ADAS Subroutine b8wrps

```
      subroutine b8wrps( open15 , iunt15 , open13 , iunt13 ,
     &                   dsninc , dsfull , dsnexp , ibsela ,
     &                   titled , date   , user   ,
     &                   ndlev  , ndtem  , ndden  , ndmet  , ndtrn ,
     &                   lnorm  ,
     &                   iz     , iz0    , iz1    ,
     &                   il     , nmet   , nord   ,
     &                   maxt   , maxd   , icntr  , icnti  , icnth ,
     &                   isa    , ila    , xja    ,
     &                   cstrga , wa     ,
     &                   icnte  ,
     &                   ie1a   , ie2a   , aa     ,
     &                   imetr  , iordr  , teva   , densa  ,
     &                   npl    , nplr   , npli   , npl3   ,
     &                   lrsel  , lisel  , lhsel  , liosel ,
     &                   lpsel  , lzsel  , lnsel  ,
     &                   wvls   , wvll   , avlt   ,
     &                   stvr   , stvi   , stvh   ,
     &                   ratpia , ratmia , stack  ,
     &                   fvionr , sgrda  ,
     &                   lsseta , lss04a
     &                 )

C-----------------------------------------------------------------------
C
C  ******************** FORTRAN77 SUBROUTINE: B8WRPS ********************
C
C  PURPOSE:  To output data to pec and sxb passing files.
C
C  CALLING PROGRAM: ADAS208
C
C
C  SUBROUTINE:
C
C  INPUT : (I*4)  IUNT15    = UNIT NUMBER FOR PECS.
C  INPUT : (L*4)  OPEN15    =.TRUE. IF SEC OUTPUT IS REQUIRED
C  INPUT : (I*4)  IUNT13    = UNIT NUMBER FOR SXBS.
C  INPUT : (L*4)  OPEN13    =.TRUE. IF SXB OUTPUT IS REQUIRED
C  INPUT : (C*44) DSNINC    = INPUT ADF04 DATA SET NAME
C  INPUT : (C*80) DSFULL    = INPUT ADF07 DATA SET NAME
C  INPUT : (C*80) DSNEXP    = INPUT EXPANSION FILE
C  INPUT : (I*4)  IBSELA(,) = IONISATION DATA BLOCK SELECTION INDICES
C                             1ST DIMENSION - (Z) ION METASTABLE COUNTER
C                             2ND DIMENSION - (Z+1) ION METASTABLE COUNTER
C  INPUT : (C*3)  TITLED    = ELEMENT SYMBOL.
C  INPUT : (C*8)  DATE      = CURRENT DATE.
C  INPUT : (C*30) USER      = FULL NAME OF AUTHOR.
C
C  INPUT : (I*4)  NDLEV     = MAXIMUM NUMBER OF LEVELS ALLOWED
C  INPUT : (I*4)  NDTEM     = MAXIMUM NUMBER OF TEMPERATURES ALLOWED
C  INPUT : (I*4)  NDDEN     = MAXIMUM NUMBER OF DENSITIES ALLOWED
C  INPUT : (I*4)  NDMET     = MAXIMUM NUMBER OF METASTABLES ALLOWED
```

```
C  INPUT : (I*4)  NDTRN     = PARAMETER = MAX. NO. OF TRANSITIONS ALLOWED
C  INPUT : (L*4)  LNORM     =.TRUE.  => IF NMET=1 THEN VARIOUS
C                                       EMISSIVITY OUTPUT FILES
C                                       NORMALISED TO STAGE TOT.POPULATN.
C                                       (** NORM TYPE = T)
C                           =.FALSE. => OTHERWISE NORMALISE TO IDENTIFIED
C                                       METASTABLE POPULATIONS.
C                                        (** NORM TYPE = M)
C  INPUT : (I*4)  IZ        =  RECOMBINED ION CHARGE READ
C  INPUT : (I*4)  IZ0       =         NUCLEAR CHARGE READ
C  INPUT : (I*4)  IZ1       = RECOMBINING ION CHARGE READ
C                             (NOTE: IZ1 SHOULD EQUAL IZ+1)
C
C  INPUT : (I*4)  IL        = NUMBER OF ENERGY LEVELS
C  INPUT : (I*4)  NMET      = NUMBER OF METASTABLES LEVELS: 1<=NMET<=NDMET
C  INPUT : (I*4)  NORD      = NUMBER OF ORDINARY LEVELS ('IL' - 'NMET')
C  INPUT : (I*4)  MAXT      = NUMBER OF INPUT TEMPERATURES ( 1 -> 'NDTEM')
C  INPUT : (I*4)  MAXD      = NUMBER OF INPUT DENSITIES ( 1 -> 'NDDEN')
C  INPUT : (I*4)  ICNTR     = NUMBER OF FREE ELECTRON RECOMBINATIONS INPUT
C  INPUT : (I*4)  ICNTI     = NUMBER OF LOWER STAGE IONISATIONS      INPUT
C  INPUT : (I*4)  ICNTH     = NO. OF CHARGE EXCHANGE RECOMBINATIONS INPUT
C  INPUT : (I*4)  ISA()     = MULTIPLICITY FOR LEVEL 'IA()'
C                             NOTE: (ISA-1)/2 = QUANTUM NUMBER (S)
C  INPUT : (I*4)  ILA()     = QUANTUM NUMBER (L) FOR LEVEL 'IA()'
C  INPUT : (R*8)  XJA()     = QUANTUM NUMBER (J-VALUE) FOR LEVEL 'IA()'
C                             NOTE: (2*XJA)+1 = STATISTICAL WEIGHT
C  INPUT : (C*18) CSTRGA()  = NOMENCLATURE/CONFIGURATION FOR LEVEL 'IA()'
C  INPUT : (R*8)  WA()      = ENERGY RELATIVE TO LEVEL 1 (CM-1)
C                             DIMENSION: LEVEL INDEX
C  INPUT : (I*4)  ICNTE     = NUMBER OF ELECTRON IMPACT TRANSITIONS INPUT
C  INPUT : (I*4)  IE1A()    = ELECTRON IMPACT TRANSITION:
C                              LOWER ENERGY LEVEL INDEX
C  INPUT : (I*4)  IE2A()    = ELECTRON IMPACT TRANSITION:
C                              UPPER ENERGY LEVEL INDEX
C  INPUT : (R*8)  AA()      = ELECTRON IMPACT TRANSITION: A-VALUE (SEC-1)
C  INPUT : (I*4)  IMETR()   = INDEX OF METASTABLE IN COMPLETE LEVEL LIST
C  INPUT : (I*4)  IORDR()   = INDEX OF ORDINARY LEVELS IN COMPLETE LEVEL
C                             LIST.
C  INPUT : (R*8)  TEVA()    = ELECTRON TEMPERATURES (UNITS: KELVIN)
C  INPUT : (R*8)  DENSA()   = ELECTRON DENSITIES  (UNITS: CM-3)
C  INPUT : (I*4)  NPL       = NO. OF METASTABLES OF(Z+1) ION ACCESSED
C                                BY EXCITED STATE IONISATION IN COPASE
C                                FILE WITH IONISATION POTENTIALS GIVEN
C                                ON THE FIRST DATA LINE
C  INPUT : (I*4)  NPLR      = NO. OF ACTIVE METASTABLES OF (Z+1) ION
C  INPUT : (I*4)  NPLI      = NO. OF ACTIVE METASTABLES OF (Z+1) ION
C  INPUT : (I*4)  NPL3      = NO. OF ACTIVE METASTABLES OF (Z+1) ION
C  INPUT   (L*4)  LRSEL     = .TRUE.  => INCLUDE FREE ELECTRON
C                                        RECOMBINATION.
C                           = .FALSE. => DO NOT INCLUDE FREE ELECTRON
C                                        RECOMBINATION.
C  INPUT : (L*4)  LISEL     = .TRUE.  => INCLUDE ELECTRON IMPACT
C                                        IONISATION.
```

```
C                              = .FALSE. => DO NOT INCLUDE FREE ELECTRON
C                                           RECOMBINATION.
C  INPUT : (L*4)  LHSEL        = .TRUE.  => INCLUDE CHARGE TRANSFER FROM
C                                           NEUTRAL HYDROGREN.
C                              = .FALSE. => DO NOT INCLUDE CHARGE TRANSFER
C                                           FROM NEUTRAL HYDROGREN.
C  INPUT : (L*4)  LIOSEL       = .TRUE.  => INCLUDE IONISATION RATES
C                              = .FALSE. => DO NOT INCLUDE IONISATION RATES
C                                 FOR RECOM AND  3-BODY
C  INPUT : (L*4)  LPSEL        = .TRUE.  => INCLUDE PROTON COLLISIONS
C                              = .FALSE. =>DO NOT INCLUDE PROTON COLLISIONS
C  INPUT : (L*4)  LZSEL        = .TRUE.  => SCALE PROTON COLLISIONS WITH
C                                           PLASMA Z EFFECTIVE'ZEFF'.
C                              = .FALSE. => DO NOT SCALE PROTON COLLISIONS
C                                            WITH PLASMA Z EFFECTIVE 'ZEFF'.
C                                 (ONLY USED IF 'LPSEL=.TRUE.')
C  INPUT : (L*4)  LNSEL        = .TRUE.  => INCLUDE PROJECTED BUNDLE-N DATA
C                                            FROM DATAFILE IF AVAILABLE
C                              = .FALSE. => DO NOT INCLUDE PROJECTED
C                                            BUNDLE-N DATA
C  INPUT : (R*8)  WVLS         = SHORT WAVELENGTH LIMIT FOR PEC & SXB (A)
C  INPUT : (R*8)  WVLL         = LONG WAVELENGTH LIMIT FOR PEC & SXB  (A)
C  INPUT : (R*8)  AVLT         = LOWER LIMIT OF A-VALUES FOR PEC & SXB
C
C  INPUT : (R*4) STVR(,,,)     = FREE ELECTRON RECOMBINATION COEFFICIENTS
C                                  1st DIMENSION: ORDINARY LEVEL INDEX
C                                  2nd DIMENSION: TEMPERATURE INDEX
C                                  3rd DIMENSION: DENSITY INDEX
C                                  4TH DIMENSION: PARENT INDEX
C  INPUT : (R*4) STVI(,,,)     = ELECTRON IMPACT IONISATION  COEFFICIENTS
C                                  1st DIMENSION: ORDINARY LEVEL INDEX
C                                  2nd DIMENSION: TEMPERATURE INDEX
C                                  3rd DIMENSION: DENSITY INDEX
C                                  4TH DIMENSION: PARENT INDEX
C  INPUT : (R*4) STVH(,,,)     =  CHARGE EXCHANGE COEFFICIENTS
C                                  1st DIMENSION: ORDINARY LEVEL INDEX
C                                  2nd DIMENSION: TEMPERATURE INDEX
C                                  3rd DIMENSION: DENSITY INDEX
C  INPUT : (R*8) RATPIA(,)     = RATIO ( N(Z+1)/N(Z)  STAGE ABUNDANCIES )
C                                  1ST DIMENSION: TEMP/DENS INDEX
C                                  2ND DIMENSION: PARENT INDEX
C  INPUT : (R*8) RATMIA(,)     = RATIO ( N(Z-1)/N(Z)  STAGE ABUNDANCIES )
C                                  1ST DIMENSION: TEMP/DENS INDEX
C                                  2ND DIMENSION: PARENT INDEX
C  INPUT : (R*4) STACK(,,,)    = POPULATION DEPENDENCE
C                                  1st DIMENSION: ORDINARY LEVEL INDEX
C                                  2nd DIMENSION: METASTABLE INDEX
C                                  3rd DIMENSION: TEMPERATURE INDEX
C                                  4th DIMENSION: DENSITY INDEX
C  INPUT : (R*8) FVIONR(,,,)= GEN. COLL. RAD. IONIS. RATE COEFFTS.
C                               1ST DIMENSION: (Z) ION METASTABLE INDEX
C                               2ND DIMENSION: (Z+1) ION METASTABLE INDEX
C                               3rd DIMENSION: TEMPERATURE INDEX
```

```
C                               4th DIMENSION: DENSITY INDEX
C  INPUT : (R*8)  SGRDA(,,) = GROUND & METASTABLE IONISATION RATE
C                               COEFFICIENTS  FROM SZD FILES (CM3 SEC-1)
C                               1ST DIMENSION: TEMPERATURE INDEX
C                               2ND DIMENSION: (Z) ION METASTABLE INDEX
C                               3RD DIMENSION: (Z+1) ION METASTABLE INDEX
C  INPUT : (L*4) LSSETA(,)  = .TRUE.  - MET. IONIS RATE SET IN B8GETS
C                               .FALSE.- MET. IONIS RATE NOT SET IN B8GETS
C                                1ST DIMENSION: (Z) ION METASTABLE INDEX
C                                2ND DIMENSION: (Z+1) ION METASTABLE INDEX
C  INPUT : (L*4) LSS04A(,)  = .TRUE. => IONIS. RATE SET IN ADF04 FILE:
C                               .FALSE.=> NOT SET IN ADF04 FILE
C                                 1ST DIM: LEVEL INDEX
C                                 2ND DIM: PARENT METASTABLE INDEX
C
C          (I*4) NOTRN      = PARAMETER = MAXIMUM NUMBER OF TRANSITIONS
C          (I*4) NDPEC      = PARAMETER = MAXIMUM NUMBER OF PECS PER
C                                         METASTABLE FOR OUTPUT
C          (I*4) METCNT     = COUNTER OF PECS FOR EACH METASTABLE
C
C          (I*4) I4UNIT     = FUNCTION (SEE ROUTINE SELECTION BELOW)
C
C          (I*4) I          = GENERAL USE
C          (I*4) IP         = GENERAL USE
C          (I*4) J          = GENERAL USE
C          (I*4) K          = GENERAL USE
C          (I*4) L          = GENERAL USE
C
C          (R*8) DUM1       = GENERAL USE- DUMMY
C          (R*8) DUM2       = GENERAL USE- DUMMY
C          (R*8) DUM3       = GENERAL USE- DUMMY
C
C ROUTINES:
C          ----------------------------------------------------------------
C          i4unit     ADAS      Fetch unit number for output of messages
C          b8norm     ADAS      Perform  stage population normalisation
C          b8corp     ADAS      'fixes' low te problem in rec. data of pecs
C          b8winf     ADAS      Determines ionis. source and writes
C                               comment block
C          xxeiam     ADAS      Get 2-character name of element.
C          xxslen     ADAS      Returns lenght of string.
C          xxordr     ADAS      Sorts a real*8 array.
C
C-----------------------------------------------------------------------
C
C NOTES: Based on b8wr11, b8wr12 and hapecf.
C
C
C VERSION  : 1.1
C DATE     : 15-02-2006
C MODIFIED : Martin O'Mullane
C              - First version.
C
```

```
C VERSION  : 1.2
C DATE     : 08-03-2006
C MODIFIED : Martin O'Mullane
C              - Make sure that CX pec data which is 0.0 is written
C                as 1.0E-70 to avoid splining problems when reading
C                adf15 datasets.
C
C-----------------------------------------------------------------------
      CHARACTER*18        CSTRGA(NDLEV)
      CHARACTER*8         DATE
      CHARACTER*80        DSFULL,      DSNEXP,      DSNINC
      CHARACTER*3         TITLED
      CHARACTER*30        USER
      INTEGER             IBSELA(NDMET,NDMET),      ICNTE,       ICNTH
      INTEGER             ICNTI,       ICNTR,       IE1A(NDTRN)
      INTEGER             IE2A(NDTRN), IL,          ILA(NDLEV)
      INTEGER             IMETR(NDMET),             IORDR(NDLEV)
      INTEGER             ISA(NDLEV),  IUNT13,      IUNT15,      IZ
      INTEGER             IZ0,         IZ1,         MAXD,        MAXT
      INTEGER             NDDEN,       NDLEV,       NDMET,       NDTEM
      INTEGER             NDTRN,       NMET,        NORD,        NPL
      INTEGER             NPL3,        NPLI,        NPLR
      LOGICAL             LHSEL,       LIOSEL,      LISEL,       LNORM
      LOGICAL             LNSEL,       LPSEL,       LRSEL
      LOGICAL             LSS04A(NDLEV,NDMET),      LSSETA(NDMET,NDMET)
      LOGICAL             LZSEL,       OPEN13,      OPEN15
      REAL*8              AA(NDTRN),   AVLT,        DENSA(NDDEN)
      REAL*8              FVIONR(NDMET,NDMET,NDTEM,NDDEN)
      REAL*8              RATMIA(NDDEN,NDMET),      RATPIA(NDDEN,NDMET)
      REAL*8              SGRDA(NDTEM,NDMET,NDMET)
      REAL                STACK(NDLEV,NDMET,NDTEM,NDDEN)
      REAL                STVH(NDLEV,NDTEM,NDDEN,NDMET)
      REAL                STVI(NDLEV,NDTEM,NDDEN,NDMET)
      REAL                STVR(NDLEV,NDTEM,NDDEN,NDMET)
      REAL*8              TEVA(NDTEM), WA(NDLEV),   WVLL,        WVLS
      REAL*8              XJA(NDLEV)
```