

www.afnor.org

En tant que titulaire des droits d'auteurs sur ce document, ayant-droit ou distributeur autorisé de ce document, AFNOR autorise la consultation et le téléchargement selon les droits qui vous sont alloués pour votre abonnement ou votre achat.

Tous autres droits relatifs à ces documents sont réservés.

AFNOR s'oppose expressément à toute intégration, transmission ou absorption totale ou partielle du présent document par des moteurs ou algorithmes d'Intelligence Artificielle (IA). AFNOR s'oppose également à toute fouille de textes et de données ou création dérivée produite par une IA et basée sur le présent document.

As the copyright holder of this document or authorized distributor, AFNOR authorizes the consultation and downloading of the document as per the rights allowed for your subscription or purchase.

All other rights related to these documents are reserved.

AFNOR, as copyright holder or authorized distributor, expressly objects to any integration, transmission or absorption, in whole or in part, of the present document by Artificial Intelligence (AI) engines or algorithms. AFNOR is also opposed to any text and data mining or derivative creation produced by an AI and based on the present document.

AFNOR

Le : 02/07/2026 à 10:14

Diffusé avec l'autorisation de l'éditeur

Distributed under licence of the publisher



**DOCUMENT PROTÉGÉ
PAR LE DROIT D'AUTEUR**

Droits de reproduction réservés. Sauf prescription différente, aucune partie de cette publication ne peut être reproduite ni utilisée sous quelque forme que ce soit et par aucun procédé, électronique ou mécanique, y compris la photocopie et les microfilms, sans accord formel.

Contacter :

AFNOR – Norm'Info

11, rue Francis de Pressensé

93571 La Plaine Saint-Denis Cedex

Tél : 01 41 62 76 44

Fax : 01 49 17 92 02

E-mail : norminfo@afnor.org

afnor

AFNOR FE : Facture électronique

Normalisation française

Norme expérimentale publiée par AFNOR

XP Z12-013

**API pour interfacier les systèmes d'informations des entreprises
avec les Plateformes de Dématérialisation Partenaires**

Date de publication : juin 2026

« En tant que titulaire des droits d'auteurs sur ce document, AFNOR autorise la consultation et le téléchargement.

Tous autres droits relatifs à ces documents sont réservés.

AFNOR s'oppose expressément à toute intégration, transmission ou absorption totale ou partielle du présent document par des moteurs ou algorithmes d'Intelligence Artificielle (IA).

AFNOR s'oppose également à toute fouille de textes et de données ou création dérivée produite par une IA et basée sur le présent document. »

Éditée et diffusée par l'Association Française de Normalisation (AFNOR) - 11, rue Francis de Pressensé -
93571 La Plaine Saint-Denis Cedex Tél.: + 33 (0)1 41 62 80 00 - Fax : + 33 (0)1 49 17 90 00 - www.afnor.org

XP Z12-013 – Standardisation des API SI/SC ⇔ PA

Sommaire

Avant-propos.....	4
Introduction	9
1 Domaine d'application	10
2 Références normatives	11
3 Termes et définitions.....	12
4 Règles communes à toutes les routes API	15
4.1 Respect des SWAGGERS	15
4.2 Authentification.....	16
4.3 Pagination.....	16
4.4 Route GET / healthcheck	16
4.5 Taille maximale d'un flux.....	16
5 API – Flux.....	17
5.1 Respect du Swagger Flow Service.....	17
5.2 Déposer un flux (Route POST / Flows)	17
5.2.1 Flux acceptés.....	17
5.2.2 'trackingId' [Champ optionnel]	17
5.2.3 'sha256' [champ optionnel].....	17
5.2.4 'processingRule' et 'processingRuleSource'	18
5.2.5 Retour de la route	19
5.2.6 Traitement attendu : la qualification du flux avec un 'FlowType'	20
5.3 Rechercher un flux (Route POST / Search).....	22
5.3.1 Gestion du différentiel.....	24
5.3.2 Garantir l'exhaustivité des flux.....	24
5.4 Récupérer un flux (Route GET / flows / {flowId})	24
5.4.1 Paramètre 'docType'	24
5.4.2 Récupération du lisible d'une facture	25
5.5 Gestion des flux irrecevables et des rejets.....	25
5.5.1 Irrecevabilité d'un flux envoyé par le Client API au Fournisseur API.....	25
5.5.2 Rejet d'un flux facture par la PAe ou la Par	26
5.5.3 Rejet d'un flux Cycle de Vie sur une Facture.....	26
5.6 Informé le Client API de l'arrivée ou de la modification d'un flux par un Webhook.....	26
5.6.1 Définition	26
5.6.2 Abonnement à un Webhook.....	28
5.6.3 Consommation de l'évènement Webhook.....	29
6 API – Annuaire.....	31
6.1 Respect du Swagger Directory Service.....	31
6.2 Consultation de l'Annuaire	31
6.2.1 Routes.....	31
6.2.2 Propriété DirectoryLineStatus.....	32
6.2.3 Propriété isSalesProspectingForbidden.....	33
6.3 Mise à jour de l'Annuaire.....	33
7 Gestion de la délégation.....	33
7.1 Principes généraux.....	33
7.2 Exigences de la gestion de la délégation.....	34
7.3 Périmètre de la délégation	34

XP Z12-013

7.4	Processus de délégation au tiers au travers d'un client id / client secret pour chaque entreprise	34
7.4.1	Description du processus	34
7.4.2	Conformité du processus	34
7.5	Processus de délégation avec une authentification unique par Tiers au sein de chaque Fournisseur API.....	36
7.5.1	Description du processus	36
7.5.2	Conformité du processus	36
7.5.3	Indiquer le périmètre organisationnel de l'appel API	36
	Annexe A Swagger de l'API Flux (Normative)	37
	Annexe B Swagger de l'API Annuaire (Normative)	38
	Annexe C Diagrammes de séquence (Normative)	39
	Bibliographie.....	44

XP Z12-013 – Standardisation des API SI/SC ⇔ PA

Avant-propos

Le présent document est destiné à tous les organismes qui souhaitent échanger des factures électroniques dans le contexte de la réglementation française (Réforme de la Facture Électronique telle que décrite aux articles 289, 289BIS, 290 et 290A du Code Général des Impôts), mais aussi plus largement dans le respect des dispositions de la Directive 2006-112-CE, modifiée par la Directive UE 2025/516 dite ViDA (VAT in the Digital Age).

Le présent document présente des API destinées à un écosystème d'acteurs impliqués dans la facturation électronique et le e-reporting, visant à standardiser les échanges de données, messages ou documents impliqués dans le traitement des factures et du e-reporting (factures, cycle de vie, annuaire, e-reporting) et à faciliter l'interopérabilité entre les systèmes d'information des entreprises au sens large, c'est-à-dire intégrant les solutions de gestion hébergées soit en propre soit en nuage (y compris les SC), mais aussi des solutions de tiers autorisés (experts-comptables, affactureurs, agents d'acheteurs, tiers payeurs, ...) et les PA.

Le présent document présente une version Beta de ces API standardisées, pour objectif de revue publique et commentaires d'un mois.

D'autres versions seront publiées dans les prochains mois pour tenir compte d'une part des retours et de l'expérimentation d'implémentation, et d'autre part des autres travaux de la Commission de Normalisation Facture Électronique, notamment sur l'instruction des cas d'usage et l'impact sur les formats et profils, en accompagnement du déploiement opérationnel de la Réforme Facture Électronique en France, et de la mise en œuvre de la généralisation de la facture électronique en Union Européenne et au-delà.

XP Z12-013**GESTION DE VERSION**

N° de Version	Date de Version	Description des évolutions
V0.1	2025 06 02	Version Beta
V1.1	2025 10 31	Version révisée avec les évolutions suivantes : 3 Références normatives Ajout de la RFC 9421 pour la gestion des Webhook 3 Termes et définitions Ajout des définitions ci-dessous : • FRR • Webhook • XML Schema / XSD 4 Règles communes à toutes les routes API • Paramètre 'offset' non disponible pour la pagination lorsque le jeu de données est dynamique. • Précision sur le degré de liberté de l'URL mise à disposition par le Fournisseur API 5 API Flux Route POST / Flows • Précision sur le multi-part • Empreinte numérique : nouvelle propriété 'sha256' • Nouveau chapitre "Qualification du flux avec un 'FlowType'" • Nouvelles propriétés 'processingRule' et 'processingRuleSource' • Ajout de 6 nouveaux FlowType en remplacement de TransactionReport et de PaymentReport Route POST / Search • Nouveau chapitre pour "Garantir l'exhaustivité des flux" • Suppression du paramètre 'Offset' • Suppression de la capacité à rechercher un flux avec un 'flowId'. Il faut désormais passer par la route GET / Flows Route GET / Flows • Ajout du docType 'MetaData' dans le paramètre 'docType' pour récupérer l'objet représentant un Flux. Nouveau chapitre sur "La gestion des flux irrecevables et des rejets" Nouveau chapitre sur "Informer le Client API de l'arrivée ou de la

XP Z12-013 – Standardisation des API SI/SC ⇔ PA

N° de Version	Date de Version	Description des évolutions
		modification d'un flux par un Webhook" 5 API Annuaire Restriction sur la consultation de l'annuaire à des recherches sur un SIREN complet pour éviter des problèmes de performance. Suppression de la route PATCH/directory-line. Annexes Création des annexes A et B et C
V1.2	2026 02	4 Règles communes à toutes les routes API • Authentification basée sur un « client id » et un « client secret » • Règles de sécurisation 5 API Flux Route POST / Flows • Taille des flux limitée à 100 mo • Limites acceptées pour la pagination • Ajout des processing rule ○ B2G : flux qui relève du e invoicing des ventes B2G ○ B2GOutOfScope : flux B2G « hors réforme » (Exemple : Frais de justice) ○ Bint2G : flux qui venant d'une société non française à transmettre à l'administration française. Webhook : Nouvelle annexe présentant une préversion non normative des webhook 6 API Annuaire API de consultation de l'annuaire : • Suppression des routes impliquant un id-instance • Nouvelle propriété directoryLineStatus • Nouvelle propriété isSalesProspectingForbidden API de mise à jour de l'annuaire : Suppression des routes. 7 Gestion de la délégation Nouveau chapitre sur la délégation
V1.3	2026-06-30	6 API Annuaire • Ajout d'une règle permettant de définir la ligne d'adressage à retourner dans le GET /v1/directory-

XP Z12-013

N° de Version	Date de Version	Description des évolutions
		line/code:{addressing-identifier} • Possibilité de ne pas gérer le retour du nombre de résultats sur les routes de recherche (-1 si non géré) • Supprimer la réponse 206 et les codes d'erreur 204 pour les recherches • Ajout « include » à la requête POST /v1/directory-line/search • Ajout du chemin de base à l'URL du serveur • Gestion de « legalUnit » et « facility » • Valeur par défaut « ignore » à 0 • Ajout de l'opérateur « startWith » à « businessName », « name », « addressLines » et « postalCode » • Correction pour faire correspondre les exemples au schéma pour le tri • Correction sur le code 'O' qui correspond à « Diffusible » 5 API Flux Suppression de l'Annexe D sur le Webhook pour intégration au chapitre 5.6 de la norme. Ajout de flowType pour qualifier les flux de e-Reporting entre le Fournisseur API et le Concentrateur Public du PPF. Séparation du Tableau 3 entre les flowType dédiés à qualifier les flux : • entre le Client API et le Fournisseur API - (Tableau 3.A) • et les flux entre le Fournisseur API et le Concentrateur Public du PPF (Tableau 3.B) Suppression du flowType StateInvoice et ajout de : • StateCustomerInvoice • StateSupplierInvoice • StateTransactionReport • StateTransactionReportLC • StatePaymentReport • StatePaymentReportLC Ajout d'un système de pagination basé sur un cursor sur la Route

XP Z12-013 – Standardisation des API SI/SC ⇔ PA

N° de Version	Date de Version	Description des évolutions
		Search Augmentation de la longueur de NotOnlyUuid à 64 caractères Mise à jour de la configuration du webhook : • Suppression de processingRule des métadonnées • Suppression des champs de métadonnées obligatoires (les métadonnées peuvent être vides) • Simplification de la configuration (suppression des en-têtes, du bloc d'authentification et de signature) • Amélioration de la flexibilité des chemins d'URL Ajout des champs obligatoires manquants pour : processingRule et flowProfile dans le schéma Flow des réponses Ajout du chemin de base et du nom du service comme variables dans l'URL du serveur Ajout des valeurs « Undefined » pour « ProfileType », « FlowType » et « ProcessingRule ». Ces trois propriétés ne peuvent pas être définies dans les états En attente et Erreur.

XP Z12-013

Introduction

Le contexte de ce travail de standardisation est la mise en place de la facturation électronique obligatoire en France.

L'objectif des API (Interface de programmation applicative) standardisées est de permettre une interface standard entre le système d'information (SI) de l'entreprise dans son ensemble intégrant aussi les SC (Solution Compatible) d'une part, et les Plateformes Agréées (PA) d'autre part, de façon à construire une interopérabilité de bout en bout permettant à toutes les solutions de l'entreprise de se connecter d'une façon standardisée aux Plateformes Agréées qui implémenteront ces API.

Les API standardisées permettent de répondre aux enjeux techniques et fonctionnels suivants :

- a) La sécurité d'accès avec un protocole d'autorisation basé sur OAuth2
- b) Une segmentation des flux, illustrée par la figure 1 ci-dessous :
 - Les flux entrants (émis par la Plateformes Agréées vers l'entreprise) versus les flux sortants (transmis par l'entreprise vers la Plateformes Agréées)
 - Les factures de Ventes versus factures d'Achats
 - Les accès par des tiers : l'expert-comptable etc...

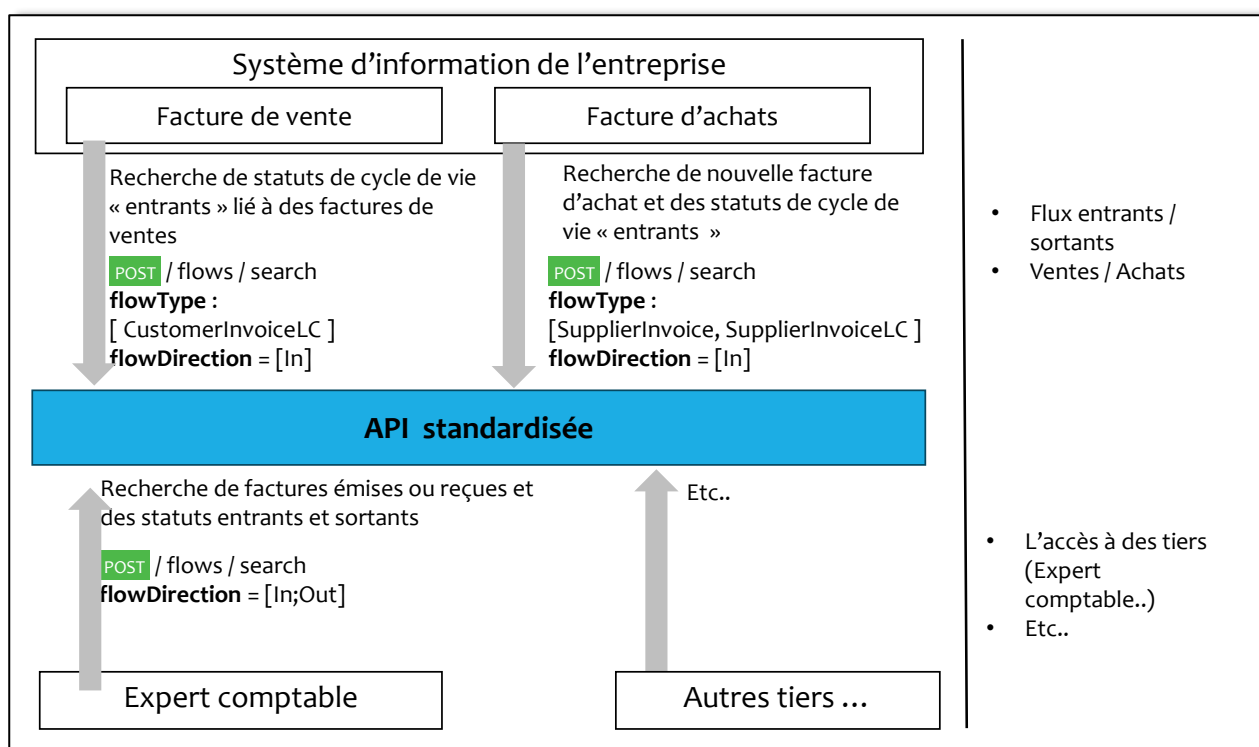


Figure 1 — Segmentation des Flux

- c) La performance avec :
 - Des API fonctionnant pour un consommateur d'API en SAAS ou OnPremise (Sur site)
 - Des API conçues pour un traitement des flux en asynchrone pour faciliter la montée en charge.

XP Z12-013 – Standardisation des API SI/SC ⇔ PA

1 Domaine d'application

Le présent document vise à décrire des API (Interface de programmation applicative) standardisées et les règles applicables afin de faciliter les échanges entre les différents acteurs de la Réforme de la Facture électronique en France. Ces API sont des interfaces communes permettant :

- L'interaction des Solutions Compatibles (SC) et les Systèmes d'Information (SI) des entreprises avec l'ensemble des Plateformes Agréées (PA). L'objectif est de faciliter l'interconnexion des SC et des systèmes d'information des entreprises en assurant des API cohérentes pour la connexion à une ou plusieurs Plateformes Agréées.
- D'assurer l'interopérabilité pour les entreprises qui souhaiteraient passer d'une Plateformes Agréées à une autre.
- De faciliter l'accès aux factures et aux données stockées dans les Plateformes Agréées par des tiers habilités par les entreprises (tels que les experts-comptables, les affactureurs, les tiers payeurs, les agents d'acheteurs, les sociétés holding, ...).

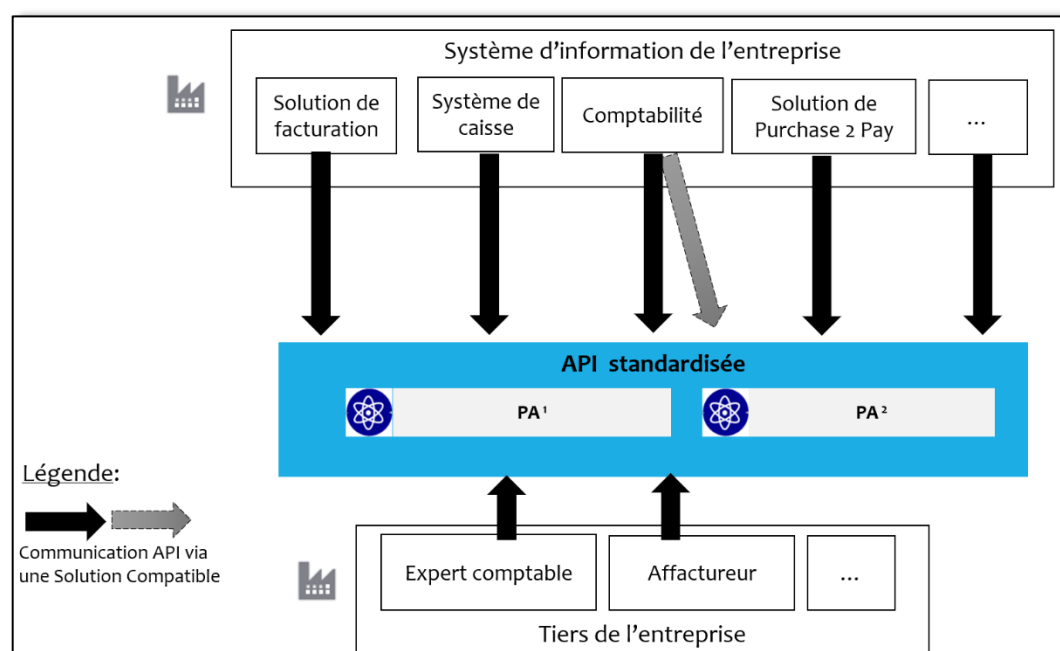
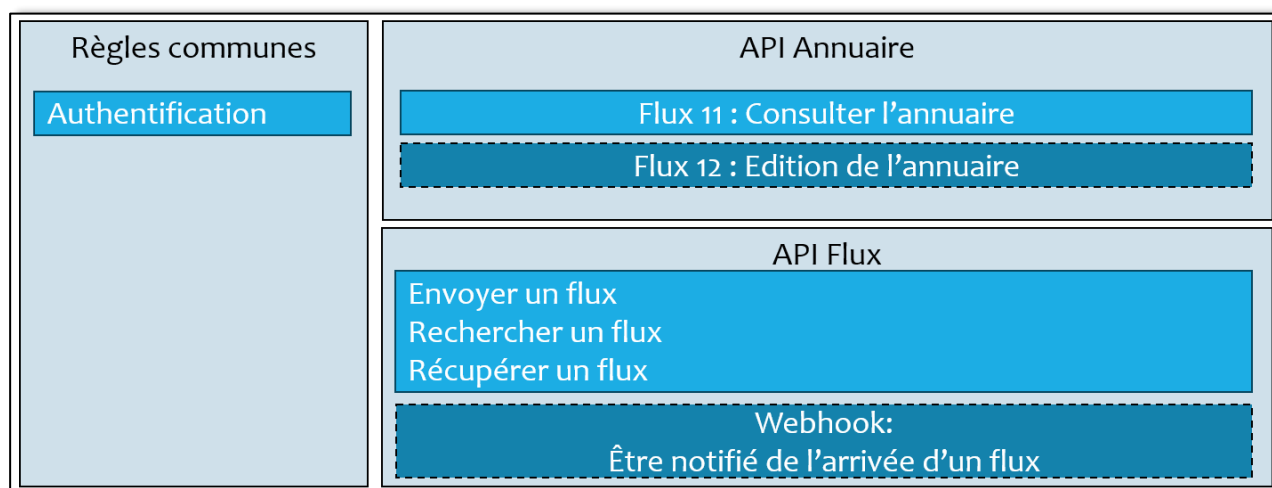


Figure 2 — Faciliter l'accès aux Plateformes Agréées

Le périmètre fonctionnel d'interface standard du présent document concerne :

- Les flux d'émission de factures électroniques vers la PAe (émettrice) et les flux de réception de factures électroniques depuis la PAr (réceptrice) : le Flux 2
- Les flux de statuts de cycle de vie reçus ou à émettre (Flux 6)
- Les flux de e-reporting (Flux 10),
- La consultation et la mise à jour de l'annuaire des assujettis à la TVA (Flux 11 et 12).

XP Z12-013



Légende

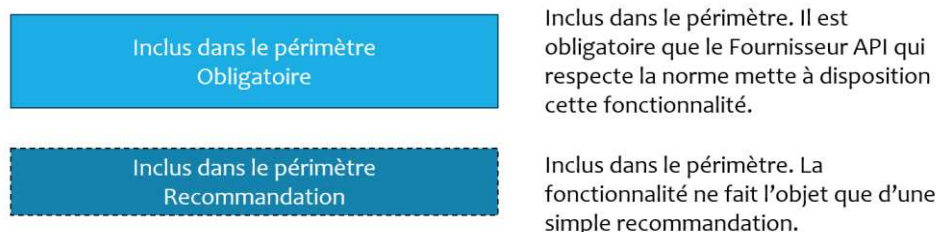


Figure 3 — Périmètre des API

La Figure 3 ci-dessus décrit :

- Ce qui est inclus en termes d'exigence dans le périmètre du présent document :
 - Règles communes : Authentification
 - API Flux : Envoyer, rechercher et récupérer un Flux
 - API Annuaire : Consulter l'annuaire
- Ce qui est inclus en termes de recommandations :
 - API Flux : Webhook permettant d'être notifié de l'arrivée d'un flux
 - API Annuaire : Editer l'annuaire

2 Références normatives

Les documents de référence suivants sont indispensables pour l'application du présent document. Pour les références datées, seule l'édition citée s'applique. Pour les références non datées, la dernière édition du document de référence s'applique (y compris les éventuels amendements).

NF EN 16931-1 - *Facturation électronique - Partie 1 : modèle sémantique de données des éléments essentiels d'une facture électronique*

CEN/TS 16931-2:2017, *Facturation électronique — Partie 2 : Liste de syntaxes conformes à l'EN 16931-1*

CEN/TS 16931-3-1:2017, *Facturation électronique — Partie 3-1 : Méthodologie applicable aux correspondances syntaxiques des éléments essentiels d'une facture électronique*

XP Z12-013 – Standardisation des API SI/SC ⇔ PA

CEN/TS 16931-3-2:2017, *Facturation électronique — Partie 3-2 : Correspondance syntaxique pour la syntaxe ISO/IEC 19845 (UBL 2.1) — Schéma UBL 2.1 Invoice et Credit Note*, publiée en juin 2020.

CEN/TS 16931-3-3:2017, *Facturation électronique — Partie 3-3 : Correspondance syntaxique pour la syntaxe Cross Industry Invoice (facture intersectorielle) — Schéma XML D16B UN/CEFACT*, publiée en juin 2020.

CEN/TR 16931-4 :2017, *Facturation électronique — Partie 4 : Lignes directrices relatives à l'interopérabilité des factures électroniques au niveau de la transmission*

CEN/TR 16931-5 :2017, *Facturation électronique — Partie 5 : Lignes directrices relatives à l'utilisation d'extensions sectorielles ou nationales en complément de l'EN 16931-1, reposant sur une méthodologie à appliquer dans l'environnement réel*

CEN/TR 16931-6, *Facturation électronique — Partie 6 : Résultat des tests de l'EN 16931-1 en ce qui concerne son application pratique pour un utilisateur final — Méthodologie de test*

La documentation Factur-X, libre de droits et disponible auprès de FNFE-MPE et du FeRD, respectivement Forums Nationaux de la Facture Électronique français et allemand, dernière Version 1.07.3 publiée le 7 mai 2025 sur le site www.fnfe-mpe.org.

PR XP Z12-012 : Formats et Profils des messages Factures et Statuts de cycle de vie, constitutifs du socle minimal applicable à la Réforme Facture Electronique en France.

RFC 6750 : The OAuth 2.0 Authorization Framework: Bearer Token Usage

RFC 9421 : Signature d'un Message HTTP

3 Termes et définitions

Pour les besoins du présent document, les termes et définitions suivants s'appliquent.

3.1

API

Interface de programmation applicative, un ensemble normalisé de classes, méthodes, fonctions et constantes permettant à un logiciel d'offrir des services à d'autres logiciels.

3.2

API REST

L'API standardisé du présent document est une API qui respecte le protocole REST. Une API REST (Representational State Transfer) ou RESTful est un type d'API, ou interface de programme d'application, qui aide les applications de services web à communiquer entre elles. Bien qu'elle soit théoriquement compatible avec n'importe quel protocole ou format de données, l'architecture REST utilise le plus souvent le protocole HTTP et transfère les données en utilisant majoritairement le JSON (JavaScript Object Notation).

Une API REST liste un ensemble de « Routes » qui permettent au Client API d'agir.

3.3

API publiée par le Fournisseur API

L'ensemble des API mises à dispositions du Fournisseur API qui sont dans le domaine d'application du présent document.

3.4

Bearer

L'authentification via un 'bearer' est un schéma d'authentification HTTP qui faisait à l'origine partie de la RFC 6750 : « The OAuth 2.0 Authorization Framework-Bearer Token Usage ». Le terme 'bearer' peut s'interpréter comme « donner accès au porteur de ce jeton ». Le jeton du porteur est une chaîne cryptée, généralement

XP Z12-013

générée par le serveur du Fournisseur API en réponse à une requête de connexion. Le Client API doit l'envoyer dans l'en-tête d'autorisation lorsqu'il adresse des requêtes à des ressources protégées.

3.5

BT

'Business Term' au sens de la norme EN 16931

3.6

Client API

L'Organisation qui consomme l'API mis à disposition par le Fournisseur API via un Compte API.

Note 1 au 3.6 : Le Client API sera le plus souvent une Entreprise ou une SC.

3.7

Compte API

Compte avec un identifiant et un mot de passe (ou équivalent) permettant au Client API de se connecter à l'API publiée par le Fournisseur API.

3.8

Facture constituée dans un format du socle

Une facture constituée dans les formats du socle est un seul fichier :

- Un fichier Factur-X
- Ou un fichier XML (CII ou UBL) contenant en option : Le lisible et des pièces jointes encodées en base64.

Un flux facture dans ce document est donc constitué d'une et une seule facture telle que décrite ci-dessus.

3.9

Flux

Les Flux nomment les différents types de messages échangés dans le cadre de la réforme :

- Flux 2 : correspond au message facture échangé entre les entités soumises à la réforme et devant être transmis par l'intermédiaire de PA, et conforme aux dispositions du présent document.
- Flux 3 : correspond au message facture échangé entre les entités soumises à la réforme et devant être transmis par l'intermédiaire de PA, MAIS qui est dans un format tiers convenu entre l'émetteur et le destinataire et contient toutes les informations requises par l'Administration fiscale sous forme structurée et permet une extraction conforme des données pour la constitution du Flux 1 ou du Flux 10.
- Flux 6 : correspond au message de statuts de cycle de vie relatif aux échanges de factures électroniques, implémenté en UN/CFACT CII.
- Flux 8 : correspond au message facture échangé entre une entité soumise à la réforme et une entité internationale conforme aux dispositions du présent document.
- Flux 9 : correspond au message facture échangé entre une entité soumise à la réforme un non assujetti établi en France (principalement un Particulier), conforme aux dispositions du présent document.
- Flux 10 : correspond au message de « e-reporting » que les entités soumises à la Réforme Facture Électronique doivent transmettre au Concentrateur de Données par le biais de leur PA.
- Flux 11 : correspond à la consultation de l'annuaire des assujettis et de leurs adresses de facturation électroniques.
- Flux 12 : correspond à la demande de mis à jour de l'annuaire à sa PA pour ses adresses de facturation électroniques (identifiant d'adressage) et ses Codes Routages/

XP Z12-013 – Standardisation des API SI/SC ⇔ PA

Dans le cadre de l'API standardisée, un flux correspond à un et un seul 'flowId'.

3.10

Formats et profils du socle minimum

Les formats et profils du socle sont les formats de données structurées ou mixtes qui doivent être supportés dans le cadre de la Réforme Facture Électronique, qui implémentent la Norme EN 16931.

D'une part, trois formats constituent ce socle pour le message Facture, et implémentent chacun 2 profils de données :

- Profil EN 16931, qui une CIUS pour la France de l'implémentation de la Norme EN 16931
- Profil EXTENDED-CTC-FR, qui est une EXTENSION pour la France de l'implémentation de la Norme EN 16931

Ces 2 profils sont implémentés dans 2 syntaxes (UBL et UN/CEFACT CII) et dans le format mixte Factur-X, plus précisément :

- Syntaxe XML ISO/IEC 19845 (UBL 2.1) : le format UBL (Universal Business Language) est conforme à la norme OASIS U.B.L. 2.1.
- Syntaxe UN/CEFACT CII. Le format CII (Cross Industry Invoice) est conforme à la norme UN/CEFACT SCRDM CII (Supply Chain Reference Data Model – Cross Industry Invoice). La version de langage retenue dans le cadre de la réforme est UN/CEFACT CII D22B.
- Factur-X. Factur-X est un format de facture électronique hybride (ou mixte), combinant un fichier PDF conforme à la Norme ISO-19005-3 PDF/A-3 constituant la représentation LISIBLE de la facture dans lequel est attaché une représentation de données structurée factur-x.xml dans la syntaxe UN/CEFACT CII. Factur-X dispose de profils additionnels (MINIMUM, BASIC WL, BASIC et EXTENDED).

D'autre part le format de statuts de cycle de vie est implémenté dans la syntaxe UN/CEFACT CDAR (Cross Domain Acknowledgement and Response), et fait aussi partie des formats et profils du socle minimum.

3.11

Fournisseur API

L'organisation qui met à disposition l'API standardisée.

Note 1 au 3.11 : Le fournisseur API sera le plus souvent une PA mais pourra également être une SC.

3.12

FRR

Syntaxe FRR pour FRENCH Reporting

Fichier XML décrivant un flux 10 de E-reporting. Le Format XML est celui décrit par la DGFIP dans les spécifications externes

3.13

SC (Solutions Compatibles)

Les opérateurs offrant des services de dématérialisation des factures mais qui ne sont pas immatriculés par l'administration. Ces opérateurs ne peuvent pas transmettre directement les factures électroniques relevant du périmètre « e-invoicing » à leurs destinataires ni transmettre de données au portail public de facturation, mais peuvent agir au nom et pour le compte de l'entreprise auprès des plateformes de leur choix (y compris ChorusPro).

Note 1 au 3.13 : Les SC sont par exemple des solutions de gestion, des ERP, des plateformes de Purchase to Pay ou d'Order to Cash, des solutions EDI, des solutions bancaires, de refinancement de factures, de carte d'achat ... Ayant à intervenir sur le périmètre facture (soit en création, soit en intégration, validation, paiement, ...).

XP Z12-013

3.14

Plateforme Agréée (ou PA), ex PDP (Plateforme de Dématérialisation Partenaire)

Plateforme Agréée (PA) : Plateforme de facturation électronique au travers de laquelle les factures électroniques entre assujettis à la TVA et relevant du périmètre « e-invoicing » de la Réforme Facture Électronique doivent être échangées, ainsi que les données de « e-reporting » de factures B2B internationales hors import de biens, de transaction et de paiement.

Note 1 au 3.14 : Les Plateformes Agréées peuvent aussi proposer des services qui relèvent d'une SC.

3.15

PAe (Plateforme Agréée Émettrice)

Plateforme Agréée en position d'émission de factures électroniques.

3.16

PAr (Plateforme Agréée Réceptrice)

Plateforme Agréée en position de réception de factures électroniques.

3.17

PPF (Portail Public de Facturation)

Plateforme publique qui administre l'Annuaire des assujettis soumis à la réforme d'une part et le Concentrateur des Données (CdD) d'autre part, qui concentre les données de facturation, de transaction et de paiement ainsi que des informations relatives aux statuts de traitement des factures (cycle de vie) exigées par l'Administration fiscale, et les transmet à l'administration fiscale.

3.18

Webhook

Un webhook, aussi appelé lien de rappel HTTP ou point d'ancrage Web, est en programmation Web une méthode permettant d'accroître ou de modifier le comportement d'une page Web ou d'une application Web avec des fonctions de rappels personnalisées. Ces fonctions peuvent être modifiées et gérées par des utilisateurs et développeurs tiers qui ne sont pas nécessairement affiliés au site Web ou à l'application d'origine. Le terme « webhook » a été inventé par Jeff Lindsay en 2007 à partir du terme de programmation informatique hook[2].

Le format est généralement le JSON. La requête est effectuée comme une requête HTTP POST.

Source : Wikipedia

3.19

XML Schema / XSD

XML Schema, publié comme recommandation par le W3C en mai 2001, est un langage de description de format de document XML permettant de définir la structure et le type de contenu d'un document XML. Cette définition permet notamment de vérifier la validité de ce document.

Une définition se compose d'un ou plusieurs documents XML, usuellement nommée (XML Schema Definition en anglais, ou fichier XSD).

Source : Wikipedia

4 Règles communes à toutes les routes API

4.1 Respect des SWAGGERS

L'API publiée par le Fournisseur API doit être compatible avec le SWAGGER (contrat API en REST) en annexe de la norme.

Cette API doit respecter le versioning indiqué dans l'URL de la route API du Swagger. Dans un objectif de simplification le versioning des routes n'est pas affiché dans le présent document.

XP Z12-013 – Standardisation des API SI/SC ⇔ PA

Dans cette API publiée par le Fournisseur API ce dernier peut :

- Avoir une URL spécifique en amont du versioning
- Ajouter des propriétés aux objets dans les requêtes.
- Ajouter des paramètres aux routes dans les requêtes.
- Ajouter des propriétés aux objets dans les réponses.
- Ajouter des codes erreurs dans les réponses.

4.2 Authentification

L'API publiée par le Fournisseur API doit être compatible avec le protocole d'authentification OAUTH2 et utiliser un Jeton (bearer) au sens de la RFC 6750.

Le Fournisseur API doit permettre au Client API de renseigner un Client Id et un Client Secret et a minima de publier la route « token URL ».

Il est recommandé au fournisseur API de mettre en place une sécurisation du Client Id / Client Secret avec l'un des systèmes ci-dessous :

- Un filtrage IP
- Un mécanisme d'expiration du secret.

Il est recommandé d'avoir un Client Secret qui répond aux recommandations de l'ANSSI sur la sécurisation des mots de passe.

Une fois connecté avec son Compte API, le Client API utilise le jeton (bearer) pour accéder à l'API publiée par le Fournisseur API.

Si un même compte est utilisé pour accéder à plusieurs entreprises ou organisations, le paramètre d'entête 'Organisation-Id' dans le header http permet de le préciser.

4.3 Pagination

L'API publiée par le Fournisseur API doit utiliser un système de pagination basé sur les propriétés 'limit' et 'offset' lors de l'appel et retourner 'total', 'offset' et 'limit' dans la réponse.

Lorsque le jeu de donnée est dynamique (route POST / Search), le paramètre 'offset' n'est pas disponible et la pagination est basée sur système de cursor opaque.

Le Fournisseur API doit accepter une valeur 'limit' inférieure ou égale à 100 et peut refuser des appels avec une valeur supérieure.

4.4 Route GET / healthcheck

L'API publiée par le Fournisseur API doit avoir une route GET / healthcheck permettant au Client API de vérifier si le service API est opérationnel.

4.5 Taille maximale d'un flux

Le Fournisseur API peut refuser des flux supérieurs à 100 mega octets.

XP Z12-013

5 API – Flux

5.1 Respect du Swagger Flow Service

L'API publiée par le Fournisseur API doit respecter le SWAGGER décrit dans l'Annexe A selon les règles décrites en 4.1.

Les routes obligatoires sont listées dans le Tableau 1 ci-dessous :

Tableau 1 — Routes obligatoires de l'API Flow Service

Route	Description	Chapitre
POST/flows	Déposer des flux	5.2
POST/search	Rechercher des flux	5.3
GET/flows/{flowId}	Récupérer un flux	5.4
GET/healthcheck	Vérifier si le service API est opérationnel.	

5.2 Déposer un flux (Route POST / Flows)

5.2.1 Flux acceptés

Route	Description
POST/flows	Déposer des flux

La route POST /flows, de l'API publiée par le Fournisseur API doit permettre de déposer :

- Une facture constituée dans un format du socle (Syntaxe CII, UBL, Factur-X)
- Un cycle de vie sur une facture (Syntaxe CDAR)
- Une transmission de données de E-Reporting (Syntaxe FRR pour FRENCH Reporting)
- Un cycle de vie sur une transmission de données de E-reporting (Syntaxe CDAR)

Le body de la route POST / Flows est un multi-part composé d'un objet payload JSON 'flowInfo' et d'un fichier binaire.

5.2.2 'trackingId' [Champ optionnel]

Le Fournisseur API doit accepter un 'trackingId' et le stocker.

Ce paramètre doit être optionnel pour le Client API.

Il permet à ce dernier de retrouver plus facilement le flux avec son propre identifiant interne.

Le Fournisseur API ne doit pas contrôler l'unicité de cet identifiant qui est un identifiant interne au Client API.

5.2.3 'sha256' [champ optionnel]

Cas d'usage n°1 :

Lorsque la propriété 'sha256' est alimentée par le Client API, il est recommandé au Fournisseur API de contrôler cette valeur avec l'empreinte numérique du fichier déposé calculée en utilisant l'algorithme sha256.

Si la comparaison est différente, le fournisseur API retourne un code 4XX

XP Z12-013 – Standardisation des API SI/SC ⇔ PA

Cas d'usage n°2 :

Si la donnée n'est pas alimentée par le client API lors de dépôt API Flux , le fournisseur API peut calculer ce sha256 et retourne ce résultat dans le message de réponse.

Dans ce cas de figure, c'est le Client Api qui peut vérifier que son flux a été correctement transmis lors de l'appel en analysant le sha256 retourné.

5.2.4 'processingRule' et 'processingRuleSource'

La propriété 'processingRule' permet de qualifier le type de traitement attendu. Pour un flux facture, cela fait référence à la règle de gestion 'BR-FR-20' de la norme XP Z12-012.

Lorsque le paramètre 'processingRule' est contradictoire avec ce qui est mentionné dans la facture (BT-21 avec le code 'BAR') le Fournisseur API doit considérer le flux comme irrecevable.

Le Fournisseur API doit alimenter la propriété 'processingRuleSource' avec la valeur 'input' si la règle de traitement était un paramètre d'entrée fourni par le Client API et avec la valeur 'Computed' si la règle de traitement a été calculée.

Il est recommandé au Fournisseur API d'appliquer les règles de gestion selon le Tableau 2 ci-dessous.

Tableau 2 — Traitement attendu pour la propriété 'processingRule'

Code	Traitement attendu
'processingRule' non alimenté ou vide	Il est recommandé au Fournisseur API : 1] Pour un flux facture, de rechercher le code indiqué en BT-22 dans le flux facture en recherchant un code 'BAR' dans le BT-21. 2] Pour un flux facture, de considérer que la facture relève du B2G si le code note (BT-21) est égal à ADN et si le contenu (BT-22) est égal à B2G. 3] Si le Fournisseur API propose cette fonctionnalité, ce dernier peut qualifier automatiquement le traitement attendu en alimentant la propriété 'processingRule' avec la règle calculée. La propriété 'processingRuleSource' doit alors avoir la valeur 'Computed'. Il est rappelé que si le Fournisseur API propose cette fonctionnalité, les règles doivent être clairement exprimées auprès du Client API. 3] A défaut de considérer le flux comme irrecevable.

Liste de valeur Codifié pour le champ ProcessingRule :

processingRule	Description et Règle de gestion
B2B	Le Fournisseur API doit traiter le flux comme un flux qui relève du e-invoicing sauf incohérence avec la note de facture (BG-1 : BAR/B2B) ci présente
B2G	Le Fournisseur API doit de traiter le flux comme un flux qui relève du e-invoicing des ventes B2G sauf incohérence avec la note de facture (BG-1 : ADN/B2G) si présente
B2GOutOfScope	Le Fournisseur API doit de traiter le flux comme un flux qui relève du e-invoicing des ventes B2G « hors réforme » sauf incohérence avec la note de facture (BG-1 : ADN/B2G et BAR/OUTOFSCOPE) si présente Exemple : Frais de justice.

XP Z12-013

processingRule	Description et Règle de gestion
Bint2G	Le Fournisseur API doit de traiter le flux comme un flux qui venant d'une société non française à transmettre à l'administration française sauf incohérence avec la note de facture (BG-1 : ADN/B2G et BAR/B2BINT) si présente
B2Bint	Le Fournisseur API doit traiter le flux comme un flux qui relève du e-reporting des ventes ou des acquisitions B2B internationales sauf incohérence avec la note de facture (BG-1 : BAR/B2BINT) si présente.
B2C	Le Fournisseur API doit traiter le flux comme un flux qui relève du e-reporting des ventes B2C sauf incohérence avec la note de facture (BG-1 : BAR/B2C) si présente.
OutOfScope	Le Fournisseur API doit traiter le flux comme un flux « hors réforme » sauf incohérence avec la note de facture (BG-1 : BAR/OUTOFSCOPE) si présente.
ArchiveOnly	Le Fournisseur API doit traiter le flux comme un flux qui ne doit pas faire l'objet d'un traitement e-invoicing (pas de flux 1, pas de transmission au destinataire) sauf incohérence avec la note de facture (BG-1 : BAR/ARCHIVEONLY) si présente. Exemple : Il peut s'agir d'un AVOIR interne créé pour annuler une facture REJETÉE ou REFUSÉE ou d'une reprise d'historique.
NotApplicable	Il est recommandé au Client API et au Fournisseur API d'utiliser ce code pour qualifier les flux qui ne nécessitent pas de préciser le traitement à appliquer : Les flux de Cycle de vie (syntaxe CDAR) ou les flux de E-Reporting (syntaxe FRR)

5.2.5 Retour de la route

Le Fournisseur API doit retourner en synchrone un objet payload json, selon la structure suivante avec en particulier une propriété '**flowId**' permettant d'identifier de manière unique le flux dans son système d'information pour un Client API donné,

Construction du message de retour :

Champs	Description du champ
flowId	Identifiant unique de dépôt: peut être construit selon différentes méthodes, un UUID ou une construction propre à chaque système mettant en œuvre l'API Flux
submittedAt	Date et heure de dépôt du fichier dans l'API Flux
trackingId	Identifier externe, qui permet de rechercher les données
name	Nom du fichier déposé
processingRule	Nature / Typologie de l'objet : [B2B, B2Bint,B2C, B2G, Bint2G, OutOfScope, B2GOutOfScope, ArchiveOnly, NotApplicable]
flowSyntax	Format du fichier déposé [CII, UBL, Factur-X, CDAR, FRR]
flowProfile	Profil du fichier Facture déposé [Basic, CIUS, Extended-CTC-FR] ** Caractéristique non renseigné pour un flux Cycle et Vie ou E-reporting
sha256	Il est recommandé au Fournisseur API de calculer l'empreinte numérique du fichier déposé en utilisant l'algorithme sha256 et de la retourner dans ce message de retour.

XP Z12-013 – Standardisation des API SI/SC ⇔ PA

5.2.6 Traitement attendu : la qualification du flux avec un 'FlowType'

À la suite du dépôt d'un Flux par la Route POST / Flows, le Fournisseur API doit qualifier le flux en lui associant un FlowType pour permettre une segmentation des Flux selon 3 familles :

- **Invoice** : CustomerInvoice, SupplierInvoice
- **LifeCycle** : CustomerInvoiceLC, SupplierInvoiceLC
- **TransactionReport et PaymentReport** : AggregatedCustomerTransactionReport, UnitaryCustomerTransactionReport, AggregatedCustomerPaymentReport, UnitaryCustomerPaymentReport, UnitarySupplierTransactionReport, MultiFlowReport

Ces flux sont détaillés dans le Tableau 3.A.

D'autre part, Il est recommandé au Fournisseur API de créer des flux reflétant les flux envoyés au Concentrateur de Données du PPF :

- Lors de la transmission d'un Flux 1 au Concentrateur de Données du PPF, il est recommandé au Fournisseur API de créer un flux avec un flowType « StateCustomerInvoice » ou « StateSupplierInvoice »
- Lors de la transmission d'un Flux CDAR au Concentrateur de Données du PPF, il est recommandé au Fournisseur API de créer un flux avec un flowType « StateCustomerInvoiceLC » et « StateSupplierInvoiceLC ».
- Lors de la transmission d'un Flux de E-Reporting au Concentrateur de Données du PPF, il est recommandé au Fournisseur API de créer un flux avec un flowType StateTransactionReport ou StatePaymentReport
- Lors de la réception d'un Flux de cycle de vie de E-Reporting envoyé par le Concentrateur de Données du PPF, il est recommandé au Fournisseur API de créer un flux avec un flowType StateTransactionReportLC ou StatePaymentReportLC

Ces flux sont détaillés dans le Tableau 3.B.

XP Z12-013

Tableau 3.A — Qualification des flux avec un FlowType pour les flux entre le Client API et le Fournisseur API

FlowType	Règle de gestion
CustomerInvoice	Un flux « facture de vente ». Doit être utilisé pour qualifier une facture sortante non auto-facturée ou pour une facture entrante en auto-facturation
SupplierInvoice	Un flux « facture d'achat ». Doit être utilisé pour qualifier une facture entrante non auto-facturée ou pour une facture sortante en auto-facturation
CustomerInvoiceLC	Doit être utilisé pour qualifier un flux cycle de vie (CDAR) d'une facture de type 'CustomerInvoice'
SupplierInvoiceLC	Doit être utilisé pour qualifier un flux cycle de vie (CDAR) d'une facture de type 'SupplierInvoice'
AggregatedCustomerTransactionReport	Doit être utilisé pour qualifier un flux de E-Reporting de transaction contenant des ventes B2C agrégées (FRR 10.3)
UnitaryCustomerTransactionReport	Doit être utilisé pour qualifier un flux de E-Reporting de transaction contenant des ventes B2B à l'international (FRR 10.1) Si le Fournisseur API accepte un flux de transaction B2C déclaré unitairement, le Fournisseur API doit utiliser ce flowType pour qualifier le flux
AggregatedCustomerPaymentReport	Doit être utilisé pour un flux de E-Reporting d'encaissements contenant des encaissements liés à des ventes B2C (FRR 10.4)
UnitaryCustomerPaymentReport	Doit être utilisé pour un flux de E-Reporting d'encaissements contenant des encaissements liés à des ventes B2B internationales unitaires (FRR 10.2) Si le Fournisseur API accepte un flux de paiement B2C déclaré unitairement, le Fournisseur API doit utiliser ce flowType pour qualifier le flux
UnitarySupplierTransactionReport	Doit être utilisé pour qualifier un flux de E-Reporting de transaction contenant des achats B2B à l'international (FRR flux 10.1)
MultiFlowReport	Doit être utilisé pour qualifier un flux de E-Reporting lorsque le flux contient au moins 2 types de flux différents (FRR flux 10) Il est recommandé au Consommateur API de vérifier la cohérence des périodes lors de l'utilisation de ce type de flux.

XP Z12-013 – Standardisation des API SI/SC ⇔ PA

Tableau 3.B — Qualification des flux avec un FlowType pour les flux entre le Fournisseur API et le PPF

FlowType	Règle de gestion
StateCustomerInvoice	Peut être utilisé pour qualifier un Flux 1 envoyé par la PAe au Concentrateur de Données du PPF d'une facture de type 'CustomerInvoice'
StateCustomerInvoiceLC	Peut être utilisé pour qualifier un flux cycle de vie (CDAR) envoyé au Concentrateur de Données du PPF d'une facture de type 'CustomerInvoice'
StateSupplierInvoice	Peut être utilisé pour qualifier un Flux 1 envoyé par la PAe au Concentrateur de Données du PPF d'une facture de type 'SupplierInvoice' (Facture sortante auto-facturée)
StateSupplierInvoiceLC	Peut être utilisé pour qualifier un flux cycle de vie (CDAR) envoyé au Concentrateur de Données du PPF d'une facture de type 'SupplierInvoice' (Facture sortante auto-facturée)
StateTransactionReport	Peut être utilisé pour qualifier un flux de E-Reporting de transaction (FRR flux 10.1 et/ou 10.3) envoyé au Concentrateur de Données du PPF.
StateTransactionReportLC	Peut être utilisé pour qualifier un flux cycle de vie (CDAR) envoyé au Concentrateur de Données du PPF d'un E-Reporting de transaction (FRR flux 10.1 et/ou 10.3)
StatePaymentReport	Peut être utilisé pour qualifier un flux de E-Reporting de paiement (FRR flux 10.2 et/ou 10.4) envoyé au Concentrateur de Données du PPF
StatePaymentReportLC	Peut être utilisé pour qualifier un flux cycle de vie (CDAR) envoyé au Concentrateur de Données du PPF d'un E-Reporting de paiement (FRR flux 10.2 et/ou 10.4)

5.3 Rechercher un flux (Route POST / Search)

Ce service permet de récupérer un ensemble de flux correspondant aux critères de recherche fournis :

XP Z12-013

Critères de recherche

Critères	Type	Description
updatedAfter	Date-Time	Date début de période de recherche : 15/01/2026
updatedBefore	Date-Time	Date fin de période de recherche : 30/01/2026
processingRule	Liste de valeurs codifiées	Nature / Typologie de l'objet Facture [B2B, B2BInt, B2C, OutOfScope, etc... ArchiveOnly, NotApplicable]
flowProfile	Liste de valeurs/ codifiées	Profil de l'objet Facture [BT-24] [Basic, CIUS, Extended-CTC-FR]
flowType	Liste de valeurs/ codifiées	Typologie de l'objet [CustomerInvoice, SupplierInvoice, StateInvoice, CustomerInvoiceLC, SupplierInvoiceLC, StateCustomerInvoiceLC, StateSupplierInvoiceLC, AggregatedCustomerTransactionReport]
flowDirection	Liste de valeurs codifiées	Sens de flux In: Incoming flow, from the PA to the SC Out: Outgoing flow, from the SC to the PA
trackingId	Texte	Id externe fixé par le système source, permettant à l'émetteur de cibler la facture
ackStatus	Liste de valeurs codifiées	Statut du traitement consommateur de ce flux - Pending : le flux n'est pas encore intégré - Error : le flux a été traité, une erreur a été identifiée - OK : l'ensemble des contrôles a été effectué avec un résultat Correct

Il faut qu'au moins un critère soit spécifié.

Conditions de recherche :

La recherche s'exécute en appliquant l'opérateur AND afin de combiner les critères.

Exemple : Une recherche avec 4 critères cumulatifs :

- Recherche de facture
 - traitée entre le 1^{er} et le 15 janvier
- ET
- Dans le sens sortant : OUT
- ET
- Avec le statut « Error »

XP Z12-013 – Standardisation des API SI/SC ⇔ PA

```

• "where": {
•   "updatedAfter": "2026-01-01T22:03:07.255Z",
•   "updatedBefore": "2026-01-15T22:03:07.255Z",
•   "flowDirection": [
•     "Out"
•   ],
•   "ackStatus": "Error"
• }

```

Condition de cumul pour les listes de valeurs :

La recherche s'exécute en appliquant l'opérateur OR afin de combiner les valeurs.

Exemple : Une recherche de l'ensemble des factures ayant pour typologie : B2B ou B2G ou B2C.

```

• "processingRule": [
•   "B2B", "B2G", "B2C"
• ],

```

Pagination :

La pagination fonctionne avec la propriété cursor.

La comparaison avec la date actuelle est stricte : updatedAt > updatedAfter

5.3.1 Gestion du différentiel

Le Fournisseur API doit mettre à disposition la route POST / Search pour permettre la récupération des flux en différentiel en permettant à ce dernier de ne récupérer que les dernières factures / statuts nouveaux depuis sa dernière requête.

5.3.2 Garantir l'exhaustivité des flux

Le Fournisseur API doit trier les résultats selon le critère 'updatedAt' en ordre strictement croissant.

Le Fournisseur doit garantir au Client API suivant les recommandations ci-dessous de pouvoir récupérer l'exhaustivité des flux.

Il est recommandé au Client API de renseigner la valeur 'updatedAfter' avec la date du dernier élément récupéré.

Note 1 au 5.3.2 La récupération de l'exhaustivité des flux peut être garantie par le Fournisseur API lorsque 'updatedAt' est une séquence strictement monotone atomiquement. Un moyen permettant de le garantir est de laisser le temps aux transactions de la base de données et aux traitements parallèles de se terminer. Par exemple, si on considère qu'un délai de 15 minutes est suffisant, il s'agit de retourner un jeu de données avec 'updatedAt' compris entre la valeur 'updatedAfter' passée en paramètre et la date système du serveur de la base de données moins 15 minutes.

5.4 Récupérer un flux (Route GET / flows / {flowId})

5.4.1 Paramètre 'docType'

Lorsque le flux est une facture, le paramètre 'docType' de la Route GET / flows permet au Client API d'indiquer le composant de facture demandé, avec les valeurs suivantes :

- « Metadata » : l'objet Json décrivant le flux.
- « Original » : La facture telle que reçue.

XP Z12-013

- « ReadableView » : La représentation lisible de la facture.
- « Converted ». Une version convertie (préalablement ou à la volée) de façon à pouvoir choisir un format préféré.

Note 1 au 5.4.1 : le Fournisseur API peut permettre au Client API de configurer la règle suivante : l'ensemble des factures reçues en UBL sont converties en Factur-x, pour que ces conversions puissent être préparées en amont.

5.4.2 Récupération du lisible d'une facture

Le lisible mis à disposition par la PA doit l'être par la route GET / flows.

Lorsque le lisible d'une facture n'est pas fourni par l'émetteur et que cette facture est dans un format du socle, le Fournisseur API doit pouvoir le générer et le retourner par la route GET / flows / {flowId}.

5.5 Gestion des flux irrecevables et des rejets

Le Fournisseur API doit distinguer

- Les contrôles techniques et applicatifs qui, en cas d'erreur, peuvent conduire à une irrecevabilité d'un Flux
- Les contrôles fonctionnels qui, en cas d'erreur, peuvent conduire à un Rejet du Flux.

Le Fournisseur API doit retourner l'erreur au Client API ;

- Par un flux CDAR lorsqu'il s'agit d'un Rejet et qu'il existe un CDAR pour le flux
- Par la propriété FlowAckStatus lorsque le flux est irrecevable ou qu'il n'existe pas de CDAR pour le flux (Exemple Flux CDAR envoyé où on ne retourne pas un autre CDAR pour informer de l'irrecevabilité)

5.5.1 Irrecevabilité d'un flux envoyé par le Client API au Fournisseur API

La propriété « FlowAckStatus » doit être renseignée par le Fournisseur API par un statut 'Error' lorsque le flux est irrecevable et par un statut « Ok » lorsque le flux est recevable.

Lorsque le flux est irrecevable, le Fournisseur API doit retourner un code erreur dans la propriété 'reasonCode' de l'objet 'AcknowledgementDetail'.

Le Fournisseur API doit utiliser les codes du tableau 4 ci-dessous mais est libre de rajouter d'autres codes erreurs.

Tableau 4 — Codes erreurs pour un flux irrecevable

Code erreur (reasonCode)	Code Z12-012	Description
EmptyAttachement	IRR_VID_PJ	L'une ou plusieurs pièces jointes sont vides
AttachmentTypeError	IRR_EXT_DOC	Le type et/ou l'extension d'une ou plusieurs pièce jointes n'est pas conforme
EmptyFlow	IRR_VIDE_F	Le flux est vide
OtherTechnicalError	AUTRE	Autre problème technique
InvalidSchema	IRR_SYNTAX	Schéma XML invalide

XP Z12-013 – Standardisation des API SI/SC ⇔ PA

FileSizeExceeded	IRR_TAILLE_F	Taille limite de fichier atteinte
FlowTypeError	IRR_TYPE_F	Le type et/ou l'extension du flux ne sont pas conformes
AlreadyExistingFlow	N/A	Le flux a déjà été envoyé et réceptionné (Exemple : empreinte numérique identique) A ne pas confondre avec le contrôle sur le doublon du n° de facture.
VirusFound	IRR_ANTIVIRUS	Contrôle anti-virus Le flux ne respecte pas les conditions de sécurité
ChecksumMismatch	N/A	L'empreinte numérique n'est pas cohérente.

5.5.2 Rejet d'un flux facture par la PAe ou la Par

Lorsqu'une facture est rejetée par la PA émettrice (PAe) ou par la PA de réception (Par) le Fournisseur API doit créer et mettre à disposition au Client API un flux CDAR avec un statut 213.

Note 1 au 5.5.2 : Pour rappel, la norme XP Z12-014 indique que statut 213 doit également être envoyé au concentrateur public.

Note 2 au 5.5.2 : Un motif de rejet peut par exemple un problème de routage ou une règle métier en erreur.

5.5.3 Rejet d'un flux Cycle de Vie sur une Facture

La propriété « FlowAckStatus » doit être renseignée par le Fournisseur API par un statut 'Error' lorsque le flux est rejeté.

Le Fournisseur API doit utiliser les codes du tableau 5 ci-dessous mais est libre de rajouter d'autres codes erreurs.

Tableau 5 — Codes erreurs pour un flux de Cycle de Vie rejeté

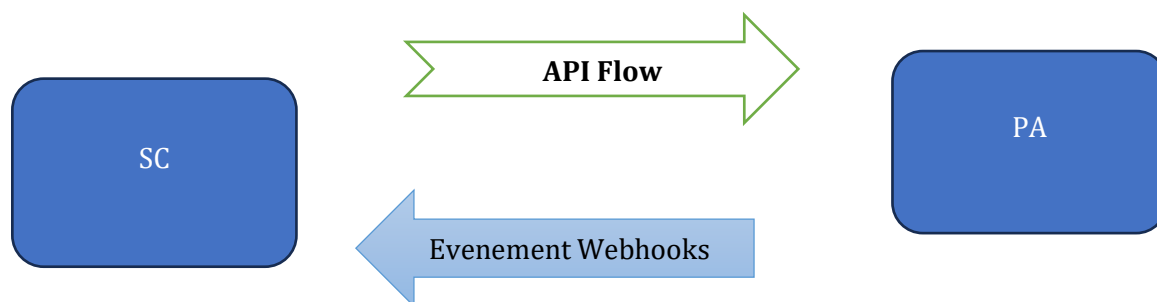
Code erreur (reasonCode)	Code des spécifications externes	Description
InvoiceLCInvalidStatus	REJ_INC	L'un ou plusieurs statuts sont incohérents
InvoiceLCStatusError	REJ_INEX	L'un ou plusieurs statuts sont incorrectes ou non autorisées
InvoiceLCRuleError	REJ_RG	L'une ou plusieurs règles de gestion ne sont pas respectées
InvoiceLCAccesDenied	REJ_HAB	L'une des requêtes n'est pas autorisées et/ou requiert une habilitation
InvoiceLCAmountError	REJ_ENCAISSEMENT	L'un ou plusieurs montants encaissés ne sont pas conformes à la répartition par taux de TVA déclarée

5.6 Informer le Client API de l'arrivée ou de la modification d'un flux par un Webhook

5.6.1 Définition

Un webhook est une communication orientée événements qui transmet de façon automatique des données entre les applications via HTTP.

Déclenchés par des événements précis, les webhooks automatisent la communication retour vers le SI à l'origine de la demande de traitement.

XP Z12-013

Le webhook va permettre au Client API de recevoir les événements traités ou reçus, par le Fournisseur API, afin de leur remonter les informations au fur à mesure de la journée.

Il n'est plus nécessaire d'interroger en mode Pooling ; les événements remontent naturellement.

Ce type de mécanisme se construit en 3 phases :

- Un abonnement via un appel API
- Le déclenchement d'un événement de type webhook
- La consommation de cet événement dans le SI Client d'origine par une API construite selon le swagger défini ci-dessous.

Les avantages du webhook :

- **Il remplace l'interrogation continue.** L'application client économise ainsi des ressources.
- **Il est rapide à configurer.** Le webhook peut être facilement configuré via une API d'abonnement. C'est là que le client ajoute l'URL de son webhook et définit les paramètres de base, comme l'événement qui l'intéresse.
- **Il automatise le transfert de données.** Les données utiles sont envoyées dès que l'événement recherché se produit. C'est cet événement qui déclenche l'échange, qui s'effectue donc aussi rapidement que tout transfert en temps réel.

Complémentarité du Webhook et de la méthode /Flows/Search**Cas d'usage :**

Le endpoint /Flows/Search peut servir de solution de backup pour récupérer un historique de flux en cas d'indisponibilité de l'API à l'écoute des événements Webhook.

Principe :

Le client API viendra s'abonner en utilisant les méthodes présentées au chapitre n°5.6.2 Abonnement Webhook...(Tableau 7)

Le Client API doit construire un endpoint API d'écoute des messages webhook, conforme au contrat swagger en cours de construction au chapitre 5.6.3 - Consommation de l'événement Webhook (Tableau 9)

XP Z12-013 – Standardisation des API SI/SC ⇔ PA

5.6.2 Abonnement à un Webhook

Il est recommandé au Fournisseur API de mettre à disposition les routes décrites dans le Tableau 6 ci-dessous

Tableau 6— Routes permettant d’administrer les abonnements Webhook

Méthode	Endpoint	Description
POST	/v1/webhooks	Créer un Webhook
PATCH	/v1/webhooks/{webhookUid}	Permet de mettre à jour un abonnement, afin de changer la configuration
DELETE	/v1/webhooks/{webhookUid}	Permet de supprimer l’abonnement à ce webhook
GET	/v1/webhooks	Liste l’ensemble des abonnements
GET	/v1/webhooks/{webhookUid}	Lire la configuration de l’abonnement concerné

Créer / modifier un abonnement au webhook

Méthode	Endpoint	Description
POST	/v1/webhooks	Créer un Webhook

Un appel en précisant plusieurs paramètres :

- - url de rappel
- - des métadonnées pour configurer cet appel (bloc callBack)
- - des métadonnées pour préciser les événements métiers (bloc metadata)

Voici la liste des champs à configurer dans le message JSON lors de l’appel API, via la méthode POST

Tableau 7— Paramètres d’appel de création / modification d’un Webhook

Param	Type	Valeur
url (*)	Text(uri)	url d’appel du système source
authType	Liste de valeurs codifiées	Méthode d’authentification [BASIC, OAUTH2]
BasicAuthentification		Configuration du mode Basic
- UserId	Texte	- UserId du compte technique concerné
- userPassword	Texte	- userPassword
oAuth2Authentification		Méthode oAuth2
- tokenUrl	url	url à appeler pour obtenir un jeton d’accès
- clientId	Texte	identifiant du compte technique API, lors d’un échange entre 2 serveurs
- clientCredentials	Texte	chaine complexe qui sert de mot de passe lors de l’échange
Signature		Permet de signer les données selon l’un des algorithmes proposés [RS256, HS256, ECDSA, EDDSA_25519]
- algo	Liste	
- key	Texte	Base 64 encoded string

XP Z12-013

Headers []		
- headerName	Texte	Couple de clé / valeur à répéter afin d'apporter des caractéristiques propres à votre SI
- headerValue	Texte	

Configuration des critères métiers qui permettent de cibler

- certains flux In/ Out
- certaines natures : B2B, B2C, etc
- certains objets : Invoice, LifeCycle, TransactionReport ...
- Certains évènements : Pending, Error, OK

Tableau 8 - Paramètres métiers configurés sur l'abonnement Webhook

Param	Type	Valeur
flowTypes	Liste de valeurs codifiées	Type de flux reçus et traités [CustomerInvoice, SupplierInvoice, StateInvoice, CustomerInvoiceLC, SupplierInvoiceLC, StateCustomerInvoiceLC, StateSupplierInvoiceLC, AggregatedCustomerTransactionReport]
flowDirection	Liste de valeurs codifiées	Direction du flux [in, out]
ackStatus	Liste de valeurs codifiées	Statuts du document [Pending, Error, OK] - Pending : le flux n'est pas encore intégré - Error : l'un des tests précédents a échoué - OK : les vérifications suivantes ont réussi
Request-Id		Permet de taguer l'abonnement avec une clé
Organization-Id		Permet de taguer l'organisation avec une clé

Exemple de réponse lors de l'abonnement au webhook :

```
{
  "webhookId": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
  "signingKey": "string",
  "createdAt": "2026-06-22T10:05:24.190Z"
}
```

5.6.3 Consommation de l'évènement Webhook

Présentation de la structure du message

l'évènement Webhook remonte un message payload JSON contenant :

- Un objet 'WebhookCallbackContent'
- Lorsque le WebHook est utilisé dans un usage avec plusieurs organisations des propriétés permettant au Client API d'identifier l'organisation, afin de permettre à ce dernier d'utiliser la bonne authentification pour récupérer les flux.

Note 1 : les propriétés permettant au Client API d'identifier l'organisation peuvent être, par exemple :

XP Z12-013 – Standardisation des API SI/SC ⇔ PA

- Un identifiant société interne au Client API paramétré lors de la configuration du Webhook,
- Un identifiant interne au Fournisseur API,
- Un identifiant métier comme un SIREN, une adresse électronique etc.

Tableau 9 – Message évènement remonté par le webhook

Param	Type	Valeur
flowId	string	Unique identifier supporting UUID but not only, for flexibility purpose
name	Texte	Name of the file
flowDirection		• In: Incoming flow, from the PA to the SC • Out: Outgoing flow, from the SC to the PA
processingRule	Enum	[B2B, B2BInt, B2C, OutOfScope, ArchiveOnly, NotApplicable]
flowSyntax	Enum	[CII, UBL, Factur-X, CDAR, FRR]
flowProfile	Enum	[Basic, CIUS, Extended-CTC-FR]
flowType	Enum	[CustomerInvoice, SupplierInvoice, StateInvoice, CustomerInvoiceLC, SupplierInvoiceLC, StateCustomerInvoiceLC, StateSupplierInvoiceLC, AggregatedCustomerTransactionReport]
processingRuleSource	Enum	[Input, Computed]
trackingId	texte	The tracking id is an external identifier and is used to track the flow by the sender
submittedAt	Date	The flow submission date and time (the date and time when the flow was created on the system) This property should be used by the API consumer as a time reference to avoid clock synchronization issues
updatedAt	Date	The last update date and time of the flow. When the flow is submitted updatedAt is equal to submittedAt. When the flow acknowledgment status is changed updatedAt date and time is updated.
Acknowledgement - Status - Détails • Level • Item • reasonCode • reasonMessage	Enum Texte Enum Texte	[Error, Warning] Item on which the error refers A predefined set of reason code values + ability to extend this set

Tableau 10 – Liste des code retours :

reasonCode	Description
EmptyAttachement	One or more attachments are empty. (ie IRR_VID_PJ)
AttachmentTypeError	The type and/or extension of one or more attachments is not compliant. (ie IRR_EXT_DOC)
EmptyFlow	The binary flow is empty. (ie IRR_VID_F)
OtherTechnicalError	Other technical error. (ie AUTRE)
InvalidSchema	Invalid XML schema. (ie IRR_SYNTAX)
FileSizeExceeded	: File size limit reached. (ie IRR_TAILLE_F)
FlowTypeError	The flow type and/or extension are not compliant. (ie IRR_TYPE_F)

XP Z12-013

AlreadyExistingFlow	The flow has already been sent and received. (ie N/A)
VirusFound	Presence of a virus. (ie IRR_ANTIVIRUS)
ChecksumMismatch	Checksum provided is different from computed one
InvoiceLCInvalidStatus	One or more statuses are inconsistent
InvoiceLCStatusError	One or more statuses are incorrect or not allowed
InvoiceLCRuleError	One or more rules are not matched
InvoiceLCAccessDenied	One of the request is not authorized and requests permissions
InvoiceLCAmountError	One or more amounts are not consistent in regards to the VAT

6 API – Annuaire

6.1 Respect du Swagger Directory Service

L'API publiée par le Fournisseur API doit respecter le SWAGGER dans l'Annexe B selon les règles décrites en 4.1.

6.2 Consultation de l'Annuaire

6.2.1 Routes

Le Fournisseur API doit publier les routes API obligatoires listées dans le tableau 11 ci-dessous.

Pour les routes POST/siren/search, POST/siret/search, POST/routing-code/search et POST/directory-line/search le Fournisseur API doit à minima permettre au Client API de récupérer tous les objets lorsqu'un SIREN ou à défaut un SIRET complet est passé en paramètre. Pour la route POST/directory-line/search cela revient à récupérer toutes les lignes d'annuaire respectivement lorsqu'un SIREN complet est passé en paramètre. Il est recommandé au Fournisseur API de proposer les autres fonctionnalités avancées de recherche (recherche sur un libellé, un SIREN ou un SIRET incomplet etc...). Le Fournisseur API doit retourner un code erreur explicite si la requête en consultation n'est pas prise en charge.

Le fournisseur de l'API n'a pas l'obligation de calculer le nombre total de résultats correspondant à une recherche, notamment pour des raisons de performance. Lorsque ce total n'est pas calculé ou n'est pas disponible, le champ totalNumberOfResults doit être renseigné avec la valeur -1.

Note 1 au 6.2 : Il est rappelé au Fournisseur API, qu'il ne doit retourner que les données publiables.

Tableau 11 — Routes de consultation de l'Annuaire

Ressource	Route	Description
SIREN	POST /siren/search	Rechercher un SIREN avec plusieurs critères en JSON
	GET /siren/code-insee:{siren}	Rechercher un SIREN avec le SIREN (test d'existence ou récupération des informations du SIREN)
SIRET	POST /siret/search	Rechercher un SIRET avec plusieurs critères en JSON

XP Z12-013 – Standardisation des API SI/SC ⇔ PA

Ressource	Route	Description
	GET /siret/code-insee:{siret}	Rechercher un SIRET avec le SIRET (test d'existence ou récupération des informations du SIRET)
Code routage	POST /routing-code/search	Rechercher un code routage avec plusieurs critères en JSON
	GET /routing-code/siret:{siret}/code:{routing-identifiant}	Rechercher un code routage avec le SIRET de rattachement et le code routage (test d'existence ou récupération des informations du code routage)
Ligne d'adressage	POST /directory-line/search	Rechercher une ligne d'annuaire avec plusieurs critères en JSON
	GET /directory-line/code:{addressing-identifiant}	Rechercher une ligne d'adressage avec l'identifiant d'adressage (test d'existence ou récupération des informations de la ligne).

6.2.2 Propriété DirectoryLineStatus

Les routes 'GET/directory-line' et 'POST/directory-line/search' doivent retourner la propriété '**directoryLineStatus**' permettant d'indiquer au Client API le statut d'une ligne.

La route 'GET/directory-line' retourne la ligne d'annuaire pour laquelle la (Nature = D) & (DateDebut <= J < DateFin (si renseignée)). Par construction de l'annuaire cette ligne est censée être unique et ne peut avoir comme valeur « disabled » ou « enabled ».

Codification	Valeur	Description
disabled	Désactivé	Une ligne d'annuaire désactivée se voit positionnée à la plateforme 9998
enabled	Activé	Une plateforme a été définie, la ligne d'annuaire est active.
upcoming	En cours	Une plateforme ou une nouvelle plateforme a été fixée et sa date de début est strictement postérieure à la date du jour. A partir de la Datedébut, la ligne d'annuaire passera au statut Enabled/Activé.

Le fournisseur de l'API doit appliquer des règles afin de déterminer les valeurs « disabled », « enabled » et « upcoming » du champ **directoryLineStatus**, telles qu'exprimées ci-dessous en pseudo-code :

- Si (Nature = D) & (DateDebut <= J < DateFin (si renseignée)) & (plateforme = 9998) => disabled
- Si (Nature = D) & (DateDebut <= J < DateFin (si renseignée)) & (plateforme != 9998) => enabled
- Si (Nature = D) & (J < DateDebut & plateforme != 9998) => upcoming

Notes au 6.2.2 :

- J : Date du jour

XP Z12-013

- DateDebut : Date de début d'effet de la ligne
- DateFin : Valeur minimale entre la date de fin d'effet et la date de fin effective de la ligne
- Nature D : Nature « Définition » dans l'annuaire public
- Nature M : Nature « Masqué » dans l'annuaire public
- Plateforme 9998 : Lorsqu'une ligne d'adressage n'a pas de plateforme agréée associée

6.2.3 Propriété isSalesProspectingForbidden

Le critère « diffusible » est matérialisé dans l'annuaire PPF.

Afin de partager cet élément lors de la consultation de l'annuaire, un critère Booléen Yes/No « isSalesProspectingForbidden », permet de notifier que les données concernées sur le bloc d'information SIREN et SIRET sont interdites à la diffusion pour des besoins de prospection et de vente.

Le fournisseur API doit appliquer les règles suivantes pour déterminer la valeur booléenne de « isSalesProspectingForbidden ».

Diffusible	Description	isSalesProspectingForbidden	Conséquences
O	Diffusible	false	Données non présentes
P	Partiellement Diffusible	false	Données confidentielles non présentes : • le numéro + la voie de l'adresse postale
M	Diffusible avec un Refus de prospection commerciale	true	Données présentes, mais en précisant le cadre d'utilisation à respecter

Ces règles sont exprimées ci-dessus à partir de la propriété Diffusible DT-3-6 (SIREN), DT-4-14 (SIRET) de l'annuaire.

6.3 Mise à jour de l'Annuaire

La mise à jour à l'annuaire pourra faire l'objet d'un travail de normalisation dans une version ultérieure.

7 Gestion de la délégation

7.1 Principes généraux

La gestion de la délégation vise à faciliter l'accès aux données stockées chez le Fournisseur API par des tiers habilités par les entreprises (tels que les experts-comptables, les affactureurs, les tiers payeurs, les agents d'acheteurs, les sociétés holding, ...).

La délégation implique donc 3 parties :

- L'entreprise qui va donner accès à ces données et qui passera par un Client API
- Le Fournisseur API
- Le Tiers qui peut être une entreprise ou une organisation.

XP Z12-013 – Standardisation des API SI/SC ⇔ PA

7.2 Exigences de la gestion de la délégation

Il est recommandé que le Fournisseur d'API et le Tiers mettent en œuvre un dispositif de gestion de la délégation conforme aux exigences suivantes :

- Assurer la sécurisation de la transmission du client id et du client secret, afin de garantir que ces éléments ne soient accessibles qu'aux personnes dûment habilitées.
- Veiller à ce que la délégation ne soit pas rattachée à une personne physique susceptible de quitter l'organisation ou l'entreprise.
- Veiller à s'assurer de l'identité du Tiers.

7.3 Périmètre de la délégation

Il est recommandé que le Fournisseur d'API permette à l'entreprise de définir le périmètre de la délégation selon les droits ci-dessous :

- Pour l'API Flux
 - Lecture (search et Get) et/ou Ecriture (Post)
 - Flowtype
 - Périmètre plus détaillé allant pour gérer par exemple des cas liés à l'affacturage, le tiers payeur etc...
- Pour l'API Directory
 - Consultation : oui/non

7.4 Processus de délégation au tiers au travers d'un client id / client secret pour chaque entreprise

7.4.1 Description du processus

Ce processus se déroule en 3 grandes étapes :

1. Création d'un client id/client secret par l'entreprise pour le Tiers dans une interface mise à disposition par le Fournisseur API.
2. La base URL, la « token URL », le client id et le client secret sont ensuite renseignés dans le système d'information du Tiers.
3. Le Tiers se connecte au Fournisseur API grâce à la base URL, la token URL, le client id et le client secret.

7.4.2 Conformité du processus

La conformité du processus va dépendre des choix d'implémentation du processus mis en place.

Le Tableau 12 ci-dessous liste 3 exemples de processus :

- L'exemple A] est un processus déconseillé car il ne répond pas aux exigences.
- Les exemples B], C] permettent de répondre aux exigences.

Sur l'objectif de veiller à s'assurer de l'identité du Tiers, le Fournisseur API est dépendant du processus mis en place le Tiers.

XP Z12-013

Tableau 12 — Exemples de processus

Description du processus	Objectif de sécurisation du client id/ client secret	Délégation à une entreprise
A] Processus non conforme A1. Création d'un client id/client secret par l'entreprise dans la PA. A2. Transmission par un moyen non sécurisé (email) à un collaborateur du tiers. A3. Le collaborateur du tiers renseigne l'URL , la « token URL », le client id et le client secret dans l'outil métier.	Non conforme : La transmission par email n'est pas sécurisée. <u>Par exemple :</u> Ce mail peut être transféré à une boîte email supplémentaire, conservé dans une messagerie personnelle, etc.	
B] Processus conforme sous réserves avec un Tiers ne disposant pas d'un portail accessible à l'entreprise. B1. Création d'un client id/client secret par l'entreprise dans la PA. B2. Transmission par un canal sécurisé à un collaborateur du Tiers. B3. Le collaborateur du tiers renseigne l'URL, le client id et le client secret dans l'outil métier	<u>Conforme sous réserve</u> d'une transmission sécurisée.	<u>Conforme sous condition</u> Le collaborateur ayant réceptionné le client id /client_secret doit associer ces éléments d'identification au Tiers concerné, dans l'outil métier Et ne doit en aucun cas l'associer à son compte personnel.
C] Processus conforme avec un Tiers disposant d'un portail accessible à l'entreprise. C1. Création d'un Compte Tiers sur la PA C2. Accréditation du Tiers par l'entreprise C3. Identification d'une clé d'organisation pour l'entreprise. => Détail Chapitre ci dessous	Conforme : La transmission est sécurisée	Conforme : En dehors des administrateurs du Tiers, aucun collaborateur du Tiers n'a accès au client secret.

XP Z12-013 – Standardisation des API SI/SC ⇔ PA

7.5 Processus de délégation avec une authentification unique par Tiers au sein de chaque Fournisseur API

7.5.1 Description du processus

Ce processus se déroule comme suit :

1. **Référencement du Tiers**

Le Tiers doit se référencer en créant un compte d'entreprise Tiers auprès de la PA / du Fournisseur API de l'Entreprise.

2. **Délégation du Tiers par l'Entreprise**

Depuis son Fournisseur API, un utilisateur habilité de l'Entreprise accrédite le Tiers précédemment référencé en définissant son périmètre de délégation.

3. **Transmission de l'identifiant d'Entreprise**

Lorsque cela est requis, l'utilisateur de l'Entreprise communique au Tiers son identifiant d'Entreprise au sein de sa PA.

Cet identifiant correspond soit :

- au SIREN de l'Entreprise,
- soit à un identifiant technique interne propre au Fournisseur API.

Il est recommandé de privilégier l'utilisation du SIREN en tant qu'identifiant d'organisation dès lors que cela est possible.

4. **Action du Tiers pour le compte de l'Entreprise**

Lorsqu'il souhaite agir pour le compte d'une Entreprise, le Tiers s'authentifie à l'aide de ses propres mécanismes d'authentification et utilise l'identifiant d'Entreprise transmis.

7.5.2 Conformité du processus

1. Veiller à s'assurer de l'identité du Tiers :

- Le fournisseur API peut mettre en place un système de vérification d'identité du Tiers.

2. Délégation à une entreprise et non à un utilisateur : **Conforme**.

3. Objectif de sécurisation du client id/ client secret : **Conforme**.

- En dehors des administrateurs du Tiers, aucun collaborateur du Tiers n'a accès au client secret.

7.5.3 Indiquer le périmètre organisationnel de l'appel API

Le client Id pouvant être unique pour un Tiers, le Client API doit pouvoir indiquer sur quelle(s) organisation(s) il veut agir.

Le Client API doit indiquer l'identifiant de l'organisation sur laquelle il faut agir en renseignant avec une balise 'Organisation-Id' dans le header http.

XP Z12-013**Annexe A**
Swagger de l'API Flux
(Normative)

SWAGGER de l'API Flux :

PR XP Z12-013 - AFNOR-Flow_Service-1.3.0-swagger.json

XP Z12-013 – Standardisation des API SI/SC ⇔ PA**Annexe B****Swagger de l'API Annuaire
(Normative)**

SWAGGER de l'API Annuaire :

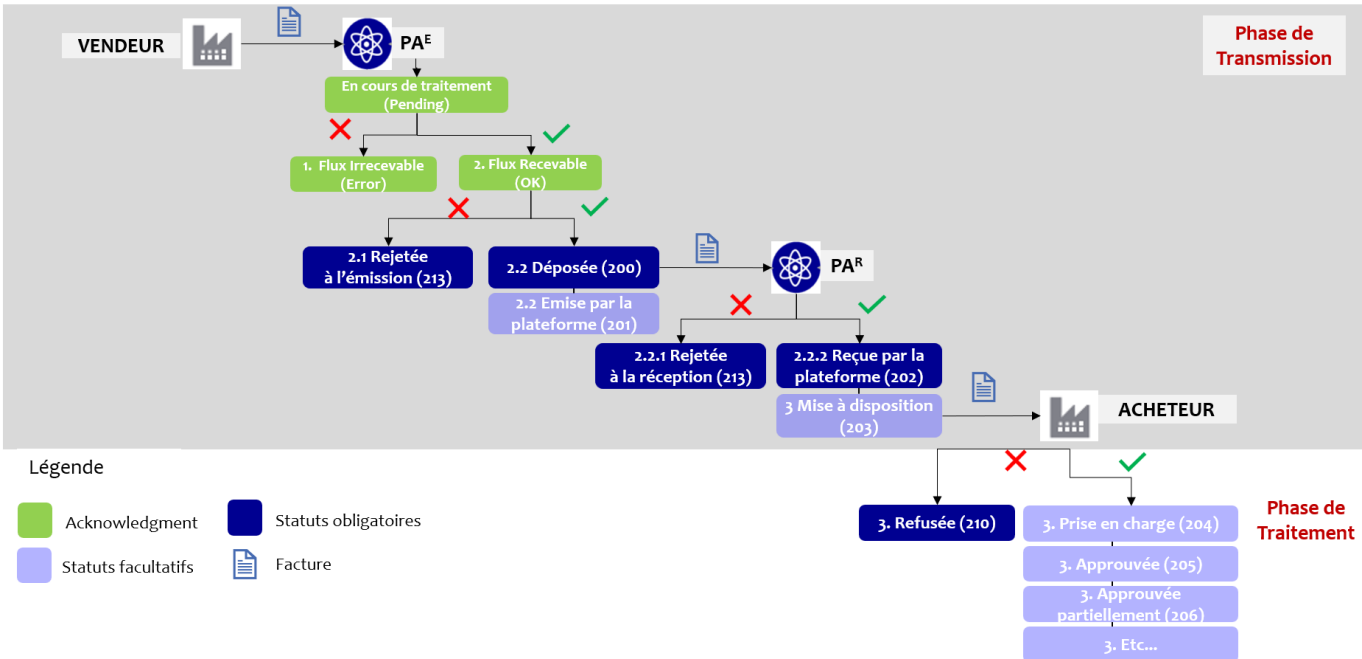
PR XP Z12-013 - AFNOR-Directory_Service-1.3.0-swagger.json

Annexe C

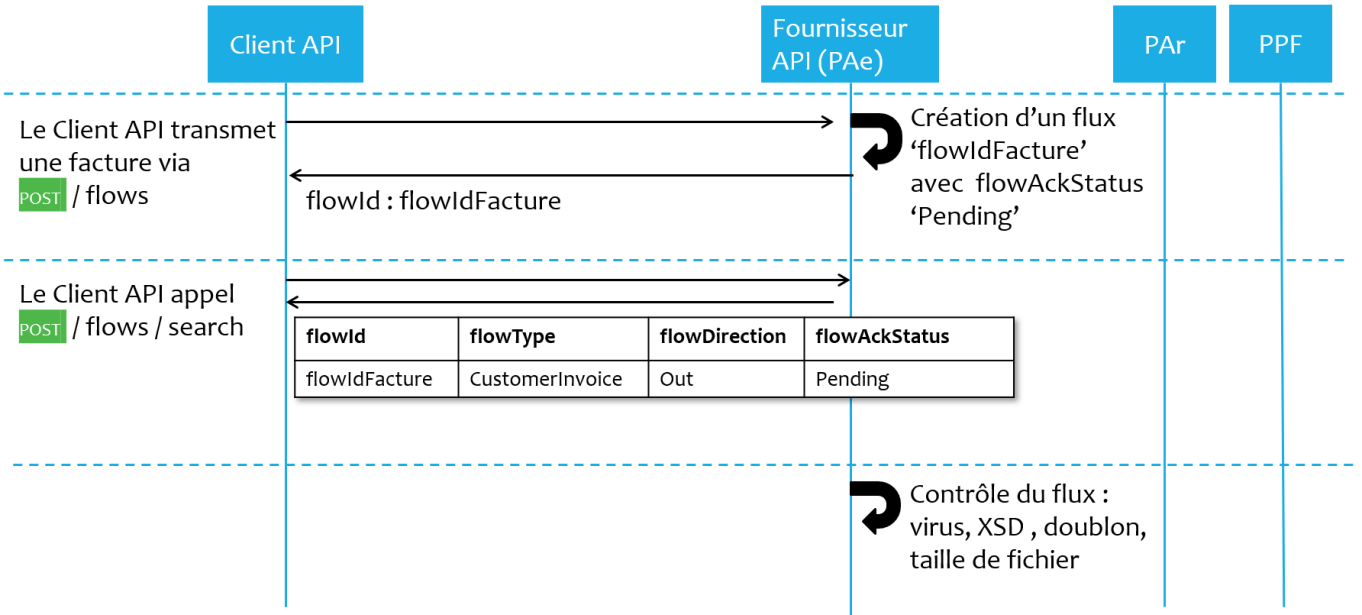
Diagrammes de séquence

(Normative)

Arbre des diagrammes

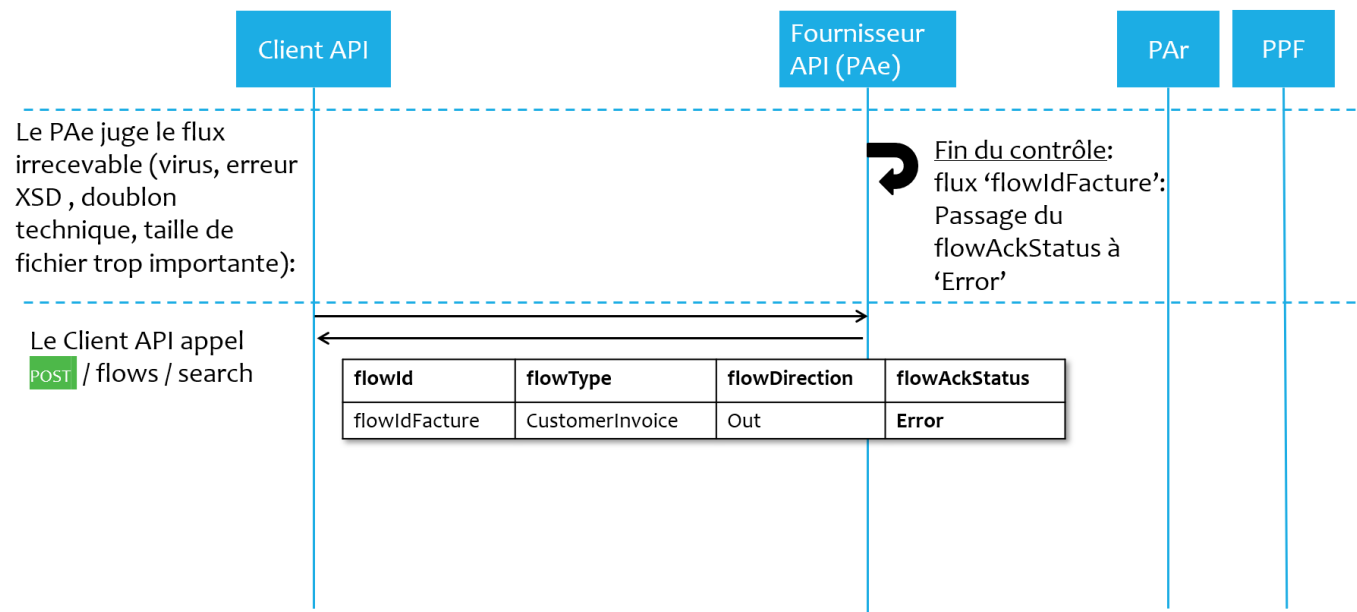


Le dépôt du Flux

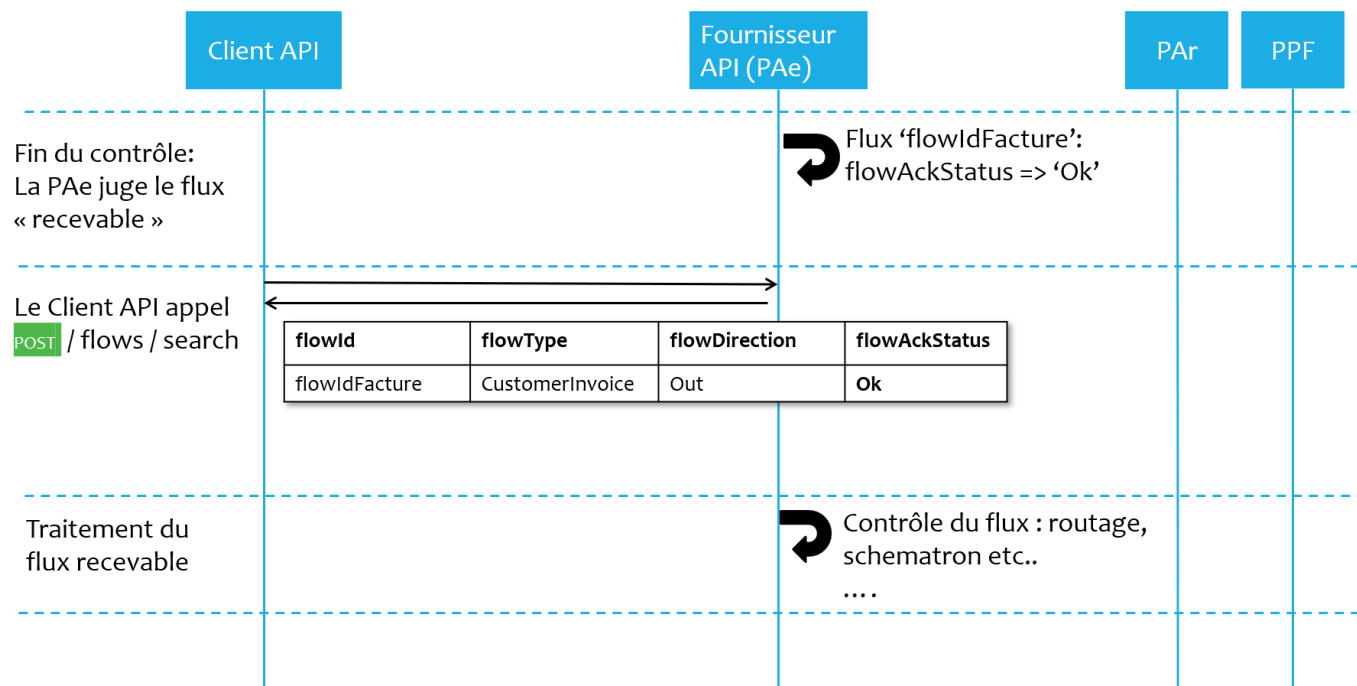


XP Z12-013 – Standardisation des API SI/SC ⇔ PA

1. Flux irrecevable

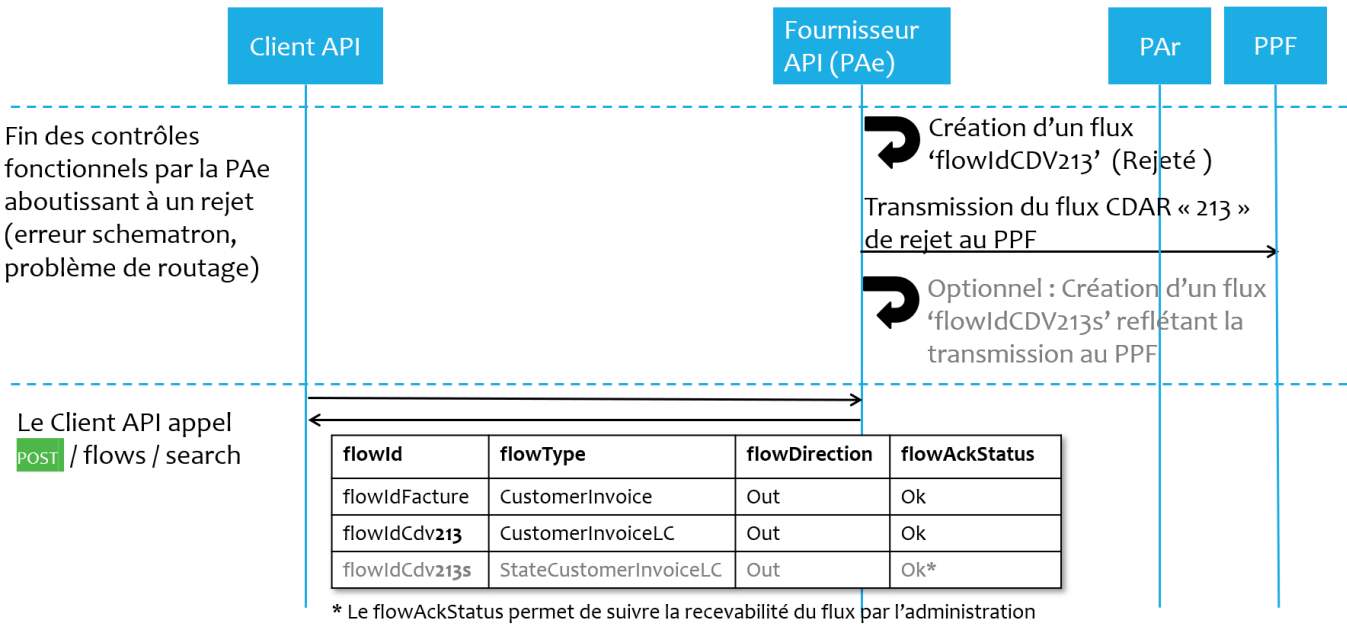


2 : Flux recevable

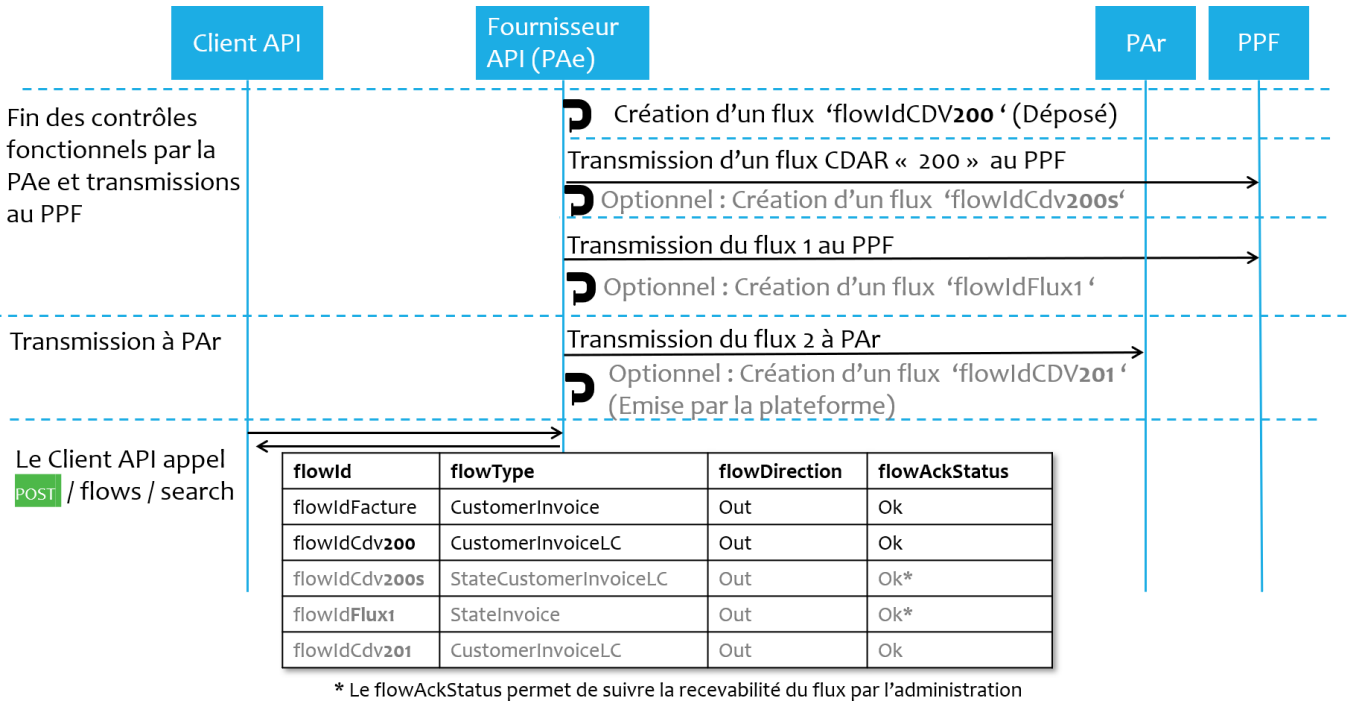


XP Z12-013

2.1 : Rejet par la PA d’émission (PAe)

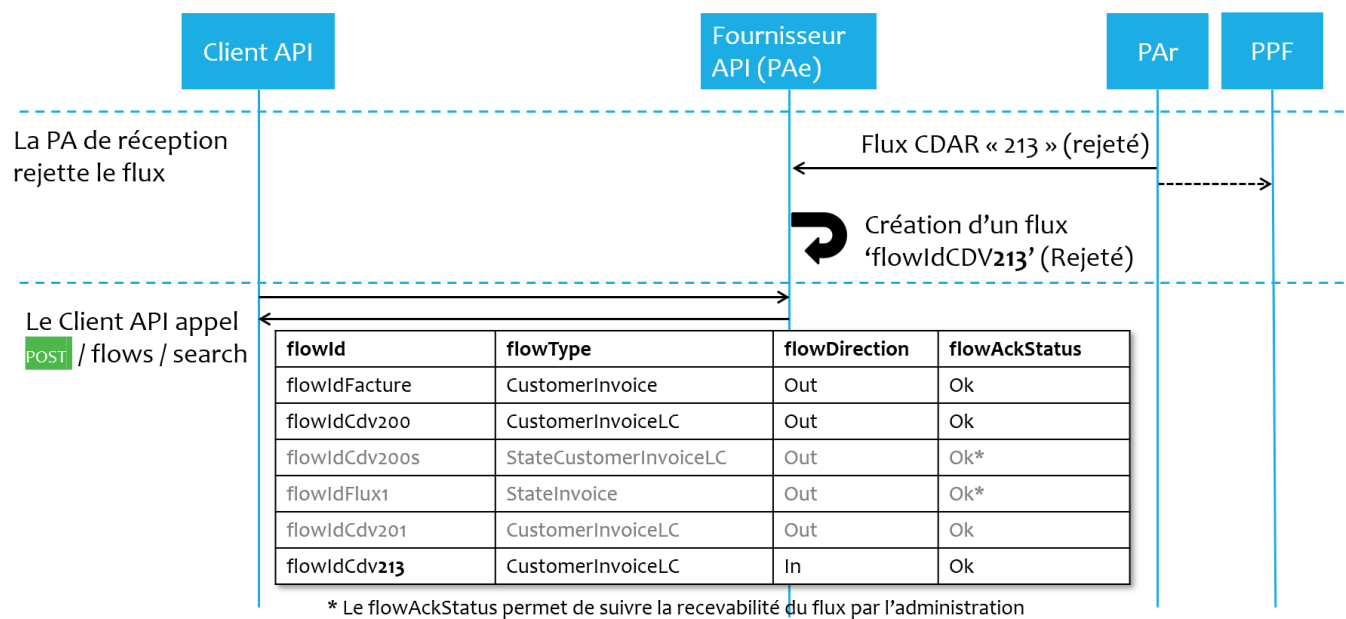


2.2 : Acceptation par la PA d’émission (PAe)

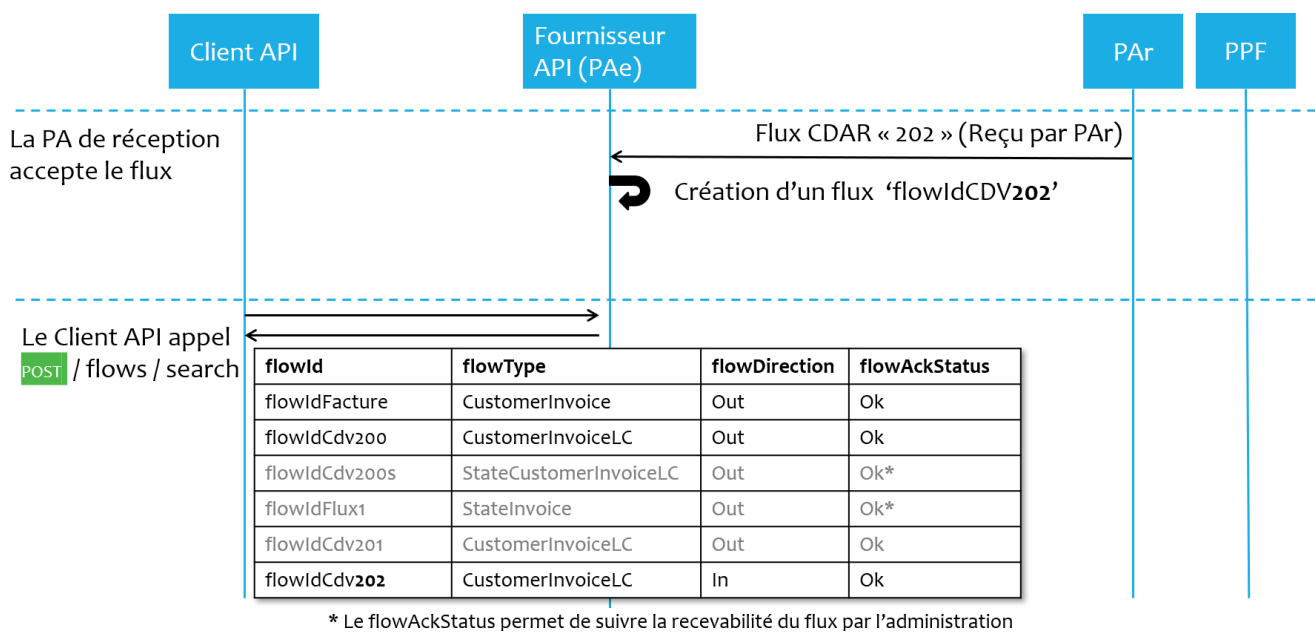


XP Z12-013 – Standardisation des API SI/SC ↔ PA

2.2.1 : Rejet par la PA de réception

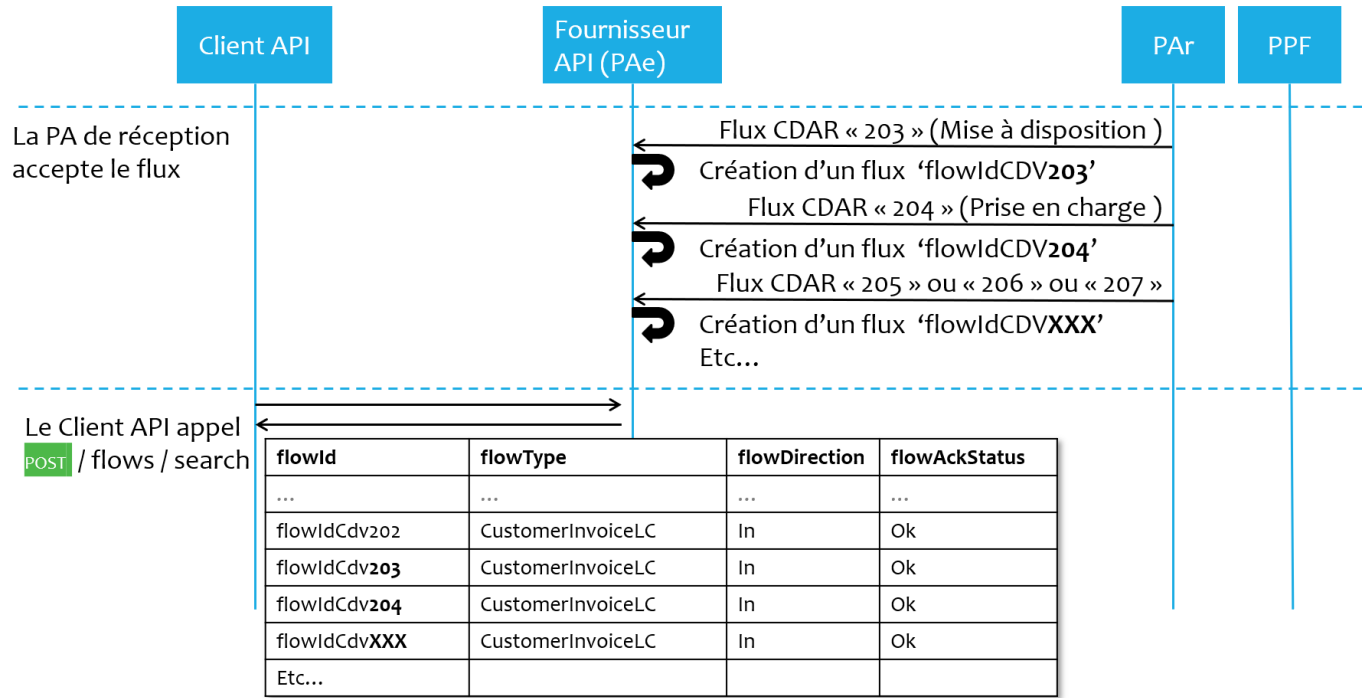


2.2.2 : Acceptation par la PA de réception



XP Z12-013

3 Autres statuts venant de PAr



XP Z12-013 – Standardisation des API SI/SC ⇔ PA**Bibliographie**

- [1] Dossier de spécifications externes de la Facture électronique 3.2 - Dossier général - Agence pour l'informatique financière de l'État.
- [2] Dossier de spécifications externes de la Facture électronique – Document Chorus Pro V1.1
- [3] Norme AFNOR XP-Z12-012 : *Formats et Profils des messages Factures et Statuts de cycle de vie, constitutifs du socle minimal applicable à la Réforme Facture Électronique en France*
- [4] Norme AFNOR XP-Z12-014 : *Cas d'usage B2B applicables dans le cadre la Réforme Facture Électronique en France*