

# Artikelserie "Programme auf dem Prüfstand"

(Auszug aus der Veröffentlichung im OKKSA-Newsletter Nr. 24 vom April 2014)

## Teil 13: Benutzerdokumentation: Wieviel Text muss sein?

(von Dr. Uwe Schwochert)

### Abgrenzung zur Produktbeschreibung

Während im letzten Teil der Artikelserie die Anforderungen an die Produktbeschreibung eines Programms betrachtet wurden, soll dieses Mal die Benutzerdokumentation im Fokus stehen.

Was ist eigentlich die Benutzerdokumentation (i. F. BD)? Im Kontext der OKKSA Kriterienkataloge wird damit der Teil der Dokumentation verstanden, der dem Benutzer Hilfestellung bei der Bedienung des Programms gibt. Das entsprechende Kriterium bezieht sich zunächst auf allgemeine Qualitätsmerkmale der BD und lautet:



[FÜ08.04] Die für den unmittelbaren Benutzer vorgesehenen Teile der Dokumentation (Benutzerdokumentation) erfüllen wesentliche Anforderungen an die Benutzerdokumentation nach [ISO25051]:

1. Vollständigkeit;
2. Richtigkeit;
3. Widerspruchsfreiheit;
4. Verständlichkeit.

[MUSS-Kriterium]

Es geht bei der BD nicht mehr darum, darzustellen, unter welchen Bedingungen das Programm welche Funktionen ausführt. Das sollte der Benutzer bereits durch die Produktbeschreibung erfahren können. Dort ist Übersichtlichkeit und Systematik gefragt, damit der (potentielle) Programmanwender leicht feststellen kann, ob das Programm eine bestimmte Funktionalität aufweist bzw. unter welchen Bedingungen diese gegeben ist.

Wenn das klar ist, stellen sich dem Benutzer jedoch weitere Fragen, z. B.:

Leistet diese Funktion auch das, was ich genau brauche?

Über welchen Menüpunkt kann ich diese Funktion aufrufen?

Was ist bei der Benutzung dieser Funktion zu beachten?

Kann ich diese Funktion austesten oder verändere ich damit Fachdaten?

Wenn diese Hürde genommen ist, könnten sich weitere Fragen stellen, z. B.:

Es tritt eine Fehlermeldung auf – was bedeutet das?

Es funktioniert nicht wie gedacht – mache ich etwas falsch oder geht das nicht anders?

Betrachten wir ein Beispiel: Eine Textverarbeitung (=Programm) beinhaltet eine Rechtschreibprüfung (=Funktionalität).

In der Programmdokumentation dieser Textverarbeitung könnten sich dazu folgende Informationen finden:

P1: Diese Textverarbeitung verfügt über eine automatische Rechtschreibprüfung.

P2: Es werden die Sprachen Deutsch, Englisch und Französisch unterstützt.

P3: Die Rechtschreibprüfung muss separat installiert / erworben werden.

Inhalt der BD sollte es nun sein, darzustellen:

B1: Wie kann dieses Modul erworben werden?

B2: Wie wird dieses Modul installiert/deinstalliert?

B3: Wie wird die Rechtschreibprüfung aufgerufen?

B4: Wie wird die Sprache eingestellt?

oder auch:

B5: Wie kann ich die Rechtschreibprüfung um eigene Wörter ergänzen?

B6: Warum wird bei mir ein falsch geschriebenes Wort trotzdem als richtig erkannt?

Es könnten darüber hinaus auch folgende Informationen bereitgestellt werden:

B7: Stellt die automatische Rechtschreibprüfung im Hintergrund eine nennenswerte Rechnerbelastung dar?

B8: Ich möchte gern ein eigentlich richtiges Wort immer als Fehler kennzeichnen lassen – ist das möglich?

### **Struktur der Benutzerdokumentation**

Am Beispiel mit der Rechtschreibprüfung wird klar, dass es keine allgemeingültigen Regeln geben kann, was genau in einer BD stehen muss. Letztlich ist beim Autor einer Benutzerdokumentation ein gutes Verständnis der Aufgabenstellungen und Sichtweise der Anwender des Programms erforderlich.

Doch eine BD kann auch in ihrer Form deutlich variabler gestaltet sein, als eine Produktbeschreibung. Denn beim Leser der BD kann davon ausgegangen werden, dass er das Programm bereits im Zugriff hat. Neben einer

- gedruckten Dokumentation

oder einer

- Dokumentation als pdf-Datei

ergeben sich damit eine Reihe alternativer und feiner gegliederter Formen der Bereitstellung dieser Informationen, wie z. B.:

- Hilfe-Funktion im Programm,
- Informationen direkt auf den jeweiligen Bedienmasken,
- Informationen in Form von Quick-Tipps (sich selbst einblendende Sprechblasen),
- Assistentenfunktionen,
- direkter Link aus dem Programm auf Internetseiten mit den Hilfetexten.



Im Rahmen der Programmprüfung werden diese Möglichkeiten hinterfragt:

[FÜ08.05] Die Programmnutzung wird durch eine direkt aus dem Programm aufrufbare **Hilfefunktion** unterstützt.

[MUSS-Kriterium]

Eine Dokumentation, die bei der Bedienung des Programms helfen soll, ist natürlich am nützlichsten, wenn sie selbst in die Programmbedienung integriert ist.

Basis ist zunächst die Bereitstellung der Dokumentationssteile über ein separates Hilfemenü (Icon-Leiste, Links, ...) im Programm. Hier sollte nun, neben dem Einstieg in die Gesamthilfe (üblicherweise auch auf die Taste F1 gelegt), auch ein direkter Zugang zu der Hilfeseite möglich sein, die die gerade genutzte Funktion betrifft (z. B. über Um-

schalt+F1). So entfällt für den Benutzer der aufwändige Weg:

"Wo ist die Programmdokumentation überhaupt?"

"In welchem der vielen Dokumente finde ich die Information zum gerade bearbeiteten Vorgang"

"Wonach muss ich dafür suchen?"

"Habe ich wichtige Informationen übersehen oder ist das alles?"

Voraussetzung hierfür ist die Bildung von funktions- bzw. maskenorientierten Dokumentationsbausteinen, welche die jeweiligen Hintergrundinformationen<sup>1</sup> bereitstellen:

[FÜ08.05a] Die **Hilfefunktion ist kontextsensitiv** und stellt bei ihrem Aufruf zielgerichtete Informationen zu den gerade in Benutzung befindlichen Programmfunktionen dar.

[KANN-Kriterium]

Zu Such-, Prüf- und Dokumentationszwecken ist es trotzdem sinnvoll, wenn die Dokumentation auch als Gesamtdokument verfügbar ist. Am Markt verfügbare Hilfe-Compiler unterstützen diesen Spagat, indem sie neben einer kompilierten und in das Programm integrierbaren Hilfe-Datei auch eine pdf-Version der Dokumentationstexte bereitstellen. Aber selbst wenn die BD "nur" als pdf-Datei bereitsteht, kann sie als kontextsensitive Hilfe bereitgestellt werden. Über eine versteckte Funktion im Adobe® Reader® kann das Programm beim Öffnen der pdf-Datei gezielt zu einer bestimmten Seite oder zu einem Stichwort springen und so die Suche der richtigen Hilfeseite deutlich vereinfachen<sup>2</sup>.

Auch innerhalb der Hilfetexte zu einem Vorgang oder einer Bildschirmmaske kann eine gute Strukturierung hilfreich sein. Statt alle Informationen zum Ausfüllen der Maske in 2-4 Dokumentationsseiten unterzubringen, ist es oftmals benutzergerechter, wenn Informationen zu einzelnen Bedien- und Eingabeelementen direkt auf dem Bildschirm sichtbar sind. Da der Text zwischen Buttons, Eingabefeldern und Tabellenbereichen meist nicht ausreicht, könnten die einzelnen Bereiche z. B. mit "Fragezeichen – Buttons" versehen werden oder auch dann erscheinen, wenn der Mauscursor längere Zeit über einem Bedienelement verweilt. Je nach Bedienkonzept ist auch ein Kontextmenü auf der rechten Maustaste geeignet, eine Beschreibung zu einem bestimmten Bedienelement aufzunehmen.



Abhängig davon, ob die Hinweise zu einem Bedienelement für den Gesamt-Bedienvorgang überhaupt relevant sind, könnte dieser Beschreibungstext in der Beschreibung der Bedienmaske insgesamt entfallen.

Bei der Beschreibung auf Ebene der Bedienelemente ist zu unterscheiden zwischen formaler Bedieninformation ("Das ist ein Datumsfeld. Geben Sie ein Datum in Form von 23.04.2014 ein. Siehe auch Bedienhandbuch unter Eingabeformate.") und inhaltlicher Information ("Geben Sie hier das Datum der Anschaffung des neu anzulegenden Vermögensgegenstandes ein. Nähere Informationen dazu finden Sie im Glossar unter Anschaffungszeitpunkt.").

Nicht durchgesetzt haben sich bisher natürlichsprachliche Hilfsassistenten, wie zum Beispiel der Office Assistant (auch "Mr. Clippy" genannt). Während seine Nachfolger wie S Voice und Siri auf Smartphones inzwischen immer mehr Verbreitung finden, ist die Nutzbarkeit derartiger Assistenzsysteme bei Fachprogrammen begrenzt - zu groß ist hier die sprachliche Variabilität der jeweiligen Anwendungsgebiete.

<sup>1</sup> Es ist naheliegenderweise wenig sinnvoll, wenn die Hilfe zu einer Programmfunktion die Inhalte der Bildschirmmaske wiedergibt. Stattdessen sollten die Mechanismen und Hintergründe der Programmfunktion so erläutert werden, dass der Benutzer Sicherheit über das gewinnt, was er gerade tut.

<sup>2</sup> Entsprechende Aufrufparameter werden durch den Adobe® Reader® bereits seit vielen Versionen unterstützt:  
 Direkt zu Seite springen: `acrord32.exe /A "page=3" "c:\test.pdf"`  
 Suche mit Suchbegriff starten: `acrord32.exe /A "search=Haushaltssatzung" "c:\test.pdf"`

Wenn nun alles masken- oder sogar eingabefeldbezogen strukturiert und integriert ist<sup>3</sup>, ist dann eine allgemeine, übergreifende Benutzerdokumentation überhaupt noch notwendig?

Ja! So wie in der Medizin der Allgemeinmediziner den Überblick haben sollte, auch wenn es Fachärzte gibt, benötigt der Benutzer eine zentrale Anlaufstelle für allgemeine, noch nicht zugeordnete oder nicht zuordenbare Benutzungaspekte. Neben den im konkreten Bediendkontext notwendigen Hilfestellungen benötigt er auch folgende Informationen:

- Beschreibung der allgemeinen Bediensystematik des Programms (inklusive des Hilfesystems),
- strukturierte Darstellung, wie bestimmte fachliche Aufgabenstellungen des Anwendungsgebietes mit dem Programm unterstützt werden (inkl. Hinweisen zu nicht unterstützten Funktionen bzw. dazu, mit welchen Zusatzmodulen/Workarounds diese Funktionen angegangen werden können),
- allgemeine Informationen zur dokumentierten unterstützten Programmversion, zu Dokumentationssystematik und zum Programmentwickler (inkl. Angabe der für die Redaktion der Dokumentation verantwortlichen Stelle),
- Beschreibung zu den Supportoptionen.

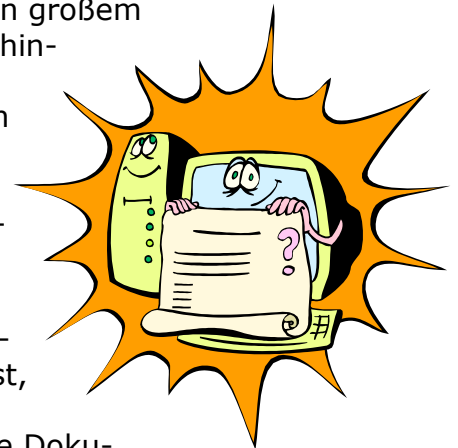
### **Dynamik der Benutzerdokumentation**

Die BD als anwendungsnahe Unterstützungsfunktion ist genau genommen noch änderungsanfälliger als das Programm selbst. Denn sie soll nicht nur die intendierte Bedienung des Programms in seiner aktuellen Version beschreiben, sondern idealerweise auch neuere Erkenntnisse z. B. aus Benutzerrückfragen aufnehmen können. Viele Hotline-Anrufe könnten gespart werden, wenn auftretende Bedienprobleme sofort in der Hilfe zur jeweiligen Bedienmaske hinterlegt werden würden.

Aus diesem Grund muss zusammen mit der Konzeption der BD auch betrachtet werden, wie diese Informationen zeitnah ergänzt und aktualisiert werden können.

So ist es zum Beispiel bei komplexen Anwendungsprogrammen unumgänglich, dass der Benutzer über die BD hinaus organisationsspezifische Hinweise und Einweisungen erhält. Diese können naheliegenderweise nicht ab Werk in der BD enthalten sein. Da sie aber oft auch direkt die Bedienung des Programms betreffen, wäre es von großem Vorteil, wenn der Programmbenutzer zusammen mit den Bedienungshinweisen ab Werk auch über die hausspezifischen Besonderheiten der Programmbedienung informiert würde. So könnte vermieden werden, dass der Benutzer Hinweise und Regelungen an vielen verschiedenen Orten erkunden muss. Dies ist erfahrungsgemäß eine große Hemmschwelle beim Zugriff auf diese Informationen.

Das spielt letztlich auch für die Programmgestaltung und deren Prüfung eine Rolle: Da die Problematik einer übersichtlichen Vor-Ort-Dokumentation für den Programmentwickler vorhersehbar ist, ist es wünschenswert, dass entsprechende Hinweise direkt auch dort integriert werden können, wo auch die programmspezifische Dokumentation zu finden ist:



<sup>3</sup> Das betrifft im Übrigen nicht nur die Benutzersicht, sondern auch die des Entwicklers: Statt einer nacheilenden Beschreibung der Programmbedienung dürfte es oftmals hilfreich sein, die Gedanken und Informationen des Programmentwicklers zu einer Bedienmaske oder einem Datenfeld direkt bei dessen Entwurf festzuhalten und programmtechnisch als kontextbezogenen Hilfetext vorzubereiten. Zu oft leidet die Qualität von Benutzerdokumentationen unter nacheilenden und entwicklungsfernen Redaktionsprozessen.

[FÜ08.05b] Die **Hilfefunktion kann** für die Anwendung vor Ort um nutzerspezifische Informationen **ergänzt werden**.

[KANN-Kriterium]

Die Umsetzung dieser Integration erfolgt im einfachsten Fall dadurch, dass bei der Programmnutzung (z. B. beim Programmstart) ein Link oder Button direkt zu organisations-spezifischen Regelungen verweist<sup>4</sup>. Weitergehend - auch das wird bereits praktiziert - könnten die kontextbezogenen Hilfeseiten einen separaten Bereich enthalten, der Ergänzungen hinsichtlich der Nutzung einer konkreten Programmfunktion innerhalb der Organisation enthält. Die Hilfe zur Erfassungsmaske eines Anlagegutes könnte z. B. zusammen mit der entwicklerseitigen Information zur Bedienung der Maske auch einen Bereich oder Verweis enthalten, der die hausinternen Vorgaben zur Verwendung von Abschreibungszeiträumen enthält.

Eine Dokumentation als sich weiterentwickelndes Dokument sollte darüber hinaus per Internet aktualisier- und ergänzbar sein:

[FÜ08.06] Die Dokumentation wird durch **über das Internet verfügbare Informationen** ergänzt und aktuell gehalten.

[KANN-Kriterium]

Das bedeutet, dass im einfachsten Fall die Benutzerdokumentation als .pdf-Datei oder HTML-Präsentation aktuell im Internet verfügbar ist (und der Benutzer darüber auch informiert wird).

Ergänzend könnten im Internet für die Programmbenutzer eine FAQ und Hilfestellungen für besondere Anwendungssituationen bis hin zu Fachartikeln bereitgestellt werden. Ggf. stellt auch ein (moderiertes) Forum zum Austausch der Programmbenutzer untereinander eine wertvolle Ergänzung dar.

## **Zusammenfassung**

Die Benutzerdokumentation wird, im Unterschied zur Programmbeschreibung, den aktiven Benutzern eines Programms in verschiedenen Formen und Strukturen bereitgestellt. Hinsichtlich der Struktur wird eine Dreigliedrigkeit empfohlen:

1. Allgemeine Informationen zum Programm, seiner Bedienung und Angehen verschiedener Aufgabenstellungen des Anwendungsbereiches (s. o.).
2. Konkrete kontextabhängige Informationen zur Bedienung einzelner Vorgänge bzw. Masken, aufrufbar aus der Bearbeitung dieser Vorgänge heraus.
3. Informationen zur Benutzung einzelner Bedien- und Eingabeelemente auf der jeweiligen Bearbeitungsmaske, getrennt nach allgemeinen, formalen Hinweisen ("Wie kann die Tabellendarstellung geändert werden?") und inhaltlichen Hilfestellungen ("Was wird in dieser Tabelle angezeigt?").



Darüber hinaus sollte der Dynamik einer Dokumentation Rechnung getragen werden, indem die Integration anwenderspezifischer Hinweise ermöglicht wird. In jedem Fall sollte es zum Programm einen Anwenderbereich im Internet geben, wo weitergehende Unterstützungsressourcen (inkl. idealerweise eines Anwenderforums) bereitgestellt werden.

Beispielhaft könnte ein entsprechender Menüpunkt "Hilfe und Information" im Programm

<sup>4</sup> Was am einfachsten darauf hinausläuft, dass an einem vordefinierten oder einstellbaren Platz eine .pdf-Datei eingestellt wird.

folgende Elemente enthalten:

- [Programmbeschreibung]
- [Allgemeine Hilfe zur Programmbenutzung]
- [Hilfe für die aktuelle Programmfunktion]
- [Hausinterne Regelungen zur Programmbenutzung]
- [Hilferessourcen im Internet]
- [Datenschutzhinweise]
- [Sicherheitshinweise]
- [Über dieses Programm]

Es ist dabei nicht entscheidend, ob diese Menüpunkte auf unterschiedliche Hilfedateien oder auf eine große Datei und dabei jeweils auf verschiedene Seiten verweisen oder ob es vielleicht zunächst nur einen einzigen Menüpunkt gibt und die Verzweigung eine Eben tiefer beginnt. Wichtig ist, dass die o. g. Inhalte für den Benutzer leicht erkenn- und auffindbar sind.

Die drei letzten Punkte werden im nächsten Fachartikel (OKKSA Newsletter Nr. 25) näher betrachtet.

*Hinweis: Die in der Artikelserie genannten Kriterien stammen aus der aktuellen 4. Ausgabe des "OKKSA Anforderungskatalogs für Fachprogramme in der Öffentlichen Verwaltung - Teilbereich Fachübergreifende Programmanforderungen (FÜ.B)". Im Kriterienkatalog sind neben den Kriterien auch weitere Rechts- und Normungsgrundlagen genannt, aus denen die genannten Kriterien abgeleitet werden.*

### **Nächster Teil der Serie:**

Was muss ein Programm über sich verraten?