

**CHOUETTE****Maintenance, accompagnement et recette de logiciels pour les échanges de données multimodales***API REST IEV V1.0*

Auteurs :	Michel ETIENNE, Daniel SURU, Marc FLORISSON (CityWay)
Relecteurs	Patrick GENDRE (CEREMA)

Résumé :

CHOUETTE est un logiciel libre développé à l'initiative du ministère français chargé des transports (et du développement durable) dans le but de faciliter l'échange de données d'offre (théorique) de transport collectif (TC), en s'appuyant pour cela sur la norme NFP 99506, dite Neptune, qui spécifie un profil d'échange XML.


Les utilisateurs visés sont les collectivités locales Autorités Organisatrices de Transport (AOT), les exploitants des réseaux TC, ainsi que leurs prestataires (bureaux d'étude ou société de services) et d'autres acteurs tels que services de l'état, éditeurs de logiciels, opérateurs de services d'information, chercheurs...

L'application CHOUETTE est disponible sous la forme d'une application WEB destinée à être déployée sur un serveur WEB, et d'une application en ligne de commande.

Ce document définit les interfaces externes du serveur IEV

Agence française pour l'information multimodale et la billettique

**Informations sur ce document :**

Organisme commanditaire : AFIMB			
Titre : API REST IEV V1.0			
Sous-titre :			
Organismes auteurs CITYWAY		Rédacteurs Daniel SURU Michel ETIENNE Marc FLORISSON	Participants Patrick Gendre, CEREMA Jean Seng, AFIMB
Maitre d'ouvrage AFIMB			
Mots clés : application Chouette, import, export, validation, Neptune, NeTEx, GTFS, KML, HUB		Diffusion :  publique (licence Creative Commons CC-by-nd)	
Nombre de pages : 24 pages	Date : Mai 2015	Confidentialité : Non	Bibliographie : Oui

Historique des versions / révisions :

Version	Date d'application	Description des changements	Auteur
1.0	16/03/2015	version initiale	M. ETIENNE D. SURU M. FLORISSON
1.1	16/03/2015	Ajout d'un chapitre propre à la navigation dans l'API	M. ETIENNE M. FLORISSON
1.2	18/03/15	Gestion des versions de l'API	M. ETIENNE M. FLORISSON
1.3	24/03/15	Relecture	M. ETIENNE M. FLORISSON
1.4	31/03/15	Prise en compte des remarques MOE	M. ETIENNE
1.5	18/05/15	Mise à jour de fin de développement	M. ETIENNE



Table des matières

1	Introduction.....	5
2	Documents.....	5
3	Protocole.....	6
3.1.	HATEOAS.....	6
3.2.	Méthodes HTTP utilisées.....	6
3.3.	Utilisation de HTTP pour les traitements asynchrones.....	7
3.4.	Contrôle des accès.....	7
3.5.	Gestion des versions de l'API IEV.....	7
4	Traitement des opérations en parallèle.....	8
5	Modèle de données.....	8
5.1.	Règles applicables au modèle de données.....	10
6	Services disponibles.....	10
6.1.	Conventions et notations.....	10
6.2.	Codes d'erreur possibles et codes HTTP associés.....	11
6.3.	Soumission d'une opération.....	11
6.4.	Suivi d'une opération en cours.....	13
6.5.	Annulation d'une opération en cours.....	14
6.6.	Consultation des résultats d'une opération terminée.....	15
6.7.	Consultation des ressources complémentaires d'une opération.....	16
6.8.	Liste des opérations.....	18
6.9.	Suppression des résultats d'une opération terminée.....	18
6.10.	Suppression de toutes les opérations sur un espace de données.....	19
7	Navigation dans les Services.....	20
7.1.	Symbolique utilisées.....	20
7.2.	Création d'une opération.....	21
7.3.	Consultation de la liste des opérations.....	21
7.4.	Cas d'utilisation : création et suivi d'une opération.....	22





1 INTRODUCTION

Ce document spécifie la version 1.0 des interfaces d'accès aux opérations d'import, d'export et de validation de Chouette.

Le serveur IEV publie une interface HTTP Rest qui permet de :

- demander une nouvelle opération
- suivre une opération en cours
- récupérer les résultats d'une opération
- lister les opérations

Vis à vis de la demande d'une nouvelle opération, le serveur IEV traite l'opération de manière asynchrone.

Ce document est accompagné d'un ensemble de documents en XML Schema cités dans les paragraphes où ils sont décrits.

2 DOCUMENTS

Ref	Titre	Contenu	Date	Version	Auteurs
[D01]	Chouette : Paramètres de validation	Définitions des paramètres de validation	15/03/2015	1.0	Michel ETIENNE
[D02]	Chouette : métadonnées à l'export	Définition des fichiers de métadonnée accompagnant les export Chouette	20/01/2015	1.0	Michel ETIENNE Daniel SURU
[D03]	Chouette : Mapping GTFS	Correspondance entre les modèles Chouette et GTFS	25/11/2014	2.0	Michel ETIENNE
[D04]	Chouette : implémentation de NEPTUNE	Correspondance entre les modèles Chouette et Neptune	18/06/2014	Draft	Michel ETIENNE
[D05]	Chouette : Spécification des échanges NeTEx	Correspondance entre les modèles Chouette et NeTEx (expérimental)	31/05/2013	1.0.1	Marc FLO-RISSON
[D06]	Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing (http://tools.ietf.org/html/rfc7230)	Syntaxe des messages du protocole HTTP 1.1	06/2014	rfc7230	W3C
[D07]	Hypertext Transfer Proto-	Syntaxe et grammaire du	06/2014	rfc72	W3C



	col (HTTP/1.1): Semantics and Content (http://tools.ietf.org/html/rfc7231)	protocole HTTP 1.1		31	
[D08]	Restfull Objects (http://restfulobjects.org/)	Spécification d'une API Restful et définition d'HATEOAS		1.0.0	Dan Haywood
[D09]	Chouette IEV : paramètres des opérations	Définition de la structure des paramètres des opérations du serveur IEV	18/05/2015	1.0	Michel ETIENNE
[D10]	Chouette IEV : rapports	Définition de la structure des rapports du serveur IEV	18/05/2015	1.0	Michel ETIENNE
[D11]	Hypertext Transfer Protocol (HTTP/1.1): 17.13.4 Form content types (http://www.w3.org/TR/html401/interact/forms.html#h-17.13.4.2)	Définition d'une requête HTTP multipart/form-data		html 401	W3C
[D12]	Chouette : document d'architecture	Présentation de l'architecture de Chouette	28/03/2015	3.0	Michel ETIENNE Marc FLORISSON

3 PROTOCOLE

Le protocole implémenté est conforme à une interface http Restful HATEOAS

3.1.HATEOAS

HATEOAS (Hypertext As The Engine Of Application State, cf D08) est utilisé pour :

- limiter la connaissance requise pour utiliser l'interface IEV
- augmenter l'évolutivité de l'interface IEV

3.2.Méthodes HTTP utilisées

Les requêtes utilisent les différentes méthodes http :

GET pour les consultations (collection ou ressource)

POST pour la création de ressource

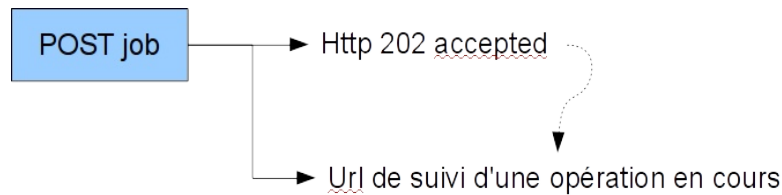
DELETE pour la suppression de ressource



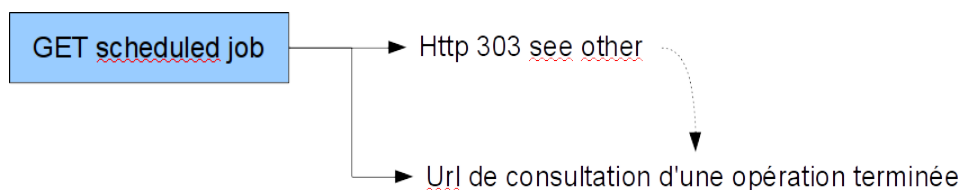
3.3. Utilisation de HTTP pour les traitements asynchrones

Les API RESTFUL s'appuient sur le protocole HTTP dès que possible ; dans le cas des traitements asynchrones du serveur IEV, 2 codes du protocole sont particulièrement exploités :

- HTTP 202 : accepted
 - une demande asynchrone a été acceptée, l'entête http renseigne alors l'url de suivi de la demande dans le champ *location*.



- HTTP 303 : see other
 - la demande suivie est terminée, l'url n'est donc plus valide, l'entête http enseigne alors l'url de consultation du résultat de la demande dans le champ *location*.



3.4. Contrôle des accès

Le serveur IEV n'ayant pas d'accès public, n'implémente aucun contrôle d'accès

3.5. Gestion des versions de l'API IEV

Comme tout composant logiciel, l'interface IEV sera amenée à évoluer. Il est donc nécessaire de mettre en place dès à présent une gestion de versions de cette API IEV afin de faciliter la continuité de fonctionnement des applications tierces en interface avec celle-ci.

Si le client des interfaces IEV ne précise rien dans ses requêtes, c'est la version par défaut choisie par le fournisseur IEV qui s'applique.

Pour spécifier une version particulière, le client indique la version d'interface utilisée dans des propriétés HTTP de la requête.

Les réponses HTTP des services retournent toutes la version de l'API IEV prise en compte.



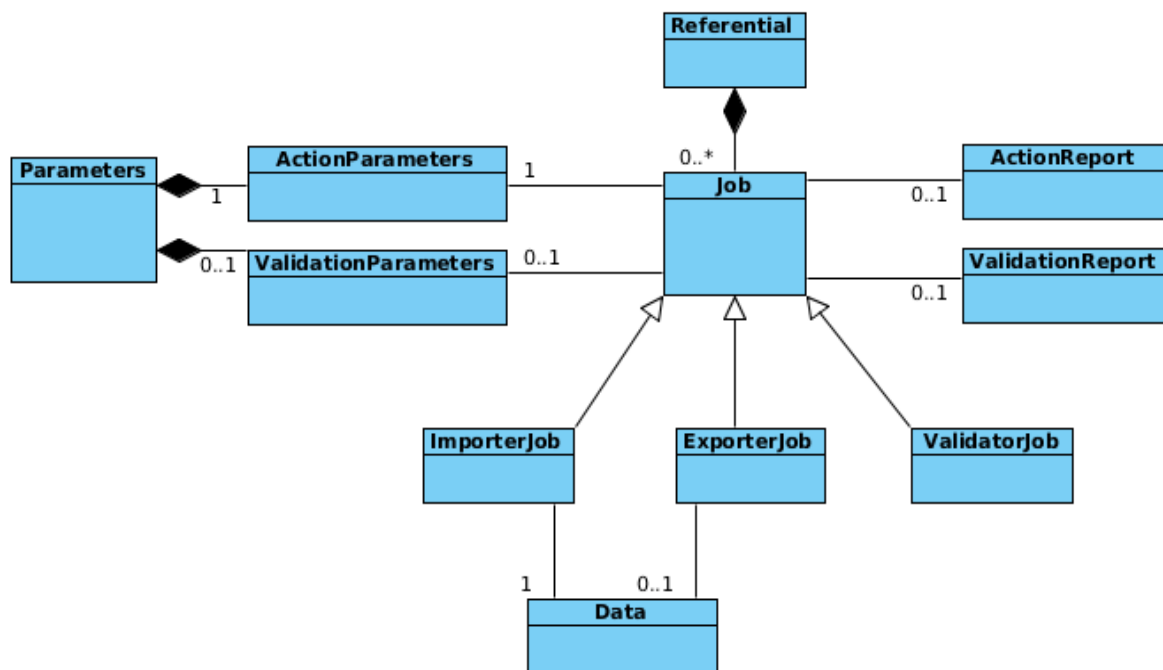
4 TRAITEMENT DES OPÉRATIONS EN PARALLÈLE

Chaque opération est enregistrée dans une file d'attente avant d'être lancée dès qu'une tâche de traitement est disponible ; deux opérations ne peuvent être lancées en parallèle que si elles ne concernent pas le même espace de données.

Lorsqu'une opération est lancée, elle peut être interrompue, mais le traitement déjà effectué sera enregistré.

5 MODÈLE DE DONNÉES

Les API du serveur IEV reposent sur le modèle de données suivant :



Job :

opération ; c'est la ressource principale manipulée par le serveur IEV

le Job est décliné en 3 types : importer, exporter et validator.

Durant son cycle de vie, le job passe par 2 états :

- **scheduled** : le job est en attente ou en cours d'exécution
- **terminated** : le job est terminé ou a été annulé

ImporterJob :



opération d'import, cette ressource dérivée de Job la complète d'un jeu de données (Data) qu'elle reçoit lors de sa création et qui sera la source de données de l'opération d'import.

ExporterJob :

opération d'export, cette ressource dérivée de Job la complète d'un jeu de données (Data) qu'elle produit en résultat d'opération.

ValidatorJob :

opération de validation, cette ressource dérivée de Job ne dispose pas de ressource externe, elle agit uniquement sur les données de transport contenue dans le schéma (SQL) du référentiel parent.

Parameters :

paramètres de l'opération ; cette ressource se décompose en 1 à 2 sous-éléments : ActionParameters et ValidationParameters

ActionParameters :

paramètres de l'action : entité abstraite déclinée pour chaque action et s'il y a lieu, pour chaque format ; les différents paramètres d'action sont définis dans le document D09.

ValidationParameters :

paramètres de validation de niveau supérieur à 2 ; ces paramètres sont définis dans le document D01

ActionReport :

rapport d'exécution de l'opération, cette ressource évolue lors de l'avancement de l'opération, elle trace le nombre d'entités traitées ainsi que les incidents de traitements rencontrés. (cf document D10)

Cette ressource est ajoutée à la ressource Job lorsque le traitement a démarré.

ValidationReport :

rapport de validation, la présence de cette ressource et son contenu dépendent de l'opération, du format des données et de la présence de paramètres de validation. (cf document D10)

Cette ressource est ajoutée à la ressource Job lorsque le traitement a démarré.

Data :

données traitées ou produites par l'action :

- à l'import, cette ressource est la source de donnée alimentant celui-ci



- à l'export, cette ressource est le résultat téléchargeable de cette opération

Referential :

Ressource externe objet des différentes opérations ; elle est directement liée au schéma (SQL) des données transport à tester. (cf [D12]).

Ce n'est pas une ressource accessible par les API du serveur mais elle intervient dans la définition des url du serveur

5.1. Règles applicables au modèle de données

Les documents D09 et D10 définissent le détail des structures du modèle de données

Règle 1 : La progression d'une action est toujours fournie

Règle 2 : Un rapport d'action en erreur (action_report.result = ERROR) contient toujours une description de l'erreur (présence de l'élément action_report.failure)

Règle 3 : Le rapport d'une action interrompue (volontairement ou non, i.e. dont la structure Job définit un status à CANCELLED ou ABORTED) contient l'état de la dernière étape effectuée, le résultat peut donc rester à l'état OK, toutefois la progression montre que le traitement n'a pas été terminé.

Règle 4 : le lien sur le rapport d'action est disponible dès la création de celle-ci, tant que l'opération n'est pas lancée, il ne contient qu'une progression à 0% et un résultat à OK.

Règle 5 : le lien sur le rapport de validation n'existe que si celui-ci contient au moins un point de contrôle.

Règle 6 : le lien sur les paramètres d'action est toujours disponibles

Règle 7 : le lien sur les paramètres de validation est disponible si de tels paramètres ont été fournis à la création de l'action

Règle 8 : le lien sur les données d'import est toujours disponibles

Règle 9 : le lien sur les données d'export n'existe que si celle-ci ont pu être produites, même partiellement.

6 SERVICES DISPONIBLES

6.1. Conventions et notations

Les url des requêtes HTTP font apparaître différentes parties variables qui sont notées selon la règle ci-dessous :

- Une partie obligatoire pouvant avoir différents contenus, ces contenus pouvant ou non appartenir à une liste de valeurs autorisées, est notée entre {}



- Par exemple : `Http://mon_domaine/{mes_ressources}`, la partie `{mes_ressources}` nécessite de renseigner une chaîne non vide.
- Une partie optionnelle pouvant avoir différents contenus, ces contenus pouvant ou non appartenir à une liste de valeurs autorisées, est notée entre `[]`
 - Par exemple : `Http://mon_domaine/lines[/single]`, la partie `[/single]` est optionnelle.

Toutes les propriétés HTTP suivent également cette règle.

Par ailleurs, l'ensemble des services gèrent de la même manière la version de l'API IEV.

La requête peut préciser la version d'API IEV souhaitée en utilisant la propriété HTTP « Accept ».

```
Accept: application/vnd.iev.{VERSION}+json
{VERSION} contenant pour l'instant exclusivement « 1.0 » en attendant que
les versions futures de l'API définissent d'autres valeurs .
```

De même les réponses intègrent toutes un paramètre applicatif spécifique « X-ChouetteIEV-Media-Type » qui explicite la version d'interface prise en compte :

```
X-ChouetteIEV-Media-Type: iev.v1.0; format=json
```

Dans cet exemple c'est la version 1.0 qui est prise en compte.

6.2. Codes d'erreur possibles et codes HTTP associés

Chacun des services peut remonter une erreur HTTP, celle-ci est accompagnée d'une structure JSON (dans le body de la réponse) contenant 1 élément :

- `error_code` : code de l'incident

```
exemple : { "error_code" : "UNKNOWN_REFERENTIAL" }
```

Les erreurs sont codées afin de permettre au client d'afficher un message clair selon la langue désirée

Code	Code HTTP	description
INVALID_REQUEST	400	Requête invalide
UNKNOWN_REFERENTIAL	404	Référentiel inconnu
UNKNOWN_FILE	404	Fichier inconnu
UNKNOWN_ACTION	400	Action ou type inconnu
DUPPLICATE_OR_MISSING_DATA	400	Donnée manquante ou en double
DUPPLICATE_PARAMETERS	400	Paramètres fournis en double
MISSING_PARAMETERS	400	Paramètres d'action manquants



INVALID_PARAMETERS	400	Paramètres d'action incorrects
UNREADABLE_PARAMETERS	400	Paramètres non lisibles (format erroné)
UNKNOWN_JOB	404	Numéro de job inconnu
SCHEDULED_JOB	405	Méthode interdite sur un job non terminé
INTERNAL_ERROR	500	Erreur interne

6.3. Soumission d'une opération

Rôle :

Ce service permet de créer une opération IEV :

- import de données dans un espace de données
- export de données depuis un espace de données
- validation de données de l'espace de données

L'opération ainsi créée est mise en file d'attente pour être exécutée ultérieurement ; le service ne peut donc pas retourner un résultat.

Conformément au mécanisme http décrit au § 3.3 il redirige dans sa réponse vers le service de suivi d'opérations.

URL de requête :

`http://[chouette_iev]/referentials/{ref}/{action}/{type}`

- ref : nom du schéma SQL de l'espace de données
- action : opération à réaliser (importer /exporter /validator)
- type : caractérisation de l'action
 - importer : neptune,netex,gifs
 - exporter : neptune,netex,gifs (puis kml et hub)
 - validator : *optionnel : identiques à l'import ; si non renseigné, alors la source des données est la base.*

Propriétés HTTP de la requête :

méthode POST multipart (content-type: multipart/form-data) (cf document D11)



La requête peut contenir 1 ou 2 parts détaillés ci-après,. l'ordre des parts dans la requête est indifférent, la sémantique des données du « part » est déterminée par l'attribut « filename ».

part 1 : paramètres de traitement (toutes actions)

Content-Disposition: form-data; name="file"; filename="parameters.json"
Content-Type: application/json

ce part doit impérativement avoir l'attribut *filename="parameters.json"*

le contenu de ce part est en json et doit contenir un élément parameters défini dans le schéma *jobs.xsd*

ce schéma est décrit dans le document D09

part 2 : données d'offre de transport (uniquement si action = importer)

Content-Disposition: form-data; name="file"; filename="{data}.{ext}"
Content-Type: application/octet-stream ou selon format du fichier

data : nom du fichier

ext : zip ou xml selon le type d'import

Réponse :

HTTP 202 (Accepted) : l'opération est mise en file d'attente

entête http :

- Location : URL de suivi de l'opération (cf § 6.4)

HTTP 400 (Bad Request) : si l'action ou le type sont inconnus.

HTTP 404 (Not Found) : si l'espace de donnée est inconnu

HTTP 500 (Internal Server Error) : erreur interne

6.4.Suivi d'une opération en cours

Rôle :

Ce service fournit l'état d'avancement de l'opération ainsi que des liens sur les ressources complémentaires.

Lorsque l'opération est terminée, le service redirige (cf § 3.3) sur la consultation des résultats d'une opération terminée (cf § 6.6)

URL de requête :

http://[chouette_iev]/referentials/{ref}/scheduled_jobs/{id}

- ref : nom du schéma SQL de l'espace de données



- id : identifiant de l'opération (fourni en retour de la soumission ou en retour de la liste des opérations)

Note : cette url est fournie en retour des services :

- soumission d'opération
- liste des opérations
- suivi d'une opération en cours
- consultation d'un rapport de validation

Propriétés HTTP de la requête :

méthode GET

Réponse :

HTTP 200 (OK) : la demande est acceptée

entête http :

- Link rel=cancel : url d'annulation de l'opération (cf §6.5)
- Link rel=data : url des données à traiter (action = import)
- Link rel=action_params : url des paramètres de l'opération (cf document [D09])
- Link rel=validation_params : url des paramètres de validation (si fournis à la soumission) (cf document [D09])
- Link rel=action_report : rapport d'opération avec progression (si l'opération est en cours) (cf document [D10])
- Link rel=validation_report : rapport de validation partiel (si l'opération est en cours) (cf document [D10])

corps de la réponse :

- Métadonnées de l'opération (si démarrée) (cf document [D09])

HTTP 303 (See Other) : si la demande est périmée, suivre la nouvelle URL fournie

entête http :

- Location : url de consultation du résultat de l'opération (cf §6.6)

HTTP 404 (Not Found) : l'espace de donnée ou l'opération sont inconnus

HTTP 500 (Internal Server Error) : erreur interne



6.5. Annulation d'une opération en cours

Rôle :

Ce service permet d'interrompre une opération en attente ou en cours de traitement ; il force l'opération à passer à l'état terminé ; les résultats partiels sont ainsi accessibles avant de supprimer toute trace de cette opération. La suppression définitive sera réalisée par le même service que pour une opération terminée normalement (cf § 6.9)

La demande est sans effet sur une opération terminée.

Note : l'annulation n'est pas immédiate ; si l'opération est en cours, celle-ci terminera l'étape élémentaire en cours afin de mettre les résultats dans un état exploitable.

URL de requête :

La forme de l'URL n'est pas publique, celle-ci est disponible uniquement au travers des réponses aux requêtes suivantes :

- la liste des opérations
- le suivi d'une opération en cours
- la consultation d'un rapport de validation

Propriétés HTTP de la requête :

méthode DELETE

Réponse :

HTTP 200 (OK) : la demande est acceptée

- entête http :
 - Link rel=delete : url d'annulation de l'opération (cf §6.5)
 - Link rel=data : url des données à traiter (action = import) ou des données produites (action) export)
 - Link rel=action_params : url des paramètres de l'opération (cf document [D09])
 - Link rel=validation_params : url des paramètres de validation (si fournis à la soumission) (cf document [D09])
 - Link rel=action_report : rapport d'opération (si l'opération était en cours) (cf document [D10])
 - Link rel=validation_report : rapport de validation (si l'opération était en cours) (cf document [D10])
 - link rel=location : Métadonnées de l'opération (cf document [D09])



HTTP 404 (Not Found) : si l'espace de donnée ou l'opération sont inconnus

HTTP 500 (Internal Server Error) : erreur interne

Note : seule une opération en attente ou en cours de traitement peut être annulée, si l'opération est déjà terminée, un retour ok est remonté.

6.6.Consultation des résultats d'une opération terminée

Rôle :

Ce service fournit l'état final de l'opération ainsi que des liens sur les ressources complémentaires.

URL de requête :

`http://[chouette_iev]/referentials/{ref}/terminated_jobs/{id}`

- ref : nom du schéma SQL de l'espace de données
- id : identifiant de l'opération (fourni en retour de la soumission ou en retour de la liste des opérations)

Note : cette url est fournie en retour des services :

- liste des opérations
- consultation d'une opération terminée
- consultation d'un rapport de validation

Propriétés HTTP de la requête :

méthode GET

Réponse :

HTTP 200 (OK) : la demande est acceptée

entête http :

- Link rel=delete : url de suppression de l'opération (cf §6.9)
- Link rel=data : url des données à traiter (action = import) ou des données produites (action) export)
- Link rel=action_params : url des paramètres de l'opération (cf document [D09])
- Link rel=validation_params : url des paramètres de validation (si fournis à la soumission) (cf document [D09])
- Link rel=action_report : rapport de l'opération (si l'opération n'a pas été annulé avant de débiter) (cf document [D10])



- Link rel=validation_report : rapport de validation (si l'opération n'a pas été annulé avant de débiter) (cf document [D10])

corps de la réponse :

- Métadonnées de l'opération (cf document [D09])

HTTP 405 (Method Not Allowed) : si l'opération n'est pas terminée

HTTP 404 (Not Found) : si l'espace de donnée ou l'opération sont inconnus

HTTP 500 (Internal Server Error) : erreur interne

6.7.Consultation des ressources complémentaires d'une opération

Rôle :

Ce service fournit une ressource associée à l'opération et les mêmes liens que ceux fournis par les services de suivi ou de consultation d'une opération selon l'état d'avancement de l'opération.

URL de requête :

`http://[chouette_iev]/referentials/{ref}/data/{id}/{ressource}`

- ref : nom du schéma SQL de l'espace de données
- id : identifiant de l'opération (fourni en retour de la soumission ou en retour de la liste des opérations)
- ressource :
 - action_report.json : rapport d'opération (cf Document [D10])
 - validation_report.json : rapport de validation (cf Document [D10])
 - action_parameters.json : paramètres de l'opération (cf Document [D09])
 - validation_parameters.json : paramètres de validation (cf Document [D09])
 - autre : données importées ou exportées (le nom de la ressource n'est disponible que par l'exploitation des liens fournis par les autres services)

Note : cette url est fournie en retour des services :

- liste des opérations
- consultation d'une opération en cours
- consultation d'une opération terminée

Propriétés HTTP de la requête :

méthode GET



Réponse :

HTTP 200 (OK) : la demande est acceptée

entête http :

- Link rel=delete : url de suppression de l'opération (cf § 6.9)
- Link rel=data : url des données à traiter (action = import) ou des données produites (action = export)
- Link rel=action_params : url des paramètres de l'opération (cf document [D09])
- Link rel=validation_params : url des paramètres de validation (si fournis à la soumission) (cf document [D09])
- Link rel=action_report : rapport de l'opération (si l'opération est en cours ou terminée) (cf document [D10])
- Link rel=validation_report : rapport de validation (si l'opération est en cours ou terminée) (cf document [D10])

corps de la réponse :

- Ressource demandée HTTP 404 (Not Found) : si l'espace de donnée ou l'opération sont inconnus ou si la ressource n'existe pas

HTTP 500 (Internal Server Error) : erreur interne

Note : tant qu'une ressource n'existe pas, le lien sur celle-ci ne sera pas fourni par les autres services

6.8. Liste des opérations

Rôle :

Ce service retourne la liste des opérations du référentiel éventuellement filtrée sur un type d'action. La liste retournée précise, dans le corps de la réponse, les liens pour exploiter les différentes ressources associées à chaque opération ; on retrouvera selon que l'opération est en cours ou terminée, les mêmes liens que ceux fournis respectivement par le service de suivi d'une opération en cours ou de consultation d'une opération terminée.

URL de requête :

http://[chouette_iev]/referentials/{ref}/jobs?action=actionName

- ref : nom du schéma SQL de l'espace de données
- actionName : filtre sur une action

Propriétés HTTP de la requête :

méthode GET



Réponse :

HTTP 200 (OK) : la demande est acceptée

corps de la réponse :

- Liste des opérations (cf document [D09])

HTTP 400 (BadRequest) : si l'action est inconnue

HTTP 404 (Not Found) : si l'espace de donnée est inconnu

HTTP 500 (Internal Server Error) : erreur interne

6.9. Suppression des résultats d'une opération terminée

Rôle :

Ce service supprime l'ensemble des informations sur une opération terminée ou annulée ; une fois l'opération supprimée, il n'existe plus de trace de son existence dans le serveur ; tout appel à un service référençant l'identifiant de cette opération retournera une erreur Http 404 (Not Found)

URL de requête :

La forme de l'URL n'est pas publique, celle-ci est disponible uniquement au travers des réponses aux requêtes suivantes :

- la liste des opérations
- la consultation d'une opération terminée
- la consultation d'un rapport de validation

Propriétés HTTP de la requête :

méthode DELETE

Réponse :

HTTP 200 (OK) : la suppression est acceptée

HTTP 405 (Method Not Allowed) : l'opération n'est pas terminée

HTTP 404 (Not Found) : l'espace de donnée ou l'opération sont inconnus

HTTP 500 (Internal Server Error) : erreur interne

Note : seule une opération terminée ou annulée peut être supprimée



6.10. Suppression de toutes les opérations sur un espace de données

Rôle :

Ce service supprime toutes les opérations d'un espace de données sans nécessairement supprimer le schéma associé à l'espace de données; le service ne retourne qu'un simple acquittement ou une erreur.

L'IHM de Chouette permet à un utilisateur de supprimer un espace de données et donc son schéma de base associé. Pour cela, le serveur IEV a besoin d'être notifié d'une demande de suppression de l'ensemble des opérations réalisées dans un espace de données concerné.

URL de requête :

`http://[chouette_iev]/referentials/{ref}/jobs`

- ref : nom du schéma SQL de l'espace de données

Propriétés HTTP de la requête :

méthode DELETE

Réponse :

HTTP 200 (OK): la suppression est acceptée

HTTP 404 (Not Found) : l'espace de donnée est inconnu

HTTP 500 (Internal Server Error) : erreur interne

7 NAVIGATION DANS LES SERVICES

Le client donc dispose de 5 services comme départ de sa navigation :

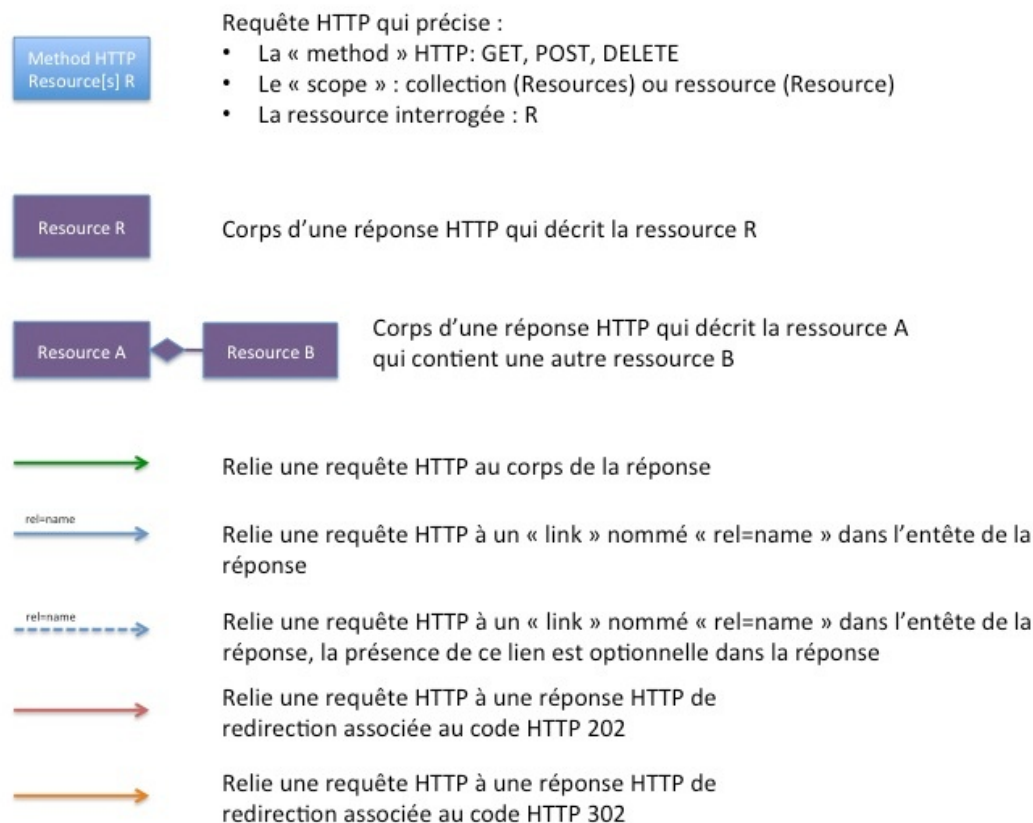
- Création d'une opération
- Suivi d'une opération en cours
- Consultation d'une opération terminée
- Consultation de la liste des opérations
- Suppression des opérations d'un référentiel

En s'appuyant sur le concept HATEOAS, la réponse peut fournir des liens vers les ressources en relation que le client peut consulter ensuite.



Les schémas qui suivent montrent comment naviguer entre les différents services, les 2 principaux services étant la création d'une opération et la consultation de la liste des opérations.

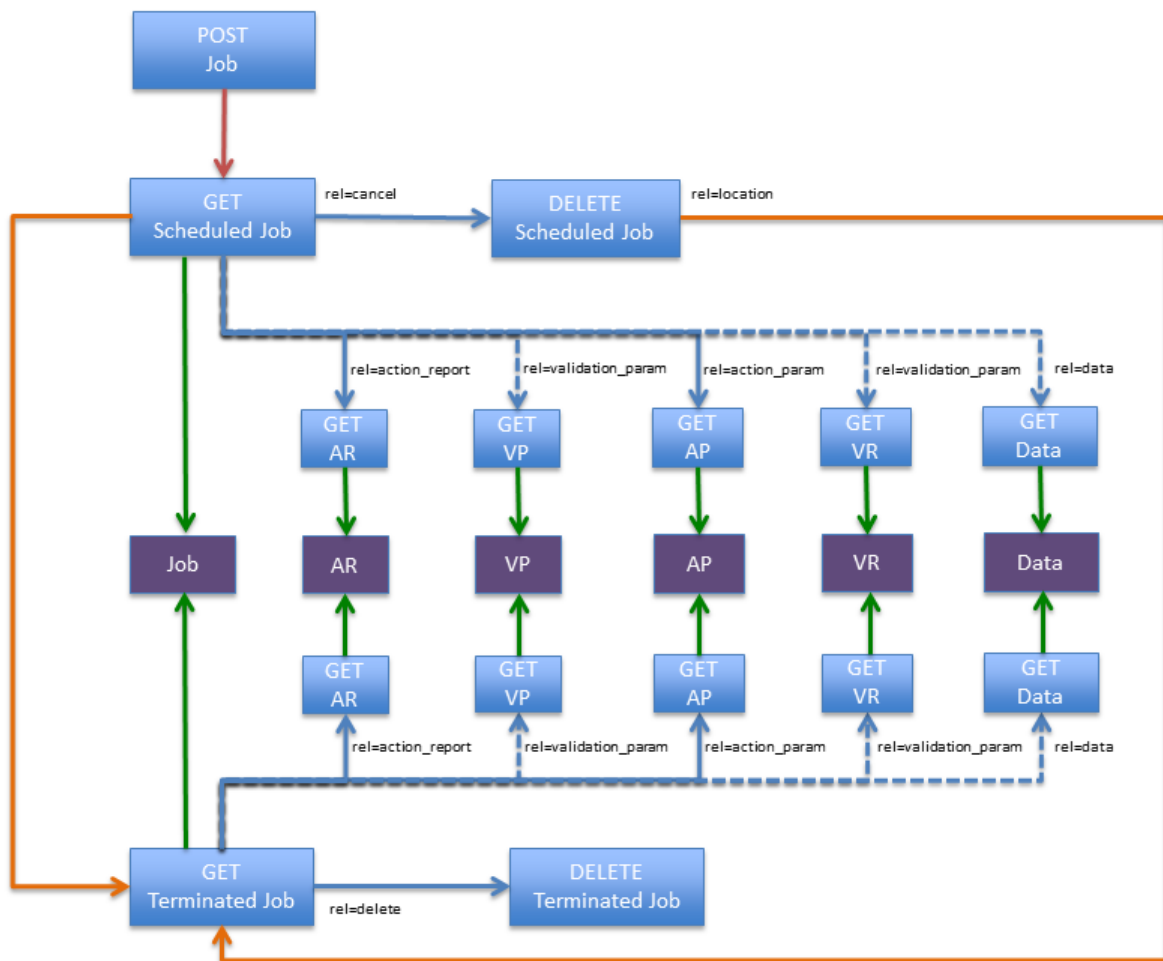
7.1. Symbolique utilisées



7.2. Création d'une opération

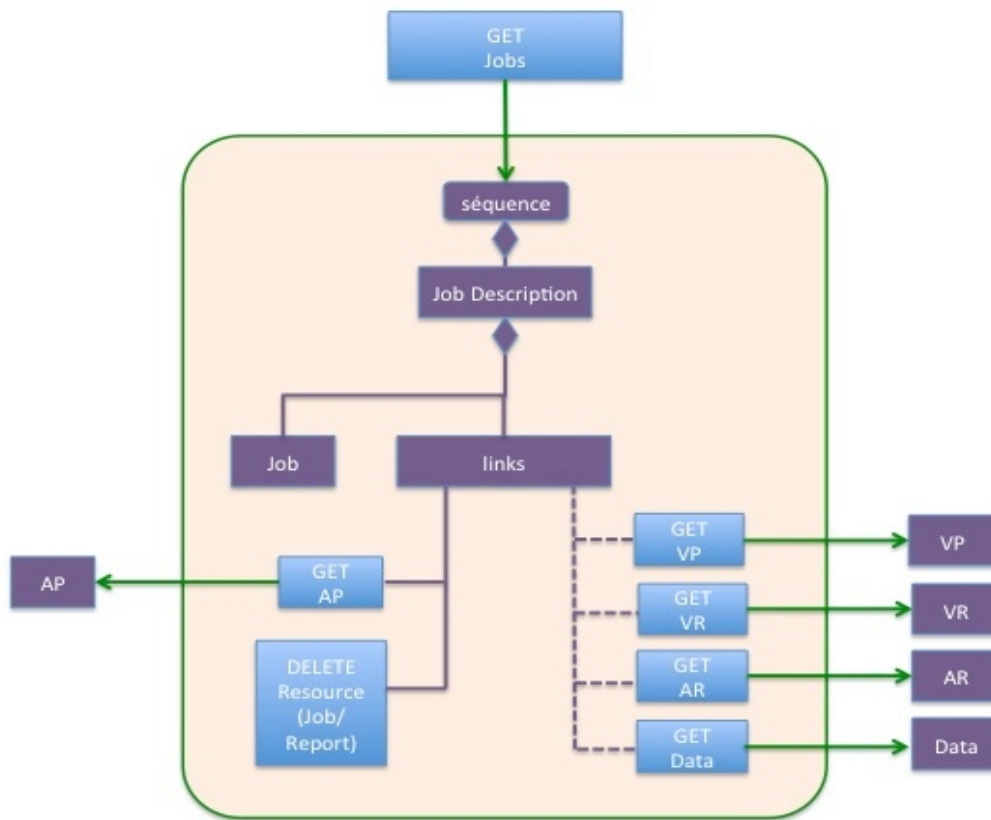
Au cours de l'exécution de l'opération (ressource Job), le client peut suivre son état d'avancement ; l'url sollicitée deviendra invalide à la fin du traitement et redirigera alors sur l'url de consultation du résultat de l'opération. (cf § 3.3)

A partir de cette API, le client dispose donc des liens de navigation suivants :



7.3. Consultation de la liste des opérations

Les opérations en cours ou terminées sont consultables sans en connaître au préalable leur identifiant ; le service de consultation de la liste des opérations offre un point de départ pour la navigation dans l'ensemble des opérations existantes.



La réponse contient une liste de 0 à n opérations en attente, en cours et terminées.

7.4.Cas d'utilisation : création et suivi d'une opération

Le diagramme de séquence ci-dessous décrit comment le client peut créer une opération et la suivre jusqu'à ce qu'elle se termine :

