# Atitit WebHook 的概念

**就是用户通过自定义回调函数的方式来改变Web应用的一种行为，这些回调函数可以由不是该Web应用官方的第三方用户或者开发人员来维护，修改。通过Webhook，你可以自定义一些行为通知到指定的URL去。**Webhook的“自定义回调函数”通常是由一些事件触发的，比如推送代码到代码库或者博客下新增一个评论，源站点会为Webhook进行HTTP请求的URI配置。用户通过配置，就可以使一个网站上的事件调用在另一个网站上表现出来，这些事件调用可以是任何事件，但通常应用的是系统集成和消息通知。

[Webhook](https://developer.github.com/webhooks/" \t "https://segmentfault.com/a/_blank)，也就是人们常说的钩子，是一个很有用的工具。你可以通过定制 Webhook 来监测你在 Github.com 上的各种事件，最常见的莫过于 ****push**** 事件。如果你设置了一个监测 push 事件的 Webhook，那么每当你的这个项目有了任何提交，这个 Webhook 都会被触发，这时 Github 就会发送一个 HTTP POST 请求到你配置好的地址。

如此一来，你就可以通过这种方式去自动完成一些重复性工作；比如，你可以用 Webhook 来自动触发一些持续集成（CI）工具的运作，比如 Travis CI；又或者是通过 Webhook 去部署你的线上服务器。

举一个简单列子说明一下，成本问题，比如我们要使用 Paypal 的 Service ，Zapier 实现他的方式不是传统的轮询调用 API 方式，他是采用 Webhook 方式，当 Paypal 完成一项任务后，会主动推送信息到 Zapier 的 Webhook，这种情况下，我们的 Zapier 才会进行相应的动作。

Webhook 不同于我们已知的 API ，API 是 Zapier 通过类似轮询的方式主动抓去数据来启动触发器， 而 Webhook 是被动服务，被动服务是指当 Service 在完成一件事情后，主动请求 Zapier，将结果告知 Zapier，这种 webhook 在粘合剂服务中，算是代价比较高的服务且，支持起来有一定难度，一般只有付费用户才能享用。