

```

1: *****
2: CLASS TYPE CONST List mX4 3.9.9.190 #locs=17995
3:
4:
5: *****
6: //sign1:max: APSN21: 19.03.2015 13:20:31
7: //#sign:Administrator: PC08: 14/03/2015 08:01:49 PM
8: //file:TypeConstClasslistSortmX4Reporfinal.txt
9: //tech:perf: threads: 5 192.168.1.40 05:02:49 PM 3.9.9.190
10: *****
11:
12:
13: 0 ***** File 1: F:\EKON15\maxbox3\mX39maxbox3\source\IFSI_gsDButils.pas
14: 1 (InheritsFrom), Classname)
15: 2 ('TQuery'), 'TQueryUniDirectionalGS') do
16: 2 ***** File 2: F:\maxbox3\mX3\maxbox3\source\IFSI_gsUtils.pas
17: 3 (InheritsFrom), Classname)
18: 3 ***** File 3: F:\maxbox3\mX397\maxbox3\source\IFSI_IdFTP.pas
19: 4 (InheritsFrom), Classname)
20: 5 ('TIdTCPClient'), 'TIdFTP') do
21: 6 ('TPersistent'), 'TIdFtpProxySettings') do
22: 7 ('TOBJECT'), 'EIdFTPFileAlreadyExists');
23: 7 ***** File 4: F:\maxbox3\mX397\maxbox3\source\IFSI_IdGlobalProtocols.pas
24: 8 (InheritsFrom), Classname)
25: 9 ('TInterfacedObject'), 'TIdInterfacedObject') do
26: 10 ('TOBJECT'), 'TIdMimeTable') do
27: 11 ('TOBJECT'), 'EIdFailedToRetrieveTimeZoneInfo');
28: 12 ('TOBJECT'), 'EIdExtensionAlreadyExists');
29: 12 ***** File 5: F:\maxbox3\mX397\maxbox3\source\IFSI_IdHTTP.pas
30: 13 (InheritsFrom), Classname)
31: 14 ('EIdProtocolReplyError'), 'EIdHTTPProtocolException') do
32: 15 ('TIdCustomHTTP'), 'TIdHTTP') do begin
33: 16 ('TIdTCPClient'), 'TIdCustomHTTP') do begin
34: 17 ('TOBJECT'), 'TIdHTTPProtocol') do begin
35: 18 ('TIdRequestHeaderInfo'), 'TIdHTTPRequest') do begin
36: 19 ('TIdResponseHeaderInfo'), 'TIdHTTPResponse') do begin
37: 20 ('TOBJECT'), 'TIdCustomHTTP');
38: 21 ('TOBJECT'), 'EIdUnknownProtocol');
39: 21 ***** File 6: F:\maxbox3\mX\maxbox3\source\IFSI_IdTCPClient.pas
40: 22 (InheritsFrom), Classname)
41: 23 ('TIdTCPConnection'), 'TIdTCPClient') do
42: 23 ***** File 7: F:\maxbox3\mX\maxbox3\source\IFSI_IdURI.pas
43: 24 (InheritsFrom), Classname)
44: 25 ('TOBJECT'), 'TIdURI') do begin
45: 26 ('TOBJECT'), 'EIdURIException');
46: 26 ***** File 8: F:\maxbox3\mX\maxbox3\source\IFSI_SysUtils_max.pas
47: 27 ('TOBJECT'), 'TTextStream') do begin
48: 28 ('TOBJECT'), 'TStopwatch') do begin
49: 29 (InheritsFrom), Classname)
50: 30 ('EOSError'), 'EWin32Error') do
51: 31 ('Exception'), 'EOSError') do begin
52: 32 ('EExternal'), 'EStackOverflow') do
53: 33 ('Exception'), 'EExternal') do begin
54: 34 ('Exception'), 'EInOutError') do begin
55: 35 ('Exception'), 'EHeapException') do begin
56: 36 ('Exception'), 'EInvalidPassword') do begin
57: 37 ('TOBJECT'), 'Exception') do begin
58: 38 ('Exception'), 'EInvalidOperation') do begin
59: 39 ('TOBJECT'), 'TLanguages') do begin
60: 40 ('TPersistent'), 'TGraphic');
61: 41 ('Class of TForm'), 'TFormClass'); //3.8
62: 42 ('TOBJECT'), 'TClass');
63: 43 ('TOBJECT'), 'EAbort');
64: 44 ('TOBJECT'), 'EOOutOfMemory');
65: 45 ('TOBJECT'), 'EExternalException');
66: 46 ('TOBJECT'), 'EIntError');
67: 47 ('TOBJECT'), 'EDivByZero');
68: 48 ('TOBJECT'), 'ERangeError');
69: 49 ('TOBJECT'), 'EIntOverflow');
70: 50 ('EExternal'), 'EMathError');
71: 51 ('TOBJECT'), 'EInvalidOp');
72: 52 ('EMathError'), 'EZeroDivide');
73: 53 ('TOBJECT'), 'EOverflow');
74: 54 ('TOBJECT'), 'EUnderflow');
75: 55 ('TOBJECT'), 'EInvalidPointer');
76: 56 ('TOBJECT'), 'EInvalidCast');
77: 57 ('TOBJECT'), 'EConvertError');
78: 58 ('TOBJECT'), 'EAccessViolation');
79: 59 ('TOBJECT'), 'EPrivilege');
80: 60 ('TOBJECT'), 'EControlC');
81: 61 ('TOBJECT'), 'EVariantError');
82: 62 ('TOBJECT'), 'EPropReadOnly');
83: 63 ('TOBJECT'), 'EPropWriteOnly');
84: 64 ('TOBJECT'), 'EAssertionFailed');
85: 65 ('TOBJECT'), 'EIntfCastError');
86: 66 ('TOBJECT'), 'EInvalidContainer');
87: 67 ('TOBJECT'), 'EInvalidInsert');
88: 68 ('TOBJECT'), 'EPackageError');
89: 69 ('TOBJECT'), 'ESafecallException');

```

```

90: 69 ***** File 9: F:\maxbox3\mX397\maxbox3\source\IFSI_WinForm1puzzle.pas
91: 70 ('TForm'), 'TwinFormp') do begin
92: 71 ('TLabel'), 'TMyLabel') do begin
93: 72 ('TForm'), 'Tdebugoutput') do begin
94: 73 (InheritsFrom), Classname)
95: 73 ***** File 10: F:\maxbox3\mX397\maxbox3\source\upSC_classes.pas
96: 74 ('TComponent'), 'TBasicAction') do begin
97: 75 ('TOBJECT'), 'TTextReader') do begin
98: 76 ('TComponent'), 'TDataModule') do begin
99: 77 ('TObject'), 'Tfiler') do begin
100: 78 ('TPersistent'), 'TStrings') do begin
101: 79 ('TInterfacedObject'), 'TStreamAdapter') do begin
102: 80 ('TStream'), 'TStringStream') do begin
103: 81 ('TOBJECT'), 'TThread') do begin
104: 82 ('TOBJECT'), 'TThreadList') do begin
105: 83 ('TStrings'), 'TStringList') do begin
106: 84 ('TObject'), 'TBits') do begin
107: 85 ('TOBJECT'), 'TStream') do begin
108: 86 ('TSTREAM'), 'THandleStream') do begin
109: 87 ('TCUSTOMMEMORYSTREAM'), 'TMemoryStream') do begin
110: 88 ('THandleStream'), 'TFileStream') do begin
111: 89 ('TSTREAM'), 'TCustomMemoryStream') do begin
112: 90 ('TCUSTOMMEMORYSTREAM'), 'TResourceStream') do begin
113: 91 ('TOBJECT'), 'TParser') do begin
114: 92 ('TCOLLECTION') = nil then cl.AddClassN(cl.
115: 93 ('TPERSISTENT'), 'TCollectionItem') do
116: 94 ('TPERSISTENT'), 'TCollection');
117: 95 ('TCOLLECTION'), 'TOwnedCollection') do
118: 96 ('TOBJECT'), 'TStringList');
119: 97 ('TOBJECT'), 'TStrings');
120: 98 ('TOBJECT'), 'Tfiler');
121: 99 ('TOBJECT'), 'TReader');
122: 100 ('TOBJECT'), 'TWriter');
123: 101 ('TOBJECT'), 'TStream');
124: 102 ('TOBJECT'), 'TList');
125: 103 ('TOBJECT'), 'EFCREATEERROR');
126: 104 ('TOBJECT'), 'EFOpenError');
127: 105 ('TOBJECT'), 'EfilerError');
128: 106 ('TOBJECT'), 'EReadError');
129: 107 ('TOBJECT'), 'EWriteError');
130: 108 ('TOBJECT'), 'EClassNotFound');
131: 109 ('TOBJECT'), 'EMethodNotFound');
132: 110 ('TOBJECT'), 'EInvalidImage');
133: 111 ('TOBJECT'), 'EResNotFound');
134: 112 ('TOBJECT'), 'EListError');
135: 113 ('TOBJECT'), 'EBitsError');
136: 114 ('TOBJECT'), 'EStringListError');
137: 115 ('TOBJECT'), 'EComponentError');
138: 116 ('TOBJECT'), 'EParserError');
139: 117 ('TOBJECT'), 'EOutOfResources');
140: 118 ('TOBJECT'), 'EInvalidOperation');
141: 118 ***** File 11: F:\maxbox3\mX397\maxbox3\source\upSC_controls.pas
142: 119 ('TDragImageList'), 'TImageList') do begin
143: 120 ('TCustomImageList'), 'TDragImageList') do begin
144: 121 ('TComponent'), 'TCONTROL') do begin
145: 122 ('TControl'), 'TWINCONTROL') do begin
146: 123 ('TControl'), 'TGRAPHICCONTROL');
147: 124 ('TWinControl'), 'TCUSTOMCONTROL');
148: 125 ('TObject'), 'TDragObject') do
149: 125 ***** File 12: F:\maxbox3\mX397\maxbox3\source\upSC_DB.pas
150: 126 (InheritsFrom), Classname)
151: 127 ('TComponent'), 'TDataSource') do begin
152: 128 ('TComponent'), 'TCustomConnection') do begin
153: 129 ('EDatabaseError'), 'EUpdateError') do
154: 130 ('TOBJECT'), 'TFields');
155: 131 ('TOBJECT'), 'TField');
156: 132 ('TOBJECT'), 'TObjectField');
157: 133 ('TOBJECT'), 'TDataLink');
158: 134 ('TOBJECT'), 'TDataSource');
159: 135 ('TOBJECT'), 'TDataSet');
160: 136 ('TOBJECT'), 'TFieldDefs');
161: 137 ('TOBJECT'), 'TIndexDefs');
162: 138 ('TOBJECT'), 'EDatabaseError');
163: 139 ('TOBJECT'), 'TParams');
164: 139 ***** File 13: F:\maxbox3\mX397\maxbox3\source\upSC_extctrls.pas
165: 140 ('TGRAPHICCONTROL'), 'TSHAPE') do begin
166: 141 ('TGRAPHICCONTROL'), 'TIMAGE') do
167: 142 ('TGRAPHICCONTROL'), 'TPAINTBOX') do
168: 143 ('TGRAPHICCONTROL'), 'TBEVEL') do
169: 144 ('TComponent'), 'TTIMER') do
170: 145 ('TCUSTOMCONTROL'), 'TCUSTOMPANEL');
171: 146 ('TCUSTOMPANEL'), 'TPANEL') do
172: 147 ('TCUSTOMCONTROL'), 'TPAGE') do
173: 148 ('TCUSTOMCONTROL'), 'TNOTEBOOK') do
174: 149 ('TCUSTOMCONTROL'), 'THEADER') do
175: 150 ('TCustomGroupBox'), 'TCustomRadioGroup') do begin
176: 151 ('TCUSTOMGROUPBOX'), 'TCUSTOMRADIOGROUP') do begin
177: 152 ('TCUSTOMRADIOGROUP'), 'TRADIOGROUP') do begin
178: 152 ***** File 14: F:\maxbox3\mX397\maxbox3\source\upSC_forms.pas

```

```

179: 153 ('TObject','TMonitor') do begin
180: 154 ('TPERSISTENT','TCONTROLSCROLLBAR') do begin
181: 155 ('TWINCONTROL','TSCROLLINGWINCONTROL') do begin
182: 156 ('TSCROLLINGWINCONTROL','TSCROLLBOX') do begin
183: 157 ('TSCROLLINGWINCONTROL','TFORM') do begin
184: 158 ('TObject','Exception') do
185: 159 ('TComponent','TAPPLICATION') do begin
186: 160 ('TObject','Exception');
187: 161 ('TMenu','TMainMenu');
188: 162 ('TScrollingWinControl','TCustomForm');
189: 163 ('TObject','Exception') do
190: 164 ('TComponent','TScreen') do begin
191: 165 ('TCustomStatusBar','TStatusBar') do
192: 166 ('TWinControl','TCustomStatusBar') do begin
193: 167 ('TCollection','TStatusPanels') do begin
194: 168 ('TCollectionItem','TStatusPanel') do begin
195: 169 ('TGraphic','TICON');
196: 170 ('TObject','TStatusBar');
197: 170 ***** File 15: F:\maxbox3\mX397\maxbox3\source\upSC_forms_.pas
198: 171 ('TPERSISTENT','TCONTROLSCROLLBAR') do
199: 172 ('TWINCONTROL','TSCROLLINGWINCONTROL') do
200: 173 ('TSCROLLINGWINCONTROL','TSCROLLBOX') do
201: 174 ('TSCROLLINGWINCONTROL','TFORM') do
202: 175 ('TComponent','TSCREEN') do begin
203: 176 ('TComponent','TAPPLICATION') do
204: 176 ***** File 16: F:\maxbox3\mX397\maxbox3\source\upSC_graphics.pas
205: 177 ('TPERSISTENT','TGRAPHICSOBJECT') do begin
206: 178 ('TPersistent','TGraphic') do begin
207: 179 ('TGraphic','TMetafile') do begin
208: 180 ('TGraphic','TIcon') do begin
209: 181 ('TGraphicsObject','TFONT') do begin
210: 182 ('TPersistent','TCANVAS') do begin
211: 183 ('TGRAPHICSOBJECT','TPEN') do begin
212: 184 ('TGRAPHICSOBJECT','TBRUSH') do begin
213: 185 ('TObject','EInvalidGraphic');
214: 186 ('TObject','EInvalidGraphicOperation');}
215: 187 ('TObject','EInvalidGraphic');
216: 188 ('TObject','EInvalidGraphicOperation');
217: 189 ('TObject','TGraphic');
218: 190 ('TObject','TBitmap');
219: 191 ('TObject','TIcon');
220: 192 ('TObject','TMetafile');
221: 193 ('TGraphic','TBitmap'); //for alloc pattern bitmap
222: 194 ('TGraphic','TBitmap') do begin
223: 195 ('TPersistent','TPicture') do begin
224: 195 ***** File 17: F:\maxbox3\mX397\maxbox3\source\upSC_std.pas
225: 196 ('TObject','TPersistent') do begin
226: 197 ('TPersistent','TComponent') do begin
227: 197 ***** File 18: F:\maxbox3\mX397\maxbox3\source\upSC_stdctrls.pas
228: 198 ('TCUSTOMCONTROL','TCUSTOMGROUPBOX');
229: 199 ('TCUSTOMGROUPBOX','TGROUPOBOX') do
230: 200 ('TGRAPHICCONTROL','TCUSTOMLABEL') do
231: 201 ('TCUSTOMLABEL','TLABEL') do
232: 202 ('TWINCONTROL','TCUSTOMEDIT') do
233: 203 ('TCUSTOMEDIT','TEDIT') do
234: 204 ('TCUSTOMEDIT','TCUSTOMMEMO') do
235: 205 ('TCUSTOMMEMO','TMEMO') do begin
236: 206 ('TWINCONTROL','TCUSTOMCOMBOBOX') do begin
237: 207 ('TCUSTOMCOMBOBOX','TCOMBOBOX') do begin
238: 208 ('TWINCONTROL','TBUTTONCONTROL') do begin
239: 209 ('TBUTTONCONTROL','TBUTTON') do begin
240: 210 ('TBUTTONCONTROL','TCUSTOMCHECKBOX') do
241: 211 ('TCUSTOMCHECKBOX','TCHECKBOX') do
242: 212 ('TBUTTONCONTROL','TRADIOBUTTON') do
243: 213 ('TWINCONTROL','TCUSTOMLISTBOX') do begin
244: 214 ('TCUSTOMLISTBOX','TLISTBOX') do begin
245: 215 ('TWINCONTROL','TSCROLLBAR') do
246: 216 ('TObject','TPopUpMenu');
247: 216 ***** File 19: F:\maxbox3\mX397\maxbox3\source\upSI_ActnList.pas
248: 217 ('TBasicActionLink','TActionLink') do
249: 218 ('TCustomAction','TAction') do
250: 219 ('TContainedAction','TCustomAction') do begin
251: 220 ('TStringList','TShortCutList') do begin
252: 221 ('TCustomActionList','TActionList') do
253: 222 ('TComponent','TCustomActionList') do begin
254: 223 ('TObject','TActionListEnumerator') do begin
255: 224 ('TBasicAction','TContainedAction') do
256: 225 ('TObject','TCustomActionList');
257: 225 ***** File 20: F:\maxbox3\mX397\maxbox3\source\upSI_ADODB.pas
258: 226 ('TMemoryStream','TADOBlobStream') do begin
259: 227 ('TCustomADODataSet','TADOStoredProc') do begin
260: 228 ('TCustomADODataSet','TADOQuery') do begin
261: 229 ('TCustomADODataSet','TADOTable') do begin
262: 230 ('TCustomADODataSet','TADODataSet') do begin
263: 231 ('TWideDataSet','TCustomADODataSet') do begin
264: 232 ('TComponent','TADOCommand') do begin
265: 233 ('TOwnedCollection','TParameters') do begin
266: 234 ('TCollectionItem','TParameter') do begin
267: 235 ('TCustomConnection','TRDSConnection') do begin

```

```
268: 236 ('TCustomConnection','TADOConnection') do begin
269: 237 ('TOBJECT','TWideStrings');
270: 238 ('TOBJECT','TADOCommand');
271: 239 ('TOBJECT','TCustomADODataSet');
272: 240 ('TOBJECT','TADODataSet');
273: 241 ('TOBJECT','TParameters');
274: 242 ('TOBJECT','TADOConnection');
275: 243 ('TOBJECT','EADOError');
276: 243 ***** File 21: F:\maxbox3\mX397\maxbox3\source\upsi_AESPassWordDlg.pas
277: 244 ('TForm','TPasswordDlg') do begin
278: 244 ***** File 22: F:\maxbox3\mX397\maxbox3\source\upsi_AsyncCalls.pas
279: 245 ('TInterfacedObject','TSyncCall') do
280: 246 ('TInterfacedObject','TAsyncCall') do
281: 247 ('TOBJECT','TInternalAsyncCall') do
282: 248 ('TOBJECT','EAsyncCallError');
283: 249 ('TOBJECT','TAsyncCall');
284: 250 ('IUnknown','TAsyncCall');
285: 251 ('TOBJECT','TAsyncCall');
286: 251 ***** File 23: F:\maxbox3\mX397\maxbox3\source\upsi_BarCodeScanner.pas
287: 252 ('TCustomComPort','TBarCodeScanner') do begin
288: 252 ***** File 24: F:\maxbox3\mX397\maxbox3\source\upsi_BoldUtils.pas
289: 253 ('TComponent','TBoldPassthroughNotifier') do
290: 253 ***** File 25: F:\maxbox3\mX397\maxbox3\source\upsi_Calendar.pas
291: 254 ('TCustomGrid','TCalendar') do begin
292: 254 ***** File 26: F:\maxbox3\mX397\maxbox3\source\upsi_CheckList.pas
293: 255 ('TCustomListBox','TCheckListBox') do
294: 255 ***** File 27: F:\maxbox3\mX397\maxbox3\source\upsi_Classes_draft.pas
295: 256 ('TComponent','TDataModule') do
296: 257 ('TComponent','TBasicAction') do
297: 258 ('TOBJECT','TBasicActionLink') do
298: 259 ('TPersistent','TComponent') do
299: 260 ('TOBJECT','TComponentEnumerator') do
300: 261 ('TOBJECT','TThread') do
301: 262 ('TOBJECT','TParser') do
302: 263 ('TFile','TWriter') do
303: 264 ('TFile','TReader') do
304: 265 ('TOBJECT','TFile') do
305: 266 ('TOBJECT','TClassFinder') do
306: 267 ('TInterfacedObject','TStreamAdapter') do
307: 268 ('TCustomMemoryStream','TResourceStream') do
308: 269 ('TStream','TStringStream') do
309: 270 ('TCustomMemoryStream','TMemoryStream') do
310: 271 ('TStream','TCustomMemoryStream') do
311: 272 ('THandleStream','TFileStream') do
312: 273 ('TStream','THandleStream') do
313: 274 ('TOBJECT','TStream') do
314: 275 ('TStrings','TStringList') do
315: 276 ('TPersistent','TStrings') do
316: 277 ('TOBJECT','TStringsEnumerator') do
317: 278 ('TCollection','TOwnedCollection') do
318: 279 ('TPersistent','TCollection') do
319: 280 ('TOBJECT','TCollectionEnumerator') do
320: 281 ('TPersistent','TCollectionItem') do
321: 282 ('TOBJECT','TRecall') do
322: 283 ('TPersistent','TInterfacedPersistent') do
323: 284 ('TOBJECT','TPersistent') do
324: 285 ('TOBJECT','TBits') do
325: 286 ('TInterfacedObject','TInterfaceList') do
326: 287 ('TOBJECT','TInterfaceListEnumerator') do
327: 288 ('TOBJECT','TThreadList') do
328: 289 ('TOBJECT','TList') do
329: 290 ('TOBJECT','TListEnumerator') do
330: 291 ('EStreamError','EFileStreamError') do
331: 292 ('TOBJECT','EStreamError');
332: 293 ('TOBJECT','EFCREATEError');
333: 294 ('TOBJECT','EFOpenError');
334: 295 ('TOBJECT','EFileError');
335: 296 ('TOBJECT','EReadError');
336: 297 ('TOBJECT','EWriteError');
337: 298 ('TOBJECT','EClassNotFound');
338: 299 ('TOBJECT','EMethodNotFound');
339: 300 ('TOBJECT','EInvalidImage');
340: 301 ('TOBJECT','EResNotFound');
341: 302 ('TOBJECT','EListError');
342: 303 ('TOBJECT','EBitsError');
343: 304 ('TOBJECT','EStringListError');
344: 305 ('TOBJECT','EComponentError');
345: 306 ('TOBJECT','EParserError');
346: 307 ('TOBJECT','EOutOfResources');
347: 308 ('TOBJECT','EInvalidOperation');
348: 309 ('TOBJECT','TStream');
349: 310 ('TOBJECT','TFile');
350: 311 ('TOBJECT','TReader');
351: 312 ('TOBJECT','TWriter');
352: 313 ('TOBJECT','TComponent');
353: 314 ('TOBJECT','TList');
354: 315 ('TOBJECT','TInterfaceList');
355: 316 ('TOBJECT','TCollection');
356: 317 ('TOBJECT','TStrings');
```

```

357: 318 ('TOBJECT'), 'TStringList');
358: 319 ('TOBJECT'), 'EThread');
359: 320 ('TOBJECT'), 'TBasicAction');
360: 320 ***** File 28: F:\maxbox3\mX397\maxbox3\source\upsi_clipbrd.pas
361: 321 ('TPersistent'), 'TClipboard') do begin
362: 321 ***** File 29: F:\maxbox3\mX397\maxbox3\source\upsi_comctrls.pas
363: 322 ('TListActionLink'), 'TComboBoxExActionLink') do
364: 323 ('TCustomComboBoxEx'), 'TComboBoxEx') do
365: 324 ('TCustomCombo'), 'TCustomComboBoxEx') do begin
366: 325 ('TCustomComboBoxStrings'), 'TComboBoxExStrings') do begin
367: 326 ('TListControlItem'), 'TComboBoxItems') do begin
368: 327 ('TListControlItem'), 'TComboBoxItem') do begin
369: 328 ('TWinControl'), 'TPageScroller') do begin
370: 329 ('TCommonCalendar'), 'TDateTimePicker') do begin
371: 330 ('TCommonCalendar'), 'TMonthCalendar') do begin
372: 331 ('TWinControl'), 'TCommonCalendar') do begin
373: 332 ('TPersistent'), 'TMonthCalColors') do begin
374: 333 ('TToolWindow'), 'TCoolBar') do begin
375: 334 ('TCollection'), 'TCoolBands') do begin
376: 335 ('TCollectionItem'), 'TCoolBand') do begin
377: 336 ('TDragDockObject'), 'TToolBarDockObject') do
378: 337 ('TToolWindow'), 'TToolBar') do begin
379: 338 ('TOBJECT'), 'TToolBarEnumerator') do begin
380: 339 ('TGraphicControl'), 'TToolButton') do begin
381: 340 ('TControlActionLink'), 'TToolButtonActionLink') do
382: 341 ('TWinControl'), 'TAnimate') do begin
383: 342 ('TListActionLink'), 'TListViewActionLink') do
384: 343 ('TCustomListView'), 'TListView') do begin
385: 344 ('TCustomMultiSelectListControl'), 'TCustomListView') do begin
386: 345 ('TPersistent'), 'TIconOptions') do begin
387: 346 ('TOwnedCollection'), 'TWorkAreas') do begin
388: 347 ('TCollectionItem'), 'TWorkArea') do begin
389: 348 ('TPersistent'), 'TListItems') do begin
390: 349 ('TOBJECT'), 'TListItemsEnumerator') do begin
391: 350 ('TPersistent'), 'TListItem') do begin
392: 351 ('TCollection'), 'TListColumns') do begin
393: 352 ('TCollectionItem'), 'TListColumn') do begin
394: 353 ('TCustomHotKey'), 'THotKey') do
395: 354 ('TWinControl'), 'TCustomHotKey') do
396: 355 ('TCustomUpDown'), 'TUpDown') do
397: 356 ('TWinControl'), 'TCustomUpDown') do
398: 357 ('TCustomRichEdit'), 'TRichEdit') do begin
399: 358 ('TCustomMemo'), 'TCustomRichEdit') do begin
400: 359 ('TOBJECT'), 'TConversion') do begin
401: 360 ('TPersistent'), 'TParaAttributes') do begin
402: 361 ('TPersistent'), 'TTextAttributes') do begin
403: 362 ('TWinControl'), 'TProgressBar') do begin
404: 363 ('TWinControl'), 'TTrackBar') do begin
405: 364 ('TCustomTreeView'), 'TTreeView') do begin
406: 365 ('TWinControl'), 'TCustomTreeView') do begin
407: 366 ('TPersistent'), 'TTreeNode') do begin
408: 367 ('TOBJECT'), 'TTreeNodeEnumerator') do begin
409: 368 ('TPersistent'), 'TTreeNode') do begin
410: 369 ('TCustomHeaderControl'), 'THeaderControl') do begin
411: 370 ('TWinControl'), 'TCustomHeaderControl') do begin
412: 371 ('TCollection'), 'THeaderSections') do begin
413: 372 ('TCollectionItem'), 'THeaderSection') do begin
414: 373 ('TCustomStatusBar'), 'TStatusBar') do begin
415: 374 ('TWinControl'), 'TCustomStatusBar') do begin
416: 375 ('TCollection'), 'TStatusPanels') do begin
417: 376 ('TCollectionItem'), 'TStatusPanel') do begin
418: 377 ('TCustomTabControl'), 'TPageControl') do begin
419: 378 ('TWinControl'), 'TTabSheet') do begin
420: 379 ('TCustomTabControl'), 'TTabControl') do
421: 380 ('TWinControl'), 'TCustomTabControl') do begin
422: 381 ('TOBJECT'), 'TCustomTabControl');
423: 382 ('TOBJECT'), 'TPageControl');
424: 383 ('TOBJECT'), 'TCustomStatusBar');
425: 384 ('TOBJECT'), 'TStatusPanel');
426: 385 ('TOBJECT'), 'TStatusPanels');
427: 386 ('TOBJECT'), 'TStatusBar');
428: 387 ('TOBJECT'), 'TCustomHeaderControl');
429: 388 ('TOBJECT'), 'THeaderControl');
430: 389 ('TOBJECT'), 'THeaderSection');
431: 390 ('TOBJECT'), 'TCustomTreeView');
432: 391 ('TOBJECT'), 'TTreeNode');
433: 392 ('TOBJECT'), 'TTreeNode');
434: 393 ('TOBJECT'), 'ETreeViewError');
435: 394 ('TOBJECT'), 'TCustomRichEdit');
436: 395 ('TOBJECT'), 'TListColumns');
437: 396 ('TOBJECT'), 'TListItem');
438: 397 ('TOBJECT'), 'TListItems');
439: 398 ('TOBJECT'), 'TCustomListView');
440: 399 ('TOBJECT'), 'TToolBar');
441: 400 ('TOBJECT'), 'TToolButton');
442: 401 ('TOBJECT'), 'TCoolBar');
443: 402 ('TOBJECT'), 'TCommonCalendar');
444: 403 ('TOBJECT'), 'ECommonCalendarError');
445: 404 ('TOBJECT'), 'EMonthCalError');

```



```

446: 405 ('TOBJECT'), 'EDatetimeError');
447: 406 ('TOBJECT'), 'TCustomComboBoxEx');
448: 406 ***** File 30: F:\maxbox3\mX397\maxbox3\source\upsi_Contrns.pas
449: 407 ('TBucketList'), 'TIntegerBucketList') do begin
450: 408 ('TBucketList'), 'TObjectBucketList') do begin
451: 409 ('TCustomBucketList'), 'TBucketList') do begin
452: 410 ('TObject'), 'TCustomBucketList') do begin
453: 411 ('TQueue'), 'TObjectQueue') do begin
454: 412 ('TOrderedList'), 'TQueue') do
455: 413 ('TStack'), 'TObjectStack') do begin
456: 414 ('TOrderedList'), 'TStack') do begin
457: 415 ('TObject'), 'TOrderedList') do begin
458: 416 ('TList'), 'TClassList') do begin
459: 417 ('TObjectList'), 'TComponentList') do begin
460: 418 ('TList'), 'TObjectList') do begin
461: 419 ('TOBJECT'), 'TClass');
462: 420 ('TOBJECT'), 'TCustomBucketList');
463: 420 ***** File 31: F:\maxbox3\mX397\maxbox3\source\upsi_ConvUtils.pas
464: 421 ('TConvTypeInfo'), 'TConvTypeProcs') do
465: 422 ('TConvTypeInfo'), 'TConvTypeFactor') do
466: 423 ('TObject'), 'TConvTypeInfo') do
467: 424 ('TOBJECT'), 'EConversionError');
468: 424 ***** File 32: F:\maxbox3\mX397\maxbox3\source\upsi_CPort.pas
469: 425 ('Exception'), 'EComPort') do begin
470: 426 ('TStream'), 'TComStream') do begin
471: 427 ('TComponent'), 'TComDataPacket') do begin
472: 428 ('TCustomComPort'), 'TComPort') do begin
473: 429 ('TComponent'), 'TCustomComPort') do begin
474: 430 ('TPersistent'), 'TComBuffer') do begin
475: 431 ('TPersistent'), 'TComParity') do begin
476: 432 ('TPersistent'), 'TComFlowControl') do begin
477: 433 ('TPersistent'), 'TComTimeouts') do begin
478: 434 ('TThread'), 'TComThread') do begin
479: 435 ('TOBJECT'), 'TComLink') do begin
480: 436 ('TOBJECT'), 'TCustomComPort');
481: 436 ***** File 33: F:\maxbox3\mX397\maxbox3\source\upsi_CPortCtl.pas
482: 437 ('TCustomComTerminal'), 'TComTerminal') do begin
483: 438 ('TCustomControl'), 'TCustomComTerminal') do begin
484: 439 ('TOBJECT'), 'TComTermBuffer') do begin
485: 440 ('TGraphicControl'), 'TComLed') do begin
486: 441 ('TCustomRadioGroup'), 'TComRadioGroup') do begin
487: 442 ('TCustomComboBox'), 'TComComboBox') do begin
488: 443 ('TOBJECT'), 'TComSelect') do begin
489: 444 ('TOBJECT'), 'TCustomComTerminal');
490: 444 ***** File 34: F:\maxbox3\mX397\maxbox3\source\upsi_CPortEsc.pas
491: 445 ('TEscapeCodes'), 'TEscapeCodesVT100') do begin
492: 446 ('TEscapeCodes'), 'TEscapeCodesVT52') do begin
493: 447 ('TOBJECT'), 'TEscapeCodes') do begin
494: 447 ***** File 35: F:\maxbox3\mX397\maxbox3\source\upsi_CPortMonitor.pas
495: 448 ('TCustomMemo'), 'TCPortMonitor') do begin
496: 448 ***** File 36: F:\maxbox3\mX397\maxbox3\source\upsi_cutils.pas
497: 449 ('TPersistent'), 'TSynHighlighterAttributes');
498: 449 ***** File 37: F:\maxbox3\mX397\maxbox3\source\upsi_DBBdeWeb.pas
499: 450 ('TDSTableProducer'), 'TQueryTableProducer') do
500: 450 ***** File 38: F:\maxbox3\mX397\maxbox3\source\upsi_DBCGrids.pas
501: 451 ('TWinControl'), 'TDBCtrlGrid') do begin
502: 452 ('TWinControl'), 'TDBCtrlPanel') do
503: 453 ('TDataLink'), 'TDBCtrlGridLink') do
504: 454 ('TOBJECT'), 'TDBCtrlGrid');
505: 454 ***** File 39: F:\maxbox3\mX397\maxbox3\source\upsi_DBClient.pas
506: 455 ('TMemoryStream'), 'TClientBlobStream') do begin
507: 456 ('TCustomClientDataSet'), 'TClientDataSet') do begin
508: 457 ('TWideDataSet'), 'TCustomClientDataSet') do begin
509: 458 ('TCollection'), 'TAggregates') do begin
510: 459 ('TCollectionItem'), 'TAggregate') do begin
511: 460 ('TCustomRemoteServer'), 'TConnectionBroker') do
512: 461 ('TCustomConnection'), 'TCustomRemoteServer') do begin
513: 462 ('EDBClient'), 'EReconcileError') do begin
514: 463 ('EDatabaseError'), 'EDBClient') do begin
515: 464 ('TOBJECT'), 'TCustomClientDataSet');
516: 465 ('TOBJECT'), 'TClientDataSet');
517: 466 ('TOBJECT'), 'TAggregate');
518: 467 ('TOBJECT'), 'TAggregates');
519: 467 ***** File 40: F:\maxbox3\mX397\maxbox3\source\upsi_DBCommon.pas
520: 468 ('TOBJECT'), 'TExprParser') do begin
521: 469 ('TOBJECT'), 'TFilterExpr') do begin
522: 469 ***** File 41: F:\maxbox3\mX397\maxbox3\source\upsi_DBCtrls.pas
523: 470 ('TCustomRichEdit'), 'TDBRichEdit') do begin
524: 471 ('TDBLookupControl'), 'TDBLookupComboBox') do begin
525: 472 ('TDBLookupListBox'), 'TPopupDataList') do
526: 473 ('TDBLookupControl'), 'TDBLookupListBox') do begin
527: 474 ('TCustomControl'), 'TDBLookupControl') do begin
528: 475 ('TDataLink'), 'TListSourceLink') do begin
529: 476 ('TDataLink'), 'TDataSourceLink') do begin
530: 477 ('TDataLink'), 'TNavDataLink') do begin
531: 478 ('TSpeedButton'), 'TNavButton') do
532: 479 ('TCustomPanel'), 'TDBNavigator') do begin
533: 480 ('TCustomControl'), 'TDBImage') do begin
534: 481 ('TCustomMemo'), 'TDBMemo') do begin

```

```

535: 482 ('TCustomRadioGroup'), 'TDBRadioGroup') do begin
536: 483 ('TCustomListBox'), 'TDBListBox') do begin
537: 484 ('TCustomComboBox'), 'TDBComboBox') do begin
538: 485 ('TCustomCheckBox'), 'TDBCheckBox') do begin
539: 486 ('TCustomLabel'), 'TDBText') do begin
540: 487 ('TCustomMaskEdit'), 'TDBEdit') do begin
541: 488 ('TOBJECT'), 'TPaintControl') do begin
542: 489 ('TDataLink'), 'TFieldDataLink') do begin
543: 490 ('TOBJECT'), 'TNavButton');
544: 491 ('TOBJECT'), 'TNavDataLink');
545: 492 ('TOBJECT'), 'TDBLookupControl');
546: 492 ***** File 42: F:\maxbox3\mX397\maxbox3\source\upsi_DBGrids.pas
547: 493 ('TCustomDBGrid'), 'TDBGrid') do begin
548: 494 ('TCustomGrid'), 'TCustomDBGrid') do begin
549: 495 ('TOBJECT'), 'TBookmarkList') do begin
550: 496 ('TDataLink'), 'TGridDataLink') do begin
551: 497 ('TCollection'), 'TDBGridColumns') do begin
552: 498 ('TCollectionItem'), 'TColumn') do begin
553: 499 ('TPersistent'), 'TColumnName') do begin
554: 500 ('TOBJECT'), 'TColumn');
555: 501 ('TOBJECT'), 'TDBGridColumns');
556: 502 ('TOBJECT'), 'TCustomDBGrid');
557: 502 ***** File 43: F:\maxbox3\mX397\maxbox3\source\upsi_DBLogDlg.pas
558: 503 ('TForm'), 'TLoginDialog') do begin
559: 503 ***** File 44: F:\maxbox3\mX397\maxbox3\source\upsi_DBPlatform.pas
560: 504 ('TOBJECT'), 'TPlatformField') do
561: 505 ('TOBJECT'), 'TPlatformPSResult') do
562: 506 ('TOBJECT'), 'TPlatformValueBuffer') do begin
563: 507 ('TOBJECT'), 'TPlatformRecordBuffer') do begin
564: 508 ('TOBJECT'), 'TPlatformBytes') do begin
565: 508 ***** File 45: F:\maxbox3\mX397\maxbox3\source\upsi_DBTables.pas
566: 509 ('TStream'), 'TBlobStream') do begin
567: 510 ('TSQLUpdateObject'), 'TUpdateSQL') do begin
568: 511 ('TDBDataSet'), 'TQuery') do begin
569: 512 ('TDBDataSet'), 'TStoredProc') do begin
570: 513 ('TComponent'), 'TBatchMove') do begin
571: 514 ('TDBDataSet'), 'TTable') do begin
572: 515 ('TStringList'), 'TIndexFiles') do begin
573: 516 ('TBDEDataSet'), 'TDBDataSet') do begin
574: 517 ('TBDEDataSet'), 'TNestedTable') do
575: 518 ('TDataSet'), 'TBDEDataSet') do begin
576: 519 ('TDataSetUpdateObject'), 'TSQLUpdateObject') do
577: 520 ('TComponent'), 'TDataSetUpdateObject') do begin
578: 521 ('TCustomConnection'), 'TDatabase') do begin
579: 522 ('TObject'), 'TParamList') do begin
580: 523 ('TComponent'), 'TSession') do begin
581: 524 ('TObject'), 'TSessionList') do
582: 525 ('TOBJECT'), 'TBDECallback') do
583: 526 ('TOBJECT'), 'TDBError') do
584: 527 ('EDatabaseError'), 'EDBEngineError') do
585: 528 ('TOBJECT'), 'TDBError');
586: 529 ('TOBJECT'), 'TSession');
587: 530 ('TOBJECT'), 'TDatabase');
588: 531 ('TOBJECT'), 'TBDEDataSet');
589: 532 ('TOBJECT'), 'TDBDataSet');
590: 533 ('TOBJECT'), 'TTable');
591: 534 ('TOBJECT'), 'ENoResultSet');
592: 534 ***** File 46: F:\maxbox3\mX397\maxbox3\source\upsi_DBWeb.pas
593: 535 ('TDSTableProducer'), 'TDataSetTableProducer') do
594: 536 ('TCustomContentProducer'), 'TDSTableProducer') do begin
595: 537 ('TCollection'), 'THTMLTableColumns') do begin
596: 538 ('TCollectionItem'), 'THTMLTableColumn') do begin
597: 539 ('TDataLink'), 'THTTpDataLink') do begin
598: 540 ('TOBJECT'), 'TDSTableProducerEditor') do begin
599: 541 ('TOBJECT'), 'TDSTableProducer');
600: 541 ***** File 47: F:\maxbox3\mX397\maxbox3\source\upsi_DBXChannel.pas
601: 542 ('TOBJECT'), 'TDBXChannel') do begin
602: 542 ***** File 48: F:\maxbox3\mX397\maxbox3\source\upsi_DBXIndyChannel.pas
603: 543 ('TDBXChannel'), 'TDBXIndyTcpChannel') do begin
604: 543 ***** File 49: F:\maxbox3\mX397\maxbox3\source\upsi_DBXpressWeb.pas
605: 544 ('TDSTableProducer'), 'TSQLQueryTableProducer') do
606: 544 ***** File 50: F:\maxbox3\mX397\maxbox3\source\upsi_DBX_connect.pas
607: 545 ('TComponent'), 'TCustomConnection') do
608: 546 ('EDatabaseError'), 'EUpdateError') do
609: 547 ('TOBJECT'), 'TFields');
610: 548 ('TOBJECT'), 'TField');
611: 549 ('TOBJECT'), 'TObjectField');
612: 550 ('TOBJECT'), 'TDataLink');
613: 551 ('TOBJECT'), 'TDataSource');
614: 552 ('TOBJECT'), 'TDataSet');
615: 553 ('TOBJECT'), 'TFieldDefs');
616: 554 ('TOBJECT'), 'TIndexDefs');
617: 555 ('TOBJECT'), 'EDatabaseError');
618: 555 ***** File 51: F:\maxbox3\mX397\maxbox3\source\upsi_Dialogs.pas
619: 556 ('TCustomTaskDialog'), 'TTaskDialog') do
620: 557 ('TComponent'), 'TCustomTaskDialog') do begin
621: 558 ('TOwnedCollection'), 'TTaskDialogButtons') do begin
622: 559 ('TOBJECT'), 'TTaskDialogButtonsEnumerator') do begin
623: 560 ('TTaskDialogBaseButtonItem'), 'TTaskDialogRadioButtonItem') do

```

```

624: 561 ('TTaskDialogBaseButtonItem', 'TTaskDialogButtonItem') do
625: 562 ('TCollectionItem', 'TTaskDialogBaseButtonItem') do
626: 563 ('TPersistent', 'TTaskDialogProgressBar') do
627: 564 ('TCustomFileSaveDialog', 'TFileSaveDialog') do
628: 565 ('TCustomFileDialog', 'TCustomFileSaveDialog') do
629: 566 ('TCustomFileOpenDialog', 'TFileOpenDialog') do
630: 567 ('TCustomFileDialog', 'TCustomFileOpenDialog') do
631: 568 ('TComponent', 'TCustomFileDialog') do begin
632: 569 ('TCollection', 'TFavoriteLinkItems') do begin
633: 570 ('TObject', 'TFavoriteLinkItemsEnumerator') do begin
634: 571 ('TCollectionItem', 'TFavoriteLinkItem') do
635: 572 ('TCollection', 'TFileTypeItems') do begin
636: 573 ('TCollectionItem', 'TFileTypeItem') do begin
637: 574 ('TFindDialog', 'TReplaceDialog') do
638: 575 ('TCommonDialog', 'TFindDialog') do begin
639: 576 ('TCommonDialog', 'TPageSetupDialog') do begin
640: 577 ('TCommonDialog', 'TPrintDialog') do begin
641: 578 ('TCommonDialog', 'TPrinterSetupDialog') do
642: 579 ('TCommonDialog', 'TFontDialog') do begin
643: 580 ('TCommonDialog', 'TColorDialog') do begin
644: 581 ('TOpenDialog', 'TSaveDialog') do
645: 582 ('TCommonDialog', 'TOpenDialog') do begin
646: 583 ('TComponent', 'TCommonDialog') do begin
647: 584 ('TObject', 'EPlatformVersionException');
648: 585 ('TObject', 'TFavoriteLinkItems');
649: 586 ('TObject', 'TCustomTaskDialog');
650: 587 ('TObject', 'TTaskDialogButtons');
651: 587 ***** File 52: F:\maxbox3\mX397\maxbox3\source\upsi_DTDSchema.pas
652: 588 ('TDTDBaseTranslator', 'TDTDImportTranslator') do
653: 589 ('TXMLSchemaTranslator', 'TDTDBaseTranslator') do
654: 589 ***** File 53: F:\maxbox3\mX397\maxbox3\source\upsi_ExtDlgs.pas
655: 590 ('TOpenTextFileDialog', 'TSaveTextFileDialog') do begin
656: 591 ('TOpenDialog', 'TOpenTextFileDialog') do begin
657: 592 ('TOpenPictureDialog', 'TSavePictureDialog') do begin
658: 593 ('TOpenDialog', 'TOpenPictureDialog') do
659: 593 ***** File 54: F:\maxbox3\mX397\maxbox3\source\upsi_FileCtrl.pas
660: 594 ('TCustomComboBox', 'TFilterComboBox') do begin
661: 595 ('TCustomComboBox', 'TDriveComboBox') do begin
662: 596 ('TCustomListBox', 'TDirectoryListBox') do begin
663: 597 ('TCustomListBox', 'TFileListBox') do begin
664: 598 ('TObject', 'TDirectoryListBox');
665: 599 ('TObject', 'TFilterComboBox');
666: 600 ('TObject', 'TDriveComboBox');
667: 600 ***** File 55: F:\maxbox3\mX397\maxbox3\source\upsi_fMain.pas
668: 601 ('TGraphicControl', 'TToolButton') do begin
669: 602 ('TComponent', 'TMenuItem') do begin
670: 603 ('TMaxForm1', 'MaxForm1') do begin
671: 604 ('TSynEdit', 'TSynMemo');
672: 605 ('TForm', 'TMaxForm1') do begin
673: 605 ***** File 56: F:\maxbox3\mX397\maxbox3\source\upsi_fMain_back.pas
674: 606 ('TForm', 'TMaxForm1') do
675: 606 ***** File 57: F:\maxbox3\mX397\maxbox3\source\upsi_FMTBcd.pas
676: 607 ('TObject', 'EBcdException');
677: 608 ('TObject', 'EBcdOverflowException');
678: 608 ***** File 58: F:\maxbox3\mX397\maxbox3\source\upsi_GUITesting.pas
679: 609 ('TTestCase', 'TGUITestCase') do begin
680: 609 ***** File 59: F:\maxbox3\mX397\maxbox3\source\upsi_HexDump.pas
681: 610 ('TCustomControl', 'THexDump') do begin
682: 610 ***** File 60: F:\maxbox3\mX397\maxbox3\source\upsi_HTTPApp.pas
683: 611 ('Exception', 'EWebBrokerException') do
684: 612 ('TInterfacedObject', 'TDefaultWebAppServices') do
685: 613 ('TObject', 'TAbstractWebModuleList') do
686: 614 ('TObject', 'TAbstractWebPageInfo') do
687: 615 ('TAbstractWebModuleFactory', 'TAbstractWebPageModuleFactory') do
688: 616 ('TObject', 'TAbstractWebModuleFactory') do
689: 617 ('TCustomWebDispatcher', 'TWebModule') do
690: 618 ('TCustomWebDispatcher', 'TWebDispatcher') do
691: 619 ('TDataModule', 'TCustomWebDispatcher') do
692: 620 ('TCollection', 'TWebActionItems') do
693: 621 ('TCollectionItem', 'TWebActionItem') do
694: 622 ('TComponent', 'TCustomContentProducer') do
695: 623 ('TObject', 'TAbstractWebSession') do
696: 624 ('TObject', 'TWebResponse') do
697: 625 ('TCollection', 'TCookieCollection') do
698: 626 ('TCollectionItem', 'TCookie') do
699: 627 ('TObject', 'TAbstractWebRequestFile') do
700: 628 ('TObject', 'TAbstractWebRequestFiles') do
701: 629 ('TAbstractContentParser', 'TContentParser') do
702: 630 ('TObject', 'TAbstractContentParser') do
703: 631 ('TObject', 'TWebRequest') do
704: 632 ('TObject', 'TWebResponse');
705: 633 ('TObject', 'TAbstractContentParser');
706: 634 ('TObject', 'TAbstractWebRequestFiles');
707: 635 ('TObject', 'TAbstractWebRequestFile');
708: 636 ('TObject', 'TCustomWebDispatcher');
709: 637 ('TObject', 'TCustomContentProducer');
710: 638 ('TObject', 'TAbstractWebModuleList');
711: 639 ('TObject', 'TAbstractWebPageInfo');
712: 639 ***** File 61: F:\maxbox3\mX397\maxbox3\source\upsi_HTTPParse.pas

```



```
713: 640 ('TObject'), 'THTTTParser') do
714: 640 ***** File 62: F:\maxbox3\mX397\maxbox3\source\upsi_IBUtils.pas
715: 641 ('TComponent'), 'TIBTimer') do begin
716: 642 ('TObject'), 'TIBTimer');
717: 642 ***** File 63: F:\maxbox3\mX397\maxbox3\source\upsi_IdAuthentication.pas
718: 643 ('TIdAuthentication'), 'TIdBasicAuthentication') do begin
719: 644 ('TPersistent'), 'TIdAuthentication') do begin
720: 645 ('Class of TIdAuthentication'), 'TIdAuthenticationClass'); //3.8
721: 646 ('TObject'), 'EIdAlreadyRegisteredAuthenticationMethod');
722: 646 ***** File 64: F:\maxbox3\mX397\maxbox3\source\upsi_IdCustomHTTPServer.pas
723: 647 ('TIdCustomHTTPServer'), 'TIdHTTPServer') do begin
724: 648 ('TIdHTTPCustomSessionList'), 'TIdHTTPDefaultSessionList') do begin
725: 649 ('TIdTCPClient'), 'TIdCustomHTTPServer') do begin
726: 650 ('TComponent'), 'TIdHTTPCustomSessionList') do begin
727: 651 ('TObject'), 'TIdHTTPSession') do begin
728: 652 ('TIdResponseHeaderInfo'), 'TIdHTTPResponseInfo') do begin
729: 653 ('TIdRequestHeaderInfo'), 'TIdHTTPRequestInfo') do begin
730: 654 ('TObject'), 'TIdHTTPSession');
731: 655 ('TObject'), 'TIdHTTPCustomSessionList');
732: 656 ('TObject'), 'TIdHTTPRequestInfo');
733: 657 ('TObject'), 'TIdHTTPResponseInfo');
734: 658 ('TObject'), 'TIdMimeTable');
735: 659 ('TObject'), 'EIdHTTPServerError');
736: 660 ('TObject'), 'EIdHTTPHeaderAlreadyWritten');
737: 661 ('TObject'), 'EIdHTTPErrorParsingCommand');
738: 662 ('TObject'), 'EIdHTTPUnsupportedAuthorisationScheme');
739: 663 ('TObject'), 'EIdHTTPCannotSwitchSessionStateWhenActive');
740: 663 ***** File 65: F:\maxbox3\mX397\maxbox3\source\upsi_IdDayTime.pas
741: 664 ('TIdTCPClient'), 'TIdDayTime') do
742: 664 ***** File 66: F:\maxbox3\mX397\maxbox3\source\upsi_IdEmailAddress.pas
743: 665 ('TOwnedCollection'), 'TIdEmailAddressList') do
744: 666 ('TCollectionItem'), 'TIdEmailAddressItem') do
745: 667 ('TObject'), 'EIdEmailParseError');
746: 667 ***** File 67: F:\maxbox3\mX397\maxbox3\source\upsi_IdFinger.pas
747: 668 ('TIdTCPClient'), 'TIdFinger') do
748: 668 ***** File 68: F:\maxbox3\mX397\maxbox3\source\upsi_IdGlobal.pas
749: 669 ('TCustomMemoryStream'), 'TIdReadStream') do
750: 670 ('TObject'), 'TIdMimeTable') do begin
751: 671 ('TEvent'), 'TIdLocalEvent') do begin
752: 672 ('TObject'), 'EIdFailedToRetrieveTimeZoneInfo');
753: 673 ('TObject'), 'EIdCorruptServicesFile');
754: 674 ('TObject'), 'EIdExtensionAlreadyExists');
755: 674 ***** File 69: F:\maxbox3\mX397\maxbox3\source\upsi_IdGlobal_10.pas
756: 675 ('TIdStream'), 'TIdBaseStream') do
757: 676 ('TIdList'), 'TIdExtList') do
758: 677 ('TCriticalSection'), 'TIdCriticalSection') do
759: 678 ('TEvent'), 'TIdLocalEvent') do
760: 679 ('TObject'), 'TCriticalSection') do
761: 680 ('TObject'), 'TEvent') do
762: 681 ('TIdFileStream'), 'TIdCreateStream') do
763: 682 ('TIdFileStream'), 'TIdReadFileNonExclusiveStream') do
764: 683 ('TIdFileStream'), 'TIdReadFileExclusiveStream') do
765: 684 ('TIdFileStream'), 'TIdAppendFileStream') do
766: 685 ('TObject'), 'EIdCorruptServicesFile');
767: 686 ('TObject'), 'EIdEndOfStream');
768: 687 ('TObject'), 'EIdInvalidIPv6Address');
769: 688 ('TObject'), 'TIdExtList');
770: 688 ***** File 70: F:\maxbox3\mX397\maxbox3\source\upsi_IdHash.pas
771: 689 ('TIdHash'), 'TIdHash128') do begin
772: 690 ('TIdHash'), 'TIdHash32') do begin
773: 691 ('TIdHash'), 'TIdHash16') do begin
774: 692 ('TObject'), 'TIdHash');
775: 692 ***** File 71: F:\maxbox3\mX397\maxbox3\source\upsi_IdHashCRC.pas
776: 693 ('TIdHash32'), 'TIdHashCRC32') do begin
777: 694 ('TIdHash16'), 'TIdHashCRC16') do begin
778: 694 ***** File 72: F:\maxbox3\mX397\maxbox3\source\upsi_IdHashMessageDigest.pas
779: 695 ('TIdHashMessageDigest4'), 'TIdHashMessageDigest5') do begin
780: 696 ('TIdHashMessageDigest4'), 'TIdHashMessageDigest4') do begin
781: 697 ('TIdHashMessageDigest'), 'TIdHashMessageDigest2') do begin
782: 698 ('TObject'), 'TIdHashMessageDigest');
783: 698 ***** File 73: F:\maxbox3\mX397\maxbox3\source\upsi_IdHashSHA1.pas
784: 699 ('TIdHash160'), 'TIdHashSHA1') do begin
785: 699 ***** File 74: F:\maxbox3\mX397\maxbox3\source\upsi_IdHeaderList.pas
786: 700 ('TStringList'), 'TIdHeaderList') do begin
787: 700 ***** File 75: F:\maxbox3\mX397\maxbox3\source\upsi_IdHTTPHeaderInfo.pas
788: 701 ('TIdEntityHeaderInfo'), 'TIdResponseHeaderInfo') do
789: 702 ('TIdEntityHeaderInfo'), 'TIdRequestHeaderInfo') do begin
790: 703 ('TPersistent'), 'TIdProxyConnectionInfo') do begin
791: 704 ('TPersistent'), 'TIdEntityHeaderInfo') do begin
792: 704 ***** File 76: F:\maxbox3\mX397\maxbox3\source\upsi_IdHTTPServer.pas
793: 705 ('TIdCustomHTTPServer'), 'TIdHTTPServer') do
794: 705 ***** File 77: F:\maxbox3\mX397\maxbox3\source\upsi_IdIcmpClient.pas
795: 706 ('TIdRawClient'), 'TIdIcmpClient') do begin
796: 706 ***** File 78: F:\maxbox3\mX397\maxbox3\source\upsi_IdIOHandlerSocket.pas
797: 707 ('TIdIOHandler'), 'TIdIOHandlerSocket') do begin
798: 708 ('TIdComponent'), 'TIdIOHandler') do begin
799: 708 ***** File 79: F:\maxbox3\mX397\maxbox3\source\upsi_IdLogFile.pas
800: 709 ('TIdLogBase'), 'TIdLogFile') do begin
801: 709 ***** File 80: F:\maxbox3\mX397\maxbox3\source\upsi_IdMailBox.pas
```

```

802: 710 ('TidBaseComponent'), 'TidMailBox') do
803: 710 ***** File 81: F:\maxbox3\mX397\maxbox3\source\uPSI_IdMessage.pas
804: 711 ('TidBaseComponent'), 'TidMessage') do begin
805: 712 ('TidOwnedCollection'), 'TidMessageParts') do
806: 713 ('TidMessagePart'), 'TidText') do
807: 714 ('TidMessagePart'), 'TidAttachment') do
808: 715 ('TidCollectionItem'), 'TidMessagePart') do
809: 716 ('TOBJECT'), 'TidMIMEBoundary') do
810: 717 ('TOBJECT'), 'TidMessageParts');
811: 718 ('TOBJECT'), 'EidMessageException');
812: 719 ('TOBJECT'), 'EidCanNotCreateMessagePart');
813: 720 ('TOBJECT'), 'EidTextInvalidCount');
814: 721 ('TOBJECT'), 'EidMessageCannotLoad');
815: 721 ***** File 82: F:\maxbox3\mX397\maxbox3\source\uPSI_IdMessageClient.pas
816: 722 ('TidTCPClient'), 'TidMessageClient') do
817: 723 ('TidIOHandlerStream'), 'TidIOHandlerStreamMsg') do
818: 723 ***** File 83: F:\maxbox3\mX397\maxbox3\source\uPSI_IdMultipartFormData.pas
819: 724 ('TStream'), 'TidMultiPartFormDataStream') do begin
820: 725 ('TCollection'), 'TidFormDataFields') do begin
821: 726 ('TCollectionItem'), 'TidFormDataField') do begin
822: 727 ('TOBJECT'), 'TidMultiPartFormDataStream');
823: 728 ('TOBJECT'), 'EidInvalidObjectType');
824: 728 ***** File 84: F:\maxbox3\mX397\maxbox3\source\uPSI_IdNetworkCalculator.pas
825: 729 ('TidBaseComponent'), 'TidNetworkCalculator') do
826: 730 ('TPersistent'), 'TipProperty') do
827: 730 ***** File 85: F:\maxbox3\mX397\maxbox3\source\uPSI_IdPOP3.pas
828: 731 ('TidMessageClient'), 'TidPOP3') do begin
829: 731 ***** File 86: F:\maxbox3\mX397\maxbox3\source\uPSI_IdQotd.pas
830: 732 ('TidTCPClient'), 'TidQOTD') do
831: 732 ***** File 87: F:\maxbox3\mX397\maxbox3\source\uPSI_IdSimpleServer.pas
832: 733 ('TidTCPConnection'), 'TidSimpleServer') do begin
833: 733 ***** File 88: F:\maxbox3\mX397\maxbox3\source\uPSI_IdSMTP.pas
834: 734 ('TidMessageClient'), 'TidSMTP') do
835: 734 ***** File 89: F:\maxbox3\mX397\maxbox3\source\uPSI_IdSNTP.pas
836: 735 ('TidUDPCClient'), 'TidSNTP') do begin
837: 735 ***** File 90: F:\maxbox3\mX397\maxbox3\source\uPSI_IdSocketHandle.pas
838: 736 ('TCollectionItem'), 'TidSocketHandle') do begin
839: 737 ('TidOwnedCollection'), 'TidSocketHandles') do begin
840: 738 ('TOBJECT'), 'TidSocketHandle');
841: 739 ('TOBJECT'), 'EidSocketHandleError');
842: 740 ('TOBJECT'), 'EidPackageSizeTooBig');
843: 741 ('TOBJECT'), 'EidNotAllBytesSent');
844: 742 ('TOBJECT'), 'EidCouldNotBindSocket');
845: 743 ('TOBJECT'), 'EidCanNotBindPortInRange');
846: 744 ('TOBJECT'), 'EidInvalidPortRange');
847: 744 ***** File 91: F:\maxbox3\mX397\maxbox3\source\uPSI_IdSSLOpenSSL.pas
848: 745 ('TObject'), 'TidSSLCipher') do begin
849: 746 ('TObject'), 'TidX509') do begin
850: 747 ('TObject'), 'TidX509Name') do begin
851: 748 ('TidServerIOHandler'), 'TidServerIOHandlerSSL') do begin
852: 749 ('TidIOHandlerSocket'), 'TidSSLIOHandlerSocket') do begin
853: 750 ('TObject'), 'TidSSLSocket') do begin
854: 751 ('TObject'), 'TidSSLContext') do begin
855: 752 ('TPersistent'), 'TidSSLOptions') do begin
856: 753 ('TOBJECT'), 'TidX509');
857: 754 ('TOBJECT'), 'TidSSLIOHandlerSocket');
858: 755 ('TOBJECT'), 'TidSSLCipher');
859: 756 ('TOBJECT'), 'EidOpenSSLError');
860: 757 ('TOBJECT'), 'EidOpenSSLLoadError');
861: 758 ('TOBJECT'), 'EidOSSLCouldNotLoadSSLLibrary');
862: 759 ('TOBJECT'), 'EidOSSLMODENotSet');
863: 760 ('TOBJECT'), 'EidOSSSLGetMethodError');
864: 761 ('TOBJECT'), 'EidOSSSLCreatingContextError');
865: 762 ('TOBJECT'), 'EidOSSSLLoadingRootCertError');
866: 763 ('TOBJECT'), 'EidOSSSLLoadingCertError');
867: 764 ('TOBJECT'), 'EidOSSSLLoadingKeyError');
868: 765 ('TOBJECT'), 'EidOSSSLSettingCipherError');
869: 766 ('TOBJECT'), 'EidOSSSLDataBindingError');
870: 767 ('TOBJECT'), 'EidOSSSLAcceptError');
871: 768 ('TOBJECT'), 'EidOSSSLConnectError');
872: 768 ***** File 92: F:\maxbox3\mX397\maxbox3\source\uPSI_IdTCPConnection.pas
873: 769 ('TidComponent'), 'TidTCPConnection') do begin
874: 770 ('TidSimpleBuffer'), 'TidManagedBuffer') do
875: 771 ('TMemoryStream'), 'TidSimpleBuffer') do
876: 772 ('TOBJECT'), 'EidTCPConnectionError');
877: 773 ('TOBJECT'), 'EidObjectTypeNotSupported');
878: 774 ('TOBJECT'), 'EidNotEnoughDataInBuffer');
879: 775 ('TOBJECT'), 'EidInterceptPropIsNil');
880: 776 ('TOBJECT'), 'EidInterceptPropInvalid');
881: 777 ('TOBJECT'), 'EidIOHandlerPropInvalid');
882: 778 ('TOBJECT'), 'EidNoDataToRead');
883: 779 ('TOBJECT'), 'EidNotConnected');
884: 780 ('TOBJECT'), 'EidFileNotFound');
885: 780 ***** File 93: F:\maxbox3\mX397\maxbox3\source\uPSI_IdTCPServer.pas
886: 781 ('TidComponent'), 'TidTCPServer') do begin
887: 782 ('TidThread'), 'TidPeerThread') do begin
888: 783 ('TidTCPConnection'), 'TidTCPServerConnection') do begin
889: 784 ('TidThread'), 'TidListenerThread') do begin
890: 785 ('TObject'), 'TidCommand') do

```

```

891: 786 ('TOwnedCollection'), 'TIdCommandHandlers') do begin
892: 787 ('TCollectionItem'), 'TIdCommandHandler') do
893: 788 ('TOBJECT'), 'TIdCommandHandler');
894: 789 ('TOBJECT'), 'TIdCommand');
895: 790 ('TOBJECT'), 'TIdPeerThread');
896: 791 ('TOBJECT'), 'TIdTCPServer');
897: 792 ('TOBJECT'), 'EIdTCPServerError');
898: 793 ('TOBJECT'), 'EIdNoExecuteSpecified');
899: 794 ('TOBJECT'), 'EIdTerminateThreadTimeout');
900: 794 ***** File 94: F:\maxbox3\mX397\maxbox3\source\upsi_IdTelnet.pas
901: 795 ('TIdTCPClient'), 'TIdTelnet') do
902: 796 ('TIdThread'), 'TIdTelnetReadThread') do
903: 797 ('TOBJECT'), 'TIdTelnet');
904: 798 ('TOBJECT'), 'EIdTelnetError');
905: 799 ('TOBJECT'), 'EIdTelnetClientConnectError');
906: 800 ('TOBJECT'), 'EIdTelnetServerOnDataAvailableIsNil');
907: 800 ***** File 95: F:\maxbox3\mX397\maxbox3\source\upsi_IdThread.pas
908: 801 ('TIdBaseThread'), 'TIdThread') do begin
909: 802 ('TThread'), 'TIdBaseThread') do begin
910: 803 ('TOBJECT'), 'EIdThreadException');
911: 804 ('TOBJECT'), 'EIdThreadTerminateAndWaitFor');
912: 805 ('TOBJECT'), 'TIdThread');
913: 805 ***** File 96: F:\maxbox3\mX397\maxbox3\source\upsi_IdTime.pas
914: 806 ('TIdTCPClient'), 'TIdTime') do
915: 806 ***** File 97: F:\maxbox3\mX397\maxbox3\source\upsi_IdTrivialFTP.pas
916: 807 ('TIdUDPCClient'), 'TIdTrivialFTP') do begin
917: 807 ***** File 98: F:\maxbox3\mX397\maxbox3\source\upsi_IdUDPBase.pas
918: 808 ('TIdComponent'), 'TIdUDPBase') do begin
919: 809 ('TOBJECT'), 'EIdUDPException');
920: 810 ('TOBJECT'), 'EIdUDPReceiveErrorZeroBytes');
921: 810 ***** File 99: F:\maxbox3\mX397\maxbox3\source\upsi_IdUDPCClient.pas
922: 811 ('TIdUDPBase'), 'TIdUDPCClient') do begin
923: 811 ***** File 100: F:\maxbox3\mX397\maxbox3\source\upsi_IdUserAccounts.pas
924: 812 ('TIdBaseComponent'), 'TIdUserManager') do begin
925: 813 ('TOwnedCollection'), 'TIdUserAccounts') do
926: 814 ('TCollectionItem'), 'TIdUserAccount') do
927: 815 ('TOBJECT'), 'TIdUserManager');
928: 815 ***** File 101: F:\maxbox3\mX397\maxbox3\source\upsi_ImgList.pas
929: 816 ('TComponent'), 'TCustomImageList') do
930: 817 ('TObject'), 'TChangeLink') do
931: 818 ('TOBJECT'), 'TCustomImageList');
932: 818 ***** File 102: F:\maxbox3\mX397\maxbox3\source\upsi_IMouse.pas
933: 819 ('TCustomPanningWindow'), 'TPanningWindow') do begin
934: 819 ***** File 103: F:\maxbox3\mX397\maxbox3\source\upsi_IniFiles.pas
935: 820 ('TCustomIniFile'), 'TIniFile') do
936: 821 ('TCustomIniFile'), 'TIniFile') do begin
937: 822 ('TMemIniFile'), 'TIniFile') do begin
938: 823 ('TCustomIniFile'), 'TMemIniFile') do begin
939: 824 ('TStringList'), 'THashedStringList') do
940: 825 ('TOBJECT'), 'TStringHash') do begin
941: 826 ('TObject'), 'TCustomIniFile') do begin
942: 827 ('TOBJECT'), 'EIniFileException');
943: 827 ***** File 104: F:\maxbox3\mX397\maxbox3\source\upsi_interface2_so.pas
944: 828 ('TInterfacedObject'), 'TIncomeRealSuper2') do begin
945: 829 ('TInterfacedObject'), 'TIncomeRealSuper') do begin
946: 830 ('TInterfacedObject'), 'TIncomeRealIntf') do begin
947: 830 ***** File 105: F:\maxbox3\mX397\maxbox3\source\upsi_JclBase.pas
948: 831 ('TList'), 'TObjectListJ') do begin
949: 832 ('EJclError'), 'EJclWin32Error') do begin
950: 833 ('Exception'), 'EJclError') do begin
951: 833 ***** File 106: F:\maxbox3\mX397\maxbox3\source\upsi_JclConsole.pas
952: 834 ('TOBJECT'), 'TJclInputBuffer') do
953: 835 ('TOBJECT'), 'TJclScreenWindow') do
954: 836 ('TOBJECT'), 'TJclScreenCursor') do
955: 837 ('TJclScreenCustomTextAttribute'), 'TJclScreenCharacter') do
956: 838 ('TJclScreenCustomTextAttribute'), 'TJclScreenTextAttribute') do
957: 839 ('TJclScreenCustomTextAttribute'), 'TJclScreenFont') do
958: 840 ('TInterfacedObject'), 'TJclScreenCustomTextAttribute') do
959: 841 ('TOBJECT'), 'TJclScreenBuffer') do
960: 842 ('TOBJECT'), 'TJclConsole') do
961: 843 ('TOBJECT'), 'TJclScreenBuffer');
962: 844 ('TOBJECT'), 'TJclInputBuffer');
963: 845 ('TOBJECT'), 'TJclScreenFont');
964: 846 ('TOBJECT'), 'TJclScreenCharacter');
965: 847 ('TOBJECT'), 'TJclScreenCursor');
966: 848 ('TOBJECT'), 'TJclScreenWindow');
967: 848 ***** File 107: F:\maxbox3\mX397\maxbox3\source\upsi_JclCounter.pas
968: 849 ('TObject'), 'TJclCounter') do
969: 850 ('TOBJECT'), 'EJclCounterError');
970: 850 ***** File 108: F:\maxbox3\mX397\maxbox3\source\upsi_JclFileUtils.pas
971: 851 ('TObject'), 'TJclFileMaskComparator') do begin
972: 852 ('TPersistent'), 'TJclMappedTextReader') do
973: 853 ('TCustomMemoryStream'), 'TJclFileMappingStream') do
974: 854 ('TJclCustomFileMapping'), 'TJclSwapFileMapping') do
975: 855 ('TJclCustomFileMapping'), 'TJclFileMapping') do
976: 856 ('TObject'), 'TJclCustomFileMapping') do
977: 857 ('TCustomMemoryStream'), 'TJclFileMappingView') do
978: 858 ('THandleStream'), 'TJclTempFileStream') do
979: 859 ('TObject'), 'TJclFileVersionInfo') do

```

```

980: 860 ('TOBJECT'), 'EJclFileVersionInfoError';
981: 861 ('TOBJECT'), 'TJclCustomFileMapping';
982: 862 ('TOBJECT'), 'EJclPathError';
983: 863 ('TOBJECT'), 'EJclFileUtilsError';
984: 864 ('TOBJECT'), 'EJclTempFileStreamError';
985: 865 ('TOBJECT'), 'EJclFileMappingError';
986: 866 ('TOBJECT'), 'EJclFileMappingViewError';
987: 866 ***** File 109: F:\maxbox3\mX397\maxbox3\source\upsi_JclMath.pas
988: 867 ('EJclMathError'), 'EJclNaNSignal' do
989: 868 ('TOBJECT'), 'TJclRational' do begin
990: 869 ('TJclASet'), 'TJclSparseFlatSet' do begin
991: 870 ('TJclASet'), 'TJclFlatSet' do begin
992: 871 ('TOBJECT'), 'TJclASet' do begin
993: 872 ('TOBJECT'), 'EJclMathError';
994: 872 ***** File 110: F:\maxbox3\mX397\maxbox3\source\upsi_JclMath_.pas
995: 873 ('EJclMathError'), 'EJclNaNSignal' do
996: 874 ('TOBJECT'), 'TJclRational' do begin
997: 875 ('TJclASet'), 'TJclSparseFlatSet' do begin
998: 876 ('TJclASet'), 'TJclFlatSet' do begin
999: 877 ('TOBJECT'), 'TJclASet' do begin
1000: 878 ('TOBJECT'), 'EJclMathError';
1001: 878 ***** File 111: F:\maxbox3\mX397\maxbox3\source\upsi_JclMiscel.pas
1002: 879 ('TOBJECT'), 'EJclCreateProcessError';
1003: 879 ***** File 112: F:\maxbox3\mX397\maxbox3\source\upsi_JclMultimedia.pas
1004: 880 ('EJclError'), 'EJclMciError' do begin
1005: 881 ('TOBJECT'), 'TJclMixer' do begin
1006: 882 ('TOBJECT'), 'TJclMixerDevice' do begin
1007: 883 ('TJclMixerLine'), 'TJclMixerDestination' do
1008: 884 ('TJclMixerLine'), 'TJclMixerSource' do
1009: 885 ('TOBJECT'), 'TJclMixerLine' do
1010: 886 ('TOBJECT'), 'TJclMixerLineControl' do
1011: 887 ('TOBJECT'), 'TJclMultimediaTimer' do begin
1012: 888 ('TOBJECT'), 'EJclMmTimerError';
1013: 889 ('TOBJECT'), 'EJclMixerError';
1014: 890 ('TOBJECT'), 'TJclMixerDevice';
1015: 891 ('TOBJECT'), 'TJclMixerLine';
1016: 892 ('TOBJECT'), 'TJclMixerDestination';
1017: 892 ***** File 113: F:\maxbox3\mX397\maxbox3\source\upsi_JclSimpleXml.pas
1018: 893 ('TInvokeableVariantType'), 'TXMLVariant' do
1019: 894 ('TOBJECT'), 'TJclSimpleXML' do begin
1020: 895 ('TJclSimpleXMLElement'), 'TJclSimpleXMLElementMSOApplication' do
1021: 896 ('TJclSimpleXMLElement'), 'TJclSimpleXMLElementSheet' do
1022: 897 ('TJclSimpleXMLElement'), 'TJclSimpleXMLElementDocType' do
1023: 898 ('TJclSimpleXMLElement'), 'TJclSimpleXMLElementHeader' do
1024: 899 ('TJclSimpleXMLElement'), 'TJclSimpleXMLElementText' do
1025: 900 ('TJclSimpleXMLElement'), 'TJclSimpleXMLElementCData' do
1026: 901 ('TJclSimpleXMLElement'), 'TJclSimpleXMLElementClassic' do
1027: 902 ('TJclSimpleXMLElement'), 'TJclSimpleXMLElementComment' do
1028: 903 ('TOBJECT'), 'TJclSimpleXMLElement' do begin
1029: 904 ('TOBJECT'), 'TJclSimpleXMLElements' do begin
1030: 905 ('TOBJECT'), 'TJclSimpleXMLNamedElems' do begin
1031: 906 ('TOBJECT'), 'TJclSimpleXMLElementsProlog' do begin
1032: 907 ('TOBJECT'), 'TJclSimpleXMLProps' do begin
1033: 908 ('TOBJECT'), 'TJclSimpleXMLProp' do begin
1034: 909 ('TOBJECT'), 'TJclSimpleXML';
1035: 910 ('TOBJECT'), 'EJclSimpleXMLError';
1036: 911 ('TOBJECT'), 'TJclSimpleXMLElement';
1037: 912 ('TOBJECT'), 'TJclSimpleXMLElements';
1038: 913 ('TOBJECT'), 'TJclSimpleXMLProps';
1039: 914 ('TOBJECT'), 'TJclSimpleXMLElementComment';
1040: 915 ('TOBJECT'), 'TJclSimpleXMLElementClassic';
1041: 916 ('TOBJECT'), 'TJclSimpleXMLElementCData';
1042: 917 ('TOBJECT'), 'TJclSimpleXMLElementDocType';
1043: 918 ('TOBJECT'), 'TJclSimpleXMLElementText';
1044: 919 ('TOBJECT'), 'TJclSimpleXMLElementHeader';
1045: 920 ('TOBJECT'), 'TJclSimpleXMLElementSheet';
1046: 921 ('TOBJECT'), 'TJclSimpleXMLElementMSOApplication';
1047: 921 ***** File 114: F:\maxbox3\mX397\maxbox3\source\upsi_JclSimpleXml_test.pas
1048: 922 ('TInvokeableVariantType'), 'TXMLVariant' do
1049: 923 ('TOBJECT'), 'TJclSimpleXML' do
1050: 924 ('TJclSimpleXMLElement'), 'TJclSimpleXMLElementMSOApplication' do
1051: 925 ('TJclSimpleXMLElement'), 'TJclSimpleXMLElementSheet' do
1052: 926 ('TJclSimpleXMLElement'), 'TJclSimpleXMLElementDocType' do
1053: 927 ('TJclSimpleXMLElement'), 'TJclSimpleXMLElementHeader' do
1054: 928 ('TJclSimpleXMLElement'), 'TJclSimpleXMLElementText' do
1055: 929 ('TJclSimpleXMLElement'), 'TJclSimpleXMLElementCData' do
1056: 930 ('TJclSimpleXMLElement'), 'TJclSimpleXMLElementClassic' do
1057: 931 ('TJclSimpleXMLElement'), 'TJclSimpleXMLElementComment' do
1058: 932 ('TOBJECT'), 'TJclSimpleXMLElement' do
1059: 933 ('TOBJECT'), 'TJclSimpleXMLElements' do
1060: 934 ('TOBJECT'), 'TJclSimpleXMLNamedElems' do
1061: 935 ('TOBJECT'), 'TJclSimpleXMLElementsProlog' do
1062: 936 ('TOBJECT'), 'TJclSimpleXMLProps' do
1063: 937 ('TOBJECT'), 'TJclSimpleXMLProp' do
1064: 938 ('TOBJECT'), 'TJclSimpleXML';
1065: 939 ('TOBJECT'), 'EJclSimpleXMLError';
1066: 940 ('TOBJECT'), 'TJclSimpleXMLElement';
1067: 941 ('TOBJECT'), 'TJclSimpleXMLElements';
1068: 942 ('TOBJECT'), 'TJclSimpleXMLProps';

```

```

1069: 943 ('TOBJECT'), 'TJclSimpleXMLElementComment';
1070: 944 ('TOBJECT'), 'TJclSimpleXMLElementClassic';
1071: 945 ('TOBJECT'), 'TJclSimpleXMLElementCData';
1072: 946 ('TOBJECT'), 'TJclSimpleXMLElementDocType';
1073: 947 ('TOBJECT'), 'TJclSimpleXMLElementText';
1074: 948 ('TOBJECT'), 'TJclSimpleXMLElementHeader';
1075: 949 ('TOBJECT'), 'TJclSimpleXMLElementSheet';
1076: 950 ('TOBJECT'), 'TJclSimpleXMLElementMSOApplication';
1077: 950 ***** File 115: F:\maxbox3\mX397\maxbox3\source\upsi_jpeg.pas
1078: 951 ('TGraphic'), 'TJPEGImage') do begin
1079: 952 ('TSharedImage'), 'TJPEGData') do
1080: 952 ***** File 116: F:\maxbox3\mX397\maxbox3\source\upsi_JvBdeUtils.pas
1081: 953 ('TTable'), 'TJvCloneTable') do
1082: 954 ('TDBDataSet'), 'TJvCloneDbDataset') do
1083: 955 ('TBDEDataSet'), 'TJvCloneDataset') do
1084: 956 ('TJvLocateObject'), 'TJvDBLocate') do
1085: 956 ***** File 117: F:\maxbox3\mX397\maxbox3\source\upsi_JvCalc.pas
1086: 957 ('TForm'), 'TJvCalculatorForm') do
1087: 958 ('TJvCommonDialogF'), 'TJvCalculator') do begin
1088: 959 ('TOBJECT'), 'TJvCalculatorForm';
1089: 959 ***** File 118: F:\maxbox3\mX397\maxbox3\source\upsi_JvCtrls.pas
1090: 960 ('TTreeView'), 'TJvTreeView') do begin
1091: 961 ('TTreeNode'), 'TJvTreeNode') do begin
1092: 962 ('TTrackBar'), 'TJvTrackBar') do begin
1093: 963 ('TPageControl'), 'TJvPageControl') do begin
1094: 964 ('TWinControl'), 'TJvIpAddress') do begin
1095: 965 ('TPersistent'), 'TJvIpAddressValues') do begin
1096: 966 ('TPersistent'), 'TJvIpAddressRange') do begin
1097: 967 ('TOBJECT'), 'TJvIpAddress';
1098: 967 ***** File 119: F:\maxbox3\mX397\maxbox3\source\upsi_JvComponent.pas
1099: 968 ('TCheckBox'), 'TJvMultiselectCheckBox') do
1100: 969 ('TCustomMemo'), 'TJvCustomMemo') do
1101: 970 ('TForm'), 'TJvForm') do
1102: 971 ('TWinControl'), 'TJvWinControl') do
1103: 972 ('TCustomControl'), 'TJvCustomControl') do
1104: 973 ('TCustomPanel'), 'TJvCustomPanel') do
1105: 974 ('TGraphicControl'), 'TJvGraphicControl') do
1106: 975 ('TComponent'), 'TJvComponent') do
1107: 975 ***** File 120: F:\maxbox3\mX397\maxbox3\source\upsi_JvComputerInfo.pas
1108: 976 ('TJvComponent'), 'TJvComputerInfo') do
1109: 976 ***** File 121: F:\maxbox3\mX397\maxbox3\source\upsi_JvCtrls.pas
1110: 977 ('TButton'), 'TJvImgBtn') do begin
1111: 978 ('TButtonActionLink'), 'TJvImgBtnActionLink') do
1112: 979 ('TJvCustomListBox'), 'TJvListBox') do
1113: 980 ('TOBJECT'), 'TJvImgBtn';
1114: 980 ***** File 122: F:\maxbox3\mX397\maxbox3\source\upsi_JvDBUtil.pas
1115: 981 ('Exception'), 'EJvScriptError') do
1116: 981 ***** File 123: F:\maxbox3\mX397\maxbox3\source\upsi_JvDBUtils.pas
1117: 982 ('TObject'), 'TJvLocateObject') do
1118: 982 ***** File 124: F:\maxbox3\mX397\maxbox3\source\upsi_JvExprParser.pas
1119: 983 ('TOBJECT'), 'TExprParser') do begin
1120: 984 ('TOBJECT'), 'EExprParserError';
1121: 984 ***** File 125: F:\maxbox3\mX397\maxbox3\source\upsi_JvFindFiles.pas
1122: 985 ('TJvCommonDialogF'), 'TJvFindFilesDialog') do begin
1123: 985 ***** File 126: F:\maxbox3\mX397\maxbox3\source\upsi_JvFormToHtml.pas
1124: 986 ('TJvComponent'), 'TJvFormToHtml') do begin
1125: 986 ***** File 127: F:\maxbox3\mX397\maxbox3\source\upsi_JvgCommClasses.pas
1126: 987 ('TPersistent'), 'TJvgBevelLines') do begin
1127: 988 ('TJvgCustomTextBoxStyle'), 'TJvgTextBoxStyle') do
1128: 989 ('TJvgCustomBoxStyle'), 'TJvgCustomTextBoxStyle') do begin
1129: 990 ('TJvgBevelOptions'), 'TJvgCustomBoxStyle') do begin
1130: 991 ('TJvgCustomListBoxItemStyle'), 'TJvgAskListBoxItemStyle') do
1131: 992 ('TJvgListBoxItemStyle'), 'TJvgSpeedButtonStyle') do
1132: 993 ('TJvgCustomListBoxItemStyle'), 'TJvgListBoxItemStyle') do
1133: 994 ('TPersistent'), 'TJvgCustomListBoxItemStyle') do begin
1134: 995 ('TJvgCustomLabelColors'), 'TJvgGroupBoxColors') do begin
1135: 996 ('TJvgCustomLabelColors'), 'TJvgLabelColors') do
1136: 997 ('TJvgCustomTextColors'), 'TJvgCustomLabelColors') do
1137: 998 ('TJvgCustomTextColors'), 'TJvgSimleLabelColors') do
1138: 999 ('TPersistent'), 'TJvgCustomTextColors') do begin
1139: 1000 ('TPersistent'), 'TJvgLabelTextStyles') do begin
1140: 1001 ('TJvg2DAlign'), 'TJvgIllumination') do begin
1141: 1002 ('TJvgBevelOptions'), 'TJvgExtBevelOptions') do begin
1142: 1003 ('TPersistent'), 'TJvgBevelOptions') do begin
1143: 1004 ('TPersistent'), 'TJvgPointClass') do begin
1144: 1005 ('TPersistent'), 'TJvg2DAlign') do begin
1145: 1006 ('TJvgCustomGradient'), 'TJvg3DGradient') do begin
1146: 1007 ('TJvgCustomGradient'), 'TJvgGradient') do begin
1147: 1008 ('TJvgTwainColors'), 'TJvgCustomGradient') do begin
1148: 1009 ('TPersistent'), 'TJvgTwainColors') do begin
1149: 1010 ('TOBJECT'), 'TJvgTwainColors';
1150: 1011 ('TOBJECT'), 'TJvgCustomGradient';
1151: 1012 ('TOBJECT'), 'TJvgGradient';
1152: 1013 ('TOBJECT'), 'TJvg3DGradient';
1153: 1014 ('TOBJECT'), 'TJvg2DAlign';
1154: 1015 ('TOBJECT'), 'TJvgPointClass';
1155: 1016 ('TOBJECT'), 'TJvgBevelOptions';
1156: 1017 ('TOBJECT'), 'TJvgExtBevelOptions';
1157: 1018 ('TOBJECT'), 'TJvgIllumination';

```



```

1158: 1019 ('TOBJECT'), 'TJvgLabelTextStyles';
1159: 1020 ('TOBJECT'), 'TJvgCustomTextColors';
1160: 1021 ('TOBJECT'), 'TJvgSimleLabelColors';
1161: 1022 ('TOBJECT'), 'TJvgCustomLabelColors';
1162: 1023 ('TOBJECT'), 'TJvgLabelColors';
1163: 1024 ('TOBJECT'), 'TJvgGroupBoxColors';
1164: 1025 ('TOBJECT'), 'TJvgListBoxItemStyle';
1165: 1026 ('TOBJECT'), 'TJvgAskListBoxItemStyle';
1166: 1027 ('TOBJECT'), 'TJvgCustomBoxStyle';
1167: 1028 ('TOBJECT'), 'TJvgCustomTextBorderStyle';
1168: 1029 ('TOBJECT'), 'TJvgTextBoxStyle';
1169: 1030 ('TOBJECT'), 'TJvgBevellLines';
1170: 1031 ('TOBJECT'), 'TJvgHintStyle';
1171: 1031 ***** File 128: F:\maxbox3\mX397\maxbox3\source\upsi_JvGenetic.pas
1172: 1032 ('TJvComponent'), 'TJvGenetic' do begin
1173: 1032 ***** File 129: F:\maxbox3\mX397\maxbox3\source\upsi_JvgLogics.pas
1174: 1033 ('TOwnedCollection'), 'TJvgCommentAreas' do
1175: 1034 ('TCollectionItem'), 'TJvgCommentArea' do
1176: 1035 ('TOwnedCollection'), 'TJvgLogicVariants' do
1177: 1036 ('TCollectionItem'), 'TJvgLogicVariant' do
1178: 1037 ('TOwnedCollection'), 'TJvgLogics' do begin
1179: 1038 ('TCollectionItem'), 'TJvgLogicElement' do
1180: 1039 ('TJvComponent'), 'TJvgLogicProducer' do
1181: 1040 ('TOBJECT'), 'TJvgLogics';
1182: 1041 ('TOBJECT'), 'TJvgLogicElement';
1183: 1042 ('TOBJECT'), 'TJvgCommentAreas';
1184: 1043 ('TOBJECT'), 'TJvgLogicVariants';
1185: 1044 ('TOBJECT'), 'TJvgLogicVariant';
1186: 1045 ('TOBJECT'), 'TJvgLogicProducer';
1187: 1045 ***** File 130: F:\maxbox3\mX397\maxbox3\source\upsi_JvGraph.pas
1188: 1046 ('TPersistent'), 'TJvGradient' do begin
1189: 1046 ***** File 131: F:\maxbox3\mX397\maxbox3\source\upsi_JvgUtils.pas
1190: 1047 ('TWinControl'), 'TJvgPublicWinControl' do begin
1191: 1048 ('TOBJECT'), 'EJvCException';
1192: 1048 ***** File 132: F:\maxbox3\mX397\maxbox3\source\upsi_JvgXMLSerializer.pas
1193: 1049 ('TJvComponent'), 'TJvgXMLSerializer' do begin
1194: 1050 ('TOBJECT'), 'EJvgXMLSerializerException';
1195: 1051 ('TOBJECT'), 'XMLSerializerException';
1196: 1052 ('TOBJECT'), 'EJvgXMLOpenTagNotFoundException';
1197: 1053 ('TOBJECT'), 'EJvgXMLCloseTagNotFoundException';
1198: 1054 ('TOBJECT'), 'EJvgXMLUnknownPropertyException';
1199: 1054 ***** File 133: F:\maxbox3\mX397\maxbox3\source\upsi_JvHtmlParser.pas
1200: 1055 ('TJvComponent'), 'TJvHTMLParser' do begin
1201: 1056 ('TList'), 'TTagInfoList' do
1202: 1057 ('TObject'), 'TJvParserInfo' do begin
1203: 1057 ***** File 134: F:\maxbox3\mX397\maxbox3\source\upsi_JvJCLUtils.pas
1204: 1058 ('TList'), 'TIntegerList' do begin
1205: 1059 ('TOBJECT'), 'EJvConvertError';
1206: 1059 ***** File 135: F:\maxbox3\mX397\maxbox3\source\upsi_JvKeyboardStates.pas
1207: 1060 ('TJvComponent'), 'TJvKeyboardStates' do begin
1208: 1060 ***** File 136: F:\maxbox3\mX397\maxbox3\source\upsi_JvLED.pas
1209: 1061 ('TJvCustomLED'), 'TJvLED' do begin
1210: 1062 ('TJvGraphicControl'), 'TJvCustomLED' do begin
1211: 1062 ***** File 137: F:\maxbox3\mX397\maxbox3\source\upsi_JvLogFile.pas
1212: 1063 ('TJvComponent'), 'TJvLogFile' do begin
1213: 1064 ('TObject'), 'TJvLogRecord' do begin
1214: 1064 ***** File 138: F:\maxbox3\mX397\maxbox3\source\upsi_JvMail.pas
1215: 1065 ('TJvComponent'), 'TJvMail' do begin
1216: 1066 ('TCollection'), 'TJvMailRecipients' do begin
1217: 1067 ('TCollectionItem'), 'TJvMailRecipient' do begin
1218: 1068 ('TOBJECT'), 'TJvMail';
1219: 1068 ***** File 139: F:\maxbox3\mX397\maxbox3\source\upsi_JvMemoryInfos.pas
1220: 1069 ('TJvComponent'), 'TJvMemoryInfos' do
1221: 1069 ***** File 140: F:\maxbox3\mX397\maxbox3\source\upsi_JvParsing.pas
1222: 1070 ('TObject'), 'TJvMathParser' do begin
1223: 1071 ('TOBJECT'), 'EJvParserError';
1224: 1071 ***** File 141: F:\maxbox3\mX397\maxbox3\source\upsi_JvSimpleXml.pas
1225: 1072 ('TPersistent'), 'TJvMaxPixel' do begin
1226: 1073 ('TObject'), 'TJvItemsSearchs' do begin
1227: 1074 ('TComponent'), 'TJvSimpleXML' do begin
1228: 1075 ('TJclSimpleXML'), 'TJclHackSimpleXML' do
1229: 1076 ('TJvXMLNode'), 'TJvXMLTree';
1230: 1076 ***** File 142: F:\maxbox3\mX397\maxbox3\source\upsi_JvTurtle.pas
1231: 1077 ('TComponent'), 'TJvTurtle' do begin
1232: 1077 ***** File 143: F:\maxbox3\mX397\maxbox3\source\upsi_JvVCLUtils.pas
1233: 1078 ('THandleStream'), 'TResourceStream' do begin
1234: 1079 ('TCanvas'), 'TMetafileCanvas' do
1235: 1080 ('TObject'), 'TBits' do begin
1236: 1081 ('TCanvas'), 'TJvScreenCanvas' do begin
1237: 1081 ***** File 144: F:\maxbox3\mX397\maxbox3\source\upsi_JvXmlDatabase.pas
1238: 1082 ('TJvComponent'), 'TJvXMLDatabase' do begin
1239: 1083 ('TObject'), 'TJvXMLQuery' do begin
1240: 1084 ('TObject'), 'TJvXMLQueryParser' do begin
1241: 1085 ('TObject'), 'TJvXMLQueryAssignment' do begin
1242: 1086 ('TObject'), 'TJvXMLQueryCondition' do begin
1243: 1087 ('TObject'), 'TJvXMLQueryOrder' do
1244: 1088 ('TObject'), 'TJvXMLQueryColumn' do
1245: 1089 ('TObject'), 'TJvXMLQueryTable' do
1246: 1090 ('TObject'), 'TJvXMLTable' do

```

```

1247: 1091 ('TOBJECT'), 'TJvXMLDatabase');
1248: 1092 ('TOBJECT'), 'TJvXMLQuery');
1249: 1093 ('TOBJECT'), 'TJvXMLQueryParser');
1250: 1094 ('TOBJECT'), 'TJvXMLDatabaseException');
1251: 1094 ***** File 145: F:\maxbox3\mX397\maxbox3\source\upsi_LinarBitmap.pas
1252: 1095 ('TMonitorObject'), 'TBitmapLoaders') do begin
1253: 1096 ('TGraphic'), 'TLinearGraphic') do begin
1254: 1097 ('TMonitorObject'), 'TBitmapLoader') do begin
1255: 1098 ('TAssignObject'), 'TLinearBitmap') do begin
1256: 1099 ('TOBJECT'), 'TBitmapLoader');
1257: 1100 ('TOBJECT'), 'ELinearBitmap');
1258: 1101 ('TOBJECT'), 'EUnsupportedFileFormat');
1259: 1101 ***** File 146: F:\maxbox3\mX397\maxbox3\source\upsi_Masks.pas
1260: 1102 ('TOBJECT'), 'TMask') do begin
1261: 1103 ('TOBJECT'), 'EMaskException');
1262: 1103 ***** File 147: F:\maxbox3\mX397\maxbox3\source\upsi_MathMax.pas
1263: 1104 ('EMathError'), 'EInvalidArgument') do
1264: 1105 ('TOBJECT'), 'EJclMathError');
1265: 1105 ***** File 148: F:\maxbox3\mX397\maxbox3\source\upsi_MPlayer.pas
1266: 1106 ('TCustomControl'), 'TMediaPlayer') do begin
1267: 1107 ('TOBJECT'), 'EMCIDeviceError');
1268: 1107 ***** File 149: F:\maxbox3\mX397\maxbox3\source\upsi_MyBigInt.pas
1269: 1108 ('TOBJECT'), 'TMyBigInt') do begin
1270: 1108 ***** File 150: F:\maxbox3\mX397\maxbox3\source\upsi_niExtendedRegularExpression.pas
1271: 1109 ('TniTestCase'), 'TniTestExtendedRegularExpression') do
1272: 1110 ('TniRegularExpression'), 'TniSubRegularExpression') do
1273: 1111 ('TObject'), 'TniExtendedRegularExpression') do
1274: 1112 ('TOBJECT'), 'TniSubRegularExpression');
1275: 1112 ***** File 151: F:\maxbox3\mX397\maxbox3\source\upsi_niRegularExpression.pas
1276: 1113 ('TniTestCase'), 'TniTestRegularExpression') do
1277: 1114 ('TObject'), 'TniRegularExpressionTransition') do
1278: 1115 ('TObject'), 'TniRegularExpressionState') do
1279: 1116 ('TObject'), 'TniRegularExpressionMatch') do
1280: 1117 ('TObject'), 'TniRegularExpressionMatcher') do
1281: 1118 ('TObject'), 'TniRegularExpression') do
1282: 1119 ('TOBJECT'), 'TniRegularExpression');
1283: 1120 ('TOBJECT'), 'TniRegularExpressionState');
1284: 1121 ('TOBJECT'), 'TniRegularExpressionTransition');
1285: 1122 ('TOBJECT'), 'TniRegularExpressionMatch');
1286: 1123 ('TOBJECT'), 'EniRegularExpressionError');
1287: 1123 ***** File 152: F:\maxbox3\mX397\maxbox3\source\upsi_niSTRING.pas
1288: 1124 ('TOBJECT'), 'EniStringError');
1289: 1124 ***** File 153: F:\maxbox3\mX397\maxbox3\source\upsi_OpenSSLUtils.pas
1290: 1125 ('TOBJECT'), 'TMessageSigner') do
1291: 1126 ('TOBJECT'), 'TPKCS7') do
1292: 1127 ('TOBJECT'), 'TX509Certificate') do
1293: 1128 ('TOBJECT'), 'TKeyPairGenerator') do
1294: 1129 ('Exception'), 'EOpenSSL') do
1295: 1129 ***** File 154: F:\maxbox3\mX397\maxbox3\source\upsi_Outline.pas
1296: 1130 ('TCustomOutline'), 'TOutline') do
1297: 1131 ('TCustomGrid'), 'TCustomOutline') do begin
1298: 1132 ('TPersistent'), 'TOutlineNode') do begin
1299: 1133 ('TOBJECT'), 'OutlineError');
1300: 1134 ('TOBJECT'), 'EOutlineError');
1301: 1135 ('TOBJECT'), 'TCustomOutline');
1302: 1135 ***** File 155: F:\maxbox3\mX397\maxbox3\source\upsi_ParallelJobs.pas
1303: 1136 ('TOBJECT'), 'TParallelJob') do begin
1304: 1137 ('TOBJECT'), 'TParallelJobLocker') do begin
1305: 1138 ('TOBJECT'), 'TParallelJobSemaphore') do begin
1306: 1139 ('TOBJECT'), 'TParallelJobInfo') do begin
1307: 1140 ('TOBJECT'), 'TJobsGroup') do begin
1308: 1141 ('TOBJECT'), 'TParallelJob');
1309: 1141 ***** File 156: F:\maxbox3\mX397\maxbox3\source\upsi_PerlRegEx.pas
1310: 1142 ('TOBJECT'), 'TPerlRegExList') do begin
1311: 1143 ('TOBJECT'), 'TPerlRegEx') do begin
1312: 1143 ***** File 157: F:\maxbox3\mX397\maxbox3\source\upsi_PNGLoader.pas
1313: 1144 ('TBitmapLoader'), 'TPNGLoader') do begin
1314: 1145 ('TLinearGraphic'), 'TMNGGraphic') do begin
1315: 1146 ('TLinearGraphic'), 'TPNGGraphic') do begin
1316: 1146 ***** File 158: F:\maxbox3\mX397\maxbox3\source\upsi_Printers.pas
1317: 1147 ('TObject'), 'TPrinter') do
1318: 1148 ('TOBJECT'), 'EPrinter');
1319: 1148 ***** File 159: F:\maxbox3\mX397\maxbox3\source\upsi_Provider.pas
1320: 1149 ('TInterfacedObject'), 'TLocalAppServer') do
1321: 1150 ('TCustomResolver'), 'TSQLResolver') do
1322: 1151 ('TCustomResolver'), 'TDataSetResolver') do begin
1323: 1152 ('TComponent'), 'TCustomResolver') do begin
1324: 1153 ('TObject'), 'TUpdateTree') do begin
1325: 1154 ('TDataSetProvider'), 'TProvider') do
1326: 1155 ('TBaseProvider'), 'TDataSetProvider') do begin
1327: 1156 ('TCustomProvider'), 'TBaseProvider') do begin
1328: 1157 ('TComponent'), 'TCustomProvider') do begin
1329: 1158 ('TCustomClientDataSet'), 'TPacketDataSet') do
1330: 1159 ('TCustomPacketWriter'), 'TDataPacketWriter') do
1331: 1160 ('TObject'), 'TCustomPacketWriter') do
1332: 1161 ('Exception'), 'EDSWriter') do
1333: 1162 ('TOBJECT'), 'TUpdateTree');
1334: 1163 ('TOBJECT'), 'TCustomResolver');
1335: 1163 ***** File 160: F:\maxbox3\mX397\maxbox3\source\upsi_ScktComp.pas

```

```

1336: 1164 ('TCustomServerSocket'), 'TServerSocket') do
1337: 1165 ('TCustomSocket'), 'TCustomServerSocket') do
1338: 1166 ('TCustomSocket'), 'TClientSocket') do
1339: 1167 ('TStream'), 'TWinSocketStream') do
1340: 1168 ('TAbstractSocket'), 'TCustomSocket') do
1341: 1169 ('TComponent'), 'TAbstractSocket') do
1342: 1170 ('TThread'), 'TServerClientThread') do
1343: 1171 ('TThread'), 'TServerAcceptThread') do
1344: 1172 ('TCustomWinSocket'), 'TServerWinSocket') do
1345: 1173 ('TCustomWinSocket'), 'TServerClientWinSocket') do
1346: 1174 ('TCustomWinSocket'), 'TClientWinSocket') do
1347: 1175 ('TObject'), 'TCustomWinSocket') do begin
1348: 1176 ('TObject'), 'ESocketError');
1349: 1177 ('TObject'), 'TSocket');
1350: 1178 ('TObject'), 'TSocketErrorProc');
1351: 1179 ('TObject'), 'TCustomWinSocket');
1352: 1180 ('TObject'), 'TCustomSocket');
1353: 1181 ('TObject'), 'TServerAcceptThread');
1354: 1182 ('TObject'), 'TServerClientThread');
1355: 1183 ('TObject'), 'TServerWinSocket');
1356: 1184 ('TObject'), 'TServerClientWinSocket');
1357: 1184 ***** File 161: F:\maxbox3\mX397\maxbox3\source\upsi_sdpStopwatch.pas
1358: 1185 ('TObject'), 'TStopwatch') do
1359: 1185 ***** File 162: F:\maxbox3\mX397\maxbox3\source\upsi_SerDlgs.pas
1360: 1186 ('TSerRTSCheckBox'), 'TSerDTRCheckBox') do
1361: 1187 ('TCustomCheckBox'), 'TSerRTSCheckBox') do
1362: 1188 ('TSerComboBox'), 'TSerHandshakeComboBox') do
1363: 1189 ('TSerComboBox'), 'TSerParityBitComboBox') do begin
1364: 1190 ('TSerComboBox'), 'TSerStopBitsComboBox') do begin
1365: 1191 ('TSerComboBox'), 'TSerDataBitsComboBox') do
1366: 1192 ('TSerComboBox'), 'TSerBaudComboBox') do
1367: 1193 ('TSerComboBox'), 'TSerPortComboBox') do
1368: 1194 ('TCustomComboBox'), 'TSerComboBox') do
1369: 1194 ***** File 163: F:\maxbox3\mX397\maxbox3\source\upsi_Serial.pas
1370: 1195 ('TComponent'), 'TSerStdDlg') do begin
1371: 1196 ('TCustomSerial'), 'TSerPort') do begin
1372: 1197 ('TCustomSerial'), 'TSerial') do begin
1373: 1198 ('TComponent'), 'TCustomSerial') do begin
1374: 1199 ('TThread'), 'TSerPortThread') do
1375: 1200 ('TThread'), 'TSerThread') do
1376: 1201 ('TObject'), 'TCOMManager') do begin
1377: 1202 ('TObject'), 'TMsgWindow') do begin
1378: 1203 ('TObject'), 'TCustomSerial');
1379: 1204 ('TObject'), 'TSerial');
1380: 1205 ('TObject'), 'TSerPort');
1381: 1205 ***** File 164: F:\maxbox3\mX397\maxbox3\source\upsi_ShadowWnd.pas
1382: 1206 ('TCustomControl'), 'TShadowWindow') do
1383: 1206 ***** File 165: F:\maxbox3\mX397\maxbox3\source\upsi_SOAPHTTPClient.pas
1384: 1207 ('TRIO'), 'THTTTPRIO') do begin
1385: 1207 ***** File 166: F:\maxbox3\mX397\maxbox3\source\upsi_SortThds.pas
1386: 1208 ('TSortThread'), 'TQuickSort') do
1387: 1209 ('TSortThread'), 'TSelectionSort') do
1388: 1210 ('TSortThread'), 'TBubbleSort') do
1389: 1211 ('TRandomSuper'), 'TRandomArray') do begin
1390: 1212 ('TObject'), 'TRandomSuper') do begin
1391: 1213 ('TThread'), 'TSortThread') do begin
1392: 1213 ***** File 167: F:\maxbox3\mX397\maxbox3\source\upsi_SqlExpr.pas
1393: 1214 ('TCustomSQLDataSet'), 'TSQLTable') do begin
1394: 1215 ('TCustomSQLDataSet'), 'TSQLStoredProc') do
1395: 1216 ('TCustomSQLDataSet'), 'TSQLQuery') do begin
1396: 1217 ('TCustomSQLDataSet'), 'TSQLDataSet') do begin
1397: 1218 ('TWideDataSet'), 'TCustomSQLDataSet') do begin
1398: 1219 ('TObject'), 'TFLDesc') do begin
1399: 1220 ('TDetailDataLink'), 'TSQLDataLink') do begin
1400: 1221 ('TCustomConnection'), 'TSQLConnection') do begin
1401: 1222 ('TObject'), 'TTransactionItem') do
1402: 1223 ('TComponent'), 'TSQLMonitor') do begin
1403: 1224 ('TMemoryStream'), 'TSQLBlobStream') do
1404: 1225 ('TObject'), 'SPPParamDesc') do begin
1405: 1226 ('TObject'), 'TSQLConnection');
1406: 1227 ('TObject'), 'TCustomSQLDataSet');
1407: 1228 ('TObject'), 'TSQLDataSet');
1408: 1229 ('TObject'), 'TSQLQuery');
1409: 1230 ('TObject'), 'TSQLStoredProc');
1410: 1231 ('TObject'), 'TSQLTable');
1411: 1231 ***** File 168: F:\maxbox3\mX397\maxbox3\source\upsi_StDecMth.pas
1412: 1232 ('TObject'), 'TStDecimal') do begin
1413: 1232 ***** File 169: F:\maxbox3\mX397\maxbox3\source\upsi_StDict.pas
1414: 1233 ('TStContainer'), 'TStDictionary') do
1415: 1234 ('TStNode'), 'TStDictNode') do
1416: 1234 ***** File 170: F:\maxbox3\mX397\maxbox3\source\upsi_StExpLog.pas
1417: 1235 ('TStComponent'), 'TStExceptionLog') do begin
1418: 1235 ***** File 171: F:\maxbox3\mX397\maxbox3\source\upsi_StExport.pas
1419: 1236 ('TObject'), 'TStDbSchemaGenerator') do begin
1420: 1237 ('TObject'), 'TStDBtoCSVExport') do begin
1421: 1237 ***** File 172: F:\maxbox3\mX397\maxbox3\source\upsi_StGenLog.pas
1422: 1238 ('TStComponent'), 'TStGeneralLog') do begin
1423: 1238 ***** File 173: F:\maxbox3\mX397\maxbox3\source\upsi_StIniStm.pas
1424: 1239 ('TObject'), 'TStIniStream') do begin

```

```

1425: 1239 ***** File 174: F:\maxbox3\mX397\maxbox3\source\upsi_StNetCon.pas
1426: 1240 ('TStComponent'), 'TStNetConnection') do begin
1427: 1240 ***** File 175: F:\maxbox3\mX397\maxbox3\source\upsi_StNetMessage.pas
1428: 1241 ('TStComponent'), 'TStNetMessage') do begin
1429: 1241 ***** File 176: F:\maxbox3\mX397\maxbox3\source\upsi_StOStr.pas
1430: 1242 ('TPersistent'), 'TStString') do begin
1431: 1242 ***** File 177: F:\maxbox3\mX397\maxbox3\source\upsi_StPtrns.pas
1432: 1243 ('TOBJECT'), 'TStChain') do
1433: 1244 ('TOBJECT'), 'TStObserver') do
1434: 1245 ('TOBJECT'), 'TStMediator') do
1435: 1246 ('TObject'), 'TStSingleton') do begin
1436: 1246 ***** File 178: F:\maxbox3\mX397\maxbox3\source\upsi_StRandom.pas
1437: 1247 ('TStRandomBase'), 'TStRandomMother') do
1438: 1248 ('TStRandomBase'), 'TStRandomCombined') do
1439: 1249 ('TStRandomBase'), 'TStRandomSystem') do
1440: 1250 ('TOBJECT'), 'TStRandomBase') do
1441: 1250 ***** File 179: F:\maxbox3\mX397\maxbox3\source\upsi_StRegEx.pas
1442: 1251 ('TStComponent'), 'TStRegEx') do begin
1443: 1252 ('TObject'), 'TStStreamRegEx') do begin
1444: 1253 ('TOBJECT'), 'TStNodeHeap') do begin
1445: 1253 ***** File 180: F:\maxbox3\mX397\maxbox3\source\upsi_StrUtils.pas
1446: 1254 ('TOBJECT'), 'EAbort');
1447: 1254 ***** File 181: F:\maxbox3\mX397\maxbox3\source\upsi_StStrms.pas
1448: 1255 ('TStream'), 'TStMemoryMappedFile') do begin
1449: 1256 ('TStBufferedStream'), 'TStAnsiTextStream') do begin
1450: 1257 ('TStream'), 'TStBufferedStream') do begin
1451: 1257 ***** File 182: F:\maxbox3\mX397\maxbox3\source\upsi_StToHTML.pas
1452: 1258 ('TStComponent'), 'TStFileToHTML') do begin
1453: 1259 ('TObject'), 'TStStreamToHTML') do begin
1454: 1259 ***** File 183: F:\maxbox3\mX397\maxbox3\source\upsi_StTxtDat.pas
1455: 1260 ('TOBJECT'), 'TStTextDataRecordSet') do begin
1456: 1261 ('TOBJECT'), 'TStTextDataRecord') do begin
1457: 1262 ('TOBJECT'), 'TStTextDataSchema') do begin
1458: 1263 ('TOBJECT'), 'TStDataFieldList') do
1459: 1264 ('TOBJECT'), 'TStDataField') do begin
1460: 1264 ***** File 184: F:\maxbox3\mX397\maxbox3\source\upsi_SyncObjs.pas
1461: 1265 ('TSynchroObject'), 'TCriticalSection') do begin
1462: 1266 ('THandleObject'), 'TMutex') do begin
1463: 1267 ('THandleObject'), 'TEvent') do begin
1464: 1268 ('TSynchroObject'), 'THandleObject') do begin
1465: 1269 ('TObject'), 'TSynchroObject') do begin
1466: 1270 ('TOBJECT'), 'TSimpleEvent');
1467: 1270 ***** File 185: F:\maxbox3\mX397\maxbox3\source\upsi_SynHighlighterMulti.pas
1468: 1271 ('TSynCustomHighlighter'), 'TSynMultiSyn') do
1469: 1272 ('TOBJECT'), 'TMarker') do
1470: 1273 ('TCollection'), 'TSchemes') do
1471: 1274 ('TCollectionItem'), 'TScheme') do
1472: 1275 ('TOBJECT'), 'TSynMultiSyn');
1473: 1275 ***** File 186: F:\maxbox3\mX397\maxbox3\source\upsi_SynMemo.pas
1474: 1276 ('TSynEdit'), 'TSynMemo') do begin
1475: 1276 ***** File 187: F:\maxbox3\mX397\maxbox3\source\upsi_SynRegExpr.pas
1476: 1277 ('Exception'), 'ERegExpr') do begin
1477: 1278 ('TOBJECT'), 'TRegExpr') do begin
1478: 1279 ('TOBJECT'), 'TRegExpr');
1479: 1279 ***** File 188: F:\maxbox3\mX397\maxbox3\source\upsi_SynURIOpener.pas
1480: 1280 ('TComponent'), 'TSynURIOpener') do begin
1481: 1280 ***** File 189: F:\maxbox3\mX397\maxbox3\source\upsi_Tabs.pas
1482: 1281 ('TCustomControl'), 'TTabSet') do
1483: 1282 ('TStringList'), 'TTabList') do
1484: 1283 ('TCustomControl'), 'TScroller') do
1485: 1284 ('TOBJECT'), 'TTabSet');
1486: 1284 ***** File 190: F:\maxbox3\mX397\maxbox3\source\upsi_ThSort.pas
1487: 1285 ('TForm'), 'TThreadSortForm') do begin
1488: 1285 ***** File 191: F:\maxbox3\mX397\maxbox3\source\upsi_ToolWin.pas
1489: 1286 ('TCustomDockForm'), 'TToolDockForm') do
1490: 1287 ('TDragDockObject'), 'TToolDockObject') do
1491: 1288 ('TWinControl'), 'TToolWindow') do
1492: 1288 ***** File 192: F:\maxbox3\mX397\maxbox3\source\upsi_TypeTrans.pas
1493: 1289 ('TOBJECT'), 'TTypeTranslator') do begin
1494: 1290 ('TOBJECT'), 'ETypeTransException');
1495: 1290 ***** File 193: F:\maxbox3\mX397\maxbox3\source\upsi_uTPLb_AES.pas
1496: 1291 ('TInterfacedObject'), 'TAES') do
1497: 1291 ***** File 194: F:\maxbox3\mX397\maxbox3\source\upsi_uTPLb_HugeCardinal.pas
1498: 1292 ('TOBJECT'), 'THugeCardinal') do
1499: 1292 ***** File 195: F:\maxbox3\mX397\maxbox3\source\upsi_uTPLb_SHA2.pas
1500: 1293 ('TInterfacedObject'), 'TSHA2') do
1501: 1293 ***** File 196: F:\maxbox3\mX397\maxbox3\source\upsi_uTPLb_StreamUtils.pas
1502: 1294 ('TStream'), 'TDesalinationWriteStream') do
1503: 1294 ***** File 197: F:\maxbox3\mX397\maxbox3\source\upsi_ValEdit.pas
1504: 1295 ('TPersistent'), 'TItemProp') do
1505: 1296 ('TStringList'), 'TValueListStrings') do
1506: 1297 ('TCustomDrawGrid'), 'TValueListEditor') do
1507: 1298 ('TOBJECT'), 'TItemProp');
1508: 1299 ('TOBJECT'), 'TValueListStrings');
1509: 1299 ***** File 198: F:\maxbox3\mX397\maxbox3\source\upsi_Variants.pas
1510: 1300 ('TCustomVariantType'), 'TInvokeableVariantType') do begin
1511: 1301 ('TObject'), 'TCustomVariantType') do begin
1512: 1302 ('TOBJECT'), 'EVariantInvalidOpError');
1513: 1303 ('TOBJECT'), 'EVariantTypeCastError');

```

```

1514: 1304 ('TOBJECT'), 'EVariantOverflowError';
1515: 1305 ('TOBJECT'), 'EVariantInvalidArgError';
1516: 1306 ('TOBJECT'), 'EVariantBadVarTypeError';
1517: 1307 ('TOBJECT'), 'EVariantBadIndexError';
1518: 1308 ('TOBJECT'), 'EVariantArrayLockedError';
1519: 1309 ('TOBJECT'), 'EVariantNotAnArrayError';
1520: 1310 ('TOBJECT'), 'EVariantArrayCreateError';
1521: 1311 ('TOBJECT'), 'EVariantNotImplError';
1522: 1312 ('TOBJECT'), 'EVariantOutOfMemoryError';
1523: 1313 ('TOBJECT'), 'EVariantUnexpectedError';
1524: 1314 ('TOBJECT'), 'EVariantDispatchError';
1525: 1315 ('TOBJECT'), 'EVariantRangeCheckError';
1526: 1316 ('TOBJECT'), 'EVariantInvalidNullOpError';
1527: 1316 ***** File 199: F:\maxbox3\mX397\maxbox3\source\upsi_VCLScannerImpl.pas
1528: 1317 ('TInvokableClass'), 'TVCLOrchestrator') do begin
1529: 1318 ('TInvokableClass'), 'TVCLScanner') do begin
1530: 1318 ***** File 200: F:\maxbox3\mX397\maxbox3\source\upsi_VListView.pas
1531: 1319 ('TForm'), 'TFormListView') do
1532: 1319 ***** File 201: F:\maxbox3\mX397\maxbox3\source\upsi_WideStrings.pas
1533: 1320 ('TWideStrings'), 'TWideStringList') do begin
1534: 1321 ('TPersistent'), 'TWideStrings') do begin
1535: 1322 ('TOBJECT'), 'TWideStringsEnumerator') do begin
1536: 1323 ('TOBJECT'), 'TWideStrings';
1537: 1324 ('TOBJECT'), 'TWideStringList';
1538: 1324 ***** File 202: F:\maxbox3\mX397\maxbox3\source\upsi_xmlutil.pas
1539: 1325 ('TOBJECT'), 'DomException';
1540: 1325 ***** File 203: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_cDateTime.pas
1541: 1326 ('TOBJECT'), 'EDateTime';
1542: 1326 ***** File 204: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_cFileUtils.pas
1543: 1327 ('Exception'), 'EFileError') do begin
1544: 1327 ***** File 205: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_CmAdmCtl.pas
1545: 1328 ('TOleServer'), 'TCOMAdminCatalogCollection') do begin
1546: 1329 ('TOBJECT'), 'CoCOMAdminCatalogCollection') do begin
1547: 1330 ('TOleServer'), 'TCOMAdminCatalogObject') do begin
1548: 1331 ('TOBJECT'), 'CoCOMAdminCatalogObject') do begin
1549: 1332 ('TOleServer'), 'TCOMAdminCatalog') do begin
1550: 1333 ('TOBJECT'), 'CoCOMAdminCatalog') do begin
1551: 1334 ('TOBJECT'), 'TCOMAdminCatalogCollection';
1552: 1334 ***** File 206: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_cPEM.pas
1553: 1335 ('TOBJECT'), 'TPEMFile') do begin
1554: 1336 ('TOBJECT'), 'EPEMFile';
1555: 1336 ***** File 207: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_cTimers.pas
1556: 1337 ('TOBJECT'), 'ETimers';
1557: 1337 ***** File 208: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_ExcelExport.pas
1558: 1338 ('TOBJECT'), 'TJsExcelExport') do begin
1559: 1338 ***** File 209: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_ExtCtrls2.pas
1560: 1339 ('TCustomTrayIcon'), 'TTrayIcon') do begin
1561: 1340 ('TCustomColorListBox'), 'TColorListBox') do begin
1562: 1341 ('TCustomListBox'), 'TCustomColorListBox') do
1563: 1342 ('TCustomColorBox'), 'TColorBox') do begin
1564: 1343 ('TCustomComboBox'), 'TCustomColorBox') do begin
1565: 1344 ('TCustomLabeledEdit'), 'TLabeledEdit') do begin
1566: 1345 ('TCustomEdit'), 'TCustomLabeledEdit') do
1567: 1346 ('TCustomLabel'), 'TBoundLabel') do
1568: 1347 ('TCustomControlBar'), 'TControlBar') do begin
1569: 1348 ('TCustomControl'), 'TCustomControlBar') do
1570: 1349 ('TGraphicControl'), 'TSplitter') do begin
1571: 1350 ('TCustomRadioGroup'), 'TRadioGroup') do
1572: 1351 ('TCustomGroupBox'), 'TCustomRadioGroup') do begin
1573: 1352 ('TCustomControl'), 'THeader') do begin
1574: 1353 ('TCustomControl'), 'TNotebook') do
1575: 1354 ('TCustomControl'), 'TPage') do
1576: 1355 ('TCustomGridPanel'), 'TGridPanel') do
1577: 1356 ('TCustomPanel'), 'TCustomGridPanel') do
1578: 1357 ('TOwnedCollection'), 'TControlCollection') do
1579: 1358 ('TCollectionItem'), 'TControlItem') do
1580: 1359 ('TCellCollection'), 'TColumnCollection') do
1581: 1360 ('TCellCollection'), 'TRowCollection') do
1582: 1361 ('TOwnedCollection'), 'TCellCollection') do
1583: 1362 ('TCollectionItem'), 'TCellItem') do
1584: 1363 ('TCustomFlowPanel'), 'TFlowPanel') do
1585: 1364 ('TCustomPanel'), 'TCustomFlowPanel') do
1586: 1365 ('TCustomPanel'), 'TPanel') do
1587: 1366 ('TCustomControl'), 'TCustomPanel') do
1588: 1367 ('TComponent'), 'TTimer') do
1589: 1368 ('TGraphicControl'), 'TBevel') do
1590: 1369 ('TGraphicControl'), 'TImage') do
1591: 1370 ('TGraphicControl'), 'TPaintBox') do
1592: 1371 ('TGraphicControl'), 'TShape') do
1593: 1372 ('TOBJECT'), 'TCustomGridPanel';
1594: 1373 ('TOBJECT'), 'EGridPanelException';
1595: 1374 ('TOBJECT'), 'TRowItem';
1596: 1375 ('TOBJECT'), 'TColumnItem';
1597: 1376 ('TOBJECT'), 'TCustomColorBox';
1598: 1377 ('TOBJECT'), 'TCustomColorListBox';
1599: 1377 ***** File 210: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_GR32.pas
1600: 1378 ('TCustomSampler'), 'TCustomResampler') do begin
1601: 1379 ('TNotifiablePersistent'), 'TCustomSampler') do begin
1602: 1380 ('TThreadPersistent'), 'TCustomBackend') do begin

```



```

1603: 1381 ('TCustomBitmap32','TBitmap32') do begin
1604: 1382 ('TCustomMap','TCustomBitmap32') do begin
1605: 1383 ('TThreadPersistent','TCustomMap') do begin
1606: 1384 ('TOBJECT','TCustomResampler');
1607: 1385 ('TOBJECT','TCustomBackend');
1608: 1385 ***** File 211: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_GR32_ExtImage.pas
1609: 1386 ('TThread','TRenderThread') do begin
1610: 1387 ('TPaintBox32','TSyntheticImage32') do begin
1611: 1388 ('TOBJECT','TRenderThread');
1612: 1388 ***** File 212: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_GR32_Image.pas
1613: 1389 ('TComponent','TBitmap32List') do begin
1614: 1390 ('TCollection','TBitmap32Collection') do
1615: 1391 ('TCollectionItem','TBitmap32Item') do begin
1616: 1392 ('TCustomImgView32','TImgView32') do
1617: 1393 ('TCustomImage32','TCustomImgView32') do begin
1618: 1394 ('TArrowBarAccess','TIVScrollProperties') do begin
1619: 1395 ('TCustomImage32','TImage32') do begin
1620: 1396 ('TCustomPaintBox32','TCustomImage32') do begin
1621: 1397 ('TCustomPaintBox32','TPaintBox32') do begin
1622: 1398 ('TCustomControl','TCustomPaintBox32') do begin
1623: 1399 ('TOBJECT','TPaintStages') do begin
1624: 1400 ('TOBJECT','TCustomImgView32');
1625: 1400 ***** File 213: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_GR32_OrdinalMaps.pas
1626: 1401 ('TCustomMap','TFloatMap') do begin
1627: 1402 ('TCustomMap','TIntegerMap') do begin
1628: 1403 ('TCustomMap','TWordMap') do begin
1629: 1404 ('TCustomMap','TByteMap') do begin
1630: 1405 ('TCustomMap','TBooleanMap') do begin
1631: 1405 ***** File 214: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_GR32_Rasterizers.pas
1632: 1406 ('TRasterizer','TMultithreadedRegularRasterizer') do
1633: 1407 ('TRasterizer','TContourRasterizer') do
1634: 1408 ('TRasterizer','TesseractRasterizer') do
1635: 1409 ('TRasterizer','TProgressiveRasterizer') do begin
1636: 1410 ('TRasterizer','TSwizzlingRasterizer') do begin
1637: 1411 ('TRasterizer','TRegularRasterizer') do begin
1638: 1412 ('TThreadPersistent','TRasterizer') do begin
1639: 1412 ***** File 215: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_GR32_System.pas
1640: 1413 ('TOBJECT','TPerfTimer') do begin
1641: 1413 ***** File 216: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_IdIdent.pas
1642: 1414 ('TIdTCPClient','TIdIdent') do begin
1643: 1415 ('TOBJECT','EIdIdentException');
1644: 1416 ('TOBJECT','EIdIdentReply');
1645: 1417 ('TOBJECT','EIdIdentInvalidPort');
1646: 1418 ('TOBJECT','EIdIdentNoUser');
1647: 1419 ('TOBJECT','EIdIdentHiddenUser');
1648: 1420 ('TOBJECT','EIdIdentUnknownError');
1649: 1421 ('TOBJECT','EIdIdentQueryTimeout');
1650: 1421 ***** File 217: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_IdIdentServer.pas
1651: 1422 ('TIdTCPServer','TIdIdentServer') do begin
1652: 1422 ***** File 218: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_IdIPWatch.pas
1653: 1423 ('TIdComponent','TIdIPWatch') do begin
1654: 1424 ('TIdThread','TIdIPWatchThread') do
1655: 1424 ***** File 219: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_IdIRCServer.pas
1656: 1425 ('TIdTCPServer','TIdIRCServer') do begin
1657: 1425 ***** File 220: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_IdMessageCoder.pas
1658: 1426 ('TObject','TIdMessageEncoderList') do
1659: 1427 ('TObject','TIdMessageEncoderInfo') do
1660: 1428 ('TIdComponent','TIdMessageEncoder') do
1661: 1429 ('TObject','TIdMessageDecoderList') do
1662: 1430 ('TObject','TIdMessageDecoderInfo') do
1663: 1431 ('TIdComponent','TIdMessageDecoder') do
1664: 1431 ***** File 221: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_IdMessageCoderMIME.pas
1665: 1432 ('TIdMessageEncoderInfo','TIdMessageEncoderInfoMIME') do
1666: 1433 ('TIdMessageEncoder','TIdMessageEncoderMIME') do begin
1667: 1434 ('TIdMessageDecoderInfo','TIdMessageDecoderInfoMIME') do
1668: 1435 ('TIdMessageDecoder','TIdMessageDecoderMIME') do begin
1669: 1435 ***** File 222: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_IdMessageCollection.pas
1670: 1436 ('TCollection','TIdMessageCollection') do begin
1671: 1437 ('TCollectionItem','TIdMessageItem') do begin
1672: 1437 ***** File 223: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_IdRemoteCMDClient.pas
1673: 1438 ('TIdTCPClient','TIdRemoteCMDClient') do begin
1674: 1439 ('TOBJECT','EIdCanNotBindRang');
1675: 1439 ***** File 224: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_IdRemoteCMDServer.pas
1676: 1440 ('TIdTCPServer','TIdRemoteCMDServer') do begin
1677: 1440 ***** File 225: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_IdRexec.pas
1678: 1441 ('TIdRemoteCMDClient','TIdRexec') do begin
1679: 1442 ('TIdRemoteCMDServer','TIdRexecServer') do begin
1680: 1442 ***** File 226: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_IdRFCReply.pas
1681: 1443 ('TOwnedCollection','TIdRFCReplies') do begin
1682: 1444 ('TCollectionItem','TIdRFCReply') do begin
1683: 1444 ***** File 227: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_IdServerIOHandler.pas
1684: 1445 ('TIdComponent','TIdServerIOHandler') do
1685: 1445 ***** File 228:
F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_IdServerIOHandlerSocket.pas
1686: 1446 ('TIdServerIOHandler','TIdServerIOHandlerSocket') do
1687: 1446 ***** File 229: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_IdUDPServer.pas
1688: 1447 ('TIdUDPClient','TIdTimeUDP') do begin
1689: 1448 ('TIdTCPServer','TIdTimeServer') do begin
1690: 1449 ('TIdUDPServer','TIdTimeUDPServer') do begin

```

```

1691: 1450 ('TidUDPBase'), 'TidUDPServer') do begin
1692: 1451 ('TidThread'), 'TidUDPListenerThread') do begin
1693: 1452 ('TObject'), 'TidUDPServer');
1694: 1453 ('TObject'), 'TidUDPServerException');
1695: 1453 ***** File 230: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JclAppInst.pas
1696: 1454 ('TObject'), 'TJclAppInstances') do begin
1697: 1454 ***** File 231: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JclComplex.pas
1698: 1455 ('TObject'), 'TJclComplex') do begin
1699: 1455 ***** File 232: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JclLocales.pas
1700: 1456 ('TObject'), 'TJclKeyboardLayoutList') do begin
1701: 1457 ('TObject'), 'TJclKeyboardLayout') do begin
1702: 1458 ('TObject'), 'TJclAvailableKeybLayout') do begin
1703: 1459 ('TObjectList'), 'TJclLocalesList') do begin
1704: 1460 ('TObject'), 'TJclLocaleInfo') do begin
1705: 1461 ('TObject'), 'TJclKeyboardLayoutList');
1706: 1461 ***** File 233: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JclMIDI.pas
1707: 1462 ('TInterfacedObject'), 'TJclMIDIOut') do begin
1708: 1463 ('TObject'), 'EJclMIDIErrors');
1709: 1463 ***** File 234: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JclNTFS.pas
1710: 1464 ('TObject'), 'EJclNtfsErrors');
1711: 1464 ***** File 235: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JclPeImage.pas
1712: 1465 ('TObjectList'), 'TJclPeMapImgHooks') do begin
1713: 1466 ('TObject'), 'TJclPeMapImgHookItem') do begin
1714: 1467 ('TCustomMemoryStream'), 'TJclPeSectionStream') do
1715: 1468 ('TThread'), 'TJclPeNameSearch') do begin
1716: 1469 ('TJclPeImage'), 'TJclPeBorImage') do begin
1717: 1470 ('TObject'), 'TJclPeBorForm') do begin
1718: 1471 ('TObject'), 'TJclPePackageInfo') do begin
1719: 1472 ('TObject'), 'TJclPeImage') do begin
1720: 1473 ('TObject'), 'TJclPeCLRHeader') do begin
1721: 1474 ('TJclPeImageBaseList'), 'TJclPeCertificateList') do
1722: 1475 ('TObject'), 'TJclPeCertificate') do begin
1723: 1476 ('TJclPeImageBaseList'), 'TJclPeDebugList') do begin
1724: 1477 ('TJclPeImageBaseList'), 'TJclPeRelocList') do begin
1725: 1478 ('TObject'), 'TJclPeRelocEntry') do begin
1726: 1479 ('TJclPeResourceList'), 'TJclPeRootResourceList') do
1727: 1480 ('TJclPeImageBaseList'), 'TJclPeResourceList') do begin
1728: 1481 ('TObject'), 'TJclPeResourceItem') do begin
1729: 1482 ('TCustomMemoryStream'), 'TJclPeResourceRawStream') do
1730: 1483 ('TJclPeImageBaseList'), 'TJclPeExportFuncList') do begin
1731: 1484 ('TObject'), 'TJclPeExportFuncItem') do begin
1732: 1485 ('TJclPeImageBaseList'), 'TJclPeImportList') do begin
1733: 1486 ('TJclPeImageBaseList'), 'TJclPeImportLibItem') do begin
1734: 1487 ('TObject'), 'TJclPeImportFuncItem') do begin
1735: 1488 ('TJclPeImagesCache'), 'TJclPeBorImagesCache') do begin
1736: 1489 ('TObject'), 'TJclPeImagesCache') do begin
1737: 1490 ('TObjectList'), 'TJclPeImageBaseList') do begin
1738: 1491 ('TObject'), 'EJclPeImageError');
1739: 1492 ('TObject'), 'TJclPeImage');
1740: 1493 ('TObject'), 'TJclPeBorImage');
1741: 1494 ('TObject'), 'TJclPeImportLibItem');
1742: 1495 ('TObject'), 'TJclPeExportFuncList');
1743: 1496 ('TObject'), 'TJclPeResourceList');
1744: 1497 ('TObject'), 'TJclPeResourceItem');
1745: 1497 ***** File 236: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JclPrint.pas
1746: 1498 ('TObject'), 'TJclPrintSet') do begin
1747: 1499 ('TObject'), 'EJclPrinterError');
1748: 1499 ***** File 237: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JclRegistry.pas
1749: 1500 ('TObject'), 'EJclRegistryError');
1750: 1500 ***** File 238: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JclSchedule.pas
1751: 1501 ('TObject'), 'ESchedule');
1752: 1501 ***** File 239: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JclSvcCtrl.pas
1753: 1502 ('TObject'), 'TJclSCManager') do begin
1754: 1503 ('TObject'), 'TJclServiceGroup') do begin
1755: 1504 ('TObject'), 'TJclNtService') do begin
1756: 1505 ('TObject'), 'TJclServiceGroup');
1757: 1506 ('TObject'), 'TJclSCManager');
1758: 1506 ***** File 240: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvAlarms.pas
1759: 1507 ('TJvComponent'), 'TJvAlarms') do begin
1760: 1508 ('TOwnedCollection'), 'TJvAlarmItems') do begin
1761: 1509 ('TCollectionItem'), 'TJvAlarmItem') do begin
1762: 1509 ***** File 241: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvAnalogClock.pas
1763: 1510 ('TJvCustomPanel'), 'TJvAnalogClock') do begin
1764: 1510 ***** File 242: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvAnimatedImage.pas
1765: 1511 ('TBitmap'), 'TJvLockedBitmap') do
1766: 1512 ('TJvImageControl'), 'TJvAnimatedImage') do begin
1767: 1513 ('TJvGraphicControl'), 'TJvImageControl') do begin
1768: 1513 ***** File 243: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvAnimTitle.pas
1769: 1514 ('TJvComponent'), 'TJvAnimTitle') do begin
1770: 1514 ***** File 244: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvAppCommand.pas
1771: 1515 ('TJvComponent'), 'TJvAppCommand') do begin
1772: 1515 ***** File 245: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvAppEvent.pas
1773: 1516 ('TJvComponent'), 'TJvAppEvents') do begin
1774: 1516 ***** File 246: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvAppInst.pas
1775: 1517 ('TComponent'), 'TJvAppInstances') do begin
1776: 1517 ***** File 247: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvBDESQLScript.pas
1777: 1518 ('TJvComponent'), 'TJvBDESQLScript') do
1778: 1519 ('TObject'), 'TJvBDESQLScript');
1779: 1519 ***** File 248: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvBrowseFolder.pas

```

```

1780: 1520 ('TJvCommonDialogF'), 'TJvBrowseForFolderDialog') do
1781: 1520 ***** File 249: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvChart.pas
1782: 1521 ('TJvGraphicControl'), 'TJvChart') do begin
1783: 1522 ('TPersistent'), 'TJvChartOptions') do begin
1784: 1523 ('TPersistent'), 'TJvChartYAxisOptions') do begin
1785: 1524 ('TObject'), 'TJvChartData') do begin
1786: 1525 ('TJvChartGradientBar'), 'TJvChartVerticalBar') do begin
1787: 1526 ('TJvChartGradientBar'), 'TJvChartHorizontalBar') do begin
1788: 1527 ('TObject'), 'TJvChartGradientBar') do begin
1789: 1528 ('TPersistent'), 'TJvChartFloatingMarker') do begin
1790: 1529 ('TObject'), 'TJvChart');
1791: 1530 ('TObject'), 'TJvChartOptions');
1792: 1530 ***** File 250: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JVCLMiscal.pas
1793: 1531 ('TStringProperty'), 'TJvDirectoryProperty') do
1794: 1532 ('TStringProperty'), 'TJvExeNameProperty') do
1795: 1533 ('TStringProperty'), 'TJvPerfStatProperty') do
1796: 1534 ('TComponentEditor'), 'TJvMailEditor') do
1797: 1535 ('TComponentEditor'), 'TJvOpenDialogEditor') do
1798: 1536 ('TEnumProperty'), 'TJvNosortEnumProperty') do
1799: 1537 ('TObject'), 'TCommonDialogHack');
1800: 1537 ***** File 251: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvConverter.pas
1801: 1538 ('TJvComponent'), 'TJvConverter2') do begin
1802: 1539 ('TPersistent'), 'TJvDateTimeFormat2') do begin
1803: 1539 ***** File 252: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvCpuUsage.pas
1804: 1540 ('TJvComponent'), 'TJvCpuUsage') do begin
1805: 1540 ***** File 253: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvCpuUsage2.pas
1806: 1541 ('TJvComponent'), 'TJvCpuUsage2') do begin
1807: 1541 ***** File 254: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvDataConv.pas
1808: 1542 ('TComponent'), 'TJvConverter') do begin
1809: 1543 ('TPersistent'), 'TJvDateTimeFormat') do begin
1810: 1543 ***** File 255: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvDBGridExport.pas
1811: 1544 ('TJvCustomDBGridTextExport'), 'TJvDBGridXMLeExport') do
1812: 1545 ('TJvCustomDBGridTextExport'), 'TJvDBGridCSVEExport') do
1813: 1546 ('TJvCustomDBGridTextExport'), 'TJvDBGridHTMLEExport') do
1814: 1547 ('TJvCustomDBGridExport'), 'TJvCustomDBGridTextExport') do
1815: 1548 ('TJvCustomDBGridOleExport'), 'TJvDBGridExcelExport') do
1816: 1549 ('TJvCustomDBGridOleExport'), 'TJvDBGridWordExport') do
1817: 1550 ('TJvCustomDBGridExport'), 'TJvCustomDBGridOleExport') do
1818: 1551 ('TJvComponent'), 'TJvCustomDBGridExport') do begin
1819: 1552 ('TObject'), 'EJvExportDBGridException');
1820: 1552 ***** File 256: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvDBQBE.pas
1821: 1553 ('TDBDataSet'), 'TJvQBEQuery') do begin
1822: 1553 ***** File 257: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvDBRichEd.pas
1823: 1554 ('TJvCustomRichEdit'), 'TJvDBRichEdit') do begin
1824: 1554 ***** File 258: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvDBSecur.pas
1825: 1555 ('TJvCustomLogin'), 'TJvDBSecurity') do begin
1826: 1555 ***** File 259: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvDBUltimGrid.pas
1827: 1556 ('TJvDBGrid'), 'TJvDBUltimGrid') do
1828: 1557 ('TObject'), 'TJvDBUltimGrid');
1829: 1557 ***** File 260: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvDice.pas
1830: 1558 ('TCustomControl'), 'TJvDice') do begin
1831: 1558 ***** File 261: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvDirectories.pas
1832: 1559 ('TJvComponent'), 'TJvDirectories') do begin
1833: 1559 ***** File 262: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvDirFrm.pas
1834: 1560 ('TJvForm'), 'TJvDirectoryListDialog') do begin
1835: 1560 ***** File 263: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvDualList.pas
1836: 1561 ('TJvComponent'), 'TJvDualListDialog') do begin
1837: 1561 ***** File 264: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvDualListForm.pas
1838: 1562 ('TJvForm'), 'TJvDualListForm') do begin
1839: 1562 ***** File 265: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvDualList_old.pas
1840: 1563 ('TJvComponent'), 'TJvDualListDialog') do begin
1841: 1563 ***** File 266: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvFloatEdit.pas
1842: 1564 ('TJvEdit'), 'TJvFloatEdit') do begin
1843: 1564 ***** File 267: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvFormAutoSize.pas
1844: 1565 ('TJvComponent'), 'TJvFormAutoSize') do begin
1845: 1565 ***** File 268: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvFormPatch.pas
1846: 1566 ('TForm'), 'TFoPatch') do begin
1847: 1566 ***** File 269: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvForth.pas
1848: 1567 ('TJvComponent'), 'TJvForthScript') do begin
1849: 1568 ('TList'), 'TAtomList') do begin
1850: 1569 ('TObject'), 'TAtomF') do begin
1851: 1570 ('TStringList'), 'TVariantList') do begin
1852: 1571 ('TObject'), 'TVariantObject') do begin
1853: 1572 ('TStringList'), 'TJvJanXMLList') do begin
1854: 1573 ('TStringList'), 'TJvJanDSOList') do begin
1855: 1574 ('TStringList'), 'TJvJanDSO') do begin
1856: 1575 ('TObject'), 'EJvJanScriptError');
1857: 1575 ***** File 270: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvgDigits.pas
1858: 1576 ('TJvGraphicControl'), 'TJvgDigits') do
1859: 1576 ***** File 271: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvImageDrawThread.pas
1860: 1577 ('TThread'), 'TJvImageDrawThread') do
1861: 1577 ***** File 272: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvImageWindow.pas
1862: 1578 ('TJvGraphicControl'), 'TJvImageSquare') do begin
1863: 1579 ('TJvGraphicControl'), 'TJvImageWindow') do begin
1864: 1579 ***** File 273: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvImagPrvw.pas
1865: 1580 ('TForm'), 'TImageForm') do begin
1866: 1580 ***** File 274: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvJanTreeView.pas
1867: 1581 ('TTreeView'), 'TJvJanTreeView') do begin
1868: 1582 ('TPersistent'), 'TTreeKeyMappings') do begin

```

```

1869: 1583 ('TComponent'), 'TJvMathParser2') do begin
1870: 1583 ***** File 275: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvJoystick.pas
1871: 1584 ('TJvComponent'), 'TJvJoystick') do begin
1872: 1585 ('TPersistent'), 'TJoystick') do begin
1873: 1585 ***** File 276: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvMailSlots.pas
1874: 1586 ('TJvComponent'), 'TJvMailSlotClient') do begin
1875: 1587 ('TJvComponent'), 'TJvMailSlotServer') do begin
1876: 1587 ***** File 277: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvMarkupCommon.pas
1877: 1588 ('TList'), 'TJvHTMLElementStack') do begin
1878: 1589 ('TObject'), 'TJvHTMLElement') do begin
1879: 1589 ***** File 278: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvMemTable.pas
1880: 1590 ('TDBDataSet'), 'TJvMemoryTable') do begin
1881: 1590 ***** File 279: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvNTEventLog.pas
1882: 1591 ('TObject'), 'TJvNTEventLogRecord') do begin
1883: 1592 ('TThread'), 'TNotifyChangeEventLog') do
1884: 1593 ('TJvComponent'), 'TJvNTEventLog') do begin
1885: 1594 ('TObject'), 'TNotifyChangeEventLog');
1886: 1595 ('TObject'), 'TJvNTEventLogRecord');
1887: 1595 ***** File 280: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvObjStr.pas
1888: 1596 ('TCollection'), 'TJvSortCollection') do
1889: 1597 ('TObject'), 'TJvHugeList') do begin
1890: 1598 ('TStringList'), 'TJvObjectStrings') do begin
1891: 1599 ('TObject'), 'TJvHugeList');
1892: 1599 ***** File 281: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvOracleDataSet.pas
1893: 1600 ('TOracleDataset'), 'TJvOracleDataset') do
1894: 1601 ('TJvBaseDatasetThreadHandler'), 'TJvOracleDatasetThreadHandler') do
1895: 1602 ('TJvBaseThreadedDatasetEnhancedOptions'), 'TJvDoaThreadedDatasetEnhancedOptions') do
1896: 1603 ('TJvBaseThreadedDatasetAllowedContinueRecordFetchOptions'),
    'TJvDoaThreadedDatasetAllowedContinueRecordFetchOptions') do
1897: 1603 ***** File 282: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvPaintFX.pas
1898: 1604 ('TComponent'), 'TJvPaintFX') do begin
1899: 1604 ***** File 283: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvParserForm.pas
1900: 1605 ('TJvForm'), 'TJvHTMLParserForm') do begin
1901: 1605 ***** File 284: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvPatchFile.pas
1902: 1606 ('TJvComponent'), 'TJvPatchFile') do begin
1903: 1606 ***** File 285: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvPicClip.pas
1904: 1607 ('TComponent'), 'TJvPicClip') do begin
1905: 1607 ***** File 286: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvPlaylist.pas
1906: 1608 ('TJvListBox'), 'TJvPlaylist') do begin
1907: 1608 ***** File 287: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvProfiler32.pas
1908: 1609 ('TJvComponent'), 'TJvProfiler') do
1909: 1610 ('TForm'), 'TProfReport') do
1910: 1610 ***** File 288: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvRas32.pas
1911: 1611 ('TJvComponent'), 'TJvRas32') do
1912: 1612 ('TObject'), 'EJvRasError');
1913: 1612 ***** File 289: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvRichEdit.pas
1914: 1613 ('TRichEdit'), 'TJvRichEdit') do begin
1915: 1613 ***** File 290: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvSearchFiles.pas
1916: 1614 ('TJvComponent'), 'TJvSearchFiles') do begin
1917: 1615 ('TPersistent'), 'TJvSearchParams') do begin
1918: 1616 ('TPersistent'), 'TJvSearchAttributes') do
1919: 1616 ***** File 291: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvSerialMaker.pas
1920: 1617 ('TJvComponent'), 'TJvSerialMaker') do begin
1921: 1617 ***** File 292: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvSimIndicator.pas
1922: 1618 ('TJvGraphicControl'), 'TJvSimIndicator') do begin
1923: 1618 ***** File 293: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvSimLogic.pas
1924: 1619 ('TGraphicControl'), 'TJvSimLogicBox') do begin
1925: 1620 ('TGraphicControl'), 'TJvSimBin') do begin
1926: 1621 ('TGraphicControl'), 'TJvSimLight') do begin
1927: 1622 ('TGraphicControl'), 'TJvSimButton') do begin
1928: 1623 ('TGraphicControl'), 'TJvSimReverse') do begin
1929: 1624 ('TGraphicControl'), 'TJvLogic') do begin
1930: 1625 ('TGraphicControl'), 'TJvSIMConnector') do begin
1931: 1626 ('TPersistent'), 'TJvPointX') do begin
1932: 1627 ('TObject'), 'TJvLogic');
1933: 1627 ***** File 294: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvSimPID.pas
1934: 1628 ('TJvGraphicControl'), 'TJvSimPID') do begin
1935: 1628 ***** File 295: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvSimPIDLinker.pas
1936: 1629 ('TComponent'), 'TJvSimPIDLinker') do begin
1937: 1629 ***** File 296: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvSoundControl.pas
1938: 1630 ('TJvComponent'), 'TJvSoundControl') do
1939: 1631 ('TPersistent'), 'TJvSoundValue') do
1940: 1631 ***** File 297: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvSpeedbarSetupForm.pas
1941: 1632 ('TJvForm'), 'TJvSpeedbarSetupWindow') do begin
1942: 1632 ***** File 298: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvSQLS.pas
1943: 1633 ('TJvComponent'), 'TJvSQLScript') do begin
1944: 1634 ('TObject'), 'TJvSQLScript');
1945: 1634 ***** File 299: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvStarfield.pas
1946: 1635 ('TGraphicControl'), 'TJvStarfield') do begin
1947: 1635 ***** File 300: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvSticker.pas
1948: 1636 ('TJvGraphicControl'), 'TJvSticker') do begin
1949: 1637 ('TJvCustomControl'), 'TJvStickSizer') do begin
1950: 1637 ***** File 301: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvSwitch.pas
1951: 1638 ('TCustomControl'), 'TJvSwitch') do begin
1952: 1638 ***** File 302: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvTFUtils.pas
1953: 1639 ('TObject'), 'EJvTFDateError');
1954: 1639 ***** File 303: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvTimerLst.pas
1955: 1640 ('TComponent'), 'TJvTimerEvent') do begin
1956: 1641 ('TComponent'), 'TJvTimerList') do begin

```



```
1957: 1642 ('TOBJECT'), 'TJvTimerEvent');
1958: 1642 ***** File 304: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvTransLED.pas
1959: 1643 ('TJvGraphicControl'), 'TJvTransLED') do begin
1960: 1643 ***** File 305: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvTransparentForm.pas
1961: 1644 ('TJvComponent'), 'TJvTransparentForm') do begin
1962: 1644 ***** File 306: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvUnicodeCanvas.pas
1963: 1645 ('TCanvas'), 'TJvUnicodeCanvas') do begin
1964: 1645 ***** File 307: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvUrlGrabbers.pas
1965: 1646 ('TJvCustomUrlGrabberDefaultProperties'), 'TJvLocalFileUrlGrabberProperties') do
1966: 1647 ('TJvCustomUrlGrabberThread'), 'TJvLocalFileUrlGrabberThread') do
1967: 1648 ('TJvCustomUrlGrabber'), 'TJvLocalFileUrlGrabber') do begin
1968: 1649 ('TJvHttpUrlGrabberDefaultProperties'), 'TJvHttpsUrlGrabberDefaultProperties') do
1969: 1650 ('TJvHttpUrlGrabber'), 'TJvHttpsUrlGrabber') do
1970: 1651 ('TJvCustomUrlGrabberThread'), 'TJvHttpUrlGrabberThread') do
1971: 1652 ('TJvProxyingUrlGrabberDefaultProperties'), 'TJvHttpUrlGrabberDefaultProperties') do
1972: 1653 ('TJvProxyingUrlGrabber'), 'TJvHttpUrlGrabber') do begin
1973: 1654 ('TJvCustomUrlGrabberThread'), 'TJvFtpUrlGrabberThread') do
1974: 1655 ('TJvProxyingUrlGrabber'), 'TJvFtpUrlGrabber') do begin
1975: 1656 ('TJvProxyingUrlGrabberDefaultProperties'), 'TJvFtpUrlGrabberDefaultProperties') do
1976: 1657 ('TJvCustomUrlGrabberDefaultProperties'), 'TJvProxyingUrlGrabberDefaultProperties') do
1977: 1658 ('TJvCustomUrlGrabber'), 'TJvProxyingUrlGrabber') do begin
1978: 1658 ***** File 308: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvValidators.pas
1979: 1659 ('TJvComponent'), 'TJvValidationSummary') do
1980: 1660 ('TJvComponent'), 'TJvValidators') do
1981: 1661 ('TJvBaseValidator'), 'TJvControlsCompareValidator') do
1982: 1662 ('TJvBaseValidator'), 'TJvCustomValidator') do
1983: 1663 ('TJvBaseValidator'), 'TJvRegularExpressionValidator') do
1984: 1664 ('TJvBaseValidator'), 'TJvRangeValidator') do
1985: 1665 ('TJvBaseValidator'), 'TJvCompareValidator') do
1986: 1666 ('TJvBaseValidator'), 'TJvRequiredFieldValidator') do
1987: 1667 ('TJvComponent'), 'TJvBaseValidator') do
1988: 1668 ('TOBJECT'), 'EValidatorError');
1989: 1669 ('TOBJECT'), 'TJvBaseValidator');
1990: 1670 ('TOBJECT'), 'TJvValidators');
1991: 1670 ***** File 309: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvWavePlayer.pas
1992: 1671 ('TJvComponent'), 'TJvWavePlayer') do begin
1993: 1671 ***** File 310: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvWinDialogs.pas
1994: 1672 ('TSaveDialog'), 'TJvSaveDialog2000') do begin
1995: 1673 ('TOpenDialog'), 'TJvOpenDialog2000') do begin
1996: 1674 ('TJvCommonDialogP'), 'TJvExitWindowsDialog') do
1997: 1675 ('TJvCommonDialogF'), 'TJvDiskFullDialog') do begin
1998: 1676 ('TJvCommonDialogP'), 'TJvOpenWithDialog') do begin
1999: 1677 ('TJvCommonDialogP'), 'TJvAddHardwareDialog') do begin
2000: 1678 ('TJvCommonDialogP'), 'TJvNewLinkDialog') do begin
2001: 1679 ('TJvCommonDialogF'), 'TJvObjectPropertiesDialog') do
2002: 1680 ('TJvCommonDialogP'), 'TJvRunDialog') do begin
2003: 1681 ('TJvCommonDialog'), 'TJvShellAboutDialog') do begin
2004: 1682 ('TJvCommonDialogF'), 'TJvChangeIconDialog') do begin
2005: 1683 ('TJvCommonDialog'), 'TJvOutOfMemoryDialog') do begin
2006: 1684 ('TJvCommonDialog'), 'TJvBrowseFolderDialog') do begin
2007: 1685 ('TJvCommonDialog'), 'TJvComputerNameDialog') do begin
2008: 1686 ('TJvCommonDialogF'), 'TJvAppletDialog') do begin
2009: 1687 ('TJvCommonDialogP'), 'TJvControlPanelDialog') do
2010: 1688 ('TJvCommonDialog'), 'TJvOrganizeFavoritesDialog') do
2011: 1689 ('TJvCommonDialogF'), 'TJvFormatDriveDialog') do begin
2012: 1690 ('TOBJECT'), 'EShellOleError');
2013: 1691 ('TOBJECT'), 'EWinDialogError');
2014: 1691 ***** File 311: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvXmlTree.pas
2015: 1692 ('TJvXMLNode'), 'TJvXMLTree') do begin
2016: 1693 ('TOBJECT'), 'TJvXMLNode') do begin
2017: 1694 ('TOBJECT'), 'TJvXMLAttribute') do begin
2018: 1695 ('TOBJECT'), 'TJvXMLFilter') do begin
2019: 1696 ('TOBJECT'), 'TJvXMLFilterAtom') do begin
2020: 1697 ('TOBJECT'), 'TJvXMLTree');
2021: 1698 ('TOBJECT'), 'TJvXMLNode');
2022: 1698 ***** File 312: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvXPCore.pas
2023: 1699 ('TPersistent'), 'TJvXPGradient') do begin
2024: 1700 ('TJvXPCustomControl'), 'TJvXPCustomStyleControl') do
2025: 1701 ('TJvXPCustomComponent'), 'TJvXPStyleManager') do begin
2026: 1702 ('TPersistent'), 'TJvXPStyle') do begin
2027: 1703 ('TJvXPCustomControl'), 'TJvXPUnlimitedControl') do
2028: 1704 ('TJvCustomControl'), 'TJvXPCustomControl') do begin
2029: 1705 ('TJvWinControl'), 'TJvXPWinControl') do
2030: 1706 ('TOBJECT'), 'TJvXPCustomStyleControl');
2031: 1707 ('TOBJECT'), 'TJvXPCustomComponent');
2032: 1707 ***** File 313: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvYearGridEditForm.pas
2033: 1708 ('TJvForm'), 'TYearGridEditForm') do begin
2034: 1708 ***** File 314: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvZoom.pas
2035: 1709 ('TJvCustomControl'), 'TJvZoom') do begin
2036: 1709 ***** File 315: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_LongIntList.pas
2037: 1710 ('TOBJECT'), 'TLongIntList') do begin
2038: 1710 ***** File 316: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_OleAuto.pas
2039: 1711 ('EOleError'), 'EOleException') do begin
2040: 1712 ('EOleError'), 'EOleSysError') do begin
2041: 1713 ('TOBJECT'), 'TAutomation') do begin
2042: 1714 ('TOBJECT'), 'TRegistryClass') do begin
2043: 1715 ('TOBJECT'), 'TAutoObject') do begin
2044: 1716 ('IDispatch'), 'TAutoDispatch') do begin
2045: 1717 ('TOBJECT'), 'TAutoObject');
```



```

2046: 1718 ('TOBJECT'), 'EOleError');
2047: 1718 ***** File 317: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_Registry.pas
2048: 1719 ('TCustomIniFile'), 'TRegistryIniFile') do begin
2049: 1720 ('TRegistry'), 'TRegIniFile') do begin
2050: 1721 ('TOBJECT'), 'TRegistry') do begin
2051: 1722 ('TOBJECT'), 'ERegistryException');
2052: 1722 ***** File 318: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_ShellZipTool.pas
2053: 1723 ('TOBJECT'), 'TShellZip') do begin
2054: 1723 ***** File 319: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_StatsClasses.pas
2055: 1724 ('TOBJECT'), 'TStatArray') do begin
2056: 1724 ***** File 320: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_StBarC.pas
2057: 1725 ('TGraphicControl'), 'TStBarCode') do begin
2058: 1726 ('TOBJECT'), 'TStBarCodeInfo') do begin
2059: 1727 ('TOBJECT'), 'TStBarData') do begin
2060: 1727 ***** File 321: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_StBarPN.pas
2061: 1728 ('TGraphicControl'), 'TStPNBarCode') do
2062: 1728 ***** File 322: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_StBase.pas
2063: 1729 ('TEdit'), 'TStBaseEdit') do
2064: 1730 ('TComponent'), 'TStComponent') do begin
2065: 1731 ('TPersistent'), 'TStContainer') do begin
2066: 1732 ('TPersistent'), 'TStNode') do begin
2067: 1733 ('EStException'), 'EStExprError') do begin
2068: 1734 ('Exception'), 'EStException') do begin
2069: 1735 ('TOBJECT'), 'EStContainerError');
2070: 1736 ('TOBJECT'), 'EStSortError');
2071: 1737 ('TOBJECT'), 'EStRegIniError');
2072: 1738 ('TOBJECT'), 'EStBCDError');
2073: 1739 ('TOBJECT'), 'EStStringError');
2074: 1740 ('TOBJECT'), 'EStVersionInfoError');
2075: 1741 ('TOBJECT'), 'EStNetException');
2076: 1742 ('TOBJECT'), 'EStBarCodeError');
2077: 1743 ('TOBJECT'), 'EStPNBarCodeError');
2078: 1744 ('TOBJECT'), 'EStStatError');
2079: 1745 ('TOBJECT'), 'EStFinError');
2080: 1746 ('TOBJECT'), 'EStMimeError');
2081: 1747 ('TOBJECT'), 'EStToHTMLError');
2082: 1748 ('TOBJECT'), 'EStSpawnError');
2083: 1749 ('TOBJECT'), 'EStMMFileError');
2084: 1750 ('TOBJECT'), 'EStBufStreamError');
2085: 1751 ('TOBJECT'), 'EStRegExError');
2086: 1752 ('TOBJECT'), 'EStDecMathError');
2087: 1753 ('TOBJECT'), 'EStPRNGError');
2088: 1754 ('TOBJECT'), 'TStContainer');
2089: 1754 ***** File 323: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_StBits.pas
2090: 1755 ('TStContainer'), 'TStBits') do begin
2091: 1756 ('TOBJECT'), 'TStBits');
2092: 1756 ***** File 324: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_StDb2DBC.pas
2093: 1757 ('TStMaxiCodeBarcode'), 'TStDbMaxiCodeBarcode') do
2094: 1758 ('TStPDF417Barcode'), 'TStDbPDF417Barcode') do
2095: 1758 ***** File 325: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_StDbBarC.pas
2096: 1759 ('TStBarCode'), 'TStDbBarCode') do
2097: 1759 ***** File 326: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_StDbPNBC.pas
2098: 1760 ('TStPNBarCode'), 'TStDbPNBarCode') do
2099: 1760 ***** File 327: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_StLArr.pas
2100: 1761 ('TStContainer'), 'TStLMatrix') do begin
2101: 1762 ('TStContainer'), 'TStLArray') do begin
2102: 1762 ***** File 328: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_StMoney.pas
2103: 1763 ('TOBJECT'), 'TStMoney') do
2104: 1764 ('TOBJECT'), 'TStExchangeRateList') do
2105: 1765 ('TOBJECT'), 'TStExchangeRate') do
2106: 1766 ('TOBJECT'), 'TStCurrencyList') do
2107: 1767 ('TOBJECT'), 'TStCurrency') do
2108: 1767 ***** File 329: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_StNet.pas
2109: 1768 ('TOBJECT'), 'TStNetwork') do begin
2110: 1769 ('TStNetItem'), 'TStNetServerItem') do begin
2111: 1770 ('TStNetItem'), 'TStNetShareItem') do
2112: 1771 ('TStNetItem'), 'TStNetGroupItem') do begin
2113: 1772 ('TStNetItem'), 'TStNetUserItem') do begin
2114: 1773 ('TOBJECT'), 'TStNetItem') do begin
2115: 1774 ('TOBJECT'), 'TStNetwork');
2116: 1774 ***** File 330: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_StNetCon.pas
2117: 1775 ('TStComponent'), 'TStNetConnection') do
2118: 1775 ***** File 331: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_StNetPfm.pas
2119: 1776 ('TStComponent'), 'TStNetPerformance') do begin
2120: 1776 ***** File 332: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_StNTLog.pas
2121: 1777 ('TStComponent'), 'TStNTEventLog') do begin
2122: 1777 ***** File 333: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_StSort.pas
2123: 1778 ('TOBJECT'), 'TStSorter') do begin
2124: 1778 ***** File 334: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_StVInfo.pas
2125: 1779 ('TStCustomVersionInfo'), 'TStVersionInfo') do begin
2126: 1780 ('TStComponent'), 'TStCustomVersionInfo') do begin
2127: 1780 ***** File 335: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_StWmDCpy.pas
2128: 1781 ('TStComponent'), 'TStWmDataCopy') do begin
2129: 1781 ***** File 336: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_SynEdit.pas
2130: 1782 ('TCustomSynEdit'), 'TSynEdit') do
2131: 1783 ('TCustomControl'), 'TCustomSynEdit') do begin
2132: 1784 ('TOBJECT'), 'TSynEditPlugin') do
2133: 1785 ('TOBJECT'), 'ESynEditError');
2134: 1786 ('TOBJECT'), 'TCustomSynEdit');

```

```
2135: 1786 ***** File 337: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_SynEditExport.pas
2136: 1787 ('TComponent'), 'TSynCustomExporter') do begin
2137: 1787 ***** File 338: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_SynEditHighlighter.pas
2138: 1788 ('TList'), 'TSynHighlighterList') do begin
2139: 1789 ('TComponent'), 'TSynCustomHighlighter') do
2140: 1790 ('TPersistent'), 'TSynHighlighterAttributes') do
2141: 1790 ***** File 339: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_SynEditKbdHandler.pas
2142: 1791 ('TObject'), 'TSynEditKbdHandler') do begin
2143: 1792 ('TOBJECT'), 'TMethodList') do begin
2144: 1793 ('TWinControl'), 'TKeyboardControl') do
2145: 1793 ***** File 340: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_SynEditKeyCmds.pas
2146: 1794 ('TCollection'), 'TSynEditKeyStrokes') do begin
2147: 1795 ('TCollectionItem'), 'TSynEditKeyStroke') do begin
2148: 1796 ('TOBJECT'), 'ESynKeyError');
2149: 1796 ***** File 341: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_SynEditMiscClasses.pas
2150: 1797 ('TObject'), 'TSynEditMarkList') do begin
2151: 1798 ('TOBJECT'), 'TSynEditMark') do begin
2152: 1799 ('TRegistry'), 'TBetterRegistry') do
2153: 1800 ('TComponent'), 'TSynEditSearchCustom') do
2154: 1801 ('TCustomControl'), 'TSynHotKey') do
2155: 1802 ('TObject'), 'TSynInternalImage') do
2156: 1803 ('TSynMethodChain'), 'TSynNotifyEventChain') do
2157: 1804 ('TObject'), 'TSynMethodChain') do
2158: 1805 ('TPersistent'), 'TSynGlyph') do
2159: 1806 ('TPersistent'), 'TSynBookMarkOpt') do
2160: 1807 ('TPersistent'), 'TSynGutter') do begin
2161: 1808 ('TPersistent'), 'TSynSelectedColor') do begin
2162: 1809 ('TOBJECT'), 'ESynMethodChain');
2163: 1809 ***** File 342: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_SynEditRegexSearch.pas
2164: 1810 ('TSynEditSearchCustom'), 'TSynEditRegexSearch') do
2165: 1810 ***** File 343: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_SynEditSearch.pas
2166: 1811 ('TSynEditSearchCustom'), 'TSynEditSearch') do begin
2167: 1811 ***** File 344: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_SynExportHTML.pas
2168: 1812 ('TSynCustomExporter'), 'TSynExporterHTML') do begin
2169: 1812 ***** File 345: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_SynExportRTF.pas
2170: 1813 ('TSynCustomExporter'), 'TSynExporterRTF') do begin
2171: 1813 ***** File 346: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_SynHighlighterAny.pas
2172: 1814 ('TSynCustomHighlighter'), 'TSynAnySyn') do begin
2173: 1815 ('TStringList'), 'TIniList') do begin
2174: 1815 ***** File 347: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_SynHighlighterMulti.pas
2175: 1816 ('TSynCustomHighlighter'), 'TSynMultiSyn') do
2176: 1817 ('TOBJECT'), 'TMarker') do
2177: 1818 ('TCollection'), 'TSchemes') do
2178: 1819 ('TCollectionItem'), 'TScheme') do
2179: 1820 ('TOBJECT'), 'TSynMultiSyn');
2180: 1820 ***** File 348: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_SynHighlighterPas.pas
2181: 1821 ('TSynCustomHighlighter'), 'TSynPasSyn') do begin
2182: 1821 ***** File 349: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_SynMacroRecorder.pas
2183: 1822 ('TCustomSynMacroRecorder'), 'TSynMacroRecorder') do
2184: 1823 ('TAbstractSynHookerPlugin'), 'TCustomSynMacroRecorder') do
2185: 1824 ('TSynBasicEvent'), 'TSynDataEvent') do
2186: 1825 ('TSynBasicEvent'), 'TSynPositionEvent') do begin
2187: 1826 ('TSynMacroEvent'), 'TSynStringEvent') do begin
2188: 1827 ('TSynMacroEvent'), 'TSynCharEvent') do begin
2189: 1828 ('TSynMacroEvent'), 'TSynBasicEvent') do begin
2190: 1829 ('TObject'), 'TSynMacroEvent') do begin
2191: 1830 ('TOBJECT'), 'TCustomSynMacroRecorder');
2192: 1830 ***** File 350: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_SynMemo.pas
2193: 1831 ('TSynEdit'), 'TSynMemo') do
2194: 1831 ***** File 351: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_SynPdf.pas
2195: 1832 ('TPdfXObject'), 'TPdfForm') do begin
2196: 1833 ('TPdfXObject'), 'TPdfImage') do begin
2197: 1834 ('TPdfDocument'), 'TPdfDocumentGDI') do begin
2198: 1835 ('TPdfPage'), 'TPdfPageGDI') do
2199: 1836 ('TPdfOutlineEntry'), 'TPdfOutlineRoot') do begin
2200: 1837 ('TPdfDictionaryWrapper'), 'TPdfOutlineEntry') do begin
2201: 1838 ('TObject'), 'TPdfDestination') do begin
2202: 1839 ('TPdfFontWinAnsi'), 'TPdfFontTrueType') do begin
2203: 1840 ('TOBJECT'), 'TPdfTTF') do begin
2204: 1841 ('TPdfFont'), 'TPdfFontCIDFontType2') do
2205: 1842 ('TPdfFontWinAnsi'), 'TPdfFontType1') do begin
2206: 1843 ('TPdfFont'), 'TPdfFontWinAnsi') do
2207: 1844 ('TPdfDictionaryWrapper'), 'TPdfFont') do begin
2208: 1845 ('TPdfDictionaryWrapper'), 'TPdfCatalog') do begin
2209: 1846 ('TPdfDictionaryWrapper'), 'TPdfInfo') do begin
2210: 1847 ('TPersistent'), 'TPdfDictionaryWrapper') do begin
2211: 1848 ('TObject'), 'TPdfCanvas') do begin
2212: 1849 ('TPdfDictionary'), 'TPdfPage') do begin
2213: 1850 ('TObject'), 'TPdfDocument') do begin
2214: 1851 ('TPdfObjectMgr'), 'TPdfXref') do begin
2215: 1852 ('TObject'), 'TPdfXrefEntry') do begin
2216: 1853 ('TObject'), 'TPdfTrailer') do begin
2217: 1854 ('TPdfObject'), 'TPdfBinary') do begin
2218: 1855 ('TPdfObject'), 'TPdfStream') do begin
2219: 1856 ('TPdfObject'), 'TPdfDictionary') do begin
2220: 1857 ('TObject'), 'TPdfDictionaryElement') do begin
2221: 1858 ('TPdfObject'), 'TPdfArray') do begin
2222: 1859 ('TPdfText'), 'TPdfName') do
2223: 1860 ('TPdfText'), 'TPdfRawText') do begin
```

```

2224: 1861 ('TPdfTextUTF8'), 'TPdfTextString') do
2225: 1862 ('TPdfObject'), 'TPdfTextUTF8') do
2226: 1863 ('TPdfObject'), 'TPdfText') do begin
2227: 1864 ('TPdfObject'), 'TPdfReal') do
2228: 1865 ('TPdfObject'), 'TPdfNumber') do
2229: 1866 ('TPdfObject'), 'TPdfNull') do
2230: 1867 ('TPdfObject'), 'TPdfBoolean') do begin
2231: 1868 ('TPdfObject'), 'TPdfVirtualObject') do
2232: 1869 ('TObject'), 'TPdfObject') do begin
2233: 1870 ('TObject'), 'TPdfObjectMgr') do begin
2234: 1871 ('TOBJECT'), 'TPdfWrite') do begin
2235: 1872 ('TPdfEncryption'), 'TPdfEncryptionRC4MD5') do
2236: 1873 ('TOBJECT'), 'TPdfEncryption') do
2237: 1874 ('TOBJECT'), 'EPdfInvalidValue');
2238: 1875 ('TOBJECT'), 'EPdfInvalidOperation');
2239: 1876 ('TOBJECT'), 'TPdfObject');
2240: 1877 ('TOBJECT'), 'TPdfCanvas');
2241: 1878 ('TOBJECT'), 'TPdfFont');
2242: 1879 ('TOBJECT'), 'TPdfFontTrueType');
2243: 1880 ('TOBJECT'), 'TPdfDocument');
2244: 1881 ('TOBJECT'), 'TPdfXObject');
2245: 1882 ('TOBJECT'), 'TPdfOutlines');
2246: 1883 ('TOBJECT'), 'TPdfInfo');
2247: 1884 ('TOBJECT'), 'TPdfCatalog');
2248: 1885 ('TOBJECT'), 'TPdfDestination');
2249: 1886 ('TOBJECT'), 'TPdfOutlineEntry');
2250: 1887 ('TOBJECT'), 'TPdfOutlineRoot');
2251: 1888 ('TOBJECT'), 'TPdfPage');
2252: 1888 ***** File 352: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_ValEdit.pas
2253: 1889 ('TPersistent'), 'TItemProp') do
2254: 1890 ('TStringList'), 'TValueListStrings') do
2255: 1891 ('TCustomDrawGrid'), 'TValueListEditor') do
2256: 1892 ('TOBJECT'), 'TItemProp');
2257: 1893 ('TOBJECT'), 'TValueListStrings');
2258: 1893 ***** File 353: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_xrtl_math_Integer.pas
2259: 1894 ('TOBJECT'), 'EXRTLMathException');
2260: 1895 ('TOBJECT'), 'EXRTLExtendInvalidArgument');
2261: 1896 ('TOBJECT'), 'EXRTLDivisionByZero');
2262: 1897 ('TOBJECT'), 'EXRTLExpInvalidArgument');
2263: 1898 ('TOBJECT'), 'EXRTLInvalidRadix');
2264: 1899 ('TOBJECT'), 'EXRTLInvalidRadixDigit');
2265: 1900 ('TOBJECT'), 'EXRTLRootInvalidArgument');
2266: 1900 ***** File 354: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_xrtl_net_NamedPipe.pas
2267: 1901 ('TXRTLNamedPipe'), 'TXRTLNamedPipeClient') do
2268: 1902 ('TXRTLNamedPipe'), 'TXRTLNamedPipeServer') do
2269: 1903 ('TOBJECT'), 'TXRTLNamedPipe') do
2270: 1903 ***** File 355: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_xrtl_net_URI.pas
2271: 1904 ('TOBJECT'), 'TXRTLURI') do begin
2272: 1904 ***** File 356: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_xrtl_util_COMUtils.pas
2273: 1905 ('TXRTLEnumXXXX'), 'TXRTLEnumString') do begin
2274: 1906 ('TXRTLEnumXXXX'), 'TXRTLEnumUnknown') do begin
2275: 1907 ('TInterfacedObject'), 'TXRTLEnumXXXX') do begin
2276: 1907 ***** File 357: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_xrtl_util_Exception.pas
2277: 1908 ('TOBJECT'), 'EXRTLException');
2278: 1909 ('TOBJECT'), 'EXRTLNotImplemented');
2279: 1910 ('TOBJECT'), 'EXRTLInvalidOperation');
2280: 1910 ***** File 358: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_xrtl_util_Map.pas
2281: 1911 ('TXRTLMap'), 'TXRTLArrayMap') do begin
2282: 1912 ('TXRTLMap'), 'TXRTLSynchronizedMap') do begin
2283: 1913 ('TXRTLKeyValueContainer'), 'TXRTLMap') do
2284: 1913 ***** File 359: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_xrtl_util_Set.pas
2285: 1914 ('TXRTLSet'), 'TXRTLArraySet') do
2286: 1915 ('TXRTLSetContainer'), 'TXRTLSet') do
2287: 1915 ***** File 360: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_xrtl_util_TimeStamp.pas
2288: 1916 ('TPersistent'), 'TXRTLTimeStamp') do begin
2289: 1916 ***** File 361: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_xrtl_util_TimeZone.pas
2290: 1917 ('TOBJECT'), 'TXRTLTimeZones') do begin
2291: 1918 ('TOBJECT'), 'TXRTLTimeZone') do begin
2292: 1918 ***** File 362: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_xrtl_util_Value.pas
2293: 1919 ('TOBJECT'), 'TXRTLValue') do begin
2294: 1919 ***** File 363: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\synopse\upsi_JvAirBrush.pas
2295: 1920 ('TJvComponent'), 'TJvAirBrush') do begin
2296: 1920 ***** File 364: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\synopse\upsi_mORMotReport.pas
2297: 1921 ('TGDIPages'), 'TRenderPages') do begin
2298: 1922 ('TOBJECT'), 'TRenderBox') do
2299: 1923 ('TScrollBar'), 'TGDIPages') do
2300: 1924 ('TOBJECT'), 'TGDIPageContent') do
2301: 1925 ('TOBJECT'), 'TGDIPageReference') do
2302: 1926 ('TPaintBox'), 'TPagePaintBox') do
2303: 1927 ('TOBJECT'), 'THeaderFooter') do begin
2304: 1928 ('TOBJECT'), 'TGDIPages');
2305: 1929 ('TOBJECT'), 'TRenderPages');
2306: 1929 ***** File 365: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\synopse\upsi_SynPdf.pas
2307: 1930 ('TPdfXObject'), 'TPdfForm') do begin
2308: 1931 ('TPdfXObject'), 'TPdfImage') do begin
2309: 1932 ('TPdfDocument'), 'TPdfDocumentGDI') do begin
2310: 1933 ('TPdfPage'), 'TPdfPageGDI') do
2311: 1934 ('TPdfOutlineEntry'), 'TPdfOutlineRoot') do begin
2312: 1935 ('TPdfDictionaryWrapper'), 'TPdfOutlineEntry') do begin

```

```
2313: 1936 ('TObject'), 'TPdfDestination') do begin
2314: 1937 ('TPdfFontWinAnsi'), 'TPdfFontTrueType') do begin
2315: 1938 ('TOBJECT'), 'TPdfTTF') do begin
2316: 1939 ('TPdfFont'), 'TPdfFontCIDFontType2') do
2317: 1940 ('TPdfFontWinAnsi'), 'TPdfFontTypel') do begin
2318: 1941 ('TPdfFont'), 'TPdfFontWinAnsi') do
2319: 1942 ('TPdfDictionaryWrapper'), 'TPdfFont') do begin
2320: 1943 ('TPdfDictionaryWrapper'), 'TPdfCatalog') do begin
2321: 1944 ('TPdfDictionaryWrapper'), 'TPdfInfo') do begin
2322: 1945 ('TPersistent'), 'TPdfDictionaryWrapper') do begin
2323: 1946 ('TObject'), 'TPdfCanvas') do begin
2324: 1947 ('TPdfDictionary'), 'TPdfPage') do begin
2325: 1948 ('TObject'), 'TPdfDocument') do begin
2326: 1949 ('TPdfObjectMgr'), 'TPdfXref') do begin
2327: 1950 ('TObject'), 'TPdfXrefEntry') do begin
2328: 1951 ('TObject'), 'TPdfTrailer') do begin
2329: 1952 ('TPdfObject'), 'TPdfBinary') do
2330: 1953 ('TPdfObject'), 'TPdfStream') do begin
2331: 1954 ('TPdfObject'), 'TPdfDictionary') do begin
2332: 1955 ('TObject'), 'TPdfDictionaryElement') do begin
2333: 1956 ('TPdfObject'), 'TPdfArray') do begin
2334: 1957 ('TPdfText'), 'TPdfName') do
2335: 1958 ('TPdfText'), 'TPdfRawText') do begin
2336: 1959 ('TPdfTextUTF8'), 'TPdfTextString') do
2337: 1960 ('TPdfObject'), 'TPdfTextUTF8') do
2338: 1961 ('TPdfObject'), 'TPdfText') do
2339: 1962 ('TPdfObject'), 'TPdfReal') do
2340: 1963 ('TPdfObject'), 'TPdfNumber') do
2341: 1964 ('TPdfObject'), 'TPdfNull') do
2342: 1965 ('TPdfObject'), 'TPdfBoolean') do
2343: 1966 ('TPdfObject'), 'TPdfVirtualObject') do
2344: 1967 ('TObject'), 'TPdfObject') do begin
2345: 1968 ('TObject'), 'TPdfObjectMgr') do
2346: 1969 ('TOBJECT'), 'TPdfWrite') do begin
2347: 1970 ('TPdfEncryption'), 'TPdfEncryptionRC4MD5') do
2348: 1971 ('TOBJECT'), 'TPdfEncryption') do
2349: 1972 ('TOBJECT'), 'EPdfInvalidValue');
2350: 1973 ('TOBJECT'), 'EPdfInvalidOperation');
2351: 1974 ('TOBJECT'), 'TPdfObject');
2352: 1975 ('TOBJECT'), 'TPdfCanvas');
2353: 1976 ('TOBJECT'), 'TPdfFont');
2354: 1977 ('TOBJECT'), 'TPdfFontTrueType');
2355: 1978 ('TOBJECT'), 'TPdfDocument');
2356: 1979 ('TOBJECT'), 'TPdfXObject');
2357: 1980 ('TOBJECT'), 'TPdfOutlines');
2358: 1981 ('TOBJECT'), 'TPdfInfo');
2359: 1982 ('TOBJECT'), 'TPdfCatalog');
2360: 1983 ('TOBJECT'), 'TPdfDestination');
2361: 1984 ('TOBJECT'), 'TPdfOutlineEntry');
2362: 1985 ('TOBJECT'), 'TPdfOutlineRoot');
2363: 1986 ('TOBJECT'), 'TPdfPage');
2364: 1986 ***** File 366: F:\maxbox3\mX397\maxbox3\source\REST\uPSI_cHTTPUtils.pas
2365: 1987 ('TOBJECT'), 'TAnsiStringBuilder') do begin
2366: 1988 ('TOBJECT'), 'THTTPContentWriter') do begin
2367: 1989 ('TOBJECT'), 'THTTPContentReader') do begin
2368: 1990 ('TOBJECT'), 'THTTPContentDecoder') do begin
2369: 1991 ('TOBJECT'), 'cTHTTPParser') do begin
2370: 1992 ('TOBJECT'), 'EHTTP');
2371: 1993 ('TOBJECT'), 'EHTTPParser');
2372: 1994 ('TOBJECT'), 'TAnsiStringBuilder');
2373: 1995 ('TOBJECT'), 'THTTPContentDecoder');
2374: 1996 ('TOBJECT'), 'THTTPContentReader');
2375: 1997 ('TOBJECT'), 'THTTPContentWriter');
2376: 1997 ***** File 367: F:\maxbox3\mX397\maxbox3\source\REST\uPSI_cJSON.pas
2377: 1998 ('TOBJECT'), 'TJSONParser') do
2378: 1999 ('TJSONValue'), 'TJSONObject') do begin
2379: 2000 ('TJSONValue'), 'TJSONArray') do
2380: 2001 ('TJSONValue'), 'TJSONNull') do
2381: 2002 ('TJSONValue'), 'TJSONBoolean') do
2382: 2003 ('TJSONValue'), 'TJSONFloat') do
2383: 2004 ('TJSONValue'), 'TJSONInteger') do
2384: 2005 ('TJSONValue'), 'TJSONString') do
2385: 2006 ('TOBJECT'), 'TJSONValue') do
2386: 2007 ('TOBJECT'), 'EJSONValue');
2387: 2008 ('TOBJECT'), 'EJSONSchema');
2388: 2009 ('TOBJECT'), 'TJSONObject');
2389: 2010 ('TOBJECT'), 'EJSONParser');
2390: 2010 ***** File 368: F:\maxbox3\mX397\maxbox3\source\REST\uPSI_CppTokenizer.pas
2391: 2011 ('TComponent'), 'TCppTokenizer') do begin
2392: 2011 ***** File 369: F:\maxbox3\mX397\maxbox3\source\REST\uPSI_cTLSUtils.pas
2393: 2012 ('Exception'), 'ETLSerror') do begin
2394: 2012 ***** File 370: F:\maxbox3\mX397\maxbox3\source\REST\uPSI_cXMLFunctions.pas
2395: 2013 ('TOBJECT'), 'Exml');
2396: 2013 ***** File 371: F:\maxbox3\mX397\maxbox3\source\REST\uPSI_devExec.pas
2397: 2014 ('TPersistent'), 'TdevExecutor') do begin
2398: 2015 ('TThread'), 'TExecThread') do begin
2399: 2015 ***** File 372: F:\maxbox3\mX397\maxbox3\source\REST\uPSI_devFileMonitorX.pas
2400: 2016 ('TWinControl'), 'TdevFileMonitor') do begin
2401: 2016 ***** File 373: F:\maxbox3\mX397\maxbox3\source\REST\uPSI_devrun.pas
```



```

2402: 2017 ('TThread'), 'TDevRun') do begin
2403: 2017 ***** File 374: F:\maxbox3\mX397\maxbox3\source\REST\upsi_Diff.pas
2404: 2018 ('TComponent'), 'TDiff') do begin
2405: 2018 ***** File 375: F:\maxbox3\mX397\maxbox3\source\REST\upsi_DosCommand.pas
2406: 2019 ('TComponent'), 'TDosCommand') do begin
2407: 2020 ('TThread'), 'TDosThread') do begin
2408: 2021 ('TTimer'), 'TProcessTimer') do begin
2409: 2022 ('TObject'), 'TCreatePipeError');
2410: 2023 ('TObject'), 'TCreateProcessError');
2411: 2024 ('TObject'), 'TDosCommand');
2412: 2024 ***** File 376: F:\maxbox3\mX397\maxbox3\source\REST\upsi_GR32_VectorMaps.pas
2413: 2025 ('TCustomMap'), 'TVectorMap') do begin
2414: 2025 ***** File 377: F:\maxbox3\mX397\maxbox3\source\REST\upsi_HttpRESTConnectionIndy.pas
2415: 2026 ('TInterfacedObject'), 'THttpConnectionIndy') do begin
2416: 2027 ('TIdHTTP'), 'TIdHTTP') do begin
2417: 2028 ('TInterface'), 'IHttpConnection');
2418: 2028 ***** File 378: F:\maxbox3\mX397\maxbox3\source\REST\upsi_HyperLabel.pas
2419: 2029 ('TCustomLabel'), 'THyperLabel') do begin
2420: 2029 ***** File 379: F:\maxbox3\mX397\maxbox3\source\REST\upsi_JclCOM.pas
2421: 2030 ('TObject'), 'EInvalidParam');
2422: 2030 ***** File 380: F:\maxbox3\mX397\maxbox3\source\REST\upsi_JclGraphics.pas
2423: 2031 ('TJclTransformation'), 'TJclLinearTransformation') do
2424: 2032 ('TObject'), 'TJclTransformation') do
2425: 2033 ('TJclCustomMap'), 'TJclByteMap') do begin
2426: 2034 ('TJclCustomMap'), 'TJclBitmap32') do begin
2427: 2035 ('TJclThreadPersistent'), 'TJclCustomMap') do begin
2428: 2036 ('TPersistent'), 'TJclThreadPersistent') do begin
2429: 2037 ('TObject'), 'TJclRegion') do begin
2430: 2038 ('TObject'), 'TJclRegionInfo') do begin
2431: 2039 ('TCanvas'), 'TJclDesktopCanvas') do
2432: 2040 ('TObject'), 'EJclGraphicsError');
2433: 2041 ('TObject'), 'TJclRegion');
2434: 2041 ***** File 381: F:\maxbox3\mX397\maxbox3\source\REST\upsi_JclMapi.pas
2435: 2042 ('TJclSimpleMapi'), 'TJclEmail') do begin
2436: 2043 ('TObjectList'), 'TJclEmailRecips') do begin
2437: 2044 ('TObject'), 'TJclEmailRecip') do begin
2438: 2045 ('TObject'), 'TJclSimpleMapi') do begin
2439: 2046 ('EJclError'), 'EJclMapiError') do begin
2440: 2046 ***** File 382: F:\maxbox3\mX397\maxbox3\source\REST\upsi_JclSynch.pas
2441: 2047 ('TObject'), 'TJclMeteredSection') do
2442: 2048 ('TObject'), 'TJclMultiReadExclusiveWrite') do
2443: 2049 ('TObject'), 'TJclOptex') do
2444: 2050 ('TJclDispatcherObject'), 'TJclMutex') do
2445: 2051 ('TJclDispatcherObject'), 'TJclSemaphore') do
2446: 2052 ('TJclDispatcherObject'), 'TJclWaitableTimer') do
2447: 2053 ('TJclDispatcherObject'), 'TJclEvent') do begin
2448: 2054 ('TJclCriticalSection'), 'TJclCriticalSectionEx') do
2449: 2055 ('TObject'), 'TJclCriticalSection') do begin
2450: 2056 ('TObject'), 'TJclDispatcherObject') do begin
2451: 2057 ('TObject'), 'EJclWin32HandleObjectError');
2452: 2058 ('TObject'), 'EJclDispatcherObjectError');
2453: 2059 ('TObject'), 'EJclCriticalSectionError');
2454: 2060 ('TObject'), 'EJclEventError');
2455: 2061 ('TObject'), 'EJclWaitableTimerError');
2456: 2062 ('TObject'), 'EJclSemaphoreError');
2457: 2063 ('TObject'), 'EJclMutexError');
2458: 2064 ('TObject'), 'EJclMeteredSectionError');
2459: 2064 ***** File 383: F:\maxbox3\mX397\maxbox3\source\REST\upsi_JvHighlighter.pas
2460: 2065 ('TGraphicControl'), 'TJvHighlighter') do begin
2461: 2065 ***** File 384: F:\maxbox3\mX397\maxbox3\source\REST\upsi_JvHLPParser.pas
2462: 2066 ('Exception'), 'EJvIParserError') do
2463: 2067 ('TObject'), 'TJvIParserW') do begin
2464: 2068 ('TObject'), 'TJvIParser') do begin
2465: 2068 ***** File 385: F:\maxbox3\mX397\maxbox3\source\REST\upsi_JvTimer.pas
2466: 2069 ('TComponent'), 'TJvTimer') do begin
2467: 2069 ***** File 386: F:\maxbox3\mX397\maxbox3\source\REST\upsi_JvXMLDatabase.pas
2468: 2070 ('TJvComponent'), 'TJvXMLDatabase') do
2469: 2071 ('TObject'), 'TJvXMLQuery') do
2470: 2072 ('TObject'), 'TJvXMLQueryParser') do
2471: 2073 ('TObject'), 'TJvXMLQueryAssignment') do
2472: 2074 ('TObject'), 'TJvXMLQueryCondition') do
2473: 2075 ('TObject'), 'TJvXMLQueryOrder') do
2474: 2076 ('TObject'), 'TJvXMLQueryColumn') do
2475: 2077 ('TObject'), 'TJvXMLQueryTable') do
2476: 2078 ('TObject'), 'TJvXMLTable') do
2477: 2079 ('TObject'), 'TJvXMLDatabase');
2478: 2080 ('TObject'), 'TJvXMLQuery');
2479: 2081 ('TObject'), 'TJvXMLQueryParser');
2480: 2082 ('TObject'), 'TJvXMLDatabaseException');
2481: 2082 ***** File 387: F:\maxbox3\mX397\maxbox3\source\REST\upsi_MultilangTranslator.pas
2482: 2083 ('TComponent'), 'TMultiLangSC') do begin
2483: 2083 ***** File 388: F:\maxbox3\mX397\maxbox3\source\REST\upsi_OysUtils.pas
2484: 2084 ('TStringList'), 'ToysStringList') do begin
2485: 2084 ***** File 389: F:\maxbox3\mX397\maxbox3\source\REST\upsi_RestRequest.pas
2486: 2085 ('TObject'), 'TRestRequest') do begin
2487: 2085 ***** File 390: F:\maxbox3\mX397\maxbox3\source\REST\upsi_Services.pas
2488: 2086 ('TObject'), 'TService') do begin
2489: 2086 ***** File 391: F:\maxbox3\mX397\maxbox3\source\REST\upsi_SimpleHl.pas
2490: 2087 ('TSynCustomHighlighter'), 'TSynDemoHl') do begin

```



```

2491: 2087 ***** File 392: F:\maxbox3\mX397\maxbox3\source\REST\upsi_Starter.pas
2492: 2088 ('TOBJECT'), 'TStarter') do begin
2493: 2088 ***** File 393: F:\maxbox3\mX397\maxbox3\source\REST\upsi_TomDBQue.pas
2494: 2089 ('TComponent'), 'TTomBackgroundQuery') do begin
2495: 2090 ('TThread'), 'TTomQueryThread') do begin
2496: 2090 ***** File 394: F:\maxbox3\mX397\maxbox3\source\REST\upsi_XmlVerySimple.pas
2497: 2091 ('TObject'), 'TXmlVerySimple') do begin
2498: 2092 ('TObject'), 'TXmlNode') do begin
2499: 2093 ('TObjectList'), 'TXmlAttributeList') do begin
2500: 2094 ('TObject'), 'TXmlAttribute') do begin
2501: 2095 ('TOBJECTList'), 'TXmlNodeList');
2502: 2096 ('TOBJECTList'), 'TXmlNodeList');
2503: 2096 ***** File 395: F:\maxbox3\mX397\maxbox3\units\upsi_ShellZipTool.pas
2504: 2097 ('TObject'), 'TShellZip') do begin
2505: 2098
2506: 2099 End CLASS List mX4
2507: 2100 *****
2508:
2509: TYPE List mX4 3.9.9.7
2510: *****
2511: 0 ***** File 1: F:\maxbox3\mX397\maxbox3\source\gewinntmax3.pas
2512: 1 TButtonSet, 'set of TMPBtnType';
2513: 1 ***** File 2: F:\maxbox3\mX397\maxbox3\source\IFSI_gsUtils.pas
2514: 2 TS_ShellExecuteCmd, '( seCmdOpen, seCmdPrint, seCmdExplore )';
2515: 2 ***** File 3: F:\maxbox3\mX397\maxbox3\source\IFSI_IdFTP.pas
2516: 3 TIdCreateFTPList, 'Procedure ( ASender : TObject; var VFTPList '
2517: 4 TIdCheckListFormat, 'Procedure ( ASender : TObject; const ALine'
2518: 5 TIdFTPTransferType, '( ftASCII, ftBinary )';
2519: 6 TIdFTPDataStructure, '( dsFile, dsRecord, dsPage )';
2520: 7 TIdFTPTransferMode, '( dmBlock, dmCompressed, dmStream )';
2521: 8 TOnAfterClientLogin, 'TNotifyEvent';
2522: 9 TIdFtpAfterGet, 'Procedure ( ASender : TObject; VStream : TStre'
2523: 10 TIdFtpProxyType, '( fpcmNone, fpcmUserSite, fpcmSite, fpcmOpen,'
2524: 10 ***** File 4: F:\maxbox3\mX397\maxbox3\source\IFSI_IdGlobalProtocols.pas
2525: 11 TIdFileName, 'WideString';
2526: 12 TIdFileName, 'String';
2527: 13 TIdReadLnFunction, 'Function : string';
2528: 14 PByte, '^Byte // will not work';
2529: 15 PWord, '^Word // will not work';
2530: 16 TIdWin32Type, '( Win32s, WindowsNT40PreSP6Workstation, WindowsN'
2531: 16 ***** File 5: F:\maxbox3\mX397\maxbox3\source\IFSI_IdHTTP.pas
2532: 17 TIdHTTPMethod, '( hmHead, hmGet, hmPost, hmOptions, hmTrace, hm'
2533: 18 TIdHTTPWhatsNext, '( wnGoToURL, wnJustExit, wnDontKnow, wnReadA'
2534: 19 TIdHTTPConnectionType, '( ctNormal, ctSSL, ctProxy, ctSSLProxy)';
2535: 20 TIdHTTPOption, '( hoInProcessAuth, hoKeepOrigProtocol, hoForceEncodeParams )';
2536: 21 TIdHTTPOptions, 'set of TIdHTTPOption';
2537: 22 TIdHTTPProtocolVersion, '( pv1_0, pv1_1 )';
2538: 23 TIdHTTPOnRedirectEvent, 'Procedure ( Sender : TObject; var dest'
2539: 24 TIdOnSelectAuthorization, 'Procedure ( Sender : TObject; '+
2540: 25 TIdOnAuthorization, 'Procedure ( Sender : TObject; Authenticati'
2541: 25 ***** File 6: F:\maxbox3\mX397\maxbox3\source\IFSI_IdURI.pas
2542: 26 TIdURIOptionalFields, '( ofAuthInfo, ofBookmark )';
2543: 27 TIdURIOptionalFieldsSet, 'set of TIdURIOptionalFields';
2544: 27 ***** File 7: F:\maxbox3\mX397\maxbox3\source\IFSI_SysUtils_max.pas
2545: 28 TConstArray, 'array of const';
2546: 29 PDayTable, '^TDayTable // will not work';
2547: 30 TFilenameCaseMatch, '( mkNone, mkExactMatch, mkSingleMatch, mkAmbiguous )';
2548: 31 TReplaceFlags, 'set of ( rfReplaceAll, rfIgnoreCase )';
2549: 32 TSysCharSet, 'set of Char'; //!!! not to define!!! in StrUtils!
2550: 33 FILEOP_FLAGS, 'Word';
2551: 34 TCaption, 'string';
2552: 35 CONTEXT, '_CONTEXT';
2553: 36 mUINT64, 'UInt64'; //3.6.3.
2554: 37 HRSCR, 'THandle';
2555: 38 TResourceHandle, 'HRSCR';
2556: 39 HMODULE, 'THandle';
2557: 40 HINST, 'THandle';
2558: 41 HFONT, 'LongWord';
2559: 42 HGLOBAL, 'THandle';
2560: 43 ATOM, 'Word';
2561: 44 TATOM, 'Word';
2562: 45 HWND, 'LongWord';
2563: 46 HMETAFILE, 'LongWord';
2564: 47 HBITMAP, 'LongWord';
2565: 48 HDC, 'LongWord';
2566: 49 HCURSOR, 'LongWord';
2567: 50 HBRUSH, 'LongWord';
2568: 51 HICON, 'LongWord';
2569: 52 HPALETTE, 'LongWord';
2570: 53 HRSRC, 'LongWord'; //Windows resource
2571: 54 HKL, 'LongWord'; //keyboard layout
2572: 55 HKEY, 'LongWord'; //win reg layout
2573: 56 HFILE, 'LongWord'; //A handle to an open file
2574: 57 HGDIOBJ, 'LongWord'; //a GDI object. Pens, device contexts, brushes, etc.
2575: 58 LPARAM, 'LongInt';
2576: 59 WPARAM, 'LongInt';
2577: 60 LRESULT, 'Longint';
2578: 61 ULONG, 'Longint';
2579: 62 UNT, 'Integer';

```

```

2580: 63 UINT', 'LongWord');
2581: 64 PUINT', '^UINT // will not work');
2582: 65 ULONG', 'Cardinal');
2583: 66 HRGN', 'LongWord');
2584: 67 HSTR', 'LongWord');
2585: 68 HTASK', 'LongWord');
2586: 69 HWINSTA', 'LongWord');
2587: 70 HKL', 'LongWord');
2588: 71 HFILE', 'LongWord');
2589: 72 HCURSOR', 'HICON');
2590: 73 COLORREF', 'DWORD');
2591: 74 MMResult', 'UInt');
2592: 75 TLargeInteger', 'Int64');
2593: 76 UCHAR', 'Byte');
2594: 77 BOOL', 'LongBool'); //long !
2595: 78 SHORT', 'SmallInt');
2596: 79 LCID', 'DWord'); //A local identifier
2597: 80 TColorRef', 'DWORD');
2598: 81 LOWORD', 'Word');
2599: 82 FARPROC', '__Pointer');
2600: 83 TFARPROC', '__Pointer');
2601: 84 LPCSTR', 'PAnsiChar');
2602: 85 LPCTSTR', 'PAnsiChar');
2603: 86 LPTSTR', 'PAnsiChar');
2604: 87 TFNThreadStartRoutine', 'TFarProc');
2605: 88 TFNFiberStartRoutine', 'TFarProc');
2606: 89 TThreadFunction2', 'procedure; stdcall');
2607: 90 _RTL_CRITICAL_SECTION', 'record DebugInfo: TFARPROC; LockCount: Longint;'+
2608: 91 TRTLCriticalSection', '_RTL_CRITICAL_SECTION');
2609: 92 RTL_CRITICAL_SECTION', '_RTL_CRITICAL_SECTION');
2610: 93 LANGID', 'Word'); //A language identifier
2611: 94 RT_RCDATA', 'Types.RT_RCDATA'); //MakeIntResource(10);
2612: 95 RT_RCDATA', 'PChar(10)'); //MakeIntResource(10);
2613: 96 RT_CURSOR', 'PChar(1)'); //MakeIntResource(10);
2614: 97 RT_BITMAP', 'PChar(2)'); //MakeIntResource(10);
2615: 98 RT_ICON', 'PChar(3)'); //MakeIntResource(10);
2616: 99 RT_STRING', 'PChar(6)'); //MakeIntResource(10);
2617: 100 RT_FONT', 'PChar(8)'); //MakeIntResource(10);
2618: 101 RT_DIALOG', 'PChar(5)'); //MakeIntResource(10);
2619: 102 RT_MESSAGETABLE', 'PChar(11)'); //MakeIntResource(10);
2620: 103 TMsg, 'record hwnd: HWND; message: longword; wParam: WPARAM; lParam: LPARAM; time: DWORD; pt: TPoint; end;
2621: 104 INT_PTR', 'Integer');
2622: 105 LONG_PTR', 'Integer');
2623: 106 UINT_PTR', 'Cardinal');
2624: 107 ULONG_PTR', 'Cardinal');
2625: 108 DWORD_PTR', 'ULONG_PTR');
2626: 109 TFloatingSaveArea', '__Pointer');
2627: 110 FLOATING_SAVE_AREA', '_FLOATING_SAVE_AREA');
2628: 111 _CONTEXT', 'record ContextFlags: DWORD; Dr0 : DWORD; Dr1 : DWORD; Dr2 : DWORD; Dr3 : DWORD; Dr6 :
DWord; Dr7 : DWORD; FloatSave : TFloati';
2629: 112 TContext', '_CONTEXT');
2630: 113 UInt64', 'UInt64');
2631: 114 IXMLNode', 'interface');
2632: 115 IXMLNode', 'IInterface');
2633: 116 TTreeView', 'TCustomTreeView');
2634: 117 TTreeView', 'TObject');
2635: 118 TTreeNode', 'TPersistent');
2636: 119 TFloatRec', 'record Exponent: Smallint; Negative: Boolean; Digits: array[0..20] of Char; end');
2637: 120 TSysLocale', 'record DefaultLCID, PriLangID, SubLangID: Integer; FarEast, MiddleEast: Boolean; end');
2638: 121 TIdentMapEntry', 'record Value: Integer; Name: String; end');
2639: 122 TMessage', 'record msg: Cardinal; wParam, lParam, result: Longint; end');
2640: 123 TMessage2, 'record msg: Cardinal; wParam, lParam, lParamhi, lParamlo, resultlo, resulthi: word; end;
2641: 124 TTerminateProc', 'function: Boolean');
2642: 125 MakeIntResource', 'PANSIChar');
2643: 126 THexArray', 'array[0..15] of char;');
2644: 127 TIntegerSet', 'set of Integer');
2645: 128 PByteArray', '^TByteArray // will not work');
2646: 129 PWordArray', '^TWordArray // will not work');
2647: 130 TFileName', 'type string');
2648: 131 TFileName', 'string');
2649: 132 Text', 'File of char');
2650: 133 CmdLine', 'PChar');
2651: 134 TGraphic', 'Class');
2652: 135 TGraphicClass', 'Class of TGraphic');
2653: 136 TPersistentClass', 'TPersistent');
2654: 137 class of TPersistent', 'TPersistentClass');
2655: 138 TShiftState', '(ssShift, ssAlt, ssCtrl, ssLeft, ssRight, ssMiddle, ssDouble)');
2656: 139 TKeyboardState', 'array[0..255] of Byte');
2657: 140 _FILETIME', 'record dwLowDateTime: DWORD; dwHighDateTime: DWORD; end');
2658: 141 TFileTime', '_FILETIME');
2659: 142 FILETIME', '_FILETIME');
2660: 143 _SYSTEMTIME', 'record wYear : Word; wMonth : Word; wDayOfWeek : '
2661: 144 TSystemTime', '_SYSTEMTIME');
2662: 145 SYSTEMTIME', '_SYSTEMTIME');
2663: 146 _BY_HANDLE_FILE_INFORMATION', 'record dwFileAttributes : DWORD; '
2664: 147 TByHandleFileInformation', '_BY_HANDLE_FILE_INFORMATION');
2665: 148 TSearchRec', 'record Time : Integer; Size : Integer; Attr : Integer; '
2666: 149 TFloatValue', '( fvExtended, fvCurrency )');
2667: 150 TFloatFormat', '( ffGeneral, ffExponent, ffFixed, ffNumber, ffCurrency )');

```

```

2668: 151 TTextLineBreakStyle', '(tlbsLF, tlbsCRLF)';
2669: 152 TTimeStamp', 'record Time : Integer; Date : Integer; end';
2670: 153 TWordArray', 'array[0..16383] of Word;';
2671: 154 TByteArray', 'array[0..32767] of Byte;'; //also in unit uPSI_StrUtils
2672: 155 TCharArray', 'array[0..32767] of Char;';
2673: 156 TBytes', 'array of Byte;'; //case of write16bitresourceheader !!
2674: 157 TRect2', 'record TopLeft, BottomRight: TPoint; end;'; //3.8
2675: 158 TFontCharset', '0..255'; // from sysutils tfontcharset
2676: 159 marrstr', 'array of string;';
2677: 160 Tdnames', 'array[1..7] of string;';
2678: 161 Tmnames', 'array[1..12] of string;';
2679: 162 TDayNames', 'array[1..7] of string;';
2680: 163 TMonthNames', 'array[1..12] of string;';
2681: 164 TStreamOriginalFormat', '(sofUnknown, sofBinary, sofText)';
2682: 165 TStreamOriginalFormat', '( sofUnknown, sofBinary, sofText )';
2683: 166 TmbcsByteType', '( mbSingleByte, mbLeadByte, mbTrailByte )';
2684: 167 TEraRange', 'record StartDate : Integer; EndDate : Integer; end';
2685: 168 ExceptClass', 'class of Exception';
2686: 169 PExceptionRecord', '^TExceptionRecord // will not work';
2687: 170 TSignalState', '( ssNotHooked, ssHooked, ssOverridden )';
2688: 171 MaxEraCount', 'Integer';
2689: 172 TWndMethod', 'Procedure (var Message : TMessage)';
2690: 173 TTextLineBreakStyle', '(tlbsLF, tlbsCRLF)';
2691: 174 UTF8String', 'String;';
2692: 175 UCS4Char', 'LongWord;';
2693: 176 UCS4String', 'array of UCS4Char;';
2694: 177 IntegerArray', 'array[0..$ffffff] of Integer;';
2695: 178 marrastr', 'array of string;';
2696: 179 TRuntimeError', '( reNone, reOutOfMemory, reInvalidPtr, reDivByZ'
2697: 180 va_list', 'PChar';
2698: 181 TSystemInfo', '_SYSTEM_INFO';
2699: 182 SYSTEM_INFO', '_SYSTEM_INFO';
2700: 183 TFarProc', '__Pointer';
2701: 184 TFNTThreadStartRoutine', 'TFarProc';
2702: 185 TFNTTopLevelExceptionFilter', 'TFarProc';
2703: 186 _CONTEXT', 'record ContextFlags : DWORD; Dr0 : DWORD; Dr1 : DWOR'
2704: 187 TContext', '_CONTEXT';
2705: 188 CONTEXT', '_CONTEXT';
2706: 189 tagKERNINGPAIR', 'record wFirst : Word; wSecond : Word; iKernAmount : Integer; end';
2707: 190 TKerningPair', 'tagKERNINGPAIR';
2708: 191 KERNINGPAIR', 'tagKERNINGPAIR';
2709: 191 ***** File 8: F:\maxbox3\mX397\maxbox3\source\IFSI_WinFormlpuzzle.pas
2710: 192 TThreadFunction', 'function: Longint; stdcall';
2711: 193 TThreadFunction3', 'function: Longint; stdcall';
2712: 194 TThreadFunction22', 'procedure; stdcall';
2713: 195 THexArray', 'array[0..15] of char;';
2714: 196 THexSet', 'HexDigits';
2715: 197 TFloatPoint2', 'record X, Y: Double; end';
2716: 198 TPointArray', 'array of TPoint;';
2717: 199 shortstring', 'string[0..255];';
2718: 200 Float', 'Double;';
2719: 201 Real', 'Double;';
2720: 202 Int', 'Integer;';
2721: 203 Long', 'Int64;';
2722: 204 Pointer2', '__Pointer;'; //3.8.1
2723: 205 TBitmapStyle', '(bsNormal, bsCentered, bsStretched)';
2724: 206 T2IntegerArray', 'array of array of integer;';
2725: 207 T2StringArray', 'array of array of string;';
2726: 208 Short', 'Smallint'; in Id_Global
2727: 208 ***** File 9: F:\maxbox3\mX397\maxbox3\source\upSC_classes.pas
2728: 209
2729: 210 TSeekOrigin', '( soBeginning, soCurrent, soEnd )';
2730: 211 TAlignment', '( taLeftJustify, taRightJustify, taCenter )';
2731: 212 TbiDiMode', '( bdLeftToRight, bdRightToLeft, bdRightToLeftNoAlig'
2732: 213 TVerticalAlignment', '( taAlignTop, taAlignBottom, taVerticalCenter )';
2733: 214 TShiftState', 'set of ( ssShift, ssAlt, ssCtrl, ssLeft, ssRight, '
2734: 215 TStringListSortCompare', 'function(List: TStringList; Index1, Index2: Integer): Integer';
2735: 216 TValueType', '( vaNull, vaList, vaInt8, vaInt16, vaInt32, vaExte'
2736: 217 TGetModuleProc', 'Procedure ( const FileName, UnitName, FormName'
2737: 218 TThreadMethod', 'Procedure';
2738: 219 TThreadPriority', '( tpIdle, tpLowest, tpLower, tpNormal, tpHigh'
2739: 220 TSynchronizeRecord', 'record FThread : TObject; FMethod : TThrea'
2740: 221 THelpType', '( htKeyword, htContext )';
2741: 222 TNotifyEvent', 'Procedure ( Sender : TObject)';
2742: 223 TGetStrProc', 'Procedure ( const S : string)';
2743: 224 TfilerFlag', '( ffInherited, ffChildPos, ffInline )';
2744: 225 TfilerFlags', 'set of TfilerFlag';
2745: 226 TReaderProc', 'Procedure ( Reader : TReader)';
2746: 227 TWriterProc', 'Procedure ( Writer : TWriter)';
2747: 228 TStreamProc', 'Procedure ( Stream : TStream)';
2748: 229 TListNotification', '( lnAdded, lnExtracted, lnDeleted )';
2749: 230 TListAssignOp', '( laCopy, laAnd, laOr, laXor, laSrcUnique, laDestUnique )';
2750: 231 TOperation', '( opInsert, opRemove )';
2751: 232 TComponentState', '( csLoading, csReading, csWriting, csD'
2752: 233 TComponentStyle', '( csInheritable, csCheckPropAvail, csS'
2753: 234 TAlignment', '(taLeftJustify, taRightJustify, taCenter)';
2754: 235 THelpEvent', 'function (Command:Word; Data:Longint;var CallHelp:Boolean):Boolean';
2755: 236 TGetStrProc', 'procedure(const S: string)';
2756: 237 TDuplicates', '(dupIgnore, dupAccept, dupError)';

```

```

2757: 238 TOperation', '(opInsert, opRemove)';
2758: 239 THANDLE', 'Longint';
2759: 240 TNotifyEvent', 'procedure (Sender: TObject)';
2760: 241
2761: 242 TByteDynArray', 'array of Byte'; //of upsi_types
2762: 243 TCharDynArray', 'array of Char';
2763: 244 TCharArray', 'array[0..32767] of Char'; //ifsi_sysutils
2764: 245 TStreamOwnership', '(soReference, soOwned)';
2765: 245 ***** File 10: F:\maxbox3\mX397\maxbox3\source\upSC_controls.pas
2766: 246 TShiftState', '(ssShift, ssAlt, ssCtrl, ssLeft, ssRight, ssMiddle, ssDouble)';
2767: 247 TShiftState', '(ssShift, ssAlt, ssCtrl, ssLeft, ssRight, ssMiddle, ssDouble, '+'
2768: 248 TShiftState', 'set of TShiftState';
2769: 249 TMouseButton', '(mbLeft, mbRight, mbMiddle)';
2770: 250 TDragMode', '(dmManual, dmAutomatic)';
2771: 251 TDragState', '(dsDragEnter, dsDragLeave, dsDragMove)';
2772: 252 TDragKind', '(dkDrag, dkDock)';
2773: 253 TMouseEvent', 'procedure (Sender: TObject; Button: TMouseButton; Shift: TShiftState; X, Y: Integer)';
2774: 254 TMouseMoveEvent', 'procedure (Sender: TObject; Shift: TShiftState; X, Y: Integer)';
2775: 255 TKeyEvent', 'procedure (Sender: TObject; var Key: Word; Shift: TShiftState)';
2776: 256 TKeyPressEvent', 'procedure (Sender: TObject; var Key: Char)';
2777: 257 TDragOverEvent', 'procedure (Sender, Source: TObject; X, Y: Integer; State: TDragState; var Accept: Boolean);
2778: 258 TDragDropEvent', 'procedure (Sender, Source: TObject; X, Y: Integer)';
2779: 259 HWND', 'Longint';
2780: 260 TEndDragEvent', 'procedure (Sender, Target: TObject; X, Y: Integer)';
2781: 261 TAlign', '(alNone, alTop, alBottom, alLeft, alRight, alClient)';
2782: 262 TAnchorKind', '(akTop, akLeft, akRight, akBottom)';
2783: 263 TAnchors', 'set of TAnchorKind';
2784: 264 TModalResult', 'Integer';
2785: 265 TCursor', 'Integer';
2786: 266 TPoint', 'record x, y: Longint; end';
2787: 267 TStartDragEvent', 'procedure (Sender: TObject; var DragObject: TDragObject)';
2788: 267 ***** File 11: F:\maxbox3\mX397\maxbox3\source\upSC_DB.pas
2789: 268 TFieldType', '(ftUnknown, ftString, ftSmallint, ftInteger, ftWord, ftBoolean, ftFloat, ftCurrency,
ftBCD, ftDate, ftTime, ftDateTime, '+
2790: 269 TFieldType', '(ftUnknown, ftString, ftSmallint, ftInteger, ftWord
2791: 270 TDataSetState', '(dsInactive, dsBrowse, dsEdit, dsInsert, dsSetKey, dsCalcFields, dsFilter, dsNewValue,
dsOldValue, dsCurValue, dsBlockRead, dsInternalCalc, dsOpening)';
2792: 271 TLocateOption', '(loCaseInsensitive, loPartialKey)';
2793: 275 TUpdateKind', '(ukModify, ukInsert, ukDelete)';
2794: 276 TUpdateAction', '(uaFail, uaAbort, uaSkip, uaRetry, uaApplied)';
2795: 277 TUpdateRecordTypes', '(rtModified, rtInserted, rtDeleted, rtUnmodified)';
2796: 278 TUpdateMode', '(upWhereAll, upWhereChanged, upWhereKeyOnly)';
2797: 279 TSPCommandType', '(ctUnknown, ctQuery, ctTable, ctStoredProc, ctSelect'
2798: 280 TPARAMTYPE', 'BYTE';
2799: 281 TDataEvent', '(deFieldChange, deRecordChange, deDataSetChange, '
2800: 282 TUpdateStatus', '(usUnmodified, usModified, usInserted, usDeleted)';
2801: 283 TUpdateStatusSet', 'set of TUpdateStatus';
2802: 284 TUpdateAction', '(uaFail, uaAbort, uaSkip, uaRetry, uaApplied)';
2803: 285 TUpdateRecordTypes', 'set of (rtModified, rtInserted, rtDeleted'
2804: 286 TUpdateMode', '(upWhereAll, upWhereChanged, upWhereKeyOnly)';
2805: 287 TUpdateKind', '(ukModify, ukInsert, ukDelete)';
2806: 288 TUpdateErrorEvent', 'Procedure (DataSet : TDataSet; E : EDataab
2807: 289 TUpdateRecordEvent', 'Procedure (DataSet : TDataSet; UpdateKind'
2808: 290 TConnectChangeEvent', 'Procedure (Sender : TObject; Connecting: Boolean)';
2809: 291 TBufferList', 'array of PChar';
2810: 292 TGetMode', '(gmCurrent, gmNext, gmPrior)';
2811: 293 TGetResult', '(grOK, grBOF, grEOF, grError)';
2812: 294 TResyncMode', '(rmExact, rmCenter)';
2813: 295 TDataAction', '(daFail, daAbort, daRetry)';
2814: 296 TBlobStreamMode', '(bmRead, bmWrite, bmReadWrite)';
2815: 297 TLocateOption', '(loCaseInsensitive, loPartialKey)';
2816: 298 TLocateOptions', 'set of TLocateOption';
2817: 299 TDataOperation', 'Procedure';
2818: 300 TDataSetNotifyEvent', 'Procedure (DataSet : TDataSet)';
2819: 301 TDataSetErrorEvent', 'Procedure (DataSet : TDataSet; E : EDataab
2820: 302 TFilterOption', '(foCaseInsensitive, foNoPartialCompare)';
2821: 303 TFilterOptions', 'set of TFilterOption';
2822: 304 TFilterRecordEvent', 'Procedure (DataSet : TDataSet; var Accept: Boolean)';
2823: 305 TDEFUPDITEMETHOD', 'Procedure';
2824: 306 TFIELDATTRIBUTES', 'BYTE';
2825: 307 TIndexOption', '(ixPrimary, ixUnique, ixDescending, ixCaseInsensitive, ixExpression, ixNonMaintained;
2826: 308 TIndexOptions', 'set of TIndexOption';
2827: 309 TINDEXOPTIONS', 'BYTE';
2828: 310 TDBScreenCursor', '(dcrDefault, dcrHourGlass, dcrSQLWait, dcrOther)';
2829: 311 TParamType', '(ptUnknown, ptInput, ptOutput, ptInputOutput, ptResult)';
2830: 312 TParamTypes', 'set of TParamType';
2831: 313 TFieldKind', '(fkData, fkCalculated, fkLookup, fkInternalCalc, '
2832: 314 TFieldKinds', 'set of TFieldKind';
2833: 315 TFIELDKINDS', 'BYTE';
2834: 316 TProviderFlag', '(pfInUpdate, pfInWhere, pfInKey, pfHidden)';
2835: 317 TProviderFlags', 'set of TProviderFlag';
2836: 318 TPROVIDERFLAGS', 'BYTE';
2837: 319 TFIELDNOTIFYEVENT', 'Procedure (SENDER : TFIELD)';
2838: 320 TFIELDGETTEXTTEVENT', 'Procedure (SENDER : TFIELD; var TEXT : S'
2839: 321 TFIELDSETTEXTTEVENT', 'Procedure (SENDER : TFIELD; const TEXT : '
2840: 322 TAUTOREFRESHFLAG', '(ARNONE, ARAUTOINC, ARDEFAULT)';
2841: 323 ObjectFieldTypes', 'BYTE';
2842: 324 LARGEINT', 'INT64';
2843: 325 TBLOBTYPE', 'BYTE';

```

```

2844: 326 TBLOBDATA, 'STRING';
2845: 327 TBookmark, '___Pointer';
2846: 328 TBookmarkStr, 'string';
2847: 329 TDATAACTION, '( DAFAIL, DAABORT, DARETRY )';
2848: 330 TBLOBSTREAMMODE, '( BMREAD, BMWRITE, BMREADWRITE )';
2849: 331 TDATAOPERATION, 'Procedure';
2850: 332 TDATASETNOTIFYEVENT, 'Procedure ( DATASET : TDATASET )';
2851: 333 TDATASETTERROREVENT, 'Procedure ( DATASET : TDATASET; E : TObject; var ACTION : TDATAACTION )';
2852: 334 TFILTERRECORDEVENT, 'Procedure ( DATASET : TDATASET; var ACCEPT : BOOLEAN )';
2853: 334 ***** File 12: F:\maxbox3\mX397\maxbox3\source\UPSC_extctrls.pas
2854: 335 TShapeType, '(stRectangle,stSquare,stRoundRect,stRoundSquare,stEllipse,stCircle)';
2855: 336 TBevelStyle, '(bsLowered, bsRaised)';
2856: 337 TBevelShape, '(bsBox,bsFrame,bsTopLine,bsBottomLine,bsLeftLine,bsRightLine,bsSpacer)';
2857: 338 TPanelBevel, '(bvNone, bvLowered, bvRaised,bvSpace)';
2858: 339 TBevelWidth, 'Longint';
2859: 340 TBorderWidth, 'Longint';
2860: 341 TSectionEvent, 'procedure(Sender: TObject; ASection, AWidth: Integer)';
2861: 342 ***** File 13: F:\maxbox3\mX397\maxbox3\source\UPSC_forms.pas
2862: 342 TMsg, 'record hwnd: longword; wParam: longint; lParam: longint;
2863: 343 TMsg record hwnd:longword;message:longword; wParam:longint;lParam:longint;time:DWORD;pt:TPoint;end;
2864: 344 TObject, 'Exception';
2865: 345 TMessageEvent, 'procedure(var Msg: TMsg; var Handled: Boolean)';
2866: 346 TExceptionEvent, 'procedure (Sender: TObject; E: Exception)';
2867: 347 TIdleEvent, 'procedure (Sender: TObject; var Done: Boolean)';
2868: 348 TScrollBarKind, '(sbHorizontal, sbVertical)';
2869: 349 TScrollBarInc, 'SmallInt';
2870: 350 TFormBorderStyle, '(bsNone, bsSingle, bsSizeable, bsDialog, bsToolWindow, bsSizeToolWin)';
2871: 351 TBorderStyle, 'TFormBorderStyle';
2872: 352 TWindowState, '(wsNormal, wsMinimized, wsMaximized)';
2873: 353 TFormStyle, '(fsNormal, fsMDIChild, fsMDIForm, fsStayOnTop)';
2874: 354 TPosition, '(poDesigned, poDefault, poDefaultPosOnly, poDefaultSizeOnly, poScreenCenter,
poDesktopCenter, poMainFormCenter, poOwnerFormCenter)';
2875: 355 TPrintScale, '(poNone, poProportional, poPrintToFit)';
2876: 356 TCloseAction, '(caNone, caHide, caFree, caMinimize)';
2877: 357 TCloseEvent, 'procedure(Sender: TObject; var Action: TCloseAction)';
2878: 358 TCloseQueryEvent, 'procedure(Sender: TObject; var CanClose: Boolean)';
2879: 359 TBorderIcon, '(biSystemMenu, biMinimize, biMaximize, biHelp)';
2880: 360 TBorderIcons, 'set of TBorderIcon';
2881: 361 THELPCONTEXT, 'Longint';
2882: 362 TMonitorDefaultTo, '( mdNearest, mdNull, mdPrimary )';
2883: 363 THintInfo, 'record HintControl : TControl; HintWindowClass : TH
2884: 364 TCMHintShow, 'record Msg : Cardinal; Reserved : Integer; HintIn
2885: 365 TCMHintShowPause, 'record Msg : Cardinal; WasActive : Integer;
2886: 366 TPopupForm, 'record FormID: Integer; Form : TCustomForm; WasPopup: Boolean; end';
2887: 367 HMONITOR, 'Integer';
2888: 368 HCURSOR, 'LongWord';
2889: 369 HICON, 'LongWord';
2890: 370 TSettingChangeEvent, 'procedure (Sender: TObject; Flag: Integer; const Section: string; var Result:
Longint) of object';
2891: 371 TStatusPanelStyle, '( psText, psOwnerDraw )';
2892: 372 TStatusPanelBevel, '( pbNone, pbLowered, pbRaised )';
2893: 373 TStatusPanelClass, 'class of TStatusPanel';
2894: 374 TCustomDrawPanelEvent, 'Procedure ( StatusBar : TCustomStatusBa
2895: 375 TSBCreatePanelClassEvent, 'Procedure ( Sender : TCustomStatusBa
2896: 376 TDrawPanelEvent, 'Procedure ( StatusBar : TStatusBar; Panel : T
2897: 377 TFormClass, 'class of TForm';
2898: 377 ***** File 14: F:\maxbox3\mX397\maxbox3\source\UPSC_forms_.pas
2899: 378 TIdleEvent, 'procedure (Sender: TObject; var Done: Boolean)';
2900: 379 TScrollBarKind, '(sbHorizontal, sbVertical)';
2901: 380 TScrollBarInc, 'SmallInt';
2902: 381 TFormBorderStyle, '(bsNone, bsSingle, bsSizeable, bsDialog, bsToolWindow, bsSizeToolWin)';
2903: 382 TBorderStyle, 'TFormBorderStyle';
2904: 383 TWindowState, '(wsNormal, wsMinimized, wsMaximized)';
2905: 384 TFormStyle, '(fsNormal, fsMDIChild, fsMDIForm, fsStayOnTop)';
2906: 385 TPosition, '(poDesigned, poDefault, poDefaultPosOnly, poDefaultSizeOnly, poScreenCenter,
poDesktopCenter, poMainFormCenter, poOwnerFormCenter)';
2907: 386 TPrintScale, '(poNone, poProportional, poPrintToFit)';
2908: 387 TCloseAction, '(caNone, caHide, caFree, caMinimize)';
2909: 388 TCloseEvent, 'procedure(Sender: TObject; var Action: TCloseAction)';
2910: 389 TCloseQueryEvent, 'procedure(Sender: TObject; var CanClose: Boolean)';
2911: 390 TBorderIcon, '(biSystemMenu, biMinimize, biMaximize, biHelp)';
2912: 391 TBorderIcons, 'set of TBorderIcon';
2913: 392 THELPCONTEXT, 'Longint';
2914: 392 ***** File 15: F:\maxbox3\mX397\maxbox3\source\UPSC_graphics.pas
2915: 393 TPoint, 'record X, Y: LongInt; end;'; in UPSC_std!!
2916: 394 TPointArray, 'array of TPoint';
2917: 395 TPointArray, 'array of TPoint';
2918: 396 TIconRec, 'record Width : Byte; Height : Byte; Colors : Word; R
2919: 397 HMETAFILE, 'THandle';
2920: 398 HENHMETAFILE, 'THandle';
2921: 399 TResData, 'record Handle : THandle; end';
2922: 400 TFontPitch, '( fpDefault, fpVariable, fpFixed )';
2923: 401 TFontName, 'string';
2924: 402 TFontCharset, 'Integer';
2925: 403 TFontStyle, '( fsBold, fsItalic, fsUnderline, fsStrikeOut )';
2926: 404 TFontStyles, 'set of TFontStyle';
2927: 405 TFillStyle, '( fsSurface, fsBorder )';
2928: 406 TFillMode, '( fmAlternate, fmWinding )';
2929: 407 TCopyMode, 'Longint';

```



```

2930: 408 TCanvasStates', '( csHandleValid, csFontValid, csPenValid, csBrushValid )';
2931: 409 TCanvasState', 'set of TCanvasStates';
2932: 410 TCanvasOrientation', '( coLeftToRight, coRightToLeft )';
2933: 411 TTextFormats', '( tfBottom, tfCalcRect, tfCenter, tfEditControl, '
2934: 412 TTextFormat', 'set of TTextFormats';
2935: 413 TTextFormat', 'set of TTextFormats';
2936: 414 TPoint', 'record x: longint; y: longint; end';
2937: 415 TPointArray', 'array of TPoint';
2938: 416 TPointArray', 'array of TPoint';
2939: 417 TFontStyle', '(FSBOLD, FSITALIC, FSUNDERLINE, FSSTRIKEOUT)';
2940: 418 TFontStyles', 'set of TFontStyle';
2941: 419 TFontStylesBase', 'set of TFontStyle';
2942: 420 TFontPitch', '(fpDefault, fpVariable, fpFixed)';
2943: 421 TPenStyle', '(psSolid, psDash, psDot, psDashDot, psDashDotDot, psClear, psInsideFrame)';
2944: 422 TPenMode', '(pmBlack, pmWhite, pmNop, pmNot, pmCopy, pmNotCopy, pmMergePenNot, pmMaskPenNot,
    pmMergeNotPen, pmMaskNotPen, pmMerge, pmNotMerge, pmMask, pmNotMask, pmXor, pmNotXor)';
2945: 423 TBrushStyle', '(bsSolid, bsClear, bsHorizontal, bsVertical, bsFdiagonal, bsBdiagonal, bsCross,
    bsDiagCross)';
2946: 424 TColor', 'integer';
2947: 425 HPEN', 'LongWord';
2948: 426 TPointArray', 'array of TPoint';
2949: 427 TPenStyle', '( psSolid, psDash, psDot, psDashDot, psDashDotDot, '
2950: 428 TPenMode', '( pmBlack, pmWhite, pmNop, pmNot, pmCopy, pmNotCopy, '
2951: 429 TPenData', 'record Handle : HPen; Color : TColor; Width : Integer;
2952: 430 TBrushStyle', '( bsSolid, bsClear, bsHorizontal, bsVertical, bsF
2953: 431 TPoint', 'record X: LongInt; Y: LongInt; end';
2954: 432 Points', 'array of TPoint';
2955: 433 TBitmapHandleType', '( bmDIB, bmDDB )';
2956: 434 TPixelFormat', '( pfDevice, pf1bit, pf4bit, pf8bit, pf15bit, pf1
2957: 435 TTransparentMode', '( tmAuto, tmFixed )';
2958: 436 HBITMAP', 'LongWord'; //3.6
2959: 437 HPALETTE', 'LongWord';
2960: 438 HDC', 'LongWord';
2961: 439 HICON', 'LongWord'; // from sysutils
2962: 440 HWNID', 'LongWord';
2963: 441 HENHMETAFILE', 'LongWord';
2964: 442 HENHMETAFILE', 'THandle';
2965: 443 TFontCharSet', '0..255'; // from sysutils
2966: 444 Points', 'array of TPoint';
2967: 444 ***** File 16: F:\maxbox3\mX397\maxbox3\source\upSC_Grids.pas
2968: 445 TGetExtentsFunc', 'Function ( Index : Longint ) : Integer';
2969: 446 TGridAxisDrawInfo', 'record EffectiveLineWidth : Integer; FixedB
2970: 447 TGridState', '( gsNormal, gsSelecting, gsRowSizing, gsColSizing, '
2971: 448 TGridOption', '( goFixedVertLine, goFixedHorzLine, goVertLine, g
2972: 449 TGridOptions', 'set of TGridOption';
2973: 450 TGridDrawStates', '( gdSelected, gdFocused, gdFixed )';
2974: 451 TGridDrawState', 'set of TGridDrawStates';
2975: 452 TGridRect', 'TRect';
2976: 453 TGridCoord', 'TPoint';
2977: 454 TGridDrawState', '( gdSelected, gdFocused, gdFixed )';
2978: 455 TGridDrawState', 'set of ( gdSelected, gdFocused, gdFixed )';
2979: 456 TGridScrollDirection', 'set of ( sdLeft, sdRight, sdUp, sdDown )';
2980: 457 TGridCoord', 'record X : Longint; Y : Longint; end';
2981: 458 TEditStyle', '( esSimple, esEllipsis, esPickList )';
2982: 459 TSelectCellEvent', 'Procedure ( Sender : TObject; ACol, ARow : L
2983: 460 TDrawCellEvent', 'Procedure ( Sender : TObject; ACol, ARow : Lon
2984: 461 TGetEditEvent', 'Procedure ( Sender : TObject; ACol, ARow : Long
2985: 462 TSetEditEvent', 'Procedure ( Sender : TObject; ACol, ARow : Long
2986: 463 TMovedEvent', 'Procedure (Sender : TObject; FromIndex, ToIndex: Longint)';
2987: 464 TOnGetPickListItems', 'Procedure (ACol, ARow: Integer; Items: TStringList)';
2988: 464 ***** File 17: F:\maxbox3\mX397\maxbox3\source\upSC_std.pas
2989: 465 TComponentStateE', '(csLoading, csReading, csWriting, csDestroying, csAncestor,
    csUpdating, csFixups, csFreeNotification, csInline, csDesignInstance)';
2990: 466 TComponentState', 'set of TComponentStateE';
2991: 467 TRect', 'record Left, Top, Right, Bottom: Integer; end';
2992: 468 TPoint', 'record X, Y: LongInt; end'; //from upsc_controls !!
2993: 469 TRect2', 'record TopLeft, BottomRight: TPoint; end'; in sysutils
2994: 469 ***** File 18: F:\maxbox3\mX397\maxbox3\source\upSC_stdctrls.pas
2995: 470 TEOwnerDrawState', '(odSelected, odGrayed, odDisabled, odChecked, '
2996: 471 TOwnerDrawState', 'set of TEOwnerDrawState';
2997: 472 TEditCharCase', '(ecNormal, ecUpperCase, ecLowerCase)';
2998: 473 TScrollStyle', '(ssNone, ssHorizontal, ssVertical, ssBoth)';
2999: 474 TComboBoxStyle', '(csDropDown, csSimple, csDropDownList, csOwnerDrawFixed, csOwnerDrawVariable)';
3000: 475 TDrawItemEvent', 'procedure(Control: TWinControl; Index: Integer; Rect: TRect; State: Byte)';
3001: 476 TDrawItemEvent', 'procedure(Control: TWinControl; Index: Integer; Rect: TRect; State:
    TOwnerDrawState)';
3002: 477 TMeasureItemEvent', 'procedure(Control: TWinControl; Index: Integer; var Height: Integer)';
3003: 478 TCheckBoxState', '(cbUnchecked, cbChecked, cbGrayed)';
3004: 479 TListBoxStyle', '(lbStandard, lbOwnerDrawFixed, lbOwnerDrawVariable)';
3005: 480 TScrollCode', '(scLineUp, scLineDown, scPageUp, scPageDown, scPosition, scTrack, scTop, scBottom,
    scEndScroll)';
3006: 481 TScrollEvent', 'procedure(Sender: TObject; ScrollCode: TScrollCode; var ScrollPos: Integer)';
3007: 482 TTextLayout', '( tlTop, tlCenter, tlBottom )';
3008: 482 ***** File 19: F:\maxbox3\mX397\maxbox3\source\upSI_ActnList.pas
3009: 483 TContainedActionClass', 'class of TContainedAction';
3010: 484 TActionEvent', 'Procedure ( Action : TBasicAction; var Handled : '
3011: 485 TActionListState', '( asNormal, asSuspended, asSuspendedEnabled)';
3012: 486 THintEvent', 'Procedure ( var HintStr : string; var CanShow : Boolean)';
3013: 487 TActionLinkClass', 'class of TActionLink';

```

```

3014: 488 TEnumActionProc', 'Procedure ( const Category : string; ActionCl'
3015: 488 ***** File 20: F:\maxbox3\mX397\maxbox3\source\upsi_ADODB.pas
3016: 489 TWideStrings', 'TWideStrings');
3017: 490 TConnectMode', '( cmUnknown, cmRead, cmWrite, cmReadWrite, cmSha'
3018: 491 TConnectOption', '( coConnectUnspecified, coAsyncConnect )';
3019: 492 TCursorLocation', '( clUseServer, clUseClient )';
3020: 493 TCursorType', '( ctUnspecified, ctOpenForwardOnly, ctKeyset, ctD'
3021: 494 TEventStatus', '( esOK, esErrorsOccured, esCantDeny, esCancel, e'
3022: 495 TExecuteOption', '( eoAsyncExecute, eoAsyncFetch, eoAsyncFetchNo'
3023: 496 TExecuteOptions', 'set of TExecuteOption';
3024: 497 TIsolationLevel', '( ilUnspecified, ilChaos, ilReadUncommitted, '
3025: 498 TADOLockType', '( ltUnspecified, ltReadOnly, ltPessimistic, ltOp'
3026: 499 TObjectState', '( stClosed, stOpen, stConnecting, stExecuting, stFetching )';
3027: 500 TObjectStates', 'set of TObjectState';
3028: 501 TSchemaInfo', '( siAsserts, siCatalogs, siCharacterSets, siColla'
3029: 502 TXactAttribute', '( xaCommitRetaining, xaAbortRetaining )';
3030: 503 TXactAttributes', 'set of TXactAttribute';
3031: 504 TBeginTransCompleteEvent', 'Procedure ( Connection : TADOConnect'
3032: 505 TCommandType', '( cmdUnknown, cmdText, cmdTable, cmdStoredProc, '
3033: 506 TConnectErrorEvent', 'Procedure ( Connection : TADOConnection; c'
3034: 507 TDisconnectEvent', 'Procedure ( Connection : TADOConnection; var'
3035: 508 TExecuteCompleteEvent', 'Procedure ( Connection : TADOConnection'
3036: 509 TWillConnectEvent', 'Procedure ( Connection : TADOConnection; va'
3037: 510 TWillExecuteEvent', 'Procedure ( Connection : TADOConnection; va'
3038: 511 TInfoMessageEvent', 'Procedure ( Connection : TADOConnection; co'
3039: 512 TDataType', 'TFieldType');
3040: 513 TParameterAttribute', '( paSigned, paNullable, paLong )';
3041: 514 TParameterAttributes', 'set of TParameterAttribute';
3042: 515 TParameterDirection', '( pdUnknown, pdInput, pdOutput, pdInputOu'
3043: 516 TPropList', 'array of PPropInfo';
3044: 517 TCursorOption', '( coHoldRecords, coMovePrevious, coAddNew, coDe'
3045: 518 TCursorOptions', 'set of TCursorOption';
3046: 519 TEventReason', '( erAddNew, erDelete, erUpdate, erUndoUpdate, er'
3047: 520 TFilterGroup', '( fgUnassigned, fgNone, fgPendingRecords, fgAffe'
3048: 521 TMarshalOption', '( moMarshalAll, moMarshalModifiedOnly )';
3049: 522 TRecordStatus', '( rsOK, rsNew, rsModified, rsDeleted, rsUnmodif'
3050: 523 TRecordStatusSet', 'set of TRecordStatus';
3051: 524 TAffectRecords', '( arCurrent, arFiltered, arAll, arAllChapters )';
3052: 525 TPersistFormat', '( pfADTG, pfXML )';
3053: 526 TSeekOption', '( soFirstEQ, soLastEQ, soAfterEQ, soAfter, soBefo'
3054: 527 PVariantList', 'TVariantList // will not work';
3055: 528 TWillChangeFieldEvent', 'Procedure ( DataSet : TCustomADODataset'
3056: 529 TFieldChangeCompleteEvent', 'Procedure ( DataSet : TCustomADODat'
3057: 530 TWillChangeRecordEvent', 'Procedure ( DataSet : TCustomADODataset'
3058: 531 TRecordChangeCompleteEvent', 'Procedure ( DataSet : TCustomADODa'
3059: 532 TEndOfRecordsetEvent', 'Procedure ( DataSet : TCustomADODataset;'
3060: 533 TFetchProgressEvent', 'Procedure ( DataSet : TCustomADODataset;'
3061: 534 TRecordsetErrorEvent', 'Procedure ( DataSet : TCustomADODataset;'
3062: 535 TRecordsetReasonEvent', 'Procedure ( DataSet : TCustomADODataset'
3063: 536 TRecordsetEvent', 'Procedure ( DataSet : TCustomADODataset; cons'
3064: 537 TRecordsetCreate', 'Procedure ( DataSet : TCustomADODataset; con'
3065: 537 ***** File 21: F:\maxbox3\mX397\maxbox3\source\upsi_AsyncCalls.pas
3066: 538 INT_PTR', 'Integer';
3067: 539 IInterface', 'IUnknown'; bug fix in 3.9.7.1
3068: 540 INT_PTR', 'Integer';
3069: 541 TAsyncIdleMsgMethod', 'Procedure';
3070: 542 TDeclFunc', '__Pointer';
3071: 543 TDeclMethod', 'TMethod';
3072: 544 TLocalAsyncProc', 'function(Param: INT_PTR): INT_PTR';
3073: 545 TLocalVclProc', 'function(Param: INT_PTR): INT_PTR';
3074: 546 TDeclMethod', 'TMethod';
3075: 547 TAsyncCallArgFunction', 'Function( Integer );
3076: 548 TAsyncCallArgFunction', 'Integer'; //tester
3077: 549 TAsyncCallArgObjectMethod', 'Function ( Arg : TObject ) : Integer';
3078: 550 TAsyncCallArgIntegerMethod', 'Function ( Arg : Integer ) : Integer';
3079: 551 TAsyncCallArgStringMethod', 'Function ( const Arg : string ) : Integer';
3080: 552 TAsyncCallArgWideStringMethod', 'Function ( const Arg : WideString ) : Integer';
3081: 553 TAsyncCallArgInterfaceMethod', 'Function ( const Arg : IInterface ) : Integer';
3082: 554 TAsyncCallArgExtendedMethod', 'Function ( const Arg : Extended ) : Integer';
3083: 555 TAsyncCallArgVariantMethod', 'Function ( const Arg : Variant ) : Integer';
3084: 556 TAsyncCallArgObjectEvent', 'Procedure ( Arg : TObject );
3085: 557 TAsyncCallArgIntegerEvent', 'Procedure ( Arg : Integer );
3086: 558 TAsyncCallArgStringEvent', 'Procedure ( const Arg : string );
3087: 559 TAsyncCallArgWideStringEvent', 'Procedure ( const Arg : WideString );
3088: 560 TAsyncCallArgInterfaceEvent', 'Procedure ( const Arg : IInterface );
3089: 561 TAsyncCallArgExtendedEvent', 'Procedure ( const Arg : Extended );
3090: 562 TAsyncCallArgVariantEvent', 'Procedure ( const Arg : Variant );
3091: 563 TAsyncCallArgObjectMethod', 'function(Arg: TObject): Integer of object';
3092: 564 TAsyncCallArgIntegerProc', 'function(Arg: Integer): Integer';
3093: 565 TAsyncCallArgStringProc', 'function(const Arg: string): Integer';
3094: 566 TAsyncCallArgStringProc', 'function(const Arg: WideString): Integer';
3095: 567 TAsyncCallArgInterfaceProc', 'function(const Arg: IInterface): Integer';
3096: 568 TAsyncCallArgInterfaceProc', 'function(const Arg: IInterface): Integer';
3097: 569 TAsyncCallArgExtendedProc', 'function(const Arg: Extended): Integer';
3098: 570 TAsyncCallArgVariantProc', 'function(const Arg: Variant): Integer';
3099: 571 TAsyncCallArgRecordMethod', 'Function ( var Arg ) : Integer';
3100: 572 TAsyncCallArgRecordEvent', 'Procedure ( var Arg );
3101: 572 ***** File 22: F:\maxbox3\mX397\maxbox3\source\upsi_BarCodeScanner.pas
3102: 573 TBarcodeEvent', 'Procedure ( var Barcode : String );

```

```

3103: 573 ***** File 23: F:\maxbox3\mX397\maxbox3\source\upsi_BoldUtils.pas
3104: 574 TBoldNotificationEvent', 'Procedure ( AComponent : TComponent; O'
3105: 574 ***** File 24: F:\maxbox3\mX397\maxbox3\source\upsi_Calendar.pas
3106: 575 TDayOfWeek', 'Integer');
3107: 575 ***** File 25: F:\maxbox3\mX397\maxbox3\source\upsi_Classes_draft.pas
3108: 576 TSeekOrigin', '( soBeginning, soCurrent, soEnd)');
3109: 633 TWndMethod', 'Procedure ( var Message : TMessage)';
3110: 633 ***** File 26: F:\maxbox3\mX397\maxbox3\source\upsi_ComCtrls.pas
3111: 634 THitTest', '( htAbove, htBelow, htNowhere, htOnItem, htOnButton, '
3112: 635 THitTests', 'set of THitTest');
3113: 636 TTabChangingEvent', 'Procedure ( Sender : TObject; var AllowChange: Boolean)';
3114: 637 TTabPosition', '( tpTop, tpBottom, tpLeft, tpRight)');
3115: 638 TTabStyle', '( tsTabs, tsButtons, tsFlatButtons)');
3116: 639 TDrawTabEvent', 'Procedure ( Control : TCustomTabControl; TabInd'
3117: 640 TTabGetImageEvent', 'Procedure ( Sender : TObject; TabIndex : In'
3118: 641 TWidth', 'Integer');
3119: 642 TStatusPanelStyle', '( psText, psOwnerDraw)');
3120: 643 TStatusPanelBevel', '( pbNone, pbLowered, pbRaised)');
3121: 644 TStatusPanelClass', 'class of TStatusPanel';
3122: 645 TCustomDrawPanelEvent', 'Procedure ( StatusBar : TCustomStatusBa'
3123: 646 TSBCreatePanelClassEvent', 'Procedure ( Sender : TCustomStatusBa'
3124: 647 TDrawPanelEvent', 'Procedure ( StatusBar : TStatusBar; Panel : T'
3125: 648 TCustomDrawTarget', '( dtControl, dtItem, dtSubItem)');
3126: 649 TCustomDrawStage', '( cdPrePaint, cdPostPaint, cdPreErase, cdPostErase)');
3127: 650 TCustomDrawState', 'set of ( cdsSelected, cdsGrayed, cdsDisabled'
3128: 651 THeaderSectionStyle', '( hsText, hsOwnerDraw)');
3129: 652 THeaderSectionClass', 'class of THeaderSection';
3130: 653 TSectionTrackState', '( tsTrackBegin, tsTrackMove, tsTrackEnd)');
3131: 654 TCustomDrawSectionEvent', 'Procedure ( HeaderControl : TCustomHe'
3132: 655 TCustomSectionNotifyEvent', 'Procedure ( HeaderControl : TCustom'
3133: 656 TCustomSectionTrackEvent', 'Procedure ( HeaderControl : TCustomH'
3134: 657 TSectionDragEvent', 'Procedure ( Sender : TObject; FromSection, '
3135: 658 TCustomHCCreateSectionClassEvent', 'Procedure ( Sender : TCustom'
3136: 659 THeaderStyle', '( hsButtons, hsFlat)');
3137: 660 TDrawSectionEvent', 'Procedure ( HeaderControl : THeaderControl; '
3138: 661 TSectionNotifyEvent', 'Procedure ( HeaderControl : THeaderContro'
3139: 662 TSectionTrackEvent', 'Procedure ( HeaderControl : THeaderControl'
3140: 663 THCCreateSectionClassEvent', 'Procedure ( Sender : THeaderContro'
3141: 664 TNodeState', '( nsCut, nsDropHilited, nsFocused, nsSelected, nsExpanded)');
3142: 665 TNodeAttachMode', '( naAdd, naAddFirst, naAddChild, naAddChildFirst, naInsert)');
3143: 666 TAddMode', '( taAddFirst, taAdd, taInsert)');
3144: 667 TTreeNodeClass', 'class of TTreeNode';
3145: 668 PNodeCache', '^TNodeCache // will not work';
3146: 669 TNodeCache', 'record CacheNode : TTreeNode; CacheIndex : Integer; end';
3147: 670 TSortType', '( stNone, stData, stText, stBoth)');
3148: 671 TMultiSelectStyles', '( msControlSelect, msShiftSelect, msVisibl'
3149: 672 TMultiSelectStyle', 'set of TMultiSelectStyles';
3150: 673 TTVChangingEvent', 'Procedure ( Sender : TObject; Node : TTreeNode'
3151: 674 TTVChangedEvent', 'Procedure ( Sender : TObject; Node : TTreeNode)';
3152: 675 TTVEditingEvent', 'Procedure ( Sender : TObject; Node : TTreeNode'
3153: 676 TTVEditedEvent', 'Procedure ( Sender : TObject; Node : TTreeNode; var S : string)';
3154: 677 TTVExpandingEvent', 'Procedure ( Sender : TObject; Node : TTreeNode'
3155: 678 TTVCollapsingEvent', 'Procedure ( Sender : TObject; Node : TTreeNode'
3156: 679 TTVExpandedEvent', 'Procedure ( Sender : TObject; Node : TTreeNode)';
3157: 680 TTVCompareEvent', 'Procedure ( Sender : TObject; Node1, Node2 : '
3158: 681 TTVCustomDrawEvent', 'Procedure ( Sender : TCustomTreeView; cons'
3159: 682 TTVCustomDrawItemEvent', 'Procedure ( Sender : TCustomTreeView; '
3160: 683 TTVAdvancedCustomDrawEvent', 'Procedure ( Sender : TCustomTreeVi'
3161: 684 TTVAdvancedCustomDrawItemEvent', 'Procedure ( Sender : TCustomTr'
3162: 685 TTVCreateNodeClassEvent', 'Procedure ( Sender : TCustomTreeView; '
3163: 686 TTrackBarOrientation', '( trHorizontal, trVertical)');
3164: 687 TTickMark', '( tmBottomRight, tmTopLeft, tmBoth)');
3165: 688 TTickStyle', '( tsNone, tsAuto, tsManual)');
3166: 689 TPositionToolTip', '( ptNone, ptTop, ptLeft, ptBottom, ptRight)';
3167: 690 TProgressRange', 'Integer';
3168: 691 TProgressBarOrientation', '( pbHorizontal, pbVertical)');
3169: 692 TAttributeType', '( atSelected, atDefaultText)');
3170: 693 TConsistentAttribute', '( caBold, caColor, caFace, caItalic, caS'
3171: 694 TConsistentAttributes', 'set of TConsistentAttribute';
3172: 695 TNumberingStyle', '( nsNone, nsBullet)');
3173: 696 TRichEditResizeEvent', 'Procedure ( Sender : TObject; Rect : TRect)';
3174: 697 TRichEditProtectChange', 'Procedure ( Sender : TObject; StartPos'
3175: 698 TRichEditSaveClipboard', 'Procedure ( Sender : TObject; NumObjec'
3176: 699 TSearchType', '( stWholeWord, stMatchCase)');
3177: 700 TSearchTypes', 'set of TSearchType';
3178: 701 TConversionClass', 'class of TConversion';
3179: 702 PConversionFormat', '^TConversionFormat // will not work';
3180: 703 TConversionFormat', 'record ConversionClass : TConversionClass; '
3181: 704 PRichEditStreamInfo', '^TRichEditStreamInfo // will not work';
3182: 705 TRichEditStreamInfo', 'record Converter : TConversion; Stream: TStream; end';
3183: 706 TUDAlignButton', '( udLeft, udRight)');
3184: 707 TUDOrientation', '( udHorizontal, udVertical)');
3185: 708 TUDBtnType', '( btNext, btPrev)');
3186: 709 TUPDownDirection', '( updNone, updUp, updDown)');
3187: 710 TUDClickEvent', 'Procedure ( Sender : TObject; Button : TUDBtnType)';
3188: 711 TUDChangingEvent', 'Procedure ( Sender : TObject; var AllowChange: Boolean)';
3189: 712 TUDChangingEventEx', 'Procedure ( Sender : TObject; var AllowCha'
3190: 713 THKModifier', '( hkShift, hkCtrl, hkAlt, hkExt)');
3191: 714 THKModifiers', 'set of THKModifier';

```

```

3192: 715 THKInvalidKey', '( hcNone, hcShift, hcCtrl, hcAlt, hcShiftCtrl, '
3193: 716 THKInvalidKeys', 'set of THKInvalidKey');
3194: 717 TDisplayCode', '( drBounds, drIcon, drLabel, drSelectBounds )';
3195: 718 TListItemClass', 'class of TListItem';
3196: 719 TIconArrangement', '( iaTop, iaLeft )';
3197: 720 TListArrangement', '( arAlignBottom, arAlignLeft, arAlignRight, '
3198: 721 TViewStyle', '( vsIcon, vsSmallIcon, vsList, vsReport )';
3199: 722 TItemState', '( isNone, isCut, isDropHilited, isFocused, isSelec
3200: 723 TItemStates', 'set of TItemState');
3201: 724 TItemChange', '( ctText, ctImage, ctState )';
3202: 725 TItemFind', '( ifData, ifPartialString, ifExactString, ifNearest);
3203: 726 TSearchDirection', '( sdLeft, sdRight, sdAbove, sdBelow, sdAll )';
3204: 727 TListHotTrackStyle', '( htHandPoint, htUnderlineCold, htUnderlineHot )';
3205: 728 TListHotTrackStyles', 'set of TListHotTrackStyle';
3206: 729 TItemRequests', '( irText, irImage, irParam, irState, irIndent )';
3207: 730 TItemRequest', 'set of TItemRequests';
3208: 731 TLVDeletedEvent', 'Procedure ( Sender : TObject; Item : TListItem)';
3209: 732 TLVEditingEvent', 'Procedure ( Sender : TObject; Item : TListIte
3210: 733 TLVEditedEvent', 'Procedure ( Sender : TObject; Item : TListItem; var S : string)';
3211: 734 TLVChangeEvent', 'Procedure ( Sender : TObject; Item : TListItem
3212: 735 TLVChangingEvent', 'Procedure ( Sender : TObject; Item : TListIt
3213: 736 TLVColumnClickEvent', 'Procedure ( Sender : TObject; Column : TListColumn)';
3214: 737 TLVColumnRClickEvent', 'Procedure ( Sender : TObject; Column : T
3215: 738 TLVCompareEvent', 'Procedure ( Sender : TObject; Item1, Item2 : '
3216: 739 TLVNotifyEvent', 'Procedure ( Sender : TObject; Item : TListItem)';
3217: 740 TLVSelectItemEvent', 'Procedure ( Sender : TObject; Item : TList
3218: 741 TLVDrawItemEvent', 'Procedure ( Sender : TCustomListView; Item : '
3219: 742 TLVCustomDrawEvent', 'Procedure ( Sender : TCustomListView; cons
3220: 743 TLVCustomDrawItemEvent', 'Procedure ( Sender : TCustomListView; '
3221: 744 TLVCustomDrawSubItemEvent', 'Procedure ( Sender : TCustomListVie
3222: 745 TLVAdvancedCustomDrawEvent', 'Procedure ( Sender : TCustomListVi
3223: 746 TLVAdvancedCustomDrawItemEvent', 'Procedure ( Sender : TCustomLi
3224: 747 TLVAdvancedCustomDrawSubItemEvent', 'Procedure ( Sender : TCusto
3225: 748 TLVOwnerDataEvent', 'Procedure ( Sender : TObject; Item : TListI
3226: 749 TLVOwnerDataFindEvent', 'Procedure ( Sender : TObject; Find : TI
3227: 750 TLVOwnerDataHintEvent', 'Procedure ( Sender : TObject; StartInde
3228: 751 TLVOwnerDataStateChangeEvent', 'Procedure ( Sender : TObject; St
3229: 752 TLVSubItemImageEvent', 'Procedure ( Sender : TObject; Item : TLI
3230: 753 TLVInfoTipEvent', 'Procedure ( Sender : TObject; Item : TListIte
3231: 754 TLVCreateItemClassEvent', 'Procedure ( Sender : TCustomListView;
3232: 755 TCommonAVI', '( aviNone, aviFindFolder, aviFindFile, aviFindComp
3233: 756 TToolButtonStyle', '( tbsButton, tbsCheck, tbsDropDown, tbsSepar
3234: 757 TToolButtonState', '( tbsChecked, tbsPressed, tbsEnabled, tbsHid
3235: 758 TToolButtonActionLinkClass', 'class of TToolButtonActionLink';
3236: 759 TTBCustomDrawFlags', 'set of ( tbNoEdges, tbHiliteHotTrack, tbNo
3237: 760 TTBGradientDrawingOptions', 'set of ( gdoHotTrack, gdoGradient )';
3238: 761 TTBDrawingStyle', '( dsNormal, dsGradient )';
3239: 762 TTBCustomDrawEvent', 'Procedure ( Sender : TToolBar; const ARect
3240: 763 TTBCustomDrawBtnEvent', 'Procedure ( Sender : TToolBar; Button : '
3241: 764 TTBAdvancedCustomDrawEvent', 'Procedure ( Sender : TToolBar; con
3242: 765 TTBAdvancedCustomDrawBtnEvent', 'Procedure ( Sender : TToolBar; '
3243: 766 TTBCustomizeQueryEvent', 'Procedure ( Sender : TToolBar; Index : '
3244: 767 TTBNewButtonEvent', 'Procedure ( Sender : TToolBar; Index : Inte
3245: 768 TTBButtonEvent', 'Procedure ( Sender : TToolBar; Button : TToolButton)';
3246: 769 TToolBandMaximize', '( bmNone, bmClick, bmDbClick )';
3247: 770 TCalDayOfWeek', '( dowMonday, dowTuesday, dowWednesday, dowThurs
3248: 771 TOnGetMonthInfoEvent', 'Procedure ( Sender : TObject; Month : Lo
3249: 772 TDateTimeKind', '( dtkDate, dtkTime )';
3250: 773 TDTDateMode', '( dmComboBox, dmUpDown )';
3251: 774 TDTDateFormat', '( dfShort, dfLong )';
3252: 775 TDTCalAlignment', '( dtaLeft, dtaRight )';
3253: 776 TDTParseInputEvent', 'Procedure ( Sender : TObject; const UserSt
3254: 777 TDateTimeColors', 'TMonthCalColors';
3255: 778 TPageScrollerOrientation', '( soHorizontal, soVertical )';
3256: 779 TPageScrollerButton', '( sbFirst, sbLast )';
3257: 780 TPageScrollerButtonState', '( bsNormal, bsInvisible, bsGrayed, b
3258: 781 TPageScrollEvent', 'Procedure ( Sender : TObject; Shift : TShift
3259: 782 TComboExItemsClass', 'class of TComboExItems';
3260: 783 TComboExItemClass', 'class of TComboExItem';
3261: 784 TComboBoxExStyle', '( csExDropDown, csExSimple, csExDropDownList)';
3262: 785 TComboBoxExStyleEx', '( csExCaseSensitive, csExNoEditImage, csEx
3263: 786 TComboBoxExStyles', 'set of TComboBoxExStyleEx';
3264: 787 TAutoCompleteOption', '( acoAutoSuggest, acoAutoAppend, acoSearc
3265: 788 TAutoCompleteOptions', 'set of TAutoCompleteOption';
3266: 789 ***** File 27: F:\maxbox3\mX397\maxbox3\source\upsi_Contrns.pas
3267: 789 TBucketItem', 'record Item : integer; Data : integer; end';
3268: 790 TBucketItemArray', 'array of TBucketItem';
3269: 791 TBucket', 'record Count : Integer; Items : TBucketItemArray; end';
3270: 792 TBucketArray', 'array of TBucket';
3271: 793 TBucketEvent', 'Procedure ( AItem, AData : Pointer; out AContinu
3272: 794 TBucketListSizes', '( bl2, bl4, bl8, bl16, bl32, bl64, bl128, bl256 )';
3273: 794 ***** File 28: F:\maxbox3\mX397\maxbox3\source\upsi_ConvUtils.pas
3274: 795 TConvFamily', 'Word';
3275: 796 TConvType', 'Word';
3276: 797 TConvTypeArray', 'array of TConvType';
3277: 798 TConvFamilyArray', 'array of TConvFamily';
3278: 799 TConvTypeList', 'array of TConvTypeInfo';
3279: 799 ***** File 29: F:\maxbox3\mX397\maxbox3\source\upsi_CPort.pas
3280: 800 TComExceptions', '( CE_OpenFailed, CE_WriteFailed, CE_ReadFailed

```



```

3281: 801 TPort', 'string');
3282: 802 TBaudRate', '( brCustom, br110, br300, br600, br1200, br2400, br'
3283: 803 TStopBits', '( sbOneStopBit, sbOne5StopBits, sbTwoStopBits )';
3284: 804 TDataBits', '( dbFive, dbSix, dbSeven, dbEight )';
3285: 805 TParityBits', '( prNone, prOdd, prEven, prMark, prSpace )';
3286: 806 TDTRFlowControl', '( dtrDisable, dtrEnable, dtrHandshake )';
3287: 807 TRTSFlowControl', '( rtsDisable, rtsEnable, rtsHandshake, rtsToggle )';
3288: 808 TFlowControl', '( fcHardware, fcSoftware, fcNone, fcCustom )';
3289: 809 TComEvent', '( evRxChar, evTxEmpty, evRxFlag, evRing, evBreak, e'
3290: 810 TComEvents', 'set of TComEvent';
3291: 811 TComSignal', '( csCTS, csDSR, csRing, csRLSD )';
3292: 812 TComSignals', 'set of TComSignal';
3293: 813 TComError', '( ceFrame, ceRxParity, ceOverrun, ceBreak, ceIO, ce'
3294: 814 TComErrors', 'set of TComError';
3295: 815 TSyncMethod', '( smThreadSync, smWindowSync, smNone )';
3296: 816 TStoreType', '( stRegistry, stIniFile )';
3297: 817 TStoredProp', '( spBasic, spFlowControl, spBuffer, spTimeouts, s'
3298: 818 TStoredProps', 'set of TStoredProp';
3299: 819 TComLinkEvent', '( leConn, leCTS, leDSR, leRLSD, leRing, leRx, l'
3300: 820 TRxCharEvent', 'Procedure ( Sender : TObject; Count : Integer)';
3301: 821 TRxBufEvent', 'Procedure ( Sender : TObject; const Buffer, Count : Integer)';
3302: 822 TComErrorEvent', 'Procedure ( Sender : TObject; Errors : TComErrors)';
3303: 823 TComSignalEvent', 'Procedure ( Sender : TObject; OnOff : Boolean)';
3304: 824 TComExceptionEvent', 'Procedure ( Sender : TObject; TComExceptio'
3305: 825 TOperationKind', '( okWrite, okRead )';
3306: 826 _OVERLAPPED', 'record Internal : DWORD; InternalHigh : DWORD; Of'
3307: 827 TOverlapped', '_OVERLAPPED';
3308: 828 TAsync', 'record Overlapped : TOverlapped; Kind : TOperationKind'
3309: 829 PAsync', 'TAsync // will not work';
3310: 830 UnicodeString', 'Widestring';
3311: 831 TComStrEvent', 'Procedure ( Sender : TObject; const Str : string)';
3312: 832 TCustPacketEvent', 'Procedure (Sender: TObject; const Str: string; var Pos : Integer)';
3313: 832 ***** File 30: F:\maxbox3\mX397\maxbox3\source\upsi_CPortCtl.pas
3314: 833 TComProperty', '( cpNone, cpPort, cpBaudRate, cpDataBits, cpStop'
3315: 834 TLedBitmap', 'TBitmap';
3316: 835 TLedKind', '( lkRedLight, lkGreenLight, lkBlueLight, lkYellowLig'
3317: 836 TComLedSignal', '( lsConn, lsCTS, lsDSR, lsRLSD, lsRing, lsRx, lsTx )';
3318: 837 TLedState', '( lsOff, lsOn )';
3319: 838 TLedStateEvent', 'Procedure ( Sender : TObject; AState : TLedState)';
3320: 839 TComTermChar', 'record Ch : Char; FrontColor : TColor; BackColor'
3321: 840 TTermEmulation', '( teVT100orANSI, teVT52, teNone )';
3322: 841 TTermCaret', '( tcBlock, tcUnderline, tcNone )';
3323: 842 TAdvanceCaret', '( acChar, acReturn, acLineFeed, acReverseLineFe'
3324: 843 TArrowKeys', '( akTerminal, akWindows )';
3325: 844 TTermAttributes', 'record FrontColor : TColor; BackColor : TColo'
3326: 845 TTermMode', 'record Keys : TArrowKeys; end';
3327: 846 TEscapeEvent', 'Procedure ( Sender : TObject; var EscapeCodes : '
3328: 847 TUnhandledEvent', 'Procedure ( Sender : TObject; Code : TEscapeC'
3329: 848 TStrRecvEvent', 'Procedure ( Sender : TObject; var Str : string)';
3330: 849 TChScreenEvent', 'Procedure ( Sender : TObject; Ch : Char)';
3331: 849 ***** File 31: F:\maxbox3\mX397\maxbox3\source\upsi_CPortEsc.pas
3332: 850 TEscapeResult', '( erChar, erCode, erNothing )';
3333: 851 TEscapeCode', '( ecUnknown, ecNotCompleted, ecCursorUp, ecCursor'
3334: 851 ***** File 32: F:\maxbox3\mX397\maxbox3\source\upsi_CPortMonitor.pas
3335: 852 TMonitorStyle', '( msAscii, msHex, msHexC, msHexVB, msHexPascal,'
3336: 853 TMonitorEvent', 'Procedure ( var DisplayValue : string; const Da'
3337: 854 TMonitorInfo', '( miCommPort, miDate, miTime, miDirection )';
3338: 855 TMonitorInfoSet', 'set of TMonitorInfo';
3339: 855 ***** File 33: F:\maxbox3\mX397\maxbox3\source\upsi_cutils.pas
3340: 856 pdevSearchResult', 'TdevSearchResult // will not work';
3341: 857 TdevSearchResult', 'record pt : TPoint; InFile : string; msg : string; end';
3342: 858 TUnitType', '( utSrc, utHead, utRes, utPrj, utOther )';
3343: 859 TExUnitType', '( utcSrc, utcppSrc, utcHead, utcppHead, utresHead, utresComp, utresSrc, utxPrj, utxOther )';
3344: 860 TFilterSet', '( ftOpen, ftHelp, ftPrj, ftSrc, ftAll )';
3345: 861 TErrFunc', 'Procedure ( Msg : String)';
3346: 862 TLineOutputFunc', 'Procedure ( Line : String)';
3347: 863 TCheckAbortFunc', 'Procedure ( var AbortThread : boolean)';
3348: 863 ***** File 34: F:\maxbox3\mX397\maxbox3\source\upsi_DateUtils.pas
3349: 864 TValueRelationship', '( -1, 0, 1)';
3350: 865 THexArray', 'array[0..15] of char';
3351: 865 ***** File 35: F:\maxbox3\mX397\maxbox3\source\upsi_DBCGrids.pas
3352: 866 TDBCtrlGridOrientation', '( goVertical, goHorizontal )';
3353: 867 TDBCtrlGridBorder', '( gbNone, gbRaised )';
3354: 868 TDBCtrlGridKey', '( gkNull, gkEditMode, gkPriorTab, gkNextTab, g'
3355: 869 TPaintPanelEvent', 'Procedure ( DBCtrlGrid : TDBCtrlGrid; Index : Integer)';
3356: 869 ***** File 36: F:\maxbox3\mX397\maxbox3\source\upsi_DBClient.pas
3357: 870 TGetUsernameEvent', 'Procedure ( Sender : TObject; var Username '
3358: 871 TAggUpdateEvent', 'Procedure ( Agg : TAggregate)';
3359: 872 TFieldDescList', 'array of DSFLDDesc';
3360: 873 TCDSRecInfo', 'TRecInfo';
3361: 874 PKeyBuffer', 'TKeyBuffer // will not work';
3362: 875 TKeyBuffer', 'record Modified : Boolean; Exclusive : Boolean; Fi'
3363: 876 TCDSKeyBuffer', 'TKeyBuffer';
3364: 877 TDataPacketFormat', '( dfBinary, dfXML, dfXMLUTF8 )';
3365: 878 TReconcileAction', '( raSkip, raAbort, raMerge, raCorrect, raCan'
3366: 879 TReconcileErrorEvent', 'Procedure ( DataSet : TCustomClientDataS'
3367: 880 TRemoteEvent', 'Procedure ( Sender : TObject; var OwnerData : Ol'
3368: 881 TReconcileInfo', 'record DataSet : TDataSet; UpdateKind : TUpdat'
3369: 882 TDataSetOption', '( doDisableInserts, doDisableDeletes, doDisabl'

```



```

3370: 883 TDataSetOptions', 'set of TDataSetOption');
3371: 884 TFetchOption', '( foRecord, foBlobs, foDetails )';
3372: 885 TFetchOptions', 'set of TFetchOption');
3373: 885 ***** File 37: F:\maxbox3\mX397\maxbox3\source\upsi_DBCommon.pas
3374: 886 TCANOperator', '( coNOTDEFINED, coISBLANK, coNOTBLANK, coEQ, coN'
3375: 887 NODEClass', '( nodeNULL, nodeUNARY, nodeBINARY, nodeCOMPARE, nod'
3376: 888 IDENTIFIEROption', '( idKeepQuotes, idMixCase, idMakeLowerCase, '
3377: 889 TExprData', 'array of Byte');
3378: 890 TParserOption', '( poExtSyntax, poAggregate, poDefaultExpr, poUs'
3379: 891 TParserOptions', 'set of TParserOption');
3380: 892 TExprNodeKind', '( enField, enConst, enOperator, enFunc )';
3381: 893 TExprScopeKind', '( skField, skAgg, skConst )';
3382: 894 PExprNode', 'TExprNode';
3383: 895 TExprNode', 'record FNext : PExprNode; FKind : TExprNodeKind; FP'
3384: 896 PExprNode', 'TExprNode';
3385: 897 TExprNode', 'record FNext : PExprNode; FKind : TExprNodeKind; FP'
3386: 898 TExprToken', '( etEnd, etSymbol, etName, etLiteral, etLParen, et'
3387: 899 TFieldInfo', 'record DatabaseName : Widestring; TableName : Wide'
3388: 900 TSQLToken', '( stUnknown, stTableName, stFieldName, stAscending, '
3389: 900 ***** File 38: F:\maxbox3\mX397\maxbox3\source\upsi_DBCommonTypes.pas
3390: 901 TLoginEvent', 'Procedure ( Sender : TObject; Username, Password '
3391: 902 FLDVchk', '( fldvNOCHECKS, fldvHASCHECKS, fldvUNKNOWN )';
3392: 903 FLDRights', '( fldrREADWRITE, fldrREADONLY, fldrNONE, fldrUNKNO'
3393: 904 pCBType', 'CBType // will not work';
3394: 905 CBType', '( cbGENERAL, cbRESERVED1, cbRESERVED2, cbINPUTREQ, cbr'
3395: 906 CBRType', '( cbrUSEDEF, cbrCONTINUE, cbrABORT, cbrCHKINPUT, cbrY'
3396: 907 TKeyIndex', '( kiLookup, kiRangeStart, kiRangeEnd, kiCurRangeSta'
3397: 907 ***** File 39: F:\maxbox3\mX397\maxbox3\source\upsi_DBCtrls.pas
3398: 908 TNavGlyph', '( ngEnabled, ngDisabled )';
3399: 909 TNavigateBtn', '( nbFirst, nbPrior, nbNext, nbLast, nbInsert, nb'
3400: 910 TNavButtonSet', 'set of TNavigateBtn';
3401: 911 TNavButtonStyle', 'set of ( nsAllowTimer, nsFocusRect )';
3402: 912 TNavButtonStyle', '( nsAllowTimer, nsFocusRect )';
3403: 913 ENavClick', 'Procedure (Sender : TObject; Button : TNavigateBtn);
3404: 914 TDropDownAlign', '( daLeft, daRight, daCenter )';
3405: 914 ***** File 40: F:\maxbox3\mX397\maxbox3\source\upsi_DBGrids.pas
3406: 915 TColumnValue', '( cvColor, cvWidth, cvFont, cvAlignment, cvReadO'
3407: 916 TBookmarkStr', 'string';
3408: 917 TColumnValues', 'set of TColumnValue';
3409: 918 TColumnButtonStyle', '( cbsAuto, cbsEllipsis, cbsNone )';
3410: 919 TColumnClass', 'class of TColumn';
3411: 920 TDBGridColumnsState', '( csDefault, csCustomized )';
3412: 921 TDBGridOption', '( dgEditing, dgAlwaysShowEditor, dgTitles, dgIn'
3413: 922 TDBGridOptions', 'set of TDBGridOption';
3414: 923 TDrawDataCellEvent', 'Procedure ( Sender : TObject; const Rect : '
3415: 924 TDrawColumnCellEvent', 'Procedure ( Sender : TObject; const Rect '
3416: 925 TDBGridClickEvent', 'Procedure ( Column : TColumn);
3417: 925 ***** File 41: F:\maxbox3\mX397\maxbox3\source\upsi_DBPlatform.pas
3418: 926 TRecordBuffer', 'IntPtr';
3419: 927 TValueBuffer', 'IntPtr';
3420: 928 TPSResult', 'TObject';
3421: 929 TRecordBuffer', 'PChar';
3422: 930 TValueBuffer', 'Pointer';
3423: 931 TPSResult', 'Pointer';
3424: 931 ***** File 42: F:\maxbox3\mX397\maxbox3\source\upsi_DBTables.pas
3425: 932 THandle', 'LongWord';
3426: 933 TFileName', 'string';
3427: 934 HDBIDB', 'longword';
3428: 935 TLocale', '___Pointer';
3429: 936 TBDECallbackEvent', 'Function ( CBInfo : Pointer ) : CBRType';
3430: 937 TConfigModes', '( cfmVirtual, cfmPersistent, cfmSession )';
3431: 938 TConfigMode', 'set of TConfigModes';
3432: 939 TPasswordEvent', 'Procedure ( Sender : TObject; var Continue : B'
3433: 940 TDatabaseEvent', '( dbOpen, dbClose, dbAdd, dbRemove, dbAddAlias'
3434: 941 TDatabaseNotifyEvent', 'Procedure ( DBEvent : TDatabaseEvent; co'
3435: 942 TTraceFlag', '( tfQPrepare, tfQExecute, tfError, tfStmt, tfConne'
3436: 943 TTraceFlags', 'set of TTraceFlag';
3437: 944 TFieldDescList', 'array of BDEFLDDesc';
3438: 945 TTransIsolation', '( tiDirtyRead, tiReadCommitted, tiRepeatableR'
3439: 946 TDatabaseLoginEvent', 'Procedure ( Database : TDatabase; LoginPa'
3440: 947 TRecNoStatus', '( rnDbase, rnParadox, rnNotSupported )';
3441: 948 TBlobDataArray', 'array of TBlobData';
3442: 949 TDBFlags', 'set of Integer';
3443: 950 TBatchMode', '( batAppend, batUpdate, batAppendUpdate, batDelete'
3444: 951 TTableType', '( ttDefault, ttParadox, ttDBase, ttFoxPro, ttASCII )';
3445: 952 TLockType', '( ltReadLock, ltWriteLock )';
3446: 953 TIndexName', 'string';
3447: 954 TIndexDescList', 'array of IDXDesc';
3448: 955 TValCheckList', 'array of VCHKDesc';
3449: 956 TParamBindMode', '( pbByName, pbByNumber )';
3450: 957 TServerDescList', 'array of TServerDesc';
3451: 958 TSPParamDescList', 'array of BDESPParamDesc';
3452: 958 ***** File 43: F:\maxbox3\mX397\maxbox3\source\upsi_DBWeb.pas
3453: 959 THTMLTableColumnClass', 'class of THTMLTableColumn';
3454: 960 THTMLColumnState', '( csDefault, csCustom )';
3455: 961 THTMLCaptionAlignment', '( caDefault, caTop, caBottom )';
3456: 962 TCreateContentEvent', 'Procedure ( Sender : TObject; var Contin'
3457: 963 THTMLGetTableCaptionEvent', 'Procedure ( Sender : TObject; var C'
3458: 964 THTMLFormatCellEvent', 'Procedure ( Sender : TObject; CellRow, C'

```

```

3459: 965 THTMLDataSetEmpty', 'Procedure ( Sender : TObject; var Continue '
3460: 965 ***** File 44: F:\maxbox3\mX397\maxbox3\source\upsi_DBX_connect.pas
3461: 966 TFieldType', '( ftUnknown, ftString, ftSmallint, ftInteger, ftWo'
3462: 967 TDataSetState', '( dsInactive, dsBrowse, dsEdit, dsInsert, dsSet'
3463: 968 TDataEvent', '( deFieldChange, deRecordChange, deDataSetChange, '
3464: 969 TUpdateStatus', '( usUnmodified, usModified, usInserted, usDelet'
3465: 970 TUpdateStatusSet', 'set of TUpdateStatus');
3466: 971 TUpdateAction', '( uaFail, uaAbort, uaSkip, uaRetry, uaApplied ););
3467: 972 TUpdateRecordTypes', 'set of ( rtModified, rtInserted, rtDeleted'
3468: 973 TUpdateMode', '( upWhereAll, upWhereChanged, upWhereKeyOnly ););
3469: 974 TUpdateKind', '( ukModify, ukInsert, ukDelete ););
3470: 975 TUpdateErrorEvent', 'Procedure ( DataSet : TDataSet; E : EDataab'
3471: 976 TUpdateRecordEvent', 'Procedure ( DataSet : TDataSet; UpdateKind'
3472: 977 TConnectChangeEvent', 'Procedure ( Sender : TObject; Connecting '
3473: 977 ***** File 45: F:\maxbox3\mX397\maxbox3\source\upsi_Dialogs.pas
3474: 978 TOpenOption', '( ofReadOnly, ofOverwritePrompt, ofHideReadOnly, '
3475: 979 TOpenOptions', 'set of TOpenOption');
3476: 980 TOpenOptionEx', '( ofExNoPlacesBar ););
3477: 981 TOpenOptionsEx', 'set of TOpenOptionEx');
3478: 982 TFileEditStyle', '( fsEdit, fsComboBox ););
3479: 983 TColorDialogOption', '( cdFullOpen, cdPreventFullOpen, cdShowHel'
3480: 984 TColorDialogOptions', 'set of TColorDialogOption');
3481: 985 TFontDialogOption', '( fdAnsiOnly, fdTrueTypeOnly, fdEffects, fd'
3482: 986 TFontDialogOptions', 'set of TFontDialogOption');
3483: 987 TFontDialogDevice', '( fdScreen, fdPrinter, fdBoth ););
3484: 988 TFDApplyEvent', 'Procedure ( Sender : TObject; Wnd : HWND););
3485: 989 TPrintRange', '( prAllPages, prSelection, prPageNums ););
3486: 990 TPrintDialogOption', '( poPrintToFile, poPageNums, poSelection, '
3487: 991 TPrintDialogOptions', 'set of TPrintDialogOption');
3488: 992 TPageSetupDialogOption', '( psoDefaultMinMargins, psoDisableMarg'
3489: 993 TPageSetupDialogOptions', 'set of TPageSetupDialogOption');
3490: 994 TPrinterKind', '( pkDotMatrix, pkHPPCL ););
3491: 995 TPageType', '( ptEnvelope, ptPaper ););
3492: 996 TPageSetupBeforePaintEvent', 'Procedure ( Sender : TObject; cons'
3493: 997 TPageMeasureUnits', '( pmDefault, pmMillimeters, pmInches ););
3494: 998 TPaintPageEvent', 'Procedure ( Sender : TObject; Canvas : TCanva'
3495: 999 TFindOption', '( frDown, frFindNext, frHideMatchCase, frHideWhol'
3496: 1000 TFindOptions', 'set of TFindOption');
3497: 1001 TFileDialogOption', '( fdoOverwritePrompt, fdoStrictFileTypes, f'
3498: 1002 TFileDialogOptions', 'set of TFileDialogOption');
3499: 1003 TTaskDialogFlag', '( tfEnableHyperlinks, tfUseHiconMain, tfUseHi'
3500: 1004 TTaskDialogFlags', 'set of TTaskDialogFlag');
3501: 1005 TTaskDialogCommonButton', '( tcbOk, tcbYes, tcbNo, tcbCancel, tc'
3502: 1006 TTaskDialogCommonButtons', 'set of TTaskDialogCommonButton');
3503: 1007 TProgressBarState', '( pbsNormal, pbsError, pbsPaused ););
3504: 1008 TTaskDialogButtonList', 'array of TTaskDialogButton');
3505: 1009 TTaskDlgClickEvent', 'Procedure ( Sender : TObject; ModalResult '
3506: 1010 TTaskDlgTimerEvent', 'Procedure ( Sender : TObject; TickCount : '
3507: 1011 TMsgDlgBtn', '( mbYes, mbNo, mbOK, mbCancel, mbAbort, mbRetry, mbIgnore, mbAll, mbNoToAll, '+
3508: 1012 TMsgDlgType', '( mtWarning, mtError, mtInformation, mtConfirmation, mtCustom ););
3509: 1013 TMsgDlgBtn', '( mbYes, mbNo, mbOK, mbCancel, mbAbort, mbRetry, mbIgnore, mbAll, mbNoToAll, '+
3510: 1014 TMsgDlgButtons', 'set of TMsgDlgBtn');
3511: 1014 ***** File 46: F:\maxbox3\mX397\maxbox3\source\upsi_FileCtrl.pas
3512: 1015 TFileAttr', '( ftReadOnly, ftHidden, ftSystem, ftVolumeID, ftDir'
3513: 1016 TFileType', 'set of TFileAttr');
3514: 1017 TDriveType', '( dtUnknown, dtNoDrive, dtFloppy, dtFixed, dtNetwo'
3515: 1018 TTextCase', '( tcLowerCase, tcUpperCase ););
3516: 1019 TSelectDirOpt', '( sdAllowCreate, sdPerformCreate, sdPrompt ););
3517: 1020 TSelectDirOpts', 'set of TSelectDirOpt');
3518: 1021 TSelectDirExtOpt', '( sdNewFolder, sdShowEdit, sdShowShares, sdN'
3519: 1022 TSelectDirExtOpts', 'set of TSelectDirExtOpt');
3520: 1022 ***** File 47: F:\maxbox3\mX397\maxbox3\source\upsi_FMTBcd.pas
3521: 1023 PBcd', '^TBcd // will not work';
3522: 1024 TBcd', 'record Precision: Byte; SignSpecialPlaces: Byte; Fraction: array [0..31] of Byte; end';
3523: 1025 TBytes', 'array of Byte';
3524: 1025 ***** File 48: F:\maxbox3\mX397\maxbox3\source\upsi_GraphUtil.pas
3525: 1026 TIdentMapEntry', 'record Value: Integer; Name: String; end';
3526: 1027 TColorRef', 'DWORD';
3527: 1028 TColorArray', 'array of TIdentMapEntry';
3528: 1029 TScrollDirection', '( sdLeft, sdRight, sdUp, sdDown ););
3529: 1030 TArrowType', '( atSolid, atArrows ););
3530: 1031 TGradientDirection', '( gdHorizontal, gdVertical ););
3531: 1032 TColorArraySortType', '( stHue, stSaturation, stLuminance, stRed'
3532: 1032 ***** File 49: F:\maxbox3\mX397\maxbox3\source\upsi_HTTPApp.pas
3533: 1033 TWebExceptionEvent', 'Procedure ( Sender : TObject; E : Exceptio'
3534: 1034 TMethodType', '( mtAny, mtGet, mtPut, mtPost, mtHead ););
3535: 1035 TContentParserClass', 'class of TAbstractContentParser';
3536: 1036 THTTPMethodEvent', 'Procedure ( Sender : TObject; Request : TWeb'
3537: 1037 TWebModuleContext', 'TObject';
3538: 1038 TWebModuleCreateMode', '( crOnDemand, crAlways ););
3539: 1039 TWebModuleCacheMode', '( caCache, caDestroy ););
3540: 1040 TModuleAddedProc', 'Procedure ( AWebModule : TComponent););
3541: 1040 ***** File 50: F:\maxbox3\mX397\maxbox3\source\upsi_HTTPUtil.pas
3542: 1041 TStringDynArray', 'array of string';
3543: 1041 ***** File 51: F:\maxbox3\mX397\maxbox3\source\upsi_IdAuthentication.pas
3544: 1042 TIdAuthenticationSchemes', '( asBasic, asDigest, asNTLM, asUnkno'
3545: 1043 TIdAuthSchemeSet', 'set of TIdAuthenticationSchemes';
3546: 1044 TIdAuthWhatsNext', '( wnAskTheProgram, wnDoRequest, wnFail ););
3547: 1045 TIdAuthenticationClass', 'class of TIdAuthentication';

```

```

3548: 1046 TIdAuthenticationClass', 'class of TIdAuthentication');
3549: 1046 ***** File 52: F:\maxbox3\mX397\maxbox3\source\upsi_IdCustomHTTPServer.pas
3550: 1047 TIdHTTPServer', 'IdCustomHTTPServer');
3551: 1048 TOnSessionEndEvent', 'Procedure ( Sender : TIdHTTPSession)';
3552: 1049 TOnSessionEndEvent', 'Procedure ( Sender : TIdHTTPSession)';
3553: 1050 TOnSessionStartEvent', 'Procedure ( Sender : TIdHTTPSession)';
3554: 1051 TOnCreateSession', 'Procedure ( ASender : TIdPeerThread; var VHT'
3555: 1052 TOnCreatePostStream', 'Procedure ( ASender : TIdPeerThread; var '
3556: 1053 TIdHTTPGetEvent', 'Procedure ( AThread : TIdPeerThread; ARequest'
3557: 1054 TIdHTTPOtherEvent', 'Procedure ( Thread : TIdPeerThread; const a'
3558: 1055 TIdHTTPInvalidSessionEvent', 'Procedure ( Thread : TIdPeerThread'
3559: 1055 ***** File 53: F:\maxbox3\mX397\maxbox3\source\upsi_IdGlobal.pas
3560: 1056 TIdFileName', 'WideString');
3561: 1057 TIdFileName', 'String');
3562: 1058 TBytes', 'array of Byte');
3563: 1059 TIdBytes', 'TBytes');
3564: 1060 TIdEncoding', '( enDefault, enANSI, enUTF8 )';
3565: 1061 Short', 'Smallint');
3566: 1062 THandle', 'LongWord');
3567: 1063 THandle', 'Integer');
3568: 1064 TBytes', 'array of Byte');
3569: 1065 TIdIPv6Address', 'array [0..7] of word');
3570: 1066 TIdMaxLineAction', '( maException, maSplit )';
3571: 1067 TIdPort', 'Integer');
3572: 1068 TIdReadLnFunction', 'Function : string');
3573: 1069 TIdOSType', '( otUnknown, otLinux, otWindows )';
3574: 1070 THandle', 'LongWord');
3575: 1071 TIdThreadPriority', 'TThreadPriority');
3576: 1072 TIdMaxLineAction', '( maException, maSplit )';
3577: 1073 TIdReadLnFunction', 'Function : string');
3578: 1074 TIdReuseSocket', '( rsOSDependent, rsTrue, rsFalse )';
3579: 1075 TIdCharSet', '( csGB2312, csBig5, csIso2022jp, csEucKR, csIso88591 )';
3580: 1076 PByte', '^Byte // will not work';
3581: 1077 PWord', '^Word // will not work';
3582: 1078 TIdIPvVersion', '( Id_IPv4, Id_IPv6 )';
3583: 1079 TIdPID', 'Integer');
3584: 1080 TIdPID', 'LongWord');
3585: 1081 TIdWin32Type', '( Win32s, WindowsNT40PreSP6Workstation, WindowsN'
3586: 1082 TIdWin32Type', '( Win32s, WindowsNT40, Windows95, Windows95OSR2,'
3587: 1082 ***** File 54: F:\maxbox3\mX397\maxbox3\source\upsi_IdGlobal_10.pas
3588: 1083 TIdEncoding', '( enDefault, enANSI, enUTF8 )';
3589: 1084 TWaitResult', '( wrSignaled, wrTimeout, wrAbandoned, wrError )';
3590: 1085 Short', 'Smallint');
3591: 1086 PCardinal', '^Cardinal // will not work';
3592: 1087 TIdBytes', 'TBytes');
3593: 1088 TIdPort', 'Integer');
3594: 1089 TIdMaxLineAction', '( maException, maSplit )';
3595: 1090 TIdOSType', '( otUnknown, otLinux, otWindows, otDotNet )';
3596: 1091 TIdIPvVersion', '( Id_IPv4, Id_IPv6 )';
3597: 1092 TIdPID', 'LongWord');
3598: 1093 TIdThreadPriority', 'TThreadPriority');
3599: 1094 TIdPID', 'LongWord');
3600: 1095 TIdThreadPriority', 'TThreadPriority');
3601: 1096 THandle', 'LongWord');
3602: 1097 THandle', 'Integer');
3603: 1098 THandle', 'Integer');
3604: 1099 TIdReuseSocket', '( rsOSDependent, rsTrue, rsFalse )';
3605: 1099 ***** File 55: F:\maxbox3\mX397\maxbox3\source\upsi_IdHash.pas
3606: 1100 T4x4LongWordRecord', 'array [0..3] of LongWord';
3607: 1101 T5x4LongWordRecord', 'array [0..4] of LongWord';
3608: 1102 T4x4LongWordRecord', 'array [0..3] of LongWord';
3609: 1103 T5x4LongWordRecord', 'array [0..4] of LongWord';
3610: 1103 ***** File 56: F:\maxbox3\mX397\maxbox3\source\upsi_IdHashMessageDigest.pas
3611: 1104 T4x4LongWordRecord', 'array[0..3] of LongWord';
3612: 1105 T5x4LongWordRecord', 'array[0..4] of LongWord';
3613: 1106 T4x4LongWordRecord', 'array [0..3] of LongWord';
3614: 1107 T5x4LongWordRecord', 'array [0..4] of LongWord';
3615: 1107 ***** File 57: F:\maxbox3\mX397\maxbox3\source\upsi_IdIcmpClient.pas
3616: 1108 TReplyStatusTypes', '( rsEcho, rsError, rsTimeout, rsErrorUnreac'
3617: 1109 TReplyStatus', 'record BytesReceived : integer; FromIpAddress : '
3618: 1110 TOnReplyEvent', 'Procedure ( ASender : TComponent; const AReplyS'
3619: 1110 ***** File 58: F:\maxbox3\mX397\maxbox3\source\upsi_IdMailBox.pas
3620: 1111 TIdMailBoxState', '( msReadWrite, msReadOnly )';
3621: 1112 TIdMailBoxAttributes', '( maNoInferiors, maNoSelect, maMarked, m'
3622: 1113 TIdMailBoxAttributesSet', 'set of TIdMailBoxAttributes';
3623: 1114 TLongIntArray', 'array of LongInt';
3624: 1114 ***** File 59: F:\maxbox3\mX397\maxbox3\source\upsi_IdMessage.pas
3625: 1115 TIdMessagePriority', '( mpHighest, mpHigh, mpNormal, mpLow, mpLo'
3626: 1116 TOnGetMessagePartStream', 'Procedure ( AStream : TStream)';
3627: 1117 TIdMessageFlags', '( mfAnswered, mfFlagged, mfDeleted, mfDraft, '
3628: 1118 TIdMessageFlagsSet', 'set of TIdMessageFlags';
3629: 1119 TIdMessagePartClass', 'class of TIdMessagePart';
3630: 1120 TIdMessageEncoding', '( meMIME, meUU )';
3631: 1121 TIdInitializeIsoEvent', 'Procedure ( var VTransferHeader : TTran'
3632: 1122 TIdMessageEvent', 'Procedure ( ASender : TComponent; var AMsg : '
3633: 1123 TIdStringMessageEvent', 'Procedure ( ASender : TComponent; const'
3634: 1123 ***** File 60: F:\maxbox3\mX397\maxbox3\source\upsi_IdNetworkCalculator.pas
3635: 1124 TNetworkClass', '( ID_NET_CLASS_A, ID_NET_CLASS_B, ID_NET_CLASS_'
3636: 1125 TIdIPAddressType', '( IPLocalHost, IPLocalNetwork, IPReserved, I'

```

```

3637: 1125 ***** File 61: F:\maxbox3\mX397\maxbox3\source\upsi_IdSMTP.pas
3638: 1126 TAuthenticationType, '( atNone, atLogin )';
3639: 1126 ***** File 62: F:\maxbox3\mX397\maxbox3\source\upsi_IdSNTP.pas
3640: 1127 TNPtGram, 'record Head1 : byte; Head2 : byte; Head3 : byte; Hea
3641: 1128 TTr', 'record L1 : byte; L2 : byte; L3 : byte; L4 : byte; end';
3642: 1128 ***** File 63: F:\maxbox3\mX397\maxbox3\source\upsi_IdSSLOpenSSL.pas
3643: 1129 TIdSSLVersion, '( sslvSSLv2, sslvSSLv23, sslvSSLv3, sslvTLSv1 )';
3644: 1130 TIdSSLMode, '( sslmUnassigned, sslmClient, sslmServer, sslmBoth)';
3645: 1131 TIdSSLVerifyMode, '( sslvrfPeer, sslvrfFailIfNoPeerCert, sslvrf
3646: 1132 TIdSSLVerifyModeSet, 'set of TIdSSLVerifyMode';
3647: 1133 TIdSSLCtxMode, '( sslCtxClient, sslCtxServer )';
3648: 1134 TIdSSLAction, '( sslRead, sslWrite )';
3649: 1135 TByteArrayR, 'record Length : Integer; Data : PChar; end';
3650: 1136 TCallbackEvent, 'Procedure ( Msg : String)';
3651: 1137 TPasswordEvent, 'Procedure ( var Password : String)';
3652: 1138 TVerifyPeerEvent, 'Function ( Certificate : TIdX509 ) : Boolean';
3653: 1139 TIOHandlerNotify, 'Procedure ( ASender : TIdSSLIOHandlerSocket)';
3654: 1139 ***** File 64: F:\maxbox3\mX397\maxbox3\source\upsi_IdTCPConnection.pas
3655: 1140 TIdBufferBytesRemoved, 'Procedure ( ASender : TObject; const AB
3656: 1140 ***** File 65: F:\maxbox3\mX397\maxbox3\source\upsi_IdTCPServer.pas
3657: 1141 TIdAfterCommandHandlerEvent, 'Procedure ( ASender : TIdTCPServe
3658: 1142 TIdBeforeCommandHandlerEvent, 'Procedure ( ASender : TIdTCPServ
3659: 1143 TIdCommandEvent, 'Procedure ( ASender : TIdCommand)';
3660: 1144 TIdNoCommandHandlerEvent, 'Procedure ( ASender : TIdTCPServer;
3661: 1145 TIdListenExceptionEvent, 'Procedure ( AThread : TIdListenerThre
3662: 1146 TIdServerThreadExceptionEvent, 'Procedure ( AThread : TIdPeerTh
3663: 1147 TIdServerThreadEvent, 'Procedure ( AThread : TIdPeerThread)';
3664: 1147 ***** File 66: F:\maxbox3\mX397\maxbox3\source\upsi_IdTelnet.pas
3665: 1148 TIdTelnetState, '( tnsDATA, tnsIAC, tnsIAC_SB, tnsIAC_WILL, tns
3666: 1149 TIdTelnetCommand, '( tncNoLocalEcho, tncLocalEcho, tncEcho )';
3667: 1150 TIdTelnetDataAvailEvent, 'Procedure ( Sender : TIdTelnet; const
3668: 1151 TIdTelnetCommandEvent, 'Procedure ( Sender : TIdTelnet; Status
3669: 1151 ***** File 67: F:\maxbox3\mX397\maxbox3\source\upsi_IdThread.pas
3670: 1152 TIdThreadStopMode, '( smTerminate, smSuspend )';
3671: 1153 TIdExceptionThreadEvent, 'Procedure ( AThread : TIdThread; AExc
3672: 1154 TIdNotifyThreadEvent, 'Procedure ( AThread : TIdThread)';
3673: 1155 TIdSynchronizeThreadEvent, 'Procedure ( AThread : TIdThread; AD
3674: 1156 TIdThreadClass, 'class of TIdThread';
3675: 1156 ***** File 68: F:\maxbox3\mX397\maxbox3\source\upsi_IdTrivialFTP.pas
3676: 1157 TIdTFTPMode, '( tfNetAscii, tfOctet )';
3677: 1157 ***** File 69: F:\maxbox3\mX397\maxbox3\source\upsi_IdTrivialFTPBase.pas
3678: 1158 TIdTFTPMode, '( tfNetAscii, tfOctet )';
3679: 1159 WordStr, 'string';
3680: 1159 ***** File 70: F:\maxbox3\mX397\maxbox3\source\upsi_IdUserAccounts.pas
3681: 1160 TOnAfterAuthentication, 'Procedure ( const AUsername : String;
3682: 1160 ***** File 71: F:\maxbox3\mX397\maxbox3\source\upsi_IdImgList.pas
3683: 1161 TDrawingStyle, '( dsFocus, dsSelected, dsNormal, dsTransparent
3684: 1162 TImageType, '( itImage, itMask )';
3685: 1163 TResType, '( rtBitmap, rtCursor, rtIcon )';
3686: 1164 TOverlay, 'Integer';
3687: 1165 TLoadResource, '( lrDefaultColor, lrDefaultSize, lrFromFile, lr
3688: 1166 TLoadResources, 'set of TLoadResource';
3689: 1167 TImageIndex, 'Integer';
3690: 1167 ***** File 72: F:\maxbox3\mX397\maxbox3\source\upsi_IdMouse.pas
3691: 1168 TPanDirection, '( pdUp, pdDown, pdLeft, pdRight )';
3692: 1169 TPanDirections, 'set of TPanDirection';
3693: 1170 TPanOption, '( poVertical, poHorizontal )';
3694: 1171 TPanOptions, 'set of TPanOption';
3695: 1172 TUpdateEvent, 'Procedure ( Sender : TObject; var Delta : TPoint
3696: 1172 ***** File 73: F:\maxbox3\mX397\maxbox3\source\upsi_IdIniFiles.pas
3697: 1173 PPHashItem, '^PHashItem // will not work';
3698: 1174 PHashItem, '^THashItem // will not work';
3699: 1175 THashItem, 'record Next : PHashItem; Key : string; Value : Inte
3700: 1175 ***** File 74: F:\maxbox3\mX397\maxbox3\source\upsi_IdInterface2_so.pas
3701: 1176 IIncome, 'IIncomeInt';
3702: 1177 TIncome, 'TIncomeRealIntf';
3703: 1177 ***** File 75: F:\maxbox3\mX397\maxbox3\source\upsi_IdJclBase.pas
3704: 1178 Float, 'Extended';
3705: 1179 Float, 'Double';
3706: 1180 Float, 'Single';
3707: 1181 LongWord, 'Cardinal';
3708: 1182 TSysCharSet, 'set of Char';
3709: 1183 TDynByteArray, 'array of Byte';
3710: 1184 TDynShortintArray, 'array of Shortint';
3711: 1185 TDynSmallintArray, 'array of Smallint';
3712: 1186 TDynWordArray, 'array of Word';
3713: 1187 TDynIntegerArray, 'array of Integer';
3714: 1188 TDynLongintArray, 'array of Longint';
3715: 1189 TDynCardinalArray, 'array of Cardinal';
3716: 1190 TDynInt64Array, 'array of Int64';
3717: 1191 TDynExtendedArray, 'array of Extended';
3718: 1192 TDynDoubleArray, 'array of Double';
3719: 1193 TDynSingleArray, 'array of Single';
3720: 1194 TDynFloatArray, 'array of Float';
3721: 1195 TDynPointerArray, 'array of __Pointer';
3722: 1196 TDynStringArray, 'array of string';
3723: 1197 IInterface, 'IUnknown'; bugfix in 3.9.71
3724: 1197 ***** File 76: F:\maxbox3\mX397\maxbox3\source\upsi_IdJclConsole.pas
3725: 1198 TJclConsoleInputMode, '( imLine, imEcho, imProcessed, imWindow, imMouse )';

```



```

3726: 1199 TJclConsoleInputModes', 'set of TJclConsoleInputMode');
3727: 1200 TJclConsoleOutputMode', '( omProcessed, omWrapAtEol )');
3728: 1201 TJclConsoleOutputModes', 'set of TJclConsoleOutputMode');
3729: 1202 TJclScreenBufferBeforeResizeEvent', 'Procedure ( Sender : TObject;
3730: 1203 TJclScreenBufferAfterResizeEvent', 'Procedure ( Sender : TObject)');
3731: 1204 TJclScreenBufferTextHorizontalAlign', '(thaCurrent, thaLeft, thaCenter, thaRight)');
3732: 1205 TJclScreenBufferTextVerticalAlign', '( tvaCurrent, tvaTop, tvaCenter, tvaBottom)');
3733: 1206 TJclScreenFontColor', '( fclBlack, fclBlue, fclGreen, fclRed, fcl
3734: 1207 TJclScreenBackColor', '( bclBlack, bclBlue, bclGreen, bclRed, bcl
3735: 1208 TJclScreenFontStyle', '( fsLeadingByte, fsTrailingByte, fsGridHo
3736: 1209 TJclScreenFontStyles', 'set of TJclScreenFontStyle');
3737: 1210 TJclScreenCursorSize', 'Integer');
3738: 1211 TJclInputCtrlEvent', '( ceCtrlC, ceCtrlBreak, ceCtrlClose, ceCtr
3739: 1212 TJclInputRecordArray', 'array of TInputRecord');
3740: 1212 ***** File 77: F:\maxbox3\mX397\maxbox3\source\upsi_JclFileUtils.pas
3741: 1213 TCompactPath', '( cpCenter, cpEnd )');
3742: 1214 TFileListOption', '( flFullNames, flRecursive, flMaskedSubfolder'
3743: 1215 TFileListOptions', 'set of TFileListOption');
3744: 1216 TJclAttributeMatch', '( amAny, amExact, amSubSetOf, amSuperSetOf'
3745: 1217 TFileFlag', '( ffDebug, ffInfoInferred, ffPatched, ffPreRelease,
3746: 1218 TFileFlags', 'set of TFileFlag');
3747: 1219 PLangIdRec', 'TLangIdRec // will not work');
3748: 1220 TJclFileMappingRoundOffset', '( rvDown, rvUp )');
3749: 1221 TJclMappedTextReaderIndex', '( tiNoIndex, tiFull )');
3750: 1222 PPCharArray', 'TPCharArray // will not work');
3751: 1222 ***** File 78: F:\maxbox3\mX397\maxbox3\source\upsi_JclGraphUtils.pas
3752: 1223 PColor32', 'TColor32 // will not work');
3753: 1224 TColor32', 'Longword');
3754: 1225 PColor32Array', 'TColor32Array // will not work');
3755: 1226 PPalette32', 'TPalette32 // will not work');
3756: 1227 TArrayOfColor32', 'array of TColor32');
3757: 1228 TPointArray', 'array of TPoint');
3758: 1229 TPointArray', 'TPointArray // will not work');
3759: 1230 TClipCode', '( ccLeft, ccRight, ccAbove, ccBelow )');
3760: 1231 TClipCodes', 'set of TClipCode');
3761: 1232 TPalette32', 'array [0..255] of TColor32');
3762: 1233 PClipCodes', 'TClipCodes // will not work');
3763: 1233 ***** File 79: F:\maxbox3\mX397\maxbox3\source\upsi_JclLANMan.pas
3764: 1234 TNetUserFlag', '( ufAccountDisable, ufHomedirRequired, ufLockout'
3765: 1235 TNetUserFlags', 'set of TNetUserFlag');
3766: 1236 TNetUserInfoFlag', '( uifScript, uifTempDuplicateAccount, uifNor'
3767: 1237 TNetUserInfoFlags', 'set of TNetUserInfoFlag');
3768: 1238 TNetUserPriv', '( upUnknown, upGuest, upUser, upAdmin )');
3769: 1239 TNetUserAuthFlag', '( afOpPrint, afOpComm, afOpServer, afOpAccounts )');
3770: 1240 TNetUserAuthFlags', 'set of TNetUserAuthFlag');
3771: 1241 TNetWellKnownRID', '( wkrAdmins, wkrUsers, wkrGuests, wkrPowerUs
3772: 1241 ***** File 80: F:\maxbox3\mX397\maxbox3\source\upsi_JclLogic.pas
3773: 1242 TBitRange', 'Byte');
3774: 1243 TBooleanArray', 'array of Boolean');
3775: 1243 ***** File 81: F:\maxbox3\mX397\maxbox3\source\upsi_JclMath.pas
3776: 1244 TPrimalityTestMethod', '( ptTrialDivision, ptRabinMiller )');
3777: 1245 TFloatingPointClass', '( fpZero, fpNormal, fpDenormal, fpInfini
3778: 1246 TNaNTag', 'Integer');
3779: 1247 PPointerArray', 'TPointerArray // will not work');
3780: 1248 TDelphiSet', 'set of Byte');
3781: 1249 PDelphiSet', 'TDelphiSet // will not work');
3782: 1249 ***** File 82: F:\maxbox3\mX397\maxbox3\source\upsi_JclMath_.pas
3783: 1250 TPrimalityTestMethod', '( ptTrialDivision, ptRabinMiller )');
3784: 1251 TFloatingPointClass', '( fpZero, fpNormal, fpDenormal, fpInfini
3785: 1252 TNaNTag', 'Integer');
3786: 1253 PPointerArray', 'TPointerArray // will not work');
3787: 1254 TDelphiSet', 'set of Byte');
3788: 1255 PDelphiSet', 'TDelphiSet // will not work');
3789: 1255 ***** File 83: F:\maxbox3\mX397\maxbox3\source\upsi_JclMiscel.pas
3790: 1256 HKEY', 'LongWord');
3791: 1256 ***** File 84: F:\maxbox3\mX397\maxbox3\source\upsi_JclMultimedia.pas
3792: 1257 TMmTimerKind', '( tkOneShot, tkPeriodic )');
3793: 1258 TMmNotificationKind', '( nkCallback, nkSetEvent, nkPulseEvent )');
3794: 1259 MCIERROR', 'DWORD');
3795: 1260 TJclCdMediaInfo', '( miProduct, miIdentity, miUPC )');
3796: 1261 TJclCdTrackType', '( ttAudio, ttOther )');
3797: 1262 TJclCdTrackInfo', 'record Minute : Byte; Second : Byte; TrackTyp
3798: 1263 TJclCdTrackInfoArray', 'array of TJclCdTrackInfo');
3799: 1263 ***** File 85: F:\maxbox3\mX397\maxbox3\source\upsi_JclSimpleXml.pas
3800: 1264 TJclOnSimpleXMLParsed', 'Procedure ( Sender : TObject; const Name: string)');
3801: 1265 TJclOnValueParsed', 'Procedure ( Sender : TObject; const Name, Value: string)');
3802: 1266 TJclOnSimpleProgress', 'Procedure ( Sender : TObject; const Posi
3803: 1267 TJclHashKind', '( hkList, hkDirect )');
3804: 1268 PJclHashElem', 'TJclHashElem // will not work');
3805: 1269 TJclHashElem', 'record Next : PJclHashElem; Obj : TObject; end');
3806: 1270 PJclHashRecord', 'TJclHashRecord // will not work');
3807: 1271 PJclHashList', 'TJclHashList // will not work');
3808: 1272 TJclHashRecord', 'record Count : Byte; end');
3809: 1273 TJclSimpleXMLElemCompare', 'Function ( Elems : TJclSimpleXMLElem
3810: 1274 TJclSimpleXMLElemClass', 'class of TJclSimpleXMLElem);
3811: 1275 TJclSimpleXMLOptions', '( sxoAutoCreate, sxoAutoIndent, s
3812: 1276 TJclSimpleXMLEncodeEvent', 'Procedure ( Sender : TObject; var Value: string)');
3813: 1277 TJclSimpleXMLEncodeStreamEvent', 'Procedure ( Sender : TObject; InStream, OutStream : TStream)');
3814: 1277 ***** File 86: F:\maxbox3\mX397\maxbox3\source\upsi_JclSimpleXml_test.pas

```



```

3815: 1278 TJclOnSimpleXMLParsed', 'Procedure ( Sender : TObject; const Nam'
3816: 1279 TJclOnValueParsed', 'Procedure ( Sender : TObject; const Name, V'
3817: 1280 TJclOnSimpleProgress', 'Procedure ( Sender : TObject; const Posi'
3818: 1281 TJclHashKind', '( hkList, hkDirect )');
3819: 1282 PJclHashElem', '^TJclHashElem // will not work');
3820: 1283 TJclHashElem', 'record Next : PJclHashElem; Obj : TObject; end');
3821: 1284 PJclHashRecord', '^TJclHashRecord // will not work');
3822: 1285 PJclHashList', '^TJclHashList // will not work');
3823: 1286 TJclHashRecord', 'record Count : Byte; end');
3824: 1287 TJclSimpleXMLElemCompare', 'Function ( Elems : TJclSimpleXMLElem'
3825: 1288 TJclSimpleXMLElemClass', 'class of TJclSimpleXMLElem');
3826: 1289 TJclSimpleXMLOptions', 'set of ( sxoAutoCreate, sxoAutoIndent, s'
3827: 1290 TJclSimpleXMLEncodeEvent', 'Procedure ( Sender : TObject; var Va'
3828: 1291 TJclSimpleXMLEncodeStreamEvent', 'Procedure ( Sender : TObject; '
3829: 1291 ***** File 87: F:\maxbox3\mX397\maxbox3\source\upsi_JclStatistics.pas
3830: 1292 TDynFloatArray', 'array of Float;');
3831: 1293 TDynDoubleArray', 'array of Double;');
3832: 1293 ***** File 88: F:\maxbox3\mX397\maxbox3\source\upsi_JclSysInfo.pas
3833: 1294 TEnvironmentOption', '( eoLocalMachine, eoCurrentUser, eoAdditio'
3834: 1295 TEnvironmentOptions', 'set of TEnvironmentOption');
3835: 1296 TAPMLineStatus', '( alsOffline, alsOnline, alsUnknown )');
3836: 1297 TAPMBatteryFlag', '( abfHigh, abfLow, abfCritical, abfCharging, '
3837: 1298 TJclTerminateAppResult', '( taError, taClean, taKill )');
3838: 1299 TWindowsVersion', '( wvUnknown, wvWin95, wvWin95OSR2, wvWin98, w'
3839: 1300 TntProductType', '( ptUnknown, ptWorkStation, ptServer, ptAdvanc'
3840: 1301 TCacheInfo', 'record D : Byte; I : string; end');
3841: 1302 TFreqInfo', 'record RawFreq : Cardinal; NormFreq : Cardinal; InC'
3842: 1303 TFreeSysResKind', '( rtSystem, rtGdi, rtUser )');
3843: 1304 TFreeSystemResources', 'record SystemRes : integer; GdiRes : int'
3844: 1304 ***** File 89: F:\maxbox3\mX397\maxbox3\source\upsi_jpeg.pas
3845: 1305 TJPEGQualityRange', 'Integer');
3846: 1306 TJPEGPerformance', '( jpBestQuality, jpBestSpeed )');
3847: 1307 TJPEGScale', '( jsFullSize, jsHalf, jsQuarter, jsEighth )');
3848: 1308 TJPEGPixelFormat', '( jf24Bit, jf8Bit )');
3849: 1309 TJPEGDefaults', 'record CompressionQuality : TJPEGQualityRange; '
3850: 1309 ***** File 90: F:\maxbox3\mX397\maxbox3\source\upsi_JvBdeUtils.pas
3851: 1310 TLocateFilter', '( lfTree, lfCallback )');
3852: 1311 TBDEDataSet', 'TDataSet';
3853: 1312 TDatabaseLoginEvent', 'TLoginEvent';
3854: 1312 ***** File 91: F:\maxbox3\mX397\maxbox3\source\upsi_JvCalc.pas
3855: 1313 TJvCalcState', '( csFirst, csValid, csError )');
3856: 1314 TBidiMode', '( bdLeftToRight, bdRightToLeft, bdRightToLeftNoAlign, bdRightToLeftReadingOnly )');
3857: 1315 TVerticalAlignment', '( taAlignTop, taAlignBottom, taVerticalCenter )');
3858: 1316 THelpType', '( htKeyword, htContext )');
3859: 1316 ***** File 92: F:\maxbox3\mX397\maxbox3\source\upsi_JvComCtrls.pas
3860: 1317 TJvIpAddressMinMax', 'record Min : Byte; Max : Byte; end');
3861: 1318 TJvIpAddrFieldChangeEvent', 'Procedure ( Sender : TJvIpAddress; '
3862: 1319 TJvIPAddressChanging', 'Procedure ( Sender : TObject; Index : In'
3863: 1320 TJvTrackToolTipSide', '( tsLeft, tsTop, tsRight, tsBottom )');
3864: 1321 TJvTrackToolTipEvent', 'Procedure (Sender: TObject; var ToolTipText: string)');
3865: 1322 TPageChangeEvent', 'Procedure ( Sender : TObject; Item : TTreeNode'
3866: 1322 ***** File 93: F:\maxbox3\mX397\maxbox3\source\upsi_JvComponent.pas
3867: 1323 TJvClipboardCommand', '( caCopy, caCut, caPaste, caUndo )');
3868: 1324 TJvClipboardCommands', 'set of TJvClipboardCommand');
3869: 1324 ***** File 94: F:\maxbox3\mX397\maxbox3\source\upsi_JvCtrls.pas
3870: 1325 TJvImgBtnLayout', '( blImageLeft, blImageRight )');
3871: 1326 TJvImgBtnKind', '( bkCustom, bkOK, bkCancel, bkHelp, bkYes, bkNo'
3872: 1327 TJvImgBtnDrawEvent', 'Procedure ( Sender : TObject; const DrawIt'
3873: 1328 TJvImgBtnAnimIndexEvent', 'Procedure ( Sender : TObject; Current'
3874: 1328 ***** File 95: F:\maxbox3\mX397\maxbox3\source\upsi_JvCtrlUtils.pas
3875: 1329 PJvLVItemStateData', '^TJvLVItemStateData // will not work');
3876: 1330 TJvLVItemStateData', 'record Caption : string; Data : _Pointer; F'
3877: 1330 ***** File 96: F:\maxbox3\mX397\maxbox3\source\upsi_JvDateUtil.pas
3878: 1331 TDateOrder', '( doMDY, doDMY, doYMD )');
3879: 1332 TDayOfWeekName', '( Sun, Mon, Tue, Wed, Thu, Fri, Sat )');
3880: 1333 TDaysOfWeek', 'set of TDayOfWeekName');
3881: 1333 ***** File 97: F:\maxbox3\mX397\maxbox3\source\upsi_JvDBUtil.pas
3882: 1334 TCommit', '( ctNone, ctStep, ctAll )');
3883: 1335 TOnProgress', 'Procedure ( UserData : Integer; var Cancel : Bool'
3884: 1335 ***** File 98: F:\maxbox3\mX397\maxbox3\source\upsi_JvDBUtils.pas
3885: 1336 Largeint', 'Longint');
3886: 1336 ***** File 99: F:\maxbox3\mX397\maxbox3\source\upsi_JvExprParser.pas
3887: 1337 TOnGetVariableValue', 'Function ( Sender : TObject; const VarNam'
3888: 1338 TOnExecuteFunction', 'Function ( Sender : TObject; const FuncNam'
3889: 1338 ***** File 100: F:\maxbox3\mX397\maxbox3\source\upsi_JvFindFiles.pas
3890: 1339 TJvSpecialFolder', '( sfRecycleBin, sfControlPanel, sfDesktop, s'
3891: 1339 ***** File 101: F:\maxbox3\mX397\maxbox3\source\upsi_JvFunctions.pas
3892: 1340 HICON', 'LongWord');
3893: 1341 HRGN', 'LongWord');
3894: 1342 TdtKind', '( dtkDateOnly, dtkTimeOnly, dtkDateTime )');
3895: 1342 ***** File 102: F:\maxbox3\mX397\maxbox3\source\upsi_JvGenetic.pas
3896: 1343 TOnTest', 'Function ( Sender : TObject; Index : Integer; Member '
3897: 1343 ***** File 103: F:\maxbox3\mX397\maxbox3\source\upsi_JvgLogics.pas
3898: 1344 TLogicRule', '( lrEqual, lrBeginWith, lrEndWith, lrContains, lrC'
3899: 1344 ***** File 104: F:\maxbox3\mX397\maxbox3\source\upsi_JvgGraph.pas
3900: 1345 TPixelFormat', '( pfDevice, pf1bit, pf4bit, pf8bit, pf24bit )');
3901: 1346 TMappingMethod', '( mmHistogram, mmQuantize, mmTrunc784, mmTrunc'
3902: 1346 ***** File 105: F:\maxbox3\mX397\maxbox3\source\upsi_JvgUtils.pas
3903: 1347 TgIItemsDrawStyle', '( idsNone, idsRecessed, idsRaised )');

```

```

3904: 1348 TglWallpaperOption', '( fwoNone, fwoStretch, fwoPropStretch, fwoTile )';
3905: 1349 TglDrawState', '( fdsDefault, fdsDisabled, fdsDelicate )';
3906: 1350 TglVertAlign', '( fvaTop, fvaCenter, fvaBottom )';
3907: 1351 TglHorAlign', '( fhaLeft, fhaCenter, fhaRight )';
3908: 1352 TglSizingDir', '( fsdIncrease, fsdDecrease )';
3909: 1353 TglScalingDir', '( fsdRaising, fsdRecessing )';
3910: 1354 TglTextStyle', '( fstNone, fstRaised, fstRecessed, fstPushed, fs'
3911: 1355 TglAutoTransparentColor', '( ftcUser, ftcLeftTopPixel, ftcLeftBo'
3912: 1356 TglGradientDir', '( fgdHorizontal, fgdVertical, fgdLeftBias, fgd'
3913: 1357 TglLinesDir', '( fldHorizontal, fldVertical, fldLeftBias, fldRightBias )';
3914: 1358 TThreeDGradientType', '( fgtFlat, fgt3D )';
3915: 1359 TglLabelDir', '( fldLeftRight, fldRightLeft, fldUpDown, fldDownUp)';
3916: 1360 TglAlignment', '( ftaLeftJustify, ftaRightJustify, ftaCenter, ft'
3917: 1361 TglGlyphKind', '( fgkCustom, fgkDefault )';
3918: 1362 TglFileType', '( fftUndefined, fftGif, fftJpeg, fftBmp )';
3919: 1363 TglLabelOption', '( floActiveWhileControlFocused, floBufferedDra'
3920: 1364 TglLabelOptions', 'set of TglLabelOption';
3921: 1365 TglStaticTextOption', '( ftoActiveWhileControlFocused, ftoBroadw'
3922: 1366 TglStaticTextOptions', 'set of TglStaticTextOption';
3923: 1367 TglCheckBoxOption', '( fcoActiveWhileControlFocused, fcoBoldChec'
3924: 1368 TglCheckBoxOptions', 'set of TglCheckBoxOption';
3925: 1369 TglGroupBoxOption', '( fgoCanCollapse, fgoCollapseOther, fgoFill'
3926: 1370 TglGroupBoxOptions', 'set of TglGroupBoxOption';
3927: 1371 TglListBoxOption', '( fboAutoCtl3DColors, fboBufferedDraw, fboCh'
3928: 1372 TglListBoxOptions', 'set of TglListBoxOption';
3929: 1373 TglProgressOption', '( fpoDelineatedText, fpoTransparent )';
3930: 1374 TglProgressOptions', 'set of TglProgressOption';
3931: 1375 TglTabOption', '( ftoAutoFontDirection, ftoExcludeGlyphs, ftoHid'
3932: 1376 TglTabOptions', 'set of TglTabOption';
3933: 1377 TglTreeViewOption', '( ftvFlatScroll )';
3934: 1378 TglTreeViewOptions', 'set of TglTreeViewOption';
3935: 1379 TFocusControlMethod', '( fcmOnMouseEnter, fcmOnMouseDown, fcmOnMouseUp)';
3936: 1380 TglOnGetItemColorEvent', 'Procedure ( Sender : TObject; Index : '
3937: 1381 TglBoxStyle', '( fbsFlat, fbsCtl3D, fbsStatusControl, fbsRecesse'
3938: 1382 TglSide', '( fsdLeft, fsdTop, fsdRight, fsdBottom )';
3939: 1383 TglSides', 'set of TglSide';
3940: 1384 TglOrigin', '( forLeftTop, forRightBottom )';
3941: 1385 TglAlign', 'record Horizontal : TglHorAlign; Vertical : TglVertAlign; end';
3942: 1386 TglHComponentAlign', '( haNoChange, haLeft, haCenters, haRight, '
3943: 1387 TglVComponentAlign', '( vaNoChange, vaTops, vaCenters, vaBottoms'
3944: 1388 TglCheckKind', '( fckCheckBoxes, fckRadioButtons )';
3945: 1389 TglGlobalData', 'record fSuppressGradient : Boolean; lp3DColors '
3946: 1390 TjvBytes', '__Pointer';
3947: 1391 IntPtr', '__Pointer';
3948: 1392 TVerticalAlignment', '( taAlignTop, taAlignBottom, taVerticalCenter)';
3949: 1393 PCaptionChar', 'PChar';
3950: 1394 THintString', 'string';
3951: 1395 THintStringList', 'TStringList';
3952: 1396 TInputKey', '(ikAll, ikArrows, ikChars, ikButton, ikTabs, ikEdit, ikNative )';
3953: 1397 TInputKeys', 'set of TInputKey';
3954: 1398 TjvRGBTriple', 'record rgbBlue: Byte; rgbGreen: Byte; rgbRed: Byte; end';
3955: 1399 TjvPropertyChangeEvent', 'Procedure ( Sender : TObject; const Pr'
3956: 1400 TjvRegKey', '( hkClassesRoot, hkCurrentUser, hkLocalMachine, hkU'
3957: 1401 TjvRegKeys', 'set of TjvRegKey';
3958: 1402 TjvLinkClickEvent', 'Procedure ( Sender : TObject; Link : string)';
3959: 1403 TjvOutputMode', '( omFile, omStream )';
3960: 1404 TjvDoneFileEvent', 'Procedure ( Sender : TObject; FileName : str'
3961: 1405 TjvDoneStreamEvent', 'Procedure ( Sender : TObject; Stream : TSt'
3962: 1406 TjvHTTPProgressEvent', 'Procedure ( Sender : TObject; UserData, '
3963: 1407 TjvFTPProgressEvent', 'Procedure ( Sender : TObject; Position : '
3964: 1408 TjvClipboardCommand', '( caCopy, caCut, caPaste, caClear, caUndo)';
3965: 1409 TjvClipboardCommands', 'set of TjvClipboardCommand';
3966: 1410 TCMForceSize', 'record Msg : Cardinal; NewSize : TSmallPoint; Se'
3967: 1411 PJvRGBArray', '^TjvRGBArray // will not work';
3968: 1412 PRGBQuadArray', '^TRGBQuadArray // will not work';
3969: 1413 PRGBPalette', '^TRGBPalette // will not work';
3970: 1414 TjvErrorEvent', 'Procedure ( Sender : TObject; ErrorMessage : string)';
3971: 1415 TjvWaveLocation', '( frFile, frResource, frRAM )';
3972: 1416 TjvPopupPosition', '( ppNone, pppForm, ppApplication )';
3973: 1417 TjvProgressEvent', 'Procedure(Sender: TObject; Current, Total : Integer)';
3974: 1418 TjvNextPageEvent', 'Procedure ( Sender : TObject; PageNumber : Integer)';
3975: 1419 TjvBitmapStyle', '( bsNormal, bsCentered, bsStretched )';
3976: 1420 TjvGradientStyle', '( grFilled, grEllipse, grHorizontal, grVerti'
3977: 1421 TjvParentEvent', 'Procedure(Sender: TObject; ParentWindow: THandle)';
3978: 1422 TjvDiskRes', '( dsSuccess, dsCancel, dsSkipfile, dsError )';
3979: 1423 TjvDiskStyle', '( idfCheckFirst, idfNoBeep, idfNoBrowse, idfNoCo'
3980: 1424 TjvDiskStyles', 'set of TjvDiskStyle';
3981: 1425 TjvDeleteStyle', '( idNoBeep, idNoForeground )';
3982: 1426 TjvDeleteStyles', 'set of TjvDeleteStyle';
3983: 1427 TjvNotifyParamsEvent', 'Procedure(Sender: TObject; Params: __Pointer)';
3984: 1428 TjvFileInfoRec', 'record Attributes : DWORD; DisplayName : strin'
3985: 1429 TjvAnimation', '( anLeftRight, anRightLeft, anRightAndLeft, anLe'
3986: 1430 TjvAnimations', 'set of TjvAnimation';
3987: 1431 TjvTriggerKind', '( tkOneShot, tkEachSecond, tkEachMinute, tkEac'
3988: 1432 PJvAniTag', '^TjvAniTag // will not work';
3989: 1433 TjvAniTag', 'record ckID : TjvFourCC; ckSize : Longint; end';
3990: 1434 TjvAniHeader', 'record dwSizeof : Longint; dwFrames : Longint; d'
3991: 1435 TjvChangeColorEvent', 'Procedure ( Sender : TObject; Foreground,'
3992: 1436 TjvLayout', '( lTop, lCenter, lBottom )';

```

```

3993: 1437 TJvBevelStyle', '( bsShape, bsLowered, bsRaised )';
3994: 1438 TJvFocusChangeEvent', 'Procedure ( const ASender : TObject; cons'
3995: 1439 TTickCount', 'Cardinal');
3996: 1440 TSetOfChar', 'TSysCharSet'; set of char!!
3997: 1441 TCharSet', 'TSysCharSet');
3998: 1442 TDateOrder', '( doMDY, doDMY, doYMD )';
3999: 1443 TDayOfWeekName', '( Sun, Mon, Tue, Wed, Thu, Fri, Sat )';
4000: 1444 TDaysOfWeek', 'set of TDayOfWeekName';
4001: 1445 TJvImageSize', '( isSmall, isLarge )';
4002: 1446 TJvImageAlign', '( iaLeft, iaCentered )';
4003: 1447 TJvDriveType', '( dtUnknown, dtRemovable, dtFixed, dtRemote, dtC'
4004: 1448 TJvDriveTypes', 'set of TJvDriveType');
4005: 1449 TJvTrackFontOption', '( hoFollowFont, hoPreserveCharSet, hoPrese'
4006: 1450 TJvTrackFontOptions', 'set of TJvTrackFontOption');
4007: 1451 TJvSortMethod', '( smAutomatic, smAlphabetic, smNonCaseSensitive'
4008: 1452 TJvListViewColumnSortEvent', 'Procedure ( Sender : TObject; Colu'
4009: 1453 TJvAddInControlSiteInfo', 'record AddInControl : TControl; Bound'
4010: 1454 TJvClickColorType', '( cctColors, cctNoneColor, cctDefaultColor,'
4011: 1455 TJvHoldCustomColorEvent', 'Procedure(Sender: TObject; AColor: TColor)';
4012: 1456 TJvColorQuadLayOut', '( cqlNone, cqlLeft, cqlRight, cqlClient )';
4013: 1457 TJvGetAddInControlSiteInfoEvent', 'Procedure ( Sender : TControl'
4014: 1458 TColorType', '( ctStandard, ctSystem, ctCustom )';
4015: 1459 TDefColorItem', 'record Value : TColor; Constant : string; Descr'
4016: 1460 TJvSizeRect', 'record Top : Integer; Left : Integer; Width : Int'
4017: 1460 ***** File 106: F:\maxbox3\mX397\maxbox3\source\upsi_JvgXMLSerializer.pas
4018: 1461 TOnGetXMLHeader', 'Procedure ( Sender : TObject; var Value : string)';
4019: 1462 TBeforeParsingEvent', 'Procedure ( Sender : TObject; Buffer : PChar)';
4020: 1463 TJvgXMLSerializerException', 'class of XMLSerializerException';
4021: 1463 ***** File 107: F:\maxbox3\mX397\maxbox3\source\upsi_JvHtmlParser.pas
4022: 1464 PTagInfo', '^TTagInfo // will not work';
4023: 1465 TTagInfo', 'record BeginPos : Integer; EndPos : Integer; BeginCo'
4024: 1466 TJvKeyFoundEvent', 'Procedure ( Sender : TObject; Key, Results, '
4025: 1467 TJvKeyFoundExEvent', 'Procedure ( Sender : TObject; Key, Results'
4026: 1467 ***** File 108: F:\maxbox3\mX397\maxbox3\source\upsi_JvJCLUtils.pas
4027: 1468 TFileTime', 'Integer');
4028: 1469 TFormatSettings', 'record DecimalSeparator : Char; end';
4029: 1470 HWND', 'LongWord';
4030: 1471 TdtKind', '( dtkDateOnly, dtkTimeOnly, dtkDateTime )';
4031: 1472 HDC', 'LongWord';
4032: 1473 TIntegerListChange', 'Procedure ( Sender : TObject; Item : Integer; Action : TListNotification)';
4033: 1473 ***** File 109: F:\maxbox3\mX397\maxbox3\source\upsi_JvMail.pas
4034: 1474 TJvMailLogonOptions', '( loLogonUI, loNewSession )';
4035: 1475 TJvMailReadOptions', '( roUnreadOnly, roFifo, roPeek, roHeaderOnly, roAttachments )';
4036: 1476 TJvMailReadedData', 'record RecipientAddress : string; Recipient'
4037: 1477 TJvMailErrorEvent', 'Procedure ( Sender : TJvMail; ErrorCode : ULONG)';
4038: 1477 ***** File 110: F:\maxbox3\mX397\maxbox3\source\upsi_JvParsing.pas
4039: 1478 TUserFunction', 'function(Value: Extended): Extended;';
4040: 1479 TParserFunc', '( pfArcTan, pfCos, pfSin, pfTan, pfAbs, pfExp, pf'
4041: 1480 TUserFunction', '__Pointer');
4042: 1480 ***** File 111: F:\maxbox3\mX397\maxbox3\source\upsi_JvSimpleXml.pas
4043: 1481 IXMLNode', 'IInterface';
4044: 1482 IXMLNode', 'Interface';
4045: 1483 TJvOnSimpleXMLParsed', 'TJclOnSimpleXMLParsed';
4046: 1484 TJvOnValueParsed', 'TJclOnValueParsed';
4047: 1485 TJvOnSimpleProgress', 'TJclOnSimpleProgress';
4048: 1486 TJvSimpleXMLElem', 'TJclSimpleXMLElem';
4049: 1487 TJvSimpleXMLElemCData', 'TJclSimpleXMLElemCData';
4050: 1488 TJvSimpleXMLElemClass', 'TJclSimpleXMLElemClass';
4051: 1489 TJvSimpleXMLElemClassic', 'TJclSimpleXMLElemClassic';
4052: 1490 TJvSimpleXMLElemComment', 'TJclSimpleXMLElemComment';
4053: 1491 TJvSimpleXMLElemCompare', 'TJclSimpleXMLElemCompare';
4054: 1492 TJvSimpleXMLElemDocType', 'TJclSimpleXMLElemDocType';
4055: 1493 TJvSimpleXMLElemHeader', 'TJclSimpleXMLElemHeader';
4056: 1494 TJvSimpleXMLElemText', 'TJclSimpleXMLElemText';
4057: 1495 TJvSimpleXMLElems', 'TJclSimpleXMLElems';
4058: 1496 TJvSimpleXMLElemSheet', 'TJclSimpleXMLElemSheet';
4059: 1497 TJvSimpleXMLElemsProlog', 'TJclSimpleXMLElemsProlog';
4060: 1498 EJvSimpleXMLError', 'EJclSimpleXMLError';
4061: 1499 TJvSimpleXMLProp', 'TJclSimpleXMLProp';
4062: 1500 TJvSimpleXMLProps', 'TJclSimpleXMLProps';
4063: 1501 TJvHashKind', 'TJclHashKind';
4064: 1502 TJvHashElem', 'TJclHashElem';
4065: 1503 PJvHashElem', 'PJclHashElem';
4066: 1504 TJvHashRecord', 'TJclHashRecord';
4067: 1505 PJvHashRecord', 'PJclHashRecord';
4068: 1506 TJvHashList', 'TJclHashList';
4069: 1507 PJvHashList', 'PJclHashList';
4070: 1508 TJvSimpleXMLOptions', 'TJclSimpleXMLOptions';
4071: 1509 TJvSimpleXMLEncodeEvent', 'TJclSimpleXMLEncodeEvent';
4072: 1510 TJvSimpleXMLEncodeStreamEvent', 'TJclSimpleXMLEncodeStreamEvent';
4073: 1510 ***** File 112: F:\maxbox3\mX397\maxbox3\source\upsi_JvStrUtil.pas
4074: 1511 TSetOfChar', 'set of Char';
4075: 1512 TSetOfChar', 'string';
4076: 1512 ***** File 113: F:\maxbox3\mX397\maxbox3\source\upsi_JvStrUtils.pas
4077: 1513 TSysCharSet', 'set of Char';
4078: 1514 TCharSet', 'TSysCharSet';
4079: 1514 ***** File 114: F:\maxbox3\mX397\maxbox3\source\upsi_JvTurtle.pas
4080: 1515 TRequestBackgroundEvent', 'Procedure ( Sender : TObject; Background : string)';
4081: 1516 TRequestFilterEvent', 'Procedure ( Sender : TObject; Filter : string)';

```

```

4082: 1517 TRequestImageSizeEvent', 'Procedure ( Sender : TObject; var ARect : TRect)';
4083: 1517 ***** File 115: F:\maxbox3\mX397\maxbox3\source\upsi_JvUtils.pas
4084: 1518 Longword', 'Cardinal'); //integer to cardinal
4085: 1519 TTickCount', 'Cardinal');
4086: 1520 TSetOfChar', 'set of Char');
4087: 1521 TSetOfChar', 'string');
4088: 1522 TMenuAnimation', '( maNone, maRandom, maUnfold, maSlide )';
4089: 1523 TProcObj', 'Procedure');
4090: 1523 ***** File 116: F:\maxbox3\mX397\maxbox3\source\upsi_JvVCLUtils.pas
4091: 1524 TFillDirection', '( fdTopToBottom, fdBottomToTop, fdLeftToRight,
4092: 1525 TVertAlignment', '( vaTopJustify, vaCenter, vaBottomJustify )');
4093: 1526 TCustomForm', 'TForm');
4094: 1527 TDate', 'TDateTime');
4095: 1528 TTime', 'TDateTime');
4096: 1529 Longword', 'Longint');
4097: 1530 TSize', 'record cx: Longint; cy: Longint; end');
4098: 1531 TJvHTMLCalcType', '(htmlShow, htmlCalcWidth, htmlCalcHeight, htmlHyperLink )';
4099: 1531 ***** File 117: F:\maxbox3\mX397\maxbox3\source\upsi_JvXmlDatabase.pas
4100: 1532 TJvXMLOrderConversion', '( ocNone, ocDate, ocInteger, ocFloat )';
4101: 1533 TJvXMLSQLOperator', '( opEquals, opGreater, opSmaller, opGreater
4102: 1534 TJvXMLSetKind', '( skConstant, skColumn )';
4103: 1535 TJvXMLSetOperator', '( soNone, soAdd, soMultiply, soDivide, soSubtract )';
4104: 1536 TJvXMLInstruction', '( xiSelect, xiUpdate, xiInsert, xiDelete )';
4105: 1536 ***** File 118: F:\maxbox3\mX397\maxbox3\source\upsi_LinarBitmap.pas
4106: 1537 TLinearBitmapArray', 'array of TLinearBitmap');
4107: 1538 TLinearBitmap', 'TLinearBitmap');
4108: 1539 TLinearGraphic', 'TLinearGraphic');
4109: 1540 PBitmapLoaderList', '^TBitmapLoaderList // will not work';
4110: 1541 TProgressUpdate', 'Procedure ( Done : Integer)';
4111: 1541 ***** File 119: F:\maxbox3\mX397\maxbox3\source\upsi_MaskUtils.pas
4112: 1542 TMaskCharType', '( mcNone, mcLiteral, mcIntlLiteral, mcDirective
4113: 1543 TMaskDirectives', '( mdReverseDir, mdUpperCase, mdLowerCase, mdLiteralChar)');
4114: 1544 TMaskedText', 'string');
4115: 1545 TEditMask', 'string');
4116: 1545 ***** File 120: F:\maxbox3\mX397\maxbox3\source\upsi_MathMax.pas
4117: 1546 TFloatingPointClass', '( fpZero, fpNormal, fpDenormal, fpInfinite
4118: 1547 TPrimalityTestMethod', '( ptTrialDivision, ptRabinMiller )';
4119: 1548 TDynCardinalArray', 'array of Cardinal');
4120: 1549 Float', 'Extended');
4121: 1550 TFloatingPointClass', '( fpZero, fpNormal, fpDenormal, fpInfinite
4122: 1551 TPaymentTime', '( ptEndOfPeriod, ptStartOfPeriod )';
4123: 1551 ***** File 121: F:\maxbox3\mX397\maxbox3\source\upsi_MathUtils.pas
4124: 1552 Float', 'Single');
4125: 1553 Float', 'Double');
4126: 1554 PFloat', '^Float // will not work';
4127: 1555 PFloatArray', '^TFloatArray // will not work';
4128: 1556 TFloatDynArray', 'array of Float');
4129: 1557 PSingleArray', '^TSingleArray // will not work';
4130: 1558 PDoubleArray', '^TDoubleArray // will not work';
4131: 1559 TFloatPoint', 'record X : Float; Y : Float; end';
4132: 1560 TFloatRect', 'record Left, Top, Right, Bottom : Float; end';
4133: 1561 TFloatPointArray', 'array of TFloatPoint');
4134: 1561 ***** File 122: F:\maxbox3\mX397\maxbox3\source\upsi_MPlayer.pas
4135: 1562 TMPBtnType', '( btPlay, btPause, btStop, btNext, btPrev, btStep,
4136: 1563 TButtonSet', 'set of TMPBtnType');
4137: 1564 TMPGlyph', '( mgEnabled, mgDisabled, mgColored )';
4138: 1565 TMPDeviceTypes', '( dtAutoSelect, dtAVIVideo, dtCDAudio, dtDAT,
4139: 1566 TMPTimeFormats', '( tfMilliseconds, tfHMS, tfMSF, tfFrames, tfSM
4140: 1567 TMPModes', '( mpNotReady, mpStopped, mpPlaying, mpRecording, mpS
4141: 1568 TMPNotifyValues', '( nvSuccessful, nvSuperseded, nvAborted, nvFa
4142: 1569 TMPDevCaps', '( mpCanStep, mpCanEject, mpCanPlay, mpCanRecord, m
4143: 1570 TMPDevCapsSet', 'set of TMPDevCaps');
4144: 1571 EMPNotify', 'Procedure ( Sender : TObject; Button : TMPBtnType;
4145: 1572 EMPPostNotify', 'Procedure ( Sender : TObject; Button : TMPBtnTy
4146: 1572 ***** File 123: F:\maxbox3\mX397\maxbox3\source\upsi_niRegularExpression.pas
4147: 1573 TniRegularExpressionParserState', '( psStates, psCharRange )';
4148: 1574 TniRegularExpressionParserFlag', '( pfEscaped )';
4149: 1575 TniRegularExpressionParserFlags', 'set of TniRegularExpressionPa
4150: 1576 TniRegularExpressionStateType', '( stNormal, stGroupStart, stGro
4151: 1577 TniRegularExpressionMatchFlag', '( mfStartOnly, mfFinishOnly, mf
4152: 1578 TniRegularExpressionMatchFlags', 'set of TniRegularExpressionMat
4153: 1579 TniRegularExpressionMatchResult', '( mrNone, mrFail, mrMatch, mr
4154: 1580 TniRegularExpressionMatchFoundEvent', 'Procedure ( const iStart
4155: 1580 ***** File 124: F:\maxbox3\mX397\maxbox3\source\upsi_niSTRING.pas
4156: 1581 TCharset', 'set of char');
4157: 1581 ***** File 125: F:\maxbox3\mX397\maxbox3\source\upsi_OpenSSLUtils.pas
4158: 1582 TEncoding', '( auto, PEM, DER, SMIME, NETSCAPE, PKCS12 )';
4159: 1583 TX509CertificateArray', 'array of TX509Certificate');
4160: 1583 ***** File 126: F:\maxbox3\mX397\maxbox3\source\upsi_Outline.pas
4161: 1584 TOutlineNodeCompare', '( ocLess, ocSame, ocGreater, ocInvalid )';
4162: 1585 TAttachMode', '( oaAdd, oaAddChild, oaInsert )';
4163: 1586 TBitmapArrayRange', 'Integer');
4164: 1587 EOutlineChange', 'Procedure ( Sender : TObject; Index : LongInt)';
4165: 1588 TOutlineStyle', '( osText, osPlusMinusText, osPictureText, osPlu
4166: 1589 TOutlineBitmap', '( obPlus, obMinus, obOpen, obClose, obLeaf )';
4167: 1590 TOutlineBitmaps', 'set of TOutlineBitmap');
4168: 1591 TOutlineType', '( otStandard, otOwnerDraw )';
4169: 1592 TOutlineOption', '( ooDrawTreeRoot, ooDrawFocusRect, ooStretchBi
4170: 1593 TOutlineOptions', 'set of TOutlineOption');

```



```

4171: 1593 ***** File 127: F:\maxbox3\mX397\maxbox3\source\upsi_ParallelJobs.pas
4172: 1594 TWaitProcessNotify, 'Procedure';
4173: 1595 TJobState, '( jgsRunning, jgsStopped )';
4174: 1596 PJobItem, '^TJobItem // will not work';
4175: 1597 TJobItem, 'record Job : Pointer; prev : PJobItem; next : PJobItem; end';
4176: 1598 PJobsGroup, '^TJobsGroup // will not work';
4177: 1598 ***** File 128: F:\maxbox3\mX397\maxbox3\source\upsi_PerlRegEx.pas
4178: 1599 TPerlRegExOption, '( preCaseLess, preMultiline, preSing'
4179: 1600 TPerlRegExState, '( preNotBOL, preNotEOL, preNotEmpty )';
4180: 1601 TPerlRegExOptions, 'set of TPerlRegExOption';
4181: 1602 PCREString, 'UTF8String';
4182: 1603 PCREString, 'AnsiString';
4183: 1604 TPerlRegExReplaceEvent, 'Procedure ( Sender : TObject; var Repl'
4184: 1604 ***** File 129: F:\maxbox3\mX397\maxbox3\source\upsi_Printers.pas
4185: 1605 TPrinterState, '( psNoHandle, psHandleIC, psHandleDC )';
4186: 1606 TPrinterOrientation, '( poPortrait, poLandscape )';
4187: 1607 TPrinterCapability, '( pcCopies, pcOrientation, pcCollation )';
4188: 1608 TPrinterCapabilities, 'set of TPrinterCapability';
4189: 1608 ***** File 130: F:\maxbox3\mX397\maxbox3\source\upsi_Provider.pas
4190: 1609 TGetRecordOption, '( grMetaData, grReset, grXML, grXMLUTF8 )';
4191: 1610 TGetRecordOptions, 'set of TGetRecordOption';
4192: 1611 TDataRequestEvent, 'Function ( Sender : TObject; Input : OleVariant ) : OleVariant';
4193: 1612 TProviderOption, '( poFetchBlobsOnDemand, poFetchDetailsOnDeman'
4194: 1613 TProviderOptions, 'set of TProviderOption';
4195: 1614 PPutFieldInfo, '^TPutFieldInfo // will not work';
4196: 1615 TPutFieldProc, 'Procedure ( Info : PPutFieldInfo)';
4197: 1616 TPutFieldInfo, 'record FieldNo : Integer; Field : TField; DataS'
4198: 1617 TInfoArray, 'array of TPutFieldInfo';
4199: 1618 TGetParamsEvent, 'Procedure ( DataSet : TDataSet; Params : TList)';
4200: 1619 TOnValidate, 'Procedure ( const Delta : OleVariant)';
4201: 1620 TResolverResponse, '( rrSkip, rrAbort, rrMerge, rrApply, rrIgnore )';
4202: 1621 TProviderDataEvent, 'Procedure ( Sender : TObject; DataSet : TC'
4203: 1622 TBeforeUpdateRecordEvent, 'Procedure ( Sender : TObject; Source'
4204: 1623 TAfterUpdateRecordEvent, 'Procedure ( Sender : TObject; Sourced'
4205: 1624 TResolverErrorEvent, 'Procedure ( Sender : TObject; DataSet : T'
4206: 1625 TBeforeCommitEvent, 'Procedure ( Sender : TObject; DeltaDS : TC'
4207: 1626 TGetTableNameEvent, 'Procedure ( Sender : TObject; DataSet : TD'
4208: 1627 TGetDSProps, 'Procedure ( Sender : TObject; DataSet : TDataSet;'
4209: 1627 ***** File 131: F:\maxbox3\mX397\maxbox3\source\upsi_ScktComp.pas
4210: 1628 TCMSocketMessage, 'record Msg : Cardinal; Socket : TSocket; Sel'
4211: 1629 TCMLookupComplete, 'record Msg : Cardinal; LookupHandle : THand'
4212: 1630 TServerType, '( stNonBlocking, stThreadBlocking )';
4213: 1631 TClientType, '( ctNonBlocking, ctBlocking )';
4214: 1632 TAsyncStyle, '( asRead, asWrite, asOOB, asAccept, asConnect, as'
4215: 1633 TAsyncStyles, 'set of TAsyncStyle';
4216: 1634 TSocketEvent, '( seLookup, seConnecting, seConnect, seDisconnec'
4217: 1635 TLookupState, '( lsIdle, lsLookupAddress, lsLookupService )';
4218: 1636 TErrorEvent, '( eeGeneral, eeSend, eeReceive, eeConnect, eeDisc'
4219: 1637 TSocketEventEvent, 'Procedure ( Sender : TObject; Socket : TCus'
4220: 1638 TSocketErrorEvent, 'Procedure ( Sender : TObject; Socket : TCus'
4221: 1639 TGetSocketEvent, 'Procedure ( Sender : TObject; Socket : TSocke'
4222: 1640 TGetThreadEvent, 'Procedure ( Sender : TObject; ClientSocket : '
4223: 1641 TSocketNotifyEvent, 'Procedure ( Sender : TObject; Socket : TCu'
4224: 1642 TThreadNotifyEvent, 'Procedure ( Sender : TObject; Thread : TSe'
4225: 1642 ***** File 132: F:\maxbox3\mX397\maxbox3\source\upsi_Serial.pas
4226: 1643 TCOMInfo, 'record Number : integer; Instance : TCustomSerial; e'
4227: 1644 PCOMInfo, '^TCOMInfo // will not work';
4228: 1645 TFeedbackProc, 'Procedure';
4229: 1646 PFeedback, '^TFeedback // will not work';
4230: 1647 TFeedback, 'record Proc : TFeedbackProc; Next : PFeedback; end';
4231: 1648 TObjProc, 'Procedure ( Message, wParam, lParam : longint)';
4232: 1649 TDataBits, '( db_4, db_5, db_6, db_7, db_8 )';
4233: 1650 TParityBit, '( none, odd, even, mark, space )';
4234: 1651 TStopBits, '( sb_1, sb_15, sb_2 )';
4235: 1652 TSerialConfig, 'record Valid : boolean; COMPort : integer; Baud'
4236: 1653 TRText, 'Function : string';
4237: 1654 TTDData, 'Function ( var Data, DataSize : integer ) : integer';
4238: 1655 TRData, 'Function ( var Buf, BufSize : integer ) : integer';
4239: 1656 TTText, 'Function ( s : string ) : integer';
4240: 1657 TTDData, 'Function ( var Data, DataSize : cardinal ) : cardinal';
4241: 1658 TRData, 'Function ( var Buf, BufSize : cardinal ) : cardinal';
4242: 1659 TTText, 'Function ( s : string ) : cardinal';
4243: 1659 ***** File 133: F:\maxbox3\mX397\maxbox3\source\upsi_ShadowWnd.pas
4244: 1660 TControlSide, '( csRight, csBottom )';
4245: 1660 ***** File 134: F:\maxbox3\mX397\maxbox3\source\upsi_ShLwApi.pas
4246: 1661 TREFIID, 'TGUID';
4247: 1662 URL_SCHEME, 'Integer';
4248: 1663 URL_PART, 'DWORD';
4249: 1664 TUrlIs, 'DWORD';
4250: 1665 TSHRegDelFlags, 'DWORD';
4251: 1666 TSHRegEnumFlags, 'DWORD';
4252: 1667 HUSKEY, 'THandle';
4253: 1668 PHUSKEY, '^HUSKEY // will not work';
4254: 1669 ASSOCF, 'DWORD';
4255: 1670 ASSOCSTR, 'DWORD';
4256: 1671 ASSOCKEY, 'DWORD';
4257: 1672 ASSOCDATA, 'DWORD';
4258: 1673 ASSOCENUM, 'DWORD';
4259: 1673 ***** File 135: F:\maxbox3\mX397\maxbox3\source\upsi_SOAPHTTIClient.pas

```



```

4260: 1674 TGUID, 'record D1: LongWord; D2: word; D3: word; D4: array[0..7] of Byte; end';
4261: 1674 ***** File 136: F:\maxbox3\mX397\maxbox3\source\upsi_SortThds.pas
4262: 1675 TSortArray, 'array[1..ARRSIZE] of Integer;';
4263: 1675 ***** File 137: F:\maxbox3\mX397\maxbox3\source\upsi_SqlExpr.pas
4264: 1676 TWideDataSet, 'TDataSet';
4265: 1677 TFieldList, 'TObjectList';
4266: 1678 TLocale, 'IntPtr';
4267: 1679 TFieldList, 'TList';
4268: 1680 TLocale, 'Pointer';
4269: 1681 SQLResult, 'TDBXErrorCode';
4270: 1682 TLocaleCode, 'Integer';
4271: 1683 TSQLErrorType, '( exceptConnection, exceptCommand, exceptCu'
4272: 1684 TTransIsolationLevel, '( xilREADCOMMITTED, xilREPEATABLEREAD, x'
4273: 1685 TConnectionUserType, '( eUserMonitor, eUserDataSet );';
4274: 1686 TTraceEvent, 'Procedure ( Sender : TObject; TraceInfo : TDBXTra'
4275: 1687 TTraceLogEvent, 'Procedure ( Sender : TObject; TraceInfo : TDBX'
4276: 1688 EConnectFlag, '( eConnect, eReconnect, eDisconnect );';
4277: 1689 TSchemaType, '( stNoSchema, stTables, stSysTables, stProcedures'
4278: 1690 TConnectionState, '( csStateClosed, csStateOpen, csStateConnect'
4279: 1691 TSQLCommandType, '(ctQuery, ctTable, ctStoredProc); //3.7
4280: 1692 TTableScope, '( tsSynonym, tsSysTable, tsTable, tsView );';
4281: 1693 TTableScopes, 'set of TTableScope';
4282: 1694 TSQLConnectionLoginEvent, 'Procedure ( Database : TSQLConnectio'
4283: 1695 TSQLSchemaInfo, 'record FType : TSchemaType; ObjectName : WideS'
4284: 1696 TFieldDescList, 'array of TFLDDesc';
4285: 1697 TParseSqlEvent, 'Procedure ( var FieldNames : TWideStrings; SQL'
4286: 1698 TParseInsertSqlEvent, 'Procedure ( var FieldNames : TWideString'
4287: 1698 ***** File 138: F:\maxbox3\mX397\maxbox3\source\upsi_SqlTimSt.pas
4288: 1699 PSQLTimeStamp, '^TSQLTimeStamp // will not work';
4289: 1700 TSQLTimeStamp, 'record Year:SmallInt; Month:Word; Day:Word; Hour:Word; Minute:Word; Second:Word;
Fractions:LongWord; end';
4290: 1701 _SYSTEMTIME, 'record wYear : Word; wMonth : Word; wDayOfWeek : '
4291: 1702 TSystemTime, '_SYSTEMTIME';};
4292: 1703 _TIME_ZONE_INFORMATION, 'record Bias:Integer;StandardName: array[0..31] of PCHAR;'+
4293: 1704 TTimeZoneInformation, '_TIME_ZONE_INFORMATION';
4294: 1705 TIME_ZONE_INFORMATION, '_TIME_ZONE_INFORMATION';
4295: 1706 TTimeZone, 'record FUsesDaylightSavings : Boolean; FInfo : TTime'
4296: 1706 ***** File 139: F:\maxbox3\mX397\maxbox3\source\upsi_StAstroP.pas
4297: 1707 TStPlanetsRec, 'record RA, DC, Elong: Double; end';
4298: 1708 TStPlanetsArray, 'array[1..8] of TStPlanetsRec;';
4299: 1708 ***** File 140: F:\maxbox3\mX397\maxbox3\source\upsi_StBCD.pas
4300: 1709 TBCDS, 'array[0..BcdSize-1] of Byte';
4301: 1710 TBCD, 'array[0..BcdSize-1] of Byte';
4302: 1710 ***** File 141: F:\maxbox3\mX397\maxbox3\source\upsi_StDecMth.pas
4303: 1711 TStRoundMethod, '( rmNormal, rmTrunc, rmBankers, rmUp );';
4304: 1711 ***** File 142: F:\maxbox3\mX397\maxbox3\source\upsi_StExpLog.pas
4305: 1712 TStOnExceptionFilter, 'Procedure ( Sender : TObject; E : Except'
4306: 1712 ***** File 143: F:\maxbox3\mX397\maxbox3\source\upsi_StExport.pas
4307: 1713 TStExportProgressEvent, 'Procedure ( Sender : TObject; Index : '
4308: 1713 ***** File 144: F:\maxbox3\mX397\maxbox3\source\upsi_StFIN.pas
4309: 1714 TStPaymentTime, '( ptEndOfPeriod, ptStartOfPeriod );';
4310: 1715 TStFrequency, '( fgAnnual, fgSemiAnnual, fgQuarterly, fgMonthly);';
4311: 1716 TStBasis, '( BasisNASD, BasisActAct, BasisAct360, BasisAct365, '
4312: 1717 TStDate, 'Longint';
4313: 1718 TStTime, 'Longint';
4314: 1718 ***** File 145: F:\maxbox3\mX397\maxbox3\source\upsi_StGenLog.pas
4315: 1719 TStGetLogStringEvent, 'Procedure ( Sender : TObject; const D1, '
4316: 1720 TStWriteMode, '( wmOverwrite, wmAppend );';
4317: 1721 PStLogRec, '^TStLogRec // will not work';
4318: 1722 TStLogRec, 'record lrTime : DWORD; lrData1 : DWORD; lrData2 : D'
4319: 1723 PStLogBuffer, '^TStLogBuffer // will not work';
4320: 1724 TStGenOptions, '( goSuppressEnableMsg, goSuppressDisableMsg );';
4321: 1725 TStGenOptionSet, 'set of TStGenOptions';
4322: 1725 ***** File 146: F:\maxbox3\mX397\maxbox3\source\upsi_StNetCon.pas
4323: 1726 TStNetConnectOptions, '( coUseConnectDialog, coPersistentConne'
4324: 1727 TStNetDisconnectOptions, '( doUseDisconnectDialog, doUpdateProf'
4325: 1728 TStNetConnectOptionsSet, 'set of TStNetConnectOptions';
4326: 1729 TStNetDisconnectOptionsSet, 'set of TStNetDisconnectOptions';
4327: 1730 TOnConnectFailEvent, 'Procedure ( Sender : TObject; ErrorCode: DWord);';
4328: 1731 TOnConnectCancelEvent, 'Procedure ( Sender : TObject; ErrorCode: DWord);';
4329: 1732 TOnDisconnectFailEvent, 'Procedure ( Sender : TObject; ErrorCode: DWord);';
4330: 1733 TOnDisconnectCancelEvent, 'Procedure ( Sender : TObject; ErrorCode: DWord);';
4331: 1733 ***** File 147: F:\maxbox3\mX397\maxbox3\source\upsi_StPtrns.pas
4332: 1734 TStMediatorAction, 'Procedure ( aInputData, aResultData : Tobje'
4333: 1735 TStObserverAction, 'Procedure ( aInputData : TObject);';
4334: 1736 TStChainAction, 'Procedure ( aInputData, aResultData : TObject; '
4335: 1736 ***** File 148: F:\maxbox3\mX397\maxbox3\source\upsi_StRegEx.pas
4336: 1737 TMatchPosition, 'record StartPos : Cardinal; EndPos : Cardinal; '
4337: 1738 TStOutputOption, '( ooUnselected, ooModified, ooCountOnly );';
4338: 1739 TStOutputOptions, 'set of TStOutputOption';
4339: 1740 TStTokens, '( tknNil, tknLitChar, tknCharClass, tknNegCharClass'
4340: 1741 PStPatRecord, '^TStPatRecord // will not work';
4341: 1742 TStPatRecord, 'record StrPtr : ^ShortString // will not work; '
4342: 1743 TStOnRegExProgEvent, 'Procedure ( Sender : TObject; Percent : W'
4343: 1744 TStOnMatchEvent, 'Procedure ( Sender : TObject; REPosition : TM'
4344: 1744 ***** File 149: F:\maxbox3\mX397\maxbox3\source\upsi_StrUtils.pas
4345: 1745 TReplaceFlag, '(rfReplaceAll, rfIgnoreCase);';
4346: 1746 TReplaceFlags, 'set of TReplaceFlag';
4347: 1747 TStringSeachOption, '(soDown, soMatchCase, soWholeWord );';

```

```

4348: 1748 TStringSearchOption', '(soDown, soMatchCase, soWholeWord)');
4349: 1749 TMsgsByteType', '(mbSingleByte, mbLeadByte, mbTrailByte)');
4350: 1750 TFloatValue', '(fvExtended, fvCurrency)');
4351: 1751 TFloatFormat', '(ffGeneral, ffExponent, ffFixed, ffNumber, ffCurrency)');
4352: 1752 TMsgDlgType', '(mtWarning, mtError, mtInformation, mtConfirmation, mtCustom)');
4353: 1753 TMsgDlgBtn(mbYes, mbNo, mbOK, mbCancel, mbAbort, mbRetry, mbIgnore, mbAll, mbNoToAll, mbYesToAll, mbHelp);
4354: 1754 TMsgDlgButtons', 'set of TMsgDlgBtn)';
4355: 1755 TValueRelationship', 'array [-1..1] of byte)';
4356: 1756 THandle', 'LongWord';
4357: 1757 TFileName', 'string';
4358: 1758 TFileTime', 'record dwLowDateTime: LongWORD; dwHighDateTime: LongWORD; end';
4359: 1759 TWin32FindData', 'record dwFileAttributes: LongWord; ftCreationTime, ftLastAccessTime, ftLastWriteTime:
TFileTime;
4360: 1760 TFormatSettings', 'record CurrencyFormat: Byte; NegCurrFormat: Byte;
4361: 1761 TSearchRec', 'record Time: Integer; Size: Int64; Attr: Integer;
4362: 1762 TSearchRec', 'record Time: Integer; Size: Int64; Attr: Integer; Name: TFileName; ExcludeAttr:
Integer; FindHandle: THandle; FindData: TWin32FindData; end';
4363: 1763 TSmallPoint', 'record x: smallint; y: smallint; end';
4364: 1764 TGUID', 'record D1: LongWord; D2: word; D3: word; D4: array[0..7] of Byte; end';
4365: 1765 TGUID', 'TGUID';
4366: 1766 HRESULT', 'longint';
4367: 1767
4368: 1768 TSysCharSet', 'set of Char';
4369: 1769 TStringSearchOptions', 'set of TStringSeachOption';
4370: 1770 TSoundexLength', 'Integer';
4371: 1771 TSoundexIntLength', 'Integer';
4372: 1772 TextFile', 'string'; //ex integer beta in mx3
4373: 1773 Text', 'file of char'; //beta in mx3
4374: 1774 File', 'string'; //ex integer beta in mx3
4375: 1775 TBuffer', 'PChar'; //beta in mx3
4376: 1776 TTextLineBreakStyle', '(tlbsLF, tlbsCRLF)'; //beta in mx3
4377: 1777 RandSeed', 'Longint'; //beta in mx3 randseed now a functionvar
4378: 1777 ***** File 150: F:\maxbox3\mx397\maxbox3\source\upsi_StStat.pas
4379: 1778 TStLinEst', 'record B0: Double; B1: double; seB0: double; seB
4380: 1778 ***** File 151: F:\maxbox3\mx397\maxbox3\source\upsi_StStms.pas
4381: 1779 TStMemSize', 'Integer';
4382: 1780 TStreamReader', 'TStAnsiTextStream';
4383: 1780 ***** File 152: F:\maxbox3\mx397\maxbox3\source\upsi_StSystem.pas
4384: 1781 MediaIDType', 'record InfoLevel: Word; SerialNumber: LongInt; VolumeLabel: array[0..10] of Char;
FileSystemID: array[0..7] of Char; end';
4385: 1782 TStDateTimeRec', 'record D: TStDate; T: TStTime; end';
4386: 1783 DiskClass', '(Floppy360, Floppy720, Floppy144, OtherF'
4387: 1784 PMediaIDType', 'MediaIDType // will not work';
4388: 1784 ***** File 153: F:\maxbox3\mx397\maxbox3\source\upsi_StToHTML.pas
4389: 1785 TStOnProgressEvent', 'Procedure (Sender: TObject; Percent: Word)';
4390: 1785 ***** File 154: F:\maxbox3\mx397\maxbox3\source\upsi_StTxtDat.pas
4391: 1786 TStSchemaLayoutType', '(ltUnknown, ltFixed, ltVarying)';
4392: 1787 TStSchemaFieldType', '(sftUnknown, sftChar, sftFloat, sftNumber'
4393: 1788 TStOnQuoteFieldEvent', 'Procedure (Sender: TObject; var Field'
4394: 1788 ***** File 155: F:\maxbox3\mx397\maxbox3\source\upsi_SyncObjs.pas
4395: 1789 PSecurityAttributes', '___pointer';
4396: 1790 TWaitResult', '(wrSignaled, wrTimeout, wrAbandoned, wrError)';
4397: 1790 ***** File 156: F:\maxbox3\mx397\maxbox3\source\upsi_SynHighlighterMulti.pas
4398: 1791 TOnCheckMarker', 'Procedure (Sender: TObject; var StartPos, Ma
4399: 1792 TgmSchemeClass', 'class of TScheme';
4400: 1793 TRangeOperation', '(roGet, roSet)';
4401: 1794 TRangeProc', 'Procedure (Operation: TRangeOperation; var Range'
4402: 1795 TCustomRangeEvent', 'Procedure (Sender: TSynMultiSyn; Operatio
4403: 1795 ***** File 157: F:\maxbox3\mx397\maxbox3\source\upsi_SynRegExpr.pas
4404: 1796 PRegExprChar', 'PWideChar';
4405: 1797 RegExprString', 'WideString';
4406: 1798 REChar', 'WideChar';
4407: 1799 PRegExprChar', 'PChar';
4408: 1800 RegExprString', 'AnsiString';
4409: 1801 REChar', 'Char';
4410: 1802 TREOp', 'REChar';
4411: 1803 PREOp', '^TREOp // will not work';
4412: 1804 TRENextOff', 'integer';
4413: 1805 PRENextOff', '^TRENextOff // will not work';
4414: 1806 TREBracesArg', 'integer';
4415: 1807 PREBracesArg', '^TREBracesArg // will not work';
4416: 1808 TRegExprInvertCaseFunction', 'Function(const Ch: REChar): REChar';
4417: 1809 PSetOfREChar', 'TSetOfREChar // will not work';
4418: 1810 TSetOfREChar', 'set of REChar';
4419: 1811 TRegExprReplaceFunction', 'Function(ARegExpr: TRegExpr): string';
4420: 1811 ***** File 158: F:\maxbox3\mx397\maxbox3\source\upsi_System.pas
4421: 1812 TRuntimeError', '(reNone, reOutOfMemory, reInvalidPtr, reDivByZ'
4422: 1812 ***** File 159: F:\maxbox3\mx397\maxbox3\source\upsi_Tabs.pas
4423: 1813 TScrollBtn', '(sbLeft, sbRight)';
4424: 1814 TScrollOrientation', '(soLeftRight, soUpDown)';
4425: 1815 TEdgePart', '(epSelectedLeft, epUnselectedLeft, epSelectedRight'
4426: 1816 TEdgeType', '(etNone, etFirstIsSel, etFirstNotSel, etLastIsSel,'
4427: 1817 TTabSetTabStyle', '(tsStandard, tsOwnerDraw, tsSoftTabs, tsMode'
4428: 1818 TTabStyle', 'TTabSetTabStyle';
4429: 1819 TMeasureTabEvent', 'Procedure (Sender: TObject; Index: Intege
4430: 1820 TTabSetDrawTabEvent', 'Procedure (Sender: TObject; TabCanvas:
4431: 1821 TDrawTabEvent', 'TTabSetDrawTabEvent';
4432: 1822 TTabChangeEvent', 'Procedure (Sender: TObject; NewTab: Intege
4433: 1822 ***** File 160: F:\maxbox3\mx397\maxbox3\source\upsi_ToolWin.pas

```

```

4434: 1823 TEdgeBorder', '( ebLeft, ebTop, ebRight, ebBottom )';
4435: 1824 TEdgeBorders', 'set of TEdgeBorder';
4436: 1825 TEdgeStyle', '( esNone, esRaised, esLowered )';
4437: 1826 TSizingOrientation', '( soNone, soHorizontal, soVertical )';
4438: 1826 ***** File 161: F:\maxbox3\mX397\maxbox3\source\upsi_Types.pas
4439: 1827 TIntegerDynArray', 'array of Integer';
4440: 1828 TCardinalDynArray', 'array of Cardinal';
4441: 1829 TWordDynArray', 'array of Word';
4442: 1830 TSmallIntDynArray', 'array of SmallInt';
4443: 1831 TByteDynArray', 'array of Byte';
4444: 1832 TCharDynArray', 'array of Char';
4445: 1833 TShortIntDynArray', 'array of ShortInt';
4446: 1834 TInt64DynArray', 'array of Int64';
4447: 1835 TLongWordDynArray', 'array of LongWord';
4448: 1836 TSingleDynArray', 'array of Single';
4449: 1837 TDoubleDynArray', 'array of Double';
4450: 1838 TBooleanDynArray', 'array of Boolean';
4451: 1839 TStringDynArray', 'array of string';
4452: 1840 TWideStringDynArray', 'array of WideString';
4453: 1841 TSize', 'tagSIZE';
4454: 1842 SIZE', 'tagSIZE';
4455: 1843 Longword', 'Cardinal'; //unsigned integer to cardinal
4456: 1844 DWORD', 'Cardinal';
4457: 1845 PDisplay', 'Pointer';
4458: 1846 PEvent', 'Pointer';
4459: 1847 TXrmOptionDescRec', 'record end';
4460: 1848 XrmOptionDescRec', 'TXrmOptionDescRec';
4461: 1849 PXrmOptionDescRec', '^TXrmOptionDescRec // will not work';
4462: 1850 Widget', 'Pointer';
4463: 1851 WidgetClass', 'Pointer';
4464: 1852 ArgList', 'Pointer';
4465: 1853 Region', 'Pointer';
4466: 1854 TOleChar', 'WideChar';
4467: 1855 POleStr', 'PWideChar';
4468: 1856 PPoleStr', '^PoleStr // will not work';
4469: 1857 PCLSID', 'PGUID';
4470: 1858 TCLSID', 'TGUID';
4471: 1859 Largeint', 'Int64';
4472: 1860 PDWORD', '^DWORD // will not work';
4473: 1861 TFileTime', '_FILETIME';
4474: 1862 FILETIME', '_FILETIME';
4475: 1863 TStatStg', 'tagSTATSTG';
4476: 1864 STATSTG', 'TStatStg';
4477: 1865 TPointArray', 'array of TPoint';
4478: 1866 TPoint', 'record X, Y: LongInt; end;';
4479: 1866 ***** File 162: F:\maxbox3\mX397\maxbox3\source\upsi_uTPLb_HugeCardinal.pas
4480: 1867 TCompareResult', '( rGreaterThan, rEqualTo, rLessThan )';
4481: 1868 TByteOrder', '( LittleEndian, BigEndian )';
4482: 1869 TProgress', 'Procedure ( Sender : TObject; BitsProcessed, TotalB
4483: 1869 ***** File 163: F:\maxbox3\mX397\maxbox3\source\upsi_uTPLb_HugeCardinalUtils.pas
4484: 1870 TPrimalityTestNoticeProc', 'Procedure ( CountPrimalityTests : integer)';
4485: 1870 ***** File 164: F:\maxbox3\mX397\maxbox3\source\upsi_uTPLb_IntegerUtils.pas
4486: 1871 uint64', 'int64';
4487: 1872 uint32', 'cardinal';
4488: 1873 uint16', 'word';
4489: 1873 ***** File 165: F:\maxbox3\mX397\maxbox3\source\upsi_uTPLb_SHA2.pas
4490: 1874 TSHA2FamilyMember', '( SHA_224, SHA_256, SHA_384, SHA_512, SHA_5
4491: 1874 ***** File 166: F:\maxbox3\mX397\maxbox3\source\upsi_ValEdit.pas
4492: 1875 TDisplayOption', '( doColumnTitles, doAutoColResize, doKeyColFixed )';
4493: 1876 TDisplayOptions', 'set of TDisplayOption';
4494: 1877 TKeyOption', '( keyEdit, keyAdd, keyDelete, keyUnique )';
4495: 1878 TKeyOptions', 'set of TKeyOption';
4496: 1879 TGetPickListEvent', 'Procedure ( Sender : TObject; const KeyName'
4497: 1880 TOnValidateEvent', 'Procedure ( Sender : TObject; ACol, ARow : L
4498: 1881 TItemProps', 'array of TItemProp';
4499: 1881 ***** File 167: F:\maxbox3\mX397\maxbox3\source\upsi_Variants.pas
4500: 1882 TVarType', 'Word'; in upscompiler!
4501: 1883 TVarData2', 'record vtype: tvartype; RawData: array [0..3] of LongInt; end;';
4502: 1884 TVarData3', 'record vtype: tvartype; end;';
4503: 1885 TVarData', 'record vtype: word; reserved1, reserved2, reserved3: word; varword: tvartype; end;';
4504: 1886 TVariantRelationship', '( vrEqual, vrLessThan, vrGreaterThan, vrNotEqual )';
4505: 1887 TVarCompareResult', '( crLessThan, crEqual, crGreaterThan )';
4506: 1888 TCustomVariantTypeClass', 'class of TCustomVariantType';
4507: 1889 TVarDataArray', 'array of TVarData';
4508: 1890 TNullCompareRule', '( ncrError, ncrStrict, ncrLoose )';
4509: 1891 TBooleanToStringRule', '( bsrAsIs, bsrLower, bsrUpper )';
4510: 1891 ***** File 168: F:\maxbox3\mX397\maxbox3\source\upsi_WideStrings.pas
4511: 1892 PWideStringItem', '^TWideStringItem // will not work';
4512: 1893 TWideStringItem', 'record FString : WideString; FObject: TObject; end;';
4513: 1894 PWideStringItemList', '^TWideStringItemList // will not work';
4514: 1894 ***** File 169: F:\maxbox3\mX397\maxbox3\source\upsi_WideStrUtils.pas
4515: 1895 TEncodeType', '( etUSASCII, etUTF8, etANSI )';
4516: 1896 CharSet', 'set of ansichar';
4517: 1896 ***** File 170: F:\maxbox3\mX397\maxbox3\source\upsi_Windows_draft.pas
4518: 3655 PDDEData', 'TDDEData // will not work';
4519: 3656 TDDEData', 'DDEDATA';
4520: 3657 PDDEPoke', 'TDDEPoke // will not work';
4521: 3658 TDDEPoke', 'DDEPOKE';
4522: 3658 ***** File 171: F:\maxbox3\mX397\maxbox3\source\upsi_xmlutil.pas

```

```

4523: 3659 IDOMNode', 'interface');
4524: 3659 ***** File 172: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_cFileUtils.pas
4525: 3660 TFileHandle', 'Integer');
4526: 3661 TFileAccess', '( faRead, faWrite, faReadWrite )';
4527: 3662 TFileSharing', '( fsDenyNone, fsDenyRead, fsDenyWrite, fsDenyReadWrite, fsExclusive )';
4528: 3663 TFileOpenFlags', '( foDeleteOnClose, foNoBuffering, foWri
4529: 3664 TFileCreationMode', '( fcCreateNew, fcCreateAlways, fcOpenExisti
4530: 3665 TFileSeekPosition', '( fpOffsetFromStart, fpOffsetFromCurrent, fpOffsetFromEnd )';
4531: 3666 PFileOpenWait', '^TFileOpenWait // will not work';
4532: 3667 TFileOpenWait', 'record Wait : Boolean; UserData : LongWord; Tim
4533: 3668 TLogicalDriveType', '( DriveRemovable, DriveFixed, DriveRemote, '
4534: 3668 ***** File 173: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_cFundamentUtils.pas
4535: 3669 Int8', 'ShortInt');
4536: 3670 Int16', 'SmallInt');
4537: 3671 Int32', 'LongInt');
4538: 3672 UInt8', 'Byte');
4539: 3673 UInt16', 'Word');
4540: 3674 UInt32', 'LongWord');
4541: 3675 UInt64', 'Int64');
4542: 3676 Word8', 'UInt8');
4543: 3677 Word16', 'UInt16');
4544: 3678 Word32', 'UInt32');
4545: 3679 Word64', 'UInt64');
4546: 3680 LargeInt', 'Int64');
4547: 3681 NativeInt', 'Integer');
4548: 3682 NativeUInt', 'Cardinal');
4549: 3683 Bool8', 'ByteBool');
4550: 3684 Bool16', 'WordBool');
4551: 3685 Bool32', 'LongBool');
4552: 3686 TCompareResult', '( crLess, crEqual, crGreater, crUndefined )';
4553: 3687 TCompareResultSet', 'set of TCompareResult';
4554: 3688 CharSet', 'set of AnsiChar');
4555: 3689 CharSet', 'set of Char'; //!!!
4556: 3690 AnsiCharSet', 'TCharSet';
4557: 3691 ByteSet', 'set of Byte');
4558: 3692 AnsiChar', 'Char');
4559: 3693 UnicodeChar', 'WideChar');
4560: 3694 TConvertResult', '( convertOK, convertFormatError, convertOverflow )';
4561: 3694 ***** File 174: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_cTimers.pas
4562: 3695 THPTimer', 'Int64');
4563: 3695 ***** File 175: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_ExtCtrls2.pas
4564: 3696 TShapeType', '( stRectangle, stSquare, stRoundRect, stRoundSquar
4565: 3697 TBevelStyle', '( bsLowered, bsRaised )';
4566: 3698 TBevelShape', '( bsBox, bsFrame, bsTopLine, bsBottomLine, bsLeft
4567: 3699 TBevelCut', '(bvNone, bvLowered, bvRaised, bvSpace)';
4568: 3700 TPanelBevel', 'TBevelCut';
4569: 3701 TFlowStyle', '( fsLeftRightTopBottom, fsRightLeftTopBottom, fsLe
4570: 3702 TSizeStyle', '( ssAbsolute, ssPercent, ssAuto )';
4571: 3703 TCellSpan', 'Integer');
4572: 3704 TExpandStyle', '( emAddRows, emAddColumns, emFixedSize )';
4573: 3705 TSectionEvent', 'Procedure ( Sender : TObject; ASection, AWidth: Integer)';
4574: 3706 NaturalNumber', 'Integer';
4575: 3707 TSplitterCanResizeEvent', 'Procedure ( Sender : TObject; var New
4576: 3708 TCanResizeEvent', 'TSplitterCanResizeEvent');
4577: 3709 TResizeStyle', '( rsNone, rsLine, rsUpdate, rsPattern )';
4578: 3710 TBandPaintOption', '( bpoGrabber, bpoFrame, bpoGradient, bpoRoundRect )';
4579: 3711 TBandPaintOptions', 'set of TBandPaintOption';
4580: 3712 TBandDrawingStyle', '( dsNormal, dsGradient )';
4581: 3713 TBandDragEvent', 'Procedure ( Sender : TObject; Control : TContr
4582: 3714 TBandInfoEvent', 'Procedure ( Sender : TObject; Control : TContr
4583: 3715 TBandMoveEvent', 'Procedure (Sender : TObject; Control : TControl; var ARect : TRect)';
4584: 3716 TBeginBandMoveEvent', 'Procedure ( Sender : TObject; Control : T
4585: 3717 TEndBandMoveEvent', 'Procedure ( Sender : TObject; Control : TControl)';
4586: 3718 TBandPaintEvent', 'Procedure ( Sender : TObject; Control : TCont
4587: 3719 TCornerEdge', '( ceNone, ceSmall, ceMedium, ceLarge )';
4588: 3720 TRowSize', 'Integer');
4589: 3721 TLabelPosition', '( lpAbove, lpBelow, lpLeft, lpRight )';
4590: 3722 TColorBoxStyles', '( cbStandardColors, cbExtendedColors, cbSyste
4591: 3723 TColorBoxStyle', 'set of TColorBoxStyles';
4592: 3724 TGetColorsEvent', 'Procedure ( Sender : TCustomColorBox; Items: TStrings)';
4593: 3725 TLBGetColorsEvent', 'Procedure ( Sender : TCustomColorListBox; Items: TStrings)';
4594: 3725 ***** File 176: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_GR32.pas
4595: 3726 PColor32', '^TColor32 // will not work';
4596: 3727 TColor32', 'Cardinal');
4597: 3728 PColor32Array', '^TColor32Array // will not work';
4598: 3729 TArrayOfColor32', 'array of TColor32';
4599: 3730 TColor32Component', '( ccBlue, ccGreen, ccRed, ccAlpha )';
4600: 3731 TColor32Components', 'set of TColor32Component';
4601: 3732 PColor32EntryArray', '^TColor32EntryArray // will not work';
4602: 3733 TArrayOfColor32Entry', 'array of TColor32Entry';
4603: 3734 PPalette32', 'TPalette32 // will not work';
4604: 3735 PFixed', 'TFixed // will not work';
4605: 3736 TFixed', 'Integer');
4606: 3737 PFixedArray', 'TFixedArray // will not work';
4607: 3738 PArrayofFixed', 'TArrayofFixed // will not work';
4608: 3739 TArrayofFixed', 'array of TFixed';
4609: 3740 PArrayofArrayofFixed', 'TArrayofArrayofFixed // will not work';
4610: 3741 TArrayofArrayofFixed', 'array of TArrayofFixed';
4611: 3742 PFloat', 'TFloat // will not work';

```



```

4612: 3743 TFloat', 'Single');
4613: 3744 PByteArray', '^TByteArray // will not work');
4614: 3745 PArrayOfByte', '^TArrayOfByte // will not work');
4615: 3746 TArrayOfByte', 'array of Byte');
4616: 3747 PWordArray', '^TWordArray // will not work');
4617: 3748 PArrayOfWord', '^TArrayOfWord // will not work');
4618: 3749 TArrayOfWord', 'array of Word');
4619: 3750 PIntegerArray', '^TIntegerArray // will not work');
4620: 3751 PArrayOfInteger', '^TArrayOfInteger // will not work');
4621: 3752 TArrayOfInteger', 'array of Integer');
4622: 3753 TArrayOfArrayOfInteger', 'array of TArrayOfInteger');
4623: 3754 PSingleArray', '^TSingleArray // will not work');
4624: 3755 PArrayOfSingle', '^TArrayOfSingle // will not work');
4625: 3756 TArrayOfSingle', 'array of Single');
4626: 3757 PFloatArray', '^TFloatArray // will not work');
4627: 3758 PArrayOfFloat', '^TArrayOfFloat // will not work');
4628: 3759 TArrayOfFloat', 'array of TFloat');
4629: 3760 PPointArray', '^TPointArray // will not work');
4630: 3761 PArrayOfPoint', '^TArrayOfPoint // will not work');
4631: 3762 TArrayOfPoint', 'array of TPoint');
4632: 3763 PArrayOfArrayOfPoint', '^TArrayOfArrayOfPoint // will not work');
4633: 3764 TArrayOfArrayOfPoint', 'array of TArrayOfPoint');
4634: 3765 PFloatPoint', '^TFloatPoint // will not work');
4635: 3766 TFloatPoint', 'record X : TFloat; Y : TFloat; end');
4636: 3767 PFloatPointArray', '^TFloatPointArray // will not work');
4637: 3768 PArrayOfFloatPoint', '^TArrayOfFloatPoint // will not work');
4638: 3769 TArrayOfFloatPoint', 'array of TFloatPoint');
4639: 3770 TArrayOfArrayOfFloatPoint', 'array of TArrayOfFloatPoint');
4640: 3771 PFixedPoint', '^TFixedPoint // will not work');
4641: 3772 TFixedPoint', 'record X : TFixed; Y : TFixed; end');
4642: 3773 PFixedPointArray', '^TFixedPointArray // will not work');
4643: 3774 PArrayOfFixedPoint', '^TArrayOfFixedPoint // will not work');
4644: 3775 TArrayOfFixedPoint', 'array of TFixedPoint');
4645: 3776 TArrayOfArrayOfFixedPoint', 'array of TArrayOfFixedPoint');
4646: 3777 TRectRounding', '( rrClosest, rrOutside, rrInside )';
4647: 3778 TDrawMode', '( dmOpaque, dmBlend, dmCustom, dmTransparent )';
4648: 3779 TCombineMode', '( cmBlend, cmMerge )';
4649: 3780 TWrapMode', '( wmClamp, wmRepeat, wmMirror )';
4650: 3781 TStretchFilter', '( sfNearest, sfDraft, sfLinear, sfCosine, sfSpline, sfLanczos, sfMitchell )';
4651: 3782 TPixelCombineEvent', 'Procedure ( F : TColor32; var B : TColor32; M : TColor32)';
4652: 3783 TAreaChangedEvent', 'Procedure (Sender: TObject; const Area: TRect; const Info : Cardinal)';
4653: 3784 TCustomBackendClass', 'class of TCustomBackend';
4654: 3785 TPixelAccessMode', '( pamUnsafe, pamSafe, pamWrap, pamTransparentEdge )';
4655: 3786 TCustomResamplerClass', 'class of TCustomResampler';
4656: 3786 ***** File 177: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_GR32_ExtImage.pas
4657: 3787 TRenderMode', '( rnmFull, rnmConstrained )';
4658: 3787 ***** File 178: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_GR32_Image.pas
4659: 3788 TPaintStageEvent', 'Procedure (Sender: TObject; Buffer: TBitmap32; StageNum: Cardinal)';
4660: 3789 TPaintStage', '^TPaintStage // will not work';
4661: 3790 TPaintStage', 'record DsgnTime: Boolean; RunTime: Boolean; Stage: Cardinal; Parameter : Cardinal;
end';
4662: 3791 TBitmapAlign', '( baTopLeft, baCenter, baTile, baCustom )';
4663: 3792 TScaleMode', '( smNormal, smStretch, smScale, smResize, smOptimal, smOptimalScaled )';
4664: 3793 TPaintBoxOptions', '( pboWantArrowKeys, pboAutoFocus )'; //set of
4665: 3794 TRepaintMode', '( rmFull, rmDirect, rmOptimizer )';
4666: 3795 TImgMouseEvent', 'Procedure ( Sender : TObject; Button : TMouseButton
4667: 3796 TImgMouseMoveEvent', 'Procedure ( Sender : TObject; Shift : TShi
4668: 3797 TPaintStageHandler', 'Procedure ( Dest : TBitmap32; StageNum : Integer)';
4669: 3798 TScrollBarVisibility', '( svAlways, svHidden, svAuto )';
4670: 3799 TSizeGripStyle', '( sgAuto, sgNone, sgAlways )';
4671: 3800 TBitmap32ItemClass', 'class of TBitmap32Item';
4672: 3800 ***** File 179: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_GR32_OrdinalMaps.pas
4673: 3801 TConversionType', '( ctRed, ctGreen, ctBlue, ctAlpha, ctUniformR
4674: 3801 ***** File 180: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_GR32_Rasterizers.pas
4675: 3802 TAssignColor', 'Procedure ( var Dst : TColor32; Src : TColor32)';
4676: 3803 PCombineInfo', '^TCombineInfo // will not work';
4677: 3804 TCombineInfo', 'record SrcAlpha : Integer; DrawMode : TDrawMode;
4678: 3805 TRasterizerClass', 'class of TRasterizer';
4679: 3805 ***** File 181: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_GR32_System.pas
4680: 3806 TCPUInstructionSet', '( ciMMX, ciEMMX, ciSSE, ciSSE2, ci3DNow, ci3DNowExt )';
4681: 3807 TCPUInstructionSet', '( ciDummy )';
4682: 3808 PCPUFeatures', '^TCPUFeatures // will not work';
4683: 3809 TCPUFeatures', 'set of TCPUInstructionSet';
4684: 3809 ***** File 182: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_IdIdentServer.pas
4685: 3810 TIdIdentQueryEvent', 'Procedure ( AThread : TIdPeerThread; AServ
4686: 3811 TIdIdentErrorType', '( ieInvalidPort, ieNoUser, ieHiddenUser, ieUnknownError )';
4687: 3811 ***** File 183: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_IdIrcServer.pas
4688: 3812 TIdIrcGetEvent', 'Procedure ( Thread : TIdPeerThread)';
4689: 3813 TIdIrcOtherEvent', 'Procedure ( Thread : TIdPeerThread; Command, Parm : String)';
4690: 3814 TIdIrcOneParmEvent', 'Procedure ( Thread : TIdPeerThread; Parm: String)';
4691: 3815 TIdIrcTwoParmEvent', 'Procedure ( Thread : TIdPeerThread; Parm1, Parm2: String)';
4692: 3816 TIdIrcThreeParmEvent', 'Procedure ( Thread : TIdPeerThread; Parm1, Parm2, Parm3 : String)';
4693: 3817 TIdIrcFiveParmEvent', 'Procedure ( Thread : TIdPeerThread; Parm1
4694: 3818 TIdIrcUserEvent', 'Procedure ( Thread : TIdPeerThread; Username,
4695: 3819 TIdIrcServerEvent', 'Procedure ( Thread : TIdPeerThread; ServerN
4696: 3819 ***** File 184: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_IdMessageCoder.pas
4697: 3820 TIdMessageCoderPartType', '( mcptUnknown, mcptText, mcptAttachme
4698: 3821 TIdMessageEncoderClass', 'class of TIdMessageEncoder';
4699: 3821 ***** File 185: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_IdMessageCollection.pas

```

```

4700: 3822 TIdMessageItems', 'class of TIdMessageItem');
4701: 3822 ***** File 186: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_IdRexec.pas
4702: 3823 TIdRexecCommandEvent', 'Procedure ( AThread : TIdPeerThread; ASt'
4703: 3823 ***** File 187: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_IdUDPServer.pas
4704: 3824 TUDPReadEvent', 'Procedure ( Sender : TObject; AData : TStream; '
4705: 3824 ***** File 188: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JclAppInst.pas
4706: 3825 TJclAppInstDataKind', 'Integer');
4707: 3825 ***** File 189: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JclComplex.pas
4708: 3826 TComplexKind', '( crRectangular, crPolar )';
4709: 3827 TCoords', 'record X : Float; Y : Float; R : Float; Theta : Float; end';
4710: 3828 TRectCoord', 'record X : Float; Y : Float; end';
4711: 3828 ***** File 190: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JclLocales.pas
4712: 3829 TJclLocalesDays', 'Integer');
4713: 3830 TJclLocalesMonths', 'Integer');
4714: 3831 TJclLocalesDateFormats', '( ldShort, ldLong, ldYearMonth )';
4715: 3832 TJclLocalesKind', '( lkInstalled, lkSupported )';
4716: 3833 TJclKeybLayoutFlag', '( klReorder, klUnloadPrevious, klSetForPro'
4717: 3834 TJclKeybLayoutFlags', 'set of TJclKeybLayoutFlag');
4718: 3834 ***** File 191: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JclMIDI.pas
4719: 3835 TMIDIChannel', 'Integer');
4720: 3836 TMIDIDataByte', 'Integer');
4721: 3837 TMIDIDataWord', 'Integer');
4722: 3838 TMIDIVelocity', 'TMIDIDataByte');
4723: 3839 TMIDIKey', 'TMIDIDataByte');
4724: 3840 TMIDINote', 'TMIDIKey');
4725: 3841 TMIDINotes', 'set of integer');
4726: 3841 ***** File 192: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JclNTFS.pas
4727: 3842 TFileCompressionState', '( fcNoCompression, fcDefaultCompression, fcLZNT1Compression )';
4728: 3843 TNtfsAllocRanges', 'record Entries : Integer; Data : PFileAlloca'
4729: 3844 TOpLock', '( olExclusive, olReadOnly, olBatch, olFilter )';
4730: 3845 TStreamId', '( siInvalid, siStandard, siExtendedAttribute, siSec'
4731: 3846 TStreamIds', 'set of TStreamId');
4732: 3847 TInternalFindStreamData', 'record FileHandle : THandle; Context '
4733: 3848 TFindStreamData', 'record Internal : TInternalFindStreamData; At'
4734: 3849 TNtfsHardLinkInfo', 'record LinkCount : Cardinal; FileIndex : Int64; end';
4735: 3849 ***** File 193: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JclPeImage.pas
4736: 3850 TJclSmartCompOption', '( scSimpleCompare, scIgnoreCase )';
4737: 3851 TJclSmartCompOptions', 'set of TJclSmartCompOption';
4738: 3852 TJclPeImageClass', 'class of TJclPeImage';
4739: 3853 TJclPeImportSort', '( isName, isOrdinal, isHint, isLibImport )';
4740: 3854 TJclPeImportLibSort', '( ilName, ilIndex )';
4741: 3855 TJclPeImportKind', '( ikImport, ikDelayImport, ikBoundImport )';
4742: 3856 TJclPeResolveCheck', '( icNotChecked, icResolved, icUnresolved )';
4743: 3857 TJclPeLinkerProducer', '( lrBorland, lrMicrosoft )';
4744: 3858 TJclPeExportSort', '( esName, esOrdinal, esHint, esAddress, esForwarded, esAddrOrFwd, esSection )';
4745: 3859 TJclPeResourceKind', '( rtUnknown0, rtCursorEntry, rtBitmap, rti'
4746: 3860 TJclPeRelocation', 'record Address: Word; RelocType: Byte; VirtualAddress : DWORD; end';
4747: 3861 TJclPeHeader', '( JclPeHeader_Signature, JclPeHeader_Machine, Jc'
4748: 3862 TJclLoadConfig', '( JclLoadConfig_Characteristics, JclLoadConfig'
4749: 3863 TJclPeFileProperties', 'record Size : DWORD; CreationTime : TDat'
4750: 3864 TJclPeImageStatus', '( stNotLoaded, stOk, stNotPE, stNotFound, stError )';
4751: 3865 TJclPeNameSearchOption', '( seImports, seDelayImports, seBoundImports, seExports )';
4752: 3866 TJclPeNameSearchOptions', 'set of TJclPeNameSearchOption';
4753: 3867 TJclPeNameSearchNotifyEvent', 'Procedure ( Sender : TObject; PeI'
4754: 3868 TJclPeNameSearchFoundEvent', 'Procedure ( Sender : TObject; cons'
4755: 3869 TJclRebaseImageInfo', 'record OldImageSize : DWORD; OldImageBase'
4756: 3870 TJclBorUmSymbolKind', '(skData, skFunction, skConstructor, skDestructor, skRTTI, skVTable )';
4757: 3871 TJclBorUmSymbolModifier', '( smQualified, smLinkProc )';
4758: 3872 TJclBorUmSymbolModifiers', 'set of TJclBorUmSymbolModifier';
4759: 3873 TJclBorUmDescription', 'record Kind : TJclBorUmSymbolKind; Modifiers : TJclBorUmSymbolModifiers;
end';
4760: 3874 TJclBorUmResult', '( urOk, urNotMangled, urMicrosoft, urError )';
4761: 3875 TJclPeUmResult', '( umNotMangled, umBorland, umMicrosoft )';
4762: 3875 ***** File 194: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JclPrint.pas
4763: 3876 PWordArray', '^TWordArray // will not work';
4764: 3876 ***** File 195: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JclRegistry.pas
4765: 3877 TExecKind', '( ekMachineRun, ekMachineRunOnce, ekUserRun, ekUser'
4766: 3877 ***** File 196: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JclSchedule.pas
4767: 3878 TScheduleRecurringKind', '( srkOneShot, srkDaily, srkWeekly, srk'
4768: 3879 TScheduleEndKind', '( sekNone, sekDate, sekTriggerCount, sekDayCount )';
4769: 3880 TScheduleWeekDay', '( swdMonday, swdTuesday, swdWednesday, swdTh'
4770: 3881 TScheduleWeekDays', 'set of TScheduleWeekDay';
4771: 3882 TScheduleIndexKind', '( sikNone, sikDay, sikWeekDay, sikWeekendD'
4772: 3882 ***** File 197: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JclSvcCtrl.pas
4773: 3883 TJclServiceType', '( stKernelDriver, stFileSystemServiceDriver, stAdapt'
4774: 3884 TJclServiceTypes', 'set of TJclServiceType';
4775: 3885 TJclServiceState', '( ssUnknown, ssStopped, ssStartPending, ssSt'
4776: 3886 TJclServiceStartType', '( sstBoot, sstSystem, sstAuto, sstDemand, sstDisabled )';
4777: 3887 TJclServiceErrorControlType', '( ectIgnore, ectNormal, ectSevere, ectCritical )';
4778: 3888 TJclServiceControlAccepted', '( caStop, caPauseContinue, caShutdown )';
4779: 3889 TJclServiceControlAccepted', 'set of TJclServiceControlAccepted';
4780: 3890 TJclServiceSortOrderType', '( sotServiceName, sotDisplayName, so'
4781: 3891 LPSERVICE_DESCRIPTIONA', '^SERVICE_DESCRIPTIONA // will not work';
4782: 3892 _SERVICE_DESCRIPTIONA', 'record lpDescription : string; end';
4783: 3893 SERVICE_DESCRIPTIONA', '_SERVICE_DESCRIPTIONA';
4784: 3894 TServiceDescriptionA', 'SERVICE_DESCRIPTIONA';
4785: 3895 PServiceDescriptionA', 'LPSERVICE_DESCRIPTIONA';
4786: 3895 ***** File 198: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JclWinMidi.pas
4787: 3896 TStereoChannel', '( scLeft, scRight )';

```

```

4788: 3896 ***** File 199: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvAlarms.pas
4789: 3897 TJvTriggerKind', '( tkOneShot, tkEachSecond, tkEachMinute, tkEac'
4790: 3898 TJvOnAlarm', 'Procedure ( Sender : TObject; const Alarm : TJvAla'
4791: 3899 ***** File 200: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvAnalogClock.pas
4792: 3899 TJvNotifyTime', 'Procedure (Sender: TObject; Hour,Min,Sec: Integer)';
4793: 3900 TJvHourStyle', '( hsLine, hsCircle, hsNumber, hsNumberInCircle )';
4794: 3901 TJvHourMarks', '( hmNone, hmFour, hmAll )';
4795: 3901 ***** File 201: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvAnimatedImage.pas
4796: 3902 TGlyphOrientation', '( goHorizontal, goVertical )';
4797: 3902 ***** File 202: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvAppCommand.pas
4798: 3903 TJvAppCommandDevice', '( acdKey, acdMouse, acdOEM )';
4799: 3904 TJvAppCommandEvent', 'Procedure ( Handle : THandle; Cmd : WORD; '
4800: 3904 ***** File 203: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvAppInst.pas
4801: 3905 TJvAppInstDataKind', 'TJclAppInstDataKind';
4802: 3906 TInstanceChangeEvent', 'Procedure ( Sender : TObject; ProcessId: Cardinal)';
4803: 3907 TUserNotifyEvent', 'Procedure ( Sender : TObject; Param : Integer)';
4804: 3908 TDataAvailableEvent', 'Procedure ( Sender : TObject; Kind : TJvA'
4805: 3909 TCmdLineReceivedEvent', 'Procedure ( Sender : TObject; CmdLine: TStrings)';
4806: 3909 ***** File 204: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvBDESQLScript.pas
4807: 3910 TOnScriptProgress', 'Procedure ( Sender : TJvBDESQLScript; var C'
4808: 3910 ***** File 205: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvBrowseFolder.pas
4809: 3911 TJvBrowsableObjectClass', '( ocFolders, ocNonFolders, ocIncludeH'
4810: 3912 TJvBrowsableObjectClasses', 'set of TJvBrowsableObjectClass';
4811: 3913 TJvBrowseAcceptChange', 'Procedure ( Sender : TObject; const New'
4812: 3914 TJvShouldShowEvent', 'Procedure (Sender: TObject; const Item: string; var DoShow : Boolean)';
4813: 3915 TJvGetEnumFlagsEvent', 'Procedure ( Sender : TObject; const AFol'
4814: 3916 TJvDirChange', 'Procedure ( Sender : TObject; const Directory:string)';
4815: 3917 TJvValidateFailedEvent', 'Procedure ( Sender : TObject; const AE'
4816: 3918 TFromDirectory', '( fdNoSpecialFolder, fdRootFolder, fdRecycleBi'
4817: 3919 TJvFolderPos', '( fpDefault, fpScreenCenter, fpFormCenter, fpTop'
4818: 3920 TOptionsDirectory', '( odBrowseForComputer, odOnlyDirectory, odO'
4819: 3921 TOptionsDir', 'set of TOptionsDirectory';
4820: 3921 ***** File 206: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvChart.pas
4821: 3922 TJvChartKind', '( ckChartNone, ckChartLine, ckChartBar, ckCharts'
4822: 3923 TJvChartPenMarkerKind', '( pmkNone, pmkDiamond, pmkCircle, pmkSquare, pmkCross )';
4823: 3924 TJvChartGradientDirection', '( grNone, grUp, grDown, grLeft, grRight )';
4824: 3925 TJvChartLegend', '( clChartLegendNone, clChartLegendRight, clChartLegendBelow )';
4825: 3926 TJvChartDataArray', 'array of array of Double';
4826: 3927 TJvChartCaptionPosition', '( cpMarker, cpXAxisBottom, cpXAxisTop, cpTitleArea )';
4827: 3928 TJvChartPaintEvent', 'Procedure ( Sender : TJvChart; ACanvas : TCanvas)';
4828: 3929 TJvChartEvent', 'Procedure ( Sender : TJvChart)';
4829: 3930 TJvChartFloatingMarkerDragEvent', 'Procedure ( Sender : TJvChart'
4830: 3931 TJvChartClickEvent', 'Procedure ( Sender : TJvChart; Button : TM'
4831: 3931 ***** File 207: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvConverter.pas
4832: 3932 TDataType', '( dtString, dtInteger, dtFloat, dtDateTime, dtDate,'
4833: 3933 TTimeFormat', '( tfHHMMSS, tfHMMSS, tfHHMM, tfHMM )';
4834: 3933 ***** File 208: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvDataConv.pas
4835: 3934 TDataTypeJ', '( dtString, dtInteger, dtFloat, dtDateTime, dtDate,'
4836: 3935 TTimeFormat', '( tfHHMMSS, tfHMMSS, tfHHMM, tfHMM )';
4837: 3935 ***** File 209: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvDBGridExport.pas
4838: 3936 TExportDestination', '( edFile, edClipboard )';
4839: 3937 TExportSeparator', '( esTab, esSemiColon, esComma, esSpace, esPipe )';
4840: 3938 TWordOrientation', '( woPortrait, woLandscape )';
4841: 3939 TOleServerClose', '( scNever, scNewInstance, scAlways )';
4842: 3940 TRecordColumn', 'record Visible : Boolean; Exportable : Boolean;
4843: 3941 TJvExportProgressEvent', 'Procedure ( Sender : TObject; Min, Max'
4844: 3942 TJvCustomDBGridExportClass', 'class of TJvCustomDBGridExport';
4845: 3943 TOnExportProgress', 'Procedure ( Progress : Integer)';
4846: 3943 ***** File 210: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvDBQBE.pas
4847: 3944 TCheckType', '( ctNone, ctCheck, ctCheckPlus, ctCheckDesc, ctChe'
4848: 3944 ***** File 211: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvDBSecur.pas
4849: 3945 TCheckUserEvent', 'Function (UsersTable: TTable;const Password: string): Boolean);
4850: 3945 ***** File 212: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvDBUltimGrid.pas
4851: 3946 TSortField', 'record Name : string; Order : Boolean; end';
4852: 3947 TSortFields', 'array of TSortField';
4853: 3948 TIndexNotFoundEvent', 'Procedure ( Sender : TJvDBUltimGrid; Fiel'
4854: 3949 TUserSortEvent', 'Procedure ( Sender : TJvDBUltimGrid; var Field'
4855: 3950 TRestoreGridPosEvent', 'Procedure ( Sender : TJvDBUltimGrid; Sav'
4856: 3951 TCheckIfValidSortFieldEvent', 'Function ( Sender : TJvDBUltimGri'
4857: 3952 TGetSortFieldNameEvent', 'Procedure ( Sender : TJvDBUltimGrid; v'
4858: 3953 TSortWith', '( swIndex, swFields, swUserFunc, swWhere )';
4859: 3953 ***** File 213: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvDice.pas
4860: 3954 TJvDiceValue', 'Integer';
4861: 3954 ***** File 214: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvDualList.pas
4862: 3955 TJvDualListCustomizeEvent', 'Procedure ( Sender : TObject; Form: TCustomForm)';
4863: 3955 ***** File 215: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvForth.pas
4864: 3956 TToken', '( dfoError, dfoNop, dfoIf, dfoElse, dfoEndIf, dfoRepea'
4865: 3957 TProcVar', 'Procedure';
4866: 3958 TOnGetVariable', 'Procedure ( Sender : TObject; Symbol : string;
4867: 3959 TOnSetVariable', 'Procedure ( Sender : TObject; Symbol : string;
4868: 3960 TOnGetSystem', 'Procedure ( Sender : TObject; Symbol, Prompt : s'
4869: 3961 TOnSetSystem', 'Procedure ( Sender : TObject; Symbol : string; V'
4870: 3962 TOnInclude', 'Procedure ( Sender : TObject; IncludeFile : string'
4871: 3962 ***** File 216: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvgDigits.pas
4872: 3963 TJvgSpecialSymbol', '( ssyNone, ssyColon, ssySlash, ssyBackslash)';
4873: 3964 TJvgGraphDigitsElem', '( dlT, dlC, dlB, dlTL, dlTR, dlBL, dlBR,dlDOT )';
4874: 3965 TJvgGraphDigitsElemSet', 'set of TJvgGraphDigitsElem';
4875: 3965 ***** File 217: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvgExport.pas
4876: 3966 TOnExportProgress', 'Procedure ( Progress : Integer)';

```

```

4877: 3966 ***** File 218: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvImageWindow.pas
4878: 3967 TJvMargin', 'Integer');
4879: 3968 TJvPositive', 'Integer');
4880: 3968 ***** File 219: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvJanTreeView.pas
4881: 3969 TGetVarEvent', 'Procedure ( Sender : TObject; VarName : string; '
4882: 3970 TParseErrorEvent', 'Procedure ( Sender : TObject; ParseError : Integer)');
4883: 3971 ErrorRange', 'Integer');
4884: 3972 TokenTypes', '( ttPlus, ttMinus, ttTimes, ttDivide, ttExpo, ttOP'
4885: 3973 TokenRec', 'record State : Byte; Value : Extended; end');
4886: 3974 TStackTop', 'Integer');
4887: 3974 ***** File 220: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvJoystick.pas
4888: 3975 TJoyCap', '( joHasZCoordinate, joHasRudder, joHasUCoordinate, jo'
4889: 3976 TJoyCaps', 'set of TJoyCap');
4890: 3977 TJoyButtonDown', 'Procedure ( Sender : TObject; X, Y : Integer; '
4891: 3978 TJoyMove', 'Procedure ( Sender : TObject; X, Y : Integer; ButlPr'
4892: 3979 TJoyZMove', 'Procedure ( Sender : TObject; Z : Integer; ButlPres'
4893: 3980 TJoyErrorMsg', 'Procedure ( Sender : TObject; code : Integer; Msg: string)');
4894: 3980 ***** File 221: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvMailSlots.pas
4895: 3981 TOnNewMessage', 'Procedure ( Sender : TObject; MessageText : string)');
4896: 3981 ***** File 222: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvObjStr.pas
4897: 3982 TDestroyEvent', 'Procedure ( Sender, AObject : TObject)');
4898: 3983 TObjectSortCompare', 'Function ( const S1, S2 : string; Item1, I'
4899: 3984 TItemSortCompare', 'Function ( Item1, Item2 : TCollectionItem) : '
4900: 3984 ***** File 223: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvPaintFX.pas
4901: 3985 TLightBrush', '( lbBrightness, lbContrast, lbSaturation, lbFishe'
4902: 3985 ***** File 224: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvPicClip.pas
4903: 3986 TCellRange', 'Integer');
4904: 3986 ***** File 225: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvSearchFiles.pas
4905: 3987 TJvAttrFlagKind', '( tsMustBeSet, tsDontCare, tsMustBeUnSet)');
4906: 3988 TJvDirOption', '( doExcludeSubDirs, doIncludeSubDirs, doExcludeI'
4907: 3989 TJvSearchOption', '( soAllowDuplicates, soCheckRootDirValid, soE'
4908: 3990 TJvSearchOptions', 'set of TJvSearchOption');
4909: 3991 TJvSearchType', '( stAttribute, stFileMask, stFileMaskCaseSensit'
4910: 3992 TJvSearchTypes', 'set of TJvSearchType');
4911: 3993 TJvFileSearchEvent', 'Procedure ( Sender : TObject; const AName : string)');
4912: 3994 TJvSearchFilesError', 'Procedure (Sender: TObject; var Handled: Boolean)');
4913: 3995 TJvCheckEvent', 'Procedure ( Sender : TObject; var Result : Boolean)');
4914: 3996 TJvErrorResponse', '( erAbort, erIgnore, erRaise)');
4915: 3996 ***** File 226: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvSimLogic.pas
4916: 3997 TJvGateStyle', '( jgsDI, jgsDO)');
4917: 3998 TJvLogicFunc', '( jlfAND, jlfOR, jlfNOT)');
4918: 3999 TJvGate', 'record Style : TJvGateStyle; State : Boolean; Active '
4919: 4000 TJvConMode', '( jcmTL, jcmTR, jcmBR, jcmBL)');
4920: 4001 TJvConPos', '( jcpTL, jcpTR, jcpBR, jcpBL)');
4921: 4002 TJvConShape', '( jcsTLBR, jcsTRBL)');
4922: 4002 ***** File 227: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvSimPID.pas
4923: 4003 TJvSymFunc', '( sfPid, sfAdd, sfCompare, sfRamp, sfMul)');
4924: 4003 ***** File 228: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvSimPIDLinker.pas
4925: 4004 TPIDS', 'array of TJvSimPID');
4926: 4004 ***** File 229: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvSoundControl.pas
4927: 4005 TBalance', 'Integer');
4928: 4005 ***** File 230: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvSQLS.pas
4929: 4006 TOnScriptProgress', 'Procedure ( Sender : TJvASQLScript; var Can'
4930: 4006 ***** File 231: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvStarfield.pas
4931: 4007 TJvStars', 'record X : Integer; Y : Integer; Color : TColor; Spe'
4932: 4007 ***** File 232: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvSwitch.pas
4933: 4008 TTextPos', '( tpNone, tpLeft, tpRight, tpAbove, tpBelow)');
4934: 4009 TSwitchBitmaps', 'set of Boolean');
4935: 4009 ***** File 233: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvTFUtils.pas
4936: 4010 TJvTFVisibleScrollBars', 'set of ( vsbHorz, vsbVert)');
4937: 4011 TTFDayOfWeek', '( dowSunday, dowMonday, dowTuesday, dowWednesday'
4938: 4012 TTFDaysOfWeek', 'set of TTFDayOfWeek);
4939: 4013 TJvTFVAlignment', '( vaTop, vaCenter, vaBottom)');
4940: 4014 TJvTFDirection', '( dirUp, dirDown, dirLeft, dirRight)');
4941: 4014 ***** File 234: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvTimerLst.pas
4942: 4015 TAllTimersEvent', 'Procedure (Sender: TObject; Handle: Longint)');
4943: 4015 ***** File 235: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvUnicodeCanvas.pas
4944: 4016 TJvExtTextOutOptionsKind', '( etoClipped, etoOpaque)');
4945: 4017 TJvExtTextOutOptions', 'set of TJvExtTextOutOptionsKind);
4946: 4017 ***** File 236: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvUrlGrabbers.pas
4947: 4018 TJvProxyMode', '( pmNoProxy, pmSysConfig, pmManual)');
4948: 4019 TJvFtpDownloadMode', '( hmBinary, hmAscii)');
4949: 4019 ***** File 237: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvValidators.pas
4950: 4020 TJvBaseValidatorClass', 'class of TJvBaseValidator');
4951: 4021 TJvValidateCompareOperator', '( vcoLessThan, vcoLessOrEqual, vco'
4952: 4022 TJvCustomValidateEvent', 'Procedure ( Sender : TObject; ValueToV'
4953: 4023 TJvValidateFailEvent', 'Procedure ( Sender : TObject; BaseValida'
4954: 4023 ***** File 238: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvWinDialogs.pas
4955: 4024 TShellLinkInfo', 'record PathName : string; Arguments : string; '
4956: 4025 TSpecialFolderInfo', 'record Name : string; ID : Integer; end);
4957: 4026 TOpenFileNameEx', 'record lStructSize : DWORD; hWndOwner : HWND; '
4958: 4027 TShellObjectType', '( sdPathObject, sdPrinterObject)');
4959: 4028 TShellObjectTypes', 'set of TShellObjectType);
4960: 4029 TJvFormatDriveKind', '( ftQuick, ftStandard, ftBootable)');
4961: 4030 TJvDriveCapacity', '( dcDefault, dcSize360kB, dcSize720kB)');
4962: 4031 TJvFormatDriveError', '( errParams, errSysError, errAborted, err'
4963: 4032 TJvFormatDriveErrorEvent', 'Procedure ( Sender : TObject; Error '
4964: 4033 TJvCplInfo', 'record Icon : TIcon; Name : string; Info : string; lData : Longint; end);
4965: 4033 ***** File 239: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvXmlTree.pas

```



```

4966: 4034 TJvXMLValueType', '( xvtString, xvtCDATA )';
4967: 4035 TJvXMLFilterOperator', '( xfoNOP, xfoEQ, xfoIEQ, xfoNE, xfoINE, '
4968: 4035 ***** File 240: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_JvXPCore.pas
4969: 4036 TJvXPBoundLine', '( blLeft, blTop, blRight, blBottom )';
4970: 4037 TJvXPBoundLines', 'set of TJvXPBoundLine';
4971: 4038 TJvXPControlStyle1', '( dsRedrawCaptionChanged, dsRedrawBo
4972: 4039 TJvXPDrawState1', '( dsDefault, dsHighlight, dsClicked, dsFocused )';
4973: 4040 TJvXPDrawState', 'set of TJvXPDrawState1';
4974: 4041 TJvXPControlStyle', 'set of TJvXPControlStyle1';
4975: 4042 TJvXPDrawState', 'set of ( dsDefault, dsHighlight, dsClicked, dsFocused )';
4976: 4043 TJvXPGlyphLayout', '( glBottom, glCenter, glTop )';
4977: 4044 TJvXPTheme', '( WindowsXP, OfficeXP )';
4978: 4045 TJvXPGradientColors', 'Integer';
4979: 4046 TJvXPGradientStyle', '( gsLeft, gsTop, gsRight, gsBottom )';
4980: 4046 ***** File 241: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_LongIntList.pas
4981: 4047 PLongIntArray', 'TLongIntArray // will not work';
4982: 4047 ***** File 242: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_OleAuto.pas
4983: 4048 TAutoClass', 'class of TAutoObject';
4984: 4049 TAutoClassInstantiating', '( acInternal, acSingleInstance, acMultiInstance )';
4985: 4050 TAutoClassInfo', 'record AutoClass : TAutoClass; ProgID : string'
4986: 4051 TStartMode', '( smStandalone, smAutomation, smRegServer, smUnregServer )';
4987: 4052 TLastReleaseEvent', 'Procedure ( var Shutdown : Boolean)';
4988: 4053 TIID', 'TGUID';
4989: 4054 TCLSID', 'TGUID';
4990: 4054 ***** File 243: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_Registry.pas
4991: 4055 TRegKeyInfo', 'record NumSubKeys : Integer; MaxSubKeyLen : Integer'
4992: 4056 TRegDataType', '(rdUnknown, rdString, rdExpandString, rdInteger, rdBinary)';
4993: 4057 TRegDataInfo', 'record RegData : TRegDataType; DataSize : Integer; end';
4994: 4057 ***** File 244: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_ShellAPI.pas
4995: 4058 HDROP', 'Longint';
4996: 4059 PPWideChar', '^PWideChar // will not work';
4997: 4060 PDragInfoA', '^_DRAGINFOA // will not work';
4998: 4061 PDragInfoW', '^_DRAGINFOW // will not work';
4999: 4062 PDragInfo', 'PDragInfoA';
5000: 4063 _DRAGINFOA', 'record uSize : UINT; pt : TPoint; fnc : BOOL; lpFi'
5001: 4064 TDragInfoA', '_DRAGINFOA';
5002: 4065 LPDRAGINFOA', 'PDragInfoA';
5003: 4066 _DRAGINFOW', 'record uSize : UINT; pt : TPoint; fnc : BOOL; lpFi'
5004: 4067 TDragInfoW', '_DRAGINFOW';
5005: 4068 LPDRAGINFOW', 'PDragInfoW';
5006: 4069 _DRAGINFO', '_DRAGINFOA';
5007: 4070 PAppBarData', '^AppBarData // will not work';
5008: 4071 _AppBarData', 'record cbSize : DWORD; hWnd : HWND; uCallbackMess'
5009: 4072 TAppBarData', '_AppBarData';
5010: 4073 APPBARDATA', '_AppBarData';
5011: 4074 FILEOP_FLAGS', 'Word';
5012: 4075 PRINTEROP_FLAGS', 'Word';
5013: 4076 PSHFileOpStructA', '^TSHFileOpStructA // will not work';
5014: 4077 PSHFileOpStructW', '^TSHFileOpStructW // will not work';
5015: 4078 PSHFileOpStruct', 'PSHFileOpStructA';
5016: 4079 _SHFILEOPSTRUCTA', 'record Wnd : HWND; wFunc : UINT; pFrom : PAn'
5017: 4080 _SHFILEOPSTRUCTW', 'record Wnd : HWND; wFunc : UINT; pFrom : PWi'
5018: 4081 _SHFILEOPSTRUCT', '_SHFILEOPSTRUCTA';
5019: 4082 TSHFileOpStructA', '_SHFILEOPSTRUCTA';
5020: 4083 TSHFileOpStructW', '_SHFILEOPSTRUCTW';
5021: 4084 TSHFileOpStruct', 'TSHFileOpStructA';
5022: 4085 SHFILEOPSTRUCTA', '_SHFILEOPSTRUCTA';
5023: 4086 SHFILEOPSTRUCTW', '_SHFILEOPSTRUCTW';
5024: 4087 SHFILEOPSTRUCT', 'SHFILEOPSTRUCTA';
5025: 4088 PSHNameMappingA', '^TSHNameMappingA // will not work';
5026: 4089 PSHNameMappingW', '^TSHNameMappingW // will not work';
5027: 4090 PSHNameMapping', 'PSHNameMappingA';
5028: 4091 _SHNAMEMAPPINGA', 'record pszOldPath : PAnsiChar; pszNewPath : P'
5029: 4092 _SHNAMEMAPPINGW', 'record pszOldPath : PWideChar; pszNewPath : P'
5030: 4093 _SHNAMEMAPPING', '_SHNAMEMAPPINGA';
5031: 4094 TSHNameMappingA', '_SHNAMEMAPPINGA';
5032: 4095 TSHNameMappingW', '_SHNAMEMAPPINGW';
5033: 4096 TSHNameMapping', 'TSHNameMappingA';
5034: 4097 SHNAMEMAPPINGA', '_SHNAMEMAPPINGA';
5035: 4098 SHNAMEMAPPINGW', '_SHNAMEMAPPINGW';
5036: 4099 SHNAMEMAPPING', 'SHNAMEMAPPINGA'; *)
5037: 4100 PShellExecuteInfoA', '^TShellExecuteInfoA // will not work';
5038: 4101 PShellExecuteInfoW', '^TShellExecuteInfoW // will not work';
5039: 4102 PShellExecuteInfo', 'PShellExecuteInfoA';
5040: 4103 _SHELLEXECUTEINFOA', 'record cbSize : DWORD; fMask : ULONG; Wnd '
5041: 4104 _SHELLEXECUTEINFOW', 'record cbSize : DWORD; fMask : ULONG; Wnd '
5042: 4105 _SHELLEXECUTEINFO', '_SHELLEXECUTEINFOA';
5043: 4106 TShellExecuteInfoA', '_SHELLEXECUTEINFOA';
5044: 4107 TShellExecuteInfoW', '_SHELLEXECUTEINFOW';
5045: 4108 TShellExecuteInfo', 'TShellExecuteInfoA';
5046: 4109 SHELLEXECUTEINFOA', '_SHELLEXECUTEINFOA';
5047: 4110 SHELLEXECUTEINFOW', '_SHELLEXECUTEINFOW';
5048: 4111 SHELLEXECUTEINFO', 'SHELLEXECUTEINFOA';
5049: 4112 PNotifyIconDataA', '^TNotifyIconDataA // will not work';
5050: 4113 PNotifyIconDataW', '^TNotifyIconDataW // will not work';
5051: 4114 PNotifyIconData', 'PNotifyIconDataA';
5052: 4115 _NOTIFYICONDATA', '_NOTIFYICONDATAA';
5053: 4116 TNotifyIconDataA', '_NOTIFYICONDATAA';
5054: 4117 TNotifyIconDataW', '_NOTIFYICONDATAW';

```

```

5055: 4118 TNotifyIconData', 'TNotifyIconDataA');
5056: 4119 NOTIFYICONDATAA', '_NOTIFYICONDATAA');
5057: 4120 NOTIFYICONDATAW', '_NOTIFYICONDATAW');
5058: 4121 NOTIFYICONDATA', 'NOTIFYICONDATAA');*)
5059: 4122 PSHFileInfoA', '^TSHFileInfoA // will not work');
5060: 4123 PSHFileInfoW', '^TSHFileInfoW // will not work');
5061: 4124 PSHFileInfo', 'PSHFileInfoA');
5062: 4125 _SHFILEINFO', '_SHFILEINFOA');
5063: 4126 TSHFileInfoA', '_SHFILEINFOA');
5064: 4127 TSHFileInfoW', '_SHFILEINFOW');
5065: 4128 TSHFileInfo', 'TSHFileInfoA');
5066: 4129 SHFILEINFOA', '_SHFILEINFOA');
5067: 4130 SHFILEINFOW', '_SHFILEINFOW');
5068: 4131 SHFILEINFO', 'SHFILEINFOA');*)
5069: 4131 ***** File 245: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_StAstro.pas
5070: 4132 TStTwilight', '( ttCivil, ttNautical, ttAstronomical )';
5071: 4133 TStRiseSetRec', 'record ORise : TStTime; OSet : TStTime; end';
5072: 4134 TStPosRec', 'record RA : Double; RC : Double; end';
5073: 4135 TStLunarRecord', 'record T: array[0..1] of TStDateTimeRec; end';
5074: 4136 TStLunarRecord', 'T: TStDateTimeRec; end';
5075: 4137 TStSunXYZRec', 'record SunX : double; SunY : double; SunZ : doub
5076: 4138 TStPhaseRecord', 'record NMDate : double; FQDate : double; FMDat'
5077: 4139 TStDLSDateRec', 'record Starts : TStDate; Ends : TStDate; end';
5078: 4140 TStMoonPosRec', 'record RA : double; DC : double; Phase : double'
5079: 4140 ***** File 246: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_StBarC.pas
5080: 4141 TStBarKind', '(bkSpace, bkBar, bkThreeQuarterBar, bkHalfBar, bkGuard, bkSupplement, bkBlankSpace)';
5081: 4142 TStBarKindSet', 'set of TStBarKind';
5082: 4143 TStBarCodeType', '( bcUPC_A, bcUPC_E, bcEAN_8, bcEAN_13, bcInter'
5083: 4144 TStCode128CodeSubset', '( csCodeA, csCodeB, csCodeC )';
5084: 4144 ***** File 247: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_StBarPN.pas
5085: 4145 TStPNBarCodeDims', 'record PixPerBar : Longint; PixPerSpace : Lo'
5086: 4146 TStPNBarCodeRes', 'record XRes : Longint; YRes : Longint; end';
5087: 4146 ***** File 248: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_StBase.pas
5088: 4147 TStLineTerminator', '( ltNone, ltCR, ltLF, ltCRLF, ltOther )';
5089: 4148 TStHwnd', 'Integer';
5090: 4149 TStHwnd', 'HWND';
5091: 4150 TUntypedCompareFunc', 'function(const Ell, El2: byte): Integer;';
5092: 4151 EStExceptionClass', 'class of EStException';
5093: 4152 PDouble', '^Double // will not work';
5094: 4153 TStFloat', 'Extended';
5095: 4154 TStFloat', 'Real';
5096: 4155 TStNodeClass', 'class of TStNode';
5097: 4156 TStCompareEvent', 'Procedure ( Sender : TObject; Data1, Data2 : '
5098: 4157 TStDisposeDataEvent', 'Procedure ( Sender : TObject; Data : __Pointer)';
5099: 4158 TStLoadDataEvent', 'Procedure ( Sender : TObject; Reader : TReader; var Data : __Pointer)';
5100: 4159 TStStoreDataEvent', 'Procedure ( Sender : TObject; Writer : TWriter; Data : __Pointer)';
5101: 4160 TStStringCompareEvent', 'Procedure(Sender:TObject;const String1,String2:string;var Compare:Integer);
5102: 4160 ***** File 249: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_StDate.pas
5103: 4161 TStDate', 'LongInt';
5104: 4162 TStDayType', '( Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday )';
5105: 4163 TStBondDateType', '( bdtActual, bdt30E360, bdt30360, bdt30360psa)';
5106: 4164 TStTime', 'LongInt';
5107: 4165 TStDateTimeRec', 'record D : TStDate; T : TStTime; end';
5108: 4165 ***** File 250: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_StMoney.pas
5109: 4166 TStConversionType', '( ctUnknown, ctTriangular, ctMultiply, ctDi'
5110: 4167 TStGetRateUpdateEvent', 'Procedure ( Sender : TObject; NewRate : '
5111: 4167 ***** File 251: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_StNet.pas
5112: 4168 TStNetItemType', '( nitUnknown, nitLocalUser, nitGlobalUser, nit'
5113: 4169 TStNetSidType', '( nstNone, nstUser, nstGroup, nstDomain, nstAli'
5114: 4170 TStNetUserPrivType', '( uptUnknown, uptGuest, uptUser, uptAdmin )';
5115: 4171 TStNetUserAuthPrivType', '( uaptPrint, uaptCommunications, uaptS'
5116: 4172 TStNetUserAuthPrivSet', 'set of TStNetUserAuthPrivType';
5117: 4173 TStNetShareType', '( stUnknown, stDisk, stPrint, stDevice, stIPC, stSpecial )';
5118: 4174 TStNetServerPlatformType', '( sptUnknown, sptDOS, sptOS2, sptNT, sptOSF, sptVMS )';
5119: 4175 TStNetServerType', '( nsvtWorkstation, nsvtServer, nsvtSQLServer'
5120: 4176 TStNetServerSet', 'set of TStNetServerType';
5121: 4177 TStNetServerRoleType', '( srtUnknown, srtStandAlone, srtMember, '
5122: 4178 TStSidRecord', 'record Value : __Pointer; ValueS : string; Length '
5123: 4178 ***** File 252: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_StNetCon.pas
5124: 4179 TStNetConnectOptions', '( coUseConnectDialog, coPersistentConnec'
5125: 4180 TStNetDisconnectOptions', '( doUseDisconnectDialog, doUpdateProf'
5126: 4181 TStNetConnectOptionsSet', 'set of TStNetConnectOptions';
5127: 4182 TStNetDisconnectOptionsSet', 'set of TStNetDisconnectOptions';
5128: 4183 TOnConnectFailEvent', 'Procedure ( Sender : TObject; ErrorCode : '
5129: 4184 TOnConnectCancelEvent', 'Procedure ( Sender : TObject; ErrorCode'
5130: 4185 TOnDisconnectFailEvent', 'Procedure ( Sender : TObject; ErrorCod'
5131: 4186 TOnDisconnectCancelEvent', 'Procedure ( Sender : TObject; ErrorC'
5132: 4186 ***** File 253: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_StNetPfm.pas
5133: 4187 TStCPFlags', '( cpfForNetCard, cpfNotRouted, cpfSlowLink, cpfDynamic )';
5134: 4188 TStCPFlagsSet', 'set of TStCPFlags';
5135: 4188 ***** File 254: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_StNTLog.pas
5136: 4189 TStNTEventType', '( etSuccess, etError, etWarning, etInfo, etAud'
5137: 4190 PStNTEventLogRec', '^TStNTEventLogRec // will not work';
5138: 4191 TStNTEventLogRec', 'record Length : DWORD; Reserved : DWORD; Rec'
5139: 4192 TStReadRecordEvent', 'Procedure ( Sender : TObject; const EventR'
5140: 4192 ***** File 255: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_StSort.pas
5141: 4193 TMergeInfo', 'record SortStatus : Integer; MergeFiles : Integer; '
5142: 4193 ***** File 256: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_StVInfo.pas
5143: 4194 PVerTranslation', '^TVerTranslation // will not work';

```

```

5144: 4195 TVerTranslation', 'record Language : Word; CharSet : Word; end');
5145: 4195 ***** File 257: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upSI_StWmDCpy.pas
5146: 4196 tagCOPYDATASTRUCT', 'record dwData: DWORD; cbData: DWORD; lpData: integer; end; ');
5147: 4197 TStOnDataReceivedEvent', 'Procedure (Sender : TObject; CopyData
5148: 4197 ***** File 258: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upSI_SynEdit.pas
5149: 4198 TSynBorderStyle', 'TBorderStyle');
5150: 4199 TSynReplaceAction', '( raCancel, raSkip, raReplace, raReplaceAll)';
5151: 4200 TDropFilesEvent', 'Procedure ( Sender : TObject; X, Y : integer;
5152: 4201 THookedCommandEvent', 'Procedure ( Sender : TObject; AfterProces
5153: 4202 TPaintEvent', 'Procedure ( Sender : TObject; ACanvas : TCanvas)';
5154: 4203 TProcessCommandEvent', 'Procedure ( Sender : TObject; var Comman
5155: 4204 TReplaceTextEvent', 'Procedure ( Sender : TObject; const ASearch
5156: 4205 TSpecialLineColorsEvent', 'Procedure ( Sender : TObject; Line :
5157: 4206 TTransientType', '( ttBefore, ttAfter )';
5158: 4207 TPaintTransient', 'Procedure ( Sender : TObject; Canvas : TCanva
5159: 4208 TScrollEvent', 'Procedure ( Sender : TObject; ScrollBar : TScrollBarKind)';
5160: 4209 TGutterGetTextEvent', 'Procedure ( Sender : TObject; aLine : int
5161: 4210 TGutterPaintEvent', 'Procedure ( Sender : TObject; aLine : integer; X, Y : integer)';
5162: 4211 TSynEditCaretType', '( ctVerticalLine, ctHorizontalLine, ctHalfBlock, ctBlock )';
5163: 4212 TSynStateFlag', '( sfCaretChanged, sfScrollbarChanged, sfLinesCh
5164: 4213 TSynStateFlags', 'set of TSynStateFlag';
5165: 4214 TScrollHintFormat', '( shfTopLineOnly, shfTopToBottom )';
5166: 4215 TSynEditorOption', '( eoAltSetsColumnMode, eoAutoIndent, eoAutoS
5167: 4216 TSynEditorOptions', 'set of TSynEditorOption';
5168: 4217 TSynStatusChange', '( scAll, scCaretX, scCaretY, scLeftChar, scT
5169: 4218 TSynStatusChanges', 'set of TSynStatusChange';
5170: 4219 TContextHelpEvent', 'Procedure ( Sender : TObject; word : string)';
5171: 4220 TStatusChangeEvent', 'Procedure ( Sender : TObject; Changes : TS
5172: 4221 TMouseCursorEvent', 'Procedure ( Sender : TObject; const aLineCh
5173: 4222 TPlaceMarkEvent', 'Procedure ( Sender : TObject; var Mark : TSynEditMark)';
5174: 4223 TGutterClickEvent', 'Procedure ( Sender : TObject; Button : TMou
5175: 4223 ***** File 259: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upSI_SynEditExport.pas
5176: 4224 PSynReplaceCharsArray', 'TSynReplaceCharsArray // will not work';
5177: 4224 ***** File 260: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upSI_SynEditHighlighter.pas
5178: 4225 TSynHighlighterCapability', '( hcUserSettings, hcRegistry )';
5179: 4226 TSynHighlighterCapabilities', 'set of TSynHighlighterCapability';
5180: 4227 TSynCustomHighlighterClass', 'class of TSynCustomHighlighter';
5181: 4227 ***** File 261: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upSI_SynEditKbdHandler.pas
5182: 4228 TMouseCursorEvent', 'Procedure ( Sender : TObject; const aLineCh
5183: 4228 ***** File 262: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upSI_SynEditKeyCmds.pas
5184: 4229 TSynEditorCommand', 'word';
5185: 4229 ***** File 263: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upSI_SynEditMiscClasses.pas
5186: 4230 TSynGutterBorderStyle', '( gbsNone, gbsMiddle, gbsRight )';
5187: 4231 TSynExceptionEvent', 'Procedure ( Sender : TObject; E : Exceptio
5188: 4232 TSynBorderStyle', 'TBorderStyle';
5189: 4233 THKModifier', '( hkShift, hkCtrl, hkAlt )';
5190: 4234 THKModifiers', 'set of THKModifier';
5191: 4235 THKInvalidKey', '( hcNone, hcShift, hcCtrl, hcAlt, hcShiftCtrl,
5192: 4236 THKInvalidKeys', 'set of THKInvalidKey';
5193: 4237 TBetterRegistry', 'TRegistry';
5194: 4237 ***** File 264: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upSI_SynEditMiscProcs.pas
5195: 4238 PIntArray', 'TIntArray // will not work';
5196: 4239
5197: 4240
5198: 4241 TStringType', '( stNone, stHalfNumAlpha, stHalfSymbol, stHalfKat
5199: 4242 TReplaceFlag', '( rfReplaceAll, rfIgnoreCase )';
5200: 4243 TReplaceFlags', 'set of TReplaceFlag';
5201: 4244 TReplaceFlag', '(rfReplaceAll, rfIgnoreCase)';
5202: 4245 TReplaceFlags', 'set of TReplaceFlag';
5203: 4246 THighlighterAttriProc', 'Function ( Highlighter : TSynCustomHigh
5204: 4246 ***** File 265: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upSI_SynEditTypes.pas
5205: 4247 TSynSearchOption', '( ssoMatchCase, ssoWholeWord, ssoBackwards,
5206: 4248 TSynSearchOptions', 'set of TSynSearchOption';
5207: 4249 TSynIdentChars', 'set of char';
5208: 4250 PSynSelectionMode', 'TSynSelectionMode // will not work';
5209: 4251 TSynSelectionMode', '( smNormal, smLine, smColumn )';
5210: 4252 TBufferCoord', 'record Char : integer; Line : integer; end';
5211: 4253 TDisplayCoord', 'record Column : integer; Row : integer; end';
5212: 4253 ***** File 266: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upSI_SynHighlighterAny.pas
5213: 4254 TtkTokenKind', '( tkComment, tkIdentifier, tkKey, tkNull, tkNumb
5214: 4255 TCommentStyle', '(csAnsiStyle, csPasStyle, csCStyle, csAsmStyle, csBasStyle, csVBStyle)';
5215: 4256 CommentStyles', 'set of TCommentStyle';
5216: 4257 TRangeState', '(rsAnil,rsAnsi,rsPasStyle,rsCStyle,rsUnKnown)';
5217: 4258 TStringDelim', '( sdSingleQuote, sdDoubleQuote )';
5218: 4259 TProcTableProc', 'Procedure';
5219: 4259 ***** File 267: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upSI_SynHighlighterMulti.pas
5220: 4260 TOnCheckMarker', 'Procedure ( Sender : TObject; var StartPos, Ma
5221: 4261 TgmSchemeClass', 'class of TScheme';
5222: 4262 TRangeOperation', '( roGet, roSet )';
5223: 4263 TRangeProc', 'Procedure ( Operation : TRangeOperation; var Range
5224: 4264 TCustomRangeEvent', 'Procedure ( Sender : TSynMultiSyn; Operatio
5225: 4264 ***** File 268: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upSI_SynHighlighterPas.pas
5226: 4265 TtkTokenKind', '( tkAsm, tkComment, tkIdentifier, tkKey, tkNull,
5227: 4266 TRangeState', '( rsAnil, rsAnsi, rsAnsiAsm, rsAsm, rsBor, rsBorA
5228: 4267 TProcTableProc', 'Procedure';
5229: 4268 PIdentFuncTableFunc', 'TIdentFuncTableFunc // will not work';
5230: 4269 TIdentFuncTableFunc', 'Function : TtkTokenKind';
5231: 4270 TDelphiVersion', '( dvDelphi1, dvDelphi2, dvDelphi3, dvDelphi4,
5232: 4270 ***** File 269: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upSI_SynMacroRecorder.pas

```

```

5233: 4271 TSynMacroState', '( msStopped, msRecording, msPlaying, msPaused)';
5234: 4272 TSynMacroCommand', '( mcRecord, mcPlayback )';
5235: 4273 TSynUserCommandEvent', 'Procedure ( aSender : TCustomSynMacroRec'
5236: 4273 ***** File 270: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_SynPdf.pas
5237: 4274 PSmallIntArray', '^TSmallIntArray // will not work';
5238: 4275 PPointArray', '^TPointArray // will not work';
5239: 4276 PSmallPointArray', '^TSmallPointArray // will not work';
5240: 4277 TMapHeader', 'record version : word; numberSubtables : word; end';
5241: 4278 TMapHHEA', 'record version : longint; ascent : word; descent : '
5242: 4279 TMapHEAD', 'record version : longint; fontRevision : longint; c'
5243: 4280 TMapFmt4', 'record format : word; length : word; language : wor'
5244: 4281 PDFString', 'AnsiString';
5245: 4282 TPdfDate', 'PDFString';
5246: 4283 TPdfPageMode', '( pmUseNone, pmUseOutlines, pmUseThumbs, pmFullScreen )';
5247: 4284 TLineCapStyle', '( lcButt_End, lcRound_End, lcProjectingSquareEnd )';
5248: 4285 TLineJoinStyle', '( ljMiterJoin, ljRoundJoin, ljBevelJoin )';
5249: 4286 TTextRenderingMode', '( trFill, trStroke, trFillThenStroke, trIn'
5250: 4287 TPdfAnnotationSubType', '( asTextNotes, asLink )';
5251: 4288 TPdfDestinationType', '( dtXYZ, dtFit, dtFitH, dtFitV, dtFitR, d'
5252: 4289 TPdfPageLayout', '( plSinglePage, plOneColumn, plTwoColumnLeft, '
5253: 4290 TPdfViewerPreference', '( vpHideToolbar, vpHideMenubar, vpHideWi'
5254: 4291 TPdfViewerPreferences', 'set of TPdfViewerPreference';
5255: 4292 TPdfPaperSize', '( psA4, psA5, psA3, psLetter, psLegal, psUserDefined )';
5256: 4293 TPdfCompressionMethod', '( cmNone, cmFlateDecode )';
5257: 4294 TXObjectID', 'integer';
5258: 4295 TPdfAlignment', '( paLeftJustify, paRightJustify, paCenter )';
5259: 4296 TGradientDirection', '( gdHorizontal, gdVertical )';
5260: 4297 TPdfRect', 'record Left : single; Top : single; Right : single; '
5261: 4298 PPdfRect', '^TPdfRect // will not work';
5262: 4299 TPdfBox', 'record Left : single; Top : single; Width : single; H'
5263: 4300 PPdfBox', '^TPdfBox // will not work';
5264: 4301 TPdfObjectType', '( otDirectObject, otIndirectObject, otVirtualObject )';
5265: 4302 TPdfEncryptionLevel', '( elNone, elRC4_40, elRC4_128 )';
5266: 4303 TPdfEncryptionPermission', '( epPrinting, epGeneralEditing, epCo'
5267: 4304 TPdfEncryptionPermissions', 'set of TPdfEncryptionPermission';
5268: 4305 TPdfPageClass', 'class of TPdfPage';
5269: 4306 TPdfGDIComment', '( pgcOutline, pgcBookmark, pgcLink )';
5270: 4307 RawUTF8', 'AnsiString';
5271: 4308 PPdfWinAnsiWidth', '^TPdfWinAnsiWidth // will not work';
5272: 4309 TScriptState_enum', '( rr0, rr1, rr2, rr3, rr4, fOverrideDirection, f'
5273: 4310 TScriptState_set', 'set of TScriptState_enum';
5274: 4311 PScriptState', 'TScriptState // will not work';
5275: 4312 TScriptAnalysis_enum', '( ss0, ss1, ss2, ss3, ss4, ss5, ss6, ss7, ss8, ss9'
5276: 4313 TScriptAnalysis_set', 'set of TScriptAnalysis_enum';
5277: 4314 PScriptAnalysis', 'TScriptAnalysis // will not work';
5278: 4315 PScriptItem', '^TScriptItem // will not work';
5279: 4316 TScriptItem', 'record iCharPos : Integer; a : TScriptAnalysis; end';
5280: 4317 TScriptProperties_enum', '( fNumeric, fComplex, fNeedsWordBreaki'
5281: 4318 TScriptProperties_set', 'set of TScriptProperties_enum';
5282: 4319 PScriptProperties', 'TScriptProperties // will not work';
5283: 4320 TScriptProperties', 'record langid : Word; fFlags : TScriptProperties_set; end';
5284: 4321 PScriptPropertiesArray', 'TPScriptPropertiesArray // will not work';
5285: 4322 TScriptVisAttr_enum', '( aa0, aa1, aa2, aa3, fClusterStart, fDiacrit'
5286: 4323 TScriptVisAttr_set', 'set of TScriptVisAttr_enum';
5287: 4324 PScriptVisAttr', '^TScriptVisAttr // will not work';
5288: 4324 ***** File 271: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_TlHelp32.pas
5289: 4325 tagHEAPLIST32', 'record dwSize : DWORD; th32ProcessID : DWORD; t'
5290: 4326 HEAPLIST32', 'tagHEAPLIST32';
5291: 4327 PHEAPLIST32', '^tagHEAPLIST32 // will not work';
5292: 4328 LPHEAPLIST32', 'tagHEAPLIST32 // will not work';
5293: 4329 THeapList32', 'tagHEAPLIST32';
5294: 4330 tagHEAPENTRY32', 'record dwSize : DWORD; hHandle : THandle; dwAd'
5295: 4331 HEAPENTRY32', 'tagHEAPENTRY32';
5296: 4332 PHEAPENTRY32', '^tagHEAPENTRY32 // will not work';
5297: 4333 LPHEAPENTRY32', 'tagHEAPENTRY32 // will not work';
5298: 4334 THeapEntry32', 'tagHEAPENTRY32';
5299: 4335 PROCESSENTRY32W', 'tagPROCESSENTRY32W';
5300: 4336 PPROCESSENTRY32W', '^tagPROCESSENTRY32W // will not work';
5301: 4337 LPPROCESSENTRY32W', 'tagPROCESSENTRY32W // will not work';
5302: 4338 TProcessEntry32W', 'tagPROCESSENTRY32W';
5303: 4339 PROCESSENTRY32', 'tagPROCESSENTRY32';
5304: 4340 PPROCESSENTRY32', '^tagPROCESSENTRY32 // will not work';
5305: 4341 LPPROCESSENTRY32', 'tagPROCESSENTRY32 // will not work';
5306: 4342 TProcessEntry32', 'tagPROCESSENTRY32';
5307: 4343 tagTHREADENTRY32', 'record dwSize : DWORD; cntUsage : DWORD; th3'
5308: 4344 THREADENTRY32', 'tagTHREADENTRY32';
5309: 4345 PTHREADENTRY32', '^tagTHREADENTRY32 // will not work';
5310: 4346 LPTHREADENTRY32', 'tagTHREADENTRY32 // will not work';
5311: 4347 TThreadEntry32', 'tagTHREADENTRY32';
5312: 4348 MODULEENTRY32', 'tagMODULEENTRY32';
5313: 4349 PMODULEENTRY32', '^tagMODULEENTRY32 // will not work';
5314: 4350 LPMODULEENTRY32', 'tagMODULEENTRY32 // will not work';
5315: 4351 TModuleEntry32', 'tagMODULEENTRY32';
5316: 4352 MODULEENTRY32W', 'tagMODULEENTRY32W';
5317: 4353 PMODULEENTRY32W', '^tagMODULEENTRY32W // will not work';
5318: 4354 LPMODULEENTRY32W', 'tagMODULEENTRY32W // will not work';
5319: 4355 TModuleEntry32W', 'tagMODULEENTRY32W';
5320: 4355 ***** File 272: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_utypes.pas
5321: 4356 Float', 'Single';

```



```

5322: 4357 Float', 'Extended');
5323: 4358 Float', 'Double');
5324: 4359 Complex', 'record X : Float; Y : Float; end');
5325: 4360 TVector', 'array of Float');
5326: 4361 TIntVector', 'array of Integer');
5327: 4362 TCompVector', 'array of Complex');
5328: 4363 TBoolVector', 'array of Boolean');
5329: 4364 TStrVector', 'array of String');
5330: 4365 TMatrix', 'array of TVector');
5331: 4366 TIntMatrix', 'array of TIntVector');
5332: 4367 TCompMatrix', 'array of TCompVector');
5333: 4368 TBoolMatrix', 'array of TBoolVector');
5334: 4369 TStrMatrix', 'array of TStrVector');
5335: 4370 RNG_Type', '( RNG_MWC, RNG_MT, RNG_UVAG )';
5336: 4371 RNG_IntType', 'Integer');
5337: 4372 RNG_LongType', 'Cardinal');
5338: 4373 StatClass', 'record Inf : Float; Sup : Float; N : Integer; F : Float; D : Float; end');
5339: 4374 TStatClassVector', 'array of StatClass');
5340: 4375 TRegMode', '( OLS, WLS )';
5341: 4376 TRegTest', 'record Vr : Float; R2 : Float; R2a : Float; F : Float; Nul : integer; Nu2 : Integer;
end');
5342: 4377 TOptAlgo', '( NL_MARQ, NL_SIMP, NL_BFGS, NL_SA, NL_GA )';
5343: 4378 TMintVar', '( Var_T, Var_S, Var_E )';
5344: 4379 TScale', '( LinScale, LogScale )';
5345: 4380 TGrid', '( NoGrid, HorizGrid, VertiGrid, BothGrid )';
5346: 4381 TArgC', 'Integer');
5347: 4382 TFunc', 'function(X : double): double;');
5348: 4383 TEquations', 'procedure(X, F : TVector);';
5349: 4384 TJacobian', 'procedure(X : TVector; D : TMatrix);';
5350: 4385 TGradient', 'procedure(X, G : TVector);';
5351: 4386 THessGrad', 'procedure(X, G : TVector; H : TMatrix);';
5352: 4387 TFuncNVar', 'function(X : TVector) : Float;';
5353: 4388 TRegFunc', 'function(X : Float; B : TVector) : Float;';
5354: 4389 TDerivProc', 'procedure(X, Y : Float; B, D : TVector);';
5355: 4390 TMintVar', '(Var_T, Var_S, Var_E)';
5356: 4391 TDiffEqs', 'procedure(X : Float; Y, Yp : TVector);';
5357: 4392 TArgC', '1..26');
5358: 4393 TWrapper', 'function(ArgC : Byte; ArgV : TVector) : Float;';
5359: 4393 ***** File 273: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_uwinplot.pas
5360: 4394 TFunc', 'function(X : double): double;';
5361: 4394 ***** File 274: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_ValEdit.pas
5362: 4395 TDisplayOption', '( doColumnTitles, doAutoColResize, doKeyColFix'
5363: 4396 TDisplayOptions', 'set of TDisplayOption');
5364: 4397 TKeyOption', '( keyEdit, keyAdd, keyDelete, keyUnique )';
5365: 4398 TKeyOptions', 'set of TKeyOption');
5366: 4399 TGetPickListEvent', 'Procedure ( Sender : TObject; const KeyName'
5367: 4400 TOnValidateEvent', 'Procedure ( Sender : TObject; ACol, ARow : L'
5368: 4401 TItemProps', 'array of TItemProp');
5369: 4401 ***** File 275: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_xrtl_math_Integer.pas
5370: 4402 TXRTLInteger', 'array of Integer');
5371: 4402 ***** File 276: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_xrtl_net_NamedPipe.pas
5372: 4403 TXRTLNamedPipeInstanceCount', 'Integer');
5373: 4403 ***** File 277: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_xrtl_util_COMCat.pas
5374: 4404 TLCID', 'DWORD');
5375: 4404 ***** File 278: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_xrtl_util_CPUUtils.pas
5376: 4405 TXRTLBitIndex', 'Integer');
5377: 4406 PDWORDArray', '^TDWORDArray // will not work';
5378: 4407 PCardinalArray', '^TCardinalArray // will not work';
5379: 4408 PInt64Array', '^TInt64Array // will not work';
5380: 4409 PCardinalBytes', '^TCardinalBytes // will not work';
5381: 4410 PCardinalWords', '^TCardinalWords // will not work';
5382: 4411 PInt64Bytes', '^TInt64Bytes // will not work';
5383: 4412 PInt64Words', '^TInt64Words // will not work';
5384: 4413 PInt64DWords', '^TInt64DWords // will not work'); *)
5385: 4413 ***** File 279: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_xrtl_util_TimeZone.pas
5386: 4414 REGTIMEZONEINFORMATION', 'record Bias : LongInt; StandardBias : '
5387: 4414 ***** File 280: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\upsi_xrtl_util_Value.pas
5388: 4415 PXRTLValueArray', '^TXRTLValueArray // will not work';
5389: 4416 TXRTLValueArray', 'array of IXRTLValue');
5390: 4416 ***** File 281: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\synopse\upsi_JvAirBrush.pas
5391: 4417 TJvAirBrushShape', '( absRound, absSquare, absLeftSlash, absRight'
5392: 4417 ***** File 282: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\synopse\upsi_mORMotReport.pas
5393: 4418 TTextAlign', '( taLeft, taRight, taCenter, taJustified)';
5394: 4419 TColAlign', '( caLeft, caRight, caCenter, caCurrency)';
5395: 4420 TLineSpacing', '( lsSingle, lsOneAndHalf, lsDouble)';
5396: 4421 TZoomStatus', '( zsPercent, zsPageFit, zsPageWidth)';
5397: 4422 TNewPageEvent', 'Procedure ( Sender : TObject; PageNumber : integer)';
5398: 4423 TZoomChangedEvent', 'Procedure ( Sender : TObject; Zoom : integer'
5399: 4424 TOnStringToUnicodeEvent', 'Function(const Text : SynUnicode):SynUnicode');
5400: 4425 TGdiPagePaperSize', '( psA4, psA5, psA3, psLetter, psLegal)';
5401: 4426 TSavedState', 'record FontName : string; FontColor : integer; Fl'
5402: 4427 TColRec', 'record ColAlign : TColAlign; ColBold : boolean; end';
5403: 4428 TPopupMenuClass', 'class of TPopupMenu');
5404: 4429 TGDIPageContentDynArray', 'array of TGDIPageContent');
5405: 4430 TSavedStateRender', 'record FirstLineIndent : Integer; Before : '
5406: 4431 PRenderBoxWord', '^TRenderBoxWord // will not work';
5407: 4432 TRenderBoxWord', 'record TextOffset : integer; TextLength : integer'
5408: 4433 PRenderBoxLayout', '^TRenderBoxLayout // will not work';
5409: 4434 TRenderBoxLayout', 'record Text : PWideChar; Length : integer; L'

```

```

5410: 4435 TReportPopupMenu', '( rNone, rNextPage, rPreviousPage, rGotoPage'
5411: 4435 ***** File 283: F:\maxbox3\mX397\maxbox3\source\fundamentals_lib\synopse\uPSI_SynPdf.pas
5412: 4436 PSmallIntArray', '^TSmallIntArray // will not work');
5413: 4437 PPointArray', '^TPointArray // will not work');
5414: 4438 PSmallPointArray', '^TSmallPointArray // will not work');
5415: 4439 TMapHeader', 'record version : word; numberSubtables : word; end';
5416: 4440 TMapHHEA', 'record version : longint; ascent : word; descent : '
5417: 4441 TMapHEAD', 'record version : longint; fontRevision : longint; c'
5418: 4442 TMapFmt4', 'record format : word; length : word; language : wor'
5419: 4443 PDFString', 'AnsiString';
5420: 4444 TPdfDate', 'PDFString';
5421: 4445 TPdfPageMode', '( pmUseNone, pmUseOutlines, pmUseThumbs, pmFullScreen )';
5422: 4446 TLineCapStyle', '( lcButt_End, lcRound_End, lcProjectingSquareEnd )';
5423: 4447 TLineJoinStyle', '( ljMiterJoin, ljRoundJoin, ljBevelJoin )';
5424: 4448 TTextRenderingMode', '( trFill, trStroke, trFillThenStroke, trIn'
5425: 4449 TPdfAnnotationSubType', '( asTextNotes, asLink )';
5426: 4450 TPdfDestinationType', '( dtXYZ, dtFit, dtFitH, dtFitV, dtFitR, d'
5427: 4451 TPdfPageLayout', '( plSinglePage, plOneColumn, plTwoColumnLeft, '
5428: 4452 TPdfViewerPreference', '( vpHideToolbar, vpHideMenubar, vpHideWi'
5429: 4453 TPdfViewerPreferences', 'set of TPdfViewerPreference';
5430: 4454 TPdfPaperSize', '( psA4, psA5, psA3, psLetter, psLegal, psUserDefined )';
5431: 4455 TPdfCompressionMethod', '( cmNone, cmFlateDecode )';
5432: 4456 TXObjectID', 'integer';
5433: 4457 TPdfAlignment', '( paLeftJustify, paRightJustify, paCenter )';
5434: 4458 TGradientDirection', '( gdHorizontal, gdVertical )';
5435: 4459 TPdfRect', 'record Left : single; Top : single; Right : single; '
5436: 4460 PPdfRect', '^TPdfRect // will not work';
5437: 4461 TPdfBox', 'record Left : single; Top : single; Width : single; H'
5438: 4462 PPdfBox', '^TPdfBox // will not work';
5439: 4463 TPdfObjectType', '( otDirectObject, otIndirectObject, otVirtualObject )';
5440: 4464 TPdfEncryptionLevel', '( elNone, elRC4_40, elRC4_128 )';
5441: 4465 TPdfEncryptionPermission', '( epPrinting, epGeneralEditing, epCo'
5442: 4466 TPdfEncryptionPermissions', 'set of TPdfEncryptionPermission';
5443: 4467 TPdfPageClass', 'class of TPdfPage';
5444: 4468 TPdfGDIComment', '( pgcOutline, pgcBookmark, pgcLink )';
5445: 4469 RawUTF8', 'AnsiString';
5446: 4470 PPdfWinAnsiWidth', '^TPdfWinAnsiWidth // will not work';
5447: 4471 TScriptState_enum', '( rr0, rr1, rr2, rr3, rr4, fOverrideDirection, f'
5448: 4472 TScriptState_set', 'set of TScriptState_enum';
5449: 4473 PScriptState', 'TScriptState // will not work';
5450: 4474 TScriptAnalysis_enum', '( ss0, ss1, ss2, ss3, ss4, ss5, ss6, ss7, ss8, ss9'
5451: 4475 TScriptAnalysis_set', 'set of TScriptAnalysis_enum';
5452: 4476 PScriptAnalysis', 'TScriptAnalysis // will not work';
5453: 4477 PScriptItem', 'TScriptItem // will not work';
5454: 4478 TScriptItem', 'record iCharPos : Integer; a : TScriptAnalysis; end';
5455: 4479 TScriptProperties_enum', '( fNumeric, fComplex, fNeedsWordBreaki'
5456: 4480 TScriptProperties_set', 'set of TScriptProperties_enum';
5457: 4481 PScriptProperties', 'TScriptProperties // will not work';
5458: 4482 TScriptProperties', 'record langid : Word; fFlags : TScriptProperties_set; end';
5459: 4483 PScriptPropertiesArray', 'TPScriptPropertiesArray // will not work';
5460: 4484 TScriptVisAttr_enum', '( aa0, aa1, aa2, aa3, fClusterStart, fDiacrit'
5461: 4485 TScriptVisAttr_set', 'set of TScriptVisAttr_enum';
5462: 4486 PScriptVisAttr', 'TScriptVisAttr // will not work';
5463: 4486 ***** File 284: F:\maxbox3\mX397\maxbox3\source\REST\uPSI_cHTTTPUtils.pas
5464: 4487 AnsiCharSet', 'set of AnsiChar';
5465: 4488 AnsiStringArray', 'array of AnsiString';
5466: 4489 cHTTTPParser', 'HTTTPParser';
5467: 4490 THTTTPProtocolEnum', '( hpNone, hpCustom, hpHTTP, hpHTTPS )';
5468: 4491 THTTTPVersionEnum', '( hvNone, hvCustom, hvHTTTP10, hvHTTTP11 )';
5469: 4492 THTTTPVersion', 'record Version : THTTTPVersionEnum; Protocol : TH'
5470: 4493 THTTTPHeaderNameEnum', '( hntCustom, hntHost, hntContentType, hnt'
5471: 4494 THTTTPHeaderName', 'record Value : THTTTPHeaderNameEnum; Custom: AnsiString; end';
5472: 4495 THTTTPCustomHeader', 'record FieldName : AnsiString; FieldValue : '
5473: 4496 PHTTTPCustomHeader', '^HTTTPCustomHeader // will not work';
5474: 4497 THTTTPContentLengthEnum', '( hcltNone, hcltByteCount )';
5475: 4498 THTTTPContentLength', 'record Value : THTTTPContentLengthEnum; ByteCount : Int64; end';
5476: 4499 PHTTTPContentLength', '^HTTTPContentLength // will not work';
5477: 4500 THTTTPContentTypeMajor', '( hctmCustom, hctmText, hctmImage )';
5478: 4501 THTTTPContentTypeEnum', '( hctNone, hctCustomParts, hctCustomStri'
5479: 4502 THTTTPContentType', 'record Value : THTTTPContentTypeEnum; CustomM'
5480: 4503 THTTTPDateFieldEnum', '( hdNone, hdCustom, hdParts, hdDateTime )';
5481: 4504 THTTTPDateField', 'record Value : THTTTPDateFieldEnum; DayOfWeek : '
5482: 4505 THTTTPTransferEncodingEnum', '( hteNone, hteCustom, hteChunked )';
5483: 4506 THTTTPTransferEncoding', 'record Value : THTTTPTransferEncodingEnu'
5484: 4507 THTTTPConnectionFieldEnum', '( hcfNone, hcfCustom, hcfClose, hcfKeepAlive )';
5485: 4508 THTTTPConnectionField', 'record Value : THTTTPConnectionFieldEnum; '
5486: 4509 THTTTPPageFieldEnum', '( hafNone, hafCustom, hafAge )';
5487: 4510 THTTTPPageField', 'record Value : THTTTPPageFieldEnum; Age : Int64; '
5488: 4511 THTTTPCacheControlFieldEnum', '( hccfNone, hccfDecoded, hccfCustom )';
5489: 4512 THTTTPCacheControlRequestSubField', '( hccsfNoCache, hccsfNoStore'
5490: 4513 THTTTPCacheControlResponseSubField', '( hccrfPublic, hccrfPrivate'
5491: 4514 THTTTPCacheControlField', 'record Value : THTTTPCacheControlFieldEnum; end';
5492: 4515 THTTTPContentEncodingEnum', '( hceNone, hceCustom, hceIdentity, h'
5493: 4516 THTTTPContentEncoding', 'record Value : THTTTPContentEncodingEnum; '
5494: 4517 THTTTPContentEncodingFieldEnum', '( hcefNone, hcefList )';
5495: 4518 THTTTPContentEncodingField', 'record Value : THTTTPContentEncoding'
5496: 4519 THTTTPRetryAfterFieldEnum', '( hrafNone, hrafCustom, hrafDate, hrafSeconds )';
5497: 4520 THTTTPRetryAfterField', 'record Value : THTTTPRetryAfterFieldEnum; '
5498: 4521 THTTTPContentRangeFieldEnum', '( hcrfNone, hcrfCustom, hcrfByteRange )';

```

```

5499: 4522 THTTTPContentRangeField', 'record Value : THTTTPContentRangeFieldE'
5500: 4523 THTTTPSetCookieFieldEnum', '( hscNone, hscDecoded, hscCustom )';
5501: 4524 THTTTPSetCookieCustomField', 'record Name : AnsiString; Value : AnsiString; end';
5502: 4525 THTTTPSetCookieCustomFieldArray', 'array of THTTTPSetCookieCustomField';
5503: 4526 THTTTPSetCookieField', 'record Value : THTTTPSetCookieFieldEnum; D'
5504: 4527 PHTTTPSetCookieField', '^THTTTPSetCookieField // will not work';
5505: 4528 THTTTPSetCookieFieldArray', 'array of THTTTPSetCookieField';
5506: 4529 THTTTPCookieFieldEnum', '( hcoNone, hcoDecoded, hcoCustom )';
5507: 4530 THTTTPCookieFieldEntry', 'record Name : AnsiString; Value : AnsiString; end';
5508: 4531 PHTTTPCookieFieldEntry', '^THTTTPCookieFieldEntry // will not work';
5509: 4532 THTTTPCookieFieldEntryArray', 'array of THTTTPCookieFieldEntry';
5510: 4533 THTTTPCookieField', 'record Value : THTTTPCookieFieldEnum; Entries'
5511: 4534 THTTTPCommonHeaders', 'record TransferEncoding : THTTTPTransferEnc'
5512: 4535 THTTTPCustomHeaders', 'array of THTTTPCustomHeader';
5513: 4536 THTTTPHeaderNameEnums', 'set of THTTTPHeaderNameEnum';
5514: 4537 THTTTPFixedHeaders', 'array[THTTTPHeaderNameEnum] of AnsiString';
5515: 4538 THTTTPFixedHeaders', 'array[0..43] of AnsiString';
5516: 4539 THTTTPFixedHeaders', 'array[0..44] of AnsiString';
5517: 4540 THTTTPMethodEnum', '( hmNone, hmCustom, hmGET, hmPUT, hmPOST, hmC'
5518: 4541 THTTTPMethod', 'record Value : THTTTPMethodEnum; Custom : AnsiString; end';
5519: 4542 THTTTPRequestStartLine', 'record Method : THTTTPMethod; URI : Ansi'
5520: 4543 THTTTPRequestHeader', 'record CommonHeaders : THTTTPCommonHeaders;'
5521: 4544 PHTTTPRequestHeader', '^THTTTPRequestHeader // will not work';
5522: 4545 THTTTPRequest', 'record StartLine : THTTTPRequestStartLine; Header'
5523: 4546 THTTTPResponseStartLineMessage', '( hslmNone, hslmCustom, hslmOK)';
5524: 4547 THTTTPResponseStartLine', 'record Version : THTTTPVersion; Code : '
5525: 4548 THTTTPResponseHeader', 'record CommonHeaders : THTTTPCommonHeaders'
5526: 4549 PHTTTPResponseHeader', '^THTTTPResponseHeader // will not work';
5527: 4550 cTHTTTPResponse', 'record StartLine : THTTTPResponseStartLine; Head'
5528: 4551 THTTTPStringOption', '( hsoNone )';
5529: 4552 THTTTPStringOptions', 'set of THTTTPStringOption';
5530: 4553 THTTTPParserHeaderParseFunc', 'Function ( const HeaderName : THTT'
5531: 4554 THTTTPContentDecoderProc', 'Procedure ( const Sender : THTTTPContentDecoder)';
5532: 4555 THTTTPContentDecoderContentType', '( crctFixedSize, crctChunked, crctUnsize )';
5533: 4556 THTTTPContentDecoderChunkState', '( crcsChunkHeader, crcsContent, '
5534: 4557 THTTTPContentDecoderLogEvent', 'Procedure ( const Sender : THTTTPC'
5535: 4558 THTTTPContentReaderMechanism', '( hcrmEvent, hcrmString, hcrmStream, hcrmFile )';
5536: 4559 THTTTPContentReaderProc', 'Procedure ( const Sender : THTTTPContentReader)';
5537: 4560 THTTTPContentReaderLogEvent', 'Procedure(const Sender:THTTTPContentReader;const LogMsg:String;const
LogLevel:Integer;
5538: 4561 THTTTPContentWriterMechanism', '( hctmEvent, hctmString, hctmStream, hctmFile )';
5539: 4562 THTTTPContentWriterLogEvent', 'Procedure ( const Sender : THTTTPCo'
5540: 4562 ***** File 285: F:\maxbox3\mX397\maxbox3\source\REST\uPSI_cJSON.pas
5541: 4563 TJSONValueType', '( jvtString, jvtInteger, jvtFloat, jvtBoolean, '
5542: 4564 TJSONStringOptions', 'set of ( jboIndent )';
5543: 4565 TJSONObjectItem', 'record Name : UnicodeString; Value : TJSONVal'
5544: 4566 PJSONObjectItem', '^TJSONObjectItem // will not work';
5545: 4567 TJSONObjectIterator', 'record InternalIndex : Integer; Item : TJ'
5546: 4568 TJSONParserToken', '( jptNone, jptEOF, jptWhiteSpace, jptLeftSqu'
5547: 4568 ***** File 286: F:\maxbox3\mX397\maxbox3\source\REST\uPSI_CppTokenizer.pas
5548: 4569 TSetOfChars', 'set of Char';
5549: 4570 TLogTokenEvent', 'Procedure ( Sender : TObjet; Msg : string)';
5550: 4571 TProgressEvent', 'Procedure (Sender: TObjet; FileName: string; Total, Current : integer)';
5551: 4572 PToken', '^CTToken // will not work';
5552: 4573 CTToken', 'record Text : string; Line : integer; end';
5553: 4573 ***** File 287: F:\maxbox3\mX397\maxbox3\source\REST\uPSI_cTLSTools.pas
5554: 4574 TTLSProtocolVersion', 'record major : Byte; minor : Byte; end';
5555: 4575 TTLSRandom', 'record gmt_unix_time : LongWord;random_bytes : array[0..27] of Byte; end';
5556: 4576 TTLSSessionID', 'shortstring';
5557: 4577 PTLSPProtocolVersion', '^TTLSProtocolVersion // will not work';
5558: 4578 PTLSPRandom', '^TTLSRandom // will not work';
5559: 4579 TTLSSignatureAndHashAlgorithm', 'record Hash : TTLSHashAlgorithm'
5560: 4580 PTLSPSignatureAndHashAlgorithm', '^TTLSSignatureAndHashAlgorithm '
5561: 4581 TTLSSignatureAndHashAlgorithmArray', 'array of TTLSSignatureAndHashAlgorithm';
5562: 4582 TTLSKeyExchangeAlgorithm', '( tksaNone, tksaNULL, tksaDHE_ '
5563: 4583 TTLSMACAlgorithm', '( tksaNone, tksaNULL, tksaHMAC_MD5, tksa_ '
5564: 4584 TTLSMACAlgorithmInfo', 'record Name : AnsiString; DigestSize : I'
5565: 4585 PTLSPMACAlgorithmInfo', '^TTLSMACAlgorithmInfo // will not work';
5566: 4586 TTLSPRFAlgorithm', '( tksaSHA256 )';
5567: 4587 TTLSKeys', 'record KeyBlock : AnsiString; ClientMACKey : AnsiStr'
5568: 4587 ***** File 288: F:\maxbox3\mX397\maxbox3\source\REST\uPSI_devFileMonitorX.pas
5569: 4588 TdevMonitorChangeType', '( mctChanged, mctDeleted )';
5570: 4589 TdevMonitorChange', 'Procedure ( Sender : TObjet; ChangeType : '
5571: 4589 ***** File 289: F:\maxbox3\mX397\maxbox3\source\REST\uPSI_devrun.pas
5572: 4590 TLineOutputEvent', 'Procedure ( Sender : TObjet; const Line : String)';
5573: 4590 ***** File 290: F:\maxbox3\mX397\maxbox3\source\REST\uPSI_Diff.pas
5574: 4591 P8Bits', 'PByte';
5575: 4592 P8Bits', 'PAnsiChar';
5576: 4593 PDiags', 'TDiags // will not work';
5577: 4594 PIntArray', 'TIntArray // will not work';
5578: 4595 TChangeKind', '( ckNone, ckAdd, ckDelete, ckModify )';
5579: 4596 TCompareRec', 'TCompareRec // will not work';
5580: 4597 TCompareRec', 'record Kind : TChangeKind; oldIndex1 : integer; o'
5581: 4598 TDiffStats', 'record matches : integer; adds : integer; deletes '
5582: 4598 ***** File 291: F:\maxbox3\mX397\maxbox3\source\REST\uPSI_DosCommand.pas
5583: 4599 TOutputType', '( otEntireLine, otBeginningOfLine )';
5584: 4600 TReturnCode', '( rcCRLF, rcLF )';
5585: 4601 TNewLineEvent', 'Procedure ( Sender : TObjet; NewLine : string;'
5586: 4602 TTerminateEvent', 'Procedure ( Sender : TObjet; ExitCode : LongWord)';

```

```

5587: 4603 TShowWindow', '( swHIDE, swMAXIMIZE, swMINIMIZE, swRESTORE, swSH'
5588: 4604 TCreationFlag', '( fCREATE_DEFAULT_ERROR_MODE, fCREATE_NEW_CONSO'
5589: 4605 TDosThreadStatus', '( dtsAllocatingMemory, dtsAllocateMemoryFail'
5590: 4605 ***** File 292: F:\maxbox3\mX397\maxbox3\source\REST\upsi_ExtPascalUtils.pas
5591: 4606 TBrowser', '( brUnknown, brIE, brFirefox, brChrome, brSafari, br'
5592: 4607 TCSSUnit', '(cssPX, cssPerc, cssEM, cssEX, cssIN, cssCM, cssMM, cssPT, cssPC, cssnone)');
5593: 4608 TExtProcedure', 'Procedure';
5594: 4608 ***** File 293: F:\maxbox3\mX397\maxbox3\source\REST\upsi_GR32_Filters.pas
5595: 4609 TLogicalOperator', '( loXOR, loAND, loOR)');
5596: 4610 TLUT8', 'array[0..255] of Byte';
5597: 4610 ***** File 294: F:\maxbox3\mX397\maxbox3\source\REST\upsi_GR32_LowLevel.pas
5598: 4611 TWrapProc', 'function(Value, Max: Integer): Integer;';
5599: 4612 TWrapProcEx', 'function(Value, Min, Max: Integer): Integer;';
5600: 4613 TWrapMode', '(wmClamp, wmRepeat, wmMirror)';
5601: 4613 ***** File 295: F:\maxbox3\mX397\maxbox3\source\REST\upsi_GR32_Math.pas
5602: 4614 TFixed', 'Integer';
5603: 4615 TFloat', 'Single';
5604: 4615 ***** File 296: F:\maxbox3\mX397\maxbox3\source\REST\upsi_GR32_VectorMaps.pas
5605: 4616 TFixedVector', 'TFixedPoint';
5606: 4617 PFixedVector', 'TFixedVector // will not work';
5607: 4618 TFloatVector', 'TFloatPoint';
5608: 4619 PFloatVector', 'TFloatVector // will not work';
5609: 4620 TArrayOfFixedVector', 'array of TFixedVector';
5610: 4621 PArrayOfFixedVector', 'TArrayOfFixedVector // will not work';
5611: 4622 TArrayOfFloatVector', 'array of TFloatVector';
5612: 4623 PArrayOfFloatVector', 'TArrayOfFixedVector // will not work';
5613: 4624 TVectorCombineMode', '( vcmAdd, vcmReplace, vcmCustom)';
5614: 4625 TVectorCombineEvent', 'Procedure ( F, P : TFixedVector; var B : '
5615: 4625 ***** File 297: F:\maxbox3\mX397\maxbox3\source\REST\upsi_JclCOM.pas
5616: 4626 TIID', 'TGUID';
5617: 4626 ***** File 298: F:\maxbox3\mX397\maxbox3\source\REST\upsi_JclGraphics.pas
5618: 4627 TDynDynIntegerArrayArray', 'array of TDynIntegerArray';
5619: 4628 TDynPointArray', 'array of TPoint';
5620: 4629 TDynDynPointArrayArray', 'array of TDynPointArray';
5621: 4630 TPointF', 'record X : Single; Y : Single; end';
5622: 4631 TDynDynPointArrayF', 'array of TPointF';
5623: 4632 TDrawMode2', '( dmOpaque, dmBlend)';
5624: 4633 TStretchFilter2', '( sfNearest, sfLinear, sfSpline)';
5625: 4634 TConversionKind', '( ckRed, ckGreen, ckBlue, ckAlpha, ckUniformR'
5626: 4635 TResamplingFilter', '( rfBox, rfTriangle, rfHermite, rfBell, rfS'
5627: 4636 TMatrix3d2', 'array[0..2] of extended';
5628: 4637 TMatrix3d', 'record A: array[0..2] of TMatrix3d2; end';
5629: 4638 TDynDynPointArrayArrayF', 'array of TDynPointArrayF';
5630: 4639 TScanLine', 'array of Integer';
5631: 4640 TScanLines', 'array of TScanLine';
5632: 4641 TColorChannel', '( ccRed, ccGreen, ccBlue, ccAlpha)';
5633: 4642 TGradientDirection', '( gdVertical, gdHorizontal)';
5634: 4643 TPolyFillMode', '( fmAlternate, fmWinding)';
5635: 4644 TJclRegionCombineOperator', '( coAnd, coDiff, coOr, coXor)';
5636: 4645 TJclRegionBitmapMode', '( rmInclude, rmExclude)';
5637: 4646 TJclRegionKind', '( rkNull, rkSimple, rkComplex, rkError)';
5638: 4647 TLUT8', 'array[0..255] of Byte';
5639: 4648 TGamma', 'array[0..255] of Byte';
5640: 4648 ***** File 299: F:\maxbox3\mX397\maxbox3\source\REST\upsi_JclMapi.pas
5641: 4649 TJclMapiClient', 'record ClientName : string; ClientPath : strin'
5642: 4650 TJclMapiClientConnect', '( ctAutomatic, ctMapi, ctDirect)';
5643: 4651 TJclEmailRecipKind', '( rkOriginator, rkTO, rkCC, rkBCC)';
5644: 4652 FLAGS', 'Cardinal';
5645: 4653 TJclEmailFindOption', '( foFifo, foUnreadOnly)';
5646: 4654 TJclEmailLogonOption', '( loLogonUI, loNewSession, loForceDownload)';
5647: 4655 TJclEmailReadOption', '( roAttachments, roHeaderOnly, roMarkAsRead)';
5648: 4656 TJclEmailFindOptions', 'set of TJclEmailFindOption';
5649: 4657 TJclEmailLogonOptions', 'set of TJclEmailLogonOption';
5650: 4658 TJclEmailReadOptions', 'set of TJclEmailReadOption';
5651: 4659 TJclEmailReadMsg', 'record ConversationID : string; DateReceived'
5652: 4660 TJclTaskWindowsList', 'array of HWND';
5653: 4660 ***** File 300: F:\maxbox3\mX397\maxbox3\source\REST\upsi_JclShell.pas
5654: 4661 TSHDeleteOption', '( doSilent, doAllowUndo, doFilesOnly)';
5655: 4662 TSHDeleteOptions', 'set of TSHDeleteOption';
5656: 4663 TSHRenameOption', '( roSilent, roRenameOnCollision)';
5657: 4664 TSHRenameOptions', 'set of TSHRenameOption';
5658: 4665 TEnumFolderFlag', '( efFolders, efNonFolders, efIncludeHidden)';
5659: 4666 TEnumFolderFlags', 'set of TEnumFolderFlag';
5660: 4667 TEnumFolderRec', 'record DisplayName : string; Attributes : DWOR'
5661: 4668 PShellLink', 'TShellLink // will not work';
5662: 4669 TShellLink', 'record Arguments : string; ShowCmd : Integer; Work'
5663: 4670 TJclFileExeType', '( etError, etMsDos, etWin16, etWin32Gui, etWin32Con)';
5664: 4670 ***** File 301: F:\maxbox3\mX397\maxbox3\source\REST\upsi_JclSynch.pas
5665: 4671 TJclWaitResult', '( wrAbandoned, wrError, wrIoCompletion, wrSignaled, wrTimeout)';
5666: 4672 POptexSharedInfo', 'TOptexSharedInfo // will not work';
5667: 4673 TOptexSharedInfo', 'record SpinCount : Integer; LockCount : Inte'
5668: 4674 TMrewPreferred', '( mpReaders, mpWriters, mpEqual)';
5669: 4675 TMrewThreadInfo', 'record ThreadId : Longword; RecursionCount : '
5670: 4676 TMrewThreadInfoArray', 'array of TMrewThreadInfo';
5671: 4677 PMetSectSharedInfo', 'TMetSectSharedInfo // will not work';
5672: 4678 TMetSectSharedInfo', 'record Initialized : Bool; SpinLock : '
5673: 4679 PMeteredSection', 'TMeteredSection // will not work';
5674: 4680 TMeteredSection', 'record Event : THandle; FileMap : THandle; SharedInfo : PMetSectSharedInfo; end';
5675: 4681 TEventInfo', 'record EventType : Longint; Signaled : LongBool; end';

```



```

5676: 4682 TMutexInfo', 'record SignalState : Longint; Owned : Boolean; Abandoned : Boolean; end');
5677: 4683 TSemaphoreCounts', 'record CurrentCount : Longint; MaximumCount : Longint; end';
5678: 4684 TTimerInfo', 'record Remaining : TLargeInteger; Signaled : LongBool; end';
5679: 4684 ***** File 302: F:\maxbox3\mX397\maxbox3\source\REST\upsi_JvHLPParser.pas
5680: 4685 TParserStyle', '(psNone, psPascal, psCpp, psPython, psVB, psHtml, psPerl, psCocoR, psPhp, psSql)';
5681: 4685 ***** File 303: F:\maxbox3\mX397\maxbox3\source\REST\upsi_JvTimer.pas
5682: 4686 TJvTimerEventTime', '(tetPre, tetPost)';
5683: 4686 ***** File 304: F:\maxbox3\mX397\maxbox3\source\REST\upsi_JvXmlDatabase.pas
5684: 4687 TJvXMLOrderConversion', '(ocNone, ocDate, ocInteger, ocFloat)';
5685: 4688 TJvXMLSQLOperator', '(opEquals, opGreater, opSmaller, opGreater';
5686: 4689 TJvXMLSetKind', '(skConstant, skColumn)';
5687: 4690 TJvXMLSetOperator', '(soNone, soAdd, soMultiply, soDivide, soSu';
5688: 4691 TJvXMLInstruction', '(xiSelect, xiUpdate, xiInsert, xiDelete)';
5689: 4691 ***** File 305: F:\maxbox3\mX397\maxbox3\source\REST\upsi_MultilangTranslator.pas
5690: 4692 tLangChanging', 'Procedure (Sender: TObject; theComponent : TComponent)';
5691: 4693 tLangChanged', 'Procedure (Sender : TObject)';
5692: 4693 ***** File 306: F:\maxbox3\mX397\maxbox3\source\REST\upsi_RestRequest.pas
5693: 4694 TBeforeRequest', 'Procedure (Sender : TObject; var AHttpClient: TIdHttp)';
5694: 4695 THttpResponse', 'record ResponseCode : integer; ResponseStr : string; end';
5695: 4695 ***** File 307: F:\maxbox3\mX397\maxbox3\source\REST\upsi_Services.pas
5696: 4696 TEventType', '(EventError, EventWarning, EventInformation)';
5697: 4696 ***** File 308: F:\maxbox3\mX397\maxbox3\source\REST\upsi_SocketsDelphi.pas
5698: 4697 cint8', 'shortint';
5699: 4698 cuint8', 'byte';
5700: 4699 cchar', 'cint8';
5701: 4700 cschar', 'cint8';
5702: 4701 cuchar', 'cuint8';
5703: 4702 cint16', 'smallint';
5704: 4703 cuint16', 'word';
5705: 4704 cshort', 'cint16';
5706: 4705 csshort', 'cint16';
5707: 4706 cushort', 'cuint16';
5708: 4707 cint32', 'longint';
5709: 4708 cuint32', 'longword';
5710: 4709 cint', 'cint32';
5711: 4710 csint', 'cint32';
5712: 4711 cuint', 'cuint32';
5713: 4712 csigned', 'cint';
5714: 4713 cunsigned', 'cuint';
5715: 4714 cint64', 'int64';
5716: 4715 clonglong', 'cint64';
5717: 4716 cslonglong', 'cint64';
5718: 4717 cbool', 'longbool';
5719: 4718 cfloat', 'single';
5720: 4719 cdouble', 'double';
5721: 4720 clongdouble', 'extended';
5722: 4721 size_t', 'cuint32';
5723: 4722 ssize_t', 'cint32';
5724: 4723 tsocklen', 'cint';
5725: 4724 sa_family_t', 'cuchar';
5726: 4725 sa_family_t', 'cushort';
5727: 4726 in_addr', 'record s_bytes: array[1..4] of byte; end';
5728: 4727 TIn_addr', 'in_addr';
5729: 4728 TInAddr', 'in_addr';
5730: 4729 PInAddr', 'pin_addr';
5731: 4730 TInetSockAddr', 'sockaddr_in';
5732: 4731 PInetSockAddr', 'psockaddr_in';
5733: 4732 TLinger', 'linger';
5734: 4733 Tsocket', 'longint';
5735: 4733 ***** File 309: F:\maxbox3\mX397\maxbox3\source\REST\upsi_WinAPI.pas
5736: 4734 time_t', 'Longint';
5737: 4735 PIP_MASK_STRING', '^IP_MASK_STRING // will not work';
5738: 4736 PIP_ADDRESS_STRING', '^IP_ADDRESS_STRING // will not work';
5739: 4737 IP_MASK_STRING', 'IP_ADDRESS_STRING';
5740: 4738 TIpAddressString', 'IP_ADDRESS_STRING';
5741: 4739 PIpAddressString', 'PIP_MASK_STRING';
5742: 4740 PIP_ADDR_STRING', '^IP_ADDR_STRING // will not work';
5743: 4741 _IP_ADDR_STRING', 'record Next : PIP_ADDR_STRING; IpAddress : IP';
5744: 4742 IP_ADDR_STRING', '_IP_ADDR_STRING';
5745: 4743 TIpAddressString', 'IP_ADDR_STRING';
5746: 4744 PIpAddressString', 'PIP_ADDR_STRING';
5747: 4745 PIP_ADAPTER_INFO', '^IP_ADAPTER_INFO // will not work';
5748: 4746 IP_ADAPTER_INFO', '_IP_ADAPTER_INFO';
5749: 4747 TIpAddressString', 'IP_ADAPTER_INFO';
5750: 4748 PIP_ADAPTER_INFO', 'PIP_ADAPTER_INFO';
5751: 4749 LPSERVICE_DESCRIPTIONA', '^SERVICE_DESCRIPTIONA // will not work';
5752: 4750 _SERVICE_DESCRIPTIONA', 'record lpDescription : string; end';
5753: 4751 SERVICE_DESCRIPTIONA', '_SERVICE_DESCRIPTIONA';
5754: 4752 TServiceDescriptionA', 'SERVICE_DESCRIPTIONA';
5755: 4753 PServiceDescriptionA', 'LPSERVICE_DESCRIPTIONA';
5756: 4754 LPSERVICE_DESCRIPTIONW', '^SERVICE_DESCRIPTIONW // will not work';
5757: 4755 _SERVICE_DESCRIPTIONW', 'record lpDescription : LPWSTR; end';
5758: 4756 SERVICE_DESCRIPTIONW', '_SERVICE_DESCRIPTIONW';
5759: 4757 TServiceDescriptionW', 'SERVICE_DESCRIPTIONW';
5760: 4758 PServiceDescriptionW', 'LPSERVICE_DESCRIPTIONW';
5761: 4759 SERVICE_DESCRIPTION', 'SERVICE_DESCRIPTIONW';
5762: 4760 LPSERVICE_DESCRIPTION', 'LPSERVICE_DESCRIPTIONW';
5763: 4761 TServiceDescription', 'TServiceDescriptionW';
5764: 4762 PServiceDescription', 'PServiceDescriptionW';

```

```

5765: 4763 SERVICE_DESCRIPTION', 'SERVICE_DESCRIPTIONA');
5766: 4764 LPSERVICE_DESCRIPTION', 'LPSERVICE_DESCRIPTIONA');
5767: 4765 TServiceDescription', 'TServiceDescriptionA');
5768: 4766 PServiceDescription', 'PServiceDescriptionA');
5769: 4767 LPSERVICE_STATUS_PROCESS', '^SERVICE_STATUS_PROCESS // will not '
5770: 4768 _SERVICE_STATUS_PROCESS', 'record dwServiceType : DWORD; dwCurre'
5771: 4769 SERVICE_STATUS_PROCESS', '_SERVICE_STATUS_PROCESS');
5772: 4770 TServiceStatusProcess', 'SERVICE_STATUS_PROCESS');
5773: 4771 PServiceStatusProcess', 'LPSERVICE_STATUS_PROCESS');
5774: 4772 _SC_ENUM_TYPE', '( SC_ENUM_PROCESS_INFO )';
5775: 4773 SC_ENUM_TYPE', '_SC_ENUM_TYPE');
5776: 4774 SC_HANDLE', 'THandle');
5777: 4775 _ENUM_SERVICE_STATUS_PROCESSA', 'record lpServiceName : LPSTR; l'
5778: 4776 ENUM_SERVICE_STATUS_PROCESSA', '_ENUM_SERVICE_STATUS_PROCESSA');
5779: 4777 TEnumServiceStatusProcessA', 'ENUM_SERVICE_STATUS_PROCESSA');
5780: 4778 PEnumServiceStatusProcessA', 'LPENUM_SERVICE_STATUS_PROCESSA');
5781: 4779 _ENUM_SERVICE_STATUS_PROCESSW', 'record lpServiceName : LPWSTR; '
5782: 4780 ENUM_SERVICE_STATUS_PROCESSW', '_ENUM_SERVICE_STATUS_PROCESSW');
5783: 4781 TEnumServiceStatusProcessW', 'ENUM_SERVICE_STATUS_PROCESSW');
5784: 4782 PEnumServiceStatusProcessW', 'LPENUM_SERVICE_STATUS_PROCESSW');
5785: 4783 ENUM_SERVICE_STATUS_PROCESS', 'ENUM_SERVICE_STATUS_PROCESSW');
5786: 4784 LPENUM_SERVICE_STATUS_PROCESS', 'LPENUM_SERVICE_STATUS_PROCESSW');
5787: 4785 TEnumServiceStatusProcess', 'TEnumServiceStatusProcessW');
5788: 4786 PEnumServiceStatusProcess', 'PEnumServiceStatusProcessW');
5789: 4787 ENUM_SERVICE_STATUS_PROCESS', 'ENUM_SERVICE_STATUS_PROCESSA');
5790: 4788 LPENUM_SERVICE_STATUS_PROCESS', 'LPENUM_SERVICE_STATUS_PROCESSA');
5791: 4789 TEnumServiceStatusProcess', 'TEnumServiceStatusProcessA');
5792: 4790 PEnumServiceStatusProcess', 'PEnumServiceStatusProcessA'); *)
5793: 4790 ***** File 310: F:\maxbox3\mX397\maxbox3\source\REST\upsi_XmlVerySimple.pas
5794: 4791 TXmlOnNodeSetText', 'Procedure(Sender: TObject; Node : TXmlNode; Text : String)';
5795: 4792
5796:
5797: (*-----*)
5798: procedure SIRegister_JclStreams(CL: TPSPascalCompiler);
5799: begin
5800:   //TSeekOrigin', '( soBeginning, soCurrent, soEnd )';
5801:   CL.AddClassN(CL.FindClass('TObject'), 'EJclStreamError');
5802:   SIRegister_TJclStream(CL);
5803:   SIRegister_TJclHandleStream(CL);
5804:   SIRegister_TJclFileStream(CL);
5805:   SIRegister_TJclEmptyStream(CL);
5806:   SIRegister_TJclNullStream(CL);
5807:   SIRegister_TJclRandomStream(CL);
5808:   SIRegister_TJclMultiplexStream(CL);
5809:   SIRegister_TJclStreamDecorator(CL);
5810:   SIRegister_TJclBufferedStream(CL);
5811:   TStreamNotifyEvent', 'Procedure ( Sender : TObject; Position : Int64; Size : Int64)';
5812:   SIRegister_TJclEventStream(CL);
5813:   SIRegister_TJclEasyStream(CL);
5814:   SIRegister_TJclScopedStream(CL);
5815:   TJclStreamSeekEvent', 'Function ( Sender : TObject; const Offset'
5816:     + ' : Int64; Origin : TSeekOrigin) : Int64';
5817:   TJclStreamReadEvent', 'Function ( Sender : TObject; var Buffer, '
5818:     + 'Offset, Count : Longint) : Longint';
5819:   TJclStreamWriteEvent', 'Function ( Sender : TObject; const Buffe'
5820:     + 'r, Offset, Count : Longint) : Longint';
5821:   TJclStreamSizeEvent', 'Procedure ( Sender : TObject; const NewSize : Int64)';
5822:   SIRegister_TJclDelegatedStream(CL);
5823:   SIRegister_TJclSectorStream(CL);
5824:   SIRegister_TJclCRC16Stream(CL);
5825:   SIRegister_TJclCRC32Stream(CL);
5826:   SIRegister_TJclSplitStream(CL);
5827:   TJclVolumeEvent', 'Function ( Index : Integer) : TStream';
5828:   TJclVolumeMaxSizeEvent', 'Function ( Index : Integer) : Int64';
5829:   SIRegister_TJclDynamicSplitStream(CL);
5830:   TJclSplitVolume', 'record MaxSize : Int64; Stream : TStream; OwnStream : Boolean; end';
5831:   //PJclSplitVolume', '^TJclSplitVolume // will not work';
5832:   SIRegister_TJclStaticSplitStream(CL);
5833: end;
5834:
5835: (*-----*)
5836: procedure SIRegister_SimpleXML(CL: TPSPascalCompiler);
5837: begin
5838:   'BinXmlSignatureSize', 'Integer').SetInt(BinXmlSignatureSize);
5839:   'BinXmlSignature', 'String').SetString(BinXmlSignature);
5840:   BINXML_USE_WIDE_CHARS', 'LongInt').SetInt( 1);
5841:   'BINXML_COMPRESSED', 'LongInt').SetInt( 2);
5842:   'DefaultHashSize', 'LongInt').SetInt( 499);
5843:   'AnsiCodepage', 'Integer').SetInt( CP_ACP);
5844:   'XSTR_NULL', 'String').SetString( '{{null}}');
5845:   'SourceBufferSize', 'LongWord').SetUInt( $4000);
5846:   PXmlChar', 'PChar');
5847:   TXmlChar', 'Char');
5848:   TXmlString', 'String');
5849:   RawByteString', 'AnsiString');
5850:   NativeInt', 'Integer');
5851: end
5852:
5853: TEDILoopStackRecord', 'record SegmentId : string; SpecStartIndex'

```

```

5854: + Integer; OwnerLoopId: string; ParentLoopId: string; EDIObject: TEDIObject; EDISpecObject: TEDIObject;
5855: end');
5856: TEDILoopStackArray', 'array of TEDILoopStackRecord');
5857: TEDILoopStackFlags', '( ediAltStackPointer, ediStackResized, ediLoopRepeated )');
5858: TEDILoopStackFlagSet', 'set of TEDILoopStackFlags');
5859: TEDILoopStackOnAddLoopEvent', 'Procedure ( StackRecord : TEDILoop
5860: +pStackRecord; SegmentId, OwnerLoopId, ParentLoopId : string; var EDIObject: TEDIObject)');
5861: (*-----*)
5862: procedure SIRegister_Barcode(CL: TPSPascalCompiler);
5863: begin
5864: 'mmPerInch', 'Extended').setExtended( 25.4);
5865: TBarcodeType', '( bcCode_2_5_interleaved, bcCode_2_5_industrial,
5866: + ' bcCode_2_5_matrix, bcCode39, bcCode39Extended, bcCode128A, bcCode128B, bc
5867: + 'Code128C, bcCode93, bcCode93Extended, bcCodeMSI, bcCodePostNet, bcCodeCoda
5868: + 'bar, bcCodeEAN8, bcCodeEAN13, bcCodeUPC_A, bcCodeUPC_E0, bcCodeUPC_E1, bcC
5869: + 'odeUPC_Supp2, bcCodeUPC_Supp5, bcCodeEAN128A, bcCodeEAN128B, bcCodeEAN128C)';
5870: TBarLineType', '( white, black, black_half )');
5871: TBarcodeOption', '( bcoNone, bcoCode, bcoTyp, bcoBoth )');
5872: TShowTextPosition', '( stpTopLeft, stpTopRight, stpTopCenter, st
5873: + 'pBottomLeft, stpBottomRight, stpBottomCenter )');
5874: TCheckSumMethod', '( csmNone, csmModulo10 )');
5875: SIRegister_TAsBarcode(CL);
5876: end;
5877: 'INVALID_IP_ADDRESS', 'LongWord').SetUInt( $ffffff);
5878: t_encoding', '( uuencode, base64, mime )');
5879: 'ShowSetupDialogOptLong', 'String').SetString( '--setup');
5880: 'PrimaryConfPathOptLong', 'String').SetString( '--primary-config-path=');
5881: PrimaryConfPathOptShort', 'String').SetString( '--pcp=');
5882: SecondaryConfPathOptLong', 'String').SetString( '--secondary-config-path=');
5883: SecondaryConfPathOptShort', 'String').SetString( '--scp=');
5884: NoSplashScreenOptLong', 'String').SetString( '--no-splash-screen');
5885: NoSplashScreenOptShort', 'String').SetString( '--nsc');
5886: StartedByStartLazarusOpt', 'String').SetString( '--started-by-startlazarus');
5887: SkipLastProjectOpt', 'String').SetString( '--skip-last-project');
5888: DebugLogOpt', 'String').SetString( '--debug-log=');
5889: DebugLogOptEnable', 'String').SetString( '--debug-enable=');
5890: LanguageOpt', 'String').SetString( '--language=');
5891: LazarusDirOpt', 'String').SetString( '--lazarusdir=');
5892:
5893: procedure SIRegister_SynEditMiscClasses(CL: TPSPascalCompiler);
5894: begin
5895: SIRegister_TSynSelectedColor(CL);
5896: TSynGutterBorderStyle', '( gbsNone, gbsMiddle, gbsRight )');
5897: CL.AddClassN(CL.FindClass('TObject'), 'ESynMethodChain');
5898: TSynExceptionEvent', 'Procedure ( Sender : TObject; E : Exceptio
5899: + 'n; var DoContinue : Boolean)';
5900: TSynBorderStyle', 'TBorderStyle');
5901: THKModifier', '( hkShift, hkCtrl, hkAlt )');
5902: THKModifiers', 'set of THKModifier');
5903: THKInvalidKey', '( hcNone, hcShift, hcCtrl, hcAlt, hcShiftCtrl,
5904: + 'hcShiftAlt, hcCtrlAlt, hcShiftCtrlAlt )');
5905: THKInvalidKeys', 'set of THKInvalidKey');
5906: TBetterRegistry', 'TRegistry');
5907: SIRegister_TSynEditMark(CL);
5908: TSynEditLineMarks', 'array[0..16] of TSynEditMark');
5909: SIRegister_TSynEditMarkList(CL);
5910: end;
5911:
5912: procedure SIRegister_LibTar(CL: TPSPascalCompiler);
5913: begin
5914: TTarPermission', '( tpReadByOwner, tpWriteByOwner, tpExecuteByOw
5915: + 'ner, tpReadByGroup, tpWriteByGroup, tpExecuteByGroup, tpReadByOther, tpWri
5916: + 'teByOther, tpExecuteByOther )';
5917: CL.AddTypeS(TTarPermissions', 'set of TTarPermission');
5918: CL.AddTypeS(TTarFileType', '( ftNormal, ftLink, ftSymbolicLink, ftCharacter, ft
5919: + 'Block, ftDirectory, ftFifo, ftContiguous, ftDumpDir, ftMultiVolume, ftVolumeHeader )');
5920: CL.AddTypeS(TTarMode', '( tmSetUid, tmSetGid, tmSaveText )');
5921: CL.AddTypeS(TTarModes', 'set of TTarMode');
5922: CL.AddTypeS(TTarDirRec', 'record Name : STRING; Size : INT64; DateTime : TDa
5923: + 'teTime; Permissions : TTarPermissions; FileType : TTarFileType; LinkName : ST
5924: + 'RING; UID : INTEGER; GID : INTEGER; UserName : STRING; GroupName : STRING;
5925: + 'ChecksumOK : BOOLEAN; Mode : TTarModes; Magic : STRING; MajorDevNo : INTE
5926: + 'GER; MinorDevNo : INTEGER; FilePos : INT64; end');
5927: ALL_PERMISSIONS', 'LongInt').Value.ts32 := ord(tpReadByOwner) or ord(tpWriteByOwner) or
ord(tpExecuteByOwner) or ord(tpReadByGroup) or ord(tpWriteByGroup) or ord(tpExecuteByGroup) or
ord(tpReadByOther) or ord(tpWriteByOther) or ord(tpExecuteByOther);
5928: READ_PERMISSIONS', 'LongInt').Value.ts32 := ord(tpReadByOwner) or ord(tpReadByGroup) or ord(tpReadByOther);
5929: WRITE_PERMISSIONS', 'LongInt').Value.ts32 := ord(tpWriteByOwner) or ord(tpWriteByGroup) or
ord(tpWriteByOther);
5930: EXECUTE_PERMISSIONS', 'LongInt').Value.ts32 := ord(tpExecuteByOwner) or ord(tpExecuteByGroup) or
ord(tpExecuteByOther);
5931: end;
5932:
5933: (*-----*)
5934: procedure SIRegister_IdTelnetServer(CL: TPSPascalCompiler);
5935: begin
5936: 'GLoginAttempts', 'LongInt').SetInt( 3);
5937: SIRegister_TTelnetData(CL);

```

```

5938:   SIRegister_TidTelnetPeerThread(CL);
5939:   TidTelnetNegotiateEvent', 'Procedure ( AThread : TidPeerThread)');
5940:   TAuthenticationEvent', 'Procedure ( AThread : TidPeerThread; con'
5941:   + 'st AUsername, APassword : string; var AAuthenticated : Boolean)');
5942:   SIRegister_TidTelnetServer(CL);
5943: end;
5944:
5945: (*-----*)
5946: procedure SIRegister_IdBlockCipherIntercept(CL: TPSPascalCompiler);
5947: begin
5948:   'IdBlockCipherBlockSizeDefault', 'LongInt').SetInt( 16);
5949:   'IdBlockCipherBlockSizeMax', 'LongInt').SetInt( 256);
5950:   CL.AddClassN(CL.FindClass('TOBJECT'), 'TidBlockCipherIntercept');
5951:   TidBlockCipherInterceptDataEvent', 'Procedure ( ASender : TidBlo'
5952:   + 'ckCipherIntercept; ASrcData, ADstData : __Pointer)');
5953:   SIRegister_TidBlockCipherIntercept(CL);
5954:   EIdBlockCipherInterceptException', 'EIdException');
5955: end;
5956:
5957: (*-----*)
5958: procedure SIRegister_IdStack(CL: TPSPascalCompiler);
5959: begin
5960:   //PidInAddr', '^TidInAddr // will not work');
5961:   TidInAddr', 'record S_addr: longword; end'); //from socks
5962:   // TSocket = u_int;
5963:   TSocket', 'Integer'); //from socks
5964:   TidStackSocketHandle', 'TSocket'); //from socks
5965:   //type //TidStackSocketHandle = TSocket;
5966:   TidIPType', '( Id_IPInvalid, Id_IPDotted, Id_IPNumeric )');
5967:   //PidIPType', '^TidIPType // will not work');
5968:   TidIPClass', '( Id_IPClassUnkn, Id_IPClassA, Id_IPClassB, Id_IPC'
5969:   + 'lassC, Id_IPClassD, Id_IPClassE )');
5970:   //PidIPClass', '^TidIPClass // will not work');
5971:   //TidSocketListClass', 'class of TidSocketList');
5972:   SIRegister_TidSocketList(CL);
5973:   SIRegister_TidStack(CL);
5974:   //TidStackClass', 'class of TidStack');
5975:   CL.AddClassN(CL.FindClass('TOBJECT'), 'EIdStackError');
5976:   CL.AddClassN(CL.FindClass('TOBJECT'), 'EIdStackInitializationFailed');
5977:   CL.AddClassN(CL.FindClass('TOBJECT'), 'EIdStackSetSizeExceeded');
5978:   CL.AddClassN(CL.FindClass('TOBJECT'), 'EIdInvalidIPAddress');
5979: end;
5980:
5981: (*-----*)
5982: procedure SIRegister_IdSocks(CL: TPSPascalCompiler);
5983: begin
5984:   TSocksVersion', '( svNoSocks, svSocks4, svSocks4A, svSocks5 )');
5985:   TSocksAuthentication', '( saNoAuthentication, saUsernamePassword)');
5986:   //ID_SOCKS_AUTH', 'TSocksVersion').SetSet(saNoAuthentication);
5987:   //ID_SOCKS_VER', '').SetString( svNoSocks);
5988:   //TidInAddr', 'record S_addr: longword; end');
5989:   { TidInAddr = record
5990:     case integer of
5991:       0: (S_un_b: TidSunB);
5992:       1: (S_un_w: TidSunW);
5993:       2: (S_addr: longword);
5994:     end;}
5995:   TidSocksRequest', 'record Version : Byte; OpCode : Byte; Port : '
5996:   + 'Word; IpAddr : TidInAddr; UserName : String; end');
5997:   TidSocksResponse', 'record Version : Byte; OpCode : Byte; Port : '
5998:   + 'Word; IpAddr : TidInAddr; end');
5999:   SIRegister_TidSocksInfo(CL);
6000: end;
6001:
6002: (*-----*)
6003: procedure SIRegister_IdHostnameServer(CL: TPSPascalCompiler);
6004: begin
6005:   THostNameGetEvent', 'Procedure ( Thread : TidPeerThread)');
6006:   THostNameOneParmEvent', 'Procedure ( Thread : TidPeerThread; Parm : String)');
6007:   SIRegister_TidHostNameServer(CL);
6008: end;
6009:
6010: procedure SIRegister_IdTunnelSlave(CL: TPSPascalCompiler);
6011: begin
6012:   CL.AddClassN(CL.FindClass('TOBJECT'), 'TSlaveThread');
6013:   CL.AddClassN(CL.FindClass('TOBJECT'), 'TidTunnelSlave');
6014:   TTunnelEvent', 'Procedure ( Thread : TSlaveThread)');
6015:   SIRegister_TClientData(CL);
6016:   SIRegister_TSlaveThread(CL);
6017:   SIRegister_TidTunnelSlave(CL);
6018: end;
6019: (*-----*)
6020: procedure SIRegister_IdTunnelMaster(CL: TPSPascalCompiler);
6021: begin
6022:   CL.AddClassN(CL.FindClass('TOBJECT'), 'TidTunnelMaster');
6023:   SIRegister_MClientThread(CL);
6024:   SIRegister_TSlaveData(CL);
6025:   CL.AddTypeS(TidSendMsgEvent', 'Procedure ( Thread : TidPeerThread; var CustomMsg : String)');
6026:   CL.AddTypeS(TidSendTrnEvent', 'Procedure (Thread:TidPeerThread; var Header:TidHeader;var CustomMsg:
String)');

```



```

6027: CL.AddTypeS(TIdSendTrnEventC', 'Procedure ( var Header : TIdHeader; var CustomMsg : String)');
6028: CL.AddTypeS(TIdTunnelEventC', 'Procedure ( Receiver : TReceiver)');
6029: CL.AddTypeS(TIdSendMsgEventC', 'Procedure ( var CustomMsg : String)');
6030: SIRegister_TIdTunnelMaster(CL);
6031: end;
6032: (*-----*)
6033: procedure SIRegister_IdTunnelCommon(CL: TPSPascalCompiler);
6034: begin
6035:   BUFFERLEN, 'LongWord').SetUInt( $4000);
6036:   'NumberOfConnectionsType', 'LongInt').SetInt( 1);
6037:   'NumberOfPacketsType', 'LongInt').SetInt( 2);
6038:   'CompressionRatioType', 'LongInt').SetInt( 3);
6039:   'CompressedBytesType', 'LongInt').SetInt( 4);
6040:   'BytesReadType', 'LongInt').SetInt( 5);
6041:   'BytesWriteType', 'LongInt').SetInt( 6);
6042:   'NumberOfClientsType', 'LongInt').SetInt( 7);
6043:   'NumberOfSlavesType', 'LongInt').SetInt( 8);
6044:   'NumberOfServicesType', 'LongInt').SetInt( 9);
6045:   'tmError', 'LongInt').SetInt( 0);
6046:   'tmData', 'LongInt').SetInt( 1);
6047:   'tmDisconnect', 'LongInt').SetInt( 2);
6048:   'tmConnect', 'LongInt').SetInt( 3);
6049:   'tmCustom', 'LongInt').SetInt( 99);
6050:   TIdStatisticsOperation', '( soIncrease, soDecrease )');
6051:   TIdHeader', 'record CRC16 : Word; MsgType : Word; MsgLen : Word;
6052:   + ' UserId : Word; Port : Word; IpAddr : TIdInAddr; end');
6053:   SIRegister_TReceiver(CL);
6054:   SIRegister_TSender(CL);
6055:   SIRegister_TLogger(CL);
6056:   CL.AddTypeS(TSendMsgEvent', 'Procedure ( Thread : TIdPeerThread; var CustomMsg : String)');
6057:   CL.AddTypeS(TSendTrnEvent', 'Procedure ( Thread : TIdPeerThread; var Header '
6058:   + ': TIdHeader; var CustomMsg : String)');
6059:   CL.AddTypeS(TSendTrnEventC', 'Procedure ( var Header : TIdHeader; var CustomMsg : String)');
6060:   CL.AddTypeS(TTunnelEventC', 'Procedure ( Receiver : TReceiver)');
6061:   CL.AddTypeS(TSendMsgEventC', 'Procedure ( var CustomMsg : String)');
6062:   CL.AddClassN(CL.FindClass('TObJECT'), 'EIdTunnelException');
6063:   TObJECT'), 'EIdTunnelTransformErrorBeforeSend');
6064:   TObJECT'), 'EIdTunnelTransformError');
6065:   TObJECT'), 'EIdTunnelConnectToMasterFailed');
6066:   TObJECT'), 'EIdTunnelDontAllowConnections');
6067:   TObJECT'), 'EIdTunnelCRCFailed');
6068:   TObJECT'), 'EIdTunnelMessageTypeRecognitionError');
6069:   TObJECT'), 'EIdTunnelMessageHandlingFailed');
6070:   TObJECT'), 'EIdTunnelInterpretationOfMessageFailed');
6071:   TObJECT'), 'EIdTunnelCustomMessageInterpretationFailure');
6072: end;
6073:
6074: procedure SIRegister_IdDateTimeStamp(CL: TPSPascalCompiler);
6075: begin
6076:   'IdMilliSecondsInSecond', 'LongInt').SetInt( 1000);
6077:   'IdSecondsInMinute', 'LongInt').SetInt( 60);
6078:   'IdMinutesInHour', 'LongInt').SetInt( 60);
6079:   'IdHoursInDay', 'LongInt').SetInt( 24);
6080:   'IdDaysInWeek', 'LongInt').SetInt( 7);
6081:   'IdDaysInYear', 'LongInt').SetInt( 365);
6082:   'IdDaysInLeapYear', 'LongInt').SetInt( 366);
6083:   'IdYearsInShortLeapYearCycle', 'LongInt').SetInt( 4);
6084:   // 'IdDaysInFourYears', ''').SetString( IdDaysInShortLeapYearCycle);
6085:   'IdYearsInCentury', 'LongInt').SetInt( 100);
6086:   'IdDaysInLeapCentury', 'LongInt').SetInt( IdDaysInCentury + 1);
6087:   'IdYearsInLeapYearCycle', 'LongInt').SetInt( 400);
6088:   'IdDaysInLeapYearCycle', 'LongInt').SetInt( IdDaysInCentury * 4 + 1);
6089:   'IdMonthsInYear', 'LongInt').SetInt( 12);
6090:   'IdBeatsInDay', 'LongInt').SetInt( 1000);
6091:   'IdHoursInHalfDay', 'LongInt').SetInt( IdHoursInDay div 2);
6092:   'TZ_NZDT', 'LongInt').SetInt( 13);
6093:   'TZ_IDLE', 'LongInt').SetInt( 12);
6094:   'TZ_NZST', 'longint').Setint(12);
6095:   'TZ_NZT', 'longint').Setint(12);
6096:   'TZ_EADT', 'LongInt').SetInt( 11);
6097:   'TZ_GST', 'LongInt').SetInt( 10);
6098:   'TZ_JST', 'LongInt').SetInt( 9);
6099:   'TZ_CCT', 'LongInt').SetInt( 8);
6100:   'TZ_WADT', 'longint').Setint(8);
6101:   'TZ_WAST', 'LongInt').SetInt( 7);
6102:   'TZ_ZP6', 'LongInt').SetInt( 6);
6103:   'TZ_ZP5', 'LongInt').SetInt( 5);
6104:   'TZ_ZP4', 'LongInt').SetInt( 4);
6105:   'TZ_BT', 'LongInt').SetInt( 3);
6106:   'TZ_FET', 'LongInt').SetInt( 2);
6107:   'TZ_MEST', 'longint').Setint(2);
6108:   'TZ_MESZ', 'longint').Setint(2);
6109:   'TZ_SST', 'longint').Setint(2);
6110:   'TZ_FST', 'longint').Setint(2);
6111:   'TZ_CET', 'LongInt').SetInt(1);
6112:   'TZ_FWT', 'longint').Setint(1);
6113:   'TZ_MET', 'longint').Setint(1);
6114:   'TZ_MEWT', 'longint').Setint(1);
6115:   'TZ_SWT', 'longint').Setint(1);

```

```

6116: 'TZ_GMT','LongInt').SetInt( 0);
6117: 'TZ_UT','LongInt').SetInt(0);
6118: 'TZ_UTC','LongInt').setInt(0);
6119: 'TZ_WET','LongInt').setInt(0);
6120: 'TZ_WAT','LongInt').SetInt( - 1);
6121: 'TZ_BST','LongInt').setInt(-1);
6122: 'TZ_AT','LongInt').SetInt( - 2);
6123: 'TZ_ADT','LongInt').SetInt( - 3);
6124: 'TZ_AST','LongInt').SetInt( - 4);
6125: 'TZ_EDT','LongInt').setInt(-4);
6126: 'TZ_EST','LongInt').SetInt( - 5);
6127: 'TZ_CDT','LongInt').setInt(-5);
6128: 'TZ_CST','LongInt').SetInt( - 6);
6129: 'TZ_MDT','LongInt').setInt(-6);
6130: 'TZ_MST','LongInt').SetInt( - 7);
6131: 'TZ_PDT','LongInt').setInt(-7);
6132: 'TZ_PST','LongInt').SetInt( - 8);
6133: 'TZ_YDT','LongInt').setInt(-8);
6134: 'TZ_YST','LongInt').SetInt( - 9);
6135: 'TZ_HDT','LongInt').setInt(-9);
6136: 'TZ_AHST','LongInt').SetInt( - 10);
6137: 'TZ_CAT','LongInt').setInt(-10);
6138: 'TZ_HST','LongInt').setInt(-10);
6139: 'TZ_EAST','LongInt').setInt(-10);
6140: 'TZ_NT','LongInt').SetInt( - 11);
6141: 'TZ_IDLW','LongInt').SetInt( - 12);
6142: 'TzM_A','LongInt').setInt( TZ_WAT);
6143: 'TzM_Alpha','LongInt').setInt( TzM_A);
6144: 'TzM_B','LongInt').setInt( TZ_AT);
6145: 'TzM_Bravo','LongInt').setInt( TzM_B);
6146: 'TzM_C','LongInt').setInt( TZ_ADT);
6147: 'TzM_Charlie','LongInt').setInt( TzM_C);
6148: 'TzM_D','LongInt').setInt( TZ_AST);
6149: 'TzM_Delta','LongInt').setInt( TzM_D);
6150: 'TzM_E','LongInt').setInt( TZ_EST);
6151: 'TzM_Echo','LongInt').setInt( TzM_E);
6152: 'TzM_F','LongInt').setInt( TZ_CST);
6153: 'TzM_Foxtrot','LongInt').setInt( TzM_F);
6154: 'TzM_G','LongInt').setInt( TZ_MST);
6155: 'TzM_Golf','LongInt').setInt( TzM_G);
6156: 'TzM_H','LongInt').setInt( TZ_PST);
6157: 'TzM_Hotel','LongInt').setInt( TzM_H);
6158: 'TzM_J','LongInt').setInt( TZ_YST);
6159: 'TzM_Juliet','LongInt').setInt( TzM_J);
6160: 'TzM_K','LongInt').setInt( TZ_AHST);
6161: 'TzM_Kilo','LongInt').setInt( TzM_K);
6162: 'TzM_L','LongInt').setInt( TZ_NT);
6163: 'TzM_Lima','LongInt').setInt( TzM_L);
6164: 'TzM_M','LongInt').setInt( TZ_IDLW);
6165: 'TzM_Mike','LongInt').setInt( TzM_M);
6166: 'TzM_N','LongInt').setInt( TZ_CET);
6167: 'TzM_November','LongInt').setInt( TzM_N);
6168: 'TzM_O','LongInt').setInt( TZ_EET);
6169: 'TzM_Oscar','LongInt').setInt( TzM_O);
6170: 'TzM_P','LongInt').setInt( TZ_BT);
6171: 'TzM_Papa','LongInt').setInt( TzM_P);
6172: 'TzM_Q','LongInt').setInt( TZ_ZP4);
6173: 'TzM_Quebec','LongInt').setInt( TzM_Q);
6174: 'TzM_R','LongInt').setInt( TZ_ZP5);
6175: 'TzM_Romeo','LongInt').setInt( TzM_R);
6176: 'TzM_S','LongInt').setInt( TZ_ZP6);
6177: 'TzM_Sierra','LongInt').setInt( TzM_S);
6178: 'TzM_T','LongInt').setInt( TZ_WAST);
6179: 'TzM_Tango','LongInt').setInt( TzM_T);
6180: 'TzM_U','LongInt').setInt( TZ_CCT);
6181: 'TzM_Uniform','LongInt').setInt( TzM_U);
6182: 'TzM_V','LongInt').setInt( TZ_JST);
6183: 'TzM_Victor','LongInt').setInt( TzM_V);
6184: 'TzM_W','LongInt').setInt( TZ_GST);
6185: 'TzM_Whiskey','LongInt').setInt( TzM_W);
6186: 'TzM_X','LongInt').setInt( TZ_NT);
6187: 'TzM_XRay','LongInt').setInt( TzM_X);
6188: 'TzM_Y','LongInt').setInt( TZ_IDLE);
6189: 'TzM_Yankee','LongInt').setInt( TzM_Y);
6190: 'TzM_Z','LongInt').setInt( TZ_GMT);
6191: 'TzM_Zulu','LongInt').setInt( TzM_Z);
6192: TDays, '( TDaySun, TDayMon, TDayTue, TDayWed, TDayThu, TDayFri, TDaySat )';
6193: TMonths, '( TMthJan, TMthFeb, TMthMar, TMthApr, TMthMay, TMthJu
6194:   + 'n, TMthJul, TMthAug, TMthSep, TMthOct, TMthNov, TMthDec )';
6195: SIRegister_TiDTimeStamp(CL);
6196: end;
6197:
6198: procedure SIRegister_BoldComUtils(CL: TPSPascalCompiler);
6199: begin
6200:   'RPC_C_AUTHN_LEVEL_DEFAULT','LongInt').SetInt( 0);
6201:   'RPC_C_AUTHN_LEVEL_NONE','LongInt').SetInt( 1);
6202:   'RPC_C_AUTHN_LEVEL_CONNECT','LongInt').SetInt( 2);
6203:   'RPC_C_AUTHN_LEVEL_CALL','LongInt').SetInt( 3);
6204:   'RPC_C_AUTHN_LEVEL_PKT','LongInt').SetInt( 4);

```

```

6205: 'RPC_C_AUTHN_LEVEL_PKT_INTEGRITY', 'LongInt').SetInt( 5);
6206: 'RPC_C_AUTHN_LEVEL_PKT_PRIVACY', 'LongInt').SetInt( 6);
6207: {'alDefault', '1').SetString( RPC_C_AUTHN_LEVEL_DEFAULT);
6208: 'alNone', '2').SetString( RPC_C_AUTHN_LEVEL_NONE);
6209: 'alConnect', '3').SetString( RPC_C_AUTHN_LEVEL_CONNECT);
6210: 'alCall', '4').SetString( RPC_C_AUTHN_LEVEL_CALL);
6211: 'alPacket', '5').SetString( RPC_C_AUTHN_LEVEL_PKT);
6212: 'alPacketIntegrity', '6').SetString( RPC_C_AUTHN_LEVEL_PKT_INTEGRITY);
6213: 'alPacketPrivacy', '7').SetString( RPC_C_AUTHN_LEVEL_PKT_PRIVACY);}
6214: 'RPC_C_IMP_LEVEL_DEFAULT', 'LongInt').SetInt( 0);
6215: 'RPC_C_IMP_LEVEL_ANONYMOUS', 'LongInt').SetInt( 1);
6216: 'RPC_C_IMP_LEVEL_IDENTIFY', 'LongInt').SetInt( 2);
6217: 'RPC_C_IMP_LEVEL_IMPERSONATE', 'LongInt').SetInt( 3);
6218: 'RPC_C_IMP_LEVEL_DELEGATE', 'LongInt').SetInt( 4);
6219: {'ilDefault', '0').SetString( RPC_C_IMP_LEVEL_DEFAULT);
6220: 'ilAnonymous', '1').SetString( RPC_C_IMP_LEVEL_ANONYMOUS);
6221: 'ilIdentiry', '2').SetString( RPC_C_IMP_LEVEL_IDENTIFY);
6222: 'ilImpersonate', '3').SetString( RPC_C_IMP_LEVEL_IMPERSONATE);
6223: 'ilDelegate', '4').SetString( RPC_C_IMP_LEVEL_DELEGATE);}
6224: 'EOAC_NONE', 'LongWord').SetUInt( $0);
6225: 'EOAC_DEFAULT', 'LongWord').SetUInt( $800);
6226: 'EOAC_MUTUAL_AUTH', 'LongWord').SetUInt( $1);
6227: 'EOAC_STATIC_CLOACKING', 'LongWord').SetUInt( $20);
6228: 'EOAC_DYNAMIC_CLOACKING', 'LongWord').SetUInt( $40);
6229: 'EOAC_ANY_AUTHORITY', 'LongWord').SetUInt( $80);
6230: 'RPC_C_AUTHN_WINNT', 'LongInt').SetInt( 10);
6231: 'RPC_C_AUTHNZ_NONE', 'LongInt').SetInt( 0);
6232: 'RPC_C_AUTHNZ_NAME', 'LongInt').SetInt( 1);
6233: 'RPC_C_AUTHNZ_DCE', 'LongInt').SetInt( 2);
6234:
6235:
6236: procedure SIRegister_Geometry(CL: TPSPascalCompiler); //openGL
6237: begin
6238:   THomogeneousByteVector, 'array[0..3] of Byte';
6239:   THomogeneousWordVector, 'array[0..3] of Word';
6240:   THomogeneousIntVector, 'array[0..3] of Integer';
6241:   THomogeneousFltVector, 'array[0..3] of single';
6242:   THomogeneousDblVector, 'array[0..3] of double';
6243:   THomogeneousExtVector, 'array[0..3] of extended';
6244:   TAffineByteVector, 'array[0..2] of Byte';
6245:   TAffineWordVector, 'array[0..2] of Word';
6246:   TAffineIntVector, 'array[0..2] of Integer';
6247:   TAffineFltVector, 'array[0..2] of single';
6248:   TAffineDblVector, 'array[0..2] of double';
6249:   TAffineExtVector, 'array[0..2] of extended';
6250:
6251:   THomogeneousByteMatrix, 'array[0..3] of THomogeneousByteVector';
6252:   THomogeneousWordMatrix, 'array[0..3] of THomogeneousWordVector';
6253:   THomogeneousIntMatrix, 'array[0..3] of THomogeneousIntVector';
6254:   THomogeneousFltMatrix, 'array[0..3] of THomogeneousFltVector';
6255:   THomogeneousDblMatrix, 'array[0..3] of THomogeneousDblVector';
6256:   THomogeneousExtMatrix, 'array[0..3] of THomogeneousExtVector';
6257:
6258:   TAffineByteMatrix, 'array[0..2] of TAffineByteVector';
6259:   TAffineWordMatrix, 'array[0..2] of TAffineWordVector';
6260:   TAffineIntMatrix, 'array[0..2] of TAffineIntVector';
6261:   TAffineFltMatrix, 'array[0..3] of TAffineFltVector';
6262:   TAffineDblMatrix, 'array[0..3] of TAffineDblVector';
6263:   TAffineExtMatrix, 'array[0..3] of TAffineExtVector';
6264:
6265:   //THomogeneousByteMatrix = array[0..3] of THomogeneousByteVector;
6266:   //THomogeneousByteVector = array[0..3] of Byte;
6267:   //THomogeneousWordVector = array[0..3] of Word;
6268:   // THomogeneousFltVector = array[0..3] of Single;
6269:   // TAffineByteVector = array[0..2] of Byte;
6270:
6271:   TVector4b, 'THomogeneousByteVector';
6272:   TVector4w, 'THomogeneousWordVector';
6273:   TVector4i, 'THomogeneousIntVector';
6274:   TVector4f, 'THomogeneousFltVector';
6275:   TVector4d, 'THomogeneousDblVector';
6276:   TVector4e, 'THomogeneousExtVector';
6277:   TVector3b, 'TAffineByteVector';
6278:   TVector3w, 'TAffineWordVector';
6279:   TVector3i, 'TAffineIntVector';
6280:   TVector3f, 'TAffineFltVector';
6281:   TVector3d, 'TAffineDblVector';
6282:   TVector3e, 'TAffineExtVector';
6283:   TVectorGL, 'THomogeneousFltVector';
6284:   THomogeneousVector, 'THomogeneousFltVector';
6285:   TAffineVector, 'TAffineFltVector';
6286:   TMatrix4b, 'THomogeneousByteMatrix';
6287:   TMatrix4w, 'THomogeneousWordMatrix';
6288:   TMatrix4i, 'THomogeneousIntMatrix';
6289:   TMatrix4f, 'THomogeneousFltMatrix';
6290:   TMatrix4d, 'THomogeneousDblMatrix';
6291:   TMatrix4e, 'THomogeneousExtMatrix';
6292:   TMatrix3b, 'TAffineByteMatrix';
6293:   TMatrix3w, 'TAffineWordMatrix';

```

```

6294:   TMatrix3i', 'TAffineIntMatrix');
6295:   TMatrix3f', 'TAffineFltMatrix');
6296:   TMatrix3d', 'TAffineDblMatrix');
6297:   TMatrix3e', 'TAffineExtMatrix');
6298:   TMatrixGL', 'THomogeneousFltMatrix');
6299:   THomogeneousMatrix', 'THomogeneousFltMatrix');
6300:   //PAffineMatrix', '^TAffineMatrix // will not work';
6301:   TAffineMatrix', 'TAffineFltMatrix');
6302:   TQuaternion', 'record Vector : TVector4f; end';
6303:   TRectangle', 'record Left : integer; Top : integer; Width : inte'
6304:   + 'ger; Height : Integer; end';
6305:   TTransType', '( ttScaleX, ttScaleY, ttScaleZ, ttShearXY, ttShear'
6306:   + 'XZ, ttShearYZ, ttRotateX, ttRotateY, ttRotateZ, ttTranslateX, ttTranslateY'
6307:   + ', ttTranslateZ, ttPerspectiveX, ttPerspectiveY, ttPerspectiveZ, ttPerspectiveW )';
6308:   'EPSILON', 'Extended').setExtended( 1E-100);
6309:   'EPSILON2', 'Extended').setExtended( 1E-50);
6310: end;
6311:
6312: (*-----*)
6313: procedure SIRegister_Textures(CL: TPSPascalCompiler);
6314: begin
6315:   MYINT1', 'LongInt').SetInt( 2147483647);
6316:   MYINT2', 'LongWord').SetUInt( 4294967295);
6317:   MYLONGWORD', 'LongWord').SetUInt( 4294967295);
6318:   TColorRGB', 'record r : byte; g : byte; b : BYTE; end';
6319:   //PColorRGB', '^TColorRGB // will not work';
6320:   //pRGBList', '^TRGBList // will not work';
6321:   TColorRGBA', 'record r : byte; g : byte; b : byte; a : BYTE; end';
6322:   //PColorRGBA', '^TColorRGBA // will not work';
6323:   //PRGBAList', '^TRGBAList // will not work';
6324:   SIRegister_TTexture(CL);
6325:   SIRegister_TVoidTexture(CL);
6326:   SIRegister_TTextureRGBA(CL);
6327:   SIRegister_TVoidTextureRGBA(CL);
6328: end;
6329:
6330:
6331: procedure SIRegister_BoldQueue(CL: TPSPascalCompiler);
6332: begin
6333:   //befIsInDisplayList', '').SetString( BoldElementFlag0);
6334:   //befStronglyDependedOfPrioritized', '').SetString( BoldElementFlag1);
6335:   //befFollowerSelected', '').SetString( BoldElementFlag2);
6336:   CL.AddClassN(CL.FindClass('TOBJECT'), 'TBoldQueue');
6337:   CL.AddClassN(CL.FindClass('TOBJECT'), 'TBoldQueueable');
6338:   TBoldQueueDisplayMode', '( dmDisplayOne, dmDisplayAll )';
6339:   SIRegister_TBoldQueueable(CL);
6340:   SIRegister_TBoldQueue(CL);
6341:   CL.AddDelphiFunction('Function BoldQueueFinalized : Boolean');
6342:   CL.AddDelphiFunction('Function BoldInstalledQueue : TBoldQueue');
6343: end;
6344:
6345: (*-----*)
6346: procedure SIRegister_JvSysComp(CL: TPSPascalCompiler);
6347: begin
6348:   CCPS_BufferSize', 'LongInt').SetInt( 1024);
6349:   CCPS_MaxBufferSize', 'LongInt').SetInt( 65536);
6350:   'EJvProcessError', 'EJclError';
6351:   'TJvProcessPriority', '( ppIdle, ppNormal, ppHigh, ppRealTime )';
6352:   'TJvConsoleOption', '( coOwnerData, coRedirect )';
6353:   'TJvConsoleOptions', 'set of TJvConsoleOption';
6354:   'TJvCPSReadEvent', 'Procedure (Sender: TObject; const S : string)';
6355:   'TJvCPSTerminateEvent', 'Procedure (Sender : TObject; ExitCode: DWORD)';
6356:   'TJvRWHandles', 'record Read : THandle; Write : THandle; end';
6357:   'TJvRWEHandles', 'record Read : THandle; Write : THandle; Error : '
6358:   + ' THandle; end';
6359:   SIRegister_TJvProcessEntry(CL);
6360:   'TJvCPSState', '( psReady, psRunning, psWaiting )';
6361:   'TJvCPSFlag', '( cfDefaultErrorMode, cfNewConsole, cfNewProcGroup'
6362:   + ', cfSeparateWdm, cfSharedWdm, cfSuspended, cfUnicode, cfDetached )';
6363:   'TJvCPSFlags', 'set of TJvCPSFlag';
6364:   'TJvCPSShowWindow', '( swHide, swMinimize, swMaximize, swNormal )';
6365:   SIRegister_TJvCPSStartupInfo(CL);
6366:   SIRegister_TJvCreateProcess(CL);
6367: end;
6368:
6369: (*-----*)
6370: procedure SIRegister_JvWinampApi(CL: TPSPascalCompiler);
6371: begin
6372:   'TWStatus', '( wsNotAvailable, wsStopped, wsPlaying, wsPaused )';
6373:   'TJvPlayOption', '( poRepeat, poShuffle )';
6374:   'TJvPlayOptions', 'set of TJvPlayOption';
6375:   SIRegister_TJvWinampApi(CL);
6376:   CL.AddClassN(CL.FindClass('TOBJECT'), 'EWinampError');
6377: end;
6378:
6379:
6380: (*-----*)
6381: procedure SIRegister_StStrms(CL: TPSPascalCompiler);
6382: begin

```



```

6383:   TStMemSize', 'Integer');
6384:   SIRegister_TStBufferedStream(CL);
6385:   SIRegister_TStAnsiTextStream(CL);
6386:   SIRegister_TStMemoryMappedFile(CL);
6387:   TStreamReader', 'TStAnsiTextStream');
6388: end;
6389:
6390: (*-----*)
6391: procedure SIRegister_JclMiscel2(CL: TPSPascalCompiler);
6392: begin
6393:   TJclKillLevel', '( klNormal, klNoSignal, klTimeOut )';
6394:   TJclAllowedPowerOperation', '( apoHibernate, apoShutdown, apoSuspend )';
6395:   TJclAllowedPowerOperations', 'set of TJclAllowedPowerOperation';
6396:   TObject', 'EJclCreateProcessError';
6397:   // with CL.Add(EJclCreateProcessError) do
6398: end;
6399:
6400: (*-----*)
6401: procedure SIRegister_Chart(CL: TPSPascalCompiler);
6402: begin
6403:   TeeMsg_DefaultFunctionName', 'String'.SetString( 'TeeFunction');
6404:   TeeMsg_DefaultSeriesName', 'String'.SetString( 'Series');
6405:   TeeMsg_DefaultToolName', 'String'.SetString( 'ChartTool');
6406:   ChartComponentPalette', 'String'.SetString( 'TeeChart');
6407:   TeeMaxLegendColumns', 'LongInt'.SetInt( 2);
6408:   TeeDefaultLegendSymbolWidth', 'LongInt'.SetInt( 20);
6409:   TeeTitleFootDistance', 'LongInt'.SetInt( 5);
6410:   SIRegister_TCustomChartWall(CL);
6411:   SIRegister_TChartWall(CL);
6412:   SIRegister_TChartLegendGradient(CL);
6413:   TLegendStyle', '( lsAuto, lsSeries, lsValues, lsLastValues, lsSeriesGroups )';
6414:   TLegendAlignment', '( laLeft, laRight, laTop, laBottom )';
6415:   CL.AddClassN(CL.FindClass('TObject'), 'LegendException');
6416:   TOnGetLegendText', 'Procedure ( Sender : TCustomAxisPanel; Legen
6417:   + 'dStyle : TLegendStyle; Index : Integer; var LegendText : String)';
6418:   CL.AddClassN(CL.FindClass('TObject'), 'TCustomChartLegend');
6419:   TLegendSymbolSize', '( lcsPercent, lcsPixels )';
6420:   TLegendSymbolPosition', '( spLeft, spRight )';
6421:   TSymbolDrawEvent', 'Procedure (Sender : TObject; Series : TChartSeries; ValueIndex : Integer; R :
TRect)';
6422:   TSymbolCalcHeight', 'Function : Integer';
6423:   SIRegister_TLegendSymbol(CL);
6424:   SIRegister_TTeeCustomShapePosition(CL);
6425:   TCheckBoxesStyle', '( cbsCheck, cbsRadio )';
6426:   SIRegister_TLegendTitle(CL);
6427:   SIRegister_TLegendItem(CL);
6428:   SIRegister_TLegendItems(CL);
6429:   TLegendCalcSize', 'Procedure ( Sender : TCustomChartLegend; var ASize : Integer)';
6430:   CL.AddClassN(CL.FindClass('TObject'), 'TCustomChart');
6431:   SIRegister_TCustomChartLegend(CL);
6432:   SIRegister_TChartLegend(CL);
6433:   SIRegister_TChartTitle(CL);
6434:   SIRegister_TChartFootTitle(CL);
6435:   TChartClick', 'Procedure ( Sender : TCustomChart; Button : TMouse
6436:   + 'eButton; Shift : TShiftState; X, Y : Integer)';
6437:   TChartClickAxis', 'Procedure ( Sender : TCustomChart; Axis : TCh
6438:   + 'artAxis; Button : TMouseButton; Shift : TShiftState; X, Y : Integer)';
6439:   TChartClickSeries', 'Procedure ( Sender : TCustomChart; Series :
6440:   + ' TChartSeries; ValueIndex : Integer; Button : TMouseButton; Shift : TShiftState; X, Y : Integer)';
6441:   TChartClickTitle', 'Procedure ( Sender : TCustomChart; ATitle :
6442:   + 'TChartTitle; Button : TMouseButton; Shift : TShiftState; X, Y : Integer)';
6443:   TOnGetLegendPos', 'Procedure ( Sender : TCustomChart; Index : In
6444:   + 'teger; var X, Y, XColor : Integer)';
6445:   TOnGetLegendRect', 'Procedure ( Sender : TCustomChart; var Rect : TRect)';
6446:   TAxisSavedScales', 'record Auto : Boolean; AutoMin : Boolean; Au
6447:   + 'toMax : Boolean; Min : Double; Max : Double; end';
6448:   TAllAxisSavedScales', 'array of TAxisSavedScales';
6449:   SIRegister_TChartBackWall(CL);
6450:   SIRegister_TChartRightWall(CL);
6451:   SIRegister_TChartBottomWall(CL);
6452:   SIRegister_TChartLeftWall(CL);
6453:   SIRegister_TChartWalls(CL);
6454:   TChartAllowScrollEvent', 'Procedure (Sender:TChartAxis;var AMin, AMax:Double;var AllowScroll: Boolean)';
6455:   SIRegister_TCustomChart(CL);
6456:   SIRegister_TChart(CL);
6457:   SIRegister_TTeeSeriesTypes(CL);
6458:   SIRegister_TTeeToolTypes(CL);
6459:   SIRegister_TTeeDragObject(CL);
6460:   SIRegister_TColorPalettes(CL);
6461: end;
6462:
6463:
6464: procedure SIRegister_TeeProcs(CL: TPSPascalCompiler);
6465: begin
6466:   //TeeFormBorderStyle', '').SetString( bsNone);
6467:   SIRegister_TMetafile(CL);
6468:   'TeeDefVerticalMargin', 'LongInt'.SetInt( 4);
6469:   'TeeDefHorizMargin', 'LongInt'.SetInt( 3);
6470:   'crTeeHand', 'LongInt'.SetInt( TCursor ( 2020 ));

```

```

6471: 'TeeMsg_TeeHand','String').SetString( 'crTeeHand');
6472: 'TeeNormalPrintDetail','LongInt').SetInt( 0);
6473: 'TeeHighPrintDetail','LongInt').SetInt( - 100);
6474: 'TeeDefault_PrintMargin','LongInt').SetInt( 15);
6475: 'MaxDefaultColors','LongInt').SetInt( 19);
6476: 'TeeTabDelimiter','Char').SetString( #9);
6477: TDateTimeStep', '( dtOneMicroSecond, dtOneMillisecond, dtOneSeco'
6478: + 'nd, dtFiveSeconds, dtTenSeconds, dtFifteenSeconds, dtThirtySeconds, dtOneM'
6479: + 'inute, dtFiveMinutes, dtTenMinutes, dtFifteenMinutes, dtThirtyMinutes, dtO'
6480: + 'neHour, dtTwoHours, dtSixHours, dtTwelveHours, dtOneDay, dtTwoDays, dtThre'
6481: + 'eDays, dtOneWeek, dtHalfMonth, dtOneMonth, dtTwoMonths, dtThreeMonths, dtF'
6482: + 'ourMonths, dtSixMonths, dtOneYear, dtNone )';
6483: SIRegister_TCustomPanelNoCaption(CL);
6484: CL.AddClassN(CL.FindClass('TOBJECT'),'TCustomTeePanel');
6485: SIRegister_TZoomPanning(CL);
6486: SIRegister_TTeeEvent(CL);
6487: //SIRegister_TTeeEventListeners(CL);
6488: TTeeMouseEventKind', '( meDown, meUp, meMove )';
6489: SIRegister_TTeeMouseEvent(CL);
6490: SIRegister_TCustomTeePanel(CL);
6491: //TChartGradient', 'TGradient';
6492: //TChartGradientClass', 'class of TChartGradient';
6493: TPanningMode', '( pmNone, pmHorizontal, pmVertical, pmBoth )';
6494: SIRegister_TTeeZoomPen(CL);
6495: SIRegister_TTeeZoomBrush(CL);
6496: TTeeZoomDirection', '( tzdHorizontal, tzdVertical, tzdBoth )';
6497: SIRegister_TTeeZoom(CL);
6498: CL.AddClassN(CL.FindClass('TOBJECT'),'TCustomTeePanelExtended');
6499: TTeeBackImageMode', '( pbmStretch, pbmTile, pbmCenter, pbmCustom )';
6500: SIRegister_TBackImage(CL);
6501: SIRegister_TCustomTeePanelExtended(CL);
6502: //TChartBrushClass', 'class of TChartBrush';
6503: SIRegister_TTeeCustomShapeBrushPen(CL);
6504: TChartObjectShapeStyle', '( fosRectangle, fosRoundRectangle, fosEllipse )';
6505: TTextFormat', '( ttfNormal, ttfHtml )';
6506: SIRegister_TTeeCustomShape(CL);
6507: SIRegister_TTeeShape(CL);
6508: SIRegister_TTeeExportData(CL);
6509: TTeeSortCompare', 'Function ( a, b : Integer) : Integer';
6510: TTeeSortSwap', 'Procedure ( a, b : Integer)';
6511: end;
6512:
6513: (*-----*)
6514: procedure SIRegister_D2XXUnit(CL: TPSPascalCompiler);
6515: begin
6516:   FT_Result', 'Integer');
6517:   //TDWordptr', '^DWord // will not work';
6518:   TFT_Program_Data', 'record Signature1 : DWord; Signature2 : DWor'
6519:   + 'd; Version : DWord; VendorID : Word; ProductID : Word; Manufacturer : PChar'
6520:   + 'r; ManufacturerID : PChar; Description : PChar; SerialNumber : PChar; MaxP'
6521:   + 'ower : Word; PnP : Word; SelfPowered : Word; RemoteWakeup : Word; Rev4 : B'
6522:   + 'yte; IsoIn : Byte; IsoOut : Byte; PullDownEnable : Byte; SerNumEnable : By'
6523:   + 'te; USBVersionEnable : Byte; USBVersion : Word; Rev5 : Byte; IsoInA : Byte'
6524:   + 'e; IsoInB : Byte; IsoOutA : Byte; IsoOutB : Byte; PullDownEnable5 : Byte; S'
6525:   + 'erNumEnable5 : Byte; USBVersionEnable5 : Byte; USBVersion5 : Word; AIsHigh'
6526:   + 'Current : Byte; BIsHighCurrent : Byte; IFAIsFifo : Byte; IFAIsFifoTar : B'
6527:   + 'yte; IFAIsFastSer : Byte; AIsVCP : Byte; IFBIsFifo : Byte; IFBIsFifoTar : B'
6528:   + 'yte; IFBIsFastSer : Byte; BIsVCP : Byte; UseExtOsc : Byte; HighDriveIOs : '
6529:   + 'Byte; EndpointSize : Byte; PullDownEnableR : Byte; SerNumEnableR : Byte; I'
6530:   + 'nvertDTR : Byte; InvertDSR : Byte; InvertRTS : Byte; InvertCTS : Byte; Inv'
6531:   + 'ertDTR : Byte; InvertDSR : Byte; InvertDCD : Byte; InvertRI : Byte; Cbus0 '
6532:   + ' : Byte; Cbus1 : Byte; Cbus2 : Byte; Cbus3 : Byte; Cbus4 : Byte; RIsVCP : B'
6533:   + 'yte; end';
6534:   //CL.AddDelphiFunction('Function Get_USB_Driver_Version( DrVersion : TDWordptr) : FT_Result');
6535:   //CL.AddDelphiFunction('Function Get_USB_Library_Version( LbVersion : TDWordptr) : FT_Result');
6536:   'FT_OK','LongInt').SetInt( 0);
6537:   FT_INVALID_HANDLE','LongInt').SetInt( 1);
6538:   FT_DEVICE_NOT_FOUND','LongInt').SetInt( 2);
6539:   FT_DEVICE_NOT_OPENED','LongInt').SetInt( 3);
6540:   FT_IO_ERROR','LongInt').SetInt( 4);
6541:   FT_INSUFFICIENT_RESOURCES','LongInt').SetInt( 5);
6542:   FT_INVALID_PARAMETER','LongInt').SetInt( 6);
6543:   FT_SUCCESS','integer').SetInt( FT_OK);
6544:   FT_INVALID_BAUD_RATE','LongInt').SetInt( 7);
6545:   FT_DEVICE_NOT_OPENED_FOR_ERASE','LongInt').SetInt( 8);
6546:   FT_DEVICE_NOT_OPENED_FOR_WRITE','LongInt').SetInt( 9);
6547:   FT_FAILED_TO_WRITE_DEVICE','LongInt').SetInt( 10);
6548:   FT_EEPROM_READ_FAILED','LongInt').SetInt( 11);
6549:   FT_EEPROM_WRITE_FAILED','LongInt').SetInt( 12);
6550:   FT_EEPROM_ERASE_FAILED','LongInt').SetInt( 13);
6551:   FT_EEPROM_NOT_PRESENT','LongInt').SetInt( 14);
6552:   FT_EEPROM_NOT_PROGRAMMED','LongInt').SetInt( 15);
6553:   FT_INVALID_ARGS','LongInt').SetInt( 16);
6554:   FT_OTHER_ERROR','LongInt').SetInt( 17);
6555:   FT_OPEN_BY_SERIAL_NUMBER','LongInt').SetInt( 1);
6556:   FT_OPEN_BY_DESCRIPTION','LongInt').SetInt( 2);
6557:   FT_OPEN_BY_LOCATION','LongInt').SetInt( 4);
6558:   FT_LIST_NUMBER_ONLY','LongWord').SetUInt( $80000000);
6559:   FT_LIST_BY_INDEX','LongWord').SetUInt( $40000000);

```

```

6560: FT_LIST_ALL', 'LongWord').SetUInt( $20000000);
6561: FT_BAUD_300', 'LongInt').SetInt( 300);
6562: FT_BAUD_600', 'LongInt').SetInt( 600);
6563: FT_BAUD_1200', 'LongInt').SetInt( 1200);
6564: FT_BAUD_2400', 'LongInt').SetInt( 2400);
6565: FT_BAUD_4800', 'LongInt').SetInt( 4800);
6566: FT_BAUD_9600', 'LongInt').SetInt( 9600);
6567: FT_BAUD_14400', 'LongInt').SetInt( 14400);
6568: FT_BAUD_19200', 'LongInt').SetInt( 19200);
6569: FT_BAUD_38400', 'LongInt').SetInt( 38400);
6570: FT_BAUD_57600', 'LongInt').SetInt( 57600);
6571: FT_BAUD_115200', 'LongInt').SetInt( 115200);
6572: FT_BAUD_230400', 'LongInt').SetInt( 230400);
6573: FT_BAUD_460800', 'LongInt').SetInt( 460800);
6574: FT_BAUD_921600', 'LongInt').SetInt( 921600);
6575: FT_DATA_BITS_7', 'LongInt').SetInt( 7);
6576: FT_DATA_BITS_8', 'LongInt').SetInt( 8);
6577: FT_STOP_BITS_1', 'LongInt').SetInt( 0);
6578: FT_STOP_BITS_2', 'LongInt').SetInt( 2);
6579: FT_PARITY_NONE', 'LongInt').SetInt( 0);
6580: FT_PARITY_ODD', 'LongInt').SetInt( 1);
6581: FT_PARITY_EVEN', 'LongInt').SetInt( 2);
6582: FT_PARITY_MARK', 'LongInt').SetInt( 3);
6583: FT_PARITY_SPACE', 'LongInt').SetInt( 4);
6584: FT_FLOW_NONE', 'LongWord').SetUInt( $0000);
6585: FT_FLOW_RTS_CTS', 'LongWord').SetUInt( $0100);
6586: FT_FLOW_DTR_DSR', 'LongWord').SetUInt( $0200);
6587: FT_FLOW_XON_XOFF', 'LongWord').SetUInt( $0400);
6588: FT_PURGE_RX', 'LongInt').SetInt( 1);
6589: FT_PURGE_TX', 'LongInt').SetInt( 2);
6590: FT_EVENT_RXCHAR', 'LongInt').SetInt( 1);
6591: FT_EVENT_MODEM_STATUS', 'LongInt').SetInt( 2);
6592: CTS', 'LongWord').SetUInt( $10);
6593: DSR', 'LongWord').SetUInt( $20);
6594: RI', 'LongWord').SetUInt( $40);
6595: DCD', 'LongWord').SetUInt( $80);
6596: FT_In_Buffer_Size', 'LongWord').SetUInt( $10000);
6597: FT_In_Buffer_Index', 'LongInt').SetInt( FT_In_Buffer_Size - 1);
6598: FT_Out_Buffer_Size', 'LongWord').SetUInt( $10000);
6599: FT_Out_Buffer_Index', 'LongInt').SetInt( FT_Out_Buffer_Size - 1);
6600: FT_DLL_Name', 'String').SetString( 'FTD2XX.DLL');
6601: end;
6602:
6603: procedure SIRegister_IdRawHeaders(CL: TPSPascalCompiler);
6604: begin
6605:   TIdNetTime', 'longword');
6606:   Id_ARP_HSIZE', 'LongWord').SetUInt( $1C);
6607:   Id_DNS_HSIZE', 'LongWord').SetUInt( $0C);
6608:   Id_ETH_HSIZE', 'LongWord').SetUInt( $0E);
6609:   Id_ICMP_HSIZE', 'LongWord').SetUInt( $04);
6610:   Id_ICMP_ECHO_HSIZE', 'LongWord').SetUInt( $08);
6611:   Id_ICMP_MASK_HSIZE', 'LongWord').SetUInt( $0C);
6612:   Id_ICMP_UNREACH_HSIZE', 'LongWord').SetUInt( $08);
6613:   Id_ICMP_TIMEEXCEED_HSIZE', 'LongWord').SetUInt( $08);
6614:   Id_ICMP_REDIRECT_HSIZE', 'LongWord').SetUInt( $08);
6615:   Id_ICMP_TS_HSIZE', 'LongWord').SetUInt( $14);
6616:   Id_IGMP_HSIZE', 'LongWord').SetUInt( $08);
6617:   Id_IP_HSIZE', 'LongWord').SetUInt( $14);
6618:   Id_RIP_HSIZE', 'LongWord').SetUInt( $18);
6619:   Id_TCP_HSIZE', 'LongWord').SetUInt( $14);
6620:   Id_UDP_HSIZE', 'LongWord').SetUInt( $08);
6621:   Id_MAX_IPOPTLEN', 'LongInt').SetInt( 40);
6622:   Id_IP_RF', 'LongWord').SetUInt( $8000);
6623:   Id_IP_DF', 'LongWord').SetUInt( $4000);
6624:   Id_IP_MF', 'LongWord').SetUInt( $2000);
6625:   Id_IP_OFFMASK', 'LongWord').SetUInt( $1FFF);
6626:   //PidIpHdr', '^TIdIpHdr // will not work;
6627:   TIdIpHdr', 'record ip_verlen : byte; ip_tos : byte; ip_len : word;
6628:     + 'd; ip_id : word; ip_off : word; ip_ttl : byte; ip_p : byte; ip_sum : word;
6629:     + ' ip_src : TIdInAddr; ip_dst : TIdInAddr; ip_options : longword; end;
6630:   Id_IP_MAXPACKET', 'LongInt').SetInt( 65535);
6631:   Id_TCP_FIN', 'LongWord').SetUInt( $01);
6632:   Id_TCP_SYN', 'LongWord').SetUInt( $02);
6633:   Id_TCP_RST', 'LongWord').SetUInt( $04);
6634:   Id_TCP_PUSH', 'LongWord').SetUInt( $08);
6635:   Id_TCP_ACK', 'LongWord').SetUInt( $10);
6636:   Id_TCP_URG', 'LongWord').SetUInt( $20);
6637:   // PidTcpHdr', '^TIdTcpHdr // will not work;
6638:   TIdTcpHdr', 'record tcp_sport : word; tcp_dport : word; tcp_seq :
6639:     + ' : longword; tcp_ack : longword; tcp_x2off : byte; tcp_flags : byte; tcp_wi
6640:     + 'n : word; tcp_sum : word; tcp_urp : word; end;
6641:   // PidUdpHdr', '^TIdUdpHdr // will not work;
6642:   TIdUdpHdr', 'record udp_sport : word; udp_dport : word; udp_ulen
6643:     + ' : word; udp_sum : word; end;
6644:   TIdEtherAddr', 'record ether_addr_octet : array [0..5] of byte; end;
6645:
6646:   {TIdEtherAddr = packed record
6647:     ether_addr_octet : array [0..Id_ETHER_ADDR_LEN-1] of byte;}
6648:

```

```

6649: Id_ICMP_ECHOREPLY', 'LongInt.SetInt( 0);
6650: Id_ICMP_UNREACH', 'LongInt.SetInt( 3);
6651: Id_ICMP_SOURCEQUENCH', 'LongInt.SetInt( 4);
6652: Id_ICMP_REDIRECT', 'LongInt.SetInt( 5);
6653: Id_ICMP_ECHO', 'LongInt.SetInt( 8);
6654: Id_ICMP_ROUTERADVERT', 'LongInt.SetInt( 9);
6655: Id_ICMP_ROUTERSOLICIT', 'LongInt.SetInt( 10);
6656: Id_ICMP_TIMXCEED', 'LongInt.SetInt( 11);
6657: Id_ICMP_PARAMPROB', 'LongInt.SetInt( 12);
6658: Id_ICMP_TSTAMP', 'LongInt.SetInt( 13);
6659: Id_ICMP_TSTAMPREPLY', 'LongInt.SetInt( 14);
6660: Id_ICMP_IREQ', 'LongInt.SetInt( 15);
6661: Id_ICMP_IREQREPLY', 'LongInt.SetInt( 16);
6662: Id_ICMP_MASKREQ', 'LongInt.SetInt( 17);
6663: Id_ICMP_MASKREPLY', 'LongInt.SetInt( 18);
6664: Id_ICMP_UNREACH_NET', 'LongInt.SetInt( 0);
6665: Id_ICMP_UNREACH_HOST', 'LongInt.SetInt( 1);
6666: Id_ICMP_UNREACH_PROTOCOL', 'LongInt.SetInt( 2);
6667: Id_ICMP_UNREACH_PORT', 'LongInt.SetInt( 3);
6668: Id_ICMP_UNREACH_NEEDFRAG', 'LongInt.SetInt( 4);
6669: Id_ICMP_UNREACH_SRCFAIL', 'LongInt.SetInt( 5);
6670: Id_ICMP_UNREACH_NET_UNKNOWN', 'LongInt.SetInt( 6);
6671: Id_ICMP_UNREACH_HOST_UNKNOWN', 'LongInt.SetInt( 7);
6672: Id_ICMP_UNREACH_ISOLATED', 'LongInt.SetInt( 8);
6673: Id_ICMP_UNREACH_NET_PROHIB', 'LongInt.SetInt( 9);
6674: Id_ICMP_UNREACH_HOST_PROHIB', 'LongInt.SetInt( 10);
6675: Id_ICMP_UNREACH_TOSNET', 'LongInt.SetInt( 11);
6676: Id_ICMP_UNREACH_TOSHST', 'LongInt.SetInt( 12);
6677: Id_ICMP_UNREACH_FILTER_PROHIB', 'LongInt.SetInt( 13);
6678: Id_ICMP_UNREACH_HOST_PRECEDENCE', 'LongInt.SetInt( 14);
6679: Id_ICMP_UNREACH_PRECEDENCE_CUTOFF', 'LongInt.SetInt( 15);
6680: Id_ICMP_REDIRECT_NET', 'LongInt.SetInt( 0);
6681: Id_ICMP_REDIRECT_HOST', 'LongInt.SetInt( 1);
6682: Id_ICMP_REDIRECT_TOSNET', 'LongInt.SetInt( 2);
6683: Id_ICMP_REDIRECT_TOSHST', 'LongInt.SetInt( 3);
6684: Id_ICMP_TIMXCEED_INTRANS', 'LongInt.SetInt( 0);
6685: Id_ICMP_TIMXCEED_REASS', 'LongInt.SetInt( 1);
6686: Id_ICMP_PARAMPROB_OPTABSENT', 'LongInt.SetInt( 1);
6687: // PidIcmpEcho', '^TidIcmpEcho // will not work;
6688:   TidIcmpEcho', 'record id : word; seq : word; end;
6689: //PidIcmpFrag', '^TidIcmpFrag // will not work;
6690:   TidIcmpFrag', 'record pad : word; mtu : word; end;
6691: //PidIcmpTs', '^TidIcmpTs // will not work;
6692:   TidIcmpTs', 'record otime : TidNetTime; rtime : TidNetTime; ttime : TidNetTime; end;
6693: Id_ICMP_MEMBERSHIP_QUERY', 'LongWord.SetUInt( $11);
6694: Id_IGMP_V1_MEMBERSHIP_REPORT', 'LongWord.SetUInt( $12);
6695: Id_IGMP_V2_MEMBERSHIP_REPORT', 'LongWord.SetUInt( $16);
6696: Id_IGMP_LEAVE_GROUP', 'LongWord.SetUInt( $17);
6697: // PidIgmphdr', '^TidIgmphdr // will not work;
6698:   TidIgmphdr', 'record igmp_type : byte; igmp_code : byte; igmp_su'
6699:     + 'm : word; igmp_group : TidInAddr; end;
6700: Id_ETHER_ADDR_LEN', 'LongInt.SetInt( 6);
6701: Id_ETHERTYPE_PUP', 'LongWord.SetUInt( $0200);
6702: Id_ETHERTYPE_IP', 'LongWord.SetUInt( $0800);
6703: Id_ETHERTYPE_ARP', 'LongWord.SetUInt( $0806);
6704: Id_ETHERTYPE_REVARP', 'LongWord.SetUInt( $8035);
6705: Id_ETHERTYPE_VLAN', 'LongWord.SetUInt( $8100);
6706: Id_ETHERTYPE_LOOPBACK', 'LongWord.SetUInt( $9000);
6707: //PidEthernetHdr', '^TidEthernetHdr // will not work;
6708:   TidEthernetHdr', 'record ether_dhost : TidEtherAddr; ether_shost'
6709:     + ' : TidEtherAddr; ether_type : word; end;
6710: Id_ARPHRD_ETHER', 'LongInt.SetInt( 1);
6711: Id_ARPOP_REQUEST', 'LongInt.SetInt( 1);
6712: Id_ARPOP_REPLY', 'LongInt.SetInt( 2);
6713: Id_ARPOP_REVREQUEST', 'LongInt.SetInt( 3);
6714: Id_ARPOP_REVREPLY', 'LongInt.SetInt( 4);
6715: Id_ARPOP_INVREQUEST', 'LongInt.SetInt( 8);
6716: Id_ARPOP_INVREPLY', 'LongInt.SetInt( 9);
6717: //PidArpHdr', '^TidArpHdr // will not work;
6718:   CL.AddTypes(TidArpHdr', 'record arp_hrd : word; arp_pro : word; arp_hln : by'
6719:     + 'te; arp_pln : byte; arp_op : word; arp_sha : TidEtherAddr; arp_spa : TidIn'
6720:     + 'Addr; arp_tha : TidEtherAddr; arp_tpa : TidInAddr; end;
6721: //PidDnsHdr', '^TidDnsHdr // will not work;
6722:   TidDnsHdr', 'record dns_id : word; dns_flags : word; dns_num_q : '
6723:     + 'word; dns_num_answ_rr : word; dns_num_auth_rr : word; dns_num_addi_rr : word; end;
6724: Id_RIPCMD_REQUEST', 'LongInt.SetInt( 1);
6725: Id_RIPCMD_RESPONSE', 'LongInt.SetInt( 2);
6726: Id_RIPCMD_TRACEON', 'LongInt.SetInt( 3);
6727: Id_RIPCMD_TRACEOFF', 'LongInt.SetInt( 4);
6728: Id_RIPCMD_POLL', 'LongInt.SetInt( 5);
6729: Id_RIPCMD_POLLENTRY', 'LongInt.SetInt( 6);
6730: Id_RIPCMD_MAX', 'LongInt.SetInt( 7);
6731: Id_RIPVER_0', 'LongInt.SetInt( 0);
6732: Id_RIPVER_1', 'LongInt.SetInt( 1);
6733: Id_RIPVER_2', 'LongInt.SetInt( 2);
6734: // PidRipHdr', '^TidRipHdr // will not work;
6735:   TidRipHdr', 'record rip_cmd : byte; rip_ver : byte; rip_rd : wor'
6736:     + 'd; rip_af : word; rip_rt : word; rip_addr : longword; rip_mask : longword;'
6737:     + ' rip_next_hop : longword; rip_metric : longword; end;

```



```

6738: end;
6739:
6740: (*-----*)
6741: procedure SIRegister_CtlPanel(CL: TPSPascalCompiler);
6742: begin
6743:   TObject('EAppletException');
6744:   TInitEvent', 'Procedure ( Sender : TObject; var AppInitOK : Boolean)';
6745:   TCountEvent', 'Procedure ( Sender : TObject; var AppCount : Integer)';
6746:   TExitEvent', 'TNotifyEvent';
6747:   TSetupEvent', 'TNotifyEvent';
6748:   TActivateEvent', 'Procedure ( Sender : TObject; Data : Longint)';
6749:   TStopEvent', 'Procedure ( Sender : TObject; Data : Longint)';
6750:   TInquireEvent', 'Procedure ( Sender : TObject; var idIcon : Integer;
6751:     +ger; var idName : Integer; var idInfo : Integer; var lData : Integer)';
6752:   TNewInquireEvent', 'Procedure ( Sender : TObject; var lData : Integer;
6753:     +teger; var hIcon : HICON; var AppletName : string; var AppletInfo : string)';
6754:   TStartWParmsEvent', 'Procedure ( Sender : TObject; Params : string)';
6755:   SIRegister_TAppletModule(CL);
6756:   //TAppletModuleClass', 'class of TAppletModule';
6757:   //TCPLAppletClass', 'class of TAppletModule';
6758:   //TDataModuleClass', 'class of TDataModule';
6759:   TOnAppletExceptionEvent', 'Procedure ( Sender : TObject; E : Exception)';
6760:   SIRegister_TAppletApplication(CL);
6761:   Function CPLApplet( hwndCPL : THandle; uMsg : DWORD; lParam1, lParam2 : Longint ) : Longint);
6762: end;
6763:
6764: (*-----*)
6765: procedure SIRegister_StSpawn(CL: TPSPascalCompiler);
6766: begin
6767:   SIRegister_TStWaitThread(CL);
6768:   TStSpawnCommand', '( scOpen, scPrint, scOther )';
6769:   TStShowState', '( ssMinimized, ssMaximized, ssNormal, ssMinNotActive )';
6770:   TStSpawnErrorEvent', 'Procedure ( Sender : TObject; Error : Word)';
6771:   TStSpawnCompletedEvent', 'Procedure ( Sender : TObject)';
6772:   TStSpawnTimeOutEvent', 'Procedure ( Sender : TObject)';
6773:   SIRegister_TStSpawnApplication(CL);
6774: end;
6775:
6776: (*-----*)
6777: procedure SIRegister_SockRequestInterpreter(CL: TPSPascalCompiler);
6778: begin
6779:   SIRegister_TWebRequestDataBlockInterpreter(CL);
6780:   SIRegister_TLogWebAppDataBlockInterpreter(CL);
6781:   SIRegister_TLogSendDataBlock(CL);
6782:   asError', 'LongWord').SetUInt( $01);
6783:   asHandleRequest', 'LongWord').SetUInt( $02);
6784:   asGetFieldByName', 'LongWord').SetUInt( $03);
6785:   asReadClient', 'LongWord').SetUInt( $04);
6786:   asReadString', 'LongWord').SetUInt( $05);
6787:   asTranslateURI', 'LongWord').SetUInt( $06);
6788:   asWriteClient', 'LongWord').SetUInt( $07);
6789:   asWriteHeaders', 'LongWord').SetUInt( $08);
6790:   asUsingStub', 'LongWord').SetUInt( $09);
6791:   asGetStringVariable', 'LongWord').SetUInt( $0A);
6792: end;
6793:
6794: procedure SIRegister_ESBMaths(CL: TPSPascalCompiler);
6795: begin
6796:   ESBMinSingle', 'Single').setExtended( 1.5e-45);
6797:   Const('ESBMaxSingle', 'Single').setExtended( 3.4e+38);
6798:   Const('ESBMinDouble', 'Double').setExtended( 5.0e-324);
6799:   Const('ESBMaxDouble', 'Double').setExtended( 1.7e+308);
6800:   Const('ESBMinExtended', 'Extended').setExtended( 3.6e-4951);
6801:   Const('ESBMaxExtended', 'Extended').setExtended( 1.1e+4932);
6802:   Const('ESBMinCurrency', 'Currency').SetExtended( - 922337203685477.5807);
6803:   Const('ESBMaxCurrency', 'Currency').SetExtended( 922337203685477.5807);
6804:   Const('ESBSqrt2', 'Extended').setExtended( 1.4142135623730950488);
6805:   Const('ESBSqrt3', 'Extended').setExtended( 1.7320508075688772935);
6806:   Const('ESBSqrt5', 'Extended').setExtended( 2.2360679774997896964);
6807:   Const('ESBSqrt10', 'Extended').setExtended( 3.1622776601683793320);
6808:   Const('ESBSqrtPi', 'Extended').setExtended( 1.77245385090551602729);
6809:   Const('ESBCbrt2', 'Extended').setExtended( 1.2599210498948731648);
6810:   Const('ESBCbrt3', 'Extended').setExtended( 1.4422495703074083823);
6811:   Const('ESBCbrt10', 'Extended').setExtended( 2.1544346900318837219);
6812:   Const('ESBCbrt100', 'Extended').setExtended( 4.6415888336127788924);
6813:   Const('ESBCbrtPi', 'Extended').setExtended( 1.4645918875615232630);
6814:   Const('ESBInvSqrt2', 'Extended').setExtended( 0.70710678118654752440);
6815:   Const('ESBInvSqrt3', 'Extended').setExtended( 0.57735026918962576451);
6816:   Const('ESBInvSqrt5', 'Extended').setExtended( 0.44721359549995793928);
6817:   Const('ESBInvSqrtPi', 'Extended').setExtended( 0.56418958354775628695);
6818:   Const('ESBInvCbrtPi', 'Extended').setExtended( 0.68278406325529568147);
6819:   Const('ESBe', 'Extended').setExtended( 2.7182818284590452354);
6820:   Const('ESBe2', 'Extended').setExtended( 7.3890560989306502272);
6821:   Const('ESBePi', 'Extended').setExtended( 23.140692632779269006);
6822:   Const('ESBePion2', 'Extended').setExtended( 4.8104773809653516555);
6823:   Const('ESBePion4', 'Extended').setExtended( 2.1932800507380154566);
6824:   Const('ESBln2', 'Extended').setExtended( 0.69314718055994530942);
6825:   Const('ESBln10', 'Extended').setExtended( 2.30258509299404568402);
6826:   Const('ESBlnPi', 'Extended').setExtended( 1.14472988584940017414);

```

```

6827: Const('ESBLog10Base2','Extended').setExtended( 3.3219280948873623478);
6828: Const('ESBLog2Base10','Extended').setExtended( 0.30102999566398119521);
6829: Const('ESBLog3Base10','Extended').setExtended( 0.47712125471966243730);
6830: Const('ESBLogPiBase10','Extended').setExtended( 0.4971498726941339);
6831: Const('ESBLogEBase10','Extended').setExtended( 0.43429448190325182765);
6832: Const('ESBPi','Extended').setExtended( 3.1415926535897932385);
6833: Const('ESBInvPi','Extended').setExtended( 3.1830988618379067154e-1);
6834: Const('ESBTwoPi','Extended').setExtended( 6.2831853071795864769);
6835: Const('ESBThreePi','Extended').setExtended( 9.4247779607693797153);
6836: Const('ESBPi2','Extended').setExtended( 9.8696044010893586188);
6837: Const('ESBPiToE','Extended').setExtended( 22.459157718361045473);
6838: Const('ESBPiOn2','Extended').setExtended( 1.5707963267948966192);
6839: Const('ESBPiOn3','Extended').setExtended( 1.0471975511965977462);
6840: Const('ESBPiOn4','Extended').setExtended( 0.7853981633974483096);
6841: Const('ESBThreePiOn2','Extended').setExtended( 4.7123889803846898577);
6842: Const('ESBFourPiOn3','Extended').setExtended( 4.1887902047863909846);
6843: Const('ESBTwoToPower63','Extended').setExtended( 9223372036854775808.0);
6844: Const('ESBOneRadian','Extended').setExtended( 57.295779513082320877);
6845: Const('ESBOneDegree','Extended').setExtended( 1.7453292519943295769E-2);
6846: Const('ESBOneMinute','Extended').setExtended( 2.9088820866572159615E-4);
6847: Const('ESBOneSecond','Extended').setExtended( 4.8481368110953599359E-6);
6848: Const('ESBGamma','Extended').setExtended( 0.57721566490153286061);
6849: Const('ESBLnRt2Pi','Extended').setExtended( 9.189385332046727E-1);
6850: //LongWord', 'Cardinal');
6851:
6852: procedure SIRegister_ZEncoding(CL: TPSPascalCompiler);
6853: begin
6854:   zCP_EBC037,'LongInt').SetInt( 37);
6855:   Const('zCP_EBC273','LongInt').SetInt( 273);
6856:   Const('zCP_EBC277','LongInt').SetInt( 277);
6857:   Const('zCP_EBC278','LongInt').SetInt( 278);
6858:   Const('zCP_EBC280','LongInt').SetInt( 280);
6859:   Const('zCP_EBC284','LongInt').SetInt( 284);
6860:   Const('zCP_DOS437','LongInt').SetInt( 437);
6861:   Const('zCP_DOS500','LongInt').SetInt( 500);
6862:   Const('zCP_DOS708','LongInt').SetInt( 708);
6863:   Const('zCP_DOS709','LongInt').SetInt( 709);
6864:   Const('zCP_DOS710','LongInt').SetInt( 710);
6865:   Const('zCP_DOS720','LongInt').SetInt( 720);
6866:   Const('zCP_DOS737','LongInt').SetInt( 737);
6867:   Const('zCP_DOS775','LongInt').SetInt( 775);
6868:   Const('zCP_DOS850','LongInt').SetInt( 850);
6869:   Const('zCP_DOS851','LongInt').SetInt( 851);
6870:   Const('zCP_DOS852','LongInt').SetInt( 852);
6871:   Const('zCP_DOS853','LongInt').SetInt( 853);
6872:   Const('zCP_DOS855','LongInt').SetInt( 855);
6873:   Const('zCP_DOS856','LongInt').SetInt( 856);
6874:   Const('zCP_DOS857','LongInt').SetInt( 857);
6875:   Const('zCP_DOS858','LongInt').SetInt( 858);
6876:   Const('zCP_DOS895','LongInt').SetInt( 895);
6877:   Const('zCP_DOS860','LongInt').SetInt( 860);
6878:   Const('zCP_DOS861','LongInt').SetInt( 861);
6879:   Const('zCP_DOS862','LongInt').SetInt( 862);
6880:   Const('zCP_DOS863','LongInt').SetInt( 863);
6881:   Const('zCP_DOS864','LongInt').SetInt( 864);
6882:   Const('zCP_DOS865','LongInt').SetInt( 865);
6883:   Const('zCP_DOS866','LongInt').SetInt( 866);
6884:   Const('zCP_DOS869','LongInt').SetInt( 869);
6885:   Const('zCP_DOS870','LongInt').SetInt( 870);
6886:   Const('zCP_DOS874','LongInt').SetInt( 874);
6887:   Const('zCP_EBC875','LongInt').SetInt( 875);
6888:   Const('zCP_MSWIN921','LongInt').SetInt( 921);
6889:   Const('zCP_MSWIN923','LongInt').SetInt( 923);
6890:   Const('zCP_EBC924','LongInt').SetInt( 924);
6891:   Const('zCP_SHIFTS','LongInt').SetInt( 932);
6892:   Const('zCP_GB2312','LongInt').SetInt( 936);
6893:   Const('zCP_EUCKR','LongInt').SetInt( 949);
6894:   Const('zCP_Big5','LongInt').SetInt( 950);
6895:   Const('zCP_IBM1026','LongInt').SetInt( 1026);
6896:   Const('zCP_IBM01047','LongInt').SetInt( 1047);
6897:   Const('zCP_IBM01140','LongInt').SetInt( 1140);
6898:   Const('zCP_IBM01141','LongInt').SetInt( 1141);
6899:   Const('zCP_IBM01142','LongInt').SetInt( 1142);
6900:   Const('zCP_IBM01143','LongInt').SetInt( 1143);
6901:   Const('zCP_IBM01144','LongInt').SetInt( 1144);
6902:   Const('zCP_IBM01145','LongInt').SetInt( 1145);
6903:   Const('zCP_IBM01146','LongInt').SetInt( 1146);
6904:   Const('zCP_IBM01147','LongInt').SetInt( 1147);
6905:   Const('zCP_IBM01148','LongInt').SetInt( 1148);
6906:   Const('zCP_IBM01149','LongInt').SetInt( 1149);
6907:   Const('zCP_UTF16','LongInt').SetInt( 1200);
6908:   Const('zCP_UTF16BE','LongInt').SetInt( 1201);
6909:   Const('zCP_WIN1250','LongInt').SetInt( 1250);
6910:   Const('zCP_WIN1251','LongInt').SetInt( 1251);
6911:   Const('zCP_WIN1252','LongInt').SetInt( 1252);
6912:   Const('zCP_WIN1253','LongInt').SetInt( 1253);
6913:   Const('zCP_WIN1254','LongInt').SetInt( 1254);
6914:   Const('zCP_WIN1255','LongInt').SetInt( 1255);
6915:   Const('zCP_WIN1256','LongInt').SetInt( 1256);

```

```

6916: Const('zCP_WIN1257','LongInt').SetInt( 1257);
6917: Const('zCP_WIN1258','LongInt').SetInt( 1258);
6918: Const('ZCP_JOHAB','LongInt').SetInt( 1361);
6919: Const('zCP_KOREAN','LongInt').SetInt( 2022);
6920: Const('zCP_macintosh','LongInt').SetInt( 10000);
6921: Const('zCP_x_mac_japanese','LongInt').SetInt( 10001);
6922: Const('zCP_x_mac_chinesetrad','LongInt').SetInt( 10002);
6923: Const('zCP_x_mac_korean','LongInt').SetInt( 10003);
6924: Const('zCP_x_mac_arabic','LongInt').SetInt( 10004);
6925: Const('zCP_x_mac_hebrew','LongInt').SetInt( 10005);
6926: Const('zCP_x_mac_greek','LongInt').SetInt( 10006);
6927: Const('zCP_x_mac_cyrillic','LongInt').SetInt( 10007);
6928: Const('zCP_x_mac_chinesesimp','LongInt').SetInt( 10008);
6929: Const('zCP_x_mac_romanian','LongInt').SetInt( 10010);
6930: Const('zCP_x_mac_ukrainian','LongInt').SetInt( 10017);
6931: Const('zCP_x_mac_thai','LongInt').SetInt( 10021);
6932: Const('zCP_x_mac_ce','LongInt').SetInt( 10029);
6933: Const('zCP_x_mac_icelandic','LongInt').SetInt( 10079);
6934: Const('zCP_x_mac_turkish','LongInt').SetInt( 10081);
6935: Const('zCP_x_mac_croatian','LongInt').SetInt( 10082);
6936: Const('zCP_utf32','LongInt').SetInt( 12000);
6937: Const('zCP_utf32BE','LongInt').SetInt( 12001);
6938: Const('zCP_x_Chinese_CNS','LongInt').SetInt( 20000);
6939: Const('zCP_x_cp20001','LongInt').SetInt( 20001);
6940: Const('zCP_x_Chinese_Eten','LongInt').SetInt( 20002);
6941: Const('zCP_x_cp20003','LongInt').SetInt( 20003);
6942: Const('zCP_x_cp20004','LongInt').SetInt( 20004);
6943: Const('zCP_x_cp20005','LongInt').SetInt( 20005);
6944: Const('zCP_x_IA5','LongInt').SetInt( 20105);
6945: Const('zCP_x_IA5_German','LongInt').SetInt( 20106);
6946: Const('zCP_x_IA5_Swedish','LongInt').SetInt( 20107);
6947: Const('zCP_x_IA5_Norwegian','LongInt').SetInt( 20108);
6948: Const('zCP_us_ascii','LongInt').SetInt( 20127);
6949: Const('zCP_x_cp20261','LongInt').SetInt( 20261);
6950: Const('zCP_x_cp20269','LongInt').SetInt( 20269);
6951: Const('zCP_IBM273','LongInt').SetInt( 20273);
6952: Const('zCP_IBM277','LongInt').SetInt( 20277);
6953: Const('zCP_IBM278','LongInt').SetInt( 20278);
6954: Const('zCP_IBM280','LongInt').SetInt( 20280);
6955: Const('zCP_IBM284','LongInt').SetInt( 20284);
6956: Const('zCP_IBM285','LongInt').SetInt( 20285);
6957: Const('zCP_IBM290','LongInt').SetInt( 20290);
6958: Const('zCP_IBM297','LongInt').SetInt( 20297);
6959: Const('zCP_IBM420','LongInt').SetInt( 20420);
6960: Const('zCP_IBM423','LongInt').SetInt( 20423);
6961: Const('zCP_IBM424','LongInt').SetInt( 20424);
6962: Const('zCP_x_EBCDIC_KoreanExtended','LongInt').SetInt( 20833);
6963: Const('zCP_IBM_Thai','LongInt').SetInt( 20838);
6964: Const('zCP_KOI8R','LongInt').SetInt( 20866);
6965: Const('zCP_IBM871','LongInt').SetInt( 20871);
6966: Const('zCP_IBM880','LongInt').SetInt( 20880);
6967: Const('zCP_IBM905','LongInt').SetInt( 20905);
6968: Const('zCP_IBM00924','LongInt').SetInt( 20924);
6969: Const('zCP_EUC_JP','LongInt').SetInt( 20932);
6970: Const('zCP_x_cp20936','LongInt').SetInt( 20936);
6971: Const('zCP_x_cp20949','LongInt').SetInt( 20949);
6972: Const('zCP_cp1025','LongInt').SetInt( 21025);
6973: Const('zCP_KOI8U','LongInt').SetInt( 21866);
6974: Const('zCP_L1_ISO_8859_1','LongInt').SetInt( 28591);
6975: Const('zCP_L2_ISO_8859_2','LongInt').SetInt( 28592);
6976: Const('zCP_L3_ISO_8859_3','LongInt').SetInt( 28593);
6977: Const('zCP_L4_ISO_8859_4','LongInt').SetInt( 28594);
6978: Const('zCP_L5_ISO_8859_5','LongInt').SetInt( 28595);
6979: Const('zCP_L6_ISO_8859_6','LongInt').SetInt( 28596);
6980: Const('zCP_L7_ISO_8859_7','LongInt').SetInt( 28597);
6981: Const('zCP_L8_ISO_8859_8','LongInt').SetInt( 28598);
6982: Const('zCP_L5_ISO_8859_9','LongInt').SetInt( 28599);
6983: Const('zCP_L6_ISO_8859_10','LongInt').SetInt( 28600);
6984: Const('zCP_L7_ISO_8859_13','LongInt').SetInt( 28603);
6985: Const('zCP_L8_ISO_8859_14','LongInt').SetInt( 28604);
6986: Const('zCP_L9_ISO_8859_15','LongInt').SetInt( 28605);
6987: Const('zCP_L10_ISO_8859_16','LongInt').SetInt( 28606);
6988: Const('zCP_x_Europa','LongInt').SetInt( 29001);
6989: Const('zCP_iso_8859_8_i','LongInt').SetInt( 38598);
6990: Const('zCP_iso_2022_jp','LongInt').SetInt( 50220);
6991: Const('zCP_csISO2022JP','LongInt').SetInt( 50221);
6992: Const('zCP_x_iso_2022_jp','LongInt').SetInt( 50222);
6993: Const('zCP_iso_2022_kr','LongInt').SetInt( 50225);
6994: Const('zCP_x_cp50227','LongInt').SetInt( 50227);
6995: Const('zCP_EUC_TC_ISO220','LongInt').SetInt( 50229);
6996: Const('zCP_EBCDIC_euc_jpe','LongInt').SetInt( 50930);
6997: Const('zCP_EBCDIC_euc_jp','LongInt').SetInt( 50931);
6998: Const('zCP_euc_jp_auto','LongInt').SetInt( 50932);
6999: Const('zCP_EBCDIC_euc_kr','LongInt').SetInt( 50933);
7000: Const('zCP_EBCDIC_euc_cn','LongInt').SetInt( 50935);
7001: Const('zCP_EBCDIC_euc_sc','LongInt').SetInt( 50936);
7002: Const('zCP_EBCDIC_USC_TC','LongInt').SetInt( 50937);
7003: Const('zCP_euc_cn_auto','LongInt').SetInt( 50939);
7004: Const('zCP_euc_kr_auto','LongInt').SetInt( 50949);

```

```

7005: Const('zCP_euc_JP_win','LongInt').SetInt( 51932);
7006: Const('zCP_EUC_CN','LongInt').SetInt( 51936);
7007: Const('zCP_euc_kr','LongInt').SetInt( 51949);
7008: Const('zCP_euc_tc','LongInt').SetInt( 51950);
7009: Const('zCP_hz_gb_2312','LongInt').SetInt( 52936);
7010: Const('zCP_GB18030','LongInt').SetInt( 54936);
7011: Const('zCP_x_iscii_de','LongInt').SetInt( 57002);
7012: Const('zCP_x_iscii_be','LongInt').SetInt( 57003);
7013: Const('zCP_x_iscii_ta','LongInt').SetInt( 57004);
7014: Const('zCP_x_iscii_te','LongInt').SetInt( 57005);
7015: Const('zCP_x_iscii_as','LongInt').SetInt( 57006);
7016: Const('zCP_x_iscii_or','LongInt').SetInt( 57007);
7017: Const('zCP_x_iscii_ka','LongInt').SetInt( 57008);
7018: Const('zCP_x_iscii_ma','LongInt').SetInt( 57009);
7019: Const('zCP_x_iscii_gu','LongInt').SetInt( 57010);
7020: Const('zCP_x_iscii_pa','LongInt').SetInt( 57011);
7021: Const('zCP_UTF8','LongInt').SetInt( 65001);
7022: Const('zCP_UTF7','LongInt').SetInt( 65000);
7023: Const('zCP_NONE','LongWord').SetUInt( $ffff);
7024:
7025: procedure SIRegister_usniffer(CL: TPSPascalCompiler);
7026: begin
7027:   MAX_PACKET_SIZE,'LongWord').SetUInt( $10000);
7028:   SIO_RCVALL,'LongWord').SetUInt( $98000001);
7029:   SIRegister_TsniffForm(CL);
7030:   USHORT,'Word');
7031:   TIPHeader,'record iph_verlen : UCHAR; iph_tos : UCHAR; iph_len'
7032:   + 'gth : USHORT; iph_id : USHORT; iph_offset : USHORT; iph_ttl : UCHAR; iph_p'
7033:   + 'rotocol : UCHAR; iph_xsum : USHORT; iph_src : ULONG; iph_dest : ULONG; end');
7034:   //PIPHeader', '^TIPHeader // will not work');
7035: end;
7036:
7037: (*-----*)
7038: procedure SIRegister_IdCookie(CL: TPSPascalCompiler);
7039: begin
7040:   GfMaxAge,'LongInt').SetInt( - 1);
7041:   TIdCookieVersion,'( cvNetscape, cvRFC2109, cvRFC2965 )');
7042:   CL.AddClassN(CL.FindClass('TOBJECT'),'TIdNetscapeCookie');
7043:   SIRegister_TIdCookieList(CL);
7044:   SIRegister_TIdNetscapeCookie(CL);
7045:   SIRegister_TIdCookieRFC2109(CL);
7046:   SIRegister_TIdCookieRFC2965(CL);
7047:   SIRegister_TIdServerCookie(CL);
7048:   TIdCookieAccess,'( caRead, caReadWrite )');
7049:   SIRegister_TIdCookies(CL);
7050:   SIRegister_TIdServerCookies(CL);
7051: end;
7052:
7053: procedure SIRegister_WDosPlcUtils(CL: TPSPascalCompiler);
7054: begin
7055:   TErrorCode,'Byte');
7056:   'ecmNoError','LongInt').SetInt( 0);
7057:   'ecmWatchdogTime','LongInt').SetInt( 1);
7058:   'ecmInOutBase','LongInt').SetInt( 10);
7059:   'ecmNext','LongInt').SetInt( 18);
7060:   'ecsNoError','LongInt').SetInt( 0);
7061:   'ecsEmergencyStop','LongInt').SetInt( 1);
7062:   'ecsHardware','LongInt').SetInt( 2);
7063:   'ecsSoftware','LongInt').SetInt( 3);
7064:   'ecsParameter','LongInt').SetInt( 4);
7065:   'ecsShortCircuit','LongInt').SetInt( 5);
7066:   'ecsProtocol','LongInt').SetInt( 6);
7067:   'ecsReport','LongInt').SetInt( 7);
7068:   'ecsNext','LongInt').SetInt( 8);
7069:   'BitNum','LongInt').SetInt( 8);
7070:   'StationNum','LongInt').SetInt( 32);
7071:   'StByteNum','LongInt').SetInt( 8);
7072:   'InOutNum','LongInt').SetInt( 8);
7073:   TBitNo,'Integer');
7074:   TStByteNo,'Integer');
7075:   TStationNo,'Integer');
7076:   TInOutNo,'Integer');
7077:   TIO,'( E, A, NE, NA )');
7078:   TBitSet,'set of TBitNo');
7079:   TAddrKind,'set of ( akBit0, akBit1, akBit2, akOut, akNot, akBus )');
7080:   TBitAddrRec,'record Kind : TAddrKind; InOutNo : TInOutNo; ByteNo : Byte; end');
7081:   TBitAddr,'LongInt');
7082:   TByteAddrRec,'record Kind : TAddrKind; ByteNo : Byte; end');
7083:   TByteAddr,'SmallInt');
7084:   TInOutState,'( iosInit, iosHalt, iosRun, iosError )');
7085:   'DWatchdogTime','LongInt').SetInt( 100);
7086:   'PlcMode','( pmoInit, pmoHalt, pmoRun, pmoError )');
7087:   CL.AddClassN(CL.FindClass('TOBJECT'),'TCustomInOut');
7088:   CL.AddClassN(CL.FindClass('TOBJECT'),'TCustomPetriNet');
7089:   TPetriNetEvent,'Procedure ( Sender : TObject; aState : Integer)';
7090:   TPetriNetStateCtrl,'Procedure (Sender : TObject; var aState : Integer)';
7091:   TErrorEvent,'Procedure ( Sender : TObject; aErrorCode : TErrorCode)';
7092:
7093: procedure SIRegister_IdSysLogMessage(CL: TPSPascalCompiler);

```



```

7094: begin
7095:   TIdSyslogPRI', 'Integer');
7096:   TIdSyslogFacility', '( sfKernel, sfUserLevel, sfMailSystem, sfSy'
7097:   + 'stemDaemon, sfSecurityOne, sfSysLogInternal, sfLPR, sfNNTP, sfUUCP, sfCloc'
7098:   + 'kDaemonOne, sfSecurityTwo, sfFTPDaemon, sfNTP, sfLogAudit, sfLogAlert, sfC'
7099:   + 'lockDaemonTwo, sfLocalUseZero, sfLocalUseOne, sfLocalUseTwo, sfLocalUseThr'
7100:   + 'ee, sfLocalUseFour, sfLocalUseFive, sfLocalUseSix, sfLocalUseSeven )');
7101:   TIdSyslogSeverity', '( slEmergency, slAlert, slCritical, slError'
7102:   + ', slWarning, slNotice, slInformational, slDebug )');
7103:   SIRegister_TIdSysLogMsgPart(CL);
7104:   SIRegister_TIdSysLogMessage(CL);
7105: end;
7106:
7107: (*-----*)
7108: procedure SIRegister_StRegIni(CL: TPSPascalCompiler);
7109: begin
7110:   TRegIniType', '( riIniType, riRegType )');
7111:   TRegIniMode', '( riSet, riGet )';
7112:   TWinVerType', '( riWin31, riWin32s, riWin95, riWinNT )';
7113:   TQueryKeyInfo', 'record QIKey : HKey; QIClassName : string; QINu'
7114:   + 'mSubKeys : DWORD; QIMaxSKNLen : DWORD; QIMaxCNLen : DWORD; QINumValues : D'
7115:   + 'WORD; QIMaxVNLLen : DWORD; QIMaxDataLen : DWORD; QISDescLen : DWORD; QIFile'
7116:   + 'Time : TFileTime; end';
7117:   RI_INVALID_VALUE', 'LongInt').SetInt( - 1);
7118:   //RIVOLATILE'', '').SetString( REG_WHOLE_HIVE_VOLATILE);
7119:   ShortBufSize', 'LongInt').SetInt( 255);
7120:   MaxBufSize', 'LongInt').SetInt( 8192);
7121:   MaxByteArraySize', 'LongInt').SetInt( 127);
7122:   RIMachine', 'String').SetString( 'MACHINE');
7123:   RIUsers', 'String').SetString( 'USERS');
7124:   RIRoot', 'String').SetString( 'ROOT');
7125:   RICUser', 'String').SetString( 'C_USERS');
7126:   SIRegister_TStRegIni(CL);
7127: end;
7128:
7129: (*-----*)
7130: procedure SIRegister_StMerge(CL: TPSPascalCompiler);
7131: begin
7132:   StDefaultTagStart', 'String').SetString( '<');
7133:   'StDefaultTagEnd', 'String').SetString( '>');
7134:   'StDefaultEscapeChar', 'String').SetString( '\');
7135:   TStGotMergeTagEvent', 'Procedure ( Sender : TObject; Tag : AnsiS'
7136:   + 'tring; var Value : AnsiString; var Discard : Boolean)';
7137:   SIRegister_TStTextMerge(CL);
7138: end;
7139:
7140: (*-----*)
7141: procedure SIRegister_StMime(CL: TPSPascalCompiler);
7142: begin
7143:   'MaxMimeLine', 'LongInt').SetInt( 78);
7144:   TStConvertState', '( csStarted, csProgressing, csFinished )';
7145:   TStProgressEvent', 'Procedure ( Sender : TObject; Status : TStCo'
7146:   + 'nvertState; PercentDone : Byte)';
7147:   TStSaveAsEvent', 'Procedure ( Sender : TObject; var FileName : string)';
7148:   CL.AddClassN(CL.FindClass('TObject'), 'TStMimeConverter');
7149:   SIRegister_TStConvertStream(CL);
7150:   SIRegister_TStRawStream(CL);
7151:   SIRegister_TStQuotedStream(CL);
7152:   SIRegister_TStUStream(CL);
7153:   SIRegister_TStBase64Stream(CL);
7154:   //TStConverterClass', 'class of TStConvertStream');
7155:   SIRegister_TStAttachment(CL);
7156:   SIRegister_TStMimeConverter(CL);
7157: end;
7158:
7159: (*-----*)
7160: procedure SIRegister_JvCommStatus(CL: TPSPascalCompiler);
7161: begin
7162:   TJvCommPort', 'Integer');
7163:   SIRegister_TJvCommWatcher(CL);
7164:   SIRegister_TJvCommStatus(CL);
7165: end;
7166: (*-----*)
7167: procedure SIRegister_JvExControls(CL: TPSPascalCompiler);
7168: begin
7169:   TDlgCode', '( dcWantAllKeys, dcWantArrows, dcWantChars, dcButton'
7170:   + ', dcHasSetSel, dcWantTab, dcNative )';
7171:   TDlgCodes', 'set of TDlgCode';
7172:   //dcWantMessage'', '').SetString( dcWantAllKeys);
7173:   SIRegister_IJvExControl(CL);
7174:   SIRegister_IJvDenySubClassing(CL);
7175:   SIRegister_TStructPtrMessage(CL);
7176:   CL.AddDelphiFunction('Procedure SetDotNetFrameColors( FocusedColor, UnfocusedColor : TColor)');
7177:   Procedure DrawDotNetControl( Control : TWinControl; AColor : TColor; InControl : Boolean);
7178:   Procedure DrawDotNetControl( DC : HDC; R : TRect; AColor : TColor; UseFocusedColor : Boolean);
7179:   Procedure HandleDotNetHighlighting( Control:TWinControl;const Msg : TMessage; MouseOver : Boolean; Color :
   TColor);
7180:   Function CreateWMMMessage( Msg : Integer; WParam : Integer; LParam : Longint ) : TMessage;
7181:   Function CreateWMMMessage( Msg : Integer; WParam : Integer; LParam : TControl ) : TMessage;

```

```

7182: Function SmallPointToLong( const Pt : TSmallPoint) : Longint'';
7183: Function ShiftStateToKeyData( Shift : TShiftState) : Longint'';
7184: Function GetFocusedControl( AControl : TControl) : TWinControl'';
7185: Function DlgcToDlgsCodes( Value : Longint) : TDlgCodes'';
7186: Function DlgCodesToDlgs( Value : TDlgCodes) : Longint'';
7187: //CL.AddDelphiFunction('Procedure GetHintColor( var HintInfo : THintInfo; AControl : TControl; HintColor
: TColor)'';
7188: Function DispatchIsDesignMsg( Control : TControl; var Msg : TMessage) : Boolean'';
7189: SIRegister_TJvExControl(CL);
7190: SIRegister_TJvExWinControl(CL);
7191: SIRegister_TJvExCustomControl(CL);
7192: SIRegister_TJvExGraphicControl(CL);
7193: SIRegister_TJvExHintWindow(CL);
7194: SIRegister_TJvExPubGraphicControl(CL);
7195: end;
7196:
7197:
7198: (*-----*)
7199: procedure SIRegister_JvSHFileOperation(CL: TPSPascalCompiler);
7200: begin
7201: TJvShFileMappingEvent', 'Procedure ( Sender : TObject; const Old'
7202: + 'FileName, NewFileName : string)'';
7203: TJvSHFileOpType', '( foCopy, foDelete, foMove, foRename )'';
7204: TJvSHFileOption', '( fofAllowUndo, fofConfirmMouse, fofFilesOnly'
7205: + ', fofMultiDestFiles, fofNoConfirmation, fofNoConfirmMkDir, fofRenameOnColl'
7206: + 'ision, fofSilent, fofSimpleProgress, fofWantMappingHandle, fofNoErrorUI, f'
7207: + 'ofNoCopySecurityAttributes, fofNoRecursion, fofNoConnectedElements, fofNoRecurseParse,
fofWantNukeWarning )'';
7208: TJvSHFileOptions', 'set of TJvSHFileOption'';
7209: SIRegister_TJvSHFileOperation(CL);
7210: end;
7211:
7212: (*-----*)
7213: procedure SIRegister_SortGrid(CL: TPSPascalCompiler);
7214: begin
7215: TPrintMode', '( pmPrint, pmPreview, pmPageCount )'';
7216: TMyPrintRange', '( prAll, prSelected )'';
7217: TSortStyle', '( ssAutomatic, ssNormal, ssNumeric, ssNumericExten'
7218: + 'ded, ssDateTime, ssTime, ssCustom )'';
7219: TSortDirection', '( sdAscending, sdDescending )'';
7220: TSetChecked', 'Procedure ( Sender : TObject; ACol, ARow : integer; State : Boolean)'';
7221: TGetCombobox', 'Procedure ( Sender : TObject; ACol, ARow : integ'
7222: + 'er; var Strs : TStringList; var Width, Height : integer; var Sorted : Boolean)'';
7223: TSetCombobox', 'Procedure ( Sender : TObject; ACol, ARow : integer; Str : String)'';
7224: TSetEllipsis', 'Procedure ( Sender : TObject; ACol, ARow : integer)'';
7225: SIRegister_TSortOptions(CL);
7226: SIRegister_TPrintOptions(CL);
7227: TSortedListEntry', 'record Str: String; RowNum : integer; SortOption: TSortOptions; end'';
7228: //PSortedListEntry', 'TSortedListEntry // will not work'';
7229: SIRegister_TSortedList(CL);
7230: TCellBevelStyle', '( cbNone, cbRaised, cbLowered )'';
7231: TCellBevel', 'record Style: TCellBevelStyle; UpperLeftColor: TColor; LowerRightColor : TColor; end'';
7232: TVertAlignment', '( taTopJustify, taBottomJustify, taMiddle )'';
7233: TFormatOptions', 'record Brush: TBrush; Font: TFont; Alignment'
7234: + 'Horz : TAlignment; AlignmentVert : TVertAlignment; Bevel : TCellBevel; HideText : Boolean; end'';
7235: SIRegister_TFontSetting(CL);
7236: SIRegister_TFontList(CL);
7237: TFormatDrawCellEvent', 'Procedure ( Sender : TObject; Col, Row : '
7238: + 'integer; State : TGridDrawState; var FormatOptions : TFormatOptions; var '
7239: + 'CheckBox, Combobox, Ellipsis : Boolean)'';
7240: TSetFilterEvent', 'Procedure ( ARows : TStringList; var Accept : Boolean)'';
7241: TSearchEvent', 'Procedure ( ARows : TStringList; var Accept : Boolean)'';
7242: TUpdateGridEvent', 'Procedure ( Sender : TObject; ARow : integer)'';
7243: TSizeChangeEvent', 'Procedure ( Sender : TObject; OldColCount, OldRowCount : integer)'';
7244: TBeginSortEvent', 'Procedure ( Sender : TObject; var Col : integer)'';
7245: TEndSortEvent', 'Procedure ( Sender : TObject; Col : integer)'';
7246: TGetSortStyleEvent', 'Procedure ( Sender : TObject; Col : integer; var SortStyle : TSortStyle)'';
7247: TCellValidateEvent', 'Procedure ( Sender : TObject; ACol, ARow : '
7248: + 'integer; const OldValue : string; var NewValue : String; var Valid : Boolean)'';
7249: SIRegister_TSortGrid(CL);
7250: Function ExtendedCompare( const Str1, Str2 : String) : Integer'';
7251: Function NormalCompare( const Str1, Str2 : String) : Integer'';
7252: Function DateTimeCompare( const Str1, Str2 : String) : Integer'';
7253: Function NumericCompare( const Str1, Str2 : String) : Integer'';
7254: Function TimeCompare( const Str1, Str2 : String) : Integer'';
7255: //Function Compare( Item1, Item2 : Pointer) : Integer'';
7256: end;
7257:
7258: (*-----*)
7259: procedure SIRegister_IBDatabase(CL: TPSPascalCompiler);
7260: begin
7261: CL.AddClassN(CL.FindClass('TObject'),'TIBDATABASE');
7262: CL.AddClassN(CL.FindClass('TObject'),'TIBTRANSACTION');
7263: CL.AddClassN(CL.FindClass('TObject'),'TIBBASE');
7264: TIBDATABASELOGINEVENT', 'Procedure ( DATABASE : TIBDATABASE; LOG'
7265: + 'INPARAMS : TSTRINGS)'';
7266: SIRegister_TIBSCHEMA(CL);
7267: TIBFILENAME', 'STRING'';
7268: SIRegister_TIBDATABASE(CL);

```

```

7269: TTRANSACTIONACTION', '( TAROLLBACK, TACOMMIT, TAROLLBACKRETAINING, TACOMMITRETAINING )');
7270: TAUTOSTOPACTION', '( SANONE, SAROLLBACK, SACOMMIT, SAROLLBACKRET'
7271: + 'AINING, SACOMMITRETAINING )');
7272: SIRegister_TIBTRANSACTION(CL);
7273: SIRegister_TIBBASE(CL);
7274: procedure GenerateDPB(sl: TStrings; var DPB: string; var DPBLength: Short););
7275: //procedure GenerateDPB(sl: TStrings; var DPB: string; var DPBLength: Short);
7276: procedure GenerateTPB(sl: TStrings; var TPB: string; var TPBLength: Short););
7277: end;
7278:
7279: procedure SIRegister_yuvconverts(CL: TPSPascalCompiler);
7280: begin
7281:   TVideoCodec, '( vcUnknown, vcRGB, vcYUY2, vcUYVY, vcBTYUV, vcYV'
7282:   + 'U9, vcYUV12, vcY8, vcY211 )';
7283:   'BI_YUY2', 'LongWord').SetUInt( $32595559);
7284:   'BI_UYVY', 'LongWord').SetUInt( $59565955);
7285:   'BI_BTUV', 'LongWord').SetUInt( $50313459);
7286:   'BI_YVU9', 'LongWord').SetUInt( $39555659);
7287:   'BI_YUV12', 'LongWord').SetUInt( $30323449);
7288:   'BI_Y8', 'LongWord').SetUInt( $20203859);
7289:   'BI_Y211', 'LongWord').SetUInt( $31313259);
7290:   Function BICompressionToVideoCodec( Value : DWord ) : TVideoCodec;
7291:   Function ConvertCodecToRGB( Codec : TVideoCodec; Src, Dst : Pointer; AWidth, AHeight : Integer ) :
   Boolean);
7292: end;
7293:
7294: (*-----*)
7295: procedure SIRegister_AviCap(CL: TPSPascalCompiler);
7296: begin
7297:   'WM_USER', 'LongWord').SetUInt( $0400);
7298:   'WM_CAP_START', 'LongWord').SetUInt($0400);
7299:   'WM_CAP_END', 'LongWord').SetUInt($0400+85);
7300:   //WM_CAP_START+ 85
7301:   // WM_CAP_SET_CALLBACK_CAPCONTROL = (WM_CAP_START+ 85);
7302:   TCapDriverCaps, 'record wDeviceIndex : WORD; fHasOverlay : BOOL'
7303:   + 'fHasDlgVideoSource : BOOL; fHasDlgVideoFormat : BOOL; fHasDlgVideoDispla'
7304:   + 'y : BOOL; fCaptureInitialized : BOOL; fDriverSuppliesPalettes : BOOL; hVid'
7305:   + 'eoIn : THANDLE; hVideoOut : THANDLE; hVideoExtIn : THANDLE; hVideoExtOut : THANDLE; end';
7306:   //PCapStatus, '^TCapStatus // will not work';
7307:   TCapStatus, 'record uiImageWidth : UINT; uiImageHeight : UINT; '
7308:   + 'fLiveWindow : BOOL; fOverlayWindow : BOOL; fScale : BOOL; ptScroll : TPOINT'
7309:   + 'T; fUsingDefaultPalette : BOOL; fAudioHardware : BOOL; fCapFileExists : BO'
7310:   + 'OL; dwCurrentVideoFrame : DWORD; dwCurrentVideoFramesDropped : DWORD; dwCu'
7311:   + 'rrentWaveSamples : DWORD; dwCurrentTimeElapsedMS : DWORD; hPalCurrent : HP'
7312:   + 'LETTE; fCapturingNow : BOOL; dwReturn : DWORD; wNumVideoAllocated : WORD; '
7313:   + 'wNumAudioAllocated : WORD; end';
7314:   //PCaptureParms, '^TCaptureParms // will not work';
7315:   TCaptureParms, 'record dwRequestMicroSecPerFrame : DWORD; fMake'
7316:   + 'UserHitOKToCapture : BOOL; wPercentDropForError : WORD; fYield : BOOL; dwI'
7317:   + 'ndexSize : DWORD; wChunkGranularity : WORD; fUsingDOSMemory : BOOL; wNumVi'
7318:   + 'deoRequested : WORD; fCaptureAudio : BOOL; wNumAudioRequested : WORD; vKey'
7319:   + 'Abort : WORD; fAbortLeftMouse : BOOL; fAbortRightMouse : BOOL; fLimitEnabl'
7320:   + 'ed : BOOL; wTimeLimit : WORD; fMCIControl : BOOL; fStepMCIDevice : BOOL; d'
7321:   + 'wMCIStartTime : DWORD; dwMCIStopTime : DWORD; fStepCaptureAt2x : BOOL; wSt'
7322:   + 'epCaptureAverageFrames : WORD; dwAudioBufferSize : DWORD; fDisableWriteCac'
7323:   + 'he : BOOL; AVStreamMaster : WORD; end';
7324:   // PCapInfoChunk, '^TCapInfoChunk // will not work';
7325:   //TCapInfoChunk, 'record fccInfoID : FOURCC; lpData : LongInt; cbData : LongInt; end';
7326:   'CONTROLCALLBACK_PREROLL', 'LongInt').SetInt( 1);
7327:   'CONTROLCALLBACK_CAPTURING', 'LongInt').SetInt( 2);
7328:   Function capCreateCaptureWindow( lpszWindowName : PChar; dwStyle : DWORD; x, y : Integer; nWidth, nHeight
   : Integer; hwndParent : THandle; nID : Integer ) : THandle;
7329:   Function capGetDriverDescription( wDriverIndex : DWORD; lpszName : PChar; cbName : Integer; lpszVer :
   PChar; cbVer : Integer ) : Boolean);
7330:   'IDS_CAP_BEGIN', 'LongInt').SetInt( 300);
7331:   'IDS_CAP_END', 'LongInt').SetInt( 301);
7332:   'IDS_CAP_INFO', 'LongInt').SetInt( 401);
7333:   'IDS_CAP_OUTOFMEM', 'LongInt').SetInt( 402);
7334:   'IDS_CAP_FILEEXISTS', 'LongInt').SetInt( 403);
7335:   'IDS_CAP_ERRORPALOPEN', 'LongInt').SetInt( 404);
7336:   'IDS_CAP_ERRORPALSAVE', 'LongInt').SetInt( 405);
7337:   'IDS_CAP_ERRORDISSAVE', 'LongInt').SetInt( 406);
7338:   'IDS_CAP_DEFAVIEXT', 'LongInt').SetInt( 407);
7339:   'IDS_CAP_DEFFPALEXT', 'LongInt').SetInt( 408);
7340:   'IDS_CAP_CANTOPEN', 'LongInt').SetInt( 409);
7341:   'IDS_CAP_SEQ_MSGSTART', 'LongInt').SetInt( 410);
7342:   'IDS_CAP_SEQ_MSGSTOP', 'LongInt').SetInt( 411);
7343:   'IDS_CAP_VIDEDITERR', 'LongInt').SetInt( 412);
7344:   'IDS_CAP_READONLYFILE', 'LongInt').SetInt( 413);
7345:   'IDS_CAP_WRITEERROR', 'LongInt').SetInt( 414);
7346:   'IDS_CAP_NODISKSPACE', 'LongInt').SetInt( 415);
7347:   'IDS_CAP_SETFILESIZE', 'LongInt').SetInt( 416);
7348:   'IDS_CAP_SAVEASPERCENT', 'LongInt').SetInt( 417);
7349:   'IDS_CAP_DRIVER_ERROR', 'LongInt').SetInt( 418);
7350:   'IDS_CAP_WAVE_OPEN_ERROR', 'LongInt').SetInt( 419);
7351:   'IDS_CAP_WAVE_ALLOC_ERROR', 'LongInt').SetInt( 420);
7352:   'IDS_CAP_WAVE_PREPARE_ERROR', 'LongInt').SetInt( 421);
7353:   'IDS_CAP_WAVE_ADD_ERROR', 'LongInt').SetInt( 422);
7354:   'IDS_CAP_WAVE_SIZE_ERROR', 'LongInt').SetInt( 423);

```

```

7355: 'IDS_CAP_VIDEO_OPEN_ERROR', 'LongInt').SetInt( 424);
7356: 'IDS_CAP_VIDEO_ALLOC_ERROR', 'LongInt').SetInt( 425);
7357: 'IDS_CAP_VIDEO_PREPARE_ERROR', 'LongInt').SetInt( 426);
7358: 'IDS_CAP_VIDEO_ADD_ERROR', 'LongInt').SetInt( 427);
7359: 'IDS_CAP_VIDEO_SIZE_ERROR', 'LongInt').SetInt( 428);
7360: 'IDS_CAP_FILE_OPEN_ERROR', 'LongInt').SetInt( 429);
7361: 'IDS_CAP_FILE_WRITE_ERROR', 'LongInt').SetInt( 430);
7362: 'IDS_CAP_RECORDING_ERROR', 'LongInt').SetInt( 431);
7363: 'IDS_CAP_RECORDING_ERROR2', 'LongInt').SetInt( 432);
7364: 'IDS_CAP_AVI_INIT_ERROR', 'LongInt').SetInt( 433);
7365: 'IDS_CAP_NO_FRAME_CAP_ERROR', 'LongInt').SetInt( 434);
7366: 'IDS_CAP_NO_PALETTE_WARN', 'LongInt').SetInt( 435);
7367: 'IDS_CAP_MCI_CONTROL_ERROR', 'LongInt').SetInt( 436);
7368: 'IDS_CAP_MCI_CANT_STEP_ERROR', 'LongInt').SetInt( 437);
7369: 'IDS_CAP_NO_AUDIO_CAP_ERROR', 'LongInt').SetInt( 438);
7370: 'IDS_CAP_AVI_DRAWDIB_ERROR', 'LongInt').SetInt( 439);
7371: 'IDS_CAP_COMPRESSOR_ERROR', 'LongInt').SetInt( 440);
7372: 'IDS_CAP_AUDIO_DROP_ERROR', 'LongInt').SetInt( 441);
7373: 'IDS_CAP_STAT_LIVE_MODE', 'LongInt').SetInt( 500);
7374: 'IDS_CAP_STAT_OVERLAY_MODE', 'LongInt').SetInt( 501);
7375: 'IDS_CAP_STAT_CAP_INIT', 'LongInt').SetInt( 502);
7376: 'IDS_CAP_STAT_CAP_FINI', 'LongInt').SetInt( 503);
7377: 'IDS_CAP_STAT_PALETTE_BUILD', 'LongInt').SetInt( 504);
7378: 'IDS_CAP_STAT_OPTPAL_BUILD', 'LongInt').SetInt( 505);
7379: 'IDS_CAP_STAT_I_FRAMES', 'LongInt').SetInt( 506);
7380: 'IDS_CAP_STAT_L_FRAMES', 'LongInt').SetInt( 507);
7381: 'IDS_CAP_STAT_CAP_L_FRAMES', 'LongInt').SetInt( 508);
7382: 'IDS_CAP_STAT_CAP_AUDIO', 'LongInt').SetInt( 509);
7383: 'IDS_CAP_STAT_VIDEOCURRENT', 'LongInt').SetInt( 510);
7384: 'IDS_CAP_STAT_VIDEOAUDIO', 'LongInt').SetInt( 511);
7385: 'IDS_CAP_STAT_VIDEOONLY', 'LongInt').SetInt( 512);
7386: 'IDS_CAP_STAT_FRAMESDROPPED', 'LongInt').SetInt( 513);
7387: 'AVICAP32', 'String').SetString( 'AVICAP32.dll');
7388: end;
7389:
7390: procedure SIRegister_ALWindows(CL: TPSPascalCompiler);
7391: begin
7392:   _ALMEMORYSTATUSEX', 'record dwLength : DWORD; dwMemoryLoad : DWO'
7393:   + 'RD; ullTotalPhys : Int64; ullAvailPhys : Int64; ullTotalPageFile : Int64; '
7394:   + 'ullAvailPageFile : Int64; ullTotalVirtual : Int64; ullAvailVirtual : Int64'
7395:   + '; ullAvailExtendedVirtual : Int64; end';
7396:   TALMemoryStatusEx', '_ALMEMORYSTATUSEX');
7397:   Function ALGlobalMemoryStatusEx( var lpBuffer : TALMEMORYSTATUSEX ) : BOOL';
7398:   Function ALInterlockedExchange64( var Target : LONGLONG; Value : LONGLONG ) : LONGLONG';
7399:   'INVALID_SET_FILE_POINTER', 'LongInt').SetInt( DWORD ( - 1 ));
7400:   'QUOTA_LIMITS_HARDWS_MIN_DISABLE', 'LongWord').SetUInt( $2);
7401:   'QUOTA_LIMITS_HARDWS_MIN_ENABLE', 'LongWord').SetUInt( $1);
7402:   'QUOTA_LIMITS_HARDWS_MAX_DISABLE', 'LongWord').SetUInt( $8);
7403:   'QUOTA_LIMITS_HARDWS_MAX_ENABLE', 'LongWord').SetUInt( $4);
7404: end;
7405:
7406: procedure SIRegister_ALHttpCommon(CL: TPSPascalCompiler);
7407: begin
7408:   TALHTTPPropertyChangeEvent', 'Procedure ( sender : TObject; cons'
7409:   + 't PropertyIndex : Integer)';
7410:   TALHTTPProtocolVersion', '( HTTPPv_1_0, HTTPPv_1_1 )';
7411:   TALHTTPMethod', '( HTTPM_Get, HTTPM_Post, HTTPM_Head, HTTPM_Trace, HTTPM_Put, HTTPM_Delete )';
7412:   TInternetScheme', 'integer';
7413:   TALIPv6Binary', 'array[1..16] of Char';
7414:   // TALIPv6Binary = array[1..16] of ansiChar;
7415:   // TInternetScheme = integer;
7416:   SIRegister_TALHTTPRequestHeader(CL);
7417:   SIRegister_TALHTTPCookie(CL);
7418:   SIRegister_TALHTTPCookieCollection(CL);
7419:   SIRegister_TALHTTPResponseHeader(CL);
7420:   Function ALHTTPDecode( const AStr : AnsiString ) : AnsiString';
7421:   Procedure ALHTTPEncodeParamNameValues( ParamValues : TALStrings)';
7422: end;
7423:
7424: procedure SIRegister_ALHttpClient(CL: TPSPascalCompiler);
7425: begin
7426:   SIRegister_EALHttpClientException(CL);
7427:   SIRegister_TALHttpClientProxyParams(CL);
7428:   TALHttpClientRedirectEvent', 'Procedure ( sender : TObject; const NewURL : AnsiString)';
7429:   TALHttpClientUploadProgressEvent', 'Procedure (sender : TObject; Sent : Integer; Total : Integer)';
7430:   TALHttpClientDownloadProgressEvent', 'Procedure (sender : TObject; Read : Integer; Total : Integer)';
7431:   SIRegister_TALHttpClient(CL);
7432:   'CALHTTIClient_MsgInvalidURL', 'String').SetString( 'Invalid url' + '%s' - only supports 'http' and
   'https' schemes);
7433:   'CALHTTIClient_MsgInvalidHTTPRequest', 'String').SetString( 'Invalid HTTP Request: Length is 0');
7434:   'CALHTTIClient_MsgEmptyURL', 'String').SetString( 'Empty URL');
7435: end;
7436:
7437: procedure SIRegister_ALInternetMessageCommon(CL: TPSPascalCompiler);
7438: begin
7439:   SIRegister_TALEMailHeader(CL);
7440:   SIRegister_TALNewsArticleHeader(CL);
7441: end;
7442:

```



```

7443: procedure SIRegister_ALWininetHttpClient(CL: TPSPascalCompiler);
7444: begin
7445:   DWORD_PTR, 'DWORD');
7446:   'INTERNET_STATUS_COOKIE_SENT', 'LongInt').SetInt( 320);
7447:   'INTERNET_STATUS_COOKIE_RECEIVED', 'LongInt').SetInt( 321);
7448:   'INTERNET_STATUS_PRIVACY_IMPACTED', 'LongInt').SetInt( 324);
7449:   'INTERNET_STATUS_P3P_HEADER', 'LongInt').SetInt( 325);
7450:   'INTERNET_STATUS_P3P_POLICYREF', 'LongInt').SetInt( 326);
7451:   'INTERNET_STATUS_COOKIE_HISTORY', 'LongInt').SetInt( 327);
7452:   TALWininetHttpInternetOpenAccessType, '( wHttpAt_Direct, wHttpA'
7453:   + 't_Preconfig, wHttpAt_Preconfig_with_no_autoproxy, wHttpAt_Proxy )');
7454:   TALWininetHttpClientInternetOption, '( wHttpIo_Async, wHttpIo_F'
7455:   + 'rom_Cache, wHttpIo_Offline, wHttpIo_Cache_if_net_fail, wHttpIo_Hyperlink, '
7456:   + 'wHttpIo_Ignore_cert_cn_invalid, wHttpIo_Ignore_cert_date_invalid, wHttpIo_'
7457:   + 'Ignore_redirect_to_http, wHttpIo_Ignore_redirect_to_https, wHttpIo_Keep_co'
7458:   + 'nnection, wHttpIo_Need_file, wHttpIo_No_auth, wHttpIo_No_auto_redirect, wH'
7459:   + 'tIo_No_cache_write, wHttpIo_No_cookies, wHttpIo_No_ui, wHttpIo_Pragma_no'
7460:   + 'cache, wHttpIo_Reload, wHttpIo_Resynchronize, wHttpIo_Secure )');
7461:   TALWininetHttpClientInternetOptionSet, 'set of TALWininetHttpClientInternetOption');
7462:   SIRegister_TALWininetHTTPClient(CL);
7463: end;
7464:
7465: procedure SIRegister_ALWininetFTPClient(CL: TPSPascalCompiler);
7466: begin
7467:   DWORD_PTR, 'DWORD');
7468:   TALWininetFTPInternetOpenAccessType, '( wFtpAt_Direct, wFtpAt_P'
7469:   + 'reconfig, wFtpAt_Preconfig_with_no_autoproxy, wFtpAt_Proxy )');
7470:   TALWininetFTPClientInternetOption, '( wFtpIo_Async, wFtpIo_From'
7471:   + '_Cache, wFtpIo_Offline, wFtpIo_Passive, wFtpIo_Hyperlink, wFtpIo_Need_file'
7472:   + ', wFtpIo_No_cache_write, wFtpIo_Reload, wFtpIo_Resynchronize )');
7473:   TALWininetFtpTransferType, '( wFtpTt_ASCII, wFtpTt_BINARY )');
7474:   TALWininetFTPClientInternetOptionSet, 'set of TALWininetFTPClientInternetOption');
7475:   //TALWininetFTPClientStatusChangeEvent, 'Procedure ( sender : Tob'
7476:   // + 'ject; InternetStatus : DWORD; StatusInformation : Pointer; StatusInformationLength : DWORD)';
7477:   SIRegister_TALWininetFTPClient(CL);
7478: end;
7479:
7480: procedure SIRegister_MonForm(CL: TPSPascalCompiler);
7481: begin
7482:   'WM_SETTRACEDATA', 'LongInt').SetInt( WM_USER + 1);
7483:   WM_UPDATESTATUS, 'LongInt').SetInt( WM_USER + 2);
7484:   'WM_UPDATEMENU', 'LongInt').SetInt( WM_USER + 3);
7485:   TWMTraceData, 'record Msg : Cardinal; X : Smallint; Y : Smallin'
7486:   + 't; Flag : TClientFlag; Result : Longint; end');
7487:   TLabelRec, 'record XLabel : TLabel; YLabel : TLabel; end');
7488:   SIRegister_TMonitorForm(CL);
7489: end;
7490:
7491: procedure SIRegister_IPCThrd(CL: TPSPascalCompiler);
7492: begin
7493:   SIRegister_THandledObject(CL);
7494:   SIRegister_TEvent(CL);
7495:   SIRegister_TMutex(CL);
7496:   SIRegister_TSharedMem(CL);
7497:   'TRACE_BUF_SIZE', 'LongInt').SetInt( 200 * 1024);
7498:   'TRACE_BUFFER', 'String').SetString( 'TRACE_BUFFER');
7499:   'TRACE_Mutex', 'String').SetString( 'TRACE_Mutex');
7500:   //PTraceEntry, '^TTraceEntry // will not work';
7501:   SIRegister_TIPCTracer(CL);
7502:   'MAX_CLIENTS', 'LongInt').SetInt( 6);
7503:   'IPCTIMEOUT', 'LongInt').SetInt( 2000);
7504:   'IPCBUFFER_NAME', 'String').SetString( 'BUFFER_NAME');
7505:   'BUFFER_Mutex_NAME', 'String').SetString( 'BUFFER_Mutex');
7506:   'MONITOR_EVENT_NAME', 'String').SetString( 'MONITOR_EVENT');
7507:   'CLIENT_EVENT_NAME', 'String').SetString( 'CLIENT_EVENT');
7508:   'CONNECT_EVENT_NAME', 'String').SetString( 'CONNECT_EVENT');
7509:   'CLIENT_DIR_NAME', 'String').SetString( 'CLIENT_DIRECTORY');
7510:   'CLIENT_DIR_Mutex', 'String').SetString( 'DIRECTORY_Mutex');
7511:   CL.AddClassN(CL.FindClass('TOBJECT'), 'EMonitorActive');
7512:   CL.AddClassN(CL.FindClass('TOBJECT'), 'TIPCThread');
7513:   TEventKind, '( evMonitorAttach, evMonitorDetach, evMonitorSigna'
7514:   + 'l, evMonitorExit, evClientStart, evClientStop, evClientAttach, evClientDet'
7515:   + 'ach, evClientSwitch, evClientSignal, evClientExit )');
7516:   TClientFlag, '( cfError, cfMouseMove, cfMouseDown, cfResize, cfAttach )');
7517:   TClientFlags, 'set of TClientFlag';
7518:   //PEventData, '^TEventData // will not work';
7519:   TEventData, 'record X : Smallint; Y : Smallint; Flag : TClientF'
7520:   + 'lag; Flags : TClientFlags; end';
7521:   TConnectEvent, 'Procedure ( Sender : TIPCThread; Connecting : Boolean)';
7522:   TDirUpdateEvent, 'Procedure ( Sender : TIPCThread)';
7523:   TIPCTNotifyEvent, 'Procedure ( Sender : TIPCThread; Data : TEventData)';
7524:   //PIPCEventInfo, '^TIPCEventInfo // will not work';
7525:   TIPCEventInfo, 'record FID : Integer; FKind : TEventKind; FData : TEventData; end';
7526:   SIRegister_TIPCEvent(CL);
7527:   //PClientDirRecords, '^TClientDirRecords // will not work';
7528:   SIRegister_TClientDirectory(CL);
7529:   TIPCTState, '( stInactive, stDisconnected, stConnected )';
7530:   SIRegister_TIPCThread(CL);
7531:   SIRegister_TIPCMonitor(CL);

```

```

7532:   SIRegister_TIPCCClient(CL);
7533:   CL.AddDelphiFunction('Function IsMonitorRunning( var Hndl : THandle) : Boolean');
7534: end;
7535:
7536: (*-----*)
7537: procedure SIRegister_MonForm(CL: TPSPascalCompiler);
7538: begin
7539:   'WM_SETTRACEDATA','LongInt').SetInt( WM_USER + 1);
7540:   'WM_UPDATESTATUS','LongInt').SetInt( WM_USER + 2);
7541:   'WM_UPDATEMENU','LongInt').SetInt( WM_USER + 3);
7542:   TWMTraceData, 'record Msg : Cardinal; X : Smallint; Y : Smallint
7543:   + 't; Flag : TClientFlag; Result : Longint; end');
7544:   TLabelRec, 'record XLabel : TLabel; YLabel : TLabel; end';
7545:   SIRegister_TMonitorForm(CL);
7546: end;
7547:
7548:
7549: procedure SIRegister_ALWinHttpWrapper(CL: TPSPascalCompiler);
7550: begin
7551:   //DWORD_PTR, 'DWORD');
7552:   HINTERNET, '___Pointer');
7553:   //PHINTERNET, '^HINTERNET // will not work');
7554:   //LPHINTERNET, 'PHINTERNET');
7555:   //INTERNET_PORT, 'Word');
7556:   //TInternetScheme, '^TInternetScheme // will not work');
7557:   TInternetScheme, 'Integer');
7558:   //PURLComponents, '^URL_COMPONENTS // will not work');
7559:   {URL_COMPONENTS, 'record dwStructSize : DWORD; lpszScheme : LPWS'
7560:   + 'TR; dwSchemeLength : DWORD; nScheme : TInternetScheme; lpszHostName : LPWS'
7561:   + 'TR; dwHostNameLength : DWORD; nPort : INTERNET_PORT; pad : WORD; lpszUserN'
7562:   + 'ame : LPWSTR; dwUserNameLength : DWORD; lpszPassword : LPWSTR; dwPasswordL'
7563:   + 'ength : DWORD; lpszUrlPath : LPWSTR; dwUrlPathLength : DWORD; lpszExtraInf'
7564:   + 'o : LPWSTR; dwExtraInfoLength : DWORD; end');}
7565:   //TURLComponents, 'URL_COMPONENTS');
7566:   //LPURL_COMPONENTS, 'PURLComponents');
7567:   //PWINHTTP_PROXY_INFO, '^WINHTTP_PROXY_INFO // will not work');
7568:   //WINHTTP_PROXY_INFO, 'record dwAccessType : DWORD; lpszProxy : L'
7569:   // + 'PWSTR; lpszProxyBypass : LPWSTR; end');
7570:   //TWINHTTP_PROXY_INFO, 'WINHTTP_PROXY_INFO');
7571:   // LPWINHTTP_PROXY_INFO, 'PWINHTTP_PROXY_INFO');
7572:   //PWINHTTP_CURRENT_USER_IE_PROXY_CONFIG, '^WINHTTP_CURRENT_USER_I'
7573:   // + 'E_PROXY_CONFIG // will not work');
7574:   //WINHTTP_CURRENT_USER_IE_PROXY_CONFIG, 'record fAutoDetect : BOO'
7575:   // + 'L; lpszAutoConfigUrl : LPWSTR; lpszProxy : LPWSTR; lpszProxyBypass : LPWSTR; end');
7576:   // TWINHTTP_CURRENT_USER_IE_PROXY_CONFIG, 'WINHTTP_CURRENT_USER_IE_PROXY_CONFIG');
7577:   // LPWINHTTP_CURRENT_USER_IE_PROXY_CONFIG, 'PWINHTTP_CURRENT_USER_IE_PROXY_CONFIG');
7578:   'INTERNET_DEFAULT_PORT','LongInt').SetInt( 0);
7579:   'INTERNET_DEFAULT_HTTP_PORT','LongInt').SetInt( 80);
7580:   'INTERNET_DEFAULT_HTTPS_PORT','LongInt').SetInt( 443);
7581:   'WINHTTP_FLAG_ASYNC','LongWord').SetUInt( $10000000);
7582:   'WINHTTP_FLAG_SECURE','LongWord').SetUInt( $00800000);
7583:   'WINHTTP_FLAG_ESCAPE_PERCENT','LongWord').SetUInt( $00000004);
7584:   'WINHTTP_FLAG_NIL_CODEPAGE','LongWord').SetUInt( $00000008);
7585:   'WINHTTP_FLAG_BYPASS_PROXY_CACHE','LongWord').SetUInt( $00000100);
7586:   'WINHTTP_FLAG_REFRESH','LongWord').SetUInt( $00000100);
7587:   'WINHTTP_FLAG_ESCAPE_DISABLE','LongWord').SetUInt( $00000040);
7588:   'WINHTTP_FLAG_ESCAPE_DISABLE_QUERY','LongWord').SetUInt( $00000080);
7589:   'SECURITY_FLAG_IGNORE_UNKNOWN_CA','LongWord').SetUInt( $00000100);
7590:   'SECURITY_FLAG_IGNORE_CERT_DATE_INVALID','LongWord').SetUInt( $00002000);
7591:   'SECURITY_FLAG_IGNORE_CERT_CN_INVALID','LongWord').SetUInt( $00001000);
7592:   'SECURITY_FLAG_IGNORE_CERT_WRONG_USAGE','LongWord').SetUInt( $00000200);
7593:   'INTERNET_SCHEME_HTTP','LongInt').SetInt( ( 1 ));
7594:   'INTERNET_SCHEME_HTTPS','LongInt').SetInt( ( 2 ));
7595:   'WINHTTP_AUTOPROXY_AUTO_DETECT','LongWord').SetUInt( $00000001);
7596:   'WINHTTP_AUTOPROXY_CONFIG_URL','LongWord').SetUInt( $00000002);
7597:   'WINHTTP_AUTOPROXY_RUN_INPROCESS','LongWord').SetUInt( $00010000);
7598:   'WINHTTP_AUTOPROXY_RUN_OUTPROCESS_ONLY','LongWord').SetUInt( $00020000);
7599:   'WINHTTP_AUTO_DETECT_TYPE_DHCP','LongWord').SetUInt( $00000001);
7600:   'WINHTTP_AUTO_DETECT_TYPE_DNS_A','LongWord').SetUInt( $00000002);
7601:   'WINHTTP_TIME_FORMAT_BUF_SIZE','LongInt').SetInt( 62);
7602:   'ICU_NO_ENCODE','LongWord').SetUInt( $20000000);
7603:   'ICU_DECODE','LongWord').SetUInt( $10000000);
7604:   'ICU_NO_META','LongWord').SetUInt( $08000000);
7605:   'ICU_ENCODE_SPACES_ONLY','LongWord').SetUInt( $04000000);
7606:   'ICU_BROWSER_MODE','LongWord').SetUInt( $02000000);
7607:   'ICU_ENCODE_PERCENT','LongWord').SetUInt( $00001000);
7608:   'ICU_ESCAPE','LongWord').SetUInt( $80000000);
7609:   'WINHTTP_ACCESS_TYPE_DEFAULT_PROXY','LongInt').SetInt( 0);
7610:   'WINHTTP_ACCESS_TYPE_NO_PROXY','LongInt').SetInt( 1);
7611:   'WINHTTP_ACCESS_TYPE_NAMED_PROXY','LongInt').SetInt( 3);
7612:   'WINHTTP_NO_PROXY_NAME','LongInt').SetInt( 0);
7613:   'WINHTTP_NO_PROXY_BYPASS','LongInt').SetInt( 0);
7614:   'WINHTTP_OPTION_CALLBACK','LongInt').SetInt( 1);
7615:   'WINHTTP_FIRST_OPTION','LongInt').SetInt( 1);
7616:   'WINHTTP_OPTION_RESOLVE_TIMEOUT','LongInt').SetInt( 2);
7617:   'WINHTTP_OPTION_CONNECT_TIMEOUT','LongInt').SetInt( 3);
7618:   'WINHTTP_OPTION_CONNECT_RETRIES','LongInt').SetInt( 4);
7619:   'WINHTTP_OPTION_SEND_TIMEOUT','LongInt').SetInt( 5);
7620:   'WINHTTP_OPTION_RECEIVE_TIMEOUT','LongInt').SetInt( 6);

```

```

7621: 'WINHTTP_OPTION_RECEIVE_RESPONSE_TIMEOUT', 'LongInt').SetInt( 7);
7622: 'WINHTTP_OPTION_HANDLE_TYPE', 'LongInt').SetInt( 9);
7623: 'WINHTTP_OPTION_READ_BUFFER_SIZE', 'LongInt').SetInt( 12);
7624: 'WINHTTP_OPTION_WRITE_BUFFER_SIZE', 'LongInt').SetInt( 13);
7625: 'WINHTTP_OPTION_PARENT_HANDLE', 'LongInt').SetInt( 21);
7626: 'WINHTTP_OPTION_EXTENDED_ERROR', 'LongInt').SetInt( 24);
7627: 'WINHTTP_OPTION_SECURITY_FLAGS', 'LongInt').SetInt( 31);
7628: 'WINHTTP_OPTION_SECURITY_CERTIFICATE_STRUCT', 'LongInt').SetInt( 32);
7629: 'WINHTTP_OPTION_URL', 'LongInt').SetInt( 34);
7630: 'WINHTTP_OPTION_SECURITY_KEY_BITNESS', 'LongInt').SetInt( 36);
7631: 'WINHTTP_OPTION_PROXY', 'LongInt').SetInt( 38);
7632: 'WINHTTP_OPTION_USER_AGENT', 'LongInt').SetInt( 41);
7633: 'WINHTTP_OPTION_CONTEXT_VALUE', 'LongInt').SetInt( 45);
7634: 'WINHTTP_OPTION_CLIENT_CERT_CONTEXT', 'LongInt').SetInt( 47);
7635: 'WINHTTP_OPTION_REQUEST_PRIORITY', 'LongInt').SetInt( 58);
7636: 'WINHTTP_OPTION_HTTP_VERSION', 'LongInt').SetInt( 59);
7637: 'WINHTTP_OPTION_DISABLE_FEATURE', 'LongInt').SetInt( 63);
7638: 'WINHTTP_OPTION_CODEPAGE', 'LongInt').SetInt( 68);
7639: 'WINHTTP_OPTION_MAX_CONNS_PER_SERVER', 'LongInt').SetInt( 73);
7640: 'WINHTTP_OPTION_MAX_CONNS_PER_1_0_SERVER', 'LongInt').SetInt( 74);
7641: 'WINHTTP_OPTION_AUTOLOGON_POLICY', 'LongInt').SetInt( 77);
7642: 'WINHTTP_OPTION_SERVER_CERT_CONTEXT', 'LongInt').SetInt( 78);
7643: 'WINHTTP_OPTION_ENABLE_FEATURE', 'LongInt').SetInt( 79);
7644: 'WINHTTP_OPTION_WORKER_THREAD_COUNT', 'LongInt').SetInt( 80);
7645: 'WINHTTP_OPTION_PASSPORT_COBRANDING_TEXT', 'LongInt').SetInt( 81);
7646: 'WINHTTP_OPTION_PASSPORT_COBRANDING_URL', 'LongInt').SetInt( 82);
7647: 'WINHTTP_OPTION_CONFIGURE_PASSPORT_AUTH', 'LongInt').SetInt( 83);
7648: 'WINHTTP_OPTION_SECURE_PROTOCOLs', 'LongInt').SetInt( 84);
7649: 'WINHTTP_OPTION_ENABLETRACING', 'LongInt').SetInt( 85);
7650: 'WINHTTP_OPTION_PASSPORT_SIGN_OUT', 'LongInt').SetInt( 86);
7651: 'WINHTTP_OPTION_PASSPORT_RETURN_URL', 'LongInt').SetInt( 87);
7652: 'WINHTTP_OPTION_REDIRECT_POLICY', 'LongInt').SetInt( 88);
7653: 'WINHTTP_OPTION_MAX_HTTP_AUTOMATIC_REDIRECTS', 'LongInt').SetInt( 89);
7654: 'WINHTTP_OPTION_MAX_HTTP_STATUS_CONTINUE', 'LongInt').SetInt( 90);
7655: 'WINHTTP_OPTION_MAX_RESPONSE_HEADER_SIZE', 'LongInt').SetInt( 91);
7656: 'WINHTTP_OPTION_MAX_RESPONSE_DRAIN_SIZE', 'LongInt').SetInt( 92);
7657: 'WINHTTP_LAST_OPTION', 'LongInt').SetInt(92);
7658: 'WINHTTP_OPTION_USERNAME', 'LongWord').SetUInt( $1000);
7659: 'WINHTTP_OPTION_PASSWORD', 'LongWord').SetUInt( $1001);
7660: 'WINHTTP_OPTION_PROXY_USERNAME', 'LongWord').SetUInt( $1002);
7661: 'WINHTTP_OPTION_PROXY_PASSWORD', 'LongWord').SetUInt( $1003);
7662: 'WINHTTP_CONNS_PER_SERVER_UNLIMITED', 'LongWord').SetUInt( $FFFFFFFF);
7663: 'WINHTTP_AUTOLOGON_SECURITY_LEVEL_MEDIUM', 'LongInt').SetInt( 0);
7664: 'WINHTTP_AUTOLOGON_SECURITY_LEVEL_LOW', 'LongInt').SetInt( 1);
7665: 'WINHTTP_AUTOLOGON_SECURITY_LEVEL_HIGH', 'LongInt').SetInt( 2);
7666: 'WINHTTP_AUTOLOGON_SECURITY_LEVEL_DEFAULT', 'LongInt').SetInt(0);
7667: 'WINHTTP_OPTION_REDIRECT_POLICY_NEVER', 'LongInt').SetInt( 0);
7668: 'WINHTTP_OPTION_REDIRECT_POLICY_DISALLOW_HTTPS_TO_HTTP', 'LongInt').SetInt( 1);
7669: 'WINHTTP_OPTION_REDIRECT_POLICY_ALWAYS', 'LongInt').SetInt( 2);
7670: 'WINHTTP_OPTION_REDIRECT_POLICY_LAST', 'LongInt').SetInt(2);
7671: 'WINHTTP_OPTION_REDIRECT_POLICY_DEFAULT', 'LongInt').SetInt(1);
7672: 'WINHTTP_DISABLE_PASSPORT_AUTH', 'LongWord').SetUInt( $00000000);
7673: 'WINHTTP_ENABLE_PASSPORT_AUTH', 'LongWord').SetUInt( $10000000);
7674: 'WINHTTP_DISABLE_PASSPORT_KEYRING', 'LongWord').SetUInt( $20000000);
7675: 'WINHTTP_ENABLE_PASSPORT_KEYRING', 'LongWord').SetUInt( $40000000);
7676: 'WINHTTP_DISABLE_COOKIES', 'LongWord').SetUInt( $00000001);
7677: 'WINHTTP_DISABLE_REDIRECTS', 'LongWord').SetUInt( $00000002);
7678: 'WINHTTP_DISABLE_AUTHENTICATION', 'LongWord').SetUInt( $00000004);
7679: 'WINHTTP_DISABLE_KEEP_ALIVE', 'LongWord').SetUInt( $00000008);
7680: 'WINHTTP_ENABLE_SSL_REVOCATION', 'LongWord').SetUInt( $00000001);
7681: 'WINHTTP_ENABLE_SSL_REVERT_IMPERSONATION', 'LongWord').SetUInt( $00000002);
7682: 'WINHTTP_HANDLE_TYPE_SESSION', 'LongInt').SetInt( 1);
7683: 'WINHTTP_HANDLE_TYPE_CONNECT', 'LongInt').SetInt( 2);
7684: 'WINHTTP_HANDLE_TYPE_REQUEST', 'LongInt').SetInt( 3);
7685: 'WINHTTP_AUTH_SCHEME_BASIC', 'LongWord').SetUInt( $00000001);
7686: 'WINHTTP_AUTH_SCHEME_NTLM', 'LongWord').SetUInt( $00000002);
7687: 'WINHTTP_AUTH_SCHEME_PASSPORT', 'LongWord').SetUInt( $00000004);
7688: 'WINHTTP_AUTH_SCHEME_DIGEST', 'LongWord').SetUInt( $00000008);
7689: 'WINHTTP_AUTH_SCHEME_NEGOTIATE', 'LongWord').SetUInt( $00000010);
7690: 'WINHTTP_AUTH_TARGET_SERVER', 'LongWord').SetUInt( $00000000);
7691: 'WINHTTP_AUTH_TARGET_PROXY', 'LongWord').SetUInt( $00000001);
7692: 'SECURITY_FLAG_SECURE', 'LongWord').SetUInt( $00000001);
7693: 'SECURITY_FLAG_STRENGTH_WEAK', 'LongWord').SetUInt( $10000000);
7694: 'SECURITY_FLAG_STRENGTH_MEDIUM', 'LongWord').SetUInt( $40000000);
7695: 'SECURITY_FLAG_STRENGTH_STRONG', 'LongWord').SetUInt( $20000000);
7696: 'WINHTTP_CALLBACK_STATUS_FLAG_CERT_REV_FAILED', 'LongWord').SetUInt( $00000001);
7697: 'WINHTTP_CALLBACK_STATUS_FLAG_INVALID_CERT', 'LongWord').SetUInt( $00000002);
7698: 'WINHTTP_CALLBACK_STATUS_FLAG_CERT_REVOKED', 'LongWord').SetUInt( $00000004);
7699: 'WINHTTP_CALLBACK_STATUS_FLAG_INVALID_CA', 'LongWord').SetUInt( $00000008);
7700: 'WINHTTP_CALLBACK_STATUS_FLAG_CERT_CN_INVALID', 'LongWord').SetUInt( $00000010);
7701: 'WINHTTP_CALLBACK_STATUS_FLAG_CERT_DATE_INVALID', 'LongWord').SetUInt( $00000020);
7702: 'WINHTTP_CALLBACK_STATUS_FLAG_CERT_WRONG_USAGE', 'LongWord').SetUInt( $00000040);
7703: 'WINHTTP_CALLBACK_STATUS_FLAG_SECURITY_CHANNEL_ERROR', 'LongWord').SetUInt( $80000000);
7704: 'WINHTTP_FLAG_SECURE_PROTOCOL_SSL2', 'LongWord').SetUInt( $00000008);
7705: 'WINHTTP_FLAG_SECURE_PROTOCOL_SSL3', 'LongWord').SetUInt( $00000020);
7706: 'WINHTTP_FLAG_SECURE_PROTOCOL_TLS1', 'LongWord').SetUInt( $00000080);
7707: 'WINHTTP_CALLBACK_STATUS_RESOLVING_NAME', 'LongWord').SetUInt( $00000001);
7708: 'WINHTTP_CALLBACK_STATUS_NAME_RESOLVED', 'LongWord').SetUInt( $00000002);
7709: 'WINHTTP_CALLBACK_STATUS_CONNECTING_TO_SERVER', 'LongWord').SetUInt( $00000004);

```

```

7710: 'WINHTTP_CALLBACK_STATUS_CONNECTED_TO_SERVER', 'LongWord').SetUInt( $00000008);
7711: 'WINHTTP_CALLBACK_STATUS_SENDING_REQUEST', 'LongWord').SetUInt( $00000010);
7712: 'WINHTTP_CALLBACK_STATUS_REQUEST_SENT', 'LongWord').SetUInt( $00000020);
7713: 'WINHTTP_CALLBACK_STATUS_RECEIVING_RESPONSE', 'LongWord').SetUInt( $00000040);
7714: 'WINHTTP_CALLBACK_STATUS_RESPONSE_RECEIVED', 'LongWord').SetUInt( $00000080);
7715: 'WINHTTP_CALLBACK_STATUS_CLOSING_CONNECTION', 'LongWord').SetUInt( $00000100);
7716: 'WINHTTP_CALLBACK_STATUS_CONNECTION_CLOSED', 'LongWord').SetUInt( $00000200);
7717: 'WINHTTP_CALLBACK_STATUS_HANDLE_CREATED', 'LongWord').SetUInt( $00000400);
7718: 'WINHTTP_CALLBACK_STATUS_HANDLE_CLOSING', 'LongWord').SetUInt( $00000800);
7719: 'WINHTTP_CALLBACK_STATUS_DETECTING_PROXY', 'LongWord').SetUInt( $00001000);
7720: 'WINHTTP_CALLBACK_STATUS_REDIRECT', 'LongWord').SetUInt( $00004000);
7721: 'WINHTTP_CALLBACK_STATUS_INTERMEDIATE_RESPONSE', 'LongWord').SetUInt( $00008000);
7722: 'WINHTTP_CALLBACK_STATUS_SECURE_FAILURE', 'LongWord').SetUInt( $00010000);
7723: 'WINHTTP_CALLBACK_STATUS_HEADERS_AVAILABLE', 'LongWord').SetUInt( $00020000);
7724: 'WINHTTP_CALLBACK_STATUS_DATA_AVAILABLE', 'LongWord').SetUInt( $00040000);
7725: 'WINHTTP_CALLBACK_STATUS_READ_COMPLETE', 'LongWord').SetUInt( $00080000);
7726: 'WINHTTP_CALLBACK_STATUS_WRITE_COMPLETE', 'LongWord').SetUInt( $00100000);
7727: 'WINHTTP_CALLBACK_STATUS_REQUEST_ERROR', 'LongWord').SetUInt( $00200000);
7728: 'WINHTTP_CALLBACK_STATUS_SENDREQUEST_COMPLETE', 'LongWord').SetUInt( $00400000);
7729: 'API_RECEIVE_RESPONSE', 'LongInt').SetInt( ( 1 ));
7730: 'API_QUERY_DATA_AVAILABLE', 'LongInt').SetInt( ( 2 ));
7731: 'API_READ_DATA', 'LongInt').SetInt( ( 3 ));
7732: 'API_WRITE_DATA', 'LongInt').SetInt( ( 4 ));
7733: 'API_SEND_REQUEST', 'LongInt').SetInt( ( 5 ));
7734: 'WINHTTP_CALLBACK_FLAG_DETECTING_PROXY', 'longword').SetUInt( $00001000);
7735: 'WINHTTP_CALLBACK_FLAG_REDIRECT', 'longword').SetUInt( $00004000);
7736: 'WINHTTP_CALLBACK_FLAG_INTERMEDIATE_RESPONSE', 'longword').SetUInt( $00008000);
7737: 'WINHTTP_CALLBACK_FLAG_SECURE_FAILURE', 'longword').SetUInt( $00010000);
7738: 'WINHTTP_CALLBACK_FLAG_SENDREQUEST_COMPLETE', 'longword').SetUInt( $00400000);
7739: 'WINHTTP_CALLBACK_FLAG_HEADERS_AVAILABLE', 'longword').SetUInt( $00020000);
7740: 'WINHTTP_CALLBACK_FLAG_DATA_AVAILABLE', 'longword').SetUInt( $00040000);
7741: 'WINHTTP_CALLBACK_FLAG_READ_COMPLETE', 'longword').SetUInt( $00080000);
7742: 'WINHTTP_CALLBACK_FLAG_WRITE_COMPLETE', 'longword').SetUInt( $00100000);
7743: 'WINHTTP_CALLBACK_FLAG_REQUEST_ERROR', 'longword').SetUInt( $00200000);
7744: 'WINHTTP_CALLBACK_FLAG_ALL_NOTIFICATIONS', 'LongWord').SetUInt( $ffffffff);
7745: 'WINHTTP_INVALID_STATUS_CALLBACK', 'LongInt').SetInt( ( - 1 ));
7746: 'WINHTTP_QUERY_MIME_VERSION', 'LongInt').SetInt( 0);
7747: 'WINHTTP_QUERY_CONTENT_TYPE', 'LongInt').SetInt( 1);
7748: 'WINHTTP_QUERY_CONTENT_TRANSFER_ENCODING', 'LongInt').SetInt( 2);
7749: 'WINHTTP_QUERY_CONTENT_ID', 'LongInt').SetInt( 3);
7750: 'WINHTTP_QUERY_CONTENT_DESCRIPTION', 'LongInt').SetInt( 4);
7751: 'WINHTTP_QUERY_CONTENT_LENGTH', 'LongInt').SetInt( 5);
7752: 'WINHTTP_QUERY_CONTENT_LANGUAGE', 'LongInt').SetInt( 6);
7753: 'WINHTTP_QUERY_ALLOW', 'LongInt').SetInt( 7);
7754: 'WINHTTP_QUERY_PUBLIC', 'LongInt').SetInt( 8);
7755: 'WINHTTP_QUERY_DATE', 'LongInt').SetInt( 9);
7756: 'WINHTTP_QUERY_EXPIRES', 'LongInt').SetInt( 10);
7757: 'WINHTTP_QUERY_LAST_MODIFIED', 'LongInt').SetInt( 11);
7758: 'WINHTTP_QUERY_MESSAGE_ID', 'LongInt').SetInt( 12);
7759: 'WINHTTP_QUERY_URI', 'LongInt').SetInt( 13);
7760: 'WINHTTP_QUERY_DERIVED_FROM', 'LongInt').SetInt( 14);
7761: 'WINHTTP_QUERY_COST', 'LongInt').SetInt( 15);
7762: 'WINHTTP_QUERY_LINK', 'LongInt').SetInt( 16);
7763: 'WINHTTP_QUERY_PRAGMA', 'LongInt').SetInt( 17);
7764: 'WINHTTP_QUERY_VERSION', 'LongInt').SetInt( 18);
7765: 'WINHTTP_QUERY_STATUS_CODE', 'LongInt').SetInt( 19);
7766: 'WINHTTP_QUERY_STATUS_TEXT', 'LongInt').SetInt( 20);
7767: 'WINHTTP_QUERY_RAW_HEADERS', 'LongInt').SetInt( 21);
7768: 'WINHTTP_QUERY_RAW_HEADERS_CRLF', 'LongInt').SetInt( 22);
7769: 'WINHTTP_QUERY_CONNECTION', 'LongInt').SetInt( 23);
7770: 'WINHTTP_QUERY_ACCEPT', 'LongInt').SetInt( 24);
7771: 'WINHTTP_QUERY_ACCEPT_CHARSET', 'LongInt').SetInt( 25);
7772: 'WINHTTP_QUERY_ACCEPT_ENCODING', 'LongInt').SetInt( 26);
7773: 'WINHTTP_QUERY_ACCEPT_LANGUAGE', 'LongInt').SetInt( 27);
7774: 'WINHTTP_QUERY_AUTHORIZATION', 'LongInt').SetInt( 28);
7775: 'WINHTTP_QUERY_CONTENT_ENCODING', 'LongInt').SetInt( 29);
7776: 'WINHTTP_QUERY_FORWARDED', 'LongInt').SetInt( 30);
7777: 'WINHTTP_QUERY_FROM', 'LongInt').SetInt( 31);
7778: 'WINHTTP_QUERY_IF_MODIFIED_SINCE', 'LongInt').SetInt( 32);
7779: 'WINHTTP_QUERY_LOCATION', 'LongInt').SetInt( 33);
7780: 'WINHTTP_QUERY_ORIG_URI', 'LongInt').SetInt( 34);
7781: 'WINHTTP_QUERY_REFERER', 'LongInt').SetInt( 35);
7782: 'WINHTTP_QUERY_RETRY_AFTER', 'LongInt').SetInt( 36);
7783: 'WINHTTP_QUERY_SERVER', 'LongInt').SetInt( 37);
7784: 'WINHTTP_QUERY_TITLE', 'LongInt').SetInt( 38);
7785: 'WINHTTP_QUERY_USER_AGENT', 'LongInt').SetInt( 39);
7786: 'WINHTTP_QUERY_WWW_AUTHENTICATE', 'LongInt').SetInt( 40);
7787: 'WINHTTP_QUERY_PROXY_AUTHENTICATE', 'LongInt').SetInt( 41);
7788: 'WINHTTP_QUERY_ACCEPT_RANGES', 'LongInt').SetInt( 42);
7789: 'WINHTTP_QUERY_SET_COOKIE', 'LongInt').SetInt( 43);
7790: 'WINHTTP_QUERY_COOKIE', 'LongInt').SetInt( 44);
7791: 'WINHTTP_QUERY_REQUEST_METHOD', 'LongInt').SetInt( 45);
7792: 'WINHTTP_QUERY_REFRESH', 'LongInt').SetInt( 46);
7793: 'WINHTTP_QUERY_CONTENT_DISPOSITION', 'LongInt').SetInt( 47);
7794: 'WINHTTP_QUERY_AGE', 'LongInt').SetInt( 48);
7795: 'WINHTTP_QUERY_CACHE_CONTROL', 'LongInt').SetInt( 49);
7796: 'WINHTTP_QUERY_CONTENT_BASE', 'LongInt').SetInt( 50);
7797: 'WINHTTP_QUERY_CONTENT_LOCATION', 'LongInt').SetInt( 51);
7798: 'WINHTTP_QUERY_CONTENT_MD5', 'LongInt').SetInt( 52);

```



```

7799: 'WINHTTP_QUERY_CONTENT_RANGE', 'LongInt').SetInt( 53);
7800: 'WINHTTP_QUERY_ETAG', 'LongInt').SetInt( 54);
7801: 'WINHTTP_QUERY_HOST', 'LongInt').SetInt( 55);
7802: 'WINHTTP_QUERY_IF_MATCH', 'LongInt').SetInt( 56);
7803: 'WINHTTP_QUERY_IF_NONE_MATCH', 'LongInt').SetInt( 57);
7804: 'WINHTTP_QUERY_IF_RANGE', 'LongInt').SetInt( 58);
7805: 'WINHTTP_QUERY_IF_UNMODIFIED_SINCE', 'LongInt').SetInt( 59);
7806: 'WINHTTP_QUERY_MAX_FORWARDS', 'LongInt').SetInt( 60);
7807: 'WINHTTP_QUERY_PROXY_AUTHORIZATION', 'LongInt').SetInt( 61);
7808: 'WINHTTP_QUERY_RANGE', 'LongInt').SetInt( 62);
7809: 'WINHTTP_QUERY_TRANSFER_ENCODING', 'LongInt').SetInt( 63);
7810: 'WINHTTP_QUERY_UPGRADE', 'LongInt').SetInt( 64);
7811: 'WINHTTP_QUERY_VARY', 'LongInt').SetInt( 65);
7812: 'WINHTTP_QUERY_VIA', 'LongInt').SetInt( 66);
7813: 'WINHTTP_QUERY_WARNING', 'LongInt').SetInt( 67);
7814: 'WINHTTP_QUERY_EXPECT', 'LongInt').SetInt( 68);
7815: 'WINHTTP_QUERY_PROXY_CONNECTION', 'LongInt').SetInt( 69);
7816: 'WINHTTP_QUERY_UNLESS_MODIFIED_SINCE', 'LongInt').SetInt( 70);
7817: 'WINHTTP_QUERY_PROXY_SUPPORT', 'LongInt').SetInt( 75);
7818: 'WINHTTP_QUERY_AUTHENTICATION_INFO', 'LongInt').SetInt( 76);
7819: 'WINHTTP_QUERY_PASSPORT_URLS', 'LongInt').SetInt( 77);
7820: 'WINHTTP_QUERY_PASSPORT_CONFIG', 'LongInt').SetInt( 78);
7821: 'WINHTTP_QUERY_MAX', 'LongInt').SetInt( 78);
7822: 'WINHTTP_QUERY_CUSTOM', 'LongInt').SetInt( 65535);
7823: 'WINHTTP_QUERY_FLAG_REQUEST_HEADERS', 'LongWord').SetUInt( $80000000);
7824: 'WINHTTP_QUERY_FLAG_SYSTEMTIME', 'LongWord').SetUInt( $40000000);
7825: 'WINHTTP_QUERY_FLAG_NUMBER', 'LongWord').SetUInt( $20000000);
7826: 'HTTP_STATUS_CONTINUE', 'LongInt').SetInt( 100);
7827: 'HTTP_STATUS_SWITCH_PROTOCOLS', 'LongInt').SetInt( 101);
7828: 'HTTP_STATUS_OK', 'LongInt').SetInt( 200);
7829: 'HTTP_STATUS_CREATED', 'LongInt').SetInt( 201);
7830: 'HTTP_STATUS_ACCEPTED', 'LongInt').SetInt( 202);
7831: 'HTTP_STATUS_PARTIAL', 'LongInt').SetInt( 203);
7832: 'HTTP_STATUS_NO_CONTENT', 'LongInt').SetInt( 204);
7833: 'HTTP_STATUS_RESET_CONTENT', 'LongInt').SetInt( 205);
7834: 'HTTP_STATUS_PARTIAL_CONTENT', 'LongInt').SetInt( 206);
7835: 'HTTP_STATUS_WEBDAV_MULTI_STATUS', 'LongInt').SetInt( 207);
7836: 'HTTP_STATUS_AMBIGUOUS', 'LongInt').SetInt( 300);
7837: 'HTTP_STATUS_MOVED', 'LongInt').SetInt( 301);
7838: 'HTTP_STATUS_REDIRECT', 'LongInt').SetInt( 302);
7839: 'HTTP_STATUS_REDIRECT_METHOD', 'LongInt').SetInt( 303);
7840: 'HTTP_STATUS_NOT_MODIFIED', 'LongInt').SetInt( 304);
7841: 'HTTP_STATUS_USE_PROXY', 'LongInt').SetInt( 305);
7842: 'HTTP_STATUS_REDIRECT_KEEP_VERB', 'LongInt').SetInt( 307);
7843: 'HTTP_STATUS_BAD_REQUEST', 'LongInt').SetInt( 400);
7844: 'HTTP_STATUS_DENIED', 'LongInt').SetInt( 401);
7845: 'HTTP_STATUS_PAYMENT_REQ', 'LongInt').SetInt( 402);
7846: 'HTTP_STATUS_FORBIDDEN', 'LongInt').SetInt( 403);
7847: 'HTTP_STATUS_NOT_FOUND', 'LongInt').SetInt( 404);
7848: 'HTTP_STATUS_BAD_METHOD', 'LongInt').SetInt( 405);
7849: 'HTTP_STATUS_NONE_ACCEPTABLE', 'LongInt').SetInt( 406);
7850: 'HTTP_STATUS_PROXY_AUTH_REQ', 'LongInt').SetInt( 407);
7851: 'HTTP_STATUS_REQUEST_TIMEOUT', 'LongInt').SetInt( 408);
7852: 'HTTP_STATUS_CONFLICT', 'LongInt').SetInt( 409);
7853: 'HTTP_STATUS_GONE', 'LongInt').SetInt( 410);
7854: 'HTTP_STATUS_LENGTH_REQUIRED', 'LongInt').SetInt( 411);
7855: 'HTTP_STATUS_PRECOND_FAILED', 'LongInt').SetInt( 412);
7856: 'HTTP_STATUS_REQUEST_TOO_LARGE', 'LongInt').SetInt( 413);
7857: 'HTTP_STATUS_URI_TOO_LONG', 'LongInt').SetInt( 414);
7858: 'HTTP_STATUS_UNSUPPORTED_MEDIA', 'LongInt').SetInt( 415);
7859: 'HTTP_STATUS_RETRY_WITH', 'LongInt').SetInt( 449);
7860: 'HTTP_STATUS_SERVER_ERROR', 'LongInt').SetInt( 500);
7861: 'HTTP_STATUS_NOT_SUPPORTED', 'LongInt').SetInt( 501);
7862: 'HTTP_STATUS_BAD_GATEWAY', 'LongInt').SetInt( 502);
7863: 'HTTP_STATUS_SERVICE_UNAVAIL', 'LongInt').SetInt( 503);
7864: 'HTTP_STATUS_GATEWAY_TIMEOUT', 'LongInt').SetInt( 504);
7865: 'HTTP_STATUS_VERSION_NOT_SUP', 'LongInt').SetInt( 505);
7866: 'HTTP_STATUS_FIRST', 'longint').SetInt(100);
7867: 'HTTP_STATUS_LAST', 'longint').SetInt(505);
7868: 'WINHTTP_NO_REFERER', 'longint').SetInt(0);
7869: 'WINHTTP_DEFAULT_ACCEPT_TYPES', 'longint').SetInt(0);
7870: 'WINHTTP_ADDREQ_INDEX_MASK', 'LongWord').SetUInt( $0000FFFF);
7871: 'WINHTTP_ADDREQ_FLAGS_MASK', 'LongWord').SetUInt( $FFFF0000);
7872: 'WINHTTP_ADDREQ_FLAG_ADD_IF_NEW', 'LongWord').SetUInt( $10000000);
7873: 'WINHTTP_ADDREQ_FLAG_ADD', 'LongWord').SetUInt( $20000000);
7874: 'WINHTTP_ADDREQ_FLAG_COALESCE_WITH_COMMA', 'LongWord').SetUInt( $40000000);
7875: 'WINHTTP_ADDREQ_FLAG_COALESCE_WITH_SEMICOLON', 'LongWord').SetUInt( $01000000);
7876: 'WINHTTP_ADDREQ_FLAG_COALESCE', 'longword').SetInt($40000000);
7877: 'WINHTTP_ADDREQ_FLAG_REPLACE', 'LongWord').SetUInt( $80000000);
7878: 'WINHTTP_NO_ADDITIONAL_HEADERS', 'LongWord').SetUInt(0);
7879: 'WINHTTP_NO_REQUEST_DATA', 'LongWord').SetUInt(0);
7880: 'WINHTTP_HEADER_NAME_BY_INDEX', 'LongWord').SetUInt(0);
7881: 'WINHTTP_NO_OUTPUT_BUFFER', 'LongWord').SetUInt(0);
7882: 'WINHTTP_NO_HEADER_INDEX', 'LongWord').SetUInt(0);
7883: 'WINHTTP_ERROR_BASE', 'LongInt').SetInt( 12000);
7884: 'WINHTTP_FLAG_NULL_CODEPAGE', 'LongWord').SetUInt( $00000008);
7885: Function WinHttpCloseHandle( hInternet : HINTERNET) : BOOL;
7886: WINHTTP_STATUS_CALLBACK', 'TFarProc';
7887: TFWinHttpStatusCallback', 'WINHTTP_STATUS_CALLBACK');

```

```

7888: //PFNWinHttpStatusCallback', '^TFNWinHttpStatusCallback // will not work');
7889: // LPWINHTTP_STATUS_CALLBACK', 'PFNWinHttpStatusCallback');
7890: //Function WinHttpSetStatusCallback( hInternet : HINTERNET; lpfnInternetCallback :
PFNWinHttpStatusCallback; dwNotificationFlags : Dword; dwReserved : DWORD) : PFNWinHttpStatusCallback);
7891: end;
7892:
7893:
7894: procedure SIRegister_ALWinHttpClient(CL: TPSPascalCompiler);
7895: begin
7896:   DWORD_PTR, 'DWORD');
7897:   TALWinHttpClientInternetOpenAccessType, '( wHttpAt_NO_PROXY, wH
+ 'tAt_DEFAULT_PROXY, wHttpAt_NAMED_PROXY )');
7898:   TALWinHttpClientInternetOption, '( wHttpIo_Async, wHttpIo_BYPAS
+ 'S_PROXY_CACHE, wHttpIo_ESCAPE_DISABLE, wHttpIo_ESCAPE_DISABLE_QUERY, wHttp
+ 'Io_ESCAPE_PERCENT, wHttpIo_NULL_CODEPAGE, wHttpIo_REFRESH, wHttpIo_SECURE,
7900:   + ' wHttpIo_Keep_connection, wHttpIo_No_cookies, wHttpIo_No_auto_redirect )');
7901:   TALWinHttpClientInternetOptionSet, 'set of TALWinHttpClientInternetOption');
7902:   // TALWinHttpClientStatusChangeEvent', 'Procedure ( sender : TObject
7903:   // + 't; InternetStatus : Dword; StatusInformation : Pointer; StatusInformationLength : Dword)');
7904:   SIRegister_TALWinHttpClient(CL);
7905: end;
7906:
7907:
7908:
7909:
7910: procedure SIRegister_ALFcnExecute(CL: TPSPascalCompiler);
7911: begin
7912:   TStartupInfo, 'TStartupInfo');
7913:   'SE_CREATE_TOKEN_NAME', 'String').SetString( 'SeCreateTokenPrivilege');
7914:   'SE_ASSIGNPRIMARYTOKEN_NAME', 'String').SetString( 'SeAssignPrimaryTokenPrivilege');
7915:   'SE_LOCK_MEMORY_NAME', 'String').SetString( 'SeLockMemoryPrivilege');
7916:   'SE_INCREASE_QUOTA_NAME', 'String').SetString( 'SeIncreaseQuotaPrivilege');
7917:   'SE_UNSOLICITED_INPUT_NAME', 'String').SetString( 'SeUnsolicitedInputPrivilege');
7918:   'SE_MACHINE_ACCOUNT_NAME', 'String').SetString( 'SeMachineAccountPrivilege');
7919:   'SE_TCB_NAME', 'String').SetString( 'SeTcbPrivilege');
7920:   'SE_SECURITY_NAME', 'String').SetString( 'SeSecurityPrivilege');
7921:   'SE_TAKE_OWNERSHIP_NAME', 'String').SetString( 'SeTakeOwnershipPrivilege');
7922:   'SE_LOAD_DRIVER_NAME', 'String').SetString( 'SeLoadDriverPrivilege');
7923:   'SE_SYSTEM_PROFILE_NAME', 'String').SetString( 'SeSystemProfilePrivilege');
7924:   'SE_SYSTEMTIME_NAME', 'String').SetString( 'SeSystemtimePrivilege');
7925:   'SE_PROF_SINGLE_PROCESS_NAME', 'String').SetString( 'SeProfileSingleProcessPrivilege');
7926:   'SE_INC_BASE_PRIORITY_NAME', 'String').SetString( 'SeIncreaseBasePriorityPrivilege');
7927:   'SE_CREATE_PAGEFILE_NAME', 'String').SetString( 'SeCreatePagefilePrivilege');
7928:   'SE_CREATE_PERMANENT_NAME', 'String').SetString( 'SeCreatePermanentPrivilege');
7929:   'SE_BACKUP_NAME', 'String').SetString( 'SeBackupPrivilege');
7930:   'SE_RESTORE_NAME', 'String').SetString( 'SeRestorePrivilege');
7931:   'SE_SHUTDOWN_NAME', 'String').SetString( 'SeShutdownPrivilege');
7932:   'SE_DEBUG_NAME', 'String').SetString( 'SeDebugPrivilege');
7933:   'SE_AUDIT_NAME', 'String').SetString( 'SeAuditPrivilege');
7934:   'SE_SYSTEM_ENVIRONMENT_NAME', 'String').SetString( 'SeSystemEnvironmentPrivilege');
7935:   'SE_CHANGE_NOTIFY_NAME', 'String').SetString( 'SeChangeNotifyPrivilege');
7936:   'SE_REMOTE_SHUTDOWN_NAME', 'String').SetString( 'SeRemoteShutdownPrivilege');
7937:   'SE_UNDOCK_NAME', 'String').SetString( 'SeUndockPrivilege');
7938:   'SE_SYNC_AGENT_NAME', 'String').SetString( 'SeSyncAgentPrivilege');
7939:   'SE_ENABLE_DELEGATION_NAME', 'String').SetString( 'SeEnableDelegationPrivilege');
7940:   'SE_MANAGE_VOLUME_NAME', 'String').SetString( 'SeManageVolumePrivilege');
7941:   Function ALGetEnvironmentString : AnsiString);
7942:   Function ALWinExec32( const FileName : AnsiString; InputStream : Tstream; OutputStream : TStream) :
Dword);
7943:   Function ALWinExecAndWait32( FileName : AnsiString; Visibility : integer) : DWORD);
7944:   Function ALWinExecAndWait32V2( FileName : AnsiString; Visibility : integer) : DWORD);
7945:   Function ALNTSetPrivilege( sPrivilege : AnsiString; bEnabled : Boolean) : Boolean);
7946: end;
7947:
7948: procedure SIRegister_ALGraphic(CL: TPSPascalCompiler);
7949: begin
7950:   NativeInt, 'Integer');
7951:   NativeUInt, 'Cardinal');
7952:   TALResamplingFilter, '( sfBox, sfTriangle, sfHermite, sfBell, s
+ 'fSpline, sfLanczos3, sfMitchell )');
7953: end;
7954:
7955:
7956: procedure SIRegister_ALMemCachedClient(CL: TPSPascalCompiler);
7957: begin
7958:   TALMemCachedClient_responseType, '( rpEND, rpOK, rpCRLF, rpSTOR
+ 'AGE, rpRETRIEVAL, rpRETRIEVALS, rpDELETE, rpINCRDECR, rpTOUCH, rpNONE )');
7959:   TALMemCachedClient_StoredItem, 'record key : ansiString; flags
+ ' integer; cas_id : int64; data : ansiString; end');
7960:   TALMemCachedClient_StoredItems, 'array of TALMemCachedClient_StoredItem');
7961:   SIRegister_EALMemCachedClientException(CL);
7962:   SIRegister_TALBaseMemCachedClient(CL);
7963:   SIRegister_TALMemCachedClient(CL);
7964:   SIRegister_TALMemCachedConnectionPoolContainer(CL);
7965:   SIRegister_TALMemCachedConnectionPoolClient(CL);
7966: end;
7967:
7968:
7969:
7970: (*-----*)
7971: procedure SIRegister_ALSMTPClient(CL: TPSPascalCompiler);
7972: begin
7973:   TALSMTPClientAuthType, '( AlsmtpClientAuthNone, alsmtpClientAut
+ 'hPlain, AlsmtpClientAuthLogin, AlsmtpClientAuthCramMD5, AlsmtpClientAuthCr

```

```

7975:     + 'amShal, AlsmtpClientAuthAutoSelect ');
7976:     TALSMTPClientAuthTypeSet, 'set of TALSMTPClientAuthType';
7977:     SIRegister_TALSMTPClient(CL);
7978: end;
7979:
7980: procedure SIRegister_ALHintBalloon(CL: TPSPascalCompiler);
7981: begin
7982:   CL.AddClassN(CL.FindClass('TOBJECT'), 'TALHintBalloon');
7983:   TALHintBalloonMsgType, '( bmtInfo, bmtError, bmtWarning, bmtNone )';
7984:   TALHintBalloonArrowPosition, '( bapTopLeft, bapTopRight, bapBottomLeft, bapBottomRight )';
7985:   TALHintBalloonAnimationType, '( batNone, batBlend, batCenter, b'
7986:     + 'atSlideLeftToRight, batSlideRightToLeft, batSlideTopToBottom, batSlideBott'
7987:     + 'omToTop, batWipeLeftToRight, batWipeRightToLeft, batWipeTopToBottom, batWipeBottomToTop )';
7988:   SIRegister_TALHintBalloon(CL);
7989:   SIRegister_TALHintBalloonControl(CL);
7990:   //CL.AddDelphiFunction('Procedure Register');
7991: end;
7992:
7993: procedure SIRegister_ALXmlDoc(CL: TPSPascalCompiler);
7994: begin
7995:   'cALXMLNodeMaxListSize', 'LongInt').SetInt( MaxInt div 16);
7996:   CL.AddClassN(CL.FindClass('TOBJECT'), 'TALXMLNode');
7997:   CL.AddClassN(CL.FindClass('TOBJECT'), 'TALXMLNodeList');
7998:   CL.AddClassN(CL.FindClass('TOBJECT'), 'TALXMLDocument');
7999:   TALXMLParseProcessingInstructionEvent, 'Procedure (Sender: TObject; const Target, Data: AnsiString)';
8000:   TALXMLParseTextEvent, 'Procedure (Sender: TObject; const str: AnsiString)';
8001:   TALXMLParseStartElementEvent, 'Procedure (Sender: TObject; co'
8002:     + 'nst Name: AnsiString; const Attributes: TALStrings)';
8003:   TALXMLParseEndElementEvent, 'Procedure (Sender: TObject; const Name: AnsiString)';
8004:   TALXMLNodeType, '( ntReserved, ntElement, ntAttribute, ntText, '
8005:     + 'ntCDATA, ntEntityRef, ntEntity, ntProcessingInstr, ntComment, ntDocument, '
8006:     + 'ntDocType, ntDocFragment, ntNotation )';
8007:   TALXMLDocOption, '( doNodeAutoCreate, doNodeAutoIndent )';
8008:   TALXMLDocOptions, 'set of TALXMLDocOption';
8009:   TALXMLParseOption, '( poPreserveWhiteSpace, poIgnoreXMLReferences )';
8010:   TALXMLParseOptions, 'set of TALXMLParseOption';
8011:   TALXMLPrologItem, '( xpVersion, xpEncoding, xpStandalone )';
8012:   PALPointerXMLNodeList, '^TALPointerXMLNodeList // will not work';
8013:   SIRegister_EALXMLDocError(CL);
8014:   SIRegister_TALXMLNodeList(CL);
8015:   SIRegister_TALXMLNode(CL);
8016:   SIRegister_TALXmlElementNode(CL);
8017:   SIRegister_TALXmlAttributeNode(CL);
8018:   SIRegister_TALXmlTextNode(CL);
8019:   SIRegister_TALXmlDocumentNode(CL);
8020:   SIRegister_TALXmlCommentNode(CL);
8021:   SIRegister_TALXmlProcessingInstrNode(CL);
8022:   SIRegister_TALXmlCDATADataNode(CL);
8023:   SIRegister_TALXmlEntityRefNode(CL);
8024:   SIRegister_TALXmlEntityNode(CL);
8025:   SIRegister_TALXmlDocTypeNode(CL);
8026:   SIRegister_TALXmlDocFragmentNode(CL);
8027:   SIRegister_TALXmlNotationNode(CL);
8028:   SIRegister_TALXMLDocument(CL);
8029:   'cALXMLUTF8EncodingStr', 'String').SetString( 'UTF-8');
8030:   cALXmlUTF8HeaderStr, 'String').SetString('<?xml version="1.0" encoding="UTF-8" standalone="yes"?>'+
8031:     #13#10);
8032:   CALNSDelim, 'String').SetString( ':');
8033:   CALXML, 'String').SetString( 'xml');
8034:   CALVersion, 'String').SetString( 'version');
8035:   CALEncoding, 'String').SetString( 'encoding');
8036:   CALStandalone, 'String').SetString( 'standalone');
8037:   CALDefaultNodeIndent, 'String').SetString( ' ');
8038:   CALXmlDocument, 'String').SetString( 'DOCUMENT');
8039:   cAlUTF8Bom, 'String').SetString(#$EF+$BB+$BF);
8040: end;
8041:
8042: procedure SIRegister_ovctimer(CL: TPSPascalCompiler);
8043: begin
8044:   TovTriggerEvent, 'Procedure (Sender: TObject; Handle: Integer; '
8045:     + 'Interval: Cardinal; ElapsedTime: LongInt)';
8046:   //PEventRec, 'TEventRec // will not work';
8047:   TEventRec, 'record erHandle: Integer; erInitTime: LongInt; er'
8048:     + 'Elapsed: LongInt; erInterval: Cardinal; erLastTrigger: LongInt; erOnTri'
8049:     + 'gger: TovTriggerEvent; erEnabled: Boolean; erRecurring: Boolean; end';
8050:   SIRegister_TOvcTimerPool(CL);
8051: end;
8052:
8053: procedure SIRegister_ovcintl(CL: TPSPascalCompiler);
8054: begin
8055:   //TCurrencySt = array[0..5] of Char;
8056:   TCurrencySt, 'array[0..5] of Char';
8057:   TIntlData, 'record CurrencyLtStr: TCurrencySt; CurrencyRtStr: '
8058:     + 'TCurrencySt; DecimalChar: Char; CommaChar: Char; CurrDigits: Byte; Sla'
8059:     + 'shChar: Char; TrueChar: Char; FalseChar: Char; YesChar: Char; NoChar: Char; end';
8060:   SIRegister_TOvcIntlSup(CL);
8061: end;
8062:
8063: procedure SIRegister_ovccmd(CL: TPSPascalCompiler);

```

```

8063: begin
8064:   'DefWsMaxCommands', 'LongInt').SetInt( 40);
8065:   'DefGridMaxCommands', 'LongInt').SetInt( 38);
8066:   TOvcProcessorState', '( stNone, stPartial, stLiteral )';
8067:   TOvcCmdRec', 'record Keys: Longint; end');
8068:   (* TOvcCmdRec = packed record
8069:     case Byte of
8070:       0 : (Key1 : Byte;      {first keys' virtual key code}
8071:          SS1 : Byte;      {shift state of first key}
8072:          Key2 : Byte;      {second keys' virtual key code, if any}
8073:          SS2 : Byte;      {shift state of second key}
8074:          Cmd : Word);      {command to return for this entry}
8075:       1 : (Keys : LongInt); {used for sorting, searching, and storing}
8076:     end; *)
8077:   TUserCommandEvent', 'Procedure ( Sender : TObject; Command : Word)';
8078:   CL.AddClassN(CL.FindClass('TObject'), 'TOvcCommandProcessor');
8079:   //SIRegister_TOvcCommandProcessor(CL);
8080:   SIRegister_TOvcCommandTable(CL);
8081:   SIRegister_TOvcCommandProcessor(CL);
8082: end;
8083:
8084: procedure SIRegister_ovcmru(CL: TPSPascalCompiler);
8085: begin
8086:   TOvcMRUOption', '( moAddAccelerators, moStripPath, moAddSeparator )';
8087:   TOvcMRUAddPosition', '( apAnchor, apTop, apBottom )';
8088:   TOvcMRUStyle', '( msNormal, msSplit )';
8089:   TOvcMRUClickAction', '( caMoveToTop, caRemove, caNoAction )';
8090:   TOvcMRUOptions', 'set of TOvcMRUOption';
8091:   TOvcMRUClickEvent', 'Procedure ( Sender : TObject; const ItemTex'
8092:     + 't : string; var Action : TOvcMRUClickAction)';
8093:   TOvcMRUAddPositions', 'set of TOvcMRUAddPosition';
8094:   SIRegister_TOvcMenuMRU(CL);
8095: end;
8096:
8097: procedure SIRegister_AfSafeSync(CL: TPSPascalCompiler);
8098: begin
8099:   'AfMaxSyncSlots', 'LongInt').SetInt( 64);
8100:   'AfSynchronizeTimeout', 'LongInt').SetInt( 2000);
8101:   TAFSyncSlotID', 'DWORD';
8102:   TAFSyncStatistics', 'record MessagesCount : Integer; TimeoutMess'
8103:     + 'ages : Integer; DisabledMessages : Integer; end';
8104:   TAFSafeSyncEvent', 'Procedure ( ID : TAFSyncSlotID)';
8105:   TAFSafeDirectSyncEvent', 'Procedure';
8106: end;
8107:
8108: procedure SIRegister_AfComPortCore(CL: TPSPascalCompiler);
8109: begin
8110:   'fBinary', 'LongWord').SetUInt( $00000001);
8111:   'fParity', 'LongWord').SetUInt( $00000002);
8112:   'fOutxCtsFlow', 'LongWord').SetUInt( $00000004);
8113:   'fOutxDsrFlow', 'LongWord').SetUInt( $00000008);
8114:   'fDtrControl', 'LongWord').SetUInt( $00000030);
8115:   'fDtrControlDisable', 'LongWord').SetUInt( $00000000);
8116:   'fDtrControlEnable', 'LongWord').SetUInt( $00000010);
8117:   'fDtrControlHandshake', 'LongWord').SetUInt( $00000020);
8118:   'fDsrSensitivity', 'LongWord').SetUInt( $00000040);
8119:   'fTXContinueOnXoff', 'LongWord').SetUInt( $00000080);
8120:   'fOutX', 'LongWord').SetUInt( $00000100);
8121:   'fInX', 'LongWord').SetUInt( $00000200);
8122:   'fErrorChar', 'LongWord').SetUInt( $00000400);
8123:   'fNull', 'LongWord').SetUInt( $00000800);
8124:   'fRtsControl', 'LongWord').SetUInt( $00003000);
8125:   'fRtsControlDisable', 'LongWord').SetUInt( $00000000);
8126:   'fRtsControlEnable', 'LongWord').SetUInt( $00001000);
8127:   'fRtsControlHandshake', 'LongWord').SetUInt( $00002000);
8128:   'fRtsControlToggle', 'LongWord').SetUInt( $00003000);
8129:   'fAbortOnError', 'LongWord').SetUInt( $00004000);
8130:   'fDummy2', 'LongWord').SetUInt( $FFFF8000);
8131:   TAFCoreEvent', '( ceOutFree, ceLineEvent, ceNeedReadData, ceException )';
8132:   CL.AddClassN(CL.FindClass('TObject'), 'EAFComPortCoreError');
8133:   CL.AddClassN(CL.FindClass('TObject'), 'TAFComPortCore');
8134:   TAFComPortCoreEvent', 'Procedure ( Sender : TAFComPortCore; Even'
8135:     + 'tKind : TAFCoreEvent; Data : DWORD)';
8136:   SIRegister_TAFComPortCoreThread(CL);
8137:   SIRegister_TAFComPortEventThread(CL);
8138:   SIRegister_TAFComPortWriteThread(CL);
8139:   SIRegister_TAFComPortCore(CL);
8140:   CL.AddDelphiFunction('Function FormatDeviceName( PortNumber : Integer) : string');
8141: end;
8142:
8143: procedure SIRegister_AfComPort(CL: TPSPascalCompiler);
8144: begin
8145:   TAFBaudrate', '( br110, br300, br600, br1200, br2400, br4800, br'
8146:     + '9600, br14400, br19200, br38400, br56000, br57600, br115200, br128000, br2'
8147:     + '56000, brUser )';
8148:   TAFParity', '( paNone, paOdd, paEven, paMark, paSpace )';
8149:   TAFDatabits', '( db4, db5, db6, db7, db8 )';
8150:   TAFStopbits', '( sbOne, sbOneAndHalf, sbTwo )';
8151:   TAFFlowControl', '( fwNone, fwXOnXOff, fwRtsCts, fwDtrDsr )';

```



```

8152:   TafComOption', '( coParityCheck, coDsrSensitivity, coIgnoreXOff, '
8153:   + ' coErrorChar, coStripNull )');
8154:   TafComOptions', 'set of TafComOption');
8155:   CL.AddClassN(CL.FindClass('TOBJECT'),'EafComPortError');
8156:   TafComPortEventKind', 'TafCoreEvent');
8157:   TafComPortEventData', 'DWORD');
8158:   TafCPTCoreEvent', 'Procedure ( Sender : TObject; EventKind : Taf'
8159:   + 'ComPortEventKind; Data : TafComPortEventData)');
8160:   TafCPTErrorEvent', 'Procedure ( Sender : TObject; Errors : DWORD)';
8161:   TafCPTDataReceivedEvent', 'Procedure (Sender: TObject; Count: Integer)';
8162:   SIRegister_TafCustomSerialPort(CL);
8163:   SIRegister_TafCustomComPort(CL);
8164:   SIRegister_TafComPort(CL);
8165: end;
8166:
8167: procedure SIRegister_AfDataDispatcher(CL: TPSPascalCompiler);
8168: begin
8169:   CL.AddClassN(CL.FindClass('TOBJECT'),'EafDispatcherError');
8170:   TafDispEventKind', '( deData, deWriteBufFree, deClear )';
8171:   TafDispEventKinds', 'set of TafDispEventKind');
8172:   TafDispLineEvent', 'Procedure ( Sender : TObject; const TextLine: String)';
8173:   TafDispLinkEvent', 'Procedure ( Sender : TObject; EventKind : TafDispEventKind)';
8174:   TafDispStreamEvent', 'Procedure ( Sender : TObject; const Position, Size : Integer)';
8175:   TafDispWriteToEvent', 'Procedure ( Sender : TObject; const Data, Size : Integer)';
8176:   CL.AddClassN(CL.FindClass('TOBJECT'),'TafCustomDataDispatcher');
8177:   SIRegister_TafDataDispConnComponent(CL);
8178:   SIRegister_TafDataDispatcherLink(CL);
8179:   SIRegister_TafCustomDataDispatcher(CL);
8180:   SIRegister_TafDataDispatcher(CL);
8181: end;
8182:
8183: procedure SIRegister_AfPortControls(CL: TPSPascalCompiler);
8184: begin
8185:   TafPortCtlOption', '( pcCheckExist, pcDisableOpen, pcHighlightOpen )';
8186:   TafPortCtlOptions', 'set of TafPortCtlOption';
8187:   //TafPortCtlOptions', 'set of ( pcCheckExist, pcDisableOpen, pcHighlightOpen )';
8188:   SIRegister_TafPortComboBox(CL);
8189:   SIRegister_TafPortRadioGroup(CL);
8190: end;
8191:
8192: procedure SIRegister_AfViewers(CL: TPSPascalCompiler);
8193: begin
8194:   'AfCLVMaxLineLength', 'LongInt').SetInt( 1024);
8195:   'UM_UPDATELINECOUNT', 'LongWord').SetUInt( WM_USER + $100);
8196:   SIRegister_TafFontStyleCache(CL);
8197:   CL.AddClassN(CL.FindClass('TOBJECT'),'EafCLVException');
8198:   TafCLVCaretType', '( ctVertical, ctHorizontal, ctBlock )';
8199:   TafCLVCharAttr', 'record BColor: TColor; FColor: TColor; Style: TFontStyles; end';
8200:   //PafCLVCharColors', '^TafCLVCharColors // will not work';
8201:   TafCLVColorMode', '( cmDefault, cmLine, cmChars, cmCheckLength )';
8202:   TafCLVFocusSource', '( fsKey, fsMouse, fsHScroll, fsVScroll )';
8203:   TafCLVLineState', '( lsNormal, lsFocused, lsSelected )';
8204:   TafCLVOptions', 'set of ( loCanSelect, loSelectByShift, loDrawFo'
8205:   + 'cusSelect, loThumbTracking, loScrollToRowCursor, loScrollToColCursor, loSh'
8206:   + 'owLineCursor, loShowCaretCursor, loTabs )';
8207:   TafCLVLeftSpace', 'Integer';
8208:   TafCLVMaxLineLength', 'Integer';
8209:   TafCLVCaretBlinkTime', 'Integer';
8210:   TafCLVCursorEvent', 'Procedure ( Sender : TObject; CursorPos : TPoint)';
8211:   TafCLVDrawLeftSpEvent', 'Procedure ( Sender : TObject; const Lin'
8212:   + 'e, LeftCharPos : Integer; Rect : TRect; State : TafCLVLineState)';
8213:   TafCLVGetTextEvent', 'Procedure ( Sender : TObject; Line : Integ'
8214:   + 'er; var Text: String; var ColorMode: TafCLVColorMode; var CharColors: TafCLVCharColors)';
8215:   SIRegister_TafCustomLineViewer(CL);
8216:   SIRegister_TafLineViewer(CL);
8217:   TafTRMCharColor', 'Integer';
8218:   TafTRMCharAttr', 'record FColor: TafTRMCharColor; BColor: TafTRMCharColor; end';
8219:   //PafTRMCharAttrs', '^TafTRMCharAttrs // will not work';
8220:   TafTRMColorMode', '( cmLDefault, cmL16_16, cmC16_16 )';
8221:   TafTRMLogging', '( lgOff, lgCreate, lgAppend )';
8222:   TafTRMBkSpcMode', '( bmBack, bmBackDel )';
8223:   TafTRMLogSize', 'Integer';
8224:   TafTRMLogFlushTime', 'Integer';
8225:   TafTRMGetColorsEvent', 'Procedure ( Sender : TObject; Line : Int'
8226:   + 'eger; var Colors : TafTRMCharAttrs)';
8227:   TafTRMLineEvent', 'Procedure ( Sender : TObject; Line : Integer)';
8228:   TafTRMProcessCharEvent', 'Procedure (Sender: TObject; var C: Char)';
8229:   TafTRMScrBckBufChange', 'Procedure ( Sender : TObject; BufferSize: Integer)';
8230:   CL.AddClassN(CL.FindClass('TOBJECT'),'TafCustomTerminal');
8231:   SIRegister_TafFileStream(CL);
8232:   SIRegister_TafTerminalBuffer(CL);
8233:   SIRegister_TafCustomTerminal(CL);
8234:   SIRegister_TafTerminal(CL);
8235:   TafCVFSScanStep', 'Integer';
8236:   SIRegister_TafCustomFileViewer(CL);
8237:   SIRegister_TafFileViewer(CL);
8238:   //CL.AddDelphiFunction('Procedure Register');
8239: end;
8240:

```

```

8241: procedure SIRegister_ALFBXClient(CL: TPSPascalCompiler);
8242: begin
8243:   TALFBXClientSQLParam', 'record Value : AnsiString; IsNull : Boolean; end');
8244:   TALFBXClientSQLParams', 'array of TALFBXClientSQLParam');
8245:   TALFBXClientSelectDataSQL', 'record SQL : AnsiString; Params : T
8246:   + 'ALFBXClientSQLParams; RowTag : AnsiString; ViewTag : AnsiString; Skip : in
8247:   + 'teger; First : Integer; CacheThreshold : Integer; end');
8248:   TALFBXClientSelectDataSQLs', 'array of TALFBXClientSelectDataSQL');
8249:   TALFBXClientUpdateDataSQL', 'record SQL : AnsiString; Params : TALFBXClientSQLParams; end');
8250:   TALFBXClientUpdateDataSQLs', 'array of TALFBXClientUpdateDataSQL');
8251:   TALFBXClientMonitoringIOStats', 'record page_reads : int64; page'
8252:   + 'writes : int64; page_fetches : int64; page_marks : int64; end');
8253:   TALFBXClientMonitoringRecordStats', 'record record_idx_reads : i
8254:   + 'nt64; record_seq_reads : int64; record_inserts : int64; record_updates : i
8255:   + 'nt64; record_deletes : int64; record_backouts : int64; record_purges : int
8256:   + '64; record_expunges : int64; end');
8257:   TALFBXClientMonitoringMemoryUsage', 'record memory_used : int64;
8258:   + 'memory_allocated : int64; max_memory_used: int64; max_memory_allocated: int64; end');
8259:   SIRegister_TALFBXClient(CL);
8260:   SIRegister_TALFBXConnectionStatementPoolBinTreeNode(CL);
8261:   SIRegister_TALFBXConnectionStatementPoolBinTree(CL);
8262:   SIRegister_TALFBXConnectionWithStmtPoolContainer(CL);
8263:   SIRegister_TALFBXConnectionWithoutStmtPoolContainer(CL);
8264:   SIRegister_TALFBXReadTransactionPoolContainer(CL);
8265:   SIRegister_TALFBXReadStatementPoolContainer(CL);
8266:   SIRegister_TALFBXStringKeyPoolBinTreeNode(CL);
8267:   SIRegister_TALFBXConnectionPoolClient(CL);
8268:   SIRegister_TALFBXEventThread(CL);
8269: end;
8270:
8271: procedure SIRegister_ALFBXLib(CL: TPSPascalCompiler);
8272: begin
8273:   TALFBXFieldType', '( uftUnknown, uftNumeric, uftChar, uftVarchar'
8274:   + ', uftCString, uftSmallint, uftInteger, uftQuad, uftFloat, uftDoublePrecisi
8275:   + 'on, uftTimestamp, uftBlob, uftBlobId, uftDate, uftTime, uftInt64, uftArray, uftNull )');
8276:   TALFBXScale', 'Integer');
8277:   CL.AddClassN(CL.FindClass('TOBJECT'), 'EALFBXConvertError');
8278:   SIRegister_EALFBXError(CL);
8279:   SIRegister_EALFBXException(CL);
8280:   CL.AddClassN(CL.FindClass('TOBJECT'), 'EALFBXGFixError');
8281:   CL.AddClassN(CL.FindClass('TOBJECT'), 'EALFBXDSQLError');
8282:   CL.AddClassN(CL.FindClass('TOBJECT'), 'EALFBXDynError');
8283:   CL.AddClassN(CL.FindClass('TOBJECT'), 'EALFBXGBakError');
8284:   CL.AddClassN(CL.FindClass('TOBJECT'), 'EALFBXGSecError');
8285:   CL.AddClassN(CL.FindClass('TOBJECT'), 'EALFBXLicenseError');
8286:   CL.AddClassN(CL.FindClass('TOBJECT'), 'EALFBXGStatError');
8287:   //EALFBXExceptionClass', 'class of EALFBXError');
8288:   TALFBXCharacterSet', '( csNONE, csASCII, csBIG_5, csCYRL, csDOS4'
8289:   + '37, csDOS850, csDOS852, csDOS857, csDOS860, csDOS861, csDOS863, csDOS865, '
8290:   + 'csEUCJ_0208, csGB_2312, csISO8859_1, csISO8859_2, csKSC_5601, csNEXT, csOC'
8291:   + 'TETS, csSJIS_0208, csUNICODE_FSS, csUTF8, csWIN1250, csWIN1251, csWIN1252, '
8292:   + 'csWIN1253, csWIN1254, csDOS737, csDOS775, csDOS858, csDOS862, csDOS864, c'
8293:   + 'sDOS866, csDOS869, csWIN1255, csWIN1256, csWIN1257, csISO8859_3, csISO8859'
8294:   + '_4, csISO8859_5, csISO8859_6, csISO8859_7, csISO8859_8, csISO8859_9, csISO'
8295:   + '8859_13, csKOI8R, csKOI8U, csWIN1258, csTIS620, csGBK, csCP943C )');
8296:   TALFBXTransParam', '( tpConsistency, tpConcurrency, tpShared, tp'
8297:   + 'Protected, tpExclusive, tpWait, tpNowait, tpRead, tpWrite, tpLockRead, tpL'
8298:   + 'ockWrite, tpVerbTime, tpCommitTime, tpIgnoreLimbo, tpReadCommitted, tpAuto'
8299:   + 'Commit, tpRecVersion, tpNoRecVersion, tpRestartRequests, tpNoAutoUndo, tpLockTimeout )');
8300:   TALFBXTransParams', 'set of TALFBXTransParam');
8301:   CL.AddDelphiFunction('Function ALFBXStrToCharacterSet( const CharacterSet : AnsiString) :
TALFBXCharacterSet');
8302:   Function ALFBXCreateDBParams( Params : AnsiString; Delimiter : Char) : AnsiString');
8303:   Function ALFBXCreateBlobParams( Params : AnsiString; Delimiter : Char) : AnsiString');
8304:   cALFBXMaxParamLength', 'LongInt').SetInt( 125);
8305:   TALFBXParamsFlag', '( pfNotInitialized, pfNotNullable )');
8306:   TALFBXParamsFlags', 'set of TALFBXParamsFlag');
8307:   //PALFBXSQLVar', '^TALFBXSQLVar // will not work');
8308:   //PALFBXSQLDaData', '^TALFBXSQLDaData // will not work');
8309:   TALFBXStatementType', '( stSelect, stInsert, stUpdate, stDelete, '
8310:   + 'stDDL, stGetSegment, stPutSegment, stExecProcedure, stStartTrans, stComm'
8311:   + 'it, stRollback, stSelectForUpdate, stSetGenerator, stSavePoint )');
8312:   SIRegister_TALFBXSQLDA(CL);
8313:   //PALFBXPtrArray', '^TALFBXPtrArray // will not work');
8314:   SIRegister_TALFBXPoolStream(CL);
8315:   //PALFBXBlobData', '^TALFBXBlobData // will not work');
8316:   TALFBXBlobData', 'record Size : Integer; Buffer : string; end');
8317:   //PALFBXArrayDesc', '^TALFBXArrayDesc // will not work');
8318:   //TALFBXArrayDesc', 'TISCArrDesc');
8319:   //TALFBXBlobDesc', 'TISCBlobDesc');
8320:   //PALFBXArrayInfo', '^TALFBXArrayInfo // will not work');
8321:   //TALFBXArrayInfo', 'record index : Integer; size : integer; info: TALFBXArrayDesc; end');
8322:   SIRegister_TALFBXSQLResult(CL);
8323:   //TALFBXSQLResultClass', 'class of TALFBXSQLResult');
8324:   SIRegister_TALFBXSQLParams(CL);
8325:   //TALFBXSQLParamsClass', 'class of TALFBXSQLParams');
8326:   TALFBXDSQLInfoData', 'record InfoCode : byte; InfoLen : Word; St'
8327:   + 'atementType : TALFBXStatementType; end');
8328:   CL.AddClassN(CL.FindClass('TOBJECT'), 'TALFBXLibrary');

```

```

8329: //PALFBXStatusVector', '^TALFBXStatusVector // will not work');
8330: TALFBXOnConnectionLost', 'Procedure ( Lib : TALFBXLibrary)');
8331: //TALFBXOnGetDBExceptionClass', 'Procedure ( Number : Integer; out'
8332: //+' Excep : EALFBXExceptionClass)');
8333: SIRegister_TALFBXLibrary(CL);
8334: cAlFBXDateOffset', 'LongInt').SetInt( 15018);
8335: cAlFBXTimeCoeff', 'LongInt').SetInt( 864000000);
8336: TALFBXParamType', '( prNone, prByte, prShrt, prCard, prStrg, prIgno )');
8337: TALFBXDPBInfo', 'record Name : AnsiString; ParamType : TALFBXParamType; end');
8338: Function ALFBXSQLQuote( const name : AnsiString) : AnsiString);
8339: Function ALFBXSQLUnQuote( const name : AnsiString) : AnsiString);
8340: end;
8341:
8342: procedure SIRegister_AlMySqlClient(CL: TPSPascalCompiler);
8343: begin
8344:   SIRegister_EALMySqlError(CL);
8345:
8346:   uLongLong', 'LongLong');
8347: cAlMySql_INVALIDLIBVERSION', 'String').SetString( 'Incorrect Database Server Version.');
```

8348: cAlMySql_CANTLOADLIB', 'String').SetString('Can't load library: %s.');

8349: CR_UNKNOWN_ERROR', 'LongInt').SetInt(2000);

8350: CR_SOCKET_CREATE_ERROR', 'LongInt').SetInt(2001);

8351: CR_CONNECTION_ERROR', 'LongInt').SetInt(2002);

8352: CR_CONN_HOST_ERROR', 'LongInt').SetInt(2003);

8353: CR_IPSOCK_ERROR', 'LongInt').SetInt(2004);

8354: CR_UNKNOWN_HOST', 'LongInt').SetInt(2005);

8355: CR_SERVER_GONE_ERROR', 'LongInt').SetInt(2006);

8356: CR_VERSION_ERROR', 'LongInt').SetInt(2007);

8357: CR_OUT_OF_MEMORY', 'LongInt').SetInt(2008);

8358: CR_WRONG_HOST_INFO', 'LongInt').SetInt(2009);

8359: CR_LOCALHOST_CONNECTION', 'LongInt').SetInt(2010);

8360: CR_TCP_CONNECTION', 'LongInt').SetInt(2011);

8361: CR_SERVER_HANDSHAKE_ERR', 'LongInt').SetInt(2012);

8362: CR_SERVER_LOST', 'LongInt').SetInt(2013);

8363: CR_COMMANDS_OUT_OF_SYNC', 'LongInt').SetInt(2014);

8364: CR_NAMEDPIPE_CONNECTION', 'LongInt').SetInt(2015);

8365: CR_NAMEDPIPEWAIT_ERROR', 'LongInt').SetInt(2016);

8366: CR_NAMEDPIPEOPEN_ERROR', 'LongInt').SetInt(2017);

8367: CR_NAMEDPIPESETSTATE_ERROR', 'LongInt').SetInt(2018);

8368: CR_CANT_READ_CHARSET', 'LongInt').SetInt(2019);

8369: CR_NET_PACKET_TOO_LARGE', 'LongInt').SetInt(2020);

8370: CR_EMBEDDED_CONNECTION', 'LongInt').SetInt(2021);

8371: CR_PROBE_SLAVE_STATUS', 'LongInt').SetInt(2022);

8372: CR_PROBE_SLAVE_HOSTS', 'LongInt').SetInt(2023);

8373: CR_PROBE_SLAVE_CONNECT', 'LongInt').SetInt(2024);

8374: CR_PROBE_MASTER_CONNECT', 'LongInt').SetInt(2025);

8375: CR_SSL_CONNECTION_ERROR', 'LongInt').SetInt(2026);

8376: CR_MALFORMED_PACKET', 'LongInt').SetInt(2027);

8377: CR_WRONG_LICENSE', 'LongInt').SetInt(2028);

8378: CR_NULL_POINTER', 'LongInt').SetInt(2029);

8379: CR_NO_PREPARE_STMT', 'LongInt').SetInt(2030);

8380: CR_PARAMS_NOT_BOUND', 'LongInt').SetInt(2031);

8381: CR_DATA_TRUNCATED', 'LongInt').SetInt(2032);

8382: CR_NO_PARAMETERS_EXISTS', 'LongInt').SetInt(2033);

8383: CR_INVALID_PARAMETER_NO', 'LongInt').SetInt(2034);

8384: CR_INVALID_BUFFER_USE', 'LongInt').SetInt(2035);

8385: CR_UNSUPPORTED_PARAM_TYPE', 'LongInt').SetInt(2036);

8386: CR_SHARED_MEMORY_CONNECTION', 'LongInt').SetInt(2037);

8387: CR_SHARED_MEMORY_CONNECT_REQUEST_ERROR', 'LongInt').SetInt(2038);

8388: CR_SHARED_MEMORY_CONNECT_ANSWER_ERROR', 'LongInt').SetInt(2039);

8389: CR_SHARED_MEMORY_CONNECT_FILE_MAP_ERROR', 'LongInt').SetInt(2040);

8390: CR_SHARED_MEMORY_CONNECT_MAP_ERROR', 'LongInt').SetInt(2041);

8391: CR_SHARED_MEMORY_FILE_MAP_ERROR', 'LongInt').SetInt(2042);

8392: CR_SHARED_MEMORY_MAP_ERROR', 'LongInt').SetInt(2043);

8393: CR_SHARED_MEMORY_EVENT_ERROR', 'LongInt').SetInt(2044);

8394: CR_SHARED_MEMORY_CONNECT_ABANDONED_ERROR', 'LongInt').SetInt(2045);

8395: CR_SHARED_MEMORY_CONNECT_SET_ERROR', 'LongInt').SetInt(2046);

8396: CR_CONN_UNKNOWN_PROTOCOL', 'LongInt').SetInt(2047);

8397: CR_INVALID_CONN_HANDLE', 'LongInt').SetInt(2048);

8398: CR_SECURE_AUTH', 'LongInt').SetInt(2049);

8399: CR_FETCH_CANCELED', 'LongInt').SetInt(2050);

8400: CR_NO_DATA', 'LongInt').SetInt(2051);

8401: CR_NO_STMT_METADATA', 'LongInt').SetInt(2052);

8402: CR_NO_RESULT_SET', 'LongInt').SetInt(2053);

8403: CR_NOT_IMPLEMENTED', 'LongInt').SetInt(2054);

8404: CR_SERVER_LOST_EXTENDED', 'LongInt').SetInt(2055);

8405: CR_STMT_CLOSED', 'LongInt').SetInt(2056);

8406: CR_NEW_STMT_METADATA', 'LongInt').SetInt(2057);

8407: CR_ALREADY_CONNECTED', 'LongInt').SetInt(2058);

8408: CR_AUTH_PLUGIN_CANNOT_LOAD', 'LongInt').SetInt(2059);

8409: MYSQL_ERRMSG_SIZE', 'LongInt').SetInt(512);

8410: SQLSTATE_LENGTH', 'LongInt').SetInt(5);

8411: MYSQL_PORT', 'LongInt').SetInt(3306);

8412: LOCAL_HOST', 'String').SetString('localhost');

8413: NOT_NULL_FLAG', 'LongInt').SetInt(1);

8414: PRI_KEY_FLAG', 'LongInt').SetInt(2);

8415: UNIQUE_KEY_FLAG', 'LongInt').SetInt(4);

8416: MULTIPLE_KEY_FLAG', 'LongInt').SetInt(8);

8417: BLOB_FLAG', 'LongInt').SetInt(16);

```

8418: UNSIGNED_FLAG', 'LongInt').SetInt( 32);
8419: ZEROFILL_FLAG', 'LongInt').SetInt( 64);
8420: BINARY_FLAG', 'LongInt').SetInt( 128);
8421: ENUM_FLAG', 'LongInt').SetInt( 256);
8422: AUTO_INCREMENT_FLAG', 'LongInt').SetInt( 512);
8423: TIMESTAMP_FLAG', 'LongInt').SetInt( 1024);
8424: SET_FLAG', 'LongInt').SetInt( 2048);
8425: NUM_FLAG', 'LongInt').SetInt( 32768);
8426: PART_KEY_FLAG', 'LongInt').SetInt( 16384);
8427: GROUP_FLAG', 'LongInt').SetInt( 32768);
8428: UNIQUE_FLAG', 'LongInt').SetInt( 65536);
8429: BINCOMP_FLAG', 'LongWord').SetUInt( $20000);
8430: GET_FIXED_FIELDS_FLAG', 'LongWord').SetUInt( $40000);
8431: FIELD_IN_PART_FUNC_FLAG', 'LongWord').SetUInt( $80000);
8432: FIELD_IN_ADD_INDEX', 'LongWord').SetUInt( $100000);
8433: FIELD_IS_RENAMED', 'LongWord').SetUInt( $200000);
8434: _CLIENT_LONG_PASSWORD', 'LongInt').SetInt( 1);
8435: _CLIENT_FOUND_ROWS', 'LongInt').SetInt( 2);
8436: _CLIENT_LONG_FLAG', 'LongInt').SetInt( 4);
8437: _CLIENT_CONNECT_WITH_DB', 'LongInt').SetInt( 8);
8438: _CLIENT_NO_SCHEMA', 'LongInt').SetInt( 16);
8439: _CLIENT_COMPRESS', 'LongInt').SetInt( 32);
8440: _CLIENT_ODBC', 'LongInt').SetInt( 64);
8441: _CLIENT_LOCAL_FILES', 'LongInt').SetInt( 128);
8442: _CLIENT_IGNORE_SPACE', 'LongInt').SetInt( 256);
8443: _CLIENT_CHANGE_USER', 'LongInt').SetInt( 512);
8444: _CLIENT_INTERACTIVE', 'LongInt').SetInt( 1024);
8445: _CLIENT_SSL', 'LongInt').SetInt( 2048);
8446: _CLIENT_IGNORE_SIGPIPE', 'LongInt').SetInt( 4096);
8447: _CLIENT_TRANSACTIONS', 'LongInt').SetInt( 8196);
8448: _CLIENT_RESERVED', 'LongInt').SetInt( 16384);
8449: _CLIENT_SECURE_CONNECTION', 'LongInt').SetInt( 32768);
8450: _CLIENT_MULTI_STATEMENTS', 'LongInt').SetInt( 65536);
8451: _CLIENT_MULTI_RESULTS', 'LongInt').SetInt( 131072);
8452: _CLIENT_PS_MULTI_RESULTS', 'LongInt').SetInt( 262144);
8453: _CLIENT_PLUGIN_AUTH', 'LongInt').SetInt( 524288);
8454: _CLIENT_SSL_VERIFY_SERVER_CERT', 'LongInt').SetInt( 1073741824);
8455: _CLIENT_REMEMBER_OPTIONS', 'LongWord').SetUInt( 2147483648);
8456: MYSQL_SHUTDOWN_KILLABLE_CONNECT', 'LongInt').SetInt( 1);
8457: MYSQL_SHUTDOWN_KILLABLE_TRANS', 'LongInt').SetInt( 2);
8458: MYSQL_SHUTDOWN_KILLABLE_LOCK_TABLE', 'LongInt').SetInt( 4);
8459: MYSQL_SHUTDOWN_KILLABLE_UPDATE', 'LongInt').SetInt( 8);
8460: STMT_FETCH_OK', 'LongInt').SetInt( 0);
8461: STMT_FETCH_ERROR', 'LongInt').SetInt( 1);
8462: STMT_FETCH_NO_DATA', 'LongInt').SetInt( 100);
8463: STMT_FETCH_DATA_TRUNC', 'LongInt').SetInt( 101);
8464:
8465: TMySQLClientOptions', '( CLIENT_LONG_PASSWORD, CLIENT_FOUND_RO'
8466: +WS, CLIENT_LONG_FLAG, CLIENT_CONNECT_WITH_DB, CLIENT_NO_SCHEMA, CLIENT_COM'
8467: +PRESS, CLIENT_ODBC, CLIENT_LOCAL_FILES, CLIENT_IGNORE_SPACE, CLIENT_CHANGE'
8468: +'_USER, CLIENT_INTERACTIVE, CLIENT_SSL, CLIENT_IGNORE_SIGPIPE, CLIENT_TRANS'
8469: +'ACTIONS, CLIENT_RESERVED, CLIENT_SECURE_CONNECTION, CLIENT_MULTI_STATEMENT'
8470: +'S, CLIENT_MULTI_RESULTS, CLIENT_PS_MULTI_RESULTS, CLIENT_PLUGIN_AUTH, CLIE'
8471: +'NT_OPT_20, CLIENT_OPT_21, CLIENT_OPT_22, CLIENT_OPT_23, CLIENT_OPT_24, CLI'
8472: +'ENT_OPT_25, CLIENT_OPT_26, CLIENT_OPT_27, CLIENT_OPT_28, CLIENT_OPT_29, CL'
8473: +'IENT_SSL_VERIFY_SERVER_CERT, CLIENT_REMEMBER_OPTIONS ');
8474: MYSQL_TIME', record year : UInt; month : UInt; day : UInt; hour'
8475: +': UInt; minute : UInt; second : UInt; second_part : UInt; neg : Byte; end');
8476:
8477: TMySQLOption', '( MYSQL_OPT_CONNECT_TIMEOUT, MYSQL_OPT_COMPRESS,'
8478: +MYSQL_OPT_NAMED_PIPE, MYSQL_INIT_COMMAND, MYSQL_READ_DEFAULT_FILE, MYSQL_'
8479: +READ_DEFAULT_GROUP, MYSQL_SET_CHARSET_DIR, MYSQL_SET_CHARSET_NAME, MYSQL_O'
8480: +PT_LOCAL_INFILE, MYSQL_OPT_PROTOCOL, MYSQL_SHARED_MEMORY_BASE_NAME, MYSQL_'
8481: +OPT_READ_TIMEOUT, MYSQL_OPT_WRITE_TIMEOUT, MYSQL_OPT_USE_RESULT, MYSQL_OPT'
8482: +'_USE_REMOTE_CONNECTION, MYSQL_OPT_USE_EMBEDDED_CONNECTION, MYSQL_OPT_GUESS'
8483: +'_CONNECTION, MYSQL_SET_CLIENT_IP, MYSQL_SECURE_AUTH, MYSQL_REPORT_DATA_TRU'
8484: +'_NCATION, MYSQL_OPT_RECONNECT, MYSQL_OPT_SSL_VERIFY_SERVER_CERT, MYSQL_PLUG'
8485: +'_IN_DIR, MYSQL_DEFAULT_AUTH ');
8486:
8487: TalMySQLClientSelectDataSQL', 'record SQL : AnsiString; RowTag : '
8488: +AnsiString; ViewTag: AnsiString; Skip: integer; First: Integer; CacheThreshold: Integer; end');
8489: TalMySQLClientSelectDataSQLs', 'array of TalMySQLClientSelectDataSQL';
8490: TalMySQLClientUpdateDataSQL', 'record SQL : AnsiString; end');
8491: TalMySQLClientUpdateDataSQLs', 'array of TalMySQLClientUpdateDataSQL';
8492: TALMySQLOption', record Option : TMySQLOption; Value : PChar; end');
8493: TALMySQLOptions', array of TALMySQLOption';
8494: SIRegister_TalMySQLClient(CL);
8495: SIRegister_TalMySQLConnectionPoolContainer(CL);
8496: SIRegister_TalMySQLConnectionPoolClient(CL);
8497: Function AlMySQLClientSlashedStr( const Str : AnsiString) : AnsiString';
8498: end;
8499:
8500: procedure SIRegister_ovcBidi(CL: TPSPascalCompiler);
8501: begin
8502: CL.AddConstantN('WS_EX_RIGHT', 'LongWord').SetUInt( $00001000);
8503: CL.AddConstantN('WS_EX_LEFT', 'LongWord').SetUInt( $00000000);
8504: 'WS_EX_RTLREADING', 'LongWord').SetUInt( $00002000);
8505: 'WS_EX_LTRREADING', 'LongWord').SetUInt( $00000000);
8506: 'WS_EX_LEFTSCROLLBAR', 'LongWord').SetUInt( $00004000);

```



```

8507: 'WS_EX_RIGHTSCROLLBAR', 'LongWord').SetUInt( $00000000);
8508: Function SetProcessDefaultLayout( dwDefaultLayout : DWORD) : BOOL);
8509: 'LAYOUT_RTL', 'LongWord').SetUInt( $00000001);
8510: 'LAYOUT_BT', 'LongWord').SetUInt( $00000002);
8511: 'LAYOUT_VBH', 'LongWord').SetUInt( $00000004);
8512: 'LAYOUT_BITMAPORIENTATIONPRESERVED', 'LongWord').SetUInt( $00000008);
8513: 'NOMIRRORBITMAP', 'LongWord').SetUInt( DWORD ( $80000000 ));
8514: Function SetLayout( dc : HDC; dwLayout : DWORD) : DWORD);
8515: Function GetLayout( dc : hdc) : DWORD);
8516: Function IsBidi : Boolean);
8517: end;
8518:
8519: procedure SIRegister_ALMultiPartBaseParser(CL: TPSPascalCompiler);
8520: begin
8521:   SIRegister_TALMultiPartBaseContent(CL);
8522:   SIRegister_TALMultiPartBaseContents(CL);
8523:   SIRegister_TALMultiPartBaseStream(CL);
8524:   SIRegister_TALMultiPartBaseEncoder(CL);
8525:   SIRegister_TALMultiPartBaseDecoder(CL);
8526: end;
8527:
8528: procedure SIRegister_ovccalc(CL: TPSPascalCompiler);
8529: begin
8530:   TOvcCalculatorButton', '( cbNone, cbTape, cbBack, cbClearEntry, '
8531:   + 'cbClear, cbAdd, cbSub, cbMul, cbDiv, cb0, cb1, cb2, cb3, cb4, cb5, cb6, cb'
8532:   + '7, cb8, cb9, cbDecimal, cbEqual, cbInvert, cbChangeSign, cbPercent, cbSqrt'
8533:   + ', cbMemClear, cbMemRecall, cbMemStore, cbMemAdd, cbMemSub, cbSubTotal )');
8534:   TOvcButtonInfo', 'record Position : TRect; Caption : string; Vis'
8535:   + 'ible : Boolean; end);
8536:   TOvcCalculatorOperation', '( coNone, coAdd, coSub, coMul, coDiv, '
8537:   + 'coEqual, coInvert, coPercent, coSqrt, coMemClear, coMemRecall, coMemStore'
8538:   + ', coMemAdd, coMemSub, coSubTotal )');
8539:   TOvcCalcState', '( csValid, csLocked, csClear )');
8540:   TOvcCalcStates', 'set of TOvcCalcState');
8541:   TOvcCalcColorScheme', '( cscalcCustom, cscalcWindows, cscalcDark'
8542:   + ', cscalcOcean, cscalcPlain )');
8543:   SIRegister_TOvcCalcColors(CL);
8544:   SIRegister_TOvcCalcPanel(CL);
8545:   SIRegister_TOvcCustomCalculatorEngine(CL);
8546:   SIRegister_TOvcCalcTape(CL);
8547:   TOvcCalcButtonPressedEvent', 'Procedure ( Sender : TObject; Butt'
8548:   + 'on : TOvcCalculatorButton)');
8549:   TOvcCalculatorOption', '( coShowItemCount, coShowMemoryButtons, '
8550:   + 'coShowClearTapeButton, coShowTape, coShowSeparatePercent )');
8551:   TOvcCalculatorOptions', 'set of TOvcCalculatorOption');
8552:   SIRegister_TOvcCustomCalculator(CL);
8553:   SIRegister_TOvcCalculator(CL);
8554: end;
8555:
8556: procedure SIRegister_MakeApp(CL: TPSPascalCompiler);
8557: begin
8558:   'aZero', 'LongInt').SetInt( 0);
8559:   'makeappDEF', 'LongInt').SetInt( - 1);
8560:   'CS_VREDRAW', 'LongInt').SetInt( DWORD ( 1 ));
8561:   'CS_HREDRAW', 'LongInt').SetInt( DWORD ( 2 ));
8562:   'CS_KEYCVTWINDOW', 'LongInt').SetInt( 4);
8563:   'CS_DBLCLKS', 'LongInt').SetInt( 8);
8564:   'CS_OWIND', 'LongWord').SetUInt( $20);
8565:   'CS_CLASSDC', 'LongWord').SetUInt( $40);
8566:   'CS_PARENTDC', 'LongWord').SetUInt( $80);
8567:   'CS_NOKEYCVT', 'LongWord').SetUInt( $100);
8568:   'CS_NOCLOSE', 'LongWord').SetUInt( $200);
8569:   'CS_SAVEBITS', 'LongWord').SetUInt( $800);
8570:   'CS_BYTEALIGNCLIENT', 'LongWord').SetUInt( $1000);
8571:   'CS_BYTEALIGNWINDOW', 'LongWord').SetUInt( $2000);
8572:   'CS_GLOBALCLASS', 'LongWord').SetUInt( $4000);
8573:   'CS_IME', 'LongWord').SetUInt( $10000);
8574:   'CS_DROPSHADOW', 'LongWord').SetUInt( $20000);
8575:   //PPanelFunc', '^TPanelFunc // will not work');
8576:   TPanelFunc', 'function(iMsg, wParam, lParam: Integer): Integer); // will not work');
8577:
8578:   // TPanelFunc = function(iMsg, wParam, lParam: Integer): Integer;
8579:   TMessageFunc', 'function(hWnd, iMsg, wParam, lParam: Integer): Integer)');
8580: Function SetWinClass( const ClassName : String; pMessFunc : TMessagefunc; wcStyle : Integer) : Word);
8581: Function SetWinClassO( const ClassName : String; pMessFunc : TObject; wcStyle : Integer) : Word);
8582: TPanelStyle', '(psEdge, psTabEdge, psBorder, psTabBorder, psTab, psNone )');
8583: TFontLook', '( flBold, flItalic, flUnderLine, flStrikeOut )');
8584: TFontLooks', 'set of TFontLook');
8585:
8586: end;
8587:
8588:
8589: procedure SIRegister_O32MouseMon(CL: TPSPascalCompiler);
8590: begin
8591:   TMouseMonHandler', 'Procedure ( const MouseMessage, wParam, lPar'
8592:   + 'am : Integer; const ScreenPt : TPoint; const MouseWnd : TOvcHWnd)');
8593:   Procedure StartMouseMonitor( Callback : TMouseMonHandler)');
8594:   Procedure StopMouseMonitor( Callback : TMouseMonHandler)');
8595: end;

```

```

8596:
8597: procedure SIRegister_ScreenSaver(CL: TPSPascalCompiler);
8598: begin
8599:   TScreenSaverOption, '( ssoAutoAdjustFormProperties, ssoAutoHook'
8600:     + 'KeyboardEvents, ssoAutoHookMouseEvents, ssoEnhancedMouseMoveDetection )';
8601:   TScreenSaverOptions, 'set of TScreenSaverOption';
8602:   'cDefaultScreenSaverOptions', 'LongInt').Value.ts32:= ord(ssoAutoAdjustFormProperties) or
ord(ssoAutoHookKeyboardEvents) or ord(ssoEnhancedMouseMoveDetection);
8603:   TScreenSaverPreviewEvent, 'Procedure ( Sender : TObject; previewHwnd: HWND)';
8604:   SIRegister_TScreenSaver(CL);
8605:   Procedure SetScreenSaverPassword';
8606: end;
8607:
8608: procedure SIRegister_VectorLists(CL: TPSPascalCompiler);
8609: begin
8610:   TBaseListOption, '( bloExternalMemory, bloSetCountResetsMemory)';
8611:   TBaseListOptions, 'set of TBaseListOption';
8612:   SIRegister_TBaseList(CL);
8613:   SIRegister_TBaseVectorList(CL);
8614:   SIRegister_TAffineVectorList(CL);
8615:   SIRegister_TVectorList(CL);
8616:   SIRegister_TTexPointList(CL);
8617:   SIRegister_TXIntegerList(CL);
8618:   //PSingleArrayList, 'TSingleArrayList // will not work';
8619:   SIRegister_TSingleList(CL);
8620:   SIRegister_TByteList(CL);
8621:   SIRegister_TQuaternionList(CL);
8622:   Procedure QuickSortLists( startIndex, endIndex : Integer; refList : TSingleList; objList : TList);';
8623:   Procedure QuickSortLists1( startIndex, endIndex : Integer; refList : TSingleList; objList : TBaseList);';
8624:   Procedure FastQuickSortLists( startIndex, endIndex : Integer; refList : TSingleList; objList :
TPersistentObjectList);';
8625: end;');';
8626: end;
8627:
8628: procedure SIRegister_JclBorlandTools(CL: TPSPascalCompiler);
8629: begin
8630:   CL.AddClassN(CL.FindClass('TOBJECT'), 'EJclBorRADException');
8631:   TJclBorRADToolKind, '( brDelphi, brCppBuilder, brBorlandDevStudio )';
8632:   TJclBorRADToolEdition, '( deOPEN, dePRO, deSVR )';
8633:   TJclBorRADToolEdition, '( deSTD, dePRO, deCSS, deARC )';
8634:   TJclBorRADToolPath, 'string';
8635:   'SupportedDelphiVersions', 'LongInt').SetInt( 5 or 6 or 7 or 8 or 9 or 10 or 11);
8636:   'SupportedBCBVersions', 'LongInt').SetInt( 5 or 6 or 10 or 11);
8637:   'SupportedBDSVersions', 'LongInt').SetInt( 1 or 2 or 3 or 4 or 5);
8638:   'BorRADToolRepositoryPagesSection', 'String').SetString( 'Repository Pages');
8639:   'BorRADToolRepositoryDialogsPage', 'String').SetString( 'Dialogs');
8640:   'BorRADToolRepositoryFormsPage', 'String').SetString( 'Forms');
8641:   'BorRADToolRepositoryProjectsPage', 'String').SetString( 'Projects');
8642:   'BorRADToolRepositoryDataModulesPage', 'String').SetString( 'Data Modules');
8643:   'BorRADToolRepositoryObjectType', 'String').SetString( 'Type');
8644:   'BorRADToolRepositoryFormTemplate', 'String').SetString( 'FormTemplate');
8645:   'BorRADToolRepositoryProjectTemplate', 'String').SetString( 'ProjectTemplate');
8646:   'BorRADToolRepositoryObjectName', 'String').SetString( 'Name');
8647:   'BorRADToolRepositoryObjectPage', 'String').SetString( 'Page');
8648:   'BorRADToolRepositoryObjectIcon', 'String').SetString( 'Icon');
8649:   'BorRADToolRepositoryObjectDescr', 'String').SetString( 'Description');
8650:   'BorRADToolRepositoryObjectAuthor', 'String').SetString( 'Author');
8651:   'BorRADToolRepositoryObjectAncestor', 'String').SetString( 'Ancestor');
8652:   'BorRADToolRepositoryObjectDesigner', 'String').SetString( 'Designer');
8653:   'BorRADToolRepositoryDesignerDfm', 'String').SetString( 'dfm');
8654:   'BorRADToolRepositoryDesignerXfm', 'String').SetString( 'xfm');
8655:   'BorRADToolRepositoryObjectNewForm', 'String').SetString( 'DefaultNewForm');
8656:   'BorRADToolRepositoryObjectMainForm', 'String').SetString( 'DefaultMainForm');
8657:   'SourceExtensionDelphiPackage', 'String').SetString( '.dpc');
8658:   'SourceExtensionBCBPackage', 'String').SetString( '.bpc');
8659:   'SourceExtensionDelphiProject', 'String').SetString( '.dpr');
8660:   'SourceExtensionBCBProject', 'String').SetString( '.bpr');
8661:   'SourceExtensionBDSProject', 'String').SetString( '.bdsproj');
8662:   'SourceExtensionDProject', 'String').SetString( '.dproj');
8663:   'BinaryExtensionPackage', 'String').SetString( '.bpl');
8664:   'BinaryExtensionLibrary', 'String').SetString( '.dll');
8665:   'BinaryExtensionExecutable', 'String').SetString( '.exe');
8666:   'CompilerExtensionDCP', 'String').SetString( '.dcp');
8667:   'CompilerExtensionBPI', 'String').SetString( '.bpi');
8668:   'CompilerExtensionLIB', 'String').SetString( '.lib');
8669:   'CompilerExtensionTDS', 'String').SetString( '.tds');
8670:   'CompilerExtensionMAP', 'String').SetString( '.map');
8671:   'CompilerExtensionDRC', 'String').SetString( '.drc');
8672:   'CompilerExtensionDEF', 'String').SetString( '.def');
8673:   'SourceExtensionCPP', 'String').SetString( '.cpp');
8674:   'SourceExtensionH', 'String').SetString( '.h');
8675:   'SourceExtensionPAS', 'String').SetString( '.pas');
8676:   'SourceExtensionDFM', 'String').SetString( '.dfm');
8677:   'SourceExtensionXFM', 'String').SetString( '.xfm');
8678:   'SourceDescriptionPAS', 'String').SetString( 'Pascal source file');
8679:   'SourceDescriptionCPP', 'String').SetString( 'C++ source file');
8680:   'DesignerVCL', 'String').SetString( 'VCL');
8681:   'DesignerCLX', 'String').SetString( 'CLX');
8682:   'ProjectTypePackage', 'String').SetString( 'package');

```

```

8683: 'ProjectTypeLibrary','String').SetString( 'library');
8684: 'ProjectTypeProgram','String').SetString( 'program');
8685: Personality32Bit,'String').SetString( '32 bit');
8686: Personality64Bit,'String').SetString( '64 bit');
8687: PersonalityDelphi,'String').SetString( 'Delphi');
8688: PersonalityDelphiDotNet,'String').SetString( 'Delphi.net');
8689: PersonalityBCB,'String').SetString( 'C++Builder');
8690: PersonalityCSB,'String').SetString( 'C#Builder');
8691: PersonalityVB,'String').SetString( 'Visual Basic');
8692: PersonalityDesign,'String').SetString( 'Design');
8693: PersonalityUnknown,'String').SetString( 'Unknown personality');
8694: PersonalityBDS,'String').SetString( 'Borland Developer Studio');
8695: 'DOFDirectoriesSection','String').SetString( 'Directories');
8696: 'DOFUnitOutputDirKey','String').SetString( 'UnitOutputDir');
8697: 'DOFSearchPathName','String').SetString( 'SearchPath');
8698: 'DOFConditionals','String').SetString( 'Conditionals');
8699: 'DOFLinkerSection','String').SetString( 'Linker');
8700: DOFPackagesKey,'String').SetString( 'Packages');
8701: DOFCompilerSection,'String').SetString( 'Compiler');
8702: DOFPackageNoLinkKey,'String').SetString( 'PackageNoLink');
8703: DOFAdditionalSection,'String').SetString( 'Additional');
8704: DOFOptionsKey,'String').SetString( 'Options');
8705: TJclBorPersonality,'(' bpDelphi32, bpDelphi64, bpBCBuilder32, b'
8706:   + 'pBCBuilder64, bpDelphiNet32, bpDelphiNet64, bpCSBuilder32, bpCSBuilder64, '
8707:   + 'bpVisualBasic32, bpVisualBasic64, bpDesign, bpUnknown )');
8708: TJclBorPersonalities,'set of TJclBorPersonality');
8709: TJclBorDesigner,'(' bdVCL, bdCLX ');
8710: TJclBorDesigners,'set of TJclBorDesigner');
8711: TJclBorPlatform,'(' bp32bit, bp64bit ');
8712: CL.AddClassN(CL.FindClass('TOBJECT'),'TJclBorRADToolInstallation');
8713: SIRegister_TJclBorRADToolInstallationObject(CL);
8714: SIRegister_TJclBorlandOpenHelp(CL);
8715: TJclHelp2Object,'(' hoRegisterSession, hoRegister, hoPlugin ');
8716: TJclHelp2Objects,'set of TJclHelp2Object');
8717: SIRegister_TJclHelp2Manager(CL);
8718: SIRegister_TJclBorRADToolIdeTool(CL);
8719: SIRegister_TJclBorRADToolIdePackages(CL);
8720: SIRegister_IJclCommandLineTool(CL);
8721: CL.AddClassN(CL.FindClass('TOBJECT'),'EJclCommandLineToolError');
8722: SIRegister_TJclCommandLineTool(CL);
8723: SIRegister_TJclBorlandCommandLineTool(CL);
8724: SIRegister_TJclBCC32(CL);
8725: SIRegister_TJclDCC32(CL);
8726: TJclDCC,'TJclDCC32');
8727: SIRegister_TJclBpr2Mak(CL);
8728: SIRegister_TJclBorlandMake(CL);
8729: SIRegister_TJclBorRADToolPalette(CL);
8730: SIRegister_TJclBorRADToolRepository(CL);
8731: TCommandLineTool,'(' clAsm, clBcc32, clDcc32, clDccIL, clMake,clProj2Mak ');
8732: TCommandLineTools,'set of TCommandLineTool');
8733: //TJclBorRADToolInstallationClass', 'class of TJclBorRADToolInstallation');
8734: SIRegister_TJclBorRADToolInstallation(CL);
8735: SIRegister_TJclBCBInstallation(CL);
8736: SIRegister_TJclDelphiInstallation(CL);
8737: SIRegister_TJclDCCIL(CL);
8738: SIRegister_TJclBDSInstallation(CL);
8739: TTraverseMethod,'Function ( Installation : TJclBorRADToolInstallation) : Boolean');
8740: SIRegister_TJclBorRADToolInstallations(CL);
8741: Function BPLFileName( const BPLPath, PackageFileName : string) : string);
8742: Function BinaryFileName( const OutputPath, ProjectFileName : string) : string);
8743: Function IsDelphiPackage( const FileName : string) : Boolean);
8744: Function IsDelphiProject( const FileName : string) : Boolean);
8745: Function IsBCBPackage( const FileName : string) : Boolean);
8746: Function IsBCBProject( const FileName : string) : Boolean);
8747: Procedure GetDPRFileInfo(const DPRFileName: string; out BinaryExtension : string; const LibSuffix :
PString);
8748: Procedure GetBPRFileInfo(const BPRFileName: string; out BinaryFileName : string; const Description :
PString);
8749: Procedure GetDPKFileInfo( const DPKFileName : string; out RunOnly : Boolean; const LibSuffix : PString;
const Description : PString);
8750: Procedure GetBPKFileInfo( const BPKFileName : string; out RunOnly : Boolean; const BinaryFileName :
PString; const Description : PString);
8751: end;
8752:
8753: procedure SIRegister_JclFileUtils_max(CL: TPSPascalCompiler);
8754: begin
8755:   'ERROR_NO_MORE_FILES','LongInt').SetInt( 18);
8756:   //CL.AddDelphiFunction('Function stat64( FileName : PChar; var StatBuffer : TStatBuf64) : Integer');
8757:   //Function fstat64( FileDes : Integer; var StatBuffer : TStatBuf64) : Integer);
8758:   //Function lstat64( FileName : PChar; var StatBuffer : TStatBuf64) : Integer);
8759:   LPathSeparator,'String').SetString( '/');
8760:   LDirDelimiter,'String').SetString( '/');
8761:   LDirSeparator,'String').SetString( ':');
8762:   'JXPathDevicePrefix','String').SetString( '\\.');
8763:   'JXPathSeparator','String').SetString( '\');
8764:   'JXDirDelimiter','String').SetString( '\');
8765:   'JXDirSeparator','String').SetString( ';');
8766:   'JXPathUncPrefix','String').SetString( '\\');
8767:   'faNormalFile','LongWord').SetUInt( $00000080);

```

```

8768: //faUnixSpecific', '').SetString( faSymLink);
8769: JXTCompactPath', '( cpCenter, cpEnd )');
8770: _WIN32_FILE_ATTRIBUTE_DATA', 'record dwFileAttributes : DWORD; f'
8771: +tCreationTime : TFileTime; ftLastAccessTime : TFileTime; ftLastWriteTime :
8772: + ' TFileTime; nFileSizeHigh : DWORD; nFileSizeLow : DWORD; end');
8773: TWIN32FileAttributeData', '_WIN32_FILE_ATTRIBUTE_DATA');
8774: WIN32_FILE_ATTRIBUTE_DATA', '_WIN32_FILE_ATTRIBUTE_DATA');
8775:
8776: procedure SIRegister_StdVCL(CL: TPSPascalCompiler);
8777: begin
8778: LIBID_StdVCL', 'string').SetString( '{EE05DFE0-5549-11D0-9EA9-0020AF3D82DA}');
8779: CL.AddConstantN('IID_IProvider', 'string').SetString( '{6E644935-51F7-11D0-8D41-00A0248E4B9A}');
8780: IID_IStrings', 'string').SetString( '{EE05DFE2-5549-11D0-9EA9-0020AF3D82DA}');
8781: IID_IDataBroker', 'string').SetString( '{6539BF65-6FE7-11D0-9E8C-00A02457621F}');
8782: CL.AddInterface(CL.FindInterface('IUNKNOWN'), IProvider, 'IProvider');
8783: //IProviderDisp', 'dispinterface');
8784: CL.AddInterface(CL.FindInterface('IUNKNOWN'), IStrings, 'IStrings');
8785: //IStringsDisp', 'dispinterface');
8786: CL.AddInterface(CL.FindInterface('IUNKNOWN'), IDataBroker, 'IDataBroker');
8787: //IDataBrokerDisp', 'dispinterface');
8788: SIRegister_IProvider(CL);
8789: SIRegister_IStrings(CL);
8790: SIRegister_IDataBroker(CL);
8791: end;
8792:
8793:
8794: procedure SIRegister_AxCtrls(CL: TPSPascalCompiler);
8795: begin
8796: //Class_DColorPropPage', 'TGUID').SetString( '{5CFF5D59-5946-11D0-BDEF-00A024D1875C}');
8797: //Class_DFontPropPage', 'TGUID').SetString( '{5CFF5D5B-5946-11D0-BDEF-00A024D1875C}');
8798: //Class_DPicturePropPage', 'TGUID').SetString( '{5CFF5D5A-5946-11D0-BDEF-00A024D1875C}');
8799: //Class_DStringPropPage', 'TGUID').SetString( '{F42D677E-754B-11D0-BDFB-00A024D1875C}');
8800: SIRegister_TOLEStream(CL);
8801: CL.AddClassN(CL.FindClass('TOBJECT'), 'TConnectionPoints');
8802: TConnectionKind', '( ckSingle, ckMulti )');
8803: SIRegister_TConnectionPoint(CL);
8804: SIRegister_TConnectionPoints(CL);
8805: TDefinePropertyPage', 'Procedure ( const GUID : TGUID)';
8806: CL.AddClassN(CL.FindClass('TOBJECT'), 'TActiveXControlFactory');
8807: SIRegister_TActiveXControl(CL);
8808: //TActiveXControlClass', 'class of TActiveXControl');
8809: SIRegister_TActiveXControlFactory(CL);
8810: SIRegister_TActiveFormControl(CL);
8811: SIRegister_TActiveForm(CL);
8812: //TActiveFormClass', 'class of TActiveForm');
8813: SIRegister_TActiveFormFactory(CL);
8814: CL.AddClassN(CL.FindClass('TOBJECT'), 'TPropertyPageImpl');
8815: SIRegister_TPropertyPage(CL);
8816: //TPropertyPageClass', 'class of TPropertyPage');
8817: SIRegister_TPropertyPageImpl(CL);
8818: SIRegister_TActiveXPropertyPage(CL);
8819: SIRegister_TActiveXPropertyPageFactory(CL);
8820: SIRegister_TCustomAdapter(CL);
8821: SIRegister_TAdapterNotifier(CL);
8822: SIRegister_IFontAccess(CL);
8823: SIRegister_TFontAdapter(CL);
8824: SIRegister_IPictureAccess(CL);
8825: SIRegister_TPictureAdapter(CL);
8826: SIRegister_TOLEGraphic(CL);
8827: SIRegister_TStringsAdapter(CL);
8828: SIRegister_TReflectorWindow(CL);
8829: Procedure EnumDispatchProperties(Dispatch:IDispatch; PropType:TGUID; VTCode:Integer;PropList:TStrings)';
8830: //Procedure GetOleFont( Font : TFont; var OleFont : IFontDisp)';
8831: //Procedure SetOleFont( Font : TFont; OleFont : IFontDisp)';
8832: //Procedure GetOlePicture( Picture : TPicture; var OlePicture : IPictureDisp)';
8833: //Procedure SetOlePicture( Picture : TPicture; OlePicture : IPictureDisp)';
8834: Procedure GetOleStrings( Strings : TStrings; var OleStrings : IStrings)';
8835: Procedure SetOleStrings( Strings : TStrings; OleStrings : IStrings)';
8836: Function ParkingWindow : HWND);
8837: end;
8838:
8839: procedure SIRegister_GLVectorFileObjects(CL: TPSPascalCompiler);
8840: begin
8841: CL.AddClassN(CL.FindClass('TOBJECT'), 'TMeshObjectList');
8842: CL.AddClassN(CL.FindClass('TOBJECT'), 'TFaceGroups');
8843: TMeshAutoCentering', '( macCenterX, macCenterY, macCenterZ, macUseBarycenter )');
8844: TMeshAutoCenterings', 'set of TMeshAutoCentering');
8845: TMeshObjectMode', '( momTriangles, momTriangleStrip, momFaceGroups )';
8846: SIRegister_TBaseMeshObject(CL);
8847: CL.AddClassN(CL.FindClass('TOBJECT'), 'TSkeletonFrameList');
8848: TSkeletonFrameTransform', '( sftRotation, sftQuaternion )';
8849: SIRegister_TSkeletonFrame(CL);
8850: SIRegister_TSkeletonFrameList(CL);
8851: CL.AddClassN(CL.FindClass('TOBJECT'), 'TSkeleton');
8852: CL.AddClassN(CL.FindClass('TOBJECT'), 'TSkeletonBone');
8853: SIRegister_TSkeletonBoneList(CL);
8854: SIRegister_TSkeletonRootBoneList(CL);
8855: SIRegister_TSkeletonBone(CL);
8856: CL.AddClassN(CL.FindClass('TOBJECT'), 'TSkeletonColliderList');

```



```

8857: SIRegister_TSkeletonCollider(CL);
8858: SIRegister_TSkeletonColliderList(CL);
8859: CL.AddClassN(CL.FindClass('TOBJECT'),'TGLBaseMesh');
8860: TBlendedLerpInfo', 'record frameIndex1 : Integer; frameIndex2 : '
8861:   +Integer; lerpFactor : Single; weight : Single; externalPositions : TAffine'
8862:   +VectorList; externalRotations : TAffineVectorList; externalQuaternions : T'
8863:   +QuaternionList; end');
8864: SIRegister_TSkeleton(CL);
8865: TMeshObjectRenderingOption', '( moroGroupByMaterial )');
8866: TMeshObjectRenderingOptions', 'set of TMeshObjectRenderingOption');
8867: SIRegister_TMeshObject(CL);
8868: SIRegister_TMeshObjectList(CL);
8869: //TMeshObjectListClass', 'class of TMeshObjectList');
8870: CL.AddClassN(CL.FindClass('TOBJECT'),'TMeshMorphTargetList');
8871: SIRegister_TMeshMorphTarget(CL);
8872: SIRegister_TMeshMorphTargetList(CL);
8873: SIRegister_TMorphableMeshObject(CL);
8874: TVertexBoneWeight', 'record BoneID : Integer; Weight : Single; end');
8875: //TVertexBoneWeightArray', '^TVertexBoneWeightArray // will not work');
8876: //PVerticesBoneWeights', 'TVerticesBoneWeights // will not work');
8877: TVertexBoneWeightDynArray', 'array of TVertexBoneWeight');
8878: SIRegister_TSkeletonMeshObject(CL);
8879: SIRegister_TFaceGroup(CL);
8880: TFaceGroupMeshMode', '( fgmTriangles, fgmTriangleStrip, fgmFl'
8881:   +atTriangles, fgmTriangleFan, fgmQuads )');
8882: SIRegister_TFGVertexIndexList(CL);
8883: SIRegister_TFGVertexNormalTexIndexList(CL);
8884: SIRegister_TFGIndexTexCoordList(CL);
8885: SIRegister_TFaceGroups(CL);
8886: TMeshNormalsOrientation', '( mnoDefault, mnoInvert )');
8887: SIRegister_TVectorFile(CL);
8888: //TVectorFileClass', 'class of TVectorFile');
8889: SIRegister_TGLGLSMVectorFile(CL);
8890: SIRegister_TGLBaseMesh(CL);
8891: SIRegister_TGLFreeForm(CL);
8892: TGLActorOption', '( aoSkeletonNormalizeNormals )');
8893: TGLActorOptions', 'set of TGLActorOption');
8894: cDefaultGLActorOptions', 'LongInt'.Value.ts32 := ord(aoSkeletonNormalizeNormals);
8895: CL.AddClassN(CL.FindClass('TOBJECT'),'TGLActor');
8896: TActorAnimationReference', '( aarMorph, aarSkeleton, aarNone )');
8897: SIRegister_TActorAnimation(CL);
8898: TActorAnimationName', 'String';
8899: SIRegister_TActorAnimations(CL);
8900: SIRegister_TGLBaseAnimationController(CL);
8901: SIRegister_TGLAnimationController(CL);
8902: TActorFrameInterpolation', '( afpNone, afpLinear )');
8903: TActorAnimationMode', '( aamNone, aamPlayOnce, aamLoop, aamBounc'
8904:   +eForward, aamBounceBackward, aamLoopBackward, aamExternal )');
8905: SIRegister_TGLActor(CL);
8906: SIRegister_TVectorFileFormat(CL);
8907: SIRegister_TVectorFileFormatsList(CL);
8908: CL.AddClassN(CL.FindClass('TOBJECT'),'EInvalidVectorFile');
8909: Function GetVectorFileFormats : TVectorFileFormatsList';
8910: Function VectorFileFormatsFilter : String';
8911: Function VectorFileFormatsSaveFilter : String';
8912: Function VectorFileFormatExtensionByIndex( index : Integer) : String';
8913: //Procedure RegisterVectorFileFormat(const aExtension, aDescription:String; aClass:TVectorFileClass)';
8914: //Procedure UnregisterVectorFileClass( aClass : TVectorFileClass)';
8915: end;
8916:
8917: procedure SIRegister_StdVCL(CL: TPSPascalCompiler);
8918: begin
8919:   LIBID_StdVCL,'string').SetString( '{EE05DFE0-5549-11D0-9EA9-0020AF3D82DA}');
8920:   IID_IProvider,'string').SetString( '{6E644935-51F7-11D0-8D41-00A0248E4B9A}');
8921:   IID_IStrings,'string').SetString( '{EE05DFE2-5549-11D0-9EA9-0020AF3D82DA}');
8922:   IID_IDataBroker,'string').SetString( '{6539BF65-6FE7-11D0-9E8C-00A02457621F}');
8923:   CL.AddInterface(CL.FindInterface('IUNKNOWN'),IProvider, 'IProvider');
8924:   //IProviderDisp', 'dispinterface');
8925:   CL.AddInterface(CL.FindInterface('IUNKNOWN'),IStrings, 'IStrings');
8926:   //IStringsDisp', 'dispinterface');
8927:   CL.AddInterface(CL.FindInterface('IUNKNOWN'),IDataBroker, 'IDataBroker');
8928:   //IDataBrokerDisp', 'dispinterface');
8929:   SIRegister_IProvider(CL);
8930:   SIRegister_IStrings(CL);
8931:   SIRegister_IDataBroker(CL);
8932: end;
8933:
8934: procedure SIRegister_AxCtrls(CL: TPSPascalCompiler);
8935: begin
8936:   Class_DColorPropPage', 'TGUID').SetString( '{5CFF5D59-5946-11D0-BDEF-00A024D1875C}');
8937:   Class_DFontPropPage', 'TGUID').SetString( '{5CFF5D5B-5946-11D0-BDEF-00A024D1875C}');
8938:   Class_DPicturePropPage', 'TGUID').SetString( '{5CFF5D5A-5946-11D0-BDEF-00A024D1875C}');
8939:   Class_DStringPropPage', 'TGUID').SetString( '{F42D677E-754B-11D0-BDFB-00A024D1875C}');
8940:   SIRegister_TOLEStream(CL);
8941:   CL.AddClassN(CL.FindClass('TOBJECT'),'TConnectionPoints');
8942:   TConnectionKind', '( ckSingle, ckMulti )');
8943:   SIRegister_TConnectionPoint(CL);
8944:   SIRegister_TConnectionPoints(CL);
8945:   TDefinePropertyPage', 'Procedure ( const GUID : TGUID)';

```

PC08 G:\sonysavefeb2014\maxboxfiles\TypeConstClasslistSortmX4Reporfinal.txt
http://www.softwareschule.ch/maxbox.htm

```

9017: Const 'ErrNoDeviceAnswer', 'LongInt').SetInt( 9995);
9018: Const 'ErrMaxBuffer', 'LongInt').SetInt( 9996);
9019: Const 'ErrTimeout', 'LongInt').SetInt( 9997);
9020: Const 'ErrNotRead', 'LongInt').SetInt( 9998);
9021: Const 'ErrFrame', 'LongInt').SetInt( 9999);
9022: Const 'ErrOverrun', 'LongInt').SetInt( 10000);
9023: Const 'ErrRxOver', 'LongInt').SetInt( 10001);
9024: Const 'ErrRxParity', 'LongInt').SetInt( 10002);
9025: Const 'ErrTxFull', 'LongInt').SetInt( 10003);
9026: Const 'dcb_Binary', 'LongWord').SetUInt( $00000001);
9027: Const 'dcb_ParityCheck', 'LongWord').SetUInt( $00000002);
9028: Const 'dcb_OutxCtsFlow', 'LongWord').SetUInt( $00000004);
9029: Const 'dcb_OutxDsrFlow', 'LongWord').SetUInt( $00000008);
9030: Const 'dcb_DtrControlMask', 'LongWord').SetUInt( $00000030);
9031: Const 'dcb_DtrControlDisable', 'LongWord').SetUInt( $00000000);
9032: Const 'dcb_DtrControlEnable', 'LongWord').SetUInt( $00000010);
9033: Const 'dcb_DtrControlHandshake', 'LongWord').SetUInt( $00000020);
9034: Const 'dcb_DsrSensitivity', 'LongWord').SetUInt( $00000040);
9035: Const 'dcb_TXContinueOnXoff', 'LongWord').SetUInt( $00000080);
9036: Const 'dcb_OutX', 'LongWord').SetUInt( $00000100);
9037: Const 'dcb_InX', 'LongWord').SetUInt( $00000200);
9038: Const 'dcb_ErrorChar', 'LongWord').SetUInt( $00000400);
9039: Const 'dcb_NullStrip', 'LongWord').SetUInt( $00000800);
9040: Const 'dcb_RtsControlMask', 'LongWord').SetUInt( $00003000);
9041: Const 'dcb_RtsControlDisable', 'LongWord').SetUInt( $00000000);
9042: Const 'dcb_RtsControlEnable', 'LongWord').SetUInt( $00001000);
9043: Const 'dcb_RtsControlHandshake', 'LongWord').SetUInt( $00002000);
9044: Const 'dcb_RtsControlToggle', 'LongWord').SetUInt( $00003000);
9045: Const 'dcb_AbortOnError', 'LongWord').SetUInt( $00004000);
9046: Const 'dcb_Reserveds', 'LongWord').SetUInt( $FFFF8000);
9047: Const 'synSB1', 'LongInt').SetInt( 0);
9048: Const 'SBandHalf', 'LongInt').SetInt( 1);
9049: Const 'synSB2', 'LongInt').SetInt( 2);
9050: Const 'synINVALID_HANDLE_VALUE', 'LongInt').SetInt( THandle ( - 1 ));
9051: Const 'CS7fix', 'LongWord').SetUInt( $00000020);
9052: synTDCB, 'record DCBlength : DWORD; BaudRate : DWORD; Flags : Long'
9053:   + 'int; wReserved : Word; XonLim : Word; XoffLim : Word; ByteSize : Byte; Par'
9054:   + 'ity : Byte; StopBits : Byte; XonChar : CHAR; XoffChar : CHAR; ErrorChar : '
9055:   + 'CHAR; EofChar : CHAR; EvtChar : CHAR; wReserved1 : Word; end';
9056: //PDCB, '^TDCB // will not work';
9057: //Const 'MaxRates', 'LongInt').SetInt( 18);
9058: //Const 'MaxRates', 'LongInt').SetInt( 30);
9059: //Const 'MaxRates', 'LongInt').SetInt( 19);
9060: Const 'O_SYNC', 'LongWord').SetUInt( $0080);
9061: Const 'synOK', 'LongInt').SetInt( 0);
9062: Const 'synErr', 'LongInt').SetInt( integer ( - 1 ));
9063: THookSerialReason, '( HR_SerialClose, HR_Connect, HR_CanRead, HR_CanWrite, HR_ReadCount, HR_WriteCount,
HR_Wait )';
9064: Type('THookSerialStatus', 'Procedure (Sender:TObject; Reason:THookSerialReason; const Value:string)');
9065: SIRegister_ESynaSerError(CL);
9066: SIRegister_TBlockSerial(CL);
9067: CL.AddDelphiFunction('Function GetSerialPortNames : string');
9068: end;
9069:
9070:
9071: procedure SIRegister_tlntsend(CL: TPSPascalCompiler);
9072: begin
9073:   Const 'cTelnetProtocol', 'String').SetString( '23');
9074:   Const 'cSSHProtocol', 'String').SetString( '22');
9075:   Const 'TLNT_EOR', 'Char').SetString( #239);
9076:   Const 'TLNT_SE', 'Char').SetString( #240);
9077:   Const 'TLNT_NOP', 'Char').SetString( #241);
9078:   Const 'TLNT_DATA_MARK', 'Char').SetString( #242);
9079:   Const 'TLNT_BREAK', 'Char').SetString( #243);
9080:   Const 'TLNT_IP', 'Char').SetString( #244);
9081:   Const 'TLNT_AO', 'Char').SetString( #245);
9082:   Const 'TLNT_AYT', 'Char').SetString( #246);
9083:   Const 'TLNT_EC', 'Char').SetString( #247);
9084:   Const 'TLNT_EL', 'Char').SetString( #248);
9085:   Const 'TLNT_GA', 'Char').SetString( #249);
9086:   Const 'TLNT_SB', 'Char').SetString( #250);
9087:   Const 'TLNT_WILL', 'Char').SetString( #251);
9088:   Const 'TLNT_WONT', 'Char').SetString( #252);
9089:   Const 'TLNT_DO', 'Char').SetString( #253);
9090:   Const 'TLNT_DONT', 'Char').SetString( #254);
9091:   Const 'TLNT_IAC', 'Char').SetString( #255);
9092:   TTelnetState, '( tsDATA, tsIAC, tsIAC_SB, tsIAC_WILL, tsIAC_DO, '
9093:   + ' tsIAC_WONT, tsIAC_DONT, tsIAC_SBIAC, tsIAC_SBDATA, tsSBDATA_IAC )';
9094:   SIRegister_TTelnetSend(CL);
9095: end;
9096:
9097: procedure SIRegister_pingsend(CL: TPSPascalCompiler);
9098: begin
9099:   Const 'ICMP_ECHO', 'LongInt').SetInt( 8);
9100:   Const 'ICMP_ECHOREPLY', 'LongInt').SetInt( 0);
9101:   Const 'ICMP_UNREACH', 'LongInt').SetInt( 3);
9102:   Const 'ICMP_TIME_EXCEEDED', 'LongInt').SetInt( 11);
9103:   Const 'ICMP6_ECHO', 'LongInt').SetInt( 128);
9104:   Const 'ICMP6_ECHOREPLY', 'LongInt').SetInt( 129);

```

```

9105: Const 'ICMP6_UNREACH', 'LongInt').SetInt( 1);
9106: Const 'ICMP6_TIME_EXCEEDED', 'LongInt').SetInt( 3);
9107: TICMPError', '( IE_NoError, IE_Other, IE_TTLExceed, IE_UnreachOt
9108: +her, IE_UnreachRoute, IE_UnreachAdmin, IE_UnreachAddr, IE_UnreachPort ));
9109: SIRegister_TPINGSend(CL);
9110: CL.AddDelphiFunction('Function PingHost( const Host : string) : Integer');
9111: CL.AddDelphiFunction('Function TraceRouteHost( const Host : string) : string');
9112: end;
9113:
9114: procedure SIRegister_asnutil(CL: TPSPascalCompiler);
9115: begin
9116: Const 'synASN1_BOOL', 'LongWord').SetUInt( $01);
9117: Const 'synASN1_INT', 'LongWord').SetUInt( $02);
9118: Const 'synASN1_OCTSTR', 'LongWord').SetUInt( $04);
9119: Const 'synASN1_NULL', 'LongWord').SetUInt( $05);
9120: Const 'synASN1_OBJID', 'LongWord').SetUInt( $06);
9121: Const 'synASN1_ENUM', 'LongWord').SetUInt( $0a);
9122: Const 'synASN1_SEQ', 'LongWord').SetUInt( $30);
9123: Const 'synASN1_SETOF', 'LongWord').SetUInt( $31);
9124: Const 'synASN1_IPADDR', 'LongWord').SetUInt( $40);
9125: Const 'synASN1_COUNTER', 'LongWord').SetUInt( $41);
9126: Const 'synASN1_GAUGE', 'LongWord').SetUInt( $42);
9127: Const 'synASN1_TIMETICKS', 'LongWord').SetUInt( $43);
9128: Const 'synASN1_OPAQUE', 'LongWord').SetUInt( $44);
9129: end;
9130:
9131: procedure SIRegister_dnssend(CL: TPSPascalCompiler);
9132: begin
9133: Const 'cDnsProtocol', 'String').SetString( '53');
9134: Const 'QTYPE_A', 'LongInt').SetInt( 1);
9135: Const 'QTYPE_NS', 'LongInt').SetInt( 2);
9136: Const 'QTYPE_MD', 'LongInt').SetInt( 3);
9137: Const 'QTYPE_MF', 'LongInt').SetInt( 4);
9138: Const 'QTYPE_CNAME', 'LongInt').SetInt( 5);
9139: Const 'QTYPE_SOA', 'LongInt').SetInt( 6);
9140: Const 'QTYPE_MB', 'LongInt').SetInt( 7);
9141: Const 'QTYPE_MG', 'LongInt').SetInt( 8);
9142: Const 'QTYPE_MR', 'LongInt').SetInt( 9);
9143: Const 'QTYPE_NULL', 'LongInt').SetInt( 10);
9144: Const 'QTYPE_WKS', 'LongInt').SetInt( 11);
9145: Const 'QTYPE_PTR', 'LongInt').SetInt( 12);
9146: Const 'QTYPE_HINFO', 'LongInt').SetInt( 13);
9147: Const 'QTYPE_MINFO', 'LongInt').SetInt( 14);
9148: Const 'QTYPE_MX', 'LongInt').SetInt( 15);
9149: Const 'QTYPE_TXT', 'LongInt').SetInt( 16);
9150: Const 'QTYPE_RP', 'LongInt').SetInt( 17);
9151: Const 'QTYPE_AFSDB', 'LongInt').SetInt( 18);
9152: Const 'QTYPE_X25', 'LongInt').SetInt( 19);
9153: Const 'QTYPE_ISDN', 'LongInt').SetInt( 20);
9154: Const 'QTYPE_RT', 'LongInt').SetInt( 21);
9155: Const 'QTYPE_NSAP', 'LongInt').SetInt( 22);
9156: Const 'QTYPE_NSAPPTR', 'LongInt').SetInt( 23);
9157: Const 'QTYPE_SIG', 'LongInt').SetInt( 24);
9158: Const 'QTYPE_KEY', 'LongInt').SetInt( 25);
9159: Const 'QTYPE_PX', 'LongInt').SetInt( 26);
9160: Const 'QTYPE_GPOS', 'LongInt').SetInt( 27);
9161: Const 'QTYPE_AAAA', 'LongInt').SetInt( 28);
9162: Const 'QTYPE_LOC', 'LongInt').SetInt( 29);
9163: Const 'QTYPE_NXT', 'LongInt').SetInt( 30);
9164: Const 'QTYPE_SRV', 'LongInt').SetInt( 33);
9165: Const 'QTYPE_NAPTR', 'LongInt').SetInt( 35);
9166: Const 'QTYPE_KX', 'LongInt').SetInt( 36);
9167: Const 'QTYPE_SPF', 'LongInt').SetInt( 99);
9168: Const 'QTYPE_AXFR', 'LongInt').SetInt( 252);
9169: Const 'QTYPE_MAILB', 'LongInt').SetInt( 253);
9170: Const 'QTYPE_MAILA', 'LongInt').SetInt( 254);
9171: Const 'QTYPE_ALL', 'LongInt').SetInt( 255);
9172: SIRegister_TDNSSend(CL);
9173: Function GetMailServers( const DNSHost, Domain : AnsiString; const Servers : TStrings) : Boolean);
9174: end;
9175:
9176: procedure SIRegister_ldapsend(CL: TPSPascalCompiler);
9177: begin
9178: Const 'cLDAPProtocol', 'String').SetString( '389');
9179: Const 'LDAP_ASNI_BIND_REQUEST', 'LongWord').SetUInt( $60);
9180: Const 'LDAP_ASNI_BIND_RESPONSE', 'LongWord').SetUInt( $61);
9181: Const 'LDAP_ASNI_UNBIND_REQUEST', 'LongWord').SetUInt( $42);
9182: Const 'LDAP_ASNI_SEARCH_REQUEST', 'LongWord').SetUInt( $63);
9183: Const 'LDAP_ASNI_SEARCH_ENTRY', 'LongWord').SetUInt( $64);
9184: Const 'LDAP_ASNI_SEARCH_DONE', 'LongWord').SetUInt( $65);
9185: Const 'LDAP_ASNI_SEARCH_REFERENCE', 'LongWord').SetUInt( $73);
9186: Const 'LDAP_ASNI_MODIFY_REQUEST', 'LongWord').SetUInt( $66);
9187: Const 'LDAP_ASNI_MODIFY_RESPONSE', 'LongWord').SetUInt( $67);
9188: Const 'LDAP_ASNI_ADD_REQUEST', 'LongWord').SetUInt( $68);
9189: Const 'LDAP_ASNI_ADD_RESPONSE', 'LongWord').SetUInt( $69);
9190: Const 'LDAP_ASNI_DEL_REQUEST', 'LongWord').SetUInt( $4A);
9191: Const 'LDAP_ASNI_DEL_RESPONSE', 'LongWord').SetUInt( $6B);
9192: Const 'LDAP_ASNI_MODIFYDN_REQUEST', 'LongWord').SetUInt( $6C);
9193: Const 'LDAP_ASNI_MODIFYDN_RESPONSE', 'LongWord').SetUInt( $6D);

```



```

9194: Const 'LDAP_ASNI_COMPARE_REQUEST', 'LongWord').SetUInt( $6E);
9195: Const 'LDAP_ASNI_COMPARE_RESPONSE', 'LongWord').SetUInt( $6F);
9196: Const 'LDAP_ASNI_ABANDON_REQUEST', 'LongWord').SetUInt( $70);
9197: Const 'LDAP_ASNI_EXT_REQUEST', 'LongWord').SetUInt( $77);
9198: Const 'LDAP_ASNI_EXT_RESPONSE', 'LongWord').SetUInt( $78);
9199: SIRegister_TLDAPAttribute(CL);
9200: SIRegister_TLDAPAttributeList(CL);
9201: SIRegister_TLDAPResult(CL);
9202: SIRegister_TLDAPResultList(CL);
9203: TLDAPModifyOp, '( MO_Add, MO_Delete, MO_Replace )';
9204: TLDAPSearchScope, '( SS_BaseObject, SS_SingleLevel, SS_WholeSubtree )';
9205: TLDAPSearchAliases, '( SA_NeverDeref, SA_InSearching, SA_FindingBaseObj, SA_Always )';
9206: SIRegister_TLDAPSend(CL);
9207: Function LDAPResultDump( const Value : TLDAPResultList ) : AnsiString';
9208: end;
9209:
9210: procedure SIRegister_slogsend(CL: TPSPascalCompiler);
9211: begin
9212:   Const 'cSysLogProtocol', 'String').SetString( '514');
9213:   Const 'FCL_Kernel', 'LongInt').SetInt( 0);
9214:   Const 'FCL_UserLevel', 'LongInt').SetInt( 1);
9215:   Const 'FCL_MailSystem', 'LongInt').SetInt( 2);
9216:   Const 'FCL_System', 'LongInt').SetInt( 3);
9217:   Const 'FCL_Security', 'LongInt').SetInt( 4);
9218:   Const 'FCL_Syslogd', 'LongInt').SetInt( 5);
9219:   Const 'FCL_Printer', 'LongInt').SetInt( 6);
9220:   Const 'FCL_News', 'LongInt').SetInt( 7);
9221:   Const 'FCL_UUCP', 'LongInt').SetInt( 8);
9222:   Const 'FCL_Clock', 'LongInt').SetInt( 9);
9223:   Const 'FCL_Authorization', 'LongInt').SetInt( 10);
9224:   Const 'FCL_FTP', 'LongInt').SetInt( 11);
9225:   Const 'FCL_NTP', 'LongInt').SetInt( 12);
9226:   Const 'FCL_LogAudit', 'LongInt').SetInt( 13);
9227:   Const 'FCL_LogAlert', 'LongInt').SetInt( 14);
9228:   Const 'FCL_Time', 'LongInt').SetInt( 15);
9229:   Const 'FCL_Local0', 'LongInt').SetInt( 16);
9230:   Const 'FCL_Local1', 'LongInt').SetInt( 17);
9231:   Const 'FCL_Local2', 'LongInt').SetInt( 18);
9232:   Const 'FCL_Local3', 'LongInt').SetInt( 19);
9233:   Const 'FCL_Local4', 'LongInt').SetInt( 20);
9234:   Const 'FCL_Local5', 'LongInt').SetInt( 21);
9235:   Const 'FCL_Local6', 'LongInt').SetInt( 22);
9236:   Const 'FCL_Local7', 'LongInt').SetInt( 23);
9237:   TSyslogSeverity, '( Emergency, Alert, Critical, Error, Warning, Notice, Info, Debug )';
9238:   SIRegister_TSyslogMessage(CL);
9239:   SIRegister_TSyslogSend(CL);
9240:   Function ToSysLog(const SyslogServer: string; Facil:Byte; Sever:TSyslogSeverity;const
Content:string): Boolean;
9241: end;
9242:
9243: procedure SIRegister_mimepart(CL: TPSPascalCompiler);
9244: begin
9245:   CL.AddClassN(CL.FindClass('TOBJECT'), 'TMimePart');
9246:   THookWalkPart, 'Procedure ( const Sender : TMimePart)';
9247:   TMimePrimary, '( MP_TEXT, MP_MULTIPART, MP_MESSAGE, MP_BINARY )';
9248:   TMimeEncoding, '( ME_7BIT, ME_8BIT, ME_QUOTED_PRINTABLE, ME_BASE64, ME_UU, ME_XX )';
9249:   SIRegister_TMimePart(CL);
9250:   Const 'MaxMimeType', 'LongInt').SetInt( 25);
9251:   CL.AddDelphiFunction('Function GenerateBoundary : string');
9252: end;
9253:
9254:
9255: procedure SIRegister_ftpsend(CL: TPSPascalCompiler);
9256: begin
9257:   Const 'cFtpProtocol', 'String').SetString( '21');
9258:   Const 'cFtpDataProtocol', 'String').SetString( '20');
9259:   Const 'FTP_OK', 'LongInt').SetInt( 255);
9260:   Const 'FTP_ERR', 'LongInt').SetInt( 254);
9261:   TFTPStatus, 'Procedure ( Sender : TObject; Response : Boolean; const Value : string)';
9262:   SIRegister_TFTPListRec(CL);
9263:   SIRegister_TFTPList(CL);
9264:   SIRegister_TFTPSend(CL);
9265: end;
9266:
9267:
9268: procedure SIRegister_ftptsend(CL: TPSPascalCompiler);
9269: begin
9270:   Const 'cTFTPProtocol', 'String').SetString( '69');
9271:   Const 'cTFTP_RRQ', 'LongInt').SetInt( word ( 1 ));
9272:   Const 'cTFTP_WRQ', 'LongInt').SetInt( word ( 2 ));
9273:   Const 'cTFTP_DTA', 'LongInt').SetInt( word ( 3 ));
9274:   Const 'cTFTP_ACK', 'LongInt').SetInt( word ( 4 ));
9275:   Const 'cTFTP_ERR', 'LongInt').SetInt( word ( 5 ));
9276:   SIRegister_TTFTPSend(CL);
9277: end;
9278:
9279: procedure SIRegister_httpsend(CL: TPSPascalCompiler);
9280: begin
9281:   Const 'cHttpProtocol', 'String').SetString( '80');

```

```

9282:   TTransferEncoding', '( TE_UNKNOWN, TE_IDENTITY, TE_CHUNKED )');
9283:   SIRegister_THTTPSend(CL);
9284: end;
9285:
9286:
9287: procedure SIRegister_sntpsend(CL: TPSPascalCompiler);
9288: begin
9289:   Const'cNtpProtocol','String').SetString( '123');
9290:   TNtp', 'record mode : Byte; stratum : Byte; poll : Byte; Precisi'
9291:   + 'on : Byte; RootDelay : Longint; RootDispersion : Longint; RefID : Longint; '
9292:   + 'Ref1 : Longint; Ref2 : Longint; Org1 : Longint; Org2 : Longint; Rcv1 : Lon'
9293:   + 'gint; Rcv2 : Longint; Xmit1 : Longint; Xmit2 : Longint; end');
9294:   SIRegister_TSNTPSend(CL);
9295: end;
9296:
9297: procedure SIRegister_snmpsend(CL: TPSPascalCompiler);
9298: begin
9299:   Const'cSnmpProtocol','String').SetString( '161');
9300:   Const'cSnmpTrapProtocol','String').SetString( '162');
9301:   Const'SNMP_V1','LongInt').SetInt( 0);
9302:   Const'SNMP_V2C','LongInt').SetInt( 1);
9303:   Const'SNMP_V3','LongInt').SetInt( 3);
9304:   Const'PDUGetRequest','LongWord').SetUInt( $A0);
9305:   Const'PDUGetNextRequest','LongWord').SetUInt( $A1);
9306:   Const'PDUGetResponse','LongWord').SetUInt( $A2);
9307:   Const'PDUSetRequest','LongWord').SetUInt( $A3);
9308:   Const'PDUTrap','LongWord').SetUInt( $A4);
9309:   Const'PDUGetBulkRequest','LongWord').SetUInt( $A5);
9310:   Const'PDUIInformRequest','LongWord').SetUInt( $A6);
9311:   Const'PDUTrapV2','LongWord').SetUInt( $A7);
9312:   Const'PDUREport','LongWord').SetUInt( $A8);
9313:   Const'ENoError','LongInt').SetInt( 0);
9314:   Const'ETooBig','LongInt').SetInt( 1);
9315:   Const'ENoSuchName','LongInt').SetInt( 2);
9316:   Const'EBadValue','LongInt').SetInt( 3);
9317:   Const'EReadOnly','LongInt').SetInt( 4);
9318:   Const'EGenErr','LongInt').SetInt( 5);
9319:   Const'ENoAccess','LongInt').SetInt( 6);
9320:   Const'EWrongType','LongInt').SetInt( 7);
9321:   Const'EWrongLength','LongInt').SetInt( 8);
9322:   Const'EWrongEncoding','LongInt').SetInt( 9);
9323:   Const'EWrongValue','LongInt').SetInt( 10);
9324:   Const'ENoCreation','LongInt').SetInt( 11);
9325:   Const'EInconsistentValue','LongInt').SetInt( 12);
9326:   Const'EResourceUnavailable','LongInt').SetInt( 13);
9327:   Const'ECommitFailed','LongInt').SetInt( 14);
9328:   Const'EUndoFailed','LongInt').SetInt( 15);
9329:   Const'EAuthorizationError','LongInt').SetInt( 16);
9330:   Const'ENotWritable','LongInt').SetInt( 17);
9331:   Const'EInconsistentName','LongInt').SetInt( 18);
9332:   TV3Flags', '( NoAuthNoPriv, AuthNoPriv, AuthPriv )';
9333:   TV3Auth', '( AuthMD5, AuthSHA1 )';
9334:   TV3Priv', '( PrivDES, Priv3DES, PrivAES )';
9335:   SIRegister_TSNMPMib(CL);
9336:   TV3Sync', 'record EngineID : AnsiString; EngineBoots : integer; '
9337:   + 'EngineTime : integer; EngineStamp : Cardinal; end';
9338:   SIRegister_TSNMPRec(CL);
9339:   SIRegister_TSNMPSend(CL);
9340: end;
9341:
9342: procedure SIRegister_pop3send(CL: TPSPascalCompiler);
9343: begin
9344:   Const'cPop3Protocol','String').SetString( '110');
9345:   TPOP3AuthType', '( POP3AuthAll, POP3AuthLogin, POP3AuthAPOP )';
9346:   SIRegister_TPOP3Send(CL);
9347: end;
9348:
9349: begin
9350:   Const'cNNTPProtocol','String').SetString( '119');
9351:   SIRegister_TNNTPSend(CL);
9352: end;
9353:
9354: procedure SIRegister_IdComponent(CL: TPSPascalCompiler);
9355: begin
9356:   TIdStatus', '( hsResolving, hsConnecting, hsConnected, hsDisconn'
9357:   + 'ecting, hsDisconnected, hsStatusText, ftpTransfer, ftpReady, ftpAborted )';
9358:   TIdStatusEvent','Procedure(ASender:TObject;const AStatus:TIdStatus;const AStatusText:string)';
9359:   TWorkMode', '( wmRead, wmWrite )';
9360:   TWorkInfo', 'record Current : Integer; Max : Integer; Level : Integer; end';
9361:   TWorkBeginEvent','Procedure(Sender:TObject;AWorkMode:TWorkMode;const AWorkCountMax : Integer)';
9362:   TWorkEndEvent', 'Procedure ( Sender : TObject; AWorkMode : TWorkMode)';
9363:   TWorkEvent', 'Procedure ( Sender : TObject; AWorkMode : TWorkMode; const AWorkCount : Integer)';
9364:   SIRegister_TIdComponent(CL);
9365: end;
9366:
9367: procedure SIRegister_dbTvRecordList(CL: TPSPascalCompiler);
9368: begin
9369:   CL.AddTypeS('TTVRecordListDifference', '( tvrldNone, tvrldText, tvrldCount, tvrldNodeMoved )');
9370:   TInternalTVFindTextOption', '( itvftCaseInsensitive, itvftPartial )';

```

```

9371:   TInternalTVFindTextOptions', 'set of TInternalTVFindTextOption');
9372:   SIRegister_TTVRecordInfo(CL);
9373:   SIRegister_TTVTextList(CL);
9374:   SIRegister_TTVParentList(CL);
9375:   SIRegister_TTVRecordList(CL);
9376: end;
9377:
9378: procedure SIRegister_JsDBTreeView(CL: TPSPascalCompiler);
9379: begin
9380:   CL.AddClassN(CL.FindClass('TOBJECT'),'TJsDBTreeNode');
9381:   CL.AddClassN(CL.FindClass('TOBJECT'),'TJsDBTreeViewDataLink');
9382:   TFieldTypes', 'set of TFieldType');
9383:   SIRegister_TJsCustomDBTreeView(CL);
9384:   SIRegister_TJsDBTreeViewDataLink(CL);
9385:   SIRegister_TJsDBTreeNode(CL);
9386:   SIRegister_TJsDBTreeView(CL);
9387:   CL.AddClassN(CL.FindClass('TOBJECT'),'EJsDBTreeViewError');
9388: end;
9389:
9390: procedure SIRegister_TabGrid(CL: TPSPascalCompiler);
9391: begin
9392:   SIRegister_TDBTabGrid(CL);
9393:   //CL.AddDelphiFunction('Procedure Register');
9394: end;
9395:
9396: procedure SIRegister_dbTreeCBox(CL: TPSPascalCompiler);
9397: begin
9398:   CL.AddClassN(CL.FindClass('TOBJECT'),'TDbTreeLookupComboBox');
9399:   TCloseUpAction', '( caCancel, caAccept, caClear )');
9400:   TCloseUpEvent', 'Procedure ( Action : TCloseUpAction)';
9401:   TAcceptNodeEvent', 'Procedure ( Node : TTreeNode; var Accept : Boolean)';
9402:   TDBTreeLCBOption', '( dtAcceptLeavesOnly, dtDontAcceptRoot, dtKeepDataSetConnected )';
9403:   TDBTreeLCBOptions', 'set of TDBTreeLCBOption';
9404:   SIRegister_TTreeSelect(CL);
9405:   CL.AddClassN(CL.FindClass('TOBJECT'),'TCustomDBLookupControl');
9406:   SIRegister_TCustomDBLookupControl(CL);
9407:   SIRegister_TCustomDBLookupControl(CL);
9408:   TCreateTreeSelectEvent', 'Function : TTreeSelect';
9409:   TGetTreeSelectEvent', 'Function : TTreeSelect';
9410:   TDropDownAlign', '( daLeft, daRight, daCenter )';
9411:   SIRegister_TDbTreeLookupComboBox(CL);
9412: end;
9413:
9414:
9415: procedure SIRegister_TreeVwEx(CL: TPSPascalCompiler);
9416: begin
9417:   CL.AddClassN(CL.FindClass('TOBJECT'),'TCustomTreeViewEx');
9418:   TTreeViewExOption', '( tveAllowDelete, tveAllowInsert, tveAutoDr
9419:   + 'agMove, tveConfirmDelete, tveInsertAsChild, tveLabelsReadOnly, tveMouseMov'
9420:   + 'eSelect, tveMultipleRootsAllowed, tveRootItemReadOnly )';
9421:   TTreeViewExOptions', 'set of TTreeViewExOption';
9422:   TTVDragImageShow', '( tvdisDefault, tvdisAlways, tvdisNever )';
9423:   TTreeViewExState', '( tvesIgnoreNextWMChar, tvesMouseStillDownAf'
9424:   + 'terDoubleClick, tvesRightButtonPressed, tvesWaitingForPopupMenu )';
9425:   TTreeViewExStates', 'set of TTreeViewExState';
9426:   TIgnoreWMChars', 'set of AnsiChar';
9427:   TTVDraggingEvent', 'Procedure ( Sender : TObject; Node : TTreeNode
9428:   + 'de; var AllowDrag : Boolean)';
9429:   TTVDeletingEvent', 'Procedure ( Sender : TObject; Node : TTreeNode
9430:   + 'de; var AllowDelete : Boolean)';
9431:   SIRegister_TCustomTreeViewEx(CL);
9432:   SIRegister_TTreeViewEx(CL);
9433: end;
9434:
9435: procedure SIRegister_dbTree(CL: TPSPascalCompiler);
9436: begin
9437:   CL.AddClassN(CL.FindClass('TOBJECT'),'TCustomDBTreeView');
9438:   TDBTreeOption', '( dtAllowDelete, dtAllowInsert, dtAutoDragMove,
9439:   + ' dtAutoExpand, dtAutoShowRoot, dtCancelOnExit, dtConfirmDelete, dtFocusOnE'
9440:   + 'dit, dtInsertAsChild, dtMouseMoveSelect, dtRebuildFocusedOnly, dtRootItemR'
9441:   + 'eadOnly, dtSynchronizeDataSet )';
9442:   TDBTreeOptions', 'set of TDBTreeOption';
9443:   TDBTVGetNextIDEvent', 'Function ( Sender : TObject; DataSet : TDataSet ) : string';
9444:   TTVFindTextOption', '( tvftCaseInsensitive, tvftPartial )';
9445:   TTVFindTextOptions', 'set of TTVFindTextOption';
9446:   TDBTreeViewState', '( dtvsBuilding, dtvsDatasetInEditMode, dtvsE'
9447:   + 'ditAfterReBuild, dtvsDatasetInInsertMode, dtvsLostFocusWhileDatasetInEditM'
9448:   + 'odes, dtvsChangingDataset, dtvsNeedReBuildAfterPost, dtvsNeedReBuild, dtvsDataSetNeedsRefresh )';
9449:   TDBTreeStates', 'set of TDBTreeViewState';
9450:   SIRegister_TTreeViewLink(CL);
9451:   SIRegister_TTreeViewIDNode(CL);
9452:   SIRegister_TCustomDBTreeView(CL);
9453:   SIRegister_TDBTreeView(CL);
9454: end;
9455:
9456: procedure SIRegister_SockTransport(CL: TPSPascalCompiler);
9457: begin
9458:   SIRegister_IDataBlock(CL);
9459:   SIRegister_ISendDataBlock(CL);

```

```

9460: SIRegister_ITransport(CL);
9461: SIRegister_TDataBlock(CL);
9462: //CL.AddTypeS('PIntArray', '^TIntArray // will not work');
9463: //CL.AddTypeS('PVariantArray', '^TVariantArray // will not work');
9464: CL.AddTypeS('TVarFlag', '( vfByRef, vfVariant )');
9465: CL.AddTypeS('TVarFlags', 'set of TVarFlag');
9466: SIRegister_TCustomDataBlockInterpreter(CL);
9467: SIRegister_TSendDataBlock(CL);
9468: const 'CallSig', 'LongWord').SetUInt( $D800);
9469: CL.AddConstantN('ResultSig', 'LongWord').SetUInt( $D400);
9470: CL.AddConstantN('asMask', 'LongWord').SetUInt( $00FF);
9471: CL.AddClassN(CL.FindClass('TOBJECT'), 'EInterpreterError');
9472: CL.AddClassN(CL.FindClass('TOBJECT'), 'ESocketConnectionError');
9473: CL.AddDelphiFunction('Procedure CheckSignature( Sig : Integer)');
9474: end;
9475:
9476: procedure SIRegister_WinInet(CL: TPSPascalCompiler);
9477: begin
9478:   Type('HINTERNET', '__Pointer');
9479:   Type('HINTERNET1', 'integer');
9480:   //CL.AddTypeS('PHINTERNET', '^HINTERNET // will not work');
9481:   //CL.AddTypeS('LPHINTERNET', 'PHINTERNET');
9482:   CL.AddTypeS('INTERNET_PORT', 'Word');
9483:   //CL.AddTypeS('PINTERNET_PORT', '^INTERNET_PORT // will not work');
9484:   //CL.AddTypeS('LPINTERNET_PORT', 'PINTERNET_PORT');
9485:   CL.AddConstantN('INTERNET_INVALID_PORT_NUMBER', 'LongInt').SetInt( 0);
9486:   Const 'INTERNET_DEFAULT_FTP_PORT', 'LongInt').SetInt( 21);
9487:   Const 'INTERNET_DEFAULT_GOPHER_PORT', 'LongInt').SetInt( 70);
9488:   Const 'INTERNET_DEFAULT_HTTP_PORT', 'LongInt').SetInt( 80);
9489:   Const 'INTERNET_DEFAULT_HTTPS_PORT', 'LongInt').SetInt( 443);
9490:   Const 'INTERNET_DEFAULT_SOCKS_PORT', 'LongInt').SetInt( 1080);
9491:   Const 'MAX_CACHE_ENTRY_INFO_SIZE', 'LongInt').SetInt( 4096);
9492:   Const 'INTERNET_MAX_HOST_NAME_LENGTH', 'LongInt').SetInt( 256);
9493:   Const 'INTERNET_MAX_USER_NAME_LENGTH', 'LongInt').SetInt( 128);
9494:   Const 'INTERNET_MAX_PASSWORD_LENGTH', 'LongInt').SetInt( 128);
9495:   Const 'INTERNET_MAX_PORT_NUMBER_LENGTH', 'LongInt').SetInt( 5);
9496:   Const 'INTERNET_MAX_PORT_NUMBER_VALUE', 'LongInt').SetInt( 65535);
9497:   Const 'INTERNET_MAX_PATH_LENGTH', 'LongInt').SetInt( 2048);
9498:   Const 'INTERNET_MAX_SCHEME_LENGTH', 'LongInt').SetInt( 32);
9499:   Const 'INTERNET_MAX_PROTOCOL_NAME', 'String').SetString( 'gopher');
9500:   Const 'INTERNET_KEEP_ALIVE_UNKNOWN', 'LongInt').SetInt( - 1);
9501:   Const 'INTERNET_KEEP_ALIVE_ENABLED', 'LongInt').SetInt( 1);
9502:   Const 'INTERNET_KEEP_ALIVE_DISABLED', 'LongInt').SetInt( 0);
9503:   Const 'INTERNET_REQFLAG_FROM_CACHE', 'LongWord').SetUInt( $00000001);
9504:   Const 'INTERNET_REQFLAG_ASYNC', 'LongWord').SetUInt( $00000002);
9505:   Const 'INTERNET_REQFLAG_VIA_PROXY', 'LongWord').SetUInt( $00000004);
9506:   Const 'INTERNET_REQFLAG_NO_HEADERS', 'LongWord').SetUInt( $00000008);
9507:   Const 'INTERNET_REQFLAG_PASSIVE', 'LongWord').SetUInt( $00000010);
9508:   Const 'INTERNET_REQFLAG_CACHE_WRITE_DISABLED', 'LongWord').SetUInt( $00000040);
9509:   Const 'INTERNET_FLAG_RELOAD', 'LongWord').SetUInt( $80000000);
9510:   Const 'INTERNET_FLAG_RAW_DATA', 'LongWord').SetUInt( $40000000);
9511:   Const 'INTERNET_FLAG_EXISTING_CONNECT', 'LongWord').SetUInt( $20000000);
9512:   Const 'INTERNET_FLAG_ASYNC', 'LongWord').SetUInt( $10000000);
9513:   Const 'INTERNET_FLAG_PASSIVE', 'LongWord').SetUInt( $08000000);
9514:   Const 'INTERNET_FLAG_NO_CACHE_WRITE', 'LongWord').SetUInt( $04000000);
9515:   //Const 'INTERNET_FLAG_DONT_CACHE', 'LongWord').SetUInt( $02000000);
9516:   Const 'INTERNET_FLAG_MAKE_PERSISTENT', 'LongWord').SetUInt( $02000000);
9517:   Const 'INTERNET_FLAG_FROM_CACHE', 'LongWord').SetUInt( $01000000);
9518:   Const 'INTERNET_FLAG_OFFLINE', 'LongWord').SetUInt( $01000000);
9519:   Const 'INTERNET_FLAG_SECURE', 'LongWord').SetUInt( $00800000);
9520:   Const 'INTERNET_FLAG_KEEP_CONNECTION', 'LongWord').SetUInt( $00400000);
9521:   Const 'INTERNET_FLAG_NO_AUTO_REDIRECT', 'LongWord').SetUInt( $00200000);
9522:   Const 'INTERNET_FLAG_READ_PREFETCH', 'LongWord').SetUInt( $00100000);
9523:   Const 'INTERNET_FLAG_NO_COOKIES', 'LongWord').SetUInt( $00080000);
9524:   Const 'INTERNET_FLAG_NO_AUTH', 'LongWord').SetUInt( $00040000);
9525:   Const 'INTERNET_FLAG_CACHE_IF_NET_FAIL', 'LongWord').SetUInt( $00010000);
9526:   Const 'INTERNET_FLAG_IGNORE_CERT_CN_INVALID', 'LongWord').SetUInt( $00001000);
9527:   Const 'INTERNET_FLAG_IGNORE_CERT_DATE_INVALID', 'LongWord').SetUInt( $00002000);
9528:   Const 'INTERNET_FLAG_IGNORE_REDIRECT_TO_HTTPS', 'LongWord').SetUInt( $00004000);
9529:   Const 'INTERNET_FLAG_IGNORE_REDIRECT_TO_HTTP', 'LongWord').SetUInt( $00008000);
9530:   Const 'INTERNET_FLAG_RESYNCHRONIZE', 'LongWord').SetUInt( $00000800);
9531:   Const 'INTERNET_FLAG_HYPERLINK', 'LongWord').SetUInt( $00000400);
9532:   Const 'INTERNET_FLAG_NO_UI', 'LongWord').SetUInt( $00000200);
9533:   Const 'INTERNET_FLAG_PRAGMA_NOCACHE', 'LongWord').SetUInt( $00000100);
9534:   Const 'INTERNET_FLAG_CACHE_ASYNC', 'LongWord').SetUInt( $00000080);
9535:   Const 'INTERNET_FLAG_FORMS_SUBMIT', 'LongWord').SetUInt( $00000040);
9536:   Const 'INTERNET_FLAG_NEED_FILE', 'LongWord').SetUInt( $00000010);
9537:   //Const 'INTERNET_FLAG_MUST_CACHE_REQUEST', 'LongWord').SetUInt( $00000001);
9538:   Const 'FTP_TRANSFER_TYPE_UNKNOWN', 'LongWord').SetUInt( $00000000);
9539:   Const 'FTP_TRANSFER_TYPE_ASCII', 'LongWord').SetUInt( $00000001);
9540:   Const 'FTP_TRANSFER_TYPE_BINARY', 'LongWord').SetUInt( $00000002);
9541:   //Const 'INTERNET_FLAG_TRANSFER_ASCII', 'LongWord').SetUInt( $00000001);
9542:   //Const 'INTERNET_FLAG_TRANSFER_BINARY', 'LongWord').SetUInt( $00000002);
9543:   //Const 'SECURITY_SET_MASK', 'LongWord').SetUInt( $00000001);
9544:   Const 'INTERNET_ERROR_MASK_INSERT_CDROM', 'LongWord').SetUInt( $1);
9545:   Const 'WININET_API_FLAG_ASYNC', 'LongWord').SetUInt( $00000001);
9546:   Const 'WININET_API_FLAG_SYNC', 'LongWord').SetUInt( $00000004);
9547:   Const 'WININET_API_FLAG_USE_CONTEXT', 'LongWord').SetUInt( $00000008);
9548:   Const 'INTERNET_NO_CALLBACK', 'LongInt').SetInt( 0);

```



```

9549: //PIInternetScheme', '^TInternetScheme // will not work');
9550: TInternetScheme', 'Integer');
9551: Const 'INTERNET_SCHEME_PARTIAL', 'LongInt').SetInt( - 2);
9552: Const 'INTERNET_SCHEME_UNKNOWN', 'LongInt').SetInt( - 1);
9553: Const 'INTERNET_SCHEME_DEFAULT', 'LongInt').SetInt( 0);
9554: Const 'INTERNET_SCHEME_FTP', 'LongInt').SetInt( 1);
9555: Const 'INTERNET_SCHEME_GOPHER', 'LongInt').SetInt( 2);
9556: Const 'INTERNET_SCHEME_HTTP', 'LongInt').SetInt( 3);
9557: Const 'INTERNET_SCHEME_HTTPS', 'LongInt').SetInt( 4);
9558: Const 'INTERNET_SCHEME_FILE', 'LongInt').SetInt( 5);
9559: Const 'INTERNET_SCHEME_NEWS', 'LongInt').SetInt( 6);
9560: Const 'INTERNET_SCHEME_MAILTO', 'LongInt').SetInt( 7);
9561: //Const 'INTERNET_SCHEME_FIRST', '').SetString( INTERNET_SCHEME_FTP);
9562: //Const 'INTERNET_SCHEME_LAST', '').SetString( INTERNET_SCHEME_MAILTO);
9563: //PIInternetAsyncResult', '^TInternetAsyncResult // will not work');
9564: TInternetAsyncResult', 'record dwResult : DWORD; dwError : DWORD; end');
9565: //PIInternetPrefetchStatus', '^TInternetPrefetchStatus // will not work');
9566: TInternetPrefetchStatus', 'record dwStatus : DWORD; dwSize : DWORD; end');
9567: Const 'INTERNET_PREFETCH_PROGRESS', 'LongInt').SetInt( 0);
9568: Const 'INTERNET_PREFETCH_COMPLETE', 'LongInt').SetInt( 1);
9569: Const 'INTERNET_PREFETCH_ABORTED', 'LongInt').SetInt( 2);
9570: INTERNET_PROXY_INFO', 'record dwAccessType:DWORD;lpszProxy:string; lpszProxyBypass : string; end');
9571: TInternetProxyInfo', 'INTERNET_PROXY_INFO');
9572: INTERNET_VERSION_INFO', 'record dwMajorVersion : DWORD; dwMinorVersion : DWORD; end');
9573: TInternetVersionInfo', 'INTERNET_VERSION_INFO');
9574: HTTP_VERSION_INFO', 'record dwMajorVersion : DWORD; dwMinorVersion : DWORD; end');
9575: THttpVersionInfo', 'HTTP_VERSION_INFO');
9576: INTERNET_CONNECTED_INFO', 'record dwConnectedState : DWORD; dwFlags : DWORD; end');
9577: TInternetConnectedInfo', 'INTERNET_CONNECTED_INFO');
9578: Const 'ISO_FORCE_DISCONNECTED', 'LongWord').SetUInt( $00000001);
9579:
9580: LPSTR', 'PChar');
9581: LPCSTR', 'PChar'); //LPCTSTR
9582: PLPSTR', 'PChar');
9583: URL_COMPONENTS', 'record dwStructSize : DWORD; lpszScheme : LPSTR'
9584: + 'R; dwSchemeLength : DWORD; nScheme : TInternetScheme; lpszHostName : LPSTR'
9585: + 'I; dwHostNameLength : DWORD; nPort : INTERNET_PORT; pad : WORD; lpszUserNam'
9586: + 'e : LPSTR; dwUserNameLength : DWORD; lpszPassword : LPSTR; dwPasswordLengt'
9587: + 'h : DWORD; lpszUrlPath : LPSTR; dwUrlPathLength : DWORD; lpszExtraInfo : L'
9588: + 'PSTR; dwExtraInfoLength : DWORD; end');
9589: TURLComponents', 'URL_COMPONENTS');
9590: INTERNET_CERTIFICATE_INFO', 'record ftExpiry : TFileTime; ftStar'
9591: + 't : TFileTime; lpszSubjectInfo : LPSTR; lpszIssuerInfo : LPSTR; lpszProtoc'
9592: + 'olName : LPSTR; lpszSignatureAlgName : LPSTR; lpszEncryptionAlgName : LPST'
9593: + 'R; dwKeySize : DWORD; end');
9594: TInternetCertificateInfo', 'INTERNET_CERTIFICATE_INFO');
9595: CL.AddDelphiFunction('Function InternetTimeFromSystemTime( const pst : TSystemTime; dwRFC : DWORD;
lpszTime : LPSTR; cbTime : DWORD) : BOOL');
9596: Const 'INTERNET_RFC1123_FORMAT', 'LongInt').SetInt( 0);
9597: Const 'INTERNET_RFC1123_BUFSIZE', 'LongInt').SetInt( 30);
9598: CL.AddDelphiFunction('Function InternetCrackUrl( lpszUrl : PChar; dwUrlLength, dwFlags : DWORD; var
lpUrlComponents : TURLComponents) : BOOL');
9599: Function InternetCreateUrl( var lpUrlComponents : TURLComponents; dwFlags : DWORD; lpszUrl : PChar; var
dwUrlLength : DWORD) : BOOL);
9600: Function InternetCanonicalizeUrl(lpszUrl:PChar;lpszBuffer:PChar;var
lpdwBufferLength:DWORD;dwFlags:DWORD):BOOL;
9601: Function InternetCombineUrl( lpszBaseUrl, lpszRelativeUrl : PChar; lpszBuffer : PChar; var
lpdwBufferLength : DWORD; dwFlags : DWORD) : BOOL);
9602: Const 'ICU_ESCAPE', 'LongWord').SetUInt( $80000000);
9603: Const 'ICU_USERNAME', 'LongWord').SetUInt( $40000000);
9604: Const 'ICU_NO_ENCODE', 'LongWord').SetUInt( $20000000);
9605: Const 'ICU_DECODE', 'LongWord').SetUInt( $10000000);
9606: Const 'ICU_NO_META', 'LongWord').SetUInt( $08000000);
9607: Const 'ICU_ENCODE_SPACES_ONLY', 'LongWord').SetUInt( $04000000);
9608: Const 'ICU_BROWSER_MODE', 'LongWord').SetUInt( $02000000);
9609: Function InternetOpen( lpszAgent : PChar; dwAccessType : DWORD; lpszProxy, lpszProxyBypass : PChar;
dwFlags : DWORD) : HINTERNET);
9610: Const 'INTERNET_OPEN_TYPE_PRECONFIG', 'LongInt').SetInt( 0);
9611: Const 'INTERNET_OPEN_TYPE_DIRECT', 'LongInt').SetInt( 1);
9612: Const 'INTERNET_OPEN_TYPE_PROXY', 'LongInt').SetInt( 3);
9613: Const 'INTERNET_OPEN_TYPE_PRECONFIG_WITH_NO_AUTOPROXY', 'LongInt').SetInt( 4);
9614: Const 'GATEWAY_INTERNET_ACCESS', 'LongInt').SetInt( 2);
9615: //Const 'CERN_PROXY_INTERNET_ACCESS', '').SetString( INTERNET_OPEN_TYPE_PROXY);
9616: Function InternetCloseHandle( hInet : HINTERNET) : BOOL);
9617: Function InternetConnect( hInet : HINTERNET; lpszServerName : PChar; nServerPort : INTERNET_PORT;
lpszUsername : PChar; lpszPassword : PChar; dwService : DWORD; dwFlags : DWORD; dwContext : DWORD) :
HINTERNET);
9618: //CL.AddDelphiFunction('Function InternetConnectA( hInet : HINTERNET; lpszServerName : PAnsiChar;
nServerPort : Const 'INTERNET_SERVICE_URL', 'LongInt').SetInt( 0);
9619: Const 'INTERNET_SERVICE_FTP', 'LongInt').SetInt( 1);
9620: Const 'INTERNET_SERVICE_GOPHER', 'LongInt').SetInt( 2);
9621: Const 'INTERNET_SERVICE_HTTP', 'LongInt').SetInt( 3);
9622: CL.AddDelphiFunction('Function InternetOpenUrl( hInet : HINTERNET; lpszUrl : PChar; lpszHeaders : PChar;
dwHeadersLength : DWORD; dwFlags : DWORD; dwContext : DWORD) : HINTERNET');
9623: //Const 'IRF_ASYNC', '').SetString( WININET_API_FLAG_ASYNC);
9624: //Const 'IRF_SYNC', '').SetString( WININET_API_FLAG_SYNC);
9625: //Const 'IRF_USE_CONTEXT', '').SetString( WININET_API_FLAG_USE_CONTEXT);
9626: Const 'IRF_NO_WAIT', 'LongWord').SetUInt( $00000008);
9627: //CL.AddDelphiFunction('Function InternetSetFilePointer( hFile : HINTERNET; lDistanceToMove : Longint;
pReserved : Pointer; dwMoveMethod, dwContext : DWORD) : DWORD');

```

```

9628: //CL.AddDelphiFunction('Function InternetWriteFile( hFile : HINTERNET; lpBuffer : Pointer;
dwNumberOfBytesToWrite : DWORD; var lpdwNumberOfBytesWritten : DWORD) : BOOL');
9629: Function InternetUnlockRequestFile( hLockRequestInfo : THANDLE) : BOOL');
9630: Const 'ISO_GLOBAL', 'LongWord').SetUInt( $00000001);
9631: Const 'ISO_REGISTRY', 'LongWord').SetUInt( $00000002);
9632: Const 'INTERNET_OPTION_CALLBACK', 'LongInt').SetInt( 1);
9633: Const 'INTERNET_OPTION_CONNECT_TIMEOUT', 'LongInt').SetInt( 2);
9634: Const 'INTERNET_OPTION_CONNECT_RETRIES', 'LongInt').SetInt( 3);
9635: Const 'INTERNET_OPTION_CONNECT_BACKOFF', 'LongInt').SetInt( 4);
9636: Const 'INTERNET_OPTION_SEND_TIMEOUT', 'LongInt').SetInt( 5);
9637: //Const 'INTERNET_OPTION_CONTROL_SEND_TIMEOUT', '').SetString( INTERNET_OPTION_SEND_TIMEOUT);
9638: Const 'INTERNET_OPTION_RECEIVE_TIMEOUT', 'LongInt').SetInt( 6);
9639: //Const 'INTERNET_OPTION_CONTROL_RECEIVE_TIMEOUT', '').SetString( INTERNET_OPTION_RECEIVE_TIMEOUT);
9640: Const 'INTERNET_OPTION_DATA_SEND_TIMEOUT', 'LongInt').SetInt( 7);
9641: Const 'INTERNET_OPTION_DATA_RECEIVE_TIMEOUT', 'LongInt').SetInt( 8);
9642: Const 'INTERNET_OPTION_HANDLE_TYPE', 'LongInt').SetInt( 9);
9643: Const 'INTERNET_OPTION_READ_BUFFER_SIZE', 'LongInt').SetInt( 12);
9644: Const 'INTERNET_OPTION_WRITE_BUFFER_SIZE', 'LongInt').SetInt( 13);
9645: Const 'INTERNET_OPTION_ASYNC_ID', 'LongInt').SetInt( 15);
9646: Const 'INTERNET_OPTION_ASYNC_PRIORITY', 'LongInt').SetInt( 16);
9647: Const 'INTERNET_OPTION_PARENT_HANDLE', 'LongInt').SetInt( 21);
9648: Const 'INTERNET_OPTION_KEEP_CONNECTION', 'LongInt').SetInt( 22);
9649: Const 'INTERNET_OPTION_REQUEST_FLAGS', 'LongInt').SetInt( 23);
9650: Const 'INTERNET_OPTION_EXTENDED_ERROR', 'LongInt').SetInt( 24);
9651: Const 'INTERNET_OPTION_OFFLINE_MODE', 'LongInt').SetInt( 26);
9652: Const 'INTERNET_OPTION_CACHE_STREAM_HANDLE', 'LongInt').SetInt( 27);
9653: Const 'INTERNET_OPTION_USERNAME', 'LongInt').SetInt( 28);
9654: Const 'INTERNET_OPTION_PASSWORD', 'LongInt').SetInt( 29);
9655: Const 'INTERNET_OPTION_ASYNC', 'LongInt').SetInt( 30);
9656: Const 'INTERNET_OPTION_SECURITY_FLAGS', 'LongInt').SetInt( 31);
9657: Const 'INTERNET_OPTION_SECURITY_CERTIFICATE_STRUCT', 'LongInt').SetInt( 32);
9658: Const 'INTERNET_OPTION_DATAFILE_NAME', 'LongInt').SetInt( 33);
9659: Const 'INTERNET_OPTION_URL', 'LongInt').SetInt( 34);
9660: Const 'INTERNET_OPTION_SECURITY_CERTIFICATE', 'LongInt').SetInt( 35);
9661: Const 'INTERNET_OPTION_SECURITY_KEY_BITNESS', 'LongInt').SetInt( 36);
9662: Const 'INTERNET_OPTION_REFRESH', 'LongInt').SetInt( 37);
9663: Const 'INTERNET_OPTION_PROXY', 'LongInt').SetInt( 38);
9664: Const 'INTERNET_OPTION_SETTINGS_CHANGED', 'LongInt').SetInt( 39);
9665: Const 'INTERNET_OPTION_VERSION', 'LongInt').SetInt( 40);
9666: Const 'INTERNET_OPTION_USER_AGENT', 'LongInt').SetInt( 41);
9667: Const 'INTERNET_OPTION_END_BROWSER_SESSION', 'LongInt').SetInt( 42);
9668: Const 'INTERNET_OPTION_PROXY_USERNAME', 'LongInt').SetInt( 43);
9669: Const 'INTERNET_OPTION_PROXY_PASSWORD', 'LongInt').SetInt( 44);
9670: Const 'INTERNET_OPTION_CONTEXT_VALUE', 'LongInt').SetInt( 45);
9671: Const 'INTERNET_OPTION_CONNECT_LIMIT', 'LongInt').SetInt( 46);
9672: Const 'INTERNET_OPTION_SECURITY_SELECT_CLIENT_CERT', 'LongInt').SetInt( 47);
9673: Const 'INTERNET_OPTION_POLICY', 'LongInt').SetInt( 48);
9674: Const 'INTERNET_OPTION_DISCONNECTED_TIMEOUT', 'LongInt').SetInt( 49);
9675: Const 'INTERNET_OPTION_CONNECTED_STATE', 'LongInt').SetInt( 50);
9676: Const 'INTERNET_OPTION_IDLE_STATE', 'LongInt').SetInt( 51);
9677: Const 'INTERNET_OPTION_OFFLINE_SEMANTICS', 'LongInt').SetInt( 52);
9678: Const 'INTERNET_OPTION_SECONDARY_CACHE_KEY', 'LongInt').SetInt( 53);
9679: Const 'INTERNET_OPTION_CALLBACK_FILTER', 'LongInt').SetInt( 54);
9680: Const 'INTERNET_OPTION_CONNECT_TIME', 'LongInt').SetInt( 55);
9681: Const 'INTERNET_OPTION_SEND_THROUGHPUT', 'LongInt').SetInt( 56);
9682: Const 'INTERNET_OPTION_RECEIVE_THROUGHPUT', 'LongInt').SetInt( 57);
9683: Const 'INTERNET_OPTION_REQUEST_PRIORITY', 'LongInt').SetInt( 58);
9684: Const 'INTERNET_OPTION_HTTP_VERSION', 'LongInt').SetInt( 59);
9685: Const 'INTERNET_OPTION_RESET_URLCACHE_SESSION', 'LongInt').SetInt( 60);
9686: Const 'INTERNET_OPTION_ERROR_MASK', 'LongInt').SetInt( 62);
9687: //Const 'INTERNET_FIRST_OPTION', '').SetString( INTERNET_OPTION_CALLBACK);
9688: //Const 'INTERNET_LAST_OPTION', '').SetString( INTERNET_OPTION_PROXY);
9689: Const 'INTERNET_PRIORITY_FOREGROUND', 'LongInt').SetInt( 1000);
9690: Const 'INTERNET_HANDLE_TYPE_INTERNET', 'LongInt').SetInt( 1);
9691: Const 'INTERNET_HANDLE_TYPE_CONNECT_FTP', 'LongInt').SetInt( 2);
9692: Const 'INTERNET_HANDLE_TYPE_CONNECT_GOPHER', 'LongInt').SetInt( 3);
9693: Const 'INTERNET_HANDLE_TYPE_CONNECT_HTTP', 'LongInt').SetInt( 4);
9694: Const 'INTERNET_HANDLE_TYPE_FTP_FIND', 'LongInt').SetInt( 5);
9695: Const 'INTERNET_HANDLE_TYPE_FTP_FIND_HTML', 'LongInt').SetInt( 6);
9696: Const 'INTERNET_HANDLE_TYPE_FTP_FILE', 'LongInt').SetInt( 7);
9697: Const 'INTERNET_HANDLE_TYPE_FTP_FILE_HTML', 'LongInt').SetInt( 8);
9698: Const 'INTERNET_HANDLE_TYPE_GOPHER_FIND', 'LongInt').SetInt( 9);
9699: Const 'INTERNET_HANDLE_TYPE_GOPHER_FIND_HTML', 'LongInt').SetInt( 10);
9700: Const 'INTERNET_HANDLE_TYPE_GOPHER_FILE', 'LongInt').SetInt( 11);
9701: Const 'INTERNET_HANDLE_TYPE_GOPHER_FILE_HTML', 'LongInt').SetInt( 12);
9702: Const 'INTERNET_HANDLE_TYPE_HTTP_REQUEST', 'LongInt').SetInt( 13);
9703: Const 'SECURITY_FLAG_SECURE', 'LongWord').SetUInt( $00000001);
9704: Const 'SECURITY_FLAG_SSL', 'LongWord').SetUInt( $00000002);
9705: Const 'SECURITY_FLAG_SSL3', 'LongWord').SetUInt( $00000004);
9706: Const 'SECURITY_FLAG_PCT', 'LongWord').SetUInt( $00000008);
9707: Const 'SECURITY_FLAG_PCT4', 'LongWord').SetUInt( $00000010);
9708: Const 'SECURITY_FLAG_IETFSSL4', 'LongWord').SetUInt( $00000020);
9709: Const 'SECURITY_FLAG_STRENGTH_WEAK', 'LongWord').SetUInt( $10000000);
9710: Const 'SECURITY_FLAG_STRENGTH_MEDIUM', 'LongWord').SetUInt( $40000000);
9711: Const 'SECURITY_FLAG_STRENGTH_STRONG', 'LongWord').SetUInt( $20000000);
9712: //Const 'SECURITY_FLAG_40BIT', '').SetString( SECURITY_FLAG_STRENGTH_WEAK);
9713: //Const 'SECURITY_FLAG_128BIT', '').SetString( SECURITY_FLAG_STRENGTH_STRONG);
9714: //Const 'SECURITY_FLAG_56BIT', '').SetString( SECURITY_FLAG_STRENGTH_MEDIUM);
9715: Const 'SECURITY_FLAG_UNKNOWNBIT', 'LongWord').SetUInt( $80000000);

```

```

9716: //Const'SECURITY_FLAG_NORMALBITNESS','').SetString( SECURITY_FLAG_40BIT);
9717: Const'SECURITY_FLAG_IGNORE_REVOCATION','LongWord').SetUInt( $00000080);
9718: Const'SECURITY_FLAG_IGNORE_UNKNOWN_CA','LongWord').SetUInt( $00000100);
9719: Const'SECURITY_FLAG_IGNORE_WRONG_USAGE','LongWord').SetUInt( $00000200);
9720: //Const'SECURITY_FLAG_IGNORE_CERT_CN_INVALID','').SetString( INTERNET_FLAG_IGNORE_CERT_CN_INVALID);
9721: //Const'SECURITY_FLAG_IGNORE_CERT_DATE_INVALID','').SetString( INTERNET_FLAG_IGNORE_CERT_DATE_INVALID);
9722: //Const'SECURITY_FLAG_IGNORE_REDIRECT_TO_HTTPS','').SetString( INTERNET_FLAG_IGNORE_REDIRECT_TO_HTTPS);
9723: //Const'SECURITY_FLAG_IGNORE_REDIRECT_TO_HTTP','').SetString( INTERNET_FLAG_IGNORE_REDIRECT_TO_HTTP);
9724: INTERNET_STATUS_CALLBACK', 'TObject');
9725: TfnInternetStatusCallback', 'INTERNET_STATUS_CALLBACK');
9726: Const'INTERNET_STATUS_RESOLVING_NAME','LongInt').SetInt( 10);
9727: Const'INTERNET_STATUS_NAME_RESOLVED','LongInt').SetInt( 11);
9728: Const'INTERNET_STATUS_CONNECTING_TO_SERVER','LongInt').SetInt( 20);
9729: Const'INTERNET_STATUS_CONNECTED_TO_SERVER','LongInt').SetInt( 21);
9730: Const'INTERNET_STATUS_SENDING_REQUEST','LongInt').SetInt( 30);
9731: Const'INTERNET_STATUS_REQUEST_SENT','LongInt').SetInt( 31);
9732: Const'INTERNET_STATUS_RECEIVING_RESPONSE','LongInt').SetInt( 40);
9733: Const'INTERNET_STATUS_RESPONSE_RECEIVED','LongInt').SetInt( 41);
9734: Const'INTERNET_STATUS_CTL_RESPONSE_RECEIVED','LongInt').SetInt( 42);
9735: Const'INTERNET_STATUS_PREFETCH','LongInt').SetInt( 43);
9736: Const'INTERNET_STATUS_CLOSING_CONNECTION','LongInt').SetInt( 50);
9737: Const'INTERNET_STATUS_CONNECTION_CLOSED','LongInt').SetInt( 51);
9738: Const'INTERNET_STATUS_HANDLE_CREATED','LongInt').SetInt( 60);
9739: Const'INTERNET_STATUS_HANDLE_CLOSING','LongInt').SetInt( 70);
9740: Const'INTERNET_STATUS_REQUEST_COMPLETE','LongInt').SetInt( 100);
9741: Const'INTERNET_STATUS_REDIRECT','LongInt').SetInt( 110);
9742: Const'INTERNET_STATUS_INTERMEDIATE_RESPONSE','LongInt').SetInt( 120);
9743: Const'INTERNET_STATUS_STATE_CHANGE','LongInt').SetInt( 200);
9744: Const'INTERNET_STATE_CONNECTED','LongWord').SetUInt( $00000001);
9745: Const'INTERNET_STATE_DISCONNECTED','LongWord').SetUInt( $00000002);
9746: Const'INTERNET_STATE_DISCONNECTED_BY_USER','LongWord').SetUInt( $00000010);
9747: Const'INTERNET_STATE_IDLE','LongWord').SetUInt( $00000100);
9748: Const'INTERNET_STATE_BUSY','LongWord').SetUInt( $00000200);
9749: Const'INTERNET_INVALID_STATUS_CALLBACK','LongInt').SetInt( ( - 1 ));
9750: Function FtpFindFirstFile( hConnect : HINTERNET; lpszSearchFile : PChar; var lpFindFileData :
Twin32FindData; dwFlags : DWORD; dwContext : DWORD) : HINTERNET);
9751: Function FtpGetFile( hConnect : HINTERNET; lpszRemoteFile : PChar; lpszNewFile : PChar; fFailIfExists :
BOOL; dwFlagsAndAttributes : DWORD; dwFlags : DWORD; dwContext : DWORD) : BOOL);
9752: Const'MAX_GOPHER_DISPLAY_TEXT','LongInt').SetInt( 128);
9753: Const'MAX_GOPHER_SELECTOR_TEXT','LongInt').SetInt( 256);
9754: //Const'MAX_GOPHER_HOST_NAME','').SetString( INTERNET_MAX_HOST_NAME_LENGTH);
9755: //PGopherFindDataA', 'GOPHER_FIND_DATAA // will not work');
9756: //PGopherFindDataW', 'GOPHER_FIND_DATAW // will not work');
9757: //PGopherFindData', 'PGopherFindDataA');
9758: Const'GOPHER_TYPE_TEXT_FILE','LongWord').SetUInt( $00000001);
9759: Const'GOPHER_TYPE_DIRECTORY','LongWord').SetUInt( $00000002);
9760: Const'GOPHER_TYPE_CSO','LongWord').SetUInt( $00000004);
9761: Const'GOPHER_TYPE_ERROR','LongWord').SetUInt( $00000008);
9762: Const'GOPHER_TYPE_MAC_BINHEX','LongWord').SetUInt( $00000010);
9763: Const'GOPHER_TYPE_DOS_ARCHIVE','LongWord').SetUInt( $00000020);
9764: Const'GOPHER_TYPE_UNIX_UUENCODED','LongWord').SetUInt( $00000040);
9765: Const'GOPHER_TYPE_INDEX_SERVER','LongWord').SetUInt( $00000080);
9766: Const'GOPHER_TYPE_TELNET','LongWord').SetUInt( $00000100);
9767: Const'GOPHER_TYPE_BINARY','LongWord').SetUInt( $00000200);
9768: Const'GOPHER_TYPE_REDUNDANT','LongWord').SetUInt( $00000400);
9769: Const'GOPHER_TYPE_TN3270','LongWord').SetUInt( $00000800);
9770: Const'GOPHER_TYPE_GIF','LongWord').SetUInt( $00001000);
9771: Const'GOPHER_TYPE_IMAGE','LongWord').SetUInt( $00002000);
9772: Const'GOPHER_TYPE_BITMAP','LongWord').SetUInt( $00004000);
9773: Const'GOPHER_TYPE_MOVIE','LongWord').SetUInt( $00008000);
9774: Const'GOPHER_TYPE_SOUND','LongWord').SetUInt( $00010000);
9775: Const'GOPHER_TYPE_HTML','LongWord').SetUInt( $00020000);
9776: Const'GOPHER_TYPE_PDF','LongWord').SetUInt( $00040000);
9777: Const'GOPHER_TYPE_CALENDAR','LongWord').SetUInt( $00080000);
9778: Const'GOPHER_TYPE_INLINE','LongWord').SetUInt( $00100000);
9779: Const'GOPHER_TYPE_UNKNOWN','LongWord').SetUInt( $20000000);
9780: Const'GOPHER_TYPE_ASK','LongWord').SetUInt( $40000000);
9781: Const'GOPHER_TYPE_GOPHER_PLUS','LongWord').SetUInt( $80000000);
9782: GOPHER_ADMIN_ATTRIBUTE_TYPE', 'record Comment : LPCSTR; EmailAddress : LPCSTR; end');
9783: TGopherAdminAttributeType', 'GOPHER_ADMIN_ATTRIBUTE_TYPE');
9784: // LPGOPHER_ADMIN_ATTRIBUTE_TYPE', 'PGopherAdminAttributeType');
9785: GOPHER_MOD_DATE_ATTRIBUTE_TYPE', 'record DateAndTime : TFileTime; end');
9786: TGopherModDateAttributeType', 'GOPHER_MOD_DATE_ATTRIBUTE_TYPE');
9787: GOPHER_TTL_ATTRIBUTE_TYPE', 'record Ttl : DWORD; end');
9788: TGopherTtlAttributeType', 'GOPHER_TTL_ATTRIBUTE_TYPE');
9789: //LPGOPHER_TTL_ATTRIBUTE_TYPE', 'PGopherTtlAttributeType');
9790: GOPHER_SCORE_ATTRIBUTE_TYPE', 'record Score : Integer; end');
9791: TGopherScoreAttributeType', 'GOPHER_SCORE_ATTRIBUTE_TYPE');
9792: //LPGOPHER_SCORE_ATTRIBUTE_TYPE', 'PGopherScoreAttributeType');
9793: GOPHER_SCORE_RANGE_ATTRIBUTE_TYPE', 'record LowerBound : Integer; UpperBound : Integer; end');
9794: TGopherScoreRangeAttributeType', 'GOPHER_SCORE_RANGE_ATTRIBUTE_TYPE');
9795: GOPHER_SITE_ATTRIBUTE_TYPE', 'record Site : LPCSTR; end');
9796: TGopherSiteAttributeType', 'GOPHER_SITE_ATTRIBUTE_TYPE');
9797: //LPGOPHER_SITE_ATTRIBUTE_TYPE', 'PGopherSiteAttributeType');
9798: GOPHER_ORGANIZATION_ATTRIBUTE_TYPE', 'record Organization : LPCSTR; end');
9799: TGopherOrganizationAttributeType', 'GOPHER_ORGANIZATION_ATTRIBUTE_TYPE');
9800: GOPHER_LOCATION_ATTRIBUTE_TYPE', 'record Location : LPCSTR; end');
9801: TGopherLocationAttributeType', 'GOPHER_LOCATION_ATTRIBUTE_TYPE');
9802: // LPGOPHER_LOCATION_ATTRIBUTE_TYPE', 'PGopherLocationAttributeType');

```



```

9803: GOPHER_GEOGRAPHICAL_LOCATION_ATTRIBUTE_TYPE', 'record DegreesNor'
9804:   + 'th : Integer; MinutesNorth : Integer; SecondsNorth : Integer; DegreesEast '
9805:   + ' : Integer; MinutesEast : Integer; SecondsEast : Integer; end');
9806: TGopherGeographicalLocationAttributeType', 'GOPHER_GEOGRAPHICAL_LOCATION_ATTRIBUTE_TYPE');
9807: //LPGOPHER_GEOGRAPHICAL_LOCATION_ATTRIBUTE_TYPE', 'PGopherGeographicalLocationAttributeType');
9808: GOPHER_TIMEZONE_ATTRIBUTE_TYPE', 'record Zone : Integer; end');
9809: TGopherTimezoneAttributeType', 'GOPHER_TIMEZONE_ATTRIBUTE_TYPE');
9810: //LPGOPHER_TIMEZONE_ATTRIBUTE_TYPE', 'PGopherTimezoneAttributeType');
9811: GOPHER_PROVIDER_ATTRIBUTE_TYPE', 'record Provider : LPCSTR; end');
9812: TGopherProviderAttributeType', 'GOPHER_PROVIDER_ATTRIBUTE_TYPE');
9813: //LPGOPHER_PROVIDER_ATTRIBUTE_TYPE', 'PGopherProviderAttributeType');
9814: GOPHER_VERSION_ATTRIBUTE_TYPE', 'record Version : LPCSTR; end');
9815: TGopherVersionAttributeType', 'GOPHER_VERSION_ATTRIBUTE_TYPE');
9816: //LPGOPHER_VERSION_ATTRIBUTE_TYPE', 'PGopherVersionAttributeType');
9817: GOPHER_ABSTRACT_ATTRIBUTE_TYPE', 'record ShortAbstract : LPCSTR; AbstractFile : LPCSTR; end');
9818: TGopherAbstractAttributeType', 'GOPHER_ABSTRACT_ATTRIBUTE_TYPE');
9819: //LPGOPHER_ABSTRACT_ATTRIBUTE_TYPE', 'PGopherAbstractAttributeType');
9820: GOPHER_VIEW_ATTRIBUTE_TYPE', 'record ContentType : LPCSTR; Language : LPCSTR; Size : DWORD; end');
9821: TGopherViewAttributeType', 'GOPHER_VIEW_ATTRIBUTE_TYPE');
9822: //LPGOPHER_VIEW_ATTRIBUTE_TYPE', 'PGopherViewAttributeType');
9823: GOPHER_VERONICA_ATTRIBUTE_TYPE', 'record TreeWalk : BOOL; end');
9824: TGopherVeronicaAttributeType', 'GOPHER_VERONICA_ATTRIBUTE_TYPE');
9825: //LPGOPHER_VERONICA_ATTRIBUTE_TYPE', 'PGopherVeronicaAttributeType');
9826: GOPHER_ASK_ATTRIBUTE_TYPE', 'record QuestionType : LPCSTR; QuestionText : LPCSTR; end');
9827: TGopherAskAttributeType', 'GOPHER_ASK_ATTRIBUTE_TYPE');
9828: //LPGOPHER_ASK_ATTRIBUTE_TYPE', 'PGopherAskAttributeType');
9829: GOPHER_UNKNOWN_ATTRIBUTE_TYPE', 'record Text : LPCSTR; end');
9830: TGopherUnknownAttributeType', 'GOPHER_UNKNOWN_ATTRIBUTE_TYPE');
9831: //LPGOPHER_UNKNOWN_ATTRIBUTE_TYPE', 'PGopherUnknownAttributeType');
9832: Const 'MAX_GOPHER_CATEGORY_NAME', 'LongInt').SetInt( 128);
9833: Const 'MAX_GOPHER_ATTRIBUTE_NAME', 'LongInt').SetInt( 128);
9834: Const 'MIN_GOPHER_ATTRIBUTE_LENGTH', 'LongInt').SetInt( 256);
9835: Const 'GOPHER_INFO_CATEGORY', 'String').SetString( '+INFO');
9836: Const 'GOPHER_ADMIN_CATEGORY', 'String').SetString( '+ADMIN');
9837: Const 'GOPHER_VIEWS_CATEGORY', 'String').SetString( '+VIEWS');
9838: Const 'GOPHER_ABSTRACT_CATEGORY', 'String').SetString( '+ABSTRACT');
9839: Const 'GOPHER_VERONICA_CATEGORY', 'String').SetString( '+VERONICA');
9840: Const 'GOPHER_ADMIN_ATTRIBUTE', 'String').SetString( 'Admin');
9841: Const 'GOPHER_MOD_DATE_ATTRIBUTE', 'String').SetString( 'Mod-Date');
9842: Const 'GOPHER_TTL_ATTRIBUTE', 'String').SetString( 'TTL');
9843: Const 'GOPHER_SCORE_ATTRIBUTE', 'String').SetString( 'Score');
9844: Const 'GOPHER_RANGE_ATTRIBUTE', 'String').SetString( 'Score-range');
9845: Const 'GOPHER_SITE_ATTRIBUTE', 'String').SetString( 'Site');
9846: Const 'GOPHER_ORG_ATTRIBUTE', 'String').SetString( 'Org');
9847: Const 'GOPHER_LOCATION_ATTRIBUTE', 'String').SetString( 'Loc');
9848: Const 'GOPHER_GEOG_ATTRIBUTE', 'String').SetString( 'Geog');
9849: Const 'GOPHER_TIMEZONE_ATTRIBUTE', 'String').SetString( 'TZ');
9850: Const 'GOPHER_PROVIDER_ATTRIBUTE', 'String').SetString( 'Provider');
9851: Const 'GOPHER_VERSION_ATTRIBUTE', 'String').SetString( 'Version');
9852: Const 'GOPHER_ABSTRACT_ATTRIBUTE', 'String').SetString( 'Abstract');
9853: Const 'GOPHER_VIEW_ATTRIBUTE', 'String').SetString( 'View');
9854: Const 'GOPHER_TREEWALK_ATTRIBUTE', 'String').SetString( 'treewalk');
9855: Const 'GOPHER_ATTRIBUTE_ID_BASE', 'LongWord').SetUInt( $abcccc00);
9856: Const 'GOPHER_CATEGORY_ID_ALL', 'LongInt').SetInt( GOPHER_ATTRIBUTE_ID_BASE + 1);
9857: Const 'GOPHER_CATEGORY_ID_INFO', 'LongInt').SetInt( GOPHER_ATTRIBUTE_ID_BASE + 2);
9858: Const 'GOPHER_CATEGORY_ID_ADMIN', 'LongInt').SetInt( GOPHER_ATTRIBUTE_ID_BASE + 3);
9859: Const 'GOPHER_CATEGORY_ID_VIEWS', 'LongInt').SetInt( GOPHER_ATTRIBUTE_ID_BASE + 4);
9860: Const 'GOPHER_CATEGORY_ID_ABSTRACT', 'LongInt').SetInt( GOPHER_ATTRIBUTE_ID_BASE + 5);
9861: Const 'GOPHER_CATEGORY_ID_VERONICA', 'LongInt').SetInt( GOPHER_ATTRIBUTE_ID_BASE + 6);
9862: Const 'GOPHER_CATEGORY_ID_ASK', 'LongInt').SetInt( GOPHER_ATTRIBUTE_ID_BASE + 7);
9863: Const 'GOPHER_CATEGORY_ID_UNKNOWN', 'LongInt').SetInt( GOPHER_ATTRIBUTE_ID_BASE + 8);
9864: Const 'GOPHER_ATTRIBUTE_ID_ALL', 'LongInt').SetInt( GOPHER_ATTRIBUTE_ID_BASE + 9);
9865: Const 'GOPHER_ATTRIBUTE_ID_ADMIN', 'LongInt').SetInt( GOPHER_ATTRIBUTE_ID_BASE + 10);
9866: Const 'GOPHER_ATTRIBUTE_ID_MOD_DATE', 'LongInt').SetInt( GOPHER_ATTRIBUTE_ID_BASE + 11);
9867: Const 'GOPHER_ATTRIBUTE_ID_TTL', 'LongInt').SetInt( GOPHER_ATTRIBUTE_ID_BASE + 12);
9868: Const 'GOPHER_ATTRIBUTE_ID_SCORE', 'LongInt').SetInt( GOPHER_ATTRIBUTE_ID_BASE + 13);
9869: Const 'GOPHER_ATTRIBUTE_ID_RANGE', 'LongInt').SetInt( GOPHER_ATTRIBUTE_ID_BASE + 14);
9870: Const 'GOPHER_ATTRIBUTE_ID_SITE', 'LongInt').SetInt( GOPHER_ATTRIBUTE_ID_BASE + 15);
9871: Const 'GOPHER_ATTRIBUTE_ID_ORG', 'LongInt').SetInt( GOPHER_ATTRIBUTE_ID_BASE + 16);
9872: Const 'GOPHER_ATTRIBUTE_ID_LOCATION', 'LongInt').SetInt( GOPHER_ATTRIBUTE_ID_BASE + 17);
9873: Const 'GOPHER_ATTRIBUTE_ID_GEOG', 'LongInt').SetInt( GOPHER_ATTRIBUTE_ID_BASE + 18);
9874: Const 'GOPHER_ATTRIBUTE_ID_TIMEZONE', 'LongInt').SetInt( GOPHER_ATTRIBUTE_ID_BASE + 19);
9875: Const 'GOPHER_ATTRIBUTE_ID_PROVIDER', 'LongInt').SetInt( GOPHER_ATTRIBUTE_ID_BASE + 20);
9876: Const 'GOPHER_ATTRIBUTE_ID_VERSION', 'LongInt').SetInt( GOPHER_ATTRIBUTE_ID_BASE + 21);
9877: Const 'GOPHER_ATTRIBUTE_ID_ABSTRACT', 'LongInt').SetInt( GOPHER_ATTRIBUTE_ID_BASE + 22);
9878: Const 'GOPHER_ATTRIBUTE_ID_VIEW', 'LongInt').SetInt( GOPHER_ATTRIBUTE_ID_BASE + 23);
9879: Const 'GOPHER_ATTRIBUTE_ID_TREEWALK', 'LongInt').SetInt( GOPHER_ATTRIBUTE_ID_BASE + 24);
9880: Const 'GOPHER_ATTRIBUTE_ID_UNKNOWN', 'LongInt').SetInt( GOPHER_ATTRIBUTE_ID_BASE + 25);
9881: Function GopherCreateLocator( lpszHost : PChar; nServerPort : INTERNET_PORT; lpszDisplayString : PChar;
lpszSelectorString : PChar; dwGopherType : DWORD; lpszLocator : PChar; var lpdwBufferLength : DWORD) :
BOOL);
9882: Const 'HTTP_MAJOR_VERSION', 'LongInt').SetInt( 1);
9883: Const 'HTTP_MINOR_VERSION', 'LongInt').SetInt( 0);
9884: Const 'HTTP_VERSION', 'String').SetString( 'HTTP/1.0');
9885: Const 'HTTP_QUERY_MIME_VERSION', 'LongInt').SetInt( 0);
9886: Const 'HTTP_QUERY_CONTENT_TYPE', 'LongInt').SetInt( 1);
9887: Const 'HTTP_QUERY_CONTENT_TRANSFER_ENCODING', 'LongInt').SetInt( 2);
9888: Const 'HTTP_QUERY_CONTENT_ID', 'LongInt').SetInt( 3);
9889: Const 'HTTP_QUERY_CONTENT_DESCRIPTION', 'LongInt').SetInt( 4);

```



```

9890: Const 'HTTP_QUERY_CONTENT_LENGTH', 'LongInt').SetInt( 5);
9891: Const 'HTTP_QUERY_CONTENT_LANGUAGE', 'LongInt').SetInt( 6);
9892: Const 'HTTP_QUERY_ALLOW', 'LongInt').SetInt( 7);
9893: Const 'HTTP_QUERY_PUBLIC', 'LongInt').SetInt( 8);
9894: Const 'HTTP_QUERY_DATE', 'LongInt').SetInt( 9);
9895: Const 'HTTP_QUERY_EXPIRES', 'LongInt').SetInt( 10);
9896: Const 'HTTP_QUERY_LAST_MODIFIED', 'LongInt').SetInt( 11);
9897: Const 'HTTP_QUERY_MESSAGE_ID', 'LongInt').SetInt( 12);
9898: Const 'HTTP_QUERY_URI', 'LongInt').SetInt( 13);
9899: Const 'HTTP_QUERY_DERIVED_FROM', 'LongInt').SetInt( 14);
9900: Const 'HTTP_QUERY_COST', 'LongInt').SetInt( 15);
9901: Const 'HTTP_QUERY_LINK', 'LongInt').SetInt( 16);
9902: Const 'HTTP_QUERY_PRAGMA', 'LongInt').SetInt( 17);
9903: Const 'HTTP_QUERY_VERSION', 'LongInt').SetInt( 18);
9904: Const 'HTTP_QUERY_STATUS_CODE', 'LongInt').SetInt( 19);
9905: Const 'HTTP_QUERY_STATUS_TEXT', 'LongInt').SetInt( 20);
9906: Const 'HTTP_QUERY_RAW_HEADERS', 'LongInt').SetInt( 21);
9907: Const 'HTTP_QUERY_RAW_HEADERS_CRLF', 'LongInt').SetInt( 22);
9908: Const 'HTTP_QUERY_CONNECTION', 'LongInt').SetInt( 23);
9909: Const 'HTTP_QUERY_ACCEPT', 'LongInt').SetInt( 24);
9910: Const 'HTTP_QUERY_ACCEPT_CHARSET', 'LongInt').SetInt( 25);
9911: Const 'HTTP_QUERY_ACCEPT_ENCODING', 'LongInt').SetInt( 26);
9912: Const 'HTTP_QUERY_ACCEPT_LANGUAGE', 'LongInt').SetInt( 27);
9913: Const 'HTTP_QUERY_AUTHORIZATION', 'LongInt').SetInt( 28);
9914: Const 'HTTP_QUERY_CONTENT_ENCODING', 'LongInt').SetInt( 29);
9915: Const 'HTTP_QUERY_FORWARDED', 'LongInt').SetInt( 30);
9916: Const 'HTTP_QUERY_FROM', 'LongInt').SetInt( 31);
9917: Const 'HTTP_QUERY_IF_MODIFIED_SINCE', 'LongInt').SetInt( 32);
9918: Const 'HTTP_QUERY_LOCATION', 'LongInt').SetInt( 33);
9919: Const 'HTTP_QUERY_ORIG_URI', 'LongInt').SetInt( 34);
9920: Const 'HTTP_QUERY_REFERER', 'LongInt').SetInt( 35);
9921: Const 'HTTP_QUERY_RETRY_AFTER', 'LongInt').SetInt( 36);
9922: Const 'HTTP_QUERY_SERVER', 'LongInt').SetInt( 37);
9923: Const 'HTTP_QUERY_TITLE', 'LongInt').SetInt( 38);
9924: Const 'HTTP_QUERY_USER_AGENT', 'LongInt').SetInt( 39);
9925: Const 'HTTP_QUERY_WWW_AUTHENTICATE', 'LongInt').SetInt( 40);
9926: Const 'HTTP_QUERY_PROXY_AUTHENTICATE', 'LongInt').SetInt( 41);
9927: Const 'HTTP_QUERY_ACCEPT_RANGES', 'LongInt').SetInt( 42);
9928: Const 'HTTP_QUERY_SET_COOKIE', 'LongInt').SetInt( 43);
9929: Const 'HTTP_QUERY_COOKIE', 'LongInt').SetInt( 44);
9930: Const 'HTTP_QUERY_REQUEST_METHOD', 'LongInt').SetInt( 45);
9931: Const 'HTTP_QUERY_REFRESH', 'LongInt').SetInt( 46);
9932: Const 'HTTP_QUERY_CONTENT_DISPOSITION', 'LongInt').SetInt( 47);
9933: Const 'HTTP_QUERY_AGE', 'LongInt').SetInt( 48);
9934: Const 'HTTP_QUERY_CACHE_CONTROL', 'LongInt').SetInt( 49);
9935: Const 'HTTP_QUERY_CONTENT_BASE', 'LongInt').SetInt( 50);
9936: Const 'HTTP_QUERY_CONTENT_LOCATION', 'LongInt').SetInt( 51);
9937: Const 'HTTP_QUERY_CONTENT_MD5', 'LongInt').SetInt( 52);
9938: Const 'HTTP_QUERY_CONTENT_RANGE', 'LongInt').SetInt( 53);
9939: Const 'HTTP_QUERY_ETAG', 'LongInt').SetInt( 54);
9940: Const 'HTTP_QUERY_HOST', 'LongInt').SetInt( 55);
9941: Const 'HTTP_QUERY_IF_MATCH', 'LongInt').SetInt( 56);
9942: Const 'HTTP_QUERY_IF_NONE_MATCH', 'LongInt').SetInt( 57);
9943: Const 'HTTP_QUERY_IF_RANGE', 'LongInt').SetInt( 58);
9944: Const 'HTTP_QUERY_IF_UNMODIFIED_SINCE', 'LongInt').SetInt( 59);
9945: Const 'HTTP_QUERY_MAX_FORWARDS', 'LongInt').SetInt( 60);
9946: Const 'HTTP_QUERY_PROXY_AUTHORIZATION', 'LongInt').SetInt( 61);
9947: Const 'HTTP_QUERY_RANGE', 'LongInt').SetInt( 62);
9948: Const 'HTTP_QUERY_TRANSFER_ENCODING', 'LongInt').SetInt( 63);
9949: Const 'HTTP_QUERY_UPGRADE', 'LongInt').SetInt( 64);
9950: Const 'HTTP_QUERY_VARY', 'LongInt').SetInt( 65);
9951: Const 'HTTP_QUERY_VIA', 'LongInt').SetInt( 66);
9952: Const 'HTTP_QUERY_WARNING', 'LongInt').SetInt( 67);
9953: Const 'HTTP_QUERY_MAX', 'LongInt').SetInt( 67);
9954: Const 'HTTP_QUERY_CUSTOM', 'LongInt').SetInt( 65535);
9955: Const 'HTTP_QUERY_FLAG_REQUEST_HEADERS', 'LongWord').SetUInt( $80000000);
9956: Const 'HTTP_QUERY_FLAG_SYSTEMTIME', 'LongWord').SetUInt( $40000000);
9957: Const 'HTTP_QUERY_FLAG_NUMBER', 'LongWord').SetUInt( $20000000);
9958: Const 'HTTP_QUERY_FLAG_COALESCE', 'LongWord').SetUInt( $10000000);
9959: Const 'HTTP_STATUS_CONTINUE', 'LongInt').SetInt( 100);
9960: Const 'HTTP_STATUS_SWITCH_PROTOCOLS', 'LongInt').SetInt( 101);
9961: Const 'HTTP_STATUS_OK', 'LongInt').SetInt( 200);
9962: Const 'HTTP_STATUS_CREATED', 'LongInt').SetInt( 201);
9963: Const 'HTTP_STATUS_ACCEPTED', 'LongInt').SetInt( 202);
9964: Const 'HTTP_STATUS_PARTIAL', 'LongInt').SetInt( 203);
9965: Const 'HTTP_STATUS_NO_CONTENT', 'LongInt').SetInt( 204);
9966: Const 'HTTP_STATUS_RESET_CONTENT', 'LongInt').SetInt( 205);
9967: Const 'HTTP_STATUS_PARTIAL_CONTENT', 'LongInt').SetInt( 206);
9968: Const 'HTTP_STATUS_AMBIGUOUS', 'LongInt').SetInt( 300);
9969: Const 'HTTP_STATUS_MOVED', 'LongInt').SetInt( 301);
9970: Const 'HTTP_STATUS_REDIRECT', 'LongInt').SetInt( 302);
9971: Const 'HTTP_STATUS_REDIRECT_METHOD', 'LongInt').SetInt( 303);
9972: Const 'HTTP_STATUS_NOT_MODIFIED', 'LongInt').SetInt( 304);
9973: Const 'HTTP_STATUS_USE_PROXY', 'LongInt').SetInt( 305);
9974: Const 'HTTP_STATUS_REDIRECT_KEEP_VERB', 'LongInt').SetInt( 307);
9975: Const 'HTTP_STATUS_BAD_REQUEST', 'LongInt').SetInt( 400);
9976: Const 'HTTP_STATUS_DENIED', 'LongInt').SetInt( 401);
9977: Const 'HTTP_STATUS_PAYMENT_REQ', 'LongInt').SetInt( 402);
9978: Const 'HTTP_STATUS_FORBIDDEN', 'LongInt').SetInt( 403);

```

```

9979: Const 'HTTP_STATUS_NOT_FOUND', 'LongInt').SetInt( 404);
9980: Const 'HTTP_STATUS_BAD_METHOD', 'LongInt').SetInt( 405);
9981: Const 'HTTP_STATUS_NONE_ACCEPTABLE', 'LongInt').SetInt( 406);
9982: Const 'HTTP_STATUS_PROXY_AUTH_REQ', 'LongInt').SetInt( 407);
9983: Const 'HTTP_STATUS_REQUEST_TIMEOUT', 'LongInt').SetInt( 408);
9984: Const 'HTTP_STATUS_CONFLICT', 'LongInt').SetInt( 409);
9985: Const 'HTTP_STATUS_GONE', 'LongInt').SetInt( 410);
9986: Const 'HTTP_STATUS_AUTH_REFUSED', 'LongInt').SetInt( 411);
9987: Const 'HTTP_STATUS_PRECOND_FAILED', 'LongInt').SetInt( 412);
9988: Const 'HTTP_STATUS_REQUEST_TOO_LARGE', 'LongInt').SetInt( 413);
9989: Const 'HTTP_STATUS_URI_TOO_LONG', 'LongInt').SetInt( 414);
9990: Const 'HTTP_STATUS_UNSUPPORTED_MEDIA', 'LongInt').SetInt( 415);
9991: Const 'HTTP_STATUS_SERVER_ERROR', 'LongInt').SetInt( 500);
9992: Const 'HTTP_STATUS_NOT_SUPPORTED', 'LongInt').SetInt( 501);
9993: Const 'HTTP_STATUS_BAD_GATEWAY', 'LongInt').SetInt( 502);
9994: Const 'HTTP_STATUS_SERVICE_UNAVAIL', 'LongInt').SetInt( 503);
9995: Const 'HTTP_STATUS_GATEWAY_TIMEOUT', 'LongInt').SetInt( 504);
9996: Const 'HTTP_STATUS_VERSION_NOT_SUP', 'LongInt').SetInt( 505);
9997: Const 'HTTP_ADDRREQ_INDEX_MASK', 'LongWord').SetUInt( $0000FFFF);
9998: Const 'HTTP_ADDRREQ_FLAGS_MASK', 'LongWord').SetUInt( $FFFF0000);
9999: Const 'HTTP_ADDRREQ_FLAG_ADD_IF_NEW', 'LongWord').SetUInt( $10000000);
10000: Const 'HTTP_ADDRREQ_FLAG_ADD', 'LongWord').SetUInt( $20000000);
10001: Const 'HTTP_ADDRREQ_FLAG_COALESCE_WITH_COMMA', 'LongWord').SetUInt( $40000000);
10002: Const 'HTTP_ADDRREQ_FLAG_COALESCE_WITH_SEMICOLON', 'LongWord').SetUInt( $01000000);
10003: //Const 'HTTP_ADDRREQ_FLAG_COALESCE', 'LongWord').SetString( HTTP_ADDRREQ_FLAG_COALESCE_WITH_COMMA);
10004: Const 'HTTP_ADDRREQ_FLAG_REPLACE', 'LongWord').SetUInt( $80000000);
10005: CL.AddDelphiFunction('Function HttpSendRequest( hRequest : HINTERNET; lpszHeaders : PChar;
dwHeadersLength : DWORD; lpOptional : Tobject; dwOptionalLength : DWORD) : BOOL');
10006: Const 'HSR_INITIATE', 'LongWord').SetUInt( $00000008);
10007: Const 'HSR_DOWNLOAD', 'LongWord').SetUInt( $00000010);
10008: Const 'HSR_CHUNKED', 'LongWord').SetUInt( $00000020);
10009: INTERNET_CACHE_ENTRY_INFOA, 'record dwStructSize : DWORD; lpszS'
10010: + 'ourceUrlName : PChar; lpszLocalFileName : PChar; CacheEntryType : '
10011: + 'DWORD; dwUseCount : DWORD; dwHitRate : DWORD; dwSizeLow : DWORD; dwSizeHig'
10012: + 'h : DWORD; LastModifiedTime : TFileTime; ExpireTime : TFileTime; LastAcces'
10013: + 'sTime : TFileTime; LastSyncTime : TFileTime; lpHeaderInfo : BYTE; dwHeade'
10014: + 'rInfoSize : DWORD; lpszFileExtension : PChar; dwReserved : DWORD; end');
10015: TINTERNETCACHENTRYINFO, 'INTERNET_CACHE_ENTRY_INFOA');
10016: Const 'FLAGS_ERROR_UI_FILTER_FOR_ERRORS', 'LongWord').SetUInt( $01);
10017: Const 'FLAGS_ERROR_UI_FLAGS_CHANGE_OPTIONS', 'LongWord').SetUInt( $02);
10018: Const 'FLAGS_ERROR_UI_FLAGS_GENERATE_DATA', 'LongWord').SetUInt( $04);
10019: Const 'FLAGS_ERROR_UI_FLAGS_NO_UI', 'LongWord').SetUInt( $08);
10020: Const 'FLAGS_ERROR_UI_SERIALIZE_DIALOGS', 'LongWord').SetUInt( $10);
10021: Const 'INTERNET_ERROR_BASE', 'LongInt').SetInt( 12000);
10022: Const 'ERROR_INTERNET_OUT_OF_HANDLES', 'LongInt').SetInt( INTERNET_ERROR_BASE + 1);
10023: Const 'ERROR_INTERNET_TIMEOUT', 'LongInt').SetInt( INTERNET_ERROR_BASE + 2);
10024: Const 'ERROR_INTERNET_EXTENDED_ERROR', 'LongInt').SetInt( INTERNET_ERROR_BASE + 3);
10025: Const 'ERROR_INTERNET_INTERNAL_ERROR', 'LongInt').SetInt( INTERNET_ERROR_BASE + 4);
10026: Const 'ERROR_INTERNET_INVALID_URL', 'LongInt').SetInt( INTERNET_ERROR_BASE + 5);
10027: Const 'ERROR_INTERNET_UNRECOGNIZED_SCHEME', 'LongInt').SetInt( INTERNET_ERROR_BASE + 6);
10028: Const 'ERROR_INTERNET_NAME_NOT_RESOLVED', 'LongInt').SetInt( INTERNET_ERROR_BASE + 7);
10029: Const 'ERROR_INTERNET_PROTOCOL_NOT_FOUND', 'LongInt').SetInt( INTERNET_ERROR_BASE + 8);
10030: Const 'ERROR_INTERNET_INVALID_OPTION', 'LongInt').SetInt( INTERNET_ERROR_BASE + 9);
10031: Const 'ERROR_INTERNET_BAD_OPTION_LENGTH', 'LongInt').SetInt( INTERNET_ERROR_BASE + 10);
10032: Const 'ERROR_INTERNET_OPTION_NOT_SETTABLE', 'LongInt').SetInt( INTERNET_ERROR_BASE + 11);
10033: Const 'ERROR_INTERNET_SHUTDOWN', 'LongInt').SetInt( INTERNET_ERROR_BASE + 12);
10034: Const 'ERROR_INTERNET_INCORRECT_USER_NAME', 'LongInt').SetInt( INTERNET_ERROR_BASE + 13);
10035: Const 'ERROR_INTERNET_INCORRECT_PASSWORD', 'LongInt').SetInt( INTERNET_ERROR_BASE + 14);
10036: Const 'ERROR_INTERNET_LOGIN_FAILURE', 'LongInt').SetInt( INTERNET_ERROR_BASE + 15);
10037: Const 'ERROR_INTERNET_INVALID_OPERATION', 'LongInt').SetInt( INTERNET_ERROR_BASE + 16);
10038: Const 'ERROR_INTERNET_OPERATION_CANCELLED', 'LongInt').SetInt( INTERNET_ERROR_BASE + 17);
10039: Const 'ERROR_INTERNET_INCORRECT_HANDLE_TYPE', 'LongInt').SetInt( INTERNET_ERROR_BASE + 18);
10040: Const 'ERROR_INTERNET_INCORRECT_HANDLE_STATE', 'LongInt').SetInt( INTERNET_ERROR_BASE + 19);
10041: Const 'ERROR_INTERNET_NOT_PROXY_REQUEST', 'LongInt').SetInt( INTERNET_ERROR_BASE + 20);
10042: Const 'ERROR_INTERNET_REGISTRY_VALUE_NOT_FOUND', 'LongInt').SetInt( INTERNET_ERROR_BASE + 21);
10043: Const 'ERROR_INTERNET_BAD_REGISTRY_PARAMETER', 'LongInt').SetInt( INTERNET_ERROR_BASE + 22);
10044: Const 'ERROR_INTERNET_NO_DIRECT_ACCESS', 'LongInt').SetInt( INTERNET_ERROR_BASE + 23);
10045: Const 'ERROR_INTERNET_NO_CONTEXT', 'LongInt').SetInt( INTERNET_ERROR_BASE + 24);
10046: Const 'ERROR_INTERNET_NO_CALLBACK', 'LongInt').SetInt( INTERNET_ERROR_BASE + 25);
10047: Const 'ERROR_INTERNET_REQUEST_PENDING', 'LongInt').SetInt( INTERNET_ERROR_BASE + 26);
10048: Const 'ERROR_INTERNET_INCORRECT_FORMAT', 'LongInt').SetInt( INTERNET_ERROR_BASE + 27);
10049: Const 'ERROR_INTERNET_ITEM_NOT_FOUND', 'LongInt').SetInt( INTERNET_ERROR_BASE + 28);
10050: Const 'ERROR_INTERNET_CANNOT_CONNECT', 'LongInt').SetInt( INTERNET_ERROR_BASE + 29);
10051: Const 'ERROR_INTERNET_CONNECTION_ABORTED', 'LongInt').SetInt( INTERNET_ERROR_BASE + 30);
10052: Const 'ERROR_INTERNET_CONNECTION_RESET', 'LongInt').SetInt( INTERNET_ERROR_BASE + 31);
10053: Const 'ERROR_INTERNET_FORCE_RETRY', 'LongInt').SetInt( INTERNET_ERROR_BASE + 32);
10054: Const 'ERROR_INTERNET_INVALID_PROXY_REQUEST', 'LongInt').SetInt( INTERNET_ERROR_BASE + 33);
10055: Const 'ERROR_INTERNET_HANDLE_EXISTS', 'LongInt').SetInt( INTERNET_ERROR_BASE + 36);
10056: Const 'ERROR_INTERNET_SEC_CERT_DATE_INVALID', 'LongInt').SetInt( INTERNET_ERROR_BASE + 37);
10057: Const 'ERROR_INTERNET_SEC_CERT_CN_INVALID', 'LongInt').SetInt( INTERNET_ERROR_BASE + 38);
10058: Const 'ERROR_INTERNET_HTTP_TO_HTTPS_ON_REDIR', 'LongInt').SetInt( INTERNET_ERROR_BASE + 39);
10059: Const 'ERROR_INTERNET_HTTPS_TO_HTTP_ON_REDIR', 'LongInt').SetInt( INTERNET_ERROR_BASE + 40);
10060: Const 'ERROR_INTERNET_MIXED_SECURITY', 'LongInt').SetInt( INTERNET_ERROR_BASE + 41);
10061: Const 'ERROR_INTERNET_CHG_POST_IS_NON_SECURE', 'LongInt').SetInt( INTERNET_ERROR_BASE + 42);
10062: Const 'ERROR_INTERNET_POST_IS_NON_SECURE', 'LongInt').SetInt( INTERNET_ERROR_BASE + 43);
10063: Const 'ERROR_INTERNET_CLIENT_AUTH_CERT_NEEDED', 'LongInt').SetInt( INTERNET_ERROR_BASE + 44);
10064: Const 'ERROR_INTERNET_INVALID_CA', 'LongInt').SetInt( INTERNET_ERROR_BASE + 45);
10065: Const 'ERROR_INTERNET_CLIENT_AUTH_NOT_SETUP', 'LongInt').SetInt( INTERNET_ERROR_BASE + 46);
10066: Const 'ERROR_INTERNET_ASYNC_THREAD_FAILED', 'LongInt').SetInt( INTERNET_ERROR_BASE + 47);

```

```

10067: Const 'ERROR_INTERNET_REDIRECT_SCHEME_CHANGE', 'LongInt').SetInt( INTERNET_ERROR_BASE + 48);
10068: Const 'ERROR_INTERNET_DIALOG_PENDING', 'LongInt').SetInt( INTERNET_ERROR_BASE + 49);
10069: Const 'ERROR_INTERNET_RETRY_DIALOG', 'LongInt').SetInt( INTERNET_ERROR_BASE + 50);
10070: Const 'ERROR_INTERNET_HTTPS_HTTP_SUBMIT_REDIRECT', 'LongInt').SetInt( INTERNET_ERROR_BASE + 52);
10071: Const 'ERROR_INTERNET_INSERT_CDROM', 'LongInt').SetInt( INTERNET_ERROR_BASE + 53);
10072: Const 'ERROR_FTP_TRANSFER_IN_PROGRESS', 'LongInt').SetInt( INTERNET_ERROR_BASE + 110);
10073: Const 'ERROR_FTP_DROPPED', 'LongInt').SetInt( INTERNET_ERROR_BASE + 111);
10074: Const 'ERROR_FTP_NO_PASSIVE_MODE', 'LongInt').SetInt( INTERNET_ERROR_BASE + 112);
10075: Const 'ERROR_GOPHER_PROTOCOL_ERROR', 'LongInt').SetInt( INTERNET_ERROR_BASE + 130);
10076: Const 'ERROR_GOPHER_NOT_FILE', 'LongInt').SetInt( INTERNET_ERROR_BASE + 131);
10077: Const 'ERROR_GOPHER_DATA_ERROR', 'LongInt').SetInt( INTERNET_ERROR_BASE + 132);
10078: Const 'ERROR_GOPHER_END_OF_DATA', 'LongInt').SetInt( INTERNET_ERROR_BASE + 133);
10079: Const 'ERROR_GOPHER_INVALID_LOCATOR', 'LongInt').SetInt( INTERNET_ERROR_BASE + 134);
10080: Const 'ERROR_GOPHER_INCORRECT_LOCATOR_TYPE', 'LongInt').SetInt( INTERNET_ERROR_BASE + 135);
10081: Const 'ERROR_GOPHER_NOT_GOPHER_PLUS', 'LongInt').SetInt( INTERNET_ERROR_BASE + 136);
10082: Const 'ERROR_GOPHER_ATTRIBUTE_NOT_FOUND', 'LongInt').SetInt( INTERNET_ERROR_BASE + 137);
10083: Const 'ERROR_GOPHER_UNKNOWN_LOCATOR', 'LongInt').SetInt( INTERNET_ERROR_BASE + 138);
10084: Const 'ERROR_HTTP_HEADER_NOT_FOUND', 'LongInt').SetInt( INTERNET_ERROR_BASE + 150);
10085: Const 'ERROR_HTTP_DOWNLEVEL_SERVER', 'LongInt').SetInt( INTERNET_ERROR_BASE + 151);
10086: Const 'ERROR_HTTP_INVALID_SERVER_RESPONSE', 'LongInt').SetInt( INTERNET_ERROR_BASE + 152);
10087: Const 'ERROR_HTTP_INVALID_HEADER', 'LongInt').SetInt( INTERNET_ERROR_BASE + 153);
10088: Const 'ERROR_HTTP_INVALID_QUERY_REQUEST', 'LongInt').SetInt( INTERNET_ERROR_BASE + 154);
10089: Const 'ERROR_HTTP_HEADER_ALREADY_EXISTS', 'LongInt').SetInt( INTERNET_ERROR_BASE + 155);
10090: Const 'ERROR_HTTP_REDIRECT_FAILED', 'LongInt').SetInt( INTERNET_ERROR_BASE + 156);
10091: Const 'ERROR_HTTP_NOT_REDIRECTED', 'LongInt').SetInt( INTERNET_ERROR_BASE + 160);
10092: Const 'ERROR_HTTP_COOKIE_NEEDS_CONFIRMATION', 'LongInt').SetInt( INTERNET_ERROR_BASE + 161);
10093: Const 'ERROR_HTTP_COOKIE_DECLINED', 'LongInt').SetInt( INTERNET_ERROR_BASE + 162);
10094: Const 'ERROR_HTTP_REDIRECT_NEEDS_CONFIRMATION', 'LongInt').SetInt( INTERNET_ERROR_BASE + 168);
10095: Const 'ERROR_INTERNET_SECURITY_CHANNEL_ERROR', 'LongInt').SetInt( INTERNET_ERROR_BASE + 157);
10096: Const 'ERROR_INTERNET_UNABLE_TO_CACHE_FILE', 'LongInt').SetInt( INTERNET_ERROR_BASE + 158);
10097: Const 'ERROR_INTERNET_TCPIP_NOT_INSTALLED', 'LongInt').SetInt( INTERNET_ERROR_BASE + 159);
10098: Const 'ERROR_INTERNET_DISCONNECTED', 'LongInt').SetInt( INTERNET_ERROR_BASE + 163);
10099: Const 'ERROR_INTERNET_SERVER_UNREACHABLE', 'LongInt').SetInt( INTERNET_ERROR_BASE + 164);
10100: Const 'ERROR_INTERNET_PROXY_SERVER_UNREACHABLE', 'LongInt').SetInt( INTERNET_ERROR_BASE + 165);
10101: Const 'ERROR_INTERNET_BAD_AUTO_PROXY_SCRIPT', 'LongInt').SetInt( INTERNET_ERROR_BASE + 166);
10102: Const 'ERROR_INTERNET_UNABLE_TO_DOWNLOAD_SCRIPT', 'LongInt').SetInt( INTERNET_ERROR_BASE + 167);
10103: Const 'ERROR_INTERNET_SEC_INVALID_CERT', 'LongInt').SetInt( INTERNET_ERROR_BASE + 169);
10104: Const 'ERROR_INTERNET_SEC_CERT_REVOKED', 'LongInt').SetInt( INTERNET_ERROR_BASE + 170);
10105: Const 'ERROR_INTERNET_FAILED_DUE_TO_SECURITY_CHECK', 'LongInt').SetInt( INTERNET_ERROR_BASE + 171);
10106: //Const 'INTERNET_ERROR_LAST', ''').SetString( ERROR_INTERNET_FAILED_DUE_TO_SECURITY_CHECK);
10107: Const 'NORMAL_CACHE_ENTRY', 'LongWord').SetUInt( $00000001);
10108: Const 'STABLE_CACHE_ENTRY', 'LongWord').SetUInt( $00000002);
10109: Const 'STICKY_CACHE_ENTRY', 'LongWord').SetUInt( $00000004);
10110: Const 'COOKIE_CACHE_ENTRY', 'LongWord').SetUInt( $00100000);
10111: Const 'URLHISTORY_CACHE_ENTRY', 'LongWord').SetUInt( $00200000);
10112: Const 'TRACK_OFFLINE_CACHE_ENTRY', 'LongWord').SetUInt( $00000010);
10113: Const 'TRACK_ONLINE_CACHE_ENTRY', 'LongWord').SetUInt( $00000020);
10114: Const 'SPARSE_CACHE_ENTRY', 'LongWord').SetUInt( $00100000);
10115: Const 'OCK_CACHE_ENTRY', 'LongWord').SetUInt( $00020000);
10116: Const 'CACHE_ENTRY_ATTRIBUTE_FC', 'LongWord').SetUInt( $00000004);
10117: Const 'CACHE_ENTRY_HITRATE_FC', 'LongWord').SetUInt( $00000010);
10118: Const 'CACHE_ENTRY_MODTIME_FC', 'LongWord').SetUInt( $00000040);
10119: Const 'CACHE_ENTRY_EXPTIME_FC', 'LongWord').SetUInt( $00000080);
10120: Const 'CACHE_ENTRY_ACCTIME_FC', 'LongWord').SetUInt( $00000100);
10121: Const 'CACHE_ENTRY_SYNCTIME_FC', 'LongWord').SetUInt( $00000200);
10122: Const 'CACHE_ENTRY_HEADERINFO_FC', 'LongWord').SetUInt( $00000400);
10123: Const 'CACHE_ENTRY_EXEMPT_DELTA_FC', 'LongWord').SetUInt( $00000800);
10124: Const 'INTERNET_CACHE_GROUP_ADD', 'LongInt').SetInt( 0);
10125: Const 'INTERNET_CACHE_GROUP_REMOVE', 'LongInt').SetInt( 1);
10126: Const 'INTERNET_DIAL_UNATTENDED', 'LongWord').SetUInt( $8000);
10127: Function InternetHangUp( dwConnection : DWORD; dwReserved : DWORD) : DWORD;
10128: Const 'INTERENT_GOONLINE_REFRESH', 'LongWord').SetUInt( $00000001);
10129: Const 'INTERENT_GOONLINE_MASK', 'LongWord').SetUInt( $00000001);
10130: Function InternetGoOnline( lpszURL : pchar; hwndParent : HWND; dwFlags : DWORD) : BOOL;
10131: Function InternetAutodial( dwFlags : DWORD; dwReserved : DWORD) : BOOL;
10132: Const 'INTERNET_AUTODIAL_FORCE_ONLINE', 'LongInt').SetInt( 1);
10133: Const 'INTERNET_AUTODIAL_FORCE_UNATTENDED', 'LongInt').SetInt( 2);
10134: Const 'INTERNET_AUTODIAL_FAILIFSECURITYCHECK', 'LongInt').SetInt( 4);
10135: Function InternetAutodialHangup( dwReserved : DWORD) : BOOL;
10136: Function InternetGetConnectedState( lpdwFlags : DWORD; dwReserved : DWORD) : BOOL;
10137: Const 'INTERNET_CONNECTION_MODEM', 'LongInt').SetInt( 1);
10138: Const 'INTERNET_CONNECTION_LAN', 'LongInt').SetInt( 2);
10139: Const 'INTERNET_CONNECTION_PROXY', 'LongInt').SetInt( 4);
10140: Const 'INTERNET_CONNECTION_MODEM_BUSY', 'LongInt').SetInt( 8);
10141: Const 'INTERNET_RAS_INSTALLED', 'LongInt').SetInt( 16);
10142: Const 'INTERNET_CONNECTION_OFFLINE', 'LongInt').SetInt( 32);
10143: Const 'INTERNET_CONNECTION_CONFIGURED', 'LongInt').SetInt( 64);
10144: Const 'INTERNET_CUSTOMDIAL_CONNECT', 'LongInt').SetInt( 0);
10145: Const 'INTERNET_CUSTOMDIAL_UNATTENDED', 'LongInt').SetInt( 1);
10146: Const 'INTERNET_CUSTOMDIAL_DISCONNECT', 'LongInt').SetInt( 2);
10147: Const 'INTERNET_CUSTOMDIAL_SHOWOFFLINE', 'LongInt').SetInt( 4);
10148: Const 'INTERNET_CUSTOMDIAL_SAFE_FOR_UNATTENDED', 'LongInt').SetInt( 1);
10149: Const 'INTERNET_CUSTOMDIAL_WILL_SUPPLY_STATE', 'LongInt').SetInt( 2);
10150: Const 'INTERNET_CUSTOMDIAL_CAN_HANGUP', 'LongInt').SetInt( 4);
10151: Function InternetSetDialState( lpszConnectoid : Pchar; dwState : DWORD; dwReserved : DWORD) : BOOL;
10152: end;
10153:
10154: procedure SIRegister_Printgri(CL: TPSPascalCompiler);
10155: begin

```

```

10156: cwMaxPages', 'LongInt').SetInt( 1000);
10157: 'cwMaxCols', 'LongInt').SetInt( 100);
10158: 'cwTopMargin', 'LongInt').SetInt( 1);
10159: 'cwBottomMargin', 'LongInt').SetInt( 2);
10160: 'cwLeftMargin', 'LongInt').SetInt( 3);
10161: 'cwRightMargin', 'LongInt').SetInt( 4);
10162: TPageNumberPos', ( pnNone, pnTopLeft, pnTopCenter, pnTopRight, pnBotLeft, pnBotCenter, pnBotRight );
10163: SIRegister_TPrintGrid(CL);
10164: // CL.AddDelphiFunction('Procedure Register');
10165: end;
10166:
10167: //*****Cindy Functions!*****
10168: procedure SIRegister_cyIndy(CL: TPSPascalCompiler);
10169: begin
10170:   CL.AddTypeS('TContentTypeMessage', '( cmPlainText, cmPlainText_Attach, cmHtml'
10171:   + '_Attach, cmHtml_RelatedAttach, cmAlterText_Html, cmAlterText_Html_Attach, '
10172:   + 'cmAlterText_Html_RelatedAttach, cmAlterText_Html_Attach_RelatedAttach, cmReadNotification )');
10173:   MessagePlainText', 'String').SetString( 'text/plain');
10174:   CL.AddConstantN('MessagePlainText_Attach', 'String').SetString( 'multipart/mixed');
10175:   MessageAlterText_Html', 'String').SetString( 'multipart/alternative');
10176:   MessageHtml_Attach', 'String').SetString( 'multipart/mixed');
10177:   MessageHtml_RelatedAttach', 'String').SetString( 'multipart/related; type="text/html"');
10178:   MessageAlterText_Html_Attach', 'String').SetString( 'multipart/mixed');
10179:   MessageAlterText_Html_RelatedAttach', 'String')('multipart/related; type="multipart/alternative"');
10180:   MessageAlterText_Html_Attach_RelatedAttach', 'String').SetString( 'multipart/mixed');
10181:   MessageReadNotification', 'String').( 'multipart/report; report-type="disposition-notification"');
10182:   SE_CREATE_TOKEN_NAME', 'String').SetString( 'SeCreateTokenPrivilege');
10183:   SE_ASSIGNPRIMARYTOKEN_NAME', 'String').SetString( 'SeAssignPrimaryTokenPrivilege');
10184:   SE_LOCK_MEMORY_NAME', 'String').SetString( 'SeLockMemoryPrivilege');
10185:   SE_INCREASE_QUOTA_NAME', 'String').SetString( 'SeIncreaseQuotaPrivilege');
10186:   SE_UNSOLICITED_INPUT_NAME', 'String').SetString( 'SeUnsolicitedInputPrivilege');
10187:   SE_MACHINE_ACCOUNT_NAME', 'String').SetString( 'SeMachineAccountPrivilege');
10188:   SE_TCB_NAME', 'String').SetString( 'SeTcbPrivilege');
10189:   SE_SECURITY_NAME', 'String').SetString( 'SeSecurityPrivilege');
10190:   SE_TAKE_OWNERSHIP_NAME', 'String').SetString( 'SeTakeOwnershipPrivilege');
10191:   SE_LOAD_DRIVER_NAME', 'String').SetString( 'SeLoadDriverPrivilege');
10192:   SE_SYSTEM_PROFILE_NAME', 'String').SetString( 'SeSystemProfilePrivilege');
10193:   SE_SYSTEMTIME_NAME', 'String').SetString( 'SeSystemtimePrivilege');
10194:   SE_PROF_SINGLE_PROCESS_NAME', 'String').SetString( 'SeProfileSingleProcessPrivilege');
10195:   SE_INC_BASE_PRIORITY_NAME', 'String').SetString( 'SeIncreaseBasePriorityPrivilege');
10196:   SE_CREATE_PAGEFILE_NAME', 'String').SetString( 'SeCreatePagefilePrivilege');
10197:   SE_CREATE_PERMANENT_NAME', 'String').SetString( 'SeCreatePermanentPrivilege');
10198:   SE_BACKUP_NAME', 'String').SetString( 'SeBackupPrivilege');
10199:   SE_RESTORE_NAME', 'String').SetString( 'SeRestorePrivilege');
10200:   SE_SHUTDOWN_NAME', 'String').SetString( 'SeShutdownPrivilege');
10201:   SE_DEBUG_NAME', 'String').SetString( 'SeDebugPrivilege');
10202:   SE_AUDIT_NAME', 'String').SetString( 'SeAuditPrivilege');
10203:   SE_SYSTEM_ENVIRONMENT_NAME', 'String').SetString( 'SeSystemEnvironmentPrivilege');
10204:   SE_CHANGE_NOTIFY_NAME', 'String').SetString( 'SeChangeNotifyPrivilege');
10205:   SE_REMOTE_SHUTDOWN_NAME', 'String').SetString( 'SeRemoteShutdownPrivilege');
10206:   SE_UNDOCK_NAME', 'String').SetString( 'SeUndockPrivilege');
10207:   SE_SYNC_AGENT_NAME', 'String').SetString( 'SeSyncAgentPrivilege');
10208:   SE_ENABLE_DELEGATION_NAME', 'String').SetString( 'SeEnableDelegationPrivilege');
10209:   SE_MANAGE_VOLUME_NAME', 'String').SetString( 'SeManageVolumePrivilege');
10210:   Def Type(TWindowsVersion', '( wvUnknown, wvWin31, wvWin95, wvWin98, wvWin'
10211:   Me, wvWinNt3, wvWinNt4, wvWin2000, wvWinXP, wvWinVista, wvWin7, wvWin8, wvWin8_Or_Upper ));
10212:   Type(TStrLocateOption', '( strloCaseInsensitive, strloPartialKey ));
10213:   Type(TStrLocateOptions', 'set of TStrLocateOption');
10214:   Type(TStringRead', '( srFromLeft, srFromRight ));
10215:   Type(TStringReads', 'set of TStringRead');
10216:   Type(TCaseSensitive', '( csCaseSensitive, csCaseNotSensitive ));
10217:   Type(TWordsOption', '( woOnlyFirstWord, woOnlyFirstCar ));
10218:   Type(TWordsOptions', 'set of TWordsOption');
10219:   Type(TCarType', '( ctAlphabeticUppercase, ctAlphabeticLowercase, ctNumeric, ctOther ));
10220:   Type(TCarTypes', 'set of TCarType');
10221:   Type(TStringsSortType', '( stNone, stStringSensitive, stStringInsensitive, stExtended ));
10222:   Type(TStringsValueKind', '( skStringSensitive, skStringInsensitive, skExtended ));
10223:   Type(TcyLocateOption', '( lCaseInsensitive, lPartialKey ));
10224:   Type(TcyLocateOptions', 'set of TcyLocateOption');
10225: procedure SIRegister_cyClasses(CL: TPSPascalCompiler);
10226: begin
10227:   SIRegister_TcyRunTimeDesign(CL);
10228:   SIRegister_TcyShadowText(CL);
10229:   SIRegister_TcyBgPicture(CL);
10230:   SIRegister_TcyGradient(CL);
10231:   SIRegister_tcyBevel(CL);
10232:   //CL.AddTypeS('TcyBevelClass', 'class of tcyBevel');
10233:   SIRegister_tcyBevels(CL);
10234:   SIRegister_TcyImagelistOptions(CL);
10235:   Procedure cyDrawBgPicture( aCanvas : TCanvas; aRect : TRect; aBgPicture : TcyBgPicture));
10236: end;
10237:
10238: procedure SIRegister_cyTypes(CL: TPSPascalCompiler);
10239: begin
10240:   Type(TGlyphAlignment', '( gaLeft, gaCenter, gaRight ));
10241:   Type(TGlyphLayout', '( glTop, glCenter, glBottom ));
10242:   Type(TDisabledGlyphOptions', '( dgDoNotDraw, dgDrawNormal, dgDrawMonochrome ));
10243:   Type(TCaptionRender', '( crNormal, crPathEllipsis, crEndEllipsis, crWordEllipsis ));
10244:   Type(TCaptionOrientation', '( coHorizontal, coHorizontalReversed, coVertical, coVerticalReversed ));

```



```

10245:   Type(TBgPosition', '( bgCentered, bgTopLeft, bgTopCenter, bgTopRight, '
10246:     + ' bgCenterRight, bgBottomRight, bgBottomCenter, bgBottomLeft, bgCenterLeft));
10247:   Type(TBgStyle', '( bgNone, bgNormal, bgMosaic, bgStretch, bgStretchProportional ));
10248:   Type(TcyBevelCut', '( bcLowered, bcRaised, bcNone, bcTransparent, bcGradientToNext ));
10249:   Type(TDgradOrientation', '( dgdVertical, dgdHorizontal, dgdAngle, dgdRadial, dgdRectangle ));
10250:   Type(TDgradOrientationShape', '( osRadial, osRectangle ));
10251:   Type(TDgradBalanceMode', '( bmNormal, bmMirror, bmReverse, bmReverseFromColor, bmInvertReverse,
    bmInvertReverseFromColor);
10252:   Type(TRunTimeDesignJob', '( rjNothing, rjMove, rjResizeTop, rjResizeBottom, rjResizeLeft,
    rjResizeTopLeft, rjResizeBottomLeft, rjResizeRight, rjResizeTopRight, rjResizeBottomRight ));
10253:   Type(TLineCoord', 'record BottomCoord : TPoint; TopCoord : TPoint; end');
10254:   cCaptionOrientationWarning', 'String')('Note that text orientation doesn't work with all fonts!');
10255: end;
10256:
10257:
10258: procedure SIRegister_WinSvc(CL: TPSPascalCompiler);
10259: begin
10260:   Const SERVICES_ACTIVE_DATABASEA', 'String').SetString( 'ServicesActive');
10261:   CL.AddConstantN('SERVICES_ACTIVE_DATABASEW', 'String').SetString( 'ServicesActive');
10262:   Const 'SERVICES_ACTIVE_DATABASE', 'String').SetString( 'SERVICES_ACTIVE_DATABASEA');
10263:   Const 'SERVICES_FAILED_DATABASEA', 'String').SetString( 'ServicesFailed');
10264:   Const 'SERVICES_FAILED_DATABASEW', 'String').SetString( 'ServicesFailed');
10265:   Const 'SERVICES_FAILED_DATABASE', 'String').SetString( 'SERVICES_FAILED_DATABASEA');
10266:   Const 'SC_GROUP_IDENTIFIER', 'String').SetString( '+');
10267:   Const 'SC_GROUP_IDENTIFIERW', 'String').SetString( '+');
10268:   Const 'SC_GROUP_IDENTIFIER', 'string').SetString( 'SC_GROUP_IDENTIFIER');
10269:   Const 'SERVICE_NO_CHANGE', 'LongWord').SetUInt( $FFFFFFF);
10270:   Const 'SERVICE_ACTIVE', 'LongWord').SetUInt( $00000001);
10271:   Const 'SERVICE_INACTIVE', 'LongWord').SetUInt( $00000002);
10272:   Const 'SERVICE_CONTROL_STOP', 'LongWord').SetUInt( $00000001);
10273:   Const 'SERVICE_CONTROL_PAUSE', 'LongWord').SetUInt( $00000002);
10274:   Const 'SERVICE_CONTROL_CONTINUE', 'LongWord').SetUInt( $00000003);
10275:   Const 'SERVICE_CONTROL_INTERROGATE', 'LongWord').SetUInt( $00000004);
10276:   Const 'SERVICE_CONTROL_SHUTDOWN', 'LongWord').SetUInt( $00000005);
10277:   Const 'SERVICE_STOPPED', 'LongWord').SetUInt( $00000001);
10278:   Const 'SERVICE_START_PENDING', 'LongWord').SetUInt( $00000002);
10279:   Const 'SERVICE_STOP_PENDING', 'LongWord').SetUInt( $00000003);
10280:   Const 'SERVICE_RUNNING', 'LongWord').SetUInt( $00000004);
10281:   Const 'SERVICE_CONTINUE_PENDING', 'LongWord').SetUInt( $00000005);
10282:   Const 'SERVICE_PAUSE_PENDING', 'LongWord').SetUInt( $00000006);
10283:   Const 'SERVICE_PAUSED', 'LongWord').SetUInt( $00000007);
10284:   Const 'SERVICE_ACCEPT_STOP', 'LongWord').SetUInt( $00000001);
10285:   Const 'SERVICE_ACCEPT_PAUSE_CONTINUE', 'LongWord').SetUInt( $00000002);
10286:   Const 'SERVICE_ACCEPT_SHUTDOWN', 'LongWord').SetUInt( $00000004);
10287:   Const 'SC_MANAGER_CONNECT', 'LongWord').SetUInt( $0001);
10288:   Const 'SC_MANAGER_CREATE_SERVICE', 'LongWord').SetUInt( $0002);
10289:   Const 'SC_MANAGER_ENUMERATE_SERVICE', 'LongWord').SetUInt( $0004);
10290:   Const 'SC_MANAGER_LOCK', 'LongWord').SetUInt( $0008);
10291:   Const 'SC_MANAGER_QUERY_LOCK_STATUS', 'LongWord').SetUInt( $0010);
10292:   Const 'SC_MANAGER_MODIFY_BOOT_CONFIG', 'LongWord').SetUInt( $0020);
10293:   Const 'SERVICE_QUERY_CONFIG', 'LongWord').SetUInt( $0001);
10294:   Const 'SERVICE_CHANGE_CONFIG', 'LongWord').SetUInt( $0002);
10295:   Const 'SERVICE_QUERY_STATUS', 'LongWord').SetUInt( $0004);
10296:   Const 'SERVICE_ENUMERATE_DEPENDENTS', 'LongWord').SetUInt( $0008);
10297:   Const 'SERVICE_START', 'LongWord').SetUInt( $0010);
10298:   Const 'SERVICE_STOP', 'LongWord').SetUInt( $0020);
10299:   Const 'SERVICE_PAUSE_CONTINUE', 'LongWord').SetUInt( $0040);
10300:   Const 'SERVICE_INTERROGATE', 'LongWord').SetUInt( $0080);
10301:   Const 'SERVICE_USER_DEFINED_CONTROL', 'LongWord').SetUInt( $0100);
10302:   Const 'SERVICE_KERNEL_DRIVER', 'LongWord').SetUInt( $00000001);
10303:   Const 'SERVICE_FILE_SYSTEM_DRIVER', 'LongWord').SetUInt( $00000002);
10304:   Const 'SERVICE_ADAPTER', 'LongWord').SetUInt( $00000004);
10305:   Const 'SERVICE_RECOGNIZER_DRIVER', 'LongWord').SetUInt( $00000008);
10306:   Const 'SERVICE_WIN32_OWN_PROCESS', 'LongWord').SetUInt( $00000010);
10307:   Const 'SERVICE_WIN32_SHARE_PROCESS', 'LongWord').SetUInt( $00000020);
10308:   Const 'SERVICE_INTERACTIVE_PROCESS', 'LongWord').SetUInt( $00000100);
10309:   Const 'SERVICE_BOOT_START', 'LongWord').SetUInt( $00000000);
10310:   Const 'SERVICE_SYSTEM_START', 'LongWord').SetUInt( $00000001);
10311:   Const 'SERVICE_AUTO_START', 'LongWord').SetUInt( $00000002);
10312:   Const 'SERVICE_DEMAND_START', 'LongWord').SetUInt( $00000003);
10313:   Const 'SERVICE_DISABLED', 'LongWord').SetUInt( $00000004);
10314:   Const 'SERVICE_ERROR_IGNORE', 'LongWord').SetUInt( $00000000);
10315:   Const 'SERVICE_ERROR_NORMAL', 'LongWord').SetUInt( $00000001);
10316:   Const 'SERVICE_ERROR_SEVERE', 'LongWord').SetUInt( $00000002);
10317:   Const 'SERVICE_ERROR_CRITICAL', 'LongWord').SetUInt( $00000003);
10318:   CL.AddTypeS('SC_HANDLE', 'THandle');
10319:   //CL.AddTypeS('LPSC_HANDLE', '^SC_HANDLE // will not work');
10320:   CL.AddTypeS('SERVICE_STATUS_HANDLE', 'DWORD');
10321:   CL.AddTypeS('SERVICE_STATUS', 'record dwServiceType : DWORD; dwCurrentState '
    + ': DWORD; dwControlsAccepted : DWORD; dwWin32ExitCode : DWORD; dwServiceSpe
    + 'cificExitCode : DWORD; dwCheckPoint : DWORD; dwWaitHint : DWORD; end');
10324:   CL.AddTypeS('SERVICE_STATUS', '_SERVICE_STATUS');
10325:   CL.AddTypeS('TServiceStatus', '_SERVICE_STATUS');
10326:   //CL.AddTypeS('PEnumServiceStatusA', '^TEnumServiceStatusA // will not work');
10327:   //CL.AddTypeS('PEnumServiceStatusW', '^TEnumServiceStatusW // will not work');
10328:   //CL.AddTypeS('PEnumServiceStatus', 'PEnumServiceStatusA');
10329:   CL.AddTypeS('_ENUM_SERVICE_STATUSA', 'record lpServiceName : PChar; lpDis
    + 'playName : PChar; ServiceStatus : TServiceStatus; end');
10331:   CL.AddTypeS('_ENUM_SERVICE_STATUSA', '_ENUM_SERVICE_STATUSA');

```

```

10332: //CL.AddTypeS('ENUM_SERVICE_STATUSW', 'record lpServiceName : PWideChar; lpDis'
10333: // '+playName : PWideChar; ServiceStatus : TServiceStatus; end');
10334: //CL.AddTypeS('ENUM_SERVICE_STATUSW', '_ENUM_SERVICE_STATUSW');
10335: CL.AddTypeS('ENUM_SERVICE_STATUS', '_ENUM_SERVICE_STATUSA');
10336: CL.AddTypeS('TEnumServiceStatusA', '_ENUM_SERVICE_STATUSA');
10337: //CL.AddTypeS('TEnumServiceStatusW', '_ENUM_SERVICE_STATUSW');
10338: CL.AddTypeS('TEnumServiceStatus', 'TEnumServiceStatusA');
10339: CL.AddTypeS('SC_LOCK', '___Pointer');
10340: //CL.AddTypeS('PQueryServiceLockStatusA', '^TQueryServiceLockStatusA // will not work');
10341: //CL.AddTypeS('PQueryServiceLockStatusW', '^TQueryServiceLockStatusW // will not work');
10342: //CL.AddTypeS('PQueryServiceLockStatus', 'PQueryServiceLockStatusA');
10343: CL.AddTypeS('_QUERY_SERVICE_LOCK_STATUSA', 'record fIsLocked : DWORD; lpLockO'
10344: + 'wner : PChar; dwLockDuration : DWORD; end');
10345: //CL.AddTypeS('_QUERY_SERVICE_LOCK_STATUSW', 'record fIsLocked : DWORD; lpLockO'
10346: // '+wner : PWideChar; dwLockDuration : DWORD; end');
10347: CL.AddTypeS('_QUERY_SERVICE_LOCK_STATUS', '_QUERY_SERVICE_LOCK_STATUSA');
10348: CL.AddTypeS('_QUERY_SERVICE_LOCK_STATUSA', '_QUERY_SERVICE_LOCK_STATUSA');
10349: //CL.AddTypeS('_QUERY_SERVICE_LOCK_STATUSW', '_QUERY_SERVICE_LOCK_STATUSW');
10350: CL.AddTypeS('QUERY_SERVICE_LOCK_STATUS', 'QUERY_SERVICE_LOCK_STATUSA');
10351: CL.AddTypeS('TQueryServiceLockStatusA', '_QUERY_SERVICE_LOCK_STATUSA');
10352: //CL.AddTypeS('TQueryServiceLockStatusW', '_QUERY_SERVICE_LOCK_STATUSW');
10353: CL.AddTypeS('TQueryServiceLockStatus', 'TQueryServiceLockStatusA');
10354: CL.AddTypeS('_QUERY_SERVICE_CONFIGA', 'record dwServiceType : DWORD; dwStartT'
10355: + 'ype : DWORD; dwErrorControl : DWORD; lpBinaryPathName : PChar; lpLoadO'
10356: + 'rderGroup : PChar; dwTagId : DWORD; lpDependencies : PChar; lpServ'
10357: + 'iceStartName : PChar; lpDisplayName : PChar; end');
10358: //CL.AddTypeS('_QUERY_SERVICE_CONFIGW', 'record dwServiceType : DWORD; dwStartT'
10359: // '+ype : DWORD; dwErrorControl : DWORD; lpBinaryPathName : PWideChar; lpLoadO'
10360: // '+rderGroup : PWideChar; dwTagId : DWORD; lpDependencies : PWideChar; lpServ'
10361: // '+iceStartName : PWideChar; lpDisplayName : PWideChar; end');
10362: CL.AddTypeS('_QUERY_SERVICE_CONFIG', '_QUERY_SERVICE_CONFIGA');
10363: CL.AddTypeS('QUERY_SERVICE_CONFIGA', '_QUERY_SERVICE_CONFIGA');
10364: //CL.AddTypeS('QUERY_SERVICE_CONFIGW', '_QUERY_SERVICE_CONFIGW');
10365: CL.AddTypeS('QUERY_SERVICE_CONFIG', 'QUERY_SERVICE_CONFIGA');
10366: CL.AddTypeS('TQueryServiceConfigA', '_QUERY_SERVICE_CONFIGA');
10367: //CL.AddTypeS('TQueryServiceConfigW', '_QUERY_SERVICE_CONFIGW');
10368: CL.AddTypeS('TQueryServiceConfig', 'TQueryServiceConfigA');
10369:
10370: procedure SIRegister_SvcMgr(CL: TPSPascalCompiler);
10371: begin
10372:   SIRegister_TEventLogger(CL);
10373:   SIRegister_TDependency(CL);
10374:   SIRegister_TDependencies(CL);
10375:   Const 'CM_SERVICE_CONTROL_CODE', 'LongInt').SetInt( WM_USER + 1);
10376:   CL.AddClassN(CL.FindClass('TOBJECT'), 'TService');
10377:   SIRegister_TServiceThread(CL);
10378:   TServiceType, '( stWin32, stDevice, stFileSystem )';
10379:   CL.AddTypeS('TCurrentStatus', '( csStopped, csStartPending, csStopPending, cs'
10380: + 'Running, csContinuePending, csPausePending, csPaused )');
10381:   TErrorSeverity, '( esIgnore, esNormal, esSevere, esCritical )';
10382:   TStartType, '( stBoot, stSystem, stAuto, stManual, stDisabled )';
10383:   TServiceEvent, 'Procedure ( Sender : TService)';
10384:   TContinueEvent, 'Procedure ( Sender : TService; var Continued : Boolean)';
10385:   TPauseEvent, 'Procedure ( Sender : TService; var Paused : Boolean)';
10386:   TStartEvent, 'Procedure ( Sender : TService; var Started : Boolean)';
10387:   TmStopEvent, 'Procedure ( Sender : TService; var Stopped : Boolean)';
10388:   SIRegister_TService(CL);
10389:   SIRegister_TServiceApplication(CL);
10390: end;
10391:
10392: procedure SIRegister_UWavein4(CL: TPSPascalCompiler);
10393: begin
10394:   Const ('MAX_BUFFERS', 'LongInt').SetInt( 8);
10395:   TypeS('TBufferFullEvent', 'Procedure ( Sender : TObjct; Data : TObjct; Size : longint)');
10396:   TypeS('TErrorEvent', 'Procedure ( s : string)');
10397:   SIRegister_TWaveIn(CL);
10398: end;
10399:
10400: procedure SIRegister_AMixer(CL: TPSPascalCompiler);
10401: begin
10402:   CL.AddClassN(CL.FindClass('TOBJECT'), 'TAudioMixer');
10403:   CL.AddTypeS('TMIXERCAPS', 'record wMid: Word; wPid: Word; +
10404: 'vDriverVersion: Longint; szPname: array[0..32] of char; fdwSupport : DWord; cDestinations: DWORD; end');
10405:   CL.AddTypeS('TPlaylistFreeItemNotify', 'Procedure ( Pntr : TObjct)');
10406:   CL.AddTypeS('TMixerChange', 'Procedure ( Sender : TObjct; MixerH : HMixer; ID : Integer)');
10407:   CL.AddTypeS('HMIXER', 'Integer');
10408:   SIRegister_TPointerList(CL);
10409:   SIRegister_TMixerControls(CL);
10410:   SIRegister_TMixerConnection(CL);
10411:   SIRegister_TMixerConnections(CL);
10412:   SIRegister_TMixerDestination(CL);
10413:   SIRegister_TMixerDestinations(CL);
10414:   SIRegister_TAudioMixer(CL);
10415:   //CL.AddDelphiFunction('Procedure Register');
10416: end;
10417:
10418:
10419: procedure SIRegister_UrlMon(CL: TPSPascalCompiler);
10420: begin

```

```

10421: //CL.AddConstantN('SZ_URLCONTEXT','POLEStr').SetString( 'URL Context');
10422: //CL.AddConstantN('SZ_ASYNC_CALLEE','POLEStr').SetString( 'AsyncCallee');
10423: CL.AddConstantN('MKSYS_URLMONIKER','LongInt').SetInt( 6);
10424:     type ULONG, 'Cardinal';
10425:     LPCWSTR, 'PChar';
10426: CL.AddTypeS('LPWSTR', 'PChar');
10427: LPSTR, 'PChar';
10428: TBindVerb, 'ULONG';
10429: TBindInfoF, 'ULONG';
10430: TBindF, 'ULONG';
10431: TBSCF, 'ULONG';
10432: TBindStatus, 'ULONG';
10433: TCIPStatus, 'ULONG';
10434: TBindString, 'ULONG';
10435: TPiFlags, 'ULONG';
10436: TOIBdgFlags, 'ULONG';
10437: TParseAction, 'ULONG';
10438: TPSUAction, 'ULONG';
10439: TQueryOption, 'ULONG';
10440: TPUAF, 'ULONG';
10441: TSZMFlags, 'ULONG';
10442: TUrlZone, 'ULONG';
10443: TUrlTemplate, 'ULONG';
10444: TZAFlags, 'ULONG';
10445: TUrlZoneReg, 'ULONG';
10446: 'URLMON_OPTION_USERAGENT','LongWord').SetUInt( $10000001);
10447: CL.AddConstantN('URLMON_OPTION_USERAGENT_REFRESH','LongWord').SetUInt( $10000002);
10448: const 'URLMON_OPTION_URL_ENCODING','LongWord').SetUInt( $10000004);
10449: const 'URLMON_OPTION_USE_BINDSTRINGCREDS','LongWord').SetUInt( $10000008);
10450: const 'CF_NULL','LongInt').SetInt( 0);
10451: const 'CFSTR_MIME_NULL','LongInt').SetInt( 0);
10452: const 'CFSTR_MIME_TEXT','String').SetString( 'text/plain');
10453: const 'CFSTR_MIME_RICHTEXT','String').SetString( 'text/richtext');
10454: const 'CFSTR_MIME_X_BITMAP','String').SetString( 'image/x-xbitmap');
10455: const 'CFSTR_MIME_POSTSCRIPT','String').SetString( 'application/postscript');
10456: const 'CFSTR_MIME_AIFF','String').SetString( 'audio/aiff');
10457: const 'CFSTR_MIME_BASICAUDIO','String').SetString( 'audio/basic');
10458: const 'CFSTR_MIME_WAV','String').SetString( 'audio/wav');
10459: const 'CFSTR_MIME_X_WAV','String').SetString( 'audio/x-wav');
10460: const 'CFSTR_MIME_GIF','String').SetString( 'image/gif');
10461: const 'CFSTR_MIME_PJPEG','String').SetString( 'image/pjpeg');
10462: const 'CFSTR_MIME_JPEG','String').SetString( 'image/jpeg');
10463: const 'CFSTR_MIME_TIFF','String').SetString( 'image/tiff');
10464: const 'CFSTR_MIME_X_PNG','String').SetString( 'image/x-png');
10465: const 'CFSTR_MIME_BMP','String').SetString( 'image/bmp');
10466: const 'CFSTR_MIME_X_ART','String').SetString( 'image/x-jg');
10467: const 'CFSTR_MIME_X_EMF','String').SetString( 'image/x-emf');
10468: const 'CFSTR_MIME_X_WMF','String').SetString( 'image/x-wmf');
10469: const 'CFSTR_MIME_AVI','String').SetString( 'video/avi');
10470: const 'CFSTR_MIME_MPEG','String').SetString( 'video/mpeg');
10471: const 'CFSTR_MIME_FRACTALS','String').SetString( 'application/fractals');
10472: const 'CFSTR_MIME_RAWDATA','String').SetString( 'application/octet-stream');
10473: const 'CFSTR_MIME_RAWDATASTRM','String').SetString( 'application/octet-stream');
10474: const 'CFSTR_MIME_PDF','String').SetString( 'application/pdf');
10475: const 'CFSTR_MIME_X_AIFF','String').SetString( 'audio/x-aiff');
10476: const 'CFSTR_MIME_X_REALAUDIO','String').SetString( 'audio/x-pn-realaudio');
10477: const 'CFSTR_MIME_XBM','String').SetString( 'image/xbm');
10478: const 'CFSTR_MIME_QUICKTIME','String').SetString( 'video/quicktime');
10479: const 'CFSTR_MIME_X_MSVIDEO','String').SetString( 'video/x-msvideo');
10480: const 'CFSTR_MIME_X_SGI_MOVIE','String').SetString( 'video/x-sgi-movie');
10481: const 'CFSTR_MIME_HTML','String').SetString( 'text/html');
10482: const 'MK_S_ASYNCHRONOUS','LongWord').SetUInt( $000401E8);
10483: const 'S_ASYNCHRONOUS','LongWord').SetUInt( $000401E8);
10484: const 'E_PENDING','LongWord').SetUInt( $8000000A);
10485: CL.AddInterface(CL.FindInterface('IUNKNOWN'),IBinding, 'IBinding');
10486: SIRegister_IPersistMoniker(CL);
10487: SIRegister_IBindProtocol(CL);
10488: SIRegister_IBinding(CL);
10489: const 'BINDVERB_GET','LongWord').SetUInt( $00000000);
10490: const 'BINDVERB_POST','LongWord').SetUInt( $00000001);
10491: const 'BINDVERB_PUT','LongWord').SetUInt( $00000002);
10492: const 'BINDVERB_CUSTOM','LongWord').SetUInt( $00000003);
10493: const 'BINDINFOF_URLCODESTGMEDDATA','LongWord').SetUInt( $00000001);
10494: const 'BINDINFOF_URLCODEDEXTRAINFO','LongWord').SetUInt( $00000002);
10495: const 'BINDF_ASYNCHRONOUS','LongWord').SetUInt( $00000001);
10496: const 'BINDF_ASYNCSTORAGE','LongWord').SetUInt( $00000002);
10497: const 'BINDF_NOPROGRESSIVERENDERING','LongWord').SetUInt( $00000004);
10498: const 'BINDF_OFFLINEOPERATION','LongWord').SetUInt( $00000008);
10499: const 'BINDF_GETNEWESTVERSION','LongWord').SetUInt( $00000010);
10500: const 'BINDF_NOWRITECACHE','LongWord').SetUInt( $00000020);
10501: const 'BINDF_NEEDFILE','LongWord').SetUInt( $00000040);
10502: const 'BINDF_PULLDATA','LongWord').SetUInt( $00000080);
10503: const 'BINDF_IGNORESECURITYPROBLEM','LongWord').SetUInt( $00000100);
10504: const 'BINDF_RESYNCHRONIZE','LongWord').SetUInt( $00000200);
10505: const 'BINDF_HYPERLINK','LongWord').SetUInt( $00000400);
10506: const 'BINDF_NO_UI','LongWord').SetUInt( $00000800);
10507: const 'BINDF_SILENTOPERATION','LongWord').SetUInt( $00001000);
10508: const 'BINDF_PRAGMA_NO_CACHE','LongWord').SetUInt( $00002000);
10509: const 'BINDF_FREE_THREADED','LongWord').SetUInt( $00010000);

```

```

10510: const 'BINDF_DIRECT_READ', 'LongWord').SetUInt( $00020000);
10511: const 'BINDF_FORMS_SUBMIT', 'LongWord').SetUInt( $00040000);
10512: const 'BINDF_GETFROMCACHE_IF_NET_FAIL', 'LongWord').SetUInt( $00080000);
10513: //const 'BINDF_DONTUSECACHE', '').SetString( BINDF_GETNEWESTVERSION);
10514: //const 'BINDF_DONTPUTINCACHE', '').SetString( BINDF_NOWRITECACHE);
10515: //const 'BINDF_NOCOPYDATA', '').SetString( BINDF_PULLDATA);
10516: const 'BSCF_FIRSTDATANOTIFICATION', 'LongWord').SetUInt( $00000001);
10517: const 'BSCF_INTERMEDIATEDATANOTIFICATION', 'LongWord').SetUInt( $00000002);
10518: const 'BSCF_LASTDATANOTIFICATION', 'LongWord').SetUInt( $00000004);
10519: const 'BSCF_DATAFULLYAVAILABLE', 'LongWord').SetUInt( $00000008);
10520: const 'BSCF_AVAILABLEDATASIZEUNKNOWN', 'LongWord').SetUInt( $00000010);
10521: const 'BINDSTATUS_FINDINGRESOURCE', 'LongInt').SetInt( 1);
10522: const 'BINDSTATUS_CONNECTING', 'LongInt').SetInt( BINDSTATUS_FINDINGRESOURCE + 1);
10523: const 'BINDSTATUS_REDIRECTING', 'LongInt').SetInt( BINDSTATUS_CONNECTING + 1);
10524: const 'BINDSTATUS_BEGINDOWNLOADDATA', 'LongInt').SetInt( BINDSTATUS_REDIRECTING + 1);
10525: const 'BINDSTATUS_DOWNLOADINGDATA', 'LongInt').SetInt( BINDSTATUS_BEGINDOWNLOADDATA + 1);
10526: const 'BINDSTATUS_ENDDOWNLOADDATA', 'LongInt').SetInt( BINDSTATUS_DOWNLOADINGDATA + 1);
10527: const 'BINDSTATUS_BEGINDOWNLOADCMPONENTS', 'LongInt').SetInt( BINDSTATUS_ENDDOWNLOADDATA + 1);
10528: const 'BINDSTATUS_INSTALLINGCOMPONENTS', 'LongInt').SetInt( BINDSTATUS_BEGINDOWNLOADCMPONENTS + 1);
10529: const 'BINDSTATUS_ENDDOWNLOADCMPONENTS', 'LongInt').SetInt( BINDSTATUS_INSTALLINGCOMPONENTS + 1);
10530: const 'BINDSTATUS_USINGCACHEDCOPY', 'LongInt').SetInt( BINDSTATUS_ENDDOWNLOADCMPONENTS + 1);
10531: const 'BINDSTATUS_SENDINGREQUEST', 'LongInt').SetInt( BINDSTATUS_USINGCACHEDCOPY + 1);
10532: const 'BINDSTATUS_CLASSIDAVAILABLE', 'LongInt').SetInt( BINDSTATUS_SENDINGREQUEST + 1);
10533: const 'BINDSTATUS_MIMETYPEAVAILABLE', 'LongInt').SetInt( BINDSTATUS_CLASSIDAVAILABLE + 1);
10534: const 'BINDSTATUS_CACHEFILENAMEAVAILABLE', 'LongInt').SetInt( BINDSTATUS_MIMETYPEAVAILABLE + 1);
10535: const 'BINDSTATUS_BEGINSYNCOOPERATION', 'LongInt').SetInt( BINDSTATUS_CACHEFILENAMEAVAILABLE + 1);
10536: const 'BINDSTATUS_ENDSYNCOOPERATION', 'LongInt').SetInt( BINDSTATUS_BEGINSYNCOOPERATION + 1);
10537: const 'BINDSTATUS_BEGINUPLOADDATA', 'LongInt').SetInt( BINDSTATUS_ENDSYNCOOPERATION + 1);
10538: const 'BINDSTATUS_UPLOADINGDATA', 'LongInt').SetInt( BINDSTATUS_BEGINUPLOADDATA + 1);
10539: const 'BINDSTATUS_ENDUPLOADDATA', 'LongInt').SetInt( BINDSTATUS_UPLOADINGDATA + 1);
10540: const 'BINDSTATUS_PROTOCOLCLASSID', 'LongInt').SetInt( BINDSTATUS_ENDUPLOADDATA + 1);
10541: const 'BINDSTATUS_ENCODING', 'LongInt').SetInt( BINDSTATUS_PROTOCOLCLASSID + 1);
10542: const 'BINDSTATUS_VERIFIEDMIMETYPEAVAILABLE', 'LongInt').SetInt( BINDSTATUS_ENCODING + 1);
10543: const 'BINDSTATUS_CLASSINSTALLLOCATION', 'LongInt').SetInt( BINDSTATUS_VERIFIEDMIMETYPEAVAILABLE + 1);
10544: const 'BINDSTATUS_DECODING', 'LongInt').SetInt( BINDSTATUS_CLASSINSTALLLOCATION + 1);
10545: const 'BINDSTATUS_LOADINGMIMEHANDLER', 'LongInt').SetInt( BINDSTATUS_DECODING + 1);
10546: const 'BINDSTATUS_CONTENTDISPOSITIONATTACH', 'LongInt').SetInt( BINDSTATUS_LOADINGMIMEHANDLER + 1);
10547: const 'BINDSTATUS_FILTERREPORTMIMETYPE', 'LongInt').SetInt( BINDSTATUS_CONTENTDISPOSITIONATTACH + 1);
10548: const 'BINDSTATUS_CLSIDCANINSTATIATE', 'LongInt').SetInt( BINDSTATUS_FILTERREPORTMIMETYPE + 1);
10549: const 'BINDSTATUS_IUNKNOWNAVAILABLE', 'LongInt').SetInt( BINDSTATUS_CLSIDCANINSTATIATE + 1);
10550: const 'BINDSTATUS_DIRECTBIND', 'LongInt').SetInt( BINDSTATUS_IUNKNOWNAVAILABLE + 1);
10551: const 'BINDSTATUS_RAWMIMETYPE', 'LongInt').SetInt( BINDSTATUS_DIRECTBIND + 1);
10552: const 'BINDSTATUS_PROXYDETECTING', 'LongInt').SetInt( BINDSTATUS_RAWMIMETYPE + 1);
10553: const 'BINDSTATUS_ACCEPTRANGES', 'LongInt').SetInt( BINDSTATUS_PROXYDETECTING + 1);
10554: const 'BINDSTATUS_COOKIE_SENT', 'LongInt').SetInt( BINDSTATUS_ACCEPTRANGES + 1);
10555: const 'BINDSTATUS_COMPACT_POLICY_RECEIVED', 'LongInt').SetInt( BINDSTATUS_COOKIE_SENT + 1);
10556: const 'BINDSTATUS_COOKIE_SUPPRESSED', 'LongInt').SetInt( BINDSTATUS_COMPACT_POLICY_RECEIVED + 1);
10557: const 'BINDSTATUS_COOKIE_STATE_UNKNOWN', 'LongInt').SetInt( BINDSTATUS_COOKIE_SUPPRESSED + 1);
10558: const 'BINDSTATUS_COOKIE_STATE_ACCEPT', 'LongInt').SetInt( BINDSTATUS_COOKIE_STATE_UNKNOWN + 1);
10559: const 'BINDSTATUS_COOKIE_STATE_REJECT', 'LongInt').SetInt( BINDSTATUS_COOKIE_STATE_ACCEPT + 1);
10560: const 'BINDSTATUS_COOKIE_STATE_PROMPT', 'LongInt').SetInt( BINDSTATUS_COOKIE_STATE_REJECT + 1);
10561: const 'BINDSTATUS_COOKIE_STATE_LEASH', 'LongInt').SetInt( BINDSTATUS_COOKIE_STATE_PROMPT + 1);
10562: const 'BINDSTATUS_COOKIE_STATE_DOWNGRADE', 'LongInt').SetInt( BINDSTATUS_COOKIE_STATE_LEASH + 1);
10563: const 'BINDSTATUS_POLICY_HREF', 'LongInt').SetInt( BINDSTATUS_COOKIE_STATE_DOWNGRADE + 1);
10564: const 'BINDSTATUS_P3P_HEADER', 'LongInt').SetInt( BINDSTATUS_POLICY_HREF + 1);
10565: const 'BINDSTATUS_SESSION_COOKIE_RECEIVED', 'LongInt').SetInt( BINDSTATUS_P3P_HEADER + 1);
10566: const 'BINDSTATUS_PERSISTENT_COOKIE_RECEIVED', 'LongInt').SetInt( BINDSTATUS_SESSION_COOKIE_RECEIVED + 1);
10567: const 'BINDSTATUS_SESSION_COOKIES_ALLOWED', 'LongInt').SetInt( BINDSTATUS_PERSISTENT_COOKIE_RECEIVED + 1);
10568: const 'BINDSTATUS_CACHECONTROL', 'LongInt').SetInt( BINDSTATUS_SESSION_COOKIES_ALLOWED + 1);
10569: const 'BINDSTATUS_CONTENTDISPOSITIONFILENAME', 'LongInt').SetInt( BINDSTATUS_CACHECONTROL + 1);
10570: const 'BINDSTATUS_MIMETEXTPLAINMISMATCH', 'LongInt').SetInt( BINDSTATUS_CONTENTDISPOSITIONFILENAME + 1);
10571: const 'BINDSTATUS_PUBLISHERAVAILABLE', 'LongInt').SetInt( BINDSTATUS_MIMETEXTPLAINMISMATCH + 1);
10572: const 'BINDSTATUS_DISPLAYNAMEAVAILABLE', 'LongInt').SetInt( BINDSTATUS_PUBLISHERAVAILABLE + 1);
10573: // PBindInfo, '^TBindInfo // will not work';
10574: { _tagBINDINFO, 'record cbSize : ULONG; szExtraInfo : LPWSTR; stg'
10575: + 'medData : TStgMedium; grfBindInfoF : DWORD; dwBindVerb : DWORD; szCustomVe'
10576: + 'rb : LPWSTR; cbstgmedData : DWORD; dwOptions : DWORD; dwOptionsFlags : DWORD'
10577: + 'RD; dwCodePage : DWORD; securityAttributes : TSecurityAttributes; iid : TGUID'
10578: + 'UID; pUnk : IUnknown; dwReserved : DWORD; end');
10579: TBindInfo, '_tagBINDINFO';
10580: BINDINFO, '_tagBINDINFO';}
10581: _REMSECURITY_ATTRIBUTES, 'record nLength : DWORD; lpSecurityDes'
10582: + 'cryptor : DWORD; bInheritHandle : BOOL; end');
10583: TRemSecurityAttributes, '_REMSECURITY_ATTRIBUTES';
10584: REMSECURITY_ATTRIBUTES, '_REMSECURITY_ATTRIBUTES';
10585: //PRemBindInfo, '^TRemBindInfo // will not work';
10586: { _tagRemBINDINFO, 'record cbSize : ULONG; szExtraInfo : LPWSTR; '
10587: + 'grfBindInfoF : DWORD; dwBindVerb : DWORD; szCustomVerb : LPWSTR; cbstgmedD'
10588: + 'ata : DWORD; dwOptions : DWORD; dwOptionsFlags : DWORD; dwCodePage : DWORD'
10589: + 'securityAttributes : TRemSecurityAttributes; iid : TGUID; pUnk : IUnknow'
10590: + 'n; dwReserved : DWORD; end');
10591: TRemBindInfo, '_tagRemBINDINFO';
10592: RemBINDINFO, '_tagRemBINDINFO';}
10593: //PRemFormatEtc, '^TRemFormatEtc // will not work';
10594: tagRemFORMATETC, 'record cfFormat : DWORD; ptd : DWORD; dwAspect : DWORD; lindex : Longint; tyMed : DWORD; end';
10595: TRemFormatEtc, 'tagRemFORMATETC';
10596: REMFORMATETC, 'tagRemFORMATETC';
10597: SIRegister_IBindStatusCallback(CLI);
10598: SIRegister_IAuthenticate(CLI);

```



```

10599: SIRegister_IHttpNegotiate(CL);
10600: SIRegister_IWindowForBindingUI(CL);
10601: const 'CIP_DISK_FULL', 'LongInt').SetInt( 0);
10602: const 'CIP_ACCESS_DENIED', 'LongInt').SetInt( CIP_DISK_FULL + 1);
10603: const 'CIP_NEWER_VERSION_EXISTS', 'LongInt').SetInt( CIP_ACCESS_DENIED + 1);
10604: const 'CIP_OLDER_VERSION_EXISTS', 'LongInt').SetInt( CIP_NEWER_VERSION_EXISTS + 1);
10605: const 'CIP_NAME_CONFLICT', 'LongInt').SetInt( CIP_OLDER_VERSION_EXISTS + 1);
10606: const 'CIP_TRUST_VERIFICATION_COMPONENT_MISSING', 'LongInt').SetInt( CIP_NAME_CONFLICT + 1);
10607: const 'CIP_EXE_SELF_REGISTRATION_TIMEOUT', 'LongInt').SetInt( CIP_TRUST_VERIFICATION_COMPONENT_MISSING + 1);
10608: const 'CIP_UNSAFE_TO_ABORT', 'LongInt').SetInt( CIP_EXE_SELF_REGISTRATION_TIMEOUT + 1);
10609: const 'CIP_NEED_REBOOT', 'LongInt').SetInt( CIP_UNSAFE_TO_ABORT + 1);
10610: const 'CIP_NEED_REBOOT_UI_PERMISSION', 'LongInt').SetInt( CIP_NEED_REBOOT + 1);
10611: SIRegister_ICodeInstall(CL);
10612: SIRegister_IWinInetInfo(CL);
10613: const 'WININETINFO_OPTION_LOCK_HANDLE', 'LongInt').SetInt( 65534);
10614: SIRegister_IHttpSecurity(CL);
10615: SIRegister_IWinInetHttpInfo(CL);
10616: SIRegister_IBindHost(CL);
10617: const 'URLOSTRM_USECACHEDCOPY_ONLY', 'LongWord').SetUInt( $00000001);
10618: const 'URLOSTRM_USECACHEDCOPY', 'LongWord').SetUInt( $00000002);
10619: const 'URLOSTRM_GETNEWESTVERSION', 'LongWord').SetUInt( $00000003);
10620: Function URLOpenStream( p1 : IUnknown; p2 : PChar; p3 : DWORD; p4 : IBindStatusCallback) : HRESULT;
10621: Function URLOpenPullStream( p1 : IUnknown; p2 : PChar; p3 : DWORD; BSC : IBindStatusCallback) : HRESULT;
10622: Function URLDownloadToFile( Caller : IUnknown; URL : PChar; FileName : PChar; Reserved : DWORD; StatusCB
: IBindStatusCallback) : HRESULT;
10623: Function URLDownloadToCacheFile( p1 : IUnknown; p2 : PChar; p3 : PChar; p4 : DWORD; p5 : DWORD; p6 :
IBindStatusCallback) : HRESULT;
10624: Function URLOpenBlockingStream(p1:IUnknown;p2:PChar;out
p3:IStream;p4:DWORD;p5:IBindStatusCallback):HRESULT;
10625: Function HlinkGoBack( unk : IUnknown) : HRESULT;
10626: Function HlinkGoForward( unk : IUnknown) : HRESULT;
10627: Function HlinkNavigateString( unk : IUnknown; szTarget : LPCWSTR) : HRESULT;
10628: // Function HlinkNavigateMoniker( Unk : IUnknown; mkTarget : IMoniker) : HRESULT;
10629: SIRegister_IInternet(CL);
10630: const 'BINDSTRING_HEADERS', 'LongInt').SetInt( 1);
10631: const 'BINDSTRING_ACCEPT_MIMES', 'LongInt').SetInt( BINDSTRING_HEADERS + 1);
10632: const 'BINDSTRING_EXTRA_URL', 'LongInt').SetInt( BINDSTRING_ACCEPT_MIMES + 1);
10633: const 'BINDSTRING_LANGUAGE', 'LongInt').SetInt( BINDSTRING_EXTRA_URL + 1);
10634: const 'BINDSTRING_USERNAME', 'LongInt').SetInt( BINDSTRING_LANGUAGE + 1);
10635: const 'BINDSTRING_PASSWORD', 'LongInt').SetInt( BINDSTRING_USERNAME + 1);
10636: const 'BINDSTRING_UA_PIXELS', 'LongInt').SetInt( BINDSTRING_PASSWORD + 1);
10637: const 'BINDSTRING_UA_COLOR', 'LongInt').SetInt( BINDSTRING_UA_PIXELS + 1);
10638: const 'BINDSTRING_OS', 'LongInt').SetInt( BINDSTRING_UA_COLOR + 1);
10639: const 'BINDSTRING_USER_AGENT', 'LongInt').SetInt( BINDSTRING_OS + 1);
10640: const 'BINDSTRING_ACCEPT_ENCODINGS', 'LongInt').SetInt( BINDSTRING_USER_AGENT + 1);
10641: const 'BINDSTRING_POST_COOKIE', 'LongInt').SetInt( BINDSTRING_ACCEPT_ENCODINGS + 1);
10642: const 'BINDSTRING_POST_DATA_MIME', 'LongInt').SetInt( BINDSTRING_POST_COOKIE + 1);
10643: const 'BINDSTRING_URL', 'LongInt').SetInt( BINDSTRING_POST_DATA_MIME + 1);
10644: //POLEStrArray, 'TOLEStrArray // will not work';
10645: SIRegister_IInternetBindInfo(CL);
10646: const 'PI_PARSE_URL', 'LongWord').SetUInt( $00000001);
10647: const 'PI_FILTER_MODE', 'LongWord').SetUInt( $00000002);
10648: const 'PI_FORCE_ASYNC', 'LongWord').SetUInt( $00000004);
10649: const 'PI_USE_WORKERTHREAD', 'LongWord').SetUInt( $00000008);
10650: const 'PI_MIMEVERIFICATION', 'LongWord').SetUInt( $00000010);
10651: const 'PI_CLSIDLOOKUP', 'LongWord').SetUInt( $00000020);
10652: const 'PI_DATAPROGRESS', 'LongWord').SetUInt( $00000040);
10653: const 'PI_SYNCHRONOUS', 'LongWord').SetUInt( $00000080);
10654: const 'PI_APARTMENTTHREADED', 'LongWord').SetUInt( $00000100);
10655: const 'PI_CLASSINSTALL', 'LongWord').SetUInt( $00000200);
10656: const 'PD_FORCE_SWITCH', 'LongWord').SetUInt( $00010000);
10657: //const 'PI_DOCFILECLSIDLOOKUP', 'LongWord').SetUInt( $00000000);
10658: //PProtocolData, 'TProtocolData // will not work';
10659: _tagPROTOCOLDATA, 'record grfFlags:DWORD; dwState:DWORD; pData : TObject; cbData : ULONG; end';
10660: TProtocolData, '_tagPROTOCOLDATA';
10661: PROTOCOLDATA, '_tagPROTOCOLDATA';
10662: CL.AddInterface(CL.FindInterface('IUNKNOWN'), IInternetProtocolSink, 'IInternetProtocolSink');
10663: SIRegister_IInternetProtocolRoot(CL);
10664: SIRegister_IInternetProtocol(CL);
10665: SIRegister_IInternetProtocolSink(CL);
10666: const 'OIBDG_APARTMENTTHREADED', 'LongWord').SetUInt( $00000100);
10667: SIRegister_IInternetSession(CL);
10668: SIRegister_IInternetThreadSwitch(CL);
10669: SIRegister_IInternetPriority(CL);
10670: const 'PARSE_CANONICALIZE', 'LongInt').SetInt( 1);
10671: const 'PARSE_FRIENDLY', 'LongInt').SetInt( PARSE_CANONICALIZE + 1);
10672: const 'PARSE_SECURITY_URL', 'LongInt').SetInt( PARSE_FRIENDLY + 1);
10673: const 'PARSE_ROOTDOCUMENT', 'LongInt').SetInt( PARSE_SECURITY_URL + 1);
10674: const 'PARSE_DOCUMENT', 'LongInt').SetInt( PARSE_ROOTDOCUMENT + 1);
10675: const 'PARSE_ANCHOR', 'LongInt').SetInt( PARSE_DOCUMENT + 1);
10676: const 'PARSE_ENCODE', 'LongInt').SetInt( PARSE_ANCHOR + 1);
10677: const 'PARSE_DECODE', 'LongInt').SetInt( PARSE_ENCODE + 1);
10678: const 'PARSE_PATH_FROM_URL', 'LongInt').SetInt( PARSE_DECODE + 1);
10679: const 'PARSE_URL_FROM_PATH', 'LongInt').SetInt( PARSE_PATH_FROM_URL + 1);
10680: const 'PARSE_MIME', 'LongInt').SetInt( PARSE_URL_FROM_PATH + 1);
10681: const 'PARSE_SERVER', 'LongInt').SetInt( PARSE_MIME + 1);
10682: const 'PARSE_SCHEMA', 'LongInt').SetInt( PARSE_SERVER + 1);
10683: const 'PARSE_SITE', 'LongInt').SetInt( PARSE_SCHEMA + 1);
10684: const 'PARSE_DOMAIN', 'LongInt').SetInt( PARSE_SITE + 1);

```

```

10685: const 'PARSE_LOCATION', 'LongInt').SetInt( PARSE_DOMAIN + 1);
10686: const 'PARSE_SECURITY_DOMAIN', 'LongInt').SetInt( PARSE_LOCATION + 1);
10687: const 'PSU_DEFAULT', 'LongInt').SetInt( 1);
10688: const 'PSU_SECURITY_URL_ONLY', 'LongInt').SetInt( PSU_DEFAULT + 1);
10689: const 'QUERY_EXPIRATION_DATE', 'LongInt').SetInt( 1);
10690: const 'QUERY_TIME_OF_LAST_CHANGE', 'LongInt').SetInt( QUERY_EXPIRATION_DATE + 1);
10691: const 'QUERY_CONTENT_ENCODING', 'LongInt').SetInt( QUERY_TIME_OF_LAST_CHANGE + 1);
10692: const 'QUERY_CONTENT_TYPE', 'LongInt').SetInt( QUERY_CONTENT_ENCODING + 1);
10693: const 'QUERY_REFRESH', 'LongInt').SetInt( QUERY_CONTENT_TYPE + 1);
10694: const 'QUERY_RECOMBINE', 'LongInt').SetInt( QUERY_REFRESH + 1);
10695: const 'QUERY_CAN_NAVIGATE', 'LongInt').SetInt( QUERY_RECOMBINE + 1);
10696: const 'QUERY_USES_NETWORK', 'LongInt').SetInt( QUERY_CAN_NAVIGATE + 1);
10697: const 'QUERY_IS_CACHED', 'LongInt').SetInt( QUERY_USES_NETWORK + 1);
10698: const 'QUERY_IS_INSTALLEDEENTRY', 'LongInt').SetInt( QUERY_IS_CACHED + 1);
10699: const 'QUERY_IS_CACHED_OR_MAPPED', 'LongInt').SetInt( QUERY_IS_INSTALLEDEENTRY + 1);
10700: const 'QUERY_USES_CACHE', 'LongInt').SetInt( QUERY_IS_CACHED_OR_MAPPED + 1);
10701: const 'QUERY_IS_SECURE', 'LongInt').SetInt( QUERY_USES_CACHE + 1);
10702: const 'QUERY_IS_SAFE', 'LongInt').SetInt( QUERY_IS_SECURE + 1);
10703: SIRegister_IInternetProtocolInfo(CL);
10704: IOInet, 'IInternet');
10705: IOInetBindInfo, 'IInternetBindInfo');
10706: IOInetProtocolRoot, 'IInternetProtocolRoot');
10707: IOInetProtocol, 'IInternetProtocol');
10708: IOInetProtocolSink, 'IInternetProtocolSink');
10709: IOInetProtocolInfo, 'IInternetProtocolInfo');
10710: IOInetSession, 'IInternetSession');
10711: IOInetPriority, 'IInternetPriority');
10712: IOInetThreadSwitch, 'IInternetThreadSwitch');
10713: //Function CopyBindInfo( const cbiSrc : TBindInfo; var biDest : TBindInfo ) : HRESULT);
10714: //Procedure ReleaseBindInfo( const bindinfo : TBindInfo);
10715: // const 'INET_E_USE_DEFAULT_PROTOCOLHANDLER', 'LongWord').SetUInt( HRESULT ( $800C0011 ));
10716: // const 'INET_E_USE_DEFAULT_SETTING', 'LongWord').SetUInt( HRESULT ( $800C0012 ));
10717: //const 'INET_E_DEFAULT_ACTION', 'LongWord').SetUInt( HRESULT ( $800C0011 ));
10718: //const 'INET_E_QUERYOPTION_UNKNOWN', 'LongWord').SetUInt( HRESULT ( $800C0013 ));
10719: //const 'INET_E_REDIRECTING', 'LongWord').SetUInt( HRESULT ( $800C0014 ));
10720: const 'PROTOCOLFLAG_NO_PICS_CHECK', 'LongWord').SetUInt( $00000001);
10721: SIRegister_IInternetSecurityMgrSite(CL);
10722: const 'MUTZ_NOSAVEDFILECHECK', 'LongWord').SetUInt( $00000001);
10723: const 'MUTZ_ISFILE', 'LongWord').SetUInt( $00000002);
10724: const 'MUTZ_ACCEPT_WILDCARD_SCHEME', 'LongWord').SetUInt( $00000080);
10725: const 'MUTZ_ENFORCERESTRICTED', 'LongWord').SetUInt( $00000100);
10726: const 'MUTZ_REQUIRESAVEDFILECHECK', 'LongWord').SetUInt( $00000400);
10727: const 'MAX_SIZE_SECURITY_ID', 'LongInt').SetInt( 512);
10728: const 'PUAF_DEFAULT', 'LongWord').SetUInt( $00000000);
10729: const 'PUAF_NOUI', 'LongWord').SetUInt( $00000001);
10730: const 'PUAF_ISFILE', 'LongWord').SetUInt( $00000002);
10731: const 'PUAF_WARN_IF_DENIED', 'LongWord').SetUInt( $00000004);
10732: const 'PUAF_FORCEUI_FOREGROUND', 'LongWord').SetUInt( $00000008);
10733: const 'PUAF_CHECK_TIFS', 'LongWord').SetUInt( $00000010);
10734: const 'PUAF_DONTCHECKBOXINDIALOG', 'LongWord').SetUInt( $00000020);
10735: const 'PUAF_TRUSTED', 'LongWord').SetUInt( $00000040);
10736: const 'PUAF_ACCEPT_WILDCARD_SCHEME', 'LongWord').SetUInt( $00000080);
10737: const 'PUAF_ENFORCERESTRICTED', 'LongWord').SetUInt( $00000100);
10738: const 'PUAF_NOSAVEDFILECHECK', 'LongWord').SetUInt( $00000200);
10739: const 'PUAF_REQUIRESAVEDFILECHECK', 'LongWord').SetUInt( $00000400);
10740: const 'PUAF_LMZ_UNLOCKED', 'LongWord').SetUInt( $00010000);
10741: const 'PUAF_LMZ_LOCKED', 'LongWord').SetUInt( $00020000);
10742: const 'PUAF_DEFAULTZONEPOL', 'LongWord').SetUInt( $00040000);
10743: const 'PUAF_NPL_USE_LOCKED_IF_RESTRICTed', 'LongWord').SetUInt( $00080000);
10744: const 'PUAF_NOUIIFLOCKED', 'LongWord').SetUInt( $00100000);
10745: const 'PUAFOUT_DEFAULT', 'LongWord').SetUInt( $0);
10746: const 'PUAFOUT_ISLOCKZONEPOLICY', 'LongWord').SetUInt( $1);
10747: const 'SZM_CREATE', 'LongWord').SetUInt( $00000000);
10748: const 'SZM_DELETE', 'LongWord').SetUInt( $00000001);
10749: SIRegister_IInternetSecurityManager(CL);
10750: SIRegister_IInternetHostSecurityManager(CL);
10751: SIRegister_IInternetSecurityManagerEx(CL);
10752: const 'URLACTION_MIN', 'LongWord').SetUInt( $00001000);
10753: const 'URLACTION_DOWNLOAD_MIN', 'LongWord').SetUInt( $00001000);
10754: const 'URLACTION_DOWNLOAD_SIGNED_ACTIVEX', 'LongWord').SetUInt( $00001001);
10755: const 'URLACTION_DOWNLOAD_UNSIGNED_ACTIVEX', 'LongWord').SetUInt( $00001004);
10756: const 'URLACTION_DOWNLOAD_CURR_MAX', 'LongWord').SetUInt( $00001004);
10757: const 'URLACTION_DOWNLOAD_MAX', 'LongWord').SetUInt( $000011FF);
10758: const 'URLACTION_ACTIVEX_MIN', 'LongWord').SetUInt( $00001200);
10759: const 'URLACTION_ACTIVEX_RUN', 'LongWord').SetUInt( $00001200);
10760: const 'URLACTION_ACTIVEX_OVERRIDE_OBJECT_SAFETY', 'LongWord').SetUInt( $00001201);
10761: const 'URLACTION_ACTIVEX_OVERRIDE_DATA_SAFETY', 'LongWord').SetUInt( $00001202);
10762: const 'URLACTION_ACTIVEX_OVERRIDE_SCRIPT_SAFETY', 'LongWord').SetUInt( $00001203);
10763: const 'URLACTION_SCRIPT_OVERRIDE_SAFETY', 'LongWord').SetUInt( $00001401);
10764: const 'URLACTION_ACTIVEX_CONFIRM_NOOBJECTSAFETY', 'LongWord').SetUInt( $00001204);
10765: const 'URLACTION_ACTIVEX_TREATASUNTRUSTED', 'LongWord').SetUInt( $00001205);
10766: const 'URLACTION_ACTIVEX_NO_WEBOC_SCRIPT', 'LongWord').SetUInt( $00001206);
10767: const 'URLACTION_ACTIVEX_CURR_MAX', 'LongWord').SetUInt( $00001206);
10768: const 'URLACTION_ACTIVEX_MAX', 'LongWord').SetUInt( $000013FF);
10769: const 'URLACTION_SCRIPT_MIN', 'LongWord').SetUInt( $00001400);
10770: const 'URLACTION_SCRIPT_RUN', 'LongWord').SetUInt( $00001400);
10771: const 'URLACTION_SCRIPT_JAVA_USE', 'LongWord').SetUInt( $00001402);
10772: const 'URLACTION_SCRIPT_SAFE_ACTIVEX', 'LongWord').SetUInt( $00001405);
10773: const 'URLACTION_SCRIPT_CURR_MAX', 'LongWord').SetUInt( $00001405);

```

```

10774: const 'URLACTION_SCRIPT_MAX', 'LongWord').SetUInt( $000015FF);
10775: const 'URLACTION_HTML_MIN', 'LongWord').SetUInt( $00001600);
10776: const 'URLACTION_HTML_SUBMIT_FORMS', 'LongWord').SetUInt( $00001601);
10777: const 'URLACTION_HTML_SUBMIT_FORMS_FROM', 'LongWord').SetUInt( $00001602);
10778: const 'URLACTION_HTML_SUBMIT_FORMS_TO', 'LongWord').SetUInt( $00001603);
10779: const 'URLACTION_HTML_FONT_DOWNLOAD', 'LongWord').SetUInt( $00001604);
10780: const 'URLACTION_HTML_JAVA_RUN', 'LongWord').SetUInt( $00001605);
10781: const 'URLACTION_HTML_CURR_MAX', 'LongWord').SetUInt( $00001605);
10782: const 'URLACTION_HTML_MAX', 'LongWord').SetUInt( $000017FF);
10783: const 'URLACTION_SHELL_MIN', 'LongWord').SetUInt( $00001800);
10784: const 'URLACTION_SHELL_INSTALL_DTITEMS', 'LongWord').SetUInt( $00001800);
10785: const 'URLACTION_SHELL_MOVE_OR_COPY', 'LongWord').SetUInt( $00001802);
10786: const 'URLACTION_SHELL_FILE_DOWNLOAD', 'LongWord').SetUInt( $00001803);
10787: const 'URLACTION_SHELL_VERB', 'LongWord').SetUInt( $00001804);
10788: const 'URLACTION_SHELL_WEBVIEW_VERB', 'LongWord').SetUInt( $00001805);
10789: const 'URLACTION_SHELL_SHELLEXECUTE', 'LongWord').SetUInt( $00001806);
10790: const 'URLACTION_SHELL_EXECUTE_HIGHRISK', 'LongWord').SetUInt( $00001806);
10791: const 'URLACTION_SHELL_EXECUTE_MODRISK', 'LongWord').SetUInt( $00001807);
10792: const 'URLACTION_SHELL_EXECUTE_LOWRISK', 'LongWord').SetUInt( $00001808);
10793: const 'URLACTION_SHELL_POPUPMGR', 'LongWord').SetUInt( $00001809);
10794: const 'URLACTION_SHELL_CURR_MAX', 'LongWord').SetUInt( $00001809);
10795: const 'URLACTION_SHELL_MAX', 'LongWord').SetUInt( $000019ff);
10796: const 'URLACTION_NETWORK_MIN', 'LongWord').SetUInt( $00001A00);
10797: const 'URLACTION_CREDENTIALS_USE', 'LongWord').SetUInt( $00001A00);
10798: const 'URLPOLICY_CREDENTIALS_SILENT_LOGON_OK', 'LongWord').SetUInt( $00000000);
10799: const 'URLPOLICY_CREDENTIALS_MUST_PROMPT_USER', 'LongWord').SetUInt( $00010000);
10800: const 'URLPOLICY_CREDENTIALS_CONDITIONAL_PROMPT', 'LongWord').SetUInt( $00020000);
10801: const 'URLPOLICY_CREDENTIALS_ANONYMOUS_ONLY', 'LongWord').SetUInt( $00030000);
10802: const 'URLACTION_AUTHENTICATE_CLIENT', 'LongWord').SetUInt( $00001A01);
10803: const 'URLPOLICY_AUTHENTICATE_CLEARTEXT_OK', 'LongWord').SetUInt( $00000000);
10804: const 'URLPOLICY_AUTHENTICATE_CHALLENGE_RESPONSE', 'LongWord').SetUInt( $00010000);
10805: const 'URLPOLICY_AUTHENTICATE_MUTUAL_ONLY', 'LongWord').SetUInt( $00030000);
10806: const 'URLACTION_NETWORK_CURR_MAX', 'LongWord').SetUInt( $00001A01);
10807: const 'URLACTION_NETWORK_MAX', 'LongWord').SetUInt( $00001BFF);
10808: const 'URLACTION_JAVA_MIN', 'LongWord').SetUInt( $00001C00);
10809: const 'URLACTION_JAVA_PERMISSIONS', 'LongWord').SetUInt( $00001C00);
10810: const 'URLPOLICY_JAVA_PROHIBIT', 'LongWord').SetUInt( $00000000);
10811: const 'URLPOLICY_JAVA_HIGH', 'LongWord').SetUInt( $00010000);
10812: const 'URLPOLICY_JAVA_MEDIUM', 'LongWord').SetUInt( $00020000);
10813: const 'URLPOLICY_JAVA_LOW', 'LongWord').SetUInt( $00030000);
10814: const 'URLPOLICY_JAVA_CUSTOM', 'LongWord').SetUInt( $00800000);
10815: const 'URLACTION_JAVA_CURR_MAX', 'LongWord').SetUInt( $00001C00);
10816: const 'URLACTION_JAVA_MAX', 'LongWord').SetUInt( $00001CFF);
10817: const 'URLACTION_INFODELIVERY_MIN', 'LongWord').SetUInt( $00001D00);
10818: const 'URLACTION_INFODELIVERY_NO_ADDING_CHANNELS', 'LongWord').SetUInt( $00001D00);
10819: const 'URLACTION_INFODELIVERY_NO_EDITING_CHANNELS', 'LongWord').SetUInt( $00001D01);
10820: const 'URLACTION_INFODELIVERY_NO_REMOVING_CHANNELS', 'LongWord').SetUInt( $00001D02);
10821: const 'URLACTION_INFODELIVERY_NO_ADDING_SUBSCRIPTIONS', 'LongWord').SetUInt( $00001D03);
10822: const 'URLACTION_INFODELIVERY_NO_EDITING_SUBSCRIPTIONS', 'LongWord').SetUInt( $00001D04);
10823: const 'URLACTION_INFODELIVERY_NO_REMOVING_SUBSCRIPTIONS', 'LongWord').SetUInt( $00001D05);
10824: const 'URLACTION_INFODELIVERY_NO_CHANNEL_LOGGING', 'LongWord').SetUInt( $00001D06);
10825: const 'URLACTION_INFODELIVERY_CURR_MAX', 'LongWord').SetUInt( $00001D06);
10826: const 'URLACTION_INFODELIVERY_MAX', 'LongWord').SetUInt( $00001Dff);
10827: const 'URLACTION_CHANNEL_SOFTDIST_MIN', 'LongWord').SetUInt( $00001E00);
10828: const 'URLACTION_CHANNEL_SOFTDIST_PERMISSIONS', 'LongWord').SetUInt( $00001E05);
10829: const 'URLPOLICY_CHANNEL_SOFTDIST_PROHIBIT', 'LongWord').SetUInt( $00010000);
10830: const 'URLPOLICY_CHANNEL_SOFTDIST_PRECACHE', 'LongWord').SetUInt( $00020000);
10831: const 'URLPOLICY_CHANNEL_SOFTDIST_AUTOINSTALL', 'LongWord').SetUInt( $00030000);
10832: const 'URLACTION_CHANNEL_SOFTDIST_MAX', 'LongWord').SetUInt( $00001EFF);
10833: const 'URLACTION_BEHAVIOR_MIN', 'LongWord').SetUInt( $00002000);
10834: const 'URLACTION_BEHAVIOR_RUN', 'LongWord').SetUInt( $00002000);
10835: const 'URLPOLICY_BEHAVIOR_CHECK_LIST', 'LongWord').SetUInt( $00010000);
10836: const 'URLACTION_FEATURE_MIN', 'LongWord').SetUInt( $00002100);
10837: const 'URLACTION_FEATURE_MIME_SNIFFING', 'LongWord').SetUInt( $00002100);
10838: const 'URLACTION_FEATURE_ZONE_ELEVATION', 'LongWord').SetUInt( $00002101);
10839: const 'URLACTION_FEATURE_WINDOW_RESTRICTIONS', 'LongWord').SetUInt( $00002102);
10840: const 'URLACTION_AUTOMATIC_DOWNLOAD_UI_MIN', 'LongWord').SetUInt( $00002200);
10841: const 'URLACTION_AUTOMATIC_DOWNLOAD_UI', 'LongWord').SetUInt( $00002200);
10842: const 'URLACTION_AUTOMATIC_ACTIVEX_UI', 'LongWord').SetUInt( $00002201);
10843: const 'URLACTION_ALLOW_RESTRICTEDPROTOCOLS', 'LongWord').SetUInt( $00002300);
10844: const 'URLPOLICY_ALLOW', 'LongWord').SetUInt( $00);
10845: const 'URLPOLICY_QUERY', 'LongWord').SetUInt( $01);
10846: const 'URLPOLICY_DISALLOW', 'LongWord').SetUInt( $03);
10847: const 'URLPOLICY_NOTIFY_ON_ALLOW', 'LongWord').SetUInt( $10);
10848: const 'URLPOLICY_NOTIFY_ON_DISALLOW', 'LongWord').SetUInt( $20);
10849: const 'URLPOLICY_LOG_ON_ALLOW', 'LongWord').SetUInt( $40);
10850: const 'URLPOLICY_LOG_ON_DISALLOW', 'LongWord').SetUInt( $80);
10851: const 'URLPOLICY_MASK_PERMISSIONS', 'LongWord').SetUInt( $0F);
10852: Function GetUrlPolicyPermissions( dw : DWORD ) : DWORD;
10853: Function SetUrlPolicyPermissions( dw, dw2 : DWORD ) : DWORD;
10854: const 'URLZONE_PREDEFINED_MIN', 'LongInt').SetInt( 0);
10855: const 'URLZONE_LOCAL_MACHINE', 'LongInt').SetInt( 0);
10856: const 'URLZONE_INTRANET', 'LongInt').SetInt( URLZONE_LOCAL_MACHINE + 1);
10857: const 'URLZONE_TRUSTED', 'LongInt').SetInt( URLZONE_INTRANET + 1);
10858: const 'URLZONE_INTERNET', 'LongInt').SetInt( URLZONE_TRUSTED + 1);
10859: const 'URLZONE_UNTRUSTED', 'LongInt').SetInt( URLZONE_INTERNET + 1);
10860: const 'URLZONE_PREDEFINED_MAX', 'LongInt').SetInt( 999);
10861: const 'URLZONE_USER_MIN', 'LongInt').SetInt( 1000);
10862: const 'URLZONE_USER_MAX', 'LongInt').SetInt( 10000);

```

```

10863: const 'URLTEMPLATE_CUSTOM', 'LongWord').SetUInt( $00000000);
10864: const 'URLTEMPLATE_PREDEFINED_MIN', 'LongWord').SetUInt( $00010000);
10865: const 'URLTEMPLATE_LOW', 'LongWord').SetUInt( $00010000);
10866: const 'URLTEMPLATE_MEDIUM', 'LongWord').SetUInt( $00011000);
10867: const 'URLTEMPLATE_HIGH', 'LongWord').SetUInt( $00012000);
10868: const 'URLTEMPLATE_PREDEFINED_MAX', 'LongWord').SetUInt( $00020000);
10869: const 'MAX_ZONE_PATH', 'LongInt').SetInt( 260);
10870: const 'MAX_ZONE_DESCRIPTION', 'LongInt').SetInt( 200);
10871: const 'ZAFLAGS_CUSTOM_EDIT', 'LongWord').SetUInt( $00000001);
10872: const 'ZAFLAGS_ADD_SITES', 'LongWord').SetUInt( $00000002);
10873: const 'ZAFLAGS_REQUIRE_VERIFICATION', 'LongWord').SetUInt( $00000004);
10874: const 'ZAFLAGS_INCLUDE_PROXY_OVERRIDE', 'LongWord').SetUInt( $00000008);
10875: const 'ZAFLAGS_INCLUDE_INTRANET_SITES', 'LongWord').SetUInt( $00000010);
10876: const 'ZAFLAGS_NO_UI', 'LongWord').SetUInt( $00000020);
10877: const 'ZAFLAGS_SUPPORTS_VERIFICATION', 'LongWord').SetUInt( $00000040);
10878: const 'ZAFLAGS_UNC_AS_INTRANET', 'LongWord').SetUInt( $00000080);
10879: const 'ZAFLAGS_USE_LOCKED_ZONES', 'LongWord').SetUInt( $00010000);
10880: //PZoneAttributes', '^TZoneAttributes // will not work');
10881: _ZONEATTRIBUTES', 'record cbSize:ULONG;szDisplayName:array [0..260-1] of Char;szDescription:array [0..200 -
1] of Char;szIconPath: array [0..260 - 1] of char; dwTemplateMinLevel: DWORD; dwTemplateRecommended:
DWORD; dwTemplateCurrentLevel: DWORD; dwFlags: DWORD; end');
10882: { _ZONEATTRIBUTES = packed record
10883:   cbSize: ULONG;
10884:   szDisplayName: array [0..260 - 1] of WideChar;
10885:   szDescription: array [0..200 - 1] of WideChar;
10886:   szIconPath: array [0..260 - 1] of WideChar;
10887:   dwTemplateMinLevel: DWORD;
10888:   dwTemplateRecommended: DWORD;
10889:   dwTemplateCurrentLevel: DWORD;
10890:   dwFlags: DWORD;
10891: end; }
10892: TZoneAttributes', '_ZONEATTRIBUTES');
10893: ZONEATTRIBUTES', '_ZONEATTRIBUTES');
10894: const 'URLZONEREG_DEFAULT', 'LongInt').SetInt( 0);
10895: const 'URLZONEREG_HKLM', 'LongInt').SetInt( URLZONEREG_DEFAULT + 1);
10896: const 'URLZONEREG_HKCU', 'LongInt').SetInt( URLZONEREG_HKLM + 1);
10897: SRegister_IInternetZoneManager(CL);
10898: SRegister_IInternetZoneManagerEx(CL);
10899: const 'SOFTDIST_FLAG_USAGE_EMAIL', 'LongWord').SetUInt( $00000001);
10900: const 'SOFTDIST_FLAG_USAGE_PRECACHE', 'LongWord').SetUInt( $00000002);
10901: const 'SOFTDIST_FLAG_USAGE_AUTOINSTALL', 'LongWord').SetUInt( $00000004);
10902: const 'SOFTDIST_FLAG_DELETE_SUBSCRIPTION', 'LongWord').SetUInt( $00000008);
10903: const 'SOFTDIST_ADSTATE_NONE', 'LongWord').SetUInt( $00000000);
10904: const 'SOFTDIST_ADSTATE_AVAILABLE', 'LongWord').SetUInt( $00000001);
10905: const 'SOFTDIST_ADSTATE_DOWNLOADED', 'LongWord').SetUInt( $00000002);
10906: const 'SOFTDIST_ADSTATE_INSTALLED', 'LongWord').SetUInt( $00000003);
10907: //PCodeBaseHold', '^TCodeBaseHold // will not work');
10908: _tagCODEBASEHOLD', 'record cbSize: ULONG; szDistUnit: LPWSTR; '
10909: + 'szCodeBase: LPWSTR; dwVersionMS: DWORD; dwVersionLS: DWORD; dwStyle: DWORD; end');
10910: TCodeBaseHold', '_tagCODEBASEHOLD');
10911: CODEBASEHOLD', '_tagCODEBASEHOLD');
10912: //PSoftDistInfo', '^TSoftDistInfo // will not work');
10913: _tagSOFTDISTINFO', 'record cbSize: ULONG; dwFlags: DWORD; dwAd
10914: + 'State: DWORD; szTitle: LPWSTR; szAbstract: LPWSTR; szHREF: LPWSTR; dwI
10915: + 'nstalledVersionMS: DWORD; dwInstalledVersionLS: DWORD; dwUpdateVersionMS
10916: + ': DWORD; dwUpdateVersionLS: DWORD; dwAdvertisedVersionMS: DWORD; dwAdve
10917: + 'rtisedVersionLS: DWORD; dwReserved: DWORD; end');
10918: TSoftDistInfo', '_tagSOFTDISTINFO');
10919: SOFTDISTINFO', '_tagSOFTDISTINFO');
10920: SRegister_ISoftDistExt(CL);
10921: Function GetSoftwareUpdateInfo( szDistUnit: LPCWSTR; var dsi: TSoftDistInfo): HRESULT);
10922: Function SetSoftwareUpdateAdvertisementState( szDistUnit: LPCWSTR; dwAdState, dwAdvertisedVersionMS,
dwAdvertisedVersionLS: DWORD): HRESULT);
10923: SRegister_IDataFilter(CL);
10924: // PProtocolFilterData', '^TProtocolFilterData // will not work');
10925: _tagPROTOCOLFILTERDATA', 'record cbSize: DWORD; ProtocolSink: '
10926: + 'IInternetProtocolSink; Protocol: IInternetProtocol; Unk: IUnknown; dwFil
10927: + 'terFlags: DWORD; end');
10928: TProtocolFilterData', '_tagPROTOCOLFILTERDATA');
10929: PROTOCOLFILTERDATA', '_tagPROTOCOLFILTERDATA');
10930: //PDataInfo', '^TDataInfo // will not work');
10931: _tagDATAINFO', 'record ulTotalSize: ULONG; ulAvrPacketSize: UL
10932: + 'ONG; ulConnectSpeed: ULONG; ulProcessorSpeed: ULONG; end');
10933: TDataInfo', '_tagDATAINFO');
10934: DATAINFO', '_tagDATAINFO');
10935: SRegister_IEncodingFilterFactory(CL);
10936: Function IsLoggingEnabled( pszUrl: PChar): BOOL);
10937: //Function IsLoggingEnabledA( pszUrl: PAnsiChar): BOOL);
10938: //Function IsLoggingEnabledW( pszUrl: PWideChar): BOOL);
10939: //PHitLoggingInfo', '^THitLoggingInfo // will not work');
10940: _tagHIT_LOGGING_INFO', 'record dwStructSize: DWORD; lpszLoggedU
10941: + 'rlName: LPSTR; StartTime: TSystemTime; EndTime: TSystemTime; lpszExtendedInfo: LPSTR; end');
10942: THitLoggingInfo', '_tagHIT_LOGGING_INFO');
10943: HIT_LOGGING_INFO', '_tagHIT_LOGGING_INFO');
10944: Function WriteHitLogging( const LoggingInfo: THitLoggingInfo): BOOL);
10945: end;
10946:
10947: procedure SRegister_UIntList(CL: TPSPascalCompiler);
10948: begin

```



```

10949: CL.AddConstantN('dffmaxlistsize','LongInt').SetInt( maxint div 32);
10950: CL.AddClassN(CL.FindClass('TOBJECT'),'TIntList');
10951: //PintItem', '^TintItem // will not work');
10952: TIntItem', 'record FInt : int64; FObject : TObject; end');
10953: //PintItemList', '^TintItemList // will not work');
10954: SIRegister_TIntList(CL);
10955: end;
10956:
10957: procedure SIRegister_UGeometry(CL: TPSPascalCompiler);
10958: begin
10959:   TrealPoint', 'record x : extended; y : extended; end');
10960:   Tline', 'record p1 : TPoint; p2 : TPoint; end');
10961:   TrealLine', 'record p1 : TRealPoint; p2 : TRealPoint; end');
10962:   TCircle', 'record cx : integer; cy : integer; r : integer; end');
10963:   TRealCircle', 'record cx : extended; cy : extended; r : extended; end');
10964:   PPResult', '( PPOutside, PPInside, PPVertex, PPEdge, PPErrors );
10965:   Function PointCircleTangentLines( const C : TCircle; const P : TPoint; var L1, L2 : TLine ) : boolean';
10966:   Function CircleCircleExtTangentLines( C1, C2 : TCircle; var C3 : TCircle; var L1, L2, PL1, PL2, TL1, TL2
    : TLine ) : Boolean';
10967: end;
10968:
10969: procedure SIRegister_UAstronomy(CL: TPSPascalCompiler);
10970: begin
10971:   TCoordType', '( ctUnknown, ctAzAlt, ctEclLonLat, ctRADecl, ctHADec, ctGallLonLat );
10972:   TDTType', '( ttLocal, ttUT, ttGST, ttLST );
10973:   TRPoint', 'record x : extended; y : extended; CoordType : TCoordType; end';
10974:   TSunrec', 'record TrueEclLon : extended; AppEclLon : extended; A'
    + 'UDistance : extended; TrueHADec : TRPoint; TrueRADecl : TRPoint; TrueAzAl'
    + 't : TRPoint; AppHADec : TRPoint; AppRADecl : TRPoint; AppAzAlt : TRPoint;'
    + 'SunMeanAnomaly : extended; end';
10978:   TMoonRec', 'record ECLonLat : TRPoint; AZAlt : TRPoint; RADecl : '
    + 'TRPoint; HorizontalParallax : extended; AngularDiameter : extended; KMtoE'
    + 'arth : extended; Phase : extended; end';
10981:   TMoonEclipseAdd', 'record UmbralStartTime : TDateTime; UmbralEnd'
    + 'Time : TDateTime; TotalStartTime : TDateTime; TotalEndTime : TDateTime; end';
10983:   TEclipseRec', 'record Msg : string; Status : integer; FirstConta'
    + 'ct : TDateTime; LastContact : TDateTime; Magnitude : Extended; MaxeclipseUTime : TDateTime; end';
10985:   TPlanet', '( MERCURY, VENUS, MARS, JUPITER, SATURN, URANUS, NEPTUNE, PLUTO );
10986:   TPlanetRec', 'record AsOf : TDateTime; Name : string; MeanLon : '
    + 'extended; MeanLonMotion : extended; LonOfPerihelion : extended; Eccentrici'
    + 'ty : extended; Inclination : extended; LonAscendingNode : extended; SemiMa'
    + 'jorAxis : extended; AngularDiameter : extended; Magnitude : extended; end';
10990:   TPlanetLocRec', 'record PlanetBaseData : TPlanetrec; HelioCentri'
    + 'cLonLat : TRpoint; RadiusVector : extended; UncorrectedEarthDistance : ext'
    + 'ended; GeoEclLonLat : TRpoint; CorrectedEarthDistance : extended; ApparentRaDecl : TRPoint; end';
10993:   SIRegister_TAstronomy(CL);
10994:   Function AngleToStr( angle : extended ) : string';
10995:   Function StrToAngle( s : string; var angle : extended ) : boolean';
10996:   Function HoursToStr24( t : extended ) : string';
10997:   Function RPoint( x, y : extended ) : TRPoint';
10998:   Function getStimename( t : TDTType ) : string';
10999: end;
11000:
11001: procedure SIRegister_UCardComponentV2(CL: TPSPascalCompiler);
11002: begin
11003:   TCardValue', 'Integer';
11004:   TCardSuit', '( Spades, Diamonds, Clubs, Hearts );
11005:   TShortSuit', '( S, D, C, H );
11006:   TDecks', '( Standard1, Standard2, Fishes1, Fishes2, Beach, Leaves1, Leaves2, Robot, Roses, Shell, Castle, Hand );
11007:   SIRegister_TCard(CL);
11008:   SIRegister_TDeck(CL);
11009: end;
11010:
11011: procedure SIRegister_UTGraphSearch(CL: TPSPascalCompiler);
11012: begin
11013:   tMethodCall', 'Procedure';
11014:   tVerbosCall', 'Procedure ( s : string )';
11015:   // PTEdge', '^TEdge // will not work';
11016:   TEdge', 'record FromNodeIndex : Integer; ToNodeIndex : Integer; '
    + 'Weight : integer; work : integer; Len : integer; Highlight : boolean; end';
11018:   SIRegister_TNode(CL);
11019:   SIRegister_TGraphList(CL);
11020: end;
11021:
11022: procedure SIRegister_UParser10(CL: TPSPascalCompiler);
11023: begin
11024:   ParserFloat', 'extended';
11025:   //PParserFloat', '^ParserFloat // will not work';
11026:   TDFFToken', '( variab, constant, minus, sum, diff, prod, divis, mod'
    + 'ulo, IntDiv, IntDIVZ, integerpower, realpower, square, third, fourth, FuncOneVar, FuncTwoVar );
11028:   //POperation', '^TOperation // will not work';
11029:   TDFFOperation', 'record Arg1 : PParserFloat; Arg2 : PParserFloat; D'
    + 'est : PParserFloat; NextOperation : POperation; Operation : TMathProcedure'
    + 'Token : TDFFToken; end';
11032:   TMathProcedure', 'procedure (AnOperation: TDFFOperation)';
11033:   (CL.FindClass('TOBJECT'),'EMathParserError');
11034:   CL.FindClass('TOBJECT'),'ESyntaxError');
11035:   (CL.FindClass('TOBJECT'),'EExpressionHasBlanks');
11036:   (CL.FindClass('TOBJECT'),'EExpressionTooComplex');

```

```

11037: (CL.FindClass('TOBJECT'),'ETooManyNestings');
11038: (CL.FindClass('TOBJECT'),'EMissMatchingBracket');
11039: (CL.FindClass('TOBJECT'),'EBadName');
11040: (CL.FindClass('TOBJECT'),'EParserInternalError');
11041: ('TExParserExceptionEvent', 'Procedure ( Sender : TObject; E : Exception)');
11042: SIRegister_TCustomParser(CL);
11043: SIRegister_TExParser(CL);
11044: end;
11045:
11046: procedure SIRegister_UcomboV2(CL: TPSPascalCompiler);
11047: begin
11048: CL.AddConstantN('UMaxEntries','LongInt').SetInt( 600);
11049: CL.AddTypeS('TCombotype', '( Combinations, Permutations, CombinationsDown, Pe'
11050: + 'rmutationsDown, CombinationsCoLex, CombinationsCoLexDown, PermutationsRepe'
11051: + 'at, PermutationsWithRep, PermutationsRepeatDown, CombinationsWithrep, Comb'
11052: + 'inationsRepeat, CombinationsRepeatDown )');
11053: CL.AddTypeS('TByteArray64', 'array[0..600 + 1] of int64;');
11054: SIRegister_TComboSet(CL);
11055: end;
11056:
11057: procedure SIRegister_cyIEUtils(CL: TPSPascalCompiler);
11058: begin
11059: CL.AddTypeS('TwbPageSetup', 'record font : String; footer : String; header : '
11060: + 'String; margin_bottom : String; margin_left : String; margin_right : Strin'
11061: + 'g; margin_top : String; Print_Background : String; Shrink_To_Fit : String; end');
11062: end;
11063:
11064: procedure SIRegister_cyBaseComm(CL: TPSPascalCompiler);
11065: begin
11066: TcyCommandType, '( ctDeveloperDefined, ctUserDefined )';
11067: TStreamContentype, '( scDeveloperDefined, scUserDefined, scString )';
11068: TProcOnReceiveCommand, 'Procedure (Sender : TObject; aCommand: Word; userParam : Integer)';
11069: TProcOnReceiveString, 'Procedure ( Sender : TObject; fromBaseCo'
11070: + 'mmHandle : THandle; aString : String; userParam : Integer)';
11071: TProcOnReceiveMemoryStream, 'Procedure ( Sender : TObject; from'
11072: + 'BaseCommHandle : THandle; aStream : TMemoryStream; userParam : Integer)';
11073: SIRegister_TcyBaseComm(CL);
11074: CL.AddConstantN('MsgCommand','LongInt').SetInt( WM_USER + 1);
11075: CL.AddConstantN('MsgResultOk','LongInt').SetInt( 99);
11076: CL.AddDelphiFunction('Function ValidateFileMappingName( aName : String) : String');
11077: end;
11078:
11079: procedure SIRegister_cyAppInstances(CL: TPSPascalCompiler);
11080: begin
11081: CL.AddTypeS('TcyInstanceInfo', 'record BaseComHandle : THandle; OwnerWindowHa'
11082: + 'ndle : THandle; ApplicationHandle : THandle; Instances : Integer; MaxInstances : Word; end');
11083: CL.AddTypeS('TProcOnReceiveCmdLine', 'Procedure ( Sender : TObject; fromInstance : TcyInstanceInfo;
Parameters : TStrings)');
11084: CL.AddTypeS('TProcOnServerExists', 'Procedure ( Sender : TObject; var DoExitProcess, DoSendCmdLine,
ShowServer : Boolean)');
11085: SIRegister_TcyAppInstances(CL);
11086: CL.AddConstantN('CmdLineParam','LongInt').SetInt( 0);
11087: CL.AddConstantN('sPrefixInstanceID','String').SetString( 'CYAPPINST');
11088: CL.AddConstantN('seNotActive','String').SetString( 'Error: not Active!');
11089: end;
11090:
11091: procedure SIRegister_cyAttract(CL: TPSPascalCompiler);
11092: begin
11093: CL.AddConstantN('constInterval','LongInt').SetInt( 3000);
11094: CL.AddClassN(CL.FindClass('TOBJECT'),'TcyAttract');
11095: SIRegister_TAtractFrame(CL);
11096: //CL.AddTypeS('TAttractFrameClass', 'class of TAttractFrame');
11097: SIRegister_TAtractFrames(CL);
11098: SIRegister_TAnimationThreadHandler(CL);
11099: SIRegister_TcyAttract(CL);
11100: end;
11101:
11102: procedure SIRegister_cyDERUtils(CL: TPSPascalCompiler);
11103: begin
11104: DERString, 'String';
11105: DERChar, 'Char';
11106: CL.AddTypeS('TElementsType', '( etText, etExpressionKeyWord, etNumbers, etInt'
11107: + 'eger, etFloat, etPercentage, etwebSite, etWebMail, etMoney, etDate, etTextLine, etParagraph )');
11108: CL.AddTypeS('TElementsTypes', 'set of TElementsType');
11109: CL.AddTypeS('DERNString', 'String');
11110: const DERDecimalSeparator, 'String').SetString( '.');
11111: const DERDefaultChars, 'String')+'%~
_.:0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNopQRSTUVWXYZ';
11112: CL.AddConstantN('DERNDefaultChars','String').SetString( '/%-0123456789abcdefghijklmnopqrstuvwxyz');
11113: end;
11114:
11115: procedure SIRegister_cyDocER(CL: TPSPascalCompiler);
11116: begin
11117: TReadingDirection, '( rdLeft, rdTop, rdRight, rdBottom )';
11118: TReadingDirections, 'set of TReadingDirection';
11119: TOCRCarInfo, 'record Value : String; PageNumber : Integer; Left'
11120: : Integer; Top : Integer; Right : Integer; Bottom : Integer; WordNumber : Integer; end';
11121: CL.AddClassN(CL.FindClass('TOBJECT'),'TcyDocER');
11122: SIRegister_TOCRExpression(CL);

```

```

11123: TSearchValueLocation', '( slFromLeft, slFromTop, slFromRight, slFromBottom )');
11124: TPatternPositionMode', '( ppFromTopLeftPage, ppFromTopLeftKeyword )');
11125: TPatternPageNumberMode', '( ppFromBeginning, ppFromEnd )');
11126: SIRegister_TElement(CL);
11127: //TElementClass', 'class of TElement';
11128: SIRegister_TElements(CL);
11129: TRecognitionElementsMode', '( reKeywords, reValues, reKeywordsAndValues )');
11130: TRecognitionPriorityMode', '( rpKeywordsLenth, rpSinglelineKeywordsLength )');
11131: TRecognitionOption', '( roKeywordsByPriority, roSmartNumbersRec, roSmartWebsiteRec, roSmartKeywordRec )');
11132: TRecognitionOptions', 'set of TRecognitionOption';
11133: TExpressionOption', '( eoDERValueAsString )');
11134: TExpressionOptions', 'set of TExpressionOption';
11135: TLocateExpressionOption', '( lePartialKey, leRelativePositionKey, leInsensitive, leSmartKeywordRec )');
11136: TLocateExpressionOptions', 'set of TLocateExpressionOption';
11137: TValidateElementValueResult', '( veValueOk, veInvalidValue, veInvalidValueStopSearching, veValueTooFar
)');
11138: TProcOnExpressionMerged', 'Procedure ( Sender : TObject; aExpressionIndex, toExpressionIndex :
Integer)';
11139: TProcOnRecognizeElementKeyword', 'Procedure (Sender : TObject; ElementIndex,
ElementKeywordIndex: Integer)';
11140: TProcValidateElementValue', 'Procedure ( Sender : TObject; Eleme
+ 'ntIndex, ElementKeywordIndex, ExpressionKeywordIndex, ExpressionValueIndex'
11141: + ' : Integer; var ValidateElementValueResult : TValidateElementValueResult)';
11142: TProcValidateElementKeyword', 'Procedure ( Sender : TObject; Ele
+ 'mentIndex, ElementKeywordIndex : Integer; ExpressionsList : TStrings; var Accept : Boolean)';
11143: TProcRetrieveElementValuesFromRect', 'Procedure ( Sender : TObje
+ 'ct; ElementIndex, ElementKeywordIndex, ExpressionKeywordIndex: Integer; ExpressionList : TStrings)';
11144: SIRegister_TcyDocER(CL);
11145: end;
11146:
11147: procedure SIRegister_ODBC(CL: TPSPascalCompiler);
11148: begin
11149:   CL.AddTypeS('FApplyEx', '( False, True )');
11150:   CL.AddTypeS('FODBCActionEx', '( ODBC_ADD_DSN, ODBC_CONFIG_DSN, ODBC_REMOVE_DS
+ 'N, ODBC_ADD_SYS_DSN, ODBC_CONFIG_SYS_DSN, ODBC_REMOVE_SYS_DSN )');
11151:   CL.AddTypeS('FSrcTypeEx', '( Access, dBase, Paradox, Excel, FoxPro )');
11152:   SIRegister_TODBC(CL);
11153:   //CL.AddDelphiFunction('Procedure Register');
11154: end;
11155:
11156: procedure SIRegister_cyBaseCommRoomConnector(CL: TPSPascalCompiler);
11157: begin
11158:   CL.AddTypeS('TNickName', 'String');
11159:   CL.AddTypeS('TcyRoomInfo', 'record ConnexionCount : Integer; Tag : Integer; end');
11160:   CL.AddTypeS('TcyConnexionInfo', 'record NickName : TNickName; BaseComHandle :
+ 'THandle; Tag : Integer; end');
11161:   CL.AddTypeS('TcyConnexions', 'array of TcyConnexionInfo');
11162:   SIRegister_TcyBaseCommRoomConnector(CL);
11163:   Const sPrefixInstanceID, 'String').SetString( 'CYCOMMROOM');
11164:   const CmdAddedConnexion, 'LongInt').SetInt( 0);
11165:   const CmdRemovedConnexion, 'LongInt').SetInt( 1);
11166:   const 'CmdUpdatedConnexion', 'LongInt').SetInt( 2);
11167: end;
11168:
11169: procedure SIRegister_cyCommunicate(CL: TPSPascalCompiler);
11170: begin
11171:   CL.AddTypeS('TProcOnRoomCommand', 'Procedure ( Sender : TObject; BaseComHandle : THandle)');
11172:   SIRegister_TcyCommunicate(CL);
11173: end;
11174:
11175: procedure SIRegister_cyBaseContainer(CL: TPSPascalCompiler);
11176: begin
11177:   CL.AddTypeS('TEnterKeyAction', '( enterKeyDefault, enterKeyNothing, enterKeyGotoNextControl )');
11178:   SIRegister_TcyCustomGrid(CL);
11179:   SIRegister_TcyBaseContainer(CL);
11180: end;
11181:
11182: procedure SIRegister_cyIniForm(CL: TPSPascalCompiler);
11183: begin
11184:   SIRegister_TcyTempForm(CL);
11185:   CL.AddTypeS('RAttributes', 'record Top : Integer; Left : Integer; Width : Int
+ 'eger; Height : Integer; Visible : Boolean; State : Integer; end');
11186:   CL.AddTypeS('TProcOnReadFile', Procedure ( Sender : TObject; IniFile : TIniFile; FileVersion : String)');
11187:   CL.AddTypeS('TProcOnReadRegistry', Procedure ( Sender : TObject; Registry :
+ 'TRegistry; RegistryVersion : String)');
11188:   CL.AddTypeS('TProcOnWriteFile', Procedure ( Sender : TObject; IniFile : TIniFile)');
11189:   CL.AddTypeS('TProcOnWriteRegistry', procedure ( Sender : TObject; Registry : TRegistry)');
11190:   CL.AddTypeS('TAttribute', '( atPosition, atSize, atVisible, atMinimized, atMaximized )');
11191:   CL.AddTypeS('TAttributes', 'set of TAttribute');
11192:   CL.AddTypeS('TMode', '( mFile, mRegistry, mBoth )');
11193:   CL.AddTypeS('TIniDirectory', '( idProgramLocation, idCommonAppData, idLocalAppData )');
11194:   CL.AddTypeS('TRegRoot', '(rrKEY_CLASSES_ROOT, rrKEY_CURRENT_USER, rrKEY_LOCAL_MACHINE, rrKEY_USERS,
rrKEY_CURRENT_CONFIG )');
11195:   SIRegister_TcyIniForm(CL);
11196:   //CL.AddConstantN('cNormal', 'LongInt').SetInt( 0);
11197:   //CL.AddConstantN('cMaximized', 'LongInt').SetInt( 1);
11198:   //CL.AddConstantN('cMinimized', 'LongInt').SetInt( 2);
11199: end;
11200:
11201:
11202:
11203:
11204:
11205:
11206:
11207:

```

```

11208: procedure SIRegister_cyVirtualGrid(CL: TPSPascalCompiler);
11209: begin
11210:   TypeS('TSizeMode', ( smManual, smAutoFixed, smAutoStretched ));
11211:   TypeS('TProcGeneratedCell', Procedure ( Sender : TObject; Rect : TRect; CoordX, CoordY : Integer));
11212:   TypeS('TProcRowSize', Procedure ( Sender : TObject; CoordY : Integer; var Size : Word));
11213:   TypeS('TProcColumnSize', Procedure ( Sender : TObject; CoordX : Integer; var Size : Word));
11214:   SIRegister_TcyVirtualGrid(CL);
11215:   CL.AddConstantN('MsgInvalidCells','String').SetString( 'Invalid generated cells!');
11216: end;
11217:
11218: procedure SIRegister_BackgroundWorker_Profiler_WaveUtils(CL: TStrings);
11219: begin
11220:   EBackgroundWorker;
11221:   TBackgroundWorker;
11222:   TWorkEvent2, 'Procedure ( Worker : TBackgroundWorker)';
11223:   TWorkProgressEvent, 'Procedure ( Worker : TBackgroundWorker; PercentDone : Integer)';
11224:   TWorkCompleteEvent, 'Procedure ( Worker : TBackgroundWorker; Cancelled : Boolean)';
11225:   TWorkFeedbackEvent, 'Procedure ( Worker : TBackgroundWorker; FeedbackID, FeedbackValue : Integer)';
11226:   SIRegister_TBackgroundWorker(CL);
11227:   TProfilerTimer
11228:   TTimeUnit
11229:   TTimeInfo
11230:   TProfilerNotifyEvent
11231:   DefaultResType
11232:   TWave
11233:   TMultimediaTimer
11234:   TMS2StrFormat, '( msHMS, msHMS, msMS, msSh, msS, msAh,msA )';
11235:   TPCMChannel, '( cMono, cStereo )';
11236:   TPCMSamplesPerSec, '( ss8000Hz, ss11025Hz, ss22050Hz, ss44100Hz, ss48000Hz )';
11237:   TPCMBitsPerSample, '( bs8Bit, bs16Bit )';
11238:   TPCMFormat, '( nonePCM, Mono8Bit8000Hz, Stereo8bit8000Hz, Mono1
11239:   +6bit8000Hz, Stereo16bit8000Hz, Mono8bit11025Hz, Stereo8bit11025Hz, Mono16b
11240:   +it11025Hz, Stereo16bit11025Hz, Mono8bit22050Hz, Stereo8bit22050Hz, Mono16b
11241:   +it22050Hz, Stereo16bit22050Hz, Mono8bit44100Hz, Stereo8bit44100Hz, Mono16b
11242:   +it44100Hz, Stereo16bit44100Hz, Mono8bit48000Hz, Stereo8bit48000Hz, Mono16b
11243:   +it48000Hz, Stereo16bit48000Hz )';
11244:   TWaveDeviceFormats, 'set of TPCMFormat';
11245:   TWaveOutDeviceSupport, '(dsVolume,dsStereoVolume,dsPitch,dsPlaybackRate,dsPosition,dsAsynchronize,
dsDirectSound);
11246:   TWaveOutDeviceSupports, 'set of TWaveOutDeviceSupport';
11247:   TWaveOutOption, '( woSetVolume, woSetPitch, woSetPlaybackRate )';
11248:   TWaveOutOptions, 'set of TWaveOutOption';
11249:   TStreamOwnership2, '( soReference, soOwned )';
11250:   TWaveStreamState, '( wssReady, wssReading, wssWriting, wssWritingEx )';
11251:   //PrawWave, '^TRawWave // will not work';
11252:   TRawWave, 'record pData : Pointer; dwSize : DWORD; end';
11253:   CL.AddClassN(CL.FindClass('TObject'),'EWaveAudioError');
11254:   CL.AddClassN(CL.FindClass('TObject'),'EWaveAudioSysError');
11255:   CL.AddClassN(CL.FindClass('TObject'),'EWaveAudioInvalidOperation');
11256:   TWaveAudioEvent, 'Procedure ( Sender : TObject)';
11257:   TWaveAudioGetFormatEvent, 'Procedure ( Sender : TObject; var pW
11258:   +aveFormat : PWaveFormatEx; var FreeIt : Boolean)';
11259:   TWaveAudioGetDataEvent, 'Function ( Sender : TObject; const Buf
11260:   //+fer : Pointer; BufferSize : DWORD; var NumLoops : DWORD) : DWORD';
11261:   TWaveAudioGetDataPtrEvent, 'Function ( Sender : TObject; var Bu
11262:   //+ffer : Pointer; var NumLoops : DWORD; var FreeIt : Boolean) : DWORD';
11263:   TWaveAudioDataReadyEvent, 'Procedure ( Sender : TObject; const
11264:   //+Buffer : Pointer; BufferSize : DWORD; var FreeIt : Boolean)';
11265:   TWaveAudioLevelEvent, 'Procedure ( Sender : TObject; Level : Integer)';
11266:   TWaveAudioFilterEvent, 'Procedure ( Sender : TObject; const Buffer : Pointer; BufferSize : DWORD)';
11267:   GetWaveAudioInfo( mmIO : HMMIO; var pWaveFormat : PWaveFormatEx; var DataSize, DataOffset : DWORD) :
Boolean);
11268:   CreateWaveAudio( mmIO : HMMIO; const pWaveFormat : PWaveFormatEx; var ckRIFF, ckData : TMMCKInfo) :
Boolean);
11269:   CloseWaveAudio( mmIO : HMMIO; var ckRIFF, ckData : TMMCKInfo);
11270:   GetStreamWaveAudioInfo(Stream: TStream; var pWaveFormat: PWaveFormatEx; var DataSize,
DataOffset:DWORD):Boolean);
11271:   CreateStreamWaveAudio(Stream:TStream;const pWaveFormat:PWaveFormatEx; var ckRIFF,ckData:TMMCKInfo) :
HMMIO);
11272:   OpenStreamWaveAudio( Stream : TStream) : HMMIO);
11273:   CalcWaveBufferSize( const pWaveFormat : PWaveFormatEx; Duration : DWORD) : DWORD);
11274:   GetAudioFormat( FormatTag : Word) : String);
11275:   GetWaveAudioFormat( const pWaveFormat : PWaveFormatEx) : String);
11276:   GetWaveAudioLength( const pWaveFormat : PWaveFormatEx; DataSize : DWORD) : DWORD);
11277:   GetWaveAudioBitRate( const pWaveFormat : PWaveFormatEx) : DWORD);
11278:   GetWaveAudioPeakLevel( const Data : Pointer; DataSize : DWORD; const pWaveFormat : PWaveFormatEx) :
Integer);
11279:   InvertWaveAudio( const Data : Pointer; DataSize : DWORD; const pWaveFormat : PWaveFormatEx) : Boolean);
11280:   SilenceWaveAudio( const Data : Pointer; DataSize : DWORD; const pWaveFormat : PWaveFormatEx) : Boolean);
11281:   ChangeWaveAudioVolume(const Data:Pointer;DataSize:DWORD; const
pWaveFormat:PWaveFormatEx;Percent:Integer):Bool;
11282:   MixWaveAudio( const RawWaves : PRawWave; Count : Integer; const pWaveFormat : PWaveFormatEx; Buffer :
Pointer; BufferSize : DWORD) : Boolean);
11283:   ConvertWaveFormat( const srcFormat : PWaveFormatEx; srcData : Pointer; srcDataSize : DWORD; const
dstFormat : PWaveFormatEx; var dstData : Pointer; var dstDataSize : DWORD) : Boolean);
11284:   SetPCMAudioFormat( const pWaveFormat : PWaveFormatEx; Channels : TPCMChannel; SamplesPerSec :
TPCMSamplesPerSec; BitsPerSample : TPCMBitsPerSample);
11285:   SetPCMAudioFormatS( const pWaveFormat : PWaveFormatEx; PCMFormat : TPCMFormat);
11286:   GetPCMAudioFormat( const pWaveFormat : PWaveFormatEx) : TPCMFormat);

```



```

11287:   GetWaveDataPositionOffset( const pWaveFormat : PWaveFormatEx; Position : DWORD) : DWORD));
11288:   MS2Str( Milliseconds : DWORD; Fmt : TMS2StrFormat) : String));
11289:   WaitForSyncObject( SyncObject : THandle; Timeout : DWORD) : DWORD));
11290:   mmIoStreamProc( lpmmIOInfo : PMMIOInfo; uMsg, lParam1, lParam2 : DWORD) : LRESULT));
11291: end;
11292:
11293: procedure SIRegister_RegStr(CL: TPSPascalCompiler);
11294: begin
11295:   REGSTR_KEY_CLASS, 'Class');
11296:   CL.AddConstantN('REGSTR_KEY_CONFIG', 'String').SetString( 'Config');
11297:   REGSTR_KEY_ENUM, 'Enum');
11298:   REGSTR_KEY_ROOTENUM 'Root');
11299:   REGSTR_KEY_BIOSENUM, 'BIOS');
11300:   REGSTR_KEY_PCMCIAENUM, 'String').SetString( 'PCMCIA');
11301:   REGSTR_KEY_PCIEENUM, 'PCI');
11302:   REGSTR_KEY_ISAENUM, 'ISAPnP');
11303:   REGSTR_KEY_EISAENUM, 'EISA');
11304:   REGSTR_KEY_ISAENUM_NEC_98, 'C98PnP');
11305:   REGSTR_KEY_EISAENUM_NEC_98, 'NESA');
11306:   REGSTR_KEY_LOGCONFIG, 'LogConfig');
11307:   REGSTR_KEY_SYSTEMBOARD, '*PNP0C01');
11308:   REGSTR_KEY_APM, '*PNP0C05');
11309:   REGSTR_KEY_INIUPDATE, 'IniUpdate');
11310:   REG_KEY_INSTDEV, 'Installed');
11311:   REGSTR_KEY_DOSOPTCDROM, 'CD-ROM');
11312:   REGSTR_KEY_DOSOPTMOUSE, 'MOUSE');
11313:   REGSTR_DEFAULT_INSTANCE, '0000');
11314:   REGSTR_PATH_SETUP, 'Software\Microsoft\Windows\CurrentVersion');
11315:   REGSTR_PATH_PIFCONVERT, 'Software\Microsoft\Windows\CurrentVersion\PIFConvert');
11316:   REGSTR_PATH_MSDOPTIONS, 'Software\Microsoft\Windows\CurrentVersion\MS-DOSOptions');
11317:   REGSTR_PATH_MSDOSEMU, 'Software\Microsoft\Windows\CurrentVersion\MS-DOS Emulation');
11318:   REGSTR_PATH_NOSUGMSDOS, 'Software\Microsoft\Windows\CurrentVersion\NoMSDOSWarn');
11319:   REGSTR_PATH_NEWDOSBOX, 'Software\Microsoft\Windows\CurrentVersion\MS-DOSSpecialConfig');
11320:   REGSTR_PATH_RUNONCE, 'Software\Microsoft\Windows\CurrentVersion\RunOnce');
11321:   REGSTR_PATH_RUN, 'Software\Microsoft\Windows\CurrentVersion\Run');
11322:   REGSTR_PATH_RUNSERVICESONCE, 'Software\Microsoft\Windows\CurrentVersion\RunServicesOnce');
11323:   REGSTR_PATH_RUNSERVICES, 'Software\Microsoft\Windows\CurrentVersion\RunServices');
11324:   REGSTR_PATH_EXPLORER, 'Software\Microsoft\Windows\CurrentVersion\Explorer');
11325:   REGSTR_PATH_DETECT, 'Software\Microsoft\Windows\CurrentVersion\Detect');
11326:   REGSTR_PATH_APPPATHS, 'Software\Microsoft\Windows\CurrentVersion\App Paths');
11327:   REGSTR_PATH_UNINSTALL, 'Software\Microsoft\Windows\CurrentVersion\Uninstall');
11328:   REGSTR_PATH_REALMODENET, 'Software\Microsoft\Windows\CurrentVersion\Network\Real Mode Net');
11329:   REGSTR_PATH_NETEQUIV, 'Software\Microsoft\Windows\CurrentVersion\Network\Equivalent');
11330:   REGSTR_PATH_CVNETWORK, 'Software\Microsoft\Windows\CurrentVersion\Network');
11331:   REGSTR_PATH_IDCONFIGDB, 'System\CurrentControlSet\Control\IDConfigDB');
11332:   REGSTR_PATH_CLASS, 'System\CurrentControlSet\Services\Class');
11333:   REGSTR_PATH_DISPLAYSETTINGS, 'Display\Settings');
11334:   REGSTR_PATH_FONTS, 'Display\Fonts');
11335:   REGSTR_PATH_ENUM, 'Enum');
11336:   REGSTR_PATH_ROOT, 'Enum\Root');
11337:   REGSTR_PATH_SERVICES, 'System\CurrentControlSet\Services');
11338:   REGSTR_PATH_VXD, 'System\CurrentControlSet\Services\VxD');
11339:   REGSTR_PATH_IOS, 'System\CurrentControlSet\Services\VxD\IOS');
11340:   REGSTR_PATH_VMM, 'System\CurrentControlSet\Services\VxD\VMM');
11341:   REGSTR_PATH_VPOWERD, 'System\CurrentControlSet\Services\VxD\VPOWERD');
11342:   REGSTR_PATH_VNETSUP, 'System\CurrentControlSet\Services\VxD\VNETSUP');
11343:   REGSTR_PATH_NWREDIR, 'System\CurrentControlSet\Services\VxD\NWREDIR');
11344:   REGSTR_PATH_NCPSEVER, 'System\CurrentControlSet\Services\NcpServer\Parameters');
11345:   REGSTR_PATH_IOARB, 'System\CurrentControlSet\Services\Arbitrators\IOArb');
11346:   REGSTR_PATH_ADDRARB, 'System\CurrentControlSet\Services\Arbitrators\AddrArb');
11347:   REGSTR_PATH_DMAARB, 'System\CurrentControlSet\Services\Arbitrators\DMAArb');
11348:   REGSTR_PATH_IRQARB, 'System\CurrentControlSet\Services\Arbitrators\IRQArb');
11349:   REGSTR_PATH_CODEPAGE, 'System\CurrentControlSet\Control\Nls\Codepage');
11350:   REGSTR_PATH_FILESYSTEM, 'System\CurrentControlSet\Control\FileSystem');
11351:   REGSTR_PATH_FILESYSTEM_NOVOLTRACK, 'System\CurrentControlSet\Control\FileSystem\NoVolTrack');
11352:   REGSTR_PATH_CDFS, 'System\CurrentControlSet\Control\FileSystem\CDFS');
11353:   REGSTR_PATH_WINBOOT, 'System\CurrentControlSet\Control\WinBoot');
11354:   REGSTR_PATH_INSTALLEDFILES, 'System\CurrentControlSet\Control\InstalledFiles');
11355:   REGSTR_PATH_VMM32FILES, 'System\CurrentControlSet\Control\VMM32Files');
11356:   REGSTR_MAX_VALUE_LENGTH, 'LongInt').SetInt( 256);
11357:   REGSTR_VAL_BITSPERPIXEL, 'BitsPerPixel');
11358:   REGSTR_VAL_RESOLUTION, 'Resolution');
11359:   REGSTR_VAL_DPILOGICALX, 'DPILogicalX');
11360:   REGSTR_VAL_DPILOGICALY, 'DPILogicalY');
11361:   REGSTR_VAL_DPIPHYSICALX, 'DPIPhysicalX');
11362:   REGSTR_VAL_DPIPHYSICALY, 'DPIPhysicalY');
11363:   REGSTR_VAL_REFRESHRATE, 'RefreshRate');
11364:   REGSTR_VAL_DISPLAYFLAGS, 'DisplayFlags');
11365:   REGSTR_PATH_CONTROLPANEL, 'Control Panel');
11366:   REGSTR_PATH_CONTROLSFOLDER, 'Software\Microsoft\Windows\CurrentVersion\Controls Folder');
11367:   REGSTR_VAL_DOSCP, 'OEMCP');
11368:   REGSTR_VAL_WINCP, 'ACP');
11369:   REGSTR_PATH_DYNA_ENUM, 'Config Manager\Enum');
11370:   REGSTR_VAL_HARDWARE_KEY, 'HardWareKey');
11371:   REGSTR_VAL_ALLOCATION, 'Allocation');
11372:   REGSTR_VAL_PROBLEM, 'Problem');
11373:   REGSTR_VAL_STATUS, 'Status');
11374:   REGSTR_VAL_DONTUSEMEM, 'DontAllocLastMem');
11375:   REGSTR_VAL_SYSTEMROOT, 'SystemRoot');

```

```
11376: REGSTR_VAL_BOOTCOUNT, 'BootCount');
11377: REGSTR_VAL_REALNETSTART, 'RealNetStart');
11378: REGSTR_VAL_MEDIA, 'MediaPath');
11379: REGSTR_VAL_CONFIG, 'ConfigPath');
11380: REGSTR_VAL_DEVICEPATH, 'DevicePath');
11381: REGSTR_VAL_SRCPATH, 'SourcePath');
11382: REGSTR_VAL_OLDWINDIR, 'OldWinDir');
11383: REGSTR_VAL_SETUPFLAGS, 'SetupFlags');
11384: REGSTR_VAL_REGOWNER, 'RegisteredOwner');
11385: REGSTR_VAL_REGORGANIZATION, 'RegisteredOrganization');
11386: REGSTR_VAL_LICENSINGINFO, 'LicensingInfo');
11387: REGSTR_VAL_OLDMSDOSVER, 'OldMSDOSVer');
11388: REGSTR_VAL_FIRSTINSTALLDATETIME, 'FirstInstallDateTime');
11389: REGSTR_VAL_INSTALLTYPE, 'InstallType');
11390: REGSTR_VAL_WRAPPER, 'Wrapper');
11391: IT_COMPACT, 'LongWord').SetUInt( $0000);
11392: IT_TYPICAL, 'LongWord').SetUInt( $0001);
11393: IT_PORTABLE, 'LongWord').SetUInt( $0002);
11394: IT_CUSTOM, 'LongWord').SetUInt( $0003);
11395: REGSTR_KEY_SETUP, '\\Setup');
11396: REGSTR_VAL_BOOTDIR, 'BootDir');
11397: REGSTR_VAL_WINBOOTDIR, 'WinbootDir');
11398: REGSTR_VAL_WINDIR, 'WinDir');
11399: REGSTR_VAL_APPINSPATH, 'AppInstallPath');
11400: REGSTR_KEY_EBDFILESLOCAL, 'EBDFFilesLocale');
11401: REGSTR_KEY_EBDFILESKEYBOARD, 'EBDFFilesKeyboard');
11402: REGSTR_KEY_EBDAUTOEXECBATLOCAL, 'EBDAutoexecBatLocale');
11403: REGSTR_KEY_EBDAUTOEXECBATKEYBOARD, 'EBDAutoexecBatKeyboard');
11404: REGSTR_KEY_EBDCONFIGSYSLOCAL, 'EBDConfigSysLocale');
11405: REGSTR_KEY_EBDCONFIGSYSKEYBOARD, 'EBDConfigSysKeyboard');
11406: REGSTR_VAL_MSOSMODE, 'MSDOSMode');
11407: REGSTR_VAL_MSOSMODEDISCARD, 'Discard');
11408: REGSTR_VAL_DOSOPTGLOBALFLAGS, 'GlobalFlags');
11409: DOSOPTGF_DEFCLEAN, 'LongWord').SetUInt( $00000001);
11410: REGSTR_VAL_DOSOPTFLAGS, 'Flags');
11411: REGSTR_VAL_OPTORDER, 'Order');
11412: REGSTR_VAL_CONFIGSYS, 'Config.Sys');
11413: REGSTR_VAL_AUTOEXEC, 'Autoexec.Bat');
11414: REGSTR_VAL_STDDOSOPTION, 'StdOption');
11415: REGSTR_VAL_DOSOPTTIP, 'TipText');
11416: DOSOPTF_DEFAULT, 'LongWord').SetUInt( $00000001);
11417: DOSOPTF_SUPPORTED, 'LongWord').SetUInt( $00000002);
11418: DOSOPTF_ALWAYSUSE, 'LongWord').SetUInt( $00000004);
11419: DOSOPTF_USESPMODE, 'LongWord').SetUInt( $00000008);
11420: DOSOPTF_PROVIDESUMB, 'LongWord').SetUInt( $00000010);
11421: DOSOPTF_NEEDSETUP, 'LongWord').SetUInt( $00000020);
11422: DOSOPTF_INDOSSTART, 'LongWord').SetUInt( $00000040);
11423: DOSOPTF_MULTIPLE, 'LongWord').SetUInt( $00000080);
11424: SUF_FIRSTTIME, 'LongWord').SetUInt( $00000001);
11425: SUF_EXPRESS, 'LongWord').SetUInt( $00000002);
11426: SUF_BATCHINF, 'LongWord').SetUInt( $00000004);
11427: SUF_CLEAN, 'LongWord').SetUInt( $00000008);
11428: SUF_INSETUP, 'LongWord').SetUInt( $00000010);
11429: SUF_NETSETUP, 'LongWord').SetUInt( $00000020);
11430: SUF_NETHDBOOT, 'LongWord').SetUInt( $00000040);
11431: SUF_NETRPLBOOT, 'LongWord').SetUInt( $00000080);
11432: SUF_SBSCOPYOK, 'LongWord').SetUInt( $00000100);
11433: REGSTR_VAL_DOSPAGER, 'DOSPager');
11434: REGSTR_VAL_VXDGROUPS, 'VXDGroups');
11435: REGSTR_VAL_VPOWERD_FLAGS, 'Flags');
11436: VPDF_DISABLEPWRMGMT, 'LongWord').SetUInt( $00000001);
11437: VPDF_FORCEAPM10MODE, 'LongWord').SetUInt( $00000002);
11438: VPDF_SKIPINTELSLCHECK, 'LongWord').SetUInt( $00000004);
11439: VPDF_DISABLEPWRSTATUSPOLL, 'LongWord').SetUInt( $00000008);
11440: REGSTR_VAL_WORKGROUP, 'Workgroup');
11441: REGSTR_VAL_DIRECTHOST, 'DirectHost');
11442: REGSTR_VAL_FILESHARING, 'FileSharing');
11443: REGSTR_VAL_PRINTSHARING, 'PrintSharing');
11444: REGSTR_VAL_FIRSTNETDRIVE, 'FirstNetworkDrive');
11445: REGSTR_VAL_MAXCONNECTIONS, 'MaxConnections');
11446: REGSTR_VAL_APISUPPORT, 'APISupport');
11447: REGSTR_VAL_MAXRETRY, 'MaxRetry');
11448: REGSTR_VAL_MINRETRY, 'MinRetry');
11449: REGSTR_VAL_SUPPORTLFN, 'SupportLFN');
11450: REGSTR_VAL_SUPPORTBURST, 'SupportBurst');
11451: REGSTR_VAL_SUPPORTTUNNELING, 'SupportTunnelling');
11452: REGSTR_VAL_FULLTRACE, 'FullTrace');
11453: REGSTR_VAL_READCACHING, 'ReadCaching');
11454: REGSTR_VAL_SHOWDOTS, 'ShowDots');
11455: REGSTR_VAL_GAPTIME, 'GapTime');
11456: REGSTR_VAL_SEARCHMODE, 'SearchMode');
11457: REGSTR_VAL_SHELLVERSION, 'ShellVersion');
11458: REGSTR_VAL_MAXLIP, 'MaxLIP');
11459: REGSTR_VAL_PRESERVECASE, 'PreserveCase');
11460: REGSTR_VAL_OPTIMIZEFSN, 'OptimizeFSN');
11461: REGSTR_VAL_NCP_BROWSEMASTER, 'BrowseMaster');
11462: REGSTR_VAL_NCP_USEPEERBROWSING, 'Use_PeerBrowsing');
11463: REGSTR_VAL_NCP_USESAP, 'Use_Sap');
11464: REGSTR_VAL_WIN31FILESYSYSTEM, 'Win31FileSystem');
```

```
11465: REGSTR_VAL_PRESERVELONGNAMES', 'PreserveLongNames');
11466: REGSTR_VAL_DRIVEWRITEBEHIND', 'DriveWriteBehind');
11467: REGSTR_VAL_ASYNCFILECOMMIT', 'AsyncFileCommit');
11468: REGSTR_VAL_PATHCACHECOUNT', 'PathCache');
11469: REGSTR_VAL_NAMECACHECOUNT', 'NameCache');
11470: REGSTR_VAL_CONTIGFILEALLOC', 'ContigFileAllocSize');
11471: REGSTR_VAL_VOLIDLETIMEOUT', 'VolumeIdleTimeout');
11472: REGSTR_VAL_BUFFIDLETIMEOUT', 'BufferIdleTimeout');
11473: REGSTR_VAL_BUFFERAGETIMEOUT', 'BufferAgeTimeout');
11474: REGSTR_VAL_NAMENUMERICTAIL', 'NameNumericTail');
11475: REGSTR_VAL_READAHEADTHRESHOLD', 'ReadAheadThreshold');
11476: REGSTR_VAL_DOUBLEBUFFER', 'DoubleBuffer');
11477: REGSTR_VAL_SOFTCOMPATMODE', 'SoftCompatMode');
11478: REGSTR_VAL_DRIVESPINDOWN', 'DriveSpinDown');
11479: REGSTR_VAL_FORCEPMIO', 'ForcePMIO');
11480: REGSTR_VAL_FORCERMIO', 'ForceRMIO');
11481: REGSTR_VAL_LASTBOOTPMDRVS', 'LastBootPMDrvs');
11482: REGSTR_VAL_VIRTUALHDIRQ', 'VirtualHDIRQ');
11483: REGSTR_VAL_SRVNAMECACHECOUNT', 'ServerNameCacheMax');
11484: REGSTR_VAL_SRVNAMECACHE', 'ServerNameCache');
11485: REGSTR_VAL_SRVNAMECACHENETPROV', 'ServerNameCacheNumNets');
11486: REGSTR_VAL_AUTOMOUNT', 'AutoMountDrives');
11487: REGSTR_VAL_COMPRESSIONMETHOD', 'CompressionAlgorithm');
11488: REGSTR_VAL_COMPRESSIONTHRESHOLD', 'CompressionThreshold');
11489: REGSTR_VAL_CDCACHESIZE', 'CacheSize');
11490: REGSTR_VAL_CDPREFETCH', 'Prefetch');
11491: REGSTR_VAL_CDPREFETCHTAIL', 'PrefetchTail');
11492: REGSTR_VAL_CDRAWCACHE', 'RawCache');
11493: REGSTR_VAL_CDEXTERRORS', 'ExtendedErrors');
11494: REGSTR_VAL_CDSVDSSENSE', 'SVDSense');
11495: REGSTR_VAL_CDSHOWVERSIONS', 'ShowVersions');
11496: REGSTR_VAL_CDCOMPATNAMES', 'MSCDEXCompatNames');
11497: REGSTR_VAL_CDNOREADAHEAD', 'NoReadAhead');
11498: REGSTR_VAL SCSI', 'SCSI\');
11499: REGSTR_VAL_ESDI', 'ESDI\');
11500: REGSTR_VAL_FLOP', 'FLOP\');
11501: REGSTR_VAL_DISK', 'GenDisk');
11502: REGSTR_VAL_CDROM', 'GenCD');
11503: REGSTR_VAL_TAPE', 'TAPE');
11504: REGSTR_VAL_SCANNER', 'SCANNER');
11505: REGSTR_VAL_FLOPPY', 'FLOPPY');
11506: REGSTR_VAL_SCSITID', 'SCSITargetID');
11507: REGSTR_VAL_SCSILUN', 'SCSILUN');
11508: REGSTR_VAL_REVLEVEL', 'RevisionLevel');
11509: REGSTR_VAL_PRODUCTID', 'ProductId');
11510: REGSTR_VAL_PRODUCTTYPE', 'ProductType');
11511: REGSTR_VAL_DEVTYPE', 'DeviceType');
11512: REGSTR_VAL_REMOVABLE', 'Removable');
11513: REGSTR_VAL_CURDRVLET', 'CurrentDriveLetterAssignment');
11514: REGSTR_VAL_USRDRVLET', 'UserDriveLetterAssignment');
11515: REGSTR_VAL_SYNCDATAXFER', 'SyncDataXfer');
11516: REGSTR_VAL_AUTOINSNOTE', 'AutoInsertNotification');
11517: REGSTR_VAL_DISCONNECT', 'Disconnect');
11518: REGSTR_VAL_INT13', 'Int13');
11519: REGSTR_VAL_PMODE_INT13', 'PModeInt13');
11520: REGSTR_VAL_USERSETTINGS', 'AdapterSettings');
11521: REGSTR_VAL_NOIDE', 'NoIDE');
11522: REGSTR_VAL_DISKCLASSNAME', 'DiskDrive');
11523: REGSTR_VAL_CDROMCLASSNAME', 'CDROM');
11524: REGSTR_VAL_FORCELOAD', 'ForceLoadPD');
11525: REGSTR_VAL_FORCEFIFO', 'ForceFIFO');
11526: REGSTR_VAL_FORCECL', 'ForceChangeLine');
11527: REGSTR_VAL_NOUSECLASS', 'NoUseClass');
11528: REGSTR_VAL_NOINSTALLCLASS', 'NoInstallClass');
11529: REGSTR_VAL_NODISPLAYCLASS', 'NoDisplayClass');
11530: REGSTR_VAL_SILENTINSTALL', 'SilentInstall');
11531: REGSTR_KEY_PCMCIA_CLASS', 'PCMCIA');
11532: REGSTR_KEY SCSI_CLASS', 'SCSIAdapter');
11533: REGSTR_KEY_PORTS_CLASS', 'ports');
11534: REGSTR_KEY_MEDIA_CLASS', 'MEDIA');
11535: REGSTR_KEY_DISPLAY_CLASS', 'Display');
11536: REGSTR_KEY_KEYBOARD_CLASS', 'Keyboard');
11537: REGSTR_KEY_MOUSE_CLASS', 'Mouse');
11538: REGSTR_KEY_MONITOR_CLASS', 'Monitor');
11539: REGSTR_VAL_PCMCIA_OPT', 'Options');
11540: PCMCIA_OPT_HAVE_SOCKET', 'LongWord').SetUInt( $00000001);
11541: PCMCIA_OPT_AUTOMEM', 'LongWord').SetUInt( $00000004);
11542: PCMCIA_OPT_NO_SOUND', 'LongWord').SetUInt( $00000008);
11543: PCMCIA_OPT_NO_AUDIO', 'LongWord').SetUInt( $00000010);
11544: PCMCIA_OPT_NO_APMREMOVE', 'LongWord').SetUInt( $00000020);
11545: REGSTR_VAL_PCMCIA_MEM', 'Memory');
11546: PCMCIA_DEF_MEMBEGIN', 'LongWord').SetUInt( $000C0000);
11547: PCMCIA_DEF_MEMEND', 'LongWord').SetUInt( $00FFFFFF);
11548: PCMCIA_DEF_MEMLEN', 'LongWord').SetUInt( $00001000);
11549: REGSTR_VAL_PCMCIA_ALLOC', 'AllocMemWin');
11550: REGSTR_VAL_PCMCIA_ATAD', 'ATADelay');
11551: REGSTR_VAL_PCMCIA_SIZ', 'MinRegionSize');
11552: PCMCIA_DEF_MIN_REGION', 'LongWord').SetUInt( $00010000);
11553: REGSTR_VAL_P1284MDL', 'Model');
```

```

11554: REGSTR_VAL_P1284MFG', 'Manufacturer');
11555: REGSTR_VAL_ISAPNP', 'ISAPNP');
11556: REGSTR_VAL_ISAPNP_RDP_OVERRIDE', 'RDPOverRide');
11557: REGSTR_VAL_PCI', 'PCI');
11558: REGSTR_PCI_OPTIONS', 'Options');
11559: REGSTR_PCI_DUAL_IDE', 'PCIDualIDE');
11560: PCI_OPTIONS_USE_BIOS', 'LongWord').SetUInt( $00000001);
11561: PCI_OPTIONS_USE_IRQ_STEERING', 'LongWord').SetUInt( $00000002);
11562: PCI_FLAG_NO_VIDEO_IRQ', 'LongWord').SetUInt( $00000001);
11563: PCI_FLAG_PCMCIA_WANT_IRQ', 'LongWord').SetUInt( $00000002);
11564: PCI_FLAG_DUAL_IDE', 'LongWord').SetUInt( $00000004);
11565: PCI_FLAG_NO_ENUM_AT_ALL', 'LongWord').SetUInt( $00000008);
11566: PCI_FLAG_ENUM_NO_RESOURCE', 'LongWord').SetUInt( $00000010);
11567: PCI_FLAG_NEED_DWORD_ACCESS', 'LongWord').SetUInt( $00000020);
11568: PCI_FLAG_SINGLE_FUNCTION', 'LongWord').SetUInt( $00000040);
11569: PCI_FLAG_ALWAYS_ENABLED', 'LongWord').SetUInt( $00000080);
11570: PCI_FLAG_IS_IDE', 'LongWord').SetUInt( $00000100);
11571: PCI_FLAG_IS_VIDEO', 'LongWord').SetUInt( $00000200);
11572: PCI_FLAG_FAIL_START', 'LongWord').SetUInt( $00000400);
11573: REGSTR_KEY_CRASHES', 'Crashes');
11574: REGSTR_KEY_DANGERS', 'Dangers');
11575: REGSTR_KEY_DETMODVARS', 'DetModVars');
11576: REGSTR_KEY_NDISINFO', 'NDISInfo');
11577: REGSTR_VAL_PROTINIPATH', 'ProtIniPath');
11578: REGSTR_VAL_RESOURCES', 'Resources');
11579: REGSTR_VAL_CRASHFUNCS', 'CrashFuncs');
11580: REGSTR_VAL_CLASS', 'Class');
11581: REGSTR_VAL_DEVDESC', 'DeviceDesc');
11582: REGSTR_VAL_BOOTCONFIG', 'BootConfig');
11583: REGSTR_VAL_DETFUNC', 'DetFunc');
11584: REGSTR_VAL_DETFLAGS', 'DetFlags');
11585: REGSTR_VAL_COMPATIBLEIDS', 'CompatibleIDs');
11586: REGSTR_VAL_DETCONFIG', 'DetConfig');
11587: REGSTR_VAL_VERIFYKEY', 'VerifyKey');
11588: REGSTR_VAL_COMINFO', 'ComInfo');
11589: REGSTR_VAL_INFNAME', 'InfName');
11590: REGSTR_VAL_CARDSPECIFIC', 'CardSpecific');
11591: REGSTR_VAL_NETOSTYPE', 'NetOSType');
11592: REGSTR_DATA_NETOS_NDIS', 'NDIS');
11593: REGSTR_DATA_NETOS_ODI', 'ODI');
11594: REGSTR_DATA_NETOS_IPX', 'IPX');
11595: REGSTR_VAL_MFG', 'Mfg');
11596: REGSTR_VAL_SCAN_ONLY_FIRST', 'ScanOnlyFirstDrive');
11597: REGSTR_VAL_SHARE_IRQ', 'ForceIRQSharing');
11598: REGSTR_VAL_NONSTANDARD_ATAPI', 'NonStandardATAPI');
11599: REGSTR_VAL_IDE_FORCE_SERIALIZE', 'ForceSerialization');
11600: REGSTR_VAL_MAX_HCIDE_LEN', 'LongInt').SetInt( 1024);
11601: REGSTR_VAL_HWREV', 'HWRevision');
11602: REGSTR_VAL_ENABLEINTS', 'EnableInts');
11603: REGDF_NOTDETIO', 'LongWord').SetUInt( $00000001);
11604: REGDF_NOTDETMEM', 'LongWord').SetUInt( $00000002);
11605: REGDF_NOTDETIRO', 'LongWord').SetUInt( $00000004);
11606: REGDF_NOTDETDMA', 'LongWord').SetUInt( $00000008);
11607: REGDF_NEEDFULLCONFIG', 'LongWord').SetUInt( $00000010);
11608: REGDF_GENFORCEDCONFIG', 'LongWord').SetUInt( $00000020);
11609: REGDF_NODETCONFIG', 'LongWord').SetUInt( $00008000);
11610: REGDF_CONFLICTIO', 'LongWord').SetUInt( $00010000);
11611: REGDF_CONFLICTMEM', 'LongWord').SetUInt( $00020000);
11612: REGDF_CONFLICTIRO', 'LongWord').SetUInt( $00040000);
11613: REGDF_CONFLICTDMA', 'LongWord').SetUInt( $00080000);
11614: REGDF_MAPIRQ2TO9', 'LongWord').SetUInt( $00100000);
11615: REGDF_NOTVERIFIED', 'LongWord').SetUInt( $80000000);
11616: REGSTR_VAL_APMBIOSVER', 'APMBiosVer');
11617: REGSTR_VAL_APMFLAGS', 'APMFlags');
11618: REGSTR_VAL_SLISUPPORT', 'SLISupport');
11619: REGSTR_VAL_MACHINETYPE', 'MachineType');
11620: REGSTR_VAL_SETUPMACHINETYPE', 'SetupMachineType');
11621: REGSTR_MACHTYPE_UNKNOWN', 'Unknown');
11622: REGSTR_MACHTYPE_IBMPC', 'IBM PC');
11623: REGSTR_MACHTYPE_IBMPCJR', 'IBM PCjr');
11624: REGSTR_MACHTYPE_IBMPCCONV', 'IBM PC Convertible');
11625: REGSTR_MACHTYPE_IBMPCXT', 'IBM PC/XT');
11626: REGSTR_MACHTYPE_IBMPCXT_286', 'IBM PC/XT 286');
11627: REGSTR_MACHTYPE_IBMPCAT', 'IBM PC/AT');
11628: REGSTR_MACHTYPE_IBMPS2_25', 'IBM PS/2-25');
11629: REGSTR_MACHTYPE_IBMPS2_30_286', 'IBM PS/2-30 286');
11630: REGSTR_MACHTYPE_IBMPS2_30', 'IBM PS/2-30');
11631: REGSTR_MACHTYPE_IBMPS2_50', 'IBM PS/2-50');
11632: REGSTR_MACHTYPE_IBMPS2_50Z', 'IBM PS/2-50Z');
11633: REGSTR_MACHTYPE_IBMPS2_55SX', 'IBM PS/2-55SX');
11634: REGSTR_MACHTYPE_IBMPS2_60', 'IBM PS/2-60');
11635: REGSTR_MACHTYPE_IBMPS2_65SX', 'IBM PS/2-65SX');
11636: REGSTR_MACHTYPE_IBMPS2_70', 'IBM PS/2-70');
11637: REGSTR_MACHTYPE_IBMPS2_P70', 'IBM PS/2-P70');
11638: REGSTR_MACHTYPE_IBMPS2_70_80', 'IBM PS/2-70/80');
11639: REGSTR_MACHTYPE_IBMPS2_80', 'IBM PS/2-80');
11640: REGSTR_MACHTYPE_IBMPS2_90', 'IBM PS/2-90');
11641: REGSTR_MACHTYPE_IBMPS1', 'IBM PS/1');
11642: REGSTR_MACHTYPE_PHOENIX_PCAT', 'Phoenix PC/AT Compatible');

```



```

11643: REGSTR_MACHTYPE_HP_VECTRA', 'HP Vectra');
11644: REGSTR_MACHTYPE_ATT_PC', 'AT&T PC');
11645: REGSTR_MACHTYPE_ZENITH_PC', 'Zenith PC');
11646: REGSTR_VAL_APMENUSUSPEND', 'APMMenuSuspend');
11647: APMENUSUSPEND_DISABLED', 'LongInt').SetInt( 0);
11648: APMENUSUSPEND_ENABLED', 'LongInt').SetInt( 1);
11649: APMENUSUSPEND_UNDOCKED', 'LongInt').SetInt( 2);
11650: APMENUSUSPEND_NOCHANGE', 'LongWord').SetUInt( $80);
11651: REGSTR_VAL_BUSTYPE', 'BusType');
11652: REGSTR_VAL_CPU', 'CPU');
11653: REGSTR_VAL_NDP', 'NDP');
11654: REGSTR_VAL_PNPBIOSVER', 'PnPBIOSVer');
11655: REGSTR_VAL_PNPSTRUCOFFSET', 'PnPStrucOffset');
11656: REGSTR_VAL_PCIBIOSVER', 'PCIBIOSVer');
11657: REGSTR_VAL_HWMCHANISM', 'HWMechanism');
11658: REGSTR_VAL_LASTPCIBUSNUM', 'LastPCIBusNum');
11659: REGSTR_VAL_CONVMEM', 'ConvMem');
11660: REGSTR_VAL_EXTMEM', 'ExtMem');
11661: REGSTR_VAL_COMPUTERNAME', 'ComputerName');
11662: REGSTR_VAL_BIOSNAME', 'BIOSName');
11663: REGSTR_VAL_BIOSVERSION', 'BIOSVersion');
11664: REGSTR_VAL_BIOSDATE', 'BIOSDate');
11665: REGSTR_VAL_MODEL', 'Model');
11666: REGSTR_VAL_SUBMODEL', 'Submodel');
11667: REGSTR_VAL_REVISION', 'Revision');
11668: REGSTR_VAL_FIFODEPTH', 'FIFODepth');
11669: REGSTR_VAL_RDINTTHRESHOLD', 'RDIntThreshold');
11670: REGSTR_VAL_WRINTTHRESHOLD', 'WRIntThreshold');
11671: REGSTR_VAL_PRIORITY', 'Priority');
11672: REGSTR_VAL_DRIVER', 'Driver');
11673: REGSTR_VAL_FUNCDESC', 'FunctionDesc');
11674: REGSTR_VAL_FORCEDCONFIG', 'ForcedConfig');
11675: REGSTR_VAL_CONFIGFLAGS', 'ConfigFlags');
11676: REGSTR_VAL_CS_CONFIGFLAGS', 'CSConfigFlags');
11677: CONFIGFLAG_DISABLED', 'LongWord').SetUInt( $00000001);
11678: CONFIGFLAG_REMOVED', 'LongWord').SetUInt( $00000002);
11679: CONFIGFLAG_MANUAL_INSTALL', 'LongWord').SetUInt( $00000004);
11680: CONFIGFLAG_IGNORE_BOOT_LC', 'LongWord').SetUInt( $00000008);
11681: CONFIGFLAG_NET_BOOT', 'LongWord').SetUInt( $00000010);
11682: CONFIGFLAG_REINSTALL', 'LongWord').SetUInt( $00000020);
11683: CONFIGFLAG_FAILEDINSTALL', 'LongWord').SetUInt( $00000040);
11684: CONFIGFLAG_CANTSTOPACHILD', 'LongWord').SetUInt( $00000080);
11685: CONFIGFLAG_OKREMOVEROM', 'LongWord').SetUInt( $00000100);
11686: CONFIGFLAG_NOREMOVEEXIT', 'LongWord').SetUInt( $00000200);
11687: CS_CONFIGFLAG_BITS', 'LongWord').SetUInt( $00000007);
11688: CS_CONFIGFLAG_DISABLED', 'LongWord').SetUInt( $00000001);
11689: CS_CONFIGFLAG_DO_NOT_CREATE', 'LongWord').SetUInt( $00000002);
11690: CS_CONFIGFLAG_DO_NOT_START', 'LongWord').SetUInt( $00000004);
11691: DMSTATEFLAG_APPLYTOALL', 'LongWord').SetUInt( $00000001);
11692: REGSTR_VAL_ROOT_DEVNODE', 'HTREE\ROOT\0');
11693: REGSTR_VAL_RESERVED_DEVNODE', 'HTREE\RESERVED\0');
11694: REGSTR_PATH_READDATAPORT', REGSTR_KEY_ISAENUM + '\ReadDataPort\0');
11695: REGSTR_PATH_MULTI_FUNCTION', 'MF');
11696: REGSTR_VAL_RESOURCE_MAP', 'ResourceMap');
11697: REGSTR_PATH_CHILD_PREFIX', 'Child');
11698: NUM_RESOURCE_MAP', 'LongInt').SetInt( 256);
11699: REGSTR_VAL_MF_FLAGS', 'MFFlags');
11700: MF_FLAGS_EVEN_IF_NO_RESOURCE', 'LongWord').SetUInt( $00000001);
11701: MF_FLAGS_NO_CREATE_IF_NO_RESOURCE', 'LongWord').SetUInt( $00000002);
11702: MF_FLAGS_FILL_IN_UNKNOWN_RESOURCE', 'LongWord').SetUInt( $00000004);
11703: MF_FLAGS_CREATE_BUT_NO_SHOW_DISABLED', 'LongWord').SetUInt( $00000008);
11704: REGSTR_VAL_EISA_RANGES', 'EISARanges');
11705: REGSTR_VAL_EISA_FUNCTIONS', 'EISAFunctions');
11706: REGSTR_VAL_EISA_FUNCTIONS_MASK', 'EISAFunctionsMask');
11707: REGSTR_VAL_EISA_FLAGS', 'EISAFlags');
11708: REGSTR_VAL_EISA_SIMULATE_INT15', 'EISASimulateInt15');
11709: REGSTR_VAL_EISA_RANGES_NEC_98', 'NESARanges');
11710: REGSTR_VAL_EISA_FUNCTIONS_NEC_98', 'NESAFunctions');
11711: REGSTR_VAL_EISA_FUNCTIONS_MASK_NEC_98', 'NESAFunctionsMask');
11712: REGSTR_VAL_EISA_FLAGS_NEC_98', 'NESAFFlags');
11713: REGSTR_VAL_EISA_SIMULATE_INT15_NEC_98', 'NESASimulateInt15');
11714: EISAFLAG_NO_IO_MERGE', 'LongWord').SetUInt( $00000001);
11715: EISAFLAG_SLOT_IO_FIRST', 'LongWord').SetUInt( $00000002);
11716: EISA_NO_MAX_FUNCTION', 'LongWord').SetUInt( $FF);
11717: NUM_EISA_RANGES', 'LongInt').SetInt( 4);
11718: REGSTR_VAL_DRVDESC', 'DriverDesc');
11719: REGSTR_VAL_DEVLOADER', 'DevLoader');
11720: REGSTR_VAL_STATICVXD', 'StaticVxD');
11721: REGSTR_VAL_PROPERTIES', 'Properties');
11722: REGSTR_VAL_MANUFACTURER', 'Manufacturer');
11723: REGSTR_VAL_EXISTS', 'Exists');
11724: REGSTR_VAL_CMENUMFLAGS', 'CMenuFlags');
11725: REGSTR_VAL_CMDRIVFLAGS', 'CMDrivFlags');
11726: REGSTR_VAL_ENUMERATOR', 'Enumerator');
11727: REGSTR_VAL_DEVICEDRIVER', 'DeviceDriver');
11728: REGSTR_VAL_PORTNAME', 'PortName');
11729: REGSTR_VAL_INFPPATH', 'InfPath');
11730: REGSTR_VAL_INFSECTION', 'InfSection');
11731: REGSTR_VAL_POLLING', 'Polling');

```

```
11732: REGSTR_VAL_DONTLOADIFCONFLICT', 'DontLoadIfConflict');
11733: REGSTR_VAL_PORTSUBCLASS', 'PortSubClass');
11734: REGSTR_VAL_NETCLEAN', 'NetClean');
11735: REGSTR_VAL_IDE_NO_SERIALIZE', 'IDENoSerialize');
11736: REGSTR_VAL_NOCMOSORFDPT', 'NoCMOSorFDPT');
11737: REGSTR_VAL_COMVERIFYBASE', 'COMVerifyBase');
11738: REGSTR_KEY_OVERRIDE', 'Override');
11739: REGSTR_VAL_CONFIGMG', 'CONFIGMG');
11740: REGSTR_VAL_SYSDM', 'SysDM');
11741: REGSTR_VAL_SYSDMFUNC', 'SysDMFunc');
11742: REGSTR_VAL_PRIVATE', 'Private');
11743: REGSTR_VAL_PRIVATEFUNC', 'PrivateFunc');
11744: REGSTR_VAL_DETECT', 'Detect');
11745: REGSTR_VAL_DETECTFUNC', 'DetectFunc');
11746: REGSTR_VAL_ASKFORCONFIG', 'AskForConfig');
11747: REGSTR_VAL_ASKFORCONFIGFUNC', 'AskForConfigFunc');
11748: REGSTR_VAL_WAITFORUNDOCK', 'WaitForUndock');
11749: REGSTR_VAL_WAITFORUNDOCKFUNC', 'WaitForUndockFunc');
11750: REGSTR_VAL_REMOVEFROMOKAY', 'RemoveRomOkay');
11751: REGSTR_VAL_REMOVEFROMOKAYFUNC', 'RemoveRomOkayFunc');
11752: REGSTR_VAL_CURCONFIG', 'CurrentConfig');
11753: REGSTR_VAL_FRIENDLYNAME', 'FriendlyName');
11754: REGSTR_VAL_CURRENTCONFIG', 'CurrentConfig');
11755: REGSTR_VAL_MAP', 'Map');
11756: REGSTR_VAL_ID', 'CurrentID');
11757: REGSTR_VAL_DOCKED', 'CurrentDockedState');
11758: REGSTR_VAL_CHECKSUM', 'CurrentChecksum');
11759: REGSTR_VAL_HWDETECT', 'HardwareDetect');
11760: REGSTR_VAL_INHIBITRESULTS', 'InhibitResults');
11761: REGSTR_VAL_PROFILEFLAGS', 'ProfileFlags');
11762: REGSTR_KEY_PCMCIA', 'PCMCIA');
11763: REGSTR_KEY_PCUNKNOWN', 'UNKNOWN_MANUFACTURER');
11764: REGSTR_VAL_PCSSDRIVER', 'Driver');
11765: REGSTR_KEY_PCMTD', 'MTD-');
11766: REGSTR_VAL_PCMTDRIVER', 'MTD');
11767: REGSTR_VAL_HARDWAREID', 'HardwareID');
11768: REGSTR_VAL_INSTALLER', 'Installer');
11769: REGSTR_VAL_INSICON', 'Icon');
11770: REGSTR_VAL_ENUMPROPPAGES', 'EnumPropPages');
11771: REGSTR_VAL_BASICPROPERTIES', 'BasicProperties');
11772: REGSTR_VAL_PRIVATEPROBLEM', 'PrivateProblem');
11773: REGSTR_KEY_CURRENT', 'Current');
11774: REGSTR_KEY_DEFAULT', 'Default');
11775: REGSTR_KEY_MODES', 'Modes');
11776: REGSTR_VAL_MODE', 'Mode');
11777: REGSTR_VAL_BPP', 'BPP');
11778: REGSTR_VAL_HRES', 'HRes');
11779: REGSTR_VAL_VRES', 'VRes');
11780: REGSTR_VAL_FONTSIZE', 'FontSize');
11781: REGSTR_VAL_DRV', 'drv');
11782: REGSTR_VAL_GRB', 'grb');
11783: REGSTR_VAL_VDD', 'vdd');
11784: REGSTR_VAL_VER', 'Ver');
11785: REGSTR_VAL_MAXRES', 'MaxResolution');
11786: REGSTR_VAL_DPMS', 'DPMS');
11787: REGSTR_VAL_RESUMERESET', 'ResumeReset');
11788: REGSTR_VAL_DESCRIPTION', 'Description');
11789: REGSTR_KEY_SYSTEM', 'System');
11790: REGSTR_KEY_USER', 'User');
11791: REGSTR_VAL_DPI', 'dpi');
11792: REGSTR_VAL_PCICOPTIONS', 'PCICOptions');
11793: PCIC_DEFAULT_IRQMASK', 'LongWord').SetUInt( $4EB8);
11794: PCIC_DEFAULT_IRQMASK_NEC_98', 'LongWord').SetUInt( $1468);
11795: PCIC_DEFAULT_NUMSOCKETS', 'LongInt').SetInt( 0);
11796: REGSTR_VAL_PCICIRQMAP', 'PCICIRQMap');
11797: REGSTR_PATH_APPEARANCE', 'Control Panel\Appearance');
11798: REGSTR_PATH_LOOKSCHEMES', 'Control Panel\Appearance\Schemes');
11799: REGSTR_VAL_CUSTOMCOLORS', 'CustomColors');
11800: REGSTR_PATH_SCREENSAVE', 'Control Panel\Desktop');
11801: REGSTR_VALUE_USESCRPPASSWORD', 'ScreenSaveUsePassword');
11802: REGSTR_VALUE_SCRPASSWORD', 'ScreenSave_Data');
11803: REGSTR_VALUE_LOWPOWERTIMEOUT', 'ScreenSaveLowPowerTimeout');
11804: REGSTR_VALUE_POWEROFFTIMEOUT', 'ScreenSavePowerOffTimeout');
11805: REGSTR_VALUE_LOWPOWERACTION', 'ScreenSaveLowPowerActive');
11806: REGSTR_VALUE_POWEROFFACTION', 'ScreenSavePowerOffActive');
11807: REGSTR_PATH_WINDOWSAPPLETS', 'Software\Microsoft\Windows\CurrentVersion\Applets');
11808: REGSTR_PATH_SYSTRAY', 'Software\Microsoft\Windows\CurrentVersion\Applets\SysTray');
11809: REGSTR_VAL_SYSTRAYSVC', 'Services');
11810: REGSTR_VAL_SYSTRAYBATFLAGS', 'PowerFlags');
11811: REGSTR_VAL_SYSTRAYPCCARDFLAGS', 'PCMCIAFlags');
11812: REGSTR_PATH_NETWORK_USERSSETTINGS', 'Network');
11813: REGSTR_KEY_NETWORK_PERSISTENT', '\Persistent');
11814: REGSTR_KEY_NETWORK_RECENT', '\Recent');
11815: REGSTR_VAL_REMOTE_PATH', 'RemotePath');
11816: REGSTR_VAL_USER_NAME', 'UserName');
11817: REGSTR_VAL_PROVIDER_NAME', 'ProviderName');
11818: REGSTR_VAL_CONNECTION_TYPE', 'ConnectionType');
11819: REGSTR_VAL_UPGRADE', 'Upgrade');
11820: REGSTR_KEY_LOGON', '\Logon');
```

```

11821: REGSTR_VAL_MUSTBEVALIDATED', 'MustBeValidated');
11822: REGSTR_VAL_RUNLOGINSRIPT', 'ProcessLoginScript');
11823: REGSTR_KEY_NETWORKPROVIDER', '\NetworkProvider');
11824: REGSTR_VAL_AUTHENT_AGENT', 'AuthenticatingAgent');
11825: REGSTR_VAL_PREFREDIR', 'PreferredRedir');
11826: REGSTR_VAL_AUTOSTART', 'AutoStart');
11827: REGSTR_VAL_AUTOLOGON', 'AutoLogon');
11828: REGSTR_VAL_NETCARD', 'Netcard');
11829: REGSTR_VAL_TRANSPORT', 'Transport');
11830: REGSTR_VAL_DYNAMIC', 'Dynamic');
11831: REGSTR_VAL_TRANSITION', 'Transition');
11832: REGSTR_VAL_STATICDRIVE', 'StaticDrive');
11833: REGSTR_VAL_LOADHI', 'LoadHi');
11834: REGSTR_VAL_LOADRMDRIVERS', 'LoadRMDrivers');
11835: REGSTR_VAL_SETUPN', 'SetupN');
11836: REGSTR_VAL_SETUPNPATH', 'SetupNPath');
11837: REGSTR_VAL_WRKGRP_FORCEMAPPING', 'WrkgrpForceMapping');
11838: REGSTR_VAL_WRKGRP_REQUIRED', 'WrkgrpRequired');
11839: REGSTR_PATH_CURRENT_CONTROL_SET', 'System\CurrentControlSet\Control';
11840: REGSTR_VAL_CURRENT_USER', 'Current User');
11841: REGSTR_PATH_PWDPROVIDER', 'System\CurrentControlSet\Control\PwdProvider';
11842: REGSTR_VAL_PWDPROVIDER_PATH', 'ProviderPath';
11843: REGSTR_VAL_PWDPROVIDER_DESC', 'Description';
11844: REGSTR_VAL_PWDPROVIDER_CHANGEPWD', 'ChangePassword';
11845: REGSTR_VAL_PWDPROVIDER_CHANGEPWDHwnd', 'ChangePasswordHwnd';
11846: REGSTR_VAL_PWDPROVIDER_GETPWDSTATUS', 'GetPasswordStatus';
11847: REGSTR_VAL_PWDPROVIDER_ISNP', 'NetworkProvider';
11848: REGSTR_VAL_PWDPROVIDER_CHANGEORDER', 'ChangeOrder';
11849: REGSTR_PATH_POLICIES', 'Software\Microsoft\Windows\CurrentVersion\Policies';
11850: REGSTR_PATH_UPDATE', 'System\CurrentControlSet\Control\Update';
11851: REGSTR_VALUE_ENABLE', 'Enable';
11852: REGSTR_VALUE_VERBOSE', 'Verbose';
11853: REGSTR_VALUE_NETPATH', 'NetworkPath';
11854: REGSTR_VALUE_DEFAULTLOC', 'UseDefaultNetLocation';
11855: REGSTR_KEY_NETWORK', 'Network';
11856: REGSTR_KEY_PRINTERS', 'Printers';
11857: REGSTR_KEY_WINOLDAPP', 'WinOldApp';
11858: REGSTR_VAL_NOFILESHARING', 'NoFileSharing';
11859: REGSTR_VAL_NOPRINTSHARING', 'NoPrintSharing';
11860: REGSTR_VAL_NOFILESHARINGCTRL', 'NoFileSharingControl';
11861: REGSTR_VAL_NOPRINTSHARINGCTRL', 'NoPrintSharingControl';
11862: REGSTR_VAL_HIDESHAREPWDS', 'HideSharePwds';
11863: REGSTR_VAL_DISABLEPWDCACHING', 'DisablePwdCaching';
11864: REGSTR_VAL_ALPHANUMPWDS', 'AlphanumPwds';
11865: REGSTR_VAL_NETSETUP_DISABLE', 'NoNetSetup';
11866: REGSTR_VAL_NETSETUP_NOCONFIGPAGE', 'NoNetSetupConfigPage';
11867: REGSTR_VAL_NETSETUP_NOIDPAGE', 'NoNetSetupIDPage';
11868: REGSTR_VAL_NETSETUP_NOSECURITYPAGE', 'NoNetSetupSecurityPage';
11869: REGSTR_VAL_SYSTEMCPL_NOVIRTMEMPAGE', 'NoVirtMemPage';
11870: REGSTR_VAL_SYSTEMCPL_NODEVMGRPAGE', 'NoDevMgrPage';
11871: REGSTR_VAL_SYSTEMCPL_NOCONFIGPAGE', 'NoConfigPage';
11872: REGSTR_VAL_SYSTEMCPL_NOFILESYSPAGE', 'NoFileSysPage';
11873: REGSTR_VAL_DISPCPL_NODISPCPL', 'NoDispCPL';
11874: REGSTR_VAL_DISPCPL_NOBACKGROUNDPAGE', 'NoDispBackgroundPage';
11875: REGSTR_VAL_DISPCPL_NOSCRSAVPAGE', 'NoDispScrSavPage';
11876: REGSTR_VAL_DISPCPL_NOAPPEARANCEPAGE', 'NoDispAppearancePage';
11877: REGSTR_VAL_DISPCPL_NOSETTINGSPAGE', 'NoDispSettingsPage';
11878: REGSTR_VAL_SECCPL_NOSECCPL', 'NoSecCPL';
11879: REGSTR_VAL_SECCPL_NOPWDPAGE', 'NoPwdPage';
11880: REGSTR_VAL_SECCPL_NOADMINPAGE', 'NoAdminPage';
11881: REGSTR_VAL_SECCPL_NOPROFILEPAGE', 'NoProfilePage';
11882: REGSTR_VAL_PRINTERS_HIDETABS', 'NoPrinterTabs';
11883: REGSTR_VAL_PRINTERS_NODELETE', 'NoDeletePrinter';
11884: REGSTR_VAL_PRINTERS_NOADD', 'NoAddPrinter';
11885: REGSTR_VAL_WINOLDAPP_DISABLED', 'Disabled';
11886: REGSTR_VAL_WINOLDAPP_NOREALMODE', 'NoRealMode';
11887: REGSTR_VAL_NOENTIRENETWORK', 'NoEntireNetwork';
11888: REGSTR_VAL_NOWORKGROUPCONTENTS', 'NoWorkgroupContents';
11889: REGSTR_VAL_MINPWDLEN', 'MinPwdLen';
11890: REGSTR_VAL_PWDEXPIRATION', 'PwdExpiration';
11891: REGSTR_VAL_WIN31PROVIDER', 'Win31Provider';
11892: REGSTR_VAL_DISABLEREGTOOLS', 'DisableRegistryTools';
11893: REGSTR_PATH_WINLOGON', 'Software\Microsoft\Windows\CurrentVersion\Winlogon';
11894: REGSTR_VAL_LEGALNOTICECAPTION', 'LegalNoticeCaption';
11895: REGSTR_VAL_LEGALNOTICETEXT', 'LegalNoticeText';
11896: REGSTR_VAL_RESTRICTRUN', 'RestrictRun';
11897: REGSTR_KEY_POL_USERS', 'Users';
11898: REGSTR_KEY_POL_COMPUTERS', 'Computers';
11899: REGSTR_KEY_POL_USERGROUPS', 'UserGroups';
11900: REGSTR_KEY_POL_DEFAULT', '.default';
11901: REGSTR_KEY_POL_USERGROUPDATA', 'GroupData\UserGroups\Priority';
11902: REGSTR_PATH_TIMEZONE', 'System\CurrentControlSet\Control\TimeZoneInformation';
11903: REGSTR_VAL_TZBIAS', 'Bias';
11904: REGSTR_VAL_TZDLTBIAS', 'DaylightBias';
11905: REGSTR_VAL_TZSTDBIAS', 'StandardBias';
11906: REGSTR_VAL_TZACTBIAS', 'ActiveTimeBias';
11907: REGSTR_VAL_TZDLTFLAG', 'DaylightFlag';
11908: REGSTR_VAL_TZSTDSTART', 'StandardStart';
11909: REGSTR_VAL_TZDLTSTART', 'DaylightStart';

```

```

11910: REGSTR_VAL_TZDLTNAME', 'DaylightName');
11911: REGSTR_VAL_TZSTDNAME', 'StandardName');
11912: REGSTR_VAL_TZNOCHANGESTART', 'NoChangeStart');
11913: REGSTR_VAL_TZNOCHANGEEND', 'NoChangeEnd');
11914: REGSTR_VAL_TZNOAUTOTIME', 'DisableAutoDaylightTimeSet');
11915: REGSTR_PATH_FLOATINGPOINTPROCESSOR', 'HARDWARE\DESCRIPTION\System\FloatingPointProcessor');
11916: REGSTR_PATH_FLOATINGPOINTPROCESSOR0', 'HARDWARE\DESCRIPTION\System\FloatingPointProcessor\0');
11917: REGSTR_PATH_COMPUTRNAME', 'System\CurrentControlSet\Control\ComputerName\ComputerName');
11918: REGSTR_VAL_COMPUTRNAME', 'ComputerName');
11919: REGSTR_PATH_SHUTDOWN', 'System\CurrentControlSet\Control\Shutdown');
11920: REGSTR_VAL_FORCEREBOOT', 'ForceReboot');
11921: REGSTR_VAL_SETUPPROGRAMRAN', 'SetupProgramRan');
11922: REGSTR_VAL_DOES_POLLING', 'PollingSupportNeeded');
11923: REGSTR_PATH_KNOWNDLLS', 'System\CurrentControlSet\Control\SessionManager\KnownDLLs');
11924: REGSTR_PATH_KNOWN16DLLS', 'System\CurrentControlSet\Control\SessionManager\Known16DLLs');
11925: REGSTR_PATH_CHECKVERDLLS', 'System\CurrentControlSet\Control\SessionManager\CheckVerDLLs');
11926: REGSTR_PATH_WARNVERDLLS', 'System\CurrentControlSet\Control\SessionManager\WarnVerDLLs');
11927: REGSTR_PATH_HACKINIFILE', 'System\CurrentControlSet\Control\SessionManager\HackIniFiles');
11928: REGSTR_PATH_CHECKBADAPPS', 'System\CurrentControlSet\Control\SessionManager\CheckBadApps');
11929: REGSTR_PATH_APPPATCH', 'System\CurrentControlSet\Control\SessionManager\AppPatches');
11930: REGSTR_PATH_KNOWNVXDS', 'System\CurrentControlSet\Control\SessionManager\KnownVxDs');
11931: REGSTR_VAL_UNINSTALLER_DISPLAYNAME', 'DisplayName');
11932: REGSTR_VAL_UNINSTALLER_COMMANDLINE', 'UninstallString');
11933: //REGSTR_PATH_DESKTOP', '').SetString( REGSTR_PATH_SCREENSAVE);
11934: REGSTR_PATH_MOUSE', 'Control Panel\Mouse');
11935: REGSTR_PATH_KEYBOARD', 'Control Panel\Keyboard');
11936: REGSTR_PATH_COLORS', 'Control Panel\Colors');
11937: REGSTR_PATH_SOUND', 'Control Panel\Sound');
11938: REGSTR_PATH_METRICS', 'Control Panel\Desktop\WindowMetrics');
11939: REGSTR_PATH_ICONS', 'Control Panel\Icons');
11940: REGSTR_PATH_CURSORS', 'Control Panel\Cursors');
11941: REGSTR_PATH_CHECKDISK', 'Software\Microsoft\Windows\CurrentVersion\Applets\Check Drive');
11942: REGSTR_PATH_CHECKDISKSET', 'Settings');
11943: REGSTR_PATH_CHECKDISKUDRVS', 'NoUnknownDDErrDrvs');
11944: REGSTR_PATH_FAULT', 'Software\Microsoft\Windows\CurrentVersion\Fault');
11945: REGSTR_VAL_FAULT_LOGFILE', 'LogFile');
11946: REGSTR_PATH_AEDEBUG', 'Software\Microsoft\Windows NT\CurrentVersion\AeDebug');
11947: REGSTR_VAL_AEDEBUG_DEBUGGER', 'Debugger');
11948: REGSTR_VAL_AEDEBUG_AUTO', 'Auto');
11949: REGSTR_PATH_GRPCONV', 'Software\Microsoft\Windows\CurrentVersion\GrpConv');
11950: REGSTR_VAL_REGITEMDELETEMESAGE', 'Removal Message');
11951: REGSTR_PATH_LASTCHECK', 'Software\Microsoft\Windows\CurrentVersion\Explorer\LastCheck');
11952: REGSTR_PATH_LASTOPTIMIZE', 'Software\Microsoft\Windows\CurrentVersion\Explorer\LastOptimize');
11953: REGSTR_PATH_LASTBACKUP', 'Software\Microsoft\Windows\CurrentVersion\Explorer\LastBackup');
11954: REGSTR_PATH_CHKLASTCHECK', 'Software\Microsoft\Windows\CurrentVersion\Applets\Check Drive\LastCheck');
11955: REGSTR_PATH_CHKLASTSURFAN', 'Software\Microsoft\Windows\CurrentVersion\Applets\Check
Drive\LastSurfaceAnalysis');
11956: //CL.AddTypes('PDSKTLSystemTime', '^TDSKTLSystemTime // will not work');
11957: CL.AddTypeS('_DSKTLSYSTEMTIME', 'record wYear : Word; wMonth : Word; wDayOfWe
+ 'ek : Word; wDay : Word; wHour : Word; wMinute : Word; wSecond : Word; wMil
+ 'liseconds : Word; wResult : Word; end');
11960: CL.AddTypeS('DSKTLSYSTEMTIME', '_DSKTLSYSTEMTIME');
11961: CL.AddTypeS('TDSKTLSystemTime', '_DSKTLSYSTEMTIME');
11962: DTRESULTOK', 'LongInt').SetInt( 0);
11963: DTRESULTFIX', 'LongInt').SetInt( 1);
11964: DTRESULTTPROB', 'LongInt').SetInt( 2);
11965: DTRESULTPART', 'LongInt').SetInt( 3);
11966: REGSTR_KEY_SHARES', 'Software\Microsoft\Windows\CurrentVersion\Network\LanMan');
11967: REGSTR_VAL_SHARES_FLAGS', 'Flags');
11968: REGSTR_VAL_SHARES_TYPE', 'Type');
11969: REGSTR_VAL_SHARES_PATH', 'Path');
11970: REGSTR_VAL_SHARES_REMARK', 'Remark');
11971: REGSTR_VAL_SHARES_RW_PASS', 'Parm1');
11972: REGSTR_VAL_SHARES_RO_PASS', 'Parm2');
11973: REGSTR_PATH_PRINT', 'System\CurrentControlSet\Control\Print');
11974: REGSTR_PATH_PRINTERS', 'System\CurrentControlSet\Control\Print\Printers');
11975: REGSTR_PATH_PROVIDERS', 'System\CurrentControlSet\Control\Print\Providers');
11976: REGSTR_PATH_MONITORS', 'System\CurrentControlSet\Control\Print\Monitors');
11977: REGSTR_PATH_ENVIRONMENTS', 'System\CurrentControlSet\Control\Print\Environments');
11978: REGSTR_VAL_START_ON_BOOT', 'StartOnBoot');
11979: REGSTR_VAL_PRINTERS_MASK', 'PrintersMask');
11980: REGSTR_VAL_DOS_SPOOL_MASK', 'DOSPoolMask');
11981: REGSTR_KEY_CURRENT_ENV', '\Windows 4');
11982: REGSTR_KEY_DRIVERS', '\Drivers');
11983: REGSTR_KEY_PRINT_PROC', '\Print Processors');
11984: REGSTR_PATH_EVENTLABELS', 'AppEvents\EventLabels');
11985: REGSTR_PATH_SCHEMES', 'AppEvents\Schemes');
11986: REGSTR_PATH_APPS', REGSTR_PATH_SCHEMES + '\Apps');
11987: REGSTR_PATH_APPS_DEFAULT', REGSTR_PATH_SCHEMES + '\Apps.Default');
11988: REGSTR_PATH_NAMES', REGSTR_PATH_SCHEMES + '\Names');
11989: REGSTR_PATH_MULTIMEDIA', REGSTR_PATH_SETUP + '\Multimedia');
11990: REGSTR_PATH_MULTIMEDIA_AUDIO', 'Software\Microsoft\Multimedia\Audio');
11991: REGSTR_PATH_MEDIARESOURCES', REGSTR_PATH_CURRENT_CONTROL_SET + '\MediaResources');
11992: REGSTR_PATH_MEDIAPROPERTIES', REGSTR_PATH_CURRENT_CONTROL_SET + '\MediaProperties');
11993: REGSTR_PATH_PRIVATEPROPERTIES', REGSTR_PATH_MEDIAPROPERTIES + '\PrivateProperties');
11994: REGSTR_PATH_PUBLICPROPERTIES', REGSTR_PATH_MEDIAPROPERTIES + '\PublicProperties');
11995: REGSTR_PATH_JOYOEM', REGSTR_PATH_PRIVATEPROPERTIES + '\Joystick\OEM');
11996: REGSTR_PATH_JOYCONFIG', REGSTR_PATH_MEDIARESOURCES + '\Joystick');
11997: REGSTR_KEY_JOYCURR', 'CurrentJoystickSettings');

```



```

11998: REGSTR_KEY_JOYSETTINGS', 'JoystickSettings');
11999: REGSTR_VAL_JOYUSERVALUES', 'JoystickUserValues');
12000: REGSTR_VAL_JOYCALLOUT', 'JoystickCallout');
12001: REGSTR_VAL_JOYNCONFIG', 'Joystick%dConfiguration');
12002: REGSTR_VAL_JOYNOEMNAME', 'Joystick%dOEMName');
12003: REGSTR_VAL_JOYNOEMCALLOUT', 'Joystick%dOEMCallout');
12004: REGSTR_VAL_JOYOEMCALLOUT', 'OEMCallout');
12005: REGSTR_VAL_JOYOEMNAME', 'OEMName');
12006: REGSTR_VAL_JOYOEMDATA', 'OEMData');
12007: REGSTR_VAL_JOYOEMXYLABEL', 'OEMXYLabel');
12008: REGSTR_VAL_JOYOEMZLABEL', 'OEMZLabel');
12009: REGSTR_VAL_JOYOEMRLABEL', 'OEMRLLabel');
12010: REGSTR_VAL_JOYOEMPOVLABEL', 'OEMPOVLabel');
12011: REGSTR_VAL_JOYOEMULABEL', 'OEMULLabel');
12012: REGSTR_VAL_JOYOEMVLABEL', 'OEMVLLabel');
12013: REGSTR_VAL_JOYOEMTESTMOVEDESC', 'OEMTestMoveDesc');
12014: REGSTR_VAL_JOYOEMTESTBUTTONDESC', 'OEMTestButtonDesc');
12015: REGSTR_VAL_JOYOEMTESTMOVECAP', 'OEMTestMoveCap');
12016: REGSTR_VAL_JOYOEMTESTBUTTONCAP', 'OEMTestButtonCap');
12017: REGSTR_VAL_JOYOEMTESTWINCAP', 'OEMTestWinCap');
12018: REGSTR_VAL_JOYOEMCALCAP', 'OEMCalCap');
12019: REGSTR_VAL_JOYOEMCALWINCAP', 'OEMCalWinCap');
12020: REGSTR_VAL_JOYOEMCAL1', 'OEMCal1');
12021: REGSTR_VAL_JOYOEMCAL2', 'OEMCal2');
12022: REGSTR_VAL_JOYOEMCAL3', 'OEMCal3');
12023: REGSTR_VAL_JOYOEMCAL4', 'OEMCal4');
12024: REGSTR_VAL_JOYOEMCAL5', 'OEMCal5');
12025: REGSTR_VAL_JOYOEMCAL6', 'OEMCal6');
12026: REGSTR_VAL_JOYOEMCAL7', 'OEMCal7');
12027: REGSTR_VAL_JOYOEMCAL8', 'OEMCal8');
12028: REGSTR_VAL_JOYOEMCAL9', 'OEMCal9');
12029: REGSTR_VAL_JOYOEMCAL10', 'OEMCal10');
12030: REGSTR_VAL_JOYOEMCAL11', 'OEMCal11');
12031: REGSTR_VAL_JOYOEMCAL12', 'OEMCal12');
12032: end;
12033:
12034: procedure SIRegister_WavePlay(CL: TPSPascalCompiler);
12035: begin
12036:   CL.AddConstantN('DefaultResType', 'String').SetString( 'WAVE');
12037:   CL.AddTypeS('TWave', 'TMemoryStream');
12038:   SIRegister_TWavePlayer(CL);
12039: end;
12040:
12041: procedure SIRegister_NamedPipes(CL: TPSPascalCompiler);
12042: begin
12043:   'DEFAULT_PIPE_BUFFER_SIZE', 'LongInt').SetInt( 4096);
12044:   'DEFAULT_PIPE_TIMEOUT', 'LongInt').SetInt( 5000);
12045:   CL.AddConstantN('PIPE_NAMING_SCHEME', 'String').SetString( '\\%s\pipe\%s');
12046:   CL.AddConstantN('WAIT_ERROR', 'LongWord').SetUInt( DWORD ( $FFFFFFF ));
12047:   CL.AddConstantN('WAIT_OBJECT_1', 'LongInt').SetInt( WAIT_OBJECT_0 + 1);
12048:   CL.AddConstantN('STATUS_SUCCESS', 'LongWord').SetUInt( $00000000);
12049:   CL.AddConstantN('STATUS_BUFFER_OVERFLOW', 'LongWord').SetUInt( $80000005);
12050:   CL.AddTypeS('TPipeDirection', '( pdir_Duplex, pdir_ClientToServer, pdir_ServerToClient )');
12051:   CL.AddTypeS('TPipeType', '( ptyp_ByteByte, ptyp_MsgByte, ptyp_MsgMsg )');
12052:   CL.AddTypeS('TOverlappedResult', '( ov_Failed, ov_Pending, ov_MoreData, ov_Complete )');
12053:   SIRegister_TNamedPipe(CL);
12054:   SIRegister_TServerPipe(CL);
12055:   SIRegister_TClientPipe(CL);
12056: end;
12057:
12058: type
12059:   TRoundToRange = -37..37;
12060: type
12061:   TRoundToEXRangeExtended = -20..20;
12062:
12063: procedure SIRegister_DPUtils(CL: TPSPascalCompiler);
12064: SIRegister_TThumbData(CL);
12065: 'PIC_BMP', 'LongInt').SetInt( 0);
12066: CL.AddConstantN('PIC_JPG', 'LongInt').SetInt( 1);
12067: 'THUMB_WIDTH', 'LongInt').SetInt( 60);
12068: 'THUMB_HEIGHT', 'LongInt').SetInt( 60);
12069:
12070: procedure SIRegister_CommonTools(CL: TPSPascalCompiler);
12071: begin
12072:   '//BYTES', 'LongInt').SetInt( 1);
12073:   'KBYES', 'LongInt').SetInt( 1024);
12074:   'DBG_ALIVE', 'LongWord').SetUInt( Integer ( $11BABEL1 ));
12075:   'DBG_DESTROYING', 'LongWord').SetUInt( Integer ( $44FADE44 ));
12076:   'DBG_GONE', 'LongWord').SetUInt( $99ACID99);
12077:   'SHELL_NS_MYCOMPUTER', 'String').SetString( '::{2D04FE0-3AEA-1069-A2D8-08002B30309D}');
12078:   SIRegister_MakeComServerMethodsPublic(CL);
12079:   CL.AddTypeS('TSomeFileInfo', '( fi_DisplayType, fi_Application )');
12080: end;
12081:
12082: procedure SIRegister_D2_VistaHelperU(CL: TPSPascalCompiler);
12083: begin
12084:   'PRODUCT_UNDEFINED', 'LongWord').SetUInt( $00000000);
12085:   'PRODUCT_ULTIMATE', 'LongWord').SetUInt( $00000001);
12086:   'PRODUCT_HOME_BASIC', 'LongWord').SetUInt( $00000002);

```

```

12087: 'PRODUCT_HOME_PREMIUM', 'LongWord').SetUInt( $00000003);
12088: 'PRODUCT_ENTERPRISE', 'LongWord').SetUInt( $00000004);
12089: 'PRODUCT_HOME_BASIC_N', 'LongWord').SetUInt( $00000005);
12090: 'PRODUCT_BUSINESS', 'LongWord').SetUInt( $00000006);
12091: 'PRODUCT_STANDARD_SERVER', 'LongWord').SetUInt( $00000007);
12092: 'PRODUCT_DATACENTER_SERVER', 'LongWord').SetUInt( $00000008);
12093: 'PRODUCT_SMALLBUSINESS_SERVER', 'LongWord').SetUInt( $00000009);
12094: 'PRODUCT_ENTERPRISE_SERVER', 'LongWord').SetUInt( $0000000A);
12095: 'PRODUCT_STARTER', 'LongWord').SetUInt( $0000000B);
12096: 'PRODUCT_DATACENTER_SERVER_CORE', 'LongWord').SetUInt( $0000000C);
12097: 'PRODUCT_STANDARD_SERVER_CORE', 'LongWord').SetUInt( $0000000D);
12098: 'PRODUCT_ENTERPRISE_SERVER_CORE', 'LongWord').SetUInt( $0000000E);
12099: 'PRODUCT_ENTERPRISE_SERVER_IA64', 'LongWord').SetUInt( $0000000F);
12100: 'PRODUCT_BUSINESS_N', 'LongWord').SetUInt( $00000010);
12101: 'PRODUCT_WEB_SERVER', 'LongWord').SetUInt( $00000011);
12102: 'PRODUCT_CLUSTER_SERVER', 'LongWord').SetUInt( $00000012);
12103: 'PRODUCT_HOME_SERVER', 'LongWord').SetUInt( $00000013);
12104: 'PRODUCT_STORAGE_EXPRESS_SERVER', 'LongWord').SetUInt( $00000014);
12105: 'PRODUCT_STORAGE_STANDARD_SERVER', 'LongWord').SetUInt( $00000015);
12106: 'PRODUCT_STORAGE_WORKGROUP_SERVER', 'LongWord').SetUInt( $00000016);
12107: 'PRODUCT_STORAGE_ENTERPRISE_SERVER', 'LongWord').SetUInt( $00000017);
12108: 'PRODUCT_SERVER_FOR_SMALLBUSINESS', 'LongWord').SetUInt( $00000018);
12109: 'PRODUCT_SMALLBUSINESS_SERVER_PREMIUM', 'LongWord').SetUInt( $00000019);
12110: 'PRODUCT_UNLICENSED', 'LongWord').SetUInt( $ABCDABCD);
12111: //CL.AddTypes('OSVERSIONINFOEXA', '_OSVERSIONINFOEXA');
12112: //CL.AddTypes('TOSVersionInfoExA', '_OSVERSIONINFOEXA');
12113: CL.AddTypes('TOSVersionInfoEx', 'TOSVersionInfo');
12114: CL.AddTypes('TDrivesProperty', 'array[1..26] of boolean;');
12115: //TDrivesProperty = array['A'..'Z'] of boolean;
12116: end;
12117:
12118: procedure SIRegister_BetterADODataset(CL: TPSPascalCompiler);
12119: begin
12120: CL.AddConstantN('BetterADODatasetVersion', 'String').SetString( '4.04');
12121: TUpdateCriteria, '( adCriteriaKey, adCriteriaAllCols, adCriteria
12122:   +aUpdCols, adCriteriaTimeStamp )';
12123: TUpdateResyncEnum, '( ResyncAutoIncrement, ResyncConflicts, Res
12124:   +yncUpdates, ResyncInserts )';
12125: TUpdateResync, 'set of TUpdateResyncEnum';
12126: TRefreshType, '( rtRequery, rtResyncCurrent, rtResyncGroup, rtR
12127:   +esyncAll, rtResyncChapters )';
12128: TJoinsResolution, '( jrAuto, jrManual, jrNone )';
12129: TJointTest, '( jtInactive, jtNotJoined, jtJoined )';
12130: TStartPos, '( spFirstRecord, spCurrentRecord )';
12131: // CL.AddTypes('TRecordBuffer', '(PChar)');
12132: CL.AddClassN(CL.FindClass('TObject'), 'TBetterADODataset');
12133: SIRegister_TADOVersion(CL);
12134: SIRegister_TBetterADODataset(CL);
12135: CL.AddDelphiFunction('Procedure Register');
12136: end;
12137:
12138: procedure SIRegister_pipes(CL: TPSPascalCompiler);
12139: begin
12140: //CL.AddTypes('PSecurityAttributes', '^TSecurityAttributes // will not work');
12141: CL.AddTypes('TSecurityAttributes', 'record nlength : Integer; lpSecurityDescriptor : __Pointer;
BinheritHandle : Boolean; end');
12142: CL.AddClassN(CL.FindClass('TObject'), 'ENoReadPipe');
12143: CL.AddClassN(CL.FindClass('TObject'), 'ENoWritePipe');
12144: CL.AddClassN(CL.FindClass('TObject'), 'EPipeSeek');
12145: CL.AddClassN(CL.FindClass('TObject'), 'EPipeCreation');
12146: SIRegister_TPipeStream(CL);
12147: SIRegister_TInputPipeStream(CL);
12148: SIRegister_TOutputPipeStream(CL);
12149: end;
12150:
12151: procedure SIRegister_dwsXPlatform(CL: TPSPascalCompiler);
12152: begin
12153: 'cLineTerminator', 'Char').SetString( #10);
12154: 'cLineTerminators', 'String').SetString( #13#10);
12155: 'INVALID_HANDLE_VALUE', 'LongInt').SetInt( DWORD ( - 1 ));
12156: SIRegister_TFixedCriticalSection(CL);
12157: SIRegister_TMultiReadSingleWrite(CL);
12158: Procedure SetDecimalSeparator( c : Char);
12159: Function GetDecimalSeparator : Char;
12160: TCollectFileProgressEvent, 'Procedure ( const directory : Strin
12161:   +g; var skipScan : Boolean);
12162: CL.AddDelphiFunction('Procedure CollectFiles( const directory, fileMask : UnicodeString; list : TStrings;
recurseSubdirectories : Boolean; onProgress : TCollectFileProgressEvent);');
12163: CL.AddTypes('NativeInt', 'Integer');
12164: //CL.AddTypes('PNativeInt', '^NativeInt // will not work');
12165: CL.AddTypes('NativeUInt', 'Cardinal');
12166: //CL.AddTypes('PNativeUInt', '^NativeUInt // will not work');
12167: //CL.AddTypes('TBytes', 'array of Byte');
12168: CL.AddTypes('RawByteString', 'UnicodeString');
12169: //CL.AddTypes('PNativeInt', '^NativeInt // will not work');
12170: //CL.AddTypes('PUInt64', '^UInt64 // will not work');
12171: SIRegister_TPath(CL);
12172: SIRegister_TFile(CL);
12173: SIRegister_TdwsThread(CL);

```

```

12174: end;
12175:
12176:
12177: procedure SIRegister_AdSocket(CL: TPSPascalCompiler);
12178: begin
12179:   'IPStrSize','LongInt').SetInt( 15);
12180:   'CM_APDSOCKETMESSAGE','LongWord').SetUInt( WM_USER + $0711);
12181:   'CM_APDSOCKETQUIT','LongWord').SetUInt( WM_USER + $0712);
12182:   'ADWSBASE','LongInt').SetInt( 9000);
12183:   CL.AddTypeS('TCMAPDSocketMessage', 'record Msg : Cardinal; Socket : TSocket; '
12184:     + 'SelectEvent : Word; SelectError : Word; Result : Longint; end');
12185:   SIRegister_EApdSocketException(CL);
12186:   TWsMode', '( wsClient, wsServer )');
12187:   TWsNotifyEvent', 'Procedure ( Sender : TObject; Socket : TSocket);
12188:   TWsSocketErrorEvent', 'Procedure ( Sender : TObject; Socket : TSocket; ErrCode : Integer)');
12189:   SIRegister_TApdSocket(CL);
12190: end;
12191:
12192: procedure SIRegister_AdPacket(CL: TPSPascalCompiler);
12193: begin
12194:   TPacketStartCond', '( scString, scAnyData )');
12195:   TPacketEndCond', '( ecString, ecPacketSize )');
12196:   TPacketEndSet', 'set of TPacketEndCond');
12197:   EscapeCharacter','string').SetString( '\');
12198:   WildCardCharacter','string').SetString( '?');
12199:   CL.AddConstantN('adpDefEnabled','Boolean');
12200:   adpDefIgnoreCase','Boolean');
12201:   adpDefIncludeStrings','Boolean');
12202:   adpDefAutoEnable','Boolean');
12203:   adpDefStartCond','string').SetString('scString');
12204:   adpDefTimeOut','LongInt').SetInt( 2184);
12205:   apdDefFlushOnTimeout','Boolean').SetInt( 2184);
12206:   CL.AddClassN(CL.FindClass('TObject'),'TApdDataPacket');
12207:   CL.AddClassN(CL.FindClass('TObject'),'TApdDataPacketManager');
12208:   SIRegister_TApdDataPacketManagerList(CL);
12209:   SIRegister_TApdDataPacketManager(CL);
12210:   TPacketMode', '( dpIdle, dpWaitStart, dpCollecting )');
12211:   TPacketNotifyEvent', 'Procedure ( Sender : TObject; Data : TObject; Size : Integer)');
12212:   TStringPacketNotifyEvent', 'Procedure ( Sender : TObject; Data : string)');
12213:   SIRegister_TApdDataPacket(CL);
12214: end;
12215:
12216: procedure SIRegister_AdPort;
12217: begin
12218:   TParity', '( pNone, pOdd, pEven, pMark, pSpace )');
12219:   TDeviceLayer', '( dlWin32, dlWinsock )');
12220:   TDeviceLayers', 'set of TDeviceLayer');
12221:   TadBaudRate', 'LongInt');
12222:   TTapiMode', '( tmNone, tmAuto, tmOn, tmOff )');
12223:   TPortState', '( psClosed, psShuttingDown, psOpen )');
12224:   THWFlowOptions', '( hwfUseDTR, hwfUserRTS, hwfRequireDSR, hwfRequireCTS )');
12225:   THWFlowOptionSet', 'set of THWFlowOptions');
12226:   TSWFlowOptions', '( swfNone, swfReceive, swfTransmit, swfBoth )');
12227:   TFlowControlState', '( fcOff, fcOn, fcDsrHold, fcCtsHold, fcDcdH
12228:     + 'old, fcXOutHold, fcXInHold, fcXBothHold )');
12229:   TTraceLogState', '( tloff, tlon, tldump, tlappend, tlcLEAR, tlpause )');
12230:   TadTriggerEvent', 'Procedure ( CP : TObject; Msg, TriggerHandle, Data : Word)');
12231:   TTriggerAvailEvent', 'Procedure ( CP : TObject; Count : Word)');
12232:   TTriggerDataEvent', 'Procedure ( CP : TObject; TriggerHandle : Word)');
12233:   TTriggerStatusEvent', 'Procedure ( CP : TObject; TriggerHandle : Word)');
12234:   TTriggerTimerEvent', 'Procedure ( CP : TObject; TriggerHandle : Word)');
12235:   TTriggerLineErrorEvent', 'Procedure ( CP : TObject; Error : Word; LineBreak : Boolean)');
12236:   TWaitCharEvent', 'Procedure ( CP : TObject; C : Char)');
12237:   TPortCallback', 'Procedure ( CP : TObject; Opening : Boolean)');
12238:   TApdCallbackType', '( ctOpen, ctClosing, ctClosed )');
12239:   TPortCallbackEx', 'Procedure ( CP : TObject; CallbackType : TApdCallbackType)');
12240:   PUserListEntry', '^TUserListEntry // will not work');
12241:   TUserListEntry', 'record Handle : THandle; OpenClose : TPortCall
12242:     + 'back; OpenCloseEx : TPortCallbackEx; IsEx : Boolean; end');
12243:   TApThreadBoost', '( tbNone, tbPlusOne, tbPlusTwo )');
12244:   adpoDefDeviceLayer','string').SetString('dlWin32');
12245:   CL.AddConstantN('adpoDefPromptForPort','Boolean');
12246:   adpoDefComNumber','LongInt').SetInt( 0);
12247:   adpoDefBaudRt','LongInt').SetInt( 19200);
12248:   //adpoDefParity','integer').SetSet(pNone);
12249:   adpoDefDatabits','LongInt').SetInt( 8);
12250:   adpoDefStopbits','LongInt').SetInt( 1);
12251:   adpoDefInSize','LongInt').SetInt( 4096);
12252:   adpoDefOutSize','LongInt').SetInt( 4096);
12253:   //adpoDefOpen','Boolean').SetInt( 4096);
12254:   //adpoDefAutoOpen','Boolean').SetInt( 4096);
12255:   //adpoDefBaseAddress','LongInt').SetInt( 0);
12256:   {adpoDefTapiMode','}).SetString( tmAuto);
12257:   adpoDefDTR','Boolean').SetString( tmAuto);
12258:   adpoDefRTS','Boolean').SetString( tmAuto);
12259:   adpoDefTracing','}).SetString( tloff);}
12260:   adpoDefTraceSize','LongInt').SetInt( 10000);
12261:   adpoDefTraceName','string').SetString( 'APRO.TRC');
12262:   adpoDefTraceHex','string').SetString( 'APRO.TRC');

```

```

12263: adpoDefTraceAllHex','string').SetString( 'APRO.TRC');
12264: //adpoDefLogging','').SetString( t10ff);
12265: adpoDefLogSize','LongInt').SetInt( 10000);
12266: adpoDefLogName','string').SetString( 'APRO.LOG');
12267: adpoDefLogHex','string').SetString( 'APRO.LOG');
12268: adpoDefLogAllHex','string').SetString( 'APRO.LOG');
12269: adpoDefUseMSRShadow','string').SetString( 'APRO.LOG');
12270: adpoDefUseEventWord','string').SetString( 'APRO.LOG');
12271: adpoDefSWFlowOptions','string').SetString( ' swfNone');
12272: adpoDefXonChar','Char').SetString( #17);
12273: adpoDefXoffChar','Char').SetString( #19);
12274: adpoDefBufferFull','LongInt').SetInt( 0);
12275: adpoDefBufferResume','LongInt').SetInt( 0);
12276: adpoDefTriggerLength','LongInt').SetInt( 1);
12277: adpoDefCommNotificationLevel','LongInt').SetInt( 10);
12278: adpoDefRS485Mode','Boolean').SetInt( 10);
12279: //SIRegister_TApdCustomComPort(CL);
12280: //SIRegister_TApdComPort(CL);
12281: Function ComName( const ComNumber : Word) : String;
12282: Function SearchComPort( const C : TComponent) : TApdCustomComPort;
12283: //ComName
12284: //SearchComPort
12285: end;
12286:
12287: procedure SIRegister_CmnFunc2(CL: TPSPascalCompiler);
12288: begin
12289:   NEWREGSTR_PATH_SETUP','String').SetString( 'Software\Microsoft\Windows\CurrentVersion');
12290:   NEWREGSTR_PATH_EXPLORER','String').SetString( NEWREGSTR_PATH_SETUP + '\Explorer');
12291:   NEWREGSTR_PATH_SPECIAL_FOLDERS','String').SetString(NEWREGSTR_PATH_EXPLORER + '\Shell Folders');
12292:   NEWREGSTR_PATH_UNINSTALL','String').SetString( NEWREGSTR_PATH_SETUP + '\Uninstall');
12293:   NEWREGSTR_VAL_UNINSTALLER_DISPLAYNAME','String').SetString( 'DisplayName');
12294:   NEWREGSTR_VAL_UNINSTALLER_COMMANDLINE','String').SetString( 'UninstallString');
12295:   KEY_WOW64_64KEY','LongWord').SetUInt( $0100);
12296:   //CL.AddTypeS('TLeadByteSet', '^TLeadByteSet // will not work');
12297:   CL.AddTypeS('TLeadByteSet', 'set of Char');
12298:   SIRegister_TOneShotTimer(CL);
12299:   CL.AddTypeS('TRegView', '( rvDefault, rv32Bit, rv64Bit )');
12300:   'RegViews64Bit','LongInt').Value.ts32 := ord(rv64Bit);
12301:   SIRegister_TWindowDisabler(CL);
12302:   TMsgBoxType','( mbInformation, mbConfirmation, mbError, mbCriticalError )');
12303:   TMsgBoxCallbackFunc','procedure(const Flags:LongInt;const After:Bool;const Param:LongInt);
12304:   CL.AddTypeS('TFindRec',
12305:     record +
12306:       Name: String;' +
12307:       Attributes: LongWord;' +
12308:       SizeHigh: LongWord;' +
12309:       SizeLow: LongWord;' +
12310:       CreationTime: TFileTime;' +
12311:       LastAccessTime: TFileTime;' +
12312:       LastWriteTime: TFileTime;' +
12313:       AlternateName: String;' +
12314:       FindHandle: THandle;' +
12315:     end');
12316: end;
12317:
12318: procedure SIRegister_ImageGrabber(CL: TPSPascalCompiler);
12319: begin
12320:   TCaptureDriverDialog','( cddDisplay, cddFormat, cddSource )');
12321:   TCaptureDriverDialogs','set of TCaptureDriverDialog');
12322:   CL.AddClassN(CL.FindClass('TOBJECT'),'TCaptureDrivers');
12323:   CL.AddClassN(CL.FindClass('TOBJECT'),'TCaptureDriver');
12324:   SIRegister_TImageGrabber(CL);
12325:   SIRegister_TCaptureDrivers(CL);
12326:   SIRegister_TCaptureDriver(CL);
12327:   TOBJECT'),'EImageGrabber');
12328:   TOBJECT'),'ECouldNotCreateCaptureWindow');
12329:   TOBJECT'),'EAlreadyConnected');
12330:   TOBJECT'),'ECaptureDriverNotConnected');
12331:   TOBJECT'),'ENoCaptureDriverSelected');
12332:   TOBJECT'),'ECouldNotConnectToCaptureDriver');
12333:   TOBJECT'),'ECouldNotDisconnectFromCaptureDriver');
12334:   TOBJECT'),'ENoCodecForVideoFormat');
12335:   TOBJECT'),'EUnableToCreateBitmap');
12336:   TOBJECT'),'ECodecError');
12337:   TOBJECT'),'EGrabError');
12338:   TOBJECT'),'ECouldNotOpenDialog');
12339: end;
12340:
12341:
12342: procedure SIRegister_SecurityFunc(CL: TPSPascalCompiler);
12343: begin
12344:   {_SID_IDENTIFIER_AUTHORITY = record
12345:     Value: array[0..5] of Byte;
12346:   end; }
12347:   CL.AddTypeS('_SID_IDENTIFIER_AUTHORITY',
12348:     record +
12349:       Value: array[0..5] of Byte;' +
12350:     end');
12351:   CL.AddTypeS('TSIDIdentifierAuthority', '_SID_IDENTIFIER_AUTHORITY');

```



```

12352: CL.AddTypeS('SID_IDENTIFIER_AUTHORITY', '_SID_IDENTIFIER_AUTHORITY');
12353: (* TGrantPermissionSid = record { must keep in synch with Helper.c }
12354:   Authority: TSIDIdentifierAuthority;
12355:   SubAuthCount: Byte;
12356:   SubAuth: array[0..1] of DWORD;
12357: end;*)
12358: CL.AddTypeS('TGrantPermissionSid',
12359:   record +
12360:     Authority: TSIDIdentifierAuthority; +
12361:     SubAuthCount: Byte; +
12362:     SubAuth: array[0..1] of DWORD; +
12363:   end);
12364: (* TGrantPermissionEntry = record { must keep in synch with Helper.c }
12365:   Sid: TGrantPermissionSid;
12366:   AccessMask: DWORD;
12367: end; *)
12368: CL.AddTypeS('TGrantPermissionEntry',
12369:   record +
12370:     Sid: TGrantPermissionSid; +
12371:     AccessMask: DWord; +
12372:   end);
12373: Function GrantPermissionOnFile( const DisableFsRedir : Boolean; Filename : String; const Entries :
TGrantPermissionEntry; const EntryCount : Integer) : Boolean);
12374: Function GrantPermissionOnKey( const RegView : TRegView; const RootKey : HKEY; const Subkey : String;
const Entries : TGrantPermissionEntry; const EntryCount : Integer) : Boolean);
12375: end;
12376:
12377: procedure SIRegister_RedirFunc(CL: TPSPascalCompiler);
12378: begin
12379:   TPreviousFsRedirectionState, 'record DidDisable : Boolean; OldValue : __Pointer; end');
12380: { type
12381:   TFileVersionNumbers = record
12382:     MS, LS: LongWord;
12383:   end;}
12384: CL.AddTypeS('TFileVersionNumbers',
12385:   record MS, LS: LongWord; +
12386:   end);
12387: end;
12388:
12389: procedure SIRegister_Int64Em(CL: TPSPascalCompiler);
12390: begin
12391:   //CL.AddTypeS('LongWord', 'Cardinal');
12392:   CL.AddTypeS('Integer64', record Lo : LongWord; Hi : LongWord; end);
12393: end;
12394:
12395: procedure SIRegister_SimpleExpression(CL: TPSPascalCompiler);
12396: begin
12397:   CL.AddClassN(CL.FindClass('TOBJECT'), 'TSimpleExpression');
12398:   CL.AddTypeS('TSimpleExpressionOnEvalIdentifier', 'Function ( Sender : TSimple'
+ 'Expression; const Name : String; const Parameters : array of const) : Boolean');
12400:   CL.AddTypeS('TSimpleExpressionOnExpandConstant', 'Function ( Sender : TSimple'
+ 'Expression; const Constant : String) : String');
12402:   SIRegister_TSimpleExpression(CL);
12403: end;
12404:
12405: procedure SIRegister_InstFunc(CL: TPSPascalCompiler);
12406: begin
12407:   //CL.AddTypeS('PSimpleStringListArray', '^TSimpleStringListArray // will not work');
12408:   SIRegister_TSimpleStringList(CL);
12409:   CL.AddTypeS('TExecWait', '( ewNoWait, ewWaitUntilTerminated, ewWaitUntilIdle)');
12410:   CL.AddTypeS('TDetermineDefaultLanguageResult', '( ddNoMatch, ddMatch, ddMatchLangParameter)');
12411:   CL.AddTypeS('TMD5Digest', array[0..15] of Byte);
12412:   CL.AddTypeS('TSHA1Digest', array[0..19] of Byte);
12413: end;
12414:
12415: procedure SIRegister_Console(CL: TPSPascalCompiler);
12416: begin
12417:   CL.AddConstantN('Black', 'LongInt').SetInt( 0);
12418:   Blue, 'LongInt').SetInt( 1);
12419:   'Green', 'LongInt').SetInt( 2);
12420:   'Cyan', 'LongInt').SetInt( 3);
12421:   'Red', 'LongInt').SetInt( 4);
12422:   CL.AddConstantN('Magenta', 'LongInt').SetInt( 5);
12423:   'Brown', 'LongInt').SetInt( 6);
12424:   'LightGray', 'LongInt').SetInt( 7);
12425:   'DarkGray', 'LongInt').SetInt( 8);
12426:   'LightBlue', 'LongInt').SetInt( 9);
12427:   'LightGreen', 'LongInt').SetInt( 10);
12428:   'LightCyan', 'LongInt').SetInt( 11);
12429:   'LightRed', 'LongInt').SetInt( 12);
12430:   'LightMagenta', 'LongInt').SetInt( 13);
12431:   'Yellow', 'LongInt').SetInt( 14);
12432:   'White', 'LongInt').SetInt( 15); }
12433: // 'Blink', 'LongInt').SetInt( 128);
12434: 'conBW40', 'LongInt').SetInt( 0);
12435: 'conCO40', 'LongInt').SetInt( 1);
12436: 'conBW80', 'LongInt').SetInt( 2);
12437: 'conCO80', 'LongInt').SetInt( 3);
12438: 'conMono', 'LongInt').SetInt( 7);

```

```

12439: 'conFont8x8','LongInt').SetInt( 256);
12440: //'C40','').SetString( C040);
12441: //'C80','').SetString( C080);
12442: end;
12443:
12444:
12445: procedure SIRegister_XPrinter(CL: TPSPascalCompiler);
12446: begin
12447:   TXPrinterOrientation', '( xpoPortrait, xpoLandscape )');
12448:   TXPrinterCapability', '( xpcCopies, xpcOrientation, xpcCollation)');
12449:   TXPrinterCapabilities', 'set of TXPrinterCapability');
12450:   TScaling', '( pscDot, pscMil, pscMetric10, pscMetric100 )');
12451:   TXPrintConfig', 'record Valid : boolean; CurrentPrinter : integer;
12452:   r; Scaling : TScaling; MarginLeft : integer; MarginTop : integer; MarginRight : integer;
12453:   MarginBottom : integer; Font_Charset : integer; Font_Color : TColor;
12454:   Font_Height : integer; Font_Name : TFontString; Font_Pitch : TFontPitch;
12455:   Font_Size : integer; Font_Style : TFontStyles; Copies : integer;
12456:   Orientation : TXPrinterOrientation; Angle10 : integer; TextRotation : boolean;
12457:   Title : TXTitleString; end';
12458:   SIRegister_TXPrint(CL);
12459:   //CL.AddDelphiFunction('Procedure Register');
12460: end;
12461:
12462: procedure SIRegister_ToolsUnit(CL: TPSPascalCompiler);
12463: begin
12464:   'MaxDataSet','LongInt').SetInt( 35);
12465:   //CL.AddTypeS('TDBConnectorClass', 'class of TDBConnector');
12466:   SIRegister_TDBConnector(CL);
12467:   SIRegister_TDBBasicsTestSetup(CL);
12468:   SIRegister_TTestDataLink(CL);
12469:   'testValuesCount','LongInt').SetInt( 25);
12470: end;
12471:
12472: procedure SIRegister_fpcunit(CL: TPSPascalCompiler);
12473: begin
12474:   SIRegister_EAssertionFailedError(CL);
12475:   CL.AddTypeS('TTestStep', '( stSetUp, stRunTest, stTearDown, stNothing )');
12476:   CL.AddTypeS('TRunMethod', 'Procedure');
12477:   CL.AddClassN(CL.FindClass('TOBJECT'),'TTestResult');
12478:   SIRegister_TTest(CL);
12479:   SIRegister_TAssert(CL);
12480:   SIRegister_TTestFailure(CL);
12481:   SIRegister_ITestListener(CL);
12482:   SIRegister_TTestCase(CL);
12483:   //CL.AddTypeS('TTestCaseClass', 'class of TTestCase');
12484:   SIRegister_TTestSuite(CL);
12485:   SIRegister_TTestResult(CL);
12486: end;
12487:
12488: procedure SIRegister_cTCPBuffer(CL: TPSPascalCompiler);
12489: begin
12490:   TOBJECT'),'ETCPBuffer');
12491:   'TTCPBuffer', 'record Ptr : TObject; Size : Integer; Max : Integer;
12492:   Head : Integer; Used : Integer; end';
12493:   'ETHERNET_MTU_100MBIT','LongInt').SetInt( 1500);
12494:   'ETHERNET_MTU_1GBIT','LongInt').SetInt( 9000);
12495:   'TCP_BUFFER_DEFAULTMAXSIZE','LongInt').SetInt( ETHERNET_MTU_1GBIT * 4);
12496:   'TCP_BUFFER_DEFAULTBUFSIZE','LongInt').SetInt( ETHERNET_MTU_100MBIT * 4);
12497: end;
12498:
12499:
12500: procedure SIRegister_Glut(CL: TPSPascalCompiler);
12501: begin
12502:   //CL.AddTypeS('PInteger', '^Integer // will not work');
12503:   //CL.AddTypeS('PPChar', '^PChar // will not work');
12504:   'GLUT_API_VERSION','LongInt').SetInt( 3);
12505:   CL.AddConstantN('GLUT_XLIB_IMPLEMENTATION','LongInt').SetInt( 12);
12506:   'GLUT_RGB','LongInt').SetInt( 0);
12507:   //'GLUT_RGBA','').SetString( GLUT_RGB);
12508:   'GLUT_INDEX','LongInt').SetInt( 1);
12509:   'GLUT_SINGLE','LongInt').SetInt( 0);
12510:   'GLUT_DOUBLE','LongInt').SetInt( 2);
12511:   'GLUT_ACCUM','LongInt').SetInt( 4);
12512:   'GLUT_ALPHA','LongInt').SetInt( 8);
12513:   'GLUT_DEPTH','LongInt').SetInt( 16);
12514:   'GLUT_STENCIL','LongInt').SetInt( 32);
12515:   'GLUT_MULTISAMPLE','LongInt').SetInt( 128);
12516:   'GLUT_STEREO','LongInt').SetInt( 256);
12517:   'GLUT_LUMINANCE','LongInt').SetInt( 512);
12518:   'GLUT_LEFT_BUTTON','LongInt').SetInt( 0);
12519:   'GLUT_MIDDLE_BUTTON','LongInt').SetInt( 1);
12520:   'GLUT_RIGHT_BUTTON','LongInt').SetInt( 2);
12521:   'GLUT_DOWN','LongInt').SetInt( 0);
12522:   'GLUT_UP','LongInt').SetInt( 1);
12523:   'GLUT_KEY_F1','LongInt').SetInt( 1);
12524:   'GLUT_KEY_F2','LongInt').SetInt( 2);
12525:   'GLUT_KEY_F3','LongInt').SetInt( 3);
12526:   'GLUT_KEY_F4','LongInt').SetInt( 4);
12527:   'GLUT_KEY_F5','LongInt').SetInt( 5);

```

```

12528: 'GLUT_KEY_F6', 'LongInt').SetInt( 6);
12529: 'GLUT_KEY_F7', 'LongInt').SetInt( 7);
12530: 'GLUT_KEY_F8', 'LongInt').SetInt( 8);
12531: 'GLUT_KEY_F9', 'LongInt').SetInt( 9);
12532: 'GLUT_KEY_F10', 'LongInt').SetInt( 10);
12533: 'GLUT_KEY_F11', 'LongInt').SetInt( 11);
12534: 'GLUT_KEY_F12', 'LongInt').SetInt( 12);
12535: 'GLUT_KEY_LEFT', 'LongInt').SetInt( 100);
12536: 'GLUT_KEY_UP', 'LongInt').SetInt( 101);
12537: 'GLUT_KEY_RIGHT', 'LongInt').SetInt( 102);
12538: 'GLUT_KEY_DOWN', 'LongInt').SetInt( 103);
12539: 'GLUT_KEY_PAGE_UP', 'LongInt').SetInt( 104);
12540: 'GLUT_KEY_PAGE_DOWN', 'LongInt').SetInt( 105);
12541: 'GLUT_KEY_HOME', 'LongInt').SetInt( 106);
12542: 'GLUT_KEY_END', 'LongInt').SetInt( 107);
12543: 'GLUT_KEY_INSERT', 'LongInt').SetInt( 108);
12544: 'GLUT_LEFT', 'LongInt').SetInt( 0);
12545: 'GLUT_ENTERED', 'LongInt').SetInt( 1);
12546: 'GLUT_MENU_NOT_IN_USE', 'LongInt').SetInt( 0);
12547: 'GLUT_MENU_IN_USE', 'LongInt').SetInt( 1);
12548: 'GLUT_NOT_VISIBLE', 'LongInt').SetInt( 0);
12549: 'GLUT_VISIBLE', 'LongInt').SetInt( 1);
12550: 'GLUT_HIDDEN', 'LongInt').SetInt( 0);
12551: 'GLUT_FULLY_RETAINED', 'LongInt').SetInt( 1);
12552: 'GLUT_PARTIALLY_RETAINED', 'LongInt').SetInt( 2);
12553: 'GLUT_FULLY_COVERED', 'LongInt').SetInt( 3);
12554: 'GLUT_RED', 'LongInt').SetInt( 0);
12555: 'GLUT_GREEN', 'LongInt').SetInt( 1);
12556: 'GLUT_BLUE', 'LongInt').SetInt( 2);
12557: 'GLUT_NORMAL', 'LongInt').SetInt( 0);
12558: 'GLUT_OVERLAY', 'LongInt').SetInt( 1);
12559: { 'GLUT_STROKE_ROMAN', 'LongInt').SetInt( Pointer ( 0 ));
12560: 'GLUT_STROKE_MONO_ROMAN', 'LongInt').SetInt( Pointer ( 1 ));
12561: 'GLUT_BITMAP_9_BY_15', 'LongInt').SetInt( Pointer ( 2 ));
12562: 'GLUT_BITMAP_8_BY_13', 'LongInt').SetInt( Pointer ( 3 ));
12563: 'GLUT_BITMAP_TIMES_ROMAN_10', 'LongInt').SetInt( Pointer ( 4 ));
12564: 'GLUT_BITMAP_TIMES_ROMAN_24', 'LongInt').SetInt( Pointer ( 5 ));
12565: 'GLUT_BITMAP_HELVETICA_10', 'LongInt').SetInt( Pointer ( 6 ));
12566: 'GLUT_BITMAP_HELVETICA_12', 'LongInt').SetInt( Pointer ( 7 ));
12567: 'GLUT_BITMAP_HELVETICA_18', 'LongInt').SetInt( Pointer ( 8 ));}
12568: 'GLUT_WINDOW_X', 'LongInt').SetInt( 100);
12569: 'GLUT_WINDOW_Y', 'LongInt').SetInt( 101);
12570: 'GLUT_WINDOW_WIDTH', 'LongInt').SetInt( 102);
12571: 'GLUT_WINDOW_HEIGHT', 'LongInt').SetInt( 103);
12572: 'GLUT_WINDOW_BUFFER_SIZE', 'LongInt').SetInt( 104);
12573: 'GLUT_WINDOW_STENCIL_SIZE', 'LongInt').SetInt( 105);
12574: 'GLUT_WINDOW_DEPTH_SIZE', 'LongInt').SetInt( 106);
12575: 'GLUT_WINDOW_RED_SIZE', 'LongInt').SetInt( 107);
12576: 'GLUT_WINDOW_GREEN_SIZE', 'LongInt').SetInt( 108);
12577: 'GLUT_WINDOW_BLUE_SIZE', 'LongInt').SetInt( 109);
12578: 'GLUT_WINDOW_ALPHA_SIZE', 'LongInt').SetInt( 110);
12579: 'GLUT_WINDOW_ACCUM_RED_SIZE', 'LongInt').SetInt( 111);
12580: 'GLUT_WINDOW_ACCUM_GREEN_SIZE', 'LongInt').SetInt( 112);
12581: 'GLUT_WINDOW_ACCUM_BLUE_SIZE', 'LongInt').SetInt( 113);
12582: 'GLUT_WINDOW_ACCUM_ALPHA_SIZE', 'LongInt').SetInt( 114);
12583: 'GLUT_WINDOW_DOUBLEBUFFER', 'LongInt').SetInt( 115);
12584: 'GLUT_WINDOW_RGBA', 'LongInt').SetInt( 116);
12585: 'GLUT_WINDOW_PARENT', 'LongInt').SetInt( 117);
12586: 'GLUT_WINDOW_NUM_CHILDREN', 'LongInt').SetInt( 118);
12587: 'GLUT_WINDOW_COLORMAP_SIZE', 'LongInt').SetInt( 119);
12588: 'GLUT_WINDOW_NUM_SAMPLES', 'LongInt').SetInt( 120);
12589: 'GLUT_WINDOW_STEREO', 'LongInt').SetInt( 121);
12590: 'GLUT_WINDOW_CURSOR', 'LongInt').SetInt( 122);
12591: 'GLUT_SCREEN_WIDTH', 'LongInt').SetInt( 200);
12592: 'GLUT_SCREEN_HEIGHT', 'LongInt').SetInt( 201);
12593: 'GLUT_SCREEN_WIDTH_MM', 'LongInt').SetInt( 202);
12594: 'GLUT_SCREEN_HEIGHT_MM', 'LongInt').SetInt( 203);
12595: 'GLUT_MENU_NUM_ITEMS', 'LongInt').SetInt( 300);
12596: 'GLUT_DISPLAY_MODE_POSSIBLE', 'LongInt').SetInt( 400);
12597: 'GLUT_INIT_WINDOW_X', 'LongInt').SetInt( 500);
12598: 'GLUT_INIT_WINDOW_Y', 'LongInt').SetInt( 501);
12599: 'GLUT_INIT_WINDOW_WIDTH', 'LongInt').SetInt( 502);
12600: 'GLUT_INIT_WINDOW_HEIGHT', 'LongInt').SetInt( 503);
12601: 'GLUT_INIT_DISPLAY_MODE', 'LongInt').SetInt( 504);
12602: 'GLUT_ELAPSED_TIME', 'LongInt').SetInt( 700);
12603: 'GLUT_HAS_KEYBOARD', 'LongInt').SetInt( 600);
12604: 'GLUT_HAS_MOUSE', 'LongInt').SetInt( 601);
12605: 'GLUT_HAS_SPACEBALL', 'LongInt').SetInt( 602);
12606: 'GLUT_HAS_DIAL_AND_BUTTON_BOX', 'LongInt').SetInt( 603);
12607: 'GLUT_HAS_TABLET', 'LongInt').SetInt( 604);
12608: 'GLUT_NUM_MOUSE_BUTTONS', 'LongInt').SetInt( 605);
12609: 'GLUT_NUM_SPACEBALL_BUTTONS', 'LongInt').SetInt( 606);
12610: 'GLUT_NUM_BUTTON_BOX_BUTTONS', 'LongInt').SetInt( 607);
12611: 'GLUT_NUM_DIALS', 'LongInt').SetInt( 608);
12612: 'GLUT_NUM_TABLET_BUTTONS', 'LongInt').SetInt( 609);
12613: 'GLUT_OVERLAY_POSSIBLE', 'LongInt').SetInt( 800);
12614: 'GLUT_LAYER_IN_USE', 'LongInt').SetInt( 801);
12615: 'GLUT_HAS_OVERLAY', 'LongInt').SetInt( 802);
12616: 'GLUT_TRANSPARENT_INDEX', 'LongInt').SetInt( 803);

```

```

12617: 'GLUT_NORMAL_DAMAGED', 'LongInt').SetInt( 804);
12618: 'GLUT_OVERLAY_DAMAGED', 'LongInt').SetInt( 805);
12619: 'GLUT_VIDEO_RESIZE_POSSIBLE', 'LongInt').SetInt( 900);
12620: 'GLUT_VIDEO_RESIZE_IN_USE', 'LongInt').SetInt( 901);
12621: 'GLUT_VIDEO_RESIZE_X_DELTA', 'LongInt').SetInt( 902);
12622: 'GLUT_VIDEO_RESIZE_Y_DELTA', 'LongInt').SetInt( 903);
12623: 'GLUT_VIDEO_RESIZE_WIDTH_DELTA', 'LongInt').SetInt( 904);
12624: 'GLUT_VIDEO_RESIZE_HEIGHT_DELTA', 'LongInt').SetInt( 905);
12625: 'GLUT_VIDEO_RESIZE_X', 'LongInt').SetInt( 906);
12626: 'GLUT_VIDEO_RESIZE_Y', 'LongInt').SetInt( 907);
12627: 'GLUT_VIDEO_RESIZE_WIDTH', 'LongInt').SetInt( 908);
12628: 'GLUT_VIDEO_RESIZE_HEIGHT', 'LongInt').SetInt( 909);
12629: 'GLUT_ACTIVE_SHIFT', 'LongInt').SetInt( 1);
12630: 'GLUT_ACTIVE_CTRL', 'LongInt').SetInt( 2);
12631: 'GLUT_ACTIVE_ALT', 'LongInt').SetInt( 4);
12632: 'GLUT_CURSOR_RIGHT_ARROW', 'LongInt').SetInt( 0);
12633: 'GLUT_CURSOR_LEFT_ARROW', 'LongInt').SetInt( 1);
12634: 'GLUT_CURSOR_INFO', 'LongInt').SetInt( 2);
12635: 'GLUT_CURSOR_DESTROY', 'LongInt').SetInt( 3);
12636: 'GLUT_CURSOR_HELP', 'LongInt').SetInt( 4);
12637: 'GLUT_CURSOR_CYCLE', 'LongInt').SetInt( 5);
12638: 'GLUT_CURSOR_SPRAY', 'LongInt').SetInt( 6);
12639: 'GLUT_CURSOR_WAIT', 'LongInt').SetInt( 7);
12640: 'GLUT_CURSOR_TEXT', 'LongInt').SetInt( 8);
12641: 'GLUT_CURSOR_CROSSHAIR', 'LongInt').SetInt( 9);
12642: 'GLUT_CURSOR_UP_DOWN', 'LongInt').SetInt( 10);
12643: 'GLUT_CURSOR_LEFT_RIGHT', 'LongInt').SetInt( 11);
12644: 'GLUT_CURSOR_TOP_SIDE', 'LongInt').SetInt( 12);
12645: 'GLUT_CURSOR_BOTTOM_SIDE', 'LongInt').SetInt( 13);
12646: 'GLUT_CURSOR_LEFT_SIDE', 'LongInt').SetInt( 14);
12647: 'GLUT_CURSOR_RIGHT_SIDE', 'LongInt').SetInt( 15);
12648: 'GLUT_CURSOR_TOP_LEFT_CORNER', 'LongInt').SetInt( 16);
12649: 'GLUT_CURSOR_TOP_RIGHT_CORNER', 'LongInt').SetInt( 17);
12650: 'GLUT_CURSOR_BOTTOM_RIGHT_CORNER', 'LongInt').SetInt( 18);
12651: 'GLUT_CURSOR_BOTTOM_LEFT_CORNER', 'LongInt').SetInt( 19);
12652: 'GLUT_CURSOR_INHERIT', 'LongInt').SetInt( 100);
12653: 'GLUT_CURSOR_NONE', 'LongInt').SetInt( 101);
12654: 'GLUT_CURSOR_FULL_CROSSHAIR', 'LongInt').SetInt( 102);
12655: 'GLUT_GAME_MODE_ACTIVE', 'LongInt').SetInt( 0);
12656: 'GLUT_GAME_MODE_POSSIBLE', 'LongInt').SetInt( 1);
12657: 'GLUT_GAME_MODE_WIDTH', 'LongInt').SetInt( 2);
12658: 'GLUT_GAME_MODE_HEIGHT', 'LongInt').SetInt( 3);
12659: 'GLUT_GAME_MODE_PIXEL_DEPTH', 'LongInt').SetInt( 4);
12660: 'GLUT_GAME_MODE_REFRESH_RATE', 'LongInt').SetInt( 5);
12661: 'GLUT_GAME_MODE_DISPLAY_CHANGED', 'LongInt').SetInt( 6);
12662: CL.AddDelphiFunction('Procedure LoadGlut( const dll : String)');
12663: CL.AddDelphiFunction('Procedure FreeGlut');
12664: end;
12665:
12666:
12667: procedure SIRegister_FileClass(CL: TPSPascalCompiler);
12668: begin
12669:   CL.AddTypeS('TFileCreateDisposition', '( fdCreateAlways, fdCreateNew, fdOpenE'
12670:     + 'xisting, fdOpenAlways, fdTruncateExisting )');
12671:   CL.AddTypeS('TFileAccess', '( faRead, faWrite, faReadWrite )');
12672:   CL.AddTypeS('TFileSharing', '( fsNone, fsRead, fsWrite, fsReadWrite )');
12673:   SIRegister_TCustomFile(CL);
12674:   SIRegister_TFile(CL);
12675:   SIRegister_TMemoryFile(CL);
12676:   SIRegister_TTextFileReader(CL);
12677:   SIRegister_TTextFileWriter(CL);
12678:   SIRegister_TFileMapping(CL);
12679:   SIRegister_EFileError(CL);
12680: end;
12681:
12682: procedure SIRegister_FileUtilsClass(CL: TPSPascalCompiler);
12683: begin
12684:   CL.AddTypeS('TFileAttributes', '( ffaReadOnly, ffaHidden, ffaSysFile, ffaVolumeID'
12685:     + ', ffaDirectory, ffaArchive, ffaAnyFile )');
12686:   CL.AddTypeS('TAttributeSet', 'set of TFileAttributes');
12687:   SIRegister_TFileSearch(CL);
12688: end;
12689:
12690: procedure SIRegister_ComPortInterface(CL: TPSPascalCompiler);
12691: begin
12692:   CL.AddConstantN('fttBinary', 'LongWord').SetUInt( $0001);
12693:   'fttParity', 'LongWord').SetUInt( $0002);
12694:   'fttOutxCtsFlow', 'LongWord').SetUInt( $0004);
12695:   'fttOutxDsrFlow', 'LongWord').SetUInt( $0008);
12696:   'fttDtrControlMask', 'LongWord').SetUInt( $0030);
12697:   'fttDsrSensitivity', 'LongWord').SetUInt( $0040);
12698:   'fttTXContinueOnXoff', 'LongWord').SetUInt( $0080);
12699:   'fttOutX', 'LongWord').SetUInt( $0100);
12700:   'fttInX', 'LongWord').SetUInt( $0200);
12701:   'fttErrorChar', 'LongWord').SetUInt( $0400);
12702:   'fttNull', 'LongWord').SetUInt( $0800);
12703:   'fttRtsControlMask', 'LongWord').SetUInt( $3000);
12704:   'fttAbortOnError', 'LongWord').SetUInt( $4000);
12705:   CL.AddTypeS('TTParity', '( parNone, parOdd, parEven, parMark, parSpace )');

```



```

12706: CL.AddTypeS('TTStopBits', '( sbOne, sbOnePointFive, sbTwo )');
12707: CL.AddClassN(CL.FindClass('TOBJECT'),'TComPorts');
12708: CL.AddClassN(CL.FindClass('TOBJECT'),'TComPort');
12709: SIRegister_TComPortInterface(CL);
12710: SIRegister_TComPorts(CL);
12711: SIRegister_TTComPort(CL);
12712: CL.AddClassN(CL.FindClass('TOBJECT'),'EComPortInterface');
12713: 'TOBJECT'),'EComPortAlreadyOpen');
12714: 'TOBJECT'),'EComPortNotOpen');
12715: 'TOBJECT'),'ENoComPortSelected');
12716: 'TOBJECT'),'ECouldNotOpenComPort');
12717: 'TOBJECT'),'ECouldCloseComPort');
12718: 'TOBJECT'),'EInvalidParameter');
12719: 'TOBJECT'),'EReadError');
12720: 'TOBJECT'),'EWriteError');
12721: end;
12722:
12723: procedure SIRegister_SwitchLed(CL: TPSPascalCompiler);
12724: begin
12725:   CL.AddTypeS('TLedColor', '( Aqua, Pink, Purple, Red, Yellow, Green, Blue, Orange, Black )');
12726:   CL.AddTypeS('TLedState', '( LedOn, LedOff, LedDisabled )');
12727:   SIRegister_TSwitchLed(CL);
12728:   //CL.AddDelphiFunction('Procedure Register');
12729: end;
12730:
12731: procedure SIRegister_uColorFunctions(CL: TPSPascalCompiler);
12732: begin
12733:   TRGBType', 'record RedHex : string; GreenHex : string; BlueHex :
12734:   + string; Red : integer; Green : integer; Blue : integer; end');
12735:   Function FadeColor( aColor : Longint; aFade : integer) : Tcolor';
12736:   Function CountColor( aColor : Tcolor; CompareColor : Tcolor; AdjustVale:integer):Tcolor;
12737: end;
12738:
12739: procedure SIRegister_uSettings(CL: TPSPascalCompiler);
12740: begin
12741:   CL.AddDelphiFunction('Procedure SaveOscSettings');
12742:   CL.AddDelphiFunction('Procedure GetOscSettings');
12743: end;
12744:
12745: procedure SIRegister_cyDebug(CL: TPSPascalCompiler);
12746: begin
12747:   TProcProcessEvent', 'Procedure ( Sender : TObject; Index : Integer)';
12748:   RecProcessEvent', 'record Name : ShortString; DurationMs : Cardinal; '
12749:   FirstDurationMs : Cardinal; LastDurationMs : Cardinal; MinDurationMs : Int'
12750:   64; MaxDurationMs : Cardinal; MarksCount : Integer; ArrayMarks : array of '
12751:   Cardinal; EnterCount : Integer; ExitCount : Integer; end');
12752:   SIRegister_TcyDebug(CL);
12753: end;
12754:
12755: procedure SIRegister_cyBaseColorMatrix(CL: TPSPascalCompiler);
12756: begin
12757:   TRGBQuadArray', 'Array[0..0] of TRGBQuad'); //, '^TRGBQuadArray // will not work');
12758:   TProcOnDrawCell', 'Procedure ( Sender: TObject; aRect : TRect; aRow:integer; aCol: integer; aColor :
   TColor)');
12759:   TProcOnCellClick', 'Procedure ( Sender : TObject; aRow : integer'
12760:   +'; aCol : integer; aColor : TColor)');
12761:   SIRegister_TcyBaseColorMatrix(CL);
12762: end;
12763:
12764: procedure SIRegister_cyBaseMeasure(CL: TPSPascalCompiler);
12765: begin
12766:   TBookmarkStyle', '( bsNone, bsArrow, bsTriangle, bsLine, bsRectangle, bsCirle, bsPicture )');
12767:   SIRegister_TBookmarkItem(CL);
12768:   //CL.AddTypeS('TBookmarkItemClass', 'class of TBookmarkItem');
12769:   SIRegister_TBookmarks(CL);
12770:   SIRegister_TLevel(CL);
12771:   //CL.AddTypeS('TLevelClass', 'class of TLevel');
12772:   SIRegister_TLevels(CL);
12773:   CL.AddTypeS('TGraduationStyle', '( gsGuideLine, gsMarks )');
12774:   SIRegister_TGraduation(CL);
12775:   //CL.AddTypeS('TGraduationClass', 'class of tGraduation');
12776:   SIRegister_TGraduations(CL);
12777:   SIRegister_TcyBaseMeasure(CL);
12778: end;
12779:
12780: procedure SIRegister_cySearchFiles(CL: TPSPascalCompiler);
12781: begin
12782:   CL.AddTypeS('TcyFileAttributeMode', '( faYes, faNo, faBoth )');
12783:   SIRegister_TcyFileAttributes(CL);
12784:   SIRegister_TSearchRecInstance(CL);
12785:   TOption', '( soOnlyDirs, soIgnoreAttributes, soIgnoreMaskInclude, soIgnoreMaskExclude )');
12786:   TOptions', 'set of TOption';
12787:   TSearchState', '( ssIdle, ssPaused, ssSearch, ssPausing, ssResuming, ssAborting )');
12788:   TProcOnValidateFileEvent', 'Procedure ( Sender : TObject; ValidM'
12789:   + 'askInclude, ValidMaskExclude, ValidAttributes : boolean; var Accept : boolean)');
12790:   TProcOnValidateDirectoryEvent', 'Procedure ( Sender : TObject; D'
12791:   + 'irectory : String; var Accept : boolean)';
12792:   SIRegister_TcyCustomSearchFiles(CL);
12793:   SIRegister_TcySearchFiles(CL);

```

```

12794:   Function FileNameRespondToMask( aFileName : String; aMask : String) : Boolean');
12795:   Function IscyFolder( aSRec : TSearchrec) : Boolean');
12796: end;
12797:
12798: procedure SIRegister_PJISStreams(CL: TPSPascalCompiler);
12799: begin
12800:   SIRegister_TPJISStreamWrapper(CL);
12801:   SIRegister_TPJHandleIStreamWrapper(CL);
12802:   SIRegister_TPJFileIStream(CL);
12803: end;
12804:
12805: procedure SIRegister_kcMapView( CL: TPSPascalCompiler);
12806: begin
12807:   CL.AddClassN(CL.FindClass('TOBJECT'),'TMapView');
12808:   CL.AddTypeS('TMapSource', '( msNone, msGoogleNormal, msGoogleSatellite, msGoo
12809:   +gleHybrid, msGooglePhysical, msGooglePhysicalHybrid, msOpenStreetMapMapnik'
12810:   +', msOpenStreetMapOsmarender, msOpenCycleMap, msVirtualEarthBing, msVirtual
12811:   +EarthRoad, msVirtualEarthAerial, msVirtualEarthHybrid, msYahooNormal, msYa
12812:   +hooSatellite, msYahooHybrid, msOviNormal, msOviSatellite, msOviHybrid, msOviPhysical )');
12813:   TArea, 'record top : Int64; left : Int64; bottom : Int64; right: Int64; end';
12814:   TRealArea, 'record top : Extended; left : Extended; bottom : Extended; right : Extended; end';
12815:   TIntPoint, 'record X : Int64; Y : Int64; end';
12816:   TkRealPoint, 'record X : Extended; Y : Extended; end';
12817:   TOnBeforeDownloadEvent, 'Procedure ( Url : string; str : TStream; var CanHandle : Boolean)';
12818:   TOnAfterDownloadEvent, 'Procedure ( Url : string; str : TStream)';
12819:   SIRegister_TCustomDownloadEngine(CL);
12820:   SIRegister_TCustomGeolocationEngine(CL);
12821:   SIRegister_TMapView(CL);
12822: end;
12823:
12824: procedure SIRegister_LedNumber(CL: TPSPascalCompiler);
12825: begin
12826:   CL.AddTypeS('TLedSegmentSize', 'Integer');
12827:   TLedNumberBorderStyle, '( lnbNone, lnbSingle, lnbSunken, lnbRaised )';
12828:   SIRegister_TCustomLEDNumber(CL);
12829:   SIRegister_TLEDNumber(CL);
12830: end;
12831:
12832: procedure SIRegister_Sensors(CL: TPSPascalCompiler);
12833: begin
12834:   TStopLights, '( slUNKNOWN, slRED, slYELLOW, slGREEN )';
12835:   SIRegister_TSensorPanel(CL);
12836:   ('TAnalogKind', '( akAnalog, akHorizontal, akVertical )');
12837:   SIRegister_TAnalogSensor(CL);
12838:   SIRegister_TStopLightSensor(CL);
12839: end;
12840:
12841: procedure SIRegister_pwnative_out(CL: TPSPascalCompiler);
12842: begin
12843:   CL.AddConstantN('STDIN','LongInt').SetInt( 0);
12844:   CL.AddConstantN('STDOUT','LongInt').SetInt( 1);
12845:   CL.AddConstantN('STDERR','LongInt').SetInt( 2);
12846: end;
12847:
12848: procedure SIRegister_synwrap1(CL: TPSPascalCompiler);
12849: begin
12850:   TSynwInfo, 'record Err : byte; UrlHtml : ansistring; ErrResponse : integer;
12851:   UltimateURL : ansistring; Headers : ansistring; end';
12852:   TUrlInfo, 'record Err : byte; UltimateURL : string; end';
12853: end;
12854:
12855: procedure SIRegister_HTMLUtil(CL: TPSPascalCompiler);
12856: begin
12857:   CL.AddDelphiFunction('Function GetVal( const tag, attribname_ci : string) : string');
12858: end;
12859:
12860: procedure SIRegister_pwmmain(CL: TPSPascalCompiler);
12861: begin
12862:   CL.AddConstantN('FUTURE_COOKIE','String').SetString( 'Mon, 01 Dec 2099 12:00:00 GMT');
12863:   EXPIRED_COOKIE, 'String'.SetString( 'Mon, 01 Jan 2001 12:00:00 GMT');
12864:   'SECURE_OFF','LongInt').SetInt( 0);
12865:   'SECURE_ON','LongInt').SetInt( 2);
12866:   'SECURE_FILTER','LongInt').SetInt( 3);
12867:   // astr = ansistring;
12868:   CL.AddTypeS('pastr', 'ansistring');
12869:   CL.AddTypeS('TFilterFunc', 'function(const s: pastr): pastr;');
12870:   //type TFilterFunc = function(const s: astr): astr;
12871:   CL.AddTypeS('TWebVar', 'record name : pastr; value : pastr; end');
12872:   CL.AddTypeS('TWebVars', 'array of TWebVar');
12873: end;
12874:
12875: procedure SIRegister_IpAnim(CL: TPSPascalCompiler);
12876: begin
12877:   //CL.AddConstantN('DefaultAggressiveDrawing','Boolean')BoolToStr( False);
12878:   //CL.AddConstantN('DefaultFrameChangeNotify','Boolean')BoolToStr( False);
12879:   TDisposalMethod, '( NODISPOSALMETHOD, DONOTDISPOSE, OVERWRITEWI
12880:   + 'THBKGCOLOR, OVERWRITEWITHPREVIOUS )';
12881:   CL.AddClassN(CL.FindClass('TOBJECT'),'TIpAnimationFrameList');
12882:   CL.AddTypeS('TOnBeforeFrameChange', 'Procedure ( Sender : TObject; CurrentFra

```

```

12883:      + 'me : Integer; var NewFrame : Integer; Frames : TipAnimationFrameList; var CanChange : boolean'))';
12884:      TOnAfterFrameChange', 'Procedure ( Sender : TObject; CurrentFrame: Integer; Frames :
TipAnimationFrameList)');
12885:      CL.AddClassN(CL.FindClass('TObject'), 'EAnimationError');
12886:      CL.AddClassN(CL.FindClass('TObject'), 'EFrameListError');
12887:      SIRegister_TipAnimationFrame(CL);
12888:      SIRegister_TipAnimationFrameList(CL);
12889:      SIRegister_TipAnimatedGraphic(CL);
12890: end;
12891:
12892: procedure SIRegister_IpUtils(CL: TPSPascalCompiler);
12893: begin
12894:   CL.AddConstantN('InternetProfessionalVersion', 'Extended').setExtended( 1.15);
12895:   sShortVersion, 'String').SetString( 'v%.2f');
12896:   CL.AddConstantN('IpMsgBase', 'LongWord').SetUInt( WM_USER + $0E90);
12897:   'CM_IPASYNCRESET', 'LongInt').SetInt( IpMsgBase + 0);
12898:   'CM_IPSOCKMESSAGE', 'LongInt').SetInt( IpMsgBase + 1);
12899:   'CM_IPSOCKETSTATUS', 'LongInt').SetInt( IpMsgBase + 2);
12900:   'CM_IPFREESOCKET', 'LongInt').SetInt( IpMsgBase + 3);
12901:   'CM_IPLINEMESSAGE', 'LongInt').SetInt( IpMsgBase + 4);
12902:   'CM_IPTERMDATA', 'LongInt').SetInt( IpMsgBase + 5);
12903:   'CM_IPTERMRESIZE', 'LongInt').SetInt( IpMsgBase + 6);
12904:   'CM_IPICMPECHO', 'LongInt').SetInt( IpMsgBase + 7);
12905:   'CM_IPHTTGETREQUEST', 'LongInt').SetInt( IpMsgBase + 8);
12906:   'CM_IPTIMESERVER', 'LongInt').SetInt( IpMsgBase + 9);
12907:   'CM_IPTIMECLIENT', 'LongInt').SetInt( IpMsgBase + 10);
12908:   'CM_IPSNTPCLIENT', 'LongInt').SetInt( IpMsgBase + 11);
12909:   'CM_IPFTPPEPLY', 'LongInt').SetInt( IpMsgBase + 12);
12910:   'CM_IPFTPSTATUS', 'LongInt').SetInt( IpMsgBase + 13);
12911:   'CM_IPFTPERROR', 'LongInt').SetInt( IpMsgBase + 14);
12912:   'CM_IPFTPTIMEOUT', 'LongInt').SetInt( IpMsgBase + 15);
12913:   'CM_IPTERMFORCEIZE', 'LongInt').SetInt( IpMsgBase + 16);
12914:   'CM_IPTERMSTUFF', 'LongInt').SetInt( IpMsgBase + 17);
12915:   'CM_IPRASSTATUS', 'LongInt').SetInt( IpMsgBase + 18);
12916:   'CM_IPFINWHOSERVER', 'LongInt').SetInt( IpMsgBase + 19);
12917:   'CM_IPUTILITYSERVER', 'LongInt').SetInt( IpMsgBase + 20);
12918:   'CM_IPSMTPEVENT', 'LongInt').SetInt( IpMsgBase + 21);
12919:   'CM_IPPOP3EVENT', 'LongInt').SetInt( IpMsgBase + 22);
12920:   'CM_IPNNTPEVENT', 'LongInt').SetInt( IpMsgBase + 23);
12921:   'CM_IPHOTINVOKE', 'LongInt').SetInt( IpMsgBase + 24);
12922:   CL.AddTypeS('TipHandle', 'Cardinal');
12923:   TipMD5StateArray', 'array[0..3] of DWORD');
12924:   CL.AddTypeS('TipMD5CountArray', 'array[0..1] of DWORD');
12925:   TipMD5ByteBuf', 'array[0..63] of Byte');
12926:   TipMD5LongBuf', 'array[0..15] of DWORD');
12927:   TipMD5Digest', 'array[0..15] of Byte');
12928:   TipLineTerminator', '( ltNone, ltCR, ltLF, ltCRLF, ltOther)');
12929:   TipMD5Context', 'record State : TipMD5StateArray; Count : TipMD5
+ 'CountArray; ByteBuf : TipMD5ByteBuf; end');
12930:   CL.AddClassN(CL.FindClass('TObject'), 'EIpBaseException');
12931:   CL.AddClassN(CL.FindClass('TObject'), 'EIpAccessException');
12932:   CL.AddClassN(CL.FindClass('TObject'), 'EIpHtmlException');
12933:   SIRegister_TipBaseAccess(CL);
12934:   SIRegister_TipBasePersistent(CL);
12935:   //TipComponentClass', 'class of TipBaseComponent');
12936:   SIRegister_TipBaseComponent(CL);
12937:   SIRegister_TipBaseWinControl(CL);
12938: end;
12939:
12940:
12941: procedure SIRegister_LrtPoTools(CL: TPSPascalCompiler);
12942: begin
12943:   CL.AddTypeS('TPostStyle', '( postStandard, postPropName, postFull)');
12944: end;
12945:
12946: procedure SIRegister_hhAvComp(CL: TPSPascalCompiler);
12947: begin
12948:   CL.AddTypeS('TpsCoursePointer', '( psArrow, psDelta, psPlane, psPtrLine)');
12949:   SIRegister_ThhAvComp(CL);
12950:   // CL.AddDelphiFunction('Procedure Register');
12951: end;
12952:
12953: procedure SIRegister_GPS2(CL: TPSPascalCompiler);
12954: begin
12955:   //CL.AddTypeS('TPrnSets', 'Integer');
12956:   TPrnSet', 'set of byte');
12957:   TGPSUnits', '( Nautical, Statute, Metric)');
12958:   TStrFormat', '( DDMSS, DDMMmm, DDdd)');
12959:   TFloatUnits', '( Deg, Min, Sec, Rad)');
12960:   TgpsdataArray', 'array [0..20] of string;');
12961:   // TdataArray = array [0..20] of string;
12962:   TRecEvent', 'Procedure ( Sender : TObject; SenID : string; A : TgpsdataArray)');
12963:   TFixEvent', 'Procedure ( Sender : TObject; Lat, Lon, SOG, COG : Double)');
12964:   TProgEvent', 'Procedure ( Sender : TObject; Progress : string; FixD : Integer)');
12965:   TQalEvent', 'Procedure ( Sender : TObject; Quality : string; FixQ : Integer)');
12966:   TErrEvent', 'Procedure ( Sender : TObject; ErrMsg : string)');
12967:   SIRegister_TGPS2(CL);
12968:   //CL.AddDelphiFunction('Procedure Register');
12969: end;
12970:

```

```

12971: procedure SIRegister_GPS(CL: TPSPascalCompiler);
12972: begin
12973:   CL.AddConstantN('MAX_SATS', 'LongInt').SetInt( 12);
12974:   CL.AddConstantN('GPMSG_START', 'String').SetString( '$');
12975:   CL.AddConstantN('GPMSG_STOP', 'String').SetString( '*');
12976:   CL.AddConstantN('SEC_BETWEEN_SEG', 'LongInt').SetInt( 5);
12977:   CL.AddTypeS('TSatellite', 'record Identification : Shortint; Elevation : Short
12978:   + 'int; Azimut : Smallint; SignLevel : Smallint; end');
12979:   CL.AddTypeS('TSatellites', 'array[1..12] of TSatellite');
12980:   //TSatellites = array[1..MAX_SATS] of TSatellite;
12981:   TGPSSatEvent = 'Procedure ( Sender : TObject; NbSat, NbSatUse: Shortint; Sats : TSatellites)';
12982:   TGPSSDats = 'record Latitude : Double; Longitude : Double; Height : Double;
12983:   + 'tAboveSea : Double; Speed : Double; UTCtime : TDateTime; Valid : Boolean;
12984:   + 'NbrSats : Shortint; NbrSatsUsed : Shortint; Course : Double; end';
12985:   CL.AddTypeS('TGPSSDatsEvent', 'Procedure ( Sender : TObject; GPSSDats : TGPSSDats)');
12986:   CL.AddTypeS('TmsgGP', '( msgGP, msgGPGGA, msgGPGLL, msgGPGSV, msgGPRMA, msgGPRMC, msgGPZDA )');
12987:   CL.AddTypeS('TSpeedUnit', '( suKilometre, suMile, suNauticalMile )');
12988:   SIRegister_TGPSLink(CL);
12989:   SIRegister_TCustomGPS(CL);
12990:   SIRegister_TGPS(CL);
12991:   SIRegister_TGPStoGPX(CL);
12992:   SIRegister_TGPSSpeed(CL);
12993:   SIRegister_TGPSSatellitesPosition(CL);
12994:   SIRegister_TGPSSatellitesReception(CL);
12995:   SIRegister_TGPSCompass(CL);
12996:   //CL.AddDelphiFunction('Procedure Register( )');
12997: end;
12998:
12999: procedure SIRegister_Spin(CL: TPSPascalCompiler);
13000: begin
13001:   CL.AddConstantN('InitRepeatPause', 'LongInt').SetInt( 400);
13002:   CL.AddConstantN('RepeatPause', 'LongInt').SetInt( 100);
13003:   CL.AddClassN(CL.FindClass('TObject'), 'TTimerSpeedButton');
13004:   SIRegister_TSpinButton(CL);
13005:   SIRegister_TSpinEdit(CL);
13006:   CL.AddTypeS('TTimeBtnStates', '(tbFocusRect, tbAllowTimer )');
13007:   CL.AddTypeS('TTimeBtnState', 'set of TTimeBtnStates');
13008:   SIRegister_TTimerSpeedButton(CL);
13009: end;
13010:
13011:
13012: procedure SIRegister_NMEA(CL: TPSPascalCompiler);
13013: begin
13014:   CL.AddTypeS('NMEADatArray', 'array of string');
13015: end;
13016:
13017: procedure SIRegister_clockExpert(CL: TPSPascalCompiler);
13018: begin
13019:   CL.AddTypeS('TDCFStatus', '( NotOpened, NoSignal, Synchronizing, ReceiveData, TimeAvailable )');
13020:   SIRegister_TExpertClock(CL);
13021:   //CL.AddDelphiFunction('Procedure Register');
13022: end;
13023:
13024: procedure SIRegister_SortUtils(CL: TPSPascalCompiler);
13025: begin
13026:   CL.AddTypeS('SortType1', 'Byte');
13027:   CL.AddTypeS('SortType2', 'Double');
13028:   CL.AddTypeS('SortType3', 'DWord');
13029:   //CL.AddTypeS('PDWordArray', '^DWordArray // will not work');
13030:   CL.AddTypeS('TDataRecord4', 'record Value : Integer; Data : Integer; end');
13031: end;
13032:
13033: procedure SIRegister_BitmapConversion(CL: TPSPascalCompiler);
13034: begin
13035:   // TMatrix3x3 = array[1..3,1..3] of Double;
13036:   // TMatrix4x4 = array[1..4,1..4] of Double;
13037:   CL.AddTypeS('TMatrix3x31', 'array[1..3] of Double');
13038:   CL.AddTypeS('TMatrix3x3', 'array[1..3] of TMatrix3x31');
13039:   CL.AddTypeS('TMatrix4x41', 'array[1..4] of Double');
13040:   CL.AddTypeS('TMatrix4x4', 'array[1..4] of TMatrix4x41');
13041: end;
13042:
13043: procedure SIRegister_ZDBCUtils(CL: TPSPascalCompiler);
13044: begin
13045:   TZSQLType = (zsqlstUnknown, zsqlstBoolean, zsqlstByte, zsqlstShort, zsqlstInteger, zsqlstLong,
13046:   + 'zsqlstFloat, zsqlstDouble, zsqlstBigDecimal, zsqlstString, zsqlstUnicodeString, zsqlstBytes,
13047:   + 'zsqlstDate, zsqlstTime, zsqlstTimestamp, zsqlstDataSet, zsqlstGUID,
13048:   + 'zsqlstAsciiStream, zsqlstUnicodeStream, zsqlstBinaryStream);
13049: end;
13050:
13051: procedure SIRegister_JclTD32(CL: TPSPascalCompiler);
13052: begin
13053:   'Borland32BitSymbolFileSignatureForDelphi', 'LongWord').SetUInt( $39304246);
13054:   CL.AddConstantN('Borland32BitSymbolFileSignatureForBCB', 'LongWord').SetUInt( $41304246);
13055:   //CL.AddTypeS('PJclTD32FileSignature', '^TJclTD32FileSignature // will not work');
13056:   CL.AddTypeS('TJclTD32FileSignature', 'record Signature : DWORD; Offset : DWORD; end');
13057:   'SUBSECTION_TYPE_MODULE', 'LongWord').SetUInt( $120);
13058:   'SUBSECTION_TYPE_TYPES', 'LongWord').SetUInt( $121);
13059:   'SUBSECTION_TYPE_SYMBOLS', 'LongWord').SetUInt( $124);

```



```

13060: 'SUBSECTION_TYPE_ALIGN_SYMBOLS','LongWord').SetUInt( $125);
13061: 'SUBSECTION_TYPE_SOURCE_MODULE','LongWord').SetUInt( $127);
13062: 'SUBSECTION_TYPE_GLOBAL_SYMBOLS','LongWord').SetUInt( $129);
13063: 'SUBSECTION_TYPE_GLOBAL_TYPES','LongWord').SetUInt( $12B);
13064: 'SUBSECTION_TYPE_NAMES','LongWord').SetUInt( $130);
13065: //CL.AddTypeS('PDirectoryEntry', '^TDirectoryEntry // will not work');
13066: CL.AddTypeS('TDirectoryEntry', 'record SubsectionType : Word; ModuleIndex : W
+ 'ord; Offset : DWORD; Size : DWORD; end');
13067: //CL.AddTypeS('PDirectoryHeader', '^TDirectoryHeader // will not work');
13068: //CL.AddTypeS('PSegmentInfo', '^TSegmentInfo // will not work');
13069: CL.AddTypeS('TSegmentInfo', 'record Segment : Word; Flags : Word; Offset : DWORD; Size : DWORD; end');
13070: //CL.AddTypeS('PSegmentInfoArray', '^TSegmentInfoArray // will not work');
13071: //CL.AddTypeS('PModuleInfo', '^TModuleInfo // will not work');
13072: //CL.AddTypeS('PLineMappingEntry', '^TLineMappingEntry // will not work');
13073: CL.AddTypeS('TOffsetPair', 'record StartOffset : DWORD; EndOffset : DWORD; end');
13074: //CL.AddTypeS('POffsetPairArray', '^TOffsetPairArray // will not work');
13075: //CL.AddTypeS('PSourceFileEntry', '^TSourceFileEntry // will not work');
13076: //CL.AddTypeS('PSourceModuleInfo', '^TSourceModuleInfo // will not work');
13077: //CL.AddTypeS('PGlobalTypeInfo', '^TGlobalTypeInfo // will not work');
13078: 'SYMBOL_TYPE_COMPILE','LongWord').SetUInt( $0001);
13079: 'SYMBOL_TYPE_REGISTER','LongWord').SetUInt( $0002);
13080: 'SYMBOL_TYPE_CONST','LongWord').SetUInt( $0003);
13081: 'SYMBOL_TYPE_UDT','LongWord').SetUInt( $0004);
13082: 'SYMBOL_TYPE_SSEARCH','LongWord').SetUInt( $0005);
13083: 'SYMBOL_TYPE_END','LongWord').SetUInt( $0006);
13084: 'SYMBOL_TYPE_SKIP','LongWord').SetUInt( $0007);
13085: 'SYMBOL_TYPE_CVRESERVE','LongWord').SetUInt( $0008);
13086: 'SYMBOL_TYPE_OBJNAME','LongWord').SetUInt( $0009);
13087: 'SYMBOL_TYPE_BPREL16','LongWord').SetUInt( $0100);
13088: 'SYMBOL_TYPE_LDATAL6','LongWord').SetUInt( $0101);
13089: 'SYMBOL_TYPE_GDATA16','LongWord').SetUInt( $0102);
13090: 'SYMBOL_TYPE_PUB16','LongWord').SetUInt( $0103);
13091: 'SYMBOL_TYPE_LPROC16','LongWord').SetUInt( $0104);
13092: 'SYMBOL_TYPE_GPROC16','LongWord').SetUInt( $0105);
13093: 'SYMBOL_TYPE_THUNK16','LongWord').SetUInt( $0106);
13094: 'SYMBOL_TYPE_BLOCK16','LongWord').SetUInt( $0107);
13095: 'SYMBOL_TYPE_WITH16','LongWord').SetUInt( $0108);
13096: 'SYMBOL_TYPE_LABEL16','LongWord').SetUInt( $0109);
13097: 'SYMBOL_TYPE_CEXMODEL16','LongWord').SetUInt( $010A);
13098: 'SYMBOL_TYPE_VFTPATH16','LongWord').SetUInt( $010B);
13099: 'SYMBOL_TYPE_BPREL32','LongWord').SetUInt( $0200);
13100: 'SYMBOL_TYPE_LDATAL32','LongWord').SetUInt( $0201);
13101: 'SYMBOL_TYPE_GDATA32','LongWord').SetUInt( $0202);
13102: 'SYMBOL_TYPE_PUB32','LongWord').SetUInt( $0203);
13103: 'SYMBOL_TYPE_LPROC32','LongWord').SetUInt( $0204);
13104: 'SYMBOL_TYPE_GPROC32','LongWord').SetUInt( $0205);
13105: 'SYMBOL_TYPE_THUNK32','LongWord').SetUInt( $0206);
13106: 'SYMBOL_TYPE_BLOCK32','LongWord').SetUInt( $0207);
13107: 'SYMBOL_TYPE_WITH32','LongWord').SetUInt( $0208);
13108: 'SYMBOL_TYPE_LABEL32','LongWord').SetUInt( $0209);
13109: 'SYMBOL_TYPE_CEXMODEL32','LongWord').SetUInt( $020A);
13110: 'SYMBOL_TYPE_VFTPATH32','LongWord').SetUInt( $020B);
13111: CL.AddTypeS('TSymbolProcInfo', 'record pParent : DWORD; pEnd : DWORD; pNext :
+ ' DWORD; Size : DWORD; DebugStart : DWORD; DebugEnd : DWORD; Offset : DWORD
+ ' ; Segment : Word; ProcType : DWORD; NearFar : Byte; Reserved : Byte; NameIndex : DWORD; end');
13112: //CL.AddTypeS('PSymbolInfo', '^TSymbolInfo // will not work');
13113: CL.AddTypeS('TSymbolInfo', 'record Size : Word; SymbolType : Word; end');
13114: //CL.AddTypeS('PSymbolInfos', '^TSymbolInfos // will not work');
13115: SIRegister_TJclModuleInfo(CL);
13116: SIRegister_TJclLineInfo(CL);
13117: SIRegister_TJclSourceModuleInfo(CL);
13118: SIRegister_TJclSymbolInfo(CL);
13119: SIRegister_TJclProcSymbolInfo(CL);
13120: CL.AddClassN(CL.FindClass('TOBJECT'),'TJclLocalProcSymbolInfo');
13121: CL.AddClassN(CL.FindClass('TOBJECT'),'TJclGlobalProcSymbolInfo');
13122: SIRegister_TJclTD32InfoParser(CL);
13123: SIRegister_TJclTD32InfoScanner(CL);
13124: SIRegister_TJclPeBorTD32Image(CL);
13125: end;
13126:
13127: procedure SIRegister_JvIni(CL: TPSPascalCompiler);
13128: begin
13129:   CL.AddTypeS('TReadObjectEvent', 'Function ( Sender : TObject; const Section,
+ 'Item, Value : string) : TObject');
13130:   TWriteObjectEvent, 'Procedure ( Sender : TObject; const Section, Item: string; Obj: TObject)';
13131:   CL.AddConstantN('idnListItem','String').SetString('Item');
13132: end;
13133:
13134: procedure SIRegister_StExpr(CL: TPSPascalCompiler);
13135: begin
13136:   //CL.AddTypeS('PStFloat', '^TStFloat // will not work');
13137:   CL.AddTypeS('TStMethod0Param', 'Function : TStFloat');
13138:   CL.AddTypeS('TStMethod1Param', 'Function ( Value1 : TStFloat) : TStFloat');
13139:   CL.AddTypeS('TStMethod2Param', 'Function ( Value1, Value2 : TStFloat) : TStFloat');
13140:   CL.AddTypeS('TStMethod3Param', 'Function ( Value1, Value2, Value3 : TStFloat) : TStFloat');
13141:   CL.AddTypeS('TStGetIdentValueEvent', 'Procedure ( Sender : TObject; const Ide
+ 'ntifier : AnsiString; var Value : TStFloat)');
13142:   CL.AddTypeS('TStToken', '( ssStart, ssInIdent, ssInNum, ssInSign, ssInExp, ss
+ 'Eol, ssNum, ssIdent, ssLPar, ssRPar, ssComma, ssPlus, ssMinus, ssTimes, ssDiv, ssEqual, ssPower )');

```

```

13149: SIRegister_TStExpression(CL);
13150: CL.AddTypeS('TStExprErrorEvent', 'Procedure ( Sender : TObject; ErrorNumber : '
13151:   + ' LongInt; const ErrorStr : AnsiString)');
13152: SIRegister_TStExpressionEdit(CL);
13153: Function AnalyzeExpr( const Expr : AnsiString) : Double);
13154: Procedure TpVal( const S : AnsiString; var V : Extended; var Code : Integer)');
13155: end;
13156:
13157: procedure SIRegister_GR32_Containers(CL: TPSPascalCompiler);
13158: begin
13159:   CL.AddConstantN('BUCKET_MASK', 'LongWord').SetUInt( $FF);
13160:   CL.AddConstantN('BUCKET_COUNT', 'LongInt').SetInt( BUCKET_MASK + 1);
13161:   //CL.AddTypeS('PItem', '^PItem // will not work');
13162:   CL.AddTypeS('PItem', 'TObject');
13163:   //CL.AddTypeS('PData', '^PData // will not work');
13164:   CL.AddTypeS('PData', 'TObject');
13165:   CL.AddTypeS('TTypeKind', '(tkUnknown, tkInteger, tkChar, tkEnumeration, tkFloat, '
13166:     + 'tkString, tkSet, tkClass, tkMethod, tkWChar, tkLString, tkWString, '
13167:     + 'tkVariant, tkArray, tkRecord, tkInterface, tkInt64, tkDynArray)');
13168:   {TTypeKind = (tkUnknown, tkInteger, tkChar, tkEnumeration, tkFloat,
13169:     tkString, tkSet, tkClass, tkMethod, tkWChar, tkLString, tkWString,
13170:     tkVariant, tkArray, tkRecord, tkInterface, tkInt64, tkDynArray);}
13171:   CL.AddTypeS('TTypeKinds', 'set of TTypeKind');
13172:   //TTypeKinds = set of TTypeKind;
13173:   CL.AddTypeS('TOrdType', '(otSByte, otUByte, otSWord, otUWord, otSLong, otULong)');
13174:   // TOrdType = (otSByte, otUByte, otSWord, otUWord, otSLong, otULong);
13175:   //CL.AddTypeS('PPointerBucketItem', '^TPointerBucketItem // will not work');
13176:   CL.AddTypeS('TPointerBucketItem', 'record Item : PItem; Data : PData; end');
13177:   CL.AddTypeS('TPointerBucketItemArray', 'array of TPointerBucketItem');
13178:   CL.AddTypeS('TPointerBucket', 'record Count : Integer; Items : TPointerBucketItemArray; end');
13179:   SIRegister_TPointerMap(CL);
13180:   SIRegister_TPointerMapIterator(CL);
13181:   //CL.AddTypeS('PPolyRects', '^TPolyRects // will not work');
13182:   SIRegister_TRectList(CL);
13183:   //SIRegister_TClassList(CL);
13184:   //CL.AddTypeS('PLinkedNode', '^TLinkedNode // will not work');
13185:   CL.AddTypeS('TLinkedNode', 'record Prev : TObject; Next : TObject; Data : TObject; end');
13186:   CL.AddTypeS('TFreeDataEvent', 'Procedure ( Data : TObject)');
13187:   SIRegister_TLinkedList(CL);
13188:   CL.AddDelphiFunction('Procedure SmartAssign( Src, Dst : TPersistent; TypeKinds : TTypeKinds)');
13189:   CL.AddDelphiFunction('Procedure Advance( var Node : TLinkedNode; Steps : Integer)');
13190: end;
13191:
13192: procedure SIRegister_JclParseUses(CL: TPSPascalCompiler);
13193: begin
13194:   CL.AddClassN(CL.FindClass('TOBJECT'), 'EUsesListError');
13195:   SIRegister_TUsesList(CL);
13196:   SIRegister_TCustomGoal(CL);
13197:   SIRegister_TProgramGoal(CL);
13198:   SIRegister_TLibraryGoal(CL);
13199:   SIRegister_TUnitGoal(CL);
13200: end;
13201:
13202: procedure SIRegister_JvFinalize(CL: TPSPascalCompiler);
13203: begin
13204:   //type
13205:   // TFinalizeProc = procedure;
13206:   CL.AddTypeS('TFinalizeProc', 'procedure');
13207: end;
13208:
13209:
13210: procedure SIRegister_uPSUtils(CL: TPSPascalCompiler);
13211: begin
13212:   PSMainProcName, 'String').SetString( '!MAIN');
13213:   PSMainProcNameOrg, 'String').SetString( 'Main Proc');
13214:   'PSLowBuildSupport', 'LongInt').SetInt( 12);
13215:   CL.AddConstantN('PSCurrentBuildNo', 'LongInt').SetInt( 23);
13216:   'PSCurrentVersion', 'String').SetString( '1.31');
13217:   'PSValidHeader', 'LongInt').SetInt( 1397769801);
13218:   'PSAddrStackStart', 'LongInt').SetInt( 1610612736);
13219:   'PSAddrNegativeStackStart', 'LongInt').SetInt( 1073741824);
13220:   //TPSBaseType, 'Byte);
13221:   'btReturnAddress', 'LongInt').SetInt( 0);
13222:   'btU8', 'LongInt').SetInt( 1);
13223:   'btS8', 'LongInt').SetInt( 2);
13224:   'btU16', 'LongInt').SetInt( 3);
13225:   'btS16', 'LongInt').SetInt( 4);
13226:   'btU32', 'LongInt').SetInt( 5);
13227:   'btS32', 'LongInt').SetInt( 6);
13228:   'btSingle', 'LongInt').SetInt( 7);
13229:   'btDouble', 'LongInt').SetInt( 8);
13230:   'btExtended', 'LongInt').SetInt( 9);
13231:   'btString', 'LongInt').SetInt( 10);
13232:   'btRecord', 'LongInt').SetInt( 11);
13233:   'btArray', 'LongInt').SetInt( 12);
13234:   'btPointer', 'LongInt').SetInt( 13);
13235:   'btPChar', 'LongInt').SetInt( 14);
13236:   'btResourcePointer', 'LongInt').SetInt( 15);
13237:   'btVariant', 'LongInt').SetInt( 16);

```

```

13238: 'btS64','LongInt').SetInt( 17);
13239: 'btU64','LongInt').SetInt( 30);
13240: 'btChar','LongInt').SetInt( 18);
13241: 'btWideString','LongInt').SetInt( 19);
13242: 'btWideChar','LongInt').SetInt( 20);
13243: 'btProcPtr','LongInt').SetInt( 21);
13244: 'btStaticArray','LongInt').SetInt( 22);
13245: 'btSet','LongInt').SetInt( 23);
13246: 'btCurrency','LongInt').SetInt( 24);
13247: 'btClass','LongInt').SetInt( 25);
13248: 'btInterface','LongInt').SetInt( 26);
13249: 'btNotificationVariant','LongInt').SetInt( 27);
13250: 'btUnicodeString','LongInt').SetInt( 28);
13251: 'btType','LongInt').SetInt( 130);
13252: 'btEnum','LongInt').SetInt( 129);
13253: 'btExtClass','LongInt').SetInt( 131);
13254: // CL.AddDelphiFunction('Function MakeHash( const s : TbtString) : Longint');
13255: 'CM_A','LongInt').SetInt( 0);
13256: 'CM_CA','LongInt').SetInt( 1);
13257: 'CM_P','LongInt').SetInt( 2);
13258: 'CM_PV','LongInt').SetInt( 3);
13259: 'CM_PO','LongInt').SetInt( 4);
13260: 'Cm_C','LongInt').SetInt( 5);
13261: 'Cm_G','LongInt').SetInt( 6);
13262: 'Cm_CG','LongInt').SetInt( 7);
13263: 'Cm_CNG','LongInt').SetInt( 8);
13264: 'Cm_R','LongInt').SetInt( 9);
13265: 'Cm_ST','LongInt').SetInt( 10);
13266: 'Cm_Pt','LongInt').SetInt( 11);
13267: 'CM_CO','LongInt').SetInt( 12);
13268: 'Cm_cv','LongInt').SetInt( 13);
13269: 'cm_sp','LongInt').SetInt( 14);
13270: 'cm_bn','LongInt').SetInt( 15);
13271: 'cm_vm','LongInt').SetInt( 16);
13272: 'cm_sf','LongInt').SetInt( 17);
13273: 'cm_fg','LongInt').SetInt( 18);
13274: 'cm_puexh','LongInt').SetInt( 19);
13275: 'cm_poexh','LongInt').SetInt( 20);
13276: 'cm_in','LongInt').SetInt( 21);
13277: 'cm_spc','LongInt').SetInt( 22);
13278: 'cm_inc','LongInt').SetInt( 23);
13279: 'cm_dec','LongInt').SetInt( 24);
13280: 'cm_nop','LongInt').SetInt( 255);
13281: 'Cm_PG','LongInt').SetInt( 25);
13282: 'Cm_P2G','LongInt').SetInt( 26);
13283: CL.AddTypeS('TbtU8', 'Byte');
13284: CL.AddTypeS('TbtS8', 'ShortInt');
13285: CL.AddTypeS('TbtU16', 'Word');
13286: CL.AddTypeS('TbtS16', 'SmallInt');
13287: CL.AddTypeS('TbtU32', 'Cardinal');
13288: CL.AddTypeS('TbtS32', 'Longint');
13289: CL.AddTypeS('TbtSingle', 'Single');
13290: CL.AddTypeS('TbtDouble', 'double');
13291: CL.AddTypeS('TbtExtended', 'Extended');
13292: CL.AddTypeS('tbtCurrency', 'Currency');
13293: CL.AddTypeS('tbts64', 'int64');
13294: CL.AddTypeS('Tbtu64', 'uint64');
13295: CL.AddTypeS('TbtString', 'string');
13296: CL.AddDelphiFunction('Function MakeHash( const s : TbtString) : Longint');
13297: // TbtString = {$IFDEF DELPHI2009UP}AnsiString{$ELSE}String{$ENDIF};
13298: // 'PointerSize','LongInt').SetInt( IPointer ( 8 4 ));
13299: 'PSMaxListSize','LongInt').SetInt( Maxint div 16);
13300: //CL.AddTypeS('PPointerList', '^TPointerList // will not work');
13301: SIRegister_TPSPList(CL);
13302: CL.AddTypeS('TIPLList', 'TPSPList');
13303: SIRegister_TPSPStringList(CL);
13304: CL.AddTypeS('TIFStringList', 'TPSPStringList');
13305: CL.AddClassN(CL.FindClass('TOBJECT'),'TPSUnitList');
13306: SIRegister_TPSUnit(CL);
13307: SIRegister_TPSPUnitList(CL);
13308: CL.AddTypeS('TPSParserErrorKind', '( iNoError, iCommentError, iStringError, iCharError, iSyntaxError )');
13309: CL.AddTypeS('TPSParserErrorEvent', 'Procedure ( Parser : TObject; Kind : TPSParserErrorKind)');
13310: SIRegister_TPSPascalParser(CL);
13311: CL.AddDelphiFunction('Function PSFloatToStr( E : Extended) : TbtString');
13312: Function FastLowerCase( const s : TbtString) : TbtString;
13313: Function Fw( const S : TbtString) : TbtString;
13314: Function FirstWord( const S : TbtString) : TbtString;
13315: CL.AddDelphiFunction('Function PSIntToStr( I : LongInt) : TbtString');
13316: Function PSStrToIntDef( const S : TbtString; Def : LongInt) : LongInt;
13317: Function PSStrToInt( const S : TbtString) : LongInt;
13318: Function PSStrToFloat( const s : TbtString) : Extended;
13319: Function FastUpperCase( const s : TbtString) : TbtString;
13320: Function GRFW( var s : TbtString) : TbtString;
13321: Function GRLW( var s : TbtString) : TbtString;
13322: 'FCapacityInc','LongInt').SetInt( 32);
13323: Function PSWideUpperCase( const S : WideString) : WideString;
13324: Function PSWideLowerCase( const S : WideString) : WideString;
13325: Procedure RaiseError( const errmsg : string; row, col : integer);
13326: Function GetLicence : string;

```

```

13327: Function GetUsedUnitList( list : TStringlist ) : string);
13328: Function GetTokenName( TokenID : TPSPasToken ) : string);
13329: 'NewLine2','String').SetString( #13#10);
13330: end;
13331:
13332: procedure SIRegister_TypInfo(CL: TPSPascalCompiler);
13333: begin
13334:   CL.AddTypeS('TTypeKind', '( tkUnknown, tkInteger, tkChar, tkEnumeration, tkFl'
13335:   + 'oat, tkString, tkSet, tkClass, tkMethod, tkWChar, tkLString, tkWString, tk'
13336:   + 'Variant, tkArray, tkRecord, tkInterface, tkInt64, tkDynArray )');
13337:   { TTypeKind = (tkUnknown, tkInteger, tkChar, tkEnumeration, tkFloat,
13338:   tkString, tkSet, tkClass, tkMethod, tkWChar, tkLString, tkWString,
13339:   tkVariant, tkArray, tkRecord, tkInterface, tkInt64, tkDynArray);}
13340:   CL.AddTypeS('TTypeKinds', 'set of TTypeKind');
13341:   CL.AddTypeS('TOrdType', '( otSByte, otUByte, otSWord, otUWord, otSLong, otULong )');
13342:   CL.AddTypeS('TFloatType', '( ftSingle, ftDouble, ftExtended, ftComp, ftCurr )');
13343:   CL.AddTypeS('TMethodKind', '( mkProcedure, mkFunction, mkConstructor, mkDestr'
13344:   + 'uctor, mkClassProcedure, mkClassFunction, mkClassConstructor, mkOperatorOv'
13345:   + 'erload, mkSafeProcedure, mkSafeFunction )');
13346:   CL.AddTypeS('TTypeInfo', 'record Kind : TTypeKind; Name : ShortString; end');
13347:   Function SamePropTypeName( const Name1, Name2 : ShortString) : Boolean);
13348:   Function FloatToStrEx( Value : Extended) : string);
13349:   Function StrToFloatEx( const S : string) : Extended);
13350: end;
13351:
13352: procedure SIRegister_ParserU(CL: TPSPascalCompiler);
13353: begin
13354:   CL.AddTypeS('TProcAttrs', '(PublicProc, IsDone, IsHelper )');
13355:   CL.AddTypeS('TProcAttr', 'set of TProcAttrs');
13356:   SIRegister_TProcList(CL);
13357:   CL.AddTypeS('TWriteln', 'Procedure ( const S : string)');
13358:   CL.AddTypeS('TReadln', 'Procedure ( var S : string; const Promote, Caption : string)');
13359:   CL.AddTypeS('TPasToken', 'record ID: TPSPasToken; Data:string; Org:String; row:integer; col:integer;
end');
13360:   CL.AddTypeS('TTimeMode', '(CompileTime, RunTime )');
13361:   SIRegister_TUnitParser(CL);
13362:   CL.AddConstantN('MaxSearchCount','LongInt').SetInt( 100);
13363: end;
13364:
13365: procedure SIRegister_ServiceMgr(CL: TPSPascalCompiler);
13366: begin
13367:   CL.AddConstantN('SERVICE_KERNEL_DRIVER','LongWord').SetUInt( $00000001);
13368:   CL.AddConstantN('SERVICE_FILE_SYSTEM_DRIVER','LongWord').SetUInt( $00000002);
13369:   CL.AddConstantN('SERVICE_ADAPTER','LongWord').SetUInt( $00000004);
13370:   CL.AddConstantN('SERVICE_RECOGNIZER_DRIVER','LongWord').SetUInt( $00000008);
13371:   CL.AddConstantN('SERVICE_WIN32_OWN_PROCESS','LongWord').SetUInt( $00000010);
13372:   CL.AddConstantN('SERVICE_WIN32_SHARE_PROCESS','LongWord').SetUInt( $00000020);
13373:   CL.AddConstantN('SERVICE_INTERACTIVE_PROCESS','LongWord').SetUInt( $00000100);
13374:   //CL.AddTypeS('PTStrings', '^TStrings // will not work');
13375:   SIRegister_TServiceManager(CL);
13376: end;
13377:
13378: procedure SIRegister_ubigFloatV3(CL: TPSPascalCompiler);
13379: begin
13380:   CL.AddTypeS('TMaxSig', 'integer');
13381:   CL.AddTypeS('TView', '( normal, Scientific )');
13382:   SIRegister_TFloatInt(CL); SIRegister_TBigFloat(CL);
13383: end;
13384:
13385: procedure SIRegister_UBigIntsV4(CL: TPSPascalCompiler);
13386: begin
13387:   TDigits, 'array of int64'); SIRegister_TInteger(CL);
13388:   Procedure SetBaseVal( const newbase : integer);
13389:   Function GetBasePower : integer);
13390:   Function GetBase : integer);
13391:   Procedure SetThreadSafe( newval : boolean);
13392: end;
13393:
13394: procedure SIRegister_TInteger(CL: TPSPascalCompiler);
13395: begin
13396:   //with RegClassS(CL,'TObject', 'TInteger') do
13397:   with CL.AddClassN(CL.FindClass('TObject'),'TInteger') do begin
13398:     Constructor Create( const initialValue : int64);
13399:     RegisterMethod('Constructor Createl');
13400:     Procedure Free);
13401:     RegisterProperty('Digits', 'TDigits', iptr);
13402:     Procedure Assign( const I2 : TInteger););
13403:     Procedure Assign1( const I2 : int64););
13404:     Procedure Assign2( const I2 : string););
13405:     Procedure AbsoluteValue);
13406:     Procedure Add( const I2 : TInteger););
13407:     Procedure Add1( const I2 : int64););
13408:     Procedure AssignZero);
13409:     Procedure AssignOne);
13410:     Procedure Subtract( const I2 : TInteger););
13411:     Procedure Subtract1( const I2 : int64););
13412:     Procedure Mult( const I2 : TInteger););
13413:     Procedure Mult1( const I2 : int64););
13414:     Procedure FastMult( const I2 : TInteger););

```



```

13415: Procedure Divide( const I2 : TInteger););
13416: Procedure Dividel( const I2 : int64););
13417: Procedure Modulo( const I2 : TInteger););
13418: Procedure Modulo1( const N : int64););
13419: Procedure ModPow( const I2, m : TInteger););
13420: Procedure InvMod( I2 : TInteger););
13421: Procedure DivideRem( const I2 : TInteger; var remain : TInteger););
13422: Procedure DivideRemTrunc( const I2 : TInteger; var remain : TInteger););
13423: Procedure DivideRemFloor( const I2 : TInteger; var remain : TInteger););
13424: Procedure DivideRemEuclidean( const I2 : TInteger; var remain : TInteger););
13425: Function Compare( I2 : TInteger) : integer;);
13426: Function Compare1( I2 : int64) : integer;);
13427: Procedure Factorial;);
13428: Function ConvertToDecimalString( commas : boolean) : string;);
13429: Function ToString( commas : boolean) : string;);
13430: Function ConvertToInt64( var N : int64) : boolean;);
13431: Function DigitCount : integer;);
13432: Procedure SetSign( s : integer););
13433: Function GetSign : integer;);
13434: Function IsOdd : boolean;);
13435: Function IsPositive : boolean;);
13436: Function IsNegative : boolean;);
13437: Function IsProbablyPrime : boolean;);
13438: Function IsZero : boolean;);
13439: Procedure ChangeSign;);
13440: Procedure Pow( const exponent : int64););
13441: Procedure Sqrt;);
13442: Procedure Square;);
13443: Procedure FastSquare;);
13444: Procedure Gcd( const I2 : TInteger););
13445: Procedure Gcd1( const I2 : int64););
13446: Procedure NRoot( const Root : int64););
13447: Function GetBase : integer;);
13448: Function BitCount : integer;);
13449: Function ConvertToHexString : String;);
13450: Function AssignRandomPrime( BitLength : integer; seed : String; mustMatchBitLength : boolean) :
boolean;);
13451: Function AssignHex( HexStr : String) : boolean;);
13452: Procedure RandomOfSize( size : integer););
13453: Procedure Random( maxint : TInteger););
13454: Procedure Getnextprime;);
13455: end;
13456: end;
13457:
13458: procedure SIRegister_TBigFloat(CL: TPSPascalCompiler);
13459: begin
13460:   //with RegClassS(CL,'TObject', 'TBigFloat') do
13461:   with CL.AddClassN(CL.FindClass('TObject'),'TBigFloat') do begin
13462:     RegisterProperty('decpart', 'TFloatInt', iptrw);
13463:     RegisterProperty('sigdigits', 'word', iptrw);
13464:     RegisterProperty('exponent', 'integer', iptrw);
13465:     RegisterMethod('Constructor Create'););
13466:     Constructor Create1( const MaxSig : TMaxSig););
13467:     Procedure Assign( A : TBigFloat););
13468:     Procedure Assign3( A : TBigFloat; SigDig : word););
13469:     Procedure Assign4( A : TInteger););
13470:     Procedure Assign5( N : int64););
13471:     Procedure Assign6( N : int64; SigDig : integer););
13472:     Procedure Assign7( d : extended););
13473:     Procedure Assign8( S : string););
13474:     Procedure Assign9( S : string; SigDig : word););
13475:     Procedure AssignZero;);
13476:     Procedure AssignOne;);
13477:     Procedure Add( B : TBigFloat););
13478:     Procedure Add1( B : int64););
13479:     Procedure AbsAdd( B : TBigFloat););
13480:     Procedure Subtract( B : TBigFloat););
13481:     Procedure Subtract1( B : int64););
13482:     Procedure Mult( B : TBigFloat););
13483:     Procedure Mult1( B : TBigfloat; const MaxSig : TMaxSig););
13484:     Procedure Mult2( B : TInteger););
13485:     Procedure Mult3( B : int64););
13486:     Procedure MultRaw( B : TBigFloat););
13487:     Procedure Reciprocal( const MaxSig : TMaxSig););
13488:     Procedure Divide( B : TBigFloat; const MaxSig : TMaxSig););
13489:     Procedure Dividel( B : TInteger; const MaxSig : TMaxSig););
13490:     Procedure Divide2( B : int64; const MaxSig : TMaxSig););
13491:     Procedure Square( const MaxSig : TMaxSig););
13492:     Function Compare( B : TBigFloat) : integer;);
13493:     Function IsZero : boolean;);
13494:     Procedure MaxBigFloat( B : TBigFloat););
13495:     Procedure MinBigFloat( B : TBigFloat););
13496:     Procedure Sqrt;);
13497:     Procedure Sqrt1( const MaxSig : TMaxSig););
13498:     Procedure NRoot( N : integer; const MaxSig : TMaxSig););
13499:     Procedure IntPower( intpower : integer; const MaxSig : TMaxSig););
13500:     Procedure Power( power : TBigfloat; const MaxSig : TMaxSig););
13501:     Procedure Log( const MaxSig : TMaxSig););
13502:     Procedure Log10( const MaxSig : TMaxSig););

```

```

13503:   Procedure Exp( const MaxSig : TMaxSig);
13504:   Procedure PiConst( const MaxSig : TMaxSig);
13505:   Procedure Log2Const( const MaxSig : TMaxSig);
13506:   Procedure RoundToPrec( const MaxSig : TMaxSig);
13507:   Procedure RoundToPrec1;
13508:   Procedure Trunc( const x : integer);
13509:   Procedure Floor( const x : integer);
13510:   Procedure Ceiling( const x : integer);
13511:   Procedure Round( const x : integer);
13512:   Procedure AbsoluteValue;
13513:   Procedure Negate;
13514:   Procedure SetSigDigits( const newsigdigits : integer);
13515:   Function ConvertToString( const View : TView) : string;
13516:   Function ToString( const View : TView) : string;
13517:   Function ConvertToExtended( var num : extended) : boolean;
13518:   Function ConvertToInt64( var N : int64) : boolean;
13519:   Function IntPart : int64;
13520: end;
13521: end;
13522:
13523: procedure SIRegister_ModbusTypes(CL: TPSPascalCompiler);
13524: begin
13525:   CL.AddTypeS('TModBusFunction', 'Byte');
13526:   TModBusHeader = record TransactionID: Word; ProtocolID: Word;
13527:     RecLength: Word; UnitID: Byte; end;
13528:   TModBusRequestBuffer = record Header: TModBusHeader; FunctionC
13529:     ode: TModBusFunction; MBPData: TModBusDataBuffer; end;
13530:   TModBusResponseBuffer = record Header: TModBusHeader; Function
13531:     Code: TModBusFunction; MBPData: TModBusDataBuffer; end;
13532:   TModBusExceptionBuffer = record Header: TModBusHeader; Excepti
13533:     +onFunction: TModBusFunction; ExceptionCode: Byte; end;
13534:   type
13535:     TModBusFunction = Byte;
13536:
13537:   type
13538:     TModBusDataBuffer = array[0..260] of Byte;
13539:   type
13540:     TModBusHeader = packed record
13541:       TransactionID: Word;
13542:       ProtocolID: Word;
13543:       RecLength: Word;
13544:       UnitID: Byte;
13545:     end;
13546:   type
13547:     TModBusRequestBuffer = packed record
13548:       Header: TModBusHeader;
13549:       FunctionCode: TModBusFunction;
13550:       MBPData: TModBusDataBuffer;
13551:     end;
13552:   type
13553:     TModBusResponseBuffer = packed record
13554:       Header: TModBusHeader;
13555:       FunctionCode: TModBusFunction;
13556:       MBPData: TModBusDataBuffer;
13557:     end;
13558:   type
13559:     TModBusExceptionBuffer = packed record
13560:       Header: TModBusHeader;
13561:       ExceptionFunction: TModBusFunction;
13562:       ExceptionCode: Byte;
13563:     end;
13564: end;
13565:
13566: unit ModbusConsts;
13567:
13568: interface
13569:
13570: const
13571:   MB_PORT = 502;
13572:   MB_IGNORE_UNITID = 255;
13573:   MB_PROTOCOL = 0;
13574:
13575: // Define constants for the ModBus functions
13576: const
13577:   mbfReadCoils = $01;
13578:   mbfReadInputBits = $02;
13579:   mbfReadHoldingRegs = $03;
13580:   mbfReadInputRegs = $04;
13581:   mbfWriteOneCoil = $05;
13582:   mbfWriteOneReg = $06;
13583:   mbfWriteCoils = $0F;
13584:   mbfWriteRegs = $10;
13585:   mbfReadFileRecord = $14;
13586:   mbfWriteFileRecord = $15;
13587:   mbfMaskWriteReg = $16;
13588:   mbfReadWriteRegs = $17;
13589:   mbfReadFiFoQueue = $18;
13590:
13591: // Define constants for the ModBus exceptions

```

```

13592: const
13593:   mbeOk = $00;
13594:   mbeIllegalFunction = $01;
13595:   mbeIllegalRegister = $02;
13596:   mbeIllegalDataValue = $03;
13597:   mbeServerFailure = $04;
13598:   mbeAcknowledge = $05;
13599:   mbeServerBusy = $06;
13600:   mbeGatewayPathNotAvailable = $0A;
13601:   mbeGatewayNoResponseFromTarget = $0B;
13602:   MaxBlockLength = 125;
13603:   MaxCoils = 2000;
13604:
13605: const
13606:   DMB_VERSION = '1.6.3'; {Do not Localize}
13607:
13608:
13609: (*-----*)
13610: procedure SIRegister_IdModBusClient(CL: TPSPascalCompiler);
13611: begin
13612:   CL.AddTypeS('TModBusClientErrorEvent', 'Procedure ( const FunctionCode : Byte'
13613:   +'; const ErrorCode : Byte; const ResponseBuffer : TModBusResponseBuffer)');
13614:   CL.AddTypeS('TModBusClientResponseMismatchEvent', 'Procedure ( const RequestF'
13615:   +unctionCode:Byte; const ResponseFunctionCode: Byte; const ResponseBuffer: TModBusResponseBuffer)');
13616:   SIRegister_TIdModBusClient(CL);
13617: end;
13618:
13619: procedure SIRegister_ActnMan(CL: TPSPascalCompiler);
13620: begin
13621:   CL.AddConstantN('CM_ITEMSELECTED', 'LongWord').SetUInt( CM_BASE + $0402);
13622:   CM_ITEMCLICKED, 'LongWord').SetUInt( CM_BASE + $0403);
13623:   CM_ITEMKEYED, 'LongWord').SetUInt( CM_BASE + $0404);
13624:   CM_SCROLLCLICKED, 'LongWord').SetUInt( CM_BASE + $0406);
13625:   CM_RESETBAR, 'LongWord').SetUInt( CM_BASE + $0409);
13626:   CM_ENTERMENULOOOP, 'LongWord').SetUInt( CM_BASE + $0410);
13627:   CM_ITEMDROPPPOINT, 'LongWord').SetUInt( CM_BASE + $0411);
13628:   CL.AddClassN(CL.FindClass('TOBJECT'), 'TActionBars');
13629:   CL.AddClassN(CL.FindClass('TOBJECT'), 'TActionListCollection');
13630:   CL.AddClassN(CL.FindClass('TOBJECT'), 'TActionClientsCollection');
13631:   CL.AddClassN(CL.FindClass('TOBJECT'), 'TActionClientItem');
13632:   //CL.AddTypeS('TActionClientItemClass', 'class of TActionClientItem');
13633:   CL.AddClassN(CL.FindClass('TOBJECT'), 'TCustomActionBar');
13634:   CL.AddClassN(CL.FindClass('TOBJECT'), 'TActionClient');
13635:   CL.AddClassN(CL.FindClass('TOBJECT'), 'TActionClients');
13636:   //CL.AddTypeS('TActionClientsClass', 'class of TActionClients');
13637:   //CL.AddTypeS('TActionBarsClass', 'class of TActionBars');
13638:   CL.AddClassN(CL.FindClass('TOBJECT'), 'TActionBarItem');
13639:   //CL.AddTypeS('TActionBarItemClass', 'class of TActionBarItem');
13640:   CL.AddClassN(CL.FindClass('TOBJECT'), 'TCustomActionManager');
13641:   //CL.AddTypeS('TGetBarsClassEvent', 'Procedure ( Sender : TCustomActionManager;'
13642:   // + ' BarsClass : TActionBarsClass)');
13643:   {CL.AddTypeS('TGetBarItemClassEvent', 'Procedure ( Sender : TCustomActionManag'
13644:   + 'er; var BarItemClass : TActionBarItemClass)');
13645:   CL.AddTypeS('TGetClientsClassEvent', 'Procedure ( Sender : TCustomActionManag'
13646:   + 'er; var ClientsClass : TActionClientsClass)');
13647:   CL.AddTypeS('TGetClientItemClassEvent', 'Procedure ( Sender : TCustomActionMa'
13648:   + 'nager; var ClientItemClass : TActionClientItemClass)');}
13649:   CL.AddTypeS('TStyleChanged', 'Procedure ( Sender : TCustomActionManager)');
13650:   CL.AddTypeS('TActionProc', 'Procedure ( AClient : TActionClient)');
13651:   CL.AddClassN(CL.FindClass('TOBJECT'), 'TCustomActionControl');
13652:   //CL.AddTypeS('TCustomActionControlClass', 'class of TCustomActionControl');
13653:   CL.AddClassN(CL.FindClass('TOBJECT'), 'TCustomActionBarColorMap');
13654:   //CL.AddTypeS('TCustomColorMapClass', 'class of TCustomActionBarColorMap');
13655:   SIRegister_TActionBarStyle(CL);
13656:   SIRegister_TCustomActionManager(CL);
13657:   SIRegister_TActionManager(CL);
13658:   SIRegister_TActionListCollection(CL);
13659:   SIRegister_TActionListCollection(CL);
13660:   SIRegister_TActionClientsCollection(CL);
13661:   CL.AddTypeS('TBackgroundLayout', '( blNormal, blStretch, blTile, blLeftBanner, blRightBanner )');
13662:   CL.AddTypeS('TChangesAllowed', '( caModify, caMove, caDelete )');
13663:   CL.AddTypeS('TChangesAllowedSet', 'set of TChangesAllowed');
13664:   SIRegister_TActionClient(CL);
13665:   SIRegister_TActionBarItem(CL);
13666:   SIRegister_TActionBars(CL);
13667:   //CL.AddTypeS('TActionClientClass', 'class of TActionClient');
13668:   CL.AddTypeS('TCaptionOptions', '( coNone, coSelective, coAll )');
13669:   SIRegister_TActionClients(CL);
13670:   CL.AddClassN(CL.FindClass('TOBJECT'), 'TActionClientLink');
13671:   //CL.AddTypeS('TActionClientLinkClass', 'class of TActionClientLink');
13672:   SIRegister_TActionClientLink(CL);
13673:   CL.AddTypeS('TMergeRange', 'Integer');
13674:   SIRegister_TActionClientItem(CL);
13675:   SIRegister_TActionDragBaseClass(CL);
13676:   CL.AddClassN(CL.FindClass('TOBJECT'), 'TCustomActionClass');
13677:   SIRegister_TActionDragObject(CL);
13678:   SIRegister_TCategoryDragObject(CL);
13679:   SIRegister_TActionItemDragObject(CL);
13680:   SIRegister_TCustomActionBarColorMap(CL);

```

```

13681: //CL.AddTypeS('TCustomActionBarClass', 'class of TCustomActionBar');
13682: CL.AddTypeS('TBarOrientation', '( boLeftToRight, boRightToLeft, boTopToBottom, boBottomToTop )');
13683: CL.AddTypeS('TControlCreatedEvent', 'Procedure ( Sender : TObject; var Control : TCustomActionControl)');
13684: //CL.AddTypeS('TGetControlClassEvent', 'Procedure ( Sender : TCustomActionBar; '
13685: // + 'AnItem : TActionClient; var ControlClass : TCustomActionControlClass)');
13686: CL.AddTypeS('TBarEdge', '( beLeft, beRight, beEither )');
13687: SIRegister_TCustomActionBar(CL);
13688: TCMItemMsg, 'record Msg : Cardinal; Unused : Integer; Sender : TCustomActionControl; Result : Longint;
end';
13689: SIRegister_TCustomActionControl(CL);
13690: SIRegister_IActionBarDesigner(CL);
13691: SIRegister_TXToolDockForm(CL);
13692: Procedure NotifyDesigner( ActionBar : TCustomActionBar)';
13693: caAllChanges, 'LongInt').Value.ts32 := ord(caModify) or ord(caMove) or ord(caDelete);
13694: //cDefaultSchedule, 'string').SetString( '0=3' #13#10 '1=3' #13#10 '2=6' #13#10 '3=9' #13#10 + '4=12'
#13#10 '5=12' #13#10 '6=17' #13#10 '7=17' #13#10 + '8=17' #13#10 '9=23' #13#10 '10=23' #13#10 '11=23'
#13#10 + '12=23' #13#10 '13=23' #13#10 '14=31' #13#10 '15=31' #13#10 + '16=31' #13#10 '17=31' #13#10
'18=31' #13#10 '19=31' #13#10 + '20=31' #13#10 '21=31' #13#10 '22=31' #13#10 '23=31' #13#10 + '24=31'
#13#10 '25=31' #13#10);
13695: SIRegister_TActionBarStyleList(CL);
13696: end;
13697:
13698: (*-----*)
13699: procedure SIRegister_Gauges(CL: TPSPascalCompiler);
13700: begin
13701:   CL.AddTypeS('TGaugeKind', '(gkText, gkHorizontalBar, gkVerticalBar, gkPie, gkNeedle )');
13702:   SIRegister_TGauge(CL);
13703: end;
13704:
13705: (*-----*)
13706: procedure SIRegister_ColorGrd(CL: TPSPascalCompiler);
13707: begin
13708:   CL.AddConstantN('NumPaletteEntries', 'LongInt').SetInt( 20);
13709:   CL.AddTypeS('TGridOrdering', '( gol6x1, go8x2, go4x4, go2x8, golx16 )');
13710:   SIRegister_TColorGrid(CL);
13711: end;
13712:
13713: procedure SIRegister_CollPanl(CL: TPSPascalCompiler);
13714: begin
13715:   SIRegister_THeaderPanel(CL);
13716:   SIRegister_TCollapsePanel(CL);
13717: end;
13718:
13719: (*-----*)
13720: procedure SIRegister_IBCtrls(CL: TPSPascalCompiler);
13721: begin
13722:   CL.AddConstantN('MaxEvents', 'LongInt').SetInt( 15);
13723:   'EventLength', 'LongInt').SetInt( 64);
13724:   SIRegister_TIBComponent(CL);
13725:   TEventAlert, 'Procedure ( Sender : TObject; EventName : string;
13726:   EventCount : longint; var CancelAlerts : Boolean)';
13727:   SIRegister_TIBEventAlerter(CL);
13728:   CL.AddClassN(CL.FindClass('TObject'), 'EIBError');
13729: end;
13730:
13731: procedure SIRegister_IdStackWindows(CL: TPSPascalCompiler);
13732: begin
13733:   SIRegister_TIdSocketListWindows(CL);
13734:   SIRegister_TIdStackWindows(CL);
13735:   CL.AddTypeS('TLinger', 'record l_onoff : Word; l_linger : Word; end');
13736:   CL.AddTypeS('TIdLinger', 'TLinger');
13737: end;
13738:
13739: procedure SIRegister_CopyPrsr(CL: TPSPascalCompiler);
13740: begin
13741:   CL.AddConstantN('ctoEOL', 'LongInt').SetInt( Char ( 5 ));
13742:   SIRegister_TCopyParser(CL);
13743: end;
13744:
13745: procedure SIRegister_CTSVendorUtils(CL: TPSPascalCompiler);
13746: begin
13747:   CL.AddConstantN('DBXCTSSEEDField', 'String').SetString( 'SEED_ID');
13748:   CL.AddConstantN('DBXCTSGenFieldSuffix', 'String').SetString( '_FIELD');
13749:   SIRegister_TCTSUtils(CL);
13750:   CL.AddTypeS('TTransactionMode', '( tmNone, tmCommit, tmRollback )');
13751:   CL.AddTypeS('TExecuteSQLOption', '( optIgnoreException, optRaiseException )');
13752:   SIRegister_TConnectedCTSHelper(CL);
13753:   SIRegister_TCTSdbObjects(CL);
13754: end;
13755:
13756: (*-----*)
13757: procedure SIRegister_VendorTestFramework(CL: TPSPascalCompiler);
13758: begin
13759:   DBXCTSSEEDField, 'String').SetString( 'SEED_ID');
13760:   CL.AddConstantN('DBXCTSGenFieldSuffix', 'String').SetString( '_FIELD');
13761:   DBXNOTNULL, 'String').SetString( ' NOT NULL');
13762:   DEFAULT_SEED, 'LongInt').SetInt( 111);
13763:   HIRAGANA_L_a, 'Char').SetString( #$3042);
13764:   HIRAGANA_U_A, 'Char').SetString( #$3041);

```



```

13765: HIRAGANA_L_i','Char').SetString( #3043);
13766: HIRAGANA_U_I','Char').SetString( #3044);
13767: HIRAGANA_L_tu','Char').SetString( #3063);
13768: HIRAGANA_U_TU','Char').SetString( #3064);
13769: HIRAGANA_U_NI','Char').SetString( #306B);
13770: HIRAGANA_U_ZO','Char').SetString( #305E);
13771: HIRAGANA_U_SO','Char').SetString( #305D);
13772: HIRAGANA_U_MO','Char').SetString( #3080);
13773: HIRAGANA_U_YU','Char').SetString( #3086);
13774: SQUARE_DAASU','Char').SetString( #3324);
13775: GEORGIAN_DON','Char').SetString( #10A3);
13776: GEORGIAN_FI','Char').SetString( #10F6);
13777: CL.AddTypeS('TSQLType','( sqlUnknown, sqlBoolean, sqlVarchar, sqlFixedChar, '
13778: + 'sqlFixedWideChar, sqlWideVarchar, sqlSmallint, sqlInteger, sqlWord, sqlLar'
13779: + 'geInt, sqlFloat, sqlCurrency, sqlDouble, sqlNumeric, sqlDecimal, sqlAutoIn'
13780: + 'c, sqlTimeStamp, sqlTime, sqlDateTime, sqlDate, sqlBytes, sqlVarBytes, sql'
13781: + 'Blob, sqlMemo, sqlWideMemo, sqlGraphic, sqlFmtMemo, sqlTypedBinary, sqlCur'
13782: + 'sor, sqlADT, sqlArray, sqlOraBlob, sqlOraClob, sqlRaw, sqlVariant, sqlGuid)');
13783: CL.AddTypeS('TSQLTypeArray','array of TSQLType');
13784: //CL.AddTypeS('TVariantArray','array of Variant');
13785: AllSqlTypes','LongInt').Value.ts32 := ord(sqlUnknown) or ord(sqlBoolean) or ord(sqlVarchar) or
ord(sqlFixedChar) or ord(sqlFixedWideChar) or ord(sqlSmallint) or ord(sqlInteger)
or ord(sqlWord) or ord(sqlLargeint) or ord(sqlFloat) or ord(sqlCurrency) or ord(sqlDouble) or
ord(sqlNumeric) or ord(sqlDecimal) or ord(sqlAutoInc) or ord(sqlTimeStamp) or ord(sqlTime) or
ord(sqlDateTime) or ord(sqlDate) or ord(sqlBytes) or ord(sqlVarBytes) or ord(sqlBlob) or ord(sqlMemo) or
ord(sqlGraphic) or ord(sqlFmtMemo) or ord(sqlTypedBinary) or ord(sqlCursor) or ord(sqlADT) or
ord(sqlArray) or ord(sqlOraBlob) or ord(sqlOraClob) or ord(sqlVariant) or ord(sqlGuid);
13786: //CL.AddTypeS('TBytes','array of Byte');
13787: CL.AddTypeS('TSQLTypeSet','set of TSQLType');
13788: SIRegister_TTypeInfo(CL);
13789: CL.AddClassN(CL.FindClass('TOBJECT'),'TColumnInfo');
13790: CL.AddTypeS('TColumnInfoArray','array of TColumnInfo');
13791: SIRegister_TColumnInfo(CL);
13792: CL.AddClassN(CL.FindClass('TOBJECT'),'ETypeRegistryException');
13793: CL.AddClassN(CL.FindClass('TOBJECT'),'EInvalidRegistrationException');
13794: SIRegister_TTypeRegistry(CL);
13795: CL.AddTypeS('TSQLGenMode','( sgmSQLExpr, sgmTDBX )');
13796: CL.AddClassN(CL.FindClass('TOBJECT'),'TBaseDialect');
13797: //CL.AddTypeS('TBaseDialectClass','class of TBaseDialect');
13798: CL.AddTypeS('EDialectException','Exception');
13799: CL.AddTypeS('EInvalidDialect','EDialectException');
13800: SIRegister_TBaseDialect(CL);
13801: CL.AddTypeS('TDataGenMode','( dgmNone, dgmUnlocked, dgmLocked )');
13802: SIRegister_TDataGenerator(CL);
13803: end;
13804:
13805: procedure SIRegister_JvFileInfo(CL: TPSPascalCompiler);
13806: begin
13807: CL.AddTypeS('TJvExeType','( etNone, etMSDos, etWin16, etWin32, etConsole )');
13808: TJvIconModifler','( imNormal, imOverlay, imSelected, imOpen, imShellSize, imSmall )');
13809: TFileInformation','record Attributes : DWord; '
13810: DisplayName : string; ExeType : Integer; Icon : HICON; '
13811: Location : string; TypeName : string; SysIconIndex : Integer; end');
13812: TAniHeader','record dwSizeof, dwFrames, dwSteps, dwCX, dwCY, '
13813: dwBitCount, dwPlanes, dwJIFRate, dwFlags : Longint; end');
13814: SIRegister_TJvFileInfo(CL);
13815: end;
13816:
13817: procedure SIRegister_JvDBLists(CL: TPSPascalCompiler);
13818: begin
13819: CL.AddTypeS('TBDEItemType','( bdDatabases, bdDrivers, bdLangDrivers, bdUsers, bdRepositories )');
13820: SIRegister_TJvCustomBDEItems(CL);
13821: SIRegister_TJvBDEItems(CL);
13822: SIRegister_TJvDBListDataSet(CL);
13823: CL.AddTypeS('TDBItemType','( dtTables, dtStoredProcs, dtFiles, dtFunctions )');
13824: SIRegister_TJvCustomDatabaseItems(CL);
13825: SIRegister_TJvDatabaseItems(CL);
13826: CL.AddTypeS('TTabItemType','( dtFields, dtIndices, dtValChecks, dtRefInt, dtSecurity, dtFamily )');
13827: SIRegister_TJvCustomTableItems(CL);
13828: SIRegister_TJvTableItems(CL);
13829: SIRegister_TJvDatabaseDesc(CL);
13830: SIRegister_TJvDriverDesc(CL);
13831: CL.AddClassN(CL.FindClass('TOBJECT'),'TJvDatabaseList');
13832: SIRegister_TJvLangDrivList(CL);
13833: SIRegister_TJvTableList(CL);
13834: SIRegister_TJvStoredProcList(CL);
13835: CL.AddClassN(CL.FindClass('TOBJECT'),'TJvFieldList');
13836: SIRegister_TJvIndexList(CL);
13837: end;
13838:
13839: procedure SIRegister_XSBuiltIns(CL: TPSPascalCompiler);
13840: begin
13841: CL.AddConstantN('SHexMarker','String').SetString( '$');
13842: SoapTimePrefix','String').SetString( 'T');
13843: XMLDateSeparator','String').SetString( '-');
13844: XMLHourOffsetMinusMarker','String').SetString( '-');
13845: XMLHourOffsetPlusMarker','String').SetString( '+');
13846: XMLTimeSeparator','String').SetString( ':');
13847: XMLMonthPos','LongInt').SetInt( 6);

```

```

13848: XMLDayPos', 'LongInt').SetInt( 9);
13849: XMLYearPos', 'LongInt').SetInt( 1);
13850: XMLMilSecPos', 'LongInt').SetInt( 10);
13851: XMLDefaultYearDigits', 'LongInt').SetInt( 4);
13852: XMLDurationStart', 'String').SetString( 'P');
13853: XMLDurationYear', 'String').SetString( 'Y');
13854: XMLDurationMonth', 'String').SetString( 'M');
13855: XMLDurationDay', 'String').SetString( 'D');
13856: XMLDurationHour', 'String').SetString( 'H');
13857: XMLDurationMinute', 'String').SetString( 'M');
13858: XMLDurationSecond', 'String').SetString( 'S');
13859: SNAN', 'String').SetString( 'NAN');
13860: SSciNotationMarker', 'Char').SetString( 'E');
13861: SDecimal', 'Char').SetString( '.');
13862: SNegative', 'Char').SetString( '-');
13863: SPlus', 'Char').SetString( '+');
13864: SLocalTimeMarker', 'String').SetString( 'Z');
13865: MaxMonth', 'LongInt').SetInt( 12);
13866: MinMonth', 'LongInt').SetInt( 1);
13867: MaxDay', 'LongInt').SetInt( 31);
13868: MinDay', 'LongInt').SetInt( 1);
13869: SoapDecimalSeparator', 'String').SetString( '.');
13870: CL.AddClassN(CL.FindClass('TOBJECT'), 'TXSDuration');
13871: CL.AddClassN(CL.FindClass('TOBJECT'), 'TXSTime');
13872: CL.AddClassN(CL.FindClass('TOBJECT'), 'TXSDate');
13873: CL.AddClassN(CL.FindClass('TOBJECT'), 'TXSDateTime');
13874: SIRegister_TXSTime(CL);
13875: SIRegister_TXSDate(CL);
13876: SIRegister_TXSCustomDateTime(CL);
13877: SIRegister_TXSDateTime(CL);
13878: SIRegister_TXSDuration(CL);
13879: CL.AddClassN(CL.FindClass('TOBJECT'), 'EXSDateTimeException');
13880: CL.AddClassN(CL.FindClass('TOBJECT'), 'EXSDecimalException');
13881: CL.AddClassN(CL.FindClass('TOBJECT'), 'EXSHexBinaryException');
13882: SIRegister_TXSHexBinary(CL);
13883: SIRegister_TXSDecimal(CL);
13884: SIRegister_TXSString(CL);
13885: CL.AddTypeS('TXSTimeInstant', 'TXSDateTime');
13886: SIRegister_TXSBoolean(CL);
13887: SIRegister_TXSInteger(CL);
13888: SIRegister_TXSLong(CL);
13889: SIRegister_TXXMLData(CL);
13890: CL.AddDelphiFunction('Function DateTimeToXMLTime( Value : TDateTime; ApplyLocalBias : Boolean) :
WideString');
13891: Function XMLTimeToDateTime( const XMLDateTime : WideString; AsUTCtime : Boolean) : TDateTime);
13892: Function DateTimeToXSDateTime( const Value : TDateTime; ApplyLocalBias : Boolean) : TXSDateTime);
13893: Function GetDataFromFile( AFileName : string) : string);
13894: Function SoapFloatToStr( Value : double) : string);
13895: Function SoapStrToFloat( Value : string) : double);
13896: Procedure InitXSTypes);
13897: end;
13898:
13899: procedure SIRegister_JvgCrossTable(CL: TPSPascalCompiler);
13900: begin
13901: CL.AddConstantN('JvDefaultCaptionsColor', 'LongWord').SetUInt( TColor ( $00FFFD2D ));
13902: CL.AddConstantN('JvDefaultResultsColor', 'LongWord').SetUInt( TColor ( $00C5DEC5 ));
13903: CL.AddConstantN('JvDefaultIntermediateResultsColor', 'LongWord').SetUInt( TColor ( $00ABCCF1 ));
13904: CL.AddTypeS('TglPrintingStatus', '( fpsContinue, fpsResume, fpsAbort )');
13905: CL.AddTypeS('TPrintQueryEvent', 'Procedure ( Sender : TObject; ColPageCount, '
+ 'RowPageCount : Cardinal; var CanPrint : Boolean)');
13907: CL.AddTypeS('TPrintNewPageEvent', 'Procedure ( Sender : TObject; ColPageNo, R'
+ 'owPageNo : Cardinal; var PrintingStatus : TglPrintingStatus)');
13908: CL.AddTypeS('TDrawCellEvent2', 'Procedure ( Sender : TObject; ColNo, RowNo : C'
+ 'ardinal; Value : string; var CanPrint : Boolean)');
13911: CL.AddTypeS('TCalcResultEvent', 'Procedure ( Sender : TObject; ColNo, RowNo : '
+ 'Cardinal; CellValue : string; IntermediateColResult, IntermediateRowResul'
+ 't, ColResult, RowResult : Single)');
13914: CL.AddTypeS('TDuplicateCellValueEvent', 'Procedure ( Sender : TObject; ColNo, '
+ 'RowNo : Cardinal; Value : string; var UseDuplicateValue : Boolean)');
13915: CL.AddTypeS('TPCTOptions', 'set of ( fcoIntermediateColResults, fcoIntermedia'
+ 'teRowResults, fcoColResults, fcoRowResults, fcoIntermediateColCaptions, fc'
+ 'oIntermediateRowCaptions, fcoIntermediateLeftIndent, fcoIntermediateTopInd'
+ 'ent, fcoIntermediateRightIndent, fcoIntermediateBottomIndent, fcoShowPageN'
+ 'umbers, fcoVertColCaptionsFont )');
13921: CL.AddTypeS('TPCTTableElement', '( teTitle, teCell, teColCapt, teRowCapt, teCo'
+ 'lRes, teRowRes, teColRes, teRowRes )');
13923: CL.AddTypeS('TPrintTableElement', 'Procedure ( Sender : TObject; var Text : s'
+ 'tring; ColNo, RowNo : Integer; TableElement : TPCTTableElement; var Font : '
+ 'TFont; var Color : TColor; var AlignFlags : Word; var CanPrint : Boolean)');
13926: SIRegister_TJvgPrintCrossTableColors(CL);
13927: SIRegister_TJvgPrintCrossTableFonts(CL);
13928: SIRegister_TJvgPrintCrossTableIndents(CL);
13929: SIRegister_TJvgPrintCrossTable(CL);
13930: end;
13931:
13932: procedure SIRegister_JvgReport(CL: TPSPascalCompiler);
13933: begin
13934: CL.AddClassN(CL.FindClass('TOBJECT'), 'TJvgReport');
13935: CL.AddTypeS('TJvgReportParamKind', '( gptUnknown, gptEdit, gptRadio, gptCheck)');

```

```

13936:   SIRegister_TJvgReportScrollBar(CL);
13937:   SIRegister_TJvgReportItem(CL);
13938:   CL.AddTypeS('TJvgReportBeforePrintEvent', 'Procedure ( Sender : TJvgReport)');
13939:   SIRegister_TJvgReport(CL);
13940: end;
13941:
13942: procedure SIRegister_WaveConverter(CL: TPSPascalCompiler);
13943: begin
13944: //type
13945:   //TriffID = array[0..3] of char;
13946:   CL.AddTypeS('TriffID', 'array[0..3] of char;');
13947:   CL.AddTypeS('TriffHeader', 'record ID : TriffID; BytesFollowing : DWord; end;');
13948:   CL.AddTypeS('TACMWaveFormat', 'record Format : TWaveFormatEx; end;');
13949:   CL.AddTypeS('TACMWaveFormat2', 'record RawData: Array[0..128] of byte; end;');
13950:   SIRegister_TWaveConverter(CL);
13951: end;
13952:
13953: procedure SIRegister_ComObj2(CL: TPSPascalCompiler);
13954: CT_USERID, 'String').SetString( 'USER ID=');
13955: CL.AddConstantN('CT_PROVIDER', 'String').SetString( 'PROVIDER=');
13956: CT_FILENAME, 'String').SetString( 'FILE NAME=');
13957: //CL.AddTypeS('CursorTypeEnum', 'TToEnum');
13958: adOpenUnspecified, 'LongWord').SetUInt( $FFFFFFF);
13959: adOpenForwardOnly, 'LongWord').SetUInt( $00000000);
13960: adOpenKeyset, 'LongWord').SetUInt( $00000001);
13961: adOpenDynamic, 'LongWord').SetUInt( $00000002);
13962: adOpenStatic, 'LongWord').SetUInt( $00000003);
13963: adHoldRecords, 'LongWord').SetUInt( $00000100);
13964: adMovePrevious, 'LongWord').SetUInt( $00000200);
13965: adAddNew, 'LongWord').SetUInt( $01000400);
13966: adDelete, 'LongWord').SetUInt( $01000800);
13967: adUpdate, 'LongWord').SetUInt( $01000800);
13968: adBookmark, 'LongWord').SetUInt( $00002000);
13969: adApproxPosition, 'LongWord').SetUInt( $00004000);
13970: adUpdateBatch, 'LongWord').SetUInt( $00010000);
13971: adResync, 'LongWord').SetUInt( $00020000);
13972: adNotify, 'LongWord').SetUInt( $00040000);
13973: adFind, 'LongWord').SetUInt( $00080000);
13974: adSeek, 'LongWord').SetUInt( $00400000);
13975: adIndex, 'LongWord').SetUInt( $00800000);
13976: // CL.AddTypeS('LockTypeEnum', 'TToEnum');
13977: adLockUnspecified, 'LongWord').SetUInt( $FFFFFFF);
13978: adLockReadOnly, 'LongWord').SetUInt( $00000001);
13979: adLockPessimistic, 'LongWord').SetUInt( $00000002);
13980: adLockOptimistic, 'LongWord').SetUInt( $00000003);
13981: adLockBatchOptimistic, 'LongWord').SetUInt( $00000004);
13982: CL.AddTypeS('ExecuteOptionEnum', 'TToEnum');
13983: adOptionUnspecified, 'LongWord').SetUInt( $FFFFFFF);
13984: adAsyncExecute, 'LongWord').SetUInt( $00000010);
13985: adAsyncFetch, 'LongWord').SetUInt( $00000020);
13986: adAsyncFetchNonBlocking, 'LongWord').SetUInt( $00000040);
13987: adExecuteNoRecords, 'LongWord').SetUInt( $00000080);
13988: CL.AddTypeS('ConnectOptionEnum', 'TToEnum');
13989: adConnectUnspecified, 'LongWord').SetUInt( $FFFFFFF);
13990: adAsyncConnect, 'LongWord').SetUInt( $00000010);
13991: CL.AddTypeS('ObjectStateEnum', 'TToEnum');
13992: adStateClosed, 'LongWord').SetUInt( $00000000);
13993: adStateOpen, 'LongWord').SetUInt( $00000001);
13994: adStateConnecting, 'LongWord').SetUInt( $00000002);
13995: adStateExecuting, 'LongWord').SetUInt( $00000004);
13996: adStateFetching, 'LongWord').SetUInt( $00000008);
13997: CL.AddTypeS('CursorLocationEnum', 'TToEnum');
13998: adUseNone, 'LongWord').SetUInt( $00000001);
13999: adUseServer, 'LongWord').SetUInt( $00000002);
14000: adUseClient, 'LongWord').SetUInt( $00000003);
14001: adUseClientBatch, 'LongWord').SetUInt( $00000003);
14002: CL.AddTypeS('DataTypeEnum', 'TToEnum');
14003: adEmpty, 'LongWord').SetUInt( $00000000);
14004: adTinyInt, 'LongWord').SetUInt( $00000010);
14005: adSmallInt, 'LongWord').SetUInt( $00000002);
14006: adInteger, 'LongWord').SetUInt( $00000003);
14007: adBigInt, 'LongWord').SetUInt( $00000014);
14008: adUnsignedTinyInt, 'LongWord').SetUInt( $00000011);
14009: adUnsignedSmallInt, 'LongWord').SetUInt( $00000012);
14010: adUnsignedInt, 'LongWord').SetUInt( $00000013);
14011: adUnsignedBigInt, 'LongWord').SetUInt( $00000015);
14012: adSingle, 'LongWord').SetUInt( $00000004);
14013: adDouble, 'LongWord').SetUInt( $00000005);
14014: adCurrency, 'LongWord').SetUInt( $00000006);
14015: adDecimal, 'LongWord').SetUInt( $0000000E);
14016: adNumeric, 'LongWord').SetUInt( $00000083);
14017: adBoolean, 'LongWord').SetUInt( $0000000B);
14018: adError, 'LongWord').SetUInt( $0000000A);
14019: adUserDefined, 'LongWord').SetUInt( $00000084);
14020: adVariant, 'LongWord').SetUInt( $0000000C);
14021: adIDispatch, 'LongWord').SetUInt( $00000009);
14022: adIUnknown, 'LongWord').SetUInt( $0000000D);
14023: adGUID, 'LongWord').SetUInt( $00000048);
14024: adDate, 'LongWord').SetUInt( $00000007);

```

```

14025: adDBDate', 'LongWord').SetUInt( $00000085);
14026: adDBTime', 'LongWord').SetUInt( $00000086);
14027: adDBTimeStamp', 'LongWord').SetUInt( $00000087);
14028: adBSTR', 'LongWord').SetUInt( $00000088);
14029: adChar', 'LongWord').SetUInt( $00000081);
14030: adVarChar', 'LongWord').SetUInt( $000000C8);
14031: adLongVarChar', 'LongWord').SetUInt( $000000C9);
14032: adWChar', 'LongWord').SetUInt( $00000082);
14033: adVarWChar', 'LongWord').SetUInt( $000000CA);
14034: adLongVarWChar', 'LongWord').SetUInt( $000000CB);
14035: adBinary', 'LongWord').SetUInt( $00000080);
14036: adVarBinary', 'LongWord').SetUInt( $000000CC);
14037: adLongVarBinary', 'LongWord').SetUInt( $000000CD);
14038: adChapter', 'LongWord').SetUInt( $00000088);
14039: adFileTime', 'LongWord').SetUInt( $00000040);
14040: adDBFileTime', 'LongWord').SetUInt( $00000089);
14041: adPropVariant', 'LongWord').SetUInt( $0000008A);
14042: adVarNumeric', 'LongWord').SetUInt( $0000008B);
14043: CL.AddTypeS('FieldAttributeEnum', 'TolEnum');
14044: adFldUnspecified', 'LongWord').SetUInt( $FFFFFFF);
14045: adFldMayDefer', 'LongWord').SetUInt( $00000002);
14046: adFldUpdatable', 'LongWord').SetUInt( $00000004);
14047: adFldUnknownUpdatable', 'LongWord').SetUInt( $00000008);
14048: adFldFixed', 'LongWord').SetUInt( $00000010);
14049: adFldIsNullbale', 'LongWord').SetUInt( $00000020);
14050: adFldMayBeNull', 'LongWord').SetUInt( $00000040);
14051: adFldLong', 'LongWord').SetUInt( $00000080);
14052: adFldRowID', 'LongWord').SetUInt( $00000100);
14053: adFldRowVersion', 'LongWord').SetUInt( $00000200);
14054: adFldCacheDeferred', 'LongWord').SetUInt( $00001000);
14055: adFldNegativeScale', 'LongWord').SetUInt( $00004000);
14056: adFldKeyColumn', 'LongWord').SetUInt( $00008000);
14057: CL.AddTypeS('EditModeEnum', 'TolEnum');
14058: adEditNone', 'LongWord').SetUInt( $00000000);
14059: adEditInProgress', 'LongWord').SetUInt( $00000001);
14060: adEditAdd', 'LongWord').SetUInt( $00000002);
14061: adEditDelete', 'LongWord').SetUInt( $00000004);
14062: CL.AddTypeS('RecordStatusEnum', 'TolEnum');
14063: adRecOK', 'LongWord').SetUInt( $00000000);
14064: adRecNew', 'LongWord').SetUInt( $00000001);
14065: adRecModified', 'LongWord').SetUInt( $00000002);
14066: adRecDeleted', 'LongWord').SetUInt( $00000004);
14067: adRecUnmodified', 'LongWord').SetUInt( $00000008);
14068: adRecInvalid', 'LongWord').SetUInt( $00000010);
14069: adRecMultipleChanges', 'LongWord').SetUInt( $00000040);
14070: adRecPendingChanges', 'LongWord').SetUInt( $00000080);
14071: adRecCanceled', 'LongWord').SetUInt( $00000100);
14072: adRecCantRelease', 'LongWord').SetUInt( $00000400);
14073: adRecConcurrencyViolation', 'LongWord').SetUInt( $00000800);
14074: adRecIntegrityViolation', 'LongWord').SetUInt( $00001000);
14075: adRecMaxChangesExceeded', 'LongWord').SetUInt( $00002000);
14076: adRecObjectOpen', 'LongWord').SetUInt( $00004000);
14077: adRecOutOfMemory', 'LongWord').SetUInt( $00008000);
14078: adRecPermissionDenied', 'LongWord').SetUInt( $00010000);
14079: adRecSchemaViolation', 'LongWord').SetUInt( $00020000);
14080: adRecDBDeleted', 'LongWord').SetUInt( $00040000);
14081: CL.AddTypeS('GetRowsOptionEnum', 'TolEnum');
14082: adGetRowsRest', 'LongWord').SetUInt( $FFFFFFF);
14083: CL.AddTypeS('PositionEnum', 'TolEnum');
14084: adPosUnknown', 'LongWord').SetUInt( $FFFFFFF);
14085: adPosBOF', 'LongWord').SetUInt( $FFFFFFFE);
14086: adPosEOF', 'LongWord').SetUInt( $FFFFFFFD);
14087: CL.AddTypeS('BookmarkEnum', 'TolEnum');
14088: adBookmarkCurrent', 'LongWord').SetUInt( $00000000);
14089: adBookmarkFirst', 'LongWord').SetUInt( $00000001);
14090: adBookmarkLast', 'LongWord').SetUInt( $00000002);
14091: CL.AddTypeS('MarshalOptionsEnum', 'TolEnum');
14092: adMarshalAll', 'LongWord').SetUInt( $00000000);
14093: adMarshalModifiedOnly', 'LongWord').SetUInt( $00000001);
14094: CL.AddTypeS('AffectEnum', 'TolEnum');
14095: adAffectCurrent', 'LongWord').SetUInt( $00000001);
14096: adAffectGroup', 'LongWord').SetUInt( $00000002);
14097: adAffectAll', 'LongWord').SetUInt( $00000003);
14098: adAffectAllChapters', 'LongWord').SetUInt( $00000004);
14099: CL.AddTypeS('ResyncEnum', 'TolEnum');
14100: adResyncUnderlyingValues', 'LongWord').SetUInt( $00000001);
14101: adResyncAllValues', 'LongWord').SetUInt( $00000002);
14102: CL.AddTypeS('CompareEnum', 'TolEnum');
14103: adCompareLessThan', 'LongWord').SetUInt( $00000000);
14104: adCompareEqual', 'LongWord').SetUInt( $00000001);
14105: adCompareGreaterThan', 'LongWord').SetUInt( $00000002);
14106: adCompareNotEqual', 'LongWord').SetUInt( $00000003);
14107: adCompareNotComparable', 'LongWord').SetUInt( $00000004);
14108: CL.AddTypeS('FilterGroupEnum', 'TolEnum');
14109: adFilterNone', 'LongWord').SetUInt( $00000000);
14110: adFilterPendingRecords', 'LongWord').SetUInt( $00000001);
14111: adFilterAffectedRecords', 'LongWord').SetUInt( $00000002);
14112: adFilterFetchedRecords', 'LongWord').SetUInt( $00000003);
14113: adFilterPredicate', 'LongWord').SetUInt( $00000004);

```



```

14114: adFilterConflictingRecords', 'LongWord').SetUInt( $00000005);
14115: CL.AddTypeS('SearchDirectionEnum', 'TolEnum');
14116: adSearchForward', 'LongWord').SetUInt( $00000001);
14117: adSearchBackward', 'LongWord').SetUInt( $FFFFFFF);
14118: CL.AddTypeS('PersistFormatEnum', 'TolEnum');
14119: adPersistADTG', 'LongWord').SetUInt( $00000000);
14120: adPersistXML', 'LongWord').SetUInt( $00000001);
14121: CL.AddTypeS('StringFormatEnum', 'TolEnum');
14122: adClipString', 'LongWord').SetUInt( $00000002);
14123: CL.AddTypeS('ConnectPromptEnum', 'TolEnum');
14124: adPromptAlways', 'LongWord').SetUInt( $00000001);
14125: adPromptComplete', 'LongWord').SetUInt( $00000002);
14126: adPromptCompleteRequired', 'LongWord').SetUInt( $00000003);
14127: adPromptNever', 'LongWord').SetUInt( $00000004);
14128: CL.AddTypeS('ConnectModeEnum', 'TolEnum');
14129: adModeUnknown', 'LongWord').SetUInt( $00000000);
14130: adModeRead', 'LongWord').SetUInt( $00000001);
14131: adModeWrite', 'LongWord').SetUInt( $00000002);
14132: adModeReadWrite', 'LongWord').SetUInt( $00000003);
14133: adModeShareDenyRead', 'LongWord').SetUInt( $00000004);
14134: adModeShareDenyWrite', 'LongWord').SetUInt( $00000008);
14135: adModeShareExclusive', 'LongWord').SetUInt( $0000000C);
14136: adModeShareDenyNone', 'LongWord').SetUInt( $00000010);
14137: CL.AddTypeS('IsolationLevelEnum', 'TolEnum');
14138: adXactUnspecified', 'LongWord').SetUInt( $FFFFFFF);
14139: adXactChaos', 'LongWord').SetUInt( $00000010);
14140: adXactReadUncommitted', 'LongWord').SetUInt( $00000100);
14141: adXactBrowse', 'LongWord').SetUInt( $00000100);
14142: adXactCursorStability', 'LongWord').SetUInt( $00001000);
14143: adXactReadCommitted', 'LongWord').SetUInt( $00001000);
14144: adXactRepeatableRead', 'LongWord').SetUInt( $00010000);
14145: adXactSerializable', 'LongWord').SetUInt( $00100000);
14146: adXactIsolated', 'LongWord').SetUInt( $00100000);
14147: CL.AddTypeS('XactAttributeEnum', 'TolEnum');
14148: adXactCommitRetaining', 'LongWord').SetUInt( $00020000);
14149: adXactAbortRetaining', 'LongWord').SetUInt( $00040000);
14150: adXactAsyncPhaseOne', 'LongWord').SetUInt( $00080000);
14151: adXactSyncPhaseOne', 'LongWord').SetUInt( $00100000);
14152: CL.AddTypeS('PropertyAttributesEnum', 'TolEnum');
14153: adPropNotSupported', 'LongWord').SetUInt( $00000000);
14154: adPropRequired', 'LongWord').SetUInt( $00000001);
14155: adPropOptional', 'LongWord').SetUInt( $00000002);
14156: adPropRead', 'LongWord').SetUInt( $00000200);
14157: adPropWrite', 'LongWord').SetUInt( $00000400);
14158: CL.AddTypeS('ErrorValueEnum', 'TolEnum');
14159: adErrInvalidArgument', 'LongWord').SetUInt( $00000BB9);
14160: adErrNoCurrentRecord', 'LongWord').SetUInt( $00000BCD);
14161: adErrIllegalOperation', 'LongWord').SetUInt( $00000C93);
14162: adErrInTransaction', 'LongWord').SetUInt( $00000CAE);
14163: adErrFeatureNotAvailable', 'LongWord').SetUInt( $00000CB3);
14164: adErrItemNotFound', 'LongWord').SetUInt( $00000CC1);
14165: adErrObjectInCollection', 'LongWord').SetUInt( $00000D27);
14166: adErrObjectNotSet', 'LongWord').SetUInt( $00000D5C);
14167: adErrDataConversion', 'LongWord').SetUInt( $00000D5D);
14168: adErrObjectClosed', 'LongWord').SetUInt( $00000E78);
14169: adErrObjectOpen', 'LongWord').SetUInt( $00000E79);
14170: adErrProviderNotFound', 'LongWord').SetUInt( $00000E7A);
14171: adErrBoundToCommand', 'LongWord').SetUInt( $00000E7B);
14172: adErrInvalidParamInfo', 'LongWord').SetUInt( $00000E7C);
14173: adErrInvalidConnection', 'LongWord').SetUInt( $00000E7D);
14174: adErrNotReentrant', 'LongWord').SetUInt( $00000E7E);
14175: adErrStillExecuting', 'LongWord').SetUInt( $00000E7F);
14176: adErrOperationCancelled', 'LongWord').SetUInt( $00000E80);
14177: adErrStillConnecting', 'LongWord').SetUInt( $00000E81);
14178: adErrNotExecuting', 'LongWord').SetUInt( $00000E83);
14179: adErrUnsafeOperation', 'LongWord').SetUInt( $00000E84);
14180: CL.AddTypeS('ParameterAttributesEnum', 'TolEnum');
14181: adParamSigned', 'LongWord').SetUInt( $00000010);
14182: adParamNullable', 'LongWord').SetUInt( $00000040);
14183: adParamLong', 'LongWord').SetUInt( $00000080);
14184: CL.AddTypeS('ParameterDirectionEnum', 'TolEnum');
14185: adParamUnknown', 'LongWord').SetUInt( $00000000);
14186: adParamInput', 'LongWord').SetUInt( $00000001);
14187: adParamOutput', 'LongWord').SetUInt( $00000002);
14188: adParamInputOutput', 'LongWord').SetUInt( $00000003);
14189: adParamReturnValue', 'LongWord').SetUInt( $00000004);
14190: CL.AddTypeS('CommandTypeEnum', 'TolEnum');
14191: adCmdUnspecified', 'LongWord').SetUInt( $FFFFFFF);
14192: adCmdUnknown', 'LongWord').SetUInt( $00000008);
14193: adCmdText', 'LongWord').SetUInt( $00000001);
14194: adCmdTable', 'LongWord').SetUInt( $00000002);
14195: adCmdStoredProc', 'LongWord').SetUInt( $00000004);
14196: adCmdFile', 'LongWord').SetUInt( $00000100);
14197: adCmdTableDirect', 'LongWord').SetUInt( $00000200);
14198: CL.AddTypeS('EventStatusEnum', 'TolEnum');
14199: adStatusOK', 'LongWord').SetUInt( $00000001);
14200: adStatusErrorsOccurred', 'LongWord').SetUInt( $00000002);
14201: adStatusCantDeny', 'LongWord').SetUInt( $00000003);
14202: adStatusCancel', 'LongWord').SetUInt( $00000004);

```

```

14203: adStatusUnwantedEvent', 'LongWord').SetUInt( $00000005);
14204: CL.AddTypeS('EventReasonEnum', 'TolEnum');
14205: adRsnAddNew', 'LongWord').SetUInt( $00000001);
14206: adRsnDelete', 'LongWord').SetUInt( $00000002);
14207: adRsnUpdate', 'LongWord').SetUInt( $00000003);
14208: adRsnUndoUpdate', 'LongWord').SetUInt( $00000004);
14209: adRsnUndoAddNew', 'LongWord').SetUInt( $00000005);
14210: adRsnUndoDelete', 'LongWord').SetUInt( $00000006);
14211: adRsnRequery', 'LongWord').SetUInt( $00000007);
14212: adRsnResynch', 'LongWord').SetUInt( $00000008);
14213: adRsnClose', 'LongWord').SetUInt( $00000009);
14214: adRsnMove', 'LongWord').SetUInt( $0000000A);
14215: adRsnFirstChange', 'LongWord').SetUInt( $0000000B);
14216: adRsnMoveFirst', 'LongWord').SetUInt( $0000000C);
14217: adRsnMoveNext', 'LongWord').SetUInt( $0000000D);
14218: adRsnMovePrevious', 'LongWord').SetUInt( $0000000E);
14219: adRsnMoveLast', 'LongWord').SetUInt( $0000000F);
14220: CL.AddTypeS('SchemaEnum', 'TolEnum');
14221: adSchemaProviderSpecific', 'LongWord').SetUInt( $FFFFFFF);
14222: adSchemaAsserts', 'LongWord').SetUInt( $00000000);
14223: adSchemaCatalogs', 'LongWord').SetUInt( $00000001);
14224: adSchemaCharacterSets', 'LongWord').SetUInt( $00000002);
14225: adSchemaCollations', 'LongWord').SetUInt( $00000003);
14226: adSchemaColumns', 'LongWord').SetUInt( $00000004);
14227: adSchemaCheckConstraints', 'LongWord').SetUInt( $00000005);
14228: adSchemaConstraintColumnUsage', 'LongWord').SetUInt( $00000006);
14229: adSchemaConstraintTableUsage', 'LongWord').SetUInt( $00000007);
14230: adSchemaKeyColumnUsage', 'LongWord').SetUInt( $00000008);
14231: adSchemaReferentialConstraints', 'LongWord').SetUInt( $00000009);
14232: adSchemaTableConstraints', 'LongWord').SetUInt( $0000000A);
14233: adSchemaColumnsDomainUsage', 'LongWord').SetUInt( $0000000B);
14234: adSchemaIndexes', 'LongWord').SetUInt( $0000000C);
14235: adSchemaColumnPrivileges', 'LongWord').SetUInt( $0000000D);
14236: adSchemaTablePrivileges', 'LongWord').SetUInt( $0000000E);
14237: adSchemaUsagePrivileges', 'LongWord').SetUInt( $0000000F);
14238: adSchemaProcedures', 'LongWord').SetUInt( $00000010);
14239: adSchemaSchemata', 'LongWord').SetUInt( $00000011);
14240: adSchemaSQLLanguages', 'LongWord').SetUInt( $00000012);
14241: adSchemaStatistics', 'LongWord').SetUInt( $00000013);
14242: adSchemaTables', 'LongWord').SetUInt( $00000014);
14243: adSchemaTranslations', 'LongWord').SetUInt( $00000015);
14244: adSchemaProviderTypes', 'LongWord').SetUInt( $00000016);
14245: adSchemaViews', 'LongWord').SetUInt( $00000017);
14246: adSchemaViewColumnUsage', 'LongWord').SetUInt( $00000018);
14247: adSchemaViewTableUsage', 'LongWord').SetUInt( $00000019);
14248: adSchemaProcedureParameters', 'LongWord').SetUInt( $0000001A);
14249: adSchemaForeignKeys', 'LongWord').SetUInt( $0000001B);
14250: adSchemaPrimaryKeys', 'LongWord').SetUInt( $0000001C);
14251: adSchemaProcedureColumns', 'LongWord').SetUInt( $0000001D);
14252: adSchemaDBInfoKeywords', 'LongWord').SetUInt( $0000001E);
14253: adSchemaDBInfoLiterals', 'LongWord').SetUInt( $0000001F);
14254: adSchemaCubes', 'LongWord').SetUInt( $00000020);
14255: adSchemaDimensions', 'LongWord').SetUInt( $00000021);
14256: adSchemaHierarchies', 'LongWord').SetUInt( $00000022);
14257: adSchemaLevels', 'LongWord').SetUInt( $00000023);
14258: adSchemaMeasures', 'LongWord').SetUInt( $00000024);
14259: adSchemaProperties', 'LongWord').SetUInt( $00000025);
14260: adSchemaMembers', 'LongWord').SetUInt( $00000026);
14261: adSchemaTrustees', 'LongWord').SetUInt( $00000027);
14262: CL.AddTypeS('SeekEnum', 'TolEnum');
14263: adSeekFirstEQ', 'LongWord').SetUInt( $00000001);
14264: adSeekLastEQ', 'LongWord').SetUInt( $00000002);
14265: adSeekAfterEQ', 'LongWord').SetUInt( $00000004);
14266: adSeekAfter', 'LongWord').SetUInt( $00000008);
14267: adSeekBeforeEQ', 'LongWord').SetUInt( $00000010);
14268: adSeekBefore', 'LongWord').SetUInt( $00000020);
14269: CL.AddTypeS('ADCPROP_UPDATECRITERIA_ENUM', 'TolEnum');
14270: adCriteriaKey', 'LongWord').SetUInt( $00000000);
14271: adCriteriaAllCols', 'LongWord').SetUInt( $00000001);
14272: adCriteriaUpdCols', 'LongWord').SetUInt( $00000002);
14273: adCriteriaTimeStamp', 'LongWord').SetUInt( $00000003);
14274: CL.AddTypeS('ADCPROP_ASYNCTHREADPRIORITY_ENUM', 'TolEnum');
14275: adPriorityLowest', 'LongWord').SetUInt( $00000001);
14276: adPriorityBelowNormal', 'LongWord').SetUInt( $00000002);
14277: adPriorityNormal', 'LongWord').SetUInt( $00000003);
14278: adPriorityAboveNormal', 'LongWord').SetUInt( $00000004);
14279: adPriorityHighest', 'LongWord').SetUInt( $00000005);
14280: CL.AddTypeS('CEResyncEnum', 'TolEnum');
14281: adResyncNone', 'LongWord').SetUInt( $00000000);
14282: adResyncAutoIncrement', 'LongWord').SetUInt( $00000001);
14283: adResyncConflicts', 'LongWord').SetUInt( $00000002);
14284: adResyncUpdates', 'LongWord').SetUInt( $00000004);
14285: adResyncInserts', 'LongWord').SetUInt( $00000008);
14286: 'adResyncAll', 'LongWord').SetUInt( $0000000F);
14287: CL.AddTypeS('ADCPROP_AUTORECALC_ENUM', 'TolEnum');
14288: 'adRecalcUpFront', 'LongWord').SetUInt( $00000000);
14289: 'adRecalcAlways', 'LongWord').SetUInt( $00000001);
14290: end;
14291:

```

```

14292: type
14293: {$IFDEF DMB_DELPHIXE3}
14294: [ComponentPlatformsAttribute(pidWin32 or pidWin64 or
14295:                               pidLinux32 or
14296:                               {$IFDEF DMB_DELPHIXE5}
14297:                               pidAndroid or
14298:                               {$ENDIF}
14299:                               {$IFDEF DMB_DELPHIXE4}
14300:                               pidiOSDevice or pidiOSSimulator or
14301:                               {$ENDIF}
14302:                               pidOSX32)]
14303: {$ENDIF}
14304:
14305: Connection String
14306: program TestDbTreeViewMainForm2_ACCESS;
14307: connectionString:='Provider=Microsoft.Jet.OLEDB.4.0;Data Source='
14308:                   +Exepath+'\examples\detail.mdb;Persist Security Info=False';
14309:
14310: Ref:
14311: 22275: 1. writeln(SHA1(Exepath+'\maxbox3.exe'))
14312: 22276: 2. shdig:= TSHA1Digest;
14313: 22277: shdig:= GetSHA1OfFile(false,exepath+'\maxbox3.exe');
14314: 22278: for i:= 0 to 19 do write(ByteToHex(shdig[i]));
14315: 22279:
14316: 22280: 3. writeln(IntToHex(CRC32OfFile(exepath+'\maxbox3.exe'),4));
14317: 22281:
14318:
14319: Doc End TYPE List mX4
14320: *****
14321:
14322: CONST List mX4 3.9.9.7
14323: *****
14324: _ABORTDOC, 'LongInt.SetInt( 2); 4383
14325: _DefaultDecimals, 'LongInt.SetInt( 10); 979
14326: _DELETE, 'LongWord.SetUInt( $00010000); 2410
14327: _ENDDOC, 'LongInt.SetInt( 11); 4392
14328: _EXTTEXTOUT, 'LongWord.SetUInt( $200); 4427
14329: _MOUSE_EVENT, 'LongInt.SetInt( 2); 7877
14330: _NoDecimal, 'LongInt.SetInt( - 255); 978
14331: _SETABORTPROC, 'LongInt.SetInt( 9); 4390
14332: _SETMITERLIMIT, 'LongInt.SetInt( 23); 4406
14333: _STARTDOC, 'LongInt.SetInt( 10); 4391
14334: _STRETCHBLT, 'LongWord.SetUInt( $800); 4435
14335: ABE_BOTTOM, 'LongInt.SetInt( 3); 2887
14336: ABE_LEFT, 'LongInt.SetInt( 0); 2884
14337: ABE_RIGHT, 'LongInt.SetInt( 2); 2886
14338: ABE_TOP, 'LongInt.SetInt( 1); 2885
14339: ABM_ACTIVATE, 'LongWord.SetUInt( $00000006); 2874
14340: ABM_GETAUTOHIDEABAR, 'LongWord.SetUInt( $00000007); 2875
14341: ABM_GETSTATE, 'LongWord.SetUInt( $00000004); 2872
14342: ABM_GETTASKBARPOS, 'LongWord.SetUInt( $00000005); 2873
14343: ABM_NEW, 'LongWord.SetUInt( $00000000); 2868
14344: ABM_QUERYPOS, 'LongWord.SetUInt( $00000002); 2870
14345: ABM_REMOVE, 'LongWord.SetUInt( $00000001); 2869
14346: ABM_SETAUTOHIDEABAR, 'LongWord.SetUInt( $00000008); 2876
14347: ABM_SETPOS, 'LongWord.SetUInt( $00000003); 2871
14348: ABM_WINDOWPOSCHANGED, 'LongWord.SetUInt( $00000009); 2877
14349: ABN_FULLSCREENAPP, 'LongWord.SetUInt( $00000002); 2880
14350: ABN_POSCHANGED, 'LongWord.SetUInt( $00000001); 2879
14351: ABN_STATECHANGE, 'LongWord.SetUInt( $00000000); 2878
14352: ABN_WINDOWARRANGE, 'LongWord.SetUInt( $00000003); 2881
14353: ABS_ALWAYSONTOP, 'LongWord.SetUInt( $00000002); 2883
14354: ABS_AUTOHIDE, 'LongWord.SetUInt( $00000001); 2882
14355: ACCESS_FILTERKEYS, 'LongWord.SetUInt( $0002); 2786
14356: ACCESS_MOUSEKEYS, 'LongWord.SetUInt( $0003); 2787
14357: ACCESS_STICKYKEYS, 'LongWord.SetUInt( $0001); 2785
14358: ACHARSET, 'String.SetString(b64_XXEncode); 2234
14359: advapi32, 'String.SetString( 'advapi32.dll; 10
14360: AI_INSTANCECREATED, 'LongWord.SetUInt( $0001); 2443
14361: AI_INSTANCEDESTROYED, 'LongWord.SetUInt( $0002); 2444
14362: AI_USERMSG, 'LongWord.SetUInt( $0003); 2445
14363: AllFilesMask, 'String.SetString( '*; 1614
14364: AllFilesMask, 'String.SetString( '*.*; 1612
14365: ALLFUNCTIONSLIST, 'String.SetString( 'upsi_allfunctionslist.txt; 938
14366: ALLFUNCTIONSLIST, 'String.SetString( 'upsi_allfunctionslist.txt; 962
14367: ALLFUNCTIONSLISTPDF, 'String.SetString( 'docs\VCL.pdf; 940
14368: ALLFUNCTIONSLISTPDF, 'String.SetString( 'maxbox_functions_all.pdf; 963
14369: ALLGLSIDES, 'LongInt.Value.ts32 := ord(fsdLeft) or ord(fsdTop) or ord(fsdRight) or ord(fsdBottom); 1597
14370: ALLOBJECTSLIST, 'String.SetString( 'docs\VCL.pdf; 940
14371: ALLOBJECTSLIST, 'String.SetString( 'docs\VCL.pdf; 964
14372: AllParamTypes, 'LongInt.Value.ts32 := ord(ptUnknown) or ord(ptInput) or ord(ptOutput) or
ord(ptInputOutput) or ord(ptResult); 883
14373: AllRecords, 'LongInt.SetInt( - 1); 884
14374: ALLRESOURCELIST, 'String.SetString( 'docs\upsi_allresourcelist.txt; 941
14375: ALLRESOURCELIST, 'String.SetString( 'docs\upsi_allresourcelist.txt; 965
14376: AMD_APIC_FLAG, 'LongWord.SetUInt( $00000200); 1465
14377: AMD_BIT_10, 'LongWord.SetUInt( $00000400); 1466
14378: AMD_BIT_11, 'LongWord.SetUInt( $00000800); 1467
14379: AMD_BIT_14, 'LongWord.SetUInt( $00004000); 1470

```

```

14380: AMD_BIT_16', 'LongWord.SetUInt( $00010000); 1472
14381: AMD_BIT_17', 'LongWord.SetUInt( $00020000); 1473
14382: AMD_BIT_18', 'LongWord.SetUInt( $00040000); 1474
14383: AMD_BIT_19', 'LongWord.SetUInt( $00080000); 1475
14384: AMD_BIT_20', 'LongWord.SetUInt( $00100000); 1476
14385: AMD_BIT_21', 'LongWord.SetUInt( $00200000); 1477
14386: AMD_BIT_22', 'LongWord.SetUInt( $00400000); 1478
14387: AMD_BIT_24', 'LongWord.SetUInt( $01000000); 1480
14388: AMD_BIT_25', 'LongWord.SetUInt( $02000000); 1481
14389: AMD_BIT_26', 'LongWord.SetUInt( $04000000); 1482
14390: AMD_BIT_27', 'LongWord.SetUInt( $08000000); 1483
14391: AMD_BIT_28', 'LongWord.SetUInt( $10000000); 1484
14392: AMD_BIT_29', 'LongWord.SetUInt( $20000000); 1485
14393: AMD_BIT_30', 'LongWord.SetUInt( $40000000); 1486
14394: AMD_BIT_31', 'LongWord.SetUInt( DWORD ( $80000000 )); 1487
14395: AMD_BIT_6', 'LongWord.SetUInt( $00000040); 1462
14396: AMD_CMOV_FLAG', 'LongWord.SetUInt( $00008000); 1471
14397: AMD_CX8_FLAG', 'LongWord.SetUInt( $00000100); 1464
14398: AMD_DE_FLAG', 'LongWord.SetUInt( $00000004); 1458
14399: AMD_FPU_FLAG', 'LongWord.SetUInt( $00000001); 1456
14400: AMD_MCE_FLAG', 'LongWord.SetUInt( $00000080); 1463
14401: AMD_MMX_FLAG', 'LongWord.SetUInt( $00800000); 1479
14402: AMD_MSR_FLAG', 'LongWord.SetUInt( $00000020); 1461
14403: AMD_MTRR_FLAG', 'LongWord.SetUInt( $00001000); 1468
14404: AMD_PGE_FLAG', 'LongWord.SetUInt( $00002000); 1469
14405: AMD_PSE_FLAG', 'LongWord.SetUInt( $00000008); 1459
14406: AMD_TSC_FLAG', 'LongWord.SetUInt( $00000010); 1460
14407: AMD_VME_FLAG', 'LongWord.SetUInt( $00000002); 1457
14408: ANSI_CHARSET', 'LongInt.SetInt( 0); 169
14409: ANTI_ALIAS_QUALITY', 'LongInt.SetInt( 4); 164
14410: Const('AnsiNull', 'Char.SetString( AnsiChar ( #0 ));
14411: AnsiSoh', 'Char.SetString( AnsiChar ( #1 ));
14412: AnsiStx', 'Char.SetString( AnsiChar ( #2 ));
14413: AnsiEtx', 'Char.SetString( AnsiChar ( #3 ));
14414: AnsiEot', 'Char.SetString( AnsiChar ( #4 ));
14415: AnsiEnq', 'Char.SetString( AnsiChar ( #5 ));
14416: AnsiAck', 'Char.SetString( AnsiChar ( #6 ));
14417: AnsiBell', 'Char.SetString( AnsiChar ( #7 ));
14418: AnsiBackspace', 'Char.SetString( AnsiChar ( #8 ));
14419: AnsiTab', 'Char.SetString( AnsiChar ( #9 ));
14420: AnsiVerticalTab', 'Char.SetString( AnsiChar ( #11 ));
14421: AnsiFormFeed', 'Char.SetString( AnsiChar ( #12 ));
14422: AnsiSo', 'Char.SetString( AnsiChar ( #14 ));
14423: AnsiSi', 'Char.SetString( AnsiChar ( #15 ));
14424: AnsiIdle', 'Char.SetString( AnsiChar ( #16 ));
14425: AnsiDc1', 'Char.SetString( AnsiChar ( #17 ));
14426: AnsiDc2', 'Char.SetString( AnsiChar ( #18 ));
14427: AnsiDc3', 'Char.SetString( AnsiChar ( #19 ));
14428: AnsiDc4', 'Char.SetString( AnsiChar ( #20 ));
14429: AnsiNak', 'Char.SetString( AnsiChar ( #21 ));
14430: AnsiSyn', 'Char.SetString( AnsiChar ( #22 ));
14431: AnsiEtb', 'Char.SetString( AnsiChar ( #23 ));
14432: AnsiCan', 'Char.SetString( AnsiChar ( #24 ));
14433: AnsiEm', 'Char.SetString( AnsiChar ( #25 ));
14434: AnsiEndOfFile', 'Char.SetString( AnsiChar ( #26 ));
14435: AnsiEscape', 'Char.SetString( AnsiChar ( #27 ));
14436: AnsiFs', 'Char.SetString( AnsiChar ( #28 ));
14437: AnsiGs', 'Char.SetString( AnsiChar ( #29 ));
14438: AnsiRs', 'Char.SetString( AnsiChar ( #30 ));
14439: AnsiUs', 'Char.SetString( AnsiChar ( #31 ));
14440: AnsiSpace', 'String.SetString( AnsiChar ( ' ' ));
14441: AnsiComma', 'String.SetString( AnsiChar ( ',' ));
14442: AnsiBackslash', 'String.SetString( AnsiChar ( '\ ' ));
14443: AnsiForwardSlash', 'String.SetString( AnsiChar ( '/' ));
14444: AnsiDoubleQuote', 'String.SetString( AnsiChar ( '"' ));
14445: AnsiSingleQuote', 'String.SetString( AnsiChar ( "'" ));
14446: AnsiWhiteSpace', 'LongInt.Value.ts32 := ord(AnsiTab) or ord(AnsiLineFeed) or ord(AnsiVerticalTab) or
ord(AnsiFormFeed) or ord(AnsiCarriageReturn) or ord(AnsiSpace);
14447: //Const('AnsiSigns', 'Set.SetSet(['-', '+']);
14448: C1_UPPER', 'LongWord.SetUInt( $0001);
14449: C1_LOWER', 'LongWord.SetUInt( $0002);
14450: C1_DIGIT', 'LongWord.SetUInt( $0004);
14451: C1_SPACE', 'LongWord.SetUInt( $0008);
14452: C1_PUNCT', 'LongWord.SetUInt( $0010);
14453: C1_CNTRL', 'LongWord.SetUInt( $0020);
14454: C1_BLANK', 'LongWord.SetUInt( $0040);
14455: C1_XDIGIT', 'LongWord.SetUInt( $0080);
14456: C1_ALPHA', 'LongWord.SetUInt( $0100);
14457: AnsiChar', 'Char;
14458: APIC_FLAG', 'LongWord.SetUInt( $00000200); 1433
14459: APPCOMMAND_BASS_BOOST', 'LongInt.SetInt( 20); 2608
14460: APPCOMMAND_BASS_BOOST', 'LongInt.SetInt( 20); 2807
14461: APPCOMMAND_BASS_DOWN', 'LongInt.SetInt( 19); 2607
14462: APPCOMMAND_BASS_DOWN', 'LongInt.SetInt( 19); 2806
14463: APPCOMMAND_BASS_UP', 'LongInt.SetInt( 21); 2609
14464: APPCOMMAND_BASS_UP', 'LongInt.SetInt( 21); 2808
14465: APPCOMMAND_BROWSER_BACKWARD', 'LongInt.SetInt( 1); 2589
14466: APPCOMMAND_BROWSER_BACKWARD', 'LongInt.SetInt( 1); 2788
14467: APPCOMMAND_BROWSER_FAVORITES', 'LongInt.SetInt( 6); 2594

```



```

14468: APPCOMMAND_BROWSER_FAVORITES', 'LongInt.SetInt( 6); 2793
14469: APPCOMMAND_BROWSER_FORWARD', 'LongInt.SetInt( 2); 2590
14470: APPCOMMAND_BROWSER_FORWARD', 'LongInt.SetInt( 2); 2789
14471: APPCOMMAND_BROWSER_HOME', 'LongInt.SetInt( 7); 2595
14472: APPCOMMAND_BROWSER_HOME', 'LongInt.SetInt( 7); 2794
14473: APPCOMMAND_BROWSER_REFRESH', 'LongInt.SetInt( 3); 2591
14474: APPCOMMAND_BROWSER_REFRESH', 'LongInt.SetInt( 3); 2790
14475: APPCOMMAND_BROWSER_SEARCH', 'LongInt.SetInt( 5); 2593
14476: APPCOMMAND_BROWSER_SEARCH', 'LongInt.SetInt( 5); 2792
14477: APPCOMMAND_BROWSER_STOP', 'LongInt.SetInt( 4); 2592
14478: APPCOMMAND_BROWSER_STOP', 'LongInt.SetInt( 4); 2791
14479: APPCOMMAND_CLOSE', 'LongInt.SetInt( 31); 2619
14480: APPCOMMAND_CLOSE', 'LongInt.SetInt( 31); 2818
14481: APPCOMMAND_COPY', 'LongInt.SetInt( 36); 2624
14482: APPCOMMAND_COPY', 'LongInt.SetInt( 36); 2823
14483: APPCOMMAND_CORRECTION_LIST', 'LongInt.SetInt( 45); 2633
14484: APPCOMMAND_CORRECTION_LIST', 'LongInt.SetInt( 45); 2832
14485: APPCOMMAND_CUT', 'LongInt.SetInt( 37); 2625
14486: APPCOMMAND_CUT', 'LongInt.SetInt( 37); 2824
14487: APPCOMMAND_DICTATE_OR_COMMAND_CONTROL_TOGGLE', 'LongInt.SetInt( 43); 2631
14488: APPCOMMAND_DICTATE_OR_COMMAND_CONTROL_TOGGLE', 'LongInt.SetInt( 43); 2830
14489: APPCOMMAND_FIND', 'LongInt.SetInt( 28); 2616
14490: APPCOMMAND_FIND', 'LongInt.SetInt( 28); 2815
14491: APPCOMMAND_FORWARD_MAIL', 'LongInt.SetInt( 40); 2628
14492: APPCOMMAND_FORWARD_MAIL', 'LongInt.SetInt( 40); 2827
14493: APPCOMMAND_HELP', 'LongInt.SetInt( 27); 2615
14494: APPCOMMAND_HELP', 'LongInt.SetInt( 27); 2814
14495: APPCOMMAND_LAUNCH_APP1', 'LongInt.SetInt( 17); 2605
14496: APPCOMMAND_LAUNCH_APP1', 'LongInt.SetInt( 17); 2804
14497: APPCOMMAND_LAUNCH_APP2', 'LongInt.SetInt( 18); 2606
14498: APPCOMMAND_LAUNCH_APP2', 'LongInt.SetInt( 18); 2805
14499: APPCOMMAND_LAUNCH_MAIL', 'LongInt.SetInt( 15); 2603
14500: APPCOMMAND_LAUNCH_MAIL', 'LongInt.SetInt( 15); 2802
14501: APPCOMMAND_LAUNCH_MEDIA_SELECT', 'LongInt.SetInt( 16); 2604
14502: APPCOMMAND_LAUNCH_MEDIA_SELECT', 'LongInt.SetInt( 16); 2803
14503: APPCOMMAND_MEDIA_CHANNEL_DOWN', 'LongInt.SetInt( 52); 2640
14504: APPCOMMAND_MEDIA_CHANNEL_DOWN', 'LongInt.SetInt( 52); 2839
14505: APPCOMMAND_MEDIA_CHANNEL_UP', 'LongInt.SetInt( 51); 2639
14506: APPCOMMAND_MEDIA_CHANNEL_UP', 'LongInt.SetInt( 51); 2838
14507: APPCOMMAND_MEDIA_FAST_FORWARD', 'LongInt.SetInt( 49); 2637
14508: APPCOMMAND_MEDIA_FAST_FORWARD', 'LongInt.SetInt( 49); 2836
14509: APPCOMMAND_MEDIA_NEXTTRACK', 'LongInt.SetInt( 11); 2599
14510: APPCOMMAND_MEDIA_NEXTTRACK', 'LongInt.SetInt( 11); 2798
14511: APPCOMMAND_MEDIA_PAUSE', 'LongInt.SetInt( 47); 2635
14512: APPCOMMAND_MEDIA_PAUSE', 'LongInt.SetInt( 47); 2834
14513: APPCOMMAND_MEDIA_PLAY', 'LongInt.SetInt( 46); 2634
14514: APPCOMMAND_MEDIA_PLAY', 'LongInt.SetInt( 46); 2833
14515: APPCOMMAND_MEDIA_PLAY_PAUSE', 'LongInt.SetInt( 14); 2602
14516: APPCOMMAND_MEDIA_PLAY_PAUSE', 'LongInt.SetInt( 14); 2801
14517: APPCOMMAND_MEDIA_PREVIOUS_TRACK', 'LongInt.SetInt( 12); 2600
14518: APPCOMMAND_MEDIA_PREVIOUS_TRACK', 'LongInt.SetInt( 12); 2799
14519: APPCOMMAND_MEDIA_RECORD', 'LongInt.SetInt( 48); 2636
14520: APPCOMMAND_MEDIA_RECORD', 'LongInt.SetInt( 48); 2835
14521: APPCOMMAND_MEDIA_REWIND', 'LongInt.SetInt( 50); 2638
14522: APPCOMMAND_MEDIA_REWIND', 'LongInt.SetInt( 50); 2837
14523: APPCOMMAND_MEDIA_STOP', 'LongInt.SetInt( 13); 2601
14524: APPCOMMAND_MEDIA_STOP', 'LongInt.SetInt( 13); 2800
14525: APPCOMMAND_MIC_ON_OFF_TOGGLE', 'LongInt.SetInt( 44); 2632
14526: APPCOMMAND_MIC_ON_OFF_TOGGLE', 'LongInt.SetInt( 44); 2831
14527: APPCOMMAND_MICROPHONE_VOLUME_DOWN', 'LongInt.SetInt( 25); 2613
14528: APPCOMMAND_MICROPHONE_VOLUME_DOWN', 'LongInt.SetInt( 25); 2812
14529: APPCOMMAND_MICROPHONE_VOLUME_MUTE', 'LongInt.SetInt( 24); 2612
14530: APPCOMMAND_MICROPHONE_VOLUME_MUTE', 'LongInt.SetInt( 24); 2811
14531: APPCOMMAND_MICROPHONE_VOLUME_UP', 'LongInt.SetInt( 26); 2614
14532: APPCOMMAND_MICROPHONE_VOLUME_UP', 'LongInt.SetInt( 26); 2813
14533: APPCOMMAND_NEW', 'LongInt.SetInt( 29); 2617
14534: APPCOMMAND_NEW', 'LongInt.SetInt( 29); 2816
14535: APPCOMMAND_OPEN', 'LongInt.SetInt( 30); 2618
14536: APPCOMMAND_OPEN', 'LongInt.SetInt( 30); 2817
14537: APPCOMMAND_PASTE', 'LongInt.SetInt( 38); 2626
14538: APPCOMMAND_PASTE', 'LongInt.SetInt( 38); 2825
14539: APPCOMMAND_PRINT', 'LongInt.SetInt( 33); 2621
14540: APPCOMMAND_PRINT', 'LongInt.SetInt( 33); 2820
14541: APPCOMMAND_REDO', 'LongInt.SetInt( 35); 2623
14542: APPCOMMAND_REDO', 'LongInt.SetInt( 35); 2822
14543: APPCOMMAND_REPLY_TO_MAIL', 'LongInt.SetInt( 39); 2627
14544: APPCOMMAND_REPLY_TO_MAIL', 'LongInt.SetInt( 39); 2826
14545: APPCOMMAND_SAVE', 'LongInt.SetInt( 32); 2620
14546: APPCOMMAND_SAVE', 'LongInt.SetInt( 32); 2819
14547: APPCOMMAND_SEND_MAIL', 'LongInt.SetInt( 41); 2629
14548: APPCOMMAND_SEND_MAIL', 'LongInt.SetInt( 41); 2828
14549: APPCOMMAND_SPELL_CHECK', 'LongInt.SetInt( 42); 2630
14550: APPCOMMAND_SPELL_CHECK', 'LongInt.SetInt( 42); 2829
14551: APPCOMMAND_TREBLE_DOWN', 'LongInt.SetInt( 22); 2610
14552: APPCOMMAND_TREBLE_DOWN', 'LongInt.SetInt( 22); 2809
14553: APPCOMMAND_TREBLE_UP', 'LongInt.SetInt( 23); 2611
14554: APPCOMMAND_TREBLE_UP', 'LongInt.SetInt( 23); 2810
14555: APPCOMMAND_UNDO', 'LongInt.SetInt( 34); 2622
14556: APPCOMMAND_UNDO', 'LongInt.SetInt( 34); 2821

```

```

14557: APPCOMMAND_VOLUME_DOWN', 'LongInt.SetInt( 9); 2597
14558: APPCOMMAND_VOLUME_DOWN', 'LongInt.SetInt( 9); 2796
14559: APPCOMMAND_VOLUME_MUTE', 'LongInt.SetInt( 8); 2596
14560: APPCOMMAND_VOLUME_MUTE', 'LongInt.SetInt( 8); 2795
14561: APPCOMMAND_VOLUME_UP', 'LongInt.SetInt( 10); 2598
14562: APPCOMMAND_VOLUME_UP', 'LongInt.SetInt( 10); 2797
14563: AppInstCmdLineDataKind', 'LongInt.SetInt( 1); 2447
14564: AppInstDataKindNoData', 'LongInt.SetInt( - 1); 2446
14565: ArcMinutesPerDeg', 'Extended.setExtended( 60.0); 2586
14566: ArcSecondsPerArcMinute', 'Extended.setExtended( 60.0); 2587
14567: AREAINFO_ABSOLUTE', 'LongWord.SetUInt( $10000000); 2397
14568: AREAINFO_ELLIPSE', 'LongWord.SetUInt( $20000000); 2396
14569: AREAINFO_LINE', 'LongWord.SetUInt( $40000000); 2395
14570: AREAINFO_MASK', 'LongWord.SetUInt( $FF000000); 2398
14571: AREAINFO_RECT', 'LongWord.SetUInt( $80000000); 2394
14572: ARRSIZE', 'LongInt.SetInt( 171); 1898
14573: ASN1_COUNTER', 'LongWord.SetUInt( $41); 1051
14574: ASN1_GAUGE', 'LongWord.SetUInt( $42); 1052
14575: ASN1_INT', 'LongWord.SetUInt( $02); 1045
14576: ASN1_IPADDR', 'LongWord.SetUInt( $40); 1050
14577: ASN1_NULL', 'LongWord.SetUInt( $05); 1047
14578: ASN1_OBJID', 'LongWord.SetUInt( $06); 1048
14579: ASN1_OCTSTR', 'LongWord.SetUInt( $04); 1046
14580: ASN1_OPAQUE', 'LongWord.SetUInt( $44); 1054
14581: ASN1_SEQ', 'LongWord.SetUInt( $30); 1049
14582: ASN1_TIMETICKS', 'LongWord.SetUInt( $43); 1053
14583: ASSOCDATA_HASPERUSERASSOC', 'LongInt.SetInt( 4); 1874
14584: ASSOCDATA_MAX', 'LongInt.SetInt( 5); 1875
14585: ASSOCDATA_MSIDESCRIPTOR', 'LongInt.SetInt( 1); 1871
14586: ASSOCDATA_NOACTIVATEHANDLER', 'LongInt.SetInt( 2); 1872
14587: ASSOCDATA_QUERYCLASSSTORE', 'LongInt.SetInt( 3); 1873
14588: ASSOCENUM_NONE', 'LongInt.SetInt( 0); 1876
14589: ASSOCF_IGNOREBASECLASS', 'LongWord.SetUInt( $00000200); 1853
14590: ASSOCF_INIT_BYEXENAME', 'LongWord.SetUInt( $00000002); 1844
14591: ASSOCF_INIT_DEFAULTTOFOLDER', 'LongWord.SetUInt( $00000008); 1847
14592: ASSOCF_INIT_DEFAULTTOSTAR', 'LongWord.SetUInt( $00000004); 1846
14593: ASSOCF_INIT_NOREMAPCLSID', 'LongWord.SetUInt( $00000001); 1843
14594: ASSOCF_NOFIXUPS', 'LongWord.SetUInt( $00000100); 1852
14595: ASSOCF_NOTRUNCATE', 'LongWord.SetUInt( $00000020); 1849
14596: ASSOCF_NOUSERSETTINGS', 'LongWord.SetUInt( $00000010); 1848
14597: ASSOCF_OPEN_BYEXENAME', 'LongWord.SetUInt( $00000002); 1845
14598: ASSOCF_REMAPPRUNDLL', 'LongWord.SetUInt( $00000080); 1851
14599: ASSOCF_VERIFY', 'LongWord.SetUInt( $00000040); 1850
14600: AssociationsCount', 'LongInt.SetInt( 7); 3393
14601: ASSOCKEY_APP', 'LongInt.SetInt( 2); 1867
14602: ASSOCKEY_BASECLASS', 'LongInt.SetInt( 4); 1869
14603: ASSOCKEY_CLASS', 'LongInt.SetInt( 3); 1868
14604: ASSOCKEY_MAX', 'LongInt.SetInt( 5); 1870
14605: ASSOCKEY_SHELLEXECCLASS', 'LongInt.SetInt( 1); 1866
14606: ASSOCSTR_COMMAND', 'LongInt.SetInt( 1); 1854
14607: ASSOCSTR_DDEAPPLICATION', 'LongInt.SetInt( 9); 1862
14608: ASSOCSTR_DDECOMMAND', 'LongInt.SetInt( 7); 1860
14609: ASSOCSTR_DDEIFEXEC', 'LongInt.SetInt( 8); 1861
14610: ASSOCSTR_DDETOPIC', 'LongInt.SetInt( 10); 1863
14611: ASSOCSTR_EXECUTABLE', 'LongInt.SetInt( 2); 1855
14612: ASSOCSTR_FRIENDLYAPPNAME', 'LongInt.SetInt( 4); 1857
14613: ASSOCSTR_FRIENDLYDOCNAME', 'LongInt.SetInt( 3); 1856
14614: ASSOCSTR_INFOTIP', 'LongInt.SetInt( 11); 1864
14615: ASSOCSTR_MAX', 'LongInt.SetInt( 12); 1865
14616: ASSOCSTR_NOOPEN', 'LongInt.SetInt( 5); 1858
14617: ASSOCSTR_SHELLNEWVALUE', 'LongInt.SetInt( 6); 1859
14618: AW_ACTIVATE', 'LongWord.SetUInt( $00020000); 415
14619: AW_BLEND', 'LongWord.SetUInt( $00080000); 417
14620: AW_CENTER', 'LongWord.SetUInt( $00000010); 413
14621: AW_HIDE', 'LongWord.SetUInt( $00010000); 414
14622: AW_HOR_NEGATIVE', 'LongWord.SetUInt( $00000002); 410
14623: AW_HOR_POSITIVE', 'LongWord.SetUInt( $00000001); 409
14624: AW_SLIDE', 'LongWord.SetUInt( $00040000); 416
14625: AW_VER_NEGATIVE', 'LongWord.SetUInt( $00000008); 412
14626: AW_VER_POSITIVE', 'LongWord.SetUInt( $00000004); 411
14627: b64_MIMEBase64', 'String.SetString( 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/; 2229
14628: b64_UUEncode', 'String.SetString( ' !"#%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_` 2230
14629: b64_XXEncode', 'String.SetString( '+-0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz 2231
14630: BACKSPACE', 'Char.SetString( #8); 1072
14631: BadDate', 'LongWord.SetUInt( LongInt ( $FFFFFFFF )); 2997
14632: BadTime', 'LongWord.SetUInt( LongInt ( $FFFFFFFF )); 3001
14633: Base', 'LongInt.SetInt( 10); 1720
14634: bcDefNarrowToWideRatio', 'LongInt.SetInt( 2); 2980
14635: BcdSize', 'LongInt.SetInt( 10); 1931
14636: bcGuardBarAbove', 'Boolean.SetBool( True); 2978
14637: bcGuardBarBelow', 'Boolean.BoolToStr( True); 2979
14638: bcMaxBarCodeLen', 'LongInt.SetInt( 255); 2977
14639: Bernstein', 'Extended.SetExtended( 0.2801694990238691330364364912307); 1667
14640: Bernstein', 'Float.SetString( 0.2801694990238691330364364912307); 1279
14641: Bernstein', 'Float.SetString( 0.2801694990238691330364364912307); 1347
14642: BIF_BROWSEINCLUDEFILES', 'LongWord.SetUInt( $4000); 2660
14643: BIF_BROWSEINCLUDEURLS', 'LongWord.SetUInt( $0080); 2657
14644: BIF_EDITBOX', 'LongWord.SetUInt( $0010); 2654
14645: BIF_NEWDIALOGSTYLE', 'LongWord.SetUInt( $0040); 2656

```

```

14646: BIF_NONNEWFOLDERBUTTON', 'LongWord.SetUInt( $0200); 2659
14647: BIF_RETURNFSANCESTORS', 'LongWord.SetUInt( $0008); 2653
14648: BIF_SHAREABLE', 'LongWord.SetUInt( $8000); 2661
14649: BIF_UAHINT', 'LongWord.SetUInt( $0100); 2658
14650: BIF_VALIDATE', 'LongWord.SetUInt( $0020); 2655
14651: BIT_10', 'LongWord.SetUInt( $00000400); 1434
14652: BIT_18', 'LongWord.SetUInt( $00040000); 1442
14653: BIT_19', 'LongWord.SetUInt( $00080000); 1443
14654: BIT_20', 'LongWord.SetUInt( $00100000); 1444
14655: BIT_21', 'LongWord.SetUInt( $00200000); 1445
14656: BIT_22', 'LongWord.SetUInt( $00400000); 1446
14657: BIT_25', 'LongWord.SetUInt( $02000000); 1449
14658: BIT_26', 'LongWord.SetUInt( $04000000); 1450
14659: BIT_27', 'LongWord.SetUInt( $08000000); 1451
14660: BIT_28', 'LongWord.SetUInt( $10000000); 1452
14661: BIT_29', 'LongWord.SetUInt( $20000000); 1453
14662: BIT_30', 'LongWord.SetUInt( $40000000); 1454
14663: BIT_31', 'LongWord.SetUInt( DWORD ( $80000000 )); 1455
14664: BitsPerByte', 'LongInt.SetInt( 8); 2184
14665: BitsPerByte', 'LongInt.SetInt( 8); 3322
14666: BitsPerCardinal', 'LongInt.SetInt( BytesPerCardinal * 8); 2211
14667: BitsPerDigit', 'LongInt.SetInt( 32); 3323
14668: BitsPerLongWord', 'LongInt.SetInt( 32); 2186
14669: BitsPerMIDDataByte', 'LongInt.SetInt( 7); 2552
14670: BitsPerMIDDataWord', 'LongInt.SetInt( BitsPerMIDDataByte * 2); 2553
14671: BitsPerNativeWord', 'LongInt.SetInt( BytesPerNativeWord * 8); 2212
14672: BitsPerWord', 'LongInt.SetInt( 16); 2185
14673: BLACKNESS', 'LongWord.SetUInt( $00000042); 797
14674: BOOTSCRIPT', 'String.SetString( 'maxbootscript.txt; 967
14675: BOOTSCRIPT', 'String.SetString( 'maxbootscript.txt; 943
14676: BorderWidth', 'LongInt.SetInt( 0); 3139
14677: BorderWidth', 'LongInt.SetInt( 2); 3138
14678: BoundPortDefault', 'LongInt.SetInt( 0); 9
14679: br_000050', 'LongInt.SetInt( 50); 1740
14680: br_000110', 'LongInt.SetInt( 110); 1741
14681: br_000150', 'LongInt.SetInt( 150); 1742
14682: br_000300', 'LongInt.SetInt( 300); 1743
14683: br_000600', 'LongInt.SetInt( 600); 1744
14684: br_001200', 'LongInt.SetInt( 1200); 1745
14685: br_002400', 'LongInt.SetInt( 2400); 1746
14686: br_004800', 'LongInt.SetInt( 4800); 1747
14687: br_009600', 'LongInt.SetInt( 9600); 1748
14688: br_014400', 'LongInt.SetInt( 14400); 1749
14689: br_019200', 'LongInt.SetInt( 19200); 1750
14690: br_028800', 'LongInt.SetInt( 28800); 1751
14691: br_038400', 'LongInt.SetInt( 38400); 1752
14692: br_056000', 'LongInt.SetInt( 56000); 1753
14693: br_057600', 'LongInt.SetInt( 57600); 1754
14694: br_115200', 'LongInt.SetInt( 115200); 1755
14695: br_128000', 'LongInt.SetInt( 128000); 1756
14696: br_230400', 'LongInt.SetInt( 230400); 1757
14697: br_256000', 'LongInt.SetInt( 256000); 1758
14698: br_460800', 'LongInt.SetInt( 460800); 1759
14699: br_921600', 'LongInt.SetInt( 921600); 1760
14700: Brackets', 'String.SetString( '(' or ')' or '[' or ']' or '{' or '}'; 1623
14701: BufferSize', 'LongInt.SetInt( 64); 1119
14702: BYTECODE', 'String.SetString( 'bytecode.psb; 954
14703: BYTECODE', 'String.SetString( 'bytecode.txt; 930
14704: Bytes1KB', 'LongInt.SetInt(1024); 2235
14705: BytesPerCardinal', 'LongInt.SetInt(Cardb); 2187
14706: BytesPerNativeWord', 'LongInt.SetInt(MaxNativeInt); 2188
14707: C3_ALPHA', 'LongWord.SetUInt( $8000); 3151
14708: C3_DIACRITIC', 'LongInt.SetInt( 2); 3141
14709: C3_FULLWIDTH', 'LongWord.SetUInt( $0080); 3147
14710: C3_HALFWIDTH', 'LongWord.SetUInt( $0040); 3146
14711: C3_HIRAGANA', 'LongWord.SetUInt( $0020); 3145
14712: C3_IDEOGRAPH', 'LongWord.SetUInt( $0100); 3148
14713: C3_KASHIDA', 'LongWord.SetUInt( $0200); 3149
14714: C3_KATAKANA', 'LongWord.SetUInt( $0010); 3144
14715: C3_LEXICAL', 'LongWord.SetUInt( $0400); 3150
14716: C3_NONSPACING', 'LongInt.SetInt( 1); 3140
14717: C3_NOTAPPLICABLE', 'LongInt.SetInt( 0); 3152
14718: C3_SYMBOL', 'LongInt.SetInt( 8); 3143
14719: C3_VOWELMARK', 'LongInt.SetInt( 4); 3142
14720: CANEXPRSIZE', 'LongInt.SetInt( 10); 885
14721: CANEXPRVERSION', 'LongInt.SetInt( 2); 887
14722: CANHDRSIZE', 'LongInt.SetInt( 8); 886
14723: Catalan', 'Extended.SetExtended( 0.9159655941772190150546035149324); 1673
14724: Catalan', 'Float.SetString( 0.9159655941772190150546035149324); 1285
14725: Catalan', 'Float.SetString( 0.9159655941772190150546035149324); 1353
14726: CATID_SafeForInitializing', 'TGUID.SetString( '{7DD95802-9882-11CF-9FA9-00AA006C42C4}; 3395
14727: CATID_SafeForScripting', 'TGUID.SetString( '{7DD95801-9882-11CF-9FA9-00AA006C42C4}; 3396
14728: CBinMax', 'LongInt.SetInt( 256); 2556
14729: Chrt10', 'Extended.SetExtended( 2.1544346900318837217592935665194); 1670
14730: Chrt10', 'Float.SetString( 2.1544346900318837217592935665194); 1282
14731: Chrt10', 'Float.SetString( 2.1544346900318837217592935665194); 1350
14732: Chrt100', 'Extended.SetExtended( 4.6415888336127788924100763509194); 1671
14733: Chrt100', 'Float.SetString( 4.6415888336127788924100763509194); 1283
14734: Chrt100', 'Float.SetString( 4.6415888336127788924100763509194); 1351

```

```
14735: Cbrt2', 'Extended.SetExtended( 1.2599210498948731647672106072782); 1668
14736: Cbrt2', 'Float.SetString( 1.2599210498948731647672106072782); 1280
14737: Cbrt2', 'Float.SetString( 1.2599210498948731647672106072782); 1348
14738: Cbrt3', 'Extended.SetExtended( 1.4422495703074083823216383107801); 1669
14739: Cbrt3', 'Float.SetString( 1.4422495703074083823216383107801); 1281
14740: Cbrt3', 'Float.SetString( 1.4422495703074083823216383107801); 1349
14741: CbrtPi', 'Extended.SetExtended( 1.4645918875615232630201425272638); 1672
14742: CbrtPi', 'Float.SetString( 1.4645918875615232630201425272638); 1284
14743: CbrtPi', 'Float.SetString( 1.4645918875615232630201425272638); 1352
14744: CCHARSET', 'String.SetString( '+-0123456789ABCDEFGHIJKLMNQRSTUUVWXYZabcdefghijklmnopqrstuvwxyz; 2232
14745: CCHARSET', 'String.SetString(b64_XXEncode); 2233
14746: CCHBinName', 'LongInt.SetInt( 24); 2554
14747: CCHPaperName', 'LongInt.SetInt( 64); 2555
14748: CelsiusAbsoluteZero', 'Extended.setExtended( - 273.15); 2571
14749: CelsiusFreezingPoint', 'Extended.setExtended( 0.0); 2568
14750: CenturyOffset', 'Byte.SetUInt( 60); 1588
14751: CenturyOffset', 'Byte.SetUInt( 60); 1601
14752: CenturyOffset', 'Byte.SetUInt( 60); 1618
14753: CError_AsyncCheck', 'LongInt.SetInt( 6); 851
14754: CError_ClearComFailed', 'LongInt.SetInt( 10); 855
14755: CError_ConnChangeProp', 'LongInt.SetInt( 14); 859
14756: CError_EnumPortsFailed', 'LongInt.SetInt( 15); 860
14757: CError_EscapeComFailed', 'LongInt.SetInt( 12); 857
14758: CError_HasLink', 'LongInt.SetInt( 22); 867
14759: CError_InvalidAsync', 'LongInt.SetInt( 4); 849
14760: CError_LedStateFailed', 'LongInt.SetInt( 19); 864
14761: CError_LoadFailed', 'LongInt.SetInt( 17); 862
14762: CError_ModemStatFailed', 'LongInt.SetInt( 11); 856
14763: CError_OpenFailed', 'LongInt.SetInt( 1); 846
14764: CError_PortNotOpen', 'LongInt.SetInt( 24); 869
14765: CError_PurgeFailed', 'LongInt.SetInt( 5); 850
14766: CError_ReadFailed', 'LongInt.SetInt( 3); 848
14767: CError_RegError', 'LongInt.SetInt( 23); 868
14768: CError_RegFailed', 'LongInt.SetInt( 18); 863
14769: CError_SetStateFailed', 'LongInt.SetInt( 7); 852
14770: CError_SetupComFailed', 'LongInt.SetInt( 9); 854
14771: CError_StoreFailed', 'LongInt.SetInt( 16); 861
14772: CError_ThreadCreated', 'LongInt.SetInt( 20); 865
14773: CError_TimeoutsFailed', 'LongInt.SetInt( 8); 853
14774: CError_TransmitFailed', 'LongInt.SetInt( 13); 858
14775: CError_WaitFailed', 'LongInt.SetInt( 21); 866
14776: CError_WriteFailed', 'LongInt.SetInt( 2); 847
14777: CF_BITMAP', 'LongInt.SetInt( 2); 255
14778: CF_DIB', 'LongInt.SetInt( 8); 261
14779: CF_DIBV5', 'LongInt.SetInt( 17); 271
14780: CF_DIF', 'LongInt.SetInt( 5); 258
14781: CF_DSPBITMAP', 'LongInt.SetInt( 130); 275
14782: CF_DSPENHMETAFILE', 'LongInt.SetInt( 142); 277
14783: CF_DSPMETAFILEPICT', 'LongInt.SetInt( 131); 276
14784: CF_DSPTTEXT', 'LongInt.SetInt( 129); 274
14785: CF_ENHMETAFILE', 'LongInt.SetInt( 14); 267
14786: CF_GDIOBJFIRST', 'LongInt.SetInt( 768); 280
14787: CF_GDIOBJLAST', 'LongInt.SetInt( 1023); 281
14788: CF_HDROP', 'LongInt.SetInt( 15); 268
14789: CF_LOCALE', 'LongWord.SetUInt( $10); 269
14790: CF_MAX', 'LongInt.SetInt( 17); 270
14791: CF_MAX_XP', 'LongInt.SetInt( 18); 272
14792: CF_METAFILEPICT', 'LongInt.SetInt( 3); 256
14793: CF_OEMTEXT', 'LongInt.SetInt( 7); 260
14794: CF_OWNERDISPLAY', 'LongInt.SetInt( 128); 273
14795: CF_PALETTE', 'LongInt.SetInt( 9); 262
14796: CF_PENDATA', 'LongInt.SetInt( 10); 263
14797: CF_PRIVATEFIRST', 'LongWord.SetUInt( $200); 278
14798: CF_PRIVATELAST', 'LongInt.SetInt( 767); 279
14799: CF_RIFF', 'LongInt.SetInt( 11); 264
14800: CF_SYLK', 'LongInt.SetInt( 4); 257
14801: CF_TEXT', 'LongInt.SetInt( 1); 254
14802: CF_TIFF', 'LongInt.SetInt( 6); 259
14803: CF_UNICODETEXT', 'LongInt.SetInt( 13); 266
14804: CF_WAVE', 'LongInt.SetInt( 12); 265
14805: CGold', 'Extended.setExtended( 0.38196601125010515179); 3214
14806: CHAR0', 'Char.SetString( #0); 1071
14807: CHAR32', 'Char.SetString( #32); 1077
14808: CHINESEBIG5_CHARSET', 'LongInt.SetInt( 136); 175
14809: CIllegalConvFamily', 'TConvFamily.SetString( '0; 844
14810: CIllegalConvType', 'TConvType.SetString( '0; 845
14811: clAliceBlue32', 'LongWord.SetUInt( TColor32 ( $FFF0F8FF )); 2256
14812: clAntiqueWhite32', 'LongWord.SetUInt( TColor32 ( $FFFAEBD7 )); 2257
14813: clAqua32', 'LongWord.SetUInt( TColor32 ( $FF00FFFF )); 1273
14814: clAqua32', 'LongWord.SetUInt( TColor32 ( $FF00FFFF )); 2255
14815: clAquamarine32', 'LongWord.SetUInt( TColor32 ( $FF7FFFD4 )); 2258
14816: clAzure32', 'LongWord.SetUInt( TColor32 ( $FFFFFF00 )); 2259
14817: clBeige32', 'LongWord.SetUInt( TColor32 ( $FFF5F5DC )); 2260
14818: clBisque32', 'LongWord.SetUInt( TColor32 ( $FFFFE4C4 )); 2261
14819: clBlack32', 'LongWord.SetUInt( TColor32 ( $FF000000 )); 1257
14820: clBlack32', 'LongWord.SetUInt( TColor32 ( $FF000000 )); 2239
14821: clBlancheDalmond32', 'LongWord.SetUInt( TColor32 ( $FFFFFFBCD )); 2262
14822: clBlue32', 'LongWord.SetUInt( TColor32 ( $FF0000FF )); 1271
14823: clBlue32', 'LongWord.SetUInt( TColor32 ( $FF0000FF )); 2253
```



```

14824: clBlueViolet32', 'LongWord.SetUInt( TColor32 ( $FF8A2BE2 )); 2263
14825: clBrown32', 'LongWord.SetUInt( TColor32 ( $FFA52A2A )); 2264
14826: clBurlyWood32', 'LongWord.SetUInt( TColor32 ( $FFDEB887 )); 2265
14827: clCadetBlue32', 'LongWord.SetUInt( TColor32 ( $FF5F9EA0 )); 2266
14828: clChartReuse32', 'LongWord.SetUInt( TColor32 ( $FF7FFF00 )); 2267
14829: clChocolate32', 'LongWord.SetUInt( TColor32 ( $FFD2691E )); 2268
14830: clCoral32', 'LongWord.SetUInt( TColor32 ( $FFFF7F50 )); 2269
14831: clCornFlowerBlue32', 'LongWord.SetUInt( TColor32 ( $FF6495ED )); 2270
14832: clCornSilk32', 'LongWord.SetUInt( TColor32 ( $FFFFFF8DC )); 2271
14833: clCream', 'LongWord.SetUInt( TColor32 ( $A6CAFA )); 1628
14834: clCream32', 'LongWord.SetUInt( TColor32 ( $FFFFFFB0 )); 2332
14835: clCrimson32', 'LongWord.SetUInt( TColor32 ( $FFDC143C )); 2272
14836: clDarkBlue32', 'LongWord.SetUInt( TColor32 ( $FF00008B )); 2273
14837: clDarkCyan32', 'LongWord.SetUInt( TColor32 ( $FF008B8B )); 2274
14838: clDarkGoldenRod32', 'LongWord.SetUInt( TColor32 ( $FFB8860B )); 2275
14839: clDarkGray32', 'LongWord.SetUInt( TColor32 ( $FFA9A9A9 )); 2276
14840: clDarkGreen32', 'LongWord.SetUInt( TColor32 ( $FF006400 )); 2277
14841: clDarkGrey32', 'LongWord.SetUInt( TColor32 ( $FFA9A9A9 )); 2278
14842: clDarkKhaki32', 'LongWord.SetUInt( TColor32 ( $FFBDB76B )); 2279
14843: clDarkMagenta32', 'LongWord.SetUInt( TColor32 ( $FF8B008B )); 2280
14844: clDarkOliveGreen32', 'LongWord.SetUInt( TColor32 ( $FF556B2F )); 2281
14845: clDarkOrange32', 'LongWord.SetUInt( TColor32 ( $FFFF8C00 )); 2282
14846: clDarkOrchid32', 'LongWord.SetUInt( TColor32 ( $FF9932CC )); 2283
14847: clDarkRed32', 'LongWord.SetUInt( TColor32 ( $FF8B0000 )); 2284
14848: clDarkSalmon32', 'LongWord.SetUInt( TColor32 ( $FFE9967A )); 2285
14849: clDarkSeaGreen32', 'LongWord.SetUInt( TColor32 ( $FF8FBC8F )); 2286
14850: clDarkSlateBlue32', 'LongWord.SetUInt( TColor32 ( $FF483D8B )); 2287
14851: clDarkSlateGray32', 'LongWord.SetUInt( TColor32 ( $FF2F4F4F )); 2288
14852: clDarkSlateGrey32', 'LongWord.SetUInt( TColor32 ( $FF2F4F4F )); 2289
14853: clDarkTurquoise32', 'LongWord.SetUInt( TColor32 ( $FF00CED1 )); 2290
14854: clDarkViolet32', 'LongWord.SetUInt( TColor32 ( $FF9400D3 )); 2291
14855: clDeepPink32', 'LongWord.SetUInt( TColor32 ( $FFFFFF1493 )); 2292
14856: clDeepSkyBlue32', 'LongWord.SetUInt( TColor32 ( $FF00BFFF )); 2293
14857: clDimGray32', 'LongWord.SetUInt( TColor32 ( $FF3F3F3F )); 1258
14858: clDimGray32', 'LongWord.SetUInt( TColor32 ( $FF3F3F3F )); 2240
14859: clDkGray32', 'LongWord.SetUInt( TColor32 ( $FF808080 )); 2329
14860: clDodgerBlue32', 'LongWord.SetUInt( TColor32 ( $FF1E90FF )); 2294
14861: clFireBrick32', 'LongWord.SetUInt( TColor32 ( $FFB22222 )); 2295
14862: clFloralWhite32', 'LongWord.SetUInt( TColor32 ( $FFFFFFAF0 )); 2296
14863: clFuchsia32', 'LongWord.SetUInt( TColor32 ( $FFFF00FF )); 1272
14864: clFuchsia32', 'LongWord.SetUInt( TColor32 ( $FFFF00FF )); 2254
14865: clGainsboro32', 'LongWord.SetUInt( TColor32 ( $FFDCDCDC )); 2297
14866: clGhostWhite32', 'LongWord.SetUInt( TColor32 ( $FFF8F8FF )); 2298
14867: clGold32', 'LongWord.SetUInt( TColor32 ( $FFFFD700 )); 2299
14868: clGoldenRod32', 'LongWord.SetUInt( TColor32 ( $FFDAA520 )); 2300
14869: clGray32', 'LongWord.SetUInt( TColor32 ( $FF7F7F7F )); 1259
14870: clGray32', 'LongWord.SetUInt( TColor32 ( $FF7F7F7F )); 2241
14871: clGreen32', 'LongWord.SetUInt( TColor32 ( $FF007F00 )); 1263
14872: clGreen32', 'LongWord.SetUInt( TColor32 ( $FF007F00 )); 2245
14873: clGreenYellow32', 'LongWord.SetUInt( TColor32 ( $FFADFF2F )); 2301
14874: clGrey32', 'LongWord.SetUInt( TColor32 ( $FF808080 )); 2302
14875: clHoneyDew32', 'LongWord.SetUInt( TColor32 ( $FFFF0FFF )); 2303
14876: clHotPink32', 'LongWord.SetUInt( TColor32 ( $FFFF69B4 )); 2304
14877: clIndianRed32', 'LongWord.SetUInt( TColor32 ( $FFCD5C5C )); 2305
14878: clIndigo32', 'LongWord.SetUInt( TColor32 ( $FF4B0082 )); 2306
14879: clInfoBk', 'LongWord.SetUInt( TColor ( $02E1FFFF )); 1626
14880: clIvory32', 'LongWord.SetUInt( TColor32 ( $FFFFFFF0 )); 2307
14881: clKhaki32', 'LongWord.SetUInt( TColor32 ( $FFFF068C )); 2308
14882: clLavender32', 'LongWord.SetUInt( TColor32 ( $FFE6E6FA )); 2309
14883: clLavenderBlush32', 'LongWord.SetUInt( TColor32 ( $FFFFFF0F5 )); 2310
14884: clLawnGreen32', 'LongWord.SetUInt( TColor32 ( $FF7CFC00 )); 2311
14885: clLegacySkyBlue32', 'LongWord.SetUInt( TColor32 ( $FFA6CAFA )); 2331
14886: clLemonChiffon32', 'LongWord.SetUInt( TColor32 ( $FFFFFFACD )); 2312
14887: clLightBlue32', 'LongWord.SetUInt( TColor32 ( $FFADD8E6 )); 2313
14888: clLightCoral32', 'LongWord.SetUInt( TColor32 ( $FFF08080 )); 2314
14889: clLightCyan32', 'LongWord.SetUInt( TColor32 ( $FFE0FFFF )); 2315
14890: clLightGoldenRodYellow32', 'LongWord.SetUInt( TColor32 ( $FFFAFAD2 )); 2316
14891: clLightGray32', 'LongWord.SetUInt( TColor32 ( $FFBFBFBF )); 1260
14892: clLightGray32', 'LongWord.SetUInt( TColor32 ( $FFBFBFBF )); 2242
14893: clLightGreen32', 'LongWord.SetUInt( TColor32 ( $FF90EE90 )); 2317
14894: clLightGrey32', 'LongWord.SetUInt( TColor32 ( $FFD3D3D3 )); 2318
14895: clLightPink32', 'LongWord.SetUInt( TColor32 ( $FFFB6C1 )); 2319
14896: clLightSalmon32', 'LongWord.SetUInt( TColor32 ( $FFFA07A )); 2320
14897: clLightSeagreen32', 'LongWord.SetUInt( TColor32 ( $FF20B2AA )); 2321
14898: clLightSkyblue32', 'LongWord.SetUInt( TColor32 ( $FF87CEFA )); 2322
14899: clLightSlategray32', 'LongWord.SetUInt( TColor32 ( $FF778899 )); 2323
14900: clLightSlategrey32', 'LongWord.SetUInt( TColor32 ( $FF778899 )); 2324
14901: clLightSteelblue32', 'LongWord.SetUInt( TColor32 ( $FFB0C4DE )); 2325
14902: clLightYellow32', 'LongWord.SetUInt( TColor32 ( $FFFFFFE0 )); 2326
14903: clLime32', 'LongWord.SetUInt( TColor32 ( $FF00FF00 )); 1269
14904: clLime32', 'LongWord.SetUInt( TColor32 ( $FF00FF00 )); 2251
14905: clLimeGreen32', 'LongWord.SetUInt( TColor32 ( $FF32CD32 )); 2333
14906: clLinen32', 'LongWord.SetUInt( TColor32 ( $FFFAF0E6 )); 2334
14907: clLtGray32', 'LongWord.SetUInt( TColor32 ( $FFC0C0C0 )); 2327
14908: clMaroon32', 'LongWord.SetUInt( TColor32 ( $FF7F0000 )); 1262
14909: clMaroon32', 'LongWord.SetUInt( TColor32 ( $FF7F0000 )); 2244
14910: clMedGray32', 'LongWord.SetUInt( TColor32 ( $FFA0A0A4 )); 2328
14911: clMediumAquamarine32', 'LongWord.SetUInt( TColor32 ( $FF66CDAA )); 2335
14912: clMediumBlue32', 'LongWord.SetUInt( TColor32 ( $FF0000CD )); 2336

```

```

14913: clMediumOrchid32', 'LongWord.SetUInt( TColor32 ( $FFBA55D3 )); 2337
14914: clMediumPurple32', 'LongWord.SetUInt( TColor32 ( $FFB )); 2338
14915: clMediumSeaGreen32', 'LongWord.SetUInt( TColor32 ( $FF3CB371 )); 2339
14916: clMediumSlateBlue32', 'LongWord.SetUInt( TColor32 ( $FF7B68EE )); 2340
14917: clMediumSpringGreen32', 'LongWord.SetUInt( TColor32 ( $FF00FA9A )); 2341
14918: clMediumTurquoise32', 'LongWord.SetUInt( TColor32 ( $FF48D1CC )); 2342
14919: clMediumVioletRed32', 'LongWord.SetUInt( TColor32 ( $FFC71585 )); 2343
14920: clMidnightBlue32', 'LongWord.SetUInt( TColor32 ( $FF191970 )); 2344
14921: clMintCream32', 'LongWord.SetUInt( TColor32 ( $FFFF5FFFA )); 2345
14922: clMistyRose32', 'LongWord.SetUInt( TColor32 ( $FFFFE4E1 )); 2346
14923: clMoccasin32', 'LongWord.SetUInt( TColor32 ( $FFFFE4B5 )); 2347
14924: clMoneyGreen', 'LongWord.SetUInt( TColor32 ( $C0DCC0 )); 1629
14925: clMoneyGreen32', 'LongWord.SetUInt( TColor32 ( $FFC0DCC0 )); 2330
14926: clNavajoWhite32', 'LongWord.SetUInt( TColor32 ( $FFFFDEAD )); 2348
14927: clNavy32', 'LongWord.SetUInt( TColor32 ( $FF00007F )); 1265
14928: clNavy32', 'LongWord.SetUInt( TColor32 ( $FF00007F )); 2247
14929: clNone', 'LongWord.SetUInt( TColor32 ( $02FFFFFF )); 1627
14930: clOldLace32', 'LongWord.SetUInt( TColor32 ( $FFFD5E6 )); 2349
14931: clOlive32', 'LongWord.SetUInt( TColor32 ( $FF7F7F00 )); 1264
14932: clOlive32', 'LongWord.SetUInt( TColor32 ( $FF7F7F00 )); 2246
14933: clOliveDrab32', 'LongWord.SetUInt( TColor32 ( $FF6B8E23 )); 2350
14934: clOlive32', 'LongWord.SetUInt( TColor32 ( $FFFA500 )); 2351
14935: clOrangeRed32', 'LongWord.SetUInt( TColor32 ( $FFFA500 )); 2352
14936: clOrchid32', 'LongWord.SetUInt( TColor32 ( $FFDA70D6 )); 2353
14937: clPaleGoldenRod32', 'LongWord.SetUInt( TColor32 ( $FFEEB8AA )); 2354
14938: clPaleGreen32', 'LongWord.SetUInt( TColor32 ( $FF98FB98 )); 2355
14939: clPaleTurquoise32', 'LongWord.SetUInt( TColor32 ( $FFAFEEEE )); 2356
14940: clPaleVioletRed32', 'LongWord.SetUInt( TColor32 ( $FFDB7093 )); 2357
14941: clPapayaWhip32', 'LongWord.SetUInt( TColor32 ( $FFFFEFD5 )); 2358
14942: clPeachPuff32', 'LongWord.SetUInt( TColor32 ( $FFFFDAB9 )); 2359
14943: clPeru32', 'LongWord.SetUInt( TColor32 ( $FFCD853F )); 2360
14944: clPlum32', 'LongWord.SetUInt( TColor32 ( $FFDDA0DD )); 2361
14945: clPowderBlue32', 'LongWord.SetUInt( TColor32 ( $FFB0E0E6 )); 2362
14946: clPurple32', 'LongWord.SetUInt( TColor32 ( $FF7F007F )); 1266
14947: clPurple32', 'LongWord.SetUInt( TColor32 ( $FF7F007F )); 2248
14948: clRed32', 'LongWord.SetUInt( TColor32 ( $FFFF0000 )); 1268
14949: clRed32', 'LongWord.SetUInt( TColor32 ( $FFFF0000 )); 2250
14950: clRosyBrown32', 'LongWord.SetUInt( TColor32 ( $FFBC8F8F )); 2363
14951: clRoyalBlue32', 'LongWord.SetUInt( TColor32 ( $FF4169E1 )); 2364
14952: clSaddleBrown32', 'LongWord.SetUInt( TColor32 ( $FF8B4513 )); 2365
14953: clSalmon32', 'LongWord.SetUInt( TColor32 ( $FFFA8072 )); 2366
14954: clSandyBrown32', 'LongWord.SetUInt( TColor32 ( $FFFA4A60 )); 2367
14955: clSeaGreen32', 'LongWord.SetUInt( TColor32 ( $FF2E8B57 )); 2368
14956: clSeaShell32', 'LongWord.SetUInt( TColor32 ( $FFFFFF5EE )); 2369
14957: CLSID_StdComponentCategoriesMgr', 'TGUID.SetString( '{0002E005-0000-0000-C000-000000000046}; 3394
14958: clSienna32', 'LongWord.SetUInt( TColor32 ( $FFA0522D )); 2370
14959: clSilver32', 'LongWord.SetUInt( TColor32 ( $FFC0C0C0 )); 2371
14960: clSkyBlue', 'LongWord.SetUInt( TColor32 ( $FFBF0 )); 1630
14961: clSkyblue32', 'LongWord.SetUInt( TColor32 ( $FF87CEEB )); 2372
14962: clSlateBlue32', 'LongWord.SetUInt( TColor32 ( $FF6A5ACD )); 2373
14963: clSlateGray32', 'LongWord.SetUInt( TColor32 ( $FF708090 )); 2374
14964: clSlateGrey32', 'LongWord.SetUInt( TColor32 ( $FF708090 )); 2375
14965: clSnow32', 'LongWord.SetUInt( TColor32 ( $FFFFFFAFA )); 2376
14966: clSpringgreen32', 'LongWord.SetUInt( TColor32 ( $FF00FF7F )); 2377
14967: clSteelblue32', 'LongWord.SetUInt( TColor32 ( $FF4682B4 )); 2378
14968: clTan32', 'LongWord.SetUInt( TColor32 ( $FFD2B48C )); 2379
14969: clTeal32', 'LongWord.SetUInt( TColor32 ( $FF007F7F )); 1267
14970: clTeal32', 'LongWord.SetUInt( TColor32 ( $FF007F7F )); 2249
14971: clThistle32', 'LongWord.SetUInt( TColor32 ( $FFD8BFD8 )); 2380
14972: clTomato32', 'LongWord.SetUInt( TColor32 ( $FFF6347 )); 2381
14973: clTrBlack32', 'LongWord.SetUInt( TColor32 ( $7F000000 )); 1275
14974: clTrBlack32', 'LongWord.SetUInt( TColor32 ( $7F000000 )); 2388
14975: clTrBlue32', 'LongWord.SetUInt( TColor32 ( $7F0000FF )); 1278
14976: clTrBlue32', 'LongWord.SetUInt( TColor32 ( $7F0000FF )); 2391
14977: clTrGreen32', 'LongWord.SetUInt( TColor32 ( $7F00FF00 )); 1277
14978: clTrGreen32', 'LongWord.SetUInt( TColor32 ( $7F00FF00 )); 2390
14979: clTrRed32', 'LongWord.SetUInt( TColor32 ( $7FFF0000 )); 1276
14980: clTrRed32', 'LongWord.SetUInt( TColor32 ( $7FFF0000 )); 2389
14981: clTrWhite32', 'LongWord.SetUInt( TColor32 ( $7FFFFFFF )); 1274
14982: clTrWhite32', 'LongWord.SetUInt( TColor32 ( $7FFFFFFF )); 2387
14983: clTurquoise32', 'LongWord.SetUInt( TColor32 ( $FF40E0D0 )); 2382
14984: clViolet32', 'LongWord.SetUInt( TColor32 ( $FFEE82EE )); 2383
14985: clWebAliceBlue', 'LongWord.SetUInt( $FFF8F0 ); 652
14986: clWebAntiqueWhite', 'LongWord.SetUInt( $D7EBFA ); 650
14987: clWebAqua', 'LongWord.SetUInt( $FFFF00 ); 732
14988: clWebAquamarine', 'LongWord.SetUInt( $D4FF7F ); 701
14989: clWebAzure', 'LongWord.SetUInt( $FFFFF0 ); 661
14990: clWebBeige', 'LongWord.SetUInt( $DCFF5F ); 649
14991: clWebBisque', 'LongWord.SetUInt( $C4E4FF ); 667
14992: clWebBlack', 'LongWord.SetUInt( $000000 ); 782
14993: clWebBlanchedAlmond', 'LongWord.SetUInt( $CDEBFF ); 666
14994: clWebBlue', 'LongWord.SetUInt( $FF0000 ); 744
14995: clWebBlueViolet', 'LongWord.SetUInt( $E22B8A ); 753
14996: clWebBrown', 'LongWord.SetUInt( $2A2AA5 ); 681
14997: clWebBurlywood', 'LongWord.SetUInt( $87B8DE ); 660
14998: clWebCadetBlue', 'LongWord.SetUInt( $A09E5F ); 739
14999: clWebChartreuse', 'LongWord.SetUInt( $00FF7F ); 699
15000: clWebChocolate', 'LongWord.SetUInt( $1E69D2 ); 682
15001: clWebCoral', 'LongWord.SetUInt( $507FFF ); 677

```

```

15002: clWebCornFlowerBlue', 'LongWord.SetUInt( $ED9564); 726
15003: clWebCornSilk', 'LongWord.SetUInt( $DCF8FF); 648
15004: clWebCrimson', 'LongWord.SetUInt( $3C14DC); 680
15005: clWebCyan', 'LongWord.SetUInt( $FFFF00); 731
15006: clWebDarkBlue', 'LongWord.SetUInt( $8B0000); 727
15007: clWebDarkCyan', 'LongWord.SetUInt( $8B8B00); 740
15008: clWebDarkGoldenRod', 'LongWord.SetUInt( $0B86B8); 712
15009: clWebDarkGray', 'LongWord.SetUInt( $A9A9A9); 781
15010: clWebDarkGreen', 'LongWord.SetUInt( $006400); 715
15011: clWebDarkKhaki', 'LongWord.SetUInt( $6BB7BD); 713
15012: clWebDarkMagenta', 'LongWord.SetUInt( $8B008B); 750
15013: clWebDarkOliveGreen', 'LongWord.SetUInt( $2F6B55); 714
15014: clWebDarkOrange', 'LongWord.SetUInt( $008CFF); 671
15015: clWebDarkOrchid', 'LongWord.SetUInt( $CC3299); 747
15016: clWebDarkRed', 'LongWord.SetUInt( $00008B); 673
15017: clWebDarkSalmon', 'LongWord.SetUInt( $7A96E9); 692
15018: clWebDarkSeaGreen', 'LongWord.SetUInt( $8FBC8F); 720
15019: clWebDarkSlateBlue', 'LongWord.SetUInt( $8B3D48); 778
15020: clWebDarkSlateGray', 'LongWord.SetUInt( $4F4F2F); 779
15021: clWebDarkTurquoise', 'LongWord.SetUInt( $D1CE00); 738
15022: clWebDarkViolet', 'LongWord.SetUInt( $D30094); 746
15023: clWebDeepPink', 'LongWord.SetUInt( $9314FF); 757
15024: clWebDeepSkyBlue', 'LongWord.SetUInt( $FFBF00); 742
15025: clWebDimGray', 'LongWord.SetUInt( $696969); 776
15026: clWebDodgerBlue', 'LongWord.SetUInt( $FF901E); 743
15027: clWebFirebrick', 'LongWord.SetUInt( $222B2B); 688
15028: clWebFloralWhite', 'LongWord.SetUInt( $F0FAFF); 644
15029: clWebForestGreen', 'LongWord.SetUInt( $228B22); 697
15030: clWebFuchsia', 'LongWord.SetUInt( $FF00FF); 749
15031: clWebGainsboro', 'LongWord.SetUInt( $DCDCDC); 780
15032: clWebGhostWhite', 'LongWord.SetUInt( $FFF8F8); 653
15033: clWebGold', 'LongWord.SetUInt( $00D7FF); 678
15034: clWebGoldenRod', 'LongWord.SetUInt( $20A5DA); 703
15035: clWebGray', 'LongWord.SetUInt( $808080); 770
15036: clWebGreen', 'LongWord.SetUInt( $008000); 706
15037: clWebGreenYellow', 'LongWord.SetUInt( $2FFFAD); 698
15038: clWebHoneydew', 'LongWord.SetUInt( $F0FFF0); 663
15039: clWebHotPink', 'LongWord.SetUInt( $B469FF); 763
15040: clWebIndianRed', 'LongWord.SetUInt( $5C5CCD); 675
15041: clWebIndigo', 'LongWord.SetUInt( $82004B); 728
15042: clWebIvory', 'LongWord.SetUInt( $F0FFFF); 647
15043: clWebKhaki', 'LongWord.SetUInt( $8CE6F0); 704
15044: clWebLavender', 'LongWord.SetUInt( $FAE6E6); 654
15045: clWebLavenderBlush', 'LongWord.SetUInt( $F5F0FF); 645
15046: clWebLawnGreen', 'LongWord.SetUInt( $00FFC7); 708
15047: clWebLemonChiffon', 'LongWord.SetUInt( $CDFAFF); 665
15048: clWebLightBlue', 'LongWord.SetUInt( $E6D8AD); 724
15049: clWebLightCoral', 'LongWord.SetUInt( $8080F0); 685
15050: clWebLightCyan', 'LongWord.SetUInt( $FFFFE0); 723
15051: clWebLightGoldenrodYellow', 'LongWord.SetUInt( $D2FAFA); 695
15052: clWebLightGreen', 'LongWord.SetUInt( $90EE90); 700
15053: clWebLightgrey', 'LongWord.SetUInt( $D3D3D3); 769
15054: clWebLightPink', 'LongWord.SetUInt( $C1B6FF); 758
15055: clWebLightSalmon', 'LongWord.SetUInt( $7AA0FF); 684
15056: clWebLightSeaGreen', 'LongWord.SetUInt( $AAB220); 721
15057: clWebLightSkyBlue', 'LongWord.SetUInt( $FACE87); 725
15058: clWebLightSlateGray', 'LongWord.SetUInt( $998877); 767
15059: clWebLightSteelBlue', 'LongWord.SetUInt( $DEC4B0); 765
15060: clWebLightYellow', 'LongWord.SetUInt( $E0FFFF); 656
15061: clWebLime', 'LongWord.SetUInt( $00FF00); 717
15062: clWebLimeGreen', 'LongWord.SetUInt( $32CD32); 716
15063: clWebLinen', 'LongWord.SetUInt( $E6F0FA); 664
15064: clWebMagenta', 'LongWord.SetUInt( $FF00FF); 748
15065: clWebMaroon', 'LongWord.SetUInt( $000080); 674
15066: clWebMediumAquaMarine', 'LongWord.SetUInt( $AACD66); 710
15067: clWebMediumBlue', 'LongWord.SetUInt( $CD0000); 736
15068: clWebMediumOrchid', 'LongWord.SetUInt( $D355BA); 754
15069: clWebMediumPurple', 'LongWord.SetUInt( $DB7093); 755
15070: clWebMediumSeaGreen', 'LongWord.SetUInt( $71B33C); 711
15071: clWebMediumSlateBlue', 'LongWord.SetUInt( $EE687B); 766
15072: clWebMediumSpringGreen', 'LongWord.SetUInt( $9AFA00); 719
15073: clWebMediumTurquoise', 'LongWord.SetUInt( $CCD148); 729
15074: clWebMediumVioletRed', 'LongWord.SetUInt( $8515C7); 751
15075: clWebMidnightBlue', 'LongWord.SetUInt( $701919); 737
15076: clWebMintcream', 'LongWord.SetUInt( $FAFFFF); 662
15077: clWebMistyRose', 'LongWord.SetUInt( $E1E4FF); 777
15078: clWebMoccasin', 'LongWord.SetUInt( $B5E4FF); 659
15079: clWebNavajoWhite', 'LongWord.SetUInt( $ADD8E6); 658
15080: clWebNavy', 'LongWord.SetUInt( $000000); 745
15081: clWebOldLace', 'LongWord.SetUInt( $E6F5FD); 646
15082: clWebOlive', 'LongWord.SetUInt( $008080); 696
15083: clWebOliveDrab', 'LongWord.SetUInt( $238E6B); 705
15084: clWebOrange', 'LongWord.SetUInt( $00A5FF); 686
15085: clWebOrangeRed', 'LongWord.SetUInt( $0045FF); 687
15086: clWebOrchid', 'LongWord.SetUInt( $D670DA); 760
15087: clWebPaleGoldenrod', 'LongWord.SetUInt( $AAE8EE); 694
15088: clWebPaleGreen', 'LongWord.SetUInt( $98FB98); 709
15089: clWebPaleTurquoise', 'LongWord.SetUInt( $EEEEAF); 722
15090: clWebPaleVioletRed', 'LongWord.SetUInt( $9370DB); 752

```

```

15091: clWebPapayaWhip', 'LongWord.SetUInt( $D5EFFF); 657
15092: clWebPeachPuff', 'LongWord.SetUInt( $B9DAFF); 668
15093: clWebPeru', 'LongWord.SetUInt( $3F85CD); 691
15094: clWebPink', 'LongWord.SetUInt( $CBC0FF); 764
15095: clWebPlum', 'LongWord.SetUInt( $DDA0DD); 761
15096: clWebPowderBlue', 'LongWord.SetUInt( $E6E0B0); 733
15097: clWebPurple', 'LongWord.SetUInt( $800080); 756
15098: clWebRed', 'LongWord.SetUInt( $0000FF); 672
15099: clWebRosyBrown', 'LongWord.SetUInt( $8F8FBC); 693
15100: clWebRoyalBlue', 'LongWord.SetUInt( $E16941); 735
15101: clWebSaddleBrown', 'LongWord.SetUInt( $13458B); 689
15102: clWebSalmon', 'LongWord.SetUInt( $7280FA); 676
15103: clWebSandyBrown', 'LongWord.SetUInt( $60A4F4); 683
15104: clWebSeaGreen', 'LongWord.SetUInt( $578B2E); 702
15105: clWebSeashell', 'LongWord.SetUInt( $EEF5FF); 655
15106: clWebSienna', 'LongWord.SetUInt( $2D52A0); 690
15107: clWebSilver', 'LongWord.SetUInt( $C0C0C0); 775
15108: clWebSkyBlue', 'LongWord.SetUInt( $EBCE87); 734
15109: clWebSlateBlue', 'LongWord.SetUInt( $CD5A6A); 772
15110: clWebSlateGray', 'LongWord.SetUInt( $908070); 773
15111: clWebSnow', 'LongWord.SetUInt( $FAFAFF); 643
15112: clWebSpringGreen', 'LongWord.SetUInt( $7FFF00); 718
15113: clWebSteelBlue', 'LongWord.SetUInt( $B48246); 771
15114: clWebTan', 'LongWord.SetUInt( $8CB4D2); 669
15115: clWebTeal', 'LongWord.SetUInt( $808000); 741
15116: clWebThistle', 'LongWord.SetUInt( $8BBD8); 762
15117: clWebTomato', 'LongWord.SetUInt( $4763FF); 679
15118: clWebTurquoise', 'LongWord.SetUInt( $D0E040); 730
15119: clWebViolet', 'LongWord.SetUInt( $EE82EE); 759
15120: clWebWheat', 'LongWord.SetUInt( $B3DEF5); 651
15121: clWebWhite', 'LongWord.SetUInt( $FFFFFF); 768
15122: clWebWhiteSmoke', 'LongWord.SetUInt( $F5F5F5); 774
15123: clWebYellow', 'LongWord.SetUInt( $00FFFF); 670
15124: clWebYellowGreen', 'LongWord.SetUInt( $32CD9A); 707
15125: clWheat32', 'LongWord.SetUInt( TColor32 ( $FFFF5DEB3 )); 2384
15126: clWhite32', 'LongWord.SetUInt( TColor32 ( $FFFFFFF )); 1261
15127: clWhite32', 'LongWord.SetUInt( TColor32 ( $FFFFFFF )); 2243
15128: clWhitesmoke32', 'LongWord.SetUInt( TColor32 ( $FFF5F5F5 )); 2385
15129: clYellow32', 'LongWord.SetUInt( TColor32 ( $FFFFFFF0 )); 1270
15130: clYellow32', 'LongWord.SetUInt( TColor32 ( $FFFFFFF0 )); 2252
15131: clYellowgreen32', 'LongWord.SetUInt( TColor32 ( $FF9ACD32 )); 2386
15132: CM_DEFERFREER', 'LongWord.SetUInt( WM_USER + $0002); 1736
15133: cm_DeferLayout', 'LongInt.SetInt( WM_USER + 100); 892
15134: CM_LOOKUPCOMPLETE', 'LongWord.SetUInt( WM_USER + $0003); 1737
15135: CM_SOCKETMESSAGE', 'LongWord.SetUInt( WM_USER + $0001); 1735
15136: cmAll', 'LongInt.Value.ts32 := ord(cfmVirtual) or ord(cfmPersistent) or ord(cfmSession); 908
15137: cmBlackness', 'LongWord.SetUInt(BLACKNESS); 799
15138: cmDstInvert', 'LongWord.SetUInt(DSTINVERT); 800
15139: cmMergeCopy', 'LongWord.SetUInt(MERGECOPY); 801
15140: cmMergePaint', 'LongWord.SetUInt(MERGEPAINT); 802
15141: cmNotSrcCopy', 'LongWord.SetUInt(NOTSRCCOPY); 803
15142: cmNotSrcErase', 'LongWord.SetUInt(NOTSRCERASE); 804
15143: CMOV_FLAG', 'LongWord.SetUInt( $00008000); 1439
15144: cmPatCopy', 'LongWord.SetUInt(PATCOPY); 805
15145: cmPatInvert', 'LongWord.SetUInt(PATINVERT); 806
15146: cmPatPaint', 'LongWord.SetUInt(PATPAINT); 807
15147: cmPersistent', 'LongInt.Value.ts32 := ord(cfmPersistent); 906
15148: cmSession', 'LongInt.Value.ts32 := ord(cfmSession); 907
15149: cmSrcAnd', 'LongWord.SetUInt(SRCAND); 808
15150: cmSrcCopy', 'LongWord.SetUInt(SRCCOPY); 809
15151: cmSrcErase', 'LongWord.SetUInt(SRCERASE); 810
15152: cmSrcInvert', 'LongWord.SetUInt(SRCINVERT); 811
15153: cmSrcPaint', 'LongWord.SetUInt(SRCPAINT); 812
15154: cmVirtual', 'LongInt.Value.ts32 := ord(cfmVirtual); 905
15155: cmWhiteness', 'LongWord.SetUInt(WHITENESS); 813
15156: CN_BANDCHANGE', 'LongWord.SetUInt( WM_USER + $1000); 837
15157: CN_DROPDOWNCLOSED', 'LongWord.SetUInt( WM_USER + $1000); 836
15158: CODECOMPLETION', 'String.SetString( 'bds_delphi.dci'; 976
15159: CODECOMPLETION', 'String.SetString( 'bds_delphi.dci'; 953
15160: ColCount', 'LongInt.SetInt( 20); 1606
15161: ColumnHeaderWidth', 'SetString( LVSCW_AUTOSIZE_USEHEADER); 834
15162: ColumnTextWidth', 'SetString( LVSCW_AUTOSIZE); 835
15163: comctl32', 'String.SetString( 'comctl32.dll; 12
15164: ComCtlVersionIE3', 'LongWord.SetUInt( $00040046); 838
15165: ComCtlVersionIE4', 'LongWord.SetUInt( $00040047); 839
15166: ComCtlVersionIE401', 'LongWord.SetUInt( $00040048); 840
15167: ComCtlVersionIE5', 'LongWord.SetUInt( $00050050); 841
15168: ComCtlVersionIE501', 'LongWord.SetUInt( $00050051); 842
15169: ComCtlVersionIE6', 'LongWord.SetUInt( $00060000); 843
15170: CPaperNames', 'LongInt.SetInt( 256); 2557
15171: CPU_TYPE_AMD', 'LongInt.SetInt( 3); 1422
15172: CPU_TYPE_CRUSOE', 'LongInt.SetInt( 4); 1423
15173: CPU_TYPE_CYRIX', 'LongInt.SetInt( 2); 1421
15174: CPU_TYPE_INTEL', 'LongInt.SetInt( 1); 1420
15175: CPUi386', 'LongInt.SetInt( 2); //3.7 364
15176: CPUi386', 'LongInt.SetInt( 2); 2117
15177: CPUi486', 'LongInt.SetInt( 3); 2118
15178: CPUi486', 'LongInt.SetInt( 3); 365
15179: CPUPentium', 'LongInt.SetInt( 4); 2119

```



```

15180: CPUPentium', 'LongInt.SetInt( 4); 366
15181: CR', 'Char.SetString( #13); 1041
15182: CR', 'Char.SetString( #13); 1074
15183: Crc16Bits', 'LongInt.SetInt( 16); 1336
15184: Crc16Bits', 'LongInt.SetInt( 16); 1404
15185: Crc16Bytes', 'LongInt.SetInt( 2); 1337
15186: Crc16Bytes', 'LongInt.SetInt( 2); 1405
15187: Crc16HighBit', 'LongWord.SetUInt( $8000); 1338
15188: Crc16HighBit', 'LongWord.SetUInt( $8000); 1406
15189: Crc16PolynomCCITT', 'LongWord.SetUInt( $1021); 1334
15190: Crc16PolynomCCITT', 'LongWord.SetUInt( $1021); 1402
15191: Crc16PolynomIBM', 'LongWord.SetUInt( $8005); 1335
15192: Crc16PolynomIBM', 'LongWord.SetUInt( $8005); 1403
15193: Crc32Bits', 'LongInt.SetInt( 32); 1343
15194: Crc32Bits', 'LongInt.SetInt( 32); 1411
15195: Crc32Bytes', 'LongInt.SetInt( 4); 1344
15196: Crc32Bytes', 'LongInt.SetInt( 4); 1412
15197: Crc32HighBit', 'LongWord.SetUInt( $80000000); 1345
15198: Crc32HighBit', 'LongWord.SetUInt( $80000000); 1413
15199: Crc32Koopman', 'LongWord.SetUInt( $741B8CD7); 1342
15200: Crc32Koopman', 'LongWord.SetUInt( $741B8CD7); 1410
15201: Crc32PolynomCastagnoli', 'LongWord.SetUInt( $1EDC6F41); 1341
15202: Crc32PolynomCastagnoli', 'LongWord.SetUInt( $1EDC6F41); 1409
15203: Crc32PolynomIEEE', 'LongWord.SetUInt( $04C11DB7); 1340
15204: Crc32PolynomIEEE', 'LongWord.SetUInt( $04C11DB7); 1408
15205: CrcBufSize', 'LongInt.SetInt( 2048); 2002
15206: CRLF', 'Char.SetString( #10); 1040
15207: CRLF', 'Char.SetString( #10); 3175
15208: CRLF', 'Char.SetString( #10); 3364
15209: CRLF', 'Char.SetString( #13 + #10); 1039
15210: crlf', 'String.SetString( #13#10); 1132
15211: crPanAll', 'LongInt.SetInt( TCursor ( - 50 )); 1238
15212: crPanDown', 'LongInt.SetInt( TCursor ( - 51 )); 1239
15213: crPanDownLeft', 'LongInt.SetInt( TCursor ( - 52 )); 1240
15214: crPanDownRight', 'LongInt.SetInt( TCursor ( - 53 )); 1241
15215: crPanLeft', 'LongInt.SetInt( TCursor ( - 54 )); 1242
15216: crPanLeftRight', 'LongInt.SetInt( TCursor ( - 55 )); 1243
15217: crPanRight', 'LongInt.SetInt( TCursor ( - 56 )); 1244
15218: crPanUp', 'LongInt.SetInt( TCursor ( - 57 )); 1245
15219: crPanUpDown', 'LongInt.SetInt( TCursor ( - 58 )); 1246
15220: crPanUpLeft', 'LongInt.SetInt( TCursor ( - 59 )); 1247
15221: crPanUpRight', 'LongInt.SetInt( TCursor ( - 60 )); 1248
15222: CS_DROP_SHADOW', 'LongWord.SetUInt( $00020000); 2758
15223: csEOLN', 'String.SetString( #13#10); 1721
15224: CSIDL_ADMIN_TOOLS', 'LongWord.SetUInt( $0030); 2680
15225: CSIDL_CDBURN_AREA', 'LongWord.SetUInt( $003B); 2688
15226: CSIDL_COMMON_ADMIN_TOOLS', 'LongWord.SetUInt( $002F); 2679
15227: CSIDL_COMMON_APPDATA', 'LongWord.SetUInt( $0023); 2670
15228: CSIDL_COMMON_DOCUMENTS', 'LongWord.SetUInt( $002E); 2678
15229: CSIDL_COMMON_DOCUMENTS', 'LongWord.SetUInt( $002e); 2968
15230: CSIDL_COMMON_MUSIC', 'LongWord.SetUInt( $0035); 2682
15231: CSIDL_COMMON_OEM_LINKS', 'LongWord.SetUInt( $003A); 2687
15232: CSIDL_COMMON_PICTURES', 'LongWord.SetUInt( $0036); 2683
15233: CSIDL_COMMON_TEMPLATES', 'LongWord.SetUInt( $002D); 2677
15234: CSIDL_COMMON_VIDEO', 'LongWord.SetUInt( $0037); 2684
15235: CSIDL_COMPUTERSNEARME', 'LongWord.SetUInt( $003D); 2689
15236: CSIDL_CONNECTIONS', 'LongWord.SetUInt( $0031); 2681
15237: CSIDL_DESKTOPDIRECTORY', 'LongWord.SetUInt( $0010); 2690
15238: CSIDL_LOCAL_APPDATA', 'LongWord.SetUInt( $001C); 2669
15239: CSIDL_MYDOCUMENTS', 'LongWord.SetUInt( $000C); 2666
15240: CSIDL_MYMUSIC', 'LongWord.SetUInt( $000D); 2667
15241: CSIDL_MYPICTURES', 'LongWord.SetUInt( $0027); 2674
15242: CSIDL_MYVIDEO', 'LongWord.SetUInt( $000E); 2668
15243: CSIDL_PROFILE', 'LongWord.SetUInt( $0028); 2675
15244: CSIDL_PROFILES', 'LongWord.SetUInt( $003E); 2691
15245: CSIDL_PROGRAM_FILES', 'LongWord.SetUInt( $0026); 2673
15246: CSIDL_PROGRAM_FILES_COMMON', 'LongWord.SetUInt( $002B); 2676
15247: CSIDL_RESOURCES', 'LongWord.SetUInt( $0038); 2685
15248: CSIDL_RESOURCES_LOCALIZED', 'LongWord.SetUInt( $0039); 2686
15249: CSIDL_SYSTEM', 'LongWord.SetUInt( $0025); 2672
15250: CSIDL_WINDOWS', 'LongWord.SetUInt( $0024); 2671
15251: CTF_COINIT', 'LongWord.SetUInt( $00000008); 1891
15252: CTF_INSIST', 'LongWord.SetUInt( $00000001); 1888
15253: CTF_PROCESS_REF', 'LongWord.SetUInt( $00000004); 1890
15254: CTF_THREAD_REF', 'LongWord.SetUInt( $00000002); 1889
15255: CubicMetersPerCord', 'Extended.SetExtended( CubicMetersPerCubicFoot * 128); 1953
15256: CubicMetersPerCordFoot', 'Extended.SetExtended( CubicMetersPerCubicFoot * 16); 1954
15257: CubicMetersPerUKBucket', 'Extended.SetExtended( CubicMetersPerUKPeck * 2); 1976
15258: CubicMetersPerUKBushel', 'Extended.SetExtended( CubicMetersPerUKBucket * 2); 1977
15259: CubicMetersPerUKGallon', 'Extended.setExtended( 0.00454609); 1969
15260: CubicMetersPerUKGill', 'Extended.SetExtended( CubicMetersPerUKPint / 4); 1973
15261: CubicMetersPerUKOunce', 'Extended.SetExtended( CubicMetersPerUKPint / 20); 1974
15262: CubicMetersPerUKPeck', 'Extended.SetExtended( CubicMetersPerUKGallon * 2); 1975
15263: CubicMetersPerUKPint', 'Extended.SetExtended( CubicMetersPerUKQuart / 2); 1972
15264: CubicMetersPerUKPottle', 'Extended.SetExtended( CubicMetersPerUKGallon / 2); 1970
15265: CubicMetersPerUKQuart', 'Extended.SetExtended( CubicMetersPerUKPottle / 2); 1971
15266: CubicMetersPerUSDryBucket', 'Extended.SetExtended( CubicMetersPerUSDryPeck * 2); 1967
15267: CubicMetersPerUSDryBushel', 'Extended.SetExtended( CubicMetersPerUSDryBucket * 2); 1968
15268: CubicMetersPerUSDryGallon', 'Extended.setExtended( CubicMetersPerCubicInch * 268.8025); 1963

```

PC08 G:\sonysavefeb2014\maxboxfiles\TypeConstClasslistSortmX4Reporfinal.txt
<http://www.softwareschule.ch/maxbox.htm>

```

15358: DEFAULT_PITCH', 'LongInt.SetInt( 0); 165
15359: DEFAULT_QUALITY', 'LongInt.SetInt( 0); 160
15360: DefaultCaseSensitivity', 'BooleanBoolToStr( False); 1620
15361: DefaultCaseSensitivity', 'BooleanBoolToStr( False); 1621
15362: DefaultCompareDelta', 'Extended.SetExtended( 1.0E-34); 2224
15363: DefaultCompareEpsilon', 'Extended.setExtended( 1.0E-10); 2228
15364: DefaultCursor', 'LongInt.SetInt( 0); 1924
15365: DefaultDateFmt', 'AnsiString.SetString( 'mm/dd/yyyy; 1986
15366: DefaultDateOrder', '.SetString( doDMY); 1587
15367: DefaultDateOrder', '.SetString( doDMY); 1600
15368: DefaultDateOrder', '.SetString( doDMY; 1617
15369: DefaultDateTimeFmt', 'AnsiString.SetString( 'mm/dd/yyyy hh:mm:ss; 1988
15370: DefaultFMTbcdScale', 'LongInt.SetInt( 6); 982
15371: DefaultHotTrackColor', 'LongWord.SetUInt( $00D2BDB6); 1604
15372: DefaultHotTrackFrameColor', 'LongWord.SetUInt( $006A240A); 1605
15373: DefaultInterval', 'LongInt.SetInt( 1000); 2750
15374: DefaultJvBrowseFolderDialogOptions', 'LongInt.Value.ts32 := ord(odStatusAvailable) or
ord(odNewDialogStyle); 2695
15375: DefaultMaxBlobSize', 'LongInt.SetInt( - 1); 1926
15376: DefaultRowsetSize', '.SetString( DBXDefaultRowsetSize); 1927
15377: DefaultSvcDesiredAccess', '.SetString( SERVICE_ALL_ACCESS); 2565
15378: DefaultTimeFmt', 'AnsiString.SetString( 'hh:mm:ss; 1987
15379: DefaultTrackFontOptions', 'LongInt.Value.ts32 := ord(hoFollowFont) or ord(hoPreserveColor) or
ord(hoPreserveStyle); 1603
15380: DefCalcPrecision', 'LongInt.SetInt( 15); 1585
15381: DefDelimiters', 'String.SetString( ; 2004
15382: DEFFILENAME', 'String.SetString( 'firstdemo.txt; 935
15383: DEFFILENAME', 'String.SetString( 'firstdemo.txt; 959
15384: DefHintColor', '.SetString( clInfoBk); 2648
15385: DefHintHidePause', 'LongInt.SetInt( DefHintPause * 5); 2651
15386: DefHintPause', 'LongInt.SetInt( 500); 2649
15387: DefHintShortPause', 'LongInt.SetInt( DefHintPause div 10); 2650
15388: DEFINIFILE', 'String.SetString( 'maxboxdef.ini; 936
15389: DEFINIFILE', 'String.SetString( 'maxboxdef.ini; 960
15390: DefQBEStartParam', 'String.SetString( '#; 2722
15391: DefQuote', 'String.SetString( ''; 2005
15392: DefRepeatValue', 'LongInt.SetInt( 1); 2006
15393: DefResetRepeat', 'BooleanBoolToStr( True); 2007
15394: DefTabSize', 'LongInt.SetInt( 8); 2008
15395: DefWrap', 'LongInt.SetInt( 80); 2009
15396: DegPerCycle', 'Extended.setExtended( 360.0); 2574
15397: DegPerGrad', 'Extended.setExtended( 0.9); 2575
15398: DegPerRad', 'Extended.setExtended( 57.295779513082320876798154814105); 2576
15399: DeltaJD', 'LongWord.SetUInt( $00232DA8); 2998
15400: DLLVER_BUILD_MASK', 'LongWord.SetUInt( Int64 ( $00000000FFFF0000 )); 1896
15401: DLLVER_MAJOR_MASK', 'LongWord.SetUInt( Int64 ( $FFFF000000000000 )); 1894
15402: DLLVER_MINOR_MASK', 'LongWord.SetUInt( Int64 ( $0000FFFF00000000 )); 1895
15403: DLLVER_PLATFORM_NT', 'LongWord.SetUInt( $00000002); 1893
15404: DLLVER_PLATFORM_WINDOWS', 'LongWord.SetUInt( $00000001); 1892
15405: DLLVER_QFE_MASK', 'LongWord.SetUInt( Int64 ( $000000000000FFFF )); 1897
15406: DosDelimSetW', 'WideString.SetString( '\'; 2986
15407: DoubleCompareDelta', 'Extended.setExtended( 1.0E-280); 2222
15408: DoubleCompareEpsilon', 'Extended.setExtended( 1.0E-13); 2226
15409: DOW_WEEKEND', 'TTFDaysOfWeek.SetString(ord(dowSunday) or ord(dowSaturday)); 2745
15410: DRAFT_QUALITY', 'LongInt.SetInt( 1); 161
15411: DriveDelim', 'String.SetString( ':; 1610
15412: dsEditModes', 'LongInt.Value.ts32 := ord(dsEdit) or ord(dsInsert) or ord(dsSetKey); 550
15413: dsMaxStringSize', 'LongInt.SetInt( 8192); 553
15414: DSTINVERT', 'LongWord.SetUInt( $00550009); 796
15415: dsWriteModes', 'LongInt.Value.ts32 := ord(dsEdit) or ord(dsInsert) or ord(dsSetKey) or ord(dsCalcFields) or
ord(dsFilter) or ord(dsNewValue) or ord(dsInternalCalc); 551
15416: DTDEXTENSION', 'String.SetString( '.dtd; 928
15417: dxColor_BgCkOXP', 'LongWord.SetUInt( TColor ( $00CC9999 )); 2866
15418: dxColor_BgOXP', 'LongWord.SetUInt( TColor ( $00D6BEB5 )); 2865
15419: dxColor_BorderLineOXP', 'LongWord.SetUInt( TColor ( $00663300 )); 2864
15420: dxColor_Btn_Dis_Border_WXP', 'LongWord.SetUInt( TColor ( $00BDC7CE )); 2848
15421: dxColor_Btn_Dis_Edges_WXP', 'LongWord.SetUInt( TColor ( $00BDC7CE )); 2850
15422: dxColor_Btn_Enb_BgFrom_WXP', 'LongWord.SetUInt( TColor ( $00FFFFFF )); 2851
15423: dxColor_Btn_Enb_BgTo_WXP', 'LongWord.SetUInt( TColor ( $00E7EBEF )); 2852
15424: dxColor_Btn_Enb_Border_WXP', 'LongWord.SetUInt( TColor ( $00733800 )); 2847
15425: dxColor_Btn_Enb_CkFrom_WXP', 'LongWord.SetUInt( TColor ( $00C6CFD6 )); 2853
15426: dxColor_Btn_Enb_CkTo_WXP', 'LongWord.SetUInt( TColor ( $00EBF3F7 )); 2854
15427: dxColor_Btn_Enb_Edges_WXP', 'LongWord.SetUInt( TColor ( $00AD9E7B )); 2849
15428: dxColor_Btn_Enb_FcFrom_WXP', 'LongWord.SetUInt( TColor ( $00FFE7CE )); 2855
15429: dxColor_Btn_Enb_FcTo_WXP', 'LongWord.SetUInt( TColor ( $00EF846D )); 2856
15430: dxColor_Btn_Enb_HlFrom_WXP', 'LongWord.SetUInt( TColor ( $00CEF3FF )); 2857
15431: dxColor_Btn_Enb_HlTo_WXP', 'LongWord.SetUInt( TColor ( $000096E7 )); 2858
15432: dxColor_Chk_Enb_Border_WXP', 'LongWord.SetUInt( TColor ( $00845118 )); 2859
15433: dxColor_Chk_Enb_GraSymb_WXP', 'LongWord.SetUInt( TColor ( $0071C671 )); 2861
15434: dxColor_Chk_Enb_NmSymb_WXP', 'LongWord.SetUInt( TColor ( $0021A621 )); 2860
15435: dxColor_DotNetFrame', 'LongWord.SetUInt( TColor ( $00F7FBFF )); 2863
15436: dxColor_Msc_Dis_Caption_WXP', 'LongWord.SetUInt( TColor ( $0094A6A5 )); 2862
15437: E', 'Float.SetString( 2.7182818284590452353602874713527); 1305
15438: E', 'Float.SetString( 2.7182818284590452353602874713527); 1373
15439: E_FAIL', 'LongWord.SetUInt( HRESULT ( $80004005 )); 2136
15440: EAMD_3DNOW_FLAG', 'LongWord.SetUInt( DWORD ( $80000000 )); 1519
15441: EAMD_BIT_10', 'LongWord.SetUInt( $00000400); 1498
15442: EAMD_BIT_12', 'LongWord.SetUInt( $00001000); 1500
15443: EAMD_BIT_14', 'LongWord.SetUInt( $00004000); 1502

```

```
15444: EAMD_BIT_17', 'LongWord.SetUInt( $00020000); 1505
15445: EAMD_BIT_18', 'LongWord.SetUInt( $00040000); 1506
15446: EAMD_BIT_19', 'LongWord.SetUInt( $00080000); 1507
15447: EAMD_BIT_20', 'LongWord.SetUInt( $00100000); 1508
15448: EAMD_BIT_21', 'LongWord.SetUInt( $00200000); 1509
15449: EAMD_BIT_22', 'LongWord.SetUInt( $00400000); 1510
15450: EAMD_BIT_24', 'LongWord.SetUInt( $01000000); 1512
15451: EAMD_BIT_25', 'LongWord.SetUInt( $02000000); 1513
15452: EAMD_BIT_26', 'LongWord.SetUInt( $04000000); 1514
15453: EAMD_BIT_27', 'LongWord.SetUInt( $08000000); 1515
15454: EAMD_BIT_28', 'LongWord.SetUInt( $10000000); 1516
15455: EAMD_BIT_29', 'LongWord.SetUInt( $20000000); 1517
15456: EAMD_BIT_30', 'LongWord.SetUInt( $40000000); 1518
15457: EAMD_BIT_6', 'LongWord.SetUInt( $00000040); 1494
15458: EAMD_BIT_9', 'LongWord.SetUInt( $00000200); 1497
15459: EAMD_CX8_FLAG', 'LongWord.SetUInt( $00000100); 1496
15460: EAMD_DE_FLAG', 'LongWord.SetUInt( $00000004); 1490
15461: EAMD_FCMOV_FLAG', 'LongWord.SetUInt( $00010000); 1504
15462: EAMD_FPU_FLAG', 'LongWord.SetUInt( $00000001); 1488
15463: EAMD_ICMOV_FLAG', 'LongWord.SetUInt( $00008000); 1503
15464: EAMD_MCE_FLAG', 'LongWord.SetUInt( $00000080); 1495
15465: EAMD_MMX_FLAG', 'LongWord.SetUInt( $00800000); 1511
15466: EAMD_MSR_FLAG', 'LongWord.SetUInt( $00000020); 1493
15467: EAMD_PGE_FLAG', 'LongWord.SetUInt( $00002000); 1501
15468: EAMD_PSE_FLAG', 'LongWord.SetUInt( $00000008); 1491
15469: EAMD_SEP_FLAG', 'LongWord.SetUInt( $00000800); 1499
15470: EAMD_TSC_FLAG', 'LongWord.SetUInt( $00000010); 1492
15471: EAMD_VME_FLAG', 'LongWord.SetUInt( $00000002); 1489
15472: ecAutoCompletion', 'LongInt.SetInt( 650); 3128
15473: ecBlockIndent', 'LongInt.SetInt( 610); 3124
15474: ecBlockUnindent', 'LongInt.SetInt( 611); 3125
15475: ecChar', 'LongInt.SetInt( 511); 3118
15476: ecClearAll', 'LongInt.SetInt( 508); 3115
15477: ecColumnSelect', 'LongInt.SetInt( 232); 3081
15478: ecCommentBlock', 'LongInt.SetInt( 251); 3084
15479: ecContextHelp', 'LongInt.SetInt( 490); 3107
15480: ecCopy', 'LongInt.SetInt( 201); 3072
15481: ecCut', 'LongInt.SetInt( 603); 3122
15482: ecDeleteBOL', 'LongInt.SetInt( 505); 3112
15483: ecDeleteChar', 'LongInt.SetInt( 502); 3109
15484: ecDeleteEOL', 'LongInt.SetInt( 506); 3113
15485: ecDeleteLastChar', 'LongInt.SetInt( 501); 3108
15486: ecDeleteLastWord', 'LongInt.SetInt( 504); 3111
15487: ecDeleteLine', 'LongInt.SetInt( 507); 3114
15488: ecDeleteWord', 'LongInt.SetInt( 503); 3110
15489: ecDown', 'LongInt.SetInt( 4); 3055
15490: ecEditCommandFirst', 'LongInt.SetInt( 501); 3050
15491: ecEditCommandLast', 'LongInt.SetInt( 1000); 3051
15492: ecEditorBottom', 'LongInt.SetInt( 16); 3067
15493: ecEditorTop', 'LongInt.SetInt( 15); 3066
15494: ecGotFocus', 'LongInt.SetInt( 480); 3105
15495: ecGotoMarker0', 'LongInt.SetInt( 301); 3085
15496: ecGotoMarker1', 'LongInt.SetInt( 302); 3086
15497: ecGotoMarker2', 'LongInt.SetInt( 303); 3087
15498: ecGotoMarker3', 'LongInt.SetInt( 304); 3088
15499: ecGotoMarker4', 'LongInt.SetInt( 305); 3089
15500: ecGotoMarker5', 'LongInt.SetInt( 306); 3090
15501: ecGotoMarker6', 'LongInt.SetInt( 307); 3091
15502: ecGotoMarker7', 'LongInt.SetInt( 308); 3092
15503: ecGotoMarker8', 'LongInt.SetInt( 309); 3093
15504: ecGotoMarker9', 'LongInt.SetInt( 310); 3094
15505: ecGotoXY', 'LongInt.SetInt( 17); 3068
15506: ecImeStr', 'LongInt.SetInt( 550); 3119
15507: ecInsertLine', 'LongInt.SetInt( 510); 3117
15508: ecInsertMode', 'LongInt.SetInt( 221); 3077
15509: ecLeft', 'LongInt.SetInt( 1); 3052
15510: ecLineBreak', 'LongInt.SetInt( 509); 3116
15511: ecLineEnd', 'LongInt.SetInt( 8); 3059
15512: ecLineSelect', 'LongInt.SetInt( 233); 3082
15513: ecLineStart', 'LongInt.SetInt( 7); 3058
15514: ecLostFocus', 'LongInt.SetInt( 481); 3106
15515: ecLowerCase', 'LongInt.SetInt( 621); 3130
15516: ecLowerCaseBlock', 'LongInt.SetInt( 626); 3134
15517: ecMatchBracket', 'LongInt.SetInt( 250); 3083
15518: ecNone', 'LongInt.SetInt( 0); 3047
15519: ecNormalSelect', 'LongInt.SetInt( 231); 3080
15520: ecOverwriteMode', 'LongInt.SetInt( 222); 3078
15521: ecPageBottom', 'LongInt.SetInt( 14); 3065
15522: ecPageDown', 'LongInt.SetInt( 10); 3061
15523: ecPageLeft', 'LongInt.SetInt( 11); 3062
15524: ecPageRight', 'LongInt.SetInt( 12); 3063
15525: ecPageTop', 'LongInt.SetInt( 13); 3064
15526: ecPageUp', 'LongInt.SetInt( 9); 3060
15527: ecPaste', 'LongInt.SetInt( 604); 3123
15528: ecRedo', 'LongInt.SetInt( 602); 3121
15529: ecRight', 'LongInt.SetInt( 2); 3053
15530: ecScrollDown', 'LongInt.SetInt( 212); 3074
15531: ecScrollLeft', 'LongInt.SetInt( 213); 3075
15532: ecScrollRight', 'LongInt.SetInt( 214); 3076
```



```

15533: ecScrollUp', 'LongInt.SetInt( 211); 3073
15534: ecSelectAll', 'LongInt.SetInt( 199); 3071
15535: ecSelection', 'LongInt.SetInt( 100); 3069
15536: ecSelWord', 'LongInt.SetInt( 198); 3070
15537: ecSetMarker0', 'LongInt.SetInt( 351); 3095
15538: ecSetMarker1', 'LongInt.SetInt( 352); 3096
15539: ecSetMarker2', 'LongInt.SetInt( 353); 3097
15540: ecSetMarker3', 'LongInt.SetInt( 354); 3098
15541: ecSetMarker4', 'LongInt.SetInt( 355); 3099
15542: ecSetMarker5', 'LongInt.SetInt( 356); 3100
15543: ecSetMarker6', 'LongInt.SetInt( 357); 3101
15544: ecSetMarker7', 'LongInt.SetInt( 358); 3102
15545: ecSetMarker8', 'LongInt.SetInt( 359); 3103
15546: ecSetMarker9', 'LongInt.SetInt( 360); 3104
15547: ecShiftTab', 'LongInt.SetInt( 613); 3127
15548: ecString', 'LongInt.SetInt( 630); 3136
15549: ecTab', 'LongInt.SetInt( 612); 3126
15550: ecTitleCase', 'LongInt.SetInt( 623); 3132
15551: ecToggleCase', 'LongInt.SetInt( 622); 3131
15552: ecToggleCaseBlock', 'LongInt.SetInt( 627); 3135
15553: ecToggleMode', 'LongInt.SetInt( 223); 3079
15554: ecUndo', 'LongInt.SetInt( 601); 3120
15555: ecUp', 'LongInt.SetInt( 3); 3054
15556: ecUpperCase', 'LongInt.SetInt( 620); 3129
15557: ecUpperCaseBlock', 'LongInt.SetInt( 625); 3133
15558: ecUserFirst', 'LongInt.SetInt( 1001); 3137
15559: ecViewCommandFirst', 'LongInt.SetInt( 0); 3048
15560: ecViewCommandLast', 'LongInt.SetInt( 500); 3049
15561: ecWordLeft', 'LongInt.SetInt( 5); 3056
15562: ecWordRight', 'LongInt.SetInt( 6); 3057
15563: ECYRIX_APIC_FLAG', 'LongWord.SetUInt( $00000200); 1561
15564: ECYRIX_BIT_11', 'LongWord.SetUInt( $00000800); 1563
15565: ECYRIX_BIT_17', 'LongWord.SetUInt( $00020000); 1569
15566: ECYRIX_BIT_18', 'LongWord.SetUInt( $00040000); 1570
15567: ECYRIX_BIT_19', 'LongWord.SetUInt( $00080000); 1571
15568: ECYRIX_BIT_20', 'LongWord.SetUInt( $00100000); 1572
15569: ECYRIX_BIT_21', 'LongWord.SetUInt( $00200000); 1573
15570: ECYRIX_BIT_22', 'LongWord.SetUInt( $00400000); 1574
15571: ECYRIX_BIT_25', 'LongWord.SetUInt( $02000000); 1577
15572: ECYRIX_BIT_26', 'LongWord.SetUInt( $04000000); 1578
15573: ECYRIX_BIT_27', 'LongWord.SetUInt( $08000000); 1579
15574: ECYRIX_BIT_28', 'LongWord.SetUInt( $10000000); 1580
15575: ECYRIX_BIT_29', 'LongWord.SetUInt( $20000000); 1581
15576: ECYRIX_BIT_30', 'LongWord.SetUInt( $40000000); 1582
15577: ECYRIX_BIT_31', 'LongWord.SetUInt( DWORD ( $80000000 )); 1583
15578: ECYRIX_CX8_FLAG', 'LongWord.SetUInt( $00000100); 1560
15579: ECYRIX_DE_FLAG', 'LongWord.SetUInt( $00000004); 1554
15580: ECYRIX_BMMX_FLAG', 'LongWord.SetUInt( $01000000); 1576
15581: ECYRIX_FCMOV_FLAG', 'LongWord.SetUInt( $00010000); 1568
15582: ECYRIX_FPU_FLAG', 'LongWord.SetUInt( $00000001); 1552
15583: ECYRIX_ICMOV_FLAG', 'LongWord.SetUInt( $00008000); 1567
15584: ECYRIX_MCA_FLAG', 'LongWord.SetUInt( $00004000); 1566
15585: ECYRIX_MCE_FLAG', 'LongWord.SetUInt( $00000080); 1559
15586: ECYRIX_MMX_FLAG', 'LongWord.SetUInt( $00800000); 1575
15587: ECYRIX_MSR_FLAG', 'LongWord.SetUInt( $00000020); 1557
15588: ECYRIX_MTRR_FLAG', 'LongWord.SetUInt( $00001000); 1564
15589: ECYRIX_PAE_FLAG', 'LongWord.SetUInt( $00000040); 1558
15590: ECYRIX_PGE_FLAG', 'LongWord.SetUInt( $00002000); 1565
15591: ECYRIX_PSE_FLAG', 'LongWord.SetUInt( $00000008); 1555
15592: ECYRIX_SEP_FLAG', 'LongWord.SetUInt( $00000400); 1562
15593: ECYRIX_TSC_FLAG', 'LongWord.SetUInt( $00000010); 1556
15594: ECYRIX_VME_FLAG', 'LongWord.SetUInt( $00000002); 1553
15595: EmptyDelphiSet', 'TDelphiSet.SetString(); 1333
15596: EmptyDelphiSet', 'TDelphiSet.SetString(); 1401
15597: ENDSESSION_LOGOFF', 'LongWord.SetUInt( DWORD ( $80000000 )); 119
15598: EOL', 'Char.SetString( #13+#10); 1075
15599: EpsilonSqr', 'Float.SetExtended(1E-20); 2450
15600: EqualsValue', 'LongInt.SetInt( 0); 2139
15601: ErrAccessViolation', 'LongInt.SetInt( 2); 1230
15602: ErrAllocationExceeded', 'LongInt.SetInt( 3); 1231
15603: ErrBadRange', 'LongInt.SetInt( 2); 2736
15604: ErrExpression', 'LongInt.SetInt( 3); 2737
15605: ErrFileAlreadyExists', 'LongInt.SetInt( 6); 1234
15606: ErrFileNotFound', 'LongInt.SetInt( 1); 1229
15607: ErrIllegalOperation', 'LongInt.SetInt( 4); 1232
15608: ErrInvalidNum', 'LongInt.SetInt( 7); 2741
15609: ErrNoSuchUser', 'LongInt.SetInt( 7); 1235
15610: ErrOpCloseParen', 'LongInt.SetInt( 6); 2740
15611: ErrOpenParen', 'LongInt.SetInt( 5); 2739
15612: ErrOperator', 'LongInt.SetInt( 4); 2738
15613: ErrOptionNegotiationFailed', 'LongInt.SetInt( 8); 1236
15614: ErrParserStack', 'LongInt.SetInt( 1); 2735
15615: ErrUndefined', 'LongInt.SetInt( 0); 1228
15616: ErrUnknownTransferID', 'LongInt.SetInt( 5); 1233
15617: EscChar', 'String.SetString( '\'; 2046
15618: eSQLAutoIncr', 'LongWord.SetUInt( $0004); 1911
15619: eSQLDefault', 'LongWord.SetUInt( $0008); 1912
15620: eSQLFunction', 'LongWord.SetUInt( $0002); 1906
15621: eSQLLocal', 'LongWord.SetUInt( $0020); 1904

```

```

15622: eSQLNonUnique', 'LongWord.SetUInt( $0001); 1913
15623: eSQLPackage', 'LongWord.SetUInt( $0004); 1907
15624: eSQLPrimaryKey', 'LongWord.SetUInt( $0004); 1915
15625: eSQLProcedure', 'LongWord.SetUInt( $0001); 1905
15626: eSQLRowId', 'LongWord.SetUInt( $0001); 1909
15627: eSQLRowVersion', 'LongWord.SetUInt( $0002); 1910
15628: eSQLSynonym', 'LongWord.SetUInt( $0008); 1902
15629: eSQLSysProcedure', 'LongWord.SetUInt( $0008); 1908
15630: eSQLSystemTable', 'LongWord.SetUInt( $0004); 1901
15631: eSQLTable', 'LongWord.SetUInt( $0001); 1899
15632: eSQLTempTable', 'LongWord.SetUInt( $0010); 1903
15633: eSQLUnique', 'LongWord.SetUInt( $0002); 1914
15634: eSQLView', 'LongWord.SetUInt( $0002); 1900
15635: NA_LoopId', 'String.SetString( 'N/A;
15636: ElementSpecId_Reserved', 'String.SetString( 'Reserved;
15637: EDIDataType_Numeric', 'String.SetString( 'N;
15638: EDIDataType_Decimal', 'String.SetString( 'R;
15639: EDIDataType_Identifier', 'String.SetString( 'ID;
15640: EDIDataType_String', 'String.SetString( 'AN;
15641: EDIDataType_Date', 'String.SetString( 'DT;
15642: EDIDataType_Time', 'String.SetString( 'TM;
15643: EDIDataType_Binary', 'String.SetString( 'B;
15644: T_OBJECT, 'TEDIObject;
15645: TEDIObjectArray', 'array of TEDIObject;
15646: Euler', 'Extended.SetExtended( 2.7182818284590452353602874713527); 1693
15647: Euler', 'Extended.SetExtended(2.7182818284590452353602874713527); // Natural constant 1717
15648: EulerMascheroni', 'Extended.SetExtended( 0.5772156649015328606065120900824); 1698
15649: EulerMascheroni', 'Float.SetString( 0.5772156649015328606065120900824); 1310
15650: EulerMascheroni', 'Float.SetString( 0.5772156649015328606065120900824); 1378
15651: EVENT_MODIFY_STATE', 'LongWord.SetUInt( $0002); 198
15652: EW_EXITANDEXECAPP', 'LongWord.SetUInt( $0044); 118
15653: EW_REBOOTSYSYSTEM', 'LongWord.SetUInt( $0043); 117
15654: EW_RESTARTWINDOWS', 'LongWord.SetUInt( $0042); 116
15655: EWX_FORCE', 'LongInt.SetInt( 4); 123
15656: EWX_FORCEIFHUNG', 'LongWord.SetUInt( $10); 125
15657: EWX_LOGOFF', 'LongInt.SetInt( 0); 120
15658: EWX_POWEROFF', 'LongInt.SetInt( 8); 124
15659: EWX_REBOOT', 'LongInt.SetInt( 2); 122
15660: EWX_SHUTDOWN', 'LongInt.SetInt( 1); 121
15661: EXCEPTLOGFILE', 'String.SetString( 'maxboxerrorlog.txt; 937
15662: EXCEPTLOGFILE', 'String.SetString( 'maxboxerrorlog.txt; 961
15663: EXENAME', 'String.SetString( 'maxbox3.exe; 946
15664: EXENAME', 'String.SetString( 'maxbox3.exe; 970
15665: ExpLimit', 'LongInt.SetInt( 11356); 2731
15666: ExpLog', 'TSTExceptionLog.SetString( 'NIL; 1985
15667: ExtendedCompareDelta', 'Extended.setExtended( 1.0E-4400); 2223
15668: ExtendedCompareEpsilon', 'Extended.setExtended( 1.0E-17); 2227
15669: ExtPascalVersion', 'String.SetString( '0.9.8; 3392
15670: faAnyFile', 'LongInt.SetInt($0000003F); 2015
15671: faAnyFile', 'LongWord.SetUInt( $0000003F); 31
15672: faArchive', 'LongInt.SetInt($00000020); 2014
15673: faDirectory', 'LongInt.SetInt($00000010); 2013
15674: faDirectory', 'LongWord.SetUInt( $00000010); 30
15675: faHidden', 'LongInt.SetInt($00000002); 2011
15676: FahrenheitAbsoluteZero', 'Extended.setExtended( - 459.67); 2572
15677: FahrenheitFreezingPoint', 'Extended.setExtended( 32.0); 2569
15678: FALT', 'LongWord.SetUInt( $10); 286
15679: FAPPCOMMAND_KEY', 'LongInt.SetInt( 0); 2841
15680: FAPPCOMMAND_MASK', 'LongWord.SetUInt( $F000); 2843
15681: FAPPCOMMAND_MOUSE', 'LongWord.SetUInt( $8000); 2840
15682: FAPPCOMMAND_OEM', 'LongWord.SetUInt( $1000); 2842
15683: faReadOnly', 'LongInt.SetInt($00000001); 2010
15684: faSymLink', 'LongInt.SetInt($00000040); // platform; 2017
15685: faSysFile', 'LongInt.SetInt($00000004); 2012
15686: faVolumeID', 'LongInt.SetInt($00000008); //platform deprecated; 2016
15687: FCONTROL', 'LongInt.SetInt( 8); 285
15688: FDomain', 'LongInt.SetInt( - 1); 3240
15689: FILE_ATTRIBUTE_ARCHIVE', 'LongWord.SetUInt( $00000020); 291
15690: FILE_ATTRIBUTE_COMPRESSED', 'LongWord.SetUInt( $00000800); 297
15691: FILE_ATTRIBUTE_DEVICE', 'LongWord.SetUInt( $00000040); 292
15692: FILE_ATTRIBUTE_DIRECTORY', 'LongWord.SetUInt( $00000010); 290
15693: FILE_ATTRIBUTE_ENCRYPTED', 'LongWord.SetUInt( $00004000); 300
15694: FILE_ATTRIBUTE_ENCRYPTED', 'LongWord.SetUInt( $4000); 2764
15695: FILE_ATTRIBUTE_HIDDEN', 'LongWord.SetUInt( $00000002); 288
15696: FILE_ATTRIBUTE_NORMAL', 'LongWord.SetUInt( $00000080); 293
15697: FILE_ATTRIBUTE_NOT_CONTENT_INDEXED', 'LongWord.SetUInt( $00002000); 299
15698: FILE_ATTRIBUTE_NOT_CONTENT_INDEXED', 'LongWord.SetUInt( $2000); 2763
15699: FILE_ATTRIBUTE_OFFLINE', 'LongWord.SetUInt( $00001000); 298
15700: FILE_ATTRIBUTE_READONLY', 'LongWord.SetUInt( $00000001); 287
15701: FILE_ATTRIBUTE_REPARSE_POINT', 'LongWord.SetUInt( $00000400); 296
15702: FILE_ATTRIBUTE_REPARSE_POINT', 'LongWord.SetUInt( $400); 2762
15703: FILE_ATTRIBUTE_SPARSE_FILE', 'LongWord.SetUInt( $00000200); 295
15704: FILE_ATTRIBUTE_SPARSE_FILE', 'LongWord.SetUInt( $200); 2761
15705: FILE_ATTRIBUTE_SYSTEM', 'LongWord.SetUInt( $00000004); 289
15706: FILE_ATTRIBUTE_TEMPORARY', 'LongWord.SetUInt( $00000100); 294
15707: FIXED_PITCH', 'LongInt.SetInt( 1); 166
15708: FixedHalf', 'LongWord.SetUInt( $7FFF); 2393
15709: FixedOne', 'LongWord.SetUInt( $10000); 2392
15710: fmClosed', 'LongInt.SetInt($D7B0); 2019

```

```

15711: fmClosed', 'LongWord.SetUInt( $D7B0); 2023
15712: fmClosed', 'LongWord.SetUInt( $D7B0); 2120
15713: fmCreate', 'LongWord.SetUInt( $FFFF); 823
15714: fmInOut', 'LongInt.SetInt($D7B3); 2022
15715: fmInOut', 'LongWord.SetUInt( $D7B3); 2123
15716: fmInOut', 'LongWord.SetUInt( $D7B3); 2026
15717: fmInput', 'LongInt.SetInt($D7B1); 2020
15718: fmInput', 'LongWord.SetUInt( $D7B1); 2024
15719: fmInput', 'LongWord.SetUInt( $D7B1); 2121
15720: fmOpenRead', '.SetString( O_RDONLY); 18
15721: fmOpenRead', 'LongWord.SetUInt( $0000); 24
15722: fmOpenReadWrite', '.SetString( O_RDWR); 20
15723: fmOpenReadWrite', 'LongWord.SetUInt( $0002); 26
15724: fmOpenWrite', '.SetString( O_WRONLY); 19
15725: fmOpenWrite', 'LongWord.SetUInt( $0001); 25
15726: fmOutput', 'LongInt.SetInt($D7B2); 2021
15727: fmOutput', 'LongWord.SetUInt( $D7B2); 2025
15728: fmOutput', 'LongWord.SetUInt( $D7B2); 2122
15729: fmShareDenyNone', 'LongWord.SetUInt( $0040); 29
15730: fmShareDenyRead', 'LongWord.SetUInt( $0030); //3.2 23
15731: fmShareDenyWrite', 'LongWord.SetUInt( $0020); 22
15732: fmShareDenyWrite', 'LongWord.SetUInt( $0020); 28
15733: fmShareExclusive', 'LongWord.SetUInt( $0010); 21
15734: fmShareExclusive', 'LongWord.SetUInt( $0010); 27
15735: FNOINVERT', 'LongInt.SetInt( 2); 283
15736: FO_COPY', 'LongWord.SetUInt( $0002); 2889
15737: FO_DELETE', 'LongWord.SetUInt( $0003); 2890
15738: FO_MOVE', 'LongWord.SetUInt( $0001); 2888
15739: FO_RENAME', 'LongWord.SetUInt( $0004); 2891
15740: FOF_ALLOWUNDO', 'LongWord.SetUInt( $0040); 2898
15741: FOF_CONFIRM_MOUSE', 'LongWord.SetUInt( $0002); 2893
15742: FOF_FILESONLY', 'LongWord.SetUInt( $0080); 2899
15743: FOF_MULTIDESTFILES', 'LongWord.SetUInt( $0001); 2892
15744: FOF_NO_CONNECTED_ELEMENTS', 'LongWord.SetUInt( $2000); 2767
15745: FOF_NOCONFIRMATION', 'LongWord.SetUInt( $0010); 2896
15746: FOF_NOCONFIRMKMDIR', 'LongWord.SetUInt( $0200); 2901
15747: FOF_NOCOPYSECURITYATTRIBS', 'LongWord.SetUInt( $800); 2765
15748: FOF_NOERRORUI', 'LongWord.SetUInt( $0400); 2902
15749: FOF_NORECURSEREPARSE', 'LongWord.SetUInt( $8000); 2768
15750: FOF_NORECURSION', 'LongWord.SetUInt( $1000); 2766
15751: FOF_RENAMEONCOLLISION', 'LongWord.SetUInt( $0008); 2895
15752: FOF_SILENT', 'LongWord.SetUInt( $0004); 2894
15753: FOF_SIMPLEPROGRESS', 'LongWord.SetUInt( $0100); 2900
15754: FOF_WANTMAPPINGHANDLE', 'LongWord.SetUInt( $0020); 2897
15755: FOF_WANTNUKEWARNING', 'LongWord.SetUInt( $4000); 2769
15756: FOk', 'LongInt.SetInt( 0); 3239
15757: FORMAT_ALIGN_MASK', 'LongWord.SetUInt( $300); 3339
15758: FORMAT_BOLD', 'LongWord.SetUInt( $400); 3344
15759: FORMAT_CENTER', 'LongWord.SetUInt( $200); 3342
15760: FORMAT_DEFAULT', 'LongWord.SetUInt( $0); 3337
15761: FORMAT_DOUBLELINE', 'LongWord.SetUInt( $4000); 3349
15762: FORMAT_ITALIC', 'LongWord.SetUInt( $1000); 3346
15763: FORMAT_JUSTIFIED', 'LongWord.SetUInt( $300); 3343
15764: FORMAT_LEFT', 'LongWord.SetUInt( $0); 3340
15765: FORMAT_LINES', 'LongWord.SetUInt( $C000); 3350
15766: FORMAT_RIGHT', 'LongWord.SetUInt( $100); 3341
15767: FORMAT_SINGLELINE', 'LongWord.SetUInt( $8000); 3348
15768: FORMAT_SIZE_MASK', 'LongWord.SetUInt( $FF); 3338
15769: FORMAT_UNDEFINED', 'LongWord.SetUInt( $2000); 3347
15770: FORMAT_UNDERLINE', 'LongWord.SetUInt( $800); 3345
15771: FORMAT_XPOS_MASK', 'LongWord.SetUInt( $FFFF0000); 3351
15772: FOverflow', 'LongInt.SetInt( - 3); 3242
15773: FPLOSS', 'LongInt.SetInt( - 6); 3245
15774: FPU_FLAG', 'LongWord.SetUInt( $00000001); 1424
15775: FSHIFT', 'LongInt.SetInt( 4); 284
15776: FSing', 'LongInt.SetInt( - 2); 3241
15777: ftADT', 'LongInt.Value.ts32:= 1; 563
15778: ftArray', 'LongInt.Value.ts32:= 2; 564
15779: ftDataSet', 'LongInt.Value.ts32:= 8; 566
15780: FTLOSS', 'LongInt.SetInt( - 5); 3244
15781: ftReference', 'LongInt.Value.ts32:= 4; 565
15782: FUnderflow', 'LongInt.SetInt( - 4); 3243
15783: FVIRTKEY', 'LongInt.SetInt( 1); 282
15784: FXSR_FLAG', 'LongWord.SetUInt( $01000000); 1448
15785: GB2312_CHARSET', 'LongInt.SetInt( 134); 174
15786: GContentType', 'String.SetString( 'text/html; 1062
15787: GCT_INVALID', 'LongWord.SetUInt( $0000); 1769
15788: GCT_LFNCHAR', 'LongWord.SetUInt( $0001); 1770
15789: GCT_SEPARATOR', 'LongWord.SetUInt( $0008); 1773
15790: GCT_SHORTCHAR', 'LongWord.SetUInt( $0002); 1771
15791: GCT_WILD', 'LongWord.SetUInt( $0004); 1772
15792: gdi32', 'String.SetString( 'gdi32.dll; 13
15793: GFContentLength', 'LongInt.SetInt( - 1); 1061
15794: GFRequestedBlockSize', 'LongInt.SetInt( 1500); 1218
15795: Gold', 'Extended.setExtended( 1.61803398874989484821); 3213
15796: GoldenMean', 'Extended.setExtended( 1.618033988749894848204586834365638); 1697
15797: GoldenMean', 'Float.SetString( 1.618033988749894848204586834365638); 1309
15798: GoldenMean', 'Float.SetString( 1.618033988749894848204586834365638); 1377
15799: GOSType', '.SetString( otDotNet); 1110

```

```

15800: GOSType', '.SetString( otLinux); 1098
15801: GOSType', '.SetString( otWindows); 1108
15802: GOSType', '.SetString( otLinux; 1081
15803: GOSType', '.SetString( otWindows; 1091
15804: GPathDelim', 'String.SetString( '/; 1080
15805: GPathDelim', 'String.SetString( '/; 1099
15806: GPathDelim', 'String.SetString( '\; 1090
15807: GPathDelim', 'String.SetString( '\; 1109
15808: GPathDelim', 'String.SetString( '\; 1111
15809: GPL', 'BooleanBoolToStr( True); 2063
15810: GPL', 'BooleanBoolToStr( True); 302
15811: GradPerCycle', 'Extended.setExtended( 400.0); 2577
15812: GradPerDeg', 'Extended.setExtended( 1.11111111111111111111111111111111); 2578
15813: GradPerRad', 'Extended.setExtended( 63.661977236758134307553505349006); 2579
15814: GramsPerDrams', 'Extended.SetExtended( GramsPerPound / 256); 1979
15815: GramsPerGrains', 'Extended.SetExtended( GramsPerPound / 7000); 1980
15816: GramsPerLongTons', 'Extended.SetExtended( GramsPerPound * 2240); 1982
15817: GramsPerOunces', 'Extended.SetExtended( GramsPerPound / 16); 1983
15818: GramsPerPound', 'Extended.setExtended( 453.59237); 1978
15819: GramsPerStones', 'Extended.SetExtended( GramsPerPound * 14); 1984
15820: GramsPerTons', 'Extended.SetExtended( GramsPerPound * 2000); 1981
15821: Graphics32Version', 'String.SetString( '1.9.1; 2238
15822: GRAY_MARGIN', 'LongInt.SetInt( 10); 3334
15823: GReceiveTimeout', 'LongInt.SetInt( 4000); 1219
15824: GRecvBufferSizeDefault', 'LongInt.SetInt( 32 * 1024); 1145
15825: GResponseNo', 'LongInt.SetInt( 200); 1060
15826: GSendBufferSizeDefault', 'LongInt.SetInt( 32 * 1024); 1146
15827: GSessionIDCookie', 'String.SetString( 'IDHTPSESSIONID; 1063
15828: GTransferMode', 'TidTFTPMode.SetInt(1); 1217
15829: GTransferMode', 'TidTFTPMode.SetSet('tfOctet; 1216
15830: GUID_NULL', 'TGUID.SetString( '{00000000-0000-0000-000000000000}; 2138
15831: HANGEUL_CHARSET', 'LongInt.SetInt( 129); 173
15832: hcAccept', 'LongWord.SetUInt( $1844); 1029
15833: hcAllow', 'LongWord.SetUInt( $3D80); 1027
15834: hcAuthorization', 'LongWord.SetUInt( $ABCA); 1035
15835: hcCacheControl', 'LongWord.SetUInt( $4FF6); 1022
15836: hcConnection', 'LongWord.SetUInt( $0EDE); 1036
15837: hcContentDisposition', 'LongWord.SetUInt( $CBEB); 1038
15838: hcContentEncoding', 'LongWord.SetUInt( $C586); 1030
15839: hcContentLength', 'LongWord.SetUInt( $B0C4); 1033
15840: hcContentType', 'LongWord.SetUInt( $F0E0); 1032
15841: hcContentVersion', 'LongWord.SetUInt( $EDF4); 1031
15842: hcCookie', 'LongWord.SetUInt( $27B3); 1037
15843: hcDate', 'LongWord.SetUInt( $0EE6); 1023
15844: hcDELETE', 'LongWord.SetUInt( $92B2); 1020
15845: hcFrom', 'LongWord.SetUInt( $418F); 1024
15846: hcGET', 'LongWord.SetUInt( $14F5); 1018
15847: hcHost', 'LongWord.SetUInt( $3611); 1025
15848: hcIfModified', 'LongWord.SetUInt( $DDF0); 1026
15849: hcPOST', 'LongWord.SetUInt( $361D); 1021
15850: hcPUT', 'LongWord.SetUInt( $4AF5); 1019
15851: hcReferer', 'LongWord.SetUInt( $CEA5); 1034
15852: hcUserAgent', 'LongWord.SetUInt( $E890); 1028
15853: hdrsize', 'LongInt.SetInt( 4); 1226
15854: HEXDIGITS', 'THexArray.SetSet('0123456789ABCDEF; 474
15855: HF32_DEFAULT', 'LongInt.SetInt( 1); 3193
15856: HF32_SHARED', 'LongInt.SetInt( 2); 3194
15857: HIDE_WINDOW', 'LongInt.SetInt( 0); 400
15858: HighValidNaNTag', 'LongWord.SetUInt( $3FFFE); 1332
15859: HighValidNaNTag', 'LongWord.SetUInt( $3FFFE); 1400
15860: HKCC', 'HKEY.SetString('HKEY_CURRENT_CONFIG; 1418
15861: HKCU', 'HKEY.SetString('HKEY_CURRENT_USER; 1415
15862: HKEY_CLASSES_ROOT', 'LongInt.SetInt($80000000); 467
15863: HKEY_CURRENT_CONFIG', 'LongInt.SetInt($80000005); 472
15864: HKEY_CURRENT_USER', 'LongInt.SetInt($80000001); 468
15865: HKEY_DYN_DATA', 'LongInt.SetInt($80000006); 473
15866: HKEY_LOCAL_MACHINE', 'LongInt.SetInt($80000002); 469
15867: HKEY_PERFORMANCE_DATA', 'LongInt.SetInt($80000004); 471
15868: HKEY_USERS', 'LongInt.SetInt($80000003); 470
15869: HKLM', 'HKEY.SetString('HKEY_LOCAL_MACHINE; 1416
15870: HKPD', 'HKEY.SetString('HKEY_PERFORMANCE_DATA; 1419
15871: HKUS', 'HKEY.SetString('HKEY_USERS; 1417
15872: hLn2Pi', 'Extended.SetExtended( 0.91893853320467274178032973640562); 1694
15873: hLn2Pi', 'Float.SetString( 0.91893853320467274178032973640562); 1306
15874: hLn2Pi', 'Float.SetString( 0.91893853320467274178032973640562); 1374
15875: HourGlassCursor', 'LongInt.SetInt( - 11); 1925
15876: HourOnly', 'String.SetString( 'h; 3021
15877: HourOnlyU', 'String.SetString( 'H; 3024
15878: HoursInDay', 'LongInt.SetInt( 24); 3005
15879: HoursPerDay', 'Extended.SetExtended( 24); 2180
15880: HoursPerDay', 'LongInt.SetInt( 24); 32
15881: HoursPerDay', 'LongInt.SetInt( 24); 3329
15882: HSHELL_ACCESSIBILITYSTATE', 'LongInt.SetInt( 11); 2780
15883: HSHELL_ACTIVATESHELLWINDOW', 'LongInt.SetInt( 3); 2772
15884: HSHELL_APPCOMMAND', 'LongInt.SetInt( 12); 2781
15885: HSHELL_ENDTASK', 'LongInt.SetInt( 10); 2779
15886: HSHELL_GETMINRECT', 'LongInt.SetInt( 5); 2774
15887: HSHELL_HIGHBIT', 'LongWord.SetUInt( $8000); 2784
15888: HSHELL_LANGUAGE', 'LongInt.SetInt( 8); 2777

```



```

15889: HSHELL_REDRAW', 'LongInt.SetInt( 6); 2775
15890: HSHELL_SYSMENU', 'LongInt.SetInt( 9); 2778
15891: HSHELL_TASKMAN', 'LongInt.SetInt( 7); 2776
15892: HSHELL_WINDOWACTIVATED', 'LongInt.SetInt( 4); 2773
15893: HSHELL_WINDOWCREATED', 'LongInt.SetInt( 1); 2770
15894: HSHELL_WINDOWDESTROYED', 'LongInt.SetInt( 2); 2771
15895: HSHELL_WINDOWREPLACED', 'LongInt.SetInt( 13); 2782
15896: HSHELL_WINDOWREPLACING', 'LongInt.SetInt( 14); 2783
15897: icMAX_CATEGORY_DESC_LEN', 'LongInt.SetInt( 128); 3397
15898: ICMP_MIN', 'LongInt.SetInt( 8); 1122
15899: ID_ACCEPT_WAIT', 'LongInt.SetInt( 1000); 1142
15900: ID_DEFAULT_IP_VERSION', 'SetString( Id_IPv4); 1113
15901: ID_DEFAULT_IP_VERSION', 'SetString( Id_IPv6); 1112
15902: ID_MSG_NODECODE', 'BooleanBoolToStr( False); 1126
15903: ID_MSG_PRIORITY', 'SetString( mpNormal); 1128
15904: ID_MSG_USENOWFORDATE', 'BooleanBoolToStr( False); 1127
15905: ID_NC_MASK_LENGTH', 'LongInt.SetInt( 32); 1137
15906: ID_NETWORKCLASS', 'SetString( 'ID_NET_CLASS_A; 1138
15907: Id_Tid_HTTPAutoStartSession', 'Boolean.SetInt( 0); 1059
15908: Id_Tid_HTTPServer_KeepAlive', 'BooleanBoolToStr( false); 1055
15909: Id_Tid_HTTPServer_ParseParams', 'BooleanBoolToStr( false); 1056
15910: Id_Tid_HTTPServer_SessionState', 'BooleanBoolToStr( false); 1057
15911: Id_Tid_HTTPSessionTimeout', 'LongInt.SetInt( 0); 1058
15912: Id_TidFTP_Passive', 'Boolean; 2
15913: Id_TidFTP_TransferType', 'TidFTPTransferType; 1
15914: Id_TidHTTP_HandleRedirects', 'Boolean.BoolToStr(False); 8
15915: Id_TidHTTP_ProtocolVersion', 'string.setString( pvl_1); 5
15916: Id_TidHTTP_ProtocolVersion', 'TidHTTPProtocolVersion.SetString( pvl_1); 6
15917: Id_TidHTTP_RedirectMax', 'LongInt.SetInt( 15); 7
15918: Id_TIDICMP_ReceiveTimeout', 'LongInt.SetInt( 5000); 1125
15919: ID_UDP_BUFFER_SIZE', 'LongInt.SetInt(8192); 1237
15920: IDANI_CAPTION', 'LongInt.SetInt( 3); 253
15921: IDANI_CLOSE', 'LongInt.SetInt( 2); 252
15922: IDANI_OPEN', 'LongInt.SetInt( 1); 251
15923: IdCommandHandlersEnabledDefault', 'BooleanBoolToStr( True); 1152
15924: IdDef_UseEhlo', 'BoolToStr(TRUE)); // .BoolToStr( TRUE); 1143
15925: IdDefIdentQueryTimeout', 'LongInt.SetInt( 60000); 2407
15926: IdDefTimeout', 'LongInt.SetInt( 0); 1149
15927: iDEFAULTPACKETSIZE', 'LongInt.SetInt( 128); 1123
15928: iDEFAULTREPLYBUFSIZE', 'LongInt.SetInt( 1024); 1124
15929: IdEnabledDefault', 'BooleanBoolToStr( True); 1150
15930: IdFetchCaseSensitiveDefault', 'BooleanBoolToStr( true); 1070
15931: IdFetchCaseSensitiveDefault', 'BooleanBoolToStr( True); 1097
15932: IdFetchDeleteDefault', 'Boolean.BoolToStr( true); 1069
15933: IdFetchDeleteDefault', 'BooleanBoolToStr( True); 1096
15934: IdFetchDelimDefault', 'String.SetString( ' ; 1066
15935: IdFetchDelimDefault', 'String.SetString( ' ; 1095
15936: IdIdentQryTimeout', 'LongInt.SetInt( 60000); 2406
15937: IdInBufCacheSizeDefault', 'LongInt.SetInt( 32 * 1024); 1148
15938: IdListenQueueDefault', 'LongInt.SetInt( 15); 1153
15939: IdMaxLineLengthDefault', 'LongInt.SetInt( 16 * 1024); 1147
15940: IdParseParamsDefault', 'Boolean.BoolToStr( True); 1151
15941: IDRemoteFixPort', 'Boolean.SetInt(1); 2441
15942: IDRemoteUseStdErr', 'Boolean.SetInt(1); 2440
15943: IdTimeoutDefault', 'LongInt.SetInt( - 1); 1064
15944: IdTimeoutDefault', 'LongInt.SetInt( - 1); 1093
15945: IdTimeoutInfinite', 'LongInt.SetInt( - 2); 1065
15946: IdTimeoutInfinite', 'LongInt.SetInt( - 2); 1094
15947: ieBadRemark', 'LongInt.SetInt( 1); 3400
15948: IID_IFolderFilter', 'TGUID.SetString( '{9CC22886-DC8E-11d2-B1D0-00C04F8EEB3E}; 2693
15949: IID_IFolderFilterSite', 'TGUID.SetString( '{C0A651F5-B48B-11d2-B5ED-006097C686F6}; 2652
15950: INADDR_ANY', 'LongInt.SetInt( CARDINAL ( 0 )); 3406
15951: INADDR_NONE', 'LongWord.SetUInt( CARDINAL ( $FFFFFFFF )); 3407
15952: INCLUDEBOX', 'String.SetString( 'pas_includebox.inc; 966
15953: INCLUDEBOX', 'String.SetString( 'pas_includebox.inc; 942
15954: IndicatorWidth', 'LongInt.SetInt( 11); 893
15955: IndyMIMEBoundary', 'String.SetString( '=MoreStuf_2zzz1234sadvnqw3nerasdf; 2411
15956: IndyMultiPartAlternativeBoundary', 'String.SetString( '=MoreStuf_2altzzz1234sadvnqw3nerasdf; 2412
15957: IndyMultiPartRelatedBoundary', 'String.SetString( '=MoreStuf_2relzzzsadvnq1234w3nerasdf; 2413
15958: INFINITE', 'LongWord.SetUInt( Cardinal ( $FFFFFFFF )); 1092
15959: INFINITE', 'LongWord.SetUInt( LongWord ( $FFFFFFFF )); 1082
15960: INFINITE', 'LongWord.SetUInt( LongWord ( $FFFFFFFF )); 1100
15961: Infinity', 'Extended.setExtended( 1.0 / 0.0); 1665
15962: Infinity', 'LongInt.SetInt( 1 / 0); 1329
15963: Infinity', 'LongInt.SetInt( 1 / 0); 1397
15964: InitRepeatPause', 'LongInt.SetInt( 400); 889
15965: INVALID_IP_ADDRESS', 'LongWord.SetUInt( $ffffff);
15966: inv2Pi', 'Extended.setExtended( 0.15915494309189533576888376337251436203445964574046); 1695
15967: inv2Pi', 'Float.SetString( 0.15915494309189533576888376337251436203445964574046); 1307
15968: inv2Pi', 'Float.SetString( 0.15915494309189533576888376337251436203445964574046); 1375
15969: InvLn10', 'Extended.setExtended( 0.43429448190325182765); 3203
15970: InvLn2', 'Extended.setExtended( 1.44269504088896340736); 3202
15971: InvSqrt2Pi', 'Extended.setExtended( 0.39894228040143267794); 3208
15972: IP_WATCH_HIST_FILENAME', 'String.SetString( 'iphist.dat; 2409
15973: IP_WATCH_HIST_MAX', 'LongInt.SetInt( 25); 2408
15974: IP_WATCH_INTERVAL', 'LongInt.SetInt( 1000); 2410
15975: IPC_ADDBOOKMARK', 'LongInt.SetInt( 129); 3281
15976: IPC_CHANGECURRENTFILE', 'LongInt.SetInt( 245); 3286
15977: IPC_CHDIR', 'LongInt.SetInt( 103); 3296

```

```

15978: IPC_DELETE', 'LongInt.SetInt( 101); 3262
15979: IPC_EXECPUG', 'LongInt.SetInt( 202); 3274
15980: IPC_GET_REPEAT', 'LongInt.SetInt( 251); 3292
15981: IPC_GET_SHUFFLE', 'LongInt.SetInt( 250); 3291
15982: IPC_GETEQDATA', 'LongInt.SetInt( 127); 3279
15983: IPC_GETINFO', 'LongInt.SetInt( 126); 3278
15984: IPC_GETLISTLENGTH', 'LongInt.SetInt( 124); 3271
15985: IPC_GETLISTPOS', 'LongInt.SetInt( 125); 3277
15986: IPC_GETMBURL', 'LongInt.SetInt( 246); 3287
15987: IPC_GETOUTPUTTIME', 'LongInt.SetInt( 105); 3265
15988: IPC_GETPLAYLISTFILE', 'LongInt.SetInt( 211); 3275
15989: IPC_GETPLAYLISTTITLE', 'LongInt.SetInt( 212); 3276
15990: IPC_GETSKIN', 'LongInt.SetInt( 201); 3273
15991: IPC_GETVERSION', 'LongInt.SetInt( 0); 3261
15992: IPC_INETAVAILABLE', 'LongInt.SetInt( 242); 3284
15993: IPC_ISPLAYING', 'LongInt.SetInt( 104); 3264
15994: IPC_JUMPTOTIME', 'LongInt.SetInt( 106); 3266
15995: IPC_MBBLOCK', 'LongInt.SetInt( 248); 3289
15996: IPC_MBOpen', 'LongInt.SetInt( 241); 3283
15997: IPC_MBOpenREAL', 'LongInt.SetInt( 249); 3290
15998: IPC_PLAYFILE', 'LongInt.SetInt( 100); 3295
15999: IPC_REFRESHPLCACHE', 'LongInt.SetInt( 247); 3288
16000: IPC_RESTARTWINAMP', 'LongInt.SetInt( 135); 3282
16001: IPC_SET_REPEAT', 'LongInt.SetInt( 253); 3294
16002: IPC_SET_SHUFFLE', 'LongInt.SetInt( 252); 3293
16003: IPC_SETEQDATA', 'LongInt.SetInt( 128); 3280
16004: IPC_SETPANNING', 'LongInt.SetInt( 123); 3270
16005: IPC_SETPLAYLISTPOS', 'LongInt.SetInt( 121); 3268
16006: IPC_SETSKIN', 'LongInt.SetInt( 200); 3272
16007: IPC_SETVOLUME', 'LongInt.SetInt( 122); 3269
16008: IPC_STARTPLAY', 'LongInt.SetInt( 102); 3263
16009: IPC_UPDTITLE', 'LongInt.SetInt( 243); 3285
16010: IPC_WRITEPLAYLIST', 'LongInt.SetInt( 120); 3267
16011: iphlpapiLib', 'String.SetString( 'iphlpapi.dll; 3431
16012: JclVersionBuild', 'LongInt.SetInt( 965); 1252
16013: JclVersionMajor', 'LongInt.SetInt( 1); 1249
16014: JclVersionMinor', 'LongInt.SetInt( 22); 1250
16015: JclVersionRelease', 'LongInt.SetInt( 1); 1251
16016: JvChartAverageLineColorIndex', 'LongInt.SetInt( - 6); 2703
16017: JvChartAxisColorIndex', 'LongInt.SetInt( - 3); 2706
16018: JvChartDefaultMarkerSize', 'LongInt.SetInt( 3); 2709
16019: JvChartDivisionLineColorIndex', 'LongInt.SetInt( - 5); 2704
16020: JvChartHintColorIndex', 'LongInt.SetInt( - 2); 2707
16021: JvChartPaperColorIndex', 'LongInt.SetInt( - 1); 2708
16022: JvChartShadowColorIndex', 'LongInt.SetInt( - 4); 2705
16023: JvChartVersion', 'LongInt.SetInt( 300); 2696
16024: JvDefaultAvgLineColor', 'LongWord.SetUInt( TColor ( $00EEDDDD )); 2698
16025: JvDefaultDivisionLineColor', 'LongWord.SetUInt( TColor($C0C0C0)); 2699
16026: JvDefaultHintColor', 'LongWord.SetUInt( TColor ( $00DDFBFA )); 2697
16027: JvDefaultPassiveColor', 'LongWord.SetUInt( TColor ( $00202020 )); 2728
16028: JvDefaultShadowColor', 'LongWord.SetUInt(TColor($C0C0C0)); 2700
16029: JvDefaultYLegends', 'LongInt.SetInt( 20); 2701
16030: JvDefPageControlBorder', 'LongInt.SetInt( 4); 1586
16031: JvGridSort_ASC', 'BooleanBoolToStr( True); 2723
16032: JvGridSort_DESC', 'BooleanBoolToStr( True); 2725
16033: JvGridSort_DOWN', 'BooleanBoolToStr( True); 2726
16034: JvGridSort_UP', 'BooleanBoolToStr( True); 2724
16035: KelvinAbsoluteZero', 'Extended.setExtended( 0.0); 2573
16036: KelvinFreezingPoint', 'Extended.setExtended( 273.15); 2570
16037: kernel32', 'String.SetString( 'kernel32.dll; 11
16038: KF_ALTDOWN', 'LongWord.SetUInt( $2000); 421
16039: KF_DLGMODE', 'LongWord.SetUInt( $800); 419
16040: KF_EXTENDED', 'LongWord.SetUInt( $100); 418
16041: KF_MENU MODE', 'LongWord.SetUInt( $1000); 420
16042: KF_REPEAT', 'LongWord.SetUInt( $4000); 422
16043: KF_UP', 'LongWord.SetUInt( $8000); 423
16044: LastDelphiVersion', '.SetString( dvDelphi2005); 3160
16045: leDisabled', 'LongInt.SetInt( 2); 1998
16046: leEnabled', 'LongInt.SetInt( 1); 1997
16047: leString', 'LongWord.SetUInt( DWORD ( $80000000 )); 1999
16048: LF', 'Char.SetString( #10); 1042
16049: LF', 'Char.SetString( #10); 1073
16050: LF', 'Char.SetString( #10); 3176
16051: LF', 'Char.SetString( #10); 3365
16052: LF32_FIXED', 'LongWord.SetUInt( $00000001); 3195
16053: LF32_FREE', 'LongWord.SetUInt( $00000002); 3196
16054: LF32_MOVEABLE', 'LongWord.SetUInt( $00000004); 3197
16055: Ln10', 'Extended.setExtended( 2.3025850929940456840179914546844); 1687
16056: Ln10', 'Extended.setExtended( 2.30258509299404568402); 3200
16057: Ln10', 'Float.SetString( 2.3025850929940456840179914546844); 1299
16058: Ln10', 'Float.SetString( 2.3025850929940456840179914546844); 1367
16059: Ln2', 'Extended.setExtended( 0.69314718055994530941723212145818); 1686
16060: Ln2', 'Extended.setExtended( 0.69314718055994530942); 3199
16061: Ln2', 'Float.SetString( 0.69314718055994530941723212145818); 1298
16062: Ln2', 'Float.SetString( 0.69314718055994530941723212145818); 1366
16063: Ln2PiDiv2', 'Extended.setExtended( 0.91893853320467274178); 3210
16064: LnPi', 'Extended.setExtended( 1.14472988584940017414); 3201
16065: LnPi', 'Extended.setExtended( 1.1447298858494001741434273513531); 1688
16066: LnPi', 'Float.SetString( 1.1447298858494001741434273513531); 1300

```

```

16067: LnPi', 'Float.SetString( 1.1447298858494001741434273513531); 1368
16068: LnSqrt2Pi', 'Extended.setExtended( 0.91893853320467274178); 3209
16069: LOCALE_SYSTEM_DEFAULT', 'LongInt.SetInt((SORT_DEFAULT shl 16) or LANG_SYSTEM_DEFAULT); 356
16070: LOCALE_USER_DEFAULT', 'LongInt.SetInt((SORT_DEFAULT shl 16) or LANG_USER_DEFAULT); 357
16071: LOCK_EXCLUSIVE', 'LongInt.SetInt( 2); 2134
16072: LOCK_ONLYONCE', 'LongInt.SetInt( 4); 2135
16073: LOCK_WRITE', 'LongInt.SetInt( 1); 2133
16074: Log2', 'Float.SetString( 0.301026398119521373889472449); 1301
16075: Log2', 'Float.SetString( 0.30102999566398119521373889472449); 1369
16076: Log2J', 'Extended.setExtended( 0.30102999566398119521373889472449); 1689
16077: Log3', 'Extended.setExtended( 0.47712125471966243729502790325512); 1690
16078: Log3', 'Float.SetString( 0.47712125471966243729502790325512); 1302
16079: Log3', 'Float.SetString( 0.47712125471966243729502790325512); 1370
16080: LogE', 'Extended.setExtended( 0.43429448190325182765112891891661); 1692
16081: LogE', 'Float.SetString( 0.43429448190325182765112891891661); 1304
16082: LogE', 'Float.SetString( 0.43429448190325182765112891891661); 1372
16083: LogPi', 'Extended.setExtended( 0.4971498726941338543512682882909); 1691
16084: LogPi', 'Float.SetString( 0.4971498726941338543512682882909); 1303
16085: LogPi', 'Float.SetString( 0.4971498726941338543512682882909); 1371
16086: LongDateSub1', 'String.SetString( 'f'; 3018
16087: LongDateSub2', 'String.SetString( 'g'; 3019
16088: LongDateSub3', 'String.SetString( 'h'; 3020
16089: LoopStackMax', 'LongInt.SetInt( 10); 2060
16090: LowValidNaNTag', 'LongWord.SetUInt( - $FFFFFFF); 1331
16091: LowValidNaNTag', 'LongWord.SetUInt( - $FFFFFFF); 1399
16092: LWS', 'LongInt.Value.ts32 := ord(TAB) or ord(CHAR32); 1079
16093: MachEp', 'Extended.setExtended( 1.08420217248550444E-19); 3231
16094: MachEp', 'Extended.setExtended( 1.192093E-7); 3215
16095: MachEp', 'Extended.setExtended( 2.220446049250313E-16); 3223
16096: MapiAddressTypeFAX', 'String.SetString( 'FAX'; 3399
16097: MapiAddressTypeSMTP', 'String.SetString( 'SMTP'; 3398
16098: MatErrDim', 'LongInt.SetInt( - 3); 3249
16099: MatNonConv', 'LongInt.SetInt( - 1); 3247
16100: MatNotPD', 'LongInt.SetInt( - 4); 3250
16101: MatOk', 'LongInt.SetInt( 0); 3246
16102: MatSing', 'LongInt.SetInt( - 2); 3248
16103: MAX_ADAPTER_ADDRESS_LENGTH', 'LongInt.SetInt( 8); 3434
16104: MAX_ADAPTER_DESCRIPTION_LENGTH', 'LongInt.SetInt( 128); 3432
16105: MAX_ADAPTER_NAME_LENGTH', 'LongInt.SetInt( 256); 3433
16106: MAX_DATETIMES', 'LongInt.SetInt( 3); 992
16107: MAX_DIAGONAL', 'LongWord.SetUInt( $FFFFFFF); 3391
16108: MAX_DOMAIN_NAME_LEN', 'LongInt.SetInt( 128); 3437
16109: MAX_HOSTNAME_LEN', 'LongInt.SetInt( 128); 3436
16110: MAX_INT64', 'Int64.SetInt64( 9223372036854775807); 3169
16111: MAX_INT64', 'Int64.SetInt64( 9223372036854775807); 3358
16112: MAX_INTEGERS', 'LongInt.SetInt( 1); 991
16113: MAX_MARKS', 'LongInt.SetInt( 16); 3038
16114: MAX_MODULE_NAME32', 'LongInt.SetInt( 255); 3187
16115: MAX_PACKET_SIZE', 'LongInt.SetInt( 1024); 1121
16116: MAX_PATH', 'LongInt.SetInt( 4095); 358
16117: MAX_SCOPE_ID_LEN', 'LongInt.SetInt( 256); 3438
16118: MAX_SCROLL', 'LongInt.SetInt( 32767); 3037
16119: MAX_STRINGS', 'LongInt.SetInt( 12); 990
16120: MAX_SUBEXPRESSIONS', 'LongInt.SetInt( 99); 1722
16121: MAX_TOKEN_SIZE', 'LongInt.SetInt( 32768); 3376
16122: MaxAngle', 'Extended.setExtended( 9223372036854775808.0); 1699
16123: MaxAngle', 'Float.SetString( 9223372036854775808.0); 1311
16124: MaxAngle', 'Float.SetString( 9223372036854775808.0); 1379
16125: MaxArg', 'LongInt.SetInt( 26); 3259
16126: MaxBcdPrecision', 'LongInt.SetInt( 18); 983
16127: MaxBcdScale', 'LongInt.SetInt( 4); 984
16128: MaxBracesArg', 'LongInt.SetInt( $FFFFFFFF - 1); 2059
16129: MaxByte', 'Byte.SetInt(maxbyte); 2190
16130: MaxCardinal', 'Cardinal.SetUInt(maxcardinal); 2206
16131: MAXCOLS', 'LongInt.SetInt( 20); 3331
16132: MaxComp', 'Extended.setExtended(9.223372036854775807e+18); 1663
16133: MaxCurrency', 'Currency.SetExtended( 922337203685477.5807); 2220
16134: MaxCustomColors', 'LongInt.SetInt( 16); 909
16135: MaxCustomExtents', 'SetString(MaxCustomExtents); 817
16136: Maxdate', 'LongWord.SetUInt( $000D6025); 2991
16137: MaxDateTime', 'TDateTime.SetExtended(2958465.99999); 48
16138: MAXDIGITS', 'LongInt.SetInt( 16); 988
16139: MaxDigArgs', 'LongInt.SetInt( 32); 2867
16140: MaxDouble', 'Double.setExtended( 1.7E+308); 2216
16141: MaxDouble', 'Extended.setExtended( 1.7e+308); 1659
16142: MaxExpLen', 'LongInt.SetInt( 4); 2733
16143: MaxExtended', 'Extended.setExtended(1.1e+4932); 1661
16144: MaxExtended', 'Extended.setExtended(1.1E+4932); 2218
16145: MaxFac', 'LongInt.SetInt( 170); 3228
16146: MaxFac', 'LongInt.SetInt( 1754); 3236
16147: MaxFac', 'LongInt.SetInt( 33); 3220
16148: MaxFactorial', 'LongInt.SetInt( 170); 1317
16149: MaxFactorial', 'LongInt.SetInt( 170); 1385
16150: MaxFactorial', 'LongInt.SetInt( 170); 1705
16151: MaxFactorial', 'LongInt.SetInt( 1754); 1313
16152: MaxFactorial', 'LongInt.SetInt( 1754); 1381
16153: MaxFactorial', 'LongInt.SetInt( 1754); 1701
16154: MaxFactorial', 'LongInt.SetInt( 33); 1321
16155: MaxFactorial', 'LongInt.SetInt( 33); 1389

```

```

16156: MaxFactorial', 'LongInt.SetInt( 33); 1709
16157: MaxFloat', 'SetString( MaxDouble); 1719
16158: MaxFloat', 'SetString( MaxSingle); 1718
16159: MaxFloatingPoint', 'Extended.SetExtended(1.189731495357231765085759326628E+4932); 1702
16160: MaxFloatingPoint', 'Float.SetString( 1.189731495357231765085759326628E+4932); 1314
16161: MaxFloatingPoint', 'Float.SetString( 1.189731495357231765085759326628E+4932); 1382
16162: MaxFloatingPoint', 'Float.SetString( 1.797693134862315907729305190789E+308); 1318
16163: MaxFloatingPoint', 'Float.SetString( 1.797693134862315907729305190789E+308); 1386
16164: MaxFloatingPoint', 'Float.SetString( 3.4028236692093846346337460743177E+38); 1322
16165: MaxFloatingPoint', 'Float.SetString( 3.4028236692093846346337460743177E+38); 1390
16166: MaxFloatingPointD', 'Extended.SetExtended(1.797693134862315907729305190789E+308); 1706
16167: MaxFloatingPointS', 'Extended.SetExtended( 3.4028236692093846346337460743177E+38); 1710
16168: MaxFMTBcdDigits', 'LongInt.SetInt( 32); 981
16169: MaxFMTBcdFractionSize', 'LongInt.SetInt( 64); 980
16170: MaxFuncNameLen', 'LongInt.SetInt( 5); 2730
16171: MaxGam', 'Extended.setExtended( 171.624376956302); 3229
16172: MaxGam', 'Extended.setExtended( 1755.455); 3237
16173: MaxGam', 'Extended.setExtended( 34.648); 3221
16174: MaxHugeListSize', 'LongInt.SetInt(MaxListSize); 2742
16175: MAXIMUM_ASYNC_WAIT_OBJECTS', 'LongInt.SetInt( MAXIMUM_WAIT_OBJECTS - 3); 818
16176: MaxInitStrNum', 'LongInt.SetInt( 9); 2710
16177: MaxInt', 'LongInt.SetInt(2147483647); 41
16178: MaxInt64', 'Int64.SetInt64(maxint64); 2202
16179: MaxInteger', 'Integer.SetInt(maxinteger); 2204
16180: MaxLgm', 'Extended.setExtended( 1.0383E+36); 3222
16181: MaxLgm', 'Extended.setExtended( 1.04848146839019521E+4928); 3238
16182: MaxLgm', 'Extended.setExtended( 2.556348E+305); 3230
16183: MaxListSize', 'LongInt.SetInt( Maxint div 16); 819
16184: MaxListSize', 'LongInt.SetInt(Maxint div 16); 482
16185: MaxLog', 'Extended.setExtended( 11355.830259113584004); 3234
16186: MaxLog', 'Extended.setExtended( 709.7827128933840); 3226
16187: MaxLog', 'Extended.setExtended( 88.72283); 3218
16188: MaxLongInt', 'LongInt.SetInt(maxshortint); 2200
16189: MaxLongWord', 'LongWord.SetInt(maxlongword); 2198
16190: MaxMIMEBinToASCIIType', 'LongInt.SetInt( 2); 2432
16191: MaxMIMECompressType', 'LongInt.SetInt( 0); 2439
16192: MaxMIMEMessageDigestType', 'LongInt.SetInt( 3); 2437
16193: MaxMIMESubTypes', 'LongInt.SetInt( 1); 2428
16194: MaxMIMETYPE', 'LongInt.SetInt( 6); 2425
16195: MaxNativeInt', 'NativeInt.SetInt(maxnativeint); 2210
16196: MaxNativeUInt', 'NativeUInt.SetUInt(maxnativeuint); 2208
16197: MaxNum', 'Extended.setExtended( 1.797693134862315E+308); 3224
16198: MaxNum', 'Extended.setExtended( 3.402823E+38); 3216
16199: MaxNum', 'Extended.setExtended( 5.9486574767861588254E+4931); 3232
16200: MaxPixelCount', 'LongInt.SetInt( 32767); 1598
16201: MaxPort', 'LongInt.SetInt( 4); 1738
16202: MaxProfEntries', 'LongInt.SetInt( 1024); 2743
16203: MaxShortInt', 'ShortInt.SetInt(maxshortint); 2194
16204: MaxShowXValueInLegends', 'LongInt.SetInt( 10); 2702
16205: MaxSingle', 'Extended.setExtended( 3.4E+38); 1657
16206: MaxSingle', 'Single.setExtended( 3.4E+38); 2214
16207: MaxSize', 'LongInt.SetInt( 2147483647); 3258
16208: MaxSize', 'LongInt.SetInt( 32767); 3257
16209: MaxSmallInt', 'SmallInt.SetInt(maxsmallint); 2196
16210: MaxStackSize', 'LongInt.SetInt( 1024); 2744
16211: MaxStringDigits', 'LongInt.SetInt( 100); 977
16212: MaxSupported', 'LongInt.SetInt( 99); 1739
16213: MAXTABS', 'LongInt.SetInt( 20); 3332
16214: MaxTanH', 'Extended.SetExtended( 354.89135644669199842162284618659); 1704
16215: MaxTanH', 'Extended.SetExtended( 44.361419555836499802702855773323); 1708
16216: MaxTanH', 'Extended.SetExtended( 5678.2617031470719747459655389854); 1700
16217: MaxTanH', 'Float.SetString( 354.89135644669199842162284618659); 1316
16218: MaxTanH', 'Float.SetString( 354.89135644669199842162284618659); 1384
16219: MaxTanH', 'Float.SetString( 44.361419555836499802702855773323); 1320
16220: MaxTanH', 'Float.SetString( 44.361419555836499802702855773323); 1388
16221: MaxTanH', 'Float.SetString( 5678.2617031470719747459655389854); 1312
16222: MaxTanH', 'Float.SetString( 5678.2617031470719747459655389854); 1380
16223: MaxTerm', 'Byte.SetUInt( 35); 2449
16224: MaxTime', 'LongInt.SetInt( 86399); 3000
16225: MaxWord', 'Word.SetInt(maxword); 2192
16226: MaxYear', 'LongInt.SetInt( 3999); 2989
16227: MB_ABORTRETRYIGNORE', 'LongWord.SetUInt( $00000002); 73
16228: MB_APPLMODAL', 'LongWord.SetUInt( $00000000); 86
16229: MB_DEFAULT_DESKTOP_ONLY', 'LongWord.SetUInt( $00020000); 92
16230: MB_DEFMASK', 'LongWord.SetUInt( $00000F00); 100
16231: MB_HELP', 'LongWord.SetUInt( $00004000); 89
16232: MB_ICONASTERISK', 'LongWord.SetUInt( $00000040); 80
16233: MB_ICONERROR', 'longword.SetUInt( MB_ICONHAND); 83
16234: MB_ICONEXCLAMATION', 'LongWord.SetUInt( $00000030); 79
16235: MB_ICONHAND', 'LongWord.SetUInt( $00000010); 77
16236: MB_ICONINFORMATION', 'longword.SetUInt( MB_ICONASTERISK); 84
16237: MB_ICONMASK', 'LongWord.SetUInt( $000000F0); 99
16238: MB_ICONQUESTION', 'LongWord.SetUInt( $00000020); 78
16239: MB_ICONSTOP', 'longword.SetUInt( MB_ICONHAND); 85
16240: MB_ICONWARNING', 'longword.SetUInt( MB_ICONEXCLAMATION); 82
16241: MB_MISCMASK', 'LongWord.SetUInt( $0000C000); 102
16242: MB_MODEMASK', 'LongWord.SetUInt( $00003000); 101
16243: MB_NOFOCUS', 'LongWord.SetUInt( $00008000); 90
16244: MB_OK', 'LongWord.SetUInt( $00000000); 71

```



```

16245: MB_OKCANCEL', 'LongWord.SetUInt( $00000001); 72
16246: MB_RETRYCANCEL', 'LongWord.SetUInt( $00000005); 76
16247: MB_RIGHT', 'LongWord.SetUInt( $00080000); 94
16248: MB_RTLREADING', 'LongWord.SetUInt( $00100000); 95
16249: MB_SERVICE_NOTIFICATION', 'LongWord.SetUInt( $00200000); 96
16250: MB_SERVICE_NOTIFICATION_NT3X', 'LongWord.SetUInt( $00040000); 97
16251: MB_SETFOREGROUND', 'LongWord.SetUInt( $00010000); 91
16252: MB_SYSTEMMODAL', 'LongWord.SetUInt( $00001000); 87
16253: MB_TASKMODAL', 'LongWord.SetUInt( $00002000); 88
16254: MB_TOPMOST', 'LongWord.SetUInt( $00040000); 93
16255: MB_TYPEMASK', 'LongWord.SetUInt( $0000000F); 98
16256: MB_USERICON', 'LongWord.SetUInt( $00000080); 81
16257: MB_YESNO', 'LongWord.SetUInt( $00000004); 75
16258: MB_YESNOCANCEL', 'LongWord.SetUInt( $00000003); 74
16259: mbAbortIgnore', 'LongInt.Value.ts32 := ord(mbAbort) or ord(mbIgnore); 920
16260: mbAbortIgnore', 'LongInt.Value.ts32 := ord(mbAbort) or ord(mbIgnore); 927
16261: mbAbortRetryIgnore', 'LongInt.Value.ts32 := ord(mbAbort) or ord(mbRetry) or ord(mbIgnore); 919
16262: mbAbortRetryIgnore', 'LongInt.Value.ts32 := ord(mbAbort) or ord(mbRetry) or ord(mbIgnore); 926
16263: mbOKCancel', 'LongInt.Value.ts32 := ord(mbOK) or ord(mbCancel); 918
16264: mbOKCancel', 'LongInt.Value.ts32 := ord(mbOK) or ord(mbCancel); 925
16265: MBVER', 'String.SetString( '399; 969
16266: MBVER', 'String.SetString( '399; 948
16267: MBVERI', 'Integer.SetInt(399); 949
16268: MBVERIALL', 'Integer').SetInt(39986);
16269: MBVERSION', 'String.SetString( '3.9.9.94; 9986
16270: MBVERSION', 'String.SetString( '3.9.9.94; 9986
16271: mbYesAllNoAllCancel', 'LongInt.Value.ts32:= ord(mbYes) or ord(mbYesToAll) or ord(mbNo) or ord(mbNoToAll) or
ord(mbCancel); 917
16272: mbYesAllNoAllCancel', 'LongInt.Value.ts32:= ord(mbYes) or ord(mbYesToAll) or ord(mbNo) or ord(mbNoToAll) or
ord(mbCancel); 924
16273: mbYesNo', 'LongInt.SetSet( 'mbYes,mbNo; 922
16274: mbYesNo', 'LongInt.Value.ts32 := ord(mbYes) or ord(mbNo); 915
16275: mbYesNo', 'LongInt.Value.ts32 := ord(mbYes) or ord(mbNo); 921
16276: mbYesNoCancel', 'LongInt.Value.ts32 := ord(mbYes) or ord(mbNo) or ord(mbCancel); 916
16277: mbYesNoCancel', 'LongInt.Value.ts32 := ord(mbYes) or ord(mbNo) or ord(mbCancel); 923
16278: MCA_FLAG', 'LongWord.SetUInt( $00004000); 1438
16279: MCE_FLAG', 'LongWord.SetUInt( $00000080); 1431
16280: mDirLiteral', 'String.SetString( '\'; 1644
16281: mDirLowerCase', 'String.SetString( '<; 1643
16282: mDirReverse', 'String.SetString( '!; 1641
16283: mDirUpperCase', 'String.SetString( '>; 1642
16284: MedianThreshold', 'LongInt.SetInt( 16); 3031
16285: MERGECOPY', 'LongWord.SetUInt( $00C000CA); 791
16286: MergeOrder', 'LongInt.SetInt( 5); 3030
16287: MERGEPaint', 'LongWord.SetUInt( $00BB0226); 792
16288: MetaDataOption', 'Integer.SetInt( 1 shl ord( grMetaData)); 1732
16289: MetersPerAstronomicalUnit', 'Extended.setExtended( 1.49598E11); 1938
16290: MetersPerChain', 'Extended.setExtended( MetersPerRod * 4); 1948
16291: MetersPerCubit', 'Extended.setExtended( 0.4572); 1942
16292: MetersPerFathom', 'Extended.setExtended( MetersPerFoot * 6); 1943
16293: MetersPerFoot', 'Extended.setExtended( MetersPerInch * 12); 1934
16294: MetersPerFurlong', 'Extended.setExtended( MetersPerYard * 220); 1944
16295: MetersPerHand', 'Extended.setExtended( MetersPerInch * 4); 1945
16296: MetersPerInch', 'Extended.setExtended( 0.0254); 1933
16297: MetersPerLightSecond', 'Extended.setExtended( 2.99792458E8); 1939
16298: MetersPerLightYear', 'Extended.setExtended( MetersPerLightSecond * 31556925.9747); 1940
16299: MetersPerLink', 'Extended.setExtended( MetersPerChain / 100); 1949
16300: MetersPerMile', 'Extended.setExtended( MetersPerFoot * 5280); 1936
16301: MetersPerNauticalMiles', 'LongInt.SetInt( 1852); 1937
16302: MetersPerPace', 'Extended.setExtended( MetersPerInch * 30); 1946
16303: MetersPerParsec', 'Extended.setExtended( MetersPerAstronomicalUnit * 206264.806247096); 1941
16304: MetersPerPica', 'Extended.setExtended( MetersPerPoint * 12); 1951
16305: MetersPerPoint', 'Extended.setExtended( MetersPerInch * 0.013837); 1950
16306: MetersPerRod', 'Extended.setExtended( MetersPerFoot * 16.5); 1947
16307: MetersPerYard', 'Extended.setExtended( MetersPerFoot * 3); 1935
16308: MIDICAllNotesOff', 'LongWord.SetUInt( $7B); 2543
16309: MIDICAllSoundOff', 'LongWord.SetUInt( $78); 2540
16310: MIDICCBalance', 'LongWord.SetUInt( $08); 2479
16311: MIDICCBalanceLSB', 'LongWord.SetUInt( $28); 2496
16312: MIDICCBankSelect', 'LongWord.SetUInt( $00); 2471
16313: MIDICCBankSelectLSB', 'LongWord.SetUInt( $20); 2488
16314: MIDICCBreathControl', 'LongWord.SetUInt( $02); 2473
16315: MIDICCBreathControlLSB', 'LongWord.SetUInt( $22); 2490
16316: MIDICCCelesteDepth', 'Longword.SetUInt( MIDICCEffects4Depth); 2531
16317: MIDICChannelVolume', 'LongWord.SetUInt( $07); 2477
16318: MIDICChannelVolumeLSB', 'LongWord.SetUInt( $27); 2494
16319: MIDICChorusSendLevel', 'LongWord.SetUInt( $5D); 2529
16320: MIDICDataEntry', 'LongWord.SetUInt( $06); 2476
16321: MIDICDataEntryDec', 'LongWord.SetUInt( $61); 2535
16322: MIDICDataEntryInc', 'LongWord.SetUInt( $60); 2534
16323: MIDICDataEntryLSB', 'LongWord.SetUInt( $26); 2493
16324: MIDICEffectControl', 'LongWord.SetUInt( $0C); 2482
16325: MIDICEffectControl2', 'LongWord.SetUInt( $0D); 2483
16326: MIDICEffectControl2LSB', 'LongWord.SetUInt( $2D); 2500
16327: MIDICEffectControlLSB', 'LongWord.SetUInt( $2C); 2499
16328: MIDICCEffects2Depth', 'LongWord.SetUInt( $5C); 2527
16329: MIDICCEffects4Depth', 'LongWord.SetUInt( $5E); 2530
16330: MIDICCEffects5Depth', 'LongWord.SetUInt( $5F); 2532
16331: MIDICCEXpression', 'LongWord.SetUInt( $0B); 2481

```

```

16332: MIDICCEXpressionLSB', 'LongWord.SetUInt( $2B); 2498
16333: MIDICCFootController', 'LongWord.SetUInt( $04); 2474
16334: MIDICCFootControllerLSB', 'LongWord.SetUInt( $24); 2491
16335: MIDICCGeneralPurpose1', 'LongWord.SetUInt( $10); 2484
16336: MIDICCGeneralPurpose1LSB', 'LongWord.SetUInt( $30); 2501
16337: MIDICCGeneralPurpose2', 'LongWord.SetUInt( $11); 2485
16338: MIDICCGeneralPurpose2LSB', 'LongWord.SetUInt( $31); 2502
16339: MIDICCGeneralPurpose3', 'LongWord.SetUInt( $12); 2486
16340: MIDICCGeneralPurpose3LSB', 'LongWord.SetUInt( $32); 2503
16341: MIDICCGeneralPurpose4', 'LongWord.SetUInt( $13); 2487
16342: MIDICCGeneralPurpose4LSB', 'LongWord.SetUInt( $33); 2504
16343: MIDICCGeneralPurpose5', 'LongWord.SetUInt( $50); 2521
16344: MIDICCGeneralPurpose6', 'LongWord.SetUInt( $51); 2522
16345: MIDICCGeneralPurpose7', 'LongWord.SetUInt( $52); 2523
16346: MIDICCGeneralPurpose8', 'LongWord.SetUInt( $53); 2524
16347: MIDICCHold2', 'LongWord.SetUInt( $45); 2510
16348: MIDICCLegato', 'LongWord.SetUInt( $44); 2509
16349: MIDICCLocalControl', 'LongWord.SetUInt( $7A); 2542
16350: MIDICCMainVolume', 'LongWord.SetUInt( MIDICChannelVolume); 2478
16351: MIDICCMainVolumeLSB', 'LongWord.SetUInt( MIDICChannelVolumeLSB); 2495
16352: MIDICCModulationWheel', 'LongWord.SetUInt( $01); 2472
16353: MIDICCModulationWheelLSB', 'LongWord.SetUInt( $21); 2489
16354: MIDICCMonoModeOn', 'LongWord.SetUInt( $7E); 2546
16355: MIDICCNNonRegParamNumLSB', 'LongWord.SetUInt( $62); 2536
16356: MIDICCNNonRegParamNumMSB', 'LongWord.SetUInt( $63); 2537
16357: MIDICCOmniModeOff', 'LongWord.SetUInt( $7C); 2544
16358: MIDICCOmniModeOn', 'LongWord.SetUInt( $7D); 2545
16359: MIDICCPan', 'LongWord.SetUInt( $0A); 2480
16360: MIDICCPanLSB', 'LongWord.SetUInt( $2A); 2497
16361: MIDICCPHaserDepth', 'LongWord.SetUInt( MIDICEffects5Depth); 2533
16362: MIDICCPolyModeOn', 'LongWord.SetUInt( $7F); 2547
16363: MIDICCPortamento', 'LongWord.SetUInt( $41); 2506
16364: MIDICCPortamentoControl', 'LongWord.SetUInt( $54); 2525
16365: MIDICCPortamentoTime', 'LongWord.SetUInt( $05); 2475
16366: MIDICCPortamentoTimeLSB', 'LongWord.SetUInt( $25); 2492
16367: MIDICCRRegParamNumLSB', 'LongWord.SetUInt( $64); 2538
16368: MIDICCRRegParamNumMSB', 'LongWord.SetUInt( $65); 2539
16369: MIDICCRResetAllControllers', 'LongWord.SetUInt( $79); 2541
16370: MIDICCRverbSendLevel', 'LongWord.SetUInt( $5B); 2526
16371: MIDICCSOftPedal', 'LongWord.SetUInt( $43); 2508
16372: MIDICCSound1', 'LongWord.SetUInt( $46); 2511
16373: MIDICCSound10', 'LongWord.SetUInt( $4F); 2520
16374: MIDICCSound2', 'LongWord.SetUInt( $47); 2512
16375: MIDICCSound3', 'LongWord.SetUInt( $48); 2513
16376: MIDICCSound4', 'LongWord.SetUInt( $49); 2514
16377: MIDICCSound5', 'LongWord.SetUInt( $4A); 2515
16378: MIDICCSound6', 'LongWord.SetUInt( $4B); 2516
16379: MIDICCSound7', 'LongWord.SetUInt( $4C); 2517
16380: MIDICCSound8', 'LongWord.SetUInt( $4D); 2518
16381: MIDICCSound9', 'LongWord.SetUInt( $4E); 2519
16382: MIDICCSustain', 'LongWord.SetUInt( $40); 2505
16383: MIDICCSustenuto', 'LongWord.SetUInt( $42); 2507
16384: MIDICCTremoloDepth', 'LongWord.SetUInt( MIDICEffects2Depth); 2528
16385: MIDICHannelMsgMask', 'LongWord.SetUInt( $F0); 2550
16386: MIDIDataMask', 'LongWord.SetUInt( $7F); 2548
16387: MIDIDataWordMask', 'LongWord.SetUInt( $3FFF); 2549
16388: MIDIInvalidStatus', 'LongInt.SetInt( TMIDIStatusByte ( 0 )); 2551
16389: MIDIMsgActiveSensing', 'LongWord.SetUInt( $FE); 2469
16390: MIDIMsgAftertouch', 'LongWord.SetUInt( MIDIMsgChannelKeyPressure); 2457
16391: MIDIMsgChannelKeyPressure', 'LongWord.SetUInt( $D0); 2456
16392: MIDIMsgContinueSequence', 'LongWord.SetUInt( $FB); 2467
16393: MIDIMsgControlChange', 'LongWord.SetUInt( $B0); 2454
16394: MIDIMsgEOX', 'LongWord.SetUInt( $F7); 2464
16395: MIDIMsgMTCQtrFrame', 'LongWord.SetUInt( $F1); 2460
16396: MIDIMsgNoteOff', 'LongWord.SetUInt( $80); 2451
16397: MIDIMsgNoteOn', 'LongWord.SetUInt( $90); 2452
16398: MIDIMsgPitchWheelChange', 'LongWord.SetUInt( $E0); 2458
16399: MIDIMsgPolyKeyPressure', 'LongWord.SetUInt( $A0); 2453
16400: MIDIMsgProgramChange', 'LongWord.SetUInt( $C0); 2455
16401: MIDIMsgSongPositionPtr', 'LongWord.SetUInt( $F2); 2461
16402: MIDIMsgSongSelect', 'LongWord.SetUInt( $F3); 2462
16403: MIDIMsgStartSequence', 'LongWord.SetUInt( $FA); 2466
16404: MIDIMsgStopSequence', 'LongWord.SetUInt( $FC); 2468
16405: MIDIMsgSysEx', 'LongWord.SetUInt( $F0); 2459
16406: MIDIMsgSystemReset', 'LongWord.SetUInt( $FF); 2470
16407: MIDIMsgTimingClock', 'LongWord.SetUInt( $F8); 2465
16408: MIDIMsgTuneRequest', 'LongWord.SetUInt( $F6); 2463
16409: MIM7Bit', 'String.SetString( '7bit'; 2415
16410: MIMEncBase64', 'String.SetString( 'base64'; 2429
16411: MIMEncNISTSHA', 'String.SetString( MIMEXVal + 'nist-sha'; 2436
16412: MIMEncRLECompress', 'String.SetString( MIMEXVal + 'rle-compress'; 2438
16413: MIMEncRSAMD2', 'String.SetString( MIMEXVal + 'rsa-md2'; 2433
16414: MIMEncRSAMD4', 'String.SetString( MIMEXVal + 'rsa-md4'; 2434
16415: MIMEncRSAMD5', 'String.SetString( MIMEXVal + 'rsa-md5'; 2435
16416: MIMEncUUEncode', 'String.SetString( MIMEXVal + 'uu'; 2430
16417: MIMEncXXEncode', 'String.SetString( MIMEXVal + 'xx'; 2431
16418: MIMEGenericText', 'String.SetString( 'text/'; 2414
16419: MIMESplit', 'String.SetString( '/'; 2416
16420: MIMESubMacBinHex40', 'String.SetString( 'mac-binhex40'; 2427

```

```

16421: MIMESubOctetStream', 'String.SetString( 'octet-stream; 2426
16422: MIMETYPEApplication', 'String.SetString( 'application' + '/; 2418
16423: MIMETYPEAudio', 'String.SetString( 'audio' + '/; 2419
16424: MIMETYPEImage', 'String.SetString( 'image' + '/; 2420
16425: MIMETYPEMessage', 'String.SetString( 'message' + '/; 2421
16426: MIMETYPEMultipart', 'String.SetString( 'multipart' + '/; 2422
16427: MIMETYPEText', 'String.SetString( 'text' + '/; 2423
16428: MIMETYPEVideo', 'String.SetString( 'video' + '/; 2424
16429: MIMEXVal', 'String.SetString( 'x-; 2417
16430: MinByte', 'Byte.setInt(minbyte); 2189
16431: MinCardinal', 'Cardinal.SetUInt(mincardinal); 2205
16432: MinComp', 'Extended.setExtended(- 9.223372036854775807e+18); 1662
16433: MinCurrency', 'Currency.SetExtended( - 922337203685477.5807); 2219
16434: Mindate', 'LongWord.SetUInt( $00000000); 2990
16435: MinDateTime', 'TDateTime.SetExtended(-657434.0); 47
16436: MinDouble', 'Double.setExtended( 5.0E-324); 2215
16437: MinDouble', 'Extended.setExtended( 5.0e-324); 1658
16438: MinExtended', 'Extended.setExtended(3.4e-4932); 1660
16439: MinExtended', 'Extended.setExtended(3.4E-4932); 2217
16440: MinFloatingPoint', 'Extended.SetExtended(3.3621031431120935062626778173218E-4932); 1703
16441: MinFloatingPoint', 'Float.SetString( 1.1754943508222875079687365372222E-38); 1323
16442: MinFloatingPoint', 'Float.SetString( 1.1754943508222875079687365372222E-38); 1391
16443: MinFloatingPoint', 'Float.SetString( 2.2250738585072013830902327173324E-308); 1319
16444: MinFloatingPoint', 'Float.SetString( 2.2250738585072013830902327173324E-308); 1387
16445: MinFloatingPoint', 'Float.SetString( 3.3621031431120935062626778173218E-4932); 1315
16446: MinFloatingPoint', 'Float.SetString( 3.3621031431120935062626778173218E-4932); 1383
16447: MinFloatingPointD', 'Extended.SetExtended(2.2250738585072013830902327173324E-308); 1707
16448: MinFloatingPointS', 'Extended.SetExtended( 1.1754943508222875079687365372222E-38); 1711
16449: MinInt64', 'Int64.SetInt64(minint64); 2201
16450: MinInteger', 'Integer.SetInt(mininteger); 2203
16451: MinLog', 'Extended.setExtended( - 11354.443964752464114); 3235
16452: MinLog', 'Extended.setExtended( - 708.3964185322641); 3227
16453: MinLog', 'Extended.setExtended( - 87.33655); 3219
16454: MinLongInt', 'LongInt.SetInt(minshortint); 2199
16455: MinLongWord', 'LongWord.SetInt(minlongword); 2197
16456: MinNativeInt', 'NativeInt.SetInt(minnativeint); 2209
16457: MinNativeUInt', 'NativeUInt.SetUInt(minnativeuint); 2207
16458: MinNum', 'Extended.setExtended( 1.175495E-38); 3217
16459: MinNum', 'Extended.setExtended( 2.225073858507202E-308); 3225
16460: MinNum', 'Extended.setExtended( 6.7242062862241870125E-4932); 3233
16461: MinOnly', 'String.SetString( 'm; 3022
16462: MinOnlyU', 'String.SetString( 'M; 3025
16463: MinRecsPerRun', 'LongInt.SetInt( 4); 3029
16464: MinShortInt', 'ShortInt.SetInt(minshortint); 2193
16465: MinSingle', 'Extended.setExtended( 1.5e-45); 1656
16466: MinSingle', 'Single.setExtended( 1.5E-45); 2213
16467: MinSmallInt', 'SmallInt.SetInt(minsmallint); 2195
16468: MinsPerDay', 'LongInt.SetInt( HoursPerDay * 60); 3330
16469: MinsPerDay', 'LongInt.SetInt(HoursPerDay * MinsPerHour); 36
16470: MinsPerHour', 'LongInt.SetInt( 60); 33
16471: MinTime', 'LongInt.SetInt( 0); 2999
16472: MinutesInDay', 'LongInt.SetInt( 1440); 3007
16473: MinutesInHour', 'LongInt.SetInt( 60); 3006
16474: MinutesPerHour', 'Extended.SetExtended( 60); 2181
16475: MinWord', 'Word.SetInt(minword); 2191
16476: MinYear', 'LongInt.SetInt( 1600); 2988
16477: MK_CONTROL', 'LongWord.SetUInt( $0008); 2644
16478: MK_LBUTTON', 'LongWord.SetUInt( $0001); 2641
16479: MK_MBUTTON', 'LongWord.SetUInt( $0010); 2645
16480: MK_RBUTTON', 'LongWord.SetUInt( $0002); 2642
16481: MK_SHIFT', 'LongWord.SetUInt( $0004); 2643
16482: MK_XBUTTON1', 'LongWord.SetUInt( $0020); 2646
16483: MK_XBUTTON2', 'LongWord.SetUInt( $0040); 2647
16484: mMskAlpha', 'String.SetString( 'L; 1645
16485: mMskAlphaNum', 'String.SetString( 'A; 1647
16486: mMskAlphaNumOpt', 'String.SetString( 'a; 1648
16487: mMskAlphaOpt', 'String.SetString( 'l; 1646
16488: mMskAscii', 'String.SetString( 'C; 1649
16489: mMskAsciiOpt', 'String.SetString( 'c; 1650
16490: mMskDateSeparator', 'String.SetString( '/; 1655
16491: mMskNumeric', 'String.SetString( '0; 1651
16492: mMskNumericOpt', 'String.SetString( '9; 1652
16493: mMskNumSymOpt', 'String.SetString( '#; 1653
16494: mMskTimeSeparator', 'String.SetString( ':; 1654
16495: MMX_FLAG', 'LongWord.SetUInt( $00800000); 1447
16496: MONO_FONT', 'LongInt.SetInt( 8); 168
16497: MonthOnly', 'String.SetString( 'm; 3008
16498: MonthOnlyU', 'String.SetString( 'M; 3011
16499: MonthsPerYear', 'LongInt.SetInt( 12); 872
16500: MOUSEEVENTF_WHEEL', 'LongWord.SetUInt( $0800); 1637
16501: mrNoToAll', 'LongInt.SetInt( mrAll + 1); 1631
16502: mrYesToAll', 'LongInt.SetInt( mrNoToAll + 1); 1632
16503: Msecsperday', 'LongInt.SetInt(SecsPerDay * MsecsPerSec); 38
16504: MsecsPerSec', 'LongInt.SetInt( 1000); 35
16505: MsgDlgCharSet', 'Integer.SetInt( DEFAULT_CHARSET); 1625
16506: msimg32', 'String.SetString( 'msimg32.dll; 17
16507: MSR_FLAG', 'LongWord.SetUInt( $00000020); 1429
16508: MSWINDOWS_DEFAULT_FONTS', 'RawUTF8.SetString( 'Arial' #13#10 'Courier New' #13#10 'Georgia' #13#10 +
'Impact' #13#10 'Lucida Console' #13#10 'Roman' #13#10 'Symbol' #13#10 + 'Tahoma' #13#10 'Times New Roman'
#13#10 'Trebuchet' #13#10 + 'Verdana' #13#10 'WingDings; 3185

```

```

16509: MSWINDOWS_DEFAULT_FONTS', 'RawUTF8.SetString( 'Arial' #13#10 'Courier New' #13#10 'Georgia' #13#10 +
      'Impact' #13#10 'Lucida Console' #13#10 'Roman' #13#10 'Symbol' #13#10 + 'Tahoma' #13#10 'Times New Roman'
      #13#10 'Trebuchet' #13#10 + 'Verdana' #13#10 'WingDings; 3374
16510: MTRR_FLAG', 'LongWord.SetUInt( $00001000); 1436
16511: MUTANT_QUERY_STATE', 'LongWord.SetUInt( $0001); 199
16512: MUTEX_ALL_ACCESS', 'longint.SetUInt( MUTANT_ALL_ACCESS); 212
16513: MUTEX_MODIFY_STATE', 'longint.SetUInt( MUTANT_QUERY_STATE); 211
16514: MWT_IDENTITY', 'LongInt.SetInt( 1); 3165
16515: MWT_IDENTITY', 'LongInt.SetInt( 1); 3354
16516: MWT_LEFTMULTIPLY', 'LongInt.SetInt( 2); 3166
16517: MWT_LEFTMULTIPLY', 'LongInt.SetInt( 2); 3355
16518: MWT_RIGHTMULTIPLY', 'LongInt.SetInt( 3); 3167
16519: MWT_RIGHTMULTIPLY', 'LongInt.SetInt( 3); 3356
16520: MWT_SET', 'LongInt.SetInt( 4); 3168
16521: MWT_SET', 'LongInt.SetInt( 4); 3357
16522: mx_BOOLFORMAT', 'String.SetString( 'Format_Bool; 2162
16523: mx_CdsSkeleton', 'String.SetString( 'CdsSkeleton; 2147
16524: mx_CurrVersion', 'String.SetString( '1.0; 2151
16525: mx_DataEncoding', 'String.SetString( 'DataEncoding; 2153
16526: mx_Datapacket', 'String.SetString( 'DATAPACKET; 2169
16527: mx_DATEFORMAT', 'String.SetString( 'Format_Date; 2167
16528: mx_DATETIMEFORMAT', 'String.SetString( 'Format_DateTime; 2166
16529: mx_DEFAULT', 'String.SetString( 'Default; 2158
16530: mx_DEST', 'String.SetString( 'dest; 2164
16531: mx_Direction', 'String.SetString( 'Direction; 2154
16532: mx_FROM', 'String.SetString( 'from; 2165
16533: mx_ID', 'String.SetString( 'id; 2157
16534: mx_MAPVALUES', 'String.SetString( 'Map_Values; 2161
16535: mx_OPTIONAL', 'String.SetString( 'Optional; 2163
16536: mx_OPTIONS', 'String.SetString( 'Options; 2160
16537: mx_Root', 'String.SetString( 'XmlTransformation; 2141
16538: mx_RootName', 'String.SetString( 'RootName; 2152
16539: mx_ROW', 'String.SetString( 'ROW; 2171
16540: mx_ROWDATA', 'String.SetString( 'ROWDATA; 2170
16541: mx_RowState', 'String.SetString( 'RowState; 2172
16542: mx_Skeleton', 'String.SetString( 'Skeleton; 2143
16543: mx_TIMEFORMAT', 'String.SetString( 'Format_Time; 2168
16544: mx_ToCds', 'String.SetString( 'ToCds; 2156
16545: mx_ToXml', 'String.SetString( 'ToXml; 2155
16546: mx_Transform', 'String.SetString( 'Transform; 2142
16547: mx_Translate', 'String.SetString( 'Select; 2145
16548: mx_TranslateEach', 'String.SetString( 'SelectEach; 2144
16549: mx_VALUE', 'String.SetString( 'value; 2159
16550: mx_Version', 'String.SetString( 'Version; 2150
16551: mx_XmlSchema', 'String.SetString( 'XmlSchema; 2146
16552: mx_XmlSkeleton', 'String.SetString( 'XmlSkeleton; 2148
16553: mx_XSLTransform', 'String.SetString( 'XslTransform; 2149
16554: MXINTERNETCHECK', 'String.SetString( 'www.ask.com; 947
16555: MXINTERNETCHECK', 'String.SetString( 'www.ask.com; 973
16556: MXMAIL', 'String.SetString( 'max@kleiner.com; 951
16557: MXMAIL', 'String.SetString( 'max@kleiner.com; 974
16558: MXSITE', 'String.SetString( 'http://www.softwareschule.ch/maxbox.htm; 950
16559: MXSITE', 'String.SetString( 'http://www.softwareschule.ch/maxbox.htm; 971
16560: MXVERSIONFILE', 'String.SetString( 'http://www.softwareschule.ch/maxvfile.txt; 972
16561: NameOnly', 'String.SetString( 'n; 3014
16562: NameOnlyU', 'String.SetString( 'N; 3015
16563: NaN', 'Extended.setExtended( 0.0 / 0.0); 1664
16564: NaN', 'LongInt.SetInt( 0 / 0); 1330
16565: NaN', 'LongInt.SetInt( 0 / 0); 1398
16566: NegInfinity', 'Extended.setExtended( - 1.0 / 0.0); 1666
16567: NIF_ICON', 'LongWord.SetUInt( $00000002); 2938
16568: NIF_INFO', 'LongWord.SetUInt( $00000010); 2941
16569: NIF_MESSAGE', 'LongWord.SetUInt( $00000001); 2937
16570: NIF_STATE', 'LongWord.SetUInt( $00000008); 2940
16571: NIF_TIP', 'LongWord.SetUInt( $00000004); 2939
16572: NIIF_ERROR', 'LongWord.SetUInt( $00000003); 2945
16573: NIIF_ICON_MASK', 'LongWord.SetUInt( $0000000F); 2946
16574: NIIF_INFO', 'LongWord.SetUInt( $00000001); 2943
16575: NIIF_NONE', 'LongWord.SetUInt( $00000000); 2942
16576: NIIF_WARNING', 'LongWord.SetUInt( $00000002); 2944
16577: NIM_ADD', 'LongWord.SetUInt( $00000000); 2932
16578: NIM_DELETE', 'LongWord.SetUInt( $00000002); 2934
16579: NIM_MODIFY', 'LongWord.SetUInt( $00000001); 2933
16580: NIM_SETFOCUS', 'LongWord.SetUInt( $00000003); 2935
16581: NIM_SETVERSION', 'LongWord.SetUInt( $00000004); 2936
16582: NIN_BALLOONHIDE', 'LongInt.SetInt( $0400 + 3); 2950
16583: NIN_BALLOONSHOW', 'LongInt.SetInt( $0400 + 2); 2949
16584: NIN_BALLOONTIMEOUT', 'LongInt.SetInt( $0400 + 4); 2951
16585: NIN_BALLOONUSERCLICK', 'LongInt.SetInt( $0400 + 5); 2952
16586: NIN_SELECT', 'LongWord.SetUInt( $0400); 2947
16587: NINF_KEY', 'LongWord.SetUInt( $1); 2948
16588: NLMaxPar', 'LongInt.SetInt( - 6); 3255
16589: NLNullPar', 'LongInt.SetInt( - 7); 3256
16590: NoColorSelected', 'LongWord.SetUInt(TColor ( $FF000000 )); 2237
16591: NoHelp', 'LongInt.SetInt( 0); 1624
16592: NONANTIALIASSED_QUALITY', 'LongInt.SetInt( 3); 163
16593: NotCrcl6HighBit', 'LongWord.SetUInt( $7FFF); 1339
16594: NotCrcl6HighBit', 'LongWord.SetUInt( $7FFF); 1407
16595: NotCrc32HighBit', 'LongWord.SetUInt( $7FFFFFFF); 1346

```



```

16596: NotCrc32HighBit', 'LongWord.SetUInt( $7FFFFFFF); 1414
16597: NOTSRCCOPY', 'LongWord.SetUInt( $00330008); 789
16598: NOTSRCERASE', 'LongWord.SetUInt( $001100A6); 790
16599: NSUBEXP', 'LongInt.SetInt( 15); 2057
16600: NSUBEXPMAX', 'LongInt.SetInt( 255); 2058
16601: NTPMaxInt', 'Extended.setExtended( 4294967297.0); 1144
16602: NULL_TERMINATOR', 'Char.SetString( #0); 1044
16603: NullDate', 'TDateTime.SetString( 0); 1602
16604: NullDate', 'TDateTime.SetString( 0; 1589
16605: NullDate', 'TDateTime.SetString( 0; 1590
16606: NullDate', 'TDateTime.SetString( 0; 1619
16607: NullEquivalentDate', 'TDateTime.SetString( 0.0; 1622
16608: NullHandle', 'LongInt.SetInt( 0); 1599
16609: NullHandle', 'LongInt.SetInt( 0); 1615
16610: OB2000', 'Extended.setExtended( 0.409092804); 1930
16611: OB2000', 'Extended.setExtended( 0.409092804); 2976
16612: ObjectFieldTypes', 'LongInt.Value.ts32 := ord(ftADT) or ord(ftArray) or ord(ftReference) or ord(ftDataSet);
552
16613: OEM_CHARSET', 'LongInt.SetInt( 255); 176
16614: OFN_EX_NOPLACESBAR', 'LongInt.SetInt( 1); 2844
16615: ONE_HOUR', 'LongInt.SetInt( 1 / 24); 2746
16616: ONE_MILLISECOND', 'LongInt.SetInt( ONE_SECOND / 1000); 2749
16617: ONE_MINUTE', 'LongInt.SetInt( ONE_HOUR / 60); 2747
16618: ONE_SECOND', 'LongInt.SetInt( ONE_MINUTE / 60); 2748
16619: OneDay', 'Extended.setExtended( 1.0); 2174
16620: OneHour', 'Extended.setExtended( OneDay / 24); 2175
16621: OneMillisecond', 'Extended.setExtended( OneSecond / 1000); 2178
16622: OneMinute', 'Extended.setExtended( OneHour / 60); 2176
16623: OneSecond', 'Extended.setExtended( OneMinute / 60); 2177
16624: OneWeek', 'Extended.setExtended( OneDay * 7); 2179
16625: OnGetDefaultIniName', 'TOnGetDefaultIniName.SetString(); 1584
16626: OnHookInstaller', 'SetString(); 2000
16627: opengl32', 'String.SetString( 'opengl32.dll; 14
16628: OPF_PATHNAME', 'LongWord.SetUInt( $02); 2846
16629: OPF_PRINTERNAME', 'LongWord.SetUInt( $01); 2845
16630: OptBigLambda', 'LongInt.SetInt( - 5); 3254
16631: OptNonConv', 'LongInt.SetInt( - 1); 3252
16632: OptOk', 'LongInt.SetInt( 0); 3251
16633: OptSing', 'LongInt.SetInt( - 2); 3253
16634: OverflowChar', 'String.SetString( '*; 1932
16635: PAE_FLAG', 'LongWord.SetUInt( $00000040); 1430
16636: PAGE_FIT', 'LongInt.SetInt( - 2); 3336
16637: PAGE_WIDTH', 'LongInt.SetInt( - 1); 3335
16638: PAGENUMBER', 'String.SetString( '<<pagenumber>>; 3333
16639: PAPERSIZE_A4_HEIGHT', 'LongInt.SetInt( 297); 3353
16640: PAPERSIZE_A4_WIDTH', 'LongInt.SetInt( 210); 3352
16641: ParserStackSize', 'LongInt.SetInt( 15); 2729
16642: PAT_FLAG', 'LongWord.SetUInt( $00010000); 1440
16643: PATCOPY', 'LongWord.SetUInt( $00F00021); 793
16644: PathDelim', 'String.SetString( '/; 1613
16645: PathDelim', 'String.SetString( '\; 1609
16646: PathDevicePrefix', 'String.SetString( '\\.; 1254
16647: PathSep', 'String.SetString( '; 1611
16648: PathSeparator', 'String.SetString( '/' '\; 2183
16649: PathSeparator', 'String.SetString( '/; 1253
16650: PathSeparator', 'String.SetString( '\; 1255
16651: PathUncPrefix', 'String.SetString( '\\; 1256
16652: PATINVERT', 'LongWord.SetUInt( $005A0049); 795
16653: PATPAINT', 'LongWord.SetUInt( $00FB0A09); 794
16654: PCF_16BITMODE', 'LongWord.SetUInt( $0200); 236
16655: PCF_DTRDSR', 'LongWord.SetUInt( $0001); 227
16656: PCF_INTTIMEOUTS', 'LongWord.SetUInt( $0080); 234
16657: PCF_PARITY_CHECK', 'LongWord.SetUInt( $0008); 230
16658: PCF_RLSD', 'LongWord.SetUInt( $0004); 229
16659: PCF_RTSCTS', 'LongWord.SetUInt( $0002); 228
16660: PCF_SETXCHAR', 'LongWord.SetUInt( $0020); 232
16661: PCF_SPECIALCHARS', 'LongWord.SetUInt( $0100); 235
16662: PCF_TOTALTIMEOUTS', 'LongWord.SetUInt( $0040); 233
16663: PCF_XONXOFF', 'LongWord.SetUInt( $0010); 231
16664: PDF_ENTRY_CLOSED', 'LongInt.SetInt( 0); 3173
16665: PDF_ENTRY_CLOSED', 'LongInt.SetInt( 0); 3362
16666: PDF_ENTRY_OPENED', 'LongInt.SetInt( 1); 3174
16667: PDF_ENTRY_OPENED', 'LongInt.SetInt( 1); 3363
16668: PDF_FREE_ENTRY', 'String.SetString( 'f; 3171
16669: PDF_FREE_ENTRY', 'String.SetString( 'f; 3360
16670: PDF_IN_USE_ENTRY', 'String.SetString( 'n; 3170
16671: PDF_IN_USE_ENTRY', 'String.SetString( 'n; 3359
16672: PDF_MAX_CHARSSPACE', 'LongInt.SetInt( 300); 3181
16673: PDF_MAX_CHARSSPACE', 'LongInt.SetInt( 300); 3370
16674: PDF_MAX_FONTSIZE', 'LongInt.SetInt( 2000); 3182
16675: PDF_MAX_FONTSIZE', 'LongInt.SetInt( 2000); 3371
16676: PDF_MAX_GENERATION_NUM', 'LongInt.SetInt( 65535); 3172
16677: PDF_MAX_GENERATION_NUM', 'LongInt.SetInt( 65535); 3361
16678: PDF_MAX_HORIZONTALSCALING', 'LongInt.SetInt( 300); 3178
16679: PDF_MAX_HORIZONTALSCALING', 'LongInt.SetInt( 300); 3367
16680: PDF_MAX_LEADING', 'LongInt.SetInt( 300); 3184
16681: PDF_MAX_LEADING', 'LongInt.SetInt( 300); 3373
16682: PDF_MAX_WORDSSPACE', 'LongInt.SetInt( 300); 3179
16683: PDF_MAX_WORDSSPACE', 'LongInt.SetInt( 300); 3368

```

```
16684: PDF_MAX_ZOOMSIZE', 'LongInt.SetInt( 10); 3183
16685: PDF_MAX_ZOOMSIZE', 'LongInt.SetInt( 10); 3372
16686: PDF_MIN_CHARSIZE', 'LongInt.SetInt( - 30); 3180
16687: PDF_MIN_CHARSIZE', 'LongInt.SetInt( - 30); 3369
16688: PDF_MIN_HORIZONTALSCALING', 'LongInt.SetInt( 10); 3177
16689: PDF_MIN_HORIZONTALSCALING', 'LongInt.SetInt( 10); 3366
16690: PGE_FLAG', 'LongWord.SetUInt( $00002000); 1437
16691: Pi', 'Extended.setExtended( 3.14159265358979323846); 3198
16692: Pi', 'Float.SetString( 3.1415926535897932384626433832795); 1286
16693: Pi', 'Float.SetString( 3.1415926535897932384626433832795); 1354
16694: PiDiv2', 'Extended.setExtended( 1.57079632679489661923); 3205
16695: PiExt', 'Extended.setExtended( 3.1415926535897932384626433832795); 1324
16696: PiExt', 'Extended.setExtended( 3.1415926535897932384626433832795); 1392
16697: PiExt', 'Extended.setExtended( 3.1415926535897932384626433832795); 1712
16698: PiJ', 'Extended.setExtended( 3.1415926535897932384626433832795); 1674
16699: PILARGE', 'String.SetString(PILARGE); 466
16700: PiOn2', 'Extended.setExtended( 1.5707963267948966192313216916398); 1675
16701: PiOn2', 'Float.SetString( 1.5707963267948966192313216916398); 1287
16702: PiOn2', 'Float.SetString( 1.5707963267948966192313216916398); 1355
16703: PiOn3', 'Extended.setExtended( 1.0471975511965977461542144610932); 1676
16704: PiOn3', 'Float.SetString( 1.0471975511965977461542144610932); 1288
16705: PiOn3', 'Float.SetString( 1.0471975511965977461542144610932); 1356
16706: PiOn4', 'Extended.setExtended( 0.78539816339744830961566084581988); 1677
16707: PiOn4', 'Float.SetString( 0.78539816339744830961566084581988); 1289
16708: PiOn4', 'Float.SetString( 0.78539816339744830961566084581988); 1357
16709: PO_DELETE', 'LongWord.SetUInt( $0013); 2903
16710: PO_PORTCHANGE', 'LongWord.SetUInt( $0020); 2905
16711: PO_REN_PORT', 'LongWord.SetUInt( $0034); 2906
16712: PO_RENAME', 'LongWord.SetUInt( $0014); 2904
16713: POWER_1', 'LongWord.SetUInt( $000000FF); 1115
16714: POWER_2', 'LongWord.SetUInt( $0000FFFF); 1116
16715: POWER_3', 'LongWord.SetUInt( $00FFFFFF); 1117
16716: POWER_4', 'LongWord.SetUInt( $FFFFFFF); 1118
16717: PROCESS_CREATE_PROCESS', 'LongWord.SetUInt( $0080); 207
16718: PROCESS_CREATE_THREAD', 'LongWord.SetUInt( $0002); 202
16719: PROCESS_DUP_HANDLE', 'LongWord.SetUInt( $0040); 206
16720: PROCESS_QUERY_INFORMATION', 'LongWord.SetUInt( $0400); 210
16721: PROCESS_SET_INFORMATION', 'LongWord.SetUInt( $0200); 209
16722: PROCESS_SET_QUOTA', 'LongWord.SetUInt( $0100); 208
16723: PROCESS_TERMINATE', 'LongWord.SetUInt( $0001); 201
16724: PROCESS_VM_OPERATION', 'LongWord.SetUInt( $0008); 203
16725: PROCESS_VM_READ', 'LongWord.SetUInt( $0010); 204
16726: PROCESS_VM_WRITE', 'LongWord.SetUInt( $0020); 205
16727: PROOF_QUALITY', 'LongInt.SetInt( 2); 162
16728: PSE_FLAG', 'LongWord.SetUInt( $00000008); 1427
16729: PSE36_FLAG', 'LongWord.SetUInt( $00020000); 1441
16730: PSINC', 'String.SetString( 'PS Includes (*.inc)|*.INC; 934
16731: PSINC', 'String.SetString( 'PS Includes (*.inc)|*.INC; 958
16732: PSMODEL', 'String.SetString( 'PS Modelfiles (*.uc)|*.UC; 932
16733: PSMODEL', 'String.SetString( 'PS Modelfiles (*.uc)|*.UC; 956
16734: PSPASCAL', 'String.SetString( 'PS Pascalfiles (*.pas)|*.PAS; 933
16735: PSPASCAL', 'String.SetString( 'PS Pascalfiles (*.pas)|*.PAS; 957 *)
16736: PST_BITMAP_FRAME', 'LongInt.SetInt( 7); 2405
16737: PST_CLEAR_BACKGND', 'LongInt.SetInt( 3); 2401
16738: PST_CLEAR_BUFFER', 'LongInt.SetInt( 2); 2400
16739: PST_CONTROL_FRAME', 'LongInt.SetInt( 6); 2404
16740: PST_CUSTOM', 'LongInt.SetInt( 1); 2399
16741: PST_DRAW_BITMAP', 'LongInt.SetInt( 4); 2402
16742: PST_DRAW_LAYERS', 'LongInt.SetInt( 5); 2403
16743: PST_FAX', 'LongWord.SetUInt( $00000021); 221
16744: PST_LAT', 'LongWord.SetUInt( $00000101); 224
16745: PST_MODEM', 'LongWord.SetUInt( $00000006); 220
16746: PST_NETWORK_BRIDGE', 'LongWord.SetUInt( $00000100); 223
16747: PST_PARALLELPORT', 'LongWord.SetUInt( $00000002); 216
16748: PST_RS232', 'LongWord.SetUInt( $00000001); 215
16749: PST_RS422', 'LongWord.SetUInt( $00000003); 217
16750: PST_RS423', 'LongWord.SetUInt( $00000004); 218
16751: PST_RS449', 'LongWord.SetUInt( $00000005); 219
16752: PST_SCANNER', 'LongWord.SetUInt( $00000022); 222
16753: PST_TCPIP_TELNET', 'LongWord.SetUInt( $00000102); 225
16754: PST_UNSPECIFIED', 'LongWord.SetUInt( $00000000); 214
16755: PST_X25', 'LongWord.SetUInt( $00000103); 226
16756: PSTEXT', 'String.SetString( 'PS Scriptfiles (*.txt)|*.TXT; 931
16757: PSTEXT', 'String.SetString( 'PS Scriptfiles (*.txt)|*.TXT; 955 *)
16758: RadCor', 'Double.setExtended( 57.29577951308232); 2001
16759: radcor', 'Extended.setExtended( 57.29577951308232); 2974
16760: RadPerCycle', 'Extended.setExtended( 6.283185307179586476925286766559); 2580
16761: RadPerDeg', 'Extended.setExtended( 0.017453292519943295769236907684886); 2581
16762: RadPerGrad', 'Extended.setExtended( 0.015707963267948966192313216916398); 2582
16763: RatioDegToGrad', 'Extended.setExtended( 200.0 / 180.0); 1327
16764: RatioDegToGrad', 'Extended.setExtended( 200.0 / 180.0); 1395
16765: RatioDegToGrad', 'Extended.setExtended( 200.0 / 180.0); 1715
16766: RatioDegToRad', 'Extended.setExtended( PiExt / 180.0); 1325
16767: RatioDegToRad', 'Extended.setExtended( PiExt / 180.0); 1393
16768: RatioDegToRad', 'Extended.setExtended( PiExt / 180.0); 1713
16769: RatioGradToDeg', 'Extended.setExtended( 180.0 / 200.0); 1328
16770: RatioGradToDeg', 'Extended.setExtended( 180.0 / 200.0); 1396
16771: RatioGradToDeg', 'Extended.setExtended( 180.0 / 200.0); 1716
16772: RatioGradToRad', 'Extended.setExtended( PiExt / 200.0); 1326
```

```

16773: RatioGradToRad', 'Extended.setExtended( PiExt / 200.0); 1394
16774: RatioGradToRad', 'Extended.setExtended( PiExt / 200.0); 1714
16775: rc3_Cursor', 'LongInt.SetInt(2); 816
16776: rc3_Icon', 'LongInt.SetInt(1); 815
16777: rc3_StockIcon', 'LongInt.SetInt(0); 814
16778: rcs_id', 'String.SetString( '#(@)$Id: GUITesting.pas,v 1.36 2006/07/19 02:45:54 judc Exp $; 987
16779: RegExprLinePairedSeparator', 'RegExprString.SetString( #d#$a); 2056
16780: RegExprLineSeparators', 'RegExprString.SetString( #d#$a + #b#$c#$2028#$2029#$85); 2055
16781: RegExprModifierG', 'boolean.BoolToStr( False); 2050
16782: RegExprModifierI', 'boolean.BoolToStr( False); 2047
16783: RegExprModifierM', 'boolean.BoolToStr( False); 2051
16784: RegExprModifierR', 'boolean.BoolToStr( False); 2048
16785: RegExprModifierS', 'boolean.BoolToStr( False); 2049
16786: RegExprModifierX', 'boolean.BoolToStr( False); } 2052
16787: RegExprSpaceChars', 'RegExprString.SetString( ' + #9#$A#$D#$C); 2053
16788: RegExprWordChars',
    'RegExprString.SetString( '0123456789'+ 'abcdefghijklmnopqrstuvwxyz'+ 'ABCDEFGHIJKLMNOPQRSTUVWXYZ_'; 2054
16789: RepeatPause', 'LongInt.SetInt( 100); 890
16790: ResetOption', 'Integer.SetInt( 1 shl ord( grReset)); 1731
16791: RSIdMessageCannotLoad', 'String.SetString( 'Cannot load message from file %s; 1129
16792: RT_RCDATA', 'LongInt.SetInt( PChar ( 10 )); 2125
16793: RTLVersion', 'Extended.setExtended( 18.00); //3.7 301
16794: RTLVersion', 'Extended.setExtended( 18.00); 2062
16795: S_IN', 'LongInt.SetInt( 0); 3408
16796: S_OUT', 'LongInt.SetInt( 1); 3409
16797: SAnd', 'String.SetString( ' and ; 1921
16798: SB_BOTH', 'LongInt.SetInt( 3); 370
16799: SB_BOTTOM', 'LongInt.SetInt( 7); 383
16800: SB_CTL', 'LongInt.SetInt( 2); 369
16801: SB_ENDSCROLL', 'LongInt.SetInt( 8); 385
16802: SB_HORZ', 'LongInt.SetInt( 0); 367
16803: SB_LEFT', 'LongInt.SetInt( 6); 382
16804: SB_LINEDOWN', 'LongInt.SetInt( 1); 373
16805: SB_LINELEFT', 'LongInt.SetInt( 0); 372
16806: SB_LINERIGHT', 'LongInt.SetInt( 1); 374
16807: SB_LINEUP', 'LongInt.SetInt( 0); 371
16808: SB_PAGEDOWN', 'LongInt.SetInt( 3); 377
16809: SB_PAGELEFT', 'LongInt.SetInt( 2); 376
16810: SB_PAGERIGHT', 'LongInt.SetInt( 3); 378
16811: SB_PAGEUP', 'LongInt.SetInt( 2); 375
16812: SB_RIGHT', 'LongInt.SetInt( 7); 384
16813: SB_THUMBPOSITION', 'LongInt.SetInt( 4); 379
16814: SB_THUMBTRACK', 'LongInt.SetInt( 5); 380
16815: SB_TOP', 'LongInt.SetInt( 6); 381
16816: SB_VERT', 'LongInt.SetInt( 1); 368
16817: sBlockSize', 'Char.SetString( 'blksize' #0); 1227
16818: SC_ARRANGE', 'LongInt.SetInt( 61712); 60
16819: SC_CLOSE', 'LongInt.SetInt( 61536); 55
16820: SC_CONTEXTHELP', 'LongInt.SetInt( 61824); 67
16821: SC_DEFAULT', 'LongInt.SetInt( 61792); 65
16822: SC_DRAGMOVE', 'LongWord.SetUInt( $F012); 2752
16823: SC_HOTKEY', 'LongInt.SetInt( 61776); 64
16824: SC_HSCROLL', 'LongInt.SetInt( 61568); 57
16825: SC_ICON', 'SetString( SC_MINIMIZE); 69
16826: SC_KEYMENU', 'LongInt.SetInt( 61696); 59
16827: SC_MAXIMIZE', 'LongInt.SetInt( 61488); 52
16828: SC_MINIMIZE', 'LongInt.SetInt( 61472); 51
16829: SC_MONITORPOWER', 'LongInt.SetInt( 61808); 66
16830: SC_MOUSEMENU', 'LongInt.SetInt( 61584); 58
16831: SC_MOVE', 'LongInt.SetInt( 61456); 50
16832: SC_NEXTWINDOW', 'LongInt.SetInt( 61504); 53
16833: SC_PREVWINDOW', 'LongInt.SetInt( 61520); 54
16834: SC_RESTORE', 'LongInt.SetInt( 61728); 61
16835: SC_SCREENSAVE', 'LongInt.SetInt( 61760); 63
16836: SC_SEPARATOR', 'LongInt.SetInt( 61455); 68
16837: SC_SIZE', 'LongInt.SetInt( 61440); 49
16838: SC_TASKLIST', 'LongInt.SetInt( 61744); 62
16839: SC_VSCROLL', 'LongInt.SetInt( 61552); 56
16840: SC_ZOOM', 'SetString( SC_MAXIMIZE); 70
16841: scAlt', 'LongWord.SetUInt( $8000); 503
16842: scAlt', 'LongWord.SetUInt( $8000); 832
16843: sCannotPause', 'String.SetString( 'Can only pause when recording; 3163
16844: sCannotPlay', 'String.SetString( 'Cannot playback macro; already playing or recording; 3162
16845: sCannotRecord', 'String.SetString( 'Cannot record macro; already recording or playing; 3161
16846: sCannotResume', 'String.SetString( 'Can only resume when paused; 3164
16847: scCtrl', 'LongWord.SetUInt( $4000); 502
16848: scCtrl', 'LongWord.SetUInt( $4000); 831
16849: scNone', 'LongInt.SetInt( 0); 504
16850: scNone', 'LongInt.SetInt( 0); 833
16851: sContentDisposition', 'String.SetString( 'Content-Disposition: form-data; name="%s"; 1133
16852: sContentTransferEncoding', 'String.SetString( 'Content-Transfer-Encoding: binary; 1136
16853: sContentTypeFormData', 'String.SetString( 'multipart/form-data; boundary=; 1130
16854: sContentTypeOctetStream', 'String.SetString( 'application/octet-stream; 1131
16855: sContentTypePlaceholder', 'String.SetString( 'Content-Type: %s; 1135
16856: scShift', 'LongWord.SetUInt( $2000); 501
16857: scShift', 'LongWord.SetUInt( $2000); 830
16858: sDateFormat', 'String.SetString( "%s", dd "%s" yyyy hh:nn:ss; 989
16859: sdfInterbase', 'String.SetString( 'CAST(''mm''/''dd''/''yyyy'' AS DATE); 1595
16860: sdfMSSQL', 'String.SetString( 'CONVERT(datetime, ''mm''/''dd''/''yyyy'', 103); 1596

```

```

16861: sdfOracle', 'String.SetString( "TO_DATE(''dd/mm/yyyy'', 'DD/MM/YYYY"; 1594
16862: sdfStandard16', 'String.SetString( '""mm'/'dd'/'yyyy'""; 1592
16863: sdfStandard32', 'String.SetString( '""dd/mm/yyyy'""; 1593
16864: SE_ERR_ACCESSDENIED', 'LongInt.SetInt( 5); 2909
16865: SE_ERR_ASSOCINCOMPLETE', 'LongInt.SetInt( 27); 2913
16866: SE_ERR_DDEBUSY', 'LongInt.SetInt( 30); 2916
16867: SE_ERR_DDEFAIL', 'LongInt.SetInt( 29); 2915
16868: SE_ERR_DDETIMEOUT', 'LongInt.SetInt( 28); 2914
16869: SE_ERR_DLLNOTFOUND', 'LongInt.SetInt( 32); 2911
16870: SE_ERR_FNF', 'LongInt.SetInt( 2); 2907
16871: SE_ERR_NOASSOC', 'LongInt.SetInt( 31); 2917
16872: SE_ERR_OOM', 'LongInt.SetInt( 8); 2910
16873: SE_ERR_PNF', 'LongInt.SetInt( 3); 2908
16874: SE_ERR_SHARE', 'LongInt.SetInt( 26); 2912
16875: SecondsInDay', 'LongInt.SetInt( 86400); 3002
16876: SecondsInHour', 'LongInt.SetInt( 3600); 3003
16877: SecondsInMinute', 'LongInt.SetInt( 60); 3004
16878: SecondsPerMinute', 'Extended.SetExtended( 60); 2182
16879: SecOnly', 'String.SetString( 's; 3023
16880: SecOnlyU', 'String.SetString( 'S; 3026
16881: SecsPerDay', 'LongInt.SetInt(MinsPerDay * SecsPerMin); 37
16882: SecsPerMin', 'LongInt.SetInt( 60); 34
16883: SEE_MASK_ASYNCOK', 'LongWord.SetUInt( $00100000); 2931
16884: SEE_MASK_CLASSKEY', 'LongWord.SetUInt( $00000003); 2919
16885: SEE_MASK_CLASSNAME', 'LongWord.SetUInt( $00000001); 2918
16886: SEE_MASK_CONNECTNETDRV', 'LongWord.SetUInt( $00000080); 2925
16887: SEE_MASK_DOENVSUBST', 'LongWord.SetUInt( $00000200); 2927
16888: SEE_MASK_FLAG_DDEWAIT', 'LongWord.SetUInt( $00000100); 2926
16889: SEE_MASK_FLAG_NO_UI', 'LongWord.SetUInt( $00000400); 2928
16890: SEE_MASK_HOTKEY', 'LongWord.SetUInt( $00000020); 2923
16891: SEE_MASK_ICON', 'LongWord.SetUInt( $00000010); 2922
16892: SEE_MASK_IDLIST', 'LongWord.SetUInt( $00000004); 2920
16893: SEE_MASK_INVOKEIDLIST', 'LongWord.SetUInt( $0000000c); 2921
16894: SEE_MASK_NO_CONSOLE', 'LongWord.SetUInt( $00008000); 2930
16895: SEE_MASK_NOCLOSEPROCESS', 'LongWord.SetUInt( $00000040); 2924
16896: SEE_MASK_UNICODE', 'LongWord.SetUInt( $00010000); 2929
16897: SEMAPHORE_MODIFY_STATE', 'LongWord.SetUInt( $0002); 200
16898: SEP_FLAG', 'LongWord.SetUInt( $00000800); 1435
16899: SERVICE_CONFIG_DESCRIPTION', 'LongInt.SetInt( 1); 2566
16900: SERVICE_CONFIG_DESCRIPTION', 'LongInt.SetInt( 1); 3440
16901: SERVICE_CONFIG_FAILURE_ACTIONS', 'LongInt.SetInt( 2); 2567
16902: SERVICE_CONFIG_FAILURE_ACTIONS', 'LongInt.SetInt( 2); 3441
16903: SERVICE_RUNS_IN_SYSTEM_PROCESS', 'LongWord.SetUInt( $00000001); 3439
16904: sFileNamePlaceholder', 'String.SetString( ' filename="%s"; 1134
16905: SHACF_AUTOAPPEND_FORCE_OFF', 'LongWord.SetUInt( DWORD ( $80000000 )); 1887
16906: SHACF_AUTOAPPEND_FORCE_ON', 'LongWord.SetUInt( $40000000); 1886
16907: SHACF_AUTOSUGGEST_FORCE_OFF', 'LongWord.SetUInt( $20000000); 1885
16908: SHACF_AUTOSUGGEST_FORCE_ON', 'LongWord.SetUInt( $10000000); 1884
16909: SHACF_DEFAULT', 'LongWord.SetUInt( $00000000); 1878
16910: SHACF_FILESYS_ONLY', 'LongWord.SetUInt( $00000010); 1883
16911: SHACF_FILESYSTEM', 'LongWord.SetUInt( $00000001); 1879
16912: SHACF_URLHISTORY', 'LongWord.SetUInt( $00000002); 1880
16913: SHACF_URLMRU', 'LongWord.SetUInt( $00000004); 1881
16914: SHACF_USETAB', 'LongWord.SetUInt( $00000008); 1882
16915: SHCONTF_INIT_ON_FIRST_NEXT', 'LongWord.SetUInt( $0100); 2662
16916: SHCONTF_NETPRINTERSRCH', 'LongWord.SetUInt( $0200); 2663
16917: SHCONTF_SHAREABLE', 'LongWord.SetUInt( $0400); 2664
16918: SHCONTF_STORAGE', 'LongWord.SetUInt( $0800); 2665
16919: shell32', 'String.SetString( 'shell32.dll; 2972
16920: shell32linux', 'String.SetString( 'libshell32.borland.so; 2973
16921: SHGFI_ATTRIBUTES', 'LongWord.SetUInt( $000000800); 2956
16922: SHGFI_DISPLAYNAME', 'LongWord.SetUInt( $000000200); 2954
16923: SHGFI_EXETYPE', 'LongWord.SetUInt( $000002000); 2958
16924: SHGFI_ICON', 'LongWord.SetUInt( $000000100); 2953
16925: SHGFI_ICONLOCATION', 'LongWord.SetUInt( $000001000); 2957
16926: SHGFI_LARGEICON', 'LongWord.SetUInt( $000000000); 2962
16927: SHGFI_LINKOVERLAY', 'LongWord.SetUInt( $000008000); 2960
16928: SHGFI_OPENICON', 'LongWord.SetUInt( $000000002); 2964
16929: SHGFI_PIDL', 'LongWord.SetUInt( $000000008); 2966
16930: SHGFI_SELECTED', 'LongWord.SetUInt( $000010000); 2961
16931: SHGFI_SHELLICONSIZE', 'LongWord.SetUInt( $000000004); 2965
16932: SHGFI_SMALLICON', 'LongWord.SetUInt( $000000001); 2963
16933: SHGFI_SYSICONINDEX', 'LongWord.SetUInt( $000004000); 2959
16934: SHGFI_TYPENAME', 'LongWord.SetUInt( $000000400); 2955
16935: SHGFI_USEFILEATTRIBUTES', 'LongWord.SetUInt( $000000010); 2967
16936: SHGNLI_UNIQUE', 'LongWord.SetUInt( $000000004); 2971
16937: SHGNLI_PIDL', 'LongWord.SetUInt( $000000001); 2969
16938: SHGNLI_PREFIXNAME', 'LongWord.SetUInt( $000000002); 2970
16939: SHIFTJIS_CHARSET', 'LongWord.SetUInt( $80); 172
16940: SHOW_FULLSCREEN', 'LongInt.SetInt( 3); 403
16941: SHOW_ICONWINDOW', 'LongInt.SetInt( 2); 402
16942: SHOW_OPENNOACTIVATE', 'LongInt.SetInt( 4); 404
16943: SHOW_OPENWINDOW', 'LongInt.SetInt( 1); 401
16944: SHREGDEL_BOTH', 'LongWord.SetUInt( $00000011); 1834
16945: SHREGDEL_DEFAULT', 'LongWord.SetUInt( $000000000); 1831
16946: SHREGDEL_HKCU', 'LongWord.SetUInt( $000000001); 1832
16947: SHREGDEL_HKLM', 'LongWord.SetUInt( $000000010); 1833
16948: SHREGENUM_BOTH', 'LongWord.SetUInt( $00000011); 1838
16949: SHREGENUM_DEFAULT', 'LongWord.SetUInt( $000000000); 1835

```



```

16950: SHREGENUM_HKCU', 'LongWord.SetUInt( $00000001); 1836
16951: SHREGENUM_HKLM', 'LongWord.SetUInt( $00000010); 1837
16952: SHREGSET_FORCE_HKCU', 'LongWord.SetUInt( $00000002); 1840
16953: SHREGSET_FORCE_HKLM', 'LongWord.SetUInt( $00000008); 1842
16954: SHREGSET_HKCU', 'LongWord.SetUInt( $00000001); 1839
16955: SHREGSET_HKLM', 'LongWord.SetUInt( $00000004); 1841
16956: SID_IFolderFilter', 'String.SetString( '{9CC22886-DC8E-11d2-B1D0-00C04F8EEB3E}; 2694
16957: SID_IFolderFilterSite', 'String.SetString( '{C0A651F5-B48B-11d2-B5ED-006097C686F6}; 2692
16958: SID_IQueryAssociations', 'String.SetString( '{c46ca590-3c3f-11d2-bee6-0000f805ca57}; 1877
16959: SignBitMask', 'LongWord.SetUInt( $80000000); 3324
16960: SingleCompareDelta', 'Extended.setExtended( 1.0E-34); 2221
16961: SingleCompareEpsilon', 'Extended.setExtended( 1.0E-5); 2225
16962: sivFirst', 'LongInt.SetInt( 1); 2558
16963: sivFourth', 'LongInt.SetInt( 4); 2561
16964: sivLast', 'LongInt.SetInt( - 1); 2562
16965: sivSecond', 'LongInt.SetInt( 2); 2559
16966: sivThird', 'LongInt.SetInt( 3); 2560
16967: SizeOfFraction', 'LongInt.SetInt( 32); 985
16968: slfAutoMinSum', 'LongInt.SetInt( - 1); 1723
16969: slfAutoMostRuns', 'LongInt.SetInt( - 2); 1724
16970: slfAutoTryAll', 'LongInt.SetInt( - 3); 1725
16971: slfAverage', 'LongInt.SetInt( 3); 1729
16972: slfNone', 'LongInt.SetInt( 0); 1726
16973: slfPaeth', 'LongInt.SetInt( 4); 1730
16974: slfSub', 'LongInt.SetInt( 1); 1727
16975: slfUp', 'LongInt.SetInt( 2); 1728
16976: sLineBreak', '.SetString( EOL); 1114
16977: sLineBreak', 'EOL.SetString(#13+#10); 1078
16978: SLineBreak', 'String.SetString( #13#10#10); 3159
16979: SM_CXVIRTUALSCREEN', 'LongInt.SetInt( 78); 2756
16980: SM_CYVIRTUALSCREEN', 'LongInt.SetInt( 79); 2757
16981: SM_MOUSEWHEELPRESENT', 'LongInt.SetInt( 75); 1636
16982: SM_XVIRTUALSCREEN', 'LongInt.SetInt( 76); 2754
16983: SM_YVIRTUALSCREEN', 'LongInt.SetInt( 77); 2755
16984: SND_ALIAS', 'LongWord.SetUInt($00010000); 110
16985: SND_ALIAS_ID', 'LongWord.SetUInt($00110000); 111
16986: SND_APPLICATION', 'LongWord.SetUInt($0080); 115
16987: SND_ASYNC', 'LongWord.SetUInt($0001); 104
16988: SND_FILENAME', 'LongWord.SetUInt($00020000); 112
16989: SND_LOOP', 'LongWord.SetUInt($0008); 107
16990: SND_MEMORY', 'LongWord.SetUInt($0004); 106
16991: SND_NODEFAULT', 'LongWord.SetUInt($0002); 105
16992: SND_NOSTOP', 'LongWord.SetUInt($0010); 108
16993: SND_NOWAIT', 'LongWord.SetUInt($00002000); 109
16994: SND_PURGE', 'LongWord.SetUInt($0040); 114
16995: SND_RESOURCE', 'LongWord.SetUInt($00040004); 113
16996: SND_SYNC', 'LongWord.SetUInt($0000); //default 103
16997: SOCK_DGRAM', 'LongInt.SetInt( 2); 3402
16998: SOCK_RAW', 'LongInt.SetInt( 3); 3403
16999: SOCK_RDM', 'LongInt.SetInt( 4); 3404
17000: SOCK_SEQPACKET', 'LongInt.SetInt( 5); 3405
17001: SOCK_STREAM', 'LongInt.SetInt( 1); 3401
17002: soFromBeginning', 'LongInt.SetInt( 0); 820
17003: soFromCurrent', 'LongInt.SetInt( 1); 821
17004: soFromEnd', 'LongInt.SetInt( 2); 822
17005: SOL_SOCKET', 'LongWord.SetUInt( $ffff); 3410
17006: SOrderBy', 'String.SetString( ' order by ; 1922
17007: SP_BAUD', 'LongWord.SetUInt( $0002); 238
17008: SP_DATABITS', 'LongWord.SetUInt( $0004); 239
17009: SP_HANDSHAKING', 'LongWord.SetUInt( $0010); 241
17010: SP_PARITY', 'LongWord.SetUInt( $0001); 237
17011: SP_PARITY_CHECK', 'LongWord.SetUInt( $0020); 242
17012: SP_RLSD', 'LongWord.SetUInt( $0040); 243
17013: SP_SERIALCOMM', 'LongWord.SetUInt( $00000001); 213
17014: SP_STOPBITS', 'LongWord.SetUInt( $0008); 240
17015: SpaceSize', 'LongInt.SetInt( 5); 891
17016: SParam', 'String.SetString( '?; 1923
17017: SPI_GETFLATMENU', 'LongWord.SetUInt( $1022); 2753
17018: SPI_GETWHEELSCROLLLINES', 'LongInt.SetInt( 104); 1638
17019: SPI_SETWHEELSCROLLLINES', 'LongInt.SetInt( 105); 1639
17020: SQLSections', 'LongInt.Value.ts32 := ord(stSelect) or ord(stFrom) or ord(stWhere) or ord(stGroupBy) or
ord(stHaving) or ord(stUnion) or ord(stPlan) or ord(stOrderBy) or ord(stForUpdate); 888
17021: SqrLimit', 'Extended.setExtended( 1E2466); 2732
17022: Sqrt10', 'Extended.SetExtended( 3.1622776601683793319988935444327); 1681
17023: Sqrt10', 'Float.SetString( 3.1622776601683793319988935444327); 1293
17024: Sqrt10', 'Float.SetString( 3.1622776601683793319988935444327); 1361
17025: Sqrt2', 'Extended.setExtended( 1.41421356237309504880); 3211
17026: Sqrt2', 'Extended.SetExtended( 1.4142135623730950488016887242097); 1678
17027: Sqrt2', 'Float.SetString( 1.4142135623730950488016887242097); 1290
17028: Sqrt2', 'Float.SetString( 1.4142135623730950488016887242097); 1358
17029: Sqrt2Div2', 'Extended.setExtended( 0.70710678118654752440); 3212
17030: Sqrt2Pi', 'Extended.SetExtended( 2.506628274631000502415765284811); 1683
17031: Sqrt2Pi', 'Extended.setExtended( 2.50662827463100050242); 3207
17032: Sqrt2Pi', 'Float.SetString( 2.506628274631000502415765284811); 1295
17033: Sqrt2Pi', 'Float.SetString( 2.506628274631000502415765284811); 1363
17034: Sqrt3', 'Extended.SetExtended( 1.7320508075688772935274463415059); 1679
17035: Sqrt3', 'Float.SetString( 1.7320508075688772935274463415059); 1291
17036: Sqrt3', 'Float.SetString( 1.7320508075688772935274463415059); 1359
17037: Sqrt5', 'Extended.SetExtended( 2.2360679774997896964091736687313); 1680

```

```

17038: Sqrt5', 'Float.SetString( 2.2360679774997896964091736687313); 1292
17039: Sqrt5', 'Float.SetString( 2.2360679774997896964091736687313); 1360
17040: SqrtPi', 'Extended.SetExtended( 1.7724538509055160272981674833411); 1682
17041: SqrtPi', 'Extended.setExtended( 1.77245385090551602730); 3206
17042: SqrtPi', 'Float.SetString( 1.7724538509055160272981674833411); 1294
17043: SqrtPi', 'Float.SetString( 1.7724538509055160272981674833411); 1362
17044: SquareMetersPerAcre', 'Extended.SetExtended( SquareMetersPerSquareYard * 4840); 1952
17045: SRCAND', 'LongWord.SetUInt( $008800C6); 786
17046: SRCAND', 'LongWord.SetUInt($008800C6); 44
17047: SRCCOPY', 'LongWord.SetUInt( $00CC0020); 784
17048: SRCCOPY', 'LongWord.SetUInt($00CC0020); 42
17049: SRCERASE', 'LongWord.SetUInt( $00440328); 788
17050: SRCERASE', 'LongWord.SetUInt($00440328); 46
17051: SRCINVERT', 'LongWord.SetUInt( $00660046); 787
17052: SRCINVERT', 'LongWord.SetUInt($00660046); 45
17053: SRCPAINT', 'LongWord.SetUInt( $00EE0086); 785
17054: SRCPAINT', 'LongWord.SetUInt($00EE0086); 43
17055: SSelect', 'String.SetString( 'select; 1916
17056: SSelectSpaces', 'String.SetString( ' select ; 1919
17057: SSelectStar', 'String.SetString( ' select * ; 1917
17058: SSelectStarFrom', 'String.SetString( ' select * from ; 1918
17059: St_WhiteSpace', 'String.SetString( #8#9#10#13); 2045
17060: StackMax', 'LongInt.SetInt( 1000); 2727
17061: STANDARD_RIGHTS_REQUIRED', 'LongWord.SetUInt( $000F0000); 197
17062: StandardColCount', 'LongInt.SetInt( 40); 1607
17063: StandardExponent', 'LongInt.SetInt( 65537); 2140
17064: StAnyFile', 'String.SetString( '.*; 2037
17065: STATUS_TIMEOUT', 'LongWord.SetUInt($00000102); 180
17066: STATUS_WAIT_0', 'LongWord.SetUInt($00000000); 182
17067: StCRLF', 'String.SetString( #13#10); 1995
17068: StdDate', 'Extended.setExtended( 2451545.0); 1929
17069: StdDate', 'Extended.setExtended( 2451545.0); 2975
17070: StDefaultComment', 'String.SetString( '; 2042
17071: StDefaultDelim', 'String.SetString( ',; 2040
17072: StDefaultFixedSep', 'String.SetString( ',; 2043
17073: StDefaultLineTerm', 'String.SetString( #13#10); 2044
17074: StDefaultQuote', 'String.SetString( "; 2041
17075: StDefBufferSize', 'LongInt.SetInt( 65536); 1992
17076: StDefHighLevel', 'LongInt.SetInt( 0); 1993
17077: StDelta', 'Extended.setExtended( 0.00001); 1989
17078: StDosAnyFile', 'String.SetString( '.*; 2035
17079: StDosPathDelim', 'String.SetString( \; 2031
17080: StDosPathSep', 'String.SetString( '; 2033
17081: StDriveDelim', 'String.SetString( ':; 2030
17082: StDriverService', 'LongInt.Value.ts32:=ord(stKernelDriver)or ord(stFileSystemDriver)or
ord(stRecognizerDriver);2563
17083: StEpsilon', 'Extended.setExtended( 0.00001); 1990
17084: STG_S_MONITORING', 'LongWord.SetUInt( $00030203); 2137
17085: STGTY_LOCKBYTES', 'LongInt.SetInt( 3); 2128
17086: STGTY_PROPERTY', 'LongInt.SetInt( 4); 2129
17087: STGTY_STORAGE', 'LongInt.SetInt( 1); 2126
17088: STGTY_STREAM', 'LongInt.SetInt( 2); 2127
17089: StHexDigitsW', 'WideString.SetString( '0123456789ABCDEF; 2985
17090: STIF_DEFAULT', 'LongWord.SetUInt( $00000000); 1767
17091: STIF_SUPPORT_HEX', 'LongWord.SetUInt( $00000001); 1768
17092: StLogFileFooter', 'String.SetString( '; 1996
17093: StMaxBlockSize', 'LongInt.SetInt( MaxLongInt); 2981
17094: StMaxFileLen', 'LongInt.SetInt( 260); 2982
17095: StMaxIterations', 'Integer.SetInt( 100); 1991
17096: StMaxLogSize', 'LongInt.SetInt( 16000000); 1994
17097: StParentDir', 'String.SetString( '..; 2039
17098: StPathDelim', 'String.SetString( \; 2028
17099: StPathSep', 'String.SetString( '; 2029
17100: STREAM_SEEK_CUR', 'LongInt.SetInt( 1); 2131
17101: STREAM_SEEK_END', 'LongInt.SetInt( 2); 2132
17102: STREAM_SEEK_SET', 'LongInt.SetInt( 0); 2130
17103: StRLEMaxCount', 'LongInt.SetInt( 127); 2983
17104: StRLERunMode', 'LongWord.SetUInt( $80); 2984
17105: strole32', 'String.SetString( 'ole32.dll; 3328
17106: StThisDir', 'String.SetString( '.; 2038
17107: StUnixAnyFile', 'String.SetString( '.*; 2036
17108: StUnixPathDelim', 'String.SetString( '/; 2032
17109: StUnixPathSep', 'String.SetString( '; 2034
17110: STVERBUILD', 'LongInt.SetInt( 2); 3034
17111: STVERMAJOR', 'LongInt.SetInt( 0); 3032
17112: STVERMINOR', 'LongInt.SetInt( 1); 3033
17113: STVERRELEASE', 'LongInt.SetInt( 3); 3035
17114: stWin32Service', 'LongInt.Value.ts32 := ord(stWin32OwnProcess) or ord(stWin32ShareProcess); 2564
17115: SW_HIDE', 'LongInt.SetInt( 0); 386
17116: SW_MAX', 'LongInt.SetInt( 10); 399
17117: SW_MAXIMIZE', 'LongInt.SetInt( 3); 391
17118: SW_MINIMIZE', 'LongInt.SetInt( 6); 394
17119: SW_NORMAL', 'LongInt.SetInt( 1); 388
17120: SW_OTHERUNZOOM', 'LongInt.SetInt( 4); 408
17121: SW_OTHERZOOM', 'LongInt.SetInt( 2); 406
17122: SW_PARENTCLOSING', 'LongInt.SetInt( 1); 405
17123: SW_PARENTOPENING', 'LongInt.SetInt( 3); 407
17124: SW_RESTORE', 'LongInt.SetInt( 9); 397
17125: SW_SHOW', 'LongInt.SetInt( 5); 393

```

```

17126: SW_SHOWDEFAULT', 'LongInt.SetInt( 10); 398
17127: SW_SHOWMAXIMIZED', 'LongInt.SetInt( 3); 390
17128: SW_SHOWMINIMIZED', 'LongInt.SetInt( 2); 389
17129: SW_SHOWMINNOACTIVE', 'LongInt.SetInt( 7); 395
17130: SW_SHOWNA', 'LongInt.SetInt( 8); 396
17131: SW_SHOWNOACTIVATE', 'LongInt.SetInt( 4); 392
17132: SW_SHOWNORMAL', 'LongInt.SetInt( 1); 387
17133: SWhere', 'String.SetString( ' where ; 1920
17134: SYMBOL_CHARSET', 'LongInt.SetInt( 2); 171
17135: SYN_ATTR_COMMENT', 'LongInt.SetInt( 0); 3041
17136: SYN_ATTR_IDENTIFIER', 'LongInt.SetInt( 1); 3042
17137: SYN_ATTR_KEYWORD', 'LongInt.SetInt( 2); 3043
17138: SYN_ATTR_STRING', 'LongInt.SetInt( 3); 3044
17139: SYN_ATTR_SYMBOL', 'LongInt.SetInt( 5); 3046
17140: SYN_ATTR_WHITESPACE', 'LongInt.SetInt( 4); 3045
17141: SYNCHRONIZE', 'LongWord.SetUInt( $00100000); 196
17142: SYNEDIT_CLIPBOARD_FORMAT', 'String.SetString( 'SynEdit Control Block Type; 3039
17143: SYNEDIT_DEFAULT_OPTIONS', 'LongInt.Value.ts32 := ord(eoAutoIndent) or ord(eoDragDropEditing) or
ord(eoEnhanceEndKey) or ord(eoScrollPastEol) or ord(eoShowScrollHint) or ord(eoSmartTabs) or
ord(eoTabsToSpaces) or ord(eoSmartTabDelete) or ord(eoGroupUndo); 3040
17144: SynLineBreakGlyph', 'Char.SetString( '¶; //chr( $B6)); 3157
17145: SynSoftBreakGlyph', 'Char.SetString( '¬; //chr( $AC)); 3156
17146: SynSpaceGlyph', 'Char.SetString( '·; //chr( $B7)); 3158
17147: SynTabGlyph', 'Char.SetString( '»; 3155
17148: SysColCount', 'LongInt.SetInt( 30); 1608
17149: SZ_CONTENTTYPE_CDF', 'string.SetString( SZ_CONTENTTYPE_CDFA); 1766
17150: SZ_CONTENTTYPE_CDFA', 'string.SetString( 'application/x-cdf; 1764
17151: SZ_CONTENTTYPE_CDFW', 'string.SetString( 'application/x-cdf; 1765
17152: SZ_CONTENTTYPE_HTML', 'string.SetString( SZ_CONTENTTYPE_HTMLA); 1763
17153: SZ_CONTENTTYPE_HTMLA', 'string.SetString( 'text/html; 1761
17154: SZ_CONTENTTYPE_HTMLW', 'string.SetString( 'text/html; 1762
17155: TAB', 'Char.SetString( #$09); 952
17156: TAB', 'Char.SetString( #$09); 975
17157: TAB', 'Char.SetString( #9); 1043
17158: TAB', 'Char.SetString( #9); 1076
17159: TComplex_VERSION', 'Extended.setExtended( 5.01); 2448
17160: tdiError', 'LongInt.SetInt( 2); 912
17161: tdiInformation', 'LongInt.SetInt( 3); 913
17162: tdiNone', 'LongInt.SetInt( 0); 910
17163: tdiShield', 'LongInt.SetInt( 4); 914
17164: tdiWarning', 'LongInt.SetInt( 1); 911
17165: TErrorMessageSize', 'LongInt.SetInt( 2048); 1928
17166: tfCRLF', 'LongWord.SetUInt( $1); 2027
17167: tfCRLF', 'LongWord.SetUInt( $1); 2124
17168: TFTP_ACK', 'LongInt.SetInt( 4); 1223
17169: TFTP_DATA', 'LongInt.SetInt( 3); 1222
17170: TFTP_ERROR', 'LongInt.SetInt( 5); 1224
17171: TFTP_OACK', 'LongInt.SetInt( 6); 1225
17172: TFTP_RRQ', 'LongInt.SetInt( 1); 1220
17173: TFTP_WRQ', 'LongInt.SetInt( 2); 1221
17174: TH32CS_INHERIT', 'LongWord.SetUInt( $80000000); 3192
17175: TH32CS_SNAPHEAPLIST', 'LongWord.SetUInt( $00000001); 3188
17176: TH32CS_SNAPMODULE', 'LongWord.SetUInt( $00000008); 3191
17177: TH32CS_SNAPPROCESS', 'LongWord.SetUInt( $00000002); 3189
17178: TH32CS_SNAPTHREAD', 'LongWord.SetUInt( $00000004); 3190
17179: THREAD_BASE_PRIORITY_IDLE', 'LongInt.SetInt( - 15); 187
17180: THREAD_BASE_PRIORITY_LOWRT', 'LongInt.SetInt( 15); 184
17181: THREAD_BASE_PRIORITY_MAX', 'LongInt.SetInt( 2); 185
17182: THREAD_BASE_PRIORITY_MIN', 'LongInt.SetInt( - 2); 186
17183: THREAD_PRIORITY_ABOVE_NORMAL', 'LongInt.SetInt( THREAD_PRIORITY_HIGHEST - 1); 192
17184: THREAD_PRIORITY_BELOW_NORMAL', 'LongInt.SetInt( THREAD_PRIORITY_LOWEST + 1); 189
17185: THREAD_PRIORITY_ERROR_RETURN', 'longint.SetInt( MAXLONG); 193
17186: THREAD_PRIORITY_HIGHEST', 'longint.SetInt( THREAD_BASE_PRIORITY_MAX); 191
17187: THREAD_PRIORITY_IDLE', 'longint.SetInt( THREAD_BASE_PRIORITY_IDLE); 195
17188: THREAD_PRIORITY_LOWEST', 'longint.SetInt( THREAD_BASE_PRIORITY_MIN); 188
17189: THREAD_PRIORITY_NORMAL', 'LongInt.SetInt( 0); 190
17190: THREAD_PRIORITY_TIME_CRITICAL', 'longint.SetInt( THREAD_BASE_PRIORITY_LOWRT); 194
17191: ThreePi', 'Extended.SetExtended( 9.4247779607693797153879301498385); 1685
17192: ThreePi', 'Float.SetString( 9.4247779607693797153879301498385); 1297
17193: ThreePi', 'Float.SetString( 9.4247779607693797153879301498385); 1365
17194: TickFrequency', 'LongInt.SetInt( 1000); 2236
17195: TIME_BASEDATE', 'LongInt.SetInt( 2); 1068
17196: TIME_BASEDATE', 'LongInt.SetInt( 2); 1214
17197: TIME_BASEDATE', 'LongInt.SetInt( 2); 4
17198: TIME_TIMEOUT', 'LongInt.SetInt( 2500); 1215
17199: TimeColon', 'String.SetString( ':'; 3028
17200: TimeOnly', 'String.SetString( 't; 3027
17201: TIMEUDP_DEFBASEDATE', 'LongInt.SetInt( 2); 2442
17202: TinySetLen', 'LongInt.SetInt( 3); 2061
17203: TLS_COMPRESSED_FRAGMENT_MAXSIZE', 'LongInt.SetInt( 16384 + 1024); 3388
17204: TLS_MAC_MAXDIGESTSIZE', 'LongInt.SetInt( 64); 3386
17205: TLS_PLAINTEXT_FRAGMENT_MAXSIZE', 'LongInt.SetInt( 16384 - 1); 3387
17206: TLSError_BadProtocol', 'LongInt.SetInt( 6); 3384
17207: TLSError_DecodeError', 'LongInt.SetInt( 5); 3383
17208: TLSError_InvalidBuffer', 'LongInt.SetInt( 1); 3379
17209: TLSError_InvalidCertificate', 'LongInt.SetInt( 3); 3381
17210: TLSError_InvalidParameter', 'LongInt.SetInt( 2); 3380
17211: TLSError_InvalidState', 'LongInt.SetInt( 4); 3382
17212: TLSError_None', 'LongInt.SetInt( 0); 3378

```

```
17213: TLSLibraryVersion', 'String.SetString( '1.00; 3377
17214: TLSSessionIDMaxLen', 'LongInt.SetInt( 32); 3385
17215: TNC_AO', 'Char.SetString( #245); 1160
17216: TNC_AYT', 'Char.SetString( #246); 1161
17217: TNC_BREAK', 'Char.SetString( #243); 1158
17218: TNC_DATA_MARK', 'Char.SetString( #242); 1157
17219: TNC_DO', 'Char.SetString( #253); 1168
17220: TNC_DONT', 'Char.SetString( #254); 1169
17221: TNC_EC', 'Char.SetString( #247); 1162
17222: TNC_EL', 'Char.SetString( #248); 1163
17223: TNC_EOR', 'Char.SetString( #239); 1154
17224: TNC_GA', 'Char.SetString( #249); 1164
17225: TNC_IAC', 'Char.SetString( #255); 1170
17226: TNC_IP', 'Char.SetString( #244); 1159
17227: TNC_NOP', 'Char.SetString( #241); 1156
17228: TNC_SB', 'Char.SetString( #250); 1165
17229: TNC_SE', 'Char.SetString( #240); 1155
17230: TNC_WILL', 'Char.SetString( #251); 1166
17231: TNC_WONT', 'Char.SetString( #252); 1167
17232: TNO_3270REGIME', 'Char.SetString( #29); 1200
17233: TNO_AMSN', 'Char.SetString( #4); 1175
17234: TNO_AUTH', 'Char.SetString( #37); 1207
17235: TNO_BINARY', 'Char.SetString( #0); 1171
17236: TNO_BYTE_MACRO', 'Char.SetString( #19); 1190
17237: TNO_DET', 'Char.SetString( #20); 1191
17238: TNO_EA', 'Char.SetString( #17); 1188
17239: TNO_ECHO', 'Char.SetString( #1); 1172
17240: TNO_ENCRYPT', 'Char.SetString( #38); 1208
17241: TNO_EOL', 'Char.SetString( #255); 1209
17242: TNO_EOR', 'Char.SetString( #25); 1196
17243: TNO_LINEMODE', 'Char.SetString( #34); 1205
17244: TNO_LOGOUT', 'Char.SetString( #18); 1189
17245: TNO_NAWS', 'Char.SetString( #31); 1202
17246: TNO_OCRD', 'Char.SetString( #10); 1181
17247: TNO_OFD', 'Char.SetString( #13); 1184
17248: TNO_OHTD', 'Char.SetString( #12); 1183
17249: TNO_OHTS', 'Char.SetString( #11); 1182
17250: TNO_OLD', 'Char.SetString( #16); 1187
17251: TNO_OLW', 'Char.SetString( #8); 1179
17252: TNO_OM', 'Char.SetString( #27); 1198
17253: TNO_OPS', 'Char.SetString( #9); 1180
17254: TNO_OVT', 'Char.SetString( #14); 1185
17255: TNO_OVTD', 'Char.SetString( #15); 1186
17256: TNO_RCTE', 'Char.SetString( #7); 1178
17257: TNO_RECONNECT', 'Char.SetString( #2); 1173
17258: TNO_RFLOW', 'Char.SetString( #33); 1204
17259: TNO_SGA', 'Char.SetString( #3); 1174
17260: TNO_SL', 'Char.SetString( #23); 1194
17261: TNO_STATUS', 'Char.SetString( #5); 1176
17262: TNO_SUPDUP', 'Char.SetString( #21); 1192
17263: TNO_SUPDUP_OUTPUT', 'Char.SetString( #22); 1193
17264: TNO_TACACS_ID', 'Char.SetString( #26); 1197
17265: TNO_TERM_SPEED', 'Char.SetString( #32); 1203
17266: TNO_TERMTYPE', 'Char.SetString( #24); 1195
17267: TNO_TIMING_MARK', 'Char.SetString( #6); 1177
17268: TNO_TLN', 'Char.SetString( #28); 1199
17269: TNO_X3PAD', 'Char.SetString( #30); 1201
17270: TNO_XDISPLOC', 'Char.SetString( #35); 1206
17271: TNOS_NAME', 'Char.SetString( #3); 1213
17272: TNOS_REPLY', 'Char.SetString( #2); 1212
17273: TNOS_TERM_IS', 'Char.SetString( #0); 1210
17274: TNOS_TERMTYPE_SEND', 'Char.SetString( #1); 1211
17275: toAccept', 'LongInt.SetInt(( 14 )); 1002
17276: toAllow', 'LongInt.SetInt(( 26 )); 1014
17277: toAuthorization', 'LongInt.SetInt(( 18 )); 1006
17278: toCacheControl', 'LongInt.SetInt(( 19 )); 1007
17279: toConnection', 'LongInt.SetInt(( 27 )); 1015
17280: toContentDisposition', 'LongInt.SetInt(( 29 )); 1017
17281: toContentEncoding', 'LongInt.SetInt(( 24 )); 1012
17282: toContentLength', 'LongInt.SetInt(( 16 )); 1004
17283: toContentType', 'LongInt.SetInt(( 15 )); 1003
17284: toContentVersion', 'LongInt.SetInt(( 25 )); 1013
17285: toCookie', 'LongInt.SetInt(( 28 )); 1016
17286: toCOPY', 'LongInt.SetInt( ( 12 )); 1000
17287: toDate', 'LongInt.SetInt(( 20 )); 1008
17288: toDELETE', 'LongInt.SetInt(( 9 )); 997
17289: toEOF', 'LongInt.SetInt( Char ( 0 )); 481
17290: toEOF', 'LongInt.SetInt( Char ( 0 )); 824
17291: toEOL', 'LongInt.SetInt(( 5 )); 993
17292: toFloat', 'LongInt.SetInt( Char ( 4 )); 828
17293: toFrom', 'LongInt.SetInt(( 21 )); 1009
17294: toGET', 'LongInt.SetInt(( 6 )); 994
17295: toHEAD', 'LongInt.SetInt(( 7 )); 995
17296: toHost', 'LongInt.SetInt(( 22 )); 1010
17297: toIfModified', 'LongInt.SetInt( ( 23 )); 1011
17298: toInteger', 'LongInt.SetInt( Char ( 3 )); 827
17299: toPATCH', 'LongInt.SetInt(( 11 )); 999
17300: toPOST', 'LongInt.SetInt(( 10 )); 998
17301: toPUT', 'LongInt.SetInt(( 8 )); 996
```



```

17302: toReferer','LongInt.SetInt(( 17 )); 1005
17303: toString','LongInt.SetInt( Char ( 2 )); 826
17304: toSymbol','LongInt.SetInt( Char ( 1 )); 825
17305: TotalErrors','LongInt.SetInt( 7); 2734
17306: toUserAgent','LongInt.SetInt(( 13 )); 1001
17307: toWString','Char.SetString(#5); 488
17308: toWString','LongInt.SetInt( Char ( 5 )); 829
17309: tpHigher','LongInt.SetInt( - 7); 1087
17310: tpHigher','LongInt.SetInt( - 7); 1105
17311: tpHighest','LongInt.SetInt( - 13); 1088
17312: tpHighest','LongInt.SetInt( - 13); 1106
17313: tpIdle','LongInt.SetInt( 19); 1083
17314: tpIdle','LongInt.SetInt( 19); 1101
17315: tpLower','LongInt.SetInt( 6); 1085
17316: tpLower','LongInt.SetInt( 6); 1103
17317: tpLowest','LongInt.SetInt( 12); 1084
17318: tpLowest','LongInt.SetInt( 12); 1102
17319: tpNormal','LongInt.SetInt( 0); 1086
17320: tpNormal','LongInt.SetInt( 0); 1104
17321: tpTimeCritical','LongInt.SetInt( - 20); 1089
17322: tpTimeCritical','LongInt.SetInt( - 20); 1107
17323: TrueExpr','String.SetString( '0=0; 1591
17324: TSC_FLAG','LongWord.SetUInt( $00000010); 1428
17325: TSynTabChar','Char.SetString( #9); 3154
17326: TSynWordBreakChars','String.SetString( '.' or ',' or ';' or ':' or '"' or ''' or '!' or '?' or
[' or ']' or '(' or ' ' or '{' or '}' or '^' or '-' or '=' or '+' or '-' or '*' or '/' or '\
or '|'; 3153
17327: TVM_GETLINECOLOR','LongInt.SetInt( TV_FIRST + 41); 2760
17328: TVM_SETLINECOLOR','LongInt.SetInt( TV_FIRST + 40); 2759
17329: TwoDigitYearCenturyWindow','Word.SetInt(50); 2018
17330: TwoPi','Extended.SetExtended( 6.283185307179586476925286766559); 1684
17331: TwoPi','Extended.setExtended( 6.28318530717958647693); 3204
17332: TwoPi','Float.SetString( 6.283185307179586476925286766559); 1296
17333: TwoPi','Float.SetString( 6.283185307179586476925286766559); 1364
17334: TwoToPower63','Extended.SetExtended( 9223372036854775808.0); 1696
17335: TwoToPower63','Float.SetString( 9223372036854775808.0); 1308
17336: TwoToPower63','Float.SetString( 9223372036854775808.0); 1376
17337: UnixDateDelta','LongInt.SetInt( 25569); 40
17338: UNIXSTARTDATE','TidDateTime.SetExtended(25569.0); 1067
17339: UNIXSTARTDATE','TidDateTime.SetString( 25569.0); 3
17340: URL_APPLY_DEFAULT','LongWord.SetUInt( $00000001); 1827
17341: URL_APPLY_FORCEAPPLY','LongWord.SetUInt( $00000008); 1830
17342: URL_APPLY_GUESSFILE','LongWord.SetUInt( $00000004); 1829
17343: URL_APPLY_GUESSSCHEME','LongWord.SetUInt( $00000002); 1828
17344: URL_BROWSER_MODE','SetString( URL_DONT_ESCAPE_EXTRA_INFO); 1815
17345: URL_CONVERT_IF_DOSPATH','LongWord.SetUInt( $00200000); 1820
17346: URL_DONT_ESCAPE_EXTRA_INFO','LongWord.SetUInt( $02000000); 1813
17347: URL_DONT_SIMPLIFY','LongWord.SetUInt( $08000000); 1817
17348: URL_DONT_UNESCAPE_EXTRA_INFO','SetString( URL_DONT_ESCAPE_EXTRA_INFO); 1814
17349: URL_ESCAPE_PERCENT','LongWord.SetUInt( $00001000); 1824
17350: URL_ESCAPE_SEGMENT_ONLY','LongWord.SetUInt( $00002000); 1825
17351: URL_ESCAPE_SPACES_ONLY','LongWord.SetUInt( $04000000); 1816
17352: URL_ESCAPE_UNSAFE','LongWord.SetUInt( $20000000); 1810
17353: URL_FILE_USE_PATHURL','LongWord.SetUInt( $00010000); 1823
17354: URL_INTERNAL_PATH','LongWord.SetUInt( $00800000); 1822
17355: URL_NO_META','longword.SetUInt( URL_DONT_SIMPLIFY); 1818
17356: URL_PART_HOSTNAME','LongInt.SetInt( 2); 1797
17357: URL_PART_NONE','LongInt.SetInt( 0); 1795
17358: URL_PART_PASSWORD','LongInt.SetInt( 4); 1799
17359: URL_PART_PORT','LongInt.SetInt( 5); 1800
17360: URL_PART_QUERY','LongInt.SetInt( 6); 1801
17361: URL_PART_SCHEME','LongInt.SetInt( 1); 1796
17362: URL_PART_USERNAME','LongInt.SetInt( 3); 1798
17363: URL_PARTFLAG_KEEPScheme','LongWord.SetUInt( $00000001); 1826
17364: URL_PLUGGABLE_PROTOCOL','LongWord.SetUInt( $40000000); 1811
17365: URL_SCHEME_ABOUT','LongInt.SetInt( 17); 1792
17366: URL_SCHEME_FILE','LongInt.SetInt( 9); 1784
17367: URL_SCHEME_FTP','LongInt.SetInt( 1); 1776
17368: URL_SCHEME_GOPHER','LongInt.SetInt( 3); 1778
17369: URL_SCHEME_HTTP','LongInt.SetInt( 2); 1777
17370: URL_SCHEME_HTTPS','LongInt.SetInt( 11); 1786
17371: URL_SCHEME_INVALID','LongInt.SetInt( - 1); 1774
17372: URL_SCHEME_JAVASCRIPT','LongInt.SetInt( 15); 1790
17373: URL_SCHEME_LOCAL','LongInt.SetInt( 14); 1789
17374: URL_SCHEME_MAILTO','LongInt.SetInt( 4); 1779
17375: URL_SCHEME_MAXVALUE','LongInt.SetInt( 19); 1794
17376: URL_SCHEME_MK','LongInt.SetInt( 10); 1785
17377: URL_SCHEME_NEWS','LongInt.SetInt( 5); 1780
17378: URL_SCHEME_NNTP','LongInt.SetInt( 6); 1781
17379: URL_SCHEME_RES','LongInt.SetInt( 18); 1793
17380: URL_SCHEME_SHELL','LongInt.SetInt( 12); 1787
17381: URL_SCHEME_SNEWS','LongInt.SetInt( 13); 1788
17382: URL_SCHEME_TELNET','LongInt.SetInt( 7); 1782
17383: URL_SCHEME_UNKNOWN','LongInt.SetInt( 0); 1775
17384: URL_SCHEME_VBSCRIPT','LongInt.SetInt( 16); 1791
17385: URL_SCHEME_WAIS','LongInt.SetInt( 8); 1783
17386: URL_UNESCAPE','LongWord.SetUInt( $10000000); 1809
17387: URL_UNESCAPE_HIGH_ANSI_ONLY','LongWord.SetUInt( $00400000); 1821
17388: URL_UNESCAPE_INPLACE','LongWord.SetUInt( $00100000); 1819

```

```

17389: URL_WININET_COMPATIBILITY', 'LongWord.SetUInt( DWORD ( $80000000 )); 1812
17390: URLIS_APPLIABLE', 'LongInt.SetInt( 4); 1806
17391: URLIS_DIRECTORY', 'LongInt.SetInt( 5); 1807
17392: URLIS_FILEURL', 'LongInt.SetInt( 3); 1805
17393: URLIS_HASQUERY', 'LongInt.SetInt( 6); 1808
17394: URLIS_NOHISTORY', 'LongInt.SetInt( 2); 1804
17395: URLIS_OPAQUE', 'LongInt.SetInt( 1); 1803
17396: URLIS_URL', 'LongInt.SetInt( 0); 1802
17397: USDecimalSeparator', 'String.SetString( '.; 1616
17398: user32', 'String.SetString( 'user32.dll; 15
17399: Usp10', 'String.SetString( 'usp10.dll; 3186
17400: Usp10', 'String.SetString( 'usp10.dll; 3375
17401: val_DateTimeDefault', 'String.SetString( 'Default Format; 2173
17402: varAny', 'LongWord.SetUInt( $0101); 2085
17403: varAny', 'LongWord.SetUInt( $0101); 324
17404: varArray', 'LongWord.SetUInt( $2000); 2087
17405: varArray', 'LongWord.SetUInt( $2000); 326
17406: varBoolean', 'LongWord.SetUInt( $000B); 2075
17407: varBoolean', 'LongWord.SetUInt( $000B); 314
17408: varByRef', 'LongWord.SetUInt( $4000); 2088
17409: varByRef', 'LongWord.SetUInt( $4000); 327
17410: varByte', 'LongWord.SetUInt( $0011); 2079
17411: varByte', 'LongWord.SetUInt( $0011); 318
17412: varCurrency', 'LongWord.SetUInt( $0006); 2070
17413: varCurrency', 'LongWord.SetUInt( $0006); 309
17414: varDate', 'LongWord.SetUInt( $0007); 2071
17415: varDate', 'LongWord.SetUInt( $0007); 310
17416: varDispatch', 'LongWord.SetUInt( $0009); 2073
17417: varDispatch', 'LongWord.SetUInt( $0009); 312
17418: varDouble', 'LongWord.SetUInt( $0005); 2069
17419: varDouble', 'LongWord.SetUInt( $0005); 308
17420: varEmpty', 'LongWord.SetUInt( $0000); 2064
17421: varEmpty', 'LongWord.SetUInt( $0000); 303
17422: varError', 'LongWord.SetUInt( $000A); 2074
17423: varError', 'LongWord.SetUInt( $000A); 313
17424: VARIABLE_PITCH', 'LongInt.SetInt( 2); 167
17425: varInt64', 'LongWord.SetUInt( $0014); 2082
17426: varInt64', 'LongWord.SetUInt( $0014); 321
17427: varInteger', 'LongWord.SetUInt( $0003); 2067
17428: varInteger', 'LongWord.SetUInt( $0003); 306
17429: varLongWord', 'LongWord.SetUInt( $0013); 2081
17430: varLongWord', 'LongWord.SetUInt( $0013); 320
17431: varNull', 'LongWord.SetUInt( $0001); 2065
17432: varNull', 'LongWord.SetUInt( $0001); 304
17433: varOleStr', 'LongWord.SetUInt( $0008); 2072
17434: varOleStr', 'LongWord.SetUInt( $0008); 311
17435: varShortInt', 'LongWord.SetUInt( $0010); 2078
17436: varShortInt', 'LongWord.SetUInt( $0010); 317
17437: varSingle', 'LongWord.SetUInt( $0004); 2068
17438: varSingle', 'LongWord.SetUInt( $0004); 307
17439: varSmallint', 'LongWord.SetUInt( $0002); 2066
17440: varSmallint', 'LongWord.SetUInt( $0002); 305
17441: varStrArg', 'LongWord.SetUInt( $0048); 2083
17442: varStrArg', 'LongWord.SetUInt( $0048); 322
17443: varString', 'LongWord.SetUInt( $0100); 2084
17444: varString', 'LongWord.SetUInt( $0100); 323
17445: varTypeMask', 'LongWord.SetUInt( $0FFF); 2086
17446: varTypeMask', 'LongWord.SetUInt( $0FFF); 325
17447: varUnknown', 'LongWord.SetUInt( $000D); 2077
17448: varUnknown', 'LongWord.SetUInt( $000D); 316
17449: varVariant', 'LongWord.SetUInt( $000C); 2076
17450: varVariant', 'LongWord.SetUInt( $000C); 315
17451: varWord', 'LongWord.SetUInt( $0012); 2080
17452: varWord', 'LongWord.SetUInt( $0012); 319
17453: VER_NT_DOMAIN_CONTROLLER', 'LongWord.SetUInt( $00000002); 3412
17454: VER_NT_SERVER', 'LongWord.SetUInt( $00000003); 3413
17455: VER_NT_WORKSTATION', 'LongWord.SetUInt( $00000001); 3411
17456: VER_PLATFORM_WIN32_CE', 'LongInt.SetInt( 3); 129
17457: VER_PLATFORM_WIN32_NT', 'LongInt.SetInt( 2); 128
17458: VER_PLATFORM_WIN32_WINDOWS', 'LongInt.SetInt( 1); 127
17459: VER_PLATFORM_WIN32s', 'LongInt.SetInt( 0); 126
17460: VER_SERVER_NT', 'LongWord.SetUInt( DWORD ( $80000000 )); 3414
17461: VER_SUITE_BACKOFFICE', 'LongWord.SetUInt( $00000004); 3418
17462: VER_SUITE_BLADE', 'LongWord.SetUInt( $00000400); 3426
17463: VER_SUITE_COMMUNICATIONS', 'LongWord.SetUInt( $00000008); 3419
17464: VER_SUITE_COMPUTE_SERVER', 'LongWord.SetUInt( $00004000); 3430
17465: VER_SUITE_DATACENTER', 'LongWord.SetUInt( $00000080); 3423
17466: VER_SUITE_EMBEDDED_RESTRICTED', 'LongWord.SetUInt( $00000800); 3427
17467: VER_SUITE_EMBEDDEDNT', 'LongWord.SetUInt( $00000040); 3422
17468: VER_SUITE_ENTERPRISE', 'LongWord.SetUInt( $00000002); 3417
17469: VER_SUITE_PERSONAL', 'LongWord.SetUInt( $00000200); 3425
17470: VER_SUITE_SECURITY_APPLIANCE', 'LongWord.SetUInt( $00001000); 3428
17471: VER_SUITE_SINGLEUSERTS', 'LongWord.SetUInt( $00000100); 3424
17472: VER_SUITE_SMALLBUSINESS', 'LongWord.SetUInt( $00000001); 3416
17473: VER_SUITE_SMALLBUSINESS_RESTRICTED', 'LongWord.SetUInt( $00000020); 3421
17474: VER_SUITE_STORAGE_SERVER', 'LongWord.SetUInt( $00002000); 3429
17475: VER_SUITE_TERMINAL', 'LongWord.SetUInt( $00000010); 3420
17476: VER_WORKSTATION_NT', 'LongWord.SetUInt( $40000000); 3415
17477: VERSION', 'String.SetString( '3.9.9.7; 945

```

```
17478: VK_ACCEPT', 'LongInt.SetInt( 30); 447
17479: VK_BACK', 'LongInt.SetInt( 8); 430
17480: VK_CANCEL', 'LongInt.SetInt( 3); 426
17481: VK_CAPITAL', 'LongInt.SetInt( 20); 438
17482: VK_CLEAR', 'LongInt.SetInt( 12); 432
17483: VK_CONTROL', 'LongInt.SetInt( 17); 435
17484: VK_CONVERT', 'LongInt.SetInt( 28); 445
17485: VK_DELETE', 'LongInt.SetInt( 46); 464
17486: VK_DOWN', 'LongInt.SetInt( 40); 458
17487: VK_END', 'LongInt.SetInt( 35); 453
17488: VK_ESCAPE', 'LongInt.SetInt( 27); 449
17489: VK_EXECUTE', 'LongInt.SetInt( 43); 461
17490: VK_F1', 'LongInt.SetInt( 112); 130
17491: VK_F10', 'LongInt.SetInt( 121); 139
17492: VK_F11', 'LongInt.SetInt( 122); 140
17493: VK_F12', 'LongInt.SetInt( 123); 141
17494: VK_F2', 'LongInt.SetInt( 113); 131
17495: VK_F3', 'LongInt.SetInt( 114); 132
17496: VK_F4', 'LongInt.SetInt( 115); 133
17497: VK_F5', 'LongInt.SetInt( 116); 134
17498: VK_F6', 'LongInt.SetInt( 117); 135
17499: VK_F7', 'LongInt.SetInt( 118); 136
17500: VK_F8', 'LongInt.SetInt( 119); 137
17501: VK_F9', 'LongInt.SetInt( 120); 138
17502: VK_FINAL', 'LongInt.SetInt( 24); 442
17503: VK_HANGUL', 'LongInt.SetInt( 21); 440
17504: VK_HANJA', 'LongInt.SetInt( 25); 443
17505: VK_HELP', 'LongInt.SetInt( 47); 465
17506: VK_HOME', 'LongInt.SetInt( 36); 454
17507: VK_INSERT', 'LongInt.SetInt( 45); 463
17508: VK_JUNJA', 'LongInt.SetInt( 23); 441
17509: VK_KANA', 'LongInt.SetInt( 21); 439
17510: VK_KANJI', 'LongInt.SetInt( 25); 444
17511: VK_LBUTTON', 'LongInt.SetInt( 1); 424
17512: VK_LEFT', 'LongInt.SetInt( 37); 455
17513: VK_MBUTTON', 'LongInt.SetInt( 4); 427
17514: VK_MENU', 'LongInt.SetInt( 18); 436
17515: VK_MODECHANGE', 'LongInt.SetInt( 31); 448
17516: VK_NEXT', 'LongInt.SetInt( 34); 452
17517: VK_NONCONVERT', 'LongInt.SetInt( 29); 446
17518: VK_PAUSE', 'LongInt.SetInt( 19); 437
17519: VK_PRINT', 'LongInt.SetInt( 42); 460
17520: VK_PRIOR', 'LongInt.SetInt( 33); 451
17521: VK_RBUTTON', 'LongInt.SetInt( 2); 425
17522: VK_RETURN', 'LongInt.SetInt( 13); 433
17523: VK_RIGHT', 'LongInt.SetInt( 39); 457
17524: VK_SELECT', 'LongInt.SetInt( 41); 459
17525: VK_SHIFT', 'LongWord.SetUInt( $10); 434
17526: VK_SNAPSHOT', 'LongInt.SetInt( 44); 462
17527: VK_SPACE', 'LongWord.SetUInt( $20); 450
17528: VK_TAB', 'LongInt.SetInt( 9); 431
17529: VK_UP', 'LongInt.SetInt( 38); 456
17530: VK_XBUTTON1', 'LongInt.SetInt( 5); 428
17531: VK_XBUTTON2', 'LongInt.SetInt( 6); 429
17532: VME_FLAG', 'LongWord.SetUInt( $00000002); 1425
17533: vmtAutoTable', 'LongInt.SetInt( - 68); 2108
17534: vmtAutoTable', 'LongInt.SetInt( - 68); 347
17535: vmtClassName', 'LongInt.SetInt( - 44); 2114
17536: vmtClassName', 'LongInt.SetInt( - 44); 353
17537: vmtDynamicTable', 'LongInt.SetInt( - 48); 2113
17538: vmtDynamicTable', 'LongInt.SetInt( - 48); 352
17539: vmtFieldTable', 'LongInt.SetInt( - 56); 2111
17540: vmtFieldTable', 'LongInt.SetInt( - 56); 350
17541: vmtInitTable', 'LongInt.SetInt( - 64); 2109
17542: vmtInitTable', 'LongInt.SetInt( - 64); 348
17543: vmtInstanceSize', 'LongInt.SetInt( - 40); 2115
17544: vmtInstanceSize', 'LongInt.SetInt( - 40); 354
17545: vmtIntfTable', 'LongInt.SetInt( - 72); 2107
17546: vmtIntfTable', 'LongInt.SetInt( - 72); 346
17547: vmtMethodTable', 'LongInt.SetInt( - 52); 2112
17548: vmtMethodTable', 'LongInt.SetInt( - 52); 351
17549: vmtParent', 'LongInt.SetInt( - 36); 2116
17550: vmtParent', 'LongInt.SetInt( - 36); 355
17551: vmtSelfPtr', 'LongInt.SetInt( - 76); 2106
17552: vmtSelfPtr', 'LongInt.SetInt( - 76); 345
17553: vmtTypeInfo', 'LongInt.SetInt( - 60); 2110
17554: vmtTypeInfo', 'LongInt.SetInt( - 60); 349
17555: vtAnsiString', 'LongInt.SetInt( 11); 2100
17556: vtAnsiString', 'LongInt.SetInt( 11); 339
17557: vtBoolean', 'LongInt.SetInt( 1); 2090
17558: vtBoolean', 'LongInt.SetInt( 1); 329
17559: vtChar', 'LongInt.SetInt( 2); 2091
17560: vtChar', 'LongInt.SetInt( 2); 330
17561: vtClass', 'LongInt.SetInt( 8); 2097
17562: vtClass', 'LongInt.SetInt( 8); 336
17563: vtCurrency', 'LongInt.SetInt( 12); 2101
17564: vtCurrency', 'LongInt.SetInt( 12); 340
17565: vtExtended', 'LongInt.SetInt( 3); 2092
17566: vtExtended', 'LongInt.SetInt( 3); 331
```

```

17567: vtInt64', 'LongInt.SetInt( 16); 2105
17568: vtInt64', 'LongInt.SetInt( 16); 344
17569: vtInteger', 'LongInt.SetInt( 0); 2089
17570: vtInteger', 'LongInt.SetInt( 0); 328
17571: vtInterface', 'LongInt.SetInt( 14); 2103
17572: vtInterface', 'LongInt.SetInt( 14); 342
17573: vtObject', 'LongInt.SetInt( 7); 2096
17574: vtObject', 'LongInt.SetInt( 7); 335
17575: vtPChar', 'LongInt.SetInt( 6); 2095
17576: vtPChar', 'LongInt.SetInt( 6); 334
17577: vtPointer', 'LongInt.SetInt( 5); 2094
17578: vtPointer', 'LongInt.SetInt( 5); 333
17579: vtPWideChar', 'LongInt.SetInt( 10); 2099
17580: vtPWideChar', 'LongInt.SetInt( 10); 338
17581: vtString', 'LongInt.SetInt( 4); 2093
17582: vtString', 'LongInt.SetInt( 4); 332
17583: vtVariant', 'LongInt.SetInt( 13); 2102
17584: vtVariant', 'LongInt.SetInt( 13); 341
17585: vtWideChar', 'LongInt.SetInt( 9); 2098
17586: vtWideChar', 'LongInt.SetInt( 9); 337
17587: vtWideString', 'LongInt.SetInt( 15); 2104
17588: vtWideString', 'LongInt.SetInt( 15); 343
17589: WAIT_OBJECT_0', 'LongWord.SetUInt($00000000+0); 183
17590: WAIT_TIMEOUT', 'LongWord.SetUInt($00000102); 181
17591: WaitCursor', 'TCursor.SetInt( crHourGlass); 1640
17592: wdDoNotSaveChanges', 'LongInt.SetInt( 0); 2711
17593: wdTableFormatGrid1', 'LongWord.SetUInt( TJvWordGridFormat ( $10 )); 2712
17594: wdTableFormatGrid2', 'LongWord.SetUInt( TJvWordGridFormat ( $11 )); 2713
17595: wdTableFormatGrid3', 'LongWord.SetUInt( TJvWordGridFormat ( $12 )); 2714
17596: wdTableFormatGrid4', 'LongWord.SetUInt( TJvWordGridFormat ( $13 )); 2715
17597: wdTableFormatGrid5', 'LongWord.SetUInt( TJvWordGridFormat ( $14 )); 2716
17598: wdTableFormatGrid6', 'LongWord.SetUInt( TJvWordGridFormat ( $15 )); 2717
17599: wdTableFormatGrid7', 'LongWord.SetUInt( TJvWordGridFormat ( $16 )); 2718
17600: wdTableFormatGrid8', 'LongWord.SetUInt( TJvWordGridFormat ( $17 )); 2719
17601: WebColorsCount', 'LongInt.SetInt( 140); 783
17602: WebNamedColorsCount', 'LongInt.SetInt( 138); 986
17603: WeekDayOnly', 'String.SetString( 'w'; 3016
17604: WeekDayOnlyU', 'String.SetString( 'W'; 3017
17605: WeeksPerFortnight', 'LongInt.SetInt( 2); 871
17606: WH_CALLWNDPROC', 'LongInt.SetInt( 4); 148
17607: WH_CALLWNDPROCRET', 'LongInt.SetInt( 12); 156
17608: WH_CBT', 'LongInt.SetInt( 5); 149
17609: WH_DEBUG', 'LongInt.SetInt( 9); 153
17610: WH_FOREGROUNDIDLE', 'LongInt.SetInt( 11); 155
17611: WH_GETMESSAGE', 'LongInt.SetInt( 3); 147
17612: WH_HARDWARE', 'LongInt.SetInt( 8); 152
17613: WH_JOURNALPLAYBACK', 'LongInt.SetInt( 1); 145
17614: WH_JOURNALRECORD', 'LongInt.SetInt( 0); 144
17615: WH_KEYBOARD', 'LongInt.SetInt( 2); 146
17616: WH_MAX', 'LongInt.SetInt( 12); 157
17617: WH_MAXHOOK', 'longint.Setint( WH_MAX); 159
17618: WH_MIN', 'LongInt.SetInt( - 1); 142
17619: WH_MINHOOK', 'longint.Setint( WH_MIN); 158
17620: WH_MOUSE', 'LongInt.SetInt( 7); 151
17621: WH_MSGFILTER', 'LongInt.SetInt( - 1); 143
17622: WH_SHELL', 'LongInt.SetInt( 10); 154
17623: WH_SYSMSGFILTER', 'LongInt.SetInt( 6); 150
17624: WHEEL_DELTA', 'LongInt.SetInt( 120); 1634
17625: WHEEL_PAGESCROLL', '.SetInt( MAXDWORD); 1635
17626: WHITENESS', 'LongWord.SetUInt( $00FF0062); 798
17627: Win32BuildNumber', 'Integer.SetInt( 0); 362
17628: Win32CSDVersion', 'string.SetString( ' '; 363
17629: Win32MajorVersion', 'Integer.SetInt( 0); 360
17630: Win32MinorVersion', 'Integer.SetInt( 0); 361
17631: Win32Platform', 'Integer.SetInt( 0); 359
17632: WINAMP_BUTTON1', 'LongInt.SetInt( 40044); 3303
17633: WINAMP_BUTTON1_CTRL', 'LongInt.SetInt( 40154); 3313
17634: WINAMP_BUTTON1_SHIFT', 'LongInt.SetInt( 40144); 3308
17635: WINAMP_BUTTON2', 'LongInt.SetInt( 40045); 3304
17636: WINAMP_BUTTON2_CTRL', 'LongInt.SetInt( 40155); 3314
17637: WINAMP_BUTTON2_SHIFT', 'LongInt.SetInt( 40145); 3309
17638: WINAMP_BUTTON3', 'LongInt.SetInt( 40046); 3305
17639: WINAMP_BUTTON3_CTRL', 'LongInt.SetInt( 40156); 3315
17640: WINAMP_BUTTON3_SHIFT', 'LongInt.SetInt( 40146); 3310
17641: WINAMP_BUTTON4', 'LongInt.SetInt( 40047); 3306
17642: WINAMP_BUTTON4_CTRL', 'LongInt.SetInt( 40157); 3316
17643: WINAMP_BUTTON4_SHIFT', 'LongInt.SetInt( 40147); 3311
17644: WINAMP_BUTTON5', 'LongInt.SetInt( 40048); 3307
17645: WINAMP_BUTTON5_CTRL', 'LongInt.SetInt( 40158); 3317
17646: WINAMP_BUTTON5_SHIFT', 'LongInt.SetInt( 40148); 3312
17647: WINAMP_FFWD5S', 'LongInt.SetInt( 40060); 3301
17648: WINAMP_FILE_PLAY', 'LongInt.SetInt( 40029); 3318
17649: WINAMP_HELP_ABOUT', 'LongInt.SetInt( 40041); 3321
17650: WINAMP_OPTIONS_AOT', 'LongInt.SetInt( 40019); 3320
17651: WINAMP_OPTIONS_EQ', 'LongInt.SetInt( 40036); 3297
17652: WINAMP_OPTIONS_PLEDIT', 'LongInt.SetInt( 40040); 3298
17653: WINAMP_OPTIONS_PREFS', 'LongInt.SetInt( 40012); 3319
17654: WINAMP_REW5S', 'LongInt.SetInt( 40061); 3302
17655: WINAMP_VOLUMEDOWN', 'LongInt.SetInt( 40059); 3300

```



```

17656: WINAMP_VOLUMEUP, 'LongInt.SetInt( 40058); 3299
17657: wintrust, 'String.SetString( 'wintrust.dll; 16
17658: WM_APPCOMMAND, 'LongWord.SetUInt( $0319); 2588
17659: WM_KEYDOWN, 'LongWord.SetUInt($00000100); 178
17660: WM_KEYFIRST, 'LongWord.SetUInt($00000100); 177
17661: WM_KEYUP, 'LongWord.SetUInt($00000101); 179
17662: WM_MOUSEWHEEL, 'LongWord.SetUInt( $020A); 1633
17663: WM_MOUSEWHEEL, 'LongWord.SetUInt( $020A); 3036
17664: WM_WA_IPC, 'SetString( WM_USER); 3260
17665: WMCOPYID, 'DWORD.SetInt($AFAF); 2987
17666: WNTYPE_DRIVE, 'LongInt.SetInt( 1); 929
17667: wsErr, 'LongInt.SetInt( 2); 1141
17668: wsOk, 'LongInt.SetInt( 1); 1140
17669: xlLandscape, 'LongWord.SetUInt( $02); 2721
17670: xlPortrait, 'LongWord.SetUInt( $01); 2720
17671: XMLOption, 'Integer.SetInt( 1 shl ord( grXML)); 1733
17672: xmlSpace, 'Char.SetString( #$20 or #$9 or #$D or #$A); 3390
17673: XMLUTF8Option, 'Integer.SetInt( 1 shl ord( grXMLUTF8)); 1734
17674: xmlVersion, 'String.SetString( '1.0; 3389
17675: XRTLEqualsValue, 'LongInt.SetInt( 0); 3326
17676: XRTLGreaterValue, 'LongInt.SetInt( 1); 3327
17677: XRTLLessValue, 'LongInt.SetInt( - 1); 3325
17678: YearOnly, 'String.SetString( 'y; 3010
17679: YearsPerCentury, 'LongInt.SetInt( 100); 874
17680: YearsPerDecade, 'LongInt.SetInt( 10); 873
17681: YearsPerMillennium, 'LongInt.SetInt( 1000); 875
17682: TidCookieVersion, '( cvNetscape, cvRFC2109, cvRFC2965 );
17683: GBase64CodeTable, 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/';
17684: GXHECodeTable, '+-0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz';
17685: GUUECodeTable, '!#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNPOQRSTUVWXYZ[\]^_';
17686: Const( 'HEXDIGITS', 'String').SetString( '0123456789ABCDEF');
17687: StHexDigits = array[0..$F] of Char = '0123456789ABCDEF';
17688: Const( 'TableUU' (!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNPOQRSTUVWXYZ[\]^_');
17689: Const( 'TableXX' (+-0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz);
17690: Const( 'DIGISET', 'String'0123456789
17691: Const( 'LETTERSET', 'String'ABCDEFGHIJKLMNPOQRSTUVWXYZ'
17692: Const( 'DIGISET2', 'Tcharset').SetSet( '0123456789'
17693: Const( 'LETTERSET2', 'Tcharset').SetSet( 'ABCDEFGHIJKLMNPOQRSTUVWXYZ'
17694: Const( 'HEXSET2', 'Tcharset').SetSet( '0123456789ABCDEF');
17695: Const( 'NUMBERSET', 'Tcharset').SetSet( '0123456789');
17696: Const( 'NUMBERS', 'String'0123456789);
17697: Const( 'LETTERS', 'String'ABCDEFGHIJKLMNPOQRSTUVWXYZ);
17698: Const( 'NUMBLETTTS', 'String').SetString( '0123456789ABCDEFGHIJKLMNPOQRSTUVWXYZ);
17699: Const( 'NUMBLETTSET', 'Tcharset').SetSet( '0123456789ABCDEFGHIJKLMNPOQRSTUVWXYZ);
17700: Function CharSetToStr( const C : CharSet) : AnsiString
17701: Function StrToCharSet( const S : AnsiString) : CharSet
17702: Function MIMEBase64Decode( const S : AnsiString) : AnsiString
17703: Function MIMEBase64Encode( const S : AnsiString) : AnsiString
17704: Function UUDecode( const S : AnsiString) : AnsiString
17705: Function XXDecode( const S : AnsiString) : AnsiString
17706:
17707:
17708: //Form Template Library FTL
17709: For Form Building out of the Script, eg. 399_form_templates.txt
17710:
17711: //////////////////////////////////////
17712: //Form Template Library FTL
17713: //////////////////////////////////////
17714:
17715: For Form Building out of the Script, eg. 399_form_templates.txt
17716:
17717: 045 unit uPSI_VListView TFormListView;
17718: 263 unit uPSI_JvProfiler32; TProfReport
17719: 270 unit uPSI_ImgList; TCustomImageList
17720: 278 unit uPSI_JvImageWindow; TjvImageWindow
17721: 497 unit uPSI_DebugBox; TDebugBox
17722: 513 unit uPSI_ImageWin; TImageForm, TImageForm2
17723: 514 unit uPSI_CustomDrawTreeView; TCustomDrawForm
17724: 515 unit uPSI_GraphWin; TGraphWinForm
17725: 516 unit uPSI_actionMain; TActionForm
17726: 518 unit uPSI_CtlPanel; TAppletApplication
17727: 529 unit uPSI_MDIEdit; TEditForm
17728: 535 unit uPSI_CoolMain; {browser} TWebMainForm
17729: 538 unit uPSI_frmExportMain; TSynexportForm
17730: 585 unit uPSI_usniffer; {PortScanForm} TSniffForm
17731: 618 unit uPSI_delphi_arduino_Unit1; TLEDForm
17732: 620 unit uPSI_fplotMain; TfplotForm1
17733: 263 unit uPSI_JvProfiler32; TProfReport
17734: 660 unit uPSI_JvDBQueryParamsForm; TjvQueryParamsDialog
17735: 677 unit uPSI_OptionsFrm; TFrmOptions;
17736: 600 unit uPSI_ThreadForm; TThreadSortForm;
17737: 618 unit uPSI_delphi_arduino_Unit1; TLEDForm
17738: 620 unit uPSI_fplotMain; TfplotForm1
17739: 660 unit uPSI_JvDBQueryParamsForm; TjvQueryParamsDialog
17740: 677 unit uPSI_OptionsFrm; TFrmOptions;
17741: 718 unit uPSI_MonForm; TMonitorForm
17742: 742 unit uPSI_SimplePortMain; TPortForm1
17743: 770 unit uPSI_ovccalc; TOvcCalculator //widget
17744: 810 unit uPSI_DbExcept; TDbEngineErrorDlg

```

```

17745: 812 unit uPSI_GL_actorUnit1;                                TglActorForm1 //OpenGL Robot
17746: 846 unit uPSI_synhttp_daemon;                                TTCPHttpDaemon, TTCPHttpThrd, TPingThread
17747: 867 unit uPSI_Debug;                                          TfrmDebug
17748: 904 unit uPSI_U_HexView;                                       THexForm2
17749: 911 unit uPSI_U_Oscilloscope4; (OscfrmMain)                  TOscfrmMain
17750: 959 unit uPSI_AdSelCom;                                         TComSelectForm
17751: 1029 unit uPSI_GPUSDemo;                                         TFDemo
17752: 1031 unit uPSI_ScreenThreeDLab;                                TFormLab3D
17753: 1051 unit uPSI_panUnit1;                                         TPanForm1 //GLScene
17754: 1052 unit uPSI_DD83ul; {Arduino Tester Frm}                  TDD83f1
17755: 1059 unit uPSI_uconvMain;                                       TfconvMain //PS
17756: 1076 unit uPSI_CustomizeDlg;                                     TCustomizeDlg / TJvAddPrinterDialog;
17757:
17758:
17759: ex.:
17760:   with TEditForm.create(self) do begin
17761:     caption:= 'Template Form Tester';
17762:     FormStyle:= fsStayOnTop;
17763:     with editor do begin
17764:       Lines.LoadFromFile(Exepath+'\docs\Readme_rus_mX2.rtf');
17765:       SelStart:= 0;
17766:       Modified:= False;
17767:     end;
17768:   end;
17769:
17770:   with TThreadSortForm.create(self) do begin
17771:     showmodal;
17772:     free;
17773:   end;
17774:
17775:   with THexForm2.Create(self) do begin
17776:     ShowModal;
17777:     Free;
17778:   end;
17779:
17780:   CONST List mX4
17781:   *****
17782:
17783: ref Docu for all Type Class and Const in maXbox_types.pdf
17784: using Docu for this file is maxbox_functions_all.pdf
17785: PEP - Pascal Education Program Lib Lab for arduino and android
17786:
17787:
17788: http://stackoverflow.com/tags/pascalscript/hot
17789: http://www.jrssoftware.org/ishelp/index.php?topic=scriptfunctions
17790: http://sourceforge.net/projects/maXbox #locs:16950
17791: http://sourceforge.net/apps/mediawiki/maXbox
17792: http://www.blaisepascal.eu/
17793: http://www.blaisepascal.eu/
17794: https://github.com/maxkleiner/maXbox3.git
17795: http://www.heise.de/download/maxbox-1176464.html
17796: http://www.softpedia.com/get/Programming/Other-Programming-Files/maXbox.shtml
17797: http://www.delphiforfun.org/Programs/Utilities/index.htm
17798: http://delphi.org/2014/01/every-android-api-for-delphi/
17799: UrlGoogleQrCode='http://chart.apis.google.com/chart?chs=%d&cht=qr&chld=%s&chl=%s';
17800: Help Online
17801: http://max.kleiner.com/maxbox_functions_all.htm
17802:
17803:
17804: Web Script Examples:
17805: http://www.softwareschule.ch/examples/performer.txt';
17806: http://www.softwareschule.ch/examples/turtle.txt';
17807: http://www.softwareschule.ch/examples/SQLExport.txt';
17808: http://www.softwareschule.ch/examples/Richter.txt';
17809: http://www.softwareschule.ch/examples/checker.txt';
17810: http://www.softwareschule.ch/examples/demoscript.txt';
17811: http://www.softwareschule.ch/examples/ibzresult.txt';
17812: http://www.softwareschule.ch/examples/performindex.txt
17813: http://www.softwareschule.ch/examples/processlist.txt
17814: http://www.softwareschule.ch/examples/game.txt
17815: http://www.softwareschule.ch/examples/GEOGPS.txt
17816: http://www.softwareschule.ch/examples/turtle2.txt
17817: http://www.softwareschule.ch/examples/turtle3.txt
17818: http://www.softwareschule.ch/examples/asyncterminal.txt
17819: http://www.softwareschule.ch/examples/snowflake.txt'
17820: http://www.softwareschule.ch/examples/arduinoled.txt
17821: http://www.softwareschule.ch/examples/moon2.txt
17822: http://www.softwareschule.ch/examples/cockpit.txt
17823: http://www.softwareschule.ch/examples/tartaruga.txt
17824: http://www.softwareschule.ch/examples/surprise.txt
17825:
17826: SHA1: maXbox3.exe DEFC2444E3DA7BE541F89A5045CCC91FF6BDF933
17827: CRC32: maXbox3.exe DCAB3447
17828:
17829: Ref:
17830: 1. writeln(SHA1(Exepath+'\maxbox3.exe'))
17831: 2. shdig: TSHA1Digest;
17832: shdig:= GetSHA1OfFile(false,exepath+'\maxbox3.exe');
17833: for i:= 0 to 19 do write(ByteToHex(shdig[i]));

```

```
17834:      3. writeln(IntToHex(CRC32OfFile(exepath+'\\maxbox3.exe'),4));
17835:
17836:      EXE List in Distribution:
17837:      Verzeichnis von E:\\maxbox3\\mXGit39988\\maxbox3
17838:      11/11/2014 09:43 AM      23,444,480 maxbox3.exe
17839:      1 Datei(en),      23,444,480 Bytes
17840:
17841:      Verzeichnis von E:\\maxbox3\\mXGit39988\\maxbox3\\exercices
17842:      08/10/2011 06:35 AM      502,784 plotit3d.exe
17843:      06/03/2003 08:30 PM      710,656 UnitTests.exe
17844:      2 Datei(en),      1,213,440 Bytes
17845:
17846:      Verzeichnis von E:\\maxbox3\\mXGit39988\\maxbox3\\exercices\\chess
17847:      18/05/1997 09:34 AM      146,432 OWLCHESS.EXE
17848:      1 Datei(en),      146,432 Bytes
17849:
17850:      Verzeichnis von E:\\maxbox3\\mXGit39988\\maxbox3\\source\\Hex_Editor_MX
17851:      26/05/2004 09:48 PM      307,712 HexeditMX.exe
17852:      1 Datei(en),      307,712 Bytes
17853:
17854:
17855:      https://www.virustotal.
com/en/file/57cc6d5d730487635d32a44fefa7f56923cdb4238e6b7a5f89c4a2cd21f196cb/analysis/1426766085/
17856:
17857:      VirusTotal metadata
17858:      -----
17859:      First submission 2015-03-19 17:03:29 UTC ( 16 hours, 48 minutes ago )
17860:      Last submission 2015-03-19 08:15:56 UTC ( 1 hour, 35 minutes ago )
17861:      File names Surprise mX4
17862:      maxbox3.exe
17863:      maxbox3_9.exe
17864:
17865:      SHA256: 57cc6d5d730487635d32a44fefa7f56923cdb4238e6b7a5f89c4a2cd21f196cb
17866:      File name: maxbox3.exe
17867:      Detection ratio: 0 / 57
17868:      2015-03-14 22:15:56 UTC ( 39 minutes ago )
17869:
17870:      MD5 57d23416bd2c7b5caa618aee7a65196
17871:      SHA1 defc2444e3da7be541f89a5045ccc91ff6bdf933
17872:      SHA256 57cc6d5d730487635d32a44fefa7f56923cdb4238e6b7a5f89c4a2cd21f196cb
17873:      ssdeep 393216:B20ZCAia9DhfFC62XwQRPzNeqHynjFQv/:BPCAI0AXwMP0q2
17874:      authentihash a7b8f61633a41f5039d46b98417ec42f681522063cb3fcf1c1b5605d5e59d299
17875:      imphash 0c55892237c705323e5d0ae689ce0e
17876:
17877:      File size 23.4 MB ( 24534528 bytes )
17878:      File type Win32 EXE
17879:      Magic literal
17880:      PE32 executable for MS Windows (GUI) Intel 80386 32-bit
17881:
17882:      TrID Windows ActiveX control (36.4%)
17883:      Inno Setup installer (34.3%)
17884:      InstallShield setup (13.4%)
17885:      Win32 EXE PECompact compressed (generic) (13.0%)
17886:      Win32 Executable (generic) (1.4%)
17887:
17888:
17889:      ExifTool file metadata
17890:      -----
17891:
17892:      SpecialBuild mX4
17893:      LegalTrademarks maXbox
17894:      SubsystemVersion 4.0
17895:      Comments reduce to the max
17896:      LinkerVersion 2.25
17897:      ImageVersion 0.0
17898:      FileSubtype 0
17899:      FileVersionNumber 3.9.9.190
17900:      LanguageCode German (Swiss)
17901:      FileFlagsMask 0x003f
17902:      FileDescription maXbox Delphi VM
17903:      CharacterSet Windows, Latin1
17904:      InitializedDataSize 5276160
17905:      FileOS Win32
17906:      TimeStamp 2015:03:19 13:42:07+01:00
17907:      MIMEType application/octet-stream
17908:      LegalCopyright Free Pascal Script
17909:      FileVersion 3.9.9.190
17910:      SpeziellesBuild mX4 Compiler Engine
17911:      FileType Win32 EXE
17912:      PEType PE32
17913:      InternalName Surprise mX4
17914:      ProductVersion 3.9 Solar mX4
17915:      UninitializedDataSize 0
17916:      OSVersion 4.0
17917:      OriginalFilename maxbox3_9.exe
17918:      Subsystem Windows GUI
17919:      MachineType Intel 386 or later, and compatibles
17920:      CompanyName kleiner kommunikation
17921:
```

```

17922: CodeSize 19257344
17923: ProductName maxbox
17924: ProductVersionNumber 3.9.9.190
17925: EntryPoint 0x125efb4
17926: ObjectFileType Executable application
17927:
17928: ExifTool file metadata
17929: -----
17930: Developer metadata
17931: Publisher kleiner kommunikation
17932: Product maxbox
17933: Original name maxbox3_9.exe
17934: Internal name Sunrise mX4
17935: File version 3.9.9.190
17936: Description maxbox Delphi VM
17937: Comments reduce to the max
17938:
17939: PE header basic information
17940: Target machine Intel 386 or later processors and compatible processors
17941: Compilation timestamp 2015-03-14 12:42:07
17942: Entry Point 0x01255FB4
17943: Number of sections 10
17944:
17945: 10 PE sections
17946:
17947: Name Virtual address Virtual size Raw size Entropy MD5
17948: .text 4096 19207348 19207680 6.60 4e44461b4cbae72ebdd80070448752cc
17949: .itext 19214336 49400 49664 6.56 252603c41039d375da3b22c0be5d6028
17950: .data 19267584 267520 267776 5.59 f0ec9c9d410184cbb85e481139ad0f34
17951: .bss 19537920 381596 0 0.00 d41d8cd98f00b204e9800998ecf8427e
17952: .idata 19922944 56334 56832 5.50 d2eac744d210028acf85682639258309
17953: .edata 19980288 77 512 0.88 a86c837de081278a738f8cb5184a5eae
17954: .tls 19984384 208 0 0.00 d41d8cd98f00b204e9800998ecf8427e
17955: .rdata 19988480 24 512 0.26 64c6a40fdedd8e996642975087308771
17956: .reloc 19992576 1225480 1225728 6.76 26bc3d19ca0203abb8eea51cccc828791
17957: .rsrc 21221376 3724800 3724800 5.36 f2f30816e499ff756843a8e36da5422d
17958:
17959: PE imports
17960: PE exports CreateIncome
17961:
17962: Number of PE resources by type
17963: RT_BITMAP 839
17964: RT_STRING 307
17965: RT_RCDATA 163
17966: RT_ICON 55
17967: RT_GROUP_ICON 44
17968: RT_CURSOR 33
17969: RT_GROUP_CURSOR 28
17970: WAVE 9
17971: RT_DIALOG 2
17972: RT_HTML 2
17973: RT_MESSAGETABLE 1
17974: RT_MANIFEST 1
17975: AVI 1
17976: RT_VERSION 1
17977:
17978: PE imports
17979: [+] AVICAP32.DLL [+] AVICAP32.dll [+] GLU32.dll
17980: [+] IMAGEHLP.DLL [+] MSVCRT.DLL [+] MSVFW32.DLL
17981: [+] OpenGL32.dll [+] SHFolder.dll [+] URLMON.DLL
17982: [+] advapi32.dll [+] comctl32.dll [+] comdlg32.dll
17983: [+] gdi32.dll [+] imagehlp.dll [+] imm32.dll
17984: [+] iphlapi.dll [+] kernel32.dll [+] mpr.dll
17985: [+] msacm32.dll [+] ole32.dll [+] oleacc.dll
17986: [+] oleaut32.dll [+] oledlg.dll [+] opengl32.dll
17987: [+] shell32.dll [+] shlwapi.dll [+] user32.dll
17988: [+] usp10.dll [+] version.dll [+] winhttp.dll
17989: [+] wininet.dll [+] winmm.dll [+] winspool.drv
17990: [+] ws2_32.dll [+] wsock32.dll [+] msimg32.dll
17991:
17992: Ref:
17993: https://www.virustotal.com/en/file/57cc6d5d730487635d32a44fefaf7f56923cdb4238e6b7a5f89c4a2cd21f196cb/analysis/1426766085/
17994: End.
17995: ---- bigbitbox code_cleared_checked----
```