

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 : 11/12/2025 à 12:48

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

ISSN 0335-3931

AFNOR FE : Facture électronique

Normalisation française

Norme expérimentale publiée par AFNOR

XP Z12-013

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

Date de publication : novembre 2025

En tant que titulaire des droits d'auteurs sur ce document, AFNOR en 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 OD/PDP

Sommaire

Avant-propos.....	4
Introduction	7
1. Domaine d'application	8
2 Références normatives	9
3 Termes et définitions.....	10
4 Règles communes à toutes les routes API	14
4.1 Respect des SWAGGERS	14
4.2 Authentification	14
4.3 Pagination.....	14
4.4 Route GET / healthcheck.....	14
5 API – Flux.....	14
5.1 Respect du Swagger Flow Service	14
5.2 Déposer un flux (Route POST / Flows)	15
5.2.1 Flux acceptés.....	15
5.2.2 'trackingId'.....	15
5.2.3 'sha256'.....	15
5.2.4 'processingRule' et 'processingRuleSource'	15
5.2.5 Retour de la route.....	16
5.2.6 Qualification du flux avec un 'FlowType'	16
5.3 Rechercher un flux (Route POST / Search)	18
5.3.1 Gestion du différentiel.....	18
5.3.2 Garantir l'exhaustivité des flux.....	18
5.4 Récupérer un flux (Route GET / flows / {flowId})	18
5.4.1 Paramètre 'docType'	18
5.4.2 Récupération du lisible d'une facture.....	19
5.5 Gestion des flux irrecevables et des rejets.....	19
5.5.1 Irrecevabilité d'un flux envoyé par le Client API au Fournisseur API.....	19
5.5.2 Rejet d'un flux facture par la PAe ou la PAr	20
5.5.3 Rejet d'un flux Cycle de Vie sur une Facture.....	20
5.6 Informer le Client API de l'arrivée ou de la modification d'un flux par un Webhook.....	20
5.6.1 Usage du Webhook et de la route POST Search.....	20
5.6.2 Configurer le Webhook.....	21
5.6.3 Fonction de rappel du WebHook	21

XP Z12-013

5.6.4	Sécurisation du WebHook	21
6	API – Annuaire	21
6.1	Respect du Swagger Directory Service.....	21
6.2	Consultation de l'Annuaire.....	21
6.3	Mise à jour de l'Annuaire	23
	Annexe A - Swagger de l'API Flux (Normative).....	24
	Annexe B - Swagger de l'API Annuaire (Normative)	25
	Annexe C – Diagrammes de séquence (Normative)	26
	Bibliographie.....	31

XP Z12-013 – Standardisation des API OD/PDP

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 OD), 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	<p>Version révisée avec les évolutions suivantes :</p> <p>3 Références normatives</p> <p>Ajout de la RFC 9421 pour la gestion des Webhook</p> <p>3 Termes et définitions</p> <p>Ajout des définitions ci-dessous :</p> <ul style="list-style-type: none"> • FRR • Webhook • XML Schema / XSD <p>4 Règles communes à toutes les routes API</p> <ul style="list-style-type: none"> • Paramètre 'offset' non disponible pour la pagination lorsque le jeu de donné est dynamique. • Précision sur le degré de liberté de l'URL mise à disposition par le Fournisseur API. <p>5 API Flux</p> <p>Route POST / Flows</p> <ul style="list-style-type: none"> • 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 <p>Route POST / Search</p> <ul style="list-style-type: none"> • 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 <p>Route GET / Flows</p> <ul style="list-style-type: none"> • Ajout du docType 'MetaData' dans le paramètre 'docType' pour récupérer l'objet représentant un Flux. <p>Nouveau chapitre sur "La gestion des flux irrecevables et des rejets"</p> <p>Nouveau chapitre sur "Informer le Client API de l'arrivée ou de la modification d'un flux par un Webhook"</p> <p>5 API Annuaire</p> <p>Restriction sur la consultation de l'annuaire à des recherches sur un SIREN complet pour éviter des problèmes de performance.</p> <p>Suppression de la route PATCH/directory-line.</p>

XP Z12-013 – Standardisation des API OD/PDP

N° de Version	Date de Version	Description des évolutions
		Annexes Création des annexes A et B et C

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 OD (Opérateur de Dématérialisation) 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...

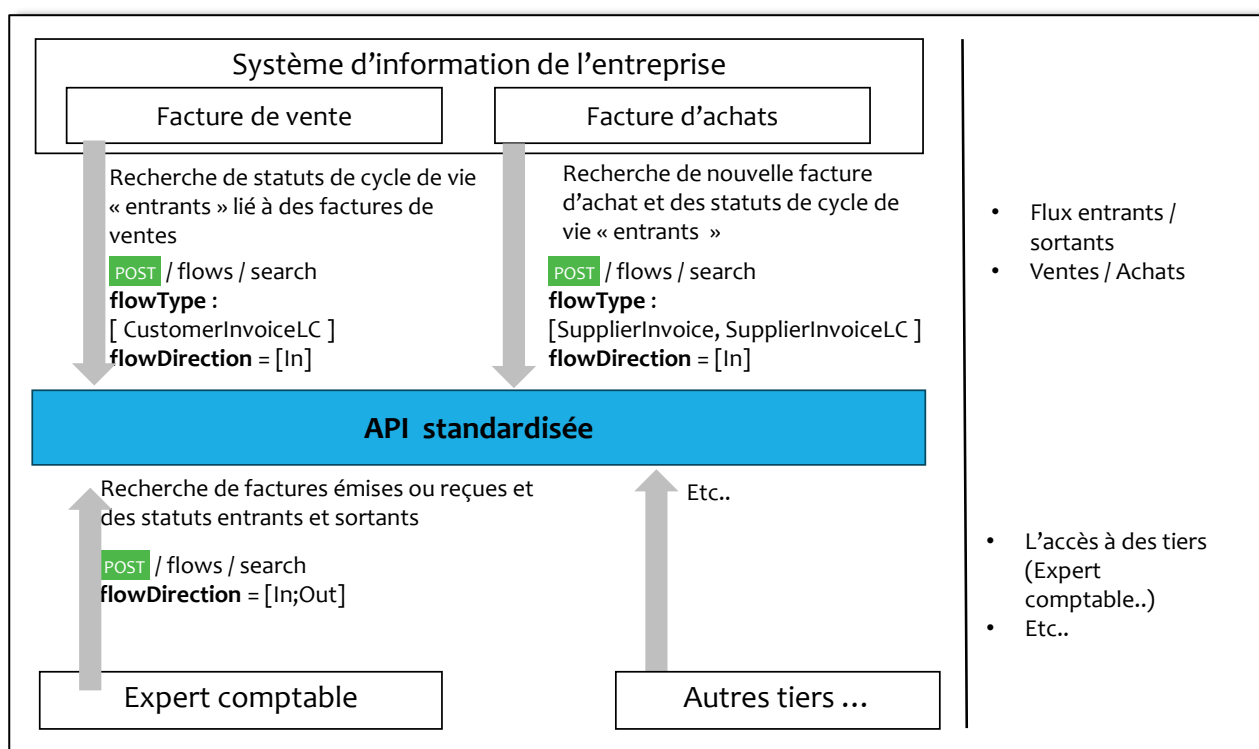


Figure 1 — Segmentation des Flux

- 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 OD/PDP

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 Opérateurs de Dématérialisation (OD) et les Systèmes d'Information (SI) des entreprises avec l'ensemble des Plateformes Agréées (PA). L'objectif est de faciliter l'interconnexion des OD 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, ...).

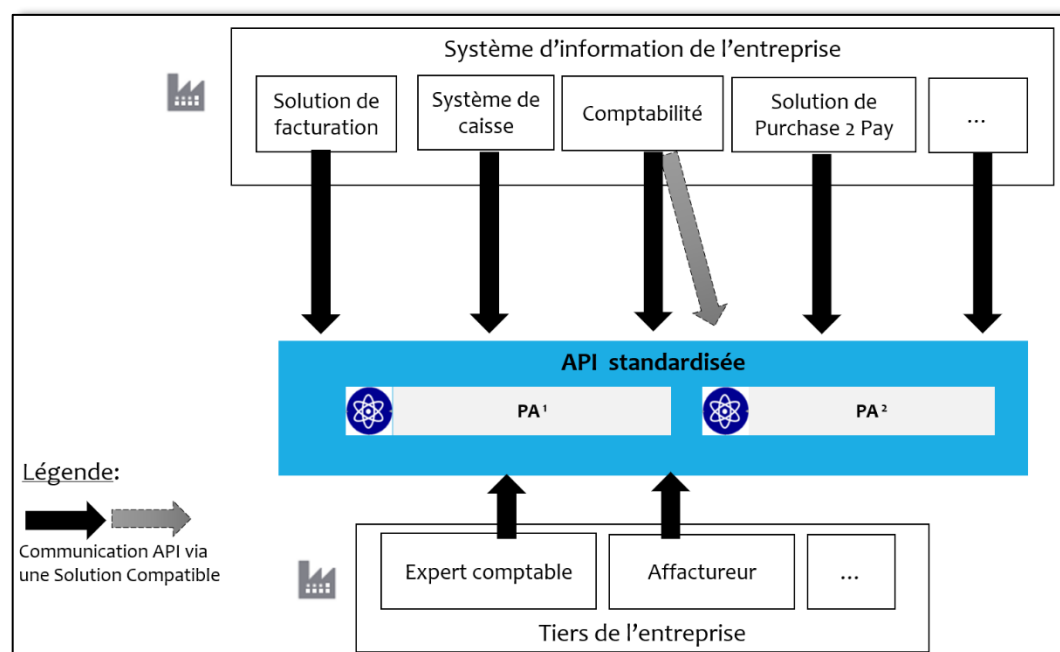
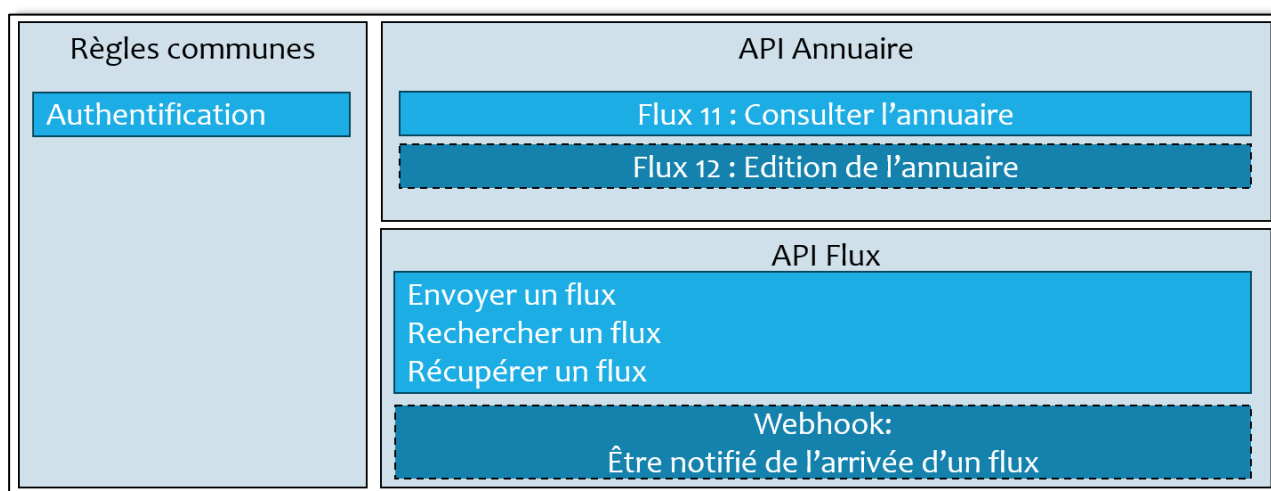


Figure 2 — Faciliter l'accès aux Plateformes Agréées

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

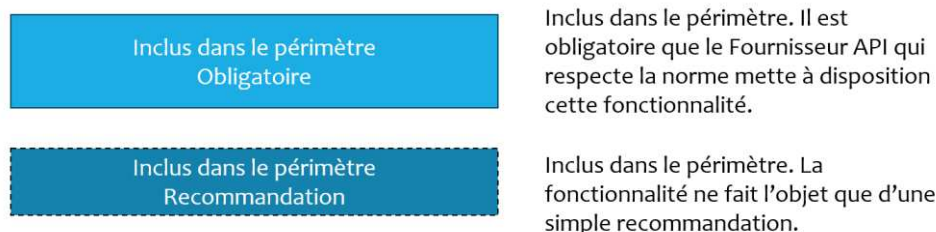


Figure 3 — Périmètre des API

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*
- 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.

XP Z12-013 – Standardisation des API OD/PDP

- 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 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.

XP Z12-013

3.5

BT

Business Term' au sens de la norme EN16931

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 un OD.

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/

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

XP Z12-013 – Standardisation des API OD/PDP

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 EN16931.

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

- Profil EN16931, qui une CIUS pour la France de l'implémentation de la Norme EN16931
- Profil EXTENDED-CTC-FR, qui est une EXTENSION pour la France de l'implémentation de la Norme EN16931

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 un OD.

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

OD (Opérateurs de Dématérialisation)

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 OD 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 d'OD.

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

XP Z12-013 – Standardisation des API OD/PDP

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.

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 compatibles avec le protocole d'authentification OAUTH2 et utiliser un 'bearer' au sens de la RFC 6750.

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

Si un même compte OAUTH2 est utilisé pour accéder à plusieurs entreprises, le 'bearer' peut contenir l'information de l'entreprise sur laquelle l'appel agit.

Ce sujet est susceptible d'être amendé et précisé dans une version ultérieure.

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.

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.

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 :

XP Z12-013

Tableau 1 — Routes obligatoires de l'API Flow Service

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

5.2 Déposer un flux (Route POST / Flows)

5.2.1 Flux acceptés

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 'flowInfo' et d'un fichier binaire.

5.2.2 'trackingId'

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'

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.

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.

XP Z12-013 – Standardisation des API OD/PDP

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] 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.
B2B	Il est recommandé au Fournisseur API de traiter le flux comme un flux qui relève du e-invoicing
B2Bint	Il est recommandé au Fournisseur API de traiter le flux comme un flux qui relève du e-reporting des ventes ou des acquisitions B2Bint
B2C	Il est recommandé au Fournisseur API de traiter le flux comme un flux qui relève du e-reporting des ventes B2C
OutOfScope	Il est recommandé au Fournisseur API de traiter le flux comme un flux « hors réforme »
ArchiveOnly	Il est recommandé au Fournisseur API de 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). Il peut s'agir d'un AVOIR interne créer pour annuler une facture REJETÉE ou REFUSÉE ou d'une reprise d'historique.
NA	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 CDAR ou les flux de E-Reporting

5.2.5 Retour de la route

Le Fournisseur API doit retourner en synchrone un objet avec une propriété 'flowId' permettant d'identifier de manière unique le flux dans son système d'information pour un Client API donné.

Il est recommandé au Fournisseur API de calculer l'empreinte numérique du fichier déposé en utilisant l'algorithme sha256 et de la retourner dans une propriété 'sha256'.

5.2.6 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.

Le Fournisseur API doit qualifier les flux selon le Tableau 3 ci-dessous.

Si un flux 10 contient plusieurs types de flux de E-Reporting, le Fournisseur API doit qualifier un flowType « MultiFlowReport ».

XP Z12-013

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 « StateInvoice ».
- 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.

Tableau 3 — Qualification des flux avec un FlowType

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
StateInvoice	Peut être utilisé pour qualifier Flux 1 envoyé par la PAe au Concentrateur de Données du PPF
CustomerInvoiceLC	Doit être utilisé pour qualifier un flux cycle de vie (CDAR) 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'
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)
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 B2Cagrégées (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)

XP Z12-013 – Standardisation des API OD/PDP

FlowType	Règle de gestion
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 différente (FRR flux 10)

5.3 Rechercher un flux (Route POST / Search)

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

XP Z12-013

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
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

XP Z12-013 – Standardisation des API OD/PDP

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 Usage du Webhook et de la route POST Search

Le Webhook permet au Client API d'être informé de l'arrivée ou de la modification d'un flux et évite d'appeler périodiquement la route POST Search.

La route POST Search est complémentaire au Webhook et est à utiliser :

- Lorsque la mise en place d'un Webhook est complexe. C'est en particulier le cas lorsque le client API ne dispose pas d'un Serveur WEB permettant de recevoir des appels de l'extérieur.
- Pour récupérer un historique de flux. Ce cas peut en particulier arriver lorsque le Serveur WEB du Client API a été indisponible et a empêché le Client API de recevoir les notifications.

XP Z12-013

5.6.2 Configurer le Webhook

Il est recommandé que le Fournisseur API mette à disposition un dispositif permettant au Client API de configurer le Webhook en permettant à ce dernier d'indiquer les notifications qu'il souhaite recevoir. Il est ainsi recommandé au Fournisseur API de permettre au Client API de filtrer les notifications selon les critères ci-dessous :

- La liste des 'FlowType' à notifier
- La liste des 'FlowDirection' à notifier
- La liste des 'FlowAckStatus' à notifier

Note 1 au 5.6.2 : le dispositif permettant au Client API de configurer le Webhook peut par exemple être une interface utilisateur de paramétrage, un fichier de configuration ou un paramétrage lié à la gestion des droits.

5.6.3 Fonction de rappel du WebHook

Il est recommandé que le Fournisseur API notifie le Client API de l'arrivée ou d'une modification d'un flux selon la configuration du WebHook par une requête HTTP POST avec un 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 au 5.6.2 : les propriétés permettant au Client API d'identifier l'organisation peuvent par exemple être :

- 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...

5.6.4 Sécurisation du WebHook

Si le Fournisseur API met à disposition un Webhook, le Fournisseur API doit le sécuriser.

Cette section fera l'objet d'une publication ultérieure en Janvier 2026.

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

Le Fournisseur API doit publier les routes API obligatoires listées dans le tableau 6 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 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.

Note 1 au 6.2 : Il est rappelé au Fournisseur API, qu'il ne doit retourner que les données publiables. Seules les lignes d'adressage actives doivent être retournées. Une ligne active est une ligne dont la date d'effet est antérieure à la date du jour.

XP Z12-013 – Standardisation des API OD/PDP

Tableau 6 — 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)
	GET/siren/id-instance:{id-instance}	Rechercher un SIREN avec l'id-instance
SIRET	POST/siret/search	Rechercher un SIRET avec plusieurs critères en JSON
	GET/siret/code-insee:{siret}	Rechercher un SIRET avec le SIRET (test d'existence ou récupération des informations du SIRET)
	GET/siret/id-instance:{id-instance}	Rechercher un SIRET avec l'id-instance
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)
	GET/routing-code/id-instance:{id-instance}	Rechercher un code routage avec l'id-instance
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).
	GET/directory-line/id-instance:{id-instance}	Rechercher une ligne d'adressage avec l'id-instance.

XP Z12-013**6.3 Mise à jour de l'Annuaire**

Il est recommandé que le Fournisseur API publie les routes API listées dans le tableau 7 ci-dessous :

Tableau 7 — Routes de demande de mise à jour de l'Annuaire

Ressource	Route	Description
Code routage	POST/routing-code	Demander la création d'un code routage
	PATCH/routing-code/id-instance{id-instance}	Demander la modification d'un code de routage
	PUT/routing-code/id-instance{id-instance}	Demander la modification d'un code de routage
Ligne d'adressage	POST/directory-line	Demander la création d'une ligne d'adressage
	PATCH/directory-line/id-instance:{id-instance}	Demander la modification d'une ligne d'adressage
	DELETE/directory-line/id-instance:{id-instance}	Demander le masquage d'une ligne d'adressage dont la date de début d'effet n'a pas été atteinte

XP Z12-013 – Standardisation des API OD/PDP**Annexe A- Swagger de l'API Flux
(Normative)**

SWAGGER de l'API Flux :

PR XP Z12-013 - AFNOR-Flow_Service-1.1.0-swagger.json

XP Z12-013

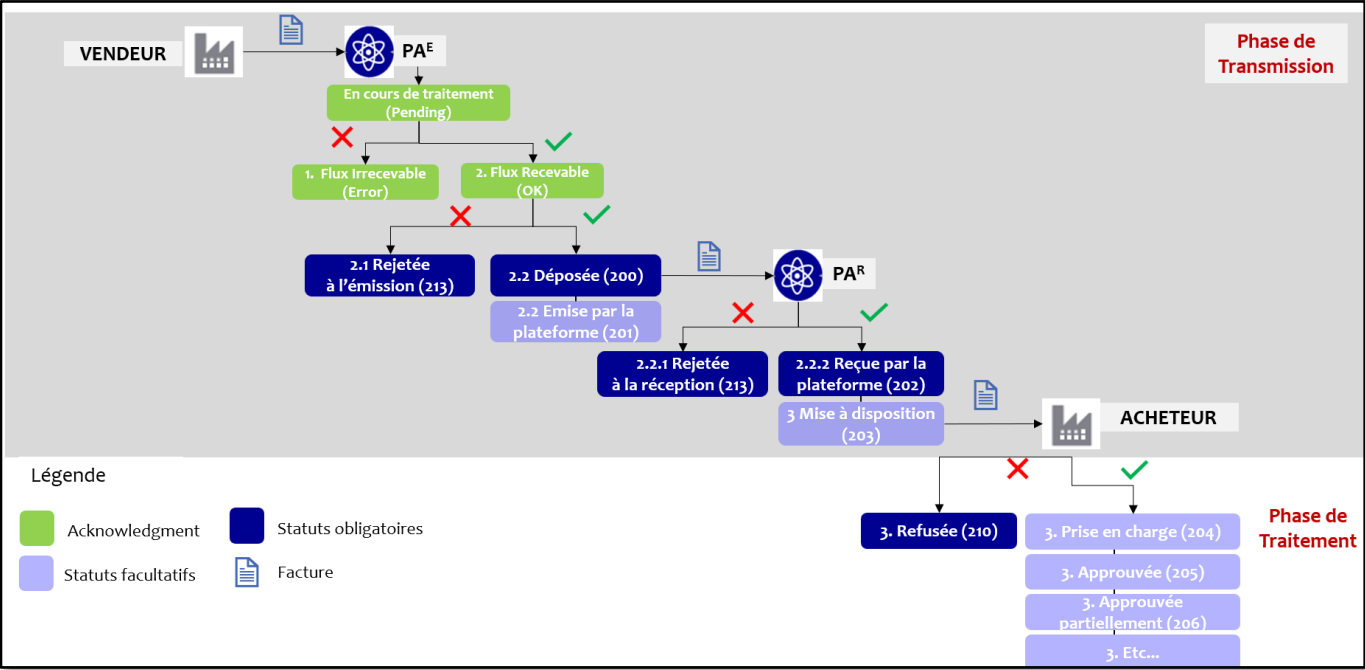
Annexe B- Swagger de l'API Annuaire (Normative)

SWAGGER de l'API Annuaire :

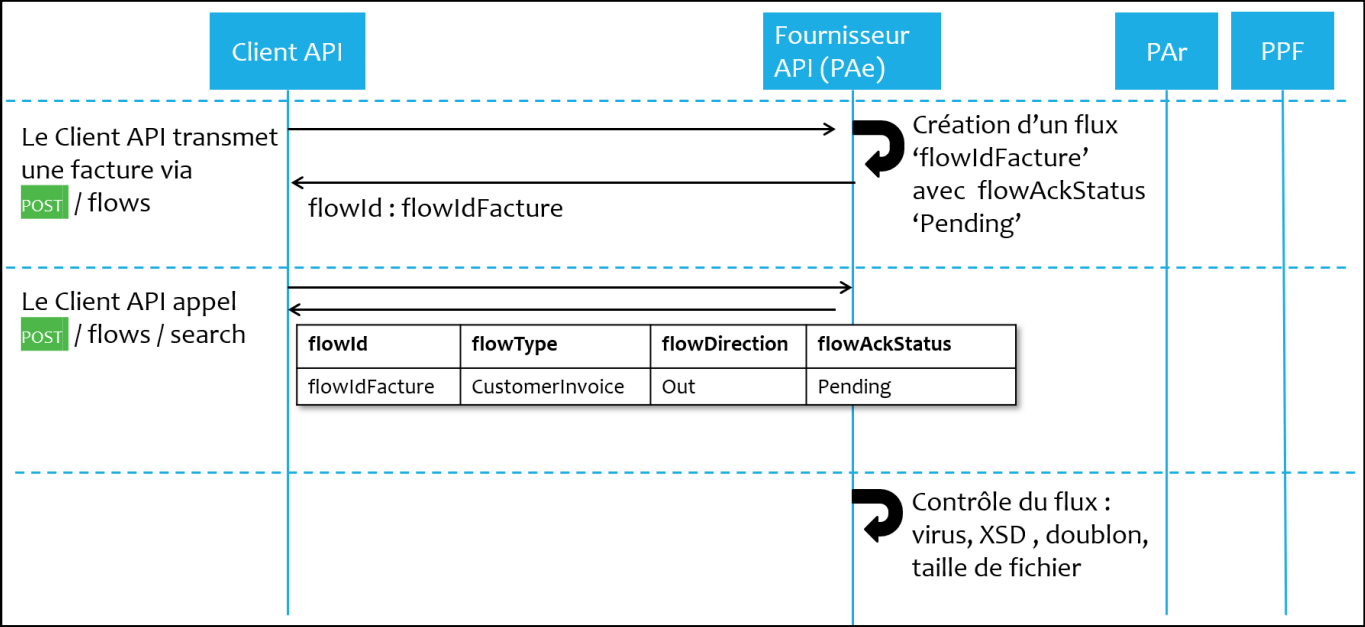
PR XP Z12-013 - AFNOR-Directory_Service-1.1.0-swagger.json

Annexe C – Diagrammes de séquence (Normative)

Arbre des diagrammes

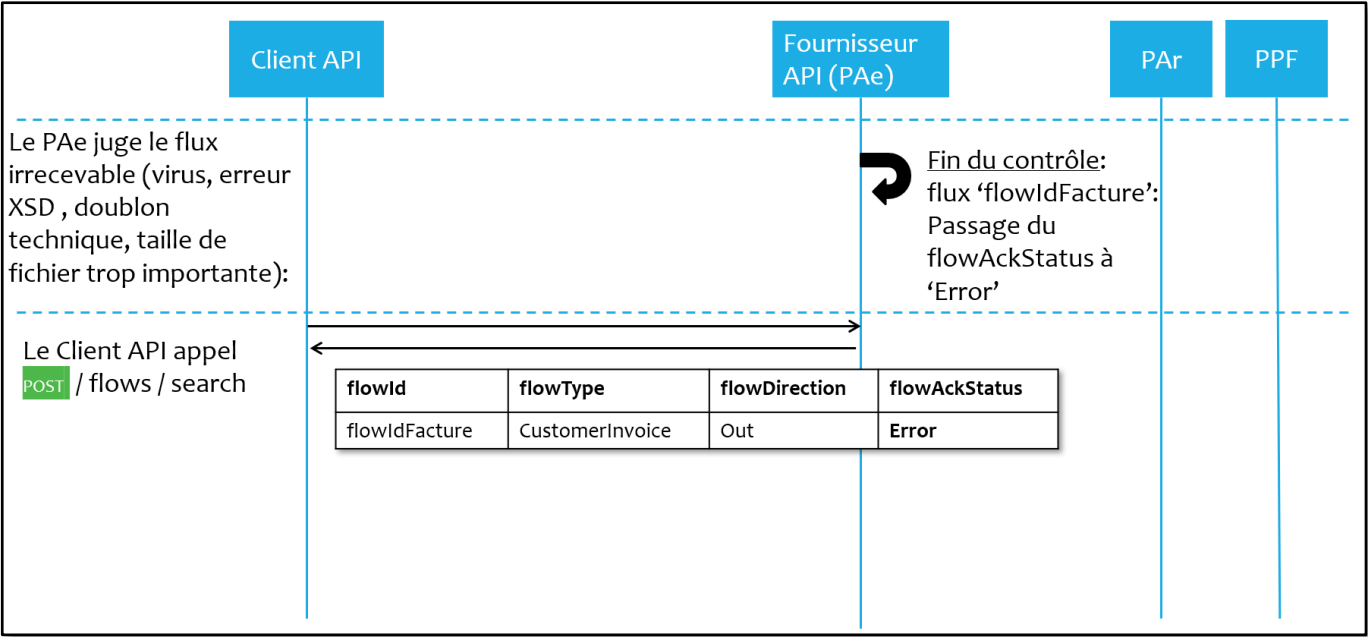


Le dépôt du Flux

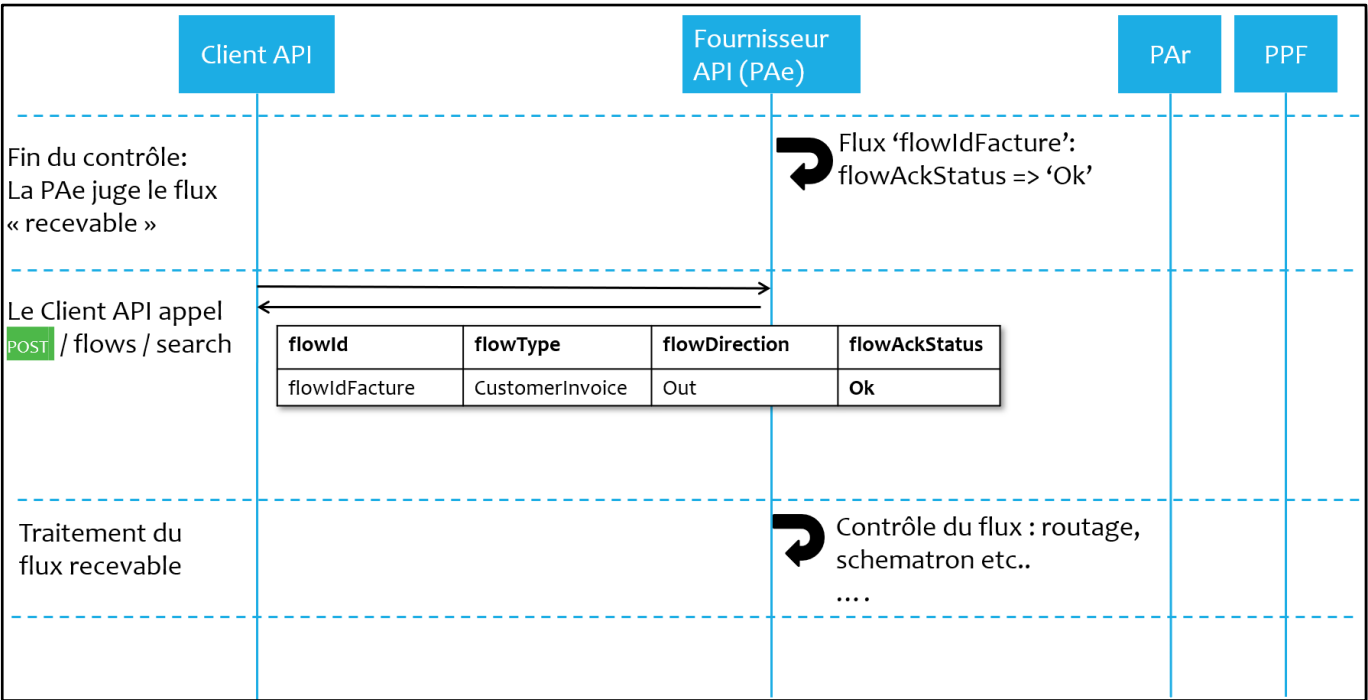


XP Z12-013

1. Flux irrecevable

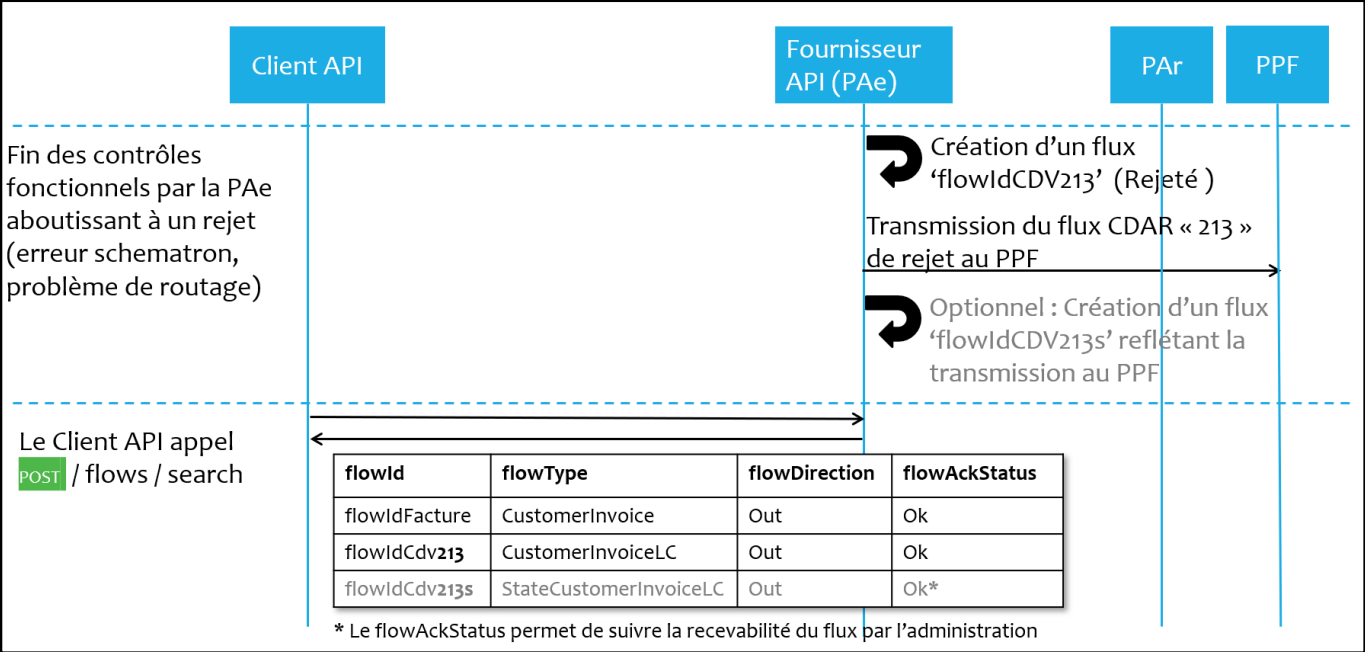


2. Flux recevable

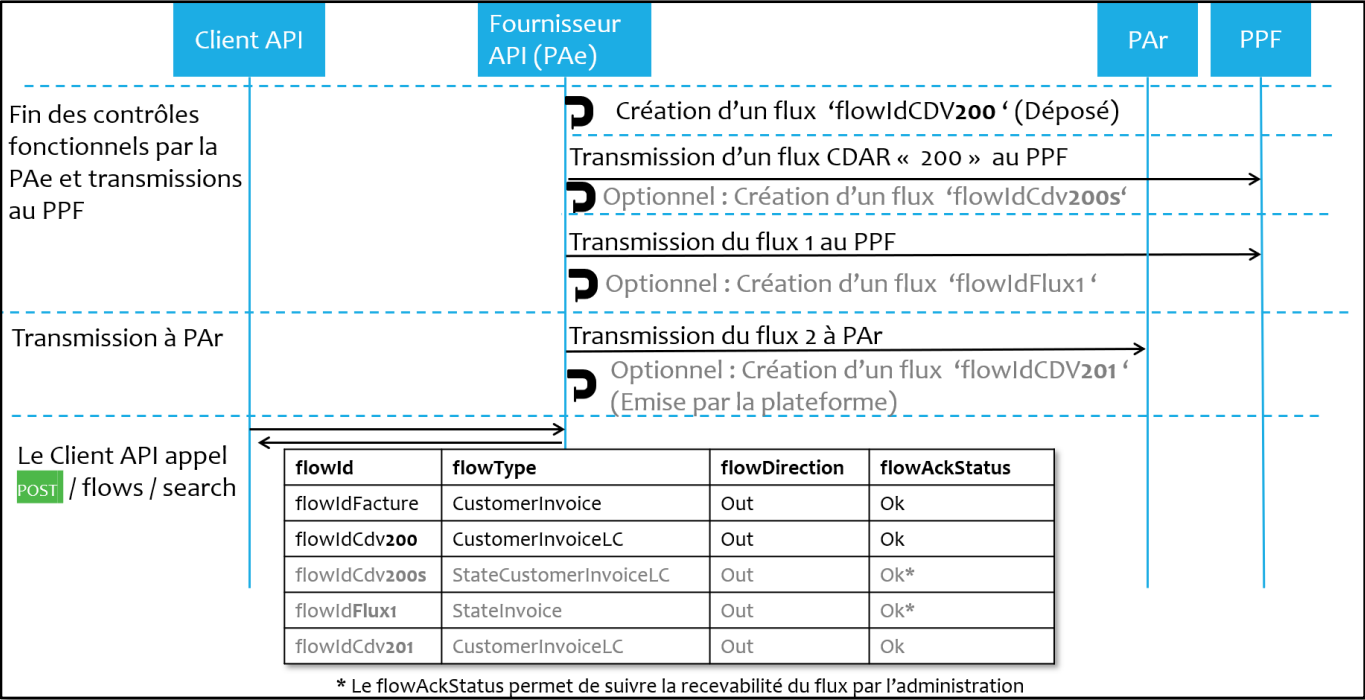


XP Z12-013 – Standardisation des API OD/PDP

2.1 Rejet par la PA d’émission (PAe)

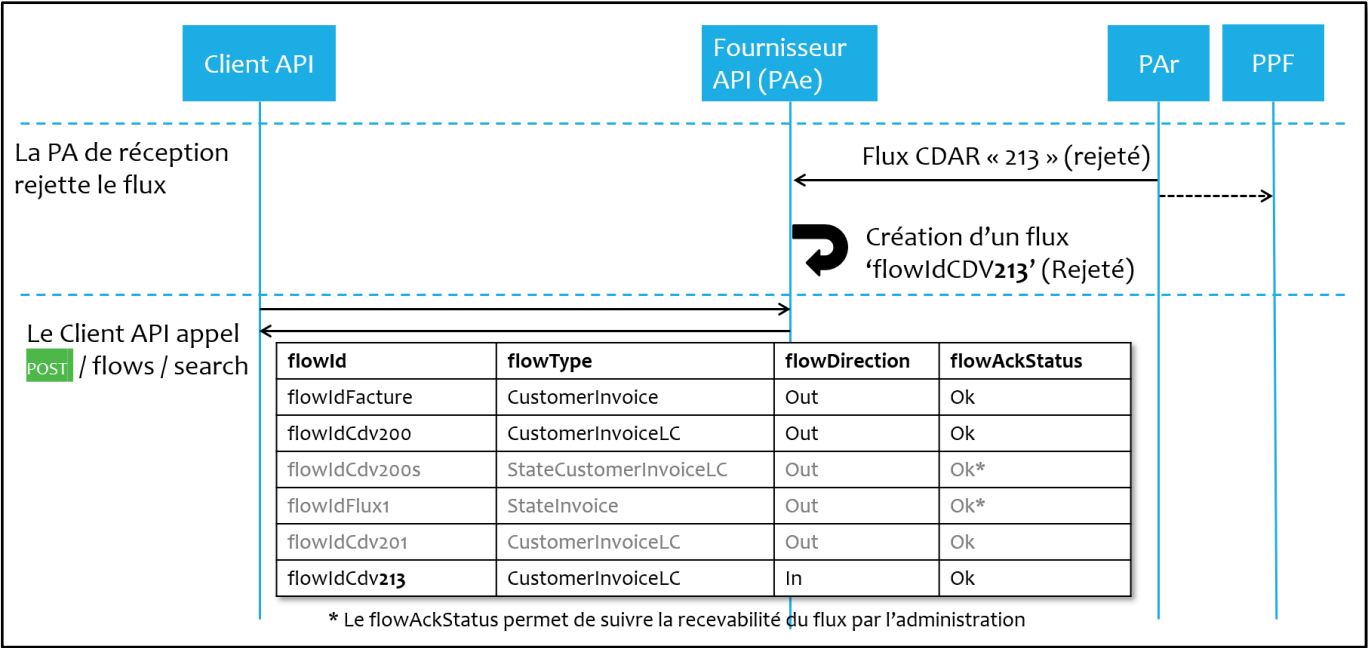


2.2 Acceptation par la PA d’émission (PAe)

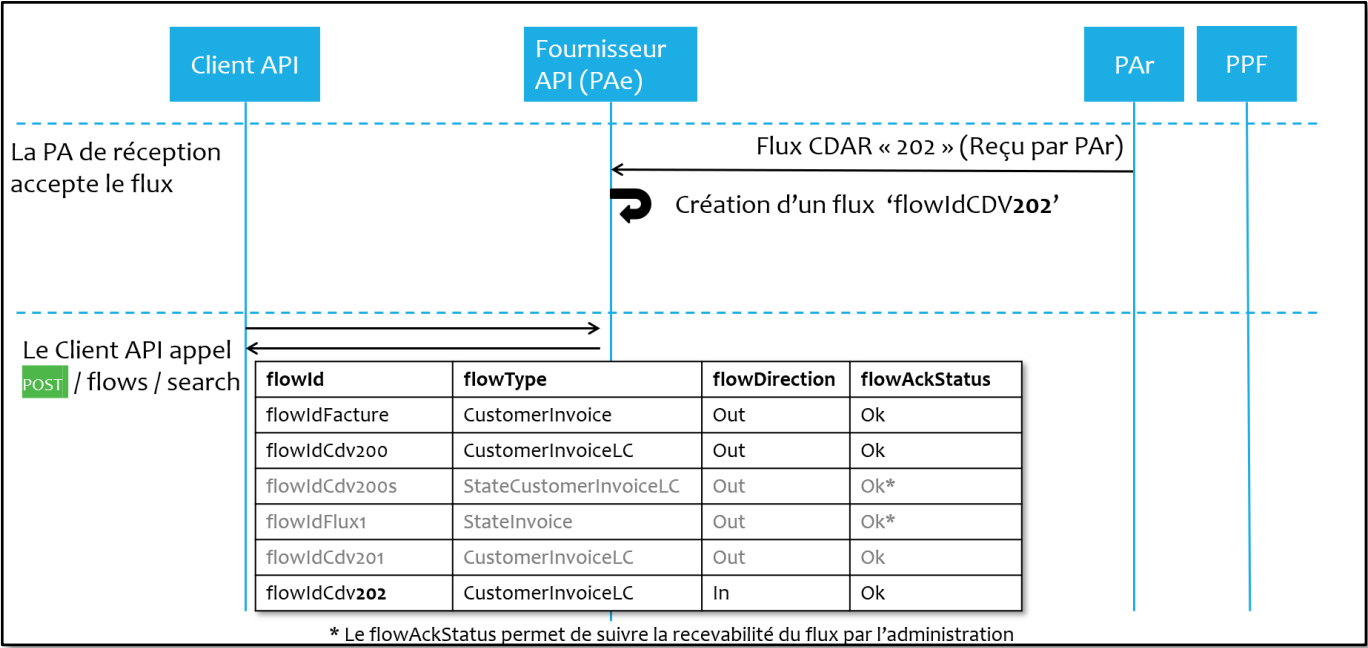


XP Z12-013

2.2.1 Rejet par la PA de réception

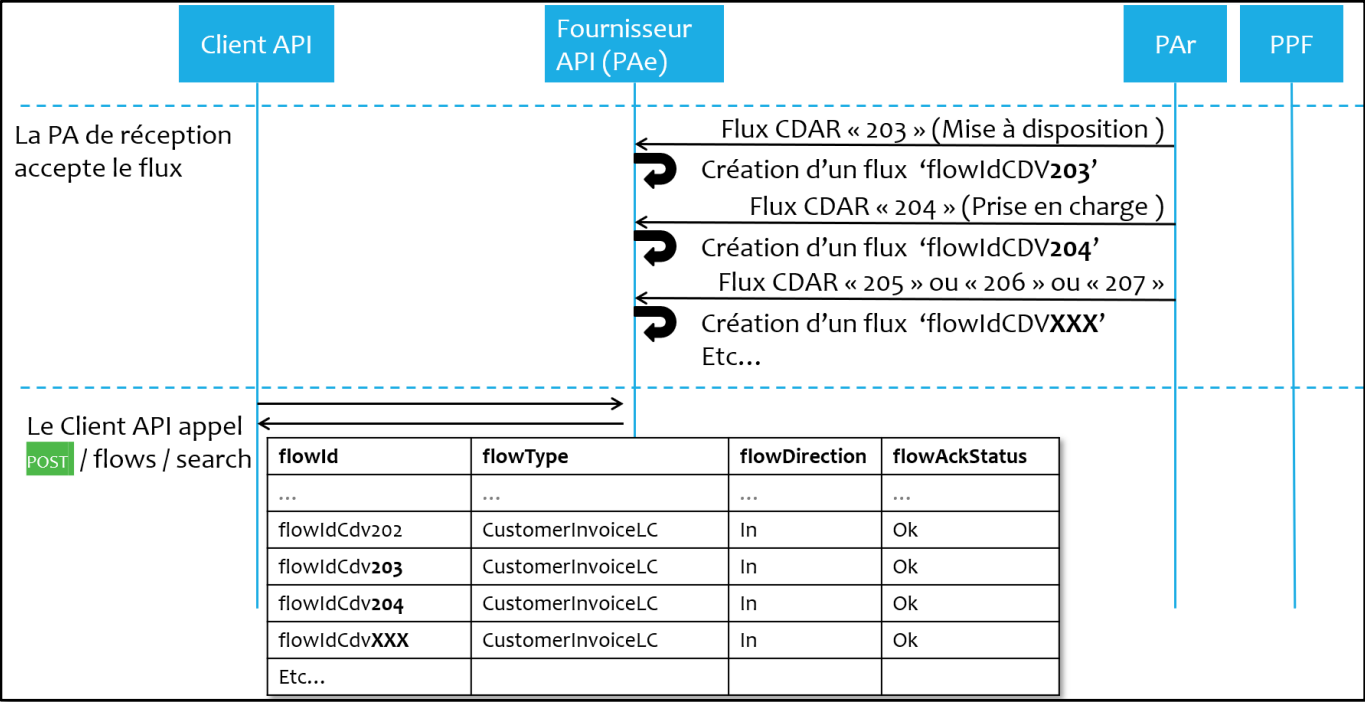


2.2.2 Acceptation par la PA de réception



3 Autres statuts venant de PAr

XP Z12-013 – Standardisation des API OD/PDP



XP Z12-013**Bibliographie**

- [1] Dossier de spécifications externes de la Facture électronique 3.1 - Dossier général - Agence pour l'informatique financière de l'État.