

Inhalt

- 1 Einleitung 1**
- 2 Objektorientierte Programmierung: Begriffe und Konzepte 5**
 - 2.1 Klassen und Objekte 5
 - 2.2 Vererbung 6
 - 2.2.1 Abstrakte Klassen 7
 - 2.2.2 Vererbung der Schnittstelle vs. Vererbung Implementation 7
 - 2.3 Dynamische Bindung, Polymorphismus, standardisierte Protokolle 7
 - 2.3.1 Typenprüfung 8
 - 2.4 Weitere Konzepte objektorientierter Programmiersprachen 9
 - 2.5 Wiederverwendung in der objektorientierten Programmierung — Klassenbibliotheken 10
 - 2.5.1 Bausteinklassen 11
 - 2.5.2 Frameworks 11
 - 2.5.3 Application-Frameworks 12
- 3 Struktur und Inhalt der ET++ Klassenbibliothek 15**
 - 3.1 Eigenschaften der ET++ Klassenbibliothek 15
 - 3.1.1 C++ 17
 - 3.2 Die Architektur von ET++ 18
 - 3.3 Systemklassen 20
 - 3.4 Fundamentale Klassen 21
 - 3.4.1 Metainformation 22
 - 3.4.2 Aktivierung/Passivierung von Objekten 25
 - 3.4.3 Abhängigkeiten zwischen Objekten (Change-Propagation) 26
 - 3.5 Datenstruktur-Klassen 27
 - 3.5.1 Die Realisierung generischer Container-Klassen 27
 - 3.5.2 Iteratoren 31
 - 3.5.3 Textklassen 34

- 3.6 Graphische Klassen 36
 - 3.6.1 Graphische Grundklassen 36
 - 3.6.2 Elementare graphische Bausteine 38
 - 3.6.3 Klassen zum Zusammensetzen elementarer Bausteine 38
 - 3.6.4 Komplexere graphische Bausteine 41
 - 3.6.5 Standard-Views 42
 - 3.6.6 Darstellung und Manipulation von Text 43
 - 3.7 Application-Framework-Klassen 45
 - 3.7.1 Die Objekthierarchie einer ET++-Applikation 48
 - 3.7.2 Ereignisbehandlung und Kontrollfluss von ET++ Applikationen 49
 - 3.7.3 Die "Hello World" Beispiel-Applikation 51
 - 3.7.4 ET++ im Vergleich mit MacApp 54
 - 3.8 Beispiele für ET++ Applikationen 55
 - 3.9 Beurteilung der ET++ Klassenbibliothek 56
- 4 ET++PE – Werkzeugunterstützung für die Applikationsentwicklung mit ET++ 57**
- 4.1 Konzepte von ET++PE 58
 - 4.1.1 Browser 58
 - 4.1.2 Exploratives Untersuchen einer Applikation durch Zeigen und Klicken 59
 - 4.1.3 Verflachung der Klassenhierarchie 61
 - 4.1.4 Instrumentierung von Klassen 62
 - 4.2 Die Browser von ET++PE 62
 - 4.2.1 Der Inspector 62
 - 4.2.2 Der Object Structure Browser 65
 - 4.2.3 Der Class Browser 68
 - 4.2.4 Der Class Hierarchy Browser 70
 - 4.3 Integration von ET++PE in einen Debugger 72
 - 4.3.1 Etdgb 73
 - 4.4 Das ET++Cookbook 74
 - 4.4.1 Konzepte 75
 - 4.4.2 Ein Beispiel für die Benutzung des ET++Cookbooks 76
 - 4.4.3 Integration des ET++Cookbooks in ET++PE 77
 - 4.5 Implementationsaspekte 78
 - 4.5.1 Metainformationen 78
 - 4.5.2 Die Benutzerschnittstelle von ET++PE 78
 - 4.5.3 Implementation des Cookbooks 78
 - 4.5.4 Debugger-Anschluss 79
 - 4.6 Beurteilung von ET++PE 80
 - 4.6.1 Mögliche Verbesserungen von ET++PE 80
 - 4.6.2 ET++PE im Vergleich mit anderen Werkzeugen 81

5	Design-Muster – ein Ansatz für die Unterstützung des objektorientierten Entwurfs	83
5.1	Konzepte zur Unterstützung des objektorientierten Entwurfs	84
5.1.1	Ansätze	84
5.1.2	Entwurfsmethoden	85
5.1.3	Stilregeln	87
5.1.4	Frameworks	88
5.2	Konzepte eines neuen Ansatzes für die Unterstützung des objektorientierten Entwurfs	88
5.2.1	Design-Muster (“Design Pattern”)	89
5.2.2	Kategorien von Design-Mustern	90
5.3	Eine graphische Notation für die Darstellung von Design-Mustern	91
5.3.1	Das Klassendiagramm	92
5.3.2	Das Objektdiagramm	93
5.3.3	Das Methoden-Flussdiagramm	94
5.4	Design-Muster für die Organisation von Klassenhierarchien	96
5.4.1	Familien	97
5.4.2	Teams	101
5.4.3	Subsysteme	103
5.4.4	Frameworks	106
5.4.5	Die globale Organisation einer Klassenbibliothek	113
5.5	Design-Muster für die Gestaltung der Interaktion zwischen Klassen (Interaktions-Design)	116
5.5.1	Black-Box- und Parameterklassen	117
5.5.2	Kompositionsklassen (“Composer”)	119
5.5.3	Umwickler (“Wrapper”)	120
5.5.4	Brücken- (“Bridges”) und Implementationsklassen	123
5.5.5	Delegationsklassen (“Forwarder”)	127
5.5.6	Vermittler (“Mediator”)	131
5.5.7	Verfahrensklassen (“Behaviours”)	135
5.5.8	Strategieklassen (“Policies”)	138
5.5.9	Adapter	142
5.6	Design-Muster für die Gestaltung der Klassenschnittstelle	144
5.6.1	Allgemeine Grundregeln	144
5.6.2	Elementare und zusammengesetzte Methoden	145
5.6.3	Virtuelle Konstruktoren	146
5.6.4	Standardmethoden (“Defaults”)	149
5.6.5	Sequenz- und Einschubmethoden	149
5.6.6	Iteratoren	150
5.6.7	Double-Dispatching	151
5.7	Muster für die Reorganisation von Klassenhierarchien	152
5.7.1	Vertikale Reorganisationen	153
5.7.2	Horizontale Reorganisationen	156
5.7.3	Klasseninterne Reorganisationen	159
5.7.4	Die Evolution von ET++	160
5.8	Allgemeine Anmerkungen zum objektorientierten Entwurf	163
5.8.1	Anwendung der Design-Muster	163

5.8.2	Verwendung von Vererbung	165
5.9	Design-Muster und Werkzeugunterstützung	167
5.9.1	ET++DE	167
6	Zusammenfassung und Schlussbemerkungen	171
6.1	Zusammenfassung	171
6.2	Schlussbemerkungen	172
	Literaturverzeichnis	175
	Index	187