

**Inhaltsverzeichnis**

	Seite
1 Problemstellung und Aufbau der Arbeit	1
2 Beschreibung des betrieblichen Anwendungsbereichs der Arbeit	4
2.1 Besonderheiten des DV-Einsatzes in kleinen und mittleren Betrieben	4
2.2 Skizzierung von Klassifikationsmöglichkeiten betrieblicher Anwendungsaufgaben	7
2.3 Eine weitere Klassifikationsmöglichkeit: Datenbearbeitungsaufgaben versus Datenverarbeitungsaufgaben i.e.S.	8
2.3.1 Begründung	8
2.3.2 Standardisierbarkeit der Anwendungsaufgaben	13
3 Das Schnittstellen-Management-System - ein Software-Entwicklungssystem für kleine und mittlere Unternehmen	16
3.1 Die erzeugten Anwendungsprogramme	16
3.1.1 Eine standardisierte Lösung als Grundlage	16
3.1.2 Zur Schnittstellen-orientierten Betrachtungsweise eines Programms	17
3.2 Generelle Funktionsweise	22
3.3 Die Programm-Verarbeitungs-Schnittstelle als Bestandteil des Schnittstellen-Management-Systems	25
3.3.1 Abbilden von Aufgaben der Datenverarbeitung i.e.S.	25
3.3.2 Notwendigkeit eines Werkzeugs zur Definition der Programm-Verarbeitungs-Schnittstelle	26
3.3.3 Stellung der werkzeugunterstützten Definition der Programm-Verarbeitungs-Schnittstelle im Software-Entwicklungsprozeß	30

4	Anforderungen an die Programm-Verarbeitungs-Schnittstelle	35
4.1	Forschungsergebnisse des Software-Engineering zur Gestaltung der Programm-Verarbeitungs-Schnittstelle	35
4.1.1	Begriffsbestimmung: Prinzipien, Methoden, Verfahren und Werkzeuge des Software-Engineering	35
4.1.2	Bedeutung des Software-Engineering für die Programm-Verarbeitungs-Schnittstelle	39
4.1.3	Prinzipien des Software-Engineering zur Gestaltung der Programm-Verarbeitungs-Schnittstelle	40
4.1.3.1	Prinzip der Abstraktion	41
4.1.3.2	Prinzip der Modularisierung	45
4.1.3.3	Prinzip der Strukturierung	48
4.1.3.4	Prinzip der Hierarchisierung	51
4.1.3.5	Prinzip der Standardisierung	53
4.2	Strukturelemente und Struktur von Verarbeitungsfunktionen in der Programm-Verarbeitungs-Schnittstelle	55
4.2.1	Elementare Verarbeitungsfunktionen	55
4.2.1.1	Formen	55
4.2.1.2	Auswahl relevanter Funktionen	56
4.2.2	Sequenz	58
4.2.3	Selektion	58
4.2.3.1	Formen	58
4.2.3.2	Auswahl	64
4.2.4	Iteration	66
4.2.4.1	Formen	66
4.2.4.2	Auswahl	68
4.2.5	Aufbau einer Struktur aus den ausgewählten Strukturelementen	69
4.3	Zusammenfassung der Anforderungen	73

---

5	Gestaltung der Programm-Verarbeitungs-Schnittstelle	75
5.1	Parametrisierung der Programmstruktur in der Programm-Verarbeitungs-Schnittstelle	75
5.2	Parametrisierung der Strukturelemente in der Programm-Verarbeitungs-Schnittstelle	78
5.3	Integrierende Darstellung in einer Dateibeschreibung	84
5.4	Beurteilung der gestalteten Programm-Verarbeitungs-Schnittstelle und weitere Entwicklungsmöglichkeiten	87
6	Konzeption eines endbenutzerorientierten Spezifikationswerkzeugs für die Programm-Verarbeitungs-Schnittstelle	89
6.1	Komponenten des Werkzeugs	89
6.2	Transformationskomponente	91
6.3	Mensch-Maschine-Schnittstelle der Dialogkomponente	94
6.3.1	Allgemeine Grundsätze der Gestaltung	94
6.3.2	Endbenutzer als Zielgruppe	97
6.3.3	Anwendung der allgemeinen Grundsätze auf ausgewählte Dialogfunktionen	99
6.3.3.1	Auswahl relevanter Dialogfunktionen	99
6.3.3.2	Sprachliche Repräsentationsformen von Verarbeitungsfunktionen	101
6.3.3.2.1	Natürliche und formale Sprachen	102
6.3.3.2.2	Prozedurale und nicht-prozedurale Sprachen	105
6.3.3.2.3	Textuelle und graphische Sprachen	112
6.3.3.3	Editieren von Verarbeitungsfunktionen	114
6.4	Zusammenfassung	120

---

7 Realisierung eines endbenutzerorientierten Spezifikationswerkzeugs für die Programm-Verarbeitungs-Schnittstelle	123
7.1 Sprachliche Repräsentation von Verarbeitungsfunktionen	124
7.1.1 Repräsentation der Programmstruktur	124
7.1.2 Repräsentation der Strukturelemente	127
7.2 Editierfunktionen zum Anlegen und Ändern von Verarbeitungsfunktionen	132
7.2.1 Dialogformen der Mensch-Maschine-Schnittstelle	132
7.2.2 Anlegen von Verarbeitungsfunktionen	133
7.2.3 Ändern von Verarbeitungsfunktionen	140
7.3 Integration in das Schnittstellen-Management-System	149
7.4 Beurteilung des Werkzeugs und weitere Entwicklungsmöglichkeiten	151
8 Zusammenfassung und Ausblick	154
Anhang	157
Literaturverzeichnis	158

**Abbildungsverzeichnis**

- Abb. 2-1 Einfache Modellvorstellung betrieblicher Anwendungsaufgaben
- Abb. 3-1 Generelle Funktionsweise des Schnittstellen-Management-Systems
- Abb. 3-2 Stellung der Programm-Verarbeitungs-Schnittstelle zwischen Endbenutzer und Rechner
- Abb. 3-3 Werkzeugunterstützung im Entwicklungsprozeß für Verarbeitungsfunktionen
- Abb. 4-1 Zielstruktur des Software-Engineering in Anlehnung an BOEHM
- Abb. 4-2 Zusammenhang zwischen Prinzipien, Methoden und Verfahren des Software-Engineering
- Abb. 4-3 Zusammenhang der Programm-Verarbeitungs-Schnittstelle mit den Prinzipien des Software-Engineering
- Abb. 4-4 Zusammenhang von Programm-Verarbeitungs-Schnittstelle und Abstraktionsprinzip
- Abb. 4-5 Zusammenhang von Programm-Verarbeitungs-Schnittstelle und Modularisierungsprinzip
- Abb. 4-6 Zusammenhang von Programm-Verarbeitungs-Schnittstelle und Strukturierungsprinzip
- Abb. 4-7 Zusammenhang von Programm-Verarbeitungs-Schnittstelle und Hierarchisierungsprinzip
- Abb. 4-8 Flußdiagramm und Struktogramm zur if-then-else-Form
- Abb. 4-9 Flußdiagramm und Struktogramm zur if-then-Form
- Abb. 4-10 Struktogramm zur if-orif-else-Form
- Abb. 4-11 Flußdiagramm zur posit-else-Form
- Abb. 4-12 Flußdiagramm und Struktogramm zur while-do-Form
- Abb. 4-13 Flußdiagramm und Struktogramm zur repeat-until-Form
- Abb. 4-14 Baumstruktur eines Programms
- Abb. 5-1 Auswertung der Programm-Verarbeitungs-Schnittstelle durch den Programmgenerator
- Abb. 6-1 Komponenten des Spezifikationswerkzeugs
- Abb. 6-2 Bestandteile der Transformationskomponente
- Abb. 6-3 Klassen von Endbenutzern nach MARTIN
- Abb. 6-4 Prozeduralität von Sprachen und Komplexität von Anwendungsproblemen
- Abb. 6-5: Interpretationsmöglichkeiten zweier verschachtelter if-then-else-Selektionen

- Abb. 7-1 Programmzweige im Beispiel Laufzeitberechnung
- Abb. 7-2 Zweiter Zweig im Beispiel Laufzeitberechnung
- Abb. 7-3 Siebter Zweig im Beispiel Laufzeitberechnung
- Abb. 7-4 Anlegen der ersten beiden Programmzweige im Beispiel Laufzeitberechnung
- Abb. 7-5 Automatisches Vorsteuern des "else"-Zweigs
- Abb. 7-6 Mögliche Kommandostruktur für die syntaxgestützte Eingabe von Selektions- und Iterationsbedingungen
- Abb. 7-7 Kommandostruktur zum Navigieren in der Programmstruktur
- Abb. 7-8 Einfügen einer Formel neben einer Selektion im Beispiel Laufzeitberechnung
- Abb. 7-9 Einfügen einer Anwendercopy zwischen zwei Formeln
- Abb. 7-10 Einfügen einer Abstraktionsstufe
- Abb. 7-11 Kommandostruktur zum Verändern einer Verarbeitungsfunktion