What are webhooks?

Triggered

Justin
Sep 13, 2021

♡ 52 💬 5

The TL;DR

Webhooks are APIs, but backwards - they're ways for applications to **send data** to other applications, automatically.

- webhooks let applications send data around when important **events happen**, like a user signing up
- Instead of APIs - which are passive - webhooks are **active**, which means you don't need to constantly poll them for updates
- Popular business apps like **Stripe** and **Github** provide Webhooks for things like customer checkout or pull requests
- The most popular use case for webhooks is **small integrations**, or getting your business apps to talk to each other in the ways that you need them to

Most mature B2B apps offer some sort of webhook integration, so it's worth understanding why they exist and what you can do with them.

Webhooks vs. APIs

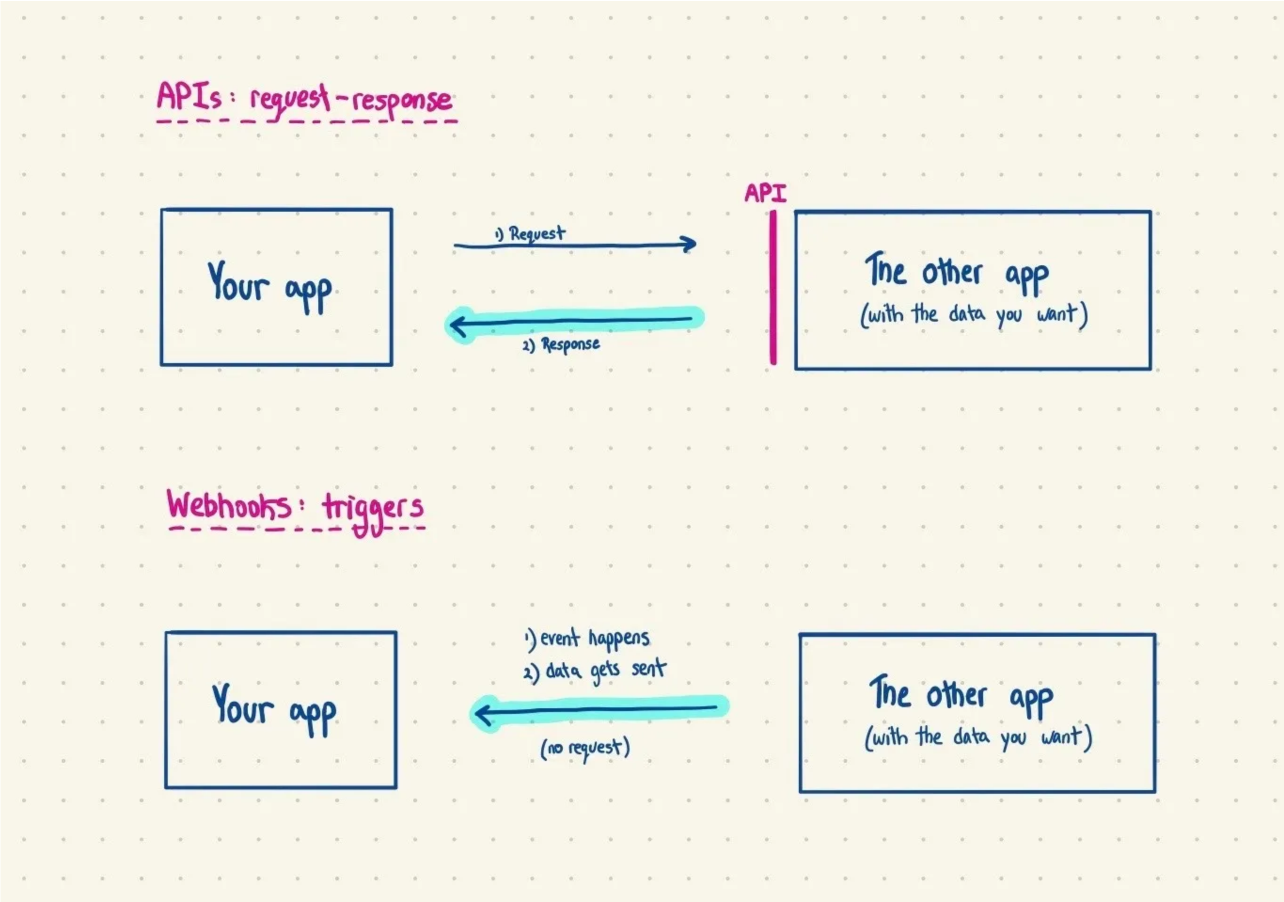
The easiest way to understand webhooks is to think of them as the opposite (in a way) of APIs.

> 🧠 **Jog your memory** 🧠
>
> A quick refresher on what an API is:
>
> - APIs let applications talk to each other - they take inputs in the form of a *request* and give you back a *response*
> - If you're building an app with a profile page, you might send a request to your users API to get back information about the current user (address,

> settings, etc.)
>
> - Most APIs are for internal use only, but apps like Twitter will often offer *external APIs* so you can work with their data programmatically
>
> If you're still a bit shaky, read [the original Technically post about APIs here](https://technically.substack.com).
>
> 🧠 **Jog your memory** 🧠

The thing about APIs is that they're **passive** - you need to send a request to get a response. APIs will never reach out to you, ask you how you're doing, and why you're behind on your car payments. Or maybe "responsive" is a better word.

Webhooks, on the other hand, are **active** - instead of taking a request and returning a response, they **send out** data when something happens internally. This is actually really similar to how you might check your Robinhood portfolio. You can open the app every day (lol more like every hour) to see how you're doing, or you can configure the app to *notify you* every time there's a meaningful change (do not do this if you value your mental health). Checking the app is passive, notifications are active.



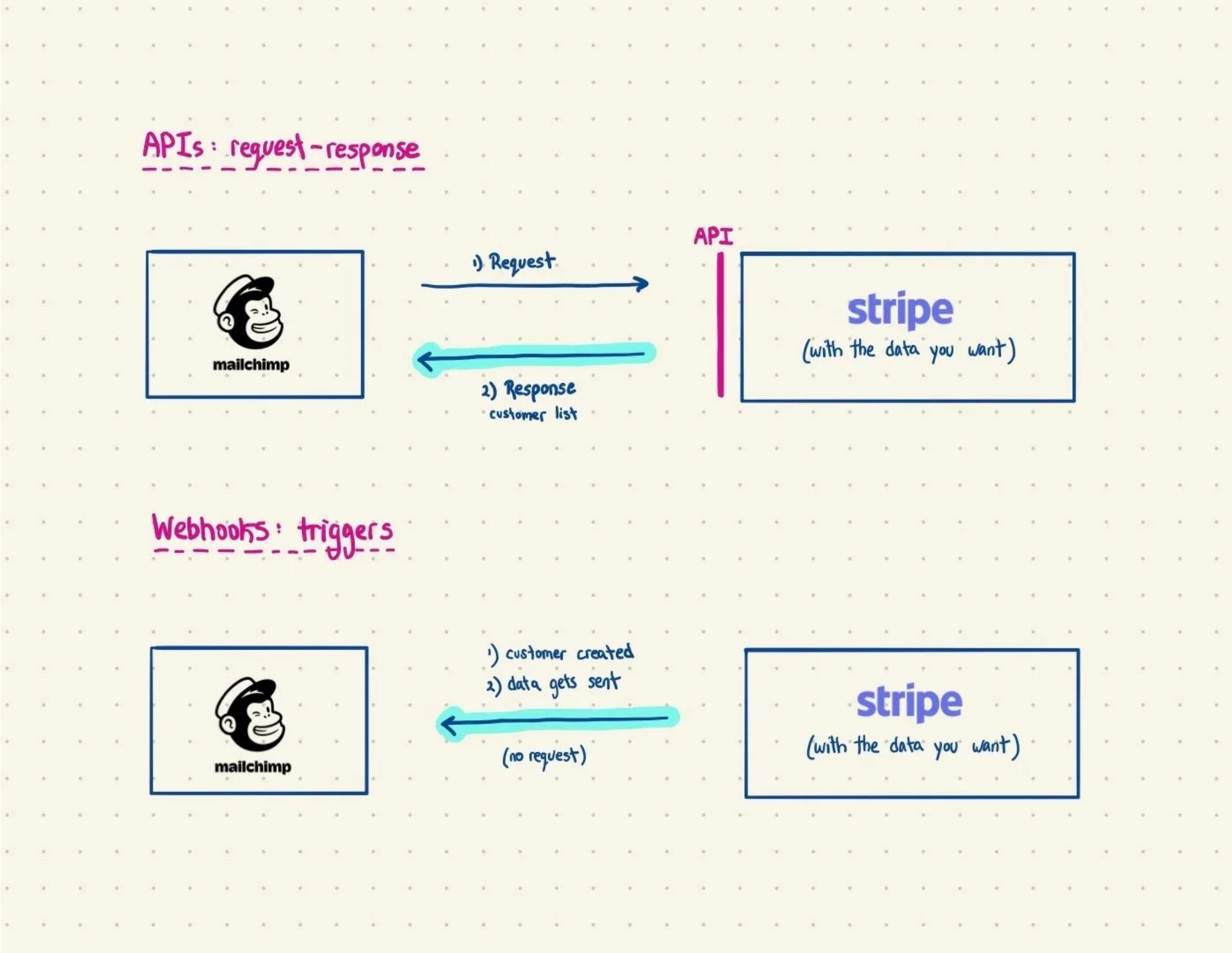
A semi-realistic Webhook example

This is all very theoretical so far, which is not what you're here for. In practice, a webhook is usually **a URL** (i.e. an endpoint on a server) that you paste into a settings field, or add into your app manually. To understand *what that means*, let's take a look at Stripe.

Stripe is a payment processor - businesses use them to process credit cards, manage subscriptions, and handle their customer data. Stripe operates mostly via their APIs - they have endpoints for charging customers, creating new customers, etc.

For our example, let's say we're Madhappy, and we're running an e-commerce store built on top of Stripe (we're not, it's on Shopify, but ignore that). Every time someone orders their first item, we want to add them to our Local Optimist newsletter. How would we do that?

Stripe has an API endpoint for listing all existing customers. We *could* repeatedly hit that endpoint, and every time a new customer is on the list, send their email address over to our newsletter system. But then we'd need to set up some sort of recurring request, check differences between lists, deal with pagination - a lot of overhead. Conveniently though, Stripe offers webhooks, so we can have Stripe *automatically* send an event to our newsletter system every time a new customer gets created. Neat.



This is exactly the difference we talked about above - APIs (like listing customers) are passive and require *our* application to take an action, while webhooks (send an event when a customer gets created) are active and *initiate* action themselves.

> 🚨 **Confusion Alert** 🚨
>
> Not every application (including your own, if you're building one) supports webhooks, and even if they do, they might not support webhooks from the application that you need them to. Popular newsletter providers like Mailchimp support webhooks out of the box.
>
> 🚨 **Confusion Alert** 🚨

In general, webhooks tend to get used for **quick, simple integrations** like our newsletter example; if the task is as simple as sending basic data (like an email address) from one place to another, webhooks are a good way to do that. But if you need a more involved integration, like with two way communication, that's typically going to require more custom work.

Webhooks mechanics

Like APIs, webhooks usually send data in a JSON format. In our Stripe example, the webhook we set up might send us something like this when a customer gets created:

```
{
  event: 'customer.created',
  email: 'justin@technically.dev',
  created_at: '2020-09-15',
  subscription_type: 'monthly'
}
```

If the app you're using (like our newsletter system) supports webhooks, you'll probably see a section of the settings page dedicated to them. You'll need to get a **URL** where the app will *receive* data, and then put that URL into the application that's actually sending the webhook in the first place.

If you're consuming webhooks in your own custom app, the logic is the same - you'll need to build an endpoint to consume events, and pass that endpoint into the application producing the webhooks (like Stripe).

Further reading

- I wrote a more technical guide to best practices for building webhooks into your app for the WorkOS blog
- Zapier's deep dive into how Webhooks work
- Test Webhooks yourself with webhook.site

5 Comments

Write a comment…

Brendan Keeler Writes Health API Guy Sep 13, 2021 ♥ Liked by Justin

Also worth calling out callbacks, which are just webhooks that aren't set out-of-band (i.e. plugging a URL into a settings page) but are set via a RESTful API call.

♡ 4 Reply Gift a subscription Collapse •••

Nick Byrd Sep 14, 2021 ♥ Liked by Justin

Can I use a webhook to connect a substack newsletter to a Discord? I wanted to set it up such that when I release a new substack newsletter, an annoucnement gets pushed into a channel in my Discord server.

Any help would be greatly appreciated! Thanks!

♡ 2 Reply Gift a subscription Collapse •••

1 reply by Justin

3 more comments...