Component.CompareTag\(string\)](#) ,
[Component.SendMessageUpwards\(string, object, SendMessageOptions\)](#) ,
[Component.SendMessageUpwards\(string, object\)](#) , [Component.SendMessageUpwards\(string\)](#) ,
[Component.SendMessageUpwards\(string, SendMessageOptions\)](#) ,

[Component.SendMessage\(string, object\)](#) , [Component.SendMessage\(string\)](#) ,
[Component.SendMessage\(string, object, SendMessageOptions\)](#) ,
[Component.SendMessage\(string, SendMessageOptions\)](#) ,
[Component.BroadcastMessage\(string, object, SendMessageOptions\)](#) ,
[Component.BroadcastMessage\(string, object\)](#) , [Component.BroadcastMessage\(string\)](#) ,
[Component.BroadcastMessage\(string, SendMessageOptions\)](#) , Component.transform ,
Component.gameObject , Component.tag , Object.GetInstanceID() , Object.GetHashCode() ,
[Object.Equals\(object\)](#) , Object.InstantiateAsync<T>(T) , Object.InstantiateAsync<T>(T, Transform) ,
Object.InstantiateAsync<T>(T, Vector3, Quaternion) ,
Object.InstantiateAsync<T>(T, Transform, Vector3, Quaternion) , [Object.InstantiateAsync<T>\(T, int\)](#) ,
[Object.InstantiateAsync<T>\(T, int, Transform\)](#) ,
[Object.InstantiateAsync<T>\(T, int, Vector3, Quaternion\)](#) ,
[Object.InstantiateAsync<T>\(T, int, ReadOnlySpan<Vector3>, ReadOnlySpan<Quaternion>\)](#) ,
[Object.InstantiateAsync<T>\(T, int, Transform, Vector3, Quaternion\)](#) ,
[Object.InstantiateAsync<T>\(T, int, Transform, ReadOnlySpan<Vector3>, ReadOnlySpan<Quaternion>\)](#) ,
Object.Instantiate(Object, Vector3, Quaternion) ,
Object.Instantiate(Object, Vector3, Quaternion, Transform) , Object.Instantiate(Object) ,
Object.Instantiate(Object, Scene) , Object.Instantiate(Object, Transform) ,
[Object.Instantiate\(Object, Transform, bool\)](#) , Object.Instantiate<T>(T) ,
Object.Instantiate<T>(T, Vector3, Quaternion) ,
Object.Instantiate<T>(T, Vector3, Quaternion, Transform) , Object.Instantiate<T>(T, Transform) ,
[Object.Instantiate<T>\(T, Transform, bool\)](#) , [Object.Destroy\(Object, float\)](#) , Object.Destroy(Object) ,
[Object.DestroyImmediate\(Object, bool\)](#) , Object.DestroyImmediate(Object) ,
[Object.FindObjectsOfType\(Type\)](#) , [Object.FindObjectsOfType\(Type, bool\)](#) ,
[Object.FindObjectsByType\(Type, FindObjectsSortMode\)](#) ,
[Object.FindObjectsByType\(Type, FindObjectsInactive, FindObjectsSortMode\)](#) ,
Object.DontDestroyOnLoad(Object) , [Object.DestroyObject\(Object, float\)](#) ,
Object.DestroyObject(Object) , [Object.FindSceneObjectsOfType\(Type\)](#) ,
[Object.FindObjectsOfTypeIncludingAssets\(Type\)](#) , Object.FindObjectsOfType<T>() ,
Object.FindObjectsByType<T>(FindObjectsSortMode) , [Object.FindObjectsOfType<T>\(bool\)](#) ,
Object.FindObjectsByType<T>(FindObjectsInactive, FindObjectsSortMode) ,
ObjectFindObjectOfType<T>() , [Object.FindObjectType<T>\(bool\)](#) ,
Object.FindFirstObjectByType<T>() , Object.FindAnyObjectByType<T>() ,
Object.FindFirstObjectByType<T>(FindObjectsInactive) ,
Object.FindAnyObjectByType<T>(FindObjectsInactive) , [Object.FindObjectsOfTypeAll\(Type\)](#) ,
[Object.FindObjectType\(Type\)](#) , [Object.FindFirstObjectByType\(Type\)](#) ,
[Object.FindAnyObjectByType\(Type\)](#) , [Object.FindObjectOfType\(Type, bool\)](#) ,
[Object.FindFirstObjectByType\(Type, FindObjectsInactive\)](#) ,
[Object.FindAnyObjectByType\(Type, FindObjectsInactive\)](#) , Object.ToString() , Object.name ,

`Object.hideFlags` , [object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Fields

FromWeight

```
[Tooltip("Start BlendShape weight")]
public float FromWeight
```

Field Value

[float](#)

Offset

```
[Range(0, 1)]
[Tooltip("Phase shift as a percentage (0 to 1.0)")]
public float Offset
```

Field Value

[float](#)

Phase

```
[Tooltip("Time in seconds to complete a single loop of the animation")]
public float Phase
```

Field Value

[float](#)

ToWeight

```
[Tooltip("Target BlendShape weight")]
public float ToWeight
```

Field Value

[float](#) ↗

Class AnimatePositionAuthoring

Namespace: [Unity.Entities.Animation](#)

Assembly: DOTSDynamicBone.dll

```
public class AnimatePositionAuthoring : MonoBehaviour
```

Inheritance

[object](#) ← Object ← Component ← Behaviour ← MonoBehaviour ← AnimatePositionAuthoring

Inherited Members

MonoBehaviour.IsInvoking() , MonoBehaviour.CancelInvoke() , [MonoBehaviour.Invoke\(string, float\)](#) ,
[MonoBehaviour.InvokeRepeating\(string, float, float\)](#) , [MonoBehaviour.CancelInvoke\(string\)](#) ,
[MonoBehaviour.IsInvoking\(string\)](#) , [MonoBehaviour.StartCoroutine\(string\)](#) ,
[MonoBehaviour.StartCoroutine\(string, object\)](#) , [MonoBehaviour.StartCoroutine\(IEnumerator\)](#) ,
[MonoBehaviour.StartCoroutine_Auto\(IEnumerator\)](#) , [MonoBehaviour.StopCoroutine\(IEnumerator\)](#) ,
MonoBehaviour.StopCoroutine(Coroutine) , [MonoBehaviour.StopCoroutine\(string\)](#) ,
MonoBehaviour.StopAllCoroutines() , [MonoBehaviour.print\(object\)](#) ,
MonoBehaviour.destroyCancellationToken , MonoBehaviour.useGUILayout ,
MonoBehaviour.runInEditMode , Behaviour.enabled , Behaviour.isActiveAndEnabled ,
[Component.GetComponent\(Type\)](#) , Component.GetComponent<T>() ,
[Component.TryGetComponent\(Type, out Component\)](#) , Component.TryGetComponent<T>(out T) ,
[Component.GetComponent\(string\)](#) , [Component.GetComponentInChildren\(Type, bool\)](#) ,
[Component.GetComponentInChildren\(Type\)](#) , [Component.GetComponentInChildren<T>\(bool\)](#) ,
Component.GetComponentInChildren<T>() , [Component.GetComponentsInChildren\(Type, bool\)](#) ,
[Component.GetComponentsInChildren\(Type\)](#) , [Component.GetComponentsInChildren<T>\(bool\)](#) ,
[Component.GetComponentsInChildren<T>\(bool, List<T>\)](#) ,
Component.GetComponentsInChildren<T>() , [Component.GetComponentsInChildren<T>\(List<T>\)](#) ,
[Component.GetComponentInParent\(Type, bool\)](#) , [Component.GetComponentInParent\(Type\)](#) ,
[Component.GetComponentInParent<T>\(bool\)](#) , Component.GetComponentInParent<T>() ,
[Component.GetComponentsInParent\(Type, bool\)](#) , [Component.GetComponentsInParent\(Type\)](#) ,
[Component.GetComponentsInParent<T>\(bool\)](#) ,
[Component.GetComponentsInParent<T>\(bool, List<T>\)](#) , Component.GetComponentsInParent<T>() ,
[Component.GetComponents\(Type\)](#) , [Component.GetComponents\(Type, List<Component>\)](#) ,
[Component.GetComponents<T>\(List<T>\)](#) , Component.GetComponents<T>() ,
Component.GetComponentIndex() , [Component.CompareTag\(string\)](#) ,
[Component.SendMessageUpwards\(string, object, SendMessageOptions\)](#) ,
[Component.SendMessageUpwards\(string, object\)](#) , [Component.SendMessageUpwards\(string\)](#) ,
[Component.SendMessageUpwards\(string, SendMessageOptions\)](#) ,

[Component.SendMessage\(string, object\)](#) , [Component.SendMessage\(string\)](#) ,
[Component.SendMessage\(string, object, SendMessageOptions\)](#) ,
[Component.SendMessage\(string, SendMessageOptions\)](#) ,
[Component.BroadcastMessage\(string, object, SendMessageOptions\)](#) ,
[Component.BroadcastMessage\(string, object\)](#) , [Component.BroadcastMessage\(string\)](#) ,
[Component.BroadcastMessage\(string, SendMessageOptions\)](#) , Component.transform ,
Component.gameObject , Component.tag , Object.GetInstanceID() , Object.GetHashCode() ,
[Object.Equals\(object\)](#) , Object.InstantiateAsync<T>(T) , Object.InstantiateAsync<T>(T, Transform) ,
Object.InstantiateAsync<T>(T, Vector3, Quaternion) ,
Object.InstantiateAsync<T>(T, Transform, Vector3, Quaternion) , [Object.InstantiateAsync<T>\(T, int\)](#) ,
[Object.InstantiateAsync<T>\(T, int, Transform\)](#) ,
[Object.InstantiateAsync<T>\(T, int, Vector3, Quaternion\)](#) ,
[Object.InstantiateAsync<T>\(T, int, ReadOnlySpan<Vector3>, ReadOnlySpan<Quaternion>\)](#) ,
[Object.InstantiateAsync<T>\(T, int, Transform, Vector3, Quaternion\)](#) ,
[Object.InstantiateAsync<T>\(T, int, Transform, ReadOnlySpan<Vector3>, ReadOnlySpan<Quaternion>\)](#) ,
Object.Instantiate(Object, Vector3, Quaternion) ,
Object.Instantiate(Object, Vector3, Quaternion, Transform) , Object.Instantiate(Object) ,
Object.Instantiate(Object, Scene) , Object.Instantiate(Object, Transform) ,
[Object.Instantiate\(Object, Transform, bool\)](#) , Object.Instantiate<T>(T) ,
Object.Instantiate<T>(T, Vector3, Quaternion) ,
Object.Instantiate<T>(T, Vector3, Quaternion, Transform) , Object.Instantiate<T>(T, Transform) ,
[Object.Instantiate<T>\(T, Transform, bool\)](#) , [Object.Destroy\(Object, float\)](#) , Object.Destroy(Object) ,
[Object.DestroyImmediate\(Object, bool\)](#) , Object.DestroyImmediate(Object) ,
[Object.FindObjectsOfType\(Type\)](#) , [Object.FindObjectsOfType\(Type, bool\)](#) ,
[Object.FindObjectsByType\(Type, FindObjectsSortMode\)](#) ,
[Object.FindObjectsByType\(Type, FindObjectsInactive, FindObjectsSortMode\)](#) ,
Object.DontDestroyOnLoad(Object) , [Object.DestroyObject\(Object, float\)](#) ,
Object.DestroyObject(Object) , [Object.FindSceneObjectsOfType\(Type\)](#) ,
[Object.FindObjectsOfTypeIncludingAssets\(Type\)](#) , Object.FindObjectsOfType<T>() ,
Object.FindObjectsByType<T>(FindObjectsSortMode) , [Object.FindObjectsOfType<T>\(bool\)](#) ,
Object.FindObjectsByType<T>(FindObjectsInactive, FindObjectsSortMode) ,
ObjectFindObjectOfType<T>() , [Object.FindObjectType<T>\(bool\)](#) ,
Object.FindFirstObjectByType<T>() , Object.FindAnyObjectByType<T>() ,
Object.FindFirstObjectByType<T>(FindObjectsInactive) ,
Object.FindAnyObjectByType<T>(FindObjectsInactive) , [Object.FindObjectsOfTypeAll\(Type\)](#) ,
[Object.FindObjectType\(Type\)](#) , [Object.FindFirstObjectByType\(Type\)](#) ,
[Object.FindAnyObjectByType\(Type\)](#) , [Object.FindObjectOfType\(Type, bool\)](#) ,
[Object.FindFirstObjectByType\(Type, FindObjectsInactive\)](#) ,
[Object.FindAnyObjectByType\(Type, FindObjectsInactive\)](#) , Object.ToString() , Object.name ,

`Object.hideFlags` , [object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Fields

FromPosition

```
[Tooltip("Local start position")]
public Vector3 FromPosition
```

Field Value

`Vector3`

Offset

```
[Range(0, 1)]
[Tooltip("Phase shift as a percentage (0 to 1.0)")]
public float Offset
```

Field Value

`float`

Phase

```
[Tooltip("Time in seconds to complete a single loop of the animation")]
public float Phase
```

Field Value

`float`

ToPosition

```
[Tooltip("Local target position")]
public Vector3 ToPosition
```

Field Value

Vector3

appendToCurrentTransform

```
[Tooltip("enable to make FromPosition and ToPosition relative to the transform's current
world space coordinates.")]
public bool appendToCurrentTransform
```

Field Value

bool ↗

initialPosition

```
[Tooltip("Tracked current position of game object transform")]
public Vector3 initialPosition
```

Field Value

Vector3

Methods

OnEnable()

```
public void OnEnable()
```

Update()

```
public void Update()
```

Class AnimateScaleAuthoring

Namespace: [Unity.Entities.Animation](#)

Assembly: DOTSDynamicBone.dll

```
public class AnimateScaleAuthoring : MonoBehaviour
```

Inheritance

[Object](#) ← Object ← Component ← Behaviour ← MonoBehaviour ← AnimateScaleAuthoring

Inherited Members

MonoBehaviour.IsInvoking() , MonoBehaviour.CancelInvoke() , [MonoBehaviour.Invoke\(string, float\)](#) ,
[MonoBehaviour.InvokeRepeating\(string, float, float\)](#) , [MonoBehaviour.CancelInvoke\(string\)](#) ,
[MonoBehaviour.IsInvoking\(string\)](#) , [MonoBehaviour.StartCoroutine\(string\)](#) ,
[MonoBehaviour.StartCoroutine\(string, object\)](#) , [MonoBehaviour.StartCoroutine\(IEnumerator\)](#) ,
[MonoBehaviour.StartCoroutine_Auto\(IEnumerator\)](#) , [MonoBehaviour.StopCoroutine\(IEnumerator\)](#) ,
MonoBehaviour.StopCoroutine(Coroutine) , [MonoBehaviour.StopCoroutine\(string\)](#) ,
MonoBehaviour.StopAllCoroutines() , [MonoBehaviour.print\(object\)](#) ,
MonoBehaviour.destroyCancellationToken , MonoBehaviour.useGUILayout ,
MonoBehaviour.runInEditMode , Behaviour.enabled , Behaviour.isActiveAndEnabled ,
[Component.GetComponent\(Type\)](#) , Component.GetComponent<T>() ,
[Component.TryGetComponent\(Type, out Component\)](#) , Component.TryGetComponent<T>(out T) ,
[Component.GetComponent\(string\)](#) , [Component.GetComponentInChildren\(Type, bool\)](#) ,
[Component.GetComponentInChildren\(Type\)](#) , [Component.GetComponentInChildren<T>\(bool\)](#) ,
Component.GetComponentInChildren<T>() , [Component.GetComponentsInChildren\(Type, bool\)](#) ,
[Component.GetComponentsInChildren\(Type\)](#) , [Component.GetComponentsInChildren<T>\(bool\)](#) ,
[Component.GetComponentsInChildren<T>\(bool, List<T>\)](#) ,
Component.GetComponentsInChildren<T>() , [Component.GetComponentsInChildren<T>\(List<T>\)](#) ,
[Component.GetComponentInParent\(Type, bool\)](#) , [Component.GetComponentInParent\(Type\)](#) ,
[Component.GetComponentInParent<T>\(bool\)](#) , Component.GetComponentInParent<T>() ,
[Component.GetComponentsInParent\(Type, bool\)](#) , [Component.GetComponentsInParent\(Type\)](#) ,
[Component.GetComponentsInParent<T>\(bool\)](#) ,
[Component.GetComponentsInParent<T>\(bool, List<T>\)](#) , Component.GetComponentsInParent<T>() ,
[Component.GetComponents\(Type\)](#) , [Component.GetComponents\(Type, List<Component>\)](#) ,
[Component.GetComponents<T>\(List<T>\)](#) , Component.GetComponents<T>() ,
Component.GetComponentIndex() , [Component.CompareTag\(string\)](#) ,
[Component.SendMessageUpwards\(string, object, SendMessageOptions\)](#) ,
[Component.SendMessageUpwards\(string, object\)](#) , [Component.SendMessageUpwards\(string\)](#) ,
[Component.SendMessageUpwards\(string, SendMessageOptions\)](#) ,

[Component.SendMessage\(string, object\)](#) , [Component.SendMessage\(string\)](#) ,
[Component.SendMessage\(string, object, SendMessageOptions\)](#) ,
[Component.SendMessage\(string, SendMessageOptions\)](#) ,
[Component.BroadcastMessage\(string, object, SendMessageOptions\)](#) ,
[Component.BroadcastMessage\(string, object\)](#) , [Component.BroadcastMessage\(string\)](#) ,
[Component.BroadcastMessage\(string, SendMessageOptions\)](#) , Component.transform ,
Component.gameObject , Component.tag , Object.GetInstanceID() , Object.GetHashCode() ,
[Object.Equals\(object\)](#) , Object.InstantiateAsync<T>(T) , Object.InstantiateAsync<T>(T, Transform) ,
Object.InstantiateAsync<T>(T, Vector3, Quaternion) ,
Object.InstantiateAsync<T>(T, Transform, Vector3, Quaternion) , [Object.InstantiateAsync<T>\(T, int\)](#) ,
[Object.InstantiateAsync<T>\(T, int, Transform\)](#) ,
[Object.InstantiateAsync<T>\(T, int, Vector3, Quaternion\)](#) ,
[Object.InstantiateAsync<T>\(T, int, ReadOnlySpan<Vector3>, ReadOnlySpan<Quaternion>\)](#) ,
[Object.InstantiateAsync<T>\(T, int, Transform, Vector3, Quaternion\)](#) ,
[Object.InstantiateAsync<T>\(T, int, Transform, ReadOnlySpan<Vector3>, ReadOnlySpan<Quaternion>\)](#) ,
Object.Instantiate(Object, Vector3, Quaternion) ,
Object.Instantiate(Object, Vector3, Quaternion, Transform) , Object.Instantiate(Object) ,
Object.Instantiate(Object, Scene) , Object.Instantiate(Object, Transform) ,
[Object.Instantiate\(Object, Transform, bool\)](#) , Object.Instantiate<T>(T) ,
Object.Instantiate<T>(T, Vector3, Quaternion) ,
Object.Instantiate<T>(T, Vector3, Quaternion, Transform) , Object.Instantiate<T>(T, Transform) ,
[Object.Instantiate<T>\(T, Transform, bool\)](#) , [Object.Destroy\(Object, float\)](#) , Object.Destroy(Object) ,
[Object.DestroyImmediate\(Object, bool\)](#) , Object.DestroyImmediate(Object) ,
[Object.FindObjectsOfType\(Type\)](#) , [Object.FindObjectsOfType\(Type, bool\)](#) ,
[Object.FindObjectsByType\(Type, FindObjectsSortMode\)](#) ,
[Object.FindObjectsByType\(Type, FindObjectsInactive, FindObjectsSortMode\)](#) ,
Object.DontDestroyOnLoad(Object) , [Object.DestroyObject\(Object, float\)](#) ,
Object.DestroyObject(Object) , [Object.FindSceneObjectsOfType\(Type\)](#) ,
[Object.FindObjectsOfTypeIncludingAssets\(Type\)](#) , Object.FindObjectsOfType<T>() ,
Object.FindObjectsByType<T>(FindObjectsSortMode) , [Object.FindObjectsOfType<T>\(bool\)](#) ,
Object.FindObjectsByType<T>(FindObjectsInactive, FindObjectsSortMode) ,
ObjectFindObjectOfType<T>() , [Object.FindObjectType<T>\(bool\)](#) ,
Object.FindFirstObjectByType<T>() , Object.FindAnyObjectByType<T>() ,
Object.FindFirstObjectByType<T>(FindObjectsInactive) ,
Object.FindAnyObjectByType<T>(FindObjectsInactive) , [Object.FindObjectsOfTypeAll\(Type\)](#) ,
[Object.FindObjectType\(Type\)](#) , [Object.FindFirstObjectByType\(Type\)](#) ,
[Object.FindAnyObjectByType\(Type\)](#) , [Object.FindObjectOfType\(Type, bool\)](#) ,
[Object.FindFirstObjectByType\(Type, FindObjectsInactive\)](#) ,
[Object.FindAnyObjectByType\(Type, FindObjectsInactive\)](#) , Object.ToString() , Object.name ,

`Object.hideFlags` , [object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Fields

FromScale

```
[Tooltip("Local start scale")]
public Vector3 FromScale
```

Field Value

Vector3

Offset

```
[Range(0, 1)]
[Tooltip("Phase shift as a percentage (0 to 1.0)")]
public float Offset
```

Field Value

[float](#)

Phase

```
[Tooltip("Time in seconds to complete a single loop of the animation")]
public float Phase
```

Field Value

[float](#)

ToScale

```
[Tooltip("Local target scale")]
public Vector3 ToScale
```

Field Value

Vector3

Class ComputeSkinMatricesBakingSystem

Namespace: [Unity.Entities.Animation](#)

Assembly: DOTSDynamicBone.dll

```
[WorldSystemFilter(WorldSystemFilterFlags.BakingSystem, WorldSystemFilterFlags.Default)]
public class ComputeSkinMatricesBakingSystem : SystemBase
```

Inheritance

[object](#) ← ComponentSystemBase ← SystemBase ← ComputeSkinMatricesBakingSystem

Inherited Members

SystemBase.Dependency , SystemBase.CheckedStateRef , SystemBase.CompleteDependency() ,
SystemBase.Entities , SystemBase.Job , SystemBase.Update() , SystemBase.GetComponent<T>(Entity) ,
SystemBase.SetComponent<T>(Entity, T) , SystemBase.HasComponent<T>(Entity) ,
SystemBase.HasBuffer<T>(Entity) , [SystemBase.GetComponentLookup<T>\(bool\)](#) ,
[SystemBase.GetComponentDataFromEntity<T>\(bool\)](#) , [SystemBase.GetBuffer<T>\(Entity, bool\)](#) ,
[SystemBase.GetBufferLookup<T>\(bool\)](#) , [SystemBase.GetBufferFromEntity<T>\(bool\)](#) ,
SystemBase.GetEntityStorageInfoLookup() , SystemBase.GetStorageInfoFromEntity() ,
SystemBase.Exists(Entity) , ComponentSystemBase.Enabled , ComponentSystemBase.EntityQueries ,
ComponentSystemBase.GlobalSystemVersion , ComponentSystemBase.LastSystemVersion ,
ComponentSystemBase.EntityManager , ComponentSystemBase.World ,
ComponentSystemBase.SystemHandle , ComponentSystemBase.SystemHandleUntyped ,
ComponentSystemBase.Time , ComponentSystemBase.WorldUpdateAllocator ,
ComponentSystemBase.OnCreate() , ComponentSystemBase.OnStartRunning() ,
ComponentSystemBase.OnStopRunning() , ComponentSystemBase.OnDestroy() ,
ComponentSystemBase.ShouldRunSystem() ,
[ComponentSystemBase.GetComponentTypeHandle<T>\(bool\)](#) ,
ComponentSystemBase.GetDynamicComponentTypeHandle(ComponentType) ,
[ComponentSystemBase.GetBufferTypeHandle<T>\(bool\)](#) ,
ComponentSystemBase.GetSharedComponentTypeHandle<T>() ,
ComponentSystemBase.GetDynamicSharedComponentTypeHandle(ComponentType) ,
ComponentSystemBase.GetEntityTypeHandle() , ComponentSystemBase.RequireForUpdate(EntityQuery) ,
ComponentSystemBase.RequireAnyForUpdate(params EntityQuery[]) ,
ComponentSystemBase.RequireAnyForUpdate(NativeArray<EntityQuery>) ,
ComponentSystemBase.RequireForUpdate<T>() ,
ComponentSystemBase.RequireSingletonForUpdate<T>() , ComponentSystemBase.HasSingleton<T>() ,
ComponentSystemBase.GetSingleton<T>() , ComponentSystemBase.GetSingletonRW<T>() ,
[ComponentSystemBase.GetSingletonBuffer<T>\(bool\)](#) ,

```
ComponentSystemBase.TryGetSingleton<T>(out T) ,  
ComponentSystemBase.TryGetSingletonBuffer<T>(out DynamicBuffer<T>) ,  
ComponentSystemBase.SetSingleton<T>(T) , ComponentSystemBase.GetSingletonEntity<T>() ,  
ComponentSystemBase.TryGetSingletonEntity<T>(out Entity) ,  
ComponentSystemBase.GetEntityQuery(params ComponentType[]) ,  
ComponentSystemBase.GetEntityQuery(NativeArray<ComponentType>) ,  
ComponentSystemBase.GetEntityQuery(params EntityQueryDesc[]) ,  
ComponentSystemBase.GetEntityQuery(in EntityQueryBuilder) , object.Equals\(object\) ,  
object.Equals\(object, object\) , object.GetHashCode\(\) , object.GetType\(\) ,  
object.MemberwiseClone\(\) , object.ReferenceEquals\(object, object\) , object.ToString\(\)
```

Methods

OnCreateForCompiler()

```
protected override void OnCreateForCompiler()
```

OnUpdate()

Implement [OnUpdate\(\)](#) to perform the major work of this system.

```
protected override void OnUpdate()
```

Remarks

By default, the system invokes `OnUpdate()` once every frame on the main thread. To skip OnUpdate if all of the system's [EntityQueries] are empty, use the [RequireMatchingQueriesForUpdateAttribute]. To limit when OnUpdate is invoked, you can specify components that must exist, or queries that match specific Entities. To do this, call `Unity.Entities.ComponentSystemBase.RequireForUpdate<T>()` or `Unity.Entities.ComponentSystemBase.RequireForUpdate(Unity.Entities.EntityQuery)` in the system's OnCreate method. For more information, see [ShouldRunSystem].

The [Entities.ForEach] and [Job.WithCode] constructions provide convenient mechanisms for defining jobs. You can also instantiate and schedule an `Unity.Entities.IJobChunk` instance; you can use the [C# Job System] or you can perform work on the main thread. If you call `Unity.Entities.EntityManager` methods that perform structural changes on the main thread, be sure to arrange the system order to minimize the performance impact of the resulting [sync points].

Namespace UnityNexus.Common

Classes

[BitmaskUtils](#)

[EnumFlagsAttribute](#)

[InGameEntitiesList](#)

Structs

[EditorEntity](#)

[EntityGameObjectLink_Manged](#)

[EntityObjectLink_ComponentData](#)

[EntityObjectLink_Manged](#)

Class BitmaskUtils

Namespace: [UnityNexus.Common](#)

Assembly: EggExtensions.dll

```
public static class BitmaskUtils
```

Inheritance

[object](#) ← BitmaskUtils

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

Methods

IsBitSet(byte, byte)

```
public static bool IsBitSet(byte mask, byte index)
```

Parameters

mask [byte](#)

index [byte](#)

Returns

[bool](#)

IsBitSet(uint, byte)

```
public static bool IsBitSet(uint mask, byte index)
```

Parameters

`mask` [uint](#)

`index` [byte](#)

Returns

[bool](#)

IsBitSet(ulong, byte)

```
public static bool IsBitSet(ulong mask, byte index)
```

Parameters

`mask` [ulong](#)

`index` [byte](#)

Returns

[bool](#)

SetBit(byte, byte)

```
public static byte SetBit(byte mask, byte index)
```

Parameters

`mask` [byte](#)

`index` [byte](#)

Returns

[byte](#)

SetBit(uint, byte)

```
public static uint SetBit(uint mask, byte index)
```

Parameters

mask [uint](#)

index [byte](#)

Returns

[uint](#)

SetBit(ulong, byte)

```
public static ulong SetBit(ulong mask, byte index)
```

Parameters

mask [ulong](#)

index [byte](#)

Returns

[ulong](#)

SetBits(byte, byte)

```
public static byte SetBits(byte mask1, byte mask2)
```

Parameters

mask1 [byte](#)

mask2 [byte](#)

Returns

SetBits(uint, uint)

```
public static uint SetBits(uint mask1, uint mask2)
```

Parameters

mask1 [uint](#)

mask2 [uint](#)

Returns

[uint](#)

SetBits(ulong, uint)

```
public static ulong SetBits(ulong mask1, uint mask2)
```

Parameters

mask1 [ulong](#)

mask2 [uint](#)

Returns

[ulong](#)

ToggleBit(byte, byte, bool)

```
public static byte ToggleBit(byte mask, byte index, bool enable)
```

Parameters

mask [byte](#)

index [byte](#)

enable [bool](#)

Returns

[byte](#)

ToggleBit(uint, byte, bool)

```
public static uint ToggleBit(uint mask, byte index, bool enable)
```

Parameters

mask [uint](#)

index [byte](#)

enable [bool](#)

Returns

[uint](#)

ToggleBit(ulong, byte, bool)

```
public static ulong ToggleBit(ulong mask, byte index, bool enable)
```

Parameters

mask [ulong](#)

index [byte](#)

enable [bool](#)

Returns

[ulong](#)

ToggleBits(byte, byte, bool)

```
public static byte ToggleBits(byte mask1, byte mask2, bool enable)
```

Parameters

mask1 [byte](#)

mask2 [byte](#)

enable [bool](#)

Returns

[byte](#)

ToggleBits(uint, uint, bool)

```
public static uint ToggleBits(uint mask1, uint mask2, bool enable)
```

Parameters

mask1 [uint](#)

mask2 [uint](#)

enable [bool](#)

Returns

[uint](#)

ToggleBits(ulong, uint, bool)

```
public static ulong ToggleBits(ulong mask1, uint mask2, bool enable)
```

Parameters

mask1 [ulong](#)

mask2 [uint](#)

enable [bool](#)

Returns

[ulong](#)

UnsetBit(byte, byte)

```
public static byte UnsetBit(byte mask, byte index)
```

Parameters

mask [byte](#)

index [byte](#)

Returns

[byte](#)

UnsetBit(uint, byte)

```
public static uint UnsetBit(uint mask, byte index)
```

Parameters

mask [uint](#)

index [byte](#)

Returns

[uint](#)

UnsetBit(ulong, byte)

```
public static ulong UnsetBit(ulong mask, byte index)
```

Parameters

mask [ulong](#)

index [byte](#)

Returns

[ulong](#)

UnsetBits(byte, byte)

```
public static byte UnsetBits(byte mask1, byte mask2)
```

Parameters

mask1 [byte](#)

mask2 [byte](#)

Returns

[byte](#)

UnsetBits(uint, uint)

```
public static uint UnsetBits(uint mask1, uint mask2)
```

Parameters

`mask1 uint`

`mask2 uint`

Returns

`uint`

UnsetBits(ulong, uint)

```
public static ulong UnsetBits(ulong mask1, uint mask2)
```

Parameters

`mask1 ulong`

`mask2 uint`

Returns

`ulong`

Struct EditorEntity

Namespace: [UnityNexus.Common](#)

Assembly: EggExtensions.dll

```
[Serializable]
public struct EditorEntity : IComponentData, IQueryTypeParameter
```

Implements

IComponentData, IQueryTypeParameter

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Constructors

EditorEntity(Entity)

```
public EditorEntity(Entity e)
```

Parameters

e Entity

Fields

Index

```
public int Index
```

Field Value

[int](#)

Version

```
public int Version
```

Field Value

[int](#)

e

```
public Entity e
```

Field Value

Entity

Properties

Null

```
public static EditorEntity Null { get; }
```

Property Value

[EditorEntity](#)

Methods

Equals(Entity)

```
public bool Equals(Entity other)
```

Parameters

other Entity

Returns

[bool](#)

Equals(EditorEntity)

```
public bool Equals(EditorEntity other)
```

Parameters

other [EditorEntity](#)

Returns

[bool](#)

Struct EntityGameObjectLinke_Managed

Namespace: [UnityNexus.Common](#)

Assembly: EggExtensions.dll

```
[Serializable]
public struct EntityGameObjectLinke_Managed
```

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Fields

entity

```
public Entity entity
```

Field Value

Entity

gameObject

```
public GameObject gameObject
```

Field Value

GameObject

Struct EntityObjectLink_ComponentData

Namespace: [UnityNexus.Common](#)

Assembly: EggExtensions.dll

```
[Serializable]
public struct EntityObjectLink_ComponentData : IComponentData, IQueryTypeParameter
```

Implements

IComponentData, IQueryTypeParameter

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Fields

ReferenceGameObjectArrayIndex

```
public int ReferenceGameObjectArrayIndex
```

Field Value

[int](#)

entity

```
public Entity entity
```

Field Value

Entity

Struct EntityObjectLink_Managed

Namespace: [UnityNexus.Common](#)

Assembly: EggExtensions.dll

```
[Serializable]
public struct EntityObjectLink_Managed
```

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Fields

entity

```
public Entity entity
```

Field Value

Entity

obj

```
public Object obj
```

Field Value

Object

Class EnumFlagsAttribute

Namespace: [UnityNexus.Common](#)

Assembly: EggExtensions.dll

```
public sealed class EnumFlagsAttribute : PropertyAttribute, _Attribute
```

Inheritance

[object](#) ← [Attribute](#) ← [PropertyAttribute](#) ← [EnumFlagsAttribute](#)

Implements

[Attribute](#)

Inherited Members

[PropertyAttribute.order](#) , [Attribute.Equals\(object\)](#) , [Attribute.GetCustomAttribute\(Assembly, Type\)](#) ,
[Attribute.GetCustomAttribute\(Assembly, Type, bool\)](#) ,
[Attribute.GetCustomAttribute\(MemberInfo, Type\)](#) ,
[Attribute.GetCustomAttribute\(MemberInfo, Type, bool\)](#) ,
[Attribute.GetCustomAttribute\(Module, Type\)](#) , [Attribute.GetCustomAttribute\(Module, Type, bool\)](#) ,
[Attribute.GetCustomAttribute\(ParameterInfo, Type\)](#) ,
[Attribute.GetCustomAttribute\(ParameterInfo, Type, bool\)](#) , [Attribute.GetCustomAttributes\(Assembly\)](#) ,
[Attribute.GetCustomAttributes\(Assembly, bool\)](#) , [Attribute.GetCustomAttributes\(Assembly, Type\)](#) ,
[Attribute.GetCustomAttributes\(Assembly, Type, bool\)](#) , [Attribute.GetCustomAttributes\(MemberInfo\)](#) ,
[Attribute.GetCustomAttributes\(MemberInfo, bool\)](#) ,
[Attribute.GetCustomAttributes\(MemberInfo, Type\)](#) ,
[Attribute.GetCustomAttributes\(MemberInfo, Type, bool\)](#) , [Attribute.GetCustomAttributes\(Module\)](#) ,
[Attribute.GetCustomAttributes\(Module, bool\)](#) , [Attribute.GetCustomAttributes\(Module, Type\)](#) ,
[Attribute.GetCustomAttributes\(Module, Type, bool\)](#) , [Attribute.GetCustomAttributes\(ParameterInfo\)](#) ,
[Attribute.GetCustomAttributes\(ParameterInfo, bool\)](#) ,
[Attribute.GetCustomAttributes\(ParameterInfo, Type\)](#) ,
[Attribute.GetCustomAttributes\(ParameterInfo, Type, bool\)](#) , [Attribute.GetHashCode\(\)](#) ,
[Attribute.IsDefaultAttribute\(\)](#) , [Attribute.IsDefined\(Assembly, Type\)](#) ,
[Attribute.IsDefined\(Assembly, Type, bool\)](#) , [Attribute.IsDefined\(MemberInfo, Type\)](#) ,
[Attribute.IsDefined\(MemberInfo, Type, bool\)](#) , [Attribute.IsDefined\(Module, Type\)](#) ,
[Attribute.IsDefined\(Module, Type, bool\)](#) , [Attribute.IsDefined\(ParameterInfo, Type\)](#) ,
[Attribute.IsDefined\(ParameterInfo, Type, bool\)](#) , [Attribute.Match\(object\)](#) , [Attribute.TypeId](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,
[object.ToString\(\)](#)

Constructors

EnumFlagsAttribute()

```
public EnumFlagsAttribute()
```

Methods

EnumToBitmask(List<int>)

```
public static uint EnumToBitmask(List<int> indices)
```

Parameters

indices [List<int>](#)

Returns

[uint](#)

GetSelectedIndexes<T>(T)

```
public static List<int> GetSelectedIndexes<T>(T val) where T : IConvertible
```

Parameters

val T

Returns

[List<int>](#)

Type Parameters

T

GetSelectedStrings<T>(T)

```
public static List<string> GetSelectedStrings<T>(T val) where T : IConvertible
```

Parameters

val T

Returns

[List](#)<[string](#)>

Type Parameters

T

Class InGameEntitiesList

Namespace: [UnityNexus.Common](#)

Assembly: EggExtensions.dll

```
[Serializable]
public class InGameEntitiesList
```

Inheritance

[object](#) ← InGameEntitiesList

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

Fields

entities

```
public List<Entity> entities
```

Field Value

[List](#)<Entity>

Methods

SetEntityManager(EntityManager)

```
public void SetEntityManager(EntityManager em)
```

Parameters

em EntityManager

SetEntityQuery(EntityQuery)

```
public void SetEntityQuery(EntityQuery query)
```

Parameters

query EntityQuery

Update()

```
public void Update()
```

Namespace UnityNexus.Extensions

Classes

[DOTSAnimationCurveAuthoring](#)

[DOTSAnimationCurveComponent](#)

[MathematicsExtensions](#)

Structs

[DOTSAnimationCurve](#)

[DOTSAnimationCurveBlobArray](#)

[DOTSKeyframe](#)

Struct DOTSAutomationCurve

Namespace: [UnityNexus.Extensions](#)

Assembly: EggExtensions.dll

```
[BurstCompile]
public struct DOTSAutomationCurve : IComponentData, IQueryTypeParameter
```

Implements

IComponentData, IQueryTypeParameter

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Fields

value

```
public BlobAssetReference<DOTSAutomationCurveBlobArray> value
```

Field Value

BlobAssetReference<[DOTSAutomationCurveBlobArray](#)>

Methods

Dispose()

```
public void Dispose()
```

Evaluate(float)

```
public readonly float Evaluate(float time)
```

Parameters

time [float](#)

Returns

[float](#)

Class DOTSAutomationCurveAuthoring

Namespace: [UnityNexus.Extensions](#)

Assembly: EggExtensions.dll

```
[BurstCompile]
public class DOTSAutomationCurveAuthoring
```

Inheritance

[object](#) ← DOTSAutomationCurveAuthoring

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

Methods

CreateDOTSAutomationCurve(AnimationCurve, Allocator, Allocator)

```
public static DOTSAutomationCurve CreateDOTSAutomationCurve(AnimationCurve curve, Allocator  
builderAllocator, Allocator blobAllocator)
```

Parameters

curve AnimationCurve

builderAllocator Allocator

blobAllocator Allocator

Returns

[DOTSAutomationCurve](#)

CreateDOTSAutomationCurve(Keyframe[], Allocator, Allocator)

```
public static DOTSAnimationCurve CreateDOTSAnimationCurve(Keyframe[] keys, Allocator  
builderAllocator, Allocator blobAllocator)
```

Parameters

keys Keyframe[]

builderAllocator Allocator

blobAllocator Allocator

Returns

[DOTSAnimationCurve](#)

Set(ref DOTSAnimationCurve, AnimationCurve)

```
public static void Set(ref DOTSAnimationCurve curve, AnimationCurve otherCurve)
```

Parameters

curve [DOTSAnimationCurve](#)

otherCurve AnimationCurve

Set(ref DOTSAnimationCurve, Keyframe[])

```
public static void Set(ref DOTSAnimationCurve curve, Keyframe[] keys)
```

Parameters

curve [DOTSAnimationCurve](#)

keys Keyframe[]

Set(ref DOTSAnimationCurve, ref DOTSAnimationCurve)

```
[BurstCompile]
public static void Set(ref DOTSAnimationCurve curve, ref DOTSAnimationCurve otherCurve)
```

Parameters

curve [DOTSAnimationCurve](#)

otherCurve [DOTSAnimationCurve](#)

Struct DOTSAutomationCurveBlobArray

Namespace: [UnityNexus.Extensions](#)

Assembly: EggExtensions.dll

```
public struct DOTSAutomationCurveBlobArray
```

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Fields

keys

```
public BlobArray<DOTSKeyframe> keys
```

Field Value

BlobArray<[DOTSKeyframe](#)>

values

```
public BlobArray<float> values
```

Field Value

BlobArray<[float](#)>

Methods

Evaluate(float)

```
public float Evaluate(float time)
```

Parameters

time [float](#)

Returns

[float](#)

Class DOTSAutomationCurveComponent

Namespace: [UnityNexus.Extensions](#)

Assembly: EggExtensions.dll

```
public class DOTSAutomationCurveComponent
```

Inheritance

[object](#) ← DOTSAutomationCurveComponent

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

Fields

animationCurve

```
public AnimationCurve animationCurve
```

Field Value

AnimationCurve

Methods

ToDOTSAutomationCurve(Allocator, Allocator)

```
public DOTSAutomationCurve ToDOTSAutomationCurve(Allocator builderAllocator,  
Allocator blobAllocator)
```

Parameters

builderAllocator Allocator

blobAllocator Allocator

Returns

[DOTSAnimationCurve](#)

Struct DOTSKeyframe

Namespace: [UnityNexus.Extensions](#)

Assembly: EggExtensions.dll

```
[Serializable]
[BurstCompile]
public struct DOTSKeyframe : IComponentData, IQueryTypeParameter
```

Implements

IComponentData, IQueryTypeParameter

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

Constructors

DOTSKeyframe(float, float)

```
public DOTSKeyframe(float time, float value)
```

Parameters

time [float](#)

value [float](#)

DOTSKeyframe(float, float, float, float)

```
public DOTSKeyframe(float time, float value, float inTangent, float outTangent)
```

Parameters

time [float](#)

value [float](#)

inTangent [float](#)

outTangent [float](#)

DOTSKeyframe(float, float, float, float, float, float)

```
public DOTSKeyframe(float time, float value, float inTangent, float outTangent, float  
inWeight, float outWeight)
```

Parameters

time [float](#)

value [float](#)

inTangent [float](#)

outTangent [float](#)

inWeight [float](#)

outWeight [float](#)

DOTSKeyframe(Keyframe)

```
public DOTSKeyframe(Keyframe keyframe)
```

Parameters

keyframe Keyframe

Properties

inTangent

```
public float inTangent { get; set; }
```

Property Value

[float](#)

inWeight

```
public float inWeight { get; set; }
```

Property Value

[float](#)

outTangent

```
public float outTangent { get; set; }
```

Property Value

[float](#)

outWeight

```
public float outWeight { get; set; }
```

Property Value

[float](#)

time

```
public float time { get; set; }
```

Property Value

[float](#)

value

```
public float value { get; set; }
```

Property Value

[float](#)

weightedMode

```
public WeightedMode weightedMode { get; set; }
```

Property Value

WeightedMode

Methods

toKeyframe()

```
public Keyframe toKeyframe()
```

Returns

Keyframe

Class MathematicsExtensions

Namespace: [UnityNexus.Extensions](#)

Assembly: EggExtensions.dll

```
public static class MathematicsExtensions
```

Inheritance

[object](#) ← MathematicsExtensions

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

Fields

kEpsilon

```
public const float kEpsilon = 1E-05
```

Field Value

[float](#)

kEpsilonNormalSqrt

```
public const float kEpsilonNormalSqrt = 1E-15
```

Field Value

[float](#)

Methods

angle(float2, float2)

```
public static float angle(float2 from, float2 to)
```

Parameters

from float2

to float2

Returns

[float](#)

angle(float3, float3)

```
public static float angle(float3 from, float3 to)
```

Parameters

from float3

to float3

Returns

[float](#)

signedAngle(float2, float2)

```
public static float signedAngle(float2 from, float2 to)
```

Parameters

from float2

to float2

Returns

[float](#) ↗

signedAngle(float3, float3, float3)

```
public static float signedAngle(float3 from, float3 to, float3 axis)
```

Parameters

from float3

to float3

axis float3

Returns

[float](#) ↗