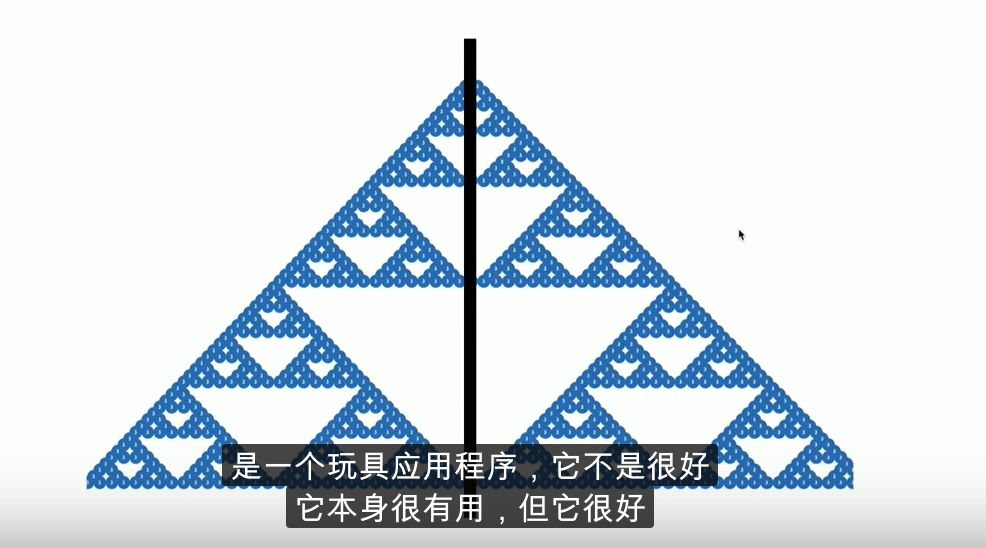


渲染引擎不是我今天想要表达的事情，我今天想要说的事情是 Fiber,

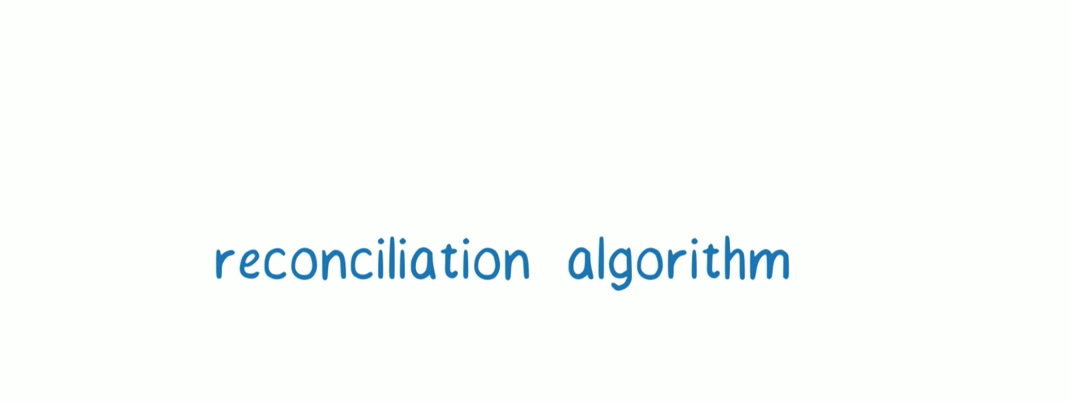
在会议上谈论Fiber, 组织者问你最重要的事情, Fiber 是最重要的。我今天很高兴跟大家来解释Fiber, 大家可能多想知道fiber 到底是什么? 如果不了解，我今天会跟大家介绍Fiber 到底是什么，他是如何工作的? 但在我做之前，我想告诉你它是做什么的? 这是左侧react 应用程序的实例.



你看到今天和现在的react ，你现在看到react 和 fiber 现状了吧.

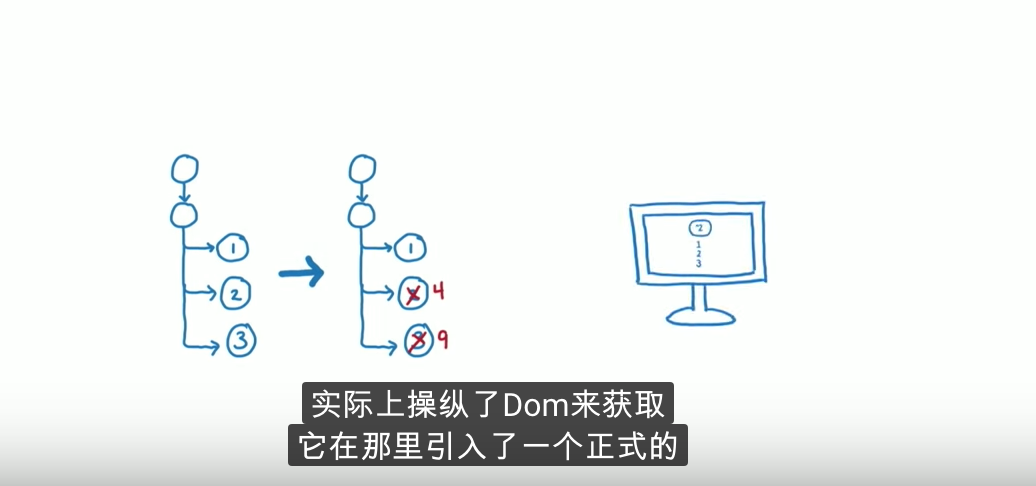
我现在来解释一下,什么是Fiber, 可以改善react 复杂应用程序的感知性能和响应能力. 现在这个你知道它做了什么. 我会解释它是什么. Fiber 是新的和解

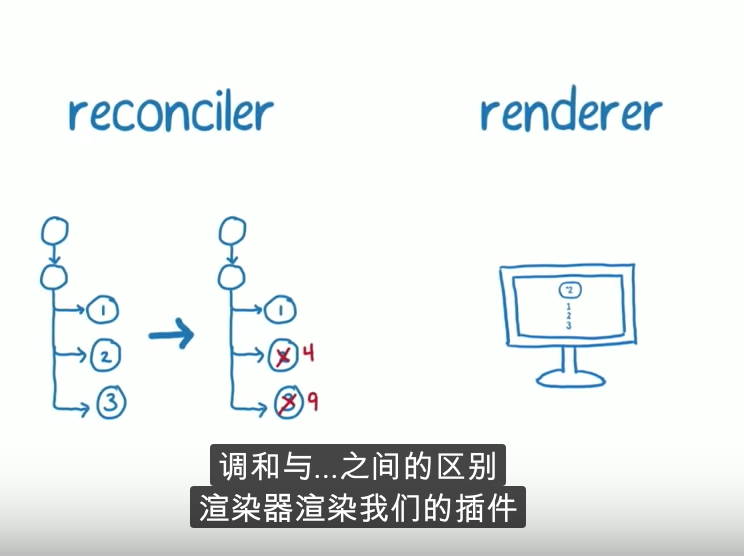
reconciliation, react 新的算法.



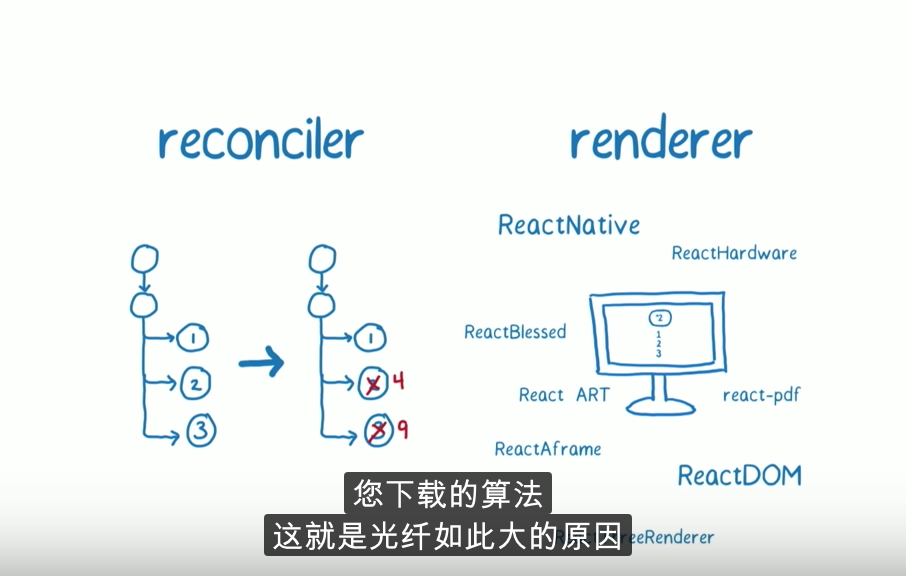
某些算法可能是新的算法，你这样解释它来自哪里, 当react 第一次出来时是杀手, 他的功能是 Virtual DOM,因为它使编码用户接口更容易.







reconciler 和 renderer 之间的区别, 渲染器渲染我们的插件你可以使用外面的渲染,



过去我们的react核心团队一直在进行彻底改写, 这个就是核心的杀手级的功能,

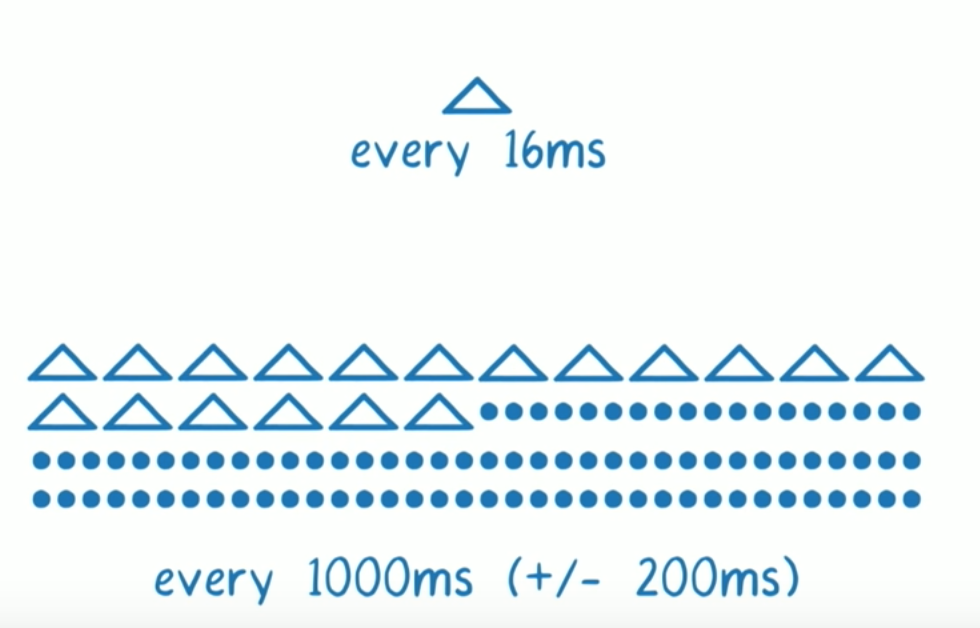
高优先级的例子, 当你打字的时候，你需要立即反馈循环，输入低优先级的实例，

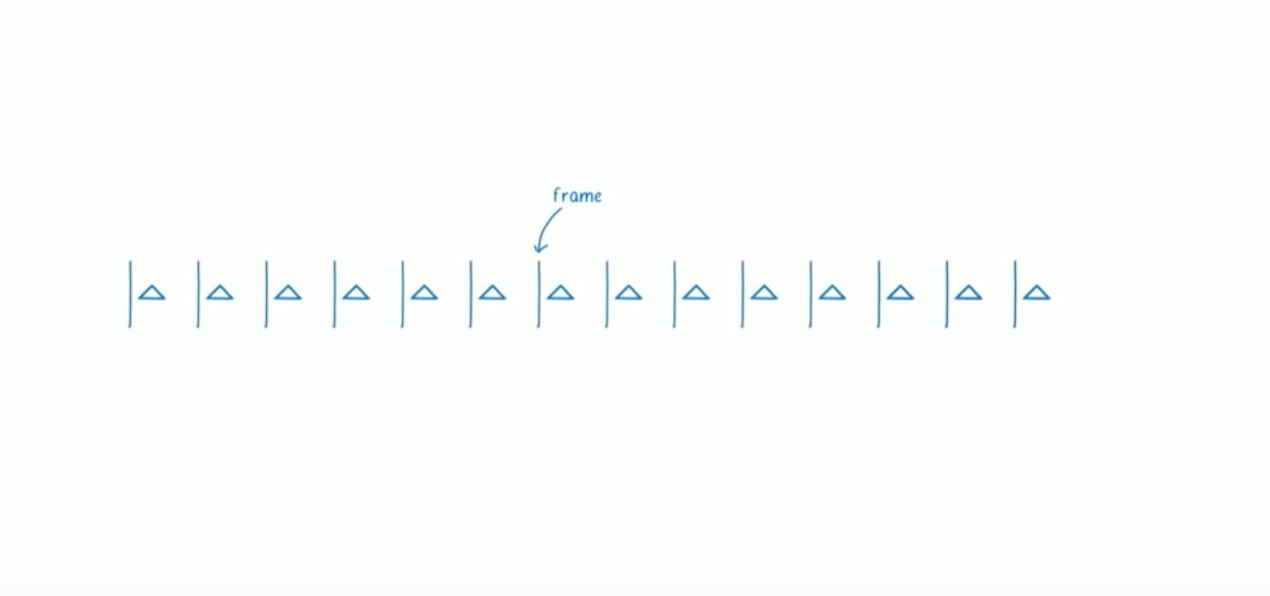
当你结合这些高优先级更新和这些低优先级更新，让我们来谈谈他们的机制,

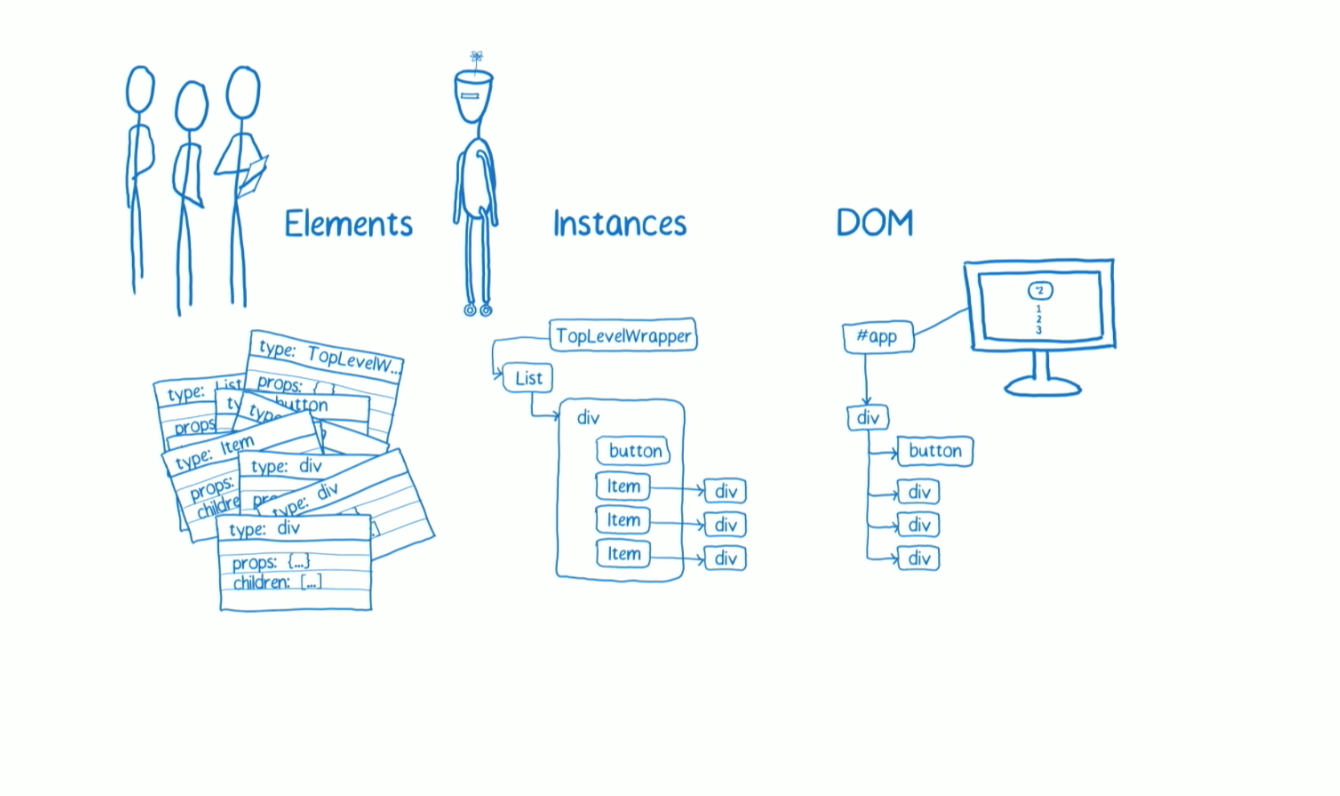
让我们来谈谈他们的机制,我们来举一个三角形的例子:

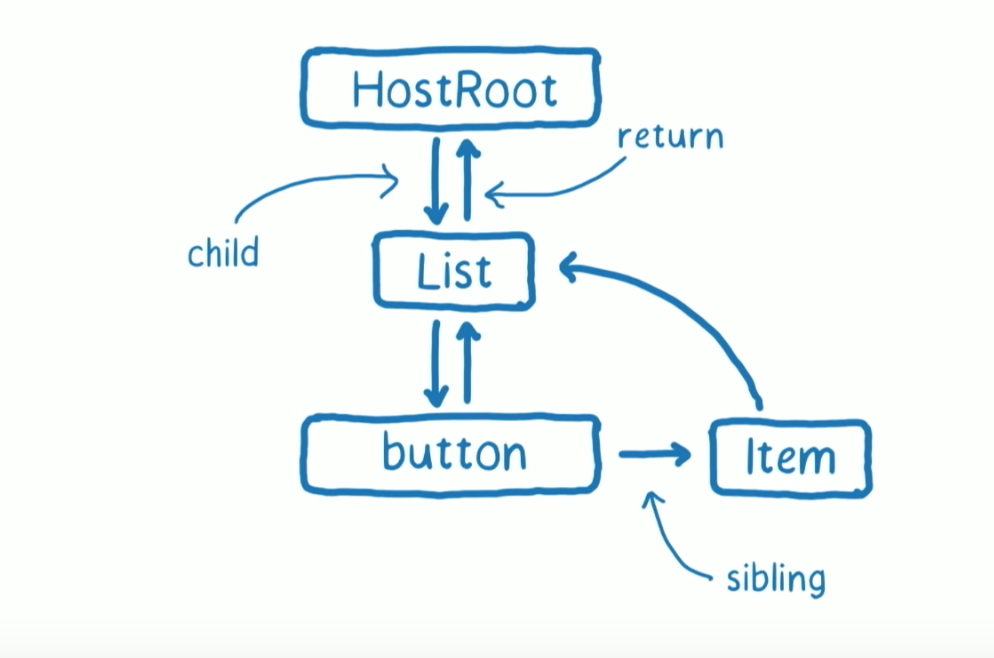
1. 我们看这个三角形，顶级然后还有更多三角形它一直向下，直到你到达你所拥有的地步,因为你不适合放点, 这些更新正在进行更多三角形,触及这个条约的不同部分更新越来越窄了，所以它不是很密集的计算,

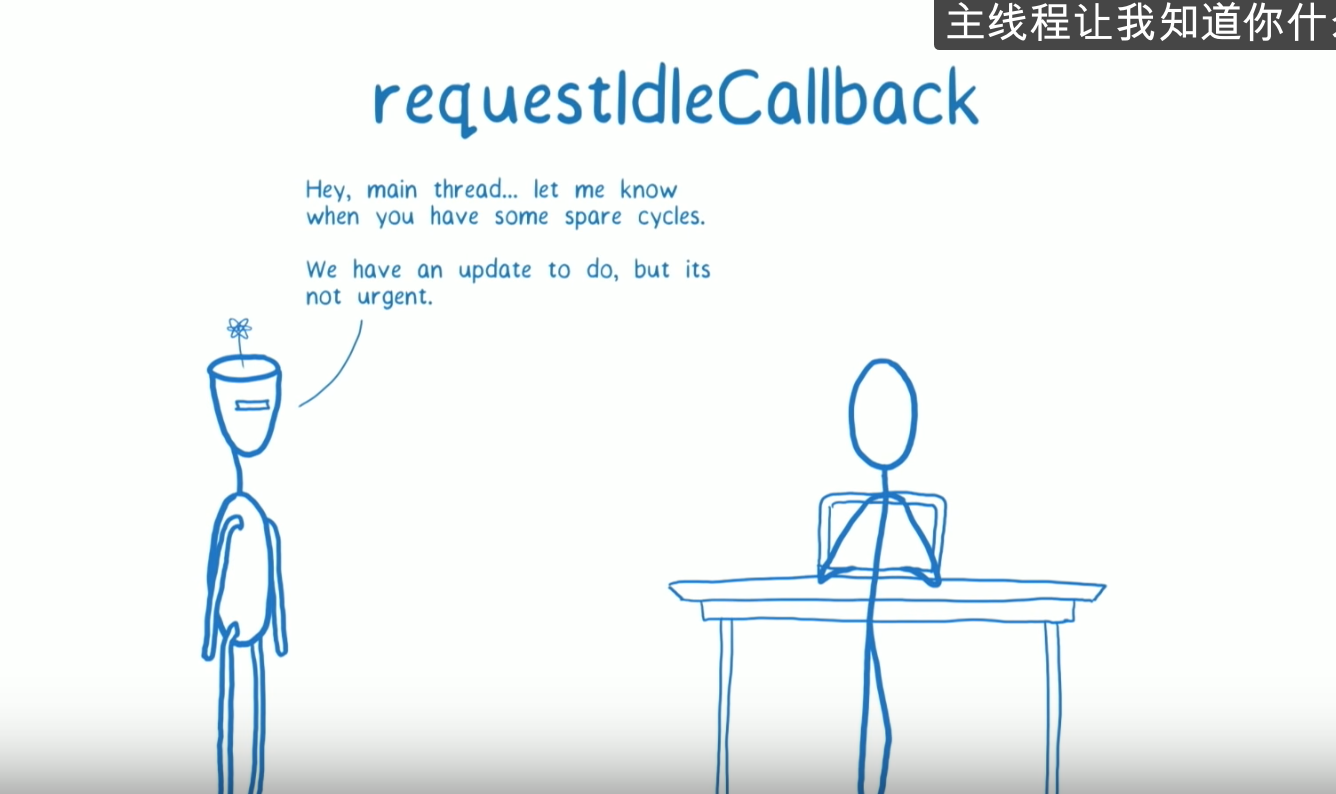
他会经常更新，如果发生的话每16ms 给你一个,另外一个流体运动，

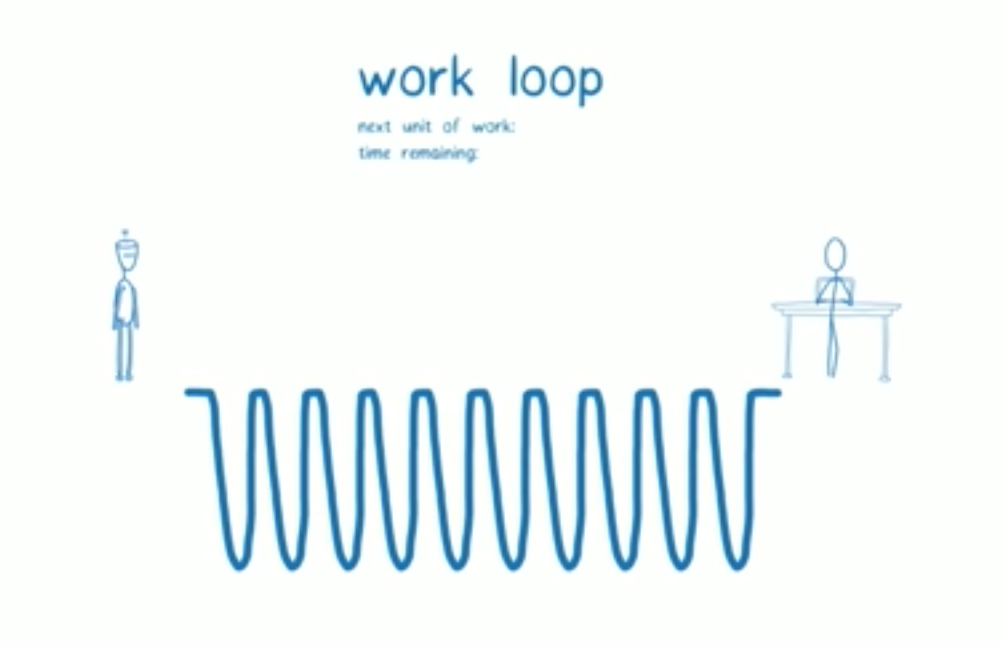


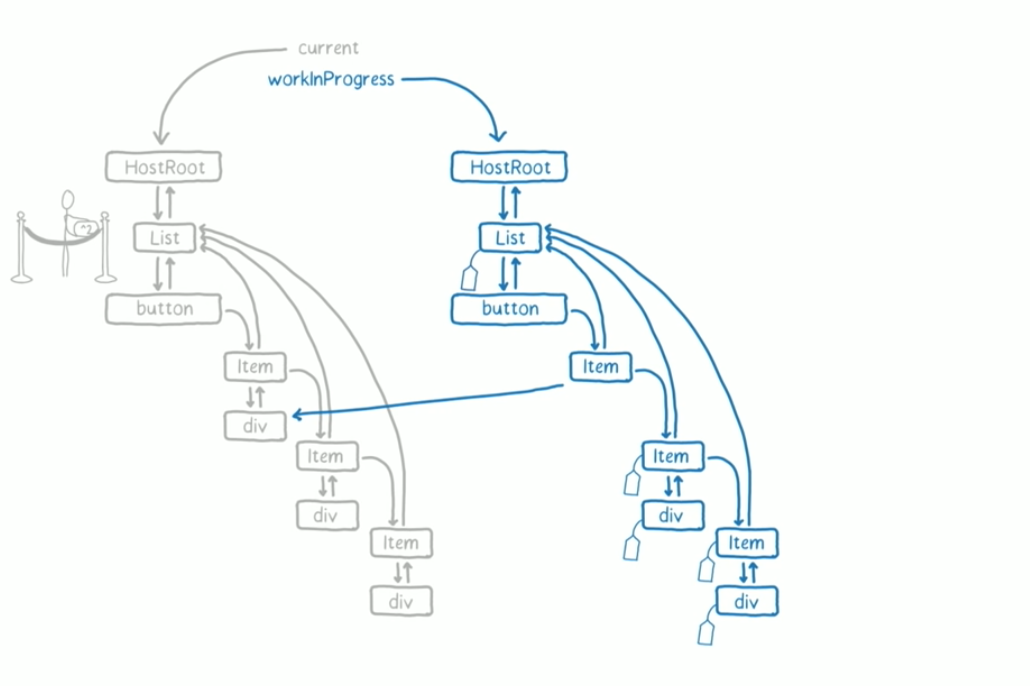


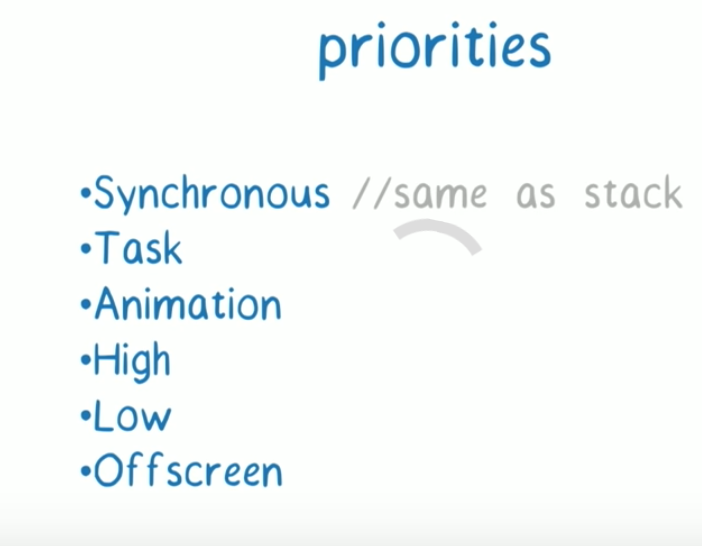












重点:

主线程和子线程之间是如何协助的.

Requstidle Callback

Work Loop

Effect List.

Work in Progress.

componentDidUpdate

分解单元的工作.

componentWillUpdate

componentDidUpdate

cooperative scheduling