02

OPEN ORIENTED

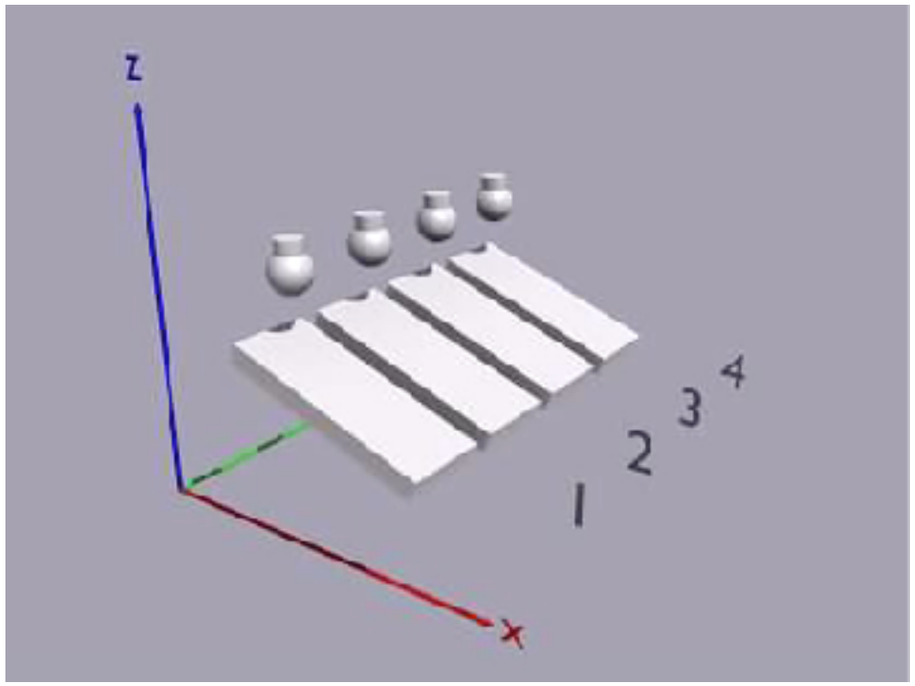
凹 凸 实 验 室

Matter.js

2D 物理引擎介绍及使用

何为物理引擎，能做什么？

- 物理引擎通过为物体（刚体）赋予真实的物理属性来计算运动、旋转和碰撞的反应，让所有物体的行为更趋向真实。
- 物理引擎的实际应用取决于物理引擎的复杂度。
- 简单的引擎，实现了刚体运动的模拟，就可以用于制作一般的游戏或者力学模拟实验。
- 复杂的引擎，实现了水流力学、空气动力学，就可以用在研究水流，模拟飞机飞行工业领域。



1、无物理效果

2、重力，没有碰撞效果

3、重力和碰撞，没有旋转效果

4、重力、碰撞以及旋转效果

物理引擎在游戏中的应用

02

CS 1.5 应用的是真实度比较差的物理引擎，人物在死亡后倒地动作是固定的，即使有障碍物也会执行这个动作，所以人物穿过了门。

而在 CS:GO 应用的 Source 物理引擎，模拟较为精细，所以人物死亡之后的姿势可以根据物理学自动计算，所以可以躺在杆上，使游戏更加真实。

Matter.js 支持的特性

02

刚体	复合体	复合材料
凹面和凸面	物理特性（质量、面积、密度等）	弹性（弹性和非弹性碰撞）
碰撞（粗测阶段、中间阶段、细测阶段）	稳定的堆叠和静止	动量守恒
摩擦力和阻力	事件监听	约束
重力	睡眠和静态物体	圆角（倒角）
视图（平移、缩放）	碰撞查询（射线追踪、区域测试）	时间缩放（减速、加速）
Canvas 渲染器（支持向量和纹理）	MatterTools 工具（创建、测试和调试）	世界状态序列化，需要 resurrect.js
跨浏览器（Chrome、Firefox、Safari、IE8+）	兼容移动端（触摸、响应）	原生 JS 实现

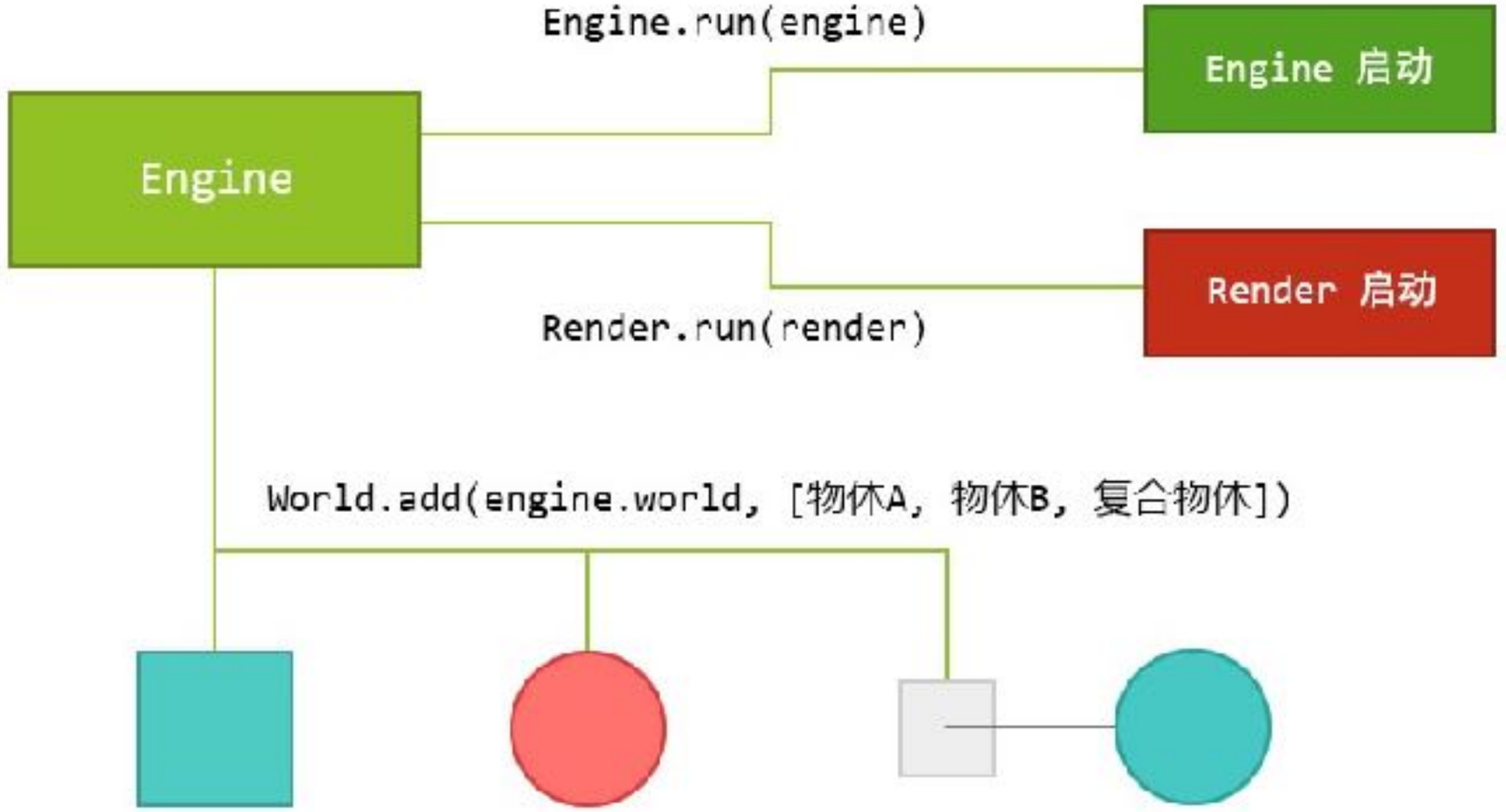
Matter.js 比较重要的概念

大多数的物理引擎对于物理模拟都有着相近的概念，不同的引擎差别在于使用的方式，功能的全面性和模拟精细度等层面。

Engine	引擎
World	世界
Render	渲染
Body	刚体
Composite	复合体
Constraint	约束
MouseConstraint	鼠标约束

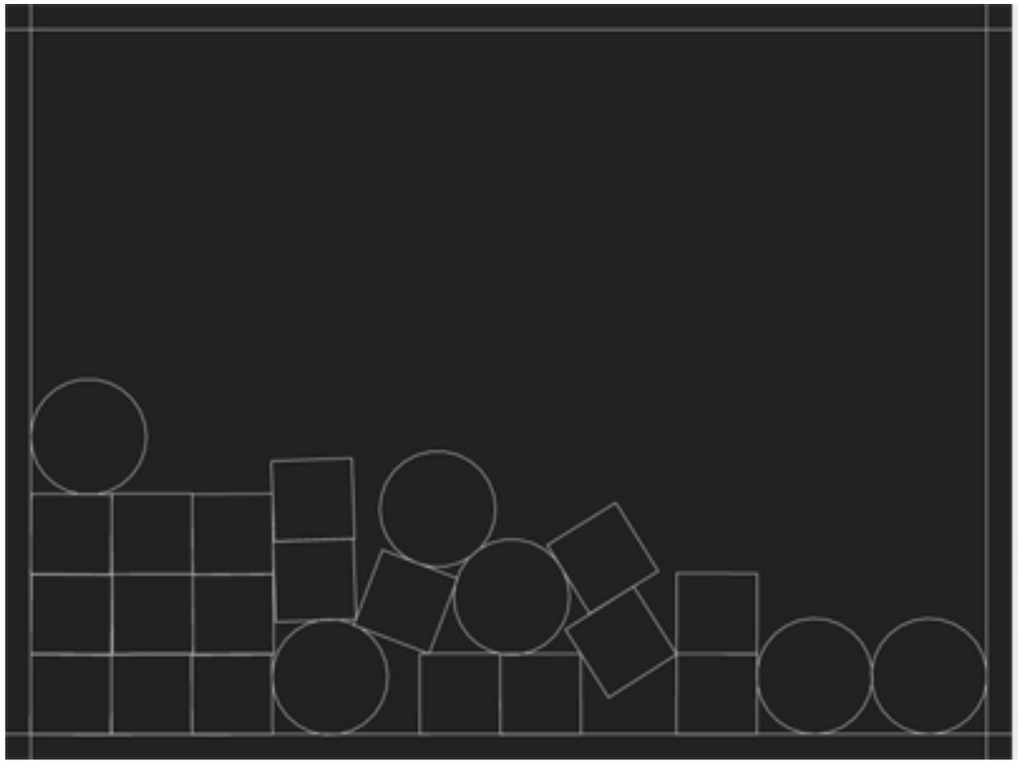
引擎（Engine）和世界（World）

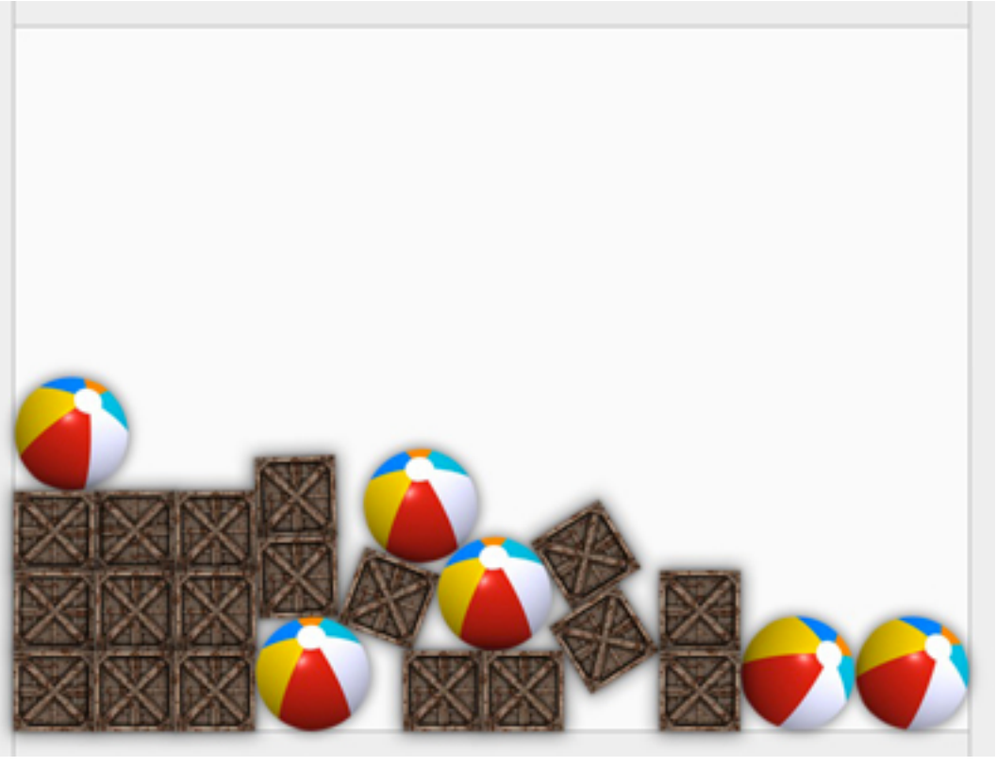
- 引擎是负责管理和更新模拟世界的容器
- 引擎可以控制时间的缩放，可以检测所有的碰撞事件，并拿到所有碰撞的物体对
- 任何物体都需要一个容身处，而存放这些物体的地方，称之为世界，物体必须添加到世界里，然后由引擎运行这个世界。



渲染（Render）

- 负责将实例渲染到 Canvas 中的渲染器，控制视图层的样式。
- 可用于开发和调试，提供了绘图功能选项，可制作简单的游戏。
- 使用时需指定要渲染 Canvas 节点和要关联的引擎

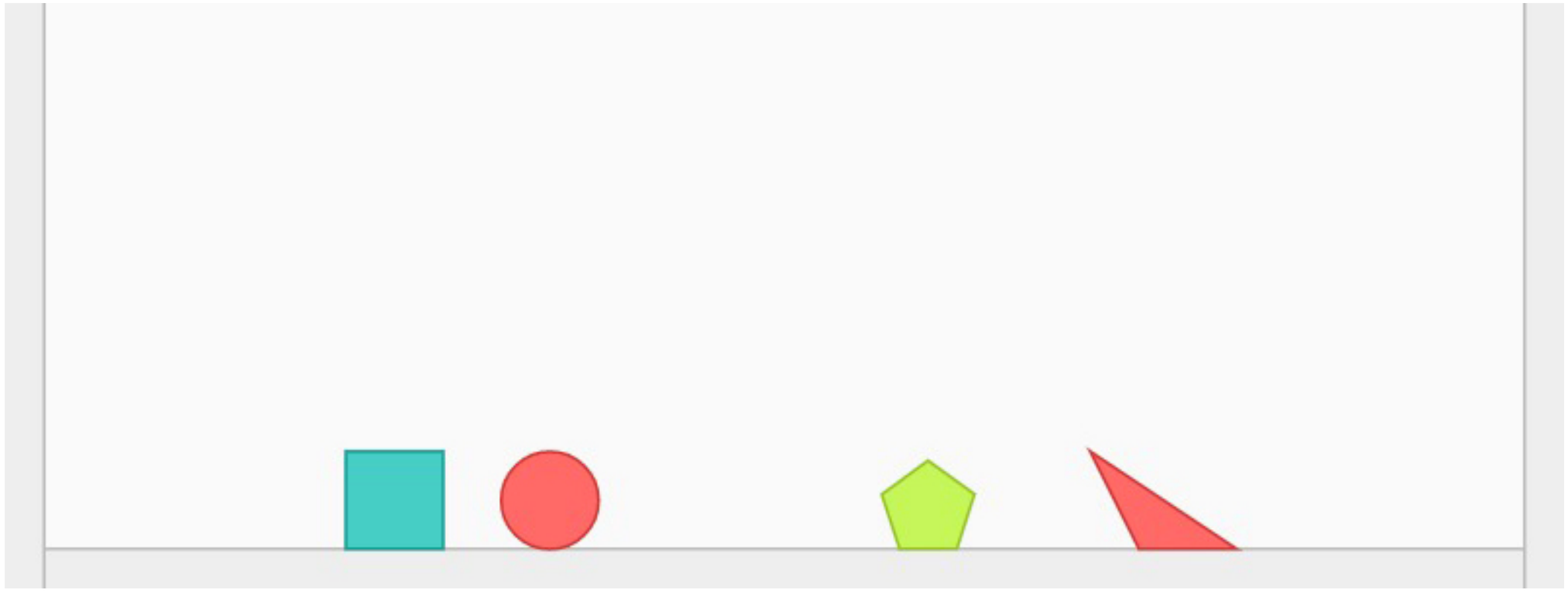




刚体（Body）

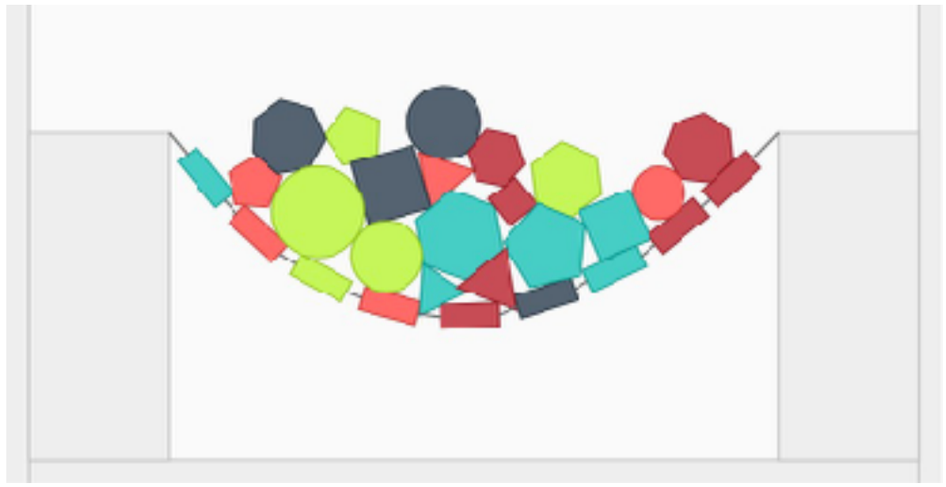
02

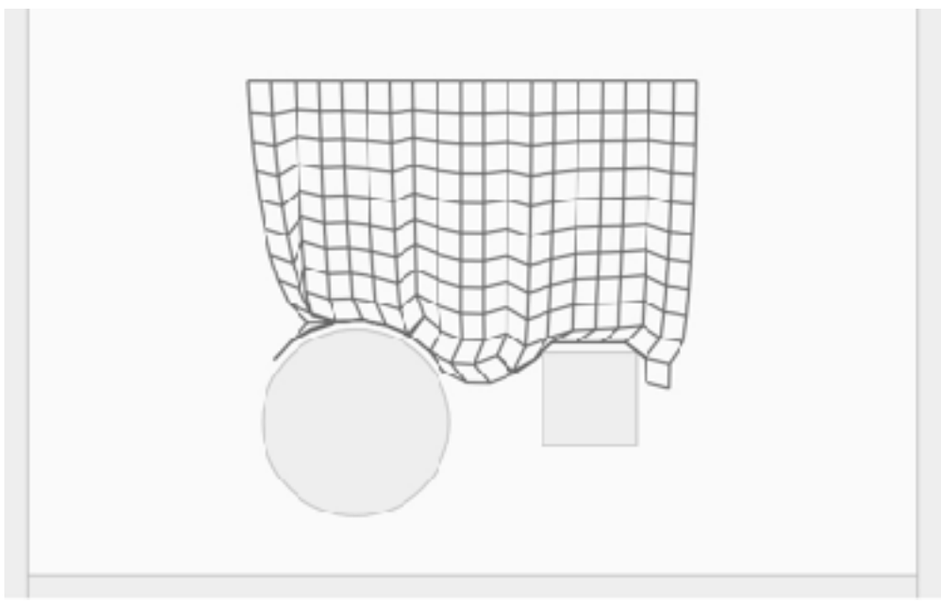
- 物体或者叫刚体，在物理引擎里特指坚硬的物体，具有固定形状，不能形变
- 每个刚体都有自己的物理属性，质量、速度、摩擦力、角度等
- Matter.js 中内置了几种刚体：矩形、多边形、圆形等等

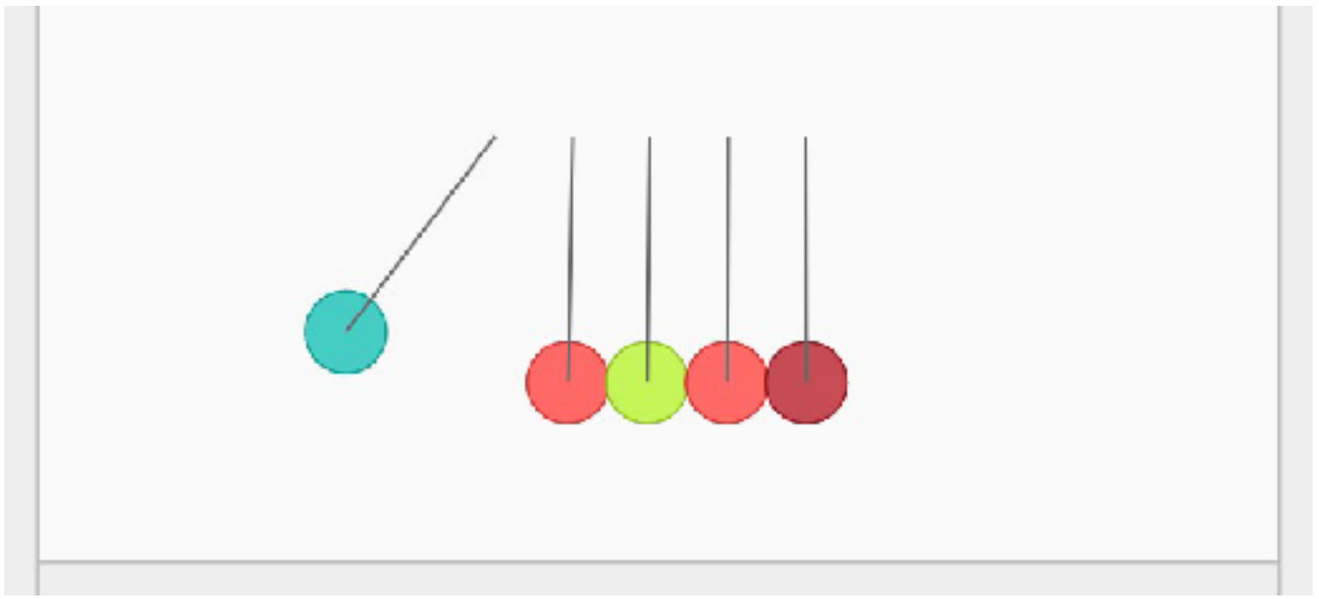


复合体（Composite）

- 由几种最基础的刚体通过约束组合到一起，就叫做复合体
- Matter.js 中内置了几种特别的复合材料：链、软体、牛顿摆球、堆叠等等

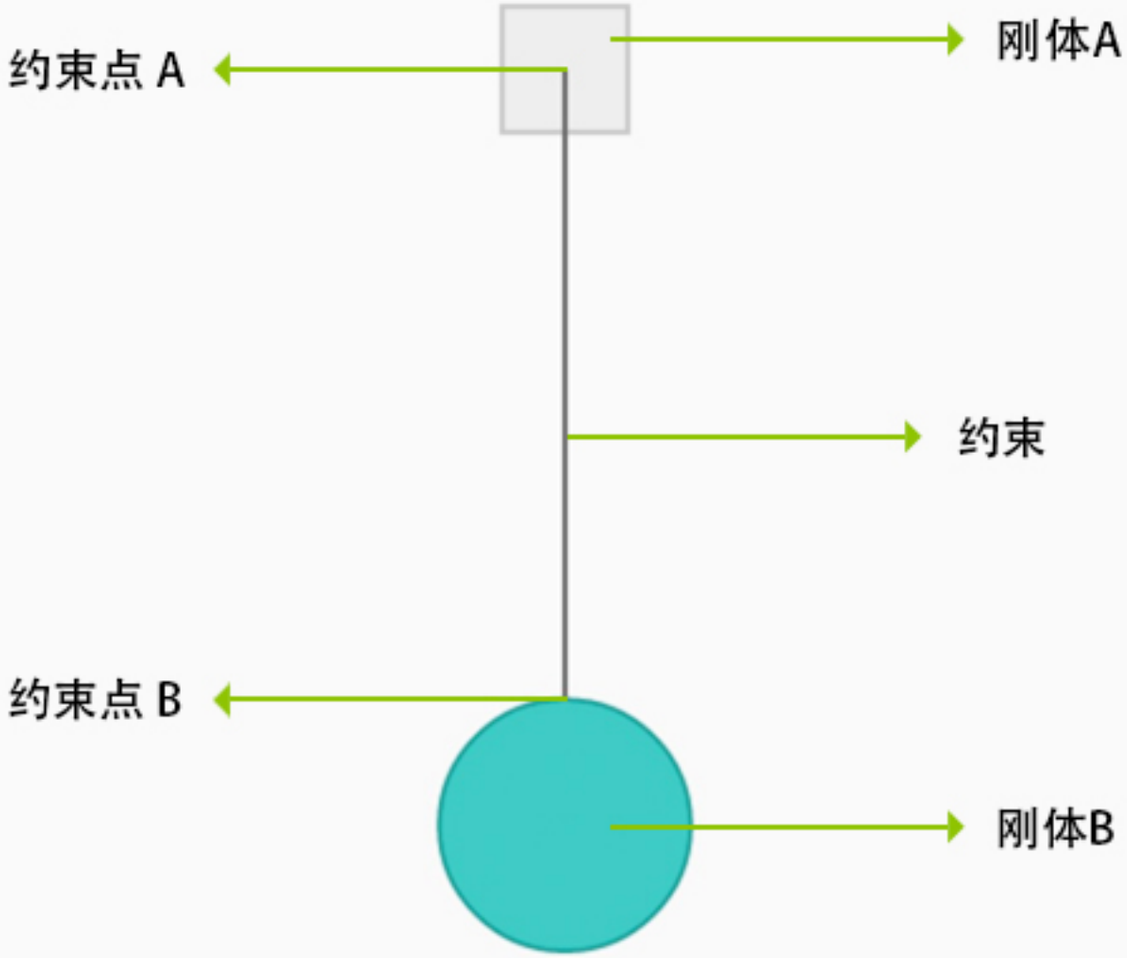






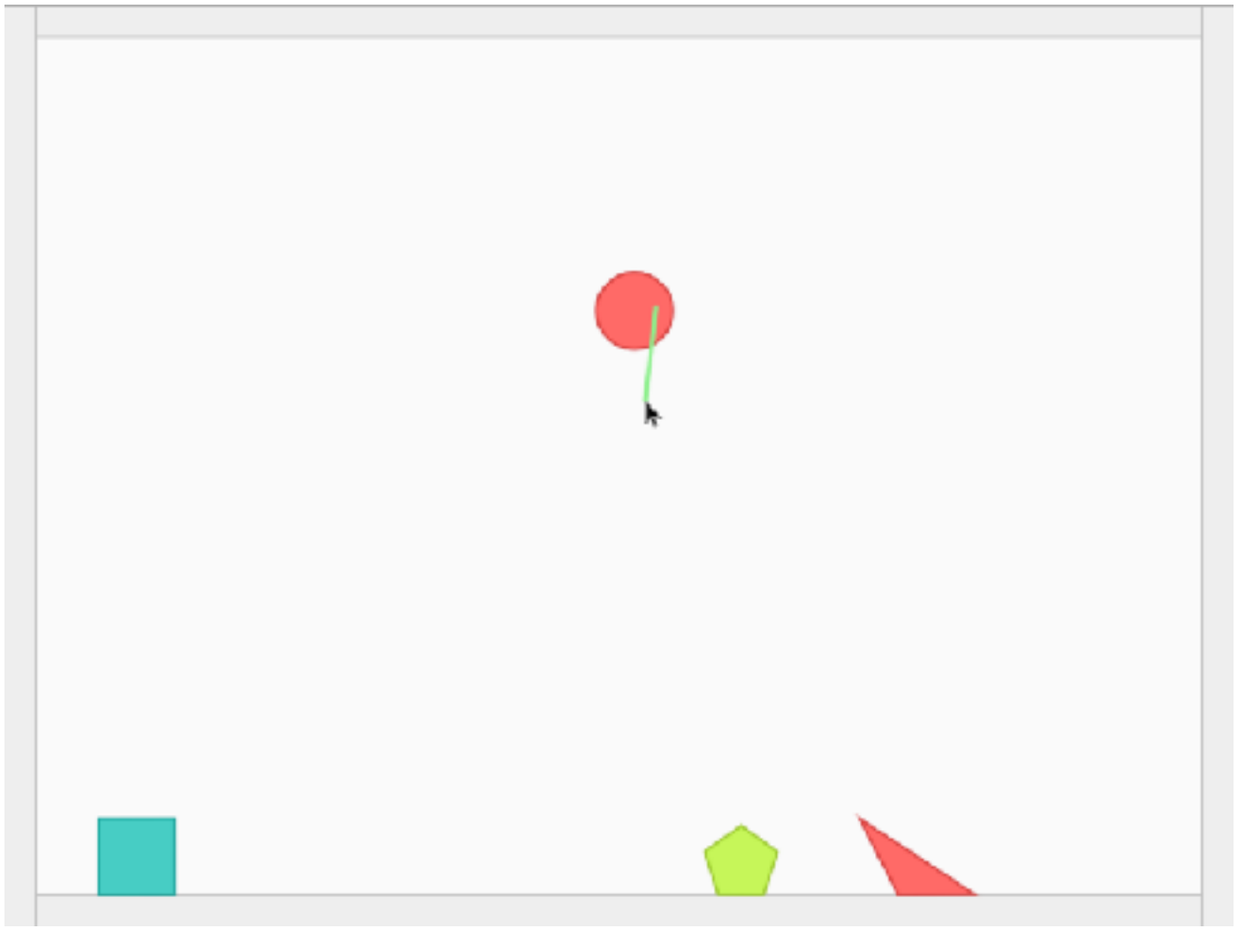
约束（Constraint）

- 约束，可理解为通过一条线，将刚体 A 和 刚体 B 两个刚体连接起来
- 被约束的两个刚体由于被连接在一起，移动就相互受到了限制。这个约束可以很宽松，也可以很紧绷，可以定义约束的距离，约束的具有弹性，可用作橡皮筋



鼠标约束（MouseConstraint）

- 如果想让刚体与用户之间有交互，那就需要在鼠标和刚体之间建立连接，也就是鼠标和刚体间的约束
- 可以设置什么标记的刚体才能被鼠标操纵
- 创建鼠标约束后，可以捕获到鼠标的各类事件



DEMO

http://brm.io/matter-js/demo/#mixed

http://magickeyboard.io/

http://msmykowski.github.io/basketball-game-matter.js/

THANKS

FOR YOUR WATCHING