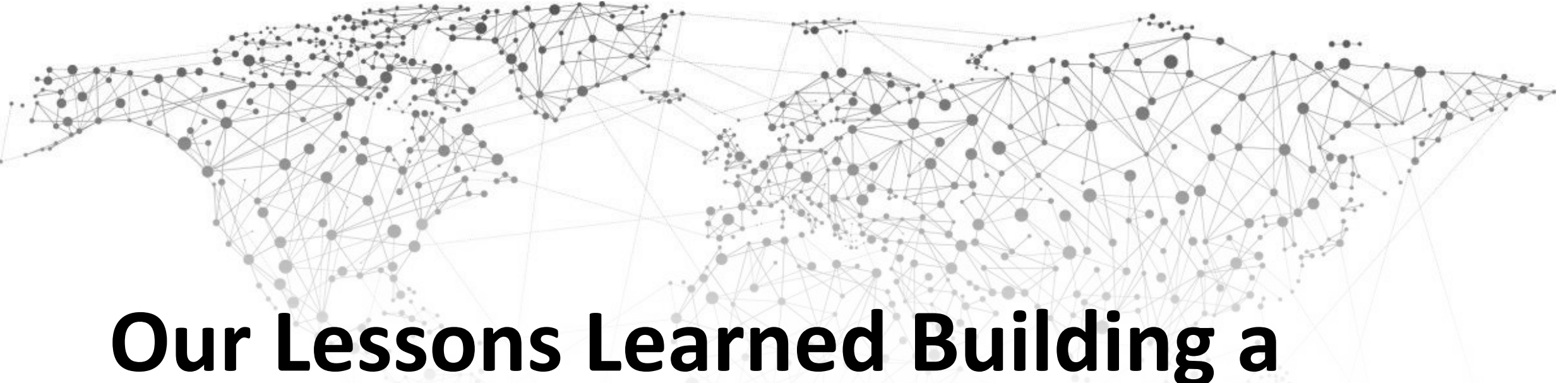




Our Lessons Learned Building a Backend-as-a-Service

A Technical Deep Dive

Felix Gessert

fg@baqend.com

09.05.2017, Netlight



@baqendcom



Web Application Needs

Non-Functional Requirements

“Over 30 percent of web development teams deliver projects late or over-budget.”

Survey by New Bamboo



Modern design



Fast Page Loads



Engaging UX



Automatic scalability



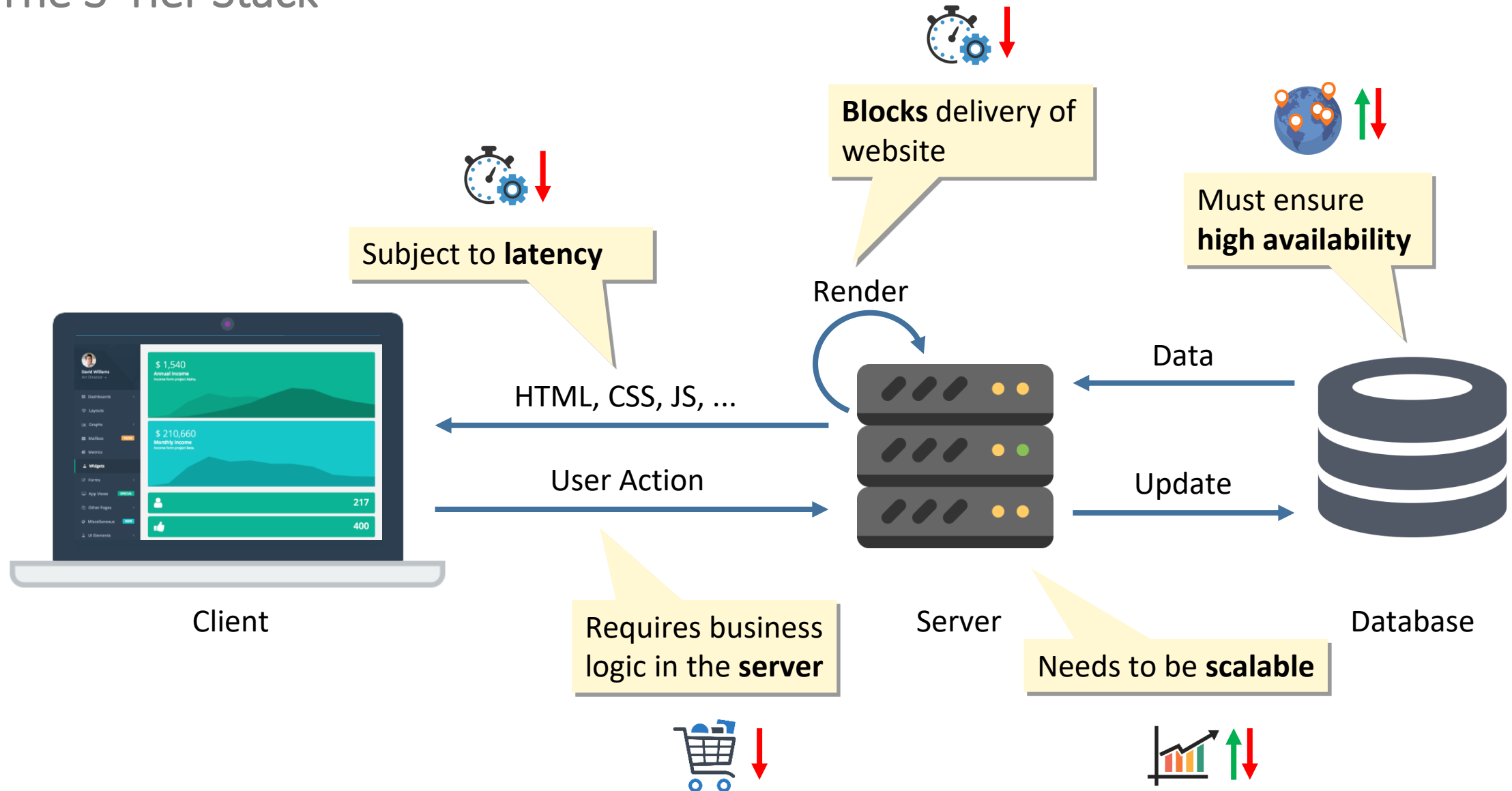
Fast time-to-market



High availability

Common Web Architectures

The 3-Tier Stack



Common Web Architectures

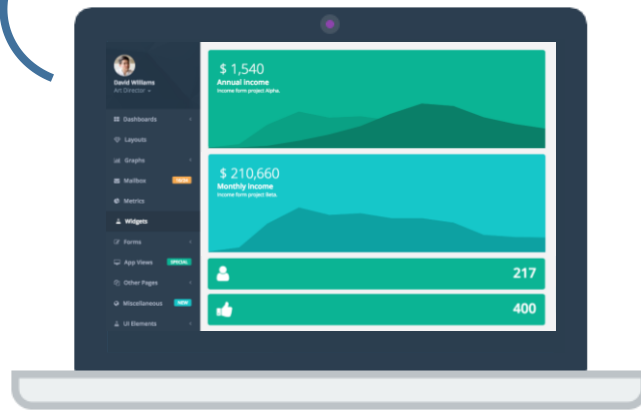
The 2-Tier Stack

Progressive client-side rendering



Render

Client



Still **latency-critical**

HTML, CSS, JS

Data

Update

Database-as-a-service

Cloud DB



Angular



React



Vue.js

Client-side business logic



DynamoDB



Elasticache



MongoDB

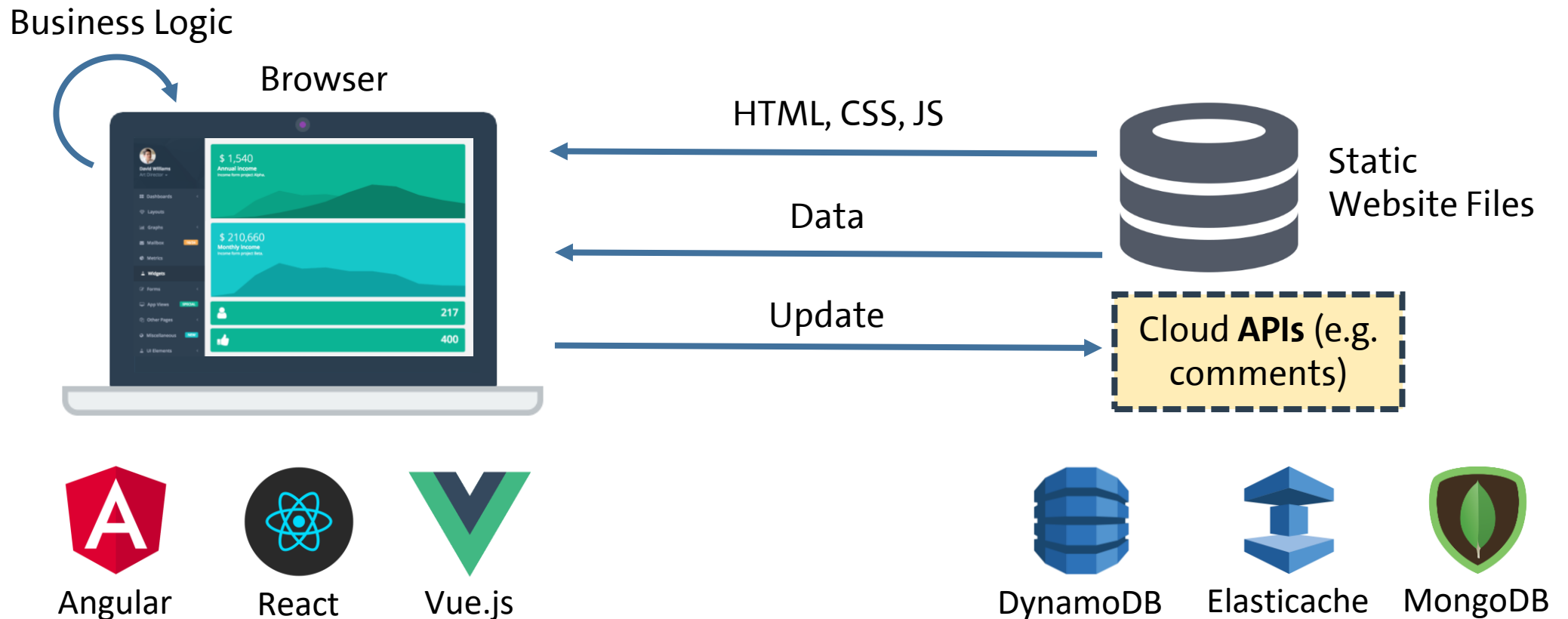
Problem:

- User Management
- Access Control
- Server-side Business Logic
- **Latency** and **scalability** still an issue



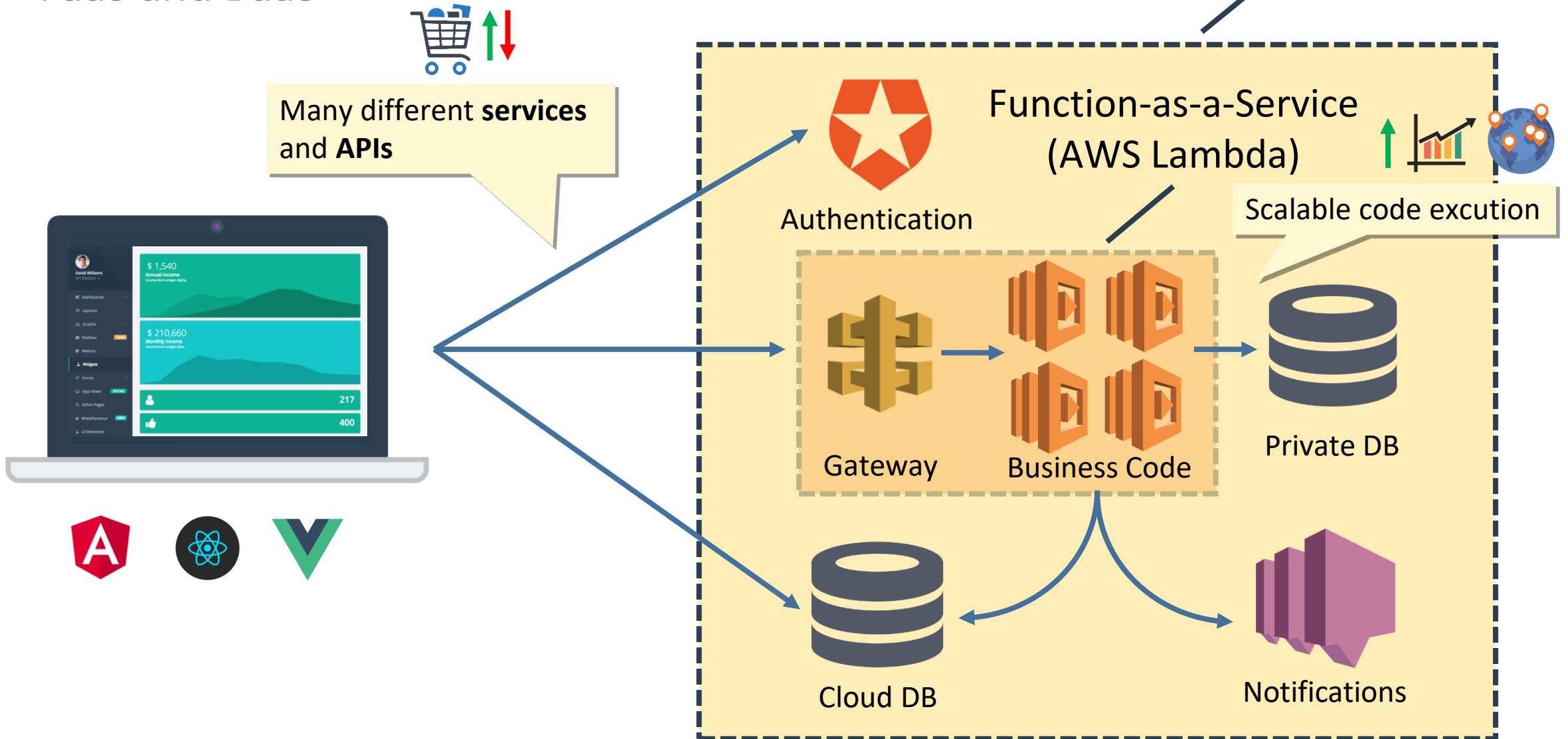
Common Web Architectures

The 2-Tier Stack



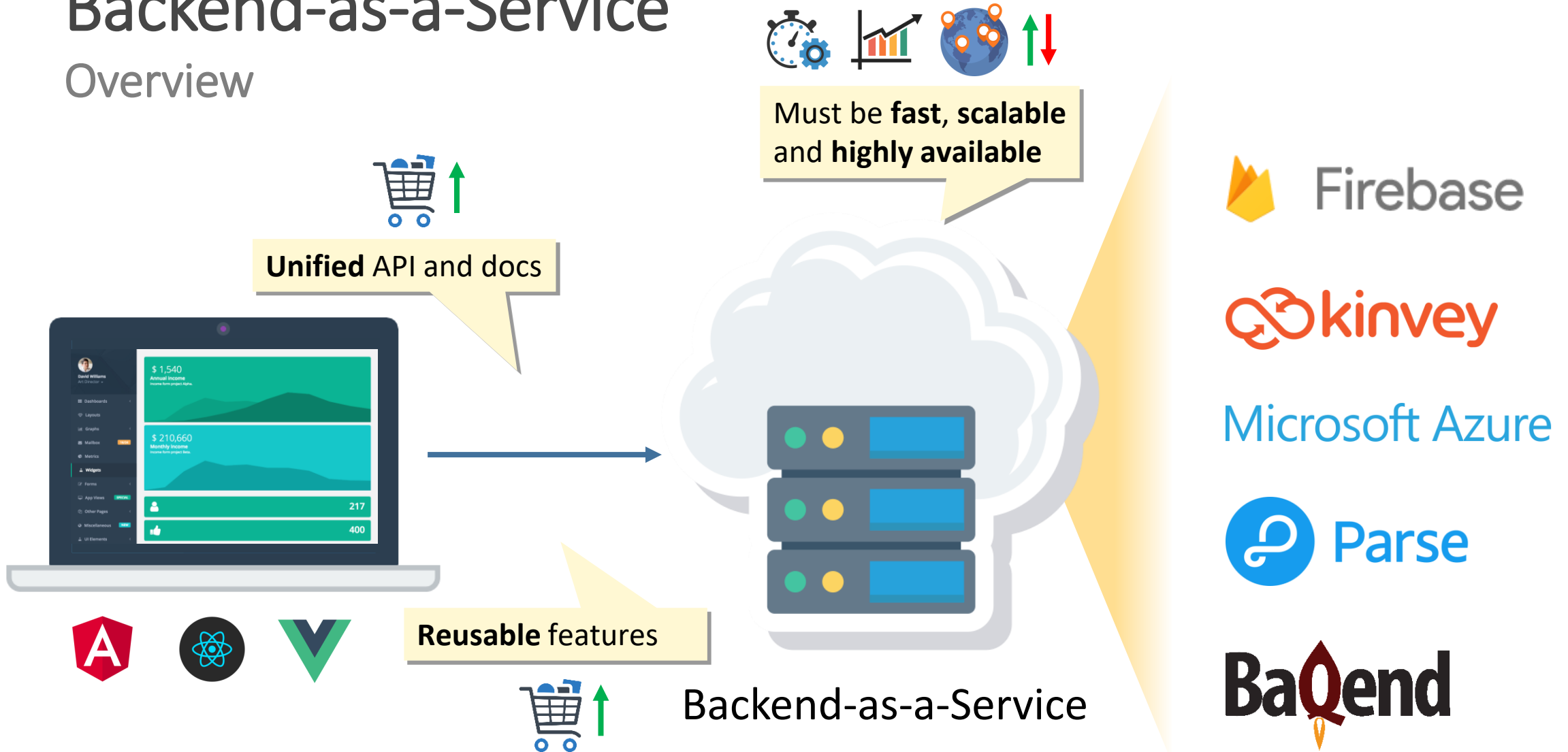
The Serverless Paradigm

FaaS and BaaS



Backend-as-a-Service

Overview

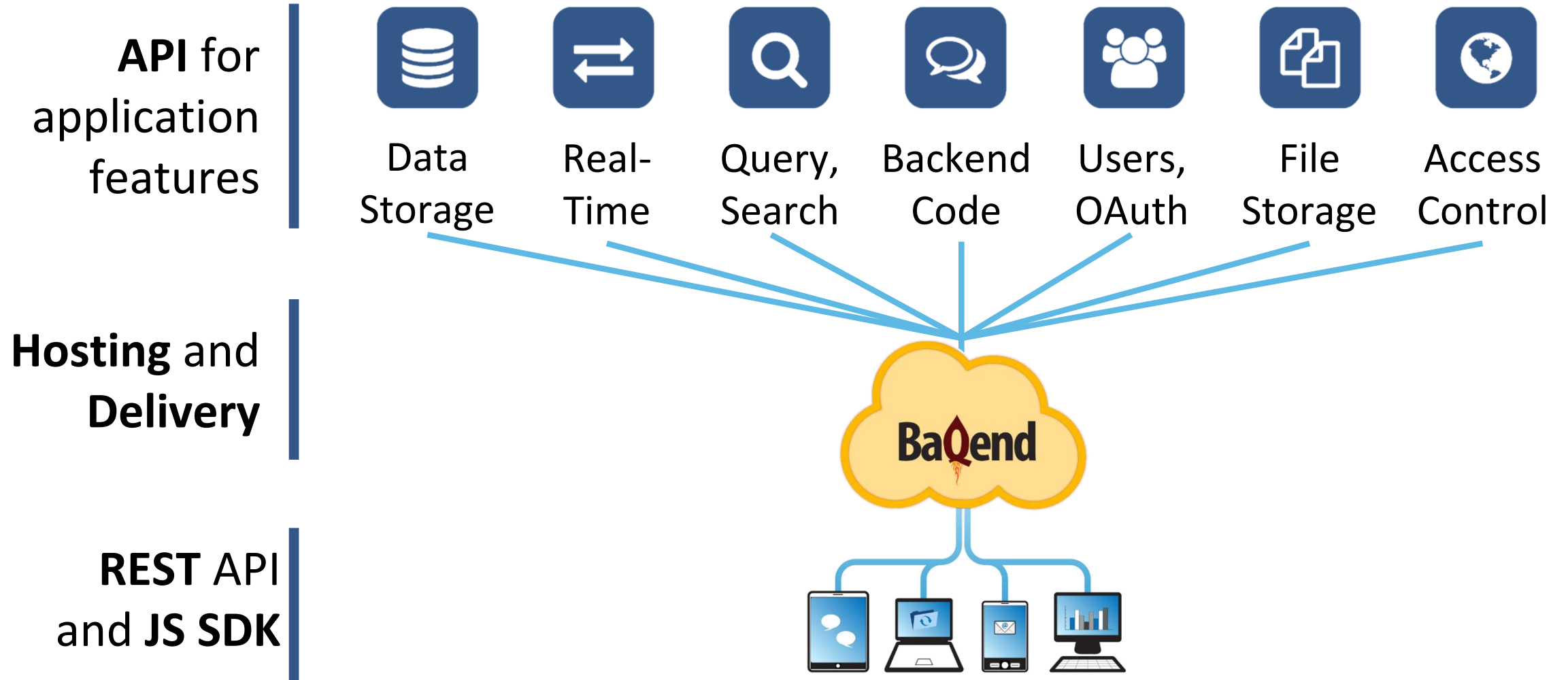


BAAS OVERVIEW

How Backend-as-a-Service works

Backend-as-a-Service

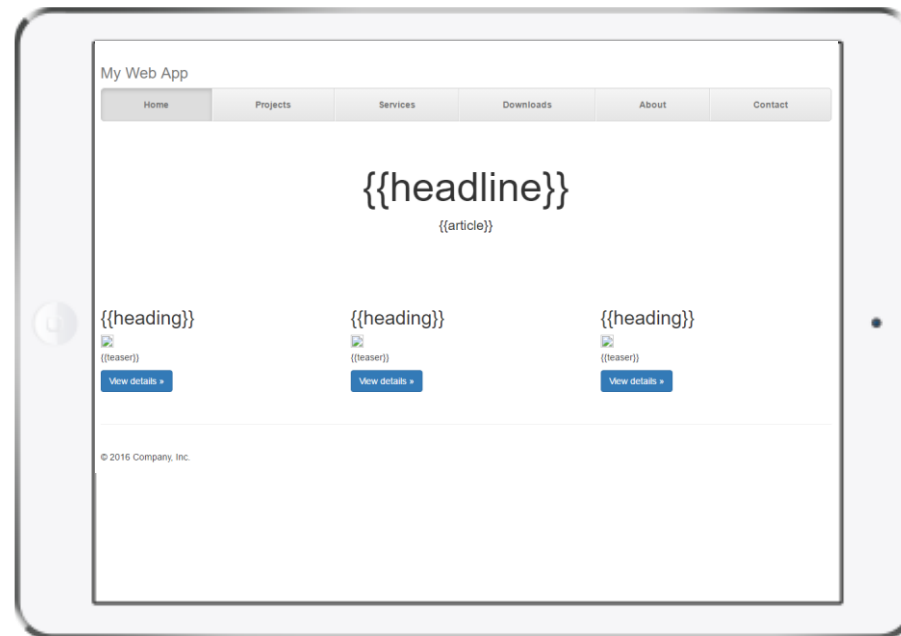
Feature Sets



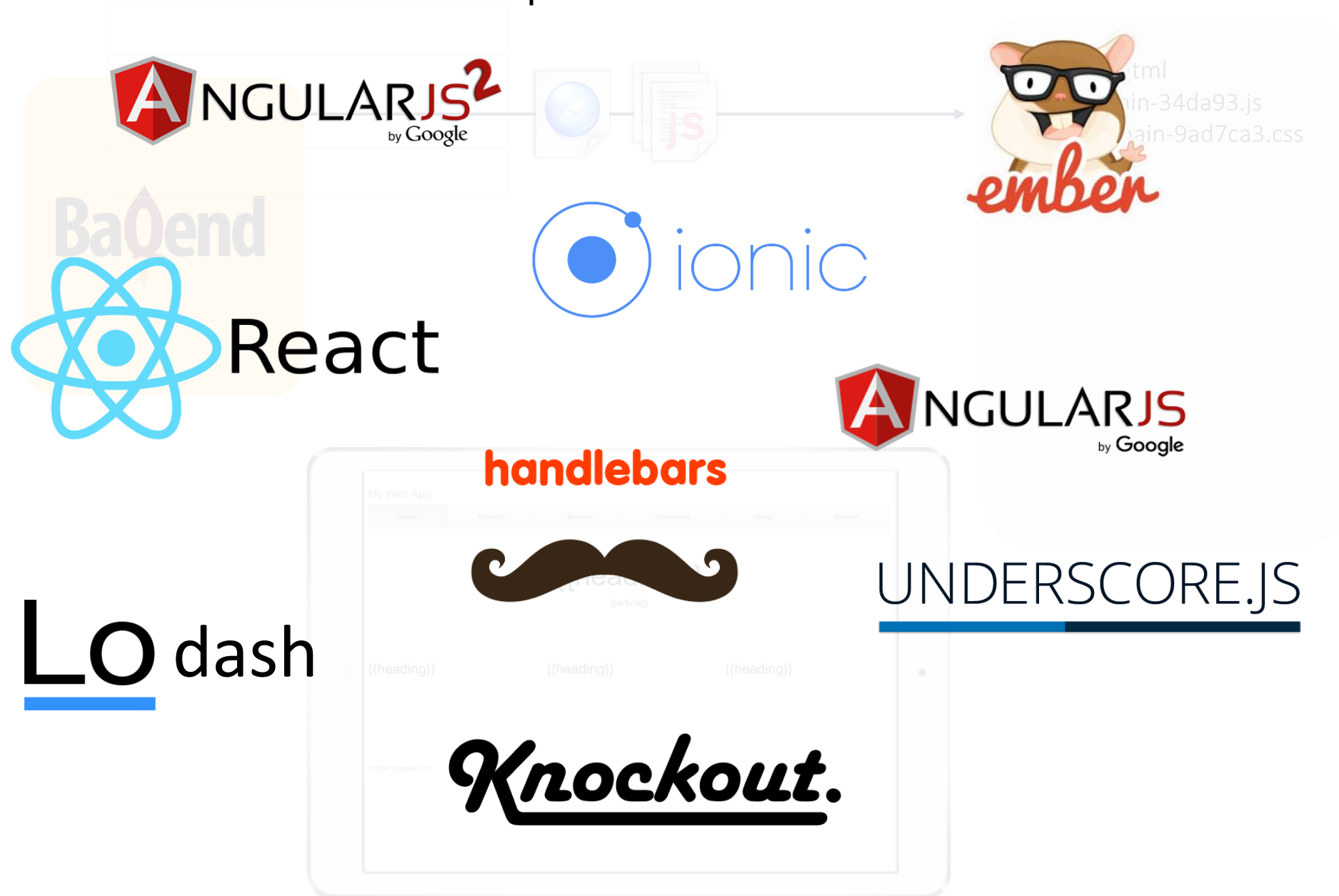
Frontend



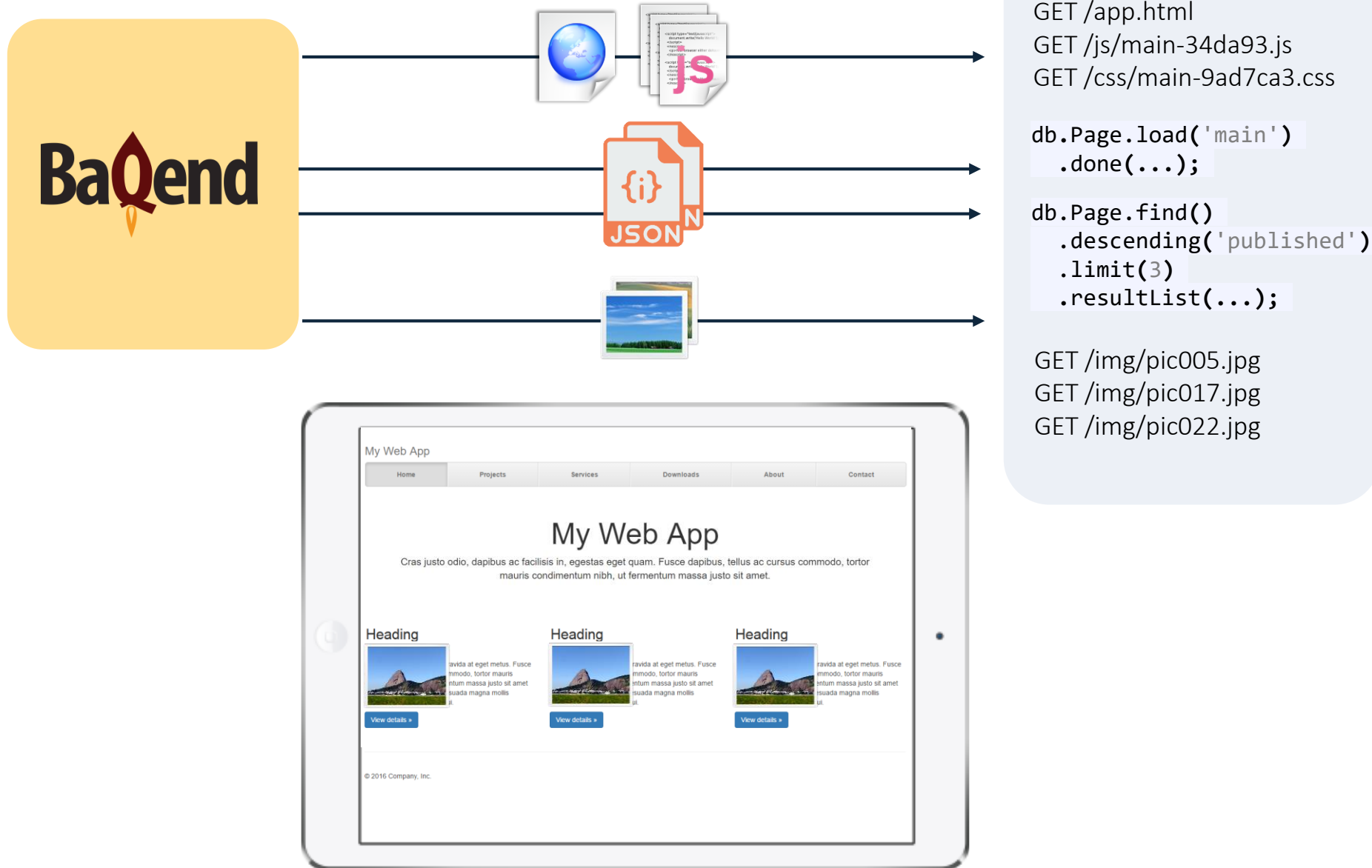
GET /app.html
GET /js/main-34da93.js
GET /css/main-9ad7ca3.css



Compatible with:



Frontend



Backend-as-a-Service

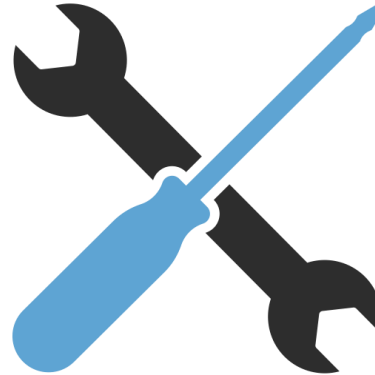
Comparison of Service Models

Backend-as-a-Service



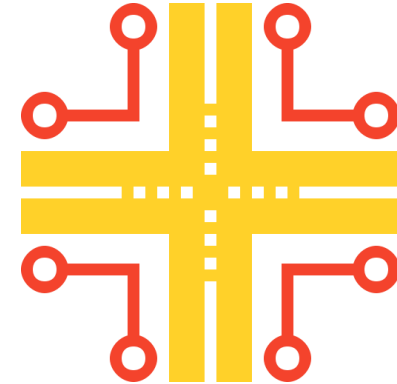
- + Fast time-to-market
- + Low maintenance
- Often limited in expressiveness, scalability or performance

Platform-as-a-Service



- + Full control of backend
- + Tools for deployment and maintenance
- Limited in platform control

Infrastructure-as-a-Service

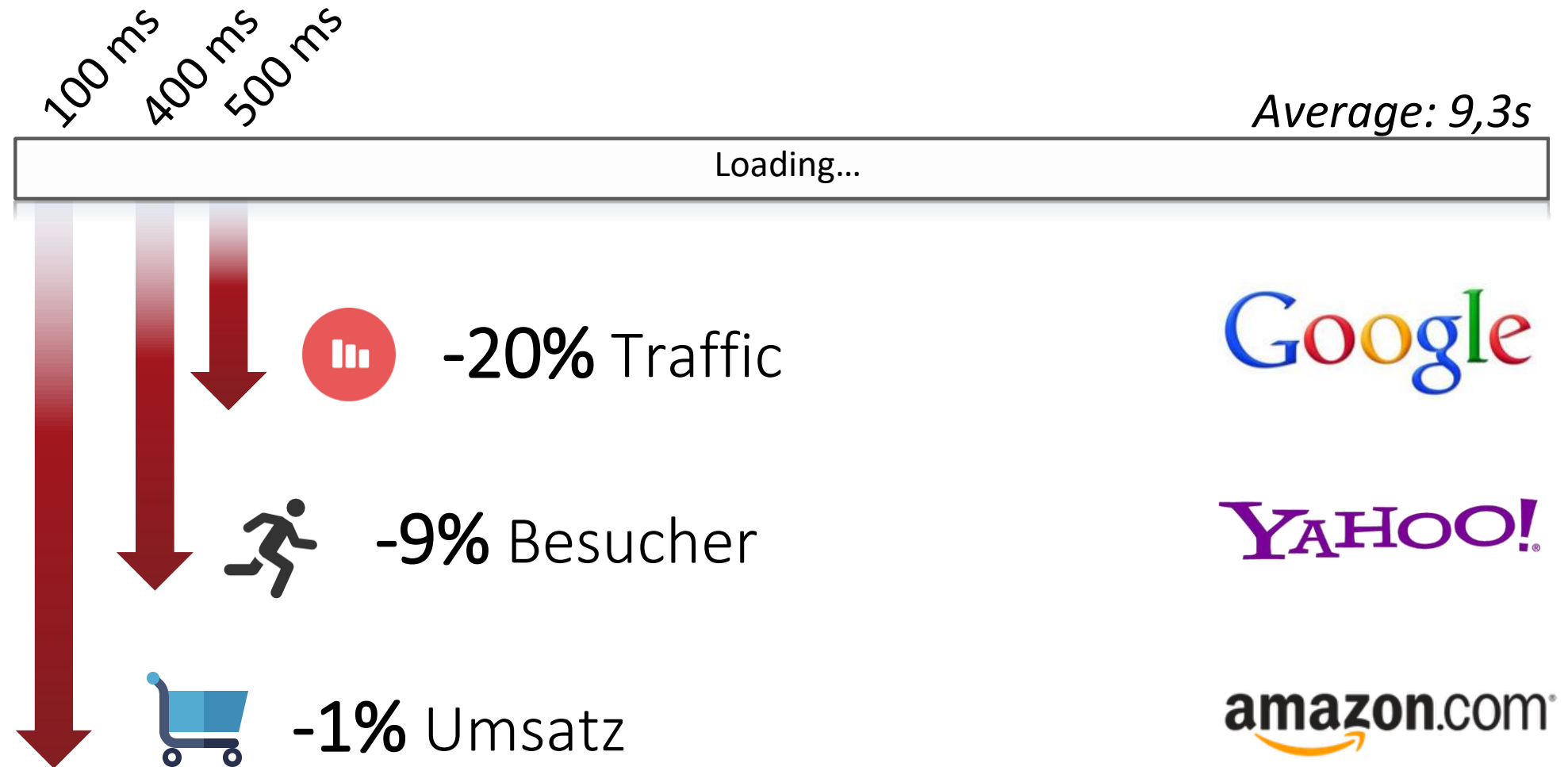


- + High flexibility and full control
- Build software from scratch
- High maintenance overhead



Presentation
is loading

The Latency Problem



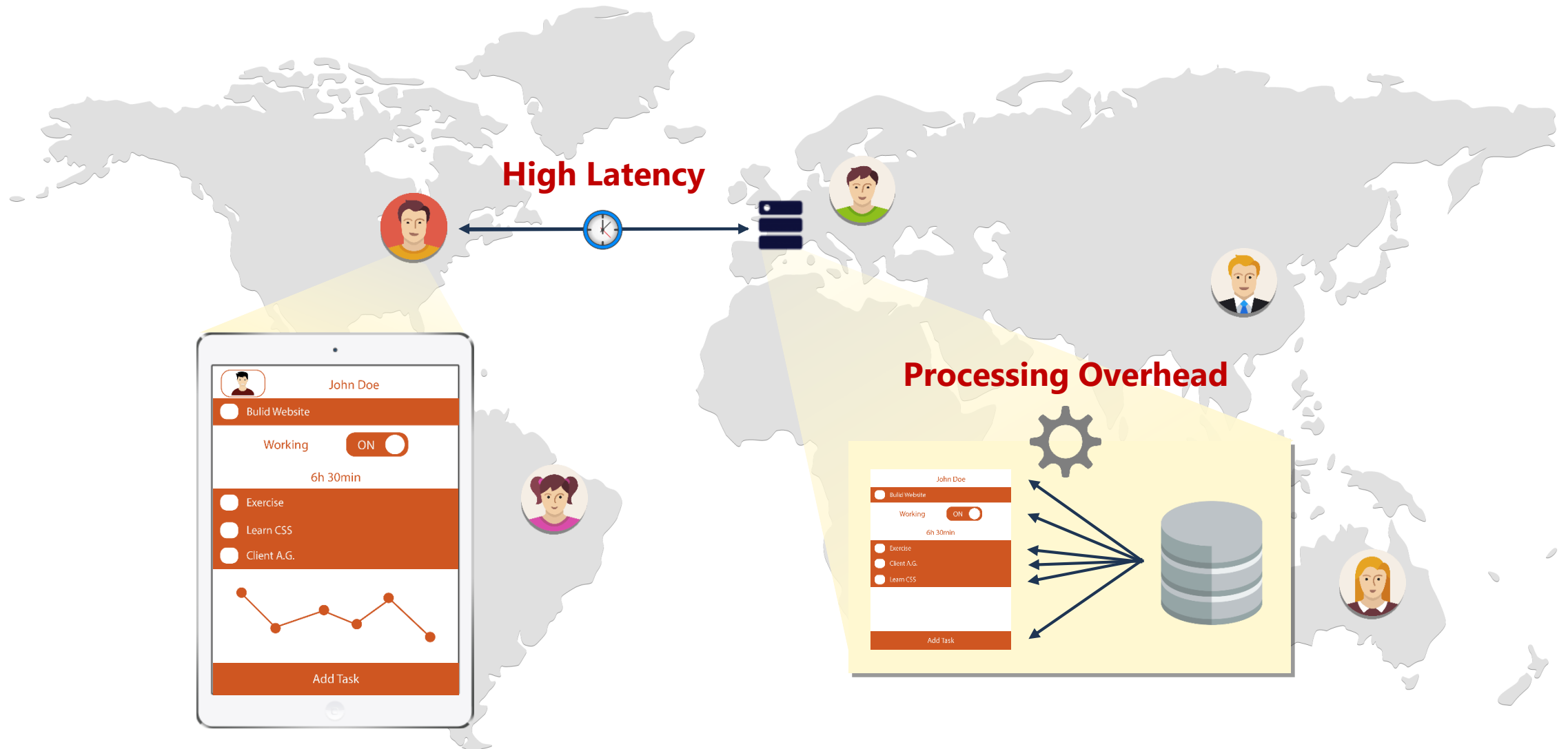


Performance affects many
Business KPIs

...what causes slow page loads?

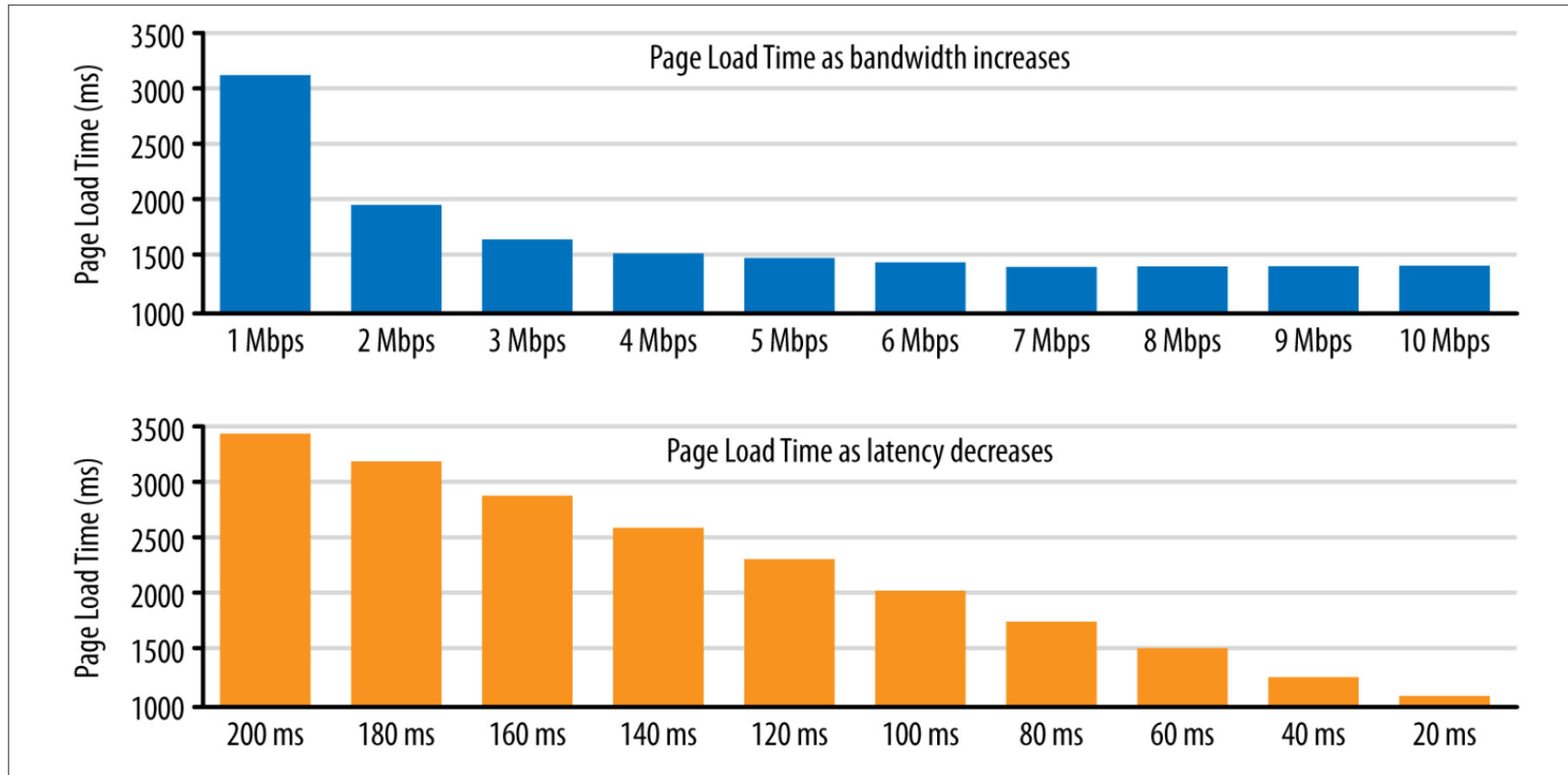
State of the Art

Two Bottlenecks: Latency and Processing



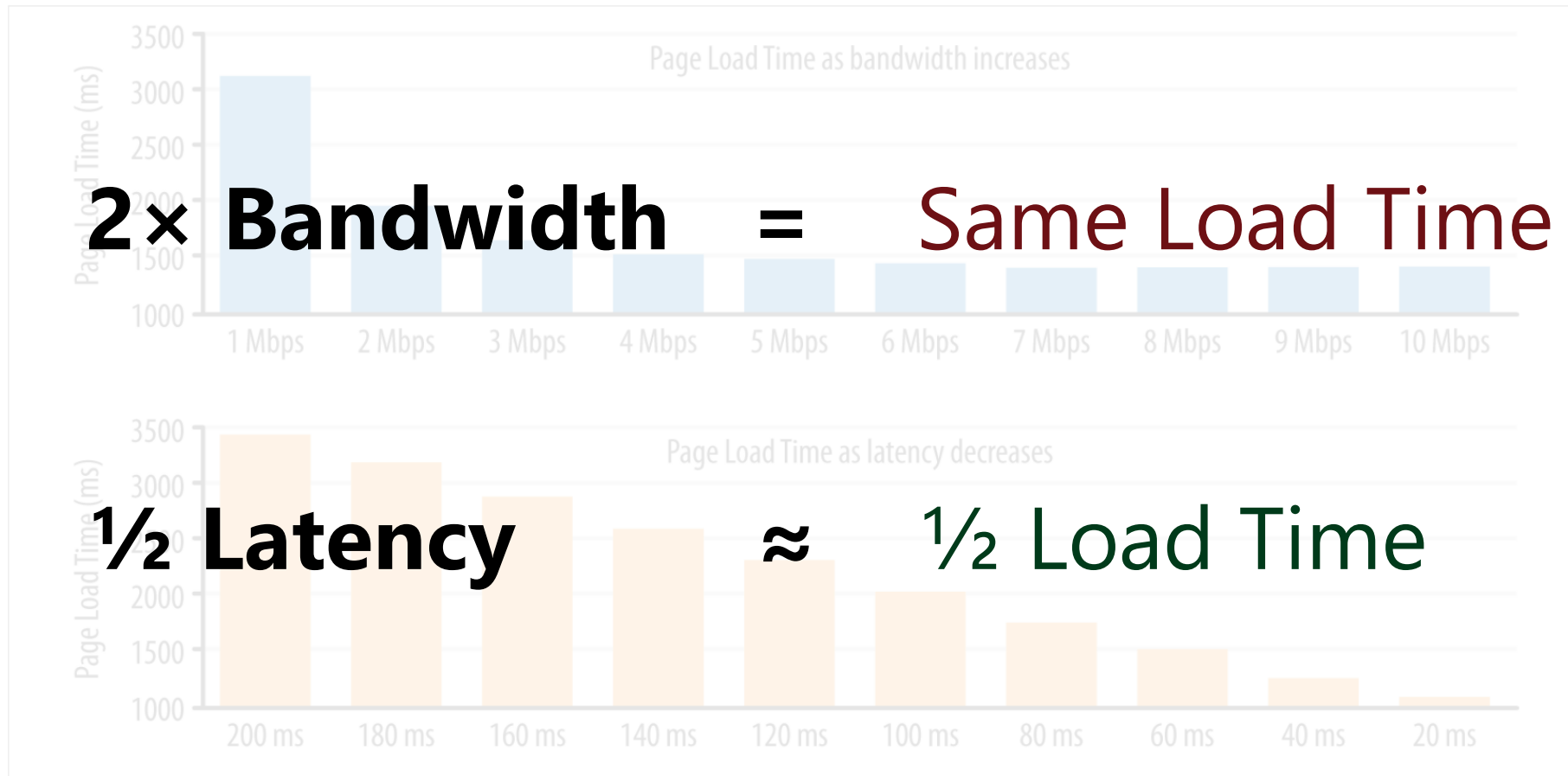
Network

Bandwidth vs. Latency



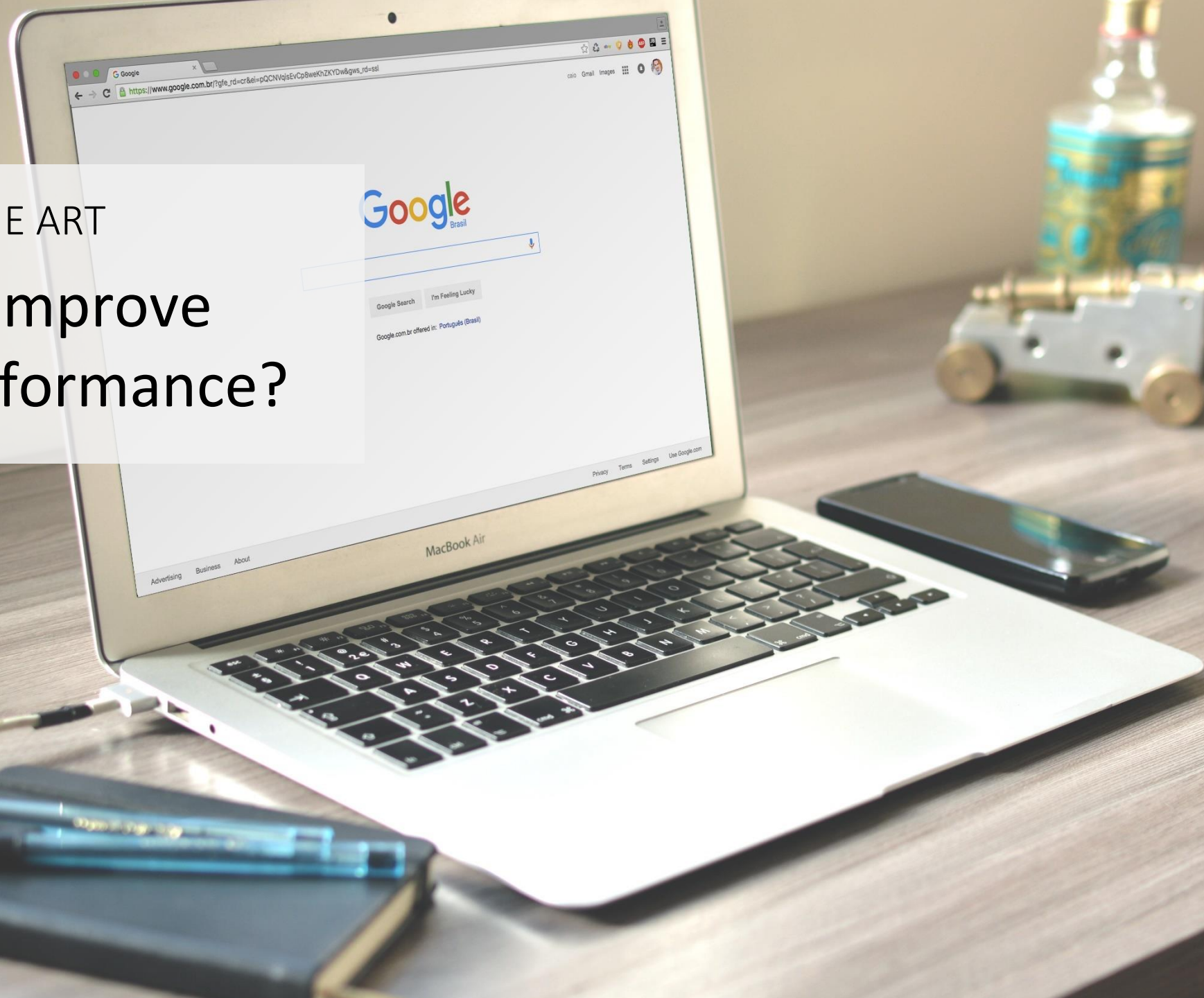
Netzwerk

Bandbreite vs. Latenz



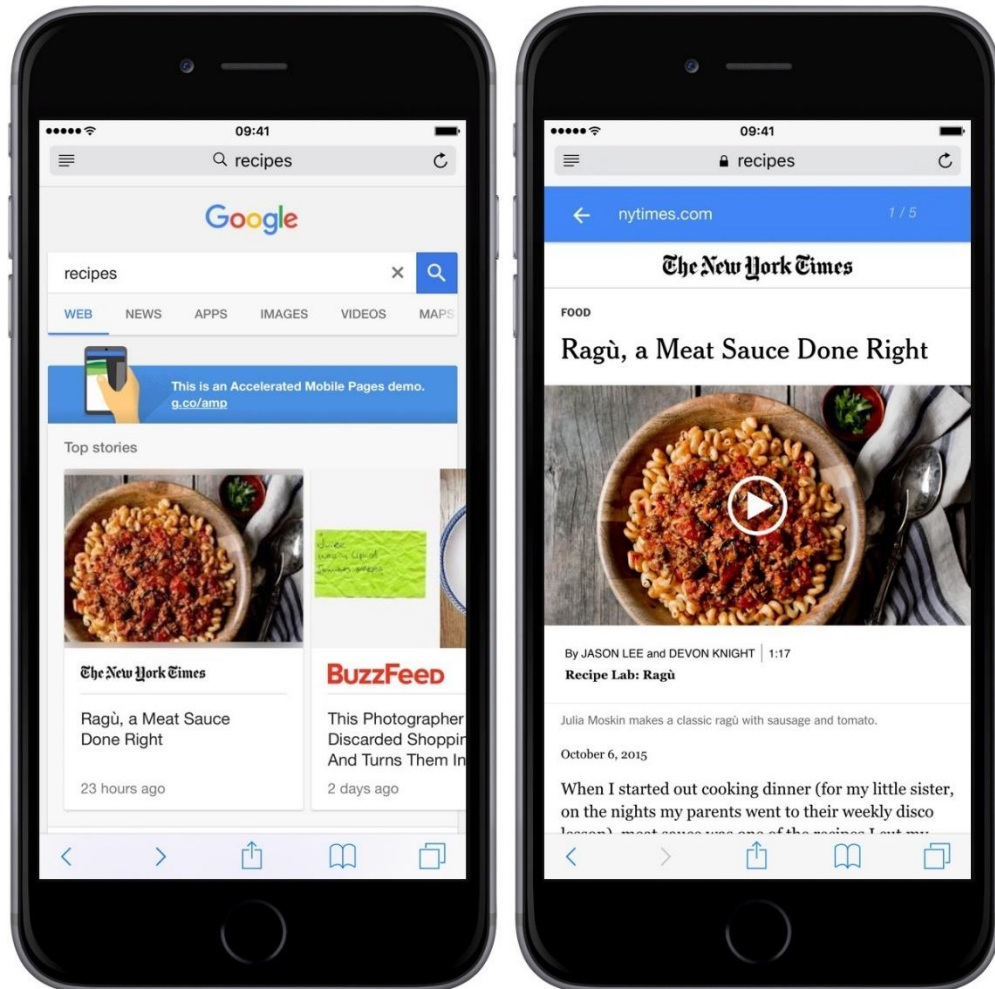
STATE OF THE ART

How to improve web performance?



What Others Do: AMP

Google's Accelerated Mobile Pages



How AMP works:

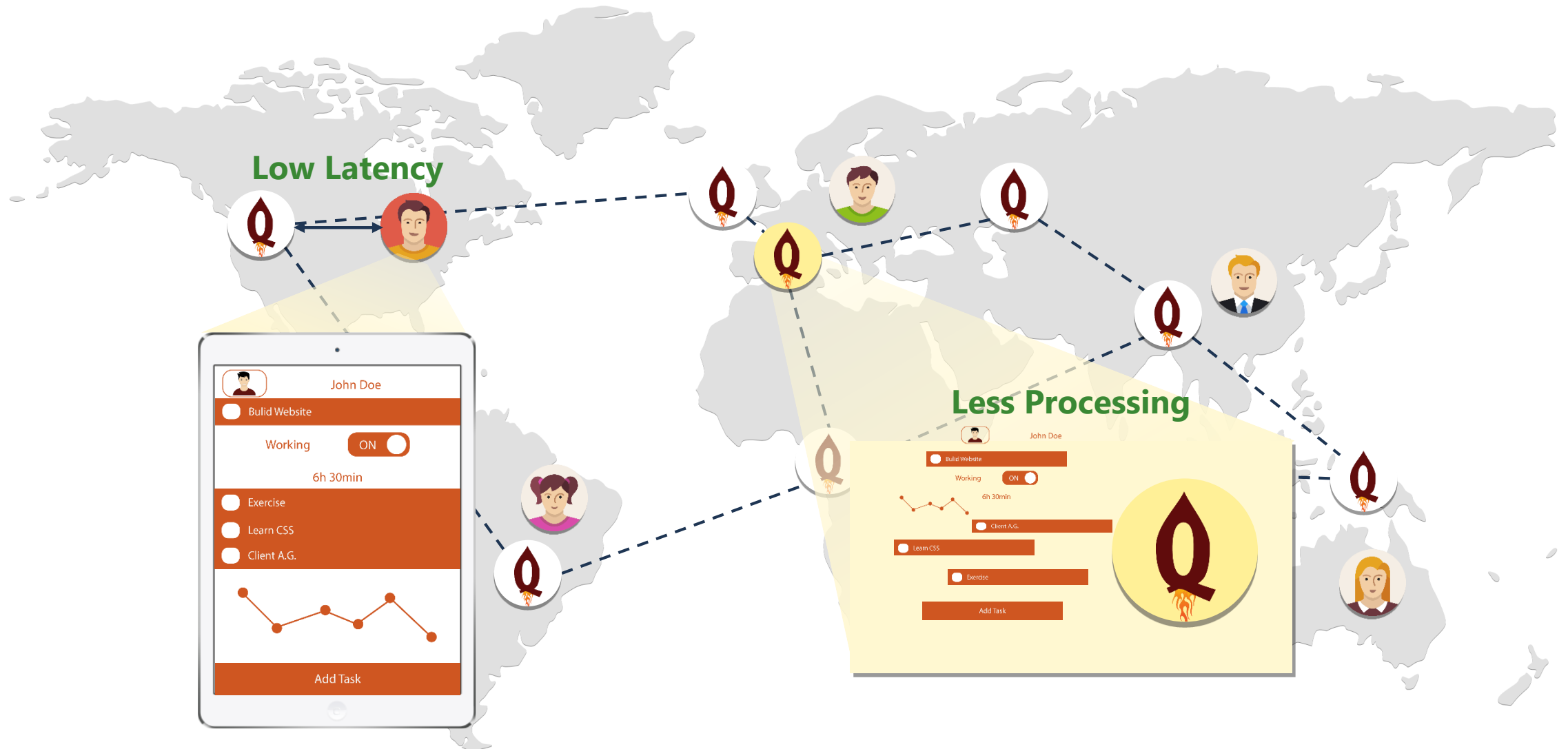
- **Stripped down** HTML + AMP tags (e.g. img)
→ rendered asynchronously by AMP runtime
- CSS must be **inlined** + **<50 KB** +
- No **custom JS** (except in iframes)
- Only static sizes → **no repaints**
- **Cached** in Google CDN, as long as it is crawled the next time
→ only suited for **static** media, e.g. news



How to apply these techniques for ***any*** website?

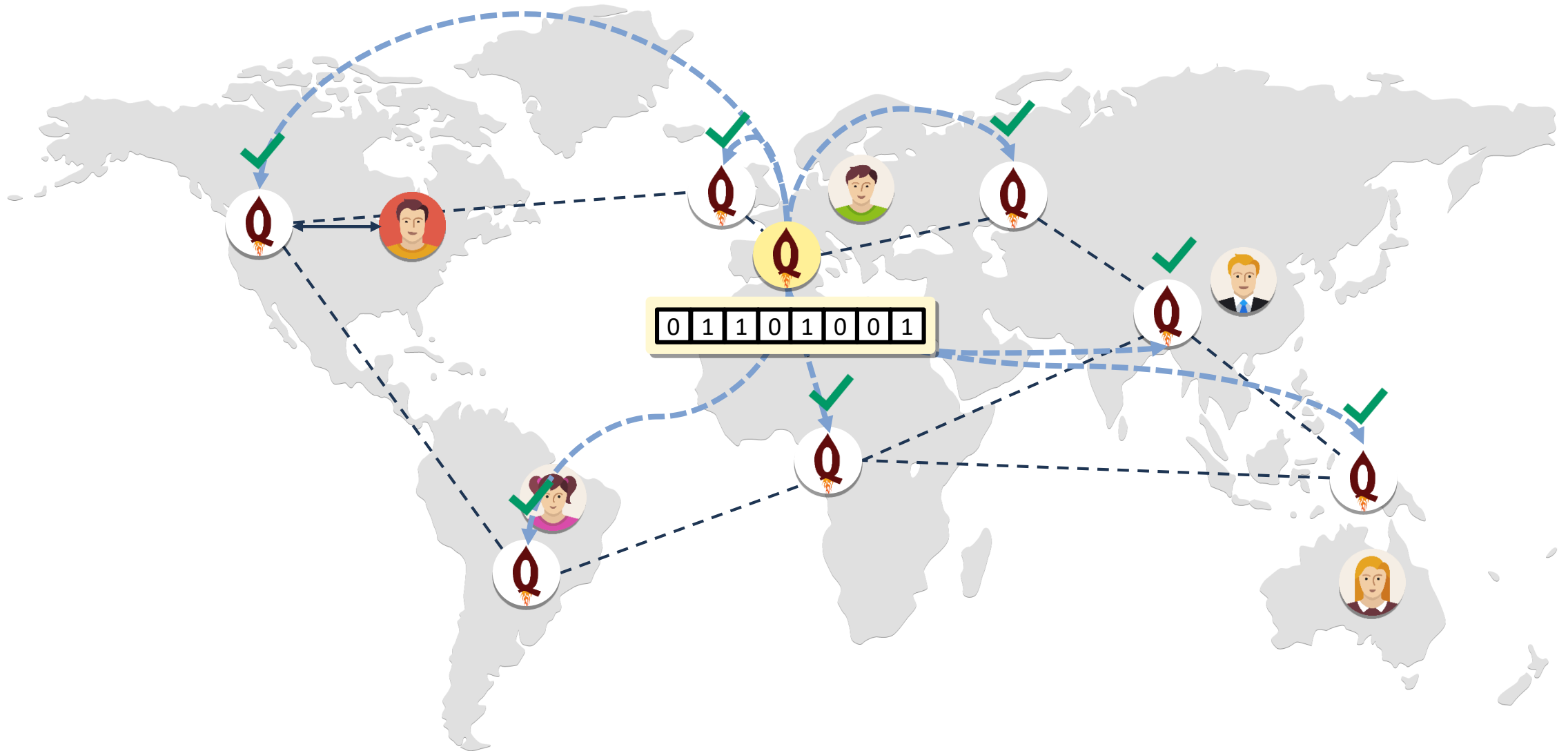
Solution: Global Caching

Fresh Data From Distributed Web Caches



New Caching Algorithms

Solve Consistency Problem



New Caching Algorithms

Solve Consistency Problem

F. Gessert, F. Bücklers, und N. Ritter, „ORESTES: a Scalable Database-as-a-Service Architecture for Low Latency“, in *CloudDB 2014*, 2014.

F. Gessert, S. Friedrich, W. Wingerath, M. Schaarschmidt, und N. Ritter, „Towards a Scalable and Unified REST API for Cloud Data Stores“, in 44. *Jahrestagung der GI*, Bd. 232, S. 723–734.

F. Gessert und F. Bücklers, „ORESTES: ein Cloud-Datenbank“, in Informatiktage

F. Gessert und F. Bücklers, *Performanzvermittlung der Web-Caching-Hierarchie*

M. Schaarschmidt, F. Gessert, und N. R. Persistence“, in BTW 2015.

S. Friedrich, W. Wingerath, F. Gessert, „Survey“, in *44. Jahrestagung der Gesell.* 704.

W. Wingerath, F. Gessert, S. Friedrich, N. Ritter „Real-time stream processing for Big Data“, *Big Data Analytics it - Information Technology*, 2016

F. Gessert, W. Wingerath, S. Friedrich, N. Ritter “NoSQL Database Systems: A Survey and Decision Guidance”, *Computer Science - Research and Development*, 2016

F. Gessert, N. Ritter „Scalable Data Management: NoSQL Data Stores in Research and Practice“, *32nd IEEE International Conference on Data Engineering, ICDE*, 2016

F. Gessert, N. Ritter „Polyglot Persistence“, *Datenbank Spektrum*, 2016.

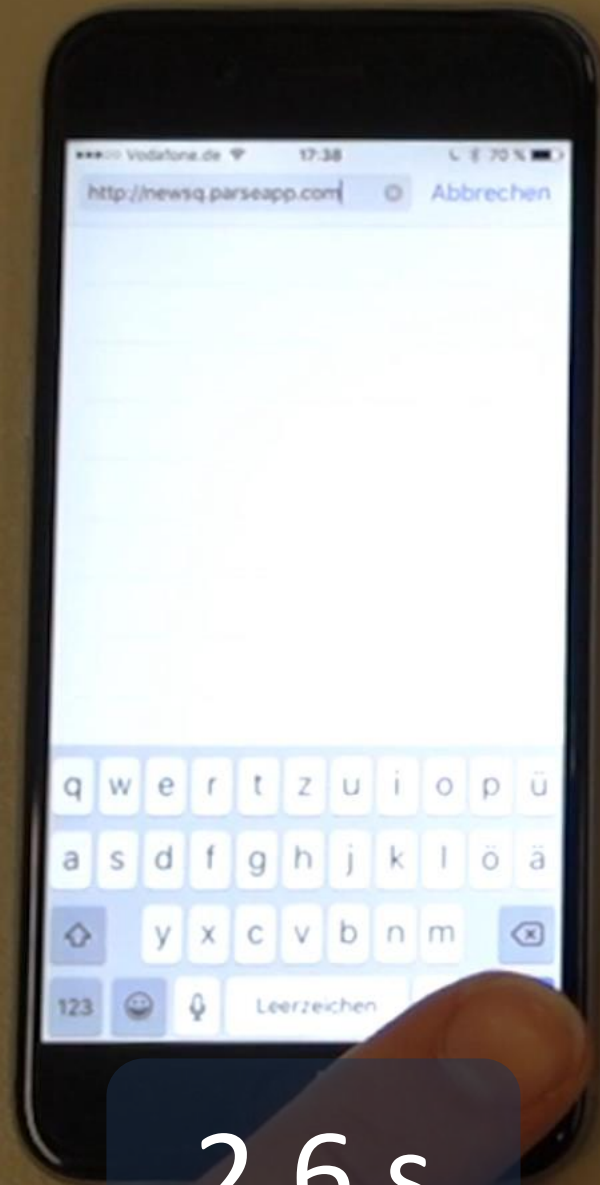


7 Year Research & Development



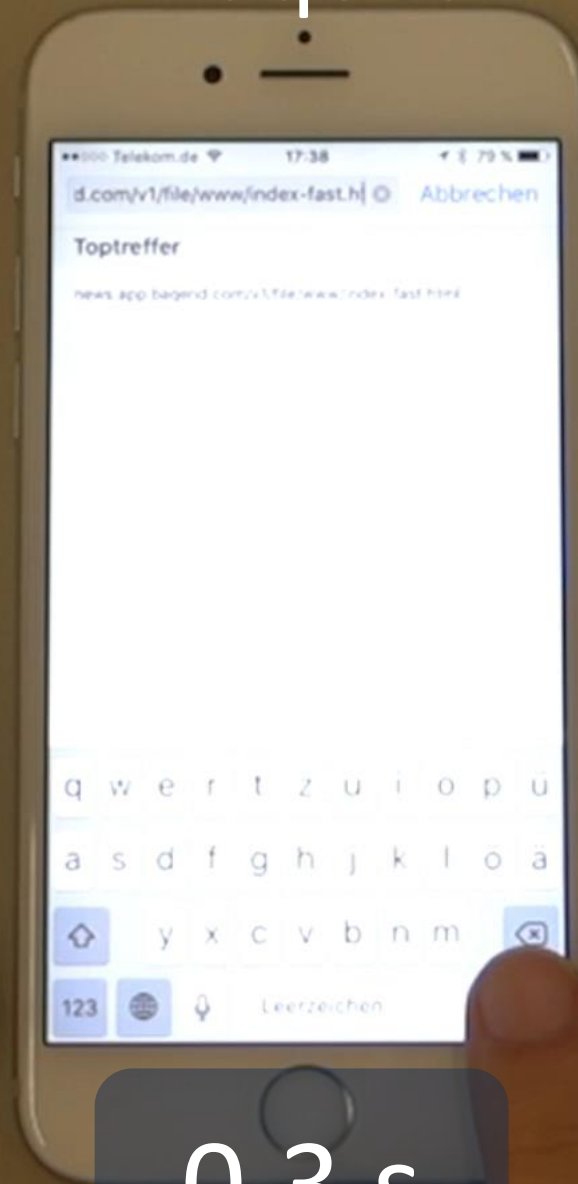
 Universität Hamburg

Parse



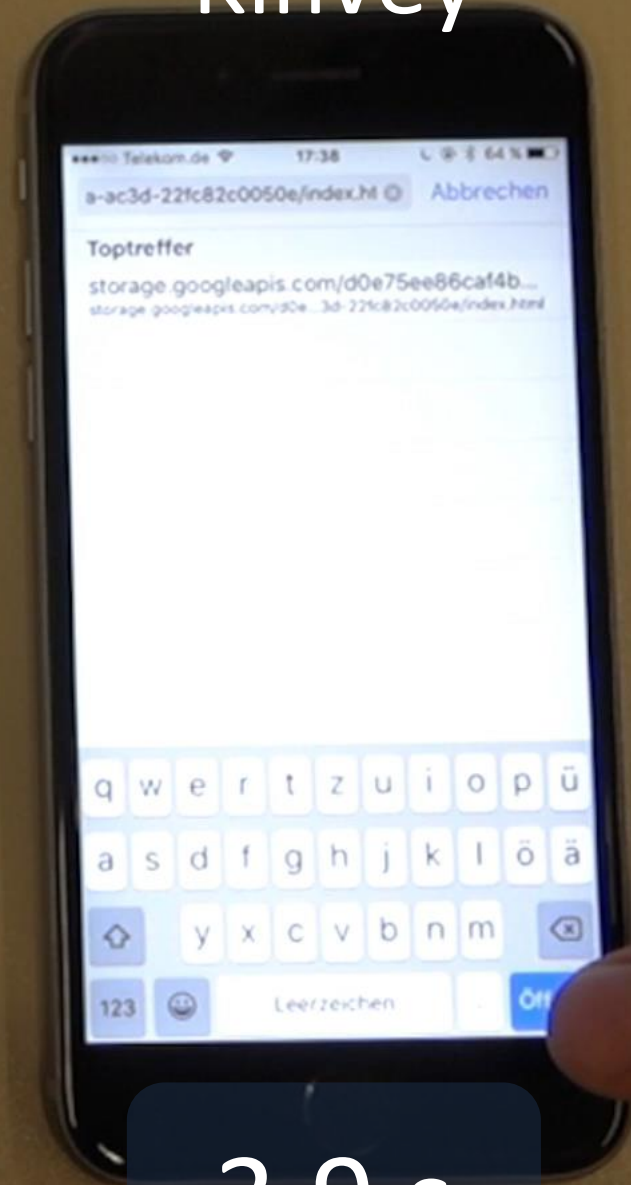
2.6 s

Baqend



0.3 s

Kinvey



3.9 s

Typical Speedup: 15x

Impact of Global Caching

TRY THIS

benchmark.baqend.com

Baqend
Azu
Par
Firebase
Kinvey
Apimio

Politik



11. November 2014 12:42 Uhr
Deutsche Rentenversicherung

Renten könnten 2015 um zwei Prozent steigen

Die Deutsche Rentenversicherung geht von einem Anstieg über der Inflationsrate aus. Abschlagsfreie Rente ab 63 Jahren stößt auf großes Interesse.



11. November 2014 10:05 Uhr
Europäischer Gerichtshof

Deutschland darf EU-Ausländern Hartz IV verweigern

Der Europäische Gerichtshof hat entschieden: Deutschland kann arbeitslose Zuwanderer aus der EU von Sozialleistungen ausschließen. Das Urteil könnte ein Signal sein.



11. November 2014 06:48 Uhr
APEC-GIPFELTREFFEN

Obama besänftigt China

Die USA wollen China nicht klein halten, sagt Präsident Obama vor dem Treffen mit Chinas Staatschef Xi. Der plädiert für mehr wirtschaftliche Verflechtung.



10. November 2014 19:17 Uhr
ISRAEL

Keiner will von Intifada sprechen

Messerattacken auf Israelis, Krawalle auf dem Tempelberg, Schirmmützen im Gassengewirr

Wirtschaft



11. November 2014 07:15 Uhr
HONORARBERATUNG

Guter Rat zur Geldanlage ist selten

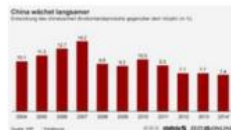
Honorarberatung ist in Deutschland endlich gesetzlich geregelt. Doch gibt es kaum Honorarberater. Und gut qualifizierte noch viel weniger.



10. November 2014 21:32 Uhr
CHINA

Der berühmteste Wohltäter Chinas – nach eigenen Angaben

Der chinesische Unternehmer Chen Guangbiao wurde ausgerechnet mit Bauschutt sehr reich. Jetzt baut er Wände aus Geldbündeln und zertrümmert öffentlich Luxusautos.



10. November 2014 19:29 Uhr
KONJUNKTUR

China steckt in der Wachstumsfalle

Jahrelang hat China die Welt mit hohen, oft zweistelligen Wachstumsraten beeindruckt. Doch diese Zeiten sind vorbei, wie unsere Grafik des Tages zeigt.



10. November 2014 13:45 Uhr
WÄHRUNG

Russlands Zentralbank lässt Rubel frei handeln

Kultur



11. November 2014 10:14 Uhr
NICOLAUS HARNONCOURT

Mozarts Triptychon

Nikolaus Harnoncourt ist der Detektiv unter den Dirigenten. Jetzt legt er Indizien vor, wie drei von Mozarts Sinfonien zu einem nie gehörten Oratorium verschmelzen.



11. November 2014 06:39 Uhr
HANS MAGNUS ENZENSBERGER

Der Unerschütterliche

Hans Magnus Enzensberger wird 85. Ein Besuch bei dem herrlich eigenwilligen Intellektuellen. Mit Tumult hat er gerade ein erstaunlich persönliches Buch veröffentlicht.



10. November 2014 um 18:25 Uhr
DDR-DESIGN

Sandmännchen und Stasi-Mikrofone

Das größte Museum für DDR-Design steht ausgerechnet in Los Angeles. Ein Buch über das Wende Museum zeigt, welche Schätze und Abgründe es dort zu entdecken gibt.



10. November 2014 um 15:25 Uhr
AZEALIA BANKS

Klare Ansage aus Harlem

Erst galt Azealia Banks als großes Raptalent, dann als streitsüchtig und selbstverliebt. Ihr seit Jahren erwartetes Debüt zeigt jetzt, wie gut das eine zum anderen passt.

2s

5,3s

6,6s

6,3s

6,5s

9,3s



STRYVE

by **THINKS**

TOWELL⁺

Platinum Grau & Anthrazit



19,95 €*

HINZUFÜGEN

PLATINUM GRAU
BEIGE

MERINO WEISS
BEIGE

GRAPHITE SCHWARZ
SCHWARZ

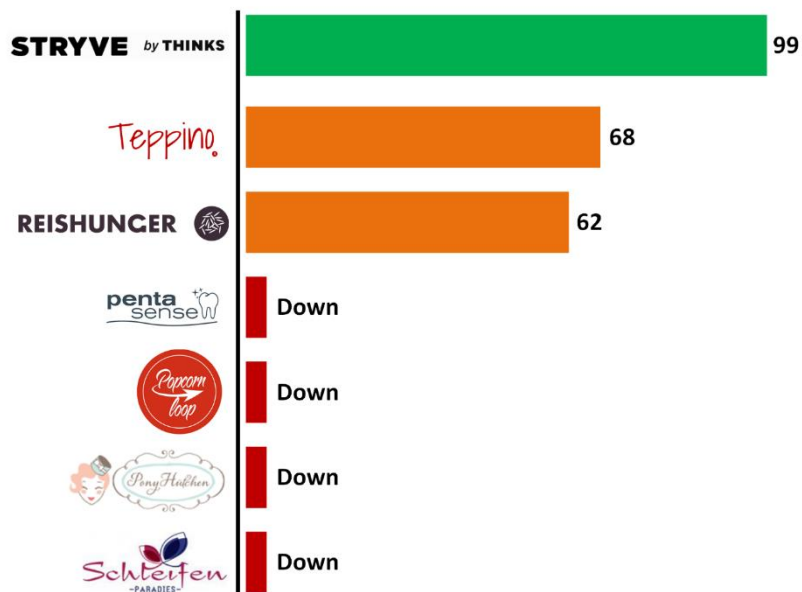
THINKS USE CASE

How do these techniques
work for a high-traffic shop?



Shops in "Die Höhle der Löwen"

The Google Page Speed Scores
for Season 3, 09/06/2016



BaQend

STRYVE by **THINKS** Shop



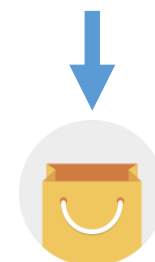
< 1 Second
Page Loads

Aktuell
46557

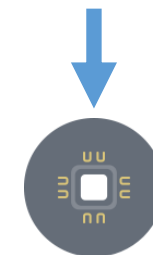
aktive Nutzer auf der Website



Concurrent
Users



7.8%
Conversion
Rate

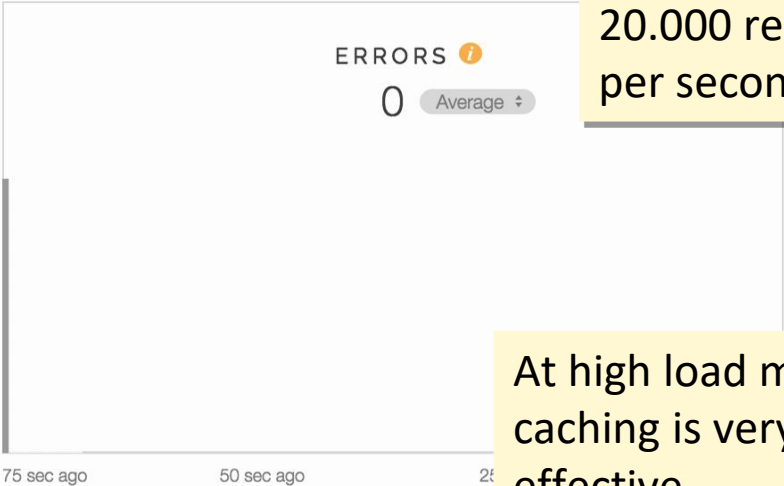
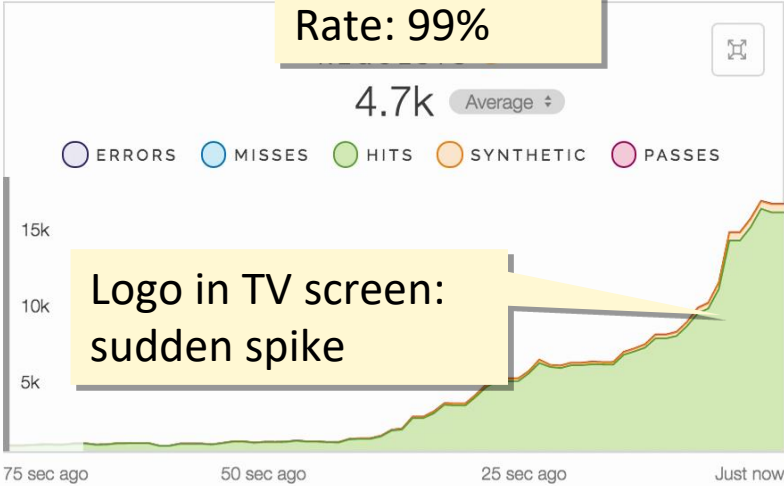


3%
Server
Usage

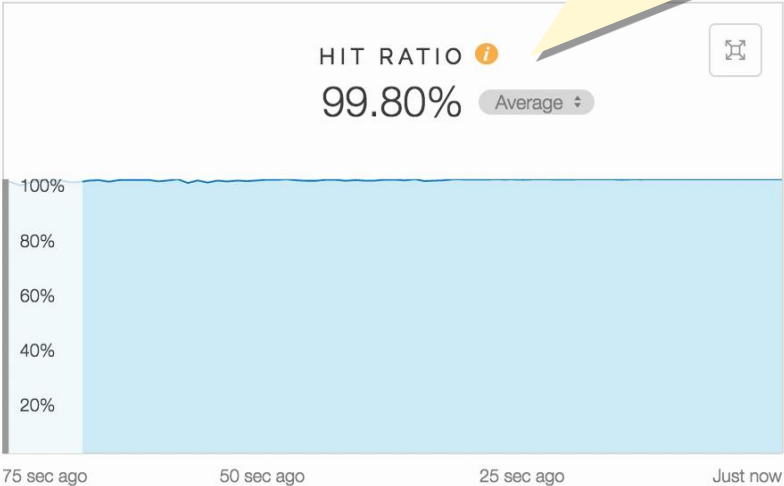
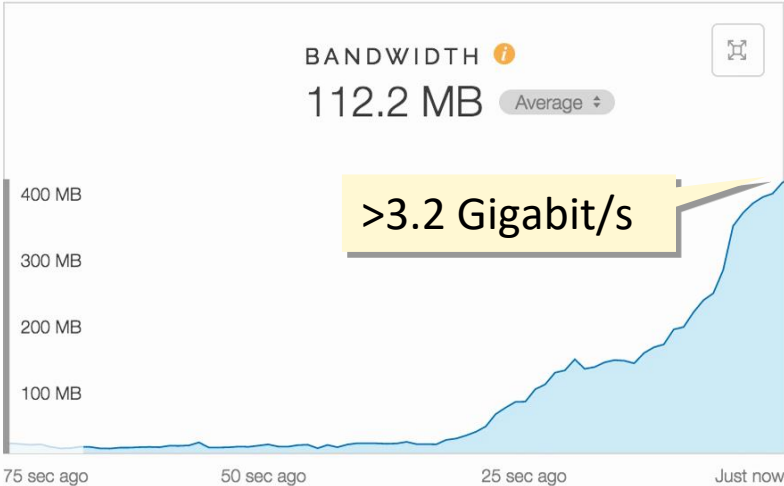
HITS	HIT TIME	MISSES	HIT RATIO	MISS TIME	REQUESTS	ERRORS
17,028	0.75ms	3	99.0%	89ms	17,475	0

High Cache Hit Rate: 99%

Peak Load of over 20.000 requests per second

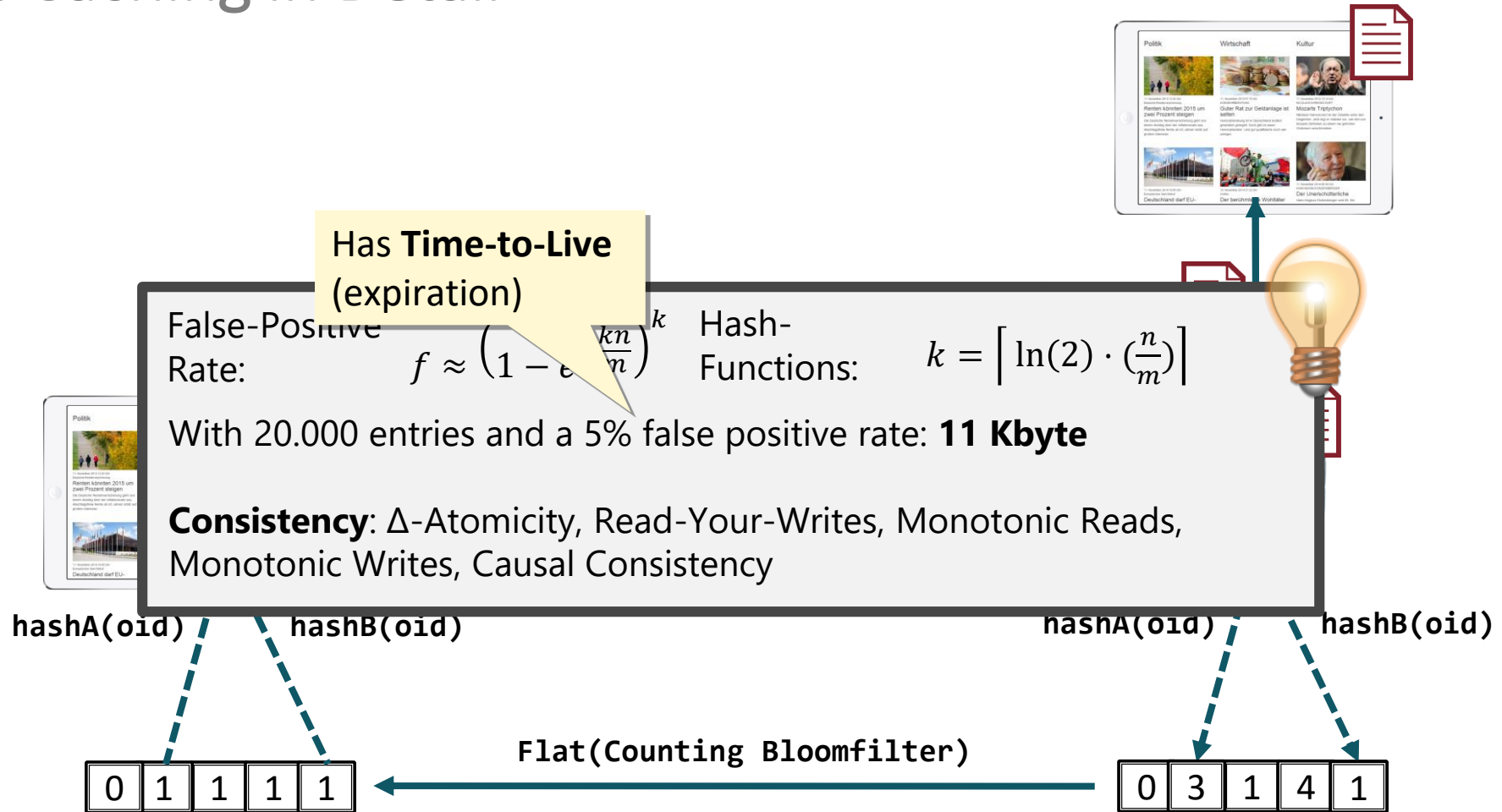


At high load micro-caching is very effective



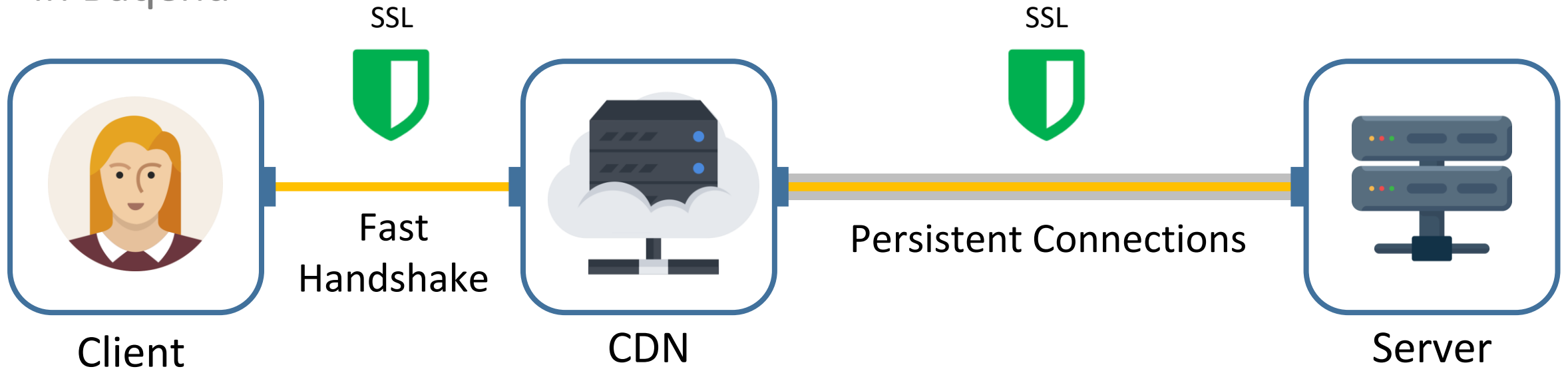
How does this work?

Dynamic Caching in Detail



Our Network Optimizations

In Baqend



- Early **SSL Termination**
- **OCSP Stapling**
- **Session Resumption**
- Warm **Backend-Connections**

- Caching at **Internet Exchange Points**
- Backend **Failover** & Stale-on-error
- Automatic **Skalierung**
- **DDOS** Protection

Why HTTP/2?

Typical Improvements

HTTP/2:

- **Multiplexing** over 1 TCP connection (no head-of-line blocking)
- Request **Pipelining**
- Server **Push**
- **Header Compression**

HTTP



3,22s

HTTPS



4,03s



A Medium Corporation [US] medium.com

HTTP



A Medium Corporation [US] https://medium.com

with CDN



A Medium Corporation (US) https://medium.com

0.44s



A Medium Corporation

HTTPS

with CDN and h2

0.35s

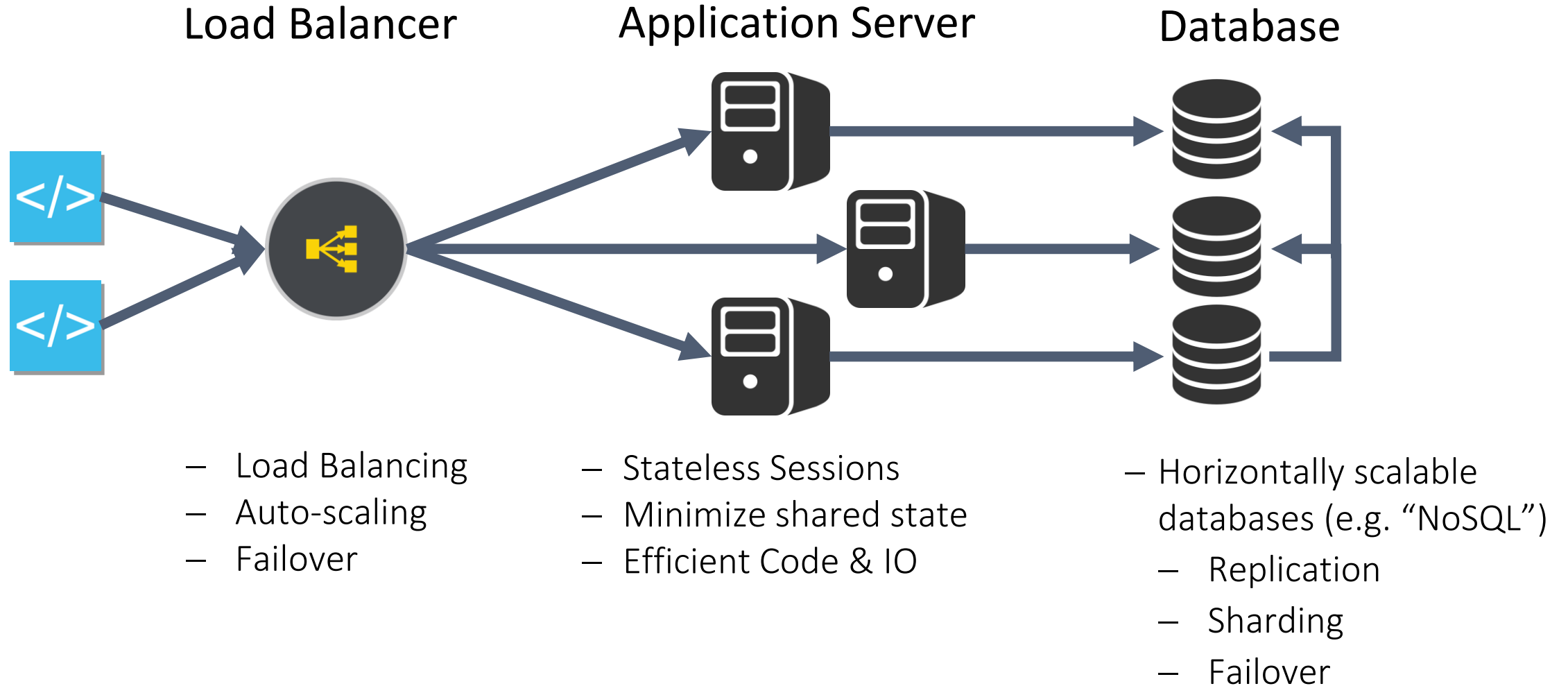


SCALING A BAAS

How to achieve
scalability?

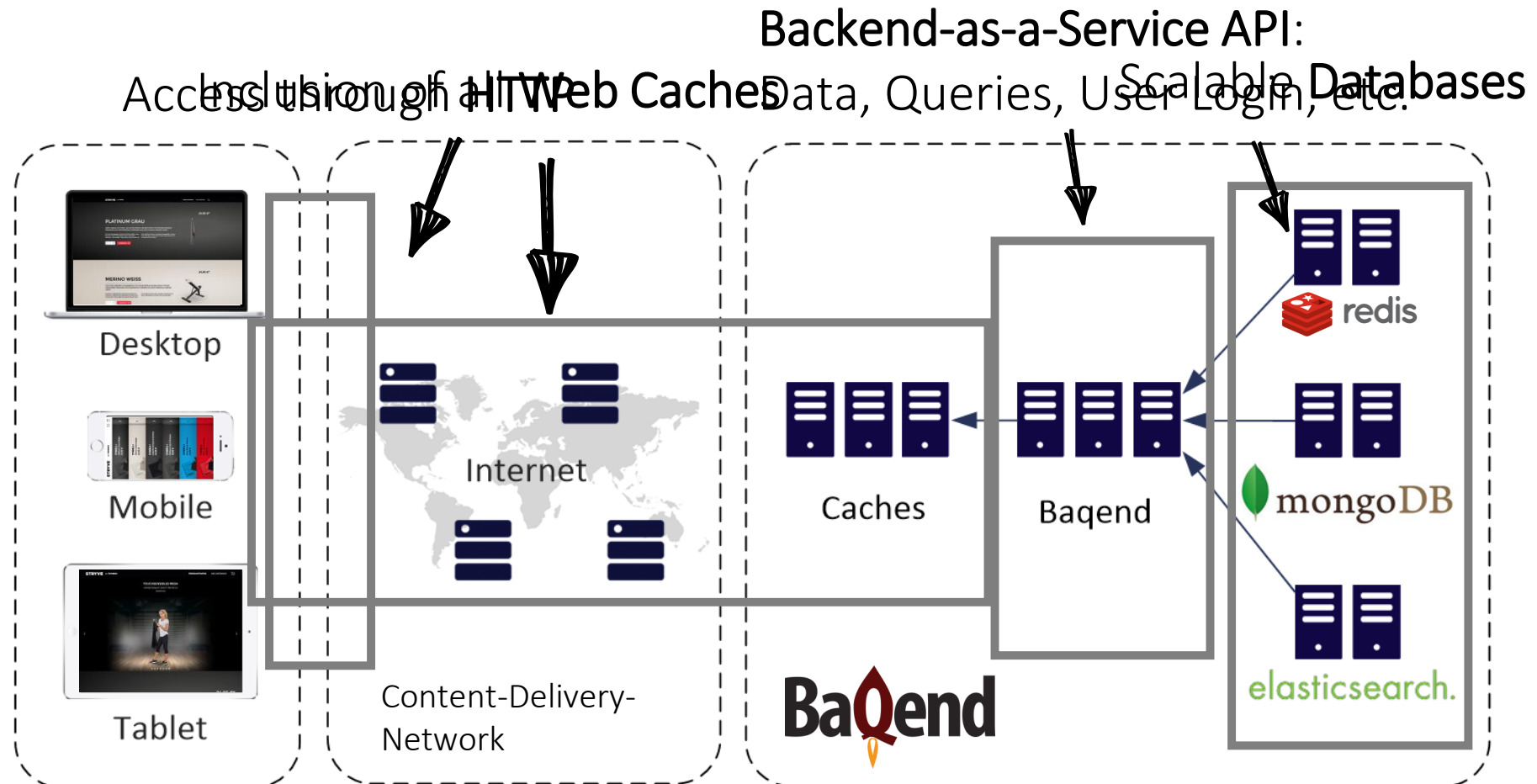
Backend Performance

Best Practices



Backend Architecture

Baqend Cloud

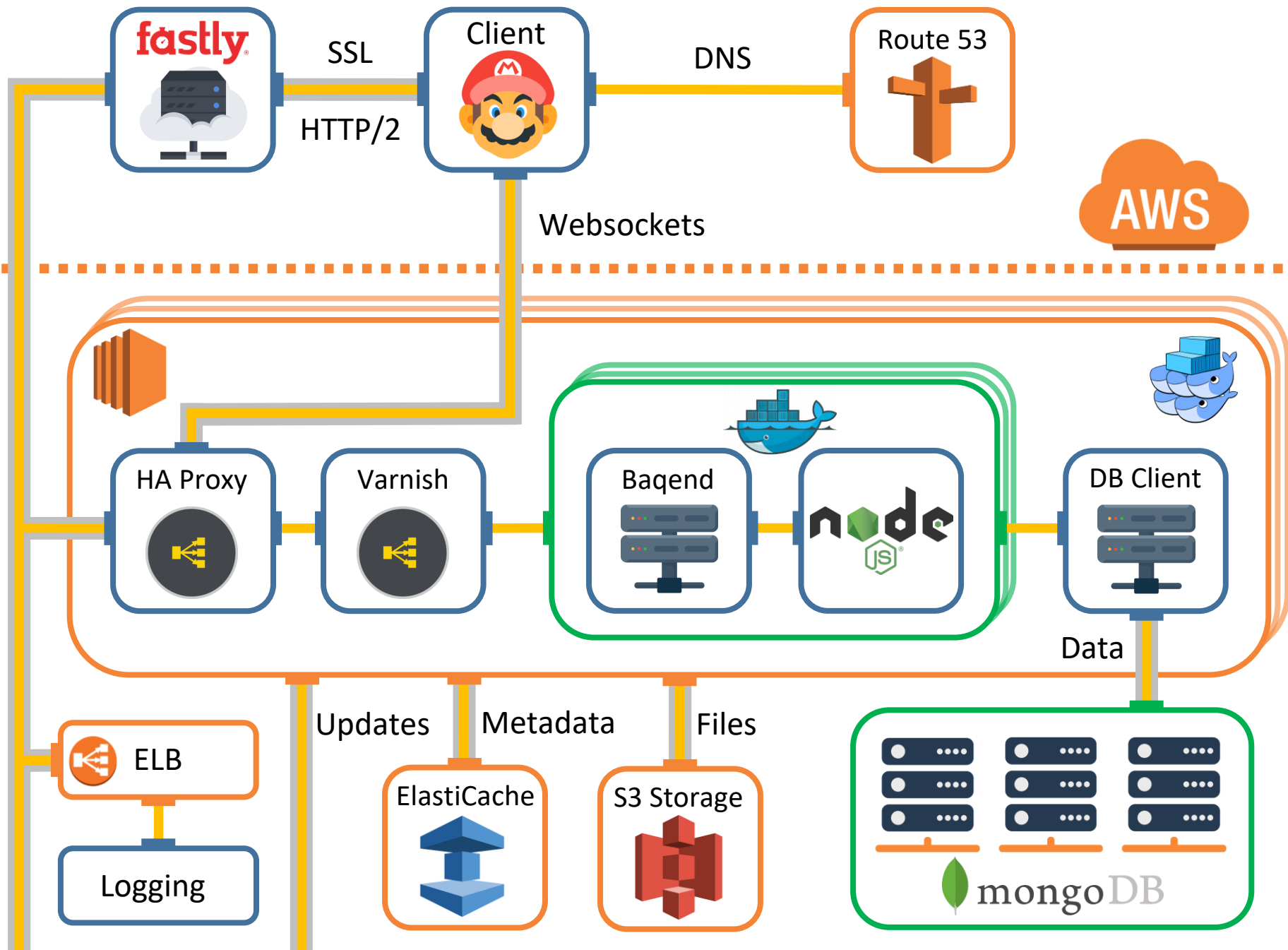


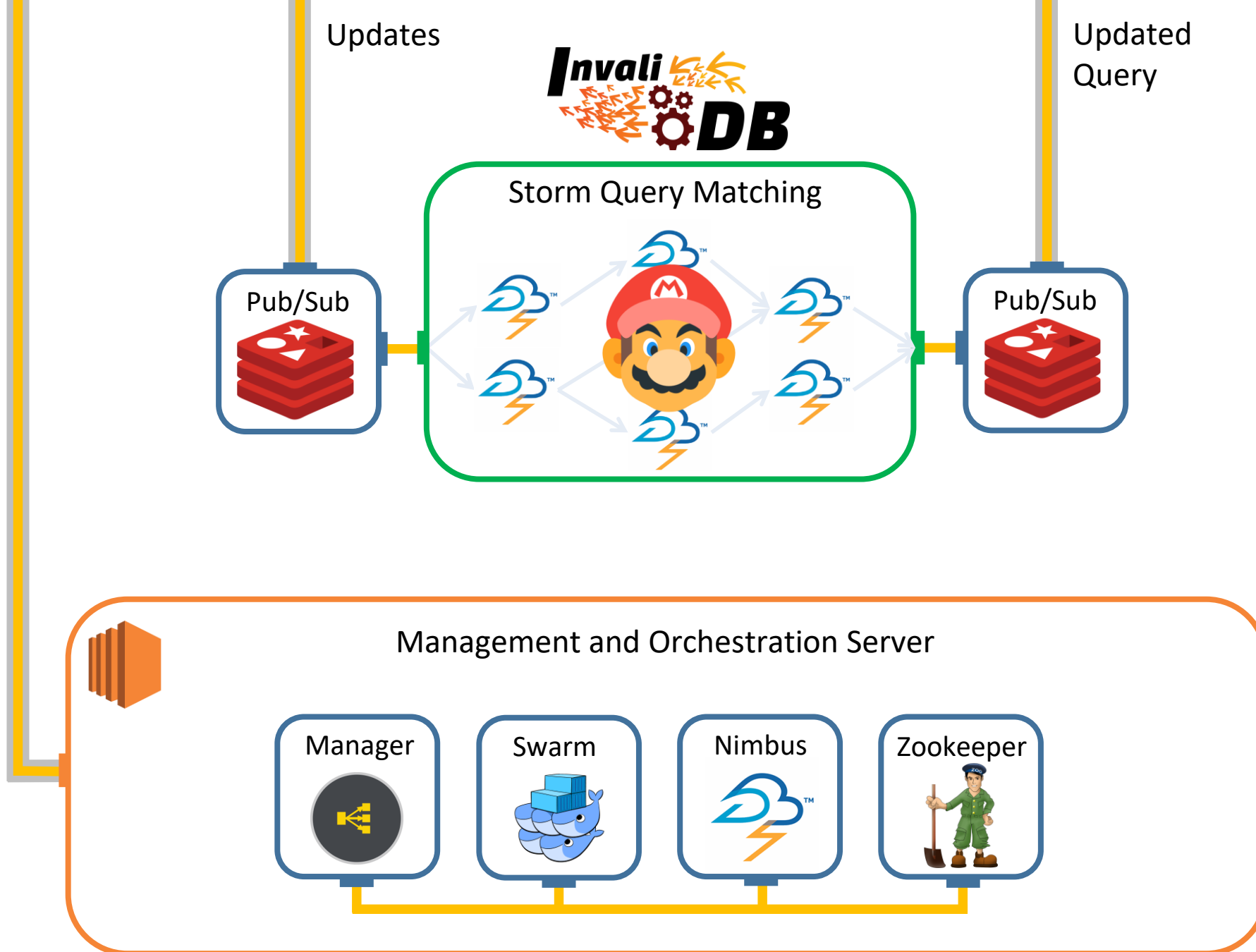
Backend Architecture

Baqend Cloud

Baqend







Overview

Optimizations



Consistent Browser Caching



Fast CDN-Invalidations



HTTP/2 & SSL



APIs for SPAs



High Availability



Automatic Scaling



DDoS Protection

Other Optimizations



Scale & Optimize Images



Load Javascript async.



Minify JS & CSS



Use Resource Hints

BAQEND OVERVIEW

How do I use Baqend?



Including Baqend

Getting Started



Product ▾ Developer ▾ About us Blog We're hiring

Log in or [Sign Up Free](#)

Querying Data

1 2 3 4 5 6 7 8 9

If we do not know the ids, we use queries to specify which objects we are looking for. The following query for instance gets us all the Todos in a specific list that are not marked as done:

```
DB.Todo.find()
  .equal("listId", "myList")
  .notEqual("done", true)
  .resultList().then(myCallback); //Returns a list of matching objects
```

`resultList` returns all matching objects, whereas `singleResult` only returns the first match.

Baqend supports many query operators, which allow very sophisticated lookups, including ordering, geospatial queries, text search and boolean combinations of query predicates.

Next >

HTMLCSSJSResult

EDIT ON CODEPEN

```
1 * class TodoService {
2   //load unfinished Todos
3   static unfinished(listId) {
4     return DB.Todo.find()
5       .equal("listId", listId)
6       .notEqual("done", true)
7       .resultList();
8   }
9 }
10
11 //Load data
12 * function onReady() {
13   insertData();
14   .then(() => TodoService.unfinished("mv-list"))
15     .then(todos => permissions.unfinished(todos), todos);
16 }
17
18 //Insert new data
19 function insertData() {
20   var todo1 = new DB.Todo();
21   todo1.id = "my-todo1";
22   todo1.listId = "my-list";
23   todo1.done = false;
24   todo1.save();
25
26   var todo2 = new DB.Todo();
27   todo2.id = "my-todo2";
28   todo2.listId = "my-list";
29   todo2.done = false;
30   todo2.save();
31
32   var todo3 = new DB.Todo();
33   todo3.id = "my-todo3";
34   todo3.listId = "my-list";
35   todo3.done = false;
36   todo3.save();
37 }
```

LIVE

Unfinished Todos:

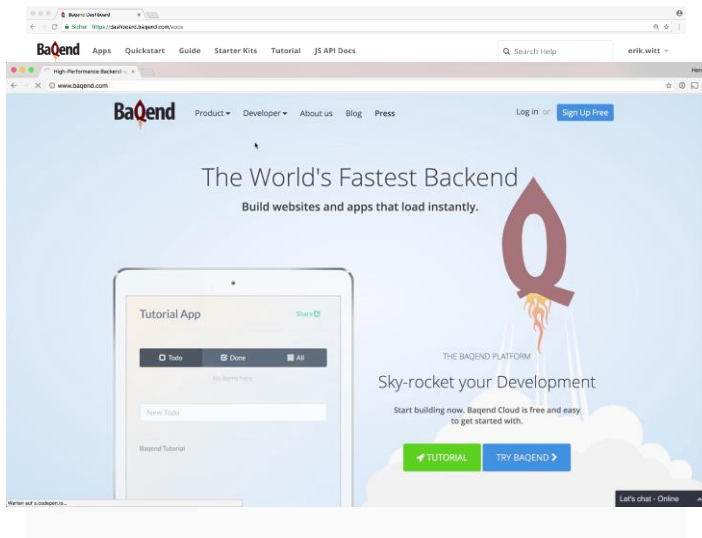
- My Todo [id:/db/Todo/my-todo] from list my-list
- My other Todo [id:/db/Todo/my-todo-2] from list my-list
- my third todo [id:/db/Todo/my-todo-3] from list my-list

Try this: www.baqend.com/tutorial.html

Development

On Baqend

Dashboard



Create Schema, Configurations, Browse Data, etc.

CLI

```
hannes@bq1:~$ baqend --help

Usage: baqend [command] [options] <args...>

Commands:

  login [options]           Logs you in and lo
  register                  Registers an accou
  open [app]                Opens the url to y
  dashboard                 Opens the url to t
  deploy [options] [app]    Deploys your baqen
  logout [options]          Removes your store
  typings [options] <app>  Generates addition
  start [name] [dir]        clones the starter

Type in one of the above commands followed by

Options:
```

Develop, deploy and test frontend und backend Code

REST & SDK

```
script.js
1 function leaveMessage(name, message) {
2   //Create new message object
3   var msg = new DB.Message();
4   //Set the properties
5   msg.name = name;
6   msg.message = message;
7   msg.date = new Date();
8   //Insert it to the database
9   msg.insert().then(showMessages);
10 }
11
12 function showMessages() {
13   DB.Message.find()
14     .descending("date")
15     .limit(30)
16     .resultList()
```

Website logic: load site, get data, login, etc.

IN DEPTH

How are the Baaend BaaS APIs used?

Baqend JS SDK

For Web & Hybrid



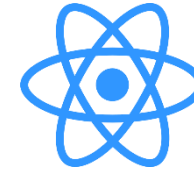
Browser



Node.js



Ionic & Cordova



React Native

- Completely **ES6-** and **TypeScript**-compatible
 - Support for Maps, Sets, Arrays
 - Models can be ES6-classes
- **Promise**-based
- Abstracts from **Caching** logic, intelligent **Object-Identity**
- Powerful **Query-Builder**
- Automatic **Change-** and **Dependency-Tracking**

Data Modelling

Combining Schemaful & Schemaless

ToDoList Object

```
{
  id : "mylist",
  name : "Todo Liste",
  owner : {
    name : "Johnny",
    age : 12
  }
}
```

Embedded Object

ToDoList Object

```
{
  id : "mylist",
  name : "Todo Liste",
  owner : "2333790852"
}
```

Reference

Owner Object

```
{
  id : "2333790852",
  name : "Johnny",
  age : 12
}
```

- **Types:** Boolean, Integer, String, DateTime, Time, Date, GeoPoint, List, Map, Set, JSON-Object, JSON-Array
- **References, embedded types & inheritance**

CRUD

Data APIs

Insert & Delete

```
var event = new DB.Event({id : 123, title : "Party"});  
event.insert();  
event.delete();
```

Updates

```
event.title = "Grillen";  
event.save();
```

Read

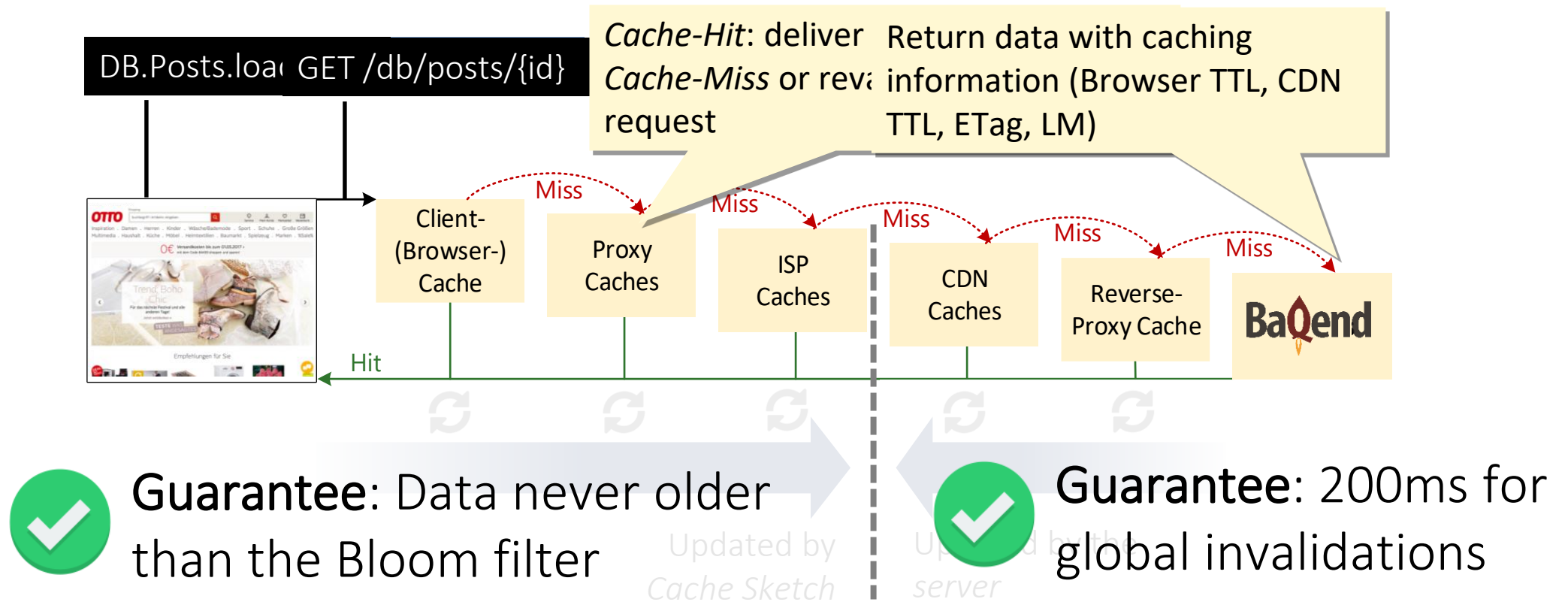
```
Db.TODO.load(123);
```

Query

```
DB.TODO.find()  
  .matches('name', /^My TODO/)   
  .equal('active', true)   
  .lessThanOrEqualTo('activities.start', new Date())  
  .resultList(...)
```

From CRUD to REST/HTTP

Cache Hierarchy



REST API

Access from all platforms

- Stateless and scalable
- Realized in Java/Jetty
 - ~15k RPS on Commodity Hardware
- Formalized in Swagger Specification
 - Language bindings can be generated
 - Browsing through REST API and its resources

orestes : Orestes Methods

[Show/Hide](#) | [List Operations](#) | [Expand Operations](#) | [Raw](#)

crud : Create Read Update Delete (CRUD) Object Methods

[Show/Hide](#) | [List Operations](#) | [Expand Operations](#) | [Raw](#)

POST	/db/{bucket}	Create object
GET	/db/{bucket}/{oid}	Get object
PUT	/db/{bucket}/{oid}	Replace object
DELETE	/db/{bucket}/{oid}	Deletes the object

schema : Object oriented methods

[Show/Hide](#) | [List Operations](#) | [Expand Operations](#) | [Raw](#)

GET	/db/all_schemas	Get all available class schemas
POST	/db/all_schemas	Create new class schemas and patch existing class schemas
PUT	/db/all_schemas	Replace all currently created schemas with the new ones
DELETE	/db/all_schemas	Remove all currently created schemas
GET	/db/{bucket}/schema	Get the class schema
POST	/db/{bucket}/schema	Update the class schema

Implementation Notes

Modify the schema definition of the class by adding all missing fields

Response Class

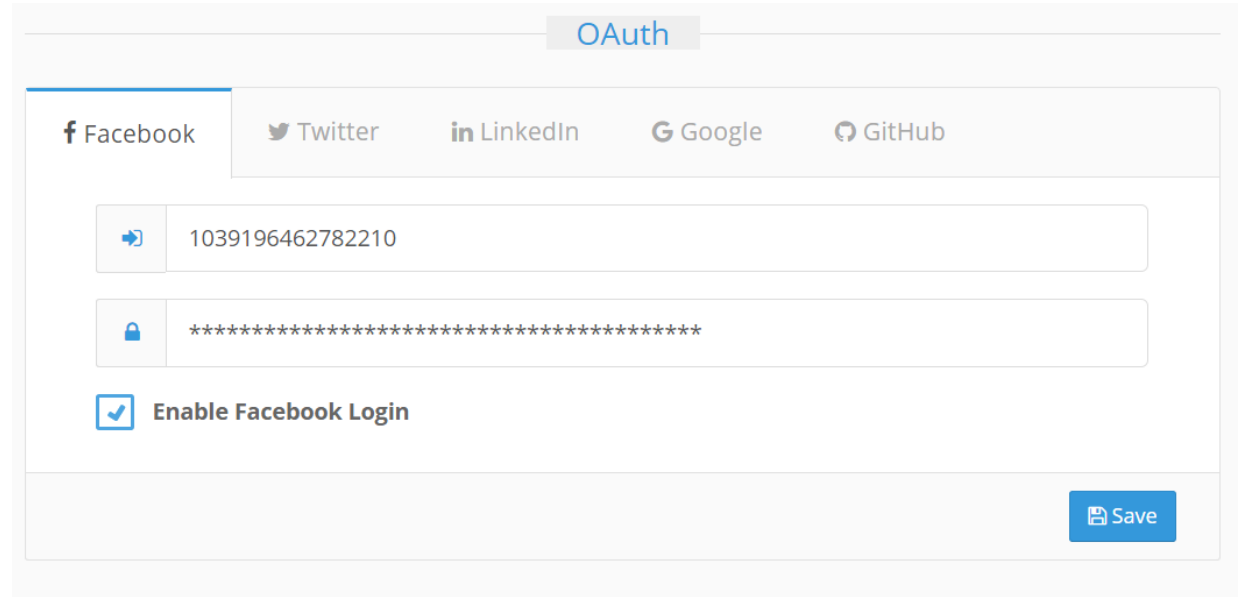
Model | Model Schema

object

User Management

Login, Registration, OAuth

- Simple **registration** and **login**
 - Via Email validation
 - Via OAuth-Providers
- User can have **roles**
- **Access rights** on user and role level
- **User-schema** is extensible
- **Sessions**, for returning users, too



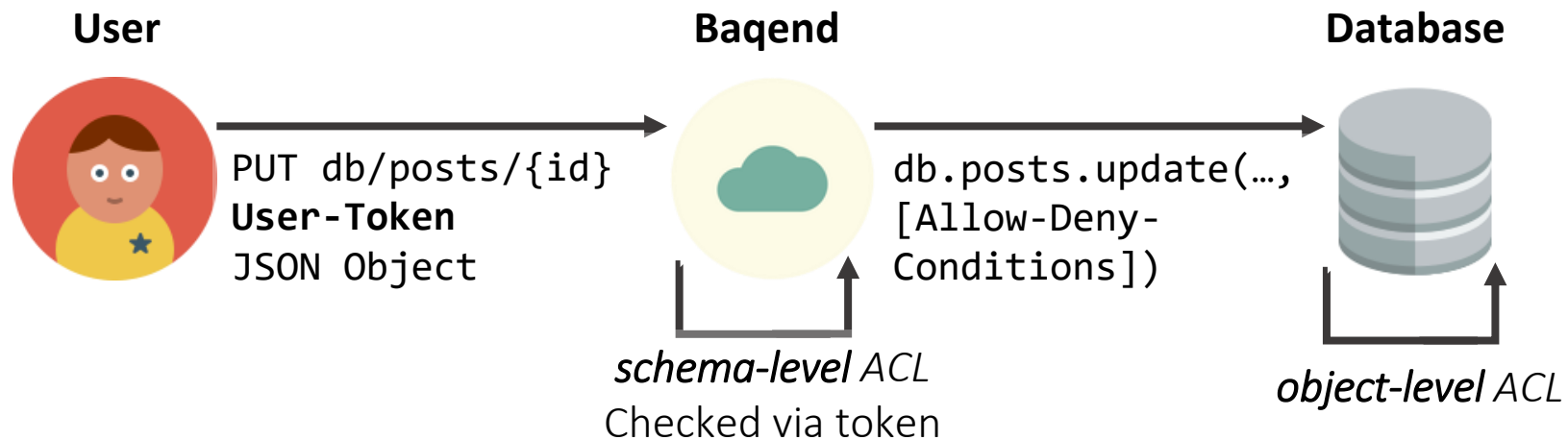
The screenshot shows a web interface for OAuth authentication. At the top, there's a tab labeled 'OAuth'. Below it, there are five tabs for different providers: Facebook, Twitter, LinkedIn, Google, and GitHub. The Facebook tab is currently selected. The form for Facebook login contains a text input field with the value '1039196462782210', a password input field with masked characters '*****', and a checkbox labeled 'Enable Facebook Login' which is checked. A blue 'Save' button is located at the bottom right of the form.

```
DB.User.login('john.doe@example.com', 'PW')
  .then(() => {
    //Hey we are logged in again
    console.log(DB.User.me.username);
    //'john.doe@example.com'
  });
```

Access Control

Authorization

	Load	Insert	Update	Delete	Query
Public	✓	✗	✗	✗	✓
 admin (/db/Role/1)	✓				✓
 node (/db/Role/2)	✓	✓	✓	✓	✓
+	felix-				
 felix-test					



Backend Code

Trusted Business Logic

- Callable Methods
- For **microservices** and custom **web APIs**

The screenshot displays the Baqend web interface. On the left is a sidebar with navigation links: Getting Started, Data, Backend Code (selected), Files, Logs, API Explorer, Settings, and Collaboration. The main area is divided into two sections. The top section, titled 'Baqend Module', contains a code editor with the following JavaScript code:

```
1 /* Require additional modules and handlers here */
2 //var module = require('module');
3 //var myCode = require('./otherModule');
4 //var updateHandler = require('./MyClass/update');
5
6 /**
7  * This method is invoked if an GET or POST request is send to this module resource
8  * Additional backend functionality can be implemented in such method
9  *
10  * @param {baqend.EntityManager} db The database instance which can be used to load and save additional objects
11  * @param {baqend.binding.User} db.User.me The actual unresolved user who requests the operation or null if the user
12  *   is not logged in
13  * @param {Object} data The request payload, the decoded query parameters of a GET request or the parsed body of a
14  *   POST request
15  * @param {express.Request} req The express request object {@link http://expressjs.com/api.html#req}
16  * @param {Object} this The module context
17  * @return {Promise<*>|Object|Array|String} A json value or string which is send back to the client
18  *
19  * @throws Abort(reason[, data]) to abort the request with the specified reason
20  */
21 exports.call = function(db, data, req) {
22   |
23   };
```

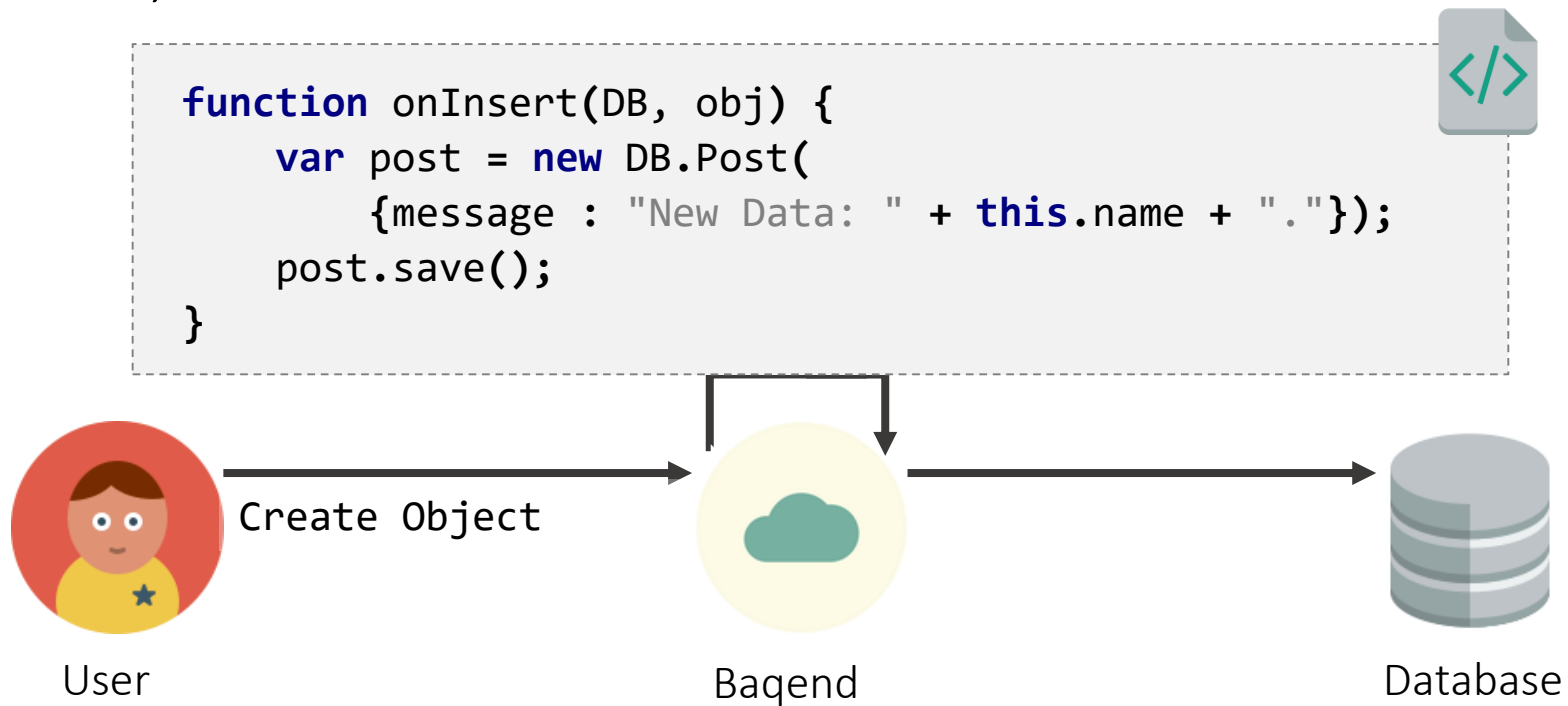
Below the code editor are two buttons: 'Discard Changes' and 'Save Module'. The bottom section, titled 'Test Module', contains configuration options:

- Type:** A dropdown menu with 'GET' selected.
- Get Params:** A text input field containing '?param=val...'.
- User:** A dropdown menu with 'Admin' selected.

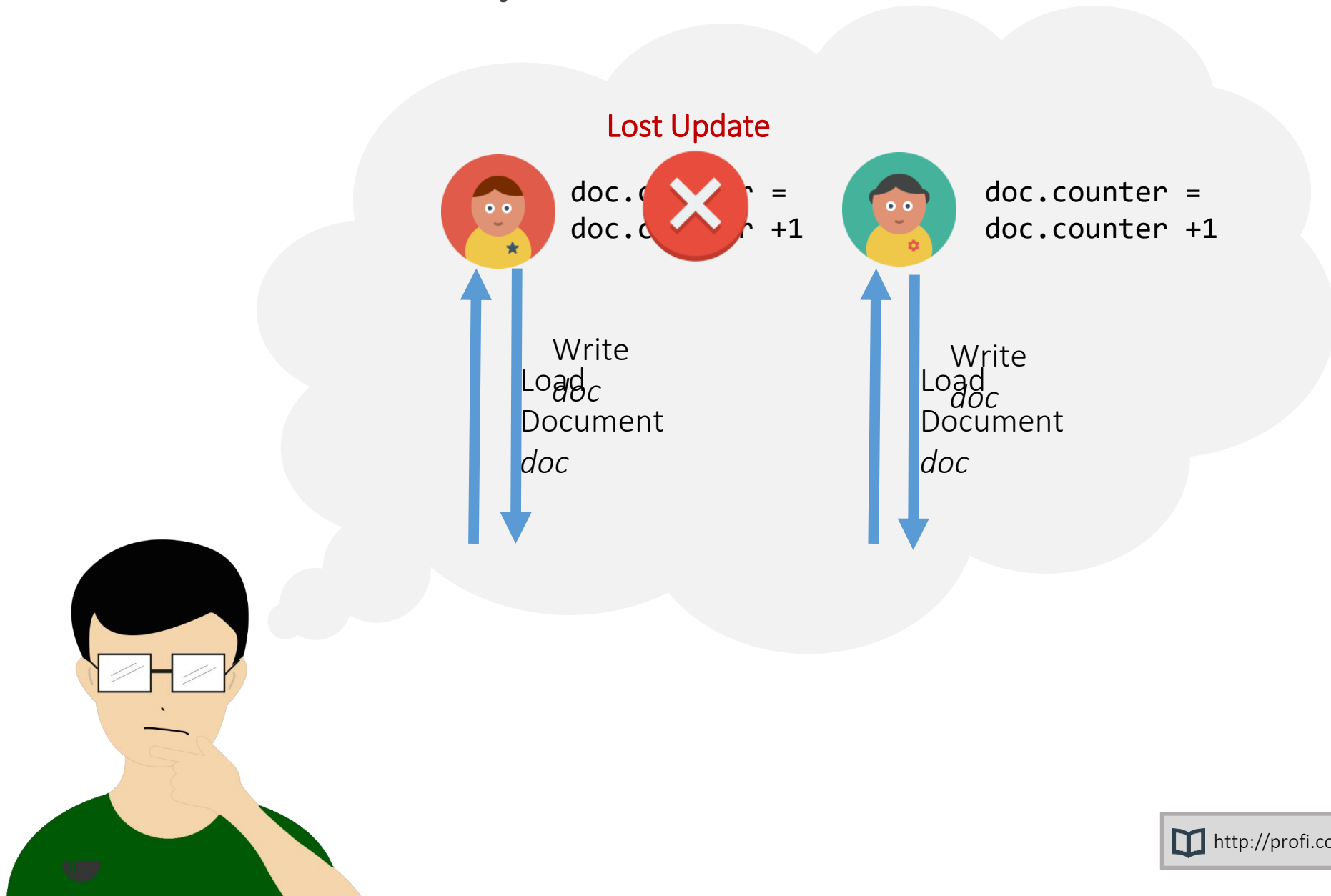
Backend Handler

Hooking Into Updates

- **Handler:** OnUpdate, OnDelete, OnInsert, OnValidate
- Node.js code, same SDK



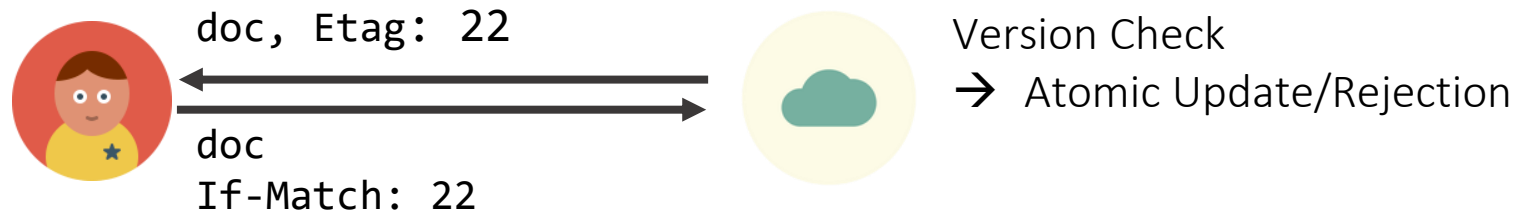
Problem: Concurrency



Concurrency in Baqend

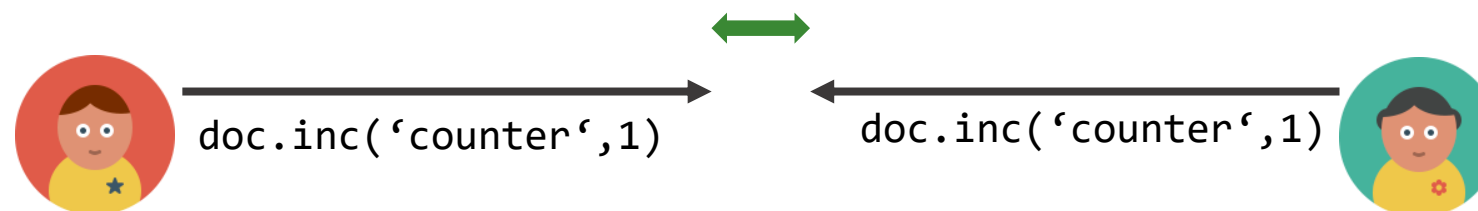
Safe Updates

Optimistic Concurrency Control:



Partial Updates:

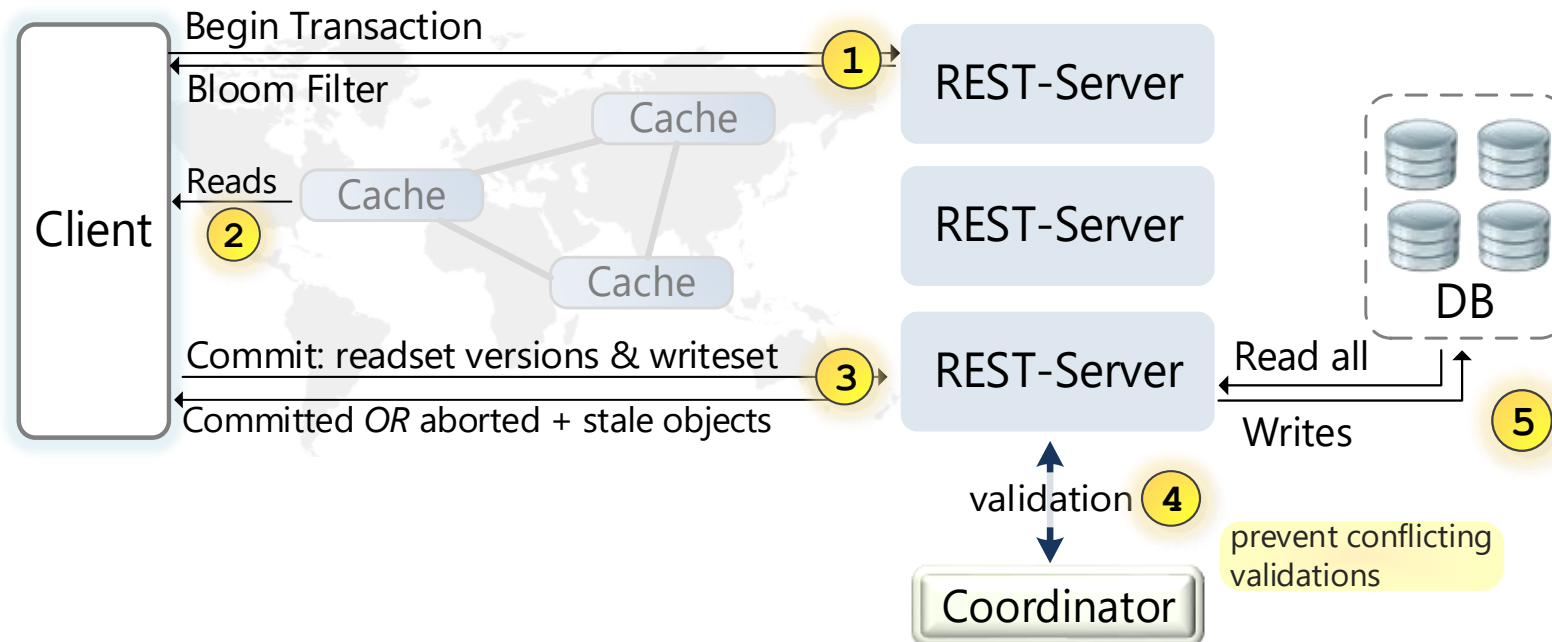
Others: push, pop, set, remove, etc. commutative



Concurrency in Baqend

Multi-Object Transactions

- Optimistic **ACID-Transactions** (*the* missing NoSQL-feature)
- Status: soon part of the SDK (currently in the REST-API)



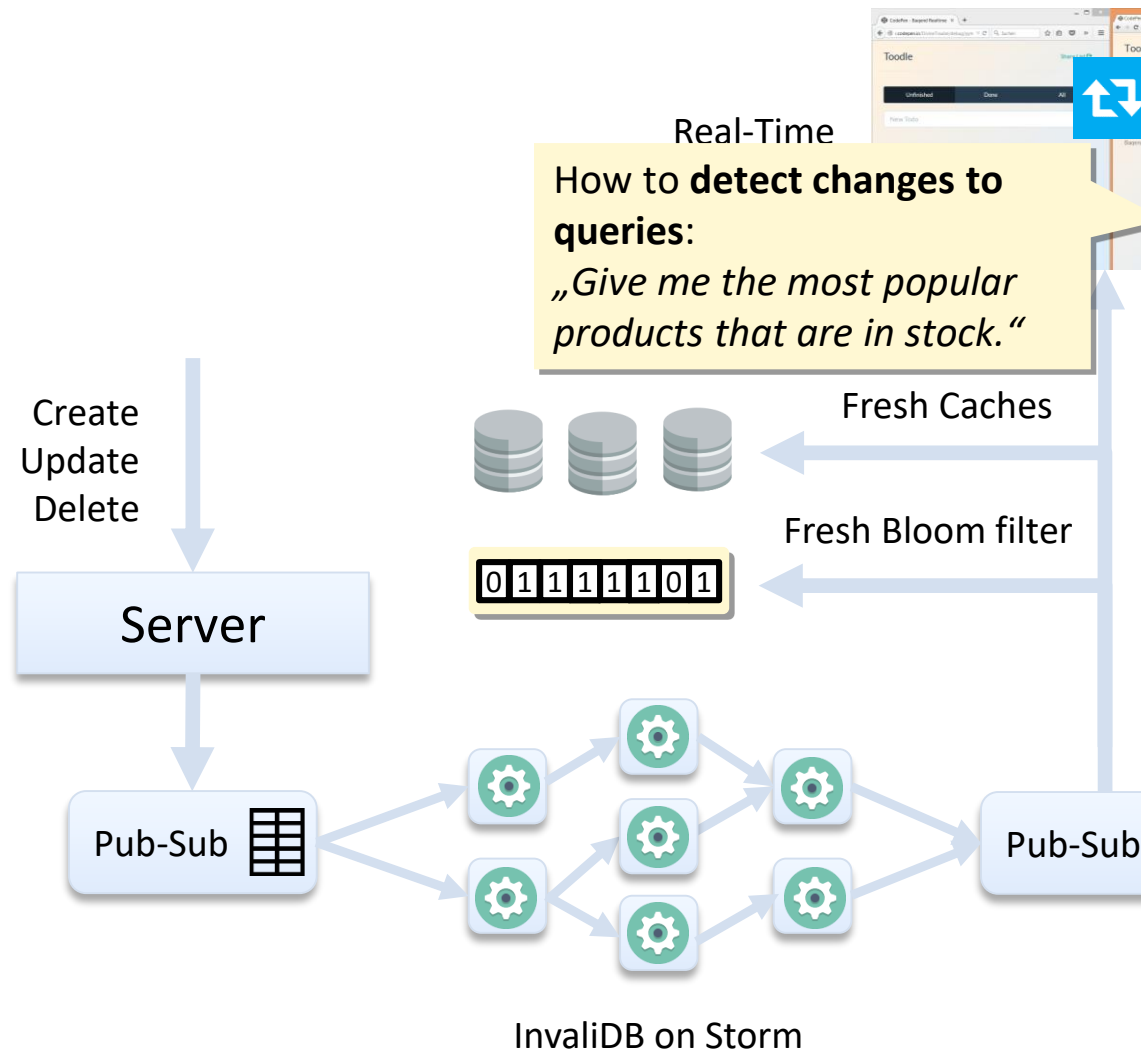
A blurred background image of a financial candlestick chart on a computer screen. The chart features blue and yellow candlesticks and a yellow moving average line. A semi-transparent grey box with a red vertical bar on the left is overlaid on the chart. Inside the box, the text 'LIVE QUERIES' is at the top, followed by 'Going Real-Time' and a circular progress indicator that is approximately 75% full. The overall image has a bokeh effect, with out-of-focus lights visible on the right side.

LIVE QUERIES

Going Real-
Time 

Going Real-Time

Query Caching & Subscribing



DEAL OF THE DAY

\$10.25 - \$179.99

Ends in 16:45:48

Up to 50% Off Handbags

★★★★★ 21

See details



DEAL OF THE DAY

\$97.99

List: ~~\$149.95~~ (35% off)

Ends in 16:45:48

[Save on Hitachi Gas Powered Leaf Blower](#)

Ships from and sold by Amazon.com.

★★★★★ 1961

Add to Cart



\$15.63 - \$16.79

9% Claimed

Ends in 4:40:49

[BESTEK surge protector](#)

Sold by BESTEK, and Fulfilled by Amazon.

★★★★★ 162

Choose options



\$18.66

Price: ~~\$39.99~~ (53% off)

18% Claimed

Ends in 3:05:49

[AUKEY Table Lamp, Touch Sensor Bedside Lamp + Dimmable War...](#)

Sold by Aukey Direct and Fulfilled by Amazon.

★★★★★ 669

Add to Cart



Filter word, e.g. "http", "Java", "Baqend"



Real-Time

Static

Last result update at 15:51:21 (less than a second ago)

1. Conju.re (conju_re, 3840 followers) tweeted:
https://twitter.com/conju_re/status/859767327570702336

Congress Saved the Science Budget—And That's the Problem <https://t.co/UdrjNidakc>
<https://t.co/xlNjpEpKZG>

2. ねぼすけゆーだい (Yuuu__key, 229 followers) tweeted:
https://twitter.com/Yuuu__key/status/859767323384623104

けいきさんと PENGUIN RESEARCHのけいたくん がリプのやり取りしてる...

3. Whitney Shackley (bschneids11, 5 followers) tweeted:
<https://twitter.com/bschneids11/status/859767319534469122>

holy..... waiting for it so long 🍷 © <https://t.co/UdXcHJb7X3>

4. Lisa Schmid (LisaMSchmid, 67 followers) tweeted on #teamscs, and #scs...
<https://twitter.com/LisaMSchmid/status/859767317311500290>

Congrats to Matthew Kent, winner of the 26th #TeamSCSCoding Challenge.
<https://t.co/vx1o0WgJrZ> #SCSChallenge

5. Brian Martin Larson (Brian_Larson, 40 followers) tweeted on #teamscs, a...
https://twitter.com/Brian_Larson/status/859767317303001089

Congrats to Matthew Kent, winner of the 26th #TeamSCSCoding Challenge.



Filter word, e.g. "http", "Java", "Baqend"



Real-Time

Static

Last result update at 15:51:21 (less than a second ago)

1. Conju.re (conju_re, 3840 followers) tweeted:
https://twitter.com/conju_re/status/859767327570702336

Congress Saved the Science Budget—And That's the Problem
<https://t.co/UdrjNidakc> <https://t.co/xlNjpEpKZG>

2. ねぼすけゆーだい (Yuuu__key, 229 followers) tweeted:
https://twitter.com/Yuuu__key/status/859767323384623104

けいきさんと PENGUIN RESEARCHのけいたくん がリプのやり取りしてる...

3. Whitney Shackley (bschneids11, 5 followers) tweeted:
<https://twitter.com/bschneids11/status/859767319534469122>

holy..... waiting for it so long 🍷 © <https://t.co/UdXcHJb7X3>

4. Lisa Schmid (LisaMSchmid, 67 followers) tweeted on #teamscs, and...
<https://twitter.com/LisaMSchmid/status/859767317311500290>

Congrats to Matthew Kent, winner of the 26th #TeamSCSCoding Challenge.
<https://t.co/vx1o0WgJrZ> #SCSChallenge

5. Brian Martin Larson (Brian_Larson, 40 followers) tweeted on #teams...
https://twitter.com/Brian_Larson/status/859767317303001089

Congrats to Matthew Kent, winner of the 26th #TeamSCSCoding Challenge.

Programming Real-Time Queries

JavaScript API

```
var query = DB.Tweet.find()  
  .matches('text', /my filter/)  
  .descending('createdAt')  
  .offset(20)  
  .limit(10);
```

Static Query

```
query.resultList(result => ...);
```

Google

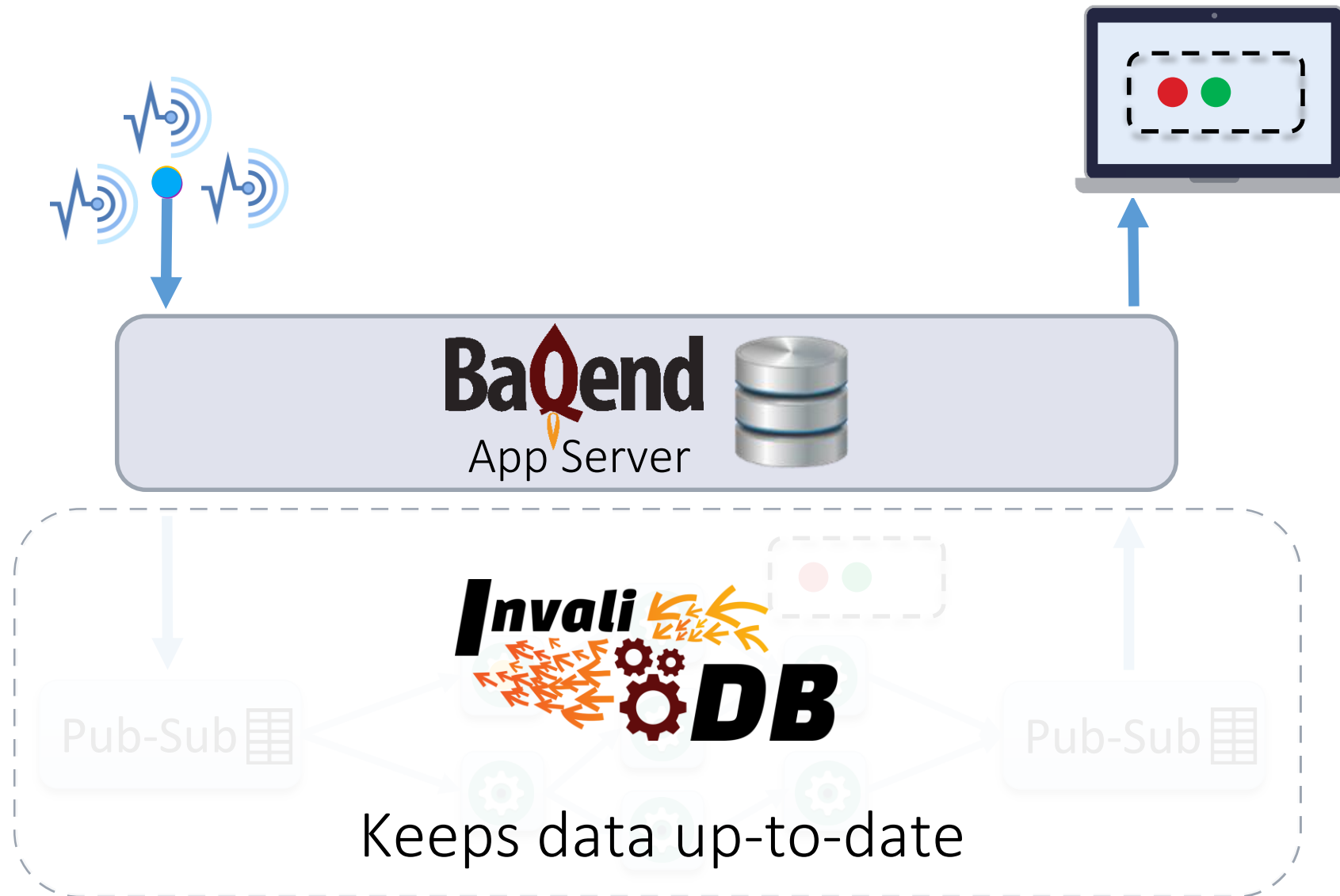
Real-Time Query

```
query.resultStream(result => ...);
```

Twoogle

Baqend Streaming Queries

Real-Time Architecture



InvaliDB

Filter Queries: Distributed Query Matching

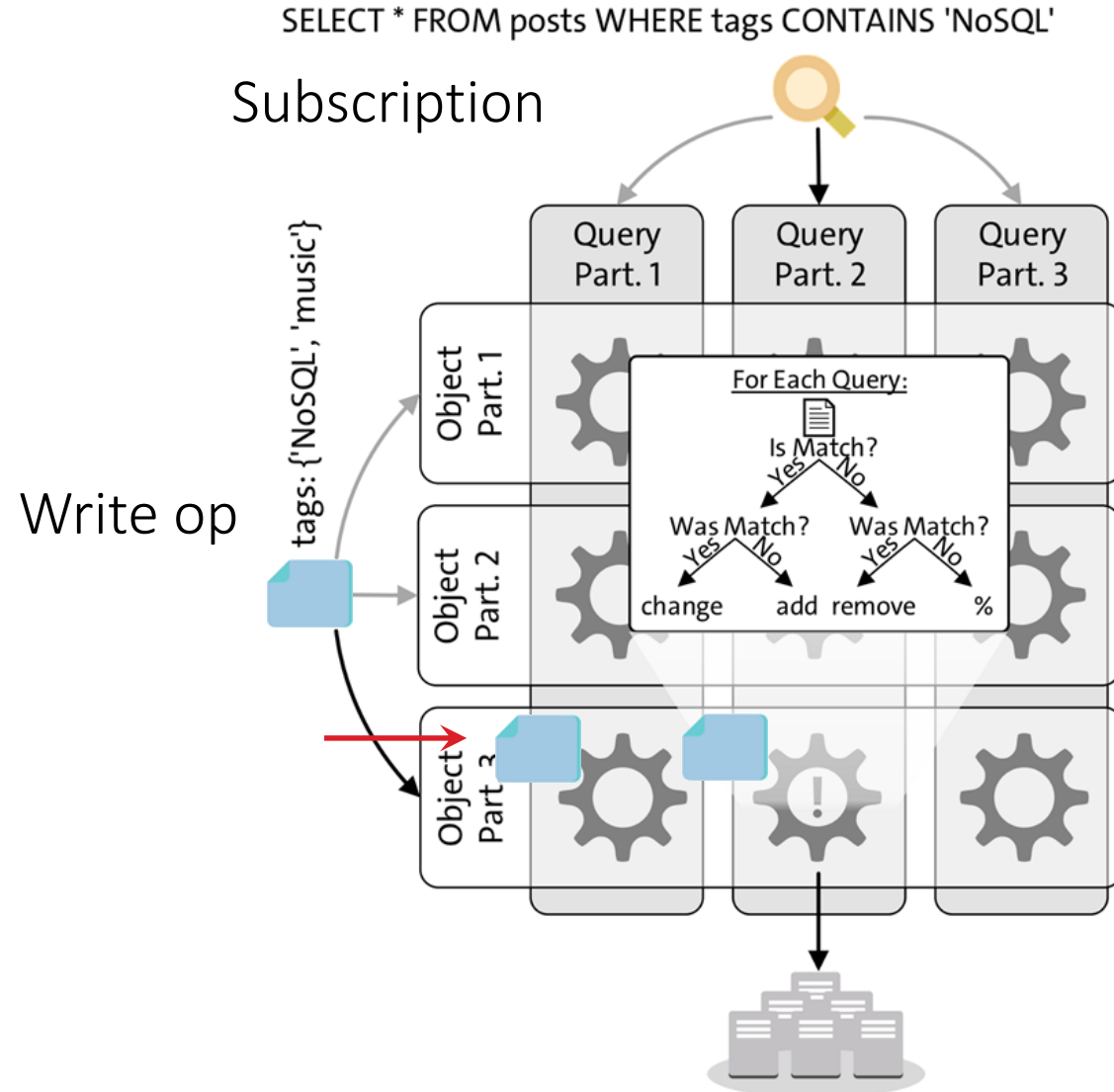
Two-dimensional partitioning:

- *by Query*
- *by Object*

→ scales with queries and writes

Implementation:

- Apache Storm & Java
- MongoDB query language
- Pluggable engine



SERVICE WORKERS

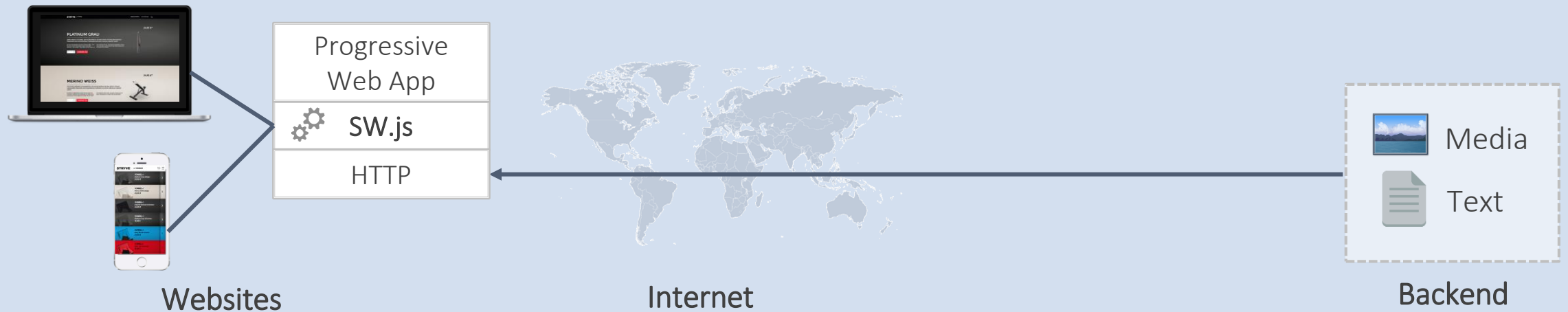
How to accelerate leagacy systems?



New Standard: Service Workers

What they do:

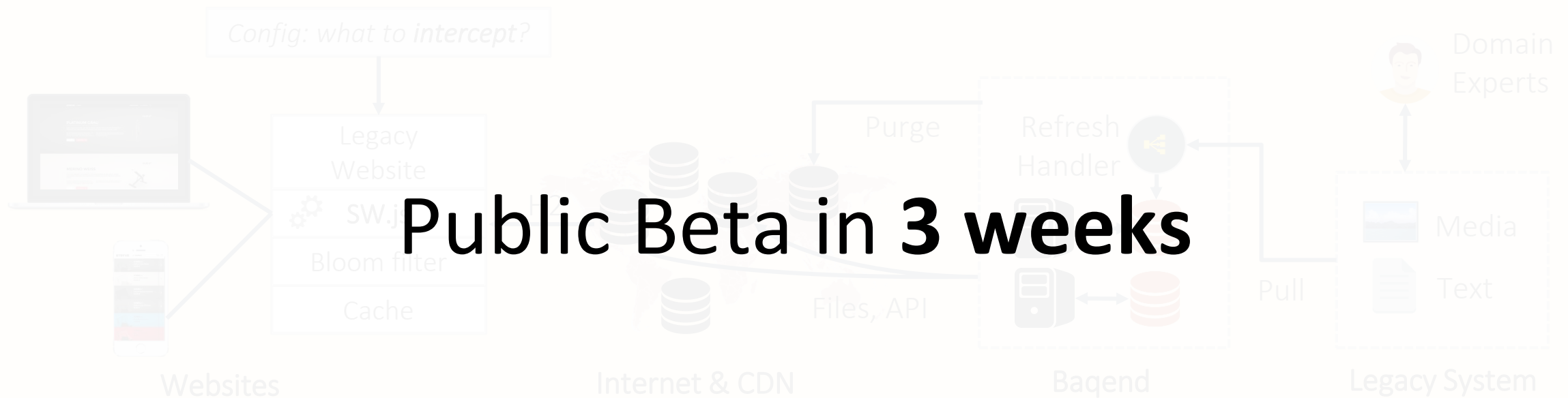
- **Proxy** any HTTP request
- **Offline Cache** for „progressive web apps“

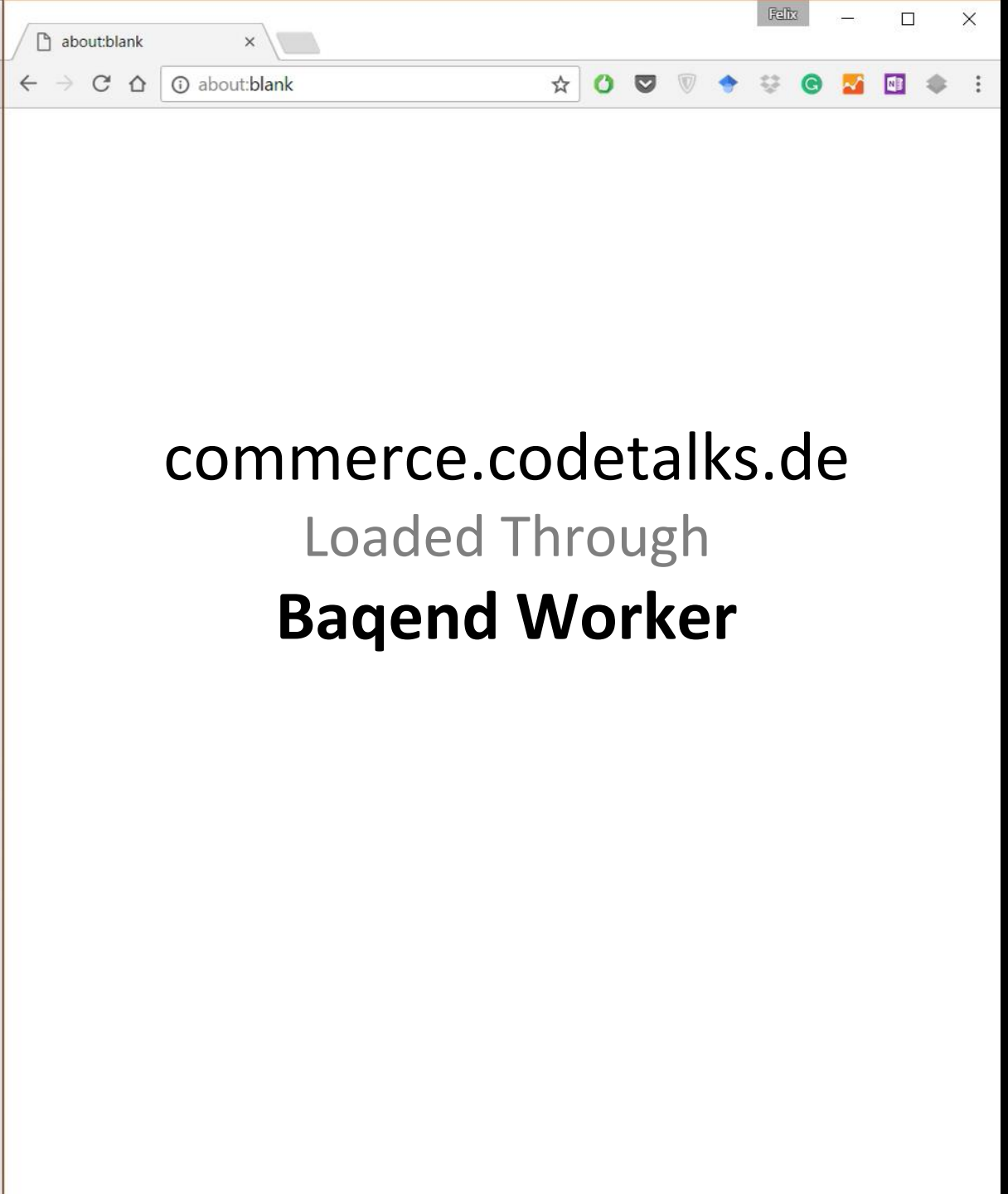


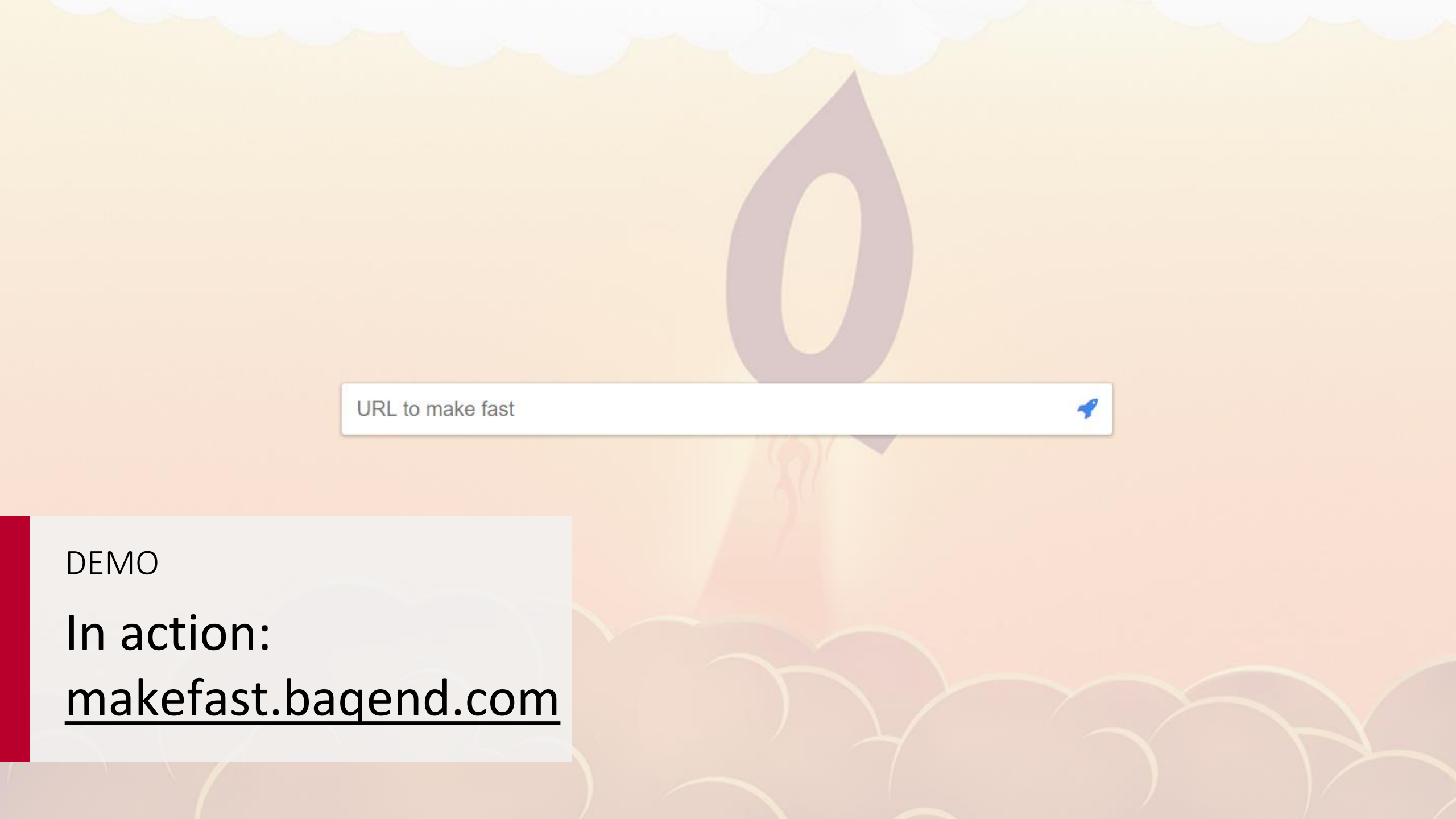
Service Workers + Baqend Caching

Baqend Worker:

- **Redirect** requests to Baqend for faster delivery by including a **snippet**
- **Update** of cached data: refresh of stored data against origin







URL to make fast



DEMO

In action:
makefast.baqend.com

Lessons Learned

Building a Scalable BaaS



Frontend



- **Single-page applications** are fast for navigation
- **Good Tooling** for optimizations (Inlining, Above-the-fold, Minification, etc.)

Network



- **Caching** in the CDN and browser
- **Dynamic data** should also be cached
- **Minimize latency**, SSL and HTTP tuning

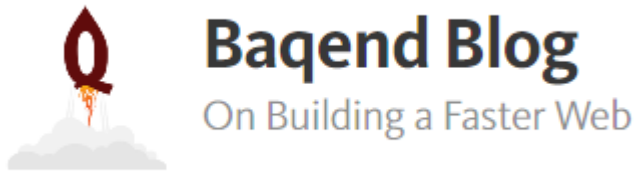
Backend



- Horizontal **scaling** with stateless web servers
- **NoSQL** databases
- **Cloud**-hosted
- **Load-Tests** important
- **Failover** and **autoscaling**



Further Reading Resources



<https://medium.baqend.com/>

- **Web Performance** in a Nutshell: HTTP/2, CDNs and Browser Caching
- High Performance Website Hosting with **SSL and HTTP/2** Made Simple
- The AWS and MongoDB Infrastructure of Parse: Lessons Learned
- NoSQL Databases: a Survey and Decision Guidance



<https://www.baqend.com/>

- **1.** Start a free app
- **2.** Check out the starter kits and tutorials
- **3.** Develop web application



>10x
Faster Loads



Automatic
Scaling



Faster
Development



BaQend

For a web without
loading times.

www.baqend.com
[@Baqendcom](https://twitter.com/Baqendcom)