Going for Speed: Full-Stack Performance Engineering in Modern Web-Based Applications

Organizers:

- Wolfram Wingerath[†] (wolfram.wingerath@baqend.com)
- Benjamin Wollmer[†§] (benjamin.wollmer@baqend.com)
- Felix Gessert[†] (felix.gessert@baqend.com)
- Stephan Succo[†] (stephan.succo@baqend.com)
- Norbert Ritter[§] (norbert.ritter@informatik.uni-hamburg.de)

[†] Baqend, Hamburg, Germany (https://baqend.com)
[§] University of Hamburg (Databases and Information Systems, DBIS), Hamburg, Germany (https://vsis-www.informatik.uni-hamburg.de)

Introduction

Our tutorial addresses the 4 primary challenges of developing fast Web applications (cf. Figure 1).

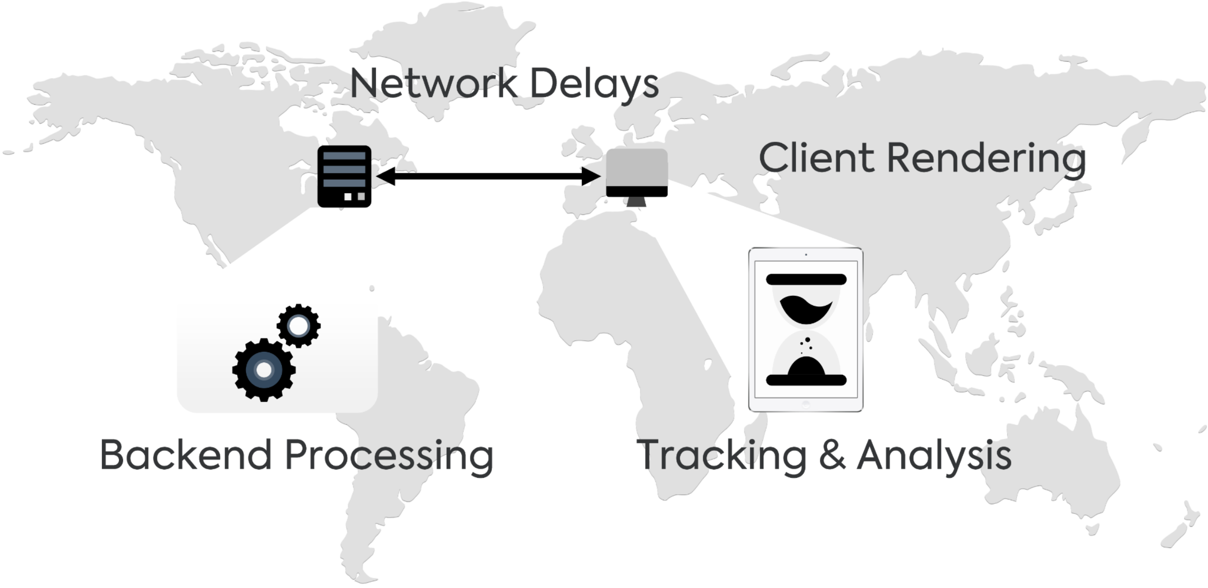

Figure 1: Page load times in Web-based applications are dominated by performance in the backend, network, and frontend. Performance tracking and analysis are required for identifying issues and evaluating the success of technical optimizations.

First, **backend processing** is mandatory for many client requests as content today is usually tailored to individual users. Rendering a website therefore does not only require fetching generic resources like images and stylesheets, but also accessing databases for personalized information such as recently viewed items or product recommendations. As the second challenge, **network delays** contribute to the overall time it takes to deliver content from the Web server to the end user's device. Content delivery networks (CDNs) are often used to retrieve data from closeby edge nodes instead of the faraway origin server. However, the dynamic nature of personalized websites prohibits traditional Web caching and thus impedes fast content delivery when server and client are physically separated. The third challenge is **client rendering**, which includes executing scripts on the user device and creating visuals in the viewport. Optimizing rendering efficiency is difficult, because it requires minimizing the number and size of critical resources and requesting them in the order in which the browser processes them. As a fourth challenge, performance **tracking & analysis** on different levels of the application stack is required to monitor the user experience and correlate it with business key performance indicators (KPIs). But since tracking introduces an overhead by itself, measuring performance without introducing additional delays can be just as challenging as improving it.

Structure & Overview

The tutorial is designed as a half-day lecturing style presentation and is structured into 4 parts:

1. Efficient Frontend Design: The key principle of frontend optimization is to make the browser's work as easy as possible. In this part, we will therefore explain what exactly happens when the browser loads a website and how this process can be accelerated. We will discuss several best practices for optimizing the critical rendering path (CRP), minimizing page weight, server-side image transcoding, and for getting the most out of browser or CDN caching. A particular focus will be on the relatively new Service Worker standard and the myriad of possibilities it adds to Web application development, such as add-to-homescreen functionality, the offline mode, push notifications, and different caching strategies (“Progressive Web Apps”). In doing so, we will provide background information on scopes, lifecycles, persistence, and other pivotal Service Worker concepts, as well as exciting features and APIs that are yet to come (e.g. for payment, sharing, and speech processing).

2. High-Performance Networking: This part starts with the basics of how the browser interacts with the network and how it handles requests. We will then deep-dive into the Internet protocol stack and discuss performance optimizations for TCP (e.g. fast-open, congestion and flow control), TLS (e.g. OCSP stapling, dynamic record sizing), CDNs (e.g. early TLS termination, warm backend connections), HTTP (REST, Web caching), and compression (e.g. Gzip, Brotli). Finally, we will dissect the HTTP/2 standard and its most significant improvements over its predecessor (e.g. multiplexing, server push, resource prioritization, and header compression). To provide a comprehensive wrap-up, we close with considerations on how to upgrade from HTTP/1.1 to HTTP/2, which performance recommendations are still valid, and which best practices have become anti-patterns.

3. Scalable Backend Architectures: For ideal performance, the backend has to produce a response for every incoming request at all times and despite a large number of concurrent users and possible network outages, machine failures, or other error scenarios. This part of the tutorial therefore provides a broad overview over data management and processing options in the backend. In more detail, we recall the basics of distributed data management (e.g. partitioning, replication, eventual consistency, NoSQL data models) and present a brief discussion of significant impossibility results (CAP and PACELC theorems). We then conduct an in-depth survey of the NoSQL landscape by discussing individual systems and their architectures, classifying them according to functional and non-functional properties in a framework we call the *NoSQL Toolbox*. Acknowledging Web applications with real-time requirements, we also cover systems specialized for push-based data access: After a short historical recap of push-based mechanisms in data management, we dissect popular real-time databases and stream processing frameworks w.r.t. their capabilities in storing, querying, and analyzing data with low latency.

4. Performance Tracking & Analysis: In this part of the tutorial, we explore the relationship between technical website performance and online business success. First, we focus on user expectations to examine when they are most likely to engage or buy. We then assume the business perspective and explore the correlation between technical performance measures such as the Google Web Vitals and pivotal business KPIs (e.g. conversion rate or revenue). Having thus established the business impact of Web performance, we turn to the challenge of measuring it correctly. To this end, we highlight the differences between technical and user-centric metrics and then compare traditional synthetic performance tests, tracking, and log analysis with more advanced approaches like CRUX data analysis and real-user monitoring. We conclude with findings from our ongoing study on how Web performance gains drive up user engagement and business success.

Literature

[1] Tammy Everts. 2016. Time Is Money: The Business Value of Web Performance (1st ed.). O'Reilly Media, Inc.
[2] Christos Faloutsos, Jan Gasthaus, Tim Januschowski, and Yuyang Wang. 2019. Classical and Contemporary Approaches to Big Time Series Forecasting. In Proceedings of the 2019 International Conference on Management of Data (Amsterdam, Netherlands) (SIGMOD '19). Association for Computing Machinery, 6.
[3] Steffen Friedrich, Wolfram Wingerath, Felix Gessert, and Norbert Ritter. 2014. NoSQL OLTP Benchmarking: A Survey. In 44. Jahrestagung der Gesellschaft für Informatik, Informatik 2014, Big Data - Komplexität meistern, 22.-26. September 2014 in Stuttgart, Deutschland. 693–704. http://subs.emis.de/LNI/Proceedings/Proceedings232/article216.html.
[4] Felix Gessert. 2019. Mobile Site Speed and the Impact on E-Commerce. CodeTalks (2019). https://www.youtube.com/watch?v=RTt1RfMUvOQ).
[5] Felix Gessert, Michael Schaarschmidt, Wolfram Wingerath, Steffen Friedrich, and Norbert Ritter. 2015. The Cache Sketch: Revisiting Expiration-based Caching in the Age of Cloud Data Management. In Datenbanksysteme für Business, Technologie und Web (BTW), 16. Fachtagung des GI-Fachbereichs "Datenbanken und Informationssysteme" (DBIS), 4.-6.3.2015 in Hamburg, Germany. Proceedings. 53–72. http://subs.emis.de/LNI/Proceedings/Proceedings241/article6.html
[6] Felix Gessert, Michael Schaarschmidt, Wolfram Wingerath, Erik Witt, Eiko Yoneki, and Norbert Ritter. 2017. Quaestor: Query Web Caching for Database-as-a-Service Providers. Proceedings of the 43rd International Conference on Very Large Data Bases (2017), 12.
[7] Felix Gessert and Wolfram Wingerath. 2020. Batching was Yesterday! Real-Time Tracking For 100+ Million Visitors. AWS Immersion Days (2020). https://www.youtube.com/watch?v=avPcOFzUa1Q.
[8] Felix Gessert, Wolfram Wingerath, Steffen Friedrich, and Norbert Ritter. 2016. NoSQL Database Systems: A Survey and Decision Guidance. Computer Science - Research and Development (2016).
[9] Felix Gessert, Wolfram Wingerath, and Norbert Ritter. 2017. Scalable Data Management: An In-Depth Tutorial on NoSQL Data Stores. In Datenbanksysteme für Business, Technologie und Web (BTW 2017) - Workshopband, 2.-3. März 2017, Stuttgart, Germany (LNI), Vol. P-266. GI, 399–402.
[10] Felix Gessert, Wolfram Wingerath, and Norbert Ritter. 2020. Fast & Scalable Cloud Data Management. Springer International Publishing.
[11] Ilya Grigorik. 2013. High Performance Browser Networking. O'Reilly Media, Inc.
[12] Liangjie Hong and Mounia Lalmas. 2019. Tutorial on Online User Engagement: Metrics and Optimization. In Companion Proceedings of The 2019 World Wide Web Conference (WWW '19). Association for Computing Machinery.
[13] Zoi Kaoudi and Jorge-Arnulfo Quiané-Ruiz. 2018. Cross-Platform Data Processing: Use Cases and Challenges. In 34th IEEE International Conference on Data Engineering, ICDE 2018, Paris, France, April 16-19, 2018. IEEE Computer Society, 1723–1726.
[14] Rishabh Mehrotra, Emine Yilmaz, and Ahmed Hassan Awadallah. 2018. Tutorial on Understanding & Inferring User Tasks and Needs. In Companion Proceedings of The 2018 World Wide Web Conference (WWW '18). Association for Computing Machinery.
[15] Davide Mottin, Matteo Lissandrini, Yannis Velegrakis, and Themis Palpanas. 2017. New Trends on Exploratory Methods for Data Analytics. Proc. VLDB Endow. 10, 12 (Aug. 2017), 1977–1980.
[16] Fatemeh Nargesian, Erkang Zhu, Renée J. Miller, Ken Q. Pu, and Patricia C. Arocena. 2019. Data Lake Management: Challenges and Opportunities. Proc. VLDB Endow. 12, 12 (Aug. 2019), 1986–1989.
[17] Addy Osmani and Ilya Grigorik. 2018. Speed is now a landing page factor for Google Search and Ads. Google Developers Blog (Web Updates) (2018). https://developers.google.com/web/updates/2018/07/search-ads-speed.
[18] Fatma Özcan, Yuanyuan Tian, and Pinar Tözün. 2017. Hybrid Transactional/Analytical Processing: A Survey. In Proceedings of the 2017 ACM International Conference on Management of Data (Chicago, Illinois, USA) (SIGMOD '17). Association for Computing Machinery, New York, NY, USA, 1771–1775.
[19] Christoph Luetke Schelhowe, Shuhei Kagawa, Thorbjoern Gruda, Jeff Cybulski, and David Martin Jones. 2018. Loading Time Matters. Zalando Engineering Blog (2018). https://engineering.zalando.com/posts/2018/06/loading-time-matters.html.
[20] Kristian Sköld. 2019. Web Performance Management with BI: Faster Business Growth through Speed. Minubo Webinar (2019). https://vimeo.com/357797550).

[21] Riccardo Tommasini, Sherif Sakr, Emanuele Della Valle, and Hojjat Jafarpour. 2020. Declarative Languages for Big Streaming Data. In Proceedings of the 23nd International Conference on Extending Database Technology, EDBT 2020, Copenhagen, Denmark, March 30 - April 02, 2020. OpenProceedings.org, 643–646.
[22] Romain Vuillemot and Jeremy Boy. 2018. Tutorial on Data Visualization for the Web. In Companion Proceedings of The 2018 World Wide Web Conference (WWW'18). Association for Computing Machinery.
[23] Wolfram Wingerath, Felix Gessert, Steffen Friedrich, and Norbert Ritter. 2016. Real-time stream processing for Big Data. it - Information Technology 58, 4 (2016), 186–194. https://doi.org/10.1515/itit-2016-0002.
[24] Wolfram Wingerath, Felix Gessert, and Norbert Ritter. 2019. NoSQL & Real-Time Data Management in Research & Practice. In Datenbanksysteme für Business, Technologie und Web (BTW 2019), 18. Fachtagung des GI-Fachbereichs „Datenbanken und Informationssysteme" (DBIS), 4.-8. März 2019, Rostock, Germany, Workshopband. 267–270. https://doi.org/10.18420/btw2019-ws-28.
[25] Wolfram Wingerath, Felix Gessert, and Norbert Ritter. 2020. InvaliDB: Scalable Push-Based Real-Time Queries on Top of Pull-Based Databases. In 36th IEEE International Conference on Data Engineering, ICDE 2020, Dallas, Texas, 2020.
[26] Wolfram Wingerath, Felix Gessert, and Norbert Ritter. 2020. InvaliDB: Scalable Push-Based Real-Time Queries on Top of Pull-Based Databases (Extended). Proceedings of the 46th International Conference on Very Large Data Bases (2020).
[27] Wolfram Wingerath, Felix Gessert, Erik Witt, Steffen Friedrich, and Norbert Ritter. 2018. Real-Time Data Management for Big Data. In Proceedings of the 21th International Conference on Extending Database Technology, EDBT 2018, Vienna, Austria, March 26-29, 2018. OpenProceedings.org.
[28] Wolfram Wingerath, Felix Gessert, Erik Witt, Hannes Kuhlmann, Florian Bücklers, Benjamin Wollmer, and Norbert Ritter. 2020. Speed Kit: A Polyglot & GDPR-Compliant Approach For Caching Personalized Content. In 36th IEEE International Conference on Data Engineering, ICDE 2020, Dallas, Texas, 2020.
[29] Wolfram Wingerath, Norbert Ritter, and Felix Gessert. 2019. Real-Time & Stream Data Management: Push-Based Data in Research & Practice. Springer International Publishing. https://doi.org/10.1007/978-3-030-10555-6.
[30] Erik Witt. 2017. Building a Shop with Sub-Second Page Loads. Code Talks Commerce (2017). https://baqend.com/files/code_talks_ecommerce.pdf.
[31] Erik Witt. 2018. The Technology Behind Progressive Web Apps. Techcamp (2018). https://baqend.com/files/techcamp-2018-erik-pwa.pdf.
[32] Erik Witt. 2019. Service Workers in Action: Building an Offline-Capable Webshop in under 90 minutes. Code Talks (2019). https://www.youtube.com/watch?v=6SnqETJ8MVU.
[33] Benjamin Wollmer, Wolfram Wingerath, and Norbert Ritter. 2020. Context-Aware Encoding & Delivery in the Web. In Web Engineering – 20th International Conference, ICWE 2020, Helsinki, Finland, June 9-12, 2020, Proceedings. Springer.