

Obsah

OBSAH	3
1 OBJEKTIVĚ ORIENTOVANÉ PARADIGMA	9
1.1 Softwarová krize.....	9
1.1.1 Současný stav.....	9
1.1.2 Historické souvislosti	9
1.2 Objektově orientovaná architektura výpočetního systému	10
1.2.1 Architektura počítače.....	10
1.2.2 von Neumannova architektura	11
1.2.3 Sémantická mezera.....	12
1.2.4 Metody překonávání sémantické mezery.....	14
1.2.4.1 Jiná konstrukce počítače	14
1.2.4.2 Zdokonalení v oblasti softwaru	15
1.2.5 Sémanticky strukturovaná paměť.....	15
1.3 Současné trendy rozvoje softwaru	16
1.3.1 Vizualní přístup.....	16
1.3.2 Objektově orientované operační systémy, počítačové agenti	17
1.3.2.1 institucionální počítač	18
1.3.2.2 osobní počítač	19
1.3.2.3 intimní počítač	19
1.4 Objektový software.....	19
1.4.1 Objekt jako abstraktní datový typ	20
1.4.2 Struktura objektově orientovaného programu	21
1.4.3 Možnosti implementace objektově orientovaného jazyka	21
1.5 Objektově orientované programovací jazyky	22
1.5.1 Vhodný programovací jazyk - Smalltalk	23
1.5.1.1 historie Smalltalku	23
1.5.1.2 Smalltalk dnes	24
1.5.1.3 současné systémy Smalltalk	25
1.6 Počítačové databáze	26
1.6.1 Současný stav.....	26
1.6.2 Vztah relační a objektové technologie.....	27
1.6.3 Současné trendy v databázové technologii	28
1.7 Programovací styly	29
1.7.1 Naivní technologie	30
1.7.2 Klasické - imperativní technologie.....	30
1.7.3 Strukturovaná technologie	30
1.7.4 Procedurální technologie	30
1.7.5 Modulární technologie.....	31
1.7.6 Neimperativní technologie.....	31
1.7.7 Funkcionální programování	31
1.7.8 Logické programování.....	31
1.7.9 Objektová technologie.....	32
1.7.10 Komponentové programování.....	32
1.8 Základní principy objektově orientovaného přístupu.....	32
1.8.1 Model výpočtu	33
1.8.2 Polymorfismus	34
1.8.3 Skládání objektů	35
1.8.4 Třídy objektů.....	35
1.8.5 Dědění mezi objekty.....	36
1.8.6 Dědění versus skládání	36
1.8.7 Dědění mezi třídami	37

1.8.8 Delegování, alternativní aktorový model	38
1.8.9 Závislost mezi objekty.....	39
2 ÚVOD DO SMALLTALKU.....	41
2.1 První seznámení s Smalltalkem	41
2.1.1 Instalace Systému VisualWorks.....	42
2.1.1.1 Microsoft Windows	42
2.1.1.2 Unix/Linux.....	42
2.1.2 Práce se systémem.....	42
2.2 Objekty a zprávy.....	44
2.3 Uživatelské prostředí	46
2.4 Jazyk Smalltalku.....	47
2.4.1 Jména objektů	48
2.4.2 Konstanty.....	48
2.4.2.1 Číselné konstanty	49
2.4.2.2 Znakové konstanty.....	49
2.4.2.3 Řetězcové konstanty.....	49
2.4.2.4 Konstanty symbol.....	49
2.4.2.5 Bajtová pole	50
2.4.2.6 Pole jiných konstant	50
2.4.3 Proměnné.....	50
2.4.4 Výrazy	51
2.4.4.1 Zprávy.....	51
2.4.4.2 Sekvence výrazů.....	52
2.4.4.3 Kaskáda zpráv.....	53
2.4.4.4 Návrátový výraz.....	53
2.4.4.5 Použití pomocných proměnných.....	53
2.4.4.6 Bloky výrazů.....	54
2.4.5 Jednoduchý program	55
2.5 Instance, třídy, metatřídy a metody	56
2.5.1 Instanční proměnné	58
2.5.2 Instanční proměnné tříd.....	59
2.5.3 NameSpaces a ostatní proměnné ve Smalltalku.....	59
2.5.4 Architektura třídního systému.....	61
2.5.5 Polymorfismus a generický kód.....	63
2.5.6 Programování nových tříd a metod	64
2.6 Systém tříd ve Smalltalku	68
2.6.1 Systém tříd Collection.....	68
2.6.1.1 Příklady práce s instancemi systému Collection	71
2.6.2 Systém tříd Magnitude	72
2.6.3 Systém tříd Stream.....	74
2.6.3.1 Příklady použití	75
2.7 Řízení výpočtu ve Smalltalku.....	76
2.7.1 Větvení	76
2.7.2 Iterace.....	77
2.7.3 Operace nad sadami (Collection)	77
2.7.3.1 Příklad použití metod z knihovny Collection - asSet	78
2.7.3.2 Příklad použití metod z knihovny Collection - asBag, occurrencesOf:	79
2.7.3.3 Příklad použití metod z knihovny Collection - select:, includes:.....	79
2.7.3.4 Příklad použití metod z knihovny Collection - inject:into:.....	79
2.7.4 Úloha polymorfismu v objektové optimalizaci	80
2.7.4.1 Využití při návrhu nových objektových komponent – vzor double dispatching.....	80
2.7.4.2 Úloha datového modelování a polymorfismu při návrhu objektových algoritmů.....	82
2.8 Ošetřování chybových stavů	84
2.8.1 Mechanismus výjimek.....	84
2.8.1.1 Zachytávání výjimek	85
2.8.1.2 Vytváření vlastních výjimek a vyvolávání výjimek	85

2.8.2	Závislost objektů	87
2.8.2.1	Malé příklady využití závislosti objektů	89
2.8.3	Architektura MVC	91
2.8.4	Komponenty grafického uživatelského rozhraní (GUI)	93
2.8.4.1	Dialogy	93
2.9	Paralelismus	94
2.9.1	Koordinace paralelních procesů	95
2.9.2	Malé příklady paralelních procesů	95
3	POKROČILÉ TECHNIKY TVORBY SOFTWARE	97
3.1	Vztah mezi informačním a řídicím systémem uvnitř organizace	97
3.2	Modelování požadavků na informační systémy	99
3.2.1	Myšlenka konvergenčního inženýrství	99
3.3	Životní cyklus vývoje informačního systému	100
3.3.1	Objektová analýza a návrh	100
3.3.2	Současné objektové metodologie	102
3.4	Řízení vývojového týmu	103
3.4.1	Softwarové profese	104
3.4.2	Organizace pracovních týmů	104
3.4.3	Algoritmizace rozpočtu	105
3.5	Alternativní metody řízení projektů	107
3.5.1	Vývoj pohledu na softwarové projekty	107
3.5.2	Tradiční přístup k řízení	107
3.5.2.1	Omezení tradičního přístupu	108
3.5.2.2	Rozsah použití	108
3.5.2.3	Požadavky na vývojové nástroje	108
3.5.3	Alternativní metody řízení	108
3.5.3.1	Řízení projektů metodou extrémního programování	109
3.5.3.2	Omezení metody XP	110
3.5.3.3	Rozsah použití	110
3.6	Návrhové vzory	110
3.6.1	Co to je návrhový vzor	111
3.6.2	Jak se návrhový vzor popisuje	112
3.6.3	Příklad návrhového vzoru Composite	112
3.6.3.1	Účel	112
3.6.3.2	Použití	112
3.6.3.3	Struktura	113
3.6.3.4	Součásti	113
3.6.3.5	Spolupráce	113
3.6.3.6	Důsledky	114
3.6.4	Známa použití	114
3.6.5	Příbuzné vzory	114
3.7	Softwarové metriky a metoda funkčních bodů	114
3.8	Techniky ladění programů	116
3.8.1	Úvod	116
3.8.2	Hledání místa vzniku chyby	116
3.8.2.1	Vizuální kontrola zdrojového kódu	117
3.8.2.2	Kontrolní výpisy	117
3.8.2.3	Prohlížení objektů za běhu – Inspector	117
3.8.2.4	Umělé zastavení programu	118
3.8.3	Debugger	118
3.8.3.1	Analýza výjimek z uživatelského rozhraní	119
3.8.4	Předcházení chybám	120
3.9	Úvod do refaktoringu	120
3.9.1	Základní refaktoriizační úkony	121
3.9.2	Refaktoriizace v prostředí VisualWorks	121
3.10	Problémy objektového návrhu	121

3.10.1 Dědění, hierarchie typů a taxonomie nejsou vždy totéž	122
3.10.2 Úspora za každou cenu	124
3.10.3 Vícenásobná dědičnost?	124
3.10.4 Třídy versus množiny objektů	125
3.10.5 Podtřídy nebo instance?	126
4 OBJEKTOVĚ ORIENTOVANÉ DATABÁZE – GEMSTONE	127
4.1 Úvod	127
4.2 Objektově orientované a objektově relační databáze	127
4.3 Objektově orientovaný datový model	129
4.4 Jak vytvořit objektovou databázovou aplikaci	130
4.5 Gemstone	131
4.5.1 Historie Gemstone	131
4.5.2 Vlastnosti Gemstone	131
4.6 Programovací jazyk Smalltalk DB	132
4.6.1 Architektura programů ve Smalltalku	133
4.6.2 Rozdíly mezi Smalltalkem-80 a Smalltalkem DB	133
4.7 Příklad objektové databáze	134
4.7.1 Popis úlohy	134
4.7.2 Implementace úlohy	134
4.7.3 Program v jazyce Smalltalk DB databázového systému Gemstone	135
4.8 Příklady dotazů	138
4.9 Shrnutí	139
5 OBJEKTOVĚ RELAČNÍ MAPOVÁNÍ – OBJECTLENS	140
5.1 Transformace relačního datového modelu do objektového	140
5.1.1 Tvorba tříd objektů na základě entitních množin (tabulek)	141
5.1.2 Úprava relačního schématu	141
5.1.3 Vytvoření vazeb skládání mezi objekty na základě klíčů v tabulkách	142
5.1.4 Návrh pohledů nad objektovou strukturou databáze a konstrukce rozhraní	144
5.2 Zhodnocení vlastností objektově relačního přístupu ObjectLens	145
6 DISTRIBUOVANÉ OBJEKTY	147
6.1 Object Request Broker (ORB)	147
6.2 Existující standardy distribuovaných objektů	147
6.3 CORBA	148
6.3.1 Základní struktura	148
6.3.2 Interface Definition Language	148
6.3.3 Rozhraní CORBA	149
6.3.4 Object Services	150
6.4 Použití distribuovaných objektů ve Smalltalku	150
7 JAZYK UML	152
7.1 Úvod	152
7.2 Vznik UML	152
7.3 Diagramy UML	152
7.3.1 Použití diagramů UML	153
7.4 Vybrané diagramy UML	153
7.4.1 Diagram tříd	154
7.4.2 Diagram aktivit	154
7.4.3 Sekvenční diagram	155
7.4.4 Diagram spolupráce	156
7.4.5 Diagram případů užití	156
7.4.6 Stavový diagram	157
8 SLOVNÍČEK POJMŮ	158
9 POUŽITÁ A DOPORUČENÁ LITERATURA	168