

Command : CRTRPGMOD
Issued by : DTECH07

Module : WSTST1
Library : KARLW
Text 'description' : *SRCMBRTXT

Source Member : WSTST1
Source File : ZWSDL2RPG
Library : KARLW
CCSID : 37
Text 'description' : Web Service: VehiclePerformance_GetByDate()
Last Change : 08/19/09 13:03:15

Generation severity level . . . : 10
Compiler options : *XREF *GEN *NOSECLVL *SHOWCPY
 *EXPDDS *EXT *NOSHOWSKP *NOSRCSTMT
 *DEBUGIO *UNREF *NOEVENTF

Debugging views : *LIST
Output : *PRINT
Optimization level : *NONE
Source listing indentation . . . : *NONE
Type conversion options : *NONE
Sort sequence : *HEX
Language identifier : *JOB RUN
Replace module : *NO
Authority : *LIBCRTAUT
Truncate numeric : *NO
Fix numeric : *NONE
Target release : V5R4M0
Allow null values : *NO
Binding directory : *NONE
Define condition names : *NONE
Enable performance collection . : *PEP
Profiling data : *NOCOL
Licensed Internal Code options . :
Generate program interface . . . : *NO
Include directory :
Preprocessor options : *NONE

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
-------------	-----------------------	----------	--------	-----------	-------------	--------	------------

Source Listing

1	/IF NOT DEFINED(PROTOTYPE_WSTST1)				000000		000001
2	*=====				000000		000002
3	* WSDL2RPG V1.7				000000		000003
4	*=====				000000		000004
5	* >>PRE-COMPILER<<				000000		000005
6	* >>CRTCMD<< CRTRPGMOD MODULE(&LI/&OB) +				000000		000006
7	* SRCFILE(&SL/&SF) +				000000		000007
8	* SRCMBR(&SM);				000000		000008
9	* >>COMPILE<<				000000		000009
10	* >>PARM<< TRUNCNBR(*NO);				000000		000010
11	* >>PARM<< DBGVIEW(*LIST);				000000		000011
12	* >>END-COMPILE<<				000000		000012
13	* >>EXECUTE<<				000000		000013
14	* >>CMD<< CRTSRVPGM +				000000		000014
15	* SRVPGM(&LI/&OB) +				000000		000015
16	* MODULE(&LI/&OB) +				000000		000016
17	* EXPORT(*ALL) +				000000		000017
18	* BNDDIR(QC2LE) +				000000		000018
19	* BNDSRVPGM(*LIBL/WSDL2RPGRT +				000000		000019
20	* *LIBL/BASICS1) +				000000		000020
21	* TEXT(' +				000000		000021
22	* Web Service: VehiclePerformance_GetByDate()+				000000		000022
23	* ') +				000000		000023
24	* DETAIL(*BASIC);				000000		000024
25	* >>END-PRE-COMPILER<<				000000		000025
26	*=====				000000		000026
27	H NOMAIN				000000		000027
28	H BNDDIR('HTTPAPI')				000000		000028
29	/ENDIF				000000		000029
30	*				000000		000030
31	* This is the procedure you call to consume the Web Service:				000000		000031
32	* -----				000000		000032
33	* Operation: tns:VehiclePerformance_GetByDate				000000		000033
34	* Returns : tns_VehiclePerformance_GetByDateResponse				000000		000034
35	*				000000		000035
36	* Include fields in your test/production programs referring to the *_t				000000		000036
37	* type templates used in this prototype, using a LIKE(*_t) or LIKEDS(*_t).				000000		000037
38	* These fields will be your request and response messages.				000000		000038
39	*				000000		000039

	* Compiler Options in Effect: *						

	* Text 'description' :						
	* Web Service: VehiclePerformance_GetByDate()						

	* Generation severity level . . . : 10 *						
	* Compiler options :						
	* *XREF *GEN *						
	* *NOSECLVL *SHOWCPY *						
	* *EXPDDS *EXT *						
	* *NOSHOWSKP *NOSRCSTMT *						
	* *DEBUGIO *NOEVENTF *						
	* Optimization level : *NONE *						
	* Source listing indentation . . . : *NONE *						

Line Number	<----- Source Specifications ----->	<----- Comments ----->	Do Num	Page Line	Change Date	Src Id	Seq Number
*	Type conversion options :	*NONE	*				
*	Sort sequence :	*HEX	*				
*	Language identifier :	*JOB RUN	*				
*	Authority :	*LIBCRTAUT	*				
*	Truncate numeric :	*NO	*				
*	Fix numeric :	*NONE	*				
*	Allow null values :	*NO	*				
*	Binding directory from Command . :	*NONE	*				
*	Binding directory from Source . :	HTTPAPI	*				
*	Library :	*LIBL	*				
*	Enable performance collection . :	*PEP	*				
*	Profiling data :	*NOCOL	*				
*	-----	*					

40	D	ReportsServiceBinding_v2_0_VehiclePerformance...			000000		000040
41	D	_GetByDate...			000000		000041
42	D	PR	likes(tns_VehiclePerformance_GetB...		000000		000042
43	D		yDateResponse_t)		000000		000043
44	D		extproc('ReportsServiceBinding_v2_0_+		000000		000044
45	D		VehiclePerformance_GetByDate')		000000		000045
46	D	i_tns_VehiclePerformance_GetByDate...			000000		000046
47	D		likes(tns_VehiclePerformance_GetB...		000000		000047
48	D		yDate_t)		000000		000048
49	D		const		000000		000049
50	D	o_msg	like(wsdl_errText_t)		000000		000050
51	*				000000		000051
52	*	Array dimensions:			000000		000052
53	D	DIM_A1 C	128		000000		000053
54	D	DIM_A2 C	128		000000		000054
55	*				000000		000055
56	*	Types:			000000		000056
57	D	s1_DTScanReportEntry_t...			000000		000057
58	D	S	128A varying		000000		000058
59	D		based(pDummy)		000000		000059
60	*				000000		000060
61	D	s1_RpgArrayOfEntry_t...			000000		000061
62	D	DS	based(pDummy)		000000		000062
63	D		qualified		000000		000063
64	D	x	10I 0		000000		000064
65	D	DTScanReportEntry...			000000		000065
66	D		like(s1_DTScanReportEntry_t)		000000		000066
67	D		dim(DIM_A1)		000000		000067
68	*				000000		000068
69	*				000000		000069
70	D	s1_RpgArrayOfTruckPCBase_t...			000000		000070
71	D	DS	based(pDummy)		000000		000071
72	D		qualified		000000		000072
73	D	x	10I 0		000000		000073
74	D	item	likes(s1_TruckPCBaseRnmd_t)		000000		000074
75	D		dim(DIM_A2)		000000		000075
76	*				000000		000076
77	*				000000		000077
78	D	s1_TruckPCBaseRnmd_t...			000000		000078
79	D	DS	based(pDummy)		000000		000079
80	D		qualified		000000		000080

Line Number	<----- Source Specifications -----><----- Comments ----->										Do Num	Page Line	Change Date	Src Id	Seq Number
81	D	TruckName	128A	varying									000000		000081
82	*												000000		000082
83	*												000000		000083
84	D	s1_VehiclePerformanceReport_t...											000000		000084
85	D	DS		based(pDummy)									000000		000085
86	D			qualified									000000		000086
87	D	Entry		likeds(s1_RpgArrayOfEntry_t)									000000		000087
88	*												000000		000088
89	*												000000		000089
90	D	tns_VehiclePerformance_GetByDate_t...											000000		000090
91	D	DS		based(pDummy)									000000		000091
92	D			qualified									000000		000092
93	D	TruckPCBase		likeds(s1_RpgArrayOfTruckPCBase_t)									000000		000093
94	D	BatchSize	10I	0									000000		000094
95	D	EndDate	Z										000000		000095
96	D	QueryByRecordDate...											000000		000096
97	D		N										000000		000097
98	D	StartDate	Z										000000		000098
99	*												000000		000099
100	*												000000		000100
101	D	tns_VehiclePerformance_GetByDateResponse_t...											000000		000101
102	D	DS		based(pDummy)									000000		000102
103	D			qualified									000000		000103
104	D	VehiclePerformance_GetByDateResult...											000000		000104
105	D			likeds(s1_VehiclePerformanceReport_t)									000000		000105
106	D	TotalRowCount	10I	0									000000		000106
107	*												000000		000107
108	*												000000		000108
109	*	Web service: Enables/disables the debug log of HTTP API											000000		000109
110	D	ReportsServiceBinding_v2_0_setHttpDebug...											000000		000110
111	D	PR											000000		000111
112	D			extproc('+									000000		000112
113	D			ReportsServiceBinding_v2_0+									000000		000113
114	D			_setHttpDebug+									000000		000114
115	D			')									000000		000115
116	D	i_status	N	const									000000		000116
117	D	i_fileName	500A	const varying options(*nopass)									000000		000117
118	*												000000		000118
119	*	Web service: Sets the proxy server to use for the connection											000000		000119
120	D	ReportsServiceBinding_v2_0_setProxy...											000000		000120
121	D	PR											000000		000121
122	D			extproc('+									000000		000122
123	D			ReportsServiceBinding_v2_0+									000000		000123
124	D			_setProxy+									000000		000124
125	D			')									000000		000125
126	D	i_server	256A	const									000000		000126
127	D	i_port	10I	0 const									000000		000127
128	*												000000		000128
129	*	Web service: Returns the endpoint of the Web service											000000		000129
130	D	ReportsServiceBinding_v2_0_getEndpoint...											000000		000130
131	D	PR	32767A	varying									000000		000131
132	D			extproc('+									000000		000132
133	D			ReportsServiceBinding_v2_0+									000000		000133
134	D			_getEndpoint+									000000		000134
135	D			')									000000		000135
136	*												000000		000136

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
137	* Web service: Sets the endpoint of the Web service				000000		000137
138	D ReportsServiceBinding_v2_0_setEndpoint...				000000		000138
139	D PR				000000		000139
140	D extproc('+				000000		000140
141	D ReportsServiceBinding_v2_0+				000000		000141
142	D _setEndpoint+				000000		000142
143	D ')				000000		000143
144	D i_url 32767A const varying options(*varsize)				000000		000144
145	* Web service: Returns the host name of the Web Service endpoint				000000		000145
146	* Web service: Returns the host name of the Web Service endpoint				000000		000146
147	D ReportsServiceBinding_v2_0_getHost...				000000		000147
148	D PR 32767A varying				000000		000148
149	D extproc('+				000000		000149
150	D ReportsServiceBinding_v2_0+				000000		000150
151	D _getHost+				000000		000151
152	D ')				000000		000152
153	* Web service: Sets the host name of the Web service endpoint				000000		000153
154	* Web service: Sets the host name of the Web service endpoint				000000		000154
155	D ReportsServiceBinding_v2_0_setHost...				000000		000155
156	D PR				000000		000156
157	D extproc('+				000000		000157
158	D ReportsServiceBinding_v2_0+				000000		000158
159	D _setHost+				000000		000159
160	D ')				000000		000160
161	D i_url 32767A const varying options(*varsize)				000000		000161
162	* Web service: Returns the port of the Web Service endpoint				000000		000162
163	* Web service: Returns the port of the Web Service endpoint				000000		000163
164	D ReportsServiceBinding_v2_0_getPort...				000000		000164
165	D PR 10I 0				000000		000165
166	D extproc('+				000000		000166
167	D ReportsServiceBinding_v2_0+				000000		000167
168	D _getPort+				000000		000168
169	D ')				000000		000169
170	* Web service: Sets the port of the Web service endpoint				000000		000170
171	* Web service: Sets the port of the Web service endpoint				000000		000171
172	D ReportsServiceBinding_v2_0_setPort...				000000		000172
173	D PR				000000		000173
174	D extproc('+				000000		000174
175	D ReportsServiceBinding_v2_0+				000000		000175
176	D _setPort+				000000		000176
177	D ')				000000		000177
178	D i_port 10I 0 const				000000		000178
179	* Registers an user callback procedure, that is called whenever				000000		000179
180	* a web service requires user authentication.				000000		000180
181	* a web service requires user authentication.				000000		000181
182	D ReportsServiceBinding_v2_0_registerLoginCallback...				000000		000182
183	D PR				000000		000183
184	D extproc('+				000000		000184
185	D ReportsServiceBinding_v2_0+				000000		000185
186	D _registerLoginCallback+				000000		000186
187	D ')				000000		000187
188	D i_pLoginProc * value procptr				000000		000188
189	* Http: Returns the last HTTP error (code) that occurred.				000000		000189
190	* Http: Returns the last HTTP error (code) that occurred.				000000		000190
191	D ReportsServiceBinding_v2_0_HttpError_getCode...				000000		000191
192	D PR like(wsdl_http_ErrorNo_t)				000000		000192

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
193	D extproc('+				000000		000193
194	D ReportsServiceBinding_v2_0+				000000		000194
195	D _HttpError_getCode+				000000		000195
196	D ')				000000		000196
197	* Http: Returns the last HTTP error (text) that occurred.				000000		000197
198	* ReportsServiceBinding_v2_0_HttpError_getText...				000000		000198
199	D PR like(wsdل_http_ErrorText_t)				000000		000199
200	D extproc('+				000000		000200
201	D ReportsServiceBinding_v2_0+				000000		000201
202	D _HttpError_getText+				000000		000202
203	D ')				000000		000203
204	D Soap: Returns the last SOAP error (code) that occurred.				000000		000204
205	* ReportsServiceBinding_v2_0_SoapError_getCode...				000000		000205
206	D PR like(wsdل_soap_faultcode_t)				000000		000206
207	D extproc('+				000000		000207
208	D ReportsServiceBinding_v2_0+				000000		000208
209	D _SoapError_getCode+				000000		000209
210	D ')				000000		000210
211	D Soap: Returns the last SOAP error (text) that occurred.				000000		000211
212	* ReportsServiceBinding_v2_0_SoapError_getText...				000000		000212
213	D PR like(wsdل_soap_faultstring_t)				000000		000213
214	D extproc('+				000000		000214
215	D ReportsServiceBinding_v2_0+				000000		000215
216	D _SoapError_getText+				000000		000216
217	D ')				000000		000217
218	D Enables/disables a workaround for Web services that are driven				000000		000218
219	D by Axis and that are affected by the AXIS-2095 bug.				000000		000219
220	D ReportsServiceBinding_v2_0_enableAxis2095...				000000		000220
221	D PR				000000		000221
222	D extproc('+				000000		000222
223	D ReportsServiceBinding_v2_0+				000000		000223
224	D _enableAxis2095+				000000		000224
225	D ')				000000		000225
226	D i_isEnabled N const				000000		000226
227	D User callback to supply login information				000000		000227
228	D Returns WSDL_SUCCESS if successful, WSDL_ERROR upon error.				000000		000228
229	D Parameter description:				000000		000229
230	D i_authType Authentication type:				000000		000230
231	D WSDL_AUTH_TYPE_HTTP - HTTP server.				000000		000231
232	D WSDL_AUTH_TYPE_PROXY - Proxy server.				000000		000232
233	D i_realm Authentication realm.				000000		000233
234	D i_numAttempts Number of login attempts.				000000		000234
235	D o_user UserName to use.				000000		000235
236	D o_password Password to use.				000000		000236
237	D ReportsServiceBinding_v2_0_supplyLoginData...				000000		000237
238	D PR 10I 0				000000		000238
239	D extproc('+				000000		000239
240	D ReportsServiceBinding_v2_0+				000000		000240
241	D _supplyLoginData+				000000		000241
242	D ')				000000		000242
243	D				000000		000243
244	D				000000		000244
245	D				000000		000245
246	D				000000		000246
247	D				000000		000247
248	D				000000		000248

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
249	D ')				000000		000249
250	D i_authType 14A const varying				000000		000250
251	D i_realm 126A const varying				000000		000251
252	D i_numAttempts 10I 0 const				000000		000252
253	D o_user 50A varying				000000		000253
254	D o_password 50A varying				000000		000254
255	*				000000		000255
256	/IF NOT DEFINED(WSDL2RPG_WS_STUB_COMMON_TYPES)				000000		000256
257	/DEFINE WSDL2RPG_WS_STUB_COMMON_TYPES				000000		000257
258	*				000000		000258
259	* Web Service error message:				000000		000259
260	D wsdl_errrText_t...				000000		000260
261	D S 128A varying based(pDummy)				000000		000261
262	*				000000		000262
263	* SOAP types:				000000		000263
264	D wsdl_http_ErrorNo_t...				000000		000264
265	D S 10I 0 based(pDummy)				000000		000265
266	*				000000		000266
267	D wsdl_http_ErrorText_t...				000000		000267
268	D S 128A varying based(pDummy)				000000		000268
269	*				000000		000269
270	D wsdl_soap_faultCode_t...				000000		000270
271	D S 256A varying based(pDummy)				000000		000271
272	*				000000		000272
273	D wsdl_soap_faultstring_t...				000000		000273
274	D S 256A varying based(pDummy)				000000		000274
275	*				000000		000275
276	* Constants:				000000		000276
277	D WSDL_HTTP_SUCCESS...				000000		000277
278	D C 1				000000		000278
279	D WSDL_HTTP_ERROR_UNEXPECTED...				000000		000279
280	D C -1				000000		000280
281	*				000000		000281
282	D WSDL_SUCCESS...				000000		000282
283	D C 1				000000		000283
284	D WSDL_ERROR...				000000		000284
285	D C 0				000000		000285
286	*				000000		000286
287	D WSDL_AUTH_TYPE_HTTP...				000000		000287
288	D C '*HTTP'				000000		000288
289	D WSDL_AUTH_TYPE_PROXY...				000000		000289
290	D C '*PROXY'				000000		000290
291	*				000000		000291
292	/ENDIF				000000		000292
293	*				000000		000293
294	/IF DEFINED(PROTOTYPE_WSTST1) LINES EXCLUDED: 1				000000		000294
295	/ENDIF				000000		000296
296	* -----				000000		000297
297	*				000000		000298
298	* HTTP header:				000000		000299
299	D cHTTP_HEADER_CONTENT_TYPE...				000000		000300
300	D C 'Content-Type'				000000		000301
301	D cHTTP_HEADER_SOAP_ACTION...				000000		000302
302	D C 'SOAPAction'				000000		000303
303	*				000000		000304

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
304	D headers_t...				000000		000305
305	D DS qualified based(pDummy)				000000		000306
306	D contentType like(wsdl_contentType_t)				000000		000307
307	D soapAction like(wsdl_soapAction_t)				000000		000308
308	*				000000		000309
309	* SOAP types:				000000		000310
310	D wsdl_soap_Fault_t...				000000		000311
311	D DS qualified based(pDummy)				000000		000312
312	D httpError likeds(wsdl_httpError_t)				000000		000313
313	D faultcode like(wsdl_soap_faultcode_t)				000000		000314
314	D faultstring like(wsdl_soap_faultstring_t)				000000		000315
315	D detail likeds(wsdl_soap_FaultDetail_t)				000000		000316
316	*				000000		000317
317	D wsdl_soap_FaultDetail_t...				000000		000318
318	D DS qualified based(pDummy)				000000		000319
319	D hostname 256A varying				000000		000320
320	*				000000		000321
321	D wsdl_httpError_t...				000000		000322
322	D DS qualified based(pDummy)				000000		000323
323	D no like(wsdl_http_ErrorNo_t)				000000		000324
324	D text like(wsdl_http_ErrorText_t)				000000		000325
325	*				000000		000326
326	/COPY QBASICS1,PBASICS1 Tools/400 Runtime Library			CPY	000000		000327

	* RPG member name : PBASICS1						1
	* External name : WSDL2RPG/QBASICS1(PBASICS1)						1
	* Last change : 08/19/09 12:46:31						1
	* Text 'description' : SrvPgm: Basic Routines - Include cpybk						1

327+	/IF NOT DEFINED(D_BASIC)	RADDAT			000000		1000100
328+	/DEFINE D_BASIC	RADDAT			000000		1000200
329+	*	RADDAT			000000		1000300
330+	* -----	RADDAT			000000		1000400
331+	* Constants	RADDAT			000000		1000500
332+	* -----	RADDAT			000000		1000600
333+	/IF NOT DEFINED(cTrue)	RADDAT			000000		1000700
334+	/DEFINE cTrue	RADDAT			000000		1000800
335+	D cTrue C const(*on)	RADDAT			000000		1000900
336+	/ENDIF	RADDAT			000000		1001000
337+	*	RADDAT			000000		1001100
338+	/IF NOT DEFINED(cFalse)	RADDAT			000000		1001200
339+	/DEFINE cFalse	RADDAT			000000		1001300
340+	D cFalse C const(*off)	RADDAT			000000		1001400
341+	/ENDIF	RADDAT			000000		1001500
342+	*	RADDAT			000000		1001600
343+	* -----	RADDAT			000000		1001700
344+	* Reference fields	RADDAT			000000		1001800
345+	* -----	RADDAT			000000		1001900
346+	*	RADDAT			000000		1002000
347+	* -----	RADDAT			000000		1002100
348+	* Prototypes	RADDAT			000000		1002200
349+	* -----	RADDAT			000000		1002300
350+	/COPY QBASICS1,PBASICS1R1 Messages			CPY	000000		1002400

	* RPG member name : PBASICS1R1						2
	* External name : WSDL2RPG/QBASICS1(PBASICS1R1)						2

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
	* Last change : 08/19/09 12:46:31						2
	* Text 'description' : SrvPgm: Basic Routines - Messages cpybk						2
351+	/IF NOT DEFINED(PBASICSR1)				000000		2000100
352+	/DEFINE PBASICSR1				000000		2000200
353+	*				000000		2000300
354+	* Reference fields				000000		2000400
355+	/COPY QBASICS1,FDMSG			CPY	000000		2000500
	* RPG member name : FDMSG						3
	* External name : WSDL2RPG/QBASICS1(FDMSG)						3
	* Last change : 08/19/09 12:46:31						3
	* Text 'description' : cpybk : Message						3
356+	/IF NOT DEFINED(msg_t)				000000		3000100
357+	/DEFINE msg_t				000000		3000200
358+	* ===== *				000000		3000300
359+	* Message structure				000000		3000400
360+	* ===== *				000000		3000500
361+	*				000000		3000600
362+D	msg_t DS qualified based(pDummy)				000000		3000700
363+	/if defined(*V5R3M0)				000000		3000800
364+D	ID likeds(msgID_t)				000000		3000900
365+	/else				000000		3001000
	LINES EXCLUDED: 1						
366+	/endif				000000		3001200
367+D	data 512A				000000		3001300
368+D	type 10A				000000		3001400
369+D	file 10A				000000		3001500
370+D	lib 10A				000000		3001600
371+	*				000000		3001700
372+	/if defined(*V5R3M0)				000000		3001800
373+D	msgID_t DS qualified based(pDummy)				000000		3001900
374+D	prefix 3A				000000		3002000
375+D	nbr 4S 0				000000		3002100
376+	/endif				000000		3002200
377+	*				000000		3002300
378+	/ENDIF				000000		3002400
379+	/COPY QBASICS1,FDRCVM0200			CPY	000000		2000600
	* RPG member name : FDRCVM0200						4
	* External name : WSDL2RPG/QBASICS1(FDRCVM0200)						4
	* Last change : 08/19/09 12:46:31						4
	* Text 'description' : cpybk : QMHRCVM API - RCVM0200						4
380+	/IF NOT DEFINED(rcvm0200_t)				000000		4000100
381+	/DEFINE rcvm0200_t				000000		4000200
382+	* ===== *				000000		4000300
383+	* QMHRCVM API - RCVM0200				000000		4000400
384+	* ===== *				000000		4000500
385+	*				000000		4000600
386+D	rcvm0200_t DS qualified based(pDummy)				000000		4000700
387+	* Bytes returned				000000		4000800
388+D	bytRet 10I 0				000000		4000900
389+	* Bytes available				000000		4001000
390+D	bytAvl 10I 0				000000		4001100

Line	----- Source Specifications -----><----- Comments ----->								Do	Page	Change	Src	Seq
Number1.....+.....2.....+.....3.....+.....4.....+.....5.....+.....6.....+.....7.....+.....8.....+.....9.....+.....10			Num	Line	Date	Id	Number					
391+	* Message Severity					RADDAT		4001200					
392+D	msgSev	10I	0			RADDAT		4001300					
393+	* Message ID					RADDAT		4001400					
394+D	msgID	7A				RADDAT		4001500					
395+	* Message Type					RADDAT		4001600					
396+D	msgType	2A				RADDAT		4001700					
397+	* Message Key					RADDAT		4001800					
398+D	msgKey	4A				RADDAT		4001900					
399+	* Message File					RADDAT		4002000					
400+D	msgF	10A				RADDAT		4002100					
401+	* Message Library Specified					RADDAT		4002200					
402+D	msgLS	10A				RADDAT		4002300					
403+	* Message Library Used					RADDAT		4002400					
404+D	msgLU	10A				RADDAT		4002500					
405+	* Sending Job					RADDAT		4002600					
406+D	sndJob	10A				RADDAT		4002700					
407+	* Sending User Profile					RADDAT		4002800					
408+D	sndUsr	10A				RADDAT		4002900					
409+	* Sending Job Number					RADDAT		4003000					
410+D	sndNbr	6A				RADDAT		4003100					
411+	* Sending Program					RADDAT		4003200					
412+D	sndPgm	12A				RADDAT		4003300					
413+	* Sending program Instr. Nbr.					RADDAT		4003400					
414+D	sndINb	4A				RADDAT		4003500					
415+	* Sending Date					RADDAT		4003600					
416+D	sndDat	7A				RADDAT		4003700					
417+	* Sending Time					RADDAT		4003800					
418+D	sndTim	6A				RADDAT		4003900					
419+	* Receiving program					RADDAT		4004000					
420+D	rcvPgm	10A				RADDAT		4004100					
421+	* Receiving Program Instr. Nbr					RADDAT		4004200					
422+D	rcvINb	4A				RADDAT		4004300					
423+	* Sending Type					RADDAT		4004400					
424+D	sndTyp	1A				RADDAT		4004500					
425+	* Receiving Type					RADDAT		4004600					
426+D	rcvTyp	1A				RADDAT		4004700					
427+	* Reserved 01					RADDAT		4004800					
428+D	reserved_01	1A				RADDAT		4004900					
429+	* Conversion Status ID Text					RADDAT		4005000					
430+D	cStIDTx	10I	0			RADDAT		4005100					
431+	* Conversion Status ID Data					RADDAT		4005200					
432+D	cStIDDt	10I	0			RADDAT		4005300					
433+	* Alert Option					RADDAT		4005400					
434+D	alrOpt	9A				RADDAT		4005500					
435+	* CCSID of msg and help					RADDAT		4005600					
436+D	ccsidMH	10I	0			RADDAT		4005700					
437+	* CCSID of replacement data					RADDAT		4005800					
438+D	ccsidRD	10I	0			RADDAT		4005900					
439+	* Length replacement data ret.					RADDAT		4006000					
440+D	lnRplDtR	10I	0			RADDAT		4006100					
441+	* Length replacement data avl.					RADDAT		4006200					
442+D	lnRplDtA	10I	0			RADDAT		4006300					
443+	* Length message ret.					RADDAT		4006400					
444+D	lnMgR	10I	0			RADDAT		4006500					
445+	* Length message avl.					RADDAT		4006600					
446+D	lnMgA	10I	0			RADDAT		4006700					

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
447+	* Length message help ret.	RADDAT			000000		4006800
448+D	lnMgHR 10I 0	RADDAT			000000		4006900
449+	* Length message help avl.	RADDAT			000000		4007000
450+D	lnMgHA 10I 0	RADDAT			000000		4007100
451+	* Value of varying length!	RADDAT			000000		4007200
452+D	value 4096A	RADDAT			000000		4007300
453+	*	RADDAT			000000		4007400
454+	/ENDIF	RADDAT			000000		4007500
455+	/COPY QBASICS1,FDRTVM0100	RADDAT		CPY	000000		2000700
-----*							
* RPG member name : FDRTVM0100							5
* External name : WSDL2RPG/QBASICS1(FDRTVM0100)							5
* Last change : 08/19/09 12:46:31							5
* Text 'description' : cpybk : QMHRTVM API - RTVM0100							5
-----*							
456+	/IF NOT DEFINED(rtv0100_t)	RADDAT			000000		5000100
457+	/DEFINE rtvm0100_t	RADDAT			000000		5000200
458+	*=====*	RADDAT			000000		5000300
459+	* QMHRTVM API - RTVM0100	RADDAT			000000		5000400
460+	*=====*	RADDAT			000000		5000500
461+	*	RADDAT			000000		5000600
462+D	rtvm0100_t DS qualified based(pDummy)	RADDAT			000000		5000700
463+	* Bytes returned	RADDAT			000000		5000800
464+D	BYTRET 9B 0	RADDAT			000000		5000900
465+	* Bytes available	RADDAT			000000		5001000
466+D	BYTAVL 9B 0	RADDAT			000000		5001100
467+	* Length of message returned	RADDAT			000000		5001200
468+D	LENMSGRET 9B 0	RADDAT			000000		5001300
469+	* Length of message available	RADDAT			000000		5001400
470+D	LENMSGAVL 9B 0	RADDAT			000000		5001500
471+	* Length of msg help returned	RADDAT			000000		5001600
472+D	LENHLPRET 9B 0	RADDAT			000000		5001700
473+	* Length of msg help available	RADDAT			000000		5001800
474+D	LENHLPVL 9B 0	RADDAT			000000		5001900
475+	* Msg text and data of varying length!	RADDAT			000000		5002000
476+D	DATA 2048A	RADDAT			000000		5002100
477+	*	RADDAT			000000		5002200
478+	/ENDIF	RADDAT			000000		5002300
479+	/COPY QBASICS1,FDRTVM0300	RADDAT		CPY	000000		2000800
-----*							
* RPG member name : FDRTVM0300							6
* External name : WSDL2RPG/QBASICS1(FDRTVM0300)							6
* Last change : 08/19/09 12:46:31							6
* Text 'description' : cpybk : QMHRTVM API - RTVM0300							6
-----*							
480+	/IF NOT DEFINED(rtv0300_t)	RADDAT			000000		6000100
481+	/DEFINE rtvm0300_t	RADDAT			000000		6000200
482+	*=====*	RADDAT			000000		6000300
483+	* QMHRTVM API - RTVM0300	RADDAT			000000		6000400
484+	*=====*	RADDAT			000000		6000500
485+	*	RADDAT			000000		6000600
486+D	rtvm0300_t DS qualified based(pDummy)	RADDAT			000000		6000700
487+	* Bytes returned	RADDAT			000000		6000800
488+D	bytRet 10I 0	RADDAT			000000		6000900
489+	* Bytes available	RADDAT			000000		6001000
490+D	bytAvl 10I 0	RADDAT			000000		6001100

Line Number	<----- Source Specifications ----->	<----- Comments ----->	Do Num	Page Line	Change Date	Src Id	Seq Number
491+	* Message severity	RADDAT			000000		6001200
492+D	msgSev 10I 0	RADDAT			000000		6001300
493+	* Alert index	RADDAT			000000		6001400
494+D	altIdx 10I 0	RADDAT			000000		6001500
495+	* Alert Option	RADDAT			000000		6001600
496+D	altOpt 9A	RADDAT			000000		6001700
497+	* Log indicator	RADDAT			000000		6001800
498+D	logInd 1A	RADDAT			000000		6001900
499+	* Message ID	RADDAT			000000		6002000
500+D	msgID 7A	RADDAT			000000		6002100
501+	* Reserved	RADDAT			000000		6002200
502+D	reserved_01 3A	RADDAT			000000		6002300
503+	* Nbr. of subst. var. formats	RADDAT			000000		6002400
504+D	numVarFmt 10I 0	RADDAT			000000		6002500
505+	* CCSID conv. stat. ind. text	RADDAT			000000		6002600
506+D	csIndTxt 10I 0	RADDAT			000000		6002700
507+	* CCSID conv. stat. ind. data	RADDAT			000000		6002800
508+D	csIndDta 10I 0	RADDAT			000000		6002900
509+	* CCSID of Text returned	RADDAT			000000		6003000
510+D	ccsidTxt 10I 0	RADDAT			000000		6003100
511+	* Offset of default reply	RADDAT			000000		6003200
512+D	ofsDftR 10I 0	RADDAT			000000		6003300
513+	* Length of default reply ret.	RADDAT			000000		6003400
514+D	lenDftRpyR 10I 0	RADDAT			000000		6003500
515+	* Length of default reply avl.	RADDAT			000000		6003600
516+D	lenDftRpyA 10I 0	RADDAT			000000		6003700
517+	* Offset of message	RADDAT			000000		6003800
518+D	ofsMsg 10I 0	RADDAT			000000		6003900
519+	* Length of message returned	RADDAT			000000		6004000
520+D	lenMsgR 10I 0	RADDAT			000000		6004100
521+	* Length of message available	RADDAT			000000		6004200
522+D	lenMsgA 10I 0	RADDAT			000000		6004300
523+	* Offset of message	RADDAT			000000		6004400
524+D	ofsMsgHlp 10I 0	RADDAT			000000		6004500
525+	* Length of message help ret.	RADDAT			000000		6004600
526+D	lenMsgHlpR 10I 0	RADDAT			000000		6004700
527+	* Length of message help avl.	RADDAT			000000		6004800
528+D	lenMsgHlpA 10I 0	RADDAT			000000		6004900
529+	* Ofs. of subst. var. formats	RADDAT			000000		6005000
530+D	ofsVarFmt 10I 0	RADDAT			000000		6005100
531+	* Length subst. var. fmt. ret.	RADDAT			000000		6005200
532+D	lenVarFmtR 10I 0	RADDAT			000000		6005300
533+	* Length subst. var. fmt. avl.	RADDAT			000000		6005400
534+D	lenVarFmtA 10I 0	RADDAT			000000		6005500
535+	* Length subst. var. element	RADDAT			000000		6005600
536+D	lenVarFmtE 10I 0	RADDAT			000000		6005700
537+	* Data	RADDAT			000000		6005800
538+D	data 4096A	RADDAT			000000		6005900
539+	* CHAR(*) Default reply	RADDAT			000000		6006000
540+	* CHAR(*) Message	RADDAT			000000		6006100
541+	* CHAR(*) Message help	RADDAT			000000		6006200
542+	* CHAR(*) Substitution variable formats	RADDAT			000000		6006300
543+	*	RADDAT			000000		6006400
544+	/ENDIF	RADDAT			000000		6006500
545+	/COPY QBASICS1,FDRTVM0400	RADDAT		CPY	000000		2000900

Line Number	<----- Source Specifications ----->	<----- Comments ----->	Do Num	Page Line	Change Date	Src Id	Seq Number
	* RPG member name : FDRTVM0400	*					7
	* External name : WSDL2RPG/QBASICS1(FDRTVM0400)	*					7
	* Last change : 08/19/09 12:46:31	*					7
	* Text 'description' : cpybk : QMHRTVM API - RTVM0400	*					7
	* -----	*					
546+	/IF NOT DEFINED(rtv0400_t)	RADDAT			000000		7000100
547+	/DEFINE rtvm0400_t	RADDAT			000000		7000200
548+	*=====	RADDAT			000000		7000300
549+	* QMHRTVM API - RTVM0400	RADDAT			000000		7000400
550+	*=====	RADDAT			000000		7000500
551+	*	RADDAT			000000		7000600
552+D	rtvm0400_t DS qualified based(pDummy)	RADDAT			000000		7000700
553+	* Bytes returned	RADDAT			000000		7000800
554+D	bytRet 10I 0	RADDAT			000000		7000900
555+	* Bytes available	RADDAT			000000		7001000
556+D	bytAvl 10I 0	RADDAT			000000		7001100
557+	* Message severity	RADDAT			000000		7001200
558+D	msgSev 10I 0	RADDAT			000000		7001300
559+	* Alert index	RADDAT			000000		7001400
560+D	altIdx 10I 0	RADDAT			000000		7001500
561+	* Alert Option	RADDAT			000000		7001600
562+D	altOpt 9A	RADDAT			000000		7001700
563+	* Log indicator	RADDAT			000000		7001800
564+D	logInd 1A	RADDAT			000000		7001900
565+	* Message ID	RADDAT			000000		7002000
566+D	msgID 7A	RADDAT			000000		7002100
567+	* Reserved	RADDAT			000000		7002200
568+D	reserved_01 3A	RADDAT			000000		7002300
569+	* Nbr. of subst. var. formats	RADDAT			000000		7002400
570+D	numVarFmt 10I 0	RADDAT			000000		7002500
571+	* CCSID conv. stat. ind. text	RADDAT			000000		7002600
572+D	csIndTxt 10I 0	RADDAT			000000		7002700
573+	* CCSID conv. stat. ind. data	RADDAT			000000		7002800
574+D	csIndDta 10I 0	RADDAT			000000		7002900
575+	* CCSID of Text returned	RADDAT			000000		7003000
576+D	ccsidTxt 10I 0	RADDAT			000000		7003100
577+	* Offset of default reply	RADDAT			000000		7003200
578+D	ofsDftR 10I 0	RADDAT			000000		7003300
579+	* Length of default reply ret.	RADDAT			000000		7003400
580+D	lenDftRpyR 10I 0	RADDAT			000000		7003500
581+	* Length of default reply avl.	RADDAT			000000		7003600
582+D	lenDftRpyA 10I 0	RADDAT			000000		7003700
583+	* Offset of message	RADDAT			000000		7003800
584+D	ofsMsg 10I 0	RADDAT			000000		7003900
585+	* Length of message returned	RADDAT			000000		7004000
586+D	lenMsgR 10I 0	RADDAT			000000		7004100
587+	* Length of message available	RADDAT			000000		7004200
588+D	lenMsgA 10I 0	RADDAT			000000		7004300
589+	* Offset of message	RADDAT			000000		7004400
590+D	ofsMsgHlp 10I 0	RADDAT			000000		7004500
591+	* Length of message help ret.	RADDAT			000000		7004600
592+D	lenMsgHlpR 10I 0	RADDAT			000000		7004700
593+	* Length of message help avl.	RADDAT			000000		7004800
594+D	lenMsgHlpA 10I 0	RADDAT			000000		7004900
595+	* Ofs. of subst. var. formats	RADDAT			000000		7005000
596+D	ofsVarFmt 10I 0	RADDAT			000000		7005100

Line Number	<----- Source Specifications -----><----- Comments ----->	Do Num	Page Line	Change Date	Src Id	Seq Number
597+	* Length subst. var. fmt. ret.			RADDAT		7005200
598+D	lenVarFmtR 10I 0			RADDAT		7005300
599+	* Length subst. var. fmt. avl.			RADDAT		7005400
600+D	lenVarFmtA 10I 0			RADDAT		7005500
601+	* Length subst. var. element			RADDAT		7005600
602+D	lenVarFmtE 10I 0			RADDAT		7005700
603+	* Reply type			RADDAT		7005800
604+D	rpyType 10A			RADDAT		7005900
605+	* Reserved			RADDAT		7006000
606+D	reserved_02 2A			RADDAT		7006100
607+	* Maximum reply length			RADDAT		7006200
608+D	maxRpyLen 10I 0			RADDAT		7006300
609+	* Maximum reply decimal positions			RADDAT		7006400
610+D	maxRpyDecPos 10I 0			RADDAT		7006500
611+	* Offset of valid reply value entries			RADDAT		7006600
612+D	ofsRpyE 10I 0			RADDAT		7006700
613+	* Number of valid reply values entries returned			RADDAT		7006800
614+D	numRpyER 10I 0			RADDAT		7006900
615+	* Length of valid reply value entries returned			RADDAT		7007000
616+D	lenRpyER 10I 0			RADDAT		7007100
617+	* Length of valid reply value entries available			RADDAT		7007200
618+D	lenRpyEA 10I 0			RADDAT		7007300
619+	* Length of valid reply value entry			RADDAT		7007400
620+D	lenRpyE 10I 0			RADDAT		7007500
621+	* Offset of special reply value entries			RADDAT		7007600
622+D	ofsSpcRpyE 10I 0			RADDAT		7007700
623+	* Number of special reply values returned			RADDAT		7007800
624+D	numSpcRpyER 10I 0			RADDAT		7007900
625+	* Length of special reply value entries returned			RADDAT		7008000
626+D	lenSpcRpyER 10I 0			RADDAT		7008100
627+	* Length of special reply value entries available			RADDAT		7008200
628+D	lenSpcRpyEA 10I 0			RADDAT		7008300
629+	* Length of special reply value entry			RADDAT		7008400
630+D	lenSpcRpyE 10I 0			RADDAT		7008500
631+	* Offset of lower range reply value			RADDAT		7008600
632+D	ofsLowRpyE 10I 0			RADDAT		7008700
633+	* Length of lower range reply value returned			RADDAT		7008800
634+D	lenLowRpyER 10I 0			RADDAT		7008900
635+	* Length of lower range reply value available			RADDAT		7009000
636+D	lenLowRpyEA 10I 0			RADDAT		7009100
637+	* Offset of upper range reply value			RADDAT		7009200
638+D	ofsUpRpyE 10I 0			RADDAT		7009300
639+	* Length of upper range reply value returned			RADDAT		7009400
640+D	ofsUpRpyER 10I 0			RADDAT		7009500
641+	* Length of upper range reply value available			RADDAT		7009600
642+D	ofsUpRpyEA 10I 0			RADDAT		7009700
643+	* Offset of relational test entry			RADDAT		7009800
644+D	ofsRelTestE 10I 0			RADDAT		7009900
645+	* Length of relational test entry returned			RADDAT		7010000
646+D	lenRelTestER 10I 0			RADDAT		7010100
647+	* Length of relational test entry available			RADDAT		7010200
648+D	lenRelTestEA 10I 0			RADDAT		7010300
649+	* Message creation date			RADDAT		7010400
650+D	crtDate 7A			RADDAT		7010500
651+	* Reserved			RADDAT		7010600
652+D	reserved_03 1A			RADDAT		7010700

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
653+	* Message creation level number	RADDAT			000000		7010800
654+D	crtLvlNbr 10I 0	RADDAT			000000		7010900
655+	* Message modification date	RADDAT			000000		7011000
656+D	modDate 7A	RADDAT			000000		7011100
657+	* Reserved	RADDAT			000000		7011200
658+D	reserved_04 1A	RADDAT			000000		7011300
659+	* Message modification level number	RADDAT			000000		7011400
660+D	modLvlNbr 10I 0	RADDAT			000000		7011500
661+	* Stored CCSID of message	RADDAT			000000		7011600
662+D	storedCCSID 10I 0	RADDAT			000000		7011700
663+	* Offset of dump list entries	RADDAT			000000		7011800
664+D	ofsDmpListE 10I 0	RADDAT			000000		7011900
665+	* Number of dump list entries returned	RADDAT			000000		7012000
666+D	numDmpListER 10I 0	RADDAT			000000		7012100
667+	* Length of dump list entries returned	RADDAT			000000		7012200
668+D	lenDmpListER 10I 0	RADDAT			000000		7012300
669+	* Length of dump list entries available	RADDAT			000000		7012400
670+D	lenDmpListEA 10I 0	RADDAT			000000		7012500
671+	* Default program name	RADDAT			000000		7012600
672+D	dftPgmName 10A	RADDAT			000000		7012700
673+	* Default program library name	RADDAT			000000		7012800
674+D	dftPgmLib 10A	RADDAT			000000		7012900
675+	* Data	RADDAT			000000		7013000
676+D	data 4096A	RADDAT			000000		7013100
677+	* CHAR(*) Default reply	RADDAT			000000		7013200
678+	* CHAR(*) Message	RADDAT			000000		7013300
679+	* CHAR(*) Message help	RADDAT			000000		7013400
680+	* CHAR(*) Substitution variable formats	RADDAT			000000		7013500
681+	* CHAR(*) Valid reply value entries	RADDAT			000000		7013600
682+	* CHAR(*) Special reply value entries	RADDAT			000000		7013700
683+	* CHAR(*) Lower range reply value	RADDAT			000000		7013800
684+	* CHAR(*) Upper range reply value	RADDAT			000000		7013900
685+	* CHAR(*) Relational test entry	RADDAT			000000		7014000
686+	* CHAR(*) Dump list entries	RADDAT			000000		7014100
687+		RADDAT			000000		7014200
688+	/ENDIF	RADDAT			000000		7014300
689+		RADDAT			000000		2001000
690+	* Constants	RADDAT			000000		2001100
691+		RADDAT			000000		2001200
692+	* Message ID special values:	RADDAT			000000		2001300
693+D	cMSG_ID_OK C const('*OK')				000000		2001400
694+D	cMSG_ID_NONE C const('')				000000		2001500
695+D	cMSG_ID_NULL C const('*NULL')	RADDAT			000000		2001600
696+		RADDAT			000000		2001700
697+	* Parameter: Stack Counter				000000		2001800
698+D	cMsg_AllInact C const(-4)	- / - /rmv RADDAT			000000		2001900
699+D	cMsg_Ext C const(-1)	snd/rcv/rmv RADDAT			000000		2002000
700+D	cMsg_Priv C const(-2)	snd/rcv/rmv RADDAT			000000		2002100
701+D	cMsg_Same C const(-3)	snd/rcv/rmv RADDAT			000000		2002200
702+		RADDAT			000000		2002300
703+	* Parameter: Procedure	RADDAT			000000		2002400
704+D	cMsg_ToMyself C const('*')	snd/rcv/rmv RADDAT			000000		2002500
705+D	cMsg_ToPgmBdy C const('*PGMBDY')	snd/ - / - RADDAT			000000		2002600
706+D	cMsg_ToCtlBdy C const('*CTLBDY')	snd/ - / - RADDAT			000000		2002700
707+D	cMsg_ToPgm C const('*PGMNAME')	snd/ - / - RADDAT			000000		2002800
708+		RADDAT			000000		2002900

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
709+	* Parameter: Message Type				000000		2003000
710+D	cMsg_Any C	const(' *ANY') - /rcv/ -			RADDAT 000000		2003100
711+D	cMsg_Comp C	const(' *COMP') snd/rcv/ -			RADDAT 000000		2003200
712+D	cMsg_Copy C	const(' *COPY') - /rcv/ -			RADDAT 000000		2003300
713+D	cMsg_Diag C	const(' *DIAG') snd/rcv/ -			RADDAT 000000		2003400
714+D	cMsg_Escape C	const(' *ESCAPE') snd/rcv/ -			RADDAT 000000		2003500
715+D	cMsg_Escp C	const(' *EXCP') - /rcv/ -			RADDAT 000000		2003600
716+D	cMsg_First C	const(' *FIRST') - /rcv/ -			RADDAT 000000		2003700
717+D	cMsg_Info C	const(' *INFO') snd/rcv/ -			RADDAT 000000		2003800
718+D	cMsg_Inq C	const(' *INQ') snd/ - / -			RADDAT 000000		2003900
719+D	cMsg_Last C	const(' *LAST') --/rcv/ -			RADDAT 000000		2004000
720+D	cMsg_Next C	const(' *NEXT') --/rcv/ -			RADDAT 000000		2004100
721+D	cMsg_Notify C	const(' *NOTIFY') snd/rcv/ -			RADDAT 000000		2004200
722+D	cMsg_Previous C	const(' *PRV') --/rcv/ -			RADDAT 000000		2004300
723+D	cMsg_Rpy C	const(' *RPY') --/rcv/ -			RADDAT 000000		2004400
724+D	cMsg_Rqs C	const(' *RQS') snd/rcv/ -			RADDAT 000000		2004500
725+D	cMsg_Status C	const(' *STATUS') snd/ - / -			RADDAT 000000		2004600
726+	* Parameter: Action				RADDAT 000000		2004700
727+	* Parameter: Action				RADDAT 000000		2004800
728+D	cMsg_Act_Old C	const(' *OLD') - /rcv/ -			RADDAT 000000		2004900
729+D	cMsg_Act_Rmv C	const(' *REMOVE') - /rcv/ -			RADDAT 000000		2005000
730+D	cMsg_Act_Same C	const(' *SAME') - /rcv/ -			RADDAT 000000		2005100
731+	* Parameter: Message to remove				RADDAT 000000		2005200
732+	* Parameter: Message to remove				RADDAT 000000		2005300
733+D	cMsg_Rmv_All C	const(' *ALL') - / - /rmv			RADDAT 000000		2005400
734+D	cMsg_Rmv_ByKey C	const(' *BYKEY') - / - /rmv			RADDAT 000000		2005500
735+D	cMsg_Rmv_New C	const(' *NEW') - / - /rmv			RADDAT 000000		2005600
736+D	cMsg_Rmv_Old C	const(' *OLD') - / - /rmv			RADDAT 000000		2005700
737+D	cMsg_Rmv_Scope C	const(' *SCOPE') - / - /rmv			RADDAT 000000		2005800
738+	* Retrieve message help text				RADDAT 000000		2005900
739+	* Retrieve message help text				RADDAT 000000		2006000
740+D	cMsg_RtnCtrlChar_Yes...				RADDAT 000000		2006100
741+D	C	const(' *YES')			RADDAT 000000		2006200
742+D	cMsg_RtnCtrlChar_No...				RADDAT 000000		2006300
743+D	C	const(' *NO')			RADDAT 000000		2006400
744+	* Sets the application message file.				RADDAT 000000		2006500
745+	* Sets the application message file.				RADDAT 000000		2006600
746+D	p_setAppMsgFile...				RADDAT 000000		2006700
747+D	PR				RADDAT 000000		2006800
748+D		extproc('BASICS1R1+			RADDAT 000000		2006900
749+D		_p_setAppMsgFile+			RADDAT 000000		2007000
750+D		')			RADDAT 000000		2007100
751+D	i_file 10A	const options(*nopass)			RADDAT 000000		2007200
752+D	i_lib 10A	const options(*nopass)			RADDAT 000000		2007300
753+	* Creates a new message.				RADDAT 000000		2007400
754+	* Creates a new message.				RADDAT 000000		2007500
755+D	f_newMsg...				RADDAT 000000		2007600
756+D	PR	likes(msg_t)			RADDAT 000000		2007700
757+D		extproc('BASICS1R1+			RADDAT 000000		2007800
758+D		_f_newMsg+			RADDAT 000000		2007900
759+D		')			RADDAT 000000		2008000
760+D	i_ID	const like(msg_t.id)			RADDAT 000000		2008100
761+D		options(*omit: *nopass)			RADDAT 000000		2008200
762+D	i_data 512A	const varying			RADDAT 000000		2008300
763+D		options(*omit: *nopass)			RADDAT 000000		2008400
764+D	i_type	const like(msg_t.type)			RADDAT 000000		2008500

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
765+D		options(*omit: *nopass)			000000		2008600
766+D	i_file	const like(msg_t.file)			000000		2008700
767+D		options(*omit: *nopass)			000000		2008800
768+D	i_lib	const like(msg_t.lib)			000000		2008900
769+D		options(*omit: *nopass)			000000		2009000
770+ *					000000		2009100
771+ *	Sends a program message.				000000		2009200
772+D	f_sndPgmMsg...				000000		2009300
773+D	PR 4A				000000		2009400
774+D		extproc('BASICS1R1+			000000		2009500
775+D		_f_sndPgmMsg+			000000		2009600
776+D		')			000000		2009700
777+D	i_msg	const likeds(msg_t)			000000		2009800
778+D	i_toStkC	10I 0 const options(*nopass: *omit)			000000		2009900
779+D	i_toProc	256A const options(*nopass: *omit)			000000		2010000
780+D	i_toMod	10A const options(*nopass: *omit)			000000		2010100
781+D	i_toPgm	10A const options(*nopass: *omit)			000000		2010200
782+D	o_msg	likeds(msg_t)			000000		2010300
783+D		options(*nopass: *omit)			000000		2010400
784+ *					000000		2010500
785+ *	Receives a program message.				000000		2010600
786+D	f_rcvPgmMsg...				000000		2010700
787+D	PR	likeds(rcvm0200_t)			000000		2010800
788+D		extproc('BASICS1R1+			000000		2010900
789+D		_f_rcvPgmMsg+			000000		2011000
790+D		')			000000		2011100
791+D	i_msgType	10A const			000000		2011200
792+D	i_msgKey	4A const options(*nopass: *omit)			000000		2011300
793+D	i_action	10A const options(*nopass: *omit)			000000		2011400
794+D	i_wait	10I 0 const options(*nopass: *omit)			000000		2011500
795+D	i_fromStkC	10I 0 const options(*nopass: *omit)			000000		2011600
796+D	i_fromProc	256A const options(*nopass: *omit)			000000		2011700
797+D	i_fromMod	10A const options(*nopass: *omit)			000000		2011800
798+D	i_fromPgm	10A const options(*nopass: *omit)			000000		2011900
799+D	o_msg	likeds(msg_t)			000000		2012000
800+D		options(*nopass: *omit)			000000		2012100
801+ *					000000		2012200
802+ *	Removes a program message.				000000		2012300
803+D	p_rmvPgmMsg...				000000		2012400
804+D	PR				000000		2012500
805+D		extproc('BASICS1R1+			000000		2012600
806+D		_p_rmvPgmMsg+			000000		2012700
807+D		')			000000		2012800
808+D	i_msgToRemove	10A const			000000		2012900
809+D	i_msgKey	4A const options(*nopass: *omit)			000000		2013000
810+D	i_fromStkC	10I 0 const options(*nopass: *omit)			000000		2013100
811+D	i_fromProc	256A const options(*nopass: *omit)			000000		2013200
812+D	i_fromMod	10A const options(*nopass: *omit)			000000		2013300
813+D	i_fromPgm	10A const options(*nopass: *omit)			000000		2013400
814+D	o_msg	likeds(msg_t)			000000		2013500
815+D		options(*nopass: *omit)			000000		2013600
816+ *					000000		2013700
817+ *	Retrieves the message text of a given message.				000000		2013800
818+ *	Replaces substitution variables with message data.				000000		2013900
819+D	f_rtvMsgText...				000000		2014000
820+D	PR 4096A	varying			000000		2014100

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
821+D	extproc('BASICS1R1+	RADDAT			000000		2014200
822+D	_f_rtvMsgText+	RADDAT			000000		2014300
823+D)	RADDAT			000000		2014400
824+D i_msg	const likeds(msg_t)	RADDAT			000000		2014500
825+D o_msg	likeds(msg_t)	RADDAT			000000		2014600
826+D	options(*nopass: *omit)	RADDAT			000000		2014700
827+ *		RADDAT			000000		2014800
828+ *	Retrieves the message help text of a given message.	RADDAT			000000		2014900
829+ *	Replaces substitution variables with message data.				000000		2015000
830+D f_rtvMsgHelpText...		RADDAT			000000		2015100
831+D PR 4096A	varying	RADDAT			000000		2015200
832+D	extproc('BASICS1R1+	RADDAT			000000		2015300
833+D	_f_rtvMsgHelpText+	RADDAT			000000		2015400
834+D)	RADDAT			000000		2015500
835+D i_msg	const likeds(msg_t)	RADDAT			000000		2015600
836+D i_rtnCtrlChr 10A	const options(*nopass: *omit)	RADDAT			000000		2015700
837+D o_msg	likeds(msg_t)	RADDAT			000000		2015800
838+D	options(*nopass: *omit)	RADDAT			000000		2015900
839+ *		RADDAT			000000		2016000
840+ *	Retrieves the message text of a given message.	RADDAT			000000		2016100
841+ *	Does not replace substitution variables with message data.				000000		2016200
842+D f_rtvMsgTextRaw...					000000		2016300
843+D PR 4096A	varying	RADDAT			000000		2016400
844+D	extproc('BASICS1R1+	RADDAT			000000		2016500
845+D	_f_rtvMsgTextRaw+				000000		2016600
846+D)	RADDAT			000000		2016700
847+D i_msg	const likeds(msg_t)	RADDAT			000000		2016800
848+D o_msg	likeds(msg_t)	RADDAT			000000		2016900
849+D	options(*nopass: *omit)	RADDAT			000000		2017000
850+ *		RADDAT			000000		2017100
851+ *	Retrieves the message help text of a given message.	RADDAT			000000		2017200
852+ *	Does not replace substitution variables with message data.				000000		2017300
853+D f_rtvMsgHelpTextRaw...					000000		2017400
854+D PR 4096A	varying	RADDAT			000000		2017500
855+D	extproc('BASICS1R1+	RADDAT			000000		2017600
856+D	_f_rtvMsgHelpTextRaw+				000000		2017700
857+D)	RADDAT			000000		2017800
858+D i_msg	const likeds(msg_t)	RADDAT			000000		2017900
859+D i_rtnCtrlChr 10A	const options(*nopass: *omit)	RADDAT			000000		2018000
860+D o_msg	likeds(msg_t)	RADDAT			000000		2018100
861+D	options(*nopass: *omit)	RADDAT			000000		2018200
862+ *		RADDAT			000000		2018300
863+ *	Converts a received message of format RCVM0200 to a message.	RADDAT			000000		2018400
864+D f_cvtRcvMsgToMsg...		RADDAT			000000		2018500
865+D PR	likeds(msg_t)	RADDAT			000000		2018600
866+D	extproc('BASICS1R1+	RADDAT			000000		2018700
867+D	_f_cvtRcvMsgToMsg+				000000		2018800
868+D)	RADDAT			000000		2018900
869+D i_rcvm0200	const likeds(rcvm0200_t)				000000		2019000
870+D i_type	const like(msg_t.type)	RADDAT			000000		2019100
871+D	options(*omit: *nopass)	RADDAT			000000		2019200
872+ *		RADDAT			000000		2019300
873+ *	Receives a program message and returns a message structure.	RADDAT			000000		2019400
874+D f_rcvPgmMsgObj...		RADDAT			000000		2019500
875+D PR	likeds(msg_t)	RADDAT			000000		2019600
876+D	extproc('BASICS1R1+	RADDAT			000000		2019700

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
877+D	f_rcvPgmMsgObj+	RADDAT			000000		2019800
878+D)	RADDAT			000000		2019900
879+D	i_msgType 10A const	RADDAT			000000		2020000
880+D	i_msgKey 4A const options(*nopass: *omit)	RADDAT			000000		2020100
881+D	i_action 10A const options(*nopass: *omit)	RADDAT			000000		2020200
882+D	i_wait 10I 0 const options(*nopass: *omit)	RADDAT			000000		2020300
883+D	i_fromStkC 10I 0 const options(*nopass: *omit)	RADDAT			000000		2020400
884+D	i_fromProc 256A const options(*nopass: *omit)	RADDAT			000000		2020500
885+D	i_fromMod 10A const options(*nopass: *omit)	RADDAT			000000		2020600
886+D	i_fromPgm 10A const options(*nopass: *omit)	RADDAT			000000		2020700
887+D	o_msg likeds(msg_t)	RADDAT			000000		2020800
888+D	options(*nopass: *omit)	RADDAT			000000		2020900
889+ *		RADDAT			000000		2021000
890+ *		RADDAT			000000		2021100
891+ *	Imported prototypes	RADDAT			000000		2021200
892+ *		RADDAT			000000		2021300
893+ /COPY QBASICS1,PBASICS1R2	API Error Handling	RADDAT		CPY	000000		2021400
* -----*							
* RPG member name :	PBASICS1R2	*					8
* External name :	WSDL2RPG/QBASICS1(PBASICS1R2)	*					8
* Last change :	08/19/09 12:46:31	*					8
* Text 'description' :	SrvPgm: Basic Routines - API Error Handling cpybk	*					8
* -----*							
894+ /IF NOT DEFINED(BASICS1R2)					000000		8000100
895+ /DEFINE BASICS1R2					000000		8000200
896+ *		RADDAT			000000		8000300
897+ * Reference fields		RADDAT			000000		8000400
898+ /COPY QBASICS1,FDERRCODE		RADDAT		CPY	000000		8000500
* -----*							
* RPG member name :	FDERRCODE	*					9
* External name :	WSDL2RPG/QBASICS1(FDERRCODE)	*					9
* Last change :	08/19/09 12:46:30	*					9
* Text 'description' :	cpybk : API error code	*					9
* -----*							
899+ /IF NOT DEFINED(errCode_t)		RADDAT			000000		9000100
900+ /DEFINE errCode_t		RADDAT			000000		9000200
901+ *=====*		RADDAT			000000		9000300
902+ * API error code		RADDAT			000000		9000400
903+ *=====*		RADDAT			000000		9000500
904+D errCode_t DS qualified based(pDummy)		RADDAT			000000		9000600
905+D bytPrv 10I 0		RADDAT			000000		9000700
906+D bytAvl 10I 0		RADDAT			000000		9000800
907+D excID 7A		RADDAT			000000		9000900
908+D reserved_1 1A		RADDAT			000000		9001000
909+D excDta 256A		RADDAT			000000		9001100
910+ *		RADDAT			000000		9001200
911+ /ENDIF		RADDAT			000000		9001300
912+ *		RADDAT			000000		8000600
913+ * Creates a new API error code.		RADDAT			000000		8000700
914+D f_newApiErrCode...		RADDAT			000000		8000800
915+D PR	likeds(errCode_t)	RADDAT			000000		8000900
916+D	extproc('BASICS1R2+	RADDAT			000000		8001000
917+D	f_newApiErrCode+	RADDAT			000000		8001100
918+D)	RADDAT			000000		8001200
919+D i_monMsg N const options(*nopass)		RADDAT			000000		8001300
920+ *		RADDAT			000000		8001400

Line Number	<----- Source Specifications -----><----- Comments ----->	Do Num	Page Line	Change Date	Src Id	Seq Number
1015+	/IF NOT DEFINED(BASICS1R2) LINES EXCLUDED: 64			000000		12000100
1016+	/ENDIF			000000		12006600
1017+	/COPY QBASICS1,PBASICS1R5 Objects		CPY	000000		11004000
	* * RPG member name : PBASICS1R5 * External name : WSDL2RPG/QBASICS1(PBASICS1R5) * Last change : 08/19/09 12:46:32 * Text 'description' : SrvPgm: Basic Routines - Objects cpybk *					13 13 13 13
1018+	/IF NOT DEFINED(BASICS1R5)			000000		13000100
1019+	/DEFINE BASICS1R5			000000		13000200
1020+	*			000000		13000300
1021+	* Reference fields			000000		13000400
1022+	/COPY QBASICS1,FDQOBJ * * RPG member name : FDQOBJ * External name : WSDL2RPG/QBASICS1(FDQOBJ) * Last change : 08/19/09 12:46:31 * Text 'description' : cpybk : Qualified object name *		CPY	000000		13000500 14 14 14 14
1023+	/IF NOT DEFINED(qObj_t)			000000		14000100
1024+	/DEFINE qObj_t			000000		14000200
1025+	* ===== *			000000		14000300
1026+	* Qualified object name *			000000		14000400
1027+	* ===== *			000000		14000500
1028+	*			000000		14000600
1029+D	qObj_t DS qualified based(pDummy)			000000		14000700
1030+D	name 10A			000000		14000800
1031+D	lib 10A			000000		14000900
1032+	*			000000		14001000
1033+	/ENDIF			000000		14001100
1034+	/COPY QBASICS1,FDOBJD0100 * * RPG member name : FDOBJD0100 * External name : WSDL2RPG/QBASICS1(FDOBJD0100) * Last change : 08/19/09 12:46:31 * Text 'description' : cpybk : QUSROBJD API - OBJD0100 *		CPY	000000		13000600 15 15 15 15
1035+	/IF NOT DEFINED(objd0100_t)			000000		15000100
1036+	/DEFINE objd0100_t			000000		15000200
1037+	* ===== *			000000		15000300
1038+	* QUSROBJD API - OBJD0300 *			000000		15000400
1039+	* ===== *			000000		15000500
1040+	*			000000		15000600
1041+D	objd0100_t DS qualified based(pDummy)			000000		15000700
1042+D	bytRet 10I 0			000000		15000800
1043+D	bytAvl 10I 0			000000		15000900
1044+D	name 10A			000000		15001000
1045+D	lib 10A			000000		15001100
1046+D	type 10A			000000		15001200
1047+D	rtnLib 10A			000000		15001300
1048+D	auxStgP 10I 0			000000		15001400
1049+D	owner 10A			000000		15001500
1050+D	domain 2A			000000		15001600

Line Number	<----- Source Specifications ----->	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
1051+D	crtDatTim 13A	RADDAT			000000		15001700
1052+D	chgDatTim 13A	RADDAT			000000		15001800
1053+	*	RADDAT			000000		15001900
1054+	/ENDIF	RADDAT			000000		15002000
1055+	/COPY QBASICS1,FDMBRD0100	RADDAT		CPY	000000		13000700

	* RPG member name : FDMBRD0100	*					16
	* External name : WSDL2RPG/QBASICS1(FDMBRD0100)	*					16
	* Last change : 08/19/09 12:46:31	*					16
	* Text 'description' : cpybk : QUSRMBRD API - MBRD0100	*					16

1056+	/IF NOT DEFINED(mbrd0100_t)	RADDAT			000000		16000100
1057+	/DEFINE mbrd0100_t	RADDAT			000000		16000200
1058+	*=====*	RADDAT			000000		16000300
1059+	* QUSRMBRD API - MBRD0100	RADDAT			000000		16000400
1060+	*=====*	RADDAT			000000		16000500
1061+D	mbrd0100_t DS qualified based(pDummy)	RADDAT			000000		16000600
1062+D	bytePrv 10I 0	RADDAT			000000		16000700
1063+D	byteAvl 10I 0	RADDAT			000000		16000800
1064+D	filNme 10A	RADDAT			000000		16000900
1065+D	fillLib 10A	RADDAT			000000		16001000
1066+D	mbr 10A	RADDAT			000000		16001100
1067+D	attr 10A	RADDAT			000000		16001200
1068+D	srcTyp 10A	RADDAT			000000		16001300
1069+D	crtDatTim 13A	RADDAT			000000		16001400
1070+D	chgDatTim 13A	RADDAT			000000		16001500
1071+D	text 50A	RADDAT			000000		16001600
1072+D	isSrcFile 1A	RADDAT			000000		16001700
1073+	*	RADDAT			000000		16001800
1074+	/ENDIF	RADDAT			000000		16001900
1075+	/COPY QBASICS1,FDMBRD0200	RADDAT		CPY	000000		13000800

	* RPG member name : FDMBRD0200	*					17
	* External name : WSDL2RPG/QBASICS1(FDMBRD0200)	*					17
	* Last change : 08/19/09 12:46:31	*					17
	* Text 'description' : cpybk : QUSRMBRD API - MBRD0200	*					17

1076+	/IF NOT DEFINED(mbrd0200_t)	RADDAT			000000		17000100
1077+	/DEFINE mbrd0200_t	RADDAT			000000		17000200
1078+	*=====*	RADDAT			000000		17000300
1079+	* QUSRMBRD API - MBRD0200	RADDAT			000000		17000400
1080+	*=====*	RADDAT			000000		17000500
1081+D	mbrd0200_t DS qualified based(pDummy)	RADDAT			000000		17000600
1082+D	bytePrv 10I 0	RADDAT			000000		17000700
1083+D	byteAvl 10I 0	RADDAT			000000		17000800
1084+D	filNme 10A	RADDAT			000000		17000900
1085+D	fillLib 10A	RADDAT			000000		17001000
1086+D	mbr 10A	RADDAT			000000		17001100
1087+D	attr 10A	RADDAT			000000		17001200
1088+D	srcTyp 10A	RADDAT			000000		17001300
1089+D	crtDatTim 13A	RADDAT			000000		17001400
1090+D	chgDatTim 13A	RADDAT			000000		17001500
1091+D	text 50A	RADDAT			000000		17001600
1092+D	isSrcFile 1A	RADDAT			000000		17001700
1093+	* - - - - -	RADDAT			000000		17001800
1094+D	rmtFile 1A	RADDAT			000000		17001900

Line	<----- Source Specifications -----><----- Comments ----->										Do	Page	Change	Src	Seq
Number	1	2	3	4	5	6	7	8	9	10	Num	Line	Date	Id	Number
1095+D	lglFile			1A						RADDAT			000000		17002000
1096+D	odpSharing			1A						RADDAT			000000		17002100
1097+D	reserved_1			2A						RADDAT			000000		17002200
1098+D	numRcdAllMbr		10I	0						RADDAT			000000		17002300
1099+D	numDltRcd		10I	0						RADDAT			000000		17002400
1100+D	dtaSpcSize		10I	0						RADDAT			000000		17002500
1101+D	accPthSize		10I	0						RADDAT			000000		17002600
1102+D	numPhyMbrs		10I	0						RADDAT			000000		17002700
1103+D	mbrChgDatTim		13A							RADDAT			000000		17002800
1104+D	mbrSavDatTim		13A							RADDAT			000000		17002900
1105+D	mbrRstDatTim		13A							RADDAT			000000		17003000
1106+D	mbrExpDate		7A							RADDAT			000000		17003100
1107+D	reserved_2		6A							RADDAT			000000		17003200
1108+D	numDaysUsed		10I	0						RADDAT			000000		17003300
1109+D	dateLastUsed		7A							RADDAT			000000		17003400
1110+D	useResetDate		7A							RADDAT			000000		17003500
1111+D	reserved_3		2A							RADDAT			000000		17003600
1112+D	dtaSpcSizeMult		10I	0						RADDAT			000000		17003700
1113+D	accPthSizeMult		10I	0						RADDAT			000000		17003800
1114+D	mbrTextCCSID		10I	0						RADDAT			000000		17003900
1115+D	ofsAddInf		10I	0						RADDAT			000000		17004000
1116+D	lenAddInf		10I	0						RADDAT			000000		17004100
1117+D	numRcdAllMbrU		10U	0						RADDAT			000000		17004200
1118+D	numDltRcdU		10U	0						RADDAT			000000		17004300
1119+D	reserved_4		6A							RADDAT			000000		17004400
1120+ *										RADDAT			000000		17004500
1121+ /ENDIF										RADDAT			000000		17004600
1122+ *										RADDAT			000000		13000900
1123+D	qProc_t	DS		qualified		based(pDummy)				RADDAT			000000		13001000
1124+D	name		256A	varying						RADDAT			000000		13001100
1125+D	module		10A							RADDAT			000000		13001200
1126+D	qPgm			likeds(qObj_t)						RADDAT			000000		13001300
1127+ *										RADDAT			000000		13001400
1128+D	actSrvPgm_t	DS		qualified		based(pDummy)				RADDAT			000000		13001500
1129+D	procptr		*	procptr						RADDAT			000000		13001600
1130+D	actMark		10I	0						RADDAT			000000		13001700
1131+D	name		10A							RADDAT			000000		13001800
1132+D	lib		10A							RADDAT			000000		13001900
1133+ *										RADDAT			000000		13002000
1134+D	actProc_t	DS		qualified		based(pDummy)				RADDAT			000000		13002100
1135+D	procptr		*	procptr						RADDAT			000000		13002200
1136+D	actMark		10I	0						RADDAT			000000		13002300
1137+D	name		256A	varying						RADDAT			000000		13002400
1138+D	srvPgm		10A							RADDAT			000000		13002500
1139+D	lib		10A							RADDAT			000000		13002600
1140+ *										RADDAT			000000		13002700
1141+ *	Checks object existence.												000000		13002800
1142+D	f_exist...									RADDAT			000000		13002900
1143+D		PR	N							RADDAT			000000		13003000
1144+D				extproc('BASICS1R5+						RADDAT			000000		13003100
1145+D				_f_exist+						RADDAT			000000		13003200
1146+D				')						RADDAT			000000		13003300
1147+D	i_obj		10A	const						RADDAT			000000		13003400
1148+D	i_lib		10A	const	options(*omit)					RADDAT			000000		13003500
1149+D	i_type		10A	const						RADDAT			000000		13003600
1150+D	i_mbr		10A	const	options(*nopass)					RADDAT			000000		13003700

Line Number	<----- Source Specifications ----->	<----- Comments ----->	Do Num	Page Line	Change Date	Src Id	Seq Number
1151+	* RADDAT				000000		13003800
1152+	* Retrieve object description. RADDAT				000000		13003900
1153+D	f_rtvObjD... RADDAT				000000		13004000
1154+D	PR 1024A RADDAT				000000		13004100
1155+D	extproc('BASICS1R5+ RADDAT				000000		13004200
1156+D	_f_rtvObjD+ RADDAT				000000		13004300
1157+D	') RADDAT				000000		13004400
1158+D	i_obj 10A const RADDAT				000000		13004500
1159+D	i_lib 10A const options(*omit) RADDAT				000000		13004600
1160+D	i_type 10A const RADDAT				000000		13004700
1161+D	i_format 8A const options(*nopass: *omit) RADDAT				000000		13004800
1162+D	o_msg liked(msg_t) RADDAT				000000		13004900
1163+D	options(*nopass: *omit) RADDAT				000000		13005000
1164+	* RADDAT				000000		13005100
1165+	* Retrieve member description. RADDAT				000000		13005200
1166+D	f_rtvMbrD... RADDAT				000000		13005300
1167+D	PR 1024A RADDAT				000000		13005400
1168+D	extproc('BASICS1R5+ RADDAT				000000		13005500
1169+D	_f_rtvMbrD+ RADDAT				000000		13005600
1170+D	') RADDAT				000000		13005700
1171+D	i_file 10A const RADDAT				000000		13005800
1172+D	i_lib 10A const options(*omit) RADDAT				000000		13005900
1173+D	i_mbr 10A const RADDAT				000000		13006000
1174+D	i_format 8A const options(*nopass: *omit) RADDAT				000000		13006100
1175+D	o_msg liked(msg_t) RADDAT				000000		13006200
1176+D	options(*nopass: *omit) RADDAT				000000		13006300
1177+	* RADDAT				000000		13006400
1178+	* Produces a temporary object name. RADDAT				000000		13006500
1179+D	f_getTmpName... RADDAT				000000		13006600
1180+D	PR like(qObj_t) RADDAT				000000		13006700
1181+D	extproc('BASICS1R5+ RADDAT				000000		13006800
1182+D	_f_getTmpName+ RADDAT				000000		13006900
1183+D	') RADDAT				000000		13007000
1184+	* RADDAT				000000		13007100
1185+	* Produces a unique object name. (experimental) RADDAT				000000		13007200
1186+D	f_getUniqueName... RADDAT				000000		13007300
1187+D	PR like(qObj_t) RADDAT				000000		13007400
1188+D	extproc('BASICS1R5+ RADDAT				000000		13007500
1189+D	_f_getUniqueName+ RADDAT				000000		13007600
1190+D	') RADDAT				000000		13007700
1191+D	i_type 10A const RADDAT				000000		13007800
1192+D	i_lib 10A const RADDAT				000000		13007900
1193+	* RADDAT				000000		13008000
1194+	* Creates a user space. RADDAT				000000		13008100
1195+D	f_crtUsrSpc... RADDAT				000000		13008200
1196+D	PR N RADDAT				000000		13008300
1197+D	extproc('BASICS1R5+ RADDAT				000000		13008400
1198+D	_f_crtUsrSpc+ RADDAT				000000		13008500
1199+D	') RADDAT				000000		13008600
1200+D	i_usrSpc 10A const RADDAT				000000		13008700
1201+D	i_lib 10A const RADDAT				000000		13008800
1202+D	i_text 50A const options(*omit: *nopass) RADDAT				000000		13008900
1203+D	i_size 10U 0 const options(*omit: *nopass) RADDAT				000000		13009000
1204+D	i_extAtr 10A const options(*omit: *nopass) RADDAT				000000		13009100
1205+D	i_pubAuth 10A const options(*omit: *nopass) RADDAT				000000		13009200
1206+D	o_msg liked(msg_t) RADDAT				000000		13009300

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
1207+D		options(*nopass: *omit)			000000		13009400
1208+ *					000000		13009500
1209+ *	Retrieves a pointer to the content of a user space.				000000		13009600
1210+D f_rtvUsrSpcPtr...					000000		13009700
1211+D PR *					000000		13009800
1212+D	extproc('BASICS1R5+				000000		13009900
1213+D	_f_rtvUsrSpcPtr+				000000		13010000
1214+D	')				000000		13010100
1215+D i_usrSpc	10A const				000000		13010200
1216+D i_lib	10A const options(*nopass: *omit)				000000		13010300
1217+D o_msg	likes(msg_t)				000000		13010400
1218+D	options(*nopass: *omit)				000000		13010500
1219+ *					000000		13010600
1220+ *	Returns the current size of a given user space.				000000		13010700
1221+D f_getUsrSpcSize...					000000		13010800
1222+D PR	10I 0				000000		13010900
1223+D	extproc('BASICS1R5+				000000		13011000
1224+D	_f_getUsrSpcSize+				000000		13011100
1225+D	')				000000		13011200
1226+D i_usrSpc	10A const				000000		13011300
1227+D i_lib	10A const options(*nopass: *omit)				000000		13011400
1228+D o_msg	likes(msg_t)				000000		13011500
1229+D	options(*nopass: *omit)				000000		13011600
1230+ *					000000		13011700
1231+ *	Returns cTrue if the user space is extended automatically by the				000000		13011800
1232+ *	system when the end of the space is encountered.				000000		13011900
1233+D f_getUsrSpcAutoExt...					000000		13012000
1234+D PR	N				000000		13012100
1235+D	extproc('BASICS1R5+				000000		13012200
1236+D	_f_getUsrSpcAutoExt+				000000		13012300
1237+D	')				000000		13012400
1238+D i_usrSpc	10A const				000000		13012500
1239+D i_lib	10A const options(*nopass: *omit)				000000		13012600
1240+D o_msg	likes(msg_t)				000000		13012700
1241+D	options(*nopass: *omit)				000000		13012800
1242+ *					000000		13012900
1243+ *	Returns the initial value of a given user space.				000000		13013000
1244+D f_getUsrSpcInitValue...					000000		13013100
1245+D PR	1A varying				000000		13013200
1246+D	extproc('BASICS1R5+				000000		13013300
1247+D	_f_getUsrSpcInitValue+				000000		13013400
1248+D	')				000000		13013500
1249+D i_usrSpc	10A const				000000		13013600
1250+D i_lib	10A const options(*nopass: *omit)				000000		13013700
1251+D o_msg	likes(msg_t)				000000		13013800
1252+D	options(*nopass: *omit)				000000		13013900
1253+ *					000000		13014000
1254+ *	Changes the size of a given user space.				000000		13014100
1255+D f_chgUsrSpcSize...					000000		13014200
1256+D PR	10I 0				000000		13014300
1257+D	extproc('BASICS1R5+				000000		13014400
1258+D	_f_chgUsrSpcSize+				000000		13014500
1259+D	')				000000		13014600
1260+D i_usrSpc	10A const				000000		13014700
1261+D i_lib	10A const options(*omit)				000000		13014800
1262+D i_size	10I 0 const				000000		13014900

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
1263+D	o_msg	liked(msg_t)			RADDAT	000000	13015000
1264+D		options(*nopass: *omit)			RADDAT	000000	13015100
1265+ *					RADDAT	000000	13015200
1266+ *	Changes the automatic extendibility of a given user space.				RADDAT	000000	13015300
1267+D	f_chgUsrSpcAutoExt...				RADDAT	000000	13015400
1268+D	PR N				RADDAT	000000	13015500
1269+D		extproc('BASICS1R5+			RADDAT	000000	13015600
1270+D		_f_chgUsrSpcAutoExt+			RADDAT	000000	13015700
1271+D		')			RADDAT	000000	13015800
1272+D	i_usrSpc	10A const			RADDAT	000000	13015900
1273+D	i_lib	10A const options(*omit)			RADDAT	000000	13016000
1274+D	i_isAutoExt	N const			RADDAT	000000	13016100
1275+D	o_msg	liked(msg_t)			RADDAT	000000	13016200
1276+D		options(*nopass: *omit)			RADDAT	000000	13016300
1277+ *					RADDAT	000000	13016400
1278+ *	Changes the initial value of a given user space.				RADDAT	000000	13016500
1279+D	f_chgUsrSpcInitValue...				RADDAT	000000	13016600
1280+D	PR 1A	varying			RADDAT	000000	13016700
1281+D		extproc('BASICS1R5+			RADDAT	000000	13016800
1282+D		_f_chgUsrSpcInitValue+			RADDAT	000000	13016900
1283+D		')			RADDAT	000000	13017000
1284+D	i_usrSpc	10A const			RADDAT	000000	13017100
1285+D	i_lib	10A const options(*omit)			RADDAT	000000	13017200
1286+D	i_initValue	1A const			RADDAT	000000	13017300
1287+D	o_msg	liked(msg_t)			RADDAT	000000	13017400
1288+D		options(*nopass: *omit)			RADDAT	000000	13017500
1289+ *					RADDAT	000000	13017600
1290+ *	Deletes a given user space.				RADDAT	000000	13017700
1291+D	f_dltUsrSpc...				RADDAT	000000	13017800
1292+D	PR N				RADDAT	000000	13017900
1293+D		extproc('BASICS1R5+			RADDAT	000000	13018000
1294+D		_f_dltUsrSpc+			RADDAT	000000	13018100
1295+D		')			RADDAT	000000	13018200
1296+D	i_usrSpc	10A const			RADDAT	000000	13018300
1297+D	i_lib	10A const options(*nopass: *omit)			RADDAT	000000	13018400
1298+D	o_msg	liked(msg_t)			RADDAT	000000	13018500
1299+D		options(*nopass: *omit)			RADDAT	000000	13018600
1300+ *					RADDAT	000000	13018700
1301+ *	Activates a service program.				RADDAT	000000	13018800
1302+D	f_activateSrvPgm...				RADDAT	000000	13018900
1303+D	PR	likeds(actSrvPgm_t)			RADDAT	000000	13019000
1304+D		extproc('BASICS1R5+			RADDAT	000000	13019100
1305+D		_f_activateSrvPgm+			RADDAT	000000	13019200
1306+D		')			RADDAT	000000	13019300
1307+D	i_srvPgm	10A const			RADDAT	000000	13019400
1308+D	i_lib	10A const options(*nopass: *omit)			RADDAT	000000	13019500
1309+D	o_msg	liked(msg_t)			RADDAT	000000	13019600
1310+D		options(*nopass: *omit)			RADDAT	000000	13019700
1311+ *					RADDAT	000000	13019800
1312+ *	Gets a procedure pointer by name.				RADDAT	000000	13019900
1313+D	f_getProcPtrByName...				RADDAT	000000	13020000
1314+D	PR	likeds(actProc_t)			RADDAT	000000	13020100
1315+D		extproc('BASICS1R5+			RADDAT	000000	13020200
1316+D		_f_getProcPtrByName+			RADDAT	000000	13020300
1317+D		')			RADDAT	000000	13020400
1318+D	i_proc	256A const			RADDAT	000000	13020500

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
1319+D	i_actSrvPgm	const likeds(actSrvPgm_t)			RADDAT		13020600
1320+D	o_msg	likeds(msg_t)			RADDAT		13020700
1321+D		options(*nopass: *omit)			RADDAT		13020800
1322+ *					RADDAT		13020900
1323+ *	Returns the name of the library that contains a given object.				RADDAT		13021000
1324+D	f_getObjLib...				RADDAT		13021100
1325+D	PR 10A				RADDAT		13021200
1326+D		extproc('BASICS1R5+			RADDAT		13021300
1327+D		_f_getObjLib+			RADDAT		13021400
1328+D		')			RADDAT		13021500
1329+D	i_obj 10A	const			RADDAT		13021600
1330+D	i_lib 10A	const options(*omit)			RADDAT		13021700
1331+D	i_type 10A	const			RADDAT		13021800
1332+ *					RADDAT		13021900
1333+ *					RADDAT		13022000
1334+ *	Imported prototypes				RADDAT		13022100
1335+ *					RADDAT		13022200
1336+ /COPY QBASICS1,MILIB	MI-Library			CPY	RADDAT	000000	13022300
-----*							
* RPG member name : MILIB							18
* External name : WSDL2RPG/QBASICS1(MILIB)							18
* Last change : 08/19/09 12:46:31							18
* Text 'description' : Prototypes for MI-Library (include)							18
-----*							
1337+ /IF NOT DEFINED(MILIB)				DEF	RADDAT	000000	18000100
1338+ /DEFINE MILIB				DEF	RADDAT	000000	18000200
1339+ *					RADDAT	000000	18000300
1340+ *	QSYSINC/MIH.MICOMMON: Requested Authority bitmasks and typedef				RADDAT	000000	18000400
1341+ *	The bitmasks are or'd together to give				RADDAT	000000	18000500
1342+ *	desired combinations				RADDAT	000000	18000600
1343+D	AUTH_OBJ_CTRL C	const(x'8000')			RADDAT	000000	18000700
1344+D	AUTH_OBJ_MGMT C	const(x'4000')			RADDAT	000000	18000800
1345+D	AUTH_POINTER C	const(x'2000')			RADDAT	000000	18000900
1346+D	AUTH_SPACE C	const(x'1000')			RADDAT	000000	18001000
1347+D	AUTH_RETRIEVE C	const(x'0800')			RADDAT	000000	18001100
1348+D	AUTH_INSERT C	const(x'0400')			RADDAT	000000	18001200
1349+D	AUTH_DELETE C	const(x'0200')			RADDAT	000000	18001300
1350+D	AUTH_UPDATE C	const(x'0100')			RADDAT	000000	18001400
1351+D	AUTH_OWNER C	const(x'0080')			RADDAT	000000	18001500
1352+D	AUTH_EXCLUDED C	const(x'0040')			RADDAT	000000	18001600
1353+D	AUTH_LST_MGMT C	const(x'0020')			RADDAT	000000	18001700
1354+D	AUTH_EXECUTE C	const(x'0010')			RADDAT	000000	18001800
1355+D	AUTH_ALTER C	const(x'0008')			RADDAT	000000	18001900
1356+D	AUTH_REF C	const(x'0004')			RADDAT	000000	18002000
1357+D	AUTH_ALL C	const(x'FF1C')			RADDAT	000000	18002100
1358+D	AUTH_NONE C	const(x'0000')			RADDAT	000000	18002200
1359+ *					RADDAT	000000	18002300
1360+ *	QSYSINC/MIH.MIOBJTYP: MI Object Type				RADDAT	000000	18002400
1361+D	MI_Program C	const(x'0201')			RADDAT	000000	18002500
1362+D	MI_Library C	const(x'0401')			RADDAT	000000	18002600
1363+D	MI_Usrspc C	const(x'1934')			RADDAT	000000	18002700
1364+D	MI_Usrq C	const(x'0A02')			RADDAT	000000	18002800
1365+D	MI_Usridx C	const(x'0E0A')			RADDAT	000000	18002900
1366+D	WLI_ALRTBL C	const(x'0E09')			RADDAT	000000	18003000
1367+D	WLI_AUTL C	const(x'1B01')			RADDAT	000000	18003100
1368+D	WLI_BLKSF C	const(x'1E05')			RADDAT	000000	18003200

Line Number	Source Specifications					Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
1369+D	WLI_BNDDIR	C			const(x'1937')	RADDAT			000000		18003300
1370+D	WLI_CFGL	C			const(x'0E0D')	RADDAT			000000		18003400
1371+D	WLI_CHTFMT	C			const(x'190D')	RADDAT			000000		18003500
1372+D	WLI_CLD	C			const(x'190B')	RADDAT			000000		18003600
1373+D	WLI_CLS	C			const(x'1904')	RADDAT			000000		18003700
1374+D	WLI_CMD	C			const(x'1905')	RADDAT			000000		18003800
1375+D	WLI_CNNL	C			const(x'1701')	RADDAT			000000		18003900
1376+D	WLI_COSD	C			const(x'1401')	RADDAT			000000		18004000
1377+D	WLI_CRG	C			const(x'192C')	RADDAT			000000		18004100
1378+D	WLI_CRQD	C			const(x'0E0F')	RADDAT			000000		18004200
1379+D	WLI_CSI	C			const(x'1935')	RADDAT			000000		18004300
1380+D	WLI_CSPMAP	C			const(x'1922')	RADDAT			000000		18004400
1381+D	WLI_CSPTBL	C			const(x'1923')	RADDAT			000000		18004500
1382+D	WLI_CTLD	C			const(x'1201')	RADDAT			000000		18004600
1383+D	WLI_DDIR	C			const(x'1F02')	RADDAT			000000		18004700
1384+D	WLI_DEVD	C			const(x'1001')	RADDAT			000000		18004800
1385+D	WLI_DIR	C			const(x'0C01')	RADDAT			000000		18004900
1386+D	WLI_DOC	C			const(x'190E')	RADDAT			000000		18005000
1387+D	WLI_DSTMF	C			const(x'1F01')	RADDAT			000000		18005100
1388+D	WLI_DTAARA	C			const(x'190A')	RADDAT			000000		18005200
1389+D	WLI_DTADCT	C			const(x'1920')	RADDAT			000000		18005300
1390+D	WLI_DTAQ	C			const(x'0A01')	RADDAT			000000		18005400
1391+D	WLI_EDTD	C			const(x'1908')	RADDAT			000000		18005500
1392+D	WLI_EXITRG	C			const(x'1913')	RADDAT			000000		18005600
1393+D	WLI_FCT	C			const(x'0E04')	RADDAT			000000		18005700
1394+D	WLI_FILE	C			const(x'1901')	RADDAT			000000		18005800
1395+D	WLI_FLR	C			const(x'1912')	RADDAT			000000		18005900
1396+D	WLI_FNTRSC	C			const(x'1926')	RADDAT			000000		18006000
1397+D	WLI_FNTTBL	C			const(x'192B')	RADDAT			000000		18006100
1398+D	WLI_FORMDF	C			const(x'1928')	RADDAT			000000		18006200
1399+D	WLI_FTR	C			const(x'0E0B')	RADDAT			000000		18006300
1400+D	WLI_GSS	C			const(x'190C')	RADDAT			000000		18006400
1401+D	WLI_IGCDCT	C			const(x'0E06')	RADDAT			000000		18006500
1402+D	WLI_IGCSRT	C			const(x'191A')	RADDAT			000000		18006600
1403+D	WLI_IGCTBL	C			const(x'1910')	RADDAT			000000		18006700
1404+D	WLI_IPXD	C			const(x'191E')	RADDAT			000000		18006800
1405+D	WLI_JOB	C			const(x'1903')	RADDAT			000000		18006900
1406+D	WLI_JOBQ	C			const(x'0E01')	RADDAT			000000		18007000
1407+D	WLI_JOBSCD	C			const(x'0E0C')	RADDAT			000000		18007100
1408+D	WLI_JRN	C			const(x'0901')	RADDAT			000000		18007200
1409+D	WLI_JRNRCV	C			const(x'0701')	RADDAT			000000		18007300
1410+D	WLI_LIB	C			const(x'0401')	RADDAT			000000		18007400
1411+D	WLI_LIND	C			const(x'1101')	RADDAT			000000		18007500
1412+D	WLI_LOCALE	C			const(x'1921')	RADDAT			000000		18007600
1413+D	WLI_MEDDFN	C			const(x'191C')	RADDAT			000000		18007700
1414+D	WLI_MENU	C			const(x'1916')	RADDAT			000000		18007800
1415+D	WLI_MGTCOL	C			const(x'192D')	RADDAT			000000		18007900
1416+D	WLI_MODD	C			const(x'1501')	RADDAT			000000		18008000
1417+D	WLI_MODULE	C			const(x'0301')	RADDAT			000000		18008100
1418+D	WLI_MSGF	C			const(x'0E03')	RADDAT			000000		18008200
1419+D	WLI_MSGQ	C			const(x'1902')	RADDAT			000000		18008300
1420+D	WLI_M36	C			const(x'1E04')	RADDAT			000000		18008400
1421+D	WLI_M36CFG	C			const(x'1924')	RADDAT			000000		18008500
1422+D	WLI_NODL	C			const(x'0E0E')	RADDAT			000000		18008600
1423+D	WLI_NTBD	C			const(x'1914')	RADDAT			000000		18008700
1424+D	WLI_NWID	C			const(x'1601')	RADDAT			000000		18008800

Line Number	----- Source Specifications -----><----- Comments ----->										Do Num	Page Line	Change Date	Src Id	Seq Number
1425+D	WLI_NWSD	C				const(x'1D01')							RADDAT	000000	18008900
1426+D	WLI_NODGRP	C				const(x'192A')							RADDAT	000000	18009000
1427+D	WLI_OOPOOL	C				const(x'2002')							RADDAT	000000	18009100
1428+D	WLI_OUTQ	C				const(x'0E02')							RADDAT	000000	18009200
1429+D	WLI_OVL	C				const(x'1929')							RADDAT	000000	18009300
1430+D	WLI_PAGDFN	C				const(x'1936')							RADDAT	000000	18009400
1431+D	WLI_PAGSEG	C				const(x'1927')							RADDAT	000000	18009500
1432+D	WLI_PDG	C				const(x'1930')							RADDAT	000000	18009600
1433+D	WLI_PGM	C				const(x'0201')							RADDAT	000000	18009700
1434+D	WLI_PNLGRP	C				const(x'1915')							RADDAT	000000	18009800
1435+D	WLI_PRDAVL	C				const(x'1933')							RADDAT	000000	18009900
1436+D	WLI_PRDDFN	C				const(x'191B')							RADDAT	000000	18010000
1437+D	WLI_PRDLOD	C				const(x'191D')							RADDAT	000000	18010100
1438+D	WLI_PSFCFG	C				const(x'1925')							RADDAT	000000	18010200
1439+D	WLI_QMFORM	C				const(x'1932')							RADDAT	000000	18010300
1440+D	WLI_QMQRY	C				const(x'1931')							RADDAT	000000	18010400
1441+D	WLI_QRYDFN	C				const(x'1911')							RADDAT	000000	18010500
1442+D	WLI_RCT	C				const(x'0E08')							RADDAT	000000	18010600
1443+D	WLI_S36	C				const(x'1919')							RADDAT	000000	18010700
1444+D	WLI_SBSD	C				const(x'1909')							RADDAT	000000	18010800
1445+D	WLI_SCHIDX	C				const(x'0E07')							RADDAT	000000	18010900
1446+D	WLI_SOCKET	C				const(x'1E03')							RADDAT	000000	18011000
1447+D	WLI_SOMOBJ	C				const(x'2001')							RADDAT	000000	18011100
1448+D	WLI_SPADCT	C				const(x'1C01')							RADDAT	000000	18011200
1449+D	WLI_SQLPKG	C				const(x'0202')							RADDAT	000000	18011300
1450+D	WLI_SQLUDT	C				const(x'191F')							RADDAT	000000	18011400
1451+D	WLI_SRVPGM	C				const(x'0203')							RADDAT	000000	18011500
1452+D	WLI_SSND	C				const(x'0E05')							RADDAT	000000	18011600
1453+D	WLI_STMF	C				const(x'1E01')							RADDAT	000000	18011700
1454+D	WLI_SVRSTG	C				const(x'1917')							RADDAT	000000	18011800
1455+D	WLI_SYMLNK	C				const(x'1E02')							RADDAT	000000	18011900
1456+D	WLI_TBL	C				const(x'1906')							RADDAT	000000	18012000
1457+D	WLI_USRPRF	C				const(x'0801')							RADDAT	000000	18012100
1458+D	WLI_USRIDX	C				const(x'0E0A')							RADDAT	000000	18012200
1459+D	WLI_USRQ	C				const(x'0A02')							RADDAT	000000	18012300
1460+D	WLI_USRSPC	C				const(x'1934')							RADDAT	000000	18012400
1461+D	WLI_VLDL	C				const(x'0E10')							RADDAT	000000	18012500
1462+D	WLI_WSCST	C				const(x'1938')							RADDAT	000000	18012600
1463+ *													RADDAT	000000	18012700
1464+ *	Activate Bound Program, Returned value: Activation mark												RADDAT	000000	18012800
1465+D	QleActBndPgm	PR		10I	0	extproc('QleActBndPgm')							RADDAT	000000	18012900
1466+D	i_pgmPtr			*		const procptr							RADDAT	000000	18013000
1467+D	i_actMark			10I	0	const options(*omit)							RADDAT	000000	18013100
1468+D	i_actInf			32767A		const options(*omit: *varsize)							RADDAT	000000	18013200
1469+D	i_actInfLen			10I	0	const options(*omit)							RADDAT	000000	18013300
1470+D	io_errCode			32767A		options(*omit: *varsize)							RADDAT	000000	18013400
1471+ *													RADDAT	000000	18013500
1472+ *	Type definitions												RADDAT	000000	18013600
1473+D	actInf_t	DS				qualified based(pDummy)							RADDAT	000000	18013700
1474+D	bytRet			1	4I	0							RADDAT	000000	18013800
1475+D	bytAvl			5	8I	0							RADDAT	000000	18013900
1476+D	reserved_1			9	16A								RADDAT	000000	18014000
1477+D	actGrpMark			17	20I	0							RADDAT	000000	18014100
1478+D	actMark			21	24I	0							RADDAT	000000	18014200
1479+D	reserved_2			25	31A								RADDAT	000000	18014300
1480+D	flags			32	32A								RADDAT	000000	18014400

Line	Number	1	2	3	4	5	6	7	8	9	10	Do Num	Page Line	Change Date	Src Id	Seq Number
1537+D	i_pSource				*	value					RADDAT			000000		18020100
1538+D	i_length				10U	0	value				RADDAT			000000		18020200
1539+	*										RADDAT			000000		18020300
1540+	*	The standard definition for time in the MI library (QSYSINC/MIH.MICOMMON):									RADDAT			000000		18020400
1541+D	mi_time_t		S		8A	inz					RADDAT			000000		18020500
1542+	*										RADDAT			000000		18020600
1543+	*	The following values can be OR'd for the Wait Options parameter									RADDAT			000000		18020700
1544+	*	on the WAITTIME invocation.									RADDAT			000000		18020800
1545+D	cWAIT_NORMAL...										RADDAT			000000		18020900
1546+D			C			0				x'0000'	RADDAT			000000		18021000
1547+D	cWAIT_MOD_AS_ENTER...										RADDAT			000000		18021100
1548+D			C			32768				x'8000'	RADDAT			000000		18021200
1549+D	cWAIT_MOD_AS_LEAVE...										RADDAT			000000		18021300
1550+D			C			16384				x'4000'	RADDAT			000000		18021400
1551+D	cWAIT_MPL_SET...										RADDAT			000000		18021500
1552+D			C			8192				x'2000'	RADDAT			000000		18021600
1553+D	cWAIT_SIGNALS...										RADDAT			000000		18021700
1554+D			C			4096				x'1000'	RADDAT			000000		18021800
1555+	*										RADDAT			000000		18021900
1556+	*	Wait On Time (WAITTIME)									RADDAT			000000		18022000
1557+D	waitTime		PR				extproc('waittime')				RADDAT			000000		18022100
1558+D	pmi_time						const like(mi_time_t)				RADDAT			000000		18022200
1559+D	option				5I	0	value				RADDAT			000000		18022300
1560+	*										RADDAT			000000		18022400
1561+	*	Use mitime function to form an AS/400 system value for time									RADDAT			000000		18022500
1562+D	mitime		PR		*		extproc('mitime')				RADDAT			000000		18022600
1563+D	mi_time						const like(mi_time_t)				RADDAT			000000		18022700
1564+D	hour				10I	0	value				RADDAT			000000		18022800
1565+D	min				10I	0	value				RADDAT			000000		18022900
1566+D	sec				10I	0	value				RADDAT			000000		18023000
1567+D	hsec				10I	0	value				RADDAT			000000		18023100
1568+	*										RADDAT			000000		18023200
1569+	/ENDIF										RADDAT		DEF	000000		18023300
1570+	/COPY QBASICS1,PBASICS1R1						Messages						CPY	000000		13022400
	*										*					
	*	RPG member name : PBASICS1R1									*					19
	*	External name : WSDL2RPG/QBASICS1(PBASICS1R1)									*					19
	*	Last change : 08/19/09 12:46:31									*					19
	*	Text 'description' : SrvPgm: Basic Routines - Messages cpybk									*					19
	*										*					
1571+	/IF NOT DEFINED(PBASICS1R1)													000000		19000100
	LINES EXCLUDED: 216															
1572+	/ENDIF										RADDAT			000000		19021800
1573+	/COPY QBASICS1,PBASICS1R7						Mathematics						CPY	000000		13022500
	*										*					
	*	RPG member name : PBASICS1R7									*					20
	*	External name : WSDL2RPG/QBASICS1(PBASICS1R7)									*					20
	*	Last change : 08/19/09 12:46:32									*					20
	*	Text 'description' : SrvPgm: Basic Routines - Mathematics cpybk									*					20
	*										*					
1574+	/IF NOT DEFINED(BASICS1R7)										RADDAT			000000		20000100
1575+	/DEFINE BASICS1R7										RADDAT			000000		20000200
1576+	*										RADDAT			000000		20000300
1577+	*	Constants									RADDAT			000000		20000400
1578+	*										RADDAT			000000		20000500
1579+	*	Generate UUID:									RADDAT			000000		20000600

Line	----- Source Specifications -----><----- Comments ----->										Do	Page	Change	Src	Seq
Number	1	2	3	4	5	6	7	8	9	10	Num	Line	Date	Id	Number
1580+D	cUUID_HEX...									RADDAT			000000		20000700
1581+D		C		const(1)						RADDAT			000000		20000800
1582+D	cUUID_STRING...									RADDAT			000000		20000900
1583+D		C		const(2)						RADDAT			000000		20001000
1584+ *										RADDAT			000000		20001100
1585+D	uuid_t	DS		qualified						RADDAT			000000		20001200
1586+D	length		5U	0						RADDAT			000000		20001300
1587+D	value		32A							RADDAT			000000		20001400
1588+ *										RADDAT			000000		20001500
1589+ *	Generate MD5:									RADDAT			000000		20001600
1590+D	cMD5_HEX...									RADDAT			000000		20001700
1591+D		C		const(4352)		256	+	4096		RADDAT			000000		20001800
1592+D	cMD5_STRING...									RADDAT			000000		20001900
1593+D		C		const(4608)		512	+	4096		RADDAT			000000		20002000
1594+D	cMD5_ONLY...									RADDAT			000000		20002100
1595+D		C		const(17)		1	+	16		RADDAT			000000		20002200
1596+D	cMD5_FIRST...									RADDAT			000000		20002300
1597+D		C		const(18)		2	+	16		RADDAT			000000		20002400
1598+D	cMD5_MIDDLE...									RADDAT			000000		20002500
1599+D		C		const(20)		4	+	16		RADDAT			000000		20002600
1600+D	cMD5_FINAL...									RADDAT			000000		20002700
1601+D		C		const(24)		8	+	16		RADDAT			000000		20002800
1602+ *										RADDAT			000000		20002900
1603+D	md5_t	DS		qualified						RADDAT			000000		20003000
1604+D	length		5U	0						RADDAT			000000		20003100
1605+D	value		32A							RADDAT			000000		20003200
1606+ *										RADDAT			000000		20003300
1607+ *	Generate SHA-1:									RADDAT			000000		20003400
1608+D	cSHA1_HEX...									RADDAT			000000		20003500
1609+D		C		const(8448)		256	+	8192		RADDAT			000000		20003600
1610+D	cSHA1_STRING...									RADDAT			000000		20003700
1611+D		C		const(8704)		512	+	8192		RADDAT			000000		20003800
1612+D	cSHA1_ONLY...									RADDAT			000000		20003900
1613+D		C		const(33)		1	+	32		RADDAT			000000		20004000
1614+D	cSHA1_FIRST...									RADDAT			000000		20004100
1615+D		C		const(34)		2	+	32		RADDAT			000000		20004200
1616+D	cSHA1_MIDDLE...									RADDAT			000000		20004300
1617+D		C		const(36)		4	+	32		RADDAT			000000		20004400
1618+D	cSHA1_FINAL...									RADDAT			000000		20004500
1619+D		C		const(40)		8	+	32		RADDAT			000000		20004600
1620+ *										RADDAT			000000		20004700
1621+D	sha1_t	DS		qualified						RADDAT			000000		20004800
1622+D	length		5U	0						RADDAT			000000		20004900
1623+D	value		40A							RADDAT			000000		20005000
1624+ *										RADDAT			000000		20005100
1625+ *	Generate CRC32:									RADDAT			000000		20005200
1626+D	cCRC32_ONLY...									RADDAT			000000		20005300
1627+D		C		const(33)		1	+	64		RADDAT			000000		20005400
1628+D	cCRC32_FIRST...									RADDAT			000000		20005500
1629+D		C		const(34)		2	+	64		RADDAT			000000		20005600
1630+D	cCRC32_MIDDLE...									RADDAT			000000		20005700
1631+D		C		const(36)		4	+	64		RADDAT			000000		20005800
1632+D	cCRC32_FINAL...									RADDAT			000000		20005900
1633+D		C		const(40)		8	+	64		RADDAT			000000		20006000
1634+ *										RADDAT			000000		20006100
1635+D	crc32_t	DS		qualified						RADDAT			000000		20006200

Line	----- Source Specifications -----><----- Comments ----->										Do	Page	Change	Src	Seq
Number	1	2	3	4	5	6	7	8	9	10	Num	Line	Date	Id	Number
1636+D	length			5U	0					RADDAT			000000		20006300
1637+D	value			40A						RADDAT			000000		20006400
1638+ *										RADDAT			000000		20006500
1639+ *	Tests for one or more bits.									RADDAT			000000		20006600
1640+D	f_isBitOn...									RADDAT			000000		20006700
1641+D		PR		N						RADDAT			000000		20006800
1642+D						extproc('BASICS1R7+				RADDAT			000000		20006900
1643+D						_f_isBitOn+				RADDAT			000000		20007000
1644+D						')				RADDAT			000000		20007100
1645+D	i_testBit			10U	0	value				RADDAT			000000		20007200
1646+D	i_value			10U	0	value				RADDAT			000000		20007300
1647+ *										RADDAT			000000		20007400
1648+ *	Computes the logarithm base 10 of a given value.									RADDAT			000000		20007500
1649+D	f_log10...									RADDAT			000000		20007600
1650+D		PR		8F						RADDAT			000000		20007700
1651+D						extproc('BASICS1R7+				RADDAT			000000		20007800
1652+D						_f_log10+				RADDAT			000000		20007900
1653+D						')				RADDAT			000000		20008000
1654+D	i_value			8F		value				RADDAT			000000		20008100
1655+ *										RADDAT			000000		20008200
1656+ *	Computes the length of a RPG integer field from its buffer length									RADDAT			000000		20008300
1657+D	f_clcLengthOfRPGIntFieldFromBufSize...									RADDAT			000000		20008400
1658+D		PR		5U	0					RADDAT			000000		20008500
1659+D						extproc('BASICS1R7+				RADDAT			000000		20008600
1660+D						_f_clcLengthOfRPGIntFieldFromBufSize+				RADDAT			000000		20008700
1661+D						')				RADDAT			000000		20008800
1662+D	i_buffSize			5U	0	value				RADDAT			000000		20008900
1663+ *										RADDAT			000000		20009000
1664+ *	Generates a Universal Unique Identifier (UUID).												000000		20009100
1665+D	f_genUUID...									RADDAT			000000		20009200
1666+D		PR				likeds(uuid_t)				RADDAT			000000		20009300
1667+D						extproc('BASICS1R7+				RADDAT			000000		20009400
1668+D						_f_genUUID+				RADDAT			000000		20009500
1669+D						')				RADDAT			000000		20009600
1670+D	i_option			5U	0	value	options(*nopass)			RADDAT			000000		20009700
1671+ *										RADDAT			000000		20009800
1672+ *	Generates a MD5 fingerprint (hash code).									RADDAT			000000		20009900
1673+D	f_genMD5...									RADDAT			000000		20010000
1674+D		PR				likeds(md5_t)				RADDAT			000000		20010100
1675+D						extproc('BASICS1R7+				RADDAT			000000		20010200
1676+D						_f_genMD5+				RADDAT			000000		20010300
1677+D						')				RADDAT			000000		20010400
1678+D	i_pString			*	const	options(*string)				RADDAT			000000		20010500
1679+D	i_option			5U	0	const	options(*nopass)			RADDAT			000000		20010600
1680+D	i_length			10U	0	const	options(*nopass)			RADDAT			000000		20010700
1681+ *										RADDAT			000000		20010800
1682+ *	Generates a SHA-1 fingerprint (hash code).									RADDAT			000000		20010900
1683+D	f_genSHA1...									RADDAT			000000		20011000
1684+D		PR				likeds(sha1_t)				RADDAT			000000		20011100
1685+D						extproc('BASICS1R7+				RADDAT			000000		20011200
1686+D						_f_genSHA1+				RADDAT			000000		20011300
1687+D						')				RADDAT			000000		20011400
1688+D	i_pString			*	const	options(*string)				RADDAT			000000		20011500
1689+D	i_option			5U	0	const	options(*nopass)			RADDAT			000000		20011600
1690+D	i_length			10U	0	const	options(*nopass)			RADDAT			000000		20011700
1691+ *										RADDAT			000000		20011800

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
1692+	* Generates a CRC32 checksum.	RADDAT			000000		20011900
1693+D	f_genCRC32...	RADDAT			000000		20012000
1694+D	PR likeds(crc32_t)	RADDAT			000000		20012100
1695+D	extproc('BASICS1R7+	RADDAT			000000		20012200
1696+D	_f_genCRC32+	RADDAT			000000		20012300
1697+D	'')	RADDAT			000000		20012400
1698+D	i_pString * const options(*string)	RADDAT			000000		20012500
1699+D	i_option 5U 0 const options(*nopass)	RADDAT			000000		20012600
1700+D	i_length 10U 0 const options(*nopass)	RADDAT			000000		20012700
1701+	*	RADDAT			000000		20012800
1702+	*	RADDAT			000000		20012900
1703+	* Imported prototypes	RADDAT			000000		20013000
1704+	*	RADDAT			000000		20013100
1705+	/COPY QBASICS1,PBASICS1R3 Strings	RADDAT		CPY	000000		20013200
-----*							
* RPG member name : PBASICS1R3							21
* External name : WSDL2RPG/QBASICS1(PBASICS1R3)							21
* Last change : 08/19/09 12:46:32							21
* Text 'description' : SrvPgm: Basic Routines - Strings cpybk							21
-----*							
1706+	/IF NOT DEFINED(BASICS1R3)				000000		21000100
1707+	/DEFINE BASICS1R3				000000		21000200
1708+	*	RADDAT			000000		21000300
1709+	* Reference fields	RADDAT			000000		21000400
1710+	*	RADDAT			000000		21000500
1711+	* Konstants: f_wordWrap()	RADDAT			000000		21000600
1712+D	cWORDWRAP_PARAGRAPH_NONE...	RADDAT			000000		21000700
1713+D	C const(1)	RADDAT			000000		21000800
1714+D	cWORDWRAP_PARAGRAPH_MSGD...	RADDAT			000000		21000900
1715+D	C const(2)	RADDAT			000000		21001000
1716+D	cWORDWRAP_PARAGRAPH_TEXT...	RADDAT			000000		21001100
1717+D	C const(4)	RADDAT			000000		21001200
1718+D	cWORDWRAP_PARAGRAPH_TRIMR...	RADDAT			000000		21001300
1719+D	C const(8)	RADDAT			000000		21001400
1720+	*	RADDAT			000000		21001500
1721+	* Returns the left portion of a string.	RADDAT			000000		21001600
1722+D	f_left...	RADDAT			000000		21001700
1723+D	PR 4096A varying	RADDAT			000000		21001800
1724+D	extproc('BASICS1R3+	RADDAT			000000		21001900
1725+D	_f_left+	RADDAT			000000		21002000
1726+D	'')	RADDAT			000000		21002100
1727+D	i_string 4096A const varying	RADDAT			000000		21002200
1728+D	i_length 10I 0 const	RADDAT			000000		21002300
1729+	*	RADDAT			000000		21002400
1730+	* Tests if the specified string starts with the specified prefix.	RADDAT			000000		21002500
1731+D	f_startsWith...	RADDAT			000000		21002600
1732+D	PR N	RADDAT			000000		21002700
1733+D	extproc('BASICS1R3+	RADDAT			000000		21002800
1734+D	_f_startsWith+	RADDAT			000000		21002900
1735+D	'')	RADDAT			000000		21003000
1736+D	i_prefix 4096A const varying	RADDAT			000000		21003100
1737+D	i_string 4096A const varying	RADDAT			000000		21003200
1738+	*	RADDAT			000000		21003300
1739+	* Returns the right portion of a string.	RADDAT			000000		21003400
1740+D	f_right...	RADDAT			000000		21003500
1741+D	PR 4096A varying	RADDAT			000000		21003600

Line	----- Source Specifications -----><----- Comments ----->										Do	Page	Change	Src	Seq
Number	1	2	3	4	5	6	7	8	9	10	Num	Line	Date	Id	Number
1742+D					extproc('BASICS1R3+					RADDAT			000000		21003700
1743+D					_f_right+					RADDAT			000000		21003800
1744+D					')					RADDAT			000000		21003900
1745+D	i_string		4096A	const	varying					RADDAT			000000		21004000
1746+D	i_length		10I 0	const						RADDAT			000000		21004100
1747+ *										RADDAT			000000		21004200
1748+ *	Tests if the specified string ends with the specified suffix.												000000		21004300
1749+D	f_endsWith...									RADDAT			000000		21004400
1750+D		PR		N						RADDAT			000000		21004500
1751+D					extproc('BASICS1R3+					RADDAT			000000		21004600
1752+D					_f_endsWith+								000000		21004700
1753+D					')					RADDAT			000000		21004800
1754+D	i_suffix		4096A	const	varying					RADDAT			000000		21004900
1755+D	i_string		4096A	const	varying					RADDAT			000000		21005000
1756+ *										RADDAT			000000		21005100
1757+ *	Produces an empty string.												000000		21005200
1758+D	f_space...									RADDAT			000000		21005300
1759+D		PR		4096A	varying					RADDAT			000000		21005400
1760+D					extproc('BASICS1R3+					RADDAT			000000		21005500
1761+D					_f_space+					RADDAT			000000		21005600
1762+D					')					RADDAT			000000		21005700
1763+D	i_length		10I 0	const						RADDAT			000000		21005800
1764+D	i_char		1A	const	options(*nopass)					RADDAT			000000		21005900
1765+ *										RADDAT			000000		21006000
1766+ *	Converts a given string to lower case.												000000		21006100
1767+D	f_lcase...									RADDAT			000000		21006200
1768+D		PR		4096A	varying					RADDAT			000000		21006300
1769+D					extproc('BASICS1R3+					RADDAT			000000		21006400
1770+D					_f_lcase+					RADDAT			000000		21006500
1771+D					')					RADDAT			000000		21006600
1772+D	i_string		4096A	const	varying					RADDAT			000000		21006700
1773+ *										RADDAT			000000		21006800
1774+ *	Converts a given string to upper case.												000000		21006900
1775+D	f_ucase...									RADDAT			000000		21007000
1776+D		PR		4096A	varying					RADDAT			000000		21007100
1777+D					extproc('BASICS1R3+					RADDAT			000000		21007200
1778+D					_f_ucase+					RADDAT			000000		21007300
1779+D					')					RADDAT			000000		21007400
1780+D	i_string		4096A	const	varying					RADDAT			000000		21007500
1781+ *										RADDAT			000000		21007600
1782+ *	Replaces all occurrences of a search argument by												000000		21007700
1783+ *	a replacement string in a given string.												000000		21007800
1784+D	f_replace...									RADDAT			000000		21007900
1785+D		PR		4096A	varying					RADDAT			000000		21008000
1786+D					extproc('BASICS1R3+					RADDAT			000000		21008100
1787+D					_f_replace+					RADDAT			000000		21008200
1788+D					')					RADDAT			000000		21008300
1789+D	i_searchArg		4096A	const	varying					RADDAT			000000		21008400
1790+D	i_rplString		4096A	const	varying					RADDAT			000000		21008500
1791+D	i_string		4096A	const	varying					RADDAT			000000		21008600
1792+D	i_startPos		10U 0	const	options(*nopass)					RADDAT			000000		21008700
1793+ *										RADDAT			000000		21008800
1794+ *	Returns the EBCDIC value of a given character.												000000		21008900
1795+D	f_ebcdic...									RADDAT			000000		21009000
1796+D		PR		3U 0						RADDAT			000000		21009100
1797+D					extproc('BASICS1R3+					RADDAT			000000		21009200

Line	Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
1798+D		_f_ebcdic+	RADDAT			000000		21009300
1799+D		'')	RADDAT			000000		21009400
1800+D	i_char	1A const options(*nopass)	RADDAT			000000		21009500
1801+ *			RADDAT			000000		21009600
1802+ *	Returns the numeric value of a string.					000000		21009700
1803+D	f_val...		RADDAT			000000		21009800
1804+D	PR	8F	RADDAT			000000		21009900
1805+D		extproc('BASICS1R3+	RADDAT			000000		21010000
1806+D		_f_val+	RADDAT			000000		21010100
1807+D		'')	RADDAT			000000		21010200
1808+D	i_string	64A const varying	RADDAT			000000		21010300
1809+D	i_decPoint	1A const options(*nopass)	RADDAT			000000		21010400
1810+D	i_comma	1A const options(*nopass)	RADDAT			000000		21010500
1811+D	i_currSym	1A const options(*nopass)	RADDAT			000000		21010600
1812+ *			RADDAT			000000		21010700
1813+ *	Performs word wrapping on a given string.					000000		21010800
1814+ *	Returns the next line of i_string with each call to f_wordWrap().					000000		21010900
1815+D	f_wordWrap...		RADDAT			000000		21011000
1816+D	PR	4096A varying	RADDAT			000000		21011100
1817+D		extproc('BASICS1R3+	RADDAT			000000		21011200
1818+D		_f_wordWrap+	RADDAT			000000		21011300
1819+D		'')	RADDAT			000000		21011400
1820+D	i_string	4096A const varying	RADDAT			000000		21011500
1821+D	i_width	10U 0 const	RADDAT			000000		21011600
1822+D	io_startPos	10U 0	RADDAT			000000		21011700
1823+D	i_options	10U 0 const options(*nopass)	RADDAT			000000		21011800
1824+ *			RADDAT			000000		21011900
1825+ *	Transforms a numeric from its internal format to a					000000		21012000
1826+ *	character form suitable for displaying.					000000		21012100
1827+D	f_editCode...		RADDAT			000000		21012200
1828+D	PR	64A varying	RADDAT			000000		21012300
1829+D		extproc('BASICS1R3+	RADDAT			000000		21012400
1830+D		_f_editCode+	RADDAT			000000		21012500
1831+D		'')	RADDAT			000000		21012600
1832+D	i_pBuffer	* const	RADDAT			000000		21012700
1833+D	i_digits	10I 0 const	RADDAT			000000		21012800
1834+D	i_decPos	10I 0 const	RADDAT			000000		21012900
1835+D	i_varClass	10A const	RADDAT			000000		21013000
1836+D	i_editCode	2A const	RADDAT			000000		21013100
1837+ *			RADDAT			000000		21013200
1838+ *	Transforms a numeric from its internal format to a					000000		21013300
1839+ *	character form suitable for displaying.					000000		21013400
1840+D	f_editWord...		RADDAT			000000		21013500
1841+D	PR	512A varying	RADDAT			000000		21013600
1842+D		extproc('BASICS1R3+	RADDAT			000000		21013700
1843+D		_f_editWord+	RADDAT			000000		21013800
1844+D		'')	RADDAT			000000		21013900
1845+D	i_pBuffer	* const	RADDAT			000000		21014000
1846+D	i_digits	10I 0 const	RADDAT			000000		21014100
1847+D	i_varClass	10A const	RADDAT			000000		21014200
1848+D	i_editWord	256A const varying	RADDAT			000000		21014300
1849+ *			RADDAT			000000		21014400
1850+ *	Gets the length of a the formatted string of a given edit code.					000000		21014500
1851+D	f_getLengthEditCode...		RADDAT			000000		21014600
1852+D	PR	5U 0	RADDAT			000000		21014700
1853+D		extproc('BASICS1R3+	RADDAT			000000		21014800

Line	----- Source Specifications -----><----- Comments ----->										Do	Page	Change	Src	Seq
Number	1	2	3	4	5	6	7	8	9	10	Num	Line	Date	Id	Number
1854+D						_f_getLengthEditCode+							000000		21014900
1855+D						'')							000000		21015000
1856+D	i_digits		10I	0	const								000000		21015100
1857+D	i_decPos		10I	0	const								000000		21015200
1858+D	i_editCode		2A		const								000000		21015300
1859+ *													000000		21015400
1860+ *	Gets the length of a the formatted string of a given edit word.												000000		21015500
1861+D	f_getLengthEditWord...												000000		21015600
1862+D		PR		5U	0								000000		21015700
1863+D						extproc('BASICS1R3+							000000		21015800
1864+D						_f_getLengthEditWord+							000000		21015900
1865+D						'')							000000		21016000
1866+D	i_digits		10I	0	const								000000		21016100
1867+D	i_editWord		256A		const	varying							000000		21016200
1868+ *													000000		21016300
1869+ *	Removes a keyword from a given string.												000000		21016400
1870+D	f_rmvKeyword...												000000		21016500
1871+D		PR		4096A		varying							000000		21016600
1872+D						extproc('BASICS1R3+							000000		21016700
1873+D						_f_rmvKeyword+							000000		21016800
1874+D						'')							000000		21016900
1875+D	i_keyword		128A		const	varying							000000		21017000
1876+D	i_string		4096A		const	varying							000000		21017100
1877+ *													000000		21017200
1878+ *	Removes quotes from a given string.												000000		21017300
1879+D	f_rmvQuotes...												000000		21017400
1880+D		PR		4096A		varying							000000		21017500
1881+D						extproc('BASICS1R3+							000000		21017600
1882+D						_f_rmvQuotes+							000000		21017700
1883+D						'')							000000		21017800
1884+D	i_string		4096A		const	varying							000000		21017900
1885+D	i_quote		1A		const								000000		21018000
1886+ *													000000		21018100
1887+ *	Adds quotes to a given string.												000000		21018200
1888+D	f_addQuotes...												000000		21018300
1889+D		PR		4096A		varying							000000		21018400
1890+D						extproc('BASICS1R3+							000000		21018500
1891+D						_f_addQuotes+							000000		21018600
1892+D						'')							000000		21018700
1893+D	i_string		4096A		const	varying							000000		21018800
1894+D	i_quote		1A		const	options(*nopass)							000000		21018900
1895+ *													000000		21019000
1896+ *	Centers a string.												000000		21019100
1897+D	f_center...												000000		21019200
1898+D		PR		4096A		varying							000000		21019300
1899+D						extproc('BASICS1R3+							000000		21019400
1900+D						_f_center+							000000		21019500
1901+D						'')							000000		21019600
1902+D	i_string		4096A		const	varying							000000		21019700
1903+D	i_length		10U	0	const	options(*nopass)							000000		21019800
1904+ *													000000		21019900
1905+ *	Right aligns a string.												000000		21020000
1906+D	f_alignR...												000000		21020100
1907+D		PR		4096A		varying							000000		21020200
1908+D						extproc('BASICS1R3+							000000		21020300
1909+D						_f_alignR+							000000		21020400

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
1910+D					000000		21020500
1911+D i_string	4096A const varying				000000		21020600
1912+D i_length	10U 0 const options(*nopass)				000000		21020700
1913+ *					000000		21020800
1914+ *	Compare strings using wildcard characters. (Compare generic)				000000		21020900
1915+D f_genStrCmp...					000000		21021000
1916+D	PR N				000000		21021100
1917+D	extproc('BASICS1R3+				000000		21021200
1918+D	_f_genStrCmp+				000000		21021300
1919+D)				000000		21021400
1920+D i_string	4096A const varying				000000		21021500
1921+D i_mask	4096A const varying				000000		21021600
1922+D i_wildcard	2A const options(*nopass)				000000		21021700
1923+ *					000000		21021800
1924+ *	Compare strings using wildcard characters.				000000		21021900
1925+D f_genStrCmpIgnCase...					000000		21022000
1926+D	PR N				000000		21022100
1927+D	extproc('BASICS1R3+				000000		21022200
1928+D	_f_genStrCmpIgnCase+				000000		21022300
1929+D)				000000		21022400
1930+D i_string	4096A const varying				000000		21022500
1931+D i_mask	4096A const varying				000000		21022600
1932+D i_wildcard	2A const options(*nopass)				000000		21022700
1933+ *					000000		21022800
1934+ *	Returns the last position of a search argument in the source string.				000000		21022900
1935+D f_scanR...					000000		21023000
1936+D	PR 5I 0				000000		21023100
1937+D	extproc('BASICS1R3+				000000		21023200
1938+D	_f_scanR+				000000		21023300
1939+D)				000000		21023400
1940+D i_searchArg	4096A const varying				000000		21023500
1941+D i_string	4096A const varying				000000		21023600
1942+D i_start	5I 0 const options(*nopass)				000000		21023700
1943+ *					000000		21023800
1944+ *	Converts a given buffer to a character string of hex values.				000000		21023900
1945+D f_cvtBufferToHexString...					000000		21024000
1946+D	PR 4096A varying				000000		21024100
1947+D	extproc('BASICS1R3+				000000		21024200
1948+D	_f_cvtBufferToHexString+				000000		21024300
1949+D)				000000		21024400
1950+D i_pBuffer	* const				000000		21024500
1951+D i_size	5U 0 const				000000		21024600
1952+ *					000000		21024700
1953+ *	Converts a given character string of hex values to a buffer.				000000		21024800
1954+D f_cvtHexStringToBuffer...					000000		21024900
1955+D	PR 5U 0 opdesc				000000		21025000
1956+D	extproc('BASICS1R3+				000000		21025100
1957+D	_f_cvtHexStringToBuffer+				000000		21025200
1958+D)				000000		21025300
1959+D i_string	4096A const options(*varsize)				000000		21025400
1960+D i_pBuffer	* const				000000		21025500
1961+D i_size	5U 0 const				000000		21025600
1962+ *					000000		21025700
1963+ *	Tests, if a given string contains an invalid character				000000		21025800
1964+ *	at a given position.				000000		21025900
1965+D f_containsInvalidCharAt...					000000		21026000

Line	----- Source Specifications -----><----- Comments ----->										Do	Page	Change	Src	Seq
Number	1	2	3	4	5	6	7	8	9	10	Num	Line	Date	Id	Number
1966+D		PR		N									000000		21026100
1967+D					extproc('BASICS1R3+								000000		21026200
1968+D					_f_containsInvalidCharAt+								000000		21026300
1969+D					'')								000000		21026400
1970+D	i_string		4096A		const	varying	options(*varsize)						000000		21026500
1971+D	i_pos		10I 0		const								000000		21026600
1972+D	i_chars		256A		const	varying	options(*varsize)						000000		21026700
1973+ *													000000		21026800
1974+ *	Tests, if a given string contains a specific string												000000		21026900
1975+ *	at a given position.												000000		21027000
1976+D	f_containsStringAt...												000000		21027100
1977+D		PR		N									000000		21027200
1978+D					extproc('BASICS1R3+								000000		21027300
1979+D					_f_containsStringAt+								000000		21027400
1980+D					'')								000000		21027500
1981+D	i_string		4096A		const	varying	options(*varsize)						000000		21027600
1982+D	i_pos		10I 0		const								000000		21027700
1983+D	i_subString		4096A		const	varying	options(*varsize)						000000		21027800
1984+ *													000000		21027900
1985+ *	Returns the string representation of a qualified object name.												000000		21028000
1986+D	f_objName...												000000		21028100
1987+D		PR		21A		varying							000000		21028200
1988+D					extproc('BASICS1R3+								000000		21028300
1989+D					_f_objName+								000000		21028400
1990+D					'')								000000		21028500
1991+D	i_qObj				const	likes(qObj_t)							000000		21028600
1992+D	i_lib		10A		const	options(*nopass)							000000		21028700
1993+ *													000000		21028800
1994+ *	Replaces unprintable characters in a given string.												000000		21028900
1995+D	f_rplUnPrtChars...												000000		21029000
1996+D		PR		4096A		varying							000000		21029100
1997+D					extproc('BASICS1R3+								000000		21029200
1998+D					_f_rplUnPrtChars+								000000		21029300
1999+D					'')								000000		21029400
2000+D	i_string		4096A		const	varying							000000		21029500
2001+D	i_rplChar		1A		const	varying	options(*nopass)						000000		21029600
2002+ *													000000		21029700
2003+ *	Formats a given date according to a given puicture string.												000000		21029800
2004+D	f_formatDate...												000000		21029900
2005+D		PR		128A		varying							000000		21030000
2006+D					extproc('BASICS1R3+								000000		21030100
2007+D					_f_formatDate+								000000		21030200
2008+D					'')								000000		21030300
2009+D	i_date		D		const								000000		21030400
2010+D	i_picture		128A		const	varying							000000		21030500
2011+ *													000000		21030600
2012+ *	Parses a string representing a date.												000000		21030700
2013+D	f_parseDate...												000000		21030800
2014+D		PR		D									000000		21030900
2015+D					extproc('BASICS1R3+								000000		21031000
2016+D					_f_parseDate+								000000		21031100
2017+D					'')								000000		21031200
2018+D	i_string		128A		const	varying							000000		21031300
2019+D	i_picture		128A		const	varying	options(*nopass)						000000		21031400
2020+ *													000000		21031500
2021+ *	Konstants: f_formatMemSize												000000		21031600

Line Number	<----- Source Specifications -----><----- Comments ----->	Do Num	Page Line	Change Date	Src Id	Seq Number
2059+	/IF NOT DEFINED(BASICS1R7) LINES EXCLUDED: 137			000000		24000100
2060+	/ENDIF			000000		24013900
2061+	*			000000		21035000
2062+	/ENDIF			000000		21035100
2063+	/COPY QBASICS1,PBASICS1R4 Assert/This/Caller		CPY	000000		20013300

	* RPG member name : PBASICS1R4					25
	* External name : WSDL2RPG/QBASICS1(PBASICS1R4)					25
	* Last change : 08/19/09 12:46:32					25
	* Text 'description' : SrvPgm: Basic Routines - Assert/This/Caller cpybk					25

2064+	/IF NOT DEFINED(BASICS1R4) LINES EXCLUDED: 40			000000		25000100
2065+	/ENDIF			000000		25004200
2066+	*			000000		20013400
2067+	/if defined(C_COMPILER) LINES EXCLUDED: 1			000000		20013500
2068+	/endif			000000		20013700
2069+	*			000000		20013800
2070+	/ENDIF			000000		20013900
2071+	*			000000		13022600
2072+	/ENDIF			000000		13022700
2073+	*			000000		11004100
2074+	/ENDIF			000000		11004200
2075+	/COPY QBASICS1,PBASICS1R5 Objects		CPY	000000		2021600

	* RPG member name : PBASICS1R5					26
	* External name : WSDL2RPG/QBASICS1(PBASICS1R5)					26
	* Last change : 08/19/09 12:46:32					26
	* Text 'description' : SrvPgm: Basic Routines - Objects cpybk					26

2076+	/IF NOT DEFINED(BASICS1R5) LINES EXCLUDED: 225			000000		26000100
2077+	/ENDIF			000000		26022700
2078+	*			000000		2021700
2079+	/ENDIF			000000		2021800
2080+	/COPY QBASICS1,PBASICS1R2 API Error Handling		CPY	000000		1002500

	* RPG member name : PBASICS1R2					27
	* External name : WSDL2RPG/QBASICS1(PBASICS1R2)					27
	* Last change : 08/19/09 12:46:31					27
	* Text 'description' : SrvPgm: Basic Routines - API Error Handling cpybk					27

2081+	/IF NOT DEFINED(BASICS1R2) LINES EXCLUDED: 64			000000		27000100
2082+	/ENDIF			000000		27006600
2083+	/COPY QBASICS1,PBASICS1R3 Strings		CPY	000000		1002600

	* RPG member name : PBASICS1R3					28
	* External name : WSDL2RPG/QBASICS1(PBASICS1R3)					28
	* Last change : 08/19/09 12:46:32					28
	* Text 'description' : SrvPgm: Basic Routines - Strings cpybk					28

2084+	/IF NOT DEFINED(BASICS1R3)			000000		28000100

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
----- Source Specifications -----><----- Comments ----->							
.....1.....2.....3.....4.....5.....6.....7.....8.....9.....10							
LINES EXCLUDED: 349							
2085+	/ENDIF	RADDAT			000000		28035100
2086+	/COPY QBASICS1,PBASICS1R4	Assert/This		CPY	000000		1002700

* RPG member name : PBASICS1R4 *							
* External name : WSDL2RPG/QBASICS1(PBASICS1R4) *							
* Last change : 08/19/09 12:46:32 *							
* Text 'description' : SrvPgm: Basic Routines - Assert/This/Caller cpybk *							

2087+	/IF NOT DEFINED(BASICS1R4)				000000		29000100
LINES EXCLUDED: 40							
2088+	/ENDIF	RADDAT			000000		29004200
2089+	/COPY QBASICS1,PBASICS1R5	Objects		CPY	000000		1002800

* RPG member name : PBASICS1R5 *							
* External name : WSDL2RPG/QBASICS1(PBASICS1R5) *							
* Last change : 08/19/09 12:46:32 *							
* Text 'description' : SrvPgm: Basic Routines - Objects cpybk *							

2090+	/IF NOT DEFINED(BASICS1R5)				000000		30000100
LINES EXCLUDED: 225							
2091+	/ENDIF	RADDAT			000000		30022700
2092+	/COPY QBASICS1,PBASICS1R6	Commands		CPY	000000		1002900

* RPG member name : PBASICS1R6 *							
* External name : WSDL2RPG/QBASICS1(PBASICS1R6) *							
* Last change : 08/19/09 12:46:32 *							
* Text 'description' : SrvPgm: Basic Routines - Commands/Environmentcpybk *							

2093+	/IF NOT DEFINED(BASICS1R6)				000000		31000100
2094+	/DEFINE BASICS1R6				000000		31000200
2095+	*				000000		31000300
2096+	* Reference fields	RADDAT			000000		31000400
2097+	/COPY QBASICS1,FDENVVAR	RADDAT		CPY	000000		31000500

* RPG member name : FDENVVAR *							
* External name : WSDL2RPG/QBASICS1(FDENVVAR) *							
* Last change : 08/19/09 12:46:30 *							
* Text 'description' : cpybk : Environment variable *							

2098+	/IF NOT DEFINED(envVar_t)	RADDAT			000000		32000100
2099+	/DEFINE envVar_t	RADDAT			000000		32000200
2100+	* ===== *	RADDAT			000000		32000300
2101+	* Environment variable.	RADDAT			000000		32000400
2102+	* ===== *	RADDAT			000000		32000500
2103+	*	RADDAT			000000		32000600
2104+D	envVar_t DS qualified based(pDummy)	RADDAT			000000		32000700
2105+D	name 128A varying				000000		32000800
2106+D	value 1024A varying	RADDAT			000000		32000900
2107+	*	RADDAT			000000		32001000
2108+	/ENDIF	RADDAT			000000		32001100
2109+	*	RADDAT			000000		31000600
2110+	* Constants	RADDAT			000000		31000700
2111+	* f_execute: Refer to description of QCAPCMD API for details	RADDAT			000000		31000800
2112+D	cCMD_PRC_EXC_QCMDExc...	RADDAT			000000		31000900
2113+D	C const(0)	RADDAT			000000		31001000

Line	Source Specifications										<----- Comments ----->		Do	Page	Change	Src	Seq
Number	1	2	3	4	5	6	7	8	9	10			Num	Line	Date	Id	Number
2114+D	cCMD_PRC_CHK_QCMDCHK...														000000		31001100
2115+D	C				const(1)										000000		31001200
2116+D	cCMD_PRC_EXC_CMDLINE...														000000		31001300
2117+D	C				const(2)										000000		31001400
2118+D	cCMD_PRC_CHK_CMDLINE...														000000		31001500
2119+D	C				const(3)										000000		31001600
2120+D	cCMD_PRC_CHK_CL_STMT...														000000		31001700
2121+D	C				const(4)										000000		31001800
2122+D	cCMD_PRC_CHK_SEU_CL...														000000		31001900
2123+D	C				const(5)										000000		31002000
2124+D	cCMD_PRC_CHK_SEU_CMD...														000000		31002100
2125+D	C				const(6)										000000		31002200
2126+D	cCMD_PRC_CHK_SEU_BND...														000000		31002300
2127+D	C				const(7)										000000		31002400
2128+D	cCMD_PRC_CHK_UDO...														000000		31002500
2129+D	C				const(8)										000000		31002600
2130+D	cCMD_PRC_CHK_SEU_CLLE...														000000		31002700
2131+D	C				const(9)										000000		31002800
2132+ *															000000		31002900
2133+D	cCMD_PRC_PROMPT_NEVER...														000000		31003000
2134+D	C				const(0)										000000		31003100
2135+D	cCMD_PRC_PROMPT_ALWAYS...														000000		31003200
2136+D	C				const(x'00010000')			65536							000000		31003300
2137+D	cCMD_PRC_PROMPT_ON_REQUEST...														000000		31003400
2138+D	C				const(x'00020000')			131072							000000		31003500
2139+D	cCMD_PRC_PROMPT_HELP...														000000		31003600
2140+D	C				const(x'00040000')			262144							000000		31003700
2141+ *															000000		31003800
2142+ *	Executes a CL command.														000000		31003900
2143+D	f_execute...														000000		31004000
2144+D	PR	N		opdesc											000000		31004100
2145+D				extproc('BASICS1R6+											000000		31004200
2146+D				_f_execute+											000000		31004300
2147+D				')											000000		31004400
2148+D	i_cmd	16384A		const	options(*varsize)										000000		31004500
2149+D	i_typeCmdPrc	10I 0		const	options(*nopass: *omit)										000000		31004600
2150+D	o_msg				likeds(msg_t)										000000		31004700
2151+D					options(*nopass: *omit)										000000		31004800
2152+ *															000000		31004900
2153+ *	Gets an environment variable.														000000		31005000
2154+D	f_getEnvVar...														000000		31005100
2155+D	PR	1024A		varying											000000		31005200
2156+D				extproc('BASICS1R6+											000000		31005300
2157+D				_f_getEnvVar+											000000		31005400
2158+D				')											000000		31005500
2159+D	i_pEnvVar		*	const	options(*string)										000000		31005600
2160+ *															000000		31005700
2161+ *	Sets an environment variable.														000000		31005800
2162+D	f_setEnvVar...														000000		31005900
2163+D	PR	N													000000		31006000
2164+D				extproc('BASICS1R6+											000000		31006100
2165+D				_f_setEnvVar+											000000		31006200
2166+D				')											000000		31006300
2167+D	i_envVar			const	like(envVar_t.name)										000000		31006400
2168+D	i_value			const	like(envVar_t.value)										000000		31006500
2169+ *															000000		31006600

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
2170+	* Deletes an environment variable.	RADDAT			000000		31006700
2171+D	f_delEnvVar...	RADDAT			000000		31006800
2172+D	PR N	RADDAT			000000		31006900
2173+D	extproc('BASICS1R6+	RADDAT			000000		31007000
2174+D	_f_delEnvVar+	RADDAT			000000		31007100
2175+D	'')	RADDAT			000000		31007200
2176+D	i_pEnvVar * const options(*string)	RADDAT			000000		31007300
2177+ *		RADDAT			000000		31007400
2178+ *	Gets the number of environment variables.	RADDAT			000000		31007500
2179+D	f_getNumEnvVar...	RADDAT			000000		31007600
2180+D	PR 10I 0	RADDAT			000000		31007700
2181+D	extproc('BASICS1R6+	RADDAT			000000		31007800
2182+D	_f_getNumEnvVar+	RADDAT			000000		31007900
2183+D	'')	RADDAT			000000		31008000
2184+ *		RADDAT			000000		31008100
2185+ *	Gets an environment variable by its index.	RADDAT			000000		31008200
2186+D	f_getEnvVarByIndex...	RADDAT			000000		31008300
2187+D	PR likeds(envVar_t)	RADDAT			000000		31008400
2188+D	extproc('BASICS1R6+	RADDAT			000000		31008500
2189+D	_f_getEnvVarByIndex+	RADDAT			000000		31008600
2190+D	'')	RADDAT			000000		31008700
2191+D	i_index 10I 0 const	RADDAT			000000		31008800
2192+ *		RADDAT			000000		31008900
2193+ *	Waits for the specified number of seconds.	RADDAT			000000		31009000
2194+D	p_wait...	RADDAT			000000		31009100
2195+D	PR	RADDAT			000000		31009200
2196+D	extproc('BASICS1R6+	RADDAT			000000		31009300
2197+D	_p_wait+	RADDAT			000000		31009400
2198+D	'')	RADDAT			000000		31009500
2199+D	i_seconds 10I 0 const	RADDAT			000000		31009600
2200+ *		RADDAT			000000		31009700
2201+ *	-----	RADDAT			000000		31009800
2202+ *	Imported prototypes	RADDAT			000000		31009900
2203+ *	-----	RADDAT			000000		31010000
2204+ /COPY QBASICS1,PBASICS1R1	Messages	RADDAT		CPY	000000		31010100

* RPG member name : PBASICS1R1 *							
* External name : WSDL2RPG/QBASICS1(PBASICS1R1) *							
* Last change : 08/19/09 12:46:31 *							
* Text 'description' : SrvPgm: Basic Routines - Messages cpybk *							

2205+ /IF NOT DEFINED(PBASICS1R1)					000000		33000100
LINES EXCLUDED: 216							
2206+ /ENDIF		RADDAT			000000		33021800
2207+ /COPY QBASICS1,PBASICS1R2	API Error Handling	RADDAT		CPY	000000		31010200

* RPG member name : PBASICS1R2 *							
* External name : WSDL2RPG/QBASICS1(PBASICS1R2) *							
* Last change : 08/19/09 12:46:31 *							
* Text 'description' : SrvPgm: Basic Routines - API Error Handling cpybk *							

2208+ /IF NOT DEFINED(BASICS1R2)					000000		34000100
LINES EXCLUDED: 64							
2209+ /ENDIF		RADDAT			000000		34006600
2210+ *		RADDAT			000000		31010300
2211+ /ENDIF		RADDAT			000000		31010400

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
2212+	/COPY QBASICS1,PBASICS1R7 Mathematics			CPY	000000		1003000
	* * RPG member name : PBASICS1R7 * External name : WSDL2RPG/QBASICS1(PBASICS1R7) * Last change : 08/19/09 12:46:32 * Text 'description' : SrvPgm: Basic Routines - Mathematics cpybk *						35 35 35 35
2213+	/IF NOT DEFINED(BASICS1R7) LINES EXCLUDED: 137	RADDAT			000000		35000100
2214+	/ENDIF	RADDAT			000000		35013900
2215+	/COPY QBASICS1,PBASICS1R8 Handle/Memory	RADDAT		CPY	000000		1003100
	* * RPG member name : PBASICS1R8 * External name : WSDL2RPG/QBASICS1(PBASICS1R8) * Last change : 08/19/09 12:46:32 * Text 'description' : SrvPgm: Basic Routines - Handle/Memory cpybk *						36 36 36 36
2216+	/IF NOT DEFINED(BASICS1R8)	RADDAT			000000		36000100
2217+	/DEFINE BASICS1R8	RADDAT			000000		36000200
2218+	*	RADDAT			000000		36000300
2219+D	HANDLE_handle_t...	RADDAT			000000		36000400
2220+D	S 10I 0 based(pDummy)	RADDAT			000000		36000500
2221+D	HANDLE_type_t...	RADDAT			000000		36000600
2222+D	S 20A based(pDummy)	RADDAT			000000		36000700
2223+	*	RADDAT			000000		36000800
2224+	* Creates a new handle.	RADDAT			000000		36000900
2225+D	HANDLE_new...	RADDAT			000000		36001000
2226+D	PR like(HANDLE_handle_t)	RADDAT			000000		36001100
2227+D	extproc('BASICS1R8+	RADDAT			000000		36001200
2228+D	_HANDLE_new+	RADDAT			000000		36001300
2229+D	')	RADDAT			000000		36001400
2230+D	i_type value like(HANDLE_type_t)	RADDAT			000000		36001500
2231+D	i_size 10U 0 value	RADDAT			000000		36001600
2232+	*	RADDAT			000000		36001700
2233+	* Creates a NULL handle.	RADDAT			000000		36001800
2234+D	HANDLE_null...	RADDAT			000000		36001900
2235+D	PR like(HANDLE_handle_t)	RADDAT			000000		36002000
2236+D	extproc('BASICS1R8+	RADDAT			000000		36002100
2237+D	_HANDLE_null+	RADDAT			000000		36002200
2238+D	')	RADDAT			000000		36002300
2239+	*	RADDAT			000000		36002400
2240+	* Checks a handle for a NULL value.	RADDAT			000000		36002500
2241+D	HANDLE_isNull...	RADDAT			000000		36002600
2242+D	PR N	RADDAT			000000		36002700
2243+D	extproc('BASICS1R8+	RADDAT			000000		36002800
2244+D	_HANDLE_isNull+	RADDAT			000000		36002900
2245+D	')	RADDAT			000000		36003000
2246+D	i_handle value like(HANDLE_handle_t)	RADDAT			000000		36003100
2247+	*	RADDAT			000000		36003200
2248+	* Checks the type of a given handle.	RADDAT			000000		36003300
2249+D	HANDLE_isTypeOf...	RADDAT			000000		36003400
2250+D	PR N	RADDAT			000000		36003500
2251+D	extproc('BASICS1R8+	RADDAT			000000		36003600
2252+D	_HANDLE_isTypeOf+	RADDAT			000000		36003700
2253+D	')	RADDAT			000000		36003800
2254+D	i_handle value like(HANDLE_handle_t)	RADDAT			000000		36003900

Line Number	<----- Source Specifications ----->	<----- Comments ----->	Do Num	Page Line	Change Date	Src Id	Seq Number
2304+	/ENDIF	RADDAT			000000		37004200
2305+	*	RADDAT			000000		36008800
2306+	/ENDIF	RADDAT			000000		36008900
2307+	/COPY QBASICS1,PBASICS1R9	RADDAT		CPY	000000		1003200

	* RPG member name : PBASICS1R9	*					38
	* External name : WSDL2RPG/QBASICS1(PBASICS1R9)	*					38
	* Last change : 08/19/09 12:46:32	*					38
	* Text 'description' : SrvPgm: Basic Routines - Display Information	*					38

2308+	/IF NOT DEFINED(BASICS1R9)				000000		38000100
2309+	/DEFINE BASICS1R9				000000		38000200
2310+	*	RADDAT			000000		38000300
2311+	* Reference Fields	RADDAT			000000		38000400
2312+D	display_mode_t S 4A based(pDummy)	RADDAT			000000		38000500
2313+	*	RADDAT			000000		38000600
2314+	* Constants	RADDAT			000000		38000700
2315+D	cDISPLAY_MODE_DS3...	RADDAT			000000		38000800
2316+D	C const('*DS3')	RADDAT			000000		38000900
2317+D	cDISPLAY_MODE_DS4...	RADDAT			000000		38001000
2318+D	C const('*DS4')	RADDAT			000000		38001100
2319+	*	RADDAT			000000		38001200
2320+	* Returns the current display mode.	RADDAT			000000		38001300
2321+D	Display_getMode...	RADDAT			000000		38001400
2322+D	PR like(display_mode_t)	RADDAT			000000		38001500
2323+D	extproc('BASICS1R9+	RADDAT			000000		38001600
2324+D	_Display_getMode+	RADDAT			000000		38001700
2325+D	')	RADDAT			000000		38001800
2326+	*	RADDAT			000000		38001900
2327+	* Returns cTrue if the display device support 27x132 mode.	RADDAT			000000		38002000
2328+D	Display_supportsDS4...	RADDAT			000000		38002100
2329+D	PR N	RADDAT			000000		38002200
2330+D	extproc('BASICS1R9+	RADDAT			000000		38002300
2331+D	_Display_supportsDS4+	RADDAT			000000		38002400
2332+D	')	RADDAT			000000		38002500
2333+	*	RADDAT			000000		38002600
2334+	* Returns cTrue if the display device supports GUI characters.	RADDAT			000000		38002700
2335+D	Display_supportsGUI...	RADDAT			000000		38002800
2336+D	PR N	RADDAT			000000		38002900
2337+D	extproc('BASICS1R9+	RADDAT			000000		38003000
2338+D	_Display_supportsGUI+	RADDAT			000000		38003100
2339+D	')	RADDAT			000000		38003200
2340+	*	RADDAT			000000		38003300
2341+	* Returns the number of rows that can be displayed on the screen.	RADDAT			000000		38003400
2342+D	Display_getRows...	RADDAT			000000		38003500
2343+D	PR 5I 0	RADDAT			000000		38003600
2344+D	extproc('BASICS1R9+	RADDAT			000000		38003700
2345+D	_Display_getRows+	RADDAT			000000		38003800
2346+D	')	RADDAT			000000		38003900
2347+D	i_dspMode const like(display_mode_t)	RADDAT			000000		38004000
2348+D	options(*nopass)	RADDAT			000000		38004100
2349+	*	RADDAT			000000		38004200
2350+	* Returns the number of columns that can be displayed on the screen.	RADDAT			000000		38004300
2351+D	Display_getColumns...	RADDAT			000000		38004400
2352+D	PR 5I 0	RADDAT			000000		38004500
2353+D	extproc('BASICS1R9+	RADDAT			000000		38004600

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
2354+D	_Display_getColumns+	RADDAT			000000		38004700
2355+D	'')	RADDAT			000000		38004800
2356+D	i_dspMode const like(display_mode_t)	RADDAT			000000		38004900
2357+D	options(*nopass)	RADDAT			000000		38005000
2358+ *		RADDAT			000000		38005100
2359+ *	-----	RADDAT			000000		38005200
2360+ *	Deprecated procedure names	RADDAT			000000		38005300
2361+ *	-----	RADDAT			000000		38005400
2362+ *		RADDAT			000000		38005500
2363+ *	Ermittelt den aktuellen Bildschirmmodus: *DS3 oder *DS4	RADDAT			000000		38005600
2364+D	f_getDisplayMode...	RADDAT			000000		38005700
2365+D	PR like(display_mode_t)	RADDAT			000000		38005800
2366+D	extproc('BASICS1R9+	RADDAT			000000		38005900
2367+D	_Display_getMode+	RADDAT			000000		38006000
2368+D	'')	RADDAT			000000		38006100
2369+ *		RADDAT			000000		38006200
2370+ *	-----	RADDAT			000000		38006300
2371+ *	Imported prototypes	RADDAT			000000		38006400
2372+ *	-----	RADDAT			000000		38006500
2373+ /COPY QBASICS1,PBASICS1R2	API Error Handling	RADDAT		CPY	000000		38006600
-----*							
* RPG member name : PBASICS1R2							
* External name : WSDL2RPG/QBASICS1(PBASICS1R2)							
* Last change : 08/19/09 12:46:31							
* Text 'description' : SrvPgm: Basic Routines - API Error Handling cpybk							
-----*							
2374+ /IF NOT DEFINED(BASICS1R2)					000000		39000100
LINES EXCLUDED: 64							
2375+ /ENDIF		RADDAT			000000		39006600
2376+ *		RADDAT			000000		38006700
2377+ /ENDIF		RADDAT			000000		38006800
2378+ /COPY QBASICS1,PBASICS1RA	BasicList	RADDAT		CPY	000000		1003300
-----*							
* RPG member name : PBASICS1RA							
* External name : WSDL2RPG/QBASICS1(PBASICS1RA)							
* Last change : 08/19/09 12:46:31							
* Text 'description' : SrvPgm: Basic Routines - BasicList cpybk							
-----*							
2379+ /IF NOT DEFINED(BASICS1RA)		RADDAT			000000		40000100
2380+ /DEFINE BASICS1RA		RADDAT			000000		40000200
2381+ *		RADDAT			000000		40000300
2382+D	BasicList_handle_t...				000000		40000400
2383+D	S * based(pDummy)				000000		40000500
2384+D	BasicList_type_t...				000000		40000600
2385+D	S 32A based(pDummy)	RADDAT			000000		40000700
2386+ *		RADDAT			000000		40000800
2387+D	cBASICLIST_REMOVE_ALL...				000000		40000900
2388+D	C const(-1)	RADDAT			000000		40001000
2389+ *		RADDAT			000000		40001100
2390+D	cBASICLIST_TYPE_UNKNOWN...				000000		40001200
2391+D	C const('*N')				000000		40001300
2392+D	cBASICLIST_TYPE_VSTRING...				000000		40001400
2393+D	C const('*VSTRING')				000000		40001500
2394+D	cBASICLIST_TYPE_CHARACTER...				000000		40001600
2395+D	C const('*CHAR')				000000		40001700
2396+D	cBASICLIST_TYPE_INTEGER...				000000		40001800

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
2397+D	C	const('*INT')			RADDAT	000000	40001900
2398+D	cBASICLIST_TYPE_UINTEGER...					000000	40002000
2399+D	C	const('*UINT')			RADDAT	000000	40002100
2400+D	cBASICLIST_TYPE_POINTER...					000000	40002200
2401+D	C	const('*POINTER')			RADDAT	000000	40002300
2402+ *					RADDAT	000000	40002400
2403+ *	Creates a new BasicList.				RADDAT	000000	40002500
2404+D	BasicList_new...				RADDAT	000000	40002600
2405+D	PR	like(BasicList_handle_t)			RADDAT	000000	40002700
2406+D		extproc('BASICS1RA+			RADDAT	000000	40002800
2407+D		_BasicList_new+			RADDAT	000000	40002900
2408+D		')			RADDAT	000000	40003000
2409+D	i_extType	const like(BasicList_type_t)				000000	40003100
2410+D	i_increment	10U 0 const options(*nopass: *omit)				000000	40003200
2411+ *					RADDAT	000000	40003300
2412+ *	Clones a given BasicList.				RADDAT	000000	40003400
2413+D	BasicList_clone...				RADDAT	000000	40003500
2414+D	PR	like(BasicList_handle_t)			RADDAT	000000	40003600
2415+D		extproc('BASICS1RA+			RADDAT	000000	40003700
2416+D		_BasicList_clone+			RADDAT	000000	40003800
2417+D		')			RADDAT	000000	40003900
2418+D	i_pList	const like(BasicList_handle_t)			RADDAT	000000	40004000
2419+ *					RADDAT	000000	40004100
2420+ *	Returns the element from the specified position.				RADDAT	000000	40004200
2421+D	BasicList_get...				RADDAT	000000	40004300
2422+D	PR	*			RADDAT	000000	40004400
2423+D		extproc('BASICS1RA+			RADDAT	000000	40004500
2424+D		_BasicList_get+			RADDAT	000000	40004600
2425+D		')			RADDAT	000000	40004700
2426+D	i_pList	const like(BasicList_handle_t)			RADDAT	000000	40004800
2427+D	i_index	10I 0 const			RADDAT	000000	40004900
2428+D	i_pEntry	* const options(*nopass)			RADDAT	000000	40005000
2429+D	i_length	10I 0 const options(*nopass)			RADDAT	000000	40005100
2430+ *					RADDAT	000000	40005200
2431+ *	Returns the string element from the specified position.				RADDAT	000000	40005300
2432+D	BasicList_getString...				RADDAT	000000	40005400
2433+D	PR	4096A varying			RADDAT	000000	40005500
2434+D		extproc('BASICS1RA+			RADDAT	000000	40005600
2435+D		_BasicList_getString+			RADDAT	000000	40005700
2436+D		')			RADDAT	000000	40005800
2437+D	i_pList	const like(BasicList_handle_t)			RADDAT	000000	40005900
2438+D	i_index	10I 0 const			RADDAT	000000	40006000
2439+ *					RADDAT	000000	40006100
2440+ *	Returns the pointer element from the specified position.				RADDAT	000000	40006200
2441+D	BasicList_getPointer...				RADDAT	000000	40006300
2442+D	PR	*			RADDAT	000000	40006400
2443+D		extproc('BASICS1RA+			RADDAT	000000	40006500
2444+D		_BasicList_getPointer+			RADDAT	000000	40006600
2445+D		')			RADDAT	000000	40006700
2446+D	i_pList	const like(BasicList_handle_t)			RADDAT	000000	40006800
2447+D	i_index	10I 0 const			RADDAT	000000	40006900
2448+ *					RADDAT	000000	40007000
2449+ *	Returns the type of the list.				RADDAT	000000	40007100
2450+D	BasicList_getListType...					000000	40007200
2451+D	PR	like(BasicList_type_t)			RADDAT	000000	40007300
2452+D		extproc('BASICS1RA+			RADDAT	000000	40007400

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
2453+D					000000		40007500
2454+D	_BasicList_getListType+				000000		40007600
2455+D	i_pList	const like(BasicList_handle_t)			000000		40007700
2456+ *					000000		40007800
2457+ *	Returns the type of the list.				000000		40007900
2458+D	BasicList_isListTypeOf...				000000		40008000
2459+D	PR N				000000		40008100
2460+D	extproc('BASICS1RA+				000000		40008200
2461+D	_BasicList_isListTypeOf+				000000		40008300
2462+D)				000000		40008400
2463+D	i_pList	const like(BasicList_handle_t)			000000		40008500
2464+D	i_type	const like(BasicList_type_t)			000000		40008600
2465+ *					000000		40008700
2466+ *	Checks an list entry for a given type.				000000		40008800
2467+D	BasicList_isTypeOf...				000000		40008900
2468+D	PR N				000000		40009000
2469+D	extproc('BASICS1RA+				000000		40009100
2470+D	_BasicList_isTypeOf+				000000		40009200
2471+D)				000000		40009300
2472+D	i_pList	const like(BasicList_handle_t)			000000		40009400
2473+D	i_index	10I 0 const			000000		40009500
2474+D	i_type	const like(BasicList_type_t)			000000		40009600
2475+ *					000000		40009700
2476+ *	Returns the type of the specified list entry.				000000		40009800
2477+D	BasicList_getType...				000000		40009900
2478+D	PR	like(BasicList_type_t)			000000		40010000
2479+D	extproc('BASICS1RA+				000000		40010100
2480+D	_BasicList_getType+				000000		40010200
2481+D)				000000		40010300
2482+D	i_pList	const like(BasicList_handle_t)			000000		40010400
2483+D	i_index	10I 0 const			000000		40010500
2484+ *					000000		40010600
2485+ *	Returns the size of of an element.				000000		40010700
2486+D	BasicList_getSize...				000000		40010800
2487+D	PR	10I 0			000000		40010900
2488+D	extproc('BASICS1RA+				000000		40011000
2489+D	_BasicList_getSize+				000000		40011100
2490+D)				000000		40011200
2491+D	i_pList	const like(BasicList_handle_t)			000000		40011300
2492+D	i_index	10I 0 const			000000		40011400
2493+ *					000000		40011500
2494+ *	Adds an element to a given BasicList.				000000		40011600
2495+D	BasicList_add...				000000		40011700
2496+D	PR	10I 0			000000		40011800
2497+D	extproc('BASICS1RA+				000000		40011900
2498+D	_BasicList_add+				000000		40012000
2499+D)				000000		40012100
2500+D	i_pList	const like(BasicList_handle_t)			000000		40012200
2501+D	i_pEntry	* const options(*string)			000000		40012300
2502+D	i_length	10I 0 const options(*nopass: *omit)			000000		40012400
2503+D	i_type	const like(BasicList_type_t)			000000		40012500
2504+D		options(*nopass: *omit)			000000		40012600
2505+ *					000000		40012700
2506+ *	Adds a NULL value to a given BasicList.				000000		40012800
2507+D	BasicList_addNull...				000000		40012900
2508+D	PR	10I 0			000000		40013000

Line	----- Source Specifications -----><----- Comments ----->										Do	Page	Change	Src	Seq
Number	1	2	3	4	5	6	7	8	9	10	Num	Line	Date	Id	Number
2509+D					extproc('BASICS1RA+					RADDAT			000000		40013100
2510+D					_BasicList_addNull+					RADDAT			000000		40013200
2511+D					')					RADDAT			000000		40013300
2512+D	i_pList				const like(BasicList_handle_t)					RADDAT			000000		40013400
2513+ *										RADDAT			000000		40013500
2514+ *	Adds a string element to a given BasicList.												000000		40013600
2515+D	BasicList_addString...												000000		40013700
2516+D		PR		10I 0						RADDAT			000000		40013800
2517+D					extproc('BASICS1RA+					RADDAT			000000		40013900
2518+D					_BasicList_addString+					RADDAT			000000		40014000
2519+D					')					RADDAT			000000		40014100
2520+D	i_pList				const like(BasicList_handle_t)					RADDAT			000000		40014200
2521+D	i_string			4096A	const varying					RADDAT			000000		40014300
2522+ *										RADDAT			000000		40014400
2523+ *	Adds a pointer element to a given BasicList.												000000		40014500
2524+D	BasicList_addPointer...												000000		40014600
2525+D		PR		10I 0						RADDAT			000000		40014700
2526+D					extproc('BASICS1RA+					RADDAT			000000		40014800
2527+D					_BasicList_addPointer+					RADDAT			000000		40014900
2528+D					')					RADDAT			000000		40015000
2529+D	i_pList				const like(BasicList_handle_t)					RADDAT			000000		40015100
2530+D	i_pointer			*	const					RADDAT			000000		40015200
2531+ *										RADDAT			000000		40015300
2532+ *	Searches the list for a given element.												000000		40015400
2533+D	BasicList_find...												000000		40015500
2534+D		PR		10I 0						RADDAT			000000		40015600
2535+D					extproc('BASICS1RA+					RADDAT			000000		40015700
2536+D					_BasicList_find+					RADDAT			000000		40015800
2537+D					')					RADDAT			000000		40015900
2538+D	i_pList				const like(BasicList_handle_t)					RADDAT			000000		40016000
2539+D	i_pEntry			*	const options(*string)					RADDAT			000000		40016100
2540+D	i_length			10I 0	const options(*nopass)					RADDAT			000000		40016200
2541+ *										RADDAT			000000		40016300
2542+ *	Searches the list for a given string.												000000		40016400
2543+D	BasicList_findString...												000000		40016500
2544+D		PR		10I 0						RADDAT			000000		40016600
2545+D					extproc('BASICS1RA+					RADDAT			000000		40016700
2546+D					_BasicList_findString+					RADDAT			000000		40016800
2547+D					')					RADDAT			000000		40016900
2548+D	i_pList				const like(BasicList_handle_t)					RADDAT			000000		40017000
2549+D	i_string			4096A	const varying					RADDAT			000000		40017100
2550+ *										RADDAT			000000		40017200
2551+ *	Changes the element at the specified position in this list.												000000		40017300
2552+D	BasicList_set...												000000		40017400
2553+D		PR		10I 0						RADDAT			000000		40017500
2554+D					extproc('BASICS1RA+					RADDAT			000000		40017600
2555+D					_BasicList_set+					RADDAT			000000		40017700
2556+D					')					RADDAT			000000		40017800
2557+D	i_pList				const like(BasicList_handle_t)					RADDAT			000000		40017900
2558+D	i_index			10I 0	const					RADDAT			000000		40018000
2559+D	i_pEntry			*	const options(*string)					RADDAT			000000		40018100
2560+D	i_length			10I 0	const options(*nopass: *omit)					RADDAT			000000		40018200
2561+D	i_type				const like(BasicList_type_t)					RADDAT			000000		40018300
2562+D					options(*nopass: *omit)					RADDAT			000000		40018400
2563+ *										RADDAT			000000		40018500
2564+ *	Changes the element at the specified position to NULL.												000000		40018600

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
2565+D	BasicList_setNull...	RADDAT			000000		40018700
2566+D	PR 10I 0	RADDAT			000000		40018800
2567+D	extproc('BASICS1RA+	RADDAT			000000		40018900
2568+D	_BasicList_setNull+	RADDAT			000000		40019000
2569+D	')	RADDAT			000000		40019100
2570+D	i_pList const like(BasicList_handle_t)	RADDAT			000000		40019200
2571+D	i_index 10I 0 const	RADDAT			000000		40019300
2572+ *		RADDAT			000000		40019400
2573+ *	Changes the string at the specified position in this list.	RADDAT			000000		40019500
2574+D	BasicList_setString...	RADDAT			000000		40019600
2575+D	PR 10I 0	RADDAT			000000		40019700
2576+D	extproc('BASICS1RA+	RADDAT			000000		40019800
2577+D	_BasicList_setString+	RADDAT			000000		40019900
2578+D	')	RADDAT			000000		40020000
2579+D	i_pList const like(BasicList_handle_t)	RADDAT			000000		40020100
2580+D	i_index 10I 0 const	RADDAT			000000		40020200
2581+D	i_string 4096A const varying	RADDAT			000000		40020300
2582+ *		RADDAT			000000		40020400
2583+ *	Changes the pointer at the specified position in this list.	RADDAT			000000		40020500
2584+D	BasicList_setPointer...	RADDAT			000000		40020600
2585+D	PR 10I 0	RADDAT			000000		40020700
2586+D	extproc('BASICS1RA+	RADDAT			000000		40020800
2587+D	_BasicList_setPointer+	RADDAT			000000		40020900
2588+D	')	RADDAT			000000		40021000
2589+D	i_pList const like(BasicList_handle_t)	RADDAT			000000		40021100
2590+D	i_index 10I 0 const	RADDAT			000000		40021200
2591+D	i_pointer * const	RADDAT			000000		40021300
2592+ *		RADDAT			000000		40021400
2593+ *	Specifies an auto-sort function for this list.	RADDAT			000000		40021500
2594+D	BasicList_setAutoSort...	RADDAT			000000		40021600
2595+D	PR N	RADDAT			000000		40021700
2596+D	extproc('BASICS1RA+	RADDAT			000000		40021800
2597+D	_BasicList_setAutoSort+				000000		40021900
2598+D	')	RADDAT			000000		40022000
2599+D	i_pList const like(BasicList_handle_t)	RADDAT			000000		40022100
2600+D	i_pSortFnc * const procptr	RADDAT			000000		40022200
2601+ *		RADDAT			000000		40022300
2602+ *	Removes the element at the specified position in this list.	RADDAT			000000		40022400
2603+D	BasicList_remove...	RADDAT			000000		40022500
2604+D	PR 10I 0	RADDAT			000000		40022600
2605+D	extproc('BASICS1RA+	RADDAT			000000		40022700
2606+D	_BasicList_remove+	RADDAT			000000		40022800
2607+D	')	RADDAT			000000		40022900
2608+D	i_pList const like(BasicList_handle_t)	RADDAT			000000		40023000
2609+D	i_index 10I 0 const	RADDAT			000000		40023100
2610+ *		RADDAT			000000		40023200
2611+ *	Returns the number of lements in this list.	RADDAT			000000		40023300
2612+D	BasicList_getNumE...	RADDAT			000000		40023400
2613+D	PR 10I 0	RADDAT			000000		40023500
2614+D	extproc('BASICS1RA+	RADDAT			000000		40023600
2615+D	_BasicList_getNumE+	RADDAT			000000		40023700
2616+D	')	RADDAT			000000		40023800
2617+D	i_pList const like(BasicList_handle_t)	RADDAT			000000		40023900
2618+ *		RADDAT			000000		40024000
2619+ *	Sorts a list using the specified sort function.	RADDAT			000000		40024100
2620+D	BasicList_sort...	RADDAT			000000		40024200

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
2621+D	PR N	RADDAT			000000		40024300
2622+D		RADDAT			000000		40024400
2623+D	extproc('BASICS1RA+_BasicList_sort+	RADDAT			000000		40024500
2624+D	')	RADDAT			000000		40024600
2625+D	i_pList const like(BasicList_handle_t)	RADDAT			000000		40024700
2626+D	i_pSortFnc * const procptr	RADDAT			000000		40024800
2627+ *		RADDAT			000000		40024900
2628+ *	Returns cTrue if a given list is sorted.				000000		40025000
2629+D	BasicList_isSorted...				000000		40025100
2630+D	PR N	RADDAT			000000		40025200
2631+D	extproc('BASICS1RA+_BasicList_isSorted+	RADDAT			000000		40025300
2632+D	')				000000		40025400
2633+D		RADDAT			000000		40025500
2634+D	i_pList const like(BasicList_handle_t)	RADDAT			000000		40025600
2635+ *		RADDAT			000000		40025700
2636+ *	Removes all entries from a given list.	RADDAT			000000		40025800
2637+D	BasicList_clear...	RADDAT			000000		40025900
2638+D	PR 10I 0	RADDAT			000000		40026000
2639+D	extproc('BASICS1RA+_BasicList_clear+	RADDAT			000000		40026100
2640+D	')	RADDAT			000000		40026200
2641+D		RADDAT			000000		40026300
2642+D	i_pList const like(BasicList_handle_t)	RADDAT			000000		40026400
2643+ *		RADDAT			000000		40026500
2644+ *	Deletes a given BasicList.	RADDAT			000000		40026600
2645+D	BasicList_delete...	RADDAT			000000		40026700
2646+D	PR 10I 0	RADDAT			000000		40026800
2647+D	extproc('BASICS1RA+_BasicList_delete+	RADDAT			000000		40026900
2648+D	')	RADDAT			000000		40027000
2649+D		RADDAT			000000		40027100
2650+D	io_pList like(BasicList_handle_t)	RADDAT			000000		40027200
2651+ *		RADDAT			000000		40027300
2652+ *	Returns a BasicList NULL handle.	RADDAT			000000		40027400
2653+D	BasicList_null...	RADDAT			000000		40027500
2654+D	PR like(BasicList_handle_t)	RADDAT			000000		40027600
2655+D	extproc('BASICS1RA+_BasicList_null+	RADDAT			000000		40027700
2656+D	')	RADDAT			000000		40027800
2657+D		RADDAT			000000		40027900
2658+ *		RADDAT			000000		40028000
2659+ *	Checks a given BasicList handle for a NULL value.				000000		40028100
2660+D	BasicList_isNull...	RADDAT			000000		40028200
2661+D	PR N	RADDAT			000000		40028300
2662+D	extproc('BASICS1RA+_BasicList_isNull+	RADDAT			000000		40028400
2663+D	')	RADDAT			000000		40028500
2664+D		RADDAT			000000		40028600
2665+D	i_pList const like(BasicList_handle_t)	RADDAT			000000		40028700
2666+ *		RADDAT			000000		40028800
2667+ *	Checks a given BasicList entry for a NULL value.				000000		40028900
2668+D	BasicList_isNullValue...	RADDAT			000000		40029000
2669+D	PR N	RADDAT			000000		40029100
2670+D	extproc('BASICS1RA+_BasicList_isNullValue+	RADDAT			000000		40029200
2671+D	')	RADDAT			000000		40029300
2672+D		RADDAT			000000		40029400
2673+D	i_pList const like(BasicList_handle_t)	RADDAT			000000		40029500
2674+D	i_index 10I 0 const	RADDAT			000000		40029600
2675+ *		RADDAT			000000		40029700
2676+ *	-----	RADDAT			000000		40029800

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
2677+	* Imported prototypes	RADDAT			000000		40029900
2678+	* -----	RADDAT			000000		40030000
2679+	/COPY QBASICS1,PBASICS1R4			CPY	000000		40030100
	* -----						
	* RPG member name : PBASICS1R4	*					41
	* External name : WSDL2RPG/QBASICS1(PBASICS1R4)	*					41
	* Last change : 08/19/09 12:46:32	*					41
	* Text 'description' : SrvPgm: Basic Routines - Assert/This/Caller cpybk	*					41
	* -----						
2680+	/IF NOT DEFINED(BASICS1R4)				000000		41000100
	LINES EXCLUDED: 40						
2681+	/ENDIF	RADDAT			000000		41004200
2682+	* -----	RADDAT			000000		40030200
2683+	/ENDIF	RADDAT			000000		40030300
2684+	/COPY QBASICS1,PBASICS1RB			CPY	000000		1003400
	* -----						
	* RPG member name : PBASICS1RB	*					42
	* External name : WSDL2RPG/QBASICS1(PBASICS1RB)	*					42
	* Last change : 08/19/09 12:46:31	*					42
	* Text 'description' : SrvPgm: Basic Routines - BasicStack cpybk	*					42
	* -----						
2685+	/IF NOT DEFINED(BASICS1RB)	RADDAT			000000		42000100
2686+	/DEFINE BASICS1RB	RADDAT			000000		42000200
2687+	* -----	RADDAT			000000		42000300
2688+D	BasicStack_handle_t...				000000		42000400
2689+D	S				000000		42000500
2690+D	like(BasicList_handle_t)				000000		42000600
2691+	* -----						
2692+	* Creates a new BasicStack.	RADDAT			000000		42000700
2693+D	BasicStack_new...	RADDAT			000000		42000800
2694+D	PR	RADDAT			000000		42000900
2695+D	like(BasicStack_handle_t)	RADDAT			000000		42001000
2696+D	extproc('BASICS1RB+	RADDAT			000000		42001100
2697+D	_BasicStack_new+				000000		42001200
2698+D	')	RADDAT			000000		42001300
2699+D	i_increment 10U 0 const options(*nopass)	RADDAT			000000		42001400
2700+	* -----	RADDAT			000000		42001500
2701+	* Clones a given BasicStack. (not yet implemented)	RADDAT			000000		42001600
2702+D	BasicStack_clone...	RADDAT			000000		42001700
2703+D	PR	RADDAT			000000		42001800
2704+D	like(BasicStack_handle_t)	RADDAT			000000		42001900
2705+D	extproc('BASICS1RB+	RADDAT			000000		42002000
2706+D	_BasicStack_clone+	RADDAT			000000		42002100
2707+D	')	RADDAT			000000		42002200
2708+D	const like(BasicStack_handle_t)	RADDAT			000000		42002300
2709+D	i_pStack	RADDAT			000000		42002400
2710+D	const options(*nopass)	RADDAT			000000		42002500
2711+D	i_pEntry	RADDAT			000000		42002600
2712+D	i_length 10I 0 const options(*nopass)	RADDAT			000000		42002700
2713+D	const like(BasicStack_handle_t)	RADDAT			000000		42002800
2714+D	const options(*nopass)	RADDAT			000000		42002900
2715+D	const options(*nopass)	RADDAT			000000		42003000
2716+D	const options(*nopass)	RADDAT			000000		42003100
2717+	* -----	RADDAT			000000		42003200
2718+	* Removes the element at the top of the stack and returns that element.				000000		42003300
2719+D	BasicStack_pop...				000000		42003400

Line	----- Source Specifications -----><----- Comments ----->										Do	Page	Change	Src	Seq
Number	1	2	3	4	5	6	7	8	9	10	Num	Line	Date	Id	Number
2720+D		PR		*						RADDAT			000000		42003600
2721+D					extproc('BASICS1RB+					RADDAT			000000		42003700
2722+D					_BasicStack_pop+								000000		42003800
2723+D					')					RADDAT			000000		42003900
2724+D	i_pStack				const	like(BasicStack_handle_t)						000000		42004000
2725+D	i_pEntry			*	const								000000		42004100
2726+D	i_length			10I 0	const								000000		42004200
2727+ *										RADDAT			000000		42004300
2728+ *	Pushes an item onto the top of the stack.												000000		42004400
2729+D	BasicStack_push...									RADDAT			000000		42004500
2730+D		PR		10I 0						RADDAT			000000		42004600
2731+D					extproc('BASICS1RB+					RADDAT			000000		42004700
2732+D					_BasicStack_push+								000000		42004800
2733+D					')					RADDAT			000000		42004900
2734+D	i_pStack				const	like(BasicStack_handle_t)						000000		42005000
2735+D	i_pEntry			*	const	options(*string)				RADDAT			000000		42005100
2736+D	i_length			10I 0	const	options(*nopass)				RADDAT			000000		42005200
2737+ *										RADDAT			000000		42005300
2738+ *	Returns the size of of the element that is on top of this stack.												000000		42005400
2739+D	BasicStack_getSize...									RADDAT			000000		42005500
2740+D		PR		10I 0						RADDAT			000000		42005600
2741+D					extproc('BASICS1RB+					RADDAT			000000		42005700
2742+D					_BasicStack_getSize+								000000		42005800
2743+D					')					RADDAT			000000		42005900
2744+D	i_pStack				const	like(BasicStack_handle_t)						000000		42006000
2745+ *										RADDAT			000000		42006100
2746+ *	Returns the number of elements in this stack.												000000		42006200
2747+D	BasicStack_getNumE...									RADDAT			000000		42006300
2748+D		PR		10I 0						RADDAT			000000		42006400
2749+D					extproc('BASICS1RB+					RADDAT			000000		42006500
2750+D					_BasicStack_getNumE+								000000		42006600
2751+D					')					RADDAT			000000		42006700
2752+D	i_pStack				const	like(BasicStack_handle_t)						000000		42006800
2753+ *										RADDAT			000000		42006900
2754+ *	Removes all entries from a given stack.												000000		42007000
2755+D	BasicStack_clear...									RADDAT			000000		42007100
2756+D		PR		10I 0						RADDAT			000000		42007200
2757+D					extproc('BASICS1RB+					RADDAT			000000		42007300
2758+D					_BasicStack_clear+								000000		42007400
2759+D					')					RADDAT			000000		42007500
2760+D	i_pStack				const	like(BasicStack_handle_t)			RADDAT			000000		42007600
2761+ *										RADDAT			000000		42007700
2762+ *	Deletes a given BasicStack.												000000		42007800
2763+D	BasicStack_delete...												000000		42007900
2764+D		PR		10I 0						RADDAT			000000		42008000
2765+D					extproc('BASICS1RB+					RADDAT			000000		42008100
2766+D					_BasicStack_delete+								000000		42008200
2767+D					')					RADDAT			000000		42008300
2768+D	io_pStack					like(BasicStack_handle_t)			RADDAT			000000		42008400
2769+ *										RADDAT			000000		42008500
2770+ *	Returns a BasicStack NULL handle.												000000		42008600
2771+D	BasicStack_null...												000000		42008700
2772+D		PR				like(BasicStack_handle_t)						000000		42008800
2773+D					extproc('BASICS1RB+					RADDAT			000000		42008900
2774+D					_BasicStack_null+								000000		42009000
2775+D					')					RADDAT			000000		42009100

Line Number	<----- Source Specifications ----->	<----- Comments ----->	Do Num	Page Line	Change Date	Src Id	Seq Number
2776+	* -----	RADDAT			000000		42009200
2777+	* Checks a given BasicStack handle for a NULL value.				000000		42009300
2778+D	BasicStack_isNull...				000000		42009400
2779+D	PR N	RADDAT			000000		42009500
2780+D	extproc('BASICS1RB+	RADDAT			000000		42009600
2781+D	_BasicStack_isNull+				000000		42009700
2782+D	')	RADDAT			000000		42009800
2783+D	i_pStack const like(BasicStack_handle_t)				000000		42009900
2784+	* -----	RADDAT			000000		42010000
2785+	* -----	RADDAT			000000		42010100
2786+	* Imported prototypes	RADDAT			000000		42010200
2787+	* -----	RADDAT			000000		42010300
2788+	/COPY QBASICS1,PBASICS1RA BasicList			CPY	000000		42010400
	* -----						
	* RPG member name : PBASICS1RA						43
	* External name : WSDL2RPG/QBASICS1(PBASICS1RA)						43
	* Last change : 08/19/09 12:46:31						43
	* Text 'description' : SrvPgm: Basic Routines - BasicList cpybk						43
	* -----						
2789+	/IF NOT DEFINED(BASICS1RA)	RADDAT			000000		43000100
	LINES EXCLUDED: 301						
2790+	/ENDIF	RADDAT			000000		43030300
2791+	/COPY QBASICS1,PBASICS1R4 Assert/This/Caller			CPY	000000		42010500
	* -----						
	* RPG member name : PBASICS1R4						44
	* External name : WSDL2RPG/QBASICS1(PBASICS1R4)						44
	* Last change : 08/19/09 12:46:32						44
	* Text 'description' : SrvPgm: Basic Routines - Assert/This/Caller cpybk						44
	* -----						
2792+	/IF NOT DEFINED(BASICS1R4)				000000		44000100
	LINES EXCLUDED: 40						
2793+	/ENDIF	RADDAT			000000		44004200
2794+	* -----	RADDAT			000000		42010600
2795+	/ENDIF	RADDAT			000000		42010700
2796+	/COPY QBASICS1,PBASICS1RC BasicMap			CPY	000000		1003500
	* -----						
	* RPG member name : PBASICS1RC						45
	* External name : WSDL2RPG/QBASICS1(PBASICS1RC)						45
	* Last change : 08/19/09 12:46:31						45
	* Text 'description' : SrvPgm: Basic Routines - BasicMap cpybk						45
	* -----						
2797+	/IF NOT DEFINED(BASICS1RC)	RADDAT			000000		45000100
2798+	/DEFINE BASICS1RC	RADDAT			000000		45000200
2799+	* -----	RADDAT			000000		45000300
2800+D	BasicMap_handle_t...				000000		45000400
2801+D	S like(BasicList_handle_t)				000000		45000500
2802+D	based(pDummy)				000000		45000600
2803+D	BasicMap_key_t...				000000		45000700
2804+D	S 4096A varying based(pDummy)				000000		45000800
2805+	* -----	RADDAT			000000		45000900
2806+D	BasicMap_KeySet_handle_t...				000000		45001000
2807+D	S * based(pDummy)				000000		45001100
2808+	* -----	RADDAT			000000		45001200
2809+	* Creates a new BasicMap.				000000		45001300
2810+D	BasicMap_new...				000000		45001400
2811+D	PR like(BasicMap_handle_t)				000000		45001500

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
2812+D	extproc('BASICS1RC+				000000		45001600
2813+D	_BasicMap_new+				000000		45001700
2814+D)				000000		45001800
2815+D	i_increment 10U 0 const options(*nopass)				000000		45001900
2816+ *					000000		45002000
2817+ *	Clones a given BasicMap.				000000		45002100
2818+D	BasicMap_clone...				000000		45002200
2819+D	PR like(BasicMap_handle_t)				000000		45002300
2820+D	extproc('BASICS1RC+				000000		45002400
2821+D	_BasicMap_clone+				000000		45002500
2822+D)				000000		45002600
2823+D	i_pMap const like(BasicMap_handle_t)				000000		45002700
2824+ *					000000		45002800
2825+ *	Removes the mapping for this key from this map if it is present.				000000		45002900
2826+D	BasicMap_remove...				000000		45003000
2827+D	PR 10I 0				000000		45003100
2828+D	extproc('BASICS1RC+				000000		45003200
2829+D	_BasicMap_remove+				000000		45003300
2830+D)				000000		45003400
2831+D	i_pMap const like(BasicMap_handle_t)				000000		45003500
2832+D	i_key const like(BasicMap_key_t)				000000		45003600
2833+D	options(*varsize)				000000		45003700
2834+ *					000000		45003800
2835+ *	Returns true if this map contains a mapping for the specified key.				000000		45003900
2836+D	BasicMap_containsKey...				000000		45004000
2837+D	PR N				000000		45004100
2838+D	extproc('BASICS1RC+				000000		45004200
2839+D	_BasicMap_containsKey+				000000		45004300
2840+D)				000000		45004400
2841+D	i_pMap const like(BasicMap_handle_t)				000000		45004500
2842+D	i_key const like(BasicMap_key_t)				000000		45004600
2843+D	options(*varsize)				000000		45004700
2844+ *					000000		45004800
2845+ *	Returns the value to which this map maps the specified key.				000000		45004900
2846+D	BasicMap_get...				000000		45005000
2847+D	PR *				000000		45005100
2848+D	extproc('BASICS1RC+				000000		45005200
2849+D	_BasicMap_get+				000000		45005300
2850+D)				000000		45005400
2851+D	i_pMap const like(BasicMap_handle_t)				000000		45005500
2852+D	i_key const like(BasicMap_key_t)				000000		45005600
2853+D	options(*varsize)				000000		45005700
2854+D	i_pEntry * const options(*nopass)				000000		45005800
2855+D	i_length 10I 0 const options(*nopass)				000000		45005900
2856+ *					000000		45006000
2857+ *	Returns the string value to which this map maps the specified key.				000000		45006100
2858+D	BasicMap_getString...				000000		45006200
2859+D	PR 4096A				000000		45006300
2860+D	extproc('BASICS1RC+				000000		45006400
2861+D	_BasicMap_getString+				000000		45006500
2862+D)				000000		45006600
2863+D	i_pMap const like(BasicMap_handle_t)				000000		45006700
2864+D	i_key const like(BasicMap_key_t)				000000		45006800
2865+D	options(*varsize)				000000		45006900
2866+ *					000000		45007000
2867+ *	Returns the integer value to which this map maps the specified key.				000000		45007100

Line Number	<----- Source Specifications ----->	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
2868+D	BasicMap_getInteger...				RADDAT	000000	45007200
2869+D	PR 10I 0				RADDAT	000000	45007300
2870+D	extproc('BASICS1RC+				RADDAT	000000	45007400
2871+D	_BasicMap_getInteger+				RADDAT	000000	45007500
2872+D	')				RADDAT	000000	45007600
2873+D	i_pMap const like(BasicMap_handle_t)				RADDAT	000000	45007700
2874+D	i_key const like(BasicMap_key_t)				RADDAT	000000	45007800
2875+D	options(*varsize)				RADDAT	000000	45007900
2876+ *					RADDAT	000000	45008000
2877+ *	Returns the value addressed by a given index.				RADDAT	000000	45008100
2878+D	BasicMap_getValueByIndex...				RADDAT	000000	45008200
2879+D	PR *				RADDAT	000000	45008300
2880+D	extproc('BASICS1RC+				RADDAT	000000	45008400
2881+D	_BasicMap_getValueByIndex+				RADDAT	000000	45008500
2882+D	')				RADDAT	000000	45008600
2883+D	i_pMap const like(BasicMap_handle_t)				RADDAT	000000	45008700
2884+D	i_index 10I 0 const				RADDAT	000000	45008800
2885+D	i_pEntry * const options(*nopass)				RADDAT	000000	45008900
2886+D	i_length 10I 0 const options(*nopass)				RADDAT	000000	45009000
2887+ *					RADDAT	000000	45009100
2888+ *	Returns the key addressed by a given index.				RADDAT	000000	45009200
2889+D	BasicMap_getKeyByIndex...				RADDAT	000000	45009300
2890+D	PR like(BasicMap_key_t)				RADDAT	000000	45009400
2891+D	extproc('BASICS1RC+				RADDAT	000000	45009500
2892+D	_BasicMap_getKeyByIndex+				RADDAT	000000	45009600
2893+D	')				RADDAT	000000	45009700
2894+D	i_pMap const like(BasicMap_handle_t)				RADDAT	000000	45009800
2895+D	i_index 10I 0 const				RADDAT	000000	45009900
2896+ *					RADDAT	000000	45010000
2897+ *	Associates the specified value with the specified key in this map.					000000	45010100
2898+ *	If the map previously contained a mapping for this key, the old					000000	45010200
2899+ *	value is replaced by the specified value.					000000	45010300
2900+D	BasicMap_put...					000000	45010400
2901+D	PR 10I 0				RADDAT	000000	45010500
2902+D	extproc('BASICS1RC+				RADDAT	000000	45010600
2903+D	_BasicMap_put+					000000	45010700
2904+D	')				RADDAT	000000	45010800
2905+D	i_pMap const like(BasicMap_handle_t)					000000	45010900
2906+D	i_key const like(BasicMap_key_t)					000000	45011000
2907+D	options(*varsize)					000000	45011100
2908+D	i_pEntry * const options(*string)				RADDAT	000000	45011200
2909+D	i_length 10I 0 const options(*nopass)				RADDAT	000000	45011300
2910+ *					RADDAT	000000	45011400
2911+ *	Associates the specified value with the specified key in this map.				RADDAT	000000	45011500
2912+ *	If the map previously contained a mapping for this key, the old				RADDAT	000000	45011600
2913+ *	value is replaced by the specified value.				RADDAT	000000	45011700
2914+D	BasicMap_putString...				RADDAT	000000	45011800
2915+D	PR 10I 0				RADDAT	000000	45011900
2916+D	extproc('BASICS1RC+				RADDAT	000000	45012000
2917+D	_BasicMap_putString+				RADDAT	000000	45012100
2918+D	')				RADDAT	000000	45012200
2919+D	i_pMap const like(BasicMap_handle_t)				RADDAT	000000	45012300
2920+D	i_key const like(BasicMap_key_t)				RADDAT	000000	45012400
2921+D	options(*varsize)				RADDAT	000000	45012500
2922+D	i_string 4096A const varying				RADDAT	000000	45012600
2923+ *					RADDAT	000000	45012700

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
2924+	* Associates the specified value with the specified key in this map.	RADDAT			000000		45012800
2925+	* If the map previously contained a mapping for this key, the old	RADDAT			000000		45012900
2926+	* value is replaced by the specified value.	RADDAT			000000		45013000
2927+D	BasicMap_putInteger...	RADDAT			000000		45013100
2928+D	PR 10I 0	RADDAT			000000		45013200
2929+D	extproc('BASICS1RC+	RADDAT			000000		45013300
2930+D	_BasicMap_putInteger+	RADDAT			000000		45013400
2931+D	')	RADDAT			000000		45013500
2932+D	i_pMap const like(BasicMap_handle_t)	RADDAT			000000		45013600
2933+D	i_key const like(BasicMap_key_t)	RADDAT			000000		45013700
2934+D	options(*varsize)	RADDAT			000000		45013800
2935+D	i_integer 10I 0 const	RADDAT			000000		45013900
2936+	* Returns the size of of the element that is associated with this key.	RADDAT			000000		45014000
2937+	* Returns the size of of the element that is associated with this key.				000000		45014100
2938+D	BasicMap_getSize...				000000		45014200
2939+D	PR 10I 0	RADDAT			000000		45014300
2940+D	extproc('BASICS1RC+	RADDAT			000000		45014400
2941+D	_BasicMap_getSize+				000000		45014500
2942+D	')	RADDAT			000000		45014600
2943+D	i_pMap const like(BasicMap_handle_t)				000000		45014700
2944+D	i_key const like(BasicMap_key_t)				000000		45014800
2945+D	options(*varsize)				000000		45014900
2946+	* Returns the size of of the element that is addressed by a given index.	RADDAT			000000		45015000
2947+	* Returns the size of of the element that is addressed by a given index.	RADDAT			000000		45015100
2948+D	BasicMap_getSizeByIndex...	RADDAT			000000		45015200
2949+D	PR 10I 0	RADDAT			000000		45015300
2950+D	extproc('BASICS1RC+	RADDAT			000000		45015400
2951+D	_BasicMap_getSizeByIndex+	RADDAT			000000		45015500
2952+D	')	RADDAT			000000		45015600
2953+D	i_pMap const like(BasicMap_handle_t)	RADDAT			000000		45015700
2954+D	i_index 10I 0 const	RADDAT			000000		45015800
2955+	* Returns the number of key-value mappings in this map.	RADDAT			000000		45015900
2956+	* Returns the number of key-value mappings in this map.				000000		45016000
2957+D	BasicMap_getNumE...				000000		45016100
2958+D	PR 10I 0	RADDAT			000000		45016200
2959+D	extproc('BASICS1RC+	RADDAT			000000		45016300
2960+D	_BasicMap_getNumE+				000000		45016400
2961+D	')	RADDAT			000000		45016500
2962+D	i_pMap const like(BasicMap_handle_t)				000000		45016600
2963+	* Removes all entries from a given map.	RADDAT			000000		45016700
2964+	* Removes all entries from a given map.	RADDAT			000000		45016800
2965+D	BasicMap_clear...	RADDAT			000000		45016900
2966+D	PR 10I 0	RADDAT			000000		45017000
2967+D	extproc('BASICS1RC+	RADDAT			000000		45017100
2968+D	_BasicMap_clear+	RADDAT			000000		45017200
2969+D	')	RADDAT			000000		45017300
2970+D	i_pMap const like(BasicMap_handle_t)	RADDAT			000000		45017400
2971+	* Deletes a given BasicMap.	RADDAT			000000		45017500
2972+	* Deletes a given BasicMap.				000000		45017600
2973+D	BasicMap_delete...				000000		45017700
2974+D	PR 10I 0	RADDAT			000000		45017800
2975+D	extproc('BASICS1RC+	RADDAT			000000		45017900
2976+D	_BasicMap_delete+				000000		45018000
2977+D	')	RADDAT			000000		45018100
2978+D	io_pMap like(BasicMap_handle_t)				000000		45018200
2979+	*	RADDAT			000000		45018300

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
2980+	* Returns a BasicMap NULL handle.				000000		45018400
2981+D	BasicMap_null...				000000		45018500
2982+D	PR				000000		45018600
2983+D	extproc('BASICS1RC+	RADDAT			000000		45018700
2984+D	_BasicMap_null+				000000		45018800
2985+D	')	RADDAT			000000		45018900
2986+	*	RADDAT			000000		45019000
2987+	* Checks a given BasicMap handle for a NULL value.				000000		45019100
2988+D	BasicMap_isNull...				000000		45019200
2989+D	PR				000000		45019300
2990+D	extproc('BASICS1RC+	RADDAT			000000		45019400
2991+D	_BasicMap_isNull+				000000		45019500
2992+D	')	RADDAT			000000		45019600
2993+D	i_pMap	const like(BasicMap_handle_t)			000000		45019700
2994+	*	RADDAT			000000		45019800
2995+	* Produces a list of keys from a given map.	RADDAT			000000		45019900
2996+D	BasicMap_KeySet_new...				000000		45020000
2997+D	PR				000000		45020100
2998+D	extproc('BASICS1RC+	RADDAT			000000		45020200
2999+D	_BasicMap_KeySet_new+				000000		45020300
3000+D	')	RADDAT			000000		45020400
3001+D	i_pMap	const like(BasicMap_handle_t)			000000		45020500
3002+	*	RADDAT			000000		45020600
3003+	* Returns the first entry of a given key set.				000000		45020700
3004+D	BasicMap_KeySet_getFirst...				000000		45020800
3005+D	PR				000000		45020900
3006+D	extproc('BASICS1RC+	RADDAT			000000		45021000
3007+D	_BasicMap_KeySet_getFirst+				000000		45021100
3008+D	')	RADDAT			000000		45021200
3009+D	i_pKeySet	const like(BasicMap_KeySet_handle_t)			000000		45021300
3010+	*	RADDAT			000000		45021400
3011+	* Returns the next entry of a given key set.				000000		45021500
3012+D	BasicMap_KeySet_getNext...				000000		45021600
3013+D	PR				000000		45021700
3014+D	extproc('BASICS1RC+	RADDAT			000000		45021800
3015+D	_BasicMap_KeySet_getNext+				000000		45021900
3016+D	')	RADDAT			000000		45022000
3017+D	i_pKeySet	const like(BasicMap_KeySet_handle_t)			000000		45022100
3018+	*	RADDAT			000000		45022200
3019+	* Deletes the key list associated to a given map.	RADDAT			000000		45022300
3020+D	BasicMap_KeySet_delete...				000000		45022400
3021+D	PR				000000		45022500
3022+D	extproc('BASICS1RC+	RADDAT			000000		45022600
3023+D	_BasicMap_KeySet_delete+				000000		45022700
3024+D	')	RADDAT			000000		45022800
3025+D	io_pKeySet	like(BasicMap_KeySet_handle_t)			000000		45022900
3026+	*	RADDAT			000000		45023000
3027+	* -----	RADDAT			000000		45023100
3028+	* Imported prototypes	RADDAT			000000		45023200
3029+	* -----	RADDAT			000000		45023300
3030+	/COPY QBASICS1,PBASICS1RA	BasicList			000000	CPY	45023400
	* -----						
	* RPG member name :	PBASICS1RA					46
	* External name :	WSDL2RPG/QBASICS1(PBASICS1RA)					46
	* Last change :	08/19/09 12:46:31					46
	* Text 'description' :	SrvPgm: Basic Routines - BasicList			cpybk		46

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
3031+	/IF NOT DEFINED(BASICS1RA) LINES EXCLUDED: 301	RADDAT			000000		46000100
3032+	/ENDIF	RADDAT			000000		46030300
3033+	/COPY QBASICS1,PBASICS1R4 Assert/This/Caller	CPY			000000		45023500

* RPG member name : PBASICS1R4 *							
* External name : WSDL2RPG/QBASICS1(PBASICS1R4) *							
* Last change : 08/19/09 12:46:32 *							
* Text 'description' : SrvPgm: Basic Routines - Assert/This/Caller cpybk *							

3034+	/IF NOT DEFINED(BASICS1R4) LINES EXCLUDED: 40				000000		47000100
3035+	/ENDIF	RADDAT			000000		47004200
3036+	*	RADDAT			000000		45023600
3037+	/ENDIF	RADDAT			000000		45023700
3038+	/COPY QBASICS1,PBASICS1RD Url	CPY			000000		1003600

* RPG member name : PBASICS1RD *							
* External name : WSDL2RPG/QBASICS1(PBASICS1RD) *							
* Last change : 08/19/09 12:46:31 *							
* Text 'description' : SrvPgm: Basic Routines - URL cpybk *							

3039+	/IF NOT DEFINED(BASICS1RD)	RADDAT			000000		48000100
3040+	/DEFINE BASICS1RD	RADDAT			000000		48000200
3041+	*	RADDAT			000000		48000300
3042+	/COPY QBASICS1,FDURL	CPY			000000		48000400

* RPG member name : FDURL *							
* External name : WSDL2RPG/QBASICS1(FDURL) *							
* Last change : 08/19/09 12:46:31 *							
* Text 'description' : cpybk : URL *							

3043+	/IF NOT DEFINED(url_t)	RADDAT			000000		49000100
3044+	/DEFINE url_t	RADDAT			000000		49000200
3045+	*	RADDAT			000000		49000300
3046+	* URL	RADDAT			000000		49000400
3047+D	url_t S 446A varying	RADDAT			000000		49000500
3048+	*	RADDAT			000000		49000600
3049+	/ENDIF	RADDAT			000000		49000700
3050+	*	RADDAT			000000		48000500
3051+	* Error ocdes	RADDAT			000000		48000600
3052+D	cURL_ERROR_EMPTY_URL...	RADDAT			000000		48000700
3053+D	C -1	RADDAT			000000		48000800
3054+D	cURL_ERROR_NO_PROTOCOL...	RADDAT			000000		48000900
3055+D	C -2	RADDAT			000000		48001000
3056+D	cURL_ERROR_UNSUPPORTED_PROTOCOL...	RADDAT			000000		48001100
3057+D	C -3	RADDAT			000000		48001200
3058+D	cURL_ERROR_INVALID_PORT_NUMBER...	RADDAT			000000		48001300
3059+D	C -4	RADDAT			000000		48001400
3060+D	cURL_ERROR_UNSUPPORTED_OPERATION...	RADDAT			000000		48001500
3061+D	C -5	RADDAT			000000		48001600
3062+D	cURL_ERROR_INVALID_OBJECT_DATA...	RADDAT			000000		48001700
3063+D	C -6	RADDAT			000000		48001800
3064+	*	RADDAT			000000		48001900
3065+	* Creates a new URL.	RADDAT			000000		48002000

Line Number	----- Source Specifications -----><----- Comments ----->						Do Num	Page Line	Change Date	Src Id	Seq Number
3066+D	Url_new...					RADDAT			000000		48002100
3067+D		PR			like(url_t)	RADDAT			000000		48002200
3068+D					extproc('BASICS1RD+	RADDAT			000000		48002300
3069+D					_Url_new+	RADDAT			000000		48002400
3070+D					')	RADDAT			000000		48002500
3071+D	i_url				const like(url_t)	RADDAT			000000		48002600
3072+D	o_rc		10I	0	options(*nopass: *omit)	RADDAT			000000		48002700
3073+ *						RADDAT			000000		48002800
3074+ *	Returns the protocol of a given URL.					RADDAT			000000		48002900
3075+D	Url_getProtocol...					RADDAT			000000		48003000
3076+D		PR			like(url_t)	RADDAT			000000		48003100
3077+D					extproc('BASICS1RD+	RADDAT			000000		48003200
3078+D					_Url_getProtocol+	RADDAT			000000		48003300
3079+D					')	RADDAT			000000		48003400
3080+D	i_url				const like(url_t)	RADDAT			000000		48003500
3081+D	o_rc		10I	0	options(*nopass: *omit)	RADDAT			000000		48003600
3082+ *						RADDAT			000000		48003700
3083+ *	Sets the protocol of a given URL.					RADDAT			000000		48003800
3084+D	Url_setProtocol...					RADDAT			000000		48003900
3085+D		PR		N		RADDAT			000000		48004000
3086+D					extproc('BASICS1RD+	RADDAT			000000		48004100
3087+D					_Url_setProtocol+	RADDAT			000000		48004200
3088+D					')	RADDAT			000000		48004300
3089+D	io_url				like(url_t)	RADDAT			000000		48004400
3090+D	i_protocol				const like(url_t) options(*varsize)	RADDAT			000000		48004500
3091+D	o_rc		10I	0	options(*nopass: *omit)	RADDAT			000000		48004600
3092+ *						RADDAT			000000		48004700
3093+ *	Returns the host name of a given URL.					RADDAT			000000		48004800
3094+D	Url_getHost...					RADDAT			000000		48004900
3095+D		PR			like(url_t)	RADDAT			000000		48005000
3096+D					extproc('BASICS1RD+	RADDAT			000000		48005100
3097+D					_Url_getHost+	RADDAT			000000		48005200
3098+D					')	RADDAT			000000		48005300
3099+D	i_url				const like(url_t)	RADDAT			000000		48005400
3100+D	o_rc		10I	0	options(*nopass: *omit)	RADDAT			000000		48005500
3101+ *						RADDAT			000000		48005600
3102+ *	Sets the host name of a given URL.					RADDAT			000000		48005700
3103+D	Url_setHost...					RADDAT			000000		48005800
3104+D		PR		N		RADDAT			000000		48005900
3105+D					extproc('BASICS1RD+	RADDAT			000000		48006000
3106+D					_Url_setHost+	RADDAT			000000		48006100
3107+D					')	RADDAT			000000		48006200
3108+D	io_url				like(url_t)	RADDAT			000000		48006300
3109+D	i_host				const like(url_t) options(*varsize)	RADDAT			000000		48006400
3110+D	o_rc		10I	0	options(*nopass: *omit)	RADDAT			000000		48006500
3111+ *						RADDAT			000000		48006600
3112+ *	Returns the port of a given URL.					RADDAT			000000		48006700
3113+D	Url_getPort...					RADDAT			000000		48006800
3114+D		PR		10I	0	RADDAT			000000		48006900
3115+D					extproc('BASICS1RD+	RADDAT			000000		48007000
3116+D					_Url_getPort+	RADDAT			000000		48007100
3117+D					')	RADDAT			000000		48007200
3118+D	i_url				const like(url_t)	RADDAT			000000		48007300
3119+D	o_rc		10I	0	options(*nopass: *omit)	RADDAT			000000		48007400
3120+ *						RADDAT			000000		48007500
3121+ *	Sets the port number of a given URL.					RADDAT			000000		48007600

Line	<----- Source Specifications -----><----- Comments ----->										Do	Page	Change	Src	Seq
Number	1	2	3	4	5	6	7	8	9	10	Num	Line	Date	Id	Number
3122+D	Url_setPort...												000000		48007700
3123+D		PR		N									000000		48007800
3124+D					extproc('BASICS1RD+								000000		48007900
3125+D					_Url_setPort+								000000		48008000
3126+D					')								000000		48008100
3127+D	io_url					like(url_t)							000000		48008200
3128+D	i_port			10I 0	const								000000		48008300
3129+D	o_rc			10I 0		options(*nopass: *omit)							000000		48008400
3130+ *													000000		48008500
3131+ *	Returns the path name of a given URL.												000000		48008600
3132+D	Url_getPath...												000000		48008700
3133+D		PR				like(url_t)							000000		48008800
3134+D					extproc('BASICS1RD+								000000		48008900
3135+D					_Url_getPath+								000000		48009000
3136+D					')								000000		48009100
3137+D	i_url				const	like(url_t)							000000		48009200
3138+D	o_rc			10I 0		options(*nopass: *omit)							000000		48009300
3139+ *													000000		48009400
3140+ *	Sets the path of a given URL.												000000		48009500
3141+D	Url_setPath...												000000		48009600
3142+D		PR		N									000000		48009700
3143+D					extproc('BASICS1RD+								000000		48009800
3144+D					_Url_setPath+								000000		48009900
3145+D					')								000000		48010000
3146+D	io_url					like(url_t)							000000		48010100
3147+D	i_path				const	like(url_t) options(*varsize)							000000		48010200
3148+D	o_rc			10I 0		options(*nopass: *omit)							000000		48010300
3149+ *													000000		48010400
3150+ *	Returns the file name of a given URL.												000000		48010500
3151+D	Url_getFile...												000000		48010600
3152+D		PR				like(url_t)							000000		48010700
3153+D					extproc('BASICS1RD+								000000		48010800
3154+D					_Url_getFile+								000000		48010900
3155+D					')								000000		48011000
3156+D	i_url				const	like(url_t)							000000		48011100
3157+D	o_rc			10I 0		options(*nopass: *omit)							000000		48011200
3158+ *													000000		48011300
3159+ *	Sets the file of a given URL.												000000		48011400
3160+D	Url_setFile...												000000		48011500
3161+D		PR		N									000000		48011600
3162+D					extproc('BASICS1RD+								000000		48011700
3163+D					_Url_setFile+								000000		48011800
3164+D					')								000000		48011900
3165+D	io_url					like(url_t)							000000		48012000
3166+D	i_file				const	like(url_t) options(*varsize)							000000		48012100
3167+D	o_rc			10I 0		options(*nopass: *omit)							000000		48012200
3168+ *													000000		48012300
3169+ *	Returns the query part of a given URL.												000000		48012400
3170+D	Url_getQuery...												000000		48012500
3171+D		PR				like(url_t)							000000		48012600
3172+D					extproc('BASICS1RD+								000000		48012700
3173+D					_Url_getQuery+								000000		48012800
3174+D					')								000000		48012900
3175+D	i_url				const	like(url_t)							000000		48013000
3176+D	o_rc			10I 0		options(*nopass: *omit)							000000		48013100
3177+ *													000000		48013200

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
3178+	* Sets the query part of a given URL.	RADDAT			000000		48013300
3179+D	Url_setQuery...	RADDAT			000000		48013400
3180+D	PR N	RADDAT			000000		48013500
3181+D	extproc('BASICS1RD+	RADDAT			000000		48013600
3182+D	_Url_setQuery+	RADDAT			000000		48013700
3183+D	')	RADDAT			000000		48013800
3184+D	io_url like(url_t)	RADDAT			000000		48013900
3185+D	i_query const like(url_t) options(*varsize)	RADDAT			000000		48014000
3186+D	o_rc 10I 0 options(*nopass: *omit)	RADDAT			000000		48014100
3187+	* Returns the string representation of a given URL.	RADDAT			000000		48014200
3188+	* Returns the string representation of a given URL.	RADDAT			000000		48014300
3189+D	Url_toString...	RADDAT			000000		48014400
3190+D	PR 512A varying	RADDAT			000000		48014500
3191+D	extproc('BASICS1RD+	RADDAT			000000		48014600
3192+D	_Url_toString+	RADDAT			000000		48014700
3193+D	')	RADDAT			000000		48014800
3194+D	i_url const like(url_t)	RADDAT			000000		48014900
3195+D	o_rc 10I 0 options(*nopass: *omit)	RADDAT			000000		48015000
3196+	* Returns the library, file and member of a URL of type 'mbr'.	RADDAT			000000		48015100
3197+	* Returns the library, file and member of a URL of type 'mbr'.	RADDAT			000000		48015200
3198+D	Url_getMbr...	RADDAT			000000		48015300
3199+D	PR N	RADDAT			000000		48015400
3200+D	extproc('BASICS1RD+	RADDAT			000000		48015500
3201+D	_Url_getMbr+	RADDAT			000000		48015600
3202+D	')	RADDAT			000000		48015700
3203+D	i_url const like(url_t)	RADDAT			000000		48015800
3204+D	o_file 10A options(*nopass: *omit)	RADDAT			000000		48015900
3205+D	o_lib 10A options(*nopass: *omit)	RADDAT			000000		48016000
3206+D	o_mbr 10A options(*nopass: *omit)	RADDAT			000000		48016100
3207+D	o_rc 10I 0 options(*nopass: *omit)	RADDAT			000000		48016200
3208+	* -----	RADDAT			000000		48016300
3209+	* -----	RADDAT			000000		48016400
3210+	* Imported prototypes	RADDAT			000000		48016500
3211+	* -----	RADDAT			000000		48016600
3212+	* -----	RADDAT			000000		48016700
3213+	/ENDIF	RADDAT			000000		48016800
3214+	/COPY QBASICS1,PBASICS1RE Date and Time	RADDAT		CPY	000000		1003700
	* -----						
	* RPG member name : PBASICS1RE						50
	* External name : WSDL2RPG/QBASICS1(PBASICS1RE)						50
	* Last change : 08/19/09 12:46:31						50
	* Text 'description' : SrvPgm: Basic Routines - Date and Time cpybk						50
	* -----						
3215+	/IF NOT DEFINED(BASICS1RE)	RADDAT			000000		50000100
3216+	/DEFINE BASICS1RE	RADDAT			000000		50000200
3217+	*	RADDAT			000000		50000300
3218+	* Produces a date from a given year, month and day.	RADDAT			000000		50000400
3219+D	f_dateSerial...	RADDAT			000000		50000500
3220+D	PR D	RADDAT			000000		50000600
3221+D	extproc('BASICS1RE+	RADDAT			000000		50000700
3222+D	_f_dateSerial+	RADDAT			000000		50000800
3223+D	')	RADDAT			000000		50000900
3224+D	i_year 10I 0 const	RADDAT			000000		50001000
3225+D	i_month 10I 0 const	RADDAT			000000		50001100
3226+D	i_day 10I 0 const	RADDAT			000000		50001200
3227+	*	RADDAT			000000		50001300

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
3228+	*	RADDAT			000000		50001400
3229+	* Imported prototypes	RADDAT			000000		50001500
3230+	*	RADDAT			000000		50001600
3231+	*	RADDAT			000000		50001700
3232+	/ENDIF	RADDAT			000000		50001800
3233+	*	RADDAT			000000		1003800
3234+	/ENDIF	RADDAT			000000		1003900
3235	/COPY QRPGLSRC,EXPAT_H	RPG defs for calling eXpat		CPY	000000		000328
-----*							
*	RPG member name : EXPAT_H						51
*	External name : LIBHTTP/QRPGLSRC(EXPAT_H)						51
*	Last change : 08/25/08 09:59:46						51
*	Text 'description' : RPG defs for calling eXpat						51
-----*							
3236+/*-		+			000000		51000100
3237+*	Copyright (c) 2004-2008 Scott C. Klement	+			000000		51000200
3238+*	All rights reserved.	+			000000		51000300
3239+*		+			000000		51000400
3240+*	Redistribution and use in source and binary forms, with or without	+			000000		51000500
3241+*	modification, are permitted provided that the following conditions	+			000000		51000600
3242+*	are met:	+			000000		51000700
3243+*	1. Redistributions of source code must retain the above copyright	+			000000		51000800
3244+*	notice, this list of conditions and the following disclaimer.	+			000000		51000900
3245+*	2. Redistributions in binary form must reproduce the above copyright	+			000000		51001000
3246+*	notice, this list of conditions and the following disclaimer in the	+			000000		51001100
3247+*	documentation and/or other materials provided with the distribution.	+			000000		51001200
3248+*		+			000000		51001300
3249+*	THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND	+			000000		51001400
3250+*	ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE	+			000000		51001500
3251+*	IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE	+			000000		51001600
3252+*	ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE	+			000000		51001700
3253+*	FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL	+			000000		51001800
3254+*	DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS	+			000000		51001900
3255+*	OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)	+			000000		51002000
3256+*	HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT	+			000000		51002100
3257+*	LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY	+			000000		51002200
3258+*	OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF	+			000000		51002300
3259+*	SUCH DAMAGE.	+			000000		51002400
3260+*/		+			000000		51002500
3261+*					000000		51002600
3262+*					000000		51002700
3263+*	This contains RPG definitions needed for calling routines from				000000		51002800
3264+*	the eXpat XML parser. This source member is distributed with				000000		51002900
3265+*	HTTPAPI for your convenience.				000000		51003000
3266+*					000000		51003100
3267+*					000000		51003200
3268+*/if defined(EXPAT_H)					000000		51003300
	LINES EXCLUDED: 1						
3269+*/endif					000000		51003500
3270+*/define EXPAT_H					000000		51003600
3271+*					000000		51003700
3272+*	*****				000000		51003800
3273+*	Define this if EXPAT was compiled to output UTF-16,				000000		51003900
3274+*	or undefine it if Expat was compiled to output UTF-8				000000		51004000
3275+*	*****				000000		51004100
3276+*/define XML_UNICODE					000000		51004200

Line	Source Specifications	Comments	Do	Page	Change	Src	Seq
Number	1	2	3	4	5	6	7
3277+						000000	51004300
3278+	*****					000000	51004400
3279+	* Status flags					000000	51004500
3280+	*****					000000	51004600
3281+D	XML_STATUS_OK...					000000	51004700
3282+D	C	1				000000	51004800
3283+D	XML_STATUS_ERROR...					000000	51004900
3284+D	C	0				000000	51005000
3285+						000000	51005100
3286+	*****					000000	51005200
3287+	* Constantst that identify the various encodings					000000	51005300
3288+	*****					000000	51005400
3289+	/if defined(XML_UNICODE)					000000	51005500
3290+D	XML_ENC_UTF8...					000000	51005600
3291+D	C	u'005500540046002d00380000'				000000	51005700
3292+D	XML_ENC_UTF16...					000000	51005800
3293+D	C	u'005500540046002d003100360000'				000000	51005900
3294+D	XML_ENC_ISO8859_1...					000000	51006000
3295+D	C	u'00490053004f002d00380038-				000000	51006100
3296+D		00350039002d00310000'				000000	51006200
3297+D	XML_ENC_US_ASCII...					000000	51006300
3298+D	C	u'00550053002d004100530043-				000000	51006400
3299+D		004900490000'				000000	51006500
3300+	/else					000000	51006600
	LINES EXCLUDED: 8						
3301+	/endif					000000	51007500
3302+						000000	51007600
3303+	*-----					000000	51007700
3304+	* XML_Parser: opaque pointer to the XML Parser state information					000000	51007800
3305+	* that's used internally by eXpat					000000	51007900
3306+	*-----					000000	51008000
3307+D	XML_Parser s * based(Template)					000000	51008100
3308+						000000	51008200
3309+	*-----					000000	51008300
3310+	* Data structure to store XML Encoding Info					000000	51008400
3311+	*					000000	51008500
3312+	* typedef struct {					000000	51008600
3313+	* int map[256];					000000	51008700
3314+	* void *data;					000000	51008800
3315+	* int (XMLCALL *convert)(void *data, const char *s);					000000	51008900
3316+	* int (XMLCALL *release)(void *data);					000000	51009000
3317+	* } XML_Encoding;					000000	51009100
3318+	*					000000	51009200
3319+	*					000000	51009300
3320+	* map = map from the encoding to UTF-8. Each item in the map					000000	51009400
3321+	* corresponds to a code point of the encoding. map[0] is					000000	51009500
3322+	* codepoint 0, map[255] is codepoint 255, etc. When a					000000	51009600
3323+	* value is set to -1, it means that value is invalid as					000000	51009700
3324+	* an initial byte in a multibyte sequence. When it's					000000	51009800
3325+	* -n, then n is the number of bytes in a sequence, and					000000	51009900
3326+	* conversion is accomplished by calling the convert()					000000	51010000
3327+	* function.					000000	51010100
3328+	*					000000	51010200
3329+	* data = user-defined data that's passed to each call to					000000	51010300
3330+	* convert() and release()					000000	51010400
3331+	*					000000	51010500

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
3332+	* convert = procedure to call when converting multi-byte				000000		51010600
3333+	* characters. May be *NULL if no multibyte				000000		51010700
3334+	* chars are required.				000000		51010800
3335+	*				000000		51010900
3336+	* D myConvert PR 10I 0				000000		51011000
3337+	* D data * value				000000		51011100
3338+	* D string * value				000000		51011200
3339+	*				000000		51011300
3340+	* release = procedure that's called when the data pointer				000000		51011400
3341+	* is to be released. May be *NULL if no data				000000		51011500
3342+	* needs releasing.				000000		51011600
3343+	*				000000		51011700
3344+	* D myRelease PR				000000		51011800
3345+	* D data * value				000000		51011900
3346+	*				000000		51012000
3347+	*-----				000000		51012100
3348+D	XML_Encoding ds based(Template)				000000		51012200
3349+D	map 10I 0 dim(256)				000000		51012300
3350+D	data *				000000		51012400
3351+D	convert * procptr				000000		51012500
3352+D	release * procptr				000000		51012600
3353+	*				000000		51012700
3354+	*-----				000000		51012800
3355+	* XML_ParserCreate(): Creates a new XML_Parser object.				000000		51012900
3356+	*				000000		51013000
3357+	* encoding = (input) null-terminated name of encoding to use				000000		51013100
3358+	* or *OMIT to let Expat try to figure it out				000000		51013200
3359+	*				000000		51013300
3360+	* returns an XML_Parser, or *NULL upon failure.				000000		51013400
3361+	*-----				000000		51013500
3362+	/if defined(XML_UNICODE)				000000		51013600
3363+D	XML_ParserCreate...				000000		51013700
3364+D	PR ExtProc('XML_ParserCreate')				000000		51013800
3365+D	like(XML_Parser)				000000		51013900
3366+D	encoding 100C const options(*varsize:*omit)				000000		51014000
3367+	/else				000000		51014100
	LINES EXCLUDED: 4						
3368+	/endif				000000		51014600
3369+					000000		51014700
3370+	*-----				000000		51014800
3371+	* XML_SetUserData(): Set user data to be sent to callbacks				000000		51014900
3372+	*				000000		51015000
3373+	* parser = (input) XML Parser to set user data for				000000		51015100
3374+	* userData = (input) pointer to user data that is to be				000000		51015200
3375+	* passed to each call-back procedure				000000		51015300
3376+	*-----				000000		51015400
3377+D	XML_SetUserData...				000000		51015500
3378+D	PR ExtProc('XML_SetUserData')				000000		51015600
3379+D	parser like(XML_Parser) value				000000		51015700
3380+D	userData * value				000000		51015800
3381+					000000		51015900
3382+	*-----				000000		51016000
3383+	* XML_SetElementHandler(): Set functions to be called for				000000		51016100
3384+	* each starting & ending element				000000		51016200
3385+	*				000000		51016300
3386+	* parser = (input) XML_Parser to set functions for				000000		51016400

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
3387+	* start = (input) pointer to call-back procedure for				000000		51016500
3388+	* start tags				000000		51016600
3389+	* end = (input) pointer to call-back procedure for				000000		51016700
3390+	* end tags				000000		51016800
3391+					000000		51016900
3392+	* D myStartHandler PR 10I 0				000000		51017000
3393+	* D userData * value				000000		51017100
3394+	* D elementName * value				000000		51017200
3395+	* D atts * dim(1000) options(*varsize)				000000		51017300
3396+					000000		51017400
3397+	userData = the pointer you supplied with XML_SetUserData				000000		51017500
3398+	elementName = null-terminated string w/the element name				000000		51017600
3399+	atts = null-terminated array of pointers, each				000000		51017700
3400+	points to a null-terminated string containing				000000		51017800
3401+	an attribute name or value (they alternate)				000000		51017900
3402+					000000		51018000
3403+	* D myEndHandler PR 10I 0				000000		51018100
3404+	* D userData * value				000000		51018200
3405+	* D elementName * value				000000		51018300
3406+					000000		51018400
3407+	userData = the pointer you supplied with XML_SetUserData				000000		51018500
3408+	elementName = null-terminated string w/the element name				000000		51018600
3409+					000000		51018700
3410+	* Note: This is a shortcut for calling XML_SetStartElementHandler()				000000		51018800
3411+	and XML_SetEndElementHandler()				000000		51018900
3412+	*-----				000000		51019000
3413+D	XML_SetElementHandler...				000000		51019100
3414+D	PR ExtProc('XML_SetElementHandler')				000000		51019200
3415+D	parser like(XML_Parser) value				000000		51019300
3416+D	start * procptr value				000000		51019400
3417+D	end * procptr value				000000		51019500
3418+					000000		51019600
3419+D	XML_SetStartElementHandler...				000000		51019700
3420+D	PR ExtProc('XML_SetStartElement-				000000		51019800
3421+D	Handler')				000000		51019900
3422+D	parser like(XML_Parser) value				000000		51020000
3423+D	start * procptr value				000000		51020100
3424+					000000		51020200
3425+D	XML_SetEndElementHandler...				000000		51020300
3426+D	PR ExtProc('XML_SetEndElement-				000000		51020400
3427+D	Handler')				000000		51020500
3428+D	parser like(XML_Parser) value				000000		51020600
3429+D	end * procptr value				000000		51020700
3430+					000000		51020800
3431+	*-----				000000		51020900
3432+	* XML_SetCharacterDataHandler(): Set handler for character data				000000		51021000
3433+					000000		51021100
3434+	parser = (input) XML_Parser to set functions for				000000		51021200
3435+	charhndl = (input) pointer to call-back procedure for				000000		51021300
3436+	character data				000000		51021400
3437+					000000		51021500
3438+	* D myCharHandler PR				000000		51021600
3439+	* D userData * value				000000		51021700
3440+	* D string 65535A const options(*varsize)				000000		51021800
3441+	* D len 10I 0 value				000000		51021900
3442+					000000		51022000

Line Number	<----- Source Specifications ----->	<----- Comments ----->	Do	Page	Change	Src	Seq
			Num	Line	Date	Id	Number
3443+	*				000000		51022100
3444+	* Note: The string is options(*varsize). You MUST use the				000000		51022200
3445+	"len" variable to determine how much was passed, and				000000		51022300
3446+	use caution not to use data past that length.				000000		51022400
3447+	*				000000		51022500
3448+	* Note: This function might be called multiple times for a single				000000		51022600
3449+	contiguous character data block. You should concatenate				000000		51022700
3450+	the data together.				000000		51022800
3451+	*-----				000000		51022900
3452+D	XML_SetCharacterDataHandler...				000000		51023000
3453+D	PR	ExtProc('XML_SetCharacterData-			000000		51023100
3454+D		Handler')			000000		51023200
3455+D	parser	like(XML_Parser) value			000000		51023300
3456+D	charhndl	* procptr value			000000		51023400
3457+					000000		51023500
3458+	*-----				000000		51023600
3459+	* XML_Parse(): Parse XML data in stream				000000		51023700
3460+	*				000000		51023800
3461+	parser = (input) XML_Parser object that will do the parsing				000000		51023900
3462+	s = (input) next chunk of data from XML stream				000000		51024000
3463+	len = (input) length of XML data in 's'				000000		51024100
3464+	isFinal = (input) flag that specifies whether this is the				000000		51024200
3465+	final chunk of the stream. Set to 1 if it				000000		51024300
3466+	is, or 0 otherwise.				000000		51024400
3467+	*				000000		51024500
3468+	Returns XML_STATUS_OK if successful				000000		51024600
3469+	or XML_STATUS_ERROR upon failure				000000		51024700
3470+	*-----				000000		51024800
3471+D	XML_Parse...				000000		51024900
3472+D	PR	10I 0 ExtProc('XML_Parse')			000000		51025000
3473+D	parser	like(XML_Parser) value			000000		51025100
3474+D	s	* value options(*string)			000000		51025200
3475+D	len	10I 0 value			000000		51025300
3476+D	isFinal	10I 0 value			000000		51025400
3477+					000000		51025500
3478+	*-----				000000		51025600
3479+	* XML_GetErrorCode(): Retrieves the error number of the				000000		51025700
3480+	last error that occurred while parsing.				000000		51025800
3481+	*				000000		51025900
3482+	parser = (input) XML_Parser object to retrieve error from				000000		51026000
3483+	*				000000		51026100
3484+	returns the error number.				000000		51026200
3485+	*-----				000000		51026300
3486+D	XML_GetErrorCode...				000000		51026400
3487+D	PR	10I 0 ExtProc('XML_GetErrorCode')			000000		51026500
3488+D	parser	like(XML_Parser) value			000000		51026600
3489+					000000		51026700
3490+	*-----				000000		51026800
3491+	* XML_ErrorString(): Returns the human-readable error message				000000		51026900
3492+	that corresponds to an error number.				000000		51027000
3493+	*				000000		51027100
3494+	code = (input) error number to get message for.				000000		51027200
3495+	*				000000		51027300
3496+	returns a null-terminated error string.				000000		51027400
3497+	*-----				000000		51027500
3498+D	XML_ErrorString...				000000		51027600

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
3499+D	PR	* ExtProc('XML_ErrorString')			000000		51027700
3500+D	code	10I 0 value			000000		51027800
3501+					000000		51027900
3502+	*-----				000000		51028000
3503+	* XML_GetCurrentLineNumber():	Get the line number of the XML			000000		51028100
3504+	* document that's currently being parsed.				000000		51028200
3505+					000000		51028300
3506+	* parser = (input) XML_Parser	to retrieve line number from			000000		51028400
3507+					000000		51028500
3508+	* returns the line number.				000000		51028600
3509+	*-----				000000		51028700
3510+D	XML_GetCurrentLineNumber...				000000		51028800
3511+D	PR	10I 0 ExtProc('XML_GetCurrentLine-			000000		51028900
3512+D		Number')			000000		51029000
3513+D	parser	like(XML_Parser) value			000000		51029100
3514+					000000		51029200
3515+	*-----				000000		51029300
3516+	* XML_GetCurrentColumnNumber():	Return the offset from the start			000000		51029400
3517+	* of the current line.				000000		51029500
3518+					000000		51029600
3519+	* parser = (input) XML_Parser	to retrieve line number from			000000		51029700
3520+					000000		51029800
3521+	* returns the column number.				000000		51029900
3522+	*-----				000000		51030000
3523+D	XML_GetCurrentColumnNumber...				000000		51030100
3524+D	PR	10I 0 ExtProc('XML_GetCurrentColumn-			000000		51030200
3525+D		Number')			000000		51030300
3526+D	parser	like(XML_Parser) value			000000		51030400
3527+					000000		51030500
3528+	*-----				000000		51030600
3529+	* XML_ParserFree():	Destroys & frees up resources used by an			000000		51030700
3530+	* XML_Parser object.				000000		51030800
3531+					000000		51030900
3532+	* parser = (input) XML_Parser	to destroy			000000		51031000
3533+					000000		51031100
3534+	*-----				000000		51031200
3535+D	XML_ParserFree...				000000		51031300
3536+D	PR	ExtProc('XML_ParserFree')			000000		51031400
3537+D	parser	like(XML_Parser) value			000000		51031500
3538+					000000		51031600
3539+					000000		51031700
3540+	*-----				000000		51031800
3541+	* XML_SetUnknownEncodingHandler():	Set a handler to handle any			000000		51031900
3542+	* encodings that eXpat is not familiar with.				000000		51032000
3543+					000000		51032100
3544+	* parser = (input) XML_Parser	that will use the encoding			000000		51032200
3545+	* enchandler = (input) procedure	to call to initialize the			000000		51032300
3546+	* XML_Encoding structure for an unknown				000000		51032400
3547+	* encoding.				000000		51032500
3548+	* info = (input) XML_Encoding	structure that will be set			000000		51032600
3549+					000000		51032700
3550+	* An encoding handler should be prototyped as:				000000		51032800
3551+					000000		51032900
3552+	* D myHandler PR	10I 0			000000		51033000
3553+	* D handlerData	* value			000000		51033100
3554+	* D name	* value			000000		51033200

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
3555+	* D info	likes(XML_Encoding)			000000		51033300
3556+	*				000000		51033400
3557+	*				000000		51033500
3558+D	XML_SetUnknownEncodingHandler...				000000		51033600
3559+D	PR	ExtProc('XML_SetUnknownEncoding-			000000		51033700
3560+D		Handler')			000000		51033800
3561+D	parser	like(XML_Parser) value			000000		51033900
3562+D	enhandler	* procptr value			000000		51034000
3563+D	info	likes(XML_Encoding)			000000		51034100
3564+					000000		51034200
3565+					000000		51034300
3566+	*				000000		51034400
3567+	* XML_ParserCreateNS(): Creates a new XML_Parser object				000000		51034500
3568+	* that uses namespace processing				000000		51034600
3569+	*				000000		51034700
3570+	* This causes the namespace qualifier to be expanded to the				000000		51034800
3571+	* full URL for a namespace. You can specify the character that				000000		51034900
3572+	* separates the namespace from the URL.				000000		51035000
3573+	*				000000		51035100
3574+	* example:				000000		51035200
3575+	* <ns:Result xmlns:ns="http://www.example.com/ns">				000000		51035300
3576+	* is expanded to:				000000		51035400
3577+	* http://www.example.com/ns<sep>Result				000000		51035500
3578+	*				000000		51035600
3579+	* encoding = (input) name of encoding to use.				000000		51035700
3580+	* or *OMIT if no special encoding is used				000000		51035800
3581+	* nsSeparator = (input) character that will separate the				000000		51035900
3582+	* URL from the element name. (or x'00' for none)				000000		51036000
3583+	*				000000		51036100
3584+	* returns an XML_Parser, or *NULL upon failure.				000000		51036200
3585+	*				000000		51036300
3586+	/if defined(XML_UNICODE)				000000		51036400
3587+D	XML_ParserCreateNS...				000000		51036500
3588+D	PR	ExtProc(*CWIDEN			000000		51036600
3589+D		: 'XML_ParserCreateNS')			000000		51036700
3590+D		like(XML_Parser)			000000		51036800
3591+D	encoding	100C const options(*varsize:*omit)			000000		51036900
3592+D	nsSeparator	1A value			000000		51037000
3593+	/else				000000		51037100
3594+	LINES EXCLUDED: 6						
3594+	/endif				000000		51037800
3595+					000000		51037900
3596+					000000		51038000
3597+	*				000000		51038100
3598+	* XML_SetStartNamespaceDeclHandler(): When namespace parsing is				000000		51038200
3599+	* turned on, this calls the handler once for each declared				000000		51038300
3600+	* namespace.				000000		51038400
3601+	*				000000		51038500
3602+	* parser = (input) XML Parser to set namespace in				000000		51038600
3603+	* start = (input) pointer to handler procedure in this format:				000000		51038700
3604+	* D handler PR				000000		51038800
3605+	* D userdata	* value			000000		51038900
3606+	* D prefix	* value			000000		51039000
3607+	* D uri	* value			000000		51039100
3608+	*				000000		51039200
3609+D	XML_SetStartNamespaceDeclHandler...				000000		51039300

Line Number	<----- Source Specifications ----->	<----- Comments ----->	Do Num	Page Line	Change Date	Src Id	Seq Number
3610+D	PR	extproc('XML_SetStartNamespace-			000000		51039400
3611+D		DeclHandler')			000000		51039500
3612+D	parser	like(XML_Parser) value			000000		51039600
3613+D	start	* procptr value			000000		51039700
3614+					000000		51039800
3615+					000000		51039900
3616+	*-----				000000		51040000
3617+	* XML_SetEndNamespaceDeclHandler(): Sets a handler for the end				000000		51040100
3618+	* of a namespace				000000		51040200
3619+	*				000000		51040300
3620+	* parser = (input) XML parser to set the handler for				000000		51040400
3621+	* end = (input) procedure to call at the end. It must be				000000		51040500
3622+	* in this format:				000000		51040600
3623+	*				000000		51040700
3624+	* D handler PR				000000		51040800
3625+	* D userdata * value				000000		51040900
3626+	* D prefix * value				000000		51041000
3627+	*-----				000000		51041100
3628+D	XML_SetEndNamespaceDeclHandler...				000000		51041200
3629+D	PR	extproc('XML_SetEndNamespace-			000000		51041300
3630+D		DeclHandler')			000000		51041400
3631+D	parser	like(XML_Parser) value			000000		51041500
3632+D	end	* procptr value			000000		51041600
3633	/COPY QRPGLSRC,HTTPAPI_H HTTP-API Service Program				000000	CPY	000329
	*-----						
	* RPG member name : HTTPAPI_H						52
	* External name : DTECHTEST2/QRPGLSRC(HTTPAPI_H)						52
	* Last change : 01/09/08 14:57:15						52
	* Text 'description' : HTTP-API header member						52
	*-----						
3634+/*-					000000		52000100
3635+	* Copyright (c) 2001-2007 Scott C. Klement				000000		52000200
3636+	* All rights reserved.				000000		52000300
3637+	*				000000		52000400
3638+	* Redistribution and use in source and binary forms, with or without				000000		52000500
3639+	* modification, are permitted provided that the following conditions				000000		52000600
3640+	* are met:				000000		52000700
3641+	* 1. Redistributions of source code must retain the above copyright				000000		52000800
3642+	* notice, this list of conditions and the following disclaimer.				000000		52000900
3643+	* 2. Redistributions in binary form must reproduce the above copyright				000000		52001000
3644+	* notice, this list of conditions and the following disclaimer in the				000000		52001100
3645+	* documentation and/or other materials provided with the distribution.				000000		52001200
3646+	*				000000		52001300
3647+	* THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS 'AS IS' AND				070828		52001400
3648+	* ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE				000000		52001500
3649+	* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPO				000000		52001600
3650+	* ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE				000000		52001700
3651+	* FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTI				000000		52001800
3652+	* DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS				000000		52001900
3653+	* OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)				000000		52002000
3654+	* HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRI				000000		52002100
3655+	* LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WA				000000		52002200
3656+	* OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF				000000		52002300
3657+	* SUCH DAMAGE.				000000		52002400
3658+	*				000000		52002500
3659+*/					000000		52002600

Line Number	<----- Source Specifications ----->	<----- Comments ----->	Do Num	Page Line	Change Date	Src Id	Seq Number
3660+					000000		52002700
3661+	/if defined(HTTPAPI_H) LINES EXCLUDED: 1				000000		52002800
3662+	/endif				000000		52003000
3663+					000000		52003100
3664+D	HTTPAPI_VERSION...				000000		52003200
3665+D	C CONST('1.21')				071001		52003300
3666+D	HTTPAPI_RELDATE...				000000		52003400
3667+D	C CONST('2007-10-01')				071001		52003500
3668+					070628		52003600
3669+	/copy LIBHTTP/qrpglesrc,config_h				070509		52003700

	* RPG member name : CONFIG_H						53
	* External name : LIBHTTP/QRPGLSRC(CONFIG_H)						53
	* Last change : 08/25/08 10:03:08						53
	* Text 'description' : HTTP-API compile-time configuration						53

3670+***	If you do not want SSL support, comment out the line below.				000000		53000100
3671+***	(You _must_ do this if you're running V4R4 or earlier)				000000		53000200
3672+					000000		53000300
3673+D	define HAVE_SSLAPI				000000		53000400
3674+					000000		53000500
3675+***	define this if your RPG compiler supports 64-bit integers				000000		53000600
3676+***	(they were introduced in V4R4)				000000		53000700
3677+***					000000		53000800
3678+					000000		53000900
3679+D	define HAVE_INT64				000000		53001000
3680+					000000		53001100
3681+***	define this if your RPG compiler supports				000000		53001200
3682+***	options(*SRCSTMT: *NODEBUGIO: *NOSHOWCPY)				000000		53001300
3683+***	(they were introduced in V4R4, but can be enabled as far				000000		53001400
3684+***	back as V3R2 using PTFs)				000000		53001500
3685+					000000		53001600
3686+D	define HAVE_SRCSTMT_NODEBUGIO				000000		53001700
3687+					000000		53001800
3688+***	In V4R5 it's possible to enable GSKit for SSL via				000000		53001900
3689+***	PTFs. However, some functionality wasn't yet available				000000		53002000
3690+***	in that release. Define this if you need SSL to be limited				000000		53002100
3691+***	to what's available on a V4R5 system. (This is ignored				000000		53002200
3692+***	when SSL is disabled.)				000000		53002300
3693+					000000		53002400
3694+D	undefine V4R5_GSKIT				000000		53002500
3695+					000000		53002600
3696+***	This is the default timeout value (in seconds) that HTTPAPI				000000		53002700
3697+***	uses if a timeout value isn't specified by the calling				000000		53002800
3698+***	program:				000000		53002900
3699+					000000		53003000
3700+D	HTTP_TIMEOUT C CONST(60)				000000		53003100
3701+					000000		53003200
3702+***	This is the 'User-Agent' name that is reported by this API				000000		53003300
3703+***	to the web servers if you don't specify it explcitley when				000000		53003400
3704+***	calling the routines.				000000		53003500
3705+					000000		53003600
3706+D	HTTP_USERAGENT C CONST('http-api/1.23')				000000		53003700
3707+					000000		53003800
3708+***	This is the 'Content-Type' that is reported by this API				000000		53003900

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
3709+***	to the web servers if you don't specify it explicitly when				000000		53004000
3710+***	calling the POST routines. (the GET routines, by default,				000000		53004100
3711+***	do not specify a content-type.)				000000		53004200
3712+					000000		53004300
3713+D	HTTP_CONTTYPE C CONST('text/xml')				000000		53004400
3714+					000000		53004500
3715+***	The original implementation of HTTPAPI used tables				000000		53004600
3716+***	(*TBL objects) to translate from ASCII to EBCDIC. These can				000000		53004700
3717+***	be enabled/set by defining the HTTP_USE_TABLES condition,				000000		53004800
3718+***	and then setting the table names. They can be overridden				000000		53004900
3719+***	at runtime by calling HTTP_setTables().				000000		53005000
3720+***					000000		53005100
3721+***	In the current implementation, we prefer that you use CCSIDs				000000		53005200
3722+***	instead of tables. When HTTP_USE_TABLES is not defined,				000000		53005300
3723+***	the HTTP_EBCDIC and HTTP_ASCII constants represent the				000000		53005400
3724+***	default CCSIDs for ASCII<-->EBCDIC translation. They can be				000000		53005500
3725+***	overridden at runtime by calling HTTP_setCCSIDs()				000000		53005600
3726+***					000000		53005700
3727+					000000		53005800
3728+	/undefine HTTP_USE_TABLES				000000		53005900
3729+	/if defined(HTTP_USE_TABLES)				000000		53006000
	LINES EXCLUDED: 2						
3730+	/else				000000		53006300
3731+D	HTTP_EBCDIC C CONST(0)				000000		53006400
3732+D	HTTP_ASCII C CONST(819)				000000		53006500
3733+	/endif				000000		53006600
3734+					000000		53006700
3735+***	This is the codepage or CCSID assigned to downloaded stream				000000		53006800
3736+***	files by default. (Note: HTTPAPI does not convert the file				000000		53006900
3737+***	to this CCSID, it just assigns this number to the file's				000000		53007000
3738+***	attributes.)				000000		53007100
3739+***					000000		53007200
3740+***	Whether this is treated as a CCSID or a codepage depends on				000000		53007300
3741+***	the value of the HTTP USE CCSID condition, below.				000000		53007400
3742+***					000000		53007500
3743+***	Can be overridden at runtime by calling HTTP_SetFileCCSID()				000000		53007600
3744+					000000		53007700
3745+D	HTTP_CCSID C CONST(819)				000000		53007800
3746+					000000		53007900
3747+***	Starting in V5R1, a full CCSID is available in the IFS				000000		53008000
3748+***	instead of a codepage. When this is defined, CCSID support				000000		53008100
3749+***	will be used instead of codepages				000000		53008200
3750+					000000		53008300
3751+D	define HTTP_USE_CCSID				000000		53008400
3752+					000000		53008500
3753+***	This is the file mode used when creating files in the IFS.				000000		53008600
3754+***	(Caution: This mode is given in DECIMAL, not octal!)				000000		53008700
3755+***	Octal 666 = Decimal 438 (RW-RW-RW-)				000000		53008800
3756+***	Octal 644 = Decimal 420 (RW-R--R--)				000000		53008900
3757+***	Octal 777 = Decimal 511 (RWXRWXRWX)				000000		53009000
3758+***	Octal 755 = Decimal 493 (RWXR-XR-X)				000000		53009100
3759+					000000		53009200
3760+D	HTTP_IFSMODE C CONST(511)				000000		53009300
3761+					000000		53009400
3762+***	HTTPAPI normally uses non-blocking sockets to ensure that				000000		53009500
3763+***	the session never "hangs". However, during the connection				000000		53009600

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
3764+***	progress, this can mean that information gets lost.				000000		53009700
3765+***					000000		53009800
3766+***	Define this to wait until after the connection is established				000000		53009900
3767+***	before switching the socket to non-blocking.				000000		53010000
3768+D/define	HTTP_BLOCK_ON_CONNECT				000000		53010100
3769+					000000		53010200
3770+***					000000		53010300
3771+***	This determines whether Cookies are turned *ON or *OFF				000000		53010400
3772+***	by default.				000000		53010500
3773+***					000000		53010600
3774+DHTTP_COOKIE_DEFAULT...					000000		53010700
3775+D	C	CONST(*ON)			000000		53010800
3776+					000000		53010900
3777+***	This changes whether debugging is on or off by default.				000000		53011000
3778+***	You can override this at runtime by calling the http_debug()				000000		53011100
3779+***	procedure.				000000		53011200
3780+D/undefine	DEBUG				000000		53011300
3781+DHTTP_DEBUG_FILE	s	500A	varying		000000		53011400
3782+D		inz('/tmp/httpapi_debug.txt')			000000		53011500
3783+					000000		53011600
3784+***	This changes how memory is allocated. If defined, HTTPAPI				000000		53011700
3785+***	will allocate memory in TERASPACE, thus allowing for very				000000		53011800
3786+***	large allocations.				000000		53011900
3787+D/undefine	USE_TS_MALLOC64				000000		53012000
3788+D/undefine	USE_TS_MALLOC64				000000		53012100
3789+***	With both turned off (default) up to 16 MB per allocation.				000000		53012200
3790+***	WARNING: This is currently experimental!! If you have				000000		53012300
3791+***	problems, make sure TERASPACE is undefined.				000000		53012400
3792+D/undefine	TERASPACE				000000		53012500
3793+D/undefine	USE_TS_MALLOC64				000000		53012600
3794+					000000		53012700
3795+***	This allows access to V5R3 (or higher) functions in the				000000		53012800
3796+***	SSL environment. Only define this if HTTPAPI will be used				000000		53012900
3797+***	on V5R3 or later.				000000		53013000
3798+D/define	V5R3_GSKIT				000000		53013100
3799+					000000		52003800
3800+	*****				000000		52003900
3801+	** procedure prototypes				000000		52004000
3802+	*****				000000		52004100
3803+					000000		52004200
3804+	*+++++				000000		52004300
3805+	* http_get(): Retrieve an HTTP document				000000		52004400
3806+	* http_url_get(): Retrieve an HTTP document				000000		52004500
3807+	*				000000		52004600
3808+	* peURL = url to grab (i.e. http://www.blah.com/dir/file.txt)				000000		52004700
3809+	* peFilename = filename in IFS to save response into				000000		52004800
3810+	* peTimeout = (optional) give up if no data is received for				000000		52004900
3811+	* this many seconds.				000000		52005000
3812+	* peModTime = (optional) only get file if it was changed since				000000		52005100
3813+	* this timestamp.				000000		52005200
3814+	* peContentType = (optional) content type to supply (mainly				000000		52005300
3815+	* useful when talking to CGI scripts.) To supply the				000000		52005400
3816+	* default value for this parm, you can supply the				000000		52005500
3817+	* HTTP_CONTTYPE constant.				000000		52005600
3818+	* peUserAgent = (optional) This specifies the name & version				000000		52005700
3819+	* of your HTTP client to the server. The server uses				000000		52005800

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
3820+	*	it for statistics and sometimes to restrict pages			000000		52005900
3821+	*	so that they're "only for Internet Explorer."			000000		52006000
3822+	*	peSOAPAction = (optional) string used to specify the action			000000		52006100
3823+	*	taken by some SOAP applications.			000000		52006200
3824+	*				000000		52006300
3825+	*	Returns -1 = internal error (check HTTP_ERROR)			000000		52006400
3826+	*	0 = timeout while receiving data or connecting			000000		52006500
3827+	*	1 = file retrieved successfully			000000		52006600
3828+	*	> 1 = HTTP response code indicating server's error reply			000000		52006700
3829+	*****				000000		52006800
3830+D	http_get PR 10I 0	extproc('HTTP_URL_GET')			000000		52006900
3831+D	peURL 32767A	varying const options(*varsize)			000000		52007000
3832+D	peFilename 32767A	varying const options(*varsize)			000000		52007100
3833+D	peTimeout 10I 0	value options(*nopass)			000000		52007200
3834+D	peUserAgent 64A	const options(*nopass:*omit)			000000		52007300
3835+D	peModTime Z	const options(*nopass:*omit)			000000		52007400
3836+D	peContentType 64A	const options(*nopass:*omit)			000000		52007500
3837+D	peSOAPAction 64A	const options(*nopass:*omit)			000000		52007600
3838+D	http_url_get PR 10I 0				000000		52007700
3839+D	peURL 32767A	varying const options(*varsize)			000000		52007800
3840+D	peFilename 32767A	varying const options(*varsize)			000000		52007900
3841+D	peTimeout 10I 0	value options(*nopass)			000000		52008000
3842+D	peUserAgent 64A	const options(*nopass:*omit)			000000		52008100
3843+D	peModTime Z	const options(*nopass:*omit)			000000		52008200
3844+D	peContentType 64A	const options(*nopass:*omit)			000000		52008300
3845+D	peSOAPAction 64A	const options(*nopass:*omit)			000000		52008400
3846+					000000		52008500
3847+					000000		52008600
3848+	*****				000000		52008700
3849+	*	http_url_post(): Post data to CGI script and get document			000000		52008800
3850+	*				000000		52008900
3851+	*	peURL = url to post to (http://www.blah.com/cgi-bin/etc)			000000		52009000
3852+	*	pePostData = pointer to data to post to CGI script.			000000		52009100
3853+	*	pePostDataLen = length of data to post to CGI script.			000000		52009200
3854+	*	peFileName = Filename in IFS to save response into			000000		52009300
3855+	*	peTimeout = (optional) give up if no data is received for			000000		52009400
3856+	*	this many seconds.			000000		52009500
3857+	*	peUserAgent = (optional) User-Agent string passed to the			000000		52009600
3858+	*	server. Pass the named constant HTTP_USERAGENT			000000		52009700
3859+	*	if you want to get the default value.			000000		52009800
3860+	*	peContentType = (optional) content type to supply (mainly			000000		52009900
3861+	*	useful when talking to CGI scripts)			000000		52010000
3862+	*	peSOAPAction = (optional) string used to specify the action			000000		52010100
3863+	*	taken by some SOAP applications.			000000		52010200
3864+	*				000000		52010300
3865+	*	Returns -1 = internal error (check HTTP_ERROR)			000000		52010400
3866+	*	0 = timeout while receiving data or connecting			000000		52010500
3867+	*	1 = file retrieved successfully			000000		52010600
3868+	*	> 1 = HTTP response code indicating server's error reply			000000		52010700
3869+	*****				000000		52010800
3870+D	http_post PR 10I 0	EXTPROC('HTTP_URL_POST')			000000		52010900
3871+D	peURL 32767A	varying const options(*varsize)			000000		52011000
3872+D	pePostData *	value			000000		52011100
3873+D	pePostDataLen 10I 0	value			000000		52011200
3874+D	peFilename 32767A	varying const options(*varsize)			000000		52011300
3875+D	peTimeout 10I 0	value options(*nopass)			000000		52011400

Line Number	<----- Source Specifications ----->	<----- Comments ----->	Do Num	Page Line	Change Date	Src Id	Seq Number
3876+D	peUserAgent 64A const options(*nopass:*omit)				000000		52011500
3877+D	peContentType 64A const options(*nopass:*omit)				000000		52011600
3878+D	peSOAPAction 64A const options(*nopass:*omit)				000000		52011700
3879+D	http_url_post PR 10I 0				000000		52011800
3880+D	peURL 32767A varying const options(*varsize)				000000		52011900
3881+D	pePostData * value				000000		52012000
3882+D	pePostDataLen 10I 0 value				000000		52012100
3883+D	peFilename 32767A varying const options(*varsize)				000000		52012200
3884+D	peTimeout 10I 0 value options(*nopass)				000000		52012300
3885+D	peUserAgent 64A const options(*nopass:*omit)				000000		52012400
3886+D	peContentType 64A const options(*nopass:*omit)				000000		52012500
3887+D	peSOAPAction 64A const options(*nopass:*omit)				000000		52012600
3888+					000000		52012700
3889+					000000		52012800
3890+	*+++++				000000		52012900
3891+	* http_url_get_raw(): Retrieve an HTTP document (in raw mode)				000000		52013000
3892+	*				000000		52013100
3893+	* peURL = url to grab (i.e. http://www.blah.com/dir/file.txt)				000000		52013200
3894+	* peFD = FD to pass back to peProc				000000		52013300
3895+	* peProc = procedure to call each time data is received.				000000		52013400
3896+	* peTimeout = (optional) give up if no data is received for				000000		52013500
3897+	* this many seconds.				000000		52013600
3898+	* peUserAgent = (optional) User-Agent string passed to the				000000		52013700
3899+	* server. Pass the named constant HTTP_USERAGENT				000000		52013800
3900+	* if you want to get the default value.				000000		52013900
3901+	* peModTime = (optiona) only get file if it was changed since				000000		52014000
3902+	* this timestamp.				000000		52014100
3903+	* peContentType = (optional) content type to supply (mainly				000000		52014200
3904+	* useful when talking to CGI scripts)				000000		52014300
3905+	* peSOAPAction = (optional) string used to specify the action				000000		52014400
3906+	* taken by some SOAP applications.				000000		52014500
3907+	*				000000		52014600
3908+	* Returns (same as http_url_get)				000000		52014700
3909+	*+++++				000000		52014800
3910+D	http_url_get_raw...				000000		52014900
3911+D	PR 10I 0				000000		52015000
3912+D	peURL 32767A varying const options(*varsize)				000000		52015100
3913+D	peFD 10I 0 value				000000		52015200
3914+D	peProc * value procptr				000000		52015300
3915+D	peTimeout 10I 0 value options(*nopass)				000000		52015400
3916+D	peUserAgent 64A const options(*nopass:*omit)				000000		52015500
3917+D	peModTime Z const options(*nopass:*omit)				000000		52015600
3918+D	peContentType 64A const options(*nopass:*omit)				000000		52015700
3919+D	peSOAPAction 64A const options(*nopass:*omit)				000000		52015800
3920+					000000		52015900
3921+					000000		52016000
3922+	*+++++				000000		52016100
3923+	* http_url_post_raw(): Post data to CGI script and get document				000000		52016200
3924+	*				000000		52016300
3925+	* peURL = url to post to (http://www.blah.com/cgi-bin/etc)				000000		52016400
3926+	* pePostData = pointer to data to post to CGI script.				000000		52016500
3927+	* pePostDataLen = length of data to post to CGI script.				000000		52016600
3928+	* peFD = FD to pass back to peProc				000000		52016700
3929+	* peProc = procedure to call each time data is received.				000000		52016800
3930+	* peTimeout = (optional) give up if no data is received for				000000		52016900
3931+	* this many seconds.				000000		52017000

Line Number	<----- Source Specifications -----><----- Comments ----->	Do	Page	Change	Src	Seq
		Num	Line	Date	Id	Number
3932+	* peUserAgent = (optional) User-Agent string passed to the			000000		52017100
3933+	* server. Pass the named constant HTTP_USERAGENT			000000		52017200
3934+	* if you want to get the default value.			000000		52017300
3935+	* peContentType = (optional) content type to supply (mainly			000000		52017400
3936+	* useful when talking to CGI scripts)			000000		52017500
3937+	* peSOAPAction = (optional) string used to specify the action			000000		52017600
3938+	* taken by some SOAP applications.			000000		52017700
3939+	*			000000		52017800
3940+	* Returns (same as http_url_post)			000000		52017900
3941+	*+++++			000000		52018000
3942+D	http_url_post_raw...			000000		52018100
3943+D	PR 10I 0			000000		52018200
3944+D	peURL 32767A varying const options(*varsize)			000000		52018300
3945+D	pePostData * value			000000		52018400
3946+D	pePostDataLen 10I 0 value			000000		52018500
3947+D	peFD 10I 0 value			000000		52018600
3948+D	peProc * value procptr			000000		52018700
3949+D	peTimeout 10I 0 value options(*nopass)			000000		52018800
3950+D	peUserAgent 64A const options(*nopass:*omit)			000000		52018900
3951+D	peContentType 64A const options(*nopass:*omit)			000000		52019000
3952+D	peSOAPAction 64A const options(*nopass:*omit)			000000		52019100
3953+	*			000000		52019200
3954+	*			000000		52019300
3955+	*+++++			000000		52019400
3956+	* http_ParseURL(): Parse URL into it's component parts			000000		52019500
3957+	*			000000		52019600
3958+	* Breaks a uniform resource locator (URL) into it's component			000000		52019700
3959+	* pieces for use with the http: or https: protocols. (would also			000000		52019800
3960+	* work for FTP with minor tweaks)			000000		52019900
3961+	*			000000		52020000
3962+	* peURL = URL that needs to be parsed.			000000		52020100
3963+	* peService = service name from URL (i.e. http or https)			000000		52020200
3964+	* peUserName = user name given, or *blanks			000000		52020300
3965+	* pePassword = password given, or *blanks			000000		52020400
3966+	* peHost = hostname given in URL. (could be domain name or IP)			000000		52020500
3967+	* pePort = port number to connect to, if specified, otherwise 0.			000000		52020600
3968+	* pePath = remaining path/request for server.			000000		52020700
3969+	*			000000		52020800
3970+	* returns -1 upon failure, or 0 upon success			000000		52020900
3971+	*+++++			000000		52021000
3972+D	http_ParseURL PR 10I 0			000000		52021100
3973+D	peURL 256A const			000000		52021200
3974+D	peService 32A			000000		52021300
3975+D	peUserName 32A			000000		52021400
3976+D	pePassword 32A			000000		52021500
3977+D	peHost 256A			000000		52021600
3978+D	pePort 10I 0			000000		52021700
3979+D	pePath 256A			000000		52021800
3980+	*			000000		52021900
3981+	*			000000		52022000
3982+	*+++++			000000		52022100
3983+	* http_build_sockaddr(): Build a socket address structure for a host			000000		52022200
3984+	*			000000		52022300
3985+	* peHost = hostname to build sockaddr_in for			000000		52022400
3986+	* peService = service name (or port) to build sockaddr_in for			000000		52022500
3987+	* peForcePort = numeric port to force entry to, overrides peService			000000		52022600

Line Number	<----- Source Specifications ----->	<----- Comments ----->	Do Num	Page Line	Change Date	Src Id	Seq Number
3988+	* peSockAddr = pointer to a location to place a sockaddr_in into.				000000		52022700
3989+	* (if *NULL, memory will be allocated, otherwise it will				000000		52022800
3990+	* be re-alloc'ed)				000000		52022900
3991+	*				000000		52023000
3992+	* returns -1 upon failure, 0 upon success				000000		52023100
3993+	*+++++				000000		52023200
3994+D	http_build_sockaddr...				000000		52023300
3995+D	PR 10I 0				000000		52023400
3996+D	peHost 256A const				000000		52023500
3997+D	peService 32A const				000000		52023600
3998+D	peForcePort 10I 0 value				000000		52023700
3999+D	peSockAddr *				000000		52023800
4000+					000000		52023900
4001+	*+++++				000000		52024000
4002+	* http_close(): close HTTP connection				000000		52024100
4003+	*				000000		52024200
4004+	* peSock = socket to close				000000		52024300
4005+	* peComm = comm driver opened with http_select_commdriver()				000000		52024400
4006+	*				000000		52024500
4007+	* returns -1 upon failure, or 0 upon success				000000		52024600
4008+	*+++++				000000		52024700
4009+D	http_close PR 10I 0				000000		52024800
4010+D	peSock 10I 0 value				000000		52024900
4011+D	peComm * value				000000		52025000
4012+					000000		52025100
4013+					000000		52025200
4014+	*+++++				000000		52025300
4015+	* http_error(): Return the last error that occurred.				000000		52025400
4016+	*				000000		52025500
4017+	* peErrorNo = (optional) error number that occurred.				000000		52025600
4018+	*				000000		52025700
4019+	* Returns the human-readable error message.				000000		52025800
4020+	*+++++				000000		52025900
4021+D	http_error PR 80A				000000		52026000
4022+D	peErrorNo 10I 0 options(*nopass:*omit)				000000		52026100
4023+					000000		52026200
4024+	/if defined(HAVE_SSLAPI)				000000		52026300
4025+					000000		52026400
4026+	*+++++				000000		52026500
4027+	* https_init(): Initialize https (HTTP over SSL/TLS) protocol				000000		52026600
4028+	*				000000		52026700
4029+	* peAppID = application ID that you registered program as				000000		52026800
4030+	* in the Digital Certificate Manager				000000		52026900
4031+	* NOTE: You can pass *BLANKS for this parameter if you				000000		52027000
4032+	* you do not wish to register your application				000000		52027100
4033+	* with the digital certificate manager.				000000		52027200
4034+	* peSSLv2 = (optional) Turn SSL version 2 *ON or *OFF				000000		52027300
4035+	* peSSLv3 = (optional) Turn SSL version 3 *ON or *OFF				000000		52027400
4036+	* peTLSv1 = (optional) Turn TLS version 1 *ON or *OFF				000000		52027500
4037+	*				000000		52027600
4038+	* You must pass all of the SSL/TLS flags or none. If you				000000		52027700
4039+	* do not pass all three flags, they are ignored.				000000		52027800
4040+	*				000000		52027900
4041+	*+++++				000000		52028000
4042+D	https_init PR 10I 0				000000		52028100
4043+D	peAppID 100A const				000000		52028200

Line Number	<----- Source Specifications ----->	<----- Comments ----->	Do Num	Page Line	Change Date	Src Id	Seq Number
4044+D	peSSLv2	1N const options(*nopass)			000000		52028300
4045+D	peSSLv3	1N const options(*nopass)			000000		52028400
4046+D	peTLSv1	1N const options(*nopass)			000000		52028500
4047+					000000		52028600
4048+					000000		52028700
4049+	*+++++				000000		52028800
4050+	* Register your application with the Digital Certificate Manager				000000		52028900
4051+	*				000000		52029000
4052+	* peAppID = application ID. IBM recommends that you do				000000		52029100
4053+	* something like: COMPANY_COMPONENT_NAME				000000		52029200
4054+	* (example: QIBM_DIRSRV_REPLICATION)				000000		52029300
4055+	*				000000		52029400
4056+	* peLimitCA = set to *On if you want to only want to allow the				000000		52029500
4057+	* certificate authorities registered in D.C.M., or set to				000000		52029600
4058+	* *Off if you'll manage that yourself.				000000		52029700
4059+	*				000000		52029800
4060+	* returns 0 for success, or -1 upon failure				000000		52029900
4061+	*+++++				000000		52030000
4062+D	https_dcm_reg PR	10I 0			000000		52030100
4063+D	peAppID	100A const			000000		52030200
4064+D	peLimitCA	1N const			000000		52030300
4065+					000000		52030400
4066+					000000		52030500
4067+	*+++++				000000		52030600
4068+	* https_cleanup(): Clean up & free storage used by the SSL				000000		52030700
4069+	* environment.				000000		52030800
4070+	*				000000		52030900
4071+	* returns 0 if successful, -1 upon failure				000000		52031000
4072+	*+++++				000000		52031100
4073+D	https_cleanup PR	10I 0			000000		52031200
4074+					000000		52031300
4075+	/endif				000000		52031400
4076+					000000		52031500
4077+					000000		52031600
4078+	*+++++				000000		52031700
4079+	* http_getauth(): Get HTTP Authentication Information				000000		52031800
4080+	*				000000		52031900
4081+	* Call this proc after you receive a HTTP_NDAUTH error				000000		52032000
4082+	* to determine the authentication credentials that are required				000000		52032100
4083+	*				000000		52032200
4084+	* The following parms are returned to your program:				000000		52032300
4085+	*				000000		52032400
4086+	* peBasic = *ON if BASIC auth is allowed				000000		52032500
4087+	* peDigest = *ON if MD5 DIGEST auth is allowed				000000		52032600
4088+	* peRealm = Auth realm. Present this to the user to identify				000000		52032700
4089+	* which password you're looking for. For example				000000		52032800
4090+	* if peRealm is "secureserver.com" you might say				000000		52032900
4091+	* "enter password for secureserver.com" to user.				000000		52033000
4092+	*				000000		52033100
4093+	* After getting the userid & password from the user (or database)				000000		52033200
4094+	* you'll need to call http_setauth()				000000		52033300
4095+	*				000000		52033400
4096+	* Returns -1 upon error, or 0 if successful				000000		52033500
4097+	*+++++				000000		52033600
4098+D	http_getauth PR	10I 0			000000		52033700
4099+D	peBasic	1N			000000		52033800

Line Number	<----- Source Specifications ----->	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
4100+D	peDigest 1N				000000		52033900
4101+D	peRealm 124A				000000		52034000
4102+					000000		52034100
4103+					000000		52034200
4104+	*+++++				000000		52034300
4105+	* http_setauth(): Set HTTP Authentication Information				000000		52034400
4106+	*				000000		52034500
4107+	* peAuthType = Authentication Type (HTTP_AUTH_BASIC or				000000		52034600
4108+	* HTTP_AUTH_MD5_DIGEST)				000000		52034700
4109+	* peUsername = UserName to use				000000		52034800
4110+	* pePasswd = Password to use				000000		52034900
4111+	*				000000		52035000
4112+	* Returns -1 upon error, or 0 if successful				000000		52035100
4113+	*+++++				000000		52035200
4114+D	http_setauth PR 10I 0				000000		52035300
4115+D	peAuthType 1A const				000000		52035400
4116+D	peUsername 80A const				000000		52035500
4117+D	pePasswd 1024A const				000000		52035600
4118+					000000		52035700
4119+					000000		52035800
4120+	*+++++				000000		52035900
4121+	* http_setproxy(): Set HTTP Proxy Address				000000		52036000
4122+	*				000000		52036100
4123+	* peHost = Proxy host name				000000		52036200
4124+	* psPort = Proxy port				000000		52036300
4125+	*				000000		52036400
4126+	* Returns -1 upon error, or 0 if successful				000000		52036500
4127+	*+++++				000000		52036600
4128+D	http_setproxy PR 10I 0				000000		52036700
4129+D	peHost 256A const				000000		52036800
4130+D	pePort 10I 0 const				000000		52036900
4131+					000000		52037000
4132+					000000		52037100
4133+	*+++++				000000		52037200
4134+	* http_proxy_setauth(): Set HTTP Proxy Authentication Information				000000		52037300
4135+	*				000000		52037400
4136+	* peAuthType = Authentication Type (HTTP_AUTH_NONE or				000000		52037500
4137+	* HTTP_AUTH_BASIC)				000000		52037600
4138+	* peUsername = UserName to use				000000		52037700
4139+	* pePasswd = Password to use				000000		52037800
4140+	*				000000		52037900
4141+	* Returns -1 upon error, or 0 if successful				000000		52038000
4142+	*+++++				000000		52038100
4143+D	http_proxy_setauth...				000000		52038200
4144+D	PR 10I 0				000000		52038300
4145+D	peAuthType 1A const				000000		52038400
4146+D	peUsername 80A const				000000		52038500
4147+D	pePasswd 1024A const				000000		52038600
4148+					000000		52038700
4149+	*+++++				000000		52038800
4150+	* http_proxy_getauth(): Get HTTP Proxy Authentication Information				000000		52038900
4151+	*				000000		52039000
4152+	* Call this proc after you receive a HTTP_PXNDAUTH error				000000		52039100
4153+	* to determine the authentication credentials that are required				000000		52039200
4154+	*				000000		52039300
4155+	* The following parms are returned to your program:				000000		52039400

Line Number	<----- Source Specifications ----->	<----- Comments ----->	Do Num	Page Line	Change Date	Src Id	Seq Number
4156+	*				000000		52039500
4157+	*	peBasic = *ON if BASIC auth is allowed			000000		52039600
4158+	*	peRealm = Auth realm. Present this to the user to identify			000000		52039700
4159+	*	which password you're looking for. For example			000000		52039800
4160+	*	if peRealm is "secureproxy.com" you might say			000000		52039900
4161+	*	"enter password for secureproxy.com" to user.			000000		52040000
4162+	*				000000		52040100
4163+	*	After getting the userid & password from the user (or database)			000000		52040200
4164+	*	you'll need to call http_proxy_setauth()			000000		52040300
4165+	*				000000		52040400
4166+	*	Returns -1 upon error, or 0 if successful			000000		52040500
4167+	*+++++				000000		52040600
4168+D	http_proxy_getauth...				000000		52040700
4169+D	PR	10I 0			000000		52040800
4170+D	peBasic	1N			000000		52040900
4171+D	peRealm	124A			000000		52041000
4172+	*				000000		52041100
4173+	*				000000		52041200
4174+	*+++++				000000		52041300
4175+	*	http_xproc(): Register a procedure to be called back at			000000		52041400
4176+	*	a given exit point			000000		52041500
4177+	*				000000		52041600
4178+	*	peExitPoint = exit point. Should be one of the constants			000000		52041700
4179+	*	HTTP_POINT_XXX defined in the HTTPAPI_H member			000000		52041800
4180+	*	peProc = address of procedure to call for this			000000		52041900
4181+	*	exit point. (pass *NULL to disable this point)			000000		52042000
4182+	*	peUserData = Pointer to user data. This will be passed			000000		52042100
4183+	*	to your call-back procedure. You can set it to			000000		52042200
4184+	*	*NULL if you don't need/want it.			000000		52042300
4185+	*				000000		52042400
4186+	*	Returns -1 upon error, or 0 if successful			000000		52042500
4187+	*+++++				000000		52042600
4188+D	http_xproc	PR 10I 0			000000		52042700
4189+D	peExitPoint	10I 0 value			000000		52042800
4190+D	peProc	* procptr value			000000		52042900
4191+D	peUserData	* value options(*nopass)			000000		52043000
4192+	*				000000		52043100
4193+	*				000000		52043200
4194+	*+++++				000000		52043300
4195+	*	http_redir_loc(): Retrieve location provided by a redirect			000000		52043400
4196+	*	request.			000000		52043500
4197+	*				000000		52043600
4198+	*	returns redirect location, or '' if no redirect was given			000000		52043700
4199+	*+++++				000000		52043800
4200+D	http_redir_loc	PR 1024A varying			000000		52043900
4201+	*				000000		52044000
4202+	*+++++				000000		52044100
4203+	*	http_url_encoder_new(): Create a URL encoder.			000000		52044200
4204+	*				000000		52044300
4205+	*	returns an (opaque) pointer to the new encoder			000000		52044400
4206+	*	or *NULL upon error.			000000		52044500
4207+	*				000000		52044600
4208+	*	WARNING: To free the memory used by this routine, you MUST			000000		52044700
4209+	*	call http_url_encoder_free() after the data is sent.			000000		52044800
4210+	*+++++				000000		52044900
4211+D	HTTP_URL_ENCODER...				000000		52045000

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
4212+D	s				000000		52045100
4213+D	http_url_encoder_new...				000000		52045200
4214+D	PR like(HTTP_URL_ENCODER)				000000		52045300
4215+	/if defined(WEBFORMS) LINES EXCLUDED: 3				070823		52045400
4216+	/endif				070823		52045800
4217+					000000		52045900
4218+					000000		52046000
4219+	*+++++				000000		52046100
4220+	* http_url_encoder_addvar(): Add a variable to what's stored				000000		52046200
4221+	* a URL encoder.				000000		52046300
4222+	*				000000		52046400
4223+	* peEncoder = pointer to encoder created by the				000000		52046500
4224+	* http_url_encoder_new() routine				000000		52046600
4225+	* peVariable = variable name to add				000000		52046700
4226+	* peData = pointer to data to store in variable				000000		52046800
4227+	* peDataSize = size of data to store in variable				000000		52046900
4228+	*				000000		52047000
4229+	* Returns *ON if successful, *OFF otherwise.				000000		52047100
4230+	*+++++				000000		52047200
4231+D	http_url_encoder_addvar...				000000		52047300
4232+D	PR 1N				000000		52047400
4233+D	peEncoder like(HTTP_URL_ENCODER) value				000000		52047500
4234+D	peVariable 50A varying value				000000		52047600
4235+D	peData * value				000000		52047700
4236+D	peDataSize 10I 0 value				000000		52047800
4237+	/if defined(WEBFORMS) LINES EXCLUDED: 6				070823		52047900
4238+	/endif				070823		52048600
4239+					000000		52048700
4240+					000000		52048800
4241+	*+++++				000000		52048900
4242+	* http_url_encoder_getptr(): Get a pointer to the encoded				000000		52049000
4243+	* data stored in a URL encoder				000000		52049100
4244+	*				000000		52049200
4245+	* peEncoder = (input) pointer to encoder				000000		52049300
4246+	* peData = (output) pointer to encoded data				000000		52049400
4247+	* peSize = (output) size of encoded data				000000		52049500
4248+	*+++++				000000		52049600
4249+D	http_url_encoder_getptr...				000000		52049700
4250+D	PR				000000		52049800
4251+D	peEncoder like(HTTP_URL_ENCODER) value				000000		52049900
4252+D	peData *				000000		52050000
4253+D	peSize 10I 0				000000		52050100
4254+	/if defined(WEBFORMS) LINES EXCLUDED: 5				070823		52050200
4255+	/endif				070823		52050800
4256+					000000		52050900
4257+					000000		52051000
4258+	*+++++				000000		52051100
4259+	* http_url_encoder_getstr(): Get encoded data he encoded				000000		52051200
4260+	* data stored in a URL encoder as a string				000000		52051300
4261+	*				000000		52051400
4262+	* peEncoder = (input) pointer to encoder				000000		52051500
4263+	*				000000		52051600
4264+	* NOTE: This routine is much slower than http_url_encoder_getptr()				000000		52051700

Line Number	<----- Source Specifications ----->	<----- Comments ----->	Do Num	Page Line	Change Date	Src Id	Seq Number
4265+	* and is limited to a 32k return value. It's suitable for				000000		52051800
4266+	* use with data that's added to a URL, such as when				000000		52051900
4267+	* performing a GET request to a web server, but you should				000000		52052000
4268+	* use http_url_encoder_getptr() for POST requests.				000000		52052100
4269+	*+++++				000000		52052200
4270+D	http_url_encoder_getstr...				000000		52052300
4271+D	PR 32767A varying				000000		52052400
4272+D	peEncoder like(HTTP_URL_ENCODER) value				000000		52052500
4273+	/if defined(WEBFORMS) LINES EXCLUDED: 4				070823		52052600
4274+	/endif				070823		52053100
4275+					000000		52053200
4276+					000000		52053300
4277+	*+++++				000000		52053400
4278+	* http_url_encoder_free(): free resources allocated by both				000000		52053500
4279+	* http_url_encoder_new() and http_url_encoder_addvar()				000000		52053600
4280+	*				000000		52053700
4281+	* peEncoder = pointer to encoder to free				000000		52053800
4282+	*				000000		52053900
4283+	* Returns *ON if successful, *OFF otherwise.				000000		52054000
4284+	*				000000		52054100
4285+	* WARNING: After calling this, do not use the encoder or				000000		52054200
4286+	* data returned by http_url_encoder_getptr() again.				000000		52054300
4287+	*+++++				000000		52054400
4288+D	http_url_encoder_free...				000000		52054500
4289+D	PR 1N				000000		52054600
4290+D	peEncoder like(HTTP_URL_ENCODER) value				000000		52054700
4291+	/if defined(WEBFORMS) LINES EXCLUDED: 3				070823		52054800
4292+	/endif				070823		52055200
4293+					000000		52055300
4294+	*+++++				000000		52055400
4295+	* http_url_encoder_addvar_s(): Simplified (but limited)				000000		52055500
4296+	* interface to http_url_encoder_addvar().				000000		52055600
4297+	*				000000		52055700
4298+	* peEncoder = (input) HTTP_url_encoder object				000000		52055800
4299+	* peVariable = (input) variable name to set				000000		52055900
4300+	* peValue = (input) value to set variable to				000000		52056000
4301+	*				000000		52056100
4302+	* Returns *ON if successful, *OFF otherwise				000000		52056200
4303+	*+++++				000000		52056300
4304+D	http_url_encoder_addvar_s...				000000		52056400
4305+D	PR 1N				000000		52056500
4306+D	peEncoder like(HTTP_URL_ENCODER) value				000000		52056600
4307+D	peVariable 50A varying value				000000		52056700
4308+D	peValue 256A varying value				000000		52056800
4309+	/if defined(WEBFORMS) LINES EXCLUDED: 5				070823		52056900
4310+	/endif				070823		52057500
4311+					000000		52057600
4312+					000000		52057700
4313+	*+++++				000000		52057800
4314+	* http_long_ParseURL(): Parse URL into it's component parts				000000		52057900
4315+	*				000000		52058000
4316+	* Breaks a uniform resource locator (URL) into it's component				000000		52058100
4317+	* pieces for use with the http: or https: protocols. (would also				000000		52058200

Line Number	<----- Source Specifications -----><----- Comments ----->	Do Num	Page Line	Change Date	Src Id	Seq Number
4318+	* work for FTP with minor tweaks)			000000		52058300
4319+	*			000000		52058400
4320+	* peURL = URL that needs to be parsed.			000000		52058500
4321+	* peService = service name from URL (i.e. http or https)			000000		52058600
4322+	* peUserName = user name given, or *blanks			000000		52058700
4323+	* pePassword = password given, or *blanks			000000		52058800
4324+	* peHost = hostname given in URL. (could be domain name or IP)			000000		52058900
4325+	* pePort = port number to connect to, if specified, otherwise 0.			000000		52059000
4326+	* pePath = remaining path/request for server.			000000		52059100
4327+	*			000000		52059200
4328+	* returns -1 upon failure, or 0 upon success			000000		52059300
4329+	*+++++			000000		52059400
4330+d	http_long_ParseURL...			000000		52059500
4331+d	PR 10I 0			000000		52059600
4332+d	peURL 32767A varying const options(*varsize)			000000		52059700
4333+d	peService 32A			000000		52059800
4334+d	peUserName 32A			000000		52059900
4335+d	pePassword 32A			000000		52060000
4336+d	peHost 256A			000000		52060100
4337+d	pePort 10I 0			000000		52060200
4338+d	pePath 32767A varying			000000		52060300
4339+				000000		52060400
4340+	*+++++			000000		52060500
4341+	* http_select_commdriver(): Select & initialize communications			000000		52060600
4342+	* driver.			000000		52060700
4343+	*			000000		52060800
4344+	* peCommType = (input) communications type (http/https)			000000		52060900
4345+	*			000000		52061000
4346+	* Returns pointer to comm driver, or *NULL upon failure			000000		52061100
4347+	*+++++			000000		52061200
4348+d	http_select_commdriver...			000000		52061300
4349+d	PR *			000000		52061400
4350+d	peCommType 32A const			000000		52061500
4351+				000000		52061600
4352+				000000		52061700
4353+	*+++++			000000		52061800
4354+	* http_url_post_raw2(): Post data to CGI script and get document			000000		52061900
4355+	*			000000		52062000
4356+	* peURL = url to post to (http://www.blah.com/cgi-bin/etc)			000000		52062100
4357+	* pePostFD = descriptor number to pass to pePostProc			000000		52062200
4358+	* pePostProc = procedure to call to get POST data.			000000		52062300
4359+	* peDataLen = total length of data that will be sent.			000000		52062400
4360+	* peSaveFD = FD to pass back to peSaveProc			000000		52062500
4361+	* peSaveProc = procedure to call each time data is received.			000000		52062600
4362+	* peTimeout = (optional) give up if no data is received for			000000		52062700
4363+	* this many seconds.			000000		52062800
4364+	* peUserAgent = (optional) User-Agent string passed to the			000000		52062900
4365+	* server. Pass the named constant HTTP_USERAGENT			000000		52063000
4366+	* if you want to get the default value.			000000		52063100
4367+	* peContentType = (optional) content type to supply (mainly			000000		52063200
4368+	* useful when talking to CGI scripts)			000000		52063300
4369+	* peSOAPAction = (optional) string used to specify the action			000000		52063400
4370+	* taken by some SOAP applications.			000000		52063500
4371+	*			000000		52063600
4372+	* Returns -1 upon failure, 0 upon timeout, or an HTTP response code			000000		52063700
4373+	*+++++			000000		52063800

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
4374+D	http_url_post_raw2...				000000		52063900
4375+D	PR 10I 0				000000		52064000
4376+D	peURL 32767A	varying const options(*varsize)			000000		52064100
4377+D	pePostFD 10I 0	value			000000		52064200
4378+D	pePostProc *	procptr value			000000		52064300
4379+D	peDataLen 10I 0	value			000000		52064400
4380+D	peSaveFD 10I 0	value			000000		52064500
4381+D	peSaveProc *	value procptr			000000		52064600
4382+D	peTimeout 10I 0	value options(*nopass)			000000		52064700
4383+D	peUserAgent 64A	const options(*nopass:*omit)			000000		52064800
4384+D	peContentType 64A	const options(*nopass:*omit)			000000		52064900
4385+D	peSOAPAction 64A	const options(*nopass:*omit)			000000		52065000
4386+					000000		52065100
4387+					000000		52065200
4388+	*+++++				000000		52065300
4389+	* http_url_post_stmf(): Post data to CGI script from stream file				000000		52065400
4390+	*				000000		52065500
4391+	* peURL = url to post to (http://www.blah.com/cgi-bin/etc)				000000		52065600
4392+	* pePostFile = Filename (in IFS) of file to send to http server				000000		52065700
4393+	* peRecvFile = Filename (in IFS) of stream file containing reply				000000		52065800
4394+	* peTimeout = (optional) give up if no data is received for				000000		52065900
4395+	* this many seconds.				000000		52066000
4396+	* peUserAgent = (optional) User-Agent string passed to the				000000		52066100
4397+	* server. Pass the named constant HTTP_USERAGENT				000000		52066200
4398+	* if you want to get the default value.				000000		52066300
4399+	* peContentType = (optional) content type to supply (mainly				000000		52066400
4400+	* useful when talking to CGI scripts)				000000		52066500
4401+	* peSOAPAction = (optional) string used to specify the action				000000		52066600
4402+	* taken by some SOAP applications.				000000		52066700
4403+	*				000000		52066800
4404+	* Returns -1 upon failure, 0 upon timeout,				000000		52066900
4405+	* 1 for success, or an HTTP response code				000000		52067000
4406+	*+++++				000000		52067100
4407+D	http_url_post_stmf...				000000		52067200
4408+D	PR 10I 0				000000		52067300
4409+D	peURL 32767A	varying const options(*varsize)			000000		52067400
4410+D	pePostFile 32767A	varying const options(*varsize)			000000		52067500
4411+D	peRecvFile 32767A	varying const options(*varsize)			000000		52067600
4412+D	peTimeout 10I 0	value options(*nopass)			000000		52067700
4413+D	peUserAgent 64A	const options(*nopass:*omit)			000000		52067800
4414+D	peContentType 64A	const options(*nopass:*omit)			000000		52067900
4415+D	peSOAPAction 64A	const options(*nopass:*omit)			000000		52068000
4416+D	http_post_stmf PR 10I 0	extproc('HTTP_URL_POST_STMF')			000000		52068100
4417+D	peURL 32767A	varying const options(*varsize)			000000		52068200
4418+D	pePostFile 32767A	varying const options(*varsize)			000000		52068300
4419+D	peRecvFile 32767A	varying const options(*varsize)			000000		52068400
4420+D	peTimeout 10I 0	value options(*nopass)			000000		52068500
4421+D	peUserAgent 64A	const options(*nopass:*omit)			000000		52068600
4422+D	peContentType 64A	const options(*nopass:*omit)			000000		52068700
4423+D	peSOAPAction 64A	const options(*nopass:*omit)			000000		52068800
4424+					000000		52068900
4425+					000000		52069000
4426+	*+++++				000000		52069100
4427+	* http_get_xml();				000000		52069200
4428+	* http_url_get_xml(): Send a GET request to an HTTP server and				000000		52069300
4429+	* receive/parse an XML response.				000000		52069400

Line	Source Specifications	Comments	Do	Page	Change	Src	Seq
Number	1.....2.....3.....4.....5.....6.....7.....8.....9.....10		Num	Line	Date	Id	Number
4430+	*				000000		52069500
4431+	* peURL = (input) URL to perform GET request to				000000		52069600
4432+	* peStartProc = (input) call-back procedure to call at the start				000000		52069700
4433+	* of each XML element received.				000000		52069800
4434+	* peEndProc = (input) call-back procedure to call at the end				000000		52069900
4435+	* of each XML element received.				000000		52070000
4436+	* peUsrDta = (input) user-defined data that will be passed to the				000000		52070100
4437+	* call-back routine				000000		52070200
4438+	*				000000		52070300
4439+	* (other parms are identical to those in HTTP_url_get())				000000		52070400
4440+	*				000000		52070500
4441+	* peStartProc should point to a procedure with a procedure				000000		52070600
4442+	* interface that's compatable with the following:				000000		52070700
4443+	*				000000		52070800
4444+	* D StartProc PR				000000		52070900
4445+	* D userdata *	value			000000		52071000
4446+	* D depth 10I 0	value			000000		52071100
4447+	* D name 1024A	varying const			000000		52071200
4448+	* D path 24576A	varying const			000000		52071300
4449+	* D attrs *	dim(32767)			000000		52071400
4450+	* D	const options(*varsize)			000000		52071500
4451+	*				000000		52071600
4452+	* peEndProc should point to a procedure with a procedure				000000		52071700
4453+	* interface that's compatable with the following:				000000		52071800
4454+	*				000000		52071900
4455+	* D EndProc PR				000000		52072000
4456+	* D userdata *	value			000000		52072100
4457+	* D depth 10I 0	value			000000		52072200
4458+	* D name 1024A	varying const			000000		52072300
4459+	* D path 24576A	varying const			000000		52072400
4460+	* D value 32767A	varying const			000000		52072500
4461+	* D attrs *	dim(32767)			000000		52072600
4462+	* D	const options(*varsize)			000000		52072700
4463+	*				000000		52072800
4464+	* Returns 1 if successful, -1 upon error, 0 if timeout				000000		52072900
4465+	*+++++						
4466+	D http_get_xml...				000000		52073100
4467+	D PR 10I 0	EXTPROC('HTTP_URL_GET_XML')			000000		52073200
4468+	D peURL 32767A	varying const options(*varsize)			000000		52073300
4469+	D peStartProc *	value procptr			000000		52073400
4470+	D peEndProc *	value procptr			000000		52073500
4471+	D peUsrDta *	value			000000		52073600
4472+	D peTimeout 10I 0	value options(*nopass)			000000		52073700
4473+	D peUserAgent 64A	const options(*nopass:*omit)			000000		52073800
4474+	D peModTime Z	const options(*nopass:*omit)			000000		52073900
4475+	D peContentType 64A	const options(*nopass:*omit)			000000		52074000
4476+	D peSOAPAction 64A	const options(*nopass:*omit)			000000		52074100
4477+	D http_url_get_xml...				000000		52074200
4478+	D PR 10I 0				000000		52074300
4479+	D peURL 32767A	varying const options(*varsize)			000000		52074400
4480+	D peStartProc *	value procptr			000000		52074500
4481+	D peEndProc *	value procptr			000000		52074600
4482+	D peUsrDta *	value			000000		52074700
4483+	D peTimeout 10I 0	value options(*nopass)			000000		52074800
4484+	D peUserAgent 64A	const options(*nopass:*omit)			000000		52074900
4485+	D peModTime Z	const options(*nopass:*omit)			000000		52075000

Line Number	<----- Source Specifications -----><----- Comments ----->	Do Num	Page Line	Change Date	Src Id	Seq Number
4486+D	peContentType 64A const options(*nopass:*omit)			000000		52075100
4487+D	peSOAPAction 64A const options(*nopass:*omit)			000000		52075200
4488+				000000		52075300
4489+	*+++++			000000		52075400
4490+	* http_get_xmltf(): Request URL from server. Receive response			000000		52075500
4491+	* to temporary file, then parse it.			000000		52075600
4492+				000000		52075700
4493+	* peURL = (input) URL to perform GET request to			000000		52075800
4494+	* peStartProc = (input) call-back procedure to call at the start			000000		52075900
4495+	* of each XML element received.			000000		52076000
4496+	* peEndProc = (input) call-back procedure to call at the end			000000		52076100
4497+	* of each XML element received.			000000		52076200
4498+	* peUsrDta = (input) user-defined data that will be passed to the			000000		52076300
4499+	* call-back routine			000000		52076400
4500+				000000		52076500
4501+	* (other parms are identical to those in HTTP_url_get())			000000		52076600
4502+				000000		52076700
4503+	* Returns 1 if successful, -1 upon error, 0 if timeout			000000		52076800
4504+	*+++++			000000		52076900
4505+D	http_get_xmltf...			000000		52077000
4506+D	PR 10I 0			000000		52077100
4507+D	peURL 32767A varying const options(*varsize)			000000		52077200
4508+D	peStartProc * value procptr			000000		52077300
4509+D	peEndProc * value procptr			000000		52077400
4510+D	peUsrDta * value			000000		52077500
4511+D	peTimeout 10I 0 value options(*nopass)			000000		52077600
4512+D	peUserAgent 64A const options(*nopass:*omit)			000000		52077700
4513+D	peModTime Z const options(*nopass:*omit)			000000		52077800
4514+D	peContentType 64A const options(*nopass:*omit)			000000		52077900
4515+D	peSOAPAction 64A const options(*nopass:*omit)			000000		52078000
4516+				000000		52078100
4517+				000000		52078200
4518+	*+++++			000000		52078300
4519+	* http_post_xml();			000000		52078400
4520+	* http_url_post_xml(): Send a POST request to an HTTP server and			000000		52078500
4521+	* receive/parse an XML response.			000000		52078600
4522+				000000		52078700
4523+	* peURL = (input) URL to perform GET request to			000000		52078800
4524+	* pePostData = (input) data to POST to the web server			000000		52078900
4525+	* pePostDataLen = (input) length of pePostData			000000		52079000
4526+	* peStartProc = (input) call-back procedure to call at the start			000000		52079100
4527+	* of each XML element received.			000000		52079200
4528+	* peEndProc = (input) call-back procedure to call at the end			000000		52079300
4529+	* of each XML element received.			000000		52079400
4530+	* peUsrDta = (input) user-defined data that will be passed			000000		52079500
4531+	* to the call-back routine			000000		52079600
4532+				000000		52079700
4533+	* (other parms are identical to those in HTTP_url_post())			000000		52079800
4534+				000000		52079900
4535+	* peStartProc should point to a procedure with a procedure			000000		52080000
4536+	* interface that's compatable with the following:			000000		52080100
4537+				000000		52080200
4538+	* D StartProc PR			000000		52080300
4539+	* D userdata * value			000000		52080400
4540+	* D depth 10I 0 value			000000		52080500
4541+	* D name 1024A varying const			000000		52080600

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
4542+	* D path 24576A varying const				000000		52080700
4543+	* D attrs * dim(32767)				000000		52080800
4544+	* D const options(*varsize)				000000		52080900
4545+	*				000000		52081000
4546+	* peEndProc should point to a procedure with a procedure				000000		52081100
4547+	* interface that's compatable with the following:				000000		52081200
4548+	*				000000		52081300
4549+	* D EndProc PR				000000		52081400
4550+	* D userdata * value				000000		52081500
4551+	* D depth 10I 0 value				000000		52081600
4552+	* D name 1024A varying const				000000		52081700
4553+	* D path 24576A varying const				000000		52081800
4554+	* D value 32767A varying const				000000		52081900
4555+	* D attrs * dim(32767)				000000		52082000
4556+	* D const options(*varsize)				000000		52082100
4557+	*				000000		52082200
4558+	* Returns 1 if successful, -1 upon error, 0 if timeout				000000		52082300
4559+	*+++++				000000		52082400
4560+D	http_post_xml...				000000		52082500
4561+D	PR 10I 0 EXTPROC('HTTP_URL_POST_XML')				000000		52082600
4562+D	peURL 32767A varying const options(*varsize)		1		070724		52082700
4563+D	pePostData * value		2		070724		52082800
4564+D	pePostDataLen 10I 0 value		3		070724		52082900
4565+D	peStartProc * value procptr		4		070724		52083000
4566+D	peEndProc * value procptr		5		070724		52083100
4567+D	peUsrDta * value		6		070724		52083200
4568+D	peTimeout 10I 0 value options(*nopass)		7		070724		52083300
4569+D	peUserAgent 64A const options(*nopass:*omit)		8		070724		52083400
4570+D	peContentType 64A const options(*nopass:*omit)		9		070724		52083500
4571+D	peSOAPAction 64A const options(*nopass:*omit)		10		070724		52083600
4572+D	http_url_post_xml...				000000		52083700
4573+D	PR 10I 0				000000		52083800
4574+D	peURL 32767A varying const options(*varsize)				000000		52083900
4575+D	pePostData * value				000000		52084000
4576+D	pePostDataLen 10I 0 value				000000		52084100
4577+D	peStartProc * value procptr				000000		52084200
4578+D	peEndProc * value procptr				000000		52084300
4579+D	peUsrDta * value				000000		52084400
4580+D	peTimeout 10I 0 value options(*nopass)				000000		52084500
4581+D	peUserAgent 64A const options(*nopass:*omit)				000000		52084600
4582+D	peContentType 64A const options(*nopass:*omit)				000000		52084700
4583+D	peSOAPAction 64A const options(*nopass:*omit)				000000		52084800
4584+					000000		52084900
4585+					000000		52085000
4586+	*+++++				000000		52085100
4587+	* http_post_xmltf(): Post data from memory. Receive				000000		52085200
4588+	* response to temporary file, then parse it.				000000		52085300
4589+	*				000000		52085400
4590+	* peURL = (input) URL to perform GET request to				000000		52085500
4591+	* pePostData = (input) data to POST to the web server				000000		52085600
4592+	* pePostDataLen = (input) length of pePostData				000000		52085700
4593+	* peStartProc = (input) call-back procedure to call at the start				000000		52085800
4594+	* of each XML element received.				000000		52085900
4595+	* peEndProc = (input) call-back procedure to call at the end				000000		52086000
4596+	* of each XML element received.				000000		52086100
4597+	* peUsrDta = (input) user-defined data that will be passed				000000		52086200

Line Number	<----- Source Specifications ----->	<----- Comments ----->	Do	Page	Change	Src	Seq
			Num	Line	Date	Id	Number
4598+	* to the call-back routine				000000		52086300
4599+	*				000000		52086400
4600+	* (other parms are identical to those in HTTP_url_post())				000000		52086500
4601+	*				000000		52086600
4602+	* Returns 1 if successful, -1 upon error, 0 if timeout				000000		52086700
4603+	*+++++				000000		52086800
4604+D	http_post_xmltf...				000000		52086900
4605+D	PR 10I 0				000000		52087000
4606+D	peURL 32767A varying const options(*varsize)				000000		52087100
4607+D	pePostData * value				000000		52087200
4608+D	pePostDataLen 10I 0 value				000000		52087300
4609+D	peStartProc * value procptr				000000		52087400
4610+D	peEndProc * value procptr				000000		52087500
4611+D	peUsrDta * value				000000		52087600
4612+D	peTimeout 10I 0 value options(*nopass)				000000		52087700
4613+D	peUserAgent 64A const options(*nopass:*omit)				000000		52087800
4614+D	peContentType 64A const options(*nopass:*omit)				000000		52087900
4615+D	peSOAPAction 64A const options(*nopass:*omit)				000000		52088000
4616+					000000		52088100
4617+					000000		52088200
4618+	*+++++				000000		52088300
4619+	* http_post_stmf_xml();				000000		52088400
4620+	* http_url_post_stmf_xml(): Post data to CGI script from stream file				000000		52088500
4621+	* and receive/parse an XML response				000000		52088600
4622+	*				000000		52088700
4623+	* peURL = (input) URL to post to				000000		52088800
4624+	* pePostFile = (input) File of stream file (in IFS) to post				000000		52088900
4625+	* peStartProc = (input) call-back procedure to call at the start				000000		52089000
4626+	* of each XML element received.				000000		52089100
4627+	* peEndProc = (input) call-back procedure to call at the end				000000		52089200
4628+	* of each XML element received.				000000		52089300
4629+	* peUsrDta = (input) user-defined data that will be passed				000000		52089400
4630+	* to the call-back routine				000000		52089500
4631+	* peTimeout = (optional) give up if no data is received for				000000		52089600
4632+	* this many seconds.				000000		52089700
4633+	* peUserAgent = (optional) User-Agent string passed to the				000000		52089800
4634+	* server. Pass the named constant HTTP_USERAGENT				000000		52089900
4635+	* if you want to get the default value.				000000		52090000
4636+	* peContentType = (optional) content type to supply (mainly				000000		52090100
4637+	* useful when talking to CGI scripts)				000000		52090200
4638+	* peSOAPAction = (optional) string used to specify the action				000000		52090300
4639+	* taken by some SOAP applications.				000000		52090400
4640+	*				000000		52090500
4641+	* Returns -1 upon failure, 0 upon timeout,				000000		52090600
4642+	* 1 for success, or an HTTP response code				000000		52090700
4643+	*+++++				000000		52090800
4644+D	http_post_stmf_xml...				000000		52090900
4645+D	PR 10I 0 extproc('HTTP_URL_POST_STMF_XML')				000000		52091000
4646+D	peURL 32767A varying const options(*varsize)				000000		52091100
4647+D	pePostFile 32767A varying const options(*varsize)				000000		52091200
4648+D	peStartProc * value procptr				000000		52091300
4649+D	peEndProc * value procptr				000000		52091400
4650+D	peUsrDta * value				000000		52091500
4651+D	peTimeout 10I 0 value options(*nopass)				000000		52091600
4652+D	peUserAgent 64A const options(*nopass:*omit)				000000		52091700
4653+D	peContentType 64A const options(*nopass:*omit)				000000		52091800

Line Number	<----- Source Specifications -----><----- Comments ----->	Do Num	Page Line	Change Date	Src Id	Seq Number
4654+D	peSOAPAction 64A const options(*nopass:*omit)			000000		52091900
4655+D	http_url_post_stmf_xml...			000000		52092000
4656+D	PR 10I 0			000000		52092100
4657+D	peURL 32767A varying const options(*varsize)			000000		52092200
4658+D	pePostFile 32767A varying const options(*varsize)			000000		52092300
4659+D	peStartProc * value procptr			000000		52092400
4660+D	peEndProc * value procptr			000000		52092500
4661+D	peUsrDta * value			000000		52092600
4662+D	peTimeout 10I 0 value options(*nopass)			000000		52092700
4663+D	peUserAgent 64A const options(*nopass:*omit)			000000		52092800
4664+D	peContentType 64A const options(*nopass:*omit)			000000		52092900
4665+D	peSOAPAction 64A const options(*nopass:*omit)			000000		52093000
4666+				000000		52093100
4667+				000000		52093200
4668+	*+++++			000000		52093300
4669+	* http_post_stmf_xmltf(): Post data from stream file. Receive			000000		52093400
4670+	* response to temporary file, then parse it.			000000		52093500
4671+	*			000000		52093600
4672+	* peURL = (input) URL to post to			000000		52093700
4673+	* pePostFile = (input) File of stream file (in IFS) to post			000000		52093800
4674+	* peStartProc = (input) call-back procedure to call at the start			000000		52093900
4675+	* of each XML element received.			000000		52094000
4676+	* peEndProc = (input) call-back procedure to call at the end			000000		52094100
4677+	* of each XML element received.			000000		52094200
4678+	* peUsrDta = (input) user-defined data that will be passed			000000		52094300
4679+	* to the call-back routine			000000		52094400
4680+	* peTimeout = (optional) give up if no data is received for			000000		52094500
4681+	* this many seconds.			000000		52094600
4682+	* peContentType = (optional) content type to supply (mainly			000000		52094700
4683+	* useful when talking to CGI scripts)			000000		52094800
4684+	*			000000		52094900
4685+	* Returns -1 upon failure, 0 upon timeout,			000000		52095000
4686+	* 1 for success, or an HTTP response code			000000		52095100
4687+	*+++++			000000		52095200
4688+D	http_post_stmf_xmltf...			000000		52095300
4689+D	PR 10I 0			000000		52095400
4690+D	peURL 32767A varying const options(*varsize)			000000		52095500
4691+D	pePostFile 32767A varying const options(*varsize)			000000		52095600
4692+D	peStartProc * value procptr			000000		52095700
4693+D	peEndProc * value procptr			000000		52095800
4694+D	peUsrDta * value			000000		52095900
4695+D	peTimeout 10I 0 value options(*nopass)			000000		52096000
4696+D	peUserAgent 64A const options(*nopass:*omit)			000000		52096100
4697+D	peContentType 64A const options(*nopass:*omit)			000000		52096200
4698+D	peSOAPAction 64A const options(*nopass:*omit)			000000		52096300
4699+				000000		52096400
4700+				000000		52096500
4701+	*+++++			000000		52096600
4702+	* http_persist_open(): Open a persistent HTTP session			000000		52096700
4703+	*			000000		52096800
4704+	* peURL = url to connect to			000000		52096900
4705+	* peTimeout = (optional) give up if no data is received for			000000		52097000
4706+	* this many seconds.			000000		52097100
4707+	*			000000		52097200
4708+	* Returns *NULL upon failure, or			000000		52097300
4709+	* pointer to HTTP communication session			000000		52097400

Line	----- Source Specifications -----><----- Comments ----->	Do	Page	Change	Src	Seq
Number1....+....2....+....3....+....4....+....5....+....6....+....7....+....8....+....9....+....10	Num	Line	Date	Id	Number
4710+	*+++++			000000		52097500
4711+D	http_persist_open...			000000		52097600
4712+D	PR			000000		52097700
4713+D	peURL 32767A varying const options(*varsize)			000000		52097800
4714+D	peTimeout 10I 0 value options(*nopass)			000000		52097900
4715+				000000		52098000
4716+				000000		52098100
4717+	*+++++			000000		52098200
4718+	* http_persist_get(): Get a file using a persistent HTTP session			000000		52098300
4719+	*			000000		52098400
4720+	peComm = (input) pointer to persistent HTTP comm session			000000		52098500
4721+	peURL = (input) URL to get from persistent HTTP			000000		52098600
4722+	peFD = (input) FD to pass back to peProc			000000		52098700
4723+	peProc = (input) procedure to call each time data is			000000		52098800
4724+	received.			000000		52098900
4725+	peTimeout = (input/optional) time-out when no data is received			000000		52099000
4726+	for this many seconds.			000000		52099100
4727+	peUserAgent = (optional) User-Agent string passed to the			000000		52099200
4728+	server. Pass the named constant called			000000		52099300
4729+	HTTP_USERAGENT if you want to get the			000000		52099400
4730+	default value.			000000		52099500
4731+	peModTime = (input/optional) only get file if it was changed			000000		52099600
4732+	since this timestamp.			000000		52099700
4733+	peContentType = (input/optional) content type to supply (mainly			000000		52099800
4734+	useful when talking to CGI scripts)			000000		52099900
4735+	peSOAPAction = (input/optional) string to send in the SOAPAction:			000000		52100000
4736+	HTTP header when making a SOAP request.			000000		52100100
4737+				000000		52100200
4738+	Returns 1 if successful,			000000		52100300
4739+	0 if timed out			000000		52100400
4740+	-1 if an internal error occurs			000000		52100500
4741+	or an HTTP response code if an error comes from the server			000000		52100600
4742+	*+++++			000000		52100700
4743+D	http_persist_get...			000000		52100800
4744+D	PR 10I 0			000000		52100900
4745+D	peComm * value			000000		52101000
4746+D	peURL 32767A varying const options(*varsize)			000000		52101100
4747+D	peFD 10I 0 value			000000		52101200
4748+D	peProc * value procptr			000000		52101300
4749+D	peTimeout 10I 0 value options(*nopass)			000000		52101400
4750+D	peUserAgent 64A const options(*nopass:*omit)			000000		52101500
4751+D	peModTime Z const options(*nopass:*omit)			000000		52101600
4752+D	peContentType 64A const options(*nopass:*omit)			000000		52101700
4753+D	peSOAPAction 64A const options(*nopass:*omit)			000000		52101800
4754+				000000		52101900
4755+				000000		52102000
4756+	*+++++			000000		52102100
4757+	* http_persist_post(): Post data to CGI script and get document			000000		52102200
4758+	using a persistent connection			000000		52102300
4759+	*			000000		52102400
4760+	peComm = (input) pointer to persistent HTTP comm session			000000		52102500
4761+	peURL = (input) URL to post to with persistent HTTP			000000		52102600
4762+	-----			000000		52102700
4763+	pePostFD = (input) Opaque integer to pass to pePostProc			000000		52102800
4764+	pePostProc = (input) Pointer to call-back procedure for			000000		52102900
4765+	posting data to server. If you pass			000000		52103000

Line Number	<----- Source Specifications -----><----- Comments ----->	Do Num	Page Line	Change Date	Src Id	Seq Number
4766+	* *NULL for this, you should use pePostData			000000		52103100
4767+	* instead.			000000		52103200
4768+	* -- or --			000000		52103300
4769+	* pePostData = (input) Pointer to data to post. If you pass			000000		52103400
4770+	* *NULL for this, you should use pePostProc			000000		52103500
4771+	* instead.			000000		52103600
4772+	* -----			000000		52103700
4773+	* pePostDataLen = (input) Total length, in bytes, of post data.			000000		52103800
4774+	* peSaveFD = (input) Opaque integer passed to peSaveProc			000000		52103900
4775+	* peSaveProc = (input) Pointer to call-back procedure that is			000000		52104000
4776+	* called when data is received from HTTP			000000		52104100
4777+	* server.			000000		52104200
4778+	* peTimeout = (input/optional) time-out when no data is received			000000		52104300
4779+	* for this many seconds.			000000		52104400
4780+	* peUserAgent = (optional) User-Agent string passed to the			000000		52104500
4781+	* server. Pass the named constant called			000000		52104600
4782+	* HTTP_USERAGENT if you want to get the			000000		52104700
4783+	* default value.			000000		52104800
4784+	* peContentType = (input/optional) content type to supply (mainly			000000		52104900
4785+	* useful when talking to CGI scripts)			000000		52105000
4786+	* peSOAPAction = (input/optional) string to send in the SOAPAction:			000000		52105100
4787+	* HTTP header when making a SOAP request.			000000		52105200
4788+	*			000000		52105300
4789+	* Returns 1 if successful,			000000		52105400
4790+	* 0 if timed out			000000		52105500
4791+	* -1 if an internal error occurs			000000		52105600
4792+	* or an HTTP response code if an error comes from the server			000000		52105700
4793+	*+++++			000000		52105800
4794+D	http_persist_post...			000000		52105900
4795+D	PR 10I 0			000000		52106000
4796+D	peComm * value			000000		52106100
4797+D	peURL 32767A varying const options(*varsize)			000000		52106200
4798+D	pePostFD 10I 0 value			000000		52106300
4799+D	pePostProc * value procptr			000000		52106400
4800+D	pePostData * value			000000		52106500
4801+D	pePostDataLen 10I 0 value			000000		52106600
4802+D	peSaveFD 10I 0 value			000000		52106700
4803+D	peSaveProc * value procptr			000000		52106800
4804+D	peTimeout 10I 0 value options(*nopass)			000000		52106900
4805+D	peUserAgent 64A const options(*nopass:*omit)			000000		52107000
4806+D	peContentType 64A const options(*nopass:*omit)			000000		52107100
4807+D	peSOAPAction 64A const options(*nopass:*omit)			000000		52107200
4808+				000000		52107300
4809+				000000		52107400
4810+	*+++++			000000		52107500
4811+	* http_persist_close(): End a persistent HTTP session			000000		52107600
4812+	*			000000		52107700
4813+	* peComm = (input) pointer to persistent HTTP comm session			000000		52107800
4814+	*			000000		52107900
4815+	* returns 0 if successful, -1 otherwise			000000		52108000
4816+	*+++++			000000		52108100
4817+D	http_persist_close...			000000		52108200
4818+D	PR 10I 0			000000		52108300
4819+D	peComm * value			000000		52108400
4820+				000000		52108500
4821+				000000		52108600

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
4822+	*****				000000		52108700
4823+	* http_mfd_encoder_open(): Create a multipart/form-data encoder				000000		52108800
4824+	*****				000000		52108900
4825+	* A multipart/form-data encoder will encode the variables				000000		52109000
4826+	* and or stream files that you pass to it and store the results				000000		52109100
4827+	* in a stream file. You can later POST those results with the				000000		52109200
4828+	* http_url_post_stmf() API.				000000		52109300
4829+	*****				000000		52109400
4830+	* peStmFile = (input) pathname to stream file to store				000000		52109500
4831+	* encoded results.				000000		52109600
4832+	*****				000000		52109700
4833+	* returns an (opaque) pointer to the new encoder				000000		52109800
4834+	* or *NULL upon error.				000000		52109900
4835+	*****				000000		52110000
4836+	* WARNING: To free the memory used by this routine and close				000000		52110100
4837+	* the stream file, you MUST call http_mfd_encoder_close()				000000		52110200
4838+	* after the data is sent.				000000		52110300
4839+	*****				000000		52110400
4840+D	http_mfd_encoder_open...				000000		52110500
4841+D	PR	*			000000		52110600
4842+D	peStmFile	*	value options(*string)		000000		52110700
4843+D	peContType	64A			000000		52110800
4844+	*****				000000		52110900
4845+	*****				000000		52111000
4846+	*****				000000		52111100
4847+	* http_mfd_encoder_addvar(): Add a variable to what's stored				000000		52111200
4848+	* a multipart/form-data encoder.				000000		52111300
4849+	*****				000000		52111400
4850+	* peEncoder = pointer to encoder created by the				000000		52111500
4851+	* http_mfd_encoder_open() routine				000000		52111600
4852+	* peVariable = variable name to add				000000		52111700
4853+	* peData = pointer to data to store in variable				000000		52111800
4854+	* peDataSize = size of data to store in variable				000000		52111900
4855+	*****				000000		52112000
4856+	* Returns *ON if successful, *OFF otherwise.				000000		52112100
4857+	*****				000000		52112200
4858+D	http_mfd_encoder_addvar...				000000		52112300
4859+D	PR	1N			000000		52112400
4860+D	peEncoder	*	value		000000		52112500
4861+D	peVariable	50A	varying value		000000		52112600
4862+D	peData	*	value		000000		52112700
4863+D	peDataSize	10I 0	value		000000		52112800
4864+	*****				000000		52112900
4865+	*****				000000		52113000
4866+	*****				000000		52113100
4867+	* http_mfd_encoder_addvar_s(): Simplified (but limited)				000000		52113200
4868+	* interface to http_mfd_encoder_addvar().				000000		52113300
4869+	*****				000000		52113400
4870+	* peEncoder = (input) HTTP_mfd_encoder object				000000		52113500
4871+	* peVariable = (input) variable name to set				000000		52113600
4872+	* peValue = (input) value to set variable to				000000		52113700
4873+	*****				000000		52113800
4874+	* Returns *ON if successful, *OFF otherwise				000000		52113900
4875+	*****				000000		52114000
4876+D	http_mfd_encoder_addvar_s...				000000		52114100
4877+D	PR	1N			000000		52114200

Line Number	<----- Source Specifications ----->	<----- Comments ----->	Do Num	Page Line	Change Date	Src Id	Seq Number
4878+D	peEncoder * value				000000		52114300
4879+D	peVariable 50A varying value				000000		52114400
4880+D	peValue 256A varying value				000000		52114500
4881+					000000		52114600
4882+					000000		52114700
4883+	*+++++				000000		52114800
4884+	* http_mfd_encoder_addstmf(): Add a stream file to what's stored				000000		52114900
4885+	* in a multipart/form-data encoder.				000000		52115000
4886+	*				000000		52115100
4887+	* peEncoder = pointer to encoder created by the				000000		52115200
4888+	* http_mfd_encoder_open() routine				000000		52115300
4889+	* peVariable = variable name to add				000000		52115400
4890+	* pePathName = Path name of stream file to add				000000		52115500
4891+	* peContType = Content-type of stream file to add				000000		52115600
4892+	*				000000		52115700
4893+	* Returns *ON if successful, *OFF otherwise.				000000		52115800
4894+	*+++++				000000		52115900
4895+D	http_mfd_encoder_addstmf...				000000		52116000
4896+D	PR 1N				000000		52116100
4897+D	peEncoder * value				000000		52116200
4898+D	peVariable 50A varying value				000000		52116300
4899+D	pePathName * value options(*string)				000000		52116400
4900+D	peContType 64A varying const				000000		52116500
4901+					000000		52116600
4902+					000000		52116700
4903+	*+++++				000000		52116800
4904+	* http_mfd_encoder_close(): close an open multipart/form-data				000000		52116900
4905+	* encoder.				000000		52117000
4906+	*				000000		52117100
4907+	* peEncoder = (input) encoder to close				000000		52117200
4908+	*+++++				000000		52117300
4909+D	http_mfd_encoder_close...				000000		52117400
4910+D	PR				000000		52117500
4911+D	peEncoder * value				000000		52117600
4912+					000000		52117700
4913+					000000		52117800
4914+	*+++++				000000		52117900
4915+	* http_debug(): Turn debugging info *ON or *OFF				000000		52118000
4916+	*				000000		52118100
4917+	* peStatus = (input) status (either *ON or *OFF)				000000		52118200
4918+	*				000000		52118300
4919+	* peFilename = (input/optional) filename that debug info will be				000000		52118400
4920+	* written to. If not defined, the value from				000000		52118500
4921+	* CONFIG_H is used.				000000		52118600
4922+	*+++++				000000		52118700
4923+D	http_debug PR				000000		52118800
4924+D	peStatus 1N const				000000		52118900
4925+D	peFilename 500A varying const options(*nopass)				000000		52119000
4926+					000000		52119100
4927+					000000		52119200
4928+	*+++++				000000		52119300
4929+	* HTTP_SetCCSIDs(): Set the CCSIDs used for ASCII/EBCDIC				000000		52119400
4930+	* translation				000000		52119500
4931+	*				000000		52119600
4932+	* pePostRem = (input) Remote CCSID of POST data				000000		52119700
4933+	* pePostLoc = (input) Local CCSID of POST data				000000		52119800

Line Number	<----- Source Specifications ----->	<----- Comments ----->	Do Num	Page Line	Change Date	Src Id	Seq Number
4934+	* peProtRem = (input) Remote CCSID of Protocol data				000000		52119900
4935+	* peProtLoc = (input) Local CCSID of Protocol data				000000		52120000
4936+	*				000000		52120100
4937+	* Returns 0 if successful, -1 otherwise				000000		52120200
4938+	*+++++				000000		52120300
4939+D	HTTP_SetCCSIDs PR 10I 0				000000		52120400
4940+D	pePostRem 10I 0 value				000000		52120500
4941+D	pePostLoc 10I 0 value				000000		52120600
4942+D	peProtRem 10I 0 value options(*nopass)				000000		52120700
4943+D	peProtLoc 10I 0 value options(*nopass)				000000		52120800
4944+	*				000000		52120900
4945+	*				000000		52121000
4946+	*+++++				000000		52121100
4947+	* HTTP_SetTables(): Set the translation tables used for				000000		52121200
4948+	* ASCII/EBCDIC translation				000000		52121300
4949+	*				000000		52121400
4950+	* peASCII = (input) Table for converting to ASCII				000000		52121500
4951+	* peEBCDIC = (input) Table for converting to EBCDIC				000000		52121600
4952+	*				000000		52121700
4953+	* Returns 0 if successful, -1 otherwise				000000		52121800
4954+	*+++++				000000		52121900
4955+D	HTTP_SetTables PR 10I 0				000000		52122000
4956+D	peASCII 10A const				000000		52122100
4957+D	peEBCDIC 10A const				000000		52122200
4958+	*				000000		52122300
4959+	*				000000		52122400
4960+	*+++++				000000		52122500
4961+	* HTTP_SetFileCCSID(): Set the CCSID that downloaded stream				000000		52122600
4962+	* files get tagged with				000000		52122700
4963+	*				000000		52122800
4964+	* peCCSID = (input) New CCSID to assign				000000		52122900
4965+	*				000000		52123000
4966+	* NOTE: HTTPAPI does not do *any* translation of downloaded				000000		52123100
4967+	* data. It only sets this number as part of the file's				000000		52123200
4968+	* attributes. You can change it with the CHGATR CL				000000		52123300
4969+	* command.				000000		52123400
4970+	*				000000		52123500
4971+	* NOTE: The IFS did not support CCSIDs in V4R5 and earlier.				000000		52123600
4972+	* On those releases, this API will be used to set the				000000		52123700
4973+	* codepage rather than the CCSID.				000000		52123800
4974+	*				000000		52123900
4975+	* Returns 0 if successful, -1 otherwise				000000		52124000
4976+	*+++++				000000		52124100
4977+D	HTTP_SetfileCCSID...				000000		52124200
4978+D	PR				000000		52124300
4979+D	peCCSID 10I 0 value				000000		52124400
4980+	*				000000		52124500
4981+	*				000000		52124600
4982+	*+++++				000000		52124700
4983+	* HTTP_xlate(): Translate data from ASCII <--> EBCDIC				000000		52124800
4984+	*				000000		52124900
4985+	* peSize = (input) Size of data to translate				000000		52125000
4986+	* peData = (input) Data				000000		52125100
4987+	* peDirection = (input) can be set to the TO_ASCII or				000000		52125200
4988+	* TO_EBCDIC constant.				000000		52125300
4989+	*				000000		52125400

Line Number	<----- Source Specifications -----><----- Comments ----->	Do Num	Page Line	Change Date	Src Id	Seq Number
4990+	* Returns 0 if successful, -1 upon failure			000000		52125500
4991+	*+++++			000000		52125600
4992+D	HTTP_xlate PR 10I 0			000000		52125700
4993+D	peSize 10I 0 value			000000		52125800
4994+D	peData 32766A options(*varsize)			000000		52125900
4995+D	peDirection 1A const			000000		52126000
4996+				000000		52126100
4997+				000000		52126200
4998+	*+++++			000000		52126300
4999+	* HTTP_xlatep(): Translate data from ASCII <--> EBCDIC			000000		52126400
5000+	* (using a pointer instead of a variable)			000000		52126500
5001+	*			000000		52126600
5002+	* peSize = (input) Size of data to translate			000000		52126700
5003+	* peData = (input) Data			000000		52126800
5004+	* peDirection = (input) can be set to the TO_ASCII or			000000		52126900
5005+	* TO_EBCDIC constant.			000000		52127000
5006+	*			000000		52127100
5007+	* Returns 0 if successful, -1 upon failure			000000		52127200
5008+	*+++++			000000		52127300
5009+D	HTTP_xlatep PR 10I 0			000000		52127400
5010+D	peSize 10I 0 value			000000		52127500
5011+D	peData * value			000000		52127600
5012+D	peDirection 1A const			000000		52127700
5013+				000000		52127800
5014+				000000		52127900
5015+	*+++++			000000		52128000
5016+	* HTTP_xlatedyn: Translate data from ASCII <--> EBCDIC			000000		52128100
5017+	* using a dynamically sized output buffer			000000		52128200
5018+	*			000000		52128300
5019+	* peSize = (input) size of data to translate			000000		52128400
5020+	* peData = (input) pointer to data to translate			000000		52128500
5021+	* peDirection = (input) TO_ASCII or TO_EBCDIC			000000		52128600
5022+	* peOutput = (output) address of newly allocated memory			000000		52128700
5023+	*			000000		52128800
5024+	* returns the length of the translated data or -1 upon failure			000000		52128900
5025+	*+++++			000000		52129000
5026+D	HTTP_xlatedyn PR 10I 0			000000		52129100
5027+D	peSize 10I 0 value			000000		52129200
5028+D	peData * value			000000		52129300
5029+D	peDirection 1A const			000000		52129400
5030+D	peOutput *			000000		52129500
5031+				000000		52129600
5032+				000000		52129700
5033+	*+++++			000000		52129800
5034+	* http_set_100_timeout(): Set value for 100-continue timeouts.			000000		52129900
5035+	*			000000		52130000
5036+	* This API lets you change the timeout value that HTTPAPI waits			000000		52130100
5037+	* for a "100 Continue" response from the server during a POST			000000		52130200
5038+	* request. Some servers send this response, others do not. By			000000		52130300
5039+	* default, HTTPAPI waits 0.5 seconds for a response, and if it's			000000		52130400
5040+	* not received, sends the POST.			000000		52130500
5041+	*			000000		52130600
5042+	* Set this to a low value (0.1?) when working with a server			000000		52130700
5043+	* that does not send a "100 continue" response. That way, you			000000		52130800
5044+	* don't have to wait for the timeout.			000000		52130900
5045+	*			000000		52131000

Line Number	<----- Source Specifications -----><----- Comments ----->	Do Num	Page Line	Change Date	Src Id	Seq Number
5046+	* Set it to a higher value (10.0?) when working with a server			000000		52131100
5047+	* whose responses are slower, but will definitely send the			000000		52131200
5048+	* "100". That way, HTTPAPI won't give up on the response and			000000		52131300
5049+	* get out of sync.			000000		52131400
5050+	*			000000		52131500
5051+	*+++++			000000		52131600
5052+D	http_set_100_timeout...			000000		52131700
5053+D	PR			000000		52131800
5054+D	peTimeout 10P 3 value			000000		52131900
5055+				000000		52132000
5056+				000000		52132100
5057+	*+++++			000000		52132200
5058+	* HTTP_xml_SetCCSIDs(): Set the CCSIDs used for ASCII/EBCDIC			000000		52132300
5059+	* translation for XML documents			000000		52132400
5060+	*			000000		52132500
5061+	* peRemote = (input) remote CCSID			000000		52132600
5062+	* peLocal = (input) local CCSID (can be 0 if you want			000000		52132700
5063+	* to use the CCSID of the current job)			000000		52132800
5064+	*			000000		52132900
5065+	* Returns 0 if successful, -1 otherwise			000000		52133000
5066+	*+++++			000000		52133100
5067+D	HTTP_xml_SetCCSIDs...			000000		52133200
5068+D	PR 10I 0			000000		52133300
5069+D	peRemote 10I 0 value			000000		52133400
5070+D	peLocal 10I 0 value			000000		52133500
5071+				000000		52133600
5072+				000000		52133700
5073+	*+++++			000000		52133800
5074+	* http_parse_xml_stmf(): Parse XML data directly from a stream file			000000		52133900
5075+	* (instead of downloading it from a server)			000000		52134000
5076+	*			000000		52134100
5077+	* peFile = (input) Stream file (in IFS) to read data from			000000		52134200
5078+	* peCCSID = (input) CCSID of stream file,			000000		52134300
5079+	* or HTTP_XML_CALC to attempt to calculate it			000000		52134400
5080+	* from the XML encoding			000000		52134500
5081+	* or HTTP_STMF_CALC to use the stream file's			000000		52134600
5082+	* CCSID attribute.			000000		52134700
5083+	* peStartProc = (input) call-back procedure to call at the start			000000		52134800
5084+	* of each XML element received.			000000		52134900
5085+	* peEndProc = (input) call-back procedure to call at the end			000000		52135000
5086+	* of each XML element received.			000000		52135100
5087+	* peUsrDta = (input) user-defined data that will be passed			000000		52135200
5088+	* to the call-back routine			000000		52135300
5089+	*			000000		52135400
5090+	* Returns -1 upon failure, 0 if successful			000000		52135500
5091+	*+++++			000000		52135600
5092+D	http_parse_xml_stmf...			000000		52135700
5093+D	PR 10I 0			000000		52135800
5094+D	peFile 32767A varying const options(*varsize)			000000		52135900
5095+D	peCCSID 10I 0 value			000000		52136000
5096+D	peStartProc * value procptr			000000		52136100
5097+D	peEndProc * value procptr			000000		52136200
5098+D	peUsrDta * value			000000		52136300
5099+				000000		52136400
5100+D	HTTP_XML_CALC C -1			000000		52136500
5101+D	HTTP_STMF_CALC C -2			000000		52136600

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
5102+					000000		52136700
5103+					000000		52136800
5104+	*+++++				000000		52136900
5105+	* http_header(): retrieve the value of an HTTP header				000000		52137000
5106+	*				000000		52137100
5107+	* name = (input) name of header to look for				000000		52137200
5108+	* pos = (input/optional) position of header if there's				000000		52137300
5109+	* more than one with the same name				000000		52137400
5110+	*				000000		52137500
5111+	* returns the value of the HTTP header, or '' if not found				000000		52137600
5112+	*+++++				000000		52137700
5113+D	http_header PR 32500A varying				000000		52137800
5114+D	name 256A varying const				000000		52137900
5115+D	pos 10I 0 value options(*nopass)				000000		52138000
5116+					000000		52138100
5117+					000000		52138200
5118+	*+++++				000000		52138300
5119+	* http_use_cookies(): Turns on/off HTTPAPI's cookie parsing and				000000		52138400
5120+	* caching routines.				000000		52138500
5121+	*				000000		52138600
5122+	* peSetting = (input) *ON = HTTPAPI will read and send cookies				000000		52138700
5123+	* *OFF = HTTPAPI will ignore cookies				000000		52138800
5124+	* (has no affect on cookies supplied				000000		52138900
5125+	* via an exit procedure)				000000		52139000
5126+	*+++++				000000		52139100
5127+D	http_use_cookies...				000000		52139200
5128+D	PR				000000		52139300
5129+D	peSetting 1N const				000000		52139400
5130+					000000		52139500
5131+					000000		52139600
5132+	*+++++				000000		52139700
5133+	* http_cookie_file(): Set the name of the file that HTTPAPI				000000		52139800
5134+	* will use to store cookies.				000000		52139900
5135+	*				000000		52140000
5136+	* peFilename = (input) Filename (IFS path) to store cookie				000000		52140100
5137+	* data into.				000000		52140200
5138+	*				000000		52140300
5139+	* If the filename is set to '', or if you do not call this API,				000000		52140400
5140+	* cookies will only be saved until the activation group is				000000		52140500
5141+	* reclaimed.				000000		52140600
5142+	*+++++				000000		52140700
5143+D	http_cookie_file...				000000		52140800
5144+D	PR				000000		52140900
5145+D	peFilename 256A varying const				000000		52141000
5146+					000000		52141100
5147+					000000		52141200
5148+	*+++++				000000		52141300
5149+	* http_conp(): Send a completion message				000000		52141400
5150+	*				000000		52141500
5151+	* peMessage = message to send.				000000		52141600
5152+	*+++++				000000		52141700
5153+D	http_comp PR				000000		52141800
5154+D	peMessage 256A const				000000		52141900
5155+					000000		52142000
5156+					000000		52142100
5157+	*+++++				000000		52142200

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
5158+	* http_diag(): Send a diagnostic message				000000		52142300
5159+	*				000000		52142400
5160+	* peMessage = message to send.				000000		52142500
5161+	*****				000000		52142600
5162+D	http_diag PR				000000		52142700
5163+D	peMessage 256A const				000000		52142800
5164+					000000		52142900
5165+					000000		52143000
5166+	*****				000000		52143100
5167+	* http_crash(): Send back an *ESCAPE message containing last				000000		52143200
5168+	* error found in HTTPAPI.				000000		52143300
5169+	*****				000000		52143400
5170+D	http_crash PR				000000		52143500
5171+					000000		52143600
5172+					000000		52143700
5173+	*****				000000		52143800
5174+	* http_tempfile(): Generate a unique temporary IFS file name				000000		52143900
5175+	*				000000		52144000
5176+	* returns the file name				000000		52144100
5177+	*****				000000		52144200
5178+D	http_tempfile PR 40A varying				000000		52144300
5179+					070329		52144400
5180+					070329		52144500
5181+	***** +				070329		52144600
5182+	* http_xmlns(): Enable XML Namespace processing				070329		52144700
5183+	*				070329		52144800
5184+	* peEnable = (input) *ON to enable parsing, *OFF to disable.				070329		52144900
5185+	* (it is disabled by default)				070329		52145000
5186+	***** +				070329		52145100
5187+D	http_xmlns PR				070329		52145200
5188+D	peEnable 1N const				070329		52145300
5189+					070816		52145400
5190+					070816		52145500
5191+	***** +				070816		52145600
5192+	* http_XmlReturnPtr(): XML End Element Handler should return a				070816		52145700
5193+	* pointer to the full element value instead of				070816		52145800
5194+	* returning a VARYING character string.				070816		52145900
5195+	* (VARYING is limited to 64k)				070816		52146000
5196+	*				070816		52146100
5197+	* peEnable = (input) *ON to return a pointer, *OFF to return				070816		52146200
5198+	* a VARYING string (*OFF = default)				070816		52146300
5199+	***** +				070816		52146400
5200+D	http_XmlReturnPtr...				070816		52146500
5201+D	PR				070816		52146600
5202+D	peEnable 1N const				070816		52146700
5203+					070912		52146800
5204+					070912		52146900
5205+	***** +				070912		52147000
5206+	* http_XmlStripCRLF(): Enable stripping of CRLF characters				070912		52147100
5207+	*				070912		52147200
5208+	* peEnable = (input) *ON to strip, *OFF to leave them in.				070912		52147300
5209+	* (they are stripped by default)				070912		52147400
5210+	*				070912		52147500
5211+	* Note: To simplify your XML string manipulations, HTTPAPI				070912		52147600
5212+	* strips CRLF characters from the response. If you would				070912		52147700
5213+	* prefer that they are left in the response, call this				070912		52147800

Line	Source Specifications	Comments	Do	Page	Change	Src	Seq
Number	1.....2.....3.....4.....5.....6.....7.....8.....9.....10		Num	Line	Date	Id	Number
5214+	* routine with a parameter of *OFF.				070912		52147900
5215+	*****				070912		52148000
5216+D	http_XmlStripCRLF...				070912		52148100
5217+D	PR				070912		52148200
5218+D	peEnable 1N const				070912		52148300
5219+					000000		52148400
5220+					000000		52148500
5221+	*****				000000		52148600
5222+	** Error codes that HTTP API can return				000000		52148700
5223+	*****				000000		52148800
5224+	** Invalid URL format				000000		52148900
5225+D	HTTP_BADURL C CONST(1)				000000		52149000
5226+	** Host not found (not a valid IP address, or DNS lookup failed)				000000		52149100
5227+D	HTTP_HOSTNF C CONST(2)				000000		52149200
5228+	** No memory available for a socket address structure				000000		52149300
5229+D	HTTP_SAALLO C CONST(3)				000000		52149400
5230+	** Unable to create a new socket				000000		52149500
5231+D	HTTP_SOCERR C CONST(4)				000000		52149600
5232+	** Unable to place the socket in non-blocking mode				000000		52149700
5233+D	HTTP_NONBLK C CONST(5)				000000		52149800
5234+	** Error when connecting to server				000000		52149900
5235+D	HTTP_BADCNN C CONST(6)				000000		52150000
5236+	** Timeout when connecting to server				000000		52150100
5237+D	HTTP_CNNTIMO C CONST(7)				000000		52150200
5238+	** Unable to translate request to ASCII				000000		52150300
5239+D	HTTP_SRXLATE C CONST(8)				000000		52150400
5240+	** Error occurred calling select() in the sendreq() procedure				000000		52150500
5241+D	HTTP_SRSELE C CONST(9)				000000		52150600
5242+	** Error occurred calling send() in the sendreq() procedure				000000		52150700
5243+	** (probably means you were disconnected from the server)				000000		52150800
5244+D	HTTP_SRXMIT C CONST(10)				000000		52150900
5245+	** Error occurred calling recv() in the recvresp() procedure				000000		52151000
5246+	** (probably means you were disconnected from the server)				000000		52151100
5247+D	HTTP_RRRECV C CONST(11)				000000		52151200
5248+	** Error occurred calling select() in the recvresp() procedure				000000		52151300
5249+D	HTTP_RRSELE C CONST(12)				000000		52151400
5250+	** HTTP response code logged (not an error, per se)				000000		52151500
5251+D	HTTP_RESP C CONST(13)				000000		52151600
5252+	** Error calling recv() in the recvdoc() procedure				000000		52151700
5253+D	HTTP_RDRECV C CONST(14)				000000		52151800
5254+	** Error calling select() in the recvdoc() procedure				000000		52151900
5255+D	HTTP_RDSELE C CONST(15)				000000		52152000
5256+	** Error calling user-specified procedure in the				000000		52152100
5257+	** recvdoc() procedure. (user proc must return full count)				000000		52152200
5258+D	HTTP_RDWERR C CONST(16)				000000		52152300
5259+	** Error occurred while calling send() in the senddoc() procedure				000000		52152400
5260+D	HTTP_SDSEND C CONST(17)				000000		52152500
5261+	** When receiving a 'chunked' document, only a partial chunk recvd				000000		52152600
5262+D	HTTP_BADCHK C CONST(18)				000000		52152700
5263+	** When receiving a 'chunked' document, couldnt read a chunk size.				000000		52152800
5264+D	HTTP_CHKSIK C CONST(19)				000000		52152900
5265+	** Unsupported transfer-encoding value				000000		52153000
5266+D	HTTP_XFRENC C CONST(20)				000000		52153100
5267+	** You dont have access to put a file there :)				000000		52153200
5268+D	HTTP_ACCES C CONST(21)				000000		52153300
5269+	** Error opening file to save data into.				000000		52153400

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
5270+D	HTTP_FDOPEN C	CONST(22)			000000		52153500
5271+	** Problem with the Application ID for the DCM				000000		52153600
5272+D	HTTP_GSKAPPID C	CONST(23)			000000		52153700
5273+	** Error setting auth type				000000		52153800
5274+D	HTTP_GSKATYP C	CONST(24)			000000		52153900
5275+	** Error initializing GSKit environment				000000		52154000
5276+D	HTTP_GSKENVI C	CONST(25)			000000		52154100
5277+	** Error opening GSKit environment				000000		52154200
5278+D	HTTP_GSKENVO C	CONST(26)			000000		52154300
5279+	** Error setting session type (client server server_auth)				000000		52154400
5280+D	HTTP_GSKSTYP C	CONST(27)			000000		52154500
5281+	** Error registering application w/DCM				000000		52154600
5282+D	HTTP_REGERR C	CONST(28)			000000		52154700
5283+	** Error open secure socket				000000		52154800
5284+D	HTTP_SSOPEN C	CONST(29)			000000		52154900
5285+	** Error setting SSL numeric file descriptor				000000		52155000
5286+D	HTTP_SSSNFD C	CONST(30)			000000		52155100
5287+	** Error setting SSL numeric timeout value				000000		52155200
5288+D	HTTP_SSSNTO C	CONST(31)			000000		52155300
5289+	** SSL handshake timed out				000000		52155400
5290+D	HTTP_SSTIMO C	CONST(32)			000000		52155500
5291+	** Select() failure in SendDoc()				000000		52155600
5292+D	HTTP_SDSELE C	CONST(33)			000000		52155700
5293+	** Select() failure in get_chunk_size()				000000		52155800
5294+D	HTTP_CSSELE C	CONST(34)			000000		52155900
5295+	** This app is not registered with digital cert mgr				000000		52156000
5296+D	HTTP_NOTREG C	CONST(35)			000000		52156100
5297+	** This URI needs authorization (user/pass)				000000		52156200
5298+D	HTTP_NDAUTH C	CONST(36)			000000		52156300
5299+	** Invalid HTTP authentication type				000000		52156400
5300+D	HTTP_ATHTYP C	CONST(37)			000000		52156500
5301+	** Error in value of an HTTP authentication string				000000		52156600
5302+D	HTTP_ATHVAL C	CONST(38)			000000		52156700
5303+	** Server didn't ask for authorizatin				000000		52156800
5304+D	HTTP_NOAUTH C	CONST(39)			000000		52156900
5305+	** HTTP server didn't send a "100 Continue" in response				000000		52157000
5306+	** to our POST request.				000000		52157100
5307+D	HTTP_PST417 C	CONST(40)			000000		52157200
5308+	** recvresp() timed out waiting for a server response code				000000		52157300
5309+D	HTTP_RRTIME C	CONST(41)			000000		52157400
5310+	** recvdoc() timed out waiting for more data				000000		52157500
5311+D	HTTP_RDRTIME C	CONST(42)			000000		52157600
5312+	** blockread() timed out waiting for more data				000000		52157700
5313+D	HTTP_BRTIME C	CONST(43)			000000		52157800
5314+	** blockread() error during recv() call				000000		52157900
5315+D	HTTP_BRRECV C	CONST(44)			000000		52158000
5316+	** blockread() error during select() call				000000		52158100
5317+D	HTTP_BRSELE C	CONST(45)			000000		52158200
5318+	** recvchunk() did not get the trailing CRLF chars				000000		52158300
5319+D	HTTP_RDCRLF C	CONST(46)			000000		52158400
5320+	** Invalid exit point registered with HTTP_Xproc()				000000		52158500
5321+D	HTTP_BADPNT C	CONST(47)			000000		52158600
5322+	** Error retrieving SSL protocol				000000		52158700
5323+D	HTTP_SSPROT C	CONST(48)			000000		52158800
5324+	** Unknown SSL protocol				000000		52158900
5325+D	HTTP_SSPUNK C	CONST(49)			000000		52159000

Line	Number	Source Specifications	Comments	Do	Page	Change	Src	Seq
	1	2	3	4	5	6	7	8
5326+	**	Error setting SSL protocol				000000	52159100	
5327+D	HTTP_SSPSET	C	CONST(50)			000000	52159200	
5328+	**	Out of memory				000000	52159300	
5329+D	HTTP_NOMEM	C	CONST(51)			000000	52159400	
5330+	**	Must give data in order to encode it				000000	52159500	
5331+D	HTTP_NODATA	C	CONST(52)			000000	52159600	
5332+	**	Pointer is invalid or already freed				000000	52159700	
5333+D	HTTP_INVPTR	C	CONST(53)			000000	52159800	
5334+	**	Not enough space to add encoded variable				000000	52159900	
5335+D	HTTP_NOSPAC	C	CONST(54)			000000	52160000	
5336+	**	Error calling send() API in BlockWrite()				000000	52160100	
5337+D	HTTP_BWSEND	C	CONST(55)			000000	52160200	
5338+	**	Error calling select() API in BlockWrite()				000000	52160300	
5339+D	HTTP_BWSELE	C	CONST(56)			000000	52160400	
5340+	**	Timeout waiting to send in BlockWrite()				000000	52160500	
5341+D	HTTP_BWTIME	C	CONST(57)			000000	52160600	
5342+	**	Lineread() had problem with recv() API				000000	52160700	
5343+D	HTTP_LRRECV	C	CONST(58)			000000	52160800	
5344+	**	Lineread() had problem with select() API				000000	52160900	
5345+D	HTTP_LRSELE	C	CONST(59)			000000	52161000	
5346+	**	Lineread() had timeout				000000	52161100	
5347+D	HTTP_LRTIME	C	CONST(60)			000000	52161200	
5348+	**	Procedure is no longer supported				000000	52161300	
5349+D	HTTP_NOTSUPP	C	CONST(61)			000000	52161400	
5350+	**	No communication driver defined				000000	52161500	
5351+D	HTTP_NOCDRIV	C	CONST(62)			000000	52161600	
5352+	**	Timeout sending data in blockwrite				000000	52161700	
5353+D	HTTP_BWTIMO	C	CONST(63)			000000	52161800	
5354+	**	Timeout sending data in blockwrite				000000	52161900	
5355+D	HTTP_SWCERR	C	CONST(64)			000000	52162000	
5356+	**	Timeout sending data in blockwrite				000000	52162100	
5357+D	HTTP_FDSTAT	C	CONST(65)			000000	52162200	
5358+	**	Error parsing XML data				000000	52162300	
5359+D	HTTP_XMLERR	C	CONST(66)			000000	52162400	
5360+	**	Error opening IFS file				000000	52162500	
5361+D	HTTP_IFOPEN	C	CONST(67)			000000	52162600	
5362+	**	Error with SSL keyring				000000	52162700	
5363+D	HTTP_GSKKEYF	C	CONST(68)			000000	52162800	
5364+	**	Must Use Table / Must not Use Table				000000	52162900	
5365+D	HTTP_MUTABLE	C	CONST(69)			000000	52163000	
5366+	**	Cookie file cant be written				000000	52163100	
5367+D	HTTP_CKDUMP	C	CONST(70)			000000	52163200	
5368+	**	Cookie file cant be read				000000	52163300	
5369+D	HTTP_CKOPEN	C	CONST(71)			000000	52163400	
5370+	**	Can't get stats on cookie file				000000	52163500	
5371+D	HTTP_CKSTAT	C	CONST(72)			000000	52163600	
5372+	**	Error converting CCSIDs				000000	52163700	
5373+D	HTTP_CONVERR	C	CONST(73)			000000	52163800	
5374+	**	Error setting stream file CCSID				000000	52163900	
5375+D	HTTP_SETATTR	C	CONST(74)			000000	52164000	
5376+	**	This Proxy server needs authorization (user/pass)				000000	52164100	
5377+D	HTTP_PXNDAUTH	C	CONST(75)			000000	52164200	
5378+						000000	52164300	
5379+						000000	52164400	
5380+	*****					000000	52164500	
5381+	*	HTTP WWW-Authentication types				000000	52164600	

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
5382+	*****				000000		52164700
5383+D	HTTP_AUTH_NONE...				000000		52164800
5384+D	C '0'				000000		52164900
5385+D	HTTP_AUTH_BASIC...				000000		52165000
5386+D	C '1'				000000		52165100
5387+D	HTTP_AUTH_MD5_DIGEST...				000000		52165200
5388+D	C '2'				000000		52165300
5389+					000000		52165400
5390+					000000		52165500
5391+	*****				000000		52165600
5392+	* HTTPAPI Exit points				000000		52165700
5393+	*****				000000		52165800
5394+	** Debug exit point: This is called when ASCII stream data is to be				000000		52165900
5395+	** to a log file. Here's the prototype for a				000000		52166000
5396+	** debug exit procedure:				000000		52166100
5397+	**				000000		52166200
5398+	** D debug_proto PR				000000		52166300
5399+	** D DataToLog * value				000000		52166400
5400+	** D Length 10I 0 value				000000		52166500
5401+	**				000000		52166600
5402+D	HTTP_POINT_DEBUG...				000000		52166700
5403+D	C 1				000000		52166800
5404+					000000		52166900
5405+	** Upload status exit point: This is called periodically during an				000000		52167000
5406+	** upload (POST) to an HTTP(S) server.				000000		52167100
5407+	** Allows you to display progress to the				000000		52167200
5408+	** user.				000000		52167300
5409+	**				000000		52167400
5410+	** D upload_proto PR				000000		52167500
5411+	** D BytesSent 10U 0 value				000000		52167600
5412+	** D BytesTotal 10U 0 value				000000		52167700
5413+	**				000000		52167800
5414+D	HTTP_POINT_UPLOAD_STATUS...				000000		52167900
5415+D	C 2				000000		52168000
5416+					000000		52168100
5417+	** Download status exit point: This is called periodically during a				000000		52168200
5418+	** download (POST or GET) from an HTTP(S)				000000		52168300
5419+	** server. Allows you to display the				000000		52168400
5420+	** progress to the user.				000000		52168500
5421+	**				000000		52168600
5422+	** D download_proto PR				000000		52168700
5423+	** D BytesRecv 10U 0 value				000000		52168800
5424+	** D BytesTotal 10U 0 value				000000		52168900
5425+	**				000000		52169000
5426+D	HTTP_POINT_DOWNLOAD_STATUS...				000000		52169100
5427+D	C 3				000000		52169200
5428+					000000		52169300
5429+	** Additional Header fields exit point:				000000		52169400
5430+	** Allows you to supply additional header data to be added				000000		52169500
5431+	** to the HTTP request chain. Data should be in EBCDIC with				000000		52169600
5432+	** x'0d25' after each header record.				000000		52169700
5433+	**				000000		52169800
5434+	** D addl_hdrs_prot PR				000000		52169900
5435+	** D HeaderData 1024A varying				000000		52170000
5436+	**				000000		52170100
5437+D	HTTP_POINT_ADDL_HEADER...				000000		52170200

Line	Source Specifications	Comments	Do	Page	Change	Src	Seq
Number	1...2...3...4...5...6...7...8...9...10		Num	Line	Date	Id	Number
5438+D	C				000000		52170300
5439+	4				000000		52170400
5440+ **	Header parse exit point:				000000		52170500
5441+ **	Allows you to examine the HTTP response chain received				000000		52170600
5442+ **	from the HTTP server.				000000		52170700
5443+ **					000000		52170800
5444+ **	D parse_hdr_prot PR				000000		52170900
5445+ **	D HeaderData 2048A const				000000		52171000
5446+ **					000000		52171100
5447+D	HTTP_POINT_PARSE_HEADER...				000000		52171200
5448+D	C				000000		52171300
5449+	5				000000		52171400
5450+ **	Header parse exit point:				000000		52171500
5451+ **	Allows you to examine the HTTP response chain received				000000		52171600
5452+ **	from the HTTP server. (allows longer headers)				000000		52171700
5453+ **					000000		52171800
5454+ **	D parse_hdr_long PR				000000		52171900
5455+ **	D HeaderData 32767A const varying				000000		52172000
5456+ **					000000		52172100
5457+D	HTTP_POINT_PARSE_HDR_LONG...				000000		52172200
5458+D	C				000000		52172300
5459+	6				000000		52172400
5460+					000000		52172500
5461+ *****					000000		52172600
5462+ *	Directions for HTTP_xlate() and HTTP_xlatep()				000000		52172700
5463+ *****					000000		52172800
5464+D	TO_ASCII C '1'				000000		52172900
5465+D	TO_EBCDIC C '2'				000000		52173000
5466	/COPY QWSDL2RPG,PWSDL2R94 WSDL2RPG: MultiRef		CPY		000000		000330

* RPG member name :	PWSDL2R94						54
* External name :	WSDL2RPG/QWSDL2RPG(PWSDL2R94)						54
* Last change :	07/29/09 10:43:00						54
* Text 'description' :	WSDL2RPG: MultiRef (Prototype)						54

5467+ /IF NOT DEFINED(PWSDL2R94)		RADDAT			580131		54000100
5468+ /DEFINE PWSDL2R94		RADDAT			580131		54000200
5469+ *		RADDAT			571108		54000300
5470+ /COPY QWSDL2RPG,TYPES		RADDAT	CPY		571108		54000400

* RPG member name :	TYPES						55
* External name :	WSDL2RPG/QWSDL2RPG(TYPES)						55
* Last change :	07/29/09 10:43:01						55
* Text 'description' :	Type definitions						55

5471+ /IF NOT DEFINED(TYPES)		RADDAT			571106		55000100
5472+ /DEFINE TYPES		RADDAT			571106		55000200
5473+ *****		RADDAT			571126		55000300
5474+ *	Important Information:	RADDAT			571126		55000400
5475+ *	All structures and fields are 16-byte aligned to avoid	RADDAT			571126		55000500
5476+ *	pointer problems.	RADDAT			571126		55000600
5477+ *****		RADDAT			571126		55000700
5478+ /COPY QBASICS1,PBASICS1	General Purpose Function Library	RADDAT	CPY		571121		55000800

* RPG member name :	PBASICS1						56
* External name :	WSDL2RPG/QBASICS1(PBASICS1)						56

Line Number	<----- Source Specifications ----->	<----- Comments ----->	Do Num	Page Line	Change Date	Src Id	Seq Number
1....+....2....+....3....+....4....+....5....+....6....+....7....+....8....+....9....+...10						
	* Last change : 08/19/09 12:46:31	*				56	
	* Text 'description' : SrvPgm: Basic Routines - Include cpybk	*				56	

5479+	/IF NOT DEFINED(D_BASIC) LINES EXCLUDED: 37	RADDAT			000000		56000100
5480+	/ENDIF	RADDAT			000000		56003900
5481+	*	RADDAT			571206		55000900
5482+	* The logger for wsdl2rpg:	RADDAT			571206		55001000
5483+D	cLOGGER C 'WSDL2RPG.client'	RADDAT			580905		55001100
5484+D	cLOGGER_NAMESPACES...	RADDAT			571219		55001200
5485+D	C '.namespaces'	RADDAT			580905		55001300
5486+D	cLOGGER_RPG_ADAPTER...	RADDAT			571219		55001400
5487+D	C '.rpgAdapter'	RADDAT			580905		55001500
5488+D	cLOGGER_TYPES...	RADDAT			580905		55001600
5489+D	C '.types'	RADDAT			580905		55001700
5490+D	cLOGGER_OPERATION...	RADDAT			580114		55001800
5491+D	C '.operation'	RADDAT			580905		55001900
5492+D	cLOGGER_XML_ARRAY...	RADDAT			571214		55002000
5493+D	C '.types.array.xml'	RADDAT			580905		55002100
5494+D	cLOGGER_RPG_ARRAY...	RADDAT			580905		55002200
5495+D	C '.types.array.rpg'	RADDAT			580905		55002300
5496+D	cLOGGER_MIME_HEADERS...	RADDAT			580905		55002400
5497+D	C '.mimeHeaders'	RADDAT			580905		55002500
5498+D	cLOGGER_STREAM...	RADDAT			580905		55002600
5499+D	C '.stream'	RADDAT			580905		55002700
5500+D	cLOGGER_MANAGED_MEMORY_DATA_SOURCE...	RADDAT			580905		55002800
5501+D	C '.managedMemoryDataSource'	RADDAT			580905		55002900
5502+D	cLOGGER_MANAGED_MEMORY_DATA_SOURCE_ICONV...	RADDAT			580905		55003000
5503+D	C '.managedMemoryDataSource.iconv'	RADDAT			580905		55003100
5504+D	cLOGGER_ATTACHMENT_PART...	RADDAT			580905		55003200
5505+D	C '.attachmentPart'	RADDAT			580905		55003300
5506+D	cLOGGER_SOAP_MESSAGE...	RADDAT			580905		55003400
5507+D	C '.soapMessage'	RADDAT			580905		55003500
5508+D	cLOGGER_SOAP_MESSAGE_DATA...	RADDAT			580905		55003600
5509+D	C '.soapMessage.data'	RADDAT			580905		55003700
5510+D	cLOGGER_APPLICATION_MESSAGE...	RADDAT			571219		55003800
5511+D	C '.appMsg'	RADDAT			580905		55003900
5512+D	cLOGGER_QUALIFIED_NAME...	RADDAT			571219		55004000
5513+D	C '.qName'	RADDAT			580905		55004100
5514+D	cLOGGER_TEMPFILE...	RADDAT			580515		55004200
5515+D	C '.TempFile'	RADDAT			580905		55004300
5516+D	cLOGGER_COMMON...	RADDAT			571219		55004400
5517+D	C '.common'	RADDAT			580905		55004500
5518+	*	RADDAT			571219		55004600
5519+D	cLOGGER_EXPAT_ADAPTER...	RADDAT			571219		55004700
5520+D	C 'WSDL2RPG.eXpatAdapter'	RADDAT			580905		55004800
5521+	*	RADDAT			571204		55004900
5522+	* WSDL command parameter	RADDAT			571204		55005000
5523+	* ... holds the proxy server and port	RADDAT			571204		55005100
5524+D	wsdl_cmdServer_t...	RADDAT			571204		55005200
5525+D	DS qualified based(pDummy)	RADDAT			571204		55005300
5526+D	numE 5I 0	RADDAT			571204		55005400
5527+D	proxyServer likeds(wsdl_qServer_t)	RADDAT			571204		55005500
5528+	*	RADDAT			571204		55005600
5529+	* ... holds the selected web service port and operations	RADDAT			571204		55005700
5530+D	wsdl_webService_t...	RADDAT			571204		55005800

Line	----- Source Specifications ----->										Comments	Do	Page	Change	Src	Seq
Number	1	2	3	4	5	6	7	8	9	10		Num	Line	Date	Id	Number
5531+D		DS		qualified			based(pDummy)				RADDAT			571204		55005900
5532+D	numParms		5I	0							RADDAT			571204		55006000
5533+D	port		256A	varying							RADDAT			571204		55006100
5534+D	ofsOperNmcs		5I	0							RADDAT			571204		55006200
5535+D	numOperations		5I	0							RADDAT			571204		55006300
5536+D	operations			likeds(wsdl_operationsX_t)							RADDAT			571204		55006400
5537+ *											RADDAT			571219		55006500
5538+ *	... holds the output source member options										RADDAT			571219		55006600
5539+D	wsdl_srcMbrOpts_t...										RADDAT			571219		55006700
5540+D		DS		qualified			based(pDummy)				RADDAT			571219		55006800
5541+D	numParms		5I	0							RADDAT			571219		55006900
5542+D	name		10A								RADDAT			571219		55007000
5543+D	doReplace		N								RADDAT			571219		55007100
5544+ *											RADDAT			571204		55007200
5545+ *	... holds the selected operations of the WSDL command										RADDAT			571204		55007300
5546+D	wsdl_operationsX_t...										RADDAT			571204		55007400
5547+D		DS		qualified			based(pDummy)				RADDAT			571204		55007500
5548+D	name		256A	dim(cMAX_OPERATIONS)	varying						RADDAT			571204		55007600
5549+ *											RADDAT			571204		55007700
5550+D	cMAX_OPERATIONS...										RADDAT			571204		55007800
5551+D		C		128							RADDAT			571204		55007900
5552+ *											RADDAT			580905		55008000
5553+ *	Holds the content type of a ManagedMemoryDataSource or AttachmentPart.										RADDAT			580905		55008100
5554+D	wsdl_contentType_t...										RADDAT			580905		55008200
5555+D		S		1024A	varying		based(pDummy)				RADDAT			580905		55008300
5556+ *											RADDAT			571108		55008400
5557+ *	Holds the name of IFS stream file.										RADDAT			571108		55008500
5558+D	wsdl_path_t...										RADDAT			571106		55008600
5559+D		S		254A	varying		based(pDummy)				RADDAT			571126		55008700
5560+ *											RADDAT			571108		55008800
5561+ *	Holds a name such as 'xsd:String' or a RPG procedure name.										RADDAT			571126		55008900
5562+D	wsdl_name_t	S		4094A	varying		based(pDummy)				RADDAT			571126		55009000
5563+ *											RADDAT			571123		55009100
5564+ *	Holds a (attribute) value.										RADDAT			571229		55009200
5565+D	wsdl_value_t	S		4094A	varying		based(pDummy)				RADDAT			571229		55009300
5566+ *											RADDAT			571229		55009400
5567+ *	Holds a key of a type. (length must be lName + ns!)										RADDAT			580529		55009500
5568+D	wsdl_mapKey_t	S		1022A	varying						RADDAT			580529		55009600
5569+D							based(pDummy)				RADDAT			571123		55009700
5570+ *	Holds a simple URL such as 'http://server:8080/WebService/Service.wsdl'										RADDAT			571207		55009800
5571+D	wsdl_url_t	S		254A	varying		based(pDummy)				RADDAT			571130		55009900
5572+ *											RADDAT			571126		55010000
5573+ *	Holds a namespace prefix such as 'xsd'.										RADDAT			571126		55010100
5574+D	wsdl_nsPrefix_t...										RADDAT			571126		55010200
5575+D		S		30A	varying		based(pDummy)				RADDAT			571126		55010300
5576+ *											RADDAT			571108		55010400
5577+ *	Holds an error code.										RADDAT			571108		55010500
5578+D	wsdl_error_t...										RADDAT			571108		55010600
5579+D		S		10I	0		based(pDummy)				RADDAT			571108		55010700
5580+ *											RADDAT			571126		55010800
5581+ *	Holds a namespace name such as: http://schemas.xmlsoap.org/wsdl/										RADDAT			571126		55010900
5582+D	wsdl_lName_t...										RADDAT			571126		55011000
5583+D		S		510A	varying		based(pDummy)				RADDAT			571206		55011100
5584+ *											RADDAT			571126		55011200
5585+ *	Holds a namespace qualifier										RADDAT			571126		55011300
5586+D	wsdl_nsValue_t...										RADDAT			580531		55011400

Line	----- Source Specifications -----><----- Comments ----->										Do	Page	Change	Src	Seq
Number	1	2	3	4	5	6	7	8	9	10	Num	Line	Date	Id	Number
5587+D		S		510A	varying		based(pDummy)			RADDAT			571206		55011500
5588+ *										RADDAT			571205		55011600
5589+ *	Holds the dimension of a type.												571205		55011700
5590+D	wsdl_dimXml_t...									RADDAT			571206		55011800
5591+D		S		30A	varying		based(pDummy)			RADDAT			571206		55011900
5592+D	wsdl_dim_t...									RADDAT			571206		55012000
5593+D		S		10I 0			based(pDummy)			RADDAT			571206		55012100
5594+ *										RADDAT			571206		55012200
5595+D	cDIM_LENGTH	C			%len(wsdl_dim_t)					RADDAT			571206		55012300
5596+D	cDIM_DIGITS	C			%decpos(wsdl_dim_t)					RADDAT			571206		55012400
5597+D	cDIM_XML_UNBOUNDED...									RADDAT			571206		55012500
5598+D		C			'unbounded'					RADDAT			571206		55012600
5599+ *										RADDAT			571125		55012700
5600+ *	Holds a decimal value. (see also: UnMarshaller)												571125		55012800
5601+D	wsdl_decimal_t...									RADDAT			571125		55012900
5602+D		S		31S15			based(pDummy)			RADDAT			571125		55013000
5603+ *										RADDAT			571125		55013100
5604+D	cDECIMAL_TYPE...									RADDAT			571125		55013200
5605+D		C			'S'					RADDAT			571125		55013300
5606+D	cDECIMAL_LEN...									RADDAT			571125		55013400
5607+D		C			%len(wsdl_decimal_t)					RADDAT			571125		55013500
5608+D	cDECIMAL_DECPOS...									RADDAT			571125		55013600
5609+D		C			%decpos(wsdl_decimal_t)					RADDAT			571125		55013700
5610+ *										RADDAT			571221		55013800
5611+ *	Holds a double value. (see also: UnMarshaller)												571221		55013900
5612+D	wsdl_double_t...									RADDAT			571221		55014000
5613+D		S			like(wsdl_decimal_t)		based(pDummy)			RADDAT			571221		55014100
5614+ *										RADDAT			571221		55014200
5615+D	cDOUBLE_TYPE...									RADDAT			571221		55014300
5616+D		C			'S'					RADDAT			571221		55014400
5617+D	cDOUBLE_LEN...									RADDAT			571221		55014500
5618+D		C			%len(wsdl_double_t)					RADDAT			571221		55014600
5619+D	cDOUBLE_DECPOS...									RADDAT			571221		55014700
5620+D		C			%decpos(wsdl_double_t)					RADDAT			571221		55014800
5621+ *										RADDAT			580319		55014900
5622+ *	Holds a float value. (see also: UnMarshaller)												580319		55015000
5623+D	wsdl_float_t...									RADDAT			580319		55015100
5624+D		S			like(wsdl_decimal_t)		based(pDummy)			RADDAT			580319		55015200
5625+ *										RADDAT			580319		55015300
5626+D	cFLOAT_TYPE...									RADDAT			580319		55015400
5627+D		C			'S'					RADDAT			580319		55015500
5628+D	cFLOAT_LEN...									RADDAT			580319		55015600
5629+D		C			%len(wsdl_float_t)					RADDAT			580319		55015700
5630+D	cFLOAT_DECPOS...									RADDAT			580319		55015800
5631+D		C			%decpos(wsdl_float_t)					RADDAT			580319		55015900
5632+ *										RADDAT			571125		55016000
5633+ *	Holds a short integer value. (see also: UnMarshaller)												580121		55016100
5634+D	wsdl_short_t...									RADDAT			580121		55016200
5635+D		S		5I 0			based(pDummy)			RADDAT			580121		55016300
5636+ *										RADDAT			580121		55016400
5637+D	cSHORT_TYPE...									RADDAT			580121		55016500
5638+D		C			'I'					RADDAT			580121		55016600
5639+D	cSHORT_LEN...									RADDAT			580121		55016700
5640+D		C			%len(wsdl_short_t)					RADDAT			580121		55016800
5641+D	cSHORT_DECPOS...									RADDAT			580121		55016900
5642+D		C			%decpos(wsdl_short_t)					RADDAT			580121		55017000

Line	Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
5643+	*					580905		55017100
5644+	*	Holds an integer value. (see also: UnMarshaller)				580905		55017200
5645+D	wsdl_integer_t...					580905		55017300
5646+D	S	10I 0	based(pDummy)			580905		55017400
5647+	*					580905		55017500
5648+D	cINTEGER_TYPE...					580905		55017600
5649+D	C	'I'				580905		55017700
5650+D	cINTEGER_LEN...					580905		55017800
5651+D	C	%len(wsdl_integer_t)				580905		55017900
5652+D	cINTEGER_DECPOS...					580905		55018000
5653+D	C	%decpos(wsdl_integer_t)				580905		55018100
5654+	*					580905		55018200
5655+	*	Holds a long integer value. (see also: UnMarshaller)				580905		55018300
5656+D	wsdl_long_t...					580905		55018400
5657+D	S	20I 0	based(pDummy)			580905		55018500
5658+	*					580905		55018600
5659+D	cLONG_TYPE...					580905		55018700
5660+D	C	'I'				580905		55018800
5661+D	cLONG_LEN...					580905		55018900
5662+D	C	%len(wsdl_long_t)				580905		55019000
5663+D	cLONG_DECPOS...					580905		55019100
5664+D	C	%decpos(wsdl_long_t)				580905		55019200
5665+	*					580905		55019300
5666+	*	Holds an unsigned short integer value. (see also: UnMarshaller)				580905		55019400
5667+D	wsdl_uShort_t...					580905		55019500
5668+D	S	5U 0	based(pDummy)			581104		55019600
5669+	*					580905		55019700
5670+D	cUSHORT_TYPE...					580905		55019800
5671+D	C	'U'				580905		55019900
5672+D	cUSHORT_LEN...					580905		55020000
5673+D	C	%len(wsdl_uShort_t)				580905		55020100
5674+D	cUSHORT_DECPOS...					580905		55020200
5675+D	C	%decpos(wsdl_uShort_t)				580905		55020300
5676+	*					580905		55020400
5677+	*	Holds an unsigned integer value. (see also: UnMarshaller)				580905		55020500
5678+D	wsdl_uInteger_t...					580905		55020600
5679+D	S	10U 0	based(pDummy)			581104		55020700
5680+	*					580905		55020800
5681+D	cUIINTEGER_TYPE...					580905		55020900
5682+D	C	'U'				580905		55021000
5683+D	cUIINTEGER_LEN...					580905		55021100
5684+D	C	%len(wsdl_uInteger_t)				580905		55021200
5685+D	cUIINTEGER_DECPOS...					580905		55021300
5686+D	C	%decpos(wsdl_uInteger_t)				580905		55021400
5687+	*					580905		55021500
5688+	*	Holds a long integer value. (see also: UnMarshaller)				580905		55021600
5689+D	wsdl_uLong_t...					580905		55021700
5690+D	S	20U 0	based(pDummy)			581104		55021800
5691+	*					580905		55021900
5692+D	cULONG_TYPE...					580905		55022000
5693+D	C	'U'				580905		55022100
5694+D	cULONG_LEN...					580905		55022200
5695+D	C	%len(wsdl_uLong_t)				580905		55022300
5696+D	cULONG_DECPOS...					580905		55022400
5697+D	C	%decpos(wsdl_uLong_t)				580905		55022500
5698+	*					580905		55022600

Line	----- Source Specifications -----><----- Comments ----->										Do	Page	Change	Src	Seq
Number1....+....2....+....3....+....4....+....5....+....6....+....7....+....8....+....9....+....10										Num	Line	Date	Id	Number
5699+ *	Holds a non negative integer value. (see also: UnMarshaller)												RADDAT	580905	55022700
5700+D	wsdl_nonNegInt_t...												RADDAT	580905	55022800
5701+D	S	10I	0						based(pDummy)				RADDAT	580905	55022900
5702+ *													RADDAT	580905	55023000
5703+D	cNON_NEG_INT_TYPE...												RADDAT	580905	55023100
5704+D	C							'I'					RADDAT	580905	55023200
5705+D	cNON_NEG_INT_LEN...												RADDAT	580905	55023300
5706+D	C							%len(wsdl_nonNegInt_t)					RADDAT	580905	55023400
5707+D	cNON_NEG_INT_DECPOS...												RADDAT	580905	55023500
5708+D	C							%decpos(wsdl_nonNegInt_t)					RADDAT	580905	55023600
5709+ *													RADDAT	580905	55023700
5710+ *	Holds a non positive integer value. (see also: UnMarshaller)												RADDAT	580905	55023800
5711+D	wsdl_nonPosInt_t...												RADDAT	580905	55023900
5712+D	S	10I	0						based(pDummy)				RADDAT	580905	55024000
5713+ *													RADDAT	580905	55024100
5714+D	cNON_POS_INT_TYPE...												RADDAT	580905	55024200
5715+D	C							'I'					RADDAT	580905	55024300
5716+D	cNON_POS_INT_LEN...												RADDAT	580905	55024400
5717+D	C							%len(wsdl_nonPosInt_t)					RADDAT	580905	55024500
5718+D	cNON_POS_INT_DECPOS...												RADDAT	580905	55024600
5719+D	C							%decpos(wsdl_nonPosInt_t)					RADDAT	580905	55024700
5720+ *													RADDAT	580905	55024800
5721+ *	Holds a negative integer value. (see also: UnMarshaller)												RADDAT	580905	55024900
5722+D	wsdl_negInt_t...												RADDAT	580905	55025000
5723+D	S	10I	0						based(pDummy)				RADDAT	580905	55025100
5724+ *													RADDAT	580905	55025200
5725+D	cNEG_INT_TYPE...												RADDAT	580905	55025300
5726+D	C							'I'					RADDAT	580905	55025400
5727+D	cNEG_INT_LEN...												RADDAT	580905	55025500
5728+D	C							%len(wsdl_negInt_t)					RADDAT	580905	55025600
5729+D	cNEG_INT_DECPOS...												RADDAT	580905	55025700
5730+D	C							%decpos(wsdl_negInt_t)					RADDAT	580905	55025800
5731+ *													RADDAT	580905	55025900
5732+ *	Holds a positive integer value. (see also: UnMarshaller)												RADDAT	580905	55026000
5733+D	wsdl_posInt_t...												RADDAT	580905	55026100
5734+D	S	10I	0						based(pDummy)				RADDAT	580905	55026200
5735+ *													RADDAT	580905	55026300
5736+D	cPOS_INT_TYPE...												RADDAT	580905	55026400
5737+D	C							'I'					RADDAT	580905	55026500
5738+D	cPOS_INT_LEN...												RADDAT	580905	55026600
5739+D	C							%len(wsdl_posInt_t)					RADDAT	580905	55026700
5740+D	cPOS_INT_DECPOS...												RADDAT	580905	55026800
5741+D	C							%decpos(wsdl_posInt_t)					RADDAT	580905	55026900
5742+ *													RADDAT	590613	55027000
5743+ *	Holds a duration value. (see also: UnMarshaller)												RADDAT	590613	55027100
5744+D	wsdl_duration_t...												RADDAT	590613	55027200
5745+D	DS							qualified					RADDAT	590613	55027300
5746+D	isNegative							N					RADDAT	590613	55027400
5747+D	years	10I	0										RADDAT	590613	55027500
5748+D	months	10I	0										RADDAT	590613	55027600
5749+D	days	10I	0										RADDAT	590613	55027700
5750+D	hours	10I	0										RADDAT	590613	55027800
5751+D	minutes	10I	0										RADDAT	590613	55027900
5752+D	seconds	10I	0										RADDAT	590613	55028000
5753+ *													RADDAT	571207	55028100
5754+ *	Holds the SOAP action of a Web Service operation												RADDAT	571207	55028200

Line	----- Source Specifications -----><----- Comments ----->								Do	Page	Change	Src	Seq
Number1....+....2....+....3....+....4....+....5....+....6....+....7....+....8....+....9....+....10								Num	Line	Date	Id	Number
5755+D wsdl_soapAction_t...								RADDAT			571207		55028300
5756+D S		510A	varying			based(pDummy)		RADDAT			571208		55028400
5757+ *								RADDAT			571126		55028500
5758+ * Holds the parameter order of a Web Service operation								RADDAT			571207		55028600
5759+D wsdl_parameterOrder_t...								RADDAT			571207		55028700
5760+D S			like(wsdl_name_t)			based(pDummy)		RADDAT			571207		55028800
5761+ *								RADDAT			571207		55028900
5762+ * Holds the protocol specified for a web service port.								RADDAT			571207		55029000
5763+D wsdl_protocol_t...								RADDAT			571207		55029100
5764+D S		14A	varying			based(pDummy)		RADDAT			571207		55029200
5765+ *								RADDAT			571207		55029300
5766+ * Holds the documentation of any element.								RADDAT			571208		55029400
5767+D wsdl_documentation_t...								RADDAT			571208		55029500
5768+D S		4094A	varying			based(pDummy)		RADDAT			580116		55029600
5769+ *								RADDAT			571229		55029700
5770+ * Specifies what encoding to use.								RADDAT			571229		55029800
5771+D wsdl_use_t...								RADDAT			571229		55029900
5772+D S		14A	varying			based(pDummy)		RADDAT			571229		55030000
5773+ *								RADDAT			571208		55030100
5774+ * Specifies the encoding style of a binding.								RADDAT			571229		55030200
5775+D wsdl_style_t...								RADDAT			571229		55030300
5776+D S		14A	varying			based(pDummy)		RADDAT			571229		55030400
5777+ *								RADDAT			580319		55030500
5778+ * Specifies the type of a type definition (*ELEM/*TYPE)								RADDAT			580319		55030600
5779+D wsdl_defType_t...								RADDAT			580319		55030700
5780+D S		5A				based(pDummy)		RADDAT			580319		55030800
5781+ *								RADDAT			571229		55030900
5782+ * Holds the name and port of a server.								RADDAT			571126		55031000
5783+D wsdl_qServer_t...								RADDAT			571126		55031100
5784+D DS			qualified			based(pDummy)		RADDAT			571126		55031200
5785+D name		126A	varying					RADDAT			571126		55031300
5786+D port		10I 0						RADDAT			571126		55031400
5787+D reserved_1		12A						RADDAT			590610		55031500
5788+ *								RADDAT			571205		55031600
5789+ * Structure to hold all services of a given WSDL file.								RADDAT			571205		55031700
5790+D wsdl_services_t...								RADDAT			571205		55031800
5791+D DS			qualified			based(pDummy)		RADDAT			571205		55031900
5792+D serviceList			likeds(wsdl_map_t)					RADDAT			571205		55032000
5793+ *								RADDAT			571205		55032100
5794+ * Structure to hold a WSDL service definition								RADDAT			571205		55032200
5795+D wsdl_service_t...								RADDAT			571205		55032300
5796+D DS			qualified			based(pDummy)		RADDAT			571205		55032400
5797+D qName			likeds(wsdl_qName_t)					RADDAT			571205		55032500
5798+D portList			likeds(wsdl_map_t)					RADDAT			571205		55032600
5799+ *								RADDAT			571205		55032700
5800+ * Structure to hold all types of a given WSDL file.								RADDAT			571205		55032800
5801+D wsdl_types_t...								RADDAT			571205		55032900
5802+D DS			qualified			based(pDummy)		RADDAT			571205		55033000
5803+D typesList			likeds(wsdl_map_t)					RADDAT			571205		55033100
5804+ *								RADDAT			571108		55033200
5805+ * Structure to hold a WSDL type definition								RADDAT			571108		55033300
5806+D wsdl_type_t...								RADDAT			571108		55033400
5807+D DS			qualified			based(pDummy)		RADDAT			571108		55033500
5808+D isChoice		N					1 byte	RADDAT			581101		55033600
5809+D isUsed		N					+ 1 byte	RADDAT			581101		55033700
5810+D isSequence		N					+ 1 byte	RADDAT			581101		55033800

Line	----- Source Specifications -----										><----- Comments ----->		Do	Page	Change	Src	Seq
Number	1	2	3	4	5	6	7	8	9	10	Num	Line	Date	Id	Number		
5811+D	isQualified		N					+ 1 byte	RADDAT			590417		55033900			
5812+D	referencesType...								RADDAT			581101		55034000			
5813+D			N					+ 1 byte	RADDAT			581101		55034100			
5814+D	arrayType		10A	varying				+12 byte	RADDAT			581101		55034200			
5815+D	arrayID		10I	0				+ 4 byte	RADDAT			581101		55034300			
5816+D	dim			like(wsdl_dim_t)			+ 4 byte	RADDAT			581101		55034400			
5817+D	defType			like(wsdl_defType_t)			+ 5 byte	RADDAT			581101		55034500			
5818+D	reserved_1		2A					+ 2 byte = 32	RADDAT			590612		55034600			
5819+D	qName			likeds(wsdl_qName_t)				RADDAT			571126		55034700			
5820+D	qType			likeds(wsdl_qName_t)				RADDAT			571126		55034800			
5821+D	subTypesList			likeds(wsdl_map_t)				RADDAT			571126		55034900			
5822+ *									RADDAT			571213		55035000			
5823+D	cARRAY_NONE...								RADDAT			571213		55035100			
5824+D		C		'_'					RADDAT			571213		55035200			
5825+D	cARRAY_XML...								RADDAT			571213		55035300			
5826+D		C		'XML'					RADDAT			571213		55035400			
5827+D	cARRAY_RPG...								RADDAT			571213		55035500			
5828+D		C		'RPG'					RADDAT			571213		55035600			
5829+D	cARRAY_VALUE...								RADDAT			571213		55035700			
5830+D		C		'ARY-VALUE'					RADDAT			571213		55035800			
5831+D	cARRAY_INDEX...								RADDAT			571213		55035900			
5832+D		C		'ARY-INDEX'					RADDAT			571213		55036000			
5833+D	cTYPE_REFERENCE...								RADDAT			580110		55036100			
5834+D		C		'*REFERENCE'					RADDAT			580110		55036200			
5835+ *									RADDAT			571205		55036300			
5836+ *	Structure to hold all messages of a given WSDL file.								RADDAT			571205		55036400			
5837+D	wsdl_messages_t...								RADDAT			571205		55036500			
5838+D		DS		qualified		based(pDummy)			RADDAT			571205		55036600			
5839+D	msgList			likeds(wsdl_map_t)				RADDAT			571205		55036700			
5840+ *									RADDAT			571126		55036800			
5841+ *	Structure to hold a WSDL message definition								RADDAT			571126		55036900			
5842+D	wsdl_message_t...								RADDAT			571126		55037000			
5843+D		DS		qualified		based(pDummy)			RADDAT			571126		55037100			
5844+D	qName			likeds(wsdl_qName_t)				RADDAT			571126		55037200			
5845+D	partsList			likeds(wsdl_map_t)				RADDAT			571126		55037300			
5846+ *									RADDAT			571205		55037400			
5847+ *	Structure to hold a WSDL (message) part definition								RADDAT			571205		55037500			
5848+D	wsdl_part_t...								RADDAT			571205		55037600			
5849+D		DS		qualified		based(pDummy)			RADDAT			571205		55037700			
5850+D	qName			likeds(wsdl_qName_t)				RADDAT			571205		55037800			
5851+D	qElement			likeds(wsdl_qName_t)				RADDAT			571205		55037900			
5852+ *									RADDAT			571205		55038000			
5853+ *	Structure to hold all port types of a given WSDL file.								RADDAT			571205		55038100			
5854+D	wsdl_portTypes_t...								RADDAT			571205		55038200			
5855+D		DS		qualified		based(pDummy)			RADDAT			571205		55038300			
5856+D	portTypeList			likeds(wsdl_map_t)				RADDAT			571205		55038400			
5857+ *									RADDAT			571205		55038500			
5858+ *	Structure to hold a WSDL port type definition								RADDAT			571205		55038600			
5859+D	wsdl_portType_t...								RADDAT			571205		55038700			
5860+D		DS		qualified		based(pDummy)			RADDAT			571205		55038800			
5861+D	qName			likeds(wsdl_qName_t)				RADDAT			571205		55038900			
5862+D	operationList			likeds(wsdl_map_t)				RADDAT			571205		55039000			
5863+ *									RADDAT			571208		55039100			
5864+ *	Structure to hold a WSDL operation								RADDAT			571208		55039200			
5865+D	wsdl_operation_t...								RADDAT			571208		55039300			
5866+D		DS		qualified		based(pDummy)			RADDAT			571208		55039400			

Line	----- Source Specifications -----><----- Comments ----->										Do	Page	Change	Src	Seq	
Number	1	2	3	4	5	6	7	8	9	10	Num	Line	Date	Id	Number	
5867+D	isUsed		N							RADDAT			571208		55039500	
5868+D	reserved_1		15A							RADDAT			571208		55039600	
5869+D	qName			likeds(wsdl_qName_t)					RADDAT			571208		55039700	
5870+D	parameterOrder...									RADDAT			571208		55039800	
5871+D				like(wsdl_parameterOrder_t)						RADDAT			571208		55039900	
5872+D	documentation			like(wsdl_documentation_t)						RADDAT			571208		55040000	
5873+D	input			likeds(wsdl_operationIO_t)						RADDAT			571208		55040100	
5874+D	output			likeds(wsdl_operationIO_t)						RADDAT			571208		55040200	
5875+ *										RADDAT			571208		55040300	
5876+ *	Structure to hold the operation messages										RADDAT			571208		55040400
5877+D	wsdl_operationIO_t...									RADDAT			571208		55040500	
5878+D		DS		qualified			based(pDummy)			RADDAT			571208		55040600	
5879+D	qName			likeds(wsdl_qName_t)					RADDAT			571208		55040700	
5880+D	qMessage			likeds(wsdl_qName_t)					RADDAT			571208		55040800	
5881+ *										RADDAT			571123		55040900	
5882+ *	Structure of a Map										RADDAT			571205		55041000
5883+D	wsdl_map_t...									RADDAT			571126		55041100	
5884+D		DS		qualified			based(pDummy)			RADDAT			571123		55041200	
5885+D	hMap			like(BasicMap_handle_t)						RADDAT			571126		55041300	
5886+D	curKey			like(wsdl_mapKey_t)					RADDAT			571126		55041400	
5887+D	isChoice		N							RADDAT			581101		55041500	
5888+ *										RADDAT			571204		55041600	
5889+ *	Structure to hold a WSDL port definition										RADDAT			571204		55041700
5890+D	wsdl_port_t...									RADDAT			571204		55041800	
5891+D		DS		qualified			based(pDummy)			RADDAT			571204		55041900	
5892+D	qName			likeds(wsdl_qName_t)					RADDAT			571204		55042000	
5893+D	protocol			like(wsdl_protocol_t)					RADDAT			571207		55042100	
5894+D	qBinding			likeds(wsdl_qName_t)					RADDAT			571204		55042200	
5895+D	address			like(wsdl_url_t)					RADDAT			571208		55042300	
5896+ *										RADDAT			571205		55042400	
5897+ *	Structure to hold all bindings of a given WSDL file										RADDAT			571205		55042500
5898+D	wsdl_bindings_t...									RADDAT			571205		55042600	
5899+D		DS		qualified			based(pDummy)			RADDAT			571205		55042700	
5900+D	bindingsList			likeds(wsdl_map_t)					RADDAT			571205		55042800	
5901+ *										RADDAT			571204		55042900	
5902+ *	Structure to hold a WSDL binding definition										RADDAT			571204		55043000
5903+D	wsdl_binding_t...									RADDAT			571204		55043100	
5904+D		DS		qualified			based(pDummy)			RADDAT			571204		55043200	
5905+D	qName			likeds(wsdl_qName_t)					RADDAT			571204		55043300	
5906+D	qType			likeds(wsdl_qName_t)					RADDAT			571204		55043400	
5907+D	style			like(wsdl_style_t)					RADDAT			571229		55043500	
5908+D	boundOperationList...									RADDAT			571204		55043600	
5909+D				likeds(wsdl_map_t)					RADDAT			571204		55043700	
5910+ *										RADDAT			571204		55043800	
5911+ *	Structure to hold a WSDL operation bound to a binding										RADDAT			571205		55043900
5912+D	wsdl_boundOperation_t...									RADDAT			571204		55044000	
5913+D		DS		qualified			based(pDummy)			RADDAT			571204		55044100	
5914+D	qName			likeds(wsdl_qName_t)					RADDAT			571204		55044200	
5915+D	soapAction			like(wsdl_soapAction_t)						RADDAT			571208		55044300	
5916+D	input			likeds(wsdl_boundOperationIO_t)					RADDAT			571229		55044400	
5917+D	output			likeds(wsdl_boundOperationIO_t)					RADDAT			571229		55044500	
5918+ *										RADDAT			571229		55044600	
5919+ *	Structure to hold the attributes of a bound operation message.										RADDAT			571229		55044700
5920+D	wsdl_boundOperationIO_t...									RADDAT			571229		55044800	
5921+D		DS		qualified			based(pDummy)			RADDAT			571229		55044900	
5922+D	qName			likeds(wsdl_qName_t)					RADDAT			571229		55045000	

Line	----- Source Specifications -----><----- Comments ----->										Do	Page	Change	Src	Seq
Number	1	2	3	4	5	6	7	8	9	10	Num	Line	Date	Id	Number
5923+D	use												571229		55045100
5924+ *													571129		55045200
5925+ *	Structure to hold a RPG D-Spec description												571129		55045300
5926+D	wsdl_dSpec_t...												571129		55045400
5927+D		DS			qualified					based(pDummy)			571129		55045500
5928+D	name		4096A		varying								571129		55045600
5929+D	declType		20A										571130		55045700
5930+D	primitive				likeds(wsdl_rpgType_t)								571129		55045800
5931+D	numKwds		10I 0										571129		55045900
5932+D	keywords		4096A		varying dim(cMAX_KWDS)								571204		55046000
5933+ *													571204		55046100
5934+D	cMAX_KWDS	C		10									571204		55046200
5935+ *													571120		55046300
5936+ *	Structure to hold a RPG type definition												571120		55046400
5937+D	wsdl_rpgType_t	DS			qualified					based(pDummy)			571120		55046500
5938+D	length		7S 0										571120		55046600
5939+D	dataType		2A										571128		55046700
5940+D	decPos		2S 0										571120		55046800
5941+D	keyword		32A		varying								571129		55046900
5942+D	refField		4096A		varying								571128		55047000
5943+ *													571108		55047100
5944+ *	Structure to hold a qualified name												571108		55047200
5945+D	wsdl_qName_t...												571108		55047300
5946+D		DS			qualified					based(pDummy)			571108		55047400
5947+D	lName				like(wsdl_lName_t)								580528		55047500
5948+D	nsPrefix				like(wsdl_nsPrefix_t)								580528		55047600
5949+D	nsValue				like(wsdl_nsValue_t)								580531		55047700
5950+ *													571229		55047800
5951+ *	Structure to hold an attribute												571229		55047900
5952+D	wsdl_attribute_t...												571229		55048000
5953+D		DS			qualified					based(pDummy)			571229		55048100
5954+D	name				like(wsdl_name_t)								571229		55048200
5955+D	value				like(wsdl_value_t)								571229		55048300
5956+ *													571110		55048400
5957+ *	Structure to hold a 'from' / 'to' range value												571110		55048500
5958+ *	See also: New_Range(from: to)												571110		55048600
5959+D	wsdl_range_t	DS			qualified					based(pDummy)			571110		55048700
5960+D	from		10I 0										571110		55048800
5961+D	to		10I 0										571110		55048900
5962+D	reserved_1		8A										571126		55049000
5963+ *													571120		55049100
5964+ *	Structure to hold a generator tag												571120		55049200
5965+D	wsdl_genTag_t	DS			qualified					based(pDummy)			571120		55049300
5966+D	command		30A		varying								571120		55049400
5967+D	variable		254A		varying								590423		55049500
5968+D	length		10I 0										571120		55049600
5969+D	startColumn		10I 0										571120		55049700
5970+D	maxColumn		10I 0										571120		55049800
5971+D	reserved_1		4A										590423		55049900
5972+D	contChars		14A		varying								571201		55050000
5973+D	extension		254A		varying								590423		55050100
5974+ *													571108		55050200
5975+ *	Error code												571108		55050300
5976+D	cWSDL_ERROR	C		-1									571108		55050400
5977+D	cWSDL_OK	C		0									571108		55050500
5978+ *													571108		55050600

Line	----- Source Specifications -----><----- Comments ----->										Do	Page	Change	Src	Seq
Number	1	2	3	4	5	6	7	8	9	10	Num	Line	Date	Id	Number
5979+	*	WSDL file constants								RADDAT			571108		55050700
5980+	*	... xPath								RADDAT			571108		55050800
5981+D	cWSDL_XPATH_DEFINITIONS...									RADDAT			571108		55050900
5982+D		C			'/definitions'					RADDAT			571108		55051000
5983+	*									RADDAT			571108		55051100
5984+	*	... attributes								RADDAT			571108		55051200
5985+D	cWSDL_ATTR_XMLNS...									RADDAT			571108		55051300
5986+D		C			'xmlns:'					RADDAT			571108		55051400
5987+	*									RADDAT			571110		55051500
5988+	*	Namespaces								RADDAT			571110		55051600
5989+D	cNS_WSDL...									RADDAT			571110		55051700
5990+D		C			'http://schemas.xmlsoap.org/wsdl/'					RADDAT			571110		55051800
5991+D	cNS_WSDL_SOAP...									RADDAT			571110		55051900
5992+D		C			'http://schemas.xmlsoap.org/wsdl/+					RADDAT			571207		55052000
5993+D					soap/'					RADDAT			571207		55052100
5994+D	cNS_WSDL_SOAP_ENVELOPE...									RADDAT			571229		55052200
5995+D		C			'http://schemas.xmlsoap.org/+					RADDAT			571229		55052300
5996+D					soap/envelope/'					RADDAT			571229		55052400
5997+D	cNS_WSDL_SOAP_ENCODING...									RADDAT			571229		55052500
5998+D		C			'http://schemas.xmlsoap.org/+					RADDAT			571229		55052600
5999+D					soap/encoding/'					RADDAT			571229		55052700
6000+D	cNS_WSDL_HTTP...									RADDAT			571207		55052800
6001+D		C			'http://schemas.xmlsoap.org/wsdl/+					RADDAT			571207		55052900
6002+D					http/'					RADDAT			571207		55053000
6003+D	cNS_APACHE_SOAP...									RADDAT			571110		55053100
6004+D		C			'http://xml.apache.org/xml-soap'					RADDAT			571110		55053200
6005+D	cNS_SCHEMA...									RADDAT			571110		55053300
6006+D		C			'http://www.w3.org/2001/XMLSchema'					RADDAT			571110		55053400
6007+D	cNS_SCHEMA_INSTANCE...									RADDAT			571231		55053500
6008+D		C			'http://www.w3.org/2001/+					RADDAT			571231		55053600
6009+D					XMLSchema-instance'					RADDAT			571231		55053700
6010+	*									RADDAT			580531		55053800
6011+	*	Namespaces of WSDL2RPG								RADDAT			580531		55053900
6012+D	cPREFIX_WSDL2RPG...									RADDAT			580604		55054000
6013+D		C			'wns'					RADDAT			580604		55054100
6014+D	cNS_WSDL2RPG...									RADDAT			580531		55054200
6015+D		C			'http://www.tools400.de/wsdl2rpg/'					RADDAT			580531		55054300
6016+	*									RADDAT			571229		55054400
6017+	*	WSDL styles								RADDAT			571229		55054500
6018+D	cWSDL_STYLE_RPC...									RADDAT			571229		55054600
6019+D		C			'rpc'					RADDAT			571229		55054700
6020+D	cWSDL_STYLE_DOCUMENT...									RADDAT			571229		55054800
6021+D		C			'document'					RADDAT			571229		55054900
6022+	*									RADDAT			571231		55055000
6023+	*	WSDL usages								RADDAT			571231		55055100
6024+D	cWSDL_USE_LITERAL...									RADDAT			571231		55055200
6025+D		C			'literal'					RADDAT			571231		55055300
6026+D	cWSDL_USE_ENCODED...									RADDAT			571231		55055400
6027+D		C			'encoded'					RADDAT			571231		55055500
6028+	*									RADDAT			571113		55055600
6029+	*	RPG generator								RADDAT			571113		55055700
6030+D	cREFERENCE_FIELD_SUFFIX...									RADDAT			571113		55055800
6031+D		C			'_t'					RADDAT			571113		55055900
6032+D	cDEFAULT_TAB_INDENT...									RADDAT			571118		55056000
6033+D		C			3					RADDAT			571118		55056100
6034+D	cDEFAULT LEFT MARGIN...									RADDAT			571118		55056200

Line Number	<----- Source Specifications ----->	<----- Comments ----->	Do Num	Page Line	Change Date	Src Id	Seq Number
6035+D	C	RADDAT			571118		55056300
6036+D	cDEFAULT_RIGHT_MARGIN...	RADDAT			571118		55056400
6037+D	C	RADDAT			571118		55056500
6038+ *		RADDAT			571118		55056600
6039+D	cCONTINUE_LITERAL...	RADDAT			571118		55056700
6040+D	C	RADDAT			571118		55056800
6041+D	cCONTINUE_NAME...	RADDAT			571118		55056900
6042+D	C	RADDAT			571118		55057000
6043+D	cCONTINUE_NEXT_LINE...	RADDAT			571129		55057100
6044+D	C	RADDAT			571129		55057200
6045+D	cEOL...	RADDAT			571118		55057300
6046+D	C	RADDAT			571118		55057400
6047+ *		RADDAT			571118		55057500
6048+D	cSPEC_IF...	RADDAT			580202		55057600
6049+D	C	RADDAT			580202		55057700
6050+D	cSPEC_ENDIF...	RADDAT			580202		55057800
6051+D	C	RADDAT			580202		55057900
6052+D	cSPEC_WHEN...	RADDAT			571118		55058000
6053+D	C	RADDAT			571118		55058100
6054+D	cSPEC_OTHER...	RADDAT			590606		55058200
6055+D	C	RADDAT			590606		55058300
6056+D	cSPEC_SELECT...	RADDAT			571118		55058400
6057+D	C	RADDAT			571118		55058500
6058+D	cSPEC_ENDSL...	RADDAT			571118		55058600
6059+D	C	RADDAT			571118		55058700
6060+D	cSPEC_FOR...	RADDAT			571214		55058800
6061+D	C	RADDAT			571214		55058900
6062+D	cSPEC_ENDFOR...	RADDAT			571214		55059000
6063+D	C	RADDAT			571214		55059100
6064+D	cSPEC_RETURN...	RADDAT			580202		55059200
6065+D	C	RADDAT			580202		55059300
6066+ *		RADDAT			571113		55059400
6067+D	cBOOL_AND...	RADDAT			571118		55059500
6068+D	C	RADDAT			571118		55059600
6069+ *		RADDAT			571120		55059700
6070+D	cGENERATOR_TAG_START...	RADDAT			571120		55059800
6071+D	C	RADDAT			571120		55059900
6072+D	cGENERATOR_TAG_OPEN_BRACKET...	RADDAT			590108		55060000
6073+D	C	RADDAT			590108		55060100
6074+D	cGENERATOR_TAG_CLOSE_BRACKET...	RADDAT			590108		55060200
6075+D	C	RADDAT			571120		55060300
6076+D	cGENERATOR_TAG_FIRST_DELIMITER...	RADDAT			571120		55060400
6077+D	C	RADDAT			571120		55060500
6078+D	cGENERATOR_TAG_NEXT_DELIMITER...	RADDAT			571120		55060600
6079+D	C	RADDAT			571120		55060700
6080+ *		RADDAT			571120		55060800
6081+D	cGENERATOR_CALLBACK_START...	RADDAT			571120		55060900
6082+D	C	RADDAT			571120		55061000
6083+D	cGENERATOR_CALLBACK_END...	RADDAT			571120		55061100
6084+D	C	RADDAT			571120		55061200
6085+ *		RADDAT			571120		55061300
6086+D	cGENERATOR_CMD_GENERATE...	RADDAT			571120		55061400
6087+D	C	RADDAT			571120		55061500
6088+D	cGENERATOR_CMD_GENERATE_OPERATION_PROCEDURE_PROTOTYPE...	RADDAT			571201		55061600
6089+D	C	RADDAT			571201		55061700
6090+D	cGENERATOR_CMD_GENERATE_OPERATION_REQUEST_PARAMETERS...	RADDAT			571201		55061800

Line	Source Specifications	Comments	Do	Page	Change	Src	Seq
Number	1.....2.....3.....4.....5.....6.....7.....8.....9.....10		Num	Line	Date	Id	Number
6091+D	C	'operationRequestParameters'			571201		55061900
6092+D	cGENERATOR_CMD_GENERATE_TYPES...				571201		55062000
6093+D	C	'types'			571201		55062100
6094+D	cGENERATOR_CMD_GENERATE_PROTOTYPES...				571201		55062200
6095+D	C	'operationPrototypes'			571201		55062300
6096+D	cGENERATOR_CMD_GENERATE_WEB_SERVICE_ENDPOINT...				571201		55062400
6097+D	C	'webServiceEndPoint'			571201		55062500
6098+D	cGENERATOR_CMD_GENERATE_OPERATION_PROCEDURE_INTERFACE...				571201		55062600
6099+D	C	'operationProcedureInterface'			571201		55062700
6100+D	cGENERATOR_CMD_GENERATE_OPERATION_RETURN_VALUE...				571201		55062800
6101+D	C	'operationReturnValue'			571201		55062900
6102+D	cGENERATOR_CMD_GENERATE_OPERATION_USERDATA...				571201		55063000
6103+D	C	'operationUserData'			571201		55063100
6104+D	cGENERATOR_CMD_GENERATE_END_OF_PROCEDURE...				571201		55063200
6105+D	C	'endOfProcedure'			571201		55063300
6106+D	cGENERATOR_CMD_GENERATE_CALLBACKS...				571120		55063400
6107+D	C	'callbackProcedures'			571120		55063500
6108+D	cGENERATOR_CMD_GENERATE_NAMESPACES...				571202		55063600
6109+D	C	'namespaces'			571202		55063700
6110+D	cGENERATOR_CMD_GENERATE_REQUEST...				571202		55063800
6111+D	C	'request'			571202		55063900
6112+D	cGENERATOR_CMD_GENERATE_DIMENSIONS...				571205		55064000
6113+D	C	'dimensions'			571205		55064100
6114+D	cGENERATOR_CMD_GENERATE_ARRAY_ITERATOR...				571214		55064200
6115+D	C	'arrayIterator'			571214		55064300
6116+ *					571120		55064400
6117+D	cGENERATOR_CMD_VARIABLE...				571120		55064500
6118+D	C	'var'			571120		55064600
6119+D	cGENERATOR_CMD_VARIABLE_OUTPUT_MEMBER...				571201		55064700
6120+D	C	'outmbr'			571201		55064800
6121+D	cGENERATOR_CMD_VARIABLE_PORT_NAME...				571227		55064900
6122+D	C	'portName'			571227		55065000
6123+D	cGENERATOR_CMD_VARIABLE_OPERATION_NAME...				571201		55065100
6124+D	C	'operationName'			571201		55065200
6125+D	cGENERATOR_CMD_VARIABLE_MEMBER_TEXT...				590326		55065300
6126+D	C	'memberText'			590326		55065400
6127+D	cGENERATOR_CMD_VARIABLE_OPERATION_RESPONSE...				571201		55065500
6128+D	C	'operationResponse'			571201		55065600
6129+D	cGENERATOR_CMD_VARIABLE_START_CALLBACK...				571201		55065700
6130+D	C	'startCallback'			571201		55065800
6131+D	cGENERATOR_CMD_VARIABLE_END_CALLBACK...				571201		55065900
6132+D	C	'endCallback'			571201		55066000
6133+D	cGENERATOR_CMD_VARIABLE_SOAP_ACTION...				571208		55066100
6134+D	C	'soapAction'			571208		55066200
6135+D	cGENERATOR_CMD_VARIABLE_ADDR_USERDATA...				581030		55066300
6136+D	C	'%addr_userData_'			590729		55066400
6137+D	cGENERATOR_CMD_VARIABLE_CLEAR_RTNVALUE...				581030		55066500
6138+D	C	'clear_rtnValue'			581030		55066600
6139+D	cGENERATOR_CMD_VARIABLE_RTNVALUE_USERDATA...				581030		55066700
6140+D	C	'rtnValue=_userData'			581030		55066800
6141+D	cGENERATOR_CMD_VARIABLE_RTNVALUE...				581030		55066900
6142+D	C	'rtnValue'			581030		55067000
6143+ *					571221		55067100
6144+D	cGENERATOR_RPG_ARRAY_PREFIX...				571221		55067200
6145+D	C	'RpgArrayOf'			571221		55067300
6146+D	cGENERATOR_RENAME_TYPE_NAME_SUFFIX...				580121		55067400

Line	Source Specifications	Comments	Do	Page	Change	Src	Seq
Number	1.....2.....3.....4.....5.....6.....7.....8.....9.....10		Num	Line	Date	Id	Number
6147+D	C	'Rnmd'	RADDAT		580118		55067500
6148+D	cGENERATOR_MOVED_TYPE_NAME_SUFFIX...		RADDAT		580121		55067600
6149+D	C	'Mvd'	RADDAT		580121		55067700
6150+D	cGENERATOR_REFERENCE_NAME_SUFFIX...		RADDAT		581028		55067800
6151+D	C	'Ref'	RADDAT		581028		55067900
6152+D	cGENERATOR_RPC_RESPONSE_NAME_SUFFIX...		RADDAT		590109		55068000
6153+D	C	'Rpc'	RADDAT		590109		55068100
6154+D	cGENERATOR_RPG_ARRAY_VALUE_NAME_1...		RADDAT		580125		55068200
6155+D	C	'item'	RADDAT		580125		55068300
6156+D	cGENERATOR_RPG_ARRAY_VALUE_NAME_2...		RADDAT		580125		55068400
6157+D	C	'value'	RADDAT		580125		55068500
6158+ *			RADDAT		571201		55068600
6159+ *	General purpose constants		RADDAT		571113		55068700
6160+D	cSINGLE_QUOTE...		RADDAT		571113		55068800
6161+D	C	' '' '	RADDAT		571113		55068900
6162+D	cDOUBLE_QUOTE...		RADDAT		571202		55069000
6163+D	C	' '' '	RADDAT		571202		55069100
6164+ *			RADDAT		580905		55069200
6165+D	cEBCDIC_CRLF C	x'0D25'	RADDAT		580905		55069300
6166+ *			RADDAT		571110		55069400
6167+ /ENDIF			RADDAT		571106		55069500
6168+ /COPY QBASICS1,PBASICS1			RADDAT	CPY	580131		54000500
-----*							
* RPG member name : PBASICS1							
* External name : WSDL2RPG/QBASICS1(PBASICS1)							
* Last change : 08/19/09 12:46:31							
* Text 'description' : SrvPgm: Basic Routines - Include cpybk							
-----*							
6169+ /IF NOT DEFINED(D_BASIC)			RADDAT		000000		57000100
LINES EXCLUDED: 37							
6170+ /ENDIF			RADDAT		000000		57003900
6171+ *			RADDAT		580131		54000600
6172+D	cATTR_ID C	'id'	RADDAT		580131		54000700
6173+ *			RADDAT		580131		54000800
6174+D	wsdl_multiRef_t...		RADDAT		580131		54000900
6175+D	DS	qualified based(pDummy)	RADDAT		580131		54001000
6176+D	userData	*	RADDAT		580131		54001100
6177+D	cbStart	* procptr	RADDAT		580131		54001200
6178+D	cbEnd	* procptr	RADDAT		580131		54001300
6179+D	isActive	N	RADDAT		580201		54001400
6180+D	depth	10I 0	RADDAT		580201		54001500
6181+D	name	1024A varying	RADDAT		580201		54001600
6182+D	namespace	1024A varying	RADDAT		580201		54001700
6183+D	pathLen	10I 0	RADDAT		580201		54001800
6184+ *			RADDAT		580131		54001900
6185+ *	Creates a <multiRef> map.		RADDAT		580131		54002000
6186+D	MultiRef_new...		RADDAT		580131		54002100
6187+D	PR		RADDAT		580201		54002200
6188+D		extproc('WSDL2R94_+	RADDAT		580131		54002300
6189+D		MultiRef_new+	RADDAT		580131		54002400
6190+D		')	RADDAT		580131		54002500
6191+ *			RADDAT		580201		54002600
6192+ *	Returns the procedure pointer of the <multiRef> start callback procedure.		RADDAT		580201		54002700
6193+D	MultiRef_getStartCallback...		RADDAT		580201		54002800
6194+D	PR	* procptr	RADDAT		580201		54002900
6195+D		extproc('WSDL2R94_+	RADDAT		580201		54003000

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
6196+D	MultiRef_getStartCallback+	RADDAT			580201		54003100
6197+D)	RADDAT			580201		54003200
6198+ *		RADDAT			580201		54003300
6199+ *	Returns the procedure pointer of the <multiRef> end callback procedure.	RADDAT			580201		54003400
6200+D	MultiRef_getEndCallback...	RADDAT			580201		54003500
6201+D	PR * procptr	RADDAT			580201		54003600
6202+D	extproc('WSDL2R94_+	RADDAT			580201		54003700
6203+D	MultiRef_getEndCallback+	RADDAT			580201		54003800
6204+D)	RADDAT			580201		54003900
6205+ *		RADDAT			580131		54004000
6206+ *	Returns cTrue if the given element references a <multiRef> element.	RADDAT			580202		54004100
6207+D	MultiRef_isReference...	RADDAT			580202		54004200
6208+D	PR N	RADDAT			580131		54004300
6209+D	extproc('WSDL2R94_+	RADDAT			580131		54004400
6210+D	MultiRef_isReference+	RADDAT			580202		54004500
6211+D)	RADDAT			580131		54004600
6212+D	attrs * dim(32767)	RADDAT			580131		54004700
6213+D	const options(*varsize)	RADDAT			580131		54004800
6214+ *		RADDAT			580202		54004900
6215+ *	Returns cTrue if the given element is a <multiRef> data element.	RADDAT			580202		54005000
6216+D	MultiRef_isData...	RADDAT			580202		54005100
6217+D	PR N	RADDAT			580202		54005200
6218+D	extproc('WSDL2R94_+	RADDAT			580202		54005300
6219+D	MultiRef_isData+	RADDAT			580202		54005400
6220+D)	RADDAT			580202		54005500
6221+D	attrs * dim(32767)	RADDAT			580202		54005600
6222+D	const options(*varsize)	RADDAT			580202		54005700
6223+ *		RADDAT			580131		54005800
6224+ *	Adds an entry to the <multiRef> map.	RADDAT			580131		54005900
6225+D	MultiRef_put...	RADDAT			580131		54006000
6226+D	PR	RADDAT			580131		54006100
6227+D	extproc('WSDL2R94_+	RADDAT			580131		54006200
6228+D	MultiRef_put+	RADDAT			580131		54006300
6229+D)	RADDAT			580131		54006400
6230+D	i_attrs * const dim(32767) options(*varsize)	RADDAT			580201		54006500
6231+D	i_userdata * const	RADDAT			580201		54006600
6232+D	i_depth 10I 0 const	RADDAT			580201		54006700
6233+D	i_namespace 1024A const varying	RADDAT			580201		54006800
6234+D	i_name 1024A const varying	RADDAT			580201		54006900
6235+D	i_path 24576A const varying	RADDAT			580201		54007000
6236+D	i_cbStart * const procptr	RADDAT			580201		54007100
6237+D	i_cbEnd * const procptr	RADDAT			580201		54007200
6238+ *		RADDAT			580131		54007300
6239+ *	Returns the <multiRef> information that is associated to a	RADDAT			580131		54007400
6240+ *	given key.	RADDAT			580131		54007500
6241+D	MultiRef_get...	RADDAT			580131		54007600
6242+D	PR likeds(wsd1_multiRef_t)	RADDAT			580131		54007700
6243+D	extproc('WSDL2R94_+	RADDAT			580131		54007800
6244+D	MultiRef_get+	RADDAT			580131		54007900
6245+D)	RADDAT			580131		54008000
6246+D	i_attrs * const dim(32767) options(*varsize)	RADDAT			580131		54008100
6247+D	i_depth 10I 0 const	RADDAT			580201		54008200
6248+D	i_name 1024A const varying	RADDAT			580201		54008300
6249+ *		RADDAT			580201		54008400
6250+ *	Releases an active <multiRef> environment.	RADDAT			580201		54008500
6251+D	MultiRef_release...	RADDAT			580201		54008600

Line	----- Source Specifications -----><----- Comments ----->										Do	Page	Change	Src	Seq
Number	1	2	3	4	5	6	7	8	9	10	Num	Line	Date	Id	Number
6252+D		PR								RADDAT			580201		54008700
6253+D					extproc('WSDL2R94_+					RADDAT			580201		54008800
6254+D					MultiRef_release+					RADDAT			580201		54008900
6255+D					')					RADDAT			580201		54009000
6256+D	io_multiRef				likeds(wsd1_multiRef_t)					RADDAT			580201		54009100
6257+ *										RADDAT			580131		54009200
6258+ *	Deletes a given <multiRef> map.												580131		54009300
6259+D	MultiRef_delete...									RADDAT			580131		54009400
6260+D		PR								RADDAT			580131		54009500
6261+D					extproc('WSDL2R94_+					RADDAT			580131		54009600
6262+D					MultiRef_delete+					RADDAT			580131		54009700
6263+D					')					RADDAT			580131		54009800
6264+ *										RADDAT			580131		54009900
6265+ *	Returns a NULL <multiRef> map.												580131		54010000
6266+D	MultiRef_null...									RADDAT			580131		54010100
6267+D		PR			like(BasicMap_handle_t)					RADDAT			580131		54010200
6268+D					extproc('WSDL2R94_+					RADDAT			580131		54010300
6269+D					MultiRef_null+					RADDAT			580131		54010400
6270+D					')					RADDAT			580131		54010500
6271+ *										RADDAT			580131		54010600
6272+ *	Returns cTrue if a given <multiRef> map is NULL.												580131		54010700
6273+D	MultiRef_isNull...									RADDAT			580131		54010800
6274+D		PR		N						RADDAT			580131		54010900
6275+D					extproc('WSDL2R94_+					RADDAT			580201		54011000
6276+D					MultiRef_isNull+					RADDAT			580201		54011100
6277+D					')					RADDAT			580201		54011200
6278+ *										RADDAT			580201		54011300
6279+ *	Handles <multiRef> elements.												580201		54011400
6280+D	MultiRef_startCallback...									RADDAT			580201		54011500
6281+D		PR								RADDAT			580201		54011600
6282+D					extproc('WSDL2R94_+					RADDAT			580201		54011700
6283+D					MultiRef_startCallback+					RADDAT			580201		54011800
6284+D					')					RADDAT			580201		54011900
6285+D	userdata			*	value					RADDAT			580201		54012000
6286+D	depth		10I	0	value					RADDAT			580201		54012100
6287+D	namespace		1024A		varying const					RADDAT			580201		54012200
6288+D	name		1024A		varying const					RADDAT			580201		54012300
6289+D	path		24576A		varying const					RADDAT			580201		54012400
6290+D	attrs			*	dim(32767)					RADDAT			580201		54012500
6291+D					const options(*varsize)					RADDAT			580201		54012600
6292+ *										RADDAT			580201		54012700
6293+ *	Returns the Soap Fault Details Information.												580201		54012800
6294+D	MultiRef_endCallback...									RADDAT			580201		54012900
6295+D		PR								RADDAT			580201		54013000
6296+D					extproc('WSDL2R94_+					RADDAT			580201		54013100
6297+D					MultiRef_endCallback+					RADDAT			580201		54013200
6298+D					')					RADDAT			580201		54013300
6299+D	userdata			*	value					RADDAT			580201		54013400
6300+D	depth		10I	0	value					RADDAT			580201		54013500
6301+D	namespace		1024A		varying const					RADDAT			580201		54013600
6302+D	name		1024A		varying const					RADDAT			580201		54013700
6303+D	path		24576A		varying const					RADDAT			580201		54013800
6304+D	value		65535A		varying const					RADDAT			580201		54013900
6305+D	attrs			*	dim(32767)					RADDAT			580201		54014000
6306+D					const options(*varsize)					RADDAT			580201		54014100
6307+ *										RADDAT			571107		54014200

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
6308+	/ENDIF	RADDAT			571108	54014300	
6309	/COPY QWSDL2RPG,PWSDL2R98	WSDL2RPG: UnMarshaller/Marshaller		CPY	000000	000331	
	* RPG member name : PWSDL2R98					58	
	* External name : WSDL2RPG/QWSDL2RPG(PWSDL2R98)					58	
	* Last change : 07/29/09 10:43:01					58	
	* Text 'description' : WSDL2RPG: UnMarshaller/Marshaller (Prototype)					58	
6310+	/IF NOT DEFINED(PWSDL2R98)	RADDAT			571120	58000100	
6311+	/DEFINE PWSDL2R98	RADDAT			571120	58000200	
6312+	*	RADDAT			571108	58000300	
6313+	/COPY QWSDL2RPG,TYPES	RADDAT		CPY	571108	58000400	
	* RPG member name : TYPES					59	
	* External name : WSDL2RPG/QWSDL2RPG(TYPES)					59	
	* Last change : 07/29/09 10:43:01					59	
	* Text 'description' : Type definitions					59	
6314+	/IF NOT DEFINED(TYPES)	RADDAT			571106	59000100	
	LINES EXCLUDED: 693						
6315+	/ENDIF	RADDAT			571106	59069500	
6316+	*	RADDAT			580206	58000500	
6317+	* Returns a given XML as 'anyType' (actually string).	RADDAT			580206	58000600	
6318+D	UnMarshaller_toAnyType...	RADDAT			580206	58000700	
6319+D	PR 65535A varying	RADDAT			580206	58000800	
6320+D	extproc('WSDL2R98_+	RADDAT			580206	58000900	
6321+D	UnMarshaller_toAnyType+	RADDAT			580206	58001000	
6322+D	')	RADDAT			580206	58001100	
6323+D	i_value 65535A const varying options(*varsize)	RADDAT			581103	58001200	
6324+	*	RADDAT			580207	58001300	
6325+	* Returns a given XML as 'any' (actually string).	RADDAT			580207	58001400	
6326+D	UnMarshaller_toAny...	RADDAT			580207	58001500	
6327+D	PR 65535A varying	RADDAT			580207	58001600	
6328+D	extproc('WSDL2R98_+	RADDAT			580207	58001700	
6329+D	UnMarshaller_toAny+	RADDAT			580207	58001800	
6330+D	')	RADDAT			580207	58001900	
6331+D	i_value 65535A const varying options(*varsize)	RADDAT			581103	58002000	
6332+	*	RADDAT			590602	58002100	
6333+	* Returns a given XML as 'schema' (actually string).	RADDAT			590602	58002200	
6334+D	UnMarshaller_toSchema...	RADDAT			590602	58002300	
6335+D	PR 65535A varying	RADDAT			590602	58002400	
6336+D	extproc('WSDL2R98_+	RADDAT			590602	58002500	
6337+D	UnMarshaller_toSchema+	RADDAT			590602	58002600	
6338+D	')	RADDAT			590602	58002700	
6339+D	i_value 65535A const varying options(*varsize)	RADDAT			590602	58002800	
6340+	*	RADDAT			571125	58002900	
6341+	* Returns a given XML as 'string'.	RADDAT			571125	58003000	
6342+D	UnMarshaller_toString...	RADDAT			571125	58003100	
6343+D	PR 65535A varying	RADDAT			571125	58003200	
6344+D	extproc('WSDL2R98_+	RADDAT			571125	58003300	
6345+D	UnMarshaller_toString+	RADDAT			571125	58003400	
6346+D	')	RADDAT			571125	58003500	
6347+D	i_value 65535A const varying options(*varsize)	RADDAT			581103	58003600	
6348+	*	RADDAT			571125	58003700	
6349+	* Returns a given XML as 'boolean'.	RADDAT			571125	58003800	
6350+D	UnMarshaller_toBoolean...	RADDAT			571125	58003900	

Line	----- Source Specifications -----><----- Comments ----->										Do	Page	Change	Src	Seq
Number	1	2	3	4	5	6	7	8	9	10	Num	Line	Date	Id	Number
6351+D		PR		N						RADDAT			571125		58004000
6352+D					extproc('WSDL2R98_+					RADDAT			571125		58004100
6353+D					UnMarshaller_toBoolean+					RADDAT			571125		58004200
6354+D					')					RADDAT			571125		58004300
6355+D	i_value			65535A	const	varying	options(*varsize)			RADDAT			581103		58004400
6356+ *										RADDAT			571125		58004500
6357+ *	Returns a given XML as 'decimal'.												571125		58004600
6358+D	UnMarshaller_toDecimal...									RADDAT			571125		58004700
6359+D		PR				like(wsd1_decimal_t)				RADDAT			571125		58004800
6360+D					extproc('WSDL2R98_+					RADDAT			571125		58004900
6361+D					UnMarshaller_toDecimal+					RADDAT			571125		58005000
6362+D					')					RADDAT			571125		58005100
6363+D	i_value			65535A	const	varying	options(*varsize)			RADDAT			581103		58005200
6364+ *										RADDAT			571221		58005300
6365+ *	Returns a given XML as 'double'.												571221		58005400
6366+D	UnMarshaller_toDouble...									RADDAT			571221		58005500
6367+D		PR				like(wsd1_double_t)				RADDAT			571221		58005600
6368+D					extproc('WSDL2R98_+					RADDAT			571221		58005700
6369+D					UnMarshaller_toDouble+					RADDAT			571221		58005800
6370+D					')					RADDAT			571221		58005900
6371+D	i_value			65535A	const	varying	options(*varsize)			RADDAT			581103		58006000
6372+ *										RADDAT			580319		58006100
6373+ *	Returns a given XML as 'float'.												580319		58006200
6374+D	UnMarshaller_toFloat...									RADDAT			580319		58006300
6375+D		PR				like(wsd1_float_t)				RADDAT			580319		58006400
6376+D					extproc('WSDL2R98_+					RADDAT			580319		58006500
6377+D					UnMarshaller_toFloat+					RADDAT			580319		58006600
6378+D					')					RADDAT			580319		58006700
6379+D	i_value			65535A	const	varying	options(*varsize)			RADDAT			581103		58006800
6380+ *										RADDAT			571125		58006900
6381+ *	Returns a given XML as 'integer'.												571125		58007000
6382+D	UnMarshaller_toInt...									RADDAT			571125		58007100
6383+D		PR				like(wsd1_integer_t)				RADDAT			571125		58007200
6384+D					extproc('WSDL2R98_+					RADDAT			571125		58007300
6385+D					UnMarshaller_toInt+					RADDAT			571125		58007400
6386+D					')					RADDAT			571125		58007500
6387+D	i_value			65535A	const	varying	options(*varsize)			RADDAT			581103		58007600
6388+ *										RADDAT			580121		58007700
6389+ *	Returns a given XML as 'short integer'.												580121		58007800
6390+D	UnMarshaller_toShort...									RADDAT			580121		58007900
6391+D		PR				like(wsd1_short_t)				RADDAT			580121		58008000
6392+D					extproc('WSDL2R98_+					RADDAT			580121		58008100
6393+D					UnMarshaller_toShort+					RADDAT			580121		58008200
6394+D					')					RADDAT			580121		58008300
6395+D	i_value			65535A	const	varying	options(*varsize)			RADDAT			581103		58008400
6396+ *										RADDAT			571125		58008500
6397+ *	Returns a given XML as 'long integer'.												580905		58008600
6398+D	UnMarshaller_toLong...									RADDAT			580905		58008700
6399+D		PR				like(wsd1_long_t)				RADDAT			580905		58008800
6400+D					extproc('WSDL2R98_+					RADDAT			580905		58008900
6401+D					UnMarshaller_toLong+					RADDAT			580905		58009000
6402+D					')					RADDAT			580905		58009100
6403+D	i_value			65535A	const	varying	options(*varsize)			RADDAT			581103		58009200
6404+ *										RADDAT			580905		58009300
6405+ *	Returns a given XML as 'unsigned integer'.												580905		58009400
6406+D	UnMarshaller_toUnsignedInt...									RADDAT			580905		58009500

Line	Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
6407+D		PR like(wsd1_uInteger_t)	RADDAT			580905		58009600
6408+D		extproc('WSDL2R98_+	RADDAT			580905		58009700
6409+D		UnMarshaller_toUnsignedInt+	RADDAT			580905		58009800
6410+D		')	RADDAT			580905		58009900
6411+D	i_value	65535A const varying options(*varsize)	RADDAT			581103		58010000
6412+ *			RADDAT			580905		58010100
6413+ *		Returns a given XML as 'unsigned short integer'.	RADDAT			580905		58010200
6414+D	UnMarshaller_toUnsignedShort...		RADDAT			580905		58010300
6415+D	PR	like(wsd1_uShort_t)	RADDAT			580905		58010400
6416+D		extproc('WSDL2R98_+	RADDAT			580905		58010500
6417+D		UnMarshaller_toUnsignedShort+	RADDAT			580905		58010600
6418+D		')	RADDAT			580905		58010700
6419+D	i_value	65535A const varying options(*varsize)	RADDAT			581103		58010800
6420+ *			RADDAT			580905		58010900
6421+ *		Returns a given XML as 'unsigned long integer'.	RADDAT			580905		58011000
6422+D	UnMarshaller_toUnsignedLong...		RADDAT			580905		58011100
6423+D	PR	like(wsd1_uLong_t)	RADDAT			580905		58011200
6424+D		extproc('WSDL2R98_+	RADDAT			580905		58011300
6425+D		UnMarshaller_toUnsignedLong+	RADDAT			580905		58011400
6426+D		')	RADDAT			580905		58011500
6427+D	i_value	65535A const varying options(*varsize)	RADDAT			581103		58011600
6428+ *			RADDAT			580905		58011700
6429+ *		Returns a given XML as 'non negative integer'.	RADDAT			580905		58011800
6430+D	UnMarshaller_toNonNegativeInteger...		RADDAT			580905		58011900
6431+D	PR	like(wsd1_nonNegInt_t)	RADDAT			580905		58012000
6432+D		extproc('WSDL2R98_+	RADDAT			580905		58012100
6433+D		UnMarshaller_toNonNegativeInteger+	RADDAT			580905		58012200
6434+D		')	RADDAT			580905		58012300
6435+D	i_value	65535A const varying options(*varsize)	RADDAT			581103		58012400
6436+ *			RADDAT			580905		58012500
6437+ *		Returns a given XML as 'non positive integer'.	RADDAT			580905		58012600
6438+D	UnMarshaller_toNonPositiveInteger...		RADDAT			580905		58012700
6439+D	PR	like(wsd1_nonPosInt_t)	RADDAT			580905		58012800
6440+D		extproc('WSDL2R98_+	RADDAT			580905		58012900
6441+D		UnMarshaller_toNonPositiveInteger+	RADDAT			580905		58013000
6442+D		')	RADDAT			580905		58013100
6443+D	i_value	65535A const varying options(*varsize)	RADDAT			581103		58013200
6444+ *			RADDAT			580905		58013300
6445+ *		Returns a given XML as 'negative integer'.	RADDAT			580905		58013400
6446+D	UnMarshaller_toNegativeInteger...		RADDAT			580905		58013500
6447+D	PR	like(wsd1_negInt_t)	RADDAT			580905		58013600
6448+D		extproc('WSDL2R98_+	RADDAT			580905		58013700
6449+D		UnMarshaller_toNegativeInteger+	RADDAT			580905		58013800
6450+D		')	RADDAT			580905		58013900
6451+D	i_value	65535A const varying options(*varsize)	RADDAT			581103		58014000
6452+ *			RADDAT			580905		58014100
6453+ *		Returns a given XML as 'positive integer'.	RADDAT			580905		58014200
6454+D	UnMarshaller_toPositiveInteger...		RADDAT			580905		58014300
6455+D	PR	like(wsd1_posInt_t)	RADDAT			580905		58014400
6456+D		extproc('WSDL2R98_+	RADDAT			580905		58014500
6457+D		UnMarshaller_toPositiveInteger+	RADDAT			580905		58014600
6458+D		')	RADDAT			580905		58014700
6459+D	i_value	65535A const varying options(*varsize)	RADDAT			581103		58014800
6460+ *			RADDAT			580905		58014900
6461+ *		Returns a given XML as 'timestamp'.	RADDAT			571125		58015000
6462+D	UnMarshaller_toDateTime...		RADDAT			571125		58015100

Line Number	----- Source Specifications -----><----- Comments ----->										Do Num	Page Line	Change Date	Src Id	Seq Number
6463+D													571125		58015200
6464+D		PR		Z									571125		58015300
6465+D					extproc('WSDL2R98_+								571125		58015400
6466+D					UnMarshaller_toDateTime+								580905		58015500
6467+D	i_value			65535A	const varying options(*varsize)								581103		58015600
6468+ *													580905		58015700
6469+ *	Returns a given XML as 'date'.												580905		58015800
6470+D	UnMarshaller_toDate...												580905		58015900
6471+D		PR		D									580905		58016000
6472+D					extproc('WSDL2R98_+								580905		58016100
6473+D					UnMarshaller_toDate+								580905		58016200
6474+D)								580905		58016300
6475+D	i_value			65535A	const varying options(*varsize)								581103		58016400
6476+ *													580905		58016500
6477+ *	Returns a given XML as 'time'.												580905		58016600
6478+D	UnMarshaller_toTime...												580905		58016700
6479+D		PR		T									580905		58016800
6480+D					extproc('WSDL2R98_+								580905		58016900
6481+D					UnMarshaller_toTime+								580905		58017000
6482+D)								571125		58017100
6483+D	i_value			65535A	const varying options(*varsize)								581103		58017200
6484+ *													581103		58017300
6485+ *	Returns a given XML as 'base64Binary'.												581103		58017400
6486+D	UnMarshaller_toBase64Binary...												581103		58017500
6487+D		PR		65535A	varying								581103		58017600
6488+D					extproc('WSDL2R98_+								581103		58017700
6489+D					UnMarshaller_toBase64Binary+								581103		58017800
6490+D)								581103		58017900
6491+D	i_value			65535A	const varying options(*varsize)								581103		58018000
6492+ *													590325		58018100
6493+ *	Returns a given XML as 'language'.												590325		58018200
6494+D	UnMarshaller_toLanguage...												590325		58018300
6495+D		PR		65535A	varying								590325		58018400
6496+D					extproc('WSDL2R98_+								590325		58018500
6497+D					UnMarshaller_toLanguage+								590325		58018600
6498+D)								590325		58018700
6499+D	i_value			65535A	const varying options(*varsize)								590325		58018800
6500+ *													590325		58018900
6501+ *	Returns a given XML as 'token'.												590325		58019000
6502+D	UnMarshaller_toToken...												590325		58019100
6503+D		PR		65535A	varying								590325		58019200
6504+D					extproc('WSDL2R98_+								590325		58019300
6505+D					UnMarshaller_toToken+								590325		58019400
6506+D)								590325		58019500
6507+D	i_value			65535A	const varying options(*varsize)								590325		58019600
6508+ *													590325		58019700
6509+ *	Returns a given XML as 'anyURI'.												590325		58019800
6510+D	UnMarshaller_toAnyURI...												590325		58019900
6511+D		PR		65535A	varying								590325		58020000
6512+D					extproc('WSDL2R98_+								590325		58020100
6513+D					UnMarshaller_toAnyURI+								590325		58020200
6514+D)								590325		58020300
6515+D	i_value			65535A	const varying options(*varsize)								590325		58020400
6516+ *													590613		58020500
6517+ *	Returns a given XML as 'duration'.												590613		58020600
6518+D	UnMarshaller_toDuration...												590613		58020700

Line	----- Source Specifications -----><----- Comments ----->										Do	Page	Change	Src	Seq	
Number	1	2	3	4	5	6	7	8	9	10	Num	Line	Date	Id	Number	
6519+D		PR			likeds(wsdL_duration_t)					RADDAT			590613		58020800	
6520+D					extproc('WSDL2R98_+					RADDAT			590613		58020900	
6521+D					UnMarshaller_toDuration+					RADDAT			590613		58021000	
6522+D					')					RADDAT			590613		58021100	
6523+D	i_value		65535A	const	varying options(*varsize)					RADDAT			590613		58021200	
6524+ *										RADDAT			571203		58021300	
6525+ *	- - - - -									RADDAT			571203		58021400	
6526+ *										RADDAT			580206		58021500	
6527+ *	Returns the XML encoded value of a given RPG anyType value.										RADDAT			580206		58021600
6528+D	Marshaller_toAnyType...									RADDAT			580206		58021700	
6529+D		PR	65535A		varying					RADDAT			580206		58021800	
6530+D					extproc('WSDL2R98_+					RADDAT			580206		58021900	
6531+D					Marshaller_toAnyType+					RADDAT			580206		58022000	
6532+D					')					RADDAT			580206		58022100	
6533+D	i_value		65535A	const	varying options(*varsize)					RADDAT			581103		58022200	
6534+ *										RADDAT			580207		58022300	
6535+ *	Returns the XML encoded value of a given RPG anyType value.										RADDAT			580207		58022400
6536+D	Marshaller_toAny...									RADDAT			580207		58022500	
6537+D		PR	65535A		varying					RADDAT			580207		58022600	
6538+D					extproc('WSDL2R98_+					RADDAT			580207		58022700	
6539+D					Marshaller_toAny+					RADDAT			580207		58022800	
6540+D					')					RADDAT			580207		58022900	
6541+D	i_value		65535A	const	varying options(*varsize)					RADDAT			581103		58023000	
6542+ *										RADDAT			571203		58023100	
6543+ *	Returns the XML encoded value of a given RPG schema value.										RADDAT			590602		58023200
6544+D	Marshaller_toSchema...									RADDAT			590602		58023300	
6545+D		PR	65535A		varying					RADDAT			590602		58023400	
6546+D					extproc('WSDL2R98_+					RADDAT			590602		58023500	
6547+D					Marshaller_toSchema+					RADDAT			590602		58023600	
6548+D					')					RADDAT			590602		58023700	
6549+D	i_value		65535A	const	varying options(*varsize)					RADDAT			590602		58023800	
6550+ *										RADDAT			590602		58023900	
6551+ *	Returns the XML encoded value of a given RPG string value.										RADDAT			571203		58024000
6552+D	Marshaller_toString...									RADDAT			571203		58024100	
6553+D		PR	65535A		varying					RADDAT			571203		58024200	
6554+D					extproc('WSDL2R98_+					RADDAT			571203		58024300	
6555+D					Marshaller_toString+					RADDAT			571203		58024400	
6556+D					')					RADDAT			571203		58024500	
6557+D	i_value		65535A	const	varying options(*varsize)					RADDAT			581103		58024600	
6558+ *										RADDAT			571203		58024700	
6559+ *	Returns the XML encoded value of a given RPG boolean value.										RADDAT			571203		58024800
6560+D	Marshaller_toBoolean...									RADDAT			571203		58024900	
6561+D		PR	65535A		varying					RADDAT			571203		58025000	
6562+D					extproc('WSDL2R98_+					RADDAT			571203		58025100	
6563+D					Marshaller_toBoolean+					RADDAT			571203		58025200	
6564+D					')					RADDAT			571203		58025300	
6565+D	i_boolean		N	const						RADDAT			571203		58025400	
6566+ *										RADDAT			571203		58025500	
6567+ *	Returns the XML encoded value of a given RPG decimal value.										RADDAT			571203		58025600
6568+D	Marshaller_toDecimal...									RADDAT			571203		58025700	
6569+D		PR	65535A		varying					RADDAT			571203		58025800	
6570+D					extproc('WSDL2R98_+					RADDAT			571203		58025900	
6571+D					Marshaller_toDecimal+					RADDAT			571203		58026000	
6572+D					')					RADDAT			571203		58026100	
6573+D	i_decimal			const	like(wsdL_decimal_t)					RADDAT			571203		58026200	
6574+ *										RADDAT			571221		58026300	

Line	----- Source Specifications -----><----- Comments ----->										Do	Page	Change	Src	Seq
Number	1	2	3	4	5	6	7	8	9	10	Num	Line	Date	Id	Number
6575+ *	Returns the XML encoded value of a given RPG double value.										RADDAT		571221		58026400
6576+D	Marshaller_toDouble...										RADDAT		571221		58026500
6577+D		PR		65535A		varying					RADDAT		571221		58026600
6578+D						extproc('WSDL2R98_+					RADDAT		571221		58026700
6579+D						Marshaller_toDouble+					RADDAT		571221		58026800
6580+D						')					RADDAT		571221		58026900
6581+D	i_double					const	like(wsd	double_t)		RADDAT		571221		58027000
6582+ *											RADDAT		580319		58027100
6583+ *	Returns the XML encoded value of a given RPG float value.										RADDAT		580319		58027200
6584+D	Marshaller_toFloat...										RADDAT		580319		58027300
6585+D		PR		65535A		varying					RADDAT		580319		58027400
6586+D						extproc('WSDL2R98_+					RADDAT		580319		58027500
6587+D						Marshaller_toFloat+					RADDAT		580319		58027600
6588+D						')					RADDAT		580319		58027700
6589+D	i_float					const	like(wsd	float_t)		RADDAT		580319		58027800
6590+ *											RADDAT		571203		58027900
6591+ *	Returns the XML encoded value of a given RPG integer value.										RADDAT		571203		58028000
6592+D	Marshaller_toInt...										RADDAT		571203		58028100
6593+D		PR		65535A		varying					RADDAT		571203		58028200
6594+D						extproc('WSDL2R98_+					RADDAT		571203		58028300
6595+D						Marshaller_toInt+					RADDAT		571203		58028400
6596+D						')					RADDAT		571203		58028500
6597+D	i_integer					const	like(wsd	integer_t)		RADDAT		571203		58028600
6598+ *											RADDAT		580121		58028700
6599+ *	Returns the XML encoded value of a given RPG short integer value.										RADDAT		580121		58028800
6600+D	Marshaller_toShort...										RADDAT		580121		58028900
6601+D		PR		65535A		varying					RADDAT		580121		58029000
6602+D						extproc('WSDL2R98_+					RADDAT		580121		58029100
6603+D						Marshaller_toShort+					RADDAT		580121		58029200
6604+D						')					RADDAT		580121		58029300
6605+D	i_short					const	like(wsd	short_t)		RADDAT		580121		58029400
6606+ *											RADDAT		571203		58029500
6607+ *	Returns the XML encoded value of a given RPG long integer value.										RADDAT		580905		58029600
6608+D	Marshaller_toLong...										RADDAT		580905		58029700
6609+D		PR		65535A		varying					RADDAT		580905		58029800
6610+D						extproc('WSDL2R98_+					RADDAT		580905		58029900
6611+D						Marshaller_toLong+					RADDAT		580905		58030000
6612+D						')					RADDAT		580905		58030100
6613+D	i_long					const	like(wsd	long_t)		RADDAT		580905		58030200
6614+ *											RADDAT		580905		58030300
6615+ *	Returns the XML encoded value of a given RPG unsigned integer value.										RADDAT		580905		58030400
6616+D	Marshaller_toUnsignedInt...										RADDAT		580905		58030500
6617+D		PR		65535A		varying					RADDAT		580905		58030600
6618+D						extproc('WSDL2R98_+					RADDAT		580905		58030700
6619+D						Marshaller_toUnsignedInt+					RADDAT		580905		58030800
6620+D						')					RADDAT		580905		58030900
6621+D	i_uInteger					const	like(wsd	uInteger_t)		RADDAT		580905		58031000
6622+ *											RADDAT		580905		58031100
6623+ *	Returns the XML encoded value of a given RPG unsigned short integer value.										RADDAT		580905		58031200
6624+D	Marshaller_toUnsignedShort...										RADDAT		580905		58031300
6625+D		PR		65535A		varying					RADDAT		580905		58031400
6626+D						extproc('WSDL2R98_+					RADDAT		580905		58031500
6627+D						Marshaller_toUnsignedShort+					RADDAT		580905		58031600
6628+D						')					RADDAT		580905		58031700
6629+D	i_uShort					const	like(wsd	uShort_t)		RADDAT		580905		58031800
6630+ *											RADDAT		580905		58031900

Line Number	<----- Source Specifications ----->	<----- Comments ----->	Do Num	Page Line	Change Date	Src Id	Seq Number
6631+	* Returns the XML encoded value of a given RPG unsigned long integer value.	RADDAT			580905		58032000
6632+D	Marshaller_toUnsignedLong...	RADDAT			580905		58032100
6633+D	PR 65535A varying	RADDAT			580905		58032200
6634+D	extproc('WSDL2R98_+	RADDAT			580905		58032300
6635+D	Marshaller_toUnsignedLong+	RADDAT			580905		58032400
6636+D	')	RADDAT			580905		58032500
6637+D	i_uLong const like(wsd1_uLong_t)	RADDAT			580905		58032600
6638+ *		RADDAT			580905		58032700
6639+ *	Returns the XML encoded value of a given RPG non negative integer value.	RADDAT			580905		58032800
6640+D	Marshaller_toNonNegativeInteger...	RADDAT			580905		58032900
6641+D	PR 65535A varying	RADDAT			580905		58033000
6642+D	extproc('WSDL2R98_+	RADDAT			580905		58033100
6643+D	Marshaller_toNonNegativeInteger+	RADDAT			580905		58033200
6644+D	')	RADDAT			580905		58033300
6645+D	i_nonNegInt const like(wsd1_nonNegInt_t)	RADDAT			580905		58033400
6646+ *		RADDAT			580905		58033500
6647+ *	Returns the XML encoded value of a given RPG non positive integer value.	RADDAT			580905		58033600
6648+D	Marshaller_toNonPositiveInteger...	RADDAT			580905		58033700
6649+D	PR 65535A varying	RADDAT			580905		58033800
6650+D	extproc('WSDL2R98_+	RADDAT			580905		58033900
6651+D	Marshaller_toNonPositiveInteger+	RADDAT			580905		58034000
6652+D	')	RADDAT			580905		58034100
6653+D	i_nonPosInt const like(wsd1_nonPosInt_t)	RADDAT			580905		58034200
6654+ *		RADDAT			580905		58034300
6655+ *	Returns the XML encoded value of a given RPG negative integer value.	RADDAT			580905		58034400
6656+D	Marshaller_toNegativeInteger...	RADDAT			580905		58034500
6657+D	PR 65535A varying	RADDAT			580905		58034600
6658+D	extproc('WSDL2R98_+	RADDAT			580905		58034700
6659+D	Marshaller_toNegativeInteger+	RADDAT			580905		58034800
6660+D	')	RADDAT			580905		58034900
6661+D	i_negInt const like(wsd1_negInt_t)	RADDAT			580905		58035000
6662+ *		RADDAT			580905		58035100
6663+ *	Returns the XML encoded value of a given RPG positive integer value.	RADDAT			580905		58035200
6664+D	Marshaller_toPositiveInteger...	RADDAT			580905		58035300
6665+D	PR 65535A varying	RADDAT			580905		58035400
6666+D	extproc('WSDL2R98_+	RADDAT			580905		58035500
6667+D	Marshaller_toPositiveInteger+	RADDAT			580905		58035600
6668+D	')	RADDAT			580905		58035700
6669+D	i_posInt const like(wsd1_posInt_t)	RADDAT			580905		58035800
6670+ *		RADDAT			580905		58035900
6671+ *	Returns the XML encoded value of a given RPG timestamp value.	RADDAT			571203		58036000
6672+D	Marshaller_toDateTime...	RADDAT			571203		58036100
6673+D	PR 65535A varying	RADDAT			571203		58036200
6674+D	extproc('WSDL2R98_+	RADDAT			571203		58036300
6675+D	Marshaller_toDateTime+	RADDAT			571203		58036400
6676+D	')	RADDAT			571203		58036500
6677+D	i_timestamp Z const	RADDAT			571203		58036600
6678+ *		RADDAT			571107		58036700
6679+ *	Returns the XML encoded value of a given RPG date value.	RADDAT			580905		58036800
6680+D	Marshaller_toDate...	RADDAT			580905		58036900
6681+D	PR 65535A varying	RADDAT			580905		58037000
6682+D	extproc('WSDL2R98_+	RADDAT			580905		58037100
6683+D	Marshaller_toDate+	RADDAT			580905		58037200
6684+D	')	RADDAT			580905		58037300
6685+D	i_date D const	RADDAT			580905		58037400
6686+D	i_timeZone 6A const options(*nopass)	RADDAT			580905		58037500

Line	<----- Source Specifications -----><----- Comments ----->										Do	Page	Change	Src	Seq
Number	1	2	3	4	5	6	7	8	9	10	Num	Line	Date	Id	Number
6687+ *										RADDAT			580905		58037600
6688+ *	Returns the XML encoded value of a given RPG time value.									RADDAT			580905		58037700
6689+D	Marshaller_toTime...									RADDAT			580905		58037800
6690+D		PR		65535A		varying				RADDAT			580905		58037900
6691+D						extproc('WSDL2R98_+				RADDAT			580905		58038000
6692+D						Marshaller_toTime+				RADDAT			580905		58038100
6693+D						')				RADDAT			580905		58038200
6694+D		i_time		T		const				RADDAT			580905		58038300
6695+D		i_timeZone		6A		const	options(*nopass)			RADDAT			580905		58038400
6696+ *										RADDAT			581103		58038500
6697+ *	Returns the XML (base64Binary) encoded value of a									RADDAT			581103		58038600
6698+ *	given RPG string value.									RADDAT			581103		58038700
6699+D	Marshaller_toBase64Binary...									RADDAT			581103		58038800
6700+D		PR		65535A		varying				RADDAT			581103		58038900
6701+D						extproc('WSDL2R98_+				RADDAT			581103		58039000
6702+D						Marshaller_toBase64Binary+				RADDAT			581103		58039100
6703+D						')				RADDAT			581103		58039200
6704+D		i_value		65535A		const	varying options(*varsize)			RADDAT			581103		58039300
6705+ *										RADDAT			590325		58039400
6706+ *	Returns the XML (language) encoded value of a									RADDAT			590325		58039500
6707+ *	given RPG string value.									RADDAT			590325		58039600
6708+D	Marshaller_toLanguage...									RADDAT			590325		58039700
6709+D		PR		65535A		varying				RADDAT			590325		58039800
6710+D						extproc('WSDL2R98_+				RADDAT			590325		58039900
6711+D						Marshaller_toLanguage+				RADDAT			590325		58040000
6712+D						')				RADDAT			590325		58040100
6713+D		i_value		65535A		const	varying options(*varsize)			RADDAT			590325		58040200
6714+ *										RADDAT			590325		58040300
6715+ *	Returns the XML (token) encoded value of a									RADDAT			590325		58040400
6716+ *	given RPG string value.									RADDAT			590325		58040500
6717+D	Marshaller_toToken...									RADDAT			590325		58040600
6718+D		PR		65535A		varying				RADDAT			590325		58040700
6719+D						extproc('WSDL2R98_+				RADDAT			590325		58040800
6720+D						Marshaller_toToken+				RADDAT			590325		58040900
6721+D						')				RADDAT			590325		58041000
6722+D		i_value		65535A		const	varying options(*varsize)			RADDAT			590325		58041100
6723+ *										RADDAT			590325		58041200
6724+ *	Returns the XML (anyURI) encoded value of a									RADDAT			590325		58041300
6725+ *	given RPG string value.									RADDAT			590325		58041400
6726+D	Marshaller_toAnyURI...									RADDAT			590325		58041500
6727+D		PR		65535A		varying				RADDAT			590325		58041600
6728+D						extproc('WSDL2R98_+				RADDAT			590325		58041700
6729+D						Marshaller_toAnyURI+				RADDAT			590325		58041800
6730+D						')				RADDAT			590325		58041900
6731+D		i_value		65535A		const	varying options(*varsize)			RADDAT			590325		58042000
6732+ *										RADDAT			590613		58042100
6733+ *	Returns the XML (duration) encoded value of a									RADDAT			590613		58042200
6734+ *	given RPG duration structure.									RADDAT			590613		58042300
6735+D	Marshaller_toDuration...									RADDAT			590613		58042400
6736+D		PR		65535A		varying				RADDAT			590613		58042500
6737+D						extproc('WSDL2R98_+				RADDAT			590613		58042600
6738+D						Marshaller_toDuration+				RADDAT			590613		58042700
6739+D						')				RADDAT			590613		58042800
6740+D		i_value				const	likes(wsdL_duration_t)			RADDAT			590613		58042900
6741+ *										RADDAT			590529		58043000
6742+ *	Sets the size of the variable that takes the content									RADDAT			590529		58043100

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
6743+	* of a XML 'schema' element.	RADDAT			590529		58043200
6744+D	Schema_initialize...	RADDAT			590601		58043300
6745+D	PR	RADDAT			590529		58043400
6746+D	extproc('WSDL2R98_+	RADDAT			590529		58043500
6747+D	Schema_initialize+	RADDAT			590601		58043600
6748+D)	RADDAT			590529		58043700
6749+D	i_size 10I 0 value	RADDAT			590529		58043800
6750+ *		RADDAT			590529		58043900
6751+ *	Retrieves athe content of a XML 'schema' element	RADDAT			590529		58044000
6752+ *	from the XML stream.	RADDAT			590529		58044100
6753+D	Schema_start...	RADDAT			590529		58044200
6754+D	PR	RADDAT			590529		58044300
6755+D	extproc('WSDL2R98_+	RADDAT			590529		58044400
6756+D	Schema_start+	RADDAT			590529		58044500
6757+D)	RADDAT			590529		58044600
6758+D	userdata *	RADDAT			590529		58044700
6759+D	depth 10I 0 value	RADDAT			590529		58044800
6760+D	namespace 1024A	RADDAT			590529		58044900
6761+D	name 1024A	RADDAT			590529		58045000
6762+D	path 24576A	RADDAT			590529		58045100
6763+D	attrs *	RADDAT			590529		58045200
6764+D	const options(*varsize)	RADDAT			590529		58045300
6765+ *		RADDAT			590529		58045400
6766+ *	Retrieves athe content of a XML 'schema' element	RADDAT			590529		58045500
6767+ *	from the XML stream.	RADDAT			590529		58045600
6768+D	Schema_end...	RADDAT			590529		58045700
6769+D	PR	RADDAT			590529		58045800
6770+D	extproc('WSDL2R98_+	RADDAT			590529		58045900
6771+D	Schema_end+	RADDAT			590529		58046000
6772+D)	RADDAT			590529		58046100
6773+D	userdata *	RADDAT			590529		58046200
6774+D	depth 10I 0 value	RADDAT			590529		58046300
6775+D	namespace 1024A	RADDAT			590529		58046400
6776+D	name 1024A	RADDAT			590529		58046500
6777+D	path 24576A	RADDAT			590529		58046600
6778+D	value 65535A	RADDAT			590529		58046700
6779+D	attrs *	RADDAT			590529		58046800
6780+D	const options(*varsize)	RADDAT			590529		58046900
6781+ *		RADDAT			590529		58047000
6782+ *	Sets the size of the variable that takes the content	RADDAT			590529		58047100
6783+ *	of a XML 'any' element.	RADDAT			590529		58047200
6784+D	Any_initialize...	RADDAT			590601		58047300
6785+D	PR	RADDAT			590529		58047400
6786+D	extproc('WSDL2R98_+	RADDAT			590529		58047500
6787+D	Any_initialize+	RADDAT			590601		58047600
6788+D)	RADDAT			590529		58047700
6789+D	i_size 10I 0 value	RADDAT			590529		58047800
6790+ *		RADDAT			590529		58047900
6791+ *	Retrieves athe content of a XML 'any' element	RADDAT			590529		58048000
6792+ *	from the XML stream.	RADDAT			590529		58048100
6793+D	Any_start...	RADDAT			590529		58048200
6794+D	PR	RADDAT			590529		58048300
6795+D	extproc('WSDL2R98_+	RADDAT			590529		58048400
6796+D	Any_start+	RADDAT			590529		58048500
6797+D)	RADDAT			590529		58048600
6798+D	userdata *	RADDAT			590529		58048700

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
6799+D	depth 10I 0 value	RADDAT			590529		58048800
6800+D	namespace 1024A varying const	RADDAT			590529		58048900
6801+D	name 1024A varying const	RADDAT			590529		58049000
6802+D	path 24576A varying const	RADDAT			590529		58049100
6803+D	attrs * dim(32767)	RADDAT			590529		58049200
6804+D	const options(*varsize)	RADDAT			590529		58049300
6805+ *		RADDAT			590529		58049400
6806+ *	Retrieves athe content of a XML 'any' element	RADDAT			590529		58049500
6807+ *	from the XML stream.	RADDAT			590529		58049600
6808+D	Any_end...	RADDAT			590529		58049700
6809+D	PR	RADDAT			590529		58049800
6810+D	extproc('WSDL2R98_+	RADDAT			590529		58049900
6811+D	Any_end+	RADDAT			590529		58050000
6812+D)	RADDAT			590529		58050100
6813+D	userdata * value	RADDAT			590529		58050200
6814+D	depth 10I 0 value	RADDAT			590529		58050300
6815+D	namespace 1024A varying const	RADDAT			590529		58050400
6816+D	name 1024A varying const	RADDAT			590529		58050500
6817+D	path 24576A varying const	RADDAT			590529		58050600
6818+D	value 65535A varying const	RADDAT			590529		58050700
6819+D	attrs * dim(32767)	RADDAT			590529		58050800
6820+D	const options(*varsize)	RADDAT			590529		58050900
6821+ *		RADDAT			580905		58051000
6822+ /ENDIF		RADDAT			571108		58051100
6823	/COPY QWSDL2RPG,PWSDL2R42 WSDL2RPG: User name and password		CPY		000000		000332
-----*							
* RPG member name :	PWSDL2R42	*					60
* External name :	WSDL2RPG/QWSDL2RPG(PWSDL2R42)	*					60
* Last change :	07/29/09 10:43:00	*					60
* Text 'description' :	WSDL2RPG: User name and password (Prototype)	*					60
-----*							
6824+ /IF NOT DEFINED(PWSDL2R42)		RADDAT			590418		60000100
6825+ /DEFINE PWSDL2R42		RADDAT			590418		60000200
6826+ *		RADDAT			571108		60000300
6827+ /COPY QWSDL2RPG,TYPES		RADDAT	CPY		571108		60000400
-----*							
* RPG member name :	TYPES	*					61
* External name :	WSDL2RPG/QWSDL2RPG(TYPES)	*					61
* Last change :	07/29/09 10:43:01	*					61
* Text 'description' :	Type definitions	*					61
-----*							
6828+ /IF NOT DEFINED(TYPES)		RADDAT			571106		61000100
LINES EXCLUDED: 693							
6829+ /ENDIF		RADDAT			571106		61069500
6830+ /COPY QWSDL2RPG,WORKSTN		RADDAT	CPY		571204		60000500
-----*							
* RPG member name :	WORKSTN	*					62
* External name :	WSDL2RPG/QWSDL2RPG(WORKSTN)	*					62
* Last change :	07/29/09 10:43:03	*					62
* Text 'description' :	Workstation specific structures and constants	*					62
-----*							
6831+ /IF NOT DEFINED(WORKSTN)		RADDAT			570326		62000100
6832+ /DEFINE WORKSTN		RADDAT			570326		62000200
6833+ *		RADDAT			570816		62000300
6834+ * Key pressed		RADDAT			570816		62000400
6835+D keyPressed_t S 1A based(pDummy)		RADDAT			570816		62000500

Line	----- Source Specifications -----><----- Comments ----->										Do	Page	Change	Src	Seq
Number	1	2	3	4	5	6	7	8	9	10	Num	Line	Date	Id	Number
6836+ *										RADDAT			570326		62000600
6837+ * Workstation information data structure										RADDAT			570326		62000700
6838+D workstn_infds_t...										RADDAT			570326		62000800
6839+D DS qualified based(pDummy)										RADDAT			570326		62000900
6840+D open 9 9A										RADDAT			570326		62001000
6841+D status 11 15S 0										RADDAT			570326		62001100
6842+D AID 369 369A										RADDAT			570326		62001200
6843+ *										RADDAT			000000		62001300
6844+ * Command keys										RADDAT			570326		62001400
6845+D cKeyF1 C CONST(X'31')										RADDAT			000000		62001500
6846+D cKeyF2 C CONST(X'32')										RADDAT			000000		62001600
6847+D cKeyF3 C CONST(X'33')										RADDAT			000000		62001700
6848+D cKeyF4 C CONST(X'34')										RADDAT			000000		62001800
6849+D cKeyF5 C CONST(X'35')										RADDAT			000000		62001900
6850+D cKeyF6 C CONST(X'36')										RADDAT			000000		62002000
6851+D cKeyF7 C CONST(X'37')										RADDAT			000000		62002100
6852+D cKeyF8 C CONST(X'38')										RADDAT			000000		62002200
6853+D cKeyF9 C CONST(X'39')										RADDAT			000000		62002300
6854+D cKeyF10 C CONST(X'3A')										RADDAT			000000		62002400
6855+D cKeyF11 C CONST(X'3B')										RADDAT			000000		62002500
6856+D cKeyF12 C CONST(X'3C')										RADDAT			000000		62002600
6857+D cKeyF13 C CONST(X'B1')										RADDAT			000000		62002700
6858+D cKeyF14 C CONST(X'B2')										RADDAT			000000		62002800
6859+D cKeyF15 C CONST(X'B3')										RADDAT			000000		62002900
6860+D cKeyF16 C CONST(X'B4')										RADDAT			000000		62003000
6861+D cKeyF17 C CONST(X'B5')										RADDAT			000000		62003100
6862+D cKeyF18 C CONST(X'B6')										RADDAT			000000		62003200
6863+D cKeyF19 C CONST(X'B7')										RADDAT			000000		62003300
6864+D cKeyF20 C CONST(X'B8')										RADDAT			000000		62003400
6865+D cKeyF21 C CONST(X'B9')										RADDAT			000000		62003500
6866+D cKeyF22 C CONST(X'BA')										RADDAT			000000		62003600
6867+D cKeyF23 C CONST(X'BB')										RADDAT			000000		62003700
6868+D cKeyF24 C CONST(X'BC')										RADDAT			000000		62003800
6869+ *										RADDAT			000000		62003900
6870+D cKeyClear C CONST(X'BD')										RADDAT			000000		62004000
6871+D cKeyDelete C CONST(X'BD')										RADDAT			000000		62004100
6872+ *										RADDAT			000000		62004200
6873+D cKeyEnter C CONST(X'F1')										RADDAT			000000		62004300
6874+ * Nicht definiert F2										RADDAT			000000		62004400
6875+D cKeyHelp C CONST(X'F3')										RADDAT			000000		62004500
6876+D cKeyPageDown C CONST(X'F4')										RADDAT			000000		62004600
6877+D cKeyPageUp C CONST(X'F5')										RADDAT			000000		62004700
6878+D cKeyPrintScr C CONST(X'F6')										RADDAT			000000		62004800
6879+ * Nicht definiert F7										RADDAT			000000		62004900
6880+D cKeyReset C CONST(X'F8')										RADDAT			000000		62005000
6881+ *										RADDAT			000000		62005100
6882+D cKeyLightPen C CONST(X'3F')										RADDAT			000000		62005200
6883+ *										RADDAT			000000		62005300
6884+D cSflFold C CONST(*off)										RADDAT			000000		62005400
6885+D cSflDrop C CONST(*on)										RADDAT			000000		62005500
6886+ *										RADDAT			570911		62005600
6887+ * 5250 color codes										RADDAT			570911		62005700
6888+ *-----*										RADDAT			570911		62005800
6889+ * Green										RADDAT			570911		62005900
6890+D c5250_GRN C CONST(X'20')										RADDAT			570911		62006000
6891+ * Green/Reverse										RADDAT			570911		62006100

Line	----- Source Specifications -----><----- Comments ----->								Do	Page	Change	Src	Seq
Number1....+....2....+....3....+....4....+....5....+....6....+....7....+....8....+....9....+....10								Num	Line	Date	Id	Number
6892+D	c5250_GRNR	C		CONST(X'21')				RADDAT			570911		62006200
6893+	* Green/Underscore							RADDAT			570911		62006300
6894+D	c5250_GRNU	C		CONST(X'24')				RADDAT			570911		62006400
6895+	* Green/Underscore/Reverse							RADDAT			570911		62006500
6896+D	c5250_GRNUR	C		CONST(X'25')				RADDAT			570911		62006600
6897+	*-----							RADDAT			570911		62006700
6898+	* White							RADDAT			570911		62006800
6899+D	c5250_WHT	C		CONST(X'22')				RADDAT			570911		62006900
6900+	* White/Reverse							RADDAT			570911		62007000
6901+D	c5250_WHTR	C		CONST(X'23')				RADDAT			570911		62007100
6902+	* White/Underscore							RADDAT			570911		62007200
6903+D	c5250_WHTU	C		CONST(X'26')				RADDAT			570911		62007300
6904+	*-----							RADDAT			570911		62007400
6905+	* Red							RADDAT			570911		62007500
6906+D	c5250_RED	C		CONST(X'28')				RADDAT			570911		62007600
6907+	* Red/Reverse							RADDAT			570911		62007700
6908+D	c5250_REDR	C		CONST(X'29')				RADDAT			570911		62007800
6909+	* Red/Blink							RADDAT			570911		62007900
6910+D	c5250_REDB	C		CONST(X'2A')				RADDAT			570911		62008000
6911+	* Red/Reverse/Blink							RADDAT			570911		62008100
6912+D	c5250_REDRB	C		CONST(X'2B')				RADDAT			570911		62008200
6913+	* Red/Underscore							RADDAT			570911		62008300
6914+D	c5250_REDU	C		CONST(X'2C')				RADDAT			570911		62008400
6915+	* Red/Underscore/Reverse							RADDAT			570911		62008500
6916+D	c5250_REDUR	C		CONST(X'2D')				RADDAT			570911		62008600
6917+	* Red/Underscore/Blink							RADDAT			570911		62008700
6918+D	c5250_REDUB	C		CONST(X'2E')				RADDAT			570911		62008800
6919+	*-----							RADDAT			570911		62008900
6920+	* Turquoise/Column Separators							RADDAT			570911		62009000
6921+D	c5250_TRQC	C		CONST(X'30')				RADDAT			570911		62009100
6922+	* Turquoise/Column Separators/Reverse							RADDAT			570911		62009200
6923+D	c5250_TRQCR	C		CONST(X'31')				RADDAT			570911		62009300
6924+	* Turquoise/Underscore							RADDAT			570911		62009400
6925+D	c5250_TRQU	C		CONST(X'34')				RADDAT			570911		62009500
6926+	* Turquoise/Underscore/Reverse							RADDAT			570911		62009600
6927+D	c5250_TRQUR	C		CONST(X'35')				RADDAT			570911		62009700
6928+	*-----							RADDAT			570911		62009800
6929+	* Yellow/Column Separators							RADDAT			570911		62009900
6930+D	c5250_YLWC	C		CONST(X'32')				RADDAT			570911		62010000
6931+	* Yellow/Column Separators/Reverse							RADDAT			570911		62010100
6932+D	c5250_YLWCR	C		CONST(X'33')				RADDAT			570911		62010200
6933+	* Yellow/Underscore							RADDAT			570911		62010300
6934+D	c5250_YLWU	C		CONST(X'36')				RADDAT			570911		62010400
6935+	*-----							RADDAT			570911		62010500
6936+	* Pink							RADDAT			570911		62010600
6937+D	c5250_PNK	C		CONST(X'38')				RADDAT			570911		62010700
6938+	* Pink/Reverse							RADDAT			570911		62010800
6939+D	c5250_PNKR	C		CONST(X'39')				RADDAT			570911		62010900
6940+	* Pink/Underscore							RADDAT			570911		62011000
6941+D	c5250_PNKU	C		CONST(X'3C')				RADDAT			570911		62011100
6942+	* Pink/Underscore/Reverse							RADDAT			570911		62011200
6943+D	c5250_PNKUR	C		CONST(X'3D')				RADDAT			570911		62011300
6944+	*-----							RADDAT			570911		62011400
6945+	* Blue							RADDAT			570911		62011500
6946+D	c5250_BLU	C		CONST(X'3A')				RADDAT			570911		62011600
6947+	* Blue/Reverse							RADDAT			570911		62011700

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
6948+D	c5250_BLUR C	CONST(X'3B')	RADDAT		570911		62011800
6949+ *	Blue/Underscore		RADDAT		570911		62011900
6950+D	c5250_BLUU C	CONST(X'3E')	RADDAT		570911		62012000
6951+ *	-----*		RADDAT		570911		62012100
6952+ *	Nondisplay		RADDAT		570911		62012200
6953+D	c5250_ND1 C	CONST(X'27')	RADDAT		570911		62012300
6954+ *	Nondisplay		RADDAT		570911		62012400
6955+D	c5250_ND2 C	CONST(X'2F')	RADDAT		570911		62012500
6956+ *	Nondisplay		RADDAT		570911		62012600
6957+D	c5250_ND3 C	CONST(X'37')	RADDAT		570911		62012700
6958+ *	Nondisplay		RADDAT		570911		62012800
6959+D	c5250_ND4 C	CONST(X'3F')	RADDAT		570911		62012900
6960+ *			RADDAT		570911		62013000
6961+ /	ENDIF		RADDAT		570911		62013100
6962+ *			RADDAT		590610		60000600
6963+D	cAUTH_TYPE_HTTP...		RADDAT		590610		60000700
6964+D	C	'*HTTP'	RADDAT		590610		60000800
6965+D	cAUTH_TYPE_PROXY...		RADDAT		590610		60000900
6966+D	C	'*PROXY'	RADDAT		590610		60001000
6967+ *			RADDAT		571204		60001100
6968+ *	Returns the login data used to login to a given web server.		RADDAT		590610		60001200
6969+D	getLoginData...		RADDAT		590418		60001300
6970+D	PR	10I 0	RADDAT		590608		60001400
6971+D		extproc('WSDL2R42_+	RADDAT		590418		60001500
6972+D		getLoginData+	RADDAT		590418		60001600
6973+D		')	RADDAT		571204		60001700
6974+D	i_authType	14A const varying	RADDAT		590610		60001800
6975+D	i_realm	126A const varying	RADDAT		590610		60001900
6976+D	i_numAttempts	10I 0 const	RADDAT		590608		60002000
6977+D	o_user	50A varying	RADDAT		590608		60002100
6978+D	o_password	50A varying	RADDAT		590608		60002200
6979+D	o_cmdKey	like(keyPressed_t)	RADDAT		590608		60002300
6980+D		options(*nopass)	RADDAT		590608		60002400
6981+ *			RADDAT		571107		60002500
6982+ /	ENDIF		RADDAT		571108		60002600
6983 *					000000		000333
6984 *	-----				000000		000334
6985 *	Internal prototypes				000000		000335
6986 *	-----				000000		000336
6987 *					000000		000337
6988 *	Sets the content type and the SOAP action of the POST message.				000000		000338
6989 D	HTTP_setContentTypeAndSOAPAction...				000000		000339
6990 D	PR				000000		000340
6991 D		extproc('+			000000		000341
6992 D		HTTP_setContentTypeAndSOAPAction+			000000		000342
6993 D		')			000000		000343
6994 D	o_addlHdrs	1024A varying			000000		000344
6995 D	i_pUserData	* value			000000		000345
6996 *					000000		000346
6997 D	tns_VehiclePerformance_GetByDateResponseStart...				000000		000347
6998 D	PR				000000		000348
6999 D		extproc('tns_VehiclePerformance_GetB+			000000		000349
7000 D		yDateResponseStart')			000000		000350
7001 D	userdata	* value			000000		000351
7002 D	depth	10I 0 value			000000		000352
7003 D	namespace	1024A varying const			000000		000353

Line Number	<----- Source Specifications ----->	<----- Comments ----->	Do Num	Page Line	Change Date	Src Id	Seq Number
7004	D name 1024A varying const				000000		000354
7005	D path 24576A varying const				000000		000355
7006	D attrs *	dim(32767)			000000		000356
7007	D	const options(*varsize)			000000		000357
7008	D tns_VehiclePerformance_GetByDateResponseEnd...				000000		000358
7009	D PR				000000		000359
7010	D	extproc('tns_VehiclePerformance_GetBy+DateResponseEnd')			000000		000360
7011	D				000000		000361
7012	D userdata *	value			000000		000362
7013	D depth 10I 0	value			000000		000363
7014	D namespace 1024A	varying const			000000		000364
7015	D name 1024A	varying const			000000		000365
7016	D path 24576A	varying const			000000		000366
7017	D value 65535A	varying const			000000		000367
7018	D attrs *	dim(32767)			000000		000368
7019	D	const options(*varsize)			000000		000369
7020	D s1_VehiclePerformance_GetByDateResultStart...				000000		000370
7021	D PR				000000		000371
7022	D	extproc('s1_VehiclePerformance_GetBy+DateResultStart')			000000		000372
7023	D				000000		000373
7024	D userdata *	value			000000		000374
7025	D depth 10I 0	value			000000		000375
7026	D namespace 1024A	varying const			000000		000376
7027	D name 1024A	varying const			000000		000377
7028	D path 24576A	varying const			000000		000378
7029	D attrs *	dim(32767)			000000		000379
7030	D	const options(*varsize)			000000		000380
7031	D s1_VehiclePerformance_GetByDateResultEnd...				000000		000381
7032	D PR				000000		000382
7033	D	extproc('s1_VehiclePerformance_GetBy+DateResultEnd')			000000		000383
7034	D				000000		000384
7035	D userdata *	value			000000		000385
7036	D depth 10I 0	value			000000		000386
7037	D namespace 1024A	varying const			000000		000387
7038	D name 1024A	varying const			000000		000388
7039	D path 24576A	varying const			000000		000389
7040	D value 65535A	varying const			000000		000390
7041	D attrs *	dim(32767)			000000		000391
7042	D	const options(*varsize)			000000		000392
7043	D s1_EntryStart_of_s1_VehiclePerformance_GetByDateResult...				000000		000393
7044	D PR				000000		000394
7045	D	extproc('s1_EntryStart_of_s1_VehiclePerformance_GetByDateResult')			000000		000395
7046	D				000000		000396
7047	D				000000		000397
7048	D userdata *	value			000000		000398
7049	D depth 10I 0	value			000000		000399
7050	D namespace 1024A	varying const			000000		000400
7051	D name 1024A	varying const			000000		000401
7052	D path 24576A	varying const			000000		000402
7053	D attrs *	dim(32767)			000000		000403
7054	D	const options(*varsize)			000000		000404
7055	D s1_EntryEnd_of_s1_VehiclePerformance_GetByDateResult...				000000		000405
7056	D PR				000000		000406
7057	D	extproc('s1_EntryEnd_of_s1_VehiclePerformance_GetByDateResult')			000000		000407
7058	D				000000		000408
7059	D				000000		000409

Line	----- Source Specifications -----><----- Comments ----->										Do	Page	Change	Src	Seq
Number	1	2	3	4	5	6	7	8	9	10	Num	Line	Date	Id	Number
7060	D	userdata		*	value								000000		000410
7061	D	depth		10I 0	value								000000		000411
7062	D	namespace		1024A	varying const								000000		000412
7063	D	name		1024A	varying const								000000		000413
7064	D	path		24576A	varying const								000000		000414
7065	D	value		65535A	varying const								000000		000415
7066	D	attrs		*	dim(32767)								000000		000416
7067	D				const options(*varsize)								000000		000417
7068	*												000000		000418
7069	*	SOAP message: Envelope											000000		000419
7070	D	SOAPMessage_Envelope...											000000		000420
7071	D	PR			extproc('+								000000		000421
7072	D				SOAPMessage_Envelope+								000000		000422
7073	D				')								000000		000423
7074	D	userdata		*	value								000000		000424
7075	D	depth		10I 0	value								000000		000425
7076	D	namespace		1024A	varying const								000000		000426
7077	D	name		1024A	varying const								000000		000427
7078	D	path		24576A	varying const								000000		000428
7079	D	attrs		*	dim(32767)								000000		000429
7080	D				const options(*varsize)								000000		000430
7081	*												000000		000431
7082	*	SOAP message: fault information											000000		000432
7083	D	SOAPMessage_FaultStart...											000000		000433
7084	D	PR			extproc('+								000000		000434
7085	D				SOAPMessage_FaultStart+								000000		000435
7086	D				')								000000		000436
7087	D	userdata		*	value								000000		000437
7088	D	depth		10I 0	value								000000		000438
7089	D	namespace		1024A	varying const								000000		000439
7090	D	name		1024A	varying const								000000		000440
7091	D	path		24576A	varying const								000000		000441
7092	D	attrs		*	dim(32767)								000000		000442
7093	D				const options(*varsize)								000000		000443
7094	*												000000		000444
7095	*	SOAP message: fault information											000000		000445
7096	D	SOAPMessage_FaultEnd...											000000		000446
7097	D	PR			extproc('+								000000		000447
7098	D				SOAPMessage_FaultEnd+								000000		000448
7099	D				')								000000		000449
7100	D	userdata		*	value								000000		000450
7101	D	depth		10I 0	value								000000		000451
7102	D	namespace		1024A	varying const								000000		000452
7103	D	name		1024A	varying const								000000		000453
7104	D	path		24576A	varying const								000000		000454
7105	D	value		65535A	varying const								000000		000455
7106	D	attrs		*	dim(32767)								000000		000456
7107	D				const options(*varsize)								000000		000457
7108	*												000000		000458
7109	*	SOAP message: fault details											000000		000459
7110	D	SOAPMessage_FaultDetailStart...											000000		000460
7111	D	PR			extproc('+								000000		000461
7112	D				SOAPMessage_FaultDetailStart+								000000		000462
7113	D				')								000000		000463
7114	D	userdata		*	value								000000		000464
7115	D	depth		10I 0	value								000000		000465

Line Number	<----- Source Specifications ----->	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
7116	D namespace 1024A varying const				000000		000466
7117	D name 1024A varying const				000000		000467
7118	D path 24576A varying const				000000		000468
7119	D attrs *	dim(32767)			000000		000469
7120	D	const options(*varsize)			000000		000470
7121	*				000000		000471
7122	* SOAP message: fault details				000000		000472
7123	D SOAPMessage_FaultDetailEnd...				000000		000473
7124	D PR	extproc('+			000000		000474
7125	D	SOAPMessage_FaultDetailEnd+			000000		000475
7126	D	')			000000		000476
7127	D userdata *	value			000000		000477
7128	D depth 10I 0	value			000000		000478
7129	D namespace 1024A	varying const			000000		000479
7130	D name 1024A	varying const			000000		000480
7131	D path 24576A	varying const			000000		000481
7132	D value 65535A	varying const			000000		000482
7133	D attrs *	dim(32767)			000000		000483
7134	D	const options(*varsize)			000000		000484
7135	*				000000		000485
7136	* SOAP message: NULL callback procedure				000000		000486
7137	D SOAPMessage_NULLStart...				000000		000487
7138	D PR	extproc('+			000000		000488
7139	D	SOAPMessage_NULL+			000000		000489
7140	D	')			000000		000490
7141	D userdata *	value			000000		000491
7142	D depth 10I 0	value			000000		000492
7143	D namespace 1024A	varying const			000000		000493
7144	D name 1024A	varying const			000000		000494
7145	D path 24576A	varying const			000000		000495
7146	D value 65535A	varying const			000000		000496
7147	D attrs *	dim(32767)			000000		000497
7148	D	const options(*varsize)			000000		000498
7149	*				000000		000499
7150	* SOAP message: NULL callback procedure				000000		000500
7151	D SOAPMessage_NULLEnd...				000000		000501
7152	D PR	extproc('+			000000		000502
7153	D	SOAPMessage_NULLEnd+			000000		000503
7154	D	')			000000		000504
7155	D userdata *	value			000000		000505
7156	D depth 10I 0	value			000000		000506
7157	D namespace 1024A	varying const			000000		000507
7158	D name 1024A	varying const			000000		000508
7159	D path 24576A	varying const			000000		000509
7160	D value 65535A	varying const			000000		000510
7161	D attrs *	dim(32767)			000000		000511
7162	D	const options(*varsize)			000000		000512
7163	*				000000		000513
7164	* -----				000000		000514
7165	* Global fields & constants				000000		000515
7166	* -----				000000		000516
7167	*				000000		000517
7168	* HTTP debug log settings				000000		000518
7169	D g_httpDebug DS	qualified			000000		000519
7170	D status N	inz			000000		000520
7171	D fileName 500A	inz varying			000000		000521

Line Number	<----- Source Specifications ----->	Comments ----->	Do Num	Page Line	Change Date	Src Id	Seq Number
7172	* ..1....+....2....+....3....+....4....+....5....+....6....+....7....+....8....+....9....+...10				000000		000522
7173	* Proxy server to use				000000		000523
7174	D g_proxy DS	qualified			000000		000524
7175	D server	256A inz			000000		000525
7176	D port	10I 0 inz			000000		000526
7177	*				000000		000527
7178	* Web service endpoint				000000		000528
7179	D g_url S	32767A varying inz(cURL)			000000		000529
7180	*				000000		000530
7181	* Switch for AXIS bug AXIS-2095				000000		000531
7182	D g_isAXIS_2095 S	N inz(cFalse)			000000		000532
7183	*				000000		000533
7184	* SOAP error information				000000		000534
7185	D g_soapFault DS	likeds(wsdl_soap_Fault_t) inz			000000		000535
7186	*				000000		000536
7187	* User callback to supply login information				000000		000537
7188	* Returns WSDL_SUCCESS if successful, WSDL_ERROR upon error.				000000		000538
7189	D g_getLoginData S	* procptr inz			000000		000539
7190	*				000000		000540
7191	*getLoginData...				R 000000		000541
7192	* PR	10I 0			R 000000		000542
7193	*	extproc(g_getLoginData)			R 000000		000543
7194	* i_authType	14A const varying			R 000000		000544
7195	* i_realm	126A const varying			R 000000		000545
7196	* i_numAttempts	10I 0 const			R 000000		000546
7197	* o_user	50A varying			R 000000		000547
7198	* o_password	50A varying			R 000000		000548
7199	*				000000		000549
7200	* Web service endpoint as specified in the WSDL file.				000000		000550
7201	D cURL C	'https://www.truck-pc.net/WebService+			000000		000551
7202	D	s/FleetWatcher/Reports/V1_9/ReportsS+			000000		000552
7203	D	ervice.asmx'			000000		000553
7204	*				000000		000554
7205	* =====				000000		000555
7206	* Operation: tns:VehiclePerformance_GetByDate				000000		000556
7207	* Returns : tns_VehiclePerformance_GetByDateResponse				000000		000557
7208	* =====				000000		000558
7209	P ReportsServiceBinding_v2_0_VehiclePerformance...				000000		000559
7210	P _GetByDate...				000000		000560
7211	P B	export			000000		000561
7212	D ReportsServiceBinding_v2_0_VehiclePerformance...				000000		000562
7213	D _GetByDate...				000000		000563
7214	D PI	likeds(tns_VehiclePerformance_GetB...			000000		000564
7215	D	yDateResponse_t)			000000		000565
7216	D i_tns_VehiclePerformance_GetByDate...				000000		000566
7217	D	likeds(tns_VehiclePerformance_GetB...			000000		000567
7218	D	yDate_t)			000000		000568
7219	D	const			000000		000569
7220	D o_msg	128A varying			000000		000570
7221	*				000000		000571
7222	* Return value				000000		000572
7223	D rtnValue DS	likeds(tns_VehiclePerformance_GetB...			000000		000573

Line	----- Source Specifications -----><----- Comments ----->										Do	Page	Change	Src	Seq
Number	1	2	3	4	5	6	7	8	9	10	Num	Line	Date	Id	Number
7224	D				yDateResponse_t)								000000		000574
7225	D				inz								000000		000575
7226	*												000000		000576
7227	*	Local fields											000000		000577
7228	D	isUnexpectedError...											000000		000578
7229	D		S	N	inz(cFalse)								000000		000579
7230	D	request	S	32767A	varying inz								000000		000580
7231	D	rc	S	10I 0	inz								000000		000581
7232	D	rcvm0200	DS		likeds(rcvm0200_t) inz								000000		000582
7233	D	msg	DS		likeds(msg_t) inz								000000		000583
7234	D	headers	DS		likeds(headers_t) inz								000000		000584
7235	*												000000		000585
7236	*	Login helper fields											000000		000586
7237	D	rc2	S	10I 0	inz								000000		000587
7238	D	err	S	10I 0	inz								000000		000588
7239	D	lastErr	S	10I 0	inz								000000		000589
7240	D	authType	S	14A	inz varying								000000		000590
7241	D	basic	S	N	inz								000000		000591
7242	D	digest	S	N	inz								000000		000592
7243	D	realm	S	124A	inz								000000		000593
7244	D	numAttempts	S	10I 0	inz								000000		000594
7245	D	userid	S	50A	inz varying								000000		000595
7246	D	password	S	50A	inz varying								000000		000596
7247	*												000000		000597
7248	*	Array iterator fields											000000		000598
7249	D	X_A1	S	10I 0	inz								000000		000599
7250	D	X_A2	S	10I 0	inz								000000		000600
7251	*												000000		000601
7252	D	userData	DS		likeds(tns_VehiclePerformance_GetB...								000000		000602
7253	D				yDateResponse_t)								000000		000603
7254	D				inz								000000		000604
7255	*	-----											000000		000605
7256	/free												000000		000606
7257													000000		000607
7258		o_msg = '';											000000		000608
7259													000000		000609
7260		// Initialize map to manage <multiRef> elements											000000		000610
7261		MultiRef_new();											000000		000611
7262													000000		000612
7263		// Enable/disable HTTP debug log											000000		000613
7264		if (g_httpDebug.status = *ON);										B01 B01	000000		000614
7265		if (g_httpDebug.fileName <> '');										B02 B02	000000		000615
7266		http_debug(*ON: g_httpDebug.fileName);										02	000000		000616
7267		else;										X02 X02	000000		000617
7268		http_debug(*ON);										02	000000		000618
7269		endif;										E02 E02	000000		000619
7270		else;										X01 X01	000000		000620
7271		http_debug(*OFF);										01	000000		000621
7272		endif;										E01 E01	000000		000622
7273													000000		000623
7274		// Set proxy server to use											000000		000624
7275		if (g_proxy.server <> '');										B01 B01	000000		000625
7276		http_setproxy(g_proxy.server: g_proxy.port);										01	000000		000626
7277		else;										X01 X01	000000		000627

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
7278	http_setproxy('': 0);		01		000000		000628
7279	endif;		E01	E01	000000		000629
7280					000000		000630
7281	// Set CCSID for POST data according				000000		000631
7282	// to the encoding of the SOAP envelope				000000		000632
7283	HTTP_SetCCSIDs(1208: HTTP_EBCDIC); // 1208 is the CCSID of UTF-8				000000		000633
7284					000000		000634
7285	request =				000000		000635
7286	'<?xml version="1.0" encoding="UTF-8"?>'	+			000000		000636
7287	'<soapenv:Envelope'	+			000000		000637
7288	' xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"'	+			000000		000638
7289	' xmlns:s1="http://fwapi.DriverTech.com/CommonTypes/2009/04"'	+			000000		000639
7290	+				000000		000640
7291	' xmlns:tns="http://fwapi.DriverTech.com/ReportsService_v2"'	+			000000		000641
7292	'>'	+			000000		000642
7293	'<soapenv:Body>'	+			000000		000643
7294	'<tns:VehiclePerformance_GetByDate>' +				000000		000644
7295	'';				000000		000645
7296	for X_A2 = 1 to i_tns_VehiclePerformance_GetByDate.TruckPCBase.x;		B01		000000		000646
7297	request = request +		01		000000		000647
7298	'<s1:TruckPCBase>' +		01		000000		000648
7299	'<s1:TruckName>' +		01		000000		000649
7300	Marshaller_toString(01		000000		000650
7301	i_tns_VehiclePerformance_GetByDate.TruckPCBase.item(X_A2).Truc...		01		000000		000651
7302	kName) +		01		000000		000652
7303	'</s1:TruckName>' +		01		000000		000653
7304	'</s1:TruckPCBase>' +		01		000000		000654
7305	'';		01		000000		000655
7306	endfor;		E01		000000		000656
7307	request = request +				000000		000657
7308	'<tns:BatchSize>' +				000000		000658
7309	Marshaller_toInt(000000		000659
7310	i_tns_VehiclePerformance_GetByDate.BatchSize) +				000000		000660
7311	'</tns:BatchSize>' +				000000		000661
7312	'<tns:EndDate>' +				000000		000662
7313	Marshaller_toDateTime(000000		000663
7314	i_tns_VehiclePerformance_GetByDate.EndDate) +				000000		000664
7315	'</tns:EndDate>' +				000000		000665
7316	'<tns:QueryByRecordDate>' +				000000		000666
7317	Marshaller_toBoolean(000000		000667
7318	i_tns_VehiclePerformance_GetByDate.QueryByRecordDate) +				000000		000668
7319	'</tns:QueryByRecordDate>' +				000000		000669
7320	'<tns:StartDate>' +				000000		000670
7321	Marshaller_toDateTime(000000		000671
7322	i_tns_VehiclePerformance_GetByDate.StartDate) +				000000		000672
7323	'</tns:StartDate>' +				000000		000673
7324	'</tns:VehiclePerformance_GetByDate>' +				000000		000674
7325	'</soapenv:Body>'	+			000000		000675
7326	'</soapenv:Envelope>;'				000000		000676
7327					000000		000677
7328	// Clear SOAP error code				000000		000678
7329	clear g_soapFault;				000000		000679
7330					000000		000680
7331	// eXpat: Enables namespaces				000000		000681
7332	http_xmlns(*ON);				000000		000682
7333					000000		000683

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
7334	// HTTPAPI: Enables additional headers				000000		000684
7335	http_xproc(HTTP_POINT_ADDL_HEADER				000000		000685
7336	: %paddr('HTTP_setContentTypeAndSOAPAction')				000000		000686
7337	: %addr(headers));				000000		000687
7338					000000		000688
7339	headers.contentType = 'text/xml';				000000		000689
7340	headers.soapAction = 'http://fwapi.DriverTech.com/ReportsService+				000000		000690
7341	_v2/VehiclePerformance_GetByDate';				000000		000691
7342					000000		000692
7343	// eXpat: Post request and parse response				000000		000693
7344	monitor;		B01		000000		000694
7345	dou (err <> HTTP_NDAUTH and err <> HTTP_PXNDAUTH);		B02	B01	000000		000695
7346	if (err <> lastErr);		B03	B02	000000		000696
7347	numAttempts = 1;		03		000000		000697
7348	lastErr = err;		03		000000		000698
7349	endif;		E03	E02	000000		000699
7350	rc = http_post_xml(g_url		02		000000		000700
7351	: %addr(request) + 2		02		000000		000701
7352	: %len(%trimR(request))		02		000000		000702
7353	: %paddr('SOAPMessage_Envelope')		02		000000		000703
7354	: *NULL		02		000000		000704
7355	: %addr(userData)		02		000000		000705
7356	: HTTP_TIMEOUT		02		000000		000706
7357	: HTTP_USERAGENT		02		000000		000707
7358	: '');		02		000000		000708
7359					000000		000709
7360	// The first time we request a document that requires user/password				000000		000710
7361	// it will return a HTTP_NDAUTH error.				000000		000711
7362	if (rc <> 1);		B03	B02	000000		000712
7363	http_error(err);		03		000000		000713
7364	if (err = HTTP_NDAUTH or err = HTTP_PXNDAUTH);		B04	B03	000000		000714
7365	if (g_getLoginData = *NULL);		B05	B04	000000		000715
7366	leave;		05		000000		000716
7367	endif;		E05	E04	000000		000717
7368					000000		000718
7369	numAttempts = numAttempts + 1;		04		000000		000719
7370					000000		000720
7371	if (err = HTTP_NDAUTH);		B05	B04	000000		000721
7372	http_getauth(basic: digest: realm);		05		000000		000722
7373	authType = WSDL_AUTH_TYPE_HTTP;		05		000000		000723
7374	else;		X05	X04	000000		000724
7375	http_proxy_getauth(basic: realm);		05		000000		000725
7376	authType = WSDL_AUTH_TYPE_PROXY;		05		000000		000726
7377	endif;		E05	E04	000000		000727
7378					000000		000728
7379	rc2 = getLoginData(authType: realm		04		000000		000729
7380	: numAttempts: userid: password);		04		000000		000730
7381	if (rc2 <> 1);		B05	B04	000000		000731
7382	userid = '';		05		000000		000732
7383	password = '';		05		000000		000733
7384	leave;		05		000000		000734
7385	endif;		E05	E04	000000		000735
7386					000000		000736
7387	if (err = HTTP_NDAUTH);		B05	B04	000000		000737
7388	if (digest);		B06	B05	000000		000738
7389	http_setAuth(HTTP_AUTH_MD5_DIGEST		06		000000		000739

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
7390	: userid: password);		06		000000		000740
7391	else;		X06	X05	000000		000741
7392	http_setAuth(HTTP_AUTH_BASIC		06		000000		000742
7393	: userid: password);		06		000000		000743
7394	endif;		E06	E05	000000		000744
7395	else;		X05	X04	000000		000745
7396	if (digest);		B06	B05	000000		000746
7397	http_proxy_setAuth(HTTP_AUTH_MD5_DIGEST		06		000000		000747
7398	: userid: password);		06		000000		000748
7399	else;		X06	X05	000000		000749
7400	http_proxy_setAuth(HTTP_AUTH_BASIC		06		000000		000750
7401	: userid: password);		06		000000		000751
7402	endif;		E06	E05	000000		000752
7403	endif;		E05	E04	000000		000753
7404					000000		000754
7405	userid = '';		04		000000		000755
7406	password = '';		04		000000		000756
7407	endif;		E04	E03	000000		000757
7408	else;		X03	X02	000000		000758
7409	err = 0;		03		000000		000759
7410	endif;		E03	E02	000000		000760
7411	enddo;		E02	E01	000000		000761
7412	on-error;		X01		000000		000762
7413	userid = '';		01		000000		000763
7414	password = '';		01		000000		000764
7415	isUnexpectedError = cTrue;		01		000000		000765
7416	rcvm0200 = f_rcvPgmMsg(cMsg_Any: *: cMsg_Act_Same: 0		01		000000		000766
7417	: cMsg_Same: cMsg_ToMyself);		01		000000		000767
7418	msg = f_cvtRcvMsgToMsg(rcvm0200);		01		000000		000768
7419	g_soapFault.httpError.no = WSDL_HTTP_ERROR_UNEXPECTED;		01		000000		000769
7420	g_soapFault.httpError.text = f_rtvMsgText(msg);		01		000000		000770
7421	rc = g_soapFault.httpError.no;		01		000000		000771
7422	o_msg = g_soapFault.httpError.text;		01		000000		000772
7423	endmon;		E01		000000		000773
7424					000000		000774
7425	if (not isUnexpectedError);		B01	B01	000000		000775
7426	g_soapFault.httpError.no = rc;		01		000000		000776
7427	if (g_soapFault.httpError.no <> 1);		B02	B02	000000		000777
7428	g_soapFault.httpError.text = http_error();		02		000000		000778
7429	if (g_soapFault.httpError.text <> '');		B03	B03	000000		000779
7430	o_msg = g_soapFault.httpError.text;		03		000000		000780
7431	else;		X03	X03	000000		000781
7432	o_msg = ReportsServiceBinding_v2_0_SoapError_getText();		03		000000		000782
7433	endif;		E03	E03	000000		000783
7434	clear rtnValue;		02		000000		000784
7435	else;		X02	X02	000000		000785
7436	o_msg = '';		02		000000		000786
7437	rtnValue = userData;		02		000000		000787
7438	endif;		E02	E02	000000		000788
7439	endif;		E01	E02	000000		000789
7440					000000		000790
7441	if (not MultiRef_isNull());		B01	B02	000000		000791
7442	MultiRef_delete();		01		000000		000792
7443	endif;		E01	E02	000000		000793
7444					000000		000794
7445	return rtnValue;				000000		000795

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
7446					000000		000796
7447	/end-free				000000		000797
7448	*				000000		000798
7449	P ReportsServiceBinding_v2_0_VehiclePerformance...				000000		000799
7450	P _GetByDate...				000000		000800
7451	P E				000000		000801
7452	*				000000		000802
7453	*				000000		000803
7454	* =====				000000		000804
7455	* Return value: tns:VehiclePerformance_GetByDateResponse				000000		000805
7456	* =====				000000		000806
7457	P tns_VehiclePerformance_GetByDateResponseStart...				000000		000807
7458	P B				000000		000808
7459	D tns_VehiclePerformance_GetByDateResponseStart...				000000		000809
7460	D PI				000000		000810
7461	D userdata *	value			000000		000811
7462	D depth 10I 0	value			000000		000812
7463	D namespace 1024A	varying const			000000		000813
7464	D name 1024A	varying const			000000		000814
7465	D path 24576A	varying const			000000		000815
7466	D attrs *	dim(32767)			000000		000816
7467	D	const options(*varsize)			000000		000817
7468	*				000000		000818
7469	D tns_VehiclePerformance_GetByDateResponse...				000000		000819
7470	D DS	likes(tns_VehiclePerformance_GetB...			000000		000820
7471	D	yDateResponse_t)			000000		000821
7472	D	based(userdata)			000000		000822
7473	* - - - - -				000000		000823
7474	/free				000000		000824
7475					000000		000825
7476	if (depth = 1 and		B01		000000		000826
7477	name = 'VehiclePerformance_GetByDateResponse'		B01		000000		000827
7478);		B01		000000		000828
7479	if (MultiRef_isReference(attrs));		B02		000000		000829
7480	MultiRef_put(02		000000		000830
7481	attrs		02		000000		000831
7482	: http_parser_get_userdata()		02		000000		000832
7483	: depth		02		000000		000833
7484	: namespace		02		000000		000834
7485	: name		02		000000		000835
7486	: path		02		000000		000836
7487	: http_parser_get_start_cb()		02		000000		000837
7488	: http_parser_get_end_cb());		02		000000		000838
7489	return;		02		000000		000839
7490	endif;		E02		000000		000840
7491	endif;		E01		000000		000841
7492					000000		000842
7493	select;		B01		000000		000843
7494	when (depth = 2		X01		000000		000844
7495	and		X01		000000		000845
7496	name = 'VehiclePerformance_GetByDateResult'		X01		000000		000846
7497	and		X01		000000		000847

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
7498	namespace = 'http://fwapi.DriverTech.com/CommonTypes/2009/04');		X01		000000		000848
7499	http_parser_switch_cb(01		000000		000849
7500	%addr(tns_VehiclePerformance_GetByDateResponse.VehiclePerforma...		01		000000		000850
7501	nce_GetByDateResult)		01		000000		000851
7502	: %paddr('s1_VehiclePerformance_GetByDateResultStart')		01		000000		000852
7503	: %paddr('s1_VehiclePerformance_GetByDateResultEnd'));		01		000000		000853
7504	return;		01		000000		000854
7505					000000		000855
7506	when (depth = 2		X01		000000		000856
7507	and		X01		000000		000857
7508	name = 'TotalRowCount'		X01		000000		000858
7509	and		X01		000000		000859
7510	namespace = 'http://fwapi.DriverTech.com/ReportsService_v2');		X01		000000		000860
7511	// Processed in corresponding END procedure				000000		000861
7512	return;		01		000000		000862
7513					000000		000863
7514	endsl;		E01		000000		000864
7515					000000		000865
7516	return;				000000		000866
7517					000000		000867
7518	/end-free				000000		000868
7519	*				000000		000869
7520	P tns_VehiclePerformance_GetByDateResponseStart...				000000		000870
7521	P E				000000		000871
7522	*				000000		000872
7523	* =====				000000		000873
7524	* Return value: tns:VehiclePerformance_GetByDateResponse				000000		000874
7525	* =====				000000		000875
7526	P tns_VehiclePerformance_GetByDateResponseEnd...				000000		000876
7527	P B				000000		000877
7528	D tns_VehiclePerformance_GetByDateResponseEnd...				000000		000878
7529	D PI				000000		000879
7530	D userdata * value				000000		000880
7531	D depth 10I 0 value				000000		000881
7532	D namespace 1024A varying const				000000		000882
7533	D name 1024A varying const				000000		000883
7534	D path 24576A varying const				000000		000884
7535	D value 65535A varying const				000000		000885
7536	D attrs * dim(32767)				000000		000886
7537	D const options(*varsize)				000000		000887
7538	*				000000		000888
7539	D tns_VehiclePerformance_GetByDateResponse...				000000		000889
7540	D DS likeds(tns_VehiclePerformance_GetB...				000000		000890
7541	D yDateResponse_t)				000000		000891
7542	D based(userdata)				000000		000892
7543	* - - - - -				000000		000893
7544	/free				000000		000894
7545					000000		000895
7546					000000		000896
7547	select;		B01		000000		000897
7548	when (name = 'TotalRowCount'		X01		000000		000898
7549	and		X01		000000		000899

Line Number	<----- Source Specifications ----->	<----- Comments ----->	Do Num	Page Line	Change Date	Src Id	Seq Number
7550	namespace = 'http://fwapi.DriverTech.com/ReportsService_v2');		X01		000000		000900
7551	tns_VehiclePerformance_GetByDateResponse.TotalRowCount =		01		000000		000901
7552	UnMarshaller_toInt(value);		01		000000		000902
7553					000000		000903
7554	endsl;		E01		000000		000904
7555					000000		000905
7556	return;				000000		000906
7557					000000		000907
7558	/end-free				000000		000908
7559	*				000000		000909
7560	P tns_VehiclePerformance_GetByDateResponseEnd...				000000		000910
7561	P E				000000		000911
7562	*				000000		000912
7563	* =====				000000		000913
7564	* Return value: s1:VehiclePerformance_GetByDateResult				000000		000914
7565	* =====				000000		000915
7566	P s1_VehiclePerformance_GetByDateResultStart...				000000		000916
7567	P B				000000		000917
7568	D s1_VehiclePerformance_GetByDateResultStart...				000000		000918
7569	D PI				000000		000919
7570	D userdata * value				000000		000920
7571	D depth 10I 0 value				000000		000921
7572	D namespace 1024A varying const				000000		000922
7573	D name 1024A varying const				000000		000923
7574	D path 24576A varying const				000000		000924
7575	D attrs * dim(32767)				000000		000925
7576	D const options(*varsize)				000000		000926
7577	*				000000		000927
7578	D s1_VehiclePerformance_GetByDateResult...				000000		000928
7579	D DS likeds(s1_VehiclePerformanceReport_t)				000000		000929
7580	D based(userdata)				000000		000930
7581	* - - - - -				000000		000931
7582	/free				000000		000932
7583					000000		000933
7584	if (depth = 1 and		B01		000000		000934
7585	name = 'VehiclePerformance_GetByDateResult'		B01		000000		000935
7586);		B01		000000		000936
7587	if (MultiRef_isReference(attrs));		B02		000000		000937
7588	MultiRef_put(02		000000		000938
7589	attrs		02		000000		000939
7590	: http_parser_get_userdata()		02		000000		000940
7591	: depth		02		000000		000941
7592	: namespace		02		000000		000942
7593	: name		02		000000		000943
7594	: path		02		000000		000944
7595	: http_parser_get_start_cb()		02		000000		000945
7596	: http_parser_get_end_cb());		02		000000		000946
7597	return;		02		000000		000947
7598	endif;		E02		000000		000948
7599	endif;		E01		000000		000949
7600					000000		000950
7601	select;		B01		000000		000951

Line Number	<----- Source Specifications -----><----- Comments ----->	Do Num	Page Line	Change Date	Src Id	Seq Number
7602	when (depth = 2	X01		000000		000952
7603	and	X01		000000		000953
7604	name = 'Entry'	X01		000000		000954
7605	and	X01		000000		000955
7606	namespace = 'http://fwapi.DriverTech.com/CommonTypes/2009/04');	X01		000000		000956
7607	http_parser_switch_cb(01		000000		000957
7608	%addr(s1_VehiclePerformance_GetByDateResult.Entry)	01		000000		000958
7609	: %paddr('s1_EntryStart_of_s1_VehiclePerformance_GetByDateResult+	01		000000		000959
7610	')	01		000000		000960
7611	: %paddr('s1_EntryEnd_of_s1_VehiclePerformance_GetByDateResult')+	01		000000		000961
7612);	01		000000		000962
=====						
*RNF0637 30 a 000962 An operand was expected but was not found; specification is ignored.						
7613	return;	01		000000		000963
7614				000000		000964
7615	endsl;	E01		000000		000965
7616				000000		000966
7617	return;			000000		000967
7618				000000		000968
7619	/end-free			000000		000969
7620	*			000000		000970
7621	P s1_VehiclePerformance_GetByDateResultStart...			000000		000971
7622	P E			000000		000972
7623	*			000000		000973
7624	* =====			000000		000974
7625	* Return value: s1:VehiclePerformance_GetByDateResult			000000		000975
7626	* =====			000000		000976
7627	P s1_VehiclePerformance_GetByDateResultEnd...			000000		000977
7628	P B			000000		000978
7629	D s1_VehiclePerformance_GetByDateResultEnd...			000000		000979
7630	D PI			000000		000980
7631	D userdata * value			000000		000981
7632	D depth 10I 0 value			000000		000982
7633	D namespace 1024A varying const			000000		000983
7634	D name 1024A varying const			000000		000984
7635	D path 24576A varying const			000000		000985
7636	D value 65535A varying const			000000		000986
7637	D attrs * dim(32767)			000000		000987
7638	D const options(*varsize)			000000		000988
7639	*			000000		000989
7640	D s1_VehiclePerformance_GetByDateResult...			000000		000990
7641	D DS likeds(s1_VehiclePerformanceReport_t)			000000		000991
7642	D based(userdata)			000000		000992
7643	* - - - - -			000000		000993
7644	/free			000000		000994
7645				000000		000995
7646				000000		000996
7647	select;	B01		000000		000997
7648	endsl;	E01		000000		000998
7649				000000		000999
7650	return;			000000		001000

Line Number	<----- Source Specifications ----->	<----- Comments ----->	Do Num	Page Line	Change Date	Src Id	Seq Number
7703					000000		001053
7704	return;				000000		001054
7705					000000		001055
7706	/end-free				000000		001056
7707	*				000000		001057
7708	P s1_EntryStart_of_s1_VehiclePerformance_GetByD...				000000		001058
7709	P ateResult...				000000		001059
7710	P E				000000		001060
7711	*				000000		001061
7712	* =====				000000		001062
7713	* Return value: s1:Entry				000000		001063
7714	* =====				000000		001064
7715	P s1_EntryEnd_of_s1_VehiclePerformance_GetByDat...				000000		001065
7716	P eResult...				000000		001066
7717	P B				000000		001067
7718	D s1_EntryEnd_of_s1_VehiclePerformance_GetByDat...				000000		001068
7719	D eResult...				000000		001069
7720	D PI				000000		001070
7721	D userdata * value				000000		001071
7722	D depth 10I 0 value				000000		001072
7723	D namespace 1024A varying const				000000		001073
7724	D name 1024A varying const				000000		001074
7725	D path 24576A varying const				000000		001075
7726	D value 65535A varying const				000000		001076
7727	D attrs * dim(32767)				000000		001077
7728	D const options(*varsize)				000000		001078
7729	*				000000		001079
7730	D s1_Entry DS likeds(s1_RpgArrayOfEntry_t)				000000		001080
7731	D based(userdata)				000000		001081
7732	* - - - - -				000000		001082
7733	/free				000000		001083
7734					000000		001084
7735					000000		001085
7736	select;		B01		000000		001086
7737	when (g_isAXIS_2095		X01		000000		001087
7738	and		X01		000000		001088
7739	name = 'VehiclePerformance_GetByDateResult'		X01		000000		001089
7740	and		X01		000000		001090
7741	namespace = 'http://fwapi.DriverTech.com/CommonTypes/2009/04')		X01		000000		001091
7742	or (not g_isAXIS_2095		X01		000000		001092
7743	and		X01		000000		001093
7744	name = 'Entry'		X01		000000		001094
7745	and		X01		000000		001095
7746	namespace = 'http://fwapi.DriverTech.com/CommonTypes/2009/04');		X01		000000		001096
7747	s1_Entry.DTScanReportEntry(s1_Entry.x) =		01		000000		001097
7748	UnMarshaller_toAnyType(value);		01		000000		001098
7749					000000		001099
7750	endsl;		E01		000000		001100
7751					000000		001101
7752	return;				000000		001102
7753					000000		001103
7754	/end-free				000000		001104

Line Number	<----- Source Specifications -----><----- Comments ----->	Do Num	Page Line	Change Date	Src Id	Seq Number
7755	*1.....2.....3.....4.....5.....6.....7.....8.....9.....10			000000		001105
7756	P s1_EntryEnd_of_s1_VehiclePerformance_GetByDat...			000000		001106
7757	P eResult...			000000		001107
7758	P E			000000		001108
7759	*			000000		001109
7760	* =====			000000		001110
7761	* Sets the content type and the SOAP action of the POST message.			000000		001111
7762	* =====			000000		001112
7763	P HTTP_setContentTypeAndSOAPAction...			000000		001113
7764	P B		ERR	000000		001114
7765	*			000000		001115
7766	D HTTP_setContentTypeAndSOAPAction...			000000		001116
7767	D PI			000000		001117
7768	D o_addlHdrs 1024A value varying			000000		001118
7769	D i_pUserData *			000000		001119
7770	*			000000		001120
7771	* User Data			000000		001121
7772	D headers DS likeds(headers_t) based(i_pUserData)			000000		001122
7773	* - - - - -			000000		001123
7774	/free			000000		001124
7775				000000		001125
7776	o_addlHdrs = cHTTP_HEADER_CONTENT_TYPE + ': ' +			000000		001126
7777	headers.contentType + cEBCDIC_CRLF +			000000		001127
7778	cHTTP_HEADER_SOAP_ACTION + ': ' +			000000		001128
7779	headers.soapAction + cEBCDIC_CRLF;			000000		001129
7780				000000		001130
7781	/end-free			000000		001131
7782	*			000000		001132
7783	P HTTP_setContentTypeAndSOAPAction...			000000		001133
7784	P E			000000		001134
7785	*			000000		001135
7786	* =====			000000		001136
7787	* SOAP message: Envelope			000000		001137
7788	* =====			000000		001138
7789	P SOAPMessage_Envelope...			000000		001139
7790	P B			000000		001140
7791	*			000000		001141
7792	D SOAPMessage_Envelope...			000000		001142
7793	D PI			000000		001143
7794	D userdata *			000000		001144
7795	D depth 10I 0 value			000000		001145
7796	D namespace 1024A varying const			000000		001146
7797	D name 1024A varying const			000000		001147
7798	D path 24576A varying const			000000		001148
7799	D attrs * dim(32767)			000000		001149
7800	D const options(*varsize)			000000		001150
7801	* - - - - -			000000		001151
7802	/free			000000		001152
7803				000000		001153

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
7804	if (path <> '/Envelope/Body');		B01		000000		001154
7805	return;		01		000000		001155
7806	endif;		E01		000000		001156
7807					000000		001157
7808	select;		B01		000000		001158
7809	when (name = 'Fault');		X01		000000		001159
7810	http_parser_switch_cb(01		000000		001160
7811	%addr(g_soapFault)		01		000000		001161
7812	: %paddr('SOAPMessage_FaultStart')		01		000000		001162
7813	: %paddr('SOAPMessage_FaultEnd'));		01		000000		001163
7814	return;		01		000000		001164
7815					000000		001165
7816	when (name = 'multiRef' and MultiRef_isData(attrs));		X01		000000		001166
7817	http_parser_switch_cb(01		000000		001167
7818	*NULL		01		000000		001168
7819	: MultiRef_getStartCallback()		01		000000		001169
7820	: MultiRef_getEndCallback());		01		000000		001170
7821	return;		01		000000		001171
7822					000000		001172
7823	other;		X01		000000		001173
7824	http_parser_switch_cb(01		000000		001174
7825	userdata		01		000000		001175
7826	: %paddr('tns_VehiclePerformance_GetByDateResponseStart')		01		000000		001176
7827	: %paddr('tns_VehiclePerformance_GetByDateResponseEnd'));		01		000000		001177
7828	return;		01		000000		001178
7829	endsl;		E01		000000		001179
7830					000000		001180
7831	return;				000000		001181
7832					000000		001182
7833	/end-free				000000		001183
7834	*				000000		001184
7835	P SOAPMessage_Envelope...				000000		001185
7836	P E				000000		001186
7837	*				000000		001187
7838	* =====				000000		001188
7839	* SOAP message: fault information				000000		001189
7840	* =====				000000		001190
7841	P SOAPMessage_FaultStart...				000000		001191
7842	P B				000000		001192
7843	*				000000		001193
7844	D SOAPMessage_FaultStart...				000000		001194
7845	D PI				000000		001195
7846	D userdata	* value			000000		001196
7847	D depth	10I 0 value			000000		001197
7848	D namespace	1024A varying const			000000		001198
7849	D name	1024A varying const			000000		001199
7850	D path	24576A varying const			000000		001200
7851	D attrs	* dim(32767)			000000		001201
7852	D	const options(*varsize)			000000		001202
7853	*				000000		001203
7854	D Fault...				000000		001204
7855	D DS	based(userdata)			000000		001205
7856	D	likeds(wsdl_soap_Fault_t)			000000		001206
7857	* - - - - -				000000		001207

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
7858	/free				000000		001208
7859					000000		001209
7860	select;		B01		000000		001210
7861	when (name = 'detail');		X01		000000		001211
7862	http_parser_switch_cb(%addr(Fault.detail)		01		000000		001212
7863	: %paddr('SOAPMessage_FaultDetailStart')		01		000000		001213
7864	: %paddr('SOAPMessage_FaultDetailEnd'));		01		000000		001214
7865	endsl;		E01		000000		001215
7866					000000		001216
7867	/end-free				000000		001217
7868	*				000000		001218
7869	P SOAPMessage_FaultStart...				000000		001219
7870	P E				000000		001220
7871	*				000000		001221
7872	* =====				000000		001222
7873	* SOAP message: fault information				000000		001223
7874	* =====				000000		001224
7875	P SOAPMessage_FaultEnd...				000000		001225
7876	P B				000000		001226
7877	*				000000		001227
7878	D SOAPMessage_FaultEnd...				000000		001228
7879	D PI				000000		001229
7880	D userdata	* value			000000		001230
7881	D depth	10I 0 value			000000		001231
7882	D namespace	1024A varying const			000000		001232
7883	D name	1024A varying const			000000		001233
7884	D path	24576A varying const			000000		001234
7885	D value	65535A varying const			000000		001235
7886	D attrs	* dim(32767)			000000		001236
7887	D	const options(*varsize)			000000		001237
7888	*				000000		001238
7889	D Fault...				000000		001239
7890	D DS	based(userdata)			000000		001240
7891	D	likeds(wSDL_soap_Fault_t)			000000		001241
7892	* - - - - -				000000		001242
7893	/free				000000		001243
7894					000000		001244
7895	select;		B01		000000		001245
7896	when (name = 'faultcode');		X01		000000		001246
7897	Fault.faultcode = value;		01		000000		001247
7898	when (name = 'faultstring');		X01		000000		001248
7899	Fault.faultstring = value;		01		000000		001249
7900	endsl;		E01		000000		001250
7901					000000		001251
7902	/end-free				000000		001252
7903	*				000000		001253
7904	P SOAPMessage_FaultEnd...				000000		001254
7905	P E				000000		001255
7906	*				000000		001256
7907	* =====				000000		001257

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
7908	* SOAP message: fault details				000000		001258
7909	* =====				000000		001259
7910	P SOAPMessage_FaultDetailStart...				000000		001260
7911	P B				000000		001261
7912	*				000000		001262
7913	D SOAPMessage_FaultDetailStart...				000000		001263
7914	D PI				000000		001264
7915	D userdata *	value			000000		001265
7916	D depth 10I 0	value			000000		001266
7917	D namespace 1024A	varying const			000000		001267
7918	D name 1024A	varying const			000000		001268
7919	D path 24576A	varying const			000000		001269
7920	D attrs *	dim(32767)			000000		001270
7921	D	const options(*varsize)			000000		001271
7922	*				000000		001272
7923	D detail...				000000		001273
7924	D DS	based(userdata)			000000		001274
7925	D	likeds(wsdl_soap_FaultDetail_t)			000000		001275
7926	* - - - - -				000000		001276
7927	/free				000000		001277
7928					000000		001278
7929	select;		B01		000000		001279
7930	endsl;		E01		000000		001280
7931					000000		001281
7932	/end-free				000000		001282
7933	*				000000		001283
7934	P SOAPMessage_FaultDetailStart...				000000		001284
7935	P E				000000		001285
7936	*				000000		001286
7937	* =====				000000		001287
7938	* SOAP message: fault details				000000		001288
7939	* =====				000000		001289
7940	P SOAPMessage_FaultDetailEnd...				000000		001290
7941	P B				000000		001291
7942	*				000000		001292
7943	D SOAPMessage_FaultDetailEnd...				000000		001293
7944	D PI				000000		001294
7945	D userdata *	value			000000		001295
7946	D depth 10I 0	value			000000		001296
7947	D namespace 1024A	varying const			000000		001297
7948	D name 1024A	varying const			000000		001298
7949	D path 24576A	varying const			000000		001299
7950	D value 65535A	varying const			000000		001300
7951	D attrs *	dim(32767)			000000		001301
7952	D	const options(*varsize)			000000		001302
7953	*				000000		001303
7954	D detail...				000000		001304
7955	D DS	based(userdata)			000000		001305
7956	D	likeds(wsdl_soap_FaultDetail_t)			000000		001306
7957	* - - - - -				000000		001307

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
7958	/free				000000		001308
7959					000000		001309
7960	select;		B01		000000		001310
7961	when (name = 'hostname');		X01		000000		001311
7962	detail.hostname = value;		01		000000		001312
7963	endsl;		E01		000000		001313
7964					000000		001314
7965	/end-free				000000		001315
7966	*				000000		001316
7967	P SOAPMessage_FaultDetailEnd...				000000		001317
7968	P E				000000		001318
7969	*				000000		001319
7970	* =====				000000		001320
7971	* SOAP message: NULL callback procedure				000000		001321
7972	* =====				000000		001322
7973	P SOAPMessage_NULLStart...				000000		001323
7974	P B				000000		001324
7975	*				000000		001325
7976	D SOAPMessage_NULLStart...				000000		001326
7977	D PI				000000		001327
7978	D userdata * value				000000		001328
7979	D depth 10I 0 value				000000		001329
7980	D namespace 1024A varying const				000000		001330
7981	D name 1024A varying const				000000		001331
7982	D path 24576A varying const				000000		001332
7983	D value 65535A varying const				000000		001333
7984	D attrs * dim(32767)				000000		001334
7985	D const options(*varsize)				000000		001335
7986	* - - - - -				000000		001336
7987	/free				000000		001337
7988					000000		001338
7989	p_assert(cFalse: 'Did never expect to get here');				000000		001339
7990					000000		001340
7991	return;				000000		001341
7992					000000		001342
7993	/end-free				000000		001343
7994	*				000000		001344
7995	P SOAPMessage_NULLStart...				000000		001345
7996	P E				000000		001346
7997	*				000000		001347
7998	* =====				000000		001348
7999	* SOAP message: NULL callback procedure				000000		001349
8000	* =====				000000		001350
8001	P SOAPMessage_NULLEnd...				000000		001351
8002	P B				000000		001352
8003	*				000000		001353
8004	D SOAPMessage_NULLEnd...				000000		001354
8005	D PI				000000		001355

Line	<----- Source Specifications -----><----- Comments ----->										Do	Page	Change	Src	Seq
Number	1	2	3	4	5	6	7	8	9	10	Num	Line	Date	Id	Number
8006	D	userdata		*	value								000000		001356
8007	D	depth		10I 0	value								000000		001357
8008	D	namespace		1024A	varying const								000000		001358
8009	D	name		1024A	varying const								000000		001359
8010	D	path		24576A	varying const								000000		001360
8011	D	value		65535A	varying const								000000		001361
8012	D	attrs		*	dim(32767)								000000		001362
8013	D				const options(*varsize)								000000		001363
8014	*	- - - - -	- - - - -	- - - - -	- - - - -	- - - - -	- - - - -	- - - - -	- - - - -	- - - - -			000000		001364
8015	/free												000000		001365
8016													000000		001366
8017		p_assert(cFalse: 'Did never expect to get here');											000000		001367
8018													000000		001368
8019		return;											000000		001369
8020													000000		001370
8021	/end-free												000000		001371
8022	*												000000		001372
8023	P	SOAPMessage_NULLEnd...											000000		001373
8024	P	E											000000		001374
8025	*												000000		001375
8026	*	=====											000000		001376
8027	*	Registers an user callback procedure, that is called whenever											000000		001377
8028	*	a web service requires user authentication.											000000		001378
8029	*	=====											000000		001379
8030	P	ReportsServiceBinding_v2_0_registerLoginCallback...											000000		001380
8031	P	B			export								000000		001381
8032	*												000000		001382
8033	D	ReportsServiceBinding_v2_0_registerLoginCallback...											000000		001383
8034	D	PI											000000		001384
8035	D	i_pLoginProc		*	value	procptr						D	000000		001385
8036	*	- - - - -	- - - - -	- - - - -	- - - - -	- - - - -	- - - - -	- - - - -	- - - - -	- - - - -			000000		001386
8037	/free												000000		001387
8038													000000		001388
8039		g_getLoginData = i_pLoginProc;											000000		001389
8040													000000		001390
8041		return;											000000		001391
8042													000000		001392
8043	/end-free												000000		001393
8044	*												000000		001394
8045	P	ReportsServiceBinding_v2_0_registerLoginCallback...											000000		001395
8046	P	E											000000		001396
8047	*												000000		001397
8048	*	=====											000000		001398
8049	*	Http: Returns the last HTTP error (code) that occurred.											000000		001399
8050	*	=====											000000		001400
8051	P	ReportsServiceBinding_v2_0_HttpError_getCode...											000000		001401
8052	P	B			export								000000		001402
8053	*												000000		001403

Line Number	<----- Source Specifications -----><----- Comments ----->	Do Num	Page Line	Change Date	Src Id	Seq Number
8054	D ReportsServiceBinding_v2_0_HttpError_getCode...			000000		001404
8055	D PI like(wsdl_http_ErrorNo_t)			000000		001405
8056	* - - - - -			000000		001406
8057	/free			000000		001407
8058				000000		001408
8059	return g_soapFault.httpError.no;			000000		001409
8060				000000		001410
8061	/end-free			000000		001411
8062	*			000000		001412
8063	P ReportsServiceBinding_v2_0_HttpError_getCode...			000000		001413
8064	P E			000000		001414
8065	*			000000		001415
8066	* =====			000000		001416
8067	* Http: Returns the last HTTP error (text) that occurred.			000000		001417
8068	* =====			000000		001418
8069	P ReportsServiceBinding_v2_0_HttpError_getText...			000000		001419
8070	P B export			000000		001420
8071	*			000000		001421
8072	D ReportsServiceBinding_v2_0_HttpError_getText...			000000		001422
8073	D PI like(wsdl_http_ErrorText_t)			000000		001423
8074	* - - - - -			000000		001424
8075	/free			000000		001425
8076				000000		001426
8077	return g_soapFault.httpError.text;			000000		001427
8078				000000		001428
8079	/end-free			000000		001429
8080	*			000000		001430
8081	P ReportsServiceBinding_v2_0_HttpError_getText...			000000		001431
8082	P E			000000		001432
8083	*			000000		001433
8084	* =====			000000		001434
8085	* Soap: Returns the last SOAP error (code) that occurred.			000000		001435
8086	* =====			000000		001436
8087	P ReportsServiceBinding_v2_0_SoapError_getCode...			000000		001437
8088	P B export			000000		001438
8089	*			000000		001439
8090	D ReportsServiceBinding_v2_0_SoapError_getCode...			000000		001440
8091	D PI like(wsdl_soap_faultcode_t)			000000		001441
8092	* - - - - -			000000		001442
8093	/free			000000		001443
8094				000000		001444
8095	return g_soapFault.faultcode;			000000		001445
8096				000000		001446
8097	/end-free			000000		001447
8098	*			000000		001448
8099	P ReportsServiceBinding_v2_0_SoapError_getCode...			000000		001449

Line Number	<----- Source Specifications -----><----- Comments ----->	Do Num	Page Line	Change Date	Src Id	Seq Number
8100	P E			000000		001450
8101	*			000000		001451
8102	* =====			000000		001452
8103	* Soap: Returns the last SOAP error (text) that occurred.			000000		001453
8104	* =====			000000		001454
8105	P ReportsServiceBinding_v2_0_SoapError_getText...			000000		001455
8106	P B export			000000		001456
8107	*			000000		001457
8108	D ReportsServiceBinding_v2_0_SoapError_getText...			000000		001458
8109	D PI like(wsd1_soap_faultstring_t)			000000		001459
8110	* - - - - -			000000		001460
8111	/free			000000		001461
8112				000000		001462
8113	return g_soapFault.faultstring;			000000		001463
8114				000000		001464
8115	/end-free			000000		001465
8116	*			000000		001466
8117	P ReportsServiceBinding_v2_0_SoapError_getText...			000000		001467
8118	P E			000000		001468
8119	*			000000		001469
8120	* =====			000000		001470
8121	* *** private/experimental ***	R		000000		001471
8122	* Enables/disables a workaround for Web services that are driven			000000		001472
8123	* by Axis and that are affected by the AXIS-2095 bug.			000000		001473
8124	* Export this procedure if you really need it.			000000		001474
8125	* =====			000000		001475
8126	P ReportsServiceBinding_v2_0_enableAxis2095...			000000		001476
8127	P B			000000		001477
8128	*			000000		001478
8129	D ReportsServiceBinding_v2_0_enableAxis2095...			000000		001479
8130	D PI			000000		001480
8131	D i_isEnabled N const			000000		001481
8132	* - - - - -			000000		001482
8133	/free			000000		001483
8134				000000		001484
8135	g_isAXIS_2095 = i_isEnabled;			000000		001485
8136				000000		001486
8137	return;			000000		001487
8138				000000		001488
8139	/end-free			000000		001489
8140	*			000000		001490
8141	P ReportsServiceBinding_v2_0_enableAxis2095...			000000		001491
8142	P E			000000		001492
8143	*			000000		001493
8144	* =====			000000		001494
8145	* Web service: Enables/disables the debug log of HTTP API			000000		001495
8146	* =====			000000		001496
8147	P ReportsServiceBinding_v2_0_setHttpDebug...			000000		001497

Line Number	<----- Source Specifications ----->	<----- Comments ----->	Do Num	Page Line	Change Date	Src Id	Seq Number
8148	P B export				000000		001498
8149	*				000000		001499
8150	D ReportsServiceBinding_v2_0_setHttpDebug...				000000		001500
8151	D PI				000000		001501
8152	D i_status N const				000000		001502
8153	D i_fileName 500A const varying options(*nopass)				000000		001503
8154	*				000000		001504
8155	* Parameter positions				000000		001505
8156	D p_fileName C 2				000000		001506
8157	* - - - - -				000000		001507
8158	/free				000000		001508
8159					000000		001509
8160	g_httpDebug.status = i_status;				000000		001510
8161					000000		001511
8162	if (%parms() >= p_fileName);	B01			000000		001512
8163	g_httpDebug.fileName = i_fileName;	01			000000		001513
8164	elseif;	X01			000000		001514
8165	g_httpDebug.fileName = '';	01			000000		001515
8166	endif;	E01			000000		001516
8167					000000		001517
8168	return;				000000		001518
8169					000000		001519
8170	/end-free				000000		001520
8171	*				000000		001521
8172	P ReportsServiceBinding_v2_0_setHttpDebug...				000000		001522
8173	P E				000000		001523
8174	*				000000		001524
8175	* =====				000000		001525
8176	* Web service: Sets the proxy server to use for the connection				000000		001526
8177	* =====				000000		001527
8178	P ReportsServiceBinding_v2_0_setProxy...				000000		001528
8179	P B export				000000		001529
8180	*				000000		001530
8181	D ReportsServiceBinding_v2_0_setProxy...				000000		001531
8182	D PI				000000		001532
8183	D i_server 256A const				000000		001533
8184	D i_port 10I 0 const				000000		001534
8185	* - - - - -				000000		001535
8186	/free				000000		001536
8187					000000		001537
8188	g_proxy.server = i_server;				000000		001538
8189	g_proxy.port = i_port;				000000		001539
8190					000000		001540
8191	return;				000000		001541
8192					000000		001542
8193	/end-free				000000		001543
8194	*				000000		001544
8195	P ReportsServiceBinding_v2_0_setProxy...				000000		001545

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
8196	P E				000000		001546
8197	*				000000		001547
8198	* =====				000000		001548
8199	* Web service: Returns the endpoint of the Web service				000000		001549
8200	* =====				000000		001550
8201	P ReportsServiceBinding_v2_0_getEndpoint...				000000		001551
8202	P B export				000000		001552
8203	*				000000		001553
8204	D ReportsServiceBinding_v2_0_getEndpoint...				000000		001554
8205	D PI 32767A varying				000000		001555
8206	* - - - - -				000000		001556
8207	/free				000000		001557
8208					000000		001558
8209	return g_url;				000000		001559
8210					000000		001560
8211	/end-free				000000		001561
8212	*				000000		001562
8213	P ReportsServiceBinding_v2_0_getEndpoint...				000000		001563
8214	P E				000000		001564
8215	*				000000		001565
8216	* =====				000000		001566
8217	* Web service: Sets the endpoint of the Web service				000000		001567
8218	* =====				000000		001568
8219	P ReportsServiceBinding_v2_0_setEndpoint...				000000		001569
8220	P B export				000000		001570
8221	*				000000		001571
8222	D ReportsServiceBinding_v2_0_setEndpoint...				000000		001572
8223	D PI				000000		001573
8224	D i_url 32767A const varying options(*varsize)				000000		001574
8225	* - - - - -				000000		001575
8226	/free				000000		001576
8227					000000		001577
8228	g_url = %subst(i_url: 1: %len(i_url));				000000		001578
8229					000000		001579
8230	return;				000000		001580
8231					000000		001581
8232	/end-free				000000		001582
8233	*				000000		001583
8234	P ReportsServiceBinding_v2_0_setEndpoint...				000000		001584
8235	P E				000000		001585
8236	*				000000		001586
8237	* =====				000000		001587
8238	* Web service: Returns the host name of the Web Service endpoint				000000		001588
8239	* =====				000000		001589
8240	P ReportsServiceBinding_v2_0_getHost...				000000		001590
8241	P B export				000000		001591
8242	*				000000		001592

Line Number	Source Specifications	Comments	Do Num	Page Line	Change Date	Src Id	Seq Number
8243	D ReportsServiceBinding_v2_0_getHost...				000000		001593
8244	D PI 32767A varying				000000		001594
8245	*				000000		001595
8246	D tmpUrl S like(url_t) inz				000000		001596
8247	* - - - - -				000000		001597
8248	/free				000000		001598
8249					000000		001599
8250	tmpUrl = URL_new(g_url);				000000		001600
8251					000000		001601
8252	return URL_getHost(tmpUrl);				000000		001602
8253					000000		001603
8254	/end-free				000000		001604
8255	*				000000		001605
8256	P ReportsServiceBinding_v2_0_getHost...				000000		001606
8257	P E				000000		001607
8258	*				000000		001608
8259	* =====				000000		001609
8260	* Web service: Sets the host name of the Web service endpoint				000000		001610
8261	* =====				000000		001611
8262	P ReportsServiceBinding_v2_0_setHost...				000000		001612
8263	P B export				000000		001613
8264	*				000000		001614
8265	D ReportsServiceBinding_v2_0_setHost...				000000		001615
8266	D PI				000000		001616
8267	D i_host 32767A const varying options(*varsize)				000000		001617
8268	*				000000		001618
8269	D tmpUrl S like(url_t) inz				000000		001619
8270	* - - - - -				000000		001620
8271	/free				000000		001621
8272					000000		001622
8273	tmpUrl = URL_new(g_url);				000000		001623
8274	URL_setHost(tmpUrl: %subst(i_host: 1: %len(i_host)));				000000		001624
8275	g_url = URL_toString(tmpUrl);				000000		001625
8276					000000		001626
8277	return;				000000		001627
8278					000000		001628
8279	/end-free				000000		001629
8280	*				000000		001630
8281	P ReportsServiceBinding_v2_0_setHost...				000000		001631
8282	P E				000000		001632
8283	*				000000		001633
8284	* =====				000000		001634
8285	* Web service: Returns the port of the Web Service endpoint				000000		001635
8286	* =====				000000		001636
8287	P ReportsServiceBinding_v2_0_getPort...				000000		001637
8288	P B export				000000		001638
8289	*				000000		001639
8290	D ReportsServiceBinding_v2_0_getPort...				000000		001640

A d d i t i o n a l D i a g n o s t i c M e s s a g e s

Msg id	Sv	Number	Seq	Message text
*RNF7030	30	7488	000838	The name or indicator HTTP_PA... is not defined.
*RNF7030	30	7487	000837	The name or indicator HTTP_PA... is not defined.
*RNF7030	30	7482	000832	The name or indicator HTTP_PA... is not defined.
*RNF7030	30	7499	000849	The name or indicator HTTP_PA... is not defined.
*RNF5409	00	7276	000626	The prototyped call returns a value which is lost when CALLP is used.
*RNF5409	00	7278	000628	The prototyped call returns a value which is lost when CALLP is used.
*RNF5409	00	7283	000633	The prototyped call returns a value which is lost when CALLP is used.
*RNF5409	00	7335	000685	The prototyped call returns a value which is lost when CALLP is used.
*RNF5409	00	7363	000713	The prototyped call returns a value which is lost when CALLP is used.
*RNF5409	00	7372	000722	The prototyped call returns a value which is lost when CALLP is used.
*RNF5409	00	7375	000725	The prototyped call returns a value which is lost when CALLP is used.
*RNF5409	00	7389	000739	The prototyped call returns a value which is lost when CALLP is used.
*RNF5409	00	7392	000742	The prototyped call returns a value which is lost when CALLP is used.
*RNF5409	00	7397	000747	The prototyped call returns a value which is lost when CALLP is used.
*RNF5409	00	7400	000750	The prototyped call returns a value which is lost when CALLP is used.
*RNF7503	30	7482	000832	Expression contains an operand that is not defined.
*RNF5410	30	7480	000830	The prototype for the call is not defined.
*RNF7503	30	7499	000849	Expression contains an operand that is not defined.
*RNF7503	30	7590	000940	Expression contains an operand that is not defined.
*RNF5410	30	7588	000938	The prototype for the call is not defined.
*RNF7503	30	7685	001035	Expression contains an operand that is not defined.
*RNF5410	30	7683	001033	The prototype for the call is not defined.
*RNF7503	30	7810	001160	Expression contains an operand that is not defined.
*RNF7503	30	7817	001167	Expression contains an operand that is not defined.
*RNF7503	30	7824	001174	Expression contains an operand that is not defined.
*RNF7503	30	7862	001212	Expression contains an operand that is not defined.
*RNF5409	00	8274	001624	The prototyped call returns a value which is lost when CALLP is used.
*RNF5409	00	8321	001671	The prototyped call returns a value which is lost when CALLP is used.

* * * * * E N D O F A D D I T I O N A L D I A G N O S T I C M E S S A G E S * * * * *

/ C o p y M e m b e r s

Line	Src	RPG name	External name	CCSID	Last change
Number	Id	Library	File	Member	Date Time
326	1	PBASICS1	WSDL2RPG	QBASICS1	37 08/19/09 12:46:31
350	2	PBASICS1R1	WSDL2RPG	PBASICS1R1	37 08/19/09 12:46:31
355	3	FDMSG	WSDL2RPG	FDMSG	37 08/19/09 12:46:31
379	4	FDRCVM0200	WSDL2RPG	FDRCVM0200	37 08/19/09 12:46:31
455	5	FDRTVM0100	WSDL2RPG	FDRTVM0100	37 08/19/09 12:46:31

479	6	FDRTVM0300	WSDL2RPG	QBASICS1	FDRTVM0300	37	08/19/09	12:46:31
545	7	FDRTVM0400	WSDL2RPG	QBASICS1	FDRTVM0400	37	08/19/09	12:46:31
893	8	PBASICS1R2	WSDL2RPG	QBASICS1	PBASICS1R2	37	08/19/09	12:46:31
898	9	FDERRCODE	WSDL2RPG	QBASICS1	FDERRCODE	37	08/19/09	12:46:30
970	10	PBASICS1R1	WSDL2RPG	QBASICS1	PBASICS1R1	37	08/19/09	12:46:31
975	11	PBASICS1R4	WSDL2RPG	QBASICS1	PBASICS1R4	37	08/19/09	12:46:32
1014	12	PBASICS1R2	WSDL2RPG	QBASICS1	PBASICS1R2	37	08/19/09	12:46:31
1017	13	PBASICS1R5	WSDL2RPG	QBASICS1	PBASICS1R5	37	08/19/09	12:46:32
1022	14	FDQOBJ	WSDL2RPG	QBASICS1	FDQOBJ	37	08/19/09	12:46:31
1034	15	FDOBJD0100	WSDL2RPG	QBASICS1	FDOBJD0100	37	08/19/09	12:46:31
1055	16	FDMBRD0100	WSDL2RPG	QBASICS1	FDMBRD0100	37	08/19/09	12:46:31
1075	17	FDMBRD0200	WSDL2RPG	QBASICS1	FDMBRD0200	37	08/19/09	12:46:31
1336	18	MILIB	WSDL2RPG	QBASICS1	MILIB	37	08/19/09	12:46:31
1570	19	PBASICS1R1	WSDL2RPG	QBASICS1	PBASICS1R1	37	08/19/09	12:46:31
1573	20	PBASICS1R7	WSDL2RPG	QBASICS1	PBASICS1R7	37	08/19/09	12:46:32
1705	21	PBASICS1R3	WSDL2RPG	QBASICS1	PBASICS1R3	37	08/19/09	12:46:32
2052	22	PBASICS1R2	WSDL2RPG	QBASICS1	PBASICS1R2	37	08/19/09	12:46:31
2055	23	PBASICS1R4	WSDL2RPG	QBASICS1	PBASICS1R4	37	08/19/09	12:46:32
2058	24	PBASICS1R7	WSDL2RPG	QBASICS1	PBASICS1R7	37	08/19/09	12:46:32
2063	25	PBASICS1R4	WSDL2RPG	QBASICS1	PBASICS1R4	37	08/19/09	12:46:32
2075	26	PBASICS1R5	WSDL2RPG	QBASICS1	PBASICS1R5	37	08/19/09	12:46:32
2080	27	PBASICS1R2	WSDL2RPG	QBASICS1	PBASICS1R2	37	08/19/09	12:46:31
2083	28	PBASICS1R3	WSDL2RPG	QBASICS1	PBASICS1R3	37	08/19/09	12:46:32
2086	29	PBASICS1R4	WSDL2RPG	QBASICS1	PBASICS1R4	37	08/19/09	12:46:32
2089	30	PBASICS1R5	WSDL2RPG	QBASICS1	PBASICS1R5	37	08/19/09	12:46:32
2092	31	PBASICS1R6	WSDL2RPG	QBASICS1	PBASICS1R6	37	08/19/09	12:46:32
2097	32	FDENVVAR	WSDL2RPG	QBASICS1	FDENVVAR	37	08/19/09	12:46:30
2204	33	PBASICS1R1	WSDL2RPG	QBASICS1	PBASICS1R1	37	08/19/09	12:46:31
2207	34	PBASICS1R2	WSDL2RPG	QBASICS1	PBASICS1R2	37	08/19/09	12:46:31
2212	35	PBASICS1R7	WSDL2RPG	QBASICS1	PBASICS1R7	37	08/19/09	12:46:32
2215	36	PBASICS1R8	WSDL2RPG	QBASICS1	PBASICS1R8	37	08/19/09	12:46:32
2302	37	PBASICS1R4	WSDL2RPG	QBASICS1	PBASICS1R4	37	08/19/09	12:46:32
2307	38	PBASICS1R9	WSDL2RPG	QBASICS1	PBASICS1R9	37	08/19/09	12:46:32
2373	39	PBASICS1R2	WSDL2RPG	QBASICS1	PBASICS1R2	37	08/19/09	12:46:31
2378	40	PBASICS1RA	WSDL2RPG	QBASICS1	PBASICS1RA	37	08/19/09	12:46:31
2679	41	PBASICS1R4	WSDL2RPG	QBASICS1	PBASICS1R4	37	08/19/09	12:46:32
2684	42	PBASICS1RB	WSDL2RPG	QBASICS1	PBASICS1RB	37	08/19/09	12:46:31
2788	43	PBASICS1RA	WSDL2RPG	QBASICS1	PBASICS1RA	37	08/19/09	12:46:31
2791	44	PBASICS1R4	WSDL2RPG	QBASICS1	PBASICS1R4	37	08/19/09	12:46:32
2796	45	PBASICS1RC	WSDL2RPG	QBASICS1	PBASICS1RC	37	08/19/09	12:46:31
3030	46	PBASICS1RA	WSDL2RPG	QBASICS1	PBASICS1RA	37	08/19/09	12:46:31
3033	47	PBASICS1R4	WSDL2RPG	QBASICS1	PBASICS1R4	37	08/19/09	12:46:32
3038	48	PBASICS1RD	WSDL2RPG	QBASICS1	PBASICS1RD	37	08/19/09	12:46:31
3042	49	FDURL	WSDL2RPG	QBASICS1	FDURL	37	08/19/09	12:46:31
3214	50	PBASICS1RE	WSDL2RPG	QBASICS1	PBASICS1RE	37	08/19/09	12:46:31
3235	51	EXPAT_H	LIBHTTP	QRPGLSRC	EXPAT_H	37	08/25/08	09:59:46
3633	52	HTTPAPI_H	DTECHTEST2	QRPGLSRC	HTTPAPI_H	37	01/09/08	14:57:15
3669	53	CONFIG_H	LIBHTTP	QRPGLSRC	CONFIG_H	37	08/25/08	10:03:08
5466	54	PWSDL2R94	WSDL2RPG	QWSDL2RPG	PWSDL2R94	37	07/29/09	10:43:00
5470	55	TYPES	WSDL2RPG	QWSDL2RPG	TYPES	37	07/29/09	10:43:01
5478	56	PBASICS1	WSDL2RPG	QBASICS1	PBASICS1	37	08/19/09	12:46:31
6168	57	PBASICS1	WSDL2RPG	QBASICS1	PBASICS1	37	08/19/09	12:46:31
6309	58	PWSDL2R98	WSDL2RPG	QWSDL2RPG	PWSDL2R98	37	07/29/09	10:43:01
6313	59	TYPES	WSDL2RPG	QWSDL2RPG	TYPES	37	07/29/09	10:43:01
6823	60	PWSDL2R42	WSDL2RPG	QWSDL2RPG	PWSDL2R42	37	07/29/09	10:43:00
6827	61	TYPES	WSDL2RPG	QWSDL2RPG	TYPES	37	07/29/09	10:43:01
6830	62	WORKSTN	WSDL2RPG	QWSDL2RPG	WORKSTN	37	07/29/09	10:43:03

* * * * * E N D O F / C O P Y M E M B E R S * * * * *

C r o s s R e f e r e n c e

File and Record References:

File Record	Device	References (D=Defined)
----------------	--------	------------------------

No references in the source.

Global Field References:

Field	Attributes	References (D=Defined M=Modified)
*RNF7031 ACTINF_T	DS(48) BASED(PDUMMY)	1473D
*RNF7031 ACTGRPMARK	I(10,0)	1477D
*RNF7031 ACTMARK	I(10,0)	1478D
*RNF7031 BYTAVL	I(10,0)	1475D
*RNF7031 BYTRET	I(10,0)	1474D
*RNF7031 FLAGS	A(1)	1480D
*RNF7031 RESERVED_1	A(8)	1476D
*RNF7031 RESERVED_2	A(7)	1479D
*RNF7031 RESERVED_3	A(16)	1481D
*RNF7031 ACTPROC_T	DS(298) BASED(PDUMMY)	1134D 1314
*RNF7031 ACTMARK	I(10,0)	1136D
*RNF7031 LIB	A(10)	1139D
*RNF7031 NAME	A(256) VARYING	1137D
*RNF7031 PROCPTR	*(16) PROCPTR	1135D
*RNF7031 SRVPGM	A(10)	1138D
*RNF7031 ACTSRVPGM_T	DS(40) BASED(PDUMMY)	1128D 1303 1319
*RNF7031 ACTMARK	I(10,0)	1130D
*RNF7031 LIB	A(10)	1132D
*RNF7031 NAME	A(10)	1131D
*RNF7031 PROCPTR	*(16) PROCPTR	1129D
*RNF7031 ANDSTR	PROTOTYPE	1511D
*RNF7031 ANY_END	PROTOTYPE	6809D
*RNF7031 ANY_INITIALIZE	PROTOTYPE	6785D
*RNF7031 ANY_START	PROTOTYPE	6794D
AUTH_ALL	CONST	1357D
AUTH_ALTER	CONST	1355D
AUTH_DELETE	CONST	1349D
AUTH_EXCLUDED	CONST	1352D
AUTH_EXECUTE	CONST	1354D
AUTH_INSERT	CONST	1348D
AUTH_LST_MGMT	CONST	1353D
AUTH_NONE	CONST	1358D
AUTH_OBJ_CTRL	CONST	1343D
AUTH_OBJ_MGMT	CONST	1344D
AUTH_OWNER	CONST	1351D
AUTH_POINTER	CONST	1345D
AUTH_REF	CONST	1356D
AUTH_RETRIEVE	CONST	1347D
AUTH_SPACE	CONST	1346D
AUTH_UPDATE	CONST	1350D
*RNF7031 BASICLIST_ADD	I(10,0) PROTOTYPE	2496D

*RNF7031	BASICLIST_ADDNULL	I(10,0) PROTOTYPE	2508D			
*RNF7031	BASICLIST_ADDPOINTER...	I(10,0) PROTOTYPE	2525D			
*RNF7031	BASICLIST_ADDSTRING...	I(10,0) PROTOTYPE	2516D			
*RNF7031	BASICLIST_CLEAR	I(10,0) PROTOTYPE	2638D			
*RNF7031	BASICLIST_CLONE	*(16) PROTOTYPE	2414D			
*RNF7031	BASICLIST_DELETE	I(10,0) PROTOTYPE	2646D			
*RNF7031	BASICLIST_FIND	I(10,0) PROTOTYPE	2534D			
*RNF7031	BASICLIST_FINDSTRING...	I(10,0) PROTOTYPE	2544D			
*RNF7031	BASICLIST_GET	*(16) PROTOTYPE	2422D			
*RNF7031	BASICLIST_GETLISTTYPE...	A(32) PROTOTYPE	2451D			
*RNF7031	BASICLIST_GETNUME	I(10,0) PROTOTYPE	2613D			
*RNF7031	BASICLIST_GETPOINTER...	*(16) PROTOTYPE	2442D			
*RNF7031	BASICLIST_GETSIZE	I(10,0) PROTOTYPE	2487D			
*RNF7031	BASICLIST_GETSTRING...	A(4096) VARYING PROTOTYPE	2433D			
*RNF7031	BASICLIST_GETTYPE	A(32) PROTOTYPE	2478D			
*RNF7031	BASICLIST_HANDLE_T...	*(16) BASED(PDUMMY)	2383D 2426 2463 2500 2538 2579 2617 2650 2689	2405 2437 2472 2512 2548 2589 2625 2654 2801	2414 2446 2482 2520 2557 2599 2634 2665	2418 2455 2491 2529 2570 2608 2642 2673
*RNF7031	BASICLIST_ISLISTTYPEOF...	N(1) PROTOTYPE	2459D			
*RNF7031	BASICLIST_ISNULL	N(1) PROTOTYPE	2661D			
*RNF7031	BASICLIST_ISNULLVALUE...	N(1) PROTOTYPE	2669D			
*RNF7031	BASICLIST_ISSORTED...	N(1) PROTOTYPE	2630D			

*RNF7031	BASICLIST_ISTYPEOF...	N(1)	2468D			
		PROTOTYPE				
*RNF7031	BASICLIST_NEW	*(16)	2405D			
		PROTOTYPE				
*RNF7031	BASICLIST_NULL	*(16)	2654D			
		PROTOTYPE				
*RNF7031	BASICLIST_REMOVE	I(10,0)	2604D			
		PROTOTYPE				
*RNF7031	BASICLIST_SET	I(10,0)	2553D			
		PROTOTYPE				
*RNF7031	BASICLIST_SETAUTOSORT...	N(1)	2595D			
		PROTOTYPE				
*RNF7031	BASICLIST_SETNULL	I(10,0)	2566D			
		PROTOTYPE				
*RNF7031	BASICLIST_SETPOINTER...	I(10,0)	2585D			
		PROTOTYPE				
*RNF7031	BASICLIST_SETSTRING...	I(10,0)	2575D			
		PROTOTYPE				
*RNF7031	BASICLIST_SORT	N(1)	2621D			
		PROTOTYPE				
*RNF7031	BASICLIST_TYPE_T	A(32)	2385D	2409	2451	2464
		BASED(PDUMMY)	2474	2478	2503	2561
*RNF7031	BASICMAP_CLEAR	I(10,0)	2966D			
		PROTOTYPE				
*RNF7031	BASICMAP_CLONE	*(16)	2819D			
		PROTOTYPE				
*RNF7031	BASICMAP_CONTAINSKEY...	N(1)	2837D			
		PROTOTYPE				
*RNF7031	BASICMAP_DELETE	I(10,0)	2974D			
		PROTOTYPE				
*RNF7031	BASICMAP_GET	*(16)	2847D			
		PROTOTYPE				
*RNF7031	BASICMAP_GETINTEGER...	I(10,0)	2869D			
		PROTOTYPE				
*RNF7031	BASICMAP_GETKEYBYINDEX...	A(4096)	2890D			
		VARYING				
		PROTOTYPE				
*RNF7031	BASICMAP_GETNUME	I(10,0)	2958D			
		PROTOTYPE				
*RNF7031	BASICMAP_GETSIZE	I(10,0)	2939D			
		PROTOTYPE				
*RNF7031	BASICMAP_GETSIZEBYINDEX...	I(10,0)	2949D			
		PROTOTYPE				
*RNF7031	BASICMAP_GETSTRING...	A(4096)	2859D			
		VARYING				
		PROTOTYPE				
*RNF7031	BASICMAP_GETVALUEBYINDEX...	*(16)	2879D			
		PROTOTYPE				

*RNF7031	BASICMAP_HANDLE_T	*(16) BASED(PDUMMY)	2801D 2831 2873 2919 2962 2993	2811 2841 2883 2932 2970 3001	2819 2851 2894 2943 2978 5885	2823 2863 2905 2953 2982 6267
*RNF7031	BASICMAP_ISNULL	N(1) PROTOTYPE	2989D			
*RNF7031	BASICMAP_KEY_T	A(4096) BASED(PDUMMY) VARYING	2804D 2864 2920 3013	2832 2874 2933	2842 2890 2944	2852 2906 3005
*RNF7031	BASICMAP_KEYSET_DELETE...	I(10,0) PROTOTYPE	3021D			
*RNF7031	BASICMAP_KEYSET_GETFIRST...	A(4096) VARYING PROTOTYPE	3005D			
*RNF7031	BASICMAP_KEYSET_GETNEXT...	A(4096) VARYING PROTOTYPE	3013D			
*RNF7031	BASICMAP_KEYSET_HANDLE_T...	*(16) BASED(PDUMMY)	2807D 3025	2997	3009	3017
*RNF7031	BASICMAP_KEYSET_NEW...	*(16) PROTOTYPE	2997D			
*RNF7031	BASICMAP_NEW	*(16) PROTOTYPE	2811D			
*RNF7031	BASICMAP_NULL	*(16) PROTOTYPE	2982D			
*RNF7031	BASICMAP_PUT	I(10,0) PROTOTYPE	2901D			
*RNF7031	BASICMAP_PUTINTEGER...	I(10,0) PROTOTYPE	2928D			
*RNF7031	BASICMAP_PUTSTRING...	I(10,0) PROTOTYPE	2915D			
*RNF7031	BASICMAP_REMOVE	I(10,0) PROTOTYPE	2827D			
*RNF7031	BASICSTACK_CLEAR	I(10,0) PROTOTYPE	2756D			
*RNF7031	BASICSTACK_CLONE	*(16) PROTOTYPE	2702D			
*RNF7031	BASICSTACK_DELETE	I(10,0) PROTOTYPE	2764D			
*RNF7031	BASICSTACK_GETNUME...	I(10,0) PROTOTYPE	2748D			
*RNF7031	BASICSTACK_GETSIZE...	I(10,0) PROTOTYPE	2740D			
*RNF7031	BASICSTACK_HANDLE_T...	*(16) BASED(PDUMMY)	2689D 2714	2694 2724	2702 2734	2706 2744

			2752	2760	2768	2772
			2783			
*RNF7031	BASICSTACK_ISNULL	N(1)	2779D			
		PROTOTYPE				
*RNF7031	BASICSTACK_NEW	*(16)	2694D			
		PROTOTYPE				
*RNF7031	BASICSTACK_NULL	*(16)	2772D			
		PROTOTYPE				
*RNF7031	BASICSTACK_PEEK	*(16)	2710D			
		PROTOTYPE				
*RNF7031	BASICSTACK_POP	*(16)	2720D			
		PROTOTYPE				
*RNF7031	BASICSTACK_PUSH	I(10,0)	2730D			
		PROTOTYPE				
	CARRAY_INDEX	CONST	5832D			
	CARRAY_NONE	CONST	5824D			
	CARRAY_RPG	CONST	5828D			
	CARRAY_VALUE	CONST	5830D			
	CARRAY_XML	CONST	5826D			
	CATTR_ID	CONST	6172D			
	CAUTH_TYPE_HTTP	CONST	6964D			
	CAUTH_TYPE_PROXY	CONST	6966D			
	CBASICLIST_REMOVE_ALL...					
		CONST	2388D			
	CBASICLIST_TYPE_CHARACTER...					
		CONST	2395D			
	CBASICLIST_TYPE_INTEGER...					
		CONST	2397D			
	CBASICLIST_TYPE_POINTER...					
		CONST	2401D			
	CBASICLIST_TYPE_UIINTEGER...					
		CONST	2399D			
	CBASICLIST_TYPE_UNKNOWN...					
		CONST	2391D			
	CBASICLIST_TYPE_VSTRING...					
		CONST	2393D			
	CBOOL_AND	CONST	6068D			
	CCMD_PRC_CHK_CL_STMT...					
		CONST	2121D			
	CCMD_PRC_CHK_CMDLINE...					
		CONST	2119D			
	CCMD_PRC_CHK_QCMDCHK...					
		CONST	2115D			
	CCMD_PRC_CHK_SEU_BND...					
		CONST	2127D			
	CCMD_PRC_CHK_SEU_CL...					
		CONST	2123D			
	CCMD_PRC_CHK_SEU_CLLE...					
		CONST	2131D			
	CCMD_PRC_CHK_SEU_CMD...					
		CONST	2125D			
	CCMD_PRC_CHK_UDO	CONST	2129D			
	CCMD_PRC_EXC_CMDLINE...					
		CONST	2117D			
	CCMD_PRC_EXC_QCMDEXC...					
		CONST	2113D			
	CCMD_PRC_PROMPT_ALWAYS...					
		CONST	2136D			

CCMD_PRC_PROMPT_HELP...	CONST	2140D			
CCMD_PRC_PROMPT_NEVER...	CONST	2134D			
CCMD_PRC_PROMPT_ON_REQUEST...	CONST	2138D			
CCONTINUE_LITERAL	CONST	6040D			
CCONTINUE_NAME	CONST	6042D			
CCONTINUE_NEXT_LINE...	CONST	6044D			
CCRC32_FINAL	CONST	1633D			
CCRC32_FIRST	CONST	1629D			
CCRC32_MIDDLE	CONST	1631D			
CCRC32_ONLY	CONST	1627D			
CDECIMAL_DECPOS	CONST	5609D			
CDECIMAL_LEN	CONST	5607D			
CDECIMAL_TYPE	CONST	5605D			
CDEFAULT_LEFT_MARGIN...	CONST	6035D			
CDEFAULT_RIGHT_MARGIN...	CONST	6037D			
CDEFAULT_TAB_INDENT...	CONST	6033D			
CDIM_DIGITS	CONST	5596D			
CDIM_LENGTH	CONST	5595D			
CDIM_XML_UNBOUNDED...	CONST	5598D			
CDISPLAY_MODE_DS3	CONST	2316D			
CDISPLAY_MODE_DS4	CONST	2318D			
CDOUBLE_DECPOS	CONST	5620D			
CDOUBLE_LEN	CONST	5618D			
CDOUBLE_QUOTE	CONST	6163D			
CDOUBLE_TYPE	CONST	5616D			
CEBCDIC_CRLF	CONST	6165D	7777	7779	
CEOL	CONST	6046D			
CFALSE	CONST	340D	7182	7229	7989
		8017			
CFLOAT_DECPOS	CONST	5631D			
CFLOAT_LEN	CONST	5629D			
CFLOAT_TYPE	CONST	5627D			
CGENERATOR_CALLBACK_END...	CONST	6084D			
CGENERATOR_CALLBACK_START...	CONST	6082D			
CGENERATOR_CMD_GENERATE...	CONST	6087D			
CGENERATOR_CMD_GENERATE_ARRAY_ITERATOR...	CONST	6115D			
CGENERATOR_CMD_GENERATE_CALLBACKS...	CONST	6107D			
CGENERATOR_CMD_GENERATE_DIMENSIONS...	CONST	6113D			
CGENERATOR_CMD_GENERATE_END_OF_PROCEDURE...	CONST	6105D			
CGENERATOR_CMD_GENERATE_NAMESPACES...	CONST	6109D			
CGENERATOR_CMD_GENERATE_OPERATION_PROCEDURE_INTERFACE...	CONST	6099D			

CGENERATOR_CMD_GENERATE_OPERATION_PROCEDURE_PROTOTYPE...	CONST	6089D
CGENERATOR_CMD_GENERATE_OPERATION_REQUEST_PARAMETERS...	CONST	6091D
CGENERATOR_CMD_GENERATE_OPERATION_RETURN_VALUE...	CONST	6101D
CGENERATOR_CMD_GENERATE_OPERATION_USERDATA...	CONST	6103D
CGENERATOR_CMD_GENERATE_PROTOTYPES...	CONST	6095D
CGENERATOR_CMD_GENERATE_REQUEST...	CONST	6111D
CGENERATOR_CMD_GENERATE_TYPES...	CONST	6093D
CGENERATOR_CMD_GENERATE_WEB_SERVICE_ENDPOINT...	CONST	6097D
CGENERATOR_CMD_VARIABLE...	CONST	6118D
CGENERATOR_CMD_VARIABLE_ADDR_USERDATA...	CONST	6136D
CGENERATOR_CMD_VARIABLE_CLEAR_RTNVALUE...	CONST	6138D
CGENERATOR_CMD_VARIABLE_END_CALLBACK...	CONST	6132D
CGENERATOR_CMD_VARIABLE_MEMBER_TEXT...	CONST	6126D
CGENERATOR_CMD_VARIABLE_OPERATION_NAME...	CONST	6124D
CGENERATOR_CMD_VARIABLE_OPERATION_RESPONSE...	CONST	6128D
CGENERATOR_CMD_VARIABLE_OUTPUT_MEMBER...	CONST	6120D
CGENERATOR_CMD_VARIABLE_PORT_NAME...	CONST	6122D
CGENERATOR_CMD_VARIABLE_RTNVALUE...	CONST	6142D
CGENERATOR_CMD_VARIABLE_RTNVALUE_USERDATA...	CONST	6140D
CGENERATOR_CMD_VARIABLE_SOAP_ACTION...	CONST	6134D
CGENERATOR_CMD_VARIABLE_START_CALLBACK...	CONST	6130D
CGENERATOR_MOVED_TYPE_NAME_SUFFIX...	CONST	6149D
CGENERATOR_REFERENCE_NAME_SUFFIX...	CONST	6151D
CGENERATOR_RENAME_TYPE_NAME_SUFFIX...	CONST	6147D
CGENERATOR_RPC_RESPONSE_NAME_SUFFIX...	CONST	6153D
CGENERATOR_RPG_ARRAY_PREFIX...	CONST	6145D
CGENERATOR_RPG_ARRAY_VALUE_NAME_1...	CONST	6155D
CGENERATOR_RPG_ARRAY_VALUE_NAME_2...	CONST	6157D
CGENERATOR_TAG_CLOSE_BRACKET...	CONST	6075D

CGENERATOR_TAG_FIRST_DELIMITER...	CONST	6077D	
CGENERATOR_TAG_NEXT_DELIMITER...	CONST	6079D	
CGENERATOR_TAG_OPEN_BRACKET...	CONST	6073D	
CGENERATOR_TAG_START...	CONST	6071D	
CHTTP_HEADER_CONTENT_TYPE...	CONST	300D	7776
CHTTP_HEADER_SOAP_ACTION...	CONST	302D	7778
CINTEGER_DECPOS	CONST	5653D	
CINTEGER_LEN	CONST	5651D	
CINTEGER_TYPE	CONST	5649D	
CKEYCLEAR	CONST	6870D	
CKEYDELETE	CONST	6871D	
CKEYENTER	CONST	6873D	
CKEYF1	CONST	6845D	
CKEYF10	CONST	6854D	
CKEYF11	CONST	6855D	
CKEYF12	CONST	6856D	
CKEYF13	CONST	6857D	
CKEYF14	CONST	6858D	
CKEYF15	CONST	6859D	
CKEYF16	CONST	6860D	
CKEYF17	CONST	6861D	
CKEYF18	CONST	6862D	
CKEYF19	CONST	6863D	
CKEYF2	CONST	6846D	
CKEYF20	CONST	6864D	
CKEYF21	CONST	6865D	
CKEYF22	CONST	6866D	
CKEYF23	CONST	6867D	
CKEYF24	CONST	6868D	
CKEYF3	CONST	6847D	
CKEYF4	CONST	6848D	
CKEYF5	CONST	6849D	
CKEYF6	CONST	6850D	
CKEYF7	CONST	6851D	
CKEYF8	CONST	6852D	
CKEYF9	CONST	6853D	
CKEYHELP	CONST	6875D	
CKEYLIGHTPEN	CONST	6882D	
CKEYPAGEDOWN	CONST	6876D	
CKEYPAGEUP	CONST	6877D	
CKEYPRINTSCR	CONST	6878D	
CKEYRESET	CONST	6880D	
CLOGGER	CONST	5483D	
CLOGGER_APPLICATION_MESSAGE...	CONST	5511D	
CLOGGER_ATTACHMENT_PART...	CONST	5505D	
CLOGGER_COMMON	CONST	5517D	
CLOGGER_EXPAT_ADAPTER...	CONST	5520D	
CLOGGER_MANAGED_MEMORY_DATA_SOURCE...	CONST	5501D	

CLOGGER_MANAGED_MEMORY_DATA_SOURCE_ICONV...	CONST	5503D	
CLOGGER_MIME_HEADERS...	CONST	5497D	
CLOGGER_NAMESPACES...	CONST	5485D	
CLOGGER_OPERATION	CONST	5491D	
CLOGGER_QUALIFIED_NAME...	CONST	5513D	
CLOGGER_RPG_ADAPTER...	CONST	5487D	
CLOGGER_RPG_ARRAY	CONST	5495D	
CLOGGER_SOAP_MESSAGE...	CONST	5507D	
CLOGGER_SOAP_MESSAGE_DATA...	CONST	5509D	
CLOGGER_STREAM	CONST	5499D	
CLOGGER_TEMPFILE	CONST	5515D	
CLOGGER_TYPES	CONST	5489D	
CLOGGER_XML_ARRAY	CONST	5493D	
CLONG_DECPOS	CONST	5664D	
CLONG_LEN	CONST	5662D	
CLONG_TYPE	CONST	5660D	
CMAK_KWDS	CONST	5932	5934D
CMAK_OPERATIONS	CONST	5548	5551D
CMD5_FINAL	CONST	1601D	
CMD5_FIRST	CONST	1597D	
CMD5_HEX	CONST	1591D	
CMD5_MIDDLE	CONST	1599D	
CMD5_ONLY	CONST	1595D	
CMD5_STRING	CONST	1593D	
CMEM_SIZE_BYTE	CONST	2023D	
CMEM_SIZE_EXA_BYTE...	CONST	2035D	
CMEM_SIZE_GIGA_BYTE...	CONST	2029D	
CMEM_SIZE_KILO_BYTE...	CONST	2025D	
CMEM_SIZE_MEGA_BYTE...	CONST	2027D	
CMEM_SIZE_PETA_BYTE...	CONST	2033D	
CMEM_SIZE_TERA_BYTE...	CONST	2031D	
CMEM_SIZE_ZETTA_BYTE...	CONST	2037D	
CMSG_ACT_OLD	CONST	728D	
CMSG_ACT_RMV	CONST	729D	
CMSG_ACT_SAME	CONST	730D	7416
CMSG_ALLINACT	CONST	698D	
CMSG_ANY	CONST	710D	7416
CMSG_COMP	CONST	711D	
CMSG_COPY	CONST	712D	
CMSG_DIAG	CONST	713D	
CMSG_ESCAPE	CONST	714D	
CMSG_ESCP	CONST	715D	
CMSG_EXT	CONST	699D	
CMSG_FIRST	CONST	716D	

5761WDS	V6R1M0	080215 RN	IBM ILE RPG	KARLW/WSTST1	S10DCEDB	08/19/09 13:07:17	Page	173
		CMSG_ID_NONE	CONST	694D				
		CMSG_ID_NULL	CONST	695D				
		CMSG_ID_OK	CONST	693D				
		CMSG_INFO	CONST	717D				
		CMSG_INQ	CONST	718D				
		CMSG_LAST	CONST	719D				
		CMSG_NEXT	CONST	720D				
		CMSG_NOTIFY	CONST	721D				
		CMSG_PREVIOUS	CONST	722D				
		CMSG_PRV	CONST	700D				
		CMSG_RMV_ALL	CONST	733D				
		CMSG_RMV_BYKEY	CONST	734D				
		CMSG_RMV_NEW	CONST	735D				
		CMSG_RMV_OLD	CONST	736D				
		CMSG_RMV_SCOPE	CONST	737D				
		CMSG_RPY	CONST	723D				
		CMSG_RQS	CONST	724D				
		CMSG_RTNCTRLCHAR_NO...						
			CONST	743D				
		CMSG_RTNCTRLCHAR_YES...						
			CONST	741D				
		CMSG_SAME	CONST	701D	7417			
		CMSG_STATUS	CONST	725D				
		CMSG_TOCTLBDY	CONST	706D				
		CMSG_TOMYSELF	CONST	704D	7417			
		CMSG_TOPGM	CONST	707D				
		CMSG_TOPGMBDY	CONST	705D				
		CNEG_INT_DECPOS	CONST	5730D				
		CNEG_INT_LEN	CONST	5728D				
		CNEG_INT_TYPE	CONST	5726D				
		CNON_NEG_INT_DECPOS...						
			CONST	5708D				
		CNON_NEG_INT_LEN	CONST	5706D				
		CNON_NEG_INT_TYPE	CONST	5704D				
		CNON_POS_INT_DECPOS...						
			CONST	5719D				
		CNON_POS_INT_LEN	CONST	5717D				
		CNON_POS_INT_TYPE	CONST	5715D				
		CNS_APACHE_SOAP	CONST	6004D				
		CNS_SCHEMA	CONST	6006D				
		CNS_SCHEMA_INSTANCE...						
			CONST	6008D				
		CNS_WSDL	CONST	5990D				
		CNS_WSDL_HTTP	CONST	6001D				
		CNS_WSDL_SOAP	CONST	5992D				
		CNS_WSDL_SOAP_ENCODING...						
			CONST	5998D				
		CNS_WSDL_SOAP_ENVELOPE...						
			CONST	5995D				
		CNS_WSDL2RPG	CONST	6015D				
*RNF7031		COMSTR	PROTOTYPE	1535D				
*RNF7031		CONVERT	*(16) PROCPTR	3351D				
		CPOS_INT_DECPOS	CONST	5741D				
		CPOS_INT_LEN	CONST	5739D				
		CPOS_INT_TYPE	CONST	5737D				
		CPREFIX_WSDL2RPG	CONST	6013D				
*RNF7031		CRC32_T	DS(42)	1635D	1694			
*RNF7031		LENGTH	U(5,0)	1636D				

5761WDS	V6R1M0	080215	RN	IBM ILE RPG	KARLW/WSTST1	S10DCEDB	08/19/09 13:07:17	Page	174
---------	--------	--------	----	-------------	--------------	----------	-------------------	------	-----

*RNF7031	VALUE	A(40)	1637D	
	REFERENCE_FIELD_SUFFIX...			
	CONST		6031D	
	CSFLDROP	CONST	6885D	
	CSFLFOLD	CONST	6884D	
	CSHA1_FINAL	CONST	1619D	
	CSHA1_FIRST	CONST	1615D	
	CSHA1_HEX	CONST	1609D	
	CSHA1_MIDDLE	CONST	1617D	
	CSHA1_ONLY	CONST	1613D	
	CSHA1_STRING	CONST	1611D	
	CSHORT_DECPOS	CONST	5642D	
	CSHORT_LEN	CONST	5640D	
	CSHORT_TYPE	CONST	5638D	
	CSINGLE_QUOTE	CONST	6161D	
	CSPEC_ENDFOR	CONST	6063D	
	CSPEC_ENDIF	CONST	6051D	
	CSPEC_ENDSL	CONST	6059D	
	CSPEC_FOR	CONST	6061D	
	CSPEC_IF	CONST	6049D	
	CSPEC_OTHER	CONST	6055D	
	CSPEC_RETURN	CONST	6065D	
	CSPEC_SELECT	CONST	6057D	
	CSPEC_WHEN	CONST	6053D	
	CTRUE	CONST	335D	7415
	CTYPE_REFERENCE	CONST	5834D	
	CUINTEGER_DECPOS	CONST	5686D	
	CUINTEGER_LEN	CONST	5684D	
	CUINTEGER_TYPE	CONST	5682D	
	CULONG_DECPOS	CONST	5697D	
	CULONG_LEN	CONST	5695D	
	CULONG_TYPE	CONST	5693D	
	CURL	CONST	7179	7201D
	CURL_ERROR_EMPTY_URL...			
	CONST		3053D	
	CURL_ERROR_INVALID_OBJECT_DATA...			
	CONST		3063D	
	CURL_ERROR_INVALID_PORT_NUMBER...			
	CONST		3059D	
	CURL_ERROR_NO_PROTOCOL...			
	CONST		3055D	
	CURL_ERROR_UNSUPPORTED_OPERATION...			
	CONST		3061D	
	CURL_ERROR_UNSUPPORTED_PROTOCOL...			
	CONST		3057D	
	CUSHORT_DECPOS	CONST	5675D	
	CUSHORT_LEN	CONST	5673D	
	CUSHORT_TYPE	CONST	5671D	
	CUUID_HEX	CONST	1581D	
	CUUID_STRING	CONST	1583D	
	CWAIT_MOD_AS_ENTER...			
	CONST		1548D	
	CWAIT_MOD_AS_LEAVE...			
	CONST		1550D	
	CWAIT_MPL_SET	CONST	1552D	
	CWAIT_NORMAL	CONST	1546D	
	CWAIT_SIGNALS	CONST	1554D	
	CWORDWRAP_PARAGRAPH_MSGD...			

5761WDS	V6R1M0	080215	RN	IBM ILE RPG	KARLW/WSTST1	S10DCEDB	08/19/09 13:07:17	Page	175
				CONST	1715D				
				CWORDWRAP_PARAGRAPH_NONE...					
				CONST	1713D				
				CWORDWRAP_PARAGRAPH_TEXT...					
				CONST	1717D				
				CWORDWRAP_PARAGRAPH_TRIMR...					
				CONST	1719D				
				CWSDL_ATTR_XMLNS	CONST	5986D			
				CWSDL_ERROR	CONST	5976D			
				CWSDL_OK	CONST	5977D			
				CWSDL_STYLE_DOCUMENT...					
				CONST	6021D				
				CWSDL_STYLE_RPC	CONST	6019D			
				CWSDL_USE_ENCODED	CONST	6027D			
				CWSDL_USE_LITERAL	CONST	6025D			
				CWSDL_XPATH_DEFINITIONS...					
				CONST	5982D				
				C5250_BLU	CONST	6946D			
				C5250_BLUR	CONST	6948D			
				C5250_BLUU	CONST	6950D			
				C5250_GRN	CONST	6890D			
				C5250_GRNR	CONST	6892D			
				C5250_GRNU	CONST	6894D			
				C5250_GRNUR	CONST	6896D			
				C5250_ND1	CONST	6953D			
				C5250_ND2	CONST	6955D			
				C5250_ND3	CONST	6957D			
				C5250_ND4	CONST	6959D			
				C5250_PNK	CONST	6937D			
				C5250_PNKR	CONST	6939D			
				C5250_PNKU	CONST	6941D			
				C5250_PNKUR	CONST	6943D			
				C5250_RED	CONST	6906D			
				C5250_REDB	CONST	6910D			
				C5250_REDR	CONST	6908D			
				C5250_REDRB	CONST	6912D			
				C5250_REDUR	CONST	6914D			
				C5250_REDUB	CONST	6918D			
				C5250_REDUR	CONST	6916D			
				C5250_TRQC	CONST	6921D			
				C5250_TRQCR	CONST	6923D			
				C5250_TRQU	CONST	6925D			
				C5250_TRQUR	CONST	6927D			
				C5250_WHT	CONST	6899D			
				C5250_WHTR	CONST	6901D			
				C5250_WHTU	CONST	6903D			
				C5250_YLWC	CONST	6930D			
				C5250_YLWCR	CONST	6932D			
				C5250_YLWU	CONST	6934D			
*RNF7031	DATA			*(16)	3350D				
	DIM_A1			CONST	53D	67			
	DIM_A2			CONST	54D	75			
*RNF7031	DISPLAY_GETCOLUMNS...								
				I(5,0)	2352D				
				PROTOTYPE					
*RNF7031	DISPLAY_GETMODE			A(4)	2322D				
				PROTOTYPE					
*RNF7031	DISPLAY_GETROWS			I(5,0)	2343D				

5761	WDS	V6R1M0	080215	RN		IBM ILE RPG	KARLW/WSTST1	S10DCEDB	08/19/09 13:07:17	Page	178
						PROTOTYPE					
*RNF7031	F_GETENVVARBYINDEX...					DS(1156)	2187D				
						PROTOTYPE					
*RNF7031	F_GETLENGTHEDITCODE...					U(5,0)	1852D				
						PROTOTYPE					
*RNF7031	F_GETLENGTHEDITWORD...					U(5,0)	1862D				
						PROTOTYPE					
*RNF7031	F_GETNUMENVVAR					I(10,0)	2180D				
						PROTOTYPE					
*RNF7031	F_GETOBJLIB					A(10)	1325D				
						PROTOTYPE					
*RNF7031	F_GETPROCPTRBYNAM...					DS(298)	1314D				
						PROTOTYPE					
*RNF7031	F_GETTMPNAME					A(20)	1180D				
						PROTOTYPE					
*RNF7031	F_GETUNIQUENAME					A(20)	1187D				
						PROTOTYPE					
*RNF7031	F_GETUSRSPCAUTOEXT...					N(1)	1234D				
						PROTOTYPE					
*RNF7031	F_GETUSRSPCINITVAL...					A(1)	1245D				
						VARYING					
						PROTOTYPE					
*RNF7031	F_GETUSRSPCSIZE					I(10,0)	1222D				
						PROTOTYPE					
*RNF7031	F_ISAPIERROR					N(1)	923D				
						PROTOTYPE					
*RNF7031	F_ISBITON					N(1)	1641D				
						PROTOTYPE					
*RNF7031	F_LCASE					A(4096)	1768D				
						VARYING					
						PROTOTYPE					
*RNF7031	F_LEFT					A(4096)	1723D				
						VARYING					
						PROTOTYPE					
*RNF7031	F_LOG10					F(8)	1650D				
						PROTOTYPE					
*RNF7031	F_NEWAPIERRCODE					DS(272)	915D				
						PROTOTYPE					
*RNF7031	F_NEWMSG					DS(549)	756D				
						PROTOTYPE					
*RNF7031	F_OBJNAME					A(21)	1987D				
						VARYING					
						PROTOTYPE					
*RNF7031	F_PARSEDATE					D(10*ISO-)	2014D				
						PROTOTYPE					
	F_RCVPGMMSG					DS(4272)	787D	7416			
						PROTOTYPE					
*RNF7031	F_RCVPGMMSGOBJ					DS(549)	875D				
						PROTOTYPE					
*RNF7031	F_REPLACE					A(4096)	1785D				
						VARYING					
						PROTOTYPE					

5761	WDS	V6R1M0	080215	RN		IBM	ILE	RPG		KARLW/WSTST1		S10DCEDB	08/19/09	13:07:17	Page	179
*RNF7031	F_RIGHT	A(4096)			1741D											
		VARYING														
		PROTOTYPE														
*RNF7031	F_RMVKEYWORD	A(4096)			1871D											
		VARYING														
		PROTOTYPE														
*RNF7031	F_RMVQUOTES	A(4096)			1880D											
		VARYING														
		PROTOTYPE														
*RNF7031	F_RPLUNPRTCHARS	A(4096)			1996D											
		VARYING														
		PROTOTYPE														
*RNF7031	F_RTVMBRD	A(1024)			1167D											
		PROTOTYPE														
*RNF7031	F_RTVMMSGHELPTTEXT	A(4096)			831D											
		VARYING														
		PROTOTYPE														
*RNF7031	F_RTVMMSGHELPTTEXTRAW...	A(4096)			854D											
		VARYING														
		PROTOTYPE														
	F_RTVMMSGTEXT	A(4096)			820D	7420										
		VARYING														
		PROTOTYPE														
*RNF7031	F_RTVMMSGTEXTRAW	A(4096)			843D											
		VARYING														
		PROTOTYPE														
*RNF7031	F_RTVOBJD	A(1024)			1154D											
		PROTOTYPE														
*RNF7031	F_RTVUSRSPCPTR	*(16)			1211D											
		PROTOTYPE														
*RNF7031	F_SCANR	I(5,0)			1936D											
		PROTOTYPE														
*RNF7031	F_SETENVVAR	N(1)			2163D											
		PROTOTYPE														
*RNF7031	F_SNDPGMMSG	A(4)			773D											
		PROTOTYPE														
*RNF7031	F_SPACE	A(4096)			1759D											
		VARYING														
		PROTOTYPE														
*RNF7031	F_STARTSWITH	N(1)			1732D											
		PROTOTYPE														
*RNF7031	F_STRERROR	A(128)			953D											
		VARYING														
		PROTOTYPE														
*RNF7031	F_THIS	A(256)			991D											
		VARYING														
		PROTOTYPE														
*RNF7031	F_UCASE	A(4096)			1776D											
		VARYING														
		PROTOTYPE														
*RNF7031	F_VAL	F(8)			1804D											
		PROTOTYPE														
*RNF7031	F_WORDWRAP	A(4096)			1816D											
		VARYING														
		PROTOTYPE														
	G_GETLOGINDATA	*(16) PROCPTR			7189D	7365	8039M									
	G_HTTPDEBUG	DS(503)			7169D	7264	7265	7266								

	FILENAME	A(500) VARYING	8160M 7171D 8165	8163M 7265	8165M 7266	8163
	STATUS	N(1)	7170D	7264	8160	
	G_ISAXIS_2095	N(1)	7182D	7737	7742	8135M
	G_PROXY	DS(260)	7174D	7275	7276	7276
	PORT	I(10,0)	8188M 7176D	8189M 7276	8189	
	SERVER	A(256)	7175D	7275	7276	8188
	G_SOAPFAULT	DS(908)	7185D	7329	7419M	7420M
			7421	7422	7426M	7427
			7428M	7429	7430	7811
			8059	8077	8095	8113
	FAULTCODE	A(256) VARYING	8095			
	FAULTSTRING	A(256) VARYING	8113			
	HTTPERROR	DS(134)	7419	7420	7421	7422
			7426	7427	7428	7429
			7430	8059	8077	
	NO	I(10,0)	7419	7421	7426	7427
			8059			
	TEXT	A(128) VARYING	7420	7422	7428	7429
			7430	8077		
	G_URL	A(32767) VARYING	7179D	7350	8209	8228M
			8250	8273	8275M	8297
			8320	8322M		
	GETLOGINDATA	I(10,0) PROTOTYPE	6970D	7379		
*RNF7031	HANDLE_DELETE	PROTOTYPE	2267D			
*RNF7031	HANDLE_GETPOINTER	*(16) PROTOTYPE	2259D			
*RNF7031	HANDLE_HANDLE_T	I(10,0) BASED(PDUMMY)	2220D	2226	2235	2246
*RNF7031	HANDLE_ISNULL	N(1) PROTOTYPE	2254	2263	2271	
*RNF7031	HANDLE_ISTYPEOF	N(1) PROTOTYPE	2242D			
*RNF7031	HANDLE_NEW	I(10,0) PROTOTYPE	2250D			
*RNF7031	HANDLE_NULL	I(10,0) PROTOTYPE	2226D			
*RNF7031	HANDLE_TYPE_T	A(20) BASED(PDUMMY)	2235D			
*RNF7031	HEADERS_T	DS(1538) BASED(PDUMMY)	2222D	2230	2255	
*RNF7031	CONTENTTYPE	A(1024) VARYING	305D	7234	7772	
*RNF7031	SOAPACTION	A(510) VARYING	306D			
	HTTP_ACCES	CONST	307D			
	HTTP_ASCII	CONST	5268D			
	HTTP_ATHTYP	CONST	3732D			
	HTTP_ATHVAL	CONST	5300D			
	HTTP_AUTH_BASIC	CONST	5302D			
	HTTP_AUTH_MD5_DIGEST...	CONST	5386D	7392	7400	
	HTTP_AUTH_NONE	CONST	5388D	7389	7397	
			5384D			

5761WDS	V6R1M0	080215 RN	IBM ILE RPG	KARLW/WSTST1	S10DCEDB	08/19/09 13:07:17	Page	181
	HTTP_BADCHK	CONST	5262D					
	HTTP_BADCNN	CONST	5235D					
	HTTP_BADPNT	CONST	5321D					
	HTTP_BADURL	CONST	5225D					
	HTTP_BRRECV	CONST	5315D					
	HTTP_BRSELE	CONST	5317D					
	HTTP_BRTIME	CONST	5313D					
*RNF7031	HTTP_BUILD_SOCKADDR...							
		I(10,0)	3995D					
		PROTOTYPE						
	HTTP_BWSELE	CONST	5339D					
	HTTP_BWSEND	CONST	5337D					
	HTTP_BWTIME	CONST	5341D					
	HTTP_BWTIMO	CONST	5353D					
	HTTP_CCSID	CONST	3745D					
	HTTP_CHKSIZ	CONST	5264D					
	HTTP_CKDUMP	CONST	5367D					
	HTTP_CKOPEN	CONST	5369D					
	HTTP_CKSTAT	CONST	5371D					
*RNF7031	HTTP_CLOSE	I(10,0)	4009D					
		PROTOTYPE						
	HTTP_CNNTIMO	CONST	5237D					
*RNF7031	HTTP_COMP	PROTOTYPE	5153D					
	HTTP_CONTTYPE	CONST	3713D					
	HTTP_CONVERR	CONST	5373D					
	HTTP_COOKIE_DEFAULT...							
		CONST	3775D					
*RNF7031	HTTP_COOKIE_FILE	PROTOTYPE	5144D					
*RNF7031	HTTP_CRASH	PROTOTYPE	5170D					
	HTTP_CSSELE	CONST	5294D					
	HTTP_DEBUG	PROTOTYPE	4923D	7266M	7268M	7271M		
*RNF7031	HTTP_DEBUG_FILE	A(500)	3781D					
		VARYING						
*RNF7031	HTTP_DIAG	PROTOTYPE	5162D					
	HTTP_EBCDIC	CONST	3731D	7283				
	HTTP_ERROR	A(80)	4021D	7363M	7428			
		PROTOTYPE						
	HTTP_FDOPEN	CONST	5270D					
	HTTP_FDSTAT	CONST	5357D					
*RNF7031	HTTP_GET	I(10,0)	3830D					
		PROTOTYPE						
*RNF7031	HTTP_GET_XML	I(10,0)	4467D					
		PROTOTYPE						
*RNF7031	HTTP_GET_XMLTF	I(10,0)	4506D					
		PROTOTYPE						
	HTTP_GETAUTH	I(10,0)	4098D	7372M				
		PROTOTYPE						
	HTTP_GSKAPPID	CONST	5272D					
	HTTP_GSKATYP	CONST	5274D					
	HTTP_GSKENVI	CONST	5276D					
	HTTP_GSKENVO	CONST	5278D					
	HTTP_GSKKEYF	CONST	5363D					
	HTTP_GSKSTYP	CONST	5280D					
*RNF7031	HTTP_HEADER	A(32500)	5113D					
		VARYING						
		PROTOTYPE						
	HTTP_HOSTNF	CONST	5227D					
	HTTP_IFOPEN	CONST	5361D					

5761WDS	V6R1M0	080215	RN	IBM	ILE	RPG	KARLW/WSTST1	S10DCEDB	08/19/09	13:07:17	Page	182
		HTTP_IFSMODE	CONST									
		HTTP_INVPTR	CONST									
*RNF7031		HTTP_LONG_PARSEURL...										
		I(10,0)										
		PROTOTYPE										
		HTTP_LRRECV	CONST									
		HTTP_LRSELE	CONST									
		HTTP_LRTIME	CONST									
*RNF7031		HTTP_MFD_ENCODER_ADDSTMF...										
		N(1)										
		PROTOTYPE										
*RNF7031		HTTP_MFD_ENCODER_ADDVAR...										
		N(1)										
		PROTOTYPE										
*RNF7031		HTTP_MFD_ENCODER_ADDVAR_S...										
		N(1)										
		PROTOTYPE										
*RNF7031		HTTP_MFD_ENCODER_CLOSE...										
		PROTOTYPE										
*RNF7031		HTTP_MFD_ENCODER_OPEN...										
		*(16)										
		PROTOTYPE										
		HTTP_MUTABLE	CONST									
		HTTP_NDAUTH	CONST				7345	7364	7371			
		HTTP_NOAUTH	CONST									
		HTTP_NOCDRIV	CONST									
		HTTP_NODATA	CONST									
		HTTP_NOMEM	CONST									
		HTTP_NONBLK	CONST									
		HTTP_NOSPAC	CONST									
		HTTP_NOTREG	CONST									
		HTTP_NOTSUPP	CONST									
*RNF7031		HTTP_PARSE_XML_STMF...										
		I(10,0)										
		PROTOTYPE										
*RNF7030		HTTP_PARSER_GET_END_CB...										
		UNDEF					7488	7596	7691			
*RNF7030		HTTP_PARSER_GET_START_CB...										
		UNDEF					7487	7595	7690			
*RNF7030		HTTP_PARSER_GET_USERDATA...										
		UNDEF					7482	7590	7685			
*RNF7030		HTTP_PARSER_SWITCH_CB...										
		UNDEF					7499M	7607M	7810M	7817M		
							7824M	7862M				
*RNF7031		HTTP_PARSEURL	I(10,0)				3972D					
		PROTOTYPE										
*RNF7031		HTTP_PERSIST_CLOSE...										
		I(10,0)					4818D					
		PROTOTYPE										
*RNF7031		HTTP_PERSIST_GET	I(10,0)				4744D					
		PROTOTYPE										
*RNF7031		HTTP_PERSIST_OPEN	*(16)				4712D					
		PROTOTYPE										
*RNF7031		HTTP_PERSIST_POST	I(10,0)				4795D					
		PROTOTYPE										
		HTTP_POINT_ADDL_HEADER...										
		CONST					5438D	7335				

5761WDS	V6R1M0	080215 RN	IBM ILE RPG	KARLW/WSTST1	S10DCEDB	08/19/09 13:07:17	Page	183
		HTTP_POINT_DEBUG	CONST	5403D				
		HTTP_POINT_DOWNLOAD_STATUS...						
			CONST	5427D				
		HTTP_POINT_PARSE_HDR_LONG...						
			CONST	5458D				
		HTTP_POINT_PARSE_HEADER...						
			CONST	5448D				
		HTTP_POINT_UPLOAD_STATUS...						
			CONST	5415D				
*RNF7031	HTTP_POST		I(10,0)	3870D				
			PROTOTYPE					
*RNF7031	HTTP_POST_STMF		I(10,0)	4416D				
			PROTOTYPE					
*RNF7031	HTTP_POST_STMF_XML...							
			I(10,0)	4645D				
			PROTOTYPE					
*RNF7031	HTTP_POST_STMF_XMLTF...							
			I(10,0)	4689D				
			PROTOTYPE					
	HTTP_POST_XML		I(10,0)	4561D	7350			
			PROTOTYPE					
*RNF7031	HTTP_POST_XMLTF		I(10,0)	4605D				
			PROTOTYPE					
	HTTP_PROXY_GETAUTH...							
			I(10,0)	4169D	7375M			
			PROTOTYPE					
	HTTP_PROXY_SETAUTH...							
			I(10,0)	4144D	7397M	7400M		
			PROTOTYPE					
	HTTP_PST417		CONST	5307D				
	HTTP_PXNDAUTH		CONST	5377D	7345	7364		
	HTTP_RDCRLF		CONST	5319D				
	HTTP_RDRECV		CONST	5253D				
	HTTP_RDSELE		CONST	5255D				
	HTTP_RDTIME		CONST	5311D				
	HTTP_RDWERR		CONST	5258D				
*RNF7031	HTTP_REDIR_LOC		A(1024)	4200D				
			VARYING					
			PROTOTYPE					
	HTTP_REGERR		CONST	5282D				
	HTTP_RESP		CONST	5251D				
	HTTP_RRRECV		CONST	5247D				
	HTTP_RRSELE		CONST	5249D				
	HTTP_RRTIME		CONST	5309D				
	HTTP_SAALLO		CONST	5229D				
	HTTP_SDSELE		CONST	5292D				
	HTTP_SDSSEND		CONST	5260D				
*RNF7031	HTTP_SELECT_COMMDRIVER...							
			*(16)	4349D				
			PROTOTYPE					
*RNF7031	HTTP_SET_100_TIMEOUT...							
			PROTOTYPE	5053D				
	HTTP_SETATTR		CONST	5375D				
	HTTP_SETAUTH		I(10,0)	4114D	7389M	7392M		
			PROTOTYPE					
	HTTP_SETCCSIDS		I(10,0)	4939D	7283M			
			PROTOTYPE					
	HTTP_SETCONTENTTYPEANDSOAPACTION...							

5761	WDS	V6R1M0	080215	RN	IBM ILE RPG	KARLW/WSTST1	S10DCEDB	08/19/09 13:07:17	Page	184
				PROTOTYPE	6990D	7763	7767	7783		
*RNF7031	HTTP_SETFILECCSID			PROTOTYPE	4978D					
	HTTP_SETPROXY			I(10,0)	4128D	7276M	7278M			
				PROTOTYPE						
*RNF7031	HTTP_SETTABLES			I(10,0)	4955D					
				PROTOTYPE						
	HTTP_SOCERR			CONST	5231D					
	HTTP_SRSELE			CONST	5241D					
	HTTP_SRXLATE			CONST	5239D					
	HTTP_SRXMIT			CONST	5244D					
	HTTP_SSOPEN			CONST	5284D					
	HTTP_SSPROT			CONST	5323D					
	HTTP_SSPSET			CONST	5327D					
	HTTP_SSPUNK			CONST	5325D					
	HTTP_SSSNFD			CONST	5286D					
	HTTP_SSSNTO			CONST	5288D					
	HTTP_SSTIMO			CONST	5290D					
	HTTP_STMF_CALC			CONST	5101D					
	HTTP_SWCERR			CONST	5355D					
*RNF7031	HTTP_TEMPFILE			A(40)	5178D					
				VARYING						
				PROTOTYPE						
	HTTP_TIMEOUT			CONST	3700D	7356				
*RNF7031	HTTP_URL_ENCODER			*(16)	4212D	4214	4233	4251		
					4272	4290	4306			
*RNF7031	HTTP_URL_ENCODER_ADDVAR...									
				N(1)	4232D					
				PROTOTYPE						
*RNF7031	HTTP_URL_ENCODER_ADDVAR_S...									
				N(1)	4305D					
				PROTOTYPE						
*RNF7031	HTTP_URL_ENCODER_FREE...									
				N(1)	4289D					
				PROTOTYPE						
*RNF7031	HTTP_URL_ENCODER_GETPTR...									
				PROTOTYPE	4250D					
*RNF7031	HTTP_URL_ENCODER_GETSTR...									
				A(32767)	4271D					
				VARYING						
				PROTOTYPE						
*RNF7031	HTTP_URL_ENCODER_NEW...									
				*(16)	4214D					
				PROTOTYPE						
*RNF7031	HTTP_URL_GET			I(10,0)	3838D					
				PROTOTYPE						
*RNF7031	HTTP_URL_GET_RAW			I(10,0)	3911D					
				PROTOTYPE						
*RNF7031	HTTP_URL_GET_XML			I(10,0)	4478D					
				PROTOTYPE						
*RNF7031	HTTP_URL_POST			I(10,0)	3879D					
				PROTOTYPE						
*RNF7031	HTTP_URL_POST_RAW			I(10,0)	3943D					
				PROTOTYPE						
*RNF7031	HTTP_URL_POST_RAW2...									
				I(10,0)	4375D					
				PROTOTYPE						
*RNF7031	HTTP_URL_POST_STMF...									
				I(10,0)	4408D					

		PROTOTYPE		
*RNF7031	HTTP_URL_POST_STMF_XML...	I(10,0)	4656D	
		PROTOTYPE		
*RNF7031	HTTP_URL_POST_XML	I(10,0)	4573D	
		PROTOTYPE		
*RNF7031	HTTP_USE_COOKIES	PROTOTYPE	5128D	
	HTTP_USERAGENT	CONST	3706D	7357
	HTTP_XFRENC	CONST	5266D	
*RNF7031	HTTP_XLATE	I(10,0)	4992D	
		PROTOTYPE		
*RNF7031	HTTP_XLATEDYN	I(10,0)	5026D	
		PROTOTYPE		
*RNF7031	HTTP_XLATEP	I(10,0)	5009D	
		PROTOTYPE		
	HTTP_XML_CALC	CONST	5100D	
*RNF7031	HTTP_XML_SETCCSIDS...	I(10,0)	5068D	
		PROTOTYPE		
	HTTP_XMLERR	CONST	5359D	
	HTTP_XMLNS	PROTOTYPE	5187D	7332M
*RNF7031	HTTP_XMLRETURNPTR	PROTOTYPE	5201D	
*RNF7031	HTTP_XMLSTRIPCRLF	PROTOTYPE	5217D	
	HTTP_XPROC	I(10,0)	4188D	7335M
		PROTOTYPE		
	HTTPAPI_RELDATE	CONST	3667D	
	HTTPAPI_VERSION	CONST	3665D	
*RNF7031	HTTPS_CLEANUP	I(10,0)	4073D	
		PROTOTYPE		
*RNF7031	HTTPS_DCM_REG	I(10,0)	4062D	
		PROTOTYPE		
*RNF7031	HTTPS_INIT	I(10,0)	4042D	
		PROTOTYPE		
*RNF7031	KEYPRESSED_T	A(1)	6835D	6979
		BASED(PDUMMY)		
*RNF7031	MAP(256)	I(10,0)	3349D	
*RNF7031	MARSHALLER_TOANY	A(65535)	6537D	
		VARYING		
		PROTOTYPE		
*RNF7031	MARSHALLER_TOANYTYPE...	A(65535)	6529D	
		VARYING		
		PROTOTYPE		
*RNF7031	MARSHALLER_TOANYURI...	A(65535)	6727D	
		VARYING		
		PROTOTYPE		
*RNF7031	MARSHALLER_TOBASE64BINARY...	A(65535)	6700D	
		VARYING		
		PROTOTYPE		
	MARSHALLER_TOBOOLEAN...	A(65535)	6561D	7317
		VARYING		
		PROTOTYPE		
*RNF7031	MARSHALLER_TODATE	A(65535)	6681D	
		VARYING		
		PROTOTYPE		

	MARSHALLER_TODATETIME...			
	A(65535)	6673D	7313	7321
	VARYING			
	PROTOTYPE			
*RNF7031	MARSHALLER_TODECIMAL...			
	A(65535)	6569D		
	VARYING			
	PROTOTYPE			
*RNF7031	MARSHALLER_TODOUBLE...			
	A(65535)	6577D		
	VARYING			
	PROTOTYPE			
*RNF7031	MARSHALLER_TODURATION...			
	A(65535)	6736D		
	VARYING			
	PROTOTYPE			
*RNF7031	MARSHALLER_TOFLOAT...			
	A(65535)	6585D		
	VARYING			
	PROTOTYPE			
	MARSHALLER_TOINT	6593D	7309	
	A(65535)			
	VARYING			
	PROTOTYPE			
*RNF7031	MARSHALLER_TOLANGUAGE...			
	A(65535)	6709D		
	VARYING			
	PROTOTYPE			
*RNF7031	MARSHALLER_TOLONG	6609D		
	A(65535)			
	VARYING			
	PROTOTYPE			
*RNF7031	MARSHALLER_TONEGATIVEINTEGER...			
	A(65535)	6657D		
	VARYING			
	PROTOTYPE			
*RNF7031	MARSHALLER_TONONNEGATIVEINTEGER...			
	A(65535)	6641D		
	VARYING			
	PROTOTYPE			
*RNF7031	MARSHALLER_TONONPOSITIVEINTEGER...			
	A(65535)	6649D		
	VARYING			
	PROTOTYPE			
*RNF7031	MARSHALLER_TOPOSITIVEINTEGER...			
	A(65535)	6665D		
	VARYING			
	PROTOTYPE			
*RNF7031	MARSHALLER_TOSCHEMA...			
	A(65535)	6545D		
	VARYING			
	PROTOTYPE			
*RNF7031	MARSHALLER_TOSHORT...			
	A(65535)	6601D		
	VARYING			
	PROTOTYPE			
	MARSHALLER_TOSTRING...			
	A(65535)	6553D	7300	
	VARYING			
	PROTOTYPE			

*5761WDS V6R1M0 080215 RN		IBM ILE RPG	KARLW/WSTST1	S10DCEDB	08/19/09 13:07:17	Page	187
*RNF7031	MARSHALLER_TOTIME	A(65535) VARYING PROTOTYPE	6690D				
*RNF7031	MARSHALLER_TOTOKEN...	A(65535) VARYING PROTOTYPE	6718D				
*RNF7031	MARSHALLER_TOUNSIGNEDINT...	A(65535) VARYING PROTOTYPE	6617D				
*RNF7031	MARSHALLER_TOUNSIGNEDLONG...	A(65535) VARYING PROTOTYPE	6633D				
*RNF7031	MARSHALLER_TOUNSIGNEDSHORT...	A(65535) VARYING PROTOTYPE	6625D				
*RNF7031	MBRD0100_T	DS(135) BASED(PDUMMY)	1061D				
*RNF7031	ATTR	A(10)	1067D				
*RNF7031	BYTEAVL	I(10,0)	1063D				
*RNF7031	BYTEPRV	I(10,0)	1062D				
*RNF7031	CHGDATTIM	A(13)	1070D				
*RNF7031	CRTDATTIM	A(13)	1069D				
*RNF7031	FILLIB	A(10)	1065D				
*RNF7031	FILNME	A(10)	1064D				
*RNF7031	ISSRCFILE	A(1)	1072D				
*RNF7031	MBR	A(10)	1066D				
*RNF7031	SRCTYP	A(10)	1068D				
*RNF7031	TEXT	A(50)	1071D				
*RNF7031	MBRD0200_T	DS(266) BASED(PDUMMY)	1081D				
*RNF7031	ACCPTHSIZE	I(10,0)	1101D				
*RNF7031	ACCPTHSIZEMULT	I(10,0)	1113D				
*RNF7031	ATTR	A(10)	1087D				
*RNF7031	BYTEAVL	I(10,0)	1083D				
*RNF7031	BYTEPRV	I(10,0)	1082D				
*RNF7031	CHGDATTIM	A(13)	1090D				
*RNF7031	CRTDATTIM	A(13)	1089D				
*RNF7031	DATELASTUSED	A(7)	1109D				
*RNF7031	DTASPCSIZE	I(10,0)	1100D				
*RNF7031	DTASPCSIZEMULT	I(10,0)	1112D				
*RNF7031	FILLIB	A(10)	1085D				
*RNF7031	FILNME	A(10)	1084D				
*RNF7031	ISSRCFILE	A(1)	1092D				
*RNF7031	LENADDINF	I(10,0)	1116D				
*RNF7031	LGLFILE	A(1)	1095D				
*RNF7031	MBR	A(10)	1086D				
*RNF7031	MBRCHGDATTIM	A(13)	1103D				
*RNF7031	MBREXPDATE	A(7)	1106D				
*RNF7031	MBRRSTDATTIM	A(13)	1105D				
*RNF7031	MBRSVDATTIM	A(13)	1104D				
*RNF7031	MBRTEXTCCSID	I(10,0)	1114D				
*RNF7031	NUMDAYSUSED	I(10,0)	1108D				
*RNF7031	NUMDLTRCD	I(10,0)	1099D				
*RNF7031	NUMDLTRCDU	U(10,0)	1118D				

*RNF7031	NUMPHYMBRS	I(10,0)	1102D			
*RNF7031	NUMRCDALLMBR	I(10,0)	1098D			
*RNF7031	NUMRCDALLMBRU	U(10,0)	1117D			
*RNF7031	ODPSHARING	A(1)	1096D			
*RNF7031	OFSADDINF	I(10,0)	1115D			
*RNF7031	RESERVED_1	A(2)	1097D			
*RNF7031	RESERVED_2	A(6)	1107D			
*RNF7031	RESERVED_3	A(2)	1111D			
*RNF7031	RESERVED_4	A(6)	1119D			
*RNF7031	RMTFILE	A(1)	1094D			
*RNF7031	SRCTYP	A(10)	1088D			
*RNF7031	TEXT	A(50)	1091D			
*RNF7031	USERESETDATE	A(7)	1110D			
*RNF7031	MD5_T	DS(34)	1603D	1674		
*RNF7031	LENGTH	U(5,0)	1604D			
*RNF7031	VALUE	A(32)	1605D			
*RNF7031	MEMORY_ALLOCATE	*(16)	2275D			
		PROTOTYPE				
*RNF7031	MEMORY_FREE	PROTOTYPE	2292D			
*RNF7031	MEMORY_FREEALL	PROTOTYPE	2284D			
	MI_LIBRARY	CONST	1362D			
	MI_PROGRAM	CONST	1361D			
*RNF7031	MI_TIME_T	A(8)	1541D	1558	1563	
	MI_USRIDX	CONST	1365D			
	MI_USRQ	CONST	1364D			
	MI_USRSPC	CONST	1363D			
*RNF7031	MITIME	*(16)	1562D			
		PROTOTYPE				
	MSG_T	DS(549)	362D	756	760	764
		BASED(PDUMMY)	766	768	777	782
			799	814	824	825
			835	837	847	848
			858	860	865	870
			875	887	931	961
			1162	1175	1206	1217
			1228	1240	1251	1263
			1275	1287	1298	1309
			1320	2150	7233	
*RNF7031	DATA	A(512)	367D			
*RNF7031	FILE	A(10)	369D	766		
*RNF7031	ID	DS(7)	364D	760		
*RNF7031	LIB	A(10)	370D	768		
*RNF7031	TYPE	A(10)	368D	764	870	
*RNF7031	MSGID_T	DS(7)	364	373D		
		BASED(PDUMMY)				
*RNF7031	NBR	S(4,0)	375D			
*RNF7031	PREFIX	A(3)	374D			
	MULTIREF_DELETE	PROTOTYPE	6260D	7442M		
*RNF7031	MULTIREF_ENDCALLBACK...					
		PROTOTYPE	6295D			
*RNF7031	MULTIREF_GET	DS(2109)	6242D			
		PROTOTYPE				
	MULTIREF_GETENDCALLBACK...					
		*(16) PROCPTR	6201D	7820		
		PROTOTYPE				
	MULTIREF_GETSTARTCALLBACK...					
		*(16) PROCPTR	6194D	7819		
		PROTOTYPE				

5761WDS	V6R1M0	080215	RN	IBM ILE RPG	KARLW/WSTST1	S10DCEDB	08/19/09 13:07:17	Page	189
	MULTIREF_ISDATA	N(1)		6217D	7816				
		PROTOTYPE							
	MULTIREF_ISNULL	N(1)		6274D	7441				
		PROTOTYPE							
	MULTIREF_ISREFERENCE...								
		N(1)		6208D	7479	7587	7682		
		PROTOTYPE							
	MULTIREF_NEW	PROTOTYPE		6187D	7261M				
*RNF7031	MULTIREF_NULL	*(16)		6267D					
		PROTOTYPE							
	MULTIREF_PUT	PROTOTYPE		6226D	7480M	7588M	7683M		
*RNF7031	MULTIREF_RELEASE	PROTOTYPE		6252D					
*RNF7031	MULTIREF_STARTCALLBACK...								
		PROTOTYPE		6281D					
*RNF7031	OBJD0100_T	DS(90)		1041D					
		BASED(PDUMMY)							
*RNF7031	AUXSTGP	I(10,0)		1048D					
*RNF7031	BYTAVL	I(10,0)		1043D					
*RNF7031	BYTRET	I(10,0)		1042D					
*RNF7031	CHGDATTIM	A(13)		1052D					
*RNF7031	CRTDATTIM	A(13)		1051D					
*RNF7031	DOMAIN	A(2)		1050D					
*RNF7031	LIB	A(10)		1045D					
*RNF7031	NAME	A(10)		1044D					
*RNF7031	OWNER	A(10)		1049D					
*RNF7031	RTNLIB	A(10)		1047D					
*RNF7031	TYPE	A(10)		1046D					
*RNF7031	ORSTR	PROTOTYPE		1519D					
	P_ASSERT	PROTOTYPE		981D	7989M	8017M			
*RNF7031	P_RMVPGMMSG	PROTOTYPE		804D					
*RNF7031	P_SETAPPMMSGFILE	PROTOTYPE		747D					
*RNF7031	P_WAIT	PROTOTYPE		2195D					
	PDUMMY	*(16)		59	62	71	79		
				85	91	102	261		
				265	268	271	274		
				305	311	318	322		
				362	373	386	462		
				486	552	904	1029		
				1041	1061	1081	1123		
				1128	1134	1473	2104		
				2220	2222	2312	2383		
				2385	2690	2802	2804		
				2807	5525	5531	5540		
				5547	5555	5559	5562		
				5565	5569	5571	5575		
				5579	5583	5587	5591		
				5593	5602	5613	5624		
				5635	5646	5657	5668		
				5679	5690	5701	5712		
				5723	5734	5745	5756		
				5760	5764	5768	5772		
				5776	5780	5784	5791		
				5796	5802	5807	5838		
				5843	5849	5855	5860		
				5866	5878	5884	5891		
				5899	5904	5913	5921		
				5927	5937	5946	5953		
				5959	5965	6175	6835		

5761WDS	V6R1M0	080215 RN	IBM ILE RPG	KARLW/WSTST1	S10DCEDB	08/19/09 13:07:17	Page	190
*RNF7031	QLEACTBNDPGM	I(10,0)	6839					
		PROTOTYPE	1465D					
*RNF7031	QLEGETEXP	*(16)	1484D					
		PROTOTYPE						
*RNF7031	QLICVTTTP	PROTOTYPE	1503D					
*RNF7031	QOBJ_T	DS(20)	1029D	1126	1180	1187		
		BASED(PDUMMY)	1991					
*RNF7031	LIB	A(10)	1031D					
*RNF7031	NAME	A(10)	1030D					
*RNF7031	QPROC_T	DS(288)	1123D					
		BASED(PDUMMY)						
*RNF7031	MODULE	A(10)	1125D					
*RNF7031	NAME	A(256)	1124D					
		VARYING						
*RNF7031	QPGM	DS(20)	1126D					
*RNF7031	RCVM0200_T	DS(4272)	386D	787	869	7232		
		BASED(PDUMMY)						
*RNF7031	ALROPT	A(9)	434D					
*RNF7031	BYTAVL	I(10,0)	390D					
*RNF7031	BYTRET	I(10,0)	388D					
*RNF7031	CCSIDMH	I(10,0)	436D					
*RNF7031	CCSIDRD	I(10,0)	438D					
*RNF7031	CSTIDDT	I(10,0)	432D					
*RNF7031	CSTIDTX	I(10,0)	430D					
*RNF7031	LNMGGA	I(10,0)	446D					
*RNF7031	LNMGHA	I(10,0)	450D					
*RNF7031	LNMGHR	I(10,0)	448D					
*RNF7031	LNMGRR	I(10,0)	444D					
*RNF7031	LNRPDLTA	I(10,0)	442D					
*RNF7031	LNRPDLTR	I(10,0)	440D					
*RNF7031	MSGF	A(10)	400D					
*RNF7031	MSGID	A(7)	394D					
*RNF7031	MSGKEY	A(4)	398D					
*RNF7031	MSGLS	A(10)	402D					
*RNF7031	MSGLU	A(10)	404D					
*RNF7031	MSGSEV	I(10,0)	392D					
*RNF7031	MSGTYPE	A(2)	396D					
*RNF7031	RCVINB	A(4)	422D					
*RNF7031	RCVPGM	A(10)	420D					
*RNF7031	RCVTYP	A(1)	426D					
*RNF7031	RESERVED_01	A(1)	428D					
*RNF7031	SNDDAT	A(7)	416D					
*RNF7031	SNDINB	A(4)	414D					
*RNF7031	SNDJOB	A(10)	406D					
*RNF7031	SNDNBR	A(6)	410D					
*RNF7031	SNDPGM	A(12)	412D					
*RNF7031	SNDTIM	A(6)	418D					
*RNF7031	SNDTYP	A(1)	424D					
*RNF7031	SNDUSR	A(10)	408D					
*RNF7031	VALUE	A(4096)	452D					
*RNF7031	RELEASE	*(16) PROCPTR	3352D					
	REPORTSSERVICEBINDING_V2_0_ENABLEAXIS2095...	PROTOTYPE	225D	8126	8130	8141		
	REPORTSSERVICEBINDING_V2_0_GETENDPOINT...	A(32767)	131D	8201	8205	8213		
		VARYING						
		PROTOTYPE						

5761WDS	V6R1M0	080215	RN	IBM ILE RPG	KARLW/WSTST1	S10DCEDB	08/19/09 13:07:17	Page	191
				REPORTSSERVICEBINDING_V2_0_GETHOST...					
				A(32767)	148D	8240	8244	8256	
				VARYING					
				PROTOTYPE					
				REPORTSSERVICEBINDING_V2_0_GETPORT...					
				I(10,0)	165D	8287	8291	8303	
				PROTOTYPE					
				REPORTSSERVICEBINDING_V2_0_HTTPERROR_GETCODE...					
				I(10,0)	192D	8051	8055	8063	
				PROTOTYPE					
				REPORTSSERVICEBINDING_V2_0_HTTPERROR_GETTEXT...					
				A(128)	200D	8069	8073	8081	
				VARYING					
				PROTOTYPE					
				REPORTSSERVICEBINDING_V2_0_REGISTERLOGINCALLBACK...					
				PROTOTYPE	183D	8030	8034	8045	
				REPORTSSERVICEBINDING_V2_0_SETENDPOINT...					
				PROTOTYPE	139D	8219	8223	8234	
				REPORTSSERVICEBINDING_V2_0_SETHOST...					
				PROTOTYPE	156D	8262	8266	8281	
				REPORTSSERVICEBINDING_V2_0_SETHTTTPDEBUG...					
				PROTOTYPE	111D	8147	8151	8172	
				REPORTSSERVICEBINDING_V2_0_SETPORT...					
				PROTOTYPE	173D	8309	8313	8328	
				REPORTSSERVICEBINDING_V2_0_SETPROXY...					
				PROTOTYPE	121D	8178	8182	8195	
				REPORTSSERVICEBINDING_V2_0_SOAPERROR_GETCODE...					
				A(256)	208D	8087	8091	8099	
				VARYING					
				PROTOTYPE					
				REPORTSSERVICEBINDING_V2_0_SOAPERROR_GETTEXT...					
				A(256)	216D	7432	8105	8109	
				VARYING	8117				
				PROTOTYPE					
*RNF7031				REPORTSSERVICEBINDING_V2_0_SUPPLYLOGINDATA...					
				I(10,0)	245D				
				PROTOTYPE					
				REPORTSSERVICEBINDING_V2_0_VEHICLEPERFORMANCE_GETBYDATE...					
				DS(16648)	42D	7209	7214	7449	
				PROTOTYPE					
*RNF7031	RSLVSP			*(16) PROCPTR	1495D				
				PROTOTYPE					
*RNF7031	RTVM0100_T			DS(2072)	462D				
				BASED(PDUMMY)					
*RNF7031	BYTAVL			B(9,0)	466D				
*RNF7031	BYTRET			B(9,0)	464D				
*RNF7031	DATA			A(2048)	476D				
*RNF7031	LENHLP AVL			B(9,0)	474D				
*RNF7031	LENHLP RET			B(9,0)	472D				
*RNF7031	LENMSG AVL			B(9,0)	470D				
*RNF7031	LENMSG RET			B(9,0)	468D				
*RNF7031	RTVM0300_T			DS(4200)	486D				
				BASED(PDUMMY)					
*RNF7031	ALTIDX			I(10,0)	494D				
*RNF7031	ALTOPT			A(9)	496D				
*RNF7031	BYTAVL			I(10,0)	490D				
*RNF7031	BYTRET			I(10,0)	488D				
*RNF7031	CCSIDTXT			I(10,0)	510D				

5761WDS V6R1M0 080215 RN		IBM ILE RPG	KARLW/WSTST1	S10DCEDB	08/19/09 13:07:17	Page	192
*RNF7031	CSINDDTA	I(10,0)	508D				
*RNF7031	CSINDTXT	I(10,0)	506D				
*RNF7031	DATA	A(4096)	538D				
*RNF7031	LENDFTRPYA	I(10,0)	516D				
*RNF7031	LENDFTRPYR	I(10,0)	514D				
*RNF7031	LENMSG	I(10,0)	522D				
*RNF7031	LENMSGHLPA	I(10,0)	528D				
*RNF7031	LENMSGHLPR	I(10,0)	526D				
*RNF7031	LENMSGR	I(10,0)	520D				
*RNF7031	LENVARFMTA	I(10,0)	534D				
*RNF7031	LENVARFMTE	I(10,0)	536D				
*RNF7031	LENVARFMTR	I(10,0)	532D				
*RNF7031	LOGIND	A(1)	498D				
*RNF7031	MSGID	A(7)	500D				
*RNF7031	MSGSEV	I(10,0)	492D				
*RNF7031	NUMVARFMT	I(10,0)	504D				
*RNF7031	OFSDFTR	I(10,0)	512D				
*RNF7031	OFMSG	I(10,0)	518D				
*RNF7031	OFMSGHLP	I(10,0)	524D				
*RNF7031	OFVARFMT	I(10,0)	530D				
*RNF7031	RESERVED_01	A(3)	502D				
*RNF7031	RTVM0400_T	DS(4360)	552D				
		BASED(PDUMMY)					
*RNF7031	ALTIDX	I(10,0)	560D				
*RNF7031	ALTOPT	A(9)	562D				
*RNF7031	BYTAVL	I(10,0)	556D				
*RNF7031	BYTRET	I(10,0)	554D				
*RNF7031	CCSIDTXT	I(10,0)	576D				
*RNF7031	CRTDATE	A(7)	650D				
*RNF7031	CRTLVLNBR	I(10,0)	654D				
*RNF7031	CSINDDTA	I(10,0)	574D				
*RNF7031	CSINDTXT	I(10,0)	572D				
*RNF7031	DATA	A(4096)	676D				
*RNF7031	DFTPGMLIB	A(10)	674D				
*RNF7031	DFTPGMNAME	A(10)	672D				
*RNF7031	LENDFTRPYA	I(10,0)	582D				
*RNF7031	LENDFTRPYR	I(10,0)	580D				
*RNF7031	LENDMPLISTEA	I(10,0)	670D				
*RNF7031	LENDMPLISTER	I(10,0)	668D				
*RNF7031	LENLOWRPYEA	I(10,0)	636D				
*RNF7031	LENLOWRPYER	I(10,0)	634D				
*RNF7031	LENMSG	I(10,0)	588D				
*RNF7031	LENMSGHLPA	I(10,0)	594D				
*RNF7031	LENMSGHLPR	I(10,0)	592D				
*RNF7031	LENMSGR	I(10,0)	586D				
*RNF7031	LENRELTESTEA	I(10,0)	648D				
*RNF7031	LENRELTESTER	I(10,0)	646D				
*RNF7031	LENRPYE	I(10,0)	620D				
*RNF7031	LENRPYEA	I(10,0)	618D				
*RNF7031	LENRPYER	I(10,0)	616D				
*RNF7031	LENSPCRPYE	I(10,0)	630D				
*RNF7031	LENSPCRPYEA	I(10,0)	628D				
*RNF7031	LENSPCRPYER	I(10,0)	626D				
*RNF7031	LENVARFMTA	I(10,0)	600D				
*RNF7031	LENVARFMTE	I(10,0)	602D				
*RNF7031	LENVARFMTR	I(10,0)	598D				
*RNF7031	LOGIND	A(1)	564D				
*RNF7031	MAXRPYDECPOS	I(10,0)	610D				

*RNF7031	MAXRPYLEN	I(10,0)	608D			
*RNF7031	MODDATE	A(7)	656D			
*RNF7031	MODLVLNBR	I(10,0)	660D			
*RNF7031	MSGID	A(7)	566D			
*RNF7031	MSGSEV	I(10,0)	558D			
*RNF7031	NUMDMPLISTER	I(10,0)	666D			
*RNF7031	NUMRPYER	I(10,0)	614D			
*RNF7031	NUMSPCRPYER	I(10,0)	624D			
*RNF7031	NUMVARFMT	I(10,0)	570D			
*RNF7031	OFSDFTR	I(10,0)	578D			
*RNF7031	OFSDMPLISTE	I(10,0)	664D			
*RNF7031	OFSLOWRPYE	I(10,0)	632D			
*RNF7031	OFSMSG	I(10,0)	584D			
*RNF7031	OFSMSGHLP	I(10,0)	590D			
*RNF7031	OFSRELTESTE	I(10,0)	644D			
*RNF7031	OFSRPYE	I(10,0)	612D			
*RNF7031	OFSSPCRPYE	I(10,0)	622D			
*RNF7031	OFSUPRPYE	I(10,0)	638D			
*RNF7031	OFSUPRPYEA	I(10,0)	642D			
*RNF7031	OFSUPRPYER	I(10,0)	640D			
*RNF7031	OFSVARFMT	I(10,0)	596D			
*RNF7031	RESERVED_01	A(3)	568D			
*RNF7031	RESERVED_02	A(2)	606D			
*RNF7031	RESERVED_03	A(1)	652D			
*RNF7031	RESERVED_04	A(1)	658D			
*RNF7031	RPYTYPE	A(10)	604D			
*RNF7031	STOREDCCSID	I(10,0)	662D			
*RNF7031	SCHEMA_END	PROTOTYPE	6769D			
*RNF7031	SCHEMA_INITIALIZE	PROTOTYPE	6745D			
*RNF7031	SCHEMA_START	PROTOTYPE	6754D			
*RNF7031	SHA1_T	DS(42)	1621D	1684		
*RNF7031	LENGTH	U(5,0)	1622D			
*RNF7031	VALUE	A(40)	1623D			
	SOAPMESSAGE_ENVELOPE...					
		PROTOTYPE	7071D	7789	7793	7835
	SOAPMESSAGE_FAULTDETAILEND...					
		PROTOTYPE	7124D	7940	7944	7967
	SOAPMESSAGE_FAULTDETAILSTART...					
		PROTOTYPE	7111D	7910	7914	7934
	SOAPMESSAGE_FAULTEND...					
		PROTOTYPE	7097D	7875	7879	7904
	SOAPMESSAGE_FAULTSTART...					
		PROTOTYPE	7084D	7841	7845	7869
	SOAPMESSAGE_NULLEND...					
		PROTOTYPE	7152D	8001	8005	8023
	SOAPMESSAGE_NULLSTART...					
		PROTOTYPE	7138D	7973	7977	7995
*RNF7031	S1_DTSCANREPORTENTRY_T...					
		A(128)	58D	66		
		BASED(PDUMMY)				
		VARYING				
	S1_ENTRYEND_OF_S1_VEHICLEPERFORMANCE_GETBYDATERESULT...					
		PROTOTYPE	7057D	7715	7720	7756
	S1_ENTRYSTART_OF_S1_VEHICLEPERFORMANCE_GETBYDATERESULT...					
		PROTOTYPE	7045D	7660	7665	7708
*RNF7031	S1_RPGARRAYOFENTRY_T...					
		DS(16644)	62D	87	7674	7730
		BASED(PDUMMY)				

5761WDS	V6R1M0	080215	RN	IBM ILE RPG	KARLW/WSTST1	S10DCEDB	08/19/09	13:07:17	Page	194
*RNF7031	DTSCANREPORTENTRY(128)...									
	A(128)		66D							
	VARYING									
*RNF7031	X	I(10,0)	64D							
*RNF7031	S1_RPGARRAYOFTRUCKPCBASE_T...									
	DS(16644)		71D	93						
	BASED(PDUMMY)									
*RNF7031	ITEM(128)	DS(130)	74D							
*RNF7031	X	I(10,0)	73D							
*RNF7031	S1_TRUCKPCBASERNMD_T...									
	DS(130)		74	79D						
	BASED(PDUMMY)									
*RNF7031	TRUCKNAME	A(128)	81D							
	VARYING									
	S1_VEHICLEPERFORMANCE_GETBYDATERESULTEND...									
	PROTOTYPE	7032D		7627	7630	7654				
	S1_VEHICLEPERFORMANCE_GETBYDATERESULTSTART...									
	PROTOTYPE	7021D		7566	7569	7621				
*RNF7031	S1_VEHICLEPERFORMANCECEREPORT_T...									
	DS(16644)		85D	105	7579	7641				
	BASED(PDUMMY)									
*RNF7031	ENTRY	DS(16644)	87D							
	TEMPLATE	*(16)	3307	3348						
*RNF7031	TNS_VEHICLEPERFORMANCE_GETBYDATE_T...									
	DS(16701)		47	91D	7217					
	BASED(PDUMMY)									
*RNF7031	BATCHSIZE	I(10,0)	94D							
*RNF7031	ENDDATE	Z(26)	95D							
*RNF7031	QUERYBYRECORDDATE...									
	N(1)		97D							
*RNF7031	STARTDATE	Z(26)	98D							
*RNF7031	TRUCKPCBASE	DS(16644)	93D							
*RNF7031	TNS_VEHICLEPERFORMANCE_GETBYDATERESPONSE_T...									
	DS(16648)		42	102D	7214	7223				
	BASED(PDUMMY)	7252		7470	7540					
*RNF7031	TOTALROWCOUNT	I(10,0)	106D							
*RNF7031	VEHICLEPERFORMANCE_GETBYDATERESULT...									
	DS(16644)		105D							
	TNS_VEHICLEPERFORMANCE_GETBYDATERESPONSEEND...									
	PROTOTYPE	7009D		7526	7529	7560				
	TNS_VEHICLEPERFORMANCE_GETBYDATERESPONSESTART...									
	PROTOTYPE	6998D		7457	7460	7520				
	TO_ASCII	CONST	5464D							
	TO_EBCDIC	CONST	5465D							
*RNF7031	UNMARSHALLER_TOANY...									
	A(65535)		6327D							
	VARYING									
	PROTOTYPE									
	UNMARSHALLER_TOANYTYPE...									
	A(65535)		6319D	7748						
	VARYING									
	PROTOTYPE									
*RNF7031	UNMARSHALLER_TOANYURI...									
	A(65535)		6511D							
	VARYING									
	PROTOTYPE									
*RNF7031	UNMARSHALLER_TOBASE64BINARY...									
	A(65535)		6487D							

		VARYING PROTOTYPE		
*RNF7031	UNMARSHALLER_TOBOOLEAN...	N(1) PROTOTYPE	6351D	
*RNF7031	UNMARSHALLER_TODATE...	D(10*ISO-) PROTOTYPE	6471D	
*RNF7031	UNMARSHALLER_TODATETIME...	Z(26) PROTOTYPE	6463D	
*RNF7031	UNMARSHALLER_TODECIMAL...	S(31,15) PROTOTYPE	6359D	
*RNF7031	UNMARSHALLER_TODOUBLE...	S(31,15) PROTOTYPE	6367D	
*RNF7031	UNMARSHALLER_TODURATION...	DS(25) PROTOTYPE	6519D	
*RNF7031	UNMARSHALLER_TOFLOAT...	S(31,15) PROTOTYPE	6375D	
	UNMARSHALLER_TOINT...	I(10,0) PROTOTYPE	6383D	7552
*RNF7031	UNMARSHALLER_TOLANGUAGE...	A(65535) VARYING PROTOTYPE	6495D	
*RNF7031	UNMARSHALLER_TOLONG...	I(20,0) PROTOTYPE	6399D	
*RNF7031	UNMARSHALLER_TONEGATIVEINTEGER...	I(10,0) PROTOTYPE	6447D	
*RNF7031	UNMARSHALLER_TONONNEGATIVEINTEGER...	I(10,0) PROTOTYPE	6431D	
*RNF7031	UNMARSHALLER_TONONPOSITIVEINTEGER...	I(10,0) PROTOTYPE	6439D	
*RNF7031	UNMARSHALLER_TOPOSITIVEINTEGER...	I(10,0) PROTOTYPE	6455D	
*RNF7031	UNMARSHALLER_TOSHEMA...	A(65535) VARYING PROTOTYPE	6335D	
*RNF7031	UNMARSHALLER_TOSHORT...	I(5,0) PROTOTYPE	6391D	
*RNF7031	UNMARSHALLER_TOSTRING...	A(65535) VARYING PROTOTYPE	6343D	
*RNF7031	UNMARSHALLER_TOTIME...	T(8*ISO.)	6479D	

		PROTOTYPE				
*RNF7031	UNMARSHALLER_TOTOKEN...	A(65535)	6503D			
		VARYING				
		PROTOTYPE				
*RNF7031	UNMARSHALLER_TOUNSIGNEDINT...	U(10,0)	6407D			
		PROTOTYPE				
*RNF7031	UNMARSHALLER_TOUNSIGNEDLONG...	U(20,0)	6423D			
		PROTOTYPE				
*RNF7031	UNMARSHALLER_TOUNSIGNEDSHORT...	U(5,0)	6415D			
		PROTOTYPE				
*RNF7031	URL_GETFILE	A(446)	3152D			
		VARYING				
	URL_GETHOST	PROTOTYPE				
		A(446)	3095D	8252		
		VARYING				
		PROTOTYPE				
*RNF7031	URL_GETMBR	N(1)	3199D			
		PROTOTYPE				
*RNF7031	URL_GETPATH	A(446)	3133D			
		VARYING				
	URL_GETPORT	PROTOTYPE				
		I(10,0)	3114D	8299		
		PROTOTYPE				
*RNF7031	URL_GETPROTOCOL	A(446)	3076D			
		VARYING				
		PROTOTYPE				
*RNF7031	URL_GETQUERY	A(446)	3171D			
		VARYING				
	URL_NEW	PROTOTYPE				
		A(446)	3067D	8250	8273	8297
		VARYING	8320			
		PROTOTYPE				
*RNF7031	URL_SETFILE	N(1)	3161D			
		PROTOTYPE				
	URL_SETHOST	N(1)	3104D	8274M		
		PROTOTYPE				
*RNF7031	URL_SETPATH	N(1)	3142D			
		PROTOTYPE				
	URL_SETPORT	N(1)	3123D	8321M		
		PROTOTYPE				
*RNF7031	URL_SETPROTOCOL	N(1)	3085D			
		PROTOTYPE				
*RNF7031	URL_SETQUERY	N(1)	3180D			
		PROTOTYPE				
*RNF7031	URL_T	A(446)	3047D	3067	3071	3076
		VARYING	3080	3089	3090	3095
			3099	3108	3109	3118
			3127	3133	3137	3146
			3147	3152	3156	3165
			3166	3171	3175	3184
			3185	3194	3203	8246
			8269	8293	8316	
	URL_TOSTRING	A(512)	3190D	8275	8322	
		VARYING				

		PROTOTYPE	
*RNF7031	UUID_T	DS(34)	1585D
			1666
*RNF7031	LENGTH	U(5,0)	1586D
*RNF7031	VALUE	A(32)	1587D
*RNF7031	WAITTIME	PROTOTYPE	1557D
	WLI_ALRTBL	CONST	1366D
	WLI_AUTL	CONST	1367D
	WLI_BLKSF	CONST	1368D
	WLI_BNDDIR	CONST	1369D
	WLI_CFGL	CONST	1370D
	WLI_CHTFMT	CONST	1371D
	WLI_CLD	CONST	1372D
	WLI_CLS	CONST	1373D
	WLI_CMD	CONST	1374D
	WLI_CNNL	CONST	1375D
	WLI_COSD	CONST	1376D
	WLI_CRG	CONST	1377D
	WLI_CRQD	CONST	1378D
	WLI_CSI	CONST	1379D
	WLI_CSPMAP	CONST	1380D
	WLI_CSPTBL	CONST	1381D
	WLI_CTLD	CONST	1382D
	WLI_DDIR	CONST	1383D
	WLI_DEVD	CONST	1384D
	WLI_DIR	CONST	1385D
	WLI_DOC	CONST	1386D
	WLI_DSTMF	CONST	1387D
	WLI_DTAARA	CONST	1388D
	WLI_DTADCT	CONST	1389D
	WLI_DTAQ	CONST	1390D
	WLI_EDTD	CONST	1391D
	WLI_EXITRG	CONST	1392D
	WLI_FCT	CONST	1393D
	WLI_FILE	CONST	1394D
	WLI_FLR	CONST	1395D
	WLI_FNTRSC	CONST	1396D
	WLI_FNTTBL	CONST	1397D
	WLI_FORMDF	CONST	1398D
	WLI_FTR	CONST	1399D
	WLI_GSS	CONST	1400D
	WLI_IGCDCT	CONST	1401D
	WLI_IGCSRT	CONST	1402D
	WLI_IGCTBL	CONST	1403D
	WLI_IPXD	CONST	1404D
	WLI_JOB	CONST	1405D
	WLI_JOBQ	CONST	1406D
	WLI_JOBSCD	CONST	1407D
	WLI_JRN	CONST	1408D
	WLI_JRNRCV	CONST	1409D
	WLI_LIB	CONST	1410D
	WLI_LIND	CONST	1411D
	WLI_LOCALE	CONST	1412D
	WLI_MEDDFN	CONST	1413D
	WLI_MENU	CONST	1414D
	WLI_MGTCOL	CONST	1415D
	WLI_MODD	CONST	1416D
	WLI_MODULE	CONST	1417D
	WLI_MSGF	CONST	1418D

WLI_MSGQ	CONST	1419D	
WLI_M36	CONST	1420D	
WLI_M36CFG	CONST	1421D	
WLI_NODGRP	CONST	1426D	
WLI_NODL	CONST	1422D	
WLI_NTBD	CONST	1423D	
WLI_NWID	CONST	1424D	
WLI_NWSD	CONST	1425D	
WLI_OOPOOL	CONST	1427D	
WLI_OUTQ	CONST	1428D	
WLI_OVL	CONST	1429D	
WLI_PAGDFN	CONST	1430D	
WLI_PAGSEG	CONST	1431D	
WLI_PDG	CONST	1432D	
WLI_PGM	CONST	1433D	
WLI_PNLGRP	CONST	1434D	
WLI_PRDAVL	CONST	1435D	
WLI_PRDDFN	CONST	1436D	
WLI_PRDLOD	CONST	1437D	
WLI_PSFCFG	CONST	1438D	
WLI_QMFORM	CONST	1439D	
WLI_QMQRV	CONST	1440D	
WLI_QRYDFN	CONST	1441D	
WLI_RCT	CONST	1442D	
WLI_SBSD	CONST	1444D	
WLI_SCHIDX	CONST	1445D	
WLI_SOCKET	CONST	1446D	
WLI_SOMOBJ	CONST	1447D	
WLI_SPADCT	CONST	1448D	
WLI_SQLPKG	CONST	1449D	
WLI_SQLUDT	CONST	1450D	
WLI_SRPVGM	CONST	1451D	
WLI_SSND	CONST	1452D	
WLI_STMF	CONST	1453D	
WLI_SVRSTG	CONST	1454D	
WLI_SYMLNK	CONST	1455D	
WLI_S36	CONST	1443D	
WLI_TBL	CONST	1456D	
WLI_USRIDX	CONST	1458D	
WLI_USRPRF	CONST	1457D	
WLI_USRQ	CONST	1459D	
WLI_USRSPC	CONST	1460D	
WLI_VLDL	CONST	1461D	
WLI_WSCST	CONST	1462D	
*RNF7031 WORKSTN_INFDS_T	DS(369)	6839D	
	BASED(PDUMMY)		
*RNF7031 AID	A(1)	6842D	
*RNF7031 OPEN	A(1)	6840D	
*RNF7031 STATUS	S(5, 0)	6841D	
*RNF7031 WSDL_ATTRIBUTE_T	DS(8192)	5953D	
	BASED(PDUMMY)		
*RNF7031 NAME	A(4094)	5954D	
	VARYING		
*RNF7031 VALUE	A(4094)	5955D	
	VARYING		
WSDL_AUTH_TYPE_HTTP...			
	CONST	288D	7373
WSDL_AUTH_TYPE_PROXY...			

5761WDS	V6R1M0	080215	RN	IBM ILE RPG	KARLW/WSTST1	S10DCEDB	08/19/09 13:07:17	Page	199
			CONST		290D	7376			
*RNF7031	WSDL_BINDING_T		DS(3169)		5904D				
			BASED(PDUMMY)						
*RNF7031	BOUNDOPERATIONLIST...								
			DS(1041)		5909D				
*RNF7031	QNAME		DS(1056)		5905D				
*RNF7031	QTYPE		DS(1056)		5906D				
*RNF7031	STYLE		A(14)		5907D				
			VARYING						
*RNF7031	WSDL_BINDINGS_T		DS(1041)		5899D				
			BASED(PDUMMY)						
*RNF7031	BINDINGSLIST		DS(1041)		5900D				
*RNF7031	WSDL_BOUNDOPERATION_T...								
			DS(3712)		5913D				
			BASED(PDUMMY)						
*RNF7031	INPUT		DS(1072)		5916D				
*RNF7031	OUTPUT		DS(1072)		5917D				
*RNF7031	QNAME		DS(1056)		5914D				
*RNF7031	SOAPACTION		A(510)		5915D				
			VARYING						
*RNF7031	WSDL_BOUNDOPERATIONIO_T...								
			DS(1072)		5916	5917	5921D		
			BASED(PDUMMY)						
*RNF7031	QNAME		DS(1056)		5922D				
*RNF7031	USE		A(14)		5923D				
			VARYING						
*RNF7031	WSDL_CMDSERVER_T		DS(146)		5525D				
			BASED(PDUMMY)						
*RNF7031	NUME		I(5,0)		5526D				
*RNF7031	PROXYSERVER		DS(144)		5527D				
*RNF7031	WSDL_CONTENTTYPE_T...								
			A(1024)		306	5555D			
			BASED(PDUMMY)						
			VARYING						
	WSDL_DECIMAL_T		S(31,15)		5602D	5607	5609	5613	
			BASED(PDUMMY)		5624	6359	6573		
*RNF7031	WSDL_DEFTYPE_T		A(5)		5780D	5817			
			BASED(PDUMMY)						
	WSDL_DIM_T		I(10,0)		5593D	5595	5596	5816	
			BASED(PDUMMY)						
*RNF7031	WSDL_DIMXML_T		A(30)		5591D				
			BASED(PDUMMY)						
			VARYING						
*RNF7031	WSDL_DOCUMENTATION_T...								
			A(4094)		5768D	5872			
			BASED(PDUMMY)						
			VARYING						
	WSDL_DOUBLE_T		S(31,15)		5613D	5618	5620	6367	
			BASED(PDUMMY)		6581				
*RNF7031	WSDL_DSPEC_T		DS(49245)		5927D				
			BASED(PDUMMY)						
*RNF7031	DECLTYPE		A(20)		5929D				
*RNF7031	KEYWORDS(10)		A(4096)		5932D				
			VARYING						
*RNF7031	NAME		A(4096)		5928D				
			VARYING						
*RNF7031	NUMKWDS		I(10,0)		5931D				
*RNF7031	PRIMITIVE		DS(4143)		5930D				

5761WDS V6R1M0 080215 RN		IBM ILE RPG	KARLW/WSTST1		S10DCEDB	08/19/09 13:07:17	Page	200
*RNF7031	WSDL_DURATION_T	DS(25) BASED(PDUMMY)	5745D	6519	6740			
*RNF7031	DAYS	I(10,0)	5749D					
*RNF7031	HOURS	I(10,0)	5750D					
*RNF7031	ISNEGATIVE	N(1)	5746D					
*RNF7031	MINUTES	I(10,0)	5751D					
*RNF7031	MONTHS	I(10,0)	5748D					
*RNF7031	SECONDS	I(10,0)	5752D					
*RNF7031	YEARS	I(10,0)	5747D					
	WSDL_ERROR	CONST	285D					
*RNF7031	WSDL_ERROR_T	I(10,0) BASED(PDUMMY)	5579D					
*RNF7031	WSDL_ERRTEXT_T	A(128) BASED(PDUMMY) VARYING	50	261D				
	WSDL_FLOAT_T	S(31,15) BASED(PDUMMY)	5624D	5629	5631	6375		
		DS(576)	6589					
*RNF7031	WSDL_GENTAG_T	BASED(PDUMMY)	5965D					
*RNF7031	COMMAND	A(30) VARYING	5966D					
*RNF7031	CONTCHARS	A(14) VARYING	5972D					
*RNF7031	EXTENSION	A(254) VARYING	5973D					
*RNF7031	LENGTH	I(10,0)	5968D					
*RNF7031	MAXCOLUMN	I(10,0)	5970D					
*RNF7031	RESERVED_1	A(4)	5971D					
*RNF7031	STARTCOLUMN	I(10,0)	5969D					
*RNF7031	VARIABLE	A(254) VARYING	5967D					
	WSDL_HTTP_ERROR_UNEXPECTED...	CONST	280D	7419				
*RNF7031	WSDL_HTTP_ERRORNO_T...	I(10,0) BASED(PDUMMY)	192	265D	323	8055		
*RNF7031	WSDL_HTTP_ERRORTXT_T...	A(128) BASED(PDUMMY) VARYING	200	268D	324	8073		
	WSDL_HTTP_SUCCESS	CONST	278D					
*RNF7031	WSDL_HTTPERROR_T	DS(134) BASED(PDUMMY)	312	322D				
*RNF7031	NO	I(10,0)	323D					
*RNF7031	TEXT	A(128) VARYING	324D					
	WSDL_INTEGER_T	I(10,0) BASED(PDUMMY)	5646D	5651	5653	6383		
		A(510)	6597					
*RNF7031	WSDL_LNAME_T	BASED(PDUMMY) VARYING	5583D	5947				
	WSDL_LONG_T	I(20,0) BASED(PDUMMY)	5657D	5662	5664	6399		
		DS(1041)	6613					
*RNF7031	WSDL_MAP_T	BASED(PDUMMY)	5792	5798	5803	5821		
			5839	5845	5856	5862		
			5884D	5900	5909			
*RNF7031	CURKEY	A(1022) VARYING	5886D					

5761WDS	V6R1M0	080215	RN	IBM ILE RPG	KARLW/WSTST1	S10DCEDB	08/19/09 13:07:17	Page	201
*RNF7031	HMAP			*(16)	5885D				
*RNF7031	ISCHOICE			N(1)	5887D				
*RNF7031	WSDL_MAPKEY_T			A(1022)	5568D	5886			
				BASED(PDUMMY)					
				VARYING					
*RNF7031	WSDL_MESSAGE_T			DS(2097)	5843D				
				BASED(PDUMMY)					
*RNF7031	PARTSLIST			DS(1041)	5845D				
*RNF7031	QNAME			DS(1056)	5844D				
*RNF7031	WSDL_MESSAGES_T			DS(1041)	5838D				
				BASED(PDUMMY)					
*RNF7031	MSGLIST			DS(1041)	5839D				
*RNF7031	WSDL_MULTIREF_T			DS(2109)	6175D	6242	6256		
				BASED(PDUMMY)					
*RNF7031	CBEND			*(16) PROCPTR	6178D				
*RNF7031	CBSTART			*(16) PROCPTR	6177D				
*RNF7031	DEPTH			I(10,0)	6180D				
*RNF7031	ISACTIVE			N(1)	6179D				
*RNF7031	NAME			A(1024)	6181D				
				VARYING					
*RNF7031	NAMESPACE			A(1024)	6182D				
				VARYING					
*RNF7031	PATHLEN			I(10,0)	6183D				
*RNF7031	USERDATA			*(16)	6176D				
*RNF7031	WSDL_NAME_T			A(4094)	5562D	5760	5954		
				BASED(PDUMMY)					
				VARYING					
	WSDL_NEGINT_T			I(10,0)	5723D	5728	5730	6447	
				BASED(PDUMMY)	6661				
	WSDL_NONNEGINT_T			I(10,0)	5701D	5706	5708	6431	
				BASED(PDUMMY)	6645				
	WSDL_NONPOSINT_T			I(10,0)	5712D	5717	5719	6439	
				BASED(PDUMMY)	6653				
*RNF7031	WSDL_NSPREFIX_T			A(30)	5575D	5948			
				BASED(PDUMMY)					
				VARYING					
*RNF7031	WSDL_NSVALUE_T			A(510)	5587D	5949			
				BASED(PDUMMY)					
				VARYING					
*RNF7031	WSDL_OPERATION_T			DS(13488)	5866D				
				BASED(PDUMMY)					
*RNF7031	DOCUMENTATION			A(4094)	5872D				
				VARYING					
*RNF7031	INPUT			DS(2112)	5873D				
*RNF7031	ISUSED			N(1)	5867D				
*RNF7031	OUTPUT			DS(2112)	5874D				
*RNF7031	PARAMETERORDER			A(4094)	5871D				
				VARYING					
*RNF7031	QNAME			DS(1056)	5869D				
*RNF7031	RESERVED_1			A(15)	5868D				
*RNF7031	WSDL_OPERATIONIO_T...								
				DS(2112)	5873	5874	5878D		
				BASED(PDUMMY)					
*RNF7031	QMESSAGE			DS(1056)	5880D				
*RNF7031	QNAME			DS(1056)	5879D				
*RNF7031	WSDL_OPERATIONSX_T...								
				DS(33024)	5536	5547D			
				BASED(PDUMMY)					

5761WDS V6R1M0 080215 RN		IBM ILE RPG		KARLW/WSTST1		S10DCEDB 08/19/09 13:07:17		Page	202								
*RNF7031	NAME(128)	A(256)	5548D														
		VARYING															
*RNF7031	WSDL_PARAMETERORDER_T...	A(4094)	5760D	5871													
		BASED(PDUMMY)															
		VARYING															
*RNF7031	WSDL_PART_T	DS(2112)	5849D														
		BASED(PDUMMY)															
*RNF7031	QELEMENT	DS(1056)	5851D														
*RNF7031	QNAME	DS(1056)	5850D														
*RNF7031	WSDL_PATH_T	A(254)	5559D														
		BASED(PDUMMY)															
		VARYING															
*RNF7031	WSDL_PORT_T	DS(2384)	5891D														
		BASED(PDUMMY)															
*RNF7031	ADDRESS	A(254)	5895D														
		VARYING															
*RNF7031	PROTOCOL	A(14)	5893D														
		VARYING															
*RNF7031	QBINDING	DS(1056)	5894D														
*RNF7031	QNAME	DS(1056)	5892D														
*RNF7031	WSDL_PORTTYPE_T	DS(2097)	5860D														
		BASED(PDUMMY)															
*RNF7031	OPERATIONLIST	DS(1041)	5862D														
*RNF7031	QNAME	DS(1056)	5861D														
*RNF7031	WSDL_PORTTYPES_T	DS(1041)	5855D														
		BASED(PDUMMY)															
*RNF7031	PORTTYPELIST	DS(1041)	5856D														
	WSDL_POSINT_T	I(10,0)	5734D	5739	5741	6455											
		BASED(PDUMMY)	6669														
*RNF7031	WSDL_PROTOCOL_T	A(14)	5764D	5893													
		BASED(PDUMMY)															
		VARYING															
*RNF7031	WSDL_QNAME_T	DS(1056)	5797	5819	5820	5844											
		BASED(PDUMMY)	5850	5851	5861	5869											
			5879	5880	5892	5894											
			5905	5906	5914	5922											
			5946D														
*RNF7031	LNAME	A(510)	5947D														
		VARYING															
*RNF7031	NSPREFIX	A(30)	5948D														
		VARYING															
*RNF7031	NSVALUE	A(510)	5949D														
		VARYING															
*RNF7031	WSDL_QSERVER_T	DS(144)	5527	5784D													
		BASED(PDUMMY)															
*RNF7031	NAME	A(126)	5785D														
		VARYING															
*RNF7031	PORT	I(10,0)	5786D														
*RNF7031	RESERVED_1	A(12)	5787D														
*RNF7031	WSDL_RANGE_T	DS(16)	5959D														
		BASED(PDUMMY)															
*RNF7031	FROM	I(10,0)	5960D														
*RNF7031	RESERVED_1	A(8)	5962D														
*RNF7031	TO	I(10,0)	5961D														
*RNF7031	WSDL_RPGTYPE_T	DS(4143)	5930	5937D													
		BASED(PDUMMY)															
*RNF7031	DATATYPE	A(2)	5939D														

*RNF7031	DECPOS	S(2,0)	5940D			
*RNF7031	KEYWORD	A(32)	5941D			
		VARYING				
*RNF7031	LENGTH	S(7,0)	5938D			
*RNF7031	REFFIELD	A(4096)	5942D			
		VARYING				
*RNF7031	WSDL_SERVICE_T	DS(2097)	5796D			
		BASED(PDUMMY)				
*RNF7031	PORTLIST	DS(1041)	5798D			
*RNF7031	QNAME	DS(1056)	5797D			
*RNF7031	WSDL_SERVICES_T	DS(1041)	5791D			
		BASED(PDUMMY)				
*RNF7031	SERVICELIST	DS(1041)	5792D			
	WSDL_SHORT_T	I(5,0)	5635D	5640	5642	6391
		BASED(PDUMMY)	6605			
*RNF7031	WSDL_SOAP_FAULT_T	DS(908)	311D	7185	7856	7891
		BASED(PDUMMY)				
*RNF7031	DETAIL	DS(258)	315D			
*RNF7031	FAULTCODE	A(256)	313D			
		VARYING				
*RNF7031	FAULTSTRING	A(256)	314D			
		VARYING				
*RNF7031	HTTPERROR	DS(134)	312D			
*RNF7031	WSDL_SOAP_FAULTCODE_T...					
		A(256)	208	271D	313	8091
		BASED(PDUMMY)				
		VARYING				
*RNF7031	WSDL_SOAP_FAULTDETAIL_T...					
		DS(258)	315	318D	7925	7956
		BASED(PDUMMY)				
*RNF7031	HOSTNAME	A(256)	319D			
		VARYING				
*RNF7031	WSDL_SOAP_FAULTSTRING_T...					
		A(256)	216	274D	314	8109
		BASED(PDUMMY)				
		VARYING				
*RNF7031	WSDL_SOAPACTION_T	A(510)	307	5756D	5915	
		BASED(PDUMMY)				
		VARYING				
*RNF7031	WSDL_SRCMBROPTS_T	DS(13)	5540D			
		BASED(PDUMMY)				
*RNF7031	DOREPLACE	N(1)	5543D			
*RNF7031	NAME	A(10)	5542D			
*RNF7031	NUMPARMS	I(5,0)	5541D			
*RNF7031	WSDL_STYLE_T	A(14)	5776D	5907		
		BASED(PDUMMY)				
		VARYING				
	WSDL_SUCCESS	CONST	283D			
*RNF7031	WSDL_TYPE_T	DS(3185)	5807D			
		BASED(PDUMMY)				
*RNF7031	ARRAYID	I(10,0)	5815D			
*RNF7031	ARRAYTYPE	A(10)	5814D			
		VARYING				
*RNF7031	DEFTYPE	A(5)	5817D			
*RNF7031	DIM	I(10,0)	5816D			
*RNF7031	ISCHOICE	N(1)	5808D			
*RNF7031	ISQUALIFIED	N(1)	5811D			
*RNF7031	ISSEQUENCE	N(1)	5810D			

5761WDS	V6R1M0	080215	RN	IBM ILE RPG	KARLW/WSTST1	S10DCEDB	08/19/09	13:07:17	Page	204
*RNF7031	ISUSED	N(1)			5809D					
*RNF7031	QNAME	DS(1056)			5819D					
*RNF7031	QTYPE	DS(1056)			5820D					
*RNF7031	REFERENCESTYPE	N(1)			5813D					
*RNF7031	RESERVED_1	A(2)			5818D					
*RNF7031	SUBTYPESLIST	DS(1041)			5821D					
*RNF7031	WSDL_TYPES_T	DS(1041)			5802D					
		BASED(PDUMMY)								
*RNF7031	TYPESLIST	DS(1041)			5803D					
	WSDL_UIINTEGER_T	U(10,0)			5679D	5684	5686	6407		
		BASED(PDUMMY)			6621					
	WSDL_ULONG_T	U(20,0)			5690D	5695	5697	6423		
		BASED(PDUMMY)			6637					
*RNF7031	WSDL_URL_T	A(254)			5571D	5895				
		BASED(PDUMMY)								
		VARYING								
*RNF7031	WSDL_USE_T	A(14)			5772D	5923				
		BASED(PDUMMY)								
		VARYING								
	WSDL_USHORT_T	U(5,0)			5668D	5673	5675	6415		
		BASED(PDUMMY)			6629					
*RNF7031	WSDL_VALUE_T	A(4094)			5565D	5955				
		BASED(PDUMMY)								
		VARYING								
*RNF7031	WSDL_WEBSERVICE_T	DS(33288)			5531D					
		BASED(PDUMMY)								
*RNF7031	NUMOPERATIONS	I(5,0)			5535D					
*RNF7031	NUMPARMS	I(5,0)			5532D					
*RNF7031	OFSOPERNMES	I(5,0)			5534D					
*RNF7031	OPERATIONS	DS(33024)			5536D					
*RNF7031	PORT	A(256)			5533D					
		VARYING								
	XML_ENC_ISO8859_1	CONST			3295D					
	XML_ENC_US_ASCII	CONST			3298D					
	XML_ENC_UTF16	CONST			3293D					
	XML_ENC_UTF8	CONST			3291D					
*RNF7031	XML_ENCODING	DS(1072)			3348D	3563				
		BASED(TEMPLATE)								
*RNF7031	XML_ERRORSTRING	*(16)			3499D					
		PROTOTYPE								
*RNF7031	XML_GETCURRENTCOLUMNNUMBER...	I(10,0)			3524D					
		PROTOTYPE								
*RNF7031	XML_GETCURRENTLINENUMBER...	I(10,0)			3511D					
		PROTOTYPE								
*RNF7031	XML_GETERRORCODE	I(10,0)			3487D					
		PROTOTYPE								
*RNF7031	XML_PARSE	I(10,0)			3472D					
		PROTOTYPE								
*RNF7031	XML_PARSER	*(16)			3307D	3365	3379	3415		
		BASED(TEMPLATE)			3422	3428	3455	3473		
					3488	3513	3526	3537		
					3561	3590	3612	3631		
*RNF7031	XML_PARSERCREATE	*(16)			3364D					
		PROTOTYPE								
*RNF7031	XML_PARSERCREATENS...	*(16)			3588D					

		PROTOTYPE	
*RNF7031	XML_PARSERFREE	PROTOTYPE	3536D
*RNF7031	XML_SETCHARACTERDATAHANDLER...		
		PROTOTYPE	3453D
*RNF7031	XML_SETELEMENTHANDLER...		
		PROTOTYPE	3414D
*RNF7031	XML_SETENDELEMENTHANDLER...		
		PROTOTYPE	3426D
*RNF7031	XML_SETENDNAMESPACEDECLHANDLER...		
		PROTOTYPE	3629D
*RNF7031	XML_SETSTARTELEMENTHANDLER...		
		PROTOTYPE	3420D
*RNF7031	XML_SETSTARTNAMESPACEDECLHANDLER...		
		PROTOTYPE	3610D
*RNF7031	XML_SETUNKNOWNENCODINGHANDLER...		
		PROTOTYPE	3559D
*RNF7031	XML_SETUSERDATA	PROTOTYPE	3378D
	XML_STATUS_ERROR	CONST	3284D
	XML_STATUS_OK	CONST	3282D
*RNF7031	XORSTR	PROTOTYPE	1527D
Field References for subprocedure REPORTSSERVICEBINDING_V2_0_VEHICLEPERFORMANCE_GETBYDATE			

Field	Attributes	References (D=Defined M=Modified)			
I_TNS_VEHICLEPERFORMANCE_GETBYDATE...	DS(16701)	7217D	7296	7301	7310
		7314	7318	7322	
BATCHSIZE	I(10,0)	7310			
ENDDATE	Z(26)	7314			
QUERYBYRECORDDATE...					
	N(1)	7318			
STARTDATE	Z(26)	7322			
TRUCKPCBASE	DS(16644)	7296	7301		
ITEM(128)	DS(130)	7301			
TRUCKNAME	A(128)	7301			
	VARYING				
X	I(10,0)	7296			
O_MSG	A(128)	7220D	7258M	7422M	7430M
	VARYING	7432M	7436M		
RTNVALUE	DS(16648)	7223D	7434	7437M	7445
ISUNEXPECTEDERROR	N(1)	7229D	7415M	7425	
REQUEST	A(32767)	7230D	7285M	7297M	7297
	VARYING	7307M	7307	7351	7352
RC	I(10,0)	7231D	7350M	7362	7421M
		7426			
RCVM0200	DS(4272)	7232D	7416M	7418	
MSG	DS(549)	7233D	7418M	7420	
HEADERS	DS(1538)	7234D	7337	7339M	7340M
CONTENTTYPE	A(1024)	7339			
	VARYING				
SOAPACTION	A(510)	7340			
	VARYING				
RC2	I(10,0)	7237D	7379M	7381	
ERR	I(10,0)	7238D	7345	7345	7346
		7348	7363	7364	7364
		7371	7387	7409M	
LASTERR	I(10,0)	7239D	7346	7348M	
AUTHTYPE	A(14)	7240D	7373M	7376M	7379
	VARYING				

BASIC	N(1)	7241D	7372	7375	
DIGEST	N(1)	7242D	7372	7388	7396
REALM	A(124)	7243D	7372	7375	7379
NUMATTEMPTS	I(10,0)	7244D	7347M	7369M	7369
		7380			
USERID	A(50)	7245D	7380	7382M	7390
	VARYING	7393	7398	7401	7405M
		7413M			
PASSWORD	A(50)	7246D	7380	7383M	7390
	VARYING	7393	7398	7401	7406M
		7414M			

*RNF7031 X_A1 I(10,0) 7249D
 X_A2 I(10,0) 7250D 7296 7301
 USERDATA DS(16648) 7252D 7355 7437
 Field References for subprocedure TNS_VEHICLEPERFORMANCE_GETBYDATERESPONSESTART

Field	Attributes	References (D=Defined M=Modified)
USERDATA	*(16)	7461D 7472
DEPTH	I(10,0)	7462D 7476 7483 7494
		7506
NAMESPACE	A(1024)	7463D 7484 7498 7510
	VARYING	
NAME	A(1024)	7464D 7477 7485 7496
	VARYING	7508
PATH	A(24576)	7465D 7486
	VARYING	
ATTRS(32767)	*(16)	7466D 7479 7481
TNS_VEHICLEPERFORMANCE_GETBYDATERESPONSE...	DS(16648)	7470D 7500
	BASED(USERDATA)	
VEHICLEPERFORMANCE_GETBYDATERESULT...	DS(16644)	7500

Field References for subprocedure TNS_VEHICLEPERFORMANCE_GETBYDATERESPONSEEND

Field	Attributes	References (D=Defined M=Modified)
USERDATA	*(16)	7530D 7542
*RNF7031 DEPTH	I(10,0)	7531D
NAMESPACE	A(1024)	7532D 7550
	VARYING	
NAME	A(1024)	7533D 7548
	VARYING	
*RNF7031 PATH	A(24576)	7534D
	VARYING	
VALUE	A(65535)	7535D 7552
	VARYING	
*RNF7031 ATTRS(32767)	*(16)	7536D
TNS_VEHICLEPERFORMANCE_GETBYDATERESPONSE...	DS(16648)	7540D 7551M
	BASED(USERDATA)	
TOTALROWCOUNT	I(10,0)	7551

Field References for subprocedure S1_VEHICLEPERFORMANCE_GETBYDATERESULTSTART

Field	Attributes	References (D=Defined M=Modified)
USERDATA	*(16)	7570D 7580
DEPTH	I(10,0)	7571D 7584 7591 7602
NAMESPACE	A(1024)	7572D 7592 7606
	VARYING	
NAME	A(1024)	7573D 7585 7593 7604

PATH	VARYING A(24576)	7574D	7594	
ATTRS(32767)	VARYING *(16)	7575D	7587	7589
S1_VEHICLEPERFORMANCE_GETBYDATERESULT...	DS(16644)	7579D	7608	
ENTRY	BASED(USERDATA) DS(16644)	7608		

Field References for subprocedure S1_VEHICLEPERFORMANCE_GETBYDATERESULTEND

Field	Attributes	References (D=Defined M=Modified)
USERDATA	*(16)	7631D 7642
*RNF7031 DEPTH	I(10,0)	7632D
*RNF7031 NAMESPACE	A(1024)	7633D
	VARYING	
*RNF7031 NAME	A(1024)	7634D
	VARYING	
*RNF7031 PATH	A(24576)	7635D
	VARYING	
*RNF7031 VALUE	A(65535)	7636D
	VARYING	
*RNF7031 ATTRS(32767)	*(16)	7637D
*RNF7031 S1_VEHICLEPERFORMANCE_GETBYDATERESULT...	DS(16644)	7641D
	BASED(USERDATA)	

Field References for subprocedure S1_ENTRYSTART_OF_S1_VEHICLEPERFORMANCE_GETBYDATERESULT

Field	Attributes	References (D=Defined M=Modified)
USERDATA	*(16)	7666D 7675
DEPTH	I(10,0)	7667D 7679 7686
NAMESPACE	A(1024)	7668D 7687 7699
	VARYING	
NAME	A(1024)	7669D 7680 7688 7697
	VARYING	
PATH	A(24576)	7670D 7689
	VARYING	
ATTRS(32767)	*(16)	7671D 7682 7684
S1_ENTRY	DS(16644)	7674D 7700M 7700
	BASED(USERDATA)	
X	I(10,0)	7700 7700

Field References for subprocedure S1_ENTRYEND_OF_S1_VEHICLEPERFORMANCE_GETBYDATERESULT

Field	Attributes	References (D=Defined M=Modified)
USERDATA	*(16)	7721D 7731
*RNF7031 DEPTH	I(10,0)	7722D
NAMESPACE	A(1024)	7723D 7741 7746
	VARYING	
NAME	A(1024)	7724D 7739 7744
	VARYING	
*RNF7031 PATH	A(24576)	7725D
	VARYING	
VALUE	A(65535)	7726D 7748
	VARYING	
*RNF7031 ATTRS(32767)	*(16)	7727D
S1_ENTRY	DS(16644)	7730D 7747M 7747
	BASED(USERDATA)	
DTSCANREPORTENTRY(128)...	A(128)	7747

X
Field References for subprocedure HTTP_SETCONTENTTYPEANDSOAPACTION

Field	Attributes	References (D=Defined M=Modified)
O_ADDLHDRS	A(1024) VARYING	7768D 7776M
I_PUSERDATA	*(16)	7769D 7772
HEADERS	DS(1538) BASED(I_PUSERDA+)	7772D 7777 7779
CONTENTTYPE	A(1024) VARYING	7777
SOAPACTION	A(510) VARYING	7779

Field References for subprocedure SOAPMESSAGE_ENVELOPE

Field	Attributes	References (D=Defined M=Modified)
USERDATA	*(16)	7794D 7825
*RNF7031 DEPTH	I(10,0)	7795D
*RNF7031 NAMESPACE	A(1024) VARYING	7796D
NAME	A(1024) VARYING	7797D 7809 7816
PATH	A(24576) VARYING	7798D 7804
ATTRS(32767)	*(16)	7799D 7816

Field References for subprocedure SOAPMESSAGE_FAULTSTART

Field	Attributes	References (D=Defined M=Modified)
USERDATA	*(16)	7846D 7855
*RNF7031 DEPTH	I(10,0)	7847D
*RNF7031 NAMESPACE	A(1024) VARYING	7848D
NAME	A(1024) VARYING	7849D 7861
*RNF7031 PATH	A(24576) VARYING	7850D
*RNF7031 ATTRS(32767)	*(16)	7851D
FAULT	DS(908) BASED(USERDATA)	7855D 7862
DETAIL	DS(258)	7862

Field References for subprocedure SOAPMESSAGE_FAULTTEND

Field	Attributes	References (D=Defined M=Modified)
USERDATA	*(16)	7880D 7890
*RNF7031 DEPTH	I(10,0)	7881D
*RNF7031 NAMESPACE	A(1024) VARYING	7882D
NAME	A(1024) VARYING	7883D 7896 7898
*RNF7031 PATH	A(24576) VARYING	7884D
VALUE	A(65535) VARYING	7885D 7897 7899
*RNF7031 ATTRS(32767)	*(16)	7886D
FAULT	DS(908) BASED(USERDATA)	7890D 7897M 7899M
FAULTCODE	A(256)	7897

FAULTSTRING VARYING
A(256) 7899

Field References for subprocedure SOAPMESSAGE_FAULTDETAILSTART

	Field	Attributes	References (D=Defined M=Modified)
	USERDATA	*(16)	7915D 7924
*RNF7031	DEPTH	I(10,0)	7916D
*RNF7031	NAMESPACE	A(1024)	7917D
		VARYING	
*RNF7031	NAME	A(1024)	7918D
		VARYING	
*RNF7031	PATH	A(24576)	7919D
		VARYING	
*RNF7031	ATTRS(32767)	*(16)	7920D
*RNF7031	DETAIL	DS(258)	7924D
		BASED(USERDATA)	

Field References for subprocedure SOAPMESSAGE_FAULTDETAILEND

	Field	Attributes	References (D=Defined M=Modified)
	USERDATA	*(16)	7945D 7955
*RNF7031	DEPTH	I(10,0)	7946D
*RNF7031	NAMESPACE	A(1024)	7947D
		VARYING	
	NAME	A(1024)	7948D 7961
		VARYING	
*RNF7031	PATH	A(24576)	7949D
		VARYING	
	VALUE	A(65535)	7950D 7962
		VARYING	
*RNF7031	ATTRS(32767)	*(16)	7951D
	DETAIL	DS(258)	7955D 7962M
		BASED(USERDATA)	
	HOSTNAME	A(256)	7962
		VARYING	

Field References for subprocedure SOAPMESSAGE_NULLSTART

	Field	Attributes	References (D=Defined M=Modified)
*RNF7031	USERDATA	*(16)	7978D
*RNF7031	DEPTH	I(10,0)	7979D
*RNF7031	NAMESPACE	A(1024)	7980D
		VARYING	
*RNF7031	NAME	A(1024)	7981D
		VARYING	
*RNF7031	PATH	A(24576)	7982D
		VARYING	
*RNF7031	VALUE	A(65535)	7983D
		VARYING	
*RNF7031	ATTRS(32767)	*(16)	7984D

Field References for subprocedure SOAPMESSAGE_NULLEND

	Field	Attributes	References (D=Defined M=Modified)
*RNF7031	USERDATA	*(16)	8006D
*RNF7031	DEPTH	I(10,0)	8007D
*RNF7031	NAMESPACE	A(1024)	8008D
		VARYING	
*RNF7031	NAME	A(1024)	8009D
		VARYING	

```

5761WDS V6R1M0 080215 RN          IBM ILE RPG          KARLW/WSTST1          S10DCEDB 08/19/09 13:07:17          Page 210

*RNF7031 PATH          A(24576)          8010D
                        VARYING
*RNF7031 VALUE          A(65535)          8011D
                        VARYING
*RNF7031 ATTRS(32767)   *(16)            8012D
Field References for subprocedure REPORTSSERVICEBINDING_V2_0_REGISTERLOGINCALLBACK

Field          Attributes          References (D=Defined M=Modified)
I_PLOGINPROC   *(16) PROCPTR        8035D 8039
Field References for subprocedure REPORTSSERVICEBINDING_V2_0_HTTPERROR_GETCODE

Field          Attributes          References (D=Defined M=Modified)

No references in the source.
Field References for subprocedure REPORTSSERVICEBINDING_V2_0_HTTPERROR_GETTEXT

Field          Attributes          References (D=Defined M=Modified)

No references in the source.
Field References for subprocedure REPORTSSERVICEBINDING_V2_0_SOAPERROR_GETCODE

Field          Attributes          References (D=Defined M=Modified)

No references in the source.
Field References for subprocedure REPORTSSERVICEBINDING_V2_0_SOAPERROR_GETTEXT

Field          Attributes          References (D=Defined M=Modified)

No references in the source.
Field References for subprocedure REPORTSSERVICEBINDING_V2_0_ENABLEAXIS2095

Field          Attributes          References (D=Defined M=Modified)
I_IENABLED     N(1)                8131D 8135
Field References for subprocedure REPORTSSERVICEBINDING_V2_0_SETHTTTPDEBUG

Field          Attributes          References (D=Defined M=Modified)
I_STATUS       N(1)                8152D 8160
I_FILENAME      A(500)              8153D 8163
                        VARYING
P_FILENAME      CONST               8156D 8162
Field References for subprocedure REPORTSSERVICEBINDING_V2_0_SETPROXY

Field          Attributes          References (D=Defined M=Modified)
I_SERVER        A(256)              8183D 8188
I_PORT          I(10,0)             8184D 8189
Field References for subprocedure REPORTSSERVICEBINDING_V2_0_GETENDPOINT

Field          Attributes          References (D=Defined M=Modified)

No references in the source.
Field References for subprocedure REPORTSSERVICEBINDING_V2_0_SETENDPOINT

Field          Attributes          References (D=Defined M=Modified)
I_URL           A(32767)            8224D 8228 8228
                        VARYING
Field References for subprocedure REPORTSSERVICEBINDING_V2_0_GETHOST

Field          Attributes          References (D=Defined M=Modified)

```

TMPURL	A(446) VARYING	8246D	8250M	8252	
Field References for	subprocedure	REPORTSSERVICEBINDING_V2_0_SETHOST			
Field	Attributes	References (D=Defined M=Modified)			
I_HOST	A(32767) VARYING	8267D	8274	8274	
TMPURL	A(446) VARYING	8269D	8273M	8274	8275
Field References for	subprocedure	REPORTSSERVICEBINDING_V2_0_GETPORT			
Field	Attributes	References (D=Defined M=Modified)			
TMPURL	A(446) VARYING	8293D	8297M	8299	
Field References for	subprocedure	REPORTSSERVICEBINDING_V2_0_SETPORT			
Field	Attributes	References (D=Defined M=Modified)			
I_PORT	I(10,0)	8314D	8321		
TMPURL	A(446) VARYING	8316D	8320M	8321	8322
Indicator References:					
Indicator	References (D=Defined M=Modified)				
* * * * *	E N D	O F	C R O S S	R E F E R E N C E	* * * * *

E x t e r n a l R e f e r e n c e s

Statically bound procedures:

Procedure	References			
rslvsp	1495			
_ORSTR	1519			
mitime	1562			
_ANDSTR	1511			
_XORSTR	1527			
_COMSTR	1535			
waittime	1557			
QleGetExp	1484			
XML_Parse	3472			
HTTP_COMP	5153			
HTTP_DIAG	5162			
HTTP_CLOSE	4009			
HTTP_ERROR	4021	7363	7428	
HTTPS_INIT	4042			
HTTP_XPROC	4188	7337		
HTTP_DEBUG	4923	7266	7268	7271
HTTP_XLATE	4992			
HTTP_CRASH	5170			
HTTP_XMLNS	5187	7332		
HTTP_XLATEP	5009			
HTTP_HEADER	5113			
QleActBndPgm	1465			
HTTP_URL_GET	3830	3838		
HTTP_GETAUTH	4098	7372		
HTTP_SETAUTH	4114	7390	7393	
HTTP_URL_POST	3870	3879		
HTTP_PARSEURL	3972			
HTTPS_DCM_REG	4062			
HTTPS_CLEANUP	4073			
HTTP_SETPROXY	4128	7276	7278	
HTTP_XLATEDYN	5026			
HTTP_TEMPFILE	5178			
XML_ParserFree	3536			
HTTP_REDIR_LOC	4200			
HTTP_GET_XMLTF	4506			
HTTP_SETCCSIDS	4939	7283		
HTTP_SETTABLES	4955			
BASICS1R3_f_val	1805			
XML_SetUserData	3378			
XML_ErrorString	3499			
HTTP_POST_XMLTF	4605			
BASICS1R4_f_this	992			
BASICS1R3_f_left	1724			
BASICS1R6_p_wait	2196			
XML_ParserCreate	3364			
XML_GetErrorCode	3487			
HTTP_URL_GET_RAW	3911			
HTTP_URL_GET_XML	4467	4478		
HTTP_PERSIST_GET	4744			
HTTP_USE_COOKIES	5128			
HTTP_COOKIE_FILE	5144			
WSDL2R98_Any_end	6810			
BASICS1R2_f_errno	940			
BASICS1R5_f_exist	1144			

5761WDS	V6R1M0	080215	RN	IBM ILE RPG	KARLW/WSTST1	S10DCEDB	08/19/09 13:07:17	Page	213
BASICS1R7_f_log10				1651					
BASICS1R3_f_right				1742					
BASICS1R3_f_space				1760					
BASICS1R3_f_lcase				1769					
BASICS1R3_f_ucase				1777					
BASICS1R3_f_scanR				1937					
BASICS1RD_Url_new				3068	8250	8273	8297		
				8320					
HTTP_URL_POST_RAW				3943					
HTTP_URL_POST_XML				4561	4573	7358			
HTTP_PERSIST_OPEN				4712					
HTTP_PERSIST_POST				4795					
HTTP_SETFILECCSID				4978					
HTTP_XMLRETURNPTR				5201					
HTTP_XMLSTRIPCRLF				5217					
BASICS1R1_f_newMsg				757					
BASICS1R4_p_assert				982	7989	8017			
BASICS1R4_f_caller				1003					
BASICS1R7_f_genMD5				1675					
BASICS1R3_f_ebcdic				1797					
BASICS1R3_f_center				1899					
BASICS1R3_f_alignR				1908					
XML_ParserCreateNS				3589					
HTTP_PROXY_SETAUTH				4144	7398	7401			
HTTP_PROXY_GETAUTH				4169	7375				
HTTP_LONG_PARSEURL				4331					
HTTP_URL_POST_RAW2				4375					
HTTP_URL_POST_STMF				4408	4416				
HTTP_PERSIST_CLOSE				4818					
HTTP_XML_SETCCSIDS				5068					
WSDL2R98_Any_start				6795					
BASICS1R5_f_rtvObjD				1155					
BASICS1R5_f_rtvMbrD				1168					
BASICS1R7_f_isBitOn				1642					
BASICS1R7_f_genUUID				1667					
BASICS1R7_f_genSHA1				1685					
BASICS1R3_f_replace				1786					
BASICS1R3_f_objName				1988					
BASICS1R6_f_execute				2145					
HTTP_BUILD_SOCKETADDR				3995					
HTTP_PARSE_XML_STMF				5093					
WSDL2R98_Schema_end				6770					
BASICS1R2_f_clrErrno				947					
BASICS1R2_f_strerror				954					
BASICS1R7_f_genCRC32				1695					
BASICS1R3_f_endsWith				1751					
BASICS1R3_f_wordWrap				1817					
BASICS1R3_f_editCode				1829					
BASICS1R3_f_editWord				1842					
BASICS1R8_HANDLE_new				2227					
BASICS1RD_Url_getMbr				3200					
HTTP_URL_ENCODER_NEW				4214					
HTTP_POST_STMF_XMLTF				4689					
HTTP_SET_100_TIMEOUT				5053					
BASICS1R1_f_sndPgmMsg				774					
BASICS1R1_f_rcvPgmMsg				788	7417				
BASICS1R1_p_rmvPgmMsg				805					
BASICS1R5_f_crtUsrSpc				1197					

BASICS1R5_f_dltUsrSpc	1293		
BASICS1R5_f_getObjLib	1326		
BASICS1R3_f_rmvQuotes	1881		
BASICS1R3_f_addQuotes	1890		
BASICS1R3_f_genStrCmp	1917		
BASICS1R3_f_parseDate	2015		
BASICS1R6_f_getEnvVar	2156		
BASICS1R6_f_setEnvVar	2164		
BASICS1R6_f_delEnvVar	2173		
BASICS1R8_HANDLE_null	2236		
BASICS1R8_MEMORY_free	2293		
BASICS1RD_Url_getHost	3096	8252	
BASICS1RD_Url_setHost	3105	8274	
BASICS1RD_Url_getPort	3115	8299	
BASICS1RD_Url_setPort	3124	8321	
BASICS1RD_Url_getPath	3134		
BASICS1RD_Url_setPath	3143		
BASICS1RD_Url_getFile	3153		
BASICS1RD_Url_setFile	3162		
XML_SetElementHandler	3414		
HTTP_URL_ENCODER_FREE	4289		
HTTP_MFD_ENCODER_OPEN	4841		
WSDL2R94_MultiRef_new	6188	7261	
WSDL2R94_MultiRef_put	6227		
WSDL2R94_MultiRef_get	6243		
WSDL2R98_Schema_start	6755		
WSDL2R42_getLoginData	6971	7380	
BASICS1R1_f_rtvMsgText	821	7420	
BASICS1R2_f_isApiError	924		
BASICS1R5_f_getTmpName	1181		
BASICS1R3_f_startsWith	1733		
BASICS1R3_f_rmvKeyword	1872		
BASICS1R3_f_formatDate	2006		
BASICS1RC_BasicMap_new	2812		
BASICS1RC_BasicMap_get	2848		
BASICS1RC_BasicMap_put	2902		
BASICS1RD_Url_getQuery	3172		
BASICS1RD_Url_setQuery	3181		
BASICS1RD_Url_toString	3191	8275	8322
BASICS1RE_f_dateSerial	3221		
HTTP_SELECT_COMMDRIVER	4349		
HTTP_URL_POST_STMF_XML	4645	4656	
HTTP_MFD_ENCODER_CLOSE	4910		
WSDL2R94_MultiRef_null	6268		
BASICS1R8_HANDLE_isNull	2243		
BASICS1R8_HANDLE_delete	2268		
BASICS1RA_BasicList_new	2406		
BASICS1RA_BasicList_get	2423		
BASICS1RA_BasicList_add	2497		
BASICS1RA_BasicList_set	2554		
BASICS1RC_BasicMap_null	2983		
HTTP_URL_ENCODER_ADDVAR	4232		
HTTP_URL_ENCODER_GETPTR	4250		
HTTP_URL_ENCODER_GETSTR	4271		
HTTP_MFD_ENCODER_ADDVAR	4859		
WSDL2R98_Any_initialize	6786		
BASICS1R1_f_rcvPgmMsgObj	876		
BASICS1R5_f_rtvUsrSpcPtr	1212		

BASICS1R6_f_getNumEnvVar	2181	
BASICS1R8_MEMORY_freeAll	2285	
BASICS1RA_BasicList_find	2535	
BASICS1RA_BasicList_sort	2622	
BASICS1RA_BasicList_null	2655	
BASICS1RB_BasicStack_new	2695	
BASICS1RB_BasicStack_pop	2721	
BASICS1RC_BasicMap_clone	2820	
BASICS1RC_BasicMap_clear	2967	
XML_SetEndElementHandler	3426	
XML_GetCurrentLineNumber	3511	
HTTP_MFD_ENCODER_ADDSTMF	4896	
WSDL2R94_MultiRef_isData	6218	7816
WSDL2R94_MultiRef_delete	6261	7442
WSDL2R94_MultiRef_isNull	6275	7441
BASICS1R1_p_setAppMsgFile	748	
BASICS1R1_f_rtvMsgTextRaw	844	
BASICS1R2_f_newApiErrCode	916	
BASICS1R2_f_cvtErrnoToMsg	962	
BASICS1R5_f_getUniqueName	1188	
BASICS1R5_f_getUsrSpcSize	1223	
BASICS1R5_f_chgUsrSpcSize	1257	
BASICS1R3_f_rplUnPrtChars	1997	
BASICS1R3_f_formatMemSize	2042	
BASICS1R8_HANDLE_isTypeOf	2251	
BASICS1R8_MEMORY_allocate	2276	
BASICS1R9_Display_getMode	2323	2366
BASICS1R9_Display_getRows	2344	
BASICS1RA_BasicList_clone	2415	
BASICS1RA_BasicList_clear	2639	
BASICS1RB_BasicStack_peek	2711	
BASICS1RB_BasicStack_push	2731	
BASICS1RB_BasicStack_null	2773	
BASICS1RC_BasicMap_remove	2828	
BASICS1RC_BasicMap_delete	2975	
BASICS1RC_BasicMap_isNull	2990	
BASICS1RD_Url_getProtocol	3077	
BASICS1RD_Url_setProtocol	3086	
HTTP_URL_ENCODER_ADDVAR_S	4305	
HTTP_MFD_ENCODER_ADDVAR_S	4877	
WSDL2R94_MultiRef_release	6253	
WSDL2R98_Marshaller_toAny	6538	
WSDL2R98_Marshaller_toInt	6594	7310
BASICS1R1_f_rtvMsgHelpText	832	
BASICS1R1_f_cvtRcvMsgToMsg	866	7418
BASICS1R5_f_activateSrvPgm	1304	
BASICS1RA_BasicList_remove	2605	
BASICS1RA_BasicList_delete	2647	
BASICS1RA_BasicList_isNull	2662	
BASICS1RB_BasicStack_clone	2703	
BASICS1RB_BasicStack_clear	2757	
BASICS1RC_BasicMap_getSize	2940	
BASICS1RC_BasicMap_getNumE	2959	
XML_SetStartElementHandler	3420	
XML_GetCurrentColumnNumber	3524	
WSDL2R98_Marshaller_toLong	6610	
WSDL2R98_Marshaller_toDate	6682	
WSDL2R98_Marshaller_toTime	6691	

WSDL2R98_Schema_initialize	6746	
BASICS1R8_HANDLE_getPointer	2260	
BASICS1RA_BasicList_getType	2479	
BASICS1RA_BasicList_getSize	2488	
BASICS1RA_BasicList_addNull	2509	
BASICS1RA_BasicList_setNull	2567	
BASICS1RA_BasicList_getNumE	2614	
BASICS1RB_BasicStack_delete	2765	
BASICS1RB_BasicStack_isNull	2780	
XML_SetCharacterDataHandler	3453	
WSDL2R98_UnMarshaller_toAny	6328	
WSDL2R98_UnMarshaller_toInt	6384	7552
WSDL2R98_Marshaller_toFloat	6586	
WSDL2R98_Marshaller_toShort	6602	
WSDL2R98_Marshaller_toToken	6719	
BASICS1R5_f_getUsrSpcAutoExt	1235	
BASICS1R5_f_chgUsrSpcAutoExt	1269	
BASICS1R5_f_getProcPtrByName	1315	
BASICS1R3_f_genStrCmpIgnCase	1927	
BASICS1R3_f_containsStringAt	1978	
BASICS1R6_f_getEnvVarByIndex	2188	
BASICS1R9_Display_getColumns	2353	
BASICS1RA_BasicList_isTypeOf	2469	
BASICS1RA_BasicList_isSorted	2631	
BASICS1RB_BasicStack_getSize	2741	
BASICS1RB_BasicStack_getNumE	2749	
BASICS1RC_BasicMap_getString	2860	
BASICS1RC_BasicMap_putString	2916	
WSDL2R98_UnMarshaller_toLong	6400	
WSDL2R98_UnMarshaller_toDate	6472	
WSDL2R98_UnMarshaller_toTime	6480	
WSDL2R98_Marshaller_toSchema	6546	
WSDL2R98_Marshaller_toString	6554	7302
WSDL2R98_Marshaller_toDouble	6578	
WSDL2R98_Marshaller_toAnyURI	6728	
BASICS1R1_f_rtvMsgHelpTextRaw	855	
BASICS1R3_f_getLengthEditCode	1853	
BASICS1R3_f_getLengthEditWord	1863	
BASICS1R9_Display_supportsDS4	2330	
BASICS1R9_Display_supportsGUI	2337	
BASICS1RA_BasicList_getString	2434	
BASICS1RA_BasicList_addString	2517	
BASICS1RA_BasicList_setString	2576	
BASICS1RC_BasicMap_getInteger	2870	
BASICS1RC_BasicMap_putInteger	2929	
BASICS1RC_BasicMap_KeySet_new	2998	
XML_SetUnknownEncodingHandler	3559	
WSDL2R94_MultiRef_isReference	6209	7479 7587 7682
WSDL2R94_MultiRef_endCallback	6296	
WSDL2R98_UnMarshaller_toFloat	6376	
WSDL2R98_UnMarshaller_toShort	6392	
WSDL2R98_UnMarshaller_toToken	6504	
WSDL2R98_Marshaller_toAnyType	6530	
WSDL2R98_Marshaller_toBoolean	6562	7318
WSDL2R98_Marshaller_toDecimal	6570	
BASICS1R2_f_cvtApiErrCodeToMsg	932	
BASICS1R5_f_getUsrSpcInitValue	1246	
BASICS1R5_f_chgUsrSpcInitValue	1281	

BASICS1RA_BasicList_getPointer	2443		
BASICS1RA_BasicList_addPointer	2526		
BASICS1RA_BasicList_findString	2545		
BASICS1RA_BasicList_setPointer	2586		
BASICS1RC_BasicMap_containsKey	2838		
XML_SetEndNamespaceDeclHandler	3629		
WSDL2R98_UnMarshaller_toSchema	6336		
WSDL2R98_UnMarshaller_toString	6344		
WSDL2R98_UnMarshaller_toDouble	6368		
WSDL2R98_UnMarshaller_toAnyURI	6512		
WSDL2R98_Marshaller_toDateTime	6674	7314	7322
WSDL2R98_Marshaller_toLanguage	6710		
WSDL2R98_Marshaller_toDuration	6737		
BASICS1RA_BasicList_getListType	2452		
BASICS1RA_BasicList_setAutoSort	2596		
BASICS1RA_BasicList_isNullValue	2670		
WSDL2R94_MultiRef_startCallback	6282		
WSDL2R98_UnMarshaller_toAnyType	6320	7748	
WSDL2R98_UnMarshaller_toBoolean	6352		
WSDL2R98_UnMarshaller_toDecimal	6360		
BASICS1R3_f_cvtBufferToHexString	1947		
BASICS1R3_f_cvtHexStringToBuffer	1956		
BASICS1RA_BasicList_isListTypeOf	2460		
BASICS1RC_BasicMap_getKeyByIndex	2891		
BASICS1RC_BasicMap_KeySet_delete	3022		
XML_SetStartNamespaceDeclHandler	3610		
WSDL2R94_MultiRef_getEndCallback	6202		
WSDL2R98_UnMarshaller_toDateTime	6464		
WSDL2R98_UnMarshaller_toLanguage	6496		
WSDL2R98_UnMarshaller_toDuration	6520		
BASICS1R3_f_containsInvalidCharAt	1967		
BASICS1RC_BasicMap_getSizeByIndex	2950		
BASICS1RC_BasicMap_KeySet_getNext	3014		
WSDL2R98_Marshaller_toUnsignedInt	6618		
BASICS1RC_BasicMap_getValueByIndex	2880		
BASICS1RC_BasicMap_KeySet_getFirst	3006		
WSDL2R94_MultiRef_getStartCallback	6195		
WSDL2R98_Marshaller_toUnsignedLong	6634		
WSDL2R98_Marshaller_toBase64Binary	6701		
WSDL2R98_UnMarshaller_toUnsignedInt	6408		
WSDL2R98_Marshaller_toUnsignedShort	6626		
WSDL2R98_UnMarshaller_toUnsignedLong	6424		
WSDL2R98_UnMarshaller_toBase64Binary	6488		
WSDL2R98_UnMarshaller_toUnsignedShort	6416		
WSDL2R98_Marshaller_toNegativeInteger	6658		
WSDL2R98_Marshaller_toPositiveInteger	6666		
WSDL2R98_UnMarshaller_toNegativeInteger	6448		
WSDL2R98_UnMarshaller_toPositiveInteger	6456		
WSDL2R98_Marshaller_toNonNegativeInteger	6642		
WSDL2R98_Marshaller_toNonPositiveInteger	6650		
ReportsServiceBinding_v2_0_supplyLoginData			

	246
WSDL2R98_UnMarshaller_toNonNegativeInteger	
	6432
WSDL2R98_UnMarshaller_toNonPositiveInteger	
	6440
BASICS1R7_f_clcLengthOfRPGIntFieldFromBufSize	
	1659

Imported fields:

Field	Attributes	Defined
No references in the source.		

Exported fields:

Field	Attributes	Defined
No references in the source.		

* * * * * E N D O F E X T E R N A L R E F E R E N C E S * * * * *

M e s s a g e S u m m a r y

Msg id	Sv	Number	Message text
*RNF5409	00	13	The prototyped call returns a value which is lost when CALLP is used.
*RNF7031	00	1034	The name or indicator is not referenced.
*RNF0637	30	1	An operand was expected but was not found; specification is ignored.
*RNF5410	30	3	The prototype for the call is not defined.
*RNF7030	30	4	The name or indicator is not defined.
*RNF7503	30	8	Expression contains an operand that is not defined.
* * * * *			
E N D O F M E S S A G E S U M M A R Y * * * * *			

F i n a l S u m m a r y

Message Totals:

Information	(00)	:	1047
Warning	(10)	:	0
Error	(20)	:	0
Severe Error	(30+)	:	16

Total	:	1063

Source Totals:

Records	:	8330
Specifications	:	4611
Data records	:	0
Comments	:	2958

* * * * * E N D O F F I N A L S U M M A R Y * * * * *
Compilation stopped. Severity 30 errors found in program.
* * * * * E N D O F C O M P I L A T I O N * * * * *