

Version 90

- [Version 1 avec nouveau majbase](#)
- [Version 2](#)
- [Version 3](#)
- [Version 4](#)
- [Version 5](#)
- [Version 6](#)
- [Version 7](#)

Version 1 avec nouveau majbase

TRANSPVMRR.TXEMP

Permet de prendre en compte la modification du taux VMRR

Avant

```
CONSTANTE( CONTRAT. VTAUX_TAXETRANSPORTVMRR)
```

Après

```
LISTEGEN( TX_TRANSPVMRR; CONSTANTE( CONTRAT. TAUX_TAXETRANSPORTVMRR ) )
```

Version 2

Création de la rubrique ACCIDENT_MISP

- Base de cotisation : BC_AT_MISP
- Taux : 100

BC_AT_MISP

Avant

```
( CONSTANTE( GENERAL. PLFSECUHEURE) *0. 07) * [ NB_HEURESTPAYE. MONTANT]
```

Après

```
( CONSTANTE( GENERAL. PLFSECUHEURE) *0. 07) * [ NB_HEURESTPAYE. MONTANT] *  
LISTEGEN( TX_ACC; CONSTANTE( CONTRAT. TAUX_ACCIDENTTRAVAIL) ) / 100
```

Création de la rubrique indemnité de volontariat associatif

SC_IND_VOLON : Indemnité volontariat associatif

Le montant a pour participation uniquement le brut en + et le brut fiscal en - (l'indemnité n'est pas soumise à l'impôt sur le revenu).

Pour les cotisations :

- VIEILLESSE_SC : Vieillesse service civique
- VIEIL_SC_TRA : Vieillesse SC sur tranche A
- N_CSGND_SC : C.S.G non déductible service civique. Base BC_CRDS Taux 9.20% CTP : 594

Version 3

Modification de la formule V_NEWAGCP.HREMU

Permet de régler le problème de saisir H_NONTRAV dans un bulletin de régul et qui n'était pas pris en compte dans le calcul agcp

[Ticket Jira](#)

Avant

```
si (([NB_HEUREREMU. BASE]=0) ou (CONSTANTE( CONTRAT. TYPESALARIE)=5))
alors (
  {Assistante Maternelle}
  si ( CONSTANTE( CONTRAT. NATURECONTRAT)=0003)
  alors (( CONSTANTE( CONTRAT. HORAIRECOLLECTIF)
    )
    *
    (
      si ([ IJBRUT. MONTANT] <>0)
      alors
([ BC_BRUT_URSAFF. MONTANT] /([ BC_BRUT_URSAFF. MONTANT] +[ IJBRUT. MONTANT] ) )
      sinon (1)
    )
  )
  sinon (
{Forfait Annuel en jours}
  si ( CONSTANTE( CONTRAT. SITUATIONHORAIRE)=4)
  alors (
    151.67*
    CONSTANTE( CONTRAT. JOUR_FORFAIT) /218
    *
    si ((([ IJBRUT. MONTANT] <>0) ou ([ ABSTOTALV. MONTANT] -
[ ABSTOTALM. MONTANT] <>0))
    alors ([ BC_BRUT_URSAFF. MONTANT]
      /
      ([ BC_BRUT_URSAFF. MONTANT] +[ IJBRUT. MONTANT]
+ [ ABSTOTALV. MONTANT] - [ ABSTOTALM. MONTANT] )
```

```

)
sinon (1)
*
si ([ JRSNONTRAV. BASE] <>0)
alors(maximum(1;( 30 - [ JRSNONTRAV. BASE])) /
30)

sinon(1)

)
sinon (
{Forfait Annuel en heures}
si ( CONSTANTE( CONTRAT. SITUATIONHORAIRE) =5)
alors
( CONSTANTE( CONTRAT. HORAIRECOLLECTIF) *CONSTANTE( CONTRAT. HEURE_FORFAIT) /1596
*
si ([ IJBRUT. MONTANT] <>0)
alors
([ BC_BRUT_URSAFF. MONTANT] /([ BC_BRUT_URSAFF. MONTANT] +[ IJBRUT. MONTANT] )
)
sinon (1)
)
{Horaire fixe / Horaire variable}
sinon (
si ( CONSTANTE( CONTRAT. TYPESALARIE) =5)
alors ([ AD_HNEXO. BASE])
sinon ([ NB_HEURESCON. MONTANT])
*
(
([ AGCP_NUMHREMU. MONTANT]
/
[B_PXHEURE. MONTANT]
)
)
+
si
( CONSTANTE( CONTRAT. TYPESALARIE) =5)

alors([ AGCP_HREMUPLUS. MONTANT] * [ AD_HNEXO. BASE] /([ AD_HNEXO. BASE] +[ AD_HEXO. BASE] ) )
sinon ([ AGCP_HREMUPLUS. MONTANT])
)

```

```

)
)
)
sinon ( [NB_HEUREREMU. BASE])
+[ NB_HREMUREGUL. BASE]
-
si ( CONSTANTE( CONTRAT. RAPPEL_FICHE) =1)
alors ( [ H_NONTRAV. BASE])
sinon ( 0)

```

Après

```

si (([NB_HEUREREMU. BASE]=0) ou ( CONSTANTE( CONTRAT. TYPESALARIE) =5) )
alors (
  {Assistante Maternelle}
  si ( CONSTANTE( CONTRAT. NATURECONTRAT) =0003)
  alors (( CONSTANTE( CONTRAT. HORAIRECOLLECTIF)
    )
    *
    (
      si ([ IJBRUT. MONTANT] <>0)
      alors
([ BC_BRUT_URSAFF. MONTANT] /([ BC_BRUT_URSAFF. MONTANT] +[ IJBRUT. MONTANT]) )
      sinon (1)
    )
  )
  sinon (
    {Forfait Annuel en jours}
    si ( CONSTANTE( CONTRAT. SITUATIONHORAIRE) =4)
    alors (
      151.67*
      CONSTANTE( CONTRAT. JOUR_FORFAIT) /218
      *
      si (([ IJBRUT. MONTANT] <>0) ou ([ ABSTOTALV. MONTANT] -
[ ABSTOTALM. MONTANT] <>0))
      alors ([ BC_BRUT_URSAFF. MONTANT]
        /
        ([ BC_BRUT_URSAFF. MONTANT] +[ IJBRUT. MONTANT]
+[ ABSTOTALV. MONTANT] - [ ABSTOTALM. MONTANT] )
      )
    )
  )
)

```

```

sinon (1)
*
si ([ JRSNONTRAV. BASE] <>0)
alors(maximum(1;(30 - [ JRSNONTRAV. BASE])) /
30)

sinon(1)

)

sinon (
{Forfait Annuel en heures}
si (CONSTANTE( CONTRAT. SITUATIONHORAIRE)=5)
alors
(CONSTANTE( CONTRAT. HORAIRECOLLECTIF)*CONSTANTE( CONTRAT. HEURE_FORFAIT) /1596
*
si ([ IJBRUT. MONTANT] <>0)
alors
([ BC_BRUT_URSAFF. MONTANT] /([ BC_BRUT_URSAFF. MONTANT] +[ IJBRUT. MONTANT] )
)
sinon (1)
)
{Horaire fixe / Horaire variable}
sinon (
si (CONSTANTE( CONTRAT. TYPESALARIE)=5)
alors ([ AD_HNEXO. BASE])
sinon ([ NB_HEURESCON. MONTANT])
*
(
([ AGCP_NUMHREMU. MONTANT]
/
[B_PXHEURE. MONTANT]
)
)
+
si
(CONSTANTE( CONTRAT. TYPESALARIE)=5)

alors([ AGCP_HREMUPLUS. MONTANT]*[ AD_HNEXO. BASE] /([ AD_HNEXO. BASE] +[ AD_HEX0. BASE] ))
sinon ([ AGCP_HREMUPLUS. MONTANT])
)
)

```

```
    )  
  )  
  sinon ( [NB_HEUREREMU. BASE])  
  +[NB_HREMUREGUL. BASE]  
  +  
  si ( CONSTANTE( CONTRAT. RAPPEL_FICHE) =1)  
  alors ([H_NONTRAV. BASE])  
  sinon (0)
```

Version 4

Correctif cotisation retraite passage apprenti vers régime général

Correction de la rubrique CHGT_APP : Toujours valorisé à NON

Modification de BC_TU1_RET

Avant

```
F_TRANCHEANew([ BC_BRUT_RET. MONTANT]; [ BC_TU1_RET. MONTANT]; [ PL_MENSA. MONTANT] )
```

Après

```
F_TRANCHEANew([ BC_BRUT_RET. MONTANT]; [ BC_TU1_RET. MONTANT]; [ PL_MENSA. MONTANT]; [ CHGT_APP. BASE] )
```

Modification de la fonction F_TRANCHEB

Transformation du paramètre SOMMEISO en MOISDEB

Avant

```
maximum
  ( minimum(
    F_HISTOCLOT( BRUT)
    +BRUT
    - ( F_HISTOCLOT( TRANCHEA)
      +TRANCHEA
    )
    ;
    F_HISTOCLOT( PLAFB)
    +PLAFB
  )
  ;
  0)
- F_HISTOCLOT( TRANCHEB)
```

Après

```
maximum
  ( minimum(
    F_HISTOCLOT( BRUT; MOISDEB; CONSTANTE( GENERAL. ANNEE_PAYE ) )
    +BRUT
    - ( F_HISTOCLOT( TRANCHEA; MOISDEB; CONSTANTE( GENERAL. ANNEE_PAYE ) )
      +TRANCHEA
    )
    ;
    F_HISTOCLOT( PLAFB; MOISDEB; CONSTANTE( GENERAL. ANNEE_PAYE ) )
    +PLAFB
  )
  ;
  0)
- F_HISTOCLOT( TRANCHEB; MOISDEB; CONSTANTE( GENERAL. ANNEE_PAYE ) )
```

Modification de la rubrique BC_TU2_RET

Avant

```
F_TRANCHEB( [ BC_BRUT_RET. MONTANT ] ; [ BC_TU1_RET. MONTANT ] ; [ BC_TU2_RET. MONTANT ] ; [ PL_MENSB_7PSS. MONTANT ] )
```

Après

```
F_TRANCHEB( [ BC_BRUT_RET. MONTANT ] ; [ BC_TU1_RET. MONTANT ] ; [ BC_TU2_RET. MONTANT ] ; [ PL_MENSB_7PSS. MONTANT ] )
```

Version 5

CSG pour les apprentis

La CSG pour les apprentis sur la partie supérieur à 50% du smic mensuel.

Le pb est que l'assiette CSG est calculée en annuel pour le problème des 4 plafonds.

Pour contourner le problème on va calculer la CSG en mensuel pour les apprentis.

V_CSG.CUM_AV_ABT

Avant

```
minimum(  
    ( F_HISTOCLOT( [ B_COT_CSG_ABT. MONTANT ] ) + [ B_COT_CSG_ABT. MONTANT ] )  
    ;  
    ( F_HISTOCLOT( [ PL_MENSA. MONTANT ] ) + [ PL_MENSA. MONTANT ] ) *4  
    )
```

Après

```
si ( ( CONSTANCE( CONTRAT. DSNINTITULE ) =64 ) ou ( CONSTANCE( CONTRAT. DSNINTITULE ) =65 ) )  
alors (   
    si (   
  
    ( CONSTANCE( CONTRAT. ANNEE_EMBAUCHE ) *100+CONSTANCE( CONTRAT. MOIS_EMBAUCHE ) >=202503 )  
        )  
    alors ( [ B_COT_CSG_ABT. MONTANT ] )  
    sinon ( 0 )  
    )  
sinon (   
    minimum(  
        ( F_HISTOCLOT( [ B_COT_CSG_ABT. MONTANT ] ) + [ B_COT_CSG_ABT. MONTANT ] )  
        ;  
        ( F_HISTOCLOT( [ PL_MENSA. MONTANT ] ) + [ PL_MENSA. MONTANT ] ) *4  
        )  
    )  
    )
```

V_CSG.CUM_NABT

Avant

```
si (( CONSTANTE( CONTRAT. DSNINTITULE) =64) ou ( CONSTANTE( CONTRAT. DSNINTITULE) =65) )
alors (
    si (

( CONSTANTE( CONTRAT. ANNEE_EмбаUCHE) *100+CONSTANTE( CONTRAT. MOIS_EмбаUCHE) >=202503)
        et ([ BC_BRUT_APP. MONTANT] >0)
    )
    alors (
        ( F_HISTOCLOT([ B_COT_CSG_ABT. MONTANT] )+[ B_COT_CSG_ABT. MONTANT] -
[V_CSG. CUM_AV_ABT] )
        +
        ( F_HISTOCLOT([ B_COT_CSG_NABT. MONTANT] )+[ B_COT_CSG_NABT. MONTANT] )
    )
    sinon ( 0)
    )
sinon (
    ( F_HISTOCLOT([ B_COT_CSG_ABT. MONTANT] )+[ B_COT_CSG_ABT. MONTANT] -
[V_CSG. CUM_AV_ABT] )
    +
    ( F_HISTOCLOT([ B_COT_CSG_NABT. MONTANT] )+[ B_COT_CSG_NABT. MONTANT] )
    )
)
```

Après

```
si (( CONSTANTE( CONTRAT. DSNINTITULE) =64) ou ( CONSTANTE( CONTRAT. DSNINTITULE) =65) )
alors (
    si (

( CONSTANTE( CONTRAT. ANNEE_EмбаUCHE) *100+CONSTANTE( CONTRAT. MOIS_EмбаUCHE) >=202503)
        et ([ BC_BRUT_APP. MONTANT] >0)
    )
    alors (
        [ B_COT_CSG_ABT. MONTANT]
        +

```

```

        [ B_COT_CSG_NABT. MONTANT]
    )
sinon (0)
)
sinon (
    ( F_HISTOCLOT([ B_COT_CSG_ABT. MONTANT] )+[ B_COT_CSG_ABT. MONTANT] -
[V_CSG. CUM_AV_ABT] )
    +
    ( F_HISTOCLOT([ B_COT_CSG_NABT. MONTANT] )+[ B_COT_CSG_NABT. MONTANT] )
)

```

BC_CSG

Avant

```

si (( CONSTANTE( CONTRAT. PAS_RESIDENTETRANGER)=1) et ( CONSTANTE( CONTRAT. PAS_ACTIF)=0) )
alors (0)
sinon (
    si (( CONSTANTE( CONTRAT. NATURECONTRAT)=0008) et ([ F0. TOTAL] <>0) )
alors ([ F0. TOTAL] )
sinon (
    si (( [ PLAFONDSECU. NUMCAS]=2) et
(CONSTANTE( CONTRAT. ANNEE_CLOTURE) <2024) )
alors ([ B_COT_CSG. MONTANT] )
sinon (
    [ V_CSG. CUM_ABT]
    +[ V_CSG. CUM_NABT]
    -
F_HISTOCLOT([ BC_CSG. MONTANT] ; [ CHGT_RGAPP. MONTANT] ; CONSTANTE( GENERAL. ANNEE_PAYE) )
)
)
)

```

Après

```

si (( CONSTANTE( CONTRAT. PAS_RESIDENTETRANGER)=1) et ( CONSTANTE( CONTRAT. PAS_ACTIF)=0) )
alors (0)
sinon (
    si (( CONSTANTE( CONTRAT. NATURECONTRAT)=0008) et ([ F0. TOTAL] <>0) )
alors ([ F0. TOTAL] )
)

```

```

sinon (
    si (([PLAFONDSECU. NUMCAS]=2) et
(CONSTANTE( CONTRAT. ANNEE_CLOTURE) <2024) )
    alors ([ B_COT_CSG. MONTANT])
    sinon (
        [ V_CSG. CUM_ABT]
        +[ V_CSG. CUM_NABT]
        -
        si (( CONSTANTE( CONTRAT. DSNINTITULE) =64) ou
(CONSTANTE( CONTRAT. DSNINTITULE) =65) )
        alors (0)
        sinon (
            F_HISTOCLOT([ BC_CSG. MONTANT]; [ CHGT_RGAPP. MONTANT]; CONSTANTE( GENERAL. ANNEE_PAYE) )
                )
            )
        )
    )
)

```

Version 6

FNGS pour les apprentis

Problème de concordance entre la base Assedic et la base FNGS

A priori, il suffit juste de modifier la base de cotisation dans la gestion des régimes.

Version 7

Ticket 742839 : Problème de calcul de la Zrr

Il faut modifier la rubrique libre V_ZRR

MTREDTRANSP

Avant

$$\frac{[V_ZRR.MTREDUCTION] * ([TRANSPORT.TXEMP])}{[B_TXZRR.MONTANT]}$$

Après

$$\frac{[V_ZRR.MTREDUCTION] * ([TRANSPORT.TXEMP])}{[V_ZRR.SOMTXZRR]}$$

MTREDFNAL

Avant

$$\frac{[V_ZRR.MTREDUCTION] * ([FNAL.TXEMP] + [FNAL_TRA.TXEMP] + [FNAL_DEPLAF.TXEMP])}{[B_TXZRR.MONTANT]}$$

Après

$$\frac{[V_ZRR.MTREDUCTION] * ([FNAL.TXEMP] + [FNAL_TRA.TXEMP] + [FNAL_DEPLAF.TXEMP])}{[V_ZRR.SOMTXZRR]}$$