Working with Webhooks

Lorna Mitchell, IBM
PHPUK, February 2018

What is a Webhook?

An HTTP POST request.

Webhooks in the Wild

Slack Integrations

GitHub (ibm-cds-labs/soingest) APP 17:52

[soingest:travis] 1 new commit by Lorna Jane Mitchell:

> `02b195b` Change the deployment to only run on master, ready for PR - Lorna Jane Mitchell

GitHub (ibm-cds-labs/soingest) APP 17:57

[ibm-watson-data-lab/soingest] Pull request submitted by lornajane

> **#21 Move to Travis for deployment**
>
> (this change also switches the notifications back to slack while hubot is under maintenance)

GitHub Builds

All checks have failed

1 failing check

continuous-integration/travis-ci/pr — The Travis CI build failed

This branch has no conflicts with the base branch

Only those with write access to this repository can merge pull requests.

Fun with Zapier

How APIs Work

How APIs Work

How APIs Work

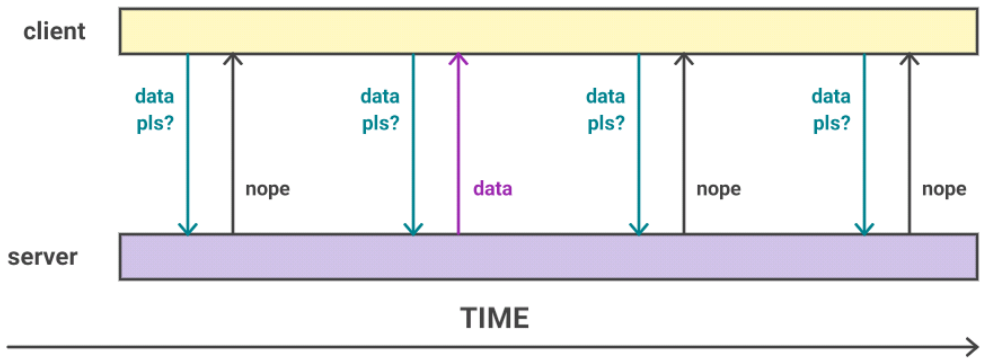
How Webhooks Work

How Webhooks Work

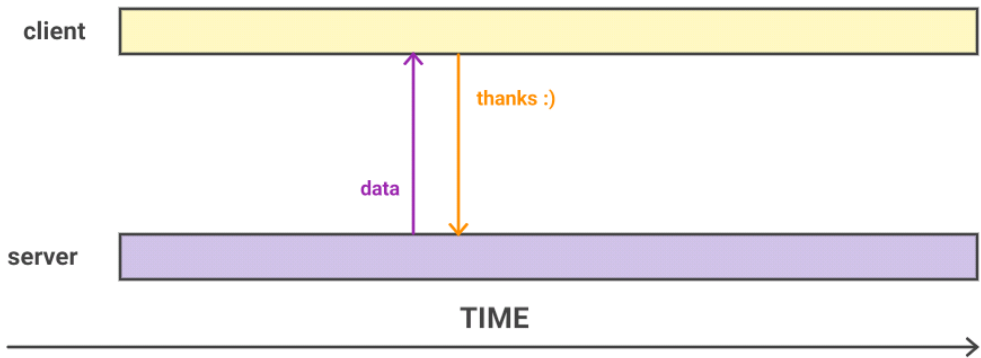
How Webhooks Work

What About Time?

APIs Over Time



Webhooks Over Time



Webhook Payloads: GitHub Push

```
"ref": "refs/heads/master",
"before": "1ae6a404351cead52df24893621d82ba6ec84a1c",
"after": "e8474d83985330fa36f8862b37ca84ada4313392",
"created": false,
"deleted": false,
"forced": false,
"compare": "https://github.com/lornajane/demo/compare/1ae6a404351c...e847
"commits": [ ... ],
"repository": { ... },
"pusher": { ... },
"sender": { ... }
```

Webhook Payloads

Consider the use cases:

- try to include all information for common outcomes
- consider impact of payload size vs potentially many followup API calls
- keep data formats simple

Webhook Security

When working with webhooks:

- be aware of attack vectors
- always use SSL
- consider shared secrets for HMAC
- all good HTTP security practices apply

Publishing Webhooks

Publishing Webhooks

Offering webhook integrations is ideal if:

- you have clients polling your API a lot
- it's common for another system to react to changes in your system
- you want to offer notifications for specific events

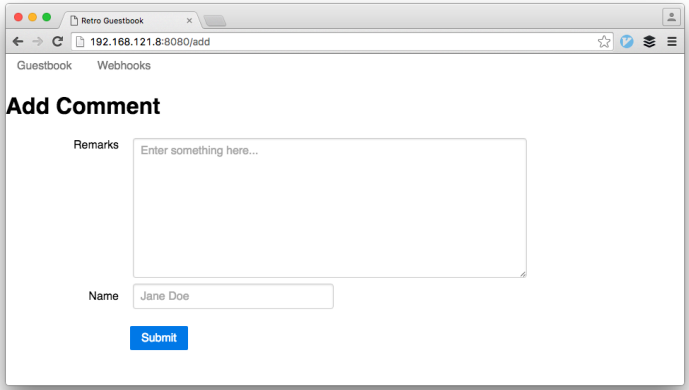
Example App: Retro Guestbook

In the olden days, we had guestbooks on our websites.

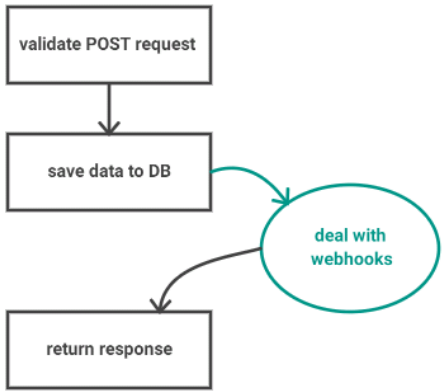
My example app is a guestbook that:

- allows a user to leave their name and a comment
- shows the comments left so far
- supports webhook notification of new comments by allowing endpoints to be registered

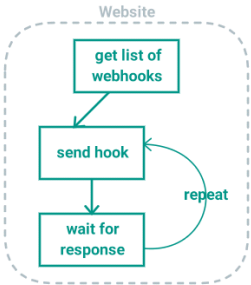
Example App: Retro Guestbook



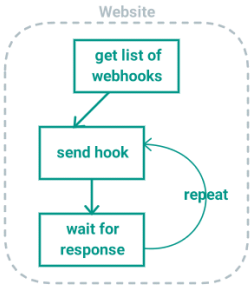
Saving Data: Basic Process



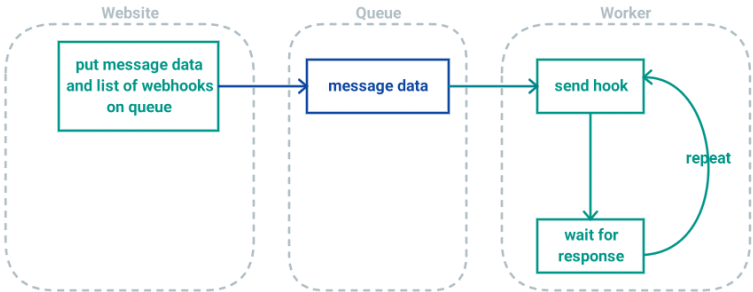
Saving Data: Handling Webhooks



Saving Data: Handling Webhooks



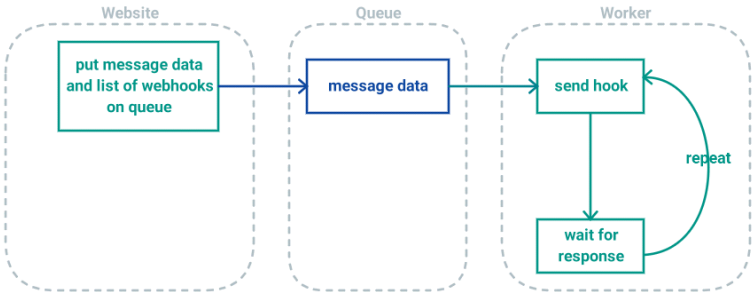
Saving Data: Handling Webhooks



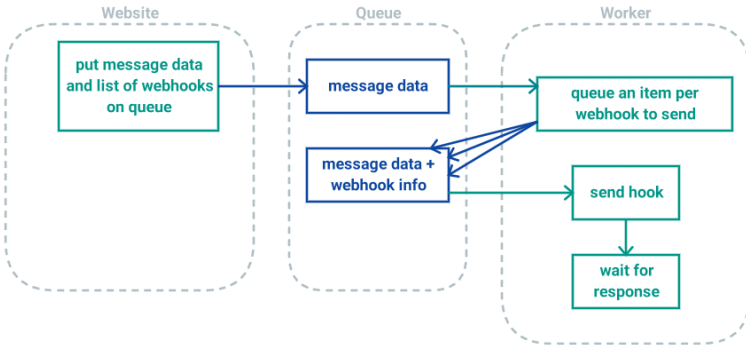
Saving Data: Handling Webhooks

```
1     $comment['name'] = filter_var($data['name'], FILTER_SANITIZE_STRIN
2     $comment['comment'] = filter_var($data['comment'], FILTER_SANITIZE
3     $comment['time'] = time();
4     // write comment to CouchDB...
5     // get the list of webhooks to notify from CouchDB...
6
7     // write comments and webhooks to queue
8     $channel = $this->rabbitmq_handle->channel();
9     $msg = new \PhpAmqpLib\Message\AMQPMessage(
10        json_encode(["comment" => $comment, "webhooks" => $webhooks]),
11    $channel->basic_publish($msg, '', 'comments');
```

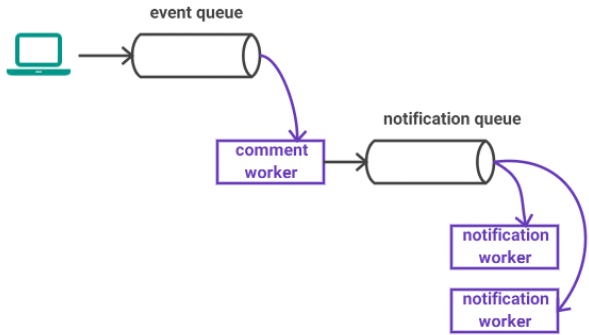
Saving Data: Handling Webhooks



Saving Data: Handling Webhooks



Saving Data: Handling Webhooks



Example: Publishing Webhooks

(includes excellent endpoint testing tool: http://requestb.in)

Receiving Webhooks

Receiving Webhooks

It's just a POST request! Advice:

- DO: accept, store and acknowledge quickly
- DON'T: validate or process before acknowledging

Serverless Webhook Endpoints

Serverless technology:

- Functions as a Service
- Scalable: ideal for bursty workloads
- Pay-as-you-go, and with free tiers
- PHP supported on some platforms (they all support NodeJS)

Serverless PHP Webhook Catcher

```
1  function main(array $params) : array {
2      $db_url = $params['cloudantURL'];
3      $incoming_body = base64_decode($params['__ow_body']);
4      $data = json_decode($incoming_body, true);
5
6      echo "Saving data ...\n";
7      $server = new \PHPCouchDB\Server(["url" => $db_url]);
8      $db = $server->useDb(["name" => "incoming"]);
9      $meta = ["received" => time(), "status" => "new"];
10     $db->create(["data" => $data, "meta" => $meta]);
11     return ["body" => "Thanks :)"];
```

Example: Receiving a Webhook

Ngrok for Testing Webhooks

https://ngrok.com/ - secure tunnel to your dev platform

Use this tool to:

- webhook into code running locally
- inspect the request and response of the webhook
- replay requests and see the responses

Webhooks

... are awesome :)

Webhooks in Your Applications

- Use them WHEN you want to notify other systems
- Examples of HOW to use webhooks hopefully gave you some ideas
- Webhooks are HTTP: we already understand this

Thanks!

- Feedback please! https://joind.in/
- IBM Cloud: https://www.ibm.com/cloud/
- Requestbin: http://requestb.in
- Ngrok: https://ngrok.com/
- PHP Web Services from O'Reilly
- Example app: https://github.com/ibm-watson-data-lab/guestbook
- PHP/CouchDB: https://github.com/ibm-watson-data-lab/php-couchdb