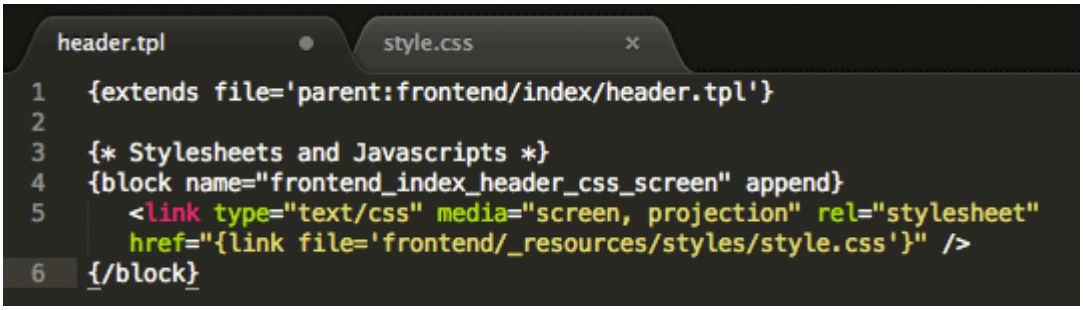
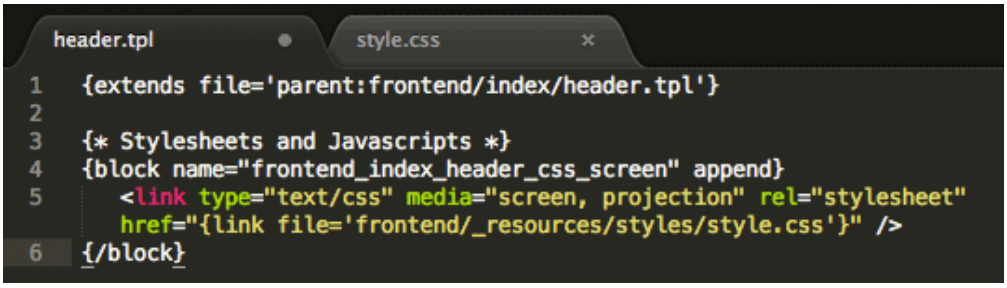
Gekaufte Shopware Templates installieren und individuell anpassen



Gekaufte Shopware Templates installieren und individuell anpassen

Mit den Shopware Templates im Community Store gibt es eine kostengünstige Alternative den eigenen Shop individueller zu gestalten. Besonders beliebt ist z.B. das Responsive Template von Conexco. Das Template nutze ich in einigen Shops als Grundlage für weitere indviduelle Designanpassungen. Vorteile

Veröffentlicht am 5. Januar 2015 von Marco



Mit den Shopware Templates im Community Store gibt es eine kostengünstige Alternative den eigenen Shop individueller zu gestalten. Besonders beliebt ist z.B. das Responsive Template von Conexco. Das Template nutze ich in einigen Shops als Grundlage für weitere indviduelle Designanpassungen. Vorteile sind z.B.:

- Optimierte Darstellung für mobile Endgeräte
- Kompatibel mit den Shopware Einkaufswelten

- Kompatibel mit einigen Drittanbieter-Plugins
- Kompatibel mit Shopware Premium Plugins (Live-Shopping, Bundle, Bonus-System, etc.)
- Ständige Weiterentwicklung und Optimierung

Gerade die Unterstützung der Einkaufswelten und Plugins ist keine Selbstverständlichkeit und sollte vor dem Kauf eines Templates überprüft werden. Ansonsten könnten zusätzliche Kosten entstehen, wenn ein Plugin nicht funktioniert oder die Darstellung fehlerhaft ist. Die meisten Plugins werden nämlich nur für das Standard-Template von Shopware entwickelt, da der Entwickler das verwendete Template ja nicht kennt.

[expand title="mehr lesen…"]

Installation und Aktivierung von gekauften Templates

Gekaufte Templates werden über den Shopware Community Store geladen und über den Plugin-Manager installiert. Die Aktivierung hängt vom verwendeten Template ab. Entweder muss das Template im Plugin aktiviert und konfiguriert werden oder es erscheint ein neuer Menüpunkt unter Einstellungen. Infos dazu findet man meist in der Dokumentation vom Template.

Nachdem ein Template installiert und konfiguriert wurde, sollte der Shop-Cache gelöscht werden, damit keine zwischengespeicherten Ansichten angezeigt werden.

Individuelle Anpassungen beim gekauften Template

Die original Template-Dateien sollten nicht verändert werden, da die Anpassungen beim nächsten Update garantiert überschrieben werden.

Um ein Template updatesicher zu verändern, erstellt man in dem Order „templates/_local“ eigene Template-Dateien und überschreibt oder erweitert so das Design des aktiven Templates.

Beispiel 1 – Eigene CSS-Datei integrieren

Schritt 1:
Datei „header.tpl“ im Order „templates/_local/frontend/index/“ anlegen. Die Ordner „frontend“ und „index“ müssen auch angelegt werden, sofern nicht vorhanden.

Schritt 2:
Folgenden Quellcode in die Datei „header.tpl“ einfügen. Dadurch binden wir die CSS-Datei „style.css“ in das Template ein.

```
[php]{extends file='parent:frontend/index/header.tpl'}

{* Stylesheets and Javascripts *}
{block name="frontend_index_header_css_screen" append}
<link type="text/css" media="screen, projection"
rel="stylesheet"
href="{link file='frontend/_resources/styles/style.css'}" />
{/block}[/php]
```

Schritt 3:
Datei „style.css“ im Order „templates/_local/frontend/_resources/styles“ anlegen. Die Ordner „_resources“ und „styles“ müssen auch angelegt werden, sofern nicht vorhanden.

Schritt 4:
Eigene CSS-Styles in der Datei „style.css“ einfügen.

Schritt 5:
Shop-Cache leeren, damit die Änderungen sichtbar werden.

[/expand]

Wie aktualisiere ich PHP für meine WordPress-Website?



Wie aktualisiere ich PHP für meine WordPress-Website?

In dieser Anleitung zeigen wir Ihnen, wie Sie PHP sicher auf die neueste Version aktualisieren können, ohne Störungen auf Ihrer WordPress-Website auszulösen. Schritt 1 – Überprüfen Sie Ihre aktuel…
In dieser Anleitung zeigen wir Ihnen, wie Sie PHP sicher auf die neueste Version aktualisieren können, ohne Störungen auf Ihrer WordPress-Website auszulösen.Schritt 1 – Überprüfen Sie Ihre aktuelle PHP-VersionSchritt 2 – Aktualisieren Sie WordPress auf die neueste VersionSchritt 3 – Installieren Sie das Plugin „PHP Compatibility Checker“Schritt 4 – Führen Sie einen Scan durch und beheben mögliche ProblemeSchritt 5 – Aktualisieren Sie PHP auf die neueste VersionSchritt 6 – Stellen Sie sicher, dass Ihre Website intakt ist

> ***Hinweis:*** *Eine PHP-Version hat in der Regel einen Lebenszyklus von drei Jahren. Danach sollte sie nicht mehr verwendet werden. Argumente für die Aktualisierung finden Sie in unserem Leitfaden: Warum muss PHP aktualisiert werden?*

[expand title="mehr lesen…"]

Schritt 1 – Überprüfen Sie Ihre aktuelle PHP-Version

Zuerst müssen Sie überprüfen, welche PHP-Version Sie gerade verwenden. Bitte warten Sie mit der Aktualisierung bis zum Schritt 5 in dieser Anleitung.

1. Melden Sie sich im One.com-Kontrollpanel an.
2. Klicken Sie auf **PHP und Datenbank Einstellungen**, zu finden unter der Kachel **Erweiterte Einstellungen**.
3. Scrollen Sie nach unten, um die **PHP-Version** zu **aktualisieren** .
4. Überprüfen Sie, welche Version Sie gerade verwenden.

Setzen Sie PHP in der Version 7.2 oder höher ein, besteht kein Problem. Nutzen Sie hingegen PHP 7.1, sollten Sie die Aktualisierung veranlassen. Bitte fahren Sie mit Schritt 2 fort.

Schritt 2 – Aktualisieren Sie WordPress auf die neueste Version

Stellen Sie sicher, dass WordPress selbst sowie alle Themes und Plugins auf die jeweils neueste Version aktualisiert wurden.

1. Loggen Sie sich in Ihren **WordPress-Adminbereich** ein.
2. Klicken Sie auf **Dashboard** > **Aktualisierungen** .
3. Stellen Sie sicher, dass Sie die neueste Version von WordPress installiert haben und dass auch alle Plugins und Themes auf dem neuesten Stand sind. Aktualisieren

Sie diese, falls erforderlich.

> ***Tipp:*** *Fragen Sie sich, weswegen es wichtig ist, WordPress zu aktualisieren? Lesen Sie unseren Leitfaden: [Warum sollten Sie WordPress immer aktuell halten sollten]*



Schritt 3 – Installieren Sie das Plugin „PHP Compatibility Checker“

> ***Hinweis:*** *Das Plugin [PHP Compatibility Checker] wird von einem Drittanbieter bereitgestellt, nicht von One.com. Es kann Ihnen dabei helfen, zu überprüfen, ob Ihre Website mit der neuesten PHP-Version kompatibel ist. Wenn Sie jedoch Probleme oder Fragen zur Verwendung haben, müssen wir Sie an das Plugin-Entwicklerteam verweisen. Unser Kundensupport kann Ihnen dabei leider nicht helfen.*

1. Gehen Sie zu **Plugins** in Ihrem WordPress-Adminbereich.
2. Klicken Sie im Infokasten zum Plugin oben rechts auf **Jetzt installieren**.
3. Suchen Sie im Suchfeld rechts nach **PHP Compatibility Checker** .
4. **Installieren** Sie das Plugin; es wird sofort aktiviert.



Schritt 4 – Führen Sie einen Scan aus und beheben mögliche Probleme

> ***Hinweis:*** *Wenn Sie das [One.com Performance Cache-Plugin]*

verwenden , müssen Sie es vorübergehend deaktivieren, während Sie den Scan ausführen. Die Deaktivierung ist im WordPress-Adminbereich unter dem Menüpunkt „Plugins“ möglich.

1. Wechseln Sie im WordPress-Adminbereich über den Menüpunkt **Werkzeuge** zum Bereich **PHP Compatibility**.
2. Wählen Sie **PHP 7.3** aus, aktivieren Sie das Kontrollkästchen bei **Alle Plug-ins und Templates durchsuchen** und klicken Sie auf **Website durchsuchen** .
3. Warten Sie, bis der Scan abgeschlossen ist.
4. Sie können drei Ergebnisse haben:
   – **Kompatibel** = alles bestens!
   – **Warnung** = sollte funktionieren, könnte aber bei der nächsten PHP-Version Probleme machen.
   – **Fehler** = Obacht, könnte nach der Umstellung Probleme verursachen.
5. Bereinigen Sie alle Plugins oder Themes, die Fehler enthalten, indem Sie es entweder auf die neueste Version aktualisieren (falls verfügbar) oder sich ein kompatibles Alternativplugin mit derselben Funktionalität suchen.

***Tipp:** Wir empfehlen, nur Plugins zu verwenden, die regelmäßig aktualisiert werden und mit der neuesten WordPress-Version kompatibel sind. Ferner empfiehlt es sich, Plugins, die Sie nicht weiter benötigen, vollständig zu löschen.*

Schritt 5 – Aktualisieren Sie PHP auf die neueste Version

Nun sind Sie bereit für die PHP-Aktualisierung. Zugleich

empfehlen wir Ihnen, PHP-Fehlermeldungen zu aktivieren. Wenn ein Problem mit dem Code auftritt, erhalten Sie innerhalb der Fehlermeldungen Informationen zur Ursache und der genauen Stelle im Code, die den Fehler enthält.

1. Rufen Sie wieder den Bereich **PHP und Datenbank** im One.com-Kontrollpanel auf.
2. Scrollen Sie nach unten zu **PHP-Fehlermeldungen** .
3. Setzen Sie die Fehlermeldungen auf **Ein** und klicken Sie auf **Aktualisieren** .
4. Direkt darunter ändern Sie die Version bitte auf **7.3** und klicken Sie auf **Update** .

> ***Tipp:*** *Wir empfehlen, die Option* ***Neueste stabile PHP-Version*** *auszuwählen. Dann stuft One.com Ihren Webspace fortlaufend auf die neueste PHP-Version hoch. Siehe auch: [Was ist die neueste stabile PHP-Version?]*

Schritt 6 – Stellen Sie sicher, dass Ihre Website intakt ist

Nachdem die PHP-Version umgestellt wurde **dauert es mindestens 20 Minuten**, bevor die Änderungen wirksam werden. Wenn Sie eine Website mit konstant vielen Besuchern haben, kann es ggf. mehrere Stunden dauern. Aus diesem Grund empfehlen wir Ihnen, Ihre Website in den darauffolgenden 24 Stunden immer wieder stichprobenartig zu überprüfen.

Wenn Ihre Website nicht wie erwartet funktioniert, liegt es wahrscheinlich an einem Plugin oder Theme. Um herauszufinden, wo das Problem liegt, gehen Sie wie folgt vor:

1. Wechseln Sie vorübergehend zu einem Standard-WordPress-

Theme wie „Twenty Seventeen“.
2. Deaktivieren Sie alle installierten Plugins.
3. Aktivieren Sie nacheinander alle Plugins und Themes. Überprüfen Sie jedes Mal, ob Ihre Website noch intakt ist.

Wenn Sie nicht auf den WordPress-Adminbereich zugreifen können, können Sie Plugins und Themes von der Datenbank aus deaktivieren. Folgende Anleitungen unterstützen Sie dabei:Deaktivieren Sie WordPress-Plugins in phpMyAdminÄndern Sie Ihr WordPress-Theme in der Datenbank

Gerne können Sie kontaktieren, um weitere Informationen zu erhalten. Unser Support kann Sie bei der Suche der Fehlerquelle unterstützen. Bedenken Sie jedoch, dass wir den Code nicht für Sie reparieren können. Dies liegt in Ihrer Verantwortung.

Verwandte Artikel: Warum muss ich PHP aktualisieren?Was sollte ich beim Upgrade von PHP beachten?

[/expand]

---

shopware 6 storefront

shopware 6 storefront

All you need to know about the new Shopware 6 storefront

Taking new paths means adjusting to new opportunities! It means critically examining long-standing approaches, thinking outside the box…

[expand title="mehr lesen…"]

All you need to know about the new Shopware 6 storefront

Martin SchindlerJan 20, 2020·14 min read

Taking new paths means adjusting to new opportunities! It means critically examining long-standing approaches, thinking outside the box and trying out new things. The experience gained from previous versions, in particular, allows developing a software solution in a sensible way to meet the changing requirements of an already fast-moving market. Otherwise, a system risks turning into a dusty monolith.

Shopware 6 has a lot of thought put into it, leaving much room for new things. Since Shopware is not only a software solution for users but also a sort of framework for developers, the requirements and needs of the developer community are also an important topic for further development.

While the lion's share is under the surface, as is so often the case in the software business, and is usually hidden from the majority of people's eyes, e-commerce is also about the design, the look & feel, the performance, etc. – a good reason to break new ground with Shopware 6 also in the frontend!

Shopware 6 is based entirely on the tried-and-tested PHP

framework called Symfony. Considering the above, it makes perfect sense that the Twig template engine, also developed by SensioLabs, has found its way into software, replacing the previously used Smarty. But standardisation doesn't stop here – on the contrary!

A Bootstrap CSS framework instead of in-house development, Sass instead of LESS, object-oriented JavaScript (ES6), Webpack… the list is long and so are the resulting advantages and changes.

Keep reading to find out what changes you can expect and what is important in the Shopware 6 storefront.


Martin Schindler, BSc in Computer Science (Software Architect at dasistweb GmbH)

The components of the Shopware 6 storefront

In order to understand the bigger picture, you often have to take a look at the individual components and how they interact first. For this reason, we are going to examine the individual building blocks of the new storefront, starting with the three elementary layers of a web-based application and the new tools and standards used. I will try to make references to the various counterparts from Shopware 5, to help you better understand why some things suddenly run differently than before.


The three typical layers of a web application

Structure layer (structure & content)

The foundation of a web frontend is the **structure layer** – in its simplest form consisting of static HTML markup, divided into logical or content-related sections. In order to meet increased technical requirements, “template engines” are used that take over the rendering of HTML markup with the help of templates. This is nothing new for anyone who has already worked with an earlier version of Shopware.

What is new, however, is that from now on Twig will be deployed as the template engine, whereas Smarty was used in previous versions including Shopware 5. Also provided from the makers of Symfony, it is a purely logical conclusion for reasons of integration and interoperability in the Symfony cosmos alone. Without going too much into the details, Twig is considered one of the fastest, safest and most flexible of the established PHP-based template engines on the market.

And again: new (or better “different”) technologies come with an initial hurdle – the syntax in Twig, for example differs from the well-known Smarty syntax. Useful information on how to use Twig, its features, extensibility and coding standards can be found in the official documentation.

The most important aspects of working with Twig in the storefront

Debugging – A simple var_dump on an object in the view could bring Smarty to its knees. With Twig, things are completely different – fortunately! Thanks to the integrated VarDumper component and Symfony’s DebugBundle, all variables in the storefront can now be output smoothly:



An additional priority was placed on the clarity of the debug output. Thanks to visual preparation, the confusing “print_r” output is now a thing of the past:


Scalar data types, arrays and objects can easily be debugged using {{ dump() }}

Text blocks – Text blocks have also been modified. While a specially developed snippet function was used and each text block was identified by a combination of name and namespace in the Smarty context, things now work slightly different in Twig:


Using the “trans” filter, which comes with Symfony and Twig by default, text blocks can be placed anywhere in the template and consumed from the underlying data source (e.g. messages.de.yaml for German text blocks, messages.en.yaml for English translations).

Extensions & include – Twig provides great technical solutions to ensure extensibility. The ability to extend the scope of functions through custom tags, filters, functions and tests by using a purely object-oriented approach and the possibility to perform unit tests on these is an attractive tool for implementing individual requirements, especially for projects in the enterprise sector.

By the way, this tool is also used by the new storefront. Since plugins and individual themes form a multi-level inheritance hierarchy that is not supported by Twig by default, the developers have implemented two Shopware-specific TokenParsers that are used in the storefront with the “sw_include” and “sw_extends” tags. These two should be used instead of the default “include” and “extends” tags, thus ensuring that template blocks can be overwritten, e.g. by plugins.

For example, if you want to have one view inherited from

another, make sure to use “sw_extends”:



The same applies if you want to include a template file in a view. The “sw_include” tag should also be used in this case:



Icons & thumbnails – Of course, implementing little helpers to make daily working life easier for template developers is always a good idea. That is why “sw_icon” was created, allowing you to conveniently configure and load SVG icons:



Using “sw_thumbnails”, which renders a <img> tag with correctly configured “srcset” and “sizes” attributes based on the provided parameters, is just as convenient:



Of course, the organisation of the Twig files in the storefront and their structure has somewhat changed compared to Shopware 5.
However, there are no great technical intricacies here that would have to be explained in greater detail. So much for the structure layer for now.

Presentation layer

In today’s world, an online shop without a modern, user-centric look & feel has become unimaginable. The **presentation layer** allows us to bind the look of an application to HTML markup using CSS.

The rule of thumb here is: the more general the CSS definitions are, the more detached they are from the actual markup. The more specific they are, the faster changes in markup can lead to unwanted, visible side effects.

Since nowadays CSS is much more than just applying styles to

selectors, CSS preprocessors have been in use for many years. These allow you to use syntactic rules and language constructs to make CSS creation more efficient.

To be more specific, a custom style sheet language extends the limited functionality of CSS through variables, functions, mixins and more. Since the browser can only process CSS, the generated files are pre-compiled by the preprocessor into valid CSS. This compilation is then loaded in HTML markup and interpreted by the browser.

Switching the preprocessor: from LESS to Sass

Anyone who has worked with Shopware 5 already knows the LESS style sheet language. With Shopware 6, a sensible switch to Sass has been made. When comparing frameworks, languages or tools, as usually, the devil is in the detail – the most famous CSS preprocessors (Sass, LESS, Stylus and PostCSS) differ about 20% from each other in terms of their functionality, and the remaining 80% are congruent. That is why, to claim that Sass is better, faster and cooler than LESS would be a purely subjective statement.

However, one criterion that speaks for the switch to Sass, and one that is not so easy to dismiss, is its widespread use. The diagram below shows the prevalence of various CSS preprocessors as determined by a survey. Sass clearly stands out from the competition:

The Front-End Tooling Survey 2018 (Ashley Nolan)
Another argument for switching from LESS to Sass is the CSS framework. After comprehensive evaluation, the decision was made to use the established Bootstrap frontend CSS framework for Shopware 6 due to its similar market leadership compared to the alternatives available. With version 4 of the

framework, source files are now written as Sass files (more precisely, using the SCSS syntax style for “Sassy CSS”, i.e. with the *.scss file extension). This is another aspect that strongly supports the switch to Sass.

Bootstrap 4: Taking advantage of source files

As mentioned above, the Bootstrap source files are available as Sass files. That’s why you can use the variables, mixins and functions defined in them for your own purposes. Information on which these are and how they can be used can be found in the official documentation.

Well, this is nothing really new here… after all, the already defined LESS constructs could be reused in Shopware 5. So, what exactly is the specific advantage for the new storefront?

Documentation of a de-facto standard – Instead of in-house implementations, the use of Bootstrap means that we relying, in a sense, on an already existing standard. Functions, mixins, etc. have already been tested thousands of times by the community alone, are regularly improved and extended, and are documented in detail. This is a circumstance that allows for rapid, targeted progress when making adjustments to the storefront.

Reduction to the essentials – Since many issues and problems (e.g. grid system, various components, browser-specific peculiarities, etc.) can already be considered addressed and solved thanks to the Bootstrap framework, the focus in the implementation of the new storefront is on what is really essential. By using the source files and also explicitly those components that are actually required in the storefront, the compilation, i.e. the CSS file created at the end, can also be reduced to the essentials. This, in turn, improves performance

and the loading time of the storefront.

Keeping your SCSS DRY – DRY (Don’t Repeat Yourself) is a well-known paradigm of software engineering. It means that redundant code should be avoided or at least reduced.

Thanks to variables, mixins and functions as well as the ability to inherit from existing selectors (using the @extend command), definitions and constructs can be easily reused. Since Bootstrap brings along a large portfolio, new buttons, labels, notifications, grids, etc. can be created quickly and easily. This is a great advantage, especially if you need to quickly implement the rough structure of a view, but also, of course, when it comes to fine-tuning.

Extensibility – Thanks to the widespread use of Bootstrap, also in other web applications and systems, numerous third-party modules can already be consumed today and used for your own purposes in the storefront. Of course, certain necessary adjustments can never be avoided entirely. After all, the frontend is naturally very specific, or it depends on the respective project scope. However, in order to achieve success or even technical breakthroughs quickly, it is an essential advantage of using the Bootstrap 4 CSS framework in the new Shopware 6 storefront.

Corporate identity – Anyone who has already worked with Bootstrap knows that there are many variables that invite the user to create custom configurations. This way, you to make key adjustments in a central location, in order to adapt the entire look & feel to your needs. Thanks to the “!default” keyword variables within the Bootstrap source files, this is a pretty handy thing, especially with regard to the “skins” in the new storefront. The included “Shopware Skin” is a collection of SCSS files that is layered above the Bootstrap style sheet definitions, decorating the default theme of the new storefront in the Shopware CI. If it is not needed in a specific project, this skin can easily be removed or replaced

by a custom skin.

Organisation is half the battle

In order to structure the organisation of folders and files – an issue that is omnipresent in a software project – the new storefront basically follows the 7–1 pattern, an organisational pattern that has established itself in the Sass environment over time. Here, various responsibilities are logically categorised in seven separate sections (or folders).

The sample organisational structure of SCSS files below illustrates this categorisation of responsibilities:



This clear division not only allows for a clearer organisation of styles, it also reminds developers to be more conscious when making adjustments and to assign new files to the appropriate subject area in a more consistent manner.

> ***Advice***
> *Components that are coupled rather loosely to other files can be easily migrated to other themes or even projects.*

Behavioural layer

Last but not least, the structure and presentation layers are covered with the **behavioural layer**. This level offers everything users need for interactive operation. In the case of a web application, this is primarily done by the JavaScript (JS) scripting language.

In recent years, JS has aroused create interest in the programming world because the standardised ECMAScript language core, which JavaScript is based on, allows object-oriented code to be generated in a syntax similar to other class-based

programming languages. This functionality is available as of version 6. And since the OOP-like syntax results in the use of principles found in software architecture (clean code principles, inheritance, composition, separation of concerns, etc.), the quality awareness in this area has also clearly changed for the better.

It quickly became clear that we will rely on ES6 and a class-based syntax for the development of the new storefront. This also led to the creation of various tools – i.e. helper classes – that take on a wide range of tasks in the new storefront. I would like to briefly highlight the most important ones below:

ViewportDetection – Whereas the "StateManager" was responsible for all sorts of things in Shopware 5, the Shopware 6 storefront is now equipped with a dedicated ViewportDetection class. In conjunction with CSS viewports provided by Bootstrap, this class allows you to react to the change of the viewport via specially created events:



In addition, there are separate events for each viewport and methods to restore the current viewport.
It should be noted that this class deals exclusively with this issue (separation of concerns).

DeviceDetection – The DeviceDetection class, in turn, has the task of making Boolean expressions about the device used with the help of small, static functions. For example, if you want to find out whether the current device is a TouchDevice, you can do this easily as follows:



DomAccess – You want to quickly and reliably check whether a node element has an attribute? This is, among other things, the task of the DomAccess helper class. The class abstracts access to node elements, attributes and data attributes, and

ensures that reliable results are returned.

Another example illustrates the additional convenience that this helper class provides. If you want to implement a class that should only be executed if the corresponding selector can be found as a condition in HTML markup, this can be done as follows:

An exception will be thrown internally if the specified query selector doesn’t return a result. This can then be responded to with try/catch. As a third optional parameter, it is also possible to switch off the strict mode so that a Boolean FALSE is returned if an element is not found. This makes checks for “!== typeof undefined”, etc. a thing of the past. Checking for “!== FALSE” will always be sufficient.

HttpClient – Anyone who has already worked with asynchronous requests in JavaScript knows what matters: cancelling existing, still active requests before placing a new one, using the correct request method, specifying callbacks, ContentType, and even setting access credentials using the new Shopware 6 API. Since all of this has now been moved behind an easy-to-use interface, the HttpClient class provides the basis for “easy-to-use” HTTP request handling without much overhead, as shown in the following extract:

Numerous other little helpers and features in the new storefront ensure a clear, modern JS code base that lets the developer achieve desired results quickly and adapt the functionality to his project-specific requirements.

Webpack module bundler

Finally, I would like to briefly touch on something that is essential for the use of the new technologies mentioned above from various points of view. While Grunt was still used as a task runner, e.g., to compile CSS and JS code in Shopware 5, Shopware 6 now comes with the Webpack module builder. International giants such as Airbnb, Trivago, Adobe, Slack and others have been relying on Webpack for some time now, and the module bundler has slowly but surely become a “state of the art” technology.

Thanks to Webpack, we can convert the JS code written in ECMAScript 6 into cross-browser-compatible JavaScript using the Babel compiler, for example. Moreover, thanks to numerous plugins and loaders, Webpack can handle the compilation of SCSS to CSS, the creation of browser-specific CSS instructions (autoprefixing), the minimisation of the compilations, etc.

This also opens up other opportunities that are particularly interesting for the enterprise sector. Just to venture a quick look into the future, Webpack allows realising “entry points” – small, bundled packages, so to speak, that only contain the code that is actually required for each view. This reduces unwanted side effects during customisation, the file sizes of CSS and JS compilations as well as the load for the client, i.e. the browser, because only the JavaScript is running that used in the context of the respective view. Of course, this is not possible without making further modifications. How these are implemented and the effort involved, however, will be explained in another article.

The fact that you have different requirements for the system environment in the development stage of a project compared to production mode can also be taken into account in the storefront area thanks to the flexible Webpack configuration. The available configuration files are:

- webpack.base.config.js – contains the basic configuration that applies to all available environments
- webpack.dev.config.js – contains the configurations that apply to Webpack in “dev” or “watch” mode
- webpack.hot.config.js – configurations that explicitly apply to the built-in Webpack HMR mode
- webpack.prod.config.js – a configuration specifically customised for production mode

The basic configuration and the specific configuration are merged for each set environment (dev|watch|hot|prod). This results in the final configuration of Webpack, which ultimately generates JS and CSS files, along with the necessary resources (fonts, images, etc.), which are pulled into the build folder and made available.

Detailed documentation is available on the official Webpack website and can be consulted for making your own adjustments.

Conclusion

Much has happened since Shopware 5. Some things have changed fundamentally to move with the times – to ensure more quality, more distinction, more possibilities – also in the storefront. This inevitably requires a certain degree of adjustment. However, it is the only way to bring Shopware 6 and the projects to be realised with it to a new level. The market is evolving, and so are the requirements. And the new Shopware 6 storefront gives you and your project all the options you need to continue to meet future needs.

Originally published at https://www.shopware.com.

Written byMartin Schindler

Bachelor of Computer Science & Software Architect

with a weakness for perfection and the awareness of human imperfection. Passionate mountain biker ????

- Shopware
- Ecommerce
- E Commerce Software
- Ecommerce Web Development
- Symfony

More from Martin Schindler

Bachelor of Computer Science & Software Architect with a weakness for perfection and the awareness of human imperfection. Passionate mountain biker ????

Nov 20, 2019

Die neue Shopware 6 Storefront und was du alles darüber wissen solltest

There is also an english version of this post. Please check out:All you need to know about the new Shopware 6 storefrontTaking new paths means adjusting to new opportunities! It means critically examining long-standing approaches, thinking…medium.com

Mit Shopware 6 wurde viel umgedacht, viel Raum für Neues geschaffen. Weil Shopware nicht nur eine reine Software für Anwender ist, sondern ebenso als eine Art Framework für Entwickler fungiert, sind auch die Anforderungen und

Bedürfnisse aus der Developer Community ein wichtiges Thema bei der Weiterentwicklung.

Während sich der Löwenanteil wie so oft im Software-Business unter der Oberfläche befindet und sich zumeist den Blicken der Mehrheit entzieht, dreht sich im E-Commerce vieles aber auch um Optik, Look & Feel, Performance und Co. …Read more · 14 min read

---

Oct 29, 2019

Why database migrations indicate the importance of quality for a programmer



Photo by Crawford Jolly on Unsplash

Assume there is a new feature you are about to develop for your project or your customer's wish. There are plenty of different ways how to deploy your code, how to handle several versions or to keep quality by using code analysis tools, for example. Deployment pipelines (CI/CD) allow you to take your brand new feature live. And in case of emergency they will easily let you rollback to previous state. But what if your feature requires changes to your database's schema or data?

This is why "migrations" exist. Migrations are a concept or even more an additional functionality put on top of the database abstraction layer (DBAL) and object-relational mapping (ORM). It allows you to implement a some kinda "versioning" for your database schema and even your data itself. …Read more · 5 min read

Published in **The Startup**·Aug 14, 2019

Be less rude – Code quality is a matter of courtesy

This is the story of my good old friend, let's call him John. He is software developer with body and soul. And he's smart, of course he is! John is able to solve almost all the tasks assigned to him. But John also has a dark side… cleanliness is not his key strength. And I'm neither talking about he wouldn't follow the company's clean desk policy nor about his personal hygiene. Not at all! He is pretty messy about his work. Unfortunately! Because…

Not infrequently, unclean work leads to long-term errors. As always when talking about software, the biggest part lies beneath the water surface. No matter if it is complexity, the LOC (lines of code) itself or even the riskiest parts of its business logic. …Read more in The Startup · 9 min read

---

Published in **The Startup**·Jul 23, 2019

Being a better programmer than this morning – some aspects to focus on

As you might know there is a bunch of loosely typed programming languages like PHP, Perl or JavaScript. Even if there might be tons and tons of questions and answers to talk about regarding this topic, this article is not about strong

or weak typing, implicit or explicit type casting or anything like that.

This article is about non performers, average performers and top performers… about professionals and those who think they ‘d be… about programmers like you and me, no matter if you are novice or expert.

While being a web developer focusing on frontend development and my years as a software engineer for PHP based applications as well as an architect designing software and reviewing millions of lines of code I was in the happy position to gain lots of experience about how programmers develop. Some bloom over the years like a delicate plant. Others stand up, say “hi” and immediately brighten the room. It is so dependent on a variety of extrinsic and intrinsic influences, I could talk years about. …

[/expand]

---

wordpress two factor authentication

WordPress Two Factor Authentication



Two Factor Authentication

Sichert den WordPress-Login mit einer Zwei-Faktor-Authentifizierung – unterstützt WP, Woo + andere Login-Formulare, HOTP, TOTP (Google Authenticator, Authy etc.)

[expand title="mehr lesen…"]

[/expand]

ASUS Server Cage Kit HDD Cage, 90-S000H6390T

ASUS Server Cage Kit HDD Cage, 90-S000H6390T



ASUS Server Cage Kit HDD Cage, 90-S000H6390T

ASUS Server Cage Kit HDD Cage, 90-S000H6390T – Kostenloser Versand ab 29€. Jetzt bei Amazon.de bestellen!

[expand title="mehr lesen…"]

[/expand]

E-Mails sichern und archivieren

E-Mails sichern und archivieren

[expand title="mehr lesen…"]

Praxis Mails archivieren



Bild: Albert Hulm

Postlagernd

E-Mails sichern und archivieren

Posteingang und Ordner des E-Mail-Programms sind häufig ein wertvoller Datenschatz. Es ist keine schlechte Idee, ihn von Zeit zu Zeit zu sichern. Von Stefan Wischner

Das wichtigste und wertvollste Sammelbecken für Informationen ist für viele Nutzer nicht etwa das Verzeichnis mit den Arbeitsdokumenten, sondern die Postfächer und Ordner im Mailprogramm. Sie dienen häufig als Projekt- und Kontakthistorie, sogar als Dateisammlung in Form von ansonsten nicht abgelegten Anhängen.

Eigentlich müsste man sich um diese Datensammlung keine großen Sorgen machen: Zumindest bei per IMAP oder Exchange angebundenen Mailkonten liegt alles sicher auf dem Mailserver, meist bei einem großen Provider, wird dort regelmäßig gesichert und ist geschützt vor jedweden Problemen des eigenen Rechners. Der Hüter dieses Schatzes ist also der Mailprovider; das Tor dorthin das Mailprogramm. Genau das ist nicht nur beruhigend. Was ist zum Beispiel, wenn der Provider das Konto sperrt oder man ihn einfach wechseln will? Oder wenn das E-Mail-Programm gerade dann zickt, wenn man dringend eine Info aus einer älteren Mailnachricht braucht?

Beruhigend wäre in jedem Fall ein lokales Backup aller Mails und Ordner, am besten in einem Format, das möglichst jedes beliebige Mailprogramm lesen kann oder – noch besser – das notfalls auch ganz ohne Mailclient seine Inhalte preisgibt. Die gute Nachricht: Das geht mit fast jedem Mailprogramm, wenn auch nicht automatisch. Die schlechte: Es gibt keine einheitliche Methode, einzelne Nachrichten, Ordnerinhalte und komplette Ordnerstrukturen zu sichern und zudem unterschiedliche Dateiformate. Denen wollen wir uns zuerst widmen.

Formate

Die Formate, in denen man Mails und Ordner exportieren kann, teilen sich in zwei Gruppen: Die eine umfasst Formate wie PST, MBOX, EML und MSG. Diese Dateien lassen sich verlustfrei auch wieder in ein Mailprogramm importieren. Für die Software- und Plattform unabhängige „Einweg“-Archivierung bieten sich allgemeine Formate wie PDF oder HTML an.

OST, PST: Dabei handelt es sich um binäre, proprietäre Formate von Microsoft Outlook. OST wurde mit Outlook 2016 eingeführt und kommt nur bei IMAP- und Exchange-Konten zum Einsatz. OST-Dateien dienen als lokaler Offline-Cache für Nachrichten, die vom Mailserver heruntergeladen wurden. Outlook verwaltet OST-Dateien selbstständig und erlaubt weder deren Ex- noch Import;

als Datensicherung oder für die Migration taugen sie daher nicht.

Anders verhält es sich mit PST-Dateien. Diese dienten früheren Outlook-Versionen als lokaler Speicher für Nachrichten inklusive Dateianhängen, Terminen, Aufgaben und Kontakten, bis sie von den erwähnten OST-Dateien weitgehend abgelöst wurden. Nur für POP3-Postfächer nutzt Outlook weiterhin PST-Dateien – und für den Im- und Export. Outlook erlaubt es, Ordnerinhalte – auch verschachtelte Strukturen – als PST-Dateien zu exportieren und kann diese auch wieder einlesen. Das gilt unabhängig davon, ob es um Mails auf einem POP3-, IMAP oder Exchange-Konto geht. Damit eignen sich PST-Files sehr gut zur Datensicherung. Der Haken: Es handelt sich um ein Binärformat, das nur Outlook selbst, einige andere E-Mail-Clients (zum Beispiel Thunderbird mit dem Add-on ImportExportTools NG und eM Client[1]) und einige Spezialprogramme (etwa MailStore, dazu später mehr) lesen können.

MBOX: Ähnlich wie Outlooks PST-Format kann eine MBOX-Datei komplette Ordnerstrukturen nebst aller enthaltenen Nachrichten speichern. MBOX kommt ursprünglich aus der UNIX-Welt und unterscheidet sich von PST in einem wesentlichen Punkt: MBOX-Dateien liegen im Textformat vor und lassen sich auch ohne Mailprogramm mit einem beliebigen Editor lesen. Das ist aber anstrengend, denn die Nachrichtentexte verstecken sich zwischen Metadaten, HTML-Code und großen Textblöcken mit Zeichensalat. Letztere sind eingebettete Dateien, per Base64 in druckbare Zeichen kodiert. Nahezu alle E-Mail-Programme (Outlook ausgenommen) können MBOX-Dateien importieren und wieder in die ursprüngliche Nachrichtenform mit Anhängen bringen. Alternativ gibt es etliche MBOX-Viewer-Tools im Netz, wenngleich deren kostenlose Versionen meist eingeschränkt sind. Die Flexibilität und Verbreitung machen MBOX zum sinnvollsten Backup- und Archivformat für Mails.



```
68      T9YafQW4mGhlpKkDntCpCFLDTHZUavP8b9lJtWpFGCd0S0NCsltvMQJcTYa7ey47cPZ/kL2rLPd
69      aJkGF+od2nOWvHPToSwwcGKLTYEc=
70  X-TMASE-SNAP-Result: 1.821001.0001-0-1-12:0,22:0,33:0,34:0-0
71  X-TMASE-INERTIA: 0-0;;;;
72  X-Spam-Score: (-) -1.9
73  X-Sender: [redacted].net
74  X-Scan-Signature: 57630e95f7146bb794c33d98cfd5f6df
75
76  -------=_NextPart_000_2749_01D6C4D9.C9507950
77  Content-Type: text/plain;
78      charset="iso-8859-1"
79  Content-Transfer-Encoding: quoted-printable
80
81  Hallo Stefan
82
83  =20
84
85  Mails, die im EML- oder MBOX-Format gespeichert sind, lassen sich zwar mit
86  einem Editor lesen. Es ist aber nicht einfach, den Nachrichteninhalt
87  zwischen all den Header-Daten, dem HTML-Code und eingebetteten Dateien zu
88  finden. Besser ist es, die Dateien in einen E-Mail-Client zu laden.
89
90  =20
91
92  [redacted]
93
94  Alter Ego
95
96  =20
97
98  =20
99
100
101 -------=_NextPart_000_2749_01D6C4D9.C9507950
102 Content-Type: text/html;
103     charset="iso-8859-1"
104 Content-Transfer-Encoding: quoted-printable
```

MBOX-Dateien enthalten Nachrichten und Dateianhänge in reiner Textform und lassen sich mit jedem Editor lesen. Die relevanten Textinhalte der Mails zu finden, kann aber mühsam sein.

EML: Wie beim MBOX-Format stecken in EML-Dateien Mailnachrichten im Textformat nebst Headern, Formatangaben, HTML-Code und Base64-kodierten Anhängen. Allerdings enthält jede EML-Datei immer nur genau eine Nachricht. Genutzt wird das Format zum Beispiel von Thunderbird. Interessanterweise kannte auch das längst eingestellte Outlook-Express EML-Dateien. Die zu Microsoft Office gehörenden Outlook-Versionen unterstützen das Format jedoch nicht.

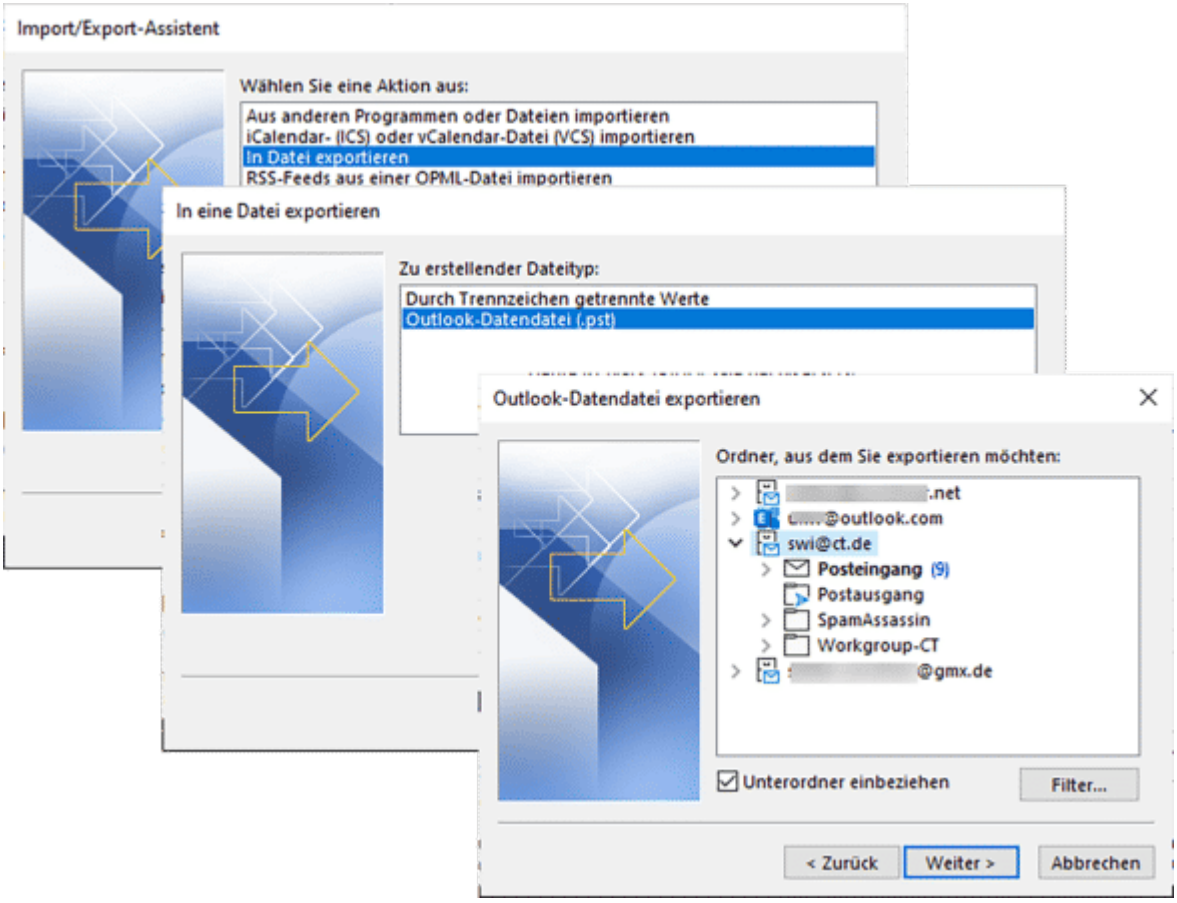
MSG: Das Outlook-Gegenstück zu EML sind MSG-Dateien, die zum Beispiel dann entstehen, wenn man aus Outlook eine Nachricht per Drag & Drop in einen Ordner oder auf den Desktop zieht oder „Datei/Speichern unter“ verwendet. MSG ist zwar wie PST ein Binärformat, das sich nur in Outlook öffnen lässt. Lädt man eine solche Datei in einem Texteditor, finden sich aber

Text- und Headerinhalte gut lesbar zwischen Blöcken von Sonderzeichen.

Die Konvertierung von EML- in MSG-Dateien und umgekehrt erlaubt zum Beispiel das für Einzeldateien kostenlose Tool E-Mail-Converter von IN MEDIA (siehe ct.de/yjr5).

PDF: An sich gäben PDF-Dateien ein hervorragendes Archivformat für E-Mails ab, sind sie doch plattformunabhängig und von vielen Programmen lesbar. Zwar exportiert nur Thunderbird mit passendem Add-on (dazu gleich mehr) Ordnerinhalte im PDF-Format, man kann sich aber bei allen anderen mit einem PDF-Drucker behelfen. Windows 10 bringt sogar schon einen mit (Microsoft Print to PDF). Die Methode hat aber einige Haken: So bleiben nicht nur extern nachgeladene Inhalte (Bilder) auf der Strecke, sondern auch angehängte Dateien. Komplette Ordnerstrukturen kann man nicht per PDF-Drucker exportieren, sondern maximal alle Nachrichten eines einzelnen Ordners. Je nach Mail-Client entsteht dabei entweder ein einziges großes PDF-Dokument (etwa bei Outlook) oder Sie werden bei der Ausgabe für jede einzelne Nachricht nach einem Dateinamen gefragt. Die Sicherung von E-Mails als PDF ist eine Einbahnstraße für Archivzwecke, ein Rückimport in ein E-Mail-Programm ist nicht möglich.

Welche Möglichkeiten und Formate es zum Export und Import von Nachrichten, Ordnern und kompletten Ordnerstrukturen eines Kontos gibt, hängt ganz vom verwendeten E-Mail-Client ab [1].



Microsoft Outlook erlaubt den verlustfreien Ex- und Import von Ordnern und Nachrichten nur in seinem eigenen PST-Format. Nur bei POP3-Konten nutzt es das auch als lokalen Datenspeicher.

Microsoft Outlook

Intern nutzt Outlook das PST-Format für POP3-Konten. Um die Ordner eines POP3-Kontos zu sichern, reicht es daher, Kopien der zugehörigen PST-Dateien anzulegen. Deren Speicherort finden Sie in Outlook unter „Datei/Kontoeinstellungen/Kontoeinstellungen …“ im Karteireiter „Datendateien“. Deutlich komfortabler und auch für IMAP- und Exchange-Konten geeignet ist die Export-Funktion von Outlook, die auch eine Auswahl der zu exportierenden Ordner erlaubt: Wählen Sie „Datei/Öffnen und Exportieren/Importieren/Exportieren“ und im folgenden Dialog „In Datei exportieren“. Nach einem Klick auf „Weiter“ wählen Sie „Outlook-Datendatei (.pst)“ und im nächsten Fenster die Daten, die Sie exportieren möchten. Sie können ein komplettes Konto oder einzelne Ordner mit oder ohne Unterordnern

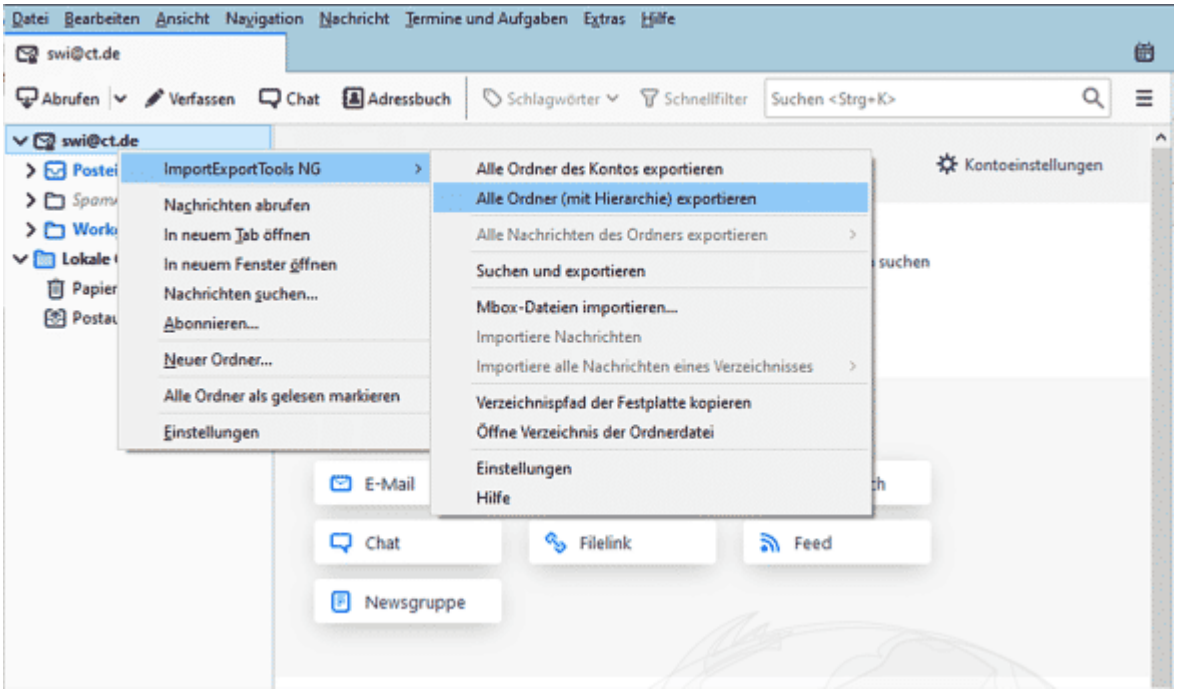
auswählen. Über den „Filter …“-Button lassen sich die Export-Daten weiter begrenzen, etwa durch einen Datumsbereich. Zuletzt legen Sie noch den Zielspeicherort und Dateinamen fest und starten den Export mit „Fertig stellen“.

Die erzeugte PST-Datei lässt sich später wieder in Outlook importieren. Einige andere Programme unterstützen das Format ebenfalls.

Mozilla Thunderbird

In der Grundausstattung fehlt Mozilla Thunderbird auch in der aktuellen Version 78 die Möglichkeit, komplette Ordnerstrukturen zu exportieren. Lediglich einzelne (oder gesammelt etwa mit Strg+A markierte) Nachrichten lassen sich im EML-Format speichern. Dazu klicken Sie die entsprechende Nachricht (oder eine der markierten) mit der rechten Maustaste an und wählen aus dem Kontextmenü „Speichern unter“. Nach der Auswahl eines Zielverzeichnisses landen alle selektierten Nachrichten als einzelne EML-Dateien mit der jeweiligen Betreffzeile als Dateiname darin.

Um komplette Ordnerstrukturen und -inhalte zu exportieren, benötigen Sie das Add-on „ImportExportTools NG“. Das fügen Sie am besten aus Thunderbird heraus über „Extras/Add-ons“ hinzu. Danach steht im „Extras“-Menü sowie in den Kontextmenüs von Konten und Ordnern in der Navigationsliste ein neuer Befehl „ImportExportTools NG“ mit diversen Untermenüs zur Verfügung. Darüber können Sie MBOX-Dateien im- und exportieren und über „Einstellungen“ vieles anpassen, wie zum Beispiel die Namensvergabe oder automatische Backups. Einzelne Ordner lassen sich über den Menüpunkt „Alle Nachrichten des Ordners exportieren“ nicht nur im MBOX-Format, sondern auch als separate PDF-, HTML- oder Textdateien speichern, optional mit Anhängen. Die landen in separaten Unterverzeichnissen mit Links (HTML) oder mit Pfad- und Dateinamen (TXT) in den jeweiligen Nachrichten. Beim PDF-Export hingegen gehen sie verloren.



Thunderbird erlaubt den Export von Ordnerstrukturen und Mailkonten nur mit dem Add-on „ImportExportTools NG“, unterstützt dann aber sehr viele, teils auch ohne Mailprogramm nutzbare Formate.

eM Client

Der an sich funktionsreiche eM Client erlaubt zwar den Import von MBOX- und PST-Dateien, nicht aber deren Export. Es lassen sich jedoch einzelne oder als Gruppe markierte Nachrichten über „Menü/Datei/Exportieren …“ im EML-Format speichern. Das klappt auch mit kompletten Mailkonten und allen enthaltenen Ordnern, die man beim Export lediglich einzeln auswählen muss. Im Zielverzeichnis entstehen dann entsprechende Unterordner mit den EML-Dateien. Es gibt zudem auch eine integrierte Backup-Funktion für komplette Konten, die sichert aber nur im vom eM CLient auch für die interne Datenhaltung genutzten Datenbankformat. Wer eine einzelne MBOX-Datei archivieren möchte, muss das mit einem anderen Client tun oder alternativ mit Mailstore Home (siehe unten).

Webmailer

Statt eines dedizierten Mailprogramms nutzen viele einen Webmailer, also das Browser-Frontend des jeweiligen Providers.

Nicht alle davon bieten die Möglichkeit, Nachrichten und Ordner als Kopie auf den lokalen Rechner herunterzuladen. GMX beispielsweise erlaubt das nicht bei einem Freemail-Konto, wohl aber in Verbindung mit DE-Mail. Im Zweifel konsultieren Sie die Hilfsfunktion des Providers oder nutzen zumindest für den Datenexport doch einen Mail-Client oder Mailstore Home (siehe unten).

Zwei der gängigsten Dienste, Google Mail und Outlook.com, erlauben den Download, haben die entsprechende Funktion nur etwas versteckt: Für Gmail-Nutzer führt der Weg zur Seite takeout.google.com. Sie ermöglicht den Download vieler persönlicher Google-Daten, etwa Fotos, Android-Einstellungen und YouTube-Suchverläufe. Da Sie nur die Mailordner herunterladen möchten, klicken Sie am Anfang der Diensteliste auf „Auswahl aufheben“, scrollen dann bis zu „Gmail“ herunter und setzen nur dort das Häkchen. Am Ende der Liste klicken Sie auf „Nächster Schritt“, wählen dann „Einmal exportieren“ und das Archivformat (ZIP). Nach einem Klick auf „Export erstellen“ landet eine Mail in Ihrem Gmail-Postfach, die einen Downloadlink für die gepackte MBOX-Datei enthält.

Nutzer eines Microsoft-Mailkontos klicken im Webmailer auf das Zahnrad rechts oben, dann ganz unten auf „Alle Outlook-Einstellungen anzeigen“. Im folgenden Fenster wählen Sie in der ersten Menüspalte „Allgemeine Einstellungen“ und in der zweiten „Datenschutz und Daten“. Klicken Sie dann rechts auf „Daten exportieren“. Wie bei Google erhalten Sie dann einen Downloadlink für eine gepackte PST-Datei per Mail, was allerdings laut Microsoft bis zu vier Tage(!) dauern kann.

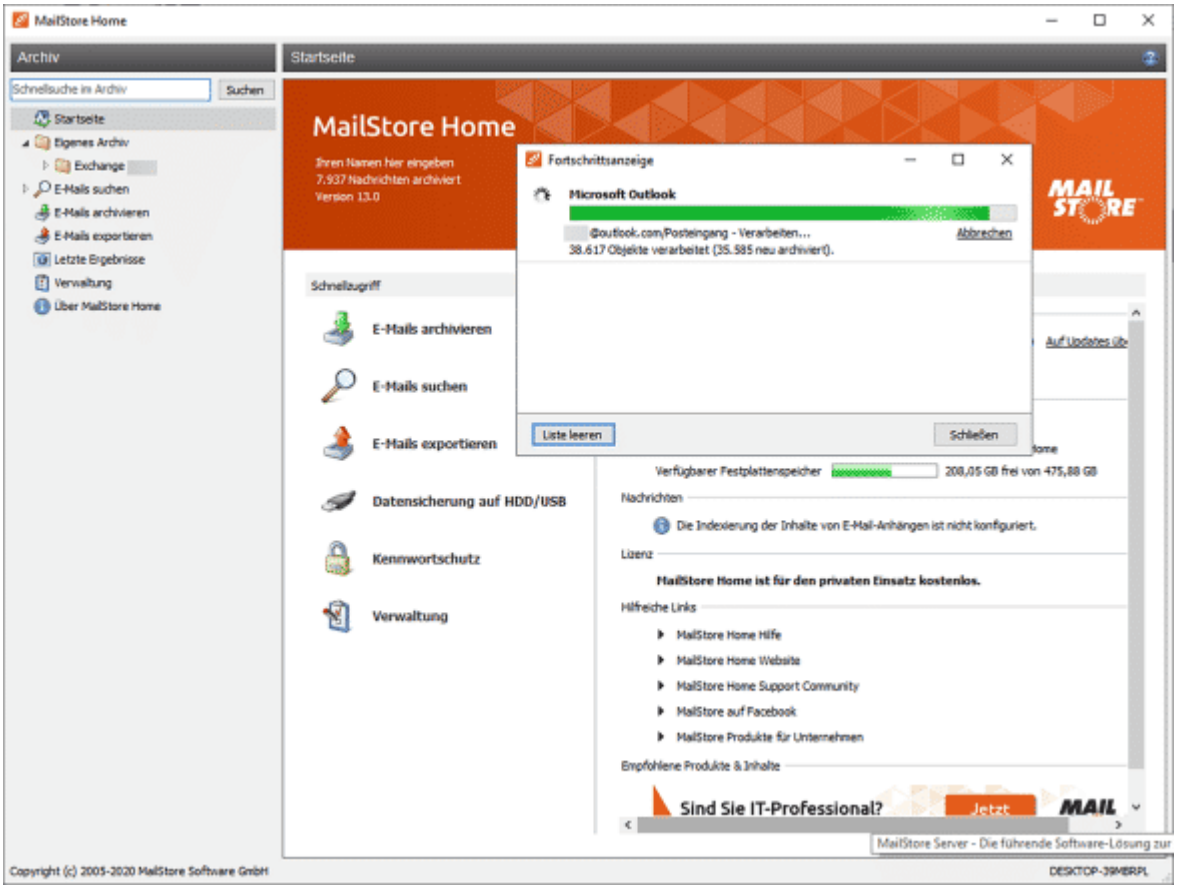
Apple Mail

Der Ex- und Import kompletter Accounts mit allen Ordnern im bordeigenen Mailclient von macOS ist sehr einfach: Klicken Sie in der Navigationsliste den entsprechenden Account mit der rechten Maustaste an, wählen Sie „Postfach exportieren“ und im nächsten Dialog einen Speicherort. Apple Mail speichert

grundsätzlich im MBOX-Format und kann solche Dateien über „Ablage/Postfächer importieren“ auch laden.

Komfortable Mail-Sicherung mit Mailstore Home

Abseits der beschriebenen Export- und Backupoptionen mit Bordmitteln der E-Mail-Clients gibt es zumindest für Windows-Nutzer eine elegante Lösung mit einem externen Programm aus deutscher Produktion: Das für die private Nutzung kostenlose Mailstore Home (siehe ct.de/yjr5) holt sich Nachrichten und Ordner inklusive Dateianhängen wahlweise aus lokalen Sicherungsdateien (PST, MBOX, EML, MSG), aus den Profilen von Thunderbird und Outlook und sogar – wenn Sie die zugehörigen Zugangsdaten eingeben – direkt vom Mailserver. Die importierten Inhalte werden in einer lokalen SQLite-Datenbank abgelegt und lassen sich mit dem Programm genauso ansehen wie in einem Mail-Client. Mailstore Home bietet zudem eine leistungsfähige Suchfunktion.



Das für die private Nutzung kostenlose Mailstore Home holt sich Mails und Ordner nebst Anhängen auf verschiedenen Wegen und speichert sie in einer eigenen durchsuchbaren Datenbank.
Es lassen sich beliebige Profile anlegen und so zum Beispiel mehrere Mailkonten in die Datenbank übertragen. Je nach Umfang der Postfächer kann das erstmalige Einlesen recht lange dauern, folgende Imports laufen deutlich schneller, weil das Programm nur noch die Änderungen abgleicht. Umgekehrt können Sie Mails aus MailStore Home exportieren, zum Beispiel direkt auf einen Mailserver, an einen Client oder in eine PST- oder einzelne MSG- oder EML-Dateien.

Tipp: Auch wenn Sie Mailstore Home regulär auf Ihren Rechner installiert haben, rufen Sie das Setup-Programm nochmals auf. Nehmen Sie das Angebot an, eine portable Version einzurichten; die legen Sie am besten auf einen externen Datenträger zusammen mit Sicherheitskopien der Mailstore-Datenbank.

Nicht GoBD-konform!

Ein wichtiger Hinweis zum Schluss: Die beschriebenen Methoden eignen sich durchwegs für die Sicherung von E-Mails und Ordnern, folgen aber nicht den Vorgaben der GoBD. Diese 2014 vom Bundesministerium für Finanzen herausgegebene und zuletzt Anfang 2020 überarbeitete Richtlinie beschreibt die ordnungsgemäße revisionssichere Archivierung von steuerrelevanten Unterlagen, zu denen auch geschäftliche E-Mails gehören können. Eine GoBD-konforme Archivierung bietet keiner der gängigen E-Mail-Clients; dazu sind entsprechende Dokumentenmanagement-Systeme, Online-Dienstanbieter oder passende Services von Mailprovidern erforderlich. Von Mailstore gibt es auch eine kostenpflichtige Server-Version, die die GoBD-konforme Archivierung von E-Mails verspricht. *(swi@ct.de)*

1. Literatur
2. Jo Bager, Holger Bleich, Sylvester Tremmel, Stefan Wischner, Solide Kuriere, 8 Mailprogramme für den

Desktop im Vergleich, c’t 25/2020, S. 72

Tools zur Mailsicherung: ct.de/yjr5

[/expand]

Performance-Probleme in Websites erkennen und beseitigen

Performance-Probleme in Websites erkennen und beseitigen

[expand title=“mehr lesen…“]

Performance-Probleme in Websites erkennen und beseitigen

Praxis Web-Performance



Bild: Rudolf A. Blaha

Ungebremst

Performance-Probleme in Websites erkennen und beseitigen

Surfer schätzen komplexe Apps, geschmeidige Animationen, Webfonts, Videos und hochauflösende Fotos. Viele Seiten laden, derart aufgemotzt, aber zu langsam. Lahme Websites wieder flott zu machen ist ein Mehrkampf mit vielen Disziplinen – ein Überblick. Von Herbert Braun

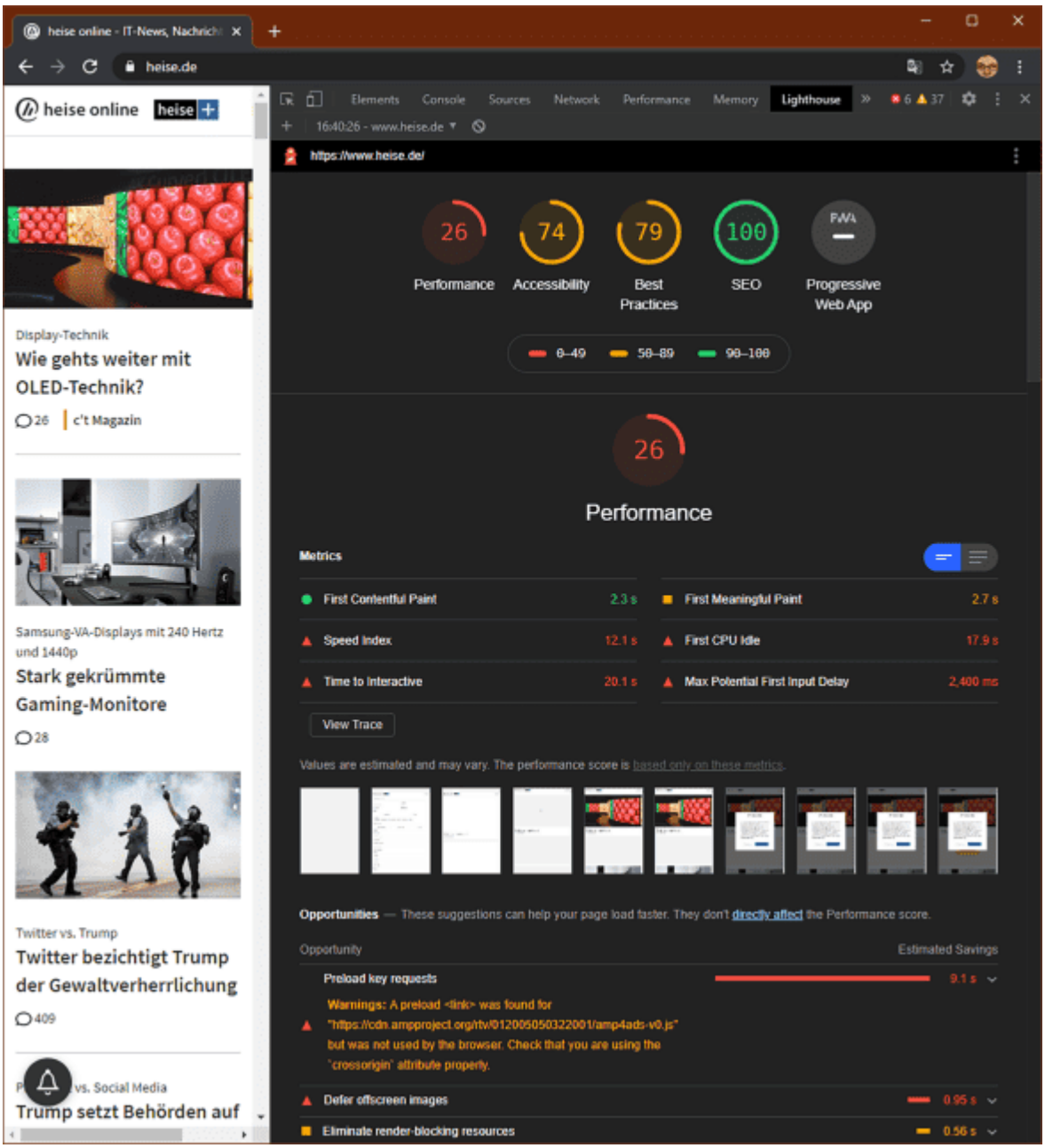
Eine durchschnittliche Webseite wiegt heute zwei MByte, die sich auf 75 HTTP-Requests verteilen (siehe ct.de/yp4b). Fast ein halbes MByte JavaScript-Code hat der Browser dabei zu verdauen. Gleichzeitig sind die Nutzer nicht mehr so geduldig wie zu ISDN-Zeiten: Drei Sekunden leerer Bildschirm sind für manchen Besucher schon zu viel. Eine Website, die nach zehn Sekunden noch nicht geliefert hat, wird den überwiegenden Teil ihrer Besucher verloren haben.

Es gibt viele sehr unterschiedliche Maßnahmen, die Sie als Website-Betreiber umsetzen können, um ihre Seiten flotter zu machen. Dieser Artikel beschreibt Optimierungen für das Frontend. Er gibt einen Überblick über das Spektrum der Möglichkeiten und geht nur vereinzelt in die Tiefe; die Umsetzung im Detail hängt ohnehin stark von den Anforderungen und Problemen der jeweiligen Website ab.

Level 0: Testwerkzeuge

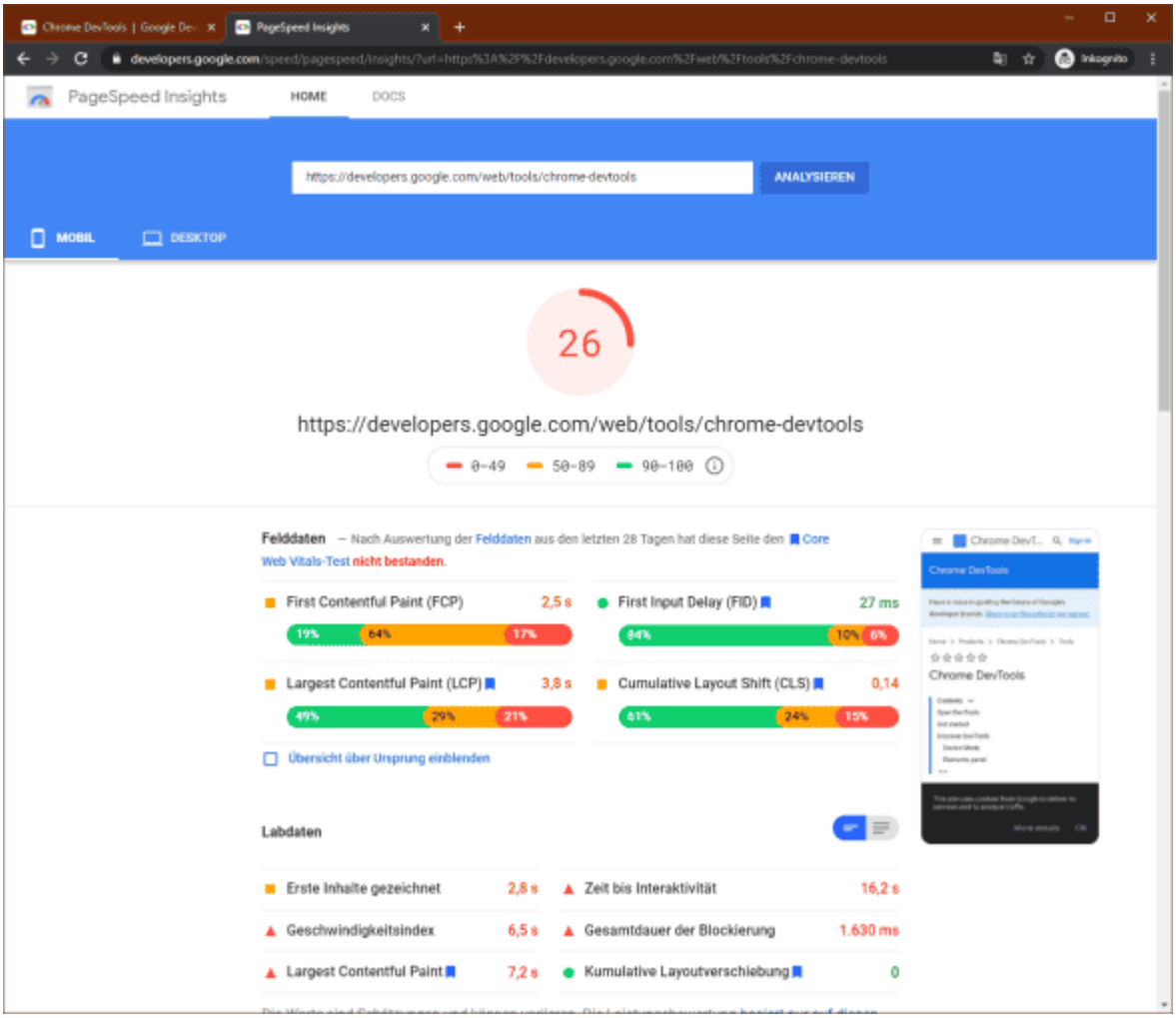
Zunächst gilt es herauszufinden, wo es klemmt. Heute benutzt man für Tests und Tipps meist Google PageSpeed Insights (PSI), Webpagetest.org oder Lighthouse. PSI ist vergleichsweise übersichtlich und eignet sich gut für Einsteiger. Das Open-Source-Projekt Webpagetest.org legt den Fokus mehr auf die Aufbereitung der Rohdaten als auf klare Handlungsanweisungen.

Lighthouse – ebenfalls Open Source – stammt wie PSI von Google, testet aber nicht nur die Performance einer Website, sondern etwa auch SEO und Barrierefreiheit; es steckt hinter den Analysefunktionen von PSI, wertet aber anders aus. Lighthouse ist kein Webdienst: Sie finden es in den Chrome-Entwicklerwerkzeugen, können es aber auch als Node.js-Anwendung installieren.



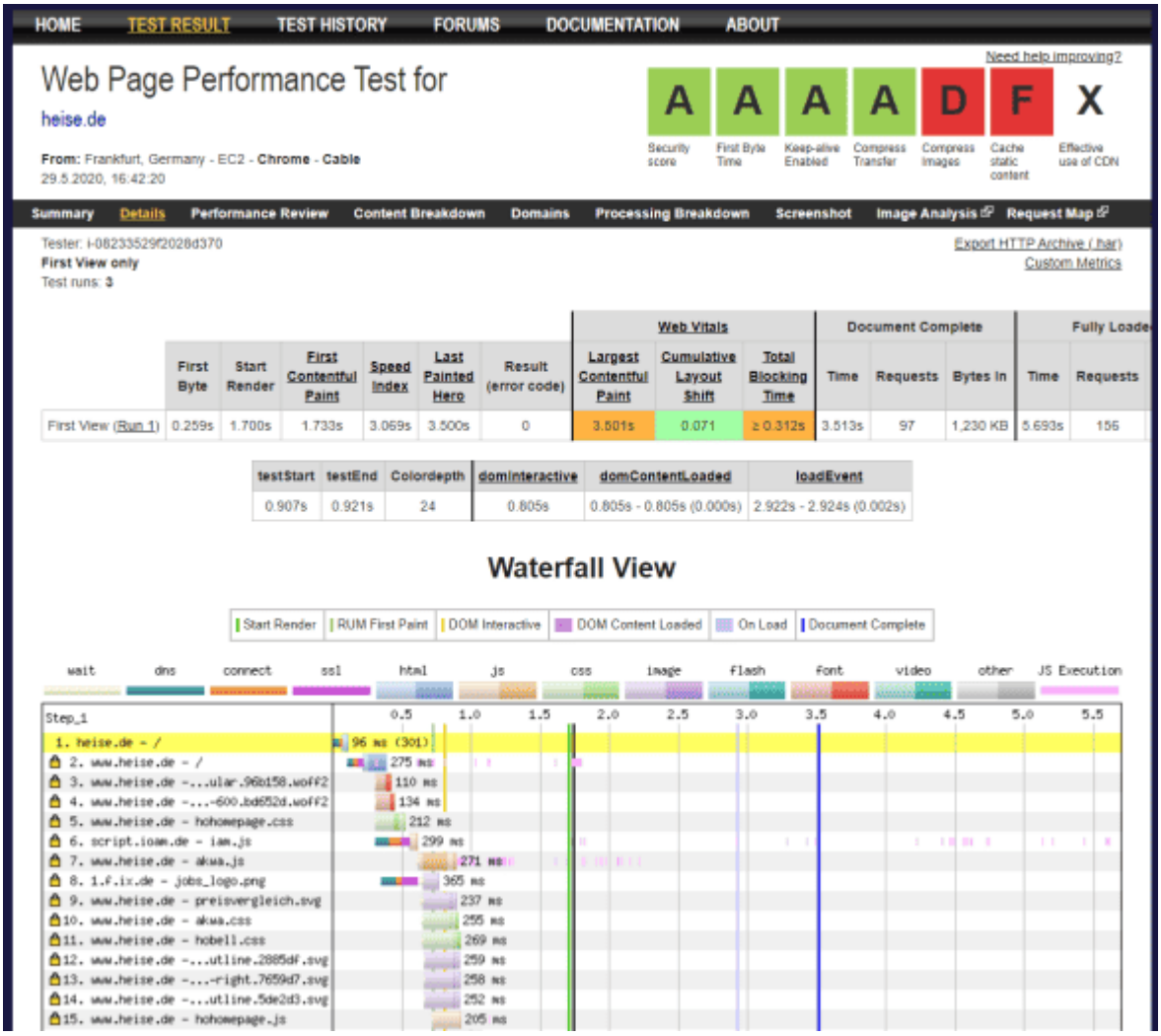
Lighthouse zeigt hübsch gestaltete Messergebnisse und wartet mit konkreten Verbesserungshinweisen auf.

Die Messergebnisse geben Anhaltspunkte, doch sollten Sie sie nicht überbewerten: Sie hängen oft von Zufällen ab und weichen zum Beispiel zwischen Lighthouse und PSI ab. Selbst bei Google-eigenen Seiten fällt der Geschwindigkeitsindex mitunter schlecht aus. Nützlicher sind die Ratschläge („Nicht genutztes JavaScript entfernen“, „Bilder richtig dimensionieren“ etc.), verbunden mit konkreten Angaben zu den betroffenen Dateien und Zeilen.



Googles PageSpeed Insights verwendet eine ähnliche Technik wie Lighthouse, kommt aber zu anderen Ergebnissen.

Unabhängig von Lighthouse finden Sie in den Entwicklerwerkzeugen der gängigen Browser Werkzeuge zum Messen von Netzwerkzugriffen, zur Rendering-Performance und zum Ressourcenverbrauch. Diese zeichnen nach der Aktivierung große Mengen an Daten auf, die Sie anschließend studieren können, um Performance-Engpässe auszumachen. Allerdings sind sie für Website-Tuning-Einsteiger kaum geeignet.



Webpagetest.org ist eine Alternative zu Googles Performance-Werkzeugen.

Level 1: Abspecken

Browser-Entwicklerwerkzeuge erlauben es, den Datendurchsatz zu drosseln, beispielsweise, um die Nutzung im Mobilfunk nachzustellen. Wer das einmal ausprobiert hat, wird sich mit mehr Engagement dem Entrümpeln und Komprimieren der Website widmen.

Das größte Einsparpotenzial haben meist die Bilder – keine andere Maßnahme wirkt so schnell wie deren Optimierung. Klar, dass ein Bild nicht größer sein sollte als das Maximum der Anzeigebreite. Was die Sache kompliziert macht, sind „Retina“-Displays, die Bilder höher auflösen können. Ein iPhone etwa stellt in der Standardskalierung jedes CSS-Pixel mit 2 × 2 Gerätepixeln dar; eine 500 × 300 Pixel große Bilddatei wird in

einem entsprechend großen CSS-Container okay aussehen, aber das Gerät könnte auf dieser Fläche auch 1000 × 600 Pixel unterbringen – ein Bild sieht so einfach schärfer aus.

Um solche Fälle und unterschiedliche Bildgrößen durch responsives Layout abzufangen, stehen Frontend-Entwicklern CSS-Media-Querys und insbesondere die HTML-Attribute srcset und sizes zur Verfügung. Der Browser ermittelt anhand dieser Angaben, welche Bilddatei am besten passt, und lädt nur diese herunter, zum Beispiel:

```
<img alt="Bild" srcset=
  "standard.jpg 1x, retina.jpg 2x">
```

Eine JPEG-Qualitätsstufe von mehr als 80 oder eine verlustfrei komprimierte PNG-Grafik sind im Web meist Bandbreitenverschwendung. Auch das Entfernen von Metadaten oder effizientere Komprimierung holen etliche KByte heraus. Umsetzen lässt sich so was mit üblicher Bildbearbeitungs- und Betrachtungs-Software oder mit Konsolen-Tools wie jpegtran, jpegoptim oder optipng. Diese Tools verarbeiten große Mengen an Bildern und lassen sich in die Build-Pipeline integrieren. Die folgende Anweisung schrumpft manche Fotos auf ein Zehntel ihrer Dateigröße (Achtung, überschreibt Quelldateien!):

```
jpegoptim -o -m75 --strip-all  --all-progressive *.jpg
```

Bei JPEGs empfiehlt sich das progressive Rendering, bei dem das Bild von Anfang an in voller Größe erscheint und während des Ladens immer detailgenauer wird – das fühlt sich für den Benutzer schneller an. Für Icons kommen heute Vektorgrafiken in Form von SVGs oder Iconfonts zum Einsatz. PNGs sind vor allem bei Transparenzen interessant. Das neue WebP-Format wiegt nur etwa 80 bis 90 Prozent einer gleichwertigen JPEG-Datei, aber Sie brauchen gegebenenfalls ein Fallback für Internet Explorer. Bislang nur in Chrome läuft AVIF, das seine Stärken bei hoher Kompressionsrate ausspielt und GIF-ähnliche Animationen erlaubt.

Für Videos setzen viele Websites auf externe Dienstleister, die beim Streamen die Wiedergabequalität an die Bandbreite anpassen. Wo aber ein <video> oder <audio> zum Einsatz kommt, das eine Mediendatei anfordert, kann die richtige Komprimierung Megabytes an Daten einsparen. Tools wie ffmpeg erledigen diesen Job zuverlässig. Leider gibt es keine Entsprechung zu srcset für gestreamte Medien.

Auch den Website-Code sollten Sie zusammenstauchen. Code-Minifizierungswerkzeuge gibt es für CSS und HTML, aber mehr holen Sie bei JavaScript heraus. Das bekannteste Tool dafür heißt „Uglify“ – sein Output ist für den Menschen kaum leserlich, doch der Maschine ist das egal.

Anstrengender, aber lohnender ist es, unnötigen Code komplett rauszuwerfen. JavaScript-Bibliotheken lassen den Code-Umfang enorm anwachsen. Daher sollte sich der Entwickler bei jedem Third-Party-Skript fragen: Brauche ich das wirklich? Muss ich moments.js einbinden, wenn ich einmal ein Datum umrechne? Lohnt sich das Karussell-Plug-in, benötige ich jQuery, weil $(...) so schön kurz ist?

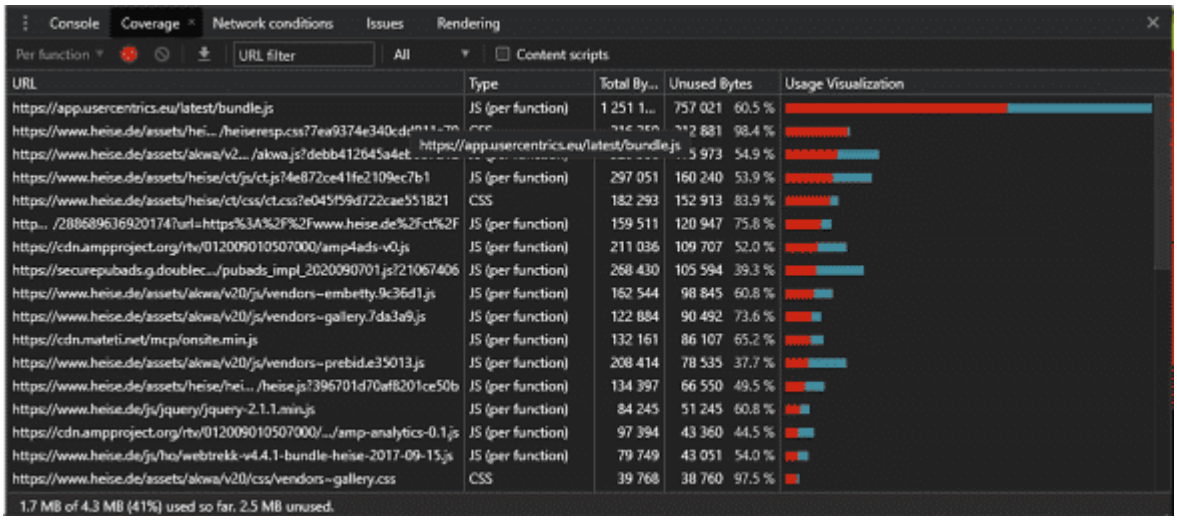
Nicht zu vergessen: Der Browser ist nach dem Download nicht fertig, sondern muss den Code auch noch verarbeiten. Während das etwa bei Bildern eine Frage von Millisekunden ist, leistet er bei JavaScript Schwerarbeit, die den Haupt-Thread oft sekundenlang blockiert. Auf einem leistungsschwachen Gerät kann das Kompilieren und Ausführen länger dauern als der Download. Tests mit realer Hardware bei unterschiedlicher Netzqualität fördern dabei mitunter Überraschendes zu Tage, sind aber aufwendig.

In gewachsenen Projekten findet sich oft erstaunliches Code-Gerümpel wie unterschiedliche jQuery-Versionen oder Polyfills, die seit Jahren keiner mehr braucht. Aber testen Sie die Seite gründlich und schmeißen Sie Code nicht vorschnell raus! Eine JavaScript-Exception stoppt nämlich die weitere Code-Ausführung, und wenn nichts mehr geht, nützt die schönste

Performance-Optimierung nichts mehr.

Chrome hat einen „Coverage“-Reiter (unter „More Tools“), der nicht benutzte CSS-Selektoren und JavaScript-Funktionen rot markiert. Bei den meisten Webseiten liegt deren Anteil bei weit über 50 Prozent. Das Node.js-Werkzeug UnCSS gibt das tatsächlich benutzte CSS aus – auch übergreifend für mehrere Seiten und Bildschirmgrößen.

Beim Import von Modulen ist es oft möglich, sich auf einzelne Komponenten zu beschränken. Moderne Bundler wie webpack oder Rollup beherrschen dieses „Tree-Shaking“ und kopieren mit `import {func1} from 'bigFile.js'` nur den zu func1 gehörenden Code ins Projekt statt der gesamten Skriptdatei.



Wie Chromes „Coverage“-Werkzeug zeigt, braucht man viele eingebundene Skripte und Stile nicht auf der aktuellen Seite.

Level 2: Ausliefern

Trotz Detailverbesserungen hat das Netzwerkprotokoll HTTP seine Wurzeln in den frühen 90er-Jahren, und TCP, auf dem es aufsetzt, ist noch älter. Beide erledigen den Job solide, aber ein bisschen umständlich – für heute übliche Szenarien mit oft mehr als hundert Requests pro Seitenaufruf waren sie jedenfalls nicht gedacht.

Der Overhead, den beide Protokolle verursachen, macht besonders die Übertragung kleiner Dateien teuer. Deshalb gilt

es als Performance-Optimierung, kleine Datenpäckchen zu größeren zusammenzufassen – etwa durch das Bündeln mehrerer Skript- und Stylesheet-Dateien („Bundling“) oder durch Tricks wie CSS-Sprites, bei denen man alle Icon-Grafiken in ein Bild stopft, um mithilfe von CSS das passende herauszufischen.

Außerdem beschränken Browser gemäß der HTTP-Spezifikation die Zahl der gleichzeitigen Verbindungen zu einem Host; typischerweise erlauben sie sechs gleichzeitige Downloads. Um das zu umgehen, setzen manche Websites „Domain-Sharding“ ein – die Aufteilung der Ressourcen auf mehrere Subdomains.

HTTP/2 macht solche Hacks überflüssig. Es benötigt nur eine TCP-Verbindung, um beliebig viele HTTP-Antworten zu liefern – auch solche, die der Client noch gar nicht angefragt hat (Server-Push). HTTP/2 ist inzwischen ein etablierter Standard, der laut W3Techs in 45 Prozent aller Websites zum Einsatz kommt [1].

Tatsächlich findet man das Protokoll bei internationalen Websites wie Google, Facebook, Amazon, eBay, LinkedIn beziehungsweise bei deren Content Delivery Networks (CDN), die diese Technik allesamt beherrschen. Auch manche Shared--Hosting-Angebote von der Stange liefern mit HTTP/2 aus, während andere Hoster den Umstieg bisher gescheut haben. - Wunderdinge sollte man von HTTP/2 allerdings nicht erwarten.

Der schnellste Download ist natürlich der, der nicht stattfindet. Geschicktes Caching kann wiederholte Seitenaufrufe enorm beschleunigen und sogar dafür sorgen, dass der Besucher etwas sieht, wenn er offline ist. Dafür setzt man die HTTP-Header Cache-Control oder Expires ein. Bei heise online zum Beispiel darf der Browser ein Bild für einen Monat im Cache behalten, während das Stylesheet nur zwei Stunden gültig bleibt; die Startseite muss er dagegen schon nach 30 Sekunden neu anfordern. Ist zusätzlich ein ETag-Header gesetzt, können Browser und Server abgleichen, ob sie beide die gleiche Dateiversion haben; in diesem Fall antwortet der

Server mit einem 304-Code, ohne Daten zu übertragen.

Einen Schritt weiter geht der Frontend-seitig programmierbare Cache, der mit Progressive Web Apps (PWA) möglich ist. Hauptzweck ist es, Websites auf Mobilgeräten offline verfügbar zu machen, aber Performance-Optimierung für den Desktop-Browser funktioniert damit ebenso gut. Doch egal, ob PWA oder Cache-Control: Übertreiben Sie nicht, sonst sieht der Besucher zu lange eine veraltete Version der Website!

Level 3: Vor- und Nachliefern

Wenn Sie die Größe des Downloads verringert haben, können Sie darüber nachdenken, wann Sie bestimmte Ressourcen benötigen. Das Standardverhalten – eine HTML-Datei saugt beim Laden sämtliche dazugehörigen Skripte, Stile und Bilder aus dem Netz – ist meistens nicht das schnellste: Manches fordert man besser vorher schon an, anderes erst später.

Aber was heißt eigentlich „Schnelligkeit“ bei einer Webseite? Man kann die Zeit messen, die vom ersten Request bis zum Eintreffen des letzten Bits vergeht, aber das ist nicht unbedingt die relevante Größe. Den Nutzer interessieren eher drei andere Ereignisse: dass irgendetwas auf dem Bildschirm erscheint, dass er im Browser-Viewport ein halbwegs fertiges Layout sieht und dass er mit dieser Ansicht interagieren kann.

Diese Ereignisse sind der „First Contentful Paint“ (FCP), der „Largest Content Paint“ (LCP) oder der „First Meaningful Paint“ (FMP) – sowie die „Time to Interactive“ (TTI).

Wenn also das Laden einer Seite fünf Sekunden dauert, sollte der Benutzer bis zu diesem Zeitpunkt nicht auf einen weißen Bildschirm starren müssen. Idealerweise sieht er innerhalb einer Sekunde relevante Inhalte, die sich anschließend nur noch wenig verändern, und kann die Seite bereits bedienen, während der Browser noch unterhalb des Fensterausschnitts liegende Bilder, Videos und Interaktionen nachlädt.

Meistens stellen Bilder den größten Datenanteil, und so hat sich Lazy Loading etabliert – der Browser fordert die Bilder erst an, wenn er Zeit hat oder sie benötigt. Moderne Browser (mit Ausnahme von Safari) brauchen dafür kein JavaScript mehr: Ein loading="lazy" im <img> genügt. Auch für IFrames funktioniert dies.

Problematisch sind vor allem die Inhalte, die das initiale Rendern blockieren: im Head eingebundene JavaScript- und Stylesheet-Dateien. Trifft der Browser auf solche Inhalte, stoppt er den Seitenaufbau, lädt die Datei herunter und parst sie beziehungsweise führt sie aus, bevor er das Rendern fortsetzt.

Die wenigsten Skripte müssen laufen, bevor die Seite gerendert wurde. Oft verschiebt man daher <script>-Elemente ans Ende des <body>. Den gleiche Effekt erzielen Sie, wenn Sie das <script>-Element im Head lassen und mit dem Attribut defer versehen – allerdings startet der Browser den Download früher, was meist wünschenswert ist. Wenn die Reihenfolge der Skripte egal ist, können Sie stattdessen mit dem Attribut async arbeiten.

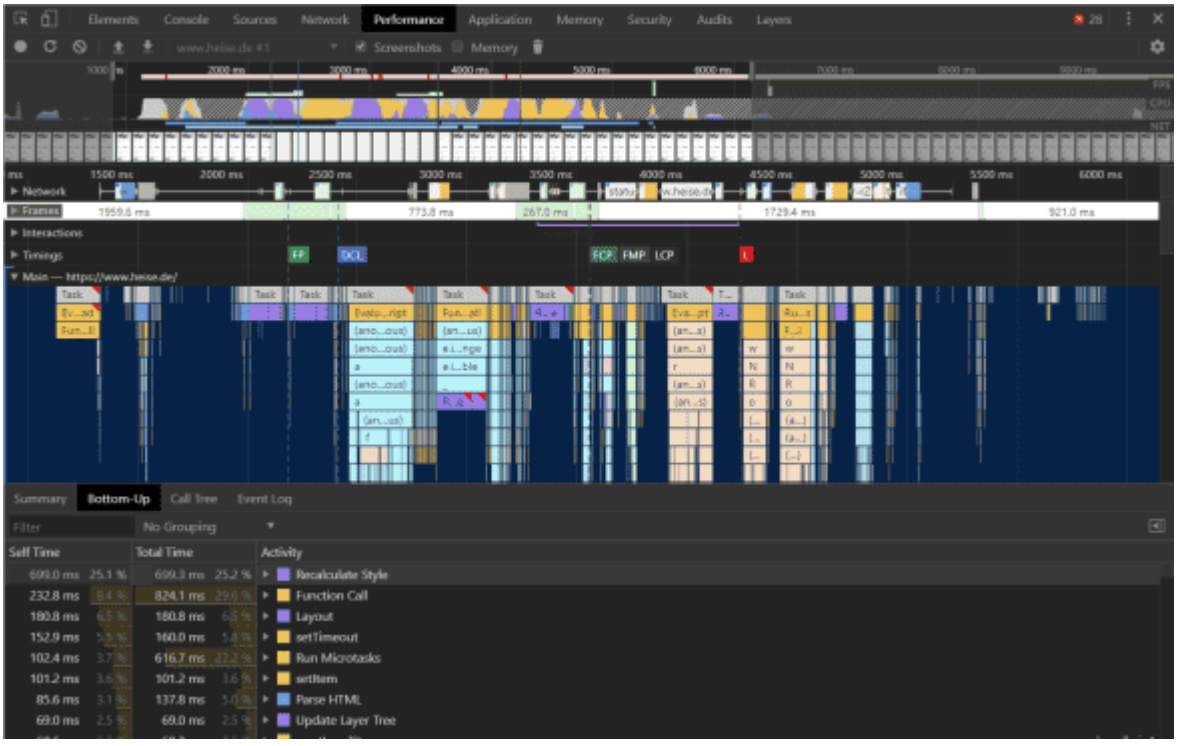
Weniger bekannt ist, dass auch Stylesheets nicht im <head>-Bereich stehen müssen. Sie können beispielsweise das CSS unterhalb des Browserfensters nachladen oder es komponentenweise aufteilen. Wenn Sie Stile für Media-Querys mit <link href="[URL]" rel="stylesheet" media="[Media-Query]"> anfordern, lädt der Browser sie nur herunter, falls er sie braucht. Das Tool Critical extrahiert die sofort benötigten Stile aus dem Stylesheet und fügt sie inline ins HTML-Dokument ein.

Dieses „Code-Splitting“ widerspricht der obigen Forderung nach möglichst großen Datenpaketen. Sie können diesen Widerspruch durch Abwägen und Messen auflösen – oder durch den Umstieg auf HTTP/2. Die technische Seite des Code-Splittings übernehmen gängige Bundler wie webpack, Rollup oder Parcel.js.

HTTP/2-Server-Push ist ein nettes Feature, aber die Frontend-seitigen Möglichkeiten sind flexibler und schicken nicht stumpf Daten durch die Leitung, die der Browser längst im Cache hat. In JavaScript laden Sie mit XMLHttpRequest oder fetch() Dateien, in HTML nutzen Sie das Tag <link href="[URL]" rel="[Typ]">, das besonders differenzierte Optionen bietet.

So spart der Typ dns-prefetch die Zeit für den DNS-Lookup, während preconnect zusätzlich TCP-Verbindung und Verschlüsselung erledigt. preload und prefetch laden eine Datei, allerdings für unterschiedliche Zwecke: prefetch hat niedrige Priorität und eignet sich für noch zu besuchende Seiten, preload dagegen – verpflichtend mit einem as-Attribut, zum Beispiel as="script" – lädt schneller und ist für die aktuelle Seite gedacht. prerender hat den Effekt, als würde man eine Seite im Hintergrund-Tab laden. Aber so mächtig diese Werkzeuge sind: Wie beim PWA-Cache ist Zurückhaltung gegenüber den Ressourcen des Nutzers angezeigt.

Level 4: Code-Feinschliff



Die Performance-Analysewerkzeuge der Browser machen Unmengen von Daten zugänglich, die sich aber erst nach längerer Beschäftigung mit dem Thema erschließen.

Das Laden ist der engste Flaschenhals im Web, deshalb kommt diesem Bereich bei der Performance-Optimierung ein besonderer Stellenwert zu – aber nach dem initialen Laden des HTML und der Render-blockenden Ressourcen muss der Browser binnen Sekundenbruchteilen ein paar Textdateien in Bildschirmpixel verwandeln. Diese Schwerstarbeit heißt „Critical Rendering Path“.

Dahinter verbergen sich mehrere Aufgaben. Der Browser wandelt HTML und CSS in Baumstrukturen (DOM und CSSOM) und führt beide im Rendering-Baum zusammen. Nun wühlt er sich durch alle DOM-Knoten und errechnet für jeden das Layout, also Größe und Position der Inhaltsboxen. In der Paint- oder Raster-Phase füllt der Browser diese Boxen mit Pixeln und ermittelt schließlich in der Compositing-Phase die Anordnung.

Eine offensichtliche Performance-Optimierung ist also, den Arbeitsaufwand überschaubar zu halten, indem man die Zahl der DOM-Knoten drosselt; Lighthouse meckert bei 1500 Elementen.

Der Umfang des CSS ist (abgesehen vom Laden) weniger problematisch, da sich dieses simple Format sehr effektiv verarbeiten lässt. Auch zusammengesetzte CSS-Selektoren (wie nav li:first-child a) ändern daran nichts: Zwar hält sich hartnäckig die Legende, dass diese die Performance beeinträchtigen, aber die Effekte bewegen sich knapp an der Messbarkeitsgrenze.

Eine spürbare Bremswirkung hingegen haben „Reflows“ in umfangreichen Dokumenten – so nennt man es, wenn bereits gerenderte Elemente erneut die Phasen Layout, Paint und Compositing durchlaufen müssen.

In der Praxis passiert dies oft durch nachgeladene Inhalte, zum Beispiel Bilder ohne vorher bekannte Dimensionen oder Webfonts, die nach dem initialen Rendern zur Verfügung stehen. Die dadurch ausgelösten Größenänderungen können eine ganze Kaskade von Reflows hinter sich herziehen. Auch CSS-

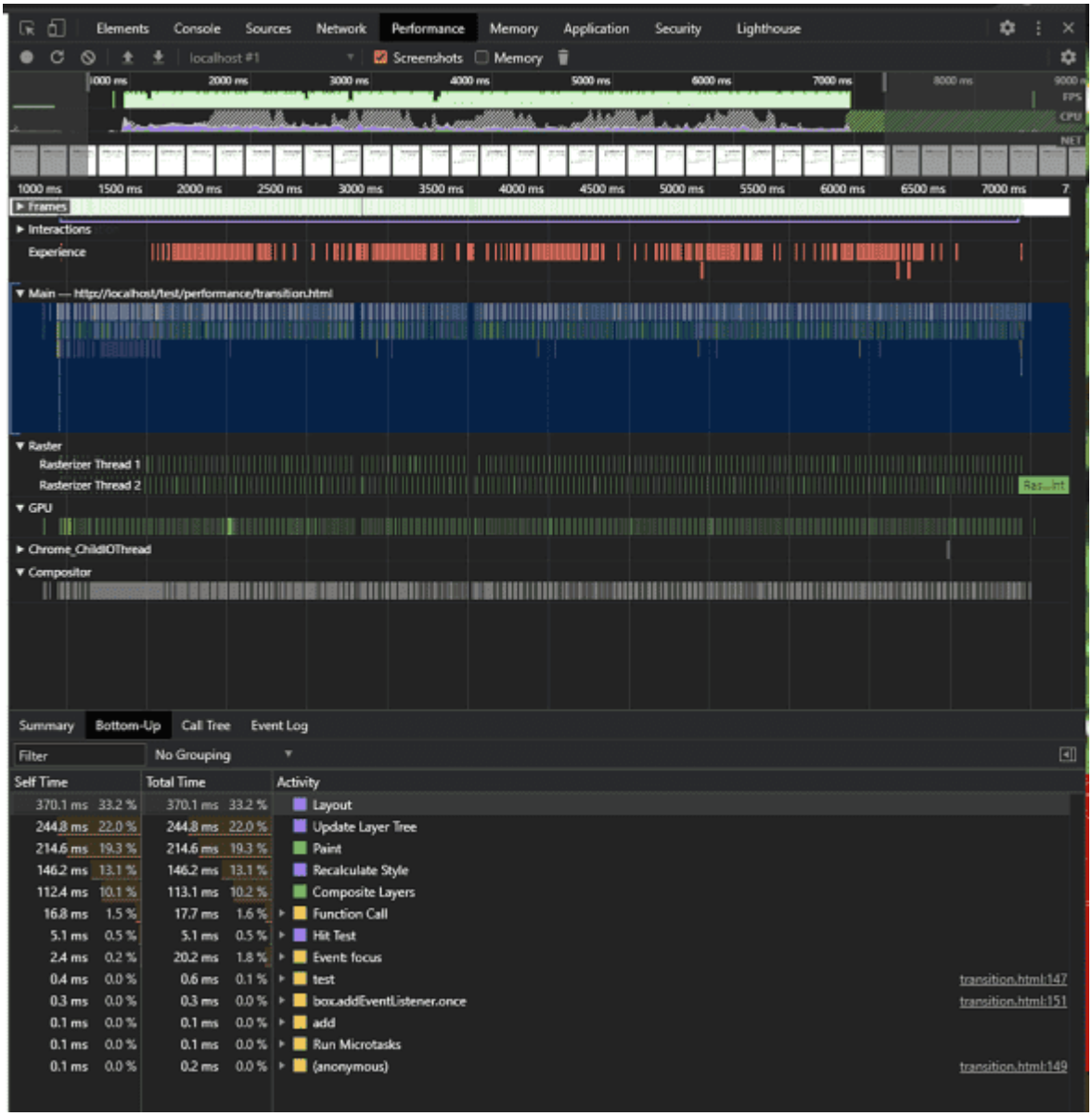
Animationen und -Übergänge sowie JavaScript-Aktionen können das verursachen.

Je nach Art der Änderung und Intelligenz des Browsers muss es nicht immer ein komplettes „Reflow“ sein. Um etwa ein Element animiert zu vergrößern und zu verschieben, bieten sich die CSS-Eigenschaften top, left, width und height an. Wo es möglich ist, sollten Sie dafür jedoch die transform-Eigenschaft mit den Funktionen translate() und scale() verwenden. Die meisten Browser überspringen dann Layout und Paint und gehen gleich zur Compositing-Phase über: Der Grafikprozessor manipuliert die Pixel des schon gerenderten Elements binnen Millisekunden.

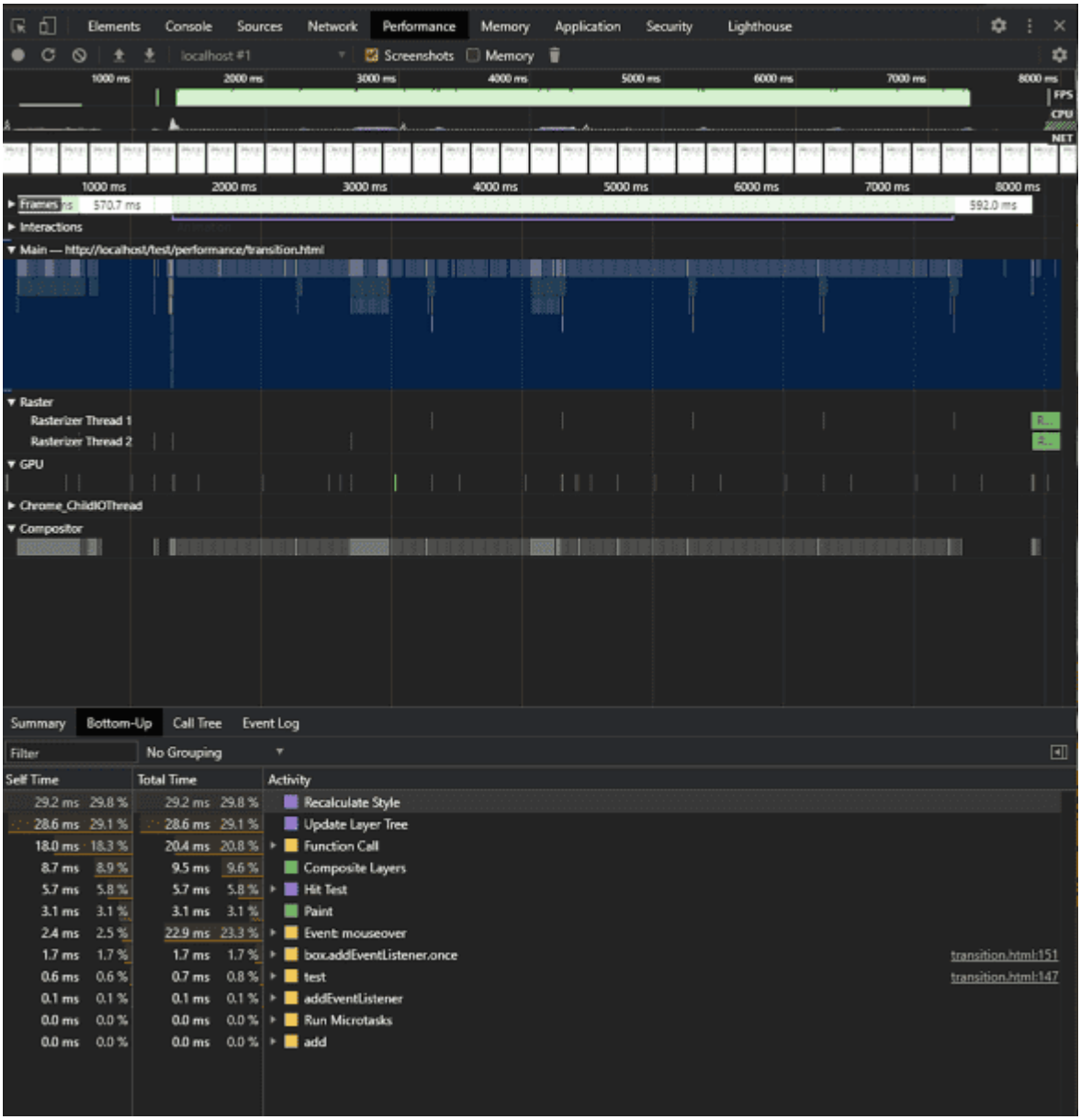
Selbst auf Mobilgeräten laufen derartige Animationen in der Regel flüssig. Wenn Sie Ihren Augen nicht trauen, können Sie die Framerate mit den Entwicklerwerkzeugen messen – in Chrome geht das mit der Kommandopalette (Strg+Umschalt+P) unter „Show frames per second meter“. Maximal erreichbar sind 60 fps.

JavaScript-Code läuft in einem einzelnen Thread. Daher kann eine langwierige Aktion den ganzen Browser zum Stehen bringen. Die Lösung für dieses Problem sind asynchrone Funktionen, was JavaScript in Form von Callbacks, Promises und async/await-Funktionen erlaubt.

Um die Vermeidung unnötiger Wartezeiten geht es auch bei passiven Event-Handlern, die für Scroll- und Touch-Events wichtig sind. Da das Scrolling in einem separaten Thread passiert, kann es auch während aufwendiger Berechnungen flüssig laufen – müsste der Browser nicht vorher prüfen, ob der Code nicht mit preventDefault() das Scrollen stoppt. Mit der Option {passive: true} in addEventListener() verspricht der Entwickler, genau das nicht zu tun.



Eine CSS-Transition, die angrenzenden Text zur Seite schiebt, zwingt den Browser zu harter Arbeit, …



… während ihn eine ähnliche Transition mit Transform-Eigenschaften keine Mühe kostet.
WebWorker können aufwendige Berechnungen in separate Threads auslagern. Das lässt sich gut mit WebAssembly kombinieren, eine auf Performance getrimmte Untermenge von JavaScript, die aus Sprachen wie C++ transpiliert wird. Und schließlich bringt WebGL die Grafikausgabe direkt auf die GPU.

Allerdings braucht man diesen Performance-Turbo außer für Spieleentwicklung nur selten, und für typische Webseiten-Aufgaben nützt er auch nicht viel – denn meistens sind Skripte auf einer Webseite mit DOM-Manipulationen beschäftigt, die mit WebWorkern, WebAssembly und WebGL nicht möglich sind.

Bei DOM-Zugriffen kann es erstaunliche Performance-

Unterschiede geben. Der wohl gängigste Weg, ins Dokument zu schreiben, ist, über die Eigenschaft innerHTML HTML-Quelltext einzufügen:

```
someData.forEach(data => {
  document.querySelector('.my-list').
  innerHTML += `<li>${data}</li>`;
});
```

Umständlicher sind die altmodischen DOM-Methoden wie document.createElement() und appendChild():

```
const list = document.
  querySelector('.my-list');
for (let i = 0;
       i < someData.length; i++) {
  const li = document.
    createElement('li');
  li.textNode = someData[i];
  list.appendChild(li);
}
```

Der Code cacht das Listenelement und hängt neue Elemente erst ins DOM ein, wenn Attribute und Inhalte vollständig sind. Die klassische for-Schleife ist minimal schneller als Array-Methoden wie forEach(). Während Letzteres jedoch wie auch das Caching nur minimale Verbesserungen bringt, beschleunigen die DOM-Methoden das Skript massiv – bei großen Listen bis um das Tausendfache.

Aber wie relevant ist das in der Praxis? „Voreilige Optimierung ist die Wurzel allen Übels“, schrieb Programmiergott Donald Knuth. Tatsächlich wird kaum ein Programmierprojekt am Performance-Unterschied zwischen for und forEach() leiden, während Wartbarkeit und Lesbarkeit vitale Bedeutung haben. Wenn Sie fünf Listenpunkte einfügen, ist es egal, welche Variante Sie wählen. Andererseits häufen sich viele kleine Performance-Sünden an, und das Bewusstsein für -effizienten Code kann entscheiden, ob eine Anwendung benutzbar ist oder nicht.

Gut studieren lässt sich dieser Effekt anhand bekannter Algorithmen, etwa für die Berechnung von Fibonacci-Zahlen – eine Reihe von Zahlen, die aus der Summe der vorherigen zwei gebildet werden (0, 1, 1, 2, 3, 5, 8, …). So könnte man die ersten n Fibonacci-Zahlen wie folgt berechnen:

```
const fib = n => n < 2?
  n : fib(n - 1) + fib(n - 2);
```

Der Algorithmus ruft sich rekursiv selbst auf, um bis zu den ersten Zahlen der Reihe zurückzugehen, die er dann addiert. Simpel, elegant – und extrem ineffizient; irgendwo bei n = 60 wird sich der Browser verabschieden. Der Rechenaufwand steigt mit jeder Iteration exponentiell an, während schlauere Algorithmen das Ergebnis in Sekundenbruchteilen liefern.

Rekursionen und verschachtelte Schleifen können die stärksten CPUs in die Knie zwingen. Wer öfter an komplexen Skripten arbeitet, sollte sich mit der O-Notation („Big O“) vertraut machen, die den Blick für solche Performance-Fallen schärft.

Fazit

Heutige Webanwendungen neigen zum Übergewicht. Aus Frameworks und Bibliotheken kommen Megabytes an oft ungenutztem Code, native Webtechniken wie Buttons, Eingabefelder oder Scrolling werden mit JavaScript nachgebaut. Kann man alles machen, solange die Seite schnell lädt und ruckelfrei läuft – nicht nur auf dem gut ausgerüsteten Entwickler-Laptop, sondern auch auf dem drei Jahre alten Billig-Handy.

Oft sind die naheliegenden Maßnahmen besonders effektiv, aber wer mehr rausholen will, muss tiefer einsteigen – und stößt dabei auf immer mehr Feinheiten beim Laden, Kompilieren und Rendern. JavaScript ist Fluch und Segen zugleich: Es trägt häufig zu Performance-Problemen bei. Funktionen wie Lazy Loading oder Service Worker können aber auch für flüssigeres Surfen sorgen. (*jo@ct.de*)

1. Literatur
2. Jan Mahn, Web-Beschleunigung, Das neue Webprotokoll HTTP/2 in der Praxis, c’t 20/2018, S. 162

Weiterführende Informationen: ct.de/yp4b

[/expand]

Fehler bei Hostern gefährden die Sicherheit von DKIM

[expand title=“mehr lesen…“]

Fehler bei Hostern gefährden die Sicherheit von DKIM

Wissen Konfigurationsfehler bei DKIM



Bild: Thorsten Hübner

DKIM-Fail

Fehler bei Hostern gefährden die Sicherheit von DKIM

Online-Kriminelle versenden regelmäßig E-Mails unter falschem Namen, um Nutzer zur Herausgabe von sensiblen Daten zu bewegen. Mit DKIM sind Spam-Filter in der Lage, solche gefälschten Mails zu erkennen. Doch unsere Analysen zeigen, dass einige Webhoster mit Fehlkonfigurationen Spammern und Phishern Tür und Tor öffnen. Von Leo Dessani und Jan Mahn

Eine neue E-Mail vom Chef. Laut Mailprogramm stammt sie auch von seiner Adresse. Offenbar steckt er im Ausland in Schwierigkeiten, hat seine Kreditkarte verloren und braucht

schnell etwas Geld vom Firmenkonto. Was auf den ersten Blick wie eine authentische E-Mail aussieht, kann sich beim zweiten Blick als Phishing-Versuch offenbaren. Ist die gefälschte Mail gut gemacht, kann sie Filter wie SpamAssassin mit einiger Wahrscheinlichkeit umgehen und landet direkt im Posteingang des Nutzers. Aber selbst wenn der Nutzer vorsichtig ist und die E-Mail-Adresse des Absenders beim Öffnen gewissenhaft prüft, ist das keine Garantie, dass die Nachricht auch tatsächlich von dieser Adresse stammt.

Kriminelle verfolgen mit Phishing-Mails ein konkretes Ziel: das Vertrauen der Nutzer zu gewinnen und sie zu animieren, vertrauliche Daten wie Passwörter preiszugeben (Social Engineering). Senden die Täter ihre Phishing-Mails von einer echten E-Mail-Adresse einer Organisation, auf die sie selbst keinen Zugriff haben, gewinnen sie potenziell mehr Vertrauen der Nutzer, denn vielen Anwendern ist nicht bewusst, dass man Absenderadressen leicht fälschen kann. Möglich ist das durch eine konzeptionelle Schwachstelle im SMTP-Protokoll: Einen Mechanismus für die Authentifizierung der Absenderadresse gibt es im Protokoll selbst nicht.

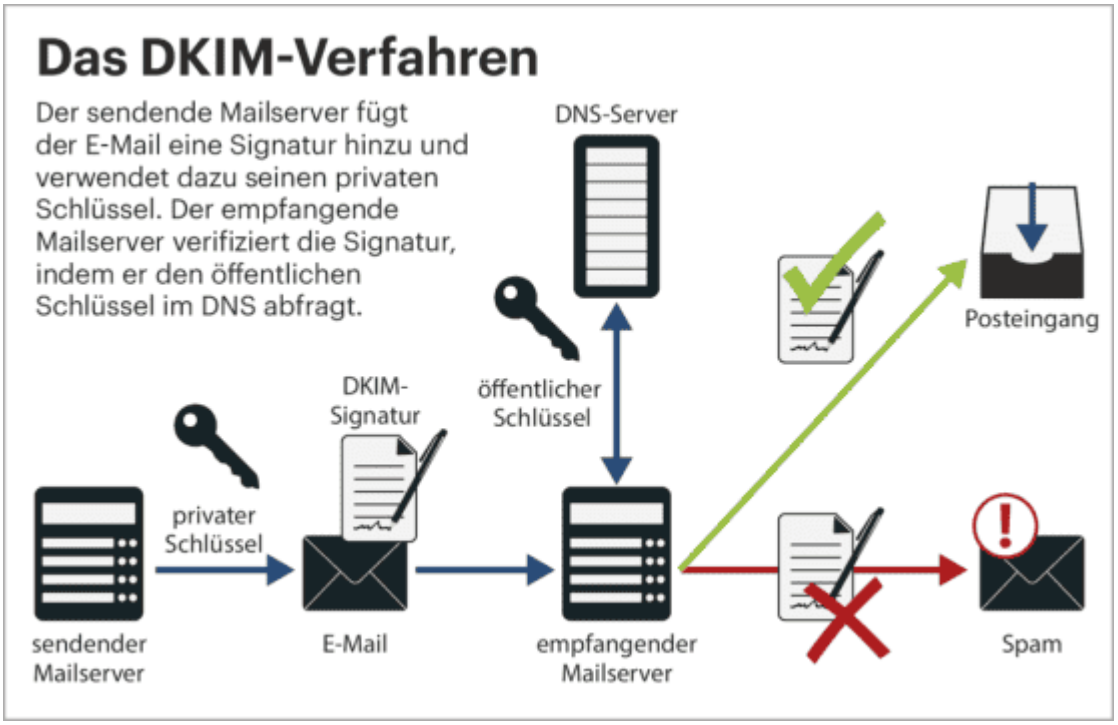
Bereits 2004 haben sich Yahoo und Cisco zusammengeschlossen und gemeinsam einen Standard konzipiert, der das Problem lösen soll: „DomainKeys Identified Mail“ (DKIM). Seit 2011 ist DKIM als Internetstandard von der Internet Engineering Task Force (IETF) anerkannt und wird von vielen Mailserverbetreibern eingesetzt. Das Fälschen von Absenderadressen (Mail-Spoofing) soll dadurch erschwert werden, dass jeder ausgehenden E-Mail eine digitale Signatur als Mail-Header beigefügt wird. Die Signatur im Header kann vom empfangenden Mailserver validiert werden. Mails mit gefälschter Absenderadresse können so erkannt und markiert oder entsorgt werden. Wie DKIM im Detail funktioniert, erfahren Sie im Kasten rechts.

DKIM: Mit Signaturen gegen Betrüger

DKIM ist ein Standard, um die Echtheit der versendenden Domain

einer E-Mail zu prüfen. Anders als zum Beispiel PGP ist für DKIM der Betreiber des Mailservers verantwortlich – als Nutzer kann man das Verfahren nicht einrichten. Ein Serverbetreiber, der DKIM-Signaturen an seine Mails anhängen möchte, generiert ein Schlüsselpaar aus öffentlichem und privatem Schlüssel. Um den öffentlichen Schlüssel bekannt zu machen, kommt DNS zum Einsatz: Den öffentlichen Schlüssel legt der Administrator als TXT-Record in der DNS-Zone seiner Domain ab. Der private Schlüssel darf den Mailserver nicht verlassen.

Beim Versenden von Mails werden zwei Prüfsummen berechnet: eine für ausgewählte Teile des Headers, eine für den Body der Mail. Die Prüfsummen werden mit dem privaten Schlüssel per RSA signiert und als Mailheader DKIM-Signature der E-Mail beigefügt, ergänzt um weitere Informationen. Zu denen zählen unter anderem die Absender-Domain, die Namen aller signierten Header-Felder sowie der sogenannte Selektor. Der Selektor entspricht dem Namen des DNS-Eintrags, in dem der öffentliche Schlüssel liegt. Die Liste der mitsignierten Header-Felder muss mindestens das Feld From: enthalten, also die Absenderadresse, die auch dem Empfänger angezeigt wird. So ist sichergestellt, dass nachträgliche Manipulationen die Signatur ungültig machen.



Empfängt ein Mailserver eine digital signierte E-Mail und ist der Server so eingerichtet, dass er DKIM prüft, fragt er aus dem DNS für die angegebene Domain den öffentlichen Schlüssel mit dem Namen des Selektors ab. Mit dem öffentlichen Schlüssel kann er die Echtheit der digitalen Signatur bestimmen. Ist die Prüfung erfolgreich, ist gewährleistet, dass die E-Mail von einem authentischen Absender stammt und nicht verändert wurde. Schlägt sie fehl, kann das ein Indiz dafür sein, dass die E-Mail gefälscht ist. Was dann passiert, kann der Betreiber des empfangenden Servers bestimmen. Oft führt das Scheitern zur sofortigen Ablehnung der E-Mail, manchmal wird sie nur als Spam-verdächtig markiert. Das Ergebnis der Prüfung fügt der empfangende Mailserver mit dem Header Authentication-Results an die Mail an. dkim=pass zeigt an, dass die Prüfung erfolgreich war, dkim=fail, dass sie fehlschlug.

Fast alle Mailserver verlassen sich nicht auf eine Methode zum Filtern allein und schalten mehrere Filter in Reihe. Mit Inhaltsfiltern reagieren sie zum Beispiel auf typische Spam-Begriffe wie „Casino“ und „Viagra“. In solchen Umgebungen vergibt jeder Filter einen Punktwert für die Einordnung der Mail – überschreitet die Summe aller Punkte einen Schwellwert, wird die Mail aussortiert oder markiert. Eine erfolgreiche DKIM-Prüfung wirkt sich in vielen Konfigurationen positiv auf die Vertrauenswürdigkeit aus und zieht Punkte ab.

Geteilte Server

Seit einigen Jahren bieten immer mehr Webhosting-Anbieter ihren Kunden das Signieren von E-Mails mit DKIM an. Bei einigen Providern ist DKIM sogar standardmäßig für alle Domains aktiviert, bei anderen reicht ein Klick im Kundencenter, um DKIM für einzelne oder alle Domains zu aktivieren. Die Hoster machen es den Kunden leicht und übernehmen das Hantieren mit Schlüsseln und DNS-Einträgen. Ohne Zutun des Kunden erstellen sie ein Schlüsselpaar, legen den öffentlichen Schlüssel im DNS als TXT-Record ab und

richten den privaten Schlüssel auf dem Mailserver ein. Fortan werden alle ausgehenden E-Mails automatisch mithilfe von DKIM signiert.

Bei Webhosting-Paketen sind sogenannte Shared Server verbreitet. Mehrere Kunden teilen sich einen Server, also dessen Ressourcen und Software. Dadurch kann der Anbieter mehr Kunden bedienen, als er tatsächlich physische Server vor Ort hat. Bei solchen Shared Servern muss gewährleistet sein, dass ein Kunde nicht auf die Daten eines anderen zugreifen kann. Für die Webseitendaten und Datenbanken funktioniert das auch sehr zuverlässig.

DMARC: Das Anti-Spam-Trio

Neben DKIM existieren zur Bekämpfung von Spam- und Phishing-Mails zwei weitere Verfahren: Sender Policy Framework (SPF) und Domain-based Message Authentication (DMARC).

SPF beruht auf der Annahme, dass alle E-Mails einer Domain von einer festen Anzahl von autorisierten Mailservern versendet werden. In einem TXT-Record veröffentlicht der Administrator die Adressen dieser Mailserver im DNS. Der Spam-Filter auf dem empfangenden Server kann bei der Entgegennahme der E-Mail durch das Abrufen dieses DNS-Eintrages prüfen, ob der sendende Mailserver zum Verschicken berechtigt ist. Was geschieht, wenn eine E-Mail über einen nicht autorisierten Mailserver versendet wird, kann ebenfalls im DNS-Eintrag festgelegt werden.

DMARC ist keine eigene Technik, sondern kombiniert die Ergebnisse der SPF- und DKIM-Prüfungen: Mit DMARC beschreibt der Administrator, ebenfalls in Form eines DNS-Eintrages, wie der empfangende Mailserver mit einer E-Mail umgehen soll, bei der die SPF- oder DKIM-Prüfungen fehlschlagen, und wen er darüber informieren soll.

Signaturen für fremde Domains

Doch werden auch die privaten DKIM-Schlüssel verschiedener Kunden sauber getrennt? DKIM ist schließlich nur sinnvoll, wenn gewährleistet ist, dass niemand gefälschte Signaturen generieren kann. Was für den Schutz von Kundendaten aufShared Servern gilt, muss auch für Schlüsselpaare gelten: Gültige DKIM-Signaturen auf Grundlage des privaten Schlüssels dürfen ausschließlich für E-Mails generiert werden, die vom Inhaber einer Domain stammen und nicht etwa von anderen Kunden, deren Accounts zufällig auf demselben Server liegen.

Providervergleich

Um herauszufinden, ob Hosting-Anbieter die DKIM-Signaturen ihrer Kunden auf demselben Server sauber trennen, haben wir 37 deutsche Anbieter unter die Lupe genommen und angefragt, ob sie DKIM für ihre Kunden aufShared Servern bereitstellen. Die Antwort: 17 Provider bieten DKIM für ihre Kunden gar nicht an. Vier Provider stellen DKIM nur aufInstanzen bereit, die nicht mit anderen Kunden-Domains geteilt werden (zum Beispiel virtuelle Server oder Managed Server). Übrig blieben 16 Provider für unsere Tests.

DKIM-Konfigurationsfehler bei deutschen Webhostern				
Anbieter	**getestetes Paket**	**DKIM-Unterstützung**	**Ergebnis**	**Reaktion**
All-Inkl.com	Premium	automatisch aktiv	verwundbar	DKIM für die PHP-Mailfunktion deaktiviert
Contabo	Paket L	automatisch aktiv	nicht verwundbar	
creoline	WordPress Hosting S	manuell aktivierbar	verwundbar	Lücke geschlossen
Febas	Professional	manuell aktivierbar	Test nicht möglich[1]	

DKIM-Konfigurationsfehler bei deutschen Webhostern				
Anbieter	**getestetes Paket**	**DKIM-Unterstützung**	**Ergebnis**	**Reaktion**
Hetzner	Level 4	manuell aktivierbar	verwundbar	Lücke geschlossen
hosting.de	Medium	automatisch aktiv	nicht verwundbar	
Hostinger	Premium	manuell aktivierbar	Test nicht möglich[1]	
netclusive	Easy 5.0	manuell aktivierbar	verwundbar	DKIM zunächst deaktiviert, Lücke später geschlossen
netcup	Webhosting 4000	automatisch aktiv	nicht verwundbar	
one.com	Entdecker	automatisch aktiv	nicht verwundbar	
Serverprofis	Private L 5.3	automatisch aktiv	nicht verwundbar	
Strato	Basic	automatisch aktiv	nicht verwundbar	
UD Media	Power 5.0	automatisch aktiv	verwundbar	Lücke geschlossen
webgo	SSD Profi	über den Support aktivierbar	teilweise verwundbar	
webhoster.de	Starter Tarif	manuell aktivierbar	nicht verwundbar	
WebhostOne	Basic	manuell aktivierbar	verwundbar	Lücke geschlossen
[1] keine anderen Kunden mit aktivem DKIM auf demselben Server				

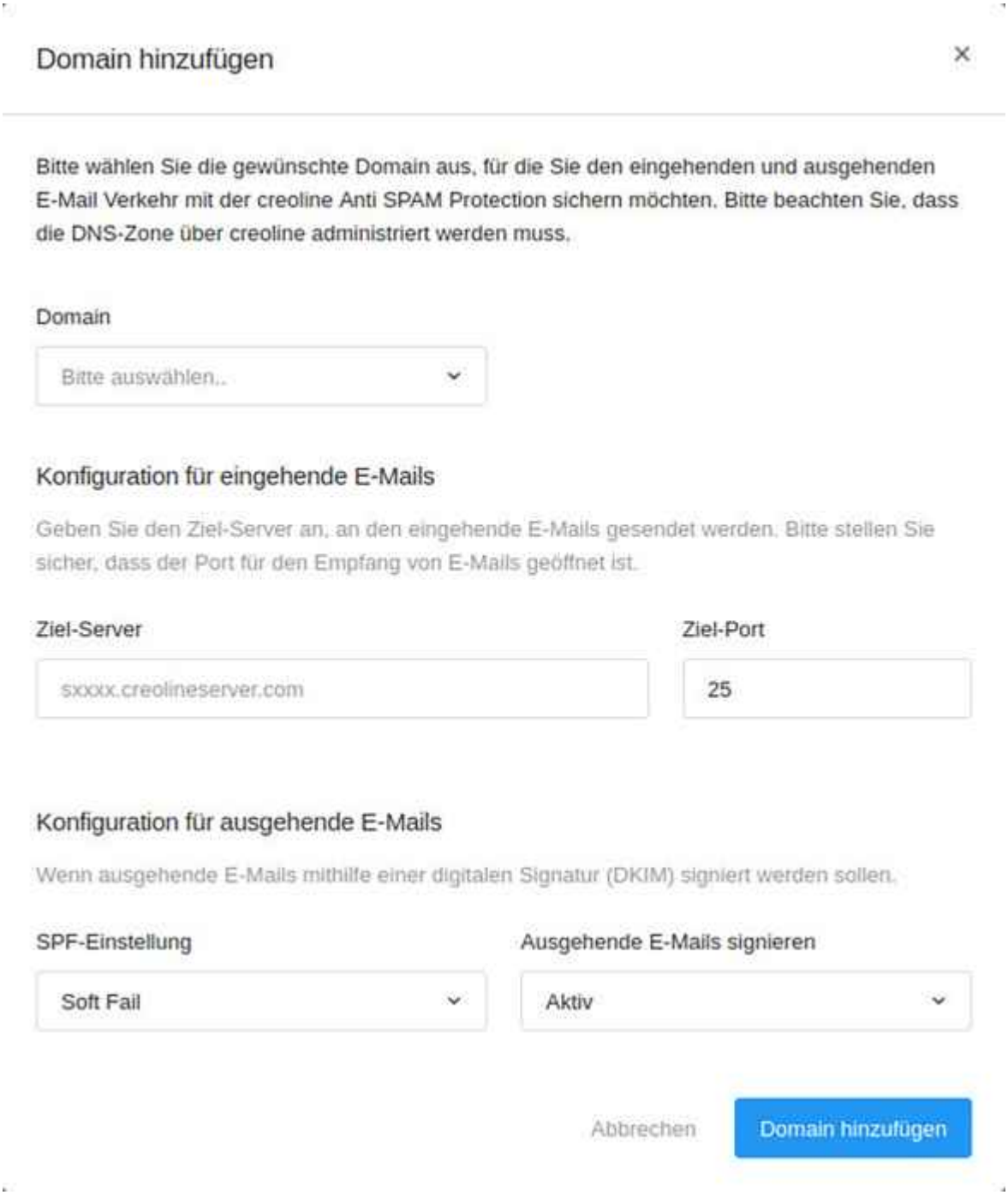
Bei All-Inkl.com, Contabo, hosting.de, netcup, one.com, Serverprofis, Strato und UD Media ist DKIM standardmäßig aktiviert. Bei einigen Anbietern war es notwendig, DKIM im Kundeninterface einzuschalten. Für unseren Test suchten wir den DKIM-Selektor unserer Test-Domains über die DNS-Einstellungen des Kundenportals. Dann gingen wir auf die Suche nach fremden Domains von anderen Kunden, die sich mit uns einen Server teilten. Diese Recherche ist mit einer Reverse--DNS-Suchmaschine im Internet schnell erledigt, indem man nach

der IP der eigenen Domain sucht. Für den Test brauchten wir eine fremde Domain, auf der ebenfalls DKIM aktiv war – ob das der Fall ist, findet man heraus, wenn man deren DNS-Einträge durchsucht. Bei den meisten Anbietern ging das schnell, da die DKIM-Selektoren für alle Domains identisch sind. All-Inkl.com, Hostinger und hosting.de vergeben individuelle DKIM-Selektoren auf Grundlage des Datums, an dem DKIM aktiviert wurde. In diesem Fall war etwas Ausdauer gefragt, da wir die fremden Domains manuell prüfen mussten. Nachdem wir fremde Domains mit aktivierter DKIM-Signatur auf „unseren“ Servern ausfindig gemacht hatten, konnte der Test beginnen.

Hoster ohne DKIM-Unterstützung	
Anbieter	DKIM-Unterstützung
1blu	nicht unterstützt
alfahosting	nicht unterstützt
centron	nicht unterstützt
checkdomain	nicht unterstützt
DM Solutions	nur für Managed Server
dogado	nicht unterstützt
DomainFactory	nicht unterstützt
ESTUGO	nicht unterstützt
goneo	nicht unterstützt
Host Europe	nicht unterstützt
Hostpress	nur für vServer
INWX	nicht unterstützt
IONOS 1&1	nicht unterstützt
manitu	nicht unterstützt
Mittwald	nicht unterstützt
OVH	nicht unterstützt
Packagecloud (D&T Internet)	nicht unterstützt
profihost	nicht unterstützt

Hoster ohne DKIM-Unterstützung	
Anbieter	DKIM-Unterstützung
Raidboxes	nur für vServer
TimmeHosting	nur für vServer
united-domains	nicht unterstützt

In allen getesteten Paketen stand uns PHP zur Verfügung – also nutzten wir die PHP-Funktion mail(), um eine E-Mail mit einer fremden Domain in der Absenderadresse, die auf demselben Server gehostet war wie unsere, an ein externes Postfach zu schicken. Eine glatte Fälschung also, die niemals hätte signiert werden dürfen.



Bei Creoline muss man DKIM im Kundencenter aktivieren. Der Anbieter war beim DKIM-Signaturdiebstahl verwundbar, konnte das Problem nach unserem Hinweis aber abstellen.
Bei sechs von sechzehn getesteten Anbietern war das Experiment erfolgreich: All-Inkl.com, creoline, Hetzner, netclusive, WebhostOne und UD Media hängten eine gültige Signatur mit dem privaten Schlüssel der fremden Domain an, obwohl wir zum Versenden nicht berechtigt waren. Unser empfangender Mailserver stufte die Mail als korrekt DKIM-signiert ein. Bei netclusive betraf dies nur Pakete auf dem Server hst1.ncsrv.de. Neue Pakete auf dem Server hst2.ncsrv.de waren

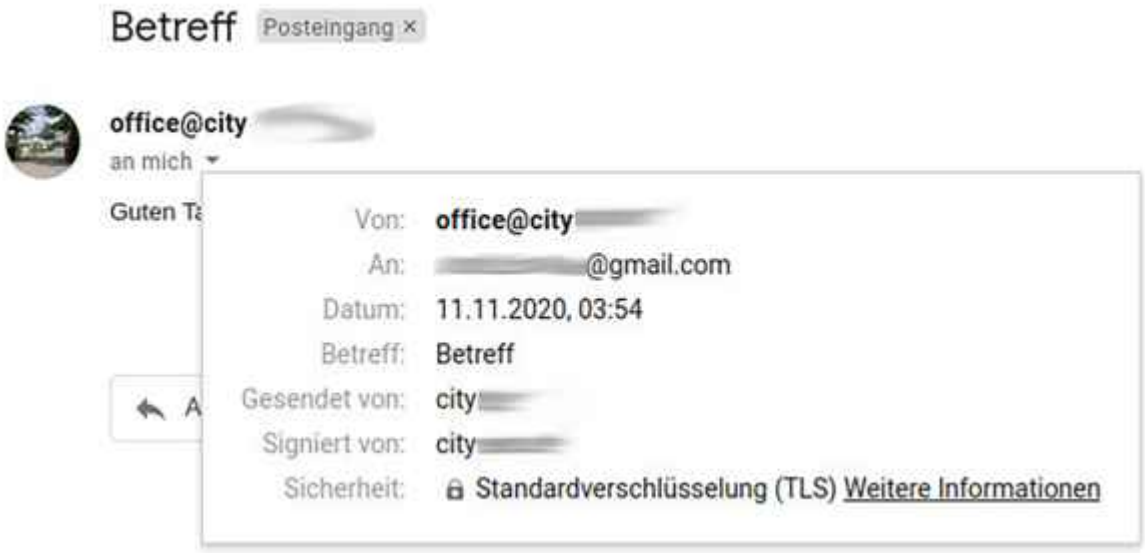
nicht betroffen.

Bei Febas, Hostinger und webgo konnten wir die Recherchen nicht abschließen, weil auf unserem Server keine anderen Domains DKIM aktiviert hatten und somit kein fremdes Schlüsselmaterial zum Testen vorhanden war.

Bei Serverprofis und Strato funktionierte der Angriffsversuch nicht. Beim Versenden aus unserem Account heraus wurde für die fremde Domain entweder eine DKIM-Signatur mit unserem privaten Schlüssel oder gar keine hinzugefügt. Zu einer unbefugt gültigen Signatur kam es nicht. Bei one.com wurden für fremde Adressen gar keine Mails verschickt, ein Angriff war also auch nicht möglich. Bei netcup und hosting.de konnten wir das Problem ebenfalls nicht reproduzieren. Dort werden Mails laut Auskunft des Supports nur dann DKIM-signiert, wenn man sie über den SMTP-Server verschickt und sich bei diesem authentifiziert. Das war hier ein wirkungsvoller Schutz gegen den Angriff.

Vertrauen verspielt

Unsere Untersuchung macht deutlich: Auch wenn das DKIM-Protokoll selbst gut konzipiert ist, haben es einige Webhoster durch fehlerhafte Konfiguration geschwächt. Bei den Anbietern, bei denen wir gefälschte E-Mails versenden konnten, haben wir die Wirksamkeit von DKIM ausgehebelt. Noch mehr: Da wir von einem autorisierten Mailserver verschickten, lieferten auch SPF und damit DMARC keine Fehler. Wir umgingen so auch vergleichsweise streng konfigurierte Spam-Filter und unsere E-Mail landete direkt im Posteingang ohne Spam-Verdacht. Auch sicherheitsbewusste Nutzer, die zum Beispiel mit dem Thunderbird-Plug-in„DKIM Verifier“ arbeiten, das bei jeder Mail das Ergebnis der Signaturprüfung prominent anzeigt, wären auf den Angriff hereingefallen.



Google Mail zeigt an, dass die Mail korrekt signiert wurde. Dabei wurde sie nicht von einem berechtigten Absender verschickt.
Für Spammer und Phisher ist dieser lockere Umgang mit den DKIM-Schlüsseln der Kunden ein großzügiges Angebot, gegen das Betreiber von Maileingangsservern und die Mailempfänger nichts tun können. Abhilfe schaffen können bei dem Problem nur die Hosting-Anbieter.

Nach unseren Experimenten kontaktierten wir die betroffenen Anbieter All-Inkl.com, creoline, Hetzner, netclusive, WebhostOne und UD Media und wiesen auf das Problem hin. Die Hoster, bei denen kein Test möglich war, wiesen wir darauf hin, dass das Problem möglicherweise auch bei ihnen besteht. Webgo bestätigte, dass die Lücke tatsächlich auf einigen Servern existiert – diese ältere Infrastruktur werde in nächster Zeit aktualisiert.

Creoline reagierte schnell mit einer Stellungnahme und wies zunächst darauf hin, dass Versuche, die Absenderadresse zu ändern, spätestens nach fünf Versuchen automatisch unterbunden wurden. Am nächsten Tag hatte man das Problem dann vollständig gelöst und die Manipulation war gar nicht mehr möglich. Netclusive antwortete einen Tag nach dem Hinweis, dass man DKIM vorübergehend ganz abgeschaltet habe, eine Woche später hatte man das Problem dann gelöst und DKIM wieder aktiviert.

Auch bei Hetzner konnte man das Problem bestätigen und stufte es als „mittelschwer“ ein – einen Tag nach der Meldung hatte man den Fehler beseitigt. Weil der Kunde DKIM selbst aktivieren muss, seien nach Angaben von Hetzner nur etwa fünf Prozent der Webhosting-Kunden betroffen gewesen. All-Inkl.com deaktivierte etwa eine Woche nach unserem Hinweis alle DKIM-Signaturen für Mails, die über die PHP-Funktion mail() verschickt wurden.

Private Schlüssel

Bei späteren Untersuchungen bemerkten wir, dass wir teilweise auch per SMTP Mails mit falscher Domain abliefern konnten, die dann signiert wurden – das Problem war bei einigen Anbietern also nicht auf die mail()-Funktion von PHP beschränkt. Die Lücke zeigte wieder ein altbekanntes Problem. DKIM basiert auf asymmetrischer Kryptografie, es gibt also einen öffentlichen und einen privaten Schlüssel. Wirklich sicher sind solche Verfahren nur, wenn der private Schlüssel auch wirklich privat bleibt. Also am besten auf einer Maschine, auf die nur der Inhaber selbst Zugriff hat. Wer DKIM bei einem Shared-Hosting-Dienst nutzt, gewinnt zwar viel Komfort, gibt aber seinen privaten Schlüssel aus der Hand und muss dem Dienstleister vertrauen. (*jam@ct.de*)

[/expand]

PHP 8

[expand title=“mehr lesen…“]

PHP 8 ist da, Version 7.2 ist tot

PHP 8 ist da, Version 7.2 ist tot

Das Entwicklerteam der meistgenutzten Web-Programmiersprache hat PHP 8 veröffentlicht. Gleichzeitig endet der Support für Version 7.2.

Neu in PHP 8 ist vor allem der Just-in-Time-Compiler (JIT), der schnellere Code-Ausführung ohne Änderungen am Code verspricht. Ob er dieses Versprechen in realen Umgebungen halten kann, wird sich erst in den nächsten Monaten zeigen, wenn größere PHP-Anwendungen bereit sind für PHP 8. Wie geplant erscheint die neue Major-Version 8 fast zeitgleich mit dem endgültigen Support-Ende von PHP 7.2. Wer diese Version noch nutzt, muss jetzt handeln, weil es keine Sicherheits-Updates mehr gibt.

Der Wechsel von 7.2 auf 7.3 (Support bis zum Nikolaustag 2021) oder 7.4 (Support bis zum 28.11.2022) verläuft vergleichsweise unspektakulär. Einige alte Konstruktionen werden als „deprecated“ markiert, weil sie in zukünftigen Versionen gestrichen werden sollen – solange man die Deprecation-Warnungen nicht anzeigen lässt, kann man die Seite damit aber problemlos betreiben.

Am besten überführt man seinen Code in eine Testumgebung, aktualisiert dort auf PHP 7.4, lässt sich alle Deprecation--Warnungen (E_DEPRECATED) anzeigen und behebt dann die Probleme – diese Warnungen sind meist recht aussagekräftig. Hinweise zur Problemlösung finden sich im Migrationsleitfaden von PHP (zu finden über ct.de/yc4z). Wer für die nächsten Jahre Ruhe haben möchte, kann es anschließend wagen, die Testumgebung auf PHP 8 umzustellen.

Der Wechsel auf PHP 8 ist durchaus mit Arbeit verbunden. Das liegt vor allem an uralten Zöpfen, die in Version 7 abgekündigt und jetzt endgültig abgeschnitten werden. Die seit

PHP 7 missbilligten Konstruktoren des alten Stils (eine Funktion, die wie die Klasse heißt) werden jetzt nicht mehr erkannt. Stattdessen muss der Konstruktor __construct() heißen (mit zwei Unterstrichen am Anfang). Es hilft nichts: Um das umzustellen, kommt man nicht umhin, sich alle eigenen Klassen nacheinander vorzunehmen. Weil dieses Relikt schon seit PHP 7.0 auf der Abschussliste steht, haben fast alle verbreiteten Open-Source-Bibliotheken die Umstellung bereits erledigt.

Wer ein Framework oder eine fertige Anwendung einsetzt, sollte vor dem Wechseln sicherstellen, dass deren Entwickler grünes Licht für PHP 8 gegeben haben. Die WordPress-Entwickler zum Beispiel wollen mit WordPress 5.6 so weit sein, das im Dezember erscheint; Nextcloud hat noch einige Baustellen vor sich. Die Entwickler des PHP-Frameworks Symfony haben ihre Hausaufgaben schon während der Beta-Phase erledigt. Das Framework kann bereits auf das mit PHP 8 eingeführte Konzept der Attributes zurückgreifen, also Meta-Informationen, die man in einer Kommentarzeile an eine Methode übergibt.

Für viele Serverbetreiber ist ein Wechsel von 7.2 auf eine noch unterstützte 7er-Version oder gar Version 8 aber nicht das dringlichste Problem: Die Statistikseite w3techs.com hat erhoben, dass Ende November 2020 noch immer 41,2 Prozent aller PHP-Websites mit Version 5 (davon rund die Hälfte mit der finalen Ausgabe 5.6) arbeiten. Sicherheits-Updates gab es dafür zuletzt am 1. Januar 2019. (jam@ct.de)

Supportzeiträume von PHP: ct.de/yc4z

[/expand]

Linux-Desktopumgebung Cinnamon 4.8 veröffentlicht

[expand title="mehr lesen…"]

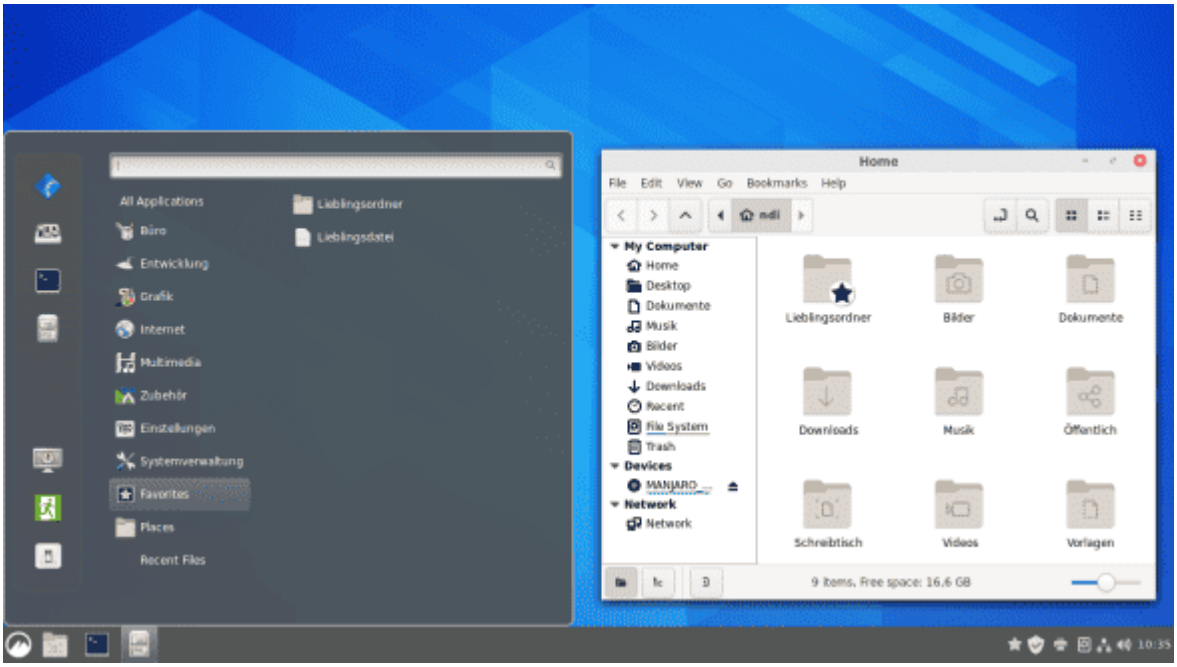
Linux-Desktopumgebung Cinnamon 4.8 veröffentlicht

Linux-Desktopumgebung Cinnamon 4.8 veröffentlicht

Die neue Version des Cinnamon-Desktops gibt einen Vorgeschmack auf das kommende Linux Mint 20.1 und hilft dabei, oft genutzte Dateien und Ordner schnell wieder zu finden.

Die Entwickler von Linux Mint und Cinnamon veröffentlichen den Cinnamon-Desktop 4.8 einige Wochen vor Linux Mint 20.1 „Ulyssa". Interessierte Nutzer können die Desktopumgebung bereits installieren und Feedback geben, bevor Cinnamon 4.8 mit dem neuen Linux Mint gegen Ende des Jahres ausgeliefert wird. Zu den Neuerungen bei Cinnamon zählt eine bessere Integration von Flatpaks.

Ist eine Anwendung als klassisches Paket und als Flatpak installiert, hängt Cinnamon 4.8 „(Flatpak)" an den Namen an. Nutzer können so unterschiedliche Versionen besser erkennen und gezielt starten. Ordner und Dateien lassen sich über das Kontextmenü im Dateimanager Nemo als „Favoriten" markieren. Zugriff auf die Favoriten besteht anschließend in der Seitenleiste von Nemo, im Hauptmenü und über das Favoriten-Applet.



In Nemo als Favoriten markierte Dateien und Ordner kann man über das Hauptmenü aufrufen.
Mit den Cinnamon „Spices“ können Nutzer den Desktop um Themes, Applets und Desklets erweitern. Ähnlich zu Gnome-Extensions sind die Spices nun in verschiedenen Versionen verfügbar. So wird die zum Cinnamon-Release passende Version der Erweiterung installiert, was Konflikte vermeidet.

Cinnamon 4.8 wechselt nach einer gewissen Zeit vom Bereitschaftsmodus in den Ruhezustand, wenn der Computer letzteren unterstützt.

Als technische Neuerung setzt der Cinnamon JavaScript Interpreter (CJS) auf Mozillas JavaScript-Engine „Mozjs78“. Dadurch soll Cinnamon schneller starten. Gleichzeitig erlaubt es anderen Distributionen den Cinnamon-Desktop leichter zu integrieren. Cinnamon 4.8 steht bereits in den Repositorys von Arch Linux zur Installation bereit. (ndi@ct.de)

OpenZFS: Linux und FreeBSD mit gemeinsamer Code-Basis

Das OpenZFS-Projekt arbeitet seit 2013 auf das Ziel hin, eine Open-Source-Alternative zu Oracles geschlossenem ZFS--Dateisystem zu entwickeln. **OpenZFS 2.0 soll den bestehenden FreeBSD-Port von ZFS und zfs-0.86 für GNU/Linux ablösen.** Die

Entwickler schaffen nun mit OpenZFS 2.0 eine geteilte Code-Basis, die auf ZFS für Linux aufbaut. Die OpenZFS-Binarys für FreeBSD und Linux können so aus dem gleichen GitHub-Repository gebaut werden. Kompatibilität mit macOS ist für ein späteres Release vorgesehen.

Darüber hinaus bietet OpenZFS 2.0 neue Funktionen. Ein fehlerhaftes Laufwerk in einem ZFS-Plattenverbund lässt sich mit dem neuen Verfahren „Sequential Resilver“ schneller wiederherstellen. Der Lese-Cache L2ARC stellt häufig genutzte Objekte bereit, die nicht mehr im Hauptspeicher vorgehalten werden. Nutzer können diesen Cache in OpenZFS 2.0 optional als persistent konfigurieren. So muss der L2ARC nicht bei jedem Neustart, Import oder Export des ZFS-Pools neu befüllt werden. Mit dem Befehl redacted zfs-send/receive können Daten beim Transfer von Datasets ausgenommen werden, um Speicherplatz zu sparen oder sensible Daten zu schützen.

Der neue Komprimierungsalgorithmus ZStandard (ZSTD) komprimiert ähnlich gut wie Gzip, aber schneller. FreeBSD-Nutzer können OpenZFS 2.0 bereits als Port installieren. Das kommende FreeBSD 13 enthält OpenZFS 2.0 als Standard. Canonical hat nach eigenen Angaben bereits in Ubuntu 20.04 LTS Funktionen aus OpenZFS 2.0 rückportiert. (ndi@ct.de)

Gnome-Projekt öffnet Türen

Softwareprojekte mussten, um Teil des Gnome-Projektes zu werden, auch dessen Infrastruktur nutzen und sich an die halbjährlichen Release-Intervalle der Desktopumgebung halten. **Mit der neuen Initiative „Gnome Circle“ möchte das Gnome–Projekt diese Barrieren für unabhängige Entwickler absenken.**

Als Anforderung gilt es weiterhin, die Gnome-Plattform-Bibliotheken zu nutzen und den Coding-Guidelines zu folgen. Im Gegenzug verspricht Gnome den Bekanntheitsgrad der Software zu steigern. Außerdem bietet die Gnome-Foundation

Reisekostenerstattungen und Zugang zu Gnome Ressourcen wie Hosting sowie Nextcloud- und GitLab-Instanzen. Entwickler können sich über die Website circle.gnome.org bewerben. (ndi@ct.de)

[/expand]