



Making Friendly Microservices

Michele Titolo

Lead Software Engineer,
Capital One

dockercon 16



Definition: **/microservice/**

Small service that does one thing well

Independent

Own its own data



Definition: /friendly/

Helpful

An ally

Kind

Easy to use, understand

Able to coexist without causing harm

What Makes A Microservice...

Helpful, An Ally, Kind, Easy to Use,
Able to Coexist



Helpful

Documentation

Documentation

Documentation

Version And Revision History

Historical record of *when* things changed
and *why*

Live Documentation

Document Dependencies



Create A Dependency Graph

use `docker_image -t`

Autogenerate Documentation

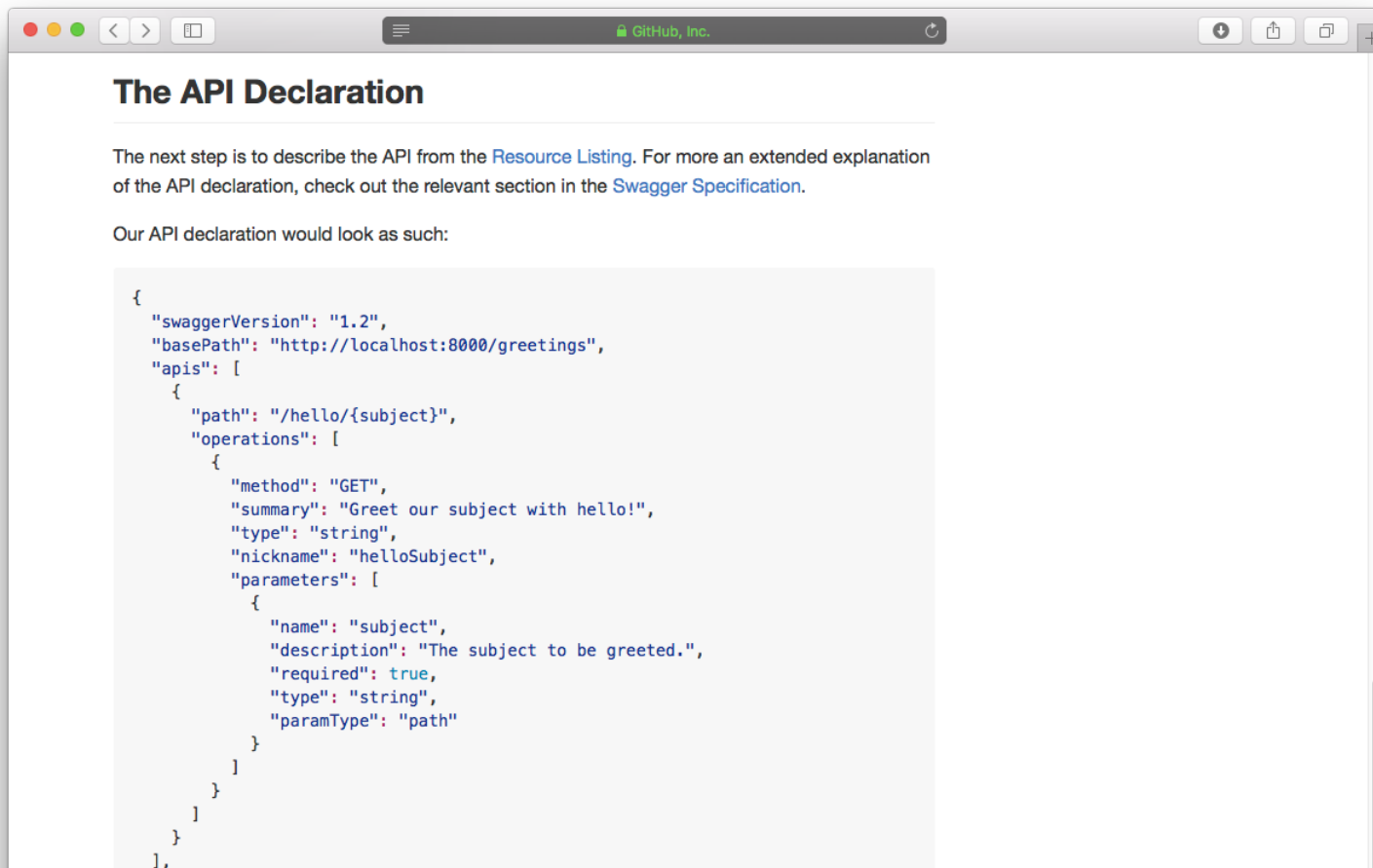
Open Source Tools

Open API Specification/Swagger

APIBlueprint

RAML

Open Api Spec



The screenshot shows a web browser window with the address bar displaying 'GitHub, Inc.'. The main content of the page is titled 'The API Declaration'. Below the title, there is a paragraph of text explaining the next step in API documentation, followed by a code block containing a JSON snippet representing an API declaration.

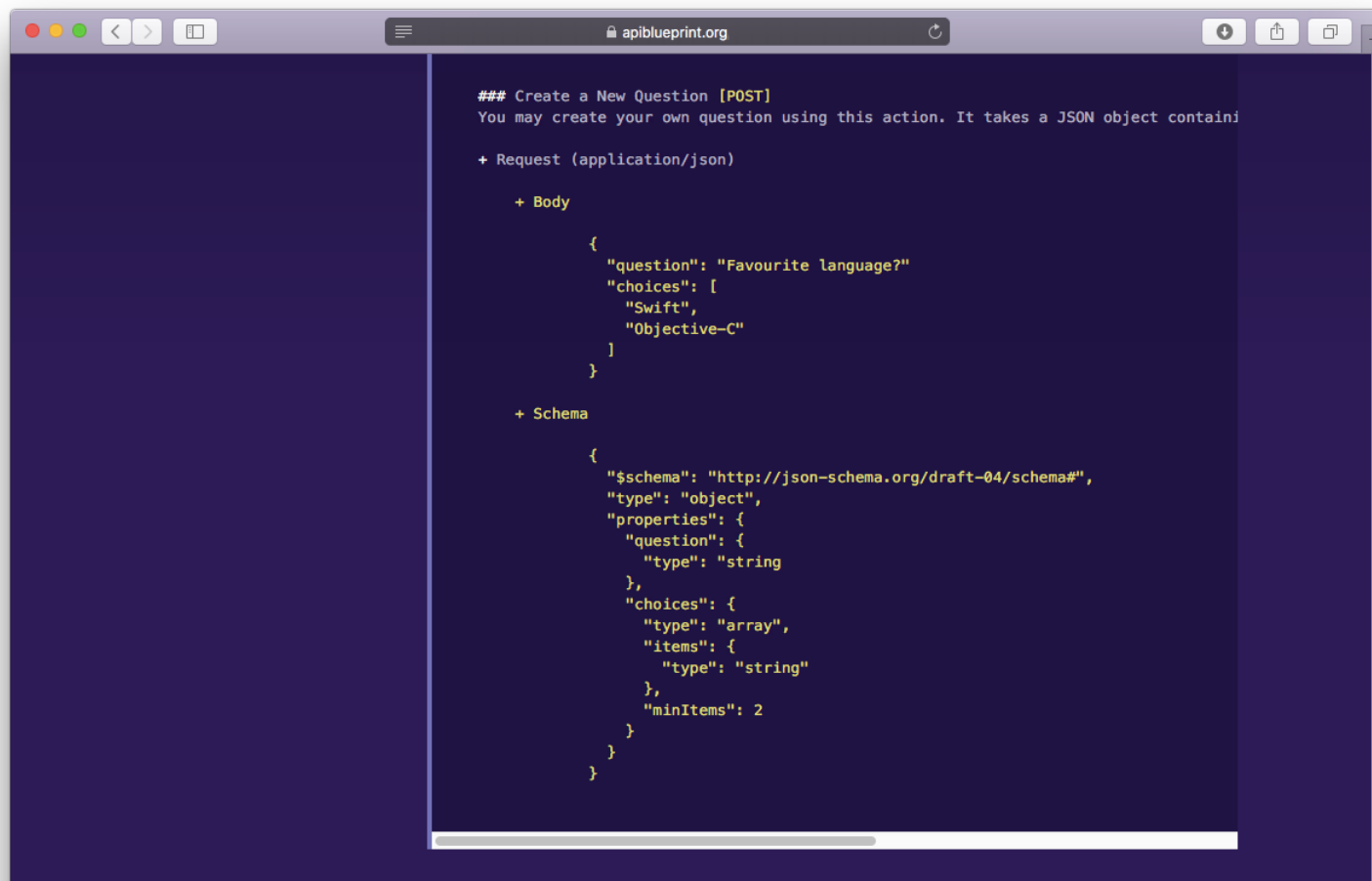
The API Declaration

The next step is to describe the API from the [Resource Listing](#). For more an extended explanation of the API declaration, check out the relevant section in the [Swagger Specification](#).

Our API declaration would look as such:

```
{
  "swaggerVersion": "1.2",
  "basePath": "http://localhost:8000/greetings",
  "apis": [
    {
      "path": "/hello/{subject}",
      "operations": [
        {
          "method": "GET",
          "summary": "Greet our subject with hello!",
          "type": "string",
          "nickname": "helloSubject",
          "parameters": [
            {
              "name": "subject",
              "description": "The subject to be greeted.",
              "required": true,
              "type": "string",
              "paramType": "path"
            }
          ]
        }
      ]
    }
  ]
},
],
```

Api Blueprint



The screenshot shows a web browser window with the URL `apiblueprint.org`. The page content is as follows:

```
### Create a New Question [POST]
You may create your own question using this action. It takes a JSON object containing the following information:

+ Request (application/json)

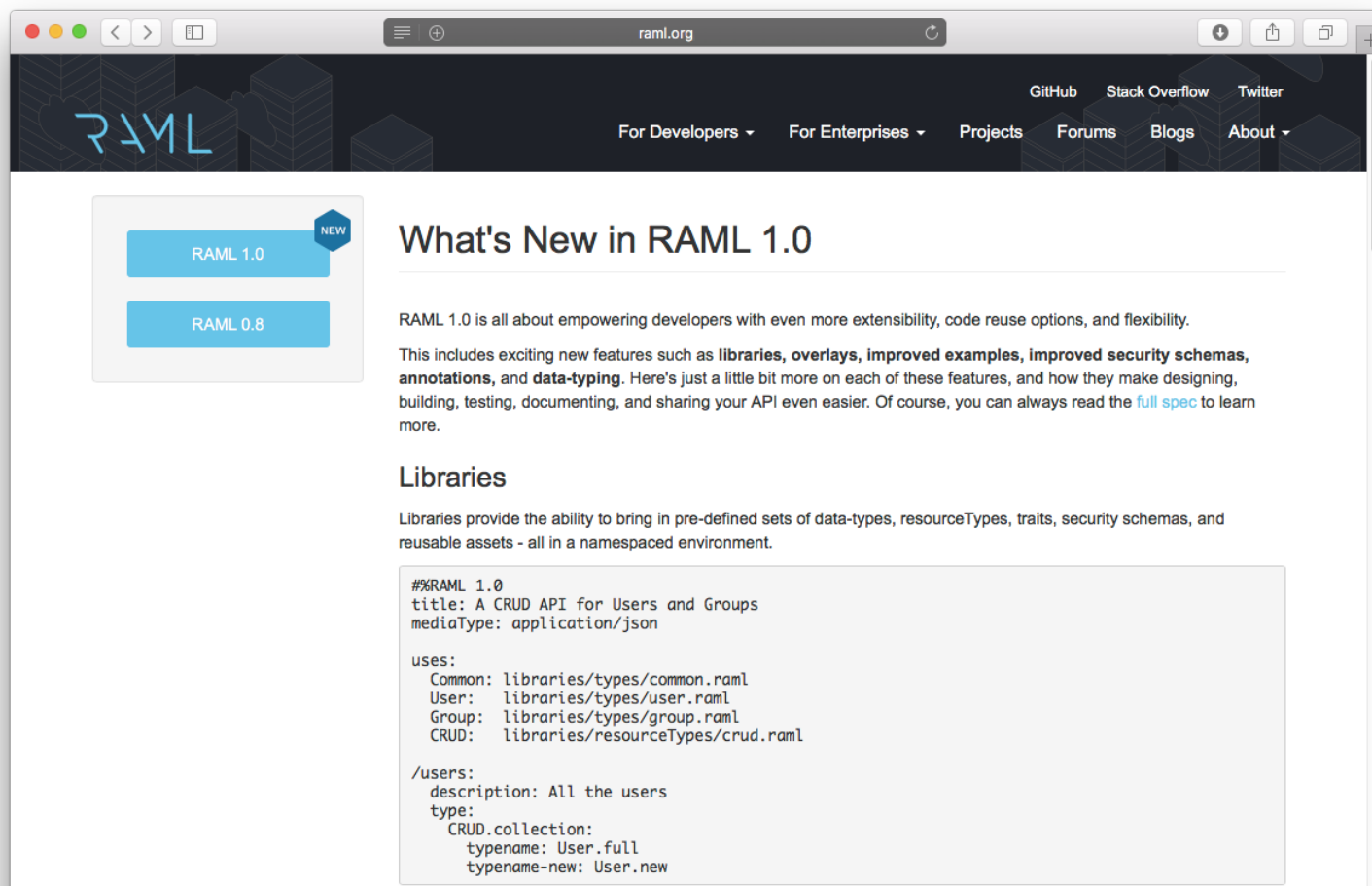
  + Body

    {
      "question": "Favourite language?"
      "choices": [
        "Swift",
        "Objective-C"
      ]
    }

  + Schema

    {
      "$schema": "http://json-schema.org/draft-04/schema#",
      "type": "object",
      "properties": {
        "question": {
          "type": "string"
        },
        "choices": {
          "type": "array",
          "items": {
            "type": "string"
          },
          "minItems": 2
        }
      }
    }
  }
```

RAML



The screenshot shows the RAML website interface. At the top, there is a navigation bar with links for GitHub, Stack Overflow, and Twitter. Below this, there are dropdown menus for 'For Developers', 'For Enterprises', 'Projects', 'Forums', 'Blogs', and 'About'. The main content area features a sidebar on the left with buttons for 'RAML 1.0' (marked as 'NEW') and 'RAML 0.8'. The main heading is 'What's New in RAML 1.0'. The text below explains that RAML 1.0 focuses on extensibility, code reuse, and flexibility, listing features like libraries, overlays, improved examples, security schemas, annotations, and data-typing. A section titled 'Libraries' describes their use in providing pre-defined data-types and reusable assets. A code block at the bottom shows a sample RAML 1.0 snippet for a CRUD API.

RAML 1.0

RAML 0.8

What's New in RAML 1.0

RAML 1.0 is all about empowering developers with even more extensibility, code reuse options, and flexibility. This includes exciting new features such as **libraries**, **overlays**, **improved examples**, **improved security schemas**, **annotations**, and **data-typing**. Here's just a little bit more on each of these features, and how they make designing, building, testing, documenting, and sharing your API even easier. Of course, you can always read the [full spec](#) to learn more.

Libraries

Libraries provide the ability to bring in pre-defined sets of data-types, resourceTypes, traits, security schemas, and reusable assets - all in a namespaced environment.

```
##%RAML 1.0
title: A CRUD API for Users and Groups
mediaType: application/json

uses:
  Common: libraries/types/common.raml
  User: libraries/types/user.raml
  Group: libraries/types/group.raml
  CRUD: libraries/resourceTypes/crud.raml

/users:
  description: All the users
  type:
    CRUD.collection:
      typename: User.full
      typename-new: User.new
```


A photograph of several walrus resting on a sandy beach. The walrus in the foreground on the right is the most prominent, showing its characteristic tusks and whiskers. The image is dimly lit, and the word "Ally" is overlaid in a large, white, sans-serif font in the center. The background shows other walrus resting on the sand, slightly out of focus.

Ally

Integrate Monitoring Tools

Log All The Things

Use unique request IDs so logs are easier to sort through

**Avoid Playing
Red Light Green Light**

What Happens When A Dependency Fails?

Hint:

**Your Service Should
Not Go Down**

Macro Dependencies

Micro Dependencies

**Help Identify
Bottlenecks
Before They Happen**

Communication

Gather Data

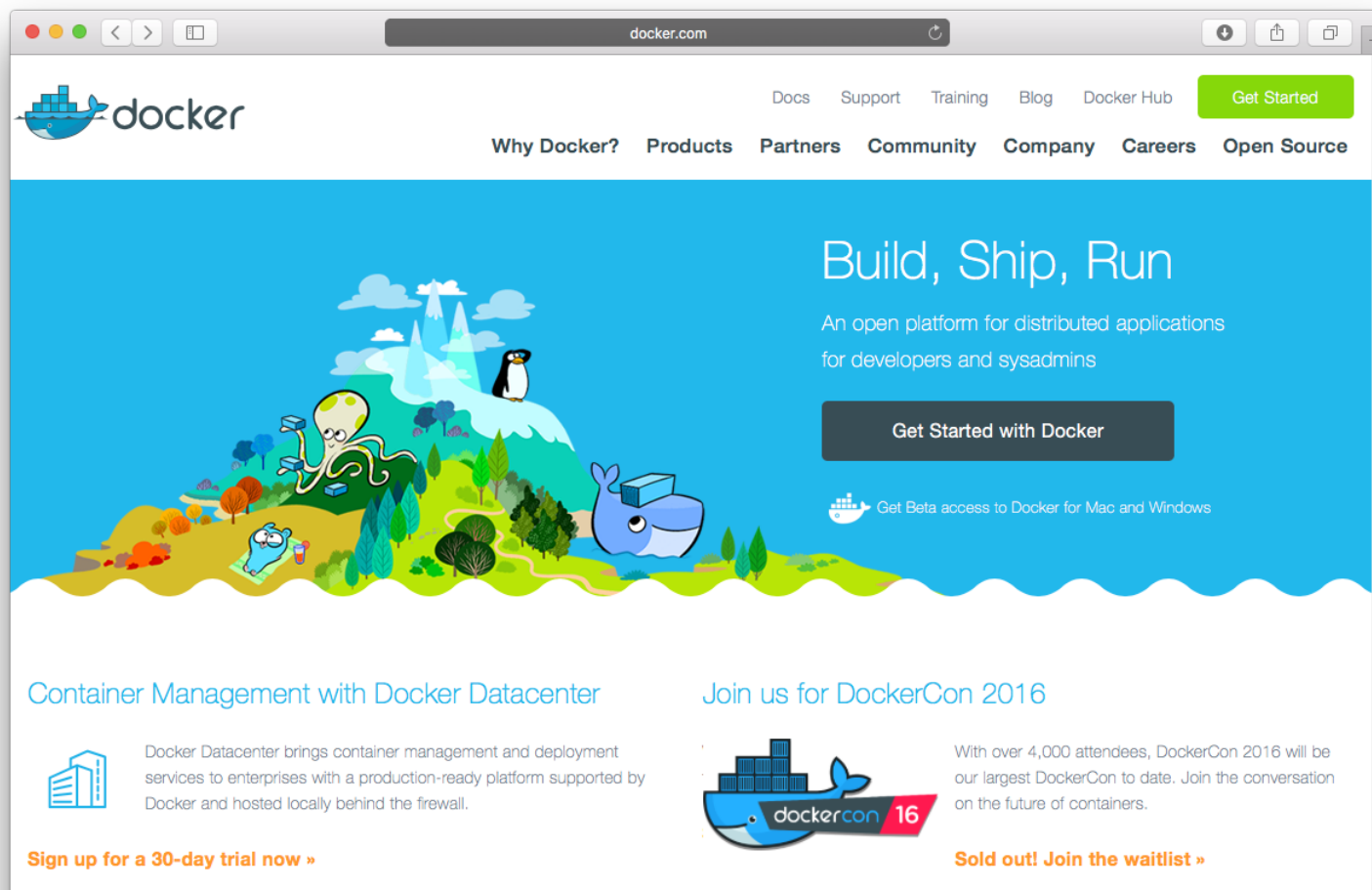
Tools

Docker

Atlas

Zipkin

Docker



The image shows a browser window displaying the Docker website. The browser's address bar shows "docker.com". The website header includes the Docker logo, navigation links for "Docs", "Support", "Training", "Blog", and "Docker Hub", and a prominent green "Get Started" button. Below the header is a main navigation menu with links for "Why Docker?", "Products", "Partners", "Community", "Company", "Careers", and "Open Source". The main content area features a large blue banner with the text "Build, Ship, Run" and a sub-headline "An open platform for distributed applications for developers and sysadmins". A dark button labeled "Get Started with Docker" is centered on the banner. Below the banner is a colorful illustration of a landscape with a penguin, an octopus, and a whale. Below the illustration are two sections: "Container Management with Docker Datacenter" and "Join us for DockerCon 2016".

docker.com

docker


Docs Support Training Blog Docker Hub [Get Started](#)

[Why Docker?](#) [Products](#) [Partners](#) [Community](#) [Company](#) [Careers](#) [Open Source](#)


Build, Ship, Run

An open platform for distributed applications for developers and sysadmins

[Get Started with Docker](#)

 [Get Beta access to Docker for Mac and Windows](#)

Container Management with Docker Datacenter

 Docker Datacenter brings container management and deployment services to enterprises with a production-ready platform supported by Docker and hosted locally behind the firewall.

[Sign up for a 30-day trial now »](#)

Join us for DockerCon 2016

 With over 4,000 attendees, DockerCon 2016 will be our largest DockerCon to date. Join the conversation on the future of containers.

[Sold out! Join the waitlist »](#)

Atlas

The screenshot shows the GitHub repository page for `Netflix / atlas`. The page includes navigation links, repository statistics, and a list of recent commits.

Repository Statistics:

- Watch: 299
- Star: 958
- Fork: 78

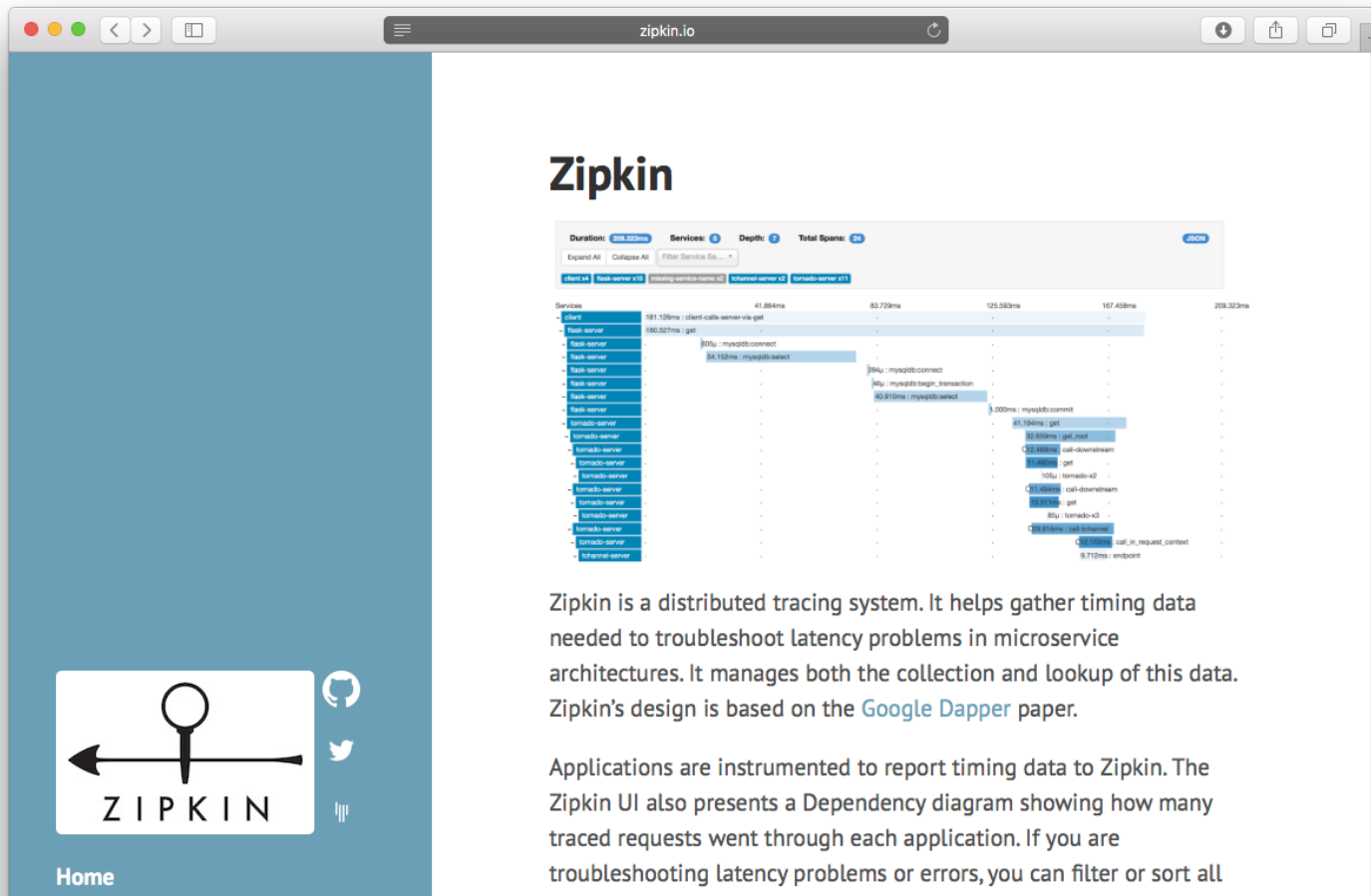
Repository Details:

- Issues: 18
- Pull requests: 0
- Wiki
- Pulse
- Graphs

Commit History:

Commit Message	Time Ago
fix bug with status when the route is created	3 months ago
update license headers	6 months ago
always add the axis color	11 days ago
work-around for null context classloader	a month ago
improve performance of misses with SmallHashMap	7 days ago
add IntHashSet util for building index	a month ago
fixes #373 - return 400 status code for bad object	8 days ago
default /healthcheck based on ServiceManager	3 months ago
fix path when test are run with being forked	4 months ago
Merge pull request #377 from bharrington/issue-373	8 days ago

Zipkin



The screenshot shows the Zipkin web interface at zipkin.io. The main content area displays a dependency diagram for a request. At the top, there are filters for Duration (0:00.00ms), Services (3), Depth (2), and Total Spans (28). Below the filters, a table lists the services and their durations. The diagram shows a flow starting from a client, through a feish-server, then to a tomcat-server, and finally to a tomcat-server. The diagram is a dependency diagram showing the flow of a request through various services. The services listed are client, feish-server, tomcat-server, and tomcat-server. The durations for each service are: client (181.120ms), feish-server (180.527ms), tomcat-server (41.854ms), and tomcat-server (83.729ms). The diagram also shows the total spans for each service: client (1), feish-server (1), tomcat-server (1), and tomcat-server (1). The diagram is a dependency diagram showing the flow of a request through various services. The services listed are client, feish-server, tomcat-server, and tomcat-server. The durations for each service are: client (181.120ms), feish-server (180.527ms), tomcat-server (41.854ms), and tomcat-server (83.729ms). The diagram also shows the total spans for each service: client (1), feish-server (1), tomcat-server (1), and tomcat-server (1).

Zipkin is a distributed tracing system. It helps gather timing data needed to troubleshoot latency problems in microservice architectures. It manages both the collection and lookup of this data. Zipkin's design is based on the [Google Dapper](#) paper.

Applications are instrumented to report timing data to Zipkin. The Zipkin UI also presents a Dependency diagram showing how many traced requests went through each application. If you are troubleshooting latency problems or errors, you can filter or sort all



Kind

**Make Your Service
Easy To Deploy
And Scale**

**Consumers Should Be
Able To Hit The API
Directly In A Non-Prod
Environment**

Consistent Error Messaging

HTTP/1.1

206 Partial Content

**Do Not Make Others
Setup A Development
Environment To
Troubleshoot Issues**

Tools

Docker

Jenkins

Capistrano

Docker

The image shows a browser window displaying the Docker website. The browser's address bar shows "docker.com". The website features a navigation menu with links for "Docs", "Support", "Training", "Blog", "Docker Hub", and a prominent green "Get Started" button. Below the navigation, there are links for "Why Docker?", "Products", "Partners", "Community", "Company", "Careers", and "Open Source". The main content area has a blue background with a colorful illustration of a landscape with a penguin, an octopus, and a whale. The headline reads "Build, Ship, Run" and is followed by the text "An open platform for distributed applications for developers and sysadmins". A dark button labeled "Get Started with Docker" is centered below this text. A link for "Get Beta access to Docker for Mac and Windows" is also present. The lower section of the page is divided into two columns. The left column is titled "Container Management with Docker Datacenter" and includes a sub-image of server racks, a short description of the service, and a link to "Sign up for a 30-day trial now ». The right column is titled "Join us for DockerCon 2016" and features a logo for "dockercon 16", a description of the event, and a link to "Sold out! Join the waitlist »".

docker.com

docker

Docs Support Training Blog Docker Hub [Get Started](#)

[Why Docker?](#) [Products](#) [Partners](#) [Community](#) [Company](#) [Careers](#) [Open Source](#)

Build, Ship, Run

An open platform for distributed applications for developers and sysadmins

[Get Started with Docker](#)


[Get Beta access to Docker for Mac and Windows](#)

Container Management with Docker Datacenter

 Docker Datacenter brings container management and deployment services to enterprises with a production-ready platform supported by Docker and hosted locally behind the firewall.

[Sign up for a 30-day trial now »](#)

Join us for DockerCon 2016

 With over 4,000 attendees, DockerCon 2016 will be our largest DockerCon to date. Join the conversation on the future of containers.

[Sold out! Join the waitlist »](#)

Containers

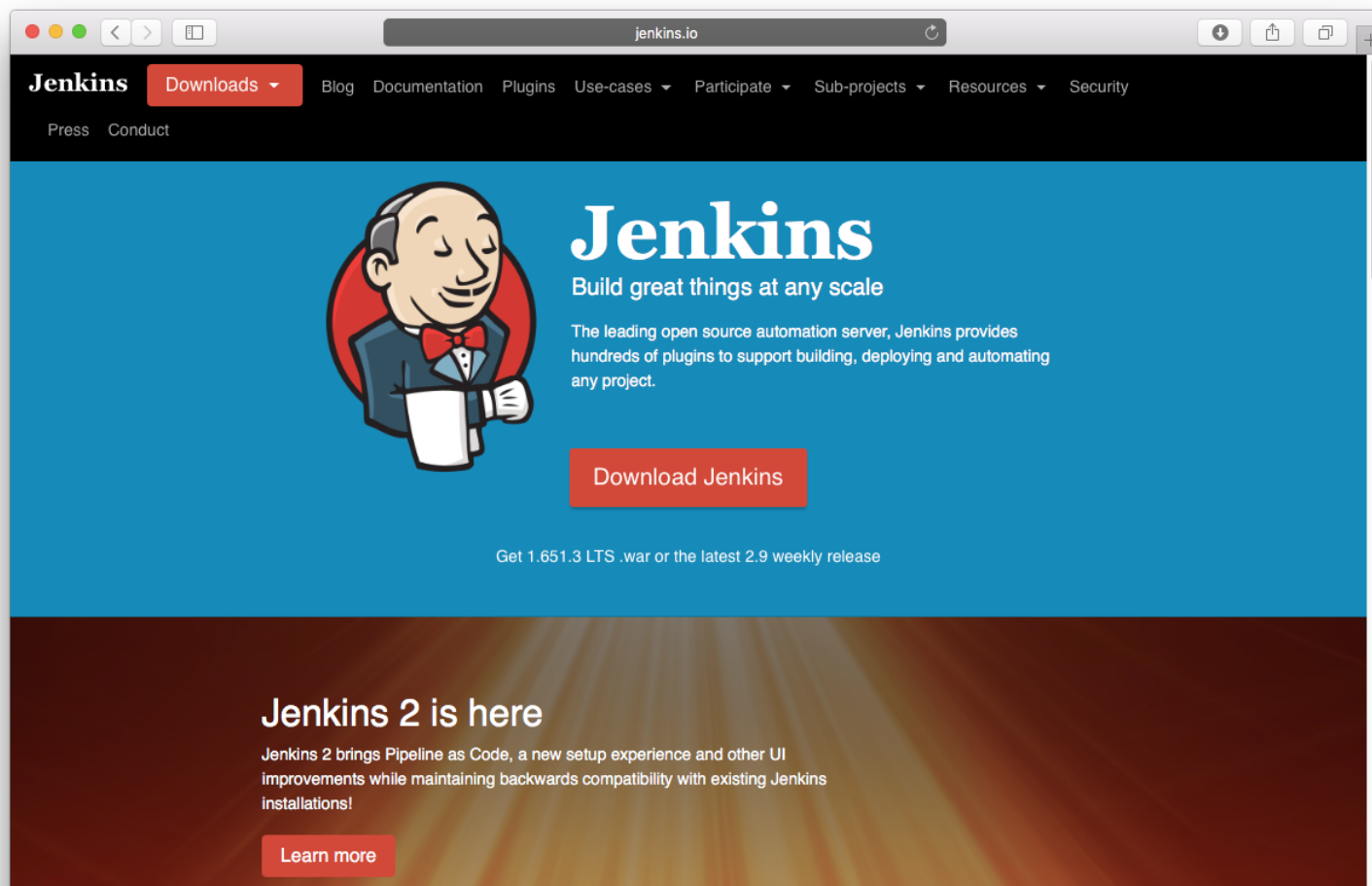
Each service should stand alone

Use `docker_compose` for dependencies

Use `depends_on` to bring up multiple services

Use Containers For Local Development

Jenkins




The image shows a browser window displaying the Jenkins website. The browser's address bar shows "jenkins.io". The website has a dark navigation bar with the "Jenkins" logo on the left and a menu of links: "Downloads", "Blog", "Documentation", "Plugins", "Use-cases", "Participate", "Sub-projects", "Resources", and "Security". Below the navigation bar, there is a blue hero section. On the left of this section is the Jenkins mascot, a cartoon man in a tuxedo. To the right of the mascot, the word "Jenkins" is written in a large, white, serif font. Below the name, the tagline "Build great things at any scale" is displayed. Further down, a paragraph describes Jenkins as the leading open source automation server. A prominent red button labeled "Download Jenkins" is centered in this section. Below the button, text provides instructions on how to get the latest version. The bottom of the page features a dark brown section with the heading "Jenkins 2 is here" and a paragraph explaining the new features of Jenkins 2. A red button labeled "Learn more" is positioned at the bottom of this section.

jenkins.io

Jenkins Downloads Blog Documentation Plugins Use-cases Participate Sub-projects Resources Security

Press Conduct



Jenkins

Build great things at any scale

The leading open source automation server, Jenkins provides hundreds of plugins to support building, deploying and automating any project.

Download Jenkins

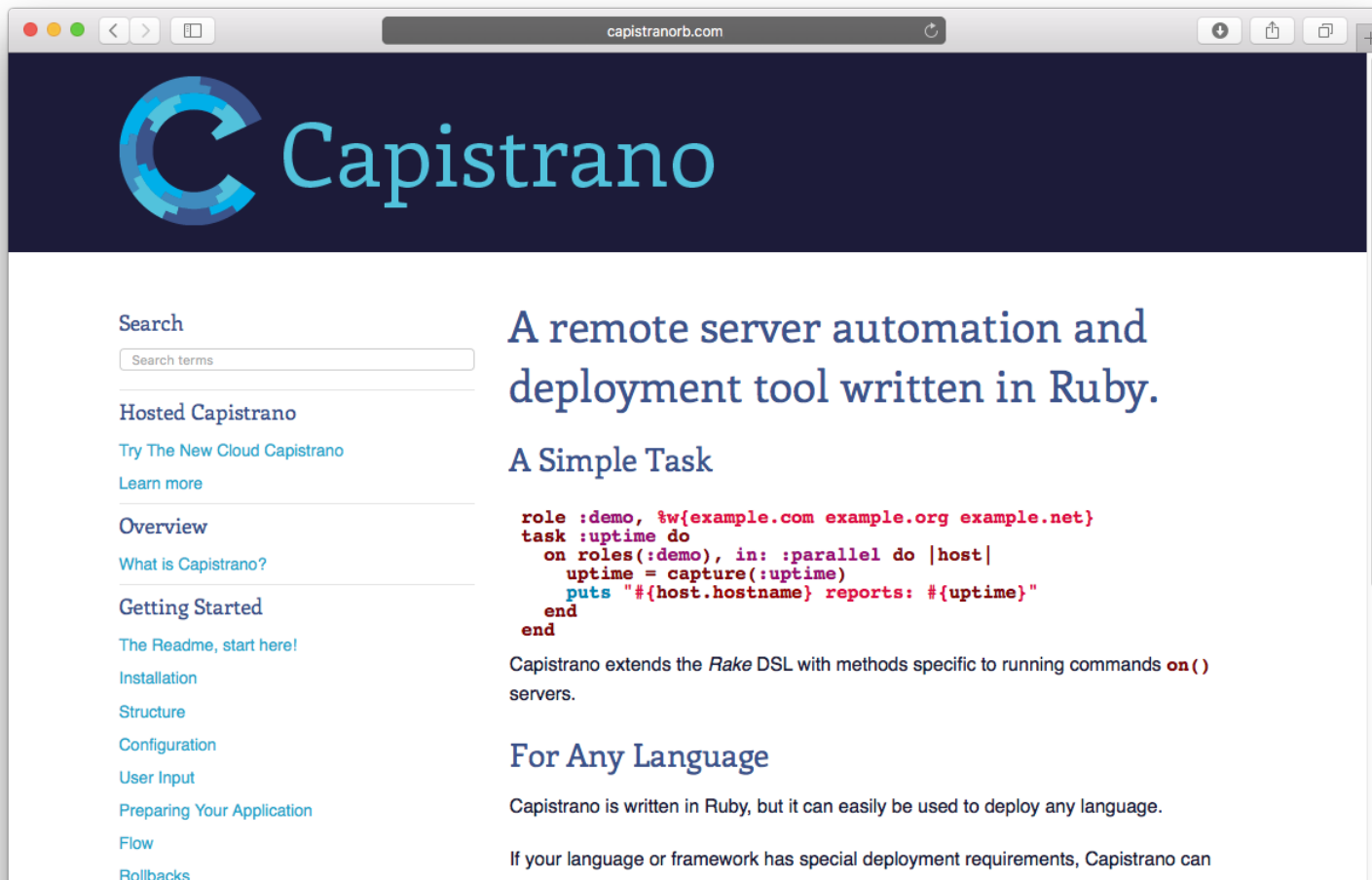
Get 1.651.3 LTS .war or the latest 2.9 weekly release

Jenkins 2 is here

Jenkins 2 brings Pipeline as Code, a new setup experience and other UI improvements while maintaining backwards compatibility with existing Jenkins installations!

Learn more

Capistrano



The image shows a browser window displaying the Capistrano website. The browser's address bar shows "capistranorb.com". The website has a dark blue header with the Capistrano logo (a stylized 'C' made of blue segments) and the word "Capistrano" in a light blue font. Below the header, the page is divided into a left sidebar and a main content area. The sidebar contains a search bar, a "Hosted Capistrano" section with links to "Try The New Cloud Capistrano" and "Learn more", an "Overview" section with links to "What is Capistrano?", "Getting Started", "The Readme, start here!", "Installation", "Structure", "Configuration", "User Input", "Preparing Your Application", "Flow", and "Rollbacks". The main content area features a large heading "A remote server automation and deployment tool written in Ruby.", followed by a sub-heading "A Simple Task" and a code block showing a Capistrano task definition. Below the code, there is a paragraph explaining that Capistrano extends the *Rake* DSL with methods for running commands on servers. Another sub-heading "For Any Language" is followed by a paragraph stating that Capistrano is written in Ruby but can be used to deploy any language.

Search

Hosted Capistrano

[Try The New Cloud Capistrano](#)

[Learn more](#)

Overview

[What is Capistrano?](#)

Getting Started

[The Readme, start here!](#)

[Installation](#)

[Structure](#)

[Configuration](#)

[User Input](#)

[Preparing Your Application](#)

[Flow](#)

[Rollbacks](#)

A remote server automation and deployment tool written in Ruby.

A Simple Task

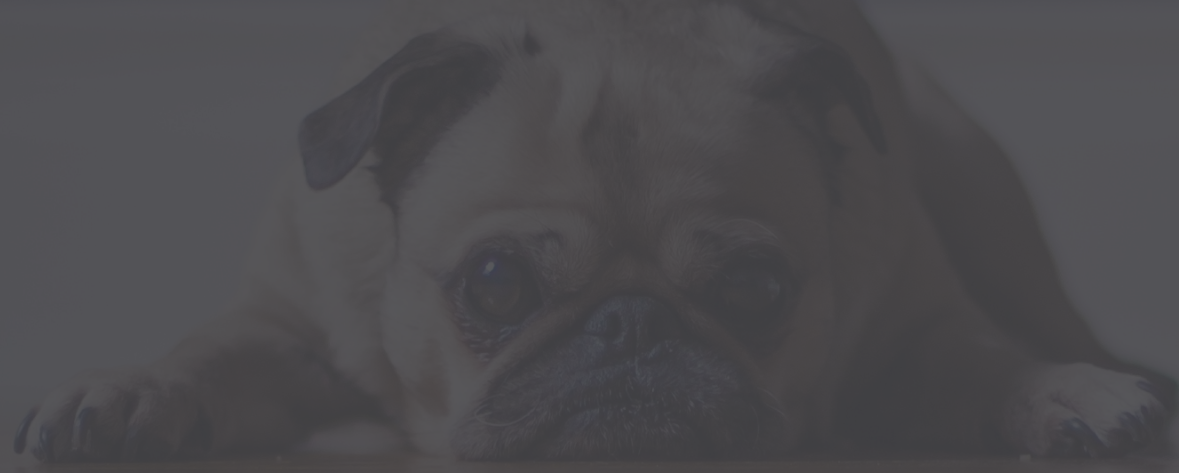
```
role :demo, %w{example.com example.org example.net}
task :uptime do
  on roles(:demo), in: :parallel do |host|
    uptime = capture(:uptime)
    puts "#{host.hostname} reports: #{uptime}"
  end
end
```

Capistrano extends the *Rake* DSL with methods specific to running commands `on()` servers.

For Any Language

Capistrano is written in Ruby, but it can easily be used to deploy any language.

If your language or framework has special deployment requirements, Capistrano can



Easy To Use

**Use One Base URL
For Everything**

**Use An API Gateway
Or Load Balancer To
Route API Calls**

**Be Security
Conscious**

Cookie Security

Don't use wildcards

Always use HTTPS and Secure

Set correct domain

Mobile Security

SSL Certificate Pinning

Tools

Zuul

NGINIX

HAProxy

Zuul

The screenshot shows the GitHub repository page for Netflix / zuul. The browser address bar shows "GitHub, Inc.". The navigation bar includes "Personal", "Open source", "Business", "Explore", "Pricing", "Blog", "Support", "This repository", "Search", "Sign in", and "Sign up". The repository name "Netflix / zuul" is displayed with 337 Watch, 1,096 Star, and 295 Fork counts. Below the repository name are tabs for "Code", "Issues 37", "Pull requests 2", "Wiki", "Pulse", and "Graphs". A description states: "Zuul is a gateway service that provides dynamic routing, monitoring, resiliency, security, and more." A progress bar shows 257 commits, 8 branches, 43 releases, and 34 contributors. Below the progress bar are buttons for "Branch: 1.x", "New pull request", "Find file", and "Clone or download". The commit history table is as follows:

Commit	Message	Time
kerumal Merge pull request #225 from Netflix#127		Latest commit 48594dc 11 days ago
codequality	Remove unneeded files	a year ago
gradle	Move to gradle 2.6 and nebula.netflixoss 3.0.0	9 months ago
zuul-core	Use daemon threads for filter pollers	a month ago
zuul-netflix-webapp	Fix the asm mismatch errors when for gradle jettyRun task	12 days ago
zuul-netflix	Fix for #127	11 days ago
zuul-simple-webapp	Change the default origin in zuul-simple-webapp to be http://httpbin.org	12 days ago
.gitignore	upgrading nebula.netflixoss to 3.1.2	7 months ago
.netflixoss	Added .netflixoss config file to tell our jenkins job to use JDK8 for...	a year ago
.travis.yml	Updated Bintray API key	22 days ago
CHANGELOG.md	New nebula.netflixoss build	a year ago

NGINX

The image shows a browser window displaying the NGINX website. The browser's address bar shows "NGINX, Inc". The website's header features the NGINX logo in green on the left, and two buttons labeled "FREE TRIAL" and "CONTACT SALES" on the right. Below the header is a navigation menu with the following items: PRODUCTS, SOLUTIONS, RESOURCES, SUPPORT & SERVICES, COMPANY, CUSTOMERS, and BLOG. A search icon and a user profile icon are also present in the navigation area. The main content area has a dark background with a pattern of light-colored cubes. On the left, a tablet displays a graphic with a grid of cubes and the text "MICROSERVICES From Design to Deployment" and the NGINX logo. To the right of the tablet, the text "Get the latest on" is followed by the large heading "Microservices Design & Deployment". Below this heading is a yellow button with the text "DOWNLOAD" and a right-pointing arrow.

NGINX

FREE TRIAL CONTACT SALES

PRODUCTS SOLUTIONS RESOURCES SUPPORT & SERVICES COMPANY CUSTOMERS BLOG

Get the latest on

Microservices Design & Deployment

DOWNLOAD


MICROSERVICES
From Design to Deployment

NGINX

HAProxy

HAProxy

The Reliable, High Performance TCP/HTTP Load Balancer




Mirror Sites: [Master](#)
Language: [English](#)

Quick links


- [Quick News](#)
- [Recent News](#)
- [Description](#)
- [Main features](#)
- [Supported Platforms](#)
- [Performance](#)
- [Reliability](#)
- [Security](#)
- [Download](#)
- [Documentation](#)
- [Live demo](#)
- [They use it!](#)
- [Commercial Support](#)
- [Add-on features](#)
- [Other Solutions](#)
- [Contacts](#)
- [External links](#)
- [Discussions](#)
- [Mailing list archives](#)
- [10GbE load-balancing \(updated\)](#)
- [Contributions](#)
- [Coding style](#)
- [Known bugs](#)

[Web Based User Interface](#)
[HA-Top: Nurses interface](#)



visitors online

[Donate \\$](#) [Donate €](#)
Thanks for your support!



Latest versions

Branch	Description	Last version	Released	Links	Notes
Development	1.7-dev	1.7-dev3	2016/05/10	git / web / dir	may be broken
1.6	1.6-stable	1.6.5	2016/05/10	git / web / dir	Stable version
1.5	1.5-stable	1.5.18	2016/05/10	git / web / dir	Stable version
1.4	1.4-stable	1.4.27	2016/03/14	git / web / dir	Critical fixes only
1.3	1.3-stable	1.3.28	2016/03/14	git / web / dir	Unmaintained
1.3.15	1.3.15-maint	1.3.28.14	2015/02/01	git / web / dir	Unmaintained
1.3.14	1.3.14-maint	1.3.28.14	2009/07/27	git / web / dir	Unmaintained
1.2	1.2-stable	1.2.18	2008/05/25	git / web / dir	Unmaintained
1.1	1.1-stable	1.1.34	2006/01/29	git / web / dir	Unmaintained
1.0	1.0-old	1.0.2	2001/12/30	git / web / dir	Unmaintained

Quick News

April, 13th, 2016 : 1.5.17

I've just released HAProxy 1.5.17 which fixes the CPU usage regression introduced in 1.5.16 after improperly fixing the buffer space calculation code. It also fixes another bug impacting header captures when there are exactly MAX_HDR_HISTORY headers captured (10 by default), in which case the process may crash by dereferencing a pointer at a negative position in the pointers history. The remaining patches are minor bugs and documentation. All users of 1.5 should upgrade to 1.5.17, there are too many bugs impacting older versions. Code and changelog are available [here](#) as usual.

March, 14th, 2016 : 1.7-dev2, 1.6.4, 1.5.16, 1.4.27, 1.3.28 (EOL)

After 3 months for the most recent versions and one year for the oldest ones, a new version was released in each supported branch. This also marks the last 1.3 release as this branch has now reached end-of-life almost 10 years after it was created.

A number of important bugs were fixed since last releases. Some of them impact 1.6 when using http-reuse (orphaned connections). A few Lua bugs were fixed as well, one of them causing a segfault and another one dead connections. Sample fetch functions were protected against misuse of layer 7 in tcp connection rules causing a segfault. And session variables could also be improperly used in connection rules with the same effect. For the less important fixes, some race conditions were addressed in the systemd wrapper possibly causing the orphaned processes some people were experiencing. In 1.5 there were two issues with idle keep-alive timeout handling on the server side, sometimes causing some short-time busy polling loops and sometimes causing a new incoming request to be aborted on a persistent connection (that browsers had to resend).

In 1.7-dev we've got the filters which introduce a number of hooks to plug some code in a more flexible way than analysers. The compression code was already

A photograph of two dogs, possibly pugs, sitting in a wire cage. The dog on the left is white with a black collar and a bone-shaped tag that reads "BABY PETER". The dog on the right is also white with a black collar and a tag. The image is dimly lit and has a dark overlay. The word "Coexist" is written in large white letters across the center of the image.

Coexist

Consistency Conventions

**How Does Your
Service Fit In With
The Rest?**

**Services Need To
Work Together**

**Don't Take Single
Responsibility
Too Far**



Assumptions

Tools

Microservice Discovery

API Discovery



The Friendly Microservice

Helpful Documentation

Built With Monitoring And Troubleshooting In Mind

Easily Deployable

Easy To Consume

Coexists With Established Conventions

Thank you!



Resources

- <https://opencredo.com/rest-api-tooling-review/>
- <http://www.capitalone.io/blog/delivering-microservices-for-enterprise-with-devops/>
- <https://medium.com/capital-one-developers/mobile-orchestration-innovation-on-the-edge-9835e4cbd69e#.vpbsiyqmq>
- <http://swagger.io/open-source-integrations/>
- https://en.wikipedia.org/wiki/List_of_HTTP_status_codes
- [https://www.owasp.org/index.php/Testing_for_cookies_attributes_\(OTG-SESS-002\)](https://www.owasp.org/index.php/Testing_for_cookies_attributes_(OTG-SESS-002))
- <http://www.onegeek.com.au/articles/waiting-for-dependencies-in-docker-compose>
- <http://techblog.netflix.com/2014/12/introducing-atlas-netflixs-primary.html>

Photo Credits

- <https://unsplash.com/photos/h13Y8vyIXNU>
- <https://unsplash.com/collections/380/petunia-the-pug>
- <https://www.flickr.com/photos/32179495@N04/21211072432/>
- <https://www.flickr.com/photos/78134717@N08/7154609638>
- <https://www.flickr.com/photos/33622296@N04/5870006633/>
- <https://www.flickr.com/photos/7255089@N05/5415983081>
- <https://www.flickr.com/photos/95477820@N02/8730925814>
- <https://www.flickr.com/photos/35137530@N00/16406998986/>