

Icanopee

- Esanté Connect - Interface REST
- Document <> DMP
 - Envoie d'un document sur le DMP
 - Structure document
 - StructuredBody
 - Supprimer un document
- Efficiency

Esanté Connect – Interface REST

Esanté Connect – Interface REST est une application autonome à installer sur un serveur exploité par le client.

Le service exposé est une API REST en JSON.

Les Fonctions exposées permettent l'accès au DMP à partir d'une application (web ou client lourd) en **authentification indirecte** via un certificat logiciel d'établissement.

Installation sur un poste dev

La procédure suivante décrit les étapes pour la version Linux du connecteur, une version darwin est aussi disponible sur l'espace Icanopée

- Sur <https://clients.icanopee.net/produit/437/builds/dist>
 - Récupérer sur le zip **dmpconnect-es-rest-mss-insi-<version>-Linux.zip**
 - Récupérer le zip **dmpconnect-es-rest_sdk-mss-insi-<version>.zip**
 - Sur <https://clients.icanopee.net/produit/437/configuration>
 - Récupérer le fichier **dcparameters.esrest.v2.dev.reconnect_esrest.dat**
1. Copier le fichier **dcparameters.esrest.v2.dev.reconnect_esrest.dat** dans le dossier **DmpConnect-ES-REST-MSS-INSI-1.10.0-Linux** en le renommant **dcparameters.dat**
 2. Lancer l'exécutable **dmpconnect-es-rest**. À ce stade, le serveur devrait être lancé.
 3. Pour tester rapidement, ouvrir le fichier **sdk/exemples_js/examples.html** dans un navigateur et lancer les exemples en cliquant sur les liens. Les requêtes/réponses effectuées seront visible dans la console du navigateur
 4. Il est possible de modifier certains paramètres du serveur en éditant le fichier **dmpconnect-esrest.xml** (port d'écoute, niveau de logs etc...). L'exécutable **dmpconnect-es-rest** doit être relancé après chaque modification.

Par défaut, l'adresse pour effectuer des requêtes au connecteur est **https://0.0.0.0:9979**

Il est tout à fait possible de tester de construire les requêtes vers le connecteur avec un client HTTP (ici Postman)

The screenshot displays a REST client interface with a POST request to `https://0.0.0.0:9979/getDmpStatus`. The request body is a JSON object containing details for a medical professional and their practice location. The response is a JSON object with administrative data and status information.

```
4 ..... "HP":
5 ..... {
6 .....   "s_name": "MED-CS RPPS0020810",
7 .....   "s_given": "ANNE",
8 .....   "s_codeProfession": "10",
9 .....   "s_professionOid": "1.2.250.1.71.1.2.7",
10 .....  "s_codeSpecialty": "SM04",
11 .....  "s_functionCodeOid": "1.2.250.1.213.1.6.1.17",
12 .....  "s_internalId": "Agent007",
13 .....  "i_internalIdType": 0,
14 .....  "i_authenticationMode": 8,
15 .....  "s_serviceName": "Service de cardiologie"
16 .....  },
17 .....  "PracticeLocation":
18 .....  {
19 .....    "s_name": "Hopital du Val de Grace",
20 .....    "s_activitySector": "SA43",
21 .....    "s_practiceSettings": "AMBULATOIRE"
22 .....  }
23 .....  },
24 .....  "s_dmpUrl": "https://lps.formation1.dmp.gouv.fr/si-dmp-servez/v2/services/"
```

Body Cookies Headers (11) Test Results Status: 200 OK Time: 360 ms Size: 706 B Save as example

```
1 {
2   "AdminData": {
3     "Ins": {
4       "s_ins": "188102B17295165",
5       "s_insType": "T"
6     }
7   },
8   "i_closingCode": 1,
9   "i_dmpStatus": 1,
10  "i_ensRelatedDmp": 0,
11  "i_userAuthorization": 2,
12  "i_webAccessIsOpen": 0,
```

La documentation plus détaillée est visible sous **sdk/doc**.

Installation sur nos serveurs

Le dossier du connecteur se trouve sous `~/rest_connector/`

Un service systemd (**dmp_connector**) est en cours de création actuellement (il nécessitera d'être dans un premier temps activé avec **sudo systemctl enable dmp_connector.service**).

Ensuite il sera possible de gérer ce service grâce aux différentes commandes **stop**, **restart**, **status**, ou encore consulter les logs via **journalctl**.

Document <> DMP

Envoie d'un document sur le DMP

/sendsubmissionset

La route /sendsubmissionset permet l'envoi d'un document ou d'une liste de documents sur le DMP d'une personne

```
{  
  
  "EsUser": {},  
  "s_dmpUrl": "",  
  "i_timeout": 30,  
  "s_dmpEsId": "",  
  "AdditionalPatientIdentifiers": [ // (Optionnel) : Permet de spécifier des identifiants secondaires pour le patient  
    (IPP, NIP, ...)  
    {  
      "s_extension": "",  
      "s_root": ""  
    }  
  ],  
  "Documents": [],  
  "ReferencedDocuments": [], // (Optionnelle) Liste des documents à référencer dans ce lot de soumission. Ce  
  sont des documents qui doivent exister dans le DMP. Chaque référence est définie par :  
    "Identity": {}, // (Optionnel) Permet de spécifier l'identité du patient  
    "s_birthPlaceCode": "", (Optionnel) Code du lieu de naissance (COG INSEE). Obligatoire si l'envoi contient des  
  pièces jointes et que les informations du patient n'ont pas été spécifiés avec Identity.  
    "i_disablePdfa1Conversion": 0, // (Optionnel) Si vrai (1) la conversion des documents PDF en PDF/A-1 est  
  désactivée. Il est alors de la responsabilité de l'éditeur de s'assurer que les documents PDF sont bien au format  
  PDF/A-1.  
    "i_pdfa1IgnoreTransparency": 0, // (Optionnel) Si vrai (1) la transparence est ignorée lors de la conversion en  
  PDF/A-1. Ceci accélère le processus mais peut générer des documents illisibles.  
    "i_pdfa1ImageResolution": 0, // (Optionnel) Indique en dpi la résolution des images à utiliser lors de la
```

conversion en PDF/A-1. Par défaut cette valeur est fixée à 200dpi.

"i_forceSchematronsValidation": "", (Optionnel) Par défaut, la validation par schématrons est désactivée pour les documents non structurés (txt, rtf, pdf, jpg et tiff), garantis conformes par le connecteur. Si

i_forceSchematronsValidation est vrai (1) alors la validation par schématrons est aussi active sur ces documents.

"i_getCdaContent": 0, // (Optionnel) Permet d'activer la sortie de chaque document envoyé au format CDA en retour de la fonction.

"i_getDocumentContent": 0, // (Optionnel) Permet de récupérer le contenu haut niveau des documents envoyés.

"i_getVihf": 0, // (Optionnel) Si vrai (1), le VIHf de la transaction DMP TD2.1 est retourné.

"i_getXdsMetadata": 0, // (Optionnel) Si vrai (1), les métadonnées XDS de la transaction DMP TD2.1 est retourné.

"i_retrieveDocumentUuid": 0, // (Optionnel) Permet, en effectuant une transaction supplémentaire par document, d'obtenir l'identifiant technique (UUID) de chaque document envoyé.

"i_transcodeTypeCode": "", // (Optionnel) Si vrai (1) le code de catégorie des documents (champ s_category) est transformé avant l'envoi afin de garantir que les codes utilisés sont toujours les plus à jour. (Par défaut : off).

"i_automaticPdfCopyEmbedding": 0, (Optionnel) Si vrai (1) le connecteur génère et ajoute automatiquement une version PDF (une « copie ») de chaque document CDAR2 N3 dans une section structurée de type FR-Document-PDF-copie.

"s_healthcareSetting" : "", // Cadre de soin lors du dépôt de document. (Cf. Cadre de soin (Healthcare settings) dans la documentation des jeux de valeurs).

"s_ins" : ""

"s_localPatientId" : "", // (Optionnel) Identifiant (unique) du patient dans le SI local.

"s_localPatientRootOid" : "", // (Optionnel) Racine des OID des Id des patients dans le SI local.

"s_submissionSetDescription" : "", // (Optionnel) Description du lot de soumission.

"s_submissionSetOid" : "", // (Optionnel) OID du lot de soumission. A utiliser dans le cas où l'établissement possède sa propre racine d'OID.

"s_submissionSetTitle" : "" //(Optionnel) Titre du lot de soumission

}

Si on ne conserve que les champs obligatoires, on obtiens cela :

```
{  
  
  "EsUser": {},  
  "s_dmpUrl": "",  
  "i_timeout": 30,  
  "s_dmpEsId": "",  
  "Documents": [],  
  "s_ins" : ""
```

```
}
```

Concernant le champ "Documents", voir [la page concernant la structure des documents](#)

Retour :

L'appel retourne cette structure de données :

```
{
  "Uniquelds" : [], // La liste des identifiant locaux (Uniqueld) des documents déposés (dans le même ordre que celui des documents en entrée).
  "Uuids" : [], // Seulement présent si l'option i_retrieveDocumentUuid est active. Il s'agit de la liste des identifiants techniques (UUID) des documents déposés (dans le même ordre que celui des documents en entrée).
  "CdaContentsInBase64" : [], // Seulement présent si l'option i_getCdaContent est active : la liste des documents envoyés au format CDA en base64 (dans le même ordre que celui des documents en entrée).
  "DocumentsContent" : [], // Seulement présent si l'option i_getDocumentContent est active. Pour chaque document déposé, sa structure haut niveau est retournée. L'ordre est identique à celui des documents en entrée. La structure haut niveau est identique au retour de la fonction /getDocumentFromCda (Cf. /getdocumentfromcda).
  "i_vihfInBase64" : "", // Seulement présent si l'option i_getVihf est active. Contient le VIHf. Le champ est encodé en base64.
  "i_xdsMetadataInBase64" : "", // Seulement présent si l'option i_getXdsMetadata est active. Contient les métadonnées XDS de l'envoi DMP. Le champ est encodé en base64.
  "s_status" : "OK"
}
```

Notes :

Dans le cas où un document est défini à partir de son CDA complet (i.e.

[via le champ s_cdaContentInBase64) il est nécessaire que l'utilisateur connecté (i.e. EsUser.HP) soit l'un des auteurs du document. La vérification est effectuée côté serveur et porte à minima sur les champs suivants :]

- Nom du professionnel (EsUser.HP.s_hpName)
- Prénom du professionnel (EsUser.HP.s_hpGiven)
- Identifiant du professionnel (EsUser.HP.s_hpInternalId et EsUser.HP.i_hpInternalIdType)
- Nom de la structure du professionnel (EsUser.PracticeLocation)

- Identifiant de la structure du professionnel (EsUser.PracticeLocation.s_practiceLocationStructureId ou identifiant du certificat d'authentification)
Si au moins l'un des champs ne correspond pas alors le document est rejeté par le serveur.

Dans le cas où l'ajout d'une copie PDF du document est demandée (option i_automaticPdfCopyEmbedding) alors :

- celle-ci est effectuée à partir de la feuille de style de l'ANS;
- elle n'est effectuée que pour des documents structurés CDAR2 N3 (ex : VSM, CrBio).

Afin d'être conforme avec le référentiel INS, depuis la version 1.9.0 du connecteur, le code du lieu de naissance (COG INSEE) est obligatoire en entrée.

Le code peut être spécifié de deux façons :

- Dans le bloc Identity (champ s_birthPlace).
- Dans le champ s_birthPlaceCode si le champ Identity n'est pas utilisé.

Les identifiants doivent être conservés en cas de suppression des dits documents

Si la structure Identity est fournie et si le connecteur dispose de l'extension INSI

- alors les données du patient spécifiées dans les documents sont obtenues à partir de cette structure. Dans le cas contraire, les données du patient sont obtenues à l'aide de la transaction DMP TD0.2 (automatiquement appelée avant l'envoi des documents). Les données issues de l'INSi incluent en particulier la liste des prénoms, contrairement à la TD 0.2 qui ne retourne qu'un seul prénom.

Les serveurs DMP limitent la tailles des documents CDA à 8 Mo.

- Si le CDA fait plus de 8 Mo alors une erreur XDSRepositoryOutOfResources est retournée.
- La taille maximale est celle du CDA, c'est-à-dire celle du document et des métadonnées CDA.
- Dans le cas d'un document brut (JPEG, RTF, PDF), le corps du CDA est constitué d'une version encodée en base64 du document à envoyer. Du fait de l'encodage base64, la version envoyée au DMP est 4/3 plus grande que celle du document. Ainsi, la limite pour le document brut est réduite approximativement à 6 Mo (aux métadonnées CDA près).
- Dans le cas d'un PDF, le connecteur effectue une conversion en PDF/A1 par défaut, or cette conversion peut produire un document plus gros que le document original.

Structure document

La structure **Document** permet de représenter un document. La structure contient les champs suivants :

```
{
  "s_title": "", // Titre du document.
  "s_description", // (Optionnel) Description du document.
  "s_category", // Catégorie du document
  "i_visibility": , // Code de visibilité
  "i_format": , // Code correspondant au format du doc
  "s_cdaContentInBase64": , //( Optionnel) Contenu complet du document CDA.
  "s_contentInBase64": , // (Exclusif avec StructuredBody) Contenu du document au format base64
  "StructuredBody": , // (Exclusif avec s_contentInBase64) Corps structuré du document
  "s_pdfCopyContentInBase64": , // (Optionnel) Copie PDF d'un document structuré.
  "s_stylesheetsInBase64": , // (Optionnel) Feuille de style au format xsl :stylesheet.
  "s_replacedDocumentUniqueId": , // (Optionnel) identifiant unique du document remplacé (dans le cas d'un
remplacement de document)
  "s_creationDate": , // (Optionnel) Date de création du document. La date doit être dans le format
AAAAMMJJhhmmss[+-]hhmm Heure locale (avec secondes) avec décalage par rapport au temps UTC.
  "s_serviceStartDate": , // (Optionnel) Date de début de l'acte médical correspondant au document
  "s_serviceStopDate": , // (Optionnel) Date de fin de l'acte médical correspondant au document
  "s_oid": , // (Optionnel) OID du document. A utiliser dans le cas où l'établissement dispose de sa propre racine
d'OID
  "s_versionNumber": , // (Optionnel) Numéro de version du document.
  "s_setIdRoot": , // (Optionnel) Identifiant de révision du document. Il s'agit d'un OID complet ou d'une racine
d'OID
  "s_setIdExtension": , // (Optionnel) Identifiant de révision du document. Il s'agit d'un complément d'OID dans le
cas où s_setIdRoot n'est qu'une racine d'OID
  "Performer": { // Pour chaque document il est nécessaire de définir celui qui est à l'origine de la création du
document. Le per- former est composé de deux champs : HP et i_role.
    "HP": {}, // Le champ HP est décrit dans la section Structure HP (cf. Structure HP).
    "i_role": // (Optionnel - Déprécié) Rôle du PS dans l'acte médical (code)
  }
  "Authors": [], // La liste des auteurs de l'acte médical référencé par le document. Les auteurs sont des objets de
type HP, décrit dans la section Structure HP (cf. Structure HP)
```

```

    "EventCodes": [], // (Optionnel) Tableau des codes de classification pour le document. Chaque code est une
    structure EventCode. Cf. Structure EventCode.
    "Informants": [], // (Optionnel) Liste des informateurs du document. La structure Informant est décrite dans la
    section Structure Informant.
    "PatientAddresses": [], // (Optionnel) Adresse(s) postale(s) du patient. Il s'agit d'un tableau de structure
    Address (cf. Structure Address).
    "PatientTelecoms": [], // (Optionnel) Adresse(s) de télécommunication du patient. Il s'agit d'un tableau de
    structure Telecom (cf. Structure Telecom).
    "TreatingPhysician": {}, // (Optionnel) Médecin traitant. Nécessaire pour certains documents (DLU par ex). Il
    s'agit d'une structure HP (cf. Structure HP).
    "IntendedRecipients": [], // (Optionnel) Destinataires en copie du message. Il s'agit d'un tableau de structure
    HP (cf. Structure HP).
    "DataEnterer": {}, // (Optionnel) Opérateur de saisie. Il s'agit d'une structure DataEnterer (cf. Structure
    DataEnterer).
    "Authenticators": [], // (Optionnel) Personne attestant de la validité des informations du document sans en
    endosser la responsabilité. Il s'agit d'un tableau de structure Authenticator (cf. Structure Authenticator)
    "Participants": [], // (Optionnel) Participants additionnels (autre que ceux déjà référencés ailleurs). Il s'agit d'un
    tableau de structure Participant (cf. Structure Participant)
    "Referencelds": [], // (Optionnel) Tableau des références (cf. Structure Referenceld).
    "Orders": [], // (Optionnel) Tableau des références de prescription (cf. Structure Order).
    "ImagingActs": [], // (Optionnel) Tableau des actes d'imagerie (cf. Structure ImagingAct).
    "i_action": , // Action demandée par l'émetteur de l'archive, que le LPS qui réceptionnera le document devra
    effectuer. Il s'agit d'une valeur dans le tableau suivant :
        1 Le document est un nouveau document, il doit être ajouté au LPS.
        2 Le document remplace un document existant, le précédent est identifié par son Uniqueld (cf. s_replacedDo
        4 Le document doit être supprimé du LPS.
    }

```

Si on ne conserve que les champs obligatoires, on obtient :

```

{
    "s_title": "", // Titre du document.
    "s_category", // Catégorie du document
    "i_visibility": , // Code de visibilité
    "i_format": , // Code correspondant au format du doc
    "s_contentInBase64": , // (Exclusif avec StructuredBody) Contenu du document au format base64
    "StructuredBody": , // (Exclusif avec s_contentInBase64) Corps structuré du document
    "Performer": { // Pour chaque document il est nécessaire de définir celui qui est à l'origine de la création du
    document. Le per- former est composé de deux champs : HP et i_role.

```

```
"HP": {}, // Le champ HP est décrit dans la section Structure HP (cf. Structure HP).
}
"Authors": [], // La liste des auteurs de l'acte médical référencé par le document. Les auteurs sont des objets de
type HP, décrit dans la section Structure HP (cf. Structure HP)
"i_action": , // Action demandée par l'émetteur de l'archive, que le LPS qui réceptionnera le document devra
effectuer. Il s'agit d'une valeur dans le tableau suivant :
    1 Le document est un nouveau document, il doit être ajouté au LPS.
    2 Le document remplace un document existant, le précédent est identifié par son UniqueId (cf. s_replacedDo
    4 Le document doit être supprimé du LPS.
}
```

Concernant "**StructuredBody**", [voir la page concernant le corps structuré d'un CDA](#) -> à noter que dans notre cas, les types de documents échangés avec le DMP ne nécessiteront pas de StructuredBody.

Nous devons donc simplement fournir le document au format base64 (**s_contentInBase64**)

StructuredBody

La structure StructuredBody permet de représenter le corps structuré d'un document [CDA](#).

```
{
  "Sections": [], // Liste des sections du corps structuré du document.
  "Immunizations": {}, // (Optionnel) Section de vaccinations. Cf. Structure ImmunizationSection.
  "PdfCopy": {}, // (Optionnel) Section PDF en copie. Cf. Structure PdfCopySection.
  "ActiveIssues": {}, // (Optionnel) Section des problèmes médicaux actifs. Cf. Structure ActiveIssueSection.
  "PreviousIssues": {}, // (Optionnel) Section des antécédants médicaux. Cf. Structure PreviousIssueSection.
  "Allergies": {}, // (Optionnel) Section des allergies et hypersensibilités. Cf. Structure AllergySection.
  "MedicalDevices": {}, // (Optionnel) Section des dispositifs médicaux. Cf. Structure MedicalDevicesSection.
  "ActsHistory": {}, // (Optionnel) Section historique des actes. Cf. Structure ActHistorySection.
  "MedicinalTreatments": {}, // (Optionnel) Section des traitements médicamenteux. Cf. Structure
  MedicinalTreatmentsSection.
}
```

Supprimer un document

/deletedocument

Supprime un document du DMP.

```
}
  "EsUser": {},
  "s_dmpUrl": "",
  "i_timeout": 30,
  "s_dmpEsId": "",
  "s_documentUniqueld": "", // Identifiant unique du document sur le DMP.
  "s_documentUuid": "", // (Optionnel) Identifiant technique du document sur le DMP.
  "s_healthcareSetting": "", // Cadre de soin dans lequel s'effectue la suppression. (Cf. Cadre de soin (Healthcare settings) dans la documentation des jeux de valeurs).
  "s_ins": "", // Numéro INS du patient.
}
```

Si l'identifiant technique du document (s_documentUuid) est fourni alors :

- La suppression s'effectue en une seule transaction DMP (TD3.3c), sinon deux transactions sont nécessaires (TD3.1 puis TD3.3c).
- La suppression est possible même lorsque la structure ne dispose pas d'autorisation d'accès au DMP du patient. Dans le cas contraire, une autorisation d'accès est nécessaire.

Efficiencie

Efficiencie Web est une application web à destination des professionnels de santé, donnant accès aux divers services socles du Cadre d'Interopérabilité des Systèmes d'Information de Santé (CI-SIS) : DMP, DP, MSS, TLSi (INSi, ALDi, ApCV, etc.).

Elle repose sur l'exploitation des connecteurs esante-connect d'icanopée, qui doivent préalablement être déployés.

Elle est développée avec le framework React.

L'application fonctionne avec les connecteurs suivants :

- esanté-connect, interface Web Socket « DmpConnect-JS »
 - Fonctionne en authentification directe par carte CPx.
 - A installer sur le poste des Professionnels de Santé (PS).
- esanté-connect, interface REST « DmpConnect-ES REST »
 - Fonctionne en authentification indirecte par certificat logiciels (d'établissement, ou de personne physique pour l'accès au Dossier Pharmaceutique (DP)).
 - A installer sur un serveur.

Nous utilisons actuellement l'application avec connecteur JS2 installé côté client.

Notre usage d'efficiencie sera principalement pour la mise à disposition de la MSS sur Reconnect Pro, via un Iframe.

Environnement de préproduction :

- Application déployée sur le serveur **book** (DocumentRoot à la racine de l'utilisateur web : ~/insi-dev)

Environnement de production :

- Application déployée sur le serveur **de production RP** (~/efficiencie)