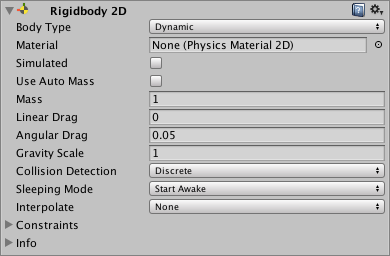
|  |  |
| --- | --- |
| 翻译前字数 | 3246 |
| 目录 | * [Unity User Manual (5.6)](https://docs.unity3d.com/560/Documentation/Manual/UnityManual.html) / [2D](https://docs.unity3d.com/560/Documentation/Manual/Unity2D.html) / [Physics Reference 2D](https://docs.unity3d.com/560/Documentation/Manual/Physics2DReference.html) / Rigidbody 2D |
| 链接 | https://docs.unity3d.com/560/Documentation/Manual/class-Rigidbody2D.html |

**Rigidbody 2D**

[SWITCH TO SCRIPTING](https://docs.unity3d.com/560/Documentation/ScriptReference/Rigidbody2D.html)

A Rigidbody 2D component places an object under the control of the physics engine. Many concepts familiar from the standard [Rigidbody](http://docs.unity3d.com/Manual/class-Rigidbody.html) component carry over to Rigidbody 2D; the differences are that in 2D, objects can only move in the XY plane and can only rotate on an axis perpendicular to that plane.

2D刚体组件是受物理引擎控制下的对象。很多部分类似于标准的刚体组件且是它的延伸。不同之处是它是用于2D中，对象只可以在XY平面移动并且仅可以在垂直于平面的一个轴旋转。

The Rigidbody 2D component. This appears differently in the Unity Editor depending on which Body Type you have selected. See [Body Type](https://docs.unity3d.com/560/Documentation/Manual/class-Rigidbody2D.html" \l "BodyType), below, to learn more.

Rigidbody2D组件。选择的Body Type不同，展示在Unity编辑器上也会有所不同。参见Body Type，以下，学习更多。

**How a Rigidbody 2D works**

**2D刚体如何工作**

Usually, the Unity Editor’s Transform component defines how a GameObject (and its child GameObjects) is positioned, rotated and scaled within the Scene. When it is changed, it updates other components, which may update things like where they render or where colliders are positioned. The 2D physics engine is able to move colliders and make them interact with each other, so a method is required for the physics engine to communicate this movement of colliders back to the Transform components. This movement and connection with colliders is what a Rigidbody 2D component is for.

通常，Unity编辑器的变换组件定义了游戏对象（和它的子GameObject）在场景中的位置，旋转和缩放。当它发生改变时，它将更新其他组件，这可能会更新他们渲染呈现的位置和碰撞器的位置。2D物理引擎会移动碰撞器并且让它们相互作用，所以物理引擎需要一个方法来将碰撞器的移动传递变换组件。2D刚体组件是为碰撞器的移动和链接所用。

The Rigidbody 2D component overrides the Transform and updates it to a position/rotation defined by the Rigidbody 2D. Note that while you can still override the Rigidbody 2D by modifying the Transform component yourself (because Unity exposes all properties on all components), doing so will cause problems such as GameObjects passing through or into each other, and unpredictable movement.

2D刚体组件覆盖并且更新了变换组件的坐标，旋转。需注意虽然您仍可以通过改变2D刚体来修改Transform组件（因为Unity公开了所有组件的属性），但这样会导致像GameObject相穿的问题，并且有不可预测的移动。

Any Collider 2D component added to the same GameObject or child GameObject is implicitly attached to that Rigidbody 2D. When a Collider 2D is attached to the Rigidbody 2D, it moves with it. A Collider 2D should never be moved directly using the Transform or any collider offset; the Rigidbody 2D should be moved instead. This offers the best performance and ensures correct collision detection. Collider 2Ds attached to the same Rigidbody 2D won’t collide with each other.

任何添加到相同游戏对象或子对象的2D碰撞体会隐式附着在2D刚体组件上。当在Rigidbody2D上赋予2D碰撞体组件时，那它将移动2D刚体。一个2D碰撞体不会直接使用变换组件或任何碰撞体的偏移来让其移动。2D刚体应该代替它移动。这样保证了最好的性能和正确的碰撞检测。2D碰撞体附着于不会碰撞其他物体的2D刚体自身。

This means you can create a set of colliders that act effectively as a single compound collider, all moving and rotating in sync with the Rigidbody 2D.

您可以创建一些碰撞体作为一整个的复合碰撞体，所有的移动和旋转都与2D刚体同步。

When designing a Scene, you are free to use a default Rigidbody 2D and start attaching colliders. These colliders allow any other colliders attached to different Rigidbody 2Ds to collide with each other.

设计场景时，您可以随意的使用默认的2D刚体并添加到碰撞体上。这些碰撞体会和任何添加到不同2D刚体的碰撞体相互碰撞。

**Tip**

**技巧**

Adding a **Rigidbody 2D** allows a sprite to move in a physically convincing way by applying forces from the scripting API. When the appropriate collider component is also attached to the sprite GameObject, it is affected by collisions with other moving GameObjects. Using physics simplifies many common gameplay mechanics and allows for realistic behavior with minimal coding.

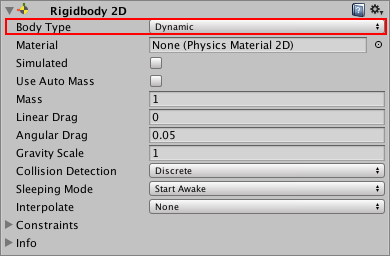
增加2D刚体会使精灵通过脚本施加的力来拥有一个令人信服的物理移动方式。当恰当的碰撞体组件也添到精灵游戏对象时，它会通过碰撞来影响其他移动着的游戏对象。使用物理简化了常见的游戏机制并且用最少的编码实现了真实的行为。

**Body Type**

**Body Type**

The Rigidbody 2D component has a setting at the top labelled **Body Type**. The option you choose for this affects the other settings available on the component.

2D刚体组件在顶部有一个名为BodyType的设置。您选择的选项会影响在组件上其他的设置。



There are three options for **Body Type**; each defines a common and fixed behavior. Any Collider 2D attached to a Rigidbody 2D inherits the Rigidbody 2D’s **Body Type**. The three options are:

BodyType有三种选项；每种都有一种常见和复杂的行为。任何添加到2D刚体的2D碰撞体会继承2D刚体的BodeType选项。

* **Dynamic**
* 动态
* **Kinematic**
* 动力学
* **Static**
* 静态

The option you choose defines:

选择定义的选项

Movement (position & rotation) behavior

运动（位置&旋转）行为

* Collider interaction

碰撞体的相互作用

Note that although Rigidbody 2Ds are often described as colliding with each other, it is the Collider 2Ds attached to each of those bodies which collide. Rigidbody 2Ds cannot collide with each other without colliders.

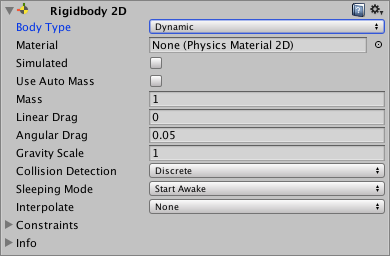
注意虽然2D刚体常用语描述相互之间的碰撞，它是添加了2D碰撞体之间的每个碰撞的部分。2D刚体不能和没有碰撞体的物体相互碰撞

Changing the Body Type of a Rigidbody 2D can be a tricky process. When a Body Type changes, various mass-related internal properties are recalculated immediately, and all existing contacts for the Collider 2Ds attached to the Rigidbody 2D need to be re-evaluated during the GameObject’s next [FixedUpdate](https://docs.unity3d.com/560/Documentation/ScriptReference/MonoBehaviour.FixedUpdate.html). Depending on how many contacts and Collider 2Ds are attached to the body, changing the Body Type can cause variations in performance.

改变2D刚体的BodyType是一个棘手的过程。当BodyType改变时，将立即重新计算各种质量有关的内部属性，并且会在游戏对象的下一个FixedUpdate重新计算所有碰到的2D碰撞器已接触的碰撞。根据2D碰撞器接触和添加到身体的2D碰撞器的数量，改变BodyType可能会导致性能的变化。

**Body Type: Dynamic**

**Body Type:动态**



A **Dynamic** Rigidbody 2D is designed to move under simulation. It has the full set of properties available to it such as finite mass and drag, and is affected by gravity and forces. A Dynamic body will collide with every other body type, and is the most interactive of body types. This is the default body type for a Rigidbody 2D, because it is the most common body type for things that need to move. It’s also the most performance-expensive body type, because of its dynamic nature and interactivity with everything around it. All Rigidbody 2D properties are available with this body type.

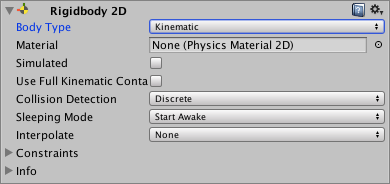
动态的2D刚体用于模拟移动。它具有可用的全部属性，类似有限的质量和阻力，并且受重力和力的影响。动态的物体会与其他类型的物体相互碰撞，并且它是影响最大的物体类型。默认的物理类型是为了Rigidbody2D,因为它是最常见的需要移动的物体类型。它也是性能昂贵的物体类型，因为自身的动态性和与周围事物的互动性。所有2D刚体属性都可以用于这种物体类型。

Do not use the Transform component to set the position or rotation of a **Dynamic** Rigidbody 2D. The simulation repositions a **Dynamic** Rigidbody 2D according to its velocity; you can change this directly via forces applied to it by scripts, or indirectly via collisions and gravity.

不要使用变换组件来设置一个动态2D刚体的位置或者选择。根据模拟其速度重新定位2D动态刚体。您可以直接通过脚本来直接改变力，或者简介通过碰撞和重力改变力。

| **Property:** | **Function:** |
| --- | --- |
| **Body Type** | Set the RigidBody 2D’s component settings, so that you can manipulate movement (position and rotation) behavior and Collider 2D interaction.  Options are: **Dynamic**, **Kinematic**, **Static**  在2D刚体组件的设置，使用运动（位置和选择）行为和2D碰撞器的相互作用。  选项：动态，动力学，静态 |
| **Material** | Use this to specify a common material for all Collider 2Ds attached to a specific parent Rigidbody 2D.  指定添加到父节点的2D刚体的所有2D碰撞器的公共材质 **Note:** A Collider 2D uses its own Material property if it has one set. If there is no Material specified here or in the Collider 2D, the default option is **None (Physics Material 2D)**. This uses a default Material which you can set in the [Physics 2D Settings](https://docs.unity3d.com/560/Documentation/Manual/class-Physics2DManager.html) window.  注意：2D碰撞器如果已有一个自己的材质则使用自己的。如果没有材质指定，或在2D碰撞器上没有，则默认的选项是空**(Physics Material 2D)**。默认材质您可以在2D物理设置窗口中设置。 A Collider 2D uses the following order of priority to determine which **Material** setting to use:  2D碰撞器使用以下材质设置的优先级决定： 1. A Physics Material 2D specified on the Collider 2D itself.  1.在2D碰撞器中自身的2D物理材质的指定。 2. A Physics Material 2D specified on the attached Rigidbody 2D.  2D刚体指定的2D物理材质。 A Physics Material 2D default material specified in the [Physics 2D Settings](https://docs.unity3d.com/560/Documentation/Manual/class-Physics2DManager.html).  在2D物理设置中指定默认的2D物理材质 **TIP:** Use this to ensure that all Collider 2Ds attached to the same **Static** Body Type Rigidbody 2D can all use the same Material.  TIP:确保所有添到相同的静态Body Type的2D刚体的2D碰撞器使用相同材质。 |
| **Simulated** | Enable **Simulated** (check the box) if you want the Rigidbody 2D and any attached Collider 2Ds and Joint 2Ds to interact with the physics simulation during run time. If this is disabled (the box is unchecked), these components do not interact with the simulation. See [Rigidbody 2D properties: Simulated](https://docs.unity3d.com/560/Documentation/Manual/class-Rigidbody2D.html#SimulatedProperty), below, for more details. This box is checked by default.  如果您想2D刚体和任何添加到2D碰撞器和2D Joint在运行时物理相互影响则启用模拟（勾选框）。如果禁用了（未勾选此框），那这些组件在模拟时将不会互相影响。参见2D刚体的属性：模拟，以下参考更多信息。此框默认是启用的。 |
| **Use Auto Mass** | Check the box if you want the Rigidbody 2D to automatically detect the GameObject’s mass from its Collider 2D.  如果您需要2D刚体自动从其2D碰撞器检测游戏对象的质量则勾选此框。 |
| **Mass** | Define the mass of the Rigidbody 2D. This is grayed out if you have selected Use Auto Mass.  定义了2D刚体的质量。如果您勾选了Use Auto Mass则显示为灰色。 |
| **Linear Drag** | Drag coefficient affecting positional movement.  影响位置移动的拖动系数 |
| **Angular Drag** | Drag coefficient affecting rotational movement.  影响旋转移动的拖动系数 |
| **Gravity Scale** | Define the degree to which the GameObject is affected by gravity.  定义了游戏物体受重力影响的程度。 |
| **Collision Detection** | Define how collisions between Collider 2D are detected.  定义了如何检测和其他游戏对象碰撞 |
| Discrete | When you set the **Collision Detection** to **Discrete**, GameObjects with Rigidbody 2Ds and Collider 2Ds can overlap or pass through each other during a physics update, if they are moving fast enough. Collision contacts are only generated at the new position.  当您将Collision Detection设为Discrete时，在物理更新时，如果移动够快，拥有2D刚体和2D碰撞器可以穿过或重叠，碰撞接触会在一个新的位置生成。 |
| Continuous | When the **Collision Detection** is set to **Continuous**, GameObjects with Rigidbody 2Ds and Collider 2Ds do not pass through each other during an update. Instead, Unity calculates the first impact point of any of the Collider 2Ds, and moves the GameObject there. Note that this takes more CPU time than **Discrete**.  当Collision Detection设为Continuous时，在更新时拥有2D刚体和2D碰撞器游戏对象将不会互相穿透。另外，Unity会计算任何2D碰撞器碰撞时的第一个点，并移动游戏对象到这里。注意，选择Discrete会更加花费CPU时间。 |
| **Sleeping Mode** | Define how the GameObject “sleeps” to save processor time when it is at rest.  定义游戏对象在休息时如何“休眠”以节省处理器的时间。 |
| Never Sleep | Sleeping is disabled (this should be avoided where possible, as it can impact system resources).  禁用休眠（尽可能避免这种可能，因为它会影响系统资源） |
| Start Awake | GameObject is initially awake.  游戏对象初始化时是醒着的 |
| Start Asleep | GameObject is initially asleep but can be woken by collisions.  游戏对象在初始化时是休眠的但是可以被碰撞唤醒。 |
| **Interpolate** | Define how the GameObject’s movement is interpolated between physics updates (useful when motion tends to be jerky).  定义了游戏对象在运动时和物理更新是如何相互作用的（当物体处于剧烈运动时起效） |
| None | No movement smoothing is applied.  应用无运动平滑 |
| Interpolate | Movement is smoothed based on the GameObject’s positions in previous frames.  基于游戏对象的前一帧的坐标的平滑运动 |
| Extrapolate | Movement is smoothed based on an estimate of its position in the next frame.  基于后一帧的估计坐标平滑运动 |
| **Constraints** | Define any restrictions on the Rigidbody 2D’s motion.  定义了在2D刚体运动的限制 |
| **Freeze Position** | Stops the Rigidbody 2D moving in the world X & Y axes selectively.  选择性的停止2D刚体在世界的X&Y轴的运动 |
| **Freeze Rotation** | Stops the Rigidbody 2D rotating around the Z axes selectively.  选择性的停止2D刚体绕Z轴的旋转 |

**Body Type: Kinematic**



A **Kinematic** Rigidbody 2D is designed to move under simulation, but only under very explicit user control. While a **Dynamic** Rigidbody 2D is affected by gravity and forces, a **Kinematic** Rigidbody 2D isn’t. For this reason, it is fast and has a lower demand on system resources than a **Dynamic** Rigidbody 2D. **Kinematic** Rigidbody 2D is designed to be repositioned explicitly via [Rigidbody2D.MovePosition](http://docs.unity3d.com/ScriptReference/Rigidbody2D.MovePosition.html) or [Rigidbody2D.MoveRotation](http://docs.unity3d.com/ScriptReference/Rigidbody2D.MoveRotation.html). Use physics queries to detect collisions, and scripts to decide where and how the Rigidbody 2D should move.

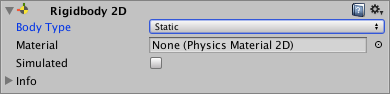
2D动力学刚体的是为移动模拟设计的，但是只能在非常明确的用户控制下。2D动态刚体受重力和力影响，而2D动力学刚体则不是。由于这个原因，它非常高效并且比2D动态刚体更省系统资源。2D动力学刚体是为明确的通过Rigidbody2D.MovePosition 或Rigidbody2D.MoveRotation而设计的。使用物理查询来检测碰撞，以及通过脚本来决定2D刚体如何移动。

A **Kinematic** Rigidbody 2D does still move via its velocity, but the velocity is not affected by forces or gravity. A **Kinematic** Rigidbody 2D does not collide with other **Kinematic** Rigidbody 2Ds or with **Static** Rigidbody 2Ds; it only collides with **Dynamic** Rigidbody 2Ds. Similar to a **Static** Rigidbody 2D (see below), a **Kinematic** Rigidbody 2D behaves like an immovable object (as if it has infinite mass) during collisions. Mass-related properties are not available with this Body Type.

2D动力学刚体通过它的速度来移动，但是它的速度不受力或重力的影响。2D动力学刚体不会和其他2D动力学刚体或者2D静态刚体发生碰撞。它只会和其他2D动态刚体发生碰撞。类似2D静态刚体（见下），2D动力学刚体在碰撞时的行为类似不动的对象（它有无限的质量）。此Body Type有关质量的属性无效。

| **Property:** | **Function:** |
| --- | --- |
| **Body Type** | Set the RigidBody 2D’s component settings, so that you can manipulate movement (position and rotation) behavior and Collider 2D interaction.  Options are: **Dynamic**, **Kinematic**, **Static**  2D刚体组件的设置，使用运动（位置和选择）行为和2D碰撞器的相互作用  选项：动态，动力学，静态 |
| **Material** | Use this to specify a common material for all Collider 2Ds attached to a specific parent Rigidbody 2D.  指定添加到父节点的2D刚体的所有2D碰撞器的公共材质 **Note:** A Collider 2D uses its own Material property if it has one set. If there is no Material specified here or in the Collider 2D, the default option is **None (Physics Material 2D)**. This uses a default Material which you can set in the [Physics 2D Settings](https://docs.unity3d.com/560/Documentation/Manual/class-Physics2DManager.html) window.  注意：2D碰撞器如果已有一个自己的材质则使用自己的。如果没有材质指定，或在2D碰撞器上没有，则默认的选项是空**(Physics Material 2D)**。默认材质您可以在2D物理设置窗口中设置。 A Collider 2D uses the following order of priority to determine which **Material** setting to use:  2D碰撞器使用以下材质设置的优先级决定： 1. A Physics Material 2D specified on the Collider 2D itself.  1.在2D碰撞器中自身的2D物理材质的指定。 2. A Physics Material 2D specified on the attached Rigidbody 2D.  2D刚体指定的2D物理材质。 A Physics Material 2D default material specified in the [Physics 2D Settings](https://docs.unity3d.com/560/Documentation/Manual/class-Physics2DManager.html).  在2D物理设置中指定默认的2D物理材质 **TIP:** Use this to ensure that all Collider 2Ds attached to the same **Static** Body Type Rigidbody 2D can all use the same Material.  TIP:确保所有添到相同的静态Body Type的2D刚体的2D碰撞器使用相同材质。 |
| **Simulated** | Enable **Simulated** (check the box) if you want the Rigidbody 2D and any attached Collider 2Ds and Joint 2Ds to interact with the physics simulation during run time.  如果您想要运行时的2D刚体和任何添加的2D碰撞器和2D joint互相受物理模拟影响则启用模拟（勾选框）。  If this is disabled (the box is unchecked), these components do not interact with the simulation. See [Rigidbody 2D properties: Simulated](https://docs.unity3d.com/560/Documentation/Manual/class-Rigidbody2D.html#SimulatedProperty), below, for more details. This box is checked by default.  如果禁用（不勾选框），那么这些组件将不会互相受模拟的影响。参见[Rigidbody 2D properties: Simulated](https://docs.unity3d.com/560/Documentation/Manual/class-Rigidbody2D.html#SimulatedProperty),以下参见更多详情，这个框默认是打开的。 |
| **Use Full Kinematic Contacts** | Enable this setting (check the box) if you want the **Kinematic** Rigidbody 2D to collide with all Rigidbody 2D Body Types.  如果您希望2D动力学刚体与所有2D刚体类型相撞，请启用此设置（勾选框）  This is similar to a **Dynamic** Rigidbody 2D, except the **Kinematic** Rigidbody 2D is not moved by the physics engine when contacting another Rigidbody 2D component; instead it acts as an immovable object, with infinite mass. When **Use Full Kinematic Contacts** is disabled, the **Kinematic** Rigidbody 2D only collides with **Dynamic** Rigidbody 2Ds. See [Rigidbody 2D properties: Use Full Kinematic Contacts](https://docs.unity3d.com/560/Documentation/Manual/class-Rigidbody2D.html#UseFullKinematicContactsProperty), below, for more details. This box is unchecked by default.  这类似于2D动态刚体，除了2D动力学刚体不会受到物理引擎接触其他2D刚体组件时移动；取而代之的是一个静止且有无限质量的对象。当**Use Full Kinematic Contacts**禁用时，2D动力学刚体仅可以碰到2D动态刚体。参见[Rigidbody 2D properties: Use Full Kinematic Contacts](https://docs.unity3d.com/560/Documentation/Manual/class-Rigidbody2D.html#UseFullKinematicContactsProperty)参考更多详情，此选项默认是关闭的。 |
| **Collision Detection** | Define how collisions between Collider 2D are detected.  定义了如何检测和其他游戏对象碰撞 |
| Discrete | When you set the **Collision Detection** to **Discrete**, GameObjects with Rigidbody 2Ds and Collider 2Ds can overlap or pass through each other during a physics update, if they are moving fast enough. Collision contacts are only generated at the new position.  当您将Collision Detection设为Discrete时，在物理更新时，如果移动够快，拥有2D刚体和2D碰撞器可以穿过或重叠，碰撞接触会在一个新的位置生成。 |
| Continuous | When the **Collision Detection** is set to **Continuous**, GameObjects with Rigidbody 2Ds and Collider 2Ds do not pass through each other during an update. Instead, Unity calculates the first impact point of any of the Collider 2Ds, and moves the GameObject there. Note that this takes more CPU time than **Discrete**.  当Collision Detection设为Continuous时，在更新时拥有2D刚体和2D碰撞器游戏对象将不会互相穿透。另外，Unity会计算任何2D碰撞器碰撞时的第一个点，并移动游戏对象到这里。注意，选择Discrete会更加花费CPU时间。 |
| **Sleeping Mode** | Define how the GameObject “sleeps” to save processor time when it is at rest.  定义游戏对象在休息时如何“休眠”以节省处理器的时间。 |
| Never Sleep | Sleeping is disabled (this should be avoided where possible, as it can impact system resources).  禁用休眠（尽可能避免这种可能，因为它会影响系统资源） |
| Start Awake | GameObject is initially awake.  游戏对象初始化时是醒着的 |
| Start Asleep | GameObject is initially asleep but can be woken by collisions.  游戏对象在初始化时是休眠的但是可以被碰撞唤醒。 |
| **Interpolate** | Define how the GameObject’s movement is interpolated between physics updates (useful when motion tends to be jerky).  定义了游戏对象在运动时和物理更新是如何相互作用的（当物体处于剧烈运动时起效） |
| None | No movement smoothing is applied.  应用无运动平滑 |
| Interpolate | Movement is smoothed based on the GameObject’s positions in previous frames.  基于游戏对象的前一帧的坐标的平滑运动 |
| Extrapolate | Movement is smoothed based on an estimate of its position in the next frame.  基于后一帧的估计坐标平滑运动 |
| **Constraints** | Define any restrictions on the Rigidbody 2D’s motion.  定义了在2D刚体运动的限制 |
| **Freeze Position** | Stops the Rigidbody 2D moving in the world X & Y axes selectively.  选择性的停止2D刚体在世界的X&Y轴的运动 |
| **Freeze Rotation** | Stops the Rigidbody 2D rotating around the Z axes selectively.  选择性的停止2D刚体绕Z轴的旋转 |

**Body Type: Static**



A **Static** Rigidbody 2D is designed to not move under simulation at all; if anything collides with it, a **Static** Rigidbody 2D behaves like an immovable object (as though it has infinite mass). It is also the least resource-intensive body type to use. A **Static** body only collides with **Dynamic** Rigidbody 2Ds. Having two **Static** Rigidbody 2Ds collide is not supported, since they are not designed to move.

Only a very limited set of properties are available for this Body Type.

| **Property:** | **Function:** |
| --- | --- |
| **Body Type** | Set the RigidBody 2D’s component settings, so that you can manipulate movement (position and rotation) behavior and Collider 2D interaction.  2D刚体组件的设置，以便在2D碰撞器相互作用时操纵移动（位置和旋转）行为 Options are: **Dynamic**, **Kinematic**, **Static**  选项：**Dynamic**, **Kinematic**, **Static** |
| **Material** | Use this to specify a common material for all Collider 2Ds attached to a specific parent Rigidbody 2D.  指定添加到父节点的2D刚体的所有2D碰撞器的公共材质 **Note:** A Collider 2D uses its own Material property if it has one set. If there is no Material specified here or in the Collider 2D, the default option is **None (Physics Material 2D)**. This uses a default Material which you can set in the [Physics 2D Settings](https://docs.unity3d.com/560/Documentation/Manual/class-Physics2DManager.html) window.  注意：2D碰撞器如果已有一个自己的材质则使用自己的。如果没有材质指定，或在2D碰撞器上没有，则默认的选项是空**(Physics Material 2D)**。默认材质您可以在2D物理设置窗口中设置。 A Collider 2D uses the following order of priority to determine which **Material** setting to use:  2D碰撞器使用以下材质设置的优先级决定： 1. A Physics Material 2D specified on the Collider 2D itself.  1.在2D碰撞器中自身的2D物理材质的指定。 2. A Physics Material 2D specified on the attached Rigidbody 2D.  2D刚体指定的2D物理材质。 A Physics Material 2D default material specified in the [Physics 2D Settings](https://docs.unity3d.com/560/Documentation/Manual/class-Physics2DManager.html).  在2D物理设置中指定默认的2D物理材质 **TIP:** Use this to ensure that all Collider 2Ds attached to the same **Static** Body Type Rigidbody 2D can all use the same Material.  TIP:确保所有添到相同的静态Body Type的2D刚体的2D碰撞器使用相同材质。 |
| **Simulated** | Enable **Simulated** (check the box) if you want the Rigidbody 2D and any attached Collider 2Ds and Joint 2Ds to interact with the physics simulation during run time.  如果您想要运行时的2D刚体和任何添加的2D碰撞器和2D joint互相受物理模拟影响则启用模拟（勾选框）。  If this is disabled (the box is unchecked), these components do not interact with the simulation. See [Rigidbody 2D properties: Simulated](https://docs.unity3d.com/560/Documentation/Manual/class-Rigidbody2D.html#SimulatedProperty), below, for more details. This box is checked by default.  如果禁用（不勾选框），那么这些组件将不会互相受模拟的影响。参见[Rigidbody 2D properties: Simulated](https://docs.unity3d.com/560/Documentation/Manual/class-Rigidbody2D.html#SimulatedProperty),以下参见更多详情，这个框默认是打开的。 |

There are two ways to mark a Rigidbody 2D as **Static**:

有两种方式让2D刚体标为静态：

1. For the GameObject with the Collider 2D component not to have a Rigidbody 2D component at all. All such Collider 2Ds are internally considered to be attached to a single hidden **Static** Rigidbody 2D component.

带2D碰撞体组件且没有2D刚体组件的游戏物体。这些2D碰撞体在内部被添加到一个隐藏2D静态刚体组件上去。

1. For the GameObject to have a Rigidbody 2D and for that Rigidbody 2D to be set to **Static**.

为游戏对象添加一个2D刚体让2D刚体设为静态。

Method 1 is a shorthand for making **Static** Collider 2Ds. When creating large numbers of **Static** Collider 2Ds, it is easier not to have to add a Rigidbody 2D for each GameObject with a Collider 2D.

方法1是标为2D静态碰撞器高效的方法。创建大量的2D静态碰撞器，要比为每个带2D碰撞器的游戏对象添加2D刚体要简单。

Method 2 exists for performance reasons. If a **Static** Collider 2D needs to be moved or reconfigured at run time, it is faster to do so when it has its own Rigidbody 2D. If a group of Collider 2Ds needs to be moved or reconfigured at run time, it is faster to have them all be children of one parent Rigidbody 2D marked as **Static** than to move each GameObject individually.

方法2存在性能问题。如果2D静态碰撞需要在运行时移动或者重新配置，那么它拥有自己的2D刚体时会更快。如果一组2D碰撞体需要在运行时移动或者重新配置，那么把它们的父节点设为静态2D刚体要比单独移动每一个游戏对象要快。

**Note:** As stated above, **Static** Rigidbody 2Ds are designed not to move, and collisions between two **Static** Rigidbody 2D objects that intersect are not registered. However, **Static** Rigidbody 2Ds and **Kinematic** Rigidbody 2Ds will interact with each other if one of their Collider 2Ds is set to be a trigger. There is also a feature that changes what a **Kinematic** body will interact with (see [Use Full Kinematic Contacts](https://docs.unity3d.com/560/Documentation/Manual/class-Rigidbody2D.html#UseFullKinematicContactsProperty), below).

注：如上所述，静态2D刚体是不会移动的，两个2D静态刚体的之间的碰撞的相交不会注册。然而，2D静态刚体和2D动力学刚体彼此相交时，其中他们中的一个设为触发器。还有一个可以改变了动力体相互影响的功能（参见以下使用完备的动力学碰撞）

**Rigidbody 2D properties**

**Rigidbody 2D properties**

**Simulated**

**模拟**

Use the **Simulated** property to stop (unchecked) and start (checked) a Rigidbody 2D and any attached Collider 2Ds and Joint 2Ds from interacting with the 2D physics simulation. Changing this property is much more memory and processor-efficient than enabling or disabling individual Collider 2D and Joint 2D components.

使用模拟属性可停止或启用在物理模拟时2D刚体和任何2D碰撞体或2D关节的相互影响。改变这项属性会比启用或者禁用单独的2D碰撞体和2D关节组件花费更多内存和处理器效率。

When the **Simulation** box is checked, the following occurs:

当模拟选项启用时，会发生：

* The Rigidbody 2D moves via the simulation (gravity and physics forces are applied)

2D刚体通过模拟移动（重力和应用到物理的力）。

* Any attached Collider 2Ds continue creating new contacts and continuously re-evaluate contacts

任何附加的2D碰撞体会继续创建新的接触并不断重新计算接触。

* Any attached Joint 2Ds are simulated and constrain the attached Rigidbody 2D

任何附加的2D关节是模拟和约束于附加的2D刚体的

* All internal physics objects for Rigidbody 2D, Collider 2D & Joint 2D stay in memory

When the **Simulated** box is unchecked, the following occurs:

任何内部的物理对象：2D刚体，2D碰撞器&2D关节都产生内存。当模拟框勾选时，会发生以下情况：

* The Rigidbody 2D is not moved by the simulation (gravity and physics forces are not applied)

2D刚体将不会受到模拟的移动（不会应用重力和物理的力）

* The Rigidbody 2D does not create new contacts, and any attached Collider 2D contacts are destroyed

2D刚体不会产生新的接触，并且和2D碰撞体产生的碰撞会销毁。

* Any attached Joint 2Ds are not simulated, and do not constrain any attached Rigidbody 2Ds

附加的2D关节不模拟，并且不会约束附加的2D刚体。

* All internal physics objects for Rigidbody 2D, Collider 2D and Joint 2D are left in memory

所有内部的物理对象:2D刚体，2D碰撞体和2D关节将不会产生内存。

**Why is unchecking Simulated more efficient than individual component controls?**

**为什么不选中模拟要比单组件控制更高效呢？**

In the 2D physics simulation, a Rigidbody 2D component controls the position and rotation of attached Collider 2D components, and allows Joint 2D components to use these positions and rotations as anchor points. A Collider 2D moves when the Rigidbody 2D it is attached to moves. The Collider 2D then calculates contacts with other Collider 2Ds attached to other Rigidbody 2Ds. Joint 2Ds also constrain Rigidbody 2D positions and rotations. All of this takes simulation time.

在2D物理模拟中，2D刚体组件控制添加在2D碰撞体组件的位置和旋转，2D关节组件通过一个锚点来控制坐标和旋转。附于2D刚体移动时，2D碰撞盒移动。2D碰撞器会计算和其他添加到2D刚体的2D碰撞器的碰撞。2D Joint也约束2D刚体的坐标和旋转。所有这一切都需要模拟时间。

You can stop and start individual elements of the 2D physics simulation by enabling and disabling components individually. You can do this on both Collider 2D and Joint 2D components. However, enabling and disabling individual elements of the physics simulations has memory use and processor power costs. When elements of the simulation are disabled, the 2D physics engine doesn’t produce any internal physics-based objects to simulate. When elements of the simulation are enabled, the 2D physics engine does have internal physics-based objects to simulate. Enabling and disabling of 2D physics simulation components means internal GameObjects and physics-based components have to be created and destroyed; disabling the simulation is easier and more efficient than disabling individual components.

您可以通过单独启用和禁用组件来停止并启动2D物理模拟的各种元素。您可以在2D碰撞器和2D Joint两者中执行此操作。然而。启用或者禁用各个元素都会使物理模拟消耗更多的内存和处理器消耗。当模拟的元素被禁用时，2D物理引擎不产生任何基于物理的内部对象模拟。当元素模拟启用时，2D物理引擎会产生基于物理的内部对象的模拟。启用和禁用2D物理模拟组件意味着内部的游戏对象和基于物理的元素必须被创建或销毁；禁用模拟比禁用单个组件更简单，更有效。

NOTE: When a Rigidbody 2D’s **Simulated** option is unchecked, any attached Collider 2D is effectively ‘invisible’, that is; it cannot be detected by any physics queries, such as [Physics.Raycast](scriptref:Physics.Raycast.html).

注：当禁用了2D刚体模拟，任何添加的2D碰撞体将隐形。它将不可被物理查询检测到，例如Physics.Raycast.

**Use Full Kinematic Contacts**

**Use Full Kinematic Contacts**

Enable this setting (check the checkbox) if you want the **Kinematic** Rigidbody 2D to collide with all Rigidbody 2D Body Types. This is similar to a **Dynamic** Rigidbody 2D, except the **Kinematic** Rigidbody 2D is not moved by the physics engine when contacting another Rigidbody 2D; it acts as an immovable object, with infinite mass.

如果您需要2D动力学刚体和其他任意类型的2D刚体碰撞则启用此设置。类似2D动态刚体，2D动力学刚体在碰撞其他2D刚体时，物理引擎不会移动它。它是一个不可移动的物体，有无限的质量。

When this setting is disabled (unchecked), a **Kinematic** Rigidbody 2D only collides with **Dynamic** Rigidbody 2Ds; it does not collide with other **Kinematic** Rigidbody 2Ds or **Static** Rigidbody 2Ds (note that trigger colliders are an exception to this rule). This means that no collision scripting callbacks ([OnCollisionEnter](scriptref:Collider2D.OnCollisionEnter2D.html), [OnCollisionStay](scriptref:Collider2D.OnCollisionStay2D.html), [OnCollisionExit](scriptref:Collider2D.OnCollisionExit2D.html)) occur.

当此选项禁用（未选择）时，2D动力学刚体仅和其他2D刚体碰撞。它不会与其他2D动力学刚体和2D静态刚体发生碰撞（注意触发器刚体是一个例外规则），意思是不会产生脚本回调([OnCollisionEnter](scriptref:Collider2D.OnCollisionEnter2D.html), [OnCollisionStay](scriptref:Collider2D.OnCollisionStay2D.html), [OnCollisionExit](scriptref:Collider2D.OnCollisionExit2D.html))。

This can be inconvenient when you are using physics queries (such as [Physics.Raycast](scriptref:Physics.Raycast.html)) to detect where and how a Rigidbody 2D should move, and when you require multiple **Kinematic** Rigidbody 2Ds to interact with each other. Enable **Use Full Kinematic Contacts** to make **Kinematic** Rigidbody 2D components interact in this way.

当您使用物理检测（类似Physics.Raycast)来决定2D刚体如何移动是不方便的，并且当您有多个2D运动学刚体碰撞时也不方便。使用完整的运动学碰撞让2D运动学刚体组件相互作用。

**Use Full Kinematic Contacts** allows explicit position and rotation control of a **Kinematic** Rigidbody 2D, but still allows full collision callbacks. In a set-up where you need explicit control of all Rigidbody 2Ds, use **Kinematic** Rigidbody 2Ds in place of **Dynamic** Rigidbody 2Ds to still have full collision callback support.

使用完整的动力学碰撞会明确的控制2D动力学刚体的位置和旋转,依然有完整的碰撞回调。需要明确的控制所有2D刚体时的体系，使用2D运动学刚体代替2D动态刚体仍有完整的碰撞回调支持。