

# Technisches Schreiben

Ein Leitfaden von Lutz Prechelt

ist schwierig. [...] Es ist schwierig, weil es eine Reihe von Aktivitäten verlangt, die gemeinhin als Denken bezeichnet werden.  
Ernst Jacobi

In diesem Anhang geht es um das Schreiben technischer Dokumente, seien es nun Software-Entwurfsdokumente, Benutzerhandbücher, wissenschaftliche Artikel oder sonst etwas derartiges.

Fertigkeiten im technischen Schreiben werden oftmals nicht als ein wesentliches Qualifikationsmerkmal von Softwaretechnikern angesehen, obwohl die Wirklichkeit dies sehr nahelegen würde: hohe Kosten entstehen bei der Wartung von Software sehr oft dadurch, daß die interne Dokumentation zu undurchsichtig organisiert oder inhaltlich zu schwer verständlich ist. Da interne Dokumentation fast stets von den Softwaretechnikern anzufertigen ist, nimmt also das technische Schreiben sogar einen sehr hohen Stellenwert ein.

Ferner ist es natürlich auch für das Schreiben von Benutzerdokumentation sehr nützlich, auch wenn diese Aufgabe zunehmend Spezialisten übertragen wird. Und drittens sind ganz ähnliche Kenntnisse und Fertigkeiten auch für das Schreiben guter Memos und das Entwerfen guter Präsentationen vonnöten. Es lohnt sich also, sich mit diesem Thema auseinanderzusetzen. Dabei dürfte klar sein, daß ein rein theoretisches Lernen hier vollkommen unmöglich ist; die Devise heißt üben, üben, üben. Das Schreiben ist ein kreativer Akt und dementsprechend befriedigend; deshalb kann solches Üben eine Menge Spaß machen. Jeder, der selber gut schreiben kann, ist als Kritiker zu gebrauchen.

In diesem Kapitel werden die Begriffe Dokument und Text weitgehend synonym gebraucht und stehen für das Ergebnis des technischen Schreibens, das außer Text auch Bilder etc. umfassen kann.

## 1 Anforderungen

Was unterscheidet denn nun das technische Schreiben von, sagen wir mal, dem Schreiben eines Romans oder eines Zeitungsartikels? Gar nicht soviel wie man meinen könnte, wie wir sehen werden.

Für die Qualität eines technischen Textes gibt es eine einzige zentrale Maßlatte: die **effektive Verständlichkeit**. Die wichtigste Frage ist also „Wieviel von der erwünschten Aussage kommt beim Leser im Endeffekt tatsächlich an?“. Diese Verständlichkeit entsteht aus verschiedenen Beiträgen, die in den folgenden Abschnitten kurz diskutiert werden:

### 1.1 Inhalt

Zuallererst muß natürlich der Inhalt des Dokumentes selbst verständlich sein. Die allererste Voraussetzung dafür ist, daß der Autor das Thema des Textes wirklich verstanden hat; es ist aber durchaus nicht schlimm, wenn das erst im Verlauf des Schreibens eintritt. Die zweite Voraussetzung ist, daß die Verständlichkeit nicht nur irgendwie und im Prinzip gegeben ist, sondern tatsächlich – und zwar für das Zielpublikum, das den Text lesen sollte.

Damit kommen wir zum Problem von **Abdeckung und Detaillierungsgrad**: Wo muß ich mit dem Erklären anfangen, was darf ich voraussetzen (Abdeckung) und wie genau muß eine Erklärung sein (Detaillierungsgrad)? Die wichtigste Erkenntnis lautet: Das kann man nie ganz richtig machen. Für eine gute Auslegung muß man vor allem sein Publikum und dessen Vorwissen gut kennen. Ist das nicht erreichbar, muß man versuchen, möglichst viele Zielgruppen zugleich anzusprechen, indem man im Text klar kennzeichnet, welche Teile für welche Breite und Tiefe von Informationsbedürfnis gelesen werden müssen – und natürlich diese Teile überhaupt erstmal voneinander trennt.

Ein zusätzliches Problem ist häufig der **Platzmangel**, der zu überkurzen Erklärungen und Formulierungen verleitet. Hier gilt meist die Regel „Weniger ist mehr“: wenige gut erläuterte Fakten sind allemal nützlicher als viele unverständliche. Die Wahl von Abdeckung und Detaillierungsgrad ist Teil eines allgemeineren Problems, der Adäquatheit, das unten angesprochen wird.

## 1.2 Textorganisation

Eine Schlüsselstellung für die Verständlichkeit nimmt eine geschickte Organisation des Dokuments ein. Das Ziel besteht darin, eine schnelle Erfassung des Wesentlichen auch ohne komplettes Lesen des Dokuments zu ermöglichen. Es soll also die Möglichkeit zum effektiven „**Querlesen**“ geschaffen werden. Dies gilt nicht nur für das Dokument im ganzen, sondern ebenso für jeden einzelnen seiner Abschnitte.

Querlesen ist deshalb wichtig, weil ein technischer Text fast immer erheblich mehr Information enthält, als der überwiegende Teil der Leserschaft benötigt. Der Leitgedanke ist folgender: Das Dokument soll einen Service für den Leser erbringen, nämlich ihm Informationen zu übermitteln, die er benötigt. Dieser Service ist umso besser erbracht, je schneller der Leser oder die Leserin diese und nur diese Informationen aus dem Gesamtdokument extrahieren kann.

Das Hauptmittel einer guten Textorganisation ist eine **logische Struktur und Abfolge der Teile**. Logisch bedeutet dabei, daß es eine einfache Begründung oder Herleitung für gerade die gewählte Struktur gibt, die der Leser schnell nachvollziehen kann. Wenn der Leser ebendies auch tut, dann ist die Textorganisation erfolgreich. Das beste Mittel, den Leser zum Nachvollziehen der Struktur zu bringen, besteht darin, die Begründung oder Erläuterung der Struktur in der Einleitung des Textes oder Textabschnitts explizit in knapper Form aufzuschreiben.

Weitere **technische Mittel** zu einer guten Textorganisation sind: Treffende Überschriften, Inhaltsverzeichnisse, Schlagwortindex und andere Verzeichnisse, sowie Querverweise.

Im Inneren eines Textabschnitts lautet die wichtigste Regel für eine gute Textorganisation: **Nur ein Gedanke pro Satz!** Wenn diese Gedanken dann eine nachvollziehbar logische Abfolge bilden, ist die Verständlichkeit des Textes schon fast gesichert.

## 1.3 Interessantheit

Die Verständlichkeit eines Dokuments ist gleich Null, wenn niemand es liest. Niemand liest gern uninteressantes Material. Als Konsequenz aus diesen beiden

Aussagen ergibt sich unmittelbar, daß ein technischer Text interessant sein muß – naja: sein sollte.

Und wie schafft man das? Dazu gibt es eine Fülle von Regeln, zum Beispiel aus dem Journalismus: Das Wichtigste zuerst; schreibe verständliches Deutsch (Ha, ein Ringschluß!); liefere etwas Neues; und so weiter [?]. Leider reichen alle diese Regeln bei technischen Texten oftmals nicht aus; selbst wenn sie alle eingehalten werden, kann das Ergebnis steril und langweilig sein.

Den Unterschied zwischen „Gäh!“ und „Staun!“ machen häufig die folgenden zwei kleinen Tipps:

1. Habe ein Anliegen! Erzähle eine Geschichte!
2. Lieber „Ich“ als niemand.

Die erste Regel betrifft vor allem die innere Einstellung. Wer als Autor oder Autorin unbeteiligt ist, sich selber nicht für seinen Text interessiert, wird es kaum schaffen, ein interessantes Dokument hervorzubringen. Wer dagegen aktiv „was rüberbringen“ möchte, vermeidet uninteressantes Geschwafel fast automatisch. Ein geschichtenhafter Text ist selbst bei abstrakten Sachverhalten oft zu erzielen, wenn man nicht nur abstrakt schreibt, sondern wo immer möglich konkrete Beispiele für abstrakte Aussagen angibt.

Die zweite Regel ist ein klares Votum für die Benutzung der ersten Person. Die meisten Leute schreiben in wissenschaftlichen Arbeiten und ebenso in technischen Dokumenten aller Art entweder ganz unpersönlich, oder sie verschansen sich hinter der scheinbar neutralen dritten Person und schreiben „die Autoren“, wo sie „wir“ meinen oder „der Autor“, wo sie „ich“ meinen.[Abb. ??] Ich plädiere dafür, wirklich „ich“ oder „wir“ zu schreiben, wenn man das meint. Wenn man das tut, fällt die Befolgung der ersten Regel nämlich leichter und der Leser merkt viel eher, daß da ein menschliches Wesen schreibt. Das tut der Interessantheit gut, weil der Text an Lebendigkeit gewinnt.

Und falls sich irgendwo die Gelegenheit ergibt, ein bißchen Humor in einem Sachtext unterzubringen, wird das der Interessantheit nicht schaden – aber bitte nur, wenn es auch hinpaßt.

Abbildung 1: Verständlichkeit von Wörtern

## 1.4 Adäquatheit

Selbst wenn ein Dokument einen korrekten, verständlichen Inhalt hat, gut organisiert und interessant geschrieben ist, stellt sich noch eine wichtige Frage: Ist er auch angemessen?

Jedes technische Dokument sollte einen möglichst genau bestimmten **Zweck** haben und muß sich fragen lassen, ob es diesen effektiv und effizient verfolgt. Dagegen gibt es drei Haupthindernisse:

1. Ein Dokument kann zu lang sein. Beispiel: ein „Einführungshandbuch“ von 400 Seiten.
2. Ein Dokument kann falsch ausgerichtete inhaltliche Schwerpunkte aufweisen. Beispiele: Ein Entwurfsdokument für eine Bibliothek, das zu sehr auf die Benutzung der Objekte eingeht anstatt auf die Entwurfsentscheidungen, oder umgekehrt ein Benutzerhandbuch, das zu sehr auf die interne Implementation der Objekte eingeht anstatt auf die Benutzung.
3. Ein Dokument kann eine unangemessene (obwohl gute) Organisation aufweisen. Beispiele: Referenzhandbücher ohne Stichwortverzeichnis, mit Zwischenüberschriften, die erst aus dem Zusammenhang verständlich werden, oder mit zu knappem Inhaltsverzeichnis.

Folglich muß der Zweck eines Dokuments beim Schreiben als zentrale Leitlinie jederzeit im Auge behalten werden.

Über diese Hauptanforderungen (Inhalt, Organisation, Interessantheit, Adäquatheit) hinaus sind folgende Unterpunkte von Belang:

## 1.5 Korrektheit

Einer der größten Feinde von Verständlichkeit sind Fehler. Dies ist für den **Inhalt** offensichtlich: Wenn die Fehler innere Widersprüche hervorrufen, geht die Verständlichkeit verloren, andernfalls wird beim Leser ein falsches Verständnis erzeugt. Aber auch auf der Ebene des Sprachstils ist Korrektheit notwendig, da sprachliche Fehler die Aufmerksamkeit vieler Leser stark ablenken und dadurch effektiv die Interessantheit des Textes verschlechtern.

Zur Rolle der Korrektheit in **Rechtschreibung, Zeichensetzung und Grammatik** kann man drei Haupttypen von Haltungen unterscheiden: Die Vorschreiber bestehen darauf, unbedingt an jeder Stelle alle Regeln genau einzuhalten; jeden kleinsten Fehler kreiden sie dick mit rot an und regen sich furchtbar über jeden Text auf, der zu viele davon aufweist. Die Zulasser nehmen die gegenteilige Haltung ein: „Ist doch egal, ob das so stimmt oder nicht. Hauptsache, man kann es verstehen.“. Dazwischen liegen die Pragmatiker, die zwar Korrektheit anstreben, aber auch abweichende Lösungen zulassen bzw. diejenige auswählen, die ihnen im vorliegenden Fall am angemessensten erscheint. Die gleichen Haltungen gibt es auch im Bezug auf Stilregeln wie die unten aufgeführten.

Ich halte die Vorschreiber-Haltung für kleinlich und ineffizient: Man kann seine Energien fast immer für etwas anderes einsetzen, das wesentlich nutzbringender ist. Bei Stilregeln kann zudem ihre sklavische Einhaltung auch zu schauerlich schlechten Resultaten führen. Die Zulasser-Haltung andererseits ist gefährlich für die Verständlichkeit, denn für einen anderen Leser kann eine Unkorrektheit hinderlich für das Verständnis sein, auch wenn sie es für mich nicht ist. Ich halte deshalb einen gesunden Pragmatismus für die einzig sinnvolle Haltung.

## 1.6 Knappheit

Zumeist ist das wichtigste Mittel zum Erreichen einer adäquaten Form, das Dokument bei gegebener Organisation so knapp wie möglich zu halten. Das bedeutet nicht, Sätze zu verstümmeln oder alle weniger wichtigen Fakten einfach wegzulassen, sondern nur, Ausschweifungen zu vermeiden – was den meisten Schreibern gar nicht leicht fällt. Insbesondere sollte inhaltliche Redundanz nur sehr gezielt dort eingesetzt werden, wo sie wirklich nötig erscheint.

## 1.7 Sprache

Unzählige Bücher befassen sich mit dem Thema der guten, verständlichen Schriftsprache. Es sei jedem empfohlen, sich mindestens eines davon zu Gemüte zu führen, z.B. das sehr unterhaltsame „Deutsch für Profis“ [?].

Ich stelle hier nur kurz die allerwichtigsten Regeln zusammen, nicht mit Blick auf sprachliche Korrektheit, sondern auf Verständlichkeit und Interessantheit.

1. Es gibt Sätze der richtigen Länge; nicht zu lang, aber auch nicht zu kurz. Wenn man seine Sätze kritisch prüft, kann man sich da einfach auf sein Gefühl verlassen. Also zum Beispiel bitte nicht

In den meisten Fällen ist es das wichtigste Mittel zum Erreichen einer adäquaten Form, wenn man sich bemüht, das Dokument bei gegebener Organisation so knapp wie möglich zu halten, was aber nicht bedeutet, Sätze zu verstümmeln oder alle weniger wichtigen Fakten einfach wegzulassen, sondern nur, Ausschweifungen zu vermeiden, was den meisten Schreibern gar nicht so leicht fällt.

und auch nicht

Meist ist das wichtigste Mittel für Adäquatheit: Knapphalten. Das bedeutet nicht, Sätze zu verstümmeln. Auch nicht, alle weniger wichtigen Fakten einfach wegzulassen. Es bedeutet nur, Ausschweifungen zu vermeiden. Das fällt den meisten Schreibern nicht leicht.

sondern eher wie oben im Abschnitt ?? über die Knappheit gelesen.

2. Kauderwelsch vermeiden, auch wenn es „im Fach“ üblich erscheint. Oft gibt es ein gutes und verständliches deutsches Wort für ein Fremdwort oder einen englischen Fachbegriff. Dokumentspezifisches Fachvokabular knapp und ordentlich einführen und dabei nach Möglichkeit deutsche Begriffe verwenden – diese Möglichkeit besteht durchaus nicht immer. Abschreckendes Beispiel:

Um den Workflow managen zu können, muß das System write-Zugriff auf alle Records der User-Datenbank haben, um auch die von den Pointern referenzierten Felder updaten zu können.

Übertrieben? Zumindest nicht sehr. Besser wäre es jedenfalls zum Beispiel so:

Um den Arbeitsablauf verwalten zu können, muß das System Schreibrecht für alle Sätze der Benutzerdatenbank haben, um auch die über Zeiger angesprochenen Felder aktualisieren zu können.

Abbildung 2: Daily Dilbert — Bingo

3. Einfache Wörter oder Formulierungen sind besser als pompöse. Auch ohne Kauderwelsch kann man sich verklausulieren. Sehr oft sieht man Formulierungen wie

Zu diesem Zeitpunkt erfolgt die Darstellung des Fensters. Hinsichtlich der Effizienz ist es vorzuziehen, die Variable zur Verwendung zu bringen, als einen Prozeduraufruf durchzuführen.

Au weia, denn es ginge ja auch so:

Nun wird das Fenster angezeigt. Die Variable zu benutzen, ist effizienter als ein Prozeduraufruf.

Auch betont exotische oder altmodische Wörter sollte man lieber vermeiden.

4. Aktiv ist besser als Passiv. Passivsätze werden typischerweise im Amtsdeutsch verwendet; von ihnen wird ein steifer und lebloser Eindruck auf den Leser ausgeübt. Ein aktiver Satz hingegen lebt und zieht den Leser mit in den Text hinein. (Übung: Warum steht hier kein Beispiel?)
5. Verben sind besser als Substantive. Also nicht

Ein Erfordernis für die Darstellung des Fensters ist die vorherige Aktualisierung des Widgets.

sondern lieber

Um das Fenster darzustellen, muß man zuvor das Widget aktualisieren.

Von allen diesen Regeln gibt es natürlich Ausnahmen. Siehe die Diskussion über Korrektheit oben.

## 1.8 Schönheit

Als einen besonderen Aspekt von Interessantheit sollte man anstreben, daß auch ein technischer Text ästhetisch befriedigend ist. Wenn man dies erreicht (das ist aber schwer!), kann daraus eine Zugeneigtheit der Leser erwachsen, die die Interessantheit und damit die effektive Verständlichkeit enorm erhöht. Merke: Beruf und Vergnügen können (und sollten) durchaus gemeinsam auftreten!

## 1.9 Fazit

Um zu der zu Beginn gestellten Frage zurückzukehren: Das Schreiben eines technischen Dokumentes ist insofern fast das gleiche wie das Schreiben eines Zeitungsartikels, als die effektive Verständlichkeit im Mittelpunkt steht. Die Anforderungen sind sogar recht ähnlich denen für das Schreiben eines guten Romans; allerdings muß beim technischen Text im Konfliktfall (aber auch nur dann) die Korrektheit und Knappheit stets der Schönheit vorgezogen werden.

## 2 Vorgehen

Ähnlich wie bei Software kann man auch bei technischer Dokumentation eine produktorientierte oder eine prozeßorientierte Sichtweise einnehmen. Die produktorientierte wurde im vorherigen Abschnitt verwendet: Im Mittelpunkt stehen erwünschte und unerwünschte Eigenschaften des Dokuments. Diese Betrachtungen stecken zwar ein Ziel ab, geben jedoch nur schwache Hinweise, wie man vorgehen sollte, um zu einem guten Dokument zu kommen.

Im diesem Abschnitt besprechen wir nun die prozeßorientierte Sichtweise des Schreibens, also das Vorgehen. Verblüffenderweise kann man sehr viel von den Vorgehensmodellen der Softwareentwicklung direkt auf das Schreiben übertragen. [?] beschreibt diese Analogie ausführlicher und enthält auch eine umfangreiche kommentierte Liste von Literatur zum Thema Schreiben.

### 2.1 Wie entwickelt man doch gleich Software?

Das Phasenmodell (Wasserfallmodell) der Softwareentwicklung kann man in seinen Einzelteilen recht gut auf die Erstellung technischer Dokumente übertragen. In den folgenden Abschnitten werde ich genau das tun. Anschließend bespreche ich auch noch die wichtigsten Vorgehensalternativen, die vom Phasenmodell abweichen.

Ebenso wie bei der Softwareentwicklung ist es auch beim Schreiben schwierig, eine lineare Abfolge der Schritte einzuhalten. Allerdings ist dies beim Schreiben ein viel geringeres Problem: Wegen der geringeren Strenge natürlicher Sprache im Vergleich zu Computerprogrammen können leichte Schwächen ohne weiteres hingegenommen werden und wegen der geringeren Größe des Entwicklungsteams (oft nur eine Person) sind nachträgliche Änderungen schon abgeschlossener Phasen viel einfacher.

### 2.2 Analyse

In der Analysephase wird eine **Problem- und Aufgabenbeschreibung** erstellt. Diese Texte sind nicht eigentlich Bestandteil des herzustellenden Doku-

ments, sondern dienen dem Autor zu seiner eigenen Orientierung. Teile der Beschreibungen finden sich jedoch normalerweise später in der Einleitung des Dokuments wieder.

Die Problem- und Aufgabenbeschreibung enthält

1. Eine Beschreibung des größeren Zusammenhangs, in dem das Dokument angesiedelt ist.
2. Die Beschreibung des konkreten Problems, das im Dokument behandelt wird.
3. Die Charakterisierung der Ziele, die das Dokument erreichen soll (z.B. der Information, die es liefern soll).
4. Eine Begründung, warum und für wen das Dokument wichtig ist.
5. Die Charakterisierung des Vorwissens der angepeilten Leserschaft.
6. Eine Auflistung relevanter Randbedingungen: Zeitbeschränkungen, Umfangsbeschränkungen, technische Randbedingungen (Medien etc.), äußere Vorgaben (Standards) für Stil, Organisation oder Format.

Von zentraler Bedeutung in der Analysephase ist die richtige Charakterisierung der Leserschaft, insbesondere ihres Vorwissens. Schreibt man ein Dokument für eine Gruppe von Gleichen, wie es in der Softwareentwicklung häufig vorkommt, so ist dies relativ einfach. In anderen Fällen empfiehlt sich ein zweistufiges Vorgehen: Man nimmt ein Modell eines gewünschten Lesers an und sammelt dann, mit welchen Unterschieden dazu bei der übrigen Leserschaft zu rechnen ist.

## 2.3 Planung

Bei größeren Dokumenten empfiehlt sich die Aufstellung eines Prozeßplans. Das gilt besonders, wenn mehr als eine Person am Dokument schreiben soll. Der Prozeßplan enthält die Zeitplanung und fixiert Entscheidungen im Hinblick auf Rahmenbedingungen wie Softwarewerkzeuge und Versionsverwaltung. Er enthält außerdem eine Liste benötigter Unterlagen sowie eine Beschreibung der Konventionen zur Entscheidung von Zweifelsfällen sprachlicher, inhaltlicher oder textstruktureller Art. Die letzteren beiden Listen müssen während der Erstellung des Dokuments ständig aktualisiert werden und sind insofern Schreibhilfsmittel, die über einen reinen Plan hinausgehen.

## 2.4 Entwurf

Als Entwurfsmethode eignet sich bei Texten das Top-Down-Vorgehen; man stellt also zu Beginn die Liste der Grobabschnitte oder Kapitel auf, etwa so, wie sie später im Inhaltsverzeichnis erscheinen wird. Dann wird schrittweise verfeinert, indem man die Überschriften der Unterabschnitte und eventuell Unter-Unterabschnitte hinzufügt. Das so entstandene Inhaltsverzeichnis bildet quasi den Grobentwurf. Dieser kann nun zu einem Feinentwurf erweitert werden, indem man zu jedem Abschnitt eine Reihe von Stichworten oder -phrasen aufschreibt, die den Inhalt des Abschnitts näher bestimmen.

Bei dieser Vorgehensweise entdeckt man ständig Fehler oder Ungeschicklichkeiten im Aufbau, die durch entsprechendes Ändern des Entwurfs ausgebügelt werden müssen. Dieses Problem entsteht aus der (im Vergleich zu Software-Entwürfen) relativen Detailarmut des Textentwurfs, durch die Inkonsistenzen erst spät sichtbar werden. Ebendiese Detailarmut ist aber zugleich ein Segen, denn sie macht Änderungen auch sehr leicht durchführbar. Die meisten guten Autoren verwenden das Top-Down-Vorgehen nur als eine Art Gerippe für ihren Schreibprozeß und schreiben in Wirklichkeit weitgehend assoziativ: Der jeweils nächste Entwicklungsschritt wird an dem Abschnitt gemacht, zu dem einem der jeweils letzte Schritt gerade den nächsten sinnvollen Gedanken eingegeben hat. Die Rückkehr zum Top-Down-Vorgehen wird immer dann vorgenommen, wenn eine Lücke in diesem Prozeß entsteht. Die Rückkehr verhindert, sich im assoziativen Durcheinander zu verzetteln.

## 2.5 Implementation

Weniger noch als beim Programmieren läßt sich beim Schreiben eine klare Grenze zwischen Entwurf und Implementation angeben. Durch das Top-Down-Vorgehen geht eine Aktivität fast stufenlos in die andere über. Zumindest muß man aber irgendwann anfangen, die einzelnen Abschnitte „endgültig“ auszuformulieren, was eindeutig eine Implementationstätigkeit ist. Egal wie gründlich die Vorarbeiten waren: Stets tauchen dabei noch neue Ungereimtheiten auf. Da paßt hier ein Gedanke nicht ohne Lücke an den nächsten und da wird dort eine Grundlage benötigt, die weiter oben noch nicht gelegt wurde. In diesen Fällen muß der Entwurf entsprechend korrigiert werden, bevor man weiter fortfahren kann. Bei solchen Änderungen erfordert es einige Disziplin, sie tatsächlich auf der entsprechenden Entwurfsebene durchzuführen, anstatt im Text zu flickschustern.

Im Prinzip gilt bei der Implementation das gleiche Spannungsverhältnis von assoziativem und Top-Down-Vorgehen. Allerdings ist hier ein assoziatives Arbeiten zusätzlich erschwert, weil so viele Details beachtet werden müssen. Deshalb sollte der Volltext als Faustregel in ununterbrochenen Arbeitsblöcken geschrieben werden, die mindestens einen ganzen Absatz umfassen.

Es gibt allerdings auch ganz andere Herangehensweisen an die Implementation als die Orientierung am Top-Down-Modell: Bei einer Arbeitsweise, die mit dem **Rapid Prototyping** verwandt ist, werden zunächst ohne viel Rücksicht auf die Qualität längere Textpassagen mit möglichst wenig Unterbrechungen geschrieben. Danach erst erfolgen Aufräum- und Verbesserungsarbeiten, die den an sich schon vollständigen Text seiner endgültigen Form annähern.

Der genau gegenteilige Ansatz ist mit dem **Risikomodell** (Spiralmodell) der Softwareentwicklung verwandt: Hier werden zuerst diejenigen Teile bearbeitet, die am schwierigsten und wichtigsten zu sein versprechen. Sie werden sehr gründlich hergestellt und gut ausgefeilt, bevor darauf aufbauend andere Teile in Angriff genommen werden.

Beim Ansatz mit **Wiederverwendung** steht am Beginn der Texterzeugung der Versuch, möglichst viele benötigte Teile aus früher geschriebenen Texten direkt oder mit Änderungen zu übernehmen. Diese Möglichkeit besteht dann, wenn man mehrere Texte zu engverwandten Themen zu schreiben hat, so daß sich etwa Definitionen oder Erläuterungen von Grundlagen übernehmen lassen. In diesem Fall sind manchmal ganze Abschnitte ohne Änderungen übernehmbar.

Ein anderer Fall ist die Wiederverwendung von Textschablonen, d.h. Absätzen oder Abschnitten, in denen ein bestimmter Gedankengang verfolgt wird und die durch die Änderung von Teilsätzen oder Sätzen an das aktuelle Thema angepaßt werden können. Beispiel: Ein Abschnitt über Zweck und Verwendung einer Prozedur X hat die Form Zweck, Vorbedingung, Nachbedingung, Besonderheiten und kann oftmals unabhängig von X die gleichen Formulierungen zur Verbindung dieser Teile verwenden; eventuell gibt es mehrere solcher Bausteine für ähnliche Zwecke, um erstens für Variabilität in der Formulierung sorgen zu können und zweitens für unterschiedliche Fälle eine angemessene Variante bereit zu haben.

Bei Wiederverwendung besteht eine gefährliche Tendenz, daß der erzeugte Text nicht gut seiner Leserschaft angemessen ist, wenn wiederverwendete Bausteine für eine andere Leserschaft geschrieben waren. Außerdem wirkt der Text selten richtig wie aus einem Guß. Wiederverwendung ist deshalb mit großer Vorsicht zu genießen. Bei der Wiederverwendung von Text anderer Autoren (auch aus der eigenen Organisation) sind die Regeln der Fairneß und die Urheberrechtsgesetze zu beachten.

## 2.6 Testen und Überarbeitung

Wenn das Produkt (sprich: Dokument) „fertig“ ist, erfolgt ähnlich wie beim Programmieren eine Testphase. Gute Schreiber beginnen damit schon während der Textproduktion und führen viele Prüfungen ständig nebenher oder gelegentlich zwischendurch aus. Beim Testen wird das Dokument auf verschiedenen Ebenen untersucht, um die Einhaltung der Ziele zu überprüfen und ggf. durch Nachbesserung zu erreichen.

Auf der **Mikroebene** geht es um lokale Eigenschaften des Textes wie

1. Stimmen Rechtschreibung, Zeichensetzung, Grammatik?
2. Ist die Wortwahl angemessen?
3. Sind die Sätze verständlich (Länge, Aufbau) und angenehm lesbar?

Diese Prüfung können alle Personen vornehmen, die ein wenig vom Themengebiet verstehen. Der Autor selbst ist allerdings weniger gut geeignet, weil er für seine eigenen Mikroebenenfehler weitgehend blind ist.

Auf der **Makroebene** werden die globalen Anforderungen an den Text geprüft. Hierzu werden die Notizen aus der Analysephase herangezogen.

1. Stellt er sein Thema und seine Ziele klar?
2. Paßt der Text zu seiner angestrebten Leserschaft (Inhalt, Detaillierungsgrad, Ton)?
3. Ist der Text seinem Zweck angemessen (Umfang, Inhalt, Aufbereitung, Form)?
4. Ist die Organisationsstruktur klar ersichtlich?
5. Ist alle benötigte Information enthalten?
6. Hat der Gedankengang Lücken?

Für die Effizienz eines Tests ist es wichtig, daß man sich vor seinem Beginn entscheidet, ob man die Mikro- oder die Makroebene prüfen will. Wer beides zugleich versucht, überfordert sich in der Regel und übersieht dann wichtige Schwächen. Bei Prüfungen auf der Makroebene sollte man sich sogar auf einen oder zwei der angeführten Punkte pro Prüfdurchgang beschränken. Die gleichen Anmerkungen gelten analog für das Vorgehen bei der Korrektur der Schwächen.

## 2.7 Wartung

Viele Dokumente müssen über eine längere Lebensdauer hinweg mehrfach angepaßt, erweitert oder geändert werden. Dabei tauchen ganz ähnliche Probleme auf, wie bei der Wartung von Software

1. Die Entwurfsentscheidungen und der entstandene Entwurf selbst sind nicht gut genug dokumentiert, um bei der Wartung verstanden zu werden. Deshalb zerstört die Wartung allmählich immer mehr von der ursprünglichen Struktur.
2. Selbst wenn dieser Fall nicht eintritt, werden die Entwurfsbeschreibungen oft nicht ausreichend mitgepflegt. Das gleiche gilt für die Anforderungsbeschreibungen.
3. Warter haben die Neigung, Änderungen möglichst lokal zu machen. Auch dies schwächt auf die Dauer die globale Struktur.
4. Eine wohldokumentierte Versions- und ggf. Konfigurationsverwaltung ist notwendig.
5. Sind mehrere Varianten eines ähnlichen Textes zu pflegen, verstärkt sich das Problem entsprechend.

Die besten Mittel zur Lösung dieser Probleme sind leicht erfaßbare, kompakte und fest an das Dokument gekoppelte Beschreibungen der Anforderungen und des Entwurfs sowie ein möglichst modularer Aufbau der Dokumente, der Zusammenhänge zwischen Teilen so gering wie möglich hält und so explizit wie möglich sichtbar macht.

Anforderungs- und Entwurfsbeschreibungen werden am besten als Annotationen technisch direkt mit ins Dokument integriert (d.h. in die entsprechende Datei, nicht in den Ausdruck). Zusammenhänge, sowohl innerhalb oder zwischen Dokumentteilen als auch zwischen Dokument und obigen Beschreibungen, kann man über technisch verfolgbare und prüfbare Querverweise explizit machen.

## 2.8 Bilder und andere Medien

Die Regel „Ein Bild sagt mehr als tausend Worte“ ist zwar auch für technisches Schreiben ein nützlicher Hinweis, muß aber stets mit Vorsicht umgesetzt werden. (Unter „Bild“ wollen wir hier jede Art graphischer Darstellung verstehen.)

Zunächst einmal kann in vielen Fällen in der Tat eine sehr große Informationsmenge mit einem Bild transportiert werden, die als Text erheblich mehr Platz verschlungen hätte. Krasse Fälle dieser Art sind aber eher selten.

Allerdings kann man sehr oft mit einem Bild eine gewisse Information erheblich schneller übermitteln, was gerade für technische Dokumentation außerordentlich wünschenswert ist. Insofern sollten Bilder möglichst häufig eingesetzt werden. Dabei sind jedoch zwei Fallen unbedingt zu vermeiden:

Erstens brauchen die meisten Bilder eine Erläuterung, zumindest, wenn mehr als ein ungefährender Eindruck von etwas übermittelt werden muß. Diese Notwendigkeit kann aber den Vorteil des Bildes wieder völlig zunichte machen, wenn die Erklärung entweder zu lang, zu umständlich oder zu verklausuliert ist oder wenn sie sich schwer auffindbar irgendwo im Text versteckt. Eine gute Erklärung sollte sich direkt in der Bildunterschrift unterbringen lassen und möglichst ohne Rückgriff auf den Text verständlich sein. Kriegt man das partout nicht hin, ist wahrscheinlich das Bild überfrachtet. Abbildung ?? ist ein abschreckendes Beispiel für ein völlig überfrachtetes Bild (Bitte nicht nach einem Sinn suchen...).

### Abbildung 3: Überfrachtete Darstellung

Zweitens erzeugt der Druck zum Einsatz von Bildern manchmal die Neigung, triviales in einer Abbildung darzustellen. Davon fühlt sich der Leser oft auf den Arm genommen und versagt dann eventuell dem Rest des Textes die ernsthafte Aufmerksamkeit. Abbildung ?? ist ein abschreckendes Beispiel dieser Art.

Für Multimedia/Hypermedia-Dokumente gelten weitgehend die gleichen Regeln, wie sie bisher unter der Annahme von Papierdokumenten besprochen wurden: Selbst ein Hypermedia-Dokument mit seinen zahlreichen Querverweisen sollte idealerweise eine gut organisierte, linear lesbare Struktur aufweisen, weil nur diese ein vollständiges Erfassen durch den Benutzer garantieren kann. Auch wenn man die Abstützung auf geschriebenen Text zur Informationsübermittlung

#### Abbildung 4: Übersichtliche Darstellung

verringert, entstehen die gleichen Probleme der Angemessenheit der Darstellung bezüglich des Dokumentenzwecks und des Zielpublikums. Die vermeintlich viel größere Interessantheit von Multimedia-Dokumenten ist zum Teil eine Illusion, die aus der relativen Neuheit dieser Dinge entsteht. Ein wirkliches inhaltliches Interesse aufrechtzuerhalten (anstatt nur eine Art unterhaltsames Rauschen zu erzeugen), ist aber mit Multimedia ungefähr genauso schwierig wie mit Papier.

### 2.9 „Goldene Regeln“

Es gibt umfangreiche Sammlungen von Regeln mit dem Ziel, korrektes, gutes, verständliches oder interessantes Deutsch (oder Englisch) zu schreiben. Da diese Sammlungen häufig zu umständlich oder umfangreich sind, gebe ich hier nur eine kleine Zahl von Meta-Regeln, mit denen man aber schon ziemlich weit kommt:

1. Habe ein Anliegen (neudeutsch: message).
2. Orientiere Dich an Deinen Lesern.
3. Schreibe wie Du sprichst und feile das Ergebnis dann aus.
4. Schreibe nur einen Gedanken pro Satz.
5. Halte Dich nie sklavisch an einer Regel fest.

Die ersten vier Regeln bilden zusammen den Schlüssel zu interessantem, lesbarem Text: Nur wenn man genau weiß, was genau es ist, das man seinen Lesern sagen will, schafft man es, sich wirklich auf das Wesentliche zu konzentrieren und eine klare Struktur in den Text zu bringen. Und nur, wenn es einem wichtig ist, das „rüberzubringen“, was man sagen möchte, schafft man es auch, die Präsentation interessant zu gestalten. Dies sichert das Interesse des Lesers auf der Ebene der Textorganisation. Die dritte und vierte Regel liefern die Methode, mit der man sich auf der Ebene einzelner Sätze vor Unverständlichem und damit Uninteressantem bewahren kann.

Die letzte Regel besagt lediglich, daß keine Regel gut genug ist, um immer zu gelten. Wenn es also einen genügend guten Grund gibt, brich sie! Ohne

Augenmaß und Entschlußkraft kommt man bei größeren Vorhaben selten zu befriedigenden Ergebnissen.

## Literatur

- [DLL92] Marcus Deininger, Horst Lichter, Jochen Ludewig, and Kurt Schneider: *Studien-Arbeiten: ein Leitfaden zur Vorbereitung, Durchführung und Betreuung von Studien-, Diplom- und Doktorarbeiten am Beispiel Informatik*, vdf Verlag der Fachvereine und Teubner, Stuttgart, 1992.
- [LPD91] Linda Levine, Linda H. Pesante, and Susan B. Dunkle: *Technical writing for software engineers*, Technical Report SEI-CM-23, Carnegie Mellon University, Software Engineering Institute, Pittsburgh, PA, November 1991.
- [Sch82] Wolf Schneider: *Deutsch für Profis. Handbuch der Journalistensprache – wie sie ist und wie sie sein könnte.*, 2. Auflage. Stern Buch, Gruner und Jahr, 1982.
- [You89] Matt Young: *The Technical Writer's Handbook*, University Science Books, Mill Valley, CA, 1989.