# Barrierefreie Dokumente in Lernkontexten

**Autoren**: Diese Handreichung wird zusammen mit Mitgliedern des Arbeitskreises Bildung der IAAP D-A-CH (Externer Link)
erstellt.

Version: 3.0 (Oktober 2026)

Sie können diese Handreichung als PDF-Datei (Öffnet PDF-Dokument) oder DOCX-Datei (Öffnet Word-Dokument) herunterladen.

## Inhaltsverzeichnis

- Hinweis des BMAS (2 min)
- Vorwort (2 min)
- Versionsänderungen (4 min)
  - Version 2.0
  - Version 2.1
  - Version 2.2
  - Version 3.0
- Hinweise zu allgemeinen Themen
- Hinweise zu Microsoft Office
- Hinweise zu PDF-Dokumenten
- Hinweise zu EPUB-Dokumenten
- Hinweise zu Typst
- Hinweise zu Google Workspace
- Anhang: Verfassende Personen (3 min)
- Anhang: Lizenzinformationen für diese Handreichung (1 min)

# Hinweis des BMAS

Online betrachten

**Ergänzender Hinweis des Bundesministeriums für Arbeit und Soziales:** In dieser Handreichung werden bestimmte Programme und Produkte genannt, die beim barrierefreien Erstellen digitaler Dokumente helfen. Diese Werkzeuge werden nicht nur an Hochschulen, sondern überall dort verwendet, wo digitale Dokumente entstehen – zum Beispiel im Handel oder in der Verwaltung. Ohne den Hinweis auf diese Programme wäre die Erklärung schwer nachzuvollziehen. Die Nennung der Programme bedeutet jedoch keine Empfehlung zum Kauf oder Einsatz. Jede Behörde muss selbst prüfen, welches Angebot am besten zu den Kriterien Wirtschaftlichkeit, Sparsamkeit und Funktionalität passt. Eine konkrete Produktempfehlung ist in dieser Handreichung nicht enthalten.

# Vorwort

Online betrachten

Diese Handreichung wurde von Menschen aus der Praxis für die Praxis geschrieben. Sie soll Orientierung geben zur Erstellung barrierefreier Dokumente.

Unter „Dokumente“ verstehen wir alle elektronischen Medien, die nicht als Teil einer Webseite angezeigt werden (sogenannte „Nicht-Web-Dokumente“ in EN 301 549 Abschnitt 10). Es besteht die gesetzliche Verpflichtung für öffentliche Stellen und Teile der Privatwirtschaft, eigenständige elektronische Dokumente ausschließlich in barrierefreier Form zur Verfügung zu stellen.

Diese Handreichung richtet sich an alle, die mit elektronischen Dokumenten zu tun haben, insbesondere: Sachbearbeitende, Dozierende, IT-Beauftragte, Barrierefreiheitsbeauftragte.

Wir bemühen uns, die Handreichung in barrierefreiem Format zur Verfügung zu stellen. Sollten Sie dennoch Barrieren finden, teilen Sie uns diese bitte per E-Mail mit. Gerne können Sie uns auch Ihr generelles Feedback zum Inhalt dieser Handreichung per E-Mail mitteilen.

# Versionsänderungen

Online betrachten

## Version 2.0

### Neue Beiträge

#### Epub

Neuen Abschnitt zu Epub-Dateien eingefügt

### Bearbeitete Beiträge

#### Hinweise zur Handreichung

Inhaltsverzeichnis grundsätzlich neu aufgebaut, neue Struktur der Handreichung zur besseren Übersichtlichkeit implementiert, Auflistung der Links angepasst mit geänderter Überschrift „Verweise“

#### ms-excel.html

Artikel um weitere Barrierefreiheitsaspekte ergänzt

#### ms-word.html

Änderungen bei „Anwendungsfälle“, „Werkzeuge“ und „Hinweise zur Erstellung von Word-Dokumenten“

#### pdf-dokumente.html

Änderungen bei „Hinweise zur Erstellung von PDF-Dokumenten“ und „Verweise“, Ergänzung von Informationen zum PDF/UA-2 Standard

#### tools.html

Ergänzung des Abschnitts zu veraPDF

#### adobe-indesign.html

Ergänzung des Kapitels „Hinweise zur Erstellung von barrierefreien PDFs aus InDesign“

#### elektronische-signatur.html

Aus dem Kapitel PDF-Formulare gelöscht
und eigenes Kapitel erstellt

### entscheidungshilfepdfkonverter.html

Unterscheidung, welche Testfälle automatisiert geprüft werden (Maschine) und welche Testfälle vom Menschen zu prüfen sind (Mensch)

### apache-fop.html

Ergänzung einer grundlegenden Abhilfe bezüglich der Spezialfälle Ligaturen und Silbentrennung

### allgemein.html

Struktur und Formulierungen des Kapitels überarbeitet

## Version 2.1

Probleme mit der Überschriftenhierarchie gelöst

## Version 2.2

### Neue Beiträge

Keine

### Bearbeitete Beiträge

### tools.html

Überarbeitung der Abschnitte zu axesPDF und veraPDF

## Version 3.0

### Neue Beiträge

### hinweis-bmas.html

Hinweise des BMAS zur Einordnung von Produktnennungen ergänzt

### vorlagenmanagement.html

Neuer Beitrag zu Hinweisen zur Implementierung eines Vorlagenmanagements ergänzt

### jspdf.html

Neuer Beitrag zum Einsatz eines eigenständigen Forks auf Basis von jsPDF für die Erstellung barrierefreier PDF/UA-Dokumente

### inhalt-hinweise-typst.html

Neue Beiträge zum Einsatz von Typst als Textsatzsystem in Bezug auf die Erstellung barrierefreier PDF/UA-Dokumente

### inhalt-hinweise-google-workspace.html

Neue Beiträge zum Einsatz von Google Workspace Produkten für die Erstellung barrierefreier PDF-Dokumente

## Bearbeitete Beiträge

### allgemein.html

Hinweise zu Kopf- und Fußzeilen ergänzt Hinweise zur barrierefreien Gestaltung von Dokumenten umformuliert.

### alternativtexte.html

Anmerkung zur Allgemeinen Einordnung ergänzt; Verweise aktualisiert

### tools.html

Informationen zur KI-Prüfung im Abschnitt zu PDF Accessibility Checker (PAC) ergänzt

### tabellen.html

Überschrift „Beispiel aus der Praxis“ spezifiziert: „(Negatives)“; Anmerkung zu den Kontrasten präzisiert; Abschnitt „Anforderungen zur Barrierefreiheit von Tabellen in PDF-Dokumenten“ aktualisiert

### ms-word.html

Tipps zur Erstellung von Dokumentvorlagen ergänzt; Tipps zur Erstellung eines Dokuments unter Berücksichtigung der Barrierefreiheitsanforderungen ergänzt

### pdf-dokument.html

PDFix in der Liste von Anwendungen zur Nachbearbeitung von PDF ergänzt; ergänzende Hinweise zu Lesezeichen; neuer Abschnitt zu Schriften

### apache-fop.html

Ergänzung im Hinblick auf Aufzählungszeichen sowie die Sprachauszeichnung für einzelne Teilbereiche und Unterschiede zu jsPDF; neuer Abschnitt zu Schriften und Font-Embedding; Verweise aktualisiert

# Hinweise zu allgemeinen Themen

Online betrachten

Im Folgenden sind alle Hinweise zu allgemeinen Themen aufgelistet.

## Inhaltsverzeichnis


- Allgemeine Anforderungen an die Barrierefreiheit von Dokumenten (12 min)
  - Allgemeine Einordnung
  - Hinweise zum Dateiformat
  - Hinweise zum Dokumenttitel
  - Hinweise für die Gestaltung und das Layout
  - Alternativtexte für grafische Darstellungen
  - Struktur der Inhalte
  - Hinweise für Sprache und Text
  - Verweise
- Hinweise zur Implementierung eines Vorlagenmanagements (12 min)
  - Einleitung
  - Bedeutung von Corporate Design (CD)
  - Prozess für das Vorlagenmanagement
  - Verweise
- Überblick über Inhaltstypen und Formate (7 min)
  - Anwendungsfälle
  - Strukturierter Text
  - Formular
  - Komplexe Modelle und Strukturen
  - Tabellenkalkulation
  - Video
  - OCR
  - Verweise
- Hinweise zur Erklärung zur Barrierefreiheit (3 min)
  - Allgemeine Einordnung
  - Verweise
- Hinweise zur Erstellung von Alternativtexten (12 min)
  - Allgemeine Einordnung
  - Grundsätzliche Überlegungen
  - Anwendungsfälle im mathematisch-naturwissenschaftlichen Bereich
  - Ausblick
  - Verweise
- Hinweise zur Gestaltung von Tabellen (8 min)
  - Allgemeine Einordnung
  - Grundsätzliche Überlegungen
  - Maßnahmen
  - Anforderungen zur Barrierefreiheit von Tabellen in PDF-Dokumenten
  - Verweise
- Hinweise zu mathematischen Darstellungen (5 min)
  - Allgemeine Einordnung
  - Mathematische Informationsvermittlung

    - Mathematik im Lern- und Lehrkontext
    - Verweise
- Hinweise zu Tools zur Überprüfung der Barrierefreiheit (11 min)
    - Allgemeine Einordnung
    - Prüfung der Barrierefreiheit in Microsoft Office-Dokumenten
    - Prüfung von PDF-Dokumenten
    - Verweise
- Produktionsprozess für barrierefreie Lernmaterialien (4 min)
    - Allgemeine Einordnung
    - Vorteile des Verfahrens
    - Nachteile des Verfahrens
    - Verweise

# Allgemeine Anforderungen an die Barrierefreiheit von Dokumenten

Online betrachten

## Allgemeine Einordnung

Barrierefreie digitale Dokumente müssen die vier Prinzipien der Web Content Accessibility Guidelines (WCAG) des W3C bzw. die entsprechenden Anforderungen aus Abschnitt 10 der EN 301 549 erfüllen. Für PDF-Dokumente gilt zusätzlich der PDF/UA-Standard (PDF/UA-1: 14289:2014 bzw. neu PDF/UA-2: ISO 14289–2:2024).

Die WCAG formuliert die folgenden Prinzipien:

### 1. Wahrnehmbar

- nicht-textuelle Inhalte: Textalternativen für nicht-textuelle Inhalte bereitstellen
- Medien auf Zeitbasis: Medien auf Zeitbasis (Video/Audio/Streams) mit Textalternativen bereitstellen
- anpassbar: das Layout kann für unterschiedliche Bedarfe umgestellt werden
- unterscheidbar: Vorder- und Hintergrund müssen gut zu unterscheiden sein (Kontraste)

### 2. Bedienbar

- Tastaturerreichbarkeit: alle Steuerelemente müssen per Tastatur erreichbar sein
- genug Zeit: genug Zeit für Eingaben lassen
- navigierbar: Mittel zur Navigation, Orientierung auf der Seite und Suchfunktionen bereitstellen
- mehrere Eingabemöglichkeiten für Formulare etc. über die Tastatur hinaus bereitstellen

### 3. Verständlich

- lesbar: der Text ist lesbar und verständlich
- vorhersehbar: Seiteninhalte erscheinen und funktionieren in vorhersehbarer Weise
- Eingabeassistenten: Unterstützung von Fehlererkennung, -vermeidung und -korrektur

### 4. Robust

- kompatibel: Unterstützung verschiedener Schnittstellen und Assistenzsysteme
- Vermeiden von Skriptfehlern in digitalen Dokumenten, da diese die Darstellung verhindern oder beeinträchtigen

Barrierefreies Webdesign: Die vier Prinzipien der Web Content Accessibility Guidelines (WCAG)

Die Grundlage für ein barrierefreies Dokument wird bereits bei der Erstellung gelegt. Werden die Anforderungen an die Barrierefreiheit von Beginn an berücksichtigt, lassen sich Dokumente meist ohne großen Mehraufwand erstellen. Eine nachträgliche Überarbeitung ist dagegen häufig deutlich aufwändiger und kann je nach Dokument erhebliche Nacharbeiten erfordern.

# Hinweise zum Dateiformat

## Hinweise zur Auswahl des Dateiformates

Die Auswahl des Dateiformats sollte entsprechend der Zielgruppe und des verfolgten Ziels des Dokuments erfolgen.

Folgende Fragen können bei der Entscheidung für ein Dateiformat helfen:

- Handelt es sich um ein Dokument, in welchem Layout und Gestaltung explizit relevant sind, wird es beispielsweise für den Druck erstellt?
- Geht es im Wesentlichen um die Weitergabe der enthaltenen Informationen unabhängig vom Dateiformat an die Zielgruppe?

## Hinweise zur Dokumentenstruktur

Verallgemeinert gesagt setzt sich die Mehrzahl von Dateiformaten für Dokumente aus zwei Ebenen zusammen: der sichtbaren Präsentationsebene und der versteckten Strukturebene. Die Präsentationsebene eines Dokuments wird ausgedruckt. Die Strukturebene enthält im Hintergrund Meta-Informationen für die sichtbaren Dokumentelemente, welche nicht ausgedruckt werden. So ist anhand der Formatierung eine Überschrift erkennbar. In der Strukturebene ist dieser Text direkt als Überschrift gekennzeichnet. Dies gilt für HTML, Word-Dokumente (DOCX) und auch PDF. Die Zuordnung der Dokumentelemente zu ihrem Typ oder Tag muss während der Dokumenterstellung mit der Anwendungssoftware z. B. über Formatvorlagen durchgeführt werden. Wenn Überschriften optisch wie Überschriften aussehen, aber in der Strukturhierarchie nicht korrekt zugeordnet werden, können diese Texte nicht als Überschriften interpretiert werden. Auf der Strukturebene gelten solche Texte nicht als Überschriften.

## Hinweise zu gescannten Dokumenten (Bilder ohne OCR)

Die Weitergabe eingescannter Dokumente ohne eine Texterkennung sollte vermieden werden. Es handelt sich hierbei um Dokumente, welche ohne Überarbeitung nur eine Präsentationsebene haben. Eingescannte Dokumente ohne Texterkennung entsprechen grundsätzlich nicht dem Stand der Technik. Sie enthalten in der Regel lediglich ein Bild der Dokumentseite und keine maschinenlesbaren Text- oder Strukturinformationen. Dadurch können Inhalte weder durchsucht noch von assistiven Technologien wie Screenreadern zuverlässig erfasst werden.

## Hinweise zu grundlegenden Dokumenteninformationen

Es sollten grundlegende Informationen zum Inhalt der Datei in den Metainformationen der Datei hinterlegt werden. Dazu gehören Angaben zur verfassenden Person oder Instituteon, das Erstellungsdatum, die verwendete Sprache und auch der Status der Datei zur Barrierefreiheit. In PDF sind die Metadaten Dokumenttitel und Dokumentsprache verpflichtend.

## Hinweise zum Dateiexport / zur Dateikonvertierung

Wenn Dateien in andere Dateiformate exportiert oder konvertiert werden sollen, muss im Zielformat die Barrierefreiheit erneut überprüft und gegebenenfalls korrigiert werden. Wenn die Quelldatei bereits barrierefrei ist, ist es am Einfachsten im Zielformat Barrierefreiheit zu gewährleisten.

## Hinweise zur Sprachauszeichnung

Für Dateien muss die verwendete Sprache des Inhalts angegeben werden, da Screenreader diese bei der Sprachausgabe verwenden. Enthaltene fremdsprachige Dokumentbereiche sollen in der Strukturebene mit der entsprechenden Sprache gekennzeichnet werden, wenn das Dateiformat dies ermöglicht.

Das Hochschulforum für Digitalisierung gibt in dem Beitrag „Barrierefreie Dokumente mit Markdown, Latex und PDF erstellen“ zu den entsprechenden Dateiformaten vertiefende Informationen.

## Hinweise zum Dokumenttitel

Vergeben Sie einen aussagekräftigen Dokumenttitel, der Thema oder Zweck des Dokuments zutreffend beschreibt. Auch der Dateiname sollte sprechend gewählt werden, weil er das Erste ist, was die Nutzenden wahrnehmen.

## Hinweise für die Gestaltung und das Layout

- Lesefreundliche Schriftart: Verwenden Sie eine gut lesbare Schrift. Je nach Art des Dokuments muss die Schriftgröße entsprechend gewählt werden. Sowohl sehr feine oder fette, als auch sehr schmale oder breite Schriftarten oder Schriftschnitte sollten vermieden werden.
- Sonderformatierungen sollten vermieden werden. Kursiv, Versalien (GROSSBUCHSTABEN), Unterstreichung und Schmuckschriftarten verschlechtern die Lesbarkeit.
- Texte sollten linksbündig, einzeilig gesetzt werden. Wir empfehlen einen Flattersatz und nicht Blocksatz zu verwenden.
- Es sollten keine leeren Absätze verwendet werden, um Abstand zwischen Textbereichen zu halten. Um dies zu verändern, sollte die Formatierung „Absatzabstand“ benutzt werden. Benutzen Sie für horizontale Abstände Tab-Stopps statt Leerzeichen.
- Der Farbkontrast zwischen dem Hintergrund und der Textfarbe muss beachtet werden. Schwarze Schrift ist auf weißem Hintergrund gut lesbar. Aufgrund der Rot-Grün-Schwäche vieler Menschen sollte eine Rot-Grün-Farbkombination vermieden werden. Dies gilt auch für Komplementärfarben (Blau-Orange, Gelb-Violett), da diese zu Flimmereffekten führen. Bei farbigen Hintergründen sollte der Kontrast möglichst hoch sein. Das kostenfreie Tool Colour Contrast Analyser prüft Kontrastverhältnisse.
- Silbentrennung sollte vermieden werden. Harte, weiche und automatische Silbentrennung führt bei der hörbaren Ausgabe von Text häufig zum Vorlesen von zwei Wörtern.
- Informationen sollten nicht nur durch Farbe vermittelt werden. Falls dies notwendig ist, sollte versucht werden, dies zusätzlich auf andere Art und Weise deutlich zu machen.

Weitere Details zu der Formatierung von Text finden Sie unter leserlich.info.

## Alternativtexte für grafische Darstellungen

Informative Grafiken, Bilder und Diagramme müssen einen Alternativtext haben, welcher die sichtbare Abbildung textuell wiedergibt. In das Beschreibungsfeld sollte eine kurze Erläuterung der Abbildung geschrieben werden. Bei Diagrammen sollten die wesentlichen Aussagen textlich beschrieben werden. Längere Erläuterungen der Abbildung gehören in den Fließtext. Reine Schmuckelemente, wie Striche oder Kästen, sollten keinen Alternativtext erhalten und als Schmuckelement gekennzeichnet werden. Abbildungen, welche nicht im Zusammenhang mit dem Text und der inhaltlichen Aussage stehen, sollten vermieden werden. Abbildungen sollten mit der Umbruchart „Mit Text in Zeile“ positioniert werden. Der vom Projekt iBoB herausgegebene Praxisleitfaden zur Erstellung textbasierter Alternativen für Grafiken „Gut fürs Image!“ und die Richtlinien zur Umsetzung taktiler Grafiken liefern Hinweise für Formulierungen

von Alternativtexten. Für grafisch dargestellte Formeln sind alternative Textbeschreibungen hinterlegt. Siehe z. B. Anleitung „Mathematische Formeln vorlesen, erstellen und zugänglich machen“ des Regionalen Rechenzentrums Erlangen (RRZE) im PDF-Format.

Weitere Details zu Alternativtexten finden Sie im Abschnitt Hinweise zur Erstellung von Alternativtexten.

## Struktur der Inhalte

### Hinweise zu Überschriften

Zur Kennzeichnung von Überschriften sind die entsprechenden Formatvorlagen zu verwenden. Hierbei sollten keine Ebenen übersprungen werden: Überschriftenebenen sind streng hierarchisch anzuwenden. Sollte die Anwendung, mit welcher das Dokument erstellt wird, keine Formatvorlagen unterstützen, verwenden Sie für Überschriften einer Ebene einheitlich eine Schriftart und Schriftgröße.

### Hinweise zu Listen

Für Listen verwenden Sie bitte die Aufzählungszeichen und Nummerierungen der Anwendung. Selbst erstellte Zeichen oder römische Ziffern stellen eine besondere Herausfordung in der Listenerstellung oder im Verständnis dar.

### Hinweise zu Tabellen

Tabellen erstellen Sie bitte mit den angebotenen Möglichkeiten der genutzten Anwendung. Zeilen und Spalten sollten entsprechend definierte Überschriftenzellen haben. Diese Überschriften sollten bei mehrseitigen Tabellen wiederholt werden. Legen Sie bei der Tabellenkonzeption besonderes Augenmerk auf verbundene Zellen, mehrere Überschriftenebenen, leere Tabellenzellen oder Tabellen in Tabellen, da diese besondere Herausforderung darstellen. Passen Sie die Tabellenkomplexität an die Zielgruppe an. Siehe auch die Hinweise zur Gestaltung von Tabellen.

### Hinweise zu Kopf- und Fußzeilen

Kopf- und Fußzeilen sollten keine Informationen enthalten, die ausschließlich dort vorkommen und für das Verständnis des Dokuments erforderlich sind. Inhaltlich wichtige Informationen gehören in den Fließtext des Dokuments.

Seitenzahlen, Dokumenttitel, Dateinamen oder andere wiederkehrende Informationen können sinnvoll in Kopf- oder Fußzeilen platziert werden. Sie erleichtern insbesondere bei ausgedruckten Dokumenten die Orientierung. Bei barrierefreien PDF-Dokumenten sollten Kopf- und Fußzeilen als Artefakte gekennzeichnet werden. Dadurch werden sie von assistiven Technologien, wie Screenreadern, nicht bei jeder Seite erneut vorgelesen und erscheinen nicht als Bestandteil der semantischen Dokumentstruktur.

Dies gilt auch dann, wenn Kopf- oder Fußzeilen Text enthalten. Beim Export oder der Konvertierung in andere Dateiformate sollte geprüft werden, ob die Dokumentstruktur und die erforderlichen Informationen vollständig übernommen werden.

### Hinweise zu weiteren Aspekten

- Längere Dokumente sollten immer ein Inhaltsverzeichnis und gegebenenfalls auch ein Glossar enthalten.
- Die Lesereihenfolge sollte bei der Erstellung von Dokumenten beachtet und überprüft werden.

- Fuß-, Endnoten und Verweise sollten mithilfe der Funktionen der Anwendung erstellt werden.

## Hinweise für Sprache und Text

Jeder Text sollte eine Sprache verwenden, die für die angesprochene Zielgruppe verständlich ist. Das kann eine fachspezifische Sprache oder auch „Leichte Sprache“, „Einfache Sprache“, „Verständliche Sprache“ oder „Leicht Lesen“-Text sein. Beachten Sie, dass die genannten Begriffe nicht bedeutungsgleich sind. Was unter „Leichter Sprache“ zu verstehen ist, kann auf der Seite des capito Netzwerks nachgelesen werden.

Weitere Hinweise unter folgenden Regelwerken und Leitfäden:

- Berliner Standards für verständliche Sprache - allgemeiner Leitfaden
- Berliner Standards für barrierefreie Sprache und Texte, Regelwerk für die Berliner Verwaltung - bisher veröffentlichte Standards zum Download
- Netzwerk Leichte Sprache
- Standards für Leichte Sprache von Inclusion Europe in mehreren Sprachen
- DIN ISO 24495-1 Einfache Sprache - Teil 1: Grundsätze und Leitlinien (ISO 24495-1:2023)

## Verweise

### Interne Links

- Hinweise zur Erstellung von Alternativtexten
- Hinweise zur Gestaltung von Tabellen

### Externe Links in diesem Artikel

- Web Content Accessibility Guidelines (W3C)
- PDF/UA-1: 14289:2014 (ISO)
- PDF/UA-2: ISO 14289–2:2024 (ISO)
- Die vier Prinzipien der WCAG 2.2 (Jan Hellbusch)
- Barrierefreie Dokumente mit Markdown, LaTeX und PDF erstellen (Hochschulforum für Digitalisierung)
- Color Contrast Checker (vispero, vormals TPGI)
- Informationen zur Darstellung von Texten (Deutscher Blinden- und Sehbehindertenverband e. V.)
- Praxisleitfaden zur Erstellung textbasierter Alternativen für Grafiken (Anja Fibich, Frauke Onken & Christian Axnick, iBoB - Projekt inklusive berufliche Bildung ohne Barrieren)
- Richtlinien zur Umsetzung taktiler Grafiken (TU Dresden)
- Anleitung zu mathematische Formeln (Gunther Heintzen, Regionales Rechenzentrum Erlangen (RRZE))
- Leichte Sprache - Begriffe, Regeln und Erklärungen (capito corp.)
- Berliner Standards für verständliche Sprache - allgemeiner Leitfaden (berlin.de)
- Berliner Standards für barrierefreie Sprache und Texte, Regelwerk für die Berliner Verwaltung - bisher veröffentlichte Standards zum Download (berlin.de)
- Regeln und Informationen zur Leichten Sprache (Netzwerk leichte Sprache e. V.)
- Standards für easy-to-read auf EU-Ebene (Inclusion Europe)
- DIN ISO 24495-1 Einfache Sprache - Teil 1: Grundsätze und Leitlinien (ISO 24495-1:2023) (DIN e. V.)

## Weiterführende Links

- Videotutorial zur Erstellung barrierefreier Dokumente in Word, Checkliste für Dokumente zum Download (Landeskompetenzzentrum für barrierefreie IT Hessen)
- Vortrag bei der vBIB24 – Digitale Teilhabe als Videobeitrag mit allgemeiner Einführung in die Thematik (Tobias Roppelt)

# Hinweise zur Implementierung eines Vorlagenmanagements

Online betrachten

## Einleitung

Über die Zeit sammeln sich in einer Organisation viele verschiedene Vorlagen für Dokumente an. Historisch werden diese oftmals nicht an zentralen Stellen abgelegt, oder Mitarbeitende legen sich eigene Kopien davon an. Eine Folge davon ist, dass es für die Organisation sehr schwer ist, eine hohe Qualität der Dokumente bezogen auf Barrierefreiheit, jedoch auch auf alle anderen Aspekte, wie verwendete Schriftarten, Logos und andere Punkte, zu gewährleisten.

Um barrierefreie Dokumente in öffentlichen Stellen gemäß der BITV 2.0 sicherzustellen, empfiehlt es sich, bereits barrierefreie, zentral bereitgestellte Dokumentvorlagen innerhalb der Organisation einzusetzen. Sie schaffen die Grundlage zur Einhaltung gesetzlicher Anforderungen, fördern Chancengleichheit und verbessern die Nutzungsfreundlichkeit von Dokumenten für alle Menschen. Durch vordefinierte barrierefreie Formateinstellungen lassen sich Arbeitsprozesse vereinfachen, automatisieren und somit beschleunigen, die Qualität von Dokumenten verbessern und das Corporate Design wahren.

## Bedeutung von Corporate Design (CD)

Damit Dokumente am Ende den Wünschen und Vorgaben der jeweiligen Organisation entsprechen und deren Look and Feel widerspiegeln, muss das Corporate Design der jeweiligen Organisation verwendet werden. In der Erarbeitung der Vorgaben des CD muss Barrierefreiheit von Beginn an berücksichtigt werden. Hier werden beispielsweise Farbpaletten, Schriftarten, Schriftgrößen, Logos und andere zentrale Elemente definiert. Wichtig ist, dass hier sichergestellt wird, dass die Kontrastverhältnisse, bzw. der Mindeskontrastabstand eingehalten wird. Im CD sollten gut lesbare Schriften verwendet werden. Merkmale gut lesbarer Schriften sind große Öffnungen, starke Haarlinien und die Reichweite der Buchstaben, sodass die Buchstaben gut erkennbar und unterscheidbar sind. Wenngleich serifenlose Schriften oft als besser lesbar gelten als Serifen-Schriften, können auch Serifen-Schriften diese Merkmale aufweisen und gut lesbar sein. (Hinweis: Die für Bundesbehörden zur Verfügung stehenden Hausschriften – BundesSans und BunderSerif – zeichnen sich beide durch eine sehr gute Lesbarkeit aus.) Weiterhin empfohlen sind mindestens 11 pt Schriftgröße mit einem Zeilenabstand von 1,2 bis 1,5 und Linksbündigkeit. Versalienschreibweise sollte generell vermieden werden. Eine Änderung des CDs ist in der Regel sehr zeitaufwändig und wird daher nur in längeren Zyklen durchgeführt. Dies bedeutet, dass bei einer zu späten Prüfung des CDs auf Barrierefreiheit eine Erstellung barrierefreier Dokumente nicht mehr möglich ist, ohne einzelne Vorgaben des CDs zu „ignorieren“. Beziehen Sie daher Fachpersonen für Barrierefreiheit in CD-Prozesse ein, um die Grundlage für barrierefreie Dokumente zu legen.

## Prozess für das Vorlagenmanagement

Zu Beginn empfiehlt sich die Klärung von Zuständigkeiten für Entwicklung, Pflege und Qualitätssicherung von Vorlagen innerhalb der Organisation. Anschließend ist eine Analyse der Anforderungen verschiedener Bereiche ratsam, um geeignete Dokumenttypen und Anforderungen zu identifizieren.

Folgende Fragestellungen können hier hilfreich sein:

- In welcher Abteilung sollen die Vorlagen gepflegt und bereitgestellt werden?
- Ist das nötige Fachwissen für die Umsetzung sowohl im Hinblick auf die Textverarbeitung als auch im Hinblick auf die digitale Barrierefreiheit vorhanden oder werden ggf. Schulungen benötigt?
- Welche Abteilung verwendet welche Vorlagen? Können ggf. Vorlagen zusammengefasst werden?

- Wie können die Vorlagen zentral bereitgestellt werden?
- Können Mechanismen implementiert werden, die die Verwendung individueller Vorlagen verhindern?
- Haben die Nutzenden der Vorlagen ausreichend Wissen, um die Vorlage korrekt anzuwenden, oder sind ggf. Schulungen notwendig?

## Erstellung der Vorlagen

Für Vorlagen existieren spezielle Dateiformate, z. B. .dotx für Microsoft Word oder .potx für Microsoft PowerPoint. Ein Doppelklick öffnet nicht die Vorlage selbst, sondern erstellt automatisch ein neues, unbenanntes Dokument auf Basis der Vorlage. Das schützt das Originaldokument (Vorlage) zuverlässig vor versehentlichen Änderungen und Überschreiben.

Im Gegensatz zu regulären Dokumenten enthalten Dokumentvorlagen vordefinierte Strukturen, Formatierungen und Platzhalter, die als Grundlage für neue Dokumente dienen. Für die Gestaltung barrierefreier Dokumentvorlagen sind die jeweiligen Anforderungen der Software zu berücksichtigen (siehe z. B. das Kapitel Allgemeine Anforderungen an die Barrierefreiheit von Dokumenten). Im Folgenden werden die zentralen Anforderungen dargestellt, die insbesondere bei der Entwicklung von Dokumentvorlagen relevant sind.

### Farben

Eine definierte Farbpalette sollte fest in der Vorlage hinterlegt werden, idealerweise auf Basis des Corporate Designs. Farben werden u. a. für Schriften, Texthervorhebungen, Linkformatierungen, Diagramme und Tabellen genutzt. Es ist darauf zu achten, dass bei der Verwendung des CDs oder anderer Farbvorgaben die Kontrastverhältnisse aus der EN 301 549 eingehalten werden. Des Weiteren dürfen nach Vorgabe der Norm Farben nur als Unterstützung und nicht als alleinige Informationsvermittlung verwendet werden.

### Tabellen

Das Tabellendesign sollte in der Vorlage festgelegt werden. In der Regel können hierzu die hinterlegten Farben der Farbpalette herangezogen werden, ggf. müssen Schriftfarben abhängig von den Hintergrundfarben angepasst werden. Zusätzlich stehen abhängig von der Software Formatierungsmöglichkeiten für die Optik gebänderter Zeilen und Rahmenlinien bereit.

### Formatvorlagen

Formatvorlagen sind zentral für die Struktur und Barrierefreiheit. Besonders relevant sind Überschriftenebenen, Absätze und Zitate, da sie die semantische Struktur für Screenreader bereitstellen. Überschriften müssen hierarchisch korrekt hinterlegt sein. Silbentrennung sollte, wenn möglich, deaktiviert werden. Bei der Konfiguration der entsprechenden Formatvorlagen ist darauf zu achten, dass die Vorgaben des CD korrekt umgesetzt werden. Auch ausreichend große Abstände, beispielsweise zwischen Überschrift und Text oder Absätzen sollte in den Formatvorlagen definiert werden. Zur besseren Anwendung und Fehlervermeidung sollten Formatvorlagen eindeutig benannt und nachvollziehbar strukturiert bzw. im Schnellzugriff angeordnet sein. Die Verwendung der Standardformatvorlage „Titel“ (z. B. in Microsoft Word) wird in einigen Quellen kritisch gesehen, da manche Screenreader diese Formatvorlage nicht unterstützen (vergleiche Erweiterte Checkliste nach EN 301 549 für barrierefreie Word-Dokumente). Eine Entfernung aus dem Schnellzugriff ist in diesem Fall sinnvoll.

## Listen und Verzeichnisse

Listen und Verzeichnisse können ebenfalls über entsprechende Formatvorlagen vordefiniert werden. Römische Ziffern stellen eine besondere Herausfordung im Verständnis dar.

## Inhaltsverzeichnis

Benutzerdefinierte Inhaltsverzeichnisvorlagen können zentral in Dokumentvorlagen hinterlegt und in weiteren Dokumenten wiederverwendet werden. Falls technisch möglich, sollte die Funktion für ein klickbares Inhaltsverzeichnis aktiviert werden, da dies die Navigation innerhalb des Dokuments deutlich verbessert. Zusätzlich kann – abhängig von der verwendeten Software – festgelegt werden, welche Überschriftenebenen im Inhaltsverzeichnis angezeigt werden sollen, um die Struktur gezielt zu steuern.

## Navigationshilfen

Navigationshilfen sind Funktionen, die helfen, sich schnell in längeren oder komplexen Dokumenten zu orientieren. Typische Navigationshilfen in Textverarbeitungsprogrammen sind Inhaltsverzeichnisse, Lesezeichen und formatierte Überschriftenebenen. Insbesondere für Menschen, die assistive Technologien wie Screenreader nutzen, erleichtern sie das Finden bestimmter Inhalte und verbessern die Übersicht. In Tabellenkalkulationsprogramme (z. B. Excel) stehen weniger Navigationsfunktionen für assistive Technologien zur Verfügung. Daher ist es ratsam, hier manuelle Navigationshilfen für Screenreader-Nutzende bereitzustellen. Beispiele für hilfreiche Navigationshilfen in Tabellenkalkulationsprogrammen sind:

- ein manuell verlinktes Inhaltsverzeichnis im ersten Tabellenblatt bereitstellen,
- pro Tabellenblatt nur eine Tabelle anführen,
- Thema der Tabelle in jeder A1-Zelle pro Tabellenblatt anführen,
- erste leere Zelle unterhalb der Tabelle mit einem textlichen Hinweis „Ende des Tabellenblatts“ oder einer ähnlichen Kennzeichnung versehen,
- Tabellenblätter aussagekräftig beschriften.

Vergleiche dazu auch den Abschnitt Hinweise zur Gestaltung von Tabellen.

## Sprache

In der Dokumentvorlage ist die passende Sprache festzulegen. Dadurch wird automatisch die Sprache für die Rechtschreibprüfung und die korrekte Sprachausgabe durch Screenreader festgelegt.

## Kopf- und Fußzeilen

Kopf- und Fußzeilen können in Vorlagen für durchlaufende Informationen wie bspw. Seitenzahlen oder der Dateiname etc. genutzt werden. Inhaltlich relevante Informationen sollten nicht in Kopf- und Fußzeilen platziert werden, da manche Softwareprogramme diese Inhalte ignorieren (Screenreader-Lesereihenfolge).

## Best practices

### Schnellbausteine

Schnellbausteine ermöglichen die Wiederverwendung häufig genutzter Inhalte wie Textbausteine oder Formatierungen. In Microsoft Word z. B. kann der Schnellbaustein „Dokumenttitel“ genutzt und platziert werden, um den Dokumentitel gleichzeitig im Dokument und in den Metadaten zu hinterlegen. Dadurch wird eine Anforderung der Barrierefreiheit erfüllt und der Bearbeitungsaufwand reduziert.

### Alternativtexte

Für Bilder (z. B. Logos) in den Vorlagen sollten bereits aussagekräftige Alternativtexte hinterlegt oder dekorative Elemente entsprechend gekennzeichnet werden.

### Beispiele

Vorlagen können Anwendungsbeispiele enthalten, etwa für barrierefreie Tabellen, Diagramme, Überschriftenhierarchien, Formulare, Titelblätter, Tabellenblätter oder Masterfolien. Dadurch wird die korrekte Nutzung erleichtert und einheitliche Ergebnisse gefördert. Für Tabellen kann z. B. die Wiederholung von Kopfzeilen bei Seitenumbrüchen bereits aktiviert werden.

### Unterstützungsangebote

Zu vielen Aspekten, die hier genannt wurden, finden Sie Informationen in weiteren Kapiteln dieser Handreichung.

## Veröffentlichung

Vor der Bereitstellung barrierefreier Dokumentvorlagen sollten diese fachlich, gestalterisch und technisch geprüft sowie offiziell freigegeben werden. Die Verantwortung dafür liegt in der Regel bei zentralen Stellen wie z. B. der Öffentlichkeitsabteilung, IT-Abteilung oder den Beauftragten für digitale Barrierefreiheit. Dadurch wird sichergestellt, dass die Vorlagen sowohl den gesetzlichen Anforderungen als auch den Vorgaben des Corporate Designs entsprechen. Ebenso wichtig ist eine klare Versionierung der Vorlagen. Änderungen, Aktualisierungen oder neue Funktionen sollten nachvollziehbar dokumentiert und an die Mitarbeitenden kommuniziert werden, beispielsweise über das Intranet, Rundmails oder Änderungsprotokolle. So wird verhindert, dass veraltete oder nicht barrierefreie Vorlagen weiterhin genutzt werden.

### Wie kommen die Dokumentvorlagen in die Organisation?

Die Verteilung der Vorlage ist einer der kritischsten Punkte. Sollte es hier Probleme geben, wird die jeweilige Vorlage nicht verwendet oder es wird weiter mit einer alten Version gearbeitet. Die Vorlagen sollten zentral und leicht zugänglich bereitgestellt werden, beispielsweise über ein Intranet, ein Dokumentenmanagementsystem oder Cloud-Dienste der Einrichtung. Es kann zwischen zwei Arten der Verteilung unterschieden werden.

### Push Prinzip

Die aktuellen Dokumentvorlagen sind mit der Erstell-Software verknüpft, d. h. bei neuen Dokumenten wird automatisch bzw. aus einem Set die entsprechende Dokumentvorlage geöffnet. Dafür gibt es neben den Boardmitteln über die Microsoft-Admin-Verwaltung etc., die sich je nach Größe und IT-Struktur nicht überall nutzen lassen, auch spezialisierte Dokumentmanagementsysteme (DMS), die sich auch in Word integrieren lassen.

### Pull Prinzip

Es gibt eine zentrale Stelle, die allen bekannt ist, von der sich jedes Mal die aktuelle Dokumentvorlage z. B. PowerPoint-Master zu ziehen ist. Dies verwenden sehr viele Organisationen. Wichtig zu wissen ist, dass hier die Verantwortung für die korrekte Verwendung mehr bei der Nutzung und weniger beim IT-System liegt.

### Sensibilisierung der Mitarbeitenden

Die Einführung barrierefreier Vorlagen erfordert eine gezielte Sensibilisierung der Mitarbeitenden für digitale Barrierefreiheit. Schulungen, Leitfäden und kurze Anleitungen unterstützen dabei, die Vorlagen korrekt anzuwenden und typische Fehler bei der Dokumentenerstellung zu vermeiden. Unterstützend können klare Stellungnahmen der Organisationsleitung wirken, um die Relevanz barrierefreier Dokumente hervorzuheben und die verbindliche Nutzung der Vorlagen zu fördern. Mitarbeitende sollten zudem angehalten werden, Vorlagen stets aus der zentralen Quelle zu verwenden, damit aktuelle Versionen und Anpassungen berücksichtigt werden.

### Implementierung und Qualitätssicherung der Vorlagen

Die Einführung neuer Vorlagen sollte schrittweise und nach klaren Prioritäten erfolgen. Empfehlenswert ist es, zunächst häufig genutzte Dokumenttypen bereitzustellen und anschließend weitere Bereiche einzubinden. Eine begleitende Qualitätssicherung kann sicherstellen, dass die Vorlagen auch nach Softwareupdates korrekt funktionieren und weiterhin barrierefreie Dokumente ermöglichen. Durch regelmäßige Stichproben kann überprüft werden, ob auch die mit den Vorlagen erstellten Dokumente noch den Anforderungen der Barrierefreiheit entsprechen. Werden dabei Probleme festgestellt, können die Vorlagen optimiert oder zusätzliche Unterstützung für Mitarbeitende angeboten werden. Ergänzend sollte ein Feedback-Management etabliert werden. Rückmeldungen von Mitarbeitenden und Nutzenden unterstützen die kontinuierliche Weiterentwicklung der Vorlagen, helfen bei der frühzeitigen Erkennung technischer Probleme und fördern die Akzeptanz innerhalb der Organisation.

## Verweise

### Interne Links

- Allgemeine Anforderungen an die Barrierefreiheit von Dokumenten
- Hinweise zur Gestaltung von Tabellen
- Hinweise zur Erstellung von Alternativtexten

### Externe Links in diesem Artikel

- Erweiterte Checkliste nach EN 301 549 für barrierefreie Word-Dokumente (Projekt SHUFFLE, Kompetenzzentrum Digitale Barrierefreiheit - abgerufen am 18.5.2026)

## Weiterführende Links

- Leitfaden Barrierefreie Textgestaltung in User-Interface Elementen (Überwachungsstelle des Bundes für Barrierefreiheit von Informationstechnik, BFIT)
- Leitfaden Kontraste für eine barrierefreie Gestaltung (Überwachungsstelle des Bundes für Barrierefreiheit von Informationstechnik, BFIT)
- Leitfaden Strukturierung von Textinhalten (Überwachungsstelle des Bundes für Barrierefreiheit von Informationstechnik, BFIT)
- Office Medien aus dem Styleguide des Bundes

# Überblick über Inhaltstypen und Formate

Online betrachten

## Anwendungsfälle

Die Wahl des Formats sollte sich nach dem Inhaltstyp richten. In diesem Artikel wird zwischen den folgenden Inhaltstypen unterschieden:

- strukturierter Text
- Formular
- komplexe Modelle und Strukturen
- Tabellenkalkulation
- Video
- OCR

## Strukturierter Text

Unter „strukturiertem Text“ verstehen wir textliche Inhalte mit einer klaren Struktur (Überschriften, Absätze, Listen, Tabellen). Auch Bilder und andere grafische Abbildungen können eingebettet sein.

Dafür sind v. a. die folgenden Dateiformate geeignet:

- **Microsoft Word oder LibreOffice Writer**. Ein Word- oder Writer-Dokument kann mit relativ geringem Aufwand barrierefrei gestaltet werden. Es gibt kostenlose Reader für alle Plattformen. Siehe Abschnitt Microsoft Word.
- **Microsoft PowerPoint oder LibreOffice Impress**. Folien im PowerPoint- oder Impress-Format können auf einfache Weise barrierefrei gestaltet werden. Es gibt kostenlose Reader für alle Plattformen. Siehe Abschnitt Microsoft PowerPoint.
- **HTML**. Ein Web-Dokument besteht fast immer aus mehreren Dateien (HTML, CSS, Bilder). Deshalb sollte es auf einem Webserver gehostet werden. Web-Dokumente sind nicht Teil dieser Handreichung.
- **EPUB**. EPUB (Electronic Publication) ist ein offenes Standardformat für digitale Bücher und Dokumente. Es handelt sich dabei im Prinzip um einen Container (ZIP-Datei), der strukturierte Webinhalte wie HTML, CSS und Metadaten enthält. Weitere Informationen unter Abschnitt Einführung in EPUB – Das neue Standardformat für digitale Bücher.
- **Adobe InDesign**. Das InDesign-Format bietet mehr gestalterische Möglichkeiten gegenüber Microsoft Word. Die Veröffentlichung geschieht dann als PDF-Dokument. Siehe Abschnitt PDF aus Adobe InDesign.
- **PDF**. Das PDF-Format bietet Vorteile gegenüber den MS-Office-Formaten bezüglich der plattformunabhängigen Darstellung und bei der Datensicherheit. Aber es ist aufwändiger in der barrierefreien Gestaltung. Deshalb sollte Sie nur dann PDF verwenden, wenn Sie auf dessen Vorteile angewiesen sind (zum Beispiel bei geschützten Formularen). Siehe Abschnitt PDF-Dokument. Allerdings lässt sich ausgehend von einem XML-Dokument unter Einsatz der Open Source Anwendung Apache-FOP ein Workflow generieren, der für gleichartige PDF-Dokumente alle benötigten Voreinstellungen automatisiert bereitstellt. Siehe Abschnitt PDF mit Apache-FOP. Die konkrete Beschreibung des gesamten Workflows ist dabei nicht Gegenstand dieser Handreichung.

- **LaTeX**. Um LaTeX-Dokumente zu erstellen, müssen Sie die Seitenbeschreibungssprache LaTeX beherrschen. Bei der Veröffentlichung wird dann meist auf HTML oder PDF zurückgegriffen. Um die Barrierefreiheit des finalen Dokuments sicherzustellen, muss eine aufwändige „Pipeline“ (Produktionsprozess) eingerichtet werden. Dies ist nicht Gegenstand dieser Handreichung.
- Ein E-Buch ist ein Wordformat, das in Schulen verwendet wird (siehe E-Buch-Steckbrief). Es wurde von der Deutschen Blindenstudienanstalt e. V. (blista), Marburg spezifiziert und wird in einigen Medienzentren der Bundesländer eingesetzt, z. B. im Medienzentrum der Johann-Peter-Schäfer-Schule. Das E-Buch ist nicht Gegenstand dieser Handreichung.

## Formular

Formulare bestehen aus strukturiertem Text mit eingebetteten Eingabefeldern, die von Nutzenden interaktiv bearbeitet werden. Bei Formularen ist die Datensicherheit ein wichtiger Aspekt.

Für Formulare sind die folgenden Dateiformate geeignet:

- **HTML**. Ein Web-Formular sammelt die Daten an zentraler Stelle auf einem Webserver. Es kann auf allen Plattformen barrierefrei ausgefüllt werden. Während des Ausfüllens ist eine Online-Verbindung erforderlich. HTML-Formulare sind nicht Gegenstand dieser Handreichung.
- **PDF**. Ein PDF-Formular kann offline ausgefüllt werden. Aber wenn das ausgefüllte PDF-Formular anschließend per E-Mail versendet wird, ist der Datenschutz nicht sicher gewährleistet. Barrierefreie PDF-Formulare können u. a. aus Word- oder InDesign-Dokumenten erzeugt werden. In beiden Fällen muss das PDF-Formular nachbearbeitet werden, z. B. in Adobe Acrobat. Siehe Abschnitt PDF-Formular.

## Komplexe Modelle und Strukturen

Unter „komplexe Modelle und Strukturen“ verstehen wir die visuelle Darstellung von meist komplexen Strukturen in wissenschaftlicher Literatur. Die Struktur wird meist in einem Domänen-spezifischen Format beschrieben. Die visuelle Darstellung soll das Erfassen erleichtern.

Einige Beispiele komplexer Modelle und Strukturen:

- **UML-Diagramme**. Visuelle Darstellung von Programmeigenschaften in der Informatik. Siehe u. a. PlantUML.
- **Mathematische Formeln**. Als Notation mathematischer Formeln hat sich LaTeX etabliert. Siehe Wikibooks LaTeX/Mathematics (en). LaTeX-Formeln können in Webseiten mittels MathJax barrierefrei eingebettet werden. Siehe Abschnitt Hinweise zu mathematischen Darstellungen.
- **Chemische Formeln**. Zur Notation chemischer Formeln kann ChemFig verwendet werden. Siehe Wikibooks LaTeX/Chemical Graphics (en).

Für die barrierefreie Darstellung komplexer Modelle und Strukturen ist es wichtig, eine „Pipeline“ (einen Produktionsprozess) aufzubauen, die zu einem barrierefreien Endergebnis führt. Dafür sind meist programmiertechnische Kenntnisse erforderlich. Weitere Details sind nicht Gegenstand dieser Handreichung.

## Tabellenkalkulation

Eine Tabellenkalkulation ist eine Software, die speziell auf die Eingabe und Verarbeitung von Daten in Form einer Tabelle sowie die Durchführung von Berechnungen ausgerichtet ist. Zu diesem Zweck ist der Arbeitsbereich in Zeilen und Spalten eingeteilt, wobei in den einzelnen Zellen Werte bzw. Eintragungen aus anderen Zellen verwendet bzw. referenziert werden können. Neben komplexer Referenzierung bietet eine

Software für Tabellenkalkulation ggf. auch unterschiedliche grafische Darstellungsmethoden. Darüber hinaus kann die Möglichkeit, Funktionen zu definieren, für komplexe statistische Datenanalysen genutzt werden.

Eine mit der sehr verbreiteten Anwendung für Tabellenkalkulation Microsoft Excel erstellte Arbeitsmappe kann barrierefrei gestaltet werden. Siehe Abschnitt Microsoft Excel.

## Video

Ein Video ist ein multimediales Dokument, das meist aus einer Bild- und einer Tonspur besteht. Aus Gründen der Barrierefreiheit sollte es auch mindestens eine Untertitelspur und in der Regel eine Audiodeskriptionsspur haben. Videos werden meist als MP4-Dateien gespeichert. Untertitel werden meist separat als VTT-Dateien oder SRT-Dateien abgelegt. Zu einem barrierefreien Video gehört auch ein barrierefreier Videoplayer.

Hinweise zur Erstellung barrierefreier Videos sind nicht Gegenstand dieser Handreichung.

## OCR

Zum Inhaltstyp OCR („Optical Character Recognition“) gehören eingescannte und abfotografierte Dokumente, die keine getaggte Struktur haben. Oft werden gescannte Dokumente in einer PDF-Datei als Bild gespeichert. Mittels OCR kann der Text aus der PDF-Datei extrahiert werden. Die Ergebnisse sind von der Qualität und Auflösung des Scan-Vorgangs abhängig.

OCR-Dokumente sollten vermieden werden, denn sie können nur unter großem Aufwand barrierefrei aufbereitet werden. Dies ist nicht Gegenstand dieser Handreichung.

## Verweise

### Interne Links

- Microsoft Word
- Microsoft PowerPoint
- PDF aus Adobe InDesign
- Einführung in EPUB – Das neue Standardformat für digitale Bücher
- PDF-Dokument
- PDF mit Apache-FOP
- PDF-Formular
- Hinweise zu mathematischen Darstellungen
- Microsoft Excel

### Externe Links in diesem Artikel

- Leitfaden barrierefreie EPUB3-E-Books (Börsenverein des deutschen Buchhandels)
- E-Buch-Steckbrief (Verband für Blinden- und Sehbehindertenpädagogik e. V.)
- Medienzentrum der Johann-Peter-Schäfer-Schule (Johann-Peter-Schäfer-Schule)
- PlantUML im Überblick (PlantUML.com)
- Wikibooks LaTeX/Mathematics (en) (wikibooks)
- JavaScript-basierte Darstellungsmöglichkeit für mathematische Formeln und Gleichungen in Webbrowsern (browserübergreifend) (MathJAX)
- LATEX/Chemical Graphics (Wikibools)

## Weiterführende Links

Keine vorhanden

# Hinweise zur Erklärung zur Barrierefreiheit

Online betrachten

## Allgemeine Einordnung

Die Gesetzgebung zur Umsetzung der EU-Richtlinie 2016/2102 sieht digitale Barrierefreiheit für Hochschulen als öffentliche Einrichtung vor. Dies betrifft z. B. alle Lehrmaterialien, die in Lernmanagementsystemen zum Download bereitgehalten werden. Aber auch Studienordnungen oder Anmeldeformulare für Prüfungen gehören dazu. Der Gesetzgeber verlangt dazu auch eine Erklärung zur Barrierefreiheit.

Dokumente, die in Webseiten eingebettet sind und die beim Rendern verwendet werden oder die dafür vorgesehen sind, zusammen mit der Webseite, in die sie eingebettet sind, gerendert zu werden, müssen entsprechend der Gesetzgebung durch eine „Erklärung zur Barrierefreiheit“ beschrieben werden. Üblich ist es, eine weitere Webseite dafür zu verwenden. Alternativ enthalten die Lehrmaterialien selbst die Angaben zur Barrierefreiheit.

Der Aufbau einer Erklärung zur Barrierefreiheit ist anschaulich auf der Seite der Landesfachstelle für Barrierefreiheit Sachsen-Anhalt dargestellt. Durch diese Angaben können Studierende u. a. erkennen, ob Barrieren bekannt sind und wie eine Verbesserung erreicht werden kann. Es ist zu beachten, dass die Erklärungen zur Barrierefreiheit der einzelnen Bundesländer und des Bundes Unterschiede aufweisen und deshalb die Hinweise auf der Webseite der Landesfachstelle für Barrierefreiheit Sachsen-Anhalt nur für dieses Bundesland maßgeblich sind. Die für jedes Bundesland eingerichtete Durchsetzungsstelle wird benannt, damit diese ggf. ein Durchsetzungsverfahren initiiert. Eine Liste aller Durchsetzungsstellen wird vom Hessischen Landeskompetenzzentrum für barrierefreie IT zur Verfügung gestellt. Die Erklärung ist mindestens einmal jährlich zu überprüfen und zu aktualisieren. Einige Hochschulen haben interne Meldestellen eingerichtet, die vorrangig kontaktiert werden können und oft auch beraten.

Die Überwachungsstellen überprüfen regelmäßig die öffentlichen Einrichtungen eines Landes von sich aus und berichten über das Ergebnis der Hochschule. Eine Liste aller Überwachungsstellen wird auf der Seite der Überwachungsstelle des Bundes für Barrierefreiheit von Informationstechnik bereitgestellt.

Die Monitoringstelle in Österreich stellt eine Mustererklärung zur Barrierefreiheit für Österreich zur Verfügung.

## Verweise

### Interne Links

Keine vorhanden

### Externe Links in diesem Artikel

- Erstellungshilfe für die Erklärung zur Barrierefreiheit (Landesfachstelle für Barrierefreiheit Sachsen-Anhalt)
- Liste der Durchsetzungsstellen in Deutschland (Landeskompetenzzentrum für barrierefreie IT Hessen)
- Liste aller Überwachungsstellen in Deutschland (Überwachungsstelle des Bundes für Barrierefreiheit von Informationstechnik)

- Mustererklärung zur Barrierefreiheit für Österreich (digitalbarrierefrei.at, Monitoringstelle in Österreich)

## Weiterführende Links

Keine vorhanden

# Hinweise zur Erstellung von Alternativtexten

Online betrachten

## Allgemeine Einordnung

Um die Bedeutung von Bilddateien erfassen zu können, sind Personen, die visuelle Inhalte nur eingeschränkt bzw. gar nicht wahrnehmen können, darauf angewiesen, dass jeweils eine zugehörige Textalternative bereitgestellt wird. Dabei tritt ein sogenannter Alternativtext ergänzend an die Stelle der Bilddatei, um Zugang zu den dargestellten Inhalten zu ermöglichen.

Um solide, umfassende Orientierung zu gewährleisten und Irritationen möglichst zu vermeiden, sollte ausnahmslos jede Bilddatei gemäß Entscheidungsbaum der W3C Web Accessibility Initiative (WAI) ausgewertet und gemäß der Empfehlung mit einem Alternativtext versehen werden, insbesondere auch dann, wenn es sich z. B. „nur“ um ein Werbebanner handelt.

Diesbezüglich kommt den Äußerungen im VISCH-Leitfaden allgemeine Gültigkeit zu:

> *Bei der Auswahl für Bilder und Abbildungen [...] geht es also nicht um die Frage, ob diese übertragen werden sollen, sondern vielmehr darum, wie dies geschehen soll. Die Frage, welche Abbildungen „relevant“ oder „wichtig“ sind, sollte keinesfalls von den Übertragern des Buches [...] entschieden werden. (blista, 2012, S.4)*

Und bereits zuvor:

> *Grundsätzlich sollten alle [...] Informationen [...] in geeigneter Form zugänglich gemacht werden. (ebd.)*

Dekorative visuelle Objekte ohne inhaltliche Bedeutung, die allein der Verzierung dienen, sowie Angaben in Kopf- und Fußzeilen in PDF-Dokumenten, die an anderer Stelle im PDF-Dokument enthalten sind, sollten explizit als Schmuckelemente gekennzeichnet werden. Eine weitere besondere Bedeutung kommt diesbezüglich den funktionalen Bilddateien zu. Vergleiche dazu auch den Abschnitt Allgemeine Anforderungen an die Barrierefreiheit von Dokumenten.

## Anmerkungen

- Alternativtexte dürfen weder aus dem reinen Dateinamen einer Grafik noch aus generischen Platzhaltern (wie z. B. „Bild 1“) bestehen.
- Außerdem müssen gemäß Kriterium 28-012 im Matterhorn-Protokoll auch Verlinkungen mit einer sogenannten Link Annotation versehen werden. Diese konkrete Anforderung ist automatisiert prüfbar, z. B. mit dem PDF Accessibility Checker (PAC). Im Sinne eines Dokuments für alle sollte sich ein solcher Link-Alternativtext nicht grundlegend von dem visuell wahrnehmbaren Linktext unterscheiden. Für vertiefende Informationen sei an dieser Stelle auf die weiterführenden Links verwiesen (siehe unten).
- Alternativtexte stellen eine Ergänzung zur jeweiligen Bilddatei (bzw. zur jeweiligen Verlinkung) dar. Die visuelle Gestaltung z. B. des PDF-Dokuments oder der Webseite etc. bleibt dadurch unverändert. Screenreader-Programme lesen zusätzlich zur Information, dass an einer bestimmten Stelle eine Bilddatei (bzw. eine Verlinkung) eingebunden ist, den für die Bilddatei (bzw. die Verlinkung) bereitgestellten zugehörigen Alternativtext aus. In geeigneten Programmen, wie z. B. der Screenreader-Vorschau von PAC, werden Alternativtexte insbesondere für Bilddateien auch visuell dargestellt.

## Grundsätzliche Überlegungen

Beim Verfassen von Alternativtexten sollten die folgenden allgemeinen Kriterien zugrunde gelegt werden:

- ohne Redundanz,
- kurz,
- neutral,
- kontextbezogen,
- einheitlich,
- systematisch.

Machen Sie sich am besten zunächst vertraut damit, wie die unterschiedlichen Elemente (Verlinkungen, Grafiken etc.) von einem Screenreader-Programm bezeichnet werden, um Doppelungen im Alternativtext zu vermeiden. Die Herausforderung besteht darin, den dargestellten Inhalt ohne Redundanz mit wenigen Worten möglichst neutral und zutreffend zu beschreiben.

Dabei könnten durchaus - als solche gekennzeichnete - Vermutungen oder Vergleiche hinzugezogen werden. Beachten Sie jedoch, dass es sich bei Formulierungen wie „Das Wetter ist schön.“ oder „Das Wetter ist schlecht.“ oder „glückliche Menschen“ oder auch „Urlaubsidylle“ aus dem täglichen Sprachgebrauch um Interpretationen bzw. Bewertungen handelt, die es in Alternativtexten zu vermeiden gilt.

Um den Kontextbezug einer Abbildung vollständig wiederzugeben, ist es nicht erforderlich und auch nicht zweckdienlich, auf jedes Detail einer Abbildung einzugehen. Vermeiden Sie außerdem die Wiederholung von Text, der bereits im Fließtext bzw. in der Bildunterschrift steht. Machen Sie sich dazu bewusst, wie zeitintensiv es ist, Text mit einem Screenreader-Programm abzuhören. Enthält ein Alternativtext lediglich identische Textpassagen bzw. bereits zuvor benannte Inhalte und ist überdies ggf. auch noch sehr lang, so investieren Sie beim Abhören viel Zeit, ohne etwas Neues zu erfahren.

Wobei Alternativtexte allerdings wiederum auch nicht dazu genutzt werden sollten, um Inhalte unterzubringen, die aus der Bilddatei selbst nicht ersichtlich sind und auch im Text nicht erwähnt werden, wie beispielsweise die Berechnungsformel für einen statistischen Kennwert. Der Alternativtext darf keine Zusatzinformation enthalten, sondern stellt nur eine Alternative für die Abbildung dar. Die **Verwendung von Fachbegriffen** hingegen ist im Hochschulkontext bzw. im wissenschaftlichen Kontext hilfreich und erforderlich.

## Beschreibung statt Formatierung

Beachten Sie darüber hinaus, dass es sich bei den einfachen Alternativtexten um rein linearen Text handelt, der nicht formatiert werden kann. **Textauszeichnungen** wie Hoch- bzw. Tiefstellungen, die insbesondere im naturwissenschaftlichen sowie mathematischen Kontext häufig benötigt werden, müssen ggf. entsprechend in einer unformatierten Formelschreibweise oder Sprechweise ausformuliert werden. Legen Sie dazu am besten vorab eine einheitliche Vorgehensweise fest.

Bei Formelzeichen, die zur Bezeichnung physikalischer Größen verwendet werden, kann ferner die Groß-/Kleinschreibung des Grundzeichens eine Rolle spielen. Beispielsweise steht „klein t“ für die Zeit, „groß T“ ggf. für die absolute Temperatur. Screenreader-Programme können so eingestellt werden, dass ein Großbuchstabe z. B. mittels Tonsignal kenntlich gemacht wird. Ist dies bei der jeweiligen Zielgruppe hinlänglich bekannt, so kann auf die Kennzeichnung „groß T“ im Alternativtext verzichtet werden. Auch beim Übertragen auf eine Braillezeile ist Groß-/Kleinschreibung unterscheidbar.

### Anmerkungen

Bei ausgelagerten Beschreibungen stehen Formatierungsmöglichkeiten wiederum in gewohntem Umfang zur Verfügung.

### Systematischer Aufbau

Generell ist ein systematischer Aufbau, der sich auf das Wesentliche beschränkt, als Orientierungshilfe optimal. Mit einem einleitenden Schlagwort, wie z. B. „Foto“, „Illustration“, „Porträt“, „Screenshot“, „Logo“ etc., lässt sich das Wesen der jeweiligen Abbildung bereits möglichst zutreffend erfassen und kategorisieren. Damit ist eine erste Einordnung gegeben, die durch die sich anschließende spezifische Beschreibung konkretisiert wird.

Screenreader-Programme bieten die Funktionalität, dass durch einzelne Elementgruppen durch Tabstops navigiert werden kann. Insbesondere in diesem Zusammenhang, d. h. beim Navigieren von einer Bilddatei zur nächsten, könnte es sich als hilfreich erweisen, wenn die Nummerierung der Abbildung, falls vorhanden, zur optimalen Orientierung vorangestellt wird.

Andererseits besagt die **Faustregel**, dass Alternativtexte eine Länge von maximal 80 bis 100 Zeichen möglichst nicht überschreiten sollte. Die Angabe zur Nummerierung sowie das einleitende Schlagwort beanspruchen mithin bereits einen Großteil der insgesamt zur Verfügung stehenden Zeichen. Diesbezüglich gilt es, den Nutzen einer solchen Vorgehensweise im jeweiligen Kontext optimal abzuwägen.

### Prüfungsaufgaben

Damit Alternativtexte zu Abbildungen in Prüfungsaufgaben die Lösung nicht aufdecken, muss der Alternativtext so abgewandelt werden, dass die Aufgabe gleichwertig bleibt.

## Anwendungsfälle im mathematisch-naturwissenschaftlichen Bereich

### Alternativtexte bei chemischen Strukturformeln und mathematischen Formeln

Es bietet sich an, bei chemischen Strukturformeln ChemFig-Code und bei mathematischen Formeln LaTeX-Code in den Alternativtext aufzunehmen. Dabei wäre je nach Zielgruppe auch eine Mischform denkbar, wie beispielsweise bei der folgenden Formulierung im Tagungsbeitrag zur automatisierten Erstellung von Journal-Ausgaben der MAK Collection:

> *Strukturformel von 2-Piperidin-1-ylethanol: \chemfig{*6(–N([:-30]-([:30]-([:-30]-OH)))—-)} (Ziemer, 2023, S.15)*

Die Bereitstellung von Alternativtexten in Klausuraufgaben stellt wiederum eine besondere Herausforderung dar. Diesbezüglich sei auf die alternative Umsetzungsmöglichkeit im vom Projekt iBoB herausgegebenen Praxisleitfaden zur Erstellung textbasierter Alternativen für Grafiken „Gut fürs Image!“ hingewiesen (vgl. S.30).

Vergleiche diesbezüglich auch den Abschnitt Hinweise zu mathematischen Darstellungen.

### Alternativtexte bei komplexen Bilddateien

Es ist empfehlenswert, komplexe Bilddateien in einem Beitrag bzw. in einer Beitragsreihe zunächst systematisch zu erfassen und zu kategorisieren. Im nächsten Schritt kann sodann insbesondere für Bilddateien zu statistischen Kennwerten eine Schablone für die Alternativtexte entwickelt werden. In der

Ausformulierung sollten in geeigneter Weise der Grafiktyp sowie die Achsenbeschriftungen und ggf. weitere Charakteristika benannt werden, wie beim nachstehend zitierten Beispiel „Diagrammtyp der Wertepaare für ‚Beschriftung x-Achse' und ‚Beschriftung y-Achse'" im Tagungsbeitrag zur automatisierten Erstellung von Journal-Ausgaben der MAK Collection:

> *Abbildung 2: Streudiagramm der Wertepaare für Naphthalin und der Summe aus 1- und 2-Naphthol im Urin mit Regressionsgerade und Bestimmtheitsmaß R hoch 2 0,720 (Ziemer, 2023, S. 14)*

Dabei könnte außerdem die Skalierung der Achsen explizit benannt werden. Falls bei der Benennung die Reihenfolge erst x-Achse, dann y-Achse nicht eingehalten wird, ist es empfehlenswert, dies gesondert zu kennzeichnen, wie in dem folgenden Beispiel bei ACM SIGACCESS (Special Interest Group on Accessible Computing):

> *Line graph showing number of pauses from 0 to 350 on the Y axis against % of movement from 0 to 100 in increments of 5 on the X axis. (Trewin, 2019)*

Bei Diagrammen z. B. mit Messwerten sollte zudem in Erwägung gezogen werden, im Anhang eine Wertetabelle bereitzustellen. Erst dann lässt sich objektiv erschließen, was genau in der Abbildung dargestellt wird und welcher Eintrag welcher Quelle bzw. welchem (Arbeits-)Bereich zuzuordnen ist.

Beim Bereitstellen von Alternativtexten von inhaltsschweren Bilddateien wird die in der Faustregel für die Länge eines Alternativtextes benannte Zeichenanzahl von 80 bis 100 Zeichen rasch überschritten werden.

Lagern Sie umfangreiche Bildbeschreibungen in separaten Dateien aus. Eine Bildbeschreibung führt im Allgemeinen vom Groben zum Feinen, wobei sich Grob- und Feinstrukturbeschreibung durchaus auch überlappen können. Beispiele für Bildbeschreibungen im schulischen Kontext finden sich z. B. im VISCH-Leitfaden der Deutschen Blindenstudienanstalt e. V. (blista).

## Ausblick

Dieser Abschnitt wird in den folgenden Versionen der Handreichung kontinuierlich erweitert werden.

## Verweise

### Interne Links

- Allgemeine Anforderungen an die Barrierefreiheit von Dokumenten
- Hinweise zu mathematischen Darstellungen

### Externe Links in diesem Artikel

- Entscheidungsbaum für Alternativtexte der Web Accessibility Initiative (WAI) (W3C)
- VISCH-Leitfaden für Visualisierte Inhalte in SCHulbüchern mit zahlreichen Beispielen (Deutschen Blindenstudienanstalt e. V., blista)
- Prüfkriterien für PDF/UA-Konformität (Matterhorn-Protocoll in der englischen Version von 2021) (PDF Association)
- Prüftool PDF Accessibility Checker (PAC) (PAC, gewartet von axes4)
- Praxisbeispiel MAK Collection - Tagungsbeitrag zur automatisierten Erstellung von Journal-Ausgaben bei ZB MED im barrierefreien PDF-Format (Anja Ziemer, o-bib. Das offene Bibliotheksjournal)
- Praxisleitfaden zur Erstellung textbasierter Alternativen für Grafiken (Anja Fibich, Frauke Onken & Christian Axnick, iBoB - Projekt inklusive berufliche Bildung ohne Barrieren)

- Beispiele zur Beschreibung von Line Graphs, Boxplot Charts, Screenshots, Flussdiagrammen sowie Fotos bei ACM SIGACCESS (Special Interest Group on Accessible Computing) (Shari Trewin)

## Weiterführende Links

### Weiterführende Literatur mit mathematisch-naturwissenschaftlichem Kontext

- Anleitung zu mathematische Formeln (Gunther Heintzen, Regionales Rechenzentrum Erlangen (RRZE))
- Author Guide to Writing Alt Text (PDF-Ausgabe) (Taylor & Francis Group)
- Author Guide to Writing Alt Text (HTML-Ausgabe) (Taylor & Francis Group)
- ExcelViZ: Automated Generation of High-Level, Adaptable Scatterplot Descriptions Based on a User Study (Christin Engel & Jan Schmalfuß-Schwarz)
- Präsentationsfolien zur Vorstellung des Praxisbeispiels MAK Collection bei der 111. BiblioCon im barrierefreien PDF-Format (Anja Ziemer)
- Projekt Math4VIP - Plattform des Kooperationsprojekts (MathVIP, Universität Marburg, Karlsruher Institut für Technologie (KIT))
- Projekt MathVIP - KIT Presseinformation 004/2023 (Zentrum für digitale Barrierefreiheit und Assistive Technologien des KIT, ACCESS@KIT)
- Projekt MathVIP - Presseinformation 004/2023 (Zentrum für digitale Barrierefreiheit und Assistive Technologien des KIT, ACCESS@KIT)
- Versprachlichung von Formeln: Poster (Wiebke Janßen & Gesche Pospiech, Technische Universität Dresden)
- Versprachlichung von Formeln: Tagungsbeitrag zu Bedeutung und Vermittlung von Formeln (Download als PDF-Dokument) (Wiebke Janßen & Gesche Pospiech, Technische Universität Dresden)

### Anleitungen zu Prüfschritten bzw. Prüfkriterien

- BITV-Softwaretest - Prüfschritt 1.1.1 Nicht-Text-Inhalte (BIT inklusiv – Barrierefreie Informationstechnik für inklusives Arbeiten)
- PDF-Prüfverfahren - Prüfschritt BITi 02.3.1 Inlinelevel Strukturelemente / Links (BIT inklusiv – Barrierefreie Informationstechnik für inklusives Arbeiten)
- PDF-Prüfverfahren - Prüfschritt BITi 02.4.1.1 Grafiken / Alternativtexte im PDF-Prüfverfahren von BIT inklusiv (BIT inklusiv – Barrierefreie Informationstechnik für inklusives Arbeiten)
- Prüfkriterien für PDF/UA-Konformität (Matterhorn-Protokoll in der deutschen Version vom 23.06.2016 im Auftrag von BIT inklusiv) (PDF Association)

# Hinweise zur Gestaltung von Tabellen

Online betrachten

## Allgemeine Einordnung

Tabellen sind ein unverzichtbares Werkzeug, um Daten in einer möglichst verständlichen visuellen Form darzustellen.

Tabellen selbst können jedoch für bestimmte Personengruppen eine erhebliche Barriere darstellen, insbesondere für Menschen, die auf Screenreader-Programme oder andere unterstützende Technologien angewiesen sind. Diese Technologien haben oft Schwierigkeiten, die komplexen Strukturen und Beziehungen innerhalb von Tabellen richtig zu interpretieren und wiederzugeben.

## Grundsätzliche Überlegungen

Ein häufiges Problem von Tabellen ist die fehlende semantische Struktur. Viele Tabellen sind nicht korrekt mit Informationen über sogenannte „Kopfzellen" (z. B. in HTML mit <th> - table header) sowie mit Attributen oder Beschreibungen versehen, sodass Screenreader-Programme nicht wissen, wie sie den Inhalt kontextualisieren sollen. Ohne diese Informationen kann es für Nutzende schwierig sein zu verstehen, welche Daten miteinander verknüpft sind und was sie bedeuten.

Außerdem fehlt oft eine logische und konsistente Anordnung der Daten. Insbesondere können verschachtelte oder verbundene Zellen zusätzliche Hindernisse darstellen, da sie die lineare Leselogik eines Screenreader-Programms stören.

Wird darüber hinaus ein Übermaß an Fakten und zusammenhängenden Daten in einer einzigen Tabelle gebündelt, so ist die Komplexität sehr rasch unübersichtlich. Die Aufteilung komplexer Tabellen in mehrere kleinere Tabellen erleichtert das Verständnis für alle Nutzende.

## (Negatives) Beispiel aus der Praxis

Dass verschachtelte Tabellen eine besondere Herausforderung darstellen, soll das folgende Beispiel aus der Praxis veranschaulichen.

| **Tag** | **Veranstaltung** | | |
|---|---|---|---|
| | **Zeitplan** | | **Thema** |
| | **Beginn** | **Ende** | |
| Montag | 8:00 Uhr | 17:00 Uhr | Einführung |
| | | | 1. Themengebiet |
| Dienstag | 8:00 Uhr | 11:00 Uhr | 2. Themengebiet |
| | 11:00 Uhr | 14:00 Uhr | 3. Themengebiet |
| | 14:00 Uhr | 16:00 Uhr | |
| Mittwoch | 8:00 Uhr | 12:00 Uhr | Ausblick |
| Legende | Morgens | Mittags | Nachmittags |

### Anmerkungen

Dieses Praxisbeispiel steht explizit für eine Umsetzung, die **nicht** empfehlenswert ist. Im zugrundeliegenden Original weisen darüber hinaus der gewählte Blauton sowie der gewählte Grünton zwar ausreichend Kontrast laut EN 301 549 auf, sind allerdings nicht angenehm zu lesen.

## Maßnahmen

Um Tabellen zugänglicher zu machen, sollten folgende Maßnahmen ergriffen werden:

- **Alternativen nutzen:** Bieten Sie nach Möglichkeit alternative Darstellungsformen an, z. B. Listen oder zumindest eine Aufteilung in kleinere, weniger komplexe Tabellen.
- **Beschriftung und Kontext:** Stellen Sie sicher, dass jede Tabelle eine klare Überschrift hat, und geben Sie gegebenenfalls eine kurze Beschreibung oder Anleitung, wie die Tabelle zu verwenden und zu verstehen ist. Fügen Sie zu diesem Zweck zusätzlich „alternative" Texte hinzu, indem Sie z. B. in Microsoft Word mit der rechten Maustaste auf die Tabelle klicken und "Tabelleneigenschaften" auswählen. Dort können Sie unter dem Reiter "Alternativtexte" eine Beschreibung eingeben.
- **Semantische Elemente verwenden:** Weisen Sie Kopfzellen einer Tabelle als Kopfzellen aus. Nutzen Sie dazu in Microsoft Word das Attribut „Gleiche Kopfzeile auf jeder Seite wiederholen" und in HTML die Kennzeichnung <th> für Kopfzellen.
- **Linearisierung überprüfen:** Testen Sie Ihre Tabelle mit einem Screenreader-Programm, um sicherzustellen, dass die Daten in einer sinnvollen Reihenfolge vorgelesen werden.
- **Konsistenz wahren:** Achten Sie darauf, dass ähnliche Datentypen immer in derselben Spalte oder Zeile stehen, und vermeiden Sie unnötige Verschachtelungen oder verbundene bzw. geteilte Zellen.

Durch die Einhaltung dieser „Best Practices" kann sichergestellt werden, dass Tabellen für Menschen, die auf Screenreader-Programme und andere assistive Technologien angewiesen sind, zugänglich sind und somit eine inklusivere digitale Umgebung geschaffen wird.

## Beispiel aus der Praxis: Alternative Darstellungen

Unter Beachtung des Maßnahmenkatalogs lassen sich für das oben angeführte Praxisbeispiel verschiedene alternative Darstellungen finden.

### Alternative 1: Komplexität vereinfachen

In einer ersten alternativen Darstellung wird das Ereignis in einem einleitenden Satz erwähnt und auf die doppelte Verschachtelung a. Veranstaltung (mit Zeitplan und Thema) sowie b. Zeitplan (mit Beginn und Ende) verzichtet. Aufgrund der Angabe der konkreten Uhrzeiten ist die farbige Legende in der letzten Zeile redundant und kann ebenfalls entfallen.

Bei Wochentagen, denen mehrere Zeilen zugeordnet sind, wird zudem darauf verzichtet, die Kopfzelle über mehrere Zeilen zu verbinden. Stattdessen wird der Wochentag wiederholt. Bei einem Ereignis, dem mehrere Themen zugeordnet sind, wird auf eine Teilung der Zelle und die Darstellung der Themen in getrennten Zeilen verzichtet. Stattdessen wird eine lineare Formulierung (Thema 1 und Thema 2) gewählt. Bei separaten Ereignissen zu demselben Thema wird - wie beim Wochentag - darauf verzichtet, die Zelle über mehrere Zeilen zu verbinden. Stattdessen wird das Thema (mit einem ergänzenden Zusatz) wiederholt.

**Tabelle XYZ:** Die folgende Tabelle zeigt den zeitlichen Ablauf der Veranstaltung XYZ.

| Tag | Beginn | Ende | Thema |
|---|---|---|---|
| Montag | 8:00 Uhr | 17:00 Uhr | Einführung und 1. Themengebiet |
| Dienstag | 8:00 Uhr | 11:00 Uhr | 2. Themengebiet |
| Dienstag | 11:00 Uhr | 14:00 Uhr | 3. Themengebiet - Teil 1 |
| Dienstag | 14:00 Uhr | 16:00 Uhr | 3. Themengebiet - Teil 2 |
| Mittwoch | 8:00 Uhr | 12:00 Uhr | Ausblick |

### Alternative 2: Alternative nutzen

Der Inhalt der Tabelle aus dem oben angeführten Praxisbeispiel lässt sich alternativ auch als Liste darstellen. Bei einem Ereignis, dem mehrere Themen zugeordnet sind, sowie bei separaten Ereignissen zu ein und demselben Thema werden dabei die in Alternative 1 getroffenen Maßnahmen beibehalten.

**Liste XYZ:** Die folgende Auflistung zeigt den zeitlichen Ablauf der Veranstaltung XYZ.

- Montag
  - 8:00 Uhr bis 17:00 Uhr: Einführung und 1. Themengebiet
- Dienstag
  - 8:00 Uhr bis 11:00 Uhr: 2. Themengebiet
  - 11:00 Uhr bis 14:00 Uhr: 3. Themengebiet - Teil 1
  - 14:00 Uhr bis 16:00 Uhr: 3. Themengebiet - Teil 2
- Mittwoch
  - 8:00 Uhr bis 12:00 Uhr: Ausblick

## Anforderungen zur Barrierefreiheit von Tabellen in PDF-Dokumenten

Wie bei den Richtlinien zur Barrierefreiheit im Allgemeinen besteht das Hauptziel von PDF/UA (Universal Accessiblity) für Tabellen darin, zu definieren, wie Tabellen in PDF-Dokumenten so dargestellt werden können, dass sie zugänglich sind.

Um als barrierefrei zu gelten, müssen Tabellen in PDF-Dokumenten geeignete semantische Strukturen verwenden und diese in einer logischen Lesereihenfolge anordnen. Weiter muss es eine Verbindung zwischen den Überschriftenzellen und den dazugehörigen Unterzellen geben. Das kann bei einfachen Tabellen mit dem sogenannten Scope-Attribut realisiert werden. Bei komplexen Tabellen wird mit der Technik der Header-IDs gearbeitet.

PDF/UA baut auf der eigentlichen PDF-Spezifikation auf und ist als ergänzender Standard gedacht, der in Kombination mit dieser verwendet wird. PDF/UA gibt es in zwei Versionen bzw. zwei Spezifikationen, PDF /UA-1 (ISO 14289-1) und PDF/UA-2 (ISO 14289-2). PDF/UA-1 ist der Unterstandard zu PDF-1.7. PDF/UA-2 ist der Unterstandard zu PDF-2.0. PDF/UA-1 gilt derzeit noch als aktueller Standard, während PDF/UA-2 im März 2024 verabschiedet wurde und noch auf seine Verbreitung wartet. Darüber hinaus stehen derzeit (Stand: Mai 2026) auch noch kaum Programme zur Verfügung, um PDF/UA-2 zu prüfen. Siehe auch Abschnitt Entscheidungshilfe zu PDF Konvertern - Software zum PDF-Export sowie den Exkurs PDF/UA-2.

Im Hinblick auf Tabellen erweitert und verstärkt PDF/UA-2 die Bestimmungen zu strukturellen und semantischen Informationen. So werden im PDF/UA-2 Standard Tabelleneigenschaften und Strukturen genauer definiert und Verweise auf Kopfzellen unterschiedlicher Ordnung eindeutiger bezeichnet. Dadurch ist für ein Screenreader-Programm klarer erkennbar, in welcher hierarchischen Reihenfolge verschachtelte Kopfzellen einer Datenzelle zuzuordnen sind.

## Verweise

### Interne Links

- Entscheidungshilfe zu PDF Konvertern - Software zum PDF-Export
- Exkurs PDF/UA-2

### Externe Links in diesem Artikel

Keine vorhanden

### Weiterführende Links

Keine vorhanden

# Hinweise zu mathematischen Darstellungen

Online betrachten

## Allgemeine Einordnung

Die Kombination von Mathematik und Barrierefreiheit ist ein komplexes Thema, das differenziert betrachtet werden muss. Zunächst stellt sich die Frage, für welche Zielgruppe und in welchem Detaillierungsgrad die Informationen gedacht sind:

- reine mathematische Informationsvermittlung oder
- Mathematik im Lern- und Lehrkontext nutzen.

## Mathematische Informationsvermittlung

Für die allgemeine Vermittlung von mathematischen Formeln, Grafiken und Inhalten sollte grundlegend darauf geachtet werden, als Darstellungsoption keine Rastergrafiken (z. B. JPG, PNG), sondern skalierbare Varianten (Vektorgrafiken, z. B. SVG, EPS) zu verwenden:

- Für mathematische **Formeln** eignen sich LaTeX bzw. MathML-/XML-basierte Darstellungen (die auf Webseiten z. B. mittels MathJAX dargestellt werden können und darüber hinaus auch in Alternativtexten zur Verfügung stehen).
- und für **Graphen/Grafiken** sollten Vektorgrafiken (SVG) genutzt werden.

Diese Varianten ermöglichen eine verlustfreie Vergrößerung der Inhalte bzw. einen Zugriff auf Alternativtexte.

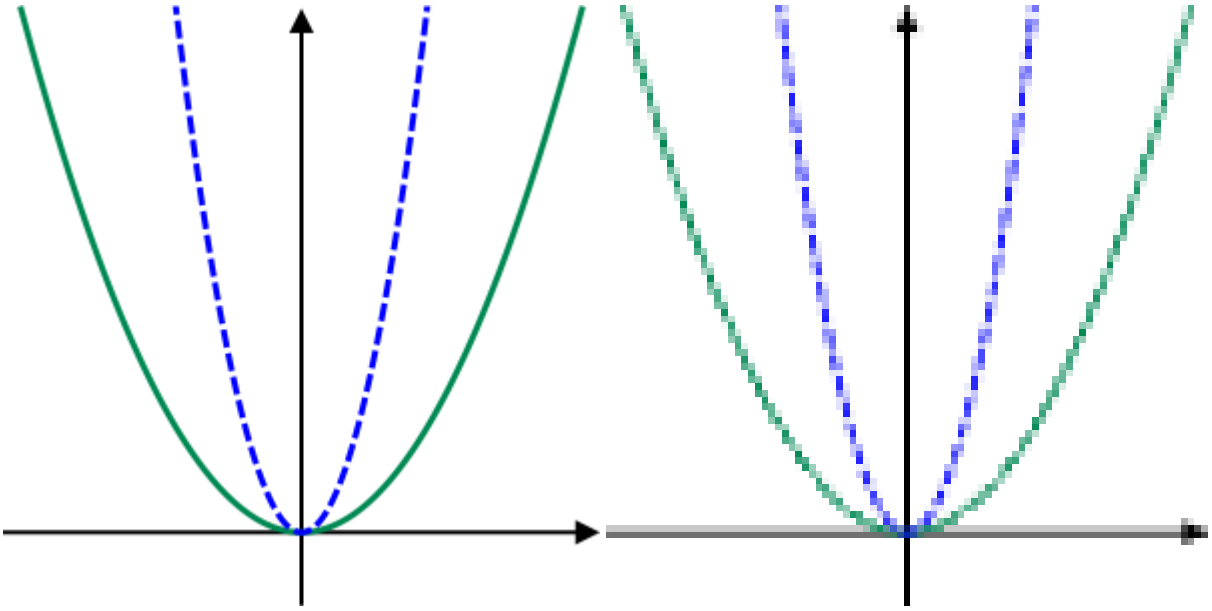
## Darstellung von Formeln

In der folgenden Tabelle ist dieselbe Formel einmal als Bild und einmal auf LaTeX basierend dargestellt, um die Unterschiede in der Darstellungsqualität zu veranschaulichen.

| Bild | MathML-Darstellung |
|---|---|
| $\frac{m_1}{m_2} = \frac{\sin \beta}{\sin \alpha}$ | $$\frac{m_1}{m_2}=\frac{\sin\beta}{\sin\alpha}$$ |

## Darstellung von Graphen/Grafiken

Nachfolgend ist derselbe Graph links als SVG-Datei, d. h. als Vektorgrafik, eingebettet und rechts als PNG-Datei, d. h. als Rastergrafik, um die Unterschiede hinsichtlich der Skalierbarkeit zu veranschaulichen.



In der Praxis ist der Unterschied oft weniger stark ausgeprägt. Um die Problematik zu verdeutlichen, wurde die Rastergrafik absichtlich in geringerer Auflösung erstellt. Besonders deutlich wird der Vorteil der SVG-Datei beim nachträglichen Vergrößern: Vektorgrafiken lassen sich beliebig skalieren, ohne an Schärfe zu verlieren.

## Mathematik im Lern- und Lehrkontext

In Bildungseinrichtungen geht es nicht nur um das Lesen, sondern auch um das Verstehen und Anwenden von Mathematik. Hier setzt das Projekt Math4VIP der Universität Marburg und des Karlsruher Instituts für Technologie an. Ziel des Projekts ist es, Studierenden mit Seheinschränkung den Zugang zu mathematischen Inhalten zu erleichtern.

Insbesondere in den MINT-Studiengängen stellen die mathematischen Anteile für sehbeeinträchtigte Studierende eine große Herausforderung dar, da die im Studium behandelten komplexen Inhalte häufig visuell vermittelt werden. Nur wenige Hochschulen bieten professionelle Unterstützung bei der Aufbereitung dieser Materialien an, sodass betroffene Studierende auf individuelle Assistenzen angewiesen sind. Dies kann sich negativ auf den Studienerfolg auswirken.

Das Ziel des Math4VIP-Projekts ist es daher, eine zentrale Plattform zu schaffen, die Informationen über barrierefreien Zugang zu Mathematik sowie Schritte zur barrierefreien Aufbereitung mathematischer Inhalte bereitstellt. Dabei werden neue Standards entwickelt und entsprechende Materialien erstellt und Leitfäden für unterschiedliche Zielgruppen verfasst. Durch Öffentlichkeitsarbeit sollen diese Maßnahmen bekannt gemacht werden. So erhalten Studierende mit Sehbeeinträchtigung unabhängig von ihrer Hochschule Zugang zu barrierefreien Materialien.

Zur Verfügung stehen gezielte Handreichungen für:

- Studierende bzw. Lernende,
- Lehrende und
- Umsetzende.

## Verweise

### Interne Links

Keine vorhanden

## Externe Links in diesem Artikel

- JavaScript-basierte Darstellungsmöglichkeit für mathematische Formeln und Gleichungen in Webbrowsern (browserübergreifend) (MathJAX)
- Plattform des Kooperationsprojekts Math4VIP (MathVIP, Universität Marburg, Karlsruher Institut für Technologie (KIT))

## Weiterführende Links

- Anleitung zu mathematische Formeln (Gunther Heintzen, Regionales Rechenzentrum Erlangen (RRZE))
- Author Guide to Writing Alt Text (PDF-Ausgabe) (Taylor & Francis Group)
- Author Guide to Writing Alt Text (HTML-Ausgabe) (Taylor & Francis Group)
- Beispiele zur Beschreibung von Line Graphs, Boxplot Charts, Screenshots, Flussdiagrammen sowie Fotos (Special Interest Group on Accessible Computing) (Shari Trewin)
- ExcelViZ: Automated Generation of High-Level, Adaptable Scatterplot Descriptions Based on a User Study (Christin Engel & Jan Schmalfuß-Schwarz)
- Projekt MathVIP - KIT Presseinformation 004/2023 (Zentrum für digitale Barrierefreiheit und Assistive Technologien des KIT, ACCESS@KIT)
- Projekt MathVIP - Presseinformation 004/2023 (Zentrum für digitale Barrierefreiheit und Assistive Technologien des KIT, ACCESS@KIT)
- Versprachlichung von Formeln: Poster (Wiebke Janßen & Gesche Pospiech, Technische Universität Dresden)
- Versprachlichung von Formeln: Tagungsbeitrag zu Bedeutung und Vermittlung von Formeln (Wiebke Janßen & Gesche Pospiech, Technische Universität Dresden)
- VISCH-Leitfaden für Visualisierte Inhalte in SCHulbüchern mit zahlreichen Beispielen (Deutschen Blindenstudienanstalt e. V., blista)

# Hinweise zu Tools zur Überprüfung der Barrierefreiheit

Online betrachten

## Allgemeine Einordnung

Es existieren unterschiedliche Tools zur Überprüfung der digitalen Barrierefreiheit von Dokumenten. Diese arbeiten unterschiedlich und überprüfen verschiedene Faktoren eines Dokuments. Aus diesem Grund empfiehlt sich in Projekten eine doppelte Kontrolle mit unterschiedlichen Prüftools. Nachfolgend finden Sie eine Übersicht einiger dieser Tools.

Bitte beachten Sie, dass es bei jeder Barrierefreiheitsprüfung Kriterien gibt, die automatisiert von einem Tool geprüft werden können, und Kriterien, die vom Menschen zu prüfen sind. So kann z. B. ein Tool prüfen, ob es einen Alternativtext für Bilder gibt; ob dieser sinnvoll ist, muss jedoch von einem Menschen geprüft werden.

## Prüfung der Barrierefreiheit in Microsoft Office-Dokumenten

In allen Microsoft Office-Programmen gibt es die Möglichkeit, eine automatisierte Barrierefreiheitsprüfung zu starten. Diese finden Sie bei den Office 365 Produkten im Menü „Überprüfen“ und dann unter dem Punkt „Barrierefreiheit überprüfen“. Zu beachten ist, dass hier nur Anforderungen geprüft werden können, die automatisiert prüfbar sind. Das sind **nicht** alle Anforderungen an die Barrierefreiheit. Zu beachten ist auch, dass die Qualität der Prüfung von der eingesetzten Version der Microsoft Office-Produkte abhängig ist. In der Regel gilt, dass mit der neuesten Version die besten Ergebnisse erzielt werden.

In jedem Fall muss die Barrierefreiheit nach einer automatisierten Prüfung auch noch manuell geprüft werden.

Bei der Konvertierung in das PDF-Format ist noch Folgendes zu beachten: Auch wenn die Barrierefreiheitsprüfung von Microsoft ohne Fehler läuft, kann es im PDF noch Barrierefreiheitsfehler geben. Genauso umgekehrt, wenn die Barrierefreiheitsprüfung von Microsoft Fehler oder Warnungen anzeigt, kann es sein, dass das PDF fehlerfrei ist. Ob das PDF noch Fehler aufweist oder nicht, hängt sehr stark vom jeweiligen Konverter ab.

## Prüfung von PDF-Dokumenten

Fertige PDF-Dateien können direkt mit Tools auf die Barrierefreiheit überprüft werden. Hierbei stehen mehrere Möglichkeiten zur Verfügung. Welche Kriterien eines PDF-Dokuments von einem Tool geprüft werden können und welche vom Menschen zu prüfen sind, wird im Matterhorn-Protokoll beschrieben: Deutsche Version des Matterhorn Protokolls 1.02.

### PDF Accessibility Checker (PAC)

Der kostenlose PAC prüft Kriterien des PDF/UA-Standards und der WCAG. Die automatische Prüfung prüft die ca. 2/3 maschinenprüfbaren Kriterien des PDF/UA-Standards. Zu diesen kann ein Detailbericht mit Position des jeweiligen Fehlers geöffnet werden.

Für die restlichen, vom Menschen zu prüfenden Kriterien sind die Screenreader-Vorschau und die Anzeige der „Logischen Struktur“ geeignet. Zusätzlich gibt es Qualitätsprüfungen, die auf mögliche Fehler und Problemstellen hinweisen. Es muss jedoch ein Mensch entscheiden, ob das der Fall ist. In diesem Sinne

sind die neuen Qualitätsprüfungen keine offiziellen technischen Kriterien und sie müssen nicht alle erfüllt sein, um das PDF-Dokument als barrierefrei auszuweisen.

PAC konnte lange Zeit nur unter Windows genutzt werden. Eine Erweiterung für andere Betriebssysteme ist vorgesehen. axes4 bietet bereits eine webbasierte Version von PAC, die entsprechend nicht nur unter Windows nutzbar ist (siehe auch unten unter axesCheck).

Auf der Webseite des PAC gibt es eine Anleitung und Termine für Webinare zur Verwendung des PAC. Auf der Webseite von axes4 werden Aufzeichnungen von Webinaren zum PAC und zu anderen Themen zur Verfügung gestellt: https://www.axes4.com/de/ressourcen-community/videos

Zum Beseitigen einzelner Probleme gibt es Anleitungen:

- PAC Fehler reparieren mit Adobe Acrobat
- PDF/UA Probleme lösen mit axesPDF

Neben dem PDF/UA-Standard müssen PDF-Dateien ebenfalls die Anforderungen des Abschnitts 10 der EN 301 549 erfüllen. Diese können über den Reiter „WCAG“ angezeigt werden. Auch hier ist zu beachten, dass nicht alle Anforderungen aus Abschnitt 10 automatisiert geprüft werden können. Eine manuelle Prüfung der Anforderungen aus Abschnitt 10 ist daher zu empfehlen.

Mit PAC 2026 ist eine Prüfung unter Verwendung von KI hinzugekommen. Die KI-Prüfung soll dabei helfen, Stellen zu identifizieren, bei denen die visuelle Darstellung und die Tag-Struktur des PDF-Dokuments voneinander abweichen. Ob die Klassifizierung durch die KI-Prüfung korrekt ist und ob die Tag-Struktur des PDF-Dokuments tatsächlich nachgebessert werden sollte, kann nur ein Mensch entscheiden. Zunächst sollte dabei sichergestellt werden, dass die durch PAC geprüften PDF/UA- sowie WCAG-Anforderungen erfüllt sind. Im Anschluss kann die KI-Prüfung ggf. eine Zeitersparnis bei der weiteren Optimierung eines PDF-Dokuments bieten. Auch bei der KI-Prüfung handelt es sich nicht um offizielle technische Kriterien und die KI-Prüfung muss nicht erfüllt sein, um das PDF-Dokument als barrierefrei auszuweisen.

## axesCheck

Eine Möglichkeit, PDF-Dateien online zu überprüfen, bietet axesCheck, eine Online-Version des PAC. Hierbei können im Browser direkt PDF-Dokumente hochgeladen werden. Diese werden online geprüft und das Ergebnis wie im PAC angezeigt. Dadurch wird das Testen unter Linux und MacOS möglich. Aus Sicht des Datenschutzes sollte hier beachtet werden, dass das PDF-Dokument auf andere Server hochgeladen wird, um ausgewertet werden zu können. Die Datenschutzbestimmungen sind einsehbar. Das Online-Tool kann komplett kostenlos benutzt werden.

## Programminterne Prüfung

Aktuell am Markt befindliche Programme (z. B. Adobe, Foxit, Tungsten Automation) beinhalten eigene Tests zum Überprüfen der Barrierefreiheit der erstellten PDF-Dateien. Diese befinden sich z. B. beim Adobe Acrobat Pro im Werkzeug Barrierefreiheit, welches rechts in der Werkzeugleiste geöffnet werden kann. Bei der Barrierefreiheitsprüfung von Adobe werden einzelne Merkmale einer barrierefreien Datei überprüft. Wichtig: Diese Prüfung stimmt nicht mit den PDF/UA-Kriterien überein. Die Beseitigung dieser Fehler führt also nicht zwangsläufig zu einer PDF-Datei, welche den PDF/UA-Standard erfüllt. Es werden außerdem auch Fehler gemeldet, die laut PDF-Standard und PDF/UA-Standard keine Fehler sind. Im Ergebnis der Prüfung können Sie sich Erklärungen zu den Fehlern anschauen und sich Lösungsvorschläge anzeigen lassen.

## PDF Accessibility Validation Engine (Pave)

Auf der Webseite von Pave können PDF-Dokumente direkt hochgeladen werden. Bei kleineren Dateien werden anschließend konkrete Tipps zum Beseitigen der Fehler in der Barrierefreiheit gegeben. Zusätzlich können einzelne Fehler direkt vom Online-Tool beseitigt werden und Sie können sich die optimierte Datei wieder herunterladen. Aus Sicht des Datenschutzes sollte auch hier beachtet werden, dass das Dokument auf andere Server hochgeladen wird und es erst dann ausgewertet bzw. modifiziert werden kann. Leider scheint es seit 2016 keine großen Updates mehr gegeben zu haben. Die Ergebnisse bilden aber trotzdem bereits eine gute Grundlage einer barrierefreien PDF-Datei.

## axesPDF

Das kostenpflichtige Tool axesPDF der Firma axes4 bietet neben dem integrierten PAC-Test auch die Möglichkeit, Fehler im Tagging direkt zu entfernen. Dadurch lässt sich viel Zeit beim Nachbearbeiten von Dateien sparen. Um die einzelnen Werkzeuge des Programms sinnvoll nutzen zu können, sollten dennoch Kenntnisse zum PDF/UA-Standard vorliegen. Aktuell läuft axesPDF nur auf Windows-Betriebssystemen.

## veraPDF

veraPDF wurde im Rahmen des PREFORMA Projekts als EU-geförderter PDF-Validator vom gleichnamigen veraPDF-Konsortium unter Führung der Open Preservation Foundation und in enger Zusammenarbeit mit der PDF Association entwickelt. Zielsetzung war dabei insbesondere, die Community für Langzeitarchivierung und die Software-Industrie zusammenzubringen und einen leistungsstarken Open-Source-Validator für PDF/A bereitzustellen.

veraPDF 1.0 wurde im Juni 2017 veröffentlicht und wurde seitdem kontinuierlich weiterentwickelt. Neben den erweiterten Konformitätsleveln PDF/A wurde mit PDF/UA, PDF 2.0 (ISO32005), dem Arlington Model und WTDPF (Well Tagged PDF) die Unterstützung weiterer Standards eingeführt. Die jeweils aktuelle Version für die Betriebssysteme Windows, Mac sowie Linux steht kostenfrei zur Verfügung.

## Prüfung der Barrierefreiheit mit veraPDF

veraPDF bietet ein eigenständiges Validierungsprofil für PDF/UA. Für eine spezifische Prüfung ist das Konfomitätslevel (PDF flavour) entsprechend auszuwählen.

## Unterschied zu PAC

veraPDF orientiert sich streng am PDF/UA-Standard und stellt keine Vermutungen zu möglicherweise existierenden Problemen an. Im Unterschied zu PAC bietet veraPDF außerdem keinen separat ausgewiesenen Qualitätscheck. Die Anforderungen des PDF/UA-Standards umfassen “must”-Forderungen, listen aber auch “should”-Forderungen auf. PAC und veraPDF beziehen die Kategorie der “should”-Forderungen in unterschiedlichem Maße in die Prüfung mit ein.

## Allgemeine Hinweise zur Nutzung von veraPDF

Die Anwendung muss nach dem Herunterladen bei der Installation in einem geeigneten Verzeichnis entpackt werden. Das gewählte Verzeichnis sollte insbesondere über Schreibrechte verfügen.

Für den Programmstart der Variante mit grafischer Oberfläche steht die ausführbare Datei verapdf-gui.bat zur Verfügung. Unter Windows öffnet sich mit Doppelklick als erstes ein Terminal-Fenster und sodann das Programmfenster. Unter Linux wird ./verapdf-gui im Terminal aufgerufen. Im Programmfenster lässt sich im Anschluss via [Choose PDF] eine Datei laden und mit [Execute] prüfen. Wird das Programmfenster via [X] geschlossen, schließt sich im Anschluss unter Windows auch das Terminal-Fenster, und zwar automatisch.

Das Ergebnis der Prüfung mit veraPDF kann als HTML zur Anzeige gebracht und als Report gespeichert werden. In diesen Dokumenten wird zu jedem Fehler zur Erläuterung die entsprechende Stelle des PDF /UA-Standards genannt. Zur besseren Interpretation der Prüfergebnisse mit veraPDF lässt sich das kostenlose Tool zur PDFix Lite von PDFix einsetzen, welches auch für MacOS verfügbar ist.

### PDFix Desktop mit veraPDF

Die kostenpflichtige Anwendung PDFix Desktop Pro für Windows oder MacOS hat die veraPDF Validierung integriert und zeigt die Fehler direkt in der PDF an. Eine Besonderheit ist die automatische Korrektur etwaiger Fehler mit Hilfe von speziellen Aktionen für das komplette Dokument. Auf diese Art können einzelne, aber auch größere Mengen im Stapelbetrieb aufbereitet werden. Mit dem PDFix SDK kann der Prozess serverseitig abgebildet und integriert werden.

## Verweise

### Interne Links

Keine vorhanden

### Externe Links in diesem Artikel

- Prüfkriterien für PDF/UA-Konformität (Matterhorn-Protokoll in der deutschen Version vom 23.06.2016 im Auftrag von BIT inklusiv) (PDF Association)
- PDF Accessibility Checker (PAC) (PAC, gewartet von axes4)
- Aufzeichnungen von Webinaren von Axes4 (axes4)
- PAC Fehler reparieren mit Adobe Acrobat (Jan Hellbusch)
- PDF/UA Probleme lösen mit axesPDF (axes4)
- axesCheck online PDF Prüfung (axes4)
- PDF Accessibility Validation Engine (PAVE) – PDF-Barrierefreiheit Überprüfen und Verbessern (PAVE)
- axesPDF (axes4)
- veraPDF (veraPDF consortium)
- PDF Validierung (PDFix)
- PDFix (PDFix)

### Weiterführende Links

- Prüfkriterien für PDF/UA-Konformität (Matterhorn-Protocoll in der englischen Version von 2021) (PDF Association)
- Die KI-Funktion in PAC 2026 – ein Schritt nach vorn? (anatom5, Dezember 2025)
- PAC-Fehler – grüne Häkchen sind kein Beweis (anatom5, Januar 2026)
- Bericht zum Fachtag KI und Automatisierung am 1. April 2026 in Hamburg, mit Vortrag von axes4 inklusive Unterlagen zum Download (Kompetent Barrierefrei, Kompetenzzentrum für ein barrierefreies Hamburg)

# Produktionsprozess für barrierefreie Lernmaterialien

Online betrachten

## Allgemeine Einordnung

Die Arbeitsgruppe SBS der TU Dresden verwendet seit mehr als 20 Jahren **HTML ohne Javascript für Skripte und Bücher**. Der Produktionsprozess wird durch ein Ticketsystem verwaltet und die Studierenden erhalten am Ende eine E-Mail zum Download des fertigen Buches bzw. einzelner Kapitel. Seit 2012 erstellen die Hilfskräfte Markdown und daraus wird das Buch in HTML generiert sowie verschiedene automatische und manuelle Qualitätsprüfungen vorgenommen. Vorlagen mit Bildern von mathematischen Formeln werden derzeit mit Mathpix nach LaTeX konvertiert und als Bildbeschreibung mit dem Bild für blinde und sehbehinderte Lesende integriert. Lange Bildbeschreibungen werden in einer separaten Datei abgespeichert und können entsprechend formatiert werden. Beispielsweise werden von Diagrammen die Wertetabellen angegeben. Für die Extraktion von Diagrammdaten wird unter anderem das Programm DigitizeIt verwendet. Der genaue Aufbau wird fortlaufend dokumentiert.

## Vorteile des Verfahrens

- plattformunabhängig, da Browser breit verfügbar sind
- unabhängig von konkreten assistiven Technologien, d. h. lesbar per Braillezeile oder Sprachsynthese bzw. mit Vergrößerungsprogramm
- Mehrsprachigkeit (gute Erfahrungen mit Sprachenausbildung Englisch, Deutsch, Französisch, Japanisch)
- lange Bildbeschreibungen mit fachlich korrekten Begriffen möglich, und leicht zwischen Text und langer Bildbeschreibung navigierbar
- Unterstützung graphischer Notationen wie UML, Schaltpläne, Zustandsübergangsdiagramme, ...
- Schriftkompetenz in wissenschaftlichen Notationen, die für schriftliche Prüfungen (Klausuren) notwendig ist, wird erworben
- für alle lesebehinderten Lesenden (blinde, sehbehinderte, dyslexisch)

## Nachteile des Verfahrens

- manuelle Bearbeitung benötigt ausreichend zeitlichen Vorlauf, Kosten für manuelle lange Bildschreibungen und Lektorat
- Zitierfähigkeit durch Seitenangaben, aber kein Erhalt des originalen Layouts

Das zentrale Format ist ein angepasstes Markdown und kann so in verschiedene Formate exportiert werden. Die Anpassungen wurden vorgenommen, um Seitenzahlen und Textboxen/-rahmen umzusetzen. Mathematische Formeln werden einfach z. B. mit Mathpix per Screen-OCR in LaTeX umgesetzt und bei der Konvertierung nach HTML in SVG umgewandelt. Im Alternativtext der SVG steht der LaTeX-Code. MathJax wird hier nicht verwendet.

## Verweise

### Interne Links

Keine vorhanden

## Externe Links in diesem Artikel

- Verfahren und Arbeitsprozess der AG Services Behinderung und Studium (Technische Universität Dresden)
- Mathpix (Mathpix)
- DigitizeIt. Digitizer software (Ingo Bormann)

## Weiterführende Links

- Erstellung barrierefreier Dokumente der AG Services Behinderung und Studium (Technische Universität Dresden)
- Dateienstruktur bei der Arbeitsgruppe Studium für Blinde und Sehbehinderte (Technische Universität Dresden)

# Hinweise zu Microsoft Office

Online betrachten

Im Folgenden sind Hinweise zur Erstellung von barrierefreien Microsoft Office Dokumenten zu finden.

## Inhaltsverzeichnis


- Microsoft Excel (10 min)
  - Anwendungsfälle
  - Grundsätzliche Überlegungen
  - Hinweise zur Erstellung von Excel-Arbeitsmappen
  - Prüfung der Barrierefreiheit
  - Verweise
- Microsoft Word (9 min)
  - Anwendungsfälle
  - Grundsätzliche Überlegungen
  - Werkzeuge
  - Hinweise zur Erstellung von Word-Dokumenten
  - Verweise
- Microsoft PowerPoint (5 min)
  - Anwendungsfälle
  - Hinweise zur Erstellung von PowerPoint-Dateien
  - Verweise

# Microsoft Excel

Online betrachten

## Anwendungsfälle

Microsoft Excel gehört zu den verbreitetsten Anwendungen für Tabellenkalkulationen. In vielen Unternehmen werden mit Excel auch komplexe grafische Auswertungen erstellt oder grundlegende statistische Datenanalysen durchgeführt. In diesem Artikel beziehen wir uns auf die Version Microsoft Excel 365. Es werden Hinweise gegeben, was getan werden muss, damit eine erstellte Excel-Arbeitsmappe barrierefrei ist und von allen Nutzenden gleichermaßen verwendet werden kann.

## Grundsätzliche Überlegungen

Bevor Sie sich für oder gegen die Verwendung von Excel zur Umsetzung Ihrer konkreten Anforderung entscheiden, sollten folgende Hinweise beachtet werden:

### Wie viel Text hat mein Dokument?

Bedenken Sie, dass Excel ein Tabellenkalkulationsprogramm, aber kein Texteditor ist. Menschen, die einen Screenreader für die Arbeit am PC nutzen, können nur sehr mühsam längere Texte in Excel-Tabellen lesen. Dies liegt daran, dass die Textstruktur (Absätze, Zeilenumbrüche, Formatierungen) vom Screenreader nicht dargestellt wird. Auch kann in Excel nicht ohne Weiteres wort- oder satzweise mit einem Screenreader navigiert werden, wie es z. B. in Microsoft Word möglich ist. Sollten Sie also ein sehr textlastiges und umfangreiches Dokument erstellen wollen, prüfen Sie die Umsetzung in Microsoft Word. Siehe Abschnitt Microsoft Word.

### Möchte ich Diagramme verwenden?

Wenn zur Visualisierung von Daten Diagramme verwendet werden sollen, wird hierfür ein Alternativtext benötigt. Dieser sollte das Diagramm kurz und sachlich beschreiben, sodass Lesenden mit Screenreadern die gleichen Informationen zur Verfügung stehen. Wenn die Datentabelle sowie das zugehörige Diagramm in derselben Arbeitsmappe bzw. Arbeitsblatt zur Verfügung gestellt werden, sollte der Alternativtext auf die Tabelle verweisen.

### Möchte ich ein Formular umsetzen?

Grundsätzlich kann ein Excel-Dokument auch zur Umsetzung eines Formulars verwendet werden. Achten Sie bei der Gestaltung darauf, dass Sie zwei Spalten verwenden. In der ersten Spalte sollte die Feldbezeichnung, also z. B. „Vorname“, stehen, in der zweiten Spalte direkt daneben sollten die entsprechenden Eingaben gemacht werden können. Sie können die Spalte mit den Beschriftungen der Formularfelder schützen, um zu verhindern, dass das Formular versehentlich verändert werden kann. Achten Sie darauf, dass die Spalte, in der die Eingaben vorgenommen werden sollen, editierbar bleibt.

### Wie viele Tabellen möchte ich erstellen?

Oft enthält eine Excel-Arbeitsmappe mehr als eine Tabelle. Achten Sie darauf, dass auf jedem Arbeitsblatt nur eine Tabelle dargestellt wird. Beschriften Sie das Arbeitsblatt entsprechend aussagekräftig, sodass eine schnelle Navigation zur gewünschten Tabelle möglich ist.

## Wie groß sollte eine Tabelle sein?

Oft werden sehr große Tabellen mit sehr vielen Spalten erstellt. Das hat für alle Nutzenden den Nachteil, dass diese schnell unübersichtlich werden. Für Nutzende, die eine Vergrößerungssoftware oder einen Screenreader verwenden, ist dies eine größere Herausforderung, da die Übersicht hierbei noch schneller verloren geht. Versuchen Sie daher, die Tabellen in der Breite so schmal wie möglich zu halten. Überlegen Sie bei sehr vielen Spalten, ob es möglich ist, die Daten sinnvoll in mehrere Tabellen aufzuteilen. Dies kann die Übersicht für alle erhöhen und stärkt damit die Barrierefreiheit. Darüber hinaus sollten verbundene Zellen vermieden werden.

## Hinweise zur Erstellung von Excel-Arbeitsmappen

Im Folgenden finden Sie einige wichtige Hinweise, um ein barrierefreies Excel-Dokument zu erstellen. Bitte beachten Sie darüber hinaus insbesondere auch die erweiterte Checkliste für Excel, um vertiefende Informationen zu erhalten.

## Formatieren von Daten als Tabelle

Auch wenn Daten in eine Arbeitsmappe eingegeben werden, die mit tabellarischen Koordinaten funktioniert, sind diese nicht automatisch als Tabelle formatiert. Eine formatierte Tabelle besitzt Spalten- und Zeilenüberschriften, die von assistiven Technologien, wie z. B. Screenreadern, ausgelesen werden können. Am einfachsten markieren Sie den kompletten Bereich, der als Tabelle formatiert werden soll. Anschließend klicken Sie auf die Schaltfläche „Start“ und wählen dann den Punkt „Als Tabelle formatieren“ aus. Es wird im Anschluss nach einem Tabellenformat gefragt. Achten Sie hier darauf, ein Format auszuwählen, welches ein gutes Kontrastverhältnis hat. Sie können die Auswahl auch später noch ändern. Außerdem können Sie festlegen, dass die Tabelle Spaltenüberschriften hat. Wenn der Dialog geschlossen wurde, befinden Sie sich in Ihrer neu formatierten Tabelle. Sie sehen nun einen neuen Reiter mit dem Namen „Tabellen Tools“. Hier können diverse Einstellungen für die Tabelle vorgenommen werden. Unter dem Punkt „Tabellenformatoptionen“ können die Spaltentitel aktiviert werden. Hierfür prüfen Sie, ob die Checkbox „Kopfzeile“ aktiviert ist, und aktivieren Sie diese gegebenenfalls. Beachten Sie, dass Zellen oberhalb einer Tabelle keine inhaltstragenden Informationen für die Tabelle enthalten sollten, da diese im Kontext der Tabelle für assistive Technologien nicht „verfügbar“ sind. Eine Zelle oberhalb einer als Tabelle ausgezeichneten Tabelle kann eine visuelle Überschrift enthalten. Diese rein visuelle Überschrift / Beschriftung sollte vergleichbar mit dem hinterlegten programmatischen Tabellennamen sein. Zusätzlich kann es sinnvoll sein, auch Zeilenüberschriften zu aktivieren. Haken Sie hierfür die Checkbox „Erste Spalte“ an.

## Gestalten von Überschriften

Oft ist es aus optischen Gründen wichtig, eine Tabelle mit einer „Überschrift“ zu versehen. Diese erfüllt einen reinen optischen Sinn. Die Überschrift sollte sich in der ersten Zeile in der ersten Spalte eines Arbeitsblattes befinden. Anstatt Zellen zu verbinden, können Sie alternativ die Option „Über Auswahl zentrieren“ verwenden. Die Überschrift sollte vergleichbar mit dem Tabellennamen sein.

## Festlegen von Zellenformaten

Ein festgelegtes Zellenformat hilft allen, den Inhalt schnell und gut zu erfassen. So ist es z. B. sehr hilfreich, Zellen, die Text enthalten, linksbündig auszurichten und Zellen mit Währungswerten entsprechend auch als Währung zu formatieren. Somit gibt ein Screenreader den Wert entsprechend korrekt aus. Wichtig hierbei ist, dass ein Screenreader auf den eigentlichen Wert der Zelle zurückgreift und nicht auf die

optisch veränderte Darstellung. So wird die Zahl „1.000“ trotz der Formatierung als „Ein-Tausend“ vorgelesen. Beachten Sie, dass optische Formatierungen wie z. B. kursiv oder fett oftmals erst nach zusätzlichen Einstellungen im Screenreader angesagt werden, also im Standard nicht gesprochen werden. Gleiches gilt für Einfärbungen von Zellen. Sie finden die Zellenformatierungen unter „Start/Formatvorlagen /Zellen/Formatieren/Zellen formatieren“. Hier können Sie entsprechende Formate wählen und anpassen.

## Umgang mit leeren Zellen

Generell sollten leere Zellen in einer Tabelle vermieden werden. Wenn eine Zelle keinen Wert enthalten kann, empfiehlt es sich, die Zelle mit „Keine Angabe“ zu füllen. Sollte dies die Sortierung der Tabelle stören, sollte die Zelle notfalls leer gelassen werden. Es sollte kein geschütztes Leerzeichen verwendet werden und auch keine anderen Sonderzeichen.

## Vermeiden von Eingabefehlern

Zu einem barrierefreien Excel-Dokument gehört auch, dass Benutzende so leicht wie möglich mit dem Excel-Dokument arbeiten können sollten für den Fall, dass Daten in die Tabelle eingetragen werden sollen. Zu diesem Zweck sollte die Eingabe nicht korrekter Daten möglichst verhindert werden. Hierzu bietet Excel z. B. die folgenden Möglichkeiten:

### Verwenden von Wertelisten

Wenn Sie z. B. ein Formular mithilfe von Excel umsetzen, kann es möglich sein, dass bestimmte Zellen als Werteliste formatiert werden können. In diesem Fall können Benutzende nur bestimmte Werte auswählen und keine eigenen Eingaben machen. Ein Beispiel hierfür wäre die Abfrage nach dem Geschlecht. Die Werte geben den Benutzenden zusätzlich Hinweise darüber, was genau angegeben werden muss. Hinweise zur Erstellung einer Werteliste finden Sie in diesem Artikel von Microsoft: Erstellen von Dropdownlisten - Microsoft-Support

### Fehlerhinweise bei nicht korrekter Eingabe

In Situationen, in denen eine Werteliste unpraktisch ist, kann es sinnvoll sein, den Benutzenden einen Hinweis zu geben, wenn ein falscher Wert eingegeben wurde. Ein Beispiel hierfür kann eine Zelle sein, in der ein Wert zwischen 1 und 100 eingegeben werden soll. Das wäre über eine Werteliste sehr umständlich zu realisieren. Excel bietet hier die Möglichkeit, über die Datenüberprüfung den eingegebenen Wert mit einer Art Regel abzugleichen und beim Nichtbestehen der Prüfung einen individuellen Fehlertext auszugeben.

Um einen Fehlerhinweis auszugeben und die Eingabe von Werten zu beschränken, gehen Sie wie folgt vor:

- Klicken Sie im Menü „Daten“ auf „Datentools“ und anschließend auf „Daten überprüfen“.
- Im sich nun öffnenden Dialog können Sie auf der Registerkarte „Einstellungen“ definieren, welche Werte eingegeben werden dürfen.
- Auf der Registerkarte „Eingabemeldung“ können Sie einen Hinweis anzeigen lassen, der den Benutzenden hilft, korrekte Werte anzugeben. Der Hinweis ist sichtbar, sobald die betreffende Zelle ausgewählt wird.
- Auf der Registerkarte „Fehlermeldung“ können Sie die Meldung konfigurieren, die erscheint, wenn ein nicht korrekter Wert angegeben wurde.

## Prüfung der Barrierefreiheit

Auch in Excel gibt es die Möglichkeit, eine automatische Barrierefreiheitsprüfung zu nutzen. Klicken Sie dafür auf „Überprüfen“ und anschließend auf „Barrierefreiheit“. Auch hier ist wichtig zu wissen, dass mit der jeweils neusten Version von Excel auch die Überprüfungsergebnisse am besten sind. Außerdem gilt auch bei dieser automatischen Überprüfung, dass nicht alle potenziellen Barrierefreiheitsfehler gefunden werden können. Um die Barrierefreiheit sicherzustellen, sollte also in jedem Fall zusätzlich manuell geprüft werden.

## Verweise

### Interne Links

- Microsoft Word

### Externe Links in diesem Artikel

- Anleitung zum Erstellen von Dropdownlisten (Microsoft)

### Weiterführende Links

- Bewährte Methoden für die Barrierefreiheit mit Excel-Tabellen (Microsoft)
- Erweiterte Checkliste nach EN 301 549 für barrierefreie Excel-Tabellen (Projekt SHUFFLE, Kompetenzzentrum Digitale Barrierefreiheit)

# Microsoft Word

Online betrachten

## Anwendungsfälle

Word gehört sowohl in der Bürokommunikation als auch in der Lehre zu den verbreitetsten Programmen zum Erstellen von Dokumenten. Es handelt sich hier um ein offenes Format, Änderungen können also direkt in der jeweiligen Datei vorgenommen werden.

Zum gemeinsamen Bearbeiten der Dokumente bieten sich grundsätzlich zwei Möglichkeiten an: Es kann die Kommentarfunktion benutzt werden, bei der auch die Möglichkeit besteht, auf einen Kommentar zu antworten. Alternativ hierzu kann der Änderungsmodus verwendet werden. Hierbei ist zu beachten, dass es bei vielen Änderungen an derselben Stelle sowohl für Menschen mit und ohne Beeinträchtigungen schwer werden kann, die jeweiligen Änderungen nachzuvollziehen. Es ist also generell sinnvoll, sich vorab auf eine Arbeitsweise zu verständigen.

Die Erstellung eines barrierefreien Dokuments ist ohne großen Aufwand möglich. Dafür ist es sehr hilfreich, mit Dokumentvorlagen zu arbeiten. Diese können vorab einmalig erstellt und anschließend von allen immer wieder genutzt werden. Des Weiteren ist es für die Barrierefreiheit hilfreich und nötig, die Funktionen von Word zu verwenden, wie sie gedacht sind, z. B. Tabellen nur für Datentabellen zu verwenden und nicht für Layout, ein Layout mit mehreren Spalten mit der Spalten-Funktion zu erstellen oder für Fußnoten die Fußnoten-Funktion zu verwenden.

Ein weiterer Vorteil von Word ist, dass ein Export in andere Formate möglich ist. Neben den eigens von Word unterstützten Formaten, können auch PDF, EPUB und andere Formate erstellt werden. Manche Formate benötigen ein Add-in andere können direkt mit Word erstellt werden. Die Barrierefreiheit der Endprodukte hängt vom Add-in ab.

**Achtung**: PDF-Dokumente sollten **nie** mittels PDF-Druckern erstellt werden. Damit wird aus jeder Seite ein Bild erstellt und somit ist jede Seite völlig unzugänglich für assistive Technologien. Steht kein Add-in für eine barrierefreie Konvertierung zur Verfügung, so sollte die Funktion „Speichern unter“ mit dem Dateityp „PDF“ verwendet werden.

## Grundsätzliche Überlegungen

Der Einsatz von Word eignet sich beispielsweise für Arbeitsblätter, Vorlesungsskripte sowie Dokumente mit hohem Textanteil und vergleichsweise schlichter Gestaltung, die editiert, ergänzt, bearbeitet, kommentiert und aktualisiert werden sollen. Word ermöglicht die barrierefreie Einbindung bzw. Aufbereitung von Tabellen, Abbildungen, Verzeichnissen und Fußnoten. Ebenso dient ein Word-Dokument als Ausgangsdatei für PDF-Dokumente und kann als zusätzliche Alternative zur Verfügung gestellt werden.

Der Einsatz von Word ist hingegen weniger geeignet für die Erstellung von Dokumenten mit hohem gestalterischem Anspruch, wie z. B. Plakate und Flyer, für Texte zur Veröffentlichung auf Webseiten oder für Dokumente, bei denen es auf eine einheitliche und verbindliche Darstellung ankommt, sowie für Dokumente, welche nicht editierbar sein sollen.

## Werkzeuge

Für Textdokumente und deren Export in andere Formate sind die folgenden Programme geeignet:

- **axesWord**: Als kostenpflichtiges Add-in dient axesWord dem Export standardkonformer PDF/UA-Dokumente in Word.
- **CommonLook Office**: Als kostenpflichtiges Add-in dient CommonLook Office dem Export standardkonformer PDF/UA-Dokumente in Word.
- **Tungsten Automation**: Als kostenpflichtiges Add-in dient Tungsten Automation dem Export standardkonformer PDF/UA-Dokumente in Word.
- **LibreOffice bzw. OpenOffice „Writer“**: Die Software „Writer“ stellt eine kostenfreie Alternative zu Microsoft Word dar. Es kann aus LiberOffice ein getaggtes PDF erstellt werden, jedoch kein standardkonformes PDF/UA-Dokument.
- **Microsoft Word**: Die Microsoft Word Software ist kostenpflichtig und verfügt über eine integrierte Barrierefreiheitsprüfung. Es kann aus Word ein getaggtes PDF erstellt werden, jedoch kein standardkonformes PDF/UA-Dokument.
- **Nitro PDF Pro**: Nitro PDF ist eine kostenpflichtige Software zum Erstellen und Bearbeiten von PDF-Dokumenten, bietet online jedoch auch kostenlose PDF-Tools an, z. B. „Word zu PDF“.
- **WordToEPUB**: Als kostenfreies Plug-in dient WordToEPUB dem Export barrierefreier EPUB-Dokumente in Word.

## Hinweise zur Erstellung von Word-Dokumenten

Grundsätzlich kann gesagt werden: Die korrekte Nutzung der Formatvorlagen und der Funktionen in Word ist die Grundlage für Barrierefreiheit.

Folgender Ablauf hat sich als Best Practice bewährt und ist demnach zu empfehlen:

### 1. Erstellung einer barrierefreien Dokumentvorlage

Um den Workflow zu beschleunigen und auch für unerfahrene Nutzende zu vereinfachen, ist zu Beginn eine universell einsetzbare Dokumentvorlage zu erstellen. Diese beinhaltet grundlegende Einstellungen bezüglich barrierefreier Gestaltung, wie z. B. Dokumentsprache, Metadaten, Formatvorlagen etc., sodass die Dokumentgestaltung und -formatierung im Anschluss wesentlich einfacher ist.

#### Tipps zur Erstellung von Dokumentvorlagen

- Wird die Dokumentvorlage als Word-Vorlage mit der Endung .dotx gespeichert, ist sie grundsätzlich vor Veränderungen geschützt.
- Ein häufig auftretender Fehler bei PDF-Dokumenten ist der fehlende Dokumenttitel. Dieser muss in Word in den Dokumenteigenschaften gesetzt werden. Die Dokumenteigenschaft “Titel” kann als Feld in der Dokumentvorlage z. B. auf einer Titelseite oder in einer Fußzeile integriert werden. Wird der Titel in dieses Feld geschrieben, so ist er automatisch in den Dokumenteigenschaften gesetzt und kann nicht mehr vergessen werden.

### 2. Erstellung eines Dokuments unter Berücksichtigung der Barrierefreiheitsanforderungen

In diesem Schritt kann die zuvor produzierte Vorlage genutzt werden, um ein neues Dokument zu erzeugen. Hier müssen lediglich neu eingefügte Funktionen bezüglich barrierefreier Gestaltung berücksichtigt werden, ansonsten ist nur noch der Inhalt selbst einzufügen.

### Tipps zur Erstellung eines Dokuments

- Falls die Dokumentvorlage als normales Word-Dokument erstellt wurde: Nutzen Sie auf jeden Fall eine Kopie und behalten Sie die ursprüngliche Vorlagendatei.
- Die Sprachhinterlegung von Textstellen, die von der Dokumentsprache abweichen, kann sehr zeitaufwändig werden. Die Verwendung von Word-Makros, die als Buttons in die Schnellwerkzeuge gelegt oder mit Tastaturkürzel hinterlegt werden, kann diesen Arbeitsschritt sehr beschleunigen.

## 3. Prüfung der Barrierefreiheit

Nachdem das neue Dokument erstellt wurde, ist es auf bestehende Barrieren zu prüfen. In Word selbst kann dies anhand der „Barrierefreiheit prüfen“-Funktion vorgenommen werden. Diese ist unter „Überprüfen – Barrierefreiheit überprüfen“ zu finden. Etwaige Fehler müssen im Anschluss korrigiert werden. Da mit dieser Prüfung nur eine geringe Zahl an möglichen Barrieren überprüft wird, ist eine zusätzliche manuelle Prüfung durch eine fachkundige Person zu empfehlen.

### Hinweise

Sollte die Option „Barrierefreiheit überprüfen“ in Word 2016 oder älter im Menü „Überprüfen“ nicht angezeigt werden, so kann folgender Weg gewählt werden: Wählen Sie das Menü „Datei“. In der nun geöffneten Ansicht wird „Auf Probleme überprüfen“ als anklickbarer Bereich angeboten. Bei Mausklick auf diese Schaltfläche öffnet sich eine Auswahlliste, in der die Auswahl „Barrierefreiheit überprüfen“ enthalten sein könnte. Ältere Word-Versionen enthalten diese Funktion allerdings noch nicht. Eine fehlerfreie Barrierefreiheitsprüfung in Word garantiert noch kein barrierefreies PDF.

## 4. Export in andere Formate

Je nach Einsatzgebiet wird nicht das finale Word-Dokument veröffentlicht, sondern ein daraus erzeugtes, neues Format, wie z. B. PDF. Die Details zu dieser Vorgehensweise finden Sie in den im Folgenden aufgelisteten Anleitungen.

## Verweise

### Interne Links

Keine vorhanden

### Externe Links in diesem Artikel

- axesWord (axes4)
- CommonLook Office (Allyant)
- Tungsten Power PDF (Tungsten Automation)
- LibreOffice bzw. OpenOffice „Writer“ (LibreOffice The Document Foundation)
- Microsoft Word (Microsoft)
- Nitro PDF Pro (Nitro)
- Word zu PDF (Nitro)
- WordToEPUB (Daisy Consortium)

# Weiterführende Links

## Anleitungen

- Anleitungen für Microsoft Office 2007, 2013, 2016 und 2019 im PDF-Format (Technische Universität Dresden)
- Anleitung für Microsoft Word 2016 (Universität Potsdam)
- Anleitung für Microsoft Word 2016 im PDF-Format (Nadine Sohn, Technische Hochschule Köln)
- Anleitung im Video-Format für Microsoft Word (HessenHub - Netzwerk Digitale Hochschullehre Hessen)

## Checklisten

- 8-Punkte-Checkliste für barrierefreie Word-Dokumente (Deutsches Zentrum für barrierefreies Lesen, Überwachungsstelle in Sachsen)
- Checkliste im PDF-Format für barrierefreie Word-Dokumente (BALLON - Barrierearmes Lernen und Lehren Online, Universität Bremen)
- Erweiterte Checkliste nach EN 301 549 für barrierefreie Word-Dokumente (Projekt SHUFFLE, Kompetenzzentrum Digitale Barrierefreiheit)

## Informationen von Microsoft

- Informationen im Video-Format zu Office 365 (Microsoft)
- Informationen zu Word für Microsoft 365, Word 2021, Word 2019, Word 2016, Word 2013, Office Business, Microsoft Office 365 Small Business im Microsoft Support-Center (Microsoft)
- Informationen zu Word für Microsoft 365, Word für Microsoft 365 für Mac, Word für das Web, Word 2021 im Microsoft Support-Center (Microsoft)

## Weitere Links

- Linksammlung, u. a. mit Informationen zur Erstellung barrierefreier Word-Dokumente (Technische Universität Chemnitz)
- Linksammlung zu barrierefreien Dokumenten im PDF-Format (Deutsches Studierendenwerk)
- Regelwerk für die Berliner Verwaltung, Berliner Standards für Word (berlin.de)

# Microsoft PowerPoint

Online betrachten

## Anwendungsfälle

Die Anwendung Microsoft PowerPoint wird für verschiedene Arten von Präsentationen eingesetzt. So werden neben der Vorstellung kleiner Projekte auch ganze Vorlesungen mit diesem Präsentationsprogramm realisiert. In allen Anwendungen ist auf einen barrierefreien Einsatz zu achten. Neben dem Export in ein PDF-Dokument werden Präsentationsfolien auch häufig direkt als PowerPoint-Dokument zur Verfügung gestellt. Eine komplett ohne Barrieren umgesetzte Präsentation wird in beiden Fällen zu guten Ergebnissen führen.

## Hinweise zur Erstellung von PowerPoint-Dateien

### Dokumenttitel

Auch in PowerPoint ist ein korrekter Dokumenttitel zwingend erforderlich. Hinzufügen können Sie diesen über den Reiter „Datei“ und dann unter dem Punkt „Informationen“. Dort finden Sie unter den Eigenschaften das Feld „Titel“. Hier können noch weitere Dokumenteigenschaften hinzugefügt werden. Für die Barrierefreiheit ist jedoch nur der Titel verpflichtend.

### Prüfung der Barrierefreiheit

PowerPoint bietet eine integrierte Prüfung auf Barrierefreiheit der Präsentation an, welche je nach Version an unterschiedlichen Stellen zu finden sein kann. In PowerPoint-Version von Microsoft 365 lässt sich diese über den Reiter „Überprüfen“ im Punkt „Barrierefreiheit überprüfen“ starten. Anschließend werden dadurch im geöffneten Fenster Fehler und Warnungen angezeigt, welche beseitigt bzw. überprüft werden müssen. Microsoft stellt hierzu Regeln für die Barrierefreiheitsprüfung und ein Video mit Beispielen zur Verfügung. Mit steigender Aktualität der PowerPoint-Version werden auch die Ergebnisse dieser Prüfung immer besser sowie mehr Fehler gefunden und die Beseitigung dieser Fehler vereinfacht. Microsoft bietet zur Unterstützung eine offizielle Anleitung zum Gestalten barrierefreier PowerPoint-Präsentationen für PowerPoint 365 an. Wie bei jeder automatischen Überprüfung gilt auch in diesem Fall, dass nicht alle potenziellen Barrierefreiheitsfehler gefunden werden können. Um die Barrierefreiheit sicherzustellen, sollte also in jedem Fall zusätzlich manuell geprüft werden.

### Einsatz der häufigsten Elemente / Objekte (z. B. Bilder, Listen, Absätze, Formeln, Diagramme & Tabellen)

Alle eingesetzten Elemente bzw. Objekte müssen barrierefrei gesetzt werden. Auch in PowerPoint benötigen demzufolge Bilder einen Alternativtext (Ausnahme: rein dekorative Bilder), Listen müssen sauber als Listen platziert werden, etc. Eine Anleitung der TU Dresden zur barrierefreien Gestaltung bietet hilfreiche Tipps.

### Lesereihenfolge

Ein wichtiger Punkt, der sich hinsichtlich der Barrierefreiheit in PowerPoint abhängig von der Programmversion bearbeiten lässt, ist die Anordnung der Lesereihenfolge aller eingesetzten Elemente.

**Bis PowerPoint 2019:** Die Lesereihenfolge lässt sich bis Programmversion 2019 über das Fenster Auswahlbereich überprüfen bzw. korrigieren. Der Auswahlbereich lässt sich im Reiter „Start“ unter dem Icon „Anordnen“ anzeigen. Hier finden sich alle eingefügten Elemente, welche in der eingefügten Reihenfolge angezeigt werden. Screenreader lesen die Elemente in **umgekehrter Reihenfolge**, also von unten nach oben, vor. Somit müssen hier die Elemente korrekt sortiert werden und schon passt die Lesereihenfolge der Elemente.

**Ab PowerPoint 2021 und in Microsoft 365:** Neue Versionen von PowerPoint bieten das Tool Lesereihenfolgebereich an. Dieses lässt sich über den Reiter „Überprüfen“ im Punkt „Barrierefreiheit überprüfen“ aufrufen. Es wird die Reihenfolge aller Elemente auf einer Folie als tatsächliche Lesereihenfolge in einer Liste angezeigt. Die Lesereihenfolge kann korrekt angeordnet werden. Zusätzlich können einzelne Elemente aus der Lesereihenfolge ausgeschlossen werden, in dem die Checkbox für das Element deaktiviert wird. Dies sollte nur für rein dekorative Elemente genutzt werden, die keine Informationen vermitteln.

## Erzeugung von PDF-Dokumenten mit Microsoft PowerPoint

Die Erzeugung eines PDF-Dokuments aus Microsoft PowerPoint heraus muss über „Speichern unter“ erfolgen. Hierbei ist es wichtig, dass die Dokumenteigenschaften und die Dokumentstruktur in den Optionen mit aktiviert sind. PDF-Drucker können hierzu nicht eingesetzt werden.

Anmerkung: Das aus Microsoft PowerPoint erzeugte PDF-Dokument ist in der Regel nicht barrierefrei nach PDF/UA. Um PDF/UA-konforme Dokumente zu erzeugen, muss aktuell auf eine kostenpflichtige Software zurückgegriffen werden (z. B. CommonLook Office, PDFix Desktop oder axesSlide).

## Verweise

### Interne Links

Keine vorhanden

### Externe Links in diesem Artikel

- Regeln für die Barrierefreiheitsprüfung (Microsoft)
- Video zur Barrierefreiheitsprüfung mit Beispielen (Microsoft)
- Anleitung zur Erstellung von PowerPoint Präsentationen (Microsoft)
- Anleitungen für Microsoft Office 2007, 2013, 2016 und 2019 im PDF-Format (Technische Universität Dresden)
- CommonLook Office (Allyant)
- axesSlide (axes4)
- PDFix Solutions (PDFix)

### Weiterführende Links

- Erweiterte Checkliste nach EN 301 549 für barrierefreie PowerPoint Präsentationen (Projekt SHUFFLE, Kompetenzzentrum Digitale Barrierefreiheit)

# Hinweise zu PDF-Dokumenten

Online betrachten

Im Folgenden sind Hinweise zur Erstellung von barrierefreien PDF-Dokumenten zu finden.

## Inhaltsverzeichnis


- PDF-Dokument (14 min)
  - Allgemeine Einordnung
  - Anwendungsfälle
  - Hinweise zur Erstellung von barrierefreien PDF-Dokumenten
  - Anwendungen zur Nachbearbeitung von PDF-Dateien
  - Exkurs PDF/UA-2: Überblick über den neuen Standard für barrierefreie PDFs
  - Verweise
- PDF-Formular (5 min)
  - Anwendungsfälle
  - Hinweise zur Erstellung eines barrierefreien PDF-Formulars
  - Verweise
- PDF aus Adobe InDesign (4 min)
  - Anwendungsfälle
  - Hinweise zur Erstellung von barrierefreien PDF-Dokumenten aus InDesign
  - Verweise
- PDF mit Apache-FOP (14 min)
  - Anwendungsfälle
  - Hinweise zur Erstellung von barrierefreien PDF-Dokumenten mit Apache-FOP
  - Spezialfälle
  - Einzelne Strukturelemente
  - Verweise
- PDF mit jsPDF-UA (21 min)
  - Allgemeine Einordnung
  - Anwendungsfälle
  - Hinweise zur Erstellung von barrierefreien PDF-Dokumenten mit jsPDF-UA
  - Spezialfälle
  - Einzelne Strukturelemente
  - Verweise
- Entscheidungshilfe zu PDF Konvertern - Software zum PDF-Export (8 min)
  - Allgemeine Einordnung
  - Testfälle
  - Barrierefreiheit des User Interface
  - Verweise
- Adobe PDF Accessibility Auto-Tag API (4 min)
  - Allgemeine Einordnung
  - Verweise
- Elektronische Signatur im PDF (4 min)
  - Anwendungsfälle
  - Signieren mit Adobe
  - Qualifizierte elektronische Signaturen in der D-A-CH-Region
  - Verweise

- Unterschied zwischen PDF- und anderen Dokumenten für assistive Technologien (7 min)

# PDF-Dokument

Online betrachten

## Allgemeine Einordnung

Ausgeschrieben lautet der Begriff: **P**ortable **D**ocument **F**ormat

Der PDF-Standard wurde von Adobe entwickelt, damit Dokumente plattformunabhängig dargestellt und ausgetauscht werden konnten. Dokumente konnten damit zwischen Computern mit verschiedenen Betriebssystemen ausgetauscht werden. Auch beim Ausdrucken auf unterschiedlichen Druckern ist das PDF von Bedeutung, da damit das Layout unverändert bleibt.

Mittlerweile ist der PDF-Standard ein DIN ISO-Standard (aktuellste Version: DIN ISO 32000-2:2020:12) und wird von der ISO mit Hilfe der PDF Association weiterentwickelt. Es gibt mehrere Substandards, z. B. PDF /UA für barrierefreie PDFs, PDF/A zur Langzeit-Archivierung oder den Druck-Standard PDF/X.

Der PDF/UA-Standard und der PDF/A-Standard können gleichzeitig einem PDF-Dokument zugewiesen werden. Es gibt Konvertersoftware, die bei der Konvertierung anbietet, ein PDF-Dokument mit beiden Standards zu erzeugen. Alternativ können PDF/UA- und PDF/A-Werkzeuge in Kombination eingesetzt werden.

Eine PDF-Datei lässt sich schnell mit unterschiedlichen Werkzeugen erzeugen. Einige Werkzeuge erstellen jedoch keine barrierefreien PDF-Dokumente. Für Menschen mit Beeinträchtigungen und für assistive Technologien sind die so erzeugten PDF-Dokumente nicht oder nur begrenzt zugänglich.

### Was zeichnet ein barrierefreies PDF-Dokument aus?

Gemäß BITV 2.0 müssen barrierefreie PDF-Dokumente verschiedene Anforderungen erfüllen.

### Der PDF/UA-Standard

Damit von einem barrierefreien PDF gesprochen werden kann, muss dieses dem PDF/UA-Standard entsprechen. UA steht für „Universal Accessibility“.

Der PDF/UA-Standard ist ebenfalls ein ISO-Standard und ist ein Substandard des PDF-Standards DIN ISO 32000-1:2008-07 (PDF 1.7). Der derzeit anwendbare PDF/UA-Standard ist PDF/UA-1 (DIN ISO 14289-1:2016: 12). Er ist anwendbar für alle PDF-Dokumente, die als PDF 1.X erzeugt werden. Im März 2024 wurde der PDF/UA-2-Standard veröffentlicht. Der PDF/UA-2-Standard ist ein Substandard des aktuellsten PDF-Standards PDF 2.0. Weiterführende Informationen zu diesem Standard finden sich im Exkurs PDF/UA-2 sowie bei den Anforderungen zur Barrierefreiheit von Tabellen in PDF-Dokumenten.

Der PDF/UA-Standard stellt klar, wie ein barrierefreies Dokument aufgebaut sein muss.

### Die EN 301 549

Die EN 301 549 (aktuelle Version EN 301 549 v3.2.1 (2021-03) ist eine harmonisierte europäische Norm, die die Barrierefreiheitsanforderungen an IKT-Produkte und -Dienstleistungen definiert. Darunter fallen auch PDF-Dokumente. In Abschnitt 10 „Nicht-Web-Dokumente“ werden die Anforderungen für Dokumente wie barrierefreie PDF-Dokumente erläutert.

Während die EN 301 549 die übergeordneten Anforderungen festlegt, bietet der PDF/UA-Standard konkrete technische Anforderungen zur Erstellung und technischen Prüfung barrierefreier PDF-Dokumente.

## Anwendungsfälle

PDF-Dokumente werden häufig eingesetzt, um unverfälschte und layoutgetreue Dokumente wiederzugeben, die komplett unabhängig vom eingesetzten System sind. Mit Standard-Programmen können schnell PDF-Dokumente erstellt werden. Jedoch sind diese meist nicht automatisch barrierefrei. Die Auswahl des korrekten Dateiformats für den jeweiligen Anwendungsfall ist zu beachten.

## Hinweise zur Erstellung von barrierefreien PDF-Dokumenten

Es gibt zahlreiche Anwendungen, mit denen PDF-Dateien barrierefrei nachbearbeitet werden können, z. B. Foxit PDF Editor, Tungsten Power PDF, PDFix Desktop, axesPDF und mehr. In dieser Handreichung wird beschrieben, worauf bei der Erstellung barrierefreier PDF-Dokumente zu achten ist. Es werden keine genauen Schritte in einer bestimmten Software beschrieben.

## Maschinenlesbar

Eine Grundvorrausetzung für ein barrierefreies PDF-Dokument ist die Maschinenlesbarkeit. Ein Dokument, das z. B. nur als Bild eingescannt wurde, kann nicht von Screenreadern gelesen werden, und darf daher nicht benutzt werden. Ob ein Dokument maschinenlesbar ist, kann z. B. durch das komplette Markieren des Textes mit der Maus oder Tastatur getestet werden. Wenn der Text sich markieren, und somit auch kopieren lässt, kann der Text als grundsätzlich maschinenlesbar betrachtet werden. Wird die ganze Seite mit einem Klick oder gar nicht markiert, handelt es sich um ein Bild des Textes. Der Text ist dann nicht maschinenlesbar. Maschinenlesbarkeit allein reicht nicht aus, um die Barrierefreiheit zu erfüllen.

## Dokumenttitel + Einstellungen

Um den Inhalt eines PDF schnell zu erfassen und um beispielsweise Dateien sauber katalogisieren zu können, ist ein aussagekräftiger Dokumenttitel essenziell. Von allen möglichen Metadaten (verfassende Personen, Stichwörter ...) ist der Titel für ein barrierefreies Dokument notwendig und in PDF/UA zwingend vorgeschrieben. Das Hinterlegen weiterer Metadaten führt zur Steigerung des Nutzungskomforts des Dokuments. So kann z. B. Transparenz hinsichtlich der Personen, die das Dokument verfasst haben, für eventuelle Rückfragen hergestellt werden. Der Dokumenttitel muss zusätzlich auch in der Titelleiste des PDF-Betrachters als Fenstertitel angezeigt werden. Der Screenreader wird dann beim Wählen des jeweiligen Fensters diesen auch vorlesen. In den Dokumenteinstellungen sollte eingestellt werden, dass - soweit vorhanden - die Lesezeichen beim Öffnen des Dokuments direkt angezeigt werden. Lesezeichen sind für die Navigation in Screenreadern wichtig, erleichtern aber auch Menschen mit Sehvermögen die Navigation im Dokument. Lesezeichen, die auf Überschriften verweisen, sollten denselben Wortlaut und dieselbe Hierarchie aufweisen wie die zugehörige Überschrift. Andernfalls wird die Orientierung für Personen, die auf einen Screenreader angewiesen sind, erschwert.

## Kontraste prüfen

Nach der harmonisierten europäischen Norm EN 301 549 müssen alle Farbkontraste in Dokumenten die Konformitätsstufe AA der WCAG erfüllen. Text und Hintergrund müssen ein Kontrastverhältnis (Kontrastabstand) von mindestens 4,5 zu 1 für normalen Text (Schriftgröße kleiner 18 pt oder 14 pt fett), bzw. mindestens 3 zu 1 bei größeren Schriften sowie informativen grafischen Elementen/Grafiken aufweisen. Zur Prüfung des Kontrastverhältnisses gibt es verschiedene Möglichkeiten bzw. Tools. Der

Colour Contrast Analyser ist eine kostenfreie Mini-Anwendung. Es öffnet sich ein kleines Fenster auf dem Bildschirm - mit Pipetten kann jeweils die Vordergrund- und Hintergrundfarbe ausgewählt werden. Der Colour Contrast Analyser zeigt dann das Verhältnis der ausgewählten Farben für die Konformitätsstufe AA und für die Konformitätsstufe AAA an.

Sollten die Kontraste in einem PDF-Dokument nicht die Konformitätsstufe AA erreichen, müssen an den entsprechenden Stellen neue Farben gewählt werden. Der Colour Contrast Analyser kann hierbei mit Schiebereglern bei der Farbwahl unterstützen. Oftmals genügt es einzelne Farben etwas heller bzw. dunkler umzusetzen. Die korrekte Farbwahl bzw. Farbanpassung sollte allerdings im Quelldokument (Word, InDesign, PowerPoint ...) erfolgen. Ein nachträgliches Bearbeiten der Farben im PDF ist möglich, da es sich bei PDF um ein objektorientiertes Datenformat handelt. Es bedarf allerdings geeigneter Werkzeuge, da sonst unter Umständen die interne Dokumentstruktur beschädigt wird.

## Dokumentstruktur

Barrierefreie Dokumente sind strukturiert aufgebaut. Alle relevanten Informationen müssen mit inhaltlich passenden Tags (Etiketten) versehen werden. Dadurch werden Inhaltselemente wie z. B. Überschriften, Listen, Tabellen, Grafiken etc. korrekt programmatisch ausgezeichnet, also getaggt. Über die Tags werden die verschiedenen Inhaltselemente für Hilfstechnologien wie Screenreader zugänglich gemacht. Neben der Kenntnis der Tags müssen auch die Schritte zum Tagging bekannt sein und korrekt angewendet werden. Ein barrierefreies PDF-Dokument ist komplett und in der richtigen Lesereihenfolge getaggt. Alle dekorativen Elemente und sich wiederholende Kopf- und Fußzeilen sind als Artefakte ausgezeichnet.

## Sprache (global und einzelne Teilbereiche)

Die Einstellung der korrekten natürlichen Sprache des PDF-Dokuments ist u. a. für die korrekte Nutzung mit Screenreadern notwendig. Nur dadurch werden Dokumente in der richtigen Sprache erkannt und vorgelesen. Wenn einzelne Abschnitte von der natürlichen Hauptsprache des Dokuments (Dokumentsprache) abweichen, dann muss für diese Abschnitte die jeweils korrekte Sprache hinterlegt werden. Die globale Dokumentsprache kann im PDF in den Dokumenteigenschaften angegeben werden. Abweichende Teilbereiche sollten bereits im Quelldokument korrekt hinterlegt werden, da diese Nachbearbeitung im PDF sehr aufwändig ist.

## Lesereihenfolge

Um die Elemente eines PDF-Dokuments auch mit assistiven Technologien in der korrekten Reihenfolge lesen zu können, ist es notwendig, die definierten Tags korrekt anzuordnen bzw. zu sortieren. Gelesen wird die Tag-Struktur des Dokuments von oben nach unten.

## Alternativtexte

Alle informativen Abbildungen, Hyperlinks und Anmerkungen benötigen einen aussagekräftigen objektiven Alternativtext. Dieser kann sowohl im Quelldokument, als auch im Nachhinein im PDF hinterlegt werden.

## Schriften

PDF/UA verlangt, dass alle verwendeten Schriften vollständig im PDF-Dokument eingebettet sind und über eine korrekte Unicodezuordnung verfügen.

## Prüfung der Barrierefreiheit

Die Prüfung eines barrierefreien PDF-Dokuments kann mit dem PDF Accessibility Checker (PAC) oder [veraPDF] (https://verapdf.org) durchgeführt werden. Gefundene Fehler und ggf. Hinweise sollten vor der endgültigen Fertigstellung des Dokuments beseitigt werden. Je besser die Ursprungsdatei barrierefrei gestaltet ist, desto weniger Fehler werden bei den abschließenden Überprüfungen gefunden.

Der PAC und veraPDF überprüfen ca. zwei Drittel der Kriterien des PDF/UA-Standards automatisiert. Eine manuelle Sichtprüfung ist für die nur durch einen Menschen prüfbaren Prüfkriterien des PDF/UA-Standards notwendig. Der PAC kann dabei durch eine Anzeige in Form von HTML unterstützen. Eine Prüfung mit assistiven Technologien wie einem Screenreader ist als weitere Maßnahme der Qualitätssicherung zu empfehlen.

## Anwendungen zur Nachbearbeitung von PDF-Dateien

- axesPDF von axes4: Speziell entwickelt zur Nachbearbeitung von PDF-Dokumenten zur Erstellung eines barrierefreien PDF-Dokuments
- Adobe Acrobat Professional bietet Funktionen für die Barrierefreiheit sowie Funktionen zur Nachbearbeitung, die Acrobat Barrierefreiheitsprüfung prüft auf häufige Probleme, jedoch nicht den PDF/UA-Standard
- PDFix Desktop und SDK Desktop- und Serverapplikation für MacOS, Windows und Linux zur automatisierten Erstellung, Korrektur, Konvertierung und Validierung von PDF/UA.
- PDF-XChange Editor bietet Funktionen für die Barrierefreiheit sowie Funktionen zur Nachbearbeitung
- PDF Editor von foxit bietet Funktionen für die Barrierefreiheit sowie Funktionen zur Nachbearbeitung
- Common Look von Allyant bietet ein Plug-in für Adobe Acrobat Pro zur besseren Nachbearbeitung von PDF-Dokumenten

## Exkurs PDF/UA-2: Überblick über den neuen Standard für barrierefreie PDFs

Der PDF/UA-2-Standard, veröffentlicht als ISO 14289-2:2024-03, ist der neueste technische Standard für barrierefreie PDF-Dokumente. PDF/UA-2 bringt gegenüber seinem Vorgänger PDF/UA-1 Verbesserungen und Klarheit über die Anforderungen für barrierefreie PDF-Dokumente.

### Was ist PDF/UA-2?

Der PDF/UA-2 ist kein Update des PDF/UA-1-Standards (DIN ISO 14289-1). PDF/UA-1 basiert auf dem ISO-Standard für alle PDF 1.X, während PDF/UA-2 den ISO-Standard für alle PDF 2.0 (DIN ISO 32000-2:2020:12) darstellt.

Mit PDF/UA-2 wird ein besserer Reuse, also eine Wiederverwendung, von Inhalten aus PDF-Dokumenten, möglich. Zu diesen Wiederverwendungen zählen beispielsweise lineare Darstellungen der Inhalte für das bessere Lesen auf unterschiedlichen Bildschirmgrößen, der Textauszug aus PDFs oder auch die verbesserte Barrierefreiheit.

### Well-Tagged PDF

Mit PDF 2.0 hat die PDF Association auch „Well-Tagged PDF (WTPDF): Using Tagged PDF for Accessibility and Re-Udefiniert wiese in PDF 2.0“ veröffentlicht. Die Publikation definiert, wie Inhalte in PDF 2.0 aufgebaut und strukturiert werden müssen, um die Wiederverwendbarkeit und Barrierefreiheit für ein möglichst großes Spektrum an Nutzungsfällen zu erlauben. WTPDF hat zwei Konformitätslevel:

- Konformitätslevel 1: Für das Wiederverwenden und
- Konformitätslevel 2: Für die Barrierefreiheit.

## Überblick Zusammenhang PDF-Version und PDF/UA

Quelle: PDF Association

| PDF-Version | PDF/UA-1 | PDF/UA-2 & WTPDF |
|---|---|---|
| PDF 1.7 | Ja | Erst Upgrade auf PDF.2.0 |
| PDF 2.0 | Nein | Ja |

## PDF/UA-2: Was ist neu?

PDF/UA-2 basiert auf der aktualisierten PDF-Spezifikation ISO 32000-2 (PDF 2.0) und bietet eine Reihe von Verbesserungen:

- **Erweiterte Barrierefreiheitsfunktionen**: Unterstützung für semantische Strukturen, verbesserte Handhabung von Alternativtexten und erweiterte Metadatenfähigkeiten.
- **Neue Inhalte und Funktionen**: Mathematische Ausdrücke (MathML), mehr als sechs Überschriftenebenen, neue Tags wie Titel, Aside und Sub sowie die Integration von ePub.
- **Interoperabilität**: Namespaces ermöglichen eine umfassendere Nutzung und Wiederverwendbarkeit von Inhalten.
- **Detaillierte Anleitungen**: Klare Richtlinien zur Erstellung barrierefreier Dokumente, die die Bedürfnisse von Menschen mit Behinderungen besser berücksichtigen.

## Unterschiede zwischen PDF/UA-1 und PDF/UA-2

Der Hauptunterschied liegt in der zugrunde liegenden PDF-Spezifikation:

- **PDF/UA-1** basiert auf PDF 1.7, während **PDF/UA-2** auf PDF 2.0 aufbaut.
- PDF/UA-2 bietet eine verbesserte Unterstützung für moderne Technologien und eine robustere Dokumentstruktur.
- Die neuen Funktionen von PDF 2.0, wie die klare Definition von Tagging-Regeln und die Möglichkeit zur Wiederverwendung von Inhalten, machen PDF/UA-2 zukunftssicherer.

## Ist PDF/UA-2 bereits Stand der Technik?

Obwohl PDF/UA-2 viele Vorteile bietet, ist dieser derzeit noch nicht Stand der Technik. Der Grund dafür liegt in der fehlenden Unterstützung durch Software.

Für barrierefreie PDF-Dokumente ist das Zusammenspiel von PDF-Engines, PDF-Readern und assistiven Technologien notwendig. Aktuell gibt es noch keine zuverlässige Erzeugung, Ausgabe in Readern und Interpretation von PDF/UA-2 durch assistive Technologien.

Wann sich PDF/UA-2 als neuer Standard etabliert, hängt von der Entwicklung und Verbreitung kompatibler Software ab.

### Fazit zu PDF/UA-2

PDF/UA-2 stellt einen bedeutenden Fortschritt für barrierefreie PDF-Dokumente dar. Er bietet zahlreiche Vorteile gegenüber PDF/UA-1. Es wird erwartet, dass PDF/UA-2 in den kommenden Jahren zunehmend als Stand der Technik anerkannt wird, sobald die notwendige Softwareunterstützung verfügbar ist. Bis dahin bleibt es ein vielversprechender Standard mit großem Potenzial für die Zukunft der Barrierefreiheit.

## Verweise

### Interne Links

- Anforderungen zur Barrierefreiheit von Tabellen in PDF-Dokumenten

### Externe Links in diesem Artikel

- PDF Association (PDF Association)
- Colour Contrast Analyser (vispero)
- Übersicht zu den PDF-Tags (Stefan Brechbühl)
- Anleitung zum Erstellen und Ändern von Tags in Adobe Acrobat (Stefan Brechbühl)
- PDF Accessibility Checker (PAC) (PAC, gewartet von axes4)
- veraPDF (veraPDF)
- axesPDF (axes4)
- PDFix Desktop und SDK (PDFix)
- Adobe Acrobat Professional (Adobe)
- PDF-XChange Editor (1 for All Software GmbH)
- PDF Editor von foxit (Foxit Software Inc.)
- Common Look (Allyant)
- Well-Tagged PDF (WTPDF) (PDF Association)

### Weiterführende Links

- 9-Punkte-Checkliste für barrierefreie PDF-Dokumente (Sächsische Überwachungsstelle)
- Hilfe für LibreOffice zu Barrierefreiheit (PDF/UA) (LibreOffice The Document Foundation)
- PDF Techniken (PDF Association, Liaison Working Group PDF Accessibility)
- Webinare und Leitfäden zu PDF (Börsenverein des deutschen Buchhandels)

# PDF-Formular

Online betrachten

## Anwendungsfälle

PDF-Formulare sind nur geeignet für **einfache, ausfüllbare Dokumente, die unterschrieben oder elektronisch signiert** werden sollen, z. B. Verträge, Anträge. Einige Anforderungen an barrierefreie Formulare können von PDF-Formularen nicht vollständig erfüllt werden. Komplexe Datenabfragen, Umfragen oder Rückmeldungen sollten als barrierefreie HTML-Formulare umgesetzt werden, um alle gesetzlichen Anforderungen an die barrierefreie Nutzung zu erfüllen.

Derzeit gibt es keine kostenfreie Software, mit der PDF/UA-konforme Formulare erstellt werden können. Nachfolgend sind einige Möglichkeiten der Erstellung mit kostenpflichtigen Tools beschrieben.

- Wird ein Formular in Quelldokumenten, z. B. Word, Excel, LibreOffice vorbereitet, so müssen die Formularfelder im Nachhinein im PDF eingefügt werden. Dies ist mit dem kostenpflichtigen Tool Adobe Acrobat Pro möglich. LibreOffice bietet im Unterschied zu MS Word die zusätzliche Möglichkeit auch Formularfelder zu definieren.
- Mit der kostenpflichtigen Software CommonLook Office kann direkt aus Word ein PDF/UA-konformes Formular erzeugt werden, das ggf. mit Adobe Acrobat Pro nachbearbeitet werden muss. Wichtig ist, dass in Word nur die alten Formularelemente verwendet werden. Bei der Verwendung spezieller Merkmale der Formularfelder (z. B. Längenbeschränkung, Platzierung) muss mit Adobe Acrobat Pro nachgearbeitet werden.
- PDF-Formulare können in Adobe InDesign bereits mit Formularfeldern erstellt werden und mit dem kostenpflichtigen Plug-in MadeToTag barrierefrei in ein PDF-Dokument exportiert werden. Die Funktionen und das Aussehen von Formularfeldern lassen sich in Adobe InDesign ebenfalls nur begrenzt steuern, dies übernimmt ebenfalls MadeToTag.
- Oft ist eine Nachbearbeitung der Formularfelder bei Verwendung spezieller Merkmale oder eines bestimmten Erscheinungsbildes mit Adobe Acrobat Pro nötig. Mit der Anwendung PDFix Desktop lassen sich alle für die technische Barrierefreiheit relevanten Korrekturen vornehmen, die z. B. bei einer Validierung aufgezeigt werden.

Dynamische PDF-Formulare, sogenannte XFA-Formulare, bei denen dynamisch neue Inhalte oder neue Seiten im PDF eingefügt werden, können nicht barrierefrei gemacht werden. Es gibt allerdings auch die statische Variante als XFA-Formular, die den klassischen Acro-Forms vergleichbar ist und sich barrierefrei umsetzen läst.

## Hinweise zur Erstellung eines barrierefreien PDF-Formulars

Damit PDF-Formulare barrierefrei sind, muss Folgendes bei der Platzierung der Formularfelder zusätzlich berücksichtigt werden:

- Jedes Formularfeld benötigt eine aussagekräftige Beschriftung (Label), die von Screenreadern ausgelesen werden kann. In Adobe Acrobat Pro wird dies in der „Quickinfo“ hinterlegt. Die Beschriftung muss so formuliert sein, dass auch ohne umliegenden Text das Formularfeld korrekt ausgefüllt werden kann.
- Die Quick-Info Beschriftung muss wie die visuelle Beschriftung auch Formatvorgaben enthalten, wenn diese bei der Dateneingabe zu berücksichtigen sind.

- Screenreader können in den Formularmodus wechseln, in dem ausschließlich Formularfelder angesprungen werden und der Text rundherum übersprungen wird.
- Für jede Eingabe muss der passende Formularfeldtyp gewählt werden, z. B. Checkboxen für Mehrfachauswahl.
- Jedes Formularfeld muss in einem Form-Tag getaggt sein und im Tag-Baum an der korrekten Stelle platziert werden.
- Die Tabulatorreihenfolge muss korrekt gesetzt sein, damit auch bei Tastaturbedienung die Arbeitsreihenfolge stimmt.
- Pflichtfelder sollten gekennzeichnet sein: programmatisch in den Formularfeld-Eigenschaften, in der visuellen Beschriftung (Label) und in der Quickinfo.

## Verweise

### Interne Links

Keine vorhanden

### Externe Links in diesem Artikel

- CommonLook Office (Allyant)
- MadeToTag (axaio)
- PDFix Desktop (PDFix)

### Weiterführende Links

- MadeToTag: Allgemeine Anleitung (Video Tutorial in englischer Sprache) (axaio)
- MadeToTag: Anleitung zur Optimierung von Formularfeldern (Video Tutorial) (axaio)
- PDF: Verfahren zum Erstellen von barrierefreien PDF-Formularen (Adobe)
- Word: Anleitung zur Erstellung von barrierefreien PDF-Formularen aus Word (YouTube Video mit Untertiteln) (Universität Potsdam)
- Word: Tutorial zur Erstellung von barrierefreien PDF-Formularen aus Word (YouTube Video mit Untertiteln) (Kompetenzzentrum für Barrierefreiheit der Hochschule für Medien)

# PDF aus Adobe InDesign

Online betrachten

## Anwendungsfälle

Adobe InDesign ist ein Layout-Programm. Es ist geeignet für **Dokumente mit professionellem grafischem Layout** und/oder wenn die Dokumente auch für professionellen Druck (in einer Druckerei) gestaltet werden sollen.

Ab der Version CS5.5 bietet InDesign einige Funktionen, um Dokumente für die Barrierefreiheit vorzubereiten. Grundsätzlich kann gesagt werden, je neuer die Version umso besser sind die Funktionen für Barrierefreiheit.

Ein barrierefreies PDF nach PDF/UA-Standard (ISO-Standard 14289) kann allerdings mit InDesign **nicht** exportiert werden. Dazu benötigt es das kostenpflichtige Plug-in MadeToTag von axaio. Mit Adobe InDesign kann ein getaggtes PDF erstellt werden. Dieses muss danach noch bezüglich Barrierefreiheit bearbeitet werden, z. B. mit Adobe Acrobat Pro, PDFix Desktop (MacOS, Windows) oder axesPDF (Windows).

In Adobe InDesign können auch **PDF-Formulare** mit Formularfeldern vorbereitet und mit MadeToTag barrierefrei in PDF exportiert werden. Die Funktionen und das Aussehen von Formularfeldern lassen sich in Adobe InDesign nur begrenzt steuern, deshalb bietet MadeToTag die Möglichkeit zur Gestaltung mit Hilfe einer Parameter-Datei. Oft ist eine Nachbearbeitung der Formularfelder mit Adobe Acrobat Professional oder dem Spezialprogramm PDFix Desktop nötig. Mehr dazu finden Sie unter PDF-Formular.

## Hinweise zur Erstellung von barrierefreien PDF-Dokumenten aus InDesign

In Adobe InDesign kann das Quelldokument bereits gut für den Export eines barrierefreien PDFs vorbereitet werden. Die vorhandenen Funktionen, die Adobe InDesign bietet, müssen dafür korrekt angewendet werden. Das sind zum Beispiel die Funktion für Listen, Fußnoten, Inhaltsverzeichnisse, Absatzformate oder Tabellen.

### Dokumentstruktur

Die Dokumentstruktur kann mit den Absatzformaten bereits gut vorbereitet werden. Die Absatzformate müssen nach Dokumentstruktur erstellt und angewendet werden. In den Absatzformaten kann definiert werden, welcher Tag daraus im PDF erstellt werden soll (Tagexport). Die Möglichkeiten, die korrekten Tags in Adobe InDesign zu definieren, sind begrenzt und können mit MadeToTag erweitert werden.

### Lesereihenfolge

Die Lesereihenfolge im PDF kann mit verknüpften Textfeldern, verankerten Objekten und der Funktion „Artikel“ vorbereitet werden. Wird nichts zur Lesereihenfolge vorbereitet, so wird die Lesereihenfolge aus der zeitlichen Einfüge-Reihenfolge der Objekte in Adobe InDesign definiert. Diese ist meist nicht korrekt.

### Dokumenttitel

Der Dokumenttitel kann bereits in den Metadaten in Adobe InDesign korrekt vorbereitet werden.

## Dokumentsprache und abweichende Textstellen

Die Dokumentsprache wird aus der am meisten verwendeten Sprache ermittelt. Textstellen, die von der Dokumentensprache abweichen, können mithilfe von Absatz- oder Zeichenformaten oder einzeln im Fenster „Zeichen“ gekennzeichnet werden.

## Verweise

### Interne Links

- PDF-Formular

### Externe Links in diesem Artikel

- MadeToTag (axaio)
- PDFix Desktop und SDK (PDFix)

### Weiterführende Links

- Anleitungen zur barrierefreien Gestaltung von Dokumenten. Adobe InDesign. (TU Dresden, AG Services Behinderung und Studium, 2020)
- YouTube-Channel für Barrierefreiheit mit InDesign. (Klaas Posselt)
- Erstellen von barrierefreien PDF-Dateien. (Support-Seite von Adobe zur Erstellung von barrierefreien PDF-Dateien aus InDesign, wird stets aktuell gehalten)
- MadeToTag Handbuch. (Handbuch von axiao - Hersteller des Plug-ins MadeToTag - mit Schritt für Schritt-Anleitung zur Handhabung von MadeToTag InDesign, wird stets aktuell gehalten)

# PDF mit Apache-FOP

Online betrachten

## Anwendungsfälle

Unter Verwendung des sogenannten Apache Formatting Objects Processors (FOP) lässt sich auf der Grundlage von spezifischen XSL-FO-Layout-Beschreibungen insbesondere das Ausgabeformat PDF erzeugen. Im Unterschied zu jsPDF übernimmt Apache FOP die Wiederholung von Footer und Header sowie die Anordnung des Contents inklusive Seitenumbrüchen automatisch, insbesondere auch bei Fußnoten.

Als freie Java-Anwendung zur Verfügung gestellt, wird Apache-FOP von der ehrenamtlich zur Förderung von Apache-Software-Projekten arbeitenden Organisation Apache Software Foundation kontinuierlich weiterentwickelt. Eine allgemeine Einführung sowie eine umfassende Dokumentation zur Syntax von XSL-FO-Dateien auf Deutsch bietet die data2type GmbH.

Der Einsatz von Apache-FOP zur Erzeugung von PDF-Dokumenten, die den Test mit dem Prüftool PDF Accessibility Checker 2026 (PAC) vollständig bestehen, ist insbesondere dann empfehlenswert, wenn die dazu benötigte XSL-FO Preview automatisiert aus einem XML-Dokument generiert werden kann. In diesem Fall können mithilfe von XSL-Transformationen alle benötigten Voreinstellungen eingerichtet und ohne weiteren Anpassungsbedarf für gleichartige PDF-Dokumente genutzt werden. Dazu werden insbesondere die im Folgenden beschriebenen Einstellungen benötigt.

## Hinweise zur Erstellung von barrierefreien PDF-Dokumenten mit Apache-FOP

### Dokumenttitel

Die Einstellung des Dokumenttitels erfolgt in den fo:declarations:

```
1   <fo:declarations>
2     <x:xmpmeta xmlns:x="adobe:ns:meta/">
3       <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
4         <rdf:Description rdf:about="">
5           <dc:title>
6             <rdf:Alt>
7               <rdf:li xml:lang="x-default">Dokumenttitel</rdf:li>
8             </rdf:Alt>
9           </dc:title>
10        </rdf:Description>
11      </rdf:RDF>
12    </x:xmpmeta>
13  </fo:declarations>
```

### Anmerkungen

Die Dokumentation zum Feature Accessibility von Apache-FOP ist diesbezüglich nicht ganz vollständig.

In den fo:declarations könnten weitere Angaben eingetragen werden, wie z. B. zur Dokumentsprache oder zu „creator“ oder auch zu „description“ wie in der Apache-FOP Referenz ausgewiesen. Allerdings kann es

dann bei Verwendung des PDF/A-Standards für die Langzeitarchivierung zu Problemen beim Test mit dem Tool veraPDF kommen (siehe auch unten unter Prüfung der Barrierefreiheit sowie Abschnitt Hinweise zu Tools zur Überprüfung der Barrierefreiheit).

## Sprache (global)

Die Einstellung der globalen Dokumentsprache kann direkt im fo:root-Element vorgenommen werden:

```
1  <fo:root language="DE" country="DE">
```

## Sprache (einzelne Teilbereiche)

Sprachauszeichnungen werden aus fo:block-Elementen übernommen. Für einen Sprachwechsel innerhalb eines Absatzes z. B. von Deutsch zu Englisch kann das fo:inline-container-Element genutzt werden:

```
1  <fo:inline-container width="18.5mm"><fo:block xml:lang="en" role="Span">Bachelor</fo:block></fo:
inline-container>
```

## Alternativtexte

Den Elementen fo:external-graphic sowie fo:basic-link muss mittels fox:alt-text-Attribut ein Alternativtext zugewiesen werden.

### Anmerkungen

Für die Transformation mit Apache-FOP muss tatsächlich für alle fo:external-graphic- sowie fo:basic-link-Elemente ein Alternativtext bereitgestellt werden. D. h. insbesondere auch dann, wenn sich die Elemente in einem Bereich befinden, der als sogenanntes Artefakt gekennzeichnet ist (siehe auch unten unter Kopf- und Fußzeilen). Die Prüfung auf Vorhandensein von Alternativtexten und die Kennzeichnung als Artefakt erfolgen unabhängig voneinander.

## Feature Accessibility

Das Feature Accessibility muss in der Konfiguration eingetragen werden:

```
1    <accessibility>true</accessibility>
2    <renderers>
3      <renderer mime="application/pdf">
4        <pdf-ua-mode>PDF/UA-1</pdf-ua-mode>
5      </renderer>
6    </renderers>
```

### Anmerkungen

Einige im Programmcode bereitgestellten Bestandteile (wie insbesondere Alternativtexte) werden erst mit dieser Einstellung bei der Transformation in das PDF-Dokument übertragen. Darüber hinaus werden mit dieser Einstellung bei der Transformation konkrete Fehlermeldungen erzeugt, die zur weiteren Optimierung genutzt werden können.

## Dokumentstruktur

Apache-FOP sorgt mit eingeschaltetem Feature Accessibility automatisch für das Tagging der einzelnen Elemente. Es ist möglich, manuell einzugreifen und das Tagging bewusst zu steuern. Zu diesem Zweck lässt sich das role-Attribut einsetzen, um dem jeweils zugehörigen Element eine passende Rolle zuzuweisen. Manuelles Nachbessern könnte insbesondere bei einem komplexen Design zwingend erforderlich sein.

## Lesezeichen

Lesezeichen lassen sich mittels fo:bookmark-tree-Element einfügen (siehe Dokumentation zur Umsetzung von Lesezeichen mit Apache-FOP).

## Kopf- und Fußzeilen

Kopf- sowie Fußzeilen lassen sich mithilfe von fo:static-content-Elementen umsetzen. Wenn es sich dabei um Design- bzw. „Schmuckelemente“ ohne relevante Inhalte handelt, sollten diese Elemente als Artefakt gekennzeichnet werden (siehe Präsentationsfolien zur Vorstellung des Praxisbeispiels MAK Collection bei der 111. BiblioCon im barrierefreien PDF-Format, 2023, S.7).

Dazu werden die betroffenen fo:static-content-Elemente folgendermaßen mittels role-Attribut ausgezeichnet:

```
1  <fo:static-content flow-name="header" role="artifact">
```

### Anmerkungen

Die Kennzeichnung als Artefakt (Englisch: „artifact“) wird insbesondere bei fo:static-content- sowie fo:wrapper-Elementen übertragen. Dabei ist an dieser Stelle die Kleinschreibung entscheidend – insbesondere im Unterschied zur Schreibweise mit einem Großbuchstaben in allen anderen Fällen.

## Erzeugung von PDF-Dokumenten mit Apache-FOP

Die Transformation kann z. B. mithilfe des Oxygen XML Editor oder auch als Job in der GitLab Pipeline ausgeführt werden.

Ein möglicher Aufruf im Linux-Terminal (unter Verwendung geeigneter Pfadangaben) könnte folgendermaßen aussehen, wobei die Datei „in.xml“ alle benötigten XSL-FO-Auszeichnungen enthält und die Datei „out.pdf“ die Ausgabedatei im Ausgabeformat PDF darstellt:

```
1  docker run --user $(id -u):$(id -g) -v $(pwd):/src -w /src -it --rm chrwahl/fop -c fop_cfg.xml in/in.xml -pdf
out/out.pdf
```

Bei der Datei „fop_cfg.xml“ handelt es sich um die entsprechend eingerichtete Konfigurationsdatei.

Der Programmierung eines automatisierten Workflows – ausgehend von einem Ausgangs-XML-Dokument, das ggf. aus einer Word-Datei extrahiert wird, über ein XML-Dokument, das alle benötigten XSL-FO-Auszeichnungen enthält, bis hin zum Ausgabeformat PDF – sind dabei keine Grenzen gesetzt.

## Schriften und Font-Embedding

Bei aktiviertem Feature Accessibility werden die Schriften gemäß der Angaben in der Konfigurationsdatei bei der Transformation automatisch in das PDF-Dokument eingebettet.

## Prüfung der Barrierefreiheit

Das PDF-Dokument sollte bereits während des Erzeugungsprozesses sowie abschließend mit dem PDF Accessibility Checker (PAC) getestet werden. axes4 bietet eine webbasierte Version von PAC, die entsprechend nicht nur unter Windows nutzbar ist.

Da die Nutzbarkeit mit einem starken Vergrößerungsfaktor für Menschen mit Sehbeeinträchtigung entscheidend ist, sollte bei der Prüfung eines PDF-Dokuments außerdem testweise ein Vergrößerungsfaktor von z. B. 300% eingestellt werden. Insbesondere im Zusammenhang mit der Nutzung des PDF/A-Standards (zusätzlich zu PDF/UA) ließ sich diesbezüglich eine fatale Auswirkung auf den Auslösebereich bei internen Verlinkungen beobachten: ein Schrumpfen auf die obere linke Ecke (siehe Präsentationsfolien zur Vorstellung des Praxisbeispiels MAK Collection bei der 111. BiblioCon im barrierefreien PDF-Format, 2023, S.7).

Einstellung der Nutzung des PDF/A-Standards zusätzlich zu PDF/UA in der FOP-Konfiguration:

```
1  <accessibility>true</accessibility>
2  <renderers>
3    <renderer mime="application/pdf">
4      <pdf-ua-mode>PDF/UA-1</pdf-ua-mode>
5      <pdf-a-mode>PDF/A-1a</pdf-a-mode>
6    </renderer>
7  </renderers>
```

Um sich die unterschiedlichen Stärken einzelner Tools zunutze zu machen, ist es darüber hinaus generell empfehlenswert, mehrere Tools zur Begutachtung hinzuzuziehen. Vergleiche dazu auch den Abschnitt Hinweise zu Tools zur Überprüfung der Barrierefreiheit.

### Anmerkungen

Die besten Ergebnisse hinsichtlich des PDF/A-Standards konnten ohne weitere Deklaration der Version erzielt werden. Informationen zu den bislang in Apache-FOP implementierten Versionen liefert die Dokumentation unter https://xmlgraphics.apache.org/fop/2.11/configuration.html.

## Spezialfälle

## Aufzählungszeichen

Ggf. sollen Aufzählungszeichen in Listen für Screenreader-Programme versteckt werden, um unnötige Vorlesezeit für “Aufzählungszeichen gefülltes Kästchen” zu vermeiden. Das gelingt mithilfe des fo:wrapper-Elements:

```
1      <fo:list-item-label start-indent="2mm" end-indent="label-end()" role="NonStruct">
2          <fo:block role="NonStruct"><fo:wrapper role="artifact">&#x2022;</fo:wrapper></fo:block>
3      </fo:list-item-label>
```

## Ligaturen

Mit einer Ligatur für z. B. die Kombination aus dem Großbuchstaben T mit einem direkt darauf in Kleinschreibung folgenden Buchstaben h soll ein gefälliges Erscheinungsbild im Schriftsatz sichergestellt werden. Der senkrechte Strich im kleingeschriebenen Buchstaben h soll dabei insbesondere nicht über den Querbalken des Großbuchstaben T hinausragen. Ziel ist eine Verbesserung der Lesbarkeit des visuellen Erscheinungsbildes.

Die Zeichencodierung von Ligaturen ist jedoch in vielen Schriftarten oftmals nicht eindeutig. Dies hat zur Folge, dass die betroffene Ligatur von einem Screenreader nicht korrekt ausgelesen werden kann: Die in dieser Ligatur zusammengefassten Buchstaben werden in einem solchen Fall übersprungen und das Wort wird unverständlich.

Visuell lassen sich die Fehlerquellen in einer geeigneten Screenreader-Vorschau (z. B. callas) aufspüren. Oder aber indem Sie den Text aus dem in Adobe Acrobat geöffneten PDF-Dokument herauskopiert und in einen Windows-Editor einfügen. In der Screenreader-Vorschau von callas und auch im Windows-Editor erscheinen Zeichen mit uneindeutiger Codierung in Form eines „Kästchens“. Im Prüftool PAC führt das Vorkommen von Zeichen, deren Codierung sich nicht eindeutig zuordnen lässt, (derzeit) nicht zu einem Fehler.

### Anmerkungen

Einen möglichen Lösungsansatz bietet in diesem Fall der Einsatz des geschützten Leerzeichens ohne eigene Breite () als Bindehemmer. Unter Verwendung der 14 Basis Fonts von Adobe konnte diese Problematik bislang nicht beobachtet werden. Achtung: Mit jeder neuen Version für Apache-FOP können weitere Ligaturen hinzukommen. Dabei ist dann ggf. auch nur ein Buchstabe der Ligatur nicht auslesbar.

Für grundlegende Abhilfe sorgt die folgende Einstellung in der Konfiguration:

```
1  <complex-scripts disabled="true"/>
```

## Silbentrennung

Bei der Transformation mit Apache-FOP kann eine automatische Silbentrennung verwendet werden. Alternativ könnten auch manuell bedingte Trennzeichen () eingefügt werden.

Bei der Kombination einer Schriftart, die nicht den 14 Basis Fonts entspricht, mit (automatischer bzw. manueller) Silbentrennung und dem Feature Accessibility konnte folgendes Phänomen beobachtet werden: Beim Herauskopieren von Text aus dem in Adobe Acrobat geöffneten PDF-Dokument in einen Windows-Editor kam es bei allen Zeilen, die mit einer Silbentrennung enden, zu zahlreichen Doppelungen. Beim Herauskopieren in ein Microsoft-Programm wie Word sowie beim Herauskopieren aus dem in einem Browser geöffneten PDF-Dokument in einen Windows-Editor kam es nicht zu Doppelungen.

### Anmerkungen

Einen möglichen Lösungsansatz bietet in diesem Fall der Einsatz eines speziellen Schutzes für das bedingte Trennzeichen gemeinsam mit den beiden umgebenden Buchstaben innerhalb eines separaten inline-Elements.

Für grundlegende Abhilfe sorgt auch in diesem Fall die folgende Einstellung in der Konfiguration:

```
1 <complex-scripts disabled="true"/>
```

### Verwendung von fo:inline-container-Elementen

Ein fo:inline-container-Element wird nicht in Form einer separaten Ebene in den Strukturbaum übernommen. Es scheint vielmehr ein Feature zu sein, dass fo:inline-container-Elemente durch einen leeren Bereich voneinander abgesetzt werden.

## Einzelne Strukturelemente

### Annotation-Tag

Um ein Annotation-Element zu erzeugen, genügt die folgende Auszeichnung:

role="Annot"

Ein Annotation-Element benötigt zwingend einen Alternativtext. Apache-FOP überträgt die Angaben für das fox:alt-text-Attribut (derzeit) nicht in einem fo:block-Element und auch nicht in fo:wrapper-Elementen.

### Anmerkungen

Bei Verwendung von Apache-FOP Version 2.7 bietet der Einsatz eines fo:basic-link-Elements mit einem nicht-existierenden Ziel einen möglichen Lösungsansatz. Bei der Transformation wird Ihnen in diesem Fall eine entsprechende Warnung dazu angezeigt, dass eine Verlinkung ins Leere weist. Nachdem Sie sich vergewissert haben, dass die Warnungen auf genau die Stellen hinweisen, bei denen Sie sich bewusst für diesen Lösungsansatz entschieden haben, um einen Alternativtext übertragen zu können, können Sie diese Warnung in Kauf nehmen. Warnungen, die sich auf andere Stellen beziehen, sollten sorgsam geprüft und korrigiert werden. Im folgenden Beispiel wird das Annotation-Element genutzt, um eine formatierte Zahlenangabe mit Maßangabe mit einem geeigneten Alternativtext zu versehen.

```
1 <fo:basic-link role="Annot" internal-destination="Null" fox:alt-text="1234 Quadratmeter">1 234 m²</fo:
basic-link>
```

Ohne Textalternative wird die formatierte Zahlenangabe „1 234 m²“ von einem Screenreader ggf. folgendermaßen vorgelesen: eins zweihundertvierunddreißig m zwei. Diese Ausgabe ist insbesondere dann zu erwarten, wenn die hochgestellte Zwei durch Formatierung realisiert wurde. Alternativ könnte dafür das spezifische Zeichen aus dem Unicode-Zeichensatz verwendet werden, der dann von einem Screenreader ggf. als „hochgestellte Zwei“ ausgelesen wird.

### Umsetzung im Screenreader

Mit VoiceOver wird bei einem Annotation-Element das gelesen, was im PDF-Dokument sichtbar ist. NVDA liest bei einem Annotation-Element den Alternativtext.

### Darstellung in der Screenreader-Vorschau von PAC

In der Screenreader-Vorschau von PAC erscheint ein Annotation-Element innerhalb eines regulären P-Tags wie ein Span-Inline-Element. Angezeigt wird der im PDF-Dokument dargestellte Text. Der zugehörige Alternativtext ist nicht ersichtlich.

## Formula-Tag

Um ein Formula-Element zu erzeugen, genügt die folgende Auszeichnung:

role="Formula"

Ein Formula-Element benötigt ebenfalls zwingend einen Alternativtext.

### Anmerkungen

Ein möglicher Lösungsansatz wurde für das Annotation-Element beschrieben.

### Umsetzung im Screenreader

Mit VoiceOver wird ein Formula-Element als Bild gekennzeichnet, gelesen wird der Alternativtext. NVDA liest den Alternativtext zu einem Formula-Element ggf. nur nach manueller Bearbeitung beispielsweise mit Adobe Acrobat Pro vor. Dafür muss insbesondere ein Eintrag im Feld „Originaltext“ vorgenommen werden. Die Übertragung eines Eintrags für Originaltext bei der Transformation mit Apache-FOP ist bislang nicht gelungen. Getestet wurde u. a. das fox:actual-Attribut:

fox:actual="Originaltext"

### Darstellung in der Screenreader-Vorschau von PAC

Die Screenreader-Vorschau von PAC zeigt ein unbearbeitetes Formula-Element als (leeres) Bild mit zugehörigem Alternativtext. Der im PDF-Dokument dargestellte Text ist nicht ersichtlich. Ein bearbeitetes Formula-Element erscheint als reguläres P-Tag ohne weitere Kennzeichnung. Angezeigt wird der zugehörige Originaltext. Weder der zugehörige Alternativtext, noch der im PDF-Dokument dargestellte Text sind ersichtlich.

## Note-Tag

Ein fo:footnote-Element wird von Apache-FOP automatisch als Note-Element getaggt.

Ein Note-Element benötigt zwingend eine id. Die Übertragung eines entsprechenden Eintrags für ein fo:footnote-Element bei der Transformation mit Apache-FOP ist bislang nicht gelungen.

### Anmerkungen

Ein möglicher Lösungsansatz, um die Funktionalität von Fußnoten dennoch nutzen zu können, besteht darin, das fo:footnote-Element zu kaschieren, und zwar beispielsweise folgendermaßen:

```
1  <fo:footnote role="Span">
```

Eine interne Verlinkung mit dem Alternativtext „intern zu Fußnote“ sowie der zugehörigen Fußnotenziffer soll weitere Orientierungshilfe bieten (siehe dazu auch den Hinweis im direkt folgenden Abschnitt).

### Umsetzung im Screenreader

NVDA liest den Fußnotentext bei einer mit Apache-FOP erzeugten Fußnote direkt im Anschluss an die zugehörige Fußnotenziffer vor. Es ist daher dringend ratsam, Fußnoten nur dann innerhalb eines Satzes zu platzieren, wenn der Einschub den Kontext nicht zerreißt. Dass dieser Abschnitt im PDF-Dokument als Fußnote umgesetzt ist, wird lediglich über einen entsprechenden Hinweis bei einer zugehörigen internen Verlinkung gekennzeichnet.

## Verweise

### Interne Links

- Hinweise zu Tools zur Überprüfung der Barrierefreiheit

### Externe Links in diesem Artikel

- Apache Formatting Objects Processors (FOP) (The Apache Software Foundation)
- The Apache Software Foundation (The Apache Software Foundation)
- Dokumentation zur XSL-FO-Syntax von Apache-FOP (data2type GmbH)
- Dokumentation zum Feature Accessibility von Apache-FOP auf Englisch (The Apache Software Foundation)
- Dokumentation zur Umsetzung von Lesezeichen mit Apache-FOP auf Englisch (data2type GmbH)
- Präsentationsfolien zur Vorstellung des Praxisbeispiels MAK Collection bei der 111. BiblioCon im barrierefreien PDF-Format (Anja Ziemer)
- Oxygen XML Editor (SyncRO Soft SRL)
- PDF Accessibility Checker (PAC) (axes4)
- webbasierte Version des PDF Accessibility Checkers (PAC) (PAC, gewartet von axes4)
- Dokumentation zur Konfiguration von Apache-FOP auf Englisch (The Apache Software Foundation)
- callas pdfToolbox Desktop (callas software GmbH)
- Dokumentation zu den 14 Basis Fonts bei Adobe (Oracle)
- Screenreader NVDA (NV Access)

### Weiterführende Links

- Praxisbeispiel MAK Collection - Tagungsbeitrag zur automatisierten Erstellung von Journal-Ausgaben bei ZB MED im barrierefreien PDF-Format (Anja Ziemer, o-bib. Das offene Bibliotheksjournal)

# PDF mit jsPDF-UA

Online betrachten

## Allgemeine Einordnung

Bei jsPDF handelt es sich um eine freie JavaScript-Bibliothek zur Erzeugung von PDF-Dokumenten. Sie läuft sowohl in modernen Browsern als auch in Node.js und unterstützt die gängigen Modulformate (UMD, ES-Module, CommonJS). jsPDF wird von MrRio und der yWorks GmbH als Open-Source-Projekt gepflegt und kommt typischerweise dort zum Einsatz, wo PDF-Dokumente direkt im Client – also ohne serverseitigen Zwischenschritt – erzeugt werden sollen. Typische Anwendungsfälle sind dynamisch generierte Berichte, Formulare, Rechnungen oder Export-Funktionen in Web-Anwendungen.

Der hier beschriebene Entwicklungsstrang jsPDF-UA ist kein Bestandteil des offiziellen jsPDF-Projekts, sondern ein eigenständiger Fork auf Basis von jsPDF. Er ergänzt die Bibliothek um eine Implementierung des Standards PDF/UA-1 (ISO 14289-1). Damit lassen sich programmatisch Strukturbaum, Alternativtexte, ausgezeichnete Sprache, eingebettete Schriften, Artefakt-Markierungen sowie zahlreiche weitere Merkmale erzeugen, die für die Barrierefreiheit eines PDF-Dokuments erforderlich sind.

Solange der Pull Request mit den Änderungen des Forks nicht in das jsPDF-Kernprojekt integriert ist, müssen Anwenderinnen und Anwender, die PDF/UA-Erzeugung mit jsPDF nutzen möchten, direkt den Fork verwenden:

- Fork-Repository (jsPDF-UA): https://github.com/heikofolkerts/jsPDF-UA
- Upstream-Repository (jsPDF): https://github.com/parallax/jsPDF

Für jsPDF-UA ist die Installation direkt aus dem Fork-Repository erforderlich (z. B. per npm install github: heikofolkerts/jsPDF-UA).

Aktualisierungen des jsPDF-Kernprojekts werden kontinuierlich in den Entwicklungsstrang jsPDF-UA integriert.

## Anwendungsfälle

Der Einsatz von jsPDF-UA ist insbesondere dann empfehlenswert, wenn

- PDF-Dokumente programmatisch und reproduzierbar aus strukturierten Daten (z. B. JSON, Datenbank-Auszügen oder XML) erzeugt werden sollen,
- die Erzeugung direkt im Browser erfolgen soll, ohne dass personenbezogene Daten den Client verlassen müssen, oder
- eine bestehende jsPDF-Codebasis um Barrierefreiheit erweitert werden soll.

Im Unterschied insbesondere zu Apache-FOP wird die Dokumentstruktur bei jsPDF-UA nicht deklarativ aus einer Markup-Datei abgeleitet, sondern programmatisch mittels API-Aufrufen erzeugt. Das bedeutet: Sie bestimmen explizit, welches Strukturelement gerade geöffnet ist und welcher Textaufruf dazugehört. Dies erfordert Sorgfalt, bietet aber zugleich maximale Kontrolle über den Strukturbaum.

Eine vollständige Referenz der API finden Sie in der Projektdokumentation des Forks unter docs/pdfua/.

Ein vollständig ausgestattetes Beispieldokument steht mit showcase-pdfua-complete.js zur Verfügung und dient auch als Grundlage der folgenden Code-Auszüge.

Das damit erzeugte PDF-Dokument lässt sich unter showcase-pdfua-complete.pdf abrufen.

## Hinweise zur Erstellung von barrierefreien PDF-Dokumenten mit jsPDF-UA

### Dokumenttitel

Die Methode setDocumentTitle schreibt den Titel zusammen mit den erforderlichen XMP-Kennungen für PDF/UA-1:

```
1 doc.setDocumentTitle("PDF/UA Complete Feature Showcase - jsPDF Implementation");
```

#### Anmerkungen

setDocumentTitle aktiviert implizit den Viewer-Eintrag DisplayDocTitle mit dem Wert „true“. Dadurch zeigt der PDF-Betrachter den vergebenen Titel statt des Dateinamens an.

Die alternativ ebenfalls verfügbare Methode setProperties({ title: "..." }) setzt lediglich den Info-Dictionary-Eintrag. Für einen PDF/UA-konformen Titel ist stets setDocumentTitle zu verwenden.

### Sprache (global)

Die primäre Dokumentsprache wird über die Methode setLanguage auf dem „Catalog“ gesetzt und zusätzlich in die XMP-Metadaten übernommen. Erwartet wird ein BCP-47-Sprachcode:

```
1 doc.setLanguage("en-US"); // alternativ z. B. "de-DE" oder "de-AT"
```

#### Anmerkungen

jsPDF-UA schreibt den Sprachcode zusätzlich in jeden Marked-Content-Operator (/Lang-Eintrag innerhalb der BDC-Property-List). Dies ist notwendig, damit sowohl Adobe Acrobat Reader als auch Screenreader wie NVDA die Sprachauszeichnung zuverlässig auswerten. Firefox ist an dieser Stelle toleranter, deckt aber nicht alle Hilfsmittel ab.

### Sprache (einzelne Teilbereiche)

Für Passagen in abweichender Sprache innerhalb eines Absatzes steht ein Span-Element mit eigenem lang-Attribut zur Verfügung (siehe auch unten unter Span-Tag).

### Alternativtexte

Alternativtexte werden in jsPDF-UA über das alt-Attribut der jeweiligen Strukturelement-Methoden übergeben. Zwingend erforderlich sind sie insbesondere für Figure-, Formula- und Annot-Elemente.

#### Anmerkungen

Der Alternativtext für Links wird über die Methode textWithLink übergeben (siehe auch unten unter Link-Tag).

## Aktivierung des PDF/UA-Modus

Der PDF/UA-Modus wird bereits beim Anlegen der Instanz aktiviert. Mit dieser einen Option werden intern zahlreiche Voreinstellungen vorgenommen, die für einen gültigen PDF/UA-Output zwingend erforderlich sind – insbesondere MarkInfo/Marked true, ViewerPreferences/DisplayDocTitle true sowie der Aufbau des Strukturbaums mit StructTreeRoot:

```
1  const { jsPDF } = require("jspdf");
2  const doc = new jsPDF({ pdfUA: true });
```

### Anmerkungen

Ohne die Option pdfUA: true werden keine Strukturbaum- und Marked-Content-Operatoren erzeugt; auch die automatische Einbettung der barrierearmen Schrift Atkinson Hyperlegible unterbleibt (siehe auch unten unter Schriften und Font-Embedding). Die Aktivierung des Modus sollte daher am Anfang stehen, noch bevor Inhalte hinzugefügt werden.

## Dokumentstruktur

Die Strukturelemente des PDF-Dokuments werden mit beginStructureElement geöffnet und mit endStructureElement geschlossen. Zwischen diesen Aufrufen werden die Textausgaben (doc.text(...)) automatisch mit passenden BDC/EMC-Operatoren und eindeutigen Marked-Content-IDs (MCIDs) versehen. Das wird in jsPDF-UA im Abschnitt Marked Content (Auto) als automatisches Tagging bezeichnet.

```
1  doc.beginStructureElement("Document");
2
3  doc.beginStructureElement("H1");
4  doc.setFontSize(22);
5  doc.setFont(undefined, "bold");
6  doc.text("PDF/UA Complete Feature Showcase", 20, 25);
7  doc.endStructureElement(); // H1
8
9  doc.beginStructureElement("P");
10  doc.setFontSize(11);
11  doc.setFont(undefined, "normal");
12  doc.text(
13    "This document demonstrates all PDF/UA accessibility features " +
14      "implemented in jsPDF.",
15    20,
16    52
17  );
18  doc.endStructureElement(); // P
19
20  doc.endStructureElement(); // Document
```

### Anmerkungen

Die oberste Ebene des Strukturbaums wird stets mit „Document“ geöffnet. Erst dann folgen Überschriften (H1 bis H6), Absätze (P) und alle weiteren Tags.

beginStructureElement und endStructureElement müssen paarweise geschachtelt sein. Wird ein Element nicht geschlossen, entsteht ein defekter Strukturbaum, der von Prüftools beanstandet wird.

Die Textposition (x, y) hat keinen Einfluss auf die logische Lesereihenfolge. Maßgeblich für Screenreader ist die Reihenfolge der Aufrufe – sie sollte der gewünschten Lesereihenfolge entsprechen, unabhängig vom visuellen Layout.

## Lesezeichen

Lesezeichen (Outlines) werden über die „outline“-API erzeugt. Dafür steht die Methode bm zur Verfügung, die auf Elemente mit id verweist (“sec-1”, etc.) und zusätzlich die zugehörige Seitenangabe benötigt:

```
1  bm(null, "1. Text Elements", "sec-1", 1);
2  const listsBookmark = bm(null, "2. Lists", "sec-2", 2);
3  bm(listsBookmark, "2.1 Unordered List", "sec-2-1", 2);
4  bm(listsBookmark, "2.2 Ordered List", "sec-2-2", 2);
```

## Anmerkungen

Der erste Parameter ist das Eltern-Lesezeichen oder „null“ für eine oberste Ebene. Die Füllelemente (Punkte) zwischen Überschrift und Seitenzahl sind als dekorative Bestandteile gekennzeichnet. Falls in der Screenreader-Ansicht zwischen Überschrift und Seitenzahl ein – auch im PDF-Dokument sichtbares – Leerzeichen stehen soll, kann das Leerzeichen wunschgemäß eingefügt werden:

```
1  bm(null, "1. Text Elements ", "sec-1", 1);
```

## Inhaltsverzeichnis

Für ein semantisch korrektes, maschinenlesbares Inhaltsverzeichnis stellt jsPDF-UA die Methoden beginTOC, addTOCEntry und endTOC bereit. Intern entstehen dabei die Strukturelemente TOC und TOCI (Table Of Contents Item), jeweils mit Sprung-Link auf die Zielseite:

```
1  doc.beginTOC();
2  const tocItems = [
3    { title: "1. Text Elements", page: 1, level: 1, id: "sec-1" },
4    { title: "2. Lists", page: 2, level: 1, id: "sec-2" },
5    { title: "2.1 Unordered List", page: 2, level: 2, id: "sec-2-1" },
6    // ...
7  ];
8  let tocY = 88;
9  doc.setFontSize(8);
10  doc.setFont(undefined, "normal");
11  tocItems.forEach(item => {
12    doc.addTOCEntry({
13      title: item.title,
14      page: item.page,
15      targetId: item.id,
16      y: tocY,
17      level: item.level,
18      indent: 25,
```

```
19      subIndent: 5,
20      rightMargin: 190
21    });
22    tocY += 5;
23  });
24  doc.endTOC();
```

## Kopf- und Fußzeilen

Kopf- und Fußzeilen sowie rein dekorative Elemente dürfen nicht im Strukturbaum erscheinen. jsPDF-UA stellt für solche Inhalte das Methodenpaar beginArtifact/endArtifact bereit:

```
1  doc.beginArtifact({ type: "Pagination", subtype: "Header" });
2  doc.setFontSize(8);
3  doc.setTextColor(89, 89, 89);
4  doc.text("PDF/UA Complete Showcase", 20, 10);
5  doc.text("Page 1", 180, 10);
6  doc.setTextColor(0, 0, 0);
7  doc.endArtifact();
```

### Anmerkungen

type kann die Werte „Pagination“ (Kopf-/Fußzeilen), „Layout“ (Trennlinien, Hintergründe) oder „Page“ (Seitenzahlen, Hintergrund-Bilder) annehmen. „Pagination“ nutzt zusätzlich subtype: "Header" oder subtype: "Footer".

## Erzeugung von PDF-Dokumenten mit jsPDF-UA

Die Erzeugung des eigentlichen PDF-Dokuments erfolgt wahlweise im Browser oder in einer Node.js-Umgebung. Im Browser wird typischerweise doc.save("out.pdf") aufgerufen, das über den FileSaver einen Download auslöst. Unter Node.js wird das PDF als ArrayBuffer ausgegeben und mit fs.writeFileSync gespeichert:

```
1  const fs = require("fs");
2  const arrayBuffer = doc.output("arraybuffer");
3  fs.writeFileSync("out.pdf", Buffer.from(arrayBuffer));
```

### Anmerkungen

Der Node-Build von jsPDF bindet keinen FileSaver ein; doc.save ist dort nicht verfügbar.

Die Node-basierte Erzeugung eignet sich insbesondere für CI/CD-Pipelines, in denen PDF-Dokumente gemeinsam mit veraPDF automatisiert geprüft werden sollen.

## Schriften und Font-Embedding

Bei aktivem PDF/UA-Modus bindet jsPDF-UA die Schriftfamilie Atkinson Hyperlegible (Regular, Bold, Italic und BoldItalic) automatisch ein und verwendet sie als Standardschrift. Diese Schrift ist gezielt für Menschen mit Sehbeeinträchtigung entworfen worden und deckt den lateinischen Zeichenvorrat inklusive deutscher Umlaute ab.

Sollen eigene TrueType-Schriften verwendet werden, lassen sich diese manuell einbetten:

```
doc.addFileToVFS("MyFont.ttf", base64EncodedFont);
doc.addFont("MyFont.ttf", "MyFont", "normal");
doc.setFont("MyFont");
```

### Anmerkungen

jsPDF-UA blendet die 14 PDF-Standardschriften (Helvetica, Times usw.) im aktiven PDF/UA-Modus aus und ersetzt Aufrufe wie setFont("Helvetica") durch die Atkinson-Variante.

## Prüfung der Barrierefreiheit

Für die Prüfung der Barrierefreiheit haben sich in jsPDF-UA die folgenden zwei Werkzeuge bewährt:

- PDF Accessibility Checker (PAC) prüft stärker auf semantische Korrektheit und ist besonders streng bei Platzierungs-Attributen (/Placement) und der korrekten Verwendung von Inline- vs. Block-Elementen.
- veraPDF liefert die technische Prüfung gegen ISO 14289-1 und ist automatisierbar.

Der Aufruf von veraPDF über Docker:

```
docker run --rm -v "$(pwd)/examples/temp:/data" verapdf/cli \ --flavour ua1 /data/pdfua-complete-showcase.pdf
```

Ergänzend empfiehlt sich der Test mit Adobe Acrobat Reader in Kombination mit einem Screenreader (NVDA unter Windows, VoiceOver unter macOS). Firefox ist für Screenreader-Tests ungeeignet, da sein PDF-Renderer Strukturfehler häufig verdeckt. Vergleiche dazu auch den Abschnitt Hinweise zu Tools zur Überprüfung der Barrierefreiheit.

### Anmerkungen

Das Beispieldokument des Projekts (showcase-pdfua-complete.js) besteht die veraPDF-Prüfung mit dem Flavour „ua1“ ohne Befund. In PAC sind für einzelne Strukturelemente die Platzierungs-Attribute zu beachten, die jsPDF-UA über die jeweiligen begin*-Methoden als Parameter entgegennimmt (siehe auch unten unter Platzierungs-Attribute).

# Spezialfälle

## Automatisches Tagging vs. manueller Eingriff

Solange Textausgaben (doc.text(...)) innerhalb eines geöffneten Strukturelements erfolgen, werden Marked-Content-IDs und BDC/EMC-Operatoren vollautomatisch erzeugt. Ein manuelles Setzen von MCIDs ist nicht erforderlich und auch nicht vorgesehen.

Soll ein Text ausdrücklich kein Strukturelement erhalten (z. B. weil er dekorativ ist), ist die Ausgabe in das Methodenpaar beginArtifact/endArtifact zu klammern.

## Compat- vs. Advanced-API

jsPDF kennt zwei Betriebsmodi: den kompatiblen („compat“) und den erweiterten („advanced“). Der Advanced-Modus stellt u. a. Transformationsmatrizen, Muster (Patterns) und Form-Objects zur Verfügung. Für die Erzeugung von PDF/UA-Inhalten genügt der Standardmodus „compat“. Wird „advanced“ benötigt, sollten strukturrelevante Aufrufe gezielt in doc.compatAPI(...)-Blöcken erfolgen, damit die Marked-Content-Logik erwartungsgemäß greift.

## Platzierungs-Attribute

Das Prüftool PAC verlangt für bestimmte Elemente die explizite Angabe des Platzierungs-Attributs /Placement /Block, sobald sie als Block-Element – also nicht innerhalb eines P-Elements – verwendet werden. Die jsPDF-UA-API setzt dafür sinnvolle Voreinstellungen:

| Element | Voreinstellung | Methode | Bemerkung |
|---|---|---|---|
| Link | Inline | beginCode | bei allein stehenden Links Block-Platzierung angeben |
| Code | Inline | beginCode | bei Bedarf Block-Platzierung angeben |
| Formula | Inline | beginFormula | bei Bedarf Block-Platzierung angeben |
| Annot | Block | beginAnnot | bei Bedarf Inline-Platzierung angeben |
| Note | Block | addFootnote | bei Bedarf Inline-Platzierung angeben |
| Form | Block | beginFormField | bei Bedarf Inline-Platzierung angeben |
| BibEntry | Block | beginBibEntry | bei Bedarf Inline-Platzierung angeben |
| Figure | Block | beginFigure | bei Bedarf Inline-Platzierung angeben |

## Einzelne Strukturelemente

## Überschriften und Abschnitte

jsPDF-UA unterstützt die Überschriftenebenen H1 bis H6. Zusätzlich existieren gruppierende Elemente wie Sect, Art, Div und Part, mit denen semantische Abschnitte gekennzeichnet werden:

```
doc.beginSect();
doc.beginStructureElement("H2");
doc.markLinkTarget("sec-1");
doc.setFontSize(14);
doc.setFont(undefined, "bold");
doc.text("1. Text Elements", 20, 200);
doc.endStructureElement();

doc.beginStructureElement("P");
 doc.setFontSize(11);
 doc.setFont(undefined, "normal");
 doc.text("This section demonstrates various text formatting options:", 20, 213);
 doc.endStructureElement();
```

```
14 // ...
15 doc.endSect();
```

### Anmerkungen

beginSect und beginArt eignen sich zum logischen Gruppieren. Art ist semantisch für eigenständige, publizierbare Abschnitte gedacht (z. B. Zeitschriftenartikel), Sect für beliebige Abschnitte wie Kapitel oder nummerierte Bereiche.

## Listen

Listen werden aus den Elementen L (Liste), LI (Listenpunkt), Lbl (Aufzählungszeichen bzw. Nummerierung) und LBody (eigentlicher Inhalt) aufgebaut:

```
1  doc.beginStructureElement("L");
2  const items = ["First item", "Second item", "Third item"];
3  let y = 50;
4  items.forEach(item => {
5    doc.beginStructureElement("LI");
6    doc.beginStructureElement("Lbl");
7    doc.text("•", 25, y);
8    doc.endStructureElement(); // Lbl
9    doc.beginStructureElement("LBody");
10    doc.text(item, 32, y);
11    doc.endStructureElement(); // LBody
12    doc.endStructureElement(); //LI
13    y += 10;
14  });
15  doc.endStructureElement();
```

### Anmerkungen

Bei nummerierten Listen wird in Lbl die Nummerierung ausgegeben (z. B. „1.“, „2.“), in LBody der Text.

## Tabellen

Tabellen werden mit Table, THead, TBody, TR, TH und TD aufgebaut. Für jede Kopfzelle muss ein scope-Attribut gesetzt werden, damit die Zuordnung zu Spalten- bzw. Zeilen-Daten für Screenreader erkennbar ist:

```
1  doc.beginStructureElement("Table");
2
3  doc.beginTableHead();
4  doc.beginStructureElement("TR");
5  doc.beginStructureElement("TH", { scope: "Column" });
6  doc.setFont(undefined, "bold");
7  doc.text("Product", 25, 175);
8  doc.endStructureElement(); // TH
9  doc.beginStructureElement("TH", { scope: "Column" });
```

```
10 doc.text("Price", 80, 175);
11 doc.endStructureElement(); // TH
12 // weitere Spaltenköpfe ...
13 doc.endStructureElement(); // TR
14 doc.endTableHead();
15
16 doc.beginTableBody();
17 doc.beginStructureElement("TR");
18 doc.beginStructureElement("TH", { scope: "Row" });
19 doc.setFont(undefined, "bold");
20 doc.text("Widget A", 25, 188);
21 doc.endStructureElement(); // TH
22 doc.setFont(undefined, "normal");
23 doc.beginStructureElement("TD");
24 doc.text("$10.00", 80, 188);
25 doc.endStructureElement(); // TD
26 // weitere Datenzellen ...
27 doc.endStructureElement(); // TR
28 doc.endTableBody();
29
30 doc.endStructureElement(); // Table
```

### Anmerkungen

scope kennt die Werte „Column“, „Row“ und „Both“. Bei Tabellen mit gleichzeitigem Zeilen- und Spaltenkopf (z. B. Pivot-Tabellen) ist „Both“ zu verwenden.

Die erste Spalte eines Datensatzes kann bei Bedarf als TH mit scope: "Row" ausgezeichnet werden, um Bezeichnungen wie Produktnamen oder Kalenderwochen als Zeilenkopf zu markieren (so im Beispiel oben).

Tabellen-Trennlinien werden in jsPDF-UA derzeit nicht automatisch als Artefakt gekennzeichnet. Falls Sie dekorative Linien zeichnen, klammern Sie diese im Methodenpaar beginArtifact({ type: "Layout" })/ endArtifact() ein, damit sie nicht im Strukturbaum auftauchen.

## Figure-Tag

Bilddateien werden als Figure-Element getaggt. Sie benötigen zwingend einen Alternativtext. Die jsPDF-Methode addImage fügt die einzubettende Datei ein:

```
1 doc.beginFigure({
2   alt:
3     "A bar chart showing quarterly sales data with Q1 at 25%, Q2 at 30%, Q3 at 20%, and Q4 at 25%",
4   bbox: [57, 338, 219, 185] // x, y (from bottom), width, height
5 });
6 doc.addImage({
7   ...
8 });
9 doc.endFigure();
```

### Anmerkungen

Das bbox-Attribut (Bounding Box) in PDF-Koordinaten ist für das Figure-Element optional, wird vom Prüftool PAC jedoch empfohlen, da es die Grafik in alternativen Darstellungsformen verortbar macht.

Für rein dekorative Grafiken soll kein Figure-Element verwendet werden – dafür ist die Kennzeichnung als Artefakt vorgesehen (siehe auch unten unter Kopf- und Fußzeilen). Vergleiche dazu auch den Abschnitt Hinweise zur Erstellung von Alternativtexten.

## Link-Tag

Links werden im Strukturbaum mit dem Link-Element ausgezeichnet und technisch über die jsPDF-Methode textWithLink umgesetzt. Gemäß PDF/UA ist ein Link inline innerhalb eines P-Elements zu platzieren:

```
1  doc.beginStructureElement("P");
2  doc.setFontSize(11);
3  doc.setFont(undefined, "normal");
4  doc.text("External link: ", 20, 253);
5  doc.beginLink();
6  doc.setTextColor(0, 0, 255);
7  doc.textWithLink("jsPDF on GitHub", 52, 253, { url: "https://github.com/parallax/jsPDF" });
8  doc.setTextColor(0, 0, 0);
9  doc.endLink();
10  doc.endStructureElement();
```

Interne Links auf eine andere Seite werden über die gleiche Methode umgesetzt:

```
1  doc.beginStructureElement("P");
2  doc.text("Internal link: ", 20, 265);
3  doc.beginLink();
4  doc.setTextColor(0, 0, 255);
5  doc.textWithLink("Jump to Bibliography (Page 6)", 51, 265, { targetId: "sec-10"});
6  doc.setTextColor(0, 0, 0);
7  doc.endLink();
8  doc.endStructureElement();
```

### Anmerkungen

Für einen alleinstehenden (Block-artigen) Link – also außerhalb eines Fließtextabsatzes – erwartet das Prüftool PAC eine Platzierungs-Angabe /Placement /Block. Diese wird mit beginLink({ placement: "Block" }) gesetzt.

## Abbreviation-Tag (Abkürzungen)

Abkürzungen werden mit beginAbbreviation(expansion) ausgezeichnet. Die ausgeschriebene Form wird als Attribut E (Expansion) in die Strukturbaum-Eigenschaften übernommen:

```
1  doc.beginStructureElement("P");
2  doc.text("The ", 20, 250);
```

```
3  doc.beginAbbreviation("World Wide Web");
4  doc.text("WWW", 32, 250);
5  doc.endAbbreviation();
6  doc.text(" and ", 46, 250);
7  doc.beginAbbreviation("Hypertext Markup Language");
8  doc.text("HTML", 58, 250);
9  doc.endAbbreviation();
10  doc.text(" are fundamental web technologies.", 73, 250);
11  doc.endStructureElement();
```

### Umsetzung im Screenreader

NVDA liest anstelle der Kurzform die ausgeschriebene Variante, sofern im Screenreader die Abkürzungserkennung aktiviert ist.

## Span-Tag

Für Inline-Container (z. B. zur Auszeichnung abweichender Sprache) steht das Methodenpaar beginSpan/endSpan zur Verfügung:

```
1  doc.beginStructureElement("P");
2  doc.text("English text with ", 20, 262);
3  doc.beginSpan({ lang: "de-DE" });
4  doc.text("deutscher Text eingebettet", 61, 262);
5  doc.endSpan();
6  doc.text(" and back to English.", 130, 262);
7  doc.endStructureElement();
```

### Anmerkungen

Der Sprachcode wird sowohl am Span-Strukturelement als auch in den zugehörigen BDC-Properties (/Lang-Eintrag) gesetzt. Ohne diese doppelte Auszeichnung wechseln Acrobat Reader und NVDA die Sprache nicht zuverlässig.

## Quote- und BlockQuote-Tag

Kurze, in den Fließtext eingebettete Zitate werden mit dem Methodenpaar beginQuote/endQuote ausgezeichnet, längere, eigenständige Zitate mit beginBlockQuote/endBlockQuote:

```
1  // Inline-Zitat
2  doc.beginStructureElement("P");
3  doc.setFontSize(11);
4  doc.setFont(undefined, "normal");
5  doc.text("Shakespeare wrote: ", 20, 145);
6  doc.beginQuote();
7  doc.setFont(undefined, "italic");
8  doc.text("'To be or not to be, that is the question.'", 63, 145);
9  doc.setFont(undefined, "normal");
10  doc.endQuote();
```

```
11 doc.endStructureElement();
12
13 // Blockzitat
14 doc.beginBlockQuote();
15 doc.beginStructureElement("P");
16 doc.setFontSize(11);
17 doc.setFont(undefined, "italic");
18 doc.text("'The only way to do great work is to love what you do.'", 30, 175);
19 doc.text("If you haven't found it yet, keep looking. Don't settle.\"", 30, 183);
20 doc.setFont(undefined, "normal");
21 doc.text("- Steve Jobs", 30, 195);
22 doc.endStructureElement();
23 doc.endBlockQuote();
```

## Code-Tag

Für Quellcode steht das Methodenpaar beginCode/endCode bereit. Für inline eingefügten Code genügt die Standardform; für größere Code-Blöcke ist die Platzierung als Block-Element vorzugeben:

```
1  // Inline-Code
2  doc.beginStructureElement("P");
3  doc.setFontSize(11);
4  doc.setFont(undefined, "normal");
5  doc.text("Inline code: ", 20, 228);
6  doc.beginCode();
7  doc.text("const x = 42;", 52, 228);
8  doc.endCode();
9  doc.endStructureElement();
10
11  // Block-Code
12  doc.beginCode({ placement: "Block" });
13  doc.text("function greet(name) {", 25, 255);
14  doc.text("  return `Hello, ${name}!`;", 25, 263);
15  doc.text("}", 25, 271);
16  doc.endCode();
```

## Formula-Tag

Mathematische Formeln werden als Formula-Element getaggt. Sie benötigen zwingend einen Alternativtext:

```
1  doc.beginStructureElement("P");
2  doc.text("Famous equation: ", 20, 274);
3  doc.beginFormula(
4   alt:
5   "E equals m times c squared, where E is energy, " +
6     "m is mass, and c is speed of light"
7  );
8  doc.text("E = mc²", 62, 274);
```

```
9  doc.endFormula();
10  doc.endStructureElement();
```

### Darstellung in der Screenreader-Vorschau von PAC

In der Screenreader-Vorschau von PAC erscheint ein Formula-Element als (leeres) Form-Tag mit zugehörigem Alternativtext.

## Annotation-Tag

Kommentar-Annotationen (gelbe Klebezettel, Freitext) werden über ein Annot-Strukturelement eingebunden. Die Annotation selbst wird mit createAnnotation erzeugt und über addAnnotationRef an die Struktur angebunden:

```
1  doc.beginAnnot({ alt: "Reviewer comment about quarterly sales figure" });
2  const annotId = doc.createAnnotation({
3    type: "text",
4    title: "Reviewer",
5    contents: "Comment: This figure shows quarterly sales distribution.",
6    bounds: { x: 90, y: 180, w: 20, h: 20 },
7    open: false
8  });
9  if (annotId) doc.addAnnotationRef(annotId);
10  doc.endAnnot();
```

Kommentar-Annotationen benötigen zwingend einen Alternativtext.

### Anmerkungen

Technisch hängt ein Annot-Strukturelement nicht am Contentstrom, sondern an der Seite. Screenreader greifen daher häufig erst nach den umgebenden Absätzen auf Annotations zu – unabhängig von deren Position im Strukturbaum.

Wenn eine bestimmte Lesereihenfolge erzwungen werden muss, kann statt einer Annot-Annotation ein Note-Element verwendet werden. Note ist eine echte Inline-Struktur mit eigenem Text, die vom Screenreader exakt an der im Strukturbaum festgelegten Stelle vorgelesen wird.

### Darstellung in der Screenreader-Vorschau von PAC

In der Screenreader-Vorschau von PAC erscheint ein Annot-Element per Default als eigenständiges Element. Angezeigt wird sowohl der sichtbare Text als auch der Alternativtext.

## Note-Tag

Für Fußnoten stellt jsPDF-UA die Hilfsmethoden addFootnoteRef (Referenz im Fließtext) und addFootnote (Fußnoten-Eintrag) zur Verfügung. Im Strukturbaum entstehen daraus das Reference- und das Note-Element. Beide werden intern über eine id miteinander verknüpft und als Link mit sichtbarer Sprungmarke abgebildet:

```
1  // Fließtext mit Fußnoten-Referenz
2  doc.beginStructureElement("P");
3  doc.setFontSize(11);
4  doc.setFont(undefined, "normal");
5  let fnX = 20;
6  doc.text("PDF/UA (Universal Accessibility) is an ISO standard", fnX, 40);
7  fnX += doc.getTextWidth("PDF/UA (Universal Accessibility) is an ISO standard");
8  const ref1X = fnX;
9  doc.addFootnoteRef("[1]", fnX, 40, { noteId: "fn1", xNote: "25", yNote: "260" });
10  fnX += doc.getTextWidth("[1]") * 0.7 + 1;
11  doc.text(" that ensures", fnX, 40);
12  doc.text("PDFs can be read by assistive technologies.", 20, 50);
13  doc.endStructureElement();
14
15  // Fußnoten-Eintrag am Seitenfuß
16  doc.setFontSize(9);
17  doc.addFootnote({
18    id: "fn1",
19    label: "[1]",
20    text: "ISO 14289-1:2014, Document management — Electronic document file format enhancement
for accessibility",
21    x: 25,
22    y: 260,
23    xRef: ref1X,
24    yRef: 40,
25    labelX: 20,
26    link: true
27  });
```

### Anmerkungen

Für eine saubere Trennung zwischen Fließtext und Fußnotenblock empfiehlt sich eine dekorative Trennlinie. Diese ist als Artefakt zu markieren (beginArtifact({ type: "Layout" })).

addFootnote setzt intern placement: "Block", weil eine allein stehende Fußnote in PAC als Block-Element erwartet wird.

Wird eine Fußnote nicht mithilfe der Convenience-API, sondern über das Methodenpaar beginNote/endNote manuell erzeugt, muss die id explizit gesetzt werden – andernfalls fehlt die Verknüpfung zwischen Referenz und Fußnotentext.

## Form-Tag (barrierefreie Formularfelder)

Interaktive Formularfelder werden nicht über die generischen beginStructureElement-Aufrufe erzeugt, sondern über spezialisierte Convenience-Methoden, die sowohl das AcroForm-Widget als auch das Form-Strukturelement anlegen:

```
1  doc.addAccessibleTextField({
2    name: "name",
```

```
3   label: "Name:", // label is visible
4   tooltip: "Enter your full name.",
5   x: 20,
6   y: 55,
7   width: 80,
8   height: 12,
9   required: true
10 });
11
12 doc.addAccessibleCheckBox({
13   name: "subscribe",
14   label: "Subscribe to newsletter.", // label is visible
15   tooltip: "Check to receive our newsletter.",
16   x: 20,
17   y: 71,
18   width: 7,
19   height: 7
20 });
21
22 doc.addAccessibleComboBox({
23   name: "country",
24   label: "Country:", // label is visible
25   tooltip: "Select your country from the list.",
26   x: 20,
27   y: 88,
28   width: 60,
29   height: 12,
30   options: ["USA", "Germany", "France", "UK", "Japan"]
31 });
```

### Anmerkungen

Der tooltip wird als TU-Eintrag (Alternativer Name) in das Widget geschrieben. In der Screenreadervorschau von PAC wird bei addAccessibleCheckBox zusätzlich ein Doppelpunkt angefügt.

Pflichtfelder sind über required: true zu kennzeichnen. Der Wert landet im /Ff-Flag des Widgets.

Vergleiche dazu auch den Abschnitt PDF-Formular.

## BibEntry-Tag

Für Bibliografie-Einträge steht das Methodenpaar beginBibEntry/endBibEntry zur Verfügung:

```
1 doc.beginBibEntry();
2 doc.text("[2] PDF Association. (2024). The Matterhorn Protocol 1.1. PDF Association.", 25, 62);
3 doc.endBibEntry();
```

## Index-Tag

Index-Einträge lassen sich mithilfe des Methodenpaars beginIndex/endIndex einbinden:

```
doc.beginIndex();
doc.setFontSize(10);
doc.setFont(undefined, "normal");
doc.beginStructureElement("P");
doc.text("Abbreviations ..... 1", 25, 140);
doc.endStructureElement();
doc.beginStructureElement("P");
doc.text("Annotations ..... 4", 25, 140);
doc.endStructureElement();
// ...
doc.endIndex();
```

## Verweise

### Interne Links

- Hinweise zu Tools zur Überprüfung der Barrierefreiheit
- Hinweise zur Erstellung von Alternativtexten
- PDF-Formular

### Externe Links in diesem Artikel

- jsPDF auf GitHub (Upstream, ohne PDF/UA-Unterstützung) (Parallax)
- PDF/UA-1 (ISO 14289-1) (PDF Association)
- jsPDF-UA auf GitHub (Fork mit PDF/UA-Implementierung) (Heiko Folkerts)
- API-Referenz der PDF/UA-Erweiterung (Heiko Folkerts)
- Vollständiges Beispieldokument (Showcase, Programmcode) (Heiko Folkerts und Anja Ziemer)
- Vollständiges Beispieldokument (Showcase, PDF-Dokument) (Heiko Folkerts und Anja Ziemer)
- Screenreader NVDA (NV Access)
- Atkinson Hyperlegible Font (Braille Institute of America)
- PDF Accessibility Checker (PAC) (axes4)
- veraPDF (Open Preservation Foundation)

### Weiterführende Links

- Dokumentation zur PDF/UA-Implementierung im Fork (Heiko Folkerts)
- Verfügbare Strukturelemente (Heiko Folkerts)
- Bekannte Einschränkungen (Heiko Folkerts)
- PDF/UA in a Nutshell (PDF Association)
- Matterhorn Protocol 1.1 (PDF Association)
- ISO 14289-1:2014 (International Organization for Standardization)
- BITi Prüfschritte (BITi-Wiki)
- Testvorgehen für PDF-Dokumente gemäß BITV 2.0 und BFSG (BIT inklusiv)

# Entscheidungshilfe zu PDF Konvertern - Software zum PDF-Export

Online betrachten

## Allgemeine Einordnung

Für die Analyse können folgende Ressourcen zugrunde gelegt werden:

- PDF/UA-Standard: Der derzeit anwendbare PDF/UA-Standard ist PDF/UA 1 (ISO 14289-1). Dieser Standard ist in den Testfällen zugrunde gelegt. Der bereits seit März 2024 veröffentlichte PDF/UA-2-Standard wird noch von keiner Software unterstützt. Daher ist er noch nicht anwendbar.
- Matterhorn Protokoll: Im Matterhorn Protokoll werden detaillierte Fehlerbedingungen genannt. Eine Unterscheidung zwischen maschinenprüfbaren Kriterien und vom Menschen zu prüfenden Kriterien gemacht. In diesem Artikel werden nur die vom Menschen zu prüfenden Kriterien explizit angeführt. Automatisiert prüfbare Kriterien, wie z. B. die bounding boxes bei Bildern, sollten mit dem PDF Accessibility Checker geprüft werden.
- Aktueller PDF Accessibility Checker (PAC): Der PAC ist Prüfwerkzeug für alle maschinenprüfbaren Kriterien und dient zur Unterstützung bei der Prüfung der vom Menschen zu prüfenden Kriterien. Eine Alternative ist die Open Source Anwendung veraPDF, die auch in einer Version für MacOS zur Verfügung steht.

## Testfälle

Im Folgenden finden Sie Testfälle für verschiedene Anforderungen der Barrierefreiheit von PDF-Dokumenten. Mit diesen Testfällen kann festgestellt werden, wie gut ein PDF Konverter arbeitet. Es geht vorwiegend um Testfälle, die vom Menschen zu prüfen sind. Tests, die automatisiert erfolgen, werden mit „Maschine“ gekennzeichnet. Tests, die vom Menschen zu prüfen sind, werden mit „Mensch“ gekennzeichnet.

## Metadaten

- Wird der Dokumenttitel korrekt gesetzt, wenn er im Quelldokument hinterlegt ist? (Maschine)
- Wird die Dokumentsprache korrekt gesetzt? (Maschine prüft, ob die Dokumentsprache vorhanden ist, Mensch prüft die Richtigkeit der Dokumentsprache)
- Werden abweichende Textstellen mit der korrekten Sprache im PDF hinterlegt, falls dies im Quelldokument vorbereitet wurde? (Mensch)
- Wird der PDF/UA Identifier in die Metadaten des PDF eingefügt? (Maschine)

## Lesereihenfolge

- Ist die Lesereihenfolge im Tag-Baum des PDFs korrekt? (Mensch)
- Gibt es eine Möglichkeit, die Lesereihenfolge festzulegen, falls diese nicht automatisch von der Quellsoftware festgelegt wird? (Mensch)

## Überschriften

- Werden alle in der Quellsoftware als Überschrift ausgezeichneten Elemente auch als Überschriften in das PDF-Dokument übernommen? (Mensch)

## Absätze

- Werden Absätze auch über einen Seitenwechsel hinweg als zusammengehörig getaggt? (Mensch)
- Werden leere Absätze, Seitenumbrüche oder Abschnittswechsel als Artefakt gekennzeichnet? (Mensch)

## Verzeichnisse

- Werden Verzeichnisse (Inhaltsverzeichnis, Abbildungsverzeichnis, Tabellenverzeichnis) als TOC-Element und die einzelnen Verzeichniseinträge als TOCI-Element getaggt und zu den zugehörigen Überschriften im PDF verlinkt? (Mensch)
- Wird die eingestellte Zoom-Stufe bei der Aktivierung der Verlinkung beibehalten? (Mensch)
- Wird ein Verzeichnis über mehr als eine Seite im PDF in der Tag-Struktur als ein zusammengehöriges Verzeichnis (z. B. TOC) ausgezeichnet? (Mensch)

## Fußnoten

- Sind die Fußnoten im Text klar als Fußnoten erkennbar (Reference-Tag)? (Mensch)
- Ist der Fußnoten-Text jeweils in einem eigenen Note-Tag getaggt? (Mensch)
- Kann der jeweils zugehörige Fußnoten-Text mit der Fußnote in Zusammenhang gebracht werden (eventuell durch die Lesereihenfolge oder durch gegenseitige Verlinkung)? (Mensch)

## Listen

- Werden Listen oder Aufzählungen korrekt in einem L-Tag umgesetzt? (Mensch)
- Wird eine zusammengehörige Liste in einem L-Tag getaggt, auch wenn sie sich über mehrere Seiten erstreckt? (Mensch)
- Ist jeder Listeneintrag korrekt in einem LI-Tag getaggt? (Mensch)
- Befindet sich ein Aufzählungszeichen oder die Nummerierung der Liste in einem Lbl-Tag? (Mensch)
- Befindet sich der Listentext in einem LBody-Tag? (Mensch)
- Werden geschachtelte Listen (Listen mit mehreren Ebenen) korrekt im Tag-Baum umgesetzt? (Mensch)

## Kopf- und Fußzeilen

Werden Kopf- und Fußzeilen als Artefakt gekennzeichnet? Ist der Type des Artefaktes „Pagination“ und der Subtype „Header“ oder „Footer“? (Mensch)

## Bilder und Diagramme

- Kann zwischen dekorativen Bildern / Grafiken und Bildern, die Information transportieren, unterschieden werden? (Mensch)
- Werden im Quelldokument als dekorativ markierte Bilder im PDF als Artefakt gekennzeichnet? (Mensch)
- Werden Bilder, die Information transportieren, korrekt als Figure getaggt und in der richtigen Reihenfolge im Tag-Baum eingefügt? (Mensch)
- Erhalten die Figure-Tags die korrekten Angaben für die Begrenzungsrahmen Bounding Box? (Maschine prüft, ob der Begrenzungsrahmen vorhanden ist, Mensch muss prüfen, ob der Begrenzungsrahmen die korrekte Größe und Position hat)

- Ist der Alternativtext im PDF vorhanden, wenn dieser im Quelldokument hinterlegt wurde? (Maschine prüft das Vorhandensein des Alternativtextes, Mensch prüft die inhaltliche Korrektheit)
- Werden korrekt erstellte (Bild-)Beschriftungen als Caption getaggt? (Mensch)

## Tabellen

- Wird die Tabelle korrekt als Table-Element mit der korrekten Unterstruktur getaggt? (Mensch)
- Wird auch bei verbundenen Zellen eine reguläre Tabelle erstellt? (Maschine)
- Werden alle Überschriftenzellen als TH (TableHeader) getaggt? (Mensch)

### Einfache Tabellen

- Gibt es eine Möglichkeit, alle Überschriftenzellen mit einem Geltungsbereich (auch Scope oder Zellenumfang) zu versehen? (Mensch)
- Wird dieser Geltungsbereich korrekt in das PDF übernommen? (Maschine prüft, sobald es TH-Tags gibt, ob eine Zuordnung zu Unterzellen da ist; Mensch prüft die Korrektheit der Zuordnung)

### Komplexe Tabellen

- Gibt es die Möglichkeit, eine Verbindung zwischen Überschriftenzellen und den dazugehörigen Unterzellen herzustellen? (Mensch)
- Wird diese Verbindung mittels IDs in das PDF übernommen? (Maschine prüft, sobald es TH-Tags gibt, ob eine Zuordnung zu Unterzellen da ist; Mensch prüft die Korrektheit der Zuordnung)

## Formularfelder

- Gibt es die Möglichkeit, in der Quellsoftware erstellte Formularfelder für das barrierefreie PDF zu bearbeiten und diese barrierefrei zu konvertieren? (Mensch)

## Konfigurierbarkeit

- Gibt es die Möglichkeit, Sonderelemente wie Formeln, Blockzitate oder Definitionslisten für Glossare bzw. Abkürzungsverzeichnisse zu konfigurieren? (Mensch)
- Wenn Tabellen zum Erreichen einer bestimmten Darstellung (als Layout-Tabellen) verwendet werden: Gibt es die Möglichkeit, diese als Layout-Tabellen zu markieren? Werden diese dann nicht mit der Tabellenstruktur in das PDF-Dokument konvertiert, sondern nur der Inhalt in der korrekten Lesereihenfolge? (Mensch)
- Werden außerhalb der angebotenen Funktion in der Quellsoftware, Möglichkeit angeboten, Elemente als Artefakt zu kennzeichnen? (Mensch)
- Gibt es einen Mechanismus, bei dem die Kopfzeile und/oder die Fußzeile so gekennzeichnet werden kann, sodass deren Inhalt einmal im Tag Baum vorkommt? (Mensch)

## Barrierefreiheit des User Interface

- Gibt es eine Aussage über die Barrierefreiheit der Anwendung zum Erstellen des PDF-Dokuments? D. h. ist die Nutzung des PDF Konverters selbst barrierefrei möglich?
- Gibt es eine Selbsterklärung zur Einhaltung der Barrierefreiheitsanforderungen für Desktop-Anwendungen oder ein entsprechendes Barrierefreiheitsgutachten?

# Verweise

## Interne Links – Testdokumente zur Prüfung

Hier finden Sie Dokumente, die Sie zur Prüfung von Konvertern verwenden können.

- Word – Download DOCX Datei
- Word – Download PDF Datei
- PowerPoint – Download PPTX Datei
- PowerPoint – Download PDF Datei

## Externe Links in diesem Artikel

- Prüfkriterien für PDF/UA-Konformität (Matterhorn-Protocoll in der englischen Version von 2021) (PDF Association)
- PDF Accessibility Checker (PAC) (PAC, gewartet von axes4)
- veraPDF (veraPDF consortium)

## Weiterführende Links

- Prüfkriterien für PDF/UA-Konformität (Matterhorn-Protokoll in der deutschen Version vom 23.06.2016 im Auftrag von BIT inklusiv) (PDF Association)

# Adobe PDF Accessibility Auto-Tag API

Online betrachten

## Allgemeine Einordnung

Adobe bietet mit der Auto-Tag API eine Dienstleistung für Entwickelnde an und stellt die Funktion auch bei Nutzung von Adobe Acrobat Pro zur Verfügung. Sie können über Adobes Service nicht-barrierefreie PDF-Dokumente taggen lassen. Der Service versucht mit KI-Unterstützung, ein möglichst gutes Ergebnis bezüglich Barrierefreiheit zu erzielen.

Allerdings können nicht alle Inhalte automatisch barrierefrei gemacht werden – das Ergebnis muss noch von Hand nachbearbeitet werden. Unter anderem müssen Alternativtexte nachgetragen werden, Formularfelder oder Anmerkungen werden nicht unterstützt und die Lesereihenfolge sowie semantische Tags (z. B. Inhaltsverzeichnisse, Fußnoten) müssen überprüft werden. Fehler in der konzeptionellen oder grafischen Gestaltung eines PDF-Dokuments können beim nachträglichen Tagging nicht behoben werden. D. h. es kann per se nur eine technische Zugänglichkeit erreicht werden.

Laut einer wissenschaftlichen Studie (Tran, 2023) erzeugt die Auto-Tag API von sich aus keine barrierefreien PDF-Dokumente, die nach PDF/UA-Standard barrierefrei sind. Aber für nicht getaggte oder überhaupt nicht barrierefreie PDF-Dokumente kann sie ein guter Startpunkt sein.

Die Ergebnisse der Studie in Kürze:

- 37% der im Matterhorn-Protokoll (PDF Association, 2021) aufgeführten Fehler wurden von der Auto-Tag API entdeckt und behoben, 42% wurden nicht behoben, der Rest konnte nicht geprüft werden.
- Werden nur die Fehler betrachtet, die nicht automatisch geprüft werden können, wurden immerhin 46% der Matterhorn-Fehler behoben.

Die Auto-Tag API gibt es online als Demo unter https://acrobatservices.adobe.com/dc-accessibility-playground/main.html. Es können nur Dokumente mit maximal 10 MB hochgeladen werden. Eine Überprüfung mit mehreren Dokumenten hat Folgendes ergeben: Bei kurzen, strukturell einfachen und vorwiegend textbasierten Dokumenten wird ein guter Tag-Baum erzeugt. Eine händische Überprüfung ist aber auf jeden Fall notwendig. Grafisch komplexe Dokumente erfordern eine umfassende Nacharbeit und werden teilweise nicht oder nicht vollständig bearbeitet. Sie erhalten die Rückmeldung, dass das Dokument zu komplex zur Bearbeitung ist, sowie einen Report in Form einer Excel-Datei.

Neben der Lösung von Adobe gibt es u. a. vergleichbare Services auf Basis der Open Source Docling IBM und Amazon Textract OCR. Die Ergebnisse unterscheiden sich im Detail, in jedem Fall ist eine Nachbearbeitung der Dokumente erforderlich. Mit Hilfe von spezialisierten Werkzeugen wie dem PDFix SDK ist eine automatisierte Korrektur von Standardproblemen möglich.

## Verweise

### Interne Links

Keine vorhanden

### Externe Links in diesem Artikel

- Matterhorn-Protokoll (PDF Association, 2021)

- Auto-Tag API Demo (Adobe)
- Docling (Docling)
- Amazon Textract OCR (Amazon)
- PDFix SDK (PDFix)
- PDFix Marketplace (PDFix)

## Weiterführende Links

- PDF Accessibility Auto-Tag API (Adobe)
- KI-unterstützte Korrektur von PDFs zur Verbesserung der Barrierefreiheit—Chancen und Herausforderungen am Beispiel von Adobe's Auto-Tag API (Hochschule der Medien Stuttgart, Marvin Tran, 2023)

# Elektronische Signatur im PDF

Online betrachten

## Anwendungsfälle

Elektronische Signaturen werden zum Signieren von digitalen Dokumenten eingesetzt.

Es gibt unterschiedliche elektronische Signaturen, die unterschiedliche Sicherheitsstufen aufweisen. Es gibt drei Sicherheitsstufen bei elektronischen Signaturen, die durch die eIDAS-Verordnung der EU definiert werden.

### Einfache elektronische Signatur (EES):

- Niedrigstes Sicherheitsniveau
- Keine gesetzlichen Formvorschriften
- Keine Identitätsprüfung oder Authentifizierung erforderlich
- Beispiele: eingescannte Unterschrift, selbst erstellte Signatur im Adobe Reader, E-Mail-Signatur

### Fortgeschrittene elektronische Signatur (FES):

- Mittleres Sicherheitsniveau
- Eindeutig dem Unterzeichnenden zugeordnet
- Ermöglicht Identifizierung des Unterzeichnenden
- Erfordert Identitätsnachweis (z. B. Einmalpasswort oder Fernüberprüfung)
- Nachträgliche Änderungen am Dokument sind erkennbar

### Qualifizierte elektronische Signatur (QES):

- Höchstes Sicherheitsniveau
- Strengste Identitätsprüfung (persönlich oder per Video)
- Zuverlässiger biometrischer Abgleich mit Multifaktor-Authentifizierung
- Rechtlich gleichgestellt mit handschriftlicher Unterschrift
- Erfüllt gesetzliche Schriftformerfordernisse

Die Wahl der Sicherheitsstufe hängt von den rechtlichen Anforderungen und dem gewünschten Sicherheitsniveau ab. Während einfache und fortgeschrittene Signaturen für viele Alltagsanwendungen ausreichen, ist die qualifizierte elektronische Signatur für besonders sensible oder rechtlich bindende Dokumente erforderlich.

Ein PDF kann auf mehreren Wegen digital signiert werden.

## Signieren mit Adobe

Adobe Acrobat bietet die Möglichkeit der elektronischen Signatur. Hier können unterschiedliche Sicherheitsstufen der elektronischen Signatur verwendet werden.

Wird in einem barrierefreien PDF-Dokument das Formularfeld für die Signatur verwendet, bleibt das Dokument auch nach der Signierung über das Signatur-Formularfeld barrierefrei. Da das Formularfeld getaggt ist, ist auch die damit eingefügte Signatur getaggt. Hier ist es egal, ob eine einfache elektronische Signatur oder eine qualifizierte elektronische Signatur verwendet wird.

Signaturen können mit Adobe Acrobat auch ohne Formularfeld in einem PDF aufgebracht werden. Dann ist das PDF-Dokument allerdings nicht mehr vollständig barrierefrei. Die Signatur ist ein nicht-getaggtes Objekt und verursacht somit eine Fehlermeldung bezüglich der Barrierefreiheit.

## Qualifizierte elektronische Signaturen in der D-A-CH-Region

### Deutschland

Es gibt verschiedene Anbieter für qualifizierte elektronische Signaturen, die über die Bundesnetzagentur abgerufen werden können.

### Österreich

Über ID Austria (ehemals Handy-Signatur) kann eine qualifizierte elektronische Signatur auf ein PDF aufgebracht werden. Dies ist **auch barrierefrei möglich**, ohne Adobe Acrobat nutzen zu müssen.

### Schweiz

In der Schweiz benötigen Sie eine Smartcard oder ein Signaturzertifikat, das bei einem Remote Signing Service hinterlegt ist. Dazu kann eine Liste der anerkannten Anbieter von Zertifizierungsdiensten abgerufen werden.

## Verweise

### Interne Links

Keine vorhanden

### Externe Links in diesem Artikel

- eIDAS-Verordnung (EU)
- Bundesnetzagentur Deutschland (Bundesnetzagentur für Elektrizität, Gas, Telekommunikation, Post und Eisenbahnen)
- ID Austria (Bundeskanzleramt Österreich)
- Liste der anerkannten Anbieter in der Schweiz (Schweizerische Akkreditierungsstelle SAS)

### Weiterführende Links

- Allgemeine Informationen zur elektronischen Signatur (Wikipedia)
- Verschiedene Möglichkeiten der barrierefreien Signatur in Österreich (online) (A-SIT Zentrum für sichere Informationstechnologie - Austria)

# Unterschied zwischen PDF- und anderen Dokumenten für assistive Technologien

Online betrachten

## Einleitung

In diesem Kapitel werden die Unterschiede in der Nutzung von PDF-Dokumenten und anderen Dateiformaten mit assistiven Technologien dargestellt. Es geht insbesondere um die Herausforderungen, aber auch die potenziellen Vorteile von PDF-Dokumenten bei der Nutzung mit assistiven Technologien. Ziel ist es, ein Verständnis für die PDF-spezifischen Besonderheiten zu vermitteln, die wichtig sind, um diese Art von Dokumenten barrierefrei zu erstellen.

## Kurze Einführung in die Funktionsweise von assistiven Technologien

Assistive Technologien helfen Nutzenden mit Beeinträchtigungen, auf dem Bildschirm angezeigte Inhalte wahrzunehmen oder mit dem jeweiligen Anwendungsprogramm zu interagieren. Dafür ist es notwendig, dass auf die auf dem Bildschirm angezeigten Informationen zugegriffen werden kann. Diese können dann beispielsweise in Sprache umgewandelt werden (Text-To-Speech), in Brailleschrift ausgegeben oder vergrößert dargestellt werden. Dies alles kann von entsprechender Software (Screenreader, Braillezeile, Vergrößerungssoftware, Spracheingabe, etc.) geleistet werden.

## Zugriff auf Informationen aus nicht PDF-Dokumenten

Bei Word- und anderen Dokumentenformaten ist es für assistive Technologien möglich, über entsprechende Mechanismen direkt auf das Dokument mit all seinen Informationen zuzugreifen. So kann beispielsweise die Struktur eines Dokuments (Überschriften, Listen …) und natürlich auch der eigentliche Text ausgelesen werden, sofern das Ausgangsdokument korrekt erstellt wurde. Auch in das Dokument eingebettete Grafiken können erkannt werden und beispielsweise mit dem entsprechend vergebenen Alternativtext ausgegeben werden. Dies ermöglicht es Nutzenden, das Dokument direkt zu lesen. Es ist neben der assistiven Technologie kein weiteres „Hilfsmittel" notwendig, um alle Inhalte im Dokument wahrnehmen zu können.

## Zugriff auf PDF-Dokumente

PDF-Dokumente wurden ursprünglich für den Druck verwendet. Hierfür hat sich das Format als besonders praktisch erwiesen, da es eine feste Seitenstruktur hat. Das Format ist außerdem auf allen Endgeräten, für die es entsprechende Viewer oder Reader gibt, lesbar und sieht auch immer gleich aus. Spezifische Anforderungen im Hinblick auf Barrierefreiheit sind in diesem Format zunächst nicht umgesetzt. Assistive Technologien können also nicht direkt auf die Informationen innerhalb des PDF-Dokuments zugreifen. Damit auch PDF-Dokumente für Menschen mit Beeinträchtigungen zugänglich sind, wurde der sogenannte PDF/UA-Standard entwickelt. Das UA steht hier für „Universal Accessibility“, also „allgemeine Zugänglichkeit“. Dieser Standard ermöglicht es assistiven Technologien, auf die Informationen zuzugreifen, die sich in dem Dokument befinden.

## Was passiert technisch?

Damit der Inhalt eines PDF-Dokuments von assistiven Technologien ausgelesen werden kann, muss der oben genannte PDF/UA-Standard umgesetzt werden. Dafür erhält jeder Inhalt im Dokument einen sogenannten „Tag“ – also ein „Etikett“. Damit wird die Art des Inhalts erkannt: Überschrift, Text, Bild, Liste, Tabelle … Die Tags gemeinsam bilden die Tag-Struktur, den sogenannten Tag-Baum. In der Tag-Struktur ist

sowohl die Art des Inhalts, wie auch die korrekte Lesereihenfolge abgebildet. Die Tag-Struktur ist für assistive Technologien ähnlich einer HTML-Struktur, die dann verarbeitet und ausgegeben werden kann. Für Screenreader beispielsweise sieht ein PDF-Dokument aus wie eine Webseite und lässt sich auch entsprechend navigieren. Viele WCAG-Erfolgskriterien (EN 301 549 Abschnitt 10) können daher auf PDF-Dokumente angewendet werden. Es gibt beispielsweise auch im PDF/UA-Standard H-Tags, wie im HTML, die zur Darstellung von Überschriften verwendet werden. Auch die Interaktion, wie beispielsweise mit Links, funktioniert dann wie auf Webseiten. Mithilfe der Tag-Struktur können auch komplexe Strukturen, wie Tabellen oder Fußnoten, abgebildet werden. Somit kann komplexe Information, wie eine Tabelle, auch für alle Nutzenden in einem PDF-Dokument mithilfe des PDF/UA-Standards barrierefrei zur Verfügung gestellt werden. Es lässt sich also sagen, dass ein PDF-Dokument ohne Tags für assistive Technologien nicht zugänglich ist, ein korrekt getaggtes PDF-Dokument hingegen kann alle Informationen barrierefrei darstellen.

## Beispiel für einen Tag-Baum

Im Folgenden wird ein Beispiel für einen Tag-Baum dargestellt, um zu illustrieren, wie die Information aus einem Dokument umgesetzt wird.

### Tag-Baum in Listenstruktur

- <Document>
    - <H1>
    - <P>
    - <P>
    - <P>
    - <H1>
    - <P>
    - <P>

## Der PDF/UA-Standard

Wie oben beschrieben, ist der Standard die Grundlage für assistive Technologien, um ein PDF-Dokument barrierefrei darzustellen. Damit dies gelingt, müssen folgende Voraussetzungen erfüllt sein:

- Das Dokument muss den PDF/UA-Standard unterstützen.
- Der Viewer oder Reader muss den PDF/UA-Standard unterstützen.
- Die assistive Technologie muss die Information korrekt verarbeiten und darstellen.

## Was passiert im Acrobat Reader, wenn keine Tags vorhanden sind?

Wenn ein Dokument nicht getaggt ist, ist es erst einmal für assistive Technologien nicht zugänglich. Der Acrobat Reader erkennt beim Öffnen eines Dokuments, ob auf dem PC beispielsweise ein Screenreader läuft. Wenn das so ist, versucht er selbstständig, eine Tag-Struktur aufzubauen. Dafür wird das Dokument analysiert und anhand der Ergebnisse wird eine entsprechende Struktur aufgebaut. Diese Struktur ist allerdings nicht immer korrekt, und daher sind die automatisch getagten Dokumente oftmals nur begrenzt zugänglich. Aus diesem Grund ist es notwendig, dass die korrekte Tag-Struktur durch die Erstellenden des Dokuments festgelegt wird.

## Vorteile der Tag-Struktur

Dass assistive Technologien nicht direkt auf die Dokumentinformationen, sondern über die Tag-Struktur an die notwendigen Informationen kommen, kann auch ein Vorteil sein. So können beispielsweise

Informationen mithilfe der Tag-Struktur anders aufbereitet werden als diese optisch auf dem Bildschirm präsentiert werden. Für die optische Darstellung kann es insbesondere von Vorteil sein, Inhalte in einer unsichtbaren Tabelle anzuordnen. In der Tag-Struktur kann es hingegen ratsam sein, eine solche Tabelle vollständig in Form von Listen bzw. Listeneinträgen darzustellen. Mit einer solchen bewusst vorgenommenen Anpassung kann der Inhalt eines Dokuments für unterschiedliche Zielgruppen auf bestmögliche Art und Weise zugänglich und verständlich gemacht werden. Vergleiche dazu auch den Abschnitt Hinweise zur Gestaltung von Tabellen.

## Fazit

PDF-Dokumente können aus der derzeitigen Standardsoftware nur mit zusätzlichen PlugIns / Add-ins barrierefrei erstellt werden. Wenn PDF-Dokumente direkt mit der Standardsoftware konvertiert werden, sind sie in der barrierefreien Herstellung komplexer als die Ausgangsformate, da im auf diese Weise erzeugten PDF-Dokument aufwändig nachgearbeitet werden muss. PDF-Dokumente bieten durch die Navigation wie auf Webseiten eine schnelle Möglichkeit für Screenreader, gezielt zu bestimmten Informationen zu springen. In Ausnahmefällen ist es möglich, für assistive Technologien eine abweichende Dokumentenstruktur aufzubauen, die besser auf die Bedürfnisse von Menschen mit Beeinträchtigungen eingeht als es ggf. bei der visuellen Darstellung der Fall ist.

## Verweise

### Interne Links

- Hinweise zur Gestaltung von Tabellen

### Externe Links in diesem Artikel

Keine vorhanden

### Weiterführende Links

Keine vorhanden

# Hinweise zu EPUB-Dokumenten

Online betrachten

Im Folgenden sind Hinweise zu EPUB-Dokumenten zu finden.

## Inhaltsverzeichnis


- Einführung in EPUB – Das neue Standardformat für digitale Bücher (5 min)
  - Was ist EPUB?
  - Warum EPUB?
  - Aufbau eines EPUB-Dokuments
  - EPUB-Versionen und ihre Entwicklung
  - Erstellung und Überprüfung von EPUB-Dateien
  - Fazit
  - Verweise
- EPUB-Reader – Darstellung und Barrierefreiheit (10 min)
  - Vorteile und Herausforderungen
  - Wichtige EPUB-Reader
  - Was muss noch verbessert werden?
  - Fazit
  - Verweise
- EPUB 3.3 – Neuerungen, Vorteile und Herausforderungen (6 min)
  - Anwendungsfälle
  - EPUB 3.3 im Vergleich zu Vorgängerversionen
  - Verbesserungen der Barrierefreiheit in EPUB 3.3
  - CSS und JavaScript in EPUB 3.3
  - Testen der Barrierefreiheit von EPUB 3.3
  - Verweise
- Export aus Textverarbeitungsprogrammen – Möglichkeiten, Grenzen und Empfehlungen (6 min)
  - Anwendungsfälle
  - Vergleich der EPUB-Erstellungstools
  - Fazit und Empfehlung
  - Tipp zur Prüfung der Barrierefreiheit
  - Verweise

# Einführung in EPUB – Das neue Standardformat für digitale Bücher

Online betrachten

## Was ist EPUB?

EPUB (Electronic Publication) ist das führende offene Standardformat für digitale Bücher und Dokumente. Es wurde ursprünglich vom International Digital Publishing Forum (IDPF) entwickelt und wird heute vom World Wide Web Consortium (W3C) weitergeführt. EPUB bietet zahlreiche Funktionen, die es zur bevorzugten Wahl für digitale Bücher machen, insbesondere im Bereich Barrierefreiheit.

Im Gegensatz zu proprietären Formaten wie Amazons MOBI oder PDFs mit festen Seitenlayouts ist EPUB ein flexibles Format, das sich dynamisch an verschiedene Bildschirmgrößen anpasst (Reflowable Layout). Dadurch eignet es sich besonders für E-Books, wissenschaftliche Dokumente und Lernmaterialien.

## Warum EPUB?

### Vorteile gegenüber anderen Formaten

EPUB bietet im Vergleich zu anderen Formaten wie PDF einige entscheidende Vorteile:

- **Flexible Darstellung**: EPUB-Inhalte passen sich dynamisch an verschiedene Bildschirmgrößen an, was insbesondere für mobile Endgeräte vorteilhaft ist.
- **Unterstützung für Multimedia**: EPUB erlaubt die Einbettung von Videos, Audiodateien und interaktiven Elementen.
- **Strukturierte Inhalte**: EPUB basiert auf modernen Webtechnologien wie HTML, CSS und XML, wodurch eine klare Struktur und semantische Auszeichnung möglich ist.
- **Barrierefreiheit**: EPUB kann mit Screenreadern genutzt werden, unterstützt alternative Texte für Bilder und ermöglicht eine einfache Navigation durch strukturierte Inhaltsverzeichnisse.

### EPUB als barrierefreies Standardformat

Barrierefreiheit ist ein zentrales Merkmal des EPUB-Formats. Es ermöglicht Nutzern mit Beeinträchtigungen den einfachen Zugang zu digitalen Inhalten durch:

- **Strukturierte Navigation**: EPUB-Dateien enthalten ein Inhaltsverzeichnis (nav.xhtml), das eine logische und hierarchische Navigation erleichtert.
- **Alternativtexte für Bilder**: Bilder sollten mit sinnvollen alt-Attributen versehen sein, sodass Screenreader den Inhalt korrekt vorlesen können.
- **Unterstützung für assistive Technologien**: Screenreader wie NVDA, JAWS oder VoiceOver können EPUBs problemlos interpretieren und vorlesen.
- **Dynamische Anpassbarkeit**: Nutzer können Schriftgrößen, Farben, Kontraste und Zeilenabstände individuell anpassen, um die Lesbarkeit zu verbessern.

## Aufbau eines EPUB-Dokuments

Eine EPUB-Datei ist technisch gesehen eine ZIP-Datei mit einer festgelegten Verzeichnisstruktur. Sie enthält unter anderem folgende Bestandteile:

```
/EPUB/
  content.opf  (Metadaten und Manifest)
```

```
  toc.xhtml  (Inhaltsverzeichnis)
  chapter1.xhtml  (Textinhalt)
  images/
  styles/
/META-INF/
  container.xml  (Zeigt auf die Hauptdatei des Buchs)
mimetype
```

### Beispiel für eine content.opf-Datei

```
1  <package xmlns="http://www.idpf.org/2007/opf" version="3.0" unique-identifier="bookid">
2    <metadata>
3      <dc:title>Mein EPUB-Buch</dc:title>
4      <dc:language>de</dc:language>
5      <meta property="dcterms:modified">2025-03-26T12:00:00Z</meta>
6    </metadata>
7    <manifest>
8      <item id="chapter1" href="chapter1.xhtml" media-type="application/xhtml+xml"/>
9      <item id="nav" href="toc.xhtml" media-type="application/xhtml+xml" properties="nav"/>
10   </manifest>
11   <spine>
12     <itemref idref="chapter1"/>
13   </spine>
14 </package>
```

## EPUB-Versionen und ihre Entwicklung

EPUB hat sich über die Jahre stetig weiterentwickelt. Wichtige Meilensteine waren:

- **EPUB 2 (2007)**: Einführung einer grundlegenden Struktur mit XHTML 1.1 und XML.
- **EPUB 3 (2011)**: Umstellung auf HTML5, Einführung von CSS3 und Unterstützung für interaktive Inhalte.
- **EPUB 3.3 (2023)**: Verbesserte Barrierefreiheitsstandards, präzisere Definitionen und eine Vereinfachung des Standards.

## Erstellung und Überprüfung von EPUB-Dateien

### Tools zur Erstellung von EPUB-Dateien

Zur Erstellung eines EPUBs gibt es verschiedene Werkzeuge:

- **Sigil**: Open-Source-Editor zur manuellen Erstellung und Bearbeitung von EPUB-Dateien.
- **Calibre**: Umfangreiches E-Book-Management-Tool mit Konvertierungsfunktionen.
- **Pandoc**: Kommandozeilen-Tool für die Umwandlung von Markdown in EPUB.
- **Adobe InDesign**: Professionelle Layout-Software mit EPUB-Exportfunktion.

### Überprüfung der Barrierefreiheit

Nach der Erstellung sollte das EPUB auf Korrektheit und Barrierefreiheit geprüft werden:

- **EPUBCheck**: Validierung des EPUB-Formats nach W3C-Standards.

- **Ace by DAISY**: Automatisierte Prüfung der Barrierefreiheit eines EPUB-Dokuments.
- **Manuelle Tests mit Screenreadern**: Beispielsweise mit NVDA (Windows) oder VoiceOver (macOS, iOS).

### Beispiel für eine EPUB-Validierung mit epubcheck

epubcheck mein_epub.epub

## Fazit

EPUB ist ein leistungsstarkes und flexibles Format für digitale Bücher und Dokumente. Es bietet zahlreiche Funktionen für eine optimale Darstellung auf verschiedenen Geräten und ist ein wichtiger Standard für barrierefreie Inhalte.

Mit der Weiterentwicklung zu EPUB 3.3 wurden insbesondere die Anforderungen an Barrierefreiheit weiter verbessert, was EPUB zu einer zukunftssicheren Wahl für digitale Publikationen macht.

## Verweise

### Interne Links

Keine vorhanden

### Externe Links in diesem Artikel

- International Digital Publishing https://idpf.org/
- World Wide Web Consortium https://www.w3.org/
- Sigil https://sigil-ebook.com/ Open-Source-Editor zur manuellen Erstellung und Bearbeitung von EPUB-Dateien.
- Calibre https://calibre-ebook.com/ Umfangreiches E-Book-Management-Tool mit Konvertierungsfunktionen.
- Pandoc https://pandoc.org/ Kommandozeilen-Tool für die Umwandlung von Markdown in EPUB.
- Adobe InDesign https://www.adobe.com/de/products/indesign.html
- EPUBCheck https://github.com/w3c/epubcheck Validierung des EPUB-Formats nach W3C-Standards.
- Ace by DAISY https://daisy.org/tools/ace/ Automatisierte Prüfung der Barrierefreiheit eines EPUB-Dokuments.

### Weiterführende Links

- EPUB 3.3 Spezifikation (W3C)

# EPUB-Reader – Darstellung und Barrierefreiheit

Online betrachten

## Vorteile und Herausforderungen

EPUB-Reader sind Software-Anwendungen oder spezialisierte Geräte, die EPUB-Dateien, ein gängiges Format für E-Books, anzeigen. Sie bieten zahlreiche Vorteile, machen das digitale Lesen zugänglicher und flexibler, bringen jedoch auch einige Herausforderungen mit sich, besonders im Hinblick auf die Barrierefreiheit.

## Vorteile

### 1. Plattformübergreifende Nutzbarkeit

EPUB-Reader sind auf verschiedenen Betriebssystemen verfügbar, darunter Windows, macOS, Linux, iOS und Android. Diese Vielseitigkeit ermöglicht den Zugang zu E-Books auf nahezu allen gängigen Geräten und fördert so die Zugänglichkeit für unterschiedliche Nutzergruppen.

### 2. Anpassbare Darstellung

Nutzer können die Darstellung von EPUB-Inhalten nach ihren Bedürfnissen anpassen. Schriftgrößen, Farben und Kontraste lassen sich modifizieren, was besonders für Menschen mit Sehbeeinträchtigungen von Vorteil ist. Diese Flexibilität trägt zur individuellen Lesbarkeit bei und verbessert das Leseerlebnis.

### 3. Unterstützung für Screenreader

Gute EPUB-Reader unterstützen Screenreader und ermöglichen es so blinden und sehbehinderten Nutzern, die Inhalte von E-Books zu hören. Insbesondere EPUB 3 hat in diesem Bereich Vorteile, da es semantische Strukturen wie Überschriften und Listen unterstützt, die für Screenreader gut lesbar sind.

### 4. Lesezeichen und Notizen

Viele EPUB-Reader bieten Funktionen wie das Setzen von Lesezeichen, das Hinzufügen von Notizen und das Markieren von Textstellen. Diese Funktionen erleichtern die Navigation und das spätere Auffinden von wichtigen Passagen, was besonders für akademische und berufliche Nutzer hilfreich ist.

## Herausforderungen

### 1. Uneinheitliche Unterstützung von EPUB 3.3

Viele Reader unterstützen nur ältere EPUB-Versionen wie EPUB 2 oder den ursprünglichen EPUB 3-Standard. EPUB 3.3, was neue Features wie verbesserte Interaktivität, Multimedia-Inhalte und eine stärkere Barrierefreiheit verspricht, wird oft nicht vollständig oder gar nicht unterstützt. Das führt dazu, dass Nutzer, die von den erweiterten Funktionen profitieren könnten, auf einige dieser Vorteile verzichten müssen.

### 2. Mangelhafte Barrierefreiheit

Auch wenn einige EPUB-Reader grundlegende barrierefreie Funktionen bieten, sind viele von ihnen in ihrer Umsetzung noch unzureichend. Interaktive Elemente wie Formulare, eingebettete Medien oder dynamische Inhalte werden oft nicht korrekt oder überhaupt nicht barrierefrei dargestellt. Dies erschwert Nutzern mit besonderen Bedürfnissen den Zugang zu allen Funktionen und Inhalten.

### 3. Probleme mit Digital Rights Management (DRM)

E-Books, die durch DRM (Digital Rights Management) geschützt sind, können oft nur in bestimmten, von den Verlagen zugelassenen EPUB-Readern geöffnet werden. Dies schränkt die Auswahl des Readers und somit die Flexibilität der Nutzer ein und verhindert die Nutzung auf anderen Geräten oder mit alternativen Softwarelösungen.

## Wichtige EPUB-Reader

Es gibt eine Reihe von EPUB-Readern, die eine unterschiedliche Nutzererfahrung bieten. Hier sind einige der bekanntesten:

### 1. Thorium Reader (Windows, macOS, Linux)

Link zum Tool: Thorium Reader Ein Open-Source-Reader, der für seine sehr gute Barrierefreiheit bekannt ist. Er unterstützt die neueren EPUB 3-Standards und bietet eine benutzerfreundliche Oberfläche für die Navigation.

### 2. Apple Books (macOS, iOS)

Link zum Tool: Apple Books Bietet gute Unterstützung für EPUB 3 und eine benutzerfreundliche Oberfläche, jedoch sind die Barrierefreiheitsfunktionen im Vergleich zu anderen Readern eher eingeschränkt.

### 3. Google Play Books (Android, Web)

Link zum Tool: Google Play Books Unterstützt EPUB 3, jedoch ist die Barrierefreiheit nicht vollständig gegeben, und einige Nutzer berichten von Problemen mit der Bedienbarkeit, insbesondere bei interaktiven Inhalten.

### 4. Adobe Digital Editions (Windows, macOS)

Link zum Tool: Adobe Digital Editions Einer der am weitesten verbreiteten EPUB-Reader, insbesondere für DRM-geschützte Inhalte. Die Barrierefreiheit ist jedoch nicht optimal und könnte verbessert werden.

## Was muss noch verbessert werden?

### 1. Bessere Unterstützung für EPUB 3.3

Obwohl EPUB 3.3 zahlreiche Verbesserungen in Bezug auf Interaktivität und Barrierefreiheit bietet, wird dieser Standard in vielen Readern noch nicht vollständig umgesetzt. Insbesondere die Unterstützung für neue Features wie eingebettete Formulare, multimediafähige Inhalte und erweiterte Semantik für Screenreader muss verbessert werden, um eine wirklich barrierefreie Nutzung zu gewährleisten.

## 2. Barrierefreie Bedienung

Die Navigation durch EPUB-Dokumente muss weiter optimiert werden, insbesondere für Menschen mit Behinderungen. Die Implementierung von Tastenkombinationen, verbesserten Text-zu-Sprache-Funktionen und einer besseren Unterstützung für Screenreader ist notwendig, um die Zugänglichkeit auf allen Geräten und in allen Konfigurationen zu gewährleisten.

## 3. Mehr Open-Source-Alternativen

Viele der gängigen EPUB-Reader sind proprietär und bieten nur eingeschränkte Anpassungsmöglichkeiten. Die Entwicklung und Verbreitung von Open-Source-Readern, die eine größere Flexibilität in Bezug auf Anpassungen und Erweiterungen ermöglichen, könnte dazu beitragen, die Vielfalt der Nutzungsmöglichkeiten zu erhöhen und die Barrierefreiheit zu verbessern.

## 4. Verbesserung für MINT-Literatur

Besonders im Kontext von MINT (Mathematik, Informatik, Naturwissenschaften und Technik) gibt es noch erhebliche Verbesserungsmöglichkeiten bei der barrierefreien Nutzung von EPUB-Readern. Formeln, Diagramme und technische Darstellungen werden häufig nicht korrekt oder gar nicht in einer für blinde oder sehbehinderte Nutzer zugänglichen Form angezeigt.

Im aktuellen Stand können viele EPUB-Reader mit wissenschaftlicher Literatur, insbesondere mit mathematischen Formeln, nicht optimal umgehen. Die Darstellung von Formeln und Diagrammen stellt für blinde und sehbehinderte Nutzer eine enorme Hürde dar, da diese oft nur als Bilder eingebunden sind und damit nicht durch Screenreader erfasst werden können. Der **MathML-Standard** für mathematische Formeln und die Integration von **LaTeX** in EPUB-Dokumente sind in diesem Zusammenhang entscheidend, um die Inhalte auch für Menschen mit Sehbehinderungen zugänglich zu machen.

Derzeit ist es noch ein ungelöstes Problem, Formeln in einer strukturierten und maschinenlesbaren Weise darzustellen, die sowohl von Screenreadern richtig interpretiert wird als auch eine alternative Anzeigeform für sehende Nutzer bietet. Ebenso fehlen barrierefreie Methoden zur Darstellung komplexer Diagramme und Graphen. Hier müssen EPUB-Reader und die zugrunde liegende EPUB-Spezifikation noch weiter entwickelt werden, um sicherzustellen, dass auch MINT-Inhalte barrierefrei zugänglich sind.

Eine Lösung könnte in der verbesserten Unterstützung von interaktiven und dynamischen Inhalten liegen, die es ermöglichen, Formeln und Diagramme nicht nur statisch darzustellen, sondern auch interaktive und anpassbare Ansichten zu bieten, die es allen Nutzern – einschließlich denen mit Behinderungen – ermöglichen, die Inhalte effizient zu nutzen.

## 5. CSS- und JavaScript-Unterstützung für Barrierefreiheit

Ein weiteres ungelöstes Problem in vielen EPUB-Readern ist der Umgang mit **CSS** und **JavaScript**. Diese Technologien sind entscheidend für die visuelle Darstellung und interaktive Funktionalität von E-Books. Während CSS verwendet wird, um das Layout und die Darstellung von Text und Medien anzupassen, ermöglicht JavaScript interaktive Elemente, die den Lesefluss und die Benutzerinteraktion fördern können.

Leider wird CSS oft nicht in der Weise implementiert, dass die Anpassung an verschiedene Nutzerbedürfnisse – wie z. B. der Wechsel zu hohem Kontrast oder die Darstellung von Text in einer benutzerdefinierten Schriftart – nahtlos funktioniert. Insbesondere für Menschen mit visuellen

Beeinträchtigungen sind diese Anpassungen von großer Bedeutung, um die Lesbarkeit zu erhöhen. In einigen Fällen können falsch implementierte CSS-Styles oder unzureichende Unterstützung für responsive Layouts dazu führen, dass die Inhalte verzerrt oder nicht korrekt angezeigt werden.

JavaScript bietet die Möglichkeit, interaktive Elemente wie Schaltflächen, Formulare und dynamische Inhalte zu erstellen, aber in vielen EPUB-Readern wird JavaScript entweder nicht korrekt ausgeführt oder ist in barrierefreien Versionen deaktiviert, um potenzielle Sicherheitsrisiken zu vermeiden. Dies führt dazu, dass Nutzer, die auf interaktive Features angewiesen sind, wie z. B. Formeln in interaktiven Übungen oder dynamische Diagramme, diese nicht vollständig nutzen können. Eine Lösung könnte in der besseren Integration von **ARIA**-Labels und **JavaScript** für barrierefreie Interaktivität bestehen, die sowohl von Screenreadern als auch von anderen assistiven Technologien korrekt interpretiert wird.

## Fazit

EPUB-Reader bieten viele Vorteile, darunter plattformübergreifende Nutzung und Anpassbarkeit der Darstellung. Für Nutzer mit besonderen Bedürfnissen, wie blinden oder sehbehinderten Menschen, ist die barrierefreie Nutzung jedoch noch nicht immer garantiert. Es besteht weiterhin Verbesserungsbedarf, insbesondere bei der Unterstützung des neuesten EPUB 3.3-Standards, der Optimierung der Navigation für Screenreader, der Entwicklung von mehr Open-Source-Alternativen und der barrierefreien Darstellung von MINT-Literatur. Besonders in den Bereichen mathematische Formeln, Diagramme und komplexe technische Darstellungen muss noch viel passieren, um sicherzustellen, dass alle Studierenden, unabhängig von ihren visuellen Fähigkeiten, Zugang zu wissenschaftlicher Literatur haben. Ebenso sind Verbesserungen im Umgang mit CSS und JavaScript erforderlich, um eine vollständig barrierefreie und interaktive Nutzererfahrung zu gewährleisten.

Ein weiteres aktuelles Problem ist, dass viele blinde Nutzer im Studium und Beruf noch lieber EPUB-Dateien entpacken und die (x)HTML-Dateien im Browser lesen, anstatt spezialisierte EPUB-Reader zu nutzen. Dies liegt daran, dass die Reader oft nicht alle barrierefreien Funktionen bieten, die für eine effiziente und angenehme Lektüre notwendig sind. Im Browser sind die Texte durch den Einsatz von Screenreadern und Browser-Funktionen oft besser zugänglich und flexibler anpassbar. Dies zeigt, dass EPUB-Reader für blinde und sehbehinderte Nutzer noch nicht vollständig ausgereift sind und dass es notwendig ist, die Software weiter zu entwickeln, um eine gleichwertige und barrierefreie Nutzung zu gewährleisten. Nur durch kontinuierliche Verbesserungen in diesen Bereichen kann das Ziel erreicht werden, dass EPUB-Reader für alle Nutzer zugänglich sind – unabhängig von ihren individuellen Bedürfnissen.

## Verweise

### Interne Links

Keine vorhanden

### Externe Links in diesem Artikel

- Thorium Reader https://www.edrlab.org/software/thorium-reader/ (Windows, macOS, Linux)
- Apple Books https://www.apple.com/de/apple-books/ (macOS, iOS)
- Google Play Books https://play.google.com/store/books (Android, Web)
- Adobe Digital Editions https://www.adobe.com/solutions/ebook/digital-editions.html (Windows, macOS)

## Weiterführende Links

Keine vorhanden

# EPUB 3.3 – Neuerungen, Vorteile und Herausforderungen

Online betrachten

## Anwendungsfälle

EPUB ist ein weit verbreitetes Format für digitale Bücher und Dokumente, das besonders für seine Flexibilität und Barrierefreiheit geschätzt wird. Die Version 3.3 bringt zahlreiche Verbesserungen mit sich, die sowohl die Erstellung als auch die Nutzung von EPUB-Dokumenten erheblich beeinflussen. Insbesondere für Menschen mit Beeinträchtigungen wurden durch die klare Definition von Barrierefreiheitsanforderungen bedeutende Fortschritte erzielt.

## EPUB 3.3 im Vergleich zu Vorgängerversionen

EPUB 3.3 ist keine vollständige Neuentwicklung, sondern eine Verbesserung von EPUB 3.2 und 3.0. Die Kernprinzipien bleiben bestehen:

- **Verwendung von XHTML und CSS** für strukturierte Inhalte und Darstellung.
- **Unterstützung für eingebettete Multimedia-Inhalte**, wie Audio und Video.
- **Interaktivität durch JavaScript** für dynamische Elemente.
- **Zugänglichkeit durch semantische HTML5-Tags** zur besseren Unterstützung von Screenreadern und assistiven Technologien.

## Wichtige Änderungen von EPUB 3.3 gegenüber EPUB 3.2

- **Klarere Barrierefreiheitsanforderungen** durch die explizite Verknüpfung mit den WCAG 2.1.
- **Vereinfachte Metadaten-Struktur** zur besseren Kennzeichnung von barrierefreien EPUBs.
- **Entfernung veralteter Features**, die in modernen Umgebungen nicht mehr relevant sind.
- **Genaue Definition der Navigation und Inhaltsstruktur**, um eine konsistentere Nutzung über verschiedene Lesesysteme hinweg zu gewährleisten.

### Beispiel für eine typische EPUB 3.3-Struktur

Eine EPUB-Publikation besteht weiterhin aus den zentralen Dateien:

```
/EPUB/
  content.opf
  toc.xhtml
  chapter1.xhtml
  images/
  styles/
/META-INF/
  container.xml
mimetype
```

Die Datei content.opf enthält wichtige Metadaten zur EPUB-Struktur und Barrierefreiheit:

```
<metadata>
  <dc:title>Beispielbuch</dc:title>
  <dc:language>de</dc:language>
```

```
4   <meta property="schema:accessibilityFeature">alternativeText</meta>
5   <meta property="schema:accessibilityFeature">longDescription</meta>
6   <meta property="schema:accessibilityFeature">structuralNavigation</meta>
7 </metadata>
```

## Verbesserungen der Barrierefreiheit in EPUB 3.3

### Strukturelle Navigation und Inhaltsverzeichnis

EPUB 3.3 verlangt ein gut strukturiertes Navigationsdokument (nav.xhtml), das insbesondere für Screenreader-Nutzende eine bessere Orientierung bietet.

#### Beispiel für eine optimierte nav`-Struktur

```
1 <nav epub:type="toc" id="toc">
2   <h2>Inhalt</h2>
3   <ol>
4     <li><a href="chapter1.xhtml">Kapitel 1: Einführung</a></li>
5     <li><a href="chapter2.xhtml">Kapitel 2: Grundlagen</a></li>
6   </ol>
7 </nav>
```

### Verbesserte Metadaten für Barrierefreiheit

EPUB 3.3 erfordert detaillierte Metadaten zur Beschreibung der Barrierefreiheit eines Dokuments. Diese helfen Nutzenden, geeignete Inhalte zu finden, z. B. durch Angabe, ob Alternativtexte für Bilder oder Untertitel für Videos vorhanden sind.

#### Beispiel für Barrierefreiheits-Metadaten in einem EPUB 3.3

```
1 <meta property="schema:accessMode">textual</meta>
2 <meta property="schema:accessModeSufficient">textual</meta>
3 <meta property="schema:accessibilityFeature">alternativeText</meta>
4 <meta property="schema:accessibilityFeature">displayTransformability</meta>
5 <meta property="schema:accessibilityHazard">none</meta>
```

### Alternative Darstellungen für visuelle Inhalte

Bilder und Diagramme müssen in EPUB 3.3 immer mit passenden Alternativtexten oder langen Beschreibungen versehen sein, damit Screenreader-Nutzende die Inhalte verstehen können.

#### Beispiel für ein barrierefreies Bild in EPUB 3.3

```
1 <figure>
2   <img src="diagramm.png" alt="Balkendiagramm zeigt Umsatzentwicklung von 2018 bis 2023." />
3   <figcaption>Dieses Diagramm veranschaulicht die Umsatzsteigerung in den letzten fünf Jahren.</
figcaption>
4 </figure>
```

### MathML-Unterstützung für barrierefreie mathematische Inhalte

EPUB 3.3 unterstützt weiterhin MathML für mathematische Formeln, was besonders für wissenschaftliche Publikationen von Vorteil ist.

#### Beispiel für eine mathematische Formel mit MathML

```
<math xmlns="http://www.w3.org/1998/Math/MathML">
  <mrow>
    <msup><mi>a</mi><mn>2</mn></msup>
    <mo>+</mo>
    <msup><mi>b</mi><mn>2</mn></msup>
    <mo>=</mo>
    <msup><mi>c</mi><mn>2</mn></msup>
  </mrow>
</math>
```

## CSS und JavaScript in EPUB 3.3

### CSS-Skripte für barrierefreie Anpassungen

CSS (Cascading Style Sheets) ist in EPUB 3.3 ein zentrales Werkzeug, um das Layout und die Darstellung von Text und anderen Elementen zu kontrollieren. Es kann verwendet werden, um Schriftgrößen zu verändern, Kontraste anzupassen und sogar die Navigation durch den Text zu optimieren. Für die Barrierefreiheit können spezifische CSS-Techniken verwendet werden, um sicherzustellen, dass die Inhalte auch für Menschen mit visuellen Einschränkungen lesbar sind.

#### Beispiel: Anpassung der Schriftgröße über CSS

```
@media (max-width: 600px) {
  body {
    font-size: 1.2em;
  }
}
```

Dies sorgt dafür, dass der Text auf Geräten mit kleiner Bildschirmgröße eine größere Schriftgröße erhält, was die Lesbarkeit verbessert.

### JavaScript zur Unterstützung interaktiver Elemente

JavaScript wird in EPUB 3.3 häufig verwendet, um interaktive Inhalte zu erstellen, wie z. B. dynamische Formulare, Quizze oder interaktive Diagramme. Barrierefreiheit kann durch korrekt implementierte ARIA (Accessible Rich Internet Applications)-Techniken und durch die Einhaltung von WCAG-Richtlinien sichergestellt werden.

#### Beispiel: Interaktive Schaltfläche mit JavaScript

```
<button onclick="alert('Dies ist eine interaktive Schaltfläche!')">Klicken Sie hier</button>
```

Durch die Verwendung von ARIA-Attributen wie aria-live und aria-label kann diese Schaltfläche für Screenreader-Nutzer zugänglich gemacht werden.

```
<button aria-live="assertive" aria-label="Schaltfläche zur Anzeige einer Nachricht" onclick="alert('Dies ist eine interaktive Schaltfläche!')">Klicken Sie hier</button>
```

Durch diese JavaScript-Interaktivität und das Hinzufügen von ARIA-Attributen wird die Bedienbarkeit auch für Menschen mit Behinderungen verbessert.

## Testen der Barrierefreiheit von EPUB 3.3

Zur Überprüfung der Barrierefreiheit eines EPUBs empfiehlt sich:

- **EPUBCheck**: Offizielles Validierungstool zur Prüfung der EPUB-Spezifikation.
- **Ace by DAISY**: Ein Tool zur Barrierefreiheitsprüfung von EPUB-Dokumenten.
- **Manuelle Tests mit Screenreadern**: Beispielsweise mit NVDA, VoiceOver oder JAWS.
- **Praktische Nutzungstests mit betroffenen Personen**, um realistische Nutzungsbedingungen zu simulieren.

### Ein Beispiel für den Test mit EPUBCheck

```
epubcheck mein_epub.epub
```

## Verweise

### Interne Links

Keine vorhanden

### Externe Links in diesem Artikel

- EPUBCheck (W3C): Offizielles Validierungstool zur Prüfung der EPUB-Spezifikation.
- Ace by DAISY (DAISY-Konsortium): Tool zur automatisierten Barrierefreiheitsprüfung von EPUB-Dokumenten.
- NVDA Screenreader (NV Access): Kostenloser Screenreader für Windows zur manuellen Testung der Barrierefreiheit.
- VoiceOver (Apple): In macOS und iOS integrierter Screenreader.
- JAWS Screenreader (Freedom Scientific): Kommerzieller Screenreader für Windows.

### Weiterführende Links

- EPUB 3.3 Spezifikation (W3C): Offizielle Definition der aktuellen EPUB-Version.
- Inclusive Publishing (DAISY-Konsortium): Ressourcen rund um barrierefreies Publizieren.
- EPUBTest (Benetech): Testplattform zur Kompatibilität von EPUB mit verschiedenen Lese-Apps und Geräten.

# Export aus Textverarbeitungsprogrammen – Möglichkeiten, Grenzen und Empfehlungen

Online betrachten

## Anwendungsfälle

Barrierefreie EPUB-Dateien ermöglichen Menschen, u. a. mit Sehbehinderung, einen flexiblen, zugänglichen Zugang zu Literatur – sei es auf Mobilgeräten, mittels Screenreadern, Vergrößerungssoftware oder Braillezeilen. Dieser Artikel gibt einen Überblick, wie sich mit gängigen Textverarbeitungsprogrammen wie Microsoft Word, LibreOffice Writer, Apple Pages sowie mit Tools wie Calibre (Calibre-Projekt) und Sigil (Sigil-Projekt) EPUB-Dateien erzeugen lassen.

## 1. Microsoft Word mit WordToEPUB

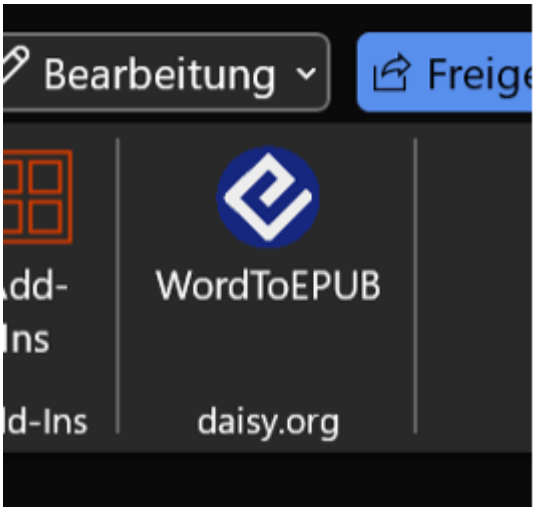
Link zum Tool: WordToEPUB (DAISY-Konsortium)

### Vorgehen

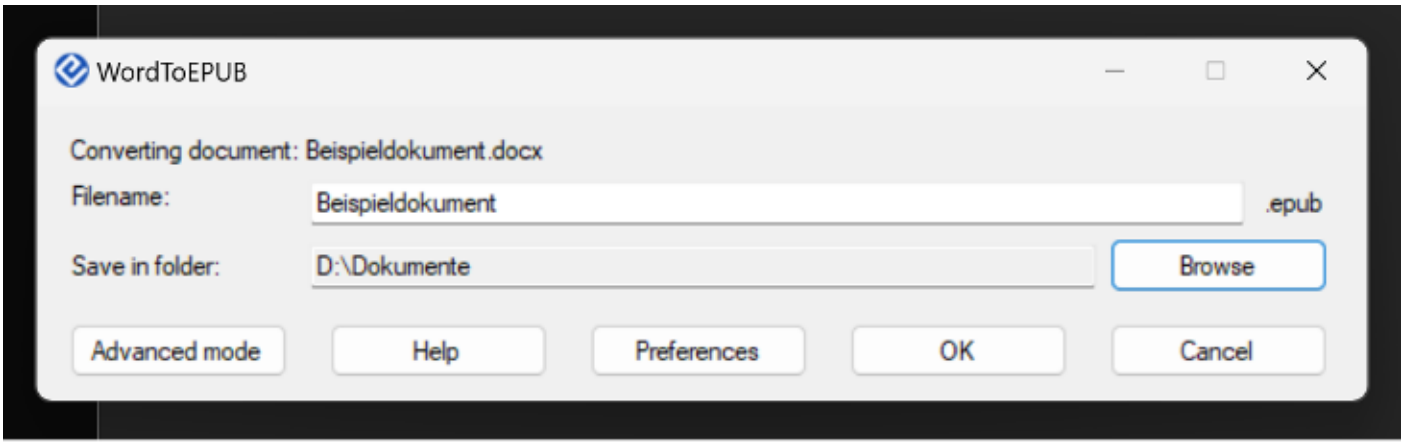
Installieren Sie das kostenlose Tool WordToEPUB.

Öffnen Sie Ihr Word-Dokument (.docx).

Klicken Sie in der Symbolleiste auf „WordToEPUB“.




Ein Assistent führt Sie durch den Exportprozess.




Die erzeugte Datei entspricht dem Standard EPUB 3.0 und kann automatisch validiert werden.

### Vorteile

- Exportiert standardkonforme EPUB 3.0-Dateien.
- Alternativtexte, semantische Strukturen und Sprachwechsel werden berücksichtigt.
- Integration eines barrierefreien Darstellungs-Overlays (z. B. Schriftgrößenanpassung).
- Automatische Validierung mit EPUBCheck und Bericht über Barrierefreiheit.
- Benutzerfreundlich, gute Anleitungen.

### Nachteile

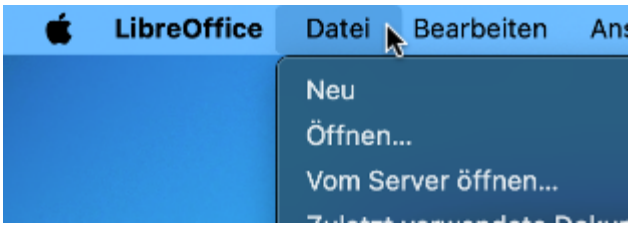
- Formeln (z. B. mit dem Word-Formeleditor) werden als Bilder exportiert, nicht als MathML.
- Metadatenbearbeitung im Export eingeschränkt – Nachbearbeitung mit Tools wie Sigil empfohlen.
- Keine integrierte Vorschau der erzeugten EPUB-Datei.

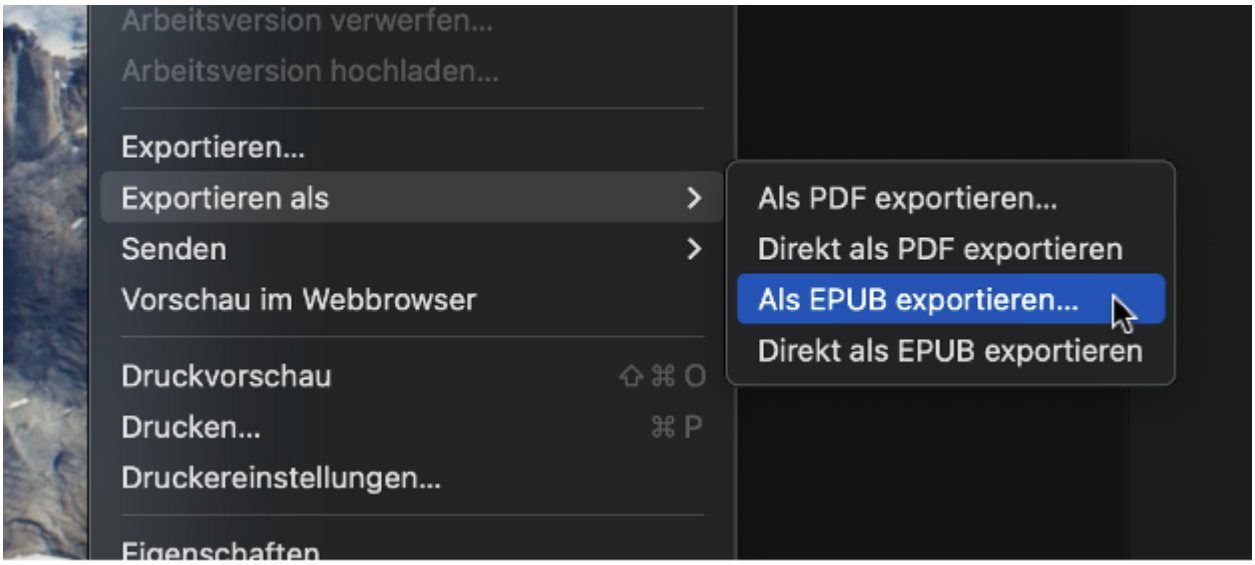
## 2. LibreOffice Writer

### Vorgehen

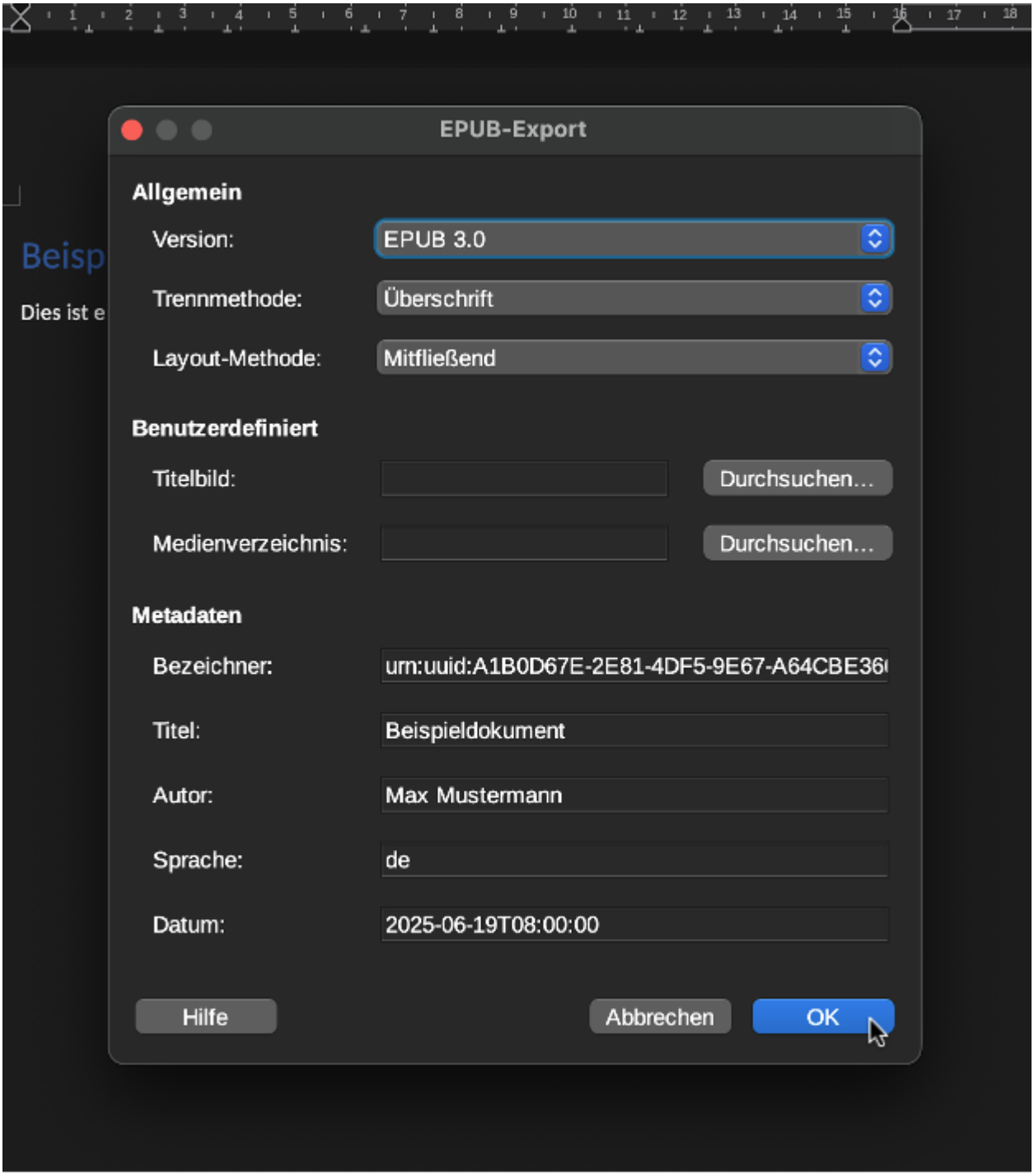
Öffnen Sie das Dokument in LibreOffice Writer.

Wählen Sie *Datei* > *Exportieren als* > *Als EPUB exportieren ...*

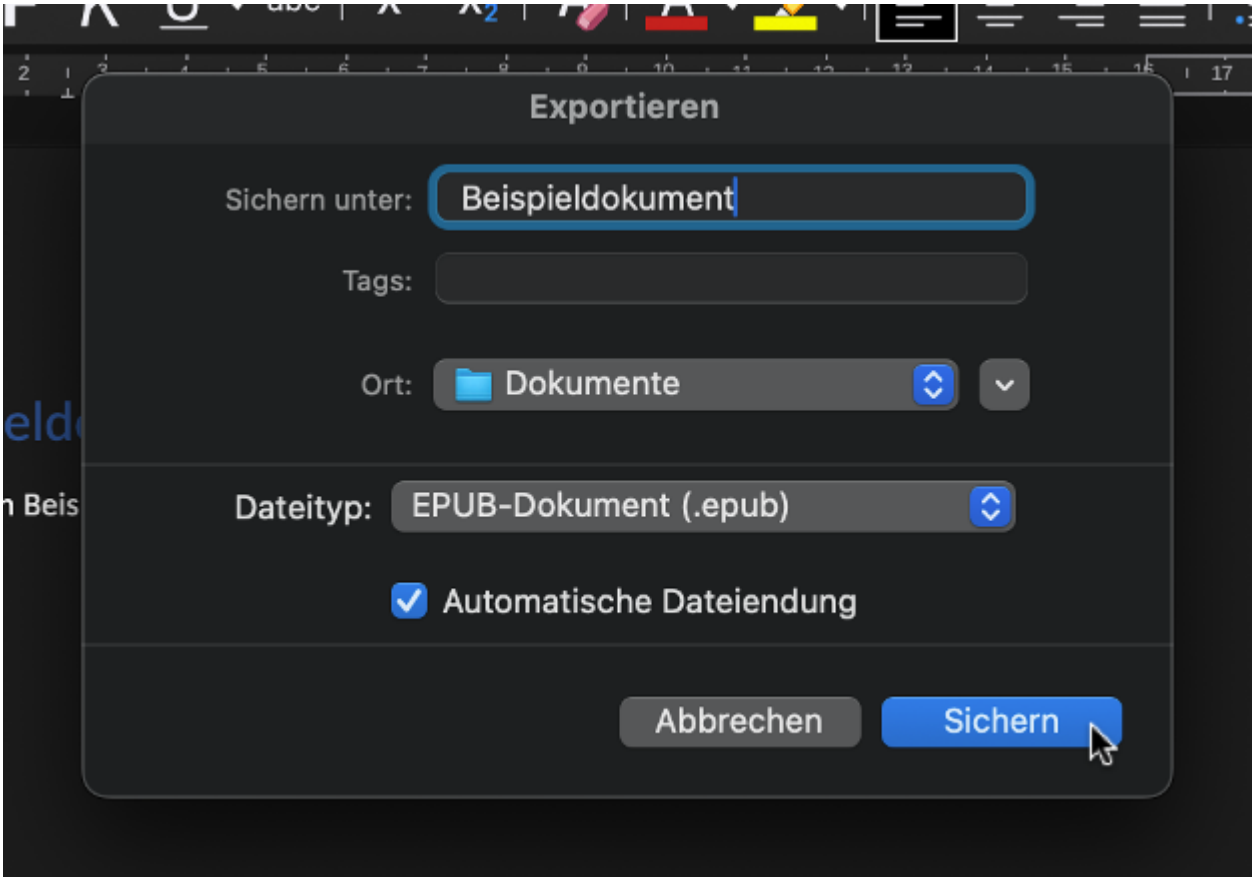





Konfigurieren Sie die Metadaten (wie zum Beispiel Titel, Sprache und Layout).



Speichern Sie das Dokument.




### Vorteile

- Export in EPUB 3.0 ohne Zusatzsoftware.
- Einfache Bedienung.
- Alternativtexte für Bilder werden berücksichtigt.
- Open-Source-Software.

### Nachteile

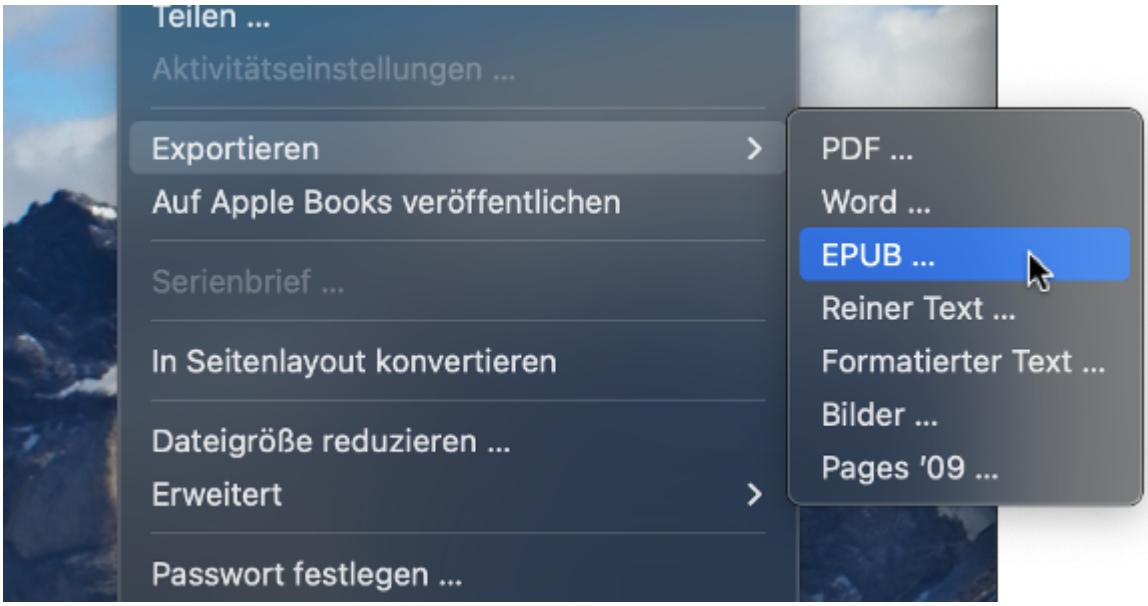
- Formeln werden nicht als MathML exportiert – häufig als Bilder oder unleserlicher Text.
- Eingeschränkte Kontrolle über semantische Struktur und Metadaten.
- Keine EPUB-Vorschau oder automatische Validierung.

## 3. Apple Pages (macOS)

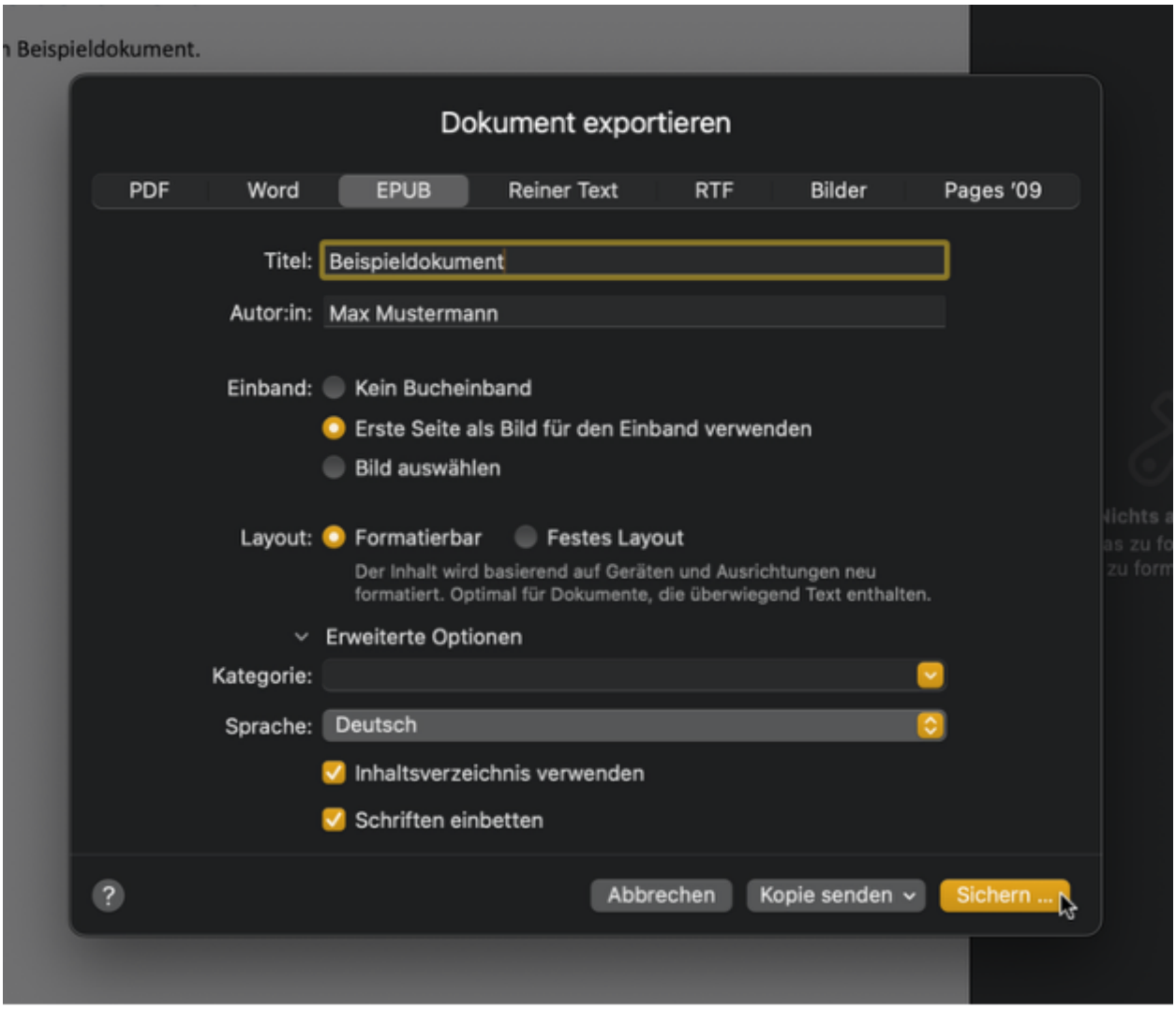
### Vorgehen

Öffnen Sie das Dokument in Pages.

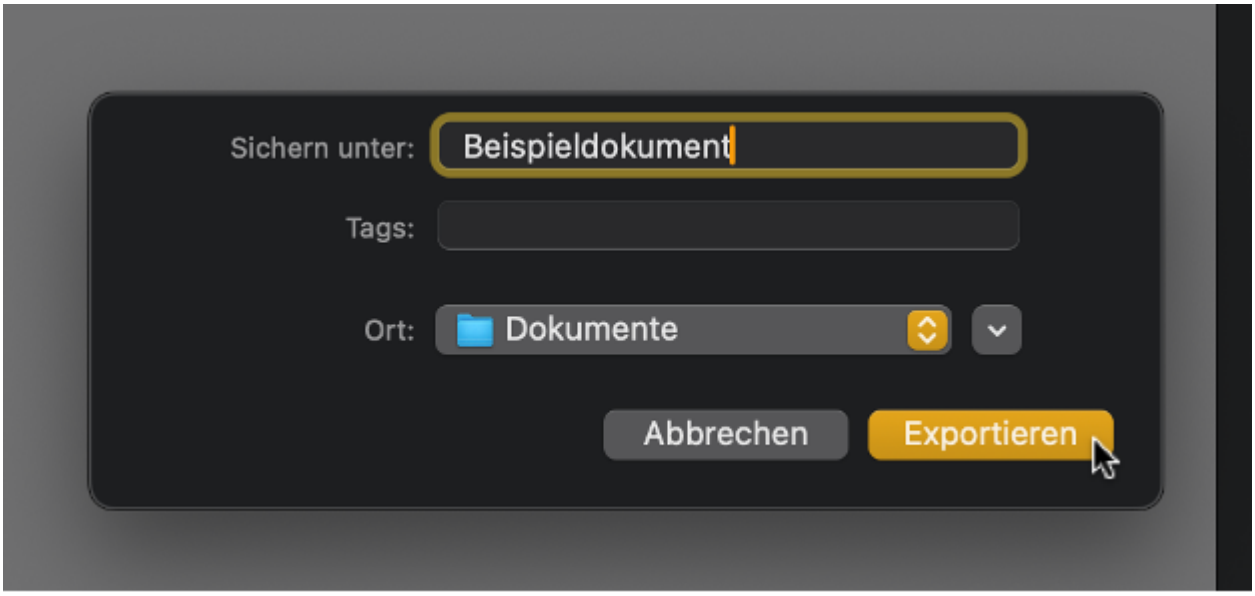
Wählen Sie *Ablage* > *Exportieren* > *EPUB...*




Geben Sie Metadaten an (Titel, Autor) und wählen Sie ein Layout (Reflow oder Fest).




Exportieren Sie die Datei.




## Vorteile

- Schneller, intuitiver Export in EPUB 3.0.
- Layoutwahl: Reflow oder Fixed Layout.
- Für einfache Inhalte ausreichend.

### Nachteile

- Alternativtexte werden teils ignoriert oder nicht korrekt übernommen.
- Formeln werden nicht als MathML exportiert.
- Eingeschränkte Kontrolle über Struktur und Barrierefreiheit.

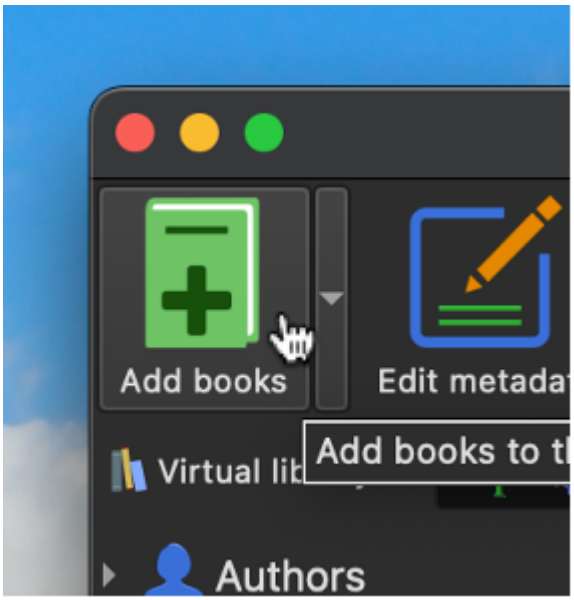
## 4. Calibre

Link zum Tool: Calibre (Calibre-Projekt)

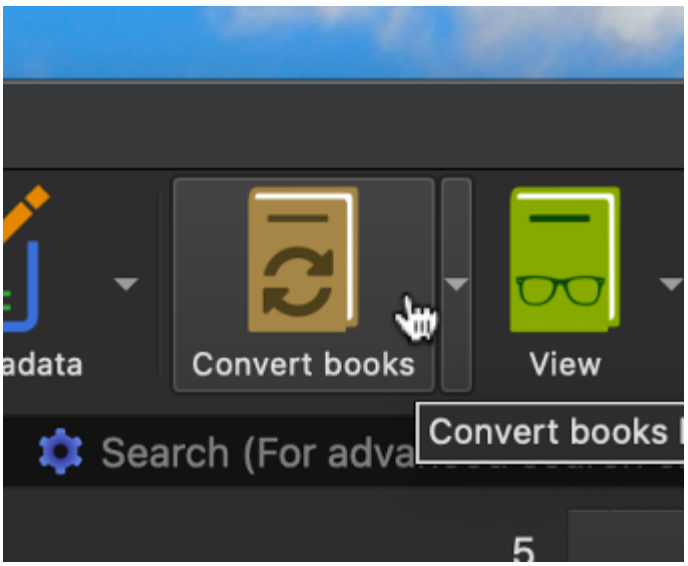
### Vorgehen

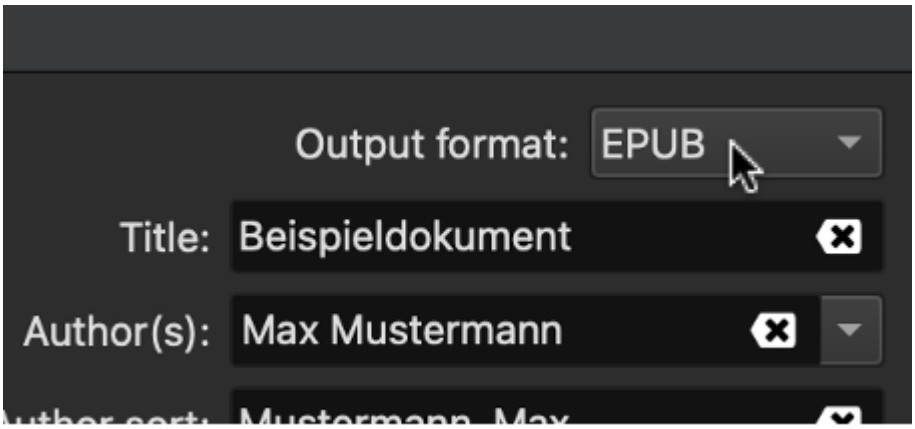
Öffnen Sie Calibre.

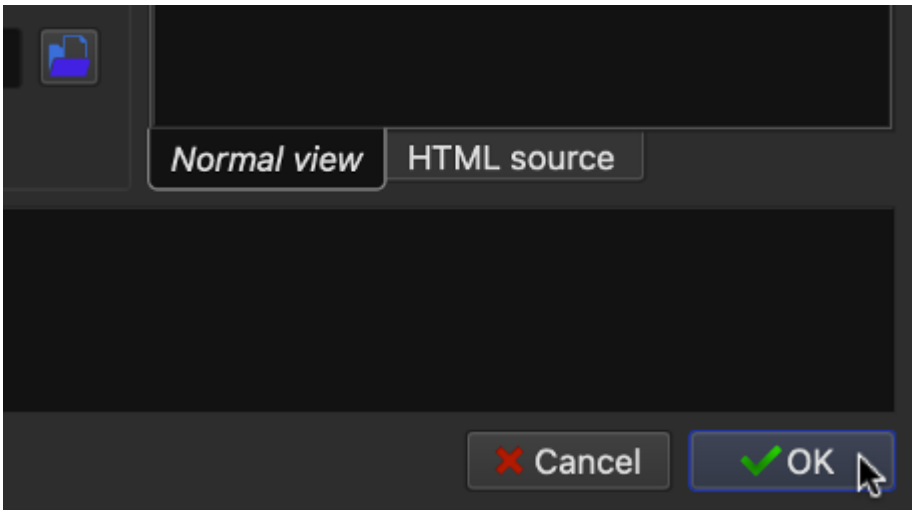
Fügen Sie ein bestehendes Dokument (z. B. DOCX, PDF) hinzu.



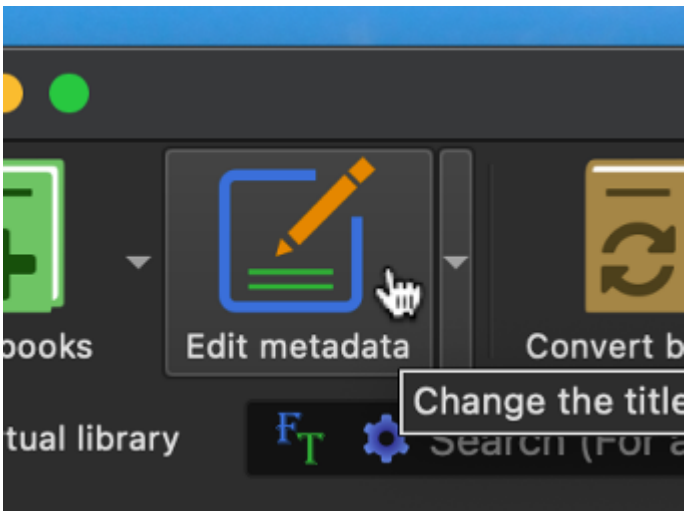

Klicken Sie auf „Bücher konvertieren“ und wählen Sie als Ausgabeformat EPUB aus.

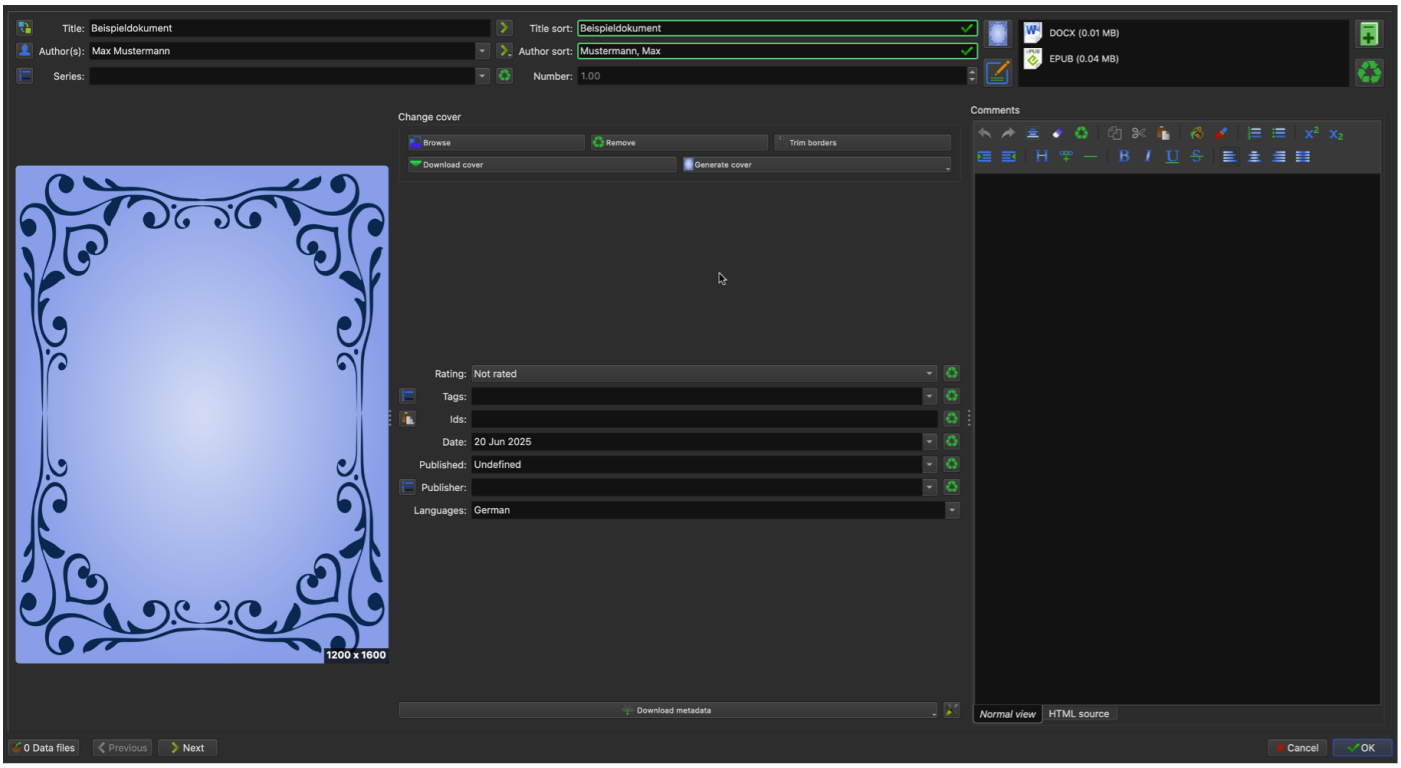




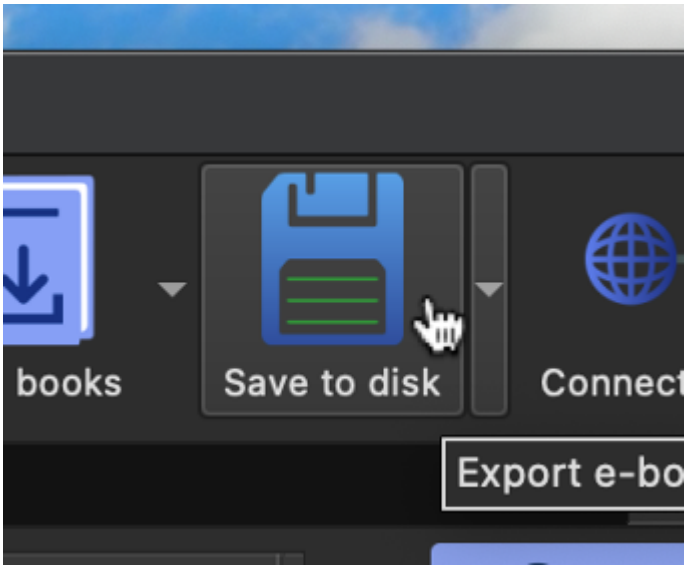




Bearbeiten Sie die Metadaten.






Speichern Sie die Datei.




## Vorteile

- Unterstützt viele Dateiformate.
- Schnelle Konvertierung.
- Einfache Metadatenbearbeitung, Cover einfügbar.

## Nachteile

- Keine Berücksichtigung von Barrierefreiheitsaspekten.
- Alternativtexte und semantische Struktur gehen oft verloren.
- Formeln werden nicht als MathML oder lesbare Inhalte exportiert.
- Nicht geeignet für strukturierte, didaktische Inhalte.

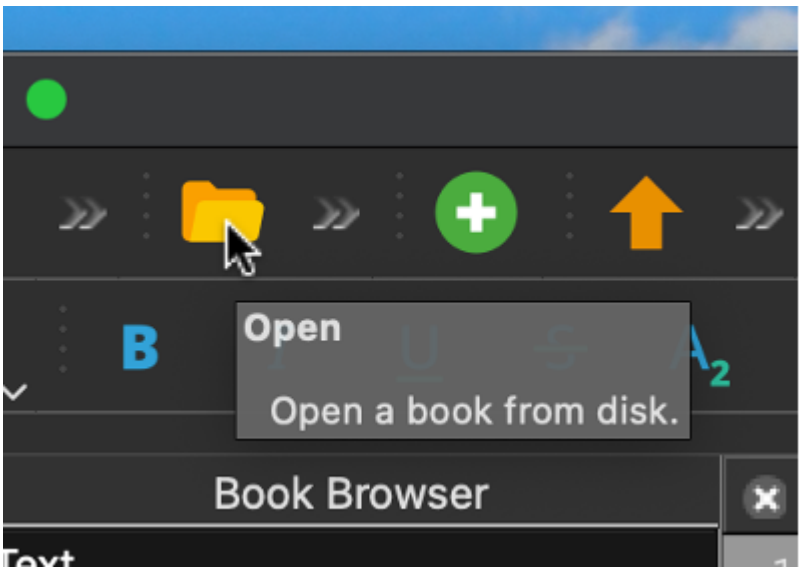
## 5. Sigil

Link zum Tool: Sigil (Sigil-Projekt)

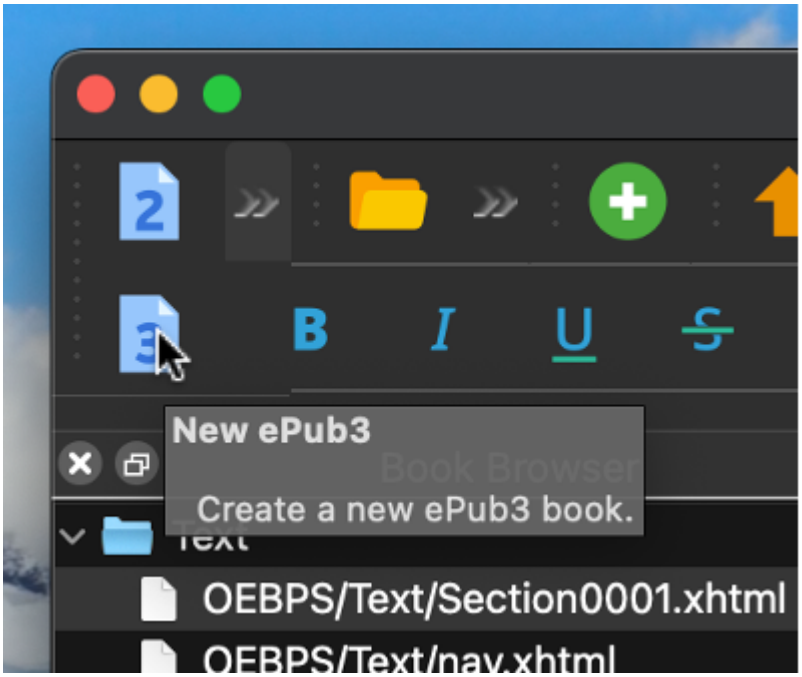
### Vorgehen

Öffnen Sie Sigil.

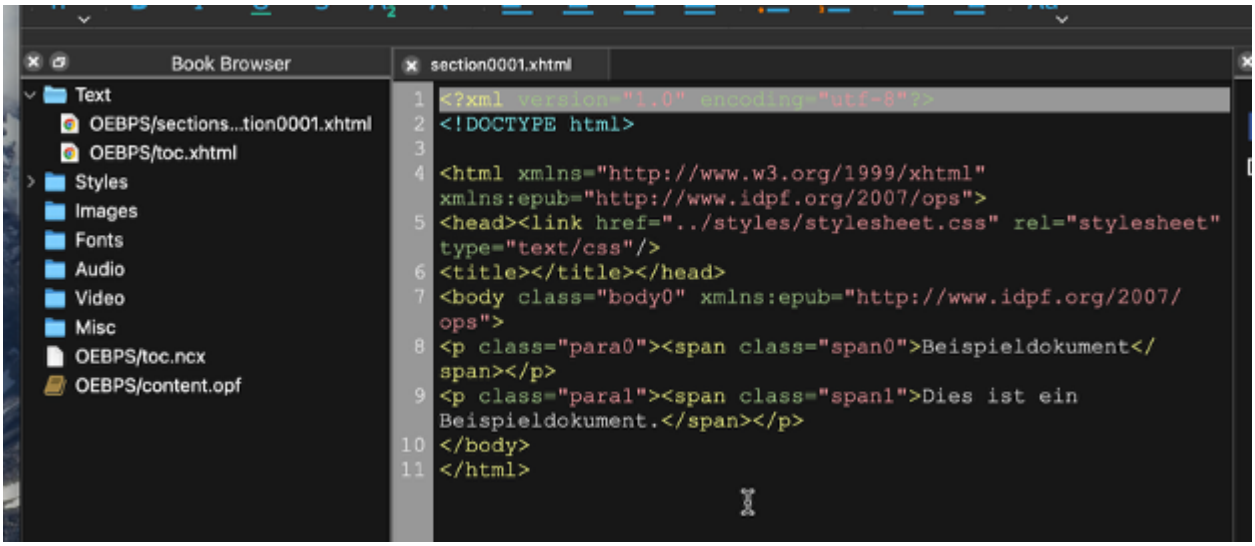
Laden Sie eine bestehende EPUB-Datei

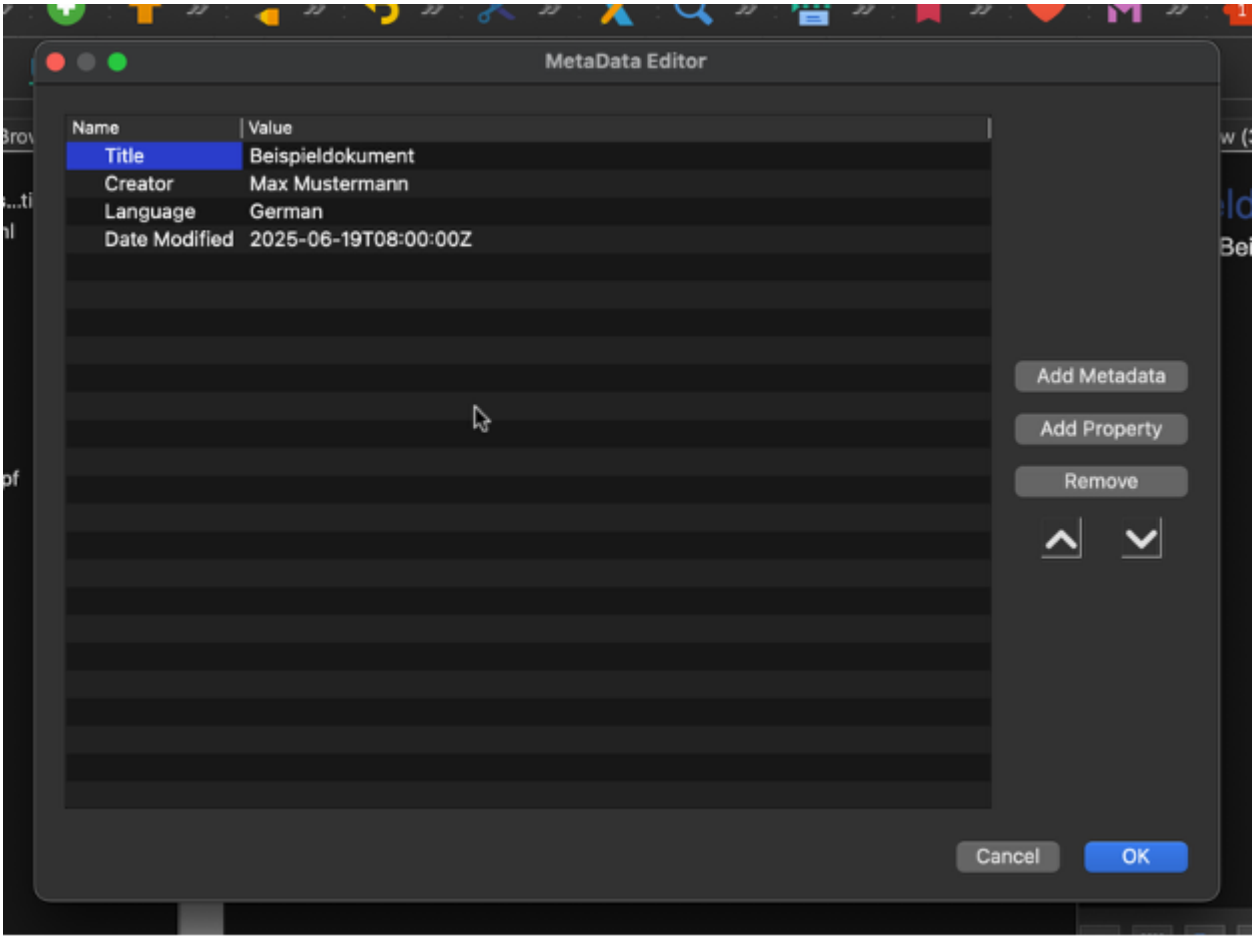



oder starten Sie ein neues Projekt.




Bearbeiten Sie Inhalte (HTML), Struktur, Navigation, Metadaten und Alternativtexte.




Nutzen Sie den integrierten EPUBCheck-Validator.

## Vorteile

- Vollständige Kontrolle über Struktur, HTML, MathML, CSS, Alt-Texte, Metadaten.
- Gut geeignet zur Nachbearbeitung von mit WordToEPUB oder LibreOffice erzeugten EPUBs.
- Unterstützt EPUB 2 bis 3.2 (teilweise 3.3).

## Nachteile

- Technisches Know-how erforderlich (HTML/CSS).
- Keine Word- oder PDF-Integration.

- Kein direktes Erstellen aus Word/LibreOffice – nur Nachbearbeitung.

## Vergleich der EPUB-Erstellungstools

| Kriterium | Word (WordToEPUB) | LibreOffice | Pages | Calibre | Sigil |
|---|---|---|---|---|---|
| EPUB-Version | EPUB 3.0 | EPUB 3.0 | EPUB 3.0 | EPUB 2 /3 | EPUB 2–3.2 |
| MathML-Unterstützung | nein (Bilder) | Teilweise | nein | nein | ja (manuell) |
| Alternativtexte für Bilder | ja | ja | Teilweise | nein | ja (manuell) |
| Metadaten editierbar | Eingeschränkt | Teilweise | Eingeschränkt | ja | ja |
| Barrierefreie Navigation | ja | Teilweise | Eingeschränkt | nein | ja |
| Validierung möglich | ja (integriert) | nein | nein | nein | ja (integriert) |

## Fazit und Empfehlung

Für barrierefreie EPUBs im Bildungsbereich ist **WordToEPUB in Kombination mit Microsoft Word** die empfohlene Lösung, besonders für strukturierte Texte mit Alternativtexten.

**LibreOffice Writer** bietet eine freie Alternative, mit Einschränkungen bei komplexen Inhalten wie Formeln.

**Apple Pages** ist für einfache EPUBs geeignet, aber nicht optimal für Barrierefreiheit.

**Calibre** ist ideal zur Konvertierung, nicht zur Erstellung barrierefreier EPUBs.

**Sigil** empfiehlt sich zur professionellen Nachbearbeitung und Ergänzung von Struktur, Metadaten, MathML und Alternativtexten.

## Tipp zur Prüfung der Barrierefreiheit

Verwenden Sie zur Prüfung Ihrer EPUB-Dateien:

- EPUBCheck (W3C) – für technische Validierung
- Ace by DAISY (DAISY-Konsortium) – für Zugänglichkeitsprüfung nach WCAG/EPUB Accessibility 1.1

## Verweise

### Interne Links

Keine vorhanden

### Externe Links in diesem Artikel

- WordToEPUB (DAISY-Konsortium)
- Calibre (Calibre-Projekt)
- Sigil (Sigil-Projekt)

- EPUBCheck (W3C)
- Ace by DAISY (DAISY-Konsortium)

## Weiterführende Links

- Inclusive Publishing (DAISY-Konsortium)
- EPUBTest (Benetech)
- EPUB 3.2 Spezifikation (W3C)

# Hinweise zu Typst

Online betrachten

Im Folgenden sind Hinweise zur Erstellung von barrierefreien PDF Dokumenten mit Hilfe von Typst zu finden.

## Inhaltsverzeichnis


- Barrierefreie PDF/UA-Dokumente mit Typst - Einführung (3 min)
  - Einführung
  - Ziele dieses Moduls
  - Was bedeutet PDF/UA?
  - Hinweis: Beispiel-Typst-Dokument
- Barrierefreie PDF/UA-Dokumente mit Typst - Semantik und Struktur (2 min)
  - Semantik und Struktur
  - Überschriften
  - Listen
  - Tabellen
  - Querverweise und Links
- Barrierefreie PDF/UA-Dokumente mit Typst - Metadaten und Sprache (4 min)
  - Metadaten und Sprache
  - Bedeutung der Metadaten
  - Metadaten in Typst setzen
  - PDF/UA-Modus aktivieren
  - Sprache definieren
  - Titel und interne Strukturen
  - Alternative Dokumenttitel (barrierefrei)
  - Sprach- und Metadatenfehler vermeiden
  - Beispiel für ein vollständiges barrierefreies Dokument
  - Zusammenfassung
- Barrierefreie PDF/UA-Dokumente mit Typst - Tabellen, Grafiken und Formeln (5 min)
  - Tabellen, Grafiken und Formeln
  - Grundprinzipien für Barrierefreiheit
  - Tabellen
  - Grafiken
  - Formeln
  - Lesereihenfolge bewahren
  - Minimalbeispiel für ein barrierefreies Dokument mit Tabellen, Grafiken und Formeln
  - Zusammenfassung
- Barrierefreie PDF/UA-Dokumente mit Typst - Barrierefreie Typst-Vorlagen (5 min)
  - Barrierefreie Typst-Vorlagen
  - Anforderungen an barrierefreie Vorlagen
  - Grundstruktur einer barrierefreien Typst-Vorlage
  - Beispielvorlage: barrierefreie Typst-Struktur
  - Makro-Erweiterungen für barrierefreie Vorlagen
  - Designempfehlungen für Institutionen
  - Beispiel: Minimaler Projektbericht auf Basis der Vorlage
  - Zusammenfassung

- Barrierefreie PDF/UA-Dokumente mit Typst - Prüfung und Validierung (4 min)
  - Prüfung und Validierung
  - Ziele
  - Grundlagen der PDF/UA-Prüfung
  - Automatische Prüfwerkzeuge
  - Manuelle Prüfung
  - Typische Fehlerquellen
  - Empfohlener Prüfworkflow
  - Beispiel: Minimaler Prüfbericht
  - Zusammenfassung

# Barrierefreie PDF/UA-Dokumente mit Typst - Einführung

Online betrachten

## Einführung

Typst ist ein modernes Textsatzsystem, mit dem strukturiert, automatisierbar und in hoher Qualität Dokumente erstellt werden können. Der Fokus liegt dabei auf einer Kombination aus einfacher Bedienbarkeit, klarer Syntax und direkter Kontrolle über das Layout. Im Unterschied zu reinem Word-ähnlichen Schreiben arbeitet Typst eher wie in einem „programmierbaren" Satzsystem: Man beschreibt Inhalt und Struktur, Typst setzt das Dokument daraus konsistent und reproduzierbar um.

Um Typst zu verwenden, nutzen Sie üblicherweise den Typst Web-Editor oder Visual Studio Code mit Typst-Erweiterung.

Dieses Modul erläutert die Grundlagen für die Erstellung barrierefreier PDF/UA-Dokumente mit Typst.

### Warum Typst?

Für barrierefreie Dokumente ist entscheidend, dass Struktur nicht nur „visuell", sondern semantisch im Dokument vorhanden ist. Typst eignet sich hierfür, weil es durch seinen quelltextbasierten Ansatz typischerweise konsistente, klar strukturierte Ausgaben unterstützt.

Weitere Vorteile sind:

- Reproduzierbarkeit & Konsistenz: Layout-Logik ist im Quelltext enthalten. Dadurch lassen sich Dokumente leichter versionieren und einheitlich wiederverwenden (z. B. für Vorlagen, Skripte oder wissenschaftliche Arbeiten).
- Automatisierung: Zentrale Elemente wie Inhaltsverzeichnisse, Verweise, Nummerierungen, Gliederungen oder Querverweise können systematisch erzeugt werden – ohne manuelles Nachpflegen.
- Weniger „Dokumentenfragilität": Änderungen am Inhalt erfordern nicht zwangsläufig umfangreiche Layoutkorrekturen; Typst sorgt typischerweise dafür, dass Strukturen konsistent bleiben.
- Lesbare Quellen: Viele finden die Typst-Syntax überschaubar und gut wartbar, weil sie strukturelle Informationen direkt abbildet.

Da der Schwerpunkt auf einer klaren Trennung von Inhalt/Struktur und Darstellung liegt, erleichtert Typst die Umsetzung von semantisch korrekten Dokumenten, die für Assistive Technologien besser nutzbar sind.

## Ziele dieses Moduls

- Grundlagen: Was bedeutet PDF/UA-Konformität?
- Rolle von Semantik und Dokumentstruktur
- Spezifika von Typst im Vergleich zu LaTeX/Word

## Was bedeutet PDF/UA?

PDF/UA („Universal Accessibility") ist der internationale Standard für barrierefreie PDF-Dokumente. Er stellt Anforderungen an:

- vollständige Tag-Struktur
- korrekte Überschriftenhierarchie

- Alternativtexte
- Lesereihenfolge
- eingebettete Unicode-Schriften
- maschinenlesbare Metadaten

## Hinweis: Beispiel-Typst-Dokument

Ein minimales PDF/UA-1-konformes Dokument befindet sich unter:

```
#set document(
  title: "Einführung in barrierefreie Typst-Dokumente",
  author: "BFIT-Team",
  lang: "de-DE"
)

#set pdf(accessibility: "ua-1")
#set text(font: "Noto Sans", lang: "de")

= Einführung
Dies ist ein minimaler barrierefreier Beispieltext.

== Abschnitt 1
Ein normaler Absatz folgt.

#figure(
  image("beispiel.png"),
  caption: "Beispielabbildung",
  alt: "Ein Screenshot mit Beispielinhalt"
)
```

# Barrierefreie PDF/UA-Dokumente mit Typst - Semantik und Struktur

Online betrachten

## Semantik und Struktur

Semantik ist der Kern barrierefreier Dokumente. Screenreader benötigen klare Strukturen, um Inhalte korrekt vorzulesen und navigierbar zu machen.

Dieses Modul zeigt, wie Typst folgende Elemente korrekt semantisch auszeichnet:

- Überschriften
- Absätze
- Listen
- Tabellen
- Querverweise

## Überschriften

Typst nutzt einfache Gleichheitszeichen zur Hierarchie:

```
= Ebene 1
== Ebene 2
=== Ebene 3
```

Wichtig: Ebenen dürfen nicht übersprungen werden.

## Listen

Ungeordnet:

```
- Punkt 1
- Punkt 2
```

Geordnet:

```
+ Schritt 1
+ Schritt 2
```

## Tabellen

Tabellen müssen Kopfzeilen haben:

```
#table(
  columns: 2,
  [
    *Name* | *Wert*
    Temperatur | 23°C
  ]
)
```

## Querverweise und Links

Typst unterstützt barrierefreie Verlinkung:

#link(destination: ref(id))[Text] // Querverweis
#link("https://typst.app")[Typst-Webseite] // Externer Link, Beispiel Typst-Webseite

# Barrierefreie PDF/UA-Dokumente mit Typst - Metadaten und Sprache

Online betrachten

## Metadaten und Sprache

Metadaten und Sprachangaben sind zentrale Voraussetzungen für barrierefreie PDF-Dokumente nach PDF/UA-1 und PDF/UA-2. Dieses Modul beschreibt, welche Informationen erforderlich sind, wie sie in Typst korrekt gesetzt werden und welche Regeln zu beachten sind.

## Bedeutung der Metadaten

Metadaten beschreiben das Dokument für Maschinen, z. B. Screenreader, Prüfwerkzeuge oder Archivsysteme
PDF/UA verlangt zwingend folgende Angaben:

- Dokumenttitel
- Autor bzw. herausgebende Stelle
- Sprache des Dokuments
- Strukturhierarchie
- PDF-Tagging aktiv

Ohne diese Angaben gilt ein PDF als nicht konform.

Metadaten ermöglichen:

- Anzeige in PDF-Readern
- korrekte Sprachausgabe in Screenreadern
- Klassifizierung in Dokumentenmanagementsystemen
- korrekte Verarbeitung durch Prüfwerkzeuge wie PAC oder veraPDF

## Metadaten in Typst setzen

Typst definiert Metadaten über den document-Block.

```
#set document(
  title: "Beispieldokument",
  author: "Michaela Musterfrau",
  lang: "de-DE"
)
```

Regeln:

- title muss eindeutig sein und darf nicht leer bleiben.
- lang muss im BCP-47-Standard angegeben werden (z. B. „de-DE“, „en-US“).
- Mehrere Autoren können als Liste, kommagetrennt, gesetzt werden.

Weitere optionale, aber sinnvolle Angaben, beispielsweise “keywords”:

```
#set document(
  title: "Beispieldokument",
```

```
  author: ["Michaela Musterfrau", "Michael Mustermann"],
  lang: "de-DE",
  keywords: ["Barrierefreiheit", "PDF/UA", "Typst"]
)
```

## PDF/UA-Modus aktivieren

Damit Typst ein barrierefreies PDF erzeugt:

```
#set pdf(accessibility: "ua-1")
```

oder für neuere Standards:

```
#set pdf(accessibility: "ua-2")
```

Wenn diese Einstellung fehlt, ist das Ergebnis *nicht* PDF/UA-konform.

## Sprache definieren

Neben den Dokumentmetadaten muss auch die Textsprache gesetzt werden:

```
#set text(lang: "de")
```

Dies steuert:

- Aussprache durch Screenreader
- Silbentrennung
- Tag-Zuweisung im PDF

### Sprachwechsel innerhalb eines Textes

PDF/UA erfordert korrekte Sprachmarkierungen für jeden Abschnitt, der nicht in der Dokumentsprache verfasst ist.

```
#text(lang: "en")[This is an English phrase.]
```

Regeln:

- Sprachwechsel nur setzen, wenn nötig.
- Möglichst kurze Sprachabschnitte markieren.
- Kein Mischen von Sprachen ohne Markierung.

## Titel und interne Strukturen

Die erste Überschrift sollte eine Ebene-1-Überschrift (=) sein.
Ein Dokument ohne H1-Überschrift ist schwer navigierbar.

Beispiel:

```
= Dokumenttitel
Einleitung ...
```

Die H1 muss inhaltlich dem title: in den Metadaten entsprechen oder diesen sinnvoll ergänzen.

## Alternative Dokumenttitel (barrierefrei)

PDF unterstützt einen „Accessible Title“.
Typst setzt den Dokumenttitel automatisch aus den Metadaten.

Beispiel einer alternativen internen Benennung:

```
#set document(
  title: "Technische Dokumentation: Typst und Barrierefreiheit",
  author: "Arbeitsgruppe Barrierefreiheit",
  lang: "de-DE"
)
```

Der sichtbare Titel im PDF-Viewer und der maschinenlesbare Titel sind identisch.

## Sprach- und Metadatenfehler vermeiden

Typische Fehlerquellen:

- Sprache nicht gesetzt # falsche Aussprache
- Sprache nur in den Metadaten gesetzt, nicht im Text (text(lang: ...))
- Platzhalter wie „Titel einfügen“
- Metadaten fehlen vollständig
- Mehrsprachige Dokumente ohne Sprachmarkierung

Diese Fehler führen in PAC und veraPDF zu Warnungen oder Fehlern.

## Beispiel für ein vollständiges barrierefreies Dokument

Minimalbeispiel:

```
#set pdf(accessibility: "ua-1")

#set document(
  title: "Barrierefreie Metadaten in Typst",
  author: "Arbeitsgruppe Barrierefreiheit",
  lang: "de-DE"
)

#set text(font: "Noto Sans", lang: "de")

= Barrierefreie Metadaten
Dies ist ein barrierefreies Beispiel.

#text(lang: "en")[This is an English sentence.]
```

Dieses Dokument erfüllt bereits alle grundlegenden Anforderungen an Metadaten und Sprachangaben.

## Zusammenfassung

Dieses Modul zeigte die notwendigen Schritte zur barrierefreien Dokumentbeschreibung:

- Dokumenttitel, Autor, Sprache und Schlüsselwörter setzen

- PDF/UA-Modus aktivieren
- Sprachwechsel korrekt markieren
- klare Titel- und Überschriftenstrukturen verwenden

Diese Angaben bilden die Grundlage für jedes PDF/UA-konforme Typst-Dokument.

# Barrierefreie PDF/UA-Dokumente mit Typst - Tabellen, Grafiken und Formeln

Online betrachten

## Tabellen, Grafiken und Formeln

Tabellen, Grafiken und Formeln gehören zu den häufigsten nicht-textuellen Inhalten in Dokumenten. Für barrierefreie PDF/UA-1- und PDF/UA-2-Dokumente müssen sie korrekt strukturiert, ausgezeichnet und beschrieben sein. Dieses Modul erläutert die Anforderungen und zeigt, wie sie in Typst umgesetzt werden.

## Grundprinzipien für Barrierefreiheit

Für PDF/UA gilt:

- Jede Grafik benötigt einen Alternativtext.
- Tabellen müssen Kopfzeilen besitzen und logisch navigierbar sein.
- Formeln benötigen zugängliche Repräsentationen und bei Bedarf textuelle Erläuterungen.
- Inhalte müssen in der logischen Lesereihenfolge stehen.
- Keine rein visuellen Informationen ohne textliche Ergänzung.

Typst unterstützt diese Anforderungen über semantische Strukturen, figure-Elemente, Tabellenfunktionen und Math-Syntax.

## Tabellen

Tabellen müssen klar strukturiert sein, Kopfzeilen besitzen und nicht für Layoutzwecke missbraucht werden.

Beispiel:

```
#table(
  columns: 3,
  align: horizon,
  [
    *Parameter* | *Wert* | *Einheit*
    Temperatur  | 23     | °C
    Länge       | 12     | cm
  ]
)
```

Regeln:

- Kopfzeilen müssen erkennbar formatiert sein (Typst markiert fett als Header-Zelle). Für Fettschrift umschließen Sie das Wort/die Wörter mit Sternchen.
- Keine verschachtelten Tabellen.
- Tabelleninhalt logisch sortieren.
- Tabellen nicht als Bild einfügen.

Fehler vermeiden:

- Leere Tabellenzellen nicht kommentarlos lassen.

- Kein Zusammenführen von Zellen für rein optische Effekte.
- Tabellen nicht zweckentfremden, um Layout zu erzeugen.

## Grafiken

Grafiken müssen mit Alternativtext (alt) und beschreibender Bildunterschrift (caption) versehen werden.

Beispiel:

```
#figure(
  image("diagramm.png"),
  caption: "Beispieldiagramm der Messwerte",
  alt: "Liniendiagramm mit drei Messpunkten zu 10, 14 und 18 Einheiten"
)
```

Regeln für Alternativtexte:

- Sachlich, kurz, beschreibend.
- Keine Phrasen wie „Bild von …“.
- Der Alt-Text ersetzt die Grafik vollständig.
- Komplexe Diagramme benötigen zusätzlich eine textliche Auswertung im Fließtext.

Empfohlen:

- Grafikdateien in hoher Auflösung oder als SVG verwenden.
- Keine Texte in Grafiken einbetten (nur wenn unvermeidbar, dann zusätzlich beschreiben).

## Formeln

Mathematische Inhalte werden in Typst über die Math-Syntax gesetzt:

Inline:

```
$E = m * c^2$
```

Blockform:

```
$
a^2 + b^2 = c^2
$
```

Barrierefreiheit:

- Typst erzeugt intern MathML-Strukturen, die für Screenreader zugänglich sind.
- Bei komplexen Formeln optional zusätzliche Beschreibung im Fließtext ergänzen.
- Keine Screenshots oder Bilder von Formeln verwenden.

Empfehlung:

- Bei langen Gleichungssystemen: Schrittweise erklären.
- Bei wissenschaftlichen Dokumenten: Glossar mit Formelzeichen bereitstellen.

## Lesereihenfolge bewahren

Tabellen, Grafiken und Formeln müssen im Dokumentfluss an der korrekten Stelle stehen. Typische Fehler:

- Seitliches Platzieren mit absoluter Positionierung.
- Grafiken mit Umfließen, das die Lesereihenfolge bricht.
- mehrere Figuren oder Tabellen ohne erklärenden Text hintereinander.

Typst folgt immer dem logischen Ablauf der Eingabe – daher visuelle Experimente vermeiden, die die Struktur beschädigen könnten.

## Minimalbeispiel für ein barrierefreies Dokument mit Tabellen, Grafiken und Formeln

```
#set pdf(accessibility: "ua-1")
#set document(
  title: "Beispiel mit Tabellen, Grafiken und Formeln",
  author: "ACCESS@KIT",
  lang: "de-DE"
)

#set text(font: "Noto Sans", lang: "de")

= Tabellen, Grafiken und Formeln

== Tabelle
#table(
  columns: 2,
  [
    *Messgröße* | *Wert*
    Temperatur  | 21°C
    Druck       | 1013 hPa
  ]
)

== Grafik
#figure(
  image("werte.png"),
  caption: "Messverlauf über die Zeit",
  alt: "Liniendiagramm mit steigendem Verlauf über fünf Zeitpunkte"
)

== Formel
Die berühmte Formel lautet:
$E = m * c^2$

#text(lang: "en")[This formula describes the equivalence of mass and energy.]
```

## Zusammenfassung

Dieses Modul zeigt, wie Tabellen, Grafiken und Formeln in Typst barrierefrei gestaltet werden:

- Tabellen mit klaren Kopfzeilen und logischer Struktur.
- Grafiken stets mit alt und caption.
- Formeln semantisch setzen und bei Bedarf erklären.
- Inhalte nur im normalen Dokumentfluss verwenden.

Damit erfüllen Typst-Dokumente zentrale Anforderungen der PDF/UA-Richtlinien für nicht-textuelle Inhalte.

# Barrierefreie PDF/UA-Dokumente mit Typst - Barrierefreie Typst-Vorlagen

Online betrachten

## Barrierefreie Typst-Vorlagen

Barrierefreie Vorlagen sind essenziell, um konsistent PDF/UA-1- oder PDF/UA-2-konforme Dokumente zu erzeugen. Dieses Modul stellt ein vollständiges Grundgerüst für Typst-Vorlagen bereit, die in Hochschulen, Behörden, Vereinen oder Projekten eingesetzt werden können.

Ziele:

- Einheitliche barrierefreie Formatierung sicherstellen
- Fehlerquellen im Layout vermeiden
- Wiederverwendbare Strukturen anbieten
- Automatisierte UA-konforme PDF-Erzeugung ermöglichen

Die hier vorgestellte Vorlage kann direkt verwendet oder institutionell angepasst werden.

## Anforderungen an barrierefreie Vorlagen

Eine barrierefreie Typst-Vorlage muss folgende Anforderungen erfüllen:

- eingebettete Unicode-Schriften
- definierte Dokumentmetadaten
- definierte Basis-Sprache
- semantische Überschriftenstruktur
- Stilregeln für Tabellen, Abbildungen, Formeln
- vordefinierte Makros für Gleichmäßigkeit
- klarer Dokumentfluss (keine absolute Positionierung)

Diese Regeln müssen in der Vorlage technisch hinterlegt sein, nicht erst beim Schreiben des Dokuments.

## Grundstruktur einer barrierefreien Typst-Vorlage

Eine vollständige Typst-Vorlage besteht aus:

- Dokumenteinstellungen
- globale Text-Formatierungen
- Kapitelstruktur
- Makros für wiederkehrende Elemente
- Musterseiten bzw. Inhaltsbausteinen
- Beispielabschnitten

Im Folgenden ein vollständiges Vorlagenbeispiel.

## Beispielvorlage: barrierefreie Typst-Struktur

```
#set page(size: "a4")
#set pdf(accessibility: "ua-1")
```

```
#set document(
  title: "Barrierefreie Typst-Vorlage",
  author: "Institution / Projekt",
  lang: "de-DE",
)

#set text(
  font: "Noto Sans",
  size: 11pt,
  lang: "de",
  leading: 1.3em,
)

#let heading-style(level) = {
  if level == 1 { text(size: 20pt, weight: "bold") }
  else if level == 2 { text(size: 16pt, weight: "bold") }
  else if level == 3 { text(size: 13pt, weight: "bold") }
  else { text(size: 11pt, weight: "bold") }
}

#set heading(numbering: "1.1.1", style: heading-style)

= Titel des Dokuments
Einleitungstext. Diese Vorlage erfüllt alle Grundlagen für barrierefreie PDF-Dokumente in Typst.

== Kapitel 1 – Grundlagen
Dies ist ein Beispielabsatz.

#figure(
  image("beispielbild.png"),
  caption: "Beispielbild",
  alt: "Beschreibung der dargestellten Inhalte"
)

== Tabelle
#table(
  columns: 2,
  align: horizon,
  [
    *Parameter* | *Wert*
    Temperatur  | 22°C
    Druck       | 1015 hPa
  ]
)

== Formel
Eine bekannte Formel lautet:
$E = m * c^2$
```

#text(lang: "en")[Mass and energy are equivalent according to Einstein's theory.]

## Makro-Erweiterungen für barrierefreie Vorlagen

Um die Bedienung zu vereinfachen, empfehlen sich zusätzliche Makros. Damit kann semantisch sichergestellt werden, dass notwendige Parameter einheitlich angegeben werden.

### Makro für standardisierte Abbildungen

```
#let altfigure(imagepath, caption: "", alt: "") = figure(
  image(imagepath),
  caption: caption,
  alt: alt,
)
```

Beispiel:

```
#altfigure(
  "diagramm.svg",
  caption: "Temperaturverlauf",
  alt: "Diagramm mit ansteigender Temperatur über drei Messpunkte"
)
```

### Makro für standardisierte Tabellen

```
#let datatable(columns: 2, body) = table(
  columns: columns,
  align: horizon,
  body
)
```

### Makro für Formelerläuterungen

```
#let explain(formula, text) = block(
  formula,
  text
)
```

Beispiel:

```
#explain(
  $a^2 + b^2 = c^2$,
  "Der Satz des Pythagoras beschreibt die Beziehung der Seiten im rechtwinkligen Dreieck."
)
```

## Designempfehlungen für Institutionen

- Kontrastreiche Farben verwenden
- Serifenschriften für Mengentext vermeiden
- Keine Layouttricks mit Textboxen oder absoluten Koordinaten
- Logo nur als dekoratives Bild ohne Informationsgehalt # alt: ""

- Standardisierte Kapitel- und Abschnittsstruktur festlegen

## Beispiel: Minimaler Projektbericht auf Basis der Vorlage

```
= Projektbericht 2026
Kurzbeschreibung des Projekts.

== Einleitung
Dieser Bericht wurde mit einer barrierefreien Typst-Vorlage erstellt.

== Methoden
Beschreibung der angewandten Methoden.

== Ergebnisse
#altfigure(
  "messung.svg",
  caption: "Messverlauf über Zeit",
  alt: "Chart mit langsamer Aufwärtsbewegung der Messwerte"
)

== Fazit
Zusammenfassung der wichtigsten Erkenntnisse.
```

## Zusammenfassung

Dieses Modul liefert eine vollwertige barrierefreie Typst-Vorlage, inklusive:

- Dokumentmetadaten
- semantischer Struktur
- Tabellen-, Grafik- und Formelelementen
- Makros zur Vereinheitlichung
- Beispielaufbau eines vollständigen Berichts

Damit können Institutionen konsistente PDF/UA-konforme Vorlagen aufbauen und langfristig barrierefreie Workflows sicherstellen.

# Barrierefreie PDF/UA-Dokumente mit Typst - Prüfung und Validierung

Online betrachten

## Prüfung und Validierung

Die Prüfung barrierefreier PDF-Dokumente ist ein zwingender Bestandteil des Qualitätsprozesses. Dieses Modul beschreibt Werkzeuge, Prüfverfahren und empfohlene Abläufe zur Validierung von PDF/UA-1- und PDF/UA-2-Dokumenten, die mit Typst erzeugt wurden.

## Ziele

- Sicherstellen, dass das erzeugte PDF vollständig den Anforderungen von PDF/UA entspricht
- Einsatz geeigneter Prüf- und Validierungstools
- Kombination aus automatischer und manueller Prüfung
- Erkennen und Beheben typischer Fehler

## Grundlagen der PDF/UA-Prüfung

PDF/UA verlangt u. a.:

- korrekte Tag-Struktur
- vollständige Dokumentmetadaten
- korrekte Lesereihenfolge
- Alternativtexte
- Unicode-konforme, eingebettete Schriften
- korrekte Rollenabbildungen
- semantische Konsistenz (keine übersprungenen Überschriftenebenen)

Diese Anforderungen müssen sowohl automatisiert als auch manuell überprüft werden.

## Automatische Prüfwerkzeuge

### PAC 2024

Das zentrale Prüfwerkzeug für PDF/UA:

- kostenlos
- prüft Tag-Struktur, Metadaten, Lesereihenfolge
- liefert verständliche Hinweise und Fehlerlisten

### veraPDF

Open-Source-Validator für PDF-Standards:

- streng regelbasiert
- unterstützt PDF/UA
- nützlich zur Prüfung technisch schwieriger Randfälle

### Adobe Acrobat Preflight

Kommerziell, aber hilfreich:

- detaillierte Analyse
- Validierungsprofile für PDF/UA-1 und PDF/UA-2
- ermöglicht gezielte Fehlerkorrektur

## Manuelle Prüfung

Automatische Prüfungen reichen nicht aus. Screenreader-Tests sind Pflicht.

### Empfohlene Screenreader

- NVDA (Windows, kostenlos)
- JAWS (Windows, kommerziell, häufig im Berufsalltag genutzt)
- VoiceOver (macOS/iOS)

### Zu prüfende Aspekte

- Stimmt die vorgelesene Überschriftenhierarchie?
- Wird der Text in der richtigen Lesereihenfolge ausgegeben?
- Sind Tabellen vollständig navigierbar?
- Werden Abbildungen mit korrekten Alternativtexten angesagt?
- Werden Formeln korrekt erkannt bzw. ist eine verbale Beschreibung vorhanden?

## Typische Fehlerquellen

### Fehlende Semantik

Ursache: Texte durch Formatierung statt durch echte Überschriften gesetzt.
Lösung: Überschriften mit = in Typst setzen.

### Fehlende Alt-Texte

Ursache: eingebundene Bilder ohne alt:-Angabe.
Lösung: Immer alt: im figure-Element angeben.

### Unvollständige Metadaten

Ursache: fehlende oder unvollständige Angaben in #set document().
Lösung: Titel, Autor, Sprache vollständig angeben.

### Tabellen ohne Kopfzeile

Ursache: erste Tabellenzeile visuell fett, aber ohne semantische Kennzeichnung.
Lösung: Kopfzeilen als *fett* (Typst erzeugt daraus <TH>-Zellen).

## Empfohlener Prüfworkflow

1. **Typst-PDF erzeugen**
   Sicherstellen, dass #set pdf(accessibility: "ua-1") bzw. "ua-2" gesetzt ist.

2. **Automatischer Test (PAC 2024)**
    - Tag-Struktur prüfen
    - Metadaten prüfen
    - Fehlermeldungen dokumentieren
3. **Abgleich mit veraPDF**
    - technische Validierung
    - Standardspezifika prüfen
4. **Manueller Screenreader-Test**
    - gesamtes Dokument testweise vorlesen lassen
5. **Korrekturen am Typst-Dokument**
    - strukturelle Fehler beheben
    - Alternativtexte ergänzen
    - Tabellen und Überschriften optimieren
6. **Erneute Validierung**
    - finaler PAC-Durchlauf muss „grün“ sein
    - bei UA-2: zusätzlich Preflight verwenden

## Beispiel: Minimaler Prüfbericht

Prüfbericht – PDF/UA-Validierung
Dokument: projektbericht.pdf
Datum: 2026-03-23

Automatische Prüfung (PAC 2024):
- Tag-Struktur: OK
- Überschriftenhierarchie: OK
- Alternativtexte: 1 Fehler (Abbildung 3 ohne alt)
- Lesereihenfolge: OK

veraPDF:
- PDF/UA-1: konform

Screenreader-Test (NVDA):
- Tabelle im Kapitel 2 vollständig navigierbar
- Abbildung 3 ohne Beschreibung # korrigiert

Fazit: Nach Korrektur vollständig PDF/UA-1-konform.

## Zusammenfassung

Dieses Modul vermittelt:

- die wichtigsten Tools zur PDF/UA-Prüfung

- die Kombination aus automatischer und manueller Kontrolle
- typische Fehler und deren Ursachen
- einen praxisbewährten Prüfworkflow

Damit lassen sich mit Typst erzeugte PDF-Dokumente zuverlässig als PDF/UA-1 oder PDF/UA-2 validieren und normgerecht veröffentlichen.

# Hinweise zu Google Workspace

Online betrachten

Im Folgenden sind Hinweise zur Erstellung von barrierefreien Dokumenten mit der Google Workspace Suite zu finden.

## Inhaltsverzeichnis


- Google Docs (4 min)
  - Grundsätzliche Überlegungen
  - Wie lässt sich mit Google Docs eine PDF/UA erstellen?
  - Verweise
- Google Sheets (4 min)
  - Grundsätzliche Überlegungen
  - Lösungswege
  - Verweise
- Google Slides (4 min)
  - Grundsätzliche Überlegungen
  - Lösungswege
  - Verweise

# Google Docs

Online betrachten

## Grundsätzliche Überlegungen

Mit Google Docs erstellte Dokumente können als PDF heruntergeladen werden. Dabei erzeugt Google Docs automatisch ein getaggtes PDF-Dokument mit den grundlegenden Strukturelementen wie Überschriften, Absätzen, Listen, Tabellen und Links. Für bestmögliche Ergebnisse sollte das Dokument auf einer Vorlage basieren, die bereits unter Berücksichtigung der Anforderungen an die Barrierefreiheit erstellt wurde.

Allerdings weist der PDF-Export noch wesentliche Einschränkungen auf. So fehlen bei Listen beispielsweise die für eine korrekte semantische Struktur erforderlichen Tags Lbl und LBody. Darüber hinaus fehlen oder sind unvollständig:

- Metadaten
- Alternativtexte für Bilder und Hyperlinks
- semantische Tags für Inhaltsverzeichnisse und Fußnoten
- Tabellenattribute
- die Kennzeichnung dekorativer Bilder als Artefakte
- Sprachinformationen
- eine konsistente Lesereihenfolge

Für einfache Dokumente ist der Standardexport mit geringem Nachbearbeitungsaufwand häufig ausreichend. Für komplexere Dokumente genügt der Standardexport jedoch nicht, sodass eine umfassende Nachbearbeitung erforderlich ist.

## Wie lässt sich mit Google Docs eine PDF/UA erstellen?

### Barrierefreiheit von Anfang an berücksichtigen

Die Barrierefreiheit beginnt bereits bei der Gestaltung eines Dokuments. Hier werden die Grundlagen für den uneingeschränkten Zugang gelegt. Dabei ist soweit möglich zu beachten:

- Dokumenttitel und Sprache festlegen
- Dokument logisch und vollständig strukturieren, sinnvolle Dokumenthierarchie
- Überschriften mit Formatvorlagen
- Ausreichend große Schriften verwenden und erkennbarer Abstände
- Listen und Tabellen inkl. Kopfzeilen korrekt anlegen
- Ausreichende Farbkontraste verwenden mit hohem Kontrast zwischen Text und Hintergrund
- Informationen niemals ausschließlich über Farbe vermitteln
- Alternativtexte für informative Bilder und Hyperlinks

In dem Beitrag über Microsoft Word finden sich ergänzende Hinweise und grundsätzliche Überlegungen zur Nutzung einer Textverarbeitung für die Erstellung barrierefreier Dokumente.

## Lösungswege

### Einsatz zusätzlicher Add-ons

Lösungen wie Grackle Workspace erweitern Google Docs um ein Framework für Barrierefreiheit und verfolgen damit einen ähnlichen Ansatz wie axesWord für Microsoft Word (vergleiche dazu auch den Abschnitt Microsoft Word). Sie bieten unter anderem Assistenten für Tabellen, Werkzeuge zur Kennzeichnung von Artefakten sowie Funktionen zur Festlegung der Dokumentensprache. Beim PDF-Export erfolgt die Ausgabe anschließend über eine eigene, standardkonforme Engine.

### Der Microsoft-Word-Weg

Das Dokument wird zunächst als .docx heruntergeladen. Anschließend erfolgt die Prüfung und Überarbeitung mit den Barrierefreiheitsfunktionen von Microsoft Word. Der Export als barrierefreies PDF-Dokument erfolgt schließlich aus der Desktop-Version von Word. Dabei sind die bekannten Einschränkungen des Word-Exports zu berücksichtigen.

### Manuelle Nachbearbeitung

Das aus Google Docs exportierte PDF-Dokument dient als Ausgangsdokument und wird anschließend mit beispielsweise Acrobat Pro, spezialisierten Lösungen wie von PDFix manuell überarbeitet oder bei Bedarf vollständig neu getaggt. Dabei werden die Tag-Struktur, die Lesereihenfolge sowie die semantische Auszeichnung und Struktur vonListen oder Tabellen korrigiert. Basiert das Dokument auf einer sauberen und konsistenten Vorlage, können Automatisierungswerkzeuge wie beispielsweise PDFix Desktop Pro die vorhandenen Strukturen erkennen und das Dokument weitgehend automatisiert in eine PDF/UA-konforme Datei überführen.

## Verweise

### Interne Links

- Microsoft Word

### Externe Links in diesem Artikel

- Grackle Workspace (Grackle)
- PDFix Solutions (PDFix)

### Weiterführende Links

- Techniques for Accessible PDF (Techniques for Accessible PDF)
- Tools for Accessible PDF

# Google Sheets

Online betrachten

## Grundsätzliche Überlegungen

Vergleichbar mit den anderen Google Anwendungen gibt es grundsätzlich 3 Wege zur Erstellung barrierefreier PDF/UA aus Google Sheets. Dabei gilt es die Barrierefreiheit von Anfang an zu berücksichtigen, denn die Barrierefreiheit beginnt bereits bei der Gestaltung eines Dokuments. Hier werden die Grundlagen für den uneingeschränkten Zugang gelegt. Dabei ist soweit in Sheets möglich zu beachten:

- Dokument logisch und vollständig strukturieren, sinnvolle Dokumenthierarchie
- Überschriften mit einheitlichen Formatvorlagen
- Ausreichend große Schriften verwenden und erkennbarer Abstände
- Ausreichende Farbkontraste verwenden mit hohem Kontrast zwischen Text und Hintergrund
- Informationen niemals ausschließlich über Farbe vermitteln

Unabhängig von der barrierfreien Gestaltung gilt es zudem zu überlegen, ob Google Sheets für den konkreten Anwendungsfall das geeignete Tool ist oder die Tabellen in einem Textverarbeitungs- oder Präsentationsprogramm platziert werden sollten. Grundsatzüberlegungen und Hinweise zu dem Aspekt finden sich in dem Beitrag über Microsoft Excel.

## Lösungswege

### Nativer Export von PDF aus Google Sheets

Der Standardexport für PDF basiert auf der Druckansicht und enthält keinen semantischen Tag-Baum. Aus diesem Grund muss die Datei im Zuge der Nachbearbeitung komplett mit Tools wie beispielsweise Adobe Acrobat Pro oder PDFix Desktop Pro neu getaggt und für PDF/UA aufbereitet werden.

### Export mit Grackle Sheets

Grackle Workspace erweitert Google Sheets um Funktionen zur Beschreibung von Tabellen und Alternativtexten. Der eigentliche Mehrwert der Lösung liegt jedoch nicht im PDF-Export, sondern in der Übernahme dieser Informationen in andere Formate, insbesondere HTML. Für die direkte Erstellung einer PDF/UA ist dieser Weg derzeit nur eingeschränkt geeignet und ersetzt keine abschließende Prüfung oder Nachbearbeitung.

### Workaround über Microsoft Excel oder Google Docs

Für hochwertige PDF/UA-Dokumente können die Tabellen nach Microsoft Excel oder Google Docs exportiert werden. In der Anwendung von Microsoft können die Barrierefreiheitsprüfung und der PDF-Export genutzt werden, während Google Docs zumindest eine Grundlage für einen getaggten PDF-Export bietet. Zur Fertigstellung muss das erzeugte PDF auch bei diesem Weg mit einem Validator wie dem PAC oder veraPDF und nachbearbeitet werden.

# Verweise

## Interne Links

- Microsoft Excel

## Externe Links in diesem Artikel

- Grackle Workspace (Grackle)
- PDFix Solutions (PDFix)
- veraPDF (veraPDF)
- PDF Accessibility Checker (PAC) (axes4)

## Weiterführende Links

- Techniques for Accessible PDF (Techniques for Accessible PDF)
- Tools for Accessible PDF (Tools for Accessible PDF)

# Google Slides

Online betrachten

## Grundsätzliche Überlegungen

Für die Erstellung einer PDF/UA aus Google Slides lassen sich drei mögliche Vorgehensweisen unterscheiden. Die Barrierefreiheit beginnt dabei bereits bei der inhaltlichen Gestaltung der Präsentation, denn hier werden die optischen sowie strkturellen Grundlagen für den uneingeschränkten Zugang gelegt. Zu diesen Zweck sollte die folgenden Regeln für Barrierefreiheit beachtet werden:

- Dokumenttitel und Sprache festlegen
- Dokument logisch und vollständig strukturieren, sinnvolle Dokumenthierarchie
- Überschriften mit Formatvorlagen oder einheitlichen Regeln
- Ausreichend große Schriften verwenden und erkennbarer Abstände zwischen den Absätzen
- Listen und Tabellen inkl. Kopfzeilen eindeutig strukturiert anlegen
- Ausreichende Farbkontraste verwenden mit hohem Kontrast zwischen Text und Hintergrund
- Informationen niemals ausschließlich über Farbe vermitteln
- Alternativtexte für informative Bilder und Hyperlinks

In dem Beitrag über Microsoft PowerPoint finden sich in Ergänzung allgemeingültige und grundsätzliche Überlegungen zur Gestaltung und Strukturierung von barrierefreien Präsentationen.

## Lösungswege

### Nativer PDF-Export

Der Standardexport in PDF ist für PDF/UA nicht ausreichend, denn in der Ausgabe fehlt ein vollständiger semantischer Tag-Baum, Überschriften werden nicht erkannt, die Lesereihenfolge ist fehlerhaft und wichtige Informationen wie Dokumentensprache, Metadaten oder Alternativtexte werden nicht oder nur unzuverlässig übernommen. Das Ergebnis ist ein visuell korrektes, aber nicht barrierefreies PDF, das mit einem Screenreader nicht gelesen werden kann.

Dies bedeutet, dass das PDF mit einem Editor wie beispielsweise Adobe Acrobat Pro oder dem Spezialprogramm PDFix Desktop Pro für PDF/UA anschließend komplett neu getaggt werden muss. Wenn die Gestaltung der Präsentation einer barrierefreien Vorlage folgt, lassen sich die notwendigen Tags per Template programmatisch erkennen, sodass der Prozess komplett automatisiert werden kann.

### Export mit Grackle Slides

Das Add-on von Grackle ergänzt Google Slides um Funktionen für Barrierefreiheit und erzeugt deutlich bessere PDF-Ergebnisse. Es erstellt einen strukturierten Tag-Baum, berücksichtigt Alternativtexte, Tabellen und die Lesereihenfolge.

Bei komplexen Präsentationen bleiben jedoch Einschränkungen bestehen, beispielsweise bei verschachtelten Strukturen. Aus diesen Grund ist eine Nachbearbeitung mit einem PDF-Editor auf Basis der Prüfergebnisse erforderlich.

### Workaround über Microsoft PowerPoint

Die Präsentation wird als .pptx exportiert, in PowerPoint geöffnet, mit der integrierten Barrierefreiheitsprüfung geprüft und anschließend als PDF ausgegeben. Dieser Weg liefert in der Regel die qualitativ beste Ausgangsbasis, da PowerPoint einen deutlich ausgereifteren PDF-Export besitzt. Dennoch gilt auch in diesem Fall, dass dieses PDF mit einem Validator für PDF/UA geprüft und nachbearbeitet werden muss.

## Verweise

### Interne Links

- Microsoft PowerPoint

### Externe Links in diesem Artikel

- Grackle Workspace (Grackle)
- PDFix Solutions (PDFix)

### Weiterführende Links

- Techniques for Accessible PDF (Techniques for Accessible PDF)
- Tools for Accessible PDF (Tools for Accessible PDF)

# Anhang: Verfassende Personen

Online betrachten

Bei der Erstellung dieser Handreichung haben mitgewirkt:

- Sven Bittenbinder, Universität Siegen, Wirtschaftsinformatik, insb. IT für die alternde Gesellschaft, IAAP-DACH AK Barrierefreiheit in der Bildung, BFIT-Bund AG12 Barrierefreie Hochschule
- Sarah Bohnert, Deutsches Zentrum für barrierefreies Lesen (dzb lesen), IAAP-DACH AK Barrierefreiheit in der Bildung, BFIT-Bund AG12 Barrierefreie Hochschule
- Christina Broo, Universität Bremen, IAAP-DACH AK Barrierefreiheit in der Bildung, BFIT-Bund AG12 Barrierefreie Hochschule
- Uwe Dombeck, Barrierefreie-Dokumente-erstellen.de, IAAP-DACH AK Barrierefreiheit in der Bildung, BFIT-Bund AG12 Barrierefreie Hochschule
- Heiko Folkerts, Gemeinsamer Fachausschuss für IT (FIT) im Deutschen Blinden- und Sehbehindertenverband (DBSV)
- Jule Günter, Universität Bielefeld, IAAP-DACH AK Barrierefreiheit in der Bildung, BFIT-Bund AG12 Barrierefreie Hochschule
- Michael Karbe, Actino Software GmbH, German Chair PDF Association
- Finnja Lüttmann, TU Dortmund, IAAP-DACH AK Barrierefreiheit in der Bildung, BFIT-Bund AG12 Barrierefreie Hochschule
- Erdmuthe Meyer zu Bexten, Hessische Landesbeauftragte für Barrierefreie IT, IAAP-DACH AK Barrierefreiheit in der Bildung, BFIT-Bund AG12 Barrierefreie Hochschule
- Birgit Peböck, barrierefrei PDF OG, IAAP-DACH AK Barrierefreiheit in der Bildung
- Alexander Pfingstl, BFIT-Bund, IAAP-DACH AK Barrierefreiheit in der Bildung, BFIT-Bund AG12 Barrierefreie Hochschule
- Thorsten Schwarz, Karlsruher Institut für Technologie (KIT) Zentrum für digitale Barrierefreiheit und Assistive Technologien - ACCESS@KIT,IAAP-DACH AK Barrierefreiheit in der Bildung, BFIT-Bund AG12 Barrierefreie Hochschule
- Jens Voegler, TU Dresden, AG Services Behinderung und Studium IAAP-DACH AK Barrierefreiheit in der Bildung, BFIT-Bund AG12 Barrierefreie Hochschule
- Gerhard Weber, TU Dresden, AG Services Behinderung und Studium IAAP-DACH AK Barrierefreiheit in der Bildung, BFIT-Bund AG12 Barrierefreie Hochschule
- Nadja Willy, Universität Bremen, IAAP-DACH AK Barrierefreiheit in der Bildung, BFIT-Bund AG12 Barrierefreie Hochschule
- Anja Ziemer, Hamburg, IAAP-DACH AK Barrierefreiheit in der Bildung, BFIT-Bund AG12 Barrierefreie Hochschule
- Gottfried Zimmermann, Kompetenzzentrum für Digitale Barrierefreiheit an der Hochschule der Medien Stuttgart, IAAP-DACH AK Barrierefreiheit in der Bildung, BFIT-Bund AG12 Barrierefreie Hochschule
- Francis Zinke, Universität Potsdam, IAAP-DACH AK Barrierefreiheit in der Bildung, BFIT-Bund AG12 Barrierefreie Hochschule

# Anhang: Lizenzinformationen für diese Handreichung

Online betrachten

Diese Handreichung wird unter der Lizenz CC-BY-SA 4.0 veröffentlicht. Sie können Sie bearbeiten und unter Namensnennung und mit gleicher Lizenz weiterverbreiten. Wenn Sie Teile davon verändern, müssen Sie das entsprechend kennzeichnen.

# Informationen zu diesem Dokument

Diese Handreichung hat die Version 3.0 und wurde am 18.10.2026 veröffentlicht.

## Allgemeine Informationspflichten gemäß § 5 Telemediengesetz und § 55 Rundfunkstaatsvertrag

Die Deutsche Rentenversicherung Knappschaft-Bahn-See ist eine rechtsfähige Körperschaft des öffentlichen Rechts mit Selbstverwaltung und besitzt Dienstherrnfähigkeit (§ 29 SGB IV in Verbindung mit § 143 Abs. 1 SGB VI).

Dieses Impressum gilt für dieses Dokument der Arbeitsgruppen des Ausschusses für barrierefreie Informationstechnik nach § 5 BITV 2.0. Die Arbeitsgruppen werden von der Überwachungsstelle des Bundes für Barrierefreiheit von Informationstechnik organisiert.

## Herausgeber

Deutsche Rentenversicherung Knappschaft-Bahn-See
Pieperstraße 14 - 28
44789 Bochum
Tel. 0234 304 - 0
Fax 0234 304 - 66050
E-Mail an die Zentrale der KBS: zentrale@kbs.de

Umsatzsteuer-Identifikationsnummer: DE 124089627

Dieses Dokument wird herausgegeben von der Deutschen Rentenversicherung Knappschaft-Bahn-See, vertreten durch die Geschäftsführung, Dr. Rainer Wilhelm.

## Zuständige Fachaufsichtsbehörde für die Überwachungsstelle des Bundes für Barrierefreiheit von Informationstechnik

Bundesministerium für Arbeit und Soziales
Wilhelmstraße 49
10117 Berlin

## Nutzungsbedingungen

Die Inhalte dieser Handreichung werden mit größtmöglicher Sorgfalt verfasst. Unser Anspruch ist es, richtige, vollständige und aktuelle Inhalte bereitzustellen. Wir übernehmen dennoch keine Gewähr für versehentlich gemachte falsche Angaben.

Diese Handreichung enthält Verknüpfungen zu Webauftritten Dritter (“externe Links”). Wir haben bei der erstmaligen Verknüpfung zu externen Links die fremden Inhalte daraufhin überprüft, ob etwaige Rechtsverstöße bestehen. Zu diesem Zeitpunkt haben wir keine Rechtsverstöße vorgefunden. Wir haben jedoch weder Einfluss auf die aktuelle und zukünftige Gestaltung der verknüpften Seiten noch auf deren Inhalte oder Angebote. Sollten uns Rechtsverstöße bekannt werden, löschen wir die betreffenden externen Links unverzüglich. Bitte weisen Sie uns gegebenenfalls darauf hin.