



Get Hooked on Webhooks

Tim Dietrich
NetSuite / SuiteScript / Xojo Developer

About Me

- Independent developer.
- Focus is on integration.
- Developing with Xojo since late 2014.
- This is my third XDC.

Session Goals

- Explain what webhooks are, and why they are so popular.
- Explain the differences between webhooks and APIs.
- Show how to develop Xojo applications that receive webhook notifications.
- Show how to develop Xojo applications that send webhook notifications.
- Discover best practices for implementing webhooks.

What are Webhooks?

What are Webhooks?



- "Push notifications for your API"
- Chris Maddern on Twitter: @chrismaddern
- Webhooks are callbacks that are triggered when an event occurs on a source system.
- The source system (the server / sender) sends an HTTP request to a receiving system (the client / receiver).

Why are Webhooks so popular?



Why Webhooks are Popular



- Easy to use and configure
- Language agnostic
- Efficient (takes load off of the client and server)
- Instant

How Webhooks Work



How Webhooks Work

Client

Event!
Server

How Webhooks Work

Client

Data

Server

How Webhooks Work

Client

"Got it!"

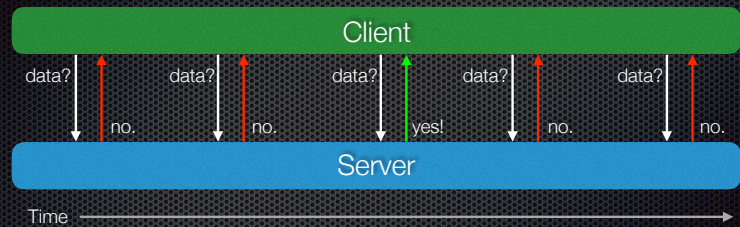
Server

Webhooks vs. APIs

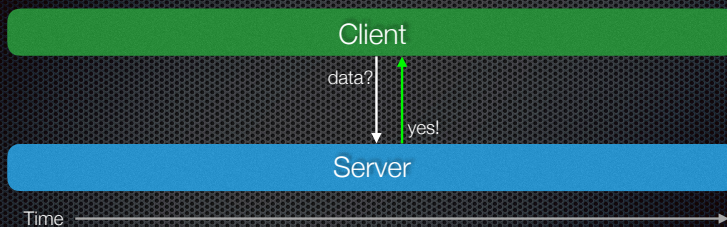
Webhooks vs. APIs

- Webhooks are often referred to as "reverse APIs."
- API: Client sends request, API responds.
- Webhook: API sends data.
- No need for a client to "poll" a server.
- "Don't call us, we'll call you!"

API



Webhook



Popular Services that Support Webhooks

Webhook Services

- Google Sheets
- Twitter
- GMail
- Slack
- Mailchimp
- Airtable
- Instagram
- Dropbox
- Stripe
- WordPress
- Wufoo
- Shopify
- Twilio
- WooCommerce
- And more...



Demo 1: A Xojo-Based Webhook Listener



Demo 2: A Xojo-Based Webhook Server



Best Practices



Best Practices: Servers



- Send complete Webhook notifications.
- Support multiple URLs.
- Support multiple events.
- Provide subscribers with both a UI and an API for specifying listener URLs.
- Secure webhooks using a token.
- Support retries.
- Send email notifications when the Webhook fails.
- Consider providing an "Events" API.
- Include a timestamp and ID in the notifications.
- Provide a way to send a test notification.

Best Practices: Clients



- Validate Webhook notifications.
- Consider using a queue.
- Use HTTPS.
- Have a plan for dealing with missed notifications.

Q & A



Tim Dietrich

Email: timdietrich@me.com

Web: <https://timdietrich.me/>

Give us feedback on this session in the XDC app!