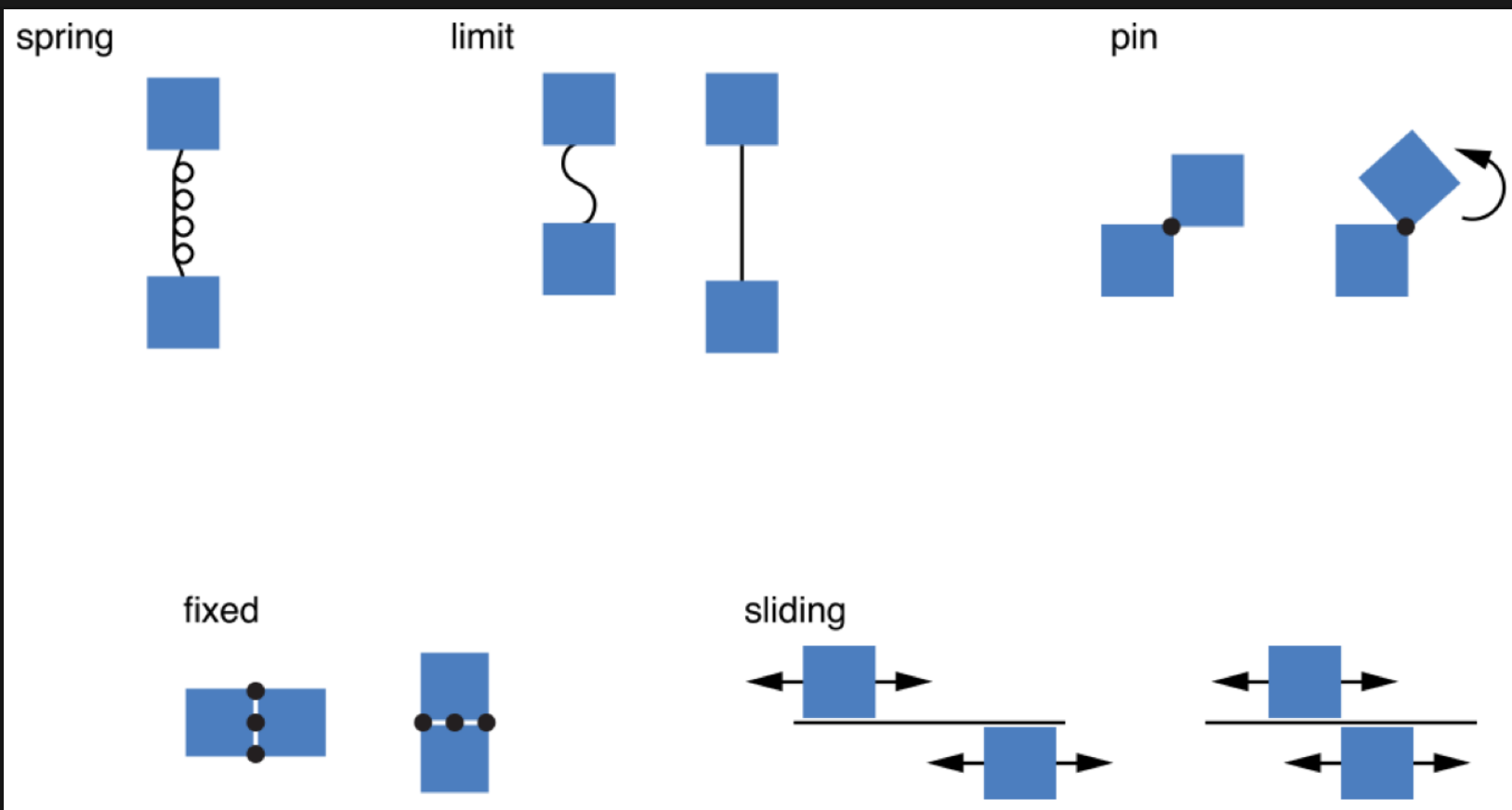


Connecting Bodies with Joints

Add joints to nodes in your scene.

Overview

Although you can make a lot of interesting games that use the physics systems already described, you can take your designs further by connecting physics bodies using joints. When the scene simulates the physics, it takes these joints into account when calculating how forces affect the bodies. The following illustration shows the types of joints you can use.



You add or remove joints using the physics world. When you create a joint, the points that connect the joint are always specified in the scene's coordinate system. This may require you to first convert from node coordinates to scene coordinates before creating the joint.

Add a Joint to Nodes in Your Scene

To use a physics joint in your app, follow these steps:

1. Create two physics bodies.
2. Attach the physics bodies to a pair of `SKNode` objects in the scene.
3. Create a joint object using one of the types shown above.
4. If necessary, configure the joint object's properties to define how the joint should operate.
5. Retrieve the scene's `SKPhysicsWorld` object.
6. Call the physics world's `add(_:)` method.

Important: The anchor arguments passed to physics joint initializers are always in the parent scene's coordinate space. You can use the `convert(_:to:)` method to calculate a node's position in scene coordinates if that node is not an immediate child of the scene.

The following code pins two nodes, `pin` and `piston`, together using a `SKPhysicsJointPin` joint. In this example, the node `pin` is a child of `connectingRod`. Because it is not an immediate child of the scene, its coordinates are converted to scene coordinates for use in initializing the joint.

```
let pinPosition = connectingRod.convert(pin.position,
                                       to: scene)
let pinJoint = SKPhysicsJointPin.joint(withBodyA: pin.physicsBody!,
                                       bodyB: piston.physicsBody!,
                                       anchor: pinPosition)
scene.physicsWorld.add(pinJoint)
```