Sujets connexes	Fichier de Prélèvement SEPA - aspect technique (ISO 20022) https://fr.wikipedia.org/wiki/ISO_20022

Export XML (prélèvement SEPA)

Cet écran permet de générer un fichier XML de prélèvement SEPA (norme bancaire), à partir d'une écriture comptable multi-ligne déjà existante — typiquement celle produite par l'écran Génération Cotisations, mais pas seulement. Le fichier généré est ensuite à transmettre à la banque pour déclencher les prélèvements.

Accès à l'écran

- Menu **Comptabilité > Utilitaires**
- Sélectionner **Prélèvement SEPA** dans l'arborescence

Organisation de l'écran

1. Choix de la multi-ligne à exporter

Une première grille liste les multi-lignes éligibles : **N° pièce**, **Date**, **Intitulé**, **Total crédit**. Une recherche libre est disponible, mais il n'y a pas de filtre par date : il faut retrouver sa multi-ligne dans la liste.

Seules les multi-lignes de type **Recette** ou **Bordereau**, non supprimées, sont proposées. Les multi-lignes de type **Dépense** (virements) ne sont **pas encore proposées** à ce stade, même si le mécanisme de génération de virement existe déjà techniquement en arrière-plan.

2. Détail des lignes de la multi-ligne sélectionnée

Une seconde grille affiche les lignes d'écriture (saisies) rattachées à la multi-ligne choisie : **Libellé**, **Adhérent**, **Montant**.

3. Paramétrage avant génération

Cette section n'apparaît qu'une fois une multi-ligne sélectionnée. Elle peut d'abord bloquer la suite avec un message si :

- une ou plusieurs lignes ne sont pas liées à un adhérent ;
- la multi-ligne ne contient pas exactement **une seule** ligne sur un compte de trésorerie (compte bancaire, classe 5) — c'est cette ligne qui sert de contrepartie (le compte débité/crédité côté association).

Deux blocs sont ensuite éditables (bouton “Modifier” / “Valider” / “Annuler” — réservé aux utilisateurs ayant le droit d'édition sur les fichiers) :

- **Identifiant Créancier SEPA (ICS) de l'organisation** : un champ unique, au format imposé par la norme SEPA. **Attention : c'est un paramètre global de la base**, pas propre à cette génération — le modifier ici le modifie pour toutes les générations futures.
- **Informations du compte de trésorerie** : IBAN, BIC, code banque, code agence, numéro de compte, clé RIB, domiciliation, informations banque 1 à 4. Le bouton “Valider” n'est actif que si IBAN et BIC sont renseignés. Raccourcis : Échap annule, Entrée valide.

Un tableau récapitulatif (lecture seule) liste ensuite chaque ligne de contrepartie avec : titre, nom de l'adhérent, montant, IBAN/BIC (repris de la fiche personne — “Informations IBAN/BIC manquantes” si absentes), et adresse. Un message global signale s'il manque des coordonnées bancaires sur au moins un adhérent.

4. Génération du fichier

Le bouton de génération s'adapte au type de multi-ligne :

- **Recette / Bordereau** → **“Générer XML pour prélèvement”**, avec un rappel au-dessus : *“Prélèvement sur le compte {compte} - {banque} pour un montant total de {montant} €”*.
- **Dépense** → “Générer XML pour virement” (cas non atteignable pour l'instant, les Dépenses étant exclues en amont).

Le bouton reste désactivé tant que les conditions ne sont pas réunies, qu'il manque des IBAN/BIC, ou qu'une édition (ICS / banque) est en cours.

Si des données sont invalides (nom, adresse, IBAN, BIC, date de mandat…), la génération est bloquée et le détail des erreurs s'affiche sous l'en-tête **“Les erreurs ci-dessous ont empêché la génération du fichier SEPA XML”**.

Si tout est valide, le fichier est **téléchargé directement par le navigateur** (pas d'envoi au serveur), nommé `directDebit_sepa_{ID}.xml` (ou `payment_sepa_{ID}.xml` pour un virement).

Points d'attention

- L'**ICS** est un paramètre **de la base**, pas de la génération en cours : une correction doit être faite une fois pour toutes, pas à chaque export.
- Si la **date de signature du mandat** n'est pas renseignée sur la fiche de l'adhérent, l'export utilise une date par défaut du **01/01/2024** — à vérifier/corriger sur la fiche adhérent pour un mandat plus récent avant de faire confiance à l'export.
- Un IBAN ou un BIC laissé **vide** n'est pas systématiquement signalé comme une erreur de format : c'est le contrôle sur les champs manquants (bouton désactivé, tableau récapitulatif) qui empêche réellement la génération, pas la validation de format en elle-même.
- Le pays de l'adresse d'un adhérent, s'il n'est pas reconnu comme un pays de la zone euro SEPA, retombe automatiquement sur “FR” — à surveiller pour les adhérents étrangers.
- L'écran ne propose aujourd'hui que les prélèvements (Recette/Bordereau) : pour un virement (Dépense), la génération n'est pas encore accessible depuis cette grille.

Annexe technique

- **Composants** : `LogeasGridExportXmlComponent` (`app-logeas-grid-export-xml`, écran principal) → `DetailExportXmlComponent` (wrapper d'affichage conditionnel) → `ExportXmlOutilComponent` (`app-export-xml-outil`, contient toute la logique métier).
- **Filtrage amont** des multi-lignes éligibles : `typeMultiligne === 1` (Recette) ou `5` (Bordereau), `status != 1` (non supprimée). Les Dépenses sont explicitement exclues dans le code (“On n'offre pas les virements SEPA pour l'instant”).
- **Détection de la ligne de trésorerie** : la saisie dont le `codeArticle` commence par `“5”` (compte de classe 5).
- **Stockage des paramètres** : ICS via `DG.BaseCourante.config.setParam$(“ICS”, ...)` (paramètre global) ; coordonnées bancaires via `DG.BaseCourante.config.setInfoBanque$(codeBanque, ...)`, indexées par code de compte bancaire.
- **Organisation créancière** : lue via `CRUD(“READ”, “Secteur”, {“ID”: DG.BaseCourante.IDSecteur})` — le secteur comptable **courant** de l'utilisateur (`data-source.service.ts`, méthode `determinerBanqueOperationTresorerie$`). Corrigé le date : lisait auparavant systématiquement le secteur ID=1, quel que soit le secteur de travail.
- **Mapping et validation** : `DataMapperService.mapAndValidateXmlData(...)` construit les structures `OrganisationDetailsBancaires` (créancier) et `PersonneDetailsBancaires` (par adhérent), en nettoyant les chaînes (`supprimerAccents()` + `nettoyerChaine()`, jeu de caractères latin SEPA `[a-zA-Z0-9 /\-?:().,'+]`). Les règles de validation (nom, IBAN, BIC, cohérence pays/IBAN/BIC, adresse, date de mandat ≤ aujourd'hui, ICS) sont centralisées dans `ValidationXmlBancaireService`.
- **Génération du fichier** : `xml-builder.service.ts`, format ISO 20022 SEPA — `pain.008.001.02` (`CstmrDrctDbtInitn`) pour un prélèvement, `pain.001.001.03` (`CstmrCdtTrfInitn`) pour un virement, construit via la librairie `fast-xml-parser`.
- **Identifiants techniques** (`MsgId`, `PmtInfId`, `EndToEndId`) sont dérivés de l'horodatage (`Date.now()`) : pas de garantie d'unicité stricte en cas de générations multiples à la même milliseconde (cas extrêmement rare en usage réel).
- **Téléchargement** : `FileDownloaderService.downloadFile()`, `Blob` local (`application/xml`) + lien `<a download>` simulé — aucun appel serveur, tout se passe côté navigateur.

From:
https://wiki-logeas.fr/logeas-web/ - **Documentation de Logeas-web.fr (client léger)**

Permanent link:
https://wiki-logeas.fr/logeas-web/doku.php?id=compta:exportxml&rev=1787750620

Last update: **2026/08/26 15:23**