

INDICE DE CONTENIDO

PREFACIO	1
Enlazadores para modo protegido	5
(Marino Posadas)	
Introducción	6
Fundamentos del modo protegido	7
El direccionamiento en modo real	8
Memoria extendida (El estándar XMS)	10
El modo protegido: una consecuencia de la multitarea	11
Cambios básicos en el hardware del 286	13
Conmutación a modo protegido	14
Selectores y tablas de descriptores global y local	16
<i>DOS Extenders</i> (extensores del DOS)	21
Funcionamiento de los extensores del DOS	22
Los estándares del modo protegido: XMS, VCPI y DPMI	23
Convirtiendo el código de modo real a modo protegido	24
La programación en modo protegido	26
El decálogo de codificación	27
Circunstancias típicas que generan GPF (<i>General Protection Faults</i>)	28
Blinker 3.x, CauseWay y ExoSpace	30

Ejemplos	31
Clipper, C y el modo protegido	41
Evaluando rendimientos	47
Blinker 3	47
CauseWay	48
ExoSpace	48
Evaluación de rendimientos en ejecución	49
Bibliografía	50
Migración de Clipper a Visual Objects	53
(Helio Yago)	
Aclarando conceptos	54
Introducción	54
Qué se entiende por migración	54
Un poco de Windows	55
La primera mirada a VO	56
IDE	56
Repositorio	56
Generando una aplicación	59
"Hola Mundo" con VO	63
Introducción	63
El editor de menús	63
Nuestro primer evento	66
En el emulador de terminal	66
Otra aplicación en modo texto	67
Empezando a migrar	68
Añadiendo funcionalidad	68
Cómo migrar	71

<i>EXPORT/IMPORT</i> del Repositorio	71
<i>Start()</i>	72
Integrar el módulo importado	72
Visualizando una DBF	73
Peculiaridades del modo terminal	75
Variables y Symbols	76
Optimizando	77
Migración y preprocesador	81
Conclusión	83
En la aplicación estándar	84
<i>Start()</i>	84
<i>Init()</i> de la clase <i>StandardShellWindow</i>	86
<i>Init()</i> de la clase <i>StdDataWindow</i>	87
Moverse por el IDE	89
El editor de ventanas	90
Editor de impresión	96
Editor de iconos	97
<i>DBServer</i> Editor	97
<i>FieldSpec</i> Editor	97
Reflexiones finales	97
Qué se puede hacer	98
Conclusión	99
El depurador de Visual Objects	101
(Vladimir Algara)	
Introducción	102
Errores de compilación	102
Errores de ejecución	105
Comportamientos no esperados	105

Comienzo de la aplicación	106
Menú local de módulos y entidades	108
Pasos habituales en la depuración	112
Ejemplo	122
Conclusión	125
La gestión de bases de datos en Visual Objects	127
(Antonio Quirós)	
Introducción	128
Visual Objects y los datos	128
La jerarquía de clases en VO	130
Clases DBF	130
Clases SQL	132
Los editores de datos	136
El editor de FieldSpec	139
El editor de DBServer	144
El editor de DataWindows	145
El editor de menús y otros	148
Compilando y ejecutando nuestra aplicación	151
ODBC - SQL	152
Manejo simple de datos con SQL	153
Y ahora sin editores	157
OOP versus NO-OOP	161
Conclusión: VO y la paradoja de los datos con y sin OOP	167

Migración de Clipper a Windows (John Skelton)	169
Introducción	170
Un poco de historia	170
El interfaz gráfico de usuario y la programación conducida por eventos	171
Comportamiento Modal y No Modal	172
Eventos en Windows	172
Un ejemplo	173
Notificación de eventos	174
Gestión de eventos	175
Conversión del software Clipper existente	175
El interfaz de usuario	176
Programación en Windows - El API de Windows	180
Conclusión	180
Las interioridades de Clipper	181
(Manu Roibal Prieto)	
Agradecimientos	182
Introducción	182
La gestión de memoria. Módulo VM	186
Gestión de eventos. Módulo Event	190

Gestión de ficheros. Módulo TSUPPORT	195
Conclusión	198
Clipper y el sistema de manejo de colas en Novell	199
(Javier Limones)	
Introducción	200
¿Qué es exactamente el sistema de manejo de colas de Novell (QMS)?	201
Partes de un sistema de colas	204
Creación y mantenimiento	204
Mandar trabajos a la cola	207
Despachando trabajos en la cola	211
Descripción de los programas ejemplo	213
El programa supervisor de colas	213
El programa servidor de colas	213
El programa cliente	213
Conclusión	213
Bibliografía	214
Software necesario	214
Construcción de núcleos OOP para Clipper	215
(Antonio J. Rojo López)	
Teoría de la programación orientada al objeto	216
Diferentes estilos de programación	216
Diferentes formas de implementar el lenguaje	218
Diferentes estilos OOP	221

OOP en Clipper	222
Clases predefinidas	223
Funciones no documentadas para creación de clases	223
Conclusiones	227
Pequeño núcleo OOP	230
Características finales	237
El interior de Clipper	238
El objeto como array	239
El módulo Send	239
La tabla de símbolos	240
Estructuras de variables en Clipper	245
La pila de Clipper	247
Más funciones indocumentadas (módulo Send)	252
Creación de núcleos OOP	258
Sin redefinir el módulo Send	264
Compatibilidad con el depurador	274
Redefinición del módulo Send	277
Hasta aquí, se entiende por OOP	280
Los objetos como datos	280
Objetos permanentes (datos)	280
Objetos permanentes (métodos)	286
Programación orientada al objeto con Clipper 5	289
(Brian Loesgen)	
Introducción	290
Qué es un objeto	290
Qué es una Clase	290
Qué es un mensaje	291
Qué hace tan buena a la programación orientada al objeto	291

Variables	293
Polimorfismo	294
Objetos compuestos	294
Delegación	297
Herencia múltiple	298
WinGauge: Alternativas de diseño	298
Tipos de métodos	299
Destruyores	300
Métodos de acceso y asignación	302
Control en la asignación de variables	302
Triggers	303
Crear ventanas	304
Ampliación de las ventanas	307
Otras formas de ampliar Clipper	312
Variables de instancia calculadas	313
Técnicas Data-Driving con objetos	314
Referencias locales	315
El problema del rendimiento de la instanciación	317
Referencias estáticas	317
Estructura de aplicaciones: Locales	318

Estructura de aplicaciones: Instanciación en funciones	319
Utilización de nombres	320
Utilización de identificadores (ID)	320
Conclusión	322
Expectativas de desarrollo bajo los nuevos entornos operativos	323
(Fernando Iglesia)	
Nuestro reto como programadores de élite	324
Requerimientos de hardware	324
Requerimientos de software	326
Qué se le pide a un entorno de desarrollo	329
Recursos	330
Amigabilidad y potencia	330
Rapidez de desarrollo	331
Personas afines. Grupos de usuarios. Ventajas	331
Precios	332
Ayudas	333
ODBC	333
Herramientas de un sistema de desarrollo visual	333
Migración	344
Qué aplicaciones debo programar en Windows	344
Qué aplicaciones debo programar en DOS	345
Qué aplicaciones debo migrar hacia Windows	345
Qué aplicaciones debo reescribir en Windows	346
Conclusión	347

Las jerarquías de clases en Visual Objects	349
(Alfonso Fraguas Bravo)	
Introducción	350
Conceptos básicos	351
Qué es una clase	351
Clase abstracta	351
Clase base	352
Clase derivada	353
Clase hija	353
Clase padre	353
Clase hermana	354
Herencia	354
Herencia Simple	358
Herencia Múltiple	358
Posesión	359
Clases contenedoras	362
Clases contenidas	362
Clases en VO	362
Qué tiene VO de todo lo dicho	362
Definición de clase	363
Métodos	369
Construcción e inicialización de objetos	371
Self y Super	373
Acceso y asignación	375
Early binding y Late binding	378
Sobrecarga de variables	382
Métodos predefinidos	383
Sobrecarga de operadores	385
Mandando mensajes a arrays de objetos	387
El objeto del sistema	387
Los objetos como referencias	388
Los objetos como parámetros	388
El browse de clases	389
Jerarquías de VO	393
Application (Aplicación)	394

Control (Control)	396
Database/File (Base de datos, fichero)	397
Display (Mostrar)	397
Event (Evento)	399
Help (Ayuda)	400
Menu (Menú)	401
Printing and Reporting (Impresión e informes)	402
Terminal (Terminal)	402
Window (Ventana)	402
 InfoBoxOOP: Un ejemplo	 406
 Conclusión	 414
 Bibliografía	 416
 Anatomía y fisiología de un sistema de ventanas	 417
(Francisco Morero)	
Introducción	418
Planteamiento de objetivos	418
Un poco de historia	419
Elementos de un sistema de ventanas	422
El servidor y las ventanas	422
Descripción de una ventana	423
Ciclo de vida de una ventana	424
Algunas peculiaridades de las ventanas	426
Programación dirigida por eventos	427
Introducción	427
Del evento al mensaje	429
El servidor del sistema y la cola de mensajes	430
El flujo de los mensajes	431
Filtros para ventanas	433
Multitarea preentiva	434
Introducción	434

Funcionamiento de la multitarea preentiva	435
Procesos modales en un entorno de multitarea	437
Programando un sistema de ventanas	437
Ventajas de programar en un entorno de ventanas	437
Tamaño de las aplicaciones	439
Reflexión final	439
 Bibliografía	 440
 Comenzando con Visual Objects	 441
(Larry Eiss)	
Introducción	442
Un cambio en el paradigma del desarrollo de aplicaciones	443
Qué es lo nuevo	444
El paradigma actual	444
Las cosas cambiaron	445
El nuevo paradigma	445
Gestión de la complejidad	446
Concurrencia	447
La aproximación orientada al objeto	448
Impacto en el desarrollo	448
Bases de datos dispares	449
Desarrollo tradicional	450
Desarrollo futuro	450
Sobre el lenguaje	451
Complejidad	452
Revolución vs. evolución	452
Una herramienta del futuro	454
 Introducción a la teoría y conceptos de la orientación a objetos	 455
Términos	455
Abstracción	456
El cambio del paradigma	456
Clases de objetos	457
Definiendo la clase casa	457

Otra abstracción	458
Jerarquía	458
Herencia	458
Propiedades	460
Estado	460
Comportamiento	461
Instanciación	462
Polimorfismo	463
Encapsulación	464
Modularidad	464
CA-Visual Objects	465
Puesta en marcha de CA-Visual Objects	469
Desarrollo basado en repositorio	470
Browsers	472
Browser de aplicación (Application Browser)	472
Module Browser	474
Entity Browser	475
Class Browser	477
Clases GUI	478
Clases DBF	478
Clases SQL	479
Clases Report	479
Clases System	480
Relación entre las clases DataWindow/DataServer	480
Editores Data Server	483
Editor de menús	493
CA-RET	494
Editor de iconos	496
Creación de un Framework estándar para la aplicación	496
Un viaje por el escritorio	501
Utilización del Module Browser	501
Utilización de la aplicación-Entity Browser	503
Acceso a los editores	503
Ver la aplicación con más detalle	504
Estructura de una aplicación MDI	505
El método App:Start()	506
La ventana Shell	507

La ventana shell vacía	512
La ventana shell estándar	514
Las ventanas hijas	517
Gestores de eventos y error por defecto	519
Construcción y ejecución del programa estándar	520
Ejecución del programa estándar	521
La ventana shell vacía	522
Apertura de ficheros de base de datos	526
Cambio entre las vistas Browse y Formulario	527
La ventana Standard Shell	530
Apertura de múltiples ventanas	531
Arrastrar y soltar	534
Utilización de los servidores de datos	535
Importación de la librería OE Data Server	536
Un rápido tour por el servidor de datos de clientes	537
Cargar el servidor de datos de Customer	537
El group box de índices (Indexes)	540
El group box de campos (Fields)	541
Creación del servidor de datos Orders	545
Abrir el editor DBServer	545
Importar el fichero de base de datos	548
Importar los ficheros índice	549
Compartir especificaciones de campos	551
Personalización de las propiedades de los campos	553
El editor FieldSpec	555
El código fuente	560
Construcción de la librería OE Data Servers	561
Añadir la librería al path de búsqueda de Order Entry	562
Creación de una ventana de datos	564
Puesta en marcha del Window Editor	565
Propiedades de la ventana	566
Utilización de Auto Layout	568
Revisión detallada de la ventana de datos principal	571
Revisión detallada de la subventana de datos	573
Continuar	577
El código fuente	577
Modificación del menú	578
Puesta en marcha del editor de menús	578
Añadir el mandato de menú Pedidos de Clientes	579

Añadir los métodos de ordenación	584
Modificación del menú	585
Añadir mandatos	586
Ejecución de la aplicación Order Entry	595
Generar un fichero ejecutable	595
Ejecutar la aplicación	596
Ver las nuevas características	597
Curso de programación orientada al objeto	603
(Antonio Quirós/Alfonso Fraguas)	
Introducción	604
Por qué OOP	604
Por qué el rechazo a la OOP	605
Algunas definiciones	606
Jerarquías de objetos	608
Tipos de datos	610
Tipos de métodos	610
Características de la OOP	611
Ventajas y desventajas de la OOP	613
Clipper y la OOP	614
High Class, núcleo OOP para Clipper	616
Un ejemplo sencillo: la clase Ventana	617
Herencia de Window con redefinición de métodos	621
Llamadas a métodos de la clase padre	622

Herencia de Window sin redefinición de métodos	623
Un objeto es una estructura	625
Herencia de las clases estándar de Clipper	643
Herencia múltiple	643
Clases contenedoras y clases contenidas	644
Variables de la clase	649
Conclusión	651
Bibliografía	651
Apéndice. Club Español de Usuarios Clipper	653
(Pablo Suárez)	
Qué es el Club Español de Usuarios Clipper	654
Un poco de historia	654
Por qué un club	655
Entidad altruista sin ánimo de lucro. Ventajas e inconvenientes	655
Fines de CEUC	657
Quiénes forman el CEUC	661
Qué servicios proporciona el CEUC	663
Qué presente y futuro tiene el CEUC	665
Qué creemos haber conseguido ya	665