

```

1.00 - // %CSTD=====*
```

2.00 -	// * Application. : ARC	PRD:ARCAD		*
3.00 -	// * Component. . : ARM090R6		Type: RPGLE	*
4.00 -	// =====*			
5.00 -	// * Sub-system . : CCM CM - Change mngmt = Skipper - Gestion de			*
6.00 -	// * Function . . : OBJ Sources/Objets en version			*
7.00 -	// * Sub-function : VER Récup. info version d'un obj., récup de			*
8.00 -	// *%S=====*			*
9.00 -	// * Description of functions:			*
10.00 -	// *			*
11.00 -	// * Display the versions of a component			*
12.00 -	// *%E=====*			*
13.00 -	// * AUTHOR: PMAGNE	10/02/1995 15:13	3.01.A	*
14.00 -	// * MODIFS: 02 GILBERT	05/09/2009 08:50	08.10.02 MR 09/00328	*
15.00 -	// * Log the type changing for a component			*
16.00 -	// * 01 MV	04/07/2000 10:58	6.04.F MR 00/00475	*
17.00 -	// * Management of ILE components			*
18.00 -	// %ECSTD=====*			*

```

19.00 - Ctl-Opt DECEDIT('0,') DATEDIT(*DMY.);
20.00 - Dcl-F ARCAPPF1 Keyed;
21.00 - Dcl-F ARMSOLF Keyed;
22.00 - Dcl-F ARMMOBL7 Keyed;
23.00 - Dcl-F ARMVERL2 Keyed;
24.00 - Dcl-F AROBJTYP Keyed;
25.00 - Dcl-F ARMAHFF1 Keyed Rename(ARMAHFF1:ARMAHFFF);
26.00 -
27.00 - Dcl-F ARM090D1 WORKSTN
28.00 - SFILE(M090S1:LIG1);
29.00 - //
30.00 - Dcl-Ds SRC_Enr ExtName('ARDSRCF1') End-Ds;
31.00 - // *****
32.00 - // For reading alias names
33.00 - /COPY ARD303M1_P
34.00 - // For reading IFS names
35.00 - /COPY ART407M1_P
36.00 - // For convert IFS names
37.00 - /COPY ART409M2_P
38.00 - // To Add quotes
39.00 - /COPY ARD151M1_P
40.00 - // To retrieve info. for a component
41.00 - /COPY ARD166M1_P
42.00 - // Prototypes of imported procedure
43.00 - /COPY ARD167M2_P
44.00 - // Command for DTC
45.00 - /COPY ART562M1_P
46.00 - // *****
47.00 - Dcl-S CST Char(30) DIM(11);
48.00 - Dcl-S CD Char(80) DIM(13) CTDATA PERRCD(1);
49.00 - Dcl-Ds CMDDs;
50.00 - CMD Char(1) DIM(2000);
50.01 - End-Ds;
51.00 - //
52.00 - Dcl-S WRet_Src Char(1);
53.00 - // -----+
54.00 - // D D S !
55.00 - // -----+
56.00 - Dcl-Ds *n;
57.00 - CD1 Char(80) Pos(1);
58.00 - MSAPP Char(3) Pos(15);
59.00 - ZSTST Char(1) Pos(26);
60.00 - ZSVERS Char(8) Pos(38);
60.01 - End-Ds;
61.00 - Dcl-Ds MSGDT1;
62.00 - WVERS Char(8) Pos(1);
63.00 - WOBJ Char(10) Pos(9);
63.01 - End-Ds;
64.00 - //
65.00 - Dcl-S Wsrc Char(21);
66.00 - Dcl-S Wpass Char(1) Inz(' ');
67.00 - Dcl-S Wmsgf Char(10) Inz(' ');
68.00 - Dcl-S Wmsgid Char(7) Inz(' ');
69.00 - Dcl-S Wmsgdta Char(2000) Inz(' ');
70.00 - Dcl-S Wmsgtyp Char(10) Inz(' ');
71.00 - Dcl-S Wret122c4 Char(1) Inz(' ');
72.00 - Dcl-S WobjLib Char(10);
73.00 - Dcl-S Walias VarChar(256);
74.00 - Dcl-S WaliasApo Char(256);

```

```

75.00 -      Dcl-S isDTC          Ind;
76.00 -      Dcl-S LRnmMso_A   Ind;
77.00 -      Dcl-S LRnmMso_S   Ind;
78.00 -      Dcl-S LRnmMob_A   Ind;
79.00 -      Dcl-S LRnmMob_S   Ind;
80.00 -      Dcl-S LChgTyp     Ind;
81.00 -      Dcl-S LTxtGen     Ind;
82.00 -
84.00 -      // IFS name and its length
85.00 -      Dcl-S WIFSNAM     Char(755);
86.00 -      Dcl-S WIFSNAMLG   Uns(5);
87.00 -      Dcl-S WIFSNomVer   Char(1);           // Version name
88.00 -      Dcl-S WIFSNomCas   Char(1);           // Type Case (L/U)
89.00 -      // Name of parameters to substitute in command to manage IFS
90.00 -      Dcl-S WIFSPar0     Char(10);           // Classical param
91.00 -      Dcl-S WIFSParT     Char(10);           // IFS Type
92.00 -      Dcl-S WIFSPar1     Char(10);           // IFS Param
93.00 -      //
94.00 -      Dcl-S IFSOBJLg     Uns(5);
95.00 -      Dcl-S TypeNom      Char(1);
96.00 -      // Variable to manage display of a component which don't have
97.00 -      // a base version
98.00 -      Dcl-S cpsttrt      Packed(3:0);       //nbr of processed c
99.00 -      Dcl-S cpstsup      Packed(3:0);       // nbr of c. with
100.00 -      //                                           deleted versions
101.00 -      Dcl-S WEtudSrc     Char(1);
102.00 -      Dcl-S WEtudObj     Char(1);
103.00 -      Dcl-S WReadMSO     Char(1);
104.00 -      Dcl-S WReadMOB     Char(1);
105.00 -      Dcl-S WEOfMSO     Char(1);
106.00 -      Dcl-S WEOfMOB     Char(1);
107.00 -      Dcl-S Wppi        Char(1);
108.00 -      Dcl-S WEnAjout     Char(1);
109.00 -      Dcl-S WVerSupp     Char(1);
110.00 -      //
111.00 -      Dcl-S PRefr        Char(1)           Inz('0');           // Refr = '0'
112.00 -      //
113.00 -      // For comparison Date and Hour for transfer to prod.
114.00 -      Dcl-Ds WVRMEXDT;
115.00 -          WVRMEXD      Like(VRMEXD);
116.00 -          WVRMEXT      Like(VRMEXT);
116.01 -      End-Ds;
117.00 -      //
118.00 -      // For comparison Date and Hour for transfer to prod.
119.00 -      Dcl-Ds WPRMEXDT;
120.00 -          WPRMEXD      Like(VRMEXD);
121.00 -          WPRMEXT      Like(VRMEXT);
121.01 -      End-Ds;
122.00 -
123.00 -      Dcl-S KAppl        Like(VrApp);
124.00 -      Dcl-S KVers        Like(VrVers);
125.00 -      Dcl-S CVerSv       Like(MSVERS);
126.00 -      Dcl-S CVerSvO      Like(MOVERS);
126.01 -      Dcl-S CLETYP       Char(7);
126.02 -      Dcl-S RIEN         Char(132);
126.03 -      Dcl-S LEN          Packed(15:5);
126.04 -      Dcl-S LEN1         Packed(15:5);
126.05 -      Dcl-S WTYPE        Char(1);
126.06 -      Dcl-S ATag         Char(14);
126.07 -      Dcl-S LIG1         Packed(4:0);
126.08 -      Dcl-S TYPCMD       Char(1);
126.09 -      Dcl-S pAToArrStr    Pointer;
126.10 -      Dcl-S AToArrStr     Char(65535)       Based(pAToArrStr);
126.11 -      Dcl-S WDAT          Char(8);
126.12 -      Dcl-S XDAT          Char(10);           // TO DATE
126.13 -      Dcl-S FMTO         Char(7);
126.14 -      Dcl-S FMTD         Char(7);
126.15 -      Dcl-S SEP          Char(7);
126.16 -      Dcl-S RETDAT        Char(1);
126.17 -      Dcl-S pAFrmArrStr   Pointer;
126.18 -      Dcl-S AFrmArrStr     Char(65535)       Based(pAFrmArrStr);
126.19 -      Dcl-S WMSVERS        Char(8);
126.20 -      Dcl-S RTVLIV        Char(4);           //Source in version.
126.21 -      Dcl-S CHKVERS       Char(5);           // Comp. in vers.
126.22 -      Dcl-S WMSCPV        Packed(3:0);
126.23 -      Dcl-S RTSRCL        Char(10);
126.24 -      Dcl-S RTSRCF        Char(10);

```

```

126.25 -      Dcl-S RTSRCT          Char(10);
126.26 -      Dcl-S RTOBJT          Char(8);
126.27 -      Dcl-S WVRVER          Char(8);
126.28 -      Dcl-S ACT            Char(1);
126.29 -      Dcl-S WRET            Char(1);
126.30 -      Dcl-S VERLIB          Char(10);
126.31 -      Dcl-S VERL1          Char(1);
126.32 -      Dcl-S MSGID          Char(7);
126.33 -      Dcl-S CMDLIN          Char(80);
126.34 -      Dcl-S REF            Char(8);
126.35 -      Dcl-S RESULT1        Char(11);
126.36 -      Dcl-S RESULT          Char(21);
126.37 -      Dcl-S WSOM            Packed(4:0);
126.38 -      Dcl-S WDIV            Packed(4:0);
126.39 -      Dcl-S REST            Packed(2:0);
126.40 -      Dcl-S DEBPAG          Packed(4:0);
126.41 -      // Prototype for àRMVPGMQ
126.42 -      Dcl-Pr Pgm_àRMVPGMQ ExtPgm('àRMVPGMQ') End-Pr;
126.43 -      // Prototype for ARCMD
126.44 -      Dcl-Pr Pgm_ARCMD ExtPgm('ARCMD');
126.45 -      TYPCMD          Char(1);
126.46 -      End-Pr;
126.47 -      // Prototype for ARCVTDAT
126.48 -      Dcl-Pr Pgm_ARCVTDAT ExtPgm('ARCVTDAT');
126.49 -      WDAT            Char(8);
126.50 -      XDAT            Char(10);
126.51 -      FMTO            Char(7);
126.52 -      FMTD            Char(7);
126.53 -      SEP            Char(7);
126.54 -      RETDAT          Char(1);
126.55 -      End-Pr;
126.56 -      // Prototype for ARM090R6
126.57 -      Dcl-Pr Pgm_ARM090R6 ExtPgm('ARM090R6');
126.58 -      POBJ            Char(10);
126.59 -      MBRTYP          Char(10);
126.60 -      PAPP            Char(3);
126.61 -      DspArchVer      Char(4);
126.62 -      IFSOBJ          Char(755);
126.63 -      End-Pr;
126.64 -      // Procedure interface for ARM090R6
126.65 -      Dcl-Pi Pgm_ARM090R6;
126.66 -      POBJ            Char(10);
126.67 -      MBRTYP          Char(10);
126.68 -      PAPP            Char(3);
126.69 -      DspArchVer      Char(4);
126.70 -      IFSOBJ          Char(755);
126.71 -      End-Pi;
126.72 -      // Prototype for ARM122C4
126.73 -      Dcl-Pr Pgm_ARM122C4 ExtPgm('ARM122C4');
126.74 -      Wmsgf            Char(10);
126.75 -      Wmsgid          Char(7);
126.76 -      Wmsgdta          Char(2000);
126.77 -      Wmsgtyp          Char(10);
126.78 -      Wret122c4        Char(1);
126.79 -      End-Pr;
126.80 -      // Prototype for ARM122R1
126.81 -      Dcl-Pr Pgm_ARM122R1 ExtPgm('ARM122R1');
126.82 -      ZObj            Char(10);
126.83 -      ZSrcT            Char(10);
126.84 -      PAPP            Char(3);
126.85 -      ZSREFR          Char(1);
126.86 -      WMSVERS          Char(8);
126.87 -      RTVLIV          Char(4);
126.88 -      CHKVERS          Char(5);
126.89 -      WMSCPv          Packed(3:0);
126.90 -      RTSRCL          Char(10);
126.91 -      RTSRCF          Char(10);
126.92 -      RTSRCT          Char(10);
126.93 -      RTOBJT          Char(8);
126.94 -      WVRVER          Char(8);
126.95 -      ACT            Char(1);
126.96 -      WRET            Char(1);
126.97 -      End-Pr;
126.98 -      // Prototype for ARSNDMSG
126.99 -      Dcl-Pr Pgm_ARSNDMSG ExtPgm('ARSNDMSG');
127.00 -      PGMQ            Char(10);
127.01 -      MSGID            Char(7);

```

```

127.02 -      OPT          Char(2);
127.03 - End-Pr;
127.04 - // Prototype for ARSNDMSG
127.05 - Dcl-Pr Pgm_ARSNDMSG_001 ExtPgm('ARSNDMSG');
127.06 -      PGMQ          Char(10);
127.07 -      MSGID         Char(7);
127.08 -      RIEN          Char(132);
127.09 - End-Pr;
127.10 - // Prototype for CLRPGMQ
127.11 - Dcl-Pr Pgm_CLRPGMQ ExtPgm('CLRPGMQ');
127.12 -      PGMQ          Char(10);
127.13 - End-Pr;
127.14 - // Prototype for QCMDEXC
127.15 - Dcl-Pr Pgm_QCMDEXC ExtPgm('QCMDEXC');
127.16 -      CMD           Char(1)          Dim(2000);
127.17 -      LEN           Packed(15:5);
127.18 - End-Pr;
127.19 -
128.00 - // *W_LibObj      S          21A
129.00 -
130.00 - // -----+
131.00 - //      P A R A M E T E R S      !
132.00 - // -----+
139.00 - // -----+
140.00 - //      K E Y S                  !
141.00 - // -----+
144.00 - //
167.00 - //
175.00 - //
179.00 - // -----+
180.00 - //      P R O C E S S            !
181.00 - // -----+
182.00 - If DspArchVer = '*YES';
183.00 -     *In70 = *Off;
184.00 - Else;
185.00 -     *In70 = *On;
186.00 - EndIf;
187.00 - //
188.00 - %Subst(PGMQ:1:8) = 'ARM090R6';
189.00 - RIEN = *Blanks;
190.00 - LEN = 2000;
191.00 - LEN1 = 501;
192.00 - // LOAD CONSTANTS
193.00 - Exsr RTVCST;
194.00 - // Retrive Application text
195.00 - Chain PAPP ARCAPPF1;
195.01 - *IN50 = not %Found;
196.00 - //
197.00 - // Indicator 46 ON : it's an IFS name
198.00 - OBJ = POBJ;
199.00 - If OBJ = '*IFS';
200.00 -     *In64 = '1';
201.00 - // Build object IFS ID
202.00 -     TypeNom = ' ';
203.00 -     IfsObjLg = %CheckR(' ' : IFSOBJ);
203.01 - // *CVTWNG : %Found is not updated by %CheckR
204.00 -     OBJ = IFS_IDGEN('JOB':TypeNom:
205.00 -         %Addr(IFSOBJ):IfsObjLg);
206.00 -     ZNIFS = IFSOBJ; // Visible name
207.00 - Else;
208.00 -     *In64 = '0';
209.00 - EndIf;
210.00 - //
211.00 - // Read object type of a component (it also find it
212.00 - // if just created in another version).
213.00 - Clear Src_Enr;
214.00 - WRet_Src = REP_SRCCPS(PAPP:'0':
215.00 -     '*NONE':OBJ:MBRTYP:
216.00 -     *OMIT:%Addr(Src_Enr));
217.00 - //
218.00 - WEtudSrc = '0';
219.00 - WEtudObj = '0';
220.00 - WTYPE = MBRTYP;
221.00 - If WTYPE <> '*' // SRC
222.00 -     and WTYPE <> '_'; // SRC
223.00 -     *IN63 = *ON;
224.00 -     WEtudSrc = '1';

```

```

225.00 -         If WRet_Src <> 'E';
226.00 -             CLETYP = SCOBJT;
227.00 -             WEtudObj = '1';
228.00 -         EndIf;
229.00 -     //
230.00 -     Else;
231.00 -         WEtudSrc = '0';
232.00 -         WEtudObj = '1';
233.00 -         *IN63 = *OFF;
234.00 -         CLETYP = MBRTYP;
235.00 -     EndIf;
236.00 -
237.00 -     // Object library:
238.00 -     If WRet_Src <> 'E';
239.00 -         WobjLib = SCOBJL;
240.00 -     EndIf;
241.00 -     // Alias / DTC:
242.00 -     wAlias = Als_SrcMToAls( PAPP : OBJ : MBRTYP); // (Variable leng
243.00 -
244.00 -     If          %len( wAlias      ) > *zero;
245.00 -         wAliasApo      =      wAlias; // (Fixed length)
246.00 -         Str_AddApo( %addr( wAliasApo ) :
247.00 -             %len( wAliasApo ) : // * Maximum
248.00 -             %len( wAlias      ) : '1'); // * Actual / Always
249.00 -     EndIf;
250.00 -
251.00 -     Chain CLETYP AROBJTYP; // OBJTYPE
252.00 -     isDTC = %found and OBJDTC = 'Y';
253.00 -
254.00 -     // ** SUBFILE INITIALIZATION
255.00 -     //
255.01 -     DoU ATag <> 'BEGIN' and ATag <> 'DISPLAY';
256.01 -         If ATag = 'BEGIN' or ATag = *Blanks;
256.02 -             ATag = *Blanks;
257.00 -             WRANG = 1;
258.00 -             LIG1 = 0;
259.00 -             *IN80 = *OFF;
260.00 -             *IN92 = '1';
261.00 -             Write M090C1;
262.00 -             *IN92 = '0';
263.00 -         //
264.00 -         // Static Load of subfile
265.00 -         Exsr RMPSF;
266.00 -         //
267.00 -         // ** DISPLAY SUBFILE
268.00 -         //
268.01 -         EndIf;
269.01 -         // branch when ATag = 'DISPLAY'
269.02 -         ATag = *Blanks;
270.00 -         // MANAGE messages subfile
271.00 -         *IN81 = '1';
272.00 -         Write MSGSFLC;
273.00 -         *IN81 = '0';
274.00 -         Write MSGSFLC;
275.00 -         *IN91 = '1';
276.00 -         Write M090C1;
277.00 -         *IN91 = '0';
278.00 -         If LIG1 = 0;
279.00 -             Write M090D2;
280.00 -         Else;
281.00 -             *IN90 = '1';
282.00 -             Write M090C1;
283.00 -             *IN90 = '0';
284.00 -         EndIf;
285.00 -         Exfmt M090D1;
286.00 -         *IN80 = '0';
287.00 -         //
288.01 -         Pgm_CLRPGMQ(PGMQ);
290.00 -         // KEY FIELDS
290.01 -         If *In03;
291.01 -             ATag = 'END1';
291.02 -         Endif;
291.03 -         If ATag = *blanks;
292.00 -         // SYSTEM Command
293.00 -         If *IN21 = *ON;
294.01 -             TYPCMD = '2';
294.02 -             Pgm_ARCMD(TYPCMD);

```

```

296.01 -           ATag = 'DISPLAY';
296.02 -           Iter;
297.00 -       EndIf;
298.00 -       //
298.01 -           If *In12;
299.01 -               ATag = 'END1';
299.02 -           Endif;
299.03 -       EndIf;
299.04 -       If ATag = *blanks;
300.00 -       // PROMPT = F4
301.00 -           If *IN04 = *ON;
302.00 -               *IN44 = '1';
303.00 -           Else;
304.00 -               *IN44 = '0';
305.00 -           EndIf;
306.00 -       //
307.00 -       // Archiving = F7
308.00 -           If *In07 = *On;
309.00 -               If *In70 = *Off;
310.00 -                   *In70 = *On;
311.00 -               Else;
312.00 -                   *in70 = *Off;
313.00 -               EndIf;
314.01 -           ATag = 'BEGIN';
314.02 -           Iter;
315.00 -       EndIf;
316.00 -       //
317.00 -           If LIG1 = 0;
317.01 -               ATag = 'DISPLAY';
317.02 -               Iter;
317.03 -           EndIf;
318.00 -       //
319.00 -       // ** Test of the Options
320.00 -           DoW '1';
321.00 -               ReadC M090S1;
321.01 -               *IN50 = %Eof;
322.00 -               If *IN50 = '1';
322.01 -                   ATag = 'DISPLAY';
322.02 -                   Leave;
322.03 -               EndIf;
323.00 -               If OPT <> ' ';
324.00 -                   *IN59 = '0';
325.00 -           // ** STANDARD OPTIONS
326.00 -           // FONCTIONALITIES
326.01 -               DoU '1';
327.00 -                   *IN59 = (OPT = '1 ');
327.01 -                   If OPT = '1 ';
327.02 -                       Exsr SBR1;
327.03 -                       Leave;
327.04 -                   EndIf;
328.00 -                   *IN59 = (OPT = ' 1');
328.01 -                   If OPT = ' 1';
328.02 -                       Exsr SBR1;
328.03 -                       Leave;
328.04 -                   EndIf;
329.00 -           // SOURCE
330.00 -                   *IN59 = (OPT = '5 ');
330.01 -                   If OPT = '5 ';
330.02 -                       Exsr SBR5;
330.03 -                       Leave;
330.04 -                   EndIf;
331.00 -                   *IN59 = (OPT = ' 5');
331.01 -                   If OPT = ' 5';
331.02 -                       Exsr SBR5;
331.03 -                       Leave;
331.04 -                   EndIf;
332.00 -           //
333.00 -                   *IN59 = (OPT = '6 ');
333.01 -                   If OPT = '6 ';
333.02 -                       Exsr SBR6;
333.03 -                       Leave;
333.04 -                   EndIf;
334.00 -                   *IN59 = (OPT = ' 6');
334.01 -                   If OPT = ' 6';
334.02 -                       Exsr SBR6;
334.03 -                       Leave;
334.04 -                   EndIf;

```

```

335.00 - //
336.00 - *IN59 = (OPT = '7 ');
336.01 - If OPT = '7 ';
336.02 - Exsr SBR7;
336.03 - Leave;
336.04 - EndIf;
337.00 - *IN59 = (OPT = '7');
337.01 - If OPT = '7';
337.02 - Exsr SBR7;
337.03 - Leave;
337.04 - EndIf;
338.00 - //
339.00 - *IN59 = (OPT = '14');
339.01 - If OPT = '14';
339.02 - Exsr SBRL4;
339.03 - Leave;
339.04 - EndIf;
340.00 - //
341.00 - *IN59 = (OPT = '20');
341.01 - If OPT = '20';
341.02 - Exsr SBR20;
341.03 - Leave;
341.04 - EndIf;
342.00 - EndDo;
343.00 - // INVALID OPTION
344.00 - If *IN59 = '0';
345.00 - *IN80 = '1';
346.01 - MSGID = 'MSG0467';
346.02 - Pgm_ARSNDMSG(PGMQ : MSGID : OPT);
350.00 - EndIf;
351.00 - //
352.00 - EndIf; // OPT <> ' '
353.00 - //
354.00 - Exsr POSCUR;
355.00 - //
356.00 - OPT = *Blanks;
357.00 - pAtoArrStr = %Addr(*IN(61));
357.01 - %Subst(AtoArrStr:1:2) = IN6162;
358.00 - *IN(65) = IN65;
359.00 - *IN(66) = IN66;
360.00 - Update M090S1;
361.00 - //
362.00 - EndDo; // DO
362.01 - If ATag = 'DISPLAY';
362.02 - Iiter;
362.03 - EndIf;
363.00 - //
364.01 - ATag = 'DISPLAY';
364.02 - Iiter;
364.03 - EndIf;
364.04 - EndDo;
365.00 - //
366.01 - // branch when ATag = 'END1'
366.02 - ATag = *Blanks;
367.00 - *INLR = '1';
368.00 - // =====*
369.00 - // Static load of versions for a component
370.00 - // =====*
371.00 - BegSr RMPSF;
372.00 - // Initialize variables to manage display of base version
373.00 - // for an added component
374.00 - zstst = ' ';
375.00 - cpsttrt = 0;
376.00 - cpstsup = 0;
377.00 - WEnAjout = '0';
378.00 - //
379.00 - WPRMEXDT = *HIVAL;
380.00 - // ** STATIC LOAD OF SUBFILE
381.00 - //
382.00 - // ** Matching for read with Source change and also Object change...
383.00 - //
384.00 - Obj1 = Obj;
385.00 - MbrTyp1 = MbrTyp;
386.00 - WReadMSO = '0';
387.00 - WReadMOB = '0';
388.00 - If WEtudSrc = '1';
389.00 - Setll (PAPP : PRefr : OBJ : MBRTYP) ARMMSELF;

```

```

390.00 -         WReadMSO = '1';
391.00 -     Else;
392.00 -         WEOFMSO = '1';
393.00 -     EndIf;
394.00 - //
395.00 -     If WETudObj = '1';
396.00 -         Setll (PAPP : PRefr : OBJ : CLETYP) ARMMOBL7;
397.00 -         WReadMOB = '1';
398.00 -     Else;
399.00 -         WEOFMOB = '1';
400.00 -     EndIf;
401.00 - //
402.00 - //
403.00 -     DoW '1';
404.00 - // --- Read MSO (version number descending)
405.00 -     If WReadMSO = '1';
406.00 -         ReadE (PAPP : PRefr : OBJ : MBRTYP) ARMMMSOLF;
406.01 -         *IN50 = %Eof;
407.00 -         WEOFMSO = *IN50;
408.00 - // IF VERSION = *NONE: DON'T USE THIS RECORD
409.00 -     If MSVERS = '*NONE' and WEOFMSO = '0';
410.00 -         Iiter;
411.00 -     EndIf;
412.00 -     If WEOFMSO = '0';
413.00 -         KAppl = MsApp;
414.00 -         KVers = MsVers;
415.00 -         Chain (KAppl : KVers) ARMAHFF1;
416.00 -         If %found and *In70 = *on;
417.00 -             Iiter;
418.00 -         EndIf;
419.00 -     EndIf;
420.00 -     WReadMSO = '0';
421.00 - EndIf;
422.00 - // --- READ MOB (version number descending)
423.00 -     If WReadMOB = '1';
424.00 -         ReadE (PAPP : PRefr : OBJ : CLETYP) ARMMOBL7;
424.01 -         *IN50 = %Eof;
425.00 -         WEOFMOB = *IN50;
426.00 - // IF VERSION = *NONE: DON'T USE THIS RECORD
427.00 -     If MOVERS = '*NONE' and WEOFMOB = '0';
428.00 -         Iiter;
429.00 -     EndIf;
430.00 -     If WEOFMOB = '0';
431.00 -         KAppl = MoApp;
432.00 -         KVers = MoVers;
433.00 -         Chain (KAppl : KVers) ARMAHFF1;
434.00 -         If %found and *In70 = *on;
435.00 -             Iiter;
436.00 -         EndIf;
437.00 -     EndIf;
438.00 -     WReadMOB = '0';
439.00 - EndIf;
440.00 - //
441.00 - // Exit when both are ended
442.00 -     If WEOFMSO = '1' and WEOFMOB = '1';
443.00 -         Leave;
444.00 -     EndIf;
445.00 - //
446.00 - // Find The P.P.I. of the 2 read ; '1' = MSO , '2' = 'MOB' and '0' = both
447.00 -     If WEOFMSO = '0';
448.00 -         If WEOFMOB = '0';
449.00 - // The 2 files have been read
450.00 -         Select;
451.00 - //     version in MSO and MOB
452.00 -         When MOVERS = MSVERS;
453.00 -             Wppi = '0';
454.00 -             WReadMSO = '1';
455.00 -             WReadMOB = '1';
456.00 - //     version in MOB to process first
457.00 -         When MOVERS > MSVERS;
458.00 -             Wppi = '2';
459.00 -             WReadMOB = '1';
460.00 - //     version in MSO to process first
461.00 -         Other;
462.00 -             Wppi = '1';
463.00 -             WReadMSO = '1';
464.00 -     EndSl;

```



```

465.00 - // Just MSO
466.00 -     Else;
467.00 -         Wppi = '1';
468.00 -         WReadMSO = '1';
469.00 -     EndIf;
470.00 - // Just MOB
471.00 -     Else;
472.00 -         Wppi = '2';
473.00 -         WReadMOB = '1';
474.00 -     EndIf;
475.00 - //
476.00 -     If Wppi = '0' or Wppi = '1';
477.00 -         Exsr SRMSO;
478.00 -     Else;
479.00 -         Exsr SRMOB;
480.00 -     EndIf;
481.00 - //
482.00 - // Retrieve version info.
483.00 - Chain (PAPP : ZSTST : ZSVERS) ARMVERL2;
483.01 - *IN50 = not %Found;
484.00 - WVerSupp = '1';
485.00 - // Deleted version ?
486.00 - If *IN50 = *ON;
487.00 -     *IN61 = *OFF;
488.00 -     *IN62 = *OFF;
489.00 -     WVerSupp = '1';
490.00 -     WTXT = CST(7);
491.00 -     VRDCLD = 0;
492.00 -     WDCLD = *Blanks;
493.00 -     VRMEXD = 0;
494.00 -     WMEXD = *Blanks;
495.00 - // Update cpstsup
496.00 -     cpstsup = cpstsup + 1;
497.00 - Else;
498.00 -     WVerSupp = '0';
499.00 -     If LTxtGen = *Off;
500.00 -         WTXT = VRTXT;
501.00 -     EndIf;
502.00 - //                               MOVE VRDCLD   WDAT   6
503.00 -     WDAT = %EditC(VRDCLD:'X');
504.01 -     XDAT = *BLANKS;
504.02 -     FMTO = '*YYMD  ';
504.03 -     FMTD = '*JOB8  ';
504.04 -     SEP = '*JOB   ';
504.05 -     Pgm_ARCVTDAT(WDAT : XDAT : FMTO : FMTD : SEP :
504.06 -         RETDAT);
511.00 -     WDCLD = XDAT;
512.00 -     WDAT = %EditC(VRMEXD:'X');
513.00 -     WMEXD = *Blanks;
514.01 -     XDAT = *BLANKS;
514.02 -     FMTO = '*YYMD  ';
514.03 -     FMTD = '*JOB8  ';
514.04 -     SEP = '*JOB   ';
514.05 -     Pgm_ARCVTDAT(WDAT : XDAT : FMTO : FMTD : SEP :
514.06 -         RETDAT);
521.00 -     WMEXD = XDAT;
522.00 - // Update cpsttrt
523.00 -     cpsttrt = cpsttrt + 1;
524.00 - // IF THE VERSION HAS A REFERENCE NUMBER WHICH IS LESS
525.00 - // DISPLAY IT WITH HI
526.00 -     If VRVREF <> '*CURRENT';
527.00 -         *IN61 = *ON;
528.00 -     EndIf;
529.00 - // Test Order of transfer to prod 62 ON if non ok
530.00 -     *IN62 = *OFF;
531.00 -     If VRMEXD <> 0;
532.00 -         WVRMEXD = VRMEXD;
533.00 -         WVRMEXT = VRMEXT;
534.00 -         If WVRMEXDT > WPRMEXDT;
535.00 -             *IN62 = *ON;
536.00 -         EndIf;
537.00 -         WPRMEXDT = WVRMEXDT;
538.00 -     EndIf;
539.00 - //
540.00 - EndIf;
541.00 - // Write in the subfile
542.00 - pAFrmArrStr = %Addr(*IN(61));

```

```

542.01 -      %Subst(IN6162:1:2) = %Subst(AFrMArrStr:1:2);
543.00 -      IN65 = *IN(65);
544.00 -      IN66 = *IN(66);
545.00 -      LIG1 += 1;
546.00 -      Write M090S1;
547.00 -      // If renamed
548.00 -      If LRnmMso_A;
549.00 -          Obj = Mso_JrnObj;
550.00 -          MbrTyp = Mso_CrnTyp;
551.00 -          CVerSv = MSVERS;
552.00 -          Setll (PAPP : PRefr : OBJ : MBRTYP : CVerSv)
552.01 -              ARMMOLF;
553.00 -          WEOFMSO = '0';
554.00 -          If WETudObj = '1';
555.00 -              CleTyp = MsObjT;
556.00 -              CVerSvO = MSVERS;
557.00 -              Setll (PAPP : PRefr : OBJ : CLETYP : CVerSvO)
557.01 -                  ARMMOBL7;
558.00 -              WReadMOB = '1';
559.00 -              WEOFMOB = '0';
560.00 -          Else;
561.00 -              WEOFMOB = '1';
562.00 -          EndIf;
563.00 -      EndIf;
564.00 -      //
565.00 -      If LRnmMso_S;
566.00 -          Action = Cst(3);
567.00 -          WTxt = VrTxt;
568.00 -          ZOBJ = MSSRCM;
569.00 -          ZSRCT = MSTYPE;
570.00 -          *In66 = *Off;
571.00 -          pAFrmArrStr = %Addr(*IN(61));
571.01 -          %Subst(IN6162:1:2) = %Subst(AFrMArrStr:1:2);
572.00 -          IN65 = *IN(65);
573.00 -          IN66 = *IN(66);
574.00 -          Lig1 = Lig1 + 1;
575.00 -          Write M090S1;
576.00 -      EndIf;
577.00 -      // If renamed
578.00 -      If LRnmMob_A;
579.00 -          Obj = Mob_JrnObj;
580.00 -          CleTyp = Mob_CrnTyp;
581.00 -          CVerSvO = MOVERS;
582.00 -          Setll (PAPP : PRefr : OBJ : CLETYP : CVerSvO)
582.01 -              ARMMOBL7;
583.00 -      EndIf;
584.00 -      //
585.00 -      If LRnmMob_S;
586.00 -          Action = Cst(3);
587.00 -          WTxt = VrTxt;
588.00 -          ZOBJ = MOOBJ;
589.00 -          ZSRCT = MOOBTP;
590.00 -          Lig1 = Lig1 + 1;
591.00 -          Write M090S1;
592.00 -      EndIf;
593.00 -      // If type has benn changed
594.00 -      If LChgTyp;
595.00 -          Action = Cst(2);
596.00 -          WTxt = VrTxt;
597.00 -          *In66 = *Off;
598.00 -          pAFrmArrStr = %Addr(*IN(61));
598.01 -          %Subst(IN6162:1:2) = %Subst(AFrMArrStr:1:2);
599.00 -          IN65 = *IN(65);
600.00 -          IN66 = *IN(66);
601.00 -          Lig1 = Lig1 + 1;
602.00 -          Write M090S1;
603.00 -      EndIf;
604.00 -      //
605.00 -      EndDo;
606.00 -      // -----
607.00 -      // If Not "Ajout" record in the file and the last
608.00 -      // environment Id done is not '*' :
609.00 -      // INSERTION OF BASE VERSION
610.00 -      //      WEnAjout      IFEQ      *OFF
611.00 -      //      ZSTST      ANDNE      '*'
612.00 -      // If ((WEnAjout = '0') and (zstst <> '*')) or
613.00 -      //      ((WEnAjout = '1') and (cpstsup <> 0) and

```

```

614.00 -             (cpsttrt = 0) and (zstst <> '*'));
615.00 - //
616.00 - // BASE VERSION
617.00 - VRTST = '*';
618.00 - Chain (PAPP : VRTST) ARMVERL2;
618.01 - *IN50 = not %Found;
619.00 - If *IN50 = *OFF;
620.00 - // Archived source ?
621.00 -     ZSTST = '*';
622.00 -     ZSVERS = VRVERS;
623.00 -     KAppl = PApp;
624.00 -     KVers = ZsVers;
625.00 -     Chain (KAppl : KVers) ARMAHHF1;
626.00 -     If %Found;
627.00 -         *In65 = *On;
628.00 -     Else;
629.00 -         *In65 = *Off;
630.00 -     EndIf;
631.00 -     WTXT = VRTXT;
632.00 -     WDCLD = *Blanks;
633.00 -     WMEXD = *Blanks;
634.00 -     ACTION = *Blanks;
635.00 -     ZSPGMR = *Blanks;
636.00 -     ZSREFR = *Blanks;
637.00 -     ZOBJ = OBJ;
638.00 -     ZSRCT = MBRTYP;
639.00 -     ZODLIB = WobjLib;
640.00 -
641.00 -     LIG1 += 1;
642.00 -     *IN61 = '0';
643.00 -     pAFrmArrStr = %Addr(*IN(61));
643.01 -     %Subst(IN6162:1:2) = %Subst(AFrmArrStr:1:2);
644.00 -     IN65 = *IN(65);
645.00 -     IN66 = *IN(66);
646.00 -     Write M090S1;
647.00 -     EndIf;
648.00 - EndIf;
649.00 - //
650.00 - EndSr;
651.00 - // -----
652.00 - BegSr SrMSO;
653.00 -     ZORIG = 'M';
654.00 -     ZSVERS = MSVERS;
655.00 -     If %Found(ARMAHHF1);
656.00 -         *In65 = *On;
657.00 -     Else;
658.00 -         *In65 = *Off;
659.00 -     EndIf;
660.00 -     ZSTST = MSTST;
661.00 -     ZSPGMR = MSPGMR;
662.00 -     ZSREFR = MSREFR;
663.00 -     ZSSRCF = MSSRCF;
664.00 -     ZSRLIB = MSRLIB;
665.00 -     ZODLIB = MSRLIB;
666.00 -     ZSCPVR = MSCPVR;
667.00 -     ZOBJ = MSSRCM;
668.00 -     ZSRCT = MSTYPE;
669.00 -     ZOBJT = MSOBT;
670.00 -     LRnmMso_A = *Off;
671.00 -     LRnmMso_S = *Off;
672.00 -     LRnmMob_A = *Off;
673.00 -     LRnmMob_S = *Off;
674.00 -     LChgTyp = *Off;
675.00 -     LTxtGen = *Off;
676.00 - // SET ACTION
677.00 - *In66 = *Off;
678.00 - Select;
679.00 -     When MSTMAJ = 'A';
680.00 -         ACTION = CST(1);
681.00 -         WEnAjout = '1';
682.00 -         If Mso_JrnObj <> *Blank;
683.00 -             LRnmMso_A = *On;
684.00 -         EndIf;
685.00 -     When MSTMAJ = 'M';
686.00 - // Type changed ?
687.00 -     If Mso_CRnTyp <> *Blank
688.00 -         and Mso_CRnTyp <> MSTYPE;

```

```

689.00 - ACTION = CST(11);
690.00 - *In66 = *On;
691.00 - WTxt = 'Old : ' + %Trim(Mso_JrnObj) + ' '
692.00 - + %Trim(Mso_CrnTyp);
693.00 - LChgTyp = *On;
694.00 - LTxtGen = *On;
695.00 - Else;
696.00 - ACTION = CST(2);
697.00 - EndIf;
698.00 - When MSTMAJ = 'S';
699.00 - If Mso_JrnObj = *Blank;
700.00 - ACTION = CST(3);
701.00 - Else;
702.00 - ACTION = CST(10);
703.00 - *In66 = *On;
704.00 - If Obj = Obj1;
705.00 - WTxt = 'New : ' + %Trim(Mso_JrnObj) + ' '
706.00 - + %Trim(Mso_CrnTyp);
707.00 - ZOBJ = Mso_JrnObj;
708.00 - ZSRCT = Mso_CrnTyp;
709.00 - LRnmMso_S = *On;
710.00 - Else;
711.00 - WTxt = 'Old : ' + %Trim(MsSrcM) + ' '
712.00 - + %Trim(MsType);
713.00 - EndIf;
714.00 - LTxtGen = *On;
715.00 - EndIf;
716.00 - When MSTMAJ = 'I';
717.00 - ACTION = CST(5);
718.00 - When MSTMAJ = 'T';
719.00 - ACTION = CST(8);
720.00 - When MSTMAJ = 'R';
721.00 - ACTION = CST(9);
722.00 - Other;
723.00 - ACTION = *Blanks;
724.00 - EndSl;
725.00 - ZTMAJ = MSTMAJ;
726.00 - EndSr;
727.00 - // -----
728.00 - BegSr SrMOB;
729.00 - ZORIG = 'O';
730.00 - ZSVERS = MOVERS;
731.00 - If %Found(ARMAHFF1);
732.00 - *In65 = *On;
733.00 - Else;
734.00 - *In65 = *Off;
735.00 - EndIf;
736.00 - ZSTST = MOTST;
737.00 - ZSPGMR = MOPGMR;
738.00 - ZSREFR = MOREFR;
739.00 - ZSSRCF = *Blanks;
740.00 - ZSRLIB = *Blanks;
741.00 - ZODLIB = MOOBLB;
742.00 - ZSCPVR = 0;
743.00 - ZOBJ = MOOBJ;
744.00 - ZSRCT = MOOBTP;
745.00 - ZOBJT = MOOBTP;
746.00 - LRnmMso_A = *Off;
747.00 - LRnmMso_S = *Off;
748.00 - LRnmMob_A = *Off;
749.00 - LRnmMob_S = *Off;
750.00 - LChgTyp = *Off;
751.00 - LTxtGen = *Off;
752.00 - // SET ACTION
753.00 - *In66 = *Off;
754.00 - Select;
755.00 - When MOTMAJ = 'A';
756.00 - WEnAjout = '1';
757.00 - ACTION = CST(1);
758.00 - If Mob_JrnObj <> *Blank;
759.00 - LRnmMob_A = *On;
760.00 - EndIf;
761.00 - When MOTMAJ = 'M';
762.00 - ACTION = CST(2);
763.00 - When MOTMAJ = 'S';
764.00 - If Mob_JrnObj = *Blank;
765.00 - ACTION = CST(3);

```

```

766.00 -         Else;
767.00 -             ACTION = CST(10);
768.00 -             *In66 = *On;
769.00 -             If Obj = Obj1;
770.00 -                 WTxt = 'New : ' + %Trim(Mob_JrnObj) + ' '
771.00 -                     + %Trim(Mob_CrnTyp);
772.00 -                 ZOBJ = Mob_JrnObj;
773.00 -                 ZSRCT = Mob_CrnTyp;
774.00 -                 LnmMob_S = *On;
775.00 -             Else;
776.00 -                 WTxt = 'Old : ' + %Trim(MoObj) + ' '
777.00 -                     + %Trim(MoObjTp);
778.00 -             EndIf;
779.00 -             LTxtGen = *On;
780.00 -             EndIf;
781.00 -             When MOTMAJ = 'U';
782.00 -                 ACTION = CST(6);
783.00 -             When MOTMAJ = 'T';
784.00 -                 ACTION = CST(8);
785.00 -             When MOTMAJ = 'R';
786.00 -                 ACTION = CST(9);
787.00 -             Other;
788.00 -             // * * eval WEnAjout = '1'
789.00 -                 ACTION = CST(4);
790.00 -             EndSl;
791.00 -             ZTMAJ = MOTMAJ;
792.00 -             EndSr;
793.00 -             // -----
794.00 -             // OPTION 1 : CHECK OUT THE COMPONENT
795.00 -             // -----
796.00 -             BegSr SBR1;
797.00 -             // ----- WHEN IT'S A SOURCE
798.00 -             If *IN63 = *ON
799.00 -                 and ZTMAJ <> 'U';
800.00 -                 Select;
801.00 -                 // No CHKOBJOUT from base version (fictive)
802.00 -                 When ZSTST = '*';
803.00 -                     MSGID = 'MSG1238';
804.00 -                     Pgm_ARSNDMSG_001(PGMQ : MSGID : RIEN);
807.00 -                 // No CHKOBJOUT from a version where it is only recompiled
808.00 -                 When ZORIG = 'O' and ZTMAJ = ' ';
809.00 -                     MSGID = 'MSG2213';
809.02 -                     Pgm_ARSNDMSG_001(PGMQ : MSGID : RIEN);
813.00 -                 Other;
814.00 -                 // FIRST RETRIEVE IT IN QTEMP
815.00 -                 //             MOVEA *BLANK CMD
816.00 -                 //             MOVEL CD(8) CMDLIN
817.00 -                 //             MOVEA CMDLIN CMD(2)
818.00 -                 //             MOVEA OBJ CMD(19)
819.00 -                 //             MOVEA MBRTYP CMD(38)
820.00 -                 //             MOVEA PAPP CMD(56)
821.00 -                 //             MOVEA ZSREFR CMD(67)
822.00 -                 //             MOVEA ZSVERS CMD(79)
823.00 -                 //             MOVEA ' ' ' ' CMD(87)
824.00 -                 //             MOVEL CD(9) CMDLIN
825.00 -                 //             MOVEA CMDLIN CMD(90)
826.00 -                 //             EXSR QCMDEX
827.00 -                 //             80 EXSR POSCUR
828.00 -                 //             80 GOTO AFFIC
829.00 -                 // Use ARM122R1 to retrieve source and object version library
830.00 -                 // used by CHKOUT (idem RTVSRC)
831.00 -                 WMSVERS = ZSVERS;
831.01 -                 DoU ATag <> 'LASTVER';
832.01 -                 // branch when ATag = 'LASTVER'
832.02 -                 ATag = *Blanks;
833.01 -                 RTVLIV = '*NO ';
833.02 -                 CHKVERS = '*YES ';
833.03 -                 Pgm_ARM122R1(ZObj : ZSrcT : PAPP : ZSREFR :
833.04 -                     WMSVERS : RTVLIV : CHKVERS : WMSCPV :
833.05 -                     RTSRCL : RTSRCF : RTSRCT : RTOBJT :
833.06 -                     WVRVER : ACT : WRET);
834.00 -                 //             PARM OBJ
835.00 -                 //             PARM MBRTYP
851.00 -                 // idem RTVSRC : change version to *LAST and retry
852.00 -                 If WRET='2' And WMSVERS<>'*LAST' And Wpass<>'1';
853.00 -                 // warning window
854.00 -                 // Eval Wmsgdta = OBJ + ZSVERS + WVRVER Target & actual v

```

```

855.00 -           Wmsgdta = ZObj + ZSVERS + WVRVER;           // Target & actual ver
856.00 -
857.01 -           Wmsgf = 'ARMSGF';
857.02 -           Wmsgid = 'MSG1322';
857.03 -           Wmsgtyp = '*DIAG';
857.04 -           Pgm_ARM122C4(Wmsgf : Wmsgid : Wmsgdta :
857.05 -             Wmsgtyp : Wret122c4);
863.00 - // Not cancelled:
864.00 -           If Wret122c4 = *blank;
865.00 -
866.00 -             WMSVERS = '*LAST';
867.00 -             Wpass = '1';
868.01 -             ATag = 'LASTVER';
868.02 -             Iter;
869.00 -             EndIf;
870.00 -           EndIf;           // WRET='2' And MSVERS
870.01 -           EndDo;
871.00 -
872.00 -           If Wret122c4 = ' ';
873.00 -             If %Subst(MBRTYP:1:3) = 'ILE'           // ILEPGM/ILESrvPGM
874.00 -               and RTSRCF = *blanks;
875.00 -               RTSRCF = '*ILESRC';
876.00 -             EndIf;
877.00 -
878.00 -             Wsrc = %Trim(RTSRCL) + '/' + %Trim(RTSRCF);
879.00 -
880.00 - // IF OK : DO THE CHECK OUT
881.01 -           Pgm_CLRPGMQ(PGMQ);
883.00 -
884.00 -           CMD = *BLANK;
885.00 -           CMDLIN = CD(5);
886.00 -           pAtoArrStr = %Addr(CMD(2));
886.01 -           %Subst(AtoArrStr:1:80) = CMDLIN;
887.00 - //           MOVEA      OBJ           CMD(19)
888.00 - //           MOVEA      MBRTYP      CMD(38)
889.00 -           pAtoArrStr = %Addr(CMD(19));
889.01 -           %Subst(AtoArrStr:1:10) = ZObj;
890.00 -           pAtoArrStr = %Addr(CMD(38));
890.01 -           %Subst(AtoArrStr:1:10) = ZSrcT;
891.00 -           If In65 = *Off;
892.00 -             CMDLIN = CD(6);
893.00 -             pAtoArrStr = %Addr(CMD(50));
893.01 -             %Subst(AtoArrStr:1:80) = CMDLIN;
894.00 -             pAtoArrStr = %Addr(CMD(63));
894.01 -             %Subst(AtoArrStr:1:21) = Wsrc;
895.00 -           Else;
896.00 -             CmdLin = CD(12);
897.00 -             pAtoArrStr = %Addr(CMD(90));
897.01 -             %Subst(AtoArrStr:1:80) = CMDLIN;
898.00 -             pAtoArrStr = %Addr(CMD(129));
898.01 -             %Subst(AtoArrStr:1:3) = pApp;
899.00 -             CMD(133) = ' ';
900.00 -             pAtoArrStr = %Addr(CMD(134));
900.01 -             %Subst(AtoArrStr:1:8) = zsVers;
901.00 -             CMD(142) = ' ';
902.00 -             pAtoArrStr = %Addr(CMD(144));
902.01 -             %Subst(AtoArrStr:1:4) = '*DFT';
903.00 -           EndIf;
904.00 -
905.00 -           Exsr QCMDEX;
906.00 -
906.01 -           If *In80;
907.01 -             Pgm_àRMVPGMQ();
907.02 -             Endif;
908.00 -           EndIf;           // Wret122c4 = ' '
909.00 -           EndSl;           // VERL1 = '*'
910.00 -
911.00 -           Wpass = ' ';
912.00 -           EndIf;           // *IN63 = *ON
913.00 - //
914.00 - // ----- IF IT'S A NON-IFS OBJECT
915.00 -           If *IN63 = *OFF           // OBJ
916.00 -             and WTYPE <> '_'       // Not IFS
917.00 -             or *IN63 = *ON         // SOURCE
918.00 -             and ZTMAJ = 'U';       // in *UPDDTA
919.00 - // RETRIEVE VERSION LIBRARY
920.00 - //

```

```

921.00 - Chain (PAPP : ZSTST : ZSVERS) ARMVERL2;
921.01 - *IN80 = not %Found;
921.02 - If *In80;
922.00 -     VERLIB = *Blanks;
922.01 - Endif;
922.02 - If not *In80;
923.00 -     VERLIB = VRLIB;
923.01 - Endif;
923.02 - If not *In80;
924.00 -     VERL1 = VERLIB;
924.01 - Endif;
925.00 - //
926.00 - If VERLIB = *BLANKS;
927.01 -     MSGID = 'MSG9804';
927.02 -     Pgm_ARSNDMSG_001(PGMQ : MSGID : RIEN);
931.00 - Endif;
932.00 - //
933.00 - // BASE VERSION
934.00 - If VERL1 = '*';
935.01 -     MSGID = 'MSG1238';
935.02 -     Pgm_ARSNDMSG_001(PGMQ : MSGID : RIEN);
939.00 - Else;
940.00 - // CALL CHKOUT COMMAND
941.00 -     CMD = *BLANK;
942.00 -     CMDLIN = CD(5);
943.00 -     pToArrStr = %Addr(CMD(2));
943.01 -     %Subst(AToArrStr:1:80) = CMDLIN;
944.00 -
945.00 -     If isDTC and %len(wAlias) > *zero;
946.00 -         pToArrStr = %Addr(CMD(19));
946.01 -         %Subst(AToArrStr:1:4) = '*DTC';
947.00 -     Else;
948.00 - //             MOVEA     OBJ             CMD(19)
949.00 -         pToArrStr = %Addr(CMD(19));
949.01 -         %Subst(AToArrStr:1:10) = ZObj;
950.00 -     Endif;
951.00 -
952.00 - //             MOVEA     MBRTYP          CMD(38)
953.00 -     pToArrStr = %Addr(CMD(38));
953.01 -     %Subst(AToArrStr:1:10) = ZSrcT;
954.00 -
955.00 - // Check out mode *CHKOUT on the object without source (or *UPDDTA)
956.00 -
957.00 -     If ZTMAJ = 'U';
958.00 -         CMDds = %trim(CMDds) + ' ACTION(*UPDDTA)';
959.00 -     Else;
960.00 -         CMDds = %trim(CMDds) + ' ACTION(*CHKOUT)';
961.00 -     Endif;
962.00 - // DTC:
963.00 -     If isDTC and %len(wAlias) > *zero;
964.00 -         CMDds = %trim(CMDds) + ' FROMDTCOBJ(' +
965.00 -             wAliasApo + ')'; // * Quotes
966.00 -
967.00 -
968.00 -         CMDds = %trim(CMDds) + ' FROMDTCCLIB(' +
969.00 -             VERLIB + ')';
970.00 -     Else;
971.00 - //             Eval     CMDds = %trim(CMDds) + ' FROMOBJ(' +
972.00 - //                 %trim(VerLib) + '/' + %trim(Obj) + ')'
973.00 -         CMDds = %trim(CMDds) + ' FROMOBJ(' +
974.00 -             %trim(VerLib) + '/' + %trim(ZObj) + ')';
975.00 -
976.00 - // *             MOVEA     CD(7)             CMDLIN
977.00 - // *             MOVEA     CMDLIN          CMD(50)
978.00 - // *             Eval     W_LibObj = %trim(VerLib) + '/' + Obj
979.00 - // *             Movea    W_LibObj        CMD(58)
980.00 -     Endif;
981.00 -
982.00 -     Exsr QCMDEX;
983.00 -
983.01 -     If *In80;
984.01 -         Pgm_armVPGMQ();
984.02 -     Endif;
985.00 - Endif; // VERL1 = '*'
986.00 - Endif; // *IN63 = *OFF
987.00 -
988.00 - // ----- IF IT'S AN IFS OBJECT

```

```

989.00 -         If *IN63 = *OFF
990.00 -             and WTYPE = '_' ;                               // SRC
991.00 -         //
992.00 -         // NO BASE VERSION HERE
993.00 -         If ZSTST = '*';
994.01 -             MSGID = 'MSG1238';
994.02 -             Pgm_ARSNDMSG_001(PGMQ : MSGID : RIEN);
998.00 -         Else;
999.00 -         //
1000.00 -        // CALL CHKOUT COMMAND
1001.00 -            CMD = *BLANK;
1002.00 -            CMDLIN = CD(5);
1003.00 -            pAtoArrStr = %Addr(CMD(2));
1003.01 -            %Subst(AtoArrStr:1:80) = CMDLIN;
1004.00 -        //             MOVEA      OBJ          CMD(19)
1005.00 -        //             MOVEA      MBRTYP       CMD(38)
1006.00 -            pAtoArrStr = %Addr(CMD(19));
1006.01 -            %Subst(AtoArrStr:1:10) = ZObj;
1007.00 -            pAtoArrStr = %Addr(CMD(38));
1007.01 -            %Subst(AtoArrStr:1:10) = ZSrcT;
1008.00 -        // Retrieve all IFS name formatted for a command (object name)
1009.00 -            WIFSNomVer = '0';                               // from ref.
1010.00 -            WIFSPar0 = 'OBJMBR';
1011.00 -            WIFSParT = 'MBRTYP';
1012.00 -            WIFSPar1 = 'IFSOBJ';
1013.00 -            Exsr CmdNomIfs;
1014.00 -        // Retrieve all IFS name formatted for a command (object version name)
1015.00 -            WIFSNomVer = '1';                               // from vers.
1016.00 -            WIFSPar0 = ' ';
1017.00 -            WIFSParT = ' ';
1018.00 -            WIFSPar1 = 'FROMIFSOBJ';
1019.00 -            Exsr CmdNomIfs;
1020.00 -            Exsr QCMDEX;
1020.01 -            If *In80;
1021.01 -                Pgm_àRMVPGMQ();
1021.02 -            Endif;
1022.00 -        EndIf;
1023.00 -        EndIf;                               // IFS Object
1024.00 -    EndSr;
1025.00 -    // -----
1026.00 -    // OPTION 6 : DISPLAY THE VERSION
1027.00 -    // -----
1028.00 -    BegSr SBR6;
1029.00 -    // BASE VERSION
1030.00 -    If ZSTST = '*';
1031.01 -        MSGID = 'MSG1238';
1031.02 -        Pgm_ARSNDMSG_001(PGMQ : MSGID : RIEN);
1035.00 -    Else;
1036.00 -        CMD = *BLANK;
1037.00 -        CMDLIN = CD(1);
1038.00 -        pAtoArrStr = %Addr(CMD(1));
1038.01 -        %Subst(AtoArrStr:1:80) = CMDLIN;
1039.00 -        pAtoArrStr = %Addr(CMD(14));
1039.01 -        %Subst(AtoArrStr:1:3) = PAPP;
1040.00 -        CMD(25) = ZSTST;
1041.00 -        pAtoArrStr = %Addr(CMD(37));
1041.01 -        %Subst(AtoArrStr:1:8) = ZSVERS;
1042.00 -        Exsr QCMDEX;
1042.01 -        If *In80;
1043.01 -            Pgm_àRMVPGMQ();
1043.02 -        Endif;
1044.00 -    EndIf;
1045.00 -    EndSr;
1046.00 -    // -----
1047.00 -    // OPTION 5 : DISPLAY THE SOURCE
1048.00 -    // -----
1049.00 -    BegSr SBR5;
1050.00 -
1051.00 -    Select;
1052.00 -        When isDTC;
1053.00 -            If ZODLIB <> *Blanks;                               // *BASE version ?
1054.00 -            //             CallP          ART562G1(PAPP :OBJ :MBRTYP :' ' (OBJATR)
1055.00 -            //             :ZODLIB :'DSP' :'0' :CmdDS )
1056.00 -            ART562G1(PAPP :ZObj :ZSrcT :' '                       // (OBJATR)
1057.00 -            :ZODLIB :'DSP' :'0' :CmdDS );
1058.00 -
1059.00 -            Exsr QCMDEX;

```



```

1059.01 -         If *In80;
1060.01 -             Pgm_àRMVPGMQ();
1061.01 -         Endif;
1062.01 -     EndIf;
1063.01 - // **             IF             *IN63 = '1'             Source
1064.01 - // **             or %Subst(MBRTYP:1:1) = '_'             or IFS
1065.01 -
1066.01 - //             When             %Subst(MBRTYP:1:1) = '_'             IFS
1067.01 -     When %Subst(ZSrcT:1:1) = '_' ;             // IFS
1068.01 -     Exsr Sbr5Ifs;
1069.01 - // **             Else
1070.01 -
1071.01 -     When *IN63;             // Source
1072.01 -         CMD = *BLANK;
1073.01 -         CMDLIN = CD(4);
1074.01 -         pAtoArrStr = %Addr(CMD(2));
1074.01 -         %Subst(AToArrStr:1:80) = CMDLIN;
1075.01 - //             MOVEA             OBJ             CMD(19)
1076.01 - //             MOVEA             MBRTYP             CMD(38)
1077.01 -         pAtoArrStr = %Addr(CMD(19));
1077.01 -         %Subst(AToArrStr:1:10) = ZObj;
1078.01 -         pAtoArrStr = %Addr(CMD(38));
1078.01 -         %Subst(AToArrStr:1:10) = ZSrcT;
1079.01 -         pAtoArrStr = %Addr(CMD(56));
1079.01 -         %Subst(AToArrStr:1:3) = PAPP;
1080.01 -         CMD(67) = ZSREFR;
1081.01 -         pAtoArrStr = %Addr(CMD(79));
1081.01 -         %Subst(AToArrStr:1:8) = ZSVERS;
1082.01 -         pAtoArrStr = %Addr(CMD(87));
1082.01 -         %Subst(AToArrStr:1:2) = ''';
1083.01 -         pAtoArrStr = %Addr(CMD(90));
1083.01 -         %Subst(AToArrStr:1:8) = 'CHKVERS(';
1084.01 -         pAtoArrStr = %Addr(CMD(98));
1084.01 -         %Subst(AToArrStr:1:5) = '*YES';
1085.01 -
1086.01 -     Exsr QCMDEX;
1086.01 -     If *In80;
1087.01 -         Pgm_àRMVPGMQ();
1087.02 -     Endif;
1088.01 - EndS1;
1089.01 -
1090.01 - EndSr;
1091.01 - // -----
1092.01 - // OPTION 7 : DISPLAY DOCUMENTS FOR THE OBJECT IN THE VERSION
1093.01 - // -----
1094.01 - BegSr SBR7;
1095.01 - // BASE VERSION
1096.01 -     If ZSTST = '*';
1097.01 -         MSGID = 'MSG1238';
1097.02 -         Pgm_ARSNDMSG_001(PGMQ : MSGID : RIEN);
1101.01 -     Else;
1102.01 -         CMD = *BLANK;
1103.01 -         CMDLIN = CD(2);
1104.01 -         pAtoArrStr = %Addr(CMD(1));
1104.01 -         %Subst(AToArrStr:1:80) = CMDLIN;
1105.01 -         pAtoArrStr = %Addr(CMD(18));
1105.01 -         %Subst(AToArrStr:1:10) = ZObj;
1106.01 -         pAtoArrStr = %Addr(CMD(37));
1106.01 -         %Subst(AToArrStr:1:10) = ZSrcT;
1107.01 -         pAtoArrStr = %Addr(CMD(55));
1107.01 -         %Subst(AToArrStr:1:3) = PAPP;
1108.01 -         CMD(66) = ZSTST;
1109.01 -         CMDLIN = CD(3);
1110.01 -         pAtoArrStr = %Addr(CMD(69));
1110.01 -         %Subst(AToArrStr:1:80) = CMDLIN;
1111.01 -         pAtoArrStr = %Addr(CMD(78));
1111.01 -         %Subst(AToArrStr:1:8) = ZSVERS;
1112.01 - // Retrieve all IFS name formatted for a command (object name)
1113.01 - //             IF             %Subst(MBRTYP:1:1) = '_'
1114.01 -         If %Subst(ZSrcT:1:1) = '_' ;
1115.01 -             WIFSNomVer = '0';             // Name Ref.
1116.01 -             WIFSPar0 = 'OBJMBR';
1117.01 -             WIFSParT = 'MBRTYP';
1118.01 -             WIFSPar1 = 'IFSOBJ';
1119.01 -             Exsr CmdNomIfs;
1120.01 -         Endif;

```

```

1121.00 -         Exsr QCMDEX;
1121.01 -         If *In80;
1122.01 -             Pgm_àRMVPGMQ();
1122.02 -         Endif;
1123.00 -     EndIf;
1124.00 - EndSr;
1125.00 - // -----
1126.00 - // OPTION 14 : COMPARISON
1127.00 - // -----
1128.00 - BegSr SBR14;
1129.00 - // Test Option not authorized for IFS component
1130.00 -     Exsr SROptNifs;
1131.00 -     If *In80 = '1';
1132.01 -         LeaveSr;
1133.00 -     Endif;
1134.00 - //
1135.00 - // COMPARISON OPTION
1136.00 -     If *IN63 = *ON;
1137.00 -         REF = '*REFFILE';
1138.00 -         CMD = *BLANK;
1139.00 -         CMDLIN = CD(10);
1140.00 -         pAtoArrStr = %Addr(CMD(1));
1140.01 -         %Subst(AToArrStr:1:80) = CMDLIN;
1141.00 -         pAtoArrStr = %Addr(CMD(17));
1141.01 -         %Subst(AToArrStr:1:10) = ZSSRCF;
1142.00 -         pAtoArrStr = %Addr(CMD(37));
1142.01 -         %Subst(AToArrStr:1:10) = ZObj;
1143.00 -         RESULT = %Trimr(ZSRLIB) + '/';
1144.00 -         RESULT = %Trimr(RESULT) + REF;
1145.00 -         pAtoArrStr = %Addr(CMD(58));
1145.01 -         %Subst(AToArrStr:1:21) = RESULT;
1146.00 -         CMD(79) = ')';
1147.00 -         Exsr QCMDEX;
1147.01 -         If *In80;
1148.01 -             Pgm_àRMVPGMQ();
1148.02 -         Endif;
1149.00 -     Endif;
1150.00 - EndSr;
1151.00 - // -----
1152.00 - // OPTION 20 : VERSIONS
1153.00 - // -----
1154.00 - BegSr SBR20;
1155.00 - //
1156.00 -     CMD = *Blanks;
1157.00 -     CMDLIN = CD(13);
1158.00 -     pAtoArrStr = %Addr(CMD(1));
1158.01 -     %Subst(AToArrStr:1:80) = CMDLIN;
1159.00 -     pAtoArrStr = %Addr(CMD(18));
1159.01 -     %Subst(AToArrStr:1:10) = ZOBJ;
1160.00 -     pAtoArrStr = %Addr(CMD(37));
1160.01 -     %Subst(AToArrStr:1:10) = ZSRCT;
1161.00 -     pAtoArrStr = %Addr(CMD(55));
1161.01 -     %Subst(AToArrStr:1:3) = PAPP;
1162.00 -     Exsr QCMDEX;
1162.01 -     If *In80;
1163.01 -         Pgm_àRMVPGMQ();
1163.02 -     Endif;
1164.00 - //
1165.00 - EndSr;
1166.00 - // =====
1167.00 - // TEST OPTION NOT AUTHORIZED FOR IFS COMPONENT
1168.00 - // =====
1169.00 - BegSr SROptNifs;
1170.00 -     If %Subst(MBRTYP:1:1) = '_';
1171.00 -         *IN80 = '1';
1172.01 -         MSGID = 'MSG1956';
1172.02 -         Pgm_ARSNDMSG(PGMQ : MSGID : OPT);
1176.00 -     Endif;
1177.00 - EndSr;
1178.00 - // =====
1179.00 - // POSITION FOR CURSOR AND PAGE
1180.00 - // =====
1181.00 - BegSr POSCUR;
1182.00 -     WRANG = LIG1;
1183.00 - // CALCULATE CURSOR POSITION
1184.00 -     WSOM = LIG1 - 1;
1185.00 -     WDIV = WSOM / 11;

```

```

1186.00 -         REST = %Rem(Wsom : 11);
1187.00 -         DEBPAG = WDIV * 11;
1188.00 -         DEBPAG += 1;
1189.00 -         CSRLIN = REST + 11;
1190.00 -         CSRPOS = 2;
1191.00 -     EndSr;
1192.00 -     // =====
1193.00 -     // SUB-ROUTINE TO LOAD DISPLAYED CONSTANT FIELDS
1194.00 -     // =====
1195.00 -     BegSr RTVCST;
1196.00 -     // USR0030 to USR0035 (1..6)
1197.00 -     CST(1) = Api_RtvCstMsg('USR0030');
1198.00 -     CST(2) = Api_RtvCstMsg('USR0031');
1199.00 -     CST(3) = Api_RtvCstMsg('USR0032');
1200.00 -     CST(4) = Api_RtvCstMsg('USR0033');
1201.00 -     CST(5) = Api_RtvCstMsg('USR0034');
1202.00 -     CST(6) = Api_RtvCstMsg('USR0035');
1203.00 -     // TEXT "DELETED VERSION"
1204.00 -     CST(7) = Api_RtvCstMsg('USR0038');
1205.00 -     // USR0081 to USR0082 (8..9)
1206.00 -     CST(8) = Api_RtvCstMsg('USR0081');
1207.00 -     CST(9) = Api_RtvCstMsg('USR0082');
1208.00 -     //
1209.00 -     CST(10) = Api_RtvCstMsg('USR0039');
1210.00 -     CST(11) = Api_RtvCstMsg('USR0047');
1211.00 -     //
1212.00 -     EndSr;
1213.00 -     // =====
1214.00 -     // EXECUTE A COMMAND
1215.00 -     // =====
1216.00 -     BegSr QCMDEX;
1217.00 -     If *IN44 = *ON;
1218.00 -         CMDDS = '?' + CMDDS;
1219.00 -     EndIf;
1220.01 -     CallP(e) Pgm_QCMDEXC(CMD : LEN);
1220.02 -     *In80 = %Error;
1222.01 -     If *In80;
1223.01 -         Pgm_àRMVPGMQ();
1223.02 -     Endif;
1224.00 -     EndSr;
1225.00 -     // =====
1226.00 -     // COMMAND TO EXECUTE TO DISPLAY AN IFS OBJECT
1227.00 -     // =====
1228.00 -     BegSr Sbr5IFS;
1229.00 -     //
1230.00 -     CMD = *BLANK;
1231.00 -     CMDLIN = CD(11);
1232.00 -     pAtoArrStr = %Addr(CMD(1));
1232.01 -     %Subst(AtoArrStr:1:80) = CMDLIN;
1233.00 -     pAtoArrStr = %Addr(CMD(19));
1233.01 -     %Subst(AtoArrStr:1:10) = MBRTYP;
1234.00 -     pAtoArrStr = %Addr(CMD(38));
1234.01 -     %Subst(AtoArrStr:1:4) = '*DSP';
1235.00 -     // Retrieve all IFS name formatted for a command (object name)
1236.00 -     WIFSNomVer = '1';
1237.00 -     WIFSPar0 = ' ';
1238.00 -     WIFSParT = 'OBJTYPE';
1239.00 -     WIFSPar1 = 'OBJ';
1240.00 -     Exsr CmdNomIfs;
1241.00 -
1242.00 -     Exsr QCMDEX;
1242.01 -     If *In80;
1243.01 -         Pgm_àRMVPGMQ();
1243.02 -     Endif;
1244.00 -
1245.00 -     EndSr;
1246.00 -     //
1247.00 -     // =====
1248.00 -     // READ IFS COMPONENT NAME --> To WIFSNAM and len in WIFSNAMLG
1249.00 -     // If WIFSNomVer = '0' --> Reference name
1250.00 -     // If WIFSNomVer = '1' --> IFS name in the version
1251.00 -     // WIFSNomCas = 'L' or 'U' --> Uppercase or Lowercase.
1252.00 -     // =====
1253.00 -     BegSr LirNomIfs;
1254.00 -     // Retrieve complete IFS name
1255.00 -     WIFSNAMLG =
1256.00 -         IFS_IDLEC('JOB':WIFSNomCas:PApp:'0': '*NONE':

```

```

1257.00 -         OBJ:%Addr(WIFSNAM):%Size(WIFSNAM) );
1258.00 - // add path name for version
1259.00 -     If WIFSNomVer = '1' and ZSTST <> '*'; // not based vers
1260.00 - // add directory for version
1261.00 -     WIFSNAMLG = IFS_CHGNAM(
1262.00 -         %Addr(WIFSNAM):%Size(WIFSNAM):WIFSNAMLG:
1263.00 -         PApp:ZSTST:ZSVERS);
1264.00 -     EndIf;
1265.00 - EndSr;
1266.00 - // =====
1267.00 - // FORMAT IFS COMPONENT NAME for a command
1268.00 - //     If WIFSNomVer = '0' --> Reference name
1269.00 - //     If WIFSNomVer = '1' --> Version IFS Name
1270.00 - //     WIFSPar0      : Parameter name for Object to replace by *IFS
1271.00 - //     WIFSParT      : Parameter name containing the type
1272.00 - //     WIFSPar1      : Parameter name receiving complete IFS name
1273.00 - // =====
1274.00 - BegSr CmdNomIfs;
1275.00 - // Retrieve complete IFS name
1276.00 -     WIFSNomCas = 'L';
1277.00 -     Exsr LirNomIFS;
1278.00 - // Format IFS name for a command
1279.00 -     IFS_CMDNAM(WIFSPar0:WIFSParT:WIFSPar1:
1280.00 -         %Addr(CMDDS):%Size(CMDDS):
1281.00 -         'JOB':%Addr(WIFSNAM):WIFSNAMLG);
1282.00 - EndSr;
1283.00 - // =====
1284.00 - **
1285.00 - DSPVER APPID( ) TSTID( ) VERSION(' ')
1286.00 - DSPOBJDOC OBJMBR( ) MBRTYP( ) APPID( ) TSTID( )
1287.00 - VERSION(' ')
1288.00 - DSPSRC OBJMBR( ) MBRTYP( ) APPID( ) ENVID( ) VERSION(' ')
1289.00 - CHKOBJOUT OBJMBR( ) MBRTYP( )
1290.00 - FROMSRCFIL( )
1291.00 - FROMOBJ( )
1292.00 - RTVSRC OBJMBR( ) MBRTYP( ) APPID( ) ENVID( ) VERSION(' ')
1293.00 - TOSRCFIL(QTEMP/QRTVSRC) UPDTFRLOG(*NO)
1294.00 - CMPMBR SRCFILE1( ) SRCMBR1( ) SRCFILE2( )
1295.00 - USEIFSOBJ OBJTYPE( ) OPTION( )
1296.00 - FROMSRCFIL(*ARCHIVE) FROMARCH( )
1297.00 - DSPOBJVER OBJMBR( ) MBRTYP( ) APPID( )

```