
cusy Design-System

Release 0.1.0

Veit Schiele

12.01.2024

Inhaltsverzeichnis

1	Das cusy Design-System ist Open Source	3
2	Unsere Leitprinzipien	5
3	Status	7
4	Wir unterstützen die Adoption	9
5	Design-System-Compliance	11
6	Kontaktiert uns	13
6.1	Prüft vorhandene Ressourcen	13
6.2	Reicht einen Vorschlag oder Beitrag ein	13
	Stichwortverzeichnis	121

Das **cusy Design-System** ist für Produkte und digitale Erlebnisse. Es besteht aus funktionierendem Code, Design-Tools und -Ressourcen und Richtlinien für die Benutzeroberfläche.

Ein Design-System ist eine Sammlung vorgefertigter, wiederverwendbarer Elemente – Komponenten, Muster, Anleitungen und Code –, mit denen konsistente digitale Erlebnisse schneller entwickelt werden können. Durch die Verwendung der vorgefertigten und universellen Elemente wird die Zeit, die Teams für Design und Entwicklung aufwenden, minimiert. Anstatt grundlegende Elemente immer wieder neu zu erstellen, kann die Zeit genutzt werden, um Produkte an spezifische Kundenbedürfnisse anzupassen.

Das cusy Design-System ist Open Source

Das Design-System wird von cusy finanziert und entwickelt, was bedeutet, dass wir es für unsere geschäftlichen Anforderungen entwickeln. Wir haben es jedoch zu einem Open-Source-System gemacht, das alle nutzen dürfen und zu dem alle beitragen können.

Unsere Leitprinzipien

offen

Das cusy Design-Sytem ist ein verteiltes Projekt, das sich an den Prinzipien der Open-Source-Bewegung orientiert. Diejenigen, die das cusy Design-Sytem nutzen, sind dieselben, die es auch entwickeln.

inklusiv

Das cusy Design-Sytem ist so konzipiert und gebaut, dass es für alle zugänglich ist, unabhängig von ihren Fähigkeiten oder ihrer Situation.

modular und flexibel

Die Modularität des Systems gewährleistet maximale Flexibilität bei der Ausführung. Die Komponenten des Systems sind so konzipiert, dass sie nahtlos ineinander greifen, je nachdem, welche Kombination den Bedürfnissen am Besten entspricht.

anwendungszentriert

Das cusy Design-Sytem ist anwendungszentriert. Durch die gründliche Erforschung der Bedürfnisse und Wünsche ist das cusy Design-Sytem auf reale Menschen ausgerichtet.

konsistent

Jedes Element und jede Komponente des cusy Design-Sytems wurde von Grund auf so konzipiert, dass es elegant mit anderen Komponenten zusammenarbeitet, um eine konsistente, kohärente User Experience zu gewährleisten.

KAPITEL 3

Status

commit activity 22/year

Wir unterstützen die Adoption

Wir führen Schulungen durch und veranstalten Meetings. Wir veranstalten regelmäßig [Meetups](#) und Design-Reviews. Außerdem stehen wir als Dozenten auf Konferenzen, in Bootcamps und auch sonst zur Verfügung. Wir engagieren uns in der Community. Wir streben danach, eines der besten Design-Systeme der Welt zu sein, und wir sind immer [offen für Feedback](#). Wir kommunizieren mit euch über den Status und die Roadmap über [GitHub](#), [Blogbeiträge](#) und [Mastodon](#). Wir bieten auch Unterstützung für diejenigen, die das cusy Design-System nutzen wollen.

Design-System-Compliance

Die Design-System-Compliance ist für die Marke cusy von entscheidender Bedeutung. Das cusy Design-System ist der digitale Ausdruck der Marke cusy und das Fundament, auf dem alle Produkte und digitalen Erlebnisse aufgebaut sind. Es ist die Grundlage für jede digitale Erfahrung, die Menschen während ihrer Beziehung zu cusy machen.

Die Schaffung von ansprechenden Benutzererlebnissen, die auf menschenzentriertem Design, hochwertigen Designstandards und der Marke cusy basieren, stellt sicher, dass wir konsistente und exzellente User Experience bieten und eine Marke aufbauen, die die Menschen über die gesamte Produktlinie hinweg lieben.

Kontaktiert uns

Habt ihr Fragen zu unserem Design-System? Habt ihr einen Fehler gefunden? Das Team des cusy-Design-System ist hier, um euch zu helfen. Bevor ihr beginnt, solltet ihr euch jedoch die folgenden Ressourcen ansehen, um zu überprüfen, ob wir eure Frage bereits beantwortet haben.

6.1 Prüft vorhandene Ressourcen

Als ersten Schritt solltet ihr den Inhalt dieser Website zu erkunden. Die Website ist umfassend und die meisten Leitlinien und Komponenten sind gut dokumentiert.

6.2 Reicht einen Vorschlag oder Beitrag ein

Das Team des cusy Design-System bietet Unterstützung für diejenigen, die das Design-System nutzen wollen:

- [GitHub Issues](#)
- [GitHub Pull Requests](#)

6.2.1 Barrierefreiheit

cusy ist der festen Überzeugung, dass Websites und Software für alle zugänglich sein sollten, unabhängig von ihren Fähigkeiten oder Beeinträchtigungen. Das cusy Design System stellt daher Richtlinien und Best Practices in Bezug auf Barrierefreiheit bereit.

Überblick

Zugänglichkeit

Barrierefreies Design hilft nicht nur Personen mit Einschränkungen, es bietet auch bessere Erfahrungen für alle. Alle cusy Design-Komponenten berücksichtigen die [Web Content Accessibility Guidelines \(WCAG\) 2.0, Section 508](#) und europäischen Standards. Die Muster des cusy Design-System sind für Personen wahrnehmbar, bedienbar und verständlich, selbst wenn ein Screen-Reader oder andere unterstützende Technologien verwendet werden.

Die Verwendung des cusy Design System wirkt sich jedoch auch auf die Zugänglichkeit eines Produkts aus. Lest diese Seite, um mehr über Barrierefreiheit im Allgemeinen zu erfahren. Verwendet dann während der Arbeit die rollenspezifischen Richtlinien zur Barrierefreiheit des cusy Design-System, um Produkte herzustellen, die jede*r verwenden kann.

Damit ein Produkt als zugänglich angesehen werden kann, sollte es

- für alle Personen die gleichen Vorteile erhalten, unabhängig von ihren Fähigkeiten
- von allem Personen dem jeweiligen Kontext angepasst werden können.

Kategorien von Einschränkungen

Einschränkungen können in drei Haupttypen unterteilt werden:

Tab. 1: Typen von Einschränkungen

Art	Beschreibung
situationsbedingt	Eine Person mit typischer Sehstärke kann z.B. Schwierigkeiten mit einem Bildschirm in heller Umgebung haben. Oder eine andere Person kann in einer Bibliothek ein Video nur mit ausgeschaltetem Ton ansehen.
temporär andauernd	Eine Person mit gebrochenem Handgelenk kann möglicherweise nicht tippen, erlangt jedoch die Fähigkeit nach der Heilung zurück. Meist wird nur diese Art als »Behinderung« verstanden.

Blinde

- Können Screen-Reader verwendet werden?
- Können Braille-Reader verwendet werden?
- Kann die Website oder die Software ohne Mauszeiger verwendet werden?
- Werden visuelle Informationen effektiv in Text übersetzt? Können Bilder und Plots nur durch die Metadaten verstanden werden?
- Lässt sich die Website oder die Software auch nur akustisch erfahren?

Sehbehinderte

Sehbehinderung kann ein teilweises Sehen mit einem oder beiden Augen umfassen und von leicht bis schwer reichen. Sie betrifft ungefähr 4% der Weltbevölkerung.

- Können sie Bildschirmleser, Bildschirmlupen, kontrastreiche Modi und/oder monochrome Anzeigen verwenden?
- Lässt sich die Website erfahren, wenn der Browser auf eine größere Schrift eingestellt ist?
- Ist eine [Keyboard-Only Navigation](#) möglich?
- Wird [NoCoffee](#) o.ä. verwendet um ein besseres Verständnis der verschiedenen Sehbehinderungen zu erhalten?

Farbenblinde

Farbenblindheit betrifft 8% aller Männer und 0,4% der Frauen.

- Werden die Kontraste der Website oder der Software in Hinblick auf Farbenblindheit überprüft, z.B. (zum Beispiel) mit dem Sketch-Plugin [Stark](#)?

Gehörlose und Schwerhörige

- Gibt es Untertitel und alternative Darstellungen von Audio? Beachtet dabei, dass alle davon profitieren können, z.B. in einer sehr lauten Umgebung.
- Werden Videos und Animationen transkribiert?

Körperliche Einschränkungen

- Sind alle Aktionen über die Tastatur zugänglich und effizient? Viele Personen bevorzugen die Navigation mit einer Tastatur vor der mit einer Maus, da dies die Produktivität erhöhen kann.

Kognitive Einschränkungen

- Bleiben Informationen sichtbar, die zu einem späteren Zeitpunkt benötigt werden?
- Sind schnell sich ändernde visuelle Reize nur optional? Werden Personen mit Epilepsie ggf. (gegebenenfalls) davor gewarnt?
- Werden Heuristiken für kognitive Belastung im Design berücksichtigt?

Globale Barrierestandards

- Die [Web Accessibility Initiative \(WAI\)](#) des [World Wide Web Consortium \(W3C\)](#) ist ein Versuch, die Zugänglichkeit des World Wide Web für Menschen aller Fähigkeiten zu verbessern.
- Mitwirkende der WAI erstellen und pflegen [Richtlinien zur Barrierefreiheit von Webinhalten \(WCAG\)](#), dem globalen Standard für Barrierefreiheit.
- Im [ARIA Authoring Practices Guide](#) wird euch gezeigt, wie barrierefreie Semantik in Web-Patterns und Widgets eingebaut werden kann.

Ressourcen

- IBM Accessibility Checklist
- W3C low vision requirements

Werkzeuge

- High-contrast Chrome plugin
- Stark Sketch plugin
- NoCoffee vision simulator
- Kontrastrechner
- Pally

Farbe

Einführung

Die Komponenten des cusy Design-System wurden entwickelt, um das gesamte Spektrum von Sehbehinderungen zu berücksichtigen. Beim Entwickeln des Systems achten wir jedoch sorgfältig darauf, dass die Komponenten auch korrekt verwendet werden.

Semantik

hr könnt in HTML und SVG Farben mit `aria-label` semantisch auszeichnen, z.B.:

```
<svg aria-label="red">
  <rect fill="#d9e288"/>
</svg>
```

Farbkontrast

Das cusy Design-System erfüllt die **WCAG AA-Standards** für die beiden Standard-Themes des Systems, einschließlich der Farbkontrastverhältnisse.

Standardtext

Standardtext und Textbilder müssen ein Kontrastverhältnis von mindestens 4,5:1 haben.

Großer Text

Großer Text (mindestens 24px normal und hell oder 19px halbfett) muss ein Kontrastverhältnis von 3:1 haben.

Text vor nicht statischem-Hintergrund

Wenn Text auf einem Verlaufshintergrund oder Bild gerendert wird, muss sichergestellt werden, dass die Textfarbe an allen Stellen, an denen sie angezeigt wird, den Kontraststandards entspricht. Dies ist besonders wichtig für Parallax-Anwendungen oder Animationen, bei denen sich Text und Hintergrund unabhängig voneinander bewegen.

UI-Komponenten

Visuelle Informationen, die zum Anzeigen von Zuständen und Grenzen von UI-Komponenten verwendet werden, müssen ein Kontrastverhältnis von 3:1 zu benachbarten Farben aufweisen. Dies schließt auch Formularelemente ein.

Siehe auch:

- [IBM Checkpoint 1.4.3 Contrast \(Minimum\)](#)

Farbenblindheit

Farbenblindheit beinhaltet Schwierigkeiten beim Wahrnehmen oder Unterscheiden zwischen Farben sowie die Empfindlichkeit gegenüber Farbhelligkeit. Es betrifft ungefähr einen von zwölf Männern und eine von zweihundert Frauen weltweit.

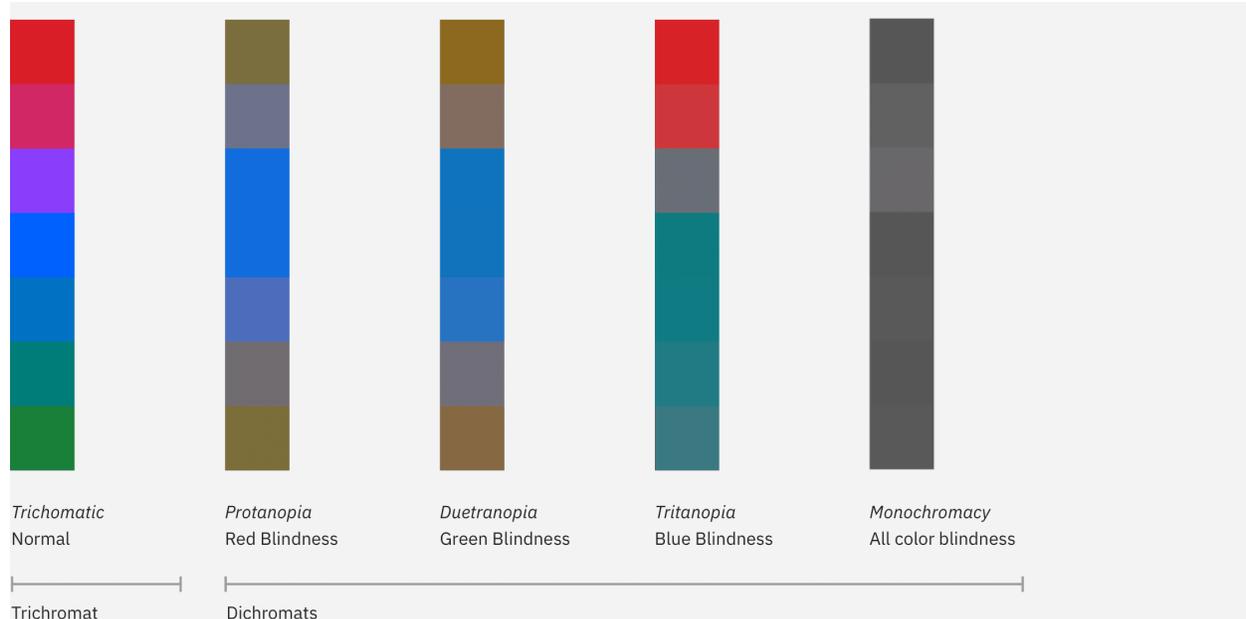
Art	Farbenblindheit
Protanopie	Rot-Grün
Tritanopie	Blau
Deutanopie	Grün
Monochromie	Alle Farben

Entwerfen für Farbenblindheit

Verlasst euch nicht nur auf Farbe, um Bedeutung zu vermitteln. Von Bedeutung ist sowohl

- Informationen übermitteln
- Aktionen anzeigen
- zum Beantworten von Fragen auffordern
- oder zwischen visuellen Elementen zu unterscheiden.

Wenn Ihr mit Farbe entwerft, kann für Euch ein Simulator für Farbenblindheit hilfreich sein, um die Sichtbarkeit von Inhalten zu überprüfen, wie z.B. das [Stark-Plugin](#) für Sketch.



Sehschwäche

Sehschwächen können teilweises Sehen eines oder beider Augen umfassen und von leicht bis schwer reichen. Sie betreffen etwa 4% der Weltbevölkerung.

Art	Sehschwäche
Geringe Schärfe	Auch als <i>verschwommenes Sehen</i> bekannt. Kann das Lesen von Text erschweren, da er unscharf erscheint.
Kontrastarme Empfindlichkeit	Verminderte Fähigkeit, feine Details zu bestimmen und ein Objekt von einem anderen Rzu unterscheiden
Eingeschränktes Sichtfeld	Die Sicht des Benutzers ist teilweise eingeschränkt. Kann zentrales Sehen und fleckiges Sehen umfassen.
Retinitis pigmentosa	Auch als <i>Tunnelblick</i> bekannt, bei dem nur zentrale Elemente zu sehen sind.

Entwerfen für Sehschwäche

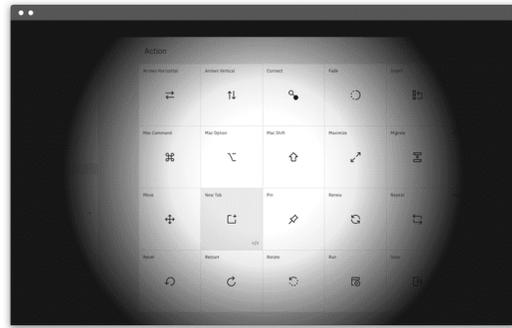
Um ein Verständnis für die verschiedenen Sehschwächen zu erhalten, empfehlen wir die Verwendung des [NoCoffee Vision Simulator](#) zur Vorschau von Websites. Ihr könnt ihn auch direkt als [Firefox add-on](#) installieren. Personen mit Sehschwächen verwenden möglicherweise einen Screen-Reader, um eine Vorschau der Website anzuzeigen. Daher sollten unbedingt die [Tastaturrichtlinien](#) befolgt werden, um sicherzustellen, dass die Seite in einer logischen Reihenfolge vorgelesen wird.

Siehe auch:

- [WhoCanUse](#)



Blurred vision



Tunnel vision

Tastaturzugriff

Tastaturzugriffe sind hilfreich für Personen, die sich auf eine Tastatur verlassen oder diese bevorzugen. Stellt sicher, dass alle interaktiven Inhalte über Tastaturfunktionen verfügen. Zu den allgemeinen Tastaturinteraktionen gehört die Verwendung der Tabulatortaste zum Auswählen verschiedener interaktiver Elemente auf einer Seite und die Verwendung der Eingabe- oder Leertaste zum Aktivieren eines fokussierten Elements.

Siehe auch:

- [IBM Checkpoint 2.1.1 Keyboard](#)

Fokusindikatoren

Die Tabulatortaste soll durch alle interaktiven Elemente auf einer Seite navigieren und zwar in der Reihenfolge, in der sie im HTML-Dokument angezeigt werden. Eine standardmäßige visuelle Anzeige wird vom verwendeten Webbrowser bereitgestellt. Zur Anzeige wird ein Rand um das fokussierte Element verwendet.

Siehe auch:

- [IBM Checkpoint 2.4.7 Focus Visible](#)

Navigationsreihenfolge

Die Reihenfolge, in der interaktive Elemente den Fokus erhalten, sollte logisch und vorhersehbar sein. Ein allgemeiner Ablauf kann mit der Kopfzeile beginnen, gefolgt von der Hauptnavigation, der Inhaltsnavigation (von links nach rechts und von oben nach unten) und mit der Fußzeile enden. Versucht, allen Personen die gleiche Erfahrung zu bieten.

Verwendet einfach zugängliche Elemente in der Navigation, um Links, Schaltflächen und Formularsteuerelemente mit einer Tastatur zu aktivieren. Verstärkt semantisches HTML, um Absicht und Bedeutung zu vermitteln, anstatt nur das Erscheinungsbild eines Elements zu definieren. Erweitert bei Bedarf die Elemente um [aria-label](#)-Attribute.

Bemerkung: Fügt keine nicht-interaktiven Elemente in die Tabulatorreihenfolge ein. Vermeidet auch mehrfach aufeinanderfolgende Links zum selben Ziel, indem ihr einige ggf. aus der Tastaturreihenfolge ausnehmt mit

```
tabindex = "-1"
```

Siehe auch:

- [WCAG success criterion: Focus Order](#)
- [Optimizing keyboard navigation using tabindex and ARIA](#)
- [IBM Checkpoint 2.4.3 Focus Order](#)

ARIA-Landmarks

Kommuniziert den Personen mit Screen-Reader die verschiedenen Bereiche des Bildschirms und eure Vorgehensweise mit ARIA-Landmarks und mithilfe geeigneter HTML5-Label. Personen mit Screen-Reader können dann schnell zu einem beliebigen Bereich springen.

Siehe auch:

- [IBM Checkpoint 1.3.1 Info and Relationships](#)

Bewährte Praktiken für HTML and CSS

Code und Navigation strukturieren

Denkt bei der Strukturierung eurer Inhalte an die Code-Hierarchie, damit Screen Reader und Personen, die nur über die Tastatur verfügen, über Tabulatoren in einer logischen und vorhersehbaren Reihenfolge auf interaktive Elemente zugreifen können. Beginnt mit der Kopfzeile, gefolgt von der Hauptnavigation, dann der Seitennavigation (von links nach rechts und von oben nach unten) und enden Sie mit der Fußzeile. Ziel ist es, Personen über die Tastatur eine vergleichbare Benutzung zu ermöglichen wie Personen, die eine Maus benutzen.

Semantisches HTML

Verwendet native HTML-Elemente so oft wie möglich und verwendet sie für den richtigen Zweck. Diese Elemente bieten integrierte Vorteile für die Barrierefreiheit. Sie informieren Personen mit Screen-Reader darüber, wo sie sind und welche Interaktionen ihnen in diesem Kontext zur Verfügung stehen. Abgesehen von einer erhöhten Zugänglichkeit werden die Seiten auch einfacher zu entwickeln und zu warten sein, besser auf Mobilgeräten funktionieren und die Suchmaschinenoptimierung verbessern.

```
<body>
  <header>
    <nav>
      <ul>
        <li><a href="#">link</a></li>
        <li><a href="#">link</a></li>
        <li><a href="#">link</a></li>
      </ul>
    </nav>
  <h1>
```

(Fortsetzung auf der nächsten Seite)

(Fortsetzung der vorherigen Seite)

```

    Page Title
  </h1>
</header>

<section>
  <h2>
    My Blog Posts
  </h2>
  <article>
    <header>
      <p>
        Article Title
      </p>
    </header>
    <p>
      content
    </p>
  </article>
  <article>
    <header>
      <p>
        Article Title
      </p>
    </header>
    <p>
      content
    </p>
  </article>
  <aside>
    <p>
      Author info
    </p>
  </aside>
</section>

<footer>
  Copyright Info
</footer>
</body>

```

Klare Sprache

Berücksichtigt beim Hinzufügen von Inhalten kognitive Einschränkungen, Personen, deren Erstsprache nicht die Sprache ist, in der Ihre Inhalte geschrieben sind und Personen, die sich die Inhalte mit einem Screen-Reader erschließen. Vermeidet nach Möglichkeit Bindestriche, Abkürzungen, Akronyme (zumindest beim ersten Mal) und Tabellenlayouts, wenn keine Tabelleninhalte vorhanden sind. Verwendet bei Abkürzungen das native `<abbr/>`-Element mit dem `title`-Attribut.

```
<abbr title="HyperText Markup Language">HTML</abbr>
```

In Sphinx könnt Ihr dies erreichen mit der `abbr`-Rolle:

```
:abbr:`HTML (HyperText Markup Language)`
```

Aussagekräftige Textbeschriftungen

Berücksichtigt beim Beschriften von Elementen sehbehinderte Personen und stellt sicher, dass für Screen-Reader ein Textkontext vorhanden ist.

```
<div role="form" id="contact-info" aria-label="Contact information">
  <!-- content of the contact form -->
</div>
```

Zugängliche Tabellen

- Gebt Tabellenüberschriften immer mit `<th/>`-Elementen an und stellt sicher, dass sie herausragen.
- Verwendet bei Bedarf das Attribut `scope` um anzugeben, ob es sich um Überschriften für Zeilen oder Spalten handelt.
- Verwendet alternativen Text zusammen mit Tabellen für sehbehinderte Personen.
- Für die Beschreibung der Tabelle wird `<caption/>` bevorzugt, aber `<table summary=""/>` funktioniert auch.

Unterscheiden zwischen Mäusen und Touchscreens

Ihr könnt die Gestaltung von Links, Tasten etc. (etcetera) einfach mit der `hover`-Funktion variieren, je nachdem ob euer Publikum eine Maus oder einen Touchscreen verwendet. Dies funktioniert in den meisten aktuellen Browsern gut, bei einigen Android-Versionen emuliert langes Drücken jedoch `Hover`, sodass die Medienabfrage `hover: hover` wahr bewertet wird. Daher sollte auch noch eine zweite Abfrage nach `pointer: fine` erfolgen:

```
.some-component {
  /* Stil für Touchscreens, einschließlich Android-Geräte */
}

@media (hover: hover) and (pointer: fine) {
  .some-component {
    /* Stil für Mäuse und ähnliche Geräte */
  }
}
```

Siehe auch:

- Media features
- Detecting Hover-Capable Devices

Multimedia-Textalternativen

Jedes Bild, das nicht dekorativ ist, sollte mit `alt`-Text und einer aussagekräftigen Beschreibung des Bildes sowie einem `title`-Attribut versehen sein. Ihr könnt auch `aria-labelledby` zusammen mit dem `id`-Attribut verwenden. Für dekorative Bilder solltet ihr ein leeres `alt`-Attribut verwenden, da andernfalls der Screen-Reader die gesamte Bild-URL liest.

Audio-Alternativen

Stellt Videos mit Untertiteln oder Transkriptionen in Audiodateien bereit:

```
<video controls>
  <source src="example.mp4" type="video/mp4" />
  <source src="example.webm" type="video/webm" />
  <track kind="subtitles" src="example_subtitles.vtt" srclang="en" />
</video>
```

Schriftsymbolbibliotheken

Wir sind bei der Verwendung von Schriftsymbolbibliotheken auf zwei verschiedene Anwendungsfälle gestoßen, die ihr berücksichtigen solltet:

Dekorative Symbole

werden nur zu dekorativen Zwecken verwendet. Wenn sie von der Seite entfernt würden, würden Personen Eure Seite weiterhin verstehen und verwenden können.

In diesem Fall kann dem Element ein `aria-hidden="true"`-Attribut hinzugefügt werden, z.B.:

```
<i class="fas fa-camera-retro" aria-hidden="true"></i>
```

Semantische Symbole

sind solche, mit denen bedeutendes vermittelt werden soll, z.B. für Button, Formularelemente etc.

Neben dem `aria-hidden="true"`-Attribut sollten zwei weitere Ergänzungen hinzugefügt werden:

- eine Textalternative z.B. in einem ``-Tag, das via CSS ausgeblendet wird.
- ein `title`-Attribut auf dem Symbol, um einen Tooltip für sehende Personen bereitzustellen.

```
<i aria-hidden="true" class="fas fa-trash" title="Delete this item?"></i>
<span class="sr-only">"Delete this item?"</span>
```

Alternativ kann SVG mit JavaScript verwendet werden mit. In diesem Fall wird durch das `title`-Attribut folgende drei Elemente hinzugefügt:

- ARIA-Rolle `role="img"`
- `title`-Tag mit `id`-Attribut
- `aria-labelledby`-Attribut

Aus

```
<i title="Magic is included!" class="fas fa-magic"></i>
```

wird

```
<svg title="Magic is included!" class="svg-inline--fa fa-magic fa-w-16" aria-
↳labelledby="svg-inline--fa-title-1" data-fa-i2svg="" data-prefix="fas" data-icon=
↳"magic" role="img" xmlns="http://www.w3.org/2000/svg" viewBox="0 0 512 512">
  <title id="svg-inline--fa-title-1">Magic is included!</title>
  <path fill="currentColor" d="M101.1 505L7 410.9c-9.4-9.4-9.4-24.6 0-33.9L377.
↳7c9.4-9.4 24.6-9.4 33.9 0L94.1 94.1c9.4 9.4 9.4 24.6 0 33.9L135 505c-9.3 9.3-24.5.
↳9.3-33.9 0zM304 159.2l48.8 48.8 89.9-89.9-48.8-48.8-89.9 89.9zM138.9 39.3l-11.7.
↳23.8-26.2 3.8c-4.7-6.6 6.5-3.2 9.8l19 18.5-4.5 26.1c-.8 4.7 4.1 8.3 8.3 6.1L144.
↳115l23.4 12.3c4.2 2.2 9.1-1.4 8.3-6.1l-4.5-26.1 19-18.5c3.4-3.3 1.5-9.1-3.2-9.
↳8L160.8 63l-11.7-23.8c-2-4.1-8.1-4.1-10.2.1zm97.7-20.7l-7.8 15.8-17.5 2.6c-3.1-5-
↳4.4 4.3-2.1 6.5l12.6 12.3-3 17.4c-.5 3.1 2.8 5.5 5.6 4L240 69l15.6 8.2c2.8 1.5 6.
↳1-.9 5.6-4l-3-17.4 12.6-12.3c2.3-2.2 1-6.1-2.1-6.5l-17.5-2.5-7.8-15.8c-1.4-3-5.4-
↳3-6.8-.1zm-192 0l-7.8 15.8L19.3 37c-3.1-5-4.4 4.3-2.1 6.5l12.6 12.3-3 17.4c-.5 3.
↳1 2.8 5.5 5.6 4L48 69l15.6 8.2c2.8 1.5 6.1-.9 5.6-4l-3-17.4 12.6-12.3c2.3-2.2 1-6.
↳1-2.1-6.5l-17.5-2.5-7.8-15.8c-1.4-3-5.4-3-6.8-.1zm416 223.5l-7.8 15.8-17.5 2.5c-3.
↳1-5-4.4 4.3-2.1 6.5l12.6 12.3-3 17.4c-.5 3.1 2.8 5.5 5.6 4l15.6-8.2 15.6 8.2c2.8.
↳1.5 6.1-.9 5.6-4l-3-17.4 12.6-12.3c2.3-2.2 1-6.1-2.1-6.5l-17.5-2.5-7.8-15.8c-1.4-
↳2.8-5.4-2.8-6.8 0z"></path>
</svg>
```

Siehe auch:

- Sara Soueidan: [Accessible Icon Buttons](#)
- [Font Awesome Accessibility](#)

Datenvisualisierung

Berücksichtigt sehbehinderte Personen bei der Einbeziehung von Datenvisualisierungen. Datentabellen können dabei eine Alternative zu den Datenvisualisierungen für Screen-Reader sein. Berücksichtigt auch die Farbauswahl für farbenblinde Personen. Weitere Empfehlungen sind:

- Verlasst euch nicht auf Farben um die Daten zu erklären
- Verwendet keine kontrastarmen Farben
- Versteckt keine wichtigen Daten hinter Interaktionen
- Überfordert euer Publikum nicht mit zu vielen Informationen
- Verlasst euch nicht auf hover-Effekte; diese funktionieren nicht auf Mobilgeräten
- Verzichtet auf Animationen und Scroll-Hijacking, wenn diese nicht wesentlich zur Erklärung der Daten beitragen
- Beachtet bereits beim Design die Zugänglichkeit
- Beachtet auch kleinere Displays, z.B. bei Mobilgeräten
- Verwendet kontrastreiche Farben und Muster
- Verwendet Etiketten und Legenden
- Übersetzt die Daten in eine klare Sprache
- Gebt Kontext an und erklärt die Visualisierung

Siehe auch:

- [Case Study: Implementing Accessible Data Charts for the Khan Academy 2018 Annual Report](#)
- [Accessibility Considerations In Data Visualization Design](#)

- [Dataviz Accessibility Resources](#)
- [Writing Alt Text for Data Visualization](#)
- [WAI tutorials: Complex Images](#)
- [Using VoiceOver to Evaluate Web Accessibility](#)
- [How Does This Data Sound? keyboard interaction, sonification etc.](#)
- [Heather Migliorisi: Accessible SVGs](#)
- [Accessible SVG line graphs](#)
- [Accessible SVG flowcharts](#)
- [Carie Fisher: Creating Accessible SVGs](#)
- [Sarah Higley: Quick Tips for High Contrast Mode](#)
- [Melanie Richards: CurrentColor SVG in forced colors modes](#)
- [Val Head: Designing With Reduced Motion For Motion Sensitivities](#)

D3.js

D3.js ist eine Javascript-Bibliothek zur Datenvisualisierung. Um Datenvisualisierungen zugänglicher zu machen, haben wir ein paar Tipps zusammengestellt:

1. Um die Farben zu erläutern, könnt ihr eine Legende hinzufügen, z.B. mit:

```
const legend = chart.append("g").attr("aria-label", "Legend");
```

Alternativ kann auch ein Titel hinzugefügt werden:

```
const legend = chart.append("g");
legend.append("text")
  .text("Legend")
  .attr("class", "legendTitle");
```

2. Nun können wir die Erläuterungen hinzufügen, z.B.:

```
legend.append("rect")
  .attr("fill", function(d){return hot(d) });

legend.append("text")
  .text(">30 °C");
```

3. Erläutern der Daten, z.B. für ein Balkendiagramm:

```
chart.selectAll(".label")
  .data(data)
  .enter().append("text")
  .text(row => row.year);
```

4. Optisch sind die Daten jetzt schon sehr viel zugänglicher, aber mit Bedienhilfen werden nun auch die Tage auf der x-Achse erschlossen, und würden z.B. alle vorgelesen werden. Die Zugänglichkeit würde deutlich erhöht werden, wenn die Ausgabe so etwas liefern würde wie *Die durchschnittliche Jahrestemperatur betrug 2011 9,6 Grad Celsius.*

Hierfür sollten dann nicht zunächst die Daten durchlaufen werden und anschließend die Achsenbeschriftungen sondern jedes Datum mit der zugehörigen Erläuterung:

```
const ticks = chart.selectAll(".tick")
  .data(data)
  .enter().append("g")
  .attr("class", "tick");

ticks.append("text")
  .text((data) => data.year);

ticks.append("text")
  .text(row => row.temperature)
  .attr("class", "label");
```

Dies gibt folgendes XML aus:

```
<g>
  <text>2011</text>
  <text>9,6 °C</text>
</g>
...
```

- Bei vielen Achswerten sollten evtl. (eventuell) nicht alle Werte auf der Achse angezeigt werden. `display: none`; oder `visibility: hidden` sind jedoch keine Lösung, da die Werte dann z.B. auch nicht vorgelesen würden. Wir können jedoch die Positionierung der Achsenbeschriftungen so verändern, dass sie außerhalb des sichtbaren Rahmens stehen.

Siehe auch:

- [Accessibility in d3 Bar Charts](#)
- [Accessibility in d3 Donut Charts](#)

Vega

Vega ist eine deklarative Sprache zum Erstellen, Speichern und Teilen interaktiver Visualisierungsdesigns. Ab Version 5.11 unterstützt sie einige ARIA-Attribute für die Ausgabe von SVG-Dateien:

aria

schließt ARIA-Attribute in SVG-Dateien ein. Der Standardwert ist `true`. Bei `false` wird das `aria-hidden`-Attribut für die entsprechende SVG-Gruppe gesetzt.

description

liefert in `aria-label` eine Textbeschreibung des SVG-ELEMENTS, wenn `aria` den Wert `true` hat.

Siehe auch:

- [Vega Marks: Accessibility Properties](#)

Mit [Flight Passengers Example](#) gibt es ein Beispiel für die Verwendung der Vega-Anweisungen `aria`- und `description` durch [Vega-Lite](#):

- Die horizontalen Linien sollen nicht zugänglich werden. Daher wurde in der Vega-Konfigurationsdatei `flight-passengers.vg.json` die horizontale Linie mit `"aria": false`, ausgezeichnet.

```

108   "marks": [
109     {
110       "type": "line",
111       "aria": false,
112       "from": {"data": "traffic"},
113       "encode": {
114         "enter": {
115           "x": {"scale": "x", "field": "unit0"},
116           "y": {"scale": "y", "field": "change"},
117           "stroke": {"value": "steelblue"},
118           "strokeWidth": {"value": 3}
119         }
120       },
121       "zindex": 2
122     },

```

In der generierten SVG-Datei `flight-passengers.svg` erhält die zugehörige Gruppe dadurch die Annotation `aria-hidden="true"`:

```

69       <g class="mark-line role-mark" aria-hidden="true">
70         <path d="M24.000000019013836,20.112588094658268L93.
↪2076502901645,26.132110486376767L160.09289618980006,14.924455087752282L229.
↪2076502860901,20.695076449129235L298.32240438238017,9.587662748507821L362.
↪9781420853612,109.65285552622753L432.0000000067907,164.6837644873604" stroke=
↪"steelblue" stroke-width="3"/>
71       </g>

```

- Die Markierungen in der Zeitachse erhalten hingegen eine ausführliche Beschreibung (Englisch description) mit Monat, Jahr und prozentualer Veränderung zum Vorjahr:

```

123     {
124       "type": "symbol",
125       "from": {"data": "traffic"},
126       "encode": {
127         "enter": {
128           "description": {
129             "signal": "timeFormat(datum.unit0, '%B %Y') + ': ' + format(datum.
↪change, '+.1%') + ' change from prior year'"
130           },
131           "tooltip": {
132             "signal": "format(datum.change, '+.1%')"
133           },
134           "x": {"scale": "x", "field": "unit0"},
135           "y": {"scale": "y", "field": "change"},
136           "fill": {"value": "steelblue"}
137         }
138       },
139       "zindex": 2
140     }

```

In der generierten SVG-Datei erhalten die Pfade dadurch ein entsprechendes `aria-label`:

```

72       <g class="mark-symbol role-mark" role="graphics-object" aria-

```

(Fortsetzung auf der nächsten Seite)

```

↪roledescription="symbol mark container">
73     <path aria-label="October 2019: +6.1% change from prior year
↪" role="graphics-symbol" aria-roledescription="symbol mark" transform=
↪"translate(24.000000019013836,20.112588094658268)" d="M4,0A4,4,0,1,1,-4,0A4,4,0,1,
↪1,4,0" fill="steelblue"/>
74     <path aria-label="November 2019: +2.0% change from prior
↪year" role="graphics-symbol" aria-roledescription="symbol mark" transform=
↪"translate(93.2076502901645,26.132110486376767)" d="M4,0A4,4,0,1,1,-4,0A4,4,0,1,1,
↪4,0" fill="steelblue"/>
75     <path aria-label="December 2019: +9.7% change from prior
↪year" role="graphics-symbol" aria-roledescription="symbol mark" transform=
↪"translate(160.09289618980006,14.924455087752282)" d="M4,0A4,4,0,1,1,-4,0A4,4,0,1,
↪1,4,0" fill="steelblue"/>
76     <path aria-label="January 2020: +5.7% change from prior year
↪" role="graphics-symbol" aria-roledescription="symbol mark" transform=
↪"translate(229.2076502860901,20.695076449129235)" d="M4,0A4,4,0,1,1,-4,0A4,4,0,1,
↪1,4,0" fill="steelblue"/>
77     <path aria-label="February 2020: +13.4% change from prior
↪year" role="graphics-symbol" aria-roledescription="symbol mark" transform=
↪"translate(298.32240438238017,9.587662748507821)" d="M4,0A4,4,0,1,1,-4,0A4,4,0,1,
↪1,4,0" fill="steelblue"/>
78     <path aria-label="March 2020: 55.6% change from prior year"
↪role="graphics-symbol" aria-roledescription="symbol mark" transform=
↪"translate(362.9781420853612,109.65285552622753)" d="M4,0A4,4,0,1,1,-4,0A4,4,0,1,
↪1,4,0" fill="steelblue"/>
79     <path aria-label="April 2020: 93.6% change from prior year"
↪role="graphics-symbol" aria-roledescription="symbol mark" transform=
↪"translate(432.0000000067907,164.6837644873604)" d="M4,0A4,4,0,1,1,-4,0A4,4,0,1,1,
↪4,0" fill="steelblue"/>
80     </g>

```

Veranstaltungen

Damit unsere Veranstaltungen für alle Menschen zugänglich sind, achten wir bei der Planung und Durchführung auf Barrierefreiheit. Wir teilen unserem Publikum vorab mit, was sie erwartet und ob sie teilhaben können. Die folgenden Piktogramme helfen, die angebotene Barrierefreiheit sofort zu erkennen.

Siehe auch:

- [Leitfaden Barrierefreie Kommunikation bei Veranstaltungen \(PDF, 1,9 MB\)](#)
- [Checklistezum LeitfadenBarrierefreie Kommunikationbei Veranstaltungen \(PDF, 242 KB\)](#)

Piktogramme

Tab. 2: „Piktogramme und ihre Bedeutung“

Piktogramme	Beschreibung
Zugänglich für gehbehinderte oder auf einen Rollstuhl angewiesene Menschen	<ul style="list-style-type: none"> • Rampe mit bis zu sechs Prozent Steigung • Stufenloser Zugang mit max. 3 cm hohen Schwellen • Türbreite mindestens 90 cm
Eingeschränkte Barrierefreiheit	<ul style="list-style-type: none"> • Möblierung für gehbehinderte oder auf einen Rollstuhl angewiesene Menschen geeignet
Aufzug mit Rollstuhl nutzbar	<ul style="list-style-type: none"> • Hilfestellung wird benötigt, z.B. beim Öffnen der Türe • Zugang nicht über den Haupteingang. • Rampen steiler als 6 Prozent Steigung
Parkplatz für Behinderte	<ul style="list-style-type: none"> • Grundfläche mindestens 110 × 140 cm (B × T)
Toilette mit Rollstuhl nutzbar	<ul style="list-style-type: none"> • Reservierter Parkplatz für Berechtigte (Parkausweis)
Für blinde und sehbehinderte Menschen zugänglich	<ul style="list-style-type: none"> • Türbreite ab 90 cm • Platz seitlich vom WC mindestens 90 cm • Platz vor dem WC mindestens 150 × 150 cm
Für hörgeschädigte und gehörlose Menschen zugänglich	<ul style="list-style-type: none"> • Informationen nach dem Zwei-Sinne-Prinzip • Markierung der ersten und letzten Treppenstufen • Begleitung durch Personal möglich • Große Schrift und taktile Hinweise
	<ul style="list-style-type: none"> • Informationen und Kommunikation nach dem Zwei-Sinne-Prinzip • Induktive Höranlage vorhanden
	<ul style="list-style-type: none"> • Ansprechperson kann Gebärdensprache

6.2.2 Schreiben

Gut geschriebene Texte ermöglichen es Menschen, unsere Informationen mit Leichtigkeit zu nutzen. Diese Richtlinien wurden anhand von Beispielen aus der Praxis entwickelt und richten sich an alle, die für cusy technische Dokumentationen schreiben.

cusy-Texte sind bestens wenn

- sie schön dargestellt sind
- sie keine impliziten Annahmen treffen
- sie so einfach wie möglich sind
- sie komplex werden können, aber nicht kompliziert
- ...

Siehe auch:

- [The Zen of Python](#)
- [Material Design: Writing](#)
- [Google developer documentation style guide](#)
- [IBM Accessibility: Content design](#)

Themen

Kurz

Strebt kurze und prägnante Textbeiträge an

Fokussiert

Beschränkt euch in einem Beitrag auf ein Thema. Beiträge, die mehrere Themen umfassen, sollten aufgebrochen werden.

Zugänglich

Verwendet eine möglichst einfache Sprache ohne Fachjargon. Erklärt komplizierte Konzepte, indem ihr sie aufschlüsselt.

Verweist zu öffentlich zugänglichen Artikeln um mehr Details zu eurem Thema zu erhalten oder einen alternativen Zugang zu eurem Thema zu erhalten.

Barrierefrei schreiben

Im Abschnitt *Barrierefreiheit* findet ihr detaillierte Anleitungen zum barrierefreien Schreiben, vor allem zu den Themen:

- Textalternativen für Bild- und Audioinhalte
- Fehler und andere Meldungen
- Beschriftungen und Eingaben für Formulare

Siehe auch:

- [Web Content Accessibility Guidelines \(WCAG\)](#)

Fragen und Feedback

Wenn euch etwas fehlt oder ihr Erläuterungen zu einem bestimmten Anwendungsfall benötigt, könnt ihr uns gerne Fragen stellen in einer [GitHub-Aufgabe](#).

Einführung

Über diesen Redaktionsleitfaden

Dieser Leitfaden enthält redaktionelle Richtlinien für das Schreiben klarer und konsistenter cusy-bezogener Dokumentation.

Wenn dieser Leitfaden neu für euch ist, und ihr nach einführenden Themen zu unserem Stil sucht, dann beginnt mit den Abschnitten *Highlights*, *Stimme und Tonfall* sowie *Schreibstil*. Andernfalls könnt ihr den Leitfaden als Referenzdokument für spezifische Fragen verwenden. Zum Beispiel könnt ihr Begriffe in der *Wortliste* nachschlagen.

Redaktionelle Ressourcen

Wir empfehlen die folgenden redaktionellen Hilfsmittel in ihrer hierarchischen Reihenfolge zu verwenden:

1. Projektspezifischer Stil

Befolgt die projekt- oder produktspezifischen Stilrichtlinien, z.B. notwendige Ausnahmen von diesem Leitfaden oder Begriffe, die nur für euer spezifisches Produkt relevant sind.

2. Dieser redaktionelle Leitfaden

Wenn die projektspezifischen Stilrichtlinien keine explizite Anleitung bieten, dann folgt diesem Leitfaden.

3. Externe Referenzen

Wenn die vorangegangenen Referenzen keine explizite Anleitung bieten, dann lest diese externen Referenzen, abhängig von der Art eurer Frage:

Tab. 3: Externe Referenzen je nach Fragestellung

Art der Frage	Referenz
Rechtschreibung	Duden , s.A. (siehe auch) Rechtschreibregeln , DWDS-Wörterbuch
Nichttechnischer Stil	Universität Stuttgart: Redaktionsleitfaden und Webstyleguide ; s.A. Bundeszentrale für politische Bildung: einfach POLITIK: Lexikon
Technischer Stil	Microsoft German Style Guide ; beachtet jedoch, dass einige Richtlinien nur für Microsoft und deren Produkte zutreffend sein werden.

Andere Redaktionsleitfäden

Hier sind noch einige andere englischsprachige Redaktionsleitfäden aus der Informationstechnologie:

- [Google developer documentation style guide](#)
- [Apple Style Guide](#)
- [IBM content design](#)

Brecht die Regeln

Brecht eher eine dieser Regeln, als dass ihr etwas schreckliches sagt.

– George Orwell, «Politik und englische Sprache» (1946)

Dieser Leitfaden enthält Richtlinien, keine Regeln. Weicht davon ab, wenn euer Inhalt dadurch besser wird.

Schlüsselanweisungen

Highlights

Einführung

Der Redaktionsleitfaden deckt sehr viel Material ab, daher bietet die folgende Seite einen Überblick über die wichtigsten Punkte. Für weitere Informationen zu den Themen auf der Seite folgt den Links.

Tonfall und Inhalt

- Seid gesprächig und freundlich, ohne forsch zu sein
- Macht in der Dokumentation keine Vorankündigungen
- Verwendet beschreibenden Linktext
- Schreibt leicht zugänglich
- Schreibt für ein globales Publikum

Sprache und Grammatik

- Verwendet die zweite Person und *ihr* eher als *wir*
- Verwendet die aktive Stimme: Macht deutlich, wer die Handlung ausführt
- Setzt Konditionalsätze vor Anweisungen, nicht danach
- Informationen zur Verwendung und Schreibweise bestimmter Wörter findet .hr in der [Wortliste](#)

Formatierung, Zeichensetzung und Organisation

- Verwendet Groß- und Kleinschreibung für Dokumententitel und Zwischenüberschriften
- Verwendet nummerierte Listen für Sequenzen
- Verwendet Aufzählungslisten für die meisten anderen Listen
- Verwenden Definitionslisten für Terme und deren Beschreibung
- Verwendet fortlaufende Kommas
- Setzt Code in nichtproportionaler Schrift
- Setzt UI-Elemente in serifenloser Schrift
- Verwendet eine eindeutige Datumsformatierung

Bilder

- Verwendet SVG-Dateien oder PNG-Bilder in passender Größe
- Gebt Alternativtext an
- Stellt, wenn möglich, hochauflösende Bilder bereit

Wortliste

Bemerkung: Dieses Dokument enthält Verweise auf Begriffe, die cusy als respektlos oder beleidigend betrachtet. Die Begriffe sind hier aufgeführt, um Hinweise zu alternativen Begriffen zu geben.

Wenn der Begriff, den ihr sucht, nicht auf dieser Seite zu finden ist, schaut in *Über diesen Redaktionsleitfaden* unter *Andere Redaktionsleitfäden*.

Wortliste

#

1.2.3.4

IP-Adressen

Verwendet in Beispielkonfigurationen keine öffentlich zugänglichen IP-Adressen, die ihr nicht selbst kontrollieren könnt. Verwendet stattdessen die in [RFC 6890](#), [RFC 5737](#) und [RFC 3849](#) dokumentierten IPv4- und IPv6-Bereiche:

- 192.0.2.0/24
- 198.51.100.0/24
- 203.0.113.0/24
- 2001:db8::/32

Siehe auch *Domännennamen*.

3D

nicht 3-D

Symbole

& (et-Zeichen)

Verwendet & nicht anstelle von *und* in Überschriften oder Texten; wenn ihr auf Elementen der Benutzeroberfläche oder in Tabellenüberschriften eine Abkürzung verwenden wollt, dann nutzt *u.*; und natürlich kann das & auch in Quellcode verwendet werden.

A-Z

agnostisch

solltet ihr nicht verwenden. Nutzt stattdessen präzisere Begriffe wie *plattformunabhängig*

API

verwendet das Akronym API erst, nachdem ihr den Begriff erläutert habt, nämlich (engl. **A**pplication **P**rogramming **I**nterface) oder, noch besser, in einem <abbr>-Tag: API (Programmierschnittstelle, engl. Application Programming Interface).

Authentifizierung und Autorisierung

solltet ihr nur verwenden, wenn *authentifiziert* sich auf Personen bezieht, und *autorisiert* nur, um sich auf Berechtigungen zu beziehen, die von einer Anwendung im Namen einer authentifizierten Person gesendet werden. Eine authentifizierte Person hat einen Identitätsnachweis erbracht, z.B. durch die Eingabe eines Passworts.

best effort

solltet ihr wenn möglich vermeiden; verwendet stattdessen spezifischere Formulierungen.

Beta

geschrieben mit Großbuchstaben am Anfang.

- `PACKAGE_NAME` Beta
- `PACKAGE_NAME` ist derzeit in der Beta-Phase.

CLI

verwendet das Akronym CLI erst, nachdem ihr den Begriff erläutert habt, nämlich (engl. **C**ommand-**L**ine **I**nterface) oder, noch besser, in einem <abbr>-Tag: CLI (Kommandozeile, engl. Command-line interface).

cons

Verwendet stattdessen *Nachteile*.

erlauben

solltet ihr nicht verwenden. Nutzt stattdessen *ermöglichen*

Graylist

siehe *Schwarze Liste*, *Blacklist*, *Black-List*

Domännennamen

Verwendet keine öffentlichen Domännennamen, die ihr nicht selbst kontrollieren könnt. Verwendet stattdessen einen der verfügbaren reservierten Domännennamen, die in **RFC 2606** dokumentiert sind, wie

- `example.com`
- `example.net`
- `.example`

Hier klicken

solltet ihr nicht verwenden; für Details und Alternativen siehe *Links*.

HTTPS

nicht *HTTPs*

IPSEC (Internet Protocol Security)

nicht *IPSec* oderr *IPSEC*.

K8s

Verwendet stattdessen *Kubernetes*.

Kontoname

solltet ihr nicht verwenden; nutzt stattdessen Anmeldename

Man-Power

Vermeidet die Verwendung geschlechtsspezifischer Begriffe. Verwendet stattdessen Begriffe wie *Personal* oder *Arbeitskraft*.

Markdown

Wird immer großgeschrieben, auch wenn ihr euch auf eine nicht standardisierte Version bezieht.

Markup (Substantiv)

Ohne Bindestrich.

Master

Verwendet es niemals in Verbindung mit *slave*. Wenn möglich, ersetzt *master* durch einen spezifischen Begriff, der für den Kontext zutreffend ist, z.B. *primär*, *main*, *Original*, *Elternteil*, *Aggregator* oder *aktiv*.

Wenn ihr einen Befehl oder Code dokumentiert, der wörtlich *master* verwendet, dann verwendet dieses Wort nur in direktem Bezug auf den formatierten Code und macht deutlich, worauf ihr euch bezieht. Verwendet danach einen spezifischeren Begriff.

Siehe *Slave*.

N/A

nicht NA. Erläutert es bei der ersten Verwendung als *nicht verfügbar* oder *nicht anwendbar*.

oberhalb

verwendet es nicht, da es ein spezifisches Layout voraussetzt; verwendet stattdessen *vorausgehend*.

Pros

Verwendet stattdessen *Vorteile*.

regex

Verwendet stattdessen *regulärer Ausdruck*.

Repo

Verwendet stattdessen Repository.

Representational State Transfer

Erläutert stattdessen, wofür REST theoretisch steht, da Für Personen, die mit REST nicht vertraut sind auch die ausgeschriebenen Worte des Akronyms bedeutungslos bleiben.

RTFM

Verwendet stattdessen z.B. *Für weitere Informationen siehe . . .*

Sanity Check

Verwendet stattdessen einen spezifischeren Begriff wie *Vorprüfung* oder Kohärenzprüfung.

Schwarze Liste, Blacklist, Black-List

solltet ihr ebensowenig verwenden wie *Whitelist* und *Graylist*; verwendet stattdessen Begriffe, die für die jeweilige Fachdomäne präziser sind wie z.B. *Excludelist* oder *Blocklist*.

Bemerkung: Häufig handelt es sich nicht um Listen im eigentlichen Sinne.

- Um Anfragen von einer IP-Adresse zu verweigern, fügt sie der Datei *dos.yaml* hinzu.

- Um Anfragen von einer IP-Adresse zu verweigern, fügt sie der Denylist hinzu.
- Um eine IP-Adresse auf die schwarze Liste zu setzen, fügt sie der Datei *dos.yaml* hinzu.

Slave

Verwendet stattdessen alternative Begriffe, die für eure Domäne besser geeignet sind, z.B. *Worker* oder *Replik*. Wenn ihr die Begriffe *Master* und *Slave* gemeinsam ersetzen wollt, dann zieht Kombinationen wie *primär/sekundär*, *Original/Replik*, *Controller/Worker*, *Publisher/Subscriber* oder *aktiv/standby* in Betracht.

Wenn der Befehl oder Code, den ihr dokumentiert, wörtlich *Slave* verwendet, dann formatiert ihn als Code und macht deutlich, worauf ihr euch bezieht. Verwendet danach einen spezifischeren Begriff, z.B.

Startet den sekundären Prozess mit `slave:start`, wenn ihr Probleme zwischen dem primären und sekundären Prozess analysieren wollt.

Smartphone

Verwendet stattdessen *Mobiltelefon*.

ssh und SSH

Verwendet `ssh` wenn ihr euch auf das Terminal-Werkzeug bezieht. Verwendet hingegen `SSH` wenn ihr euch auf das Kommunikationsprotokoll bezieht.

Tarball

verwendet stattdessen `tar`-Datei.

TL;DR (Too long; didn't read)

Verwendet stattdessen etwas wie *Zusammenfassend*.

und/oder

ist manchmal die klarste und effizienteste Art, etwas auszudrücken. Dennoch solltet ihr überlegen, ob sich die Aussage umschreiben lässt ohne dass sie schwerer zu verstehen ist.

untar, unzip

verwendet stattdessen *entpacken*.

unterhalb

verwendet es nicht, da es ein spezifisches Layout voraussetzt; verwendet stattdessen *nachfolgend*.

URL, der

in Großbuchstaben; der Plural ist URLs.

v (Version)

verwendet die Kleinschreibung zur Abkürzung von Version.

via

Verwendet stattdessen *über* um Zwischenstationen zu benennen.

Vice versa

Verwendet es nicht. Verwendet stattdessen eine Phrase wie *andersherum*, *umgekehrt* oder *anders*. In manchen Kontexten ist *vice versa* unklar oder ungenau, weil in einem komplexen Satz schwer zu wissen ist, welche zwei Dinge miteinander vertauscht werden. Macht in solchen Fällen explizit klar, welche zwei Dinge vertauscht werden.

vs.

Verwendet `vs.` nicht als Abkürzung für *versus*; verwendet stattdessen das ungekürzte *versus*.

Webmaster

Verwendet stattdessen einen genaueren Begriff, um die spezifische Rolle zu beschreiben, z.B. *Website von ...*, *Website-Administration* ETC.

Web-Server

nicht Websserver

Website

nicht Web-Site

White-box

Vermeidet die Verweundung von white-box und verwendet stattdessen präzisere Begriffe wie

- introspektives Monitoring
- Clear-Box-Testing

Whitelist

siehe *Schwarze Liste, Blacklist, Black-List*

Wi-Fi

nicht wifi oder WiFi. Dies bezeichnet sowohl das Firmenkonsortium wie auch die zugehörigen Markenbegriffe; WLAN hingegen bezeichnet das Funknetzwerk.

Zugriff

Vermeidet es, wenn möglich. Verwendet stattdessen freundlichere und präzisere Wörter wie *sehen, bearbeiten, finden, verwenden* oder *ansehen*.

Stimme und Tonfall

Achtet in euren Dokumenten auf eine Stimme und einen Ton, der umgangssprachlich, freundlich und respektvoll ist, ohne kumpelhaft oder frivol zu sein; eine Stimme, die zwanglos natürlich und ansprechbar ist, nicht pedantisch oder aufdringlich. Versucht, sachkundig, freundschaftlich und verständnisvoll zu klingen.

Versucht nicht genau so zu schreiben, wie ihr sprecht; ihr sprecht wahrscheinlich umgangssprachlicher und ausführlicher, als ihr schreiben solltet, zumindest für technische Dokumentationen. Aber strebt eher einen geschäftlichen Ton an, als einen formellen.

Versucht nicht, super-unterhaltsam zu sein, aber strebt auch nicht nach völliger Versachlichung. Seid menschlich, zeigt eure Persönlichkeit, seid einprägsam; ihr könnt sogar ab und zu ein wenig witzig sein. Aber denkt daran, dass der Hauptzweck des Dokuments darin besteht, Informationen für euer Zielpublikum bereitzustellen.

Denkt daran, dass für euer Publikum nicht unbedingt Deutsch die erste Sprache gewesen sein muss, dass viele von ihnen aus anderen Kulturen kommen, und dass euer Text vielleicht in andere Sprachen übersetzt wird.

Was ihr vermeiden solltet

- Buzzwords oder Fachjargon
- niedlich oder albern sein wollen
- aktuelle popkulturelle Anspielungen
- Witze auf Kosten der Konkurrenz oder anderer Personen
- Formulierungen, die andere verunglimpfen oder beleidigen
- *Ableistische Sprache* oder Redewendungen
- das Vermischen oder Übertreiben von Metaphern
- Phrasen wie *Bitte beachtet* oder *Zu diesem Zeitpunkt*
- abgehackte oder langatmige Sätze
- alle Sätze mit der gleichen Phrase beginnen (z.B. *ihr könnt*)
- Ausrufezeichen, außer in sehr seltenen, wirklich spannenden Momenten

- Vereinnahmende Formulierungen wie *Lasst uns das machen*.
- Die Verwendung von Ausdrücken wie *einfach, es ist so einfach* oder *schnell*
- Internet-Slang oder -Abkürzungen wie TL;DR oder YMMV (englisch: Je nach dem, wie weit ihr geht, kann das variieren)

Was ihr berücksichtigen solltet

- Wenn ihr Schwierigkeiten habt, etwas auszudrücken, tretet einen Schritt zurück und fragt euch: *Was versuche ich zu sagen?*
- Wenn ihr euch bei einer Formulierung oder einem Tonfall unsicher sind, bittet ein Mitglied eures Teams, einen Blick darauf zu werfen
- Versucht, Teile eures Dokuments laut vorzulesen oder zumindest die Wörter mit dem Mund zu sprechen. Klingt es natürlich?

Beachtet jedoch, dass nicht alle Sätze natürlich klingen müssen; es sind schließlich schriftliche Dokumente. Aber wenn ihr auf einen Satz stoßt, der beim Sprechen unbeholfen oder verwirrend ist, überlegt, ob ihr ihn umgangssprachlicher formulieren könnt.

- Verwendet Übergänge zwischen den Sätzen. Phrasen wie *Obwohl* oder *So* können Absätze weniger gestelzt wirken lassen.
- Auch wenn ihr Schwierigkeiten habt, den richtigen Ton zu treffen, stellt sicher, dass ihr die nützlichen Informationen klar und direkt vermittelt; das ist der wichtigste Teil.

Seid höflich

Es ist schön, höflich zu sein, aber die Verwendung von *bitte* in einer Anweisung ist übertrieben.

- Klickt auf Ansicht, um das Dokument anzuzeigen.
- Um das Dokument anzusehen, klickt bitte auf Anzeigen.
- Weitere Informationen finden Sie unter ...
- Für weitere Informationen klickt bitte auf ...

Beispiele

Tab. 4: Beispiele für zu informelle, passende und zu formelle Sprache

zu informell	passend	zu formell
Cool! Die API ist total genial!	Mit dieser API könnt ihr die gewünschten Daten erhalten.	Diese Dokumentation beschreibt die API vollständig.
Mit diesem Aufruf erhältst du sofort jede Telefonnummer.	Ihr könnt die Telefonnummer abfragen mit <code>user.phoneNumber</code>	Für das Abfragen der Telefonnummer wird zunächst der Wert für <code>user</code> benötigt. Anschließend kann die Telefonnummer mit der Methode <code>getPhoneNumber</code> abgefragt werden.

Grundlagen

Barrierefreiheit

Textalternativen

Personen, die nicht gut sehen oder hören können, müssen sich auf Textalternativen für visuelle oder auditive Inhalte verlassen. Auf dieser Seite wird beschrieben, wie ihr Bilder, Grafiken, Audio und Video zugänglich machen könnt.

Textalternativen für Tabellen

Tabellen bieten im Wesentlichen eine Gliederung über die Anordnung der Inhalte. Daher empfehlen wir euch, Tabellen mit `<caption/>` zu beschreiben, alternativ könnt ihr jedoch auch `<table summary=""/>` verwenden.

In reStructuredText könnt ihr für `<caption/>` die `table`-Direktive verwenden, z.B.:

```
.. table:: Tabellentitel

+-----+-----+
...

```

oder:

```
.. csv-table:: Tabellentitel
```

oder:

```
.. list-table:: Tabellentitel
```

Dies führt zu folgendem HTML:

```
<table class="docutils align-default" id="id1">
  <caption>
    <span class="caption-text">
      Tabellentitel
    </span>
    <a class="headerlink" href="#id1" title="Link zu dieser Tabelle">
      ¶
    </a>
  </caption>
  ...
</table>
```

Alternative Texte für aussagekräftige Bilder und visuelle Daten

- Kommentiert dekorative Bilder

Manche Bilder enthalten keine aussagekräftigen Informationen oder Zusammenhänge sondern sind dekorativ oder dienen nur der visuellen Formatierung. Idealerweise sollten solche Bilder entfernt werden. Falls dies nicht möglich ist, solltet ihr sie als *dekorativ* kommentieren oder mit `alt = ""`.

Siehe auch:

- [An alt Decision Tree](#)
- [WAI Decorative Images](#)
- [WebAIM Decorative Images](#)

- Wenn ein aussagekräftiges Bild leicht beschrieben werden kann, gebt eine Textalternative an

Schreibt klare und prägnante Textalternativen für die verbleibenden Bilder im Inhalt.

Wenn das Bild anderes Material auf der Seite nur verstärkt, sollten nur sehr wenige Wörter verwendet werden. Viele Bilder benötigen nur ein Nomen, `alt="Portrait"` oder eine Nomen-Verb-Substantiv-Beschreibung `alt="Wahl der richtigen Datenbank"`. Bei Bildern mit aussagekräftigem Text, z.B. dem Screenshot einer Fehlermeldung, sollte die Textalternative dieselben Wörter wiederholen.

- Ist das Bild ein Steuerelement, soll die Alternative den Zweck beschreiben, nicht das Bild.

Siehe auch:

- [Functional Images](#)

- Wird der Alternativtext länger, solltet ihr andere Optionen in Betracht ziehen

Viele Browser und Leseprogramme beschränken die Zeichenlänge des alternativen Textes. In diesem Fall solltet ihr einen kurzen alternativen Text zusammen mit ausführlicheren Bildunterschriften erwägen.

Siehe auch:

- [Complex Images](#)

Namen und Symbole für Videos

- Gebt kurze Namen für Symbole ohne Beschriftungen an.

Siehe auch:

- [Icon Usability](#)

- Gebt Videos einen Titel und eine kurze Beschreibung

Siehe auch:

- [How to Write the Best YouTube Descriptions: Tips and Examples](#)
- [Providing a short text alternative that describes the purpose of live audio-only and live video-only content](#)
- [Providing a short text alternative which is the accepted name or a descriptive name of the non-text content](#)

- Untertitel hinzufügen

Stellt sicher, dass die Untertitel korrekt, verständlich und vollständig sind

Siehe auch:

- [Captions/Subtitles](#)

- 508 Accessible Videos – How to Caption Videos
- BBC Subtitle Guidelines

Medialalternativen bereitstellen

- Stellt aussagekräftige Alternativen zu visuellen Inhalten und Videos bereit.

Siehe auch:

- Providing audio that describes the important video content and describing it as such
- Understanding Success Criterion 1.2.5: Audio Description (Prerecorded)

- Stellt eine Beschreibung des sinnvollen visuellen Inhalts in einem Audiotrack oder Transkript bereit. Dies kann eine separate Audiospur mit Beschreibungen der aussagekräftigen Visuals sein oder eine Volltextalternative ähnlich einem Drehbuch.

Siehe auch:

- Transcripts
- Providing a movie with extended audio descriptions
- Providing an alternative for time based media
- Placing a link to the alternative for time-based media immediately next to the non-text content
- Providing a second, user-selectable, audio track that includes audio descriptions
- Providing an alternative for time-based media for video-only content
- Providing a version of a movie with audio descriptions
- Understanding Success Criterion 1.2.8: Media Alternative (Prerecorded)
- WCAG confusion around audio description

Gendergerechte Sprache

Mit diesen Richtlinien zu gendergerechter Sprache wollen wir möglichst präzise und frei von Diskriminierungen formulieren. Viele alltägliche Redewendungen sind leider sehr ungenau, lassen viel Interpretationsspielraum zu und können diskriminierend sein, z.B.:

- Neue Vorschriften für Software-Entwickler
- Die Fortbildung richtet sich an alle Referenten
- Suche Putzfrau für Büroreinigung
- Neue Vorschriften für die Software-Entwicklung
- Die Fortbildung richtet sich an alle Vortragenden
- Suche Personal für Büroreinigung

Bemerkung: Auch im Englischen solltet Ihr auf eine gendergerechte Sprache achten; seht hierzu z.B. Singular “their” in *Jane Austen and elsewhere*.

Wir sind uns bewusst: *Bis Gendern zur Alltagssprache gehört, dauert es lange*. Dennoch bemühen wir uns schon jetzt um eine gerechtere Sprache. So werden wir, sofern wir in unseren Texten nicht auf spezifische Geschlechter abzielen, eine genderneutrale Sprache verwenden, unter anderem:

Direkte Anrede

Häufig sprechen wir unser Publikum direkt an, z.B. mit *ihr könnt* . . .

Damit umgehen wir meist, für die gendergerechte Sprache Passivkonstruktionen zu verwenden, s.A. *Aktiv und Passiv*.

Auch substantivierte Partizipien, wie z.B. *Studierende*, können wir damit vermeiden.

Direkte persönliche Anrede

Sofern wir die Geschlechteridentität nicht kennen, wählen wir zunächst eine genderneutrale Anrede, so z.B. statt *Lieber Kai* sprechen wir die Person mit *Hallo Kai* an.

Auch bei der Anschrift in Briefen legen wir das Geschlecht nicht voreilig fest und verwenden statt *Herr Kai Mustermann* nur *Kai Mustermann*.

Konjugationen

Institutions- und Kollektivbezeichnungen lassen sich häufig durch Konjugationen ersetzen. So kann z.B. *Es waren 20 Teilnehmer* ersetzt werden durch *20 Personen haben teilgenommen*.

Partizip Perfekt

Auch mit dem Partizip Perfekt betonen wir die aktive Tätigkeit gegenüber der zugeschriebenen Rolle; z.B. statt *Herausgeber ist* . . . verwenden wir . . . *wird herausgegeben von* . . .

Adjektive

Die Beratung des Betriebsarztes kann durch *Betriebsärztliche Beratung* ersetzt werden.

Substantive mit den Endungen *-kraft, -person, -berechtigte* und *-ung*

- *Eine Fortbildung für Lehrer* kann ersetzt werden durch *eine Fortbildung für Lehrkräfte*
- *Den Wählern wurde mitgeteilt* kann ersetzt werden durch *Den Wahlberechtigten wurde mitgeteilt*
- *Der Geschäftsführer* kann ersetzt werden durch *Die Geschäftsführung*

Genderneutrale Begriffe

In anderen Fällen wählen wir genderneutrale Begriffe wie das vorher schon erwähnte *Publikum*, aber auch *Person*, *Mitglied* usw. (und so weiter). Meist finden wir auch genderneutrale Pronomen; wie z.B. folgende Ersetzungen:

- *jede* und *jeder* durch *alle*
- *keine* und *keiner* durch *niemand*
- *Redakteure, die den Kurs besucht haben* . . . durch *Diejenigen, die den Kurs besucht haben* . . . oder *Wer den Kurs besucht hat* . . . oder *Jene, die den Kurs besucht haben* . . .

Bei gendersensibler Sprache werden alle Geschlechter oder Identitäten gleichermaßen sichtbar und wertschätzend angesprochen. Wenn wir bei cusy diese Vielfalt sichtbar machen wollen, verwenden wir das Gendersternchen in geschlechtlichen Begriffen. Damit wollen wir zum Ausdruck bringen, dass es mehr als nur zwei Geschlechter gibt und die Zuordnungen soziale Konstrukte sind.

Bemerkung: Auch der DBSV (Deutsche Blinden- und Sehbehindertenverband) empfiehlt in solchen Fällen das Gendersternchen; es sei die am häufigsten verwendete Form und käme dem Wunsch nach einem Konsenszeichen am nächsten. Zudem seien Unterstrich und Doppelpunkt für sehbehinderte Menschen schlechter erkennbar.¹

Der Artikel *Genderinklusive Sprache & Barrierefreiheit*² kommt zum Schluss, dass die Unterstützung von Screenreadern für gendersensible Sprache noch zu wünschen übrig lässt, aber technisch möglich ist.

¹ Gendern mit Sternchen, Unterstrich und Doppelpunkt, März 2021

² Genderinklusive Sprache & Barrierefreiheit, 21. April 2021

Bemerkung: Die Quarks-Redaktion versucht in *Was Gendern bringt – und was nicht* die häufig emotionale Diskussion zu versachlichen und stellt einige Untersuchungsergebnisse zum Verständnis geschlechtergerechter Sprache vor.

Siehe auch:

- BFIT-Bund: Empfehlung zu gendergerechter, digital barrierefreier Sprache – eine repräsentative Studie
 - Bundesverband der Kommunikatoren: Kompendium Gendersensible Sprache. Strategien zum fairen Formulieren (2020)
 - Geschickt gendern – das Genderwörterbuch
 - Genderleicht.de
-

Schreibstil

Das Schreiben sollte immer einfach, klar und leicht verständlich sein. Die Verwendung der Alltagssprache ist im Ton freundlich, menschlich und einladend. Bevorzugt kurze Wörter vor langen, einschüchternd wirkenden Worten.

Groß-/Kleinschreibung

Verwendet Groß-/Kleinschreibung für alle Textelemente der Bedienoberfläche. Dieser Stil verwendet überwiegend Kleinbuchstaben. Lediglich der Anfangsbuchstabe des ersten Wortes im Text oder andere Worte, die großgeschrieben werden müssen, wie Eigennamen.

Das cusy-Design-System unterstützt kein Konzept *wichtiger* oder *besonderer* Wörter, da diese Bedeutung höchst subjektiv ist und schnell zu Inkonsistenzen innerhalb einer Organisation führen kann. Eine Ausnahme hiervon bilden markenrechtlich geschützte Namen.

Versalien oder Kapitälchen

Texte, die nur in Großbuchstaben oder Kapitälchen geschrieben sind, lassen sich sehr viel schwerer lesen als die Groß-/Kleinschreibung. Außerdem können Eigennamen und Substantive schwerer unterschieden werden.

Großbuchstaben

Großbuchstaben sollten verwendet werden

- wenn das Wort einen Satz oder eine Phrase beginnt
- bei Namen, z.B. *cusy Design-System*
- bei Initialismen wie z.B. BBC, HTML oder Akronymen wie NASA (National Aeronautics and Space Administration)
- bei Verweisen auf Label der Bedienoberfläche, die selbst großgeschrieben werden.

Bemerkung: Manchmal verwenden offizielle Firmen- und Produktnamen jedoch unübliche Groß-/Kleinschreibung. Wenn ihr euch auf sie beziehen wollt, solltet ihr die Schreibweise beibehalten, also z.B. bei *GitHub* und *iOS*.

Bemerkung: Mehrere Wörter eines Hashtags sollten mit Großbuchstaben gekennzeichnet werden, also z.B. #EinfacheSprache.

Einfaches Schreiben

Verwendet einfache Wörter und Sätze

- Verwendet den einfachsten Begriff, der für das Publikum geeignet ist. Verwendet beispielsweise *groß* statt *voluminös* und *klein* anstelle von *Diminutiv*.
- Seid prägnant und haltet die Sätze so kurz und einfach wie möglich. Lasst wortreiche oder überflüssige Sätze weg.
- Ermöglicht die Dinge schnell und einfach lesen zu können. Beschränkt euch auf so wenige Wörter wie möglich, aber seid nicht zu knapp.
- Vermeidet Höflichkeitsformen wie *Bitte* und *Danke* in der Bedienoberfläche, da sie in manchen kulturellen Kontexten unangemessen oder beleidigend sein könnten.

Bemerkung: Erstellt eine Liste mit Wörtern für euer Thema, die sowohl bevorzugte wie auch nicht zu verwendende Wörter enthält. Dieses einfache Werkzeug kann im Laufe der Zeit zur Konsistenz beitragen, insbesondere wenn mehrere Personen zu diesem Thema schreiben.

Verwendet die einfache Gegenwartsform

Verwendet einfache Verben und Zeitformen und haltet die Sätze prägnant, einfach, freundlich und ausdrucksvoll. Konzentriert euch auf den Kontext des Publikums und macht die Inhalte relevant.

Wenn ihr Vergangenheits- oder Zukunftsformen verwenden müsst, vermeidet Verbformen mit den Wörtern *haben*, *hat*, *hatte*, *gewesen*, *sollten*, *würden* und *werden*.

Konversationsstil

Um den passenden Ton zu finden, hilft die Vorstellung, dass mit der Software wie bei einem Gespräch interagiert wird. Das Zusammenspiel von Text, Bildern und Interaktionen bildet die abwechslungsreiche Konversation mit der Software.

Die Konversationsebene wird bestimmt durch die Aufgabe, die das Publikum zu einem bestimmten Zeitpunkt ausführt. Die meisten Konversationen durchlaufen dabei die Phasen *Entdecken*, *Ausprobieren* und *Fixieren*.

Unabhängig von der Gesprächsebene sollte das Schreiben immer einfach, klar und leicht verständlich sein. Haltet den Ton freundlich, menschlich und einladend. Verwendet Alltagssprache, keinen Jargon.

Formaler versus lässiger Ton

Während bei Geschäftskorrespondenz häufig ein formellerer Ton angebracht ist, wird bei technischen Dokumentationen und User Interfaces häufig ein lockerer Ton erwartet.

- Kontraktionen können verwendet werden, wenn sie zum Kontext passen und den Lesefluss verbessern.
- Sätze können mit *Und*, *Aber* oder *So* beginnen wenn hierdurch kürzere, prägnantere Sätze möglich sind. Diese Möglichkeit sollte jedoch nicht überbeansprucht werden.
- Verwendet Ausrufezeichen nur positiv, nicht negativ. Stellt sicher, dass ihr nicht mehr als ein Ausrufezeichen im Kontext verwendet.

Inklusive Sprache

cusy setzt sich für die Verwendung einer Sprache ein, die rassistische, kulturelle oder geschlechtsspezifische Vorurteile beseitigt.

Vermeidet Analogien, Gleichnisse und Metaphern, die stark von Demographie, Geographie, Religion oder Kultur abhängig sind.

Siehe auch:

- [macht.sprache](#).

Geschlechtergerechte Sprache

Bevorzugt Begriffe, die eine geschlechtergerechte Sprache ermöglichen, also z.B. *Person* anstatt *Benutzer*.

Siehe auch:

- [Das Genderwörterbuch](#).

Akronyme

Nicht geläufige Akronyme sollten vollständig erläutert werden bevor sie das erste Mal verwendet werden. Zudem sollten sie durch das `abbr`-Tag gekennzeichnet sein, z.B.:

Das User Interface (UI (User Interface)) erlaubt die vollständige Bedienung über Tastatureingaben.

Bemerkung: In unserem Dokumentationssystem `Sphinx` könnt ihr diesen Tag mit der `:abbr:`-Rolle hinzufügen.

Links

Links sollten Kontext für die Inhalte bieten, auf die sie verweisen. Vermeidet mehrdeutige Inhalte wie *hier klicken*.

Anmaßende Aussagen

Vermeidet wertende Begriffe wie *nur*, *einfach*, *offensichtlich* usw.

Ableistische Sprache

Die Diskriminierung von Menschen mit Behinderungen wird als Ableismus (englisch Ableism) bezeichnet. Personen dürfen nicht auf ihre Beeinträchtigung reduziert werden; auch Vorstellungen von Minderwertigkeit sind genauso wenig akzeptabel wie Grenzüberschreitungen, Einschränkungen der Autonomie und Ausschluss.

Siehe auch:

- [Autistic Hoya: Ableism/Language](#)
- [Guidelines for Writing About People With Disabilities](#)
- [Choosing Words for Talking About Disability](#)
- [Introduction to Disability Terminology](#)

Zeichensetzung

- Schreibt vollständige Sätze.
- Verwendet Ausrufezeichen sparsam.
- Vermeidet rhetorische Fragen.
- Vermeidet nachfolgende Gedanken.
- Versucht so wenig wie möglich Kommas, Bindestriche und Semikolons zu verwenden, da diese das Verständnis des Satzes erschweren.
- Vermeidet Klammern in Sätzen.

Pronomen

Verwendet so oft wie möglich die freundliche und engagierte zweite Person (*ihr*, *euer* etc.) um euer Publikum zu adressieren.

Die erste Person kann in Überschriften oder Label verwendet werden, die spezifisch für die betreffende Person sind, z.B. *Mein Konto* oder *Meine Einstellungen*. Wechselt jedoch im erklärenden Text wieder zur zweiten Person, z.B. *eure Nutzung wird am Ende des Monats berechnet*.

Vermeidet geschlechtsspezifische Pronomen in der dritten Person.

Aktiv und Passiv

Aktivsätze sind direkt und betonen das Thema. So enthält der Satz *Als Nächstes konfiguriert ihr eure Einstellungen* sehr viel direkter und auffordernder als *Als nächstes können die Einstellungen konfiguriert werden*.

Der Passivsatz dreht die Konstruktion so um, dass das Subjekt dem Verb und dem Objekt untergeordnet ist. In bestimmten Anwendungsfällen kann jedoch das Passiv natürlicher erscheinen, z.B. *Dies startet die Datenbankverbindung neu*.

Leseniveau überprüfen

Versucht, das Leseniveau der siebten Klasse nicht zu überschreiten. Vermeidet unnötigen Jargon und ähnliche Metaphern. Mit dem [Lesbarkeitsindex](#) oder für die englische Sprache mit der [Hemingway App](#) könnt ihr berechnen, wie komplex euer Text ist.

Verlinkung

Links

Um Linktext zu schreiben, verwendet beschreibende Phrasen, die Kontext für den Inhalt liefert, auf den ihr verlinkt.

Personen, die euren Text lesen, erleben die Links unterschiedlich. Einige springen von einem Link zum nächsten um relevante Quellen zu finden ohne jedoch den Text zwischen den Links zu lesen; andere erwarten, dass der Link vorher im Text beschrieben wird. Effektiver Link-Text hilft, die Zugänglichkeit und Lesbarkeit zu verbessern.

Linktext strukturieren

Um effektiven Linktext zu schreiben, verwendet eine der folgenden Formen:

- Sorgt dafür, dass der Linktext genau dem Text des Titels entspricht, auf den er verweist.
 - Weitere Informationen findet ihr unter *Schreibstil*.

Informationen zur Großschreibung solcher Verweise findet ihr unter Großschreibung in Verweisen auf Titel und Überschriften.

- Beschreibt die Zielseite im Linktext so, als ob sie Teil des Satzes wäre.
 - Ihr könnt *Sprachstil überprüfen* verwenden um eure Dokumentation automatisch überprüfen zu lassen.

Manchmal müsst ihr einen Satz überarbeiten, um eine Phrase einzuschließen, die einen guten Link-Text ergibt.

Schreiben von Linktext

Wendet die folgenden Richtlinien an, wenn ihr Linktext schreiben:

- Wenn ihr einen vollständigen Satz schreibt, der euer Publikum auf ein anderes Thema verweist, leitet den Link mit der Phrase *Für weitere Informationen, siehe* oder *Für weitere Informationen über ... , siehe* ein.
- Schreibt eindeutigen, beschreibenden Linktext, der ohne den umgebenden Text Sinn ergibt. Verwendet keine Phrasen wie *hier klicken* oder *dieses Dokument*; sie erschweren die Zugänglichkeit.

Interpunktion bei Links

Wenn ihr eine Interpunktion unmittelbar vor oder nach einem Link habt, setzt die Interpunktion nach Möglichkeit außerhalb der Link-Tags.

Gestaltung von Linktext

Wendet die übliche Gestaltung auf Linktext an; dies hilft eurem Publikum, Links in eurem Inhalt zu finden.

Kontrastiert die Farbe des Linktextes mit der Farbe des normalen Textes. Um eurem Publikum das Erkennen von Links zu erleichtern, sollte sich der Linktext vom restlichen Text auf der Seite abheben.

Unterstreicht Link-Text. Die horizontale Linie zeigt sehr klar an, wo ein Link endet und der nächste beginnt. Dies erleichtert das Scannen der Website und hilft, Links zu finden.

Besuchte Links sollten die Farbe wechseln. Verwendet farbenblind-freundliche Farbwechsel, um Links, denen bereits gefolgt wurde, von noch nicht besuchten Links zu unterscheiden. Dies erleichtert eurem Publikum, effektiv durch eure Website zu navigieren, ohne dass es Inhalte, die es bereits gelesen hat, erneut aufrufen zu müssen.

Verhalten von Links

Ihr könnt Angaben machen, ob ein Link in einem neuen Fenster geöffnet werden soll oder nicht. Personen, die euren Text lesen, werden das jeweilige Standardverhalten ihres Browsers erwarten. Wenn ihr also festlegen wollt, dass ein Link in einem neuen Fenster oder Tab geöffnet werden soll, solltet ihr zumindest vorher auf dieses Verhalten hinweisen, z.B. mit einem nachfolgenden Icon:

Links zu E-Mail-Adressen oder Telefonnummern

Beim Klicken auf einen E-Mail-Link öffnet sich der E-Mail-Client der jeweiligen Person, damit diese sofort eine E-Mail schreiben kann. Hierfür könnt ihr statt einer Webadresse z.B. einfach `mailto:info@cusy.io` verwenden. Auch Betreff, CC, BCC und Haupttext können in einem solchen Link angegeben werden:

```
<a href="mailto:info@cusy.io?cc=info@cusy.io, support@cusy.io&subject=cusy%20Support&body=Hier%20kann%der%20Haupttext%20stehen">
  ↳ Schreibt uns!
</a>
```

Ein Link kann auch automatisch eine Telefonnummer wählen, z.B.:

```
<style>
  a[href^="tel:"]:before {
    content: "\260e";
    margin-right: 0.5em;
  }
</style>

<a href="tel:+493022430082">+49 30 22430082</a>
```

Siehe auch:

- [The Current State of Telephone Links](#)

Links zu Dateien

- [WebAIM Quick Reference](#)
- [WebAIM Quick Reference \(PDF, 327 KB\)](#)

Der erste Link sieht aus wie ein normaler Link, und diejenige Person, die auf den Link klickt, wird vermutlich erwarten, dass eine neue Seite im Browser geöffnet wird. Der zweite Link informiert vorab, dass es sich beim Linkziel um eine PDF-Datei handelt und über die Dateigröße dieses PDF. Das Verhalten ist im zweiten Fall sehr viel weniger überraschend.

Computer-Schnittstellen

API-Referenz

Wenn ihr eine API dokumentiert, stellt eine vollständige API-Referenz bereit, die in der Regel aus dem Quellcode unter Verwendung von Kommentaren generiert wird und alle öffentlichen Klassen, Methoden, Konstanten ETC. beschreibt.

Verwendet die grundlegenden Richtlinien in diesem Dokument, wie es für eine bestimmte Programmiersprache angemessen ist. In diesem Dokument wird nicht angegeben, wie die Kommentare formatiert sein sollen; weitere Informationen findet ihr in den spezifischen Richtlinien für jede Programmiersprache.

Siehe auch:

- [Docstrings](#)

Diese Seite deckt auch keine Web-APIs ab. Die später aufgeführten Stilvorschläge können bei der Dokumentation von Web-APIs nützlich sein.

Siehe auch:

- [FastAPI](#)

Grundlagen

Die API-Referenz muss für jeden der folgenden Punkte eine Beschreibung enthalten:

- jede Klasse, Schnittstelle, Struktur und jede andere Komponente der API (z.B. Union-Typen in C++)
- jede Konstante, jedes Feld, jedes Enum, jede Typedef, usw. (und so weiter)
- jede Methode, mit einer Beschreibung für jeden Parameter, den Rückgabewert und alle möglichen Exceptions (engl.: Ausnahmen)

Veralteter Code

Wenn Code veraltet ist, sagt, was alternativ verwendet werden soll. Und wenn ihr eure API versioniert, erwähnt, in welcher Version es zuerst veraltet war.

Nur der erste Satz einer Beschreibung erscheint in der Zusammenfassung und im Index, also fügt die wichtigsten Informationen dort ein. Die folgenden Sätze können erklären, warum die Methode veraltet ist, zusammen mit allen anderen Informationen, die für die weitere Entwicklung und Nutzung der API nützlich sind.

Wenn eine Methode veraltet ist, sagt eurem Publikum, was es tun muss, damit der Code weiterhin funktioniert.

Siehe auch:

- [Veralteter Code](#)

Code

Verwendet in normalen Textsätzen (im Gegensatz zu z.B. Codebeispielen) eine nicht-proportionale Schriftart, um Code auszuzeichnen.

- verwendet in HTML das `<code>`-Element.
- in Markdown einen Backtick ``` vor und nach dem Code
- in reStructuredText zwei Backticks ```` davor und danach

Diese Seite erklärt, wie ihr Code in normalen Textsätzen formatiert.

Einige spezifische Code-Elemente

Die folgende Liste enthält Elemente, die als Code dargestellt werden sollten; sie ist jedoch nicht vollständig.

- Attributnamen und -werte
- Klassennamen
- Datentypen
- Konstanten
- Namen von Umgebungsvariablen
- Schlüsselwörter der Programmiersprache
- Methoden- und Funktionsnamen
- Namespace-Aliase
- Namen von Befehlszeilenprogrammen
- Elementnamen (XML und HTML)
Setzt spitze Klammern (`<>`) um den Elementnamen.
- HTTP-Statuscodes und -Content-Types
- DNS Record-Typen
- Dateinamen, Dateinamenerweiterungen (falls verwendet) und Pfade
- Ordner und Verzeichnisse

Bemerkung: Generell solltet ihr keinen Code in Anführungszeichen setzen, es sei denn, die Anführungszeichen sind Teil des Codes.

Sprachunterstützung in Sphinx

Sphinx unterstützt verschiedene Sprachen, u.A. (unter anderem)

- Python
- C
- C++
- JavaScript

- Math
- MATLAB
- Scala

Siehe auch:

- Code-Blöcke

Elemente, die normal dargestellt werden sollen

Die folgende Liste enthält Elemente, die nicht als Code dargestellt werden sollten; die Liste ist jedoch nicht vollständig.

- E-Mail-Adressen
- Namen von Produkten, Dienstleistungen und Organisationen

Oft werden die Namen von Befehlszeilen-Dienstprogrammen genauso geschrieben wie das Software-Projekt oder -Produkt, mit dem sie verbunden sind. Verwendet in solchen Fällen Code für den Befehl und die normale Schrift für den Namen des Projekts oder Produkts.

Beispiel: Ruft den GCC-Compiler mit `gcc` auf.

- URLs

Siehe auch:

- *Links*
- *Platzhalter*

Platzhalter

In diesem Abschnitt wird erklärt, wie Platzhaltervariablen in Befehlen und Codebeispielen formatiert werden soll.

Platzhalter in Beispielcode und Befehlen stellen Werte dar, die beim Verwenden des Codes ersetzt werden müssen. Platzhalter in Beispielausgaben können die Werte auch variieren. Im Allgemeinen hat ein Platzhalter einen beschreibenden Namen als Standardwert.

Der Platzhalter `PROJECT_ID` steht für eine beispielhafte Projektbezeichnung, die mit `PROJECT_` beginnt, gefolgt von einer variablen ID.

Beim Platzhalter `HTTP_RESPONSE_CODE` wird signalisiert, dass ein HTTP-Antwortcode zwischen 100 und 599 erwartet wird.

Text für Platzhaltervariablen

- Verwendet Großbuchstaben mit Unterstrich als Trennzeichen.
 - `https://cusy.io/LANGUAGE_CODE/`
 - `https://cusy.io/LANGUAGE_CODE/`
 - `https://cusy.io/language-code/`
 - `https://cusy.io/LanguageCode/`
- Wenn der Kontext, in dem ihr Platzhaltervariablen verwenden wollt, Großbuchstaben und Unterstriche missverständlich sein können, dann verwendet etwas anderes, aber seid intern konsistent.

- Verwendet keine Possessivpronomen in Platzhaltern.
 - `https://cusy.io/MY_LANGUAGE/`
 - `https://cusy.io/YOUR_LANGUAGE/`

Erläutert Platzhalter

Wenn ihr einen Platzhalter in Text oder Code verwendet, fügt bei der ersten Verwendung eine Erklärung für den Platzhalter ein. Wenn nach der Verwendung dieses Platzhalters noch weitere Schritte und andere Platzhalter folgen, könnt ihr sie auch alle auf einmal erklären.

Verwendet bei Prozeduren die folgende Reihenfolge:

1. Erklärt, was gemacht werden soll
2. Nennt den Befehl
3. Erläutert den Platzhalter
4. Erläutert den Befehl bei Bedarf ausführlicher
5. Führt die Ausgabe auf, falls erforderlich
6. Erläutert ggf. die Ausgabe

-

1. Damit euch die Spack-Umgebung zur Verfügung steht, müsst ihr folgendes in die `~/.bashrc` oder `~/.profile` eingetragen:

```
export SPACK_ROOT=~/.spack
. $SPACK_ROOT/share/spack/setup-env.sh
```

2. `SPACK_ROOT` gibt dabei den Pfad zu eurer Spack-Installation an.
 - . definiert den Pfad zur Datei `setup-env.sh`, wodurch ihr zukünftig die Spack-Befehle direkt in eurer Konsole aufrufen könnt.
3. Die geänderte Konfiguration wird nun übernommen mit `source`, z.B. mit:

```
$ source ~/.bashrc
```

4. Schließlich könnt ihr eure Shell-Konfiguration überprüfen, z.B.:

```
$ spack compiler find
==> Found no new compilers
==> Compilers are defined in the following files:
    /Users/veit/.spack/darwin/compilers.yaml
```

5. `spack` konnte direkt im Pfad aufgerufen werden und auch die Direktive `compiler` war verfügbar. Damit sind die Pfadangaben in der Shell-Konfiguration richtig und wirksam.

Siehe auch:

- [Platzhalter](#)

UI-Elemente und Interaktionen

Konzentriert euch auf die Aufgabe

Wenn es sinnvoll ist, solltet ihr Anweisungen in Bezug auf das formulieren, was ihr erreichen wollt. Indem ihr den Verweis auf UI-Elemente vermeidet, helft ihr, den Zweck einer Anweisung zu verstehen, und tragt dazu bei, Abläufe effektiver zu machen.

Allerdings solltet ihr die Zielgruppe kennen und den Kontext verstehen. Zudem kann es auch sinnvoll sein, zunächst alle Elemente einer Seite vorzustellen.

- Klickt auf *Aktualisieren*
- Klickt auf die Schaltfläche *Aktualisieren*
- Wählt *Erweiterte Optionen*
- Wählt *Einstellungen* → *Erweiterte Optionen*

Formatieren von UI-Elementen

Wenn ihr auf den Namen eines beliebigen UI-Elements verweisen wollt, setzt den Namen in serifenloser Schrift. Dazu gehören Namen für Schaltflächen, Menüs, Dialoge, Fenster, Listenelemente oder jede andere Funktion auf der Seite, die einen sichtbaren Namen hat. Verwendet keine Code-Schrift für UI-Elemente, es sei denn, es handelt sich um ein Element, das die Anforderungen für Code erfüllt.

Folgt in den meisten Fällen der Groß- und Kleinschreibung, wie sie auch auf der Seite erscheint. Wenn die Beschriftungen jedoch inkonsistent sind oder nur in Großbuchstaben geschrieben werden, verwendet die übliche Schreibweise im Satz. Verwendet UI-Elemente jedoch nicht so, als wären sie Verben oder Substantive im Satz.

- Aktiviert im Fenster *Neues Projekt* das Kontrollkästchen *Neue Aktivität* und klickt dann auf *Weiter*
- Aktiviert im Fenster »Neues Projekt« das Kontrollkästchen »Neue Aktivität« und klickt dann auf die Schaltfläche »Weiter«
- Klickt auf *Aktualisieren*
- Klickt auf *AKTUALISIEREN*
- Gebt im Feld *Name* einen Kontonamen ein
- Gebt den *Namen* für das Konto ein
- Um die Einstellungen zu speichern, klickt auf *Speichern*
- *Speichern* Sie die Einstellungen
- Gebt im Feld *Servicekonto-ID* einen Namen ein
- Gebt eine *Servicekonto-ID* ein
- Gebt für *Servicekonto-ID* einen Namen an

Sphinx

Siehe auch:

- [UI-Elemente und Interaktionen](#)

Testen

Testen

Durch das Testen eurer Inhalte könnt ihr sicherstellen, dass sie sich in einem konsistenten Zustand befinden. Die Erfahrung beim Lesen wird besser und problemloser.

Continuous Integration

Wir überprüfen systematisch, ob die Dokumentation des cusy Design-System erstellt werden kann und zeigen euch dies auch mit folgendem Abzeichen (ENGL. (englisch): Badge) an: .

Dieses Abzeichen wird vom Dokumentationsserver [Read the Docs](#), auf dem unsere Dokumentation des cusy Design-System veröffentlicht wird, bereitgestellt.

Weitere automatisch generierte Abzeichen findet ihr auf unserer Seite [cusy Design-System](#) mit Hinweisen zur Anzahl der Mitwirkenden, zur Aktualisierungshäufigkeit, zur Lizenz und zu Mastodon.

Solche Abzeichen lassen sich jedoch nicht nur verwenden um den Status eures Projekts zu dokumentieren, sondern auch um einen stets aktuellen Überblick über viele verschiedene Projekte zu erhalten. Ein Beispiel hierfür findet ihr in unserem [PyViz-Tutorial](#).

Sphinx-Checks

Siehe auch:

- [Build-Fehler](#)
- [Links überprüfen](#)

Bemerkung: Falls ihr ein anderes Werkzeug zum technischen Schreiben verwendet, könnt ihr z.B. [HTMLProofer](#) verwenden um Links, Bilder, Titel und Tags zu überprüfen.

Code überprüfen

Ihr könnt automatisch euren Quellcode überprüfen und ggf. auch neu formatieren lassen. Einen Überblick über solche Werkzeuge erhaltet ihr in unserem Jupyter-Tutorial unter [Checker](#) und [Formatter](#).

Syntax überprüfen

Es gibt auch Werkzeuge, die eure Inhalte anhand von Regeln überprüfen.

Siehe auch:

- [Regulating English with retext-mapbox-standard](#)
- [Writing automated tests for your documentation](#)

Sprachstil überprüfen

Prosa-Linters wie z.B. [Vale](#) gehen weit über Rechtschreib- und Grammatikprüfungen hinaus und betrachten auch den Sprachgebrauch: Wiederholt sich das Gesagte? Ist die Sprache zu informell? Ist die Ansprache inkonsistent? Werden unerwünschte Klischees bedient? Oder ist die Sprache sexistisch?

Vale wird von vielen Open-Source-Projekten genutzt, u.A. von

- [GitLab](#) ([.vale.ini](#), [Regeln](#))
- [Homebrew](#) ([.vale.ini](#), [Regeln](#))

Mit Vale selbst kommen die folgenden Stile mit:

Microsoft

Eine Implementierung des [Microsoft Writing Style Guide](#).

Google

Eine Implementierung des [Styleguides für den Google developer documentation style guide](#).

write-good

Eine Umsetzung der vom [write-good-Linter](#) erzwungenen Richtlinien.

proselint

Eine Umsetzung der vom [proselint-Linter](#) erzwungenen Richtlinien.

Joblint

Eine Umsetzung der vom [Joblint-Linter](#) erzwungenen Richtlinien.

Ihr könnt Vale installieren mit

```
$ brew install vale
```

Wenn ihr als Dokumentationswerkzeug Sphinx nutzt, solltet ihr noch den `rst2html`-Parser installieren mit

```
$ brew install docutils
```

Nun könnt ihr Vale konfigurieren in `.vale.ini`:

```
StylesPath = ./styles
MinAlertLevel = suggestion

[*.{md,rst}]
BasedOnStyles = mystyles

vale.Redundancy = YES
vale.Repetition = YES
vale.GenderBias = YES
```

Anschließend werden die Stile definiert, u.A. in `styles/MY_STYLE/Polite.yml`, z.B. mit:

```
extends: existence
message: 'Do not use "%s" in technical documentation.'
level: error
ignorecase: true
tokens:
  - please
  - thank you
```

Und schließlich könnt ihr eure Dokumentation überprüfen mit:

```
$ vale docs/
✓ 0 errors, 0 warnings and 0 suggestions in 201 files.
```

Vale liefert nur ein Wörterbuch für **amerikanisches-Englisch** mit. Ihr könnt jedoch auch andere Wörterbücher hinzufügen, z.B. aus github.com/freedesktop/libreoffice-dictionaries. Diese könnt ihr einbinden indem ihr sie lokal zur Verfügung stellt, z.B. unter `~/Library/Spelling/de_DE.dic,aff` und sie anschließend in eurem Projekt in der Datei `styles/MY_STYLE/Spelling.yml` referenziert:

```
extends: spelling
message: "Rechtschreibprüfung: '%s'"
dicpath: ~/Library/Spelling
dictionaries:
  - de_DE_frami
level: warning
ignore: styles/cusy/ignore.txt
```

Bemerkung: Wenn ihr den Inhalt eures GitHub-Repository mit Vale überprüfen wollt, könnt ihr eine GitHub-Action hierfür einrichten: [vale-action](#).

Üblicherweise werden *literal blocks*, *inline literals* und *code-blocks* ignoriert. Ihr könnt jedoch auch Bereiche aus der Überprüfung herausnehmen mit:

```
.. vale off

Text, der nicht überprüft werden soll.

.. vale on
```

Siehe auch:

- [Docs](#)
- [GitHub](#)
- [check package](#)

6.2.3 Design-Themen

Mithilfe von Design-Themen können vorhandene Komponenten an einen bestimmten visuellen Stil angepasst werden. Dabei werden im cusy Design-System meist nur eine Reihe universeller Variablen geändert und die meisten Komponenten passen sich dann automatisch an. Lediglich in Ausnahmefällen müssen einzelne Komponenten geändert werden.

Das cusy Design-System bietet zwei verschiedene Design-Themen, wie auf der Seite [Farbpaletten](#) gezeigt: *Dark* und *Light*. Beide Themen können in CSS angegeben werden mit:

```
:root {
  color-scheme: light dark;
}
```

Um nicht die Farbe jedes Elements im DOM überschreiben zu müssen, sollten die CSS-Anweisungen bei benutzerdefinierten Eigenschaften oder Variablen beginnen:

```
:root {
  --background-color: #161616;
  --text-color: #f4f4f4;
  --link-color: #78a9ff;
}
body {
  background-color: var(--background-color);
  color: var(--text-color);
}
a {
  color: var(--link-color);
}
```

Das *Light*-Thema wird dann in einer `@media`-Anweisung definiert:

```
@media (prefers-color-scheme: light) {
  :root {
    --background-color: #ffffff;
    --text-color: #161616;
    --link-color: #0f62fe;
  }
}
```

Dabei ist jedoch zu beachten, dass Grafiken und Bilder auf unterschiedlichen Hintergründen meist anders dargestellt werden sollten. Für SVG-Dateien können die Farben einfach überschrieben werden, z.B. mit:

```
svg.activity-sparkline path {
  stroke: var(--text-color);
}
```

Bei Fotos genügt im allgemeinen, sie etwas aufzuhellen und den Kontrast zu reduzieren:

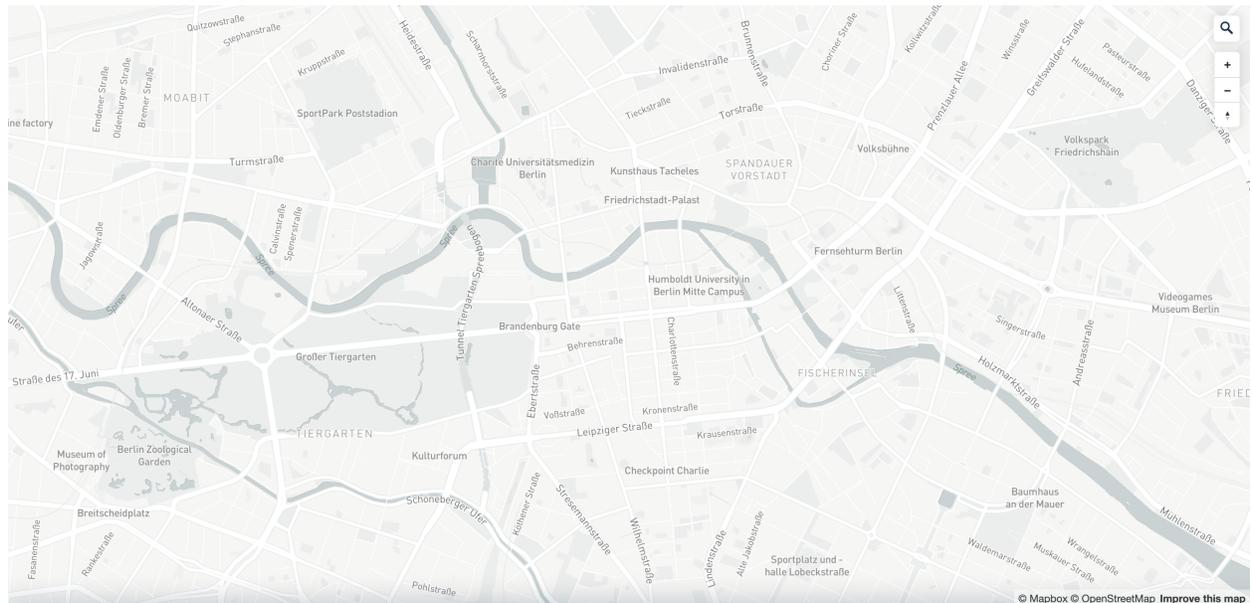
```
@media (prefers-color-scheme: light) {
  img {
    filter: brightness(1.24) contrast(0.81);
  }
}
```

Manchmal lassen sich bestehende Bilder jedoch nicht einfach optimieren und sollten dann durch andere Bilder ersetzt werden:

```
<picture>
  <source
    srcset="light.png"
    media="(prefers-color-scheme: light)"></source>
  
</picture>
```

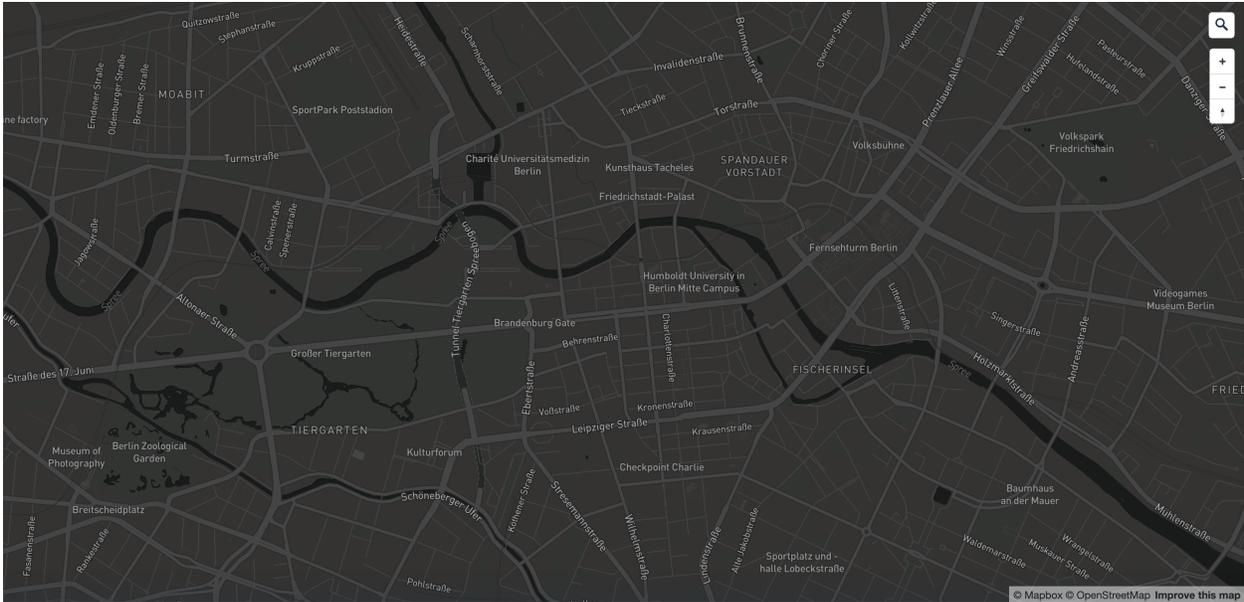
Schließlich stehen auch für die Darstellung von mapbox-Karten ein **Dark Mode** und ein **Light Mode** zur Verfügung. Diese können verwendet werden mit:

```
<picture>
  <source
    media="(prefers-color-scheme: light)"
    srcset="https://api.mapbox.com/styles/v1/mapbox/light-v10/static...">
  
</picture>
```



Siehe auch:

- [Mozilla web docs: Using CSS custom properties \(variables\)](#)
- [Mozilla web docs: prefers-color-scheme](#)
- [Material System: Dark theme](#)



6.2.4 Typographie

Typographie kann dabei helfen, Texte zu organisieren und Personen durch eine Seite zu führen. Dabei wird der geeignete Schriftstil durch den Bereich im Layout bestimmt.

Schrift

Das cusy Design-System verwendet die Open-Source-Schrift **M+ 2c**. Sie kann aus dem [M+ Fonts-Repository](#) heruntergeladen werden.

Sans-serif

```
font-family: 'M PLUS 1p', sans-serif;
```

Mono

```
font-family: 'M PLUS 1nm', monospace;
```

Skalierung

Die cusy Schriftskala basiert auf einerseits auf dem Goldenen Schnitt, andererseits auf der Verdoppelung, um die Texte hierarchisch zu strukturieren:

Tab. 5: cusy Schriftskala

rem	px
0.625	10
0.813	13
1	16
1.313	21
1.625	26

Gewichte

Die Schriftgröße wird im cusy Design-System für die hierarchische Gliederung der Texte verwendet. Eine fettere Schrift innerhalb eines Textes betont jedoch diesen Textabschnitt.

Die Wirkung der Schriftstärke unterscheidet sich jedoch zwischen dem cusy Dark-Theme und dem cusy Light-Theme:

Tab. 6: Schriftstärke in Abhängigkeit von Textabschnitt, Dark- und Light-Theme

Format	cusy Dark-Theme	cusy Light-Theme
Überschrift	Bold	Medium
Unterüberschrift	Bold	Medium
Zusammenfassung	Regular	Light
Textkörper	Regular	Light
Fußnoten, Marginalien	Regular	Light

Kursiv

Es gibt keine Kursive. Dabei werden die Titel von Werken, Fachbegriffe und Namen unterschiedlich kenntlich gemacht:

- Titel von Werken werden in Anführungszeichen gesetzt, z.B. das »PyViz-Tutorial«.
- Für Fachbegriffe wird bei Mouseover ein Tooltip mit dem Beginn der Kurzbeschreibung angezeigt.
- Namen werden nur mit ®, TM o.Ä. (oder ähnlich) ausgezeichnet, sofern zutreffend.

Textfarbe

Überschriften werden in der cusy Hausfarbe Purple `rgba(132, 14, 255, 1)` gestaltet.

Text soll neutral erscheinen, also * White* im cusy Dark-Theme und *Black* im cusy Light-Theme.

Links erhalten die Farbe *Blue*.

Syntax-Hervorhebungen in Quellcode erfolgen mit [Highlight.js plugin for Vue.js](#) und unterscheiden zwischen cusy Dark-Theme und cusy Light-Theme.

Weitere Farben dürfen nur verwendet werden, sofern sie bedeutend sind, z.B.

- *Red* für Fehler
- *Orange* für Warnungen
- *Blue* für Informationen
- *Green* für Bestätigungen

6.2.5 Datenvisualisierung

Mithilfe von cusy-Charts könnt ihr genaue und überzeugende Geschichten rund um Daten mit schönen und zugänglichen Visualisierungen erzählen.

Diagrammtypen

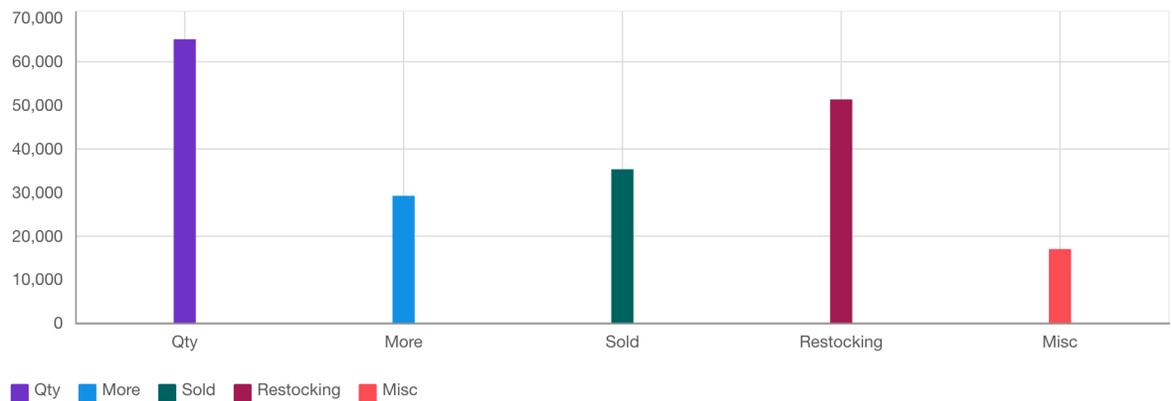
Bestimmt zunächst den Zweck der Visualisierung und wählt dann den geeigneten Diagrammtyp.

Siehe auch:

- [Visual Vocabulary – Designing with data](#)

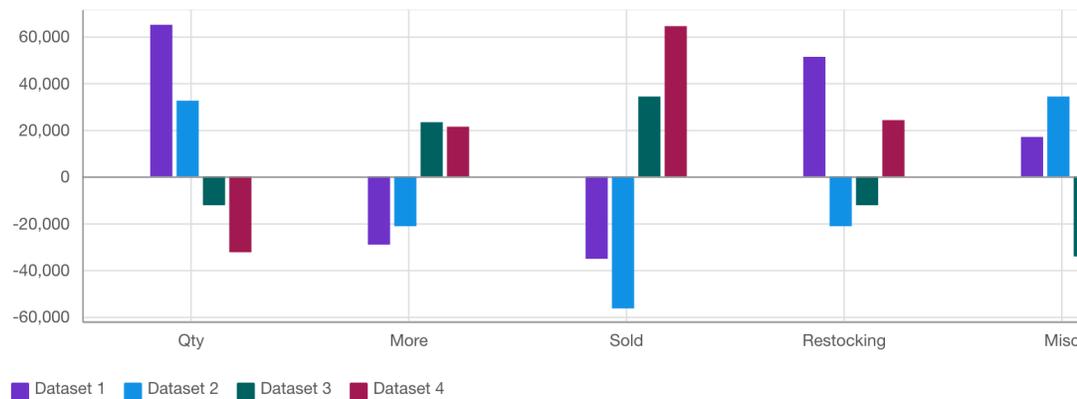
Vergleiche

Simple bar (discrete)



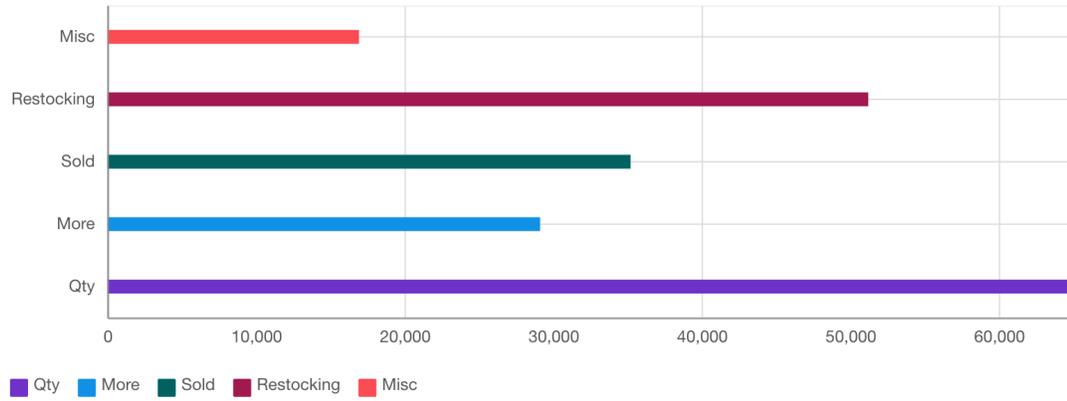
Balkendiagramm

Grouped bar (discrete)



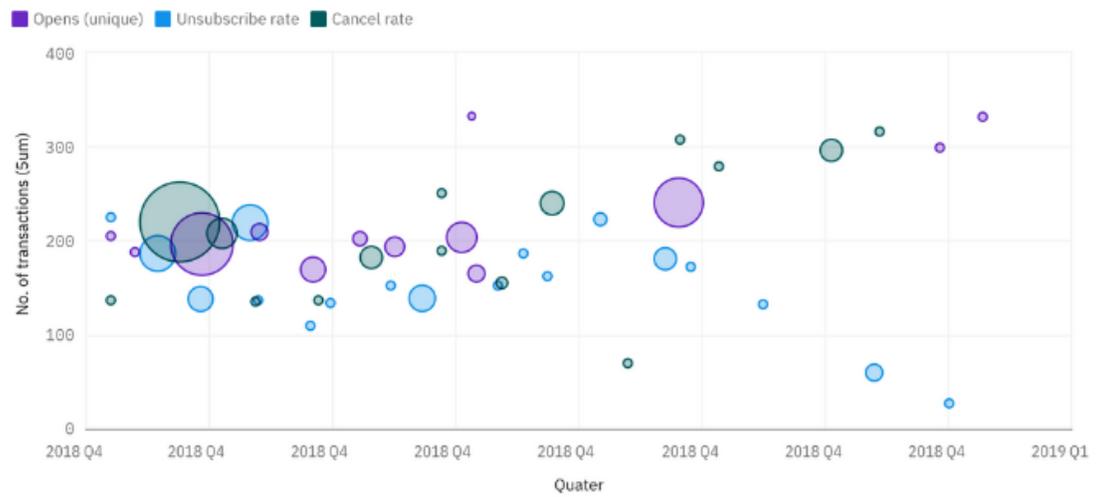
Gruppiertes Balkendiagramm

Simple horizontal bar (discrete)



Horizontales Balkendiagramm

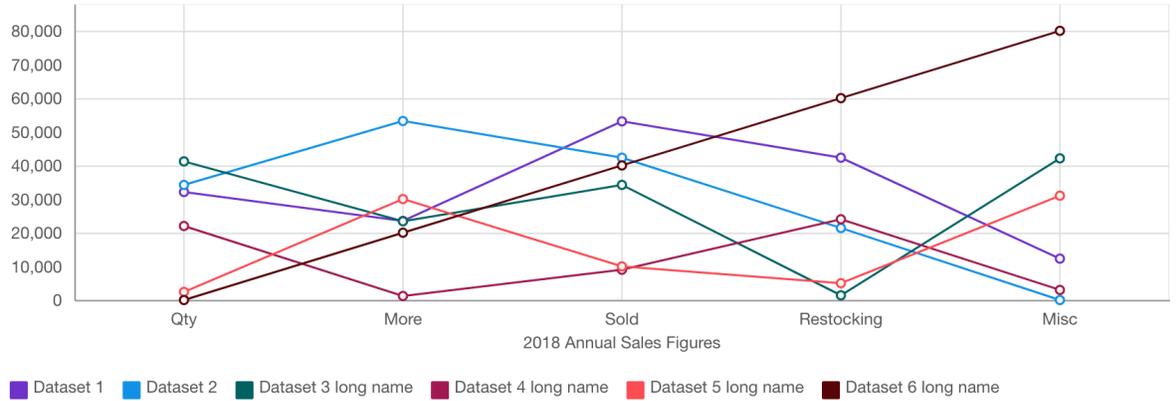
Sales by product



Blasendiagramm

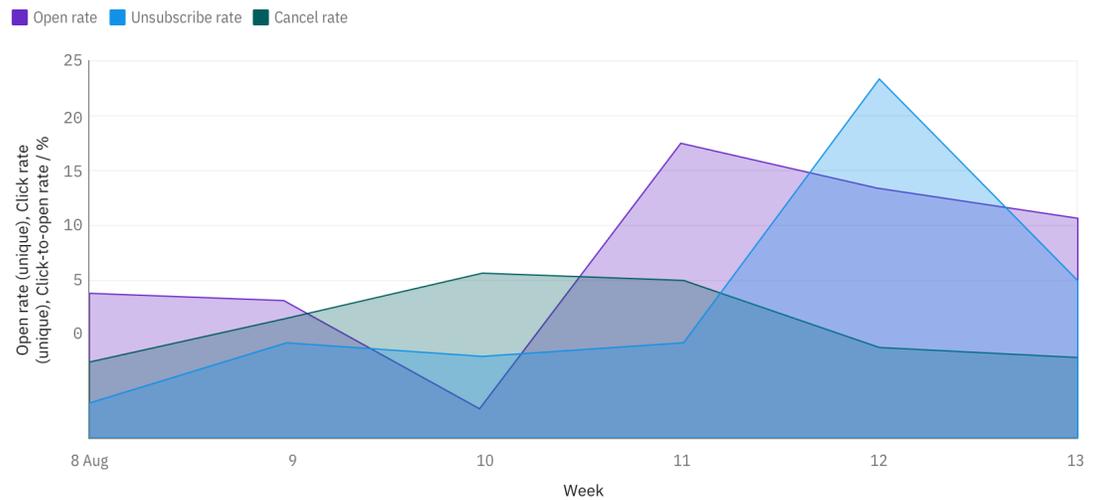
Trends

Line (discrete)



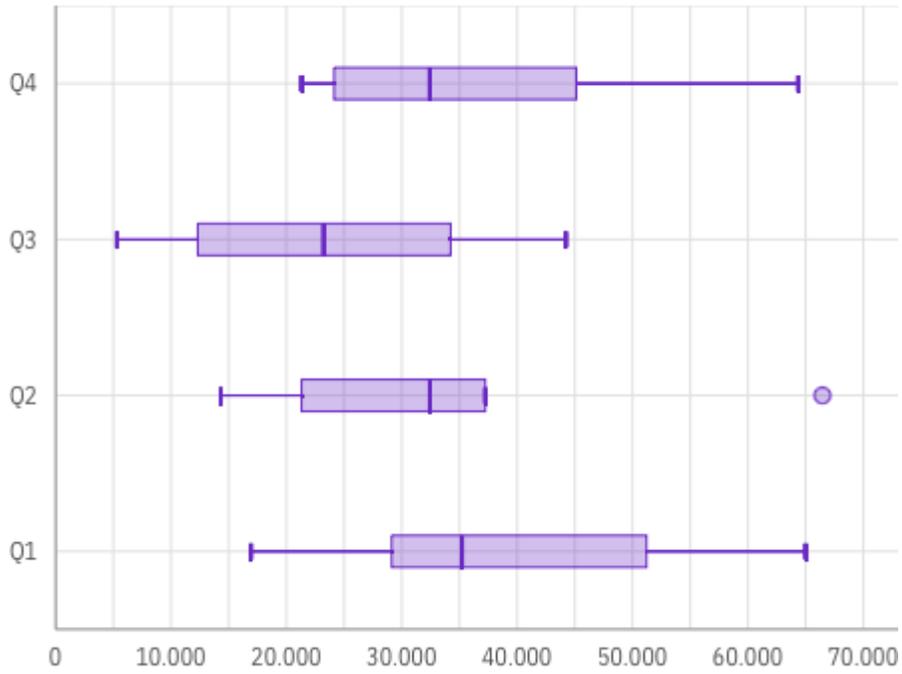
Liniendiagramm

Optimization rates (overall)



Flächendiagramm

Horizontal box plot



Boxplot

Teil-zu-Ganzes

Donut



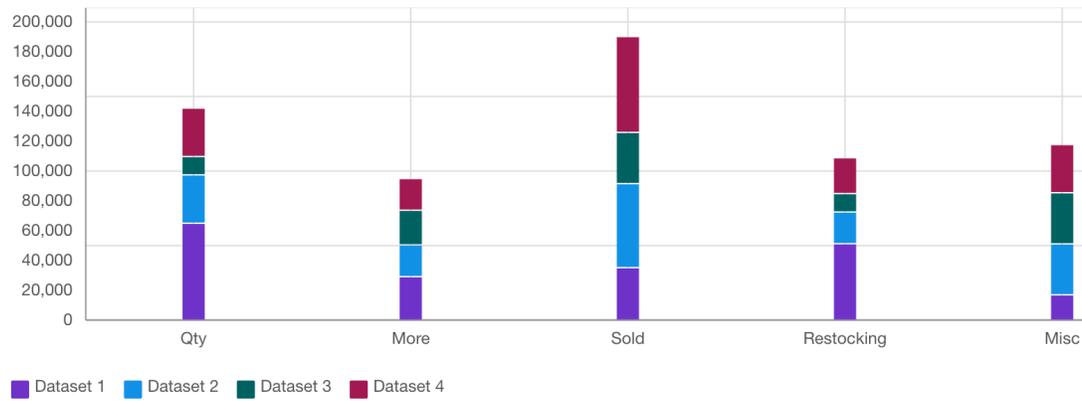
Ringdiagramm

Pie



Kreisdiagramm

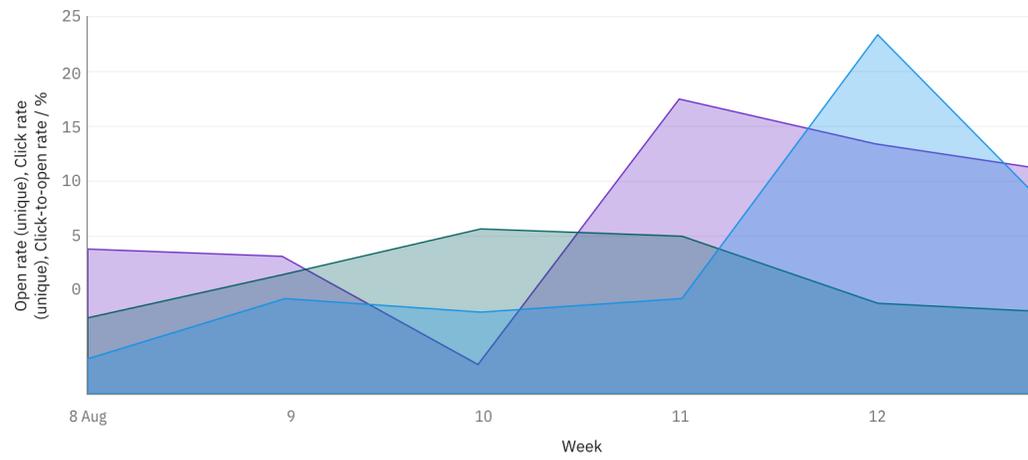
Stacked bar (discrete)



Gestapeltes Balkendiagramm

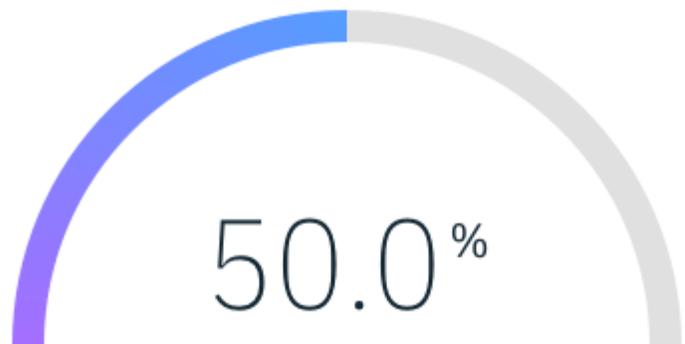
Optimization rates (overall)

Open rate Unsubscribe rate Cancel rate



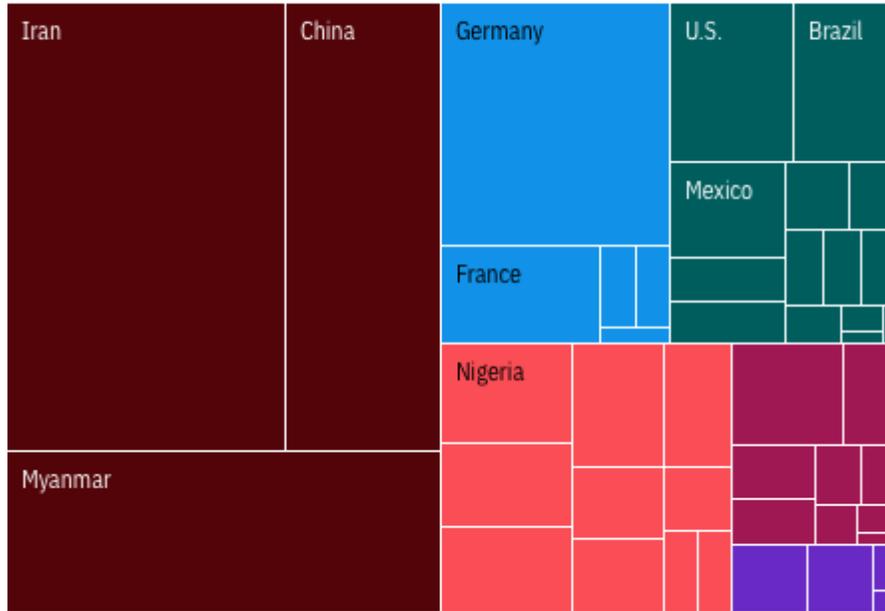
Gestapeltes Flächendiagramm

Memory utilization



Halbkreisförmiges Ringdiagramm

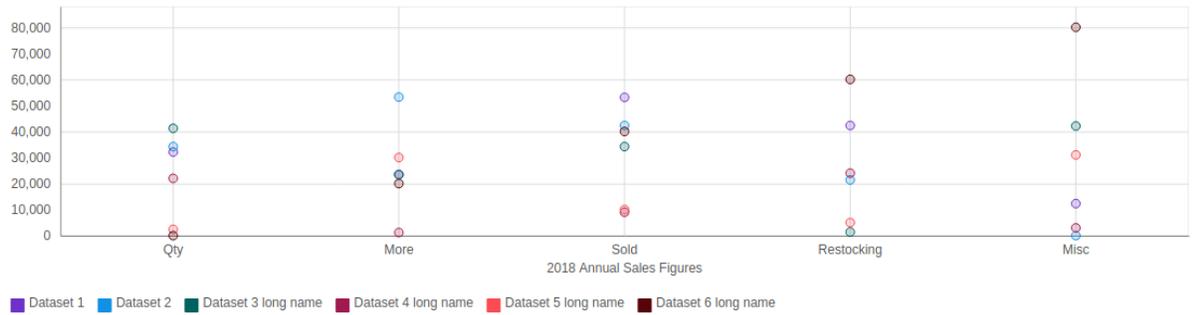
Treemap



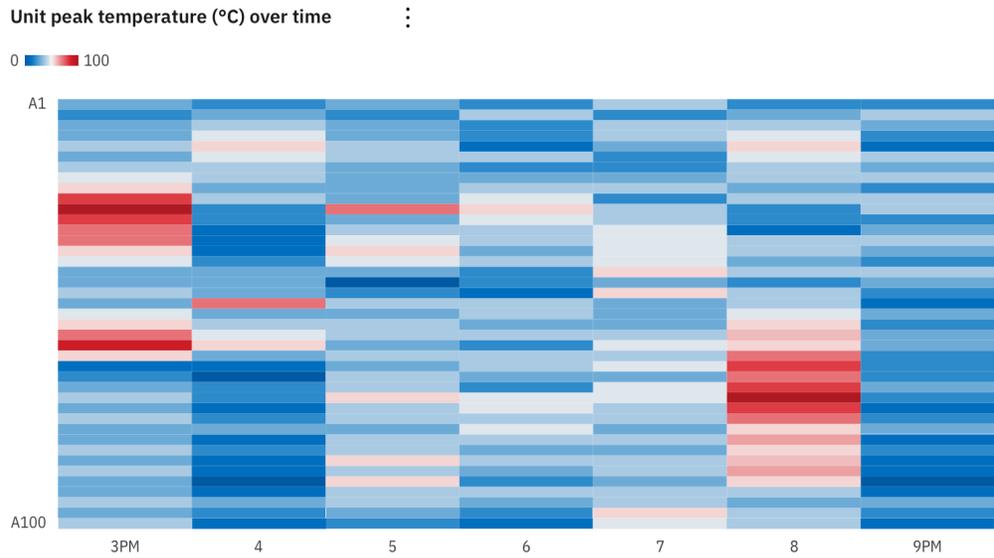
Tree Map ■ Oceania ■ Europe ■ America ■ Australia ■ Africa ■ Asia

Korrelationen

Scatter (discrete)

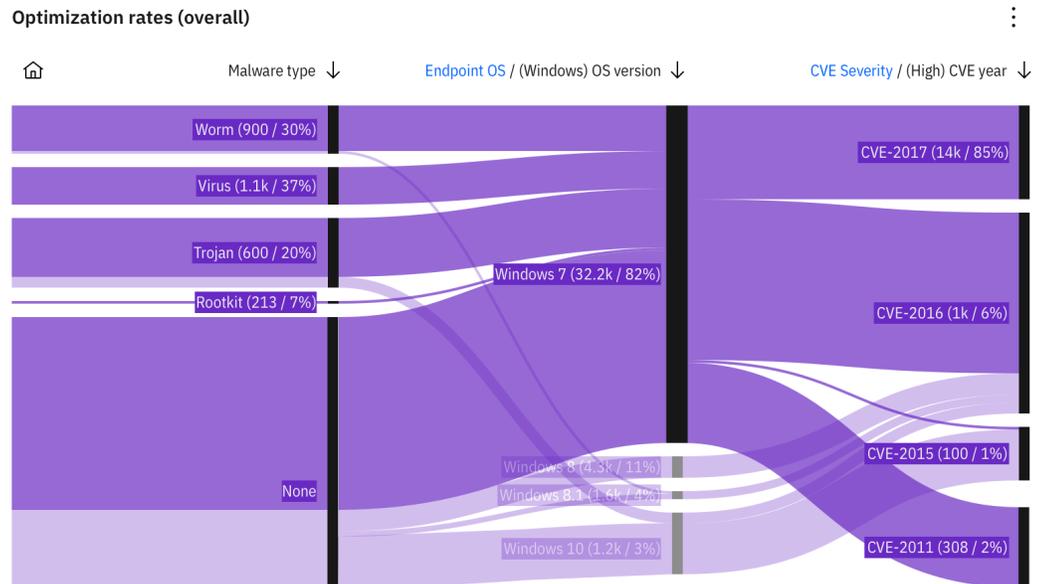


Streudiagramm

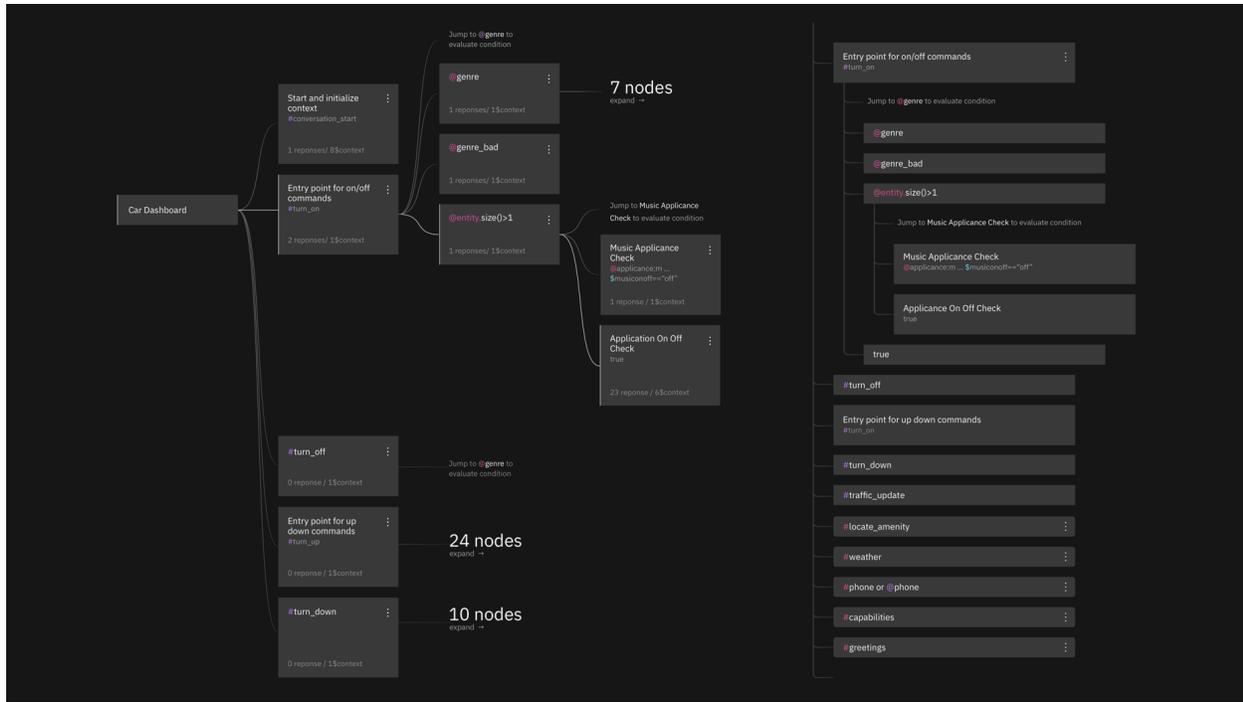


Heatmap

Verbindungen

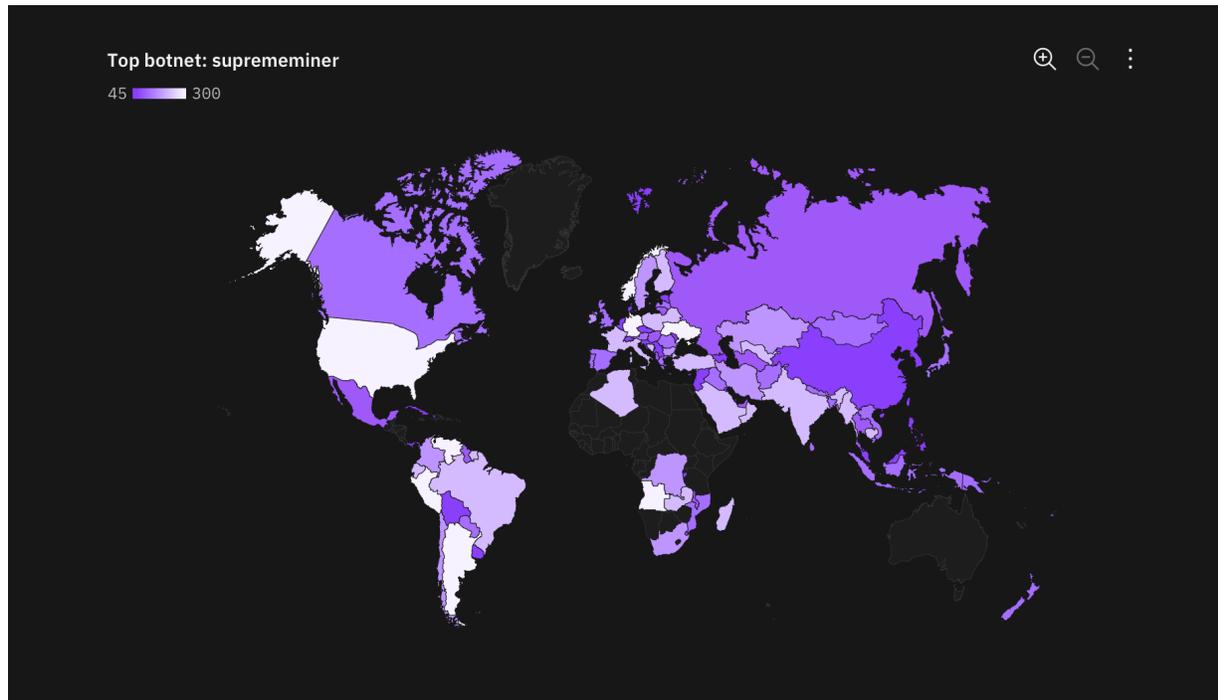


Alluvial-Diagramm



Baumdiagramm

Geodaten



Choroplethenkarte

Optimization rates (overall)



Symbole für Proportionen

Optimization rates (overall)

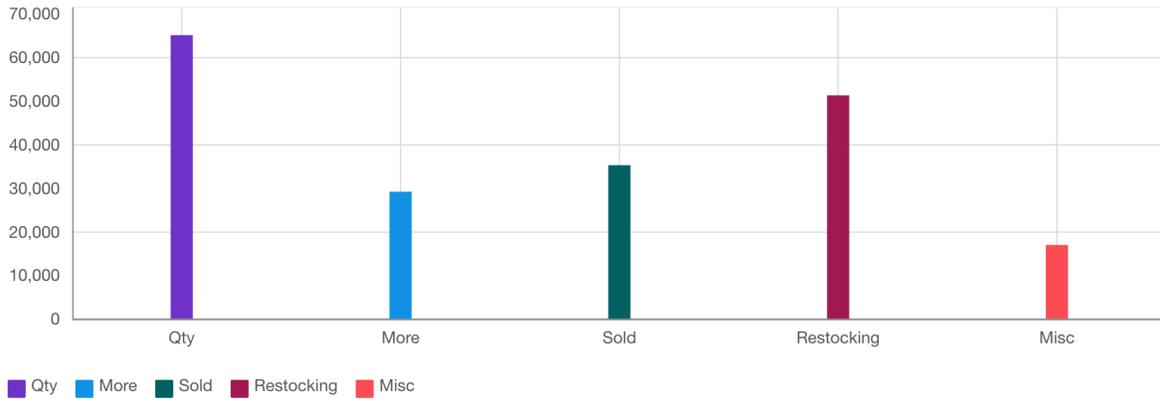


Verbindungskarte

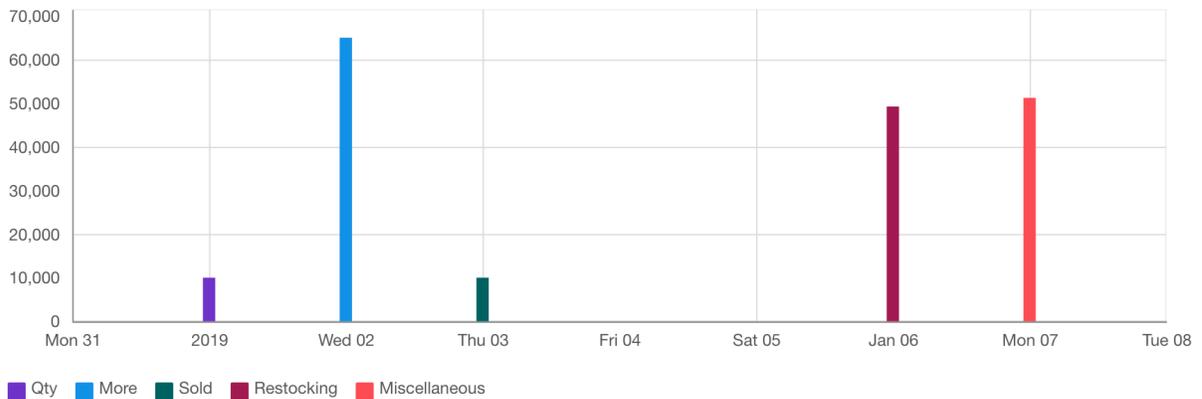
Balkendiagramme

Balkendiagramme verwenden vertikale oder horizontale Datenmarkierungen, um einzelne Werte zu vergleichen. Sie können verwendet werden, um diskrete Daten zu vergleichen oder Trends im Zeitverlauf anzuzeigen.

Simple bar (discrete)



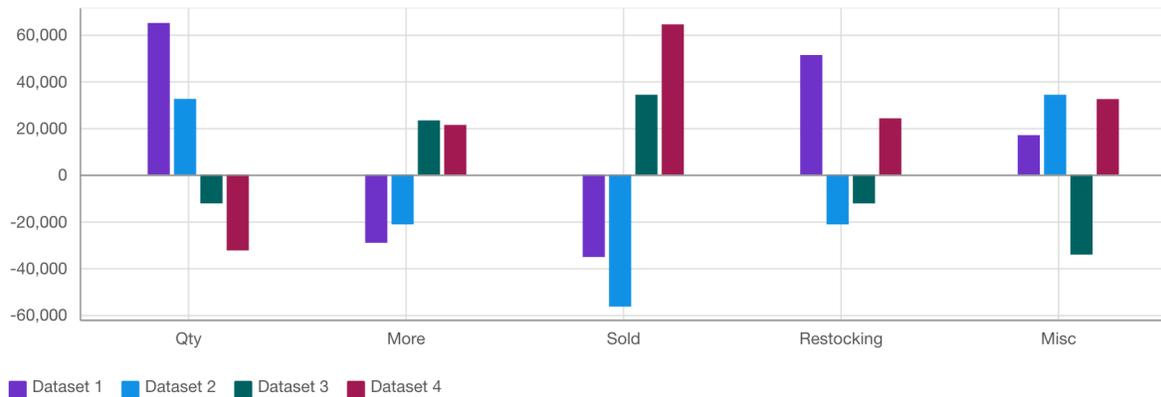
Simple bar (time series)



Gruppierte Balkendiagramme

Ein gruppiertes Balkendiagramm mit mehreren Sätzen erlaubt Werte über mehrere Kategorien hinweg zu vergleichen.

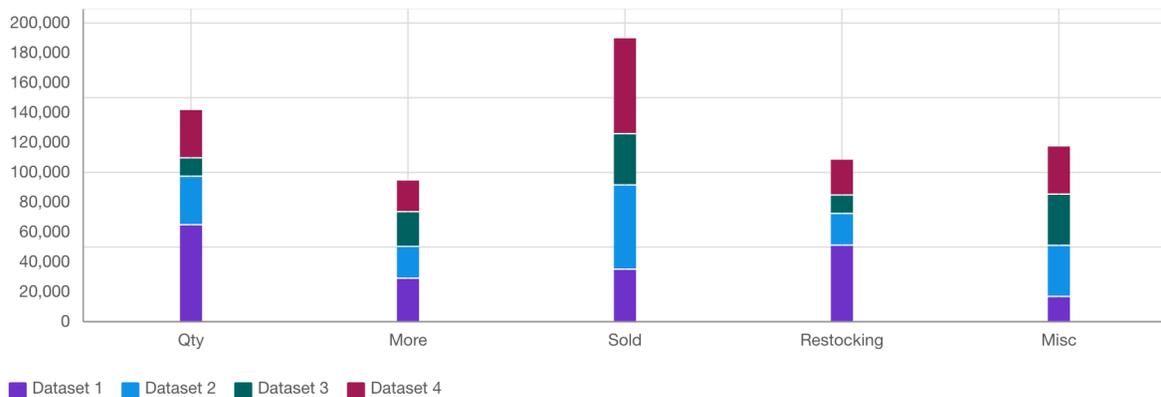
Grouped bar (discrete)



Gestapelte Balkendiagramme

Gestapelte Balkendiagramme sind nützlich, um proportionale Beiträge innerhalb einer Kategorie zu vergleichen. Sie zeigen den relativen Wert aller Datenreihen zur Gesamtsumme.

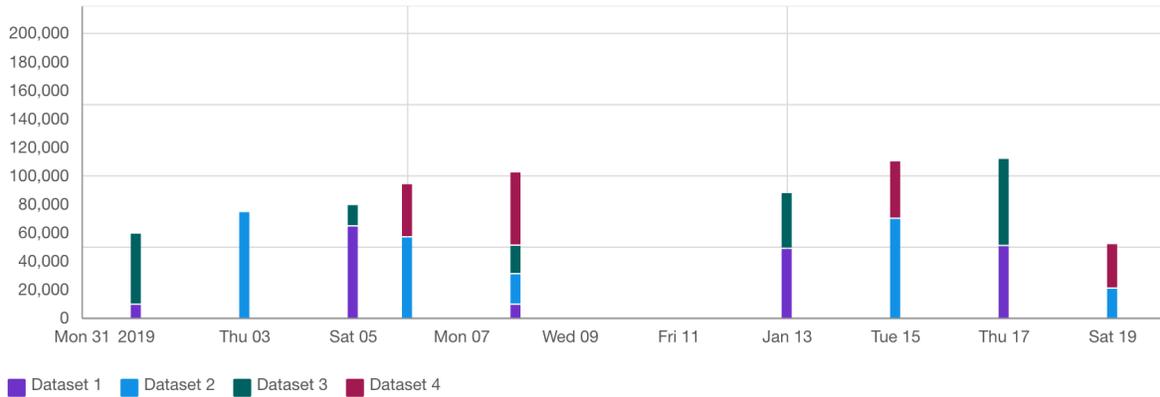
Stacked bar (discrete)



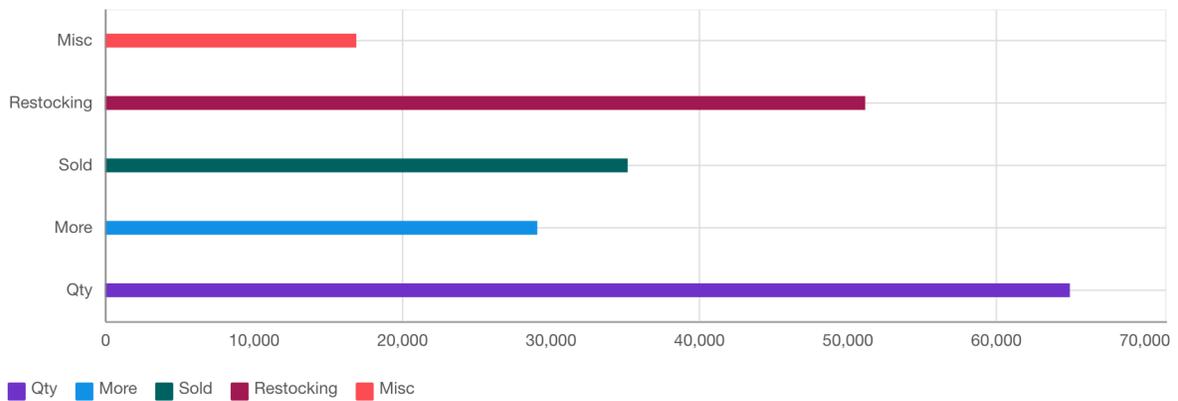
Horizontale Balkendiagramme

Balkendiagramme verwenden vertikale oder horizontale Datenmarkierungen, um einzelne Werte zu vergleichen. Sie können verwendet werden, um diskrete Daten zu vergleichen oder Trends im Zeitverlauf anzuzeigen.

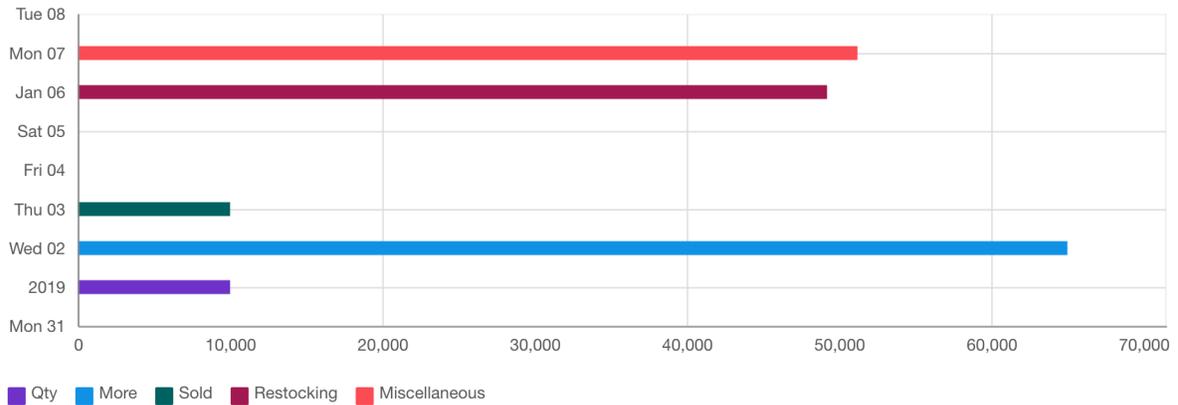
Stacked bar (time series)



Simple horizontal bar (discrete)



Simple horizontal bar (time series)



Grouped horizontal bar (discrete)

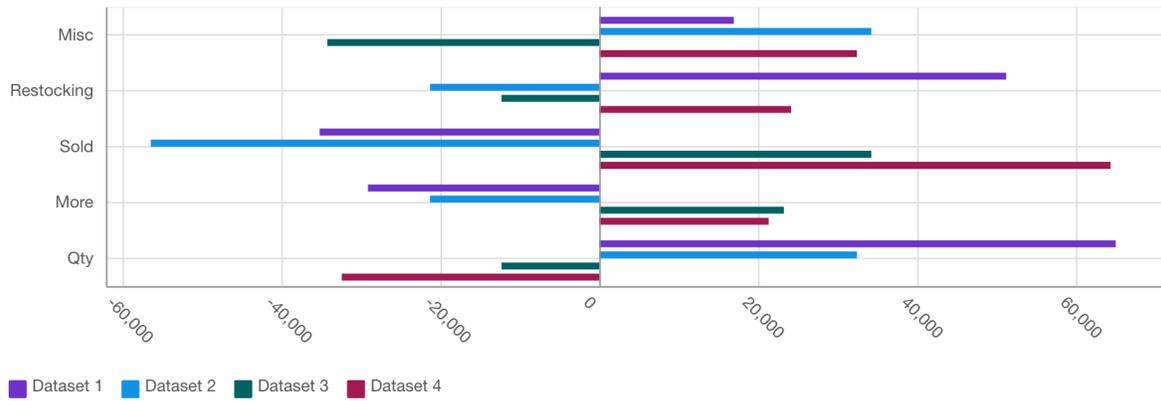
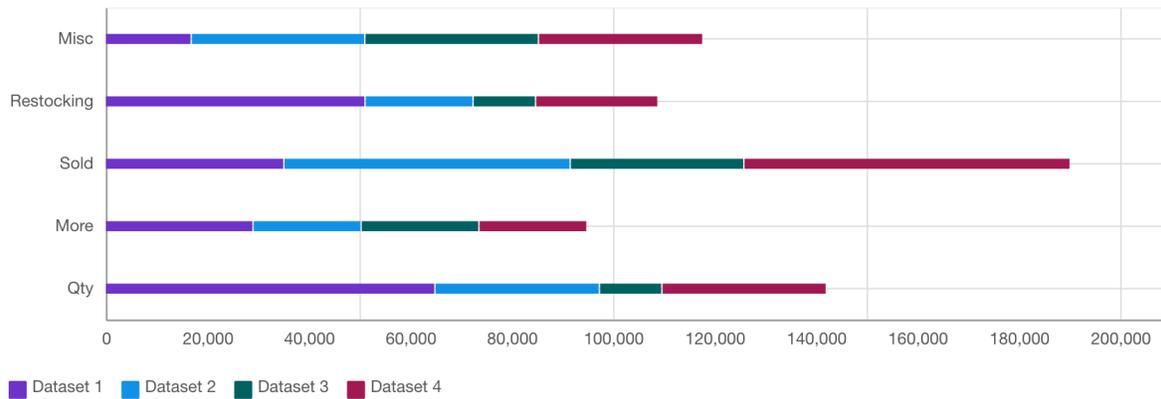


Abb. 1: Ein gruppiertes Balkendiagramm ist eine Balkendiagramm, mit dem Werte über mehrere Kategorien hinweg verglichen werden können.

Stacked horizontal bar (discrete)

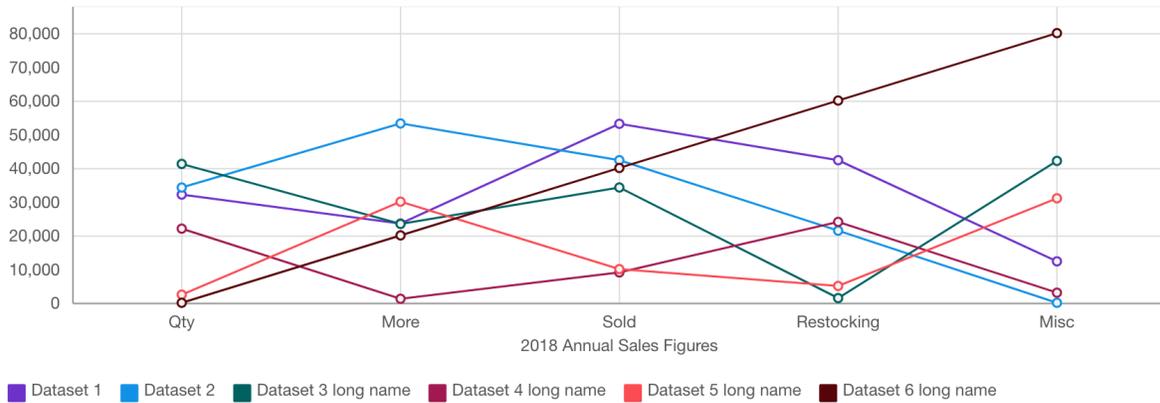


Linien- und Flächendiagramme

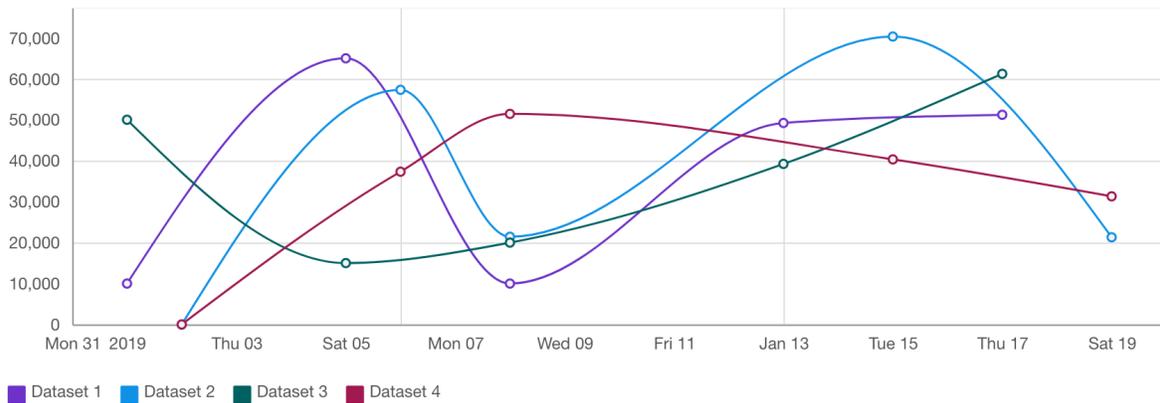
Linendiagramme

Linendiagramme zeichnen Daten in regelmäßigen Abständen, die durch Linien verbunden sind. Mithilfe von Linien-
diagrammen könnt ihr Trends im Zeitverlauf anzeigen und mehrere Datensätze vergleichen.

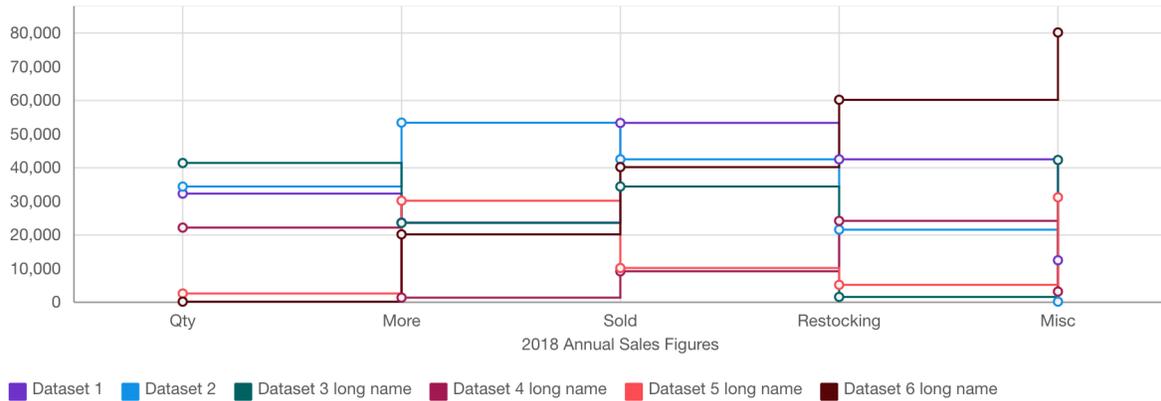
Line (discrete)



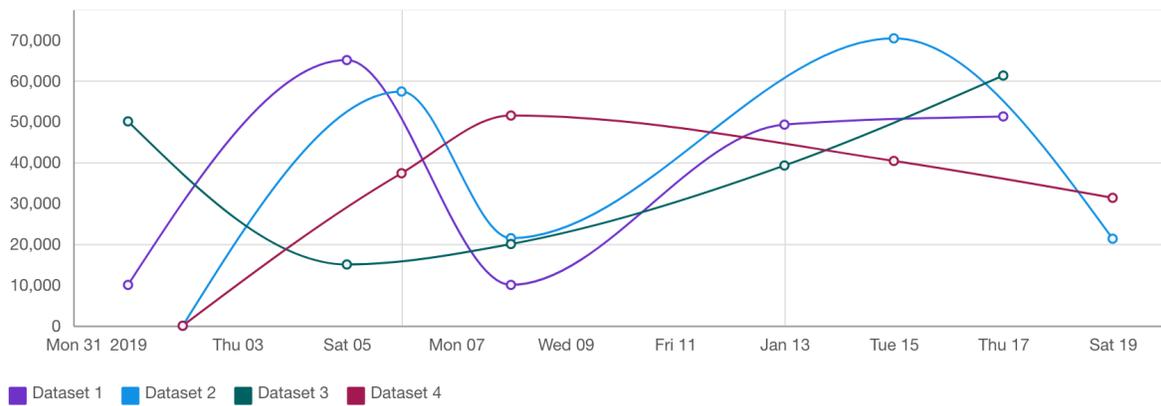
Line (time series)



Step (discrete)

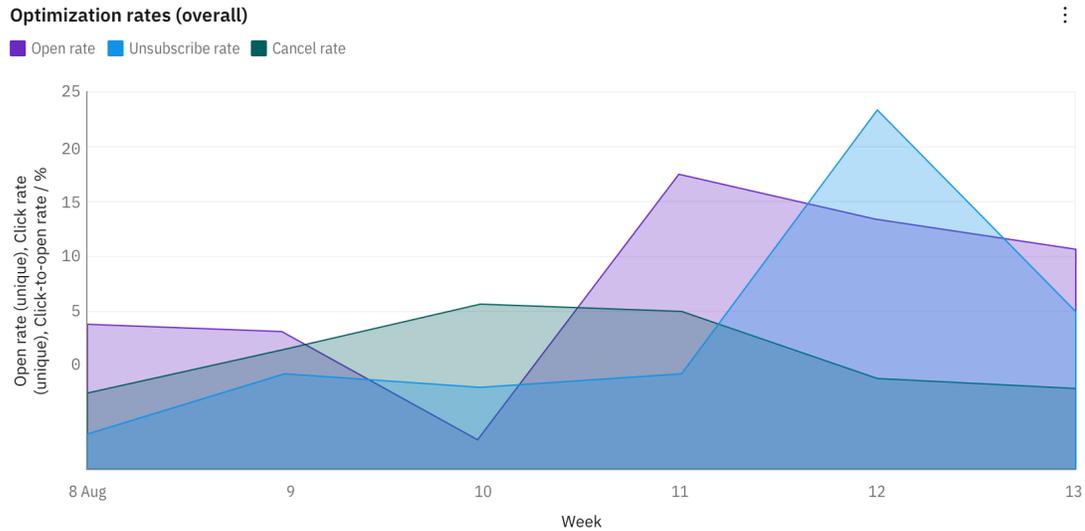


Line (time series)



Flächendiagramme

Flächendiagramme ähneln Liniendiagrammen, aber die Bereiche unter den Linien sind mit Farben oder Mustern gefüllt.



Gestapelte Flächendiagramm

Gestapelte Flächendiagramme sind nützlich, um proportionale Beiträge innerhalb einer Kategorie zu vergleichen. Sie zeichnen den relativen Wert auf, den jede Datenreihe zur Gesamtsumme beiträgt.

Kreis- und Ringdiagramme

Kreisdiagramm

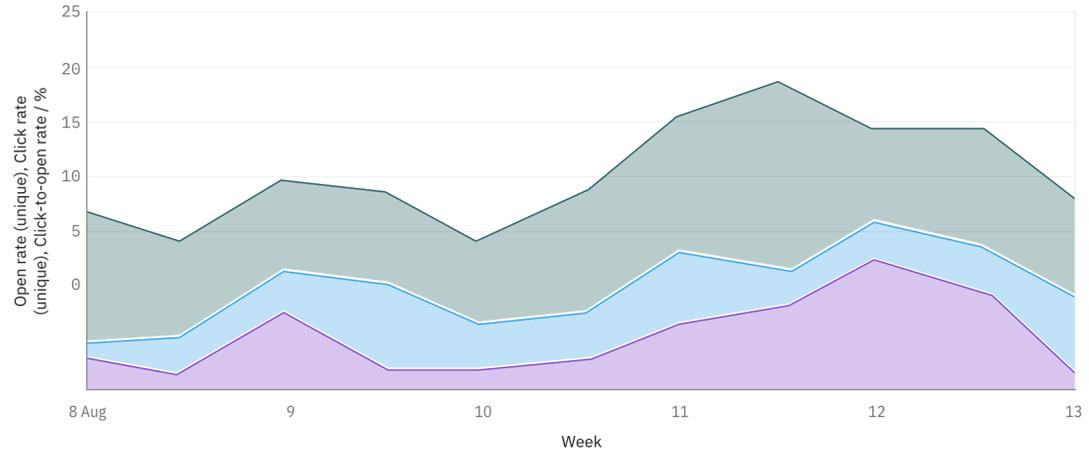
Ringdiagramm

Halbkreisförmiges Ringdiagramm

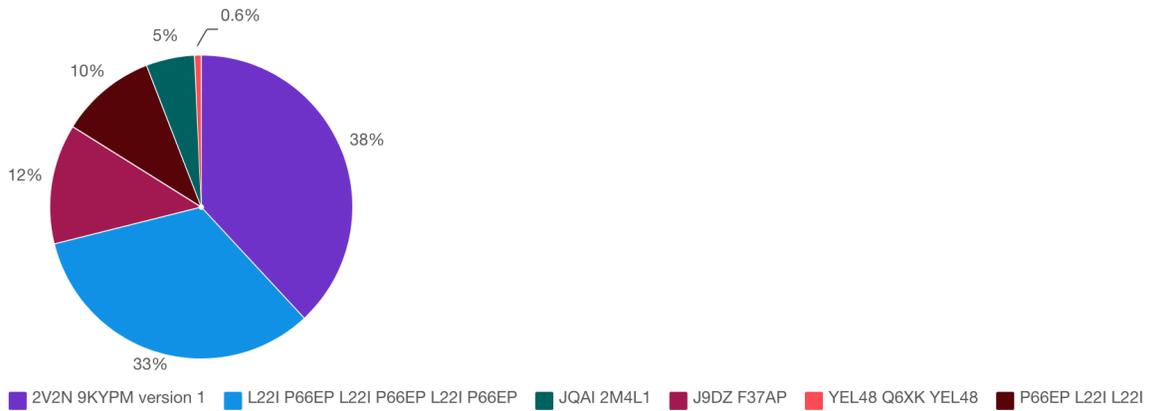
Halbkreisförmige Ringdiagramme sind nützlich, um Werte zwischen einer kleinen Anzahl von Variablen anzuzeigen, indem entweder mehrere Markierungen auf demselben Gradmesser oder mehrere Diagramme verwendet werden.

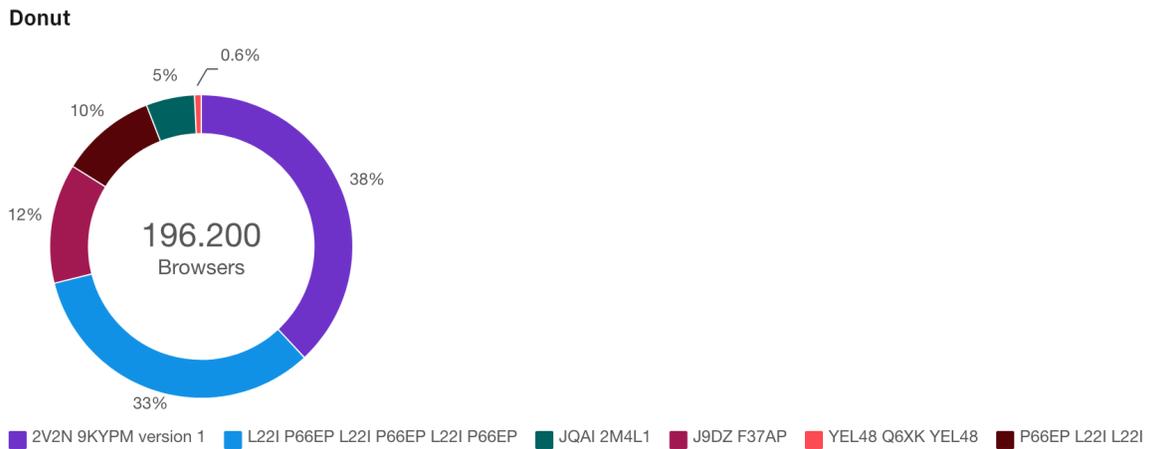
Optimization rates (overall)

Open rate Unsubscribe rate Cancel rate



Pie





Streu- und Blasendiagramme

Streudiagramme

Streudiagramme (engl. Scatter-Plots) verwenden Datenpunkte, um zwei Kennzahlen an einer beliebigen Stelle entlang einer Skala zu zeichnen.

Blasendiagramme

Blasendiagramme verwenden Datenpunkte und Blasen, um Kennzahlen entlang zweier Achsen zu zeichnen. Die Größe der Blase ist das dritte Maß.

Graphen und Flussdiagramme

Graphen

Baumdiagramme

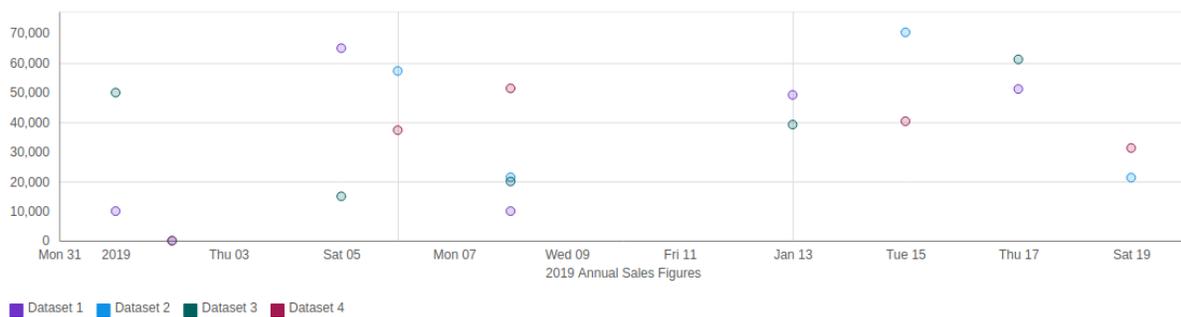
Ein Baumdiagramm, auch Stemma genannt, ist eine graphische Darstellung, welche hierarchische Abhängigkeiten durch Verbindungslinien darstellt. Mögliche Anwendungsfälle sind z.B. Stammbäume, Entscheidungsbäume und Mindmaps.

Memory utilization



Abb. 2: Die Anzeige kann beim Laden der Seite auf den aktuellen Wert animiert werden, um eine Hervorhebung zu erzielen.

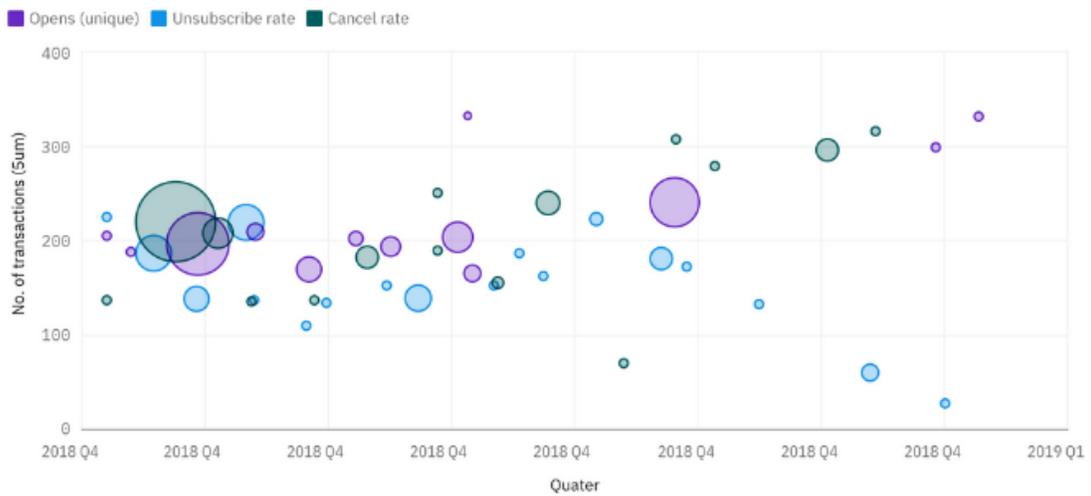
Scatter (time series)



Scatter (discrete)



Sales by product



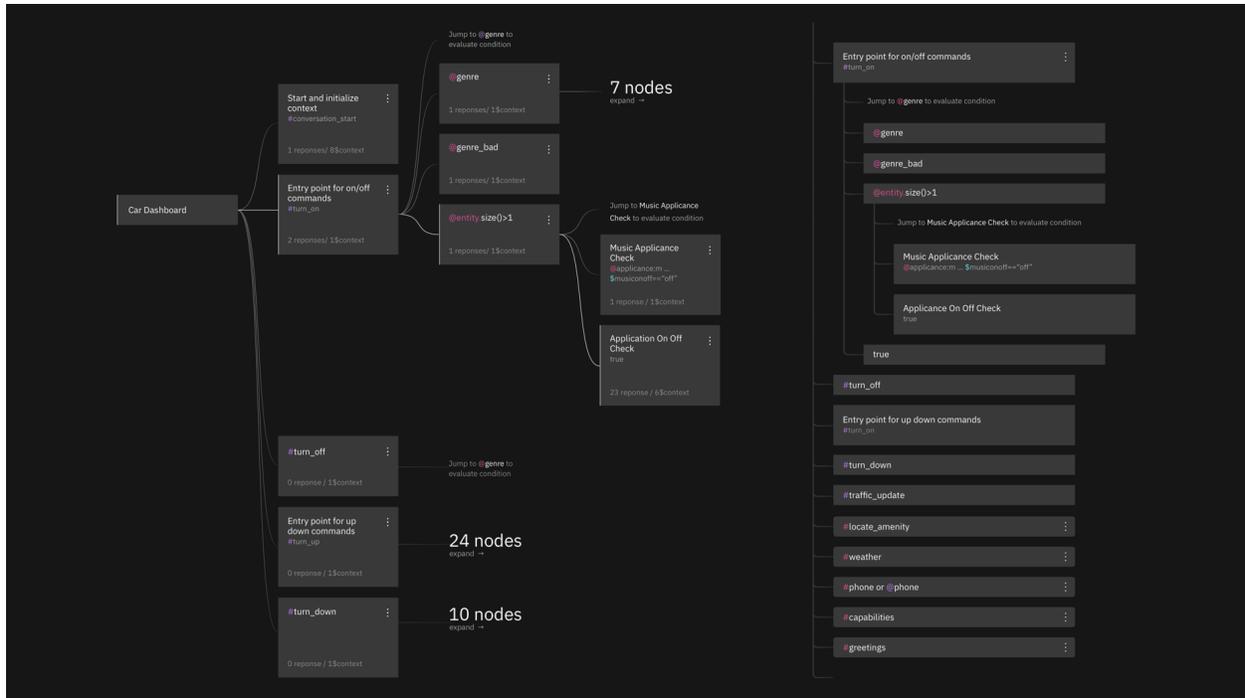


Abb. 3: Baumdiagramm, Knotenansicht vs. Listenansicht

Flussdiagramme

Es gibt verschiedene Typen von Flussdiagrammen, z.B.

Alluviale Diagramme

betont signifikante Veränderungen in Netzwerken und fasst sie ggf. zusammen.

Kontrollflussdiagramme

zur Beschreibung von Regelkreisläufen

Kumulative Flussdiagramme

werden z.B. in der [Queueing Theory](#) verwendet.

Zustandsübergangdiagramme

zur grafischen Darstellung endlicher Automaten

Karten

Heatmaps

Eine Heatmap oder Wärmekarte ist eine grafische Darstellung von Daten, bei der einzelne in einer Matrix enthaltene Werte als Farben dargestellt werden.

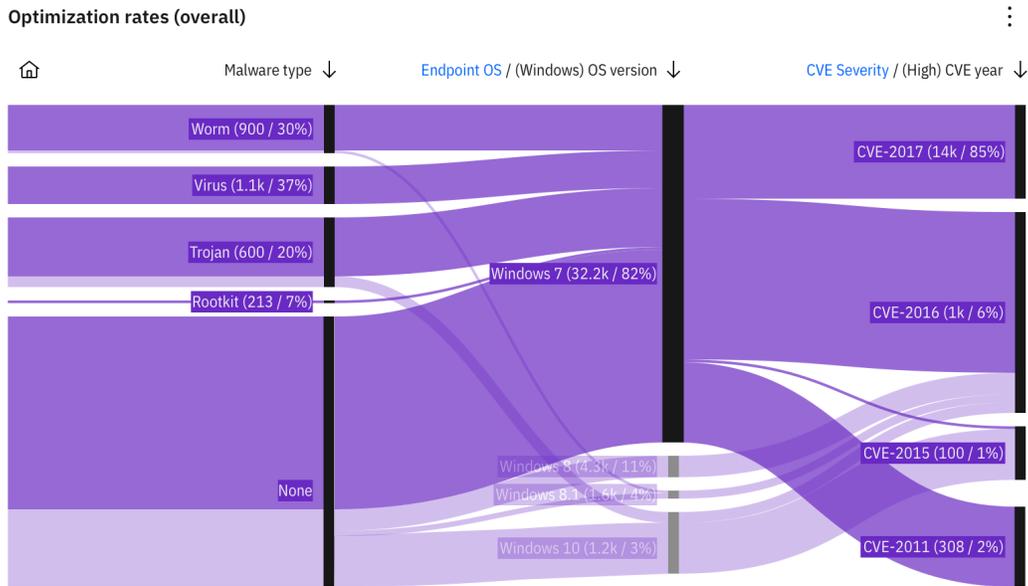
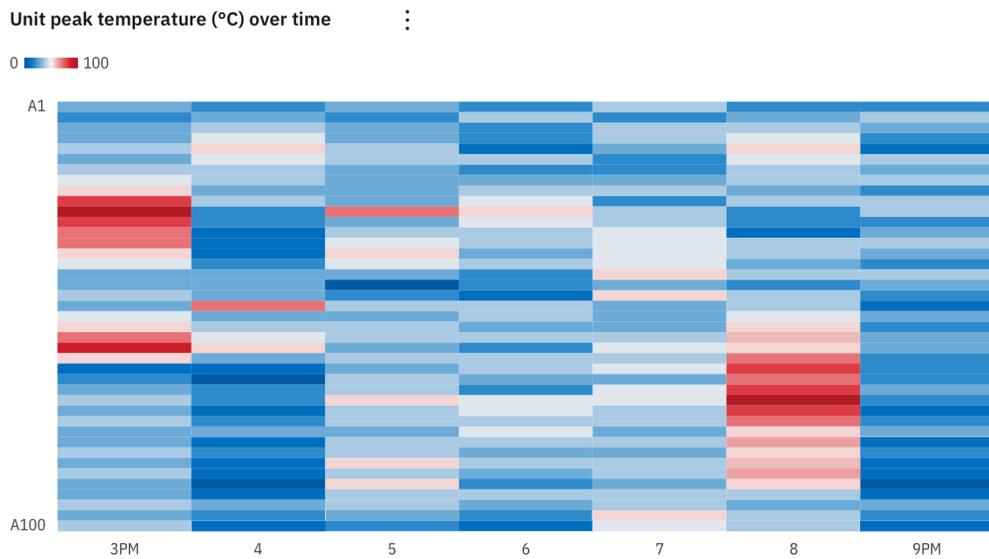


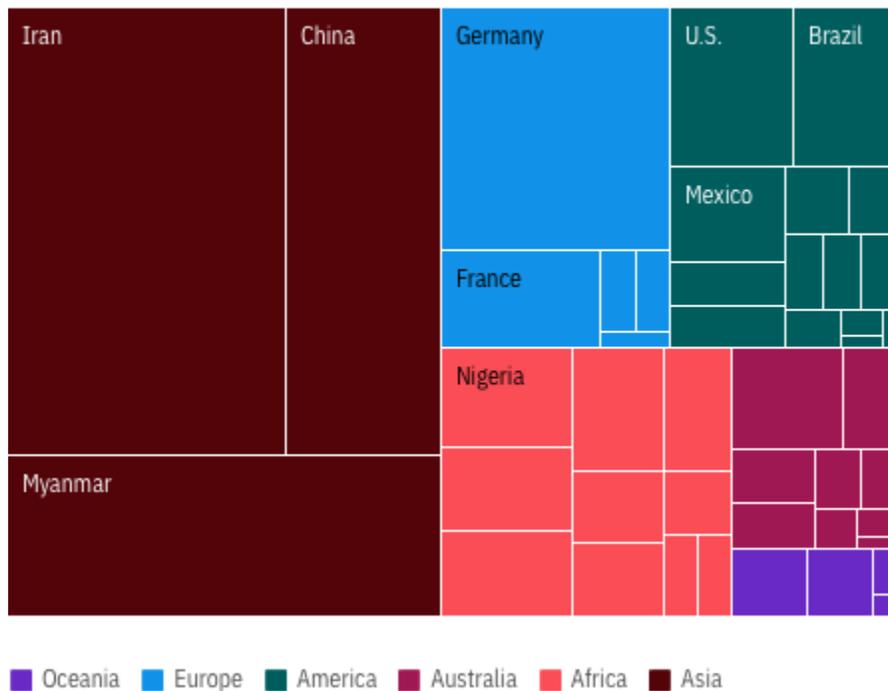
Abb. 4: Beispiel eines alluvialen Diagramms



Tree Maps

Eine Tree Map oder ein Kacheldiagramm dient der Visualisierung hierarchischer Strukturen mit ineinander verschachtelten Rechtecken. Damit können anschaulich Größenverhältnisse dargestellt werden, indem die Fläche der Rechtecke proportional zur Datenmenge dargestellt wird.

Treemap



Geographische Karten mit Überlagerungen

Choroplethenkarten

Eine Choroplethenkarte verwendet Unterschiede in Schattierung, Färbung oder Platzierung von Symbolen in vordefinierten Bereichen, um die Durchschnittswerte einer Eigenschaft oder Menge in diesen Bereichen anzugeben.

Symbole für Proportionen

Geographische Regionen werden von, durch Daten gesteuerten Symbolen für Proportionen überlagert. Ein häufig verwendetes Symbol ist eine Blase, deren Kreisfläche proportional zu ihrem Wert im Datensatz ist.

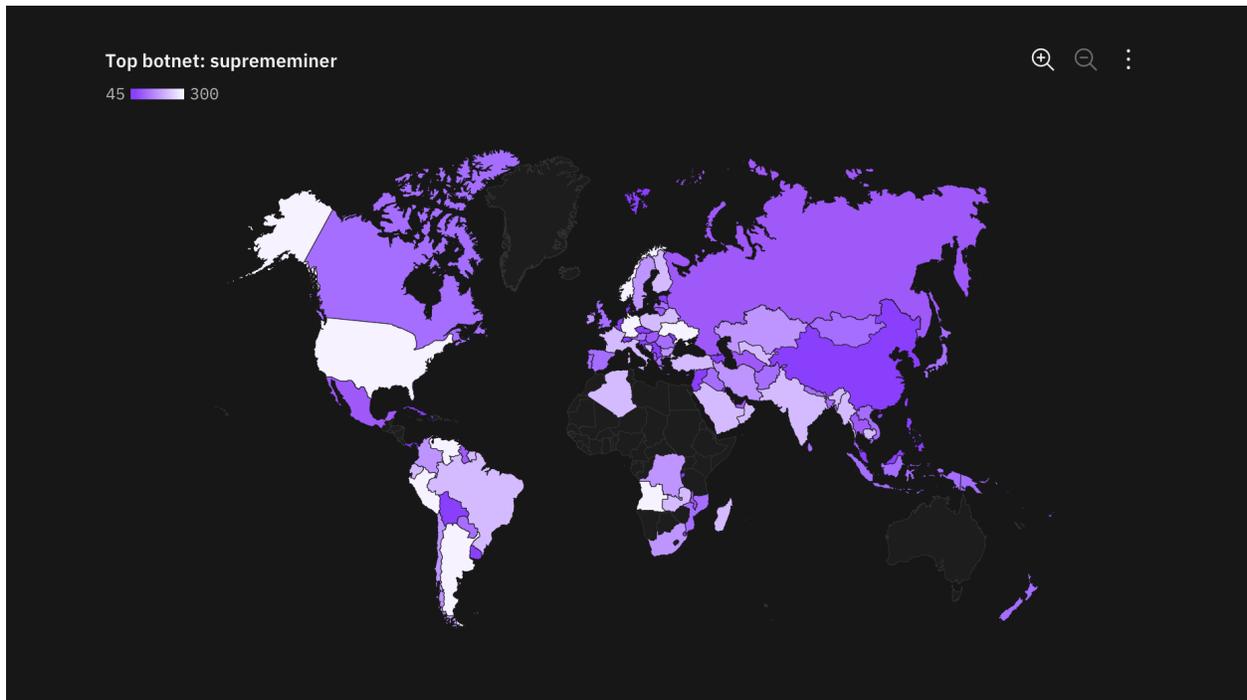


Abb. 5: Beispiel einer Choroplethenkarte

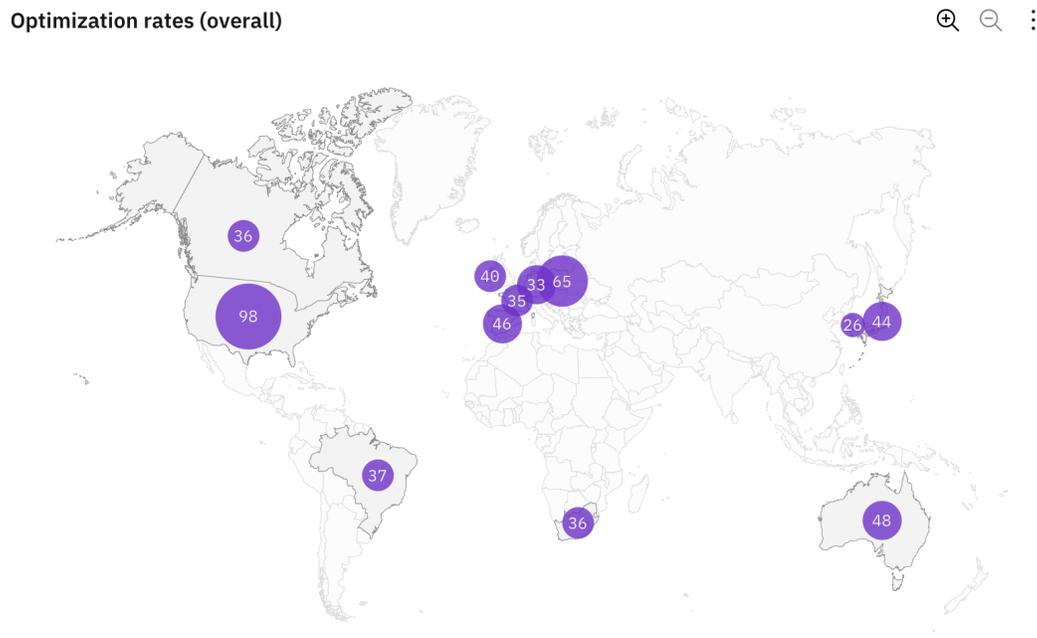


Abb. 6: Beispiel eines proportionalen Symbols

Verbindungskarten

Verbindungskarten zeigen Punkte auf einer Karte an, die durch gerade oder gekrümmte Linien verbunden sind.

Verbindungszuordnungen sind am nützlichsten, wenn Verbindungen und Beziehungen geografisch angezeigt werden. Sie können auch verwendet werden, um Kartenrouten über eine einzelne Kette von Gliedern anzuzeigen. Auch räumliche Muster lassen sich durch die Verteilung von Verbindungen oder durch die Konzentration von Verbindungen verdeutlichen.

Optimization rates (overall)



Abb. 7: Beispiel einer Karte mit Verbindungslinien

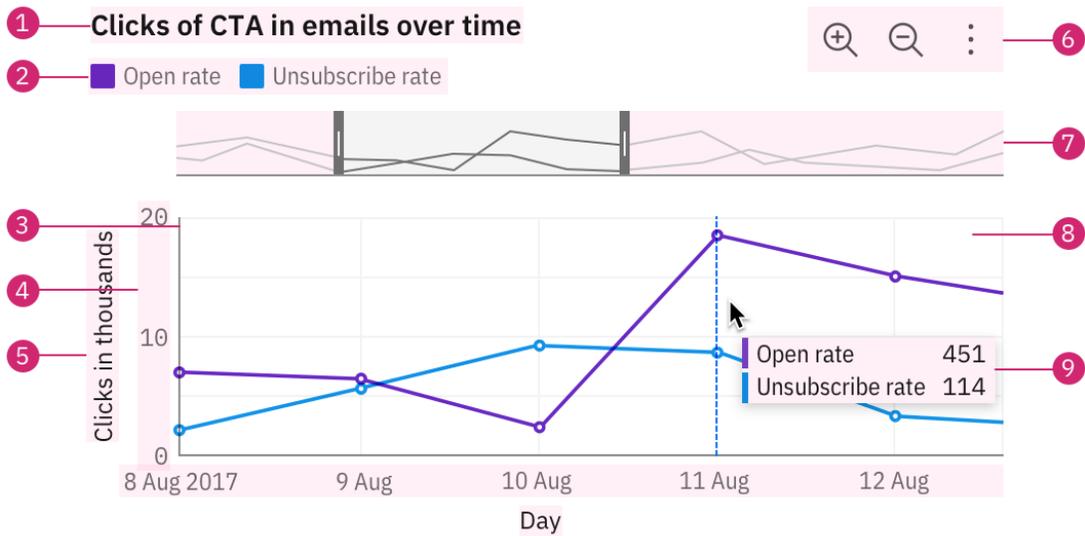
Aufbau eines Diagramms

Die Elemente, aus denen jedes Diagramm besteht, sind so konzipiert, dass sie harmonisch funktionieren. Jedes Teil spielt eine wichtige Rolle in der Datenkommunikation.

Rechteckige Diagramme

Die meisten Datenvisualisierungen sind rechteckige Diagramme mit zwei Dimensionen, die auf einer vertikalen und einer horizontalen Achse dargestellt werden. Rechteckige Diagramme werden normalerweise mit einer Reihe gemeinsamer Elemente erstellt, darunter eine Legende, Achsentitel und Navigationswerkzeuge wie eine Zoomleiste und ein Tooltip.

1. Diagrammtitel
2. Legenden
3. Achsen
4. Maßstab
5. Achsentitel



- 6. Symbolleiste
- 7. Zoomleiste
- 8. Diagrammrahmen
- 9. Tooltip

Diagrammtitel

Der Titel einer Visualisierung sollte präzise und beschreibend sein und die Erkenntnisse widerspiegeln, die die Daten liefern.

Legende

Viele Diagramme verwenden unterschiedliche visuelle Eigenschaften wie Farben, Texturen und Formen, um unterschiedliche Kategorien oder Dimensionen von Daten darzustellen. Eine Legende sagt, was diese Assoziationen bedeuten, und hilft, die Bedeutung des Diagramms zu bestimmen. Weitere Informationen findet ihr in *Legenden*.

Achsen

Die X-Achse ist horizontal und die Y-Achse ist vertikal. Achsen sollten vor ihrem Hintergrund mit einem Kontrastverhältnis von 3:1 zugänglich sein. Weitere Informationen findet ihr in *Achsen*.

Tooltip

Ein Tooltip ist eine Meldung, die angezeigt wird, wenn sich ein Cursor über einem Element befindet, z.B. über einem Datenpunkt oder über einer Schaltfläche. Standardmäßig werden Tooltips bei *hover* angezeigt, um detailliertere Informationen oder den Kontext für bestimmte Diagrammelemente anzuzeigen. Ein Tooltip sollte die entsprechenden Werte des Datenpunkts auf beiden Achsen wiederholen und weitere relevante Details anzeigen.

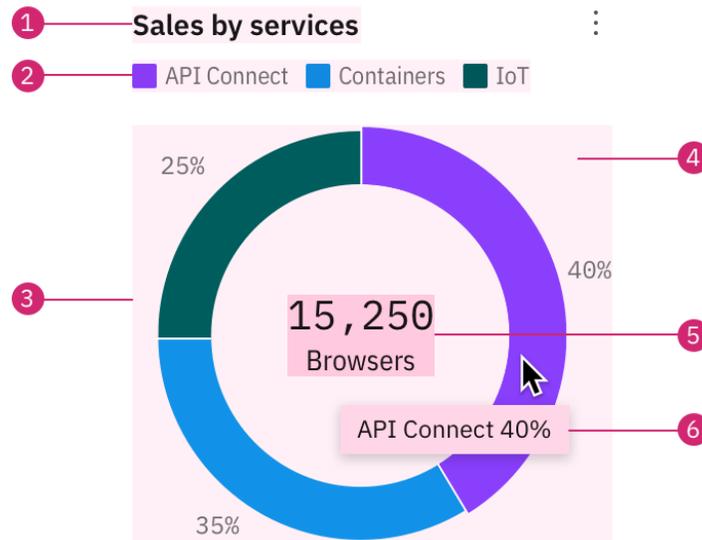
Grafikrahmen

Der Grafikrahmen ist der Bereich, in dem Daten mit Grafiken visualisiert werden. Gitterlinien helfen dem Betrachter, die Werte eines Datenpunkts zu schätzen, sind jedoch nicht unbedingt erforderlich.

Bemerkung: Verwendet Gitterlinien sparsam, da zu viel visuelles Rauschen die Fähigkeit des Betrachters zur

Interpretation der Daten negativ beeinflusst.

Kreisdiagramme



1. Diagrammtitel
2. Legenden
3. Etikett
4. Diagrammrahmen
5. Leistungskennzahl
6. Tooltip

Etikett

Es liefert den prozentualen Wert für eine einzelne Kategorie.

Leistungskennzahl

Er besteht aus einer Zahl mit einer einzelnen Wortbeschreibung. Beispiele hierfür sind *15,250 Browser*, *1,3 Millionen US-Dollar Umsatz* oder *Insgesamt 450*. Sie kann innerhalb eines Ringdiagramms verwendet werden, um eine Gesamtsumme oder die individuelle Anzahl eines Teils bei Interaktion anzuzeigen. Dieses Element kann auch unabhängig in *Dashboards* verwendet werden.

Farbpaletten

Die Farbpalette für Datenvisualisierungen ist eine ausgewählte Teilmenge der cusy-Design-Farbpalette. Sie wurde entwickelt, um die Zugänglichkeit und Harmonie innerhalb einer Seite zu verbessern.

Kategorien

Kategoriale (oder qualitative) Paletten eignen sich am besten, wenn Sie diskrete Datenkategorien unterscheiden sollen, die keine inhärente Korrelation aufweisen.

Die Farben dieser Palette sollten nacheinander genau wie unten beschrieben angewendet werden. Die Sequenz wird sorgfältig kuratiert, um den Kontrast zwischen benachbarten Farben zu maximieren und die visuelle Unterscheidung zu erleichtern.

Alternativen für das Light Theme

Alternativen für das Dark Theme

Sequenzen

Einfarbig

Monochromatische Paletten eignen sich gut für Beziehungs- und Trenddiagramme. Beim hellen Theme bezeichnet die dunkelste Farbe die größten Werte. In dunklen Themes bezeichnet die hellste Farbe die größten Werte.

Warm-Kalt

Die Rot-Cyan-Palette hat einen natürlichen Zusammenhang mit der Temperatur. Verwendet diese Palette für Daten, die heiß-vs-kalt darstellen sollen.

Verläufe ohne Farbassoziationen

Alarm

Warnfarben werden verwendet, um den Status wiederzugeben. In der Regel steht Rot für Gefahr oder Fehler. Orange ist eine ernsthafte Warnung. Gelb steht für eine einfache Warnung und Grün für Normal oder Erfolg.

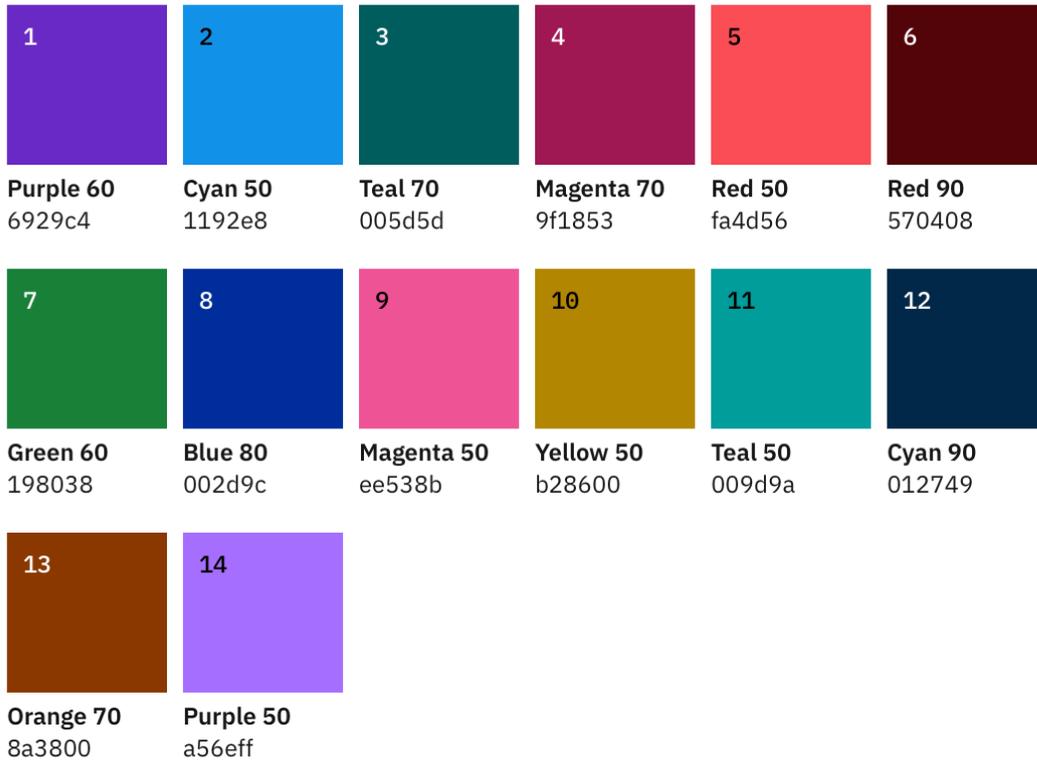
Farbverlauf

Farbverläufe eignen sich gut zum Hervorheben von Extremen in einem Wertebereich. Verwendet einen Verlauf nur bei Bedarf für Visualisierungen separater Kategorien. Mehrere Farbverläufe sind oft nicht zugänglich und nur selten empfehlenswert. Farbverläufe sollten nicht verwendet werden, um eine Progression oder Divergenz darzustellen.

Bemerkung: Verwendet niemals einen Farbverlauf anstelle einer sequentiellen Palette.

Siehe auch:

- Paul Tol



Set 1



Set 2



Set 3



Set 1



Set 2



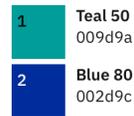
Set 3



Set 4



Set 5



Set 1



Set 2



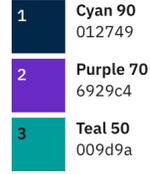
Set 3



Set 4



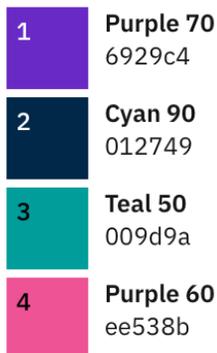
Set 5



Set 1

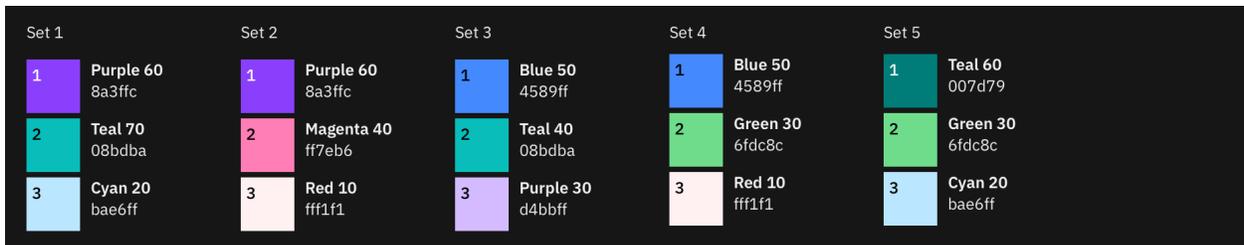
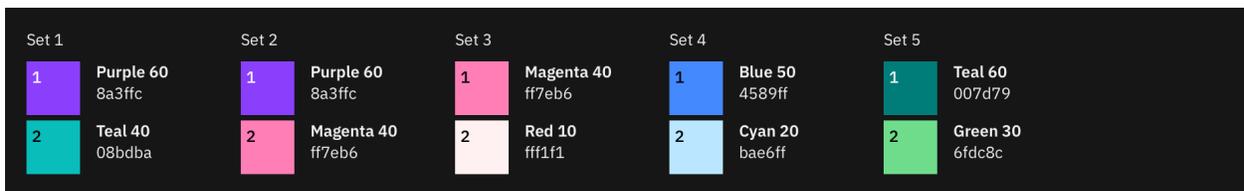


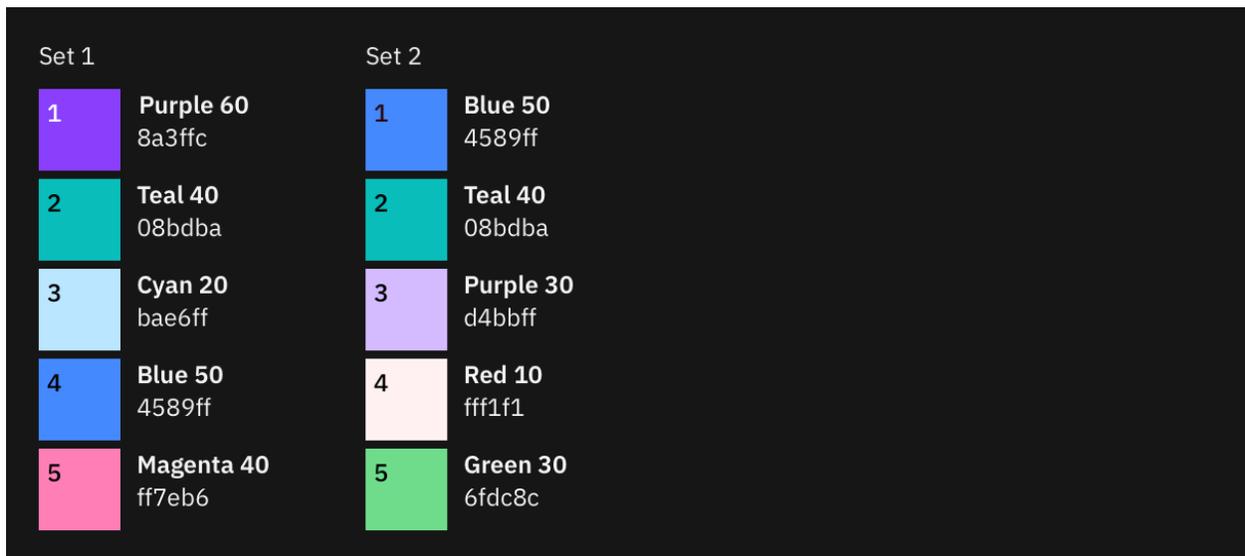
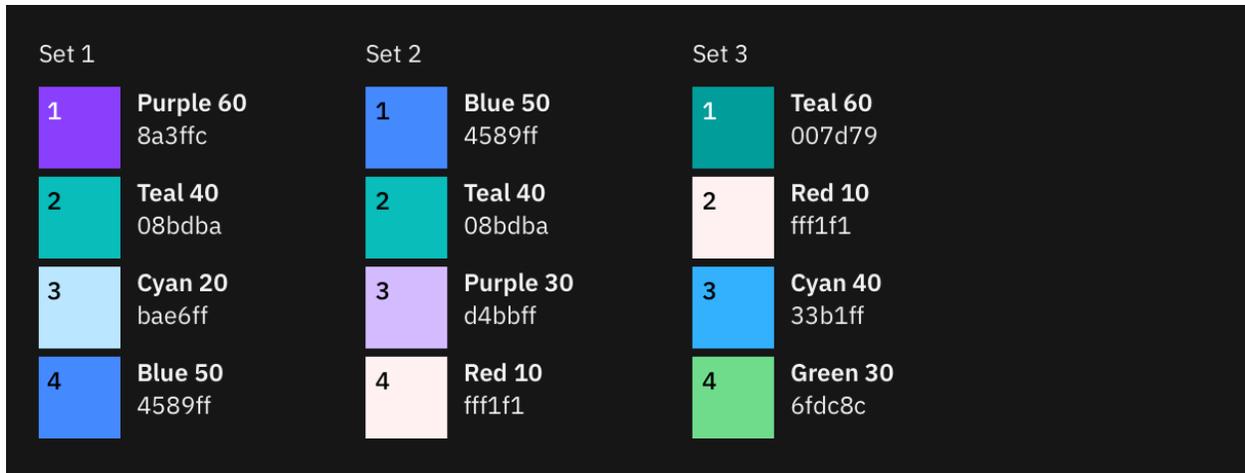
Set 2



Set 3

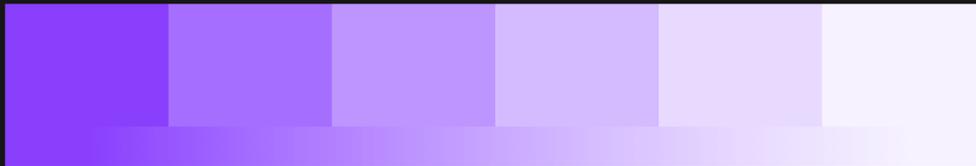








Purple 50 a56eff	Purple 60 8a3ffc	Purple 70 6929c4	Purple 80 491d8b	Purple 90 31135e	Purple 100 1c0f30
----------------------------	----------------------------	----------------------------	----------------------------	----------------------------	-----------------------------



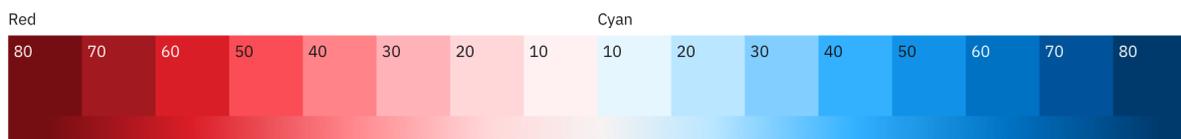
Purple 60 8a3ffc	Purple 50 a56eff	Purple 40 be95ff	Purple 30 d4bbff	Purple 20 e8daff	Purple 10 f6f2ff
----------------------------	----------------------------	----------------------------	----------------------------	----------------------------	----------------------------

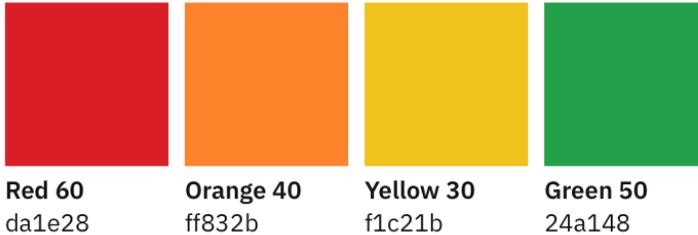
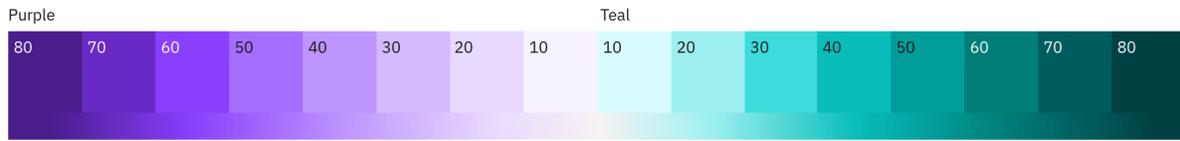


Blue 50 4589ff	Blue 60 0f62fe	Blue 70 0043ce	Blue 80 002d9c	Blue 90 001d6c	Blue 100 001141
--------------------------	--------------------------	--------------------------	--------------------------	--------------------------	---------------------------



Blue 60 0f62fe	Blue 50 4589ff	Blue 40 78a9ff	Blue 30 a6c8ff	Blue 20 d0e2ff	Blue 10 edf5ff
--------------------------	--------------------------	--------------------------	--------------------------	--------------------------	--------------------------





- [bokeh Accessible Palettes](#)

Achsen

Achsen und Beschriftungen bieten einen kritischen Kontext für Informationen in einem Diagramm. Verwendet einfache, leicht verständliche Beschreibungen und Metriken, um euer Diagramm und eure Achsen zu kennzeichnen.

Beginnend mit Null

Bei Teil-zu-Ganzes- und Vergleichsdiagrammen sollten die numerischen Achsen immer bei Null beginnen. Und auch bei Balken- und Flächendiagrammen kann eine abgeschnittene Achse die wahrgenommene Skalierung eines Diagramms verzerren, sodass ein kleiner Unterschied signifikanter erscheint als er tatsächlich ist.

Linien- und Streudiagramme können jedoch mit einem Wert über Null starten. Diese Arten von Visualisierungen reagieren weniger empfindlich auf Verzerrungen, da sie eher Trends als Unterschiede in Größe oder Menge kommunizieren. In diesen Fällen hilft das Zuschneiden der Y-Achse dem Betrachtern, die Änderungsrichtung besser zu identifizieren.

Lücken

Manchmal ist es nützlich, einen Teil der Achse zu überspringen, um Daten an den äußersten Enden ohne Verzerrung sichtbar zu machen. Wenn Achsen einen Bruch enthalten, wird die gerade Achsenlinie durch eine Sinuslinie ersetzt.

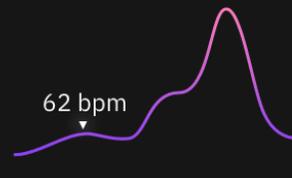
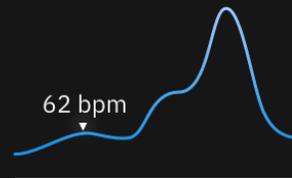
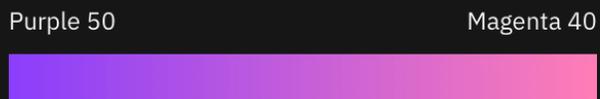
Auf der X-Achse kann die Unterbrechung fließend sein, sollte jedoch 16px nicht unterschreiten. Auf der Y-Achse sollte die Distanzunterbrechung exakt 16px betragen.



Memory utilization



Memory utilization



Clicks by mailing

■ Opens (unique) ■ Clicks (unique)

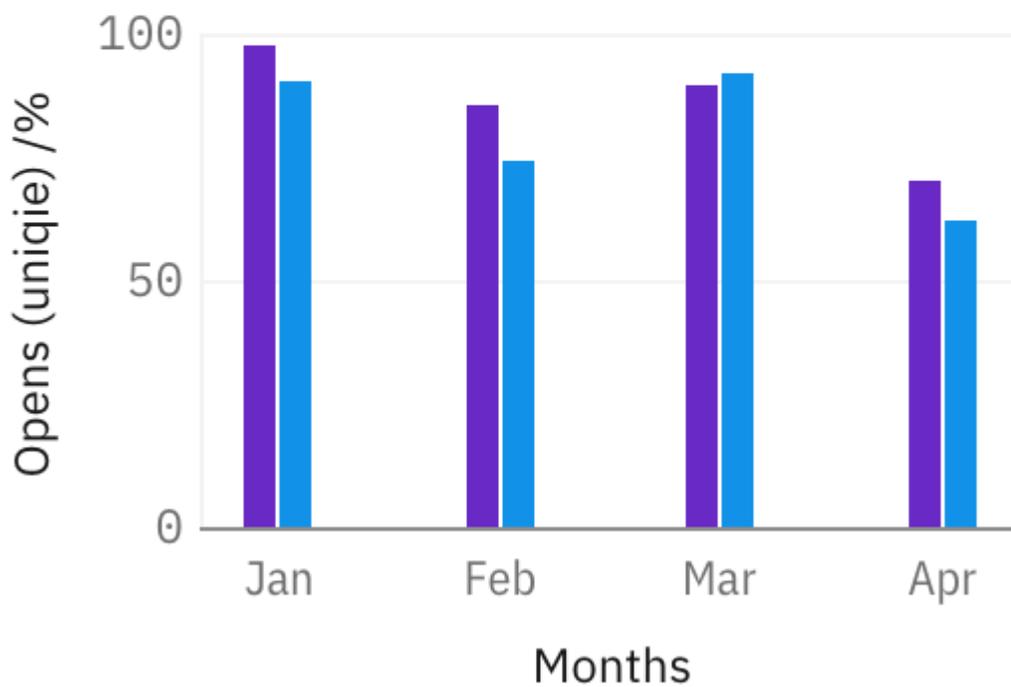


Abb. 8: Bei Balkendiagrammen sollte die numerische Achse bei Null beginnen.

Clicks by mailing

■ Opens (unique)
 ■ Clicks (unique)

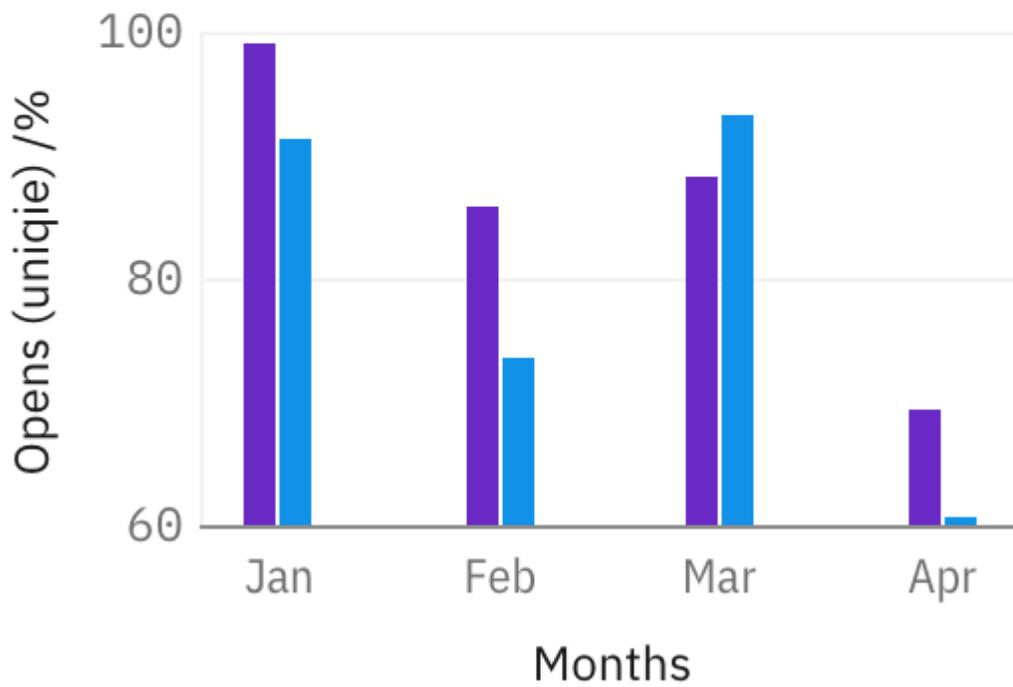


Abb. 9: Wenn eine Achse nicht bei Null beginnt, sind die prozentualen Unterschiede zwischen den Balken übertrieben.

Total stock market capitalization (in USD)

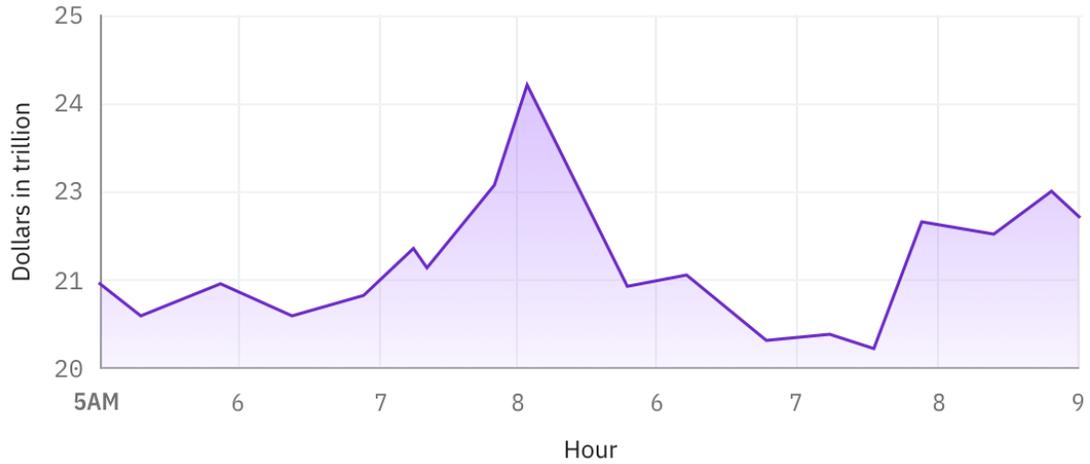


Abb. 10: Für Liniendiagramme, in denen Trends angezeigt werden, ist das Vorhandensein von Spitzen und Tälern wichtiger als die wahre Größe der Änderung. Durch Verläufe, die am gedachten Nullpunkt keine Sättigung aufweisen, sind dann völlig ausreichend um zu visualisieren, dass die Werte nicht bei Null starten.

Sales by quarter

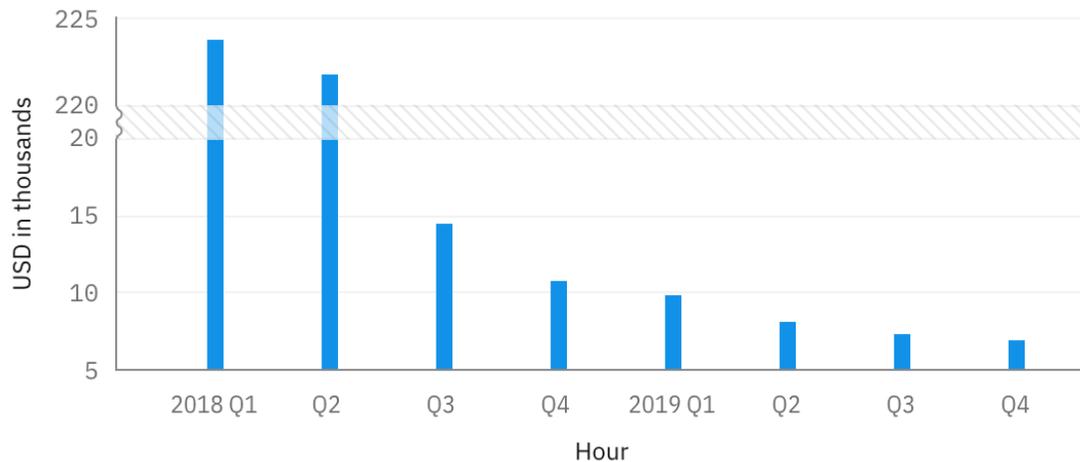


Abb. 11: Wenn während eines Achsenbruchs Daten verfügbar sind, solltet ihr die Liniensegmente so anpassen, dass sie eine Sättigung von 50% erhalten; Kreise und Datenpunkte sollten hingegen verborgen werden.

Email CTA click rate over time

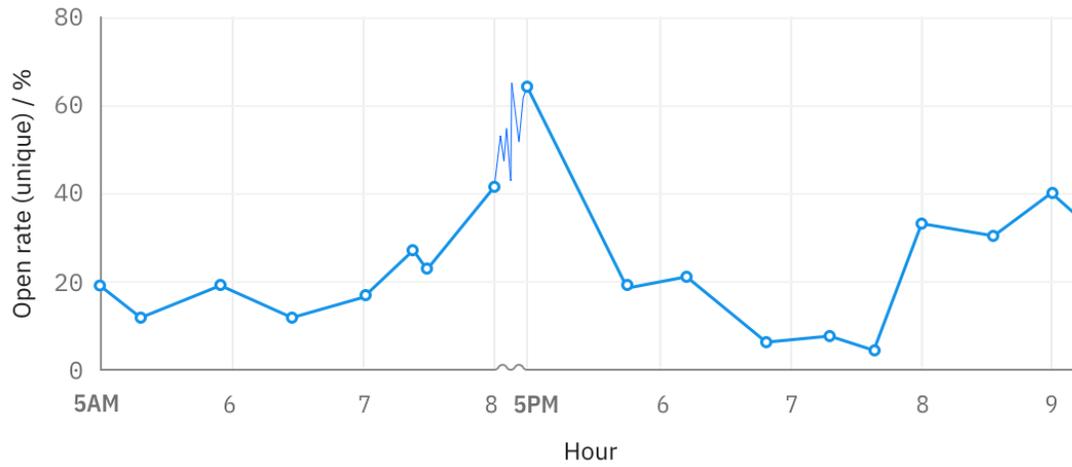
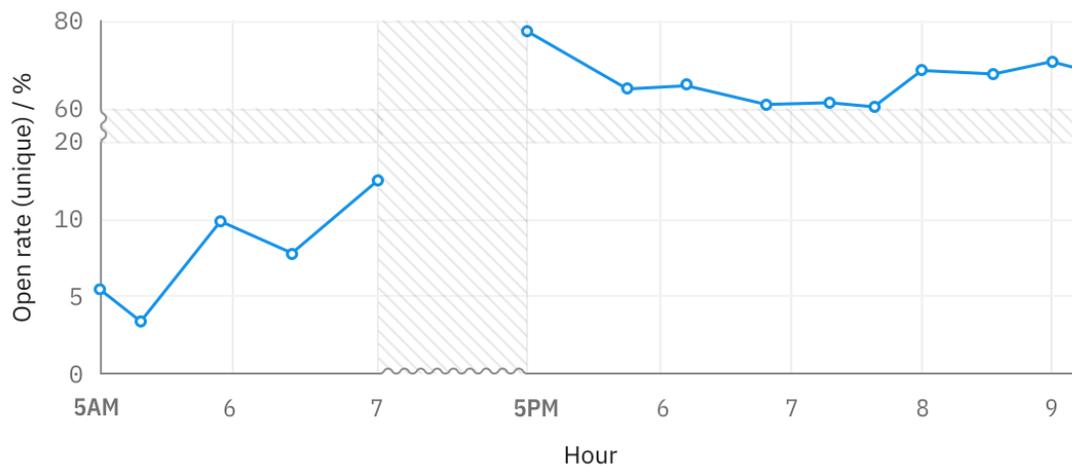


Abb. 12: Wenn zwischen den Datenpunkten keine Daten verfügbar sind, wird die Lücke mit einer Textur gekennzeichnet.

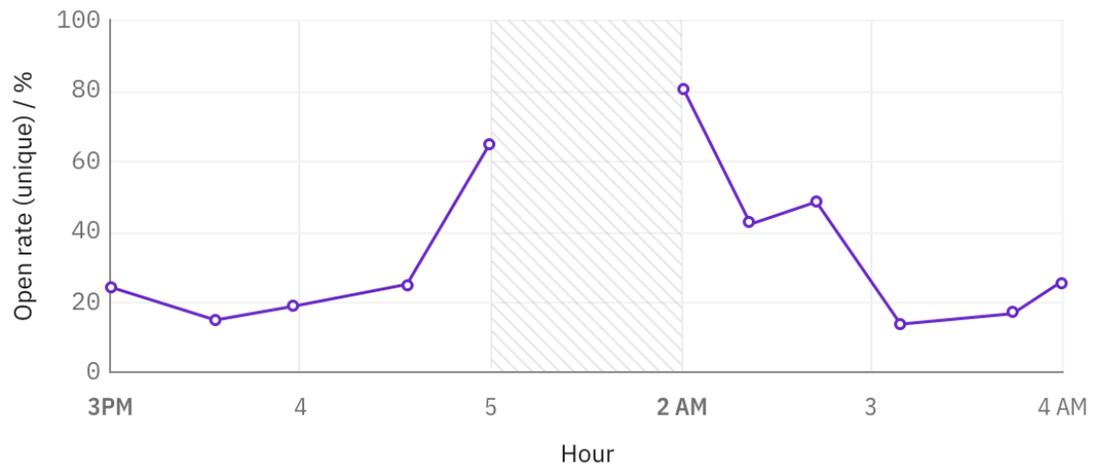
Email CTA click rate over time



Datenlücken

Um den Bereich oder Zeitraum anzugeben, in dem keine Daten verfügbar sind, kann eine Textur verwendet werden. Beschriftet immer die Start- und Endpunkte, an denen keine Daten verfügbar sind.

Gap in data example



Zeitfolgen

Konsistente Inkremente

Wenn eine Form der Achsenkomprimierung erforderlich ist, verwendet das bereitgestellte *Achsenbruch*-Styling, um die Komprimierung darzustellen.

Lokalisierung

In Zeitreihen spiegeln die Beschriftungen der X-Achse das Zeitinkrement in den Daten wider. Verwendet nach Möglichkeit das lokalisierte Datums- und Uhrzeitformat oder die Benutzereinstellungen. Andernfalls verwendet das Diagramm standardmäßig das ISO 8601-Format YYYY-MM-DD hh:mm:ss.

Orientierungspunktetiketten

Wenn Daten in einen neuen Zeitzyklus übergehen, z.B. einen neuen Tag, einen neuen Monat oder ein neues Jahr, wird das Etikett halbfett beschriftet, um den Übergang hervorzuheben.

Email CTA click rate over time

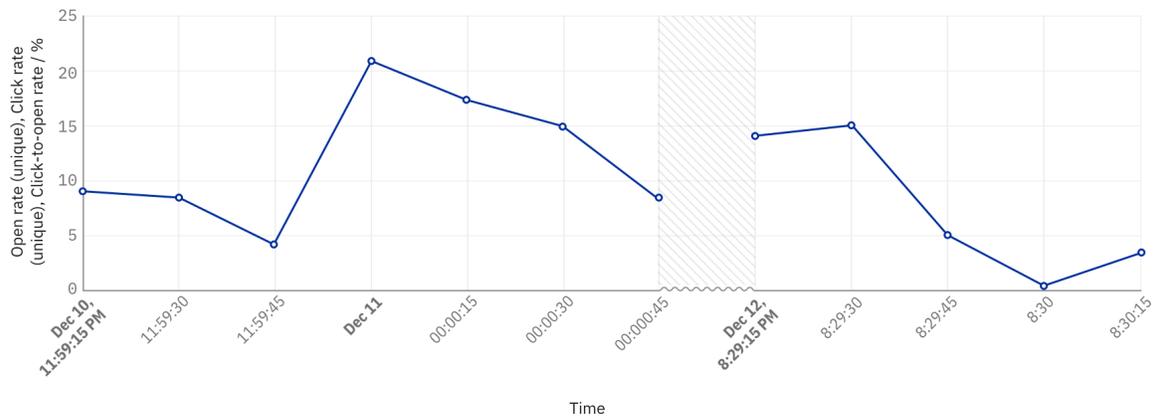


Abb. 13: Beispiel einer Zeitreihe in Intervallen von 15 Sekunden

Legenden

Legenden fassen die unterscheidenden visuellen Eigenschaften wie Farben oder Textur zusammen, die in der Visualisierung verwendet werden. Eine Legende hilft den Betrachtern, die erforderlichen Assoziationen zu erstellen, um das Diagramm zu verstehen.

Verwendung

Vermeidet nach Möglichkeit die direkte Verwendung von Legenden und beschriftet die Daten direkt. Legenden basieren auf visuellen Assoziationen, die das Verständnis eines Diagramms erschweren können.

Diagramme benötigen nicht zwingend eine Legende, wenn sie nur eine Datenkategorie enthalten. Verwendet eine Legende nur, wenn ihr nicht sicher davon ausgehen könnt, dass genügend Platz vorhanden ist, um Beschriftungen direkt anzubringen.

Verwendet eine klare Sprache und vermeidet Akronyme in Legenden. Dies gilt auch für Titel und Achsenbeschriftungen.

Sales by quarter

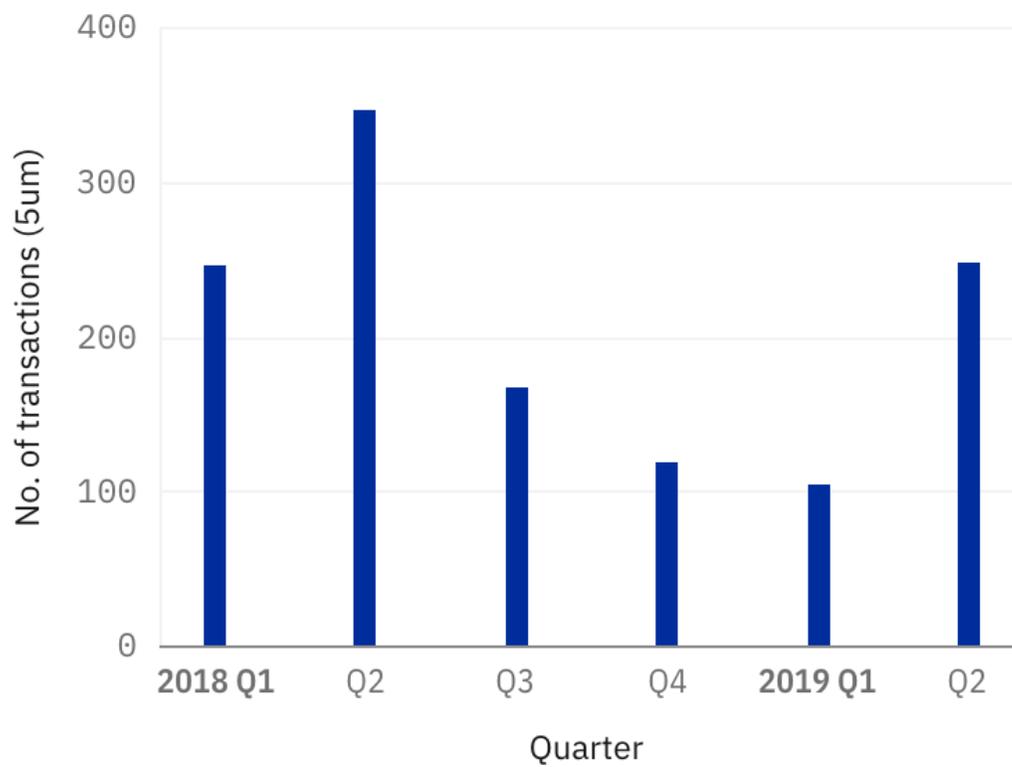


Abb. 14: Entfernt Legenden, um das Diagramm zu vereinfachen, wenn nur eine Datenkategorie benötigt oder nur eine Farbe verwendet wird.

Sales by services

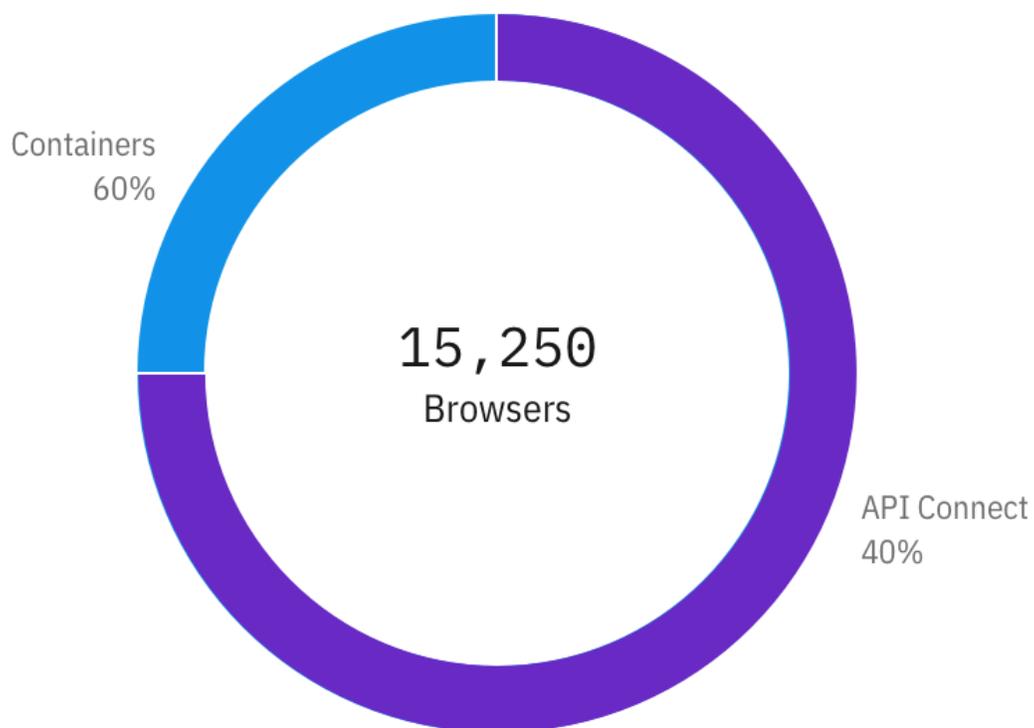


Abb. 15: Beschriftungen in einer Grafik sind ideal für Diagramme mit vorhersehbaren Daten und viel Leerraum, z.B. innerhalb eines Ringdiagramms.

Farbe und Textur

Legenden verwenden üblicherweise Farben zur Unterscheidung der Eigenschaften von Datensätzen und Werten. Textur kann anstelle oder zusätzlich zur Farbe verwendet werden, um Diagramme für Betrachter mit Sehbehinderung zugänglich zu machen.

Clicks by mailing

▣ Opens (unique) ▨ Clicks (unique) ▤ Clicks (unique)

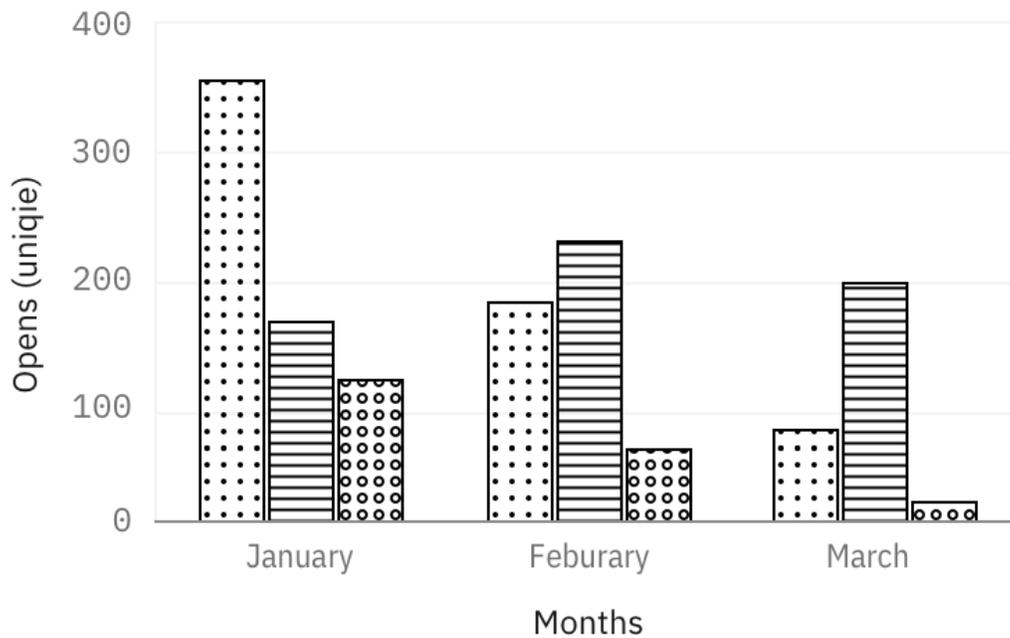


Abb. 16: Texturen können die Zugänglichkeit verbessern.

Position

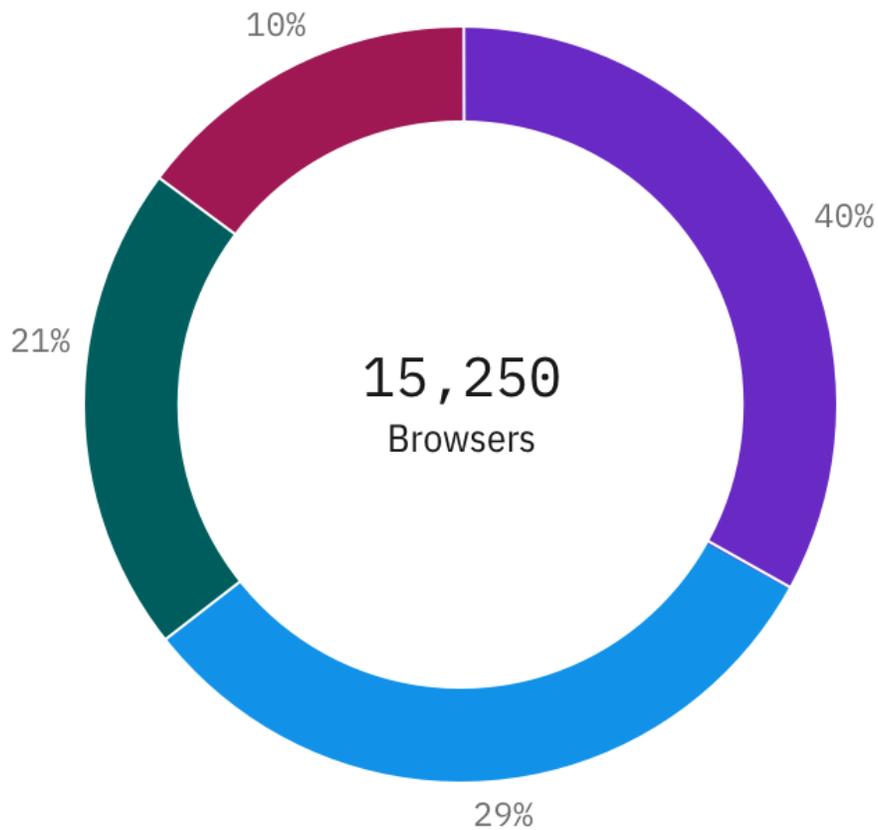
Die Legenden werden üblicherweise unter dem Titel des Diagramms positioniert. Je nach Layout der Seite und Kontext, könnt ihr jedoch die Legenden auch unten, links oder rechts im Grafikrahmen positionieren.

Oben (Standard) und unten

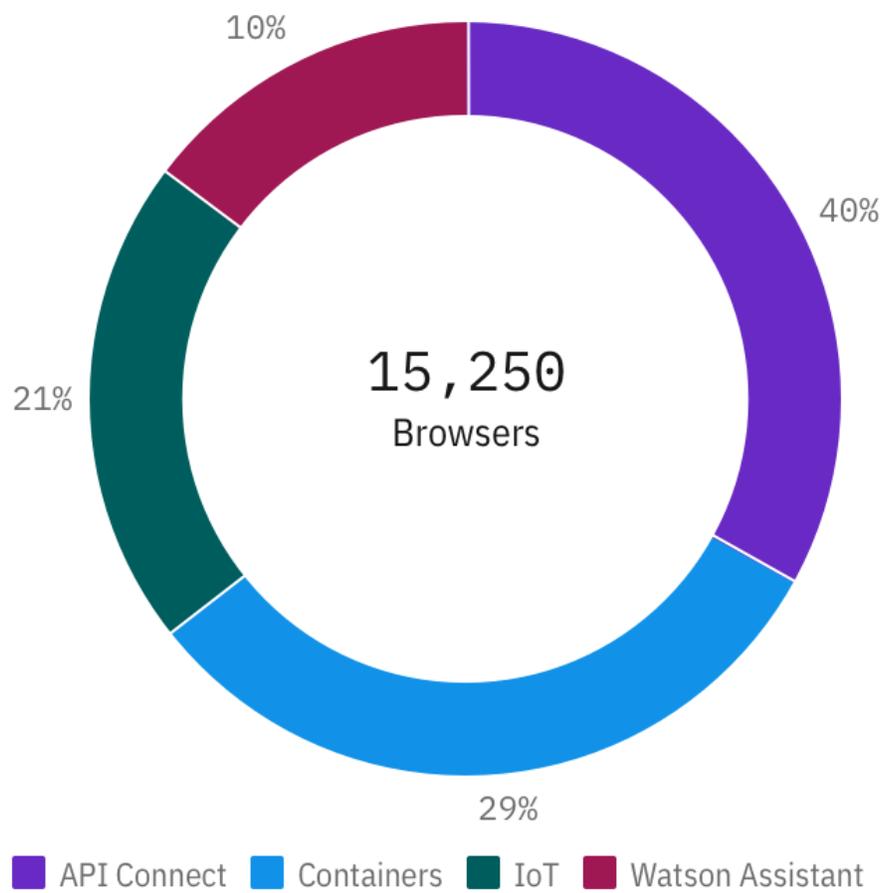
Positioniert die Legende oben oder unten in einem Diagramm, wenn der Platz knapp ist, z.B. in einem Dashboard.

Sales by services

■ API Connect
 ■ Containers
 ■ IoT
 ■ Watson Assistant

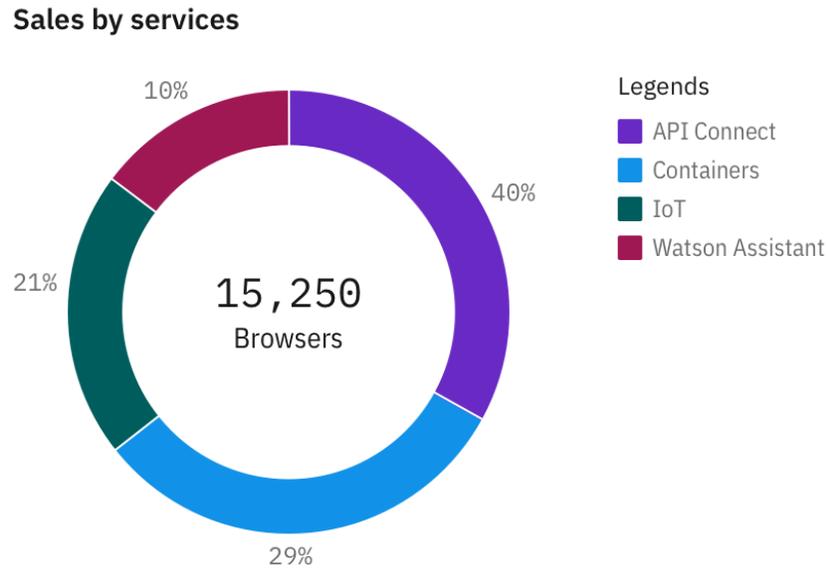


Sales by services



Rechts

Positioniert die Legende rechts neben dem Diagramm, wenn genügend Platz vorhanden ist oder wenn es den Kontext erhöht, z.B. bei gestapelten Balkendiagrammen.



Links

Positioniert die Legende links im Diagramm, wenn eine bessere Schriftausrichtung erforderlich ist. Stellt sicher, dass die umgebenden Elemente des Diagramms nicht zu eng an die Legende heranreichen.

Überlagerung

In Geodiagrammen können Legenden über einen Diagrammrahmen gelegt werden.

Interaktionen

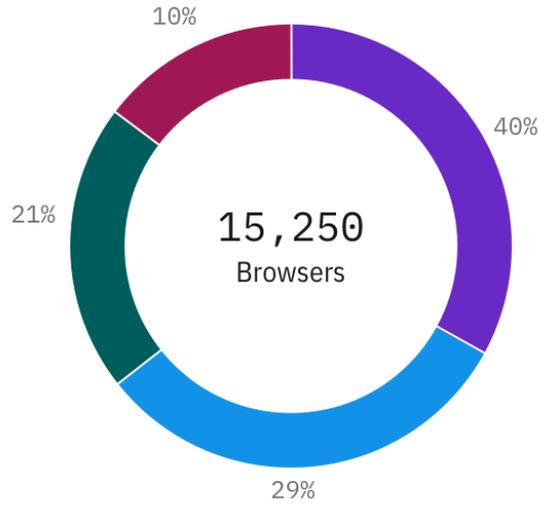
Mauszeiger in der Legende über einer Kategorie

Wenn der Mauszeiger über die Legende einer Kategorie bewegt wird, wird die Deckkraft aller anderen Kategorien im Diagramm auf 30 Prozent gesenkt.

Sales by services

Legends

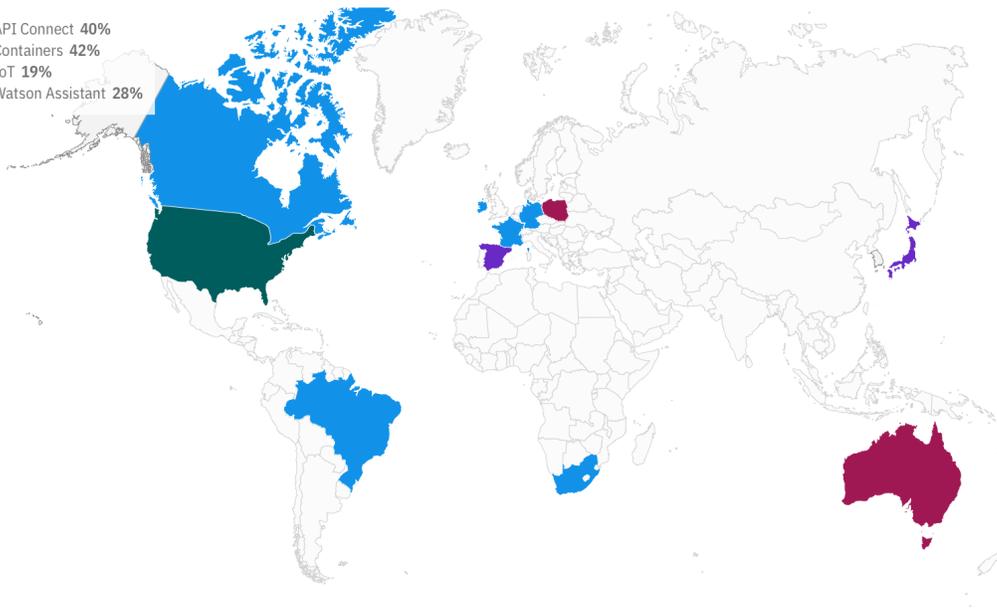
- API Connect
- Containers
- IoT
- Watson Assistant



Optimization rates (overall)

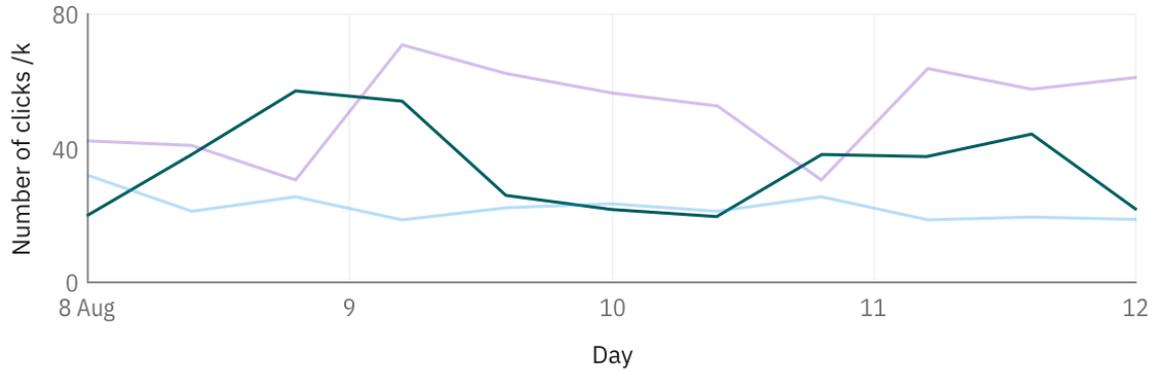


- API Connect 40%
- Containers 42%
- IoT 19%
- Watson Assistant 28%



Customer clicks over time

Open rate
 Close rate
 Cuts

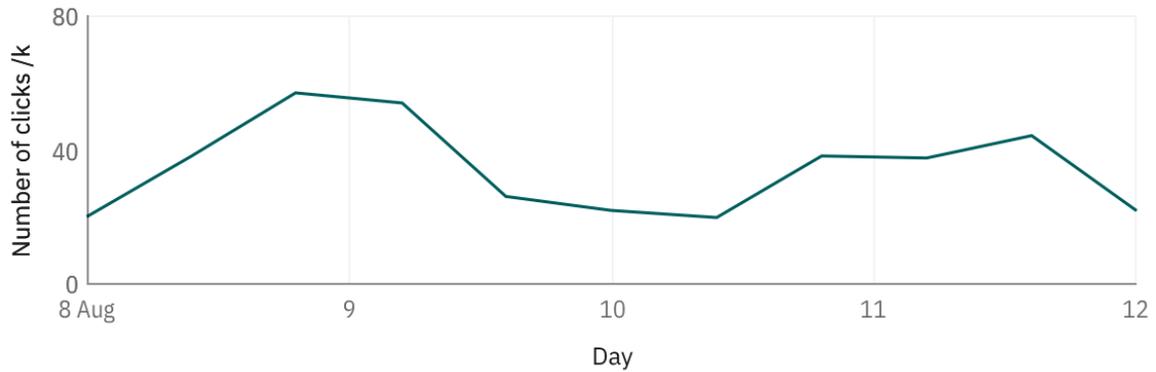


Klicken zum Filtern

Durch Klicken auf die Kategorie einer Legende werden die Informationen isoliert und alle anderen Kategorien ausgeblendet. Die Legende erhält beim Klicken ein Häkchen und wechselt in einen ausgewählten Status.

Customer clicks over time

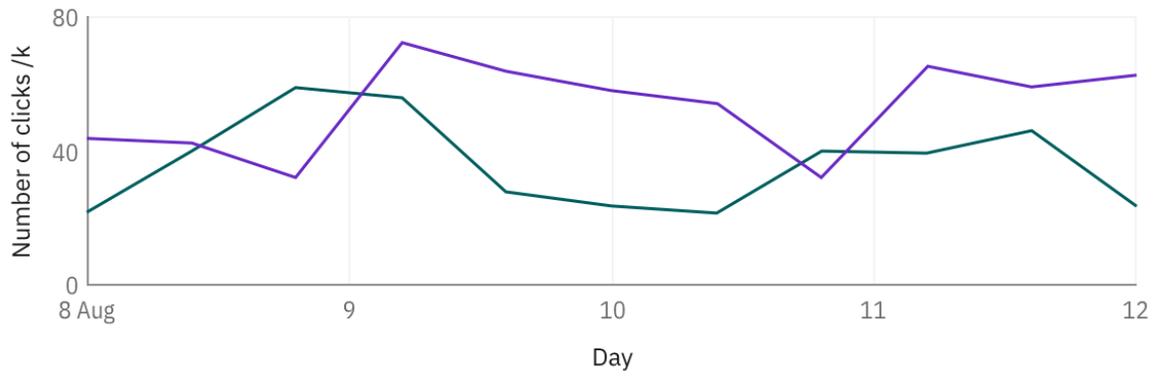
Open rate
 Close rate
 Cuts



Wenn alle Kategorien ausgewählt sind, verschwinden die Häkchen in den Legenden und die Legende wird auf ihren Standardzustand zurückgesetzt.

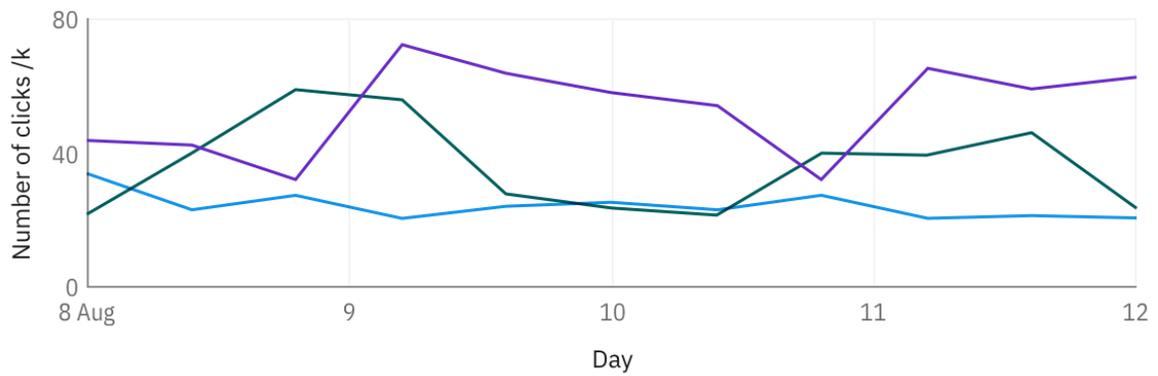
Customer clicks over time

Open rate Close rate Cuts



Customer clicks over time

Open rate Close rate Cuts



Versteckte Legenden

Bitte beachtet, dass sich das Ausblenden von Legenden in Datenvisualisierungen nur sehr selten empfiehlt, es sei denn, es wird nur eine Datenkategorie angezeigt. Dieses Design ist für kleine mobile Displays gedacht, bei denen Legenden den Blick auf die Datenvisualisierung versperren würde.

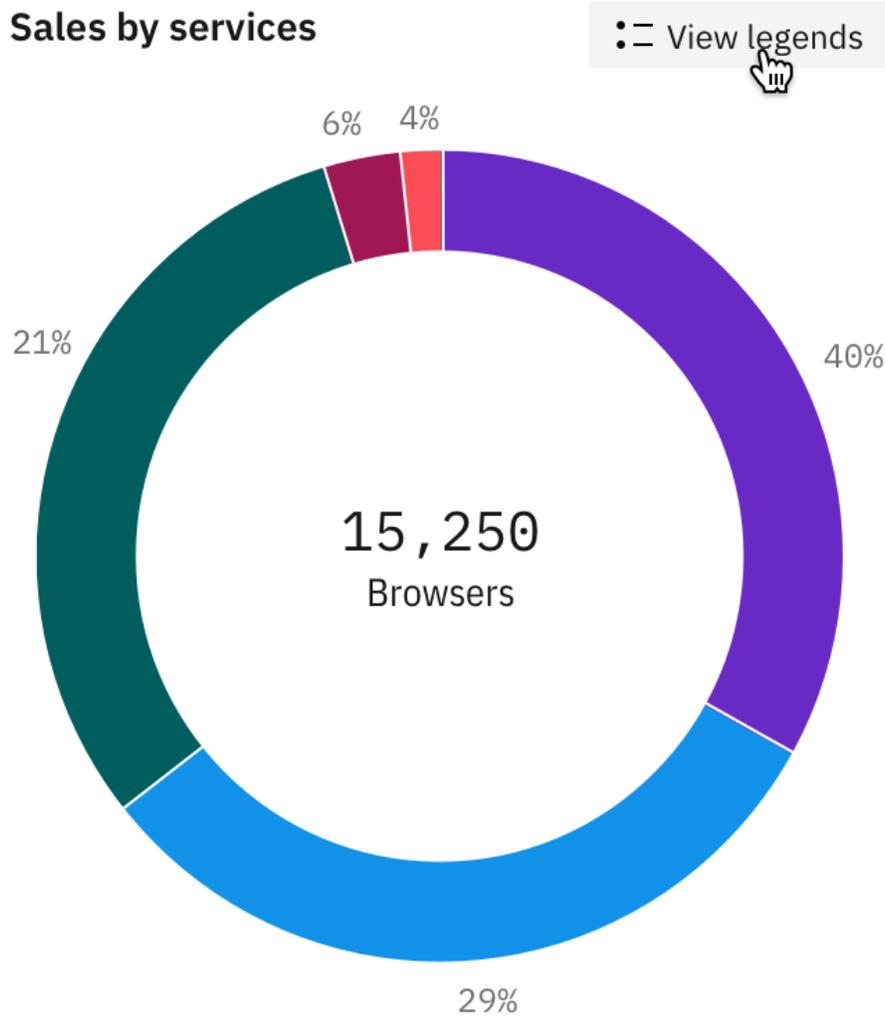


Abb. 17: Wenn Legenden ausgeblendet sind, wird eine Schaltfläche *Legenden anzeigen* hinzugefügt, damit die Legende auch bei der Tastatursteuerung angezeigt werden kann.

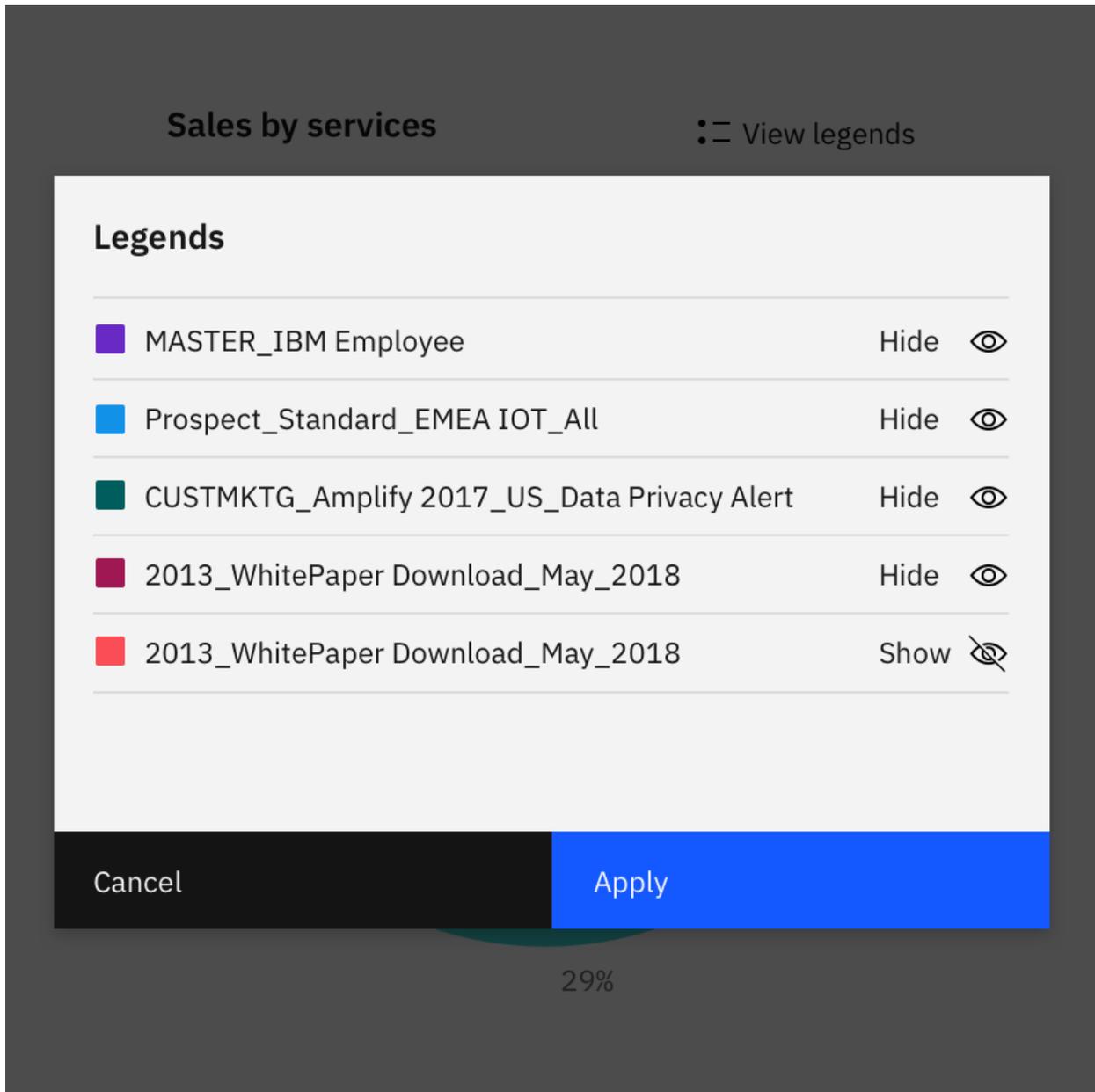


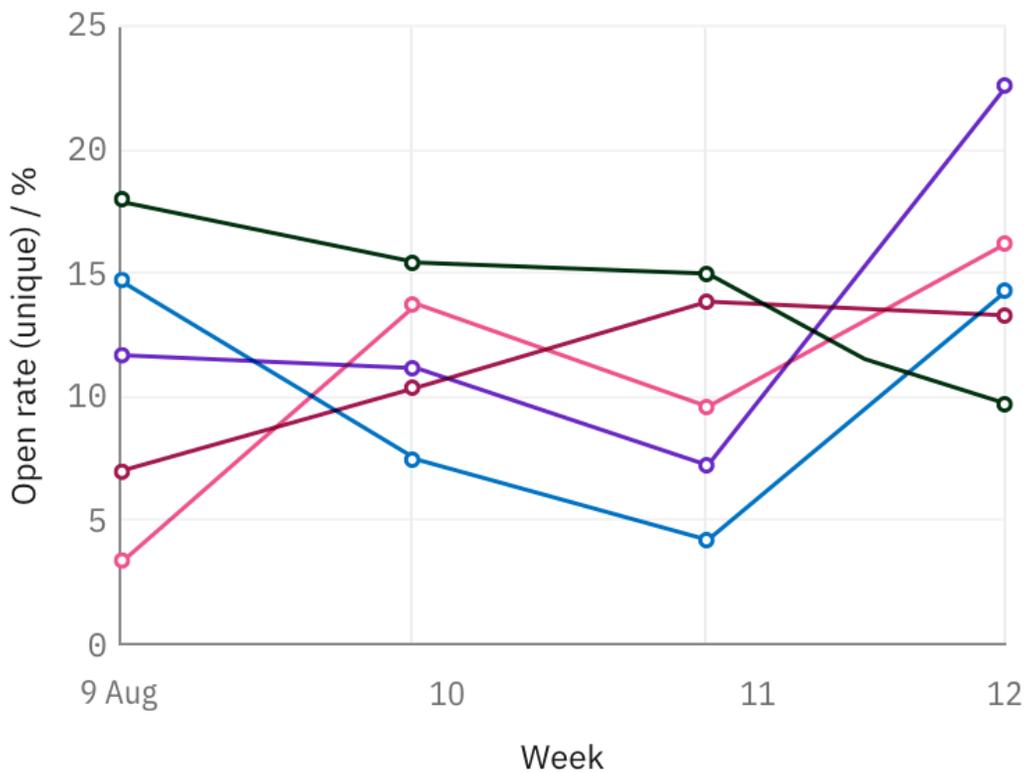
Abb. 18: Wenn ihr auf *Legenden anzeigen* klickt, wird ein modales Fenster mit einer Liste von Legenden mit Optionen zum Ein- und Ausschalten sämtlicher Datenkategorien angezeigt.

Legendenüberlauf

Standardmäßig werden bis zu zwei Legendenzeilen angezeigt. Wenn *Mehr* angeklickt wird, wird der Legendenbereich erweitert, um alle Legenden anzuzeigen. Eine erweiterte Legende sollte jedoch niemals größer als 30 Prozent der Diagrammhöhe sein.

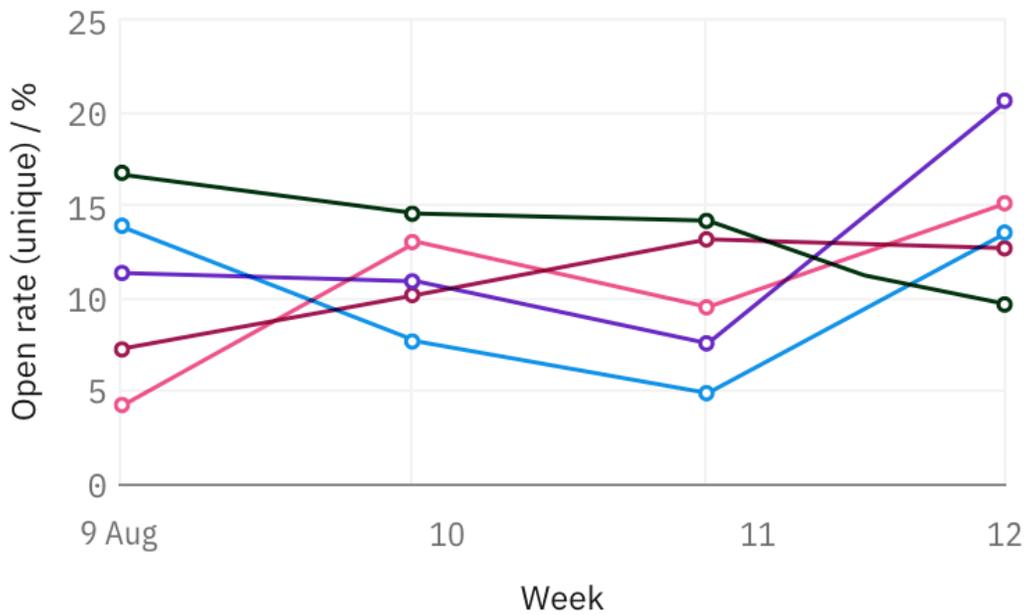
Customer clicks over time

- Open rate
- Click rate (unique)
- Double click a longer title
- Right-click to topen
- Click-to-submit
- [View more](#)



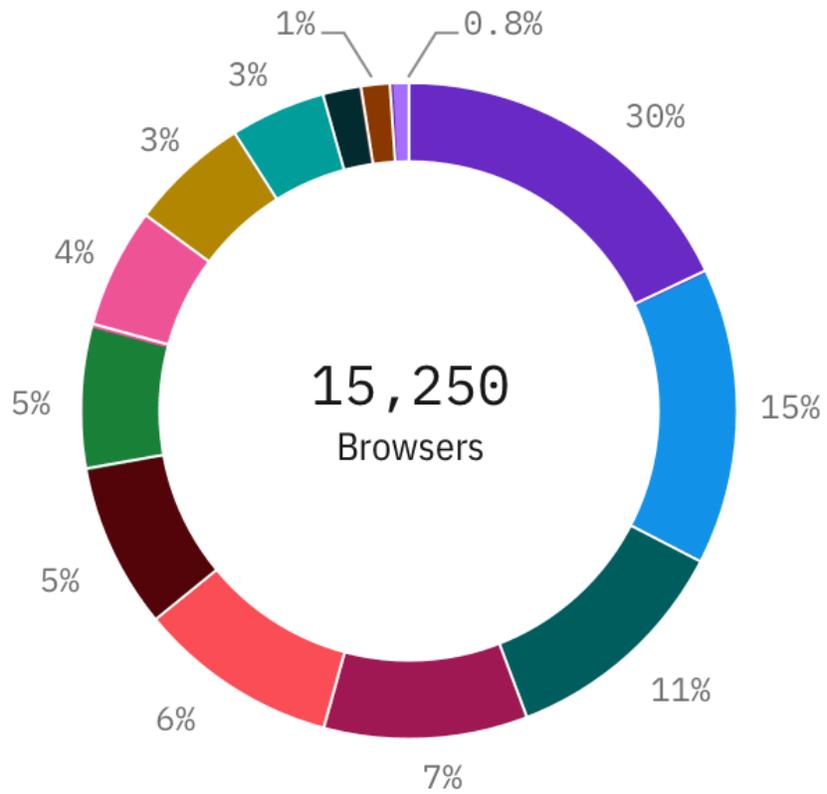
Customer clicks over time

- Open rate
 - Click rate (unique)
 - Double click a longer title
 - Right-click to topen
 - Click-to-submit
 - Double click title very long title
 - Click rate (unique)
 - Right-click
 - Click-to-submit
 - Open rate may have a longer title
 - Double click a longer title
 - Click rate (unique)
- [View less](#)



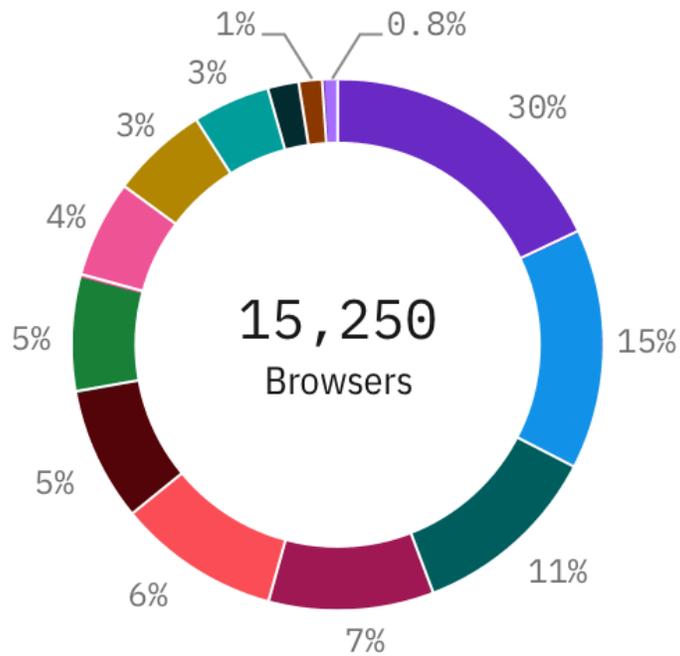
Sales by services

- Open rate
- Click rate (unique)
- Double click
- Watson Assistant
- API Connect
- [View less](#)



Sales by services

- Watson Assistant ■ API Connect
- Double click title very long title ■ Watson Assistant
- Watson Assistant ■ Open rate may have a longer title
- Double click a longer title ■ API Connect
- Longer name here ■ API Connect [View less](#)



Dashboards

Dashboards variieren stark je nach Kontext, für den sie entwickelt wurden. Im folgenden unterscheiden wir zwischen der Präsentation und der Exploration von Daten.

Präsentation-Dashboards

Präsentation-Dashboards zeigen den Betrachtern den aktuellen Status eines Problems mit den relevanten Leistungsindikatoren. Ein gutes Präsentation-Dashboard bietet eine Gesamtansicht der Daten und dient gleichzeitig als Leitfaden für die Betrachter, um zu entscheiden, auf welche Bereiche sie sich konzentrieren sollten um sie weiter zu erkunden.

Vorgehensweise

1. Priorisieren

Priorisiert Daten nach Wichtigkeit und erstellt dann eine klare visuelle Hierarchie. Die wichtigsten Daten sollten den höchsten Kontrast aufweisen und die größte Fläche einnehmen.

Die meisten Menschen im Westen lesen von links nach rechts und dann von oben nach unten. Gemäß dieses **F-förmigen Muster** sollte das Wichtigste oben auf der Seite und anschließend die verbleibenden Elemente platziert werden, wobei mit den weniger wichtigen Informationen abgeschlossen werden sollte.

2. Reduzieren

Unwesentliche Informationen sollten nicht dargestellt werden. Entwerft ein Dashboard so, dass die Komplexität einer Seite so gering wie nötig ist, indem ihr alles entfernt, was Betrachter von der Interpretation der wesentlichen Informationen ablenken könnte.

3. Vereinheitlichen

Verwendet für jeden Datensatz in einem Dashboard immer einheitliche Farben.

4. Gliedern

Unterschiedliche Leerräume trennen oder verbinden Elemente, wodurch die Betrachter durch ein Dashboard geführt werden.

«Layout on a Web page (...) may not measurably influence performance, but it does influence satisfaction.»¹

Exploration-Dashboards

Mithilfe von Exploration-Dashboards können Personen mit den Daten interagieren, um Erkenntnisse zu gewinnen und Muster zu identifizieren. Beispiele für Aktionen, die eine Person an Daten ausführen kann, sind Suchen, Sortieren und Filtern von Daten. Exploration-Dashboards sind für Personen gedacht, die sich interaktiv die Daten erschließen wollen.

¹ (Chaparro, B. S., Shaikh, A. D., and Baker, J. R. (2005). «Reading Online Text with a Poor Layout: Is Performance Worse?» Usability News, 7.

Vorgehensweise

1. Vereinheitlichen

Alle Diagramme sollten dasselbe Layout und denselben Abstand verwenden und Legenden dieselbe Position relativ zum Diagrammbereich haben.

2. Verknüpfte Diagramme

Wenn eine Person ein Diagramm bearbeitet, sollten andere Diagramme mit zugehörigen Datensätzen automatisch aktualisiert werden, um mehrdimensionale Ansichten der Aktion für diese Person bereitzustellen.

3. Anmerkungen

Verwendet Anmerkungen, um Trends, Durchschnittswerte, Spitzen und Täler hervorzuheben und zusätzliche Informationen bereitzustellen. Anmerkungen sollen helfen, Schwankungen in den Daten zu interpretieren. Umgekehrt solltet ihr Anmerkungen vermeiden, die die Ansicht von Daten behindern.

6.2.6 Ressourcen

Entwicklungsressourcen

Design-Ressourcen

Theme-Bibliothek

Farben, Raster und Symbole

Werkzeuge

- [Kontrastrechner](#)

Zum Weiterlesen

- [Alla Kholmatova: Design Systems](#)
- [Christian Beck: The Path to Design System Maturity](#)

Sonderzeichen

., 52

A

API, 34

Authentifizierung und Autorisierung, 34

B

best effort, 34

Beta, 34

C

CLI, 34

D

Domännennamen, 34

I

IP-Adressen, 33

M

Markup (*Substantiv*), 35

R

RFC

RFC 2606, 34

RFC 3849, 33

RFC 5737, 33

RFC 6890, 33

S

SPACK_ROOT, 52

U

Umgebungsvariable

., 52

SPACK_ROOT, 52

und/oder, 36

& (*et-Zeichen*), 34

Kontoname, 35

ssh und SSH, 36

Zugriff, 37

3D, 33

HTTPS, 34

IPsec, 35

Markdown, 35

N/A, 35

URL, der, 36

v, 36

Web-Server, 36

Website, 37

Wi-Fi, 37

1.2.3.4, 33

agnostisch, 34

cons, 34

erlauben, 34

Graylist, 34

Hier klicken, 34

K8s, 35

Man-Power, 35

Master, 35

oberhalb, 35

Pros, 35

regex, 35

Repo, 35

Representational State Transfer, 35

RTFM, 35

Sanity Check, 35

Schwarze Liste, Blacklist, Black-List, 35

Slave, 36

Smartphone, 36

Tarball, 36

tl

dr, 36

untar, unzip, 36

unterhalb, 36

via, 36

Vice versa, 36

vs., [36](#)
Webmaster, [36](#)
White-box, [37](#)
Whitelist, [37](#)