



CONTRAT D'INTERFACE – API DE COLLECTE SIDOBA

CNSA

SIDOBA

Référence :
Date : 24/10/2024
Version : 2.3

HISTORIQUE

[Ce chapitre présente l'historique des versions du Dossier d'Architecture Technique basés sur 4 entrants : la date de la version, la version, le rédacteur de la version et l'objet de la version]

Date	Version	Rédacteurs	Objet
19/09/2023	0.1	Olivier NYECK	Création du document
22/09/2023	1.0	Olivier NYECK	Livraison initiale
18/10/2023	2.0	Olivier NYECK	Prise en compte des retours éditeurs Ajout du contrat d'interface pour la collecte de données de niveau individuelles Modifications principales impactant les chapitres 2, 3.6, 4.1 et 4.2.
14/11/2023	2.1	Olivier NYECK	Description de la totalité des champs du document CDA Description d'une basic-auth localisée dans son propre chapitre Description des headers localisée dans son propre chapitre Modification des signatures et objets des services GET pour implémenter une basic-auth
13/12/2023	2.2	Olivier NYECK	Mise à jour du schéma d'orchestration des services API pour la collecte de données individuelles Mise à jour des signatures des services pour une cohérence avec le swagger Mise à jour des headers à transmettre par service
24/10/2024	2.3	Jérémy DESPEAUX	Généralisation API Mise à jour WS 05 pour prendre en compte la collecte de données individuelles clé/valeur Ajout du WS 08 pour la gestion des traitements

TABLE DES MATIERES

[Table des matières du document. A mettre à jour après chaque modification effectuée sur le document]

1	Introduction	5
1.1	Périmètre du document	5
1.2	Objet du document	5
1.3	Public concerné	5
1.4	Documents de référence	6
2	Présentation des services échangés	8
2.1	Niveau des données échangées entre les progiciels partenaires et SIDOBA	8
2.1.1	Stockage des données métier	8
2.1.2	Collecte des données	8
2.1.3	Format des messages appelant l'API de Collecte	9
2.2	Vue globale	9
2.2.1	Orchestration des appels aux services de collecte de données de niveau structure et dossier	10
2.2.2	Orchestration des appels aux services de collecte de données de niveau individuel	15
2.3	Synthèse des services consommés	19
3	Description des interfaces	20
3.1	WS01 Récupérer les campagnes ouvertes	20
3.1.1	Présentation	20
3.1.2	Format des messages échangés	20
3.2	WS02 Récupérer un modèle métier	22
3.2.1	Présentation	22
3.2.2	Format des messages échangés	22
3.3	WS03 Valider un modèle métier	23
3.3.1	Présentation	23
3.3.2	Format des messages échangés	23
3.4	WS04 Comparer deux modèles métier	25
3.4.1	Présentation	25
3.4.2	Format des messages échangés	25
3.5	WS05 Collecter des données cle/valeur	27
3.5.1	Présentation	27
3.5.2	Format des messages échangés	27
3.6	WS07 Collecter des données de niveau individuel	30
3.6.1	Présentation	30
3.6.2	Format des messages échangés	31
3.7	WS06 Consulter l'état d'une collecte	43
3.7.1	Présentation	43
3.7.2	Format des messages échangés	43
3.8	Principes généraux	44
3.8.1	Authentification	44
3.8.2	Headers	45
3.8.3	Transmission des informations de connexion à l'API	45
4	Annexe	46
4.1	Exemple d'appels aux webservices décrits	46
4.1.1	WS01 Récupérer les campagnes ouvertes	46
4.1.2	WS02 Récupérer un modèle métier	47
4.1.3	WS03 Valider un modèle métier	50
4.1.4	WS04 Comparer deux modèles métier	53
4.1.5	WS05 Collecter des données cle/valeur	57

4.1.6	WS07 Collecter des données de niveau individuel	59
4.1.7	WS06 Consulter l'état d'une collecte	61
4.1.8	WS08 Modifier le statut de la collecte.....	62
4.2	Exemples de modèle métier	62
4.2.1	Modèle métier d'une campagne de module APA-PCH au format JSON	62
4.2.2	Modèle métier d'une campagne Tarification (voir page suivante)	62
4.3	Modèle métier du service de collecte de données par API de niveau structure/dossier	62
4.4	Modèle métier du service de collecte de données par API de niveau individuel	63
4.5	Exemple de fichier FHIR	63
4.6	Exemples de documents CDA	63

1 INTRODUCTION

1.1 PÉRIMÈTRE DU DOCUMENT

Le Contrat d'Interface a pour objectif de décrire comment utiliser le composant en interfaçage.

Il décrit notamment :

- Les fonctionnalités proposées par chaque service de l'API de collecte
- L'orchestration des appels aux services de l'API de collecte
- Les champs et documents échangés et leur format

Ce document sera mis à jour pendant toute la durée du projet par les retours sur les mises en œuvre et les évolutions de périmètre.

1.2 OBJET DU DOCUMENT

Ce document est le Contrat d'Interface de l'API de collecte SIDOBA. Son objectif est de définir le contrat d'interface de SIDOBA avec les progiciels partenaires qui font de la collecte de données référencées dans le catalogue SIDOBA

Le Contrat d'Interface se décline en plusieurs volets :

- **La présentation des services échangés**
L'objectif de ce chapitre est de présenter la vision globale où s'inscrit cette interface. Il présente la vue globale, la cinématique et les services fonctionnels associés.
- **La description des interfaces**
L'objectif de ce chapitre est de détailler les services proposés, les champs et leur format.

1.3 PUBLIC CONCERNÉ

Les publics concernés par ce dossier sont les maitrises d'œuvre et d'ouvrage du projet, les architectes et développeurs, ainsi que les décideurs impliqués dans le projet et devant participer à la décision sur le scénario d'architecture globale de la solution cible à retenir, et/ou d'un premier palier de transition. Ce dossier s'adresse également aux maitrises d'œuvre et d'ouvrage consommateurs de ces interfaces.

1.4 DOCUMENTS DE RÉFÉRENCE

La liste ci-dessous référence de tous les documents cités dans ce contrat d'interface

DOCUMENTS DE REFERENCE

Date	Version	Rédacteurs	Objet	Emplacement dans l'archive	Abbréviation dans ce document
10/10/2023	1.0.0-ballot-STU1	Agence du Numérique en Santé (ANS)	Guide d'implémentation de la norme FHIR pour le transport de données DUI	\\Guide d'implémentation TDDUI\	R1
26/10/2023	1.0	Capgemini	Modèle métier API de collecte données de niveau structure	\\Modèle métier - Données Structures\IMPORT_CA_ONY_SSIAD_MM_API_POC_v1_0_20231026.json	R2
26/10/2023	1.0	Capgemini	Modèle métier API de collecte données de niveau individuel	\\Modèle métier - Données Individuelles\SSIAD_ONY_INDIV_MM_API_POC_var_man_v_1_0_20231026.json	R3
13/11/2023	N/A	Capgemini	Exemple de modèle métier APA PCH (hors POC)	\\Modèle métier - Exemple\modele_metier_apa_pch_20231311.json	R4
13/11/2023	N/A	Capgemini	Exemple de modèle métier tarification(hors POC)	\\Modèle métier - Exemple\modele_metier_tarification_20231311.json	R5
16/11/2023	1.0	Capgemini	Exemple de format FHIR	2023-11-16- _ESMSBundleResultatRechercheDecisionEvaluation_minimal_v1_0.json	R6

27/10/2023	1.0	Capgemini	Exemple de document CDA	20231027_TDDUI-ese_V1_1_bundle_document_1_v1_0.xml	R7
27/10/2023	1.0	Capgemini	Exemple de document CDA	20231027_TDDUI-ese_V1_1_bundle_document_2_v1_0.xml	R8
17/11/2023	1.0	Capgemini	Swagger décrivant les services de l'API	\Swagger - services API\swagger_api_collecte_structure_v_1_0_2023_11_17.yaml	R9
24/10/2024	1.0	Capgemini	MAJ swagger	api-docs_23102024.yaml	R10

2 PRÉSENTATION DES SERVICES ÉCHANGÉS

2.1 NIVEAU DES DONNEES ECHANGEES ENTRE LES PROGICIELS PARTENAIRES ET SIDOBA

2.1.1 STOCKAGE DES DONNÉES MÉTIER

La maille niveau des variables collectées par SIDOBA indique l'entité SIDOBA à laquelle sont rattachées les variables.

- Les **données de niveau structure** sont rattachées à des structures médico-sociales accueillant des usagers. Elles peuvent être :
 - Des données comptables
 - Des données sur les usagers qui passent dans les établissements
- Les **données de niveau dossier** sont rattachées à un ensemble de structures
- Les **données de niveau individuelles** sont rattachées aux usagers qui sont dans les structures médico-sociales. Ces données peuvent être :
 - Des données permettant d'identifier l'utilisateur (NIR, date de naissance ou autre trait d'identification)
 - Une évaluation GIR
 - Une date d'entrée ou de sortie dans un établissement médico-social

Toutes les variables stockées dans SIDOBA sont enregistrées au format clé/valeur.

Par exemple :

- Le nombre de fiches usager d'un établissement ESMS est une donnée structure.
 - Elle peut être stockée dans SIDOBA pour une structure de FINESS ET 350000001 sous la clé `CR__ACTI__FICHU__REAANN0` et peut porter la valeur 10.
- La commune de naissance est une donnée de niveau individuelle
 - Elle peut être stockée dans SIDOBA pour un usager de NIR 9102540100125 sous la clé `CRIN__IDEN__CPCOM__REAANN0` et peut porter la valeur 92035.

2.1.2 COLLECTE DES DONNÉES

Les utilisateurs qui collectent ces variables (= ils transmettent ces données pour le compte d'une structure à SIDOBA) sont habilités à faire cette action sur des modules donnés.

Ces modules caractérisent un métier (Tarification, SSIAD, etc.). Pour chaque module les périodes de collecte sont appelées campagnes. Un unique modèle métier applicable est associé à chaque campagne.

Ce modèle métier décrit :

- Les variables collectées
 - Leur libellé
 - Leur format
 - Etc.
- Les règles métier des variables
 - Valeurs numériques possibles
 - Enumération respectées par les variables
 - Etc.
- Les blocs répétables
 - Décrit des variables à occurrences multiples
 - Exemple : des périodes de passage pour un usager dans une ESMS

2.1.3 FORMAT DES MESSAGES APPELANT L'API DE COLLECTE

Le cahier des charges de l'API collecte de SIDOBA exprime les principes suivants de structure des messages envoyés :

- Les données de niveau structure et dossier sont transmises au format clé/valeur de SIDOBA
- Les données de niveau individuel sont transmises à SIDOBA
 - en conformité avec la norme FHIR pour le transport des messages
 - en conformité avec la norme CDA pour la structure des messages

Impact du principe au niveau structure/dossier : La connaissance du modèle métier par l'utilisateur souhaitant faire de la collecte est suffisant.

Impact du principe au niveau individuel : La connaissance des standards FHIR et CDA est nécessaire. Il n'est pas utile de récupérer le modèle métier.

2.2 VUE GLOBALE

Ce contrat d'interface décrit l'ouverture de la collecte SIDOBA par API.

8 services sont ouverts :

- Service de récupération des campagnes ouvertes
- Service de récupération d'un modèle métier
- Service de vérification de la validité d'un modèle métier
- Service de calcul de l'écart entre deux modèles métier
- Service de collecte de données clé/valeur:
 - Collecte de niveau structure/dossier
 - Collecte de niveau individuel
- Service de collecte de données FHIR/CDA de niveau individuel
- Service de récupération de l'état d'une collecte par API
- Service de mise à jour du statut de collecte

2.2.1 ORCHESTRATION DES APPELS AUX SERVICES DE COLLECTE DE DONNÉES DE NIVEAU STRUCTURE ET DOSSIER

Le diagramme suivant décrit la séquence de services à mettre en œuvre pour qu'un utilisateur dispose des données prérequis pour une collecte niveau structure/dossier sur une campagne cible. Chaque service (bloc vert) présenté dans le diagramme ci-dessous est détaillé dans [Description des interfaces](#).

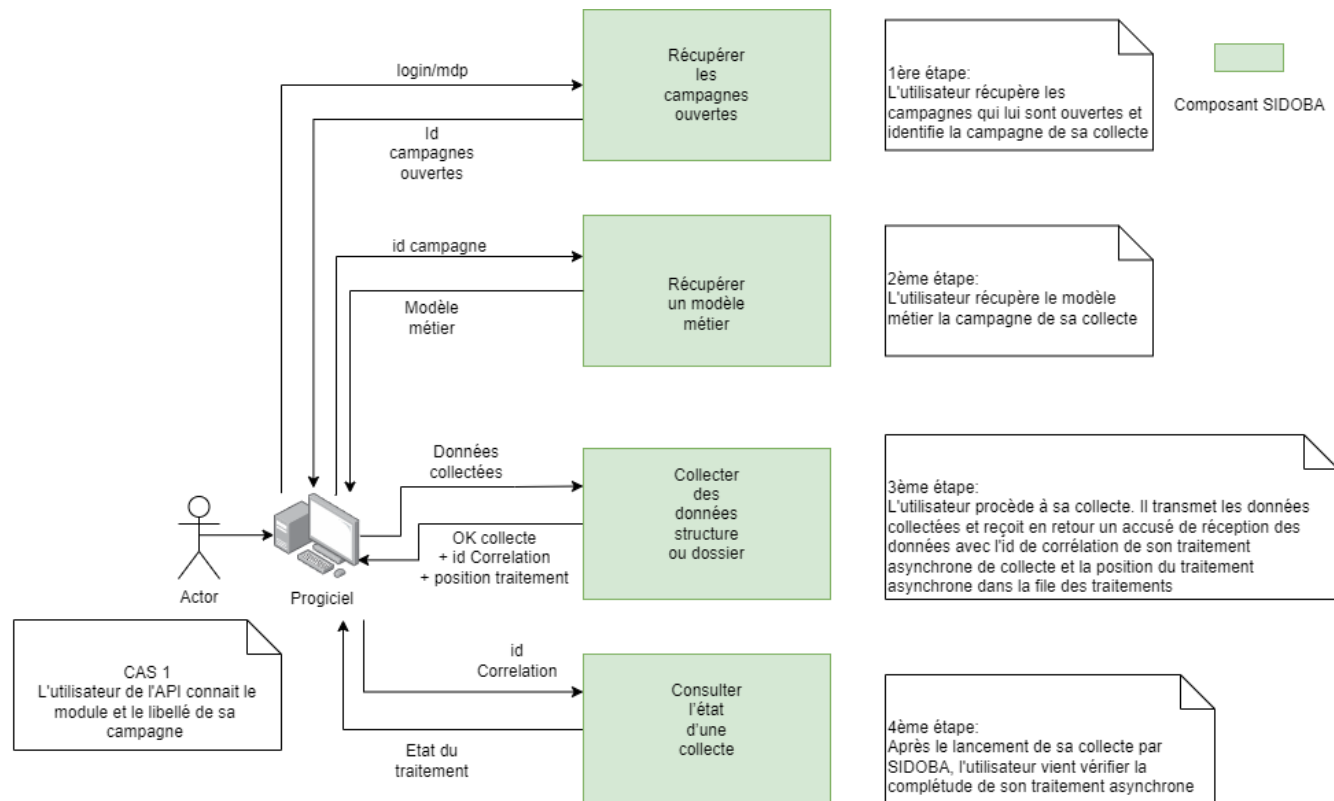


Figure 1 : Orchestration des appels aux services de l'API de collecte niveau structure et dossier dans le cas de la première collecte sur une campagne

Nous identifions deux cas d'utilisation.

- **1^{er} cas d'utilisation** : L'utilisateur de l'API fait une première collecte sur la campagne cible
 - L'utilisateur n'a donc pas de modèle métier paramétré sur son progiciel
 - L'exécution de la deuxième étape lui permet de récupérer le modèle métier à implémenter puis de faire sa collecte
 - Description des étapes d'appel aux services API
 - 1ère étape: L'utilisateur récupère les campagnes qui lui sont ouvertes et identifie la campagne de sa collecte
 - 2ème étape: L'utilisateur récupère le modèle métier de la campagne de sa collecte
 - 3ème étape: L'utilisateur procède à sa collecte. Il transmet les données collectées et reçoit en retour
 - un accusé de réception des données
 - l'id de son traitement asynchrone de collecte
 - la position du traitement asynchrone dans la file des traitements asynchrones
 - 4ème étape: Après le lancement de sa collecte par SIDOBA, l'utilisateur vient vérifier la complétude de son traitement asynchrone

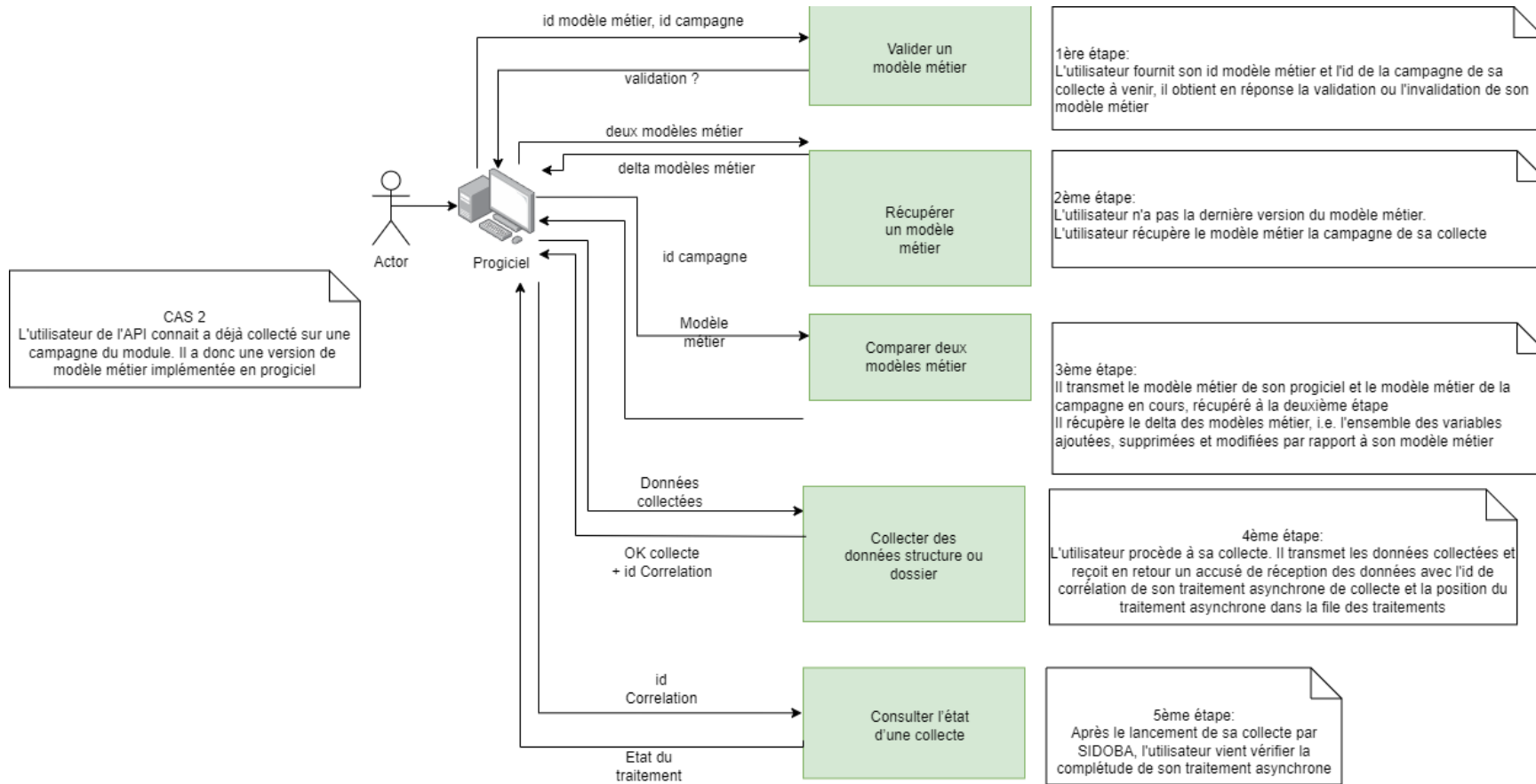


Figure 2 : Orchestration des appels aux services de l'API de collecte structure et- dossier dans le cas d'un complément ou d'une correction

- **2ème cas d'utilisation :** L'utilisateur de l'API a déjà collecté sur une campagne du module. Il a donc une version de modèle métier implémentée en progiciel
 - L'orchestration des appels aux services de l'API permet de valider que le modèle métier implémenté sur le progiciel est la version valide du modèle métier pour la campagne de collecte

- Dans le cas où des variations des variables du modèle métier sont identifiées, alors le progiciel implémente les variables ajoutées et modifiées. Il supprime les variables non utilisées dans son modèle métier
- Description des étapes d'appel aux services API
 - 1ère étape: L'utilisateur valide que le modèle métier implémenté par le progiciel partenaire est le modèle métier actif de la campagne de collecte cible
 - 2ème étape: Dans le cas de variations de modèles métier, l'utilisateur récupère le modèle métier la campagne de sa collecte
 - 3ère étape: L'utilisateur compare le modèle métier de son progiciel au modèle métier au modèle métier de la campagne ouverte.
 - 4ème étape: L'utilisateur procède à sa collecte. Il transmet les données collectée et reçoit en retour
 - un accusé de réception des données
 - l'id de son traitement asynchrone de collecte
 - la position du traitement asynchrone dans la file des traitements asynchrones
 - 5ème étape: Après le lancement de sa collecte par SIDOBA, l'utilisateur vient vérifier la complétude de son traitement asynchrone

La présentation ci-dessus en deux cas d'utilisation a pour objectif d'expliquer la cohérence des services sur l'ensemble du cycle de collecte d'un progiciel partenaire.

Il faut noter qu'il n'est pas nécessaire de parcourir toutes les étapes d'un cas d'utilisation pour mener une collecte.

Par exemple, un utilisateur qui a fait une collecte sur une campagne ouverte peut compléter les données collectées en appelant directement le service de collecte.

La cohérence du cas 2 se justifie à travers deux parcours différents

- Premier parcours : si l'utilisateur a déjà fait une collecte sur la campagne de collecte cible
 - il exécute la première étape du cas 2 pour valider que son modèle métier est toujours correct
 - Dans le cas où le modèle métier a évolué, il exécute les deuxième et troisième étapes pour récupérer le modèle métier courant et identifier les variations de modèle métier
 - Sinon, le modèle métier est toujours valide et il poursuit le processus en partant de la quatrième étape de collecte des données structure et dossier
- Deuxième parcours : l'utilisateur collecte sur une campagne différente du même module

Il poursuit le process à l'exécution des deuxièmes et troisièmes étapes pour récupérer le modèle métier en vigueur et identifier les variations

2.2.2 ORCHESTRATION DES APPELS AUX SERVICES DE COLLECTE DE DONNÉES DE NIVEAU INDIVIDUEL

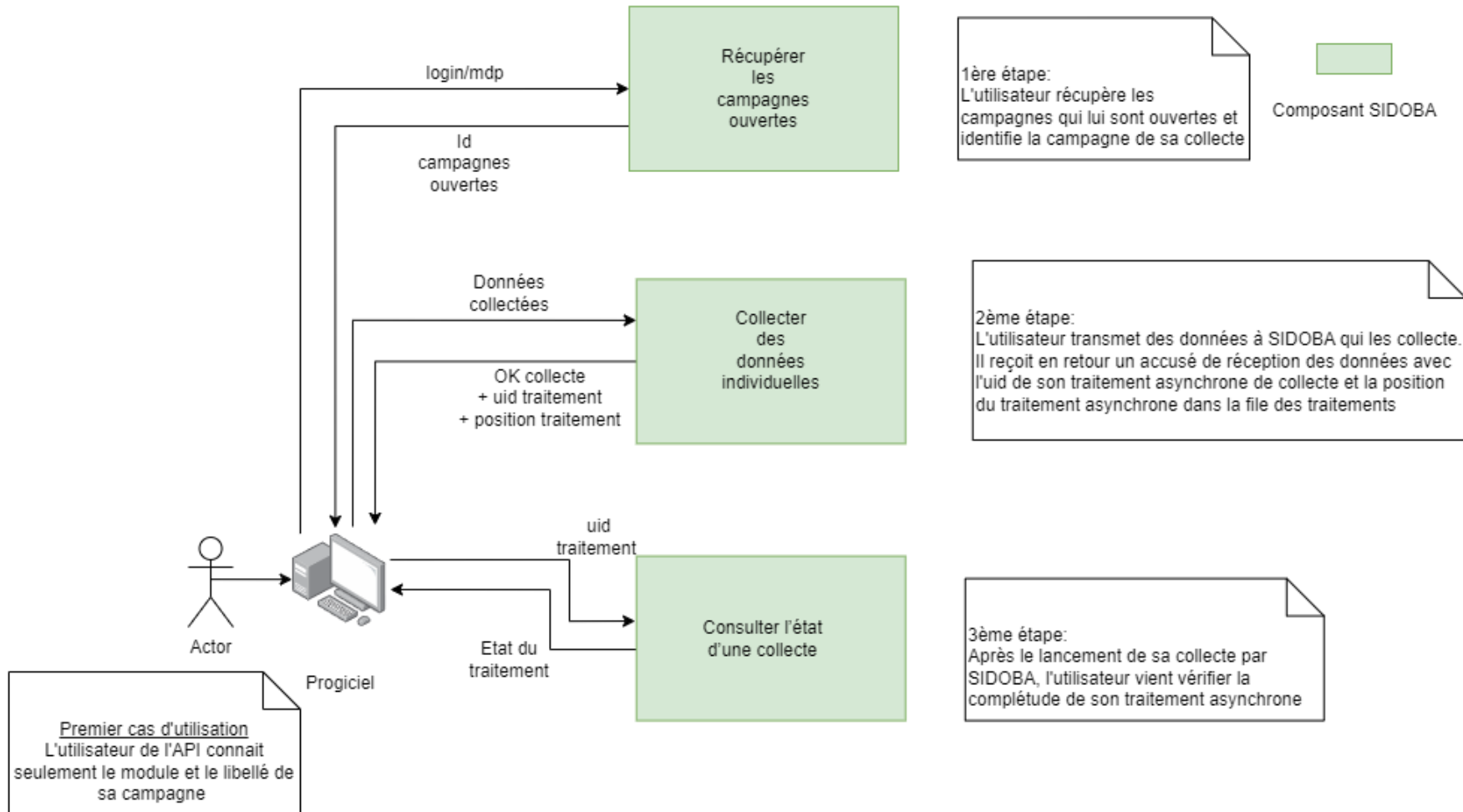


Figure 3: Diagramme logique de l'orchestration des services de l'API de collecte données individuelles

- **1^{er} cas d'utilisation** : L'utilisateur de l'API fait une première collecte sur la campagne cible
 - L'utilisateur n'a donc pas de modèle métier paramétré sur son progiciel
 - L'exécution de la deuxième étape lui permet de récupérer le modèle métier à implémenter puis de faire sa collecte
 - Description des étapes d'appel aux services API
 - 1ère étape : L'utilisateur récupère les campagnes qui lui sont ouvertes et identifie la campagne de sa collecte
 - 2ème étape : L'utilisateur procède à sa collecte. Il transmet les données collectées et reçoit en retour
 - un accusé de réception des données
 - l'id de son traitement asynchrone de collecte
 - la position du traitement asynchrone dans la file des traitements asynchrones
 - 3ème étape : Après le lancement de sa collecte par SIDOBA, l'utilisateur vient vérifier la complétude de son traitement asynchrone

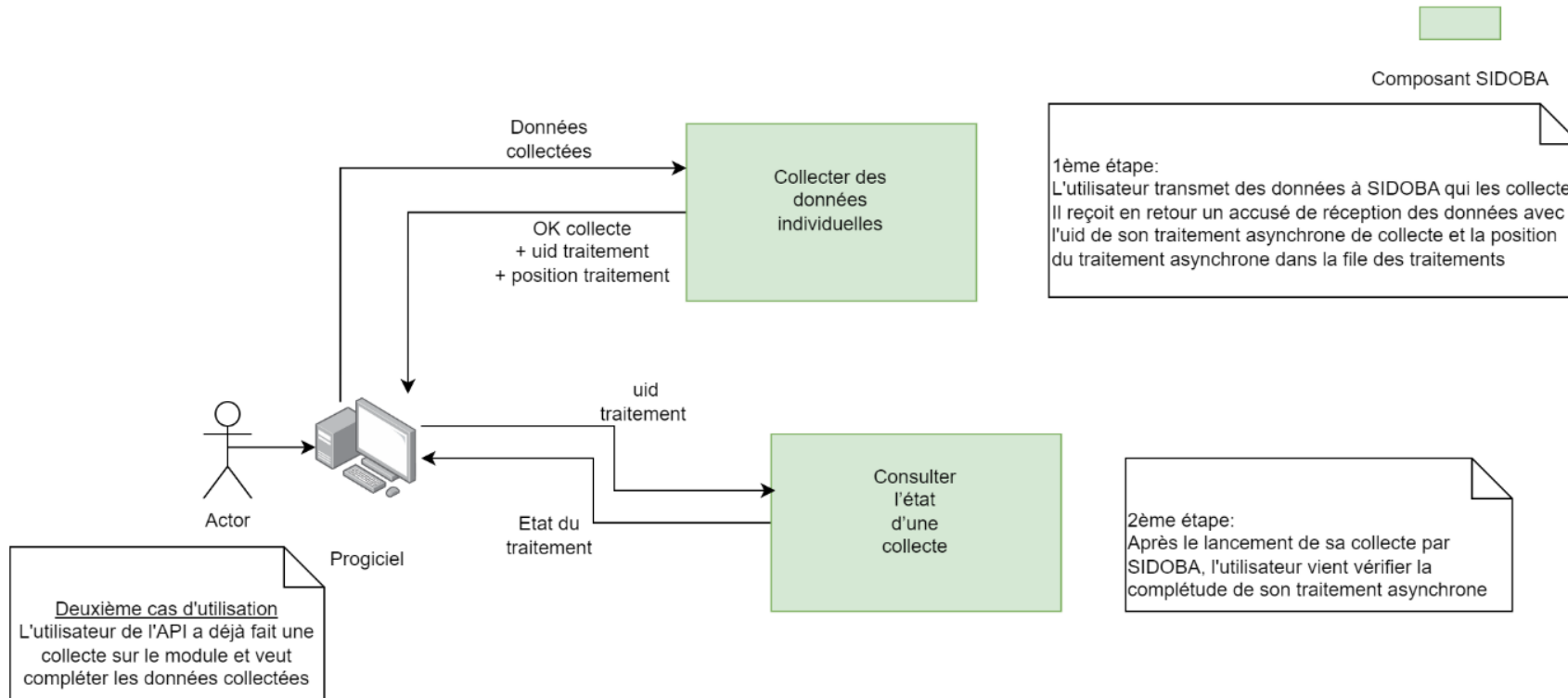


Figure 4: Orchestration des services de l'API de collecte de données individuelles dans le cas d'une correction ou d'un complément sur la campagne

■
2^{ème} cas d'utilisation : L'utilisateur de l'API a déjà collecté sur une campagne du module. Il connaît donc l'identifiant de la campagne cible

- L'orchestration des appels aux services de l'API permet de valider que le modèle métier implémenté sur le progiciel est la version valide du modèle métier pour la campagne de collecte
- Dans le cas où des variations des variables du modèle métier sont identifiées, alors le progiciel implémente les variables ajoutées et modifiées. Il supprime les variables non utilisées dans son modèle métier

- Description des étapes d'appel aux services API
 - 1ère étape : L'utilisateur procède à sa collecte. Il transmet les données collectées et reçoit en retour
 - un accusé de réception des données
 - l'id de son traitement asynchrone de collecte
 - la position du traitement asynchrone dans la file des traitements asynchrones
 - 2ème étape : Après le lancement de sa collecte par SIDOBA, l'utilisateur vient vérifier la complétude de son traitement asynchrone

La présentation ci-dessus a pour objectif d'expliquer la cohérence des services sur l'ensemble du cycle de collecte d'un progiciel partenaire.

Il faut noter qu'il n'est pas nécessaire de parcourir toutes les étapes d'un cas d'utilisation pour mener une collecte.

Par exemple, un utilisateur qui a fait une collecte sur une campagne ouverte peut compléter les données collectées en appelant directement le service de collecte.

2.3 SYNTHÈSE DES SERVICES CONSOMMÉS

Nom du service	Niveau de données collectées en cible	Description du service
WS01 Récupérer les campagnes ouvertes	Structure/dossier ou Individuel	SIDOBA donne accès aux identifiants des campagnes ouvertes aux progiciels partenaires via l'API.
WS02 Récupérer un modèle métier	Structure/dossier	SIDOBA met le modèle métier en vigueur d'une campagne à disposition des progiciels partenaires
WS03 Valider un modèle métier	Structure/dossier	SIDOBA permet à un utilisateur de vérifier que le modèle métier implémenté par le progiciel partenaire est le modèle métier en cours pour la campagne de collecte
WS04 Comparer deux modèles métier	Structure/dossier	Ce service permet à un utilisateur de récupérer le delta entre deux modèles métier (afin d'identifier les nouveautés depuis une version donnée)
WS05 Collecter des données clé/valeur	Structure/dossier Individuel	L'utilisateur appelle ce service pour collecter des données au format clé/valeur en cohérence avec un modèle métier donné
WS07 Collecter des données FHIR/CDA	Individuel	Ce service exposé par SIDOBA permet la collecte de données de niveau individuel.
WS06 Consulter l'état d'une collecte	Structure/dossier ou Individuel	Ce service permet de récupérer l'état de traitement d'une collecte en cours
WS08 Mise à jour du statut de collecte	Individuel	Ce service permet la mise à jour du statut du traitement de la collecte des données individuelles

3 DESCRIPTION DES INTERFACES

3.1 WS01 RECUPERER LES CAMPAGNES OUVERTES

3.1.1 PRÉSENTATION

SIDOBA donne accès aux identifiants des campagnes ouvertes aux progiciels partenaires via l'API.

- La fonctionnalité exposée par ce service sert deux prérequis
 - Identifier la campagne de collecte
 - Être en possession de l'id de la campagne de collecte

3.1.2 FORMAT DES MESSAGES ÉCHANGÉS

3.1.2.1 OPÉRATION RECUPERER CAMPAGNES OUVERTES

Nom du paramètre	Sens	Type	Multiplicité
N/A (méthode GET)	Entrée	N/A (méthode GET)	N/A (méthode GET)
CampagnesOuvertes	Sortie (en cas code retour « 200 »)	CampagnesOuvertes	[1...1]
Erreur	Sortie (en cas code retour différent de « 200 »)	Erreur	[1...1]

3.1.2.1.1 CampagnesOuvertes

Libellé	Description	Format	Exemple
Campagne	Liste des campagnes ouvertes à l'utilisateur appelant	[Libelle, id]	[0..n]

3.1.2.1.2 Détail de la liste CampagneOuvertes

Libellé	Description	Format	Multiplicité	Exemple
libelle	Libellé de la campagne	string	[1...1]	campagne_API_spec_6275
id	Id technique SIDOBA de la campagne	string	[1...1]	a635d5f1-3dfd0-45fa-8f26-55d8d1a44876

3.1.2.1.3 Erreur

Libellé	Description	Format	Multiplicité	Exemple
code	Code réponse technique de l'API SIDOBA	Int32	[1...1]	400

type	Libellé de l'erreur	string	[1...1]	"Missing request header 'X-Request-progiciel' for method parameter of type String"
-------------	---------------------	--------	---------	--

3.1.2.2 CARACTÉRISTIQUES TECHNIQUES

Type	Description
Format	JSON
Méthode	GET
Authentification	voir le chapitre dédié à l'authentification
Headers (détail dans les exemples d'appel)	Voir le chapitre dédié à la construction des headers

3.2 WS02 RECUPERER UN MODELE METIER

3.2.1 PRÉSENTATION

SIDOBA met le modèle métier en vigueur d'une campagne à disposition des progiciels partenaires.

3.2.2 FORMAT DES MESSAGES ÉCHANGÉS

3.2.2.1 OPERATION RECUPERERMODELEMETIER

Nom du paramètre	Sens	Type	Multipl licité
N/A (méthode GET)	Entrée	N/A (méthode GET)	N/A (métho de GET)
ModeleMetier	Sortie (en cas code retour « 200 »)	ModeleMetier	[1...1]
Erreur	Sortie (en cas code retour différent de « 200 »)	Erreur	[1...1]

3.2.2.2 MODELEMETIER

Libellé	Description	Format	Multipl licité	Exemple
idModele	Id technique SIDOBA du modèle métier	string	[1...1]	e1ceaeae- 5930-11ee- 8c99- 0242ac120002
corpsModele	Forme textuelle (JSON) du modèle métier	string	[1...1]	voir Exemple de modèle métier

3.2.2.3 ERREUR

Libellé	Description	Format	Multipl licité	Exemple
code	Code réponse technique de l'API SIDOBA	Int32	[1...1]	400

3.2.2.4 CARACTERISTIQUES TECHNIQUES

Type	Description
Format	JSON
Méthode	GET
Authentification	Voir le chapitre dédié à l'authentification
Headers (détail dans les exemples d'appel)	Voir le chapitre dédié à la construction des headers

3.3

3.3 WS03 VALIDER UN MODELE METIER

3.3.1 PRÉSENTATION

SIDOBA permet à un utilisateur de vérifier que le modèle métier implémenté par le progiciel partenaire est le modèle métier en cours pour la campagne de collecte

3.3.2 FORMAT DES MESSAGES ÉCHANGÉS

3.3.2.1 OPERATION VALIDERMODELEMETIER

Nom du paramètre	Sens	Type	Multiplicité
ModeleMetier	Entrée	ModeleMetier	[1...1]
validerModeleMetierReponse	Sortie	validerModeleMetierReponse	[1...1]
Erreur	Sortie (en cas code retour différent de « 200 »)	Erreur	[1...1]

3.3.2.2 MODELEMETIER

Libellé	Description	Format	Multiplicité	Exemple
idModele	Id technique SIDOBA du modèle métier	string	[1...1]	e1ceaeae-5930-11ee-8c99-0242ac120002
corpsModele	Forme textuelle (JSON) du modèle métier	string	[1...1]	voir Exemple de modèle métier

3.3.2.3 VALIDERMODELEMETIERREPONSE

Libellé	Description	Format	Multiplicité	Exemple
validation	OK si le modèle métier est la version en cours pour la campagne (=dernière version du modèle métier) KO sinon	string	[1...1]	
message	message associé à la validation « modèle métier à jour » si validation = « OK » « modèle métier obsolète » si validation = « KO »	string	[1...1]	

3.3.2.4 ERREUR

Libellé	Description	Format	Multiplicité	Exemple
code	Code réponse technique de l'API SIDOBA	Int32	[1...1]	400

3.3.2.5 CARACTERISTIQUES TECHNIQUES

Type	Description
Format	JSON
Méthode	POST
Authentification	voir le chapitre dédié à l'authentification
Headers (détail dans les exemples d'appel)	Voir le chapitre dédié à la construction des headers

3.4 WS04 COMPARER DEUX MODÈLES MÉTIER

3.4.1 PRÉSENTATION

Ce service permet à un utilisateur de récupérer le delta entre deux modèles métier (afin d'identifier les nouveautés depuis une version donnée)

Ce delta est la différence des variables référencées entre les deux modèles métier

3.4.2 FORMAT DES MESSAGES ÉCHANGÉS

3.4.2.1 OPERATION COMPARERMODELESMETIER

Nom du paramètre	Sens	Type	Multiplicité
ModeleMetier	Entrée	ModeleMetier	[1...1]
DeltaModeleMetierReponse	Sortie	DeltaModeleMetierReponse	[1...1]

3.4.2.2 MODELEMETIER

Libellé	Description	Format	Multiplicité	Exemple
idModele	Id technique SIDOBA du modèle métier	string	[1...1]	e1ceaeae-5930-11ee-8c99-0242ac120002
corpsModele	Forme textuelle (JSON) du modèle métier	string	[1...1]	voir Exemple de modèle métier

3.4.2.3 DELTAMODELEMETIERREPOSE

Libellé	Description	Format	Multiplicité	Exemple
ajout	Objet décrivant les variables ajoutées	AjoutVariables	[1...1]	
modification	Objet décrivant les variables modifiées	ModificationVariables	[1...1]	
suppression	Objet décrivant les variables supprimées	SuppressionVariables	[1...1]	

3.4.2.3.1 AjoutVariables

Libellé	Description	Format	Multiplicité	Exemple
variables	Liste des variables ajoutées	Variable	[1...n]	voir

3.4.2.3.2 ModificationVariables

Libellé	Description	Format	Multiplicité

variables	Liste des variables ajoutées	Variable	[1...n]
------------------	------------------------------	----------	---------

3.4.2.3.3 ModificationVariables

Libellé	Description	Format	Multiplicité
variables	Liste des variables ajoutées	VariableSupprime	[1...n]

3.4.2.3.4 Variable

Libellé	Description	Format	Multiplicité	Exemple
nom	Nom de la variable	string	[1...1]	Places autorisées au 31/12/N-1
libelle	Libellé de la variable	string		CRRACASTAT__DOMREGAOREA ANNO
type	type de la variable	string		M
taille	nombre de caractères de la variable	string		2
valeurs	valeurs acceptées de la variable	string		{1,2,3,4}

3.4.2.3.5 VariableSupprime

Libellé	Description	Format	Multiplicité	Exemple
nom	Nom de la variable supprimée	string	[1...1]	CRRACASTAT__DOMREGAOREAAN NO

3.4.2.4 CARACTERISTIQUES TECHNIQUES

Type	Description
Format	JSON
Méthode	POST
Authentification	voir le chapitre dédié à l'authentification
Habilitations	Voir le chapitre dédié à la construction des headers

3.5 WS05 COLLECTER DES DONNEES CLE/VALEUR

3.5.1 PRÉSENTATION

L'utilisateur appelle ce service pour collecter des données au format clé/valeur en cohérence avec un modèle métier donné pour des données de niveau structure/dossier ou individuel

3.5.2 FORMAT DES MESSAGES ÉCHANGÉS

3.5.2.1 OPÉRATION COLLECTERDONNEESSTRUCTURE

Nom du paramètre	Sens	Type	Multiplicité	Exemple
Donnees	Entrée	Donnees	[1...1]	voir détail dans ce chapitre
CollecterDonneesReponse	Sortie	CollecterDonneesReponse	[1...1]	voir détail dans ce chapitre

3.5.2.2 DONNEES

Libellé	Description	Format	Multiplicité	Exemple
Donnee	Données brutes collectées à transmettre à SIDOBA	Donnee	[1...n]	

3.5.2.2.1 Donnee

Libellé	Description	Format	Multiplicité	Exemple
cle	clé CNSA de la variable collectée	string	[1...1]	CRHAPATARF_PSPLAUT__RRDANM1
valeur	Valeur de la variable collectée	string	[1...1]	12
proprietes	Propriétés de la variable collectée	Liste d'objets	[1...N]	"proprietes": [{ "cle": "increment", "valeur": "0001" }]

3.5.2.3 COLLECTERDONNEESREPOSE

Libellé	Description	Format	Multiplicité	Exemple
positionTraitement	Position du traitement de collecte asynchrone déclenché dans SIDOBA	int	[1...1]	10
idTraitement	id du traitement asynchrone déclenché dans SIDOBA	string	[1...1]	44939

3.5.2.4 CARACTERISTIQUES TECHNIQUES

Type	Description
Format	JSON
Méthode	POST
Authentification	voir le chapitre dédié à l'authentification
Habilitations	Voir le chapitre dédié à la construction des headers

3.6 WS07 COLLECTER DES DONNEES DE NIVEAU INDIVIDUEL

3.6.1 PRESENTATION

Ce service exposé par SIDOBA permet la collecte de données de niveau individuel.

Pour exprimer l'objectif de ce service dans un lexique de la sphère sociale, ce service permet à un logiciel DUI de transmettre un ensemble de documents DUI à destination de SIDOBA.

SIDOBA collecte les données contenues dans les documents DUI.

La norme FHIR R4 régit le transport des messages traités par ce service.

La norme CDA R2 régit la structure des messages traités par ce service.

3.6.1.1 TRANSPORT DU MESSAGE D'APPEL A L'API DE COLLECTE DES DONNEES DE NIVEAU INDIVIDUEL DE SIDOBA

Le transport des messages des appels à l'API de collecte de données de niveau individuel de SIDOBA est conforme au standard FHIR en version R4.

Les éditeurs de logiciel implémentent cet appel à l'API selon le guide d'implémentation [R1].

L'API SIDOBA de collecte des données de niveau individuel implémente le cas d'utilisation d'ajout d'un lot de documents DUI est le cas d'utilisation décrit par le chapitre 4.1 **Synthèse des flux du document** [R1]. Les flux de ce cas d'utilisation sont les suivants

- Le **flux 2.1. Ajout LotDocuments** pour l'appel au service de collecte de données de niveau individuel
 - Il s'agit d'un flux HTTP POST vers l'API de collecte SIDOBA
- Le **flux 2.2 ResultatAjoutLotDocuments** pour la réponse du service de collecte de données de niveau individuel

Le corps du message entrant de l'appel à l'API emploie les ressources suivantes décrites dans le chapitre 8 **Artifacts Summary** du document [R1]

- La ressource **DocumentReference** qui contient
 - des métadonnées
 - les documents DUI encodés en base 64
- La ressource **Bundle** contient
 - des métadonnées
 - un ensemble de DocumentReference

3.6.1.2 STRUCTURE DES MESSAGES AU FORMAT CDA

La structure des documents DUI transportés dans l'enveloppe FHIR est conforme au standard CDA R2.

Le document DUI est un message XML encodé dans l'enveloppe FHIR.

Le document [R1] décrit les ressources CDA formant le document DUI dans le chapitre 5.1 **Ressources CDA**.

3.6.2 FORMAT DES MESSAGES ÉCHANGÉS

Ce document décrit le format minimal de message accepté par l'API de collecte des données individuelles SIDOBA. Le seul champ « attachement », décrit dans [Attachement](#) porte de l'information à valeur métier pour SIDOBA. Il est surligné en jaune . Ce champ porte le document CDA traité par l'API SIDOBA en base 64.

3.6.2.1 OPERATION COLLECTERDONNEESINDIVIDUELLES

Nom du paramètre	Sens	Type	Multiplicité
FormatageFHIR	Entrée	Bundle	[1..1]
CollecterDonneesIndividuellesResponse	Sortie avec succès	Bundle	[1..1]
CollecterDonneesIndividuellesErreur	Sortie en erreur	OperationOutcome	[1..1]

3.6.2.2 OPERATIONOUTCOME

Libellé	Description	Format	Multiplicité	Exemple
resourceType	Valeur fixe : type d'objet portant les lots de documents DUI selon les standards ANS pour le transport de données DUI	string	[1..1]	Valeur fixe « OperationOutcome »
id	Valeur fixe : id de l'erreur	string	[1..1]	Valeur fixe : «101»
issue	description de l'erreur rencontrée	Issue	[1..n]	voir objet Issue décrit dans ce chapitre

3.6.2.2.1 Issue

Libellé	Description	Format	Multiplicité	Exemple
severity	Valeur fixe : Gravité du problème	string	[1..1]	Valeur fixe «error»
Code	Valeur fixe : code de l'erreur	string	[1..1]	«bad request»

3.6.2.3 BUNDLE

Libellé	Description	Format	Multiplicité	Exemple
resourceType	Valeur fixe : type d'objet portant les lots de documents DUI selon les standards ANS pour le transport de données DUI	string	[1..1]	Valeur fixe « Bundle »
meta	Métadonnées permettant d'associer les données collectées à des concepts métier clairs	string	[1..1]	voir description de l'objet Meta plus bas
type	Valeur fixe	string	[1..1]	« transaction-response »
entry	La liste des objets DocumentReference portant les documents DUI	Entry	[1..n]	voir objet Entry décrit dans ce chapitre

3.6.2.3.1 Meta

Libellé	Description	Format	Multiplicité	Exemple
versionId	L'id de la campagne récupéré en réponse du service WS01	string	[1..1]	«b60ab9de-cf8f-4b66-bdce-9748a9f19184»

3.6.2.3.2 Entry

Libellé	Description	Format	Multiplicité	Exemple
resource	Le document instancié de l'objet DocumentReference	Document Reference	[1..1]	« d6eef477-70a6-4570-a7c5-54d45b717286 »

3.6.2.3.3 DocumentReference

Libellé	Description	Format	Multiplicité	Exemple
resourceType	Valeur fixe : type d'objet portant les données DUI selon les standards ANS	string	[1..1]	Valeur fixe « DocumentReference »
status	Valeur fixe Décrit la validité du document	string	[1..1]	Valeur fixe « current »
content	Documents contenus	Content	[1..n]	Voir objet Content dans ce chapitre

3.6.2.3.4 Content

Libellé	Description	Format	Multiplicité	Exemple
attachm ent	Document contenu	Attachem ent	[1..1]	voir objet Attachement décrit dans ce chapitre

3.6.2.3.5 Attachment

Libellé	Description	Format	Multiplicité	Exemple
context Type	Valeur fixe Le format du document	string	[1..1]	Valeur fixe « text/plain »
languag e	Valeur fixe Langue du document	string	[1..1]	Valeur fixe « fr »
title	Titre du document	string	[1..1]	« Grille GIR de Ruth NESSI »
data	Le contenu du document CDA en base 64	string	[1..1]	« SGVsbG8gd29ybGQh » Voir le détail dans la description détaillée du chapitre Data

3.6.2.3.6 Data

Le champ Data contient le document CDA encodé en base 64. Au sein de ce document CDA, celui-ci doit contenir des champs interprétés par SIDOBA qui sont matérialisés surlignés de jaune .

Ce document liste également les champs qui ne sont pas interprétés par SIDOBA qui sont sans surlignage.

La description des champs est conforme avec les directives de l'ANS détaillées dans le document [R1].

Libellé	Description	Format	Attribut du champ traité par SIDOBA	Multiplicité	Exemple
ClinicalDocument	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.realmCode	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.typeID	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.templateID	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.templateID	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.templateID	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.id	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.code	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.title	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.effectiveTime	Heure de l'émission de la requête	String Format W3C YYYYMMDDThhmmssTZD			20230824111700+0100
ClinicalDocument.confidentialityCode	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.languageCode	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés

ClinicalDocument.recordTarget	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.author	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.custodian	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.legalAuthenticator	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.documentationOf	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.componentOf	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structureBody	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structureBody.component	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structureBody.component.section FR-Evaluation-dustatut-fonctionnel	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structureBody.component	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structureBody.component.section FR-Documentssajoutes	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés

ClinicalDocument.recordTarget					Voir fichiers exemple CDA attachés
ClinicalDocument.recordTarget.patientRole					Voir fichiers exemple CDA attachés
ClinicalDocument.recordTarget.patientRole.id	Identifiant de l'utilisateur pour l'entité envoyant les données. Il sera la clé technique d'identification de l'utilisateur dans SIDOBA	string	extension	[1..1]	"89044"
ClinicalDocument.recordTarget.patientRole.patient.name.	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.recordTarget.patientRole.patient.name.family	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.recordTarget.patientRole.patient.name.given	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.recordTarget.patientRole.patient.name.family	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.recordTarget.patientRole.patient.name.given	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.recordTarget.patientRole.patient.administrativeGenderCode	Sexe de l'utilisateur	string	code	[1..1]	« F »
ClinicalDocument.recordTarget.patientRole.patient.birthTime	Date et heure de naissance	string	texte du champ	[1..1]	« 19770714170555+01:00 »
ClinicalDocument.recordTarget.patientRole.patient.birthplace.sdct:multipleBirthOrderNumber	Rang de naissance	string	value	[1..1]	« 42 »

ClinicalDocument.recordTarget.patientRole.patient.birthplace.place	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.recordTarget.patientRole.patient.birthplace.place.name	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.recordTarget.patientRole.patient.birthplace.place.address	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.recordTarget.patientRole.patient.birthplace.place.address.county	Lieu de naissance format court	string	texte du champ	[1..1]	« 75035 »
ClinicalDocument.author.time	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.author.assignedAuthor.id	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.author.assignedAuthor.assignedAuthor	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.author.assignedAuthor.assignedAuthor.softwareName	Nom du progiciel utilisé	string	texte du champ	[1..1]	“PROG1”
ClinicalDocument.custodian.assignedCustodian	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.custodian.assignedCustodian.representedCustodianOrganization	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.custodian.assignedCustodian.representedCustodianOrganization.id	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.legalAuthenticator.time	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.legalAuthenticator.signatureCode	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.legalAuthenticator.assignedEntity	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.legalAuthenticator.assignedEntity.id	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés

ClinicalDocument.documentationOf.serviceEvent	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.documentationOf.serviceEvent.effectiveTime	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.documentationOf.serviceEvent.performer	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.documentationOf.serviceEvent.performer.assignedEntity	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.documentationOf.serviceEvent.performer.assignedEntity.id	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.documentationOf.serviceEvent.performer.assignedEntity.code	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.documentationOf.serviceEvent.performer.assignedEntity.representedOrganization	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.componentOf.encompassingEncounter	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.componentOf.encompassingEncounter	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.componentOf.encompassingEncounter.effectiveTime.low	Date de début de prise en charge	string	value	[1..1]	« 20200318115500+0100 »
ClinicalDocument.componentOf.encompassingEncounter.effectiveTime.high	Date de début de prise en charge	string	value	[1..1]	« 20200325102200+0100 »
ClinicalDocument.componentOf.encompassingEncounter.location	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.componentOf.encompassingEncounter.location.healthCareFacility	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.componentOf.encompassingEncounter.location.healthCareFacility.code	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structuredBody.component.section	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés

ClinicalDocument.component.structuredBody.component.section.temp latelid	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structuredBody.component.section.temp latelid	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structuredBody.component.section.temp latelid	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structuredBody.component.section.temp latelid	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structuredBody.component.section.code	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structuredBody.component.section.title	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structuredBody.component.section.text	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structuredBody.component.section.com ponent	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structuredBody.component.section.com ponent.section	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structuredBody.component.section.com ponent.section.templateid	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structuredBody.component.section.com ponent.section.templateid	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structuredBody.component.section.com ponent.section.code	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structuredBody.component.section.com ponent.section.title	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structuredBody.component.section.com ponent.section.text	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structuredBody.component.section.com ponent.section.entry	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés

ClinicalDocument.component.structuredBody.component.section.component.section.entry.observation	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structuredBody.component.section.component.section.entry.observation.templateId	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structuredBody.component.section.component.section.entry.observation.templateId	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structuredBody.component.section.component.section.entry.observation.templateId	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structuredBody.component.section.component.section.entry.observation.templateId	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structuredBody.component.section.component.section.entry.observation.id	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structuredBody.component.section.component.section.entry.observation.code	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structuredBody.component.section.component.section.entry.observation.id	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structuredBody.component.section.component.section.entry.observation.text	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structuredBody.component.section.component.section.entry.observation.text.reference	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structuredBody.component.section.component.section.entry.observation.statusCode	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structuredBody.component.section.component.section.entry.observation.effectiveTime	Date de l'évaluation	string	value	[1..1]	« 20200319091100+0100 »
ClinicalDocument.component.structuredBody.component.section.component.section.entry.observation.value	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structuredBody.component.section.component.section.entry.observation.value.originalText	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés

ClinicalDocument.component.structuredBody.component.section.component.section.entry.observation.value.originalText.reference	Si l'attribut <i>code</i> de ClinicalDocument.component.structuredBody.component.section.value a la valeur « MED-339 » Résultat (score) de l'évaluation	string	value	[1..1]	« 2 »
ClinicalDocument.component.structuredBody.component.section.component.section.entry.observation.entryRelationship	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structuredBody.component.section.component.section.entry.observation.entryRelationship.act	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structuredBody.component.section.component.section.entry.observation.entryRelationship.templateId	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structuredBody.component.section.component.section.entry.observation.entryRelationship.templateId	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structuredBody.component.section.component.section.entry.observation.entryRelationship.templateId	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structuredBody.component.section.component.section.entry.observation.entryRelationship.code	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structuredBody.component.section.component.section.entry.observation.entryRelationship.text	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés
ClinicalDocument.component.structuredBody.component.section.component.section.entry.observation.entryRelationship.reference	Défini dans le document [R1]	Défini dans le document [R1]	N/A	Défini dans le document [R1]	Voir fichiers exemple CDA attachés

3.6.2.4 CARACTERISTIQUES TECHNIQUES

Type	Description
Format	JSON
Méthode	POST
Authentification	voir le chapitre dédié à l'authentification
Habilitations	Voir le chapitre dédié à la construction des headers

Un objet Attachment d'une instance de la ressource FHIR DocumentReference contient le document DUI conforme à la norme CDA encodé en base 64.

Il s'agit du seul champ à valeur métier pour SIDOBA, i.e. SIDOBA ne stocke que les données qui sont dans ce champ.

3.7 WS06 CONSULTER L'ETAT D'UNE COLLECTE

3.7.1 PRÉSENTATION

Ce service permet de récupérer l'état de traitement d'une collecte en cours

3.7.2 FORMAT DES MESSAGES ECHANGES

3.7.2.1 OPERATION LIREETATCOLLECTE

Nom du paramètre	Sens	Type	Multipl icité
N/A (méthode GET)	Entrée	N/A (méthode GET)	N/A (métho de GET)
etatCollecteReponse	Sortie (en cas code retour « 200 »)	etatCollecteReponse	[1...1]
Erreur	Sortie (en cas code retour différent de « 200 »)	Erreur	[1...1]

3.7.2.2 ETATCOLLECTEREPOSE

Libellé	Description	Format	Multipl ité	Exemple
etat	id de la campagne	string	[1...1]	37c68394- 5865-11ee- 8c99- 0242ac120002
position	position dans la file des traitements asynchrones	int	[1...1]	10
heureFin	Heure de fin du traitement	string	[0...1]	2022-01- 19T16:17:18.7 34Z

3.7.2.3 CARACTERISTIQUES TECHNIQUES

Type	Description
Format	JSON
Méthode	GET
Authentification	voir le chapitre dédié à l'authentification
Headers (détail dans les exemples d'appel)	Voir le chapitre dédié à la construction des headers

3.8

3.8 PRINCIPES GENERAUX

3.8.1 AUTHENTIFICATION

L'utilisateur s'authentifie auprès de l'API via un token JWT.

Pour obtenir ce token JWT, l'utilisateur s'authentifie auprès des Services Centraux par certificat. Les SC retournent à l'utilisateur un token d'authentification, celui-ci est alors utilisé pour appeler les services de l'API.

3.8.2 HEADERS

Pour tous les services API mis à disposition les headers véhiculent la chaîne de caractère interprétée par la fonctionnalité de l'API de collecte SIDOBA

Cette chaîne de caractère prend la forme suivante : *Authorization: Bearer {donnees_authentification}* où *donnees_authentification* est le token JWT retourné par les SC.

De plus, pour les services GET, les headers véhiculent les métadonnées associées aux données récupérées par l'utilisateur

Le tableau ci-dessous liste les headers reconnus par l'API de collecte SIDOBA par service

Header	Services	Obligatoire	Description
<i>Authorization: Bearer {donnees_authentification}</i> :	WS01, WS02, WS03, WS04, WS05, WS06, WS07, WS08	oui	Token JWT
X-Request-progiciel	WS01, WS02, WS03, WS04, WS05, WS06, WS07, WS08	oui	Nom du progiciel Fixe à la Valeur «PROG1»
X-Request-campagne :	WS02, WS03, WS04, WS05, WS06, WS07	oui	id de la campagne cible que l'on récupère du WS01
X-Request-structure	WS07	oui	FINESS ET de la structure des données collectées
X-Request-niveauDonnee	WS05	oui	Niveau des données de la collecte (STRUCTURE, INDIVIDUEL)
X-Request-idIndividuStructure	WS05	non	Uuid de l'individu
X-Request- idTraitement :	WS06	oui	id Traitement fourni par la réponse des WS05, WS07

Les exemples d'appels des webservices du chapitre 4 décrivent également l'authentification par le header des requêtes d'appel aux services.

3.8.3 TRANSMISSION DES INFORMATIONS DE CONNEXION A L'API

Un Keypass est partagé avec les éditeurs ayant accès à l'API de collecte.

Il contient les données suivantes :

- URL des services

4 ANNEXE

4.1 EXEMPLE D'APPELS AUX WEBSERVICES DECRITS

Dans cette section, nous donnons un exemple d'appel cURL et de réponse de chaque webservice décrit.

Tous ces exemples mettent en jeu des données qui ne sont pas celles qui seront exposées par SIDOBA dans le cadre du POC API de collecte.

4.1.1 WS01 RÉCUPÉRER LES CAMPAGNES OUVERTES

Cet exemple illustre le cas fictif suivant :

- L'utilisateur envoie une requête via son progiciel en spécifiant dans le header
 - Son token d'authentification
 - Son code progiciel « PROG1 »
- SIDOBA renvoie l'unique campagne dans le périmètre de l'utilisateur nommée « *campagne_API_spec_6275* »

Message cURL d'entrée :

```
curl -X 'GET' \  
'https://flux.esms.sans.restrictions.api.sidoba.fr/api/v1/recupererCampagnesOuvertes' \  
-H 'Authroization: Bearer xxxxxxxxxxxxxxxx' \  
-H 'accept: application/json' \  
-H 'X-Request-progiciel: prog1'
```

Réponse de l'API :

```
[  
  {  
    "libelle": "campagne_API_spec_6275",  
    "id": "a635d5f1-3dfd0-45fa-8f26-55d8d1a44876"  
  }  
]
```

4.1.2 WS02 RÉCUPÉRER UN MODÈLE MÉTIER

L'exemple suivant décrit

- Un appel cURL au service de l'utilisateur souhaitant récupérer le modèle métier d'une campagne dont il connaît l'identifiant
- La réponse du service

Le cas présenté est le suivant

- L'utilisateur envoie une requête via son progiciel en spécifiant dans le header
 - Son token d'authentification
 - Son code progiciel «PRO1»
 - l'id campagne pour lequel il souhaite récupérer
- Le service répond en transmettant la dernière version de modèle métier de la campagne

Message cURL d'entrée :

```
curl -X 'GET' \  
'https://flux.esms.sans.restrictions.api.sidoba.fr/v1/recupererModeleMetier' \  
-H 'accept: application/json' \  
-H 'Authroization: Bearer xxxxxxxxxxxxxxxx' \  
-H 'X-Request-progiciel: PRO1' \  
-H 'X-Request-campagne: 54c595d6-585e-11ee-8c99-0242ac120002'
```

Réponse de l'API :

```
{  
  "idModele": "e1ceaeae-5930-11ee-8c99-0242ac120002",  
  "corpsModele": {  
    "nom": "6fcc32db-e0f8-4c30-9af8-e2eed5a972d5_modele_metier",  
    "version": "3.0",  
    "variables": [  
      {  
        "libelle": "Nombre d'enfants domiciliés dans une autre région",  
        "type": "N",  
        "nom": "CRRACASTAT__DOMREGA0REAANN0",  
        "taille": "2",  
        "valeurs": null  
      }  
    ],  
    "typesDonnees": [  
      {  
        "nom": "categorie1",  
        "obligatoire": true,  
        "numeroOrdre": 1,  
        "niveauDonnee": "DOSSIER",  
        "reglesMetier": [  
          {  
            "nomFormule": null,  
            "referenceFormule": null,  
            "numeroOrdre": 1,  
            "type": "CONTROLE",
```

```
"typeRegleCalcul": null,
"variableCible": "CRRACASTAT__DOMREGA0REAANN0",
"estBloquant": false,
"referenceControle": "RM_CONT_ENCAD_STRICT",
"nomControle": "Encadrement",
"listeParametresRegleMetier": [
  {
    "numeroOrdre": null,
    "typeParam": "CONSTANTE",
    "valeurParam": "co",
    "numeroListe": 1,
    "coordonneeParam": null,
    "niveauDonnee": null,
    "caracteristique": null
  },
  {
    "numeroOrdre": null,
    "typeParam": "CONSTANTE",
    "valeurParam": "constante",
    "numeroListe": 1,
    "coordonneeParam": null,
    "niveauDonnee": null,
    "caracteristique": null
  }
]
},
"referencesVariables": [
  {
    "nom": "CRRACASTAT__DOMREGA0REAANN0",
    "libelle": "Nombre d'enfants domiciliés dans une autre région",
    "numeroOrdre": null,
    "donneeSynthese": false,
    "cachee": false,
    "caracteristique": "NORMALE"
  }
]
},
"formulaires": [
  {
    "nom": "Form",
    "niveauDonnee": "DOSSIER",
    "rubriqueFormulaire": {
      "id": 803,
      "nom": "Form",
      "descriptif": null,
      "numOrdre": 1,
      "repetable": false,
      "niveauRubrique": 0,

```



```
"rubriqueFormulaireEnfants": [  
  {  
    "id": 804,  
    "nom": "Rubrique",  
    "descriptif": null,  
    "numOrdre": 1,  
    "repetable": false,  
    "niveauRubrique": 1,  
    "rubriqueFormulaireEnfants": [],  
    "champsFormulaire": [  
      {  
        "libelleChamp": "Champ1",  
        "descriptif": "Champ test",  
        "numOrdre": 1,  
        "nonModifiable": false,  
        "obligatoire": true,  
        "typeDonneeVariable": null,  
        "infosPresentation": {  
          "affichage": "Liste déroulante"  
        }  
      }  
    ],  
    "blocRepetableARubriqueId": null,  
    "infosPresentation": {  
      "disposition": "Horizontale"  
    }  
  },  
  {  
    "champsFormulaire": [],  
    "blocRepetableARubriqueId": null,  
    "infosPresentation": {  
      "disposition": "Verticale"  
    }  
  }  
],  
"blocsRepetables": []  
}
```

4.1.3 WS03 VALIDER UN MODÈLE MÉTIER

L'exemple décrit le cas d'utilisation suivant

- L'utilisateur envoie une requête via son progiciel en spécifiant dans le header
 - Son token d'authentification
 - Son code progiciel «PRO1»
 - l'id campagne pour lequel il souhaite valider
- L'utilisateur transmet dans le corps de la requête
 - Le modèle métier au format JSON qui est configuré dans son progiciel

Message cURL d'entrée :

```
curl -X 'POST' \  
'https://flux.esms.sans.restrictions.api.sidoba.fr/api/v1/validerModeleMetier' \  
-H 'accept: application/json' \  
-H 'Authorization: Bearer xxxxxxxxxxxxxxxx' \  
-H 'X-Request-progiciel: prog1' \  
-H 'X-Request-campagne: 54c595d6-585e-11ee-8c99-0242ac120002' \  
-H 'Content-Type: application/json' \  
-d '{  
  "idModele": "e1ceaeae-5930-11ee-8c99-0242ac120002",  
  "corpsModele": {  
    "nom": "6fcc32db-e0f8-4c30-9af8-e2eed5a972d5_modele_metier",  
    "version": "3.0",  
    "variables": [  
      {  
        "libelle": "Nombre d'enfants domiciliés dans une autre région",  
        "type": "N",  
        "nom": "CRRACASTAT__DOMREGA0REAANN0",  
        "taille": "2",  
        "valeurs": null  
      }  
    ],  
    "typesDonnees": [  
      {  
        "nom": "categorie1",  
        "obligatoire": true,  
        "numeroOrdre": 1,  
        "niveauDonnee": "DOSSIER",  
        "reglesMetier": [  
          {  
            "nomFormule": null,  
            "referenceFormule": null,  
            "numeroOrdre": 1,  
            "type": "CONTROLE",  
            "typeRegleCalcul": null,  
            "variableCible": "CRRACASTAT__DOMREGA0REAANN0",  
            "estBloquant": false,  
            "referenceControle": "RM_CONT_ENCAD_STRICT",
```

```
"nomControle": "Encadrement",
"listeParametresRegleMetier": [
  {
    "numeroOrdre": null,
    "typeParam": "CONSTANTE",
    "valeurParam": "co",
    "numeroListe": 1,
    "coordonneeParam": null,
    "niveauDonnee": null,
    "caracteristique": null
  },
  {
    "numeroOrdre": null,
    "typeParam": "CONSTANTE",
    "valeurParam": "constante",
    "numeroListe": 1,
    "coordonneeParam": null,
    "niveauDonnee": null,
    "caracteristique": null
  }
]
},
"referencesVariables": [
  {
    "nom": "CRRACASTAT___DOMREGA0REAANN0",
    "libelle": "Nombre d'\nenfants domiciliés dans une autre région",
    "numeroOrdre": null,
    "donneeSynthese": false,
    "cachee": false,
    "caracteristique": "NORMALE"
  }
]
},
"formulaires": [
  {
    "nom": "Form",
    "niveauDonnee": "DOSSIER",
    "rubriqueFormulaire": {
      "id": 803,
      "nom": "Form",
      "descriptif": null,
      "numOrdre": 1,
      "repetable": false,
      "niveauRubrique": 0,
      "rubriqueFormulaireEnfants": [
        {
          "id": 804,
          "nom": "Rubrique",
```

```
"descriptif": null,  
"numOrdre": 1,  
"repetable": false,  
"niveauRubrique": 1,  
"rubriqueFormulaireEnfants": [],  
"champsFormulaire": [  
  {  
    "libelleChamp": "Champ1",  
    "descriptif": "Champ test",  
    "numOrdre": 1,  
    "nonModifiable": false,  
    "obligatoire": true,  
    "typeDonneeVariable": null,  
    "infosPresentation": {  
      "affichage": "Liste déroulante"  
    }  
  }  
],  
"blocRepetableARubriqueId": null,  
"infosPresentation": {  
  "disposition": "Horizontale"  
}  
],  
"champsFormulaire": [],  
"blocRepetableARubriqueId": null,  
"infosPresentation": {  
  "disposition": "Verticale"  
}  
],  
"blocsRepetables": []  
}  
'
```

Réponse de l'API :

```
{  
  "validation": "OK",  
  "message": "Modèle métier à jour"  
}
```

4.1.4 WS04 COMPARER DEUX MODÈLES MÉTIER

L'exemple décrit le cas d'utilisation suivant

- L'utilisateur envoie une requête via son progiciel en spécifiant dans le header
 - Son token d'authentification
 - Son code progiciel «PRO1»
 - l'id campagne pour lequel il souhaite comparer
- L'utilisateur transmet dans le corps de la requête
 - Le modèle métier au format Json qui est configuré dans son progiciel
- Le service répond
 - En fournissant un code validant ou invalidant le fait que le modèle métier configuré est la dernière version du modèle métier

Message cURL d'entrée :

```
curl -X 'POST' \  
'https://flux.esms.sans.restrictions.api.sidoba.fr/api/v1/comparerModelesMetier' \  
-H 'accept: application/json' \  
-H 'Authroization: Bearer xxxxxxxxxxxxxx' \  
-H 'X-Request-progiciel: prog1' \  
-H 'X-Request-campagne: 54c595d6-585e-11ee-8c99-0242ac120002' \  
-H 'Content-Type: application/json' \  
-d '{  
  "idModele": "e1ceaeae-5930-11ee-8c99-0242ac120002",  
  "corpsModele": {  
    "nom": "6fcc32db-e0f8-4c30-9af8-e2eed5a972d5_modele_metier",  
    "version": "3.0",  
    "variables": [  
      {  
        "libelle": "Nombre d'\nenfants domiciliés dans une autre région",  
        "type": "N",  
        "nom": "CRRACASTAT__DOMREGA0REAANN0",  
        "taille": "2",  
        "valeurs": null  
      }  
    ],  
    "typesDonnees": [  
      {  
        "nom": "categorie1",  
        "obligatoire": true,  
        "numeroOrdre": 1,  
        "niveauDonnee": "DOSSIER",  
        "reglesMetier": [  
          {  
            "nomFormule": null,  
            "referenceFormule": null,  
            "numeroOrdre": 1,  
            "type": "CONTROLE",  
            "typeRegleCalcul": null,  
          }  
        ]  
      }  
    ]  
  }  
}
```

```
"variableCible": "CRRACASTAT__DOMREGA0REAANN0",
"estBloquant": false,
"referenceControle": "RM_CONT_ENCAD_STRICT",
"nomControle": "Encadrement",
"listeParametresRegleMetier": [
  {
    "numeroOrdre": null,
    "typeParam": "CONSTANTE",
    "valeurParam": "co",
    "numeroListe": 1,
    "coordonneeParam": null,
    "niveauDonnee": null,
    "caracteristique": null
  },
  {
    "numeroOrdre": null,
    "typeParam": "CONSTANTE",
    "valeurParam": "constante",
    "numeroListe": 1,
    "coordonneeParam": null,
    "niveauDonnee": null,
    "caracteristique": null
  }
]
},
"referencesVariables": [
  {
    "nom": "CRRACASTAT__DOMREGA0REAANN0",
    "libelle": "Nombre d'\enfant domiciliés dans une autre région",
    "numeroOrdre": null,
    "donneeSynthese": false,
    "cachee": false,
    "caracteristique": "NORMALE"
  }
]
},
"formulaires": [
  {
    "nom": "Form",
    "niveauDonnee": "DOSSIER",
    "rubriqueFormulaire": {
      "id": 803,
      "nom": "Form",
      "descriptif": null,
      "numOrdre": 1,
      "repetable": false,
      "niveauRubrique": 0,
      "rubriqueFormulaireEnfants": [
```

```

{
  "id": 804,
  "nom": "Rubrique",
  "descriptif": null,
  "numOrdre": 1,
  "repetable": false,
  "niveauRubrique": 1,
  "rubriqueFormulaireEnfants": [],
  "champsFormulaire": [
    {
      "libelleChamp": "Champ1",
      "descriptif": "Champ test",
      "numOrdre": 1,
      "nonModifiable": false,
      "obligatoire": true,
      "typeDonneeVariable": null,
      "infosPresentation": {
        "affichage": "Liste déroulante"
      }
    }
  ],
  "blocRepetableARubriqueId": null,
  "infosPresentation": {
    "disposition": "Horizontale"
  }
},
"champsFormulaire": [],
"blocRepetableARubriqueId": null,
"infosPresentation": {
  "disposition": "Verticale"
}
},
"blocsRepetables": []
}
}'

```

Réponse de l'API:

```

{
  "ajout": {
    "variables": [
      {
        "libelle": "Nombre d'enfants domiciliés dans une autre région",
        "typeVariable": "N",
        "nom": "CRRACASTAT__DOMREGA0REAANN0",
        "taille": "2",
        "valeurs": null
      }
    ]
  }
}

```

```
},  
"modification": {  
  "variables": [  
    {  
      "libelle": "Nombre d'enfants domiciliés dans une autre région",  
      "typeVariable": "N",  
      "nom": "CRRACASTAT__DOMREGA0REAANN0",  
      "taille": "2",  
      "valeurs": null  
    }  
  ]  
},  
"suppression": {  
  "variables": [  
    {  
      "nom": "CRRACASTAT__DOMREGA0REAANN0"  
    }  
  ]  
}  
}
```


4.1.5 WS05 COLLECTER DES DONNÉES CLE/VALEUR

L'exemple décrit le cas d'utilisation suivant

- L'utilisateur envoie une requête via son progiciel en spécifiant dans le header
 - Son token d'authentification
 - Son code progiciel «*PRO1*»
 - l'id campagne pour lequel il souhaite collecter
 - le niveau de données de collecte (STRUCTURE, INDIVIDUEL)
 - l'id de l'individu :
 - optionnel pour la collecte de niveau STRUCTURE,
 - obligatoire pour la collecte de niveau INDIVIDUEL
- L'utilisateur transmet dans le corps de la requête
 - Le Finess ET de sa structure «980502983»
 - Les données de collecte à destination de SIDOBA
- Le service répond
 - La position du traitement asynchrone de collecte dans la file des traitements asynchrones
 - L'id du traitement

Message cURL d'entrée :

```
curl --location 'http://localhost:8082/api/v1/donneesDeCollecte' \  
--header 'X-Request-campagne: 198d0297-e3e9-4490-b278-1abaf904b901' \  
--header 'X-Request-progiciel: progiciel' \  
--header 'X-Request-niveauDonnee: INDIVIDUEL' \  
--header 'X-Request-idIndividuStructure: 7bfe5d56-6ca7-41ff-989c-56a6849d0467' \  
--header 'Content-Type: application/json' \  
--header 'Authorization: Bearer xxxxxxxxxxxx' \  
--data '[  
  {  
    "structure": "980502983",  
    "donnees": [  
      {  
        "cle": "CR___IDEN___TEL___REAANNO",  
        "valeur": "0606060606"  
      },  
      {  
        "cle": "CR___IDEN___COMMU___REAANNO",  
        "valeur": "64000"  
      },  
      {  
        "cle": "CRINPHMEDE_FPCOHEREN_REAANNO",  
        "valeur": "Non",  
        "proprietes": [  
          {  
            "cle": "increment",  
            "valeur": "0001"  
          }  
        ]  
      }  
    ]  
  }  
'
```

Réponse de l'API :

```
{  
  "positionTraitement": 5,  
  "idTraitement": "44939"  
}
```

4.1.6 WS07 COLLECTER DES DONNÉES DE NIVEAU INDIVIDUEL

L'exemple décrit le cas d'utilisation suivant

- L'utilisateur envoie une requête via son progiciel en spécifiant dans le header
 - Son token d'authentification
 - Son code progiciel «PRO1»
 - l'id campagne pour lequel il souhaite collecter
 - l'id de la structure pour lequel il souhaite collecter
- L'utilisateur transmet dans le corps de la requête
 - Les données de collecte au format FHIR/CDA à destination de SIDOBA
- Le service répond
 - La position du traitement asynchrone de collecte dans la file des traitements asynchrones
 - En envoyant un code signifiant la réception des données de collecte

Message cURL d'entrée :

```
curl -X 'POST' \  
  'https://flux.esms.sans.restrictions.api.sidoba.fr/api/v1/donneesDeCollecteIndividuelles' \  
  -H 'accept: */*' \  
  -H 'Authorization: Bearer xxxxxxxxxxxx' \  
  -H 'X-Request-progiciel: prog1' \  
  -H 'X-Request-campagne: 54c595d6-585e-11ee-8c99-0242ac120002' \  
  -H 'X-Request-structure: 23117856' \  
  -H 'Content-Type: application/json' \  
  -d '{  
    "resourceType": "Bundle",  
    "type": "transaction-response",  
    "entry": [  
      {  
        "resourceType": "DocumentReference",  
        "status": "current",  
        "content": [  
          {  
            "attachment": {  
              "contentType": "text/plain",  
              "language": "fr",  
              "title": "Grille GIR de Ruth NESSI",  
              "data": "SGVsbG8gd29ybGQ=" }  
          }  
        ]  
      }  
    ]  
  }'  
'
```

Réponse de l'API :

Code 201 en synchronie

4.1.7 WS06 CONSULTER L'ÉTAT D'UNE COLLECTE

L'exemple suivant décrit

- Un appel cURL au service de l'utilisateur souhaitant récupérer l'état d'un traitement asynchrone de collecte dont il connaît l'iid
- La réponse du service

Le cas présenté est le suivant

- L'utilisateur envoie une requête via son progiciel en spécifiant dans le header
 - Son token d'authentification,
 - Son code progiciel «*PRO1*»
 - L'id du traitement asynchrone **retourné par le WS05 ou le WS07**
- Le service répond en transmettant
 - L'état du traitement asynchrone
 - La position du traitement asynchrone de collecte dans la file des traitements asynchrones ci celui-ci est en cours

Message cURL d'entrée :

```
curl -X 'GET' \  
'https://flux.esms.sans.restrictions.api.sidoba.fr/api/v1/lireEtatCollecte' \  
-H 'Authorization: Bearer xxxxxxxxxxxx' \  
-H 'X-Request-progiciel: prog1' \  
-H 'X-Request-idTraitement: 449391'"
```

Réponse du service:

```
{ "position": 10,  
  "etat": "EN COURS",  
  "heureFin": "2022-01-19T16:17:18.734Z"  
}
```

4.1.8 WS08 MODIFIER LE STATUT DE LA COLLECTE

L'exemple suivant décrit

- Un appel cURL au service de la moulinette API souhaitant modifier le statut d'une collecte de données individuelles
- La réponse du service

Le cas présenté est le suivant

- La moulinette API envoie une requête en spécifiant dans le header
 - Son token d'authentification,
 - Son code progiciel «Moulinette API»
- L'utilisateur transmet dans le corps de la requête
 - L'UUID du traitement,
 - La date,
 - Le statut
- Le service répond en transmettant
 - Le code retour

Message cURL d'entrée :

```
curl -X 'POST' \  
  'https://flux.esms.sans.restrictions.api.sidoba.fr/api/v1/statutCollecte \  
--header 'X-Request-progiciel: Moulinette API' \  
--header 'Content-Type: application/json' \  
--header 'Authorization: Bearer xxxxxxxxxx' \  
--data '{  
  "idTraitement": "cef0cbf3-6458-4f13-a418-ee4d7e7505dd",  
  "date": 1727788041,  
  "statut": "Traité moulinette SSIAD"  
}'
```

4.2 EXEMPLES DE MODELE METIER

4.2.1 MODÈLE MÉTIER D'UNE CAMPAGNE DE MODULE APA-PCH AU FORMAT JSON

L'exemple est fictif et n'a pas pour cible l'implémentation en cible POC API de collecte
Disponible dans le document référencé R4

4.2.2 MODÈLE MÉTIER D'UNE CAMPAGNE TARIFICATION (VOIR PAGE SUIVANTE)

L'exemple est fictif et n'a pas pour cible l'implémentation en cible POC API de collecte
Voir \Modèles métier exemple\exemple_fichier modele_metier_tarification_20231311.txt dans l'archive
Disponible dans le document référencé R5

4.3 MODELE METIER DU SERVICE DE COLLECTE DE DONNEES PAR API DE NIVEAU STRUCTURE/DOSSIER

Dans le cadre du POC de collecte par API de données de niveau structure/dossier, Capgemini et l'éditeur partenaire ont convenu que les campagnes de collecte adopteront un unique modèle métier.

Il contient dix variables dont une calculée (= non collectée)

Il est disponible dans l'archive dans le document référencé R3

4.4 MODELE METIER DU SERVICE DE COLLECTE DE DONNEES PAR API DE NIVEAU INDIVIDUEL

Dans le cadre du POC de collecte par API de données de niveau individuel, Capgemini et l'éditeur partenaire ont convenu que les campagnes de collecte adopteront un unique modèle métier.

Il contient dix variables.

Il est disponible dans l'archive dans le document référencé R2

4.5 EXEMPLE DE FICHIER FHIR

Un fichier formaté FHIR répondant aux contraintes des fichiers entrants du service WS07 est fourni dans l'archive.

Il décrit le cas suivant :

Un utilisateur transmet à l'API de collecte de données de niveau individuel un fichier qui contient un DUI de deux individus

Document référencé R6

4.6 EXEMPLES DE DOCUMENTS CDA

Deux documents formatés CDA répondant aux contraintes du service WS07 sont fournis dans l'archive.

Ils décrivent les cas suivants :

- Document référencé R7
 - Une grille GIR de Ruth NESSI, GIR valorisé à 3

- Document référencé R8
 - Une grille GIR de Kazemir MALEVICH, GIR valorisé à 2