

# NF14657 — GESTION DE LA SAUVEGARDE SEGMENTÉE AVEC POSSIBILITÉ DE SÉLECTIONNER LES DONNÉES À SAUVEGARDER

---

Disponible depuis la version 7.01.0.32015

*Voir la carte de la fonctionnalité : [A classer](#)*

Gestion de la sauvegarde segmentée.

## Sommaire

### 1 Concept

1.1 Fonctionnalité

1.2 Finalité

### 2 Fonctionnement

2.1 Généralités

2.2 A savoir

2.3 Typologie du nom des archives

2.4 Inconsistance de clés

### 3 Paramétrage

# Concept

---

## Fonctionnalité

Lors de la synchronisation d'un poste secondaire, seul un sous-ensemble des données lui est désormais transmis. Les données sont en effet filtrées et il ne reçoit plus notamment :

- selon son paramétrage, certaines données concernant des magasins identifiés (filtrage par magasin) ;
- les photos ;
- les pièces jointes des messages ;
- les journaux (système et audit) ;
- les archives fiscales ;
- et enfin, le contenu des opérations de réplication qui sont déjà incluses dans la base de données ;

Les photos sont transmises dans un flux séparé qui ne bloque pas le travail (voir [NF14658 — Séparation des photos de la base de données](#)), et les pièces jointes sont rapatriées à la consultation de messages (voir [NF14659 — Séparation des pièces jointes de message de la base de données](#)).

## Finalité

Nous avons entrepris la réalisation de la segmentation des données pour les raisons suivantes :

- **gain de temps au déploiement** : lors d'une resynchronisation, nous ne renvoyons plus les données superflues telles que photo, pièce-jointe et journaux qui sont traitées en arrière-plan : les données transmises sont minimales et permettent de reprendre l'activité beaucoup plus rapidement, et ce, même sur une petite connexion Internet ;
- **gain d'espace** : de même, en permettant de ne renvoyer aux magasins que les données qui les concerne sur les grosses structures, nous pouvons diviser la taille des bases de manière substantielle et donc la capacité mémoire de masse (disque dur) des machines utilisées en caisse, leur permettant d'utiliser des disques SSD plus rapide. Les indexes étant réduits, nous diminuons également l'impact sur la mémoire vive des machines, et leur rapidité d'exécution. Une base segmentée dans une grosse ou moyenne structure signifie une caisse plus rapide et moins couteuse, sans perte de fonctionnalité ;
- **sécurité** : la segmentation de gestion / stock apporte toute la sécurité nécessaire dans le cas de franchisés qui ne peuvent plus consulter les chiffres des autres magasins, même en fouillant dans le système : les données ne leur sont plus envoyées ;
- et enfin **respect des obligations légales** : en matière de données personnelles, il est possible de ne plus répliquer le fichier client sur tout les services de réplication, aidant ainsi à satisfaire aux obligations du [RGPD](#), notamment dans le cas de multi-sociétés et de magasins répartis dans plusieurs pays, dont certains en dehors de l'Union Européenne.

# Fonctionnement

---

## A savoir :

**L'option de mandataire (proxy) a été supprimée. Elle a été rendue obsolète avec le retrait des données volumineuses des bases de données.**

## Généralités

1. Lorsqu'il doit synchroniser sa base, le poste secondaire effectue un appel à l'API [WebAPI\\_Bdd.Sync](#) sur le service de réplication superviseur ;
2. ce dernier regarde les autorisations et le paramétrage du poste effectuant la demande, et regarde s'il a une archive prête à lui renvoyer (moins de 24h) ;
3. dans le cas contraire, il répond avec une erreur "*503 ServiceUnavailable*" et prépare une archive générale si manquante, puis une archive destinée à la configuration du poste appelant ;
4. le poste secondaire reçoit soit l'erreur 503, et il bouclera jusqu'à temps d'avoir une archive à télécharger, soit le nom de l'archive à télécharger ;
5. il télécharge l'archive spécifiée à l'aide de l'API [WebAPI Files.Download](#) ; retente plusieurs essais en cas d'erreur ;
6. une fois reçue, l'archive est vérifiée puis déployée ;
7. une fois déployée, le système continue de manière traditionnelle et planifie un téléchargement des photos en arrière-plan.

## A savoir

- les archives sont stockées dans `/var/polaris/backups` ; elles peuvent être effacées manuellement sans que cela pose de soucis ;
- un fichier `.acl` est généré pour chaque archive ; ce dernier est entretenu automatiquement par la méthode `Bdd.Sync`. S'il est effacé ou manquant, le poste ne pourra pas télécharger l'archive, mais le système devrait se rétablir automatiquement ;
- un tag aléatoire est ajouté à chaque nom d'archive pour garantir plus de sécurité, un fonctionnement avec un éventuel futur mandataire (proxy) ; et pour éviter de télécharger par erreur une archive qui aurait été régénérée ;
- tous les fichiers de synchronisation commencent par "sync-" ; ils sont automatiquement générés et supprimés et ne contiennent que la base de données minimales et potentiellement segmentée ;
- vous pouvez télécharger et transférer manuellement ces archives avec un jeton de maintenance en les téléchargeant depuis la zone dédiée de l'[interface de dialogue](#) ;
- le système n'effectue la sauvegarde de la base de données qu'une fois et la place dans une archive nommée "*masterfile*" ; lors de leur demande, les archives segmentées sont construites à partir de celle-ci

## Typologie du nom des archives

Pour la *masterfile* :

```
sync-PL-VEGA-0001-nur-1-masterfile-tag-
b3uuOu3Ju6LpIE3x8sbJuZDl4BbdRDiRN0Zgxx.polaris
  +-> n° licence      |      |
                    |      +-> "masterfile"
                    +-> nur du TLR      |
                                       +-> "tag" aléatoire
```

Et pour une archive segmentée à destination d'un poste :

```
sync-PL-VEGA-0001-nur-1-file-for-16385-49153-tag-
EDOLVhQCVnC3150Toku3ORL0sZ4x.polaris
  +-> n° licence      |      |
                    |      +-> "file-for"
                    +-> nur du TLR      |
                                       +-> "tag" aléatoire
```

Les deux numéros après *file-for* représentent :

1. le numéro interne du groupe de magasins pour les données de stock pour lequel ces données ont été préparées,
2. le numéro interne du groupe de magasins pour les données des clients pour lequel ces données ont été préparées.

## Inconsistance de clés

Quoiqu'il arrive, les clés étrangères seront rétablies, garantissant qu'il est impossible d'insérer des valeurs désynchronisées dans la base de données ; mais il peut arriver que certaines données n'aient pas été renvoyées car segmentées et pourtant référencées par un enregistrement. C'est même le fonctionnement normal pour certaines tables.

Dans ce cas, la clé est positionnée, mais n'est pas validée lors de la restauration, produisant des avertissements dans le journal et pouvant expliquer certains comportements erratiques non prévu du programme.

La liste des clés qui n'ont pas pu être validées est disponible dans [l'Interface de dialogue](#), dans l'onglet "Base de données". Cette information est à titre informatif.

# Paramétrage

---

A savoir :

**Cette option n'est pas fonctionnelle et sera implémentée ultérieurement.**

Pour segmenter les données au niveau magasin, le paramétrage a été placé dans chaque service de réplication, onglet "Filtrage de la réplication".

Vous pouvez spécifier en outre, une fois la segmentation activée :

- un groupe de magasin pour les données de stock (entrées et sorties de stock : commandes, livraisons, ventes, ...)
- un pour les données du fichiers clients ;

Un service de réplication qui a été assigné à un groupe reçoit les données communes et celles des magasins constituant ce groupe.

Un service de réplication qui n'a pas été assigné à un groupe reçoit automatiquement que les données communes et celles du magasin qu'il gère.

Après avoir paramétré ou modifié le paramétrage de la segmentation, il convient de resynchroniser le service, *cette opération n'étant pas automatique.*