

Guide d'intégration Payt

Alimenter Payt en débiteurs et factures par dépôt de fichiers CSV, pour déclencher le suivi de relance automatisé.

Périmètre : alimentation par fichiers CSV (import par lot) · **Version :** 1.0 · **Date :** juin 2026

Endpoint d'import : <https://backend.paytsoftware.com/import/files/csv>

Sommaire

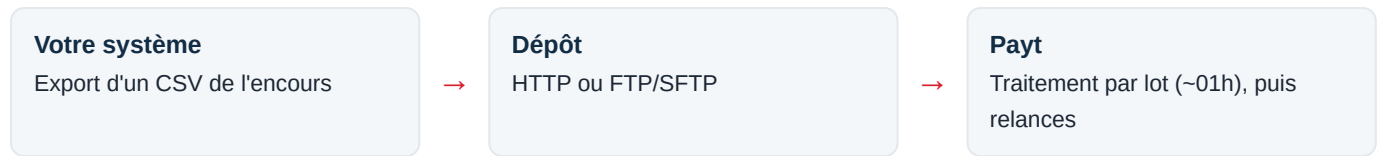
- 1. Objet et périmètre du guide**
- 2. Le modèle « image complète » — le concept central**
- 3. Prérequis, environnements et configuration**
- 4. Format et types de données du fichier CSV**
- 5. Les colonnes : débiteur, facture, contact**
- 6. Paiements (colonne JSON)**
- 7. Lignes de facture (colonne JSON)**
- 8. Dépôt du fichier (HTTP et FTP/SFTP)**
- 9. Documents PDF, UBL et pièces jointes**
- 10. Traitement, erreurs et pièges fréquents**
- 11. Checklist de mise en production**
- 12. Annexes — colonnes et références**

Ce guide décrit l'alimentation de Payt par fichiers CSV. L'alimentation équivalente via l'API REST (écriture ligne à ligne, en flux) fait l'objet d'un document distinct. Les deux voies ne se cumulent pas sur une même administration.

1. Objet et périmètre du guide

Ce guide explique comment votre système (ERP, logiciel de facturation, outil métier) transmet ses débiteurs et ses factures à Payt sous forme de fichiers CSV déposés par lot. Une fois les données reçues et traitées, Payt prend en charge automatiquement les rappels, relances et la collecte.

Le flux en une image



Ce que couvre ce guide

- Le modèle de données par « image complète » propre à l'import CSV et ses conséquences.
- La configuration préalable de l'administration côté Payt.
- Le format, les types de données et les colonnes du fichier CSV.
- Le dépôt du fichier en HTTP ou par FTP/SFTP, et le rattachement des PDF.
- Le traitement quotidien, la gestion des erreurs et les pièges à éviter.

Deux voies qui ne se cumulent pas

Sur une administration donnée, on alimente Payt **soit** par import de fichiers (ce guide), **soit** par l'API REST — jamais les deux. Lorsque l'auto-import est activé, la création de factures via l'API est refusée (`403` avec le message `"Not available with auto import turned on"`). Tranchez une seule voie par administration.

2. Le modèle « image complète » — le concept central

C'est la différence fondamentale avec l'API REST, et la source de la plupart des erreurs d'intégration. À lire avant tout le reste.

L'API REST fonctionne en **incrémental** : on envoie une facture, elle est créée ou mise à jour. L'import CSV, lui, fonctionne en **image complète (snapshot)** : chaque fichier doit présenter **la totalité des factures** que Payt doit traiter (envoi, relance, etc.) à cet instant. Payt compare le fichier à l'état qu'il connaît et en déduit les changements.

La conséquence à retenir absolument

Si une facture **disparaît** du fichier, Payt en déduit qu'elle est **payée** et la solde. Par extension, un **fichier vide** est interprété comme « plus aucune facture en cours » : **toutes les factures sont considérées comme payées**. Un réglage de protection contre le chargement d'un fichier vide est disponible dans l'application — activez-le.

Ce que cela implique pour votre export

- **N'exportez pas seulement « l'encours »**. Un filtre sur les seuls postes ouverts peut être incomplet : une facture créée puis lettrée avec un paiement comptant dans les 24 h sort de l'encours alors que Payt doit peut-être encore l'envoyer. Payt recommande de **conserver une facture dans le fichier pendant un mois** pour ne manquer ni facture, ni avoir, ni facture à zéro.
- **Les paiements sont déduits** de l'écart entre le montant total et le montant ouvert de la facture — vous n'avez pas à envoyer d'évènement de paiement (mais vous pouvez en prendre le contrôle, voir §6).
- **Une facture reste dans chaque fichier** tant qu'elle doit être suivie ; elle n'en sort que lorsqu'elle est réellement soldée.

En résumé

Le fichier CSV n'est pas un « lot de nouveautés » : c'est la photographie complète de vos créances à suivre. Rationnez état, pas évènement.

3. Prérequis, environnements et configuration

Configuration côté Payt (préalable)

Dans l'onglet `Import` de l'administration concernée :

- Activer le rattachement « **sur la base des fichiers CSV fournis** » et « **import automatique** ».
- Renseigner le **code d'administration** (`administration_code`) **fourni par Payt** — c'est la clé que portera chaque fichier.

Le rattachement des PDF de factures se fait par nom de fichier (voir §9) et nécessite un réglage côté Payt, effectué par le spécialiste d'implémentation ou le service desk.

Rythme de traitement

Les fichiers sont relevés et traités **une fois par jour, généralement autour de 01h00**. L'heure exacte figure dans les réglages d'import de l'administration. Si aucun fichier n'est disponible lors de l'import automatique, une notification est ajoutée à la page d'import.

Règles transverses

Règle	Détail
Transport	SSL obligatoire, TLS 1.2 minimum.
Encodage des fichiers de données	UTF-8.
Fichiers binaires (PDF...)	Encodés en Base64 dans le corps JSON.
Débit	Au plus 3 fichiers par seconde ; au-delà, réponse <code>429</code> et fichier à re-soumettre.
Compression	En HTTP, les fichiers <code>.gz</code> sont décompressés automatiquement (gardez l'extension d'origine, ex. <code>20260624.csv.gz</code>). En FTP/SFTP, les archives <code>.zip</code> sont désarchivées (fichiers seulement, les dossiers sont ignorés).

4. Format et types de données du fichier CSV

Structure du fichier

- Encodage **UTF-8**. La spécification **RFC 4180** sert de référence.
- Séparateur : point-virgule (;) **ou** virgule (,).
- Les champs texte sont entourés de **guillemets doubles** ("). Un guillemet à l'intérieur d'un champ se protège en le **doublant** (" ").
- Longueur maximale d'un champ : **255 caractères**.
- La **première ligne** contient les en-têtes de colonnes.
- Certaines colonnes sont **obligatoires** (voir §5) ; les colonnes inutilisées peuvent être omises. **L'ordre des colonnes n'a pas d'importance**.

Exemple minimal (séparateur point-virgule)

```
"administration_code";"debtor_code";"debtor_company_name";"debtor_post_street_1";  
"debtor_post_postalcode";"debtor_post_city";"debtor_post_country_code";"debtor_email";  
"debtor_language_code";"invoice_number";"invoice_date";"invoice_due_date";  
"invoice_total_amount_inc_vat";"invoice_open_amount_inc_vat"  
"ADM-FR-01";"CLI-1001";"Boulangerie Martin SARL";"12 rue des Lilas";  
"01000";"Bourg-en-Bresse";"FR";"compta@boulangerie-martin.fr";  
"fr";"2026-000142";"2026-06-01";"2026-07-01";  
"1200.00";"1200.00"
```

Les retours à la ligne ci-dessus ne sont là que pour la lisibilité : dans le fichier réel, chaque enregistrement tient sur une seule ligne.

Types de données

Type	Description	Exemple
boolean	"0" / "false" pour faux, "1" / "true" pour vrai	false
country_code	ISO 3166-1 Alpha-2	FR
currency_code	ISO 4217 (3 lettres)	EUR
date	yyyy-mm-dd	2026-06-01
integer	Nombre entier	7
decimal	Deux décimales max, séparateur point	1200.00
email_address	Adresse email, validée par motif	info@exemple.fr
gender_code	f / female ou m / male	female
JSON	JSON valide, correctement échappé dans le champ CSV	{ "key": "value" }
language_code	ISO 639-1 (2 lettres)	fr
phone_number	Format international recommandé ; validé (avertissement, jamais erreur)	+33612345678
postal_code	Requis et validé par motif pour NL, BE, DE, FR, LU	01000
text	255 caractères max	abc def

Langues gérées dans l'application : néerlandais, allemand, anglais, français, espagnol, italien, polonais. Pour tout autre code, l'application retombe sur la langue de repli définie sur l'administration.

5. Les colonnes : débiteur, facture, contact

Chaque ligne du fichier réunit, à plat, un débiteur et une facture. Il n'y a pas de ressources séparées comme dans l'API : tout est dans les colonnes.

Colonnes obligatoires

Colonne	Rôle
<code>administration_code</code>	Identifiant de l'administration, fourni par Payt (voir l'encadré ci-dessous).
<code>debtor_code</code>	Identifiant lisible unique du débiteur.
<code>debtor_company_name</code> ¹	Raison sociale du débiteur.
<code>invoice_number</code>	Identifiant lisible unique de la facture.
<code>invoice_date</code>	Date comptable de la facture.
<code>invoice_due_date</code>	Date d'échéance.
<code>invoice_total_amount_inc_vat</code>	Montant total TTC, dans la devise de la facture.
<code>invoice_open_amount_inc_vat</code>	Montant restant dû TTC, dans la devise de la facture.

¹ Pour un débiteur particulier (personne physique), `debtor_firstname` + `debtor_lastname` peuvent tenir lieu de `debtor_company_name`. Ces champs prénom/nom ne sont pas obligatoires en eux-mêmes.

administration_code : une administration = une entité

Payt attribue un `administration_code` par administration, et **chaque administration correspond à une entité (filiale)**. Si vous alimentez plusieurs filiales, associez chacune au bon `administration_code` et portez-le dans chaque ligne du fichier correspondant. Ce même code est à renseigner côté Payt dans l'onglet Import (§3).

Au moins un canal de contact par débiteur

Pour que Payt puisse joindre le débiteur, fournissez **au moins l'un** des éléments suivants : une **adresse email** (`debtor_email`), une **adresse postale complète** (`debtor_post_street_1` + `debtor_post_postalcode` + `debtor_post_city` + `debtor_post_country_code`), ou un **numéro de téléphone** (`debtor_phone_number` / `debtor_sms_number`). **Privilégiez l'email** : c'est le canal le plus efficace et le moins coûteux.

Colonnes débiteur fréquemment utilisées (optionnelles)

Colonne	Rôle
<code>debtor_origin_id</code>	Identifiant stable et immuable du débiteur. À défaut, <code>debtor_code</code> est utilisé — auquel cas ce dernier ne doit pas changer.
<code>debtor_is_company</code>	Entreprise ou particulier (<code>boolean</code>).
<code>debtor_email</code> · <code>debtor_invoice_email</code> · <code>debtor_reminder_email</code>	Adresses email (générale, factures, relances) — sans email, pas de relance par email.
<code>debtor_language_code</code>	Langue de communication avec le débiteur.
<code>debtor_vat_number</code> · <code>debtor_coc_number</code>	N° de TVA / d'immatriculation.
<code>debtor_payment_condition</code>	Identifiant de condition de paiement du système source, mappé vers un flux via l'interface.
<code>debtor_category</code>	Catégorie de débiteur (filtre dans l'application ; 50 car. max).
<code>debtor_credit_limit_amount</code>	Encours maximal autorisé (<code>decimal</code> ≥ 0 ; 0 = aucun encours autorisé).

Cohérence des champs débiteur

Les champs `debtor_*` doivent être **identiques sur toutes les factures d'un même débiteur** (même adresse, mêmes emails, même gestionnaire...). Pour un besoin propre à une facture, utilisez plutôt les champs `invoice_contact_*`.

Colonnes facture fréquemment utilisées (optionnelles)

Colonne	Rôle
<code>invoice_currency_code</code>	Devise de la facture. Si la colonne est présente, elle doit être remplie.
<code>invoice_book_amount_total</code> · <code>invoice_book_amount_open</code>	Montants total / ouvert dans la devise comptable de l'administration. Si la colonne est présente, elle doit être remplie.
<code>invoice_description</code> · <code>invoice_reference</code>	Description ; référence par laquelle le débiteur reconnaît la facture.
<code>invoice_payment_condition</code>	Condition de paiement au niveau facture — prioritaire sur <code>debtor_payment_condition</code> .
<code>invoice_payment_method</code>	<code>bank_transfer</code> , <code>direct_debit</code> ou <code>cash</code> .
<code>invoice_payment_term</code>	Nombre de jours pour payer. À défaut, calculé entre <code>invoice_date</code> et <code>invoice_due_date</code> .
<code>invoice_category</code>	Catégorie de facture (filtre ; 50 car. max).
<code>invoice_order_number</code> · <code>invoice_purchase_order_number</code>	N° de commande / n° de bon de commande.
<code>invoice_paused</code>	Met la facture en pause. Une facture mise en pause par cette colonne ne peut être réactivée que par un nouvel import où la colonne vaut <code>false</code> .
<code>invoice_origin_id</code>	Identifiant base immuable de la facture.
<code>payments</code>	Paievements de la facture (<code>JSON</code>) — voir §6.
<code>invoice_lines</code>	Lignes de la facture (<code>JSON</code>) — voir §7.

Contact spécifique à une facture

Les champs `invoice_contact_*` permettent d'attacher à une facture un contact dont les coordonnées **remplacent celles du débiteur** pour cette facture. Si un contact spécifique est fourni, alors `invoice_contact_origin_id` **et** (`invoice_contact_firstname` **ou** `invoice_contact_lastname`) doivent être renseignés.

Si `invoice_contact_origin_id` est fourni mais que le nom manque, le contact n'est pas chargé et une erreur de validation s'affiche — la facture et le débiteur, eux, sont bien chargés. Une cascade de repli s'applique ensuite pour l'adresse email (contact facture → débiteur facture → contact général → débiteur général).

6. Paiements (colonne JSON)

Par défaut, si vous ne fournissez pas de paiements, Payt en crée un automatiquement, égal à l'écart entre le montant total et le montant ouvert de la facture (avec pour date celle de l'import). La colonne `payments` vous permet de **prendre le contrôle** des paiements et de refléter les modifications de votre comptabilité.

Règles impératives

- Fournissez **tous** les paiements de la facture, **y compris les historiques** — Payt détecte ainsi les paiements modifiés ou supprimés dans votre comptabilité.
- Ordre **chronologique**, du plus ancien au plus récent.
- La **somme** des montants doit être égale à l'écart entre le montant total et le montant ouvert de la facture.

Champs d'un paiement

Champ	Rôle
<code>origin_identifier</code> *	Identifiant du paiement, unique dans l'administration.
<code>payment_date</code> *	Date de la transaction.
<code>amount</code> *	Montant payé, dans la devise de la facture.
<code>book_amount</code>	Montant dans la devise de l'administration. Calculé via le taux de change si absent. Requis si <code>transaction_type = revaluation</code> .
<code>transaction_type</code>	<code>payment</code> , <code>reversal</code> , <code>credit</code> , <code>write_off</code> , <code>settlement</code> , <code>revaluation</code> . Pour <code>revaluation</code> , <code>amount</code> doit valoir 0 et <code>book_amount</code> être non nul.
<code>payment_method</code>	Pertinent uniquement pour <code>transaction_type = payment</code> .
<code>reversal_code</code>	Code bancaire du rejet, pour <code>transaction_type = reversal</code> .

Dans une cellule CSV (guillemets doublés)

```
invoice_number;payments
"2026-000142";"[{"origin_identifier":"bank-tx-99812","amount":"1200.00",
"payment_date":"2026-06-20","transaction_type":"payment",
"payment_method":"bank_transfer"}]"
```

Pour une facture sans paiement, fournissez un tableau vide : `[]`.

7. Lignes de facture (colonne JSON)

Les lignes de facture (`invoice_lines`) décrivent les produits et services facturés. Elles ne sont **pas nécessaires** pour la plupart des fonctions de Payt — mais elles deviennent **obligatoires** si vous voulez que Payt génère le document de facture ou l'UBL.

Règles impératives

- La **somme** des montants des lignes doit être égale au montant total de la facture.
- Toute modification détectée est appliquée ; les lignes absentes du nouvel ensemble sont **supprimées**. Un ensemble vide supprime toutes les lignes.
- Une fois la facture publiée (envoyée au débiteur), les lignes **ne peuvent plus être modifiées**.

Champs d'une ligne

Champ	Rôle
<code>description</code> *	Libellé du produit ou service.
<code>product_price</code> *	Prix unitaire.
<code>product_unit</code> · <code>product_quantity</code>	Unité (défaut « piece ») et quantité (défaut « 1 »).
<code>total_excl_tax_amount</code> · <code>total_incl_tax_amount</code>	Montant HT / TTC. Au moins l'un des deux ; l'autre est calculé.
<code>vat_rate_code</code> *	Code du taux de TVA — doit être présent dans l'administration .
<code>vat_percentage</code> · <code>vat_amount</code>	Taux (doit correspondre au <code>vat_rate_code</code>) et montant de TVA ; calculés à défaut.
<code>line_order</code>	Position de la ligne. Si fourni, chaque ligne doit avoir un entier unique.

Dans une cellule CSV

```
invoice_number;invoice_lines
"2026-000142";["{"description":"Prestation de conseil","product_price":"1000.00",
"product_quantity":"1","vat_rate_code":"STD20","vat_percentage":"20.0",
"total_excl_tax_amount":"1000.00","total_incl_tax_amount":"1200.00",
"vat_amount":"200.00","line_order":1}"]
```

8. Dépôt du fichier (HTTP et FTP/SFTP)

Dépôt en HTTP

Envoyez une requête `POST` à l'endpoint correspondant au type de fichier (§9 pour les PDF). Un code `2xx` indique le succès ; sinon, le dépôt a échoué et doit être retenté.

```
POST https://backend.paytsoftware.com/import/files/csv
Accept-Charset: UTF-8
Accept: application/json
Content-Type: application/json

{
  "file": {
    "filename": "20260624.csv",
    "base64_data": "ImFkbWluaXN0cmF0aW9uX2NvZGUiOyJkZWJ0b3JfY29kZSI7..."
  },
  "administration_code": "ADM-FR-01",
  "import_token": "4e1243bd22c66e76c2ba9eddc1f91394e57f9f83"
}
```

Champ	Rôle
<code>file.filename *</code>	Nom du fichier (ex. daté <code>20260624.csv</code>).
<code>file.base64_data *</code>	Contenu du fichier encodé en Base64.
<code>administration_code</code>	Code de l'administration. On peut utiliser <code>administration_id</code> à la place.
<code>import_token</code>	Jeton d'import lié au compte de l'administration.

Piège Base64 fréquent

De nombreuses bibliothèques Base64 insèrent un retour à la ligne tous les 76 caractères (héritage des e-mails). Ces retours à la ligne sont **invalides dans une chaîne JSON** et doivent être **retirés**.

Dépôt par FTP/SFTP

Les fichiers peuvent aussi être déposés par FTP/SFTP. Dans ce cas, les archives `.zip` sont automatiquement désarchivées dans le même répertoire (les fichiers uniquement ; les dossiers internes à l'archive sont ignorés).

Débit

Au plus 3 fichiers par seconde. Au-delà, réponse `429` : le fichier doit être re-soumis.

9. Documents PDF, UBL et pièces jointes

Les documents s'envoient sur des endpoints dédiés, dans le même format d'enveloppe JSON (fichier en Base64, `administration_code` / `administration_id`, éventuel `import_token`).

Type de fichier	Endpoint
Fichier de données CSV / XML	<code>.../import/files/csv</code>
PDF de facture	<code>.../import/files/invoice_pdf</code>
UBL de facture	<code>.../import/files/invoice_ubl</code>
Pièce jointe de facture	<code>.../import/files/invoice_attachment</code>
Fichier de déclaration (MZ301/AP304/ZIP)	<code>.../import/files/declaration_file</code>

Base commune : `https://backend.paytsoftware.com`

Rattachement des PDF par nom de fichier

Les PDF sont rattachés à leur facture d'après le **nom du fichier**, qui doit contenir le **numéro exact** de la facture.

Exemple : `141234.PDF` se rattache à la facture `141234`. Ce rattachement nécessite un réglage préalable côté Payt (spécialiste d'implémentation ou service desk).

10. Traitement, erreurs et pièges fréquents

Comment le fichier est traité

- Traitement par lot, une fois par jour (~01h00 ; heure exacte dans les réglages d'import).
- Absence de fichier lors de l'import automatique : une notification est ajoutée à la page d'import.
- Les erreurs de validation sont visibles dans l'interface. Une ligne au contact spécifique incomplet charge tout de même la facture et le débiteur (mais pas le contact).

Pièges fréquents

À vérifier en priorité

- **Image complète, pas delta** — chaque fichier contient toutes les factures à suivre. Une facture qui disparaît est soldée ; un fichier vide solde tout (activez la protection contre le fichier vide).
- **N'exportez pas que l'encours strict** — gardez les factures un mois dans le fichier pour ne pas manquer les cas limites (facture créée puis lettrée dans les 24 h).
- **Retours à la ligne Base64** à retirer avant l'envoi HTTP.
- **Guillemets doublés** dans les cellules JSON (`payments` , `invoice_lines`) et pour tout guillemet interne à un champ.
- **Cohérence `debtor_*`** sur toutes les factures d'un même débiteur ; sinon, passez par `invoice_contact_*` .
- **Code TVA préexistant** — un `vat_rate_code` utilisé en ligne de facture doit exister dans l'administration.
- **Numéro de facture unique par administration** — le même numéro peut exister dans une autre administration.
- **Auto-import et API s'excluent** — voir §1 (`403 "Not available with auto import turned on"`).

11. Checklist de mise en production

#	Vérification
1	Onglet Import configuré : rattachement « sur la base des fichiers CSV », import automatique activé, <code>administration_code</code> saisi.
2	Protection contre le chargement d'un fichier vide activée.
3	Export en image complète (toutes les factures à suivre), avec conservation ~1 mois — pas seulement l'encours strict.
4	Fichier UTF-8, RFC 4180, séparateur cohérent, champs entre guillemets, guillemets internes doublés.
5	Colonnes obligatoires présentes et remplies ; chaque débiteur dispose d'au moins un canal de contact (idéalement l'email).
6	Codes TVA nécessaires présents dans l'administration (si <code>invoice_lines</code> utilisées).
7	Champs <code>debtor_*</code> cohérents pour un même débiteur.
8	Base64 sans retours à la ligne ; débit ≤ 3 fichiers/s (gestion du <code>429</code>).
9	Rattachement des PDF par nom de fichier réglé côté Payt (si documents envoyés).
10	Une seule voie d'alimentation (CSV ou API) par administration.
11	Traitement quotidien vérifié sur l'environnement de démonstration avant bascule.

12. Annexes

Annexe A — Colonnes obligatoires

Colonne	Type
<code>administration_code</code> (fourni par Payt)	text
<code>debtor_code</code>	text
<code>debtor_company_name</code> ¹	text
<code>invoice_number</code>	text
<code>invoice_date</code> · <code>invoice_due_date</code>	date
<code>invoice_total_amount_inc_vat</code> · <code>invoice_open_amount_inc_vat</code>	decimal

¹ Pour un particulier, `debtor_firstname` + `debtor_lastname` peuvent tenir lieu de `debtor_company_name` ; ces deux champs ne sont pas obligatoires en eux-mêmes.

Canal de contact — au moins l'un (idéalement l'email)

`debtor_email`, **ou** une adresse postale complète (`debtor_post_street_1` + `debtor_post_postalcode` + `debtor_post_city` + `debtor_post_country_code`), **ou** un téléphone (`debtor_phone_number` / `debtor_sms_number`).

Annexe B — Endpoints d'import

Type	Endpoint (base <code>backend.paytsoftware.com</code>)
CSV / XML	<code>/import/files/csv</code>
PDF de facture	<code>/import/files/invoice_pdf</code>
UBL de facture	<code>/import/files/invoice_ubl</code>
Pièce jointe	<code>/import/files/invoice_attachment</code>
Fichier de déclaration	<code>/import/files/declaration_file</code>

Annexe C — Références

Sujet	Page
Imports (vue d'ensemble)	<code>docs.paytsoftware.com/imports/overview</code>
Format CSV	<code>docs.paytsoftware.com/imports/csv</code>
Format XML	<code>docs.paytsoftware.com/imports/xml</code>
Format JSON	<code>docs.paytsoftware.com/imports/json</code>
Adresses IP (allowlist)	<code>docs.paytsoftware.com/reference/ip-addresses</code>