

Schneller und sicherer Surfen im Internet

Frank Hofmann

30. April 2016

- 1 Über den Referenten
- 2 Ausgangspunkt
- 3 Daten beziehen und Messen
- 4 Zeitfresser
- 5 Automatisierung
- 6 Ausblick und Links

OpenSource-Aktivitäten und Projekte



Chemnitzer
Linux-Tage
seit 2000



Brandenburger
Linux-Info-Tag
(BLIT)
2006-2012



seit 2009

Regionales
LUG-Treffen
Berlin-
Brandenburg
seit 2008



LinuxBus
Berlin-
Chemnitz
seit 2007

Über Hofmann EDV – Linux, Layout und Satz



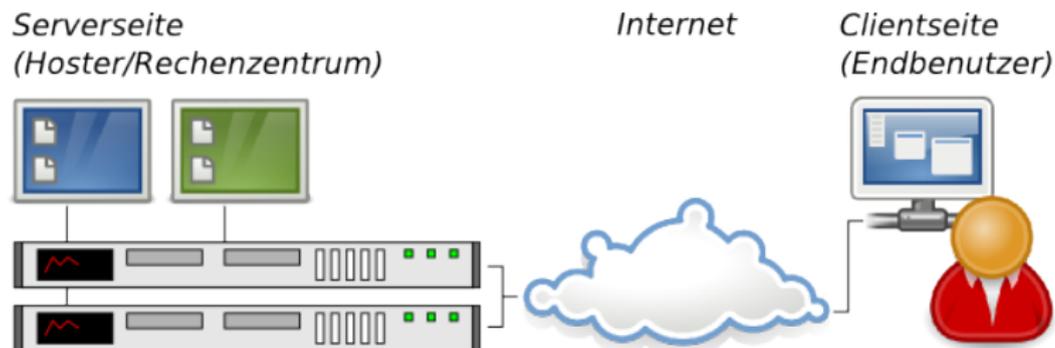
Linux, Layout & Satz



WIZARDS OF FOSS
Open Source Schulungen

- Layout und Satz, Druckvorstufe
- Administration und Service
 - Programmierung und Automatisierung auf der Basis von PHP und Python
 - Authentifizierung
- Trainings für IT-Spezialisten
Mitbegründer, Gesellschafter und Trainer

Was betrachten wir



- Server und Hosting
Webseite bereitstellen
- Redakteur
Inhalte einpflegen
- Benutzer
bezieht die Inhalte der Webseite via Webbrowser

Faktoren

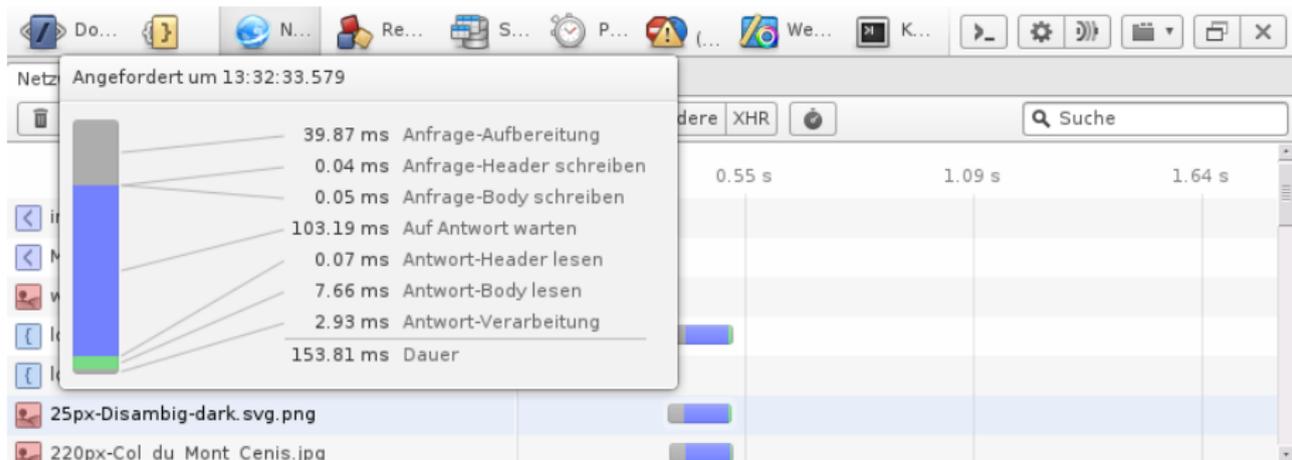
- Server und Hosting
 - Hardware
 - Netzanbindung (IPv4, IPv6)
 - DNS-Einträge
 - verfügbarer RAM
- Daten (die Webseite selbst) – statisch oder dynamisch
- Netzwerkverbindung (der Weg zum Webserver)
- Ausgabegerät – Monitor oder Smartphone
- Webbrowser
 - Encoding
 - Benutzersprache
 - verwendete Schriften
 - Bilder
 - nachgeladene Inhalte
 - Werbung
 - Pipelining

Die Webseite beziehen

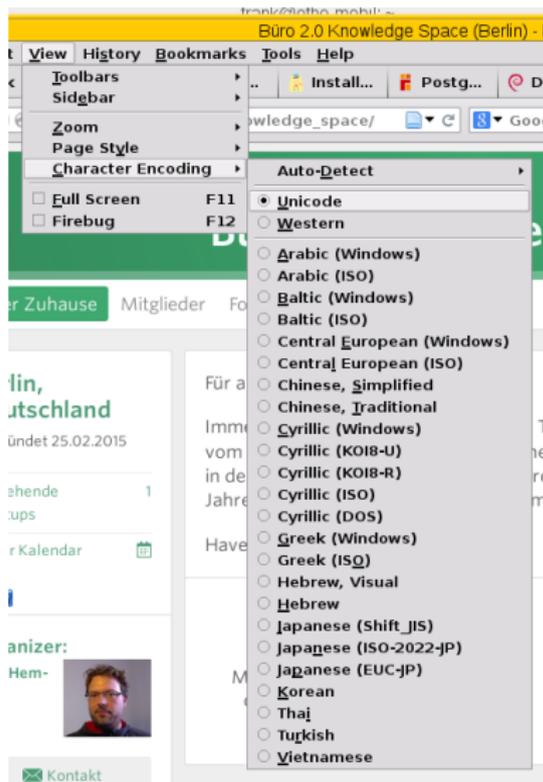
- 1 DNS-Verbindungsanfrage
- 2 HTML-Dokument als Datenstrom beziehen
- 3 alle im HTML-Dokument referenzierten, externen Daten beziehen
 - Bilder
 - CSS-Formatvorlagen
 - JavaScript-Dateien
 - Flash-Inhalte inkl. Vorschaubild
- 4 Rendering der bezogenen Inhalte und Ausgabe im Browserfenster

Bezugszeit messen

- Firefox/Iceweasel: Firebug
- Opera: DragonFly
- Chrome: Developer Extension



HTML und Encoding



• HTML

- falsche Reihenfolge
- vergessene, nicht geschlossene HTML-Tags
- vergessene Anführungszeichen der Attribute
- Browserweichen

• Encoding

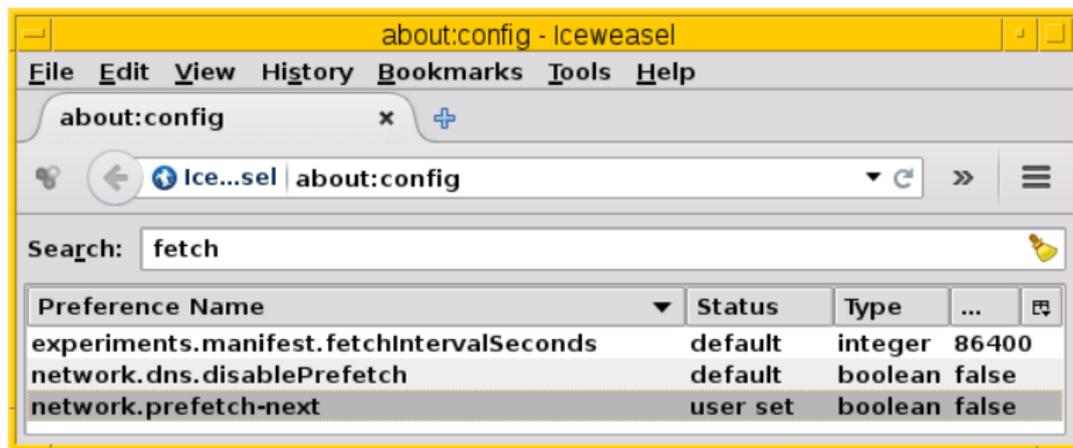
- Deklaration im HTML-Header
- gespeichertes Datenformat
- Einstellung im Webbrowser

Externe Elemente (Teil 1)

- CSS-Formatvorlagen
 - mehrere Dateien zu einer Vorlage zusammenfassen
 - welche CSS-Vorlagen werden tatsächlich benutzt?
- Werbung und Banner
bewirken etwa 70% der Datenpakete
- Gastbeiträge deaktivieren
 - Personalisierung der Inhalte abschalten
 - Inhalte von Flickr, Twitter, Blogroll, Facebook und Google entfernen

Externe Elemente (Teil 2)

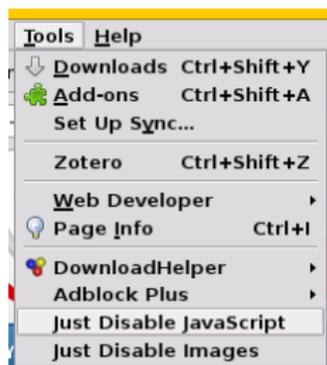
- vorausschauendes Laden von Inhalten abschalten
 - Laden von Inhalten im Hintergrund
 - Vorbereiten der Darstellung potentiell interessanter Inhalte
Darstellung geht dann deutlich flinker
 - belegt mehr Bandbreite
 - erzeugt Netzwerklast
 - erzeugt Einträge im Browsercache zu Webseiten, die Sie nie besucht haben



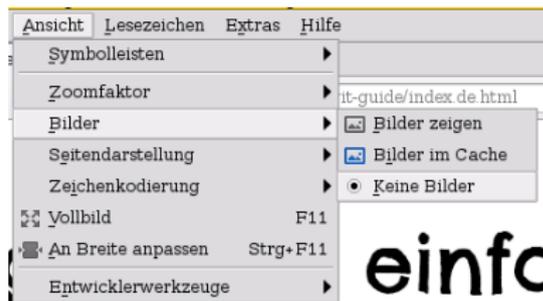
The screenshot shows the 'about:config' page in the Iceweasel browser. The search bar contains the text 'fetch'. Below the search bar, a table lists the search results:

| Preference Name | Status | Type | ... | |
|---|----------|---------|-------|--|
| experiments.manifest.fetchIntervalSeconds | default | integer | 86400 | |
| network.dns.disablePrefetch | default | boolean | false | |
| network.prefetch-next | user set | boolean | false | |

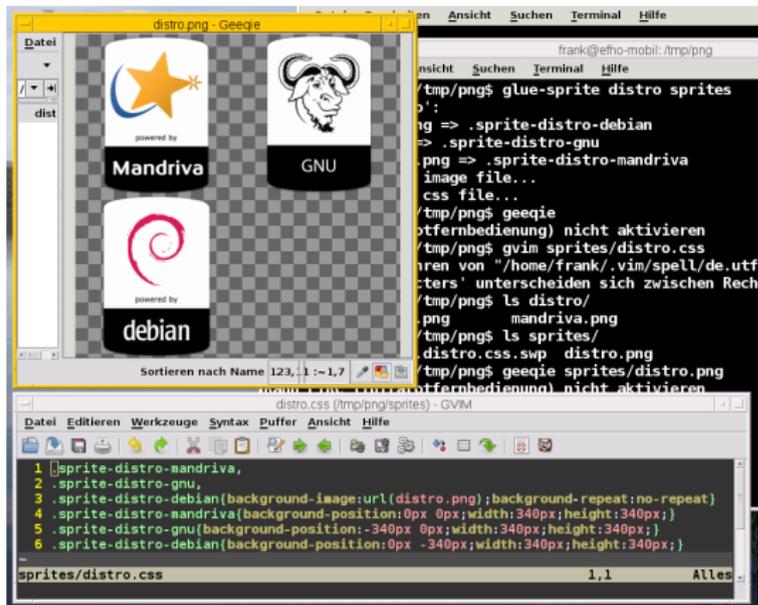
Externe Elemente: Bilder (Teil 3)



- Firefox/Iceweasel
 - pro Webseite abschalten
 - generell abschalten
 - Plugin *Just Disable Stuff*
- Opera
 - Menüeintrag Ansicht



Externe Elemente: Bilder optimieren (Teil 4)



- Bilder verkleinern, bspw. mit `optipng`
- mehrere Bilder mittels `glue-sprite` in einer einzigen Datei kombinieren
 - erzeugt Bild
 - ... plus passendes CSS zur Referenzierung des Ausschnitts

Sprache und Fonts

- die Sprache auswählen, mit der Sie am schnellsten die Inhalte erfassen
Vorzug der effizientesten Sprache
- Festlegung eines Standardfonts
 - gilt für serifenbehaftete und serifenlose Schriften
 - Nachladen von Schriften kostet Zeit

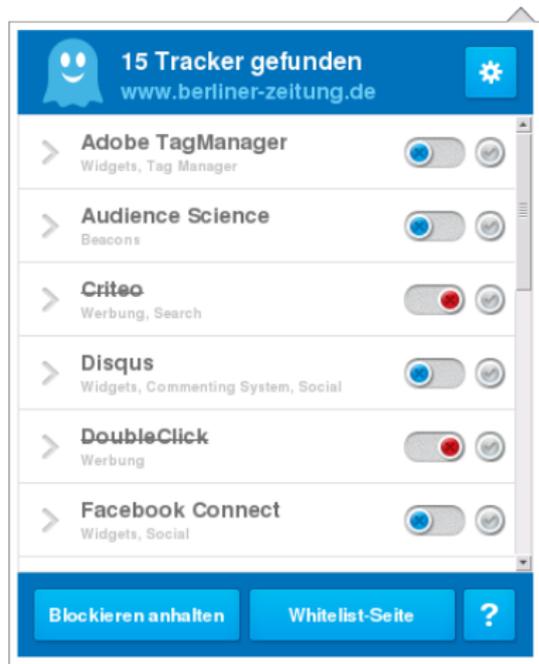
Pipelining

- paralleler Datentransfer über die gleiche Netzwerkverbindung
 - Firefox/Iceweasel: einschaltbar
 - Opera: aktiv (per Voreinstellung)
 - Chrome: einschaltbar

The screenshot shows the 'about:config' page in the Iceweasel browser. A search bar at the top contains the text 'pipelining'. Below the search bar, a table lists various preference names related to HTTP pipelining, along with their status, type, and value.

| Preference Name | Status | Type | Value |
|---|---------|---------|--------|
| network.http.pipelining | default | boolean | false |
| network.http.pipelining.abttest | default | boolean | false |
| network.http.pipelining.aggressive | default | boolean | false |
| network.http.pipelining.max-optimistic-requests | default | integer | 4 |
| network.http.pipelining.maxrequests | default | integer | 32 |
| network.http.pipelining.maxsize | default | integer | 300000 |
| network.http.pipelining.read-timeout | default | integer | 30000 |
| network.http.pipelining.reschedule-on-timeout | default | boolean | true |
| network.http.pipelining.reschedule-timeout | default | integer | 1500 |
| network.http.pipelining.ssl | default | boolean | false |
| network.http.proxy.pipelining | default | boolean | false |

Tracking und Spyware



- Analyse der Besucher einer Webseite
 - Wiederholungstäter
 - von welcher URL komme ich auf die Webseite?
- Nützliche Plugins (Auswahl)
 - *Just Disable Stuff*
 - *Adblock Plus*
 - *Ghostery*

Werkzeuge zur automatischen Erstellung

- Pandoc
- AsciiDoc
- T_EXto HTML translator
- plasT_EX
- HyperL_AT_EX
- ...

HTML-/XML-Inhalte validieren

- W3C Markup Validation Service
- XML Schema Validator
- Firefox-Plugins Validator und HTML Validator



JavaScript-Inhalte validieren

- JSLint (webbasiert)
- JSHint (webbasiert)
- acorn.js (für die Kommandozeile)
- ESLint (für die Kommandozeile)

```
1 function main() {  
2   return 'Hallo Welt!';  
3 }  
4  
5 main();  
6 main(variable);
```

CONFIGURE

Metrics

There is only one function in this file.
It takes no arguments.
This function contains only one statement.
Cyclomatic complexity number for this function is 1.

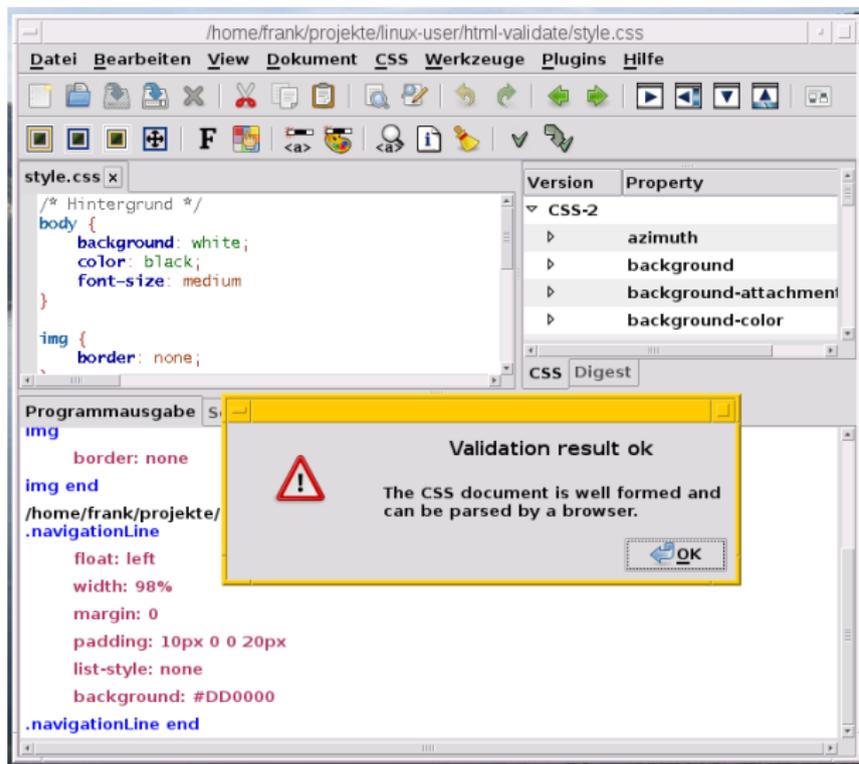
Two warnings

5 Expected an identifier and instead saw '!',
6 Expected ')' and instead saw 'main'.

One undefined variable

6 variable

CSS validieren, optimieren und zusammenfassen



- csstidy (Kommandozeile)
- cssed (GTK)

Inhalte als nachladbar kennzeichnen

Attribute in HTML5:

- prefetch
Inhalte nachladen
- prerender
Inhalte nachladen und rendern

Darstellung:

```
<link rel="prefetch" href="meineurl">
```

```
<link rel="prerender" href="meineurl">
```

Vorraussetzung:

Option `network.prefetch-next` im Webbrowser einschalten

HTML kompakt formulieren

Werkzeuge:

- JTidy (Java-basiert)
- PTidy (Perl-basiert)
- TidyLib (Python-basiert)
- htmlclean

Aufruf:

```
htmlclean -v *.html
```

Inhalte mit Verfallsdatum ausliefern

- Module `mod_expires` für Apache und nginx
- Apache-Konfiguration:
`ExpiresByType image/jpg "access plus 30 days"`
`ExpiresByType image/png "access plus 1 month"`
- nginx-Konfiguration:
`location * .png${expires 30d;}`
- Zeitbegrenzung überprüfen (Expires-Eintrag im Header):
`curl -I http://webseite.de`

Erweiterungen für den Webserver

- statische und dynamische Inhalte cachen
Module `mod_cache`, `mod_disk_cache` (Apache 2.2) bzw. `mod_cache_disk` (ab Apache 2.4), `mod_file_cache`
- Inhalte komprimiert ausliefern
Modul `mod_deflate`
- Inhalte vor der Auslieferung bereinigen
Modul `mod_tidy`
- Einsatz eines Caching Proxy, bspw. Rabbit
- Wechsel auf HTTP/2

Literatur

- Gerold Rupprecht, Frank Hofmann:
Beschleunigt. Die Ladezeit von Webseiten verringern
Teil 1 – Endbenutzer
veröffentlicht in LinuxUser 06/2015
- Gerold Rupprecht, Frank Hofmann:
Beschleunigt. Die Ladezeit von Webseiten verringern
Teil 2 – Webseitenbetreuer
veröffentlicht in LinuxUser 07/2015
- Gerold Rupprecht, Frank Hofmann:
Beschleunigt. Die Ladezeit von Webseiten verringern
Teil 3 – Admin
veröffentlicht in LinuxUser 1/2016

Vielen Dank!

Lassen Sie es setzen.



Linux, Layout & Satz



Kontakt:

Dipl.-Inf. Frank Hofmann
Hofmann EDV – Linux, Layout und Satz
Email frank.hofmann@efho.de
web <http://www.efho.de>
c/o büro 2.0 Raum 108
Weigandufer 45 – 12059 Berlin