



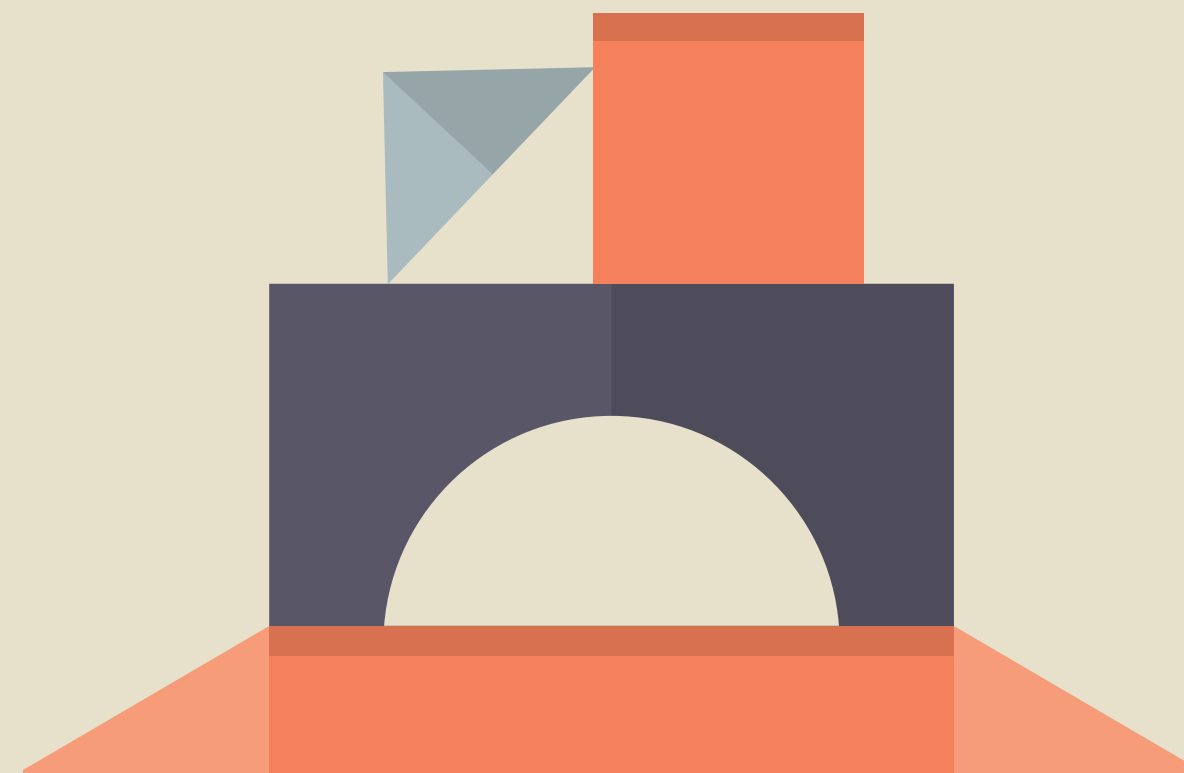
Scaling up tech and product from
0 to 3+ million users at Postman

About Me



**Founder of
Postman**

+



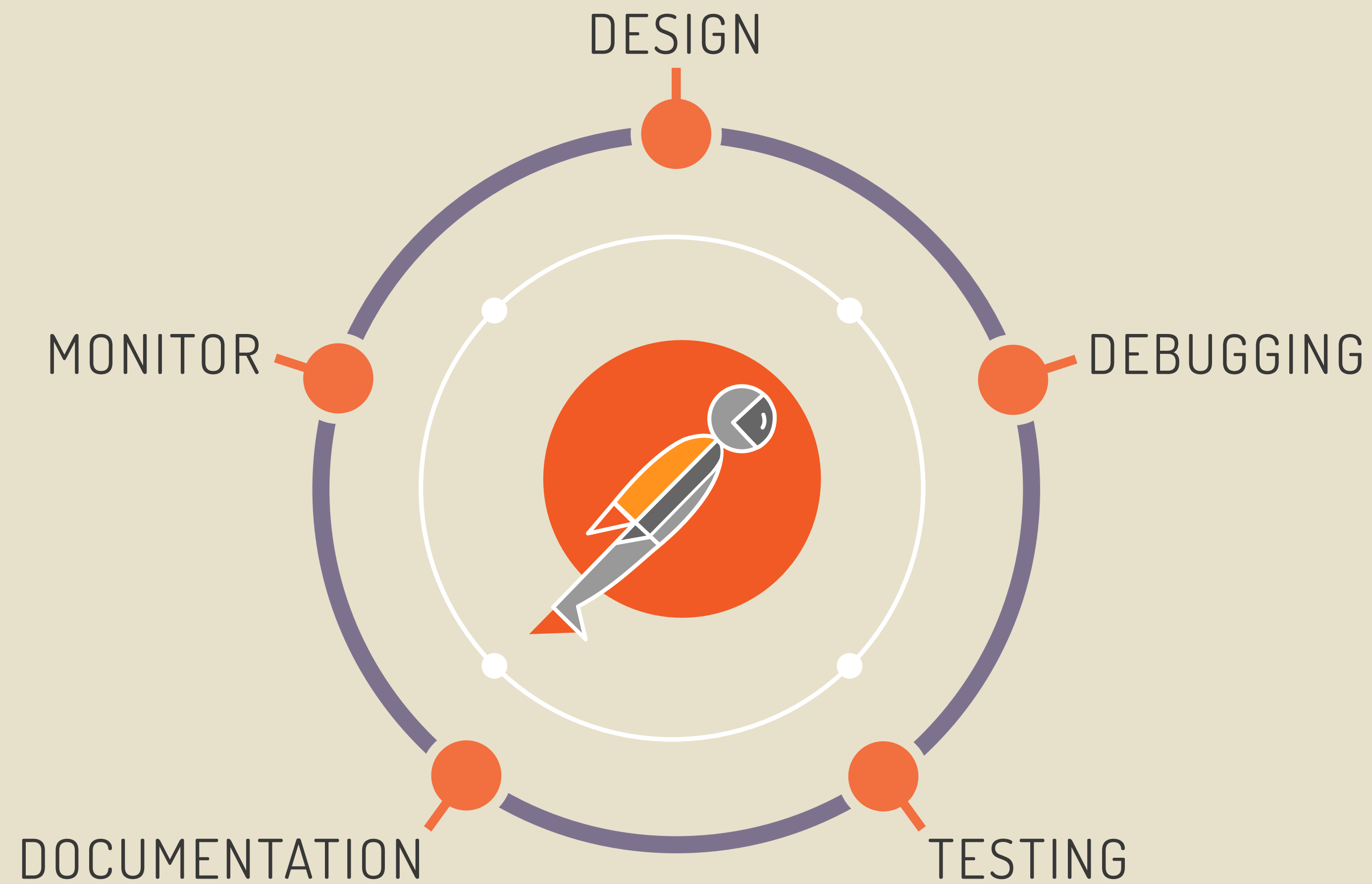
**A Builder
Since 2012**

+



**Passionate
about APIs**

Postman is solving for the API workflow



2012 - Legacy app

POSTMAN

History

Collections

Nothing in your history yet. You can automatically save and access your sent requests here.

Normal

Basic Auth

Digest Auth

OAuth 1.0

👁

No environment

⌨

🔧

0

http://httpbin.org/post

POST

📝 URL params

📝 Headers (0)

form-data

x-www-form-urlencoded

raw

foo

value_foo

Text

✕

bar

value_bar

Text

✕

Key

Value

Text

Send

Preview

Add to collection

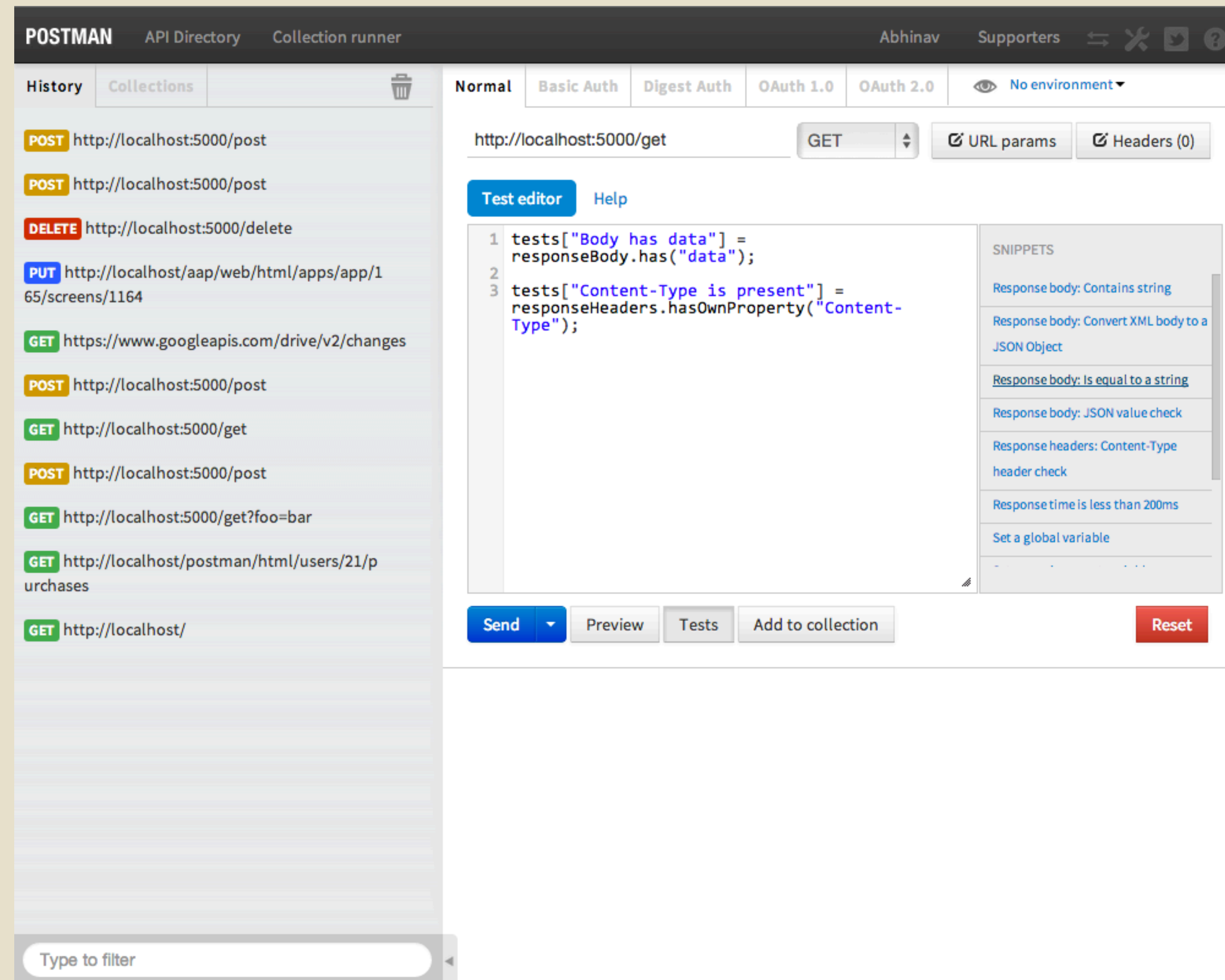
Reset



- Legacy features: **JSON parsing, JSON editing, Authorization flows, Organize your API requests in Collections**
- New Chrome app platform comes along!
 - New features: **OAuth 2, Sharing links, User accounts**
 - Paid feature: Jetpacks (**Testing sandbox**)

2013 - Chrome app

- Lots of refactoring = Lots of files split across folders. **jQuery and plugins**
- Adopted **Backbone.js**. Decided against using AngularJS or any other fancy framework - Didn't understand it.
- Legacy app between **100-150K users**.



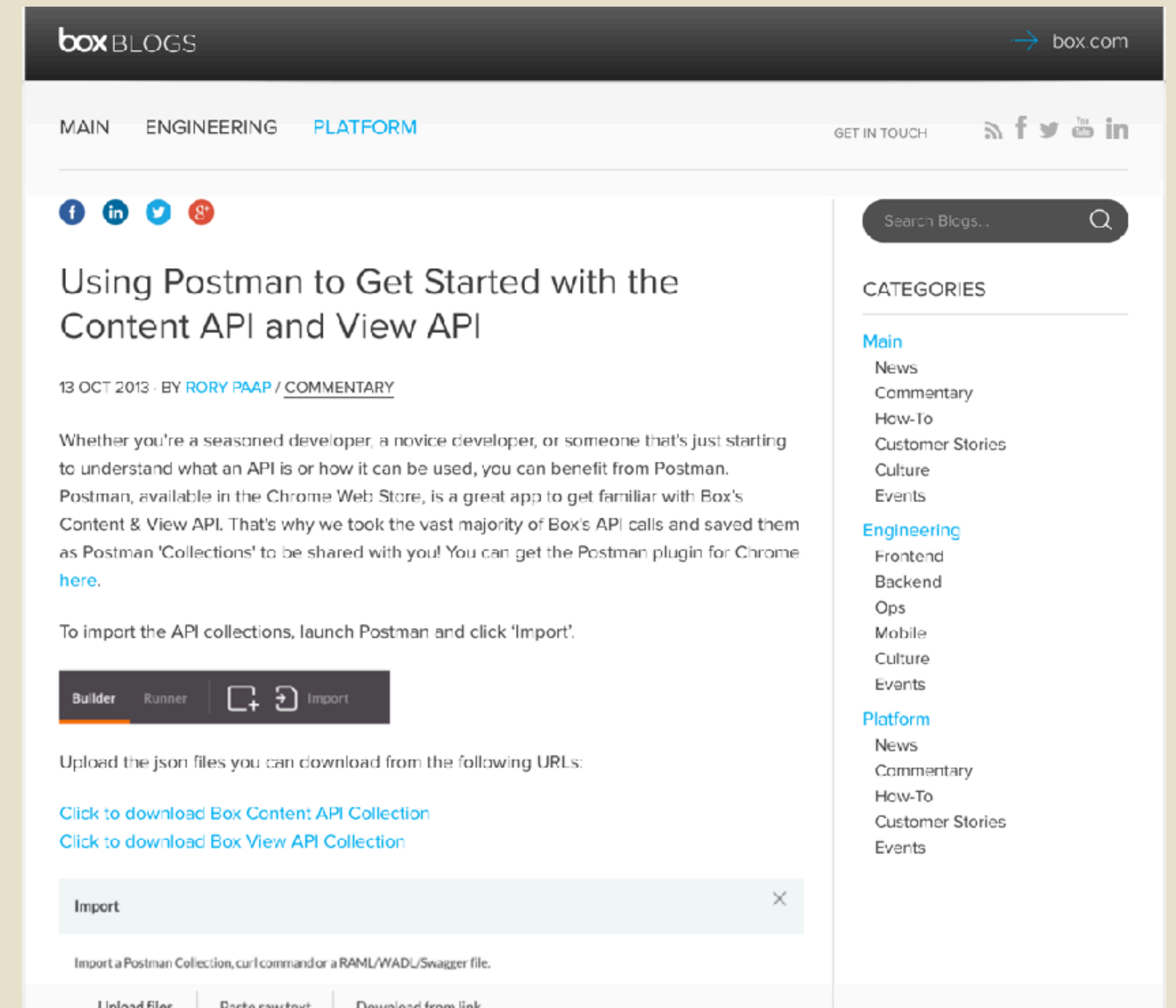
2013 - Chrome app

- Postman Interceptor - **vanilla Javascript**
- Also had a website: getpostman.com -
- **Linode** and **PHP** using the **Slim Framework, MySQL**



2014

- Newman for build system integration - **Node.js**
- **Box** publishes a blog post talking about **Collections** and how it's super easy to share API calls.
- **500K users** on Chrome Web Store - both **legacy + Chrome app**





Got a tip? [Let us know.](#)

[News](#) ▾ [Video](#) ▾ [Events](#) ▾ [Crunchbase](#)

Follow Us



Message Us

[Search](#)



10TH ANNUAL CRUNCHIES AWARDS Celebrate The Best Startups At TechCrunch's Annual Awards Show This February [Get Your Tickets Now](#) ▶

postman

Nexus Venture Partners

Asia

Popular Posts

Postman, An API Development Platform Used By Box And Others, Raises \$1M

Posted May 26, 2015 by [Jon Russell \(@jonrussell\)](#)



- Let's build a collaboration platform and solve for the API workflow!
- Wait...let's make it real-time and solve account syncing first.
- Sails.js based MVP for syncing account data



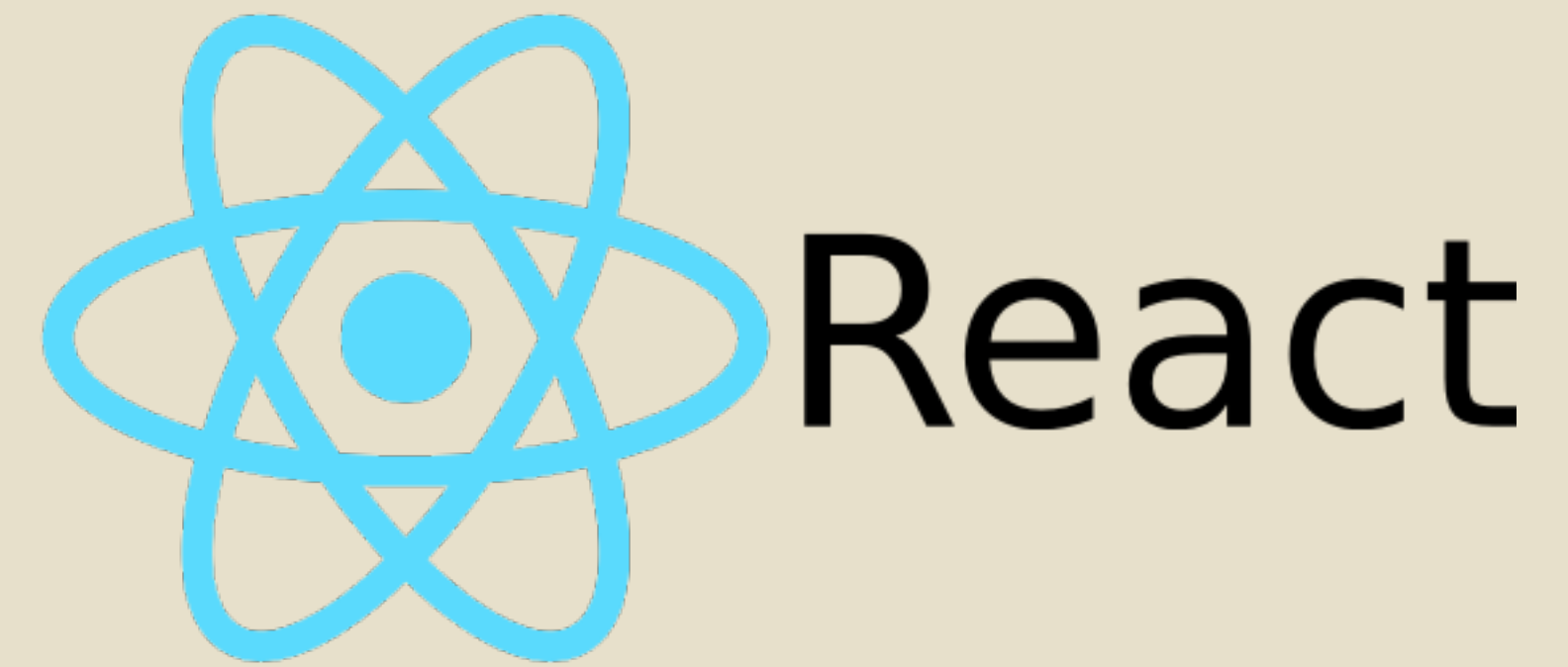
2015

- Users start signing up in thousands per day
- Scale up Sync!
 - Move to Amazon Web Services
 - Start fixing the platform parity problem
 - Need to improve the user experience for the app



2015

- Start refactoring the app (again)
- React.js
- Individual components
- But: people want native apps. They don't love the Chrome app so much. Also, Chrome is not evolving. Let's build a native app.



Filter

History Collections

All Me Team

Newman API 36 requests

Weather to Slack Demo Test 3 requests

Postman Products [Monitor] 10 requests

PowerBI API 10 requests

Postman Pro API 17 requests

Weather to Slack Demo Monitor 3 requests

Demo Collection 2 requests

Test collection 2 requests

Postman Website: Demo 3 requests

Publishing test 1 request

JSONBlob Core API 4 requests

JSONBlob Core API [Published] 4 requests

Demo - Chaining #demo 2 requests

https://echo.getpostman.com/get?test=123

GET https://echo.getpostman.com/get?test=123 Params Send Save

Status: 200 OK Time: 1012 ms Size: 820 B

Auth Headers Body Pre-req. Tests Cookies Code

Type No Auth

Body Cookies (1) Headers (14) Tests

Pretty Raw Preview JSON

```
1 {
2   "args": {
3     "test": "123"
4   },
5   "headers": {
6     "host": "echo.getpostman.com",
7     "accept": "*/*",
8     "accept-encoding": "gzip, deflate",
9     "cookie": "sails.sid=s%3AfcJbZd5CPGAieQw3LvYu0xPQNG_rnoRI
10      .op0IPwwzA5XLuqs%2BkMR3v%2BnWpgJwVEe1e33mG9NfDsM",
11     "user-agent": "PostmanRuntime/3.0.9",
12     "x-forwarded-port": "443",
13     "x-forwarded-proto": "https"
14   },
15   "url": "https://echo.getpostman.com/get?test=123"
16 }
```

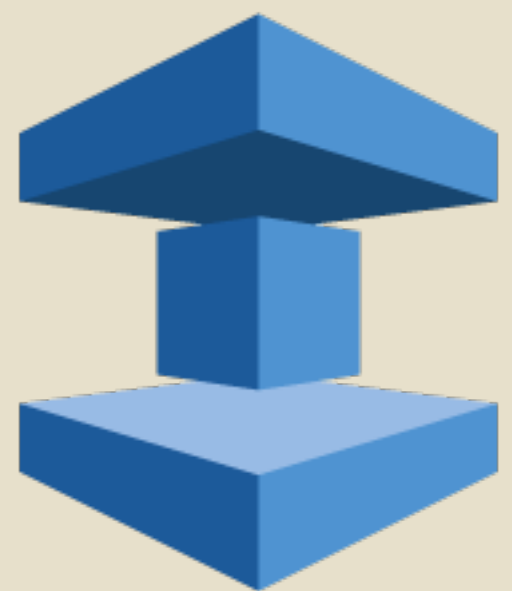
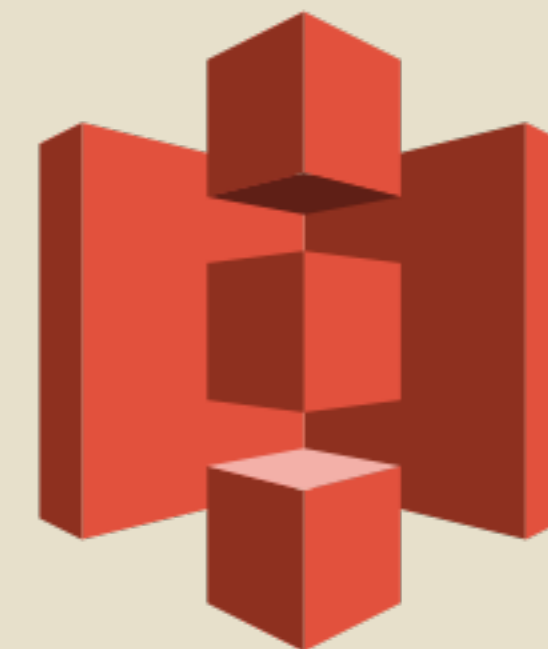
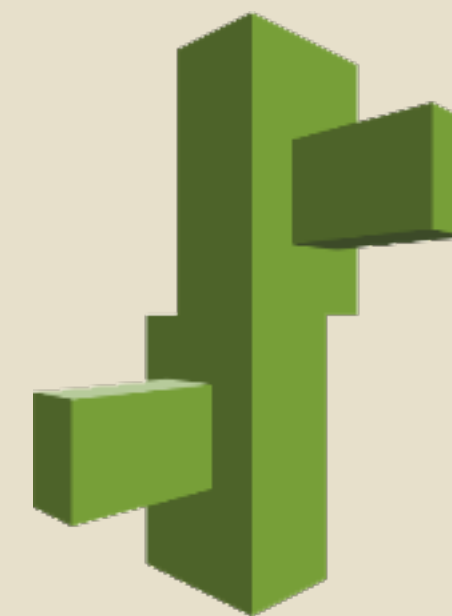
2016

- Apps on 4 different environments: **Windows, MacOS, Linux, Chrome Apps**
 - **React + Electron + Node.js** (Native apps managed through our own release server)
- Command line: **Node.js**
- Server side
 - **PHP on Linode**
 - **Node.js + Sails.js on AWS**

2016

- Product complexity has been increasing with every launch
- PHP was being a drag on the development team. Lots of APIs started accumulating in the PHP code base. Moved everything to **AWS**.
- Started building new features (documenter) separately. **Node.js + Sails.js**
- Launched **monitors** with Postman Pro - open sourced **Postman Runtime**
- Launched the **Postman Pro API**

- Other parts of the stack
 - **ElasticSearch** for analytics
 - **PowerBI** for business dashboards
- AWS usage has gone up significantly
 - **Elastic Beanstack, S3, Amazon RDS,**
 - **Elastic Load Balancing, ElastiCache**
 - **AWS Virtual Private Cloud**
 - **Lambda** triggers in some cases



2017 and beyond

- Tons of new features coming up for Postman and Postman Pro
- How to maintain complexity while scaling up? We use **Postman** for that.
- **Conway's law** is a huge problem: Issues come up at the seams. People don't want to go into codebases of others.
- Refactoring and testing needs to be built into product timelines

Lessons learnt

- Use only what you understand
- Your tech stack is for your users
- You will have to re-write it anyway
- Tech validation only happens after product validation
- Remember Conway's law

THANK YOU!

@a85

Q & A

