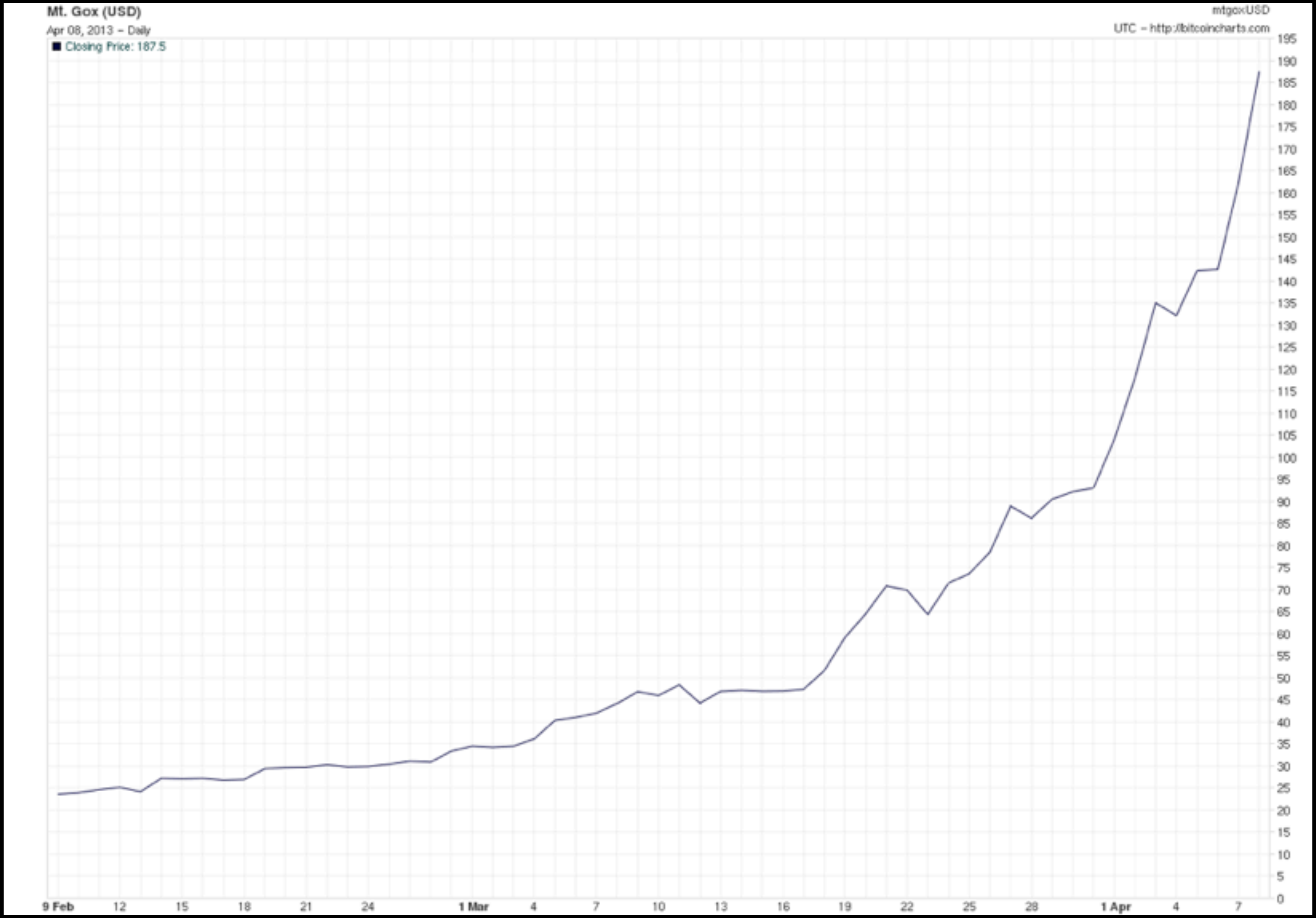
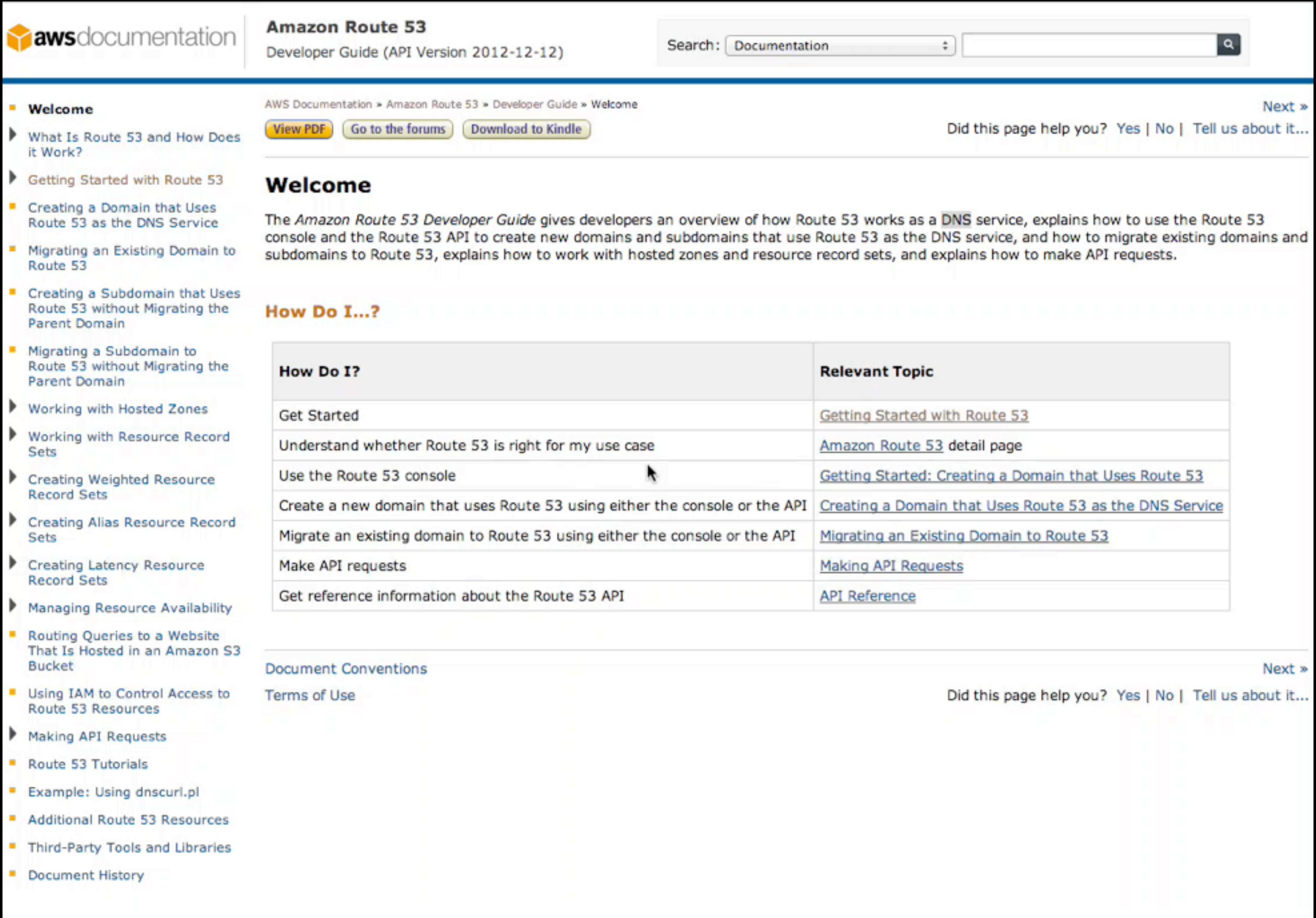
Single Page Docs

@BrandonPhilips
brandon@ifup.org


Mt. Gox (USD)
Apr 08, 2013 – Daily
■ Closing Price: 187.5
mtgoxUSD
UTC – http://bitcoincharts.com
195
190
185
180
175
170
165
160
155
150
145
140
135
130
125
120
115
110
105
100
95
90
85
80
75
70
65
60
55
50
45
40
35
30
25
20
15
10
5
0
9 Feb
12
15
18
21
24
1 Mar
4
7
10
13
16
19
22
25
28
1 Apr
4
7

Doc Anti-Patterns



Amazon Route 53

Developer Guide (API Version 2012-12-12)

AWS Documentation » Amazon Route 53 » Developer Guide » Welcome

Welcome

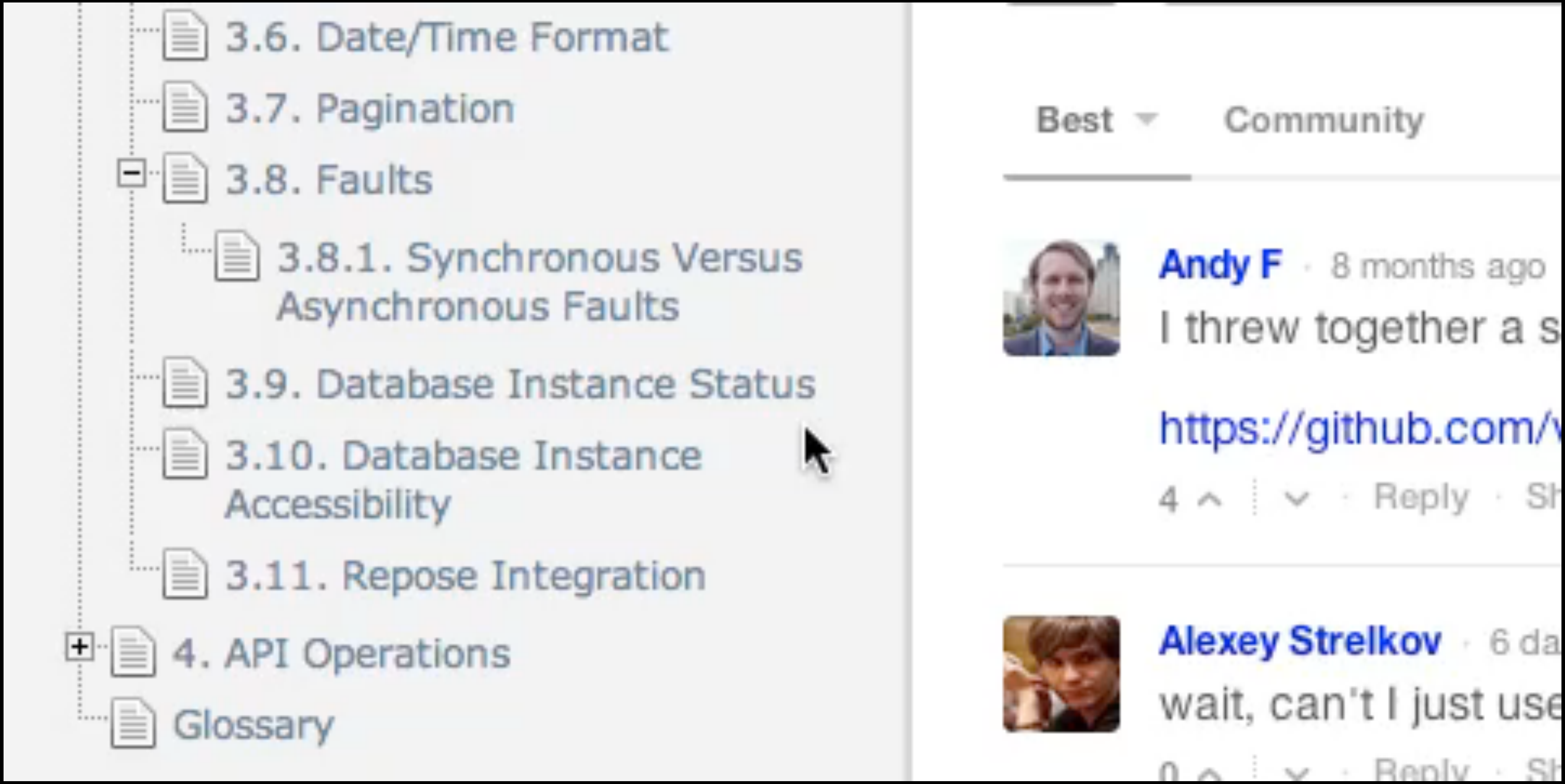
The *Amazon Route 53 Developer Guide* gives developers an overview of how Route 53 works as a DNS service, explains how to use the Route 53 console and the Route 53 API to create new domains and subdomains that use Route 53 as the DNS service, and how to migrate existing domains and subdomains to Route 53, explains how to work with hosted zones and resource record sets, and explains how to make API requests.

How Do I...?

How Do I?	Relevant Topic
Get Started	Getting Started with Route 53
Understand whether Route 53 is right for my use case	Amazon Route 53 detail page
Use the Route 53 console	Getting Started: Creating a Domain that Uses Route 53
Create a new domain that uses Route 53 using either the console or the API	Creating a Domain that Uses Route 53 as the DNS Service
Migrate an existing domain to Route 53 using either the console or the API	Migrating an Existing Domain to Route 53
Make API requests	Making API Requests
Get reference information about the Route 53 API	API Reference

Document Conventions

Terms of Use



3.6. Date/Time Format

3.7. Pagination

3.8. Faults

3.8.1. Synchronous Versus Asynchronous Faults

3.9. Database Instance Status

3.10. Database Instance Accessibility

3.11. Repose Integration

4. API Operations

Glossary

Andy F · 8 months ago

I threw together a s

https://github.com/

Alexey Strelkov · 6 da

wait, can't I just use

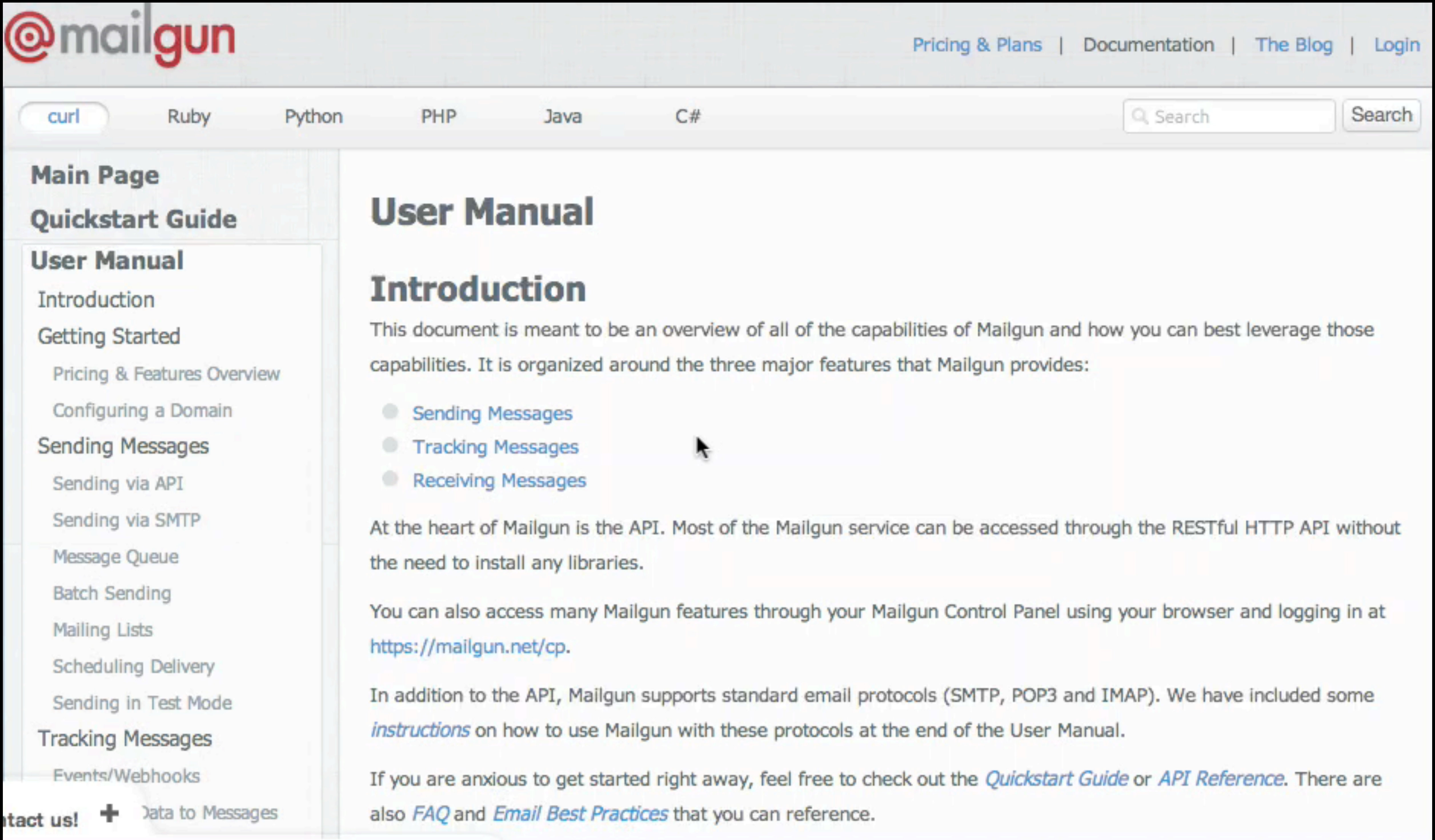
Single Page Patterns

- Single Page

- Single Page
- Persistent Navigation

- Single Page
- Persistent Navigation
- Dynamic Orientation

Single Page



User Manual

Introduction

This document is meant to be an overview of all of the capabilities of Mailgun and how you can best leverage those capabilities. It is organized around the three major features that Mailgun provides:

- Sending Messages
- Tracking Messages
- Receiving Messages

At the heart of Mailgun is the API. Most of the Mailgun service can be accessed through the RESTful HTTP API without the need to install any libraries.

You can also access many Mailgun features through your Mailgun Control Panel using your browser and logging in at https://mailgun.net/cp.

In addition to the API, Mailgun supports standard email protocols (SMTP, POP3 and IMAP). We have included some *instructions* on how to use Mailgun with these protocols at the end of the User Manual.

If you are anxious to get started right away, feel free to check out the *Quickstart Guide* or *API Reference*. There are also *FAQ* and *Email Best Practices* that you can reference.

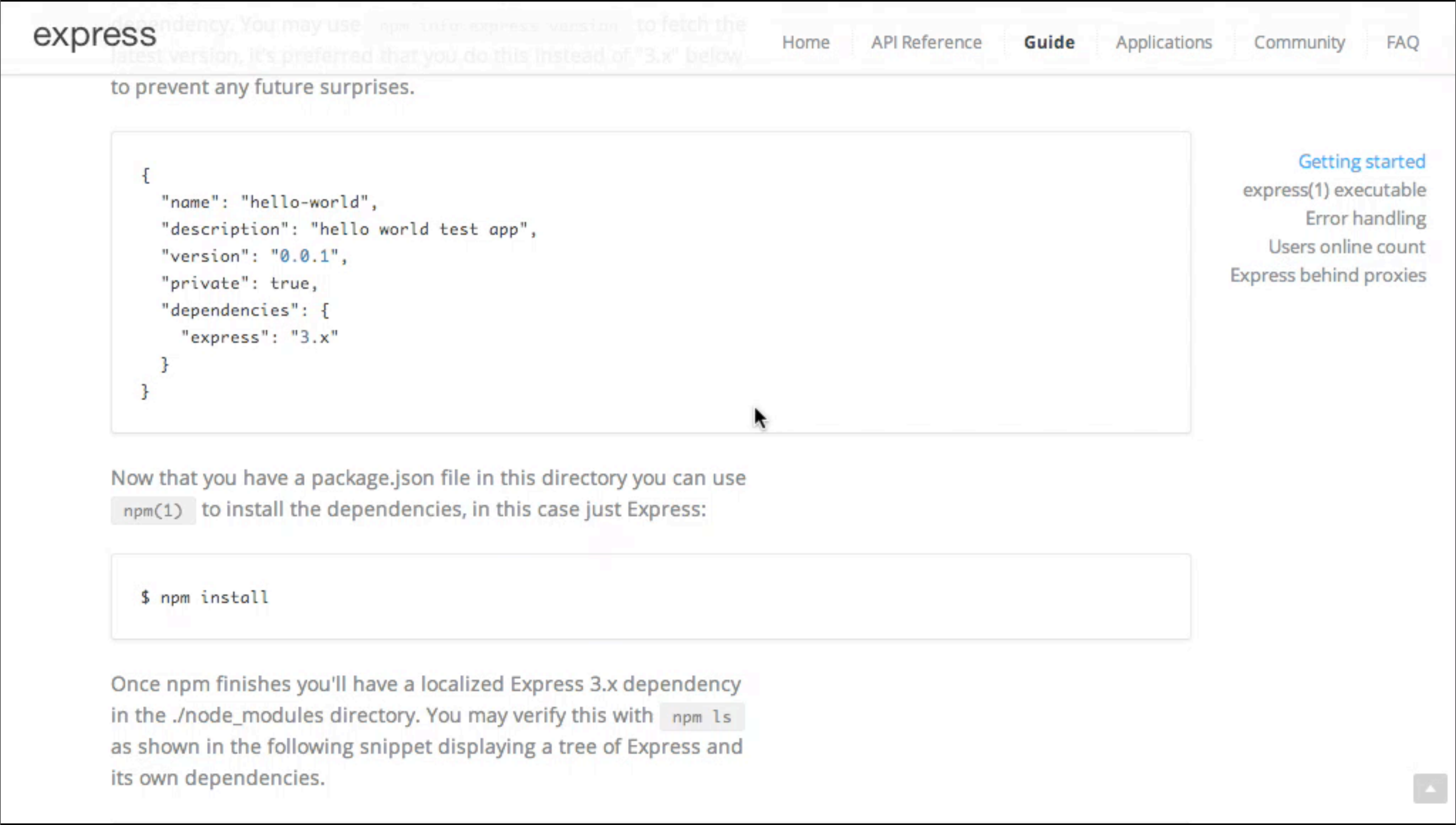
- Scrolling is natural

- Scrolling is natural
- Search is free in browsers

- Forces beginning, middle and end thinking

- Forces beginning, middle and end thinking
- Fewer and simpler editor windows

Persistent Navigation



dependency. You may use `npm info express version` to fetch the latest version, it's preferred that you do this instead of "3.x" below to prevent any future surprises.

```
{
  "name": "hello-world",
  "description": "hello world test app",
  "version": "0.0.1",
  "private": true,
  "dependencies": {
    "express": "3.x"
  }
}
```

Now that you have a package.json file in this directory you can use `npm(1)` to install the dependencies, in this case just Express:

```
$ npm install
```

Once npm finishes you'll have a localized Express 3.x dependency in the ./node_modules directory. You may verify this with `npm ls` as shown in the following snippet displaying a tree of Express and its own dependencies.

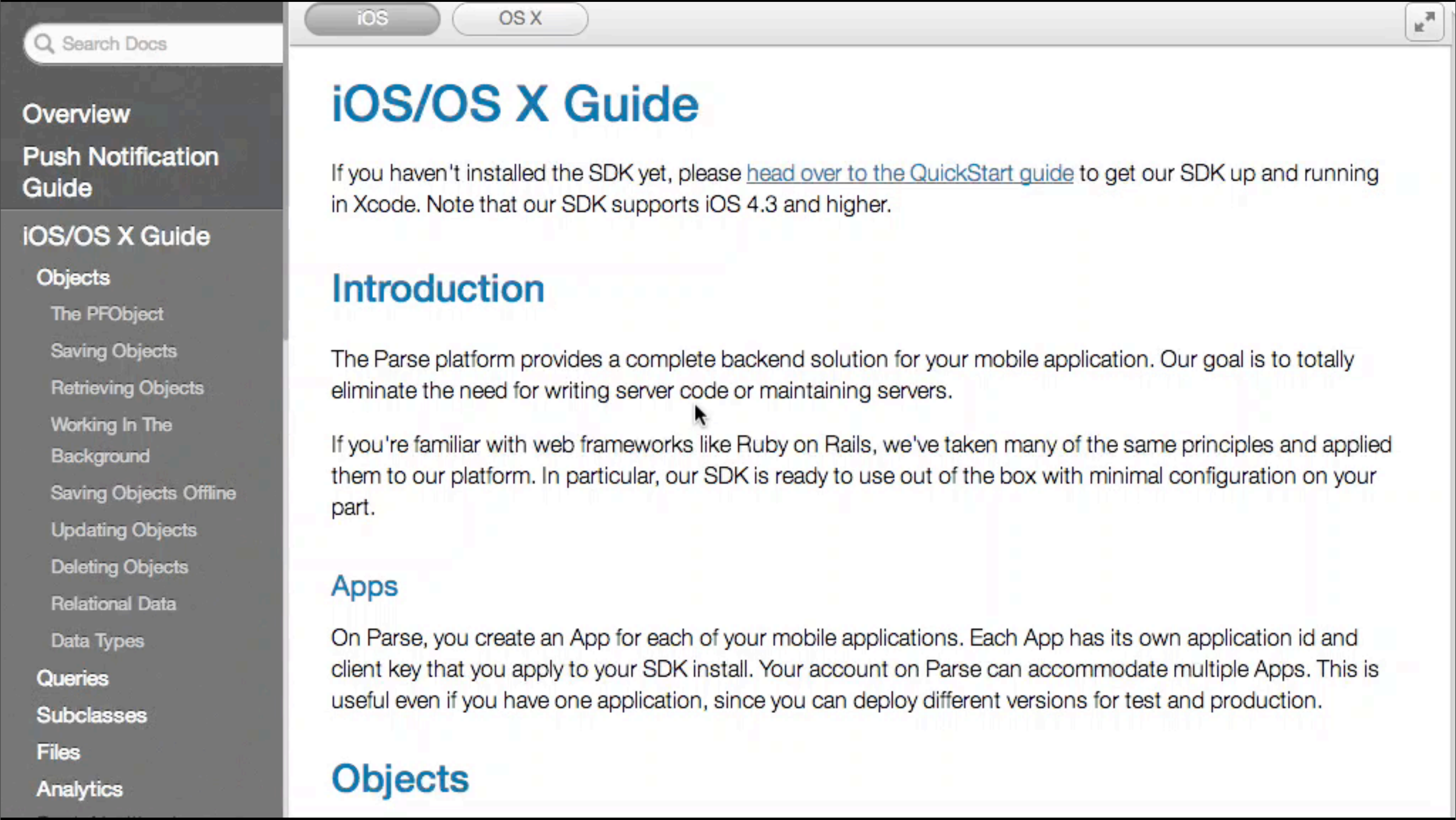
- Reduces frustration of skipping ahead

- Reduces frustration of skipping ahead
- Feels natural to quickly find TOC

- Constantly reminded of section bloat

- Constantly reminded of section bloat
- Reduce length of section names

Dynamic Orientation



iOS/OS X Guide

If you haven't installed the SDK yet, please head over to the QuickStart guide to get our SDK up and running in Xcode. Note that our SDK supports iOS 4.3 and higher.

Introduction

The Parse platform provides a complete backend solution for your mobile application. Our goal is to totally eliminate the need for writing server code or maintaining servers.

If you're familiar with web frameworks like Ruby on Rails, we've taken many of the same principles and applied them to our platform. In particular, our SDK is ready to use out of the box with minimal configuration on your part.

Apps

On Parse, you create an App for each of your mobile applications. Each App has its own application id and client key that you apply to your SDK install. Your account on Parse can accommodate multiple Apps. This is useful even if you have one application, since you can deploy different versions for test and production.

Objects

- How much more?

- How much more?
- What did I read?

- How much more?
- What did I read?
- Should I skip ahead?

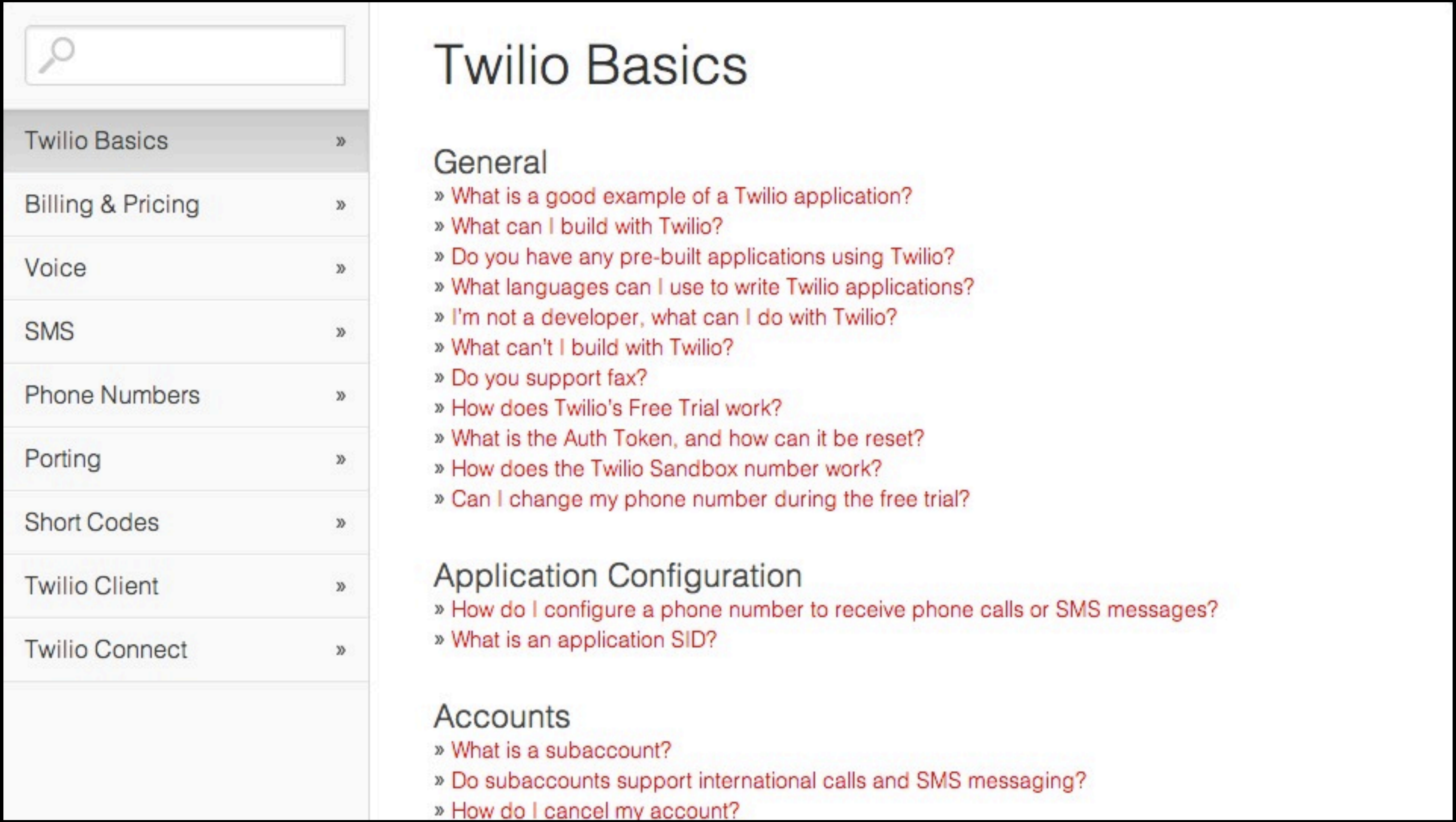
Terminology

Fixie Docs

Building a Fixie

Know the Downsides

The Right Match



Twilio Basics

General

» What is a good example of a Twilio application?
» What can I build with Twilio?
» Do you have any pre-built applications using Twilio?
» What languages can I use to write Twilio applications?
» I'm not a developer, what can I do with Twilio?
» What can't I build with Twilio?
» Do you support fax?
» How does Twilio's Free Trial work?
» What is the Auth Token, and how can it be reset?
» How does the Twilio Sandbox number work?
» Can I change my phone number during the free trial?

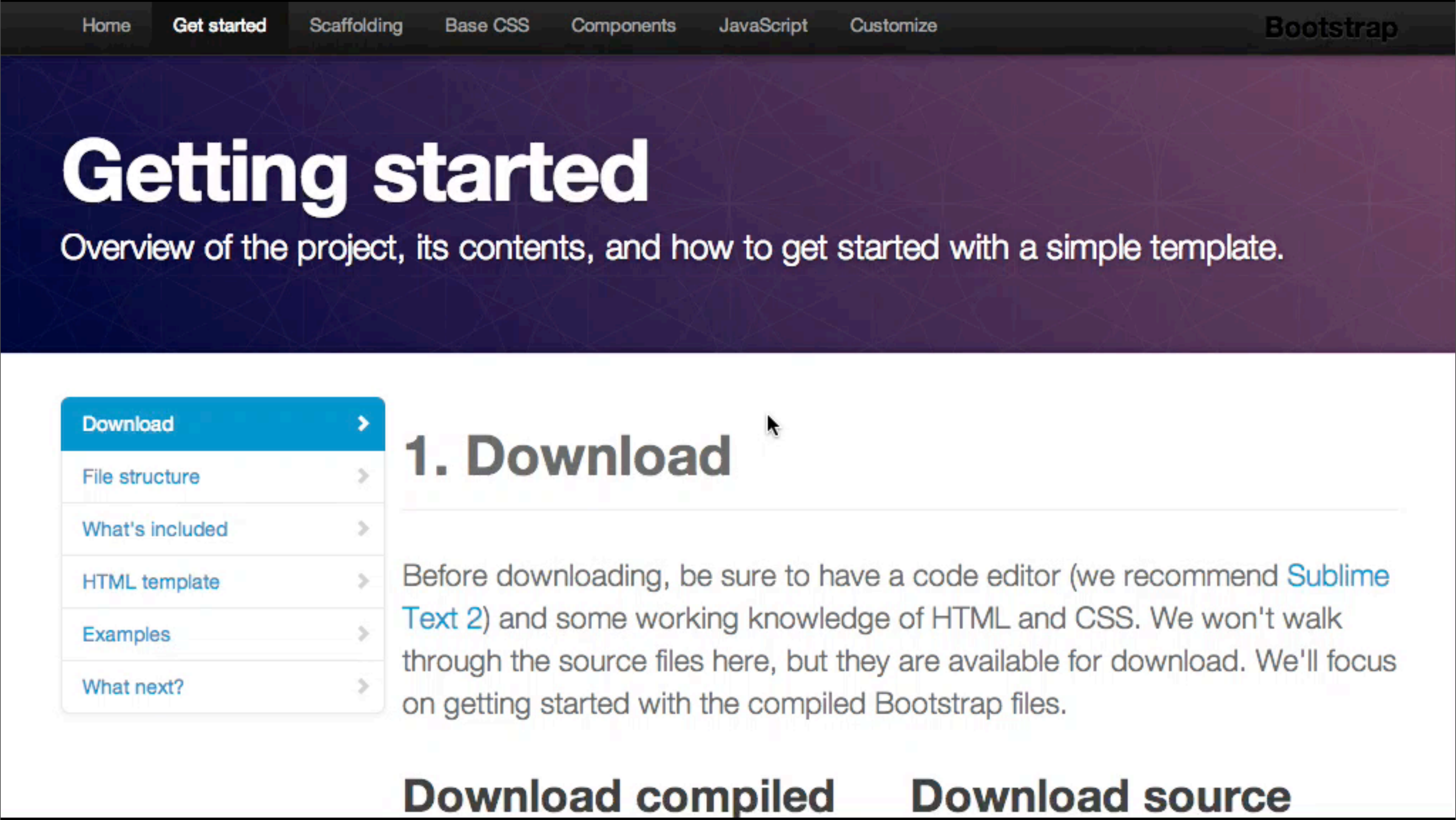
Application Configuration

» How do I configure a phone number to receive phone calls or SMS messages?
» What is an application SID?

Accounts

» What is a subaccount?
» Do subaccounts support international calls and SMS messaging?
» How do I cancel my account?

Splitting the Docs



Getting started

Overview of the project, its contents, and how to get started with a simple template.

1. Download

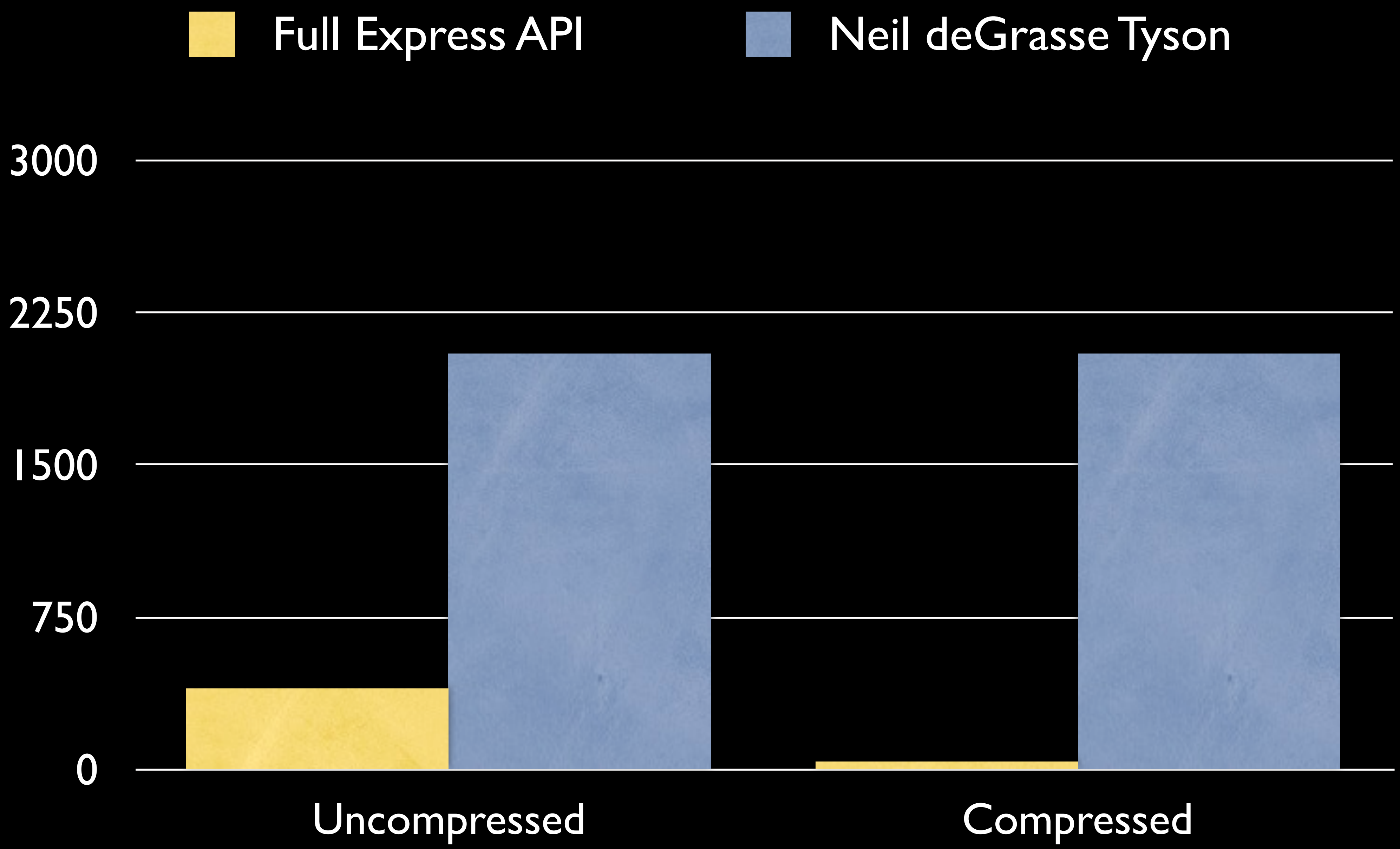
Before downloading, be sure to have a code editor (we recommend Sublime Text 2) and some working knowledge of HTML and CSS. We won't walk through the source files here, but they are available for download. We'll focus on getting started with the compiled Bootstrap files.

Download compiled

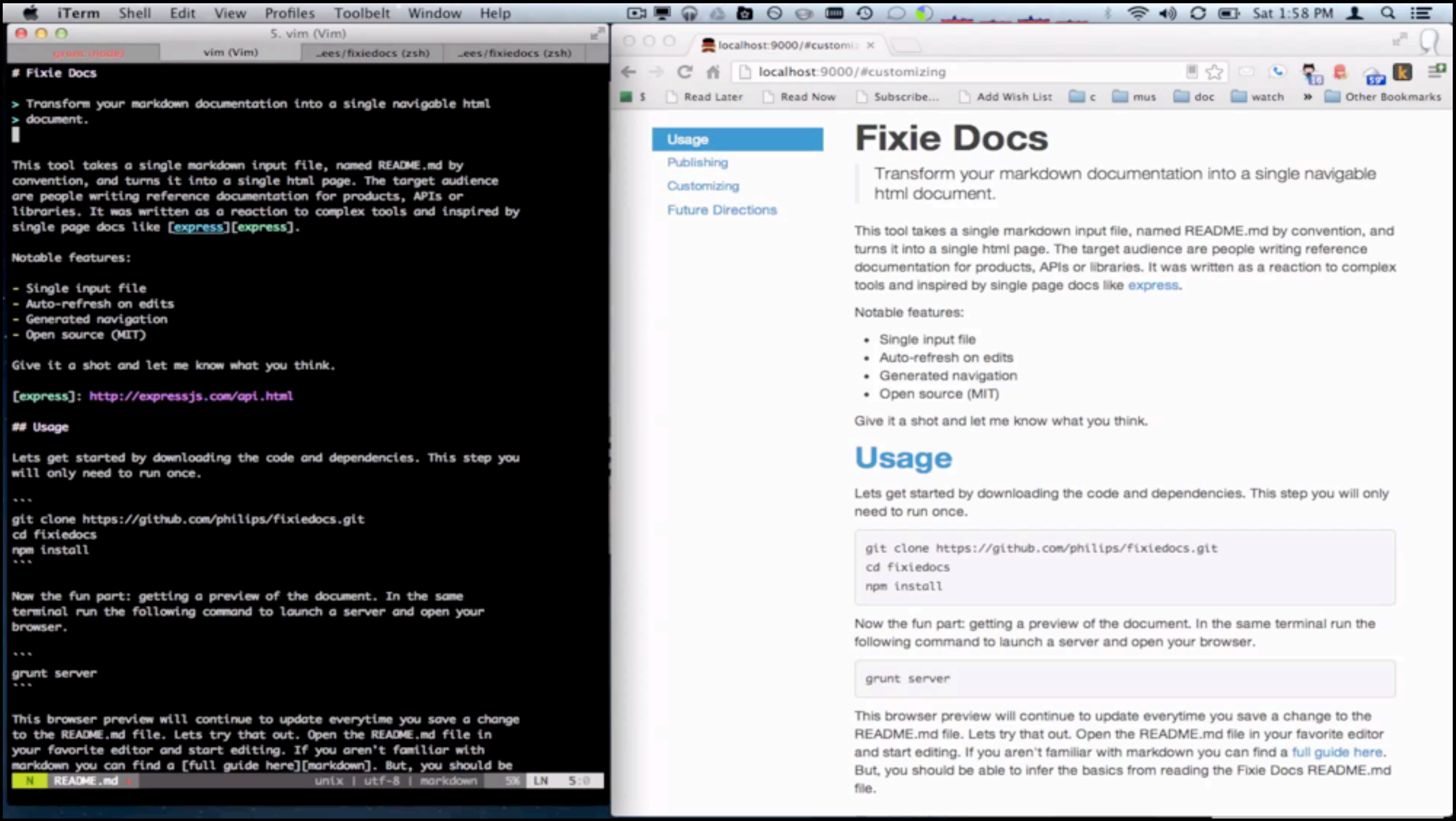
Download source

Tools

What about size?


Full Express API
Neil deGrasse Tyson
3000
2250
1500
750
0
Uncompressed
Compressed

A Product*


iTerm Shell Edit View Profiles Toolbelt Window Help
Sat 1:58 PM
5. vim (Vim)
grunt (node)
vim (Vim)
..ees/fixiedocs (zsh)
..ees/fixiedocs (zsh)
localhost:9000/#customiz ×
localhost:9000/#customizing
$ Read Later Read Now Subscribe... Add Wish List c mus doc watch » Other Bookmarks

```
# Fixie Docs

> Transform your markdown documentation into a single navigable html
> document.

This tool takes a single markdown input file, named README.md by
convention, and turns it into a single html page. The target audience
are people writing reference documentation for products, APIs or
libraries. It was written as a reaction to complex tools and inspired by
single page docs like [express][express].

Notable features:

- Single input file
- Auto-refresh on edits
- Generated navigation
- Open source (MIT)

Give it a shot and let me know what you think.

[express]: http://expressjs.com/api.html

## Usage

Lets get started by downloading the code and dependencies. This step you
will only need to run once.

```
git clone https://github.com/philips/fixiedocs.git
cd fixiedocs
npm install
```

Now the fun part: getting a preview of the document. In the same
terminal run the following command to launch a server and open your
browser.

```
grunt server
```

This browser preview will continue to update everytime you save a change
to the README.md file. Lets try that out. Open the README.md file in
your favorite editor and start editing. If you aren't familiar with
markdown you can find a [full guide here][markdown]. But, you should be
 N README.md +   unix | utf-8 | markdown 5% LN 5:0
```

Fixie Docs

> Transform your markdown documentation into a single navigable html document.

This tool takes a single markdown input file, named README.md by convention, and turns it into a single html page. The target audience are people writing reference documentation for products, APIs or libraries. It was written as a reaction to complex tools and inspired by single page docs like express.

Notable features:

- Single input file
- Auto-refresh on edits
- Generated navigation
- Open source (MIT)

Give it a shot and let me know what you think.

Usage

Lets get started by downloading the code and dependencies. This step you will only need to run once.

```
git clone https://github.com/philips/fixiedocs.git
cd fixiedocs
npm install
```

Now the fun part: getting a preview of the document. In the same terminal run the following command to launch a server and open your browser.

```
grunt server
```

This browser preview will continue to update everytime you save a change to the README.md file. Lets try that out. Open the README.md file in your favorite editor and start editing. If you aren't familiar with markdown you can find a full guide here. But, you should be able to infer the basics from reading the Fixie Docs README.md file.

github.com/philips/fixiedocs

Thanks!

@BrandonPhilips

http://ifup.org