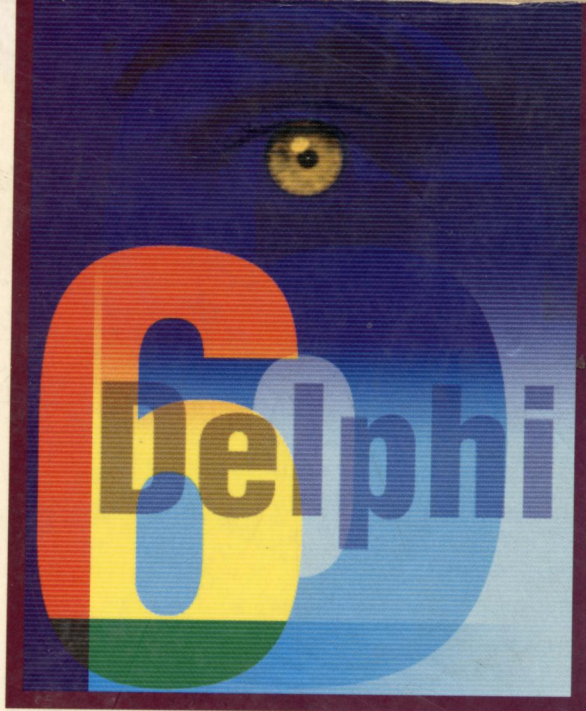


DÉVELOPPEMENT



Steve Teixeira et Xavier Pacheco

Borland®

Delphi 6

-1

Collection
Référence



En cadeau ! Un CD-ROM contenant de nombreux outils complémentaires (en version de démonstration) et tous les codes sources de l'ouvrage.


CampusPress

www.pearsoneducation.fr



2-005-474-1

2-005-474-1

Table des matières

Delphi™ 6

Steve Teixeira, Xavier Pacheco



CAMPUSPRESS

Table des matières

Avant-propos de David Intersimone	XIX
Introduction	1
A qui s'adresse ce livre ?	1
Structure du livre	2
Conventions typographiques	2
Contenu du CD d'accompagnement	3
Par où commencer ?	3
Avant-propos de l'édition française	3
Exemples	3
Remerciements	4

Partie I

Principes de développement

Chapitre 1. Approche globale de la programmation Delphi	7
La famille Delphi	8
Delphi 6 Personal	8
Delphi 6 Professional	9
Delphi 6 Enterprise	10
Delphi : ses avantages remarquables et leurs fondements	10
Grande qualité de l'environnement de développement intégré EDI	11
Rapidité du compilateur sans conséquence pour l'efficacité du code exécutable généré	13
Puissance du langage sans complexité inutile	14
Souplesse et évolutivité de l'architecture de bases de données	15
Conditions de réutilisation de composants permises par l'infrastructure	16

Bref historique de Delphi	16
Delphi 1	16
Delphi 2	17
Delphi 3	18
Delphi 4	19
Delphi 5	20
Delphi 6	20
L'environnement EDI de Delphi	21
La fenêtre principale	22
Le concepteur de fiche	23
L'Inspecteur d'objets	24
L'éditeur de code	25
L'explorateur de code	25
L'arborescence des objets	25
Petite visite guidée d'un texte source Pascal	26
Exemple de conception d'une application	28
Les événements, c'est fantastique !	30
Une programmation non contraignante	30
Turbo prototypage	31
Composants personnalisés et environnement évolutif	32
Le Top 10 des caractéristiques de l'EDI à connaître	32
1. Assistance à la saisie du code (<i>class completion</i>)	32
2. Navigateur AppBrowser	33
3. Navigation entre Interface et Implementation	33
4. Ancrage des éléments !	33
5. Explorateur de projet (d'objets)	34
6. Insertion d'un identificateur Guid	34
7. Coloration syntaxique de type C++	35
8. Liste de choses à faire	35
9. Gestionnaire de projet	35
10. Outil Code Insight pour insérer les déclarations et les paramètres	36
Résumé	36
Chapitre 2. Langage Pascal Objet	37
Commentaires	38
Les caractéristiques évoluées des procédures et fonctions	39
Parenthèses dans les appels	39

Surcharge des fonctions	40
Paramètres par valeur implicites	40
Les variables	41
Les constantes	43
Les opérateurs	45
Opérateurs d'affectation	45
Opérateurs de comparaison	45
Opérateurs logiques	46
Opérateurs arithmétiques	47
Opérateurs binaires	48
Procédures d'incrémentatation et de décrémentatation	48
Opérateurs hybrides	49
Les types de données du Pascal Objet	49
Comparaison des types de données	50
Les caractères	52
Une foultitude de chaînes	52
Opérations sur les chaînes	56
Compatibilité avec Win32	57
Considérations de portabilité	58
Le type fourre-tout <i>Variant</i>	64
Le type monétaire <i>Currency</i>	76
Les types définis par le programmeur	76
Les tableaux <i>Array</i>	76
Tableaux dynamiques	77
Enregistrements (<i>structures record</i>)	79
Ensembles (<i>sets</i>)	80
Objets	82
Pointeurs	83
Les alias de types	86
Le transtypage et la conversion de type	87
Les chaînes littérales de ressources	88
Les instructions de test	89
L'instruction de test <i>if</i>	89
L'instruction de test <i>case</i>	90
Les boucles d'exécution répétées	90
La boucle <i>for</i>	91
La boucle <i>while</i>	91

La boucle <i>repeat..until</i>	92
La procédure de sortie <i>Break()</i>	92
La procédure <i>Continue()</i>	93
Les procédures et fonctions	93
Passage des paramètres en Pascal	94
La portée des variables et des routines	98
Les unités	99
La clause <i>uses</i>	100
Références circulaires entre unités	101
Les paquetages (<i>packages</i>)	101
Mode d'emploi d'un paquetage Delphi	102
Syntaxe des paquetages	102
La programmation orientée objets	103
Programmation orientée ou basée objets ?	105
Utilisation des objets Delphi	105
Déclaration et construction d'un objet	106
Destruction d'une instance	107
Méthodes	107
Types de méthodes	108
Propriétés des objets	111
Spécificateurs de visibilité (de portée)	112
Implantation mémoire des objets	113
TObject : l'ancêtre ultime	114
Interfaces d'objets	115
Gestion structurée des exceptions	119
Classes d'exceptions	122
Flux d'exécution	124
Redéclenchement d'une exception	126
Les informations de type à l'exécution (RTTI)	126
Résumé	128
Chapitre 3. Voyage parmi les messages système	129
Qu'est-ce qu'un message ?	130
Types de messages	131
Fonctionnement du mécanisme de messages de Windows	132

Gestion des messages de Delphi	133
Structures record spécifiques à certains messages	134
Traitement des messages	135
Contractualisation du traitement des messages	137
Affectation d'une valeur de retour de message	138
L'événement <i>OnMessage</i> de <i>TApplication</i>	138
Emission de messages personnalisés	139
La méthode <i>Perform()</i>	140
Les fonctions <i>SendMessage()</i> et <i>PostMessage()</i>	140
Messages non standard	141
Messages de notification	141
Messages internes à la VCL	143
Messages définis par le programmeur	143
Anatomie d'un mécanisme de messages : celui de la VCL	146
Relations entre messages et événements	152
Résumé	153

Partie II

Techniques évoluées

Chapitre 4. Ecriture de code portable	157
Considérations générales de compatibilité	158
Quelle version de Delphi ?	158
Unités, composants et paquetages	159
Considérations relatives à l'environnement EDI	160
La compatibilité Delphi/Kylix	161
Eléments absents sous Linux	161
Caractéristiques du compilateur et du langage	162
Nouveautés de Delphi 6	163
Prise en charge des variants	163
Valeurs énumérées	164
Directive <i>\$IF</i>	164
Incompatibilité possible des fichiers DFM	164
Migration depuis Delphi 5	164
Constantes typées modifiables	164
Négation unaire cardinale	165

Migration depuis Delphi 4	165
Considérations relatives à la librairie RTL	165
Considérations relatives à la librairie VCL	165
Considérations relatives au développement Internet	166
Considérations relatives aux bases de données	166
Migration depuis Delphi 3	167
Entiers non signés sur 32 bits	167
Entiers sur 64 bits	168
Le type historique <i>Real</i>	168
Migration depuis Delphi 2	168
Evolution du type <i>Boolean</i>	169
Chaînes de ressources <i>ResourceString</i>	169
Modifications de la librairie d'exécution RTL	170
La classe <i>TCustomForm</i>	170
La procédure <i>GetChildren()</i>	170
Serveurs d'automation	171
Migration depuis Delphi 1	171
Résumé	171
Chapitre 5. Techniques d'exécution multithread	173
Principes des exétons (<i>threads</i>)	173
Multitâche préemptif ou coopératif ?	174
Utilisation des exétons dans les applications Delphi	174
Limitations des exétons	175
L'objet de base <i>TThread</i>	176
Les principes de <i>TThread</i>	176
Les instances d'exétons	179
Fin d'exécution d'un exétron	179
Synchronisation avec la librairie VCL	181
Une application de démonstration	184
Priorités et planification	185
Suspension et reprise d'un exétron	188
Contrôle temporel d'un exétron	189
Gestion d'exétons multiples	191
Stockage local à l'exétron	191
Synchronisation des exétons	195
Un exemple complet d'application multithread	206

L'interface utilisateur	207
Réglage de la priorité	219
Accès multithread au moteur BDE	221
Traitements graphiques multithread	227
Fibres (sous-exétons)	231
Résumé	237
Chapitre 6. Librairies ("bibliothèques") DLL	239
Constitution d'une DLL	240
Liaison statique ou liaison dynamique	243
Retour sur les avantages des DLL	244
Partage de code, de ressources et de données entre plusieurs applications	244
Masquage des détails de l'implémentation	245
Création et exploitation d'une DLL	246
Conception d'une DLL minimale	246
Affichage d'une fiche modale <i>via</i> une DLL	249
Affichage d'une fiche non modale intégrée à une DLL	251
Utilisation des DLL dans les applications Delphi	253
Chargement explicite d'une DLL	255
La fonction d'entrée et de sortie de DLL	257
Routine d'initialisation et de purge de processus ou d'exétron	258
Un exemple d'entrée et de sortie de DLL	259
Les exceptions et les DLL	262
Interception d'une exception en Delphi 16 bits	262
Les exceptions et la directive safecall	264
Fonctions de rappel (<i>callback</i>)	264
La fonction de rappel	267
Affichage d'une liste personnalisée	267
Appel d'une fonction de rappel depuis une DLL	268
Partage des données d'une DLL entre plusieurs processus	270
Création d'une DLL avec une zone de mémoire partagée	271
Utilisation d'une DLL avec la mémoire partagée	274
Exportation d'un objet depuis une DLL	277
Résumé	282

Partie III

Exploitation des bases de données

Chapitre 7. Architectures d'accès aux bases de données	285
Les modèles d'accès aux bases	285
Architecture d'accès aux données Delphi	286
Connexion à un serveur de bases de données	287
Principes de connexion à une base	287
Etablissement d'une connexion à une base	288
Exploitation d'un jeu de données	288
Ouverture et fermeture d'un jeu de données	289
Navigation dans un jeu de données	292
Les propriétés BOF et EOF et les boucles	292
Manipulation d'un jeu de données	297
Statuts d'un jeu de données	300
Exploitation des champs	301
Valeurs des champs	302
Types de données des champs	303
Noms et numéros des champs	303
Manipulation des données des champs	304
L'outil d'édition de champs	305
Exploitation des champs BLOB	310
Filtrage des données	316
Recherche dans un jeu de données	318
Utilisation de modules de données	322
La démonstration SRF (<i>Search, Range, Filter</i>)	322
Signets de données (<i>bookmarks</i>)	331
Résumé	331
Chapitre 8. Développement pour bases de données avec dbExpress	333
Présentation de dbExpress	334
Jeux de données unidirectionnels (en lecture seule)	334
dbExpress comparé au moteur Borland Database Engine (BDE)	334
dbExpress et le développement multi-plate-forme	335
Composants dbExpress	335
TSQLConnection	335

Le composant <i>TSQLDataset</i>	338
Composants de compatibilité	342
Le composant <i>TSQLMonitor</i>	342
Conception d'applications dbExpress modifiables	343
Le composant <i>TSQLClientDataset</i>	343
Déploiement d'une application dbExpress	344
Résumé	345
Chapitre 9. Développements pour bases de données avec dbGo for ADO	347
Présentation de dbGo	347
La stratégie Universal Data Access de Microsoft	347
Principes de OLE DB, ADO et ODBC	348
Utilisation de dbGo for ADO	348
Définition d'un fournisseur OLE DB pour ODBC	348
La base de démonstration Access	350
Les composants dbGo for ADO	350
Le composant <i>TADOConnection</i>	351
Suppression ou remplacement de la boîte d'identification d'accès	353
Le composant <i>TADOCommand</i>	355
Le composant <i>TADODataset</i>	356
Composants de jeux de données de style BDE	356
Traitement de transactions	358
Résumé	359

Partie IV

Conception de composants

Chapitre 10. Architecture des composants : VCL et CLX	363
La nouvelle librairie CLX	365
Les principes des composants	365
La hiérarchie des composants	366
Les composants non visuels	367
Les composants visuels	367
La structure d'un composant	369
Les propriétés des composants	370
Types de propriétés	372

Méthodes	372
Événements	373
Support des flux (<i>Streams</i>)	375
Notion de possession (<i>Ownership</i>)	375
Parenté ou tutorat	376
La hiérarchie des composants visuels	376
La classe <i>TPersistent</i>	377
Les méthodes de <i>TPersistent</i>	377
La classe <i>TComponent</i>	378
La classe <i>TControl</i>	380
Les composants <i>TWinControl</i> et <i>TWidgetControl</i>	381
La classe <i>TGraphicControl</i>	382
La classe <i>TCustomControl</i>	383
Autres classes	383
Les informations de type à l'exécution (RTTI)	387
L'unité <i>TypInfo.pas</i> pour définir les informations RTTI	388
Récupération d'informations RTTI	390
Récupération d'informations RTTI pour des pointeurs de méthode	398
Récupération d'informations RTTI pour les types ordinaux	402
Affectation d'une valeur à une propriété via RTTI	408
Résumé	410
Chapitre 11. Conception de composants VCL	411
Principe de la conception des composants	412
Concevoir ou ne pas concevoir ?	412
Étapes de la conception d'un composant	413
Constructeurs et destructeurs	433
Recensement du composant	435
Test du composant	437
Choix d'une icône de composant	439
Quelques exemples de composants	439
Extension des capacités d'habillage des composants Win32	440
Création de propriétés (<i>TddgRunButton</i>)	449
Un composant conteneur (<i>TddgButtonEdit</i>)	455
Réflexion préalable de conception	455
Propagation en surface de propriétés internes	456

Propagation en surface des événements	457
Ajout d'une fiche à la palette de composants	462
Résumé	465
Chapitre 12. Conception de composants VCL sophistiqués	467
Composants pseudo-visuels	467
Composants d'animation	472
Conception d'un éditeur de propriété	486
Editeurs de composant	498
Mise en flux de données de composants non publiées	503
Catégories de propriétés	512
Listes de composants (<i>TCollection</i> et <i>TCollectionItem</i>)	518
Résumé	534
Chapitre 13. Développement de composants CLX	535
Principes de CLX	535
Architecture CLX	537
Considérations de portabilité	540
Disparition des messages	541
Quelques exemples de composants	541
Le composant <i>TddgSpinner</i>	542
Améliorations pour la phase de conception	554
Association du composant à une liste <i>ImageList</i>	560
Composants CLX à liaison de données	566
Editeurs de composants CLX	574
Paquetages	578
Conventions de nommage	578
Paquetages d'exécution	580
Paquetages de conception	583
Unités de recensement	585
Images binaires et icônes de composants	586
Résumé	588
Chapitre 14. Paquets Delphi	589
Domaines d'emploi des paquets	590
Réduction de la taille du code	590
Modularisation des applications (partitionnement)	591
Rassemblement de composants	591

Contre-indications par rapport aux paquets	591
Types de paquets	592
Fichiers des paquets	592
Utilisation des paquets d'exécution	592
Emplacement des paquets dans l'EDI de Delphi	593
Création d'un paquet	594
L'éditeur de paquets	595
Scénarios d'organisation de paquets	596
Gestion des versions des paquets	600
Directives de compilation des paquets	600
La directive {\$WEAKPACKAGEUNIT}	601
Conventions de nommage des paquets	602
Applications extensibles <i>via</i> des paquets d'exécution optionnels	602
Exemple de gestion de fiches optionnelles	602
Exportation de fonctions logées dans un paquet	609
Appel à une fonction qui réside dans un paquet	609
Récupération d'informations de paquets	612
Résumé	615
Chapitre 15. Développements COM	617
Principes de COM	618
COM : Component Object Model	618
COM comparé à ActiveX et à OLE	618
Terminologie COM	619
Les avantages de ActiveX	620
OLE 1 ou OLE 2	620
Stockage structuré	621
Transferts de données uniformes (UDT)	621
Modalités d'exécution comme exétons (thread)	621
L'évolution COM+	622
Relations entre COM et le Pascal Objet	622
Utilisation des interfaces	625
Le type de valeur renvoyé HRESULT	630
Les objets COM et les fabriques de classes	630
<i>TComObject</i> et <i>TComObjectFactory</i>	631
Les serveurs COM en-processus	632
Les serveurs COM hors-processus	635

Agrégations	636
COM distribué	636
Automation	637
L'interface <i>IDispatch</i>	637
Informations de type	639
Liaison tardive ou précoce	639
Inscription dans la base de registre	640
Création d'un serveur d'automation	640
Création d'un contrôleur d'automation	655
Techniques d'automation avancées	662
Evénements d'automation	663
Collections d'automation	675
Nouveaux types d'interface dans la librairie de types	685
Echanges de données binaires	686
Les coulisses du support COM dans Delphi	689
Les <i>Variants</i>	689
Les chaînes <i>WideString</i>	692
Interfaces	692
TOLeContainer	694
Une petite application d'exemple	695
Une application plus complète	697
Résumé	706
Chapitre 16. Applications transactionnelles avec COM+/MTS	707
Principes de COM+	707
Pourquoi avoir conservé une base COM ?	708
Services COM+	709
Les transactions COM+	709
La sécurité dans COM+	710
L'activation juste-à-temps (JIT)	716
Les composants mis en file	716
Partage d'objets (<i>pooling</i>)	725
Les événements de COM+	726
L'environnement d'exécution de COM+	735
La base de données de recensement (<i>RegDB</i>)	735
Les composants configurés	736

Les contextes	736
Exétons neutres	736
Création des applications COM+	736
Objectif premier : l'évolutivité	737
Contexte d'exécution	737
Avec ou sans maintien de l'état	738
Gestion de la durée de vie	739
Structure des applications COM+	740
A propos des transactions	740
Les ressources COM+	741
COM+ dans Delphi	741
Les assistants COM+	741
L'infrastructure COM+ et la classe <i>TMtsAutoObject</i>	743
Une application d'exemple : un jeu de morpions COM+	745
L'application serveur COM+ (TTTServer)	747
Débogage des applications COM+	762
Résumé	763

Partie V

Développement Internet

Chapitre 17. Développement ASP	767
Principes des objets ASP	767
Active Server Pages	768
L'assistant de création d'objets ASP	770
L'éditeur de librairie de types	773
L'objet ASP Response	777
Première exécution de l'exemple	778
L'objet ASP Request	779
Recompilation des objets ASP	780
Exécution d'une page ASP depuis une page HTML	781
Les objets ASP Session, Server et Application	783
Objets ASP et bases de données	784
Objets ASP et support de NetCLX	787
Débogage d'objets ASP	789
Débogage des objets ASP avec COM+ (ou MTS)	790

Débogage sous Windows NT 4 avec MTS	791
Débogage sous Windows 2000 avec COM+	793
Résumé	795
Chapitre 18. Conception d'applications WebSnap	797
Caractéristiques de WebSnap	798
Modules Web multiples	798
Scripts du côté serveur	798
Composants <i>TAdapter</i>	798
Méthodes de distribution multiples	799
Composants générateurs de page	799
Gestion des sessions	799
Services de connexion	799
Suivi des visiteurs	800
Administration des fichiers HTML	800
Services de téléchargement	800
Création d'une application WebSnap	800
Création du cœur de l'application	801
Enrichissement fonctionnel de l'application	809
Test de la barre de navigation (<i>wmSimple</i>)	810
Page de réussite de connexion (<i>wmLoggedIn</i>)	811
Page de connexion (<i>wmLogin</i>)	812
Gestion des préférences des visiteurs	815
Mémorisation des préférences entre sessions	819
Affichage d'images	821
Affichage de données d'une base	823
Conversion d'une application Web au format ISAPI DLL	828
Autres fonctions	829
LocateFileServices	829
Téléchargement de fichiers	831
Exploitation de modèles de pages personnalisés	833
Composants spécifiques dans <i>TAdapterPageProducer</i>	833
Résumé	836
Index	837