GANZ EINFACH

# macOS für Power-User



Danilo Sieren

Danilo Sieren

# macOS für Power User Ganz Einfach

*Terminal Automation und Systemverwaltung verständlich und praktisch*

Erweiterte Neuauflage 2026

Verstehen · Anwenden · Überprüfen

# Willkommen in deiner verlässlichen Mac Umgebung

Ein leistungsfähiger Mac entsteht durch verständliche Arbeitsabläufe. Dieses Buch zeigt dir, wie du Dateien zuverlässig verwaltest, Werkzeuge passend auswählst und Änderungen so vorbereitest, dass du ihre Wirkung beurteilen kannst. Du arbeitest mit Finder und Systemeinstellungen ebenso wie mit Terminal, Kurzbefehlen und Git. Jedes Fachkapitel verbindet die Grundlagen mit einer praktischen Aufgabe.

Die Neuauflage richtet sich an Menschen, die ihren Mac bereits im Alltag nutzen und Zusammenhänge genauer verstehen möchten. Du brauchst keine Programmiererfahrung. Für die späteren Kapitel solltest du jedoch bereit sein, Befehle genau zu lesen und mit kleinen Beispielen zu üben. Wer Terminal und Git schon kennt, kann die Grundlagen als Nachschlagewerk verwenden und direkt zu den jeweiligen Projekten wechseln.

## Orientierung und Versionsstand

Redaktionsstand ist der 31. Oktober 2026. Als aktuelle Hauptlinie dient macOS Golden Gate 27.2. Apple nennt für macOS 27 Macs mit Apple-Chip; Hinweise zu Intel-Macs betreffen ausdrücklich ältere kompatible Systemversionen. Prüfe vor einem Upgrade immer die Modellliste sowie die Freigaben deiner unverzichtbaren Programme. [A01, A02, A82]

Funktionsumfang und Menübezeichnungen hängen teilweise von Modell, Sprache, Region, Benutzerrechten und verwalteten Vorgaben ab. Wenn ein beschriebener Punkt fehlt, suche zunächst in den Systemeinstellungen nach seiner Bezeichnung. Die Quellen am Ende des Buches führen zu den offiziellen Anleitungen und zu den Dokumentationen der genannten Projekte.

Die Kapitel bauen in Gruppen aufeinander auf. Kapitel 1 bis 3 schaffen Orientierung im System und in den Dateien. Kapitel 4 bis 7 behandeln Kommandozeile, Softwareverwaltung, Automation und Netzwerk. Kapitel 8 und 9 verbinden Datensicherheit mit nachvollziehbarer Leistungsanalyse. Kapitel 10 bis 13 erweitern die Arbeitsumgebung. Kapitel 14 hilft bei Störungen, Kapitel 15 bei einem sachkundigen KI-Einsatz.

## Mit den Beispielen arbeiten

Führe ein kleines Lernprotokoll mit Datum, Systemversion, Ziel, Beobachtung und Rückweg. Kopiere zunächst keine Befehlsfolge vollständig, sondern prüfe jeden Schritt. Die Codekästen enthalten keine vorangestellten Promptzeichen. Ein Dollarzeichen innerhalb eines Befehls gehört zur Shell-Syntax. Platzhalter wie deinname und example.com ersetzt du nur dort, wo dies im Text erklärt wird.

Zusammengehörige Befehle setzen mitunter dieselbe Terminal-Sitzung voraus, weil eine Variable dort definiert wurde. Eine neue Registerkarte kennt diese Variable nicht automatisch. Lies deshalb die Vorbereitung eines Projekts auch dann, wenn du später nur einen einzelnen Befehl nachschlagen möchtest. Zeilenumbrüche im Code sind Teil des Beispiels; automatisch umgebrochene lange Zeilen werden nicht durch zusätzliche Return-Tasten ergänzt.

Verwende für Versuche Kopien und eindeutig benannte Lernordner. Eine erfolgreich ausgeführte Übung erteilt keine allgemeine Freigabe für die gleiche Operation auf einer Mediathek, einem Cloudordner oder einer produktiven Datenbank. Diese Daten besitzen zusätzliche Zusammenhänge, die eine einfache Dateiansicht nicht zeigt.

## Ergebnisse prüfen und Grenzen erkennen

Nach jeder Änderung folgt eine passende Kontrolle. Bei einer Sicherung ist das eine Wiederherstellung, bei einer Automation das erwartete Protokoll, bei einer Übertragung ein nachvollziehbarer Vergleich. Eine fehlende Fehlermeldung genügt nicht. Halte fest, was du tatsächlich beobachtet hast, und trenne es von einer Vermutung.

Nicht jede Hardwarekombination, Erweiterung und regionale Funktion lässt sich allgemein zusichern. Rechenzeiten, Akkulaufzeiten und Wiederherstellungserfolge sind keine festen Eigenschaften einer Anleitung. Wo ein Ergebnis von solchen Bedingungen abhängt, beschreibt das Buch die Voraussetzungen und einen sinnvollen Prüfweg. Menübilder werden durch eigene schematische Abbildungen ergänzt, damit technische Zusammenhänge auch bei veränderter Oberfläche verständlich bleiben.

# Inhalt

Willkommen in deiner verlässlichen Mac Umgebung ..... 2
1 Das Fundament von macOS verstehen ..... 6
2 Systemeinstellungen für einen verlässlichen Arbeitsplatz ..... 17
3 Dateien mit dem Finder zuverlässig verwalten ..... 27
4 Mit der Zsh sicher und verständlich arbeiten ..... 38
5 Homebrew und Entwicklungswerkzeuge kontrolliert verwalten ..... 48
6 Automationen mit Kurzbefehlen AppleScript und launchd ..... 58
7 Netzwerk SSH und Datentransfer nachvollziehbar nutzen ..... 68
8 Daten verschlüsseln sichern und wiederherstellen ..... 78
9 Leistung Speicher und Hintergrundprozesse untersuchen ..... 89
10 Zusätzliche Werkzeuge nach ihrem Nutzen auswählen ..... 98
11 Änderungen mit Git nachvollziehbar verwalten ..... 107
12 Virtuelle Maschinen und Container als Testumgebung ..... 117
13 Apple Geräte und iCloud bewusst verbinden ..... 127
14 Störungen eingrenzen und Wiederherstellung vorbereiten ..... 137
15 KI auf dem Mac sachkundig einsetzen ..... 147
Abschlussprojekt Ein nachvollziehbares Dateiarchiv aufbauen ..... 158
Anhang Werkzeuge und Begriffe im Zusammenhang ..... 166
Abschnittsverzeichnis ..... 175
Stichwortverzeichnis ..... 183
Quellen und weiterführende Dokumentation ..... 186
Impressum ..... 195

# 1 Das Fundament von macOS verstehen

Ein Mac bietet mehrere Wege zur gleichen Aufgabe. Du kannst eine Datei im Finder verschieben, sie mit einem Shell-Befehl kopieren oder einen Kurzbefehl dafür erstellen. Ein verlässlicher Arbeitsablauf entsteht, wenn du die passende Ebene wählst und ihre Grenzen kennst. Dafür brauchst du weder einen ständig geöffneten Terminalbildschirm noch möglichst viele Zusatzprogramme. Du brauchst ein verständliches Modell von System, Benutzerkonto, Anwendungen und Daten.

Dieses Kapitel baut dieses Modell auf. Es erklärt, warum Unix-Werkzeuge unter macOS funktionieren, welche Aufgaben Apple Silicon und Rosetta übernehmen und weshalb weder Administratorrechte noch ein APFS-Snapshot sämtliche Probleme lösen. Das Ziel ist eine Arbeitsumgebung, deren Verhalten du erklären und überprüfen kannst.

## 1.1 Version und Hardware zuerst bestimmen

Öffne das Apple-Menü und wähle „Über diesen Mac“. Notiere den Modellnamen, den Chip beziehungsweise Intel-Prozessor, den Arbeitsspeicher und die macOS-Version. Diese Angaben entscheiden darüber, ob eine Anleitung passt. Ein Menüpfad für macOS 27 muss auf einem älteren Intel-Mac mit einer anderen Systemversion nicht identisch aussehen.

Zum Redaktionsstand ist macOS Golden Gate 27.2 die aktuelle Hauptlinie. Apple führt dafür Macs mit Apple-Chip auf; Intel-Macs können diese Hauptversion nicht installieren. Für ältere Systeme veröffentlicht Apple weiterhin gesonderte Aktualisierungen, deren Verfügbarkeit du anhand der Sicherheitsveröffentlichungen und der Kompatibilität deines Modells prüfst. Eine noch verfügbare Aktualisierung ist keine

Zusage für einen bestimmten zukünftigen Unterstützungszeitraum. [A01, A02]

Eine Inventarliste enthält auch die Anwendungen, auf die du angewiesen bist: Audio-Plug-ins, Druckertreiber, VPN-Software, Sicherungsprogramme und Erweiterungen. Eine kompatible Hauptanwendung genügt nicht, wenn ein benötigtes Plug-in nicht mehr funktioniert. Vor einem großen Upgrade prüfst du deshalb eine vollständige Arbeitskette, etwa Dokument öffnen, bearbeiten, exportieren und drucken.

## 1.2 Darwin und der XNU Kernel

Darwin bezeichnet wesentliche offene Grundlagen von macOS. Dazu gehören unter anderem der XNU-Kernel und Unix-nahe Systemkomponenten. XNU verbindet Mach-Technik mit BSD-Bestandteilen und weiteren Komponenten. Darwin ist damit weder ein einzelner Kommandozeilenbefehl noch die vollständige grafische Mac-Oberfläche. Apple veröffentlicht Quelltexte verschiedener Systembestandteile auf seiner Open-Source-Plattform. [A03]

Der Kernel vermittelt zwischen Prozessen und Hardware, verwaltet Speicher und stellt grundlegende Systemdienste bereit. Dein Texteditor arbeitet als eigener Prozess. Wenn er nicht mehr reagiert, bedeutet das zunächst nicht, dass der Kernel ausgefallen ist. Umgekehrt können Treiber-, Hardware- und Systemfehler den gesamten Rechner beeinträchtigen. Aus einer Architektur mit getrennten Prozessen folgt keine Garantie, dass ein Mac niemals abstürzt oder nach Wochen ohne Neustart immer gleich schnell bleibt.

Für die Fehlersuche ist die Trennung hilfreich: Betrifft ein Problem eine Datei, eine App, ein Benutzerkonto oder den gesamten Rechner? Diese Frage verhindert, dass du wegen eines defekten App-Dokuments unnötig das Betriebssystem neu installierst.

## 1.3 Oberfläche und Kommandozeile zusammendenken

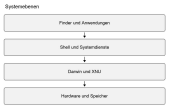
Finder, Terminal und Systemeinstellungen sind unterschiedliche Zugänge. Im Terminal startet eine Shell Programme und wertet ihre Argumente aus. Du sprichst damit nicht unmittelbar mit dem Kernel. Auch ein einfacher Listenbefehl verwendet Systemdienste und muss die geltenden Zugriffsregeln beachten.

Grafische Frameworks stellen Funktionen bereit, die Anwendungen verwenden können. Metal ist beispielsweise eine Schnittstelle für Grafik- und Rechenaufgaben auf der GPU. Daraus folgt nicht, dass jede App automatisch besonders schnell arbeitet oder eine feste höchste Priorität erhält. Das Ergebnis hängt von Programm, Daten und Hardware ab.

Wähle den Zugang nach der Aufgabe. Für zehn einzelne Fotos ist eine visuelle Kontrolle im Finder oft sinnvoll. Für eine wiederkehrende Auswertung vieler Textdateien kann ein Skript nachvollziehbarer sein. Eine gute Kombination besteht darin, das Ergebnis zunächst im Terminal aufzulisten und anschließend mit „open“ im Finder oder in einer passenden App zu prüfen.

**Tabelle 1.1 – Die passende Ebene wählen**

| Aufgabe | Geeigneter Einstieg | Kontrolle |
|---|---|---|
| Dokument finden | Finder oder Spotlight | Pfad und Vorschau |
| Viele Textdateien auswerten | Shell oder Skript | Stichprobe und Fehlerausgabe |
| Systemoption ändern | Systemeinstellungen | Vorherigen Wert notieren |
| App-Funktion automatisieren | Kurzbefehle oder AppleScript | Ergebnis in der App |
| Startproblem untersuchen | Wiederherstellung | Modellgerechte Anleitung |



## 1.4 Apple Silicon und gemeinsame Ressourcen

Macs mit Apple-Chip verbinden mehrere Komponenten in einem System auf einem Chip. CPU, GPU und weitere Einheiten teilen sich auf unterstützter Hardware einen gemeinsamen Speicherbereich. Dieser Unified Memory vermeidet bestimmte Kopierwege, macht den Arbeitsspeicher aber nicht unbegrenzt. Eine große virtuelle Maschine, ein Videoschnittprojekt und ein lokales Sprachmodell konkurrieren weiterhin um endliche Ressourcen.

Performance- und Effizienzkerne sind Aufgaben des System-Schedulers. Du solltest nicht aus einer bestimmten Prozessbezeichnung ableiten, auf welchem Kerntyp sie immer ausgeführt wird. Auch bei virtuellen Maschinen lässt sich eine Anzahl virtueller CPUs nicht pauschal in reservierte physische Effizienzkerne übersetzen.

Praktisch beobachtest du Speicherdruck, Laufzeiten und Reaktionsfähigkeit. Ein Arbeitsablauf mit 16 GB kann für Textbearbeitung ausreichen und für mehrere große Gäste oder Modelle zu knapp sein. Statt eine universelle Mindestmenge zu behaupten, misst du deinen tatsächlichen Lastfall und lässt Reserven für macOS und andere Apps.

## 1.5 Rosetta ist ein Übergangswerkzeug

Rosetta ermöglicht auf unterstützten macOS-Versionen die Ausführung vieler Intel-basierter Mac-Apps auf Apple Silicon. Es ist keine Windows-Umgebung und ersetzt keine beliebigen alten Treiber. Eine Universal-App enthält passende Programmanteile für mehrere Architekturen; eine reine Apple-Chip-App benötigt für ihre native Ausführung keine Intel-Übersetzung.

Apple beschreibt Rosetta für macOS 27 und ältere Versionen. Ab macOS 28 ist nur noch eine eingeschränkte Funktionalität für

bestimmte ältere Spiele vorgesehen. Behandle das als Migrationsaufgabe: Erfasse Intel-Abhängigkeiten und suche nach nativen Fassungen, statt die weitere Ausführbarkeit vorauszusetzen. Auch Plug-ins und Kommandozeilenprogramme gehören in diese Bestandsaufnahme. [A04]

Wenn eine Universal-App ausdrücklich mit Rosetta geöffnet wird, kann dies der Kompatibilität mit einem älteren Zusatzmodul dienen. Dokumentiere den Grund. Ein unbemerkt als Intel gestartetes Terminal kann sonst eine zweite Werkzeugwelt mit anderen Installationspfaden begünstigen. Kapitel 5 behandelt diese Pfadfragen ausführlich.

## 1.6 APFS vom Datenträger bis zur Datei

Ein physischer Datenträger kann Partitionen enthalten. Ein APFS-Container stellt Speicher für darin angelegte APFS-Volumes bereit. Diese können freien Platz gemeinsam nutzen; optionale Reserven und Quoten beeinflussen die Verteilung. Ein zusätzliches Volume ist deshalb keine zweite unabhängige Festplatte und schützt nicht vor dem Ausfall desselben Speichermediums.

APFS unterstützt Copy-on-Write und Dateiklone. Wenn ein passender Kopiervorgang einen Klon innerhalb des geeigneten APFS-Kontexts erstellt, können Dateien zunächst Datenblöcke gemeinsam verwenden. Spätere Änderungen benötigen zusätzlichen Platz. Nicht jede Kopie zwischen beliebigen Laufwerken ist ein Klon, und Metadaten brauchen ebenfalls Speicher. Die pauschale Aussage „eine Kopie kostet null Byte“ wäre falsch. [A05]

Für Projekte bedeutet das: Zwei sichtbare Dateien können unterschiedliche logische Größen und einen anderen tatsächlichen Speicherverbrauch haben. Ein Klon kann dir ein schnelles Arbeitsduplikat liefern. Er ist aber keine unabhängige

Sicherung, solange beide Fassungen auf demselben Datenträger liegen.

## 1.7 Snapshots und Sicherungen unterscheiden

Ein APFS-Snapshot hält einen Volumezustand fest. Durch spätere Änderungen bleiben dafür ältere Blöcke belegt. Ein Snapshot kann bei der Erstellung wenig zusätzlichen Platz benötigen und danach erheblich wachsen. Seine Verfügbarkeit hängt vom jeweiligen Volume und der Verwaltung durch System oder Sicherungsprogramm ab.

Ein lokaler Snapshot auf der internen SSD verschwindet mit dieser SSD. Er hilft außerdem nur, wenn ein geeigneter Zustand tatsächlich noch vorhanden ist. Time Machine kann lokale Schnappschüsse zum Wiederherstellen verwenden; das ist keine universelle Rückkehr des gesamten Macs einschließlich beliebiger Apps, externer Daten und Firmware auf Knopfdruck. [A06]

Vermeide deshalb Experimente nach dem Muster „ein Snapshot wird mich schon retten“. Vor einer wichtigen Änderung prüfst du eine unabhängige Sicherung und den Wiederherstellungsweg. Kapitel 8 trennt Dateiwiederherstellung, Systemneuinstallation, Migration und bootfähige Kopien. Diese Aufgaben haben unterschiedliche Voraussetzungen und unterschiedliche Zeitbedarfe.

## 1.8 Systemvolume und Datenvolume

Aktuelle macOS-Versionen trennen geschützte Systeminhalte und veränderliche Daten. Das Signed System Volume schützt Systeminhalte zusätzlich kryptografisch. Die im Finder sichtbare Ordnerstruktur verdeckt einen Teil dieser technischen Trennung, damit Anwendungen mit gewohnten Pfaden arbeiten können. [A07]

Daraus darfst du nicht schließen, dass alle Dateien unter einem ähnlich klingenden Ordner denselben Schutz genießen. Deine Benutzerdateien und viele installierte Programme sind veränderlich. Schadsoftware muss den Systemkernel nicht austauschen, um lesbare persönliche Daten zu stehlen oder Dateien in deinem Konto zu beschädigen.

SIP, sichere Startfunktionen, App-Signaturen und Datenschutzfreigaben ergänzen einander. Keiner dieser Mechanismen ist ein Ersatz für Datensicherung oder eine sorgfältige Softwareauswahl. Für die Übungen dieses Buches ist es nicht nötig, das versiegelte Systemvolume zu verändern oder die Systemintegrität pauschal abzuschalten.

## 1.9 Benutzerkonten und Administratorrechte

Ein Standardkonto darf viele alltägliche Aufgaben erledigen, aber nicht jede systemweite Änderung vornehmen. Ein Administratorkonto kann bestimmte Änderungen bestätigen. Im Terminal bleibt auch ein Administrator zunächst ein normaler Benutzerprozess; er arbeitet nicht automatisch dauerhaft als root.

„sudo“ führt einen ausdrücklich angegebenen Befehl mit anderen Rechten aus, soweit die Sicherheitsregeln es erlauben. Es hebt weder alle Datenschutzgrenzen noch SIP beliebig auf. Ein „Zugriff verweigert“ ist deshalb eine Diagnoseinformation und keine Aufforderung, wahllos „sudo“ vorzusetzen. Ursache können Dateirechte, eine gesperrte Datei, ein schreibgeschütztes Volume oder eine Datenschutzfreigabe sein.

Trenne persönliche Konten, statt ein gemeinsames Administratorkonto für mehrere Menschen zu verwenden. Dadurch bleiben Dateien, Schlüsselbünde und Einstellungen zuordenbar. Auf verwalteten Macs können zusätzliche Richtlinien gelten. Versuche nicht, diese durch private Änderungen an Systemdateien zu umgehen.

## 1.10 Datenschutzfreigaben gezielt verstehen

macOS schützt unter anderem Kamera, Mikrofon und bestimmte Datenbereiche über Einwilligungen. Die sichtbaren Kategorien findest du unter „Systemeinstellungen > Datenschutz & Sicherheit“. Eine App kann etwa eine Automationsfreigabe zum Steuern einer anderen App benötigen, ohne deshalb Festplattenvollzugriff zu brauchen.

Bedienungshilfen erlauben andere Eingriffe als Bildschirmaufnahme oder Dateizugriff. Prüfe bei einer fehlgeschlagenen Automation zunächst, welche Anwendung tatsächlich läuft und welche Funktion abgewiesen wird. Ein manuell im Terminal gestartetes Skript und derselbe Text als Hintergrundauftrag besitzen nicht zwangsläufig denselben Zugriffskontext.

Erteile nur die für einen nachvollziehbaren Zweck benötigte Freigabe. Entfernst du ein Werkzeug später, kontrolliere die noch vorhandenen Einträge. Eine Liste alter Freigaben ist nicht automatisch gefährlich, aber eine regelmäßig überprüfte Liste lässt sich besser erklären und pflegen. Die passende Kategorie ist Teil der Projektdokumentation.

## 1.11 Metadaten und Dateiinhalte

Eine Datei hat mehr Eigenschaften als ihren Namen. Größe, Datumsangaben, Besitzer, Zugriffsrechte und erweiterte Attribute unterstützen Suche und Verarbeitung. Finder-Tags gehören zu dieser zusätzlichen Information. Beim Austausch mit anderen Dateisystemen oder Archivformaten können Eigenschaften verloren gehen oder in Zusatzdateien gespeichert werden.

Ein umbenanntes Dokument erhält dadurch keinen anderen inneren Dateityp. Ein Änderungsdatum ist nicht automatisch das Aufnahmedatum eines Fotos. Ein aus dem Internet geladener

Inhalt kann ein Quarantäneattribut tragen; dessen Entfernung macht den Inhalt nicht vertrauenswürdig. Für automatisierte Entscheidungen musst du die Bedeutung einer Eigenschaft kennen.

Wenn Metadaten wichtig sind, teste einen vollständigen Rundweg: kopieren, archivieren, auf dem Ziel auspacken und die benötigten Eigenschaften kontrollieren. Ein erfolgreicher Kopierdialog bestätigt noch nicht, dass alle Tags, Berechtigungen und App-spezifischen Informationen unverändert geblieben sind.

## 1.12 Praxisprojekt Deinen Mac nachvollziehbar erfassen

Erstelle eine neue Notiz mit dem Titel „Systemstand“. Verwende keine Passwörter, Wiederherstellungsschlüssel oder Seriennummern in einer öffentlich geteilten Fassung. Die Notiz soll die Auswahl passender Anleitungen erleichtern und Veränderungen nachvollziehbar machen.

### Die Ausgangsdaten aufschreiben

Trage Modell, Architektur, macOS-Version und Speicherausstattung ein. Ergänze das Datum der Erfassung. Notiere, ob es sich um einen persönlichen oder verwalteten Mac handelt. Öffne drei wichtige Anwendungen und erfasse ihre Versionsangaben. Wenn du ein Audio- oder Video-Plug-in benötigst, füge es als eigenen Eintrag hinzu.

Kennzeichne jede Intel-Abhängigkeit als „native Alternative vorhanden“, „Herstellerprüfung offen“ oder „Migration geplant“. Erfinde keine Kompatibilitätsaussage aus einem erfolgreich gestarteten Programm. Öffne stattdessen ein unkritisches Beispieldokument, bearbeite es und exportiere das Ergebnis. Die getesteten Schritte schreibst du daneben.

## Daten und Sicherungen zuordnen

Wähle einen gewöhnlichen Projektordner. Öffne im Finder „Informationen“ und stelle fest, auf welchem Volume er liegt. Prüfe, ob er lokal verfügbar ist oder über einen Cloud-Dienst verwaltet wird. Ein sichtbarer Dateiname ist kein Nachweis, dass sein Inhalt ohne Internet gelesen werden kann.

Notiere den letzten tatsächlich abgeschlossenen Sicherungslauf und den Speicherort der unabhängigen Kopie. Wenn du das noch nicht feststellen kannst, schreibe „offen“. Ein ehrlicher offener Punkt ist hilfreicher als ein Haken hinter einer ungeprüften Annahme. Die praktische Wiederherstellungsprobe folgt in Kapitel 8.

**Tabelle 1.2 – Das Systemprotokoll**

| Bereich | Einzutragende Information | Prüfschritt |
|---|---|---|
| Betriebssystem | Version und Datum | Über diesen Mac |
| Architektur | Apple-Chip oder Intel | Modellangabe |
| Hauptanwendung | Version und Zusatzmodule | Beispieldatei exportieren |
| Projektordner | Volume und Cloud-Status | Informationen und Datei öffnen |
| Sicherung | Datum und unabhängiges Ziel | Letzten Abschluss kontrollieren |
| Offene Frage | Konkreter nächster Schritt | Quelle oder Test benennen |

## Eine kleine Änderung dokumentieren

Verändere eine harmlose Darstellungseinstellung, etwa die Anzeige der Finder-Pfadleiste. Notiere den vorherigen Zustand, die Änderung und das Ergebnis. Stelle danach den gewünschten Zustand her. Du übst damit den Ablauf, den spätere Kapitel für

komplexere Eingriffe verwenden: Ausgangslage feststellen, gezielt ändern, Wirkung prüfen und Rückweg kennen.

Lass eine zweite Person anhand deiner Notiz erkennen, welche Anleitung zu deinem Mac passen würde. Sie muss keine Geheimnisse sehen und keine Systemänderung durchführen. Wenn ihr aus derselben Notiz unterschiedliche Architektur- oder Versionsannahmen ableitet, fehlen Angaben. Ergänze sie, bis die Ausgangslage eindeutig ist.

Deine fertige Übersicht nennt System, Datenorte und den letzten überprüften Sicherungsstand. Später ergänzt du Werkzeuge, Automationen und Wiederherstellungswege.

# 2 Systemeinstellungen für einen verlässlichen Arbeitsplatz

Ein gut eingerichteter Mac passt zu deinen Aufgaben. Er muss nicht jede Animation verlieren und nicht jeden Hintergrunddienst abschalten. Manche Verzögerung entsteht durch eine unpassende Fensteranordnung, eine andere durch einen Netzwerkspeicher oder ein tatsächlich ausgelastetes Programm. Dieses Kapitel behandelt Einstellungen, die du bewusst auswählen und wieder zurückstellen kannst.

Arbeite jeweils mit einer Veränderung. Notiere bei einem Vergleich dieselbe App, dieselbe Datei und eine ähnliche Situation. „Fühlt sich schneller an“ kann ein nützlicher erster Eindruck sein, ersetzt aber keine überprüfbare Beobachtung. Wenn du nach zehn gleichzeitigen Änderungen eine Verbesserung siehst, kennst du deren Ursache noch nicht.

## 2.1 Einstellungen finden und vergleichen

Öffne die Systemeinstellungen über das Apple-Menü. Nutze das Suchfeld, wenn ein Bereich anders angeordnet ist als in einer älteren Anleitung. Achte darauf, ob du eine systemweite Option, eine App-Einstellung oder eine für dein Benutzerkonto geltende Eigenschaft veränderst.

Eine ausgegraute Option kann einen bestimmten Gerätetyp voraussetzen oder durch eine Verwaltungsvorgabe gesperrt sein. Die Abwesenheit eines Schalters beweist keinen Systemfehler. Batterieoptionen unterscheiden sich beispielsweise zwischen einem MacBook und einem Desktop-Mac; Referenzmodi gibt es nur für passende Displays.

Lege für wichtige Einstellungen eine kurze Liste an. Ein Bildschirmfoto erleichtert den Vergleich, sofern keine privaten Daten sichtbar sind. Der entscheidende Inhalt ist der tatsächliche Wert und sein Zweck: „Bildschirm nach Inaktivität sperren“ ist aussagekräftiger als „Sicherheit optimiert“.

## 2.2 Tastaturkurzbefehle sinnvoll vergeben

Unter „Systemeinstellungen > Tastatur > Tastaturkurzbefehle“ findest du die verfügbaren Bereiche. App-Kurzbefehle beziehen sich auf Menübefehle, deren Bezeichnung du exakt übernehmen musst. Ein anders gesetztes Auslassungszeichen oder ein nicht vorhandener Menüeintrag kann verhindern, dass ein Kürzel greift. [A08]

Prüfe zunächst bestehende Belegungen. Command-Leertaste wird typischerweise für Spotlight verwendet. Andere Werkzeuge können dieselbe Kombination beanspruchen. Ändere nicht gleichzeitig das Systemkürzel und die Belegung in einem Launcher, sonst wird ein Konflikt schwerer zuzuordnen.

Für häufige Aktionen reichen wenige gut gewählte Kürzel. Vergib beispielsweise eines für einen tatsächlich vorhandenen Exportbefehl in deiner Hauptanwendung. Teste es mit einem unkritischen Dokument und kontrolliere das Ergebnis. Ein Kürzel, das einen Dialog öffnet, ist noch keine unbeaufsichtigte Exportautomation.

## 2.3 Trackpad und Bedienungshilfen

Die Trackpad-Einstellungen zeigen Gesten und deren Bedeutung. Probiere eine neue Geste mit mehreren Fenstern aus. Eine theoretisch kürzere Bewegung ist nicht automatisch angenehmer, wenn du sie häufig versehentlich auslöst. Das gilt besonders für Gesten, die mit App-eigenen Wischbewegungen konkurrieren.

Unter „Bedienungshilfen > Zeigersteuerung“ findest du je nach Eingabegerät zusätzliche Optionen, darunter Trackpad-Optionen

für das Bewegen von Objekten. Der Bildschirmzoom liegt im Bereich „Zoom“. Richte ihn so ein, dass Text lesbar bleibt und du die vergrößerte Ansicht sicher wieder verlassen kannst. [A09]

Bedienungshilfen sind normale Werkzeuge für unterschiedliche Arbeitsweisen. Eine Einstellung soll deine konkrete Bedienung erleichtern. Aus einer bestimmten Maus, Tastatur oder Geste lässt sich keine allgemeine gesundheitliche Wirkung ableiten. Entscheidend sind Erreichbarkeit, angemessene Bewegung und eine für dich brauchbare Darstellung.

## 2.4 Fensteranordnung ohne Zusatzprogramm

macOS bietet eine integrierte Kachelanordnung. Du kannst Fenster an Bildschirmränder ziehen, die grüne Fenstertaste verwenden oder im Menü „Fenster > Bewegen & Skalieren“ ein passendes Layout wählen. Unter „Systemeinstellungen > Schreibtisch & Dock“ lassen sich Optionen für das Anordnen von Fenstern ändern. [A10, A11]

Unterscheide Füllen, Kacheln und Vollbild. Ein gefülltes Fenster kann auf dem normalen Schreibtisch bleiben, während Vollbild einen eigenen Arbeitsbereich verwendet. Split View und gewöhnliche nebeneinander angeordnete Fenster sind deshalb nicht in jeder Hinsicht gleich. Prüfe insbesondere den Wechsel zwischen Apps und den Zugriff auf die Menüleiste.

Beginne mit zwei Arbeitsflächen: links das Dokument, rechts die Referenz. Teste danach eine kleinere Anzeige und einen externen Monitor. Ein Layout, das nur auf deinem größten Bildschirm funktioniert, ist keine verlässliche Lösung für unterwegs. Kapitel 10 zeigt, wann ein zusätzliches Werkzeug dennoch sinnvoll sein kann.

## 2.5 Mission Control und aktive Ecken

Mission Control zeigt Fenster und Spaces. Nutze wenige verständliche Arbeitsbereiche, etwa Schreiben und

Testumgebung. Zu viele ähnliche Spaces können den Wechsel eher erschweren. Ob Stage Manager dazu passt, hängt von deinen App-Gruppen und der Bildschirmfläche ab; es ist kein Pflichtschritt für einen professionellen Arbeitsplatz.

Aktive Ecken richtest du unter „Systemeinstellungen > Schreibtisch & Dock > Aktive Ecken“ ein. Du kannst eine Sondertaste mit einer Ecke kombinieren, um versehentliche Auslösungen zu vermeiden. Eine Ecke für den Sperrbildschirm ist beispielsweise gut prüfbar: Zeiger bewegen, gegebenenfalls Sondertaste halten, Sperre kontrollieren. [A12]

Notiere die Belegung, wenn mehrere Menschen den Mac benutzen. Eine unerwartete Sperre oder Schnellnotiz am Bildschirmrand ist sonst leicht mit einem App-Problem zu verwechseln. Die Zuordnung gilt für die Bedienung und ersetzt keine automatische Sperre nach Inaktivität.

## 2.6 Spotlight ist mehr als eine Dateisuche

Spotlight kann Apps, Dateien, Aktionen und je nach Einstellung Inhalte der Zwischenablage finden. In macOS 27 werden Suchkategorien und Datenschutzoptionen im eigenständigen Bereich „Spotlight“ verwaltet. Alte Anleitungen mit dem gemeinsamen Bereich „Siri & Spotlight“ passen deshalb nicht durchgehend zur aktuellen Oberfläche. [A13]

Die Zwischenablageansicht erreichst du innerhalb von Spotlight mit Command-4. Dort können kopierte vertrauliche Inhalte erscheinen. Über die Einstellungen steuerst du die Nutzung und den Aufbewahrungszeitraum des Verlaufs; du kannst ihn auch löschen. Ein Clipboard-Manager ist somit nicht mehr zwingend nötig, nur um mehrere kopierte Inhalte wiederzufinden. [A14]

Teste die Suche mit einer selbst erstellten Datei. Prüfe Suchumfang, Dateiname und Inhalt getrennt. Das Ausblenden einer Ergebniskategorie ist etwas anderes als der Ausschluss eines Ordners vom Index. Ein ausgeschlossener Projektordner