



**PREMIER  
MINISTRE**

*Liberté  
Égalité  
Fraternité*

## HUB D'ÉCHANGE DE L'ÉTAT

**Présentation des API**

# Sommaire



**Architecture – Présentation générale**

**Architecture – Fonctions des API et modèle objet**

**Les API du Hub**

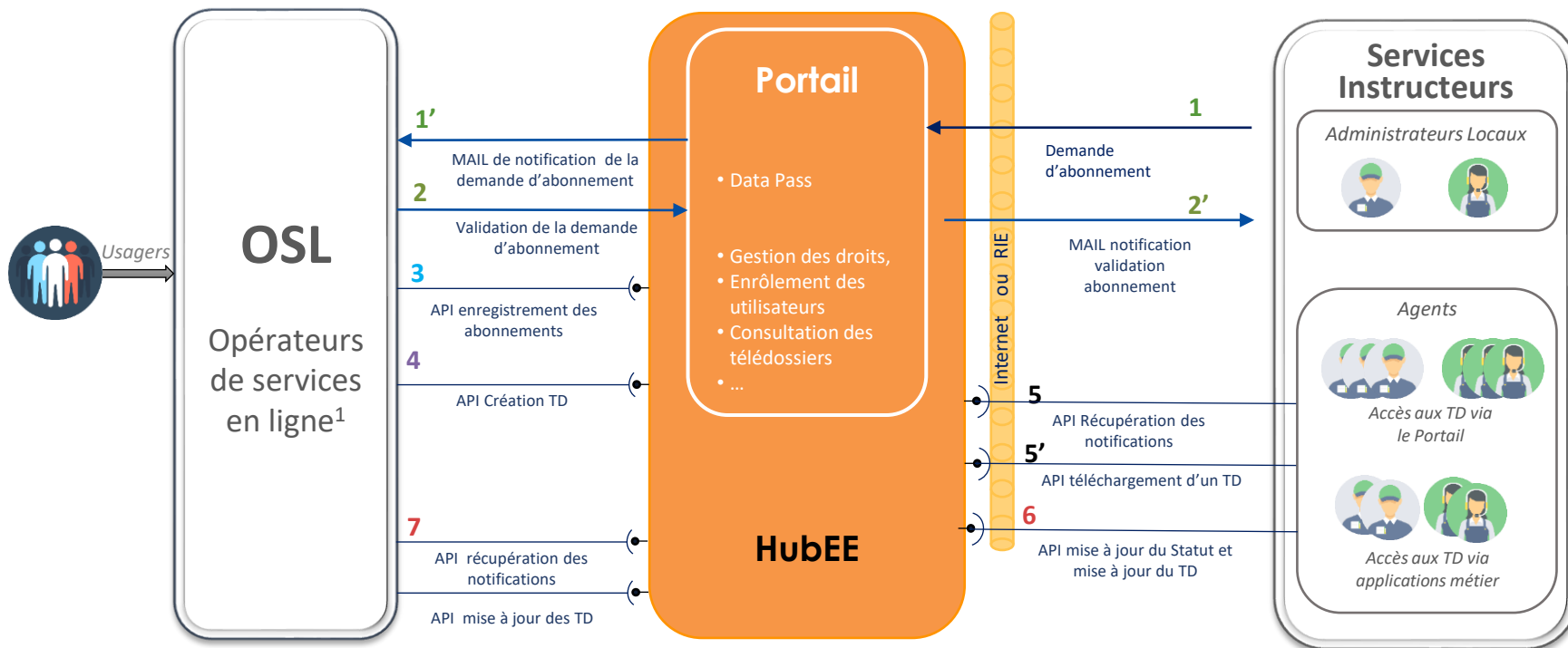
**Cinématiques et API**

**Planning global**

# Architecture du Hub d'Échange de l'État

Intitulé de la direction/service

## Les principes



<sup>1</sup> DILA via sa Plateforme de Service en Ligne (PSL), CNAF via son site web, DGS via l'application CertDc

<sup>2</sup> Mairies, Conseils départementaux, Ministères, opérateurs de la protection sociale...

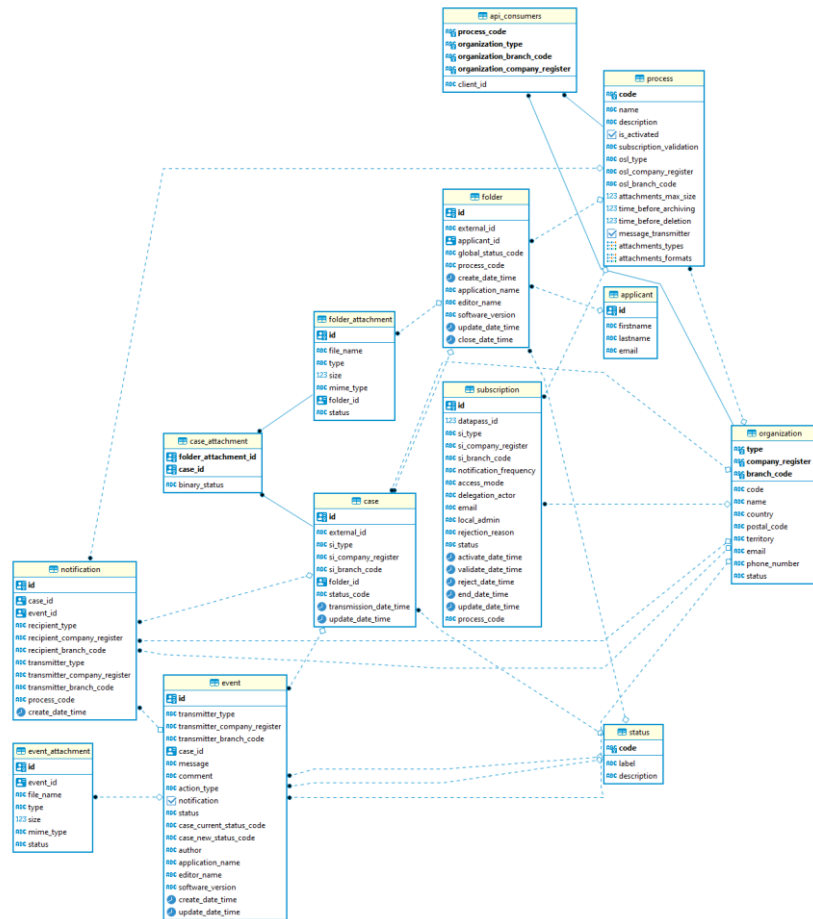
## Les Fonctions des API

- API First : HubEE fonctionne nativement avec des API
  - API REST
  - Authentification OAuth 2
  - TLS 1.2
  - Accès des SI sur RIE aux API via la PFAI
  - Principe de gestion des versions sur 3 digits (X.Y.Z = <majeur>. <mineur>.<correctif>)
- Exigences non fonctionnelles :
  - Garantie de délivrance des télédossiers
  - Garantie de l'intégrité des télédossiers (contrôle antivirus, type et taille de pj, ...)
  - Traçabilité des événements pour chaque dossier
  - Les OSL sont responsables de la maintenance 1<sup>er</sup> niveau
  - Supervision de tous les événements

# Architecture HubEE

## Modèle objet

- Télédossiers
  - Métadonnées
    - Folder = les métadonnées communes à tous les destinataires
    - Case = les métadonnées liées à un destinataire
    - Event = les métadonnées liées à la mise à jour des TD
  - Attachement = les PJ des TD liées
    - au Folder
    - au Case
    - À l'Event d'un case si maj d'un TD



Un TD contient donc autant de Cases que de destinataires. C'est l'OSL qui est responsable de la création des folders et cases et de l'attachement des PJ

	OSL	SI	Admin	Datapass
Folder	X		X	
Case	X	X	X	
Attachment	X	X	X	
Event	X	X	X	
Subscription	X		X	X
Notification	X	X		
User			X	
Organization			X	X
Process			X	

# Les API HubEE

## Liste des API disponibles

id	OSL	SI	ADMIN	Datap ass	Méthode	API	Ressource	URI	Description
1	x		x		POST	TD	Folder	/folders	Permet la création des métadonnées sur TD
2	x		x		PUT	TD	NA	/folders/{idFolder}/attachments/{idAttachment}	Permet l'ajout des PJ dans le HUB
3	x	x	x		POST	TD	Case	/cases/{idCase}/events	Permet l'ajout d'un Event dans le TD
4	x		x		PATCH	TD	Folder	/folders/{idFolder}	Permet la modification du statut du TD
5	x	x	x		PATCH	TD	Case	/cases/{idCase }	Permet la modification du statut d'un Case
6	x	x	x		PATCH	TD	Case	/cases/{idCase}/events/{idEvent}	Permet la modification du statut de prise en compte de l'Event par le destinataire
7	x	x	x		PUT	TD	Case	/cases/{idCase}/events/{idEvent}/attachments/{idAttachment}	Perme l'ajout d'une PJ dans l'Event d'un Case
8	x		x		GET	TD	Folder	/folders/{idFolder}	Permet la récupération d'un TD
9	x	x	x		GET	TD	Case	/cases/{idCase}/events/{idEvent}/attachments/{idAttachment}	Permet la récupération d'un Case
10	x		x		GET	Abonnement	Subscription	/subscriptions	Permet de récupérer les abonnements par lot des Services instructeurs par les OSL.
11	x		x		GET	Abonnement	Subscription	/subscriptions/{idSubscription}	Permet de récupérer un abonnement par son identifiant
12	x	x	x		GET	TD	Folder	/notifications	Permet de récupérer par lot les notification
13	x	x	x		DELETE	TD	Folder	/notifications/{idNotification}	Permet de supprimer physiquement les notifications
14		x	x		GET	TD	Case	/cases/{idCase}	Permet de récupérer un Case par son identifiant technique
15		x	x		GET	TD	Case	/cases/{idCase}/attachments/{idAttachment}	Permet de récupérer le fichier binaire d'un Case
16	x	x	x		GET	TD	Case	/cases/{idCase}/events/{idEvent}	Permet de récupérer un Event d'un Case par son identifiant
17	x		x		DELETE	TD	Folder	/folders/{idFolder}	Suppression d'un folder par son identifiant
18			x	x	POST	Abonnement	Subscription	/subscriptions	Permet de créer un abonnement dans le hub
19			x		PUT	Abonnement	Subscription	/subscriptions/{idSubscription}	Permet de modifier un abonnement existant
20			x		POST	Organisation	Organization	/organizations	Permet de créer des organisations dans le hub
21			x		PUT	Organisation	Organization	/organizations/{idOrganization}	Permet de modifier une organisation du hub
22			x		GET	Organisation	Organization	/organizations/{idOrganization}	Permet de récupérer une organisation du hub
23			x		GET	Organisation	Organization	/organizations/	Permet de récupérer par lot les organisations
24			x		POST	Démarche	Process	/processes	Permet de créer des démarches dans le hub
25			x		PUT	Démarche	Process	/processes/{code}	Permet de modifier une démarche du hub
26			x		GET	Démarche	Process	/processes/{code}	Permet de récupérer une démarche du hub
27			x		GET	Démarche	Process	/processes	Permet de récupérer les démarches

# Les API HubEE

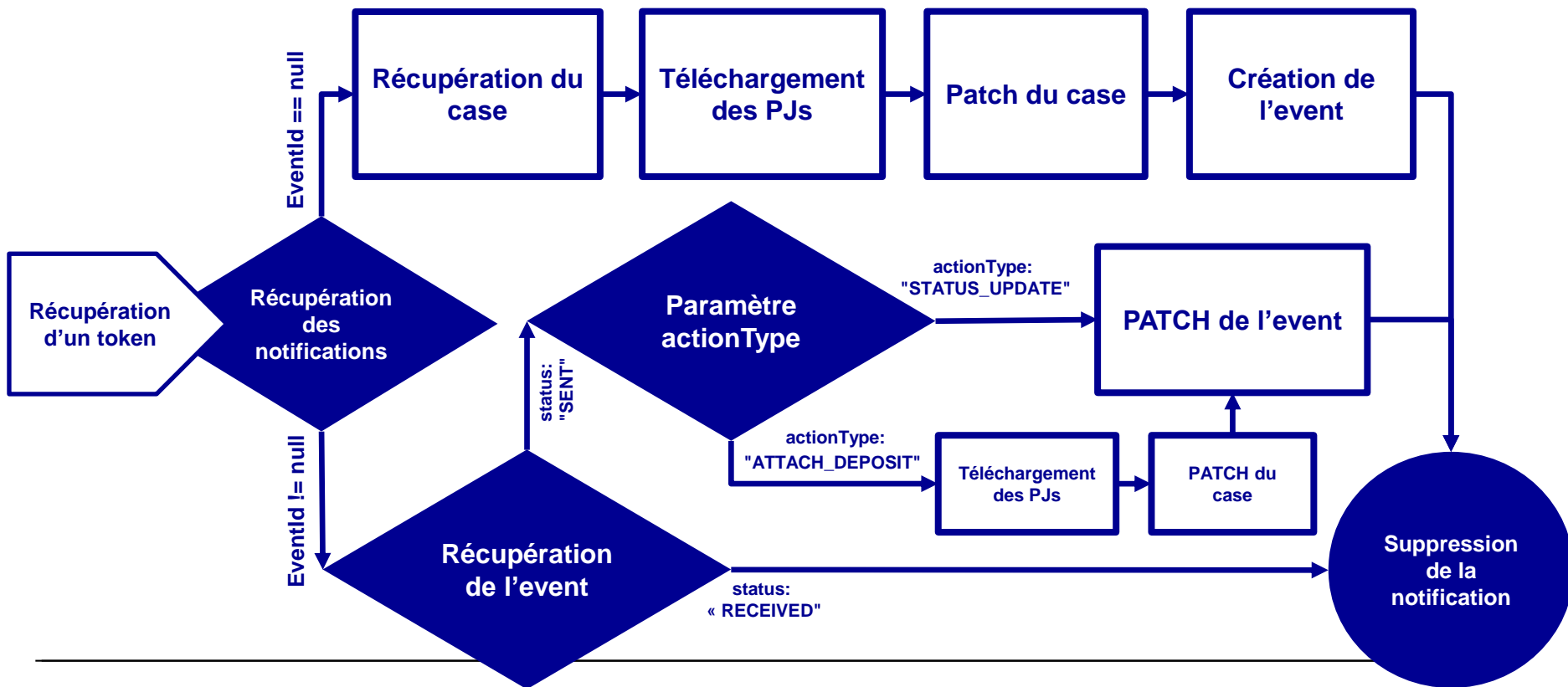
## API Folder

### ■ Les points d'attention :

- Principes des API REST standards (Ressource et méthode)
  - Informations techniques à renseigner dans le header des requêtes
  - Attention au type de body : application/json ou application/octet-stream
  - Le swagger est disponible sur le GIT publique (CF annexe-1)
-

# Les cinématiques

## Cinématique global de récupération d'un nouveau télédossier



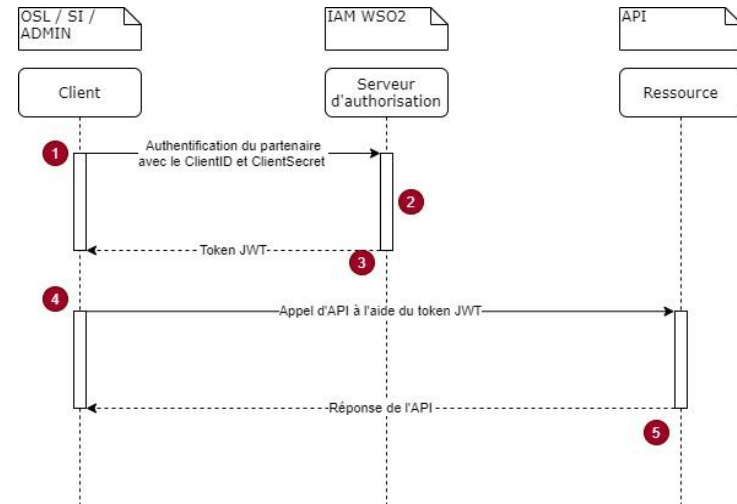


# Les API HubEE

## Authentification HubEE : pour les SI - OAuth2

**POST** : <https://auth.bas.hubee.numerique.gouv.fr/oauth2/token>

- 1 Le client renseigne :  
**ClientId** : à récupérer via le portail hubee (cf annexe-2)  
**ClientSecret** : à récupérer via le portail hubee (cf annexe-2)  
**Scope** : SI
- 3 Le serveur vérifie l'authenticité des informations et délivre un **token** valide **1 heure** en cas de **succès**. En cas **d'erreur** un **401 Unauthorized** est renvoyé
- 4 Il est obligatoire d'associer son token avec chaque requête
- 5 L'utilisation d'une requête sans un token valide entraine automatiquement un cas **d'erreur** et un **401 Unauthorized** est renvoyé

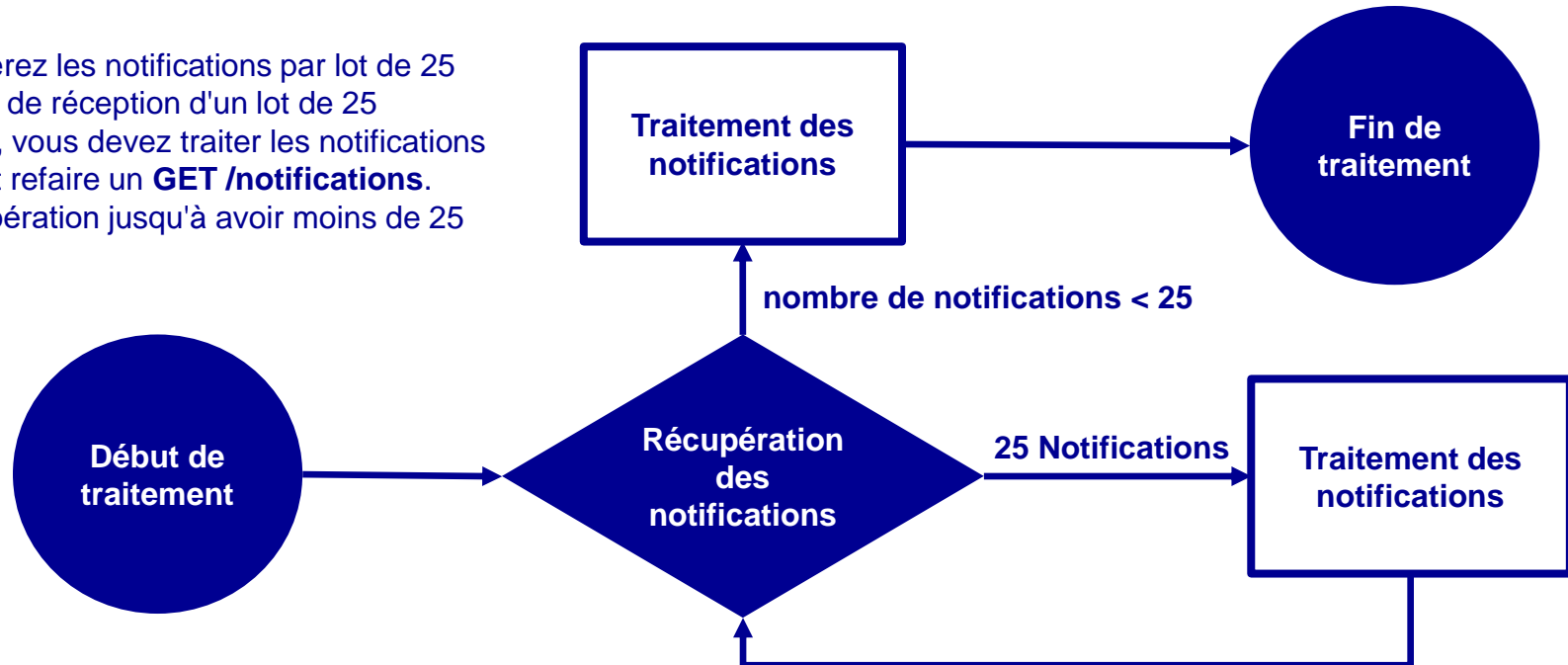


# Les cinématiques

## Récupération des notifications par un SI

GET : /notifications

Vous récupérez les notifications par lot de 25 max, en cas de réception d'un lot de 25 notifications, vous devez traiter les notifications présentes et refaire un **GET /notifications**. Répétez l'opération jusqu'à avoir moins de 25 notifications



# Les cinématiques

## Traitement des notifications par un SI

GET : /notifications

Il existe deux grandes cinématiques suivant la notification :

- Si le paramètre eventId d'une notification est **null**, vous devez récupérer le nouveau télé dossier.
- Si le paramètre eventId d'une notification est **différent de null**, il s'agit d'un échange d'information sur un télé dossier existant entre un SI et l'OSL, vous devez traiter ce point avec la gestion des events

### Notification avec eventId

```
{  
  "id": "3fa85f64-5717-4562-b3fc-2c963f66afa7",  
  "processCode": "CERTDC",  
  "caseId": "sdeabb0c-5dd1-4ere0-9fg5-22526fb52494",  
  "eventId": "a66abb0c-52d1-4e50-9195-22526fb7ce99",  
  "dateTime": "2022-03-01T16:24:44.521Z"  
}
```

### Notification sans eventId

```
{  
  "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",  
  "processCode": "CERTDC",  
  "caseId": "uueabb0c-5dd1-4ere0-9fg5-22526fb52492",  
  "eventId": null,  
  "dateTime": "2022-03-01T16:24:44.521Z"  
}
```

# Les cinématiques

## Traitement d'un nouveau télédossier

Identifier une notification avec le paramètre eventId à null

récupération du paramètre caseId et Id (Id de la notification)

**GET** : /cases/{caseId}

```
"attachments": [  
  {"id": "a66abb0c-52d1-4e50-9195-22526fb7ce92",  
   "status": "RECEIVED",  
   "fileName": "file.pdf",  
   "type": "VA_CertificatdeDeces",  
   "size": 1024,  
   "mimeType": "application/pdf"},  
  {"id": "a66abb0c-52d1-4e50-9195-22526fb7ce92",  
   "status": "RECEIVED",  
   "fileName": "file.pdf",  
   "type": "VA_CertificatdeDeces",  
   "size": 1024,  
   "mimeType": "application/pdf"}  
],
```

# Les cinématiques

## Téléchargement des pièces jointes

Téléchargement unitaire d'une pièce jointe

récupération du paramètre Id (qui correspond à l'attachmentId de la pièce jointe)

**GET** : /cases/{caseId}/attachments/{attachmentId}

Référez vous au paramètre Type pour en déduire la PJ transmise et l'intégrer dans votre infrastructure au bon endroit (*CF Annexe 3*). Vous devez répéter l'opération autant de fois qu'il y a de PJs

```
"attachments": [  
  {"id": "a66abb0c-52d1-4e50-9195-22526fb7ce92",  
    "status": "RECEIVED",  
    "fileName": "file.pdf",  
    "type": "VA_CertificatdeDeces",  
    "size": 1024,  
    "mimeType": "application/pdf"},  
  {"id": "a66abb0c-52d1-4e50-9195-22526fb7ce92",  
    "status": "RECEIVED",  
    "fileName": "file.pdf",  
    "type": "VA_CertificatdeDeces",  
    "size": 1024,  
    "mimeType": "application/pdf"  
  }],
```

# Les cinématiques

## Mise à jour du statut d'un télédossier

**PATCH : /cases/{caseld}**

```
{  
  "status": "SENT"  
}
```

Liste des différents statuts utilisables sur HUBEE (*Slide 24*)

---

# Les cinématiques

## Création d'un event pour envoyer un message

**POST** : /cases/{caseId}/events

```
{  
  "actionType": "SENDING_MESSAGE",  
  "author": "Service instructeur X ou OSL",  
  "notification": true,  
  "message": "Ceci un message à destination du SI ou de l'OSL"  
}
```

Le paramètre `actionType` : `SENDING_MESSAGE` est utilisé dans le cas d'un event qui porte un message vers l'OSL ou vers le SI

Le paramètre `message` va être communiqué au SI ou à l'OSL

# Les cinématiques

## Création d'un event pour un changement de statut simple

**POST** : /cases/{caseId}/events

```
{  
  "actionType": "STATUS_UPDATE",  
  "author": "Service instructeur X",  
  "notification": true,  
  "caseCurrentStatus": "HUBEE_NOTIFIED",  
  "caseNewStatus": "SENT",  
  "message": "Changement de statut à SENT"  
}
```

Le paramètre `actionType` : `STATUS_UPDATE` est utilisé dans le cas d'un event qui change seulement le statut d'un télédossier. `ATTACH_DEPOSIT` est utilisé dans le cas d'envoi d'une PJ en retour vers l'OSL ou vers le SI

Le paramètre `caseCurrentStatus` correspond au statut du télédossier avant votre changement de statut

Le paramètre `caseNewStatus` correspond au statut du télédossier que vous venez d'envoyer

Le paramètre `message` doit être explicite sur l'action que vous êtes en train de faire



# Les cinématiques

## Création d'un event pour un changement de statut et partage d'une pièce jointe

1

**POST** : /cases/{caselId}/events

```
{
  "actionType": "ATTACH_DEPOSIT",
  "author": "Service instructeur X",
  "notification": true,
  "caseCurrentStatus": "HUBEE_NOTIFIED",
  "caseNewStatus": "SENT",
  "message": "Changement de statut à SENT"
}
"attachments": [{
  "fileName": "ER20190860001_20190517_1512_DD.xml",
  "type": "DD_Certificatdedeces",
  "size": 988,
  "mimeType": "application/xml"}]
```

Le paramètre actionType : ATTACH\_DEPOSIT est utilisé pour envoyer des pieces jointes vers l'OSL ou vers le SI

Le paramètre type est alimenté suivant le besoin vu entre le SI et l'OSL (CF Annexe 2).

2

Au succès de création de l'event il faut récupérer le AttachmentId et l'id de l'event

**PUT** :

/cases/{caselId}/events/{eventId}/attachments/{attachmentId}

L'upload est à répéter autant de fois qu'il y a de PJs

**PATCH** : /cases/{caselId}/events/{eventId}

```
{
  "status": "SENT"
}
```

Modification du statut de l'event pour envoyer la notification au destinataire



# Les cinématiques

## Création d'un event sans changement de statut et partage d'une pièce jointe

1

**POST** : /cases/{caselId}/events

```
{
  "actionType": "ATTACH_DEPOSIT",
  "author": "Service instructeur X",
  "notification": true,
  "caseCurrentStatus": "SI_RECEIVED",
  "caseNewStatus": "SI_RECEIVED",
  "message": "Voici ci-joint mon retour de PJ"
}
"attachments": [{
  "fileName": "ER20190860001_20190517_1512_DD.xml",
  "type": "DD_Certificatdedeces",
  "size": 988,
  "mimeType": "application/xml"}]
```

Le paramètre actionType : ATTACH\_DEPOSIT est utilisé pour envoyer des pieces jointes vers l'OSL ou vers le SI

Le paramètre type est alimenté suivant le besoin vu entre le SI et l'OSL (CF Annexe 2).

2

Au succès de création de l'événement il faut récupérer le AttachmentId et l'id de l'événement

**PUT** :

/cases/{caselId}/events/{eventId}/attachments/{attachmentId}

L'upload est à répéter autant de fois qu'il y a de PJs

**PATCH** : /cases/{caselId}/events/{eventId}

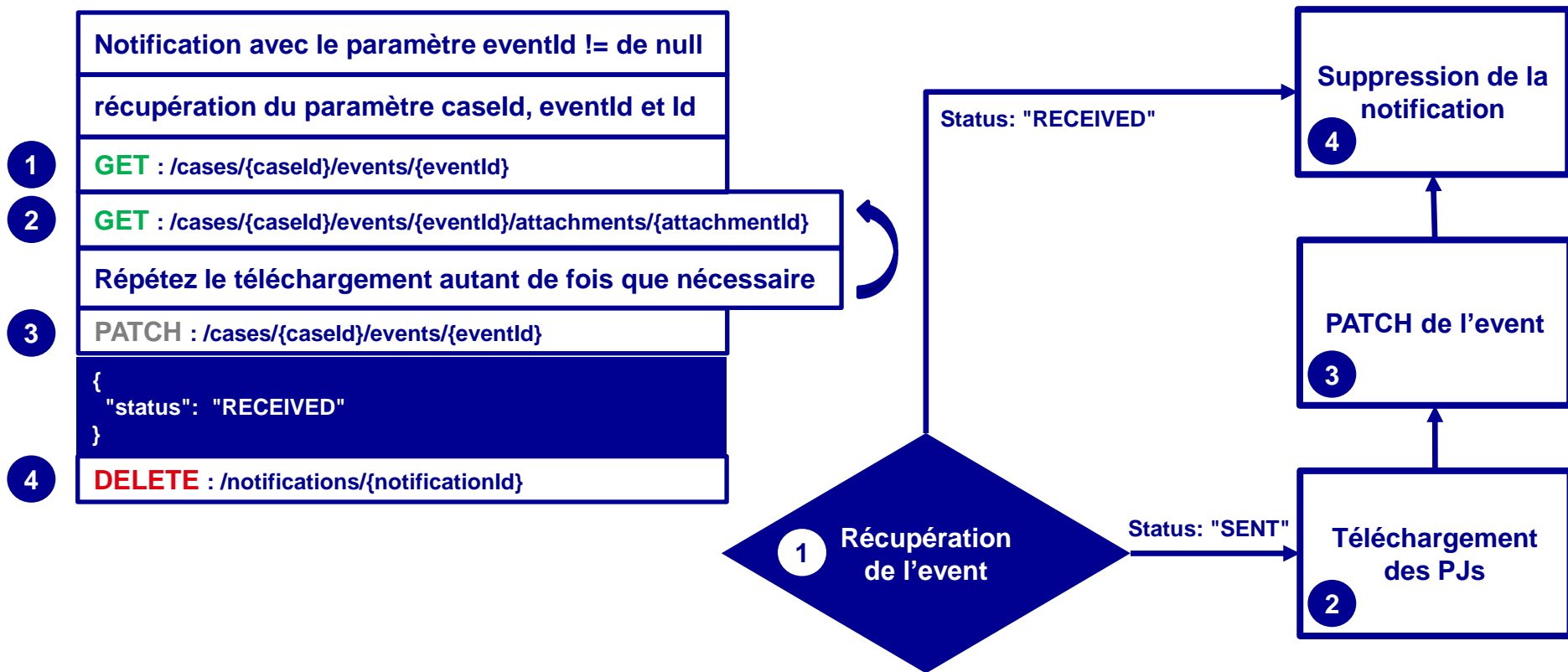
```
{
  "status": "SENT"
}
```

Modification du statut de l'événement pour envoyer la notification au destinataire



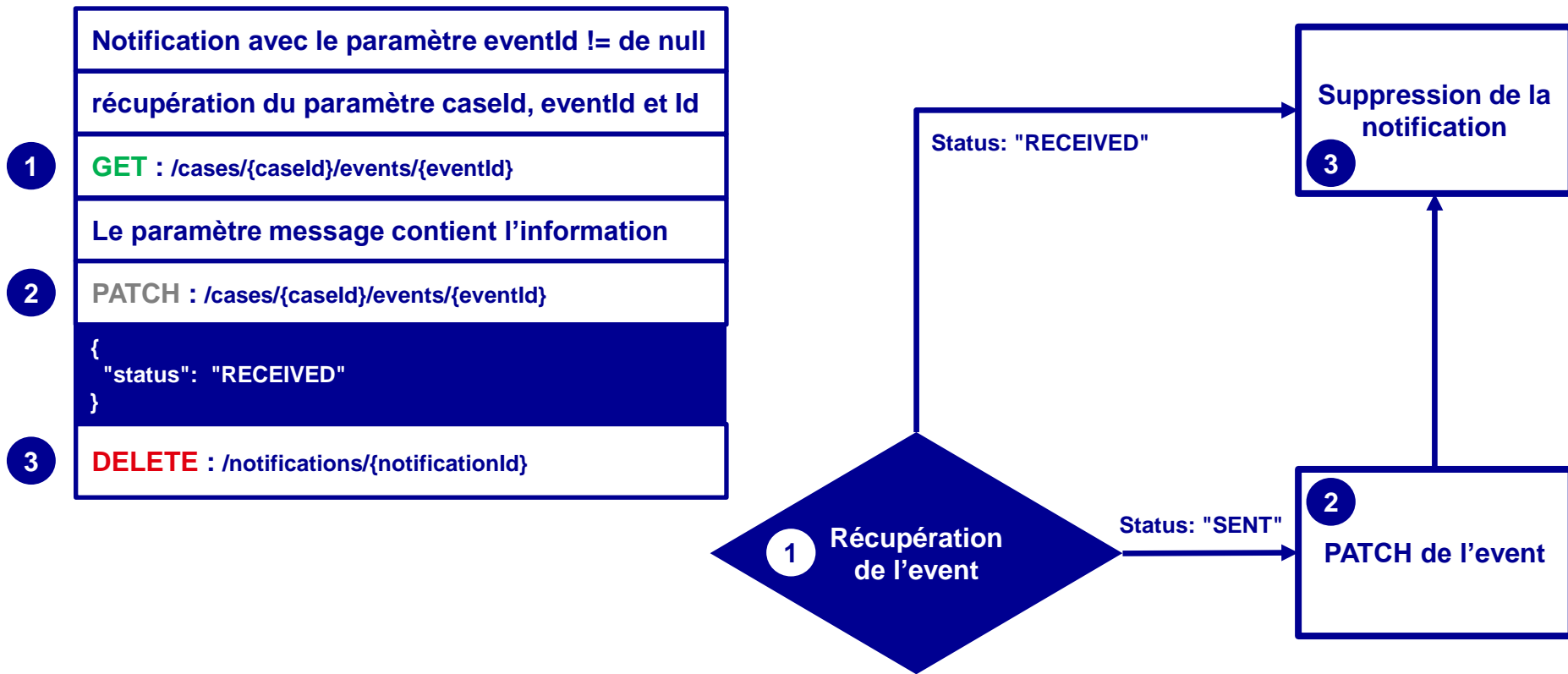
# Les cinématiques

Traitement d'un télé dossier existant avec un event actionType : ATTACH\_DEPOSIT



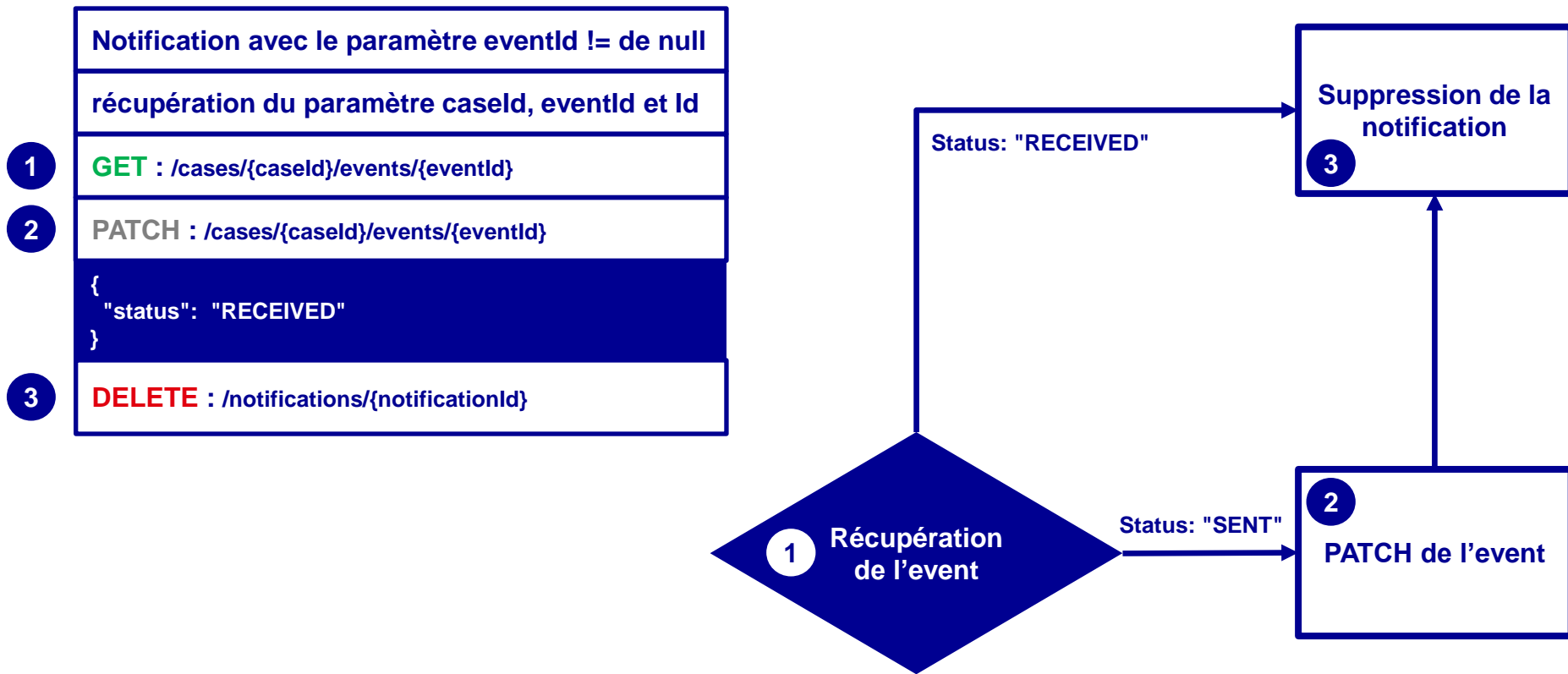
# Les cinématiques

Traitement d'un télé dossier existant avec un event actionType : SENDING\_MESSAGE



# Les cinématiques

## Traitement d'un télédossier existant avec un event STATUS\_UPDATE



# Les cinématiques

## Fermeture d'un télé dossier par un SI

Il est obligatoire de fermer le télé dossier à la fin de son traitement

**PATCH** : /cases/{caseId}/events

```
{  
  "status": "DONE",  
}
```

**POST** : /cases/{caseId}/events

```
{  
  "actionType": "STATUS_UPDATE",  
  "author": "Service instructeur X",  
  "notification": true,  
  "caseCurrentStatus": "IN_PROGRESS",  
  "caseNewStatus": "DONE",  
  "message": "Changement de statut à SENT"  
}
```

Liste des statuts possible pour le caseCurrentStatus, caseNewStatus et status (Slide 24).

Le paramètre caseCurrentStatus correspond au statut du télé dossier avant votre changement de statut

Le paramètre caseNewStatus correspond au statut du télé dossier que vous venez d'envoyer

Le paramètre message doit être explicite sur l'action que vous êtes en train de faire

# Les cinématiques

## Suppression de la notification

A la fin de chaque grande cinématique, vous devez absolument supprimer la notification

**DELETE** : /notifications/{notificationId}

---

# Le Hub d'Échange de l'État

## Les statuts des dossiers

Identifiant technique	Libellé	Description	Statuts entrants	Acteur du changement
HUBEE_RECEIVED	Reçu par le HUBEE	L'OSL soumet le TD au HUBEE via l'API. Le HUBEE transmet l'Id Tech du TD en retour à l'OSL à la création	N/A	HUBEE
HUBEE_COMPLETED	Complété	Le stockage des données du TD et de ses PJ est effectué au sein du HUB	HUBEE_ENRICHMENT	OSL
HUBEE_NOTIFIED	Notifié (au SI)	La liste des notifications a été mise à jour par le HUBEE	HUBEE_COMPLETED	HUBEE
SENT	Transmis (au SI)	Les metadatas du TD ont été récupérées via l'API du SI (portail SI ou progiciel SI)	HUBEE_NOTIFIED	SI
SI_RECEIVED	Reçu (par un agent du SI)	Le télé dossier est intégré correctement dans le système d'information du service instructeur. A l'ouverture/Consultation du TD un acquittement de réception est transmis automatiquement au HUBEE.	SENDED	SI
IN_PROGRESS	En cours	Après acquittement de réception et téléchargement des pièces associées au télé dossier, celui-ci est considéré en cours de traitement par le service instructeur	SI_RECEIVED	SI
ADD_AWAITING	En attente de compléments	Un message est envoyé à l'utilisateur en complément du changement de statut afin qu'il complète son dossier initial. L'OSL sera donc notifié de ce changement afin qu'il transmette le message à l'utilisateur.	IN_PROGRESS	SI
<b>REFUSED</b> (statut final)	Refusé	Après examen, le télé dossier ne peut pas être traité, dans la majorité des cas par défaut d'information (pièce justificative manquante, donnée obligatoire non renseignée ou erronée) ou pour cause de non-éligibilité de l'utilisateur (demande non légitime ou inadaptée à la situation)	SI_RECEIVED / IN_PROGRESS / ADD_AWAITING	SI
<b>DONE</b> (statut final)	Traité	Le traitement du télé dossier est terminé, il n'y a plus d'action côté service traitant, hors remontée de l'information à l'utilisateur et des pièces administratives à retourner (attestations, certificat, carte, etc.)	SI_RECEIVED (hors portail SI) / IN_PROGRESS / ADD_AWAITING	SI
CLOSED	Clos	Le télé dossier est clos dès lors que l'utilisateur a été informé de la fin de son traitement et a reçu les pièces administratives	SI_RECEIVED / IN_PROGRESS / ADD_AWAITING / DONE / REFUSED	OSL
<b>SI_INTG_ERROR</b>	Erreur d'intégration	L'intégration du télé dossier dans votre SI n'est pas possible	SI_RECEIVED / IN_PROGRESS / ADD_AWAITING / DONE	SI



# Contact du support

Pour toute demande, y compris la mise à disposition d'un compte de test, l'équipe Projet HubEE se tient à votre disposition via l'adresse générique **[contact@hubee.numerique.gouv.fr](mailto:contact@hubee.numerique.gouv.fr)**

En cas d'anomalie, nous vous proposons de nous transmettre la fiche complétée avec les éléments ci-dessous :

## Description

- **Environnement** : *Recette*
- **Date et heure** :
- **Jeux de données utilisés** :
- **Description de l'anomalie** :  
*Le cheminement qui a permis de détecter l'anomalie, étapes par étapes  
Ne pas hésiter à partir du principe que même une personne ne connaissant pas  
l'application pourra reproduire l'anomalie.*
- **Résultat observé** :  
*Expliquer le dysfonctionnement constaté*
- **Résultat attendu** :  
*Expliquer, décrire le résultat normalement attendu  
Règle de gestion appuyant cet attendu*
- **Les pièces jointes** :  
*Toute information utile permettant d'identifier l'anomalie*



# PREMIER MINISTRE

*Liberté  
Égalité  
Fraternité*

Isabelle Carton

Directrice de Projet : Hub d' Échange de l' État

[Isabelle.carton@modernisation.gouv.fr](mailto:Isabelle.carton@modernisation.gouv.fr)

[contact@hubee.numerique.gouv.fr](mailto:contact@hubee.numerique.gouv.fr)

Tél : 06.89.22.08.05

Direction interministérielle du numérique

# Annexe - 1

Git publique : <https://github.com/dinum-HubEE/Documentations>

Swagger : <https://github.com/dinum-HubEE/Documentations/releases/latest/download/swagger.zip>

Valise générique : <https://github.com/dinum-HubEE/Documentations/releases/latest/download/valiseGenerique.zip>

# Annexe - 2

Environnement : <https://github.com/dinum-HubEE/Documentations#les-environnements>

Liste des statuts disponibles : à venir