

Web Caching: Moving Away From a Binary Decision

Benjamin Wollmer

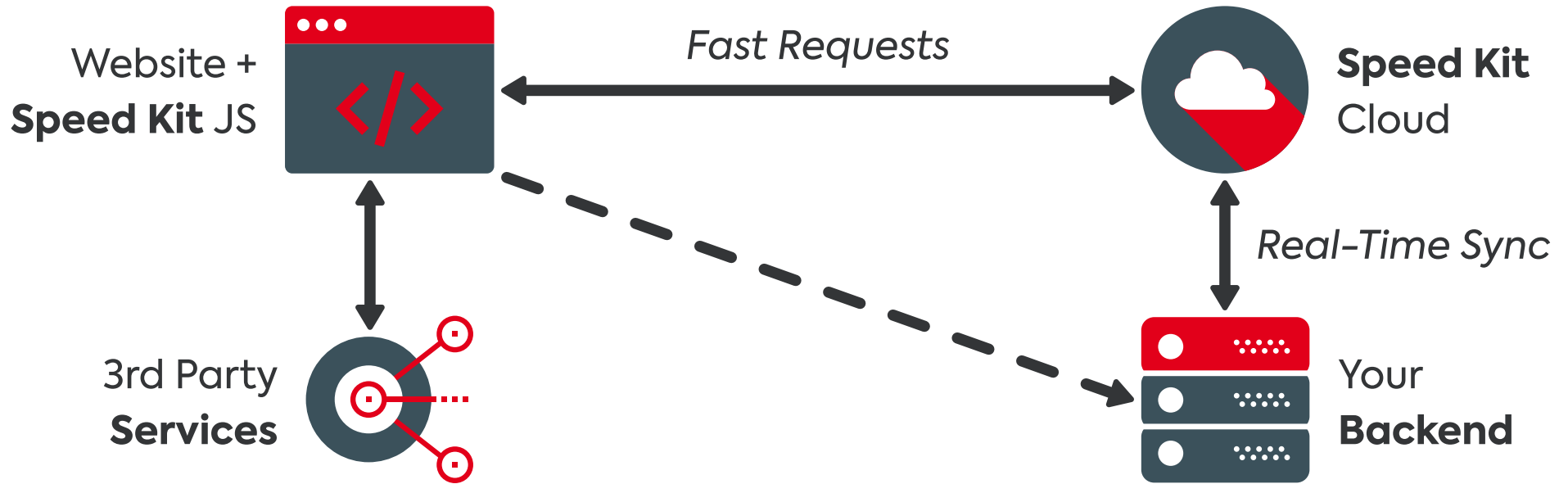
Universität Hamburg

20 February 2020

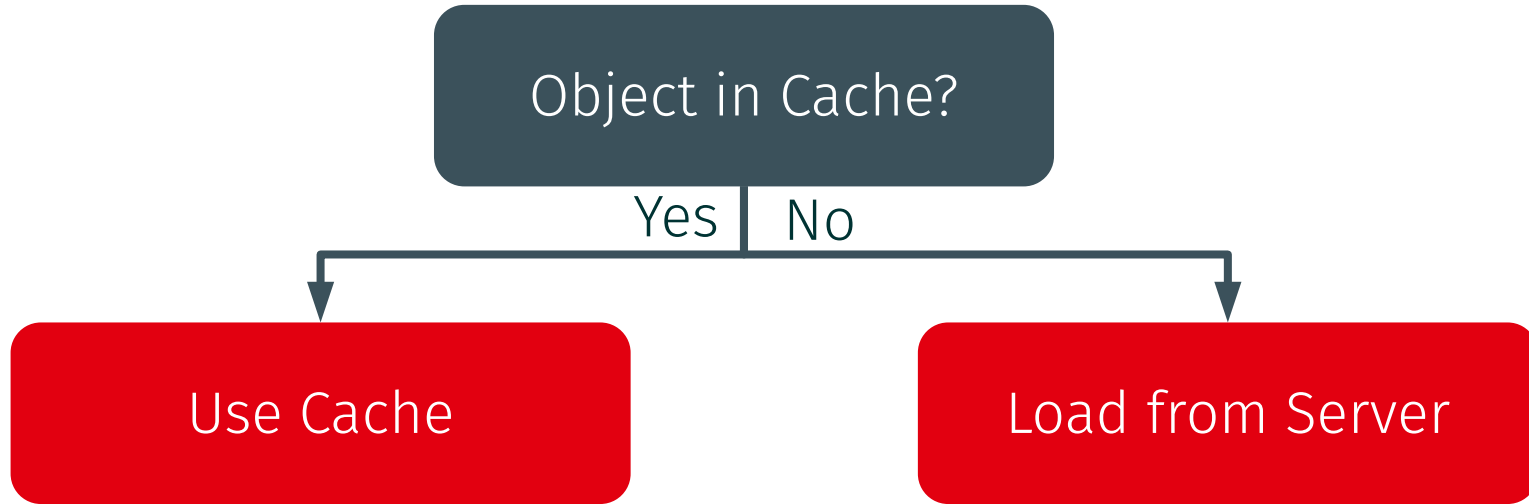
The Business Impact of Site Speed



Speed Kit



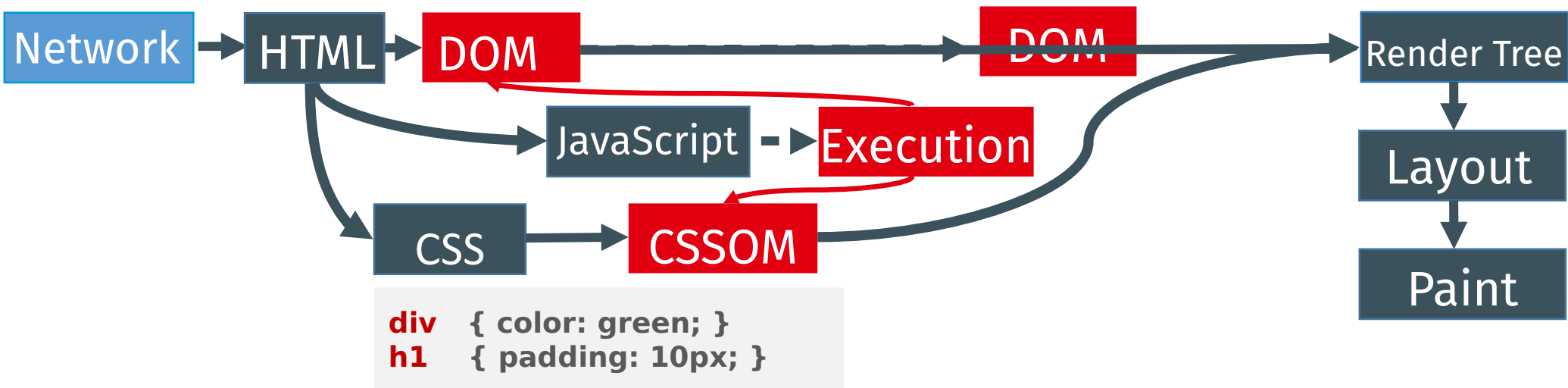
A Binary Decision...



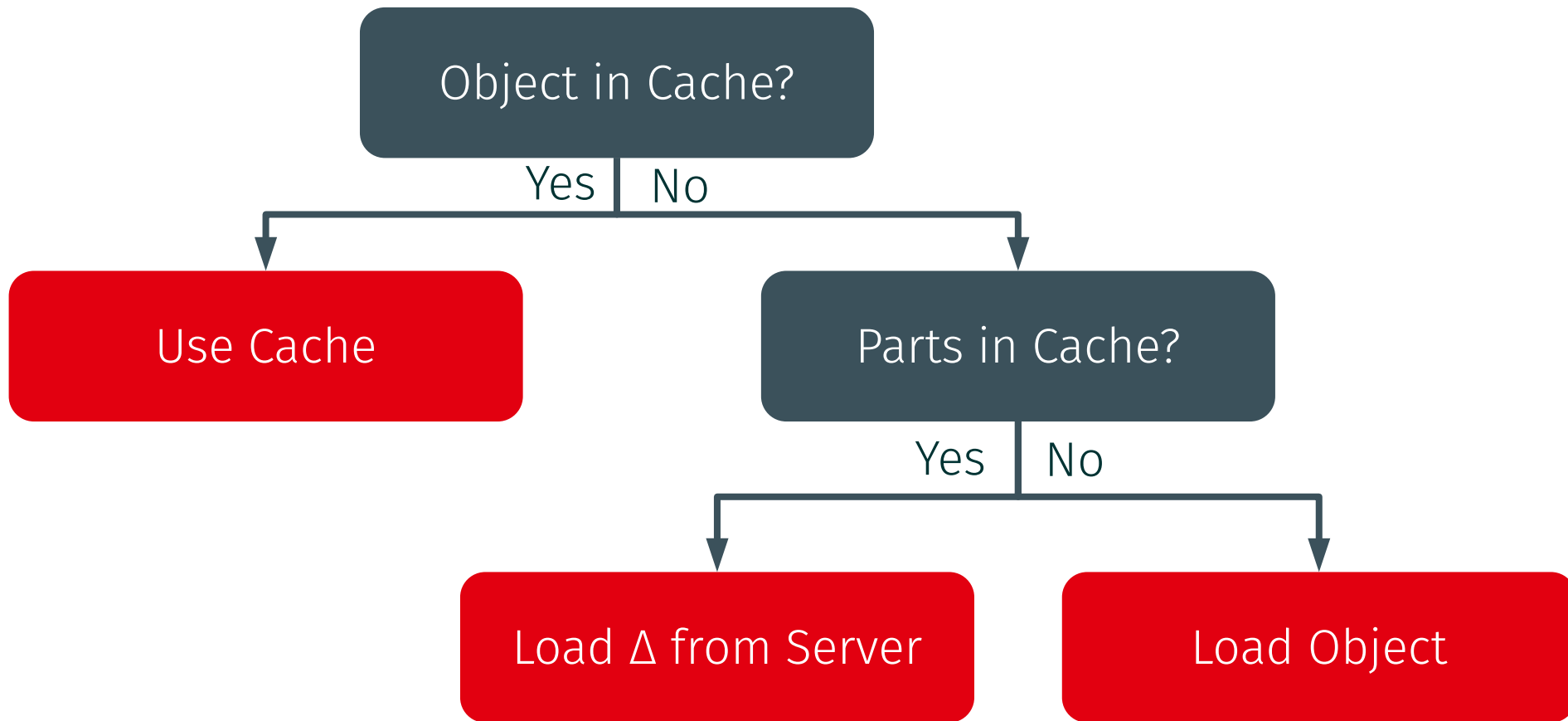
The Critical Rendering Path

```
<!doctype html>  
<title>Code Talks</title>  
<link href=all.css rel=stylesheet />  
<script src=app.js ></script>  
<div>  
  <h1>Web Performance</h1>  
</div>
```

```
<script>  
  elem.style.width = "50px";  
  document.write("Some Text!");  
</script>
```



... Partially Solved



Potential Partial Caches

Potential benefits of delta encoding and data compression for HTTP

Jeffrey C. Mogul (Digital Equipment Corporation Western Research Laboratory)
250 University Avenue, Palo Alto, CA 94301; mogul@wrl.dec.com

Fred Douglass, Anja Feldmann, Balachander Krishnamurthy (AT&T Labs - Research)
180 Park Avenue, Florham Park, NJ 07932-0971; {douglass,anja,bala}@research.att.com

Abstract

Caching in the World Wide Web currently follows a naive model, which assumes that resources are referenced many times between changes. The model also provides no way to update a cache entry if a resource does change, except by transferring the resource's entire new value. Several previous papers have proposed updating cache entries by transferring only the differences, or "delta," between the cached entry and the current value.

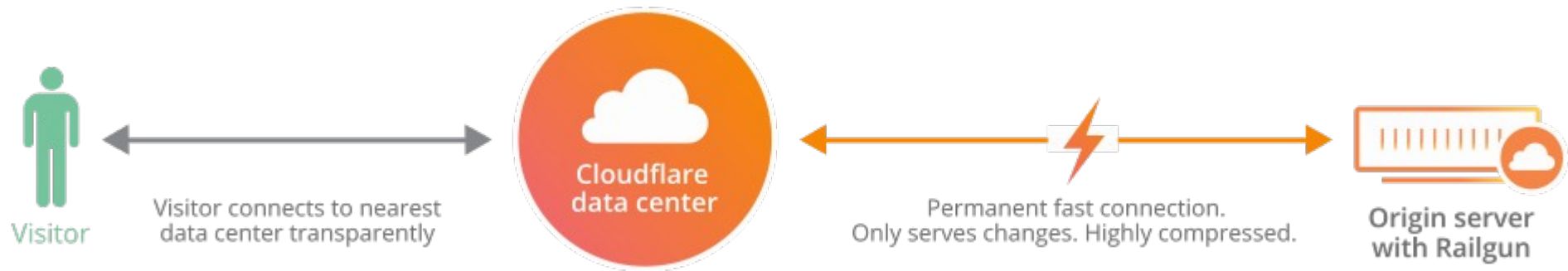
In this paper, we make use of dynamic traces of the full contents of HTTP messages to quantify the potential benefits of delta-encoded responses. We show that delta encoding can provide remarkable improvements in response size and response delay for an important subset of HTTP content types. We also show the added benefit of data compression, and that the combination of delta encoding and data compression yields the best results.

retrieval. Upon receiving a conditional request, the server may either reply with a full response, or, if the resource has not changed, it may send an abbreviated reply, indicating that the client's cache entry is still valid. HTTP/1.0 also includes a means for the server to indicate, via an "expires" timestamp, that a response will be valid until that time; if so, a client may use a cached copy of the response until that time, without first validating it using a conditional retrieval.

The proposed HTTP/1.1 specification [6] adds many new features to improve cache coherency and performance. However, it preserves the all-or-none model for responses to conditional retrievals: either the server indicates that the resource value has not changed at all, or it must transmit the entire current value.

Common sense suggests (and traces confirm), however, that even when a Web resource does change, the new instance is often substantially similar to the old one. If the difference (or *delta*) between the two instances could be sent to the client

Delta Encoding at Cloudflare




Entity Transition

PRESSE KUS-PORTAL STINE PRESSE KUS-PORTAL STINE EINRICHTUNGEN UND ZIELGRUPPEN LANGUAGE

UHH Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

STUDIUM FORS



CAMPUS-CENTER

UHH → Campus-Center → **Campus-Leben**

CAMPUS-LEBEN

Was macht die Uni Hamburg aus?


Eine Universität bedeutet nicht nur Studieren. Beim Studium geht es auch darum, Mensc treffen, gemeinsam Spaß zu haben, sich weiterzuentwickeln. Und der Campus ist der Ort, all das passiert.

Das Leben auf dem Campus hat seinen eigenen Rhythmus, den die Vorlesungen und Sem prägen. Aber es bleibt immer auch Zeit für Besuche in der Mensa oder in den Cafés. Und a musische Interessen kommen nicht zu kurz: Orchester und Chor, Theatergruppen und übe Sportarten stehen allen Studierenden offen.

Zum Studierendenleben gehört aber auch soziales oder politisches Engagement. Es gibt d unterschiedlichsten Gruppen, in denen man sich engagieren und etwas Positives bewirke

UHH Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

STUDIUM **FORSCHUNG** INTERNATIONALES UNIVERSITÄT



1919 2019
100 JAHRE
WISSENSWERT
Universität Hamburg

Foto: UHH/KlimaCampus/Ausserhofer

UHH → Forschung → **Forschungsprofil**

FORSCHUNGSPROFIL

Die Universität Hamburg ist die größte und vielfältigste Forschungseinrichtung Norddeutschlands. Ihr Forschungsprofil umfasst exzellente Grundlagenforschung genauso wie anwendungsnahe Forschungs- und Transferprojekte. Die Forschung wird stark geprägt durch **fünf Forschungsschwerpunkte** und **fünf Potenzialbereiche**.

Die Universität Hamburg ist sehr stark in der Drittmittelerwerbun, sowohl in der Einzelforschung wie auch der kooperativen Forschung. Neben vier **Exzellenzclustern**, an denen geforscht wird, sind wir derzeit an **17 Sonderforschungsbereichen**, **20 DFG-Forschungsgruppen** und **40 DFG-Schwerpunktprogrammen** beteiligt und übernehmen in der Mehrzahl die Sprecher/innen-Rolle.

In der Nachwuchsförderung qualifizieren wir junge Wissenschaftler/innen in **DFG-Graduiertenkollegs**, zahlreichen **Nachwuchsgruppen** und durch Doktorandenprogramme in Kooperation u.a. mit **außeruniversitären Forschungseinrichtungen**.

Forschungsprofil

Neues aus der Forschung

Forschungsschwerpunkte, Potenzialbereiche und Profilinitiativen

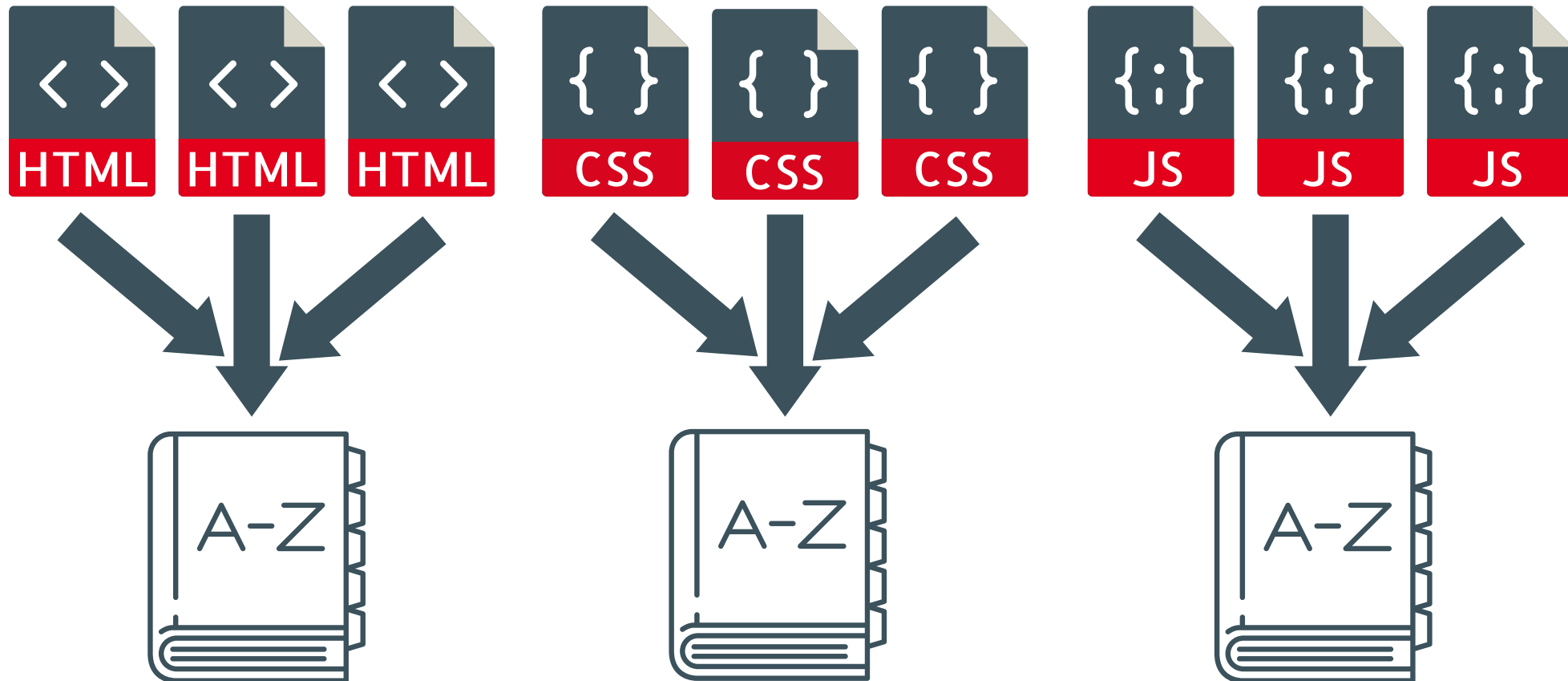
Exzellenzcluster

Forschungsprojekte

Forschungszentren

-

Shared Compression



A Proposal for Shared Dictionary Compression over HTTP

Authors: Jon Butler, Wei-Hsin Lee, Bryan McQuade, Kenneth Mixer
Google, Inc.
Last Update : September 8, 2008

Abstract

This paper proposes an HTTP/1.1-compatible extension that supports inter-response data compression by means of a reference dictionary shared between user agent and server.

- 1282 CSS Files
- 6225 JS Files
- 81% better compression
- 400 μ s encoding

Shared Dictionary Compression for HTTP at LinkedIn.



Omer Shapira March 4, 2015



HTTP protocol has been the glue that holds together the Web, mobile apps and servers. In the world of constant innovation, HTTP/1.1 appeared to be the only safe island of consistency until the announcement of HTTP/2. Yet, even with HTTP being as robust and efficient as it is, there is still room for improvement, and this is what this post is about. LinkedIn's Traffic Infrastructure team is making LinkedIn faster by exploring ways in which HTTP can be improved.

Traditionally, HTTP communications are compressed using either [gzip](#) or [deflate](#) algorithms. Of the two, gzip strikes the balance of being aggressive enough to reach a good compression ratio, while not having ridiculous CPU requirements. Algorithms such as [Burrows-Wheeler transform](#) (popularized through bzip2) offer higher degrees of compression, but have higher CPU requirements. Until recently it was generally accepted that gzip is the best way to compress HTTP traffic. Yet there is an alternative to using computationally intense compression algorithms to transfer less data over the wires - this alternative is to start with sending less data in the first place.

Enter SDCH

SDCH (pronounced "Sandwich") is an HTTP/1.1-compatible extension, which reduces the required bandwidth through the use of a dictionary shared between the

SDCH Already Unshipped

[blink-dev](#) >

Intent to Unship: SDCH

42 Einträge von 17 Autoren ▾



Ryan Sleevi

22.11.16



Andere Empfänger: rds...@chromium.org

[Nachricht auf Deutsch übersetzen](#)

Primary Eng:

rs...@chromium.org

Summary

Move SDCH behind an experimental flag until the implementation is staffed, the specification matures, and broader consensus emerges. If those don't happen, remove support from code entirely.

Motivation

Since its first release, Chromium has supported SDCH, an experimental compression protocol proposed in 2008. [1] Unfortunately, since this original proposal, few non-Chromium browsers have adopted support for this and it has seen limited standards activity or cross-browser interest.

As the original proposal had IPR concerns [2] that prevented it from being standardized, a new version has been submitted to the IETF [3]. While the current I-D is as an Individual submission, the intent is to work with other interested vendors, such as Yandex and LinkedIn, on standardizing this effort. This was presented at IETF97 and

Brotli

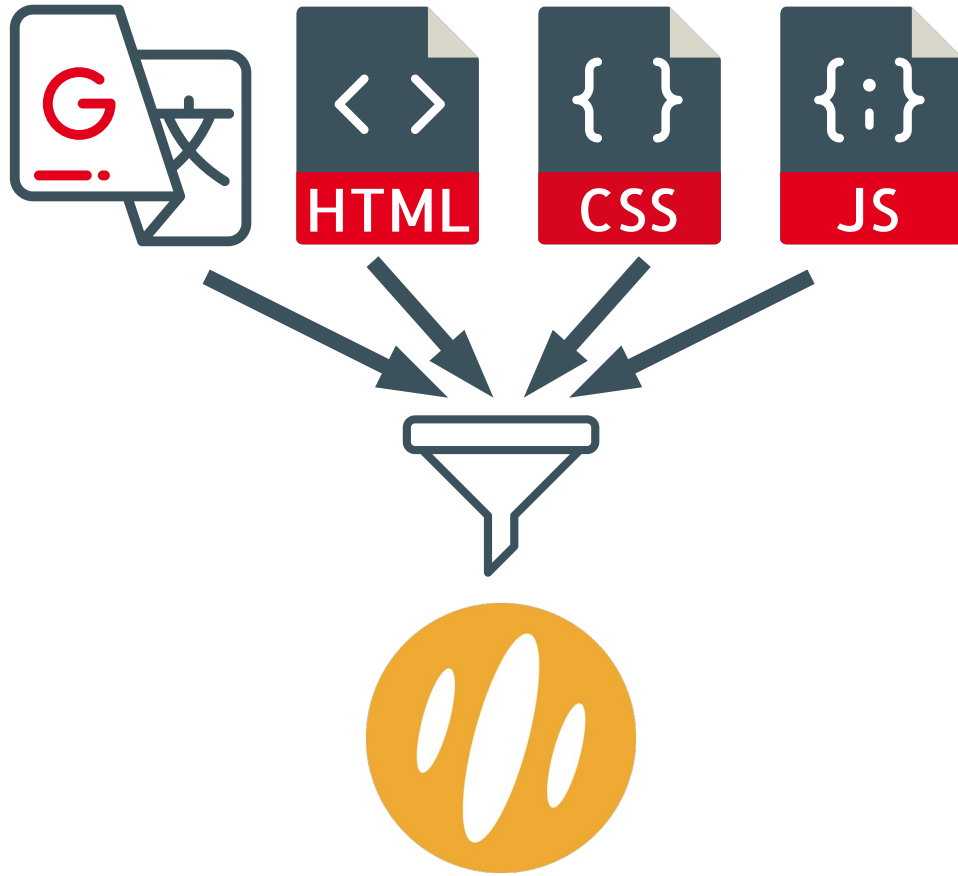


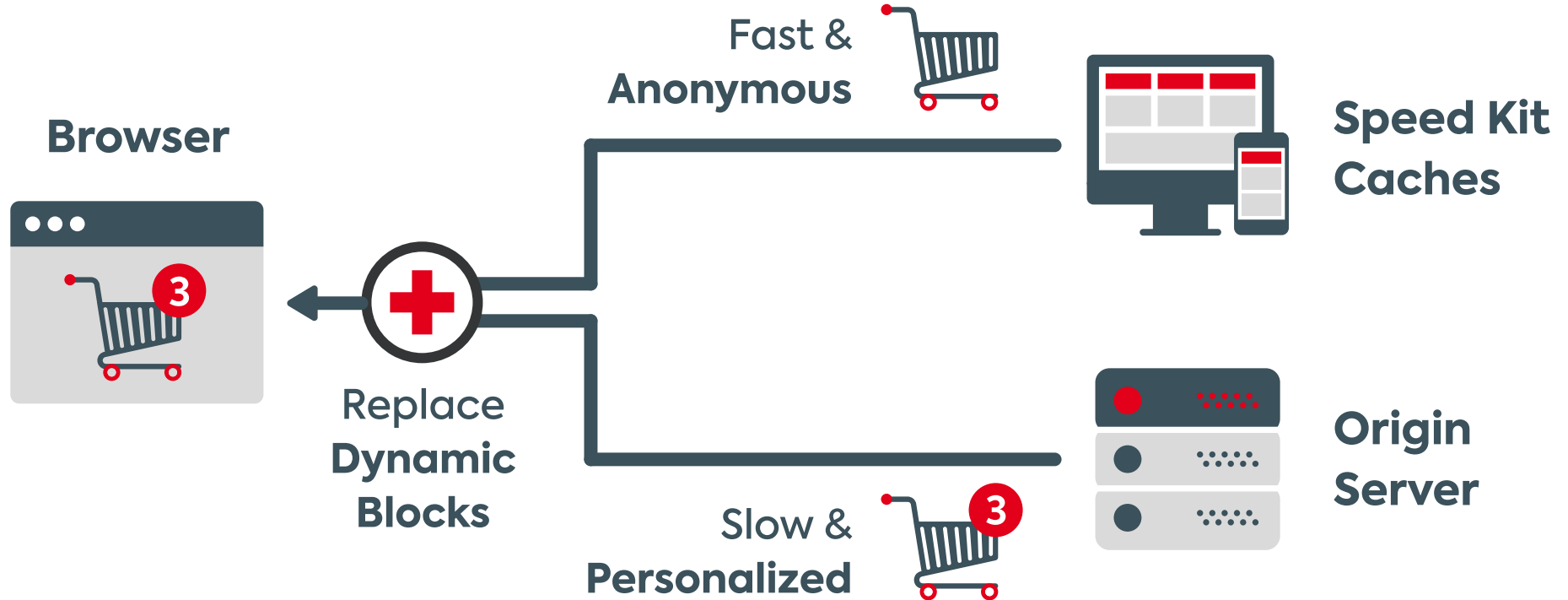
Image Variations



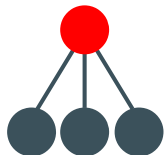
Image Variations cont.



Dynamic Blocks (Speed Kit)



Conclusion & Goals



Unified

Combine compression methodologies



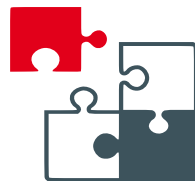
End to End

Control over the whole architecture to benefit end users



Adaptive

Autonomously choose protocol given runtime context



Opt-in

Pluggable and compatible with legacy tech stacks

How can partial caching and compression methods be used to accelerate data access in a distributed architecture with heterogeneous clients and servers?

Progress & Next Steps

Speed Kit: A Polyglot & GDPR-Compliant Approach For Caching Personalized Content

Wolfram Wingerath*, Felix Gessert*, Erik Witt*, Hannes Kuhlmann*,
Florian Bücklers*, Benjamin Wollmer[†], and Norbert Ritter[†]

**Baqend GmbH*, Stresemannstraße 23, 22769 Hamburg, Germany
{ww, fg, ew, hk, fb}@baqend.com

[†]*University of Hamburg*, Databases and Information Systems, Vogt-Kölln-Straße 30, 22527 Hamburg, Germany
{wollmer, ritter}@informatik.uni-hamburg.de

Abstract—Users leave when page loads take too long. This simple fact has complex implications for virtually all modern businesses, because accelerating content delivery through caching is not as simple as it used to be. As a fundamental technical challenge, the high degree of personalization in today's Web has seemingly outgrown the capabilities of traditional content delivery networks (CDNs) which have been designed for distributing static assets under fixed caching times. As an additional legal challenge for services with personalized content, an increasing number of regional data protection laws constrain the ways in which CDNs can be used in the first place. In this paper, we present Speed Kit as a radically different approach for content distribution that combines (1) a polyglot architecture for efficiently caching personalized content with (2) a natively GDPR-compliant client

critical website itself (i.e. the HTML) is typically considered uncacheable due to personalization and therefore delivered by the origin server. The second critical open challenge is related to the **legal ramifications** of using a CDN, since routing all incoming user traffic through it is mandatory for deployment. Since this effectively grants the CDN provider full access to information that is protected by regulations such as the General Data Protection Regulation (GDPR) [15] or the California Consumer Privacy Act of 2018 (CCPA) [9], employing a CDN requires careful consideration to avoid hefty fines [27] in case of non-compliance or data breaches.

To address the above issues, we propose Speed Kit as an

Reevaluate Benefits of Delta Encoding for HTTP

- Things have changed
 - Frameworks
 - Bundling
 - ...

=> Reevaluation necessary

- Alexa top 50 per category (~800 pages)
- 12 times a day
- Focus on text content (html, css, js, svg...)

Potential benefits of delta encoding and data compression for HTTP

Jeffrey C. Mogul (Digital Equipment Corporation Western Research Laboratory)
250 University Avenue, Palo Alto, CA 94301; mogul@wrl.dec.com

Fred Douglass, Anja Feldmann, Balachander Krishnamurthy (AT&T Labs - Research)
180 Park Avenue, Florham Park, NJ 07932-0971; {douglass,anja,bala}@research.att.com

Abstract

Caching in the World Wide Web currently follows a naive model, which assumes that resources are referenced many times between changes. The model also provides no way to update a cache entry if a resource does change, except by transferring the resource's entire new value. Several previous papers have proposed updating cache entries by transferring only the differences, or "delta," between the cached entry and the current value.

In this paper, we make use of dynamic traces of the full contents of HTTP messages to quantify the potential benefits of delta-encoded responses. We show that delta encoding can provide remarkable improvements in response size and response delay for an important subset of HTTP content types. We also show the added benefit of data compression, and that the combination of delta encoding and data compression yields the best results.

retrieval. Upon receiving a conditional request, the server may either reply with a full response, or, if the resource has not changed, it may send an abbreviated reply, indicating that the client's cache entry is still valid. HTTP/1.0 also includes a means for the server to indicate, via an "expires" timestamp, that a response will be valid until that time; if so, a client may use a cached copy of the response until that time, without first validating it using a conditional retrieval.

The proposed HTTP/1.1 specification [6] adds many new features to improve cache coherency and performance. However, it preserves the all-or-none model for responses to conditional retrievals: either the server indicates that the resource value has not changed at all, or it must transmit the entire current value.

Common sense suggests (and traces confirm), however, that even when a Web resource does change, the new instance is often substantially similar to the old one. If the difference (or *delta*) between the two instances could be sent to the client

Thanks!

Benjamin Wollmer
wollmer@informatik.uni-hamburg.de