mod_h2

Tales of
Mystery and Imagination

Stefan Eissing, greenbytes GmbH / @icing

Who?

- Computerized since 1982
- Co-founded <green/>bytes in 2000
- Small Dev-Shop with some High Profile Customers
- Hired gun, casual gamer
- Started mod_h2 early 2015 on behalf of GSMA
- Apache httpd PMC member since July

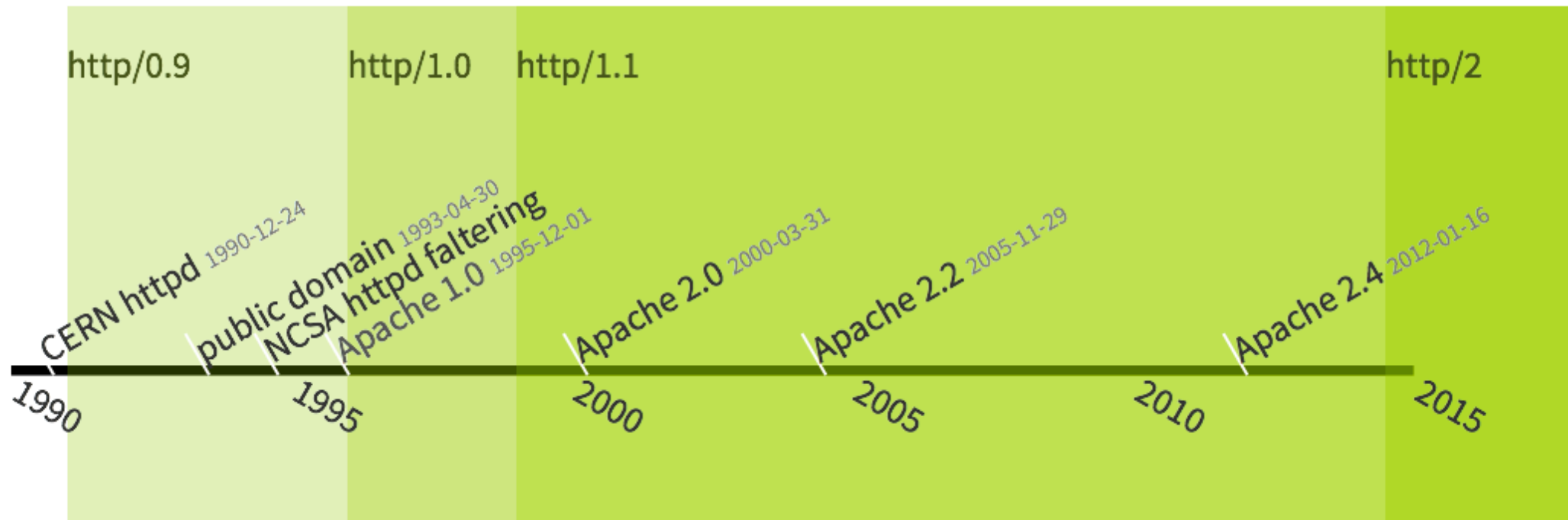Stefan Eissing, greenbytes GmbH / @icing

Tales of
Mystery and Imagination

- Mysteries
  - HTTP Performance
  - httpd/mod_h2 Implementation
- Imagination
  - httpd/mod_h2 Development
  - HTTP Future

httpd

http/0.9   http/1.0   http/1.1   http/2

CERN httpd 1990-12-24

public domain 1993-04-30
NCSA httpd faltering
Apache 1.0 1995-12-01

Apache 2.0 2000-03-31

Apache 2.2 2005-11-29

Apache 2.4 2012-01-16

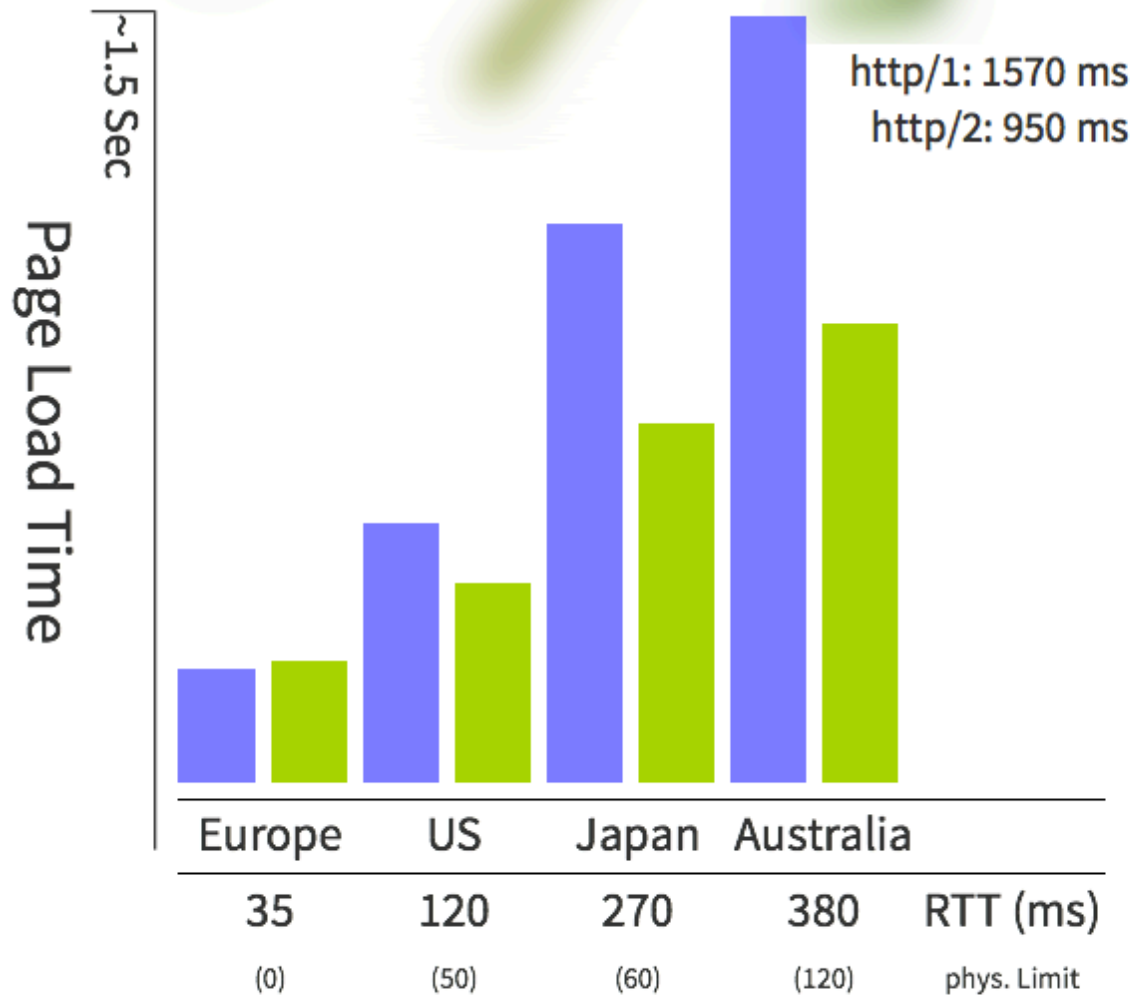1990   1995   2000   2005   2010   2015

HTTP Performance

- Fight the Latency!
- Local Bliss
- Not What It Used to be
- The New Kid
- Release it already!
- Deploy a New Toy

Fight the Latency!

10 GB/s from Europe to apache.org in...

http/1: 1570 ms
http/2: 950 ms

~1.5 Sec

Page Load Time

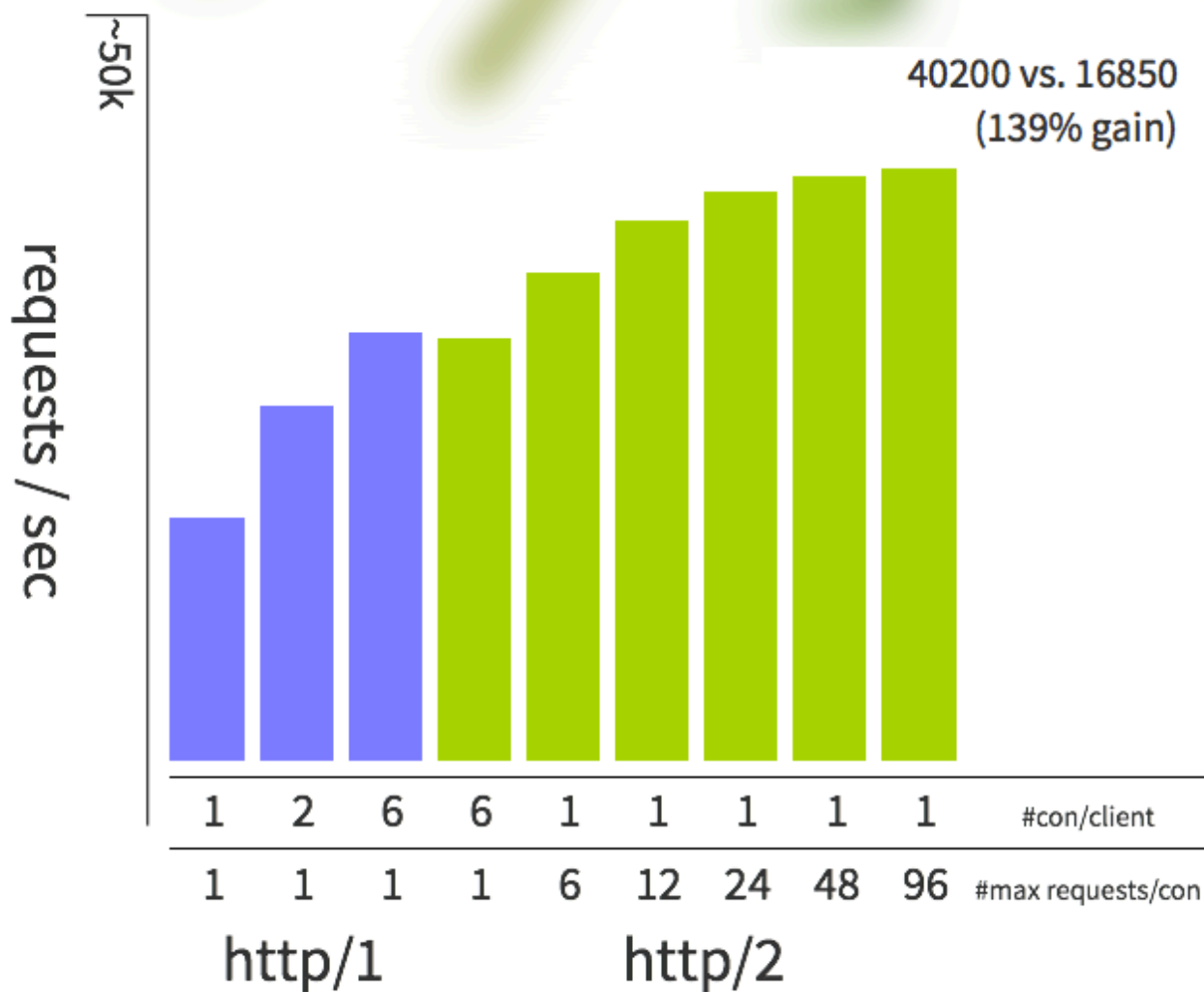| Europe	US	Japan	Australia	
35	120	270	380	RTT (ms)
(0)	(50)	(60)	(120)	phys. Limit

1 html, 5 css, 5 js, 6 img, no server push

How?

Latency Influence:

- 17 resources are retrieved
- http/1 uses 6 connections
- 17/6 = 2.83, so 3 x RTT
- + ~1 RTT for 1st resource
- Server Push will reduce by ~.5 RTT

Local Bliss

HTTP with ~0 Latency

2k resource, retrieved 500k times from localhost by 4 clients

How?

Influence of Parallel Requests:

- 4 clients on 8 core system
- A game of Ping Pong
- http/1: 1 req/conn, 4*6 conn => max: 24
- http/2: 96 req/conn, 4*1 conn => max: 384

A game with 24 vs 384 ping pong balls...

httpd/mod_h2 Implementation

- Not What It Used to be
- HTTP/2 Architecture
- Inheritance
- The New Kid
- Hooked
- Configured

Not What It Used to be

Processing a h2 connection is vastly different in terms of (Connection/Request/Thread).
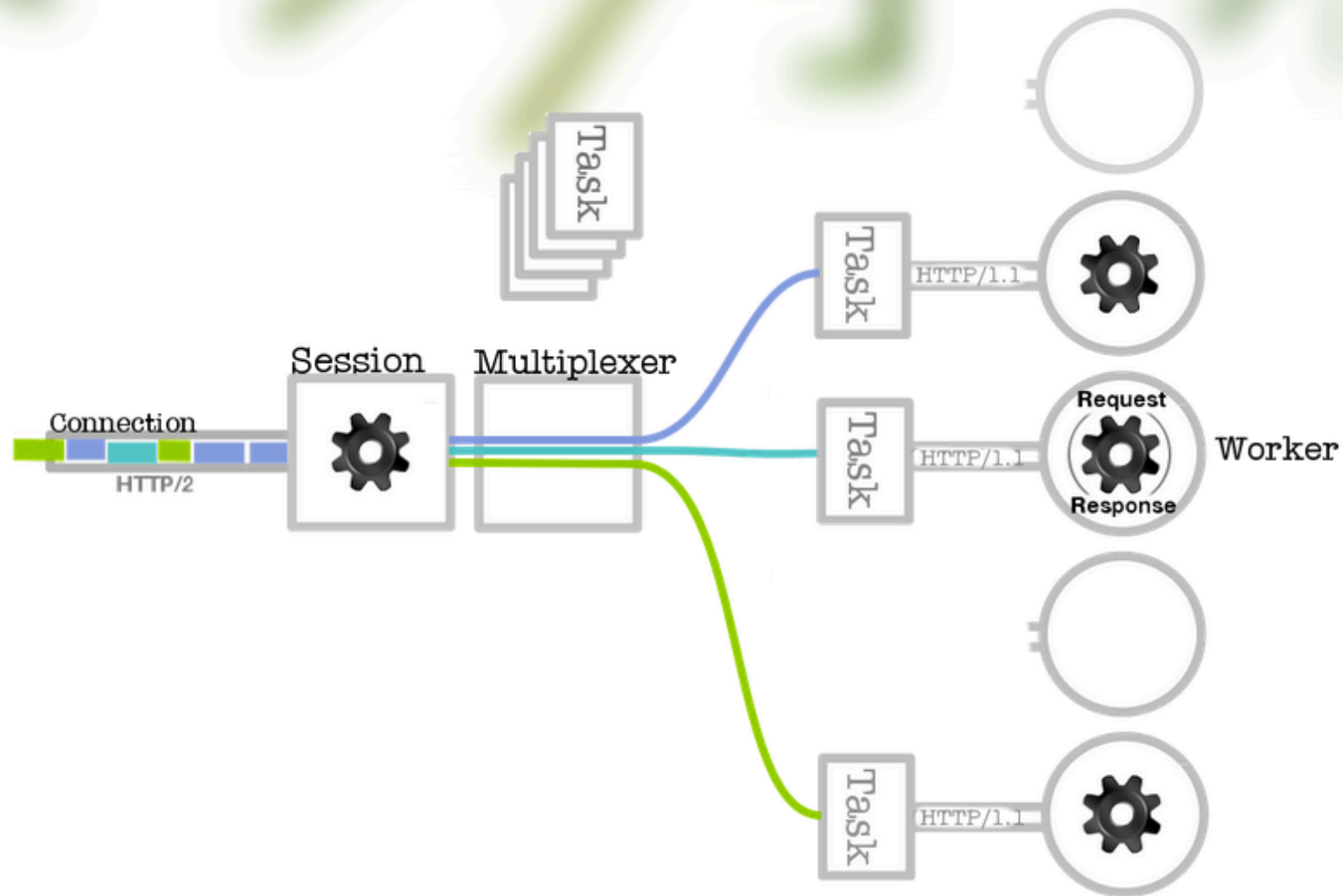
- $1-1-1$ (http/1)
- $1-n-m$ (h2)

In h2, 1 connection has n ongoing request, worked on by m threads.

Existing runtimes are optimized for 1-1-1

HTTP/2 Architecture

Inheritance

mod_spdy

- C/C++ hybrid SPDY engine
- Apache httpd 2.2 pluggable
- Solved many plumbing details
- Honored Ancestor...

Pioneered by Matthew Steele and Bryan McQuade!

The New Kid

mod_h2

- Completely C, written from scratch
- Dependency: nghttp2 only
- github.com/icing/mod_h2
- Release Planned in 2.4.18

Donated by GSMA and greenbytes in July

Hooked

- HTTP/2 Stream Processing
  - process_connection (processing)
  - pre_connection (input_filter)
  - post_read_request (output filters)
- ALPN/Upgrade
  - protocol_propose
  - protocol_switch
  - protocol_get

Configured

Things to observe:

- Needs enable via `Protocols`
- ALPN-able SSL Library needed
- SSL Ciphers set properly
- No Renegotiations!
- Not all modules compatible (mod_logio)

HTTP/mod_h2 Future

- Growing Up
- VPNs and Services
- Blind Caches

"You're off the edge of the map, mate. Here there be monsters."

Growing Up

Full list at github icing/mod_h2

- Server Push
- MPM Integration, Ressources
- Performance, Master/Slave Connections
- HTTP/2 in Backend
- HTTP/2 Extension API

VPNs and Services

- Virtual Private Networking over HTTP/2 (Nottingham)
- Opportunistic Security for HTTP (Nottingham/Thomson)
- HTTP Alternative Services (Nottingham/McManus/Reschke)
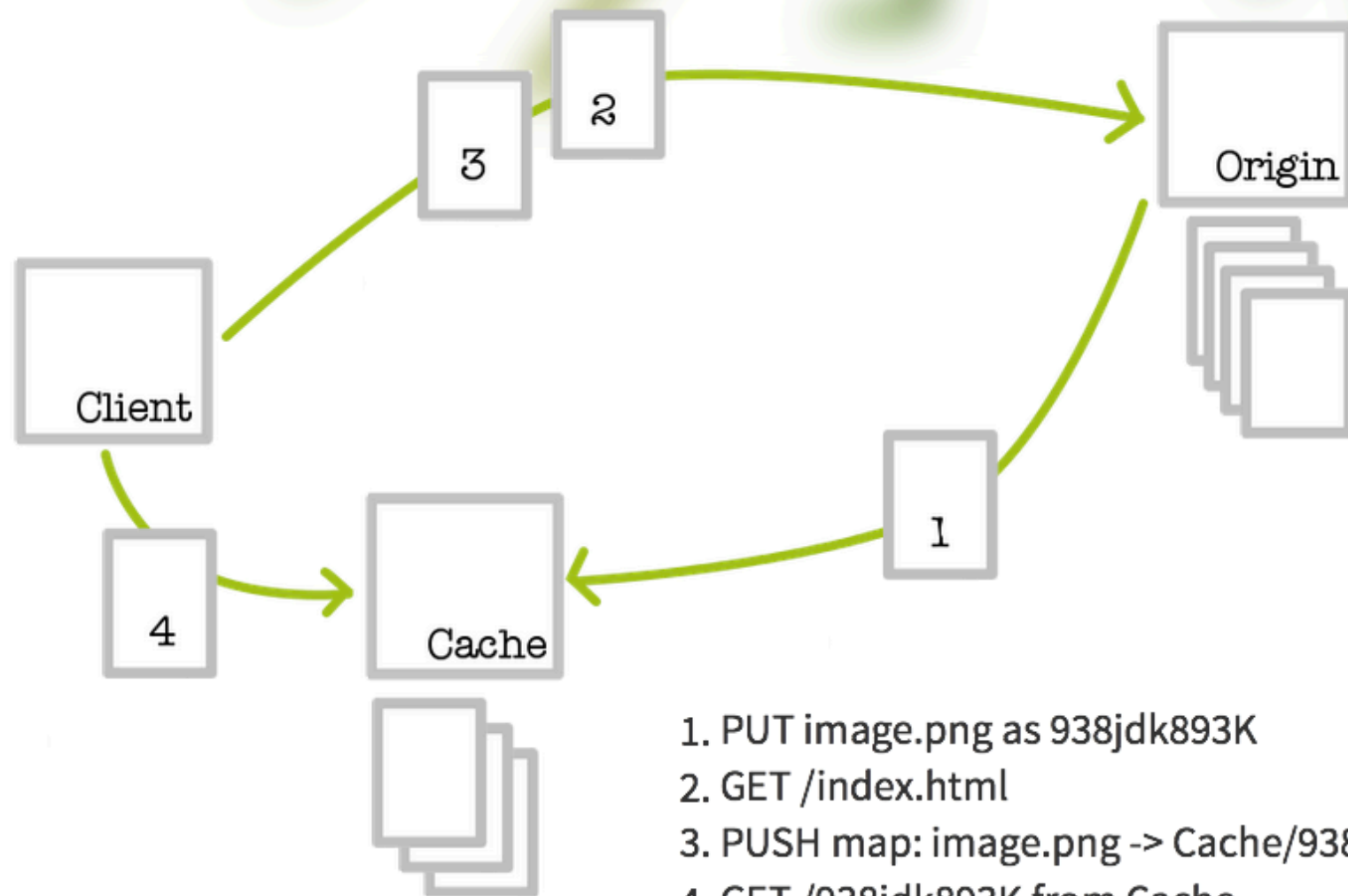- DNSCrypt by Frank Denis

Blind Cache

- Idea Proposed by Martin Thomson (Mozilla)
- Caches in "HTTPS Everywhere Land":
    - Without Middlemen
    - TLS Secured Communication
    - Caching Chunks of Bytes

Ericcson Research working on a Draft, greenbytes consulting

Blind Cache, Principle

1. PUT image.png as 938jdk893K
2. GET /index.html
3. PUSH map: image.png -> Cache/938jdk893K
4. GET /938jdk893K from Cache

`httpd+h2`

Our prayers go to `httpd 2.4.18`!
Experimental at github.com/icing/mod_h2

Thanks!