

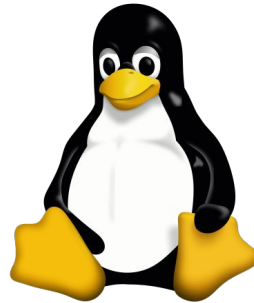
Webentwicklung

FWPM

Web Infrastruktur

LAMP Stack

- **L**inux, **A**pache, **M**ySQL, **P**HP/**P**erl/**P**ython
- DER klassische serverseitiger Web-Stack
- Besteht aus
 - Betriebssystem
 - Webserver
 - Programmiersprache
 - Persistenz/Datenbank
 - (in dieser Reihenfolge)
- In vielen Variationen denkbar



Betriebssystem (Linux)

- 1969 Start Unix
- 1991 Erster Release Linux
- 70-90% aller Web Hosting Server nutzen Unix/Linux als Betriebssystem [1]
- Technologische Basis des Stack
- Betriebssystem Wahl schränkt teilweise andere Komponenten ein
- Kaum Auswirkungen auf Arbeit als Webentwickler
 - Sprachen sind mittlerweile größtenteils nicht mehr systemnah
 - **ABER: Schnittstellen zum Betriebssystem mit Vorsicht genießen**
- DevOps sollten konsequent auf Linux setzen

Webserver (Apache)

- Der “Server” in Client/Server
- Liefert Inhalte aus
- Drei bekannte Server (teilen sich das Web unterschiedlich[2][3])
 - Apache HTTP Server (1995)
 - Flexibel aber langsam
 - NGINX (2004)
 - Schnell aber schlicht
 - IIS (1995)
 - Von Microsoft

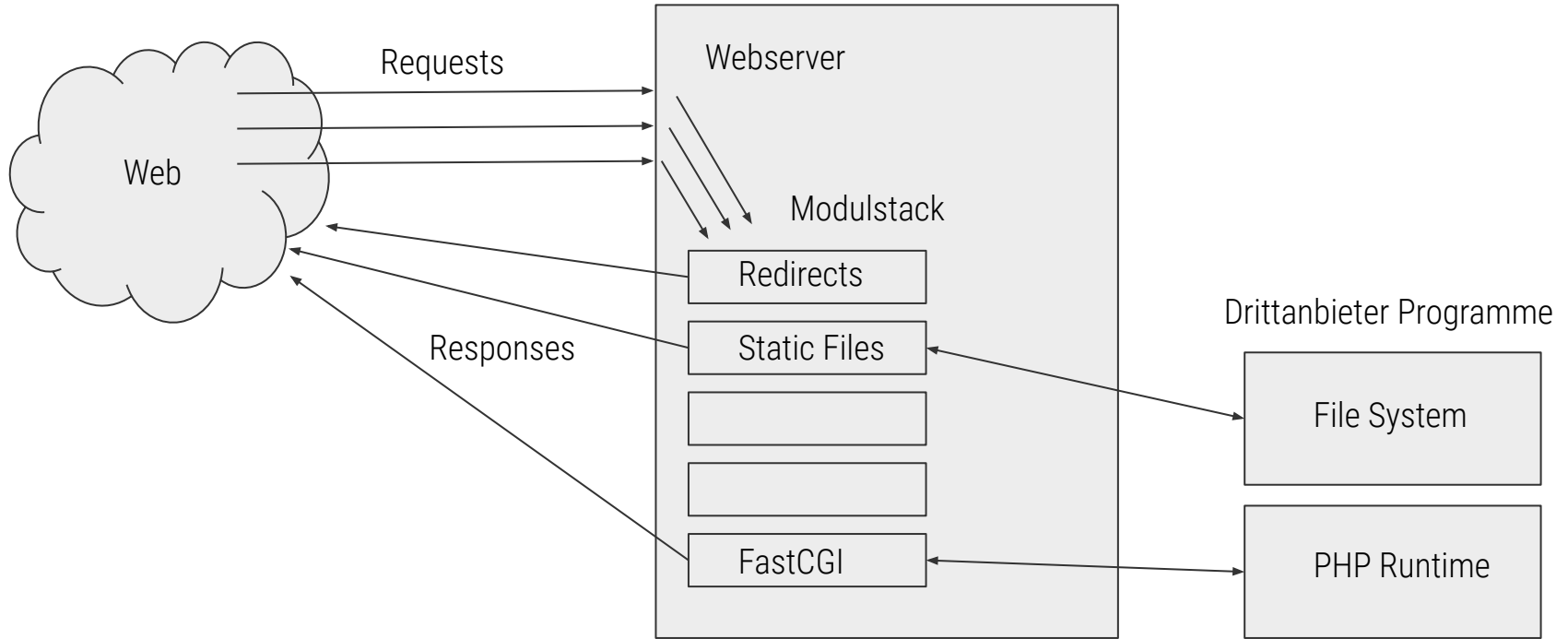
Webserver - Funktion

- Nimmt Requests an und verwaltet Verbindungen
- Anwendungs-agnostisch
- Übersetzt Request und bereitet ihn auf (Schicht 6 OSI Modell)
 - Encryption/Decryption
 - Daten Kompression
- Kann Request selbst verarbeiten
 - Z.B. bei statischen Files
- Kann Anfragen weiterleiten
 - Z.B. bei dynamischen Inhalten

Webserver - Funktionalität

1. Annahme Request
 - a. Verbindungsmanagement
2. Aufbereiten Request
 - a. Z.B. URL zerlegen
 - b. Oder HTTPS Decryption
3. Verarbeitungslogik
 - a. Geschieht nach Konfiguration
 - b. Kann Anfragen an Drittsoftware weiterleiten
4. Aufbereitung Response

Webserver - Funktionalität



Persistenz (MySQL)

- Meistens in Form von SQL Datenbanken
 - MySQL (1995)
- Erlaubt das dynamische Speichern von Daten
 - Theoretisch in beliebiger Form
 - Baustein **weg vom Stateless Web**
- Ansteuerung über Anwendungscode
 - Frei durch Webentwickler gestaltbar
- Je nach Anforderung mehr oder weniger normalisiert
 - Art der Persistenz variiert stark nach Anwendungsfall

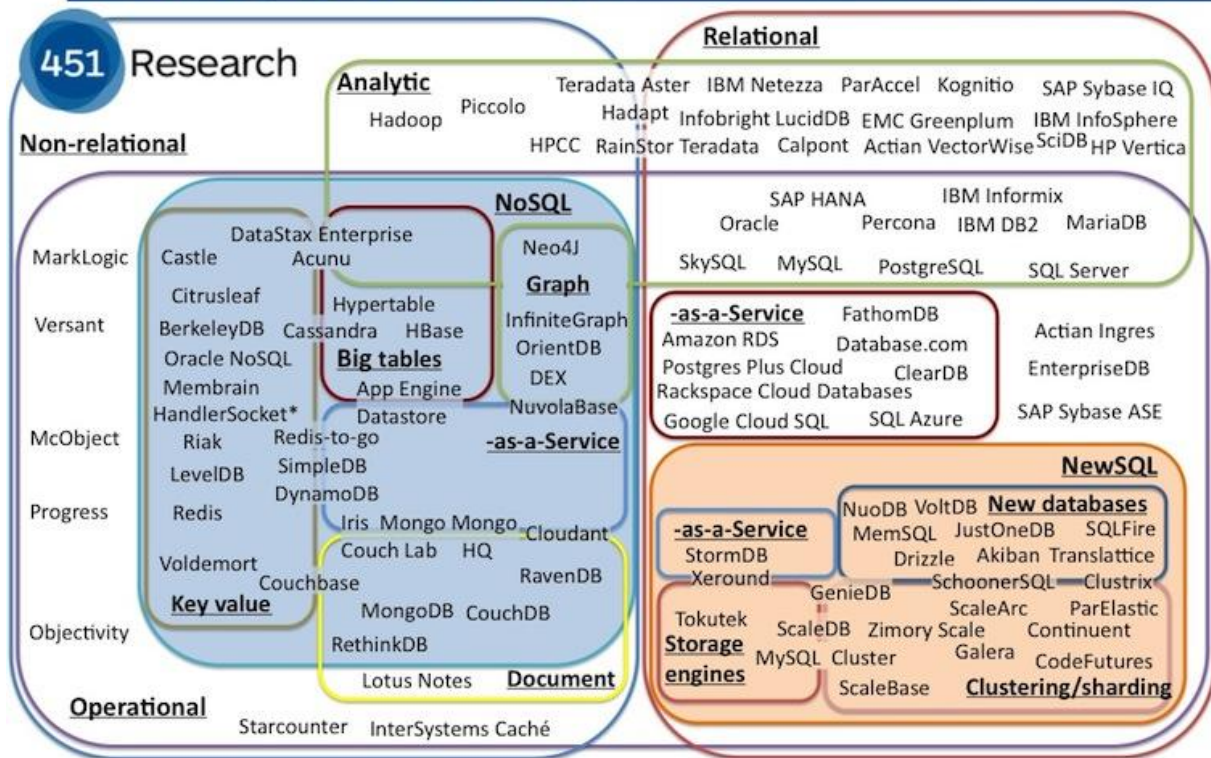
Persistenz - Arten

- Extrem weites, stark spezialisiertes Feld
- Relational/SQL
 - Integrität, Struktur und universelle Einsetzbarkeit
 - Z.B. MySQL/MariaDB, Oracle
- Non-Relational/NoSQL
 - Alles andere
 - Meist schemalos und performant (lesend)
 - Stark spezialisiert
 - Z.B. MongoDB, Elasticsearch, Redis, Neo4j



Persistenz - Arten

The evolving database landscape



Programmiersprache (PHP)

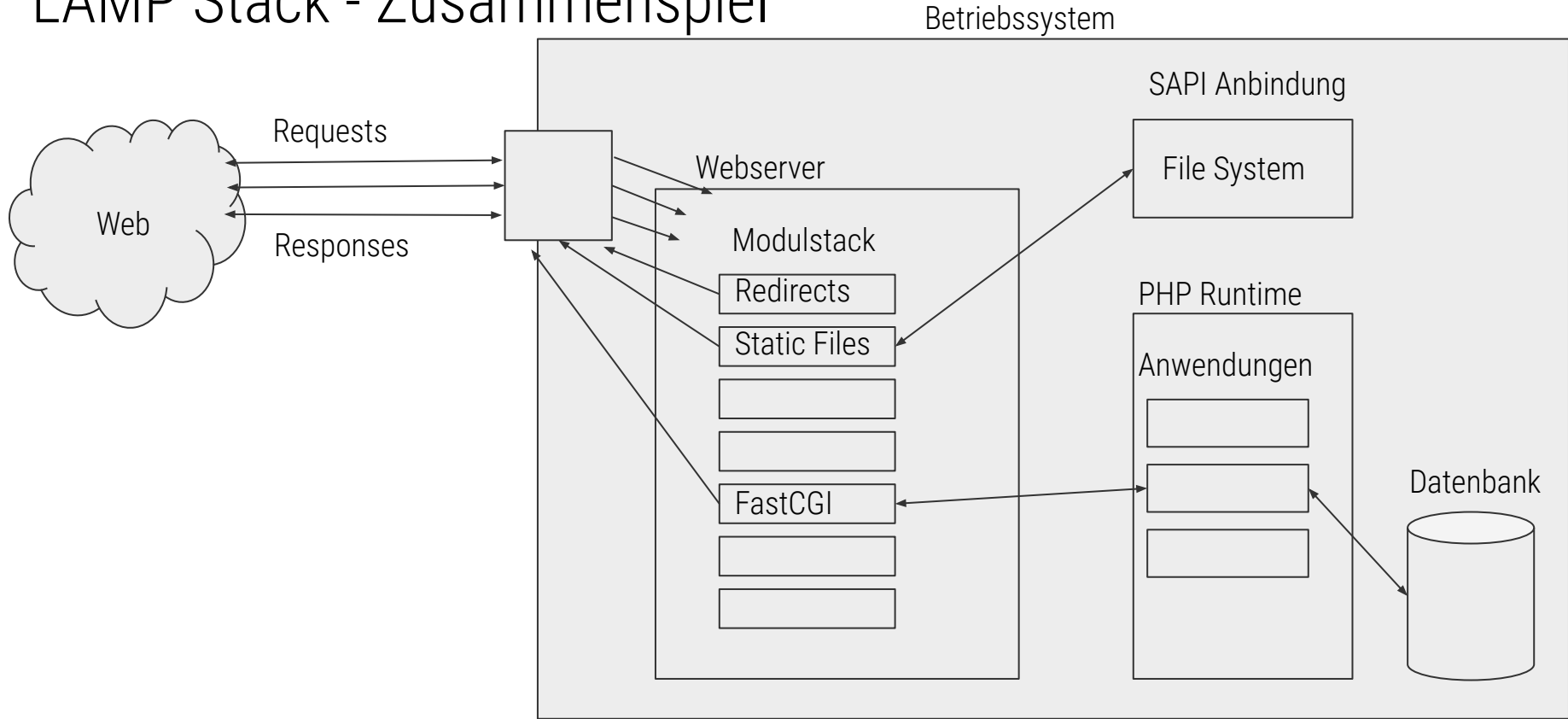
- **PHP: Hypertext Preprocessor** (1995)
 - In der Praxis nicht auf PHP eingegrenzt
 - 78% aller Webseiten nutzen PHP [4]
- Erlaubt **dynamische Inhalte** innerhalb von Webanwendungen
 - In Kombination mit anderen Sprachen (z.B. HTML)
- Hauptarbeit bei der Webentwicklung
- Bindeglied zwischen anderen Technologien
 - Z.B. Webdienste, Datenbank, etc.



Programmiersprache - Theorie

- Eine Programmiersprache ist
 - formale Sprache: abstrakte Sprache zur mathematischen Verwendung (Definition statt Kommunikation)
 - maschinenlesbar
 - beschreibt
 - Datenstrukturen
 - **Algorithmen**
- Eine Programmiersprache braucht
 - **Eine Laufzeitumgebung**
 - Schnittstellen zur Ein- und Ausgabe
 - Einen Satz an Sprachkonstrukten

LAMP Stack - Zusammenspiel



Caches

- In vielen verarbeitenden Schichten zu finden
 - Im Browser
 - Vor Webserver
 - In der Webanwendung
 - ...
- Caches ermöglichen sehr starke Performance Verbesserungen
- Cache Handling **ist kompliziert**
 - Was, wann cachen (und wann nicht mehr)
 - Nicht unter eigener Kontrolle (Browsercache!!)
 - Vielzahl von Caches unübersehbar

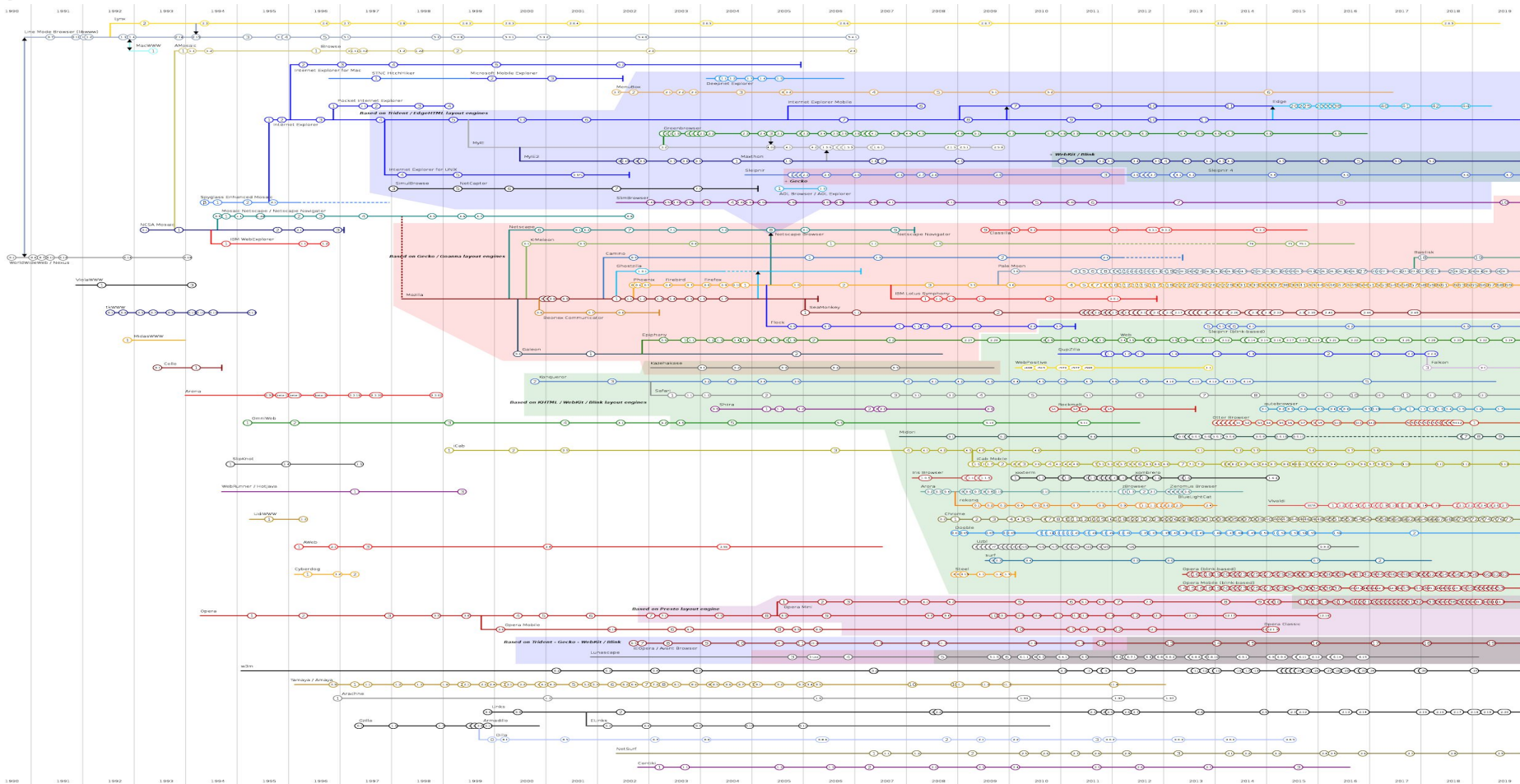
Proxy

- Stehen zwischen Webserver und Client
- Open Proxy
 - Z.B. Caching oder Anonymisierung
- Reverse Proxy
 - Vor-Verarbeitung vor Server
 - Z.B. Load Balancing
- Wenig bis kein Einfluss als Anwendungsentwickler
- Greifen in Kommunikation mit Client ein
 - Z.B. Änderung von Headern, Filterung von Inhalten
 - Caches wie Varnish sind auch Proxies

? `cache-control: public, max-age=14400`
? `content-encoding: gzip`
? `content-type: text/html; charset=UTF-8`
? `date: Thu, 09 Apr 2020 11:19:04 GMT`
? `last-modified: Wed, 25 Mar 2020 12:32:33 GMT`
? `server: thttpd/2.29 23May2018`
? `vary: Accept-Encoding`
? `via: 1.1 varnish (Varnish/6.4)`
`X-Firefox-Spdy: h2`
`x-varnish: 1838798 3114164`

Browser

- DER Client im Web
- Beinhaltet zwei wichtige Komponenten
 - **Render Engine** zur Anzeige von Inhalten
 - **Javascript** Laufzeitumgebung
- Vielzahl war lang großes Problem
 - Die sog. **Browserkriege** (1990er Jahre und ab 2003)
 - Nicht durch Entwickler beeinflussbar
 - Man musste Anwender kennen!
 - Erst in den letzten Jahren besser geworden (Lebwohl IE)
- Faustregel: Neu und bekannt sollte er sein



Quellen:

- [1] W3Techs, https://w3techs.com/technologies/overview/operating_system
- [2] W3Techs, https://w3techs.com/technologies/overview/web_server
- [3] Netcraft, <https://news.netcraft.com/archives/2014/02/03/february-2014-web-server-survey.html>
- [4] W3Techs, https://w3techs.com/technologies/overview/programming_language

Bildquellen:

- Apache Logo By The Apache Software Foundation, Apache License 2.0, <https://commons.wikimedia.org/w/index.php?curid=47190352>
- MySQL Logo By Source, Fair use, <https://en.wikipedia.org/w/index.php?curid=17119753>
- PHP Logo By Colin Viebrock, CC BY-SA 4.0, <https://commons.wikimedia.org/w/index.php?curid=9632398>
- Elasticsearch Logo By Elasticsearch - [1], Apache License 2.0, <https://commons.wikimedia.org/w/index.php?curid=78473677>
- MariaDB Logo Von MariaDB - [1], LGPL, <https://commons.wikimedia.org/w/index.php?curid=55946550>
- Types of Databases By Matthew Aslett 451 Research, https://blogs.the451group.com/information_management/2012/11/02/updated-database-landscape-graphic/
- List of Web Browsers By I, ADeveria, CC BY-SA 3.0, <https://commons.wikimedia.org/w/index.php?curid=2237030>